

Mikko-Petteri Pikkarainen

# Raspberry Pi -pohjainen RGB- & IR- kuvantamis- ja mittausjärjestelmä

Insinööri (AMK),  
Tietotekniikka

Kevät 2015



KAJAANIN  
AMMATTIKORKEAKOULU  
UNIVERSITY OF APPLIED SCIENCES

## TIIVISTELMÄ

**Tekijä(t):** Mikko-Petteri Pikkarainen

**Työn nimi:** Raspberry Pi -pohjainen RGB- & IR- kuvantamis- ja mittausjärjestelmä

**Tutkintonimike:** Insinööri (AMK), Tietotekniikka

**Asiasanat:** Raspberry Pi, Python, IR & RGB -kamera, GPS, digitaalinen anturi, säikeistys

Insinööritöiden tavoitteena oli suunnitella ja toteuttaa Raspberry Pi -pohjainen kuvantamis- ja mittausjärjestelmä. Työn tilaajana oli Oulun yliopiston mittaustekniikan yksikkö CEMIS-Oulu. Kuvantamis- ja mittausjärjestelmä testattiin, suunniteltiin ja valmistettiin CEMIS-Oulun tiloissa Kajaanis- sa.

Insinööritöissä käytetään luottokortin kokoista Raspberry Pi tietokoneen RPI B+ versiota järjestelmän pohjana. Työssä RPI:hin liitetään kaksi kappaletta Raspberry Pi -kameran: normaali RGB- ja NOIR-kamera. Kameroiden liittäminen ja ohjaus tapahtuu erillisellä lisäkortilla, koska RPI:ssä on suunniteltu liitettäväksi vain yksi kamera. Lisäksi kuvantamis- ja mittausjärjestelmään asennettiin sarjaliikennettä käyttävä GPS-vastaanotin ja I<sup>2</sup>C-rajapintaa käyttävä digitaalinen mittausanturi. Kuvantamis- ja mittausjärjestelmän käyttökohde on Phantom 2, kauko-ohjattava UAV-helikopteri.

Insinööritöitä käsitellään kahdessa osassa: suunnittelu ja testaus. Suunnittelu tapahtui Raspberry Pi:n teknisten ja järjestelmän vaatimusten pohjalta. Ohjelmointi tapahtui Python-ohjelmointikielellä, ja järjestelmän ohjelmakoodi suunniteltiin säikeistettäväksi. Säikeistetty ohjelma mahdollisti järjestelmään liitettävien kameroiden, GPS-vastaanottimen ja digitaalisen anturin yhtäaikaisten toimien. Lisäksi uusien antureiden liittäminen säikeistettyyn ohjelmakoodiin on varsin helppoa. Järjestelmän testaus tapahtui porrastetusti. Jokainen liitettävä komponentti ja anturi testattiin toimivaksi ennen kokonaisuuden asennusta ja testausta. Toimintoja testattiin insinööritöiden aikana laboratorio-oloissa.

Työn tuloksena oli Raspberry Pi:llä toimiva kuvantamis- ja mittausjärjestelmä, jonka toiminta on varmistettu testeissä. Järjestelmä pystyi toimimaan kahdella kameralla, lukemaan GPS-paikkatiedon arvoja ja synkronoimaan paikkatiedon otettuun kuvaan. Lisäksi järjestelmä pystyi lähettämään kuvia ja paikkatiedon langattomasti toiselle valvontaohjelmalle. Kaiken tämän lisäksi digitaalisen anturin luku ja mittaukset pystyttiin tallentamaan järjestelmässä.

## ABSTRACT

**Author(s):** Mikko-Petteri Pikkarainen

**Title of the Publication:** Raspberry Pi- Based RGB and IR Imaging and Measurement System

**Degree Title:** Bachelor of Engineering, Information Technology

**Keywords:** Raspberry Pi, Python, IR & RGB –camera, GPS, Digital sensor, multithreading

The aim of this Bachelor's thesis was to design and create a Raspberry Pi-based imaging and measurement system. The commissioner of the thesis was Oulu University's unit CEMIS-Oulu that specializes in measurement technology. Imaging and measurement system was tested, designed and produced in CEMIS-Oulu's facilities in Kajaani

A Raspberry Pi B+ model credit card-sized computer system was the basis in this thesis. Two pieces of Raspberry Pi cameras are connected to Raspberry Pi. Cameras are normal RGB and NOIR cameras. The camera connection and control is a separate additional card, because the RPI is designed to connect only one camera module at the time. In addition, the imaging and measurement system will be installed in a serial communication using GPS receiver and an I<sup>2</sup>C interface using digital measurement sensor. The target use of imaging and measurement is the Phantom 2 remote-controlled UAV helicopter.

There are two parts in this thesis; design and testing. Raspberry Pi was designed with technical and system technical requirements. Programming took place in the Python programming language environment and the system's program frame was designed to multithreading. Multithreaded program enabled the system to connect cameras, a GPS receiver and a digital sensor simultaneously. In addition, it is quite easy to connect new sensors to the multithread program. The system was tested in cascades. Each connected component and sensor was tested before the whole assembly and testing took place. The software and hardware were tested in laboratory work conditions.

The result was the Raspberry Pi- based imaging and measurement system whose operation has been verified in tests. The system was able to work with two cameras, read the GPS location data values and synchronize GIS data at captured image. In addition, the system was able to send images and GIS data wirelessly to another control program. Added to all this, it was possible to save the digital sensor readings and the measurements to the system.

## SISÄLLYS

1 JOHDANTO.....	1
2 RASPERRY PI -KEHITYSALUSTA.....	2
2.1 Laitteisto/tekniikka .....	2
2.1.1 GPIO laitteen ohjauksessa.....	3
2.1.2 Raspberry Pi -kamerat .....	5
2.2 Ohjelmisto .....	6
3 ILMASSA LIIKKUVAT MITTALAITEALUSTAT .....	7
4 KUVANTAMIS- & MITTAUSJÄRJESTELMÄN SUUNNITTELU.....	9
4.1 IR- ja RGB-kamera.....	9
4.2 Raspberry Pi -kameran multiplekseri.....	10
4.3 GPS-signaali .....	11
4.4 Digitaaliset anturit.....	12
4.5 Tiedonsiirto UDP .....	13
4.6 Pythonilla toteutettava ohjelma.....	14
5 KUVANTAMIS- & MITTAUSJÄRJESTELMÄN TESTAUS.....	16
5.1 GPS-datan kirjoittaminen ja kuvanotto .....	16
5.2 IVPort Raspberry Pi -kameramultiplekseri.....	18
5.3 GPS & kuvantamisen sekä tiedonsiirron säikeistys.....	20
5.3.1 LabVIEW-valvontaohjelma .....	21
5.3.2 Kuvantamisen ja mittauksen pääohjelma .....	22
5.4 Digitaalisen anturin kytkentä.....	24
6 RPI KUVANTAMIS- & MITTAUSJÄRJESTELMÄ.....	27
7 YHTEENVETO .....	30
LÄHTEET .....	31
LIITTEET	

## TERMILUETTELO

AD	Analog to Digital
CSI	Camera Serial Interface
EEPROM	Electrically Erasable Programmable Read-Only Memory
GPIO	General Purpose Input / Output
GPS	Global Position System
GYRO	Gyroskooppi
ID	Identifier
IR	Infra-Red
JPEG	Joint Photographic Experts Group
NMEA	National Marine Electronics Association
NOIR	No Infra-red
RGB	Red Green Blue
RPI	Raspberry Pi
TCP	Transmission Control Protocol
UAV	Unmanned Aerial Vehicle
UDP	User Datagram Protocol

## 1 JOHDANTO

Tässä työssä toteutetaan kuvantamis- ja mittausjärjestelmä. Järjestelmän ominaisuudet ovat ympäristön kuvaaminen RGB- ja IR -kameroilla. Kuvausjärjestelmä myös lukee paikkatiedon Raspberry Pi:hin liitetyltä GPS-vastaanottimelta. Lisäksi järjestelmä tekee mittauksia siihen mahdollisesti kytkettävillä analogisilla tai digitaalisilla antureilla. Tässä insinööriyössä mittausjärjestelmään kytketään yksi digitaalinen anturi.

Kuvantamis- ja mittausjärjestelmä on tarkoitus asentaa myöhemmin kauko-ohjattavaan Phantom 2 UAV -helikopteriin. Vaikka tämän insinööriyön ensisijainen tavoite on järjestelmän liittäminen Phantom 2 UAV-helikopteriin, niin tarkoituksena on myös tehdä suunnitellusta mittajärjestelmästä modulaarinen ja helposti liikutettava sulautettu laitejärjestelmä. Järjestelmää pystytään tämän ansiosta käyttämään mahdollisesti muissa UAV-laitteissa tai sovelluksissa. Tähän tähtää myös järjestelmän langattomalla yhteydellä suunniteltu ja toteutettu valvontaohjelma, joka voidaan asentaa erilliseen tietokoneeseen. Tämän sovelluksen ansiosta järjestelmä voi toimia tietynlaisissa valvovissa tehtävissäkin, mutta se ei ole tavoitteena tässä työssä.

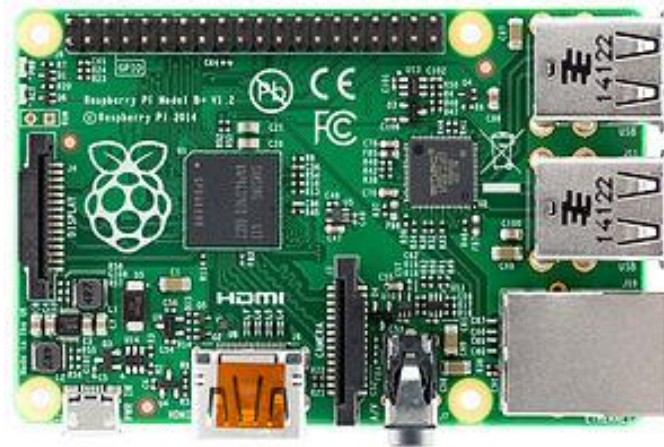
Tämän insinööriyön tilaaja on Oulun yliopiston mittaustekniikan yksikkö CEMIS-Oulu, joka sijaitsee Kajaanissa Petäisenniskassa. Insinööriyö liittyy yksikössä käynnissä olevaan liikkuvat laitealustat (LILA) –projektiin. Projektin tavoitteena on tutkia ja soveltaa liikkuvia laitealustoja uudentyyppisiin mittauksiin sekä kehittää siihen liittyviä mittausratkaisuja. Luotava ja kehitettävä kuvantamis- ja mittausjärjestelmä on soveltuva erityisesti ilmassa liikkuvien laitealustoille.

## 2 RASPBERRY PI -KEHITYSALUSTA

Raspberry Pi on luottokortin kokoluokkaa oleva pieni tietokone, jonka käyttöön tarvitaan lisäksi vain näyttö ja näppäimistö; hiirikin on vain vaihtoehtoinen. Tietokoneen kehittäjänä toimii brittiläinen Raspberry Pi Foundation. Raspberry Pi Foundationin päämäärä on kehittää koulutusta aikuisten ja lasten keskuudessa tietotekniikan osa-alueilla. Tähän päämäärään tähtäävät myös Raspberry Pi -tietokoneet. Ne luovat yksinkertaisen oppimisympäristön tietokoneohjelmointikielten (C-kielen, Pythonin ym.) opiskeluun, ohjelmointiin sekä teknisten projektien toteuttamiseen (esim. sääasema, infrapunakamerakuvaaminen, autotallin oven kauko-ohjaus). [1.] [2.]

### 2.1 Laitteisto/tekniikka

Raspberry Pi -tietokoneet perustuvat Broadcomin BCM2835-järjestelmäpiireihin. Raspberrystä on julkaistu neljä erilaista versiota: Model A, -A+, -B ja -B+. Raspberry Pi Model A sisältää vain tarvittavan 256 MB RAM ja yhden USB-portin. Model A on varsin pieni ja vähemmän tehoa tarvitseva piiri verrattuna toisiin malleihin, minkä vuoksi se sopii varsin yksinkertaisiin ja vähän tehoa vaativiin tehtäviin robotiikassa. Model B on Raspberry-tietokoneiden peruskivi. Se sisältää jo 512 MB RAM, kaksi USB-porttia ja 100 MB Ethernet-portin. Raspberry Pi Model B:stä on julkaistu laajalti sovelluksia ja projektitöitä aina verkkoserverin tehtävistä robottikäsien ohjaukseen. Heinäkuussa 2014 julkaistu Raspberry Pi Model B+ (kuva 1) on parannettu versio Model B:stä, mutta samaan hintaan. Model B+ -versioon on päivitetty 4 kpl USB-portteja ja muista malleista poiketen SD-muistikorttipaikalle on asennettu pienempi micro SD –muistikorttipaikka. [ 3, s. 6, 9 – 11.] [ 4, s. 5 – 8.] [5.] [6.] [7.] [8.]

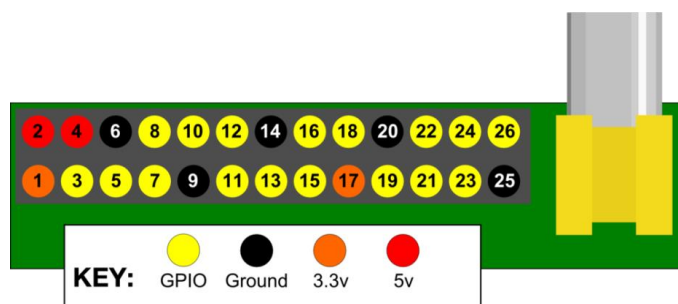


Kuva 1. Raspberry Pi Model B+, julkaisiin 14. Heinäkuuta 2014 [1.]

Uusin Raspberry Pi -versio on Model A+, joka julkistettiin marraskuussa 2014. B+:n tapaan siinäkin on 40 GPIO-pinniä ja micro SD -muistikorttipaikka. A+ on pienempi kuin toiset Raspberry Pi -versiot. Se on tarkoitettu projekteihin, joissa tarkoituksena on tuottaa energiatehokas laite. [ 3, s. 6, 9 – 11.] [ 4, s. 5 – 8.] [5.] [6.] [7.] [8.]

### 2.1.1 GPIO laitteen ohjauksessa

Luottokortin kokoisesta Raspberry Pi -tietokoneesta tekee erinomaisen sen kyky liittää itseensä erilaisia ohjelmoitavia oheislaitteita GPIO-pinnien avulla. Raspberry Pi A- ja B -malleissa pinnien määrä on 26. Näistä pinneistä 17 toimittaa tehtävänsä GPIO-pinneinä. Toiset ovat joko virtalähde- (+3,3 V ja +5,0 V) tai maapinnejä (kuva 2). [9.]

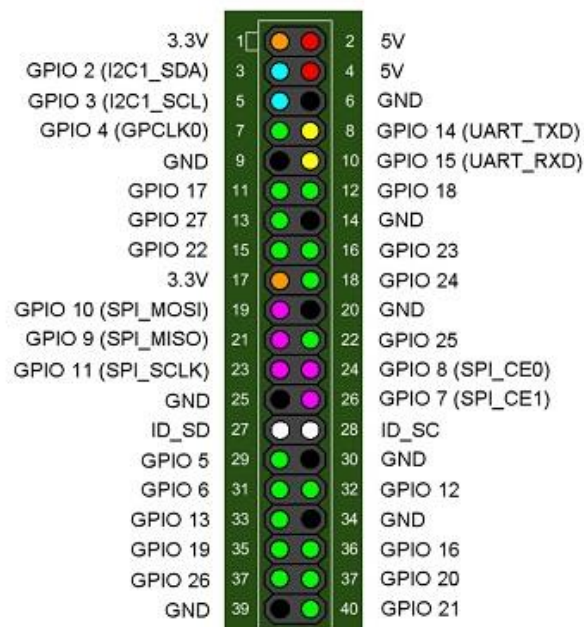


Kuva 2. Raspberry Pi -mallien A ja B GPIO-pinnit.[9]



GPIO-pinnit mahdollistavat kytkimien ja ledien käytön Raspberry Pi -ohjelmoitujen ohjelmien hallinnassa ja toimintojen ilmaisemisessa. GPIO-pinnit mahdollistavat myös I<sup>2</sup>C-, SPI- ja sarjaliikenneväylän käytön Raspberry Pi:hin liitettävissä olevissa laitteissa, kuten esim. GPS-vastaanottimessa tai AD-muuntimessa. [9.] [1.]

Malleista A ja B poiketen uudemmat mallit A+ ja B+ on varustettu 40 GPIO-pinnillä (kuva 3). Ensimmäisten 26 pinnin järjestys ei ole muuttunut +-malliin siirryessä. 16 lisäpinnistä 11 GPIO-pinniä on tarkoitettu ledien ja kytkinten ohjaukseen, ja 3 pinniä toimivat mahdollisten kytkentöjen maadoittajina, mutta erityisesti kahdessa pinnissä, pinnit 27 ja 28, on uusi Raspberry Pi:n ominaisuus; HAT (Hardware Attacher on Top). Pinnien ominaisuutena on ID EEPROM (Electrically Erasable Programmable Read-Only Memory), jolla tarkoitetaan haihtumattomia puolijohdemuistia. Tämän ansiosta Raspberry Pi B+ -malli voi muistaa siihen liitetyt lisälaitteet. Lisäksi laitteet, jotka tukevat ID EEPROM:a, voivat kertoa Raspberry Pi:lle, mitkä pinnit tulee asettaa käyttöön, mikä laite on kytketty pinnihin ja sallii myös automaattisesti Raspberry Pi -alustan ladata käynnistyksen kannalta välttämättömät laiteajurit ja tarvittavat käyttöohjelmat. [10.] [11.] [12.]



Kuva 3. Raspberry Pi -mallien A+ ja B+ GPIO-pinnit.[15.]

### 2.1.2 Raspberry Pi -kamerat

Raspberry Pi:hin on mahdollista liittää myös kamera. Mahdollisia suoraliitännäisiä kameroita on kahta mallia (kuva 4). Ensimmäisellä Raspberry Pi:n kameramoduulilla voidaan kuvata Full HD 1080p -videota tai ottaa valokuva ohjelmoitavalla ohjauksella. Kameramoduuli mahdollistaa monenlaisen kuvaamisen aina time-lapse ja slow-motion-videoista lähtien. Kameran kiinnitys Raspberry Pi:hin on yksinkertaista lattakaapelilla moxeliittimen avulla CSI-porttiin. [1.] [13.] [14.]



Kuva 4. Raspberry Pi -kameramoduulit. Oikealla NOIR-kameramoduuli ja vasemmalla perinteinen kameramoduuli. [13.] [14.]

Toinen Raspberry Pi -kameramoduulivaihtoehtoista on NOIR-kameramoduuli. Se eroaa tavallisesta kameramoduulista mahdollisuudella kuvata infrapuna-aallonpituuksilla. NOIR (No Infra-red) -kamerassa ei ole infrapunasuodatinta. Tämä mahdollistaa kameran käytön pimeänäkökamerana infrapunavalonheittimellä. Erilaisilla kalvoilla infrapuna-aallonpituuksia suodattamalla voidaan NOIR-kameramoduulista muokata lähi-infrapunaspektroskooppi, jolla voidaan kuvantaa lehtivihreää kasvillisuudesta. NOIR-kameran mukana tulee matkassa sininen kalvo, joka mahdollistaa lehtivihreän kuvantamisen. [13.] [14.]

## 2.2 Ohjelmisto

Raspberry Pi -tietokoneet käyttävät ARM-pohjaista suoritinarkkitehtuuria. Useat 2000-luvun elektroniset kannettavat laitteet käyttävät samaa suoritinarkkitehtuuria sen tehokkuuden takia. Tämän takia Raspberry Pi -ohjelmiston rakenne perustuu Linux Kernel -pohjaisiin käyttöjärjestelmiin. 32-bittisen mikroprosessoriarkkitehtuurin takia tämän päivän käyttöjärjestelmäversiot tietokoneista eivät tue Raspberry Pi:tä. Siksi Raspberry Pi:lle on tarjolla monenlaisia Linux-pohjaisia käyttöjärjestelmiä. Käyttöjärjestelmät eroavat pieniltä yksityiskohdiltaan ja tarkoituserialta keskenään, mutta perusohjelmisto on samanlainen kussakin. Viisi yleisintä käyttöjärjestelmää ovat Raspbian, Pidora, Openelec, Raspbmc ja Risc OS. Käyttöjärjestelmien mukana tulevat monipuoliset ohjelmistot esim. medialaitteiden, palvelimien ja ohjelmien ohjelmointiin. [ 3, s. 7 – 8.] [16.] [1.]

### 3 ILMASSA LIIKKUVAT MITTALAITEALUSTAT

UAV (Unmanned Aerial Vehicle) on automaattisesti tai kauko-ohjatusti lentävä miehittämätön laite. Yleisimmin UAV on muutamien kilogrammojen tai kymmeni- en kilogrammojen painoinen, yhden tai kahden ihmisen kannettavissa oleva laite, joka liikkuu sähkö- tai polttomoottorikäyttöisen voiman avulla. UAV-laitteita on kahta päätyyppiä. Lentokonetyyppinen UAV on nopea etenemään, mutta ei kykene pystysuoraan liikkeellelähtoon, laskeutumiseen tai leijuntaan. Tämän tyyppisen UAV:n laukaisu tapahtuu joko ilmaan heittämällä, linkoamalla tai kiitoradalta kiihdyttämällä. Helikopterityyppisen UAV:n vahvuuksia ovat pystysuoraan liikkeellelähto ja laskeutuminen. Helikopterityyppinen UAV pystyy myös paikallaan leijuntaan ja hitaaseen lentoon. UAV-laitteita käytetään esim. ilmakuvaukseen ja metsäpalon tai kaasuputken valvontaan. Sotilaskäytössä UAV:tä käytetään mm. tiedustelussa, mutta myös taistelu- ja huoltotehtävissä. Tieteellisessä käytössä UAV:tä on käytetty esim. trooppisten hirmumyrskyjen tutkimisessa.

#### Phantom 2

Tässä työssä kehiteltävä ja rakennettava mittausjärjestelmä tullaan kehittämään UAV-helikopterille. Phantom 2 (kuva 5) on DJI-yhtiön kaupalliseen käyttöön valmistama UAV-helikopteri.



Kuva 5. Phantom 2 UAV -helikopteri. [17.]

Pääsääntöinen kohderyhmä kopterin käyttäjille ovat video- ja valokuvaajat. Phantom 2 on nelimoottorinen helikopteri, jossa on Naza-V2-ohjausyksikkö. Phantom 2:lle luvataan 25 minuutin lentoaikaa ja tarkkaa toimintaa ilmassa. Nämä ominaisuudet tekevät Phantom 2:sta vakaan ja erinomaisen kuvaamisen kannalta. Phantom 2:een on mahdollista asentaa kameragimbal, joka luo videokuvaan vakautta ja pitää kameraa tasallaan lennon aikana välittämättä sen hetkisen säätilan aiheuttamasta vaikutuksesta lennokkiin. Tosin Phantom 2 on suunniteltu lennätettäväksi vain poutasäällä, sillä helikopterin sähköisiä osia ei ole suojattu roiskevettä vastaan. Taulukko 1:ssä on esitelty mainitsematta jääneet Phantom 2:n tekniset ominaisuudet. [17.] [18.]

Taulukko 1. Phantom 2:n tekniset ominaisuudet. [18.]

Paino	1000 g
Nopeus	6 m/s Nousu/Laskunopeus n. 10 m/s lentonopeus
Lentoaika	25 min
Virtalähde	5200 mAh LiPo 3S akku, Paino 340 g
Radiotaajuus	2,4 Ghz ISM
paikallaan pysyvyys	0,8 m/pystysuunta ja 2,5 m vaakasuunta
Nostokyky	1,3 kg MAX
Lentosäde	1000 m
Virrankulutus	100 mA/6 V
Latausteho	4 A
Toimintalämpötila	
Max kulmat	Yaw kulma: 200 astetta / sek / max tilt kulma: 35 astetta

## 4 KUVANTAMIS- & MITTAUSJÄRJESTELMÄN SUUNNITTELU

Kuvantamisjärjestelmän tarkoituksena on kuvantaa Phantom 2 -helikopterilla kasvillisuudesta lehtivihreää sekä IR- että RGB-kamerajärjestelmällä kameramoduulien avulla. IR- ja RGB-kameroina käytetään Raspberry Pi:n omia kameramoduuleja: Raspberry Pi -kamera ja NOIR-kamera. Yksi työn haaste on saada kaksi RPI-kameraa liitettyä itse Raspberry Pi:hin. Raspberry Pi:ssä itsessään on vain yksi kamerapaikka. Kuvien ottohetkellä luetaan GPS-signaali, tallennetaan paikkatieto signaalista taulukkoon ja tallennetaan myös lentolokitiedot. Kameramoduulien ohjaus, kuvanotto ja GPS-tiedon tallentaminen tapahtuu opinnäytetyönä Python-ohjelmointikielellä luodussa ohjelmassa. Lisäksi Raspberry Pi:hin liitetään digitaalisia antureita, kuten kiintyvyyssantureita, kertomaan itse helikopterin asennosta ja suunnasta, mutta myös kameroiden asennosta. Nämäkin tiedot tallennetaan kuvienottohetkenä, mutta myös jatkuva-aikaisesti. Kuvantamisjärjestelmän valvominen ja ohjaus tapahtuu kannettavalta tietokoneelta langattomasti. LabVIEW:lla toteutettava käyttöliittymä ottaa vastaan kuvia lennon aikana käyttäjän nähtäville. Kaikki kuvat ja tiedot tallentuvat SD-muistikortille, josta ne voidaan poimia ja kolmannen osapuolen ohjelmalla analysoida esim. Mosaic-kartaksi tai ortokuviksi.

### 4.1 IR- ja RGB-kamera

Työssä käytetään aiemmin esiteltyjä Raspberry Pi -kameramoduuleja kuvien ottamiseksi. Raspberry Pi -kameramoduuli on ominaisuuksiltaan 5 MP, 2592x1944 resoluutiolla varustettu erittäin pienikokoinen kameramoduuli. Työssä puhutuista IR- ja RGB-kamerasta tämä kameramoduuli on RGB-kamera. IR-kameraa työssä edustaa Raspberry Pi NOIR-kameramoduuli. NOIR-kameramoduulissa ei ole infrapunasuodatinta, jonka vuoksi sillä voidaan kuvata infrapuna-aallonpituuksia, kuten lehtivihreää. Muilta ominaisuuksiltaan NOIR-kamera on RGB-kameran kaltainen. Paitsi NOIR-kamerassa on itsessään jo sisäisesti ohjelmoituja automaattisia kuvantamisfunktioita ja ohjelmoitava kuvantamistaajuus.

## 4.2 Raspberry Pi -kameran multiplekseri

Tavallisesti Raspberry Pi:hin saa vain yhden kameramoduulin kiinni kerrallaan molex-liitännällä CSI-väylään. Tämä väylä on erityisesti suunniteltu kyseisille kameramoduuleille. Tällä väylällä on erittäin suuret datanopeudet, josta väylä vie pikselidatan suoraan BCM2835-prosessorille. Suuren datanopeusvaatimuksen takia vaaditaan myös kameraa vaihtavalta kytkimeltä nopeutta. Tämän takia kahden kameran kytkeminen Raspberry Pi:hin vaati erillisen lisäkortin, jossa on kytkentälogiikalla ohjattava multiplekseri. Tätä tarkoitusta varten on turkkilainen IVmech Mekatronik suunnitellut, kehittänyt ja tuonut markkinoille IVPort Raspberry Pi Camera Module Multiplexer:n (kuva 6). Tämä Raspberry Pi -lisäkortti on varustettu I<sup>2</sup>C multiplekserillä, jolla voidaan ohjata neljää Raspberry Pi -kameramoduulia. Lisäkortti pystyy vaihtamaan kameroita keskenään 300 millisekunnissa. Lisäkortteja voidaan myös pinota päällekkäin, ja näin saadaan jopa 16 kameran kytkentämahdollisuus Raspberry Pi:hin. [19.]

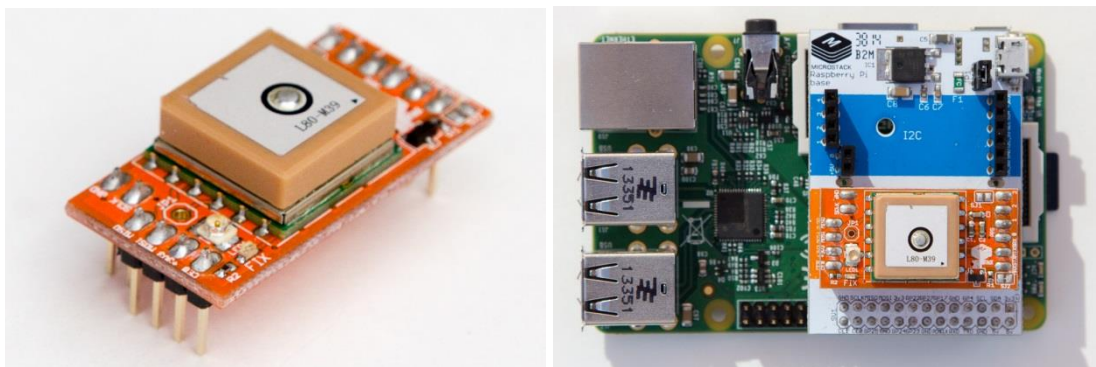


Kuva 6. IVPort Raspberry Pi Camera Module Multiplexer. [19.]

### 4.3 GPS-signaali

GPS eli Global Position System (suom. ”maailmanlaajuinen paikallistamisjärjestelmä”) on Yhdysvaltain puolustusministeriön rahoittama ja kehittämä satelliittipaikannusjärjestelmä [20].

Opinnäytetyössä kuvienottohetkellä luetaan myös Raspberry Pi:hin liitettyä olevaa GPS-anturia. Työhön valikoimme Microstack GPS-anturin (kuva 7).



Kuva 7. Mikrostack GPS-anturi ja Raspberry Pi model B+. [21.]

Mikrostack GPS-anturi on matalavirtainen, pulssi per sekunti GPS-signaalia vastaanottava, sisäänrakennetulla GPS-datatallennuksella ohjelmoitava GPS-anturi. Se on rakennettu Raspberry Pi:n käyttöön. Mikrostack GPS-anturille on suunniteltu valmiiksi ohjelmointikirjasto, jota käyttämällä GPS-anturin ohjelmointi onnistuu yksinkertaisesti ja GPS-datan poiminta tehokkaasti. GPS-moduuli kommunikoi RPI:n kanssa sarjaliikenteen kautta. Sarjaliikenteessä ei tällöin voi olla muita laitteita, jotka käyttävät sarjaliikennettä kommunikointiin. GPS-moduulia käytetään kuvienottohetkellä ja paikkatieto tallennetaan taulukkomuotoiseen tiedostoon. GPS-tietoja, kuvia, voidaan käyttää tämän jälkeen kolmannen osapuolen ohjelmassa erilaisten karttapohjien luomiseen, kuten Mosaic-kartan tai ortokuvien tekemiseen.[22.] [23.]



#### 4.4 Digitaaliset anturit

Insinööriyöhön kuuluu digitaalisen mittausanturin kytkeminen RPI:hin. Insinööri-työn aikana Raspberry Pi -mittausjärjestelmään valittiin liitettäväksi yksi mittaus-anturi: MiniIMU-9 v3-mittausyksikkö.

##### MiniIMU-9 v3 -mittausyksikkö

Insinööriyössä liitettiin Raspberry Pi:hin digitaalinen kiihtyvyyssanturi. Kiihtyvyyssanturiksi valikoimme Pololun MiniIMU-9 v3 -mittausyksikön (kuva 8), joka tarkemmin on inertiaalinen mittausyksikkö. MiniIMU-9 v3 pitää sisällään L3GD20H 3-akselisen gyro-anturin, LSM303D 3-akselisen kiihtyvyyssanturin ja 3-akselisen magneettisuusanturin. Mittausyksikkö käyttää kommunikointiin I<sup>2</sup>C-rajapintaa, jonka kautta voidaan kaikilta yhteensä yhdeksältä akselilta mitata kierto-, kiihtyvyyss- ja magneettiset arvot, joiden avulla voidaan laskea anturin absoluuttinen suunta. Mittausyksikkö sisältää myös jänniteregulaattorin, jonka ansiosta piirilevy pystyy operoimaan 2,5 V – 5,5 V jännitealueella. [24.]



Kuva 8. Pololun MiniIMU-9 v3 -mittausyksikkö. [24.]

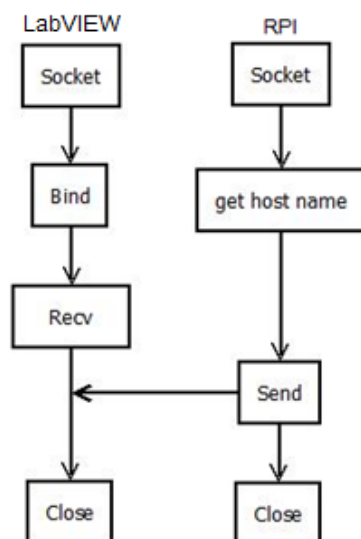
Taulukko 2. Pololun MiniIMU-9 v3 -mittausyksikön tekniset ominaisuudet. [24.]

Mitat	20 mm × 13 mm × 3 mm
Paino	0.7 g
Käyttöjännite	2,5 V to 5,5 V
Virta	6 mA
Herkkyyalueet:	
Gyro	±245, ±500, or ±2000°/s
Kiihtyvyys	±2, ±4, ±6, ±8, ±16 g
Magneettisuus	±2, ±4, ±8, ±12 gauss

## 4.5 Tiedonsiirto UDP

Raspberry Pi:llä toteutettavan kuvantamis- ja mittausjärjestelmän valvominen tapahtuu langattomalla yhteydellä RPI:stä Pythonilla toteutettavalla ohjelmalla National Instrumentin LabVIEW-ohjelmaan, joka toimii erillisessä kannettavassa tai pöytätietokoneessa. Pythonilla toteutettavan kuvantamisohjelman ja LabVIEW:llä toteutettavan valvontaohjelman keskinäinen liikenne tapahtuu UDP-protokollalla, joka on ns. yhteydetön protokolla. Se ei vaadi laitteiden välistä yhteyttä, mutta tiedonsiirto on tällöin mahdollinen. UDP:ssä tiedonsiirtopakettien perillemeno ei varmisteta kuittauksilla tai alkukäsittelyllä. Tämän ansiosta UDP on myös kevyempi suoritettavana prosessina kuin esim. TCP. [25]

Ohjelman toiminta on yksinkertainen. Raspberry Pi:ssä kuvantamisohjelma ottaa valokuvat ja poimii GPS-koordinaatit kuvanottohetkeltä. Tässä prosessista poimitaan joka viides kuva ja kuvan GPS-koordinaatti lähetettäväksi. Pythonilla toteutettava pääohjelman säieohjelma lähettää kuvat ja datan LabVIEW:n ohjelmalle. LabVIEW-ohjelmassa kuva ja sen koordinaattitieto esitellään käyttäjälle, joka valvoo lennon kuvauksen suorittamista. Tiedonsiirron vuokaavio on esillä kuvassa 9. Valvontaohjelmalta ei lähetetä puolestaan mitään tietoa taikka muuttujia Raspberry Pi:llä toimivalle kuvantamis- ja mittausohjelmalle.



Kuva 9. UDP-protokollan lähetysvuokaavio.

#### 4.6 Pythonilla toteutettava ohjelma

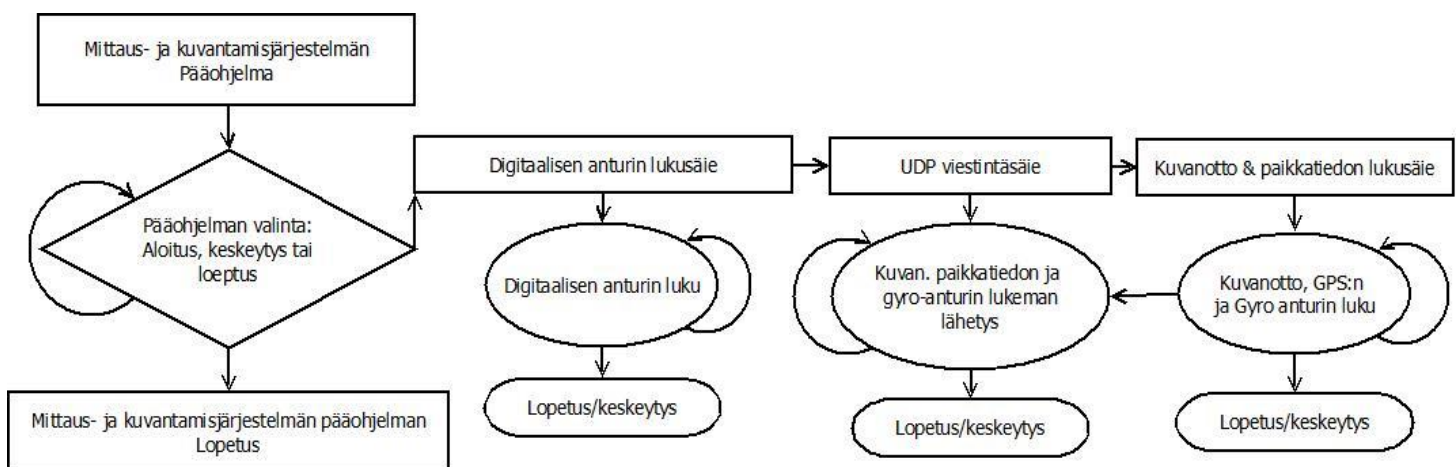
Ohjelmointikielenä insinööriyössä käytetään Pythonia. Python on olio-ohjelmallinen ja aktiivinen ohjelmointikieli. Pythonia voidaan verrata muihin ohjelmointikieliin, kuten C#, Visual Basic tai Java. Pythonin ohjelmakääntäjä puolestaan rakentuu C- ja C++-kieleen. [26.] [27.]

Raspberry Pi:llä toteutettavan kuvantamis- ja mittausjärjestelmän ohjelman ydin on Python-ohjelmointikielellä toteutettava pääohjelma, jolla ohjataan suoritettavia säikeitä. Pääohjelman jako ohjelmakokonaisuuksiin tapahtuu säikeistämällä (eng. multithreading). Säikeistäminen on ohjelman prosessointitapa, jolla pystytään ajamaan kahta tai useampaa prosessia rinnakkain, ilman että ne häiritsevät toinen toistaan. Suunniteltuja säikeitä ovat GPS-signaalin käsittely- ja kuvanotto-säie, UDP-protokollalla toteutettava GPS-tietoja ja -kuvia lähettävä säie LabVIEW:llä suoritettavalle valvontaohjelmalle ja digitaalista anturia lukeva ja tietoja kirjoittava säie. Ohjelman pitää siis sisältää kolme kappaletta säikeitä ja pääohjelman suorituksen. Ohjelman toiminnan kulkua esitetään kuvan 10 vuokaaviossa.

Seuraavaksi käyn läpi ohjelman suoritusta. Ohjelman suoritus alkaa GPS:n, kameroiden, päivämäärän, I<sup>2</sup>C-rajapinnan, kameroita ohjaavan multiplekserin ja monien muiden ohjelmakirjastojen alustuksesta. Suunnitellun ohjelman suoritus jatkuu tässä pääohjelmassa seuraavaan valintaan, jossa päätetään, käynnistetäänkö kuvan ottaminen, GPS-datan kirjoitus ja digitaalisen anturidatan kirjoitus. Digitaalisena anturina tässä säikeessä toimii aikaisemmin esitelty Pololun Mini-MU-9 v3, joka käyttää Raspberry Pi:n kanssa kommunikointiin I<sup>2</sup>C-rajapintaa. GPS-anturina tehtävänsä toimittaa Microstackin GPS-vastaanotin, ja kuvanotto tapahtuu Raspberry Pi:n kameroilla. Käyttäjä voi tässä pääohjelmassa valita aloittavansa kuvanotton ja GPS-tietojen talteenottosäikeen. Kun käyttäjä on valintansa tehnyt ja esim. aloittaa GPS & kuvantamisen, käyttäjä näkee, että kuvan ottaminen ja GPS-tietojen tallennus on alkanut, mutta voi myös valita keskeyttävänsä tai lopettavansa aloittamansa ohjelman. GPS & kuvanotto-ohjelmasäikeessä tapahtuu toistorakenne kuvanoton, GPS-datan tallennuksen

ja UDP-lähetyskyselyn välillä. Käyttäjälle näkymättömät säikeet ovat digitaalisen datan kirjoittava ja UDP-protokollalla toimiva kuvia ja GPS-tietoja lähettävä säie. Nämä säikeet käynnistyvät ja alustuvat jo ennen GPS & kuvanotto -säiettä. Täysin näkymättömiä säikeet eivät käyttäjälle ole, vaan säikeiden käynnistyminen ja lopettaminen näkyvät pääohjelmassa. Tietoja lähettävä säie saa luvan ja tiedot GPS & kuvanotto -säikeeltä lähettää kuva ja GPS-tiedot valvovalle ohjelmalle joka viidennellä ohjelman toistokerralla. Digitaalista anturia lukeva ja kirjoittava ohjelma toimii aloituksen jälkeen itsenäisesti kirjoittaen anturin tietoja talteen Raspberry Pi:lle. Yksinkertaisesti anturidata-tietoja ei käsitellä mitenkään ohjelman suorituksen aikana reaaliaikaisesti, ainoastaan reaaliaikainen tallennus on tärkeää tässä säikeessä.

Kuten aiemmin mainittiin, käyttäjä pystyy keskeyttämään tai lopettamaan ohjelman suorittamisen koska vain. Päätettäessä näin tapahtuu säieohjelmien portaittainen lopetus, mutta kukin ohjelma suorittaa tehtävänsä lopetuslippuun (exit-Flag) asti. Keskeytettäessä ohjelman suorittamisen säikeiden toiminta myös loppuu, mutta ohjelman suoritus palaa pääohjelmaan, josta se voidaan jälleen käynnistää. Ohjelman lopetuksessa pääohjelmassa tapahtuu Raspberry Pi -kameroita ohjaavan multiplekserin alustus, jotta kamerat siirtyvät lepotilaan oikein. Onnistuneessa tapauksessa ohjelman suoritus pääsee päätökseensä ilman virheitä. Ohjelman suorituksen vuokaavio on kokonaisuudessaan nähtävillä liitteessä 1.



Kuva 10. Toteutettavan ohjelman yksinkertainen vuokaavio.

## 5 KUVANTAMIS- & MITTAUSJÄRJESTELMÄN TESTAUS

Kuvantamis- ja mittausjärjestelmän testaaminen tapahtui pienissä erissä aina ohjelmakoodin onnistuneesta luomisesta. Testaamisessa keskityttiin ohjelman käytettävyyden, toimivuuden ja luotettavuuden varmistamiseen. Lopullinen testi tapahtui ohjelman säikeistämisen luotettavuuden ja toimivuuden testauksessa.

### 5.1 GPS-datan kirjoittaminen ja kuvanotto

Ensimmäisenä kuvantamisjärjestelmän ohjelmistosta valmistui Microstackin GPS-anturin ja yhden RPI-kameramoduulin käytön testaus. Kuten aiemmin esiteltiin, tälle GPS-anturille on sen valmistaja luonut oman Python 3.0 -ohjelmakirjaston. Tämä nopeuttaa huomattavasti työn edistymistä, koska anturin käyttämää sarjaliikennettä rajapinnasta ei tarvitse itse parsia tai poimia sarjaliikenteestä oikeaa GPS NMEA -tunnusta ja siitä paikkatietoja. Microstacknoden ohjelmakirjastoon voidaan valita sopiva GPS-signaalista poimittava NMEA-protokolla, josta löydämme kaiken tarvittavan tiedon tiedostoon kirjoitettavaksi. Työssä hyödynnetään GPGGA NMEA-protokollaa, jota käytetään GPS-signaalikäsitelyssä tarkan 3D-mallisen paikan määrittämiseen. Esimerkkisignaalin `$GPGGA, 170834, 4124.8963, N, 08151.6838, W, 1, 05, 1.5, 280.2, M, -34.0, M, , *75` käsittelemä tieto on esitetty taulukossa 3. Tästä tiedosta otetaan talteen aika, leveysaste (Latitude), pituusaste (Longitude) ja korkeus (Altitude), jotka kirjoitetaan tiedostoon aina kuvanottohetkeltä. [28.] [20.]

Kuvanottohetkellä Raspberry Pi -kameramoduuli nappaa kuvan aina tietyn ajan välein. Raspberry Pi -kameramoduulin käyttöönotto RPI:ssä on varsin yksinkertainen prosessi. Ensin sallitaan RPI-kameran käyttö Raspberry Pi:n esiasetusten (Raspi-Config) määrittelyssä, jonka jälkeen ladataan Pythonin ohjelmakirjastot kameramoduulin käyttöä varten. Pythonin RPI-kameran ohjelmakirjaston avulla

kameran käyttö on yksinkertaista, sama kirjasto toimii myös Raspberry Pi:n normaalin RGB-kameran kuin NOIR-kameran kanssa. GPS-datan kirjoittaminen ja kuvanotto tapahtuu aikaisemmin esitellyn kuvan 10 vuokaavion paikkatieto & kuva -säikeessä.

Ohjelmasäikeessä tapahtuu GPS-signaalista GPGLL:n lukeminen, taulukkotiedoston luominen ja sinne kirjoittaminen sekä lisäksi kuvan ottaminen. Näitä ominaisuuksia testattiin ensimmäisissä testeissä. Mitään suuria ongelmia ei testien aikana ollut. GPS-tiedot ja kuvat tallentuivat kansioihin ja työn suoritus jatkui seuraavaan vaiheeseen.

Taulukko 3. GPGLL:n sisältämä tieto. [28.]

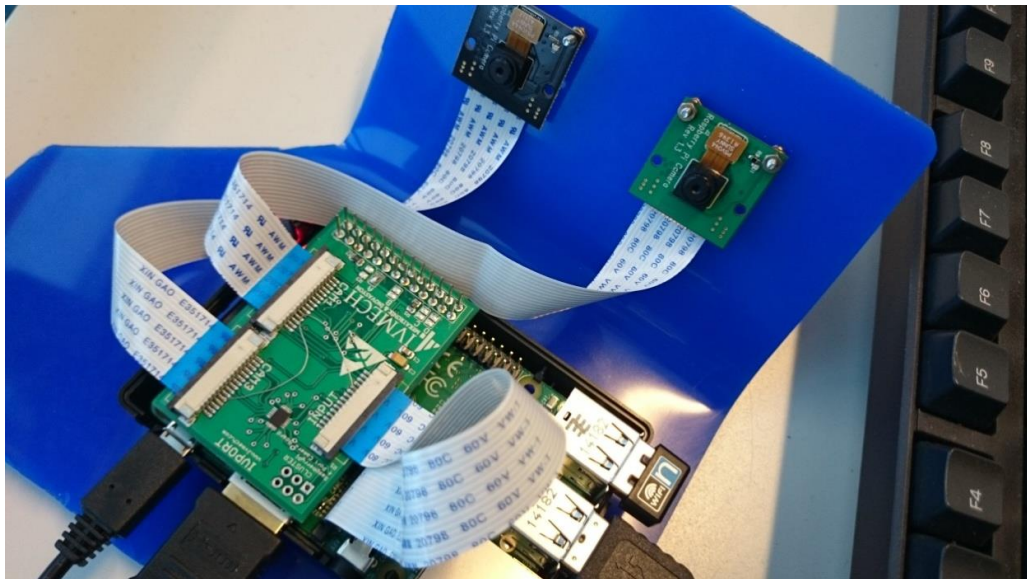
Nimi	Esim.	Kuvaus
Tunniste	\$GPGLL	Maailmanlaajuisen Paikannusjärjestelmän Fix data
Aika	170834	17:08:34
Leveysaste/Latitude	4124.8963, N	41° 24' 54" N
Pituusaste/Longitude	08151.6838, W	81° 51' 41" W
Fix data: 0=Pätemätön, 1= GPS Fix, 2=DGPS Fix	1	Tieto GPS Fixiltä
Satelliittien määrä	5	
Vaakatarkkuuden virhe (HDOP)	1,5	
Korkeus/Altitude	280.21, M	280.2 metriä meren pinna yläpuolella
WGS84 Korkeus	-34.0 , M	Geoidin WGS85 Korkeus
Aika viimeisestä DGPS päivityksestä	tyhjä	ei käytössä GPGLL:ssa
DGPS tukiasema tunnus	tyhjä	ei käytössä GPGLL:ssa
Checksum	*75	Lähetysvirheiden tarkistus

## 5.2 IVPort Raspberry Pi -kameramultiplekseri

Luvussa 4.2 esitelty RPI-kameramultiplekseri on pitkään etsitty ja odotettu lisäkortti Raspberry Pi:lle. Lisäkortin valmistaja IVmech Mekatronik oli tämän työn tekoaikana ainoa, jolla oli markkinoilla kyseiseen tarkoitukseen kehitetty lisäkortti. Lisäkortti oli myös tullut julki vasta hiljattain vuoden 2014 syyskuun aikana. Lisäkortin saapumista jouduttiin odottamaan aina marraskuun loppuun saakka.

### IVPort RPI -multiplekserin kytkentä

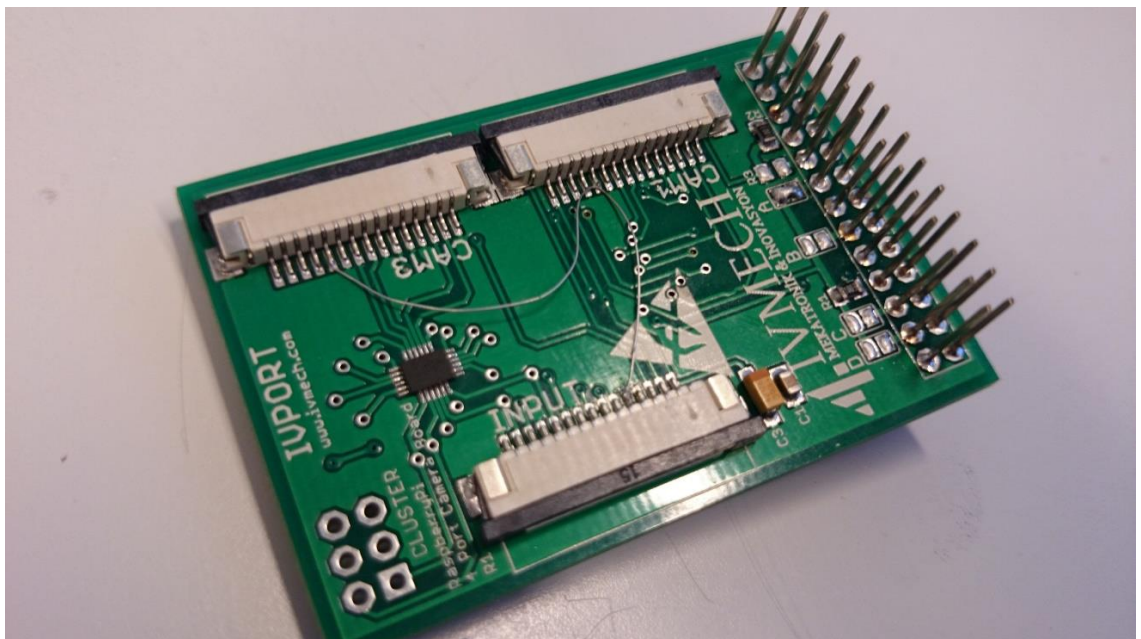
RPI-kameramultiplekserilisäkortin saavuttua kortti liitettiin Raspberry Pi:hin kuvan 11 mukaisesti. Valmistajalla on kotisivuillansa valmiita Pythonin esimerkkejä, joita lähdettiin testaamaan kortilla. Kameroiden ohjaus tapahtuu multiplekserilisäkortissa totuustaulukon avulla, jossa annetaan kolmelle parametrille, tässä tapauksessa kolmelle eri pinnille, Raspberry Pi:llä arvot joko 0 tai 1.



Kuva 11. Raspberry Pi -kameramultiplekseri liitettynä RPI:hin.

Moduuli vaikutti olevan fyysisesti ehjä ja ohjeiden mukaiset juotoksetkin korttiin onnistuivat. Mutta ongelmia oli alkuvaiheessa. Niin valmistajan antamat testauskoodit kuin itse kehitetyt testaukset eivät onnistuneet Raspberry Pi:llä. Ongelmana oli, että toinen kahdesta korttiin kiinnitetyistä kamerasta otti hyvän kuvan, kun toinen kameroista ei saanut muodostettua kuvaa lainkaan tai kuva oli erittäin vir-

heellinen. Myöhemmin valmistaja tiedotti, että kyseisen IVPort RPI -multiplekseri lisäkortin piirilevyn suunnittelussa on tapahtunut virhe. Tämä johtui siitä, että kameroiden molex-porteista oli kokonaan jäänyt 11-pinnin kohdalta vetämättä piirilevylinja muihin portteihin, jolloin varsinkin uudempien kameramoduulien kanssa tapahtui virheellinen ohjelmallinen toiminto, kuten NOIR-kameramoduulin kanssa. Valmistaja suosittelikin, että ensimmäisen version lisäkortteihin juotettaisiin hyppylangat jokaisen 11 pinnin kohdalle. Kuvassa 12 on havainnoitu tämän hyppylangan juotosta. Onnistuneen juotoksen jälkeen kameran vaihto alkoi toimimaan. [29.]



Kuva 12. Hyppylangan juotos RPI-kameran ohjainkorttiin.

#### IVPort RPI -multiplekserin ohjelmointi

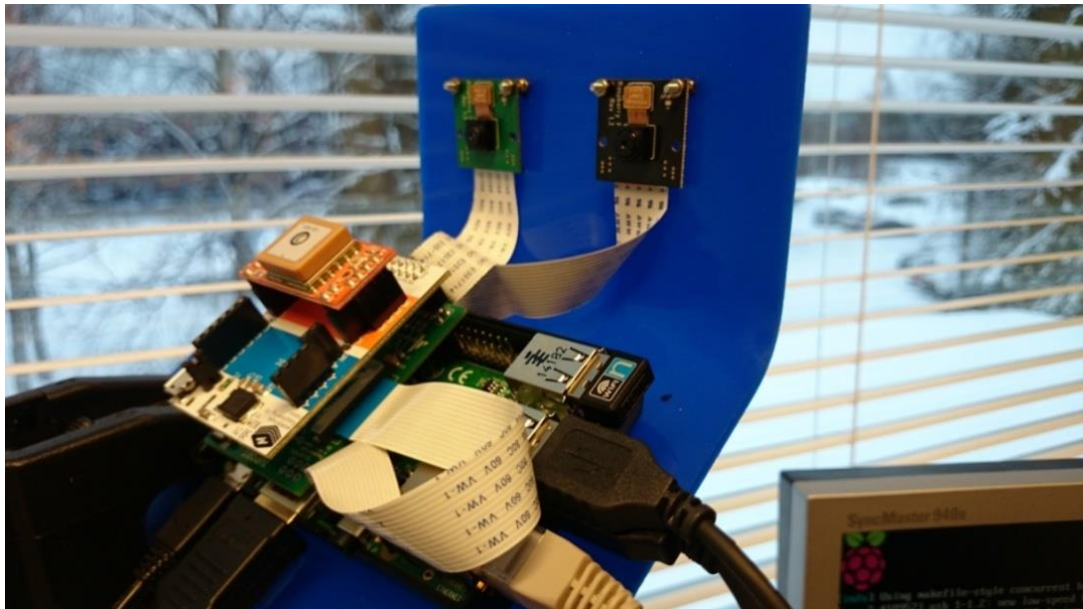
Raspberry Pi -kameramultiplekserin fyysisen korjauksen jälkeen päästiin jatkaamaan ohjelman testaukseen. Pythonilla suoritettulla testausohjelmalla havaittiin että RPI-multiplekserin lupaama 300 millisekunnin kameran vaihtonopeus oli jostain syystä liian nopea RPI:n suoritettavaksi. Ongelma korjaantui lisäämällä ohjelmoidusti viivettä kameroiden vaihtokomentojen välille noin 100 millisekunnin verran. Näin ollen kamerat ottavat kuvan noin 400 millisekunnin viiveellä toisistaan. Eli Raspberry Pi -kameramultiplekserillä saadaan otettua kuva samalla se-



kunnilla molemmilla kameroilla. Kameran vaihtoajan vaikutus kuvaan voidaan kylläkin huomata kuvattaessa liikettä, esimerkiksi käden heilautusta. Ensimmäisen kameran kuvassa käsi voi olla kuvan yläreunassa, kun toisen kameran kuvassa käsi on laskeutunut jo alemmas. Tätä ongelmaa ei voida poissulkea muulla kuin kahdella Raspberry Pi -alustalla, jotka ovat keskenään synkronoitu. RPI-multiplekserin viive on pysyvä ja yksi kortin ominaisuuksista.

### 5.3 GPS & kuvantamisen sekä tiedonsiirron säikeistys

Luvussa 5.1 todettu ja testattu GPS:n ja kameran yhteistoiminta muokattiin Raspberry PI -kameramultiplekserin testin myötä toimivaksi kahden kameran kanssa. Aiemmin testattun yhden kameran ja GPS:n yhdistelmää ei kuitenkaan unohdeta. Yhdellä kameralla kuvantaminen on mahdollinen vaihtoehto jatkoa ajatellen, kun tarkoituksena on nopeammassa liikkeessä kuvaaminen ja GPS-tietojen talteenottaminen. Työn kuvantamis- ja mittausjärjestelmä GPS-anturin ja kahden kameran kanssa on esitetty kuvassa 13.



Kuva 13. Raspberry Pi, kaksi kameraa ja GPS.

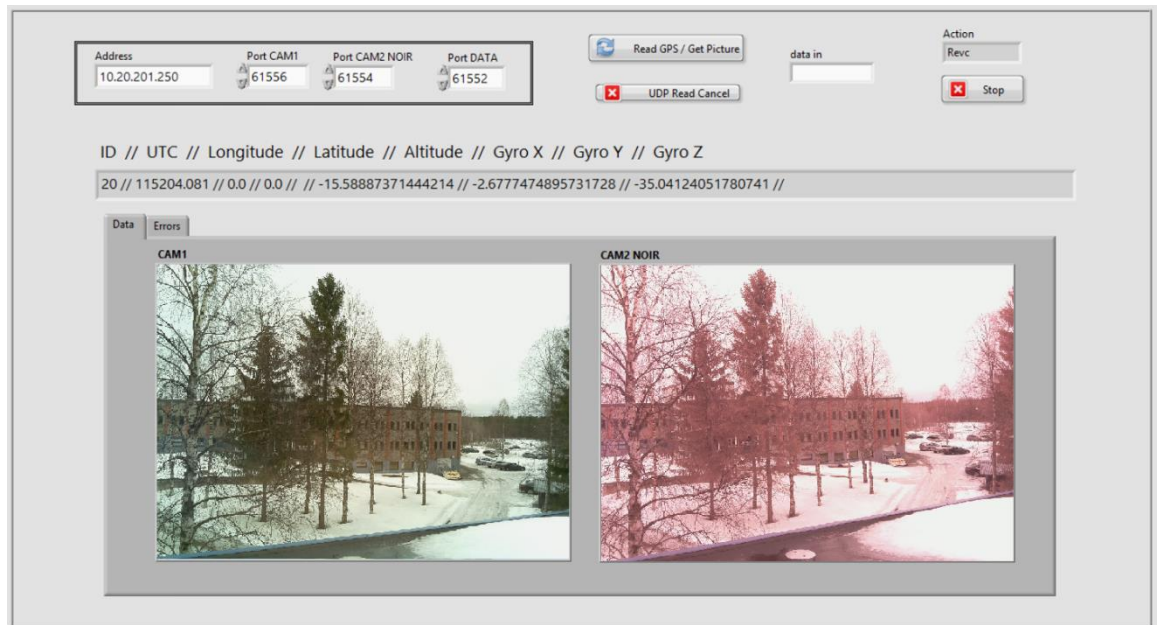
Itse GPS-anturin ja kameran yhteistyön testauksesta kerrottiin aiemmassa luvussa 5.1. Tässä työvaiheessa tapahtui ohjelmallinen muutos. Aikaisemmassa testissä ohjelman suoritus tapahtui vain GPS:n ja kameran välillä pääohjelmassa ilman säikeistymistä. Tässä vaiheessa luodaan säikeistetty ohjelma, eli luodaan pääohjelma, jonka avulla hallitaan kuvantamis- ja mittausjärjestelmän toimintaa.

Ohjelmäsäierakenne on esitelty luvussa 4.6 kuvassa 10 ja liitteessä 1. Säieohjelmoinnilla saadaan aikaiseksi moniosainen ohjelma, jossa ohjelman itsenäiset komponentit (liite 1, paikkatieto & kuva -säie) eivät ole riippuvaisia pääohjelmäsäikeen tapahtumista ja toisin päin. Pääohjelmäsäikeen tapahtumat voivat silti vaikuttaa muiden säieohjelmien toimintaan käyttäjän määrittelemillä muuttujien arvoilla, jotka ovat näkyviä jokaiselle ohjelmäsäikeelle. Insinööriyössä tällaisia muuttujia ovat keskeytys/lopetuslippu (liite 1, ExitFlag), joiden avulla keskeytetään tai lopetetaan paikkatieto & kuva -säikeen toiminta. Tämän lisäksi lähetyslippu (liite 1, sendFlag) vastaa lähetyksluvan antamisesta UDP-säikeelle.

### 5.3.1 LabVIEW-valvontaohjelma

Toinen insinööriyön säie on UDP-yhteydellä lähetettävät GPS-anturin tiedot ja molempien kameroiden ottamat kuvat joka viidenneltä kuvantamiskerralta RPI-kuvantamis- ja mittausjärjestelmää valvovalle LabVIEW:llä toteutetulle ohjelmalle (kuva 14). Ohjelman työstön aikana tehtiin myös päätös lisätä gyro-anturilta kulmatiedot paikkatiedon lähetyksen yhteyteen. Tietoja lähetävä säie toimii itsenäisesti joka viidennellä kuvantamis- ja GPS:n talteenottokerralla. Ohjelma ottaa käsittelyyn viimeisimmän viidellä jaollisen olevan kuvan kansioista, johon kuvat ovat tallentuneet. Se muokkaa kuvat sopivan kokoisiksi lähetystä varten ja suorittaa lähetyksen. Kuvan lähetyks tapahtuu siten, että RPI:ssä toimiva ohjelma luo JPEG-tyyppisestä kuvasta base64-koodatun tekstidatan, jonka se lähettää toiselle ohjelmalle. LabVIEW:n ohjelmassa base64-koodaus puretaan ja kuva muodostetaan tekstidatasta, joka vastaa RPI:llä otettua kuvaa.

Paikkatiedon UDP-protokollalla toimiva ohjelmasäie ottaa kuvanottohetkellä tallennetusta GPS-vastaanottimen lukukerrasta, joka on sama kuin lähetettävä kuvan paikkatieto. Gyro-anturilta luetaan kulma-arvot samaan lähetykseen, jossa GPS-tiedot lähetetään valvovalle ohjelmalle. Kuvien ja tietojen lähetyksen jälkeen tämä lähetettävä säie jää odottamaan seuraavaa lähetyksertaa.



Kuva 14. LabVIEW-valvontaohjelman käyttöliittymä.

### 5.3.2 Kuvantamisen ja mittauksen pääohjelma

Säikeistettävän ohjelman luominen ja testaus oli tämän insinööriyön haastavin ja aikaa vievin työosuus. Aikaisempi kokemus säikeistettävän ohjelman luomisesta ja toiminnasta rajoittui vain syksyllä 2014 suoritetuille opinnoille Kajaanin ammatikorkeakoulussa. Nämä opinnot suoritettiin C-kielellä Linux-ympäristössä. Python-ohjelmakielikin oli insinööriyön aikana itsenäisesti opiskeltava ohjelmointikieli. Useamman testauksen ja yrityksen jälkeen saatiin aikaan toimiva säikeistetty ohjelma. Ohjelman visuaalinen toiminnan esimerkki on esillä kuvassa 15.

```

pi@raspberrypi ~ $ sudo python3 PhantomRev1.0.py

$GPGGA,112548.000,6413.1013,N,02746.7086,E,1,4,2.57,127.7,M,19.4,M,,*51

### Phnatom 2. GPS Pic Program ### MAIN ###
--> c for Continue / t to Stop GPSPIC / q for Quit program: c
--> Set up gap between pictures, default min 2.0 seconds: 2

* Flight/GPSPIC Rounds: 01

* files Create/Writed

* Starting DigiRaw-Thread
* Starting UDPSend-Thread
* Starting GPSPIC-Thread
# Campture and GPS: Start #

### MAIN ###
--> Hit c for Continue / t to Stop GPSPIC / q for Quit program and ENTER:
$GPGGA,112549.000,6413.1001,N,02746.7084,E,1,4,2.57,127.9,M,19.4,M,,*5F

<> 01 <> 2.421112 second
$GPGGA,112550.000,6413.1002,N,02746.7074,E,1,4,2.57,127.8,M,19.4,M,,*5A

<> 02 <> 2.124825 second
$GPGGA,112551.000,6413.1002,N,02746.7063,E,1,4,2.57,127.8,M,19.4,M,,*5D

<> 03 <> 2.190582 second
$GPGGA,112552.000,6413.1003,N,02746.7043,E,1,4,2.57,128.0,M,19.4,M,,*5A

<> 04 <> 2.190315 second
$GPGGA,112553.000,6413.0998,N,02746.7028,E,1,3,3.98,128.1,M,19.4,M,,*58

<> 05 <> 2.223492 second
$GPGGA,112554.000,6413.0991,N,02746.7014,E,1,4,2.57,128.2,M,19.4,M,,*5F

<> 06 <> 2.438989 second
$GPGGA,112555.000,6413.0985,N,02746.6991,E,1,4,2.57,128.2,M,19.4,M,,*5E

<> 07 <> 1.811540 second
$GPGGA,112556.000,6413.0974,N,02746.6974,E,1,4,2.57,128.4,M,19.4,M,,*5E

<> 08 <> 2.124748 second
$GPGGA,112557.000,6413.0962,N,02746.6955,E,1,4,2.56,128.5,M,19.4,M,,*5B

t
* Exiting UDPSend-Thread

* Exiting DigiRaw-Thread
<> 09 <> 2.110334 second

# Campture and GPS: Stop #
* Exiting GPSPIC-Thread

### MAIN ###
--> Hit c for Continue / t to Stop GPSPIC / q for Quit program and ENTER:
q

### Phnatom 2. GPS Pic Program ### MAIN ### Ending, Good bey!

```

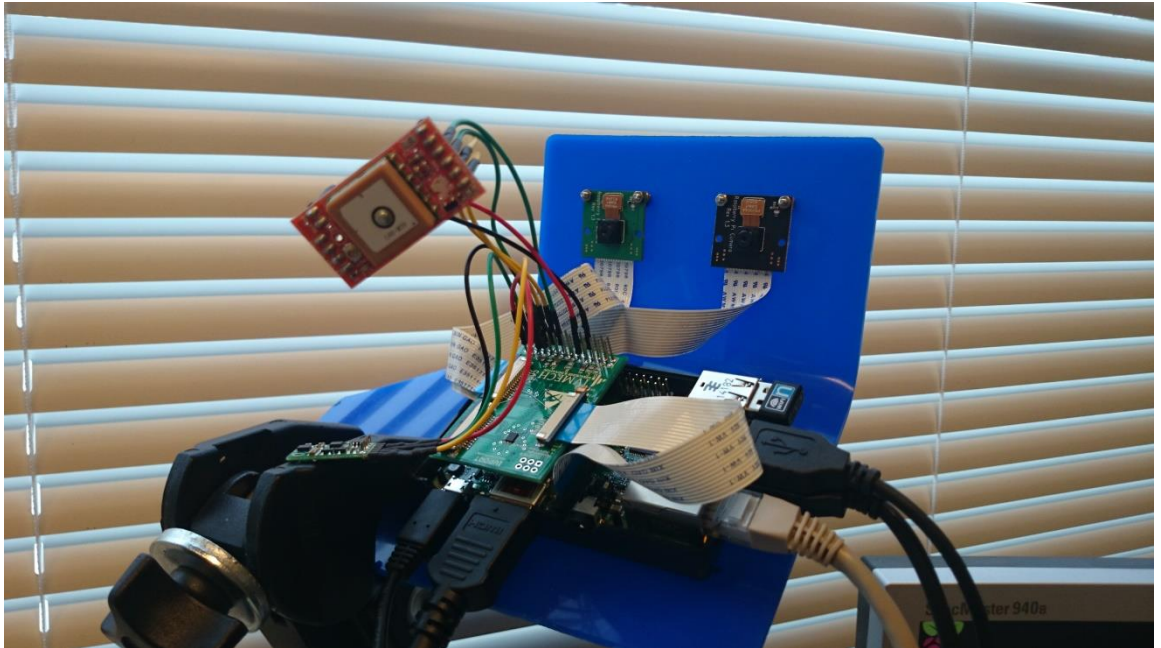
Kuva 15. Säikeistetyn ohjelman visuaalinen näkymä RPI-terminaalissa.

Ohjelma näyttää käyttäjälle ensimmäisenä GPS-signaalista alustettavan NMEA-osuuden ja NMEA:n senhetkisen arvon. Tämän jälkeen käyttäjä tekee valinnan: aloittaako kuvantamis- ja mittausjärjestelmän käytön vai lopettaako ohjelman. Käyttäjän jatkaessa ohjelman suorittamista ohjelma ilmoittaa aloittavansa kuvantamisen ja mittauksen sekä luovansa kansion, johon tallentaa kuvat ja GPS:n paikkatiedon sekä digitaalisen anturin mittautustiedot. Ohjelma tulostaa käyttäjälle näkyville GPS-signaalista valitun NMEA-arvon jälleen, jotta käyttäjä voi nähdä GPS-signaalin olevan kunnossa.

Käyttäjän on suotavaa olla perehtynyt GPS-signaalin NMEA-protokollaan ja sen arvoihin. Tässä työssä käytetään NMEA GPGGA -protokollaa (luku 5.1: GPGGA). GPS-anturikin ilmoittaa toiminnan tilastansa punaisella vilkkuvalla led-valolla. Ohjelman suorituksen aikana käyttäjälle ilmoitetaan aina kuvanottohetkeltä senhetkiset GPS-tiedot ja kuvauskerta. Tämän lisäksi ohjelma ilmoittaa, kun se aloittaa kuvien ja GPS-tietojen lähettämisen valvontaohjelmalle. Keskeyttäkseen tai lopettaakseen ohjelman suorituksen käyttäjää ohjeistetaan kuvantamisen ja mittauksen aloituksessa kirjoittamaan joko  $t + enter$  keskeyttäkseen tai  $q + enter$  lopettaakseen ohjelman suorituksen. Keskeyttäessään ohjelman suorituksen käyttäjä jatkaa takaisin ohjelman aloitukseen, jossa hän voi aloittaa tai lopettaa ohjelman suorituksen. Jatkaessaan ohjelman suoritusta uudestaan ohjelma luo uuden kansion, johon se tallentaa kuvat ja mittaustiedot GPS- ja digitaaliselta anturilta. Missä vaiheessa tahansa käyttäjän päättäessä lopettaa ohjelman suorituksen tapahtuu hallintaohjelman loppualustus, prosessien lopetus ja viimeinen tulostus, jossa käyttäjälle kerrotaan ohjelman suorituksen päättymisestä.

#### 5.4 Digitaalisen anturin kytkentä

Viimeisimpänä työvaiheena insinööriyössä oli digitaalisen anturin Pololun Mini-MU-9 v3 kytkeminen sekä ohjelmoiminen Raspberry PI:lle ja pääohjelman kulkuun. Tässä vaiheessa työtä RPI:hin on liitetty kaikki tarvittavat komponentit (kuva 16). Digitaalinen mittausyksikkö on esitelty aiemmassa luvussa 4.4. Mini-MU-9 v3 mittaa kolmea erilaista parametria: magneettisuutta, kiihtyvyyttä ja kulmaa. Kulman mittaukseen anturi käyttää digitaalista gyroskooppia. Kaikkien anturin mittaamia arvoja luetaan I<sup>2</sup>C-rajapinnan kautta.



Kuva 16. Raspberry Pi:hin kytkettynä Microstack GPS ja Pololun MinIMU-9 v3 anturit.

MinIMU-9 v3:lla ei ollut saatavilla Raspberry Pi Python-ohjelmakielelle suunnattuja valmiita kirjastoja, joten digitaalisen anturin ohjelmointi tapahtui antureiden MinIMU-9 v3 sisältämien L3GD20H- ja LSM303D-piirien ominaisuuksien ja I<sup>2</sup>C-rekistereiden perusteella. L3GD20H on 3-akselisen digitaalisen gyroskoopin ja LSM303D on 3-akselisen kiihtyvyyden ja magneettisuusanturi. Valtaosa ajasta kuului opiskellessa rekistereiden alustukset ja oppiessa Raspberry Pi I<sup>2</sup>C-rajapinnan ohjelmakirjastojen käyttöä.

MinIMU-9 v3 -mittausyksikölle on kuitenkin saatavilla testausohjelma Raspberry Pi:lle; `minimu9-ahrs`-ohjelma tukee myös MinIMU-9:n aiempia versioita. Ohjelmalla pystytään visualisoimaan anturin asentoa ja myös tulostuttamaan antureiden raakadatatietoa. Tätä raakadatan tietoa verrattiin Pythonilla luotuun lukuohjelmaan, jotta varmistuttiin rekistereiden oikeanlaisesta alustuksesta ja luvun onnistumisesta (kuva 17). [30.]

	Magneettisuus			Kiihtyvyyys			Digitaalinen Gyro		
2209	525	-6827	-184	-1120	4091	50	1	13	
2209	525	-6827	-196	-1131	4092	50	1	13	
2221	519	-6824	-183	-1132	4084	50	1	13	
2221	519	-6824	-174	-1126	4090	50	1	14	
2221	519	-6824	-187	-1126	4109	50	1	14	
2221	519	-6824	-195	-1143	4103	50	1	14	

X Y Z

```

pi@raspberrypi ~ $ sudo python3 TestGyro.py
LSM303D detected successfully.
L3GD20H detected successfully.
Magnetic: 2217 532 -6836
Acceleration: -11 -4547 15481
Gyro: 51 1 14
(x, y, z)

```

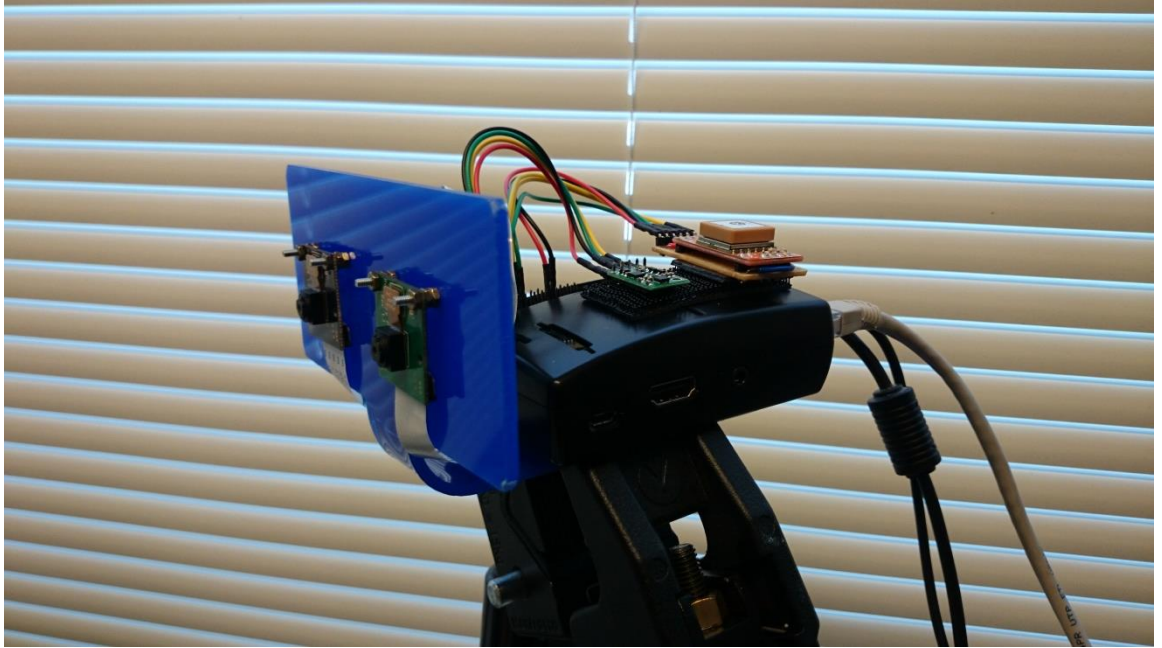
Kuva 17. MinIMU-9 v3 mittausdata x-, y- ja z-akselilta minu9-ahrs testiohjelmalla (ylempi) ja Pythonilla luodulla ohjelmalla (alempi).

Myöhemmässä vaiheessa tutkiessani enemmän MinIMU-9 v3 -anturin sovelluksia tutustuin Richards-Tech:n RTIMLib-ohjelmakirjastoon, jossa käytettiin MinIMU-9 v3 -anturia kulmien mittaamiseen. Ohjelman toteutus tapahtui Python 2.0 -ohjelmaversiolla, jota ei pystytty käyttämään tässä insinööriyössä GPS-anturin vaatimalla Python 3.0 -versiolla. Tutkittuani syytä tähän ongelmaan huomasin, että yksinkertainen ongelma oli Richards-Techin C++-ohjelmakielen kääntäjässä Python 3.0 -versiolle. Käyttöoikeuksien takia otin yhteyden sähköpostitse Richards-Techille ja pyysin heitä päivittämään RTIMULib-ohjelmakirjaston ohjelmakoodin kääntäjän Python 3.0 -versiolle mahdolliseksi. Myöhemmin sain käyttööni ja testaukseen RTIMULib -kirjaston Python 3.0 -versiolle. Tämä mahdollisti sen, että pystyin lukemaan MinIMU-9 v3 digitaalisen anturin absoluuttisia kulma-arvoja, joitten ansiosta anturin asentoa pystyy hahmottamaan helpommin kolmiulotteisesti. Nämä arvot kirjattiin lähetettävän paikkatiedon yhteyteen UDP:llä suoritettavaan lähetykseen ja lähetettiin valvontaohjelmalle käyttäjän näkyville.

[31]

## 6 RPI KUVANTAMIS- & MITTAUSJÄRJESTELMÄ

Kuvassa 18 voimme nähdä insinööriyön aikana kootun Raspberry Pi -pohjaisen kuvantamis- ja mittausjärjestelmän työpöytäversion prototyypikoteloinnin.



Kuva 18. RPI kuvantamis- ja mittausjärjestelmä prototyypikotelointi.

Versio on testattu ja kehitelty toimistotiloissa ja kannettavassa käytössä toimivaksi, kun Raspberry Pi:hin on liitetty akku ja nettiyhteys on varmistettu kuvantamis- ja mittausjärjestelmän sekä valvontaohjelman välillä. Järjestelmä pystyy kuvaamaan ympäristöään kahdella kameralla noin 400 ms viiveellä toisistaan ja kuvasarjan otto tapahtuu kahden sekunnin välein. Järjestelmään lisättiin ominaisuus, jonka avulla käyttäjä voi itse päättää kuvienottovälin pituuden (esillä visuaalisessa näkymässä kuvassa 15). Lisäksi järjestelmä tallentaa GPS-paikkatiedon, digitaalisen mittausyksikön kulmatiedot kolmelta akselilta sekä digitaalisen mittausyksikön kiihtyvyyden, magneettisuuden ja digitaalisen gyroskoopin arvon kolmelta akselilta. Digitaaliset anturin mittaustiedot tallentuvat omaan taulukkomuotoiseen tiedostoon ja digitaalisen anturin arvoista mitatut kulmarvot tallentuvat paikkatiedon kanssa omaan tiedostoon (taulukko 3).



Kuvassa 19 ja 20 on esitetty kuvantamis- ja mittausjärjestelmän kuvanottotuloksia. Kuvassa 19 on esillä kuvat tavalliselta RPI-kameralta ja NOIR-kameralta. NOIR-kameran kuvasta nähdään, että infrapunasuodattimen puuttumisen vuoksi NOIR-kameran kuva on lievästi ylivalottunut ja seepian värinen. Pimeällä kuvattaessa Infrapunavalonheittimen avulla kameraa voitaisiin käyttää pimeänäkökamerana.



Kuva 19. RPI kuvantamis- ja mittajärjestelmän kuvaotokset normaalilla RGB- ja NOIR-kameralla. ID 17. ja 18. taulukko 3.

Kuvassa 20 on esitetty kuvat normaalilta RPI-kameralta ja NOIR-kameralta. NOIR-kameran linssin eteen on kuitenkin asetettu sininen suodatinkalvo, joka nostaa esiin lehtivihreän määrän infrapuna-aallonpituudelta. NOIR-kameran kuvasta voidaan nähdä, että etualalla olevat männyt ja taka-alalla oleva metsä ovat hiukan kirkkaammin valottuneempia kuin muu kevättalvinen ympäristö. Tämä johtuu siitä, että männyn neulasissa on lehtivihreää, jota voidaan kuvasta havainnoida.



Kuva 20. RPI kuvantamis- ja mittajärjestelmän kuvaotokset normaalilla RGB- ja NOIR-kameralla lehtivihreän havaitsemiseksi. ID 21. ja 22. taulukko 3.

Taulukkoon 4 on poimittu esitettäväksi mittaustiedot kuvanottohetkeltä. Kuvien 17, 18, 21 ja 22 ottohetkeltä on nähtävillä paikkatiedot leveysaste (Latitude), pituusaste (Longitude) ja korkeus (Altitude). Lisäksi digitaaliselta anturilta mitatut kulmat ovat tuloksissa mukana. Kulmat on mitattu RPI kuvantamis- ja mittausjärjestelmän prototyypikoteloinnin päältä kuvan 18 mukaisesti. Kaiken lisäksi GPS signaalista on mittausdataan poimittu aikaleima (UTC) kuvanoton ja järjestelmän toimivuuden kuvantamiseksi. Taulukon 4 mittaustulokset ja paikkatietoarvot lähetetään vielä kuvien kanssa UDP-yhteydellä valvontaohjelmalle (kuva 14), jolla käyttäjä voi seurata kuvantamisen ja mittauksen suoritusta.

Taulukko 4. Mittaustulokset kuvanottohetkiltä kuvista 19 ja 20.

ID	UTC	Longitude	Latitude	Altitude	Gyro X	Gyro Y	Gyro Z
17	70055.0	27.77848	64.218085	130.5	8.63273	3.77766	76.00053
18	70056.0	27.77849	64.218083	130.5	6.96498	4.09375	81.60446
21	70058.0	27.77850	64.218087	130.5	6.04274	4.12519	79.78368
22	70059.0	27.77852	64.218088	130.5	5.62433	4.04085	81.93360

## 7 YHTEENVETO

Insinööriyössä suunniteltiin ja toteutettiin Raspberry Pi -pohjainen kuvantamis- ja mittausjärjestelmä RGB- ja IR-kameroiden lisäksi GPS-vastaanottimelle ja digitaaliselle anturille. Järjestelmän kehityksessä panostettiin luotettavasti toimivan ohjelman rakentamiseen. Tässä myös onnistuttiin ohjelman säikeistämisen avulla. Näin kuvantamis- ja mittausohjelma pystyi käsittelemään kuvanoton, kuvien lähettämisen, GPS-vastaanottimen luvun, paikkatiedon kirjaamisen, digitaalisen anturin luvun, mittaustietojen kirjaamisen, paikkatiedon ja asentoarvojen lähettämisen. Kaikki tämä toiminta pystyi tapahtumaan kahden sekunnin aikana.

Täysin ongelmattonta ei ohjelman tai komponenttien kehittäminen ole insinööri-työn aikana ollut. Suurin haaste oli kahden kameran saaminen Raspberry Pi -tietokoneeseen. IVPort Raspberry Pi -kameramoduulin multiplekserin piti vastata tähän haasteeseen, mutta odottamattomat virheet piirilevyn suunnittelussa tekivät työstä haastavaa. Tämän lisäksi monien ohjelmaosioiden toimiminen yhtenä ohjelmakokonaisuutena vaati lukuisia testauksia. Haasteista kuitenkin selvitettiin ja toimiva ohjelma luotiin kuvantamis- ja mittausjärjestelmälle.

Työn aikana perehdyttiin varsin hyvin Python-ohjelmointikieleen, Raspberry Pi -tietokoneeseen ja langattomaan tiedonsiirtoon UDP-protokollalla. Jatkokehityksessä tullaan parantamaan järjestelmän langatonta toimintaa.

Kuvantamis- ja mittausjärjestelmä saatiin valmiiksi siltä osin, mitä on laboratorio-oloissa mahdollista. Seuraavat testit suoritetaan järjestelmän koteloinnin valmistuttua ja säiden salliessa ulkoilmassa kentällä, Phantom 2 UAV -helikopterilla ja fyysisesti käsin.

## LÄHTEET

- (1) Raspberry Pi Wikipedia. 2014; Available at: [http://en.wikipedia.org/wiki/Raspberry\\_Pi](http://en.wikipedia.org/wiki/Raspberry_Pi). Accessed 10/15, 2014.
- (2) Raspberry Pi Homepage. What is Raspberry Pi. 2014; Available at: <http://www.raspberrypi.org/help/what-is-a-raspberry-pi/>. Accessed 10/20, 2014.
- (3) Sarja J. Raspberry Pi : käyttöönotto ja sovellustestaus. Oulu; 2013.
- (4) Kemppainen M. Graafisen etäkäyttöliittymän kehittäminen Raspberry Pi:lle. Kajaani: Kajaanin ammattikorkeakoulu; 2014.
- (5) Raspberry Pi Homepage. Raspberry PI model B. 2014; Available at: <http://www.raspberrypi.org/products/model-b/>. Accessed 09/29, 2014.
- (6) Raspberry Pi Homepage. Raspberry pi model A. 2014; Available at: <http://www.raspberrypi.org/products/model-a/>. Accessed 09/29, 2014.
- (7) Raspberry Pi Homepage. Raspberry Pi model B+. 2014; Available at: <http://www.raspberrypi.org/products/model-b-plus/>. Accessed 10/20, 2014.
- (8) Raspberry Pi Homepage. Raspberry Pi model A+. 2014; Available at: <http://www.raspberrypi.org/products/model-a-plus/>. Accessed 11/28, 2014.
- (9) Raspberry Pi Homepage. RPI GPIO. 2014; Available at: <http://www.raspberrypi.org/documentation/usage/gpio/>. Accessed 12/15, 2014.
- (10) Raspberry Pi Homepage. Introducing Raspberry Pi Hats. 2014; Available at: <http://www.raspberrypi.org/introducing-raspberry-pi-hats/>. Accessed 01/06, 2015.
- (11) Wikipedia EEPROM. 2013; Available at: <http://fi.wikipedia.org/wiki/EEPROM>. Accessed 12/17, 2014.
- (12) GitHub. Raspberry Pi HAT. 2014; Available at: <https://github.com/raspberrypi/hats>. Accessed 12/16, 2014.
- (13) Raspberry Pi Homepage. Raspberry Pi Camera-module. 2015; Available at: <http://www.raspberrypi.org/products/camera-module/>. Accessed 01/26, 2015.
- (14) Raspberry Pi Homepage. Raspberry Pi Noir-camera. 2015; Available at: <http://www.raspberrypi.org/products/pi-noir-camera/>. Accessed 01/26, 2015.
- (15) DesignSpark Homepage. Raspberry Pi model B+ GPIO. 2014; Available at: <http://www.rs-online.com/designspark/electronics/eng/blog/introducing-the-raspberry-pi-b-plus>. Accessed 12/16, 2014.

- (16) Raspberry Pi Homepage. RPI OS. 2014; Available at: <http://www.raspberrypi.org/downloads/>. Accessed 09/29, 2014.
- (17) Phantom 2 Homepage. Phantom 2 feature. 2015; Available at: <http://www.dji.com/product/phantom-2/feature>. Accessed 03/04, 2015.
- (18) Phantom 2 Homepage. Phantom 2 spec 2015; Available at: <http://www.dji.com/product/phantom-2/spec>. Accessed 03/04, 2015.
- (19) IVMECH Homepage. IVPort Raspberry Pi Camera Module Multiplexer. 2014; Available at: <http://www.ivmech.com/magaza/en/ivmech-m-2/ivport-raspberry-pi-camera-module-multiplexer-p-90>. Accessed 11/02, 2014.
- (20) Wikipedia GPS. 2014; Available at: <http://fi.wikipedia.org/wiki/GPS>. Accessed 12/28, 2014.
- (21) Microstack Homepage. Microstack GPS anturi/vastaanottimen kotisivu. 2014; Available at: <http://www.microstack.org.uk/products/microstack-gps/>. Accessed 10/25, 2014.
- (22) Microstack GPS GitHub. Microstack GPS vastaanottimen python 3.0 ohjelmakirjastot ja ominaisuudet. 2014; Available at: <https://github.com/microstack-iot/python3-microstacknode>. Accessed 11/05, 2014.
- (23) Microstack GPS. Tekniset tiedot ja ominaisuudet. 2014; Available at: <http://www.microstack.org.uk/assets/gps/FormattedGPSgettingstarted.pdf>. Accessed 11/26, 2014.
- (24) Pololu Homepage. Pololu MinIMU-9 v3 digitaalinen mittausyksikkö. 2015; Available at: <https://www.pololu.com/product/2468>. Accessed 01/15, 2015.
- (25) Wikipedia UDP. UDP protocol. 2015; Available at: <http://fi.wikipedia.org/wiki/UDP>. Accessed 02/15, 2015.
- (26) Python wiki. 2014; Available at: <https://wiki.python.org/moin/FrontPage>. Accessed 03/03, 2015.
- (27) Python Homepage. 2015; Available at: <https://www.python.org/>. Accessed 01/26, 2015.
- (28) GPS information. GPS NMEA Protocol. 2014; Available at: <http://www.gpsinformation.org/dale/nmea.htm>. Accessed 12/28, 2014.
- (29) IVMECH GitHub Homepage. IVMECH GitHub. 2014; Available at: <https://github.com/ivmech/ivport>. Accessed 12/18, 2014.
- (30) David Grayson GitHub. Pololu MinIMU-9 v3 testaus ohjelmakirjasto. 2015; Available at: <https://github.com/DavidEGrayson/minimu9-ahrs/wiki>. Accessed 01/23, 2015.

(31) Richards-Tech GitHub. Richards-Tech RTIMULib ohjelmakirjasto Pololu MinIMU-9 v3 ohjelmointi. 2015; Available at: <https://github.com/richards-tech/RTIMULib>. Accessed 02/05, 2015.

LIITTEET

RPI kuvantamis- & mittausjärjestelmän ohjelman vuokaavio

