

**OULUN
YLIOPISTO**

TEKNILLINEN TIEDEKUNTA

Kustomoitava Human Input Device – Harjoitustyö

Eetu Karvonen 2476742

KONETEKNIIKAN TUTKINTO-OHJELMA

462106A Hienomekaniikka 2019

SISÄLLYSLUETTELO

KANSILEHTI.....	1
SISÄLLYSLUETTELO.....	2
TYÖN ESITTELY.....	3
KOTELON SUUNNITTELU.....	4
3D-TULOSTAMINEN.....	7
LAITTEEN KASAUS.....	9
RISKIANALYYSI.....	12
CE-MERKINTÄ.....	13
OHJELMOINTI.....	14
HARJOITUksen ARVIONTI.....	17

TYÖN ESITTELY

Harjoitystyön tarkoituksena on tehdä laite, jolla voi antaa input-komentoja tietokoneelle langattomasti. Laitteen tulisi olla kohtalaisen helposti kustomoitavissa halutun käyttötarkoituksen mukaan. Harjoitustyö sisältää laitteen suunnittelun sekä toteutuksen.

Toteutustapa

Laitteen toiminnallisuus toteutetaan kahden Arduino Pro Micro -mikroprosessorin (jäljempänä pelkkä arduino) avulla. Toinen arduino ottaa vastaan käyttäjän antamia input-komentoja ja lähettää komennot langattoman radiolähettimen avulla. Toinen arduino puolestaan ottaa omalla radiovastaanottimellaan komennot vastaan ja välittää tiedot input-komennoista tietokoneelle USB-väylää pitkin.

Input-komentojen toiminnallisuudet ohjelmoidaan Arduino IDE-kehitysympäristön avulla.

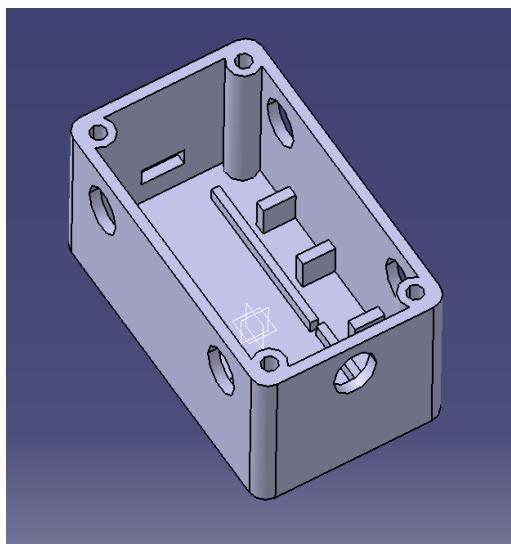
Inputeiksi valittiin painonappeja sekä joystick-tyylinen input.



Kuva 1. Hyödynnettävät komponentit: Arduino Pro Micro ATmega32U4 5V/16MHz, Joystick ohjain arduinolle tatti irroitettuna, 433MHz RF lähetin ja vastaanotin, sekä kaksi erilaista painonappia.

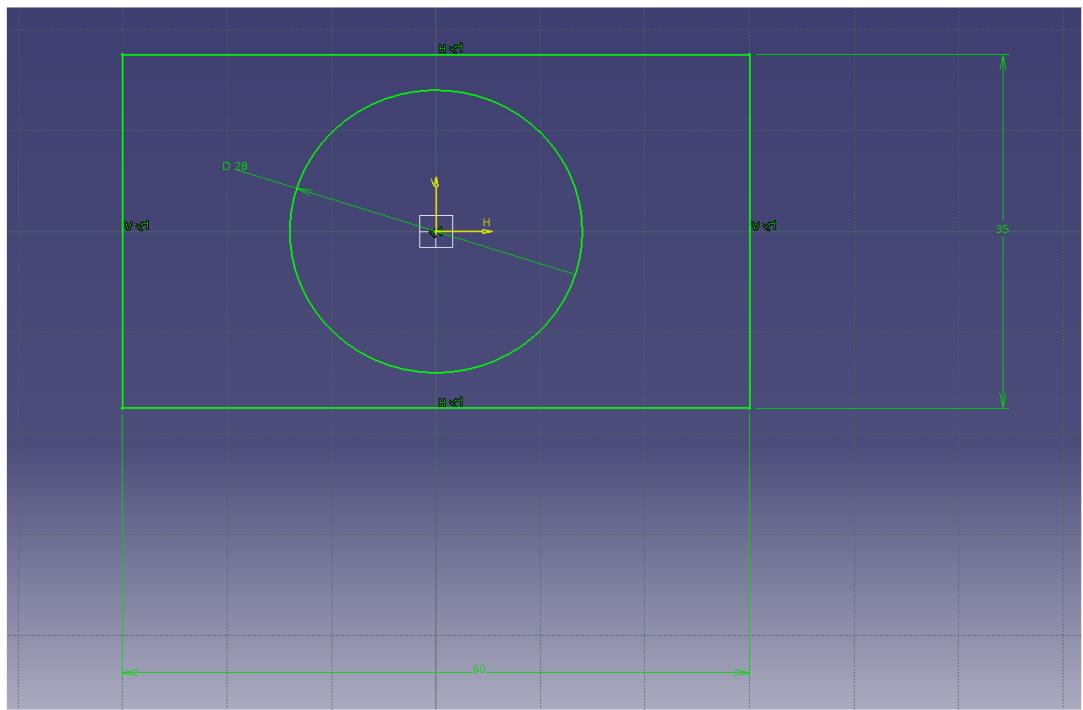
KOTELON SUUNNITTELU

Input-komentoja vastaanottavalle arduinolla ja halutuille inputeille suunniteltiin kotelo käyttäen CATIA v5 CAD-suunnitteluoohjelmistoa. Alku vaiheessa pohdittiin tehdääkö kotelosta neliskulmikas vai pyöreä kellomainen kotelo. Päädyttiin tekemään neliskulmikas kotelo yksinkertaisuuden vuoksi sekä tyhjäntilan minimoimiseksi kotelossa. Kotelo koostuu itse kotelo-osasta, kotelon sulkevasta kannesta sekä joystick-ohjaimen ja arduinon väliin tulevasta välilevystä.

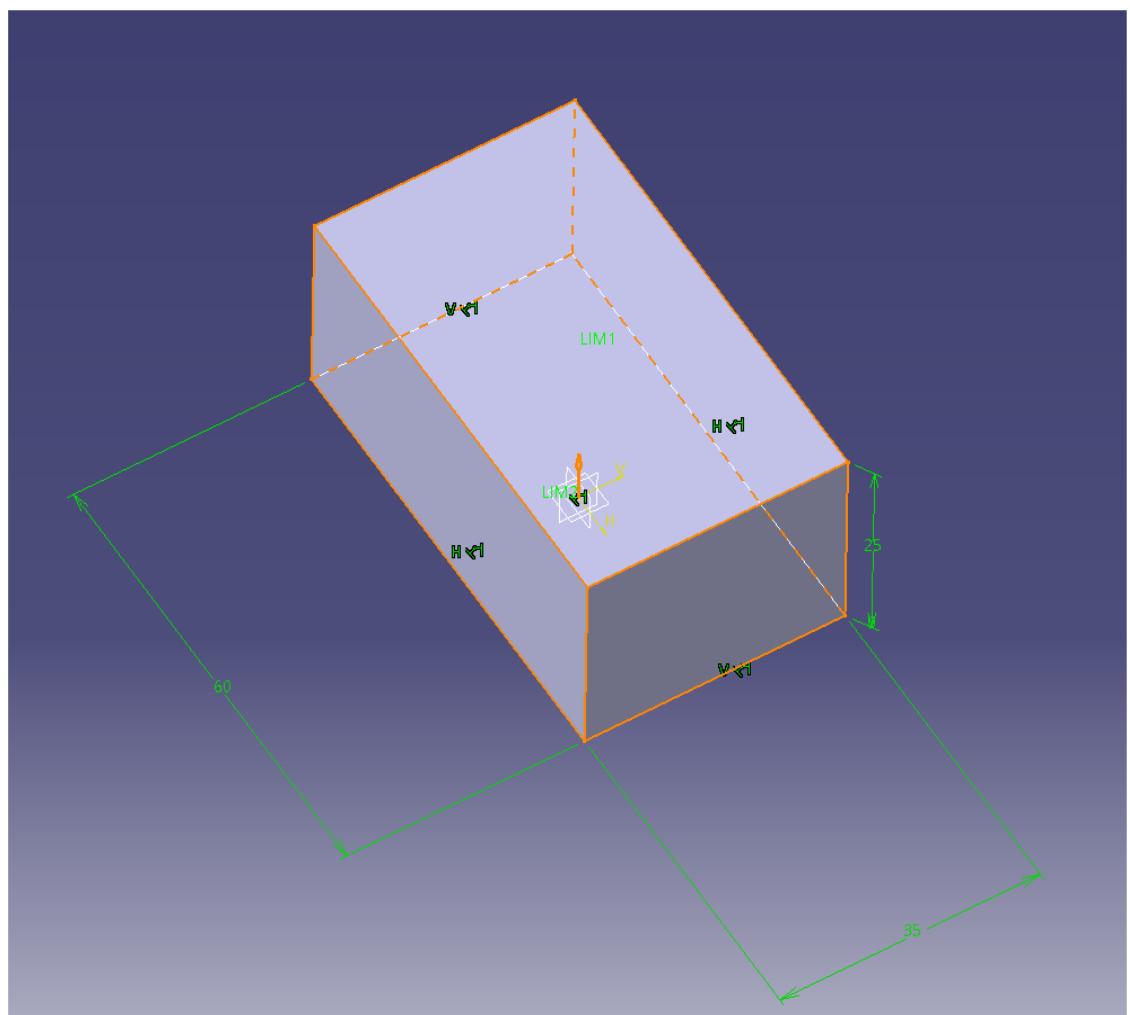


Kuva 2. CAD-kuva kotelosta

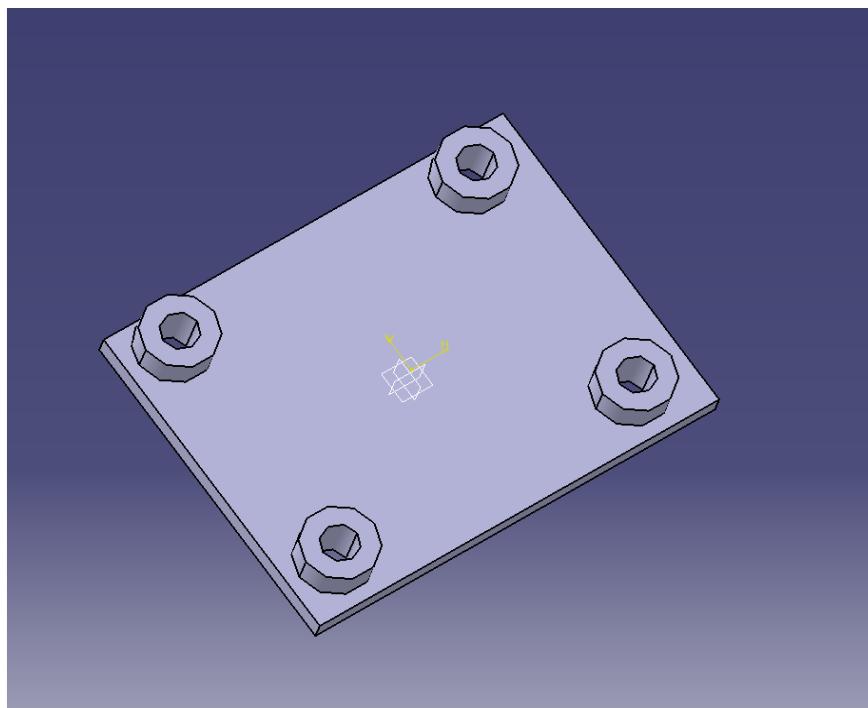
Kotelon seinämissä olevat pyöreät reiät ovat painonappeja varten. Suorakulmion muotoinen reikä puolestaan on arduinon micro-usb porttia varten. Kotelon pohjassa olevat rivat nostavat arduinon ja radiolähettimen irti pohjasta. Reunoilla olevat rivat puolestaan ovat estämässä arduinon ja radiolähettimen liiallista sivuttaisliikettä. Kulmissa olevat reiät ovat kannen sulkemista varten ruuvien avulla. Kotelon päämitat ovat 60x35x25 mm (pituus x syvyys x korkeus).



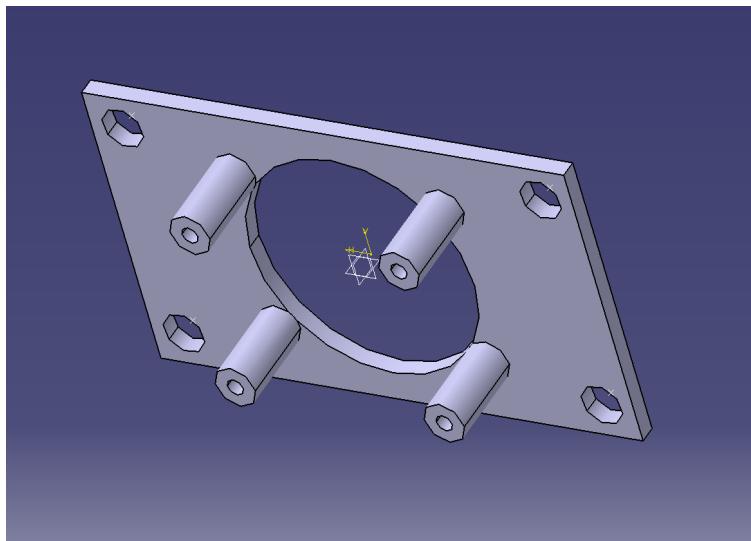
Kuva 3. Kannen CAD-sketch ja päämitat



Kuva 4. Kotelon päämitat



Kuva 5. CAD-kuva välilevystä



Kuva 6. CAD-kuva kotelon kannesta

Kannen tarkoitus on kotelon sulkemisen lisäksi kiinnittää joystick-piirilevy koteloon. Joystick tulee keskellä olevan ison reiän läpi. Joystickin piirilevy tulee kannen tolppien ”pääälle”. Joystickin piirilevyn alle tulee vielä välilevy ja piirilevy ja välilevy kiinnitetään kanteen ruuveilla. Välilevyn tarkoitus on suojata joystickin ja alapuolella olevan arduinon kontakttilta.

3D-TULOSTUS

Kotelon valmistamiseen käytettiin **Stratasys Fortus 380mc** 3D-tulostinta. CAD-mallit tallennettiin STL-tiedostomuodossa, jotka ajettiin Insight ohjelmistolla 3D-tulostimelle. Materiaalina oli musta ABS-M30 muovi ja tukimateriaalina SR30. Materiaalia kului 24.6 cm^3 ja tukimateriaalia 11.4 cm^3 . Kerroksia oli yhteensä 104. Aikaa 3D-tulostamiseen kului 2h 58 min.



Kuva 7. Kotelon osat 3D-tulostettuna

Mittojen vertailu

Osien 3D-tulostuksen jälkeen verrattiin CAD-suunnitelmiin nimellismittoja osien todellisiin mittoihin. 3D-tulostin teki yllättävän tarkkaa työtä, mutta reikien tekemisessä olisi vielä parantamisen varaa. Varsinkin kotelon sivuilla olevat nappien reiät tulivat melko kulmikkaaksi. Tämä johtuu varmaan siitä, että reiät on täytynyt tehdä kerros kerrokselta tukimateriaalia hyväksi käyttäen. Reiän halkaisijan nimellismitan ollessa 8mm reiän todellinen halkaisija oli kulmikkudesta johtuen noin 7,5-8,5mm. Laitteen toiminnan kannalta epätarkkuudella ei ole merkitystä, sillä napit saa kuitenkin kiinnitettyä hyvin.

Kannen ison reiän halkaisijan nimellismitta oli 28mm ja todellisuudessakin halkaisija on hyvin lähelle 28mm. Tämän reiän 3D-tulostin pystyikin tekemään suoraan ympyräliikkeellä, mutta pieni kulmikkuutta on silmin havaittavissa.

Kotelon päämitat ovat hyvin tarkasti oikeat. Kotelon suorat seinämät ovatkin varmasti 3D-tulostimelle helppoja tehdä tarkasti.



Kuva 8. Kotelon pohjan pituus hyvin tarkasti 60mm.

LAITTEEN KASAUS

Kotelon 3D-tulostuksen jälkeen voitiin alkaa kasaamaan laitetta. Huomattiin, että kotelon ensimmäinen versio sisälsi suunnitteluvirheitä. Tämä olikin odotettavissa, sillä kotelon suunnitteleminen ilman fyysistä ensimmäistä versiota oli hyvin vaikeaa.

Kannen reikä joystickille tuli hieman liian pieneksi ja näin ollen rajoittaa hieman joystickin liikerataa. Kotelosta tuli muutenkin kokonaisuudessaan liian pieni. Tyhjänä kotelon osat sopivat oikein hyvin toisiinsa, mutta komponenttien kanssa kansi ei mahdu aivan täydellisesti kiinni joystickin ja välilevyn kanssa, sillä välilevy ottaa kiinni kotelon pohjalla olevaan radiolähettimeen. Lisäksi kotelon sivuilla olevat nappien reikien paikat ovat liian ahtaat eikä suunnitellut napit sovi koteloon. Kotelon päädyssä olevaan napin reikään sen sijaan nappi voidaan sijoittaa.

Komponenttien juottaminen

Komponenttien juotokset suunniteltiin valmiiksi ja juotettiin paikoilleen. Arduino toimi pääasiallisena alustana, johon kaikki muut komponentit yhdistettiin johtimilla.

Arduinon VCC-pinnistä saadaan 5 voltin käyttöjännite, joten siihen pinniin juotettiin sähköjohdon välyksellä joystickin ja radiolähettimen käyttöjännite pinnit. VCC-pinnistä otettiin myös käyttöjännite painonapeille.

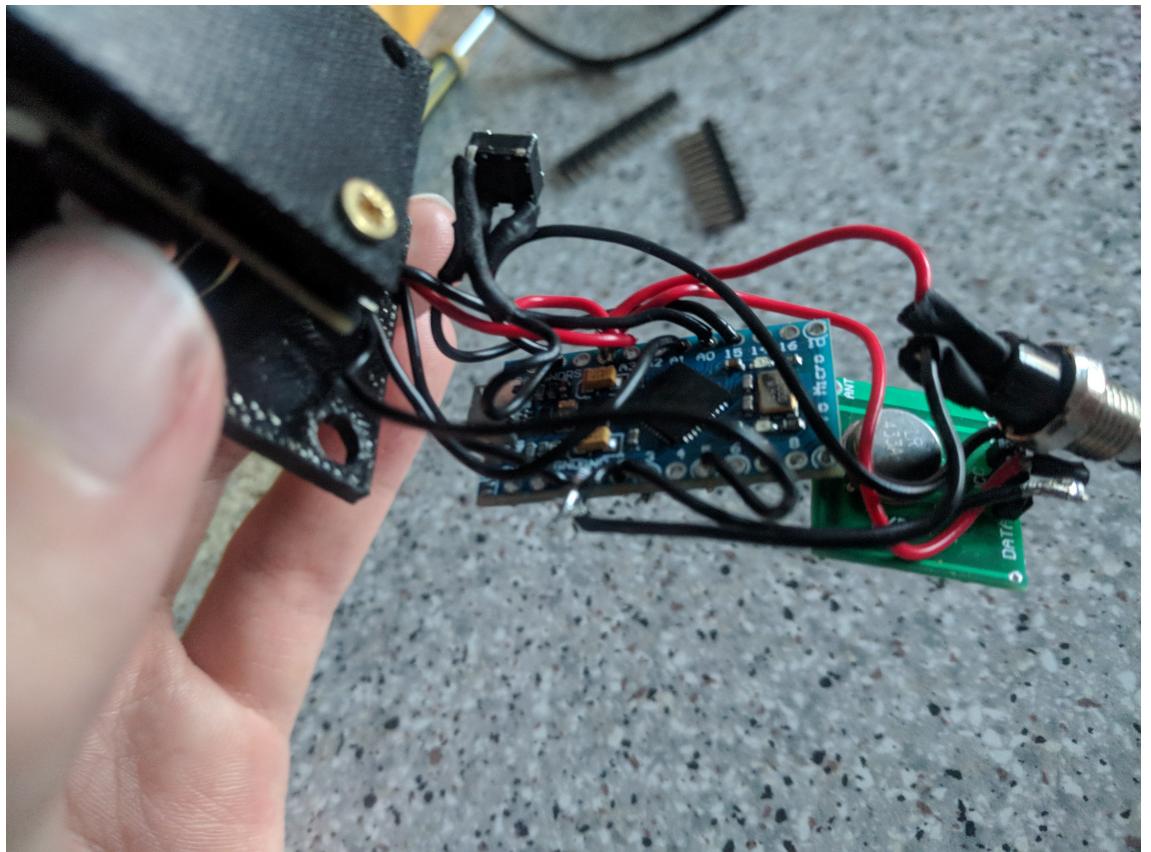
Arduinon GND-pinni eli maadoituspinni yhdistettiin joystickin ja radiolähettimen maadoituspinneihin, sekä painonappien toinen jalka yhdistettiin arduinon GND-pinneihin alasvetovastuksen kautta. Alasvetovastuksena toimi 11 kOhm:n metallikalvovastus, jonka tarkoitus on varmistaa, että painonapin looginen tila on 0 silloin kun nappia ei paineta. Nämä painonappien jalat jotka on yhdistetty maahan, yhdistettiin myös arduinon digitaaliin pinneihin, joista painonappien tilat voidaan ohjelmallisesti lukea.

Myös radiolähettimen data-pinni yhdistettiin arduinon digitaaliseen pinniin, jonka avulla langaton tiedonsiirto onnistuu.

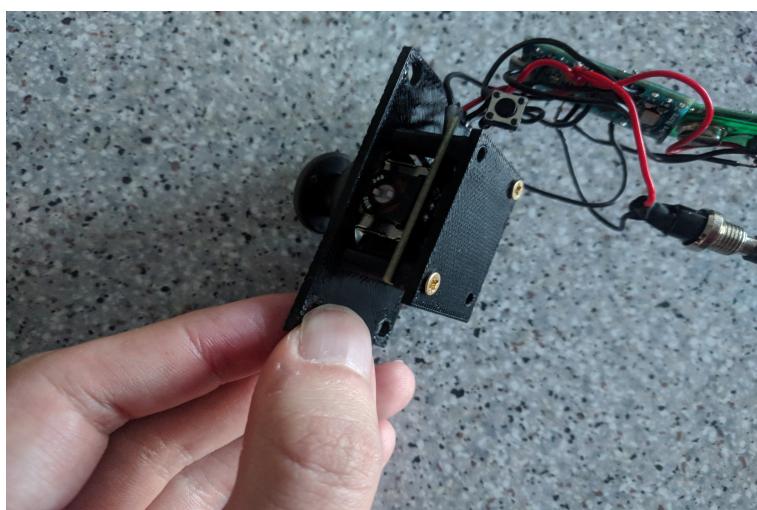
Joystickin x- ja y-pinnit yhdistettiin arduinon analogisiin pinneihin, sillä joystickin asentotiedot tulevat analogisena signaalina.

Osien lopullinen kasaus

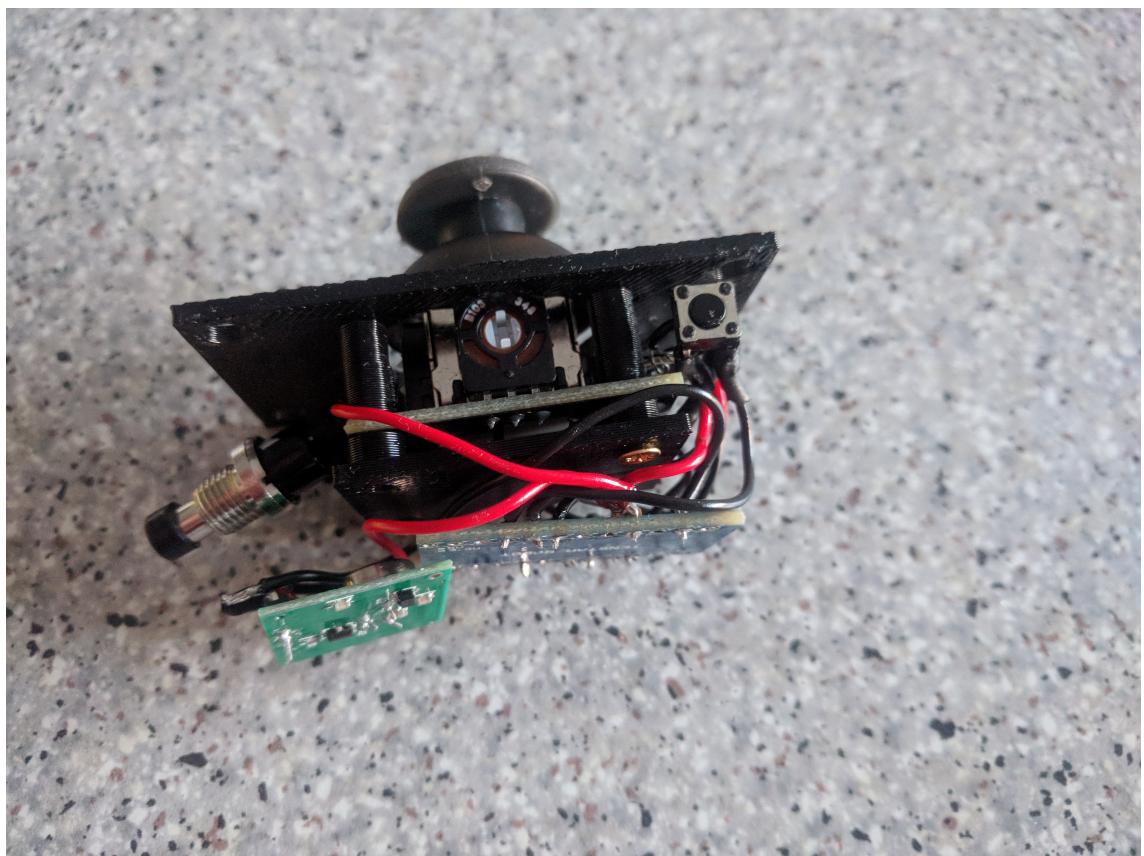
Komponenttien juottamisen jälkeen huomattiin, että käytetyt johdot veivät yllättävän paljon tilaa ja että kotelo on aivan liian pieni komponenttien sijoittamiseen koteloon. Muutakin johtojen määrä laitteessa on sen verran suuri, ettei laitteen toteutusta voi pitää kovin hyvänä.



Kuva 9. Juotokset valmiina



Kuva 10. Joystick ja välilevy yhdistettyynä kanteen kahdella ruuvilla



Kuva 11. Kaikki osat koteloa lukuunottamatta aseteltuna suunnilleen omille paikoilleen



Kuva 12. Laite valmiina niin hyvin kuin laitteen sai kasattua

Riskianalyysi

Laitteen kasausen jälkeen pohdittiin mitä riskejä tuoteeseen sisältyy.

Ensinnäkin kannen kulmien pyöristykset unohtuivat suunnitteluvaiheessa, joten ne ovat näin ollen hieman teräviä. Itse koteloon pyöristykset muistettiin tehdä, joten osat eivät ole siis täysin yhtenevät. Tämä on helppo korjata tuotteen seuraavaan versioon.

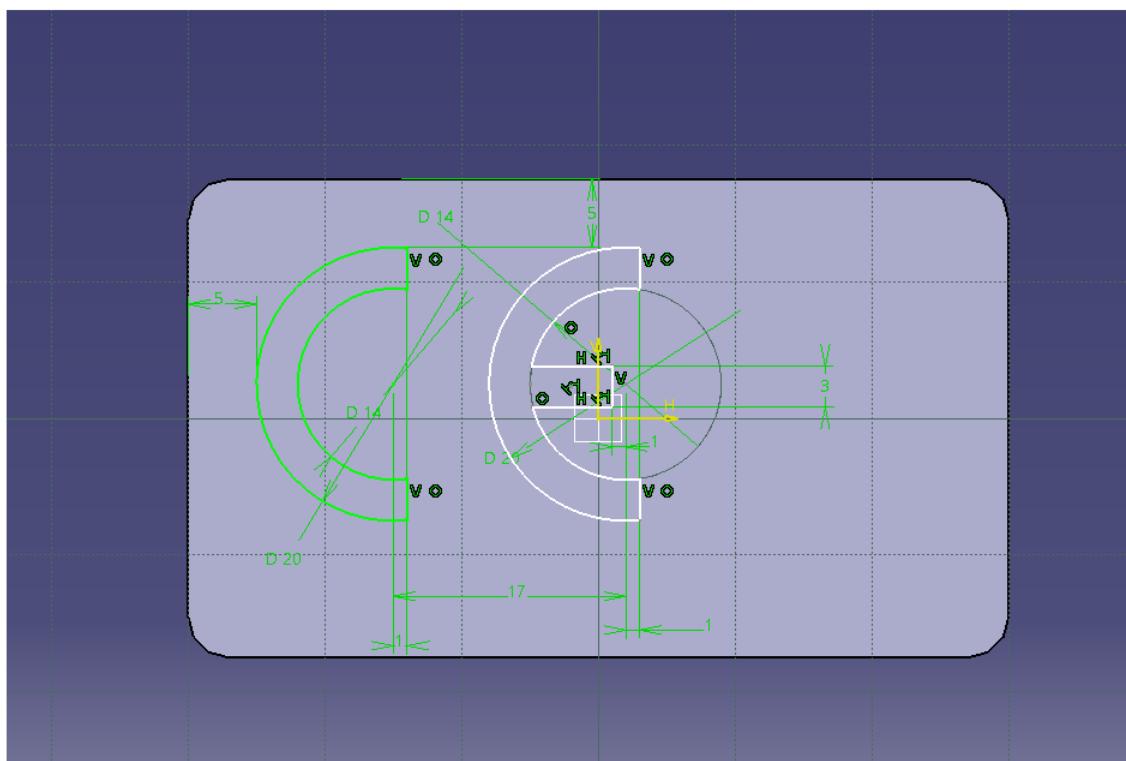
Toinen riski joka tulee mieleen, on jos kotelon päälle sattuu vaikka astumaan ja kotelo rikkoontuu terävästi. Tämä tosin tuntuu melko pieneltä riskiltä. Kotelo tuntuu olevan melko jämäkkää tekoa ja rikkoontuminen vaatisi kohtalaisen kovaa voimaa.

Muuten laitteeseen ei tuntuisi liittyvän juurikaan riskejä.

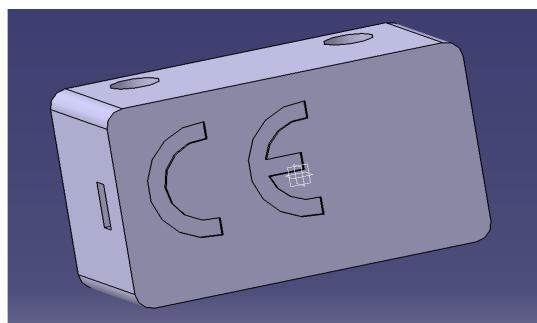
CE-merkintä

Jotta tuoteen voisi tuoda myyntiin Euroopan markkinoille, tuotteesta tulisi löytyä CE-merkintä. CE-merkki takaa, että tuote täyttää Euroopan direktiivien vaatimukset ja on turvallinen käytävä.

CE-merkin voisi sijoittaa vaikka kotelon pohjaan tulevalla tarralla. Vaihtoehtoisesti CE-merkin voisi jo suunnitteluvaiheessa tehdä kotelon pohjaan. Tällöin CE-merkille tehtäisi CE-merkin muotoinen kolo, joka tulisi 3D-tulostettaessa esiin. Ei tosin ole täytä varmuutta, onko tällainen menetelmä sallittu, vai pitääkö merkin olla musta valkoisella pohjalla.



Kuva 13. CE-merkki luonnos kotelon pohjassa. CE-merkin mitoitus on tarkkaa.



Kuva 14. CAD-malli kotelon pohjassa olevasta CE-merkistä

Ohjelointi

Arduinojen ohjelointi tehtiin arduinon omassa kehitysympäristössä. Lähettävälle ja vastaanottavalle arduinoille tehtiin omat ohjelmat. Langattomaan tiedonsiirtoon käytettiin RadioHead kirjastoa (<https://github.com/PaulStoffregen/RadioHead>).

Ohjelmalla testattiin, että laitteen toiminnallisuus toimii, eli napin painallukset rekisteröidään ja tieto saadaan siirrettyä toiselle arduinolle ja usb-väylää pitkin tietokoneelle. Testiohjelma tulostaa napin painalukset ja joystickin asennon sarjamonitorille. Mahdolliset inputtien toiminnalisuudet voi lisätä ohjelmaan esimerkiksi käyttäen keyboard ja mouse kirjastoja.



The image shows two side-by-side screenshots of the Arduino IDE. Both windows have the title 'Transmitter | Arduino 1.8.9' and show the same code for a transmitter sketch. The code includes imports for RadioHead, RH_ASK, and SPI libraries, defines pins for two buttons (nappi, nappi2), and initializes a joystick. It then enters a loop where it reads button states, maps them to values 9 or 8, and sends them via SPI. It also reads joystick values and maps them to 0-255 before sending. The code ends with a note about receiving on port 0-255.

```
#include <RadioHead.h>
#include <RH_ASK.h>
#include <SPI.h>

// Ajurin luonti viestin lähettämiseen
RH_ASK driver(2000, 4, 3);

//Painonappien pinni numerot
int nappi = 15;
int nappi2 = 2;

//Muuttujat painonappien tilaa varten
int val = 0;
int val2 = 0;

//Joystickin x ja y pinnit
int joyx = 0;
int joyy = 1;

//Muuttuja joystickin tilaa varten
uint8_t joystick[2];

void setup() {
    //ajurin alustus
    driver.init();

    Serial.begin(9600);
    pinMode(nappi, INPUT);
    pinMode(nappi2, INPUT);
}

void loop() {

    // Luetaan painonappien tilat
    val = digitalRead(nappi);
    val2 = digitalRead(nappi2);

    // Jos nappia painettiin, lähetetään viesti
    if (val == HIGH) {
        uint8_t nappil[2];
        nappil[0] = 9;
        nappil[1] = 9;
        driver.send(nappil, sizeof(nappil));
        driver.waitPacketSent();
    }

    // Jos nappia2 painettiin, lähetetään viesti
    if (val2 == HIGH) {
        uint8_t nappi2[2];
        nappi2[0] = 9;
        nappi2[1] = 8;
        driver.send(nappi2, sizeof(nappi2));
        driver.waitPacketSent();
    }

    // Luetaan joystickin asento ja mapataan arvot 0-255
    joystick[0] = map(analogRead(joyx), 0, 1023, 0, 255);
    joystick[1] = map(analogRead(joyy), 0, 1023, 0, 255);

    // Lepoasennossa joystick saa arvot x=127, y=125.
    // Jos arvot poikkeavat näistä, lähetetään viesti
    if (joystick[0] != 127 || joystick[1] != 125) {
        driver.send(joystick, sizeof(joystick));
    }
}
```

Kuva 15. Lähettävän arduinon koodi



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** Receiver | Arduino 1.8.9
- Menu Bar:** Tiedosto Muokkaa Sketsi Työkalut Apua
- Toolbar:** Includes icons for Save, Load, Open, Upload, and Download.
- Sketch Name:** Receiver
- Code Area:** Contains the following C++ code for a receiver sketch using RadioHead library:

```
//receiver, vastaanotto

#include <RadioHead.h>
#include <RH_ASK.h>
#include <SPI.h>

RH_ASK driver(2000, 4, 3);

void setup() {
    driver.init();
    Serial.begin(9600);
}

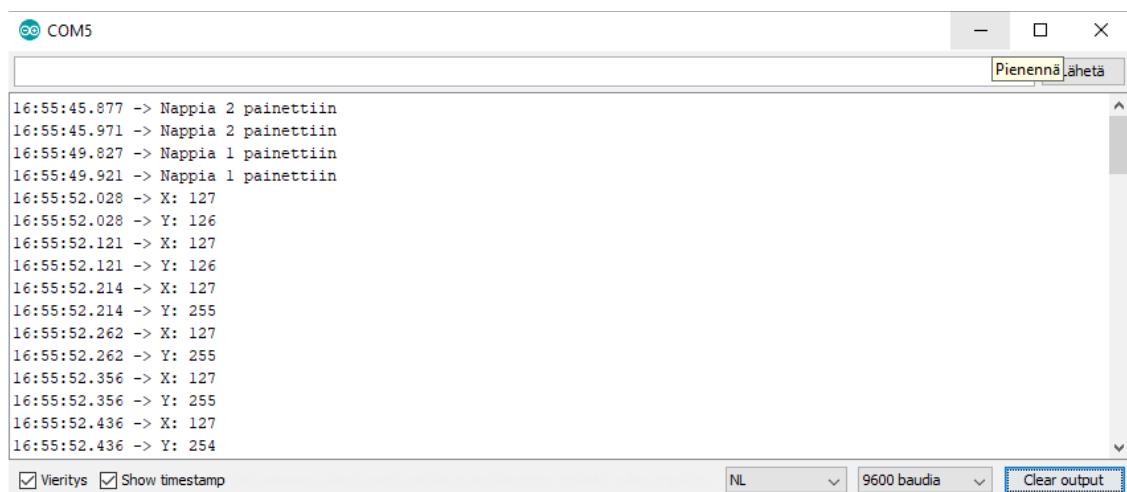
void loop() {
    uint8_t buf[2];
    uint8_t buflen = sizeof(buf);
    if (driver.recv(buf, &buflen)) {      // Jos viesti vastaanotettiin.
        if (buf[0] == 9) {
            if (buf[1] == 9) {
                Serial.println("Nappia 1 painettiin");
            } else {
                Serial.println("Nappia 2 painettiin");
            }
        } else {
            Serial.print("X: ");
            Serial.println(buf[0]);
            Serial.print("Y: ");
            Serial.println(buf[1]);
        }
    }
}
```

Kuva 16. Vastaanottavan arduinon koodi

Ohjelmien yhteistoimintaperiaate on seuraava:

Lähetettävä viesti on taulukko, jossa on kaksi uint8_t muotoista kokonaislukua. Mikäli nappia painetaan, taulukon ensimmäiseksi luvuksi asetetaan 9. Tällöin taulukon toinen luku kertoo kumpi nappi on kyseessä. Toisella napilla taulukon toinen luku on 9 ja toisella napilla taulukon toinen luku on 8. Mikäli lähetettävä viesti on tieto joystickin käytöstä, taulukon ensimmäinen luku kertoo joystickin x-asennon ja taulukon toinen luku kertoo joystickin y-asennon.

Vastaanottavassa päässä viestin sisältö tarkistetaan ja mikäli taulukon ensimmäinen luku on 9 tiedetään, että viesti lähetettiin, koska jompaakumpaa nappia painettiin. Tällöin voidaan katsoa taulukon toinen luku ja näin saadaan tieto kumpaa nappia on painettu ja ohjelma tulostaa tapahtumasta ilmoituksen sarjamonitoriin. Muutoin taulukosta saadaan joystickin asentotieto, joka tulostetaan myös sarjamonitoriin.



```
16:55:45.877 -> Nappia 2 painettiin  
16:55:45.971 -> Nappia 2 painettiin  
16:55:49.827 -> Nappia 1 painettiin  
16:55:49.921 -> Nappia 1 painettiin  
16:55:52.028 -> X: 127  
16:55:52.028 -> Y: 126  
16:55:52.121 -> X: 127  
16:55:52.121 -> Y: 126  
16:55:52.214 -> X: 127  
16:55:52.214 -> Y: 255  
16:55:52.262 -> X: 127  
16:55:52.262 -> Y: 255  
16:55:52.356 -> X: 127  
16:55:52.356 -> Y: 255  
16:55:52.436 -> X: 127  
16:55:52.436 -> Y: 254
```

Kuva 17. Sarjamonitorin tulostuksia laitteen testauksessa

HARJOITUKSEN ARVIOINTI

Harjoitus oli omasta mielestäni erittäin mielenkiintoinen. Idea tällaiselle laitteelle syntyi jo ennen kurssin alkua, kun keksin, että tällaiselle laitteelle olisi itselläni käyttöä.

Laitteen toimintaperiaate toimikin aivan odotetusti, eli inputit saatiin siirrettyä langattomasti tietokoneelle, vaikkakin itse tuotteesta tuli todella karu.

Tästä harjoitustyöstä syntykin oikeastaan vasta prototyppi versio ja aika meinasi käydä siihenkin vähiihin. Laitteen ideaa voisi kehittää ja muokata eteenpäin paremmaksi tuotteeksi.

Ensinnäkin kotelon muotoilu oli tässä versiossa aivan yksinkertainen, jonka tarkoitus oli vain saada komponentit pakettiin, jotta ideaa pystyi testaamaan. Eteenpäin kehittäessä, kotelon voisi tehdä esimerkiksi paremmin käden muotoiseksi tai vaikkapa ranteeseen kiinnitettäväksi. Paino nappien määräänkin pystyy tietysti vaikka lisäämään halutessaan tai lisäämään muun tyypisiä inputteja.

Toinen kehittämisen paikka laitteelle on komponenttien yhteenliittäminen jotenkin paremmalla tavalla. Nyt käytetty johdoilla yhteen juottaminen oli erittäin huono toteutustapa. Parempi tapa voisi olla esimerkiksi tehdä piirilevy, johon komponentit sisäti kiinteämmin liitettyä.

Kotelon suunnittelun olisi pitänyt voida käyttää enemmän aikaa. Parempi olisi ollut, kun olisi voinut ensimmäisen kotelon jälkeen suunnitella vielä toisen, jonka suunnittelu olisi ollut varmasti jo paljon helpompaa. Valitettavasti aika ei millään tähän riittänyt.

Ensimmäinen versio ei ole ollenkaan toimiva laite, sillä laite tuli suunniteltua aivan liian pieneksi, eikä kotelo siis mahtunut edes kiinni. Yksi asia jonka tekisinkin toisin on se, että suunnittelisin ensimmäisen version reilusti isoksi, jotta kaikki osat varmasti mahtuvat kotelon sisään. Nyt ihan turhaan yritin suunnitella juuri sopivan kokoisena ja näin ollen, kun en ottanut huomioon johtojen tarvitsemaa yllättävän suurta tilaa, kotelosta tulikin aivan liian pieni.

Loppujen lopuksi olen ihan tyytyväinen saavutettuun lopputulokseen ja luultavasti yritän vapaa-ajallani kehittää laitteesta vielä toimivamman tuotteen.