

Bachelorthesis

Draadloze communicatie in een MRI-omgeving

Tobias Dekker 1528688

Michiel Jaspers 1517295

Faculteit Elektrotechniek, Wiskunde en Informatica

30 juni 2012

Delft University of Technology



Voorwoord

Deze thesis is geschreven voor het bacheloreindproject van de opleiding Electrical Engineering aan de Technische Universiteit Delft.

De opdracht, het verbeteren van de sensor en het transmissiesysteem voor het meten van een hartsignaal in een MRI-scanner, is bedacht door Wyger Brink, een PhD-student die momenteel onderzoek doet bij het LUMC. Hier is geconstateerd dat de nieuwste MRI-scanner niet volledig correct functioneert.

De opdracht is opgedeeld in drie onderdelen waar elk twee studenten aan werken. Deze thesis bevat het deel over de transmissie van sensordata in een ruimte met een 7 T MRI-scanner.

Onze dank gaat uit naar:

Gerard Janssen (Technische Universiteit Delft)

Angeniet Kam (Technische Universiteit Delft)

Wyger Brink (Leids Universitair Medisch Centrum)

Auke Booij (Student Technische Universiteit Delft)

Michiel Jaspers en Tobias Dekker
Delft, 30 juni 2012,

Samenvatting

Met behulp van een MRI-scanner kan er bij een patiënt een afbeelding gemaakt worden van de binnenkant van het lichaam zonder dat deze opengemaakt hoeft te worden. Deze afbeeldingen worden gemaakt om een diagnose te stellen. De arts kan beter zien wat er aan de hand is met de patiënt als de resolutie van de afbeelding verbetert. De resolutie kan verbeterd worden door een MRI-scanner te kiezen met een hogere magnetische veldsterkte. Een ander aspect dat meespeelt hoe goed de arts kan zien wat er aan de hand is, is de scherpte van de afbeelding. Deze wordt mede bepaald door hoe nauwkeurig de hartslag kan worden gemeten, want met behulp van de hartslag kan op dezelfde fase in de hartcyclus een scan worden gemaakt. Door verschillende scans die op hetzelfde moment in de cyclus zijn genomen over elkaar te leggen wordt het contrast beter.

Uit een Electrocardiogram (ECG) kunnen vaste punten in de hartcyclus worden gedetecteerd. Deze hartmeting wordt echter ernstig verstoord in een MRI-scanner door het statische magnetische veld. Hierdoor worden op verkeerde momenten in de hartcyclus een scan gemaakt. Dit zorgt voor minder scherpe afbeeldingen. Dit is een probleem voor ziekenhuizen met een 7 T scanner zoals het LUMC.

Om het probleem op te lossen wordt een nieuwe draadloze sensor ontwikkeld die de hartslag kan meten. De ontwikkeling van de sensor is opgedeeld in drie onderdelen:

- Acquisitie van het biomedisch signaal waarop cardiac triggering kan plaatsvinden;
- Omzetting van het analoge sensorsignaal in een digitaal signaal en onderdrukken van de ruis;
- Transmissie van het digitale signaal vanaf de sensor op de patiënt in de MRI-scanner naar een computer en detectie van het triggersignaal.

Drie verschillende teams werken elk aan hun eigen onderdeel. In deze thesis is onderzocht wat de beste methode is om draadloos een sensorsignaal te versturen in een MRI-scanneromgeving. Het sterke magneetveld en de RF golven op 298 MHz vormen de grootste storingsbronnen. Er wordt gekozen voor een draadloos systeem omdat een bedraad systeem in een MRI-omgeving kan leiden tot ernstige brandwonden. Daarnaast heeft de opdrachtgever kenbaar gemaakt een draadloos systeem te prefereren zodat de sensormodule een grote mobiliteit heeft.

De volgende vormen van draadloze communicatie zijn onderzocht: optische communicatie, akoestische communicatie en radiofrequente communicatie. Op basis van de eigenschappen van deze technieken is een keuze gemaakt voor de gebruikte communicatie.

Optische communicatie op basis van infrarood is voor dit doel niet geschikt omdat er een direct zicht nodig is. Dit is een probleem omdat de communicatielink verbroken zou kunnen worden indien de patiënt beweegt. Een ander nadeel is dat de maximale afstand die gegarandeerd wordt slechts 1 meter is.

Voor afstanden van meer dan 3 meter is akoestische communicatie niet geschikt, omdat de betrouwbaarheid van de verbinding erg slecht wordt. Daarnaast bleek verder dat de datasnelheid veel te laag was.

Voor digitale radiofrequente communicatie zijn een aantal protocollen beschikbaar. Deze verschillen voornamelijk in maximale datasnelheid, afstand en vermogen.

Bluetooth en ZigBee zijn het best geschikt voor het product, waarbij Bluetooth een hogere datasnelheid heeft en ZigBee minder energie verbruikt. Aangezien de datasnelheid van ZigBee voldoende is om meerdere signalen te zenden en te ontvangen wordt ZigBee geselecteerd. De MRF24J40MA, een ZigBee module, is

uiteindelijk gekozen, omdat deze als enige een minimale hoeveelheid ferromagnetische materialen bevat en op korte termijn leverbaar is.

Er worden ZigBee transceivers gebruikt omdat er op een bidirectionele manier moet worden gecommuniceerd. Één ZigBee transceiver wordt aangesloten op een computer zodat deze de hartslag en de temperatuur kan weergeven. Een microcontroller zorgt voor de interface tussen de transceiver en de pc.

De andere microcontroller aan de kant van de sensor detecteert de belangrijkste harttoon uit het akoestische hartsignaal en biedt aan de uitgang een puls aan naar de MRI-scanner. Voor het prototype is deze uitgang aangesloten op een Light-emitting diode(LED).

Voor het prototype wordt er gebruik gemaakt van een eerder opgenomen akoestisch signaal. Dit signaal duurt in totaal 6 seconden en wordt continu herhaald.

Er is ook een Graphical User Interface (GUI) gemaakt. Deze interface biedt een medicus de mogelijkheid om snel en duidelijk te zien wat de hartslag en de temperatuur van de patiënt is. In de GUI wordt weergegeven waar er wordt getriggerd in het akoestische hartsignaal. De GUI is gemaakt met MATLAB aangezien dit het minste tijd zou kosten en de auteurs de meeste ervaring hadden met deze programmeertaal.

Helaas is het niet gelukt om het transmissie gedeelte aan te sluiten op de producten van de andere teams. Ook zijn er nog geen testen gedaan in de MRI-scanner. Het is waarschijnlijk dat het transmissiesysteem goed zal functioneren in de MRI-scanner, aangezien de ZigBee modules zeer weinig ferromagnetische materialen bevatten. Dit kan echter pas getest worden als het hele systeem op elkaar is aangesloten.

Lijst van afkortingen

<i>AIr</i>	Advanced infrared
<i>BER</i>	Bit error rate
<i>CSMA/CA</i>	Carrier Sense Multiple Access with Collision Avoidance
<i>DSSS</i>	Direct Sequence Spread Spectrum
<i>ECG</i>	Electrocardiogram
<i>FOV</i>	Field of View
<i>GUI</i>	Graphical user interface
<i>IrDA</i>	Infrared Data Association
<i>IR</i>	Infrarood
<i>LED</i>	Light-emitting diode
<i>LOS</i>	Line of Sight
<i>LUMC</i>	Leids Universitair Medisch Centrum
<i>MAC</i>	Media acces control
<i>MHD</i>	Magneto-hydrodynamische
<i>MRI</i>	Magnetic Resonance Imaging
<i>NRZ</i>	Non-return-to-zero
<i>OSI</i>	Open Systems Interconnecting
<i>PCG</i>	Phonocardiogram
<i>RF</i>	Radiofrequent
<i>RoHS</i>	Restriction of Hazardous Substances Directive
<i>SPI</i>	Serial Peripheral Interface Bus
<i>STFT</i>	Short-Time Fourier Transform
<i>TDM</i>	time-division-multiplexing
<i>TTL</i>	Transistor transistor logic
<i>WPAN</i>	Wireless Personal Area Network

Inhoudsopgave

Voorwoord	i
Samenvatting	ii
Lijst van afkortingen	iv
1 Inleiding	2
2 Literatuurstudie	4
2.1 Effecten op apparatuur in een MRI-omgeving	4
2.2 Draadloze communicatie op korte afstand	5
2.2.1 Akoestische communicatie	5
2.2.2 Infraroodlicht	5
2.2.3 Radiofrequente golven	5
2.3 Analyse van het hartsignaal voor triggering	6
3 Programma van eisen	8
3.1 Eisen vanuit het beoogde gebruik	8
3.2 Eisen vanuit de ecologische situering van het systeem in de omgeving	9
3.3 Eisen met betrekking tot het ontwerpen van het systeem	9
3.4 Eisen met betrekking tot het eventueel te ontwikkelen productiesysteem	9
3.5 Eisen met betrekking tot het te ontwikkelen liquidatiesysteem	9
3.6 Eisen vanuit bedrijfsstrategische, marketing- en verkooptechnische omstandigheden	10
3.7 Randvoorwaarden	10
4 Communicatiesysteem in een MRI-scanner	11
4.1 Hardware MRI	11
4.2 Werkingsprincipe MRI	12
4.3 Effecten op transmissie	13
4.4 Communicatievormen	13
4.4.1 Optische communicatie	13
4.4.2 Akoestische communicatie	14
4.4.3 Radiofrequente communicatie	15
5 Keuze van het definitieve ontwerp	17
5.1 Criteria	17
5.2 Toetsing van de ontwerpalternatieven aan de criteria	18
5.2.1 Optische communicatie	18
5.2.2 Akoestische communicatie	18
5.2.3 Radiofrequente communicatie	18

6	Topologie van het ontwerp	20
6.1	Functionaliteiten van het systeem	20
6.2	Topologie van het systeem	20
7	Implementatie en realisatie	23
7.1	Communicatie link	23
7.1.1	Opbouw van de datapakketten	25
7.2	Keuze microcontroller en interface met ZigBee en computer	26
7.3	Graphical User Interface	27
7.3.1	Afweging voor programmeertaal	28
7.3.2	GUI in MATLAB	29
7.4	Triggering	31
7.5	Programmeren Arduino	31
8	Testen prototype	32
8.1	Opstelling	32
8.2	Prestaties van ZigBee	32
8.3	Eindresultaten	34
9	Conclusies en aanbevelingen	38
9.1	Conclusies	38
9.2	Aanbevelingen	38
	Bibliografie	41
A	Instelregisters van de ZigBee	42
B	MATLAB-code	43
C	Code Arduino Zigbee 1	50
D	Code Arduino ZigBee 2	54

Hoofdstuk 1

Inleiding

MRI (Magnetic Resonance Imaging) is een belangrijke diagnostische techniek in ziekenhuizen. Met een MRI-scanner kan een afbeelding worden gemaakt met behulp van magnetische resonantie. Aandoeningen in het lichaam kunnen zichtbaar worden gemaakt voor medici, zonder dat daarvoor het lichaam opengemaakt hoeft te worden. Commerciële MRI-scanners hebben statische magnetische veldsterktes variërend tussen de 0,5 T en 7 T. Ter vergelijking: een koelkastmagneet heeft een magnetische veldsterkte die kleiner is dan 0,1 T. Een 7 T MRI-scanner heeft dus een 70 keer zo sterke aantrekkingskracht op een ferromagnetisch materiaal.

Op het gebied van MRI zijn de ontwikkelingen nog in volle gang. Heden wordt er onderzoek gedaan naar MRI-scanners op hogere magnetische veldsterktes. Naarmate de veldsterkte hoger wordt, kan de resolutie van de afbeelding steeds groter worden. Op de afbeelding zijn meer details te zien en er kan een betere diagnose worden gesteld.

Bij cardiac MRI wordt een afbeelding gemaakt van het hart. Het is hierbij van belang dat een scan telkens op dezelfde fase in de hartcyclus wordt gemaakt. De scan moet dus gesynchroniseerd worden met de hartslag. Een veel gebruikte methode hiervoor is de electrocardiographic (ECG) triggeringstechniek. Niendorf et al hebben in hun studie [1] laten zien dat ECG-methoden negatief beïnvloed worden door interferentie van elektromagnetische velden en magneto-hydrodynamische (MHD) effecten. Het concept achter MHD is dat onder invloed van magnetisme stromen, geïnduceerd worden in geleidende vloeistoffen. Deze effecten worden sterker bij een toenemende magnetische veldsterkte.

De ECG wordt bij 7 T zodanig beïnvloed, dat de elementen in de ECG die benodigd zijn voor triggering niet meer eenduidig uit het signaal gehaald kunnen worden. De triggering vindt hierdoor plaats op verschillende tijdstippen in de hartcyclus, een ongewenst effect. Scans van het hart die zijn genomen op een verkeerd moment zorgen voor artefacten, waardoor minder scherpe afbeeldingen ontstaan. Een recente studie [2] heeft aangetoond dat er triggeringmethodes zijn die minder onderhevig zijn aan de effecten van sterke magnetische velden, waardoor de afbeelding van een betere kwaliteit kan zijn in sterke magnetische velden.

Het Leids Universitair Medisch Centrum (LUMC) kampt al jaren met het probleem dat er geen scherpe afbeeldingen van het hart gemaakt kunnen worden. Daarom heeft het de opdracht gegeven om een draadloos sensorsysteem te ontwikkelen dat beter geschikt is om scherpe scans te maken in een 7 T MRI-scanner. Deze opdracht is opgedeeld in drie onderdelen:

- acquisitie van een biomedisch signaal waarop cardiac triggering kan plaatsvinden;
- omzetting van het analoge sensorsignaal in een digitaal signaal en onderdrukken van de ruis;
- transmissie van het digitale signaal vanaf de sensor op de patiënt in de MRI-scanner naar een computer en aanbieden van het triggersignaal.

Aan elk deelprobleem is een ander team gekoppeld. In deze thesis wordt onderzocht wat de beste methode is om draadloos een sensorsignaal te versturen in een MRI-scanneromgeving met hoge magnetische veldsterkte

en een sterk radiofrequent (RF) veld op 298 MHz. Een draadloos systeem heeft vele voordelen boven een systeem dat gebruikt maakt van een kabel. Shellock en Kanal hebben aangetoond dat gebruik van geleidende bekabelde apparatuur in een MRI-omgeving kan leiden tot ernstige brandwonden bij de patiënt [3]. Daarnaast heeft de opdrachtgever kenbaar gemaakt een draadloos systeem te prefereren zodat de sensormodule een grotere mobiliteit heeft.

Tijdens het ontwerp worden verschillende draadloze transmissiesystemen tegen elkaar afgewogen. Daarnaast wordt er onderzocht welke vereisten er zijn om de transmissie op een betrouwbare manier te laten lopen.

In deze thesis wordt bovendien onderzocht hoe het ontvangen sensorsignaal op een duidelijk en eenvoudige manier kan worden weergegeven op een computerscherm. Hierbij wordt een afweging gemaakt tussen populaire programmeer- en scripttalen.

Daarnaast wordt een vast punt in de hartcyclus gedetecteerd op basis van het sensorsignaal. De MRI-scanner kan aan de hand van dit vaste punt de scan beginnen.

De thesis is opgebouwd uit verschillende hoofdstukken. In hoofdstuk 3 zal ingaan worden op de eisen waaraan het systeem moet voldoen. Deze bestaan uit eisen van het LUMC en eisen die door de andere twee teams, die bezig zijn met het MRI-project, worden gesteld. Hoofdstuk 4 bespreekt de werking van een MRI-scanner, welke fysische verschijnselen daarbij optreden en de mogelijke draadloze communicatiemethoden. Hoofdstuk 5 gaat in op de keuze die gemaakt is voor de hardware en software. De keuze zal gemaakt worden op basis van de eisen in hoofdstuk 3. Hoofdstuk 6 bespreekt vervolgens de topologie van het systeem. De implementatie van het systeem wordt besproken in hoofdstuk 7. In hoofdstuk 8 worden de testresultaten van het prototype beschreven. Tot slot staan in hoofdstuk 9 de conclusie en de aanbevelingen.

Hoofdstuk 2

Literatuurstudie

Dit hoofdstuk beschrijft de literatuurstudie die is gedaan naar de onderwerpen die relevant zijn voor deze thesis. Ieder hoofdstuk bespreekt een deelonderwerp. In paragraaf 2.1 worden de effecten op apparatuur en metingen in een MRI-omgeving onderzocht. Vervolgens wordt in paragraaf 2.2 onderzocht welke draadloze communicatietechnieken bestaan voor communicatie op korte afstand. In paragraaf 2.3 wordt onderzocht welke manieren er bestaan om biomedische hartsignalen te analyseren.

2.1 Effecten op apparatuur in een MRI-omgeving

Het LUMC kampt met het probleem dat afbeeldingen van het hart in een 7 T scanner niet scherp genoeg zijn. Gupta onderkent dit probleem al in 2008: onder invloed van een sterk magnetisch veld gaan geladen deeltjes zich anders gedragen [4]. De deeltjes ondervinden een kracht die loodrecht staat op de baan en loodrecht op het magnetisch veld, de Lorentzkracht. Hierdoor ontstaat een potentiaalverschil aan twee zijden van een geleider. Dit wordt de Hallspanning genoemd. Met behulp van de Magnetofluid Dynamics (MFD) vergelijkingen kunnen de effecten in de vaten van het hart waardoor het bloed stroomt zichtbaar worden. Omdat er in bloed ook geleidende deeltjes zitten, wordt een ECG-meting ernstig verstoord door deze effecten.

Deze verstoringen waren reden genoeg voor Frauenrath om in 2010 andere manieren voor triggering te onderzoeken [2]. Zo blijkt dat met behulp van Acoustic Cardiac Triggering (ACT) grote verbeteringen in de scankwaliteit gehaald kunnen worden. ACT is vrij van verstoringen door MHD effecten en interferentie van elektromagnetische velden. De kwaliteit van de afbeelding is bij ECG-triggering slechts 1.7, terwijl de afbeeldingen die gemaakt zijn met ACT een score van 2.4 noteren op een schaal van 0 tot 3.

Het belangrijkste wat Frauenrath concludeerde was dat jitter, de verschuiving van het moment waarop de afbeelding wordt genomen, de grootste invloed heeft op de kwaliteit van de afbeeldingen. De standaarddeviatie van de jitter moet daarom zo klein mogelijk worden. Bij het ontwerp van het systeem voor deze thesis moet rekening worden gehouden dat de jitter in ieder geval niet groter mag zijn dan 60 ms, de maximale waarde om nog een redelijke afbeelding te krijgen. De minimale jitter die gehaald is met ACT is 5 ms.

Het sterke magnetische veld kan op kleinere schaal ook invloed hebben. Al in 1969 was bekend dat sterke magnetische velden ook invloed hebben op de instellingen van transistoren [5]. Transistoren gemaakt van silicium zijn minder onderhevig dan transistoren van germanium. Omdat het artikel enigszins gedateerd is (1969), is het de vraag hoe relevant dit artikel is voor hedendaagse transistors. De grootte van transistors is immers flink afgenomen sinds die tijd.

2.2 Draadloze communicatie op korte afstand

2.2.1 Akoestische communicatie

Akoestische communicatie is communicatie door middel van geluidsgolven. Geluid kan binnen of buiten het menselijk hoorgebied liggen, dat is afhankelijk van de frequentie van het geluid. Geluiden hoger dan 20 KHz worden ultrasoon genoemd. Vleermuizen maken hier bijvoorbeeld gebruik van om te navigeren.

In de literatuur zijn een aantal onderzoeken te vinden waarin een akoestisch communicatiesysteem is opgezet. In 2000 hebben Lopes en Aguiar een systeem gebouwd dat berichten in muziek kan verweven [6]. Zhang et al hebben met zeer eenvoudige hardware, zoals een buzzer en een microfoon, die vaak aanwezig is op sensorbordjes, een akoestische transmissielink opgezet die kan dienen als backup kanaal voor traditionele RF communicatie [7]. Hoorbaar geluid kan handig zijn als mensen op de hoogte moeten zijn van het feit dat er gecommuniceerd wordt. Het grootste nadeel is de beperkte bandbreedte die hiervoor gebruikt kan worden. De maximale datasnelheid die behaald kan worden is 10 kbit/s. Het tweede nadeel is dat de MRI-scanner zelf ook erg veel geluid produceert en storing op kan leveren voor het communicatiesysteem. Het lawaai van een MRI-scanner kan oplopen tot wel 110 dB SPL.

In 2008 maakten Li et al een draadloze verbinding met ultrasone geluiden werd ontworpen. Met eenvoudige hardware, zoals membranen en piëzo-elektrische componenten, kan een verbinding worden gemaakt. Deze componenten zijn erg goedkoop en commercieel beschikbaar. Het grote nadeel van ultrasoon geluid, ontworpen met goedkope componenten, is dat de Bit Error Rate (BER) zeer hoog is. Op een afstand van 2.8 m is deze al 10% en neemt daarna zeer sterk toe met de afstand. Hierdoor lijkt deze techniek niet geschikt voor onze thesis.

2.2.2 Infraroodlicht

Infraroodcommunicatie maakt gebruik van elektromagnetische golven. Dit licht is niet zichtbaar voor het menselijk oog. In 1997 legde Kahn en Barry een brede theoretische basis voor draadloze communicatie met infraroodlicht [8]. In 2001 is een vergelijking getrokken tussen het protocol AIr (Advanced infrared) en de veel voorkomende WLAN [9]. Hieruit bleek dat er zeer grote datasnelheden bereikt kunnen worden met Infrarood (IR), tot aan 4 Mb/s. Deze datasnelheden worden echter alleen gegarandeerd voor een afstand tot 1 m.

Infraroodlicht kan niet door muren en niet-transparante objecten penetreren. Hierdoor moeten zender en ontvanger op elkaar gericht zijn, zonder tussenkomst van andere objecten. De situatie in een MRI-scanner maakt het lastig om dit te realiseren. De patiënt ligt in een kleine cilindrische opening, waardoor de zender die verbonden is aan de hartslagsensor niet de ontvanger kan bereiken aan de rand van de ruimte.

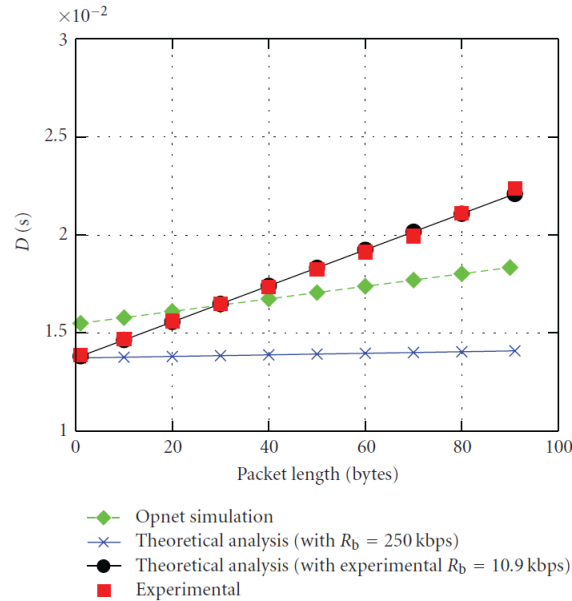
2.2.3 Radiofrequente golven

Om tot een keuze te komen voor de beste techniek voor communicatie met radiogolven, moeten een aantal zaken worden afgewogen. Zo zal een grotere datasnelheid vaak een grotere bandbreedte vereisen, maar ook zorgen voor een kortere transmissietijd. In 2007 is een studie gedaan naar de verschillende digitale protocollen voor RF (radiofrequente) communicatie [10]. De drie voornaamste protocollen voor Wireless Personal Area Network (WPAN) worden naast elkaar gelegd, Bluetooth, UWB en ZigBee. Deze zijn geschikt voor draadloze verbindingen tot 10 meter. De grootste verschillen tussen deze drie zijn de maximale datasnelheid en het nominale vermogen.

De datasnelheden die gegeven worden in een datasheet of die gegarandeerd worden door het protocol zijn vaak theoretisch. In de praktijk worden vaak lagere datasnelheden behaald. In 2007 heeft Ferrari daarom onderzocht wat nu de maximaal behaalde snelheid is voor een ZigBee verbinding [11]. Deze bleek veel lager dan de opgegeven maximale snelheid van 250 kbit/s, namelijk 32 kbit/s.

Omdat de transmissietijd mogelijk invloed kan hebben op de jitter van de MRI-scanner, zie paragraaf 2.3, moet deze ook mee worden genomen in de afweging tussen verschillende transmissie protocollen. De

transmissietijd blijkt groter te worden naarmate de lengte van de datapakketjes groter wordt. Uit figuur 2.1 kan worden afgeleid dat deze bij een grootte van 10 bytes al 15 ms bedraagt.



Figuur 2.1: Transmissietijd ten opzichte van grootte datapakketjes ZigBee, simulatie en praktijk [11]

In 2009 is een nieuw draadloos protocol gestandaardiseerd door de IEEE: RuBee. RuBee maakt gebruik van magnetische golven met grote golflengte. Mede omdat de frequentie zo laag is, kan het vermogensverbruik van de apparaten die met RuBee werken zeer klein blijven, in de orde van enkele tientallen microwatts. De apparaten kunnen gevoed worden door een batterij, maar het grote voordeel van RuBee is dat ze ook gevoed kunnen worden door een draaggolf. De maximale datasnelheid die wordt gespecificeerd is echter wel laag, slechts 9.6 kbit/s. Indien de voeding voor de apparatuur een probleem is kan deze techniek dus gebruikt worden, mits de hoeveelheid data die verstuurd moet worden niet te groot is.

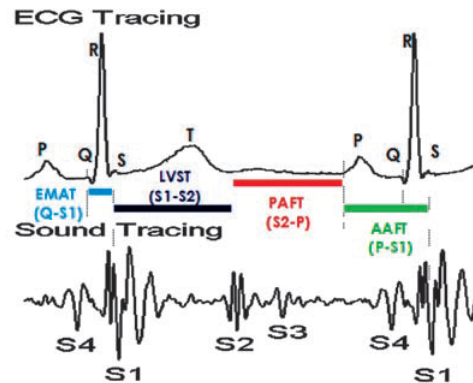
De noodzaak om biomedische signalen van een patiënt in de gaten te houden is niet nieuw. Een onderzoeksgroep heeft zeer recentelijk een sensor gebouwd die draadloos de ECG van alle patiënten in een ziekenhuis af kan lezen [12]. De groep gebruikt hier ZigBee modules voor. Veel elementen die in de thesis nodig zijn, zoals het weergeven van de ECG-data met behulp van computersoftware, worden door deze groep ook gebruikt.

2.3 Analyse van het hartsignaal voor triggering

Een arts kan met een stethoscoop, een medisch instrument, luisteren naar het hart en de longen. Hiermee kan hij afleiden wat de hartslag is van een persoon en kan hij vaststellen of het hart op een goede manier functioneert. Harttonen zijn de geluiden die het hart maakt. De twee hoofdtonen worden S_1 en S_2 genoemd en worden veroorzaakt door het sluiten van de kleppen in het hart.

Deze hartgeluiden kunnen opgenomen worden met een microfoon en weergegeven worden in een diagram. Dit wordt in de biomedische literatuur een phonocardiogram (PCG) genoemd. Uit een analyse van een PCG kan de hoofdtoon S_1 worden afgeleid.

De aansturing van de MRI-scanner, het triggeren van de scan, moet telkens op hetzelfde moment in de hartcyclus gebeuren. Conventioneel wordt het ECG gebruikt om een MRI-scanner aan te sturen. Voor medici aan het LUMC is het ECG daarom de gouden standaard. De R-wave, afgebeeld in figuur 2.2, wordt



Figuur 2.2: ECG en akoestische meting van het hartsignaal [13]

gebruikt als oriëntatiepunt om de scan te nemen. Omdat het tijdsverschil tussen de S_1 harttoon en de R-wave minimaal is[2], kan de S_1 harttoon ook als referentiepunt genomen worden. Om scherpe afbeeldingen te maken is het alleen van belang dat de jitter, de verschuiving in de tijd ten opzichte van het referentiepunt, niet te groot is.

De detectie van de harttonen kan met verschillende technieken gedaan worden. In het jaar 2000 lieten Djebbari en Reguig zien dat het PCG signaal geanalyseerd kan worden met de Short-Time Fourier Transform (STFT) [14]. Omdat het PCG-signaal niet stationair is, wordt het onderverdeeld in kleine, stationaire delen en elk deel wordt naar het frequentie domein omgezet met de Fouriertransformatie. Doordat het frequentiespectrum van harttoon S_1 verschilt van de andere harttonen, kan het tijdstip waarin S_1 voorkomt uit het signaal worden gehaald. Hiermee kan dus de MRI-scanner aangestuurd worden.

In 2007 liet Nazeran zien dat met behulp van een Wavelet transformatie een betere analyse gedaan kan worden van een PCG [15]. Bij STFT moet een afweging worden gemaakt tussen de grootte van stukjes waarin je het signaal opdeelt en de resolutie van de frequentie. Met de Wavelet transformatie bestaat dit probleem niet. Deze methode is dus beter geschikt om hartgeluiden te segmenteren.

Om te meten wat de hartslag van een patiënt is aan de hand van een PCG, hoeft niet altijd een ingewikkelde frequentiemeting, zoals hierboven beschreven staat, gedaan worden. Met behulp van een comparator kan bepaald worden wanneer het signaal boven een bepaalde drempelwaarde uitkomt. Omdat de amplitude van het PCG signaal het grootst is gedurende harttoon S_1 , kan de comparator ingesteld worden op een niveau waarbij S_1 gedetecteerd kan worden.

In 2002 werd door Lee and Moghavvemi al een systeem gebouwd dat op deze manier de hartslag van een persoon kan meten [16]. Het grootste probleem hiervan is dat het PCG signaal per persoon flink kan verschillen en dat het meetsysteem gevoelig is voor piekverstoringen. Een dergelijke detectiemethode kan analoog en digitaal eenvoudig gerealiseerd worden.

Hoofdstuk 3

Programma van eisen

In dit hoofdstuk wordt duidelijk gemaakt aan welke eisen het product moet voldoen. De eisen zijn ingedeeld in een aantal onderdelen. Eisen vanuit het beoogde gebruik (paragraaf 3.1) laat zien welke eisen er zijn vanuit de opdrachtgever. In eisen vanuit de ecologische situering (paragraaf 3.2) wordt er verteld aan welke veiligheidseisen het systeem moet voldoen. In paragraaf 3.3 wordt duidelijk gemaakt aan welke eisen het systeem moet voldoen om in het totale eindproduct te passen. Eisen met betrekking tot het eventueel te ontwikkelen productiesysteem (paragraaf 3.4) laat zien hoe het product gefabriceerd wordt en wat de eisen zijn om tot een goed product te komen. In eisen met betrekking tot het te ontwikkelen liquidatiesysteem (paragraaf 3.5) wordt uitgelegd dat er geen liquidatiesysteem nodig is om dit product goed te verwijderen. In 3.6 wordt uitgelegd aan welke eisen het systeem moet voldoen met betrekking tot de levering en het onderhoud. In de randvoorwaarden (paragraaf 3.7) wordt uitgelegd aan welke voorwaarden het transmissiesysteem moet voldoen om goed aan te sluiten op andere teams in het bacheloreindproject.

3.1 Eisen vanuit het beoogde gebruik

- 3.1.1 De afmetingen van de behuizing van het systeem dat aangesloten is op de sensor bedragen maximaal 10 x 10 x 5 cm (lengte, breedte, hoogte).
- 3.1.2 Het totale gewicht van de behuizing van de sensor en componenten bedraagt maximaal 0,5 kg.
- 3.1.3 De accu van het systeem gaat gedurende actieve werking minstens 4 uur mee.
- 3.1.4 De temperatuur van de behuizing van de sensor mag niet boven de 25 graden Celcius uitkomen.
- 3.1.5 De draadloze transmissie moet kunnen plaatsvinden over minimaal 10 meter.
- 3.1.6 De MRI-scanner moet getriggerd worden op een vast punt in de hartcyclus. De verschuiving ten opzichte van dit punt (jitter) mag niet meer bedragen dan 60 ms.
- 3.1.7 De sensorbehuizing mag niet sterk aangetrokken worden door de MRI-scanner, de componenten bevatten hiertoe een minimale hoeveelheid aan ferromagnetische materialen;
- 3.1.8 De transmissiemethode mag geen storing veroorzaken op de ziekenhuisapparatuur.
- 3.1.9 Het systeem moet robuust zijn tegen de volgende vormen van interferentie: radiofrequente golven van 298 MHz, het statische magnetische veld van de MRI-scanner, laagfrequente magnetisch veld en geluidsgolven (veroorzaakt door gradiëntmagneten).
- 3.1.10 Op de frequentieband waarin de transmissie plaatsvindt berust geen licentie.

3.2 Eisen vanuit de ecologische situering van het systeem in de omgeving

- 3.2.1 De componenten mogen geen straling uitzenden die gevaarlijk is voor de patiënten en het personeel.
- 3.2.2 Het systeem moet voldoen aan veiligheidseisen die gesteld zijn in de reglementen van het ACR en CE.
- 3.2.3 Alle componenten die gebruikt worden moeten voldoen aan de voorwaarden van de RoHS-richtlijn.

3.3 Eisen met betrekking tot het ontwerpen van het systeem

Gebruikskenmerken

- 3.3.1 Het product moet *plug and play* zijn waardoor bij een defect direct een vervangend model kan worden opgestuurd.
- 3.3.2 De kosten van het systeem mogen niet meer dan €100 zijn.

Productie- en ingebruikstellingskenmerken

- 3.3.3 Het transmissiesysteem moet gemakkelijk aan te sluiten zijn op de producten van de andere teams.
- 3.3.4 Het signaal uit de sensor moet met een vaste vertraging worden verzonden naar de computer. De vertraging mag niet meer bedragen dan 100 ms, zodat het signaal *real-time* weergegeven kan worden.
- 3.3.5 Verschuivingen van de patiënt in de MRI-scanner mogen geen belemmerend effect hebben op de transmissie. De patiënt hoeft niet doodstil te liggen.
- 3.3.6 Het uitgangssignaal van het totale systeem is een TTL signaal, waarop de MRI-scanner getriggerd kan worden.
- 3.3.7 De ontvanger kan verbonden worden met een PC door middel van een USB-kabel met een lengte van 3 meter.

Liquidatiekenmerken

- 3.3.8 Het systeem moet uit componenten bestaan die voor 90% te recyclen zijn zodat het geen schadelijk effecten heeft voor het milieu.

3.4 Eisen met betrekking tot het eventueel te ontwikkelen productiesysteem

Bij alle technieken is het het geval dat het eindproduct bestaat uit modules die bij een derde partij besteld moeten worden en dan nog geprogrammeerd en op elkaar aangesloten moeten worden. De enige eis die er dus is met het betrekking tot het productiesysteem is:

- 3.4.1 Dat het systeem de bij een derde partij bestelde modules moeten kunnen programmeren.

3.5 Eisen met betrekking tot het te ontwikkelen liquidatiesysteem

- 3.5.1 Alle componenten kunnen bij het gewone afval worden weggegooid. Alleen de accu moet worden gedeponeerd bij het klein chemisch afval. Er zijn dus geen verdere vereisten aan het liquidatiesysteem.

3.6 Eisen vanuit bedrijfsstrategische, marketing- en verkooptechnische omstandigheden

3.6.1 Het gehele systeem moet binnen een maand geleverd kunnen worden.

3.7 Randvoorwaarden

De kenmerken in deze sectie zijn opgedragen door de signaalbewerkingsgroep. Dit zijn dus harde eisen om ervoor te zorgen dat de twee systemen op elkaar aangesloten kunnen worden

3.7.1 Het binnenkomende digitale signaal heeft een datasnelheid van ten hoogste 120 kbit/s.

3.7.2 Het binnenkomende digitale signaal is gecodeerd met unipolar non-return-to-zero (NRZ) codering (high 3.3V, low 0V).

3.7.3 Het temperatuursignaal, akoestisch signaal en het signaal van hartslag worden aangeboden via een Serial Peripheral Interface Bus (SPI) poort. Het formaat waar de data aan moet voldoen is beschreven in paragraaf 7.1.2.

Hoofdstuk 4

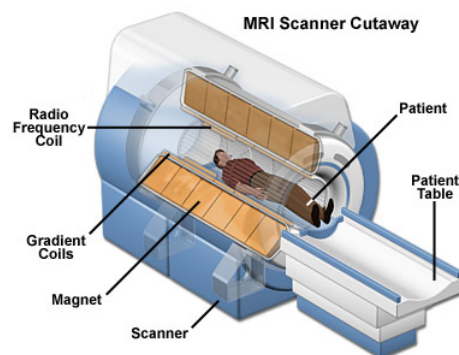
Communicatiesysteem in een MRI-scanner

In dit hoofdstuk worden de concepten gepresenteerd van een draadloos communicatiesysteem. In paragraaf 4.1 wordt beschreven hoe een MRI-scanner is opgebouwd. Paragraaf 4.2 beschrijft de fysische principes van een MRI-scanner. In paragraaf 4.3 wordt uitgelegd hoe deze fysische effecten invloed kunnen hebben op het communicatiesysteem. Daarna worden de verschillende protocollen op het gebied van draadloze communicatie onderzocht. Eerst wordt voor ieder protocol uitgelegd op welke fysische werking deze gestoeld is. Daarna worden de verschillende internationale standaarden die voor het protocol bestaan besproken. In hoofdstuk 5 zal vervolgens een keuze worden gemaakt voor het protocol en de gebruikte technieken.

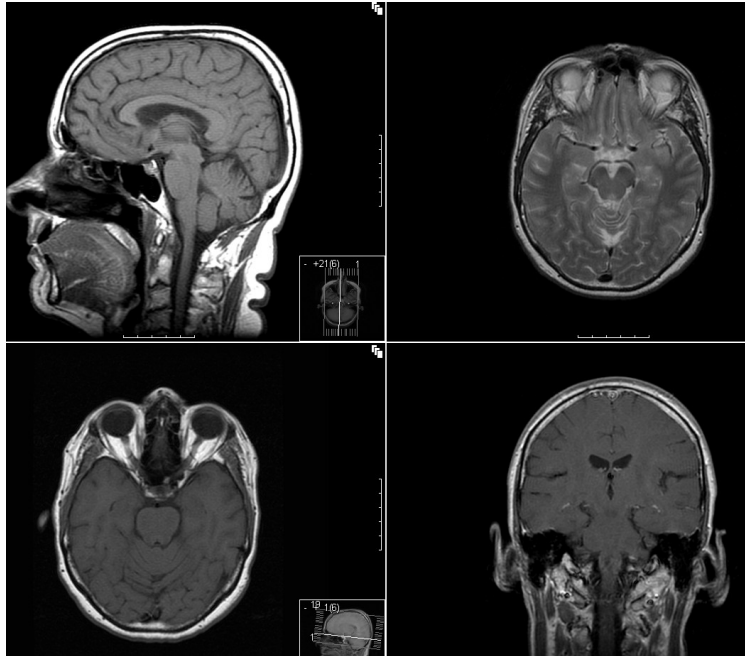
4.1 Hardware MRI

Een MRI-scanner bestaat uit een aantal standaard onderdelen: een magneet voor het sterke magnetische veld, een spoel die een gradiëntveld creëert en een spoel die de RF signalen verzendt en opvangt. In figuur 4.1 worden deze onderdelen weergegeven.

Er kunnen verschillende soorten magneten worden gebruikt in de scanner. Zo bestaan scanners met permanente magneten, maar ook met elektromagnetten en supergeleidende magneten. De magneet zorgt voor een uniform magnetisch veld in de cilindrische opening van de scanner. De sterkte van dit veld is afhankelijk van het type scanner. De scanner waarop het eindsysteem getest gaat worden heeft een veldsterkte van 7 T. Het LUMC gebruikt deze scanner voor medische toepassingen en voor onderzoek.



Figuur 4.1: Hardware van een MRI-scanner [17]



Figuur 4.2: Doorsnede van het hoofd gemaakt door een MRI-scanner [18]

Een ander essentieel onderdeel in de scanner is een kleinere elektromagnetische spoel die een magnetisch gradiëntveld genereert. Deze wordt beschreven in paragraaf 4.2. Daarnaast zijn er spoelen die zorgen voor de radiofrequente transmissie en ontvangst van de afbeeldingen.

De sterke magneten bieden gevaar voor de apparatuur in de nabijheid van de scanner. De ruimte van de scanner is daarom afgeschermd met magnetisch schild. Daarnaast is een kooi van Faraday om de ruimte gebouwd. Deze kooi zorgt er voor dat apparaten buiten de MRI-ruimte geen bronnen van storing vormen en invloed zouden kunnen hebben op de afbeeldingen.

4.2 Werkingsprincipe MRI

In het menselijk lichaam is waterstof het meest voorkomende atoom. Waterstof protonen zijn onderhevig aan het statische magneetveld B_0 . Ze richten zich naar de oriëntatie van het magneetveld B_0 . Als er een RF pulse met de resonantiefrequentie van het proton ω_0 uitgezonden wordt door de spoelen, zal de energietoestand van het proton veranderen naar een hoger niveau. De frequentie waarbij dit gebeurt heet de Larmorfrequentie:

$$\omega_0 = \gamma \cdot B_0 \quad (4.1)$$

waar γ de gyromagnetische verhouding is, een constante waarde die afhankelijk is van het atoomnummer, en B_0 de sterkte van het magneetveld. Op het moment dat er geen radiofrequente pulsen meer uitgezonden worden, veranderen de protonen weer naar hun lagere energie niveau. De protonen zenden hierbij radiofrequente golven uit met dezelfde frequentie ω_0 . Voor een 7 T scanner is deze frequentie 298 MHz.

Met het magnetische gradiëntveld kan de magnetische veldsterkte plaatselijk worden veranderd. De veldsterkte is voor verschillende protonen dus hoger of lager, afhankelijk van de positie van het proton. Uit formule 4.1 volgt dan dat ook de Larmorfrequentie van de protonen verschillend is.

Met behulp van een computer kunnen de binnengekomen RF signalen worden geanalyseerd. De frequentie van het signaal bepaalt dan op welke positie een proton straling heeft uitgezonden.

Uiteindelijk krijgt de arts een afbeelding te zien zoals in figuur 4.2. Hier is een scan gemaakt van de hersenen.

4.3 Effecten op transmissie

Uit het bovenstaande blijkt dat er twee bronnen zijn die grote invloed kunnen hebben op de communicatie.

De eerste bron is het statische magneetveld B_0 . Indien het sensorsysteem in de MRI-scanner is geplaatst, zal het systeem onderhevig zijn aan het magneetveld. Allereerst worden ferromagnetische materialen aangetrokken door de magneet. Een demonstratie met een ijzeren schaar in een MRI-scanner liet zien dat het object de scanner in werd getrokken en heftig ging bewegen in de scanner. De krachten die op ferromagnetische objecten werken zijn enorm.

Het statische magneetveld heeft ook invloed op de instellingen van transistoren [5]. Bewegende ladingen in transistoren ondervinden een Lorentzkracht:

$$\vec{F}_L = q \cdot \vec{v}_d \times \vec{B} \quad (4.2)$$

Doordat negatieve en positieve ladingsdragers elk naar een andere kant in een geleider worden getrokken door de Lorentzkracht, ontstaat er een potentiaalverschil. Dit wordt de Hallspanning genoemd. Hudson heeft laten zien dat het Hall-effect een kortere kanaallengte van een transistor veroorzaakt [5]. Hierdoor wordt de versterkingsfactor H_{fe} verminderd. Aangezien er veel geïntegreerde chips op een communicatiemodule zitten, die op hun beurt weer veel transistoren bevatten, kan de verandering van de versterkingsfactor een uitwerking hebben op de functionaliteit van de module.

Naast het statische magneetveld wordt er in de MRI-scanner ook gebruik gemaakt van RF golven. Indien er gebruik wordt gemaakt van een radiofrequent transmissiesysteem kan er interferentie optreden. Enerzijds mag het transmissiesysteem niet de werking van de MRI-scanner beïnvloeden, anderzijds zullen de RF golven het transmissiesysteem verstoren indien de twee systemen op dezelfde frequentieband werken.

4.4 Communicatievormen

Op het gebied van draadloze communicatie bestaan verschillende protocollen. Voor communicatie op afstanden kleiner dan 100 m wordt de techniek die berust op radiogolven veel gebruikt. Door de storingsbronnen die aanwezig zijn in de MRI-scanner, met name de RF golven die nodig zijn voor het maken van de afbeelding, is het wenselijk om ook andere communicatievormen te overwegen.

Veel gebruikte alternatieve technieken zijn infrarood communicatie en in mindere mate akoestische communicatie.

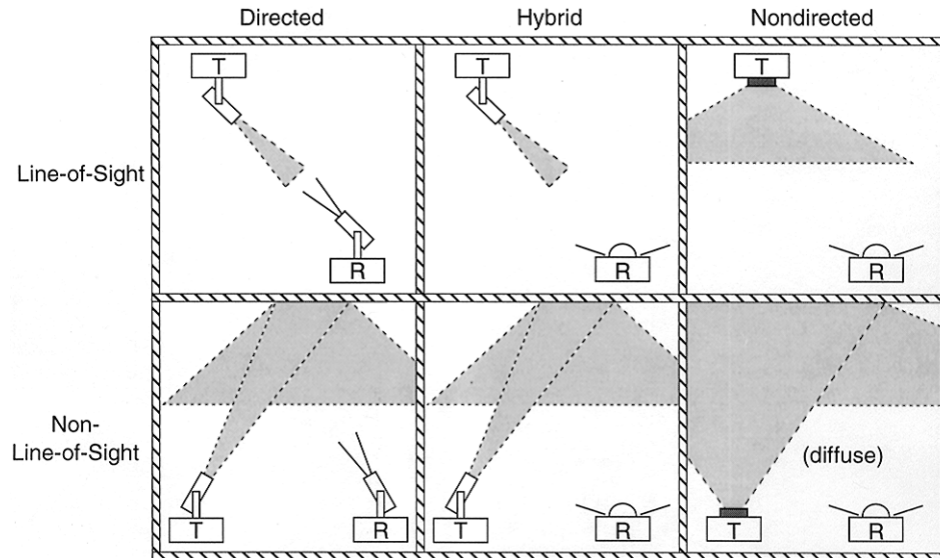
De volgende drie vormen van communicatie worden onderscheiden:

1. Optische communicatie
2. Akoestische communicatie
3. Radiofrequente of microwave communicatie

4.4.1 Optische communicatie

Een veel voorkomende methode voor communicatie is gebaseerd op infrarood licht. Deze techniek wordt veel toegepast in afstandsbedieningen, laptops en andere mobiele applicaties die communicatie op korte afstand vereisen. Infrarood is voor het oog niet zichtbare elektromagnetische straling met een golflengte tussen 780 nm en 1 μm .

Twee classificaties zijn mogelijk binnen infrarood systemen. Eerste classificatie vindt plaats op grond van de mate waarin de zender en ontvanger op elkaar gericht moeten zijn. *Directed links* vereisen dat zender en ontvanger op elkaar gericht zijn en *non-directed links* hoeven dat niet, zoals is weergegeven in figuur 4.3. De



Figuur 4.3: Uitleg over (non-)Line of sight en (non-)directed, waarbij T staat voor transmitter (verzender) en R staat voor receiver (ontvanger) [8].

Tabel 4.1: Vergelijking IrDA-standaard met remote control module (Ozitrionics)

	IrDA[19]	Ozitrionics[20]
Afstand	0-1m	Max 15m
Max datasnelheid	115,2 kbit/s	115,2kbit/s
Verzendvermogen	3,5 mW	4,5 mW
Classificatie	Non-directed LOS	Non-directed LOS

vermogen efficiëntie is bij de eerstgenoemde veel hoger, aangezien de *path losses* kleiner zijn. Daarnaast kan er onderscheid gemaakt worden op het gebied van Line Of Sight (LOS). Bij *LOS links* moet er een direct pad zonder obstakels bestaan, waar dat bij *non-LOS links* niet noodzakelijk is [8].

De voornaamste bron van ruis voor infrarood links is omgevingslicht. Vormen van omgevingslicht zijn zonlicht of kunstmatig licht, bijvoorbeeld gloeilampen of tl-buizen. In deze lichtbronnen zijn dezelfde frequentiecomponenten aanwezig als in infrarood licht [8].

De Infrared Data Association (IrDA) heeft een standaard ontwikkeld op het gebied van LOS infrarood communicatie. De datasnelheden waaraan de links moeten voldoen lopen van 115.2 kbit/s tot 4 MB/s [19]. De *Bit error rate (BER)* moet hierbij minder dan 10^{-9} bedragen en de afstand moet minimaal 1 m zijn. Voor de ontvangers geldt dat ze een Field of View (FOV) moeten hebben van minimaal 15° .

In tabel 4.1 staan de specificaties van twee modules die gebruikt zouden kunnen worden voor transmissie met behulp van infrarood.

4.4.2 Akoestische communicatie

Ultrasone communicatietechniek wordt veel gebruikt voor onderzeecommunicatie. Het grootste nadeel van ultrasone technieken in lucht is dat de intensiteit exponentieel afneemt met de afstand:

$$P_d = P_0 \cdot e^{-\frac{a \cdot d}{2}} \quad (4.3)$$

waar P_d de druk is op een afstand d van geluidsbron P_0 en a de absorbtiecoëfficiënt.

Als gevolg hiervan geldt dat de *Bit error rate (BER)* erg sterk toeneemt met de afstand [21]. De BER is een

begrip dat de kans beschrijft dat een bitfout in de transmissie optreedt. Zo is de BER bij BPSK modulatie voor akoestische transmissie op een afstand van 3 meter al 0,5. Voor afstanden van meer dan 3 meter is deze techniek dus niet geschikt. In hetzelfde rapport is aangetoond dat met de modulatietechniek QPSK ultrasone signalen verzonden kunnen worden over een reikwijdte van 1 tot 2 meter [22].

Tabel 4.2: Specificaties akoestische communicatie

	Acoustic DBPSK [23]
Frequentie	10 kHz
Bereik	3 m
Maximale datasnelheid	5 kb/s
Gevoeligheid positie	Gevoelig voor verschuiving door multi-path probleem

4.4.3 Radiofrequente communicatie

Met radiofrequente communicatie wordt communicatie via elektromagnetische golven met een frequentie tussen de 50 kHz en 5 GHz bedoeld. De free space gain van radiofrequente communicatie wordt gegeven door de volgende formule:

$$G_{fs} = \left(\frac{\lambda}{4 \cdot \pi \cdot d}\right)^2 \quad (4.4)$$

Hierin is G_{fs} de versterkingsfactor van het signaal, d de afstand vanaf de zender en λ de golflengte. Het vermogen neemt dus kwadratisch af met de afstand. In sommige gevallen zal een direct pad bestaan tussen zender en ontvanger, maar bij indoor situaties kan dit directe pad geblokkeerd worden door muren of objecten. Een signaal dat door een muur of een object zoals een MRI-scanner propageert, verliest al snel meer energie dan in het geval waarin er een vrije zichtlijn bestaat [24]. De theoretische maximale afstand die opgegeven wordt in datasheets zal voor communicatie in een MRI-omgeving dus niet altijd opgaan. Voor RF communicatie zijn vele standaarden ontwikkeld, met elk hun eigen toepassingen. Hieronder zullen alleen ZigBee, Wi-Fi en Bluetooth worden besproken aangezien Rubee, UWB en Z-wave bij voorbaat al niet geschikt zijn om te voldoen aan het programma van eisen.

ZigBee

ZigBee is ontwikkeld als een aanvulling op de bestaande technieken Bluetooth en Wi-Fi. Vooral vanuit de industrie was er veel vraag naar draadloze verbindingen voor sensor- en controlenetwerken [12]. Omdat vele ZigBee modules samen kunnen werken, kan er een robuust netwerk van sensoren gebouwd worden. Ook als er slechts een punt-naar-puntverbinding nodig is, wordt ZigBee vaak ingezet om een betrouwbare verbinding op te stellen. Door het extreem lage vermogensverbruik (≤ 20 dBm) werken ZigBee sensors vaak jaren op een enkele batterij, ideaal voor binnenshuis toepassingen zoals lichtschakelaars.

ZigBee is gebaseerd op de IEEE 802.15.4 standaard. In deze standaard worden de uitwerking van de fysieke laag en de MAC laag van het OSI-model gedefinieerd. De fysieke laag staat twee frequentiebanden toe in Europa: de 915 MHz band en de 2.4 GHz band. De laatste frequentieband laat een hogere datasnelheid toe van maximaal 250 kbit/s. ZigBee gebruikt Direct Sequence Spread Spectrum (DSSS) met 16 kanalen, en elk kanaal heeft een bandbreedte van 2 MHz.

Wi-Fi

Wi-Fi is de benaming voor draadloze producten die gebaseerd zijn op de IEEE 802.11 standaard. Het bereik van Wi-Fi kan oplopen tot 100 m. Mede daardoor wordt de techniek vaak ingezet in thuisnetwerken, en publiek toegankelijke locaties zoals vliegvelden, restaurants, universiteiten en bibliotheken. IEEE 802.11.g, de meest gebruikte standaard, definieert een maximale datasnelheid van 54 Mb/s. De nieuwste standaard, IEEE 802.11n, heeft een maximale datasnelheid van 600 Mb/s.

Bluetooth

Bluetooth werkt eveneens op de 2.4 GHz band en is gebaseerd op de IEEE 802.15.1 standaard. Het grote verschil met Wi-Fi is het nominale zendvermogen, dit ligt wel 15 dB lager. Door het lage vermogen, maar toch hoge datasnelheid van 2 Mbit/s wordt het vaak gebruikt voor het verzenden van bestanden tussen apparaten, bijvoorbeeld voor het zenden van audio naar een draadloze koptelefoon of het uitwisselen van foto's tussen mobiele telefoons.

Bluetooth radio's zijn onderverdeeld in drie klassen, gebaseerd op de maximale communicatie-afstand:

1. klasse 1: een afstand tot 100 m;
2. klasse 2: een afstand tot 10 m;
3. klasse 3: een afstand tot 1 m.

Voor het sensorsysteem dat ontworpen wordt zou dus een radio uit klasse 1 of 2 nodig zijn.

In tabel 4.3 staat voor elke techniek die vaak gebruikt wordt voor het zenden van RF signalen een voorbeeld van een module weergegeven. Voor RuBee zijn er helaas nog geen standaard modules te vinden. Daar zijn de specificaties ingevuld waaraan de IEEE standaard moet voldoen.

Tabel 4.3: Verschillende RF technieken en hun eigenschappen

	ZigBee[10]	Bluetooth [10] [25]	Wi-Fi [10] [26]	UWB [27][28][10]	Rubee (IEEE 1902,1) [29]	Z-wave [30]
Frequentie	868/915 MHz. 2,4 GHz	2,4 GHz	2,4 GHz; 5 GHz	3,1-10,6 GHz	131 kHz	868.42 MHz
Bereik	10-100m	10-100m	100m	10m	0,5-50m	30m
Max datasnelheid	250 kb/s	1 Mb/s	54 Mb/s	110 Mb/s	9,6kb/s	9,6 kb/s of 40 kb/s
Zendvermogen	-25-0 dBm	0 - 10 dBm	15-20 dBm	-41,3/MHz	0-5 dBm	-22- -2 dBm
Bandbreedte	0,3/0,6 Mhz ; 2 Mhz	1 Mhz	22 Mhz	500 MHz-7,5 GHz	n.v.t.	FSK: frequency deviation ± 20 KHz
Interferentie magnetisch veld	Nee	Nee	Nee	Nee	Ja	Nee
Gevoelig voor verschuivingen	Nee	Nee	Nee	Nee	Nee	Nee
Dimensies	2,44 x 2,76 cm ²	2,7 x 1,4 cm ²	23,1 x 2,1 cm ²	0,5 x 0,5 cm ²	n.v.t.	1,25 x 1,36 cm ²

Hoofdstuk 5

Keuze van het definitieve ontwerp

In dit hoofdstuk worden de criteria opgesteld waaraan de zendtechniek moet voldoen volgens het Programma van Eisen (paragraaf 5.1). In de sectie Toetsing van de ontwerpalternatieven aan de criteria (paragraaf 5.2) wordt duidelijk gemaakt waarom er voor ZigBee wordt gekozen en waarom niet voor een andere zendtechniek. Ook wordt er verteld welke hardware er is gekozen voor de zendtechniek en wat de redenen daarvoor zijn.

5.1 Criteria

Er zijn een aantal delen van het programma van eisen die essentieel zijn om aan de opdracht te voldoen. De rest van de eisen die in het programma van eisen staan zijn allemaal te voldoen met elke transmissiemethode. De percentages achter de eisen geven aan voor hoeveel procent deze eis meetelt in het eindoordeel. De volgende eisen zijn essentieel voor de werking van het systeem:

1. De draadloze transmissie moet kunnen plaatsvinden over minimaal 10 meter. (15%)
De afstand die overbrugd moet worden is de afstand tussen de transmissiemodule in de MRI-scanner en de transmissiemodule aan de rand van de scankamer. Deze afstand bedraagt in het geval van het LUMC 8 meter.
2. De patiënt moet lichtelijk kunnen bewegen zonder dat de transmissie wordt verbroken. (15%)
3. Er moet met een datasnelheid van minstens 120 kbit/s data kunnen worden verstuurd. (15%)
Het versturen van minstens 120 kbit/s is een eis die is gesteld vanuit het andere team dat de signaalbewerking doet.
4. De componenten moeten een minimale hoeveelheid ferromagnetische materialen bevatten. (20%)
In een MRI-scanner is een zeer sterk magnetisch veld aanwezig. De transmissiemodule en de batterij mogen niet aangetrokken worden door de MRI-scanner. Dit zou er namelijk voor kunnen zorgen dat de patiënt gewond raakt en dat het systeem beschadigd raakt.
5. Het systeem moet bestand zijn tegen de volgende vormen van interferentie: 298 MHz radiofrequente golven, statische magnetische veld van de MRI scanner, laagfrequente magnetisch veld, omgevingslicht en geluidsgolven (gradiëntmagneten). (15%)
Het is van groot belang dat de signalen goed aankomen en hierbij niet gestoord worden door de golven die de MRI-scanner produceert. Behalve omgevingslicht zijn er geen andere stoorsignalen aanwezig in de ruimte waarin de transmissie moet plaatsvinden dan de signalen die de MRI-scanner produceert.

De zendmethodes die hieraan voldoen kunnen verder worden gerangschikt aan de hand van de volgende criteria:

1. Het energieverbruik; (7%)

De transmissiemodule die aangesloten zit op de sensor moet van energie worden voorzien door een batterij. Door de opdrachtgever is verteld dat het handig zou zijn als de batterij de transmissiemodule in ieder geval 4 uur kan voorzien van voldoende energie. Hiervoor is het van belang dat de transmissiemodule een zo laag mogelijk energieverbruik heeft. Dit geldt alleen voor de transmissiemodule aan de kant van sensor want de transmissiemodule aan de kant van de computer wordt gevoed vanuit het elektriciteitsnet.

2. De afmetingen; (7%)

De opgegeven maximale grootte van het systeem, door de opdrachtgever, is 10 x 10 x 5 cm. Aangezien in dit systeem ook nog een microprocessor en een batterij moeten worden geplaatst is het van belang dat de transmissiemodule niet te veel ruimte in neemt.

3. Het gewicht. (6%)

Het is een eis van de opdrachtgever dat het maximale gewicht niet boven de 0,5 kilogram uit komt. Het gewicht is voornamelijk afhankelijk van de batterij en niet van de transmissiemodule.

Deze eisen zijn minder essentieel omdat het product nog wel zou werken als het niet aan deze criteria voldoet. Echter het zou niet het gewenste gewicht of grootte hebben.

5.2 Toetsing van de ontwerpalternatieven aan de criteria

5.2.1 Optische communicatie

Bij het zenden door middel van optische technieken wordt er gebruik gemaakt van infrarood. Een voordeel van infrarood is dat er geen magnetische interferentie plaatsvindt en dat ook de andere signalen die de MRI-scanner produceert geen invloed hebben op de zender. Een nadeel is dat het slecht voor de ogen is als men in een infraroodled kijkt. Het grootste nadeel is dat er een *Line of sight* nodig is. Dit betekent dat er een directe lijn nodig is tussen de verzender en de ontvanger. Dit is een groot nadeel omdat de zender kan bewegen doordat de patiënt beweegt. Hierdoor zou de *Line of sight* kunnen komen te vervallen. Daardoor zou er geen transmissie meer mogelijk zijn waardoor de scan zou moeten worden gestopt en opnieuw zou moeten worden gestart als de zender weer goed is gelegd.

Aangezien de benodigde data rate 120 kbit/s is, zou het zenden via infrarood niet voldoen. Infrarood garandeert (voor een afstand van maximaal 1 m) een data rate van 115,2 kbit/s.

5.2.2 Akoestische communicatie

Bij akoestisch zenden wordt er weinig gebruik gemaakt van het zenden door lucht. Er zijn voornamelijk onderzoeken bekend die akoestisch zenden onderwater.

Bij het zenden door de lucht kan maar een lage datasnelheid gehaald worden, dit is bij benadering 5 kb/s. Dit is te weinig voor het versturen van het signaal van één sensor. Bij meerdere sensoren gaat het dus zeker niet lukken.

Bovendien heeft akoestisch het probleem dat er niet meer dan 3 meter overbrugd kan worden door het multipathing probleem, veroorzaakt door reflecties van geluidsgolven.

5.2.3 Radiofrequente communicatie

Bij de radio frequente signalen zijn er twee technieken die niet aan de belangrijkste criteria voldoen. Dit zijn Z-wave en Rubee. Bij Z-wave is het grootste probleem dat er een maximale datasnelheid kan worden gehaald van 40 kb/s. Dit is te weinig voor één sensor en dus zeker te weinig voor meerdere sensoren.

Bij Rubee is het probleem dat het werkt door middel van magnetische signalen met een lange golflengte. Dit gaat waarschijnlijk interfereren met het magneetveld van de MRI-scanner. Bovendien is ook hier de

Tabel 5.1: Afweging tussen verschillende RF-technieken; ++=10 +=8 +/-6 -=4 - -=2

	ZigBee[10]	Bluetooth [10] [25]	Wi-Fi [10] [26]	UWB [27][28]	Rubee (IEEE 1902,1) [29]	Z-wave [30]
Bereik (20%)	++	++	++	+/-	++	++
Max datasnelheid (20%)	+	++	++	++	--	--
Energieverbruik (10%)	++	+/-	--	--	+/-	++
Interferentie magnetisch-veld (20%)	++	++	++	++	--	++
Gevoeligheid positie (20%)	++	++	++	++	++	++
Dimensies (10%)	++	++	--	++	n.v.t.	++
Eindoordeel	9,6	9,6	8,4	8,4	6,1	8,4

datasnelheid te laag om het signaal van één sensor over heen te sturen. Rubee is op dit moment ook niet commercieel beschikbaar, er zijn geen modules te vinden waar de Rubee techniek in is geïmplementeerd.

De rest van de technieken (ZigBee, Bluetooth, WiFi en Ultra Wide Band (UWB)) voldoen wel aan de eisen die in de eerste paragraaf zijn gesteld. Deze technieken zijn in tabel 5.1 gezet. Uit de tabel is af te leiden dat UWB en Wi-Fi niet voldoen doordat ze te veel energie verbruiken. Bovendien is de Wi-Fi module te groot om in de behuizing te passen.

ZigBee en Bluetooth zijn de twee meest geschikte technieken, waarbij Bluetooth een hogere datasnelheid heeft en ZigBee minder energie verbruikt. Aangezien de datasnelheid van ZigBee voldoende is om meerdere signalen te zenden en te ontvangen wordt er uiteindelijk voor ZigBee gekozen.

Module

Er zijn veel verschillende ZigBee modules bij veel verschillende fabrikanten. Veel webshops bieden dezelfde componenten aan. Er wordt gekozen voor de webshop van Farnell, omdat het LUMC hier al veel zaken mee doet. Bij Farnell zijn er in totaal 95 ZigBee modules. Veel van deze modules hebben een *shield* erop zitten die van ferromagnetische materialen is gemaakt. Met dit *shield* kunnen RF signalen buiten de deur worden gehouden. Dit ferromagnetische *shield* is een probleem omdat deze wordt aangetrokken door de MRI-scanner, wat een risico kan opleveren voor de patiënt. Er zijn twee modules die geen ferromagnetisch *shield* hebben. Er wordt gekozen voor een component waar ook de aansluitdraden goed op te solderen zijn. De uiteindelijke module wordt de AC163028, een ZigBee tranceiver, een module die zowel kan zenden als ontvangen.

Hoofdstuk 6

Topologie van het ontwerp

Het te ontwerpen systeem kan op verschillende manier worden ingericht. In dit hoofdstuk komen twee mogelijke topologieën aan bod. In de eerste paragraaf zullen de functionaliteiten van het systeem nog even kort worden toegelicht. In paragraaf twee wordt besproken welke topologie beter presteert.

6.1 Functionaliteiten van het systeem

De hoofdtak van het te ontwerpen systeem is de analyse van het akoestische hartsignaal, ook wel Phonocardiogram (PCG) genoemd. Uit dit periodieke signaal moet een referentiepunt worden gehaald, waarop de MRI-scanner een afbeelding kan genereren. De harttoon S_1 , uitgelegd in paragraaf 2.1, is de meest geschikte kandidaat hiervoor. De amplitude van het signaal is dan het grootst en er komen duidelijk herkenbare frequenties in het signaal voor.

Ten tweede moet de gebruiker van het systeem *real-time* de PCG, temperatuurmeting en ECG kunnen zien op een computerscherm. *Real-time* betekent in dit geval dat de gegevens meteen zichtbaar moeten zijn na acquisitie van de sensor. De vertraging mag dus bijvoorbeeld niet 1 seconde bedragen, maar aan een kleine vertraging door de transmissietijd valt niet te ontkomen. Een grondigere analyse van de transmissietijd is beschreven in paragraaf 6.2.

Ten derde moet de gebruiker een verschuiving kunnen aanbrengen aan het referentiepunt in de PCG voor de MRI-scanner. Een medicus kan zo de scan laten beginnen op een ander moment in de hartcyclus.

6.2 Topologie van het systeem

Om aan al deze taken te voldoen zijn twee topologieën ontworpen. Deze zijn afgebeeld in figuur 6.1 en 6.2. De pijlen in de afbeelding geven de richting van de communicatie aan. Het systeem is ontworpen om de volgende binnenkomende signalen af te handelen: ECG, PCG en temperatuurmeting. Hoewel het systeem hiervoor ontworpen is, worden de biomedische signalen niet aangeboden door een sensor.

De PCG-metingen zijn opgenomen met een microfoon, gedigitaliseerd met behulp van een oscilloscoop en MATLAB en vervolgens opgeslagen in het geheugen van microcontroller 1. De temperatuurmetingen worden random gegenereerd en opgeslagen in het geheugen van dezelfde microcontroller.

Vervolgens worden de biomedische signalen verstuurd via de draadloze link en afgehandeld door de tweede microcontroller. Deze microcontroller communiceert met een computer via een USB-poort. Via software op een computer worden de signalen ingelezen, geanalyseerd en weergegeven op een scherm. Hierin zijn beide topologieën gelijk aan elkaar.

Het grootste verschil tussen topologie 1 en topologie 2 is de locatie waar de detectie van het referentiepunt voor triggering plaats vindt. In topologie 1 bepaalt de software waar het referentiepunt ligt in het signaal. Op het moment dat de scan moet worden genomen, wordt vervolgens via de draadloze link het bericht teruggestuurd naar microcontroller 1. Deze zal vervolgens een trigger aanbieden aan de MRI-scanner.

In tegenstelling tot topologie 1, wordt het referentiepunt bij topologie 2 niet bepaald door software op de computer. Microcontroller 1 zal direct de PCG-meting analyseren en de MRI-scanner aansturen. Daarnaast verstuurt de microcontroller naar de computer op welk moment in de tijd de trigger aan de MRI-scanner aangeboden is, zodat de computer dit kan weergeven op het computerscherm.

Het grootste euvel van topologie 1 is de transmissietijd. Er zit een tijd tussen het moment van versturen vanaf ZigBee 1 en het moment van verwerken met software. De directe transmissietijd of vertraging D_{direct} vanaf ZigBee 1 naar ZigBee 2 kan uitgedrukt worden met de volgende formule [11]:

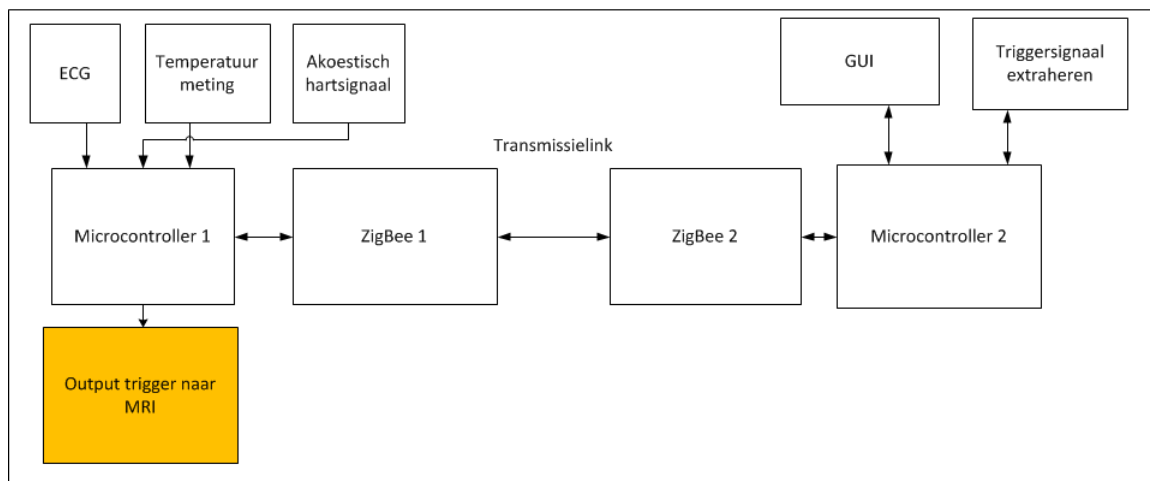
$$D_{direct} = \frac{L}{R_b} + T_{prop} + T_{proc} \quad (6.1)$$

waar L de grootte van het datapakket is, R_b de datasnelheid, T_{prop} de propagatietijd van elektromagnetische golven in lucht en T_{proc} de tijd die nodig is om een datapakket te verwerken. De propagatietijd op een afstand van 10 meter is 33 ns en is verwaarloosbaar. De verwerkingstijd T_{proc} wordt bepaald door de tijd die de ZigBee hardware nodig heeft om een datapakket te bufferen en aan te bieden aan de microcontroller, en door de vertraging die geïntroduceerd wordt door het CSMA algoritme [11]. De *jitter* wordt veroorzaakt door verschuivingen in de triggermomenten en dient zo klein mogelijk te zijn. Indien de transmissietijd D_{direct} niet constant is voor elk datapakket, wordt er op een ander moment getriggerd in de hartcyclus en zal de jitter dus groter worden, met onscherpe scanafbeeldingen als gevolg. Een indicatie van de transmissietijd in de praktijk is gegeven in paragraaf 8.2.

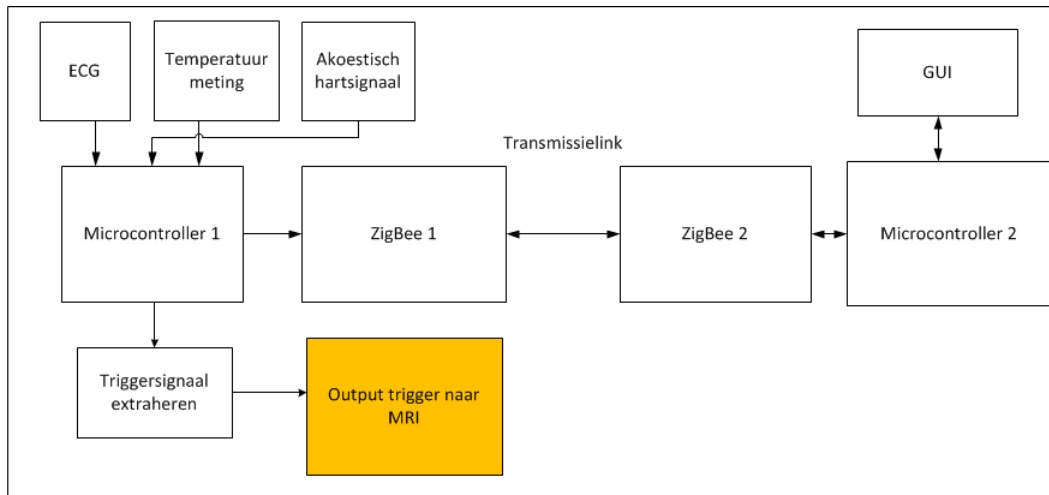
Sommige datapakketten zullen bitfouten bevatten. Indien een pakket wordt ontvangen met een bitfout zal geen ontvangstbevestiging verstuurd worden en moeten een pakket opnieuw worden verstuurd. De ZigBee hardware biedt de mogelijkheid om tot 3 keer toe het pakket te versturen voordat de transmissie wordt gestaakt. Herhaling van de transmissie zal leiden tot een grotere transmissietijd D_{direct} en jitter veroorzaken.

Aan topologie 1 kleven niet alleen maar nadelen. De PCG kan geanalyseerd worden met software op de computer, wat veel voordelen biedt ten opzichte van de analyse met een microcontroller. De computer heeft beschikking over een veel groter geheugen en grotere kloksnelheden. Als de analyse van de PCG gebruik maakt van ingewikkelde algoritmen en de middelen van een microcontroller ontoereikend zijn, biedt topologie 1 een uitweg.

Voor het prototype wordt geen geavanceerd algoritme gebruikt om het referentiepunt S_1 te detecteren. Daarom wordt het systeem ontworpen volgens topologie 2. Het PCG signaal wordt direct geanalyseerd op microcontroller 1 en de variabele transmissietijd zal dus niet leiden tot *jitter*.



Figuur 6.1: Topologie 1



Figuur 6.2: Topologie 2

Hoofdstuk 7

Implementatie en realisatie

In het vorige hoofdstuk is besproken welke topologie gebruikt is voor de implementatie van het systeem. In dit hoofdstuk wordt besproken hoe de modules met elkaar verbonden zijn en hoe de interface tussen de verschillende componenten eruit ziet. In paragraaf 7.1 wordt de communicatie link beschreven. Vervolgens wordt in paragraaf 7.2 besproken hoe de microcontrollers communiceren met de ZigBee modules en waarom voor bepaalde microcontrollers is gekozen. In paragraaf 7.3 komt de implementatie van de software op de computer aan bod.

7.1 Communicatie link

De ZigBee modules communiceren op een bidirectionele manier. Voor deze communicatie zijn twee ZigBee modules vereist. Enerzijds verstuurt ZigBee 1 de biomedische signalen en de detectie van de harttoon naar ZigBee 2. Anderzijds kan via ZigBee 2 de verschuiving in de detectie ingesteld worden.

De transmissie link wordt opgebouwd met twee Microchip AC163028 bordjes. De belangrijkste component op dit bordje is de MRF24J40MA ZigBee transceiver. Hoewel het bordje ontworpen is voor de PIC-familie microcontrollers, is het bordje ook programmeerbaar met een andere willekeurige microcontroller. Het geheugen van de chip is geïmplementeerd als statische RAM en wordt onderverdeeld in *Control Registers* en data buffers. De *Control Registers* bepalen de instellingen van de chip.

Communicatiesystemen tussen twee ZigBee modules, ook wel *nodes* of knopen genoemd, kunnen worden opgedeeld in de volgende klassen:

- Simplex;
- Half-duplex;
- Full-duplex.

Simplex systemen staan communicatie toe in één richting, van zender naar ontvanger. Een voorbeeld hiervan is FM radio of TV. In het geval van FM radio communiceert één zender met alle ontvangers die op een bepaalde frequentie staan ingesteld. De ontvangers luisteren slechts en sturen niets terug. Voor het totale communicatiesysteem in de MRI-scanner voldoet een simplex systeem niet, omdat communicatie in beide richtingen nodig is.

In een half-duplex systeem kunnen beide *nodes* zenden en ontvangen, maar niet tegelijkertijd. Een voorbeeld van een dergelijk systeem is een walkietalkie. De ontvanger kan de zender niet verstaan als ze tegelijkertijd aan het praten zijn.

Een manier om een half-duplex systeem te organiseren is door *time-division multiplexing (TDM)* toe te passen [31]. Iedere node krijgt een vaste tijdslot toegedeeld waarin deze kan zenden. De node kan dus periodiek gebruik maken van de hele capaciteit van het communicatiemedium. De MRF24J40MA heeft geen

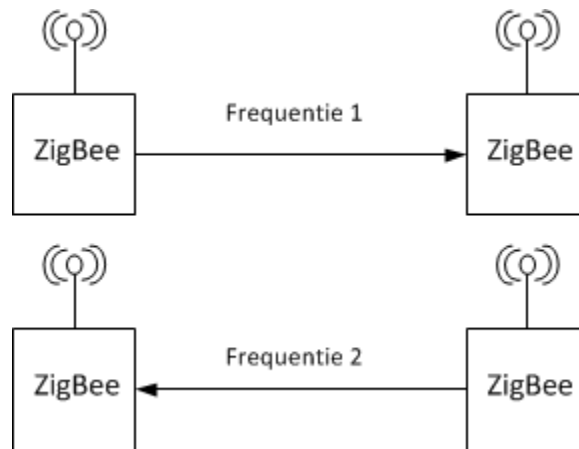
standaard optie om *time-division-multiplexing* toe te passen. Deze techniek zal dus extra geïmplementeerd moeten worden.

Een tweede manier is om alleen data te verzenden indien het transmissiemedium niet bezet is. Dit concept wordt Carrier Sense Multiple Access (CSMA) genoemd. Voor de verschillende vormen van CSMA zie Van Mieghem [31]. De MRF24J40MA biedt de mogelijkheid om *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) in te schakelen. Iedere node kijkt of het communicatiekanaal gebruikt wordt door een andere node. Indien dit waar is, wacht de node een bepaalde periode voordat opnieuw bepaald wordt of het kanaal gebruikt wordt. Deze periode wordt de *backoff period* genoemd. Bij elke keer dat een bezet kanaal gedetecteerd wordt, neemt deze periode toe.

In een full-duplex systeem kan er bidirectioneel gecommuniceerd worden, maar in tegenstelling tot half-duplex kunnen nodes dit nu wel tegelijk doen. Hiervoor wordt veelal *frequency-division multiplexing (FDM)* gebruikt. Voor een point-to-point verbinding kan voor de heengaande communicatie een andere frequentie gebruikt worden dan voor de teruggaande berichten.

Het voordeel hiervan ten opzichte van half-duplex is dat nodes niet op elkaar hoeven wachten met verzenden van informatie en daardoor kan de vertraging tot een minimum teruggebracht worden. Ook is de datasnelheid hoger, omdat van de volledige capaciteit van het medium gebruikt kan worden gemaakt. Daarnaast hoeft er ook geen mechanisme voor foutcontrole ingebouwd te worden, omdat er nooit botsingen tussen datapakketten optreden.

In theorie lijkt full-duplex veel voordelen te bieden boven half-duplex, maar in de praktijk is het bouwen van een full-duplex systeem lastiger. Omdat een ZigBee module slechts op één frequentie is ingesteld, is er een tweede paar nodes nodig om op een andere frequentie te communiceren. De kosten van het systeem nemen hiermee ook toe. Figuur 6.3 geeft een schematische weergave van een full-duplex systeem, opgebouwd met twee paar radio's.



Figuur 7.1: full-duplex systeem

Het is de vraag of een full-duplex systeem wel nodig is. De enige communicatie naar node 1 toe betreft het instellen van de verschuiving van de harttoon detectie. Dit kan voorafgaand aan het starten van de scanprocedure gebeuren. Gedurende de scan worden alleen de biomedische signalen verzonden van node 1 naar node 2. Omdat er voor foutcontrole nog steeds ontvangstbevestigingen terug gestuurd moeten worden, is er wel nog steeds sprake van een bidirectioneel systeem.

Er is gekozen voor een half-duplex systeem gebaseerd op CSMA. Deze techniek is standaard beschikbaar op de hardware van de MRF24J40, de implementatietijd is dus kort in vergelijking met de andere opties. Bij CSMA kan de vertraging door de aard van de techniek groter zijn. De biomedische signalen worden dan een fractie later weergegeven door de software. Vertraging heeft geen invloed op de hoofdtaak van het systeem, detectie van de harttoon en aansturing van de MRI-scanner. Daarom weegt deze techniek wel op tegen de hogere kosten en langere implementatietijd van een full-duplex systeem. Een opsomming van de

Tabel 7.1: Afwegingen voor verschillende duplex systemen; ++=10 +=8 +/-6 -=4 - -=2

	Half-duplex		Full-duplex
Techniek	TDM	CSMA	FDM
Kosten (25%)	++	++	-
Datasnelheid (35%)	-	+/-	++
Implementatietijd (40%)	--	++	+/-
Eendoordeel	5,5	8,6	6,9

weegfactoren van de verschillende technieken zijn weergegeven in tabel 7.1.

7.1.1 Opbouw van de datapakketten

De MRF24J40 is gebaseerd op de IEEE 802.15.4 standaard. Deze standaard geeft aan hoe een datapakket opgebouwd moet worden. De standaard specificeert alleen de fysieke laag (PHY) en een gedeelte van de *data link* laag (MAC). Dit zijn twee lagen die worden gedefinieerd in het Open Systems Interconnecting (OSI) model. In totaal zijn er zeven verschillende lagen en iedere laag heeft een hoger abstractie niveau. PHY is de minst abstracte laag en bepaalt de elektrische en fysieke eigenschappen van apparaten. Figuur 7.2 geeft weer hoe de twee lagen zijn opgebouwd en hoe ze zich tot elkaar verhouden. Omdat de ZigBee op meerdere manier gebruikt kan worden is de MAC laag opgedeeld in verschillende deellagen. Hieronder wordt besproken welke deellagen dit zijn, ook wel *frames* genoemd, en waarom bepaalde frames al dan niet gebruikt worden.

Met een *Acknowledgment Frame* bevestigt een module of een pakket goed ontvangen is. Voor een correcte weergave van de biomedische signalen is het belangrijk dat alle datapakketten goed ontvangen worden.

De MAC Command Frame biedt de mogelijkheid om individuele nodes draadloos in te stellen. Dit kan bijvoorbeeld handig zijn als de ZigBee nodes op een lastig te bereiken plek zitten, of als erg veel nodes ingesteld moeten worden. Door middel van commando's worden registers, besproken in appendix A, ingesteld. Deze registers bepalen de functionaliteit van de node. Aangezien slechts twee ZigBee's gebruikt worden, is het gemakkelijker om ze direct in te stellen via de microcontroller.

Een beacon frame is bedoeld om ZigBee modules vanuit de slaapstand naar een actieve stand te brengen. Aangezien de ZigBee nodes in de voorgestelde topologie continu de signalen moeten versturen en dus altijd al in actieve stand staan, zal deze frame nooit gebruikt worden.

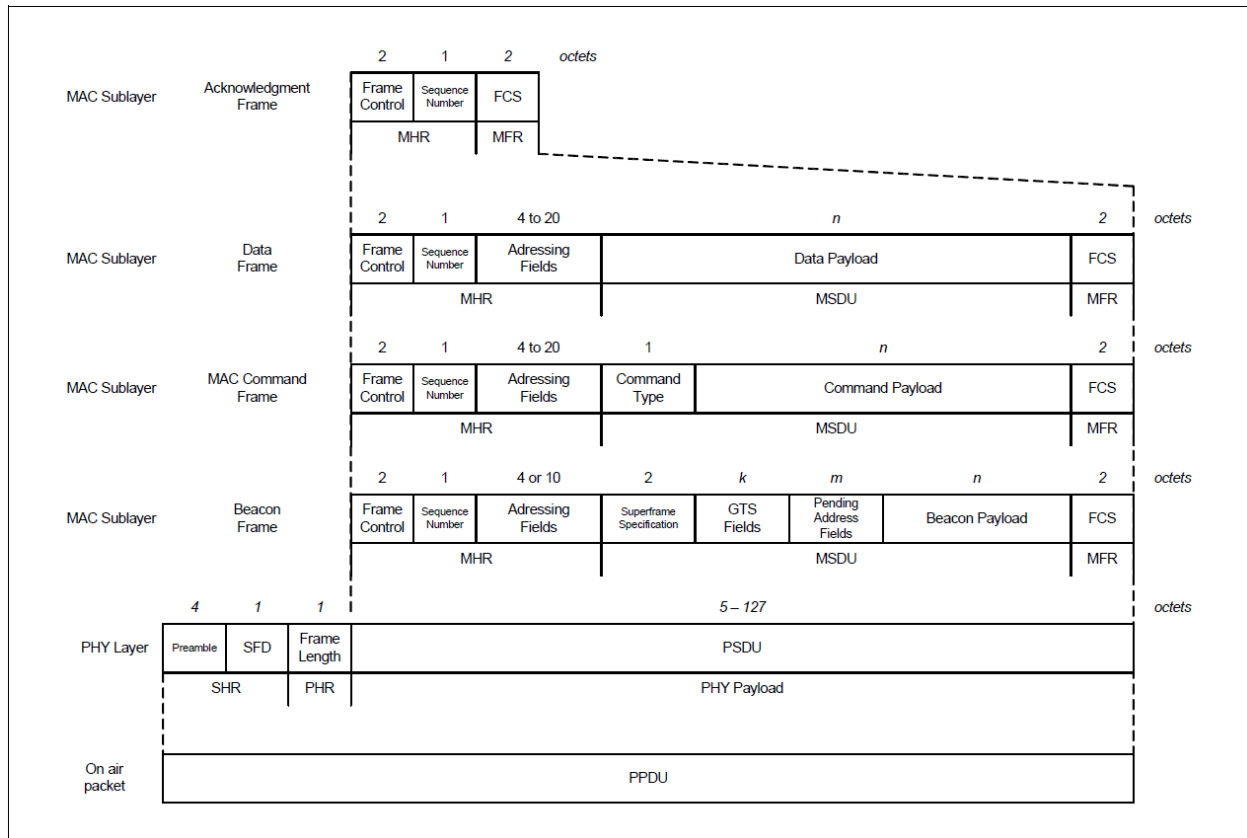
In een Data Frame worden de daadwerkelijke signalen verstuurd. Omdat er meerdere signalen verstuurd kunnen worden is het belangrijk voor de zender en ontvanger dat er overeenstemming is over wat voor soort signaal verstuurd wordt.

Hoewel alle signalen in digitale vorm worden aangeboden, is de structuur van ieder signaal wel anders. Een temperatuurmeter vindt twee keer plaats in een seconde, terwijl de meting van het akoestische hartsignaal met een frequentie van maximaal 600 Hz gedaan wordt. Het aantal bits per meting is ook verschillend.

De maximale grootte van een datapakket is 127 bytes. Een aantal bytes hiervan worden gebruikt voor de header van de *PHY Layer* en voor de *MAC Layer*, zoals te zien is in figuur 7.2. In deze layer staat onder andere welke node de pakket verzonden heeft, en waar de pakket naar toe moet. De daadwerkelijke data die verzonden wordt, ook wel *data payload* genoemd, kan maximaal 102 bytes omvatten.

In figuur 7.3 is de structuur afgebeeld van de Data Payload, bij het ontwerpen van de structuur is rekening gehouden met de verscheidenheid aan signalen en de maximale lengte van 102 bytes.

Tussen de twee ZigBee modules moet overeenstemming bestaan wat voor soort signaal verstuurd wordt. Het vlak Type geeft daarom het type weer van het signaal. In tabel 7.2 is weergegeven waar ieder type voor



Figuur 7.2: MAC lagen [32]

staat.

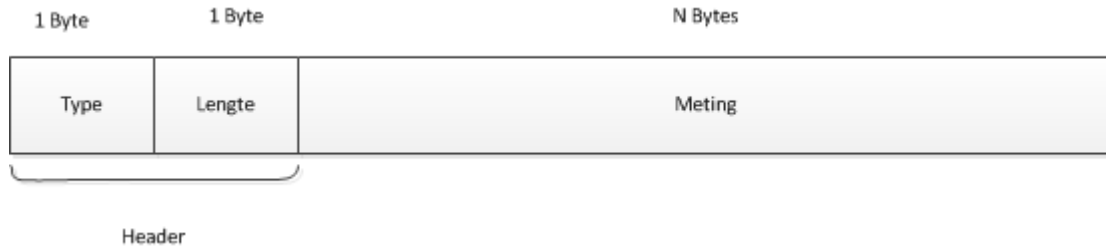
De ontvanger van een datapakket weet op het moment van ontvangen niet uit hoeveel bytes de meting bestaat. Omdat deze lengte voor de verschillende signalen niet hetzelfde is, wordt met dit vak aangegeven hoeveel bytes ingelezen moeten worden.

In dit gedeelte van de pakket worden de ruwe meetgegevens geplaatst. Het aantal bits van ieder meetpunt kan verschillen. Omdat uitlezen per 8 bits (1 byte) gemakkelijk is, worden alle gegevens geencapsuleerd in bytes. Dit betekent bijvoorbeeld voor een 10-bit meting dat er 2 bytes gebruikt worden om deze te versturen.

7.2 Keuze microcontroller en interface met ZigBee en computer

De voornaamste communicatiepoorten op de pc zijn de USB-poort en de seriële poort. Op veel laptops en pc's is meestal geen seriële poort meer aanwezig. Daarom is ervoor gekozen om de USB interface te gebruiken. Omdat er geen directe conversie bestaat tussen SPI en USB wordt er een microcontroller ingezet om te communiceren met de pc, in de topologie van hoofdstuk 6 is deze aangeduid als microcontroller 2. De microcontroller moet aan een aantal eisen voldoen:

- Eenvoudig te programmeren, zodat er geen tijd verloren gaat met randzaken;
- USB-poort connectie aanwezig;
- 6 Aansluitpinnen voor de MRF24J40MA;
- Kloksnelheid groot genoeg om de maximale snelheid van ZigBee aan te kunnen: 250 kbit/s;



Figuur 7.3: De structuur van de data payload van een pakket

Tabel 7.2: De verschillende typen signalen

Type (hexadecimaal)	Signaal
0x01	Akoestische hart-meting
0x02	Temperatuurmeting
0x03	ECG-meting
0x04	Trigger

- Een levertijd van minder dan 3 dagen gezien de korte looptijd van het project;
- Een maximale prijs van 30€, om niet buiten het budget van het LUMC te stijgen.

De Arduino Uno voldoet aan al deze eisen. Hij kan geprogrammeerd worden via een USB-kabel, er is geen externe programmer nodig. De Uno heeft een ATmega328 microcontroller, die kloksnelheden aankan tot 20 MHz. De kosten van het bordje zijn ook laag: slechts 25€. In feite volstaan bijna alle hedendaagse microcontrollers qua snelheid. Door de vele randaccessoires, zoals een USB-connectie en LEDs, is de Arduino Uno een goede ontwikkelomgeving voor het prototype.

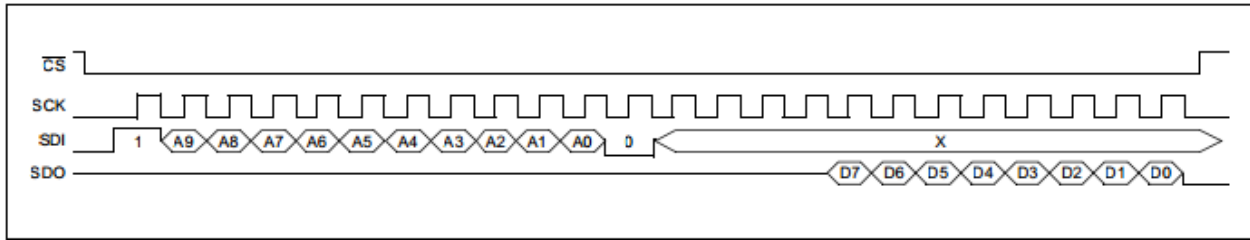
Communicatie met de MRF24J40MA verloopt via een *4-wire Serial Peripheral Interface (SPI)* poort. Hoe de input- en outputbuffers uitgelezen en geschreven kunnen worden is afgebeeld in figuur 7.4 respectievelijk figuur 7.5. De snelheid van het uitlezen kan bepaald worden door de microcontroller via de SCK pin. Gedurende iedere klokcyclus kan een bit worden ingelezen of geschreven worden, als de CS pin op een logische '0' wordt gezet.

Voor het uitlezen wordt eerst een adres meegegeven in de bits aangegeven met letter A. De 8 bits waarde op dat adres wordt vervolgens meegegeven op de SDO pin, aangegeven met letter D. Op deze manier kunnen registers en buffers worden uitgelezen. Schrijven naar de MRFJ40MA gebeurt op een soortgelijke manier.

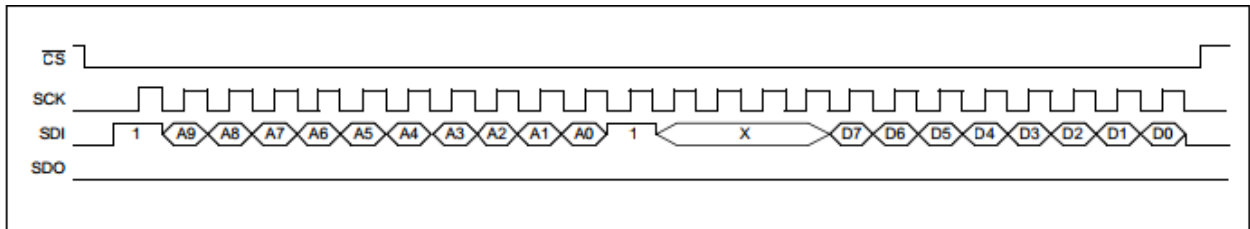
De Arduino Uno kan via een virtuele seriële poort communiceren met de computer over een USB-kabel. Indien de correcte drivers worden geïnstalleerd op de computer, ziet de computer deze verbinding dus ook als een seriële verbinding. De maximale datasnelheid die hierover behaald kan worden is 115.2 kbit/s. De software op de computer moet vervolgens wel de data inlezen op deze snelheid.

7.3 Graphical User Interface

Een arts of onderzoeker bekijkt de resultaten van een MRI-scan op de computer. Vaak worden er tijdens het scannen al afbeeldingen weergegeven van de patiënt. Tegelijkertijd ziet de arts ook de hartslag van de patiënt. De triggermomenten zijn als zwarte verticale strepen weergegeven over deze hartslag heen, zoals in figuur 7.6. De arts kan vervolgens de scan starten of stoppen met de aangegeven knoppen. Een dergelijk venster wordt een Graphical User Interface (GUI) genoemd.



Figuur 7.4: Uitlezen van SPI



Figuur 7.5: Schrijven naar SPI

Om te demonstreren dat de transmissie van de signalen goed verloopt, worden deze weergegeven in een zelf ontworpen Graphical user interface (GUI). De GUI moet in essentie lijken op figuur 7.6, zodat medici van het LUMC die de oude GUI gewend zijn snel kunnen werken met de nieuwe GUI. De GUI biedt de gebruiker de mogelijkheid om aanpassingen te maken aan het triggermoment. In deze paragraaf zal beschreven worden op welke manier de GUI opgebouwd is. Eerst wordt uitgelegd welke programmeertaal hiervoor gebruikt is.

7.3.1 Afweging voor programmeertaal

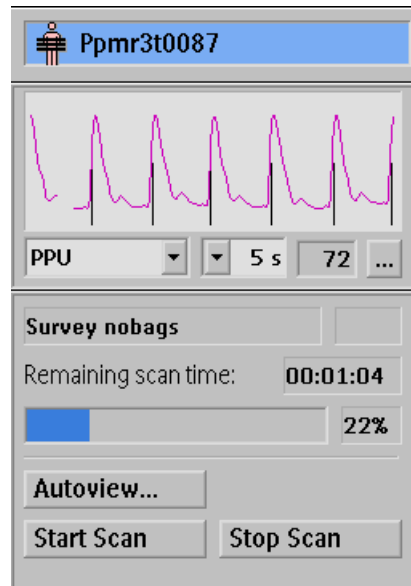
Er zijn vele verschillende programmeertalen beschikbaar waarmee een GUI geschreven kan worden. Bekende talen zijn C++/C, Java en Python. Ook met MATLAB, een programmeeromgeving die speciaal gemaakt is voor beeld- en signaalbewerking, is het mogelijk om een GUI te schrijven. Er zijn grote verschillen tussen de talen op het gebied van snelheid, lengte van de code en reeds beschikbare code (libraries).

De volgende zaken moeten in ieder geval mogelijk zijn met de programmeertaal:

- uitlezen van en communiceren met een USB poort;
- weergeven van grafieken in een scherm;
- instellen van het triggermoment door middel van een schuifbalk;

Met alle bovengenoemde programmeertalen kunnen deze taken wel worden uitgevoerd. Voor sommige bestaan al libraries, bibliotheken van reeds geschreven code. Voor Java bestaat bijvoorbeeld de Swing bibliotheek om snel complexe grafische applicaties te bouwen. In MATLAB is het weer makkelijker om rijen en matrices te bewerken.

Omdat de periode waarin de software ontwikkeld moet worden erg kort is, minder dan drie weken, is het van belang hoeveel code er geschreven moet worden om de aangegeven zaken werkend te krijgen. In het algemeen kunnen programma's die minder code vereisen sneller ontwikkeld worden. Er is gekeken naar het aantal regels dat nodig is om een seriële poort uit te lezen op een bepaalde baudrate. Dit is bij MATLAB en Python heel weinig, in de orde van 5 regels. Bij C++ en Java waren dit al 20 regels. Daarnaast is MATLAB speciaal ontworpen om grafieken te weergeven. Bij de andere programmeertalen is dit niet standaard in de taal geïmplementeerd en moeten daarom additionele libraries geïnstalleerd worden.



Figuur 7.6: Huidige GUI voor weergave van ECG in het LUMC

Tabel 7.3: Verschillende programmeertalen en hun voor- en nadelen; +=10 +=8 +/-6 -=4 - -=2

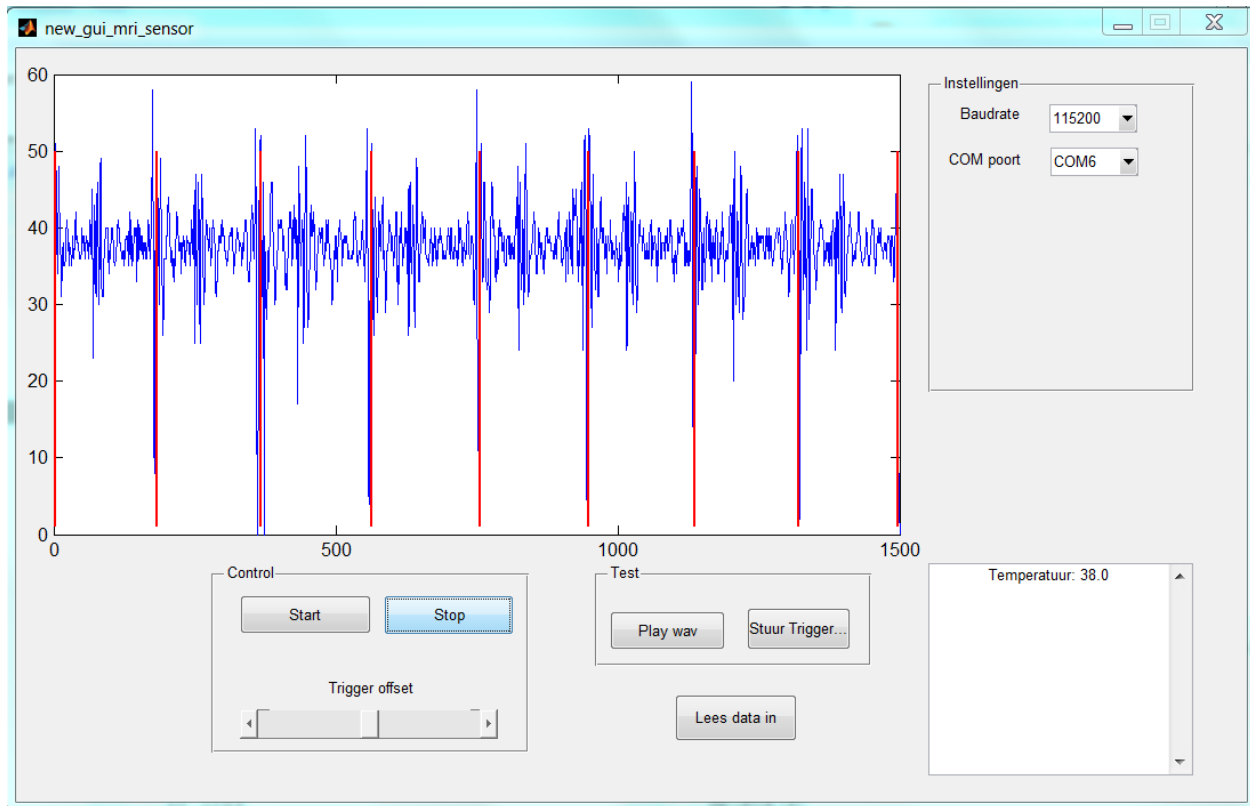
	MATLAB	C++/C	Java	Python
Ervaring (50%)	++	+/-	++	-
Beschikbare bibliotheken (25%)	++	+	+	+
Regels code (25%)	++	-	+	++
Eindoordeel	5,5	8,6	6,9	

Ook de ervaring in de programmeertalen speelt een rol. Bij de opleiding Elektrotechniek hebben de auteurs onderricht gekregen in Java. Door de vele practica hebben de auteurs van deze thesis ook veel ervaring met MATLAB. Wanneer veel vaardigheden in een programmeertaal al aanwezig zijn, worden minder snel fouten gemaakt en hierdoor volgt dus ook een kortere ontwikkelingstijd. Tabel 7.3 vat de voor en nadelen van iedere programmeertaal samen. Er is voor gekozen om de GUI in MATLAB te ontwikkelen.

7.3.2 GUI in MATLAB

De GUI is gemaakt met behulp van GUIDE, een tool in MATLAB. Grafische componenten, zoals knopjes, tekstvakken en schuifbalken kunnen verslept worden naar een venster. Wanneer de layout van het venster wordt opgeslagen, maakt MATLAB een .m bestand en een .fig bestand aan. In het .fig bestand staat welke grafische componenten op welke plek staan. In de .m file staan de *callbacks*. Een *callback* functie controleert het gedrag van een grafische component door een actie uit te voeren wanneer er bijvoorbeeld op de muis geklikt wordt, of een knop ingedrukt wordt. De volledige MATLAB code voor de GUI is gegeven in bijlage B.

Het eerste ontwerp van de GUI ziet er uit zoals in figuur 7.7. Als er op de start knop wordt gedrukt, begint het inlezen van de ZigBee. Met de stop knop wordt het inlezen gestopt. In het settings menu kan ingesteld



Figuur 7.7: De Graphical user interface (GUI) in Matlab

worden welke COM poort uitgelezen wordt en met welke datasnelheid dit gedaan wordt. De volgende regels code hieronder laten zien hoe deze poort kan worden uitgelezen.

```
s = serial('COM4'); % initialiseert com poort 4
s.InputBufferSize = 10000; % zet aantal bytes dat in de buffer kan worden opgeslagen
s.BaudRate = 9600; % symboolsnelheid op 9600
fopen(s); % opent de com poort
data = fread(s, s.BytesAvailable); % lees alle bytes in de buffer in, en sla ze op in de vector data.
```

Alle informatie die binnenkomt via de COM poort wordt opgeslagen in een buffer in het geheugen van MATLAB. De grootte van deze buffer wordt bepaald door de variable *s.InputBufferSize*. Wanneer de buffer vol is, kunnen geen andere pakketten ingelezen worden. Aangezien een pakket maximaal uit 102 bytes bestaat, moet de *InputBufferSize* ook minimaal 102 bytes zijn. Omdat het geheugen van hedendaagse computers groot genoeg is om grote hoeveelheden data aan te kunnen, kan de buffergrootte ook veel hoger ingesteld worden. Meerdere pakketten kunnen dan opgeslagen worden in de buffer.

De eerste byte van een pakket bepaalt het type meting dat verstuurd wordt. De tweede byte in een pakket vertelt de lengte van de meting. Indien deze lengte groter is dan het aantal bytes in de buffer, moet gewacht worden totdat de buffer gevuld is. Daarna kan de volledige meting verwerkt worden. Op basis hiervan kan de data ingelezen worden, en kan de bijbehorende meting weergegeven worden in het MATLAB venster.

De temperatuurmetingen worden weergegeven in een apart kader met tekstveld. De meest recente binngekomen temperatuur overschrijft de oude temperatuur.

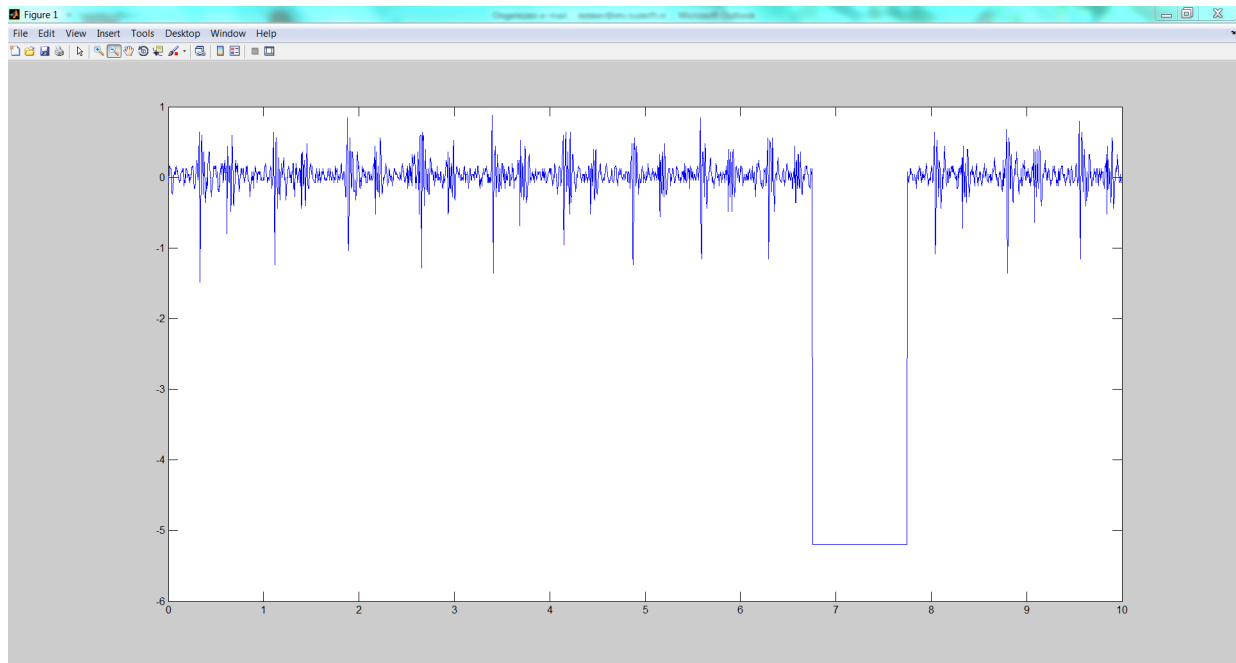
7.4 Triggering

Het akoestische signaal dat benodigd is om het prototype te testen is van te voren opgenomen door het team dat de sensor maakt. Zij hebben het voltage van de sensor om de 4 ms, dus met een frequentie van 250 Hz, met een oscilloscoop gemeten. De gegevens van de oscilloscoop zijn uitgelezen via de USB-poort en zijn opgeslagen als MATLAB-file. Dit signaal is geplott en weergegeven in figuur 7.8.

De amplitude van het signaal ligt tussen de -2 V en 2 V. De spanningen zijn omgezet naar waarden tussen de 0 en 60, zodat het volledig positief is en de informatie in het signaal behouden bleef. Daarna is ieder meetpunt omgezet naar een 8-bits waarde, zodat het op de Arduino kan worden opgeslagen.

Het signaal wordt om de 20 ms verstuurd in datapakketten van 5 metingen. Uit paragraaf 8.2 zal blijken dat dit de minst grote vertraging oplevert, zodanig dat het systeem nog kan functioneren.

Het triggersignaal moet gegenereerd worden op eenzelfde moment in de hartcyclus. Uit figuur 7.8 blijkt dat harttoon S_1 gekenmerkt wordt door een grote negatieve piek. In de eerste 6 seconden zitten acht hartslagen. Dit is vooral goed te zien door naar de pieken onder de 0 V te kijken. Door de amplitude van het signaal te vergelijken met een drempelwaarde kan de Arduino bepalen wanneer het triggersignaal moet worden gegenereerd.



Figuur 7.8: Een plot van de akoestische data

7.5 Programmeren Arduino

Met de Arduino integrated development environment (IDE) kan eenvoudig code voor de microcontrollers worden geschreven. De code kan vervolgens eenvoudig geupload worden met één van de knoppen in de IDE. Bij het programmeren van de Arduino Uno is gebruik gemaakt van bestaande code. Voor de communicatie via de SPI poort is een library gebruikt. Arduino kan geprogrammeerd worden in de programmeertaal C. De volledige code is beschreven in appendix C en D.

Hoofdstuk 8

Testen prototype

In de sectie opstelling (paragraaf 8.1) wordt behandeld welke testopstelling gebruikt is. In paragraaf 8.2 is onderzocht wat de invloed is van de grootte van een datapakket op de transmissietijd en de maximale datasnelheid. In paragraaf 8.3 wordt tot slot uitgelegd wat de eindresultaten zijn.

8.1 Opstelling

Op dit moment is het systeem alleen nog maar getest buiten de MRI-scanner. De Microcontroller 1 heeft ferromagnetische materialen in de USB connector zitten en op enkele andere plaatsen. De microcontroller zou dus erg aangetrokken worden door de scanner. Dit bleek ook uit een test in de MRI-scanner. In het totaal product van alle drie de onderdelen zou dit geen probleem zijn aangezien de microcontroller aan de kant van de sensor hier niet voor gebruikt wordt maar een microcontroller van het signaalbewerkingsteam.

In figuur 8.1 is te zien dat de ZigBee module aangesloten zit op een Arduino die op zijn beurt weer aangesloten zit op een USB-ingang van de computer.

In figuur 8.2 is te zien dat de Zigbee module is aangesloten op de Arduino en op het LEDje. Op dit moment wordt deze Arduino nog door middel van de USB kabel voorzien van stroom. Deze stroom zou ook kunnen worden voorzien door een batterij of een andere stroombron.

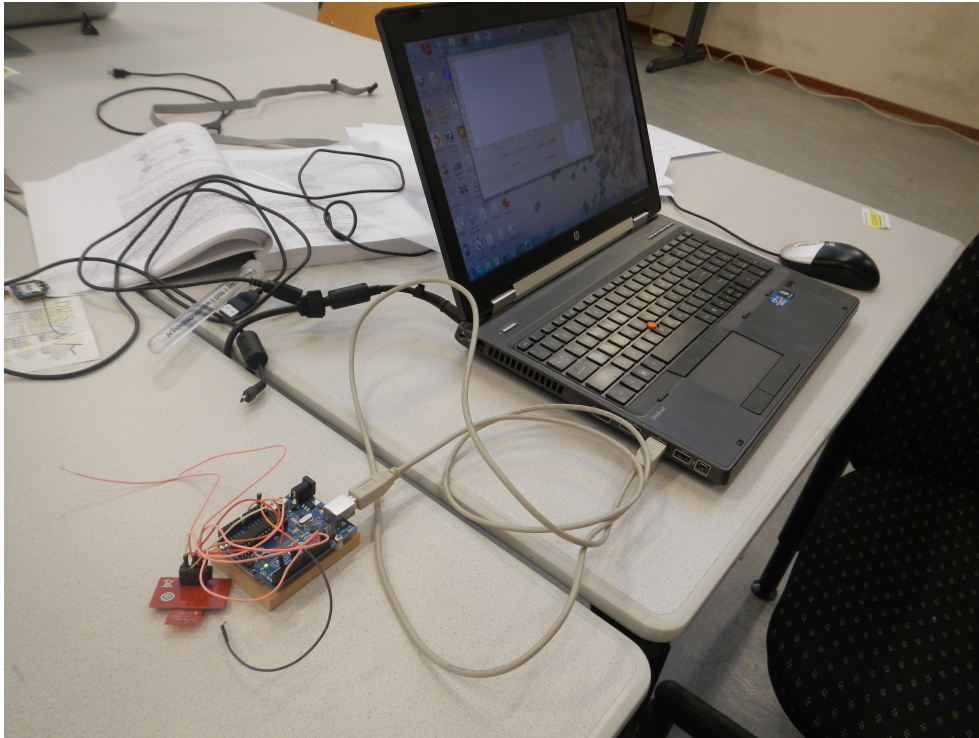
8.2 Prestaties van ZigBee

Het belangrijkste criterium voor de keuze van de topologie was de transmissietijd van de datapakketten. De theoretische formule voor de vertraging van de transmissie is gegeven in formule 6.1. In deze paragraaf worden de testen die zijn gedaan met betrekking tot de transmissietijd besproken.

De transmissietijd is getest door beide microcontrollers te synchroniseren. Met behulp van een drukknop is een interrupt gegenereerd op beide microcontrollers. Via een teller is bijgehouden hoe lang het datapakket erover deed om ZigBee 2 te bereiken. Deze teller is in werking gezet op het moment dat de datapakket verzonden is en gestopt op het moment dat het pakketje binnenkwam. De stand van de teller bepaalt nu het tijdsverschil tussen zenden en ontvangen.

In figuur 8.3 zijn de resultaten van deze test zichtbaar gemaakt. In totaal is de test tien keer gedaan en iedere test is met een andere kleur weergegeven. De x-as weergeeft de grootte van het verstuurd datapakket, zonder de overhead. De overhead heeft een vaste lengte van 11 bytes en moet nog bij de grootte worden opgeteld. De grootte is variabel gemaakt en het is zichtbaar dat de transmissietijd lineair toeneemt met de grootte van een pakket. De pieken in de figuur worden veroorzaakt door hertransmissies, waardoor de vertraging een stuk groter is geworden.

Uit de afbeelding blijkt dat de transmissietijd bij iedere test verschilt. Voor een gegeven grootte is het verschil tussen de maximale vertraging en minimale vertraging bepaald en weergegeven in figuur 8.4. De



Figuur 8.1: De ZigBee module en de Arduino aan de kant van de laptop

pieken worden veroorzaakt door de pieken in de transmissietijd. Uit de afbeelding blijkt dat het verschil in vertraging niet afhankelijk is van de grootte van een datapakket.

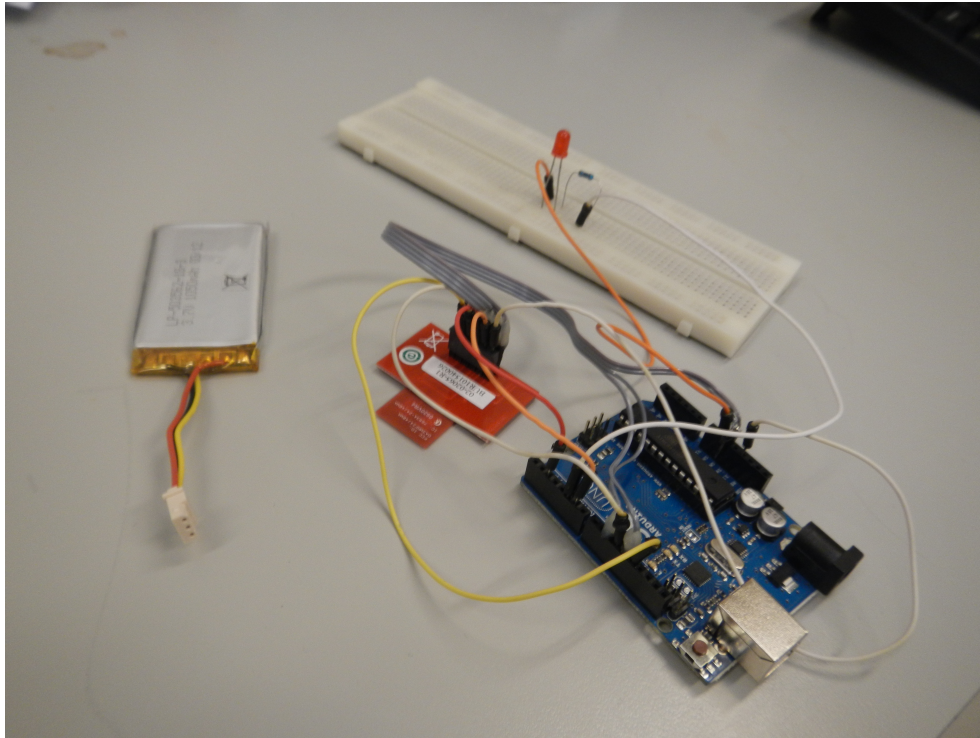
Dit verschil heeft invloed op de jitter van de harttoon detectie. Wanneer immers een pakket 2 milliseconden later aankomt, kan ook pas een zelfde tijd later de harttoon detectie plaatsvinden. De detectie kan in het ergste geval zelfs 6 milliseconden later in de hartcyclus gebeuren dan in het meest gunstige geval. Wat de invloed hiervan is op de kwaliteit van de afbeelding, moet nog onderzocht worden.

Daarnaast is gekeken naar de minimale tijd die de hardware van de MRF24J40MA nodig heeft om een pakket te versturen en weer gereed te maken voor het volgende pakket. Dit is getest door zonder vertragingen de hardware pakketten te laten versturen. De test is tien keer gedaan, het gemiddelde van de testresultaten is afgebeeld in figuur 8.5. Pakketten met een grootte van 40 byte kunnen bijvoorbeeld om de 18 milliseconden verstuurd worden.

Het grootste probleem voor topologie 1 wordt met deze figuur meteen zichtbaar. Indien een sensormeting met een grootte van 1 byte gedigitaliseerd en direct verzonden wordt, moet er gemiddeld 12.5 milliseconden worden gewacht totdat opnieuw een meting kan worden verstuurd. De frequentie waarmee gemeten kan worden wordt hiermee drastisch verlaagd tot 80 Hz! Voor een meting die met een frequentie van 250 Hz gedaan wordt, moeten eerst 4 waarden gedigitaliseerd en opgeslagen worden voordat ze kunnen worden verstuurd.

Indien een tweede biomedisch signaal, zoals een ECG, ook verstuurd moet worden, zal de grootte van de datapakketten steeds groter worden. Omdat de metingen eerst gedigitaliseerd en opgeslagen moeten worden, wordt de vertraging in de transmissie steeds groter. Weergave van de signalen wordt hiermee minder *real-time*.

Uit figuur 8.5 kan worden afgeleid wat de maximale datasnelheid is bij een gegeven grootte van een datapakket. Hoewel de datasheet aangeeft dat de maximale datasnelheid 250 kbit/s bedraagt [32], blijkt deze in de praktijk een stuk lager te liggen. Uit figuur 8.6 kan worden afgeleid dat deze ongeveer 4 kb/s is, wat neer komt op 32 kbit/s. Deze snelheid kan alleen gehaald worden wanneer datapakketten van meer dan

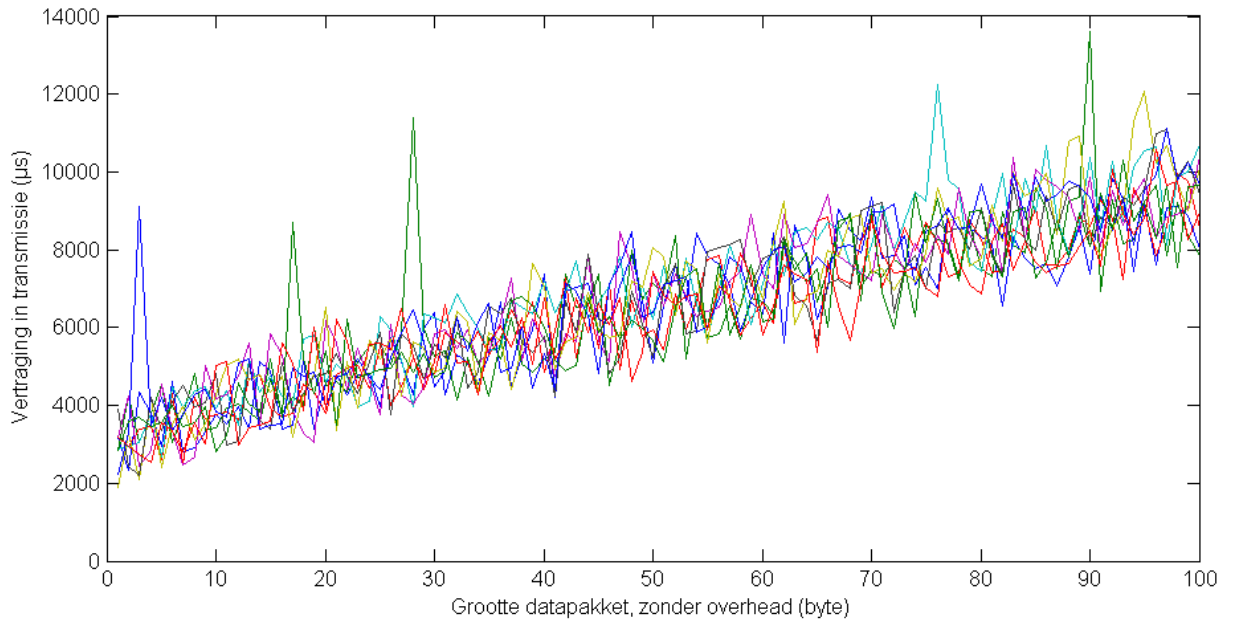


Figuur 8.2: De ZigBee module, Arduino en het LEDje aan de kant van de MRI-scanner

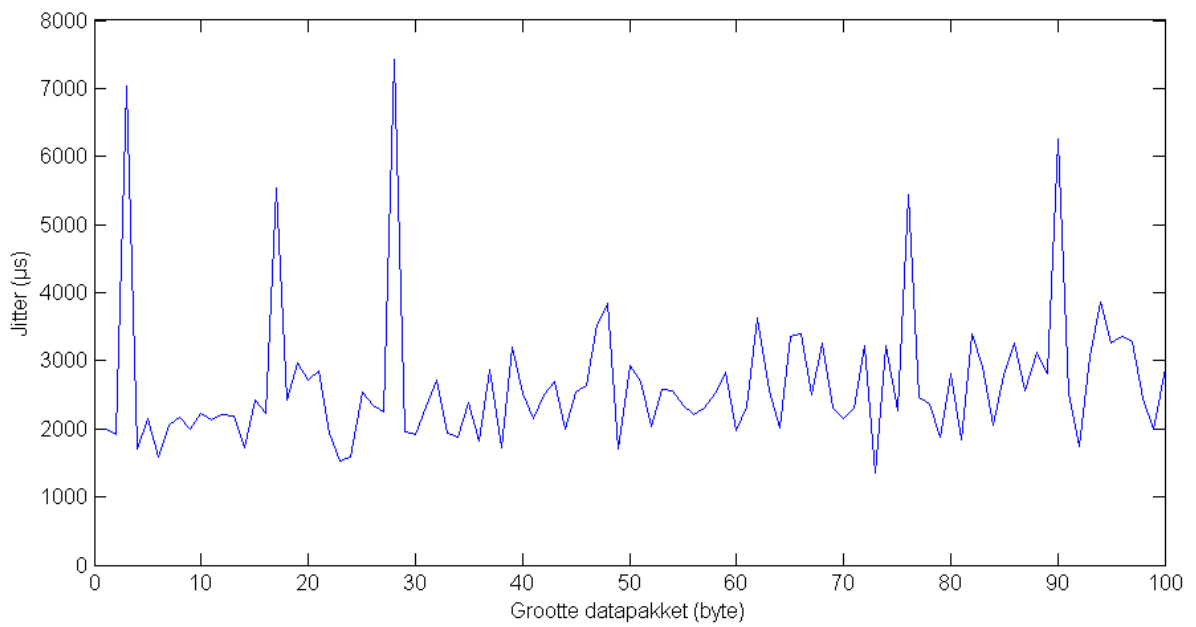
100 bytes worden gebruikt.

8.3 Eindresultaten

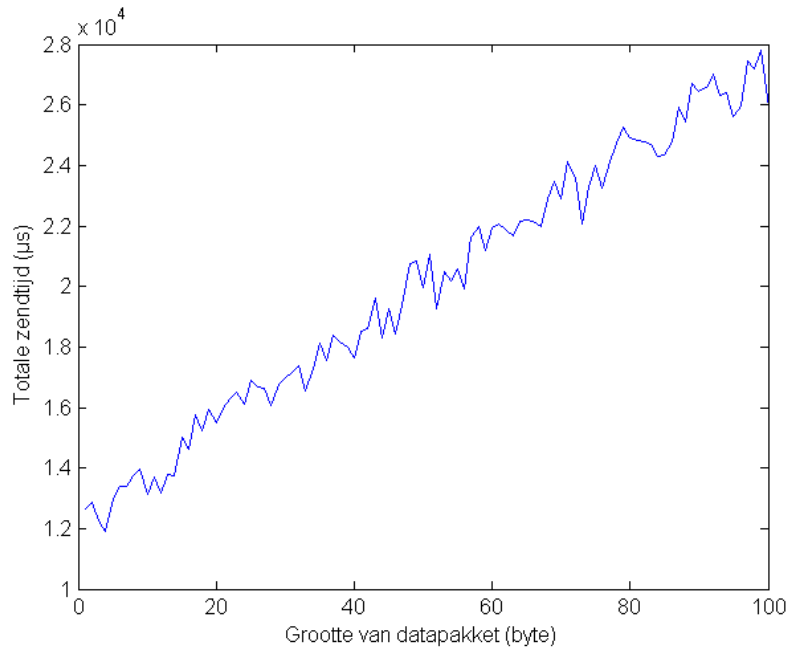
Het LEDje knippert mee op de trigger die wordt weergegeven in de GUI. De GUI is weergegeven in figuur 8.7. In de GUI is ook te zien dat er een temperatuur wordt gemeten. De temperatuur is een random gegenereerde waarde die vanaf de ZigBee module aan de kant van de sensor naar de andere ZigBee module verstuurd. Deze waarde wordt daarna weer weergegeven op de computer.



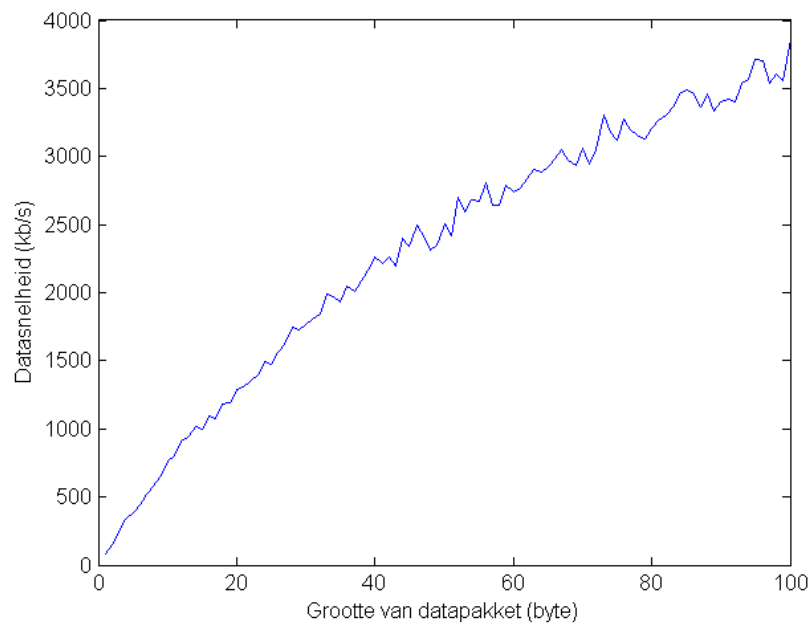
Figuur 8.3: Vertraging in transmissie



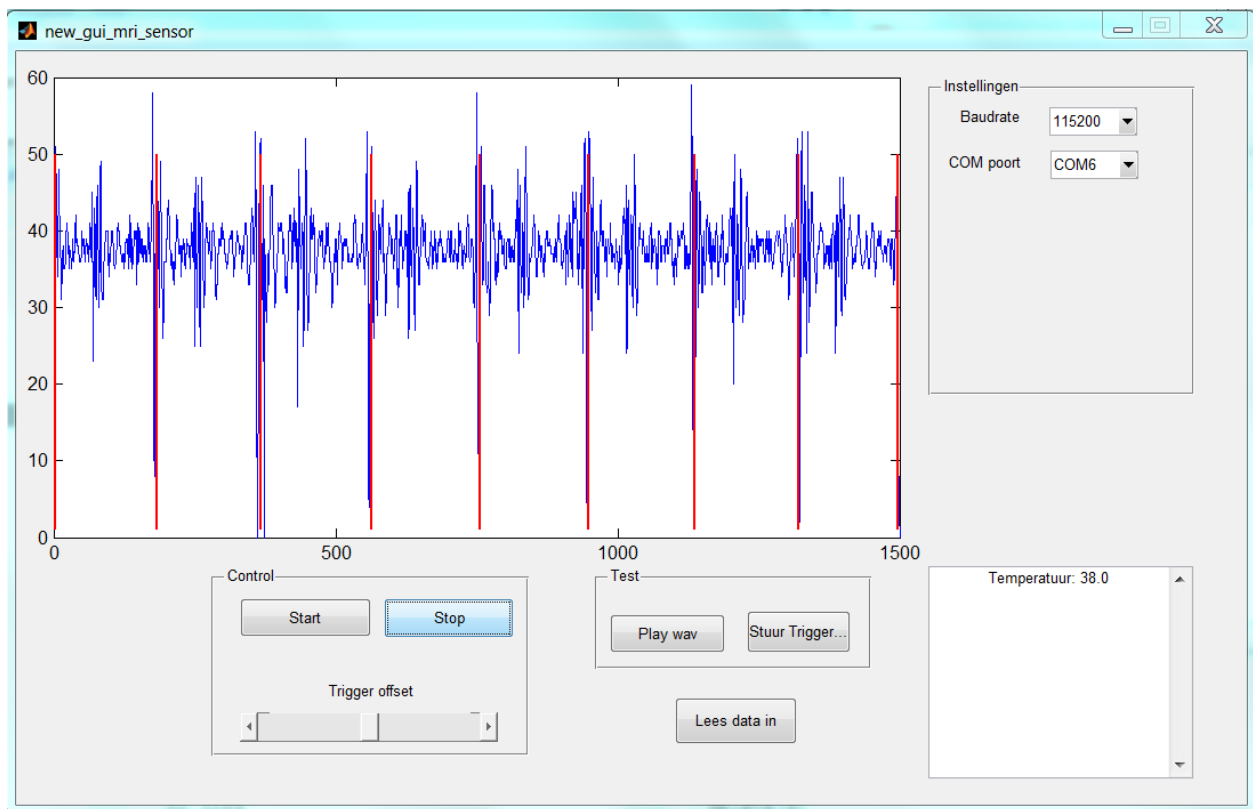
Figuur 8.4: Jitter in transmissie



Figuur 8.5: Transmissietijd voor hardware



Figuur 8.6: Maximale dat snelheid bij verschillende pakketgrootte



Figuur 8.7: De Graphical user interface (GUI) in Matlab

Hoofdstuk 9

Conclusies en aanbevelingen

In dit hoofdstuk worden de conclusies besproken in de eerste paragraaf. Verder wordt er in de tweede paragraaf behandeld wat er nog moet gebeuren om het project volledig aan het programma van eisen te laten voldoen.

9.1 Conclusies

Het doel van deze thesis is om te onderzoeken wat de beste manier is om een draadloos communicatiesysteem op te zetten in een MRI-omgeving. De componenten van de zender mogen geen ferromagnetische materialen bevatten die sterk aangetrokken worden door de 7 T magneet. De gebruiker van het systeem moet de biomedische signalen van een patiënt, zoals de akoestische hartslag, temperatuur en Electrocardiogram op een computerscherm kunnen zien. Het uiteindelijke doel was om de kwaliteit van de afbeeldingen uit de MRI-scanner te verbeteren.

Het is mogelijk om een draadloos communicatiesysteem te ontwikkelen voor een MRI-scanner. In dit project zijn modules gekozen die een minimale hoeveelheid aan ferromagnetische materialen bevatten en daardoor niet worden aangetrokken door de magneet in de scanner.

De testen buiten de MRI-scanner zijn goed verlopen. Het uiteindelijke doel om de drie deelsystemen aan elkaar te koppelen zodat er één eindproduct ontstaat, bestaande uit een sensor, een signaalbewerkingsmodule en een transmissiemodule, is helaas niet gelukt.

Het weergeven van het temperatuursignaal en akoestische signaal inclusief de plek waar er wordt getriggerd in de GUI is wel gelukt. De signalen zijn hierbij echter niet real-time opgenomen door de sensor, maar van te voren opgenomen.

Aan de kant van de sensor is het gelukt om uit het van de voren opgenomen signaal een trigger te bepalen met behulp van de Arduino. De Arduino laat een LED knipperen op het moment dat er getriggerd moet worden.

De jitter bij topologie 2 is klein genoeg om een scherpe afbeelding te maken. Bij topologie 1 is dit niet het geval.

9.2 Aanbevelingen

Het deelproduct transmissie werkt naar behoren buiten de MRI-scanner. Om het product compleet te maken moeten er nog een aantal dingen gedaan worden:

- de drie deelsystemen aan elkaar koppelen;
Dit zal ervoor zorgen dat er getest kan worden of het systeem ook realtime werkt met een echt signaal, dus niet een van te voren opgenomen signaal zoals dat nu gebeurt. Ook kan er zo gekeken worden of de aansluitingen tussen het signaalbewerkingsgedeelte en het transmissiegedeelte goed gaan.

- de batterij aansluiten op het systeem;
Hierdoor kan er gekeken worden of het systeem het doet op een batterij en kan er getest worden hoe lang het systeem het volhoudt op de batterij. Uit de datasheet volgt dat als alleen de transmissiemodule van de accu gebruik zou maken de accu het 45 uur zou volhouden. Ook kan er dan gekeken worden of het systeem aan de kant van de sensor (inclusief batterij) voldoet aan het plan van eisen (kleiner is dan 10 x 10 x 5 cm). Ook kan er dan gekeken worden of het systeem zwaarder of minder zwaar is dan 0,5 kg.
- het totale systeem testen in de MRI-scanner;
Als aan de vorige twee eisen voldaan zijn kan het totale systeem getest worden in de MRI-scanner. Dan zou er gekeken kunnen worden wat de MRI-scanner doet met de signalen en of het systeem bestand is tegen 300 MHz radiofrequente golven, het statische magnetische veld van de MRI-scanner, laagfrequente magnetisch veld en geluidsgolven. Ook moet dan nog de temperatuur van de behuizing van het systeem aan de kant van de sensor worden gemeten of het niet boven de 25 graden Celsius uitkomt. Indien de radiofrequente golven voor te veel storing zorgen, kan gekeken worden naar een koperen shield om de module af te schermen.
- de trigger aansluiten op de MRI-scanner;
De uitgang van het prototype is aangesloten op een LED. Met een pulsformigsignaal wordt aangegeven dat de MRI-scanner een scan moet maken. Uiteindelijk zal de uitgang aangesloten moeten worden op de MRI-scanner.
- De drempelwaarde waardoor er getriggerd is per persoon verschillend. Er zou onderzocht moeten worden of de analysetechnieken in paragraaf 2.3 van de literatuurstudie beter voldoen voor alle patiënten.

Als al deze aanbevelingen zijn doorgevoerd zal het systeem voldoen aan het programma van eisen.

Bibliografie

- [1] T. Niendorf *et al.*, “Toward cardiovascular mri at 7 t: clinical needs, technical solutions and research promises,” *European Radiology*, vol. 20, pp. 2806–2816, 2010. 10.1007/s00330-010-1902-8.
- [2] T. Frauenrath *et al.*, “Acoustic cardiac triggering: a practical solution for synchronization and gating of cardiovascular magnetic resonance at 7 tesla,” *Journal of Cardiovascular Magnetic Resonance*, vol. 12, no. 1, p. 67, 2010.
- [3] F. G. Shellock and E. Kanal, “Burns associated with the use of monitoring equipment during mr procedures,” *Journal of Magnetic Resonance Imaging*, vol. 6, no. 1, pp. 271–272, 1996.
- [4] A. Gupta *et al.*, “Simulation of elevated t-waves of an ecg inside a static magnetic field (mri),” *Biomedical Engineering, IEEE Transactions on*, vol. 55, pp. 1890–1896, july 2008.
- [5] W. Hudson *et al.*, *Transistor performance in intense magnetic fields*. No. v. 5428 in NASA technical note, National Aeronautics and Space Administration, 1969.
- [6] C. Lopes and P. Aguiar, “Aerial acoustic communications,” in *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*, pp. 219–222, 2001.
- [7] J. Zhang *et al.*, “Acoustic communication in wireless sensor networks.”
- [8] J. Kahn and J. Barry, “Wireless infrared communications,” *Proceedings of the IEEE*, vol. 85, pp. 265–298, feb 1997.
- [9] P. Barker and A. Boucouvalas, “Performance comparison of the ieee 802.11 and air infrared wireless mac protocols,” *Proc. of the 2nd Annual Symposium on the Convergence of Telecommunications, Networking and Broadcasting*, pp. 113–118, june 2001.
- [10] J.-S. Lee *et al.*, “A comparative study of wireless protocols: Bluetooth, uwb, zigbee, and wi-fi,” in *Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE*, pp. 46–51, nov. 2007.
- [11] G. Ferrari *et al.*, “Wireless sensor networks: performance analysis in indoor scenarios,” *EURASIP J. Wirel. Commun. Netw.*
- [12] S.-K. Chen *et al.*, “A reliable transmission protocol for zigbee-based wireless patient monitoring,” *Information Technology in Biomedicine, IEEE Transactions on*, vol. 16, pp. 6–16, jan. 2012.
- [13] P. Erne, “Beyond auscultation—acoustic cardiography in the diagnosis and assessment of cardiac disease.,” *Swiss Med Wkly*, vol. 138, no. 31-32, pp. 439–52, 2008.
- [14] A. Djebbari and F. Bereksi Reguig, “Short-time fourier transform analysis of the phonocardiogram signal,” in *Electronics, Circuits and Systems, 2000. ICECS 2000. The 7th IEEE International Conference on*, vol. 2, pp. 844–847 vol.2, 2000.

-
- [15] H. Nazeran, "Wavelet-based segmentation and feature extraction of heart sounds for intelligentpda-based phonocardiography," *Methods Inf. Med*, vol. 46, 2007.
- [16] Y. Lee and M. Moghavvemi, "Remote heart rate monitoring system based on phonocardiography," in *Research and Development, 2002. SCORed 2002. Student Conference on*, pp. 27 – 30, 2002.
- [17] K. Coyne, "MRI: a guided tour." <http://www.magnet.fsu.edu/education/tutorials/magnetacademy/mri/>, 2012. [Online; accessed -4-6-2012].
- [18] G. ziekenhuizen, "MRI-scan." <http://www.gelreziekenhuizen.nl/internet/bestanden/afdelingen/radi-a/xmr.jpg>, 2012. [Online; accessed -4-6-2012].
- [19] Vishays Semiconductors, *TFDU4101 Data Sheet*.
- [20] Waitrony, *PIC-1018SCL Datasheet*.
- [21] C. Li *et al.*, "Short-range ultrasonic digital communications in air," *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, vol. 55, pp. 908 –918, april 2008.
- [22] C. Li *et al.*, "Short-range ultrasonic communications in air using quadrature modulation," *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, vol. 56, pp. 2060 –2072, october 2009.
- [23] K. Mizutani *et al.*, "Acoustic communication in air using differential biphase shift keying with influence of impulse response and background noise," *Japanese Journal of Applied Physics*, vol. 46, no. 7B, pp. 4541–4544, 2007.
- [24] L. W. Couch, II, *Digital and Analog Communication Systems*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 7th ed., 2007.
- [25] MDFLY, *RF-BT0417C Datasheet*.
- [26] ZeroG Wireless, *ZG100M Datasheet*.
- [27] B. Razavi *et al.*, "A uwb cmos transceiver," *Solid-State Circuits, IEEE Journal of*, vol. 40, pp. 2555 – 2562, dec. 2005.
- [28] T. Nakagawa *et al.*, "1-cc computer using uwb-ir for wireless sensor network," in *Design Automation Conference, 2008. ASPDAC 2008. Asia and South Pacific*, pp. 392 –397, march 2008.
- [29] "Ieee standard for long wavelength wireless network protocol," *IEEE Std 1902.1-2009*, pp. 1 –25, 31 2009.
- [30] Zensys, *ZM3102N Z-Wave Module Datasheet*.
- [31] P. Van Mieghem, *Data Communications Networking*. Amsterdam, The Netherlands: Techne Press, 2nd ed., 2011.
- [32] Microchip Technology Inc., *MRF24J40 Datasheet*.

Bijlage A

Instelregisters van de ZigBee

De MRF24J40MA kan op vele manieren geconfigureerd worden. De module zou niet alleen in een point-to-point verbinding gebruikt kunnen worden, maar ook in een netwerkconfiguratie. Ook kan het frequentiegebied worden ingesteld, of kan de maximale datasnelheid worden bepaald. Om de module in te stellen zodat hij voldoet aan de gestelde eisen, kunnen specifieke registers in het geheugen van de MRF24J40MA ingesteld worden. Omdat er een bidirectionele link opgesteld moet worden, moet iedere module kunnen dienen als zender én ontvanger.

De registers zijn onderverdeeld in korte adressen en lange adressen. Ieder register correspondeert met een adres in het geheugen, het register TXMCR zit bijvoorbeeld op adres 0x11. De registers die belangrijk zijn voor de werking van het systeem zijn hieronder beschreven.

In het register TXMCR kan de werking van CSMA aangepast worden. CSMA kan aan- of uitgezet worden en het minimale en maximale aantal pogingen om een onbezet kanaal te detecteren kan ingesteld worden. Het minimale aantal pogingen moet groter zijn dan 0, anders zal de module geen CSMA gebruiken.

Met INTCON kunnen interrupts worden aangezet. Wanneer een pakket verstuurd of ontvangen wordt zal de MRF24J40MA een interrupt signaal genereren op een output pin. De microcontroller weet dan dat er iets succesvol is verzonden of ontvangen en kan hierop reageren.

Iedere module kan geïdentificeerd worden met zijn eigen adres. Dit is vergelijkbaar met een IP-adres van een router of het adres van een huis. Het adres wordt opgeslagen in de registers SADRL en SADRH. Deze vormen samen een 16-bits adres. Elk pakket dat verstuurd wordt, bevat het adres van de ontvanger. Een module zal een pakket dat niet voor hem bestemd is niet verder afhandelen. Het is dus belangrijk dat beide modules op het goede adres worden ingesteld.

In het register TXNCON kunnen zeer belangrijke instellingen veranderd worden. Indien TXNACKREQ op '1' wordt gezet, moet een ontvangstbevestiging (ACK) verstuurd worden door de ontvanger. Als de ontvangstbevestiging niet ontvangen wordt, zal de zender het bericht opnieuw moeten verzenden. Met TXNSECEN kan een bericht al dan niet beveiligd worden volgens de IEEE 802.15.3-2003 standaard. De hardware zal er dan voor zorgen dat het bericht versleuteld wordt. Aangezien versleuteling er voor zorgt dat de verzonden pakket nog groter wordt, zal de transmissietijd ook groter worden. Omdat een MRI-ruimte is afgeschermd voor RF interferentie, zullen gebruikers buiten de ruimte niet de berichten kunnen onderscheppen. Versleuteling is dus niet noodzakelijk.

Met RFCONF0 kan het frequentiekanaal worden ingesteld. Het is van groot belang dat beide modules op hetzelfde kanaal worden ingesteld voor correcte werking. Een volledig overzicht van alle registers kan bekeken worden in de datasheet van de MRF24J40MA [32].

Bijlage B

MATLAB-code

```
function varargout = new_gui_mri_sensor(varargin)
% NEW_GUI_MRI_SENSOR MATLAB code for new_gui_mri_sensor.fig
% NEW_GUI_MRI_SENSOR, by itself, creates a new NEW_GUI_MRI_SENSOR or raises the existing
% singleton*.
%
% H = NEW_GUI_MRI_SENSOR returns the handle to a new NEW_GUI_MRI_SENSOR or the handle to
% the existing singleton*.
%
% NEW_GUI_MRI_SENSOR('CALLBACK',hObject,eventData,handles,...) calls the local
% function named CALLBACK in NEW_GUI_MRI_SENSOR.M with the given input arguments.
%
% NEW_GUI_MRI_SENSOR('Property','Value',...) creates a new NEW_GUI_MRI_SENSOR or raises the
% existing singleton*. Starting from the left, property value pairs are
% applied to the GUI before new_gui_mri_sensor_OpeningFcn gets called. An
% unrecognized property name or invalid value makes property application
% stop. All inputs are passed to new_gui_mri_sensor_OpeningFcn via varargin.
%
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
% instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help new_gui_mri_sensor

% Last Modified by GUIDE v2.5 31-May-2012 14:59:44

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @new_gui_mri_sensor_OpeningFcn, ...
    'gui_OutputFcn', @new_gui_mri_sensor_OutputFcn, ...
    'gui_LayoutFcn', [] , ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

```

% — Executes just before new_gui_mri_sensor is made visible.
function new_gui_mri_sensor_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% varargin command line arguments to new_gui_mri_sensor (see VARARGIN)

% Choose default command line output for new_gui_mri_sensor
handles.output = hObject;
global record;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes new_gui_mri_sensor wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% — Outputs from this function are returned to the command line.
function varargout = new_gui_mri_sensor_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% — Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject handle to slider1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
% get(hObject,'Min') and get(hObject,'Max') to determine range of slider

% — Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject handle to slider1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% — Executes on selection change in listbox1.
function listbox1_Callback(hObject, eventdata, handles)
% hObject handle to listbox1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns listbox1 contents as cell array
% contents{get(hObject,'Value')} returns selected item from listbox1

```

```

% — Executes during object creation, after setting all properties.
function listBox1.CreateFcn(hObject, eventdata, handles)
% hObject handle to listBox1 (see GCBO)
% eventdata reserved – to be defined in a future version of MATLAB
% handles empty – handles not created until after all CreateFcns called

% Hint: listBox controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% — Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton1 (see GCBO)
% eventdata reserved – to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global record;
record = 1; % set the record flag to true
global s;

global trigger_list;
trigger_list = [];

handles.ak = []
global ecg_buffSize;
ecg_buffSize = 1500;
global acg_buffSize;
acg_buffSize= 1500;
global temp_buffSize;
temp_buffSize = 1500;

com_list = get(handles.popupmenu2,'String') % pak de lijst van com poorten
com_val = get(handles.popupmenu2,'Value') % pak het geselecteerde nummer in de lijst
com_port = com_list{com_val}; % com poort uit de lijst halen, met het geselecteerde nummer

baud_rate_list = get(handles.popupmenu1,'String') % pak de lijst van baudrates
selected = get(handles.popupmenu1,'Value')
selected_baudrate = str2num(baud_rate_list{selected}) % vertaal baudrate naar integer

close_ports(com_port)

s = serial(com_port); % initialiseert com poort
s.InputBufferSize = 10000; % zet aantal bytes dat in de buffer kan worden opgeslagen
s.BaudRate = selected_baudrate; % baudrate op geselecteerde waarde
fopen(s); % opent de com poort
global ecg_buffer;
global ecg_recvBytes;
global acg_recvBytes;
global temp_recvBytes;

ecg_buffer = zeros(1,ecg_buffSize);
acg_buffer = zeros(1,acg_buffSize);
temp_buffer = zeros(1,temp_buffSize);
ecg_recvBytes = 1;
acg_recvBytes = 1;
temp_recvBytes = 1;

axes(handles.axes1)

% send the ms offset
fwrite(s,[1 0 30]);

```

```

awaitBuffer(s,100)

while(record == 1)
    numBytes = s.BytesAvailable;
    data = fread(s, 2);
    type = data(1);
    len = data(2);
    if type < 0 || type > 4 % wrong packet
        continue
    end

    % len = 1000;

    clear raw_data;
    waitBufferInput(s,len);
    raw_data = fread(s, len);
    switch type
        case 1 %akoestisch
            parse.ACG(handles,hObject,raw_data,len)
        case 2 %temperatuur
            parse.TEMP(handles,hObject,raw_data,len)
        case 3 %ecg
            parse.ECG(handles,hObject,raw_data,len)
        otherwise
            warning('Unexpected packet received');
        end

end

function waitBufferInput(serial_object,size)
while serial_object.BytesAvailable <= size

end

function awaitBuffer(serial_object,size)
% first read 500 bytes
while(1)
    waitBufferInput(serial_object,size)

    % look for valid start
    wait_data_buffer = fread(serial_object,500)
    l = length(wait_data_buffer)
    for i=1:l
        type = wait_data_buffer(i)
        if wait_data_buffer(i) >=1 && wait_data_buffer(i) <= 4
            len = wait_data_buffer(i+1)
            data = wait_data_buffer(i+2:i+2+len-1)
            return
        end
    end
end

% — Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton2 (see GCBO)
% eventdata reserved — to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global record;
global s;
record = 0; % set the record flag to false
fclose(s);
delete(s);

```

```

function parse_ACG(handles,hObject,data,len)

function parse_TEMP(handles,hObject,data,len)
%str = get(handles.edit1, 'String')
tmp = typecast(uint8(data),'uint16');
decimaal = mod(tmp,10);
remainder = tmp / 10;
str = sprintf('%d.%d',[remainder decimaal])

b = ['Temperatuur: ',str]
set(handles.edit1,'String',b)

guidata(hObject,handles)

function parse_ECG(handles,hObject,data,len)
global ecg_buffer;
global ecg_recvBytes;
global ecg_buffSize;
global trigger_list;

% Bij meer gegevens, alles doorschuiven naar links
% Eerste resultaten zullen dus wegvallen
if len + ecg_recvBytes > ecg_buffSize
%doorschuiven
ecg_buffer(1:ecg_buffSize - len) = ecg_buffer(1+len:ecg_buffSize) ;
% nieuwe data aan toevoegen
ecg_buffer(ecg_buffSize - len:ecg_buffSize-1) = data;

% alle triggers naar links doorschuiven met 5 bytes
trigger_list = trigger_list - 5;
trigger_list(trigger_list < 0) = [];

if len >= 6 %% we found a trigger!

% verwijder negatieve waarden

% teken alle trigger lines

trig_position = data(6);
hor_position = ecg_buffSize - len + trig_position; % + offset
trigger_list = [ trigger_list hor_position];
end

else
ecg_buffer(ecg_recvBytes:ecg_recvBytes+len-1) = data;
ecg_recvBytes = ecg_recvBytes + len
if len >= 6
trig_position = data(6);
hor_position = ecg_recvBytes -len + trig_position; % + offset
trigger_list = [ trigger_list hor_position];
end
end
axes(handles.axes1)
axis([0 ecg_buffSize 0 100])
plot(ecg_buffer)
for i=1:length(trigger_list)
x = trigger_list(i)

```

```

l1 = line([x x],[1 50])
set(l1,'Color','red');
set(l1,'LineWidth',1.5);
end
guidata(hObject,handles)

% — Executes on selection change in listBox2.
function listBox2_Callback(hObject, eventdata, handles)
% hObject handle to listBox2 (see GCBO)
% eventdata reserved – to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns listBox2 contents as cell array
% contents{get(hObject,'Value')} returns selected item from listBox2

% — Executes during object creation, after setting all properties.
function listBox2_CreateFcn(hObject, eventdata, handles)
% hObject handle to listBox2 (see GCBO)
% eventdata reserved – to be defined in a future version of MATLAB
% handles empty – handles not created until after all CreateFcns called

% Hint: listBox controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% — Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton1 (see GCBO)
% eventdata reserved – to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global ecg_buffer;
soundsc(ecg_buffer,1000);

function edit1_Callback(hObject, eventdata, handles)
% hObject handle to edit1 (see GCBO)
% eventdata reserved – to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
% str2double(get(hObject,'String')) returns contents of edit1 as a double

% — Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit1 (see GCBO)
% eventdata reserved – to be defined in a future version of MATLAB
% handles empty – handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% — Executes on button press in pushbutton6. % TRIGGER
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton6 (see GCBO)

```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global s;
fwrite(s, [2]);

% — Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
% hObject handle to popupmenu1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu1 contents as cell array
% contents{get(hObject,'Value')} returns selected item from popupmenu1

% — Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
% hObject handle to popupmenu1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% — Executes on selection change in popupmenu2.
function popupmenu2_Callback(hObject, eventdata, handles)
% hObject handle to popupmenu2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu2 contents as cell array
% contents{get(hObject,'Value')} returns selected item from popupmenu2

% — Executes during object creation, after setting all properties.
function popupmenu2_CreateFcn(hObject, eventdata, handles)
% hObject handle to popupmenu2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% — Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton7 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global s;
% we assume s is opened
fwrite(s, [1]);
pause(0.001);
data = fread(s,s.BytesAvailable)

```


Bijlage C

Code Arduino Zigbee 1

```
/*
int resetPin = 6; // digital 6
int interruptPin = 2;
int mosiPin = 11;
int sckPin = 13;
int csPin = 10;
int misoPin = 12;
*/

/*ZIGBEE ZENDER BIJ SENSOR*/

#include <mrf24j.h>
#include <SPI.h>
#include <signal.h>
#include <MsTimer2.h>

const int pin_reset = 6;
const int pin_cs = 10; // default CS pin on ATmega8/168/328
const int pin_interrupt = 2; // default interrupt pin on ATmega8/168/328
const int ledPin = 5;

unsigned int sig_counter = 0;
unsigned int temp_counter = 380;
uint8_t seq = 1;

unsigned int ms_wait = 50;
//int STATE = 1;

Mrf24j mrf(pin_reset, pin_cs, pin_interrupt);

void setup() {
  Serial.begin(9600);

  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);

  mrf.reset();
  mrf.init();

  mrf.set_pan(0xcafe);
  // This is our address
  mrf.address16_write(0x6001);
}
```

```

mrf.set_promiscuous(0);
//mrf.set_palna(true);

mrf.set_bufferPHY(true);

attachInterrupt(0, interrupt_routine, CHANGE); // interrupt 0 equivalent to pin 2(INT0) on ATmega8/168/328

MsTimer2::set(ms_wait,timerIsr);

interrupts();
}

void blink_led(){
digitalWrite(ledPin, digitalRead(ledPin) ^1);
/* digitalWrite(ledPin,HIGH);
delay(15);
digitalWrite(ledPin,LOW);*/
}

void onReceive(){
;
}

void interrupt_routine() {
mrf.interrupt_handler(&onReceive);// mrf24 object interrupt routine
}

void send_temperature(unsigned int value){
char tmp_storage[4] = {0x2,0x2,value & 0xFF, value >> 8 & 0xFF};
mrf.send16(0x4202,tmp_storage,4,seq);
seq++;
}

void loop(){
static char send_buf[8];

//Serial.println("lala");

mrf.check_flags(&handle_rx, &handle_tx);

//if (STATE == 1){
if( sig_counter >= 1500 ){
sig_counter = 0;
send_temperature(temp_counter);
temp_counter++;
delay(10);
}

send_buf[0] = 3; send_buf[1] = 5; send_buf[2] = signaal[sig_counter]; send_buf[3] = signaal[sig_counter+1];
send_buf[4] = signaal[sig_counter+2];send_buf[5] = signaal[sig_counter+3];send_buf[6] = signaal[sig_counter+4];

//char d[7] = {3,5,signaal[sig_counter],signaal[sig_counter+1],signaal[sig_counter+2],signaal[sig_counter+3],sig

byte found = 0;
unsigned int j = 0;
//Serial.println("Testing");
for(j = 0; j < 5; j++){
if(send_buf[j+2] <= 15){
blink_led();
MsTimer2::start();
found = 1;
break;
}
}

```

```

}
}

if(found == 1){
send_buf[1] = 6; send_buf[7] = j;
mrf.send16(0x4202,send_buf,8,seq); //make freaking static buf!
}else{
mrf.send16(0x4202,send_buf,7,seq);
}

seq++;
delay(20);
sig_counter += 5;
// }
}
//
//
//send_temperature(5);

void handle_rx() {
  blink_led();
  //Serial.println("Ontvangen");
  //Serial.print("received a packet ");Serial.print(mrf.get_rxinfo()->frame_length, DEC);Serial.println(" bytes lo

  /*if(mrf.get_bufferPHY()){
  Serial.println("Packet data (PHY Payload):");
  for (int i = 0; i < mrf.get_rxinfo()->frame_length; i++) {
  Serial.print(mrf.get_rxbuf()[i],HEX);Serial.print(" ");
  }
  }*/

  // Serial.println("\r\nASCII data (relevant data):");
  // for (int i = 0; i < mrf.rx_datalength(); i++) {
  // Serial.write(mrf.get_rxinfo()->rx_data[i]);
  // }
  /*if(mrf.rx_datalength() == 3){
  if(mrf.get_rxinfo()->rx_data[0] == 1){
  Serial.println("Trigger offset ontvangen!");
  byte b1 = mrf.get_rxinfo()->rx_data[1];
  byte b2 = mrf.get_rxinfo()->rx_data[2];
  uint16_t offset = b1 << 8 | b2;
  Serial.println(offset,DEC);

  STATE = 1;
  }
  }*/

  /* Serial.print("\r\nLQI/RSSI=");
  Serial.print(mrf.get_rxinfo()->lqi, DEC);
  Serial.print("/");
  Serial.println(mrf.get_rxinfo()->rssi, DEC);*/
}

void handle_tx() {
  /* if (mrf.get_txinfo()->tx_ok) {
  Serial.println("TX went ok, got ack");
  } else {
  Serial.print("TX failed after ");Serial.print(mrf.get_txinfo()->retries);Serial.println(" retries\n");
  }*/
}

void timerIsr(){

```

```
MsTimer2::stop();  
blink.led();  
}
```

Bijlage D

Code Arduino ZigBee 2

```
/*
int resetPin = 6; // digital 6
int interruptPin = 2;
int mosiPin = 11;
int sckPin = 13;
int csPin = 10;
int misoPin = 12;
*/

/**
ZIGBEE ontvanger aan computer
*/

/**
 * Example code for using a microchip mrf24j40 module to send and receive
 * packets using plain 802.15.4
 * Requirements: 3 pins for spi, 3 pins for reset, chip select and interrupt
 * notifications
 * This example file is considered to be in the public domain
 * Originally written by Karl Palsson, karlp@tweak.net.au, March 2011
 */
#include <mrf24j.h>
#include <SPI.h>

const int pin_reset = 6;
const int pin_cs = 10; // default CS pin on ATmega8/168/328
const int pin_interrupt = 2; // default interrupt pin on ATmega8/168/328
const int maxBufferSize = 1000;
const int ledPin = 5;

int STATE = 1;

uint8_t lastReceivedSeq = 0;

byte recvBuffer[maxBufferSize];
long recvBufferCursor = 0;

uint8_t seq = 1;
```

```

Mrf24j mrf(pin_reset, pin_cs, pin_interrupt);

long last_time;
long tx_interval = 5000;

void setup() {
  Serial.begin(115200);

  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);

  mrf.reset();
  mrf.init();

  mrf.set_pan(0xcafe);
  // This is our address
  mrf.address16.write(0x4202);

  // uncomment if you want to receive any packet on this channel
  mrf.set_promiscuous(true);

  // uncomment if you want to enable PA/LNA external control
  //mrf.set_palna(true);

  // uncomment if you want to buffer all PHY Payload
  mrf.set_bufferPHY(true);

  attachInterrupt(0, interrupt_routine, CHANGE); // interrupt 0 equivalent to pin 2(INT0) on ATmega8/168/328
  last_time = millis();
  interrupts();
}

void blink_led(){
  digitalWrite(ledPin, digitalRead(ledPin) ^ 1 );
}

void interrupt_routine() {
  mrf.interrupt_handler(&onReceive); // mrf24 object interrupt routine
}

byte incomingByte = 0;

void loop() {

  mrf.check_flags(&handle_rx, &handle_tx);

  long time_now = millis();

  if( STATE == 0){
    if(Serial.available() > 0 ){
      incomingByte = Serial.read();
      if(incomingByte == 1){
        while(Serial.available() < 2){}; // wait for 2 bytes to receive
        byte b1 = Serial.read();
        byte b2 = Serial.read();
        uint16_t ms_offset = b1 << 8 | b2;
        char d[3] = {1, b1, b2};

        mrf.send16(0x6001, d, 3, seq);
        seq++;
        delay(10);

      }else if(incomingByte == 2){
        send_trigger();
      }
    }
  }
}

```

```

blink.led();
}else{

}
}
}

char trigger[2] = {0x04,0x00};
void send_trigger(){
  mrf.send16(0x6001,trigger,2,seq);
  seq++;
}

boolean stored;

void onReceive(){
}

void handle_rx() {
  //Serial.print("received a packet ");Serial.print(mrf.get_rxinfo()->frame.length, DEC);Serial.println(" bytes lo
  byte seqRecv = mrf.get_rxbuf()[2]; // sequence number

  if(seqRecv != lastReceivedSeq){
    lastReceivedSeq = seqRecv;

    if(STATE == 1){

      if(mrf.get_bufferPHY()){

        // Serial.print("Data length: ");Serial.println(mrf.rx.datalength());
        // Serial.println("Packet data (PHY Payload):");

        delay(8); // dont ask me why this is needed!
        /* for (int i = 0; i < mrf.get_rxinfo()->frame.length; i++) {
          Serial.print(mrf.get_rxbuf()[i],HEX);Serial.print(" ");
          //mrf.get_rxbuf()[i];
        }*/

        /*
          Serial.println("\r\nASCII data (relevant data):");
          for (int i = 0; i < mrf.rx.datalength(); i++) {
            Serial.print(mrf.get_rxinfo()->rx_data[i],HEX);Serial.print(" ");
          }
          Serial.println(""); */

      }

      int ln = mrf.rx.datalength();

      Serial.write(mrf.get_rxinfo()->rx_data,ln);
      Serial.flush();
    }

  }

  /* Serial.print("\r\nLQI/RSSI=");
  Serial.print(mrf.get_rxinfo()->lqi, DEC);
  Serial.print("/");
  Serial.println(mrf.get_rxinfo()->rssi, DEC);*/
}

```

```
void handle_tx() {  
  if (mrf.get_txinfo()->tx_ok) {  
    //Serial.println("TX went ok, got ack");  
  } else {  
    //Serial.print("TX failed after ");Serial.print(mrf.get_txinfo()->retries);Serial.println(" retries\n");  
  }  
}
```