

Trådlös mätning av vibrationer på godståg

Examensarbete i Elektrisk mätteknik

Studenter

Anton Magnusson
Christoffer Gydenius

Handledare LTH

Hans W Persson

Handledare Lundinova

Martin Belohorka

Examinator

Johan Nilsson

Institutionen för Biomedicinsk teknik

2014-01-29



LUNDS UNIVERSITET
Lunds Tekniska Högskola

Lund  **inova**

Abstract

The goal is to construct a system that can predict and prevent failures on freight trains.

As the resulting product is a proof of concept, the main focus is to show the capabilities of this concept. To predict and prevent failures, a system needs to be implemented with constant vibration measurement on every buggy along a train with up to 60 wagons. The challenges faced for solving this is the lack of electricity, hard environment, and a long distance to the locomotive for some wagons.

This project shows a solution using mesh network topology and low power circuits for long battery life.

Keywords: mesh network, freight trains, wireless sensors, data communication.

Förord

Examensarbetet utfördes på Lundinova AB där Martin Belohorka var handledare och på institutionen för elektrisk mätteknik på LTH där Hans W Persson var handledare.

Det började med ett förslag att implementera ett meshnätverk till Lundinova. Vid det tillfället hade Lundinova en kund med ett önskemål som hade likheter med förslaget vilket vidare specificerade målen för arbetet. Det slutgiltiga målet skulle innefatta att ta fram en fysiskt produkt som aktivt mäter vibrationerna på varje boggi längs med ett godståg.

Lundinova är ett utvecklingsbolag i Lund med fokus på produktutveckling och konsultverksamhet inom hård- och mjukvara. De har i dagsläget ca 30 st anställda utvecklingsingenjörer.

Det riktas ett tack till ett flertal anställda på Lundinova och framför allt till handledarna för detta examensarbete.

Innehållsföreteckning

1. Introduktion	1
1.1 Bakgrund.....	1
1.2 Mål.....	2
1.3 Översiktlig lösning.....	3
1.4 Begränsningar.....	4
1.5 Delmoment.....	5
1.6 Disposition.....	6
2. Teori	8
2.1 Smith chart.....	8
2.2 Accelerometer.....	9
2.3 Radiokommunikation.....	11
2.3.1 Digitala signaler.....	11
2.3.2 Elektromagnetisk strålning.....	11
3. Hårdvara	14
3.1 NRF51822.....	15
3.2 Accelerometer.....	16
3.3 Mönsterkort.....	16
3.3.1 Tillvägagångssätt för design av mönsterkort.....	19
3.4 Antenn och radio.....	19
3.4.1 Kalibrering av antenn.....	20
3.4.1.1 Rätt antennlängd hittades.....	22
3.4.1.2 Mätning av anpassningsnätverket.....	24
3.5 Låda och övrig hårdvara.....	26
4. Mjukvara	27
4.1 Simulator.....	28
4.2 MG-Controller.....	29
4.2.1 Loggning.....	30
4.2.2 Hämta vibrationsdata.....	30
4.2.3 Övervakning av systemet.....	30
4.3 Programmering av NRF51822.....	30
4.3.1 Timer.....	31
4.3.2 Signal processing.....	33
4.3.3 Drivrutin för accelerometern.....	34

5. Kommunikation	36
5.1 Radiokommunikation.....	36
5.1.1 Reflektion av elektromagnetisk strålning.....	36
5.1.2 Absorption av elektromagnetisk strålning.....	36
5.1.3 Impedansändring hos antennen.....	37
5.2 Protokoll.....	38
5.2.1 Physical layer.....	41
5.2.2 Network layer.....	44
5.2.2.1 Begränsande av sändningfrekvens.....	44
5.2.2.2 Hårdvaruadress.....	44
5.2.3 Discovery layer.....	44
5.2.4 Transport layer.....	45
5.2.5 Application layer.....	46
5.2.5.1 Token.....	48
5.2.5.2 Initieringsprocess.....	49
5.2.5.3 NätverksID.....	49
5.2.5.4 Adress.....	49
5.2.5.5 Vägval.....	49
6. Resultat	50
6.1 Länkpålitlighet.....	51
6.2 Strömförbrukning.....	52
7. Diskussion	53
7.1 Rekommendationer för vidareutveckling.....	53
7.2 Alternativ lösning.....	54
7.3 Resultatanalys.....	55
8. Slutsats	56
9. Referenser	57
10. Ordlista	58
11. Bilagor	59
11.1 Bill of material.....	59

1. Introduktion

1.1 Bakgrund

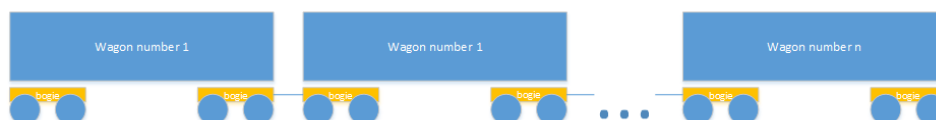
En urspårning för ett godståg kan skada godståget och dess vagnar men en betydligt allvarligare skada blir det på tågrälsen. Detta resulterar i logistikproblem och reparation av tågräls vilket är en dyr historia.

Varför dessa fel kan förekomma beror, enligt uppdragsgivaren, på att vagnarna på godståg inte har någon kontakt med loket och att varken ljud- eller synintryck lyckas ta sig förbi andra vagnar. Att godståg inte har någon elektricitet gör det svårt att applicera ett system som övervakar alla vagnar och kommunicerar med loket när det väl rullar.

Kan urspårningar förutses och upptäckas i förtid så minimeras skadan på rälsen och kostnader sparas relaterat till detta. Ett system som är enkelt att applicera på befintliga godståg som kan detektera fel har potential att betala sig själv genom de besparingar som kunde ha gjorts. Att tolka ett system genom vibrationsmätningar är en teknik som redan finns inom industrin och ses som ett möjligt sätt att upptäcka fel och förutse urspårning. Denna rapport handlar om en realisering av denna metod och därmed måste informationsöverförings- och energiproblemet lösas.

1.2 Mål

Målet med arbetet var att ta fram några prototyper för ett system som mäter vibrationer på varje boggi (se *figur 1*) längs med ett godståg och som kan kommunicera med loket. Prototyperna ska visa möjligheterna med vibrationsmätning på godståg och är ett förslag på hur detta kan lösas.



Figur 1: Varje vagn har 2 boggi där det intressant att mäta vibrationer

Svårigheterna med detta är att godståg inte har någon elektricitet på vagnarna och att varje nod ska vara monterad på undersidan av en boggi där miljön är hård.

Följande specifikationer är arbetet grundat på:

- En vagn är 10 till 15 meter lång.
- Ett tåg kan ha upp till ca 60 vagnar.
- Systemet ska kunna upptäcka en tappad sensor.
- Boggin utsätts för höga och låga temperaturer.
- Två tåg med samma typ av system kan vara nära varandra utan att det ska uppstå konflikter.
- Noderna behöver inte ha fri sikt mellan varandra och kan ha metallföremål som skymmer.
- Systemet ska skilja på vagnar och hålla reda på vagnsnummer.

De krav som ställs på en slutgiltig produkt är att den måste fungera med de specifikationer som står ovan. Prototyperna har inte samma krav och

behöver inte vara många i antal men de måste fungera på ett tåg under testförhållande. Detta innebär kontrollerad temperatur och kort körsträcka.

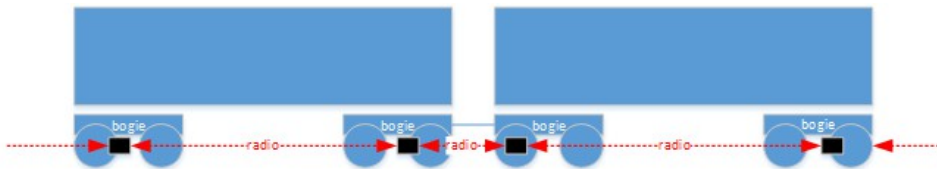
Även om det inte är stor fokus på ett lågt pris på prototyperna som har tagits fram är det en fördel att kunna visa på att konceptet har potential att bli billigt. Samma princip gäller för strömförsörjningen och att kunna bevisa låg strömförbrukning kan ge möjligheter till antingen en lång batteritid med icke utbytbara batterier eller energiupptagning från omgivningen. Radiodelen står för en stor del av energiförbrukningen och därför bör denna vara aktivt så kort tid som möjligt. Fokus har inte lagts på att göra en optimerad prototyp, utan på att visa på möjligheterna som finns inom området.

1.3 Översiktlig lösning

För att kunna förmedla all information till loket kommer informationen med ett egenskrivet protokoll att vidarebefordras mellan noderna (se figur 2). När systemet är aktivt samplar noderna kontinuerligt in vibrationsdata, behandlar denna och vid rätt tillfälle lämnar den till en förbipasserande paket.

Under den aktiva tiden avgör även noderna i vilken ordning de är placerade längs med tåget vilket används för att kunna veta ifall någon har fallit bort, samt för att koppla vibrationsdatan till vagnsnummer.

I loket finns en dator med vilken ett program styr systemet och bestämmer när ett nätverk ska upprättas och systemet ska bli aktivt. En operatör avgör i sin tur när systemet ska vara aktivt via programmet. Datorn loggar all information och på skärmen kan systemets resultat observeras av operatören i loket.



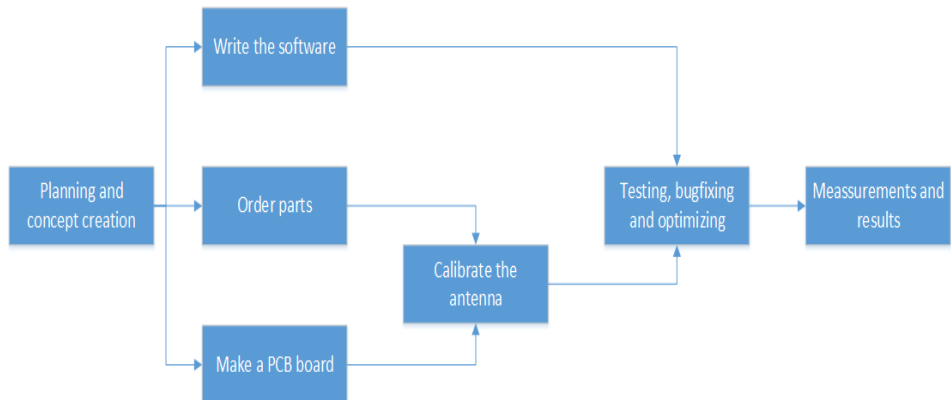
Figur 2: Hur kommunikationen mellan noderna kommer se ut

1.4 Begränsningar

Då arbetet innehöll en del omfattande områden fick något begränsas för att kunna få plats inom ramarna för ett examensarbete. Vilka område som har begränsats och hur berörs i respektive avsnitt. Den största begränsningen gäller optimeringen av antennen som ansågs vara tillräckligt optimerad med de analyser/mätningar som gjorts. Detta med anledningen att det bara är en prototyp och att informationen kring tågets utformning och miljö är begränsad.

1.5 Delmoment

Det följande diagram(*figur 3*) visar, i stort, i vilken ordning de olika delmomenten utfördes och vilka som var beroende av varandra.



Figur 3: Diagram över de olika delmomenten i projektet

Prototypen, som är det fysiska resultatet, kan delas upp i tre huvudområden:

- Hårdvara.
- Mjukvara.
- Kommunikation.

1.6 Disposition

De tre huvudområdena hårdvara, mjukvara och kommunikation återspeglar tre av avsnitten då denna rapport eftersom målet är att ta fram ett fysiskt resultat.

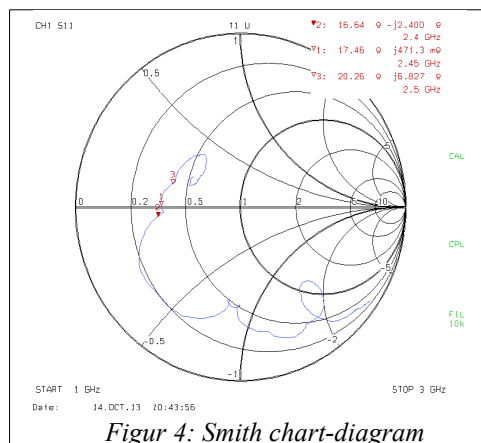
1. **Introduktion** Detta är avsett för att ge läsaren en insikt i vad denna rapport handlar om samt specificerar syfte och mål.
2. **Teori** Genomgång av den grundläggande teori som krävs för att förstå innehållet i rapporten.
3. **Hårdvara.** I detta avsnitt berörs hur prototypen är konstruerad och med vilka komponenter den består av och motiverar valet av dessa. Här resoneras kring strömförsörjning, kretskort, låda, antenn och radio.
4. **Mjukvara.** Den mjukvara som hör till projektet och metoder förklaras som användes vid utvecklingen. Här finns även visuell struktur för hur kod/delprogram har kopplingar.
5. **Kommunikation** Här berörs radioaspekter, hur systemets kommunikation fungerar och protokollets syfte. Då mjukvaran motsvarar strukturen hos protokollet finns en stor del av mjukvaran i detta kapitel.
6. **Resultat.** Det som kan mätas gällande hur bra målen är uppfyllda visas i detta avsnitt. Detta gäller framför allt strömförbrukning och mätningar som ger en uppfattning av länkstabilitet.
7. **Diskussion.** Här diskuteras resultaten och hur de ska tolkas. Avsnittet innehåller även en del rekommendationer för fortsatt arbete samt reflektioner på alternativa lösningar.
8. **Referenser.** Innehåller de referenser varifrån information har använts. Stilen *IEEE* används för hänvisning och upplägg av dessa.

9. *Ordlista.* En lista av ord med förklaringar för de ord som förekommer i rapporten. Om ett ord påträffas som du som läsare inte förstår, bör du kolla i denna lista för betydelse.

2. Teori

2.1 Smith chart

Diagrammet är ett grafiskt verktyg för att visualisera impedansen och reflektionskoefficient som en funktion av frekvensen [1]. Det är användbart vid impedansmatchning och då information om egenskaperna för ett antensystem efterfrågas.



Diagrammet i *figur 4* är en skärmdump från nätverksanalysatorn som nämns senare i denna rapport. Den visar ett *Smith chart*-diagram i vilket man kan observera en linje som är resultatet av mätningar av hur reflektionskoefficientens värde varierar med frekvensen. Det finns *Smith chart*-diagram med mer omfattande rutnät vilka inte berörs i denna teoridel.

För en ökad förståelsen är det av intresse hur reflektionskoefficient och impedans förhåller sig till varannat för en transmissionsledare. Om en transmissionsledare antas ha den karakteristiska impedansen Z_0 och kopplas till en lastimpedans Z_L ges då reflektionskoefficienten Γ av:

$$\Gamma = \frac{Z_L - Z_0}{Z_L + Z_0}$$

Det omvända fungerar också där Z_L kan fås om reflektionskoefficienten och Z_0 är känd:

$$Z_L = Z_0 \frac{1 + \Gamma}{1 - \Gamma}$$

Avläsning av reflektionskoefficienten:

Det komplexa värdet av reflektionskoefficientens avläses med y-värdet som imaginärdel och x-värdet som realdel. För den yttre cirkeln är där absolutbeloppet för reflektionskoefficienten 1 och i centrumunkten i diagrammet är värdet 0. Att reflektionskoefficienten är 0 innebär således en perfekt impedansmatchning.

Rutnät:

Rutnätet för diagrammet i *figur 4* består av en linje, cirklar och delcirklar. Cirklarna visar hur reflektionskoefficientens värde ändras med en ändring på reaktansen för den karakteristiska impedansen. Med en ökad reaktans kommer värdet att följa cirklarnas i riktning med klockan. Halvcirklarna visar hur reflektionskoefficientens värde ändras med en ändring på resistansen. Högre resistans ger en förflyttning i höger riktning i diagrammet längs med halvcirklarna som närmar sig värdet $1 + 0j$ för reflektionskoefficienten. Detta rutnät gör det således möjligt att avläsa impedansen och enkelt att observera hur den karakteristiska impedansen måste kompenseras för att nå det efterfrågade värdet [2].

2.2 Accelerometer

Den resulterande prototypen använder en MEMS-accelerometer. MEMS[3] står för *Micro Electrical Mechanical System* och är små system där små mekaniska konstruktioner kombineras med elektronik. Ofta är

konstruktionerna gjorda av kisel och tillämpningarna kan vara accelerometrar, gyro och små motorer.

Det finns ett antal olika typer av accelerometrar[4] och däribland kapacitiv som mäter kapacitansen mellan en fjädrande massa och fasta stänger, piezoelektrisk som utnyttjar piezoelektriska kristaller vilka ger en spänning under tryck och piezoresistiv som baseras på material som ändrar resistans vid påfrestning. En stor praktisk skillnad mellan dessa nämna typer är att den piezoelektriska inte kan mäta statisk acceleration.

Kapacitansen, spänningen eller resistansen omvandlas till en analog spänning eller mäts av en krets inom samma kapsel. Kretsen kan då kommunicera med externa kretsar och tillhandahålla det uppmätta värdet via digital kommunikation. Exempel på detta är via *I2C*-bus, *SPI*, och *UART*.

2.3 Radiokommunikation

Det finns många olika variationer på radiokommunikation som alla bygger på samma grund: överföra information trådlöst genom modulation av elektromagnetisk strålning. Modulation innebär förändring av strålningen över tid. Det de flesta tänker på när de hör ordet ”radio” är FM-radio, där ljudet från radiopratarerna är det som direkt styr modulationen. Ett annat exempel som skiljer sig från denna princip är kommunikation för mobiltelefoner som kommunicerar med digital information. I detta fallet är det istället bitströmmarna som styr själva modulationen.

2.3.1 Digitala signaler

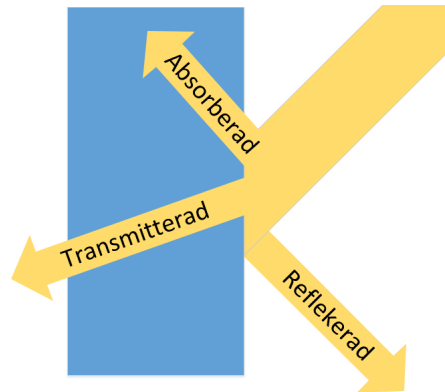
Att skicka digitalt möjliggör fel-detektering[5] och -korrigering av den överförda informationen. Informationen delas upp i flera paket och dessa skickas i en följd till mottagare. Denna princip möjliggör att paketen kan innehålla information som mottagaren kan använda för upptäcka fel vid överföringen. Misstolkning av information kommer då att undvikas och ett nytt paket kan begäras vid behov.

2.3.2 Elektromagnetisk strålning

Elektromagnetisk strålning kan både ses som ett vågfenomen och som partiklar (fotoner). För synsättet vågfenomen kan traditionella lagar för vågor tillämpas med få undantag, där ett av de främsta är förmågan att färdas i vakuum, med avsaknaden av medium. Vågorna består av ett oscillerande elektrisk och magnetisk fält som samverkar och har en vinkelrät orientering mot varandra och vågens riktning.

När en elektromagnetisk våg infaller mot ett föremål (se figur 5) kommer energin att delas upp i tre delar vilka är en reflekterad, en transmitterad

och en absorberad del. Delarnas respektive omfattning är från 0 till 100 % och summan av alla är 100 %[6].



Figur 5: Infallande elektromagnetisk våg mot ett föremål (övergång från ett medium till ett annat).

Reflektion

Efter att den infallande vågen har infallit mot ett föremål ändrar den riktning och rör sig då ifrån föremålet. Utfallsvinkeln bestäms av infallsvinkeln och det finns en begränsad infallsvinkel för att uppnå reflektion definierad mellan olika medium.

Transmission

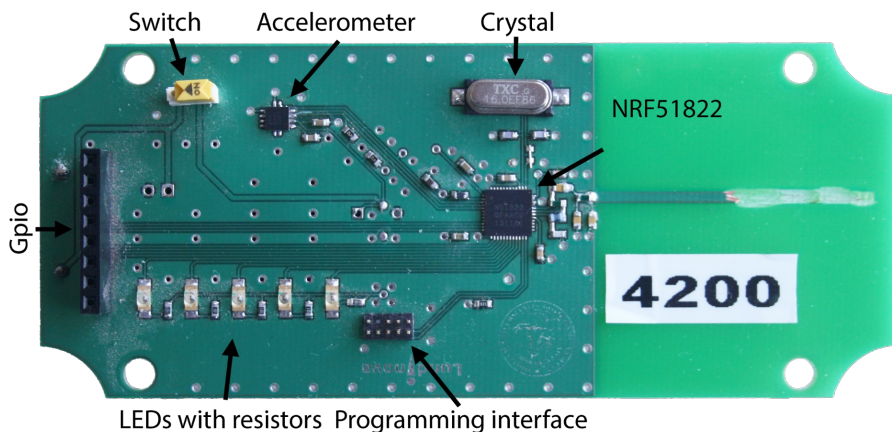
Vågen övergår i det nya mediet och färdriktningen kan påverkas. Även här spelar infallsvinkeln mot föremålet roll för utfallsvinkeln i det nya mediet. Ett föremål som har en hög transmissionsförmåga kallas transparent. Högre frekvenser tenderar att transmittas mindre.

Absorption

Vågens energi övergår i en annan energiform i mediet vilket oftast resulterar i värmeenergi i föremålet som absorberade vågen.

3. Hårdvara

De grundläggande funktionerna i detta system kräver en accelerometer för att tolka vibrationerna, en kommunikationsdel för att förmedla informationen och någon typ av beräkningsdel för att skapa logiken i systemet.



Figur 6: Kretskortet som sitter i modulerna

Till fördel för radioegenskaperna har mönsterkortet (se figur 6) dubbla jordplan som är monterat med nylonskruvar i en låda av plast.

Som nämnt i introduktionsavsnittet är det en fördel att kunna visa på att den slutgiltiga produkten som denna prototyp baseras på har potential att bli billig. Låg strömförbrukning kopplas ofta till ett låg pris och det finns många små, strömsnåla och billiga kretsar på marknaden som kan kommunicera med radio. Valet av komponenter till denna prototyp baserades på att de ansågs ingå i denna kategori och ansågs kunna ersättas av billigare komponenter med liknande funktion i en riktig produkt.

3.1 NRF51822

Denna komponent står för kommunikationen med accelerometern och andra noder och med mjukvara även för logiken i prototypen.

Huvudfunktionerna för detta *SoC*[7] är en 32-bit *ARM Cortex-M0* CPU och en integrerad radiodel för 2,4 GHz. Den försörjs på ett spänningsintervall mellan 1,8 och 3,6V och är designad för låg strömförbrukning vilket är en stor fördel för denna prototyp. Enligt specifikationerna är strömförbrukningen 2,6 μ A med alla funktioner inaktiva, 10,5 mA i sändarläge och 13 mA i mottagarläge. Den klarar av att hantera protokollen *Enhanced ShockBurst*, *Gazelle* och *Bluetooth Low Energy* där det senaste från början var tänkt att användas som en del av protokollstacken beskriven i protokollavsnittet. För att använda Bluetooth Low Energy krävs att enheten programmeras med en *softdevice* vilken vid det tillfället inte hade stöd för att upprätta nätverk med andra noder (agera "Central role"). Istället för att anropa metoder för andra protokoll skrevs kod för att hantera register kopplade till hårdvarufunktioner till radion.

Systemfrekvenser är 16 MHz och kristallen, som syns på *figur 6* ovan, är källan för denna. Även om den har en intern mindre exakt RC-oscillator med samma frekvens kräver *NRF51822* en extern kristall när radion är aktiv. En intern RC-oscillator med frekvensen 32,768 kHz används för tidsevent och det är denna som timerfunktionen är baserad på, vilken är förklarad i avsnitt 4.3.1.

Denna komponent valdes för sin låga strömförbrukning i kombination med utvecklingsmöjligheterna som beräkningskraften och de olika protokollen medförde.

3.2 Accelerometer

Krav här är lättanvänd och strömsnål accelerometer som klarar av upp till 8 g i kraft i alla tre dimensioner, har tillräcklig upplösning och klarar en tillräckligt hög samplingsfrekvens med marginal. En fördel är att den kan försörjas på samma spänningsintervall som övriga komponenter vilket gör att en spänningsregulator inte behövs.

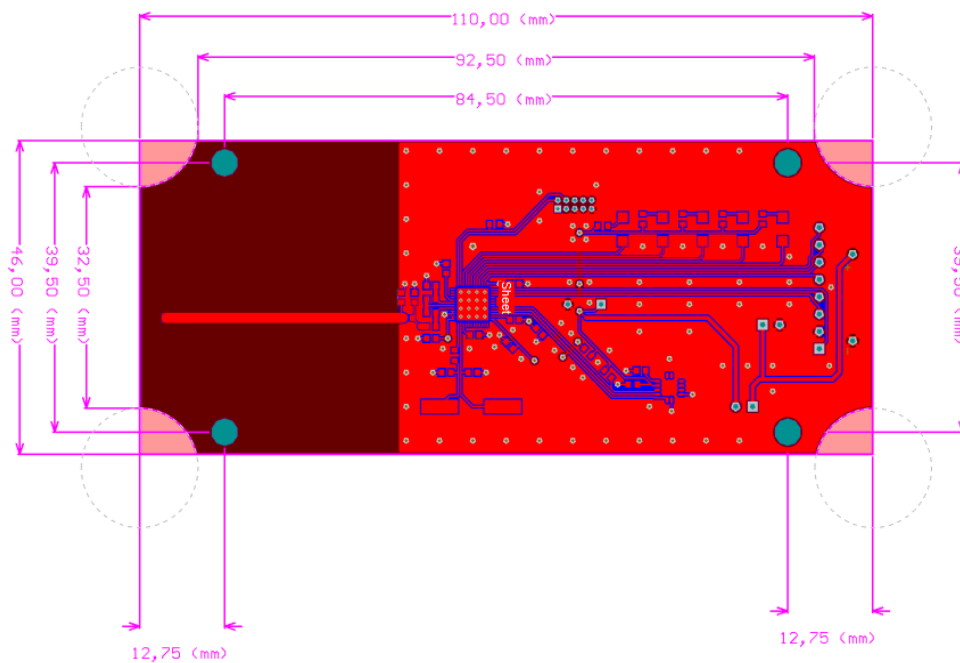
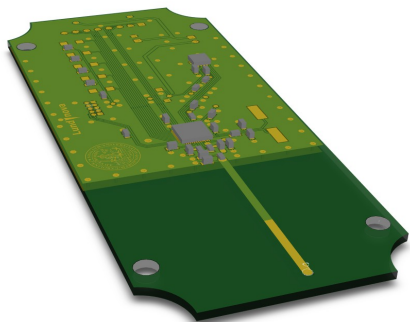
Accelerometern *MMA8491QR1* från *Freescale semiconductor* valdes och denna har i2c-kommunikation för avläsning av kraft i alla 3 axlar. Varje dimension har en upplösning på 14 bitar vilket motsvarar mellan -8 g till 8 g. Enligt databladet är accelerometerns strömförbrukning 400 nA per Hz. Temperaturpåverkan kan ge en känslighetförändring på $\pm 0,0014 \text{ \%}/^{\circ}\text{C}$ och en nivåförändring på $\pm 0,98 \text{ mg}/^{\circ}\text{C}$ [8]. Temperaturpåverkan ansågs vara tillräckligt låg för att det inte ska vara någon påtaglig skillnad för resultaten och därför gjordes ingen ansträngning för att ta hand om detta avvikande i mätresultat.

3.3 Mönsterkort

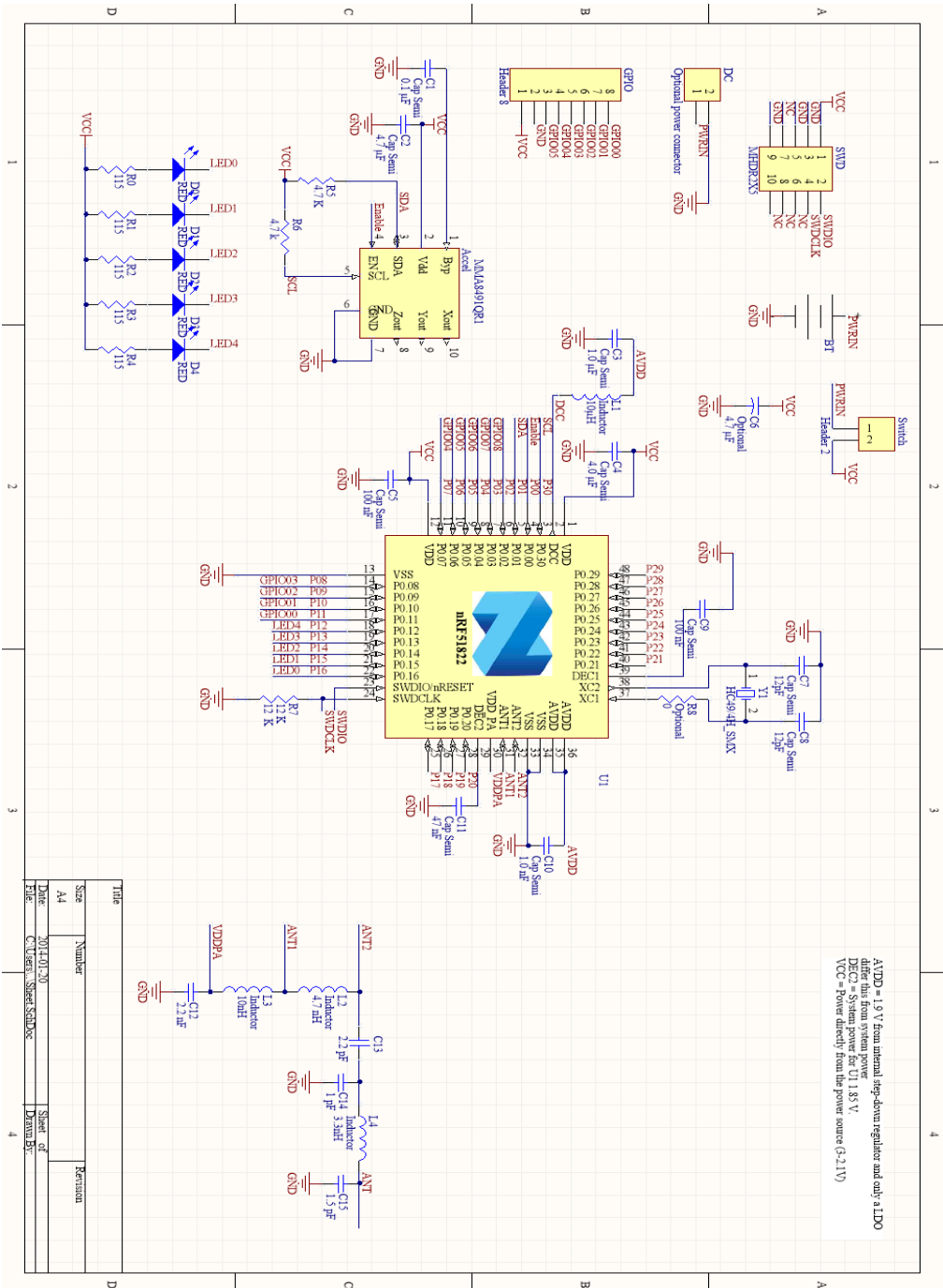
Programmet som användes för att designa mönsterkortet var *Altium Designer*. Mönsterkorten beställdes från *Malmö mönsterkort* är av typen FR4 med specifikationerna två lager, 1,6 mm tjocklek och HASL-ytbeläggning. Kretskortets utformning kan observeras i *figur 7* och *8* såväl som tillhörande kretsschema i *figur 9*.

Även då det är en möjlighet att simulera eller räkna ut optimal antennlängd ansågs det mest tidseffektivt att mäta fram den optimala längden. Antennen på mönsterkortet designades därför med överdriven längd för att sedan skrapa den del som var överflödigt efter mätning.

Figur 7: Genererad 3-dimensionell bild av det beställda mönsterkortet



Figur 8: 2-dimensionell bild med mått för det beställda mönsterkortet



Figur 9: Kretschema för modulen

3.3.1 Tillvägagångssätt för design av mönsterkort

Kraven på utformningen av prototypen var lösa och det som prioriterades var bra möjligheter att fästa den i en boggi samt bra möjligheter för felsökning. Då antennen inte bör ha metallföremål i sin nära omgivning dedikerades ena delen av prototypen helt till denna.

När utrymmet var känt för batterierna och antenndelen valdes en låda alla komponenter fick plats i. Med vald låda designades sedan kretskortet för att passa in med komponenter.

3.4 Antenn och radio

Antennen som valdes är en rak ledare i mönsterkortet riktad 90 grader från jordplanet. Valet av den raka designen av antenn berörs djupare i avsnitt 5.1. Frekvensen radion arbetar med använder ett intervall kring 2,45 GHz och det är enkelt att välja en specifik frekvens i koden med en globalt definierad variabel. De luftburna paketen tolkas med GFSK i hårdvaran och har en maximal teoretiskt hastighet på 1 MBit/s.

Antennutgången på *NRF51822* är av differentiell typ och har utgångsimpedansen $15+85i$ Ohm. Utgångarna är anslutna till ett anpassningsnätverk som består av spolar och kondensatorer där designen har tagits från referensdesignen från tillverkaren av *NRF51822*. Detta anpassningsnätverk har som uppgift att anpassa impedansen från antennen till antennutgången samt den differentiella utgången till monopolantenn. Av de möjliga sätt att kommunicera trådlöst med *NRF51822* valdes 1 MBit/s där frekvensen kan väljas mellan 2,4 till 2,483 GHz och frekvensavvikelsen max får vara som maximalt ± 170 kHz.

3.4.1 Kalibrering av antenn

Om inte antennen kalibreras ger det en lägre verkningsgrad och lägre sändningsstyrka samt ökar risken för interferensproblem[1]. All mätning gällande detta avsnitt gjordes i en laborationssal på LTH med en nätverksanalysator vilken användes för att mäta reflektionskoefficienten hos antennen och anpassningsnätverket.

Kalibrering av antennen är uppdelad i två följande delar och ska göras i den ordning de står:

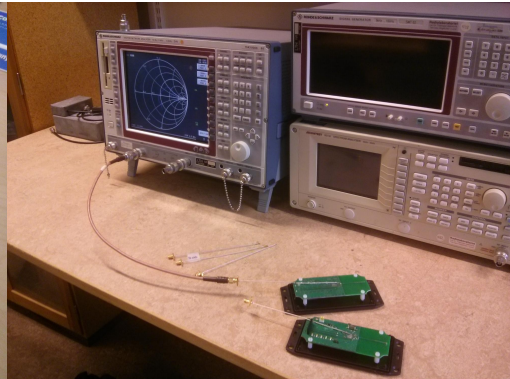
1. Hitta rätt antennlängd som ger resonansfrekvensen i mitten av arbetspektrumet. 2,45 GHz är önskvärt för denna prototyp.
2. Anpassa impedansen till antennutgångarna för ett stående vågförhållande (SWR) som ska vara så nära 1 som möjligt då ett högre värde ger med reflekterad effekt. En så låg reflekterad effekt som möjligt är önskvärt då detta ger högre verkningsgrad och högre utsänd effekt.

Det faktum att antennens impedans påverkas av objekt i nära omgivning gör det viktigt att uppställningen vid mättillfället motsvarar omgivningen då produkten används. Av den anledningen förberedes två mönsterkort(se *figur 10*) utan komponenter men med låda för att motsvara den slutgiltiga produkten, vilka kan observeras i bilden. Det ena mönsterkortet var avsett till att hitta rätt antennlängd (del 1) och det andra för att hitta rätt impedans för anpassningsnätverket(del 2).

På varje mönsterkort löddades en SMA-kontakt som passade nätverksanalysator. För kortet som användes för att hitta rätt antennlängd löddades SMA-kontakten direkt till antennen (nod "ANT" på *figur 15*) och



Figur 10: De två modulerna anpassade för mätning av radiodel



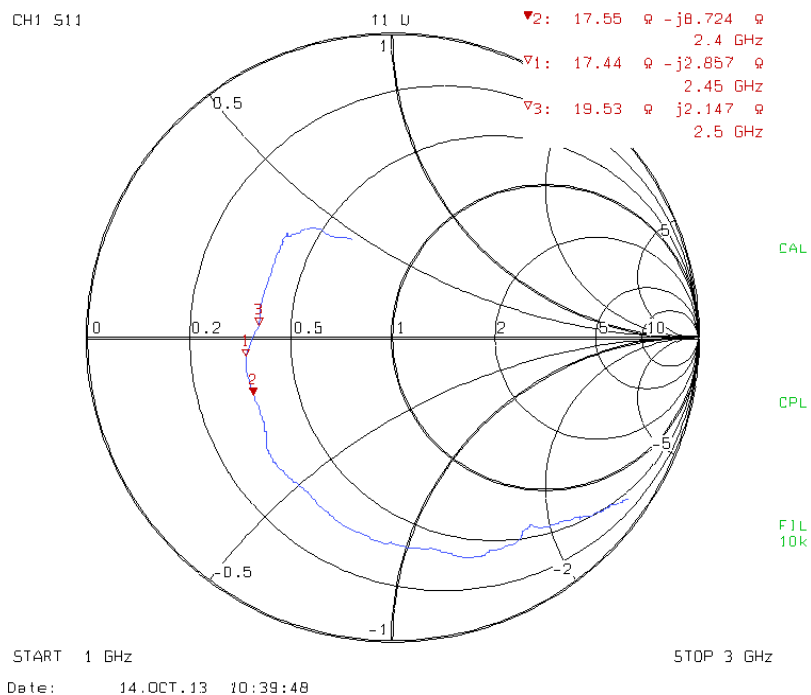
Figur 11: Nätverksanalysatorn i laborationssalen där mätningarna gjordes

på det andra kortet löddades SMA-kontakten till noden för antennutgången på *NRF51822* (nod “ANT2” på *figur 15*). På *figur 11* kan nätverksanalysatorn observeras tillsammans med korten som den kopplades till vid mätning.

3.4.1.1 Rätt antennlängd hittades

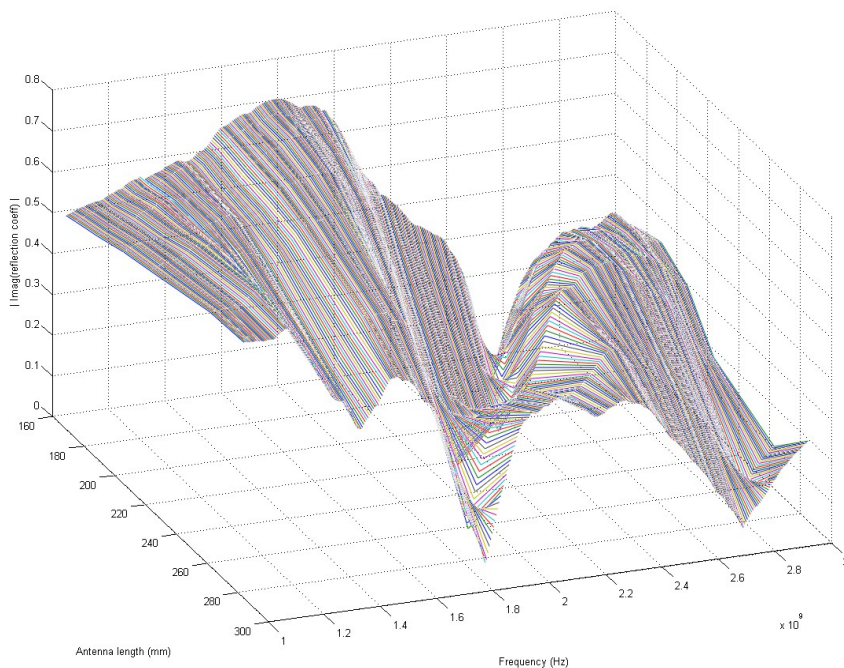
Tillvägagångssättet för att hitta en rätt antennlängd var att göra en mätning, spara mätdata, skrapa av en bit av antennen och göra om allt igen till rätt längd har hittats. Detta gjordes för antennlängder från 284 mm till 169 mm.

Nätverksanalysatorn, med den karakteristiska impedansen 50 Ohm, ställdes in på att svepa mellan 1 till 3 GHz för att även se eventuella fenomen utanför arbetspektrumet. Vidare visualiserade den mätningarna med ett *Smith chart*-diagram där den uppmätta kurvan visar hur reflektionskoefficienten ändras med frekvensen. Vid resonansfrekvensen är detta förhållande helt reellt vilket gör det önskvärt att där kurvan skär den reella axeln är frekvensen 2,45 GHz.



Figur 12: Resultande mätning vid en antennlängd på 175 mm

Smith chart-diagrammet i *figur 12* är en skärmdump från nätverksanalysatorns skärm vid mättillfället och visar mätningen för vad som ansågs vara rätt antennlängd, 175 mm. Reflektionskoefficienten är i princip helt reell vid resonansfrekvensen då antennlängden är 175 mm, vilket kan observeras i grafen. Av de tre pilarna i diagrammet markerar de två yttre frekvensintervallet 2,4 - 2,5 GHz och den mittersta 2,45 GHz.



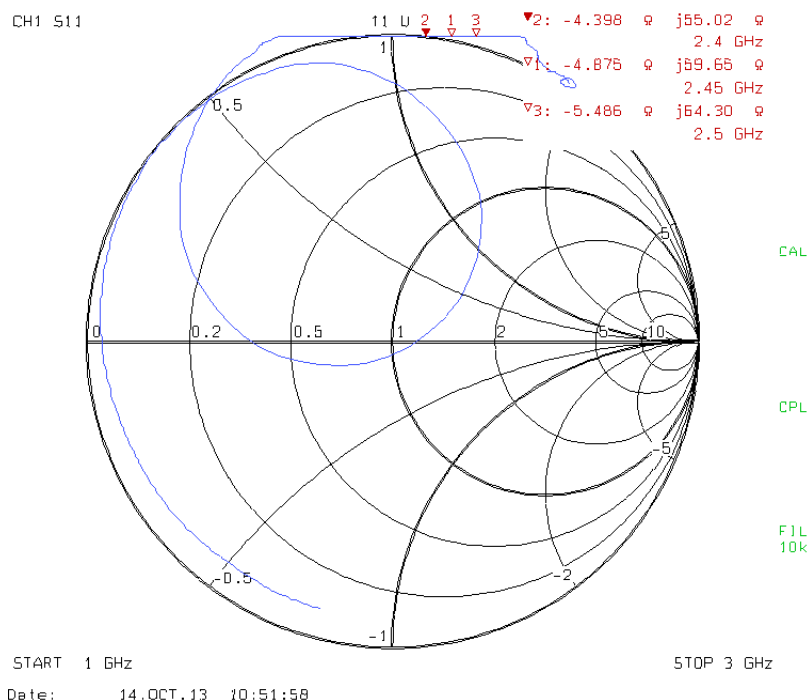
Figur 13 - Mätningar sammanställda med olika antennlängder

På *figur 13* är alla mätningarna sammanställda där resonansfrekvensen kan utläsas samt hur den ändras med längden på antennen.

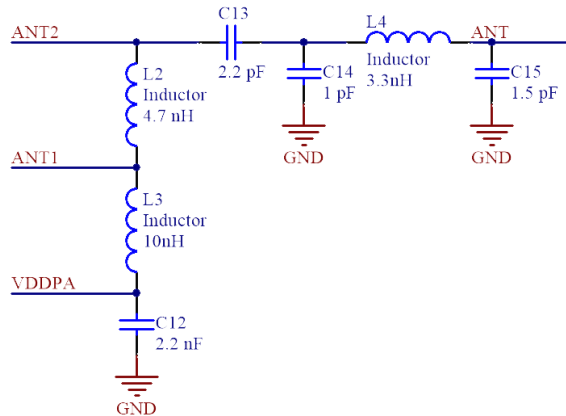
3.4.1.2 Mätning av anpassningsnätverket

Komponenterna i anpassningsnätverket valdes med omsorg för att ha bra frekvensegenskaper och liten avvikelse i komponentvärde i frekvenser upp till 2,4 GHz. Med den funna antennlängden på 175 mm gjordes mätningar med nätverksanalysator på en av utgångarna. Vad som eftersträvas är ett stående våg-förhållande på 1, vilket betyder att utgångsimpedansen ska vara samma som lasten av antennen med anpassningsnätverket.

Punkten på kurvan för reflektionskoefficienten vid 2,45 GHz är utanför området men impedansen vid denna frekvens uppmättes till $-4,875 + 59,85i$ Ohm. Denna kurva kan observeras på diagrammet i figur 14. Impedansen är lasten sett från antennutången (ANT2) med alla komponenter enligt figur 15.



Figur 14: Smith chart-diagram för mätning av impedansen för



Figur 15: Antennnätverkets kretsschema vid mätning och innan korrigering

Den önskade impedansen, $Z_d = 15 + 85i$, skiljer sig således från det uppmätta. Att kompensera genom att ändra komponentvärdet på några komponenter kan föra impedansen närmre det önskvärda. Hänsyn bör dock tas till att komponenter avviker från sitt teoretiska värde och signallängden kan ändras och därför bör impedansen mätas efter komponentbyte. Placeringen av de tre impedanserna C14, L4, C15, utgör ett pi-nätverk, vilket möjliggör anpassning till vilken önskvärd impedans som helst genom ändring av komponentvärde. Att ta reda på och beställa exakta komponentvärde och göra en ny mätning ansågs överflödigt då det är en prototyp och då den ansågs fungera tillräckligt efter att C13 ersatts med en kortslutning.

Då C13 är direkt ansluten till noden ANT2 och i serie med resten av antennnätverket kommer en ersättning av C13 med en kortslutning att minska antennnätverkets impedans med värdet av impedansen av $Z_{C13} = 1/i\omega(C13)$. Således blir då antennnätverkets impedans efter C13 ersatts med en kortslutning:

$$(-4,875 + 59,85i) - (Z_{C13}) = (-4,875 + 59,85i) - (-i \div 2 * \pi * w) \approx -4,88 + 89,8$$

3.5 Låda och övrig hårdvara

Lådan valdes för sin passande storlek, utformning och möjlighet till infästning på boggin. Då det inte är bestämt för hur produkten ska vara monterad under boggin är det bra att öppna för möjligheter.

Lysdioderna och GPIO-anslutningarna är endast för utvecklingssyfte och används inte när noderna är monterade på tågagnarna med locket på.

På baksidan av kortet är en batterihållare monterad som håller två st AA-batterier. Mönsterkortet kan monteras på 2 håll varav det ena är en fördel vid felsökning och det andra en fördel vid batteribyte. Detta har att göra med vilken sida av kortet som är åtkomligt när locket på lådan är av.

4. Mjukvara

Mjukvaran innefattar ett program på datorn skrivet i *java* och ett program för noderna skrivit i *C*. Javaprogrammet har fått namnet “MG-Controller” och dess syfte är att logga och visa mätdata samt övervaka systemet. Där det är begränsade beräkningsresurser, och det befintliga avsedda språket för *NRF51822* är *C*, är programmet i noderna helt skrivet i det programspråket. *Eclipse* användes som utvecklargränssnitt för båda programmen för en enhetlig användning. Revision- och filhanteringen gjordes med *git* för att kunna samarbeta med delprogram. För *NRF51822* användes *Sourcery CodeBench Lite Edition for ARM EABI* för kompilering av kod och programmerades med en programmerare från *developer kit:et* till detta *SoC*, vilket beställdes i början av detta examensarbete.

Huvudnoden som kommunicerar med datorn (grön på bilden nedan) är ansluten och försörjd via USB-porten på datorn och har fler uppgifter att rapportera information till datorn men har samtidigt inga krav på att vara strömsnål. Den har ett annat program som har samma kod men med en definierad variabel som gör att kompilatorn väljer bort kod som inte är anpassad för den typ av nod som koden ska kompileras till. Detta görs genom “*#ifdef*”-funktionen hos *preprocessorn*.

Det finns 2 olika fysiska typer av noder(se *figur 16*) med olika kod:

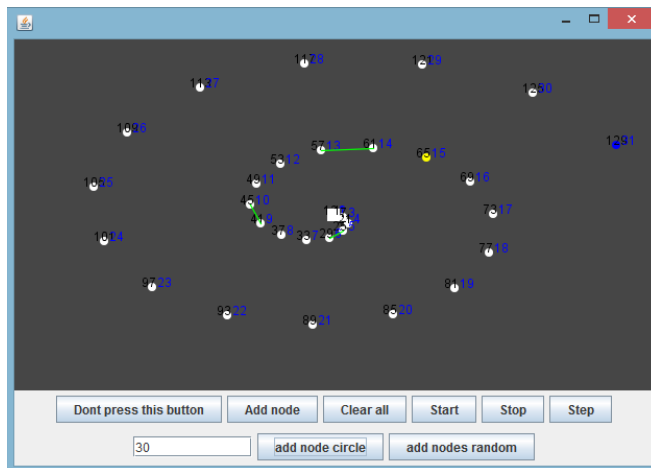
- Huvudnod som är direkt kopplad och försörjd av datorns usb-port.
- Slavnoderna som är de som sitter på vagnarna.



Figur 16: Typ av mjukvara på de olika delarna i systemet

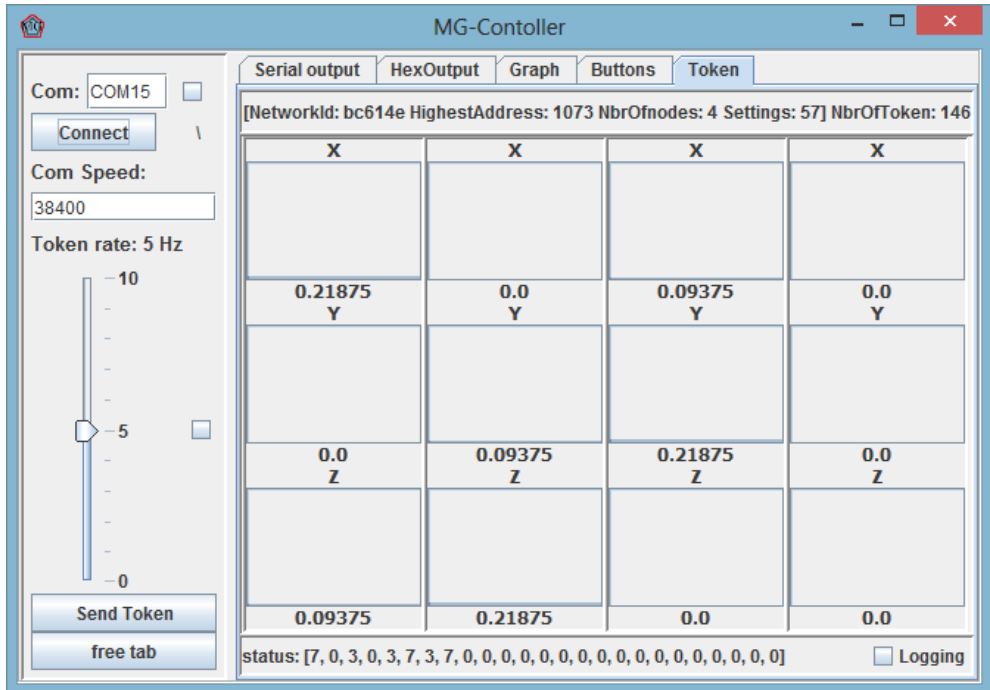
4.1 Simulator

Syftet med denna (se figur 17) var att i ett tidigt stadié kunna visualisera och testa olika teorier för hur det slutgiltiga resultatet kommer att bli och reducerar en del av de idéer som inte fungerar. Den är enkel i sin grund och visar noderna i en 2-dimensionell placering där paketen som skickas via noderna kan observeras.



Figur 17: Bild på simulatören

4.2 MG-Controller



Figur 18: Bild på gränssnittet för programmet "MG-Controller"

Detta program (*se figur 18*) är helt skrivet i java och körs på datorn i loket som står för systemets interaktion med människan. Den kommunicerar med huvudnoden via COM-port och använder sig av ett tillägg för denna funktion. Att den använder java innebär att det fungerar på *Linux*, *Mac* och *Windows* då *java virtual machine* är installerad och rätt dll-fil vald för COM-port för respektive plattform. Utöver felsökningsfunktionerna som har använts under projektets gång har programmet funktioner för loggning av data från noderna, hämtning av data och övervakning av systemet. För den framtida produkten som denna prototyp representerar är det inte bestämt hur interaktionen med människan ska gå till. Detta program är främst avsett för att samla in den data som kommer från noderna och visa människorna som testar denna prototyp.

4.2.1 Loggning

Varje paket som innehåller vibrationsinformation sparas som en .csv-fil i samma mapp som programmet befinner sig. I filnamnet finns en tids- och datumstämpel som har ett extra syfte att undvika konflikter för filer med samma namn. Dessa filer kan bland annat öppnas enkelt i *Matlab* eller *Excel* för en tolkning av resultatet vid ett senare tillfället.

4.2.2 Hämta vibrationsdata

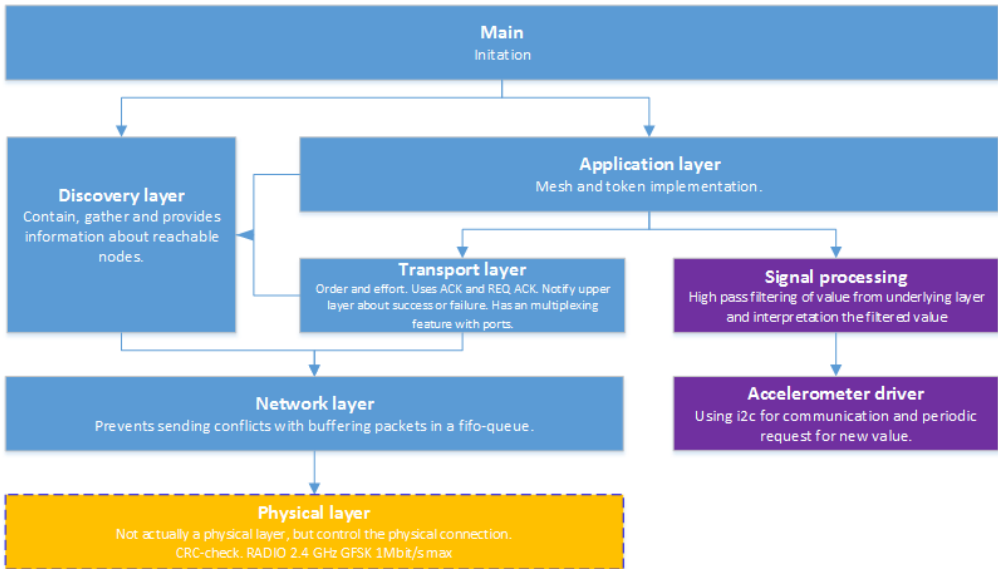
Detta kan göras antingen manuellt på knappen “Send token” eller genom infyllnad av *checkbox* för periodisk utsändning. *Slidern* bredvid denna *checkbox*en bestämmer med vilken frekvens token ska sändas ut och systemet uppdatera sin information.

4.2.3 Övervakning av systemet

Token innehåller vibrationsinformationen och under fliken “token” kan denna observeras. Det är även under denna flik loggningen av data kan aktiveras eller avaktiveras genom en *checkbox* nere i högre hörnet av fönstret.

4.3 Programmering av NRF51822

För att lättare testa, få struktur på koden och lättare kunna identifiera buggar delades den största delen programmet upp i lager med likheter till en trädstruktur (*visas i figur 19*). Varje lager initierar underliggande och hela programmet byggdes upp underifrån lager för lager. Pilarna ska visa på den hierarkiska ordningen där anrop görs i pilens riktning, med få undantag. Om ett undre lager vill anropa ovanstående använder den sig av en funktionspekare som överliggande lager skickat in vid initiering.



Figur 19: Uppdelning av mjukvara för NRF51822

De delar som inte delades upp i lager och som inte är med i diagrammet men som skrevs för detta projekt hade ett flertal andra delar av programmet tillgång till. All kod är kommenterad och finns att tillgå men endast de mest relevanta filerna för projektet valdes att tas med här i rapporten.

För att undvika konflikter med andra filer lades bokstavskombinationen “mg” till för på en del filer vilket står för “Magnusson Gydenius”.

4.3.1 Timer

Denna består av “mg_timer.c” och “mg_timer_h” i mappen “util”. Denna del tillhandahåller funktioner för att schemalägga händelser form av funktionspekare som pekar på metoder med argument.

Den kräver att en metod körs i main-loopen där den väntar på avbrott . Då avbrotten är aktiverade för RTC-klockan kommer denna metod att köras

med minst den frekvens dessa avbrott körs. Det avbrottsrutinen gör är att räkna upp en tid med vilken denna de schemalagda händelserna baserar sin tid på. När ett avbrott inväntar avbrott aktiverar *CPU*n sitt strömsparläge.

De schemalagda händelserna ligger i en sorterad länkad lista vilket och varje gång ett avbrott inträffar behöver kolla ifall det är tid för första händelsen att köras, trots att det kan vara flera som väntar på att köras senare.

Då alla event körs i main-metoden med samma *stack* kommer varje att köras klart innan nästa börjar. Fördelarna med detta är, förutom enkelheten relativt till ett trådbaserat system, att delade resurser kommer att användas endast av ett event samtidigt och det är inte behov av semaforer och ömsesidig uteslutning. Nackdelarna är att alla delar av programmet måste ta hänsyn till att uppgiften den schemalägger inte tar för lång tid att fördröja någon tidskritiskt.

4.3.2 Signal processing

Består av "mg_sampler.c" och "mg_sampler.h" i mappen "sensor" och har som huvuduppgift att utföra enklare behandling av de utlästa värdena från accelerometern med huvudsyfte att minska mängden data som skickas. Värdena från accelerometerns drivrutin filtreras genom ett högpasfilter för att sedan spara undan den högst påträffade amplituden från den filtrerade signalen. Varje rumsdimension har sin egen filtrering och sitt eget sparade maxvärde för amplituden.

Högpasfiltret har en brytningsfrekvens på 0,5 Hz och syftet är att filtrera bort jordens gravitation från accelerationssignalen som innehåller den intressanta vibrationsinformationen.

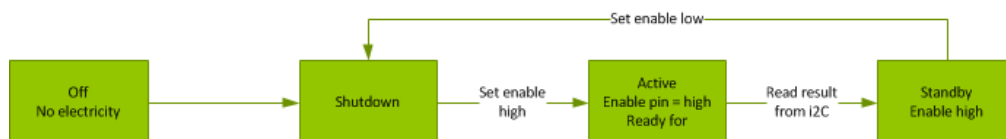
Ett annat delprogram kan anropa "mg_sampler" för att få ut maxvärdet för respektive dimension, varpå värdet då nollställs. Det hela upprepar sig då för att generera ett nytt maxvärde till nästa tillfälle då maxvärdet utläses.

4.3.3 Drivrutin för accelerometern

Denna drivrutin består av "mg_accel.h" och "mg_accel_mma.c" och uppgiften är att tillhandahålla funktioner som startar och stoppar periodiska mätningar av accelerationen via *i2c* och vidarebefordrar de till en vald metod.

Accelerometern har olika lägen den går igenom för varje gång en mätning görs vilket drivrutinen håller reda på. I vilken ordning de olika lägena aktiveras kan observeras på *figur 20*. Varje avläsning startas av ett periodiskt schemalagt event med frekvensen 64 Hz. Detta innebär att man kan enligt Nyquist-teoremet återskapa en accelerationsignal på halva samplingsfrekvensen, 32 Hz. Om signaler mäts med frekvenser över denna kommer således vikiningsfenomen att uppstå.

Samplingsfrekvensen valdes godtyckligt då det inte fanns något krav om hur hög den behövdes vara. 64 Hz ansågs som tillräckligt och koden anpassades för att klara av en ändring för eventuella framtida behov. Anpassningen avser främst filterkoefficienter i "mg_sampler.c" och "mg_sampler.h" som beräknas vid kompilering och är beroende av samplingsfrekvensen och brytfrekvensen.



Figur 20: Blockschema över accelerometers olika läge

Först när accelerometern får ström samtidigt som *ENABLE*-pinnen på accelerometern är satt till låg nivå är den i *SHUTDOWN*-läget vilket kan ses som ett lågenergiläge. För att starta accelerometern sätts *ENABLE*-pinnen till hög, varpå *ACTIVE*-läget aktiveras och en mätning görs.

Efter en mätning går accelerometern in i *STANDBY*-läget, ett läge där endast möjligheten finns för att hämta ut data. *NRF51822*:n läser ut denna data från från de relevanta registeradresserna via *i2c*. Då samplingen är utläst sätts *ENABLE*-pinnen till låg nivå och accelerometern återgår till *SHUTDOWN*-läget.

5. Kommunikation

5.1 Radiokommunikation

En trådlös anslutning är inte alltid tillförlitlig på grund av att en del faktorer påverkar som strålningsbilden för antenner samt objekt i omgivningen. Objekt i nära omgivning kan ändra antennens impedans vilket ändrar resonansfrekvensen och därmed sänker sändningsstyrkan. Objekt på längre avstånd både reflekterar och absorberar den elektromagnetiska strålningen vilket kan både vara till fördel och nackdel.

Att göra en större mätning/analys kring fältstyrkan är både svår att göra då inte en specifik vagn kan antas men även då det är utanför ramarna för arbetet. Antaganden har dock gjorts kring hur omgivningen påverkar.

5.1.1 Reflektion av elektromagnetisk strålning

Noderna är inte garanterade att ha fri sikt mellan varandra och att ha en riktad antenn ses därför som en risk. En antenn med en mer spridd strålbild ger fler vägar till mottagaren som ska ta emot signalen och en del av reflektionerna mot marken och vagnarnas undersida antas kunna utnyttjas i detta syfte.

Om delar av signalen tar olika vägar samtidigt från en sändare till en mottagare kommer dessa att interferera och skapa antingen destruktiv eller konstruktiv interferens.

5.1.2 Absorption av elektromagnetisk strålning

Ett försök till att utnyttja omgivningen har gjorts genom att designa antennen och därmed strålningsbilden och få en lägre fältstyrka i riktningar där inga noder finns.

Även om det inte är under största delen av tågets användning så antas tåget köra under långa perioder på fält där omgivning i sidled kommer att absorbera all elektromagnetisk energi. Att sända ut elektromagnetisk energi i dessa riktningar anses som överflödigt och en förenkling av detta fenomen kan ses som att energin i dessa riktningar absorberas av omgivningen.

5.1.3 Impedansändring hos antennen

Antenners impedans påverkas av objekt i omgivning och framförallt de av metaller som har bra konduktivitet. Detta för att det blir en kapacitiv koppling mellan antenn, objekt och jordplan som annars är försumbar på längre avstånd. Infästningen på tågagnarna är okänd och bör inte vara ett större problem men hänsyn bör ändå tas till vid framtida ändring av infästning.

5.2 Protokoll

Protokollet är uppdelat i ett antal lager och återspeglar hur koden är strukturerad. Varje lager finns detaljerat beskrivit längre fram i rapporten.

Syftet för hela protokoll-stapeln är att tillhandahålla följande funktioner:

1. **Initiera nätverk.** Denna funktion aktiveras av en operatör i loket och det önskvärda är att tågtilhörighet stämmer överens med nätverkstillhörighet. Nätverket antas initieras då intresset finns att mäta vibrationer, vilket är då tåget börjar röra på sig. Det finns då tillfällen då flera närliggande tåg börjar röra på sig samtidigt och då en sannolikhet att noderna hos dessa tåg får tillhörighet i varandras nätverk. Detta kräver dock att noderna på tågen uppfattar varandras noder som närmare än sina egna. Operatören i loket antas upptäcka detta fel då noder uppfattas som bortfallna vid tillfället då tågen avlägsnas från varandra. För att initiera om nätverket skickas inget nytt paket till vagnarna och efter en specificerad tid i ”*application layer*” (15 sekunder då denna rapport skrevs) kommer nätverket att nollställa sig igen. När detta är gjort kan initieringsprocessen upprepas. Om två tåg följs åt efter att de börjat röra på sig och fortfarande håller ett väldigt litet avstånd mellan varandra längs med färden kan detta hindra en korrekt initiering av nätverket. Att detta inträffar anses dock som tillräckligt osannolikt för att det ska tas hänsyn till för denna prototyp. Vad som antas minimera denna risk ytterligare är faktumet att antennens strålningsbild gör att den utstrålade effekten vinkelrätt mot tågets färdriktning är lägre vilket nämns i avsnitt 4.2.
2. **Hämta data från alla noder.** En *token* skickas ut och alla noder lägger till den information som efterfrågas av huvudnoden. Token kan ställas in på att skickas ut periodiskt.

3. **Upptäcka bortfallna noder.** Om en nod försvinner kommer inte noden att lägga till informationen på sin plats i *token* och därmed tolkar "MG-Controller" det som att noden har fallit bort.

Att hämta data betyder vanligtvis att hämta vibrationsdata men det finns rum för att utvidga informationen till förslagsvis batterinivå och temperatur. Att många funktioner är *token*-baserade är främst av 2 anledningar:

1. **För att undvika konflikter kopplat till att fler än en nod introduceras till nätverket samtidigt.**
2. **Bättre utnyttjning av paket vilket kan bidra till färre paket, lägre strömförbrukning och lägre fördröjning.** Informationen varje nod vill förmedla är en liten del av vad ett paket som innehåller som är adresser och *CRC-check*. Att skicka flera noders information i samma paket innebär därför en bättre paketutnyttjning och färre paket. Trots att protokollet klarar att köa flera paket och återsändning av försvunna introduceras en fördröjning varje gång ett paket försvinner mellan två noder. En *token* gör att färre paket försvinner då två noder som når varandra har lägre sannolikhet att sända samtidigt.

Vad ett paket innehåller definieras som en "struct" i "network_layer.h" med typnamnet "network_packet". Här följer ett urklipp på just "network_packet" för en beskrivning av hur ett paket är strukturerat följt av en visuell motsvarighet (*figur 21 och 22*):

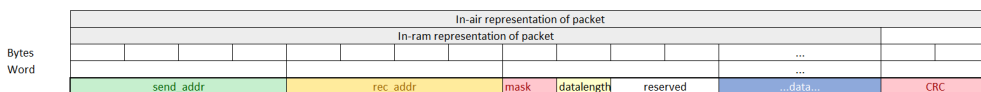
```

typedef struct { //Order is important
    //-----AVAILABLE FOR RADIO-----
    uint32_t send_addr; // <--This is where the radio write the incoming packets
    uint32_t rec_addr;
    union{
        uint8_t mask;
        struct{
            unsigned port : NBR_OF_BITS_FOR_PORT_FIELD; //2
            unsigned ack : NBR_OF_BITS_FOR_ACK_FIELD; //1
            unsigned req_ack : NBR_OF_BITS_FOR_REQ_ACK_FIELD; //1
            unsigned pid : NBR_OF_BITS_FOR_PID_FIELD; //4
        };
    };
    uint8_t datalength;
    uint32_t data[NETWORK_PACKET_MAX_DATA_SIZE_IN_WORDS];
    //-----AVAILABLE FOR RADIO-----

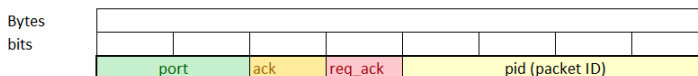
    //Other attributes
    uint8_t rssi;
    uint8_t nbr_of_retrys;
    uint8_t empty;

    network_node* node;
    void (*success_handler)(void* packet, uint8_t success);
} network_packet;

```



Figur 21: Visuellt beskrivning över hur data är strukturerat i ett paket



Figur 22: Bitfältet "mask" med attribut från paketstrukturen

Datafältets storlek anges av definitionen “NETWORK_PACKET_MAX_DATA_SIZE_IN_WORDS” som är deklarerad i “physical_layer.h”. Som nämnt i hårdvaruavsnittet har arkitekturen för *NRF51822* instruktioner för 32 bitar och för att utnyttja detta fullt ut har därför de flesta fält i paketet en längd delbar med 32 bitar och en adress i minnet för detta. Fördelarna är operationer på paketen för att jämföra adresser och att rensa, kopiera information till eller från till datafältet kräver färre beräkningar för *CPU*n.

5.2.1 Physical layer

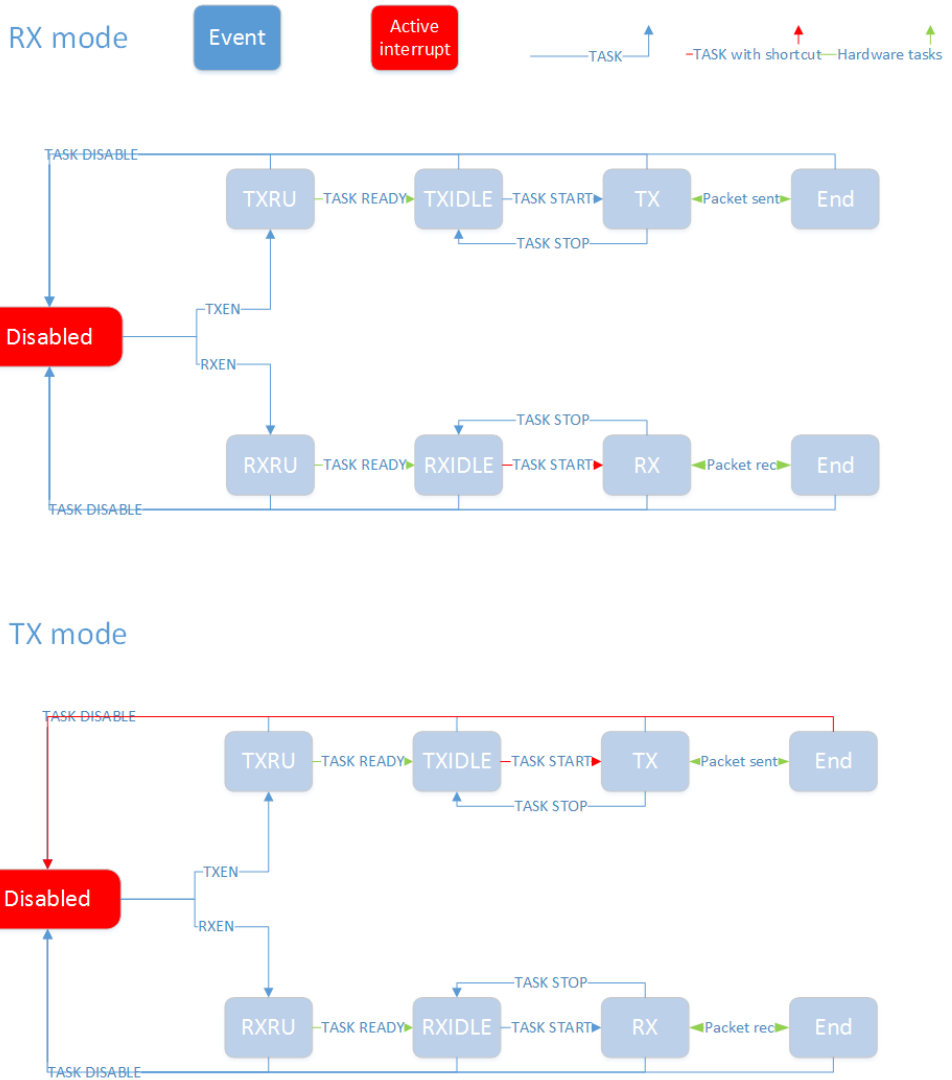
Namnet på detta lager kan vara missvisande då det inte är ett fysiskt lager men har fått sitt namn då det tar hand om hårdvarufunktioner relaterat till protokollet som radio, RSSI-avläsning och feldetektering med *CRC-16*. Detta sker genom skrivning till och läsning från register samt hantering av avbrott relaterat till dessa. Den är konfigurerat att ge radion maximal effekt men det går att sänka denna med ändring av en global variabel. Maximal effekt är +4dBm för *NRF51822*.

Dess syfte till övre lager är att skicka paket men ger ingen bekräftelse eller garanti. För att skicka ett paket anropas en metod och för att för att tolka de mottagna skickas en funktionspekare in vid initiering av detta lager. Funktionspekaren anropas med en pekare till paketet då paketet är korrekt mottaget. Innan det mottagna paketet skickas till funktionspekaren skrivs det uppmätta RSSI-värdet till objektet som har ett värde mellan 0 och -127 i dBm. Ett mer negativt värde betyder en lägre uppmätt signalamplitud vid tillfället paketet togs emot.

Grundprincipen för detta lager är att lyssna konstant och att aktivera sändläge så kort tid som möjligt för att vara i aktivt sändläge endast då ett paket behöver skickas. När ett paket är skickat återgår läget till

mottagning och lämpar sig därför till övre lager som har ACK-funktionalitet. För en mer detaljerad förklaring till hur detta lager hanterar olika lägen hänvisas till *figur 23*, där *TX* och *RX* motsvarar sänd- respektive mottagarläge.

Något som ett övre lager bör ta hänsyn till är att byta pekare till var i ramminnet ett nyinkommet paket bör skrivas till, om det vill kunna spara undan paketet utan överskrivning av ett nytt.



Figur 23: Flödesschema för "physical layer"

5.2.2 Network layer

5.2.2.1 Begränsande av sändningfrekvens

Detta lager definierar den minimala tiden som går mellan sändning av två paket. Om ett överliggande lager skickar ett flertal paket direkt in på varandra kommer alla utom det första att schemaläggas och skickas med ett intervall som motsvarar den kortaste tiden mellan sändning av paket. Denna funktion finns för att inte överbelasta underliggande lager samt för att kunna lyssna till ett *ACK* som svar efter ett sänt paket.

Ytterligare en funktion är att varje varje anrop som görs från underliggande lager schemaläggs för att behandlas så snabbt som möjligt. Detta för att isolera det understa lagret, som kör metoder från avbrott, från resten av programmet. Om resten av programmet inte körs från avbrott är det mer förutsägbart enligt *timer* i kapitel 4.3.1 som schemalägger alla händelser.

5.2.2.2 Hårdvaruadress

Adressen fås från ett hårdkodat register som är unikt för alla noder som läses in vid initiering av lagret. Lagret ansvarar för att alla utgående paket som skickas innehåller nodens adress. På motsvarande vis kollar den även alla inkommande paket och vilka adresser de kommer ifrån. Om adressen är 0 vidarebefordrar den paketet till *broadcasthandler* och om de är en adressmatchning kommer paketet att vidarebefordras till *socketlistener*.

5.2.3 Discovery layer

Denna del har kopplingar till flera lager då den tillhandahåller en lista på noder som anses som nåbara för ögonblicket. Då detta initieras börjar noden att kontinuerligt skicka ut information om sig själv som benämns som *broadcast* i koden. På motsvarande vis lyssnar den på alla mottagna paket för att noderna på listan och deras uppdatera informationen kring

mottagningsstyrkan, hårdvaruadress och adress på nätverket. Paketen den lyssnar på är, utöver *broadcasts* från andra noder, paket som adresserade till denna nod eller andra noder. Om den inte hör något paket från en nod på listan inom en definierad tid tas denna nod bort från listan och anses som icke nåbar.

Varje nod i listan har ett värde som ger en uppskattning om hur nära en nod i listan är. Värdet grundar sig på RSSI-mätning som gör på alla inkomna paket med sparar ett genomsnittligt värde genom lågpasfiltrering. Värdet använder *application layer* sig av för att bedöma i vilken ordning noderna är ordnade på vagnarna och detta ligger till grund till vem den ska skicka *token* till.

5.2.4 Transport layer

Detta lager lägger till en pålitlighet och rapportering på underliggande lager som är av typen “best effort”. Om ett paket skickas via detta lager finns en möjlighet att skicka med en funktionspekare som sedan rapporterar framgången av det sända paketet. Detta är en kritiskt funktion som *application layer* tillförlitar sig på vid sändning av token för att inte token ska försvinna.

När ett paket skickas via detta lager schemaläggs en rutin för att se ifall paketet har fått ett svar (ACK). Om inget svar har kommit inom denna tid skickas paketet ut igen. När ett svar kommer rapporteras detta via funktionspekaren direkt för minimal fördröjning. Det maximala antalet återförsök finns globalt definierat som “MAX_NBR_OF_ATTEMPS” och är således enkelt att ändra i koden. Om antalet återförsök överstiger detta definierade antal kommer det att rapportera misslyckande till den funktionspekare i övre lager som skickades i samband med begäran om att skicka paketet.

5.2.5 Application layer

Detta lager i protokollet står för kommunikationen via andra noder och det introducerar därför ett ansvar hos varje nod att vidarebefordra andra noders paket korrekt. Hela protokollet är baserat på en så kallat *token* som kan jämföras med en stafettpinne och logiken för detta lager finns förklarat i ett flödesdiagram nedan.

Utöver logiken som diagrammet har detta lager en tidsgräns för hur länge en nod får hålla en token och hur länge den får vara med i ett nätverk när den inte har fått *token* inom en definierad tid. Dessa två tidsgränser är definierade i sekunder som

“TIMEOUT_HOLDING_TOKEN_IN_SECONDS” och

“TIMEOUT_NETWORK_INHERENCE_IN_SECONDS” med en

införstådd avvikelse i tid på grund av avrundning då dessa tider kollas periodiskt. Syftet med dessa tidsgränser är att ett fel på en nod inte ska

hindra *tokens* framfart för andra noder. Om exempelvis två nätverk

initieras på två olika tåg samtidigt finns det en risk att noderna på

respektive tåg kan bli medlemmar i varandras nätverk. När tågen rör sig

ifrån varandra kommer noderna ifrån sitt nätverk de var medlemmar i och automatisk lämna detta efter

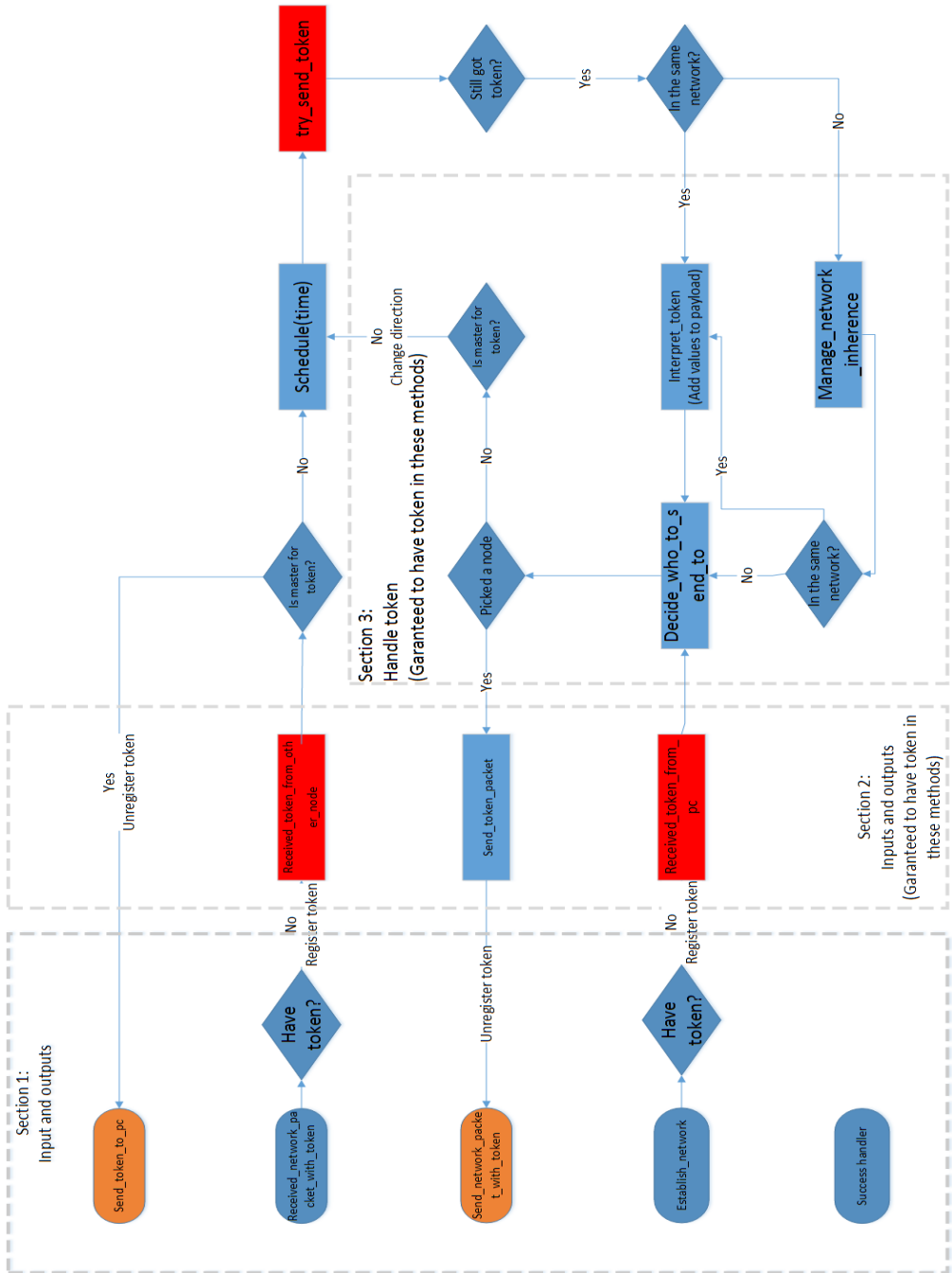
“TIMEOUT_NETWORK_INHERENCE_IN_SECONDS” sekunder. För

att upprätta ett nytt nätverk med de korrekta noderna väntar man ut tiden

det tar för huvudnoden att nollställa sitt nätverk vilket innebär att alla

andra noder också har nollställt sin nätverkstillhörighet. Därefter skickas

en ny token ut som upprättar nätverket på nytt.



Figur 24: Flödesdiagram för logiken bakom application layer

De olika sektionerna som kan observeras i *figur 24* är till för att enkelt hitta var de olika delarna är identifierade i programkoden för detta lager. All schemaläggning använder sig av timerfunktionen beskriven i avsnitt 4.3.1 och därför kommer blocken i diagrammet som följer från detta inte att avbrytas. Efter en schemaläggning måste dock *tokens* existens bekräftas då någonting kan ha ändrat.

5.2.5.1 Token

För att det inte två noder ska kunna bli introducerade i nätverket samtidigt kan endast noden med token introducera en ny nod.

Det token innehåller är följande:

1. NätverksID.
2. Högsta adressen i nätverket.
3. Information om vilken riktning (ifrån eller till huvudnoden).
4. Information om hur många bytes varje nod ska rapportera.
5. Vibrationsinformation från noderna. Denna nyttolast är den information som systemets huvudsyfte är att förmedla till huvudnoden och datorn i loket.

Då *token* innehåller den högsta adressen på nätverket och därmed räknar systemet med att alla noder kommer att lägga sin information på sin plats i nyttolasten. Platsen i nyttolasten motsvarar adressen på nätverket och är således unik. Värdet "0" i nyttolasten betyder att noden inte har fått token och värdet "1" betyder det lägsta värdet, motsvarande 0 som mätsignal. Denna reservering av värde används för att kunna ta reda på ifall en nod inte har fått det mottagna *token*-paketet och inte gett något svar. Endast i riktningen från huvudnoden lastas fältet för nyttolast med vibrationsinformationen.

5.2.5.2 Initieringsprocess

Att initiera ett nätverk innebär att skapa en tom token med en riktning bakåt och med ett *nätverksID* som är samma som hårdvaruadressen till den nod som initierar. Noden som initierar nätverket är huvudnoden som är kopplad till datorn i tåget.

5.2.5.3 NätverksID

NätverksID finns för att undvika konflikter med ett annat tåg och är baserat på hårdvaruadressen hos den nod som initierar nätverket och är således unik. Det finns ett *nätverksid* och adress som är specifik för detta lager och adressen tilldelas när en nod går med i ett nätverk. Om *token* hamnar hos en nod som tillhör ett annat nätverk kommer denna noden inte att påverka token utan bara vidarebefordra denna.

5.2.5.4 Adress

Adressen kopplat till detta nätverk används för att kunna särskilja noderna från varandra och den kan ses som . Denna adress kopplat till nätverket är känd som "sw_addr" i koden för att särskilja från adressen som används i "network layer" för den direkta radiokontakten.

5.2.5.5 Vägval

När *token* ska skickas vidare till en ny nod väljes den nod som har lägst negativt RSSI-värde, vilket ska ge en uppskattning om vilken som är närmast. Om endast noden *token* skickades ifrån finns tillgänglig kommer noden att uppfatta sig som den sista på kedjan och byta riktning på *token*, kallad *forward* i koden. Denna riktning betyder att *token* ska gå tillbaka samma väg som den kom ifrån och målet är att komma tillbaka till huvudnoden som skickade ut den från början.

6. Resultat



Figur 25: Modulerna färdigmonterade

Det fysiska resultatet av lösningen på produkten är 5 stycken prototyper av noder som kan observeras på *figur 25*. Huvudnoden som är nämnd i rapporten är en enhet från det *developer kit* som inhandlades för utveckling med *NRF51822*.

6.1 Länkpålitlighet



Figur 26: Mätningen med 3 noder på stolar



Figur 27: Mätningen med 2 noder på marken

Test utfördes på en parkering med singelsten. Det första testet bestod i placera två noder på marken med antennerna parallella med varandra och markplanet (*se figur 27*). Det sågs till att endast en av dessa två noder hade kontakt med en huvudnod kopplad till en laptop. På laptopens skärm kunde systemet observeras och ifall paketen kom tillbaka. Testet upprepades med ökat avstånd mellan de två noderna och vid ca 4-5 meter tappade noderna kontakten.

Det andra testet utfördes som det första men med skillnaden att noderna låg på var sin stol (*figur 26*) av plast då de kom ca 40 cm över marken. De var fortfarande parallella med varandra och markplanet. Resultatet visade en avsevärd ökning och först vid 20-25 meter började paket att försvinna.

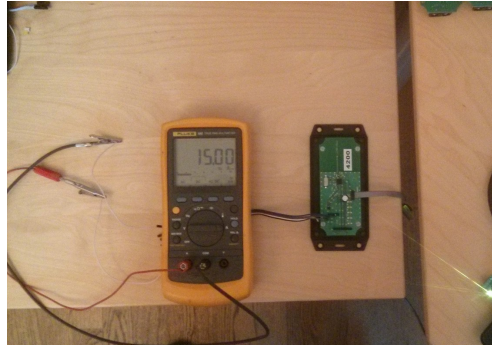
För det tredje testat introducerades en tredje nod på sin plaststol för att testa kommunikation mellan 3 noder. Detta test visade på ett maximalt avstånd på ca 15 meter mellan varje nod. Ytterligare observationer som bör tilläggas är att vid de uppmätta avstånden gällde när noderna och omgivningen var fast och om någon nod rörde på sig under mätningen kunde ett längre avstånd uppnås.

6.2 Strömförbrukning

För mätning av strömförbrukning (se figur 28) kopplades en multimeter till ett mönsterkort där strömbrytaren annars är placerad. Alla funktioner som används i felsökningsläge av inaktiverade, vilket innefattar lysdioder och kommunikation via UART.

Strömförbrukningen

observerades under tiden huvudnoden med olika frekvenser sände ut token till endast denna nod. Det syntes ingen påtaglig skillnad mellan de olika frekvenserna upp till 5 Hz och strömförbrukningen låg konstant nära 15 mA.



Figur 28: Mätning av strömförbrukning

7. Diskussion

7.1 Rekommendationer för vidareutveckling

I en miljö där det finns mycket elektronik som sänder på samma frekvenser som noderna skulle *FHSS*[9] kunna implementeras. *Frequency-Hopping Spread Spectrum* innebär att sändare och mottagare kommer att hoppa mellan olika frekvenser i ett förutbestämt mönster. Detta innebär att risken att störa annan utrustning eller själv bli störd är mycket mindre.

För att minska strömförbrukningen vid inaktivitet som då tåget står stilla vid bangården vid denna situation skulle MCU:n kunna gå ner i ett djupare sovläge där den endast lyssnar på avbrott från en viss pinne.

Detta avbrott kan då kopplas till en accelerometer med enklare hårdvarufunktion såsom *Tilt-detection* där accelerometern har en dedikerad pinne per axel som blir hög om totalvärdet för accelerationen på en axel överstiger ett visst värde. På så sett så kan systemet sova när tåget står stilla men vakna upp så fort tåget börjar röra på sig.

Valet av batteri för prototypen hade inte hög prioritet och det enda som spelade in i val av batteri var spänningen och storlek vilket resulterade i att prototyperna använder vanliga alkaliska AAA-batterier. Alkaliska batterier uppfyller sin funktion men i den mer extrema temperaturerna som den färdiga produkten kommer utsättas för och med ett krav på en lång hållbarhet kan ett Litiumbatteri vara mer lämpligt[10][11].

7.2 Alternativ lösning

Att detta protokoll är skrivet helt från grunden är ineffektivt ur ett tidsperspektiv för utveckling där det blir fler delar av programmet som måste göras och testas. Ett alternativ är att lägre lager hade varit befintliga protokoll, likt *TCP/IP* över *Wi-Fi*, *Bluetooth*. Ett annat alternativ hade varit att ersätta en ännu större del av programmet med befintliga flerpunktsprotokoll likt *WirelessHART*, *ZigBee*, *SNAP* eller *ANT*.

Fördelarna med att allt är gjort från grunden är att systemet är helt anpassat till sitt användningsområde med endast de nödvändiga funktionerna och således finns det en möjlighet till lägre resurskrav.

Ett system som kräver så lite underhåll som möjligt är önskvärt och med låg strömförbrukning tillåter det fler sätt att strömförsörja produkten på. Om batteridrift ska vara aktuellt för den framtida produkten kräver det att batterierna behöver bytas med lång tid i mellan, eller inte alls då batteriernas livslängd motsvarar resten av komponenternas livslängd. Andra förslag på framtida energiförsörjning som har uppkommit under projektets gång är utnyttjande av vibrations-, rotations- eller ljusenergin. Den hårda miljön gjorde att ett alternativ med ljusenergin förkastades och den mest potentiella bland de förnyelsebara energikällorna ansågs vara att utnyttja vibrationerna med ett piezoelektriskt element för konvertering till elektrisk energi.

7.3 Resultatanalys

Strömmätningarna visade på 15 mA strömförbrukning som var oberoende av frekvensen av antalet passerande paket. Detta känns som rimligt då *NRF51822* kräver 13 mA i aktivt mottagarläge och detta är konstant aktivt utöver den tiden som information sänds. Denna strömförbrukning är förhållandevis hög i jämförelse till, 2,3 μA , då alla tillbehör på detta *SoC* är inaktiva. Den låga förbrukningen visar på hur mycket energi man kunnat spara ifall systemet hade varit inaktiv en större tid med uppväckning av en lågfrekvent extern kristaloscillator eller den interna RC-oscillator. Tiden systemet vaknar upp på kan vara en avtalad tidpunkt mellan enheter då systemet utför sina radiorelaterade aktiviteter för att sedan gå ner i strömsparläge igen. Denna teknik finns på befintliga protokoll och skulle kunna implementeras i "physical layer" från 4.3.1.

Testet för länktillförlitlighet är till för att ge en uppskattning för vilka avstånd som är möjliga att kommunicera på och marginalen för dessa. Miljön under tågagnar nprototyperna är avsedda att kommunicera i är svår att återskapa och även den är inte helt fastställd med olika typer av tågagnar och miljön utanför dessa.

När noderna höjdes upp på en stol ökade räckvidden avsevärt vilket kan antas vara reflektionerna mot marken som bidrog. Vinkeln av den reflekterade signalvägen blev således större mot marken och mindre av den utsända effekten sändes mot himmeln, vilken antas inte reflektera tillbaka någon energi. Detta kan styrka teorin om att monterad på sitt avsedda ställe kunna utnyttja reflektionerna mot marken och undersidan av tågagnar för längre räckvidd. Dock bär hänsyn tas till att mätningarna togs med underlag av torra singelstenar och det finns betydligt mindre fördelaktig markbeläggning som inte reflekterar lika bra.

8. Slutsats

Målet med arbetet uppfylldes då resultatet var ett förslag på hur ett system som ska mäta vibration på ett tåg och föra över denna information trådlöst till loket. Förslaget består av 5 st fungerande prototypmoduler, program- och hårdvara för datorn. Modulerna använder varandra för att vidarebefordra informationen och deras hårdvara består av ett enkelt system med få komponenter med potential att väldigt vara strömsnålt. Det visar därmed på möjligheterna för ett slutgiltig produkt att bli billig. Även om systemet syftar till att användas för tågagnar gjordes ingen mätning på ett tåg utan istället gjordes uppskattningar för hur det borde fungera i praktiken. Det hade dock varit intressant att se resultaten för en mätning hur väl prototyperna hade motsvarat förväntningarna. Förhoppningar fanns för om att göra detta men tiden räckte inte till.

Prototyperna är en del av utvecklingen av ett system i en tidig fas. Detta gjorde kraven lösa och tanken är att framtagningen av denna prototyp ska ge en bättre uppfattning av vad som kommer att krävas av den slutgiltiga produkten. Möjligheten finns att denna kommer efter några iterationer av nyare prototyper att vara väldigt olik resultatet av detta arbete.

Projektet har varit en givande utmaning för oss båda och vi har båda lärt oss en hel del om utveckling. Den största utmaning var omfattningen med att få alla områden med mekanik, konstruktion, kommunikation och mjukvara att fungera tillsammans och att samtidigt ta fram konceptet.

9. Referenser

- [1] <http://www.antenna-theory.com/tutorial/smith/chart.php>.
- [2] Antenna tuning nWP-017, Nordic Semiconductor 2012.
- [3] <https://www.mems-exchange.org/MEMS/what-is.html>.
- [4] <http://www.sensr.com/pdf/practical-guide-to-accelerometers.pdf>.
- [5] Maria Kihl, *Datakommunikation : en inledande översikt*, 2006 sida 53.
- [6] http://uotechnology.edu.iq/appsciences/material/Lecture_material/second_class/Materials_properties/8.pdf.
- [7] NRF51822 reference manual v1.1, Nordic Semiconductor 2013.
- [8] [http://cache.freescale.com/files/sensors/doc/ data_sheet/MMA8491Q.pdf](http://cache.freescale.com/files/sensors/doc/d ata_sheet/MMA8491Q.pdf).
- [9] <http://www.futaba-rc.com/technology/fhss.html>.
- [10] <http://www.doitpoms.ac.uk/tlplib/batteries/primary.php>.
- [11] <http://www.kjell.com/fraga-kjell/hur-funkar-det/elelektronik/batterier/engangs-batterier>.

10. Ordlista

1. PPI - Programmable Peripheral Interface. Kopplar integrerad hårdvara till varandra med tasks och events utan att inblandning av CPU:n. Det möjliggör att genom ett event kan aktivera en specifik *task* varje gång det inträffar.
2. Event - Säger vad som har aktiverat interruptrutinen. Flera *events* kan trigga samma avbrottsrutin.
3. Task - En uppgift för hårdvaran som aktiveras från hård- eller mjukvara. Kan t.ex. vara att aktivera mottagarläget för radion.
4. Shortcuts/Shorts - Gör att en task aktiveras när ett event har utfört inom samma funktion utan någon inblandning av CPU.
5. GFSK - Gaussion Frequency Shift Keying. Anger hur radion tolkar bitarna genom frekvensskift.
6. ACK och NACK - ACKnowledgement and Negative ACKnowledgement.
7. RSSI - Radio Signal Strength Index. Ett värde i dBm i hur stor amplitud det är på signalen från antennen. Ger ett uppskattat värde på signalstyrkan för en inkommet paket.
8. Best effort - Används i samband med datakommunikation. Innebär att överföringen gör vad den kan för att överföra information men garanterar inte att paketet blir överfört.
9. Boggi (Bogie, eng.) - Den del av en tågagn hjulens axel är fäst i. Det är denna del som är intressant att mäta vibrationer på.
10. Funktionspekare - En referens till en exekverbar del av kod i minnet.
11. SoC - System On a Chip. En integrerbar krets som integrerar komponenter från elektrisk system i ett och samma chip. Kan innehåller alla komponenter till en dator och samtidigt analoga kretsar.

11. Bilagor

11.1 Bill of material

Förklaring	Värde/egenskap	Betäckning (PCB)	Länk	Extra information		tal
	0.1 µF	C1	http://se.farnell.com/avx/cm105x7r104k16at/cap-mlcc-0-1uf-16v-x7r-10-0603/dp/1216538RL		1	10
	4.7 µF	C2, C4	http://se.farnell.com/murata/grm188r60j475ke19d/capacitor-0603-4-7uf-6-3v/dp/1735527RL		2	20
	1.0 F	C3	http://se.farnell.com/multicomp/mc0603x105k100ct/cap-ceramic-1uf-10v-x5r-0603/dp/1759399RL		1	10
	12 kOhm	R7	http://se.farnell.com/multicomp/mc-0-063w-0603-5-12k/resistor-12-kohm-0-063w-5/dp/9331760RL		1	10
	4.7 kOhm	R5, R6	http://se.farnell.com/bourns/cr0603-jw-472elf/resistor-0603-4-7k-5-0-1w/dp/2008358RL		2	20
						0
Resistorer till LEDs	100 Ohm	R0-R4	http://se.farnell.com/bourns/cr0603-jw-101glf/resistor-0603-100r-5-0-1w/dp/2008349RL		5	50
Kondensator 2.2 nF		C12	http://se.farnell.com/murata/grm1885c1h222ja01d/cap-ceramic-2200pf-50v-c0g-np0/dp/8819947RL		1	10
Kondensator 2.2 pF		C13	http://se.farnell.com/avx/ml03v12r2bat2a/capacitor-2-		1	10

			2pf-0-1pf-250v/dp/2252666			
Kondensator 1.0 pF		C14	http://se.farnell.com/avx/ml03v11r0bat2a/capacitor-1-0pf-0-1pf-250v/dp/2252666		1	10
Kondensator 1.5 pF		C15	http://se.farnell.com/avx/ml03v11r5bat2a/capacitor-1-5pf-0-1pf-250v/dp/2252658		1	10
kontakt till programmering	5x2 pinnars 1,27 mm pitch	SWD	http://se.farnell.com/harwin/m50-3500542/connector-header-tht-1-27mm-10way/dp/1022305		1	10
Female Header till GPIO		GPIO	http://se.farnell.com/multicomp/2212s-08sg-85/socket-pcb-1-row-8way/dp/1593463		1	10
motsvarande kontakt till sladd			http://se.farnell.com/molex/22-01-3027/housing-square-pin-2-54mm-2way/dp/1462825		1	10
Distanser			http://se.farnell.com/durato/d01496/hex-threaded-spacer-nylon66-natural/dp/1733428?crosssellid=1733428&crosssell=true&in_merch=true&		4	40
Skruv (förp 25 st)			http://se.farnell.com/durato/d00687/screw-pan-m3x6-pk25/dp/1514409			3
						0
AAA-batterihållare		BT1	http://se.farnell.com/keystone/2468/battery-holder-pcb-2-aaa/dp/1650679		1	10
DC-kontakt till PCB		DC	http://se.farnell.com/molex/22-05-3021/connector-header-tht-ra-2-		1	10

			54mm/dp/1756797			
Strömswitch		switch	http://se.farnell.com/erg-components/jsa4-1-go/switch-jumper-1-way-spst-slide/dp/3289758			1 10
Accelerometer	mma8491qr1	ACCEL	http://se.farnell.com/freescale-semiconductor/mma8491qr1/accelerometer-3-axis-8g-12qfn/dp/2291592			1 10
Låda			http://se.farnell.com/hammond/1591xxcsflbk/box-abs-flanged-121x66x40mm/dp/1511194			8
LEDs		LEDn, n=[0:1:4]	http://se.farnell.com/kingbright/kptd-3216src-prv/smd-led-dome-lens-1206-red/dp/2001662RL	Röda 1206 1.85 V		5 50
Krystall		Y1	http://se.farnell.com/txc/9c-16-000meej-t/xtal-16-000mhz-18pf-smd-hc-49s/dp/1842294	16 pF +- 10ppm		1 10
Induktans till antenn 10 nH		L3	http://se.farnell.com/wurth-elektronik/744761110a/inductor-0603-10nh-0-7a/dp/1748760RL	Till antenn (high freq) 10 nH 0603 ± 5%		1 10
Induktans till antenn 4.7 nH		L2	http://se.farnell.com/epcos/b82496c3479j/inductor-4-7nh-800ma-5-0603/dp/3876998RL	± 5%		1 10
Induktans till antenn 3.3 nH		L4	http://se.farnell.com/epcos/b82496c3339a/inductor-3-3nh-0-3nh-1200ma-5-5ghz/dp/3876986RL	± 0.3nH		1 10
Induktans till transformatorn		L1	http://se.farnell.com/tdk/mlz1608m100wt/inductor-shielded-10uh-20/dp/2215635	17 MHz +- 20%		1 10

Kapacitans till DEC2 (krav > 47 nF)	C11	http://se.farnell.com/avx/06033c473kat2a/cap-mlcc-0-047uf-25v-x7r-10-reel/dp/2280847		1	10
Kap till DEC1 och VDD (krav > 100nF)	C9, C5	http://se.farnell.com/avx/cm105x7r104k16at/cap-mlcc-0-1uf-16v-x7r-10-0603/dp/1216538RL		2	20
Kapacitans till kristall 8.2 pF	C7,C8	http://se.farnell.com/vishay/vj0603d8r2cxpaj/capacitor-0603-8-2pf-0-25pf-250v/dp/2134040RL	0.25 pF	2	20
Kapacitans till kristall 9 pF (extra)	C7,C8	http://se.farnell.com/multicomp/mcca000191/mlcc-0603-np0-50v-9pf/dp/1759052RL	0.5 pF	2	20
Kapacitans för matning på AVVD 1 nF	C10	http://se.farnell.com/avx/06035c102kaz2a/cap-mlcc-1000pf-50v-x7r-10-0603/dp/7569513RL		1	10