

Opinnäytetyö (AMK)

Tietotekniikan koulutusohjelma

Sulautetut järjestelmät

2011

Tenho Tuhkala

PAIKKATIEDON SIIRTO RADIOLINKIN AVULLA



TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

Tenho Tuhkala

PAIKKATIEDON SIIRTO RADIOLINKIN AVULLA

Nykyinen GSM-tekniikka on mahdollistanut GPS-paikannustiedon käytön maastossa liikkumisen helpottamiseksi, muun muassa yleisesti radioperustaisten koiratutkien korvaajina. GSM/GPS-koiratutka luottaa siihen, että GSM-verkko toimii aina siellä, missä tutkaa käytetään. Tämä ei kuitenkaan ole mahdollista suhteellisen suurilla ja todella harvaan asutuilla alueilla Suomessa, pääasiassa Lapin läänissä. Siellä metsästystä ja maastossa liikkumista tapahtuu alueella, joissa lähimpään toimivaan GSM-tukiasemaan voi olla kymmeniä kilometrejä tai maaston muoto voi aiheuttaa pahojakin katvealueita. Vaikka GSM-määrittelyissä on määritelty pitkätkin kantamat, maaston muoto aiheuttaa herkästi isoja katvealueita ja luonnollisesti tällaisella alueella GSM-verkon käyttöön pohjautuvat tutkat ja paikannuslaitteet eivät toimi oikein.

Tämän opinnäytetyön tarkoituksena on tutkia, onko mahdollista tehdä laitetta, joka on riippumaton GSM-tekniikasta ja onko sitä mahdollista käyttää seuduilla, joissa GSM-verkon kuuluvuus ei ole tarpeeksi kattava. Lisäksi tutkitaan, mitä muuta lisätietoa pystytään radiotiellä siirtämään paikannustiedon lisäksi. Tällaista tietoa voi olla esimerkiksi koiran haukun ilmaisu tai mahdollisesti toisen maastossa liikkujan hätäilmoitus vaaran uhatessa tai vahingon tapahduttua. Laitteen on tarkoitus olla mahdollisimman monikäyttöinen ja päätelaiteriippumaton, joten se voi antaa paikannustietonsa ulos joko Bluetoothin ja/tai sarjaportin avulla NMEA-standardin mukaisessa muodossa.

Laitteen kehittäminen onnistui odotusten mukaisesti, vaikka varsinaista maastotestausta ei päästy tekemään laillisten radiomodeemien puutteen vuoksi. Paikannustiedon lisäksi linkin yli oli erittäin helppo siirtää muutakin tietoa ilman, että se häiritsisi päätelaitteen samaa paikkatietosignaalia.

ASIASANAT:

GPS, paikannus, radiolinkki, hälytys

BACHELOR'S THESIS | ABSTRACT
TURKU UNIVERSITY OF APPLIED SCIENCES

Information Technology | Embedded Systems

December 2011 | 29 pages

Tiina Fern

Tenho Tuhkala

TRANSMITTING POSITION DATA OVER RADIO LINK

Availability of GSM based locators for hunters and other active hikers and rangers has made the activities easier. Those devices basically work well in area where GSM signal strength is good as they use short message services to transfer data between transmitter and receiver. However, there are many areas in Finland, mainly in Lapland, where GSM is not available at all because of landscape or distance to stations.

This purpose of this thesis is to study the possibilities to build such a device which works well also in challenging area and obeys the local laws. Another goal is to find possibility to transfer other information alongside location data, such as notification of dog behaviour or other people's emergency messages. Device itself has multiple connection types over Bluetooth and RS232 serial connection and provides NMEA standard output of location data.

Building and testing device went reasonably well, even missing legal radio modems. Due to lacking modems, field testing was not accomplished in full strength. Transmitting additional data along with positioning data was very easy without disrupting original data transmitted to end user device.

KEYWORDS:

GPS, radiolink, positioning

SISÄLTÖ

SISÄLTÖ	4
LIITTEET	5
KUVAT	5
KUVIOT	5
1 KÄYTETYT LYHENTEET JA SANASTO	6
2 JOHDANTO	7
PAIKANTAMISEN HISTORIAA JA PERUSTEITA	8
2.1 Kolmiomittaus	8
2.2 Paikannus GPS:n avulla	8
3 GPS PAIKANTIMEN RAKENNE JA SUUNNITTELU	10
3.1 Bluetooth	11
3.2 USB portti	11
3.3 RS232 sarjaportti	12
3.4 Radiomoduuli	12
3.5 Signaalin käsittely	12
3.6 Lähetin	13
3.7 Vastaanotin	13
4 VIRHEENTARKISTUS	15
4.1 NMEA datan virheentarkistus	16
4.2 Tarkistussumman kaava	16
4.2.1 Esimerkkilasku tarkistussumman laskemisesta	17
4.3 Tarkistussumman tarkistus	17
4.3.1 Lähetin	18
4.3.2 Vastaanotin	18
5 YHTEYDEN MUODOSTAMINEN PÄÄTELAITTEELLE	19
5.1 Oman tiedon suodattaminen GPS tiedosta	20
5.2 Tiedon lisääminen lähettimessä	20
5.3 Tiedon suodattaminen vastaanottimessa	20
6 JOHTOPÄÄTÖS	21
LÄHTEET	22

LIITTEET

Liite 1. Lähettimen lähdekoodi
Liite 2. Vastaanottimen lähdekoodi

KUVAT

Kuva 1. GPS:n toimintaperiaate9

KUVIOT

Kaavio 1. Laitteen rakenne 10

Kaavio 2. Ohjelman toimintaperiaate 14

1 KÄYTETYT LYHENTEET JA SANASTO

GLONASS	Globalnaya navigatsionnaya sputnikovaya sistema tai Global Navigation Satellite System, Venäjän avaruushallinnon kehittämä koko maapallon kattava satelliittipaikannusjärjestelmä. Kilpaileva järjestelmä GPS:n kanssa.
GPS	Global Positioning System, USA:n puolustushallinnon kehittämä koko maapallon kattava satelliittiperustainen paikannusjärjestelmä
GSM	Groupe Spécial Mobile tai Global System for Mobile Communications, Euroopan telehallintojen yhteistyössä kehittämä matkapuhelinverkkojärjestelmä.
NMEA	National Marine Electronics Association, Yhdysvaltain kansallinen merielektroniikan yhdistys http://www.nmea.org/
NMEA 0183	NMEA:n määrittelemä protokolla, jota käytetään paikannustiedon siirtoon laitteelta toiselle.
TTL	Transistor-transistor Logic. Kehitetty 1960-luvun alussa helpottamaan mikropiirien ja elektroniikan kehittämistä. TTL-mikropiirien tavallinen käyttöjännite on 5 V, mutta voi olla matalampi 3.7 voltia.
TTL-tasoinen signaali	Signaali on ns. TTL-tasoista, kun sen ollessa 0-tilassa jännite on 0–0,8 voltia ja 1 -tilassa vähintään 2,2 voltia, maksimin ollessa ulosantavan piirin käyttöjännite.
I/O portti	Input/Output-portti. Liitäntä piirillä jota voidaan käyttää sekä tiedon vastaanottamiseen, että lähettämiseen.
Rx	Liittimen vastaanottava liitos
Tx	Liittimen lähettävä liitos

2 JOHDANTO

Työn tarkoituksena on tutkia liikkuvan kohteen GPS-paikantamista sekä muun tarvittavan datan siirtoon maastossa, missä ei välttämättä ole GSM-verkkoa käytettävissä. Laitteen lähettävässä päässä on mikrokontrolleri, jonka avulla saadaan lisättyä siirrettävään tietoon GPS-tiedon lisäksi muuta haluttua tietoa ja vastaanottavassa päässä on niin ikään mikrokontrolleri joka suodattaa ja käsittelee tämän tiedon sekä tarkistaa GPS datan virheettömyyden ja välittää sen päätelaitteelle. Vastaanotin tarjoaa päätelaitteille myös kolme erilaista yhteystapaa jotka on Bluetooth, RS232 -sarjaportti sekä USB portti.

Työn pääpaino on signaalin virheettömyyden tarkistamisessa sekä paikannustiedon lisäksi siirrettävän oheisdatan kuljettaminen varsinaisen paikkatiedon ohessa ilman, että häiritään päätelaitteelle välitettävää paikkatietoa. Paikannuslaitteen tarkoituksena on käyttää radiolinkkiä tiedon välitykseen. On erittäin olennaista huomioida myös käytettävä radiotaajuus sekä Viestintäviraston asettamat rajoitukset kyseiselle taajuusalueelle.

Näiden lisäksi oman haasteensa asettaa virrankulutuksen hallinta. Tarkoituksena on käyttää laitteissa olemassa olevia matkapuhelimien akkuja, jonka vuoksi ei tarvitse uhrata rajallisia resursseja akkujen lataamiseen käytettävään logiikkaan tai ohjelmistoon. Tämän vuoksi koko laitteen suunnittelussa asia otetaan huomioon ainoastaan käytettävien komponenttien käyttöjännitteessä.

|

PAIKANTAMISEN HISTORIAA JA PERUSTEITA

Idea satelliittiperustaiseen paikannukseen saatiin, kun Johns Hopkinsin yliopistossa mitattiin Sputnik 1:n lentorataa sen lentäessä Yhdysvaltojen yli lokakuussa 1957. Yhdysvaltain laivaston tarpeita varten suunnitellun Transit navigaationsatelliitin kehittäminen aloitettiin 1950-luvun loppupuolella ja se saatiin valmiiksi 1960-luvun puolivälissä. Yhdysvaltain Ilmavoimat alkoi kehittää omaa 621B -paikannusta ja laivasto jatkoi Transit-järjestelmän kehittämistä Timation-projektina. Hankkeet yhdistettiin 1970-luvun alussa GPS-järjestelmäksi, joka otettiin käyttöön 1990-luvun alussa lukuisten viivästysten myötä. (Poutanen 1998, 19.)

2.1 Kolmiomittaus

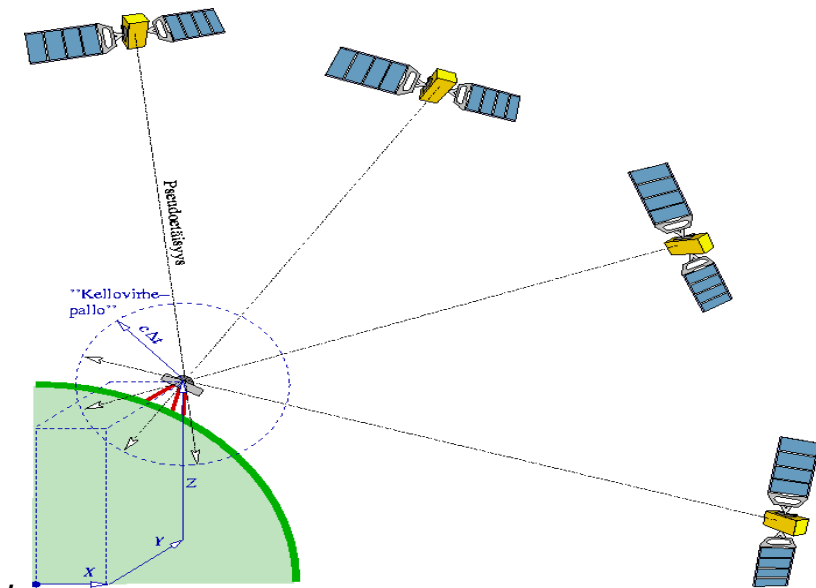
Yksinkertainen ja helpoin tapa paikantaa jokin haluttu piste on kolmiomittaus. Kolmiomittauksessa käytetään kolmea kiintopistettä, joiden sijainti ja etäisyys toisistaan tiedetään tarkasti. Paikannettaessa haluttua pistettä mitataan etäisyydet näihin kolmeen jo tiedettyyn pisteeseen ja siitä voidaan laskea paikannettavan pisteen sijainti. Ihanteellisin tilanne kolmiomittauksessa on, että paikannettava piste on tiedettyjen kiintopisteiden muodostaman kolmion sisällä. Tällöin saavutetaan kolmiomittauksessa tarkin mahdollinen tulos.

Kun kiintopisteitä on käytettävissä enemmän kuin kolme, mittaustuloksen tarkkuus suhteessa kiintopisteisiin säilyy etäisyyden kasvaessa. Tästä on apua GPS-paikannuksessa, kun satelliittien lentoradat ovat noin 20 200 km korkeudella maanpinnasta. (Poutanen 1998, 117.)

2.2 Paikannus GPS:n avulla

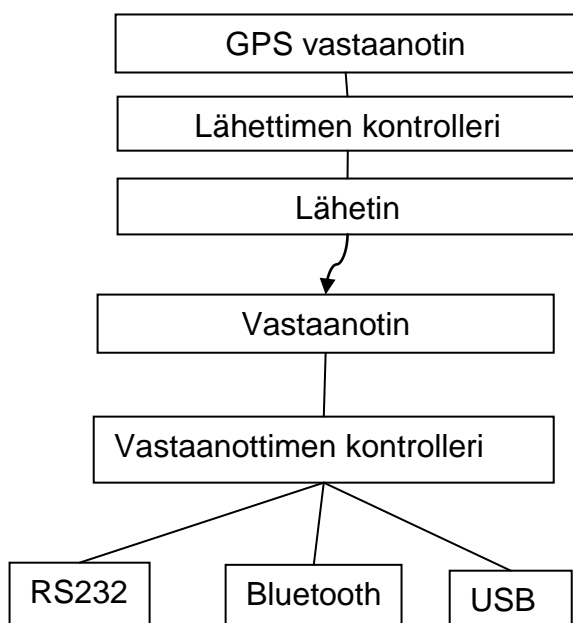
GPS-paikannuksessa tarvitaan kuitenkin vähintään neljän satelliitin näkyminen päätelaitteelle johtuen eri tekijöiden epätarkkuudesta. Näitä tekijöitä on ilmakehän vesihöyryn tuntematon jakautuminen, auringon aktiivisuuden vaikuttaminen ionosfääriin elektronien määrään sekä päätelaitteen marginaalinen kellovirhe.

Paikannus kolmella satelliitilla on mahdollista, mutta tällöin ei voida laskea päätelaitteen korkeutta merenpinnasta. Tästä ei kuitenkaan ole haittaa, mikäli oletetaan päätelaitteen pysyvän aina samalla korkeudella, kuten merialuksissa. (Poutanen 1998)



Kuva 1. GPS:n toimintaperiaate

3 GPS PAIKANTIMEN RAKENNE JA SUUNNITTELU



Kaavio 1. Laitteen rakenne

Laitteen perusideana on olla mahdollisimman kevyt, pienikokoinen ja akunkestoltaan pitkäikäyttöinen. Tämä mahdollistaisi käytön mm. metsästyskoirien seurannassa eikä vastaanottimen koko häiritsisi koiranohjaajaa.

Sekä lähettimessä, että vastaanottimessa on Atmelin ATmega328p-mikrokontrolleri. Kyseinen mikrokontrolleri on todettu parhaiten työhön sopivaksi helpon saatavuutensa ja ohjelmoitavuutensa ansiosta. Lisäksi piirissä on tarpeeksi liitännöitä erilaisiin tarpeisiin. Piirissä on myös sisäänrakennettuna TTL tasoista signaalia käyttävä sarjaportti, johon on helppo liittää piirit Bluetooth, USB ja RS232 -sarjaportteja varten.

Tämän lisäksi mistä tahansa digitaalisesta I/O portista on mahdollista ohjelmallisesti lisätä sarjaportteja mikä on hyvä esimerkiksi Bluetooth -moduulin alustamisessa ilman, että tarvittavat alustuskomennot näkyisi RS232 tai USB portteihin. Joskin tässä työssä käytettyä Bluetooth -moduulia ei tarvitse erikseen alustaa jokaisella käynnistyksellä vaan moduuli tallentaa asetukset omaan pysyväsmuistiinsa. (Atmel 2011)

Varsinaiseen paikannukseen käytettäväksi GPS moduuliksi kelpaa mikä tahansa moduuli, joka antaa TTL tasoisena sarjasignaalina NMEA mukaisen paikkatiedon. Olennaisinta on 3.3 voltin käyttöjännite ja mahdollisimman pieni virrankulutus.

3.1 Bluetooth

Bluetooth on ensisijaisesti kehitetty matkapuhelimien lisälaitteita varten, joten sen kantamaa ei ole alun perin ajateltu mitenkään pitkäksi, vain 10 metriä. Tämä pääasiassa riittää moneenkin laitteeseen, kuten kuulokkeisiin ja tiedonsiirtoon tietokoneelle tai toiseen puhelimeen. (Absolute Astronomy 2011)

Tässä työssä on tarkoitus tarjota paikkatieto näytettäväksi laitteissa, joissa on mahdollista käyttää Bluetooth yhteensopivaa GPS moduulia paikannukseen. Tällöin päätelaite luulee saavansa paikkatiedon lähellä olevasta GPS moduulista, vaikka todellisuudessa se voi olla kilometrien päässä. Tämä on käytännöllistä silloin, kun paikannuslaitetta käytetään koiran paikantamiseen ja koiranohjaaja tietää itse missä on.

Sopivaksi moduuliksi on havaittu mahdollisimman helppokäyttöinen joka ei käynnistysvaiheessa tarvitse erityisempiä toimenpiteitä saadakseen toimivaksi sekä saadaan toimimaan 3,3 voltin jännitteellä.

3.2 USB portti

USB portti suunniteltiin alun perin helpoksi ja nopeaksi oheislaiteliitännäksi tietokoneisiin. Sittemmin sen käyttö on laajentunut muuhun viihde-elektroniikkaan ja se on yleistynyt myös akkukäyttöisten laitteiden latausliittimenä.

FTDI valmistaa piiriä, jolla voidaan liittää TTL-signaalitasoisen sarjaporttiliitännän tarjoava laite tietokoneen USB-porttiin. Myös mikrokontrolleri on mahdollista ohjelmoida tämän portin kautta, kunhan kontrolleriin on ensin saatu ladattua niin sanottu alkulatain, joka käynnistää piirin ja huomaa ohjelmointiyhteyden kontrollerin sarjaportin kautta. (FTDI 2011)

3.3 RS232 sarjaportti

RS232-sarjaportti tunnetaan paremmin COM-porttina, joita tietokoneissa käytettiin ennen USB-porttien kehittämistä mm. modeemien, tulostimien ja hiirien liittämiseen. Myös joissakin vanhemmissa paikannusjärjestelmissä on mahdollista liittää GPS-moduuli sarjaportin kautta. RS232 määritysten mukainen sarjaportti toteutetaan Maximin tarkoitukseen tehdyllä MAX13223E piirillä, joka tarjoaa oikeat signaalitasot sekä mikrokontrollerille, että RS232 -liittimeen käyttöjännitteen ollessa halutut 3,3 voltia. (Maxim IC 2011)

3.4 Radiomoduuili

Koska laitteesta halutaan radioliikenneluvasta vapaa käytännössä käytettävissä olevat taajuudet rajoittuvat 869,400 – 869,650 MHz väliin. Tälle taajuusalueelle on kuitenkin asetettu rajoituksia lähetystehon ja -ajan sekä kaistanleveyden suhteen. Tästä on hyötyä suunnitellessa virransäästöominaisuuksia lähettimeen, sillä kyseisellä taajuusalueella suurin sallittu lähetysteho on 500 mW ja lähetysaika enintään 10 % yhden tunnin jaksossa, mikä käytännössä tarkoittaa 6 minuuttia tunnin aikana. (Viestintävirasto, Määräys 15)

3.5 Signaalinkäsittely

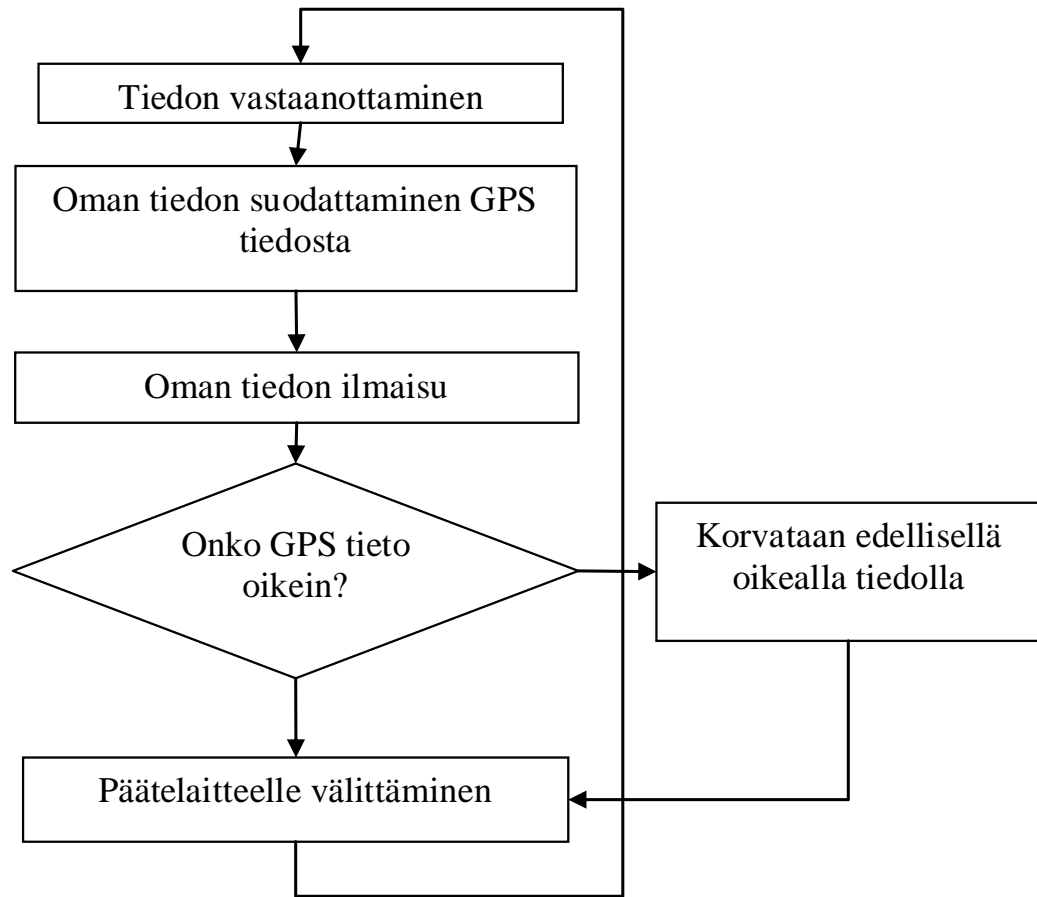
Yksinkertaisimmillaan laitteessa ei tarvitse juuri minkäänlaista signaalinkäsittelyä, koska GPS moduuli antaa suoraan päätelaitteelle kelpaavaa tietoa sellaisenaan. Radiotie on erittäin herkkä erilaisille häiriöille, joten tiedon vastaanottavassa päässä hyvä tehdä tarkistussumman uudelleenlaskenta. Kun pelkän paikkatiedon lisäksi halutaan välittää myös omaa tietoa, joka on tarkoitettu vain välitettäväksi mikrokontroleiden välillä, tarvitaan signaalinkäsittelyä molemmissa päissä.

3.6 Lähetin

Lähtevän yksikön mikrokontrolleri käsittelee lähetettävää dataa siten, että lähtevän GPS tietoon mahdollisesti lisätään omaa dataa, mikäli halutaan kertoa vastaanottajalle jotain muutakin kuin pelkkä paikkatieto. Tällaista tietoa voi olla esimerkiksi haukuntunnistus tai muu kutsu, esim. hätätilanteessa. Tietoa ei lisätä, mikäli sitä ei ole ja tällöin vähennetään radioliikennettä sekä virrankulutusta. Ääntä tai kuvia ei ole tässä versiossa mahdollista lähettää, koska käytetyllä taajuusalueella ei ole sallittua lähettää kerralla niin pitkiä aikoja, että ääni tai kuva saataisiin perille kokonaan sallitussa ajassa.

3.7 Vastaanotin

Vastaanottimen mikrokontrolleri erottelee signaalista GPS paikkatiedon ja mahdollisen oman datan. Lisäksi GPS tiedosta lasketaan tarkistussumma ja verrataan sitä signaalin mukana tulleeseen. Näin varmistutaan siitä, ettei ole tullut siirtovirhettä ja päätelaitteelle voidaan välittää oikeaa paikkatietoa. Mikäli siirtovirhe havaitaan, kasvatetaan virhelaskuria ja välitetään edellinen oikeaksi todettu paikkatieto. Virhelaskuri nollataan jokaisesta oikeaksi todetusta paikkatiedosta. Mikäli laskuri kasvaa liiaksi, välitetään virheellinen tieto päätelaitteelle jolloin käyttäjä saa tiedon siitä, ettei vastaanotettu signaali ole käyttökelpoista. Ohjelman rakenne on koottu kaavioon 2.



Kaavio 2. Ohjelman toimintaperiaate

4 VIRHEENTARKISTUS

\$GPGGA,114258.000,4140.000,N,00053.000,W,0,00,99.0,0145.4,M,51.6,M,,*45

Yllä esimerkki vastaanotetusta signaalista.

Jokainen signaali alkaa \$ merkillä. Seuraavaksi on lähettävän laitteen tyyppi. GP tarkoittaa GPS-laitetta, GL GLONASS-laitetta, joka on venäläisten kehittämä GPS järjestelmän kilpailija. Seuraavat kolme merkkiä on viestin tyyppi. Niitä on määritelty 29 erilaista. Tässä laitteessa käytetään ainoastaan yhtä, GGA:ta. Ensimmäisen pilkun jälkeen oleva numerosarja on kelloaika, jolloin signaali on vastaanotettu. Tämän jälkeen tulee leveysaste ja onko kyseessä maapallon pohjoinen vai eteläinen puolisko. Heti seuraavaksi tulee pituusaste ja sen jälkeen merkintä, onko kyseessä läntinen vai itäinen pituus. Seuraava numero kertoo datan luotettavuuden, näitä on 9 erilaista:

- 0 Ei lukitusta
- 1 GPS-lukitus
- 2 DGPS-lukitus
- 3 PPS-lukitus
- 4 Reaaliaikainen kinematiikka
- 5 Kelluva reaaliaikainen kinematiikka
- 6 Arvioitu aika
- 7 Manuaalinen tila
- 8 Simulointitila

|

Sitten on näkyvien satelliittien määrä, horisontin aiheuttama vaimennus, korkeus meren pinnasta, meren pinnan korkeus WGS84 ellipsoidin yläpuolella. Molempien korkeusarvojen jälkeen oleva M -kirjain tarkoittaa lukeman olevan metreissä. Kaksi seuraavaa on aika edellisestä DGPS päivityksestä ja DGPS tunnusnumero. Tähti ja kaksi numeroa on signaalin tarkistussumma. Tähti pysyy aina samanlaisena ja tarkistussumma lasketaan XOR –kaavalla dollari –merkin ja tähden välistä.

4.1 NMEA datan virheentarkistus

GPS tieto lähetetään NMEA -muotoisena joka sisältää paikkatiedon ASCII merkkeinä. Jokaisen tilatiedon lopussa on kaksi merkkiä varattu virheentarkistukselle. Tämä tarkistussummaksi sanottu lukupari lasketaan yksinkertaisella XOR -operaatiolla varsinaisesta datasta. Tämä laskutapa on tarpeeksi luotettava kertomaan, että onko data oikeellista, mutta silti riittävän yksinkertainen ettei vie liikaa laskentatehoa laitteelta ja näin ollen on nopeasti laskettavissa. (NMEA, viitattu 21. 11. 2011)

4.2 Tarkistussumman kaava

Laskettavasta paikkatiedosta jätetään ensimmäisenä oleva \$ merkki ja * merkin sekä *:n jälkeen olevat merkit huomiotta. Esimerkissä xorBuffer -muuttuja sisältää vain laskennassa tarvittavan datan.

```
CalcCRC = 0; // Asetetaan lähtöarvoksi nolla.  
  
s = strlen(xorBuffer); // Otetaan merkkijonon pituus  
  
for(i = 0; i < s; i++) { // Silmukka pyörii niin kauan kuin merkkejä on.  
  
CalcCRC ^= (unsigned char)xorBuffer[i]; // Lasketaan XOR.  
  
}
```


4.2.1 Esimerkkilasku tarkistussumman laskemisesta

Laskenta tapahtuu XOR operaatiolla, joka tarkoittaa exclusive or, eli poissulkeva tai. Jos molemmat tarkasteltavat bitit ovat joko 0 tai 1, niin summasta tulee 0, jos vain toinen biteistä on 1, niin summasta tulee 1. Jokainen merkki muodostuu GPS signaalin tapauksessa 8 bitistä. Tarkistussumman lähtöarvon ollessa nolla sivuutetaan ensimmäinen laskutoimitus, joka asettaa summan samaksi kuin kirjain G. Seuraavan merkki, P, muuttaa jo summaa joten käytetään sitä mallina.

$$\begin{array}{r} G = 01000111 \\ P = 01010000 \\ \hline \text{Summa} \quad 00010111 \end{array}$$

Tätä laskentatapaa käytetään kunnes koko signaalisanoma on käyty läpi ja sitä verrataan vastaanotettuun tarkistussummaan. Mikäli summat ovat samat, laskennan suorittava funktio palauttaa arvon tosi ja näin pääohjelma voi välittää vastaanotetun signaalin päätelaitteelle. Jos summat eivät täsmää kasvattaa pääohjelma virhelaskuria ja välittää edellisen vastaanotetun oikeaksi todetun signaalin päätelaitteelle. Kolmen virheen jälkeen aletaan lähettämään tätä virheellistä signaalia päätelaitteelle, jotta päätelaite voi ilmoittaa käyttäjälle virheellisestä signaalista. Virhelaskurin tarkoituksena on poistaa yksittäiset siirtovirheet jotka eivät kuitenkaan vaikuta paikkatietoon.

4.3 Tarkistussumman tarkistus

Tarkistussumma on syytä tarkistaa aina tietoa vastaanottaessa, että mahdolliset siirtovirheet voida havaita. On myös mahdollista, että oman tiedon lisääminen lähettimessä saattaa sotkea GPS-tietoa ilman, että tämä huomataan vasta tarkistussumman laskentavaiheessa. Siksi on tärkeää, ettei tällaista tietoa

välitetä päätelaitteelle, koska ei voida tietää miten käytössä oleva päätelaite reagoi täysin virheelliseen GPS tietoon.

4.3.1 Lähetin

Lähetin ei ota kantaa tarkistussumman oikeellisuuteen. Tämä pääasiassa virransäästösyistä, mutta myös voidaan olettaa, että suoraan kontrollerissa kiinni oleva GPS moduuli antaa oikeellista dataa.

4.3.2 Vastaanotin

Vastaanottimen mikrokontrolleri laskee paikkatiedon tarkistussumman uudelleen ja vertaa sitä vastaanotetun signaalin mukana saatuun summaan. Tämä tarkistus tehdään aliohjelmassa ja mikäli summat täsmää aliohjelma palauttaa arvon tosi, jolloin data voidaan välittää päätelaitteelle. Kuitenkin ennen päätelaitteelle välittämistä pitää vastaanotetusta datasta suodattaa ylimääräinen lähetävän osan mikrokontrollerin lisäämä tieto pois ja käsitellä asiaan kuuluvalla tavalla.

|

5 YHTEYDEN MUODOSTAMINEN PÄÄTELAITTEELLE

NMEA 0183 standardissa määritellään yhteyden nopeus, pariteetti ja stop – bittien määrä, ne asetetaan kaikille yhteystyypeille samoiksi. Tämä antaa myös lisäaikaa oman tiedon ja virheiden käsittelylle.

RS232 sarjaportti ei vaadi erityisempiä alustuskomentoja toimiakseen päätelaitteessa. Käytännössä riittää, että yhdistetään Rx, Tx ja maajohdin.

USB portti sen sijaan vaatii päätelaitteelta tuen FTDI:n valmistamalle FT8U232AM tai FT232R piirille. Ensin mainittu on jo tuotannosta poistunut USB 1.1 määrittymisen täyttävä piiri ja näin ollen oletettavasti hyvin tuettu. FT232R on tuotannossa oleva laajennettu USB 2.0 määrittymisen mukainen piiri jota vanhemmat laitteet eivät välttämättä tue kunnolla. Koska molemmat piirit tarjoaa kuitenkin määrittymisen mukaiset rajapinnat USB sarjaportille, ongelmaa ei pitäisi olla. Koska FT232R on seuraaja FT8U232AM piirille, sisältää se vähintään samat ominaisuudet kuin FT8U232AM. (FTDI 2011a ja 20011b)

Bluetooth yhteyttä käytettäessä ei varsinaisesti tarvitse erikseen alustaa jokaisella käynnistyksellä, mutta ennen tuotantoon ottoa on hyvä vaihtaa jokaisen bluetooth piirin oletusnimi tarkoituksellisempään ja PIN koodi tämän tyyppisissä laitteissa oletuskäytäntönä olevaan 0000. Valitettavasti käytössä olevalle piirille ei löydy mallinumeron perusteella suoraan datalehteä. (Trainelectornics 2011)

5.1 Oman tiedon suodattaminen GPS tiedosta

Oman tiedon lähettäminen GPS tiedon seassa on sinänsä aika helppoa, on sen suodattamiseen pois ennen GPS tiedon välittämistä päätelaitteelle painotettava jonkin verran enemmän huomiota. Tätä helpottaa, kun valitaan oman tiedon alkumerkiksi jokin sellainen ASCII merkki, jota ei esiinny missään vaiheessa NMEA 0183 määritysten mukaisessa GPS tiedossa. Tällainen merkki on esimerkiksi # merkki. Toinen lopetusmerkiksi sopiva merkki olisi @. Näistä merkeistä voidaan helposti päätellä, että välissä oleva tieto ei ole NMEA määritysten mukaisia ja niitä ei pidä välittää päätelaitteelle. Haluttaessa vain ilmaista jotain yksinkertaista voidaan lähettää vain aloitus- ja lopetusmerkki, joten ei tuhjata kallisarvoista ilmatietä.

5.2 Tiedon lisääminen lähettimessä

Lähettimessä käytetään keskeytystä lisäämään tieto lähetettävään signaaliin. Näin siksi, ettei erikseen tarvitse tutkia mitään erityistä pinniä dataa varten tai mahdollisesti menettää joko GPS tietoa tai käyttäjän omaa lähetettäväksi tarkoitettavaa tietoa.

5.3 Tiedon suodattaminen vastaanottimessa

Vastaanottaessa ei tarvitse käyttää erillisiä keskeytyksiä datan suodattamiseen, koska kaikki vastaanotettava tieto tulee samasta lähteestä ja vastaanottamiseen tarkoitettun ohjelmallisen sarjaportin luomiseen käytettävä kirjasto tukee puskurointia.

6 JOHTOPÄÄTÖS

Laitetta suunniteltaessa tuli huomattavia ongelmia löytää valmiita komponenttikokonaisuuksia etenkin radiomoduulin valinnassa. Valmiita Viestintäviraston radioluvasta vapautettuja laitteita ei löytynyt kuin yhdeltä valmistajalta ja siltäkään ei saanut laitteita kokeiltavaksi. Tämän takia varsinainen kokonaisuuden kenttätestaus on käytännössä jäänyt tekemättä. Kenttätestauksen sijaan keskityttiin virheidenkorjaukseen sekä muun tiedon välittämiseen paikannustiedon lisäksi.

Mahdollisuus välittää paikkatiedon lisäksi muutakin tietoa oli yksi tärkeimmistä kriteereistä tämän laitteen suunnittelemiselle. Toinen tärkeä kriteeri oli, että laite olisi maantieteellisesti katsottuna mahdollisimman paikkariippumaton ja sitä kautta helposti käytettävissä koko Suomessa ilman, että loppukäyttäjän tarvitsisi miettiä toimintaedellytyksiä esimerkiksi matkapuhelinverkon kuuluvuuden suhteen.

LÄHTEET

Markku Poutanen, GPS-paikanmääritys, Ursa, 1998

Viestintävirasto, Määräys 15, 20.10.2011

NMEA, NMEA 0183 Standard, viitattu 21.11.2011,
http://www.nmea.org/content/nmea_standards/nmea_083_v_400.asp

MAXIM IC, +5 Powered, Multichannel RS-232 Drivers, viitattu 21.11.2011,
<http://www.maxim-ic.com/datasheet/index.mvp/id/1798>

Trainelectronics, Bluetooth to Serial modules, viitattu 21.11.2011,
<http://trainelectronics.com/Bluetooth/>

FTDI 2011a, FT8U232AM, viitattu 21.11.2011,
<http://www.ftdichip.com/Products/ICs/FT8U232AM.htm>

FTDI 2011b, FT232E, viitattu 21.11.2011, <http://www.ftdichip.com/Products/ICs/FT232R.htm>

Atmel, Atmega328P Parameters, viitattu 21.11.2011, <http://bit.ly/vUIKvj>

Absolute Astronomy, Bluetooth, viitattu 21.11.2011,
<http://www.absoluteastronomy.com/topics/Bluetooth#encyclopedia>

GPS lähettimen lähdekoodi

```
#include <SoftwareSerial.h>

/*
 *
 *
 * GPS lähettimen koodi
 *
 *
 */

// Vakiot
#define rxPin 12
#define txPin 9
volatile int notsend = LOW;

// set up the serial port

SoftwareSerial gpsSerial = SoftwareSerial(rxPin, txPin);

// variables
byte byteGPS = 0;
int i = 0;
int indices[13];
int cont = 0;
int conta = 0;

char inBuffer[300] = "";
```

```
int k = 0;

void setup(){
  Serial.begin(4800);

  Serial.print("Alustetaan GPS"); //Jos käytetään tuotannossa, niin varmasti
  sekoittaa softat

  //setup for gpsSerial port
  pinMode(rxPin, INPUT);
  Serial.print(".");
  pinMode(txPin, OUTPUT);
  Serial.print(".");
  gpsSerial.begin(4800);
  Serial.print(".");
  //setup for Serial port

  // alustetaan GPS moduuli

  delay(1000);
  Serial.print(".");
  gpsSerial.println("$PSTMNMEACONFIG,0,4800,1,1"); // Vain GGA
  Serial.print(".");
  delay(100);
  Serial.print(".");
  gpsSerial.println("$PSTMINITGPS,4140.000,N,00053.000,W,0197,22,10,2007,
  11,40,00");
  Serial.print(".");
```



```
attachInterrupt(1,blink,CHANGE);  
  
Serial.println("done"); //Jos käytetään tuotannossa, niin varmasti sekoittaa  
softat  
  
}
```

```
void blink(){  
  if (notsend) {  
    Serial.print("#@"); //Lähetetään merkit, että omaa tietoa löytyy.  
    notsend = !notsend;  
  } else {  
    notsend = !notsend;  
  }  
}
```

```
void loop(){  
  
  byteGPS = 0;  
  i = 0;  
  while(byteGPS != 42){  
    byteGPS = gpsSerial.read();  
    inBuffer[i]=byteGPS;  
    i++;  
    Serial.print(byteGPS);  
  }  
}
```

Vastaanottimen lähdekoodi

```
#include <NewSoftSerial.h>

/*
 *
 *
 * GPS vastaanottimen koodi
 *
 *
 */

// Vakiot
#define rxPin 11
#define txPin 9

// sarjaportin luominen

NewSoftSerial mySerial(rxPin, txPin);

// muuttujat
byte byteGPS = 0;
int i = 0;
char inBuffer[80];
boolean xorcrc (char *buffer){
char rawBuffer[80];
char * CRC;
char * strpBuffer;
```

```

char * xorBuffer;

int i;

int iCRC;

unsigned long s;

unsigned char CalcCRC;

boolean checkok = false; // Asetetaan tarkistustulos oletuksena epätodeksi,
jotta ei tarvi tuhlasta koodia turhiin vaihdoksiin.

strcpy(rawBuffer,buffer); // Kopioidaan osoitteen data muuttujaan, ettei muuteta
alkuperäistä.

xorBuffer = strtok(rawBuffer,"*"); // Napataan datasta talteen kaikki ennen *
merkkiä. * itsestään on tarpeeton

CRC = strtok(NULL,"*"); // Sitten otetaan talteen saatu tarkiste, * on edelleen
tarpeeton.

iCRC = atoi(CRC);

if (strcspn(rawBuffer,"$") == 0) { // Jos muuttujasta löydetty $ on ensimmäinen
merkki, niin tarkistettava rivi ainakin alkaa oikein.

    CalcCRC = 0;

    s = strlen(xorBuffer); // Otetaan rivin pituus.

    for(i = 1; i < s; i++) { // Kierretään ympyrää kunnes kaikki merkit on käyty läpi.

        CalcCRC ^= (unsigned char)xorBuffer[i]; // Lasketaan simppele XOR tarkistus

    }

    if (CalcCRC == iCRC,DEC) // Verrataan laskettua ja saatua. Laskettu on
desimaali ja saatu hekso, joten muutos tarvitaan.

        checkok = true; // paluuarvo on oletuksena epätosi, joten sitä ei tarvitse
erikseen laittaa jos tarkistus ei täsmää.

    }

return checkok;

}

```

```
void setup(){

    //Asetetaan sarjaportit
    pinMode(rxPin, INPUT);
    pinMode(txPin, OUTPUT);
    mySerial.begin(4800); // Digitaaliportti, josta luetaan tuleva GPS.
    Serial.begin(4800); // Piirin varsinainen portti, johon lähetetään tarkistettu
    GPS.
}

void loop(){

    int crcError = 0;
    byteGPS = 0;
    while (mySerial.available()) {
        byteGPS = mySerial.read();
        if (byteGPS == '\n') ;
        else if (byteGPS == '@') {
        }
        else if (byteGPS == '\r') {
            inBuffer[i] = 0;
            if (xorcrc(inBuffer)) {
                crcError = 0;
                Serial.println(inBuffer);
            }
        }
        else
            crcError++;
    }
}
```

```
if (crcError > 3 )  
    Serial.println(inBuffer);  
    i = 0;  
    break;  
}  
else inBuffer[i++] = byteGPS;
```

```
}  
}
```