

Emil Rekola

# MOOTTORIPYÖRÄN GPS-SEURANTAJÄRJESTELMÄ

Informaatioteknologian ja viestinnän tiedekunta  
Kandidaatintyö  
Toukokuu 2019

# TIIVISTELMÄ

Kandidaatintyö  
Tampereen yliopisto  
Tieto- ja sähkötekniikan kandidaatin tutkinto-ohjelma, Sähkötekniikka  
Toukokuu 2019  
Emil Rekola: Moottoripyörän GPS-seurantajärjestelmä

---

Tässä kandidaatintyössä tehtiin GPS-seurantajärjestelmä ajoneuvoon. Järjestelmä on hyödyllinen työkalu, jolla voidaan kasvattaa mahdollisuutta saada ajoneuvo takaisin varkauden tapahtuessa. Järjestelmästä yritettiin tehdä suhteellisen pieni kooltaan, jotta se saataisiin mahdollisimman hyvin piiloon ja se ei olisi tiellä estämässä ajoneuvon normaaleja käyttöoperaatioita. Täten GPS-järjestelmä rakennettiin osittain valmiista komponenteista, kuten Arduino Uno -kehitysalustasta, step-down -regulaattorista ja operaatiovahvistimista. Näitä osia hyödyntämällä saavutettiin pienempi tilan käyttö verrattuna edeltä mainittujen komponenttien itse rakentamiseen, pois lukien Arduino, joka valittiin järjestelmän mahdollista laajentamista varten tulevaisuudessa.

Työ jakautuu rakenteeltaan kahteen erilliseen osuuteen: kirjallisuus- ja rakenteluosuuteen. Kirjallisuusosuudessa käsitellään GPS-järjestelmän eri komponenttien ja lohkojen toimintaa sekä mihin niitä käytetään. Koko järjestelmän toiminta esitellään sen jälkeen, kun on käyty järjestelmän eri osat läpi. Rakenteluosuudessa valmistetaan järjestelmä, kirjoitetaan järjestelmän suorittama ohjelma ja testataan sen toimintaa verraten sitä järjestelmälle asetettuihin tavoitteisiin. Testauksen jälkeen keskustellaan mahdollisista parannuskohteista ja mitä ei saatu aikaan kandidaatintyön puitteissa valmiiksi.

Työssä käytetty ajoneuvo, johon järjestelmä integroidaan, on moottoripyörä. Arduino Uno toimii järjestelmän ytimenä, joka tarkkailee ajoneuvon asentoa ja lähettää SMS-viestin ajoneuvon omistajalle, jos ajoneuvoon huomataan kohdistuvan luvaton liikettä. Esimerkiksi tilanteessa, jossa ajoneuvo yritetään varastaa, järjestelmä lähettää omistajalle 2,5 minuutin välein ajoneuvon sijaintitiedot tekstiviestillä. Järjestelmä ottaa käyttöjännitteensä moottoripyörän akusta, mutta akun kytkettyessä irti on varalla myös akkupankki.

Rakennettu järjestelmä testattiin useissa eri skenaarioissa, kuten simuloitu kaatuminen ja ajoneuvon varkaus, joista järjestelmä suoriutui odotetusti. Läpäistyjen testien pohjalta voidaan todeta järjestelmän rakentamisen olleen onnistunut ja sen täyttävän kaikki sille asetetut vaatimukset. Täten voidaan todeta järjestelmän toteutuksen kokonaisuudessaan olleen onnistunut.

Avainsanat: Sulautettu järjestelmä, GPS, Arduino, SMS, Turvallisuus, Moottoripyörä, Elektroniikka

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck -ohjelmalla.

## ALKUSANAT

Idean kandidaatintyön aiheeseen sain ystäväni kanssa pohtiessamme, millaisen rakentamiseen pohjautuvan kandidaatintyön kumpikin tekisimme, jolloin hän ehdotti GPS-jäljityslaitteen rakentamista moottoripyörääni. Tämä idea mahdollisti kahden minulle mielenkiintoisen alueen yhdistämisen nostaan työn merkitystä ja kiinnostusta omassa mielessäni entistä korkeammalle.

Tahdon kiittää kandidaatintyöni ohjaajaa Erja Sipilää työhön liittyvistä ohjeista ja ohjauksesta, ystäväni Roope Keskistä loistavasta kandidaatintyön ideasta ja tyttöystäväni Anna Kleemolaa kieliopin tarkastamisesta.

Tampereella, 10.5.2019

Emil Rekola

# SISÄLLYSLUETTELO

1. JOHDANTO .....	1
2. GPS-JÄRJESTELMÄN TOIMINNAN TEORIA.....	3
2.1 Yleiskuva järjestelmän toiminnasta .....	3
2.2 Mikrokontrolleri.....	4
2.3 Arduino-kehitysalusta .....	7
2.4 SIM5320E-moduuli .....	8
2.5 Step-down -regulaattori.....	9
2.6 Akkupankki .....	11
2.7 Pietsosähköinen värähtelijä.....	12
2.8 Operaatiovahvistin .....	12
2.9 GPS-järjestelmän toiminta.....	13
3. JÄRJESTELMÄN RAKENTAMINEN .....	16
3.1 Järjestelmän IC-komponenttien valinta .....	16
3.2 Prototyyppi.....	18
3.3 Kasausprosessi.....	21
3.4 Ohjelmakoodin kirjoittaminen .....	23
4. JÄRJESTELMÄN TESTAUS.....	26
4.1 Ulkoinen testaus .....	26
4.2 Järjestelmän asennus ja testaus .....	30
4.3 Kehityskohteet .....	34
5. YHTEENVETO.....	35
LÄHTEET .....	37
LIITE A: OHJELMAKOODI .....	40

## KUVALUETTELO

<b>Kuva 1.</b>	<i>GPS-seurantalaitteen lohkokaavio. ....</i>	<i>4</i>
<b>Kuva 2.</b>	<i>8-bittisen mikrokontrollerin lohkokaavio [6, s.15]. ....</i>	<i>5</i>
<b>Kuva 3.</b>	<i>Pelkistetty piirros step-down -regulaattorista on- ja off-tiloissa. ....</i>	<i>10</i>
<b>Kuva 4.</b>	<i>Step-down -regulaattoriipiirit [19]. ....</i>	<i>11</i>
<b>Kuva 5.</b>	<i>Järjestelmän piirikaavio. ....</i>	<i>14</i>
<b>Kuva 6.</b>	<i>Differentiaalivahvistinkytkentä. ....</i>	<i>17</i>
<b>Kuva 7.</b>	<i>Komparaattorikytkentä. ....</i>	<i>18</i>
<b>Kuva 8.</b>	<i>GPS-seurantajärjestelmän prototyyppi. ....</i>	<i>19</i>
<b>Kuva 9.</b>	<i>Saapunut testiviesti. ....</i>	<i>21</i>
<b>Kuva 10.</b>	<i>Suurimmalta osin valmiiksi juotetut erilliset prototyyppilevyt. ....</i>	<i>22</i>
<b>Kuva 11.</b>	<i>Päällekkäin asetellut prototyyppilevyt. ....</i>	<i>23</i>
<b>Kuva 12.</b>	<i>Ohjelman toiminnan vuokaavio. ....</i>	<i>24</i>
<b>Kuva 13.</b>	<i>VN2106 MOSFET-transistorin virtaraja. [29] ....</i>	<i>27</i>
<b>Kuva 14.</b>	<i>Saadut sijaintitiedot. ....</i>	<i>28</i>
<b>Kuva 15.</b>	<i>Järjestelmän ilmoittama sijainti verrattuna puhelimen sijaintiin. ....</i>	<i>29</i>
<b>Kuva 16.</b>	<i>GPS-antennin kiinnitys sijainti. ....</i>	<i>30</i>
<b>Kuva 17.</b>	<i>GPS-järjestelmä kiinnitettynä väliaikaisesti moottoripyörään. ....</i>	<i>31</i>
<b>Kuva 18.</b>	<i>GPS-järjestelmällä kartoitettu ajoreitti. ....</i>	<i>33</i>
<b>Kuva 19.</b>	<i>GPS-järjestelmän suojakotelo. ....</i>	<i>34</i>

## LYHENTEET JA MERKINNÄT

AC	engl. Alternating Current, vaihtovirta.
ADC	engl. Analog-to-Digital Converter, analogi-digitaalimuunnin.
CPU	engl. Central Processing Unit, prosessori.
DAC	engl. Digital-to-Analog Converter, digitaali-analogimuunnin.
DIP	engl. Dual In-line Package, kaksoisrivipakkaus.
DRAM	engl. Dynamic Random-Access Memory, dynaaminen luku- ja kirjoitusmuisti.
EEPROM	engl. Electrically Erasable Programmable Read-Only Memory, sähköisesti tyhjennettävä ja uudelleenohjelmoitava muisti.
GPRS	engl. General Packet Radio Service, yleinen radiopakettipalvelu.
GPS	engl. Global Positioning System, globaali paikallistamisjärjestelmä.
GSM	engl. Global System for Mobile Communications, maailmanlaajuinen mobiilikommunikaatiojärjestelmä.
I/O	engl. Input/Output, sisäänmeno/ulostulo.
IC	engl. Integrated Circuit, mikropiiri.
IDE	engl. Integrated Development Environment, ohjelmointiympäristö.
MOSFET	engl. Metal-Oxide-Semiconductor Field-Effect Transistor, metallioksidi-puolijohdekanavatransistori.
PCB	engl. Printed Circuit Board, piirilevy.
PWM	engl. Pulse-Width modulation, pulssinleveysmodulaatio.
SMS	engl. Short Message Service, tekstiviesti.
SPDT	engl. Single Pole Double Throw, kaksi sisäänmenoa, yksi ulostulo.
SRAM	engl. Static Random-Access Memory, staattinen luku- ja kirjoitusmuisti.
USART	engl. Universal Synchronous/Asynchronous Receiver-Transmitter, universaali synkroninen/asynkroninen sarjaliikenteen vastaanotto- ja lähetyspiiri.
USB	engl. Universal Serial Bus, universaali sarjaväylä.

# 1. JOHDANTO

Nykypäivänä elektroniikkaa ja varsinkin älykkäitä järjestelmiä löytää melkein kaikkialta, paikoista missä sitä ei osattu edes kuvitella muutama vuosikymmen takaperin. Eikä pelkästään elektroniikan yleistyminen ole hämmästyttävää, vaan myös teknologian kehittymisen mahdollistama transistorien kutistaminen nanometrien mittakaavaan. Gordon Moore teki tunnetun ennustuksen vuonna 1965, että noin kahden vuoden välein transistorien määrä kaksinkertaistuu käytettyyn pinta-alaan nähden [1], mutta mitä luultavimmin edes Moore ei osannut ennustaa millaiseen maailmaan teknologian nopea kehittyminen johtaisi. Kaikki tämä on mahdollistanut pienien, jopa sormenpäätä pienempien, energiatehokkaiden tietokoneiden leviämisen kaikkialle teollisuuden robottien ohjauksesta elektroniikkaharrastajien omiin projekteihin, kuten myös tähän kandidaatintyöhön.

Mikrokontrollerit ovat näitä edellä mainittuja todella pieniä ja energiatehokkaita tietokoneita, jotka pitävät kaiken muistista ja suorittimesta lähtien sisällään, vieden vain pienen alueen piirilevyiltä [2]. Modernien mikrokontrollerien erittäin pieni koko on myös eduksi varsinkin turvallisuuden kannalta, mahdollistaen älykkäiden, turvallisuutta lisäävien ominaisuuksien lisäämisen ns. ”tyhmiin” tavaroihin, kuten esimerkiksi mobiililaitteella ohjattavan kauko-ohjausjärjestelmän integroimisen oviin yritysten ja yksittäisten henkilöiden toimesta [3].

Tässä kandidaatintyössä on tavoitteena suunnitella ja toteuttaa mikrokontrollerin hallinnoima moottoripyörän GPS (engl. Global Positioning System, maailmanlaajuinen paikallistamisjärjestelmä) -seurantajärjestelmä, joka pystytään rakentamaan pieneen tilaan. Järjestelmän pieni koko on tärkeä, sillä moottoripyörässä ei ole paljon ylimääräistä tilaa. Varastettujen moottoripyörien jäljittäminen ja takaisin saaminen on vaikeampaa verrattuna autoihin, joten tällä järjestelmällä saadaan aikaan omistajalle huomattavasti parempi mielenrauha. Järjestelmän ollessa keskittynyt mikrokontrollerin ympärille, on se myös helposti laajennettavissa tulevaisuudessa esimerkiksi sammuttamaan moottorin kolarin tapahtuessa.

Työ jakaantuu kahteen osaan: teoriaosuuteen, jossa ensin käydään läpi järjestelmän eri osien teoriaa ja sen jälkeen käsitellään järjestelmän toteutus, sekä työn rakentamiseen. Ensin luvussa 2 käydään läpi järjestelmän toimintaa teoreettisella tasolla. Tämän jälkeen luvussa 3 perehdytään järjestelmän rakentamiseen sekä järjestelmän asennukseen

moottoripyörän omaan sähköjärjestelmään. Järjestelmän rakentamisen selostuksen yhteydessä käydään myös läpi mikrokontrollerille kirjoitettua koodia, jolla järjestelmää hallitaan. Luku 4 käsittelee järjestelmän toimivuuden testaamista ja mahdollisia parannuskohteita. Luku 5 pitää sisällään yhteenvedon kandidaatintyöstä.



## 2. GPS-JÄRJESTELMÄN TOIMINNAN TEORIA

Työssä on tavoitteena suunnitella ja rakentaa moottoripyörään GPS-seurantajärjestelmä, jota on mahdollisuus kehittää haluttaessa pidemmälle monipuolisen mikrokontrollerin ympärille. Rakennetun järjestelmän pieni koko on tärkeää, sillä se mahdollistaa helpomman piilotuksen moottoripyörän rakenteiden lomaan. Mikrokontrollerin sisältämä Arduino Uno -kehitysalusta saataisiin viemään vähemmän tilaa, jos koko PCB (engl. printed circuit board, piirilevy) suunniteltaisiin itse. Liiallinen tilan minimoiminen voisi kuitenkin heikentää järjestelmän laajentamisen mahdollisuuksia tulevaisuudessa, joten työssä päätettiin käyttää valmista kehitysalustaa.

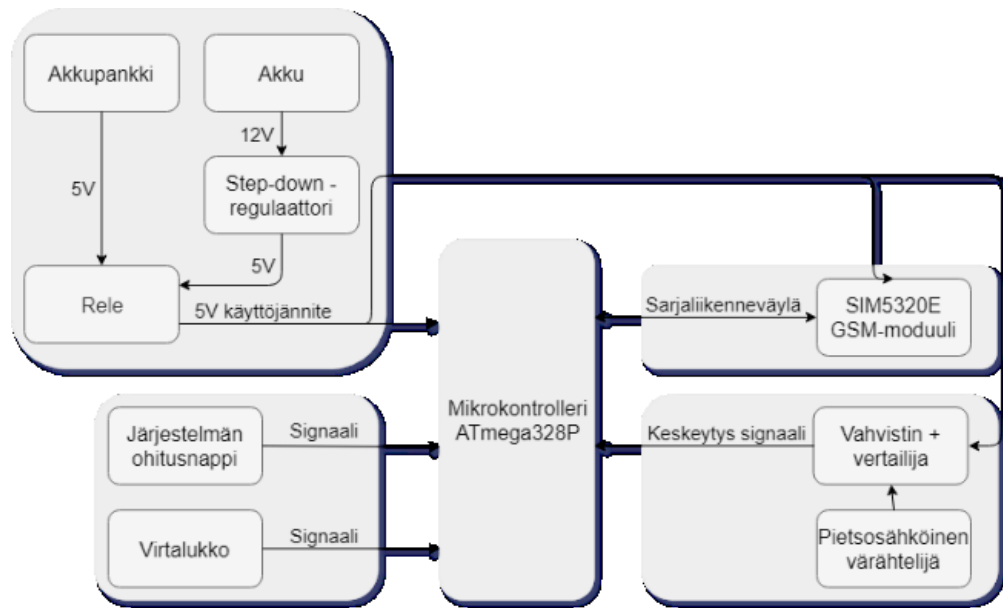
Työn järjestelmällä on neljä tärkeää tavoitetta, jotka takaavat järjestelmälle suunnitellun toimivuuden. Nämä tavoitteet ovat:

- Seurata moottoripyörää luvattoman liikkeen varalta
- Pystyä lähettämään pyörän sen hetkinen sijainti omistajalle tarpeeksi hyvällä tarkkuudella
- Toimia tarpeeksi pitkään, kun moottoripyörästä irrotetaan akku
- Pystyä keskeyttämään asennon tarkkailu mahdollisia huoltotoimenpiteitä tai muita tilanteita varten.

Järjestelmä rakennettiin näiden neljän tavoitteen ympärille ja jos yksikin näistä tavoitteista ei täyty, järjestelmää ei voida laskea onnistuneeksi.

### 2.1 Yleiskuva järjestelmän toiminnasta

Työssä toteutettava fyysinen osa järjestelmästä voidaan jakaa viiteen toiminnalliseen lohkokon: käyttöjännitteen regulointi ja valinta, järjestelmän toiminnan ohitus, mikrokontrolleri, GSM (engl. Global System for Mobile Communications, maailmanlaajuinen mobiilikommunikaatiojärjestelmä) -moduuli ja liikkeen havainnointi. Järjestelmän lohkokaavio on esitettyä kuvassa 1.

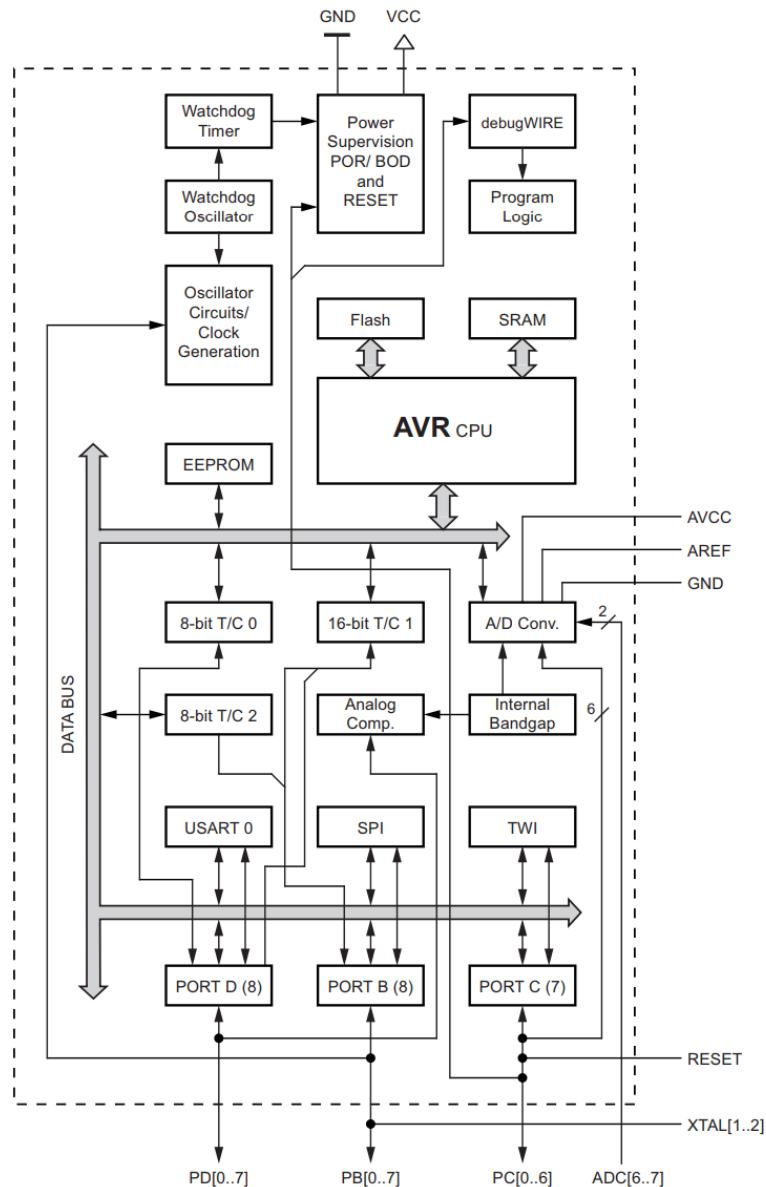


**Kuva 1.** GPS-seurantalaitteen lohkokaavio.

Järjestelmän tarvitsema käyttöjännite saadaan ensisijaisesti moottoripyörän akusta ja toissijaisena käyttöjännitteenä on varalla akkupankki, joka kytketään kiinni automaattisesti järjestelmään relettä käyttäen. Moottoripyörän akun tuottama 12 V jännite lasketaan järjestelmälle sopivalle 5 V jännitetasolle step-down -regulaattorilla. Järjestelmän toimintaa ohjaa mikrokontrolleri, joka alkaa paikantamaan moottoripyörän sijaintia ja lähettää sijainnin moottoripyörän omistajalle, jos pietsosähköinen värähtelijä havaitsee liikettä. Järjestelmän paikannusominaisuus ja viestin lähettäminen toteutuvat GSM-moduulia hyödyntämällä, jonka kanssa mikrokontrolleri kommunikoi USART (engl. Universal Synchronous/Asynchronous Receiver-Transmitter, universaali synkroninen/asynkroninen sarjaliikenteen vastaanotto- ja lähetyspiiri) -sarjaliikenneväylää käyttämällä. Järjestelmän eri lohkojen ja lohkojen osien toimintaa käsitellään seuraavissa luvuissa.

## 2.2 Mikrokontrolleri

Mikrokontrolleri on pieni mikropiiri, jonka keskeisiä komponentteja ovat prosessori, erinäiset muistit ja I/O-väylät. Mikroprosessori eroaa mikrokontrollerista siten, että mikroprosessori on oma yksinäinen piirinsä, eli siihen ei ole integroitu käyttömuistia tai muita oheislaitteita vaan ne pitää yhdistää ulkoisesti. Mikrokontrolleri määritellään mikropiiriksi, johon nämä kaikki eri lohkot on integroitu. [4][5] Todellisuudessa 8-bittinen AVR-mikrokontrolleri voi olla erittäin monimutkainen järjestelmä; esimerkkinä kuvassa 2 nähdään 8-bittisen AVR-mikrokontrolleriperheen lohkokaavio, johon GPS-järjestelmässä käytetty ATmega328P kuuluu. Seuraavaksi käydään läpi työn toiminnalle keskeisimpien lohkojen toiminnan perusteet.



**Kuva 2.** 8-bittisen mikrokontrollerin lohkokaavio [6, s. 15].

CPU (engl. Central Processing Unit, prosessori) suorittaa sille annetun konekielelle käännetyn ohjelman [2]. Konekielisen ohjelman käskyt suoritetaan käyttäen aritmeettisia-, loogisia- ja I/O (engl. Input/Output, sisäänmeno/ulostulo) -operaatioita. Pohjimmiltaan prosessori prosessoi binäärilukujonoja, esimerkiksi muuttaen ohjelman antamien käskyjen mukaan binäärilukujonosta yhden bitin arvosta 0 arvoon 1.

SRAM (engl. Static Random-Access Memory, staattinen luku- ja kirjoitusmuisti) on staattinen muistityyppi, jonka muistisolut eivät vaadi perinteisen DRAM (engl. Dynamic Random-Access Memory, dynaaminen luku- ja kirjoitusmuisti) -muistin tavoin virkistysyhtiä datan säilyttämiseksi soluissa. SRAM-muistisolu koostuu muutamalla transistorilla rakennetusta kiikkupiiristä, jonka ulostulo esittää muistiin asetetun bitin arvoa. SRAM on "haihtuvaa muistia", joka menettää pitämänsä datan ilman

käyttöjännitettä. SRAM on kuitenkin kallista valmistaa ja vie paljon tilaa, joten yleisesti sen kapasiteetti on pieni ja sitä käytetään yleensä välimuistina, johon asetetaan ohjelman suorittamiseen tarvittavat muuttujat. [7]

Flash-muisti on sähköisesti uudelleenohjelmoitava muistityyppi, joka koostuu transistoreilla toteutetuista muistisolusta. Flash-muisti säilyttää datansa, vaikka käyttöjännite katkaistaan, eli flash-muisti on haihtumaton muistityyppi, toisin kuin SRAM- ja DRAM-muistit. ATmega328P sisältää kahdenlaista flash-muistia: käyttömuistin ja EEPROM (engl. Electrically Erasable Programmable Read-Only Memory, Sähköisesti tyhjennettävä ja uudelleenohjelmoitava muisti) -muistin. Käyttömuisti ja EEPROM eroavat toisistaan muistisolujen arkkitehtuurissa siten, että käyttömuisti käyttää NAND-muistisoluja ja EEPROM NOR-muistisoluja. Käyttömuisti on hitaampaa, mutta halvempaa valmistaa verrattuna EEPROM-muistiin, joten käyttömuistia on yleisesti enemmän ja sitä käytetään useimmiten ohjelmien tallentamiseen ja lukemiseen. EEPROM on kalliimpaa valmistaa, joten EEPROM-muistia on yleisesti vähemmän mikrokontrollerilla käytettävänä, mutta sitä pystytään lukemaan ja uudelleenkirjoittamaan yksi tavu kerrallaan. Tämä on joissain tapauksissa hyödyllisempää verrattuna käyttömuistiin, jota pitää lukea sivu (engl. page) kerrallaan. Yksi tällainen sivu voi tyypillisesti koostua 512, 2048 tai 4096 tavusta. Tämän takia esimerkiksi asetustiedot ja muut parametrit voidaan asettaa EEPROM-muistiin, josta ne voidaan lukea ja kirjoittaa tavu kerrallaan. [7][8]

USART on yksi monesta sarjaliikenneprotokollasta. USART-lohkoja voi esiintyä useampi kappale mikrokontrollerin sisällä, mutta tässä työssä käytetyssä mikrokontrollerissa on vain yksi tällainen lohko. USART toimii käyttäen joko kahta tai kolmea johtoa, riippuen onko käytössä synkroninen vai asynkroninen yhteys. Synkronisessa yhteydessä yksi johdin on datan lähetystä varten, toinen johdin datan vastaanottamista varten ja kolmas laitteiden synkronointiin tarvittavalle kellosignaaliille. Asynkroninen yhteys ei tarvitse kolmatta kellosignaaliille tarkoitettua johdinta, mutta laitteiden täytyy sopia yhteinen datan siirtonopeus. USART on suosittu sarjaliikenneväylä sen helppokäyttöisyyden vuoksi, mutta väylän huonoina puolina on muun muassa mahdollisuus yhdistää vain yksi pari laitteita toisiinsa. [9, s. 67, s. 81] Työssä käytetään asynkronista UART-sarjaliikenneväylää, jolloin kommunikointiin GSM-moduulin ja mikrokontrollerin välillä tarvitaan vain kaksi johtoa verrattuna synkronisen tarvitsemaan kolmeen johtoon.

Mikrokontrolleri on yhteydessä ympäristöönsä käyttäen I/O-portteja, jotka ovat merkittynä kuvassa 2 nimellä PORT. I/O-portit voidaan jakaa kahteen kategoriaan: digitaalisiin ja analogisiin. Digitaaliset I/O-portit voidaan nimensä mukaisesti asettaa joko ulostuloksi tai sisäänmenoksi. Asetettaessa portti ulostuloksi, portin impedanssi on

matala ja se voidaan asettaa korkeaan ulostulotilaan eli käyttöjännitteeseen tai matalaan ulostulotilaan eli maahan. Matala impedanssi mahdollistaa ulkoisten laitteiden ohjauksen, sillä portti ja siihen kytketty ulkoinen laite muodostavat jännitteenjakajan. Portin pieni impedanssi johtaa kuitenkin siihen, että portin yli syntyy vain pieni jännitehäviö eikä se rajoita virtaa. Kun portti asetetaan sisäänmenoksi, kytkeytyy portti korkea-impedanssi -tilaan vähentämään siihen kytketyn laitteen kuormitusta. Portti pystyy lukemaan siihen tulevan jännitteen joko ykkösenä tai nollana, jolloin se voidaan raportoida mikrokontrollerille yhtä bittiä käyttäen. Analogisilla porteilla on samat ominaisuudet kuin digitaalisilla, mutta asetettuna sisääntuloiksi ne pystyvät raportoimaan siihen tulevan jännitteen käyttäen ADC:tä (engl. Analog-to-Digital Converter, analogi-digitaalimuunnin). ADC:n eli AD-muuntimen resoluutio määrää, kuinka tarkasti siihen tuleva jännite voidaan lukea. Järjestelmässä käytetyssä mikrokontrollerissa on 10-bittinen AD-muunnin, jolla voidaan erottaa  $2^{10}$  eli 1024 eri jännitetasoa [6], mutta tehokkaammissa mikrokontrollereissa voi olla 12-bittinen AD-muunnin eli voidaan erottaa 4096 eri jännitetasoa. Analogisen portin ulostulo eroaa myös hieman digitaalisen portin toiminnasta. Analogisen portin ulostulo voidaan asettaa tuottamaan ulostulojännite käyttöjännitteen ja maan väliltä käyttämällä DAC:ta (engl. Digital-to-Analog Converter, digitaal-analogimuunnin), jonka resoluutio määrää ulostulojännitteen tarkkuuden. Järjestelmän sisältämän mikrokontrollerin DAC:n eli DA-muuntimen resoluutio on 8-bittiä, joka tarkoittaa että voidaan tuottaa 256 eri jännitetasoa. Eri jännitteiden tuottaminen onnistuu syöttämällä PWM (engl. Pulse-Width modulation, pulssinleveysmodulaatio) -signaalin ulkoiseen alipäästösuotimeen. Ulostuloon ajetaan siis kanttiaaltopulssi, jonka leveyttä voidaan säätää järjestelmässä käytetyn mikrokontrollerin tapauksessa 256:en eri leveyteen 0 ja 100 %:n välillä [2][6], jota säätämällä saadaan alipäästösuotimen läpi eri tasoinen jännite.

### 2.3 Arduino-kehitysalusta

Työssä rakennetun järjestelmän keskeisimpänä elementtinä on Arduino Uno -kehitysalusta. Arduino syntyi tutkimusprojektin seurauksena 2000-luvun alussa Interaction Design Institute Ivreassa Pohjois-Italiassa. Vuonna 2005 ensimmäinen Arduino esiteltiin laitteena, joka tulee helpottamaan prototyyppien tekemistä. Arduino oli kohdistettu varsinkin opiskelijoille, joilla ei ollut aikaisempaa kokemusta elektroniikasta. [9, s. 2][10] Arduino-kehitysalustat ovat suurimmalta osin pohjautuneet Atmelin, nykyisin Microchip, valmistamiin monipuolisiin ja tehokkaisiin AVR-mikroprosessoreihin, mutta myös Intelin ja ARM prosessoreita on käytössä eri Arduino-kehitysalustoissa. [11]

Tässä työssä käytetään Arduino Uno -kehitysalustaa, joka on erittäin monipuolinen, helppokäyttöinen ja suosittu kehitysalusta. Työssä olisi voitu käyttää pienempää Arduino Nano -kehitysalustaa tai suunnitella itse piirilevy mikrokontrollerille omien tarpeiden mukaisesti, mutta tämä päätös voisi mahdollisesti rajoittaa järjestelmän laajentamista tulevaisuudessa. Arduino pohjautuu edellä mainittuun ATmega328P-mikroprosessoriin, jonka monipuoliset ominaisuudet mahdollistavat järjestelmän joustavan toteutuksen. Alla taulukossa 1 on esitettyä Arduino Uno -kehitysalustan tärkeimmät tiedot.

Taulukko 1. *Arduino Uno -kehitysalustan keskeiset tiedot [12].*

Ominaisuus	Arvo
Mikrokontrolleri	ATmega328P
Käyttöjännite	5 V
Digitaaliset I/O-pinnit	14 kpl
PWM	6 kpl
Flash-muisti	32 KB
SRAM-muisti	2 KB
EEPROM-muisti	1 KB
Kellotaajuus	16 MHz
Koko	68,6mm x 53,4mm

Arduino Unon ohjelmointi tapahtuu USB:n (engl. Universal Serial Bus, universaali sarjaliikenneväylä) välityksellä liittämällä Arduino tietokoneeseen USB-kaapelilla. Todellinen ohjelman kirjoitus tapahtuu Arduinon omassa IDE:ssä (engl. Integrated development environment, ohjelmointiympäristö), jossa voidaan muun muassa varmistaa kirjoitetun ohjelman oikea syntaksi ja tarkkailla sarjaliikennettä.

## 2.4 SIM5320E-moduuli

Järjestelmän GPS- ja SMS (engl. Short Message Service, lyhytviestipalvelu) -toiminnan takana on SIMcom:in valmistama SIM5320E-moduuli, jota ohjataan mikrokontrollerille kirjoitetulla ohjelmalla. SIM5320E-moduulin ominaisuuksia ovat tiedonsiirto GPRS (engl. General Packet Radio Service, yleinen radiopakettipalvelu) -verkossa, GPS-paikannus, SMS-toiminnallisuus sekä puheluiden soittaminen [13]. Työssä näistä ominaisuuksista

hyödynnetään GPS-paikannusta sekä SMS-toiminnallisuutta ajoneuvon paikantamiseen ja sijainnin, suunnan ja nopeuden lähettämiseen SMS-viestillä.

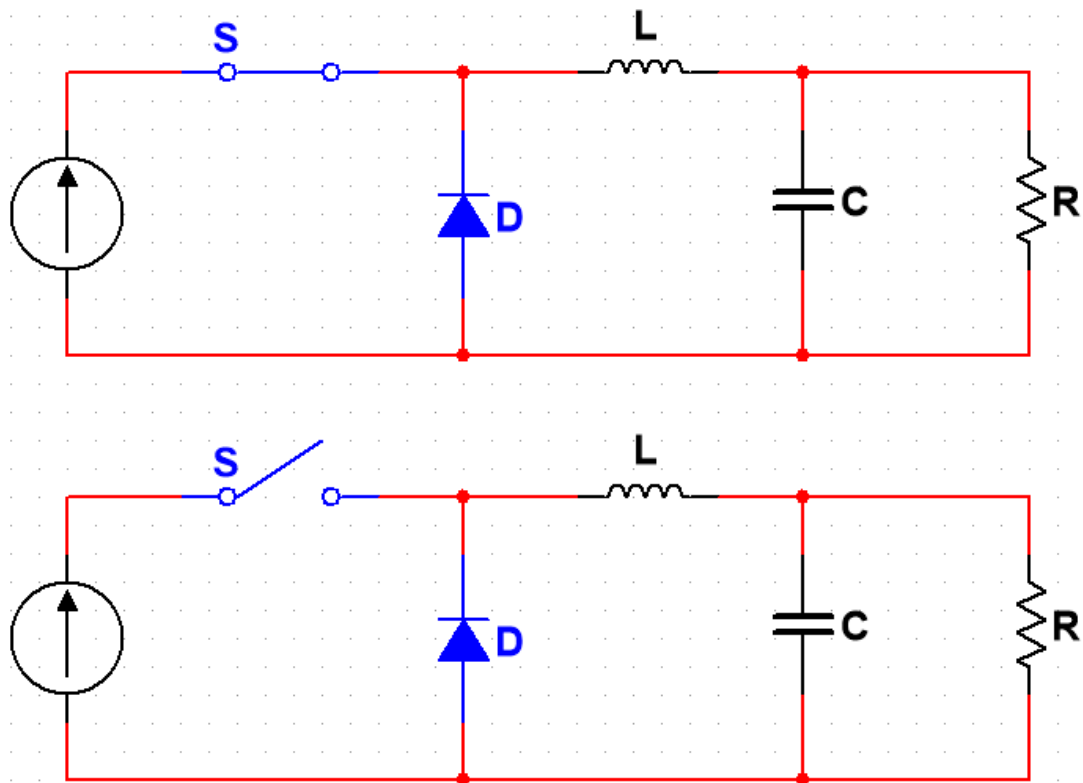
SIM5320E-moduulin toimintaa ohjataan mikrokontrollerin ohjelmaan koodatuilla AT-komennoilla [14]. Nämä AT-komennot koostuvat lyhyistä tekstijonoista, joita lähettämällä ohjataan moduulin eri ominaisuuksia, kuten SMS-viestin lähettämistä ja GPS-sijainnin hakemista.

Laitteen lähettäessä SMS-viestin toiseen SMS-viestejä tukevaan laitteeseen pitää lähettävän laitteen olla tukiaseman kantaman sisällä. Lähetetty viesti kulkeutuu lyhytviestikeskukseen, joka selvittää vastaanottajalaitteen sijainnin, jotta viesti voidaan ohjata oikealle tukiasemalle. Kun vastaanottaja löytyy tukiaseman kantamasta, viesti lähetetään tukiaseman kautta vastaanottajalle. Jos vastaanottaja ei ole minkään tukiaseman kantaman sisällä, säilyttää edellä mainittu keskus viestiä, kunnes vastaanottaja kytkeytyy taas tukiaseman verkkoon. SMS-viestit käyttävät omaa erillistä kanavaa verrattuna soittoihin, joten SMS-viestejä pystytään lähettämään soittoliikenneväylien ruuhkautumisesta huolimatta. [15]

GPS-paikannus perustuu halutun laitteen paikantamiseen käyttämällä maata kiertäviä satelliitteja. Paikantamiseen tarvittavien satelliittien määrä vaihtelee paikannettavan laitteen mukaan. Paikannettavan laitteen ollessa ilma-alus, kuten lentokone, tarvitaan pääasiassa neljä satelliittia seuraamaan nopeuden lisäksi x-, y- ja z-suuntaista liikettä ja ajoittain viides satelliitti tulee avustamaan, jos jokin näistä neljästä satelliitista on menettämässä näköyhteyden kohteeseen. Maa-ajoneuvojen, esimerkiksi auton, tilanteessa vaaditaan vain kolme satelliittia seuraamaan nopeuden lisäksi vain x- ja y-suuntaista liikettä, sillä ajoneuvon z-suuntainen liike suhteessa x- ja y-suuntaiseen liikkeeseen on niin pieni. 1900-luvun loppupuolella siviilikäyttöön tarkoitetuissa laitteissa paikannustarkkuus saattoi vaihdella jopa 100 m, kun taas nykyaikana päästään muutaman metrin päähän oikeasta sijainnista. [16][17]

## 2.5 Step-down -regulaattori

Järjestelmä on tarkoitettu kytkettäväksi ajoneuvon omaan sähköjärjestelmään, josta suurin osa toimii 12 V jännitteellä, seurantajärjestelmän käyttäessä 5 V käyttöjännitettä. Työssä käytetään step-down -regulaattoria, jolla saadaan laskettua ajoneuvon akun 12 V jännite mikrokontrollerille yhteensopivalle 5 V tasolle. Alla olevassa kuvassa 3 on piirretty erittäin alkeellinen step-down -regulaattori, jossa vastus R esittää piiriin kiinnitettyä kuormaa.



**Kuva 3.** Pelkistetty piirros step-down -regulaattorista on- ja off-tiloissa.

Regulaattorin ollessa on-tilassa eli kytkimen S ollessa kiinni, virta kulkee käämin L läpi, joka pyrkii vastustamaan tätä virtaa luomalla vastakkaisuuntaisen jännitteen. Tämä käämin L muodostama vastakkaisuuntaisen jännite pienentää lähteestä tulevaa jännitettä, laskien ulostulossa näkyvää jännitettä. Kytkimen S avautuessa regulaattori siirtyy off-tilaan ja käämi toimii virtalähteenä lyhyen ajan pitäen yllä kuorman yli olevaa jännitettä. Mikäli kytkin avataan käämin läpikulkevan virran muuttuessa, on ulostulossa aina pienempi jännite kuin sisääntulossa, mutta jos virta ei muutu, käämi ei synnytä vastakkaisuuntaista jännitettä. Kytkimen asennon muuttuessa tarpeeksi nopeasti saadaan ulostuloon lähes vakiojännite, ja muuttamalla kytkimen aukioloajan kesto voidaan muuttaa ulostulon jännitettä [18]. Piirissä oleva diodi D varmistaa halutun jännitteen kulkusuunnan kytkimen S eri asennoissa. Kondensaattori C on niin kutsuttu ulostulokondensaattori, jonka tarkoituksena on vähentää kytkimen asennon jatkuvasta vaihtelusta ulostulojännitteeseen syntyvää aaltomaista häiriötä, eli rippeliä.



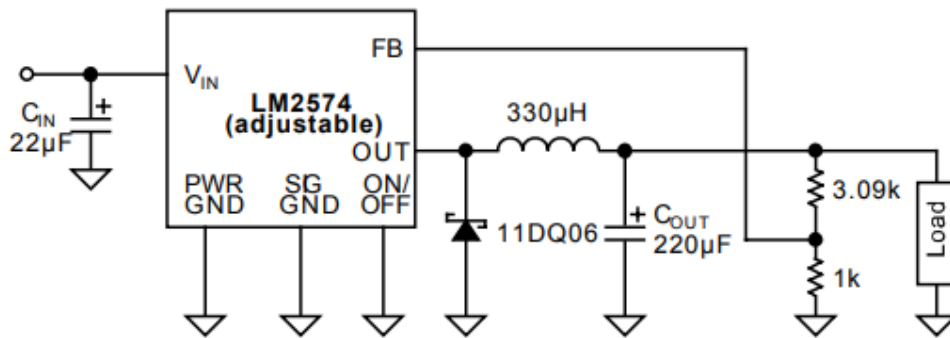


Figure 2. Adjustable Regulator Test Circuit

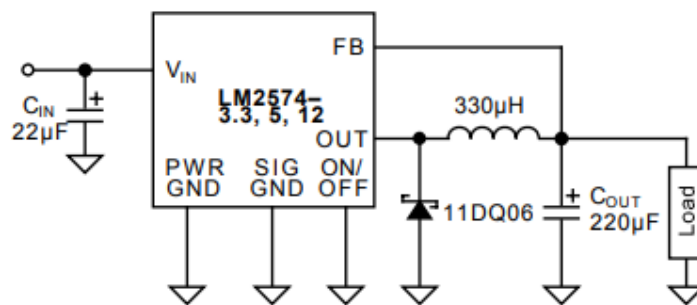


Figure 3. Fixed Regulator Test Circuit

Kuva 4. Step-down -regulaattoriipiirit [19].

Kuvassa 4 nähdään työssä käytetyn LM2574 step-down -regulaattorin tarvitsevan vain muutaman ulkoisen passiivisen komponentin toimiakseen. Käytetty regulaattori on erittäin monipuolinen, sillä sitä voidaan käyttää joko vakituksessa tilassa tai muuttuvassa tilassa. Vakituksessa tilassa regulaattorin ulostulojännite pysyy asetetussa arvossa kuormasta riippumatta, kun taas muuttuvassa tilassa ulostulojännitteen taso on säädettävissä sen ollessa toiminnassa.

## 2.6 Akkupankki

Järjestelmä ottaa pääasiassa tarvitsemansa käyttöjännitteen ajoneuvon akusta, mutta jos akun jännite kytkeytyy pois, on varalla akkupankki pitämässä järjestelmän käynnissä. Suurin osa akkupankeista koostuu joko litiumioni- tai litiumpolymeerisoluista, jotka ovat molemmat uudelleen ladattavia. Nämä solut toimivat normaalien paristojen tavoin pitäen sisällään tietyn varauksen, jota voidaan käyttää erinäisten laitteiden toimintaan. Solujen lataaminen tapahtuu kääntämällä akun purkautumisessa tapahtuva kemiallinen prosessi ympäri, jolloin saadaan varattua solut uudelleen. Solut kuitenkin kuluvat purkautumis-latautumis-sykleissä, mikä voidaan havaita solujen varauskapasiteetin vähentymisenä

elinkaaren aikana. [20] Tässä työssä akkupankki ei lataa itseään moottoripyörän omalla akulla, vaan se on varalla, jos moottoripyörän akun tuottama käyttöjännite kytketään pois.

Järjestelmän toiminnan vaatima 5 V käyttöjännite saadaan akkupankkiin integroidusta USB-liittimestä, jota käytetään muiden USB-laitteiden lataamiseen. USB-liittimen spesifikaatioissa on annettu  $\pm 5\%$  marginaalit reguloidulle 5 V ulostulolle, joka on täysin riittävä tälle järjestelmälle. Myös akkupankin antama virta on tarpeeksi suuri ajamaan kaikkia järjestelmän eri lohkoja. [21, s. 43]

Akkupankkiin olisi voitu haluttaessa suunnitella tai ostaa latauspiiri, jolla sitä olisi ladattu suoraan moottoripyörän akusta. Litiumioniakkujen itsepurkautumisajat ovat kuitenkin nykypäivänä erittäin matalat, joten moottoripyöräajokauden keston aikana akkupankki ei ehdi itsestään tyhjentyä normaaliolosuhteissa [22].

## 2.7 Pietsosähköinen värähtelijä

Pietsosähköinen värähtelijä tuottaa pienen AC (engl. Alternating Current, vaihtovirta) -jännitteen, kun sitä kuormitetaan mekaanisesti tietyssä suunnassa esimerkiksi puristamalla tai saamalla se värähtelemään. Lepotilassa värähtelijäelementin sisällä olevan kiteen positiivisten ja negatiivisten ionien muodostamat dipolit mitätöivät toisensa, jolloin ei synny sähkökenttää eikä jännitettä. Värähtelijäelementtiä rasitettaessa kide muuttaa muotoaan, mikä aiheuttaa dipolien epätasapainon ja sähkökentän, joka taas johtaa pieneen AC-jännitteeseen. Tätä tapahtumaa kutsutaan pietsosähköiseksi ilmiöksi. [23][24]

Pietsosähköistä värähtelijää käytetään työssä havaitsemaan tärähdykset, jotka voivat johtua moottoripyörään kohdistuvasta ilkeväkivallasta tai varkausyrityksestä. Värähtelijän tuottama jännitesignaali on matala, joten signaalin hyödyntämiseksi pitää signaali vahvistaa ulkoisella vahvistinpiirillä mikrokontrollerille sopivalle jännitetasolle. Signaalia käytetään synnyttämään keskeytyssignaali mikrokontrollerille, mikä herättää mikrokontrollerin unesta ja aloittaa ohjelman suorittamisen.

## 2.8 Operaatiovahvistin

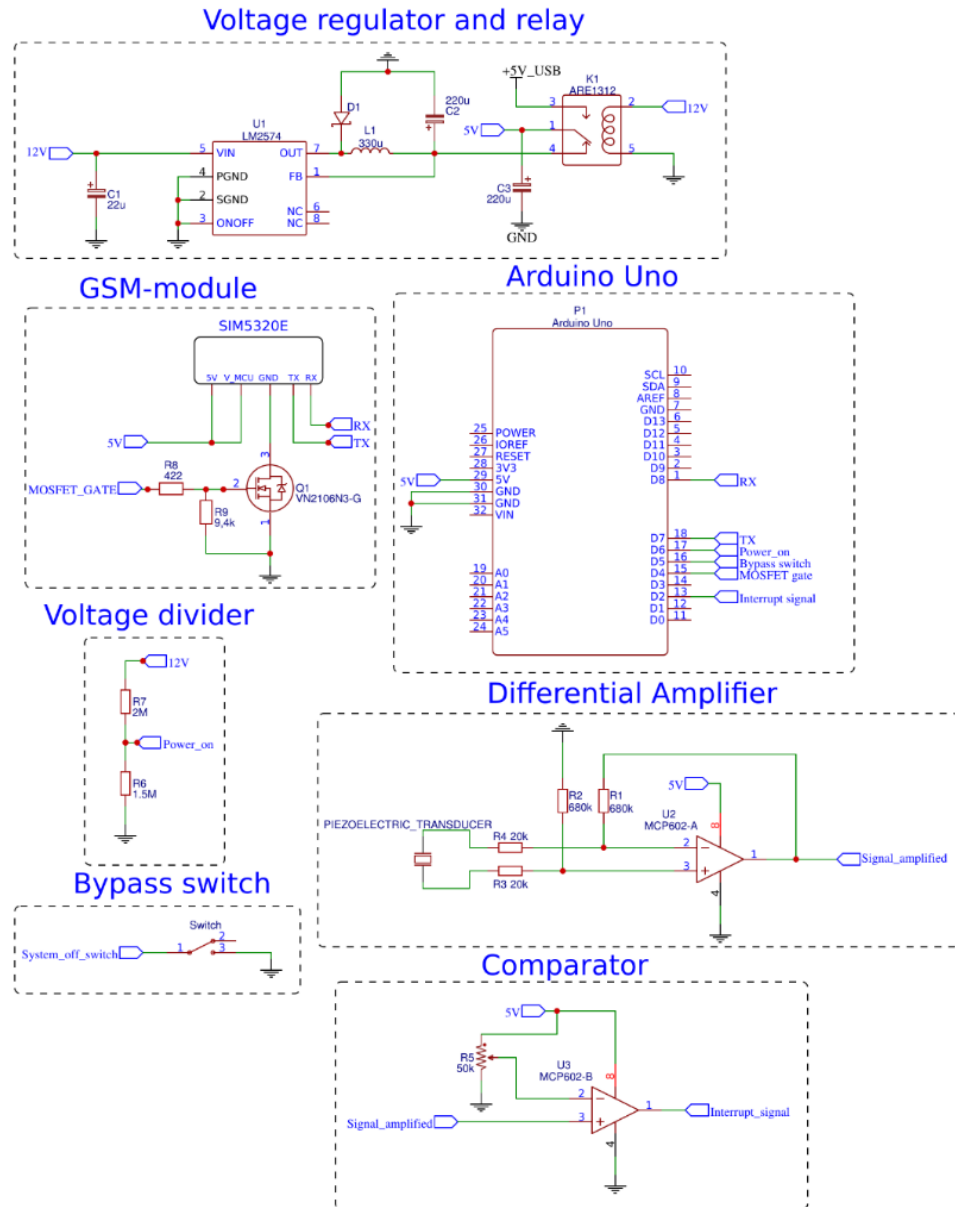
Operaatiovahvistimet ovat monipuolisia ja yleisiä komponentteja. Operaatiovahvistimen toiminta perustuu sen sisäänmenoihin tulevan jännitteen erotuksen vahvistamiseen. Riippuen siitä, miten operaatiovahvistimen sisällä oleva mikropiiri on toteutettu, voi vahvistimen avoimen silmukan vahvistuskerroin olla erittäin suuri; esimerkiksi työssä

käytetyn MCP602-operaatiovahvistimen avoimen silmukan vahvistuskertoimen on noin 110 dB eli jännitteen voi vahvistaa yli 300 000 kertaiseksi. [25]

Työssä hyödynnetään operaatiovahvistimia toimintaperiaatteeltaan kahdella eri tavalla. Ensimmäisessä tapauksessa rakennetaan niin kutsuttu differentiaalivahvistin, joka vahvistaa edellä mainitun pietsosähköisen värähtelijän tuottaman AC-signaalin mikrokontrollerille sopivalle tasolle. Toisessa tapauksessa käytetään operaatiovahvistinta vertailijana, joka vertaa sen sisääntuloihin tulevia jännitteitä ja sen mukaan säättää ulostulonsa joko operaatiovahvistimen positiiviseen tai negatiiviseen käyttöjännitteeseen. [26]

## **2.9 GPS-järjestelmän toiminta**

Kuva 5 esittää koko järjestelmää ja sitä, miten kaikki edellä mainitut komponentit ja lohkot yhdistyvät toisiinsa. GPS-järjestelmää ohjaa Arduino Uno -kehitysalustassa oleva ATmega328P-mikrokontrolleri, joka lähettää ja vastaanottaa tietoa sekä ohjaa eri komponentteja kehitysalustalla olevien I/O-pinnien avulla.



**Kuva 5.** Järjestelmän piirikaavio.

Järjestelmän tarvitsema 5 V käyttöjännite saadaan ensisijaisesti muuttamalla moottoripyörän akun tuottama 12 V jännite step-down -muuntimella 5 V jännitetasolle, mutta toissijaisesti käyttöjännite saadaan akkupankista akun ollessa kytkettynä irti. Vaihdos eri käyttöjännitteiden välillä tehdään käyttämällä SPDT (engl. Single Pole Double Throw, kaksi sisäänmeno, yksi ulostulo) -relettä, jonka sisäisen johtimen asentoa ohjataan akun jännitteellä. Kun akun jännitettä ei ole saatavilla, releen sisäinen johdin palautuu perusasentoon, mikä yhdistää akkupankin tuottaman jännitteen järjestelmään. Mikrokontrolleri, GSM-moduuli sekä operaatiovahvistimet tarvitsevat kaikki saman 5 V käyttöjännitteen toimiakseen.

SIM5320E-moduuli kommunikoi mikrokontrollerin kanssa sarjaliikenneväylän kautta, jota käyttämällä voidaan moduulille antaa käskyjä ja vastaanottaa moduulin lähettämä

data. GSM-moduuli tuo järjestelmään tekstiviestiominaisuuden ja kyvyn paikantaa järjestelmän sijainti. Näitä ominaisuuksia käyttämällä saadaan paikannettua moottoripyörä ja lähetettyä omistajalle moottoripyörän sen hetkinen sijainti. GSM-moduuli on myös mahdollista sammuttaa, kun järjestelmä asetetaan lepotilaan. Sammutus tapahtuu käyttämällä N-kanavaista MOSFET (engl. Metal-oxide-semiconductor field-effect transistor, Metallioksidi-puolijohdekanavatransistori) - transistoria. N-kanavaisen transistorin nielu ja lähde eivät johda, ellei hilaalle kohdisteta sopivaa jännitettä. Transistoria voidaan siis ohjata mikrokontrollerilla yhdistämällä digitaalipinni transistorin hilaan ja syöttämällä siihen 5 V jännite, jolloin transistorin nielun ja lähteen välille syntyy johtava käytävä, joka yhdistää moduulin maahan.

Differentiaalivahvistimella saadaan vahvistettua pietsosähköisen värähtelijän tuottama AC-signaali, kun siihen kohdistuu rasiusta. Vahvistettu signaali ohjataan komparaattorin positiiviseen sisäänmenoon, joten signaalin ollessa tarpeeksi voimakas, komparaattorin ulostuloon tulee 5 V. Komparaattorin vertailukynnystä voidaan säätää muuttamalla potentiometrin resistanssia, jolloin komparaattori, ja siitä seuraten koko järjestelmä, saadaan reagoimaan herkemmin pietsosähköisen värähtelijän signaaleihin.

Järjestelmä täytyy haluttaessa pystyä sammuttamaan, joten manuaalista sammutusta varten on olemassa järjestelmän ohitusnappi. Napin ollessa päällä kytkeytyy nappiin liitetty digitaalipinni maahan. Mikrokontrollerille ladatun ohjelman huomattessa muutoksen digitaalipinnin tilassa järjestelmä asetetaan uneen, kunnes nappi painetaan pois päältä. Toinen järjestelmän sammutustapa on moottoripyörän käynnissä olon tunnistaminen. Moottoripyörän ollessa päällä syötetään 12 V jännite jännitteenjakajaan, johon kytketty digitaalipinni lukee siihen tulevan jännitteen loogisena ykkösenä eli korkeana tilana. Ohjelma tunnistaa pinnin korkean tilan ja asettaa järjestelmän unitilaan, kunnes moottoripyörä sammutetaan ja pinni menee jälleen matalaan tilaan eli maapotentiaaliin.

## 3. JÄRJESTELMÄN RAKENTAMINEN

Tässä luvussa käydään läpi järjestelmän rakentamiseen kuuluvia eri välivaiheita. Toimivan järjestelmän toteuttaminen koostuu neljästä välivaiheesta: IC-komponenttien valinta, prototyypin rakentaminen ja testaus, järjestelmän rakennus ja ohjelmakoodin kirjoittaminen.

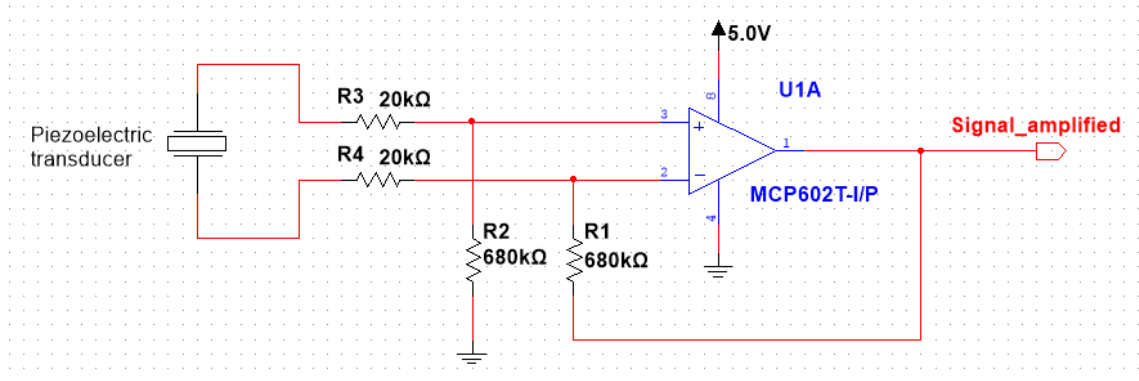
### 3.1 Järjestelmän IC-komponenttien valinta

Työssä hyödynnettiin muutamaa IC (engl. Integrated Circuit, mikropiiri) -komponenttia, joita käyttämällä mahdollistettiin järjestelmän koon pienentäminen. Jos työssä käytetyt IC-komponentit olisi rakennettu erillisistä komponenteista, työn koko olisi helposti paisunut liian isoksi ja se olisi ollut erittäin vaikea asentaa huomaamattomaan paikkaa moottoripyörän rakenteisiin.

Järjestelmä tarvitsee toimiakseen 5 V käyttöjännitteen, joten moottoripyörän akun tuottama 12 V pitää saada laskettua järjestelmälle yhteensopivalle tasolle. Jännitteen laskemiseen voidaan käyttää kahta erilaista regulaattoria, hakkuriregulaattoria tai lineaarista regulaattoria. Työhön valittiin step-down -hakkuriregulaattori, jonka parempi hyötysuhde ja pieni koko sopivat työn käyttötarkoitukseen hyvin. Työhön valittiin käytettäväksi Micrelin valmistama LM2574-5.0YN step-down -regulaattori, joka aikaisemmin nopeasti esiteltiin luvussa 2.5 ja kuvassa 4. Regulaattorin keskeisiä ominaisuuksia ovat muun muassa pienikokoinen DIP (engl. Dual In-line Package, kaksoisrivipakkaus) -kotelo ja korkea hyötysuhde, ja se hyväksyy käyttöjännitteeksi mitä tahansa 40 V ja 5 V väliltä sekä tuottaa tällä sisäänmenojännitevälillä aina 500 mA ulostulovirran. [19] Regulaattori vaatii muutaman komponentin pakkauksensa ulkopuolelle, koska esimerkiksi keloja ei voida valmistaa suoraan mikropiiriin ja jopa matalakapasitanssiset kondensaattorit ovat erittäin kalliita valmistaa suoraan mikropiiriin.

Pietsoelektrisen värähtelijän tuottaman signaalin jännitearvoiksi saatiin mitattua noin 300-500 mV. Täten kyseistä signaalia täytyy saada vahvistettua noin 10-kertaiseksi, jotta työssä käytetty mikrokontrolleri onnistuu tunnistamaan jännitteen. Jännitteen vahvistus saadaan aikaan käyttämällä operaatiovahvistinta differentiaalivahvistinmuodossa, joka ottaa sisäänmenoihinsa tulevat jännitteet ja vahvistaa jännitteiden erotuksen. Näin saadaan aikaan tarpeeksi suuri jännite, jonka mikrokontrolleri pystyy tunnistamaan.

Työtä varten valittiin Microchipin valmistama MCP602, joka käyttää käyttöjännitteensä 2,7 V - 5,5 V ja jonka avoimen silmukan vahvistus on noin 110 dB. [25]



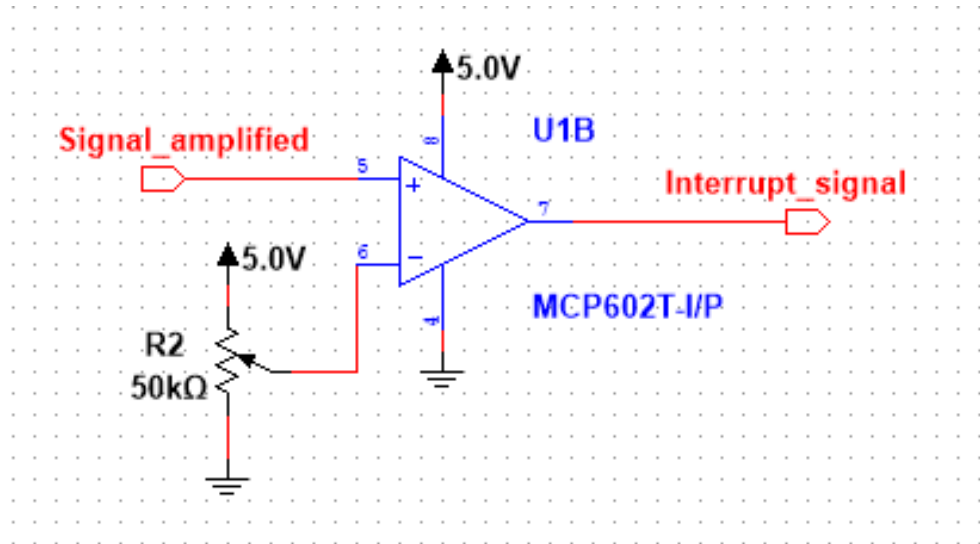
**Kuva 6.** Differentiaalivahvistinkytkentä.

Tällöin kytkettäessä operaatiovahvistimeen kuvan 6 mukaisesti 20kΩ ja 680kΩ vastukset, saadaan laskettua ulostulon jännite seuraavalla kaavalla

$$V_{out} = \left(\frac{R_2}{R_3 R_2}\right) * \left(\frac{R_4 R_1}{R_4}\right) * V^- - \frac{R_1}{R_4} * V^+ = \left(\frac{R_1}{R_4}\right) * (V^+ - V^-) \quad (1)$$

Kaavalla 1 saadaan laskettua differentiaalivahvistinpiirin vahvistukseksi  $A_v = 34$ , jolloin saadaan aikaan huomattavasti alkuperäistä voimakkaampi signaali. Signaalin tuottama mitattu jännite ylittää suurimman osan ajasta mikrokontrollerin tunnistaman noin 3 V rajan, jolloin sitä voitaisiin käyttää sellaisenaan keskeytysignaalien luomiseen.

MCP602-IC pitää sisällään kaksi erillistä operaatiovahvistinta samassa pakkauksessa, joista työssä ensimmäistä hyödynnetään edellä mainitussa differentiaalivahvistimessa ja toista käytetään komparaattorikytkennässä. Komparaattori, eli vertailija, vertailee sen sisäänmenoissa olevia jännitteitä. Työssä käytetty komparaattori on avoin operaatiovahvistinkytkentä, jonka korkeasta avoimen silmukan vahvistuksesta seuraa, että ulostulo saturoituu joko operaatiovahvistimen positiiviseen 5 V käyttöjännitteeseen tai maapotentiaaliin. Komparaattoria voidaan siis käyttää analogisella signaalilla ohjattavana 1-bittisenä analogi-digitaalimuuntimena.



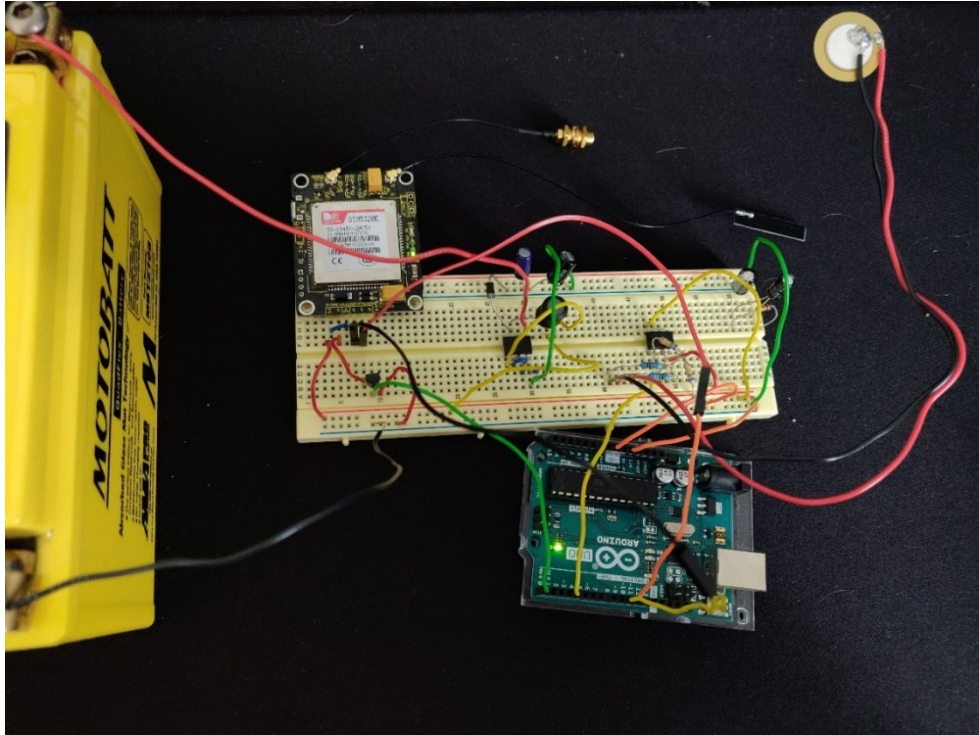
**Kuva 7.** Komparaattorikytkentä.

Työssä käytetään kuvassa 7 esitettyä komparaattorikytkentää, jossa positiiviseen sisäänmenoon tulee pietsosähköisen elementin tuottama ja differentiaalivahvistimen vahvistama signaali. Negatiiviseen sisäänmenoon on kytketty potentiometriä käyttämällä tehty jännitteenjakaja, jota käyttämällä voidaan säätää negatiiviseen sisäänmenoon pääsevää jännitettä. Vahvistetun signaalin ollessa suurempi kuin jännitteenjakajalla säädetty jännite, asettuu komparaattorin ulostulo positiiviseen käyttöjännitteeseen. Kytkentää käyttämällä saadaan rakennettua säädettävä keskeytyssignaali tuottaja, jota käyttämällä voidaan helposti säätää järjestelmän herkkyyttä ilman järjestelmän poistamista moottoripyörästä.

### 3.2 Prototyyppi

Järjestelmän eri lohkojen toimivuuden varmistamiseksi järjestelmä kasataan ensin prototyyppinä koekytkentälevylle, jossa kaikki eri lohkot testataan ensin erikseen ja sen jälkeen yhdessä. Näin voidaan helposti tutkia järjestelmän eri osien mahdollisia toimimattomuuden syitä ja ratkaista nämä ongelmat ennen oikean järjestelmän kasaamista ja juottamista kytkentälevylle. Alla olevassa kuvassa 8 on järjestelmän eri osat yhdistetty toisiinsa ja järjestelmä on toimintakykyinen pois lukien järjestelmän sammuttamiskytkimen, varavirtalähteen ja jännitteenjakajapiirin.





**Kuva 8.** GPS-seurantajärjestelmän prototyyppi.

Kuvassa 8 järjestelmä yritettiin saada mahdollisimman lähelle todellista toimintaympäristöä, minkä vuoksi moottoripyörän akku toimii käyttöjännitteen syöttäjänä. Prototyypillä testattiin muun muassa step-down -regulaattorin antaman maksimivirran riittävyys, pietsosähköisen värähtelijän tuottaman signaalin vahvistus ja SIM5320E-toiminta muiden komponenttien kanssa.

Ennen testauksen aloittamista täytyi vaihtaa GSM-moduulin oletusarvoisesti käyttämä baud-nopeus 115200 bit/s nopeudesta 4800 bit/s nopeuteen, koska käytetty sarjaliikenneväylä on ohjelmistolla toteutettu ja 115200 bit/s datanopeus on liian nopea. Baud-nopeuden asettamiseen käytettiin Adafruit\_FONA -kirjastoa [27], josta löytyi ohjelma, joka muutti baud-nopeuden automaattisesti. Testauksen helpottamiseksi mikrokontrollerille kirjoitettiin ja ladattiin ohjelmassa 1 nähtävä yksinkertainen testikoodi, jolla varmistettiin GSM-moduulin tekstiviestin lähettämisen toiminta.

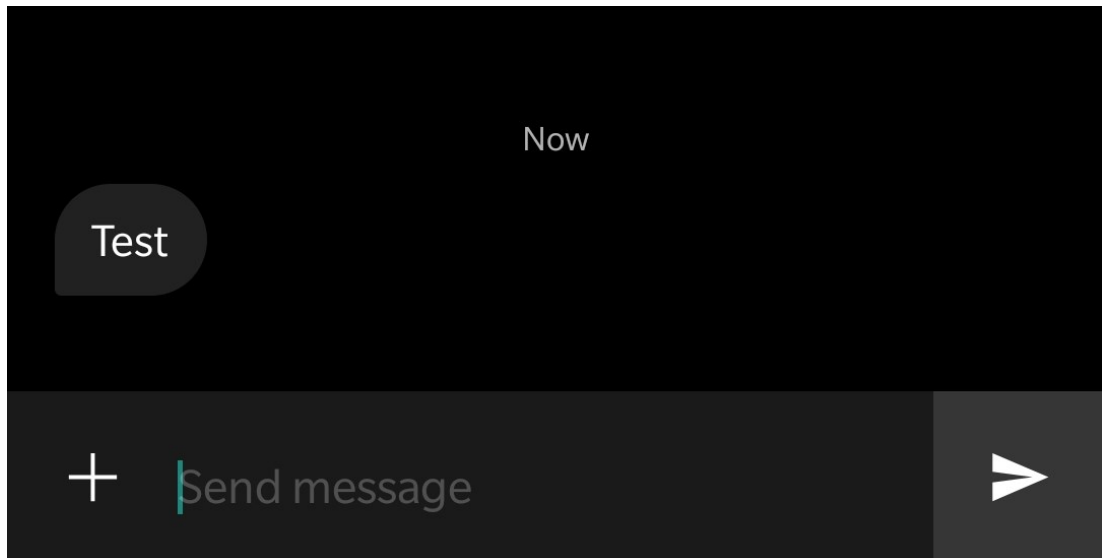
```

1 #include <SoftwareSerial.h>
2 SoftwareSerial SIM(8,7);
3
4 const int piezoPin = 2;
5 volatile bool sendsSMS = false;
6
7 void setup() {
8     Serial.begin(4800);
9     SIM.begin(4800);
10    pinMode(piezoPin, INPUT);
11    attachInterrupt(digitalPinToInterrupt(piezoPin), piezo, RISING);
12 }
13
14 void loop() {
15     while(SIM.available()) {
16         Serial.write(SIM.read());
17     }
18
19     while(Serial.available()) {
20         SIM.write(Serial.read());
21     }
22
23     if(sendsSMS) {
24         String command = "AT+CMGS=";
25         command += "'";
26         command += "0408674704";
27         command += "'";
28
29         String message = "Test";
30
31         SIM.println("AT+CMGF=1");
32         delay(250);
33         SIM.println(command);
34         delay(250);
35         SIM.print(message);
36         delay(250);
37         SIM.println((char)26);
38         sendsSMS = false;
39     }
40 }
41
42 void piezo() {
43     sendsSMS = true;
44 }
45

```

### **Ohjelma 1. Testiohjelman koodi.**

Koodin toimiessa oikein pitäisi pietsosähköisen elementin tärähdyksen synnyttää tarpeeksi vahva signaali, jonka mikrokontrolleri tunnistaa. Mikrokontrollerin havaitessa sisään tulevan signaalin lähetetään asetettuun puhelinnumeroon testiviesti "Test".



**Kuva 9.** Saapunut testiviesti.

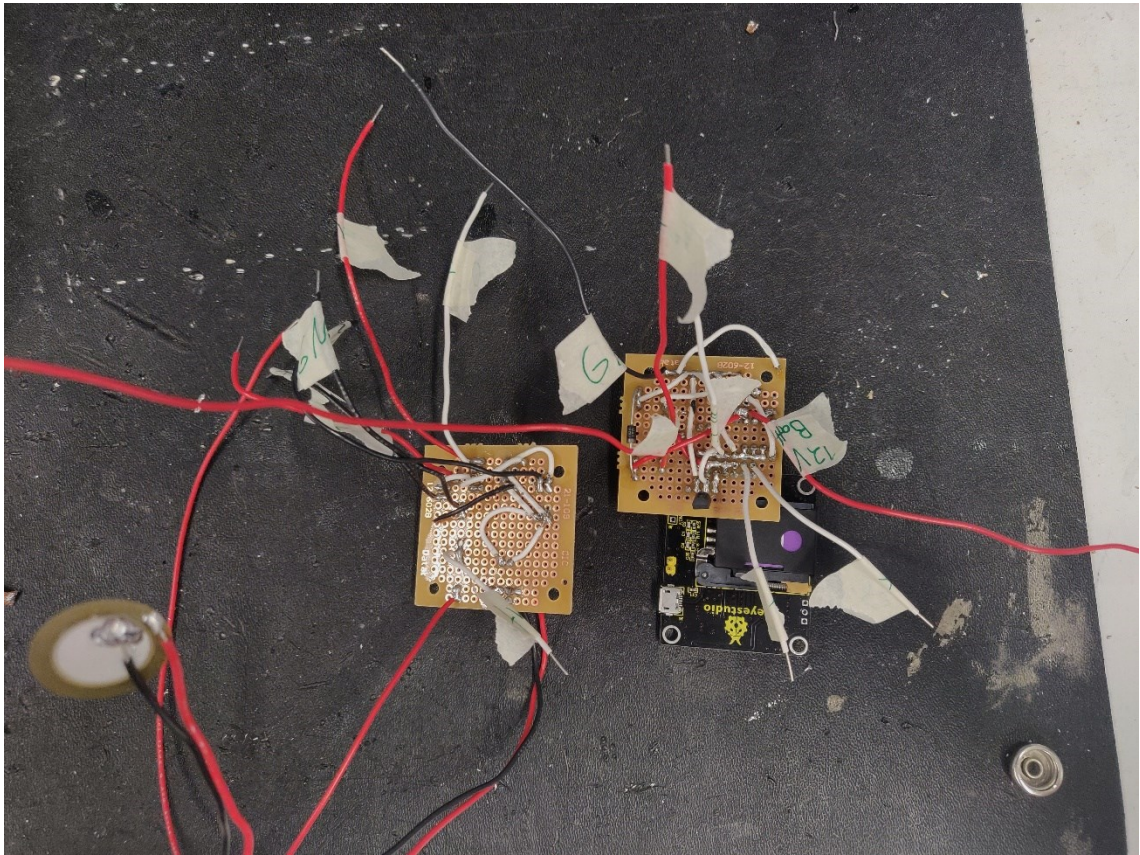
Pelkistetyllä järjestelmällä suoritettu testi osoittautui onnistuneeksi, kun kuvassa 9 näkyvä viesti saapui muutaman sekunnin odotuksen jälkeen perille puhelimeen. Prototyypin toimiessa pelkistetylle järjestelmälle asetettujen spesifikaatioiden mukaan voidaan siirtyä järjestelmän kasaamiseen prototyypilevyille.

### 3.3 Kasausprosessi

Järjestelmän rakentamista varten oli hankittu kaksi pientä 4,5 cm x 4,6 cm kokoista prototyypilevyä, joille pystyi helposti juottamaan kiinni tarvittavat läpiladottavat komponentit. Ensimmäiselle piirilevyille juotettiin käyttöjännitteen step-down -regulaattori ja sen tarvitsemat passiivikomponentit, käyttöjännitteen valintaan käytetty rele sekä GSM-moduuli. Toiselle levyille juotettiin differentiaalivahvistin ja komparaattori, jännitteenjakaja pyörän käynnistyksen tarkkailuun sekä järjestelmän ohituskytkin. Kun komponentit erotettiin niiden toiminnan perusteella kahteen loogiseen osaan, voitiin varmistaa vielä juottamisen jälkeen eri lohkojen toiminta. Esimerkiksi ensimmäisellä levyllä on työn toiminnalle keskeinen GSM-moduuli ja käyttöjännitteen regulointi, joten moduulin toiminnan testaus moottoripyörän akun tuottamaa jännitettä reguloimalla oli helppoa.

Ennen komponenttien lopullista paikalleen juottamista komponentit yritettiin sijoitella prototylevyille mahdollisimman optimaalisiin paikkoihin toisiinsa nähden. Kun komponentit asetellaan järkevästi, saadaan esimerkiksi komponenttien yhteinen maa tai käyttämä käyttöjännite helposti kaikkien komponenttien ulottuville. Myös pietsosähköinen

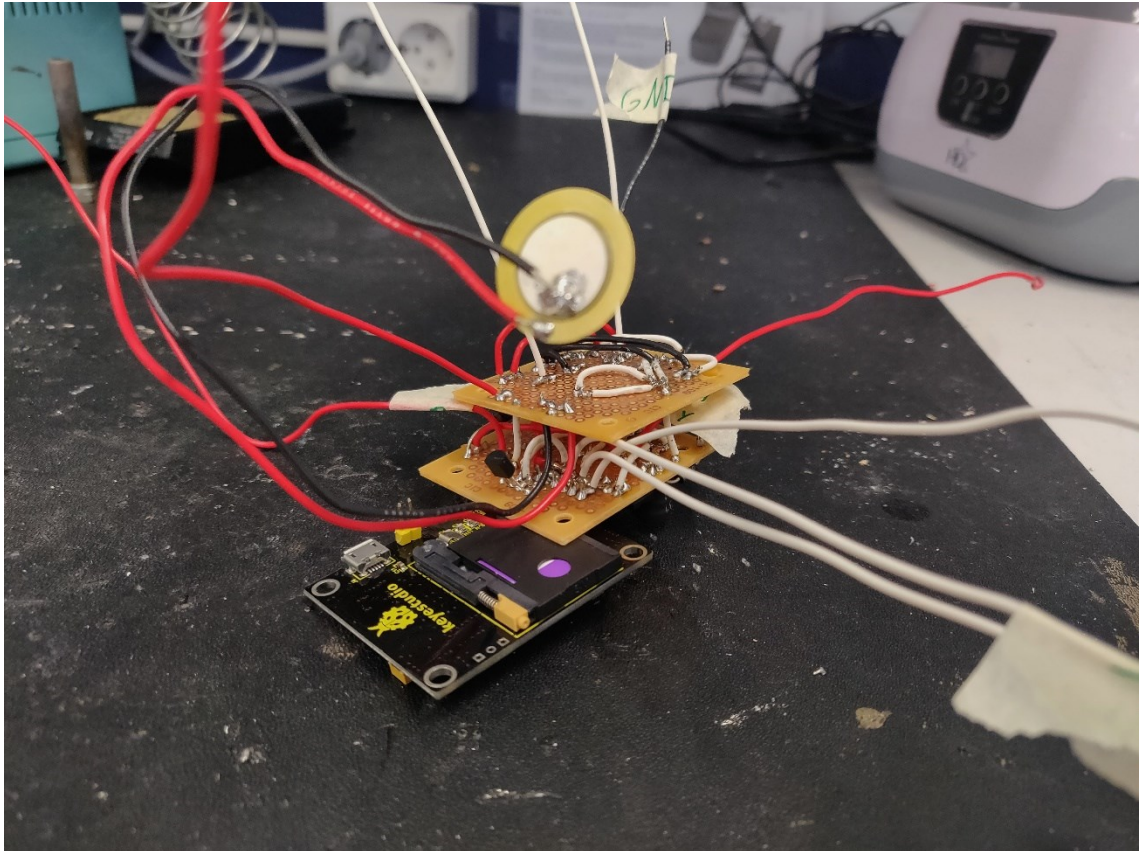
värähtelijä ja säätövastus oli tärkeä saada mahdollisimman reunalle, jolloin ne ovat helposti saavutettavissa.



**Kuva 10.** Suurimmalta osin valmiiksi juotetut erilliset prototyypilevyt.

Kuvassa 10 on oikealla aikaisemmin mainittu ensimmäinen levy, josta löytyvät muun muassa GSM-moduuli sekä käyttöjännitteen regulointipiiri, ja vasemmalla toinen levy, josta löytyvät muun muassa operaatiovahvistin ja pietsosähköinen värähtelijä. Kaikkien komponenttien toimiessa ja niiden ollessa juotettuna protolevyille voitiin levyt yhdistää toisiinsa. Levyt päädyttiin testiasetteluiden pohjalta asettelemaan päällekkäin, jolloin järjestelmä vie huomattavasti vähemmän tilaa verrattuna muihin harkittuihin konfiguraatioihin. Kuvassa 11 on nähtävillä päällekkäin asetellut protolevyt.





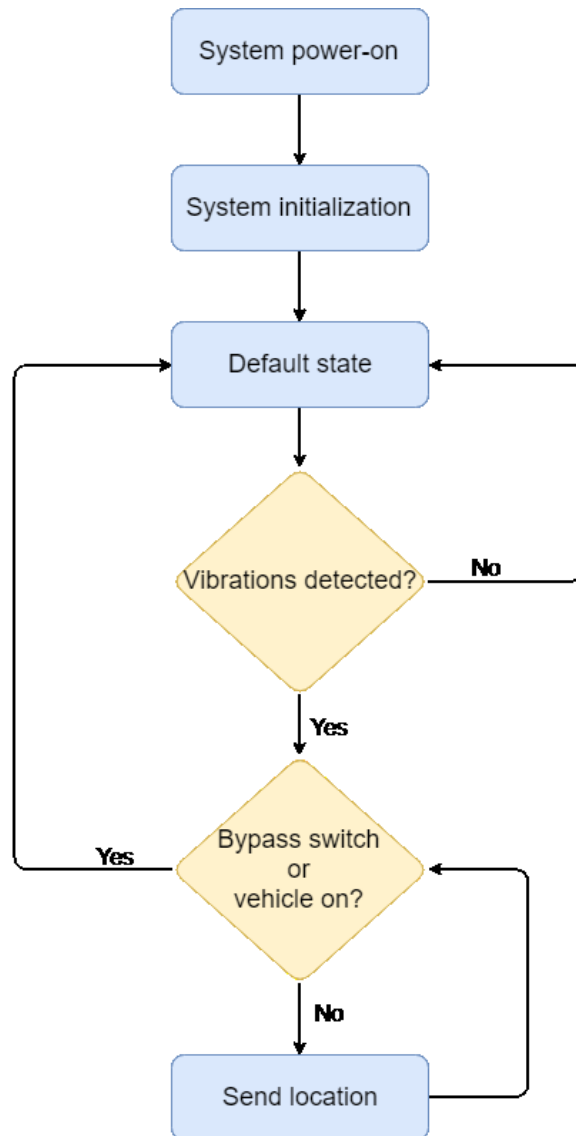
*Kuva 11. Päälekkäin asetellut prototyypilevyt.*

Levyt yhdistettiin toisiinsa juottamalla maatasot yhteen ja viemällä käyttöjännite toiselle levyille, jolloin saatiin aikaan yhtenäinen järjestelmä. Levyjen välille juotettiin myös kolme lisätukea tuovaa johtoa, jolloin levyt eivät pääse painamaan niiden välissä olevia johtoja ja aiheuttamaan johtojen juotoskohdille aiheetonta räsitusta. Kuvassa 11 ei ole vielä kiinnitettynä järjestelmän ohitusnappia, joka kiinnitettiin kuvan jälkeen pidempien akulle menevien johtimien kanssa.

### **3.4 Ohjelmakoodin kirjoittaminen**

Järjestelmän fyysisen osan ollessa lähestulkoon valmis siirryttiin kirjoittamaan mikrokontrollerin ohjelmakoodia, jonka valmis ja kommentoitu versio on nähtävissä liitteessä A. Ohjelmakoodia kirjoitettaessa täytyi ottaa huomioon muun muassa seuraavia asioita: milloin lähetetään viesti, kuinka usein lähetetään viesti, miten ilmoittaa

pyörän sijainti ja mikä pysäyttää viestien lähettämisen. Ohjelman rakenteen hahmottamista varten tehtiin vuokaavio.



**Kuva 12.** Ohjelman toiminnan vuokaavio.

Järjestelmän käynnistyessä alustetaan tarvittavat lohkot, kuten sarjaliikenneväylä ja digitaalipinnit, ja itse järjestelmä oikeaan tilaan. Alustamisprosessin jälkeen siirytään ensimmäiseen ongelmaan: mikä on järjestelmän oletustila, johon aina palataan? Oletustilaksi valittiin tila, jossa järjestelmä vie mahdollisimman vähän virtaa, eli GSM-moduuli on sammutettu ja mikrokontrolleri on asetettu uneen. Mikrokontrollerin datalehdeltä nähdään eri unitilojen eroavaisuudet, joista valittiin käytettäväksi power-down mode eli virrankatkaisutila. Virrankatkaisutilassa mikrokontrollerin kellosignaali sammutetaan, jolloin vain asynkroniset lohkot ovat toiminnassa; tällöin mikrokontrollerin voi herättää unesta vain esimerkiksi ulkoinen keskeytys, kuten pietsosähköisen elementin tuottama signaali. [6, s.39] Unitila integroitiin koodiin hyödyntämällä

rocketscreamin Low-Power -kirjastoa [28], jota käyttämällä voidaan helposti asettaa nukkumisaika ja järjestelmän eri lohkojen sammutus.

Pietsosähköinen värähtelijä tuottaa keskeytyssignaalin mikrokontrollerille, jolloin mikrokontrolleri herää unesta ja tarkistaa, onko moottoripyörä päällä tai onko ohitusnappi päällä. Jos kumpikaan edellä mainituista ehdoista toteutuu, palataan järjestelmän perustilaan eli unitilaan, kunnes tulee uusi keskeytyssignaali. Molempien ehtojen ollessa epätosia siirrytään suorittamaan sijainnin hakemista, jonka onnistuessa lähetetään sijaintitiedot tekstiviestillä 2,5 min välein pyörän omistajalle. Tämä paikannus-lähetys-odotus kierto jatkuu niin pitkään, kunnes ohjelma huomaa toisen edellä mainituista ehdoista täyttyvän tai käyttöjännite kytketään irti.

## 4. JÄRJESTELMÄN TESTAUS

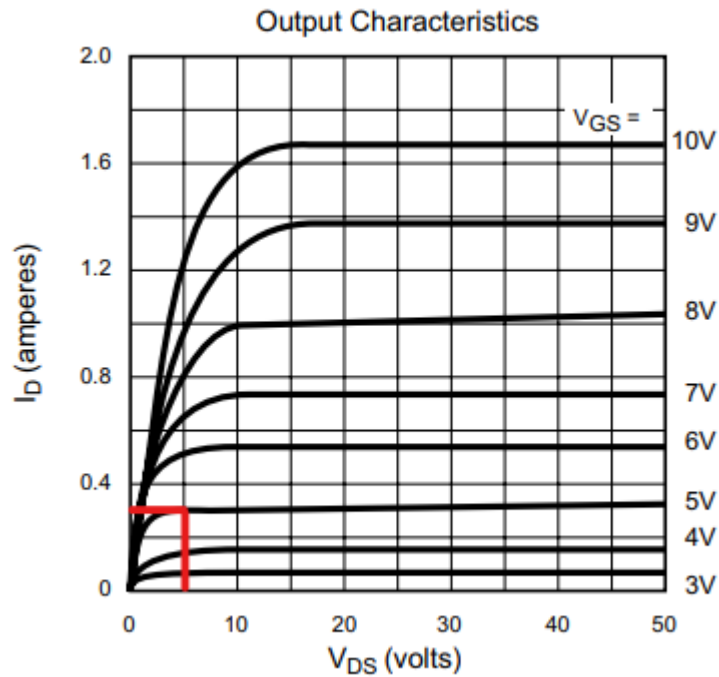
Järjestelmän toiminnan tarkempi testaus koostuu kahdesta vaiheesta: Testaus erillään moottoripyörästä ja testaus moottoripyörään asennettuna. Näin voidaan vielä tehdä järjestelmään helposti muutoksia, jos havaitaan järjestelmältä luvatonta käytöstä. Kun järjestelmä on asennettuna moottoripyörään, järjestelmän toiminnan herkkyyttä voidaan säätää haluttuun kohtaan, jolloin järjestelmä ei reagoi haluttua pienempiin tärähdyksiin.

### 4.1 Ulkoinen testaus

Valmiiksi saadun järjestelmän testaus toteutettiin kytkemällä järjestelmä releeseen, johon moottoripyörän akku sekä USB-akkupankki ovat kytkettyinä. Arduino-kehitysalusta, ja sen mukana mikrokontrolleri, saivat aluksi käyttöjännitteensä tietokoneeseen kytketystä USB-kaapelista. USB-kaapelin avulla tarkkailtiin järjestelmän sarjaliikenneväylällä tapahtuvaa kommunikaatiota, mutta kehitysalusta siirrettiin myöhemmin käyttämään samaa käyttöjännitettä kuin muu järjestelmä. Tällä mahdollisimman todenmukaisella konfiguraatiolla voitiin varmistaa järjestelmän oikea toimiminen, kun järjestelmän ensisijainen käyttöjännitelähde kytkeytyy pois käytöstä. Kun akku kytkettiin pois järjestelmästä, palautui rele perusasentoonsa, jolloin USB-akkupankki kytkeytyi antamaan virtaa järjestelmälle. Järjestelmä käynnistyi uudelleen käyttöjännitteen vaihtuessa, minkä olisi voinut ehkäistä asettamalla releen ulostuloon suuremman kondensaattorin; tällöin kondensaattori olisi voinut pitää mikrokontrolleria päällä, kunnes releen ulostuloon kytkeytyy toinen käyttöjännite.

Testauksessa huomattiin GSM-moduulin satunnaisesti sammuvan käynnistysvaiheessa, erityisesti jos käynnistyksen aikana mikrokontrolleri yritti kommunikoida moduulin kanssa. Aluksi sammumisen syyksi epäiltiin step-down -regulaattorin 500 mA virtarajaa, sillä mitattaessa regulaattorin ulostuloa nähtiin pieniä jännitteen notkahduksia. Tarkasteltaessa muita mahdollisuuksia päädyttiin testaamaan myös moduulia ohjaavaa MOSFET-transistoria, jonka lähde ja nielu yhdistettiin hyppylangalla toisiinsa. Näin saatiin luotua suora reitti moduulin ja maan välille, mikä ohitti transistorin, ja järjestelmän huomattiin toimivan täysin. Tarkasteltaessa transistorin datalehteä huomattiin 5 V hila-lähde-jännitteellä läpi kulkevan jatkuvan virran rajoittuvan noin 300 mA.

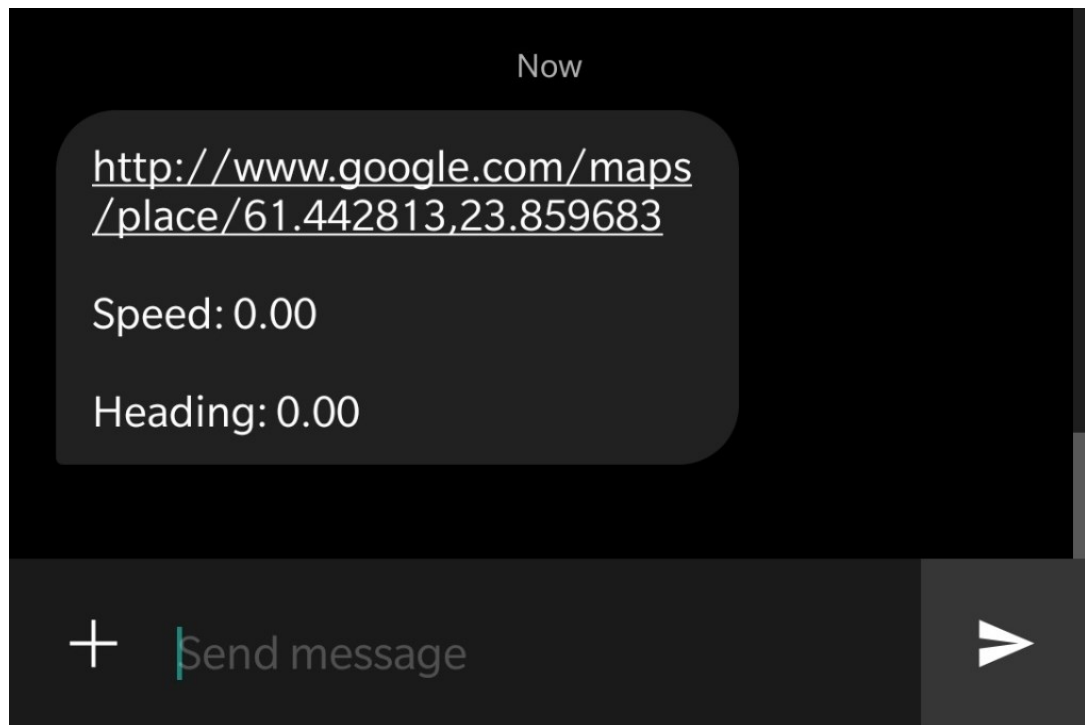




**Kuva 13.** VN2106 MOSFET-transistorin virtaraja. [29]

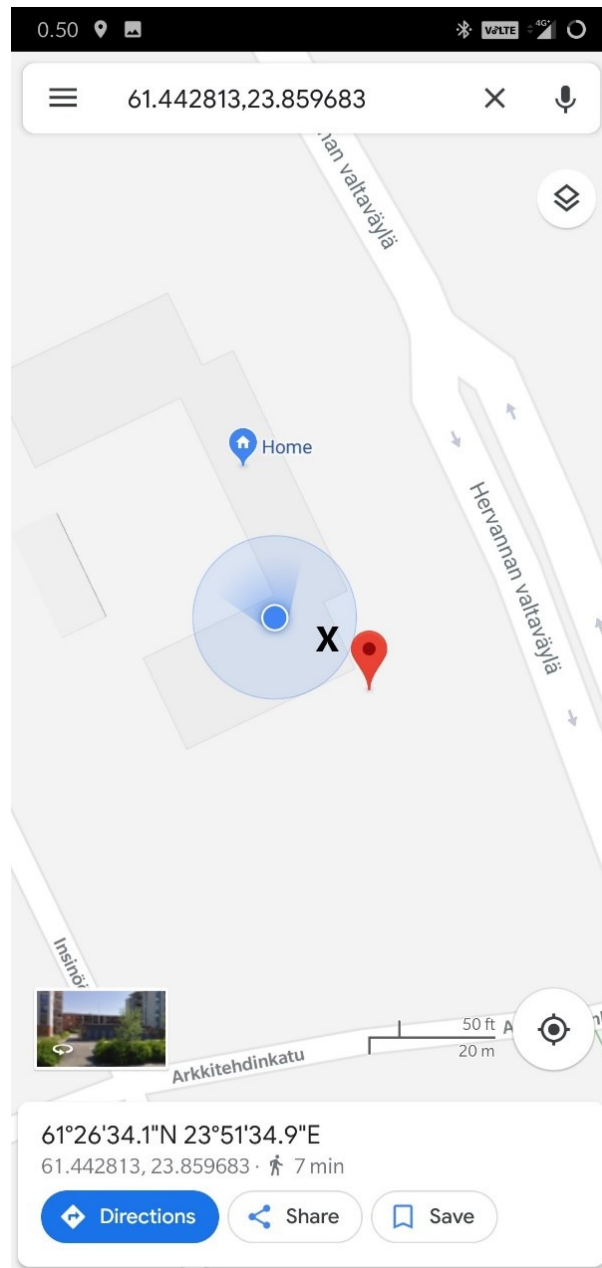
Järjestelmän halutun ja luotettavan toiminnan takaamiseksi piti VN2106 MOSFET vaihtaa toiseen N-kanavaiseen transistoriin, jonka läpi pääsee tarpeeksi virtaa saatavilla olevalla 5 V jännitteellä. Järjestelmä piti saada nopeasti takaisin toimintakuntoon jatkotestausta varten, joten vanhan transistorin tilalle juotettiin ON semiconductorin valmistama RFP50N06 teho-MOSFET. Uuden transistorin datalehdeltä varmistettiin ensin kuinka paljon virtaa 5 V hila-lähde-jännitteellä transistorin läpi pystyy kulkemaan; läpikulkevan virran arvoksi saatiin noin 20A [30]. Transistori on huomattavasti ylimitoitettu rakennettavaan järjestelmään, mutta sitä käyttämällä saatiin järjestelmä toimimaan moitteettomasti ja päästiin jatkamaan testausta.

Kun järjestelmä kytkettiin uudelleen mahdollisimman todenmukaiseen konfiguraatioon, joka aikaisemmin mainittiin, saatiin todennettua järjestelmän toiminta molempia käyttöjännitteitä käytettäessä. Seuraavaksi ladattiin toimiva ohjelma mikrokontrollerille, mutta pidettiin mikrokontrolleri kiinni tietokoneessa USB-kaapelilla, jolloin voitiin tarkkailla helposti sarjaliikennettä ja varmistaa ohjelman toiminta. Ohjelman toiminnan varmistuttua kytkettiin koko järjestelmä samaan käyttöjännitteeseen ja simuloitiin liikettä kohdistamalla pietsosähköiseen elementtiin voimakas tärähdys. Hetken odottelun jälkeen saapui tekstiviesti, jonka sisältönä löytyi linkki pyörän sijaintiin sekä nopeus ja suunta.



**Kuva 14.** Saadut sijaintitiedot.

Paikannuksen tarkkuuden varmistamiseksi avattiin kuvassa 14 esitetty saapunut linkki ja verrattiin koordinaatteja puhelimen ilmoittamaan sijaintiin. Nopeus ja suunta olivat molemmat odotetusti 0, sillä järjestelmä oli paikallaan ulkoisen testauksen aikana.



**Kuva 15.** Järjestelmän ilmoittama sijainti verrattuna puhelimen sijaintiin.

Kuvasta 15 nähdään järjestelmän ilmoittaman sijainnin olevan erittäin lähellä verrattaessa oikeaan sijaintiin. Todellisuudessa järjestelmän tarkka sijainti paikannushetkellä oli kuvassa olevan mustan "X"-merkin kohdalla. Järjestelmä jatkoi viestien lähettämistä, kunnes mikrokontrollerille annettiin tieto mennä takaisin uneen, mikä testattiin eri tapauksissa joko käyttämällä ohitusnappia tai simuloimalla pyörän päällä oloa.

GPS-järjestelmän virrankulutusta mitattiin testien aikana, kun kaikki järjestelmän osat ottivat käyttöjännitteensä moottoripyörän akusta. Lepotilassa järjestelmän keskimääräinen mitattu virrankulutus oli noin 90 mA ja järjestelmän suorittaessa

paikannusta virrankulutus oli noin 300 mA. Virrankulutuksen pohjalta voitiin laskea akkupankin kesto, jos moottoripyörän akku kytkeytyisi irti. Akkupankin kapasiteetti on 5000 mAh, joten lepotilassa teoreettisesti akkupankki kestäisi 55,6 tuntia ja paikannettaessa ajoneuvoa 16,7 tuntia. Järjestelmän akunkesto akkupankkia käytettäessä on hyväksyttävä ja se on helposti kasvatettavissa isompaa akkupankkia käyttämällä. Kun järjestelmä osoittautui odotetulla tavalla toimivaksi, voitiin siirtyä integroimaan järjestelmä moottoripyörään.

## 4.2 Järjestelmän asennus ja testaus

Testausta varten GPS-järjestelmä asennettiin helposti käsiksi päästävään osaan moottoripyörästä, joka on tässä tapauksessa mittariston alapuolella oleva rakenne. Järjestelmä suojattiin vedeltä ja muilta uhkatekijöiltä asettamalla se kahden anti-staattisen muovipussin sisälle ja kiinnittämällä se tukevasti moottoripyörään kiinni. Järjestelmän paikannusta auttava ja kuvassa 16 esillä oleva GPS-antenni piti myös asentaa järkevään paikkaan, jossa sillä olisi mahdollisimman hyvä yhteys satelliitteihin. Kiinnityspaikka löytyi lopulta moottoripyörän matkustajan penkin alla olevasta säilytystilasta, sillä moottoripyörän kyydissä on harvoin matkustajaa johtuen moottoripyörän ajoasennosta.



**Kuva 16.** GPS-antennin kiinnitys sijainti.

Järjestelmän virtajohtojen vieminen moottoripyörän akulle oli seuraava vaihe, jossa johdot yritettiin viedä mahdollisimman huomaamattomasti rakenteitten seassa. Johtojen

piilottaminen onnistui hyvin, sillä moottoripyörän kanteet peittävät ja suojaavat suurimman osan pyörästä. Myös järjestelmän ohitusnappi piilotettiin kanteiden sekaan, jolloin vain omistaja voi kytkeä järjestelmän pois päältä esimerkiksi huoltotoimenpiteitä varten. Moottoripyörän päällä olon aistinta toteutettiin kytkemällä sisäänmeno pääajovalon positiiviseen terminaaliin, sillä ajovalo syttyy automaattisesti, kun virtalukko käännetään ON-asentoon. Moottoripyörässä ei ole mahdollisuutta sammuttaa pääajovaloa, joten GPS-järjestelmä asettuu uneen moottoripyörän ollessa käynnissä. Pietsosähköinen värähtelijä kiinnitettiin kaksipuoleisella teipillä runkoon, jonka kautta kaikki mahdolliset moottoripyörään kohdistuvat tärähdykset voidaan aistia järjestelmällä.

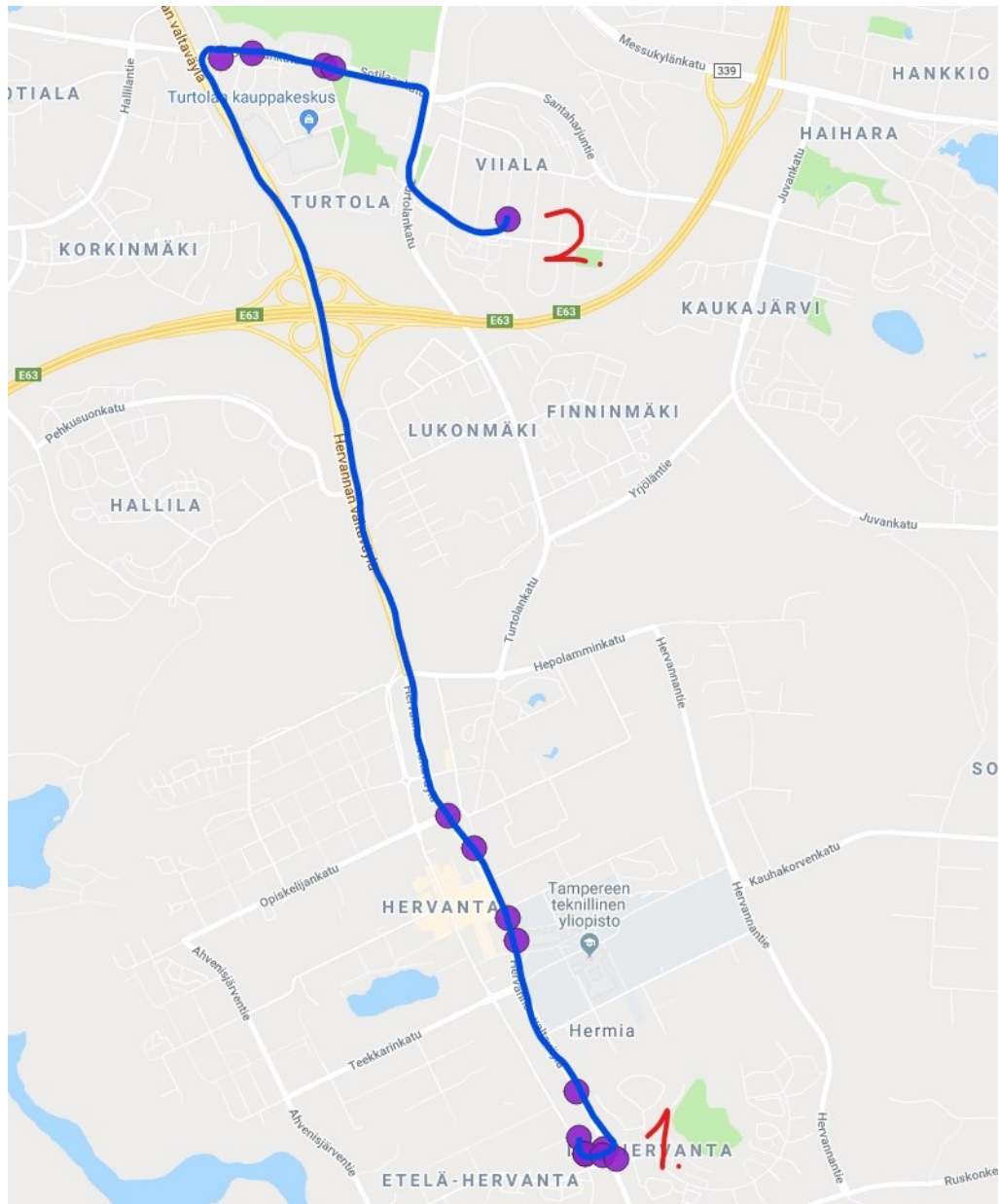


**Kuva 17.** GPS-järjestelmä kiinnitettynä väliaikaisesti moottoripyörään.

Järjestelmän kaikkien osien ollessa asennettuina paikoilleen varmistettiin vielä kuvassa 17 näkyvän järjestelmän kiinnitys, asetettiin järjestelmän herkkyys potentiometrillä vähän alle puolenvälin ja suoritettiin reaaliaikainen testaus. Suunniteltu testaus koostuu kolmesta eri vaiheesta: moottoripyörän simuloitu kaataminen, siirtäminen akun ollessa

irti kytkettynä ja GPS-järjestelmän testaus ajaessa. GPS-järjestelmän pitää jokaisessa vaiheessa aktivoitua ja lähettää sijainti, jotta järjestelmä voidaan todeta toiminnalliseksi. Moottoripyörän kaatumista simuloitiin manuaalisesti moottoripyörää kaatamalla realistisella nopeudella, mutta estettiin kuitenkin mahdollinen vahingoittuminen pysäyttämällä liike äkkinäisesti ennen pyörän osumista maahan. Viesti saapui perille hetken viiveellä testin toteuttamisesta, joten ensimmäinen vaihe todettiin onnistuneeksi. Toisessa vaiheessa irrotettiin moottoripyörän akku ja varmistettiin laitteen siirtyneen ottamaan käyttöjännitteensä akkupankista. Varmistuksen jälkeen lähdettiin siirtämään pyörää ympäri talon vieressä olevaa parkkialuetta. Moottoripyörä ei näyttänyt reagoivan alussa vaihteen vapaalle vaihdosta johtuvaan tärähdykseen, mutta järjestelmä käynnistyi hetkeä myöhemmin renkaan osuessa maakaivon hieman koholla olevaan kanteen. Pyörää talutettiin noin 5 min ajan ja tänä aikana perille tuli 3 viestiä, joten toinenkin vaihe voidaan todeta onnistuneeksi. Kolmas ja viimeinen vaihe koostuu reitin ajamisesta, jossa simuloidaan moottoripyörän kuljetusta toisella ajoneuvolla. Testiä varten irrotettiin pääajovalosta päällä olon tunnistava johdin, jotta GPS-järjestelmä kuvittelee moottoripyörän olevan pois päältä. Kuvasta 18 nähdään kaikki testauksen ajalta tulleiden viestien koordinaatit violetteina ympyröinä ja ajettu reitti sinisenä viivana.





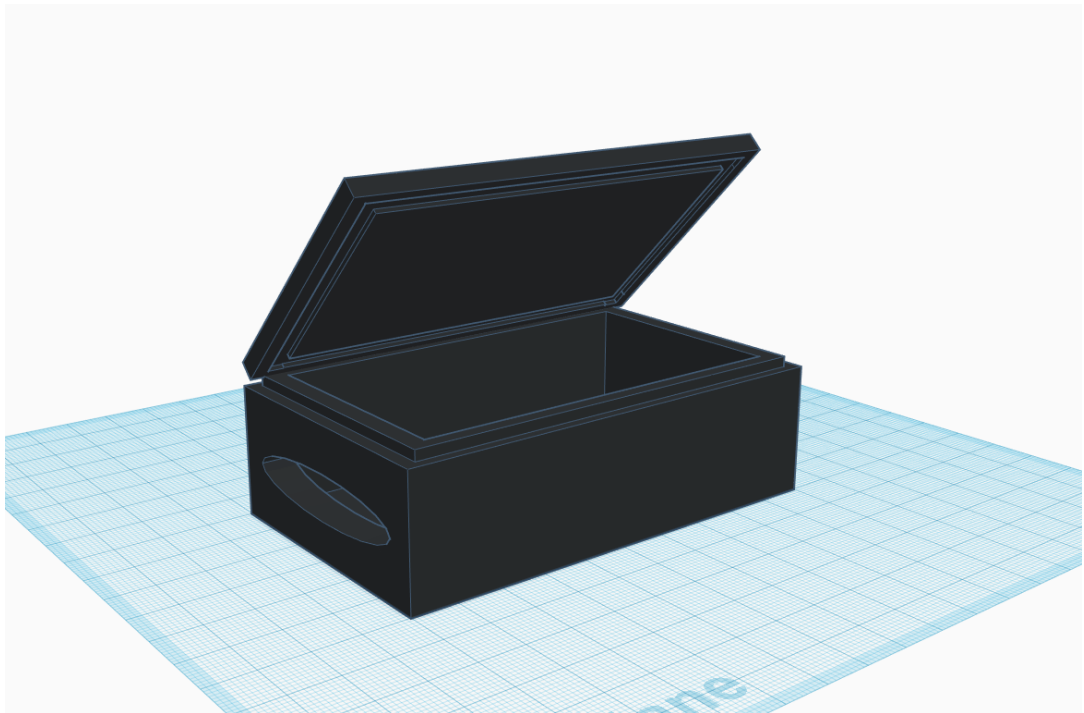
**Kuva 18.** GPS-järjestelmällä kartoitettu ajoreitti.

Kuvassa 18 punaisella numerolla 1 on merkitty aloitussijainti ja numerolla kaksi on merkitty kohdesijainti. Aloitussijainnissa nähdään aikaisempien testien antamien koordinaattien olevan lähekkäin, joten voidaan päätellä paikannetun sijainnin olevan tarpeeksi tarkka. Ajon aikana päätettiin testata myös ohitusnapin toimintaa pysähtyessä liikennevaloihin, josta seurauksena matkan varrella on yksi pitkä väli kahden violetin koordinaattiympyrän välillä. Sopivan hetken tullen järjestelmän ohitusnappi asetettiin pois päältä, jonka jälkeen koordinaatit jatkuvat kohdesijaintiin asti. GPS-järjestelmä toimi testin aikana moitteettomasti ja onnistui asettumaan lepotilaan sekä jatkamaan toimintaansa halutusti, mistä voidaan todeta viimeisen testin olevan myös onnistunut.

### 4.3 Kehityskohteet

Prototyypilevyt, joille järjestelmä rakennettiin, olisi voitu korvata itse suunnitellulla PCB:llä, joka kiinnittyisi suoraan Arduino Uno -kehitysalustan päälle. Tällöin lopputulos olisi ollut paljon siistimpi ja helpommin hahmotettavissa. Työn alkuvaiheissa PCB:n suunnittelu olisi kuitenkin ollut erittäin haastavaa, sillä piirikaavio muuttui jatkuvasti järjestelmän kehittyessä nykyiseen muotoonsa. Nykyinen järjestelmä on toimiva sekä kustannustehokkaampi verrattuna suunniteltavaan ja teetetävään PCB:en.

Järjestelmän koteloa ei ehditty saada valmiiksi tämän kandidaatintyön puitteissa, mutta sen 3D-malli kuitenkin on olemassa. Järjestelmän kiinnitys testausvaiheessa oli riittävä testausta varten, mutta todellista käyttöönottoa varten järjestelmälle 3D-tulostetaan kuvassa 19 esitetyn mukainen kotelo.



**Kuva 19.** GPS-järjestelmän suojakotelo.

Kotelon pohjasta löytyvät kiinnityspisteet, joiden avulla kotelo voidaan kiinnittää helposti ja tukevasti ajoneuvoon. Kotelon kyljessä on reikä kaikille eri johdoille, joita järjestelmän ja moottoripyörän välillä kulkee. Reiät ja muut saumat tiivistetään käyttämällä silikonilla, jolloin kotelon pitäisi pystyä sietämään kosteutta ja pölyä tarpeeksi hyvin.



## 5. YHTEENVETO

Työssä tarkoituksena oli toteuttaa järjestelmä, jonka avulla voidaan ehkäistä tai lieventää moottoripyörään kohdistuvaa ilkivaltaa ja varkausyrityksiä sekä paikantaa moottoripyörä ilkvallan tai varkauden tapahtuessa. Laitteen piti toteuttaa neljä eri tavoitetta: seurata moottoripyörää luvattoman liikkeen varalta, pystyä lähettämään pyörän sen hetkinen sijainti omistajalle tarpeeksi hyvällä tarkkuudella, toimia tarpeeksi pitkään, kun moottoripyörästä irrotetaan akku ja pystyä keskeyttämään asennon tarkkailu mahdollisia huoltotoimenpiteitä tai muita tilanteita varten. GPS-järjestelmää rakennettaessa esiintyi pieni ongelma, jossa GSM-moduuli ei toiminut järjestelmän vaatimalla luotettavuuden tasolla. Ongelma saatiin ratkaistua, kun huomattiin järjestelmässä käytetyn MOSFET-transistorin olevan syy GSM-moduulin käytökseen. Järjestelmän toiminta saatiin vakiinnutettua odotetulla tavalla toimivaksi vaihtamalla käytetty MOSFET-transistori. Pienen järjestelmän muokkauksen jälkeen voitiin suorittaa toiminnallisuustestit. Ensimmäisessä testissä testattiin kaatumisen tunnistus, joka toteutettiin simuloimalla moottoripyörän kaatuminen. Toisessa testissä testattiin järjestelmän toiminta ilman moottoripyörän omaa akkua, jossa järjestelmän pitäisi toimia omaa akkupankkia käyttämällä. Viimeinen testi koostui simuloidusta ajoneuvon varkaudesta, jossa järjestelmän paikannuskykyä testattiin ajamalla testireitti ajoneuvolla. Järjestelmä läpäisi kaikki testit ja niiden perusteella kaikki neljä aikaisemmin mainittua vaatimusta toteutui, joten GPS-järjestelmän voidaan todeta onnistuneeksi.

Taulukossa kaksi on esitettyä GPS-järjestelmän komponenttien muodostama kokonaiskustannus ja niiden alla on samankaltaisten järjestelmien hinnat. Tähdellä (\*) merkittyjen komponenttien tilausmäärä oli minimissään 5 kpl ja sulkeissa on toisen SIM5320E-moduulin hinta mukaan lukien, sillä ensimmäinen moduuli lopetti toimintansa tuntemattomasta syystä järjestelmää testattaessa.

Taulukko 2. GPS-järjestelmän komponenttien hinnat ja vertaus kilpailijoihin.

Komponentti	Määrä (kpl)	Hinta (€)
Arduino Uno	1	18,69
SIM5320E	1 (2)	52,92 (105,84)
Protolevy	1	1,61
Johtimet	2	0,552
LM2574 regulaattori	1	1,31
Operaatiovahvistin	1	0,502
VN2106 MOSFET	1	0,279
RFP50N06 MOSFET	1	1,24
ARE1312 Rele	1	5,01
50kΩ potentiometri	1	0,963
Diodi	5*	1,695
220µF kondensaattori	5*	1,61
22µF kondensaattori	5*	0,535
220µH kela	1	0,979
Pietsosähköinen värähtelijä	1	0,80
<b>Kokonaishinta</b>		<b>88,70 (141,62)</b>
Spy Spot GL300MA [31]		149,00 \$
BikeTrac [32]		299 £

Rakennettu GPS-järjestelmä on hintansa puolesta hyvin kilpailukykyinen muihin myynnissä oleviin samankaltaisiin järjestelmiin verrattuna. Molemmat edellä mainituista kaupallisista järjestelmistä vaativat ostohinnan lisäksi kuukausimaksun, joka aiheuttaa jatkuvia kuluja käyttäjälle. Rakennettu järjestelmä aiheuttaa kuluja käyttäjälle vain silloin, kun järjestelmä lähettää ajoneuvon omistajalle sijaintitietoja. Kaupalliset järjestelmät eivät myöskään tarjoa samanlaajuista laajentamispotentiaalia verrattuna rakennettuun järjestelmään, jonka voi teoriassa laajentaa tukemaan kaikkia ominaisuuksia mitä järjestelmän käyttäjä keksii. Mainitut kaupalliset järjestelmät ovat kuitenkin toiminnaltaan sivistyneempiä, sillä molemmat valmistajat tarjoavat ajoneuvon seurantaan varten hyödyllisen mobiilisovelluksen.

# LÄHTEET

- [1] Shalf, J., Leland, R. & Sandia National Lab. (SNL-NM), Computing beyond Moore's Law, Computer, 2015, 10 p. Saatavilla (viitattu 22.5.2019): <https://ieeexplore-ieee-org.libproxy.tuni.fi/document/7368023>
- [2] Ghadre, D., Programming and customizing the the AVR microcontroller, McGraw-Hill, 2000, 339 p.
- [3] Control Door Lock Remotely Using Smartphone, Arduino project hub, 2017. Saatavilla (viitattu 23.5.2019): <https://create.arduino.cc/projecthub/soadr/control-door-lock-remotely-using-smartphone-a46573>
- [4] Tufoi, M., Vela, I., Marta, C. & Stroia, M. Microcontroller's Applications in Driving of Industrial Robots, Analele Universității "Eftimie Murgu" Reșița: Fascicola I, Inginerie, 2010, 10 p. Saatavilla (viitattu: 23.5.2019): <https://doaj.org/article/54f9da2dd68142d2807c357cd29896d9>
- [5] Crisp, J., Introduction to Microprocessors and Microcontrollers, Amsterdam: Newnes, 2004, 277 p. Saatavilla (viitattu 23.5.2019): <http://search.ebsco-host.com/login.aspx?direct=true&AuthType=cookie.ip.uid&db=nlebk&AN=195647&site=ehost-live&scope=site>
- [6] ATmega328P datalehti, Microchip, 2018, 662 p. Saatavilla (viitattu: 28.01.2019): <http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf>
- [7] Arduino muistityypit, Arduino. Saatavilla (viitattu: 19.02.2019): <https://www.arduino.cc/en/tutorial/memory>
- [8] Arduino flash, Arduino. Saatavilla (viitattu 1.3.2019): <https://playground.arduino.cc/Code/EEPROM-Flash>
- [9] Wheat, D., Arduino internals, Apress, 2011, 386 p. Saatavilla (viitattu 14.3.2019): <https://link-springer-com.libproxy.tuni.fi/book/10.1007%2F978-1-4302-3883-6>
- [10] Tietoa Arduinosta, Arduino. Saatavilla (viitattu 15.3.2019): <https://www.arduino.cc/en/Main/AboutUs>
- [11] Arduino-kehitysalustojen vertailu, Arduino. Saatavilla (viitattu 22.5.2019): <https://www.arduino.cc/en/products/compare>
- [12] Arduino Uno, Arduino. Saatavilla (viitattu 15.3.2019): <https://store.arduino.cc/arduino-uno-rev3>
- [13] SIM5320E datalehti, SIMCom. Saatavilla (viitattu 23.5.2019): <https://simcom.ee/modules/wcdma-hspa/sim5320/>
- [14] AT-komennot, European Telecommunications Standards Institute, 1996, 77 p. Saatavilla (viitattu: 10.02.2019): [https://www.etsi.org/deliver/etsi\\_gts/07/0707/05.00.00\\_60/gsm/0707v050000p.pdf](https://www.etsi.org/deliver/etsi_gts/07/0707/05.00.00_60/gsm/0707v050000p.pdf)

- [15] Henry-Labordère, A., Jonack, V., SMS and MMS Interworking in Mobile Networks, Boston: Artech House Inc, 2004, 334 p. Saatavilla (viitattu 10.4.2019): <http://search.ebscohost.com/login.aspx?direct=true&AuthType=cookie,ip,uid&db=nlebk&AN=121770&site=ehost-live&scope=site&auth-type=sso&custid=s4778523>
- [16] Parkinson, B., Spilker, J., Global Positioning System : Theory and Applications, American Institute of Aeronautics and Astronautics, 1995, 781 p. Saatavilla (viitattu 10.4.2019): <https://ebookcentral.proquest.com/lib/tampere/detail.action?docID=3111663>
- [17] Zandbergen, P., Barbeau, S., Positional Accuracy of Assisted GPS Data from High-Sensitivity GPS-enabled Mobile Phones, The Journal of Navigation, 2011, 20 p. Saatavilla (viitattu 1.3.2019): <https://www.cambridge.org/core/journals/journal-of-navigation/article/positional-accuracy-of-assisted-gps-data-from-high-sensitivity-gps-enabled-mobile-phones/E1EE20CD1A301C537BEE8EC66766B0A9>
- [18] Kazimierczuk M., Pulse-Width Modulated DC-DC Power Converters, New York: John Wiley & Sons, Incorporated, 2015, 1457 p. Saatavilla (viitattu: 1.5.2019): <https://ebookcentral.proquest.com/lib/tampere/detail.action?docID=4040504>
- [19] LM2574 datalehti, Micrel, 2005, 8 p. Saatavilla (viitattu 1.3.2019): <http://ww1.microchip.com/downloads/en/DeviceDoc/lm2574.pdf>
- [20] Goodenough, J., Park, K., The Li-Ion Rechargeable Battery: A Perspective, Journal of the American Chemical Society, 2013, 10 p. Saatavilla (viitattu 23.5.2019): <https://pubs-acscs.org.libproxy.tuni.fi/doi/10.1021/ja3091438#>
- [21] Battery Charging Specification, USB Implementers Forum, 2012, 71 p. Saatavilla (viitattu: 28.01.2019): <https://www.usb.org/document-library/battery-charging-v12-spec-and-adopters-agreement>
- [22] Overview of Lithium Ion Batteries, Sanyo, 2016, 29 p. Saatavilla (viitattu 8.5.2019): [http://www.rathboneenergy.com/articles/sanyo\\_lionT\\_E.pdf](http://www.rathboneenergy.com/articles/sanyo_lionT_E.pdf)
- [23] Siddique, A., Mahmud, S., & Heyst, B., A comprehensive review on vibration based micro power generators using electromagnetic and piezoelectric transducer mechanisms, University of Guelph , 2015, 20 p. Saatavilla: <https://www.science-direct.com/science/article/pii/S0196890415009164>
- [24] Signal Conditioning Piezoelectric Sensors, Texas Instruments, 2000, 6 p. Saatavilla (viitattu 18.4.2019): <http://www.ti.com/lit/an/sloa033a/sloa033a.pdf>
- [25] MCP602 datalehti, Microchip, 2007, 34 p. Saatavilla (viitattu 1.3.2019): <http://ww1.microchip.com/downloads/en/DeviceDoc/21314g.pdf>
- [26] Handbook of Operational Amplifier Applications, Texas Instruments, 2016, 94 p. Saatavilla (viitattu 18.4.2019): <http://www.ti.com/lit/an/sboa092b/sboa092b.pdf>
- [27] Adafruit\_FONA -kirjasto. Saatavilla (viitattu 3.5.2019): [https://github.com/adafruit/Adafruit\\_FONA](https://github.com/adafruit/Adafruit_FONA)
- [28] Low\_Power -kirjasto. Saatavilla (viitattu 3.5.2019): <https://github.com/rocketscream/Low-Power>

- [29] VN2106 datalehti, Supertex inc., 2008, 5 p . Saatavilla (viitattu 1.3.2019): <http://ww1.microchip.com/downloads/en/DeviceDoc/vn2106.pdf>
- [30] RFP50N06 datalehti, ON semiconductor, 2002, 10 p. Saatavilla (viitattu 2.5.2019): <https://www.onsemi.com/pub/Collateral/RFP50N06-D.pdf>
- [31] Spy Spot GL300MA, Saatavilla (viitattu 22.5.2019): <https://shop.spy-spot.com/new-4g-lte-gl300ma-micro-gps-tracker-real-time.html>
- [32] BikeTrac. Saatavilla (viitattu 22.5.2019): <https://biketrac.co.uk/>

## LIITE A: OHJELMAKOODI

```

#include "Adafruit_FONA.h"
#include "LowPower.h"
#include <SoftwareSerial.h>

// Define SIM5320E data pins. Reset is not used,
// but needed by the Adafruit_FONA library.
const int SIM_RS = 9;
const int SIM_RX = 8;
const int SIM_TX = 7;

// Pins used to monitor and control the system.
const int POWER_ON = 6;
const int BYPASS = 5;
const int MOSFET = 4;

// Interrupt pin for the piezoelectric transducer.
const int PIEZO = 2;

// Set digital pins to be used for serial communication.
SoftwareSerial SIM = SoftwareSerial(SIM_TX, SIM_RX);

// Adafruit_FONA library needs a serial port pointer to be given to it,
// so create a pointer for the previously created software serial port.
SoftwareSerial *SimSerial = &SIM;

// Initialize fona class for use.
Adafruit_FONA_3G fona = Adafruit_FONA_3G(SIM_RS);

// Variables for use when finding location using GPS.
float latitude, longitude, speed_kph, heading, speed_mph, altitude;
bool gps_success;

// Set the number of the vehicle's owner.
const String ownerNum = "AT+CMGS=\"0408674704\"";

// Link to google maps into which latitude and longitude coordinates
// are added to get a link directly to the location of the vehicle.
const String mapsLink = "http://www.google.com/maps/place/";

// Time interval (in ms) for message sending. Currently set to 2,5 min.
const unsigned long INTERVAL = 150000;

// Other variables used to aid the program.
unsigned long timer = 0;
unsigned long currentMs = 0;
unsigned long previousMs = 0;
volatile bool vibration = false;

// Setup is run once on boot to initialize e.g.
// digital pins or the interrupt service.
void setup()
{
  // Start serial communication with baud speed of 4800.
  Serial.begin(4800);
  SIM.begin(4800);

```

```

// Set right mode for different pins.
pinMode(MOSFET, OUTPUT);
pinMode(BYPASS, INPUT_PULLUP);
pinMode(POWER_ON, INPUT);

// Set piezoelectric element to cause an interrupt.
attachInterrupt(digitalPinToInterrupt(PIEZO), vibrationDetected, RISING);

// Turn system off to save power.
digitalWrite(MOSFET, LOW);
LowPower.powerDown(SLEEP_FOREVER, ADC_OFF, BOD_OFF);
}

// Code written inside the function is meant to run as long as the
// microcontroller has power. Used to implement the main code that
// is run by the system.
void loop()
{
// If either the bypass switch is on or the vehicle is turned on,
// the system won't react to the interrupt signal sent by the
// piezoelectric element.
if ((digitalRead(BYPASS) == LOW) || (digitalRead(POWER_ON) == HIGH)) {

// Set variables to default values and put system to sleep.
timer = 0;
currentMs = 0;
previousMs = 0;
vibration = false;
digitalWrite(MOSFET, LOW);
LowPower.powerDown(SLEEP_FOREVER, ADC_OFF, BOD_OFF);
}

else {

// True if there has been a strong enough vibration that has been
// registered by the piezoelectric element.
if (vibration) {

// Send the first SMS right after the vibration.
if (timer == 0) {
getLocation();
sendSMS();
}

// Counts how much time has passed since last SMS was sent.
currentMs = millis();
timer = timer + (currentMs - previousMs);
previousMs = currentMs;

// For every 2,5 minutes, send a new message until the bypass
// switch is pressed or the vehicle is turned on.
if (timer > INTERVAL) {
getLocation();
sendSMS();
timer = 1;
}
}
}
}

```

```

    }
}

void getLocation()
{
    // Turn SIM5320E back on, if waking the system up from sleep.
    if (digitalRead(MOSFET) == LOW) {
        digitalWrite(MOSFET, HIGH);

        // Give SIM5320E 15 seconds to boot and initialize itself.
        delay(15000);
    }

    gps_success = false;

    // Tries to establish a serial communication line between the
    // microcontroller and the SIM5320E module.
    if (! fona.begin(*SimSerial)) {
        while (1);
    }

    // Turn on GPS within the module
    fona.enableGPS(true);

    // Loops until the module gets an accurate location of the vehicle.
    while(!gps_success){
        delay(2000);
        gps_success = fona.getGPS(&latitude, &longitude, &speed_kph,
                                &heading, &altitude);
    }
}

// sendSMS() takes the location, speed and heading of the vehicle and
// writes a message containing the aforementioned data.
void sendSMS()
{
    // Set the SIM5320E to SMS text mode.
    SIM.println("AT+CMGF=1");
    delay(500);

    // Set the number to which the message is sent to.
    SIM.println(ownerNum);
    delay(500);

    // Create a link to google maps based on the coordinates.
    SIM.print/mapsLink);
    SIM.print(latitude, 6);
    SIM.print(",");
    SIM.println(longitude, 6);
    delay(500);

    // Insert current speed of the vehicle and heading to the message.
    SIM.print("Speed: ");
    SIM.println(speed_kph);
    SIM.print("Heading: ");
    SIM.print(heading);
    delay(500);
}

```



```
// Send ASCII character 26, which is a control character to signify
// the end of the SMS message and to send the message.
SIM.println((char)26);
delay(500);
}

// vibrationDetected() is executed when the piezoelectric element
// shakes/vibrates with enough force to cause an interrupt and
// wake the microcontroller from sleep.
void vibrationDetected()
{
  vibration = true;
}
```