Expertise
and insight
for the future

Akseli Peltola

GPS device development and design for infrastructure builders

Metropolia University of Applied Sciences

Bachelor of Engineering

Information & Communications Technology

Bachelor’s Thesis

10 May 2021

Author Title Number of Pages Date	Akseli Peltola GPS device development and design for infrastructure builders 27 pages 10 May 2021
Degree	Bachelor of Engineering
Degree Programme	Information & Communications Technology
Professional Major	Smart Systems
Instructors	Sami Sainio, Lecturer Kimmo Sauren, Principal Lecturer

Underground cable locations are often poorly documented at best. This makes them vulnerable to damage due to construction work, which in turn can lead to power outages and, therefore to expensive repairs. In Finland, from January 2021 onwards, water, energy, communication, and traffic infrastructure owners must report all new infrastructure in cities with 10 cm and outside with 50 cm accuracy. Distance between measurement points on a straight line needs to be under 10 m and corners at most 1 m.

Wizense Oy won electrical network owner Caruna's innovation challenge competition in fall 2019 with a product concept that would be accurate, real-time, and easy to use. Early measurement tests were made using a GNSS module development kit to test if the accuracy would be enough.

The product development started with clarifying product requirements. A wet and dry working environment in the field and usability were driving conditions for the product creation. The product needed big buttons that can be used with gloves. The interface on display needed to be easy to use in an outdoor environment. Bright LEDs and a beeper were also necessary to provide feedback to the user. A threaded insert was required to the bottom of the device to mount it on the rover rod.

The first task was to make drivers for the OLED display and inertia module using the Nordic SDK. This module is used to get device orientation. After those two drivers, started 3D modeling with CAD for the product started. At the parallel to modeling started firmware development with Zephyr OS. OLED display and inertia module drivers needed to be ported to Zephyr OS. The modem and Global Navigation Satellite System (GNSS) module drivers were next. Then simple interface for the display was added for showing coordinates. To achieve the required accuracy, a ground plane was needed under the antenna, and elevation needed to be converted to a local height system. After the interface was ready to be used, accuracy validation against other GNSS devices and reference points around the airport and Metropolia Myyrmäki started. Accuracy met the requirements set for it.

Keywords	IoT, GPS, RTK, 4G, MQTT

Tekijä Otsikko Sivumäärä Aika	Akseli Peltola GPS device development and design for infrastructure builders 27 sivua 10 toukokuuta 2021
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tieto- ja viestintätekniikka
Ammatillinen pääaine	Smart Systems
Ohjaajat	Ohjaava opettaja, Sami Sainio Tutkinto vastaava, Kimmo Saureen

Suuri osa kaapeleiden sijainneista ei ole tiedossa tai dokumentaatio on heikkoa. Tämä voi johtaa sähkökatkoihin ja kalliisiin korjauslaskuihin, kun kaivinkoneet rikkovat kaapeleita vahingossa. Vuoden 2021 alusta Suomessa kaikki uusi viestintä-, energia-, vesihuolto- ja liikenneverkkojen infrastruktuuri pitää dokumentoida taajamassa 10 cm:n ja ulkopuolella 50 cm:n tarkkuudella. Mittapisteiden välinen etäisyys suoralla saa olla enintään 10 metriä ja kaarteessa enintään metrin välein.

Wizense Oy voitti Carunan innovaatiokilpailun syksyllä 2019 GPS-ratkaisun konseptilla, joka olisi tarkka, helppo käyttää ja data liikkuisi reaaliajassa.

Tuotteen kehitys alkoi määrittämällä vaatimukset tuotteen toiminnalle. Vaikeat työnteko-olosuhteet kentällä ja tuotteen helppokäyttöisyys olivat tärkeitä vaatimuksia tuotteelle. Samalla tuotteessa pitäisi olla isot napit, joita olisi mahdollista käyttää hanskatkin kädessä. Näytöllä oleva käyttöliittymän pitäisi olla yksinkertainen käyttää. Tuotteessa pitäisi olla myös kirkkaita ledejä ja summeri, joilla voitaisiin informoida käyttäjää helposti. Tuotteen pohjaan tarvitaan standardi kierre, jolla tuote kiinnitetään mittatikun päähän.

Tuotteen kehitys alkoi OLED:in ja inertiamoduulin ajureista, jotka kirjoitettiin Nordic Semiconductorin SDK:ta käyttäen. Tällä moduulilla saadaan laitteen orientaatio. Näiden kahden ajurin jälkeen alkoi tuotteen 3D-mallintaminen. Samaan aikaan aloitettiin laiteohjelmiston kirjoittaminen Zephyr OS:n päälle. Kehitys alkoi modeemin ja GNSS-moduulin ajureista. Samalla aiemmat ajurit muutettiin toimimaan Zephyr OS:n päällä. Kun ajurit toimivat, näytölle tehtiin yksinkertainen käyttöliittymä, jolta nähtiin GPS-koodinaatit. GNSS-antennin alle piti laittaa metallilevy, jotta tarkkuus saataisiin kohdilleen. Ainoastaan korkeus näytti useamman senttimetrin väärin. Tämä virhe johtui GPS-korkeusjärjestelmien erosta. Kun korkeus järjestelmä virhe saatiin korjattua, alkoi tarkkuustestaus muita GNSS-laitteita ja tarkkuuspisteitä vasten. Tuotteen tarkkuus oli samalla tasolla muiden laitteiden kanssa.

Loppukesästä tuotteen testaaminen aloitettiin kenttäolosuhteissa. Saadun palautteen perusteella tuotteeseen tehtiin monia muutoksia. Asiakkaat olivat samalla tyytyväisiä tuotteen helppokäyttöisyyteen ja tarkkuuteen.

Avainsanat	IoT, GPS, RTK, 4G, MQTT

Contents

List of Abbreviations

ARM	Advanced RISC Machine is a processor architect.
DNS	Domain Name Service converts www address to IP address
FPU	Floating-point unit is inside of processor and specializes in floating-point number calculations
GNSS	Global Navigation Satellite System
GPIO	General Purpose Input Output. Processor pins that can be configured as input or output.
GPS	Global Positioning System
HTTPS	Hypertext Transfer Protocol Secure is an internet protocol and used for getting and sending data.
I2C	Inter-Integrated Circuit is a serial communication bus for low-speed and short-distance devices.
IMU	Inertial measurement unit measures orientation
JSON	JavaScript Object Notation
LoRa	Is a proprietary low-power wide-area network modulation technique.
MCU	Microcontroller unit is a small computer on a small chip. It can contain multiple processor cores.

MQTT	Message Queuing Telemetry Transport is a commonly used protocol in embedded devices to send data to the server.
NTP	Network Time Protocol for getting the current time
RTK	Real-time kinematic is a satellite technology to get more accurate GPS positioning.
SPI	Serial Peripheral Interface is a synchronous serial communication interface for short distances and is used in embedded devices.
TLS	Transport Layer Security
UART	Universal asynchronous receiver-transmitter is an asynchronous serial communication protocol
USB	Universal Serial Bus is industry standard for connecting computers and peripherals
Zephyr OS	Real-time operating system for embedded processors

1 Introduction

Underground cable locations are often poorly documented at best. This makes them vulnerable to damage due to construction work, which in turn can lead to power outages and therefore to expensive repairs. A typical trenching model bulldozer team only does cable laying. The field survey team is following weeks or months after to measure the route using GNSS survey device and cable radar from the surface. This post-measurement approach is not accurate and has many flaws leading to quality issues. Also, the trenching documentation was kept on the white papers. Changes were marked with a thick pen causing change tracking not to be accurate and a lot of manual work post-processing the changes made to white papers. Also, image documentation is required, and the delivery process was not constant due many different tools used. Quality control during the installation of the cabling is essential.

In Finland, Traficom M71 requirement requires all new water, energy, communication, and traffic infrastructure owners to report all new infrastructure in cities with 10cm and outside with 50cm accuracy from January 2021 onwards. The distance between two measurement points on a straight line can be at most 10 meters. More measurement points are needed if trench depth changes more than 10 cm. In corners, measurement needs to be more frequent to capture curve shape. Below are two examples of the measurements. Green dots present the correct way to measure point, and red dots are presenting the wrong way. (Figure 1) [1]

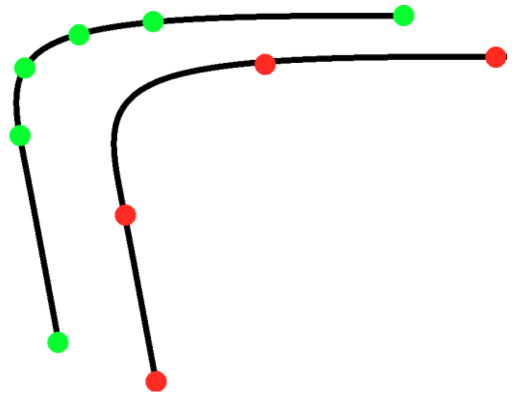

Figure 1. Measurement points

The principal company (later referred to as the company) won Caruna's innovation competition in fall 2019. Caruna was looking for new ways to document cable infrastructure location. The company had an idea to solve this problem with new positioning device (later referred to as the product). At the point of thesis completion, the first production version is almost ready for a broader market.

The significant benefit of the company product is that all these customer requirements are solved in real-time with a one-button press. It delivers data to cloud service and into network infrastructure API interface. This change saves time because manual work and post-measurement with cable radar are not needed anymore.

Cable trenching teams have been using the product since summer 2020. During the testing phase, the field operators provided feedback, which was used to improve the product, resulting in a better end-product.

This thesis is empirical documentation of how the product and logic, including user experience, were build up from an item to a product. The goal of this work was to make the embedded software and mechanics. The first task was to develop drivers for the OLED display and IMU module. These both are now part of the product. In the early year 2020, tasks were to model mechanics to the product and learn how Zephyr OS works. (Figure 2)

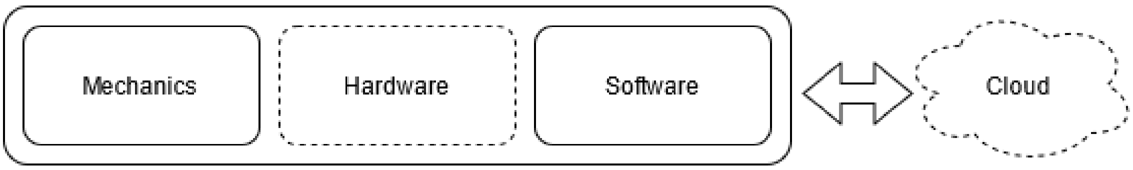

Figure 2. Responsibility areas

2 Technology

2.1 GNSS

GNSS device uses multiple satellites to triangulate user position. Four global navigation systems are useable in Finland. The United States launched their first global navigation system (GPS) satellite on the year 1978. It has operated globally since 1994. Russia has their navigation system called GLONASS. The first satellite was launched on the year 1982, and complete global coverage was reached in 1995. The Chinese have their navigation satellites also. This system is called BeiDou. The first satellite was launched in the year 2000. European Union's satellite system Galileo got the first satellite to the sky in 2011. Fully functionality is expected by 2021. [6] [7] (Table 1)

Table 1. GNSS Systems [6] [7]

	GPS	GLONEASS	BeiDou	Galileo
Omistaja	United States	Russia	China	European Union
Ensimmäinen laukaisu	1978	1982	2000	2011
Satelliittien määrä	31	27	49	26
Maailman laajuinen kattavuus	1994	1995	2018	2016

All these satellite systems are widely used in different applications. Not just GNSS measurement devices but also multiple applications when location is needed. The product's location technology is based on these satellites. The product can use all the systems simultaneously. It is better, because near big buildings and forest areas, signals are not ideal. It helps if the product detects more satellite systems, not just one.

2.1.1 RTK

A GNSS device needs a correction signal to achieve around 1cm accuracy. Real-time kinematic positioning is commonly used to get more accurate locations in land surveying and construction sites. This correction data can come from two different places. Usually, construction sites have a physical base station. The base station position needs to be known. Suppose its position is 5cm away from the correct location. In that case, all devices that get correction data from that base station are mismeasuring points with 5cm.

This correction data can be sent over a cellular network or locally over short-range radio. The local radio approach is used commonly in construction sites. It is easier because users do not need to change credentials when moving between locations. When correction data comes through a cellular network like 4G, a user needs to sign into a separate server to receive data. The image below shows how the GNSS measuring device is getting correction data from a base station. (Figure 3.)

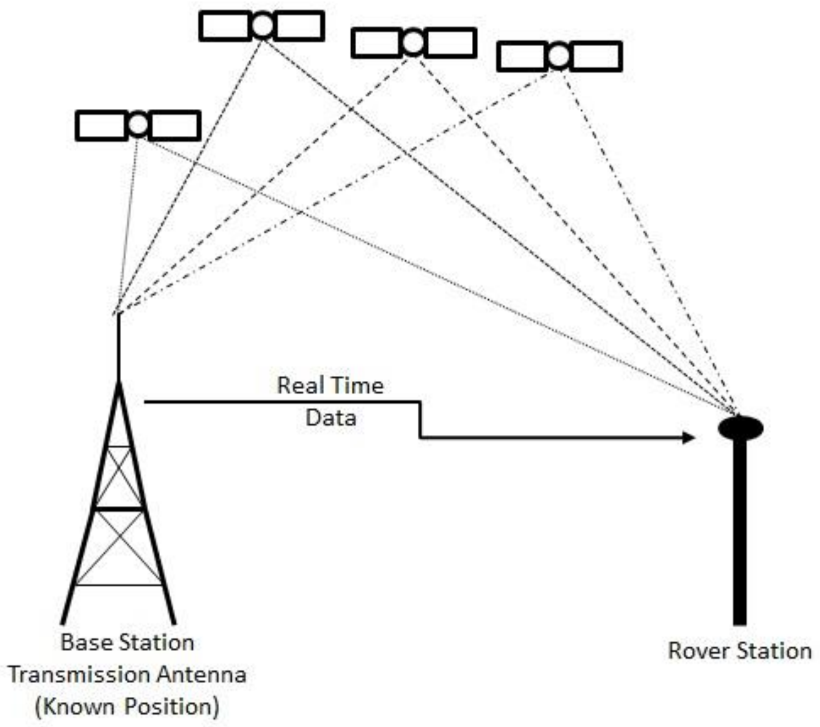

Figure 3. RTK GNSS data flow [8]

The second option is to use correction services from Leica or Trimble. These services cost several thousand euros each year. One license used for one device. These companies use their own base stations around Finland to make virtual reference stations as near to the user as possible. These virtual stations are not as accurate as a local base station, but accuracy is still under 2cm. [9] [10]

There are two states to this correction. When a device first gets corrections data stream from the base station, the GPS state changes to RTK float state. In this state, accuracy is already more accurate than the normal GPS state but not in 1cm accuracy. After some time, the state switches to RTK FIX. In this state, GPS location is confirmed, and accuracy is around 1cm. A GNSS device needs to see multiple identical satellites than the base station, to get an RTK FIX state.

2.2 Zephyr OS

When driver and firmware development started, Nordic Semiconductor's SDK was used for software development. It was not the most user-friendly because there was a multi thousand-line config file for enabling different peripherals and features. There was not any threading support, and everything would have run on one loop. The latest Nordic processor nRF5340 does not even support Nordic SDK. It is only using Nordic Connect SDK, which is based on Zephyr OS. [11]

Just before the product's actual firmware development started, the product firmware moved to use Zephyr OS. It is an open-source real-time operating system for embedded devices. Zephyr OS development began in November 2015 by Wind River Systems, and the project was named Rocket. Zephyr OS become Linux Foundations one project in 2016. Currently, multiple big companies are supporting this project, such as Intel, Google, and Facebook. It supports multiple threads, various boards, and is easier to use than the one provided by Nordic's SDK. [12] [13]

2.3 Bluetooth

On the product, there are two kinds of Bluetooth communication. Bluetooth low energy is used for communication with depth handle and later with a phone to configure settings. The product has a separate Bluetooth Classic module for sending GPS NMEA data to third-party devices like phones and tablets. Bluetooth Classic module is needed because Bluetooth inside the processor is only supporting Bluetooth low power.

2.3.1 Bluetooth Classic

Bluetooth Classic is using the serial port profile to send between devices. It acts like UART between devices but wirelessly. It is used on most external GPS devices to transfer NMEA GPS data to an end device. The maximum payload size on this protocol is 128 bytes. Some of the customers need this feature to get data to their applications. Bluetooth Classic is also required when pairing the product with a third-party cable radar. This radar is used to post-measure cables, and it uses Bluetooth Classic to send measurement data. [4]

2.3.2 Bluetooth Low Energy

Bluetooth low energy is behaving entirely differently than Bluetooth Classic. It has a tree-like structure that contains services and under them characteristics. Characteristic is one value, and the limit is 23 Bytes, but devices can agree on the size. This maximum size is 512 Bytes. On the product, the depth handle writes depth value to correct characteristic, and interrupt gets triggered to notify the application. [5]

3 Product

3.1 Mechanics

Mechanically the product is used in harsh field conditions needs to be tough. The product meets moisture (rain, slush, snow), dry, and warm (+60˚Celsius) conditions in regular use. Material thickness and materials used are a big part of product design. An external company made the prototype design, and few versions of it were 3D printed to test PCB fitment. On to the bottom, this product needs a standard 5/8x11-inch thread that is used in all GPS measurement devices. This thread is used for attaching the product to 2 meters long GPS rover rod.

3.1.1 Design Process

One limiting factor for the overall device dimensions was the size of the panel. The panel needed to have big buttons and enough space for an OLED display and few LEDs. Some design considerations for the chassis that were found important in this project were the rubber bumper, enough space for the interface panel, and small size. A rubber bumper is needed to take most of the collision when the GPS survey equipment may fall from a 2-meter height. It also would protect the display from getting impacts when dropped to straight to the ground. Versions were 3D printed out because it is a cost-effective way to iterate product design. (Figure 4)

Figure 4. First versions of the product

On the next version, corners were straighter and inner parts were also 3D modeled and printed out to test PCB fitment. The left model was also optimized inside to have ribs next to screw towers for improved durability. Everything had a slight taper to them. This taper helps plastic parts to be removed from the mold more easily. At this point, the design was ready to be made with a silicone mold. However, a few days before launch, the whole product needed to be remodeled because of small, not smooth lines on the lid and body corners. The right image is entirely built from the ground up again to make corners smooth. After a two-day remodel, parts were ordered from silicon plastic molding place. Same time black rubber bumpers were ordered. (Figure 5)

Figure 5. Final designs

Inner parts were 3D printed. The only part that might break inside is the battery holder. It needs to support two 190g Ni-MH batteries. All other components are acceptable to be printed because weights are low. All PCBs are also on a separate moving plane with a rubber insulator dumber. They block minor hits and vibrations from going to PCBs.

This same product can also be used on cable ploughing. In that use case, forces are much higher than on top of the stick when measuring. The material thickness needs to be higher and better support around the thread area. In initial customer testing of cable ploughing, the bottom thread area broke off. This area needed enforcement in the next iteration.

3.2 Hardware

There were two iterations of the product hardware. Both versions are using almost the same hardware but a little bit different configuration. The older version had a separate GEO PCB with GPS modules, all interface panel components, and powers. Now on the second iteration mainboard has GPS chips. The power module is now also separate just because it is usually the first thing that may break. Interface panel controls and OLED screen connectors are now on separate small PCB. OLED screen cable is too sort for it to reach the mainboard. The Modem board is an independent board on both versions, but the newer modem also supports 5GHz Wi-Fi and Bluetooth. There is also output for a buzzer to give indication to the user. It notifies the user when the measurement is finished or when it has failed with different beep sounds. (Figure 6)

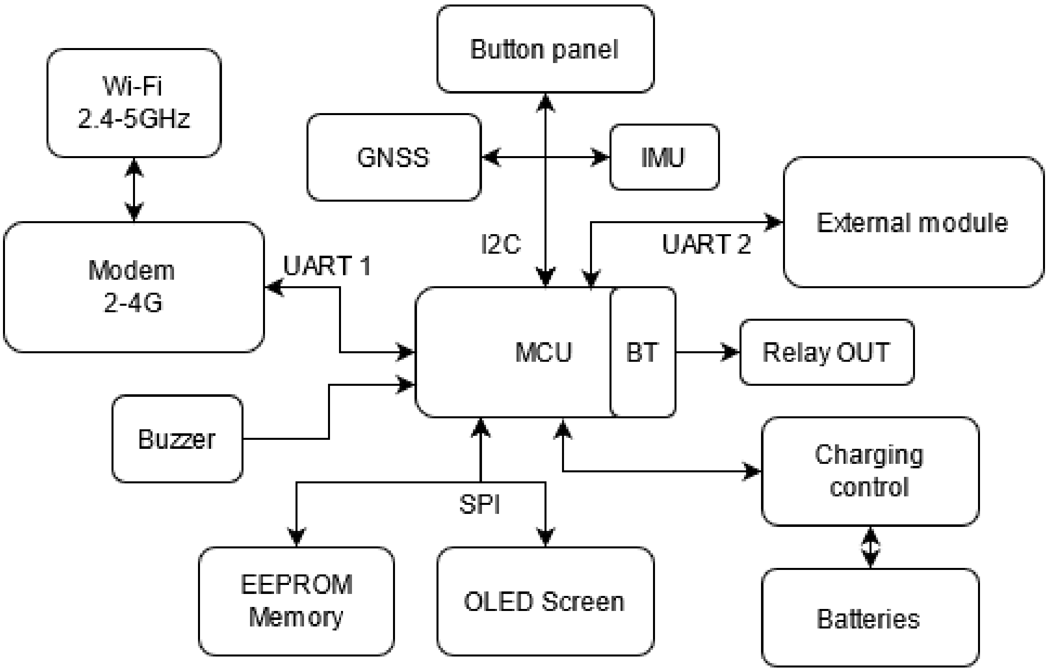

Figure 6. Hardware layout

Relay is used to power up the external Bluetooth Classic module. It is just a workaround because there is only a 3.3 V shutdown circuit. 5 V is always on, and this external Bluetooth Classic module requires 5 V to operate. When 3.3 V power is shut down, the processor still gets some power through UART lines from this external module and keeps the processor I/O powered. Before the product is shut down, this relay needs to be turned off to make the whole product turn off.

3.2.1 Processor

The product mainboard uses Fanstel's BT840F module. This processor is based on ARM Cortex M4, and it has a separate FPU. The processor has 1 MB flash and 256 KB RAM. A small amount is reserved for application usage and bootloader. The rest is cut in half. When updating, the new firmware is loaded to one of these slots. If the update fails, it does not corrupt the whole product and can start an application from another slot. When developing, J-Link is used for programming this processor. In this case, the bootloader is not needed. When optimizations are turned off for debugging, the firmware size is almost double, and it might not fit to flash if it is split in half. This processor also has

Bluetooth 5.0 low energy for communication with other devices like depth handle. This handle is used for getting trench depth and triggering measurement. [2]

3.2.2 GNSS

To the product achieve the required accuracy for infrastructure documentation, the product needs a high-end GNSS module. Higher-end modules can use more satellites due to the increased number of channels and keep multiple links open to one satellite. This module can get sub-two-meter accuracy with an active antenna. However, it is not enough for cable location survey. GNSS module needs connection data. This Real-Time Kinematic data comes through the 4G to the GNSS module. After opening this data stream, the product's GPS accuracy first goes RTK float state. When the state changes to RTK, the position is confirmed with a precision of 2 cm or better. In some cases, the module has been in FIX state, and the accuracy has been over 10 cm.

On the mainboard, there is also a reservation for the second GNSS chip. This chip is not production-ready. It could provide RTK correction data stream through satellite connection. It cannot achieve 1 cm accuracy like RTK stream through the cellular network, but 10 cm would be enough for cable trenching outside of cities.

When measurements are taken outside the cellular network range, the product needs to get RTK correction stream from some other source than the cellular network. In this case, all measurements are stored in memory. When there is an internet connection again, the product will send all saved samples to the cloud.

3.2.3 IMU

The inertial data module is a powerful component that can compensate for measurement data when the measuring stick is not straight. When rotation vector operation is enabled on this IMU chip, it uses an accelerometer, gyroscope, and magnetometer to make a quaternion referenced to magnetic north. Usually, orientation data needs to be read from the sensors separately and then combined mathematically in the processor. It is harder to do and takes some processing power from the processor. [3]

3.3 Software

Software side, the product usage needs to be made easy for a user. Below is illustrated what screens are on the product. At first, when the product booted, it went to customer select and then straight to project select. Project select went to the main screen. After a while, a 4-digit PIN code was added to protect customers from seeing each other projects. Now when the product boots, it goes to the 6-digit project select screen. These codes are randomly generated for every project. Users can still use the old customer and project select screen by entering six zeros to the project code. The screen where the user can select the team is also something that was needed because one project can have multiple teams doing trenching. (Figure 7)

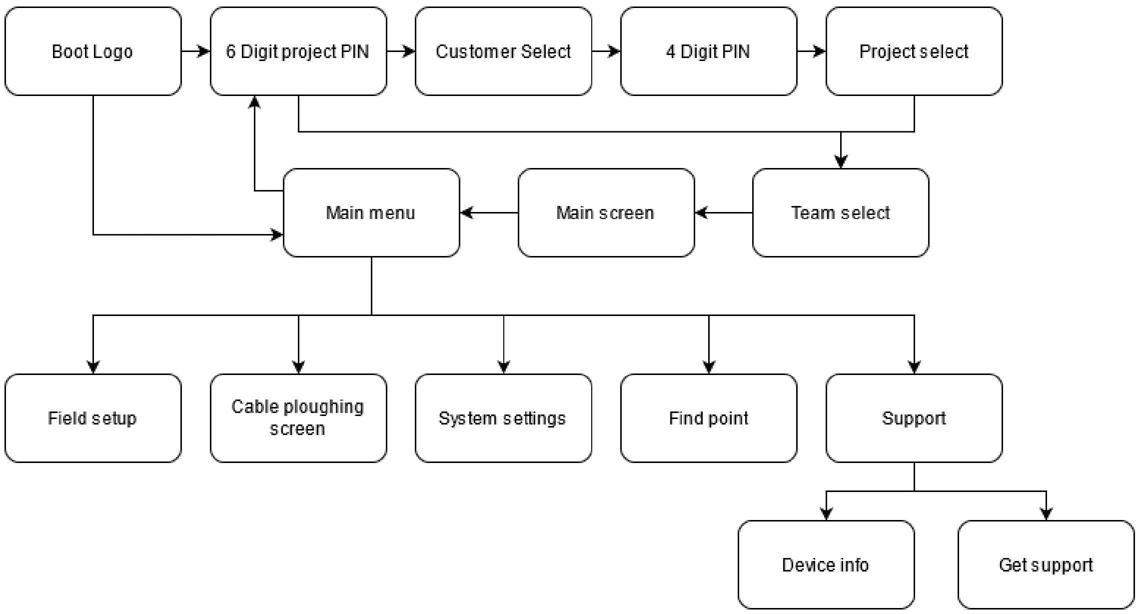

Figure 7. Menu structure

3.3.1 Usability

Some cable trenching teams might have two bulldozers. One creates a trench, and another comes behind and puts the dug land back to the trench. There is a shovel person laying cable down to the trench and a new role making these cable survey measurements between these bulldozers. If the product had errors in the field, it would lead to a bad user experience and become expensive when everyone is waiting for measurements from the open trench.

The device usage is made user-friendly. When the product boots up, the user needs to enter only one project-specific code to start measurements. When the project code is entered, the user needs to select the correct measurement team. One trenching project can have multiple teams making progress in different parts of the project.

On the main screen, the user can see the current customer's name on top. Next to it, there is a satellite count to indicate how many satellites the product is seeing. There are mobile network indications for cellular network type and strength also on the top line. In the top right corner, there is a battery indication bar. On the following line, the user can see the current project name. Under that user can select whose cable is being measured in the trench. Every network owner has different types of line- and point objects. Protection classes are usually the same, but numeric codes for those can be different. If one trench has multiple cables for different network owners, the user needs to switch this field according to that. This option might be made more accessible by configuring these settings possible before measurement and just measuring once.

On the following line, the user can select line or point measurement. On that same line, the user can see what the current point/line number is. If the user presses Enter key on this line, the point/line number can be modified with arrow keys and press ESC to exit. When the line ends, it can be terminated by pressing ESC. The following line on the screen tells the user what kind of line or point object is measured.

Under this line, there is the depth field for measured trench depth. This value comes from a separate device, and it also can be overwritten with arrow keys. When the user changes this value manually, it is marked to the user-changed value to measurement data. Next to the depth value is GNSS module horizontal accuracy. The last field tells the user what type of protection is used for the cable. This protection class setting is made easy for the user to change. It is the most used setting because measuring cable the protection on top of the cable can change frequently depending on the environment. The most bottom line always tells the user some info about what is happening. (Figure 8)

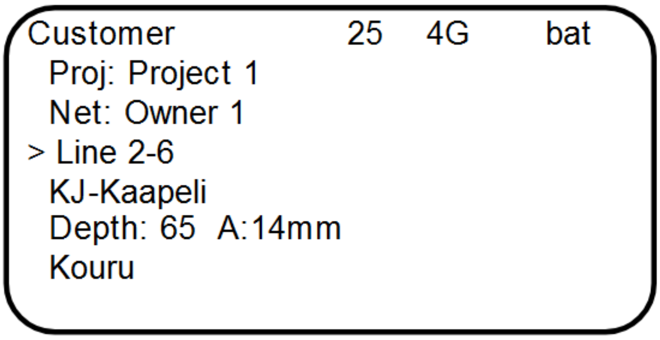

Figure 8. Main screen

Users can access the setting by pressing the enter key on the main screen. Under every menu item, there is a sub-menu for changing the different settings. The user can go back to the 6-digit project code screen from this menu when the project needs to be changed or change team.

Inside field setting there is most of setting that user can change. The first value on this menu tells the user how long the rover rod is. Usually, rover rods are 200cm high. If this value is not correct, elevation measurement is not correct either. The next value tells what trench width is. This value is usually 40cm, but the trench is generally wider when putting many cables to the same trench. The following line tells how often the product sends its position to the cloud. The cloud has a UI that shows the product position and the planned route. Trenching teams can follow the planned route easier with this feature.

The user can change how many samples the device takes when measuring a line or point object on the following line. Currently, 4-5 samples are possible to be taken in a second. Now there is one thread that is polling data from the GNSS module through I2C. The following setting changes minimum accepted GPS module state. This value can be changed to FIX, FLOAT or GPS. GPS means that everything is accepted. The following line limits maximum accuracy. All samples that are underspecified limit are saved. It also can be turned off. The user can change the RTK service on the following field if the selected project has multiple RTK licenses allocated to it. The last item on this menu limits how far points can be when measuring a line. The image is just for reference. All these options will not fit to screen at once. (Figure 9)

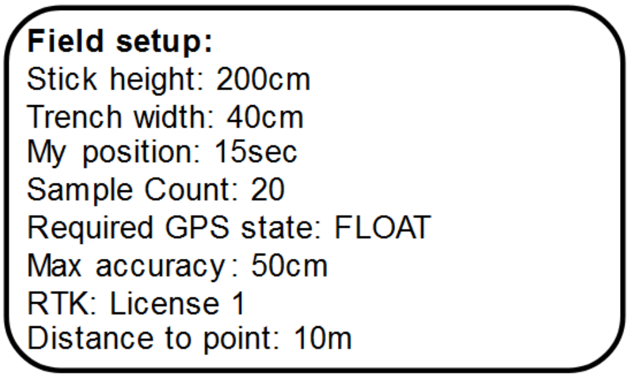

Figure 9. Field setup screen

Inside the system setting, there are not many settings for the user to change. The first setting for selecting location data source between NMEA data and straight module read. This option is not needed anymore because all values are read straight from the module. The following setting is for changing the main screen type. There is a type for accurate measurement of latitude, longitude, ellipsoidal elevation, and horizontal and vertical accuracy. There is also an option for a test development screen. Calibration and self-test are both placeholders. When IMU is going to be in more extensive use, then calibration needs to be implemented too. After that, there is an update button. If an update is available, there is yes, and if not, there is no. By pressing enter when an update is yes, the product downloads the latest firmware and installs the update. This functionality was not yet implemented during the thesis completion. UpdateHub is used for over-air firmware updates. It is already part of Zephyr OS. This update method was tested. Next on the list user can release the RTK license. It stops RTK stream and tells the cloud that this license is free to use on some other device. It should happen when the product power is switched off, but it is not possible because the power button turns the power off without any processor interaction. The last system settings option is reset. It erases all settings from EEPROM and reboots. (Figure 10)

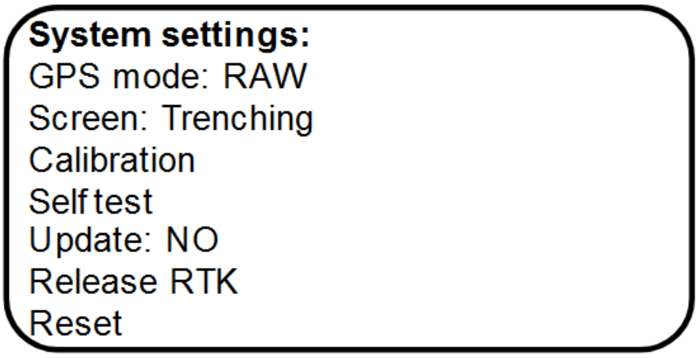

Figure 10. System settings screen

In the main menu, there is also an option for cable ploughing. It opens a new screen where a user can see the same top row information as on the main screen. It also is showing the current project. A user only needs to change the offset value before starting ploughing cables to the ground. This value tells the product how far the distance is from the bottom of it to where cables are laid. When cable ploughing mode is turned on with a play button, big ON text appears on the OLED display, and OK LED starts to bling, notifying the user. A user does not need to press any buttons after this. When the product moves a set distance, the point is saved.

In the find point screen, the user can select a point that needs to be found. Currently, points are statically defined on the code side, and this needs to change at some point that points come from the cloud when the product requests them. When a point is selected, the user can see the difference between latitude and longitude values compared to the selected point. There is also a difference between elevations, but it does not match because of height system differences. The last line tells the distance to the selected point in meters.

The last menu item in the setting screen is for support. The user can send a support request with a phone number and the product data to the company support. There is also a screen for device info in the support menu, but it is currently still empty. It should show battery voltage, mobile network RSSI, mobile operator, and many other values that the user might want to see.

4 Firmware

In a nutshell, the product measures a point and sends it then to the cloud. All settings are in separate EEPROM memory to keep them safe when the product is shut down. Next time, when the product starts, all settings are read from EEPROM, and the user can continue measurements.

4.1 Logic Description

When the product boots up, all device drivers are first initialized. I2C I/O expanders get initialized first. They are used to turn on and off multiple LEDs, power on modem, control relay, and buzzer. There is also one chip for controlling buttons on the panel. Most of the pins in the processor are used for different interfaces like I2C, SPI, and UARTs. (Figure 11)

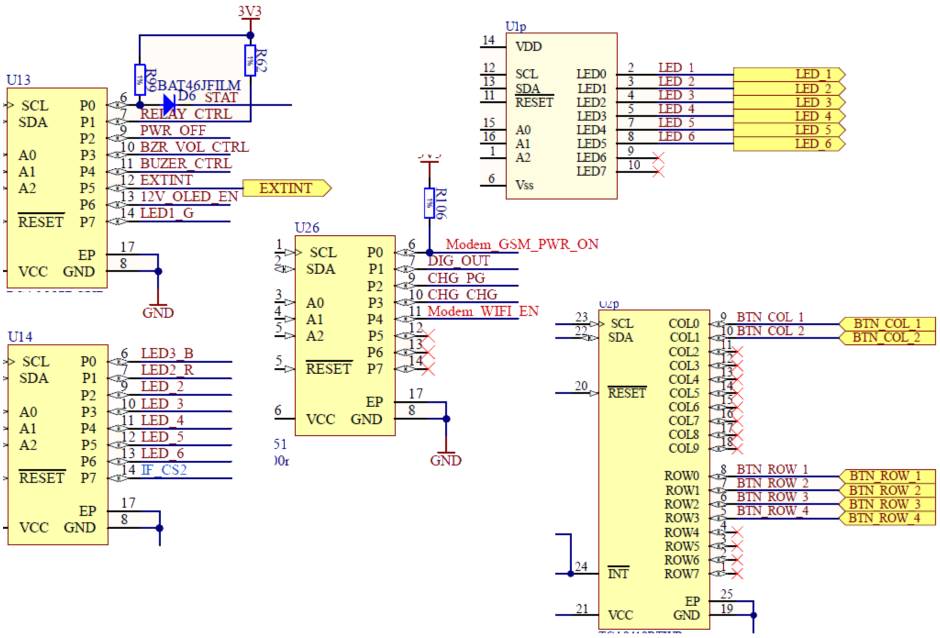

Figure 11. I/O expanders

After that OLED display is initialized, it shows the user what is happening, and the boot logo appears on the screen. GNSS module initializes next. It is switched to high precision mode to get more decimals out to NMEA strings. It also unlocks two more decimals when reading straight data registers. All unnecessary NMEA messages are also turned off, and UART speed is set to 115200 to get RTK stream from the modem. After this modem is initialized and it starts up. If the SIM card is not installed, cellular initialization is skipped. If the product does not have a network connection when it goes to the main, it cannot perform anything.

After that, Zephyr OS starts executing code from main. The main is first getting references to all control panel LEDs and buttons. After panel initialization, the application can access them. Then the product makes a new thread for reading GPS position data from the GNSS module. Then it reads the cloud access key from flash. If the flash does not have this key, it asks the cloud new key from the cloud. When UpdateHub updates, work updates are checked next. If an update is available, firmware notifies the user, but it does not update the product. Then Bluetooth low energy is initialized, and an external Bluetooth Classic module is turned on. That private key that was fetched earlier is now used for connecting to the MQTT server. When this connection is established, the product reads project data from EEPROM memory. If there is data from previous usage time, it connects to the RTK service and enters the main loop. (Figure 12)

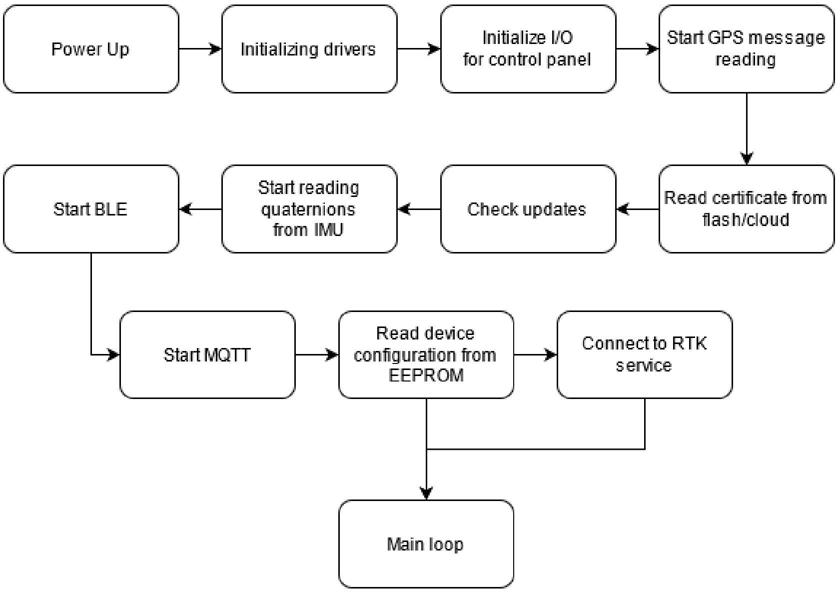

Figure 12. Firmware logic

There is also a possibility that EEPROM has not any data. In that case, the product starts executing the main loop, and the user is put to a 6-digit project pin screen as described earlier. The main loop is updating the screen and LEDs every second. It also updates the latest data from sensors to data structure. When the user presses a button, the screen is updated right after when press happens.

4.2 GNSS Module

The modem UART transmit pin is connected to the processor, and it is also connected to the GNSS module. GNSS module can filter out RTK data from the UART data stream. Without this correction data stream, GPS accuracy with an active antenna is somewhere around 2 meters. When the correction stream is enabled, accuracy gets to around 1cm depending on how accurate the correction service is. When taking measurements, local base stations close to the product are more accurate than paid services such as Leica Hexagon Smartnet or Trimble VRS.

Sometimes the Hexagon service has some problems when the 4G connection is not most optimal. At first, the data stream comes correctly. Then there is a slight pause, and then there comes big junk of data. It might happen few times, and after that, the product work as expected.

Every time the product starts up, it asks from the cloud available RTK license list. I2C is used for reading GPS data from the GNSS module. Now the main processor is asking for data and reading it as fast that it can. UART would be a much better way to do this because when the GNSS module sends data, an interrupt is triggered. However, the product's processor has only 2 UARTs. One is going to the modem and the second one is for external use. It is used for a separate Bluetooth Classic module or LoRa module.

GNSS module outputs standard NMEA data by default. There are also commands to read data straight from the module and get more data about GNSS module accuracy. There are registers for horizontal and vertical accuracy. Latitude longitude values can also be read with 9-decimal accuracy. In millimeters, it is about 0.1mm. It is more decimals that are needed.

Elevation was initially a problem. It was multiple centimeters off from actual measurement. The GNSS module was using the global height system WGS84. All Finland's reference points are reported in the N2000 height system. Ellipsoidal height (h) is needed to convert between WGS84 and N2000 height systems. Ellipsoidal height is read from the GNSS module and send to the cloud. In the NMEA sentence, there is a mean sea level (N) and geoid separation (H) that can also be used to get the ellipsoidal height (h). (Figure 13)

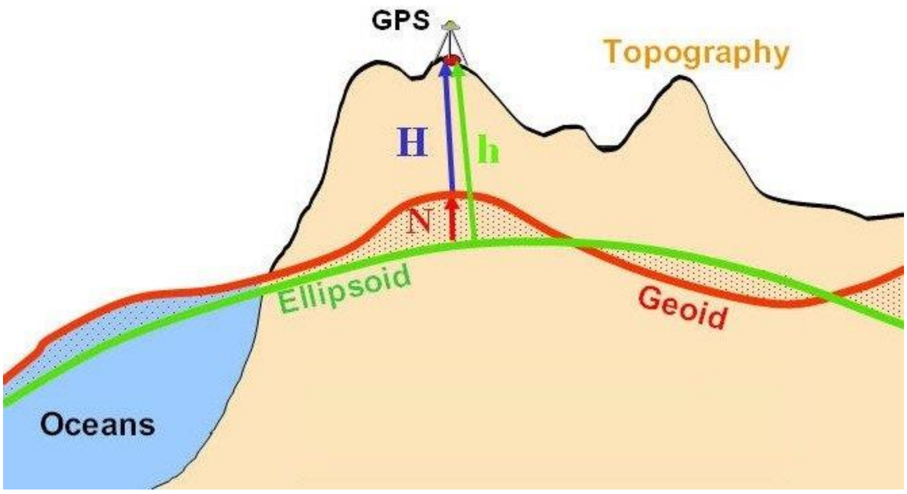

Figure 13. Height systems [14]

This conversion is done in the cloud because a large file is needed in this conversion, and the embedded processor does not have enough space for it. This file contains height values for all latitude and longitude values in Finland. Cloud finds the closest values from the matrix and calculates the mean from them. (Figure 14)

		17.48°	17.52°	17.56°
		↓	↓	↓
70.70°	→	34.415	34.318	34.219
70.68°	→	34.433	34.336	34.235
70.66°	→	34.450	34.350	34.250

Figure 14. Elevation conversion file [15]

4.3 Modem

The product needed a driver for the Quectel modem to get data to the cloud. Zephyr OS has multiple drivers and examples of using them. However, the modem that is used inside the product was not included in them. An already existing modem driver was used as a base. After a few weeks modem driver was working. It still had bugs, and features were missing. There have been multiple bug fixes to get it to the current state. When the application makes a new socket, it is automatically calling modem driver functions. In the product, there are always two sockets open. One thread is sending MQTT data to the cloud, and another is receiving RTK data. One bug was also on the Zephyr OS modem socket side. Only four connections could be opened and closed. If the application tried to open the fifth socket connection, it would not work. By commenting out one line, it worked again. This problem is now fixed on the latest Zephyr OS release.

Since the first time making this driver, there have been significant changes. It did not support multiple connections through Wi-Fi. This support is now added. Same time BT, Wi-Fi, and cellular functions were moved to their own C-files to keep file length reasonable and make functions easier to find.

4.4 Display

The OLED screen is one color consisting of 128x64 pixels. This driver was made using the nRF52840 dongle. Devkit version was needed to get better logging support. This OLED display contains eight pages on every column. One page has 8 pixels. If just one pixel needs to be updated driver needs to write the whole page 8-bits. OLED display uses SPI, and there is no MISO to read current values from the screen. It made driver development a little bit harder because there was no way to read data from the OLED to see if the configuration was written correctly. There are two buffers, the current state and the next state, on the driver's side. When OLED display update happens, only needed registers are updated. First, the OLED ZIF cable was the wrong way around, and it did not work because of that, and registers were configured wrong. When the OLED screen was work, it needed 3D printed mount. (Figure 15)

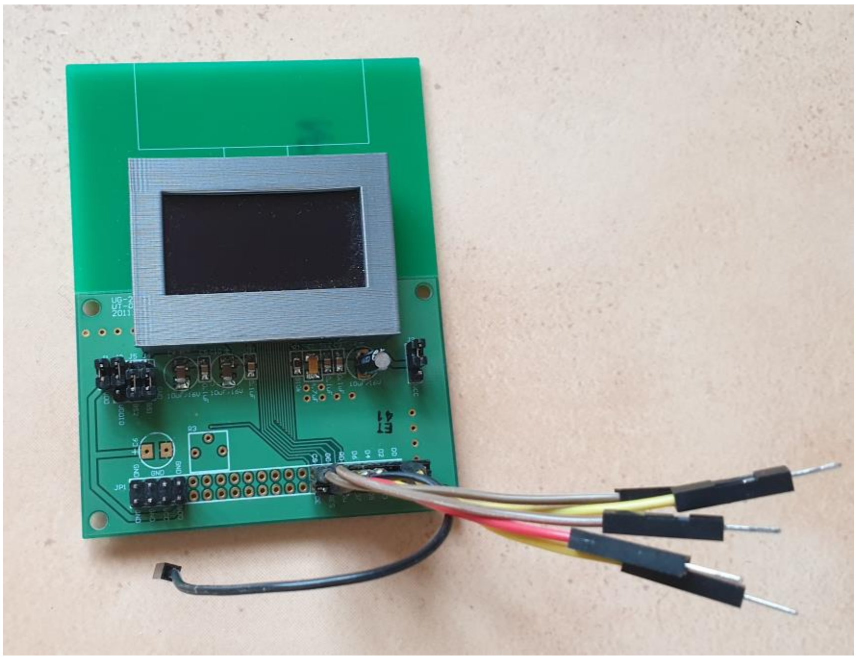

Figure 15. OLED devkit

4.5 Inertial Module

IMU driver was developed with nRF52840 dongle and a separate Sparkfun BNO085 module. This IMU has processing inside, and it gives user rotation quaternions. For testing, a simple mobile application was made to receive quaternions over Bluetooth Low Energy. (Figure 16)

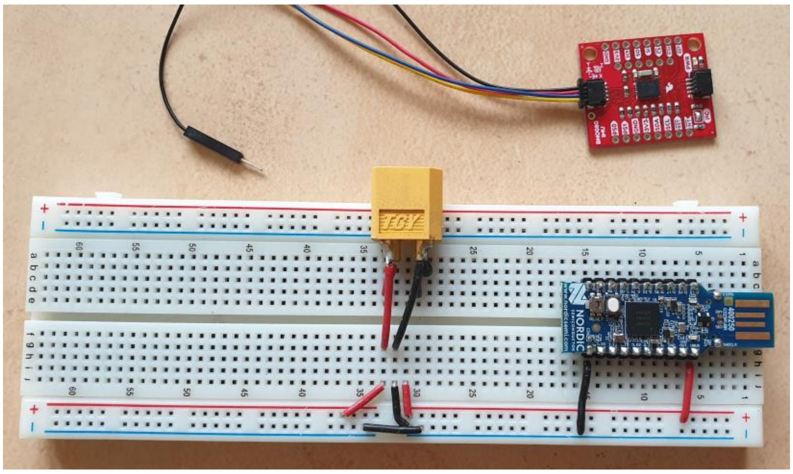

Figure 16. Nordic dongle and Sparkfun BNO085 IMU

5 Field Testing

When the product got stable build iteration, field-testing started. Early summer 2020, there were testing days on Mondays with different trenching companies. The product got early feedback from every site. After every field day, the rest of the week went for making changes based on the feedback. There were more ideas for a different product, like make it split because it gets muddy and may get stuck to the rover rod. On these test days, there were problems with the RTK correction stream. Sometimes it rebooted and did not want to work at all. However, after several reboots, it worked just enough to demo it. The first pilot started with a trenching company during summer 2020. Later that fall more companies started testing. Multiple days went in the field, visiting customers and updated software to fix bugs. The product started to work more reliably after each update.

Accuracy testing was also done in co-operation with Metropolia survey engineering department. Around of Myyrmäki Metropolia campus are multiple known accuracy points. In the early device version, NMEA data was used to get latitude and longitude values. Coordinates from NMEA were converted to floats but later noticed that float was not accurate enough. Double is more accurate than float, but it was not used at first. Coordinate values were converted to character arrays when calculating latitude and longitude values from NMEA sentences. Integer and fraction parts were calculated separately with integers. It worked, but when ellipsoid elevation and module accuracy was needed from the GPS module. Latitude and longitude values were also read from it straight. NMEA string was not enough anymore. At some point, double calculations were tested with the processor, and they worked. Processor FPU just does two operations. After that, all latitude and longitude values were changed to doubles from char arrays.

One of the company's customers helped to compare accuracy to Leica GNSS devices at the airport with a local base station and around Metropolia. Metropolia's land surveying students made accurate test points around the Myyrmäki campus with a total station. Accuracy on the product was as accurate that other GNSS devices that were tested. Accuracy was tested with five different antennas. Small antennas did not have any ground plane to block signal reflections from under them. At the first test without any ground plane, accuracy was not anywhere near 1cm. Then a ground plane was added under smaller antennas, and accuracy got better than it was. It also helped to get the GPS FIX state more easily. More expensive antennas have ground planes integrated into them. They were a little bit better but not much. Some of the cheaper ones got RTK FIX state much quicker.

After fall 2020 and when the trenching season ended, there have been various updates to data sending and cable ploughing. Users do not need to wait for any more data uploads. There is a separate thread that processes the queue from EEPROM. Cable ploughing works as well. When it is enabled, every time the product moves over a selected distance, it saves that point to EEPROM. After multiple saved samples, it sends them to the cloud in JSON format through MQTT. This feature earlier test version was tested on customer trenching site, but the product mechanics broke in ploughing testing. The second feature that is now tested is triggering 360-camera on every point and line measurement. Photo sending to the cloud is currently taking over 30-seconds, and the user needs to wait every time after measurement. In this area, there is still room for improvement.

6 Conclusion

In this study, the product development was done in close collaboration with customers. Feedback from the field has been positive. There have been many iterations of the product during the process, and it keeps improving in increments. The product development has not been only C-coding but also 3D modeling and using a 3D printer and assembling the products that are going to the field. The product met all requirements that were required at the start. Accuracy got to the required level after many software and mechanical changes. All measurements are in one cm range compared to accurate reference points around Metropolia Myyrmäki campus.

This same product can be used in other land surveying use cases. One of these use cases is making cable trenching route planing before trenching teams dig cables to the ground. These plans are usually made using expensive GNSS devices, but now cable contractors could use this product instead.

6.1 Other Applications for GNSS Technology

This same cm positioning technology can be used in other applications, such as autonomous vehicles and machines. The same hardware can be fitted to a smaller form factor without large Ni-MH batteries, interface panel, OLED display. The internal antenna can also be removed, and an external antenna can be used instead. There was a customer requirement to understand if they were harvesting trees from the right side of the property border. This smaller device was used to give accurate GPS data to the forest machine operator through Bluetooth Classic. Before this, they used GPS data from a mobile device that was not accurate enough to know the forest machine's actual location. This device can be used in multiple other use-cases, such as retrofit to an older generation of machines not having accurate positioning technology built-in.

Another application for this same technology is in autonomous vehicles with all other sensors for sensing the environment. GNSS technology can provide an accurate position for these vehicles to make them stay in the correct lane. Before this, all road systems need to be documented more precisely to benefit from using GNSS technology for autonomous vehicles. When road location is accurately known, autonomous cars could make corrections to driving direction by comparing their own location to accurate map data. If every car would use this technology, they could know each other's position on the road. This would lead to fewer traffic accidents, and traffic lights are not needed anymore. This same concept could be used on a smaller scale in freight yards and other smaller places where autonomous vehicles would make operations more efficient.

6.2 Next Steps

The next steps for the product iterations are going to be a smaller design. Other high-end product requirements include hot-swap batteries. There is a need for improvement in UART about up to three to make it run smoother on the hardware side. Almost every construction site is using UHF radio to give correction service to nearby devices. Before that LoRa module is put into the product to make communication between two devices more accessible, it is used to give a correction signal from base to rover. Firmware needs to be optimized to consume less power and turn off automatically when the product is not used. There is also a need for Wi-Fi support. Currently, all data is going through a cellular network. In some places, cellular connectivity might not work. In these cases, a Wi-Fi access point would be used to send data to the cloud and receive the RTK correction data stream.

References

1 Traficom M71 requrement. Online. URL: https://www.traficom.fi/sites/default/files/media/regulation/M%C3%A4%C3%A4r%C3%A4ys_71_verkkotietojen_ja_verkon_rakentamissuunnitelmien_toimittamisesta.pdf

2 nRF52840 processor datasheet. Online. URL: https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.1.pdf

3 IMU BNO085 datasheet. Online. URL: https://www.ceva-dsp.com/wp-content/uploads/2019/10/BNO080_085-Datasheet.pdf

4 Bluetooth profiles. Online URL: https://en.wikipedia.org/wiki/List_of_Bluetooth_profiles#Serial_Port_Profile_(SPP)

5 Bluetooth low energy. Online. URL: https://interrupt.memfault.com/blog/ble-throughput-primer

6 GNSS satellites. Online. URL: https://qzss.go.jp/en/technical/satellites/index.html#GPS

7 GNSS. Online URL: https://en.wikipedia.org/wiki/Satellite_navigation#Global_navigation_satellite_systems

8 RTK GNSS picture. Online. URL: https://www.polyu.edu.hk/proj/gef/index.php/glossary/real-time-kinematic-mode/

9 Hexagon RTK service website. Online. URL: https://hxgnsmartnet.com/fi-fi/services/verkko-rtk

10 Geotrim RTK service website. Online. URL: https://geotrim.fi/palvelut/trimnet-vrs/trimnet-vrs-rtk/

11 nRF5340 processor. Online. URL: https://www.nordicsemi.com/Products/Low-power-short-range-wireless/nRF5340

12 Zephyr OS Wikipedia. Online. URL:https://en.wikipedia.org/wiki/Zephyr_(operating_system)#cite_note-6

13 Zephyr OS home site. Online. URL: https://zephyrproject.org/

14 Height system picture. Online. URL: https://www.researchgate.net/profile/Gomaa-Dawod/publication/329686559/figure/fig1/AS:705281319251971@1545163578991/Orthometric-geoid-and-ellipsoid-heights-Ellipsoidal-Orthometric-N-h-H-1.jpg

15 Maanmittauslaitos coordinate conversions. Online. URL: https://www.maanmittauslaitos.fi/kartat-ja-paikkatieto/asiantuntevalle-kayttajalle/koordinaattimuunnokset