

Teemu Saarimaa

Hitsausajan seurantalaitte

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Sähkö- ja automaatiotekniikka

Insinöörityö

13.3.2018

Tekijä Otsikko	Teemu Saarimaa Hitsausajan seurantalaitte
Sivumäärä Aika	39 sivua 13.3.2018
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Sähkö- ja automaatiotekniikka
Ammatillinen pääaine	Automaatiotekniikka
Ohjaajat	Lehtori Kristian Junno Hallintopäällikkö Jetro Tilhe
<p>Tämä opinnäytetyö toimi kehitysprojektina Viafin Oy:lle. Kehitysprojektissa oli tarkoitus luoda laite, joka nauhoittaa ja tallentaa työntekijöiden suorittamaa hitsaustyötä. Nauhoitettu data siirtyy tietokantaan, ja tämä data luetaan tietokannasta ja se esitetään visualisoinnissa kuvaajina, lukuina ja taulukoina. Laitteen keräämää dataa on tarkoitus käyttää mm. työnajan seurannassa sekä jossain määrin myös työn laadun valvonnassa.</p> <p>Projektissa kehiteltiin ensiksi ohjelmisto, jolla hitsaustyötä pystyttiin nauhoittamaan sekä visualisoimaan. Kun ohjelmisto oli valmis, keskityttiin sen jälkeen etsimään laitteelle sopivat komponentit. Sopivat komponentit selvitettiin suunnittelutyöllä sekä käytännön testauksilla. Komponenttien tutkimisen jälkeen luotiin laitekokonaisuus, josta laite tulisi koostumaan, jonka jälkeen laitteita rakennettiin isompi määrä kerralla. Lopuksi kaikki laitteet testattiin konepajalla käytännön olosuhteissa.</p> <p>Kehitysprojektin lopputuloksena saatiin kehitettyä hitsausajan seurantalaitte, joka on yksinkertainen ja helppokäyttöinen. Laite toimii langattomasti, se on jokaiselle työntekijälle henkilökohtainen, ja se kerää dataa työntekijöiden suorittamasta hitsaustyöstä. Tätä dataa voidaan hyödyntää yrityksen sisäisissä asioissa. Myöhemmin laitetta optimoitiin, jotta sen kustannukset saataisiin pienemmiksi. Komponenttien optimoinnilla laitteen kustannukset saatiin laskettua tasolle, jolloin siitä saatiin hyvin edullinen.</p>	
Avainsanat	IoT, Raspberry Pi, Node-RED, InfluxDB, Grafana

Author Title	Teemu Saarimaa Welding Monitoring Device
Number of Pages Date	39 pages 13 March 2018
Degree	Bachelor of Engineering
Degree Programme	Electrical and Automation Engineering
Professional Major	Automation Engineering
Instructors	Kristian Junno, Senior Lecturer Jetro Tilhe, Head of Administration
<p>This study was done as a development project for Viafin Oy. The purpose of this project was to create a device that records and stores information about welding work done by workers. Recorded data is transferred into a database, and this data is read from the database and visualization shows this information as graphs, numbers and tables. Data gathered by the device can be used in e.g. monitoring of working hours and quality of work to some extent.</p> <p>First, a software was developed for this project so welding work could be recorded and visualized. When the software was ready, the focus was shifted to finding the optimal components for this device. This was done by planning and practical testing. A configuration was made after the researching of components was finished, and larger number of devices were made using this configuration. Finally, each device was tested in a workshop in a practical environment.</p> <p>The result of the development project was a welding monitoring device that is simple and easy to use. The device operates wirelessly, it is personal for every worker and it collects data on the welding work performed by workers. This data can be used in company's internal affairs. Later the device was optimized so that its expenses could be lowered. By optimizing the components used in the device, the expenses were lowered to a level that made the device very affordable.</p>	
Keywords	IoT, Raspberry Pi, Node-RED, InfluxDB, Grafana

Sisällys

Lyhenteet

1	Johdanto	1
1.1	Projektin tavoitteet	1
1.2	Viafin	2
2	Taustatyö	3
2.1	Ohjausyksikkö	3
2.2	Anturi	6
2.3	Virtalähde	8
2.4	Kotelointi	9
2.5	Ohjelmisto	12
3	Alustavat testaukset	13
3.1	Menetelmät	13
3.2	Havainnot, ongelmat, sekä niiden ratkaisut	15
4	Toteutus	18
4.1	Laitteisto	23
4.2	Ohjelmisto	25
4.2.1	Raspberry Pi	25
4.2.2	InfluxDB ja Grafana	25
4.2.3	Etäyhteys	28
4.3	Kokonaisuus	29
5	Kenttätestit	31
5.1	Testien havainnot	32
5.2	Muut havainnot	34
5.3	Tulokset ja jatkokehitys	35
6	Yhteenveto	37
6.1	Pohdinta	37
6.2	Tulokset	38
	Lähteet	39

Lyhenteet

CAD	<i>Computer Aided Design</i> , tietokoneavusteinen suunnittelu.
GND	<i>Ground</i> , maa tai maadoitus.
GPIO	<i>General Purpose Input Output</i> , Raspberry Pi -tietokoneessa olevat I/O-nastat.
I/O	<i>Input/Output</i> , siirräntä. Tiedon siirtämistä elektronisten laitteiden tai niiden komponenttien välillä.
IoT	<i>Internet of Things</i> , esineiden Internet. Tunnetaan myös nimellä teollinen Internet. Tarkoitetaan Internetin laajentumista laitteisiin ja koneisiin, jota kautta niitä voidaan mitata, valvoa ja ohjata.
IP	<i>Internet Protocol</i> , Internet-kerroksen protokolla.
LED	<i>Light Emitting Diode</i> , hohtodiodi, valoa säteilevä diodi.
SSH	<i>Secure Shell</i> , salattuun tietoliikenteeseen tarkoitettu protokolla.

1 Johdanto

1.1 Projektin tavoitteet

Yhtenä Viafin IoT-kehitysprojektin (Internet of Things) tavoitteena oli kehittää laite, jolla voidaan nauhoittaa ja seurata työntekijöiden suorittamaa hitsaustyötä. Kehitysprojektissa luotavan laitteen tuli toimia langattomasti, ja sen nauhoittama data tuli olla helposti luettavissa. Laitteiden oli tarkoitus olla jokaiselle työntekijälle henkilökohtainen, ja sen tuli lähettää keräämänsä dataa yrityksen serverikoneen tietokantaan. Serverikoneella oleva visualisointiohjelma hakee tiedot tietokannasta ja esittää ne graafisesti kuvaajina ja numeerisina arvoina. Projektin tavoite oli kerätä dataa työntekijöiden suorittamasta hitsaustyöstä sekä näiden työn laadusta, ja kerättyä dataa oli tarkoitus käyttää yrityksen sisäisesti tehokkuuden ja laadun parantamiseksi.

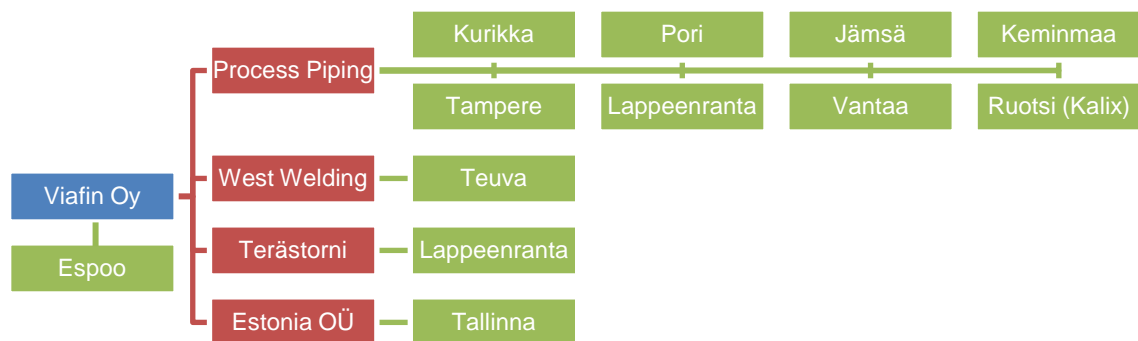
Laitteen toteutustapa oli vapaa, sekä sen kehittämiseen oli annettu hyvin resursseja. Laite muutti useasti muotoaan projektin edetessä, ja suunnittelutyötä tehtiin jatkuvasti toteutuksen lomassa. Laitteella tehtiin kenttätestejä Viafinin omilla konepajoilla, ja kentältä kerätty kokemus antoi uusia ideoita sille, mihin suuntaan tätä projektia täytyi viedä.

Projektissa käytettiin avoimen lähdekoodin ratkaisuja, mikä loi eri lähestymistapoja projektin suorittamiselle. Avoin lähdekoodi mahdollisti myös sen, että saatavilla olevat ohjelmistot olivat vapaasti saatavilla, monipuolisia, sekä myös täysin muokattavissa omien tarpeiden mukaiseksi [1].

1.2 Viafin

Viafin Oy perustettiin vuonna 2006, mutta sen historia ulottuu pidemmälle 1960-luvulle, kun Terästorni Oy perustettiin Lappeenrantaan vuonna 1968. Se teki vaativia terästuotteita mm. metsä-, kemian- ja prosessiteollisuuteen sekä myöhemmin myös vientitoimintuksia. West Welding Oy perustettiin Teuvalla vuonna 1992. Siellä valmistettiin prosessiteollisuuden painesäiliöitä, joita käytetään mm. öljynporauslautoilla, öljynjalostamoilla sekä metsäteollisuuden tuotantolaitoksilla. Viafin Oy yhdisti nämä kaksi yritystä Viafin-konsernin alle, ja samaan konserniin kuuluu myös Viafin Process Piping sekä Viafin Estonia OÜ. [2.] Kuvassa 1 on havainnollistettu konsernin yritysraakenne.

- Perustamisvuosi 1968 (Terästorni), 1992 (West Welding)
- Liikevaihto 60 M€
- Henkilöstö 280 hlöä
- Toimipaikat 10 kpl
 - Viafin Oy: Suomi (Espoo)
 - Viafin Process Piping: Suomi (Kurikka, Pori, Jämsä, Keminmaa, Tampere, LPR, Vantaa) Ruotsi (Kalix)
 - Viafin West Welding: Suomi (Teuva)
 - Viafin Terästorni: Suomi (Lappeenranta)
 - Viafin Estonia OÜ: Viro (Tallinna)



Kuva 1. Viafin-konsernin yritysraakenne

2 Taustatyö

Projektia lähdettiin kehittämään sen jälkeen, kun projektin idea ja tavoitteet olivat selvillä. Taustatyöhön kuului muun muassa ohjausyksiköiden, anturien, virtalähteiden sekä koteloinnin tutkimista. Tavoitteena oli löytää sellaiset komponentit, jotka täyttivät tarvittavat vaatimukset, sekä olivat myös kustannuksiltaan edullisia. Taustatyötä tehtiin sekä pitkän esisuunnittelun että käytännön kokeilujen kautta. Laitte pyrittiin pitämään mahdollisimman yksinkertaisena niin ulkomuodoltaan kuin toiminnollisuuksiltaan.

Hitsaustyötä aiottiin seurata siten, että anturilla mitataan maakaapelissa kulkevaa virtaa. Kun työkohteena olevaa kappaletta hitsataan, niin silloin hitsattavan kohteen ja hitsauslaitteen välisessä maakaapelissa kulkee virtaa, ja tämä virta pystytään havaitsemaan tietyillä antureilla. Tässä projektissa ei mitattu virran suuruutta, vaan pelkästään se, että kulkeeko kaapelissa virtaa vai ei.

2.1 Ohjausyksikkö

Laitteen ohjausyksikölle oli monia eri vaatimuksia. Ensinnäkin sen oli pystyttävä vastaanottamaan sekä käsittelemään anturin lähettämää dataa, sekä tekemään sen avulla tarvittavat laskelmat. Ohjausyksikön nauhoittama data oli myös pystyttävä lähettämään langattomasti serverille. Muita huomioon otettavia ominaisuuksia olivat mm. virrankulutus, kustannukset ja fyysinen koko. Ohjausyksiköksi valittiin Raspberry Pi, joka on yhden piirilevyn tietokone. Tässä projektissa käytettiin kahta Raspberry Pi -mallia, jotka löytyvät kuvasta 2.

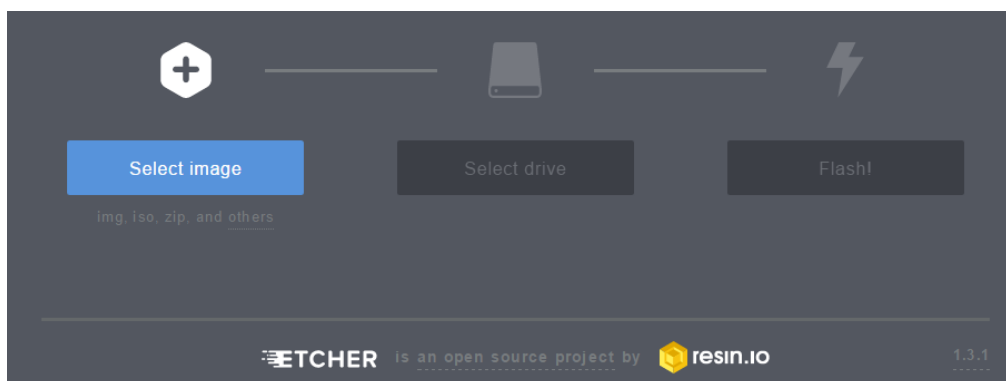
Raspberry Pi on brittiläisen Raspberry Pi Foundationin kehittämä yhden piirilevyn tietokone, joka luotiin ohjelmoinnin opetuskäyttöön sekä käytettäväksi elektroniikan projekteissa. Sitä voidaan käyttää kuin tavanomaista PC:tä. Laitteen mallista riippuen siihen on mahdollista kytkeä kaikki mitä tavalliseen tietokoneeseenkin: näppäimistö, hiiri, ääntoisto, näyttö sekä Ethernet-kaapeli Internet-yhteyttä varten. Uusimmissa malleissa on myös Wi-Fi, joka mahdollistaa langattoman verkkoyhteyden. [3; 4.]



Kuva 2. Raspberry Pi -tietokoneita. Vasemmalla Raspberry Pi 3 Model B, oikealla Raspberry Pi Zero [4].

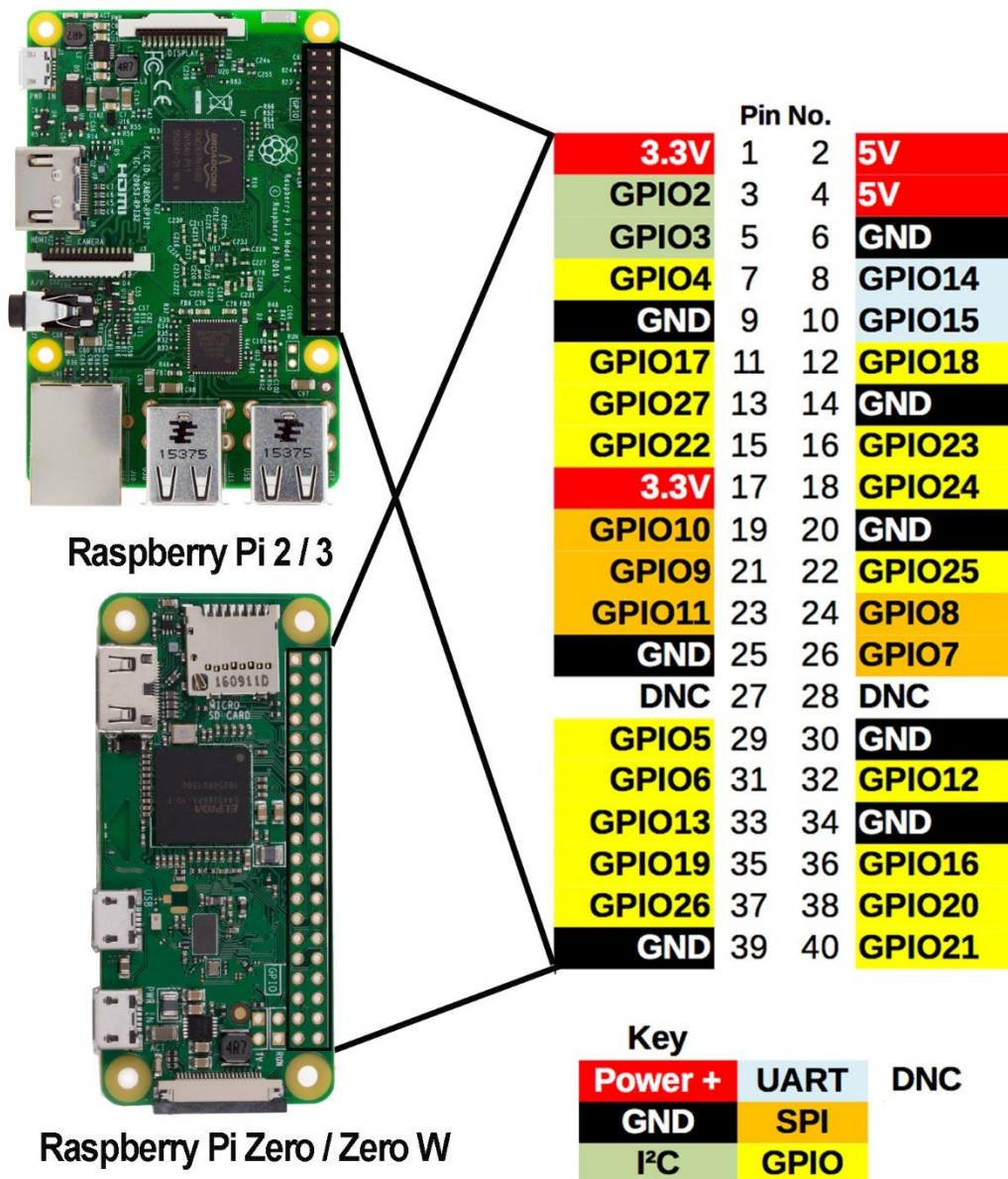
Raspberry Pin edut tulevat esiin siinä, että se on hyvin kompakti paketti sekä edullinen siihen nähden, mitä sillä on mahdollista tehdä. Se saa käyttäjännitteensä MicroUSB-liitännästä, ja sen käyttöjärjestelmien valikoima on laaja. Tässä projektissa käyttöjärjestelmänä oli valmistajan suosittelu Raspbian, joka on Debian-pohjainen Linux-käyttöjärjestelmä. Käyttöjärjestelmä sekä muu tallennustila sijaitsevat irrotettavalla microSD-muistikortilla. Tämä tarkoittaa myös sitä, että Raspberry Pi vaatii aina käyttöjärjestelmällä varustetun muistikortin käynnistyäkseen.

Raspbian-käyttöjärjestelmän voi ladata ilmaiseksi Raspberry Pin virallisilta verkkosivuilta, ja se asennetaan muistikortille käyttäen erillistä ohjelmaa [5]. Tässä projektissa siihen käytettiin Etcher-nimistä ohjelmaa, jonka käyttöliittymä on kuvassa 3. Ohjelmassa valitaan ensiksi levytila, jonne käyttöjärjestelmä asennetaan. Sen jälkeen valitaan käyttöjärjestelmän image-tiedosto, jonka jälkeen asennus voi alkaa. Asentaminen kesti noin vartin verran, jonka jälkeen muistikortti voitiin sijoittaa Raspberry Pihin. Käynnistyksen jälkeen laite oli heti käyttövalmis.



Kuva 3. Etcher-ohjelman pääikkuna

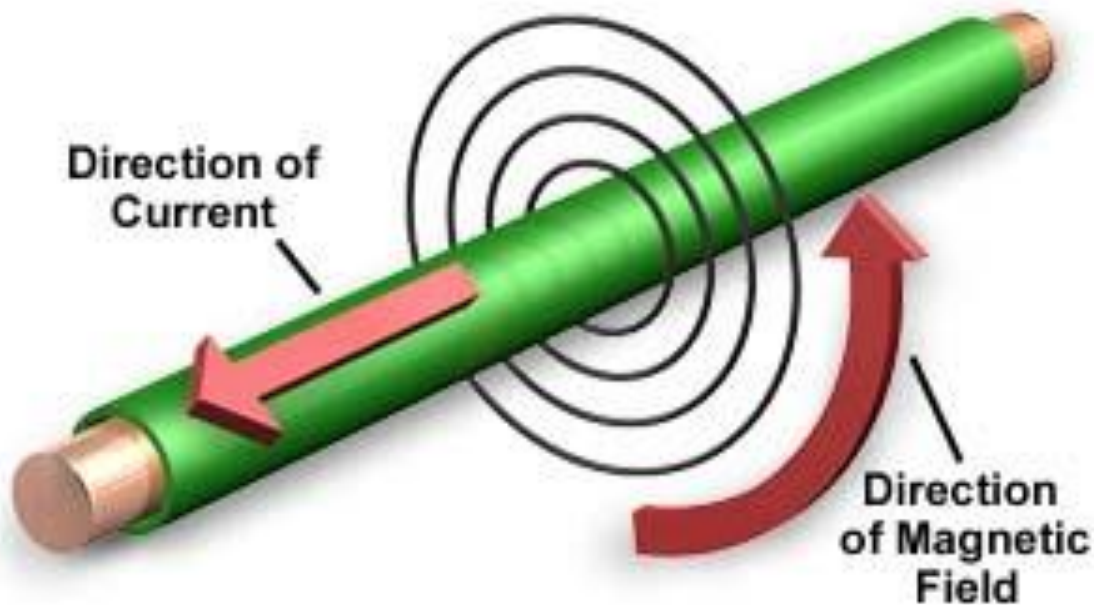
Raspberry Pi -tietokoneista löytyy GPIO (General Purpose Input Output), jonka kautta Raspberry Pi voi vastaanottaa sekä lähettää signaaleja. Näiden sijainti Raspberry Pi:ssä on havainnollistettu kuvassa 4. GPIO-nastoja voidaan ohjelmoida tarpeen mukaan. Sen eri nastat ovat eri käyttötarkoitusta varten, ja tietyt anturit voidaan kytkeä vain tiettyihin nastoihin. GPIO on hyvin monipuolinen: siihen voidaan liittää tavallisten kytkimien, nappien ja valojen lisäksi mm. lämpötila-, kiihtyvyyss- tai asentoantureita, sekä sillä voidaan ohjata tasasähkömoottoreita. [6.] Tässä projektissa GPIO-nastoihin liitettiin anturi, painonappi sekä LED (Light Emitting Diode).



Kuva 4. GPIO eri Raspberry Pi -malleissa [7]

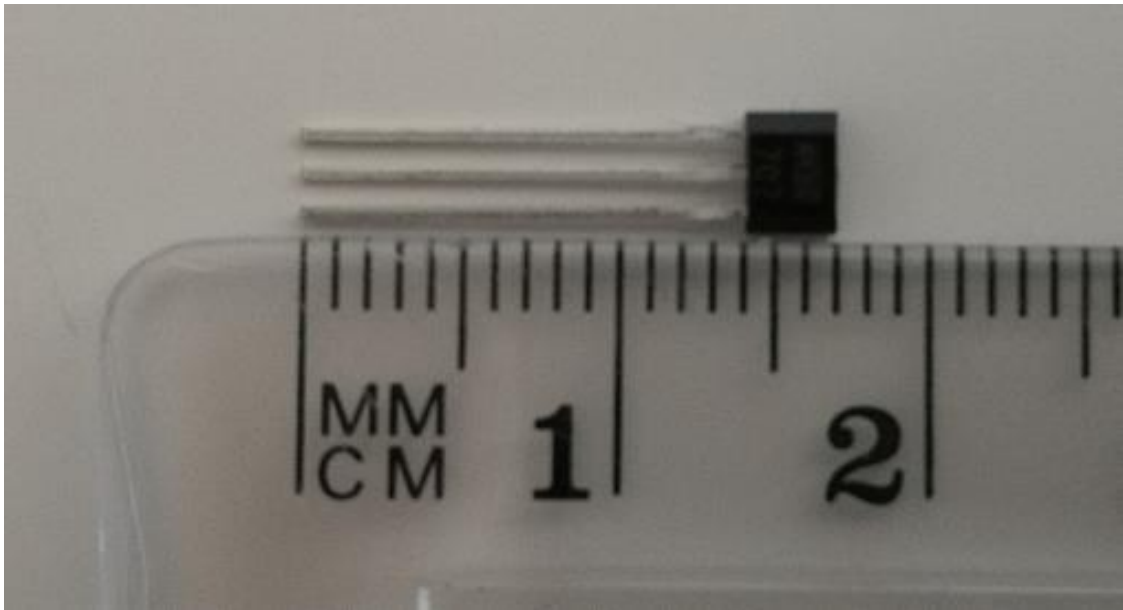
2.2 Anturi

Kun maakaapelissa kulkee hitsauksen aikana virtaa, virta luo ympärilleen magneettikentän, joka on riippuvainen virran suuruudesta. Tämä tunnetaan nimellä Amperén laki, joka on havainnollistettu kuvassa 5. [8.] Virran synnyttämä magneettivuon tiheys pystytään havaitsemaan tietyillä antureilla, ja tässä projektissa siihen käytettiin magnetoresistiivisiä anturia. Anturi kiinnitetään maakaapeliin, ja kun kaapelissa kulkee virtaa, niin anturi havaitsee magneettikentän ja antaa signaalin. Virran suunta määrittää myös magneettikentän suunnan, mutta tässä tapauksessa sillä ei ollut merkitystä.



Kuva 5. Amperén laki [8]

Hitsauksessa käytettävät virrat vaihtelevat suuresti, mutta ne ovat yleensä monia kymmeniä ampeereja. Anturi oli valikoitava sen mukaan, että sen oli kyettävä tunnistamaan pienetkin hitsausvirrat. Magnetoresistiiviset anturit eivät voi rikkoutua liian suurista magneettivuon tiheyksistä, joten isot virrat eivät vaikuttaneet anturin valintaan. Ainoa asia millä oli väliä mittauksen kannalta, oli että kuinka pieni virta kaapelissa kulkee. Laitteessa käytettävä anturi valikoitiin myöhemmin käytännön testien jälkeen.



Kuva 6. Magnetoressistiivinen anturi

Anturin on avautuva kytkin ja sen käyttöjännite on välillä 3—24 voltia ja virrankulutus milliampeerien luokkaa, eli sen pystyi liittämään suoraan Raspberry Pin GPIO-nastoihin. Anturin antama signaali kulkee kaapelia pitkin ohjausyksikköön, joka käsittelee saadun signaalin. Kuva 6 havainnollistaa, että itse anturi on hyvin pieni komponentti. Sen takia sille täytyi olla oma kaapelinsa, jotta sen signaali voisi kulkea ohjausyksikölle asti. Kaapeliksi valittiin datasiirtoon tarkoitettu kuparilankapunoksella suojattu kaapeli, jonka toiseen päähän anturi juotettiin kiinni ja toiseen päähän kiinnitettiin liittimet.

Anturi täytyi myös pystyä kiinnittämään maakaapeliin. Kiinnitystavalla oli muutamia vaatimuksia. Sen täytyi olla mahdollisimman tiiviisti kiinni maakaapelissa, sillä magneettivuoro heikkenee nopeasti etäisyyden kasvaessa. Sen täytyi myös olla kohtisuorassa kaapelia kohden, sillä ainoastaan tällä tavoin anturin tunnistusherkkyyks oli suurin. Jos anturi oli vähänkin vinossa, niin se ei välttämättä tunnistanut kaapelissa kulkenutta virtaa. Hitsausvirran ollessa pieni tällä oli erityisen suuri merkitys. Kiinnityksen täytyi myös olla jyrkevä, kestävä sekä helppokäyttöinen. Projektissa kokeiltiin erilaisia kiinnitystapoja, mutta vasta käytännön testauksien jälkeen saatiin lopulta selville, minkälaista kiinnitystapaa anturille on syytä käyttää.

2.3 Virtalähde

Ohjausyksikkö vaati oman erillisen virtalähteensä, jonka ominaisuudet täytyi myös selvittää tässä projektissa. Helppoisimpana ja käytännönläheisempänä ratkaisuna oli käyttää varavirtalähdettä. Sen avulla laitteelle pystyi syöttämään riittävän määrän virtaa, sen uudelleen lataaminen oli helppoa, ja sitä oli saatavilla eri kokoisina niin kapasiteetiltaan kuin fyysiseltä kooltaan. Näiden seikkojen ansiosta voitiin valita juuri sellainen virtalähde, mitä tämä projekti vaati.

Varavirtalähteen tarkan mallin valikointi vaati käytännön testausta. Oli selvitettävä, kuinka kauan virtalähde pystyi syöttämään virtaa laitteille ennen kuin akku loppui. Tämä testattiin yksinkertaisesti jättämällä laite päälle niin kauaksi aikaa, kunnes akku kului loppuun.

Alussa käytössä oli Ankerin 10 400 mAh:n varavirtalähteet, mikä näkyy kuvassa 7. Nämä varavirtalähteet pitivät laitteen toiminnassa noin 13 tunnin ajan. Tämä oli riittävä, sillä konepajoilla voidaan tehdä pitkiäkin päiviä, ja laitteen täytyisi kestää yhdellä latauksella koko työpäivän ajan. Tästä huolimatta varavirtalähteen kapasiteettia päätettiin nostaa, jotta varaus riittäisi varmasti koko päivän ajan.



Kuva 7. Projektin alkuvaiheessa käytetyt Ankerin 10 400 mAh:n varavirtalähteet

Varavirtalähteeltä vaadittiin myös tietynlaista autonomisuutta. Kaikki komponentit, mukaan lukien varavirtalähde, tulisivat lopulta olemaan kotelon sisällä. Käyttäjän ei tulisi missään vaiheessa koskea varavirtalähteeseen, vaan sen oli pystyttävä syöttämään virtaa jatkuvasti.

Eri valmistajilta kysyttiin niiden virtalähteiden ominaisuuksia. Tärkeimpinä kriteereinä olivat virtalähteen mitat, kapasiteetti sekä kyky autonomisuuteen. Kyselyiden ja vertailujen tuloksena päädyttiin tilaamaan 20 000 mAh:n varavirtalähteitä, jotka tulisivat myös olemaan sopivan kokoisia tulevaa koteloa varten.

2.4 Kotelointi

Laitteen lopullinen ulkonäkö sekä fyysinen koko olivat myös yksi projektin olennainen seikka. Koska laitteen tuli olla langaton sekä sen täytyi kulkea aina mukana, sen fyysiseen kokoon oli kiinnitettävä huomiota. Laitetta tullaan käyttämään konepajaolosuhteissa, joten sen täytyi myös kestää vaativaa ympäristöä. Kotelo oli suunniteltava siten, että laitetta olisi mahdollisimman yksinkertainen käyttää, eikä sen käyttäminen vaatisi määräänsä enempää aikaa tai opettelua.

Käyttöympäristö asetti vaatimuksia kotelon tiiviydelle sekä iskunkestävyydelle. Tämä vaikutti myös siihen, että mitä kaikkea toimintoja laitteessa voi olla, sillä kaikkien kotelon pinnalla sekä sen ulkopuolella olevien komponenttien tulisi myös sietää tätä ympäristöä. Tästä syystä laitteen ulkomuoto haluttiin pitää mahdollisimman minimaalisena: laite sisälsi ainoastaan LED-valolla varustetun virtanapin, latausportin, anturikaapelin sekä läpivedon anturikaapelille. Kotelossa täytyi olla riittävästi tilaa komponentteja, kaapeleita ja johtoja varten. Se ei kuitenkaan saisi olla liian iso, sillä laitteen oli tarkoitus olla mahdollisimman helppokäyttöinen.

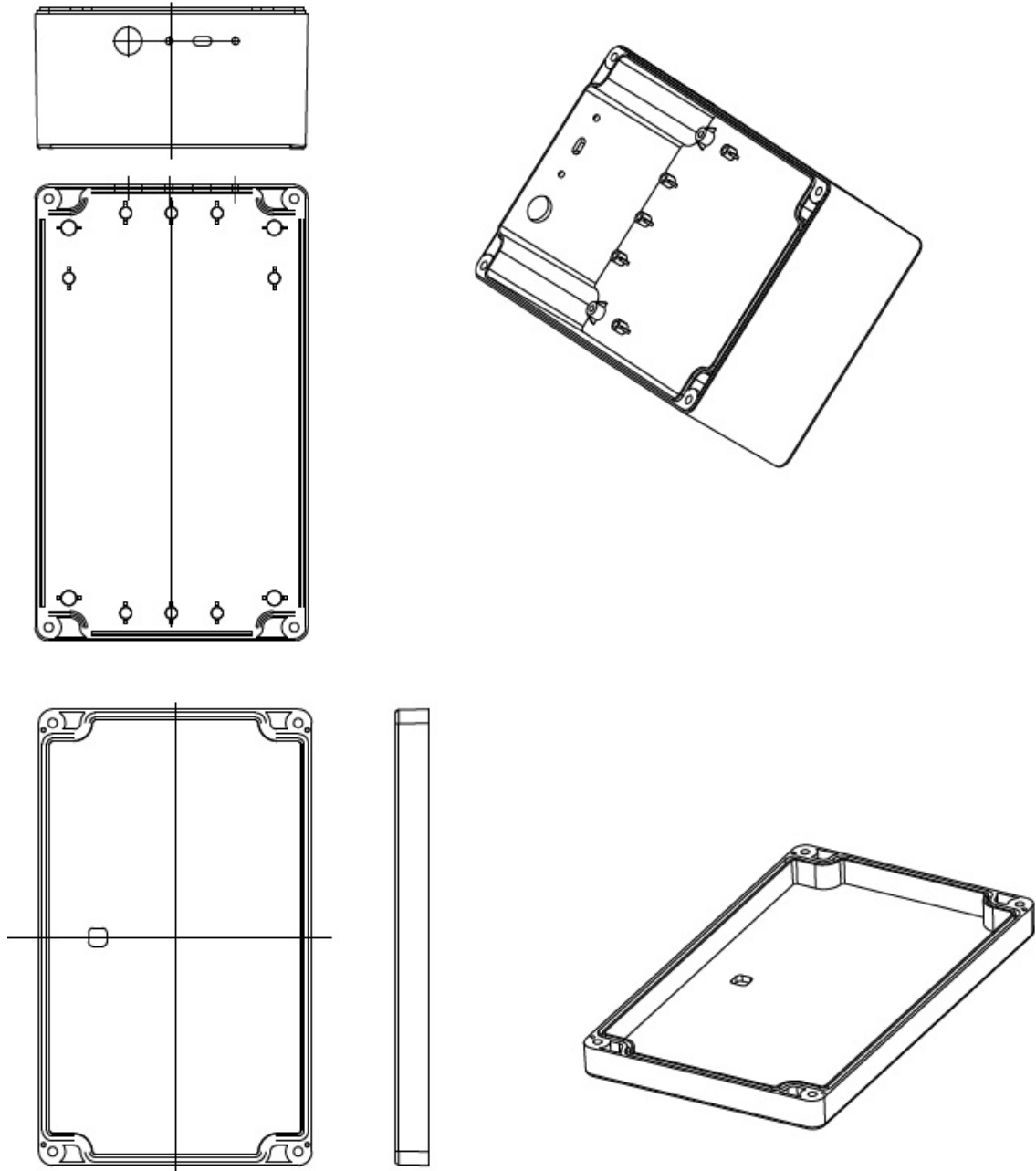
Ennen lopullisten koteloiden tilaamista laitteita testattiin väliaikaisissa koteloissa, jotka ovat kuvattuna kuvassa 8. Koteloihin porattiin reiät anturikaapelin läpivientiä ja latausporttia varten. Väliaikaisen kotelon käytöllä selvitettiin myös, miten paljon tilaa laitteen komponentit vievät ja mitkä olisivat lopullisen kotelon mitat.



Kuva 8. Laitteet niiden väliaikaisissa koteloidissa

Markkinoilla on saatavilla eri kokoisia koteloidia, ja näitä voidaan tehdä tilaustyönä asiakkaan toiveiden mukaiseksi. Kun sopivan kokoinen kotelo löytyi, otettiin yhteyttä valmistajaan, joka toimitti kotelon CAD-kuvat (Computer Assisted Drawing). Tässä työssä oli käytössä ohjelma nimeltä CADS 17, jota käytettiin opiskelijalisenssillä.

CAD-kuviin lisättiin reikien sijainnit latausportille, anturikaapelin läpiviennille sekä virtanapille. Kuvassa 9 on CAD-kuva koteloista, jotka tilattiin tähän projektiin. Kotelon kansi valittiin läpinäkyväksi sekä siihen tuli Viafinin logo. Läpinäkyvä kansi haluttiin siksi, jotta nähtäisiin sisällä olevan varavirtalähteen varaustaso. Kotelon kotelointiluokitus oli IP65.



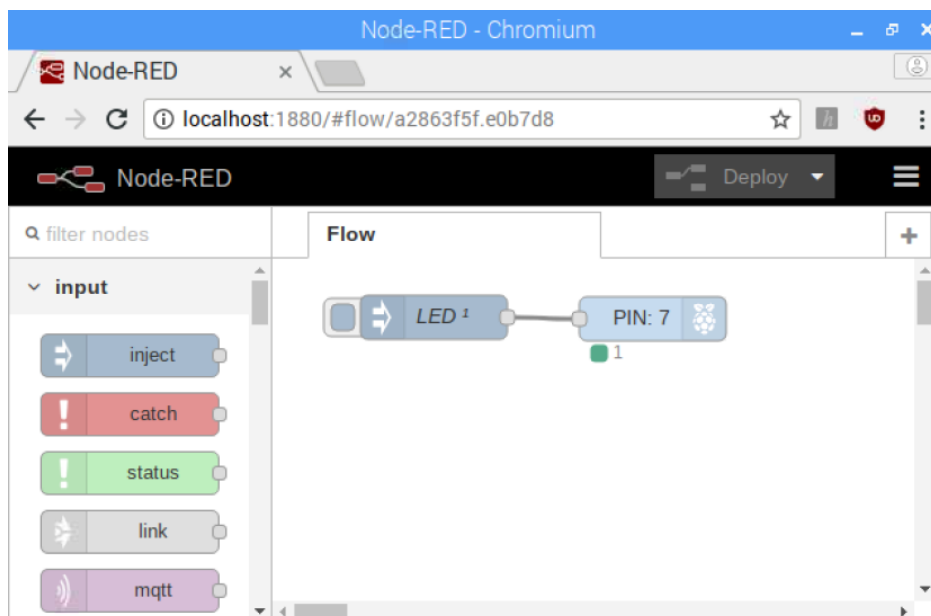
Kuva 9. CAD-piirros uusista kotelosta sekä reikien sijainnista

2.5 Ohjelmisto

Raspberry Pi:ssä oli asennettuna Raspbian-käyttöjärjestelmä, jonka päällä kaikki muut ohjelmat ja koodit toimivat. Projektin alussa siihen asennettiin pelkästään Node-RED hit-saustyön nauhoitusta varten. Node-RED on IBM:n kehittämä flow-pohjainen ohjelmointityökalu, jonka lähdekoodi tehtiin avoimeksi vuonna 2013. Nykyään ohjelman kehityksestä huolehtii JS-foundation. [9.]

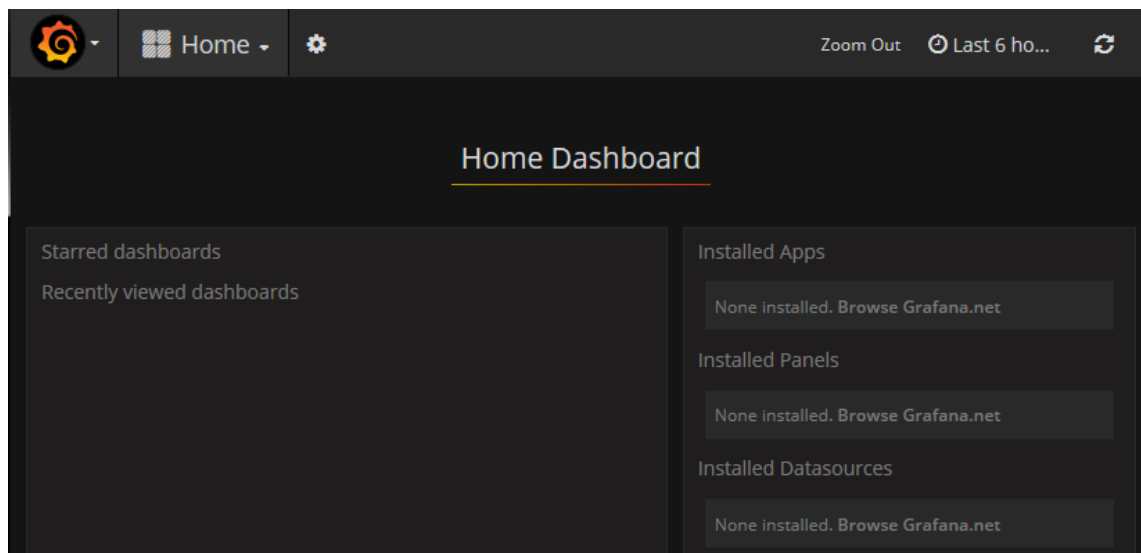
Node-REDin ohjelmointipalikkana toimii solmu, tai *node*. Data ja signaalit kulkevat virtauksenomaisesti nodesta toiseen, ja jokainen node käsittelee sen vastaanottamaa dataa, jonka jälkeen data välitetään eteenpäin seuraavalle nodelle. Flow-ohjelmointi on hyvin visuaalista, joten sitä on helppo ymmärtää. Sen sijaan, että koko ohjelma olisi yksi iso koodirivistö, ovat koodit sijoitettuna nodejen sisälle. [9.]

Node-RED on selainpohjainen, ja ohjelmaa pääsee muokkaamaan menemällä selaimella laitteen IP-osoitteeseen (Internet Protocol) porttiin 1880. Node-RED-ohjelmaa kehitetään yhteisön voimin vapaaehtoisesti, ja siihen on saatavilla eri lisäosia eri käyttötarkoituksia varten [9]. Tässä projektissa Node-REDiin asennettiin InfluxDB-lisäosa, jonka avulla data saatiin välitettyä serverikoneelle siinä olevaan InfluxDB-tietokantaan. Kuvassa 10 on Node-RED Internet-selaimessa sekä sen esimerkkiohjelma LEDin syyttämiseen.



Kuva 10. Node-REDin ohjelmointi-ikkuna

Serverikoneen käyttöjärjestelmänä toimi Ubuntu, ja siihen asennettiin InfluxDB sekä Grafana. InfluxDB on tietokantaohjelmisto, joka kerää ja tallentaa laitteiden lähettämää dataa. Grafana on alusta, jolla pystytään esittämään graafisesti tietokannan, eli tässä tapauksessa InfluxDB:n, keräämää dataa. Kuvassa 11 on Grafanassa oletuksena näkyvä pääikkuna. Grafana toimii serveritietokoneella, ja siihen pystyy kirjautumaan sisään menemällä Internet-selaimella serverin IP-osoitteeseen porttiin 3000 ja syöttämällä käyttäjätunnuksen ja salasanan.



Kuva 11. Grafanan koti-ikkuna

3 Alustavat testaukset

3.1 Menetelmät

Ennen varsinaisia kenttätestejä laitteita sekä antureita testattiin toimistossa magneetin avulla. Magneetilla simuloitiin maakaapelissa kulkevan virran aiheuttamaa magneettikenttää. Kun ohjelmisto oltiin saatu kehitettyä riittävän pitkälle testejä varten, laitteet vietiin konepajalle käytännön testaukseen. Testauksia suoritettiin Kurikassa sijaitsevassa Viafin Process Pipingin konepajassa, sekä myöhemmin Viafin Uusimaa Pipingin esivalmistuskonepajassa Vantaalla. Laitteita testattiin useampaan kertaan suunnittelun lomassa. Ensimmäisissä testeissä testattiin lähinnä anturin, ohjausyksikön sekä tiedonsiirron toimivuutta. Myöhemmissä testeissä mukaan tuli muita komponentteja, jotka päätyivät edelleen lopulliseen laitteeseen.

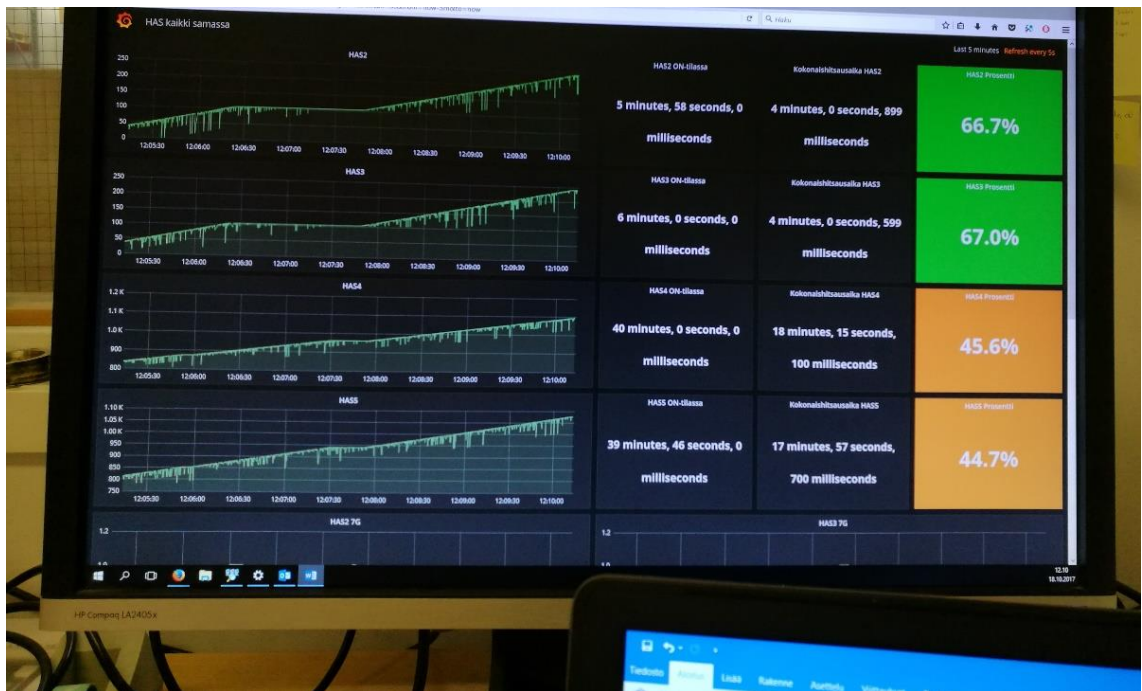
Ensimmäisissä testeissä laitteet täytyi ensin kytkeä lähiverkkoon, jotta ne pystyivät lähettämään dataa verkon yli serverikoneelle. Konepajoissa oli olemassa Wi-Fi-verkko, johon laitteet yhdistettiin yksitellen. Sen jälkeen laitteet siirrettiin hallin puolelle, ja anturit kiinnitettiin hitsauslaitteiden maakaapeleihin teipillä. Laitteet sijoitettiin maahan maakaapelin läheisyyteen, mikä on havainnollistettu kuvassa 12.



Kuva 12. Laitteiden testausta Kurikassa

Joissakin hitsauslaitteen maakaapelissa saattoi olla kiinni useampi anturi, jolloin laitteiden vastaanottamaa dataa pystyttiin vertailemaan keskenään. Myöhemmissä testeissä kokeiltiin erilaisia anturin kiinnitystapoja maakaapeliin, sekä Wi-Fi:n sijaan laite oli yhteydessä serveriin 3G-yhteydellä, joka tapahtui morkulan avulla.

Laitteiden toimintaa seurattiin paikan päällä kannettavalta tietokoneelta ja testien edistymisestä tehtiin muistiinpanoja. Testien päätyttyä testeistä luotiin raportti, joka lähetettiin projektissa mukana oleville henkilöille. Testit olivat suurilta osin onnistuneita, mutta testeissä havaittiin myös vähäisiä ongelmia. Laitteen eri komponentit olivat toimintavarmoja ja ohjausyksikkö kykeni suoriutumaan testausympäristössä, mutta ohjelmisto toimi vaihtelevasti.

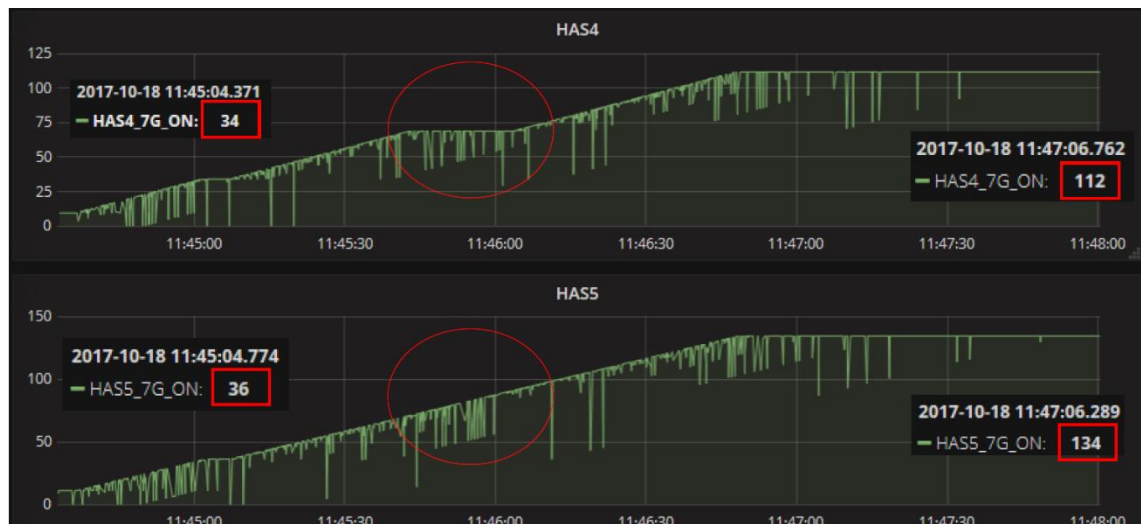


Kuva 13. Seurantalaitteiden monitorointia Kurikassa

3.2 Havainnot, ongelmat, sekä niiden ratkaisut

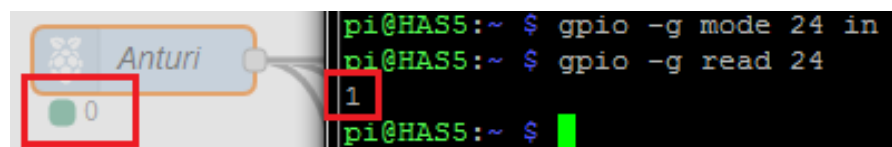
Ensimmäisissä testeissä tavoitteena oli selvittää, miten erilaiset anturit suoriutuivat ja kuinka luotettavaa tietoa ne antoivat. Käytössä oli kolmea eri magneetoresistiivistä anturia, josta jokaisella oli eri herkkyys. Testeissä ilmeni, että kaikista herkin anturi suoriutui testeistä parhaiten, eikä sillä ollut mitään heikkouksia. Tämä anturi valikoitui käytettäväksi tässä projektissa.

Laitteet nauhoittivat hitsausta niin kuin niiden oli tarkoituskin, mutta toisissa ja kolmansissa testeissä nauhoituksessa ilmeni ongelmia, joita simulaatiossa ei ilmennyt. Välillä anturi ei näyttänyt tunnistavan ollenkaan, mutta välillä se jäi tunnistustilaan, vaikka hitsaustyötä ei tuolloin tehty. Ongelmien paikantaminen ja selvittäminen vei aikansa, sillä oli vaikea suoraan sanoa, mikä tarkalleen oli ongelman aiheuttaja. Ongelman saattoi aiheuttaa anturi, ohjauksikkö tai siihen vaikuttavat ulkoiset häiriöt tai ohjauksikköissä oleva ohjelmisto. Paikantamista hankaloitti myös se seikka, että vikatilien aiheuttaneiden olosuhteiden toisintaminen toimisto-olosuhteissa oli lähes mahdotonta. Ongelma vaikutti ilmaantuvan sattumalta.



Kuva 14. Häiriö mittauksessa. Ylemmässä kuvaajassa laite lakkaa tunnistamasta hitsauksen päätyttyä, mutta alemmassa kuvaajassa laite jatkaa edelleen tunnistamista.

Lopulta selvisi, että ongelmat aiheutuivat Node-REDistä, jonka oma node digitaalitulon lukua varten ei antanut luotettavaa informaatiota, joka näkyy kuvassa 15. Syy saatiin selville siten, että avattiin Raspberry Pin Terminal-ohjelma, jossa voidaan suorittaa komentoja. Terminalissa kirjoitettiin komento, joka lukee GPIO:n tilan, ja tätä tietoa verrattiin Node-REDin noden antamaan tietoon. Tässä huomattiin ristiriita, sillä Node-REDin digitaalitulo antoi virheellistä tietoa verrattuna Terminalissa suoritettuun komentoon.



Kuva 15. Ristiriita signaaleissa. Vasemmalla näkyvä Node-REDin node antaa virheellisesti tietoa, että anturi tunnistaisi. Oikealla näkyvässä Terminalissa näkyy oikea tieto, eli anturi ei tunnista.

Ongelma saatiin korjattua siten, että ohjelmaan lisättiin node, joka lukee tiedon suoraan GPIO:sta samalla tavalla kuin Terminalissa suoritettu komento. Tämä node luetaan joka sekunnin välein, ja tieto välitetään eteenpäin muuhun ohjelmaan.

Datan välittämisessä laitteelta serverille oli myös pieniä ongelmia. Nämä näkyivät Grafanassa, jonka piirtämissä kuvaajissa esiintyy arvojen heittoja sekä sahalaitoja, jotka näkyvät kuvissa 13 ja 14. Nämä ongelmat johtuivat yhteysongelmasta laitteen ja serverin välillä, jolloin tämä näkyi myös virheviestinä Node-RED-ohjelman Debug-välilehdellä, joka on nähtävissä kuvassa 16.

```
4.9.2017 klo 14.35.10 node: 3bc2f71a.01cod8
msg : error
"Error: Request timed out"

4.9.2017 klo 14.35.10 node: 3bc2f71a.01cod8
msg : error
"Error: No host available"

4.9.2017 klo 14.35.11 node: 3bc2f71a.01cod8
msg : error
"Error: Request timed out"

4.9.2017 klo 14.35.11 node: 3bc2f71a.01cod8
msg : error
"Error: No host available"
```

Kuva 16. Debug-välilehdellä näkyvät virheviestit, kun laitteen yhteys on epävaka.

Tämä ongelma oli kuitenkin lähinnä kosmeettinen. Vaikka kuvaajat heittelehtivät sekä näyttivät antavan epätarkkaa dataa, niin ennen pitkää tiedot päivittyivät Grafanassa, jonka jälkeen kuvaaja näytti viimeisintä dataa, jonka se oli vastaanottanut. Kuvaaja voi vääristyä katkoksen ollessa pidempiaikainen, mutta pahimmassakin tapauksessa katkos kesti korkeintaan minuutin.

Ensimmäisissä testeissä anturien toimivuutta testattiin teippaamalla ne kiinni maakaapeliin. Myöhemmissä testeissä käytettiin kuvassa 17 näkyviä metallisia hauenleukoja, jonka sisällä anturit olivat. Tämä toimi hyvin kiinnityksenä maakaapeliin, mutta testeissä kuitenkin ilmeni, että anturit eivät tunnistanee hitsausvirtaa ollenkaan. Magneetilla testattuna anturit tunnistivat, mutta hitsauksessa maakaapelin ympärilleen luoma magneetikenttä oli paljon heikompi kuin magneetin oma, ja se ei ollut riittävä antureille.



Kuva 17. Metalliset hauenleuat

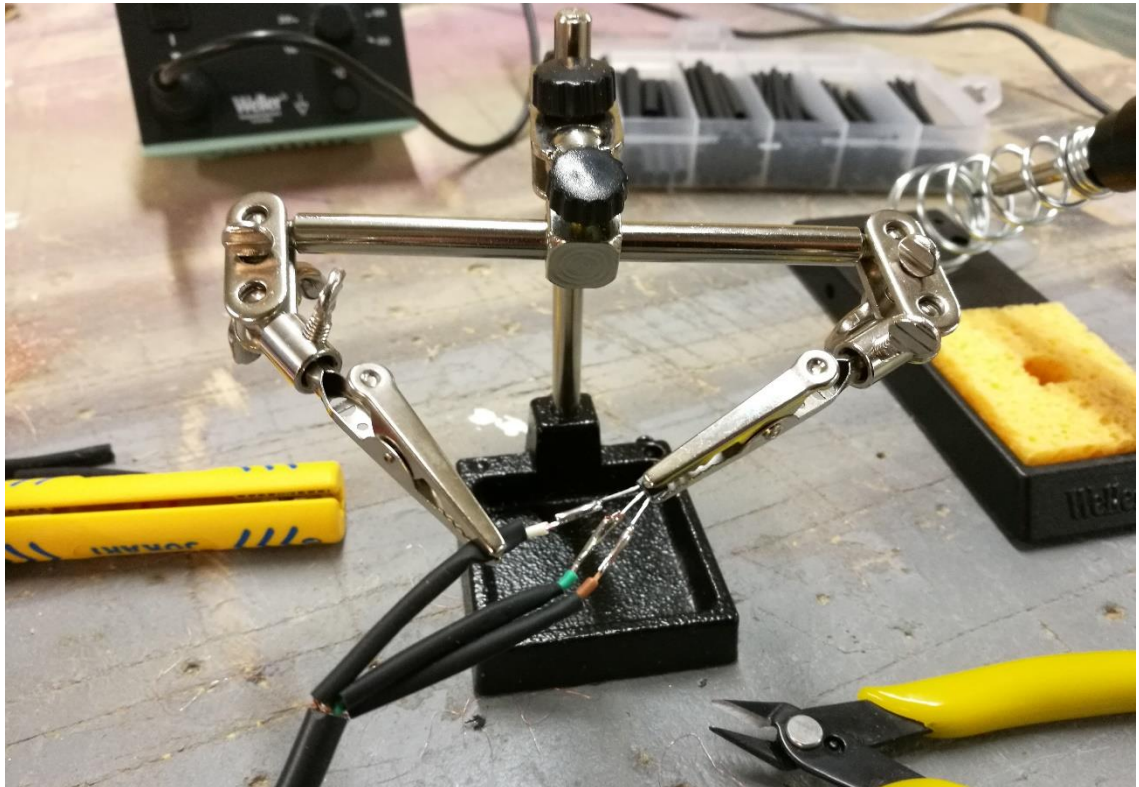
Ongelmaa lähdettiin ratkomaan vertailemalla antureita eri tilanteissa. Pian selvisi, että ilman hauenleukoja anturit tunnistivat, mutta hauenleukojen sisällä olevat anturit eivät tunnistaneet. Tämä johtui luultavasti siitä, että hauenleuat olivat tehty teräksestä, joka on magneettinen metalli. Teräksestä tehdyt hauenleuat näyttivät vaikuttavan magneettikenttään sen verran, että anturit eivät enää havainneet kaapelin ympärilleen luomaa magneettikenttää. Kun anturit otettiin pois hauenleuoista ja se kiinnitettiin uudelleen maakaapeliin teipillä, niin anturit tunnistivat jälleen. Tämän takia päädyttiin ratkaisuun, että anturi oli muovisten puristimien sisällä, jolloin anturin ympärillä ei ollut magneettista materiaalia.

4 Toteutus

Taustatyön ja testauksien jälkeen saatiin lopullinen kuva siitä, minkälaisista komponenteista laite tulee koostumaan. Ensimmäisissä testeissä laitteita oli käytössä neljä kappaletta, ja ne olivat aluksi ilman koteloa. Myöhemmissä testeissä laitteet olivat väliaikaisessa kotelossa ja lopullisissa testeissä käytettiin tilaustyönä tehtyjä koteluita.

Uusia koteluita ja varavirtalähteitä tilattiin 20 kappaletta. Kaikkea muuta tilattiin 16 kappaletta lisää siten, että laitteita tulisi olemaan testikäytössä yhteensä 20 kappaletta. Ensimmäiset 20 laitetta rakennettiin komponenteista, joiden toiminnollisuudet sekä toimintavarmuus oli testattu ja todistettu riittäviksi.

Lähes kaikki komponentit saapuivat viikon sisällä tilaamisen jälkeen, mutta koteloiden, mokkuloiden sekä varavirtalähteiden toimitus kesti noin kaksi viikkoa. Näiden laitteiden toimitusta odotellessa voitiin rakentaa valmiiksi anturikaapelit sekä testata näiden toimivuus. Kaapeli sekä johtimet kuorittiin, minkä jälkeen anturi juotettiin kiinni johtimiin, mikä näkyy kuvassa 18. Kontaktipinnat suojattiin kutistesukalla, sekä lopuksi anturin päälle tuli vielä toinen kerros kutistesukkaa. Kaapelin toiseen päähän tulivat naarasliittimet, jotka liitettiin Raspberry Pin GPIO-nastoihin. Lopuksi anturi sijoitettiin muovipuristimien sisään.



Kuva 18. Anturin juottamista kaapeliin

Muovipuristimet valikoitiin sen kokoisiksi, että sinne pystyisi sijoittamaan ja kiinnittämään anturin, sekä että se olisi riittävän napakasti kiinni kaapelissa. Anturi sijoitettiin muovipuristimien sisäpintaan, ja se pystyi kiinni nippusiteillä. Anturin ja puristimien leukojen ympärille laitettiin vielä kerros kutistesukkaa, jotta anturi pysyisi kiinni leukojen sisäpinnassa. Muovipuristimet on kuvattu kuvassa 19.



Kuva 19. Muovipuristin

Kuvassa 20 näkyviä uusia varavirtalähteitä testattiin saman tien, kun ne olivat saapuneet. Ne laitettiin latautumaan, ja samalla otettiin aikaa, kuinka kauan niillä kestää latautua tyhjästä täyteen. Tämän jälkeen niitä testattiin jättämällä ne syöttämään virtaa Raspberry Pille niin kauan, kunnes akku loppui. Uudet varavirtalähteet latautuivat täyteen noin yhdeksässä tunnissa ja pystyivät syöttämään virtaa yhtäjaksoisesti noin 20 tunnin ajan.



Kuva 20. Uusi 20 000 mAh:n varavirtalähde

Lopuksi saapuivat uudet kotelot, joka näkyy kuvassa 21. Kun nämä saapuivat, voitiin aloittaa laitteiden sarjatuotanto. Komponentit pyrittiin valikoimaan sen mukaan, että laitteen kokoonpanoon kuluisi mahdollisimman vähän aikaa ja vaivaa. Eniten aikaa vievä toimenpide laitteen kokoonpanossa oli anturikaapelin rakentaminen. Kaikki muu asentaminen hoitui melko nopeasti.



Kuva 21. Uusi kotelo

Painonapeissa liittimet olivat valmiina, mikä näkyy kuvassa 22. Myös nappien mukana tulleissa johtimissa toisessa päässä liittimet olivat valmiina, joten ainoastaan toiseen päähän piti kiinnittää omat liittimet. Painonapin mallia vaihdettiin vielä viime hetkellä, joten kotelon kannessa olevan napin reikää täytyi suurentaa sitä varten.



Kuva 22. Lukkiutuva painonappi

Mokkulat kiinnitettiin suoraan Raspberry Pin USB-porttiin, ja se käynnistyi ja otti yhteyden automaattisesti laitteen käynnistyttyä. Jokaista mokkulaa kohden oli oma 3G-liittymänsä. Liittymät tilattiin Elisalta, ja ne avattiin kaikki kerralla. Mokkula on kuvattua kuvassa 23.



Kuva 23. 3G-mokkula

Kaiken kaikkiaan 20 laitteen rakentamisessa kului aikaa reilut kaksi viikkoa. Kun kaikki laitteet olivat valmiit, ne vietiin vielä viimeistä kertaa kenttätesteihin. Nämä testit toimivat kenraaliharjoituksena projektille, ja siinä haluttiin selvittää, miten serverikone kestää 20:tä yhtäaikaista yhteyttä. Samalla haluttiin selvittää, miten lopulliset laitteet toimivat käytännössä, sillä nyt osa niistä jätettiin konepajoille testattavaksi ilman aktiivista valvontaa. Laitteet vietiin työpäivän päätteeksi lataukseen ja otettiin uudelleen käyttöön seuraavana päivänä.

4.1 Laitteisto

Laitteeseen kuuluu ohjausyksikkö, jona toimii Raspberry Pi 3 Model B. Sen virtalähteenä toimii 20 000 mAh:n varavirtalähde, joka syöttää jatkuvasti virtaa Raspberry Pille. Varavirtalähdettä ladataan kotelon kyljessä olevasta latausportista. Raspberry Pin USB-portissa on kiinni 3G-mokkula, ja GPIO-nastoissa ovat kiinni lukkiutuva painonappi sekä kaapeli, jonka päässä anturi on.

Anturi kulkee läpiviennin kautta ulos laitteesta, ja se on kiinnitetty puristimien sisäpintaan. Varavirtalähde on kiinnitetty kotelon pohjalle, ja sen päälle on kiinnitetty Raspberry Pi. Laatikossa on metallinarun päässä haka, jolla se kiinnitetään johonkin hitsauskoneen läheisyyteen. Kuvassa 24 näkyy laitteen sisällä olevat komponentit.

Kannessa oleva lukkiutuva painonappi toimii yhtä aikaa sekä virtanappina että nauhoitusnappina. Kun nappia painetaan, laite käynnistyy ja aloittaa nauhoituksen heti, kun Node-RED käynnistyy. Samalla laitteen mokkula ottaa automaattisesti yhteyden 3G-verkkoon ja sitä kautta serveriin. Kun nauhoitus on päällä, kannessa olevan napin sininen LED-valo syttyy. Kun nappi kytketään pois päältä, LED sammuu, nauhoitus loppuu ja laite sammuu.



Kuva 24. Laite kansi avattuna

Jos laite menettää yhteyden serveriin, virtanapissa oleva LED alkaa vilkkumaan, ja samalla laite yrittää muodostaa yhteyden uudelleen. Kun yhteyden muodostus jälleen onnistuu, vilkkuminen loppuu ja valo alkaa palaa jälleen jatkuvana. Vaikka laite menettäisi yhteyden serveriin, niin nauhoitus jatkuu silti. Kun yhteys on jälleen muodostettu, se lähettää viimeisimmän tiedon serverille, joten yhteyskatkon aikana nauhoitettu data ei häviä minnekään. Data häviää vain, jos laite sammutetaan katkoksen aikana.

Kaikki komponentit pois lukien anturi sekä muovipuristin ovat koteloituna muoviseen koteloon. Kotelon pinnalla on ainoastaan painonappi, latausportti varavirtalähteelle sekä läpivienti anturin kaapelille. Kotelon kansi on läpinäkyvä, jotta sen kautta voidaan nähdä komponenttien merkkivalot sekä varavirtalähteen varaustaso sitä ladattaessa. Mokkula näyttää eri merkkivaloa riippuen siitä, onko se toiminnassa tai onko se yhdistänyt verkkoon. Mokkulan merkkivalo näyttää joko sinistä tai vaaleansinistä väriä, kun se on muodostanut yhteyden. Jos se ei saa yhteyttä, merkkivalo vilkkuu sinisenä.

4.2 Ohjelmisto

4.2.1 Raspberry Pi

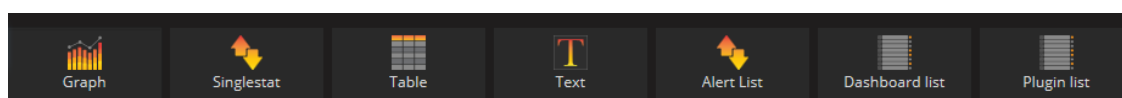
Raspberry Pi:ssä olevaan Node-RED-ohjelmaan ohjelmoitiin kaikki hitsaustyön seurantaan tarvittava ohjelmakoodi. Ohjelma nauhoittaa aikaa, kuinka kauan laite on ollut päällä, sekä kuinka pitkään anturi on tunnistanut maakaapelissa kulkevaa virtaa. Ohjelma laskee myös näiden kahden tiedon osamäärän, eli kuinka monta prosenttia koko ajasta on käytetty aktiiviseen hitsaamiseen. Ohjelma laskee myös sen, kuinka monta kertaa hitsauspilli on sytytty. Kaikki tämä data välittyy InfluxDB-nodejen kautta serverikoneen tietokantaan.

Raspberry Pi:hin piti myös luoda ohjelmakoodit mokkulaa sekä painonappia varten. Mokkula ei yhdistänyt itsestään verkkoon, vaan sitä varten piti asentaa eri ohjelmia, jotta se saatiin toimimaan. Nappia varten luotiin myös oma ohjelmakoodi, jonka avulla laite käynnistyi ja sammui. Sama nappi on myös tulotietona Node-RED:ssä, jossa se käynnistää sekä lopettaa nauhoituksen.

4.2.2 InfluxDB ja Grafana

Laitteet lähettivät dataa serverikoneelle, jossa data tallentui InfluxDB-tietokantaan. Uuden laitteen tai näiden mittauksien lisääminen ei vaatinut toimenpiteitä serveritietokoneella. Se tapahtui käyttäen Node-REDin InfluxDB-nodea, johon syötettiin serverikoneen IP-osoite sekä haluttu nimi mittaukselle. Kun tähän nodeen syötettiin dataa, serverikoneella oleva InfluxDB loi tästä automaattisesti uuden mittauksen tietokantaan.

Grafana luki dataa serverikoneen InfluxDB-tietokannasta, ja Grafanaan voitiin luoda erilaisia paneeleja, jotka näyttivät InfluxDB:hen tallennettua dataa halutulla tavalla. Eri paneelivaihtoehdot ovat kuvattuna kuvassa 25. Niitä oli mm. kuvaaja, taulukko, hälytyslista tai tietokantaan tallennetun tiedon näyttäminen lukuarvona. Paneeleja on mahdollista ladata lisää Grafanan verkkosivuilta.



Kuva 25. Grafanan paneelin eri vaihtoehtoja

Grafanaan luotiin kojelautoja, *dashboardeja*, joihin näitä paneeleja voitiin lisätä. Käyttöliittymän dashboardit suunniteltiin siten, että niistä oli mahdollista nähdä yhden laitteen nauhoittama data yhdeltä riviltä. Yhteen dashboardiin oli mahdollista sijoittaa monen eri laitteen nauhoittama data, ja tässä dashboardissa nämä tiedot olivat allekkain. Hyvän suorituskyvyn ja helppolukuisuuden takaamiseksi kaikkia laitteita ei sijoitettu yhteen dashboardiin, vaan ne jaettiin useammalle dashboardille.

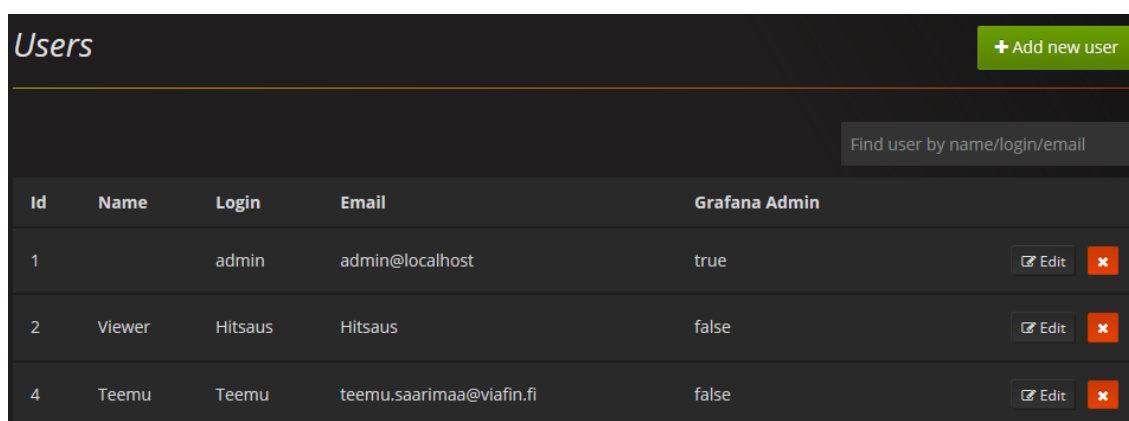
Yhden laitteen nauhoittama data näkyy siten, että vasemmalla ensimmäisenä näkyy kuvaaja. Kuvaajan pystyakselilla näkyy aika, kuinka pitkään anturi on tunnistanut. Kun maakaapelissa kulkee virtaa, kuvaaja lähtee nousuun. Kuvaajan vaaka-akselilla näkyy kellonaika. Kuvaajan vieressä näkyy numeroina se aika, kuinka kauan laite on ollut päällä ja nauhoittamassa. Tämän oikealla puolella näkyy sama data mikä ensiksi mainitussa kuvaajassakin, eli anturin tunnistusaika, mutta tässä se näkyy numeroina. Seuraavaksi näkyy kahden viimeksi edellä mainitun luvun osamäärä, eli toisin sanoen se, kuinka monta prosenttia nauhoitetusta ajasta on tehty aktiivista hitsaustyötä. Viimeisenä näkyy tieto, kuinka monta kertaa hitsauspili on sytytetty nauhoituksen aikana. Kuvassa 26 näkyy viiden eri laitteen nauhoittama data noin tunnin ajalta.



Kuva 26. Grafanan dashboard. Prosenttilukujen taustavärit ovat vain suuntaa antavia, tarkempi taustavärien käyttö määriteltiin myöhemmin.

Grafanaan oli mahdollista luoda eri käyttäjätilejä, ja käyttäjätilejä oli mahdollista hallinnoida siten, että eri käyttäjätilit näkivät eri organisaatioita. Organisaatiot sisälsivät näitä dashboardeja, ja järjestelmänvalvoja pystyi määrittämään, mitkä käyttäjät näkevät mitkäkin organisaatiot. Käyttäjälista on havainnollistettu kuvassa 27.

Käyttäjätilejä voidaan luoda eri henkilöille tai vaikkapa eri yrityksille, joten samassa tietokannassa voi olla eri monen yrityksen omat laitteet ja ne kaikki löytyvät samasta osoitteesta. Järjestelmänvalvojan suomat käyttöoikeudet varmistavat sen, että käyttäjät voivat lukea vain niiden organisaatioiden dashboardeja, mihin niille on annettu lupa.



Id	Name	Login	Email	Grafana Admin	
1		admin	admin@localhost	true	Edit X
2	Viewer	Hitsaus	Hitsaus	false	Edit X
4	Teemu	Teemu	teemu.saarimaa@viafin.fi	false	Edit X

Kuva 27. Grafanan käyttäjälista

Tiedot säilyvät serverikoneen tietokannassa niin kauan kuin siinä kapasiteettia riittää. Dataa on mahdollista myös selata taaksepäin, ja data ulottuu niin kauaksi kuin dataa on tietokantaan tallennettu. Näitä historiatietoja voidaan analysoida ja hyödyntää tarpeen mukaan.

4.2.3 Etäyhteys

Laitteeseen voidaan ottaa myös etäyhteys, jolloin ohjelmistoa voidaan päivittää tai muokata etänä. Raspberry Pihin voidaan normaalitilanteessa ottaa etäyhteys ainoastaan lähiverkon kautta, mutta morkulan ja 3G-yhteyden takia se vaatii erillisen ohjelmiston ja palvelun toimiakseen. Tämä onnistui käyttämällä remot3.it-nimistä palvelua, jonka käyttöliittymä näkyy kuvassa 28. Palvelu on saatavilla ilmaiseksi henkilökohtaiseen käyttöön, mutta liiketoimintaa tehtäessä siitä on maksettava. Laitteita voidaan liittää palveluun niin paljon kuin tarve vaatii, rajoituksia ei ole.

Kun käyttäjätili oli remot3.it-palveluun tehty, Raspberry Pihin asennettiin tämän ohjelmisto ja se liitettiin tähän käyttäjätiliin. Ensiksi laitteelle annettiin nimi, jona se näkyi palvelussa. Sen jälkeen lisättiin ne palvelut, mitä laitteeseen haluttiin. Tässä tapauksessa tarvittiin SSH ja Node-RED. SSH eli Secure Shell on protokolla, jolla voidaan ottaa etäyhteys laitteeseen, ja sillä pystyttiin suorittamaan komentoja Raspberry Pissä. Node-RED-palvelulla pystyttiin muokkaamaan nauhoitusohjelmaa Internet-selaimessa.

The screenshot shows the remot3.it web interface. On the left is a dark sidebar with a user profile for 'teemu.saarimaa@viafin.fi' and navigation options: Manage Devices, Storage, Job Status, Register Devices, My Account, and Support. The main content area has a header with the remot3.it logo and a 'Devices' section. Below this is a table with columns for Status, Device Name, Share, HWID, and Internal IP. The table lists seven devices, with the first four having a green checkmark status and the last three having a red minus sign. The last three devices also show their last online time.

	Status	Device Name	Share	HWID	Internal IP
<input type="checkbox"/>	✓	HAS_1		b8:27:eb:32:0a:0c-xa	100.85.19.89
<input type="checkbox"/>	✓	HAS_2		-y8IFKkyjTueetrfD9P	100.84.141.29
<input type="checkbox"/>	✓	HAS_3		b8:27:eb:3a:84:0d-5c	100.85.31.197
<input type="checkbox"/>	✓	HAS_4		b8:27:eb:27:c5:77-la	100.85.44.32
<input type="checkbox"/>	✗	HAS_5 Last Online 2018-01-11T09:30:06.167Z		b8:27:eb:ee:ab:64-oc	100.88.123.62
<input type="checkbox"/>	✗	HAS_6 Last Online 2018-01-14T13:38:32.057Z		b8:27:eb:ba:4a:22-Bs	100.85.165.142
<input type="checkbox"/>	✗	HAS_7 Last Online 2018-01-11T09:32:38.673Z		b8:27:eb:a9:7c:43-8x	100.88.128.186

Kuva 28. remot3.it-käyttöliittymä

4.3 Kokonaisuus

Laite toimii täysin itsenäisesti ja ihmiskäyttäjän ainoa tehtävä on käynnistää laite nappia painamalla sekä kiinnittää anturi kiinni hitsauskaapeliin. Kaikki muu hoituu ohjelmien kautta automaattisesti. Laite nauhoittaa hitsaustyötä ja lähettää tämän datan langattomasti serverikoneelle. Tiedon lähettämiseen vaaditaan, että 3G-kattavuus on riittävä ja että verkkokäytön liittymä pystyy toimimaan siinä verkossa. Nauhoitus alkaa välittömästi, kun Node-RED käynnistyy ja laite saa muodostettua yhteyden serveriin.

Laite sijoitetaan hitsauskoneen läheisyyteen kiinnittämällä se haan avulla johonkin lähellä olevaan poikkitankoon. Yleensä hitsauskoneet ovat sijoitettuna kaasupulloineen pieneen kääryyn, joten seurantalaitte voidaan kiinnittää johonkin kääryssä oleviin poikkitankoihin. Lian, kolhujen ja vahinkojen välttämiseksi on suositeltavaa laittaa laite roikkumaan tangosta kuin jättää se lattialle.

Laite voi olla päällä koko työpäivän ajan, ja laite välittää dataa päivän aikana suoritetusta hitsaustyöstä. Kun laitetta ei enää käytetä, se sammutetaan nappia painamalla, jonka jälkeen laite vietään lataukseen. Latausporttina toimii useimmista matkapuhelinmaileista löytyä microUSB.

Konepajaympäristössä on paljon likaa ja pölyä, joten latausportin suojaaminen oli tärkeää. Latausportin suojana toimii kuvassa 29 näkyvä pieni metallilevy, joka on kiinnitetty ruuvilla kotelon kylkeen. Suoja käännetään sivuun latauksen ajaksi ja latauksen päätyttyä käännetään takaisin latausportin eteen.



Kuva 29. Latausportti ja sen suoja Levy

Vaikka laitteet on levitetty hajautetusti, toimivat ne silti hyvin keskitetysti muutaman palvelun kautta. Tämä takaa sen, että laitteiden monitorointi, ylläpito, päivitykset sekä mahdolliset vikojen ja bugien korjaaminen ohjelmassa pystytään suorittamaan etätyönä. Aina vaatuksena on, että laite saa onnistuneesti muodostettua 3G-yhteyden, muussa tapauksessa laite pitää ottaa fyysiseen huoltoon. Kuvassa 30 on valmis laite, joka vietiin seuraavaksi käytännön testaukseen.



Kuva 30. Valmis laite

5 Kenttätetit

Projektin saattaminen loppuun vaati vielä sen, että kaikkia uusia 20:tä laitetta testattiin samanaikaisesti. Testausajankohta oli kuitenkin sellainen, että kaikkia 20:tä laitetta ei pystytty testaamaan kerralla yhtä aikaa. Tuohon aikaan konepajoilla ei ollut paljon projekteja käynnissä, joten hitsaustyötä ei suoritettu suurissa määrin. Hitsaustyötä tekeviä työntekijöitä ei myöskään ollut niin montaa sillä hetkellä työssä, että jokaiselle olisi voitu osoittaa yksi laite testattavaksi.

Laitteita testattiin aluksi Vantaalla viiden laitteen erissä, ja ne olivat kaikki kiinni yhdessä hitsauskoneessa. Näin haluttiin varmistaa, että kaikki laitteet toimivat oikein kenttäolosuhteissa. Tämän jälkeen kaksi työntekijää perehdytettiin laitteiden käyttöön, ja heille osoitettiin laitteet itsenäiseen käyttöön. Kuvassa 31 laite on hitsauskoneessa kiinni haalla, ja anturi on kiinnitettyä puristimien avulla maakaapeliin. Laitteille oli pajoilla oma latauspisteensä, jonne ne jätettiin yön ajaksi latautumaan. Kaksi laitetta jätettiin Vantaalle, ja kymmenen laitetta vietiin testattavaksi Kurikkaan ja Teuvalle.



Kuva 31. Laite testissä Vantaalla

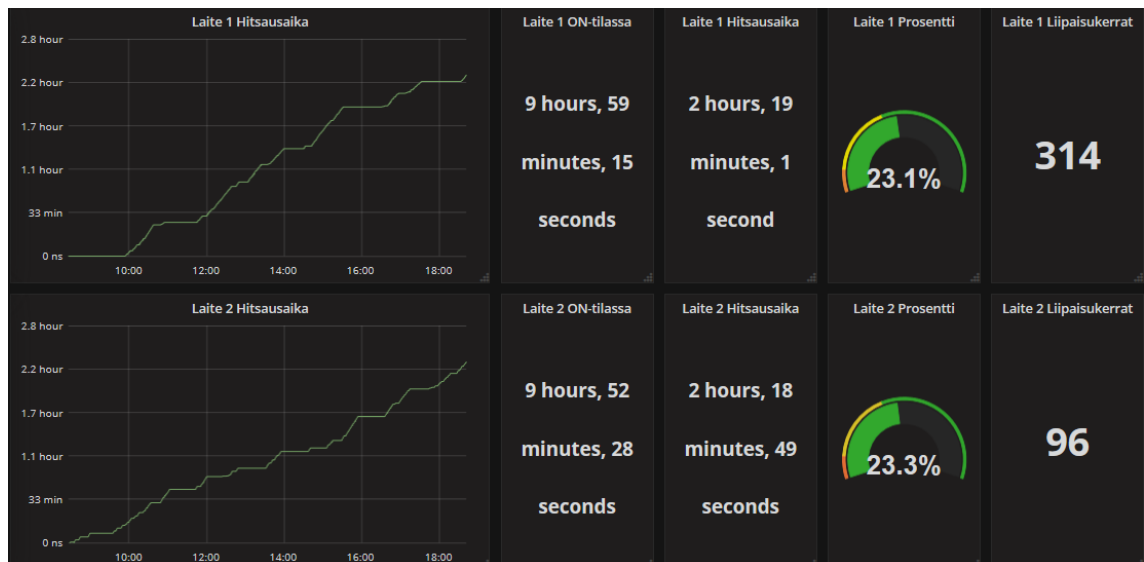
5.1 Testien havainnot

Konepajoilla oli käytössä erityyppisiä hitsauskoneita. Niitä oli mm. TIG, TIP-TIG, pulssi-MAG ja jauhekaaritorni. Laitteita testattiin näissä kaikissa ja pieniä vaihtelevuuksia lukuun ottamatta laitteet nauhoittivat ja lähettivät dataa onnistuneesti. Tietyillä TIG-hitsauslaitteilla sekä jauhekaaritornilla nauhoituksessa oli eroja.

TIG-hitsauslaitteessa on Minilog-toiminto, jolla pystyy vaihtamaan hitsausvirtaa kahden virtatason välillä. Jos toinen virtataso oli liian pieni, niin silloin anturi lakkasi tunnistamasta. Usein tämä pienemmän virtatason käyttö kesti vain sekunteja, jonka jälkeen virtataso vaihdettiin jälleen korkeampaan. Silloin anturi tunnisti jälleen. Minilog-toimintoa käytettäessä hitsauspilli ei sammunut virtatasoa vaihdettaessa, mutta anturin tunnistuskatkoksesta johtuen se havaitsi sytytyskertoja enemmän kuin mitä todellisuudessa oli.

Jauhekaaritornilla nauhoittaessa oli havaittavissa samaa. Testeissä se teki yhtä pitkää saumaa, mutta nauhoitusohjelmassa näkyi lyhyitä parin sekunnin katkoksia, jotka lisäsivät ylimääräisiä sytytyskertoja. Tarkkaa syytä tälle ei selvinnyt, mutta se saattoi johtua jauhekaaritornin tavasta tehdä hitsaussaumaa.

Laitteita oli nyt myös useampi päällä yhtä aikaa, jolloin serverin vastaanottama data oli suurempaa verrattuna aiempiin testeihin. Näissä testeissä oli havaittavissa enemmän yhteysongelmia, mikä näkyi kuvaajissa sahalaitoina sekä viiveinä datan päivityksissä. Tästä huolimatta laitteet lähettivät onnistuneesti dataa serverin tietokantaan, eikä pitkiä katkoksia ilmennyt.



Kuva 32. Päivän aikana nauhoitettu data

Laitteen varavirtalähteet olivat ladattu täyteen ennen työpäivän alkua, ja laitteet olivat testeissä päällä koko työpäivän ajan. Kuvassa 32 näkyy yhden työpäivän aikana nauhoitettu data. Yhden työpäivän aikana akkujen varaustaso oli laskenut 40 %:iin. Päivän jälkeen laitteet laitettiin lataukseen yön ajaksi, ja ne olivat latautuneet täyteen seuraavaan aamuun mennessä. Varavirtalähteet suoriutuivat siis kenttätesteissä onnistuneesti.

Metallisten hauenleukojen korvaaminen muovisilla ratkaisi ongelman anturin havaitsemisherkkyudessa. Metallisilla hauenleuoilla anturi tunnisti hyvin vaihtelevasti, ja silloinkin ainoastaan suurilla virroilla hitsattaessa. Muovipuristimia käytettäessä ongelmia ei ilmennyt, ja ne toimivat kiitettävästi kiinnityksenä kaapeliin.

Mittauksien tarkkuudessa ilmeni pieniä eroja, mikä johtui Node-REDin epätarkkuudessa laskea sekunteja. Esimerkiksi laitteen ollessa päällä viisi minuutta laite nauhoitti neljä minuuttia ja 57 sekuntia. Ero on vain prosenttien luokkaa, mutta laitteen ollessa päällä esimerkiksi 12 tuntia voi tämä ero olla jo merkittävä. Tätä eroa pystyttiin kaventamaan lisäämällä visualisointiohjelman mittaustiedolle korjauskerron, jolloin data saatiin vastaamaan enemmän todellisuutta. Erot vaihtelivat kolmesta prosentista viiteen prosenttiin, joten tässä jouduttiin tekemään kompromissi korjauskertoimelle. Nauhoitusdataa ei saada koskaan täydelliseksi, mutta se saadaan riittävän tarkaksi.

5.2 Muut havainnot

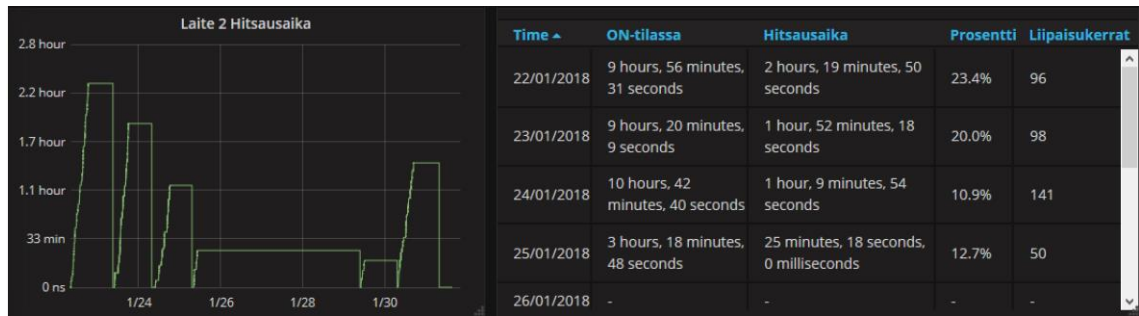
Kaikki laitteen komponentit olivat koteloituna pölytiiviseen muovikoteloon. Tässä ratkaisussa kääntöpuolena oli se, että siitä ei voi koskaan kokonaan katkaista virtoja, sillä varavirtalähde syöttää jatkuvasti virtaa Raspberry Pille. Raspberry Pistä ei voi katkaista virtoja ohjelmallisesti, vaan sen voi ainoastaan viedä valmiustilaan.

Ainoa tapa katkaista virrat laitteesta on joko ottaa virtajohto irti tai sammuttaa varavirtalähde sen omasta napista. Molemmat kuitenkin vaatisivat kotelon avaamisen tai että nämä komponentit olisivat jatkuvasti näkyvillä. Tässä käytössä ja näissä olosuhteissa tämä ei ollut tarkoituksenmukaista. Toinen vaihtoehto olisi ollut käyttää virrat katkaisevaa kytkintä kotelon kannessa, mutta tähän ei ollut saatavilla järkevän oloista kytkintä, joten se olisi pitänyt rakentaa itse, mikä olisi taas ollut liian vaivalloista ja kustannustehotonta.

Yksi ongelma oli myös laitteen tahaton käynnistyminen. Kun varavirtalähdettä alettiin ladata tai latausjohto otettiin pois, virransyöttö katkesi hetkellisesti, mutta jatkui jälleen hyvin lyhyen katkon jälkeen. Tämä aiheutti sen, että Raspberry Pi käynnistyi tahattomasti. Ongelma ratkaistiin siten, että Raspberry Pissä pyörii jatkuvasti koodi, joka tarkistaa, onko nappi ON- vai OFF-tilassa. Jos nappi on OFF-tilassa, laite sammuu automaattisesti.

Laite saattoi myös käynnistyä tahattomasti satunnaisina aikoina. Tämä johtui luultavasti siitä, että se GPIO-nasta jossa virtanappi oli kiinni, ei ollut 0-tilassa napin ollessa pois päältä, vaan se oli float-tilassa. Tämä tarkoitti sitä, että sen tila kellui ykkösen ja nollan välissä. Tällöin pienikin ulkoinen häiriö saattoi viedä sen 1-tilaan hetkellisesti, jolloin laite sai herätteen ja käynnistyi. Tämän olisi luultavasti voinut korjata lisäämällä alasvetovastuksen (pull-down resistor) GND-nastan (Ground) ja GPIO-nastan väliin. Tätä ei kuitenkaan nähty tarpeelliseksi, sillä tahaton käynnistyminen oli melko harvinaista ja aiemmin mainittu ohjelmakoodi piti huolen siitä, että laite sammui välittömästi käynnistyttyään, jos nappi ei ollut pohjassa.

Nauhoitettua dataa on tarkoitus käsitellä enemmän viikko- ja kuukausitasolla kuin päivätasolla, joten pidempiaikaisten testien seurauksena saatiin nyt ensimmäistä kertaa dataa pidemmältä aikaväliltä. Kuvassa 33 näkyy nauhoitettu data kahdelta viikolta. Visualisointiohjelmalla muokattiin siten, että sitä voitaisiin tarkastella muutenkin kuin pelkästään päivätasolla.



Kuva 33. Kahden työviikon aikana nauhoitettu data

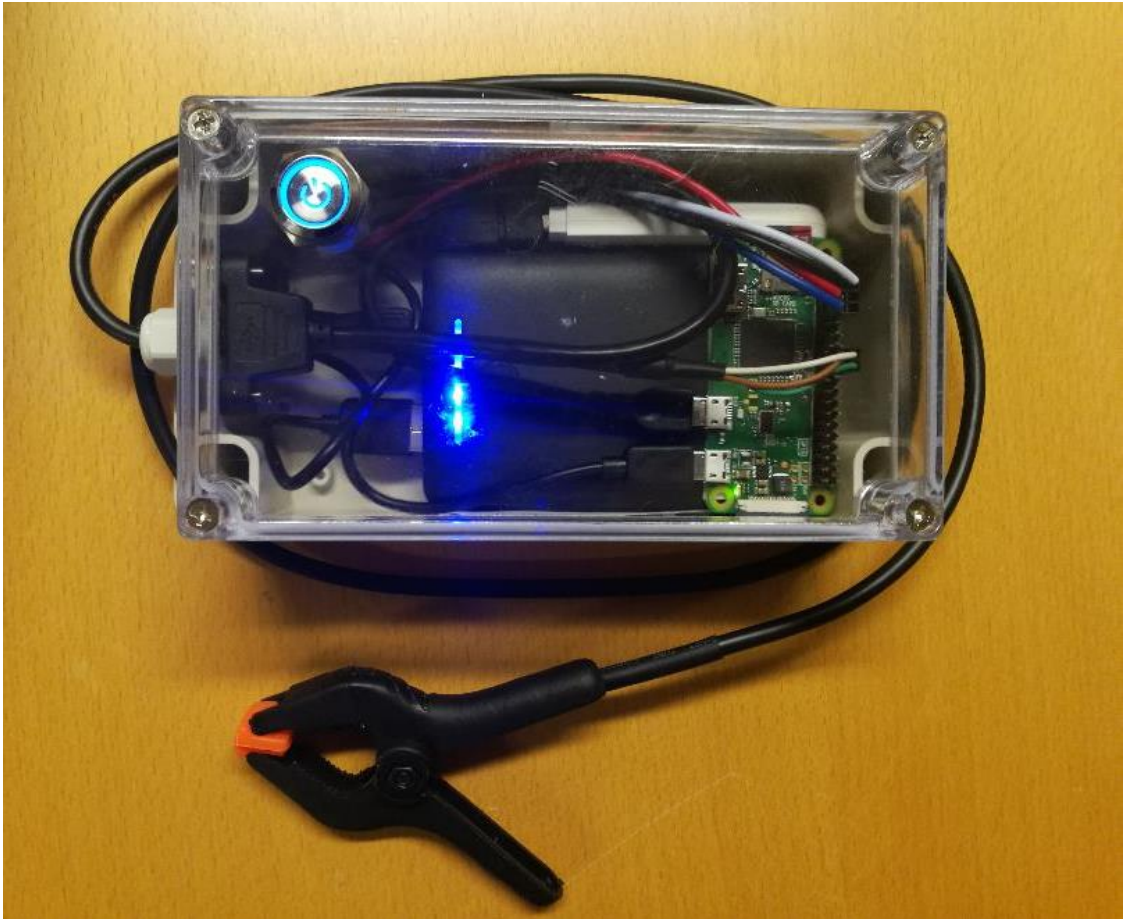
Nauhoitettu data vietiin taulukkoon, jossa jokaisella rivillä on siltä päivältä nauhoitettu data. Taulukko on mahdollista viedä .csv-tiedostomuotoon, jolloin se voidaan avata ja dataa voidaan tarkastella sekä muokata Excelissä. Visualisoinnissa aika-alueen voi rajata oman mielen mukaan, joten dataa voidaan tarkastella juuri sellaiselta ajanjaksolta kuin käyttäjä itse haluaa.

5.3 Tulokset ja jatkokehitys

Kenttätesteissä käytetyt laitteet olivat ensimmäisiä, jotka päätyivät varsinaisiin käytännön testeihin. Niitä testattiin niissä olosuhteissa sekä niillä menetelmillä, jossa niitä lopulta tullaan käyttämään. Nämä laitteet olivat melko suurikokoisia sekä kalliita, joten laitetta optimoitiin siten, että sen koko sekä kustannukset saatiin pienemmäksi. Ratkaisevin seikka optimoinnin suhteen oli sen ohjausyksikkö Raspberry Pi.

Ensimmäisissä laitteissa käytettiin Raspberry Pi 3 Model B:tä, joka on malleista uusin sekä samalla myös kallein. Tähän projektiin se oli kuitenkin liian kallis, eikä tässä läheskään hyödynnetty kaikkea sitä potentiaalia, mitä laitteella voitaisiin tehdä. Laite koitettiin saada toimimaan käyttämällä Raspberry Pi Zeroa, joka on Raspberry Pi -perheen halvin sekä kompaktein malli. Siinä on vain vähän tehoa verrattuna Pi 3 Model B:hen, mutta se oli samalla paljon pienempi sekä halvempi. Tähän projektiin sen suorituskyky oli riittävä, sekä sen alhaisempi koko ja virrankulutus antoivat mahdollisuuden miettiä uudelleen koitelointia sekä varavirtalähdettä.

Laitetta ei aluksi saatu toimimaan Raspberry Pi Zerolla, sillä tiettyjä oleellisia ohjelmia ei ollut tuolloin vielä saatavilla tälle laitteelle. Myöhemmin nämä ohjelmistot tulivat saataville ja seurantaohjelma saatiin toimimaan, joten seuraavat 20 laitetta tilattiin käyttäen Raspberry Pi Zeroa. Tämän seurauksena voitiin myös tilata pienemmät kotelot sekä varavirtalähteet. Näiden testaaminen jäi kuitenkin tulevaisuuteen, mutta alustavat testit antoivat ymmärtää, että ne suoriutuvat yhtä hyvin kuin aiemmin käytetyt Raspberry Pi 3 Model B:t. Uudempi valmis laite on kuvattu kuvassa 34.



Kuva 34. Valmis laite, jossa ohjausyksikkönä Raspberry Pi Zero

6 Yhteenveto

6.1 Pohdinta

Hitsausajan seurantaan laite on melko toimintavarma kaikki seikat huomioiden. Se pystyy suoriutumaan tehtävästään hitsaustyön nauhoittamisesta luotettavasti, sen käyttäminen on helppoa sekä sen pitäisi sietää vaativaa ympäristöä. Laitteiden kokoonpano ei vaadi paljoa aikaa, joten niitä voidaan rakentaa monta kappaletta lyhyessä ajassa. Samalla kustannukset pyrittiin pitämään minimissä.

Laitteen olisi voinut suunnitella monella eri tavalla, mutta aiemmin mainittujen vaatimusten ja seikkojen ansiosta päädyttiin tämän tyyppiseen laitekokonaisuuteen. Koska kyseessä oli kehitysprojekti, niin tärkeimpänä prioriteettina oli tekninen toteutus. Laitetta suunniteltaessa nämä seikat pidettiin mielessä ja sen tuloksena määräytyi laitteen ulkomuoto.

Markkinoilla on olemassa valmiita ratkaisuja hitsausajan seurantaan. Tiedetyt valmiit ratkaisut voivat olla halpoja ja pienempikokoisempia, sekä niistä voidaan kerätä enemmän dataa kuin pelkkää hitsausajan mittaamista, mutta näissä valmiissa ratkaisuissa voi ilmetä muita sivuseikkoja. Esimerkiksi itse laite voi olla edullinen, mutta se saattaa olla luotu vain tietyn valmistajan koneelle.

Toinen seikka valmiisiin ratkaisuihin liittyy tiedonkeruuseen, johon vaaditaan oma palvelunsa. Valmistaja saattaa tarjota laitteen lisäksi palvelua tiedon keräämiseen ja esittämiseen. Tällöin itse laitteen hinta on toissijainen seikka, jos palvelun käytöstä koituu suurempi kustannus. Tässä projektissa luotiin laite, jolla pystytään nauhoittamaan hitsausaika riippumatta siitä, minkä valmistajan tai minkälaisella hitsauskoneella hitsaustyötä tehdään. Samalla tiedot säilöttiin Viafinin omalle serverille, ja sitä voitiin käsitellä juuri niin kuin sitä itse haluttiin.

6.2 Tulokset

Kehitysprojektin lopputuloksena saatiin luotua työntekijöiden työaikaa ja hitsaustyötä seuraava laite. Visualisoinnissa päästään tarkastelemaan dataa, joka sijaitsee Viafinin omalla serverikoneella. Kerätystä datasta nähdään helposti työpäivän aikana suoritettu hitsaustyö, ja tietoa voidaan myös tarkastella pidemmällä aikavälillä. Visualisointiohjelma mahdollistaa sen, että nauhoitetusta hitsaustyöstä voidaan luoda raportteja siltä aikaväliltä, mitä käyttäjä itse haluaa.

Laitteet toimivat itsenäisesti sekä tulevat olemaan jokaiselle työntekijälle henkilökohtaiset. Laitteita rakennettiin 20 kappaletta erissä, ja kehitysprojektin lopussa niitä oli käytössä 40 kappaletta. Laitteet tulivat aluksi käyttöön Viafinin sisällä, minkä jälkeen niitä aiottiin markkinoida asiakkaille.

IoT-alustana laite voi myös olla hyvin toimiva ja joustava vastaavia projekteja ajatellen. Laite on kannettava ja langaton, ja Raspberry Pi -tietokoneeseen on mahdollista liittää eri antureita sekä siihen on saatavilla eri ohjelmistoja tarpeen mukaan. Data pystytään välittämään langattomasti sijainnista riippumatta, ja laitteiden akut kestävät koko työpäivän ajan ja ne ehtivät latautua täyteen yön aikana.

Lähteet

- 1 What is open source? Verkkodokumentti. Opensource. <<https://opensource.com/resources/what-open-source>>. Luettu 16.1.2018.
- 2 Viafin Group. Verkkodokumentti. Viafin. <<http://www.viafin.fi/konserni>>. Luettu 16.1.2018.
- 3 Raspberry Pi FAQs – Frequently Asked Questions. Verkkodokumentti. Raspberry Pi. <<https://www.raspberrypi.org/help/faqs/>>. Luettu 16.1.2018.
- 4 Products. Verkkodokumentti. Raspberry Pi. <<https://www.raspberrypi.org/products/>>. Luettu 13.3.2018.
- 5 Installing operating system images. Verkkodokumentti. Raspberry Pi. <<https://www.raspberrypi.org/documentation/installation/installing-images/>>. Luettu 13.3.2018.
- 6 GPIO: Raspberry Pi Models A and B – Raspberry Pi Documentation. Verkkodokumentti. Raspberry Pi. <<https://www.raspberrypi.org/documentation/usage/gpio/README.md>>. Luettu 16.1.2018.
- 7 Raspberry Pi Starter Kit Lesson 2: Introduction of Raspberry Pi GPIO. Verkkodokumentti. Osoyoo. <<http://osoyoo.com/zh/2017/06/26/introduction-of-raspberry-pi-gpio/>>. Luettu 16.1.2018.
- 8 Generators and Motors. Verkkodokumentti. Molecular Expressions. <<http://micro.magnet.fsu.edu/electromag/electricity/generators/>>. Luettu 13.3.2018.
- 9 Node-RED:About. Verkkodokumentti. Node-RED. <<https://nodered.org/about/>>. Luettu 18.1.2018.