Development of a modular low power IEEE 802.15.4 wireless sensor module

A dissertation submitted in fulfilment of the requirements for the degree

Master of Engineering

in

Computer and Electronic Engineering

at the Potchefstroom campus of the

North-West University

G.A. du Plessis

20102070

Supervisor:

Prof. A.S.J. Helberg

Co-Supervisor:

Mrs. M.J. Grobler

November 2010
Declaration

I, Gerhard Andries du Plessis hereby declare that this dissertation entitled “Development of a low power modular IEEE 802.15.4 wireless network node” is my own original work and has not already been submitted to any other university or institution for examination.

_______________________
G.A. du Plessis

Student number: 20102070

Signed on the __________ of ______________________ at ______________.
"1 That which was from the beginning, which we have heard, which we have seen with our eyes, which we have looked upon, and our hands have handled, of the Word of life;

2 For the life was manifested, and we have seen it, and bear witness, and shew unto you that eternal life, which was with the Father, and was manifested unto us;

3 That which we have seen and heard declare we unto you, that ye also may have fellowship with us: and truly our fellowship is with the Father, and with his Son Jesus Christ.

4 And these things write we unto you, that your joy may be full."

- 1 John 1:1-4
Acknowledgements

Firstly, a word of thanks to our heavenly Father for his encouragement and close comfort, for seeing me through these tough and difficult times, and for always giving me a place of rest and salvation – to this I am forever grateful.

Thank you, to all the following people who had directly influenced the outcome and success of this project:

- *Prof. A.S.J. Helberg*, my primary supervisor, for your expert advice, analysis and feedback.
- *My family: Casper sr., Gerda and Casper jr.*: Thank you for your love and support through these times and for always making sure I have full plate each day.
- *Mrs. M.J. Grobler*, my co-supervisor, for all your arrangements and time to make our research group a pleasant place to be in.
- *Prof. JEW Holm*, lecturer of advanced electronic development at the NWU. A special thanks to you for opening up a new path in my life.
- *Adriaan Labuschagne*, the client of this research. I show gratitude to you for sponsoring this research and giving me the opportunity to successfully complete this cycle of my life.
- To all my friends I give a special word of thank because it was you who made these past few years an extraordinary experience.
- The *TeleNet research group* for making research exciting.
Abstract

ZaPOP is a specialist in the in-store marketing arena. They are in the process of broadening their marketing capabilities with a new system for which they require the development of three very specific electronic devices.

These three electronic devices need to operate within the existing business structure of the client. To develop these devices we have to use the correct design and development strategy, but this was a problem for us because we had little to none experience in advanced electronic development. This is the development of electronic devices for a client with a specific need and objective and that need to operate in the existing structure of the client’s company. Our problem was to find a suitable methodology for this development, develop the devices according to this methodology and to verify and validate the development.

At the start of this research and development, we first set out to find a scientific methodology for this development. In attending a course in advance electronic development and through literature studies we decided to base our development on systems engineering principles and concepts. These principles and concepts were used in a specific manner to develop the electronic devices for our client. The methodology and development are presented in this dissertation, as well as the validation and verifications aspects thereof.

Systems engineering development allowed us to move from client need and concept, to a product of high quality in a gradual and complete manner. It is gradual because we could focus on the fit, function and form aspects of the devices at different times, and yet allow them to intervene on each other through reviews and syntheses. All possible external entities, interfaces, factors and life-cycle outcomes were included into the design, making it a complete development.

This approach worked, because the devices needed to be able to operate in the structure of our client’s company and the devices were designed for that purpose from the start of the development. Systems engineering allowed us to design and develop devices that are functionally capable of delivering on the needs of the client. The devices were designed to be reliable, efficiency, maintainable and user friendly, as a result of this complete approach.
Opsomming

ZaPOP is ’n spesialis van bemarking binne winkels. Hulle is tans in die proses om die bemarking geleenthede wat hulle bied uit die brei met die bekendstelling van ’n nuwe bemarkingstelsel. Om hierdie stelsel te realiser moet drie spesifieke elektroniese toestelle ontwikkel word.

Die drie toestelle moet binne die huidige besigheidsstruktuur van die klënt werk en om dit te realiseer moet die korrekte ontwikkeling strategie gebruik word. Hier het ons n probleem gehad, aangesien ons weile of min onderving het in die ontwikkeling van produkte vir ’n spesifieke doel en wat in die huidige struktuur van ’n klënt se besigheid moet pas. Ons het dus ’n probleem geïdentifiseer wat behels die soek na ’n gepaste wetenskaplike produk ontwikkelings metode, die toepassing daarvan op ons ontwikkeling en om die metode sowel as eindprodukt te valideer en te verifieer.

Die eerste stap wat ons gedoen het om ’n ontwikkelings metode te vind was om ’n vak te loop in gevorderde produkontwikkeling. Met die agtergrond wat ons verkry het deur hierdie vak te loop asook verder literatuur navorsing het ons besluit om die ontwikkelings konsepte en beginsels te gebruik wat stelselsingenieurswese ons bied. Die konsepte en beginsels wat ons nagevors het is op ’n spesifieke maëier en volgorde toegepas op ons ontwikkeling. In hierdie verhandeling word die ontwikkelings metodiek, die ontwikkeling self en die validasie en verifikasie daarvan gegee.

Dit is was moontlik gewees om van doel, na konsep, na eindprodukt te beweeg in ’n trapgewyse en komplette manier deur gebruik te maak van stelselsingenieurswese. Dit is trapgewys omdat ons op verkillende tye kon fokus op die passing, funksie en vorm van die drie toestelle, maar ook omdat dit op mekaar ’n invloed hê d.m.v. ontwerphersienings en sintesis. Die komplette ontwikkeling het ons toegelaat om alle eksterne objekte, intervlakke, faktore en lewensklikte uitkomste in te sluit in die ontwerp.

Hierdie benadering het gewerk omdat ons toestelle kon ontwikkel wat in die besigheidsstruktuur van ons klënt kan funksioneer. Die toestelle is ontwerp om funksioneel daartoe in staat te wees om aan die klënt se behoeftes te voldoen. Deur van hierdie kompleet benadering gebruik te maak kon ons produkte ontwikkel wat hoog betroubaar, effektief, onderhoubaar en gebruikersvriendelik is.
# Table of Contents

Declaration................................................................................................................. i

Acknowledgements........................................................................................................ iii

Abstract......................................................................................................................... iv

Opsomming ..................................................................................................................... v

Table of Contents ........................................................................................................... vi

List of Figures ................................................................................................................... ix

List of Tables .................................................................................................................... xiii

Nomenclature .................................................................................................................. xv

List of Abbreviations ....................................................................................................... xv

Chapter 1 ....................................................................................................................... 1-1

1.1 Background.............................................................................................................. 1-1

1.2 Problem Discussion ............................................................................................... 1-3

1.3 Research .................................................................................................................. 1-4

1.4 Dissertation Outline ............................................................................................... 1-7

1.5 Conclusion .............................................................................................................. 1-7

Chapter 2 ....................................................................................................................... 2-1

2.1 Introduction ............................................................................................................ 2-1

2.2 Systems Engineering Management ....................................................................... 2-2

2.3 Technical Development ......................................................................................... 2-4

2.4 SE Aspects and Concepts not Used .................................................................... 2-9

2.5 Conclusion .............................................................................................................. 2-10

Chapter 3 ...................................................................................................................... 3-1

3.1 Introduction ............................................................................................................ 3-1
Table of Contents

3.2 Complete Process Model ................................................................. 3-1
3.3 Process Model Sections ................................................................. 3-3
3.4 Conclusion .................................................................................. 3-8

Chapter 4 ........................................................................................... 4-1
4.1 Introduction ................................................................................ 4-1
4.2 Literature Study and Analysis ..................................................... 4-2
4.3 Requirements Analysis ............................................................ 4-6
4.4 Functional Analysis .................................................................. 4-7
4.5 Resource and Requirements Allocation .................................... 4-10
4.6 Design Synthesis ....................................................................... 4-12
4.7 Design Review and Feedback .................................................. 4-13
4.8 Conclusion ................................................................................ 4-13

Chapter 5 ........................................................................................... 5-1
5.1 Introduction ................................................................................ 5-1
5.2 Conceptual Design Output Analysis ......................................... 5-2
5.3 Literature Study and Analysis .................................................. 5-2
5.4 Requirements Analysis ............................................................ 5-18
5.5 Functional Analysis .................................................................. 5-20
5.6 Resource and Requirements Allocation .................................... 5-27
5.7 Trade-Off Analysis .................................................................... 5-28
5.8 Design Synthesis ....................................................................... 5-38
5.9 Design Review and Feedback .................................................. 5-41
5.10 Conclusion ................................................................................ 5-42

Chapter 6 ........................................................................................... 6-1
6.1 Introduction ................................................................................ 6-1
# Table of Contents

6.2 Preliminary Design Output Analysis ................................................................. 6-2
6.3 Detailed Functional Analysis............................................................................... 6-3
6.4 Model Design...................................................................................................... 6-26
6.5 Model Testing ...................................................................................................... 6-30
6.6 Design Review.................................................................................................... 6-41
6.7 Conclusion........................................................................................................... 6-41

Chapter 7 ...................................................................................................................... 7-1
7.1 Introduction .......................................................................................................... 7-1
7.2 Analysis................................................................................................................ 7-1
7.3 Methodology Verification .................................................................................... 7-2
7.4 Product Validation ............................................................................................... 7-5
7.5 Overall.................................................................................................................. 7-6
7.6 Conclusion............................................................................................................ 7-6

Chapter 8 ...................................................................................................................... 8-1
8.1 Introduction .......................................................................................................... 8-1
8.2 Research Objective ............................................................................................. 8-1
8.3 Final Comments ................................................................................................... 8-6

Bibliography .................................................................................................................. 9-1

Appendixes ................................................................................................................... 10-1
10.1 Appendix A – Data CD .................................................................................... 10-1
List of Figures

Figure 1-1: System Concept ................................................................................................................. 1-2
Figure 1-2: Research. ................................................................................................................................. 1-4
Figure 2-1: Generic Model for Development Phasing. .............................................................................. 2-3
Figure 2-2: System Life-Cycle Example. ................................................................................................. 2-3
Figure 2-3: The Systems Engineering Process ......................................................................................... 2-5
Figure 3-1: Phasing Model for Development Methodology ........................................................................ 3-2
Figure 3-2: Systems Engineering Process for Development Methodology. ........................................... 3-3
Figure 3-3: Conceptual Design Described. ............................................................................................. 3-4
Figure 3-4: Preliminary Design Described. ............................................................................................ 3-5
Figure 3-5: Detail Design Explained. ...................................................................................................... 3-6
Figure 4-1: Examples of Motes. [12][13] ............................................................................................... 4-3
Figure 4-2: Difference between WSNs and DSNs. .................................................................................. 4-4
Figure 4-3: System Functional Architecture. ......................................................................................... 4-8
Figure 4-4: System Software and Firmware Architecture. ...................................................................... 4-9
Figure 4-5: Conceptual Design Synthesis. ............................................................................................. 4-12
Figure 5-1: IEEE 802.15.4 WPAN. [20] ................................................................................................. 5-4
Figure 5-2: Elements of EMI. [23] .......................................................................................................... 5-15
Figure 5-3: Operational Analysis for Programming Personnel. ............................................................... 5-20
Figure 5-4: Operational Analysis for Inspection Personnel. ................................................................. 5-21
Figure 5-5: Operational Analysis for Maintenance Personnel. ............................................................. 5-21
Figure 5-6: Operational Analysis for Installation Personnel. ............................................................... 5-22
Figure 5-7: OmnipoMote Functional Architecture. ............................................................................... 5-23
<p>| Figure 5-8 | OmnipoController Functional Architecture | 5-24 |
| Figure 5-9 | OmnipoBatteryPack Functional Architecture | 5-25 |
| Figure 5-10 | USB Cable Functional Architecture | 5-25 |
| Figure 5-11 | Power Cable Functional Architecture | 5-26 |
| Figure 5-12 | Programming Apparatus Functional Architecture | 5-26 |
| Figure 5-13 | Basic Operation of a Tilt Sensor | 5-30 |
| Figure 5-14 | FFD Data Switch Scenarios: PAN Coordinator | 5-33 |
| Figure 5-15 | FFD Data Switch Scenarios: for normal Coordinator operations | 5-33 |
| Figure 5-16 | USB Controller Conceptual Layout | 5-33 |
| Figure 5-17 | Preliminary Design Synthesis - OmnipoMote Specific Architecture | 5-38 |
| Figure 5-18 | Preliminary Design Synthesis - OmnipoController Specific Architecture | 5-39 |
| Figure 5-19 | Preliminary Design Synthesis – OmnipoBatteryPack Specific Architecture | 5-39 |
| Figure 5-20 | Preliminary Design Synthesis – USB Cable | 5-40 |
| Figure 5-21 | Preliminary Design Synthesis – Power Cable | 5-40 |
| Figure 5-22 | Preliminary Design Synthesis – Programming Apparatus | 5-40 |
| Figure 5-23 | Preliminary Design Synthesis – USB Host Software | 5-41 |
| Figure 5-24 | Preliminary Design Synthesis – Programming PC Software | 5-41 |
| Figure 6-1 | Detailed Operational Analysis – M/F a2 | 6-4 |
| Figure 6-2 | Detailed Operational Analysis – O/F a3 | 6-6 |
| Figure 6-3 | Detailed Operational Analysis – M/F b2 | 6-7 |
| Figure 6-4 | Detailed Operational Analysis – O/F b3 | 6-7 |
| Figure 6-5 | Detailed Operational Analysis – O/F d3 | 6-9 |
| Figure 6-6 | Detailed Functional Architecture – F/U 1.1.1 | 6-10 |
| Figure 6-7 | Detailed Functional Architecture – F/U 1.1.2 | 6-10 |
| Figure 6-8 | Detailed Functional Architecture – F/U 1.1.3 | 6-11 |</p>
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6-9</td>
<td>Detailed Functional Architecture – F/U 1.1.4.</td>
<td>6-12</td>
</tr>
<tr>
<td>6-10</td>
<td>Detailed Functional Architecture – F/U 1.1.7.</td>
<td>6-12</td>
</tr>
<tr>
<td>6-11</td>
<td>Detailed Functional Architecture – F/U 1.1.8.</td>
<td>6-13</td>
</tr>
<tr>
<td>6-12</td>
<td>Detailed Functional Architecture – F/U 1.2.3.</td>
<td>6-13</td>
</tr>
<tr>
<td>6-13</td>
<td>Detailed Functional Architecture – F/U 1.2.4.</td>
<td>6-14</td>
</tr>
<tr>
<td>6-14</td>
<td>Detailed Functional Architecture – F/U 1.2.5.</td>
<td>6-14</td>
</tr>
<tr>
<td>6-15</td>
<td>Detailed Functional Architecture – F/U 1.2.6.</td>
<td>6-14</td>
</tr>
<tr>
<td>6-16</td>
<td>Detailed Functional Architecture – F/U 1.2.7.</td>
<td>6-15</td>
</tr>
<tr>
<td>6-17</td>
<td>Detailed Functional Architecture – F/U 1.2.8.</td>
<td>6-16</td>
</tr>
<tr>
<td>6-18</td>
<td>Detailed Functional Architecture – F/U 1.2.9.</td>
<td>6-16</td>
</tr>
<tr>
<td>6-19</td>
<td>Detailed Functional Architecture – F/U 1.2.10.</td>
<td>6-17</td>
</tr>
<tr>
<td>6-20</td>
<td>Detailed Functional Architecture – F/U 1.2.11.</td>
<td>6-18</td>
</tr>
<tr>
<td>6-21</td>
<td>Detailed Functional Architecture – F/U 1.2.12.</td>
<td>6-18</td>
</tr>
<tr>
<td>6-22</td>
<td>Detailed Functional Architecture – F/U 1.2.13.</td>
<td>6-19</td>
</tr>
<tr>
<td>6-23</td>
<td>Detailed Functional Architecture – F/U 1.2.14.</td>
<td>6-19</td>
</tr>
<tr>
<td>6-24</td>
<td>Detailed Functional Architecture – F/U 1.3.2.</td>
<td>6-20</td>
</tr>
<tr>
<td>6-25</td>
<td>Detailed Functional Architecture – F/U 1.3.3.</td>
<td>6-20</td>
</tr>
<tr>
<td>6-26</td>
<td>Detailed Functional Architecture – F/U 1.3.5.</td>
<td>6-20</td>
</tr>
<tr>
<td>6-27</td>
<td>Detailed Functional Architecture – F/U 1.1.6 – 1st Generation.</td>
<td>6-21</td>
</tr>
<tr>
<td>6-28</td>
<td>Detailed Functional Architecture – F/U 1.2.2 – 1st Generation.</td>
<td>6-22</td>
</tr>
<tr>
<td>6-29</td>
<td>Detailed Functional Architecture – F/U 1.3.4 – 1st Generation.</td>
<td>6-23</td>
</tr>
<tr>
<td>6-30</td>
<td>Detailed Functional Architecture – F/U 1.1.6 – 2nd Generation (Part 1).</td>
<td>6-23</td>
</tr>
<tr>
<td>6-31</td>
<td>Detailed Functional Architecture – F/U 1.1.6 – 2nd Generation (Part 2).</td>
<td>6-24</td>
</tr>
<tr>
<td>6-32</td>
<td>Detailed Functional Architecture – F/U 1.2.2 – 2nd Generation (Part 1).</td>
<td>6-24</td>
</tr>
<tr>
<td>6-33</td>
<td>Detailed Functional Architecture – F/U 1.2.2 – 2nd Generation (Part 2).</td>
<td>6-25</td>
</tr>
</tbody>
</table>
List of Figures

Figure 6-34: Detailed Functional Architecture – F/U 1.3.4 – 2nd Generation.................................................. 6-25
Figure 6-35: RF unit XDM Model................................................................. 6-27
Figure 6-36: Mini-controller XDM Model...................................................... 6-27
Figure 6-37: Battery Pack XDM Model......................................................... 6-27
Figure 6-38: RF unit EDM Model................................................................. 6-29
Figure 6-39: Mini-controller EDM Model...................................................... 6-29
Figure 6-40: Battery Pack EDM Model......................................................... 6-29
Figure 6-41: Test Points on RF unit for Spectrum Analyzer Test .................. 6-34
List of Tables

Table 4-1: Features of Mote Examples. [12][13] ................................................................. 4-4
Table 4-2: Client Requirements – Conceptual Design. .............................................................. 4-7
Table 4-3: Technology Requirements – Conceptual Design. ..................................................... 4-7
Table 4-4: Feasibility Requirements – Conceptual Design. ......................................................... 4-7
Table 4-5: Functional Unit Definitions ...................................................................................... 4-9
Table 4-6: System Interface Descriptions .................................................................................. 4-10
Table 4-7: Functional Units Requirements Allocation - Conceptual Design............................... 4-11
Table 4-8: Interfaces Requirements Allocation - Conceptual Design ......................................... 4-11
Table 5-1: IEEE 802.15.4 Frequency Options. [20] ................................................................. 5-5
Table 5-2: RF Unit Specific Requirements .................................................................................. 5-18
Table 5-3: Mini-Controller Specific Requirements ..................................................................... 5-18
Table 5-4: Battery Pack Specific Requirements ......................................................................... 5-19
Table 5-5 USB Cable Specific Requirements ............................................................................ 5-19
Table 5-6 USB Host Specific Requirements .............................................................................. 5-19
Table 5-7: Power Cable Specific Requirements ......................................................................... 5-19
Table 5-8: Programming Apparatus Specific Requirements .................................................... 5-19
Table 5-9: Programming PC Software Specific Requirements ................................................ 5-19
Table 5-10 USB Flash Drive Specific Requirements .................................................................. 5-19
Table 5-11 Design Specific Requirements ................................................................................ 5-19
Table 5-12 System Operational Requirements ......................................................................... 5-20
Table 5-13: Functional Descriptions of Programming Tasks .................................................... 5-21
Table 5-14: Functional Descriptions of Inspection Tasks ......................................................... 5-21
<table>
<thead>
<tr>
<th>Table 5-15: Functional Descriptions of Maintenance Tasks</th>
<th>5-22</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 5-16: Functional Descriptions of Installation Tasks</td>
<td>5-22</td>
</tr>
<tr>
<td>Table 5-17: OmnipoMote Functional Units Definitions</td>
<td>5-23</td>
</tr>
<tr>
<td>Table 5-18: OmnipoMote Interface Descriptions</td>
<td>5-23</td>
</tr>
<tr>
<td>Table 5-19: OmnipoController Functional Units Definitions</td>
<td>5-24</td>
</tr>
<tr>
<td>Table 5-20: OmnipoController Interface Descriptions</td>
<td>5-24</td>
</tr>
<tr>
<td>Table 5-21: OmnipoBatteryPack Functional Units Definitions</td>
<td>5-25</td>
</tr>
<tr>
<td>Table 5-22: OmnipoBatteryPack Interface Descriptions</td>
<td>5-25</td>
</tr>
<tr>
<td>Table 5-23: USB Cable Functional Units Definitions</td>
<td>5-26</td>
</tr>
<tr>
<td>Table 5-24: Power Cable Functional Units Definitions</td>
<td>5-26</td>
</tr>
<tr>
<td>Table 5-25: Programming Apparatus Functional Units Definitions</td>
<td>5-27</td>
</tr>
<tr>
<td>Table 5-26: Operational Functions Requirements Allocation - Preliminary Design</td>
<td>5-27</td>
</tr>
<tr>
<td>Table 5-27: Functional Units Requirements Allocation - Preliminary Design</td>
<td>5-27</td>
</tr>
</tbody>
</table>
Nomenclature

List of Abbreviations

In order of appearance, the following abbreviations are found in this dissertation:

- RF: Radio Frequency
- USB: Universal Serial Bus
- IP: Internet Protocol
- DSN: Distributed Sensor Network
- WSN: Wireless Sensor Network
- PC: Personal Computer
- SE: Systems Engineering
- FMEA: Failure Mode and Effects Analysis
- CA: Critical Analysis
- FMECA: Failure Mode Effects and Critical Analysis
- SEP: Systems Engineering Process
- XDM: Experimental Model
- CAD: Computer Aided Design
- EDM: Engineering Model
- ADM: Advanced Engineering Model
- MANETS: Mobile Ad-Hoc Networks
- PHY: Physical
- MAC: Media Access
- FFD: Full Functional Device
- RFD: Reduced Functional Device
- OS: Operating System
- EMC: Electromagnetic Compatibility
- WPAN: Wireless Personal Area Network
- PAN: Personal Area Network
- PPDU: PHY Protocol Data Units
- PLME: Physical Layer Management Entity
- ED: Energy Detection
- LQI: Link Quality Indication
- CCA: Clear Channel Assessment
- DSSS: Direct Sequence Spread Spectrum
- MLME: MAC Sub-layer Management Entity
- SAP: Service Access Point
- MLMESAP: MAC Sub-layer Management Entity Service Access Point
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPDU</td>
<td>MAC Protocol Data Units</td>
</tr>
<tr>
<td>GTS</td>
<td>Guaranteed Time Slots</td>
</tr>
<tr>
<td>CAP</td>
<td>Contention Access Periods</td>
</tr>
<tr>
<td>CSMA-CA</td>
<td>Carrier Sense Multiple-Access with Collision Avoidance</td>
</tr>
<tr>
<td>ECU</td>
<td>Electronic Control Unit</td>
</tr>
<tr>
<td>CFP</td>
<td>Contention Free Periods</td>
</tr>
<tr>
<td>IDE</td>
<td>Integrated Development Environment</td>
</tr>
<tr>
<td>EMI</td>
<td>Electromagnetic Interference</td>
</tr>
<tr>
<td>IC</td>
<td>Integrated Circuit</td>
</tr>
<tr>
<td>I/O</td>
<td>Input/Output</td>
</tr>
<tr>
<td>HID</td>
<td>Human Interface Device</td>
</tr>
<tr>
<td>ADC</td>
<td>Analogue to Digital Conversion</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Fundamental questions associated with the origins of this research are addressed in this opening chapter. The reader is provided with background information on the research, and the problems associated with the development that lead to this research are discussed. This is followed by the methodology by which the research is completed and the chapter concludes with a description of what will follow in the rest of this dissertation.

1.1 Background

ZaPOP is a company that specializes in providing in-store marketing solutions for marketeers. They are leaders in providing media solutions like radio retail, basket defenders and eye catchers, to name but a few. All media solutions are designed to improve brand awareness, capture the attention of possible buyers and to trigger a purchase response from the consumer. [1]

The company provides these media solutions to numerous supermarkets and convenience stores in southern Africa. They are also constantly seeking new media solutions and as such are in process of broadening their capabilities with a new system. [1]

In this new system, advertisements will be played on digital displays, like plasma screens. Some stores are already equipped with displays that show advertisements, but ZaPOP wants to control the advertisements showing in each store, from their office building. In doing this, the company will have
the ability of very fast turnaround times on the deployment of new or updated advertisements, as well as the removal thereof.

To realize this new system, the company have need for the development of three custom electronic devices with specific capabilities in wireless radio frequency (RF) and universal serial bus (USB) communication control. The devices are required to form communication networks inside convenience markets. Normal internet protocol (IP) networks are used for communication between the office building and operational areas.

The client specifically requires the following three devices: an RF-unit, a mini-controller unit and a battery pack. To realize the proposed system, the client will either use combinations of the RF-unit and mini-controller unit, or the RF-unit and battery-pack. The first combination will be used in distributed sensor networks (DSNs) and the later in wireless sensor networks (WSNs). Next, we look at a concept that will describe this system in more detail.

1.1.1 System Concept

Figure 1-1 shows a visual illustration of what ZaPOP desire to accomplish with the new system. It illustrates how IP networks are used to communicate between the office building and all operational areas. Communication networks inside these areas consist of WSNs or DSNs, as they are the focal point of the RF-unit.

Note: In this dissertation all references to WSNs also refers to DSNs, and vice versa.
The system concept presents key units in the form of personnel, personal computers (PCs), store-and-send PCs, network coordinators, routers, motes and digital displays. All communication data (advertisements, commands and so forth) are transferred between the operating PCs at the office building and the store-and-send PCs via IP networks. A network coordinator transfers the data between the store-and-send PC and routers or motes with an established WSN. Routers are used to display advertisements on the digital displays.

All store-and-send PCs are temporary storage areas for data transferred between the IP networks and WSNs. The operational outcomes of the network coordinator and routers are realized using combinations of the RF-unit and mini-controller unit. A mote is the term used for the electronic devices powered by batteries, with RF communication capabilities for operation in WSNs.

Our client’s objective and need is now known. The complete system is a big undertaking for any one man to complete, and therefore the following development scope is defined. This scope defines the first phase of the total system development.

1.1.2 Development Scope

The extent to which the development of the devices is completed is defined by this scope. Section 1.1.1 gave information on the need and objective of the client regarding the complete system.

As a first phase of the total development, ZaPOP only requires hardware units designed for functional capability. Application firmware for these devices will be designed and developed by personnel of the company. These personnel will then test the firmware on the units in the system. Therefore, in this research the devices are designed and developed up to a level where the functional capability of the devices is verified.

1.2 Problem Discussion

Verifying the devices for functional capability in the proposed system, is the main outcome of the development and this led to two daunting tasks to complete. The first is to use a development methodology that will ensure the devices are developed for functional capability, and the second is to verify that the devices will satisfy the needs of the client.

At the beginning of this development, we had little experience in the development of electronic devices of this nature. This is the development of devices for the consumer market, which need to operate in currently standing company structures and operational objectives. Functional capability is now technical and operational feasibility. Based on this, the following research objective is recognized.
1.2.1 Research Objective

The objective of this research is to find and implement a scientific design and development methodology for the development of the RF unit, mini-controller unit and battery pack. These devices must be verified for functional capability up to the development scope.

1.3 Research

The research has three different aspects associated with it, as shown in Figure 1-2. The basic and ideal order is shown in red, although deviations may occur depending on the results of each. The three aspects are summarized as follows:

- To develop a suitable scientific methodology for the development.
- To implement this methodology on the design and development of the three electronic devices.
- To validate and verify the methodology and end product.

![Figure 1-2: Research.](image)

The different research aspects are integrated because of their mutual influences. The development methodology dictates the terms and operations of how the development of the product occurs, as well as the validation and verification methods. Knowledge gained through the development can be used to improve the methodology, validation and verification steps. The validation and verification identifies errors that can be used to optimize and correct the development methodology and/or product.

1.3.1 Methodology

We were in a quandary ever since the client came to us with his need. We were unsure on how to satisfy the needs and requirements of our client. Because of this, we decided to attend a course in advanced electronic development at the NWU, which we have passed with honours.
In attending this course, we found similarities between our development and many other developments, no matter what the application of the product under development. In general, all products are influenced externally and internally by other factors. This could be client inputs, government regulations, operational environment, people, objects, and so on. One can think of these factors as operating entities with their own attributes having inputs, processes and outputs. In our development, the electronic devices are influenced by numerous such components. [2]

Any combination or assembly of these components form a system that works together towards a common objective. For our client, this is the advertisement concept as described in Section 1.1. This is why the development methodology must be based on systems engineering principles as learned during the course. There is also an added advantage using these principles, because we can treat the validation and verification aspects as a separate, yet integrated, component in the system. [2]

1.3.1.1 Engineering our System

Systems engineering is a collaborative and interdisciplinary approach, that uses life-cycle outcomes to derive, evolve and verify a solution that will satisfy the needs and expectations of the client. A big advantage of systems engineering is that it can be used in a variety of engineering disciplines. The way the development occurs is defined by a process model. [3]

The process model dictates the terms and conditions of what to do and when to do it. No two process models would be alike even if both are based on the same systems engineering principles, because different applications require different tools. To us, systems engineering is merely a concept that provides a basket of tools for development, but is not a methodology in itself. The process model is the methodology, and for our purpose will be based on systems engineering principles. To ensure that our development adheres to this, we must make sure that we use the correct process model. [2][3]

To guarantee that our process model is accurate and verifiable, an extensive study on systems engineering is completed together with knowledge that we gained through taking the course. This knowledge is used to develop a process model for our development.

1.3.2 Validation and Verification

Functional capability is assured by two methods: a top-to-bottom approach and a bottom-to-top approach. The first is done with the development methodology and the latter by empirical study, using laboratory testing.
In the coming chapters it will become clear that validation and verification is not just a component of our development methodology, but also an integral part of the system engineering process. Because systems engineering uses a top-to-bottom method, the devices are developed to fit into a system rather than just a single operational objective. [2]

Laboratory testing verifies the internal and external interfaces of the devices at a detailed level. Using a bottom-top approach, it can be verified that the devices are functionally capable at a concept level. [3]

1.3.3 Originality and Uniqueness

Although the means by which this development will occur is not new to the research community, it is however original and unique because of the following:

- The needs and requirements of our client differ from other developments, which makes it an original development.
- It has a unique application and objective. Our client is in the forefront of the marketing arena and stands much to gain from this development.
- The process model used is tailor made for this development.

1.3.4 Contributions

In a research like this, it is necessary to ask how we contribute to the engineering community. There are numerous outcomes, but the main outcome is a real life practical example of consumer development – the development of products for a client with a specific need, to operate in his existing business.

Any engineering student can use this work to gain knowledge in product development, as there is no confidentially, expect for the financial impacts of the development. Another important outcome is the enhancement and improvement of ZaPOP’s marketing capabilities. The successful realization of their system largely depends on the success of this research and as such, the two will reflect on each other because of mutual influences.

As an ongoing development, the experience gained through this work would be advantageous to the client, because the engineer will go on and work on this project for the client on completion of this study. Therefore, we can conclude that there is both a theoretical and practical contribution to the community.
1.4 Dissertation Outline

To keep the focus squarely on the aspects of this research, the dissertation is structured in a manner that always keeps its reader interested. All features of the research are spread out over eight chapters with Chapter 1 being this chapter.

Section 1.3 showed the three aspects of this research. The remaining seven chapters are spread and linked together to address these aspects. Chapters 2 and 3 give the reader information on the methodology behind developing the required devices. The development itself is discussed in Chapters 4, 5 and 6. The validation and verification aspects of the research are handled throughout the development, but are discussed in Chapter 7. Chapter 8 reflects on the accomplishments.

The process model on which the development is based is given in Chapter 3, but in order to fully understand the concepts and principles behind it, a study on systems engineering is done in Chapter 2. This study is aimed at providing the reader with detailed discussions and examinations of the systems engineering principles used to develop the process model for our development. Chapter 3 not only gives information on the process model, but serves as an example of how the different concepts and principles of systems engineering functions together.

In Chapter 4, the conceptual design phase of the process model is completed. In this design phase, the client’s needs, objectives and system concepts are used to develop relationships between system entities to define the fit of the devices. Chapter 5 focuses on the function of the devices by evaluating the operational requirements of the devices and other system entities. The fit and function outcomes of the system entities are then used in Chapter 6 in a detail design to establish the form of units. This form is actual prototype models are tested for functional capability.

Chapter 7 shows the steps taken to validate the development methodology and verify the devices for functional capability. It uses the results from previous chapters to do this. The author’s final comments on this work, future recommendations and a critical review on the developed devices and the methodology used, are given in Chapter 8.

1.5 Conclusion

An introduction to this research was given in this chapter and the success of the decisions made, to complete the research and development, will become visible during the next chapters. The effectiveness of the systems engineering approach on this development and the functionality of the developed devices shall be the main points of a critical review in the final chapter. Only after we examine this evidence can we truly comment on the decisions made.
Chapter 2
Systems Engineering

This chapter gives a more accurate and meaningful description of systems engineering. Principles and concepts of this field as used in our development methodology are illustrated by these descriptions.

2.1 Introduction

Any operational environment consists of an integrated composite of people, products and processes that work together towards a common goal and/or objective. This integration is commonly known as a system. [3]

In the previous chapter we stated that systems engineering (SE) is a collaborative and interdisciplinary approach, that uses the life-cycle outcomes of a product to derive, evolve and verify a solution that will satisfy the needs and expectations of the client. SE uses both system management techniques and technical knowledge to derive this solution. The life-cycle outcomes, operational environment and design knowledge constitutes the technical domain, whereas the management domain determines the method of development. [3]

The product of SE is an engineered system, aimed at making the world better for people. This engineered system should be designed to satisfy the need of the client while reducing costs, ecological and negative societal impacts. SE management should ensure that all internal and external impacts are designed for, from the start of development. Technical personnel should adhere to the development method given by management and provide solutions for all the imposed requirements. [2][3]
2.2 Systems Engineering Management

The management domain of SE concerns itself with development phasing to control and coordinate the design process. Development phasing is the technique used to establish how phases and stages are used within a process model. It provides a baseline for requirements-tracking throughout the development and integrates life cycle outcomes for a complete development. [3]

2.2.1 Top-Down Approach

SE follows a top-down approach, moving from a concept level, to a system level and then to a component level. The concept level produces a description of the system that is used to form requirements for the system at a system level analysis. These requirements become more detailed as development moves from this system level to a component level.

Movement is completed in steps called phases and stages. A phase represents a design area like the conceptual design, and a stage is a focus area within a phase. [2][3]

2.2.2 Development Phasing

The integrated composite of people, products and processes in a system forms a hierarchy and each design phase focuses on a different level in this hierarchy. The highest level is usually represented by buildings, people, equipment, objects, products and so forth, while the lowest level is made up of the inner building blocks from a single system entity, like a specific microprocessor. [2][3]

Figure 2-1 shows the development phasing for a generic process model. This generic model shows how development moves through the different levels in the hierarchy. Stages within each phase represent the technical expertise required for completing the phase. In this generic model, the conceptual design is used to develop requirements for the system, the preliminary design refines the system requirements into single entities and the detail design phase is used to design and construct the entity to satisfy the requirements of the previous design phases. [2]
Ideally, engineers want to finish a development in one development cycle - to go through each development phase one time. This however does not happen, as there are always changes in requirements and designs, which require new attention into previous design phases. This is why the waterfall model fails as a development methodology, because there is no means for feedback. [2][4]

A process model dictates the order of the phases and stages. In the spiral model each design cycle focuses on a different portion of the design. For example, the door of a vehicle is a focus area in the development of a vehicle. However, the different design cycles all share a common baseline (order of stages) for developing requirements and architectures, and then allocating them to each other. This allows for consistency, which makes future integration, backtracking, fault detection and error correction easier. The model also teaches us to have equilibrium between the amount and type of tools used in the development. It is a risk driven approach and as such focuses more on the involved risks of the system rather than the actual product itself, which is not always advantageous. [2][5]

### 2.2.3 Life-Cycle Engineering

SE addresses all the aspects associated with life-cycle engineering. Concurrent consideration of the life-cycle needs of a system can only be achieved through an integrated development. The characteristic actions of this cycle are known as life-cycle functions and an example thereof is given in Figure 2-2. The most common life-cycle functions are need, acquisition, production, construction, training, deployment, operation, support, disposal and phase-out. [2][3]
The engineer must be sensitive to these utilization outcomes early during development to foresee the life-cycle functions and functioning of the system. Life-cycle outcomes of any system are not just limited to that of the product under development, but to all system entities. [2][3]

### 2.2.4 Completeness

Process models based on SE force a complete effort in all areas of design, specifically in the initial definition of system requirements and characteristics. Requirements and characteristics are used in and are related to specific design criteria, system resources, functional units and follow-on analysis. The follow-on analysis ensures the effectiveness of early decision making. This complete approach allows for a rational design in which all design decisions are documented. [2][4]

Development in this a manner guarantees product competitiveness, as the product should satisfy the needs of the client and in respect, the needs of the clients’ consumer. A challenge thus exists to bring cost-effective products and systems into being. To follow a complete systematic approach during development ensures traceability. This makes backtracking, error correction and fault detection easier, because there is stability in the development. [2][6]

A complete approach in systems engineering also requires a Failure Mode and Effects Analysis (FMEA) and a Critical Analysis (CA) (FMECA). The FMEA identifies all the different failure modes, and the effects that they will have on the system. CA evaluates these modes in terms of severity and probability, by classifying and prioritizing them. [6]

### 2.2.5 Interdisciplinary

SE is an interdisciplinary science because it requires expertise in many different disciplines. The various disciplines is enforced by the existence of numerous and very different system entities and/or components. These include many different design disciplines like manufacturability, usability and so forth, as well as the skills required in communication and documentation. [2]

### 2.3 Technical Development

The technical side of SE deals with the actual development, by following the method described and defined by the management domain. The process of development is called the systems engineering process (SEP) and is applied sequentially top-down, one level at a time with each new level adding more detail. [3]
The SEP is used to transform needs and requirements (inputs) into system and/or product descriptions and requirements (outputs). The method by which this is accomplished is defined by the stages within each phase. The first design loop (for example the conceptual design) transforms the need of a client into system concepts and requirements. This is then used as next level inputs in the preliminary design. Figure 2-3 shows a graphical representation of what the SEP is all about. [2][3]

Figure 2-3: The Systems Engineering Process.

The SEP is used to achieve a consistent and common baseline in each design phase, discussed in Section 2.2.2. The figure indicates that the SEP has three distinct sections: process inputs, design loop and process outputs. Each of these sections will now be examined. [3]

2.3.1 SEP Inputs

Inputs for the SEP comes from a wide variety of sources but primarily from the client, consumer, government and technologies associated with the design. The operational environment, system needs and objectives impose requirements to improve the operational feasibility of the product. The most common SEP inputs can be summarized as follows [2][3]:

1. The needs and objectives of the client and consumer are self explanatory, but there are instances where they impose their own specific design requirements like dimensions and size, for example. [2][3]
2. The technologies that will be used in the design and development of the system and/or product have their own requirements, which need to be taken into account. [2]
3. Government regulations enforce requirements that protect the environment and society. [2][3]
4. High operational feasibility is achieved by making sure that all the little details are taken into account. This means to establish a correct and precise set of design criteria. The most common items for a design criteria includes supportability, safety, manufacturability,
Chapter 2 – Systems Engineering

Development of a modular low power IEEE 802.15.4 wireless sensor module

5. The outputs of the SEP are used as inputs in ongoing iterations of the SEP, which are then refined and described in more detail. [2]

2.3.2 SEP Design Loop

The design loop is the process of transforming inputs into requirements and to develop a functional analysis that support those requirements. Requirements and functional analysis become more detailed after design loop iterations. The stages of the design loop are discussed next. [3]

2.3.2.1 Literature Study

Studies to gather any form of information is generally known as a literature study. In these studies information is gathered on topics associated with the development. From this requirements are gathered to which the design must comply. [2][7]

Client inputs, system concepts, associated technologies and so forth are all forms of literature that requires analysis. Some pieces of literature constantly changes (for example the need of the client or end objective), and some of them are fixed engineering documents (like the operations of a certain communications protocol and component datasheets). The literature study is a means by which the SEP inputs are analyzed. [3]

2.3.2.2 Requirements Analysis

A requirements analysis is an integrated part of SE and is used to develop functional and performance requirements for the system and/or product under development. These requirements define what the system must do and how well it must do it. [3]

Requirements are written to be understandable for anyone and they are unambiguous. They are also comprehensive, because they should cover all aspects of their subject. Requirements are also complete and precise. [2][3]

All system and product requirements must be achievable, verifiable, unambiguous, complete and must contain all mission profiles, as well as operational and maintenance concepts. It also must be stated in
terms of need and not give the end solution. Requirements must be consistent with each other and appropriate for the current iteration level of the SEP. [3]

2.3.2.3 Functional Analysis

The functional analysis is used to decompose higher-level functions into lower-level functions. Operational functions, entities, interfaces and relationships are found by decomposing the requirements of the system. Thus, the functional analysis is used to describe what the system must accomplish in terms of its requirements. [2][3]

The functional analysis avoids early commitment to specific design details such as schematics, components, and so forth. It defines the operational and architectural outcomes of the system, as well as the relationships within and between them. The most common tools for a functional analysis are [2]:

1. Operational procedures to describe the actions and functions of system entities, specifically how people should interact with the system.
2. Functional architectures that illustrate how all system entities fit together. It is a formal description of their fit and function.

2.3.2.4 Resource Allocation

The outputs of the functional analysis are directly drawn from the characteristics and requirements revealed by the literature study. Therefore, requirements are related to the functional analysis with the help of resource allocation tables. [2]

Higher-level requirements are allocated to lower level functions identified through the functional analysis. Resource allocation tables provide a solution for tracking requirements throughout the whole design effort. This has numerous advantageous associated with it, since it aids in future analysis, error correction, fault detection and so on. [2][3]

2.3.2.5 Trade-Off Studies

Resource allocation tables provide a means, by which solutions for system functions can be found. The requirements associated with a product characterize that product, which should be used to identify the solution. [2]
The best solutions are usually elected by completing trade-off studies. These studies weigh up the different possible solutions against each other and against the characteristics of the functional unit. The chosen solution is the one who fits the specifications the best. [2]

2.3.2.6 Design Synthesis

The requirements analysis, functional analysis, resource allocation tables and trade-off studies provides more than enough information to develop concepts and actual design solutions. A design synthesis puts all the information, provided by these design stages, together to give a solution for the product under development. [2][3]

2.3.2.7 Design Review

In systems engineering there are two methods of design reviews: formal and informal. Formal reviews are performed before major evolutionary steps, while informal reviews take place continuously on a day-to-day basis, throughout the development. [2]

2.3.3 SEP Outputs

SEP outputs depend largely on the level of development. In early stages of development they are requirements, low-level architectures, specifications and baselines. Moving through more levels of development the outputs becomes more detailed until prototype models are developed. [3]

Prototype models themselves move through different stages of complexity and purpose. The first prototype model is usually called the experimental model (XDM) and is used for proof of concept studies, like breadboard designs and computer aided design (CAD) models. Then there is the engineering model (EDM) that is capable of being used in a lab environment that mimics the actual operational environment of the product. The advanced engineering model (ADM) is the final output model destined for the actual environment. [2][3]

All outputs of the SEP are used as next level inputs for new iterations of the SEP. Information, experienced and knowledge gained from the outputs are used to further improve the design and development in terms of requirements and characteristics. [2]
Chapter 2 – Systems Engineering

2.4 SE Aspects and Concepts not Used

There are two aspects of SE that we did not include in our development methodology, due to the scope of the development. These are known as the acquisition and enterprise processes of SE. The prior deals with the origins of the project and the latter with how developed products are linked to business strategies.

In this section, we will look more closely at these two aspects, and show why they will not be included in our process model. We will also look at known process models that can be used in a SE environment and state why we did not adopt one of them.

2.4.1 Acquisition Process

The process to identify and describe needs, requirements and the method to meet those requirements are known as the acquisition process. It is a process that takes place before the SEP, as shown in Figure 2-3. It is from this process that the client formulates needs and objectives. [8]

This acquisition process should not be confused with the acquisition phases of the system life-cycle, (Figure 2-2). The last are acquisition phases, because they resemble the process of achieving the goal, while acquisition processes are plans, focussed on business and technical criteria, to achieve the objectives of a program. [8]

A program may refer to any undertaking within a business. In our case it is the system concept as given in Section 1.1. The acquisition planning of this concept is completed by our client and we have no involvement in that whatsoever. Acquisition processes are left out of our process model for that reason.

2.4.2 Enterprise Process

In Section 2.2.4 we described the completeness of a SE management plan. System resources do not just involve internal units like personnel, management, manufacturing, maintenance, and so forth, but a complete effort in linking business strategies to the developed system architectures. To do this, the enterprise process is used. [9]

The architecture of any enterprise process must include all business related outcomes of the program together with the strategic strategies, policies, standards and general specifications of the business. It also includes the intricate interrelationships of people, processes, available technologies and objectives of the business (or enterprise). [9][10]
All enterprise outcomes, associated with the system we will be developing, are handled by our client. The SEP that we will present in the next chapter should attempt to balance the outputs from the acquisition process, the inputs from the enterprise process and that what is technically achievable within the budget limit. Thus, we will use the enterprise inputs from our client and not develop them ourselves.

2.4.3 Process Models for SE

Process models like the spiral model, V-model, Agile model, and so forth, can be used as a SE development methodology. These models are all very complex in their structure and can be used for all the different processes of SE – acquisition, management, technical development and enterprise.

It was decided not to use and adapt a known process model, but rather use the general model as used in the discussions in Sections 2.3 and 2.4, for the following reasons:

- Due to their complexity and wide range of usage (all SE processes), since we need not to include the acquisition or the enterprise processes of SE.
- A comparison would have to be drawn between all the different models to select one. This comparison will require extensive research and testing of the models which in itself is a research problem, and it would take much time away from the actual development.
- Clear understanding of SE concepts will come forth using the general process model.

2.5 Conclusion

In this chapter, an overview of SE concepts was presented, based on a few comprehensive reference sources and knowledge obtained through consultation with experts actively involved in the industry. As such, it doesn't address the research question on the development of the three specific electronic devices, but does address the chosen methodology for the development.

SE was expanded into two concepts – SE management and technical development. The first enabled us to understand how the different phases and stages of a development are structured. It is an essential part of the SEP, because it controls and coordinates the design process for a complete and total development.

Technical development deals with the development and design of the product in accordance with the SEP. The SEP shows technical personnel how to transform the needs and objectives of the client into requirements and products ready for delivery through different levels of design. This complete interaction between the management and technical domains forms a process model. This process model is a development methodology as it dictates the development steps and criteria.
At this moment, it is still unclear what the effect on our development would be, due to the decision to not include the acquisition and enterprise processes, as well as the decision not to use a known process model. The effectiveness of our research on SE principles and concepts will become visible throughout the development of the methodology and the actual development of the product.
Chapter 3

Development Methodology

The methodology used for the development of the three electronic devices is presented in this chapter. It is based on systems engineering principles and concepts, and includes mechanisms for research validation and verification.

3.1 Introduction

The theory on SE concepts and principles, presented in the previous chapter, will now be used to develop a methodology on which the development of the electronic devices will be based. A complete overview of the process model is given first.

All development phasing and SEP aspects of the process model are discussed and described. After this discussion, the focus of the chapter moves towards the individual phases of the process model. These phases are described in terms of their internal development stages and SEP.

3.2 Complete Process Model

An overview of the complete process model is given in this section. It starts with the development phasing of the model, and then the SEP thereof.
3.2.1 Model Development Phasing

Figure 3-1 shows the phasing for the process model. The focus of each phase differs slightly from the general model, to tailor it to our specific development program.

Figure 3-1: Phasing Model for Development Methodology.

The objective of each development phase in the phasing model is defined as:

1. **Client Need and System Concept**: This phase is designated for conversations with the client to formulate needs, objectives, requirements and system concepts.
2. **Conceptual Design**: In this phase, the three electronic devices are analyzed and developed for the system they will operate. Attention is given to all entities and interfaces of the system, as well as the technologies involved.
3. **Preliminary Design**: This phase of development focus on the devices themselves. They are analyzed in terms of their external interfaces, as well as internal building blocks and interfaces. The technologies required to realize the devices are examined, and solutions are found through trade-off analysis studies.
4. **Detail Design**: This design phase concerns itself with the detailed design of the three modules. The designs progresses through various prototype models until the functional capability of the devices are assured.
5. **System Utilization**: The required life-cycle outcomes and- functions of the system are determined by this phase.
6. **Continuous Review, Feedback, Analysis and Control**: This is the process where developers continuously monitor, validate and verify the design and development.

3.2.2 Model Process

The SEP of our development phasing model is shown in Figure 3-2. It is based on the model given in Chapter 2, but with a few differences to make it more applicable to our development program. Firstly, a new stage is introduced within the design loop, called the detailed design. This stage is used during the
Chapter 3 – Development Methodology

detail design phase for printed circuit board and firmware design and development. The functions of the other design loop stages stay the same as described in the previous chapter.

There is a sense of rational design (ideal development procedure) within the design loop. Process inputs and outputs make way for a common baseline between all phases of development. In the next sections of this chapter it will be shown just how phasing and the SEP works together to achieve this.

![Figure 3-2: Systems Engineering Process for Development Methodology.](image)

3.3 Process Model Sections

In this section the individual development phases of the process model are discussed in more detail. Illustrations and descriptions are given to show how development phasing and SEP works together towards one common goal, which is a strong scientific development methodology.

3.3.1 Client Need and System Concept

The basic needs and objectives of the client, and a system concept thereof, were given in Section 1.1. A more precise description of this is given in the conceptual design phase, where an analysis of all client inputs is given.
During this phase of development there are meetings and conversations with the client, to get a better understanding of the needs and objectives of the client, as well as specific requirements the client may have.

### 3.3.2 Conceptual Design

The most important outcomes of the conceptual design are architectures and requirements that describe and define the operation of the system. In Figure 3-3, the process model of the conceptual design is given in terms of inputs, design loop stages and outputs.

![Conceptual Design Described](image)

A majority of the input sources come directly from the client and the technologies used in the system, as they are the source of the operational objectives, characteristics and requirements from a system perspective. There are also inputs from the operational feasibility criteria that evaluate the system in regard to its performance, environmental compatibility, effectiveness and economic feasibility.

Focus is on what the three devices should accomplish, and not how they should accomplish it. This enables engineers not to subject themselves to specific designs early during development, where concept studies are performed. It also allows for the correct identification of system resources through
functional analyses, which must relate to the identified requirements. This sets the tone (specifications, baselines and decision database) for the future iterations of the SEP.

3.3.3 Preliminary Design

In this design phase, the viewpoint moves from a system perspective towards individual entities and prominence is on the development of requirements and resources for the three electronic devices. Here the outputs from the conceptual design are evaluated on a more detailed level.

The process model for the preliminary design is given in Figure 3-4. Outputs from the conceptual design are joined by new SEP inputs. Detail requirements and characteristics of the three devices are used in a functional analysis to develop their detail functional architectures. These are related to one another, and used during a trade-off analysis to find preliminary solutions for each unit. These solutions are functional descriptions in terms of configurations, specific components and so forth, but do not portray the final design.

![Figure 3-4: Preliminary Design Described.](image-url)
3.3.4 Detail Design

Previous design phases were concerned with the development of characteristics, requirements, resources and conceptual solutions. This phase differs from previous phases, because the outputs are used to develop detailed designs of the three devices. Figure 3-5 shows the relating process model.

The development of detailed designs has different output levels (prototypes), where each level differs in complexity and purpose. In our process model, three levels are defined: experimental models, engineering models and advanced engineering models.

![Diagram showing the process model](image)

**Figure 3-5: Detail Design Explained.**

### 3.3.4.1 Experimental Models (XDM)

Experimental models are aimed at proving the functional feasibility of the system within a laboratory environment. In our development, XDMs are used in the following perspectives:

- To confirm that the solutions provided by the trade-off analysis are possible
- Breadboard testing (if possible) for preliminary functional capability testing
- CAD models to ensure that mechanical requirements can be met
3.3.4.2 Engineering Models (EDM)

These models are live representations of the design given by the XDMs. An EDM is first tested in a laboratory environment and then in a real-life environment. It enables developers to test and confirm the following aspects of the development:

- Both internal and external interfaces of the three devices
- Operational feasibility criteria
- Technical feasibility

*Note: Due to the scope of the development (Section 1.1.2), we only have to develop an EDM model of each of the three electronic devices, and test them in a laboratory environment. These devices will then be tested in a real environment once in the hands of the client. The results from these tests will then be used in the development of new and improved EDMs or ADMs.*

3.3.4.3 Advanced Engineering Models (ADM)

ADMs resemble the actual product that will be used by our client when the system is operational. The model is also the subject of ongoing evaluation, to further improve its operational capabilities. As mentioned above, this model is not required for development.

3.3.5 System Utilization

This development phase constitutes the operation of the ADM. It describes future life-cycle outcomes and operations of the system and product.

3.3.6 Continuous Review, Feedback and Control of SEP

The outcomes required for the validation and verification of product development is made possible through this development phase. All design and development decisions are constantly monitored, evaluated, validated and verified. Its main aim is to maintain a constant basis for improvements via feedback and control. In our process model, there are three different levels for this phase:

1. **Stage level reviews:** These low-level reviews are aimed at identifying and correcting errors within and between stages on a day-to-day basis.
2. **Phase level reviews:** Almost the same as stage level reviews, but is used to identify and correct errors within and between phases. It is also completed on a day-to-day basis.
3. **Formal reviews**: This is a high-level and formal design reviews, scheduled at the end of every development phase.

Informal day-to-day reviews are completed to ensure that the designed system do not deviate from the established system requirements. Formal reviews provide an audit of the design phase, ensures a common baseline through all phases, and improves the decision database.

*Note. Evidence for reviews does not exist explicitly, for reasons that we shall mention in Chapter 8. Reviews results come from the positive experienced we felt using the different levels of reviews, and the effect they had on the developed product.*

### 3.4 Conclusion

This chapter focussed on presenting the methodology used for the development of the three electronic devices – RF unit, mini-controller unit and battery pack. We also showed how the process model for the development is based on systems engineering principles.

A top-down approach is followed in predefined phases of development. Each development phase focuses on different levels of complexity in the design. The different design phases are **Client Need and System Concept**, **Conceptual Design**, **Preliminary Design**, **Detail Design**, **System Utilization** and **Continuous Review, Feedback and Control**.

Development progresses from a concept, to a system, to a detailed analysis of the devices as movement progresses from one phase to another. Three different design models are also noted for the detail design phase and each of them differ in complexity and in focus. XDMs are used for testing in the laboratory, EDMs for operational testing in an environment that mimics the actual operational environment and ADMs are the final design model ready for production and full operation.

The process model also incorporates various different design reviews, feedback and control mechanisms for the SEP and development iterations. These mechanisms ensure a full and complete development, and aids in the validation and verification aspects of the research.

Overall, the presented methodology for development should provide a good and effective baseline to develop the devices for our client. How good and how effective it is, is an entire question in itself which we shall address in Chapters 7 and 8.
4.1 Introduction

The purpose of performing the conceptual design is to develop and analyze requirements and characteristics that describe the system. This is a system representative of the application area and environment of the three devices.

Steps and processes required to perform the conceptual design were given in Chapter 3. The first step is to analyze the inputs from the client, where after the technologies associated with the system, are examined. A discussion on the operational feasibility of the system is also performed. All this constitutes the literature study of the conceptual design.

Requirements for the development are drawn from this literature study and used in the development of a functional analysis that accurately describes the system. All identified requirements are related to their respected operational procedures and/or functional architectures through resources allocation tables. The conceptual design ends with a design synthesis and a design review.
4.2 Literature Study and Analysis

Presented in this section is a study that analyzes the inputs from the clients, technologies associated with the development and the operational feasibility of the system.

4.2.1 Analysis of Client’s Needs and Objectives

ZaPOP has a need for the development of three specific electronic devices. Custom enclosures for these units are not mandatory, but need to be taken into account in the design of the devices. The three devices are destined for different operational functions - RF communication unit, mini-controller unit with specific USB functions and a battery pack.

The RF communication unit is a small module that will be used for wireless communication via the IEEE 802.15.4 protocol. Data distribution for all network functions including routing, beacon and coordinator modes is made possible by this unit.

In WSN applications, the RF unit will be used in conjunction with the battery pack to form a sensor node or mote, and in DSN applications with the mini-controller to form a multipurpose controller module. This will enable the RF unit to communicate with USB peripherals and other devices. Connections between the mini-controller and USB hosts or other devices will be made possible via cable. The USB host refers to either an operating PC or a digital display.

A description is now formulated for both the system and product based on the system concept and client need. The description of the product is:

- The three electronic devices are developed because of the need of a client.
- The devices are aimed at bringing improvements and enhancements in the business of the client, and therefore need to operate in the existing business definition.
- The RF unit and battery pack can be used together to satisfy the need of WSN applications.
- The combination of the RF -and controller units satisfies the need for DSN applications.

Note: In the above, business definition describes how the business of our client operates. This is confidential to our client and will not be presented in this dissertation.

A description for the system in which the product must be capable of operating is:

- The system must form part of the existing business definition and system definition.
- The operating environment for the product is WSNs and DSNs, in convenience markets.
- Normal IP networks are used for communication between the office and convenience markets.
Chapter 4 – Conceptual Design

- A PC exists in the operating environment that acts as a store-and-send link between the IP networks and the WSNs or DSNs.

*Note:* The system definition resembles the existing system resources and architectures of the client. The system definition and business definition goes hand in hand. In this dissertation limited resources and architectures (only those that influence the product directly) are given due to confidentiality.

### 4.2.2 Associated Technologies

The technologies used in the described system have their roots in Mobile Ad-hoc Networks (MANETS) in which the communicating nodes are mobile. They have no infrastructure and use peer-to-peer communication, to communicate in a multi-hop fashion. [11]

Devices used for communication in these networks force constraints on their embedded software due to their limited memory and computing power. The need to have a long battery life places a constraint on power consumption.

A specific application of MANETS is WSNs. Both use small and cheap RF communicating devices with quantities from a few to thousands. MANETS are used in tracking and information distribution, but WSNs have an additional function in having the ability to monitor the environment via sensors. [11]

The concept of small cheap sensor devices with RF communication in WSNs is known as *motes*. It was first introduced by Berkeley in 1998. Figure 4-1 shows two examples of motes. The features of these two motes are shown in Table 4-1, and it gives an indication of their limitation in hardware resources. Other commonly known motes on the market are the ‘weC’, ‘Dot’ and ‘Mica’ motes. [11]

![Figure 4-1: Examples of Motes.](image)

**Figure 4-1: Examples of Motes.** [12][13]
Motes in WSNs are self-configuring because there is no fixed network topology. However, cases do exist where motes are infrastructure dependent, like in DSNs. In WSNs, all nodes share a common application, but in DSNs the application is distributed among the nodes. Figure 4-2 illustrates this difference. [11][14]

As mentioned, the motes in these networks are low-power devices with limited computing power. It is clear to see that in WSNs, the more data a mote processes, the more likely it is that it will fail first as a result of battery power usage. This is not the case in DSNs as the application is distributed among its devices. An example of such an application is to allow some devices to route information and other to monitor the environment. The IEEE 802.15.4 protocol allows developing engineers to distribute application outcomes between devices. [14][15]

This protocol provides a means for the network to uniquely identify every radio (node) in the network, as well as a method and format for communicating between them. It works on the physical (PHY) and media access control (MAC) layers of the protocol stack. On the MAC layer, the standard defines two different types of nodes: a full functional device (FFD) and a reduced functional device (RFD). The FFD operates as the network coordinator (network gateway) or as a router, whereas an RFD only operates as a router or a normal end-node. [15]
Another aspect worth mentioning of motes is TinyOS and NesC. TinyOS is an open source operating system (OS) specifically designed by Berkeley for motes, to limit their power consumption. NesC is the programming language for TinyOS applications. [11]

4.2.3 Operational Feasibility

An analysis will now be completed on aspects that will make the system operationally feasible. The operational feasibility of the system comes directly from the deployment characteristics of the system.

Deployment characteristics illustrate how system resources work together. The main resources in a typical system, and coherent with our system, are office buildings, operational sites, manufacturers, depot-and-store facilities, personnel and management. These are personnel with occupations in installation, operation, maintenance, inspections, and so on. System management consists of one, two or more people that act as the manager(s) of the system.

The three devices will be used inside convenience markets, to establish WSNs. These devices are controlled from the office building and stored at a depot-and-store facility when not used. During operation, system tasks are handled by specific individuals (personnel).

These are tasks such as operating the system, programming devices, installing new devices into the system, inspecting the installed and stored devices, and conducting maintenance or repairs on damaged and faulty devices. Operating personnel interface the system through the operating PC and communicate with the system manager(s). All other personnel are capable of coming into direct contact with the three devices and system manager(s). Interactions of these personnel should be included in the development.

System effectiveness and performance is assured through the development of highly reliable, efficient, durable, dependable, user friendly and cost-effective solutions for system entities. On a system-level, the following subsections cover the operational feasibility of the system. They include discussions on the economic feasibility, effectiveness factors, performance criteria and environmental compatibility of the system.

Note: The three devices will be mounted against mounting structures, the same as current products, and all resources draw power from mains supply.

4.2.3.1 Economic Feasibility

All economic aspects of the system are completed by the client and will not be given in this dissertation, for reasons of confidentiality.
4.2.3.2 Effectiveness Factors

Factors for system effectiveness include reliability, efficiency, durability, and user friendliness. These factors are defined as follows [2]:

- **Reliability** is the probability that a designed and developed system or system resource will accomplish its task. Therefore, designs should be aimed at achieving high reliability.
- **Efficiency** is the probability that the prescribed operational demand is met within a certain time period and predefined operating conditions. System resources should strive for high efficiency.
- **Durability** is the ability of the system and system resources to endure. A system of high durability will last longer, which results in a reduction of system running costs.
- **User friendliness** defines the extent to which the system and system resources must be usable by personnel. A user-friendly development considers factors like the required skill level required to handle the devices, anthropometric -, sensory -and physiological factors of humans.

4.2.3.3 Performance Criteria

Performance factors enforce both performance and effectiveness requirements for the system, and more specifically for system resources and/or components. These requirements need to be stipulated by the client. No performance requirements were received, except functional capability.

4.2.3.4 Environmental Compatibility

Environmental factors include temperature, humidity, vibration and noise. They can affect the psychological state, motivation, initiative, self-confidence and efficiency of people. Consideration is required for the following [2]:

1. Difficult installation areas.
2. High temperature variations due to the huge distribution area of operational sites.
3. The operational areas (convenience markets) pose difficulties due to factors like people movement, different communication obstacles (trolleys, cellular phones), and so forth.

4.3 Requirements Analysis

All conceptual design requirements are given in Table 4-2, Table 4-3 and Table 4-4.
4.4 Functional Analysis

The previous sections covered the characteristics and requirements of the system and product. This information is now used in an operational and architectural analysis to identify and define tasks, operational procedures and architectural resources to realise the system.
4.4.1 Operational Analysis

An operational analysis describing system level procedures of system manager(s) or any personnel is not possible, because they are part of the existing business definition.

4.4.2 Functional Architecture

Figure 4-3 shows a functional architecture of our system in the form of resource units and interfaces. In this architecture all functional units represents only an instance of itself.

![System Functional Architecture](image.png)

*Figure 4-3: System Functional Architecture.*

Note. *OmnipoMote, OmnipoController and OmnipoBatteryPack is the names given by the client for the RF unit, mini-controller unit and battery pack respectively.*

A software functional architecture for the system is given in Figure 4-4.
4.4.3 Functional Units Definitions

Descriptions and/or definitions of all identified functional units are given in Table 4-5.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description / Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 1.1 and F/U 1.3</td>
<td>ZaPOPmote</td>
<td>Represents instances where the RF unit is used together with the battery pack to form a mote device.</td>
</tr>
<tr>
<td>F/U 1.1 and F/U 1.2</td>
<td>ZaPOPcontroller as PAN Coordinator</td>
<td>The RF unit is used in conjunction with the mini-controller to establish a multifunctional device with PAN coordinator capabilities.</td>
</tr>
<tr>
<td>F/U 1.1 and F/U 1.2</td>
<td>ZaPOPcontroller as Router</td>
<td>The RF unit is used in conjunction with the mini-controller to establish a multifunctional device with normal FFD and RFD capabilities.</td>
</tr>
<tr>
<td>F/U 2</td>
<td>USB Cable</td>
<td>This cable is used to connect the mini-controller to any form of peripheral with USB host communication capabilities.</td>
</tr>
<tr>
<td>F/U 3</td>
<td>Gateway PC</td>
<td>The gateway PC is used by a FFD to communicate with the office building.</td>
</tr>
<tr>
<td>F/U 4</td>
<td>Peripheral Cable</td>
<td>This cable is used to connect extra peripherals to the mini-controller.</td>
</tr>
<tr>
<td>F/U 5</td>
<td>Attachable Peripheral</td>
<td>Any peripheral attached to the mini-controller is symbolized by this unit.</td>
</tr>
<tr>
<td>F/U 6</td>
<td>Digital Display</td>
<td>The digital displays used for advertisements are represented by this unit.</td>
</tr>
<tr>
<td>F/U 7</td>
<td>Operating PC</td>
<td>This is the PC used in the office building to control and operate the system.</td>
</tr>
<tr>
<td>F/U 8</td>
<td>Operating Personnel</td>
<td>This unit represents all personnel at the office building who operate the system.</td>
</tr>
<tr>
<td>F/U 9</td>
<td>Storage Shelf</td>
<td>The storage shelf represents any form of shelf, cabinet, drawer, and so on used for storing the three devices.</td>
</tr>
<tr>
<td>F/U 10</td>
<td>Product Package</td>
<td>This is the package housing the devices and enclosures, during storage.</td>
</tr>
<tr>
<td>F/U 11</td>
<td>Power Source</td>
<td>The power source is any 220 Vrms 50 Hz source.</td>
</tr>
<tr>
<td>F/U 12</td>
<td>Power Cable</td>
<td>The cable used to connect the power source to all units that require power.</td>
</tr>
<tr>
<td>F/U 1.1 or F/U 1.2</td>
<td>ZaPOPmote or ZaPOPcontroller</td>
<td>Personnel can either program the RF unit or the mini-controller. This unit represents any instance where either one is being programmed.</td>
</tr>
<tr>
<td>F/U 13</td>
<td>Programming Apparatus</td>
<td>This apparatus is the device used for programming the devices.</td>
</tr>
<tr>
<td>F/U 14</td>
<td>Programming Cable</td>
<td>This cable connects the programming apparatus to the programming PC.</td>
</tr>
<tr>
<td>F/U 15</td>
<td>Programming PC</td>
<td>This unit represents the PC used for programming.</td>
</tr>
<tr>
<td>F/U 16</td>
<td>Programming Personnel</td>
<td>Personnel responsible for programming any one of the devices are represented by this unit.</td>
</tr>
<tr>
<td>F/U 17</td>
<td>Mounting Structure</td>
<td>The mounting structure resembles any form of object or wall onto which any system entity is mounted.</td>
</tr>
<tr>
<td>F/U 18</td>
<td>Inspection Personnel</td>
<td>This unit is responsible for inspections on system entities, resources and so on.</td>
</tr>
<tr>
<td>F/U 19</td>
<td>Maintenance Personnel</td>
<td>This unit represents personnel that conduct maintenance on system entities, resources and so forth.</td>
</tr>
<tr>
<td>F/U 20</td>
<td>Installation Personnel</td>
<td>Personnel responsible for installing system resources or entities are represented by this functional unit.</td>
</tr>
<tr>
<td>F/U 21</td>
<td>System Manager(s)</td>
<td>Personnel responsible for managing the system are represented by this unit.</td>
</tr>
</tbody>
</table>
Table 4-5 continuous.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description / Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 22</td>
<td>OmnipoMote Firmware</td>
<td>Firmware required on the RF unit for it to accomplish its requirements is represented by this unit.</td>
</tr>
<tr>
<td>F/U 23</td>
<td>OmnipoController Firmware</td>
<td>This unit represents the firmware required on the mini-controller in order for it to accomplish its required operational outputs.</td>
</tr>
<tr>
<td>F/U 24</td>
<td>USB Host Software</td>
<td>The software on the Gateway PC or any type of USB host.</td>
</tr>
<tr>
<td>F/U 25</td>
<td>Operating PC Software</td>
<td>The unit represents the software on the operating PC.</td>
</tr>
<tr>
<td>F/U 26</td>
<td>Programming PC Software</td>
<td>The required software to program the devices is represented by this unit.</td>
</tr>
</tbody>
</table>

4.4.4 Interface Definitions

All Interfaces identified in the architectural analysis is described and assigned to specific types in this section. Table 4-6 shows the definitions for all system interfaces.

Table 4-6: System Interface Descriptions.

<table>
<thead>
<tr>
<th>Interface Number</th>
<th>Name</th>
<th>Description / Definition</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF 1</td>
<td>IEEE 802.15.4 Wireless RF Communication</td>
<td>The IEEE 802.15.4 wireless RF communication used to establish WSNs or DSNs.</td>
<td>RF Communication</td>
</tr>
<tr>
<td>IF 2</td>
<td>USB Cable</td>
<td>The link that allows for communication between the PAN coordinator and gateway PC.</td>
<td>Electrical, Electronically, Mechanical.</td>
</tr>
<tr>
<td>IF 3</td>
<td>Peripheral Cable</td>
<td>Any data communication link between a router instance and an attached peripheral.</td>
<td>Electrical, Electronically, Mechanical.</td>
</tr>
<tr>
<td>IF 4</td>
<td>IP Network</td>
<td>This is the network used for communication between the office building and the operational sites.</td>
<td>Electronically, Mechanical.</td>
</tr>
<tr>
<td>IF 5</td>
<td>Operating PC</td>
<td>The means by which personnel operate the operating PC.</td>
<td>Human Interaction.</td>
</tr>
<tr>
<td>IF 6</td>
<td>Package Storage</td>
<td>The physical contact between any instance of the device package and the package shelf is defined by this interface.</td>
<td>Mechanical.</td>
</tr>
<tr>
<td>IF 7</td>
<td>Power Cable</td>
<td>The power on the power cable resembles this interface.</td>
<td>Electrical, Mechanical.</td>
</tr>
<tr>
<td>IF 8</td>
<td>Programmer</td>
<td>The physical and data connection between the programming apparatus and the target device is defined by this interface.</td>
<td>Electrical, Electronically, Mechanical.</td>
</tr>
<tr>
<td>IF 9</td>
<td>Programming Cable</td>
<td>The link that allows program data to flow from the programming PC to the programming apparatus.</td>
<td>Electrical, Electronically, Mechanical.</td>
</tr>
<tr>
<td>IF 10</td>
<td>Programming PC</td>
<td>The interface by which programming personnel operates the programming PC is represented by this interface.</td>
<td>Human Interaction.</td>
</tr>
<tr>
<td>IF 11</td>
<td>Mounting</td>
<td>The means by which a unit is mounted onto a structure.</td>
<td>Mechanical.</td>
</tr>
<tr>
<td>IF 12</td>
<td>Inspection</td>
<td>The method by which personnel do inspections.</td>
<td>Human Interaction.</td>
</tr>
<tr>
<td>IF 13</td>
<td>Maintenance</td>
<td>The method by which personnel do maintenance on units.</td>
<td>Human Interaction.</td>
</tr>
<tr>
<td>IF 14</td>
<td>Installation</td>
<td>The means by which personnel do installations.</td>
<td>Human Interaction.</td>
</tr>
<tr>
<td>IF 15</td>
<td>Management</td>
<td>This is the way management interacts with the system.</td>
<td>Human Interaction.</td>
</tr>
</tbody>
</table>

4.5 Resource and Requirements Allocation

Now we will relate the identified requirements to the identified functional units and interfaces. Performing this has numerous advantageous associated with the development. It provides a decision database for development. Through this database design decisions, relations, and so forth can be tracked which makes ideal for backtracking purposes, error detection and fault correction mechanisms.
A second advantage is that it ensures the developed system and three devices conform to all the identified requirements.

The resource allocation tables for this design phase are given in Table 4-7 and Table 4-8 respectively.

---

### Table 4-7: Functional Units Requirements Allocation - Conceptual Design.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Allocated Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 1.1</td>
<td>CR 1, CR 4, CR 5, CR 6, CR 7, CR 8, CR 9, TR 1, TR 3, TR 4, TR 6, TR 7, FR 13, FR 14, FR 15, FR16, FR 17</td>
</tr>
<tr>
<td>F/U 1.2</td>
<td>CR 2, CR 4, CR 8, CR 9, TR 2, TR 5, TR 6, FR 13, FR 14, FR 15, FR16, FR 17, FR 19</td>
</tr>
<tr>
<td>F/U 1.3</td>
<td>CR 3, CR 5, CR 7, TR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 2</td>
<td>CR 15, CR 23, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 3</td>
<td>CR 23, FR 8, FR 13, FR 14, FR 15, FR16, FR 19</td>
</tr>
<tr>
<td>F/U 4</td>
<td>CR 19, CR 24, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 5</td>
<td>CR 24, FR 13, FR 14, FR 15, FR16, FR 19</td>
</tr>
<tr>
<td>F/U 6</td>
<td>CR 23, FR 13, FR 14, FR 15, FR16, FR 19</td>
</tr>
<tr>
<td>F/U 7</td>
<td>FR 7, FR 13, FR 14, FR 15, FR16, FR 19</td>
</tr>
<tr>
<td>F/U 8</td>
<td>FR 3, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 9</td>
<td>FR 10, FR 12, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 10</td>
<td>FR 10, FR 11, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 11</td>
<td>FR 19, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 12</td>
<td>FR 19, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 13</td>
<td>FR 17, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 14</td>
<td>FR 17, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 15</td>
<td>FR 17, FR 13, FR 14, FR 15, FR16, FR 19</td>
</tr>
<tr>
<td>F/U 16</td>
<td>FR 6, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 17</td>
<td>FR 18, FR 13, FR 14, FR 15, FR16, FR 18</td>
</tr>
<tr>
<td>F/U 18</td>
<td>FR 4, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 19</td>
<td>FR 2, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 20</td>
<td>FR 5, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 21</td>
<td>FR 1, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 22</td>
<td>CR 1, CR 4, CR 5, CR 8, TR 1, TR 6, TR 7, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 23</td>
<td>CR 2, CR 15, CR 16, CR 17, CR 18, TR 2, TR5, TR 6, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 24</td>
<td>FR 8, FR 9, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 25</td>
<td>FR 7, FR 9, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>F/U 26</td>
<td>FR 17, FR 13, FR 14, FR 15, FR16</td>
</tr>
</tbody>
</table>

### Table 4-8: Interfaces Requirements Allocation - Conceptual Design.

<table>
<thead>
<tr>
<th>Interface Number</th>
<th>Allocated Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF 1</td>
<td>CR 4, FR 13, FR 14, FR 15</td>
</tr>
<tr>
<td>IF 2</td>
<td>CR 11, FR 16, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 3</td>
<td>(detail requirement from client), FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 4</td>
<td>CR 9, FR 13, FR 14, FR 15</td>
</tr>
<tr>
<td>IF 5</td>
<td>FR 3, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 6</td>
<td>FR 12, FR 13, FR 14, FR 15</td>
</tr>
<tr>
<td>IF 7</td>
<td>FR 19, FR 13, FR 14, FR 15</td>
</tr>
<tr>
<td>IF 8</td>
<td>FR 17, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 9</td>
<td>FR 17, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 10</td>
<td>FR 6, FR 17, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 11</td>
<td>FR 18, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 12</td>
<td>FR 4, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 13</td>
<td>FR 2, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 14</td>
<td>FR 5, FR 13, FR 14, FR 15, FR16</td>
</tr>
<tr>
<td>IF 15</td>
<td>FR 1, FR 13, FR 14, FR 15, FR16</td>
</tr>
</tbody>
</table>
4.6 Design Synthesis

Thus far, the development has been done from a system viewpoint. To perform a preliminary design on the three devices, we need to synthesize the current level of design from the viewpoint of the devices.

This design synthesis can be thought of as a transition link between the conceptual and preliminary design phases. It defines the areas where further development needs to be done. Figure 4-5 illustrates the functional architecture for the development from the viewpoint of the three devices.

In this figure, new functional units and interfaces are shown. These new units and interfaces form part of the preliminary design and are formally described and defined in Chapter 5. It is units and interfaces not applicable on a system level and therefore not seen in the functional analysis of the conceptual design. A short description of these units and interfaces are:

- F/U 27 represents a USB flash drive.
- IF 16 describes the physical contact between the OmnipoMote and OmnipoController.
- IF 17 portray the physical contact between the OmnipoMote and OmnipoBatteryPack.
- IF 18 represents the interface between the USB flash drive and the OmnipoController.
4.7 Design Review and Feedback

Design reviews, as required by the process model, were performed during the conceptual design phase. Insight obtained through fault detection and error correction was used to improve the design. The most important feedback items while performing these reviews were:

- Continuous analysis of client requirements as a result of change. The changes came as a result of better understanding the need, technologies and feasibility characteristics of the system that required new and/or improved requirements.
- Improvements on functional architectures and interfaces for a more accurately description.
- Numerous iterations were required to accurately state which aspects are appropriate to this design phase. In the beginning, a lot of information was used as conceptual, when they actually were information appropriate for a preliminary design level, and vice versa.

In all, the design reviews did function as anticipated by aiding in the validation and verification aspects of the research, concerned with this level of design.

4.8 Conclusion

The conceptual design was presented in this chapter. Here the needs and objectives of the client were transformed to a system concept capable of living up to the expectations of the client. A literature study on the different technologies and operational feasibility of the system revealed key development considerations, for this conceptual level.

Requirements that came from the literature study and client need analysis were used to develop a functional description of the system in the form of a functional architecture. The identified requirements and units of the functional analysis were related to one another, to establish a decision database.

Based on the requirements allocation tables and development scope, a synthesis was completed on the system functional analysis. This was completed to identify all the relevant units of the system that requires further development in the preliminary design phase.

All the necessary design reviews and feedback were completed to ensure that the conceptual design is finished with high standards. Up to now the SE methodology is living up to expectation, because we were able to define a system concept that will satisfy the needs of the client. However, since we did not receive any performance criteria from the client, it is unsure how this developed architecture will perform. The developed interfaces and architectures will satisfy the needs of the client given the inputs we had available for this development phase.
Chapter 5

Preliminary Design

The preliminary design phase of the development is performed in this chapter. Here, emphasis is on the development of functional analyses and solutions for the synthesis of the previous design phase.

5.1 Introduction

Relationships between all the different kinds of system resources were given in the previous chapter. The fit of a single specified entity will become clear if one were to see these relationships from the viewpoint of that entity. The design synthesis of the previous chapter did this for the three devices.

It is now possible to establish the function of the devices knowing both the fit and operational outcomes thereof. The operational outcomes come from the need and objective of the client, as well as further literature studies. Studies on technologies, objectives, feasibility outcomes, government regulations associated with the system and outputs of the conceptual design, are inputs for this design phase.

In this chapter, we will first examine these inputs to define exactly how they are used in this design phase. A literature study is then performed on the different types of inputs. Design and development requirements are drawn from the literature to aid in the development of more detailed functional analysis. The requirements and functional analysis are related using resource allocation tables.

These tables give a perfect summary of the fit and function, and are used in a trade-off analysis to find preliminary solutions for units. A design synthesis summarizes the development of this chapter. Formal and informal reviews are also performed throughout the life-cycle of this design phase.
5.2 Conceptual Design Output Analysis

Outputs from the conceptual design phase are analyzed using the design synthesis and resource allocation tables of the previous chapter. The design synthesis revealed system units that are subjected to further development in this design phase. Requirements associated with these units require closer examination for a detail analysis on their function.

The following are to be discussed in the following sections by means of a literature study:

1. Technologies used in the system.
2. Functional research in on the units defined by the previous design synthesis.
3. Feasibility research on the feasibility criteria of the system.
4. Research on electro-magnetic compatibility (EMC) –and government specifications.

No more analysis on software architectures, except for the programming apparatus and programming software, is given since no firmware is required for these devices. The programming apparatus enforce design requirements on the devices and the programming software may require a specific programming apparatus.

5.3 Literature Study and Analysis

All the studies completed on the inputs of this design phase are given in this section.

5.3.1 Associated Technologies

A more detailed explanation of the technologies involved in the system is given in this subsection. The most important of these technologies are WSNs and DSNs, IEEE 802.15.4 and motes.

5.3.1.1 WSNs and DSNs

WSNs usually consist of a large number of small wireless communicating nodes. Applications for these networks range from industrial sensor networks, volcano monitoring, habitat monitoring, to military and medical applications. The end application of these networks has a great influence on the deployment of the nodes, as well as the hardware and firmware aspects thereof. [16]
Node deployment also depends on the environment in which the nodes are deployed. Fixed strategies are required to establish a baseline for node deployment and network topology recognition. Depending on the application and environment, the following scenarios would highlight this point [11][16]:

1. During military applications, the nodes might be dropped from an aeroplane and there is no possible way of controlling the position in which they will fall.
2. The environment in which nodes are deployed differs from application to application.
3. Cases exist where sensor nodes are mobile, like inside buildings with toxic leaks.
4. Coverage areas and node numbers also differ between applications and environments.
5. In some applications, nodes might not work upon deployment or may fall outside the planned coverage area, like in the case of an aeroplane deployment.

It becomes evident that knowledge of the application, application environment and method of deployment is necessary. Topology recognition of nodes after deployment should occur automatically, as the nodes would be distributed in random positions. Therefore, nodes must be capable to adapt to the environment automatically using topology recognition software. However, this software depends on the hardware of the node. [16][17]

As mentioned, hardware resources are uncommon in these nodes, which imply that certain platforms can be used for a variety of applications, while others cannot. Projected costs associated with development, production, operation, network scalability, number of nodes, reliability and efficiency all influence the design of a sensor node. Therefore, the design or choosing of a node for a specific application, should be done carefully. [11][18]

Once in operation, a few techniques can be implemented in software to control the reliability and efficiency of these networks. Nodes with the highest power should control data flow and compute the more complex operations within the network, while other nodes are only used for sensing, and in some cases, low complex computations. This is also known as energy control. [19]

Energy control is a key feature in WSNs, because the platforms are battery powered. In some cases, it is impossible to change the batteries due to the location of deployment. Protocols and other communication aspects must use as little power as possible without affecting system reliability. [17][19]

5.3.1.2 IEEE 802.15.4

To fully understand the IEEE 802.15.4 protocol, we need to look at the wireless personal area network (WPAN), the PHY layer and the MAC layer of the protocol.
5.3.1.2.1 IEEE 802.15.4 WPAN

Low power usage, low data rates and inexpensive platforms are assured by the protocol. It is ADHOC and self-organizing for both stationary and moving nodes. [20]

It was stated in Chapter 4 that the protocol defines two types of devices: an FFD and an RFD device. An IEEE 802.15.4 WPAN will always include at least one FFD that operates as a personal area network (PAN) coordinator. A PAN coordinator is a central controller (gateway) and usually gets its power from mains supply, as it is processing intensive. There are also normal FFD coordinators together with the PAN coordinator. These coordinators form their own PANs within the overall network. A RFD can only communicate with an FFD, but an FFD can communicate with both RFDs and FFDs. RFDs are also known as end nodes. Figure 5-1 shows the IEEE 802.15.4 WPAN. [19][20]

The figure describing the IEEE 802.15.4 WPAN shows that there are three network topologies possible within the protocol stack: star network, cluster-tree and peer-to-peer. [20]

![Figure 5-1: IEEE 802.15.4 WPAN.](image)

5.3.1.2.2 IEEE 802.15.4 PHY

The PHY layer of the protocol consists of the RF transceiver and other low-level mechanisms. A PHY data service and PHY management service is provided by this layer. PHY protocol data units (PPDU) are received and transmitted by the PHY data service across the radio channel (air) and the management service is the interface to the physical layer management entity (PLME). This layer has the following features [20]:

- Radio transceiver activation and deactivation
- Energy detection (ED)
- Link quality indication (LQI)
- Communication channel selection
• Clear channel assessment (CCA)
• Packet transmission and reception

Receiver ED is an estimation of the received signal strength within the bandwidth of a channel. It can be used by the network layer for channel selection. Methods used by the layer to perform CCA are energy above threshold, carrier sense only or carrier sense with energy above threshold. [20]

The PHY layer of the IEEE 802.15.4 protocol has a high and low frequency option to choose from, and both use direct sequence spread spectrum (DSSS). All available options are listed in Table 5-1. [20]

### Table 5-1: IEEE 802.15.4 Frequency Options. [20]

<table>
<thead>
<tr>
<th>Carrier Frequency</th>
<th>Data Rate (kbps)</th>
<th>Channels</th>
<th>Receiver Sensitivity (dBm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.4GHz</td>
<td>250kbps</td>
<td>16 (2.4GHz - 2.4835GHz)</td>
<td>-85</td>
</tr>
<tr>
<td>915MHz</td>
<td>40kbps</td>
<td>10 (902MHz - 928MHz)</td>
<td>-92</td>
</tr>
<tr>
<td>868MHz</td>
<td>20kbps</td>
<td>1 (868MHz - 868.6MHz)</td>
<td>-92</td>
</tr>
</tbody>
</table>

5.3.1.2.3 IEEE 802.15.4 MAC

As with the PHY layer, there are two services that the MAC layer provides: MAC data service and MAC management service. The management service is the interface to the MAC sub-layer management entity (MLME) service access point (SAP) (MLMESAP). [20]

Transmissions and receptions of all MAC protocol data units (MPDU) across the PHY data service are enabled by the MAC data service. The MAC sub-layer has the following features associated with it [20]:

• Beacon management
• Channel access
• Guaranteed time slot (GTS) management
• Frame validation
• Acknowledged frame delivery
• Association and disassociation

The MAC layer uses beacons for communication synchronization. If an FFD is enabled to transmit beacons it is the PAN coordinator, otherwise it is just a normal FFD device. However, normal FFD devices can transmit beacons at a later stage to form their own PAN, but only after it has successfully been associated with the PAN coordinator of the network. [20]

Beacon frames are at the boundary of the overall super-frame or communication frame. The super-frame is consists of 16 slots and can have both inactive and active portions. In inactive portions, the PAN coordinator can enter into sleep mode, thus saving power by not interfacing with its PAN. The
beacon frame is transmitted in the first slot of this super-frame for synchronization. In the active portion of the super-frame, communication with the PAN coordinator is based on the following methods [20]:

1. In the contention free period (CFP) there are guaranteed time slots for the devices.
2. During contention access periods (CAP) devices compete for the channel, using carrier sense multiple-access with collision avoidance (CSMA-CA).

This is the basic description of how the IEEE 802.15.4 MAC layer works and operates. The working of the different sections, how data is transmitted and the actual structure of the frames is too complex for this development phase. [20]

5.3.1.3 Motes

Firmware compatibility between the RF unit and the TelosB mote is required by the client. Thus an description of the characteristics of this TelosB mote is thus given. The TelosB mote uses a 250kbps, 2.4GHz, IEEE 802.15.4 Chipcon wireless transceiver (the CC2420) as the communication radio and a Texas Instruments MSP430F1611 microprocessor for an electronic control unit (ECU). Data collection and device programming is supported through a USB interface. It also includes temperature, light and humidity sensors. [13]

5.3.1.4 Other

There are also other technologies involved in the development. One such technology is universal serial bus communication, but is handled completely by specific components as we will see later in this chapter. Therefore, an extensive literature study was not performed on USB communication.

Studies on WSNs, IEEE 802.15.4 and the TelosB mote were necessary because it required deeper understanding for its correct use in the development. It also gives a better understanding of the application environment of the three devices, because it is not technologies we usually work with on an everyday basis, but the other (USB, C programming, etc.) we are truly familiar with.

5.3.2 Functional Background

The rationale behind the study presented in this section, is to better describe the functional purpose of three devices and their boundary units, as defined by the synthesis of the previous design phase.
5.3.2.1 RF Unit - OmnipoMote

This device is responsible for wireless RF communication using the IEEE 802.15.4 protocol. The client requires the mote to use a ceramic chip antenna and the device must be small in size (3cm x 5cm).

Currently the client does firmware development for this system on the TelosB mote. There is therefore a need that the firmware be interchangeable between the TelosB mote and the OmnipoMote. This would make portability easier, save development time and costs.

It would be best to use the same hardware configuration between the devices to make software and firmware interchangeable between two devices. This will enforce minimal changes in the firmware. Our client also requires that the device include a motion activation and temperature sensor. The motion activation sensor must wake up the unit from sleeping to active mode, when it is moved.

This is only a communication unit and needs to be connected to the mini-controller to transfer advertisements to the displays. If operating as a mote, the battery pack is used. It depends on the mini-controller to transfer data between itself and other peripherals. Based on the need of the client, the device has the following functional tasks:

- Wireless communication using the IEEE 802.15.4 protocol.
- External wake-up using the tilt sensor and temperature sensing.
- Communication with the mini-controller unit.
- Computation capabilities for RFD and FFD modes.

5.3.2.2 Mini-controller - OmnipoController

Numerous operational functions required to make the system a reality is made possible through this device. It controls the flow of data between the RF unit and USB peripherals.

No functional or form requirements were received for this device, but the client gave us the freedom for its development. However, the following functional requirements are required:

- Direct communication to a USB flash drive for file storage and handling.
- Release the connection of the USB flash drive, and transfer handling of it to the USB host.
- Send and receive data to and from a PC via USB.
- Send and receive data to and from the RF unit.
- Communication with other peripherals in the future is required.

Some of the above functions require support architecture for realization. A special unit is required to switch the use of the USB flash drive between the electronic control unit of the OmnipoController and the USB host.
The USB flash drive can be used for data storage, which eliminates the use for extra memory. It would be great if there were manual mode selections on the device, to control the operating mode of the OmnipoController between diagnostics, RFD, PAN coordinator or coordinator.

5.3.2.3 Battery Pack – OmnipoBatteryPack

During mote applications, the RF unit draws its power from the battery pack. The client requests that a battery pack small in size and only limited by the size of the batteries to used. ZaPOP requires that the battery pack use rechargeable AAA batteries.

5.3.2.4 Boundary Units

The boundary units of the three devices have the ability to interface the devices directly and therefore the design. How these units influence the design will become clear in this subsection.

Note: Most of the information given in this section does not come from literature, but from experience and knowledge obtained through implementing and designing the devices. It is information fed back into early decision making via feedback and design reviews.

5.3.2.4.1 IF 1 – IEEE 802.15.4 Wireless Communication

Operational functions enforced by this unit were given in Sections 4.2.2 and 5.3.1.2.

5.3.2.4.2 F/U 2 (IF 2) - USB Cable

This is the cable that connects the mini-controller with a USB host (PC or digital display). USB sockets on these hosts are type-A female sockets. Thus, a cable need to be chosen that has a type-A male connector on the one end and on the other end either a male type-A, type-B or mini-USB connector.

5.3.2.4.3 F/U 3 or F/U 6 – USB Host Peripherals (F/U 24 USB Host Software)

Host peripherals in this synthesis are either PCs or digital displays. There are no known constraints on the design by these units other than the communication protocol and commands to be received by the mini-controller. The communication protocol is USB and the final commands will be established once the devices are approved for functional capability.
5.3.2.4.4 F/U 4 (IF 3) - Peripheral Cable

No information is available on the cables used to attach peripherals to the mini-controller in the future, but it should have provisions for this in the form of open pads.

5.3.2.4.5 F/U 5 – Attachable Peripherals

These peripheral devices do not yet exist and therefore do not impose any design requirements. However, we assume that they will have their own power, and that only signal and ground connections will be required when these devices are connected to the mini-controller.

5.3.2.4.6 F/U 12 (IF 7) – Power Cable

Connections between the mini-controller and power source are established by this cable. The cable must deliver the correct power as required by the mini-controller and therefore we can assume that it may have a transformer and rectifier circuit included, just like laptop power cables.

5.3.2.4.7 F/U 11 – Power Source

The power source is the mains power supply, 220V-240V 50Hz.

5.3.2.4.8 F/U 13 – Programming Apparatus

This unit describes the programming device (JTAG, PicKit2, etc.). By implementing this unit into our system, we open ourselves to do research on different programming devices (corresponding to the manufacturer of the microprocessor used) and select a programmer before designing the PCB boards. Once we have chosen a programmer, can we go ahead and make provisions for connecting it with the PCB boards of the devices. A programming apparatus for each programmable component is required.

5.3.2.4.9 F/U 14 (IF 9) – USB Cable

The cable used to connect the programming apparatus to the programming PC depends on the chosen apparatus, and it usually comes with it when purchased. From this unit we learn how personnel should use the cable when programming the devices.
5.3.2.4.10 F/U 15 – Programming PC (F/U 25 Programming PC Software)

This software is the integrated development environment (IDE) in which firmware is written for devices. IDEs for every type of microprocessor and programming adapter needs to be chosen.

5.3.2.4.11 F/Us 16, 18, 19, 20 and 21 – Personnel and System Manager(s)

These personnel have the ability to come in direct contact with the three devices and boundary units upon request from the system manager(s). Operational actions that these personnel must complete when handling any of the three devices can aid in making the devices more user friendly.

5.3.2.4.12 F/U 27 – USB Flash Drive

USB flash drives are well known, as we use them on an everyday basis. Research would have been required on how to interface these devices from a microcontroller, but as we will see later a component exists that does this for us. Therefore no extensive studies are needed on USB flash drives other than making sure it connects correctly to the mini-controller.

5.3.3 Operational Feasibility

In the previous design phase, the criteria for operational feasibility only consisted of reliability, efficiency, durability, and user friendliness. Maintainability of units, firmware/software compatibility and testability forms part of this criterion now that prominence is on individual units of the system. The devices also need to be compatible with government regulations and EMC requirements.

5.3.3.1 Performance Criteria

No performance criteria were received from the conceptual design.

5.3.3.2 Economic Feasibility

Economic aspects of the research are completed by the client.
5.3.3.3 Reliability

System reliability is not just assured by the reliability of one aspect, but by all system aspects. The following can make the whole system more reliable [2] [21]:

1. Follow the procedures for software and firmware compatibility.
2. Use common and well known components during development.
3. Use well known CAD tools during PCB layouts and prototype modelling.
4. Continuous documentation of design baselines.
5. Design user-friendly interfaces through which maintenance can be carried out.

5.3.3.4 Maintainability

System maintainability is ensured through effective maintenance procedures that are divided into either corrective or preventive maintenance categories. Corrective maintenance is carried out, at unscheduled times due to failures that occur without notice, with the aim of restoring the system to full operation. It is time consuming and results in the lost of system availability, as failures need to be identified and corrected. [2]

Preventive maintenance ensures that corrective maintenance is not necessary by detecting possible errors before they occur. This maintenance procedure are incorporated into the system from the beginning and as such do not result into a loss of system availability. Software, hardware and other parts of the system are replaced with new or upgraded versions periodically as a result. [2]

The system and product must be designed to incorporate both categories of maintenance to ensure that the system is serviceable and repairable [21].

5.3.3.5 Usability

System usability is a very important and sometimes neglected factor of system effectiveness. Usability requirements are not only used in the detailed design of resources, but also in the definition of operational procedures. Operational procedures describe how people use, handle, operate, dispose, and so on of system resources. Therefore, system effectiveness is also assured by its usability [2]

Product usability also affects the effectiveness and productiveness of people that interacts with it. A negative or positive alteration in the psychological state of human minds comes from experience in interacting with things. This interaction is not just in smell, sight and hearing, but also in all forms of anthropometric - , sensory - and physiological factors. [2] [21]
5.3.3.5.1 Anthropometric Factors

These factors consider the physical dimensions of the human body like weight, height, arm reach, hand size, and so forth. Information provided by these factors aids in the development of operating stations, panels, and in our case, instalment areas and maintenance. [2][7]

5.3.3.5.2 Human sensory Factors

Sensory capabilities of humans, like vision, hearing, smell and touch, all influence the way we interact with a device. [2]

5.3.3.5.3 Physiological Factors

The environment and user friendliness of units affect the physiological state of humans. Extreme temperature, humidity, vibration, close spaces, hard to reach places, and many more affect the physiological state of the human mind. [2]

5.3.3.6 Manufacturability

Product parts and gravity are sources that can impact manufacturing negatively. The following steps can ease manufacturing, save time and money [2][7]:

1. Use parts that are light, because they are easier to work with.
2. Use as few parts as possible.
3. Tolerances of parts used must be compatible with the assembly methods.
4. Use parts and components that are standard, because this eliminates the need and cost of developing nonstandard parts.
5. Add more functions per part to keep costs down and to use fewer parts.
6. Single sided PCB boards with the minimum required layers keeps costs down and eases manufacturing. Assembly costs rapidly get more expensive when components are placed at both sides of the board and when more layers are added.

After manufacturing is finished, all parts need to be assembled to realize a working product. Criteria to make the assembly processes easier are the following [2]:

1. Specify parts that can be automatically assembled.
2. Avoid parts that are not supplied pre-oriented in reels, tubes or matrix arrays.
3. Minimize directional changes in assembly direction. Develop parts that can be assembled in one direction such as the top-down approach.
4. Use standard parts, standard processes and standard assembly techniques.
5. Parts should be designed in such a way that they can be disassembled easily.
6. Use and/or design parts that is easily graspable by the automated handling purposes.
7. Use stiff components and not flexible ones.

5.3.3.7 Firmware/Software Compatibility

Firmware portability, readability, constructability and traceability are improved using the right development procedures. Development of high quality firmware and software can be accomplished using the following procedures [7][22]:

1. Produce pseudo-code images of software (high-level language of the code outline).
2. Execute the code in your head for conceptual verification.
3. Implement the pseudo-code in actual microprocessor language.
4. Integrate, evaluate and test code to verify the program flow and interactions between segments.
5. Document all developments, changes and results for future work and reference.

The following guidelines can make firmware and software easily maintainable, upgradeable and more effective [22]:

- **Code portability** is achieved, because pseudo code is derived from high-level functional analysis that creates a common abstraction level between all programming languages.
- **Code readability** is accomplished as a result of using pseudo code to represent the program flow. It is made more effective when indentations, comments, and descriptive names for functions and variables are used.
- **Code construction** is the ability to write code in structures such as objects and procedures which can be designed, evaluated and tested separately, before being integrated.
- **Code traceability** is achieved through good documentation that describes how firmware code is linked to the functional analysis. Descriptive names that are unique for all source files, variables, functions, and the use of known compilers and CAD tools will help to achieve this.

5.3.3.8 Testability

Testing procedures are either automated, or carried out with human interaction. Whatever the means, interfaces are required to view and interpret the results. Tests are defined in terms of visibility and
controllability. Test visibility defines how humans can observe outputs and internal states of devices, and the ability of a human to subject the test to faulty inputs defines its controllability. There are different categories of testing [2] [21]:

1. Design reviews in the design process.
2. Analytical evaluation (CAD models) is used to verify design relationships.
3. Breadboards, engineering and software models (level 1) are used to test and verify relationships, concepts and interfaces.
4. Pre-production prototypes (level 2) serve as the initial qualification of the system and/or product.
5. Production prototypes (level 3) test the utilization outcomes of the system and/or product.

Each level of testing is carried out by different personnel. As the development moves from the laboratory to the operational environment, tests are carried out first by the development team and in the later by the end users of the product. In each level of testing, results appropriate for the level need to be obtained for interpretations and analyses. These are some of the different kinds of tests [2]:

1. Technical tests to verify the functional outcomes of an entity.
2. Performance tests to verify individual characteristics.
3. Environmental verification in terms of temperature, shock, vibration and so forth.
4. Structural tests to reveal the characteristics of materials.
5. Reliability qualification in terms of maintenance time, component life-time and so on.
6. Maintainability demonstrations to reveal the maintenance characteristics of the system.
7. Support equipment compatibility tests to verify the interaction of equipment and resources.
8. Personnel testing and evaluation to verify relationships between devices and personnel.
9. Software verification to test both operational and maintenance software.

In this development design reviews, CAD models and pre-production prototypes are used for testing designs. For functional capability, it is only necessary to perform technical verification and all tests are carried out by the developing engineer.

5.3.3.9 Environmental Compatibility

Environmental factors include temperature, humidity, vibration, noise, and so forth. These can affect the psychological state of person, but also have an impact the performance of equipment. Based on the development there are two environments: the operational environment and laboratory environment.
5.3.3.9.1 Operational Environment

This environment is defined by the convenience markets and depot-and-store facilities that resemble the final operational locations of the three devices.

Operational sites and depot-and-store facilities pose many obstacles to the devices like trolleys, shelves, different selling goods, cellular phones, light sources and radio communications sources, to name but a few. These pose a threat to the efficiency and reliability of the RF unit and mini-controller operations.

In our development scope, it is not necessary to include the effect of these into the design, but it is necessary to know of their existence. Knowing their existence and possible influence will open areas of improvement and help to design for the future.

5.3.3.9.2 Laboratory Environment

The laboratory environment will not have the many obstacles as in the operational environment. In this environment, the only sources of possible obstacles are computers. This environment mimics the actual operational environment in the true sense of a perfect scenario (no external influences, like trolleys that effect the communication for example).

5.3.3.10 Electromagnetic Compatibility

A serious form of environmental pollution in electronic circuits is electro-magnetic interference (EMI) and the designer must always strive to remove all sources of EMI, to achieve the desired EMC. EMC at board level is not enough as the circuit board is also part of the system and as such can radiate noise to other parts in the system or vice versa. Early consideration to EMC requirements is the most cost-effective way to design. [23]

The electromagnetic environment consists of three elements: EMI source, coupling path and receptor, as shown in Figure 5-2. The source resembles causes of EMI, the coupling path refers to conductors and connection lines, and the receptor symbolizes components susceptible to EMI. [23]

![Figure 5-2: Elements of EMI. [23]](image-url)
The EMI source and coupling path must be designed to minimize EMI emissions, where the receptor and coupling path must be designed with low EMI susceptibility. The selection of components, PCB layouts, and so on all affect the EMC of a product.

5.3.3.10.1 Component Selection

Component selection constitutes a major part in the EMC of a circuit, because each individual electronic component has its own unique EMC characteristics. The following can be done to enhance the EMC of a circuit [23]:

- Component Packages: Surface mount components and leadless packages are preferred over leaded packages.
- Resistors: Use surface mount resistors, or carbon film resistors if the design requires it.
- Capacitors solve many EMC problems: In low frequency designs use aluminium and tantalum capacitors, for mid frequencies (kHz to MHz) use ceramic capacitors and for high frequencies use low-loss ceramic and mica capacitors. Also use bypass and decoupling capacitors to enhance the EMC of the circuit.
- Inductors: There is a minimum difference between the EMC of leaded and leadless inductors, but the most effective is closed loop core inductors.
- Diodes: All diodes have different EMC characteristics. The type of diode to choose, depends on the application.
- Integrated circuits (ICs): CMOS technologies are preferred, but never use a combination of packages like CMOS and TTL. Surface mount packages affects EMC positively.
- Voltage regulators: Always use adequate decoupling capacitors and place them as close as possible to the output of the regulator.
- Line terminators: Impedance matching between source and destination is very important, especially for high-speed circuits.
- Microcontroller circuits: The power supply circuit must be put close to the microcontroller. Pull-up and pull-down resistors minimize noise on the input/output (I/O) pins of the microcontroller, especially for the reset and IRQ pins.

5.3.3.10.2 PCB Layout Techniques

PCB layouts cause more EMC problems than the components on it. It adds no extra cost to design a good circuit board from the beginning since the board is part of the system. All the tracks on the circuit board have a resistance, capacitance and inductance. The designer must take great care in considering the following [23]:

---

Development of a modular low power IEEE 802.15.4 wireless sensor module
1. Crosstalk is minimized by increasing the separation between tracks.
2. PCB capacitance is maximized when ground and power lines are in parallel.
3. High frequency and sensitive tracks must be placed away from noisy power tracks.
4. Large track widths reduce the power and ground impedance.

General design techniques for PCB layouts are [23]:

1. **Segmentation:** Physical separations of different circuits sections, like separating the digital circuit from the analogue circuit.
2. **Decoupling:** Reduces noise along the supply rail.
3. **Return Paths:** The EMC performance of a PCB increases as the impedance of the return paths are made smaller.
4. **Trace Separation:** Crosstalk is minimized by the use of trace separation.
5. **Guard and Shunt Traces:** These increase the protection of clock lines against EMI sources.
6. **Ground Techniques:** A very wide (at least 1.5mm) ground track on single layer circuits is necessary. On multilayer or double-layer circuits a ground plane is used.
7. **Track Layout Techniques:** The amount of vias on the board should be kept to a minimum. Track turns should be 45 degrees with a constant width.
8. **Filtering and Shielding:** I/O lines can be filtered using a combination of capacitors connected to the ground together with common mode chokes. Sensitive and noisy areas can be enclosed by shielding methods.

**5.3.4 Governmental Regulations**

Authorities exist that enforce specifications and regulations on designs. In our development, we have identified radio regulations and EMC regulations as points of concern.

**5.3.4.1 Radio Regulations**

The RF unit operates in the 2.4 GHz ISM band. This is a license free band subjected to certain technical restrictions. The radio transceiver used in the design is certified for this spectrum. [24]

**5.3.4.2 EMC Regulations**

It will be beneficial for the overall project if the devices pass EMC certifications due to large quantities involved in the implementation. However, for the development scope it is not necessary as the devices are only tested for functional capability.
The devices can be tuned and tailored to meet the specifications enforced by EMC regulation authorities, once the ADM model is developed. For our development, the procedures set out in Section 5.3.3.10 will be sufficient to ensure a good EMC number for the devices.

5.4 Requirements Analysis

In the previous section, a literature study on the relevant technical and design aspects of the development was given. During the first iterations of this design phase some requirements (WSNs, DSNs and IEEE 802.15.4) have been extracted that were of no use in the design phase, and therefore not given as requirements in ongoing design iterations.

Components were found in the trade-off analysis that satisfied those requirements automatically, but this does not mean that the conducted studies were unnecessary. In contrast, the studies helped us to understand the working of those technologies so that we can better use the components.

Requirements will now be given per unit, as the purpose of the chapter is to develop architectures for specific units. The requirements are given in Table 5-2 to Table 5-12.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-RF 1</td>
<td>Shall be designed to use the IEEE 802.15.4 communication capabilities.</td>
</tr>
<tr>
<td>PR-RF 2</td>
<td>Shall use a ceramic chip antenna.</td>
</tr>
<tr>
<td>PR-RF 3</td>
<td>Shall be small in size, with a maximum dimension of 3cm x 5cm.</td>
</tr>
<tr>
<td>PR-RF 4</td>
<td>Firmware shall be compatible with the TelosB mote.</td>
</tr>
<tr>
<td>PR-RF 5</td>
<td>Shall be used for FFD operational capabilities.</td>
</tr>
<tr>
<td>PR-RF 6</td>
<td>Shall be used for RFD operational capabilities.</td>
</tr>
<tr>
<td>PR-RF 7</td>
<td>Shall receive and send data to and from the mini-controller</td>
</tr>
<tr>
<td>PR-RF 8</td>
<td>Shall receive power from the mini-controller</td>
</tr>
<tr>
<td>PR-RF 9</td>
<td>Shall receive power from the battery pack</td>
</tr>
<tr>
<td>PR-RF 10</td>
<td>Shall be capable of processing data.</td>
</tr>
<tr>
<td>PR-RF 11</td>
<td>Shall include an activation sensor, for device wake-up from sleep mode.</td>
</tr>
<tr>
<td>PR-RF 12</td>
<td>Shall include a temperature sensor for accurate temperature reading.</td>
</tr>
</tbody>
</table>

Note. The designator “PR-RF x” stands for Preliminary Requirement – RF plus number.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-MC 1</td>
<td>Shall be able to communicate with a USB flash drive.</td>
</tr>
<tr>
<td>PR-MC 2</td>
<td>Shall be capable of communicating with a USB host.</td>
</tr>
<tr>
<td>PR-MC 3</td>
<td>Shall be able to switch usage of USB flash drive between the USB host and onboard ECU</td>
</tr>
<tr>
<td>PR-MC 4</td>
<td>Shall be able to switch USB host communication on and off.</td>
</tr>
<tr>
<td>PR-MC 5</td>
<td>Shall receive and send data to and from the RF unit.</td>
</tr>
<tr>
<td>PR-MC 6</td>
<td>Shall deliver power to the RF unit.</td>
</tr>
<tr>
<td>PR-MC 7</td>
<td>Shall have provisional connections in the form of open pads.</td>
</tr>
<tr>
<td>PR-MC 8</td>
<td>Shall use the USB flash drive for data storage.</td>
</tr>
<tr>
<td>PR-MC 9</td>
<td>Shall enable the RF unit to act as a FFD or PAN coordinator.</td>
</tr>
<tr>
<td>PR-MC 10</td>
<td>Shall draw power from the USB host if possible.</td>
</tr>
<tr>
<td>PR-MC 11</td>
<td>Shall have provisions to draw power from the mains supply.</td>
</tr>
<tr>
<td>PR-MC 12</td>
<td>Shall have mode selection switches.</td>
</tr>
</tbody>
</table>

Note. The designator “PR-MC x” stands for Preliminary Requirement – Mini-Controller plus number.
Table 5-4: Battery Pack Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-BP 1</td>
<td>Shall deliver power to the RF unit.</td>
</tr>
<tr>
<td>PR-BP 2</td>
<td>Shall use rechargeable AAA batteries.</td>
</tr>
<tr>
<td>PR-BP 3</td>
<td>Shall protect the unused pins of RF unit.</td>
</tr>
</tbody>
</table>

Note. The designator "PR-BP x" stands for Preliminary Requirement - Battery Pack plus number.

Table 5-5 USB Cable Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-UC 1</td>
<td>A USB Cable shall be used to connect the mini-controller to a USB host.</td>
</tr>
<tr>
<td>PR-UC 2</td>
<td>On this side a type A male shall be used.</td>
</tr>
<tr>
<td>PR-UC 3</td>
<td>Connector on the other side shall not conflict with that of the USB flash drive.</td>
</tr>
</tbody>
</table>

Note. The designator "PR-UC x" stands for Preliminary Requirement – USB H plus number.

Table 5-6 USB Host Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-UH 1</td>
<td>Communication with the mini-controller shall be in the form of commands and file data.</td>
</tr>
<tr>
<td>PR-UH 2</td>
<td>Shall use mini-controller to access the RF unit for wireless transmission and reception.</td>
</tr>
<tr>
<td>PR-UH 3</td>
<td>Shall use mini-controller to for direct access with the USB flash drive.</td>
</tr>
</tbody>
</table>

Note. The designator "PR-UH x" stands for Preliminary Requirement – USB H plus number.

Table 5-7: Power Cable Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-PC 1</td>
<td>A power cable shall deliver power to the mini-controller from the mains supply.</td>
</tr>
<tr>
<td>PR-PC 2</td>
<td>Shall deliver the correct power, in terms of voltage, to the mini-controller from the mains supply.</td>
</tr>
<tr>
<td>PR-PC 3</td>
<td>Shall deliver the correct power, in terms of current, to the mini-controller from the mains supply.</td>
</tr>
<tr>
<td>PR-PC 4</td>
<td>A mains compatible connector shall be used on the one side.</td>
</tr>
<tr>
<td>PR-PC 5</td>
<td>Connector on the other side shall not conflict with other connectors on the mini-controller.</td>
</tr>
</tbody>
</table>

Note. The designator "PR-PC x" stands for Preliminary Requirement – Power Cable plus number.

Table 5-8: Programming Apparatus Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-PA 1</td>
<td>A well-known programming apparatus shall be used for programming components.</td>
</tr>
<tr>
<td>PR-PA 2</td>
<td>Connectors of the apparatus shall be designed for during the detail design.</td>
</tr>
</tbody>
</table>

Note. The designator "PR-PA x" stands for Preliminary Requirement – Programming Apparatus plus number.

Table 5-9: Programming PC Software Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-PCS 1</td>
<td>Well-known IDEs shall be used for programming each type of microcontroller.</td>
</tr>
</tbody>
</table>

Note. The designator "PR-PCS x" stands for Preliminary Requirement – PC Software plus number.

Table 5-10 USB Flash Drive Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-UFD 1</td>
<td>Shall be used by both the USB host and mini-controller for data storage.</td>
</tr>
<tr>
<td>PR-UFD 2</td>
<td>Standard USB flash drive shall be used.</td>
</tr>
</tbody>
</table>

Note. The designator "PR-UFD x" stands for Preliminary Requirement – USB Flash Drive plus number.

Table 5-11 Design Specific Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-DS 1</td>
<td>The procedures in Section 5-11 shall be followed to achieve high reliability.</td>
</tr>
<tr>
<td>PR-DS 2</td>
<td>Designs shall include means to prevent and aid in corrective—and preventive maintenance.</td>
</tr>
<tr>
<td>PR-DS 3</td>
<td>Designs shall not be limited by any form of anthropometric factors of humans.</td>
</tr>
<tr>
<td>PR-DS 4</td>
<td>Designs shall make effective use of human sensory factors.</td>
</tr>
<tr>
<td>PR-DS 5</td>
<td>Devices shall have minimal negative impact on the physiological state of users.</td>
</tr>
<tr>
<td>PR-DS 6</td>
<td>The criteria in Section 5-12 shall be used to ease the manufacturing process and safe costs.</td>
</tr>
<tr>
<td>PR-DS 7</td>
<td>The procedures set out in Section 5-13 shall be used to achieve software/firmware capability.</td>
</tr>
</tbody>
</table>
Table 5-11 continuous

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-DS 8</td>
<td>The testing criteria in Section 5-13 shall be used in all areas of design tests.</td>
</tr>
<tr>
<td>PR-DS 9</td>
<td>For the development scope, the devices shall be compatible to the laboratory environment.</td>
</tr>
<tr>
<td>PR-DS 10</td>
<td>The design procedures in Section 5.3.10 shall be used to achieve high levels of EMC.</td>
</tr>
</tbody>
</table>

Note: The designator 'PR-DS x' stands for Preliminary Requirement – Design Specific plus number.

Table 5-12 System Operational Requirements.

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-SO 1</td>
<td>Programming personnel operational tasks shall be evaluated and the effects thereof included in the design.</td>
</tr>
<tr>
<td>PR-SO 2</td>
<td>Inspection personnel operational tasks and the effects thereof included in the design.</td>
</tr>
<tr>
<td>PR-SO 3</td>
<td>Maintenance personnel operational tasks and the effects thereof included in the design.</td>
</tr>
<tr>
<td>PR-SO 4</td>
<td>Installation personnel operational tasks and the effects thereof included in the design.</td>
</tr>
</tbody>
</table>

Note: The designator 'PR-SO x' stands for Preliminary Requirement – System Operation plus number.

5.5 Functional Analysis

Resource identification comes in the form of an operational analysis and a functional architecture.

5.5.1 Operational Analysis

This operational analysis examines how programming, inspection, maintenance and installation personnel will interact with the system resources defined by the previous design synthesis. The actions of personnel is divided into either operational functions (O/F) and Maintenance Functions (M/F).

5.5.1.1 F/U 16 – Programming Personnel

Basic operational tasks for programming either the RF unit or the mini-controller, is shown in Figure 5-3. A functional description of the operational tasks is given in Table 5-13.

Figure 5-3: Operational Analysis for Programming Personnel.
Table 5-13: Functional Descriptions of Programming Tasks.

<table>
<thead>
<tr>
<th>Operational Task</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O/F a1</td>
<td>Retrieve Unit Package</td>
<td>Personnel must use the travel means provided by the client (flying, vehicle, etc.) to locate and retrieve a package from the depot-and-store facility.</td>
</tr>
<tr>
<td>M/F a2</td>
<td>Inspect Unit Package</td>
<td>The unit package is inspected for defects and shortcomings.</td>
</tr>
<tr>
<td>O/F a3</td>
<td>Install Unit for Programming</td>
<td>On successful inspection, the inspected unit is installed for programming.</td>
</tr>
<tr>
<td>O/F a4</td>
<td>Program Unit</td>
<td>The new firmware is then programmed onto the installed device.</td>
</tr>
<tr>
<td>M/F a5</td>
<td>Evaluate Programming</td>
<td>The new firmware is tested and evaluated.</td>
</tr>
<tr>
<td>O/F a6</td>
<td>Reassemble Unit Package</td>
<td>The unit package is reassembled once programming is finished.</td>
</tr>
<tr>
<td>O/F a7</td>
<td>Store Unit Package</td>
<td>The assembled unit package is stored at the depot-and-store facility.</td>
</tr>
</tbody>
</table>

5.5.1.2 F/U 18 – Inspection Personnel

Operational tasks for inspections are shown in Figure 5-4 and described in Table 5-14.

Figure 5-4: Operational Analysis for Inspection Personnel.

Table 5-14: Functional Descriptions of Inspection Tasks.

<table>
<thead>
<tr>
<th>Operational Task</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M/F b1</td>
<td>Go to Installed Unit</td>
<td>Personnel use the travel means provided by the client to locate a unit of the system.</td>
</tr>
<tr>
<td>M/F b2</td>
<td>Identify Installed Unit</td>
<td>Personnel need to identify the install unit when arriving.</td>
</tr>
<tr>
<td>O/F b3</td>
<td>Inspect Unit</td>
<td>Installation is inspected and the results from diagnostic routines observed.</td>
</tr>
</tbody>
</table>

5.5.1.3 F/U 19 – Maintenance Personnel

Figure 5-5 shows operational tasks for these personnel to complete. They are described in Table 5-15.

Figure 5-5: Operational Analysis for Maintenance Personnel.
Table 5-15: Functional Descriptions of Maintenance Tasks.

<table>
<thead>
<tr>
<th>Operational Task</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M/F c1</td>
<td>Get Package of Damaged Unit</td>
<td>Unit packages marked for maintenance are retrieved.</td>
</tr>
<tr>
<td>M/F c2</td>
<td>Identify Damaged Unit</td>
<td>Before maintenance can start, personnel must first identify the unit.</td>
</tr>
<tr>
<td>O/F c3</td>
<td>Identify Errors and Faults</td>
<td>Personnel must identify the type of errors or failure.</td>
</tr>
<tr>
<td>M/F c4</td>
<td>Identify Level of Maintenance</td>
<td>The correct maintenance level must be identified for the damaged unit.</td>
</tr>
<tr>
<td>M/F c5</td>
<td>Do Unit Maintenance</td>
<td>Maintenance is done according to the criteria for the identified level.</td>
</tr>
<tr>
<td>M/F c6</td>
<td>Reassemble Unit Package</td>
<td>After maintenance, the unit package is reassembled.</td>
</tr>
<tr>
<td>M/F c7</td>
<td>Store Unit Package</td>
<td>The reassembled package is stored and marked ready for use.</td>
</tr>
</tbody>
</table>

5.5.1.4 F/U 20 – Installation Personnel

Figure 5-6 shows operational tasks for installation personnel. These tasks are described in Table 5-16.

Table 5-16: Functional Descriptions of Installation Tasks.

<table>
<thead>
<tr>
<th>Operational Task</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M/F d1</td>
<td>Retrieve Unit Package</td>
<td>Personnel retrieves the unit package of the device that require installation</td>
</tr>
<tr>
<td>M/F d2</td>
<td>Inspect Unit Package</td>
<td>The unit package is inspected to determine if the unit is present.</td>
</tr>
<tr>
<td>O/F d3</td>
<td>Install Unit</td>
<td>If inspection is successful the unit is installed based on its attributes</td>
</tr>
<tr>
<td>O/F d4</td>
<td>Evaluate Installation</td>
<td>The success of the installation is then determined.</td>
</tr>
<tr>
<td>O/F d5</td>
<td>Reassemble Unit Package</td>
<td>The remaining parts of the unit are reassembled into the unit package.</td>
</tr>
<tr>
<td>M/F d6</td>
<td>Store Unit Package</td>
<td>The reassembled package is stored for future use.</td>
</tr>
</tbody>
</table>

5.5.2 Functional Architecture

The architectures for the three devices and attached functional units are presented in this section.

5.5.2.1 F/U 1.1 – RF Unit

The functional architecture of this unit is shown in Figure 5-7, and definitions for the identified functional units and interfaces are given in Table 5-17 and Table 5-18 respectively.
Chapter 5 – Preliminary Design

Figure 5-7: Omnipomote Functional Architecture.

Table 5-17: Omnipomote Functional Units Definitions.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 1.1.1</td>
<td>RF Antenna</td>
<td>The antenna is responsible for transmitting or receiving electromagnetic waves.</td>
</tr>
<tr>
<td>F/U 1.1.2</td>
<td>Radio</td>
<td>This unit resembles a wireless transceiver module.</td>
</tr>
<tr>
<td>F/U 1.1.3</td>
<td>ECU</td>
<td>The electronic control unit is responsible controlling the actions of this module.</td>
</tr>
<tr>
<td>F/U 1.1.4</td>
<td>Visual Indicators</td>
<td>The means by which internal states are communicated to users.</td>
</tr>
<tr>
<td>F/U 1.1.5</td>
<td>Module Enclosure</td>
<td>Secure housing for the unit comes in the form of this enclosure.</td>
</tr>
<tr>
<td>F/U 1.1.6</td>
<td>PCB Board</td>
<td>The PCB board is the mechanical structure of communication tracks and components.</td>
</tr>
<tr>
<td>F/U 1.1.7</td>
<td>Configuration Connectors</td>
<td>The connectors through which the RF unit connects to either the mini-controller or the battery pack.</td>
</tr>
<tr>
<td>F/U 1.1.8</td>
<td>Sensors</td>
<td>The activation and temperature sensor of the unit.</td>
</tr>
</tbody>
</table>

Table 5-18: Omnipomote Interface Descriptions.

<table>
<thead>
<tr>
<th>Interface Number</th>
<th>Function</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF a1</td>
<td>Resembles the electrical currents between the radio and the antenna.</td>
<td>Analogue</td>
</tr>
<tr>
<td>IF a2</td>
<td>This is the communication interface between the ECU and the radio unit.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF a3</td>
<td>The interface through which the ECU changes the states of the indicators.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF a4</td>
<td>The means by which the enclosure and PCB board are combined.</td>
<td>Mechanical</td>
</tr>
<tr>
<td>IF a5</td>
<td>This interface is the physical contact between electronic components and the PCB board.</td>
<td>Mechanical</td>
</tr>
<tr>
<td>IF a6</td>
<td>The device receives power and data via this interface from other devices</td>
<td>Electrical / Electronic</td>
</tr>
<tr>
<td>IF a7</td>
<td>The activation signal and temperature received from the sensors.</td>
<td>Electronic</td>
</tr>
</tbody>
</table>

5.5.2.2 F/U 1.2 – Mini-Controller

Figure 5-8 shows the functional architecture for the mini-controller device. The definitions of the identified functional units and interfaces are shown in Table 5-19 and Table 5-20 respectively.
Chapter 5 – Preliminary Design

Development of a modular low power IEEE 802.15.4 wireless sensor module

Figure 5-8: OmnipoController Functional Architecture.

Table 5-19: OmnipoController Functional Units Definitions.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 1.2.1</td>
<td>Module Enclosure</td>
<td>This unit resembles the casing that protects the mini-controller.</td>
</tr>
<tr>
<td>F/U 1.2.2</td>
<td>PCB Board</td>
<td>The electronic components are mounted on this PCB board.</td>
</tr>
<tr>
<td>F/U 1.2.3</td>
<td>Power Supply</td>
<td>Onboard power regulator.</td>
</tr>
<tr>
<td>F/U 1.2.4</td>
<td>Operation Selector</td>
<td>For personnel to select between different operational objectives.</td>
</tr>
<tr>
<td>F/U 1.2.5</td>
<td>Power Connector</td>
<td>The connector by which power is connected to the unit.</td>
</tr>
<tr>
<td>F/U 1.2.6</td>
<td>USB Flash Connector</td>
<td>The connector used to attach a USB flash drive to the mini-controller.</td>
</tr>
<tr>
<td>F/U 1.2.7</td>
<td>Flash Drive Controller</td>
<td>Controls communication between the USB flash drive and onboard ECU.</td>
</tr>
<tr>
<td>F/U 1.2.8</td>
<td>Visual Indicators</td>
<td>Visual indicators for personnel to view internal states of the unit.</td>
</tr>
<tr>
<td>F/U 1.2.9</td>
<td>USB Controller</td>
<td>This controller is used by the ECU to switch the USB data signals between units.</td>
</tr>
<tr>
<td>F/U 1.2.10</td>
<td>ECU</td>
<td>The actions and operations of the module are determined by this unit.</td>
</tr>
<tr>
<td>F/U 1.2.11</td>
<td>Peripheral Connectors</td>
<td>Provisional pads to attach other peripherals to this the mini-controller.</td>
</tr>
<tr>
<td>F/U 1.2.12</td>
<td>USB Host Connector</td>
<td>Connector to attach the mini-controller with a USB host.</td>
</tr>
<tr>
<td>F/U 1.2.13</td>
<td>Host Controller</td>
<td>A unit to control communication between the mini-controller and host PC.</td>
</tr>
<tr>
<td>F/U 1.2.14</td>
<td>Configuration Connector</td>
<td>The RF unit is connected to the mini-controller via this unit.</td>
</tr>
<tr>
<td>F/U 27</td>
<td>USB Flash Drive</td>
<td>A standard USB flash drive memory stick.</td>
</tr>
</tbody>
</table>

Table 5-20: OmnipoController Interface Descriptions.

<table>
<thead>
<tr>
<th>Interface Number</th>
<th>Function</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF b1</td>
<td>The physical bond between the PCB board and unit enclosure.</td>
<td>Mechanical</td>
</tr>
<tr>
<td>IF b2</td>
<td>The means by which the electronic components are fastened to the PCB board.</td>
<td>Mechanical</td>
</tr>
<tr>
<td>IF b3</td>
<td>Power connections between onboard power supply and components.</td>
<td>Electrical</td>
</tr>
<tr>
<td>IF b4</td>
<td>Interface through which the ECU reads the mode inputs</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b5</td>
<td>Communications interface between the ECU and the USB flash drive controller.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b6</td>
<td>This interface resembles the means by which the ECU controls the state of the indicators.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b7</td>
<td>USB communication lines between the USB flash drive and switching device.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b8</td>
<td>USB communication lines between the switching device and controller</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b9</td>
<td>The ECU controls the switching action of the USB controller switch via this interface.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b10</td>
<td>ECU communication interface with other peripheral connectors.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b11</td>
<td>USB communication lines between the host and switching device.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b12</td>
<td>USB communication lines between the switching device and controller.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b13</td>
<td>Communication interface between the ECU and the USB host controller.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF b14</td>
<td>RF unit communication and power interface.</td>
<td>Electrical / Electronic</td>
</tr>
<tr>
<td>IF b15</td>
<td>Power connections between the power unit and the power cable</td>
<td>Electrical</td>
</tr>
</tbody>
</table>
5.5.2.3 F/U 1.3 – Battery Pack

In Figure 5-9 the functional architecture of battery pack is given. Table 5-21 and Table 5-22 describe all identified functional units and interfaces.

![Figure 5-9: OmnipoBatteryPack Functional Architecture.](image)

**Table 5-21: OmnipoBatteryPack Functional Units Definitions.**

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 1.3.1</td>
<td>Module Enclosure</td>
<td>The protective casing of the battery pack.</td>
</tr>
<tr>
<td>F/U 1.3.2</td>
<td>Pin Protection</td>
<td>The RF unit uses the same connectors to attach to the mini-controller and battery pack. Therefore, the unused pins (communication pins) needs to be protected by this unit.</td>
</tr>
<tr>
<td>F/U 1.3.3</td>
<td>Configuration Connectors</td>
<td>The connector by which the RF unit is connected to the battery pack.</td>
</tr>
<tr>
<td>F/U 1.3.4</td>
<td>PCB Board</td>
<td>The unit on which all electrical and electronic components are mounted.</td>
</tr>
<tr>
<td>F/U 1.3.5</td>
<td>Battery Holder</td>
<td>The battery holder holds the specified batteries.</td>
</tr>
</tbody>
</table>

**Table 5-22: OmnipoBatteryPack Interface Descriptions.**

<table>
<thead>
<tr>
<th>Interface Number</th>
<th>Function</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF c1</td>
<td>The bond between the battery pack and the enclosure is defined by this interface.</td>
<td>Mechanical</td>
</tr>
<tr>
<td>IF c2</td>
<td>The means by which the unused pins of the connector are held at a stable state.</td>
<td>Electronic</td>
</tr>
<tr>
<td>IF c3</td>
<td>The power transferred from the battery pack to the connectors is described by this interface.</td>
<td>Electrical</td>
</tr>
<tr>
<td>IF c4</td>
<td>The physical connection between all components and the PCB board.</td>
<td>Mechanical</td>
</tr>
</tbody>
</table>

5.5.2.4 F/U 2 – USB Cable

A functional architecture for the USB cable is given in Figure 5-10, and the identified functional units is defined in Table 5-23.

![Figure 5-10: USB Cable Functional Architecture.](image)
Table 5-23: USB Cable Functional Units Definitions.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 2.1</td>
<td>Connector Side 1</td>
<td>The connector type for one side of the cable.</td>
</tr>
<tr>
<td>F/U 2.2</td>
<td>Cable Type</td>
<td>The cable properties are described by this unit.</td>
</tr>
<tr>
<td>F/U 2.3</td>
<td>Connector Side 2</td>
<td>The connector type for the other side of the cable.</td>
</tr>
</tbody>
</table>

5.5.2.5 F/U 12 – Power Cable

Two possible methods can be used to connect power to the mini-controller. The one requires a large power supply on the mini-controller while the other has power supply in line with the cable. Functional architectures for these two solutions are given in Figure 5-11 and are defined in Table 5-24.

Figure 5-11: Power Cable Functional Architecture.

Table 5-24: Power Cable Functional Units Definitions.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 12.1</td>
<td>Connector Side 1</td>
<td>The connector type for one side of the cable.</td>
</tr>
<tr>
<td>F/U 12.2</td>
<td>Cable Type</td>
<td>The properties of the cable are described by this unit.</td>
</tr>
<tr>
<td>F/U 12.3</td>
<td>Connector Side 2</td>
<td>The connector type for the other side of the cable.</td>
</tr>
<tr>
<td>F/U 12.4</td>
<td>Transformer and rectifier</td>
<td>Convert mains supply to the input required by the mini-controller.</td>
</tr>
</tbody>
</table>

5.5.2.6 F/U 13 – Programming Apparatus

Figure 5-12 and Table 5-25 shows the architecture and unit descriptions of the programming apparatus.

Figure 5-12: Programming Apparatus Functional Architecture.
### Chapter 5 – Preliminary Design

#### Table 5-25: Programming Apparatus Functional Units Definitions.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 13.1</td>
<td>Connector Side 1</td>
<td>The connector type for one side of the apparatus.</td>
</tr>
<tr>
<td>F/U 13.2</td>
<td>Programming apparatus</td>
<td>The specific programming used.</td>
</tr>
<tr>
<td>F/U 13.3</td>
<td>Connector Side 2</td>
<td>The connector type for the other side of the apparatus.</td>
</tr>
</tbody>
</table>

#### 5.6 Resource and Requirements Allocation

In this phase, a resource and requirements allocation is completed for the identified operational functions and functional units. The resource allocation tables are given in Table 5-26 and Table 5-27.

#### Table 5-26: Operational Functions Requirements Allocation - Preliminary Design.

<table>
<thead>
<tr>
<th>Operational Function</th>
<th>Allocated Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>O/F a1 to O/F a7</td>
<td>PR-SO 1, PR-DS 3, PR-DS 4, PR-DS 5, PR-DS 6, PR-DS 7, PR-DS 8, PR-DS 9, PR-DS 10</td>
</tr>
<tr>
<td>O/F b1 to O/F b3</td>
<td>PR-SO 2, PR-DS 3, PR-DS 4, PR-DS 5, PR-DS 6, PR-DS 7, PR-DS 8, PR-DS 9, PR-DS 10</td>
</tr>
<tr>
<td>O/F c1 to O/F c7</td>
<td>PR-SO 3, PR-DS 2, PR-DS 3, PR-DS 4, PR-DS 5, PR-DS 6, PR-DS 7, PR-DS 8, PR-DS 9, PR-DS 10</td>
</tr>
<tr>
<td>O/F d1 to O/F d7</td>
<td>PR-SO 4, PR-DS 3, PR-DS 4, PR-DS 5, PR-DS 6, PR-DS 7, PR-DS 8, PR-DS 9, PR-DS 10</td>
</tr>
</tbody>
</table>

#### Table 5-27: Functional Units Requirements Allocation - Preliminary Design.

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Allocated Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 1.1</td>
<td>PR-RF 1, PR-RF 2, PR-RF 3, PR-RF 4, PR-RF 5, PR-RF 6, PR-RF 7, PR-RF 8, PR-RF 9, PR-RF 10, PR-RF 11, PR-RF 12, PR-DS 1, PR-DS 2, PR-DS 3, PR-DS 4, PR-DS 5, PR-DS 6, PR-DS 7, PR-DS 8, PR-DS 9, PR-DS 10</td>
</tr>
<tr>
<td>F/U 1.1.1</td>
<td>PR-RF 2, PR-RF 11, PR-RF 12</td>
</tr>
<tr>
<td>F/U 1.1.2</td>
<td>PR-RF 1, PR-RF 4</td>
</tr>
<tr>
<td>F/U 1.1.3</td>
<td>PR-RF 4, PR-RF 5, PR-RF 6, PR-RF 7, PR-RF 10, PR-RF 11</td>
</tr>
<tr>
<td>F/U 1.1.4</td>
<td>PR-DS 4, PR-DS 8</td>
</tr>
<tr>
<td>F/U 1.1.5</td>
<td>Only provisions for development scope.</td>
</tr>
<tr>
<td>F/U 1.1.6</td>
<td>PR-RF 3</td>
</tr>
<tr>
<td>F/U 1.1.7</td>
<td>PR-RF 7, PR-RF 8, PR-RF 9, PR-RF 11</td>
</tr>
<tr>
<td>F/U 1.1.8</td>
<td>PR-RF 11, PR-RF 12</td>
</tr>
<tr>
<td>F/U 1.2.1</td>
<td>Only provisions for development scope.</td>
</tr>
<tr>
<td>F/U 1.2.2</td>
<td>No limitations – freedom for development.</td>
</tr>
<tr>
<td>F/U 1.2.3</td>
<td>PR-MC 6, PR-MC 10, PR-MC 11</td>
</tr>
<tr>
<td>F/U 1.2.4</td>
<td>PR-MC 12</td>
</tr>
<tr>
<td>F/U 1.2.5</td>
<td>No limitations – freedom to choose.</td>
</tr>
<tr>
<td>F/U 1.2.6</td>
<td>PR-MC 1, PR-MC 8</td>
</tr>
<tr>
<td>F/U 1.2.7</td>
<td>PR-MC 1, PR-MC 8</td>
</tr>
<tr>
<td>F/U 1.2.8</td>
<td>PR-DS 4, PR-DS 8</td>
</tr>
<tr>
<td>F/U 1.2.9</td>
<td>PR-MC 3, PR-MC 4</td>
</tr>
<tr>
<td>F/U 1.2.10</td>
<td>PR-MC 1, PR-MC 2, PR-MC 3, PR-MC 4, PR-MC 5, PR-MC 8, PR-MC 9, PR-MC 7, PR-MC 11, PR-MC 12, PR-DS 1, PR-DS 2, PR-DS 3, PR-DS 4, PR-DS 5, PR-DS 6, PR-DS 7, PR-DS 8, PR-DS 9, PR-DS 10</td>
</tr>
<tr>
<td>F/U 1.2.11</td>
<td>PR-MC 7</td>
</tr>
<tr>
<td>F/U 1.2.12</td>
<td>PR-MC 2, PR-MC 10</td>
</tr>
<tr>
<td>F/U 1.2.13</td>
<td>PR-MC 2</td>
</tr>
<tr>
<td>F/U 1.2.14</td>
<td>PR-MC 5, PR-MC 6, PR-MC 9</td>
</tr>
<tr>
<td>F/U 1.3</td>
<td>PR-BP 1, PR-BP 2, PR-BP 3, PR-DS 1, PR-DS 2, PR-DS 3, PR-DS 4, PR-DS 5, PR-DS 6, PR-DS 8, PR-DS 9, PR-DS 10</td>
</tr>
<tr>
<td>F/U 1.3.1</td>
<td>Only provisions for development scope.</td>
</tr>
<tr>
<td>F/U 1.3.2</td>
<td>PR-BP 3</td>
</tr>
<tr>
<td>F/U 1.3.3</td>
<td>PR-BP 1</td>
</tr>
<tr>
<td>F/U 1.3.4</td>
<td>Limited to size of batteries</td>
</tr>
</tbody>
</table>

Development of a modular low power IEEE 802.15.4 wireless sensor module
Table 5.27 continuous

<table>
<thead>
<tr>
<th>Functional Unit</th>
<th>Allocated Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>F/U 1.3.5</td>
<td>PR-BP 2</td>
</tr>
<tr>
<td>F/U 2</td>
<td>PR-UC 1, PR-UC 2, PR-UC 3</td>
</tr>
<tr>
<td>F/U 2.1</td>
<td>PR-UC 2</td>
</tr>
<tr>
<td>F/U 2.2</td>
<td>PR-UC 1</td>
</tr>
<tr>
<td>F/U 2.3</td>
<td>PR-UC 3</td>
</tr>
<tr>
<td>F/U 3</td>
<td>PR-UH 1, PR-UH-2, PR-UH 3</td>
</tr>
<tr>
<td>F/U 6</td>
<td>PR-UH 3</td>
</tr>
<tr>
<td>F/U 12</td>
<td>PR-PC 1, PR-PC 2, PR-PC 3, PR-PC 4, PR-PC 5</td>
</tr>
<tr>
<td>F/U 12.1</td>
<td>PR-PC 4</td>
</tr>
<tr>
<td>F/U 12.2</td>
<td>PR-PC 1</td>
</tr>
<tr>
<td>F/U 12.3</td>
<td>PR-PC 5</td>
</tr>
<tr>
<td>F/U 12.4</td>
<td>PR-PC 2, PR-PC 3</td>
</tr>
<tr>
<td>F/U 13</td>
<td>PR-PA 1, PR-PA 2</td>
</tr>
<tr>
<td>F/U 13.1</td>
<td>Not part of design</td>
</tr>
<tr>
<td>F/U 13.2</td>
<td>PR-PA 1</td>
</tr>
<tr>
<td>F/U 13.3</td>
<td>PR-PA 2</td>
</tr>
<tr>
<td>F/U 15</td>
<td>PR-PCS 1</td>
</tr>
<tr>
<td>F/U 27</td>
<td>PR-MC 1, PR-MC 8</td>
</tr>
</tbody>
</table>

5.7 Trade-Off Analysis

Here, possible solutions for units are selected and presented. This selection is based on criteria that consist of a combination of requirements, cost, size, attributes, complexity and quantity required.

Note: It is impossible to compare different solutions with each other using trade-off tables because no effectiveness or performance criteria were received. If these factors were known the solutions might change, but great care and consideration was taken to select the solutions given.

PCB descriptions and enclosure designs are completed in Sections 5.7.4 and 5.7.5 respectively and the connectors (including pin protection on the battery pack) used to connect the three devices together are analyzed in Section 5.7.6.

5.7.1 F/U 1.1 - RF Unit

In this section, an analysis is completed on the functional units that make up the RF unit.

5.7.1.1 F/U 1.1.1 - RF Antenna

Specifications from the client require that the antenna be a ceramic chip antenna because of their low cost, small size and between moderate and good performance.
We chose the Fractus® Reach Xtend™ Chip Antenna for this purpose. It is specifically designed for high performance and low cost applications that operate at 2.4 GHz. This chip antenna was chosen because of its structure and polarization spread. [25]

It has four connection pads on its square structure, making it strong and durable when soldered on to a PCB board. It has a planar monopole antenna ideal for indoor and scattered environments. The antenna spread is unidirectional, and due to this uniform radiation pattern, the antenna will optimize the usage of the OmnipoMote. [25]

5.7.1.2 F/U 1.1.2 – Radio and F/U 1.1.3 - ECU

Requirements specify that these units be firmware-compatible with the TelosB mote. The easiest way to accomplish this, is to adopt the same hardware architecture than the TelosB mote. It uses the Chipcon SmartRF® CC2420 radio transceiver chip and Texas Instruments (TI) MSP430F1611 microprocessor. [13]

A true single-chip solution is provided by the CC2420. It is a 2.4GHz IEEE 802.15.4 compliant RF transceiver designed for low power and low cost applications. It is used for communications in the 2.4GHz unlicensed ISM band and complies with the following standards: ETSI EN 300 328 and EN 300 440 class 2 (Europe), FCC CFR47 Part 15 (US) and ARIB STD-T66 (Japan). [24]

5.7.1.3 F/U 1.1.4 - Visual Indicators

Visual confirmation and indication is the output function of this unit. It helps employees to see internal states of the RF unit during inspection, installation, diagnostics and maintenance.

To keep current consumption to a minimum, costs low and size small, it was decided to implement three visual LED indicators: The first indicator is a power indicator and the other two are used to indicate the following:

- RF –and mini-controller communication during normal operation.
- Results from diagnostic routines during installation, inspection and maintenance.

All three indicators are controlled by the microcontroller. Usually a power indicator would be placed directly between the power terminals of a device for constant indication, but because battery life is important we chose to control it from the ECU. Now this indicator can be shut down to save power.
5.7.1.4 F/U 1.1.8 – Sensors

Two sensing capabilities are required for this unit – an activation sensor and a temperature sensor. The ECU of the RF unit can be put into sleep mode and awaken by an external interrupt event. This external interrupt will be triggered by a tilt switch. An illustration is given in Figure 5-13.

![Symmetry](image.png)

**Figure 5-13:** Basic Operation of a Tilt Sensor. [26]

A conducting piece of material makes contact between two points to close the circuit when the tilt sensor is at a certain angle. If the device vibrates the material will move around and close the circuit to trigger the external interrupt of the electronic control unit. The chosen tilt sensor is the RBS 040200 unit from Active Switch and Sensor Limited. These sensors are designed for low current IC triggers. [26]

The temperature sensor is straightforward and we chose the TC1047 temperature sensor from Microchip. It is small in size, very low in current, with high accuracy and temperature range. It also has a linear analogue output of the temperature. [27]

5.7.2 F/U 1.2 - Mini-Controller

The functional units that make up the mini-controller are analyzed in this section.

5.7.2.1 F/U 1.2.3 - Power Supply

The functional objective of this unit is enforced by the components that make up the RF unit and mini-controller. Components on these devices have certain power requirements and it is the objective of this unit to deliver on these requirements.

Detail design feedback indicates that the power supply must be capable of delivering both 3.3V and 5V to the mini-controller device and 3.3V to the RF unit. A large transformer circuit on the mini-controller to regulate the mains supply directly is not a viable option because it is an expensive solution that will take lot of development time to get right and it requires large PCB space. Fixed dc voltage regulators are the
most viable due to cost and size. This means that a power cable with an onboard transformer and rectifier circuit is used.

5.7.2.2 F/U 1.2.4 – Operation Selector

2-Way headers and jumpers, or dip switches is viable solutions for this unit, but we chose the dip switches for the following reasons:

- It is a single component that makes fabrication easier (it however is not the cheapest).
- It is easier to work with by employees as the jumpers take a longer time to change and they can also be misplaced easily.
- Dip switches are usually numbered, which can be used to indicate the current mode of the mini-controller, during configuration, to employees.

5.7.2.3 F/U 1.2.5 – Power Connector

The connector on the power cable enforces requirements for the socket to use. For instance if a laptop cable were used, then a DC barrel would be the best solution for this unit.

At the point of making a decision for this unit, it was unclear what cable the client would use. It is also a small and not very important part of the device that can easily be changed if required. Therefore we chose 2-way Molex connectors for this development scope.

5.7.2.4 F/U 1.2.6 – USB Flash Connector

Functionally this unit must enable users to connect USB flash drives to the module. The mini-connector should use a female type-A socket to connect USB flash drives to it, since the flash drives uses male type-A connectors.

5.7.2.5 F/U 1.2.7 – Flash Drive Controller

Low cost USB flash memory is everywhere available today, but has their focus on the PC market. Implementation, cost and power consumption are the main issues when trying to use these peripherals in the embedded microcontroller environment. [28]
Microcontrollers usually lack in interfaces, resources and performance to be a USB host, and the software required to interface slave devices is difficult. The VNC1L Vinculum controller IC provides a solution for this. It is has a USB2.0 full-speed port that can act as the interface between the ECU of the mini-controller and the USB flash drive. Vinculum specifically targets the USB embedded market and the VNC1L is the only device that we found (more may exist) that will satisfy our needs. [28]

5.7.2.6 F/U 1.2.8 – Visual Indicators

This unit is functionally the same as the visual indicators of the RF unit, but differ in quantity. The following indicators are used during normal system operation:

- One LED to indicate that the device has power.
- One LED indicator used during USB host communications.
- One LED indicator for use during USB flash drive communications.
- An indicator LED that is used during communications with the RF unit.
- Two more LED indicators for other activities that the microcontroller may need to perform.

The last two indicators are the result from reviews and feedback. It was found during implementation that more indicators were necessary for other tasks that the ECU may need to perform, like USB switching and controlling peripherals.

All indicators are controlled by the ECU of the mini-controller except for the power indicator. It is possible to place the power indicator between the power terminals because the device is powered externally.

5.7.2.7 F/U 1.2.9 – USB Controller

The purpose of this unit is to alter the direction of USB communication lines. Figure 5-14 and Figure 5-15 shows the four possible scenarios that can happen for FFD device operations:

- Scenario a (PAN Coordinator): The USB host communicates with other nodes in the network via the RF capabilities of the OmnipoMote.
- Scenario b (PAN Coordinator): The USB host communicates with a USB flash drive via the ECU of the mini-controller.
- Scenario c (Coordinator): Data received from the RF unit is stored on the USB flash drive, and results from data interpretations are sent back via RF.
- Scenario d (Coordinator): Commands are received from operators via RF communication. The commands enable the ECU to change the direction flow of the USB communication lines.
Texas instruments have a 1:2 multiplexer / demultiplexer USB switch called the TS3USB221. It is a high speed USB 2.0 switch designed for USB switching in any form of consumer devices like cell phones, notebooks and so on. The conceptual layout using two TS3USB221 devices is shown in Figure 5-16. [29]
5.7.2.8 F/U 1.2.10 - ECU

All functional operations of the mini-controller are accomplished with this unit. The ECU is not required to have onboard USB interfaces as F/Us 1.2.7 and 1.2.12 provides these functions. However it is required that the unit have two UART modules, two serial peripheral interface (SPI) communication ports and at least 30 interface pins.

We chose the PIC18LF6722 family microprocessors from Microchip to perform the functions of this unit, because it comes from a manufacturer whose products we have experience in working with. The number of I/O pins also gives great flexibility to the design. It has large program and data memory, which is great because this is an ongoing development.

5.7.2.9 F/U 1.2.11 – Peripheral Connectors

Our client requests connections for future portable devices in the form of open pads on the mini-controller, but the pads themselves must be suitable for a connector. Therefore we chose to design the pads for general single row, 2mm pitched connectors.

It was decided to use pins on the ECU that can also be used for SPI or I2c communication. This will allow multiple devices to be connected to the same connector in the future.

5.7.2.10 F/U 1.2.12 – USB Host Connector

The mini-controller has a female type-A socket for the USB flash drive. Based on the requirements enforced on the design, the sockets for the host and flash drive cannot be the same. Two possible solutions exist for this unit in the form of a female type-B socket or a mini USB socket. The client was asked about which one he would prefer and he chose the female type-B socket, because there is no size limitation on the mini-controller PCB board and this socket is more commonly used.

5.7.2.11 F/U 1.2.13 – Host Controller

The ECU of the mini-controller must communicate with a USB host. Numerous microprocessors exist on the market with onboard USB slave capabilities, but PC drivers need to be written to interface them with the host except when it is a human interface device (HID).
HID devices do not require custom drivers as they use the HID driver class on all Microsoft, Linux and Macintosh operating systems. To save development costs, manufacturing costs, development time and ECU processing, we decided to use the USB232™ device from Firmware Factory. [30]

The USB232™ is a driver free (HID compliant) USB to asynchronous serial transceiver. The device is firmware made for two specific microchip microprocessors: PIC18F14K50 and PIC18LF2450. It establishes USB connections for embedded controllers and can be used with any of the three operating systems. They come with computer software that can be used to configure the unit and to communicate with it during testing. [30]

5.7.3 F/U 1.3 – Battery Pack

An analyses is only required for the battery holder. The client requires that the battery pack uses AAA rechargeable batteries. Therefore, an AAA battery holder will be used that also protects the batteries from being inserted in the wrong way.

5.7.4 PCB Boards

All PCB boards should be designed to the criteria and guidance presented in Section 5.3.3.10.2.

5.7.5 Enclosures

The development scope does not require that enclosures be designed, but the following points can still be made on this subject:

- Enclosures are subjected to the final dimensions of the three devices.
- Need to be water resistant because of the application area.
- They can be designed or purchased.
- Current designs should include means to connect the devices to their enclosure.

It would be best to design enclosures for these devices due to the custom size and quantity of the units involved. This will make a perfect fit between the enclosure and the device possible.
5.7.6 Connectors

Connectors to connect the different devices together must be robust, strong, durable, easily configurable and cost effective. It is used for the following connections:

- Power connections between the RF unit and battery pack.
- Power connections between the RF unit and mini-controller device.
- Data connections between the RF unit and mini-controller device.

The cheapest solution would be to use the same power terminals for all possible connections, to keep the costs down and dimensions low. The power terminals on the mini-controller are on the same connector as the data connection terminals. This means that the battery pack will use the same connector as the mini-controller. Because of this, unused pins on the battery pack needs to be stabilized in using pull-up or pull-down resistors.

Communication between the RF unit and mini-controller will be accomplished through SPI, which requires three communication lines. Provisions are also made for three more communication lines to use for control purposes. Although this is not required, it helps to have an extra method of communication, to externally trigger actions of the RF unit (external wake-up, cannot send data, able to sent data, and so on). This means that eight connection lines are required.

Numerous connectors exist that can be used for this unit, but they are very expensive. Good performance can be achieved through the use of header pins and sockets which are very inexpensive. They can be very effective if used correctly like in a double row configuration for instance.

5.7.7 Other

Presented in this section are the solutions for the boundary units of the three devices. The boundary units include the USB cable, software on the USB host, power cable, programming apparatus, software used to program components and the USB flash drive.

5.7.7.1 F/U 2 - USB Cable

A male type-A connector must be used on this cable to connect it to the USB host. Based on the analysis given in Section 5.7.2.11 (USB host connector) the other side of the cable must use a male type-B connector.
5.7.7.2 F/U 12 - Power Cable

Results from Section 5.7.2.1 (power supply) require the use of a power cable that includes the transformer and regulator circuitry.

*Note. It is still unclear what specific type of power cable the client will use.*

5.7.7.3 F/U 13 - Programming Apparatus

Closer study reveals that the following components require programming: F/U 1.1.3 (MSP430F1611), F/U 1.2.7 (VNC1L), F/U 1.2.10 (PIC18LF6722) and F/U 1.2.13 (USB232™).

The MSP430GANG programmer from Texas Instruments was chosen to program the MSP430F1611 microprocessor. This is a production programmer capable of programming up to eight devices at the same time. It was chosen because it would aid in future mass production of the RF unit, which also makes a positive impact on time and costs. [31]

The VNC1L must be programmed once via RS232 connections for which the serial port of a PC can be used. On the other hand, the USB232™ device comes with its firmware on it, but can be configured directly via the USB communication lines. [28] [32]

A microchip Pickit® 3 programmer was already in our possession before we knew about this project, and is more than adequate for programming the PIC18LF6722. We found no reason to buy a new programmer for this purpose.

5.7.7.4 F/U 24 - USB Host Software

No software development is required for this development scope. The USB232™ configuration software, provided by Firmware Factory, is used in testing the USB host communication.

5.7.7.5 F/U 26 - Programming PC Software

The IAR embedded workbench of Texas Instruments is used for MSP430F1611 firmware development and is a free downloadable IDE for development. This also saves manufacturing costs.

Vinculum provides a terminal application to program the VNC1L, and the USB232™ is configured via the application firmware provided by Firmware Factory.
Microchips’ MPLAB IDE together with the C18 compiler is used for developing the firmware for the PIC18LF6722. This IDE comes with a programmer and the compiler is specific to this series of microprocessors.

5.7.7.6 F/U 27 – USB Flash Drive

A standard USB flash drive can be used for data storage. It is recommended that the flash drive be no larger than 4 GB due to the limitations of the VNC1L device.

5.8 Design Synthesis

This section marks the bridge between the preliminary design and the detail design phases. A summary of each unit under development is given, aimed at making the detailed development easier.

Specific architectures for the three main devices under development (OmnipoMote, OmnipoController and OmnipotBatteryPack) is shown in Figure 5-17, Figure 5-18 and Figure 5-19 respectively.

Figure 5-17: Preliminary Design Synthesis - OmnipomoMote Specific Architecture.
Chapter 5 – Preliminary Design

Development of a modular low power IEEE 802.15.4 wireless sensor module

Chapter 5 – Preliminary Design

Figure 5-18: Preliminary Design Synthesis - OmnipoController Specific Architecture.

Figure 5-19: Preliminary Design Synthesis – OmnipoBatteryPack Specific Architecture.

For the boundary units only the USB and power cables require further architectures. The updated architectures are given in Figure 5-20 and Figure 5-21. Architectures of USB hosts, USB host software, programming PCs, programming PC software, programming apparatus and USB flash drives already exist, as they are developed by third parties. We can only allocate the solution to the functional blocks of each unit.
Figure 5-20: Preliminary Design Synthesis – USB Cable.

Figure 5-21: Preliminary Design Synthesis – Power Cable.

Figure 5-22 shows specific functional architectures for the four programming solutions. In the third and fourth instances, the apparatus is the cable connecting the programming PC to the device. A new unit is also introduced in F/U 28. The laptop we used for programming had no fixed serial port and this unit represents the USB-to-serial converter to emulate a serial port on the laptop.

The USB host software is the HIDConfig application provided by Firmware Factory and is shown in Figure 5-23. In Figure 5-24 specific architectures for the programming PC are given.
The operational analysis of the personnel presented in Section 5.5.1 also needs to be taken down to a detailed interaction level. These operational tasks stay the same and will not be illustrated again.

5.9 Design Review and Feedback

All design reviews and error corrections performed on the preliminary design led to numerous feedbacks into this and other design phases. The following items, in no particular order, were the most noticeable of all improvements during the review process:

- Ongoing and continuous study of technical documents, functional objectives, operational and government requirements and regulations, because of the following:
  - Change in client requirements.
  - Better understanding of technologies.
  - Results from the detail design phase.
  - Improvements of literature in the respected fields of study.

- Improvement and continuous adjustment of functional architectures to fit the requirements. The most noticeable architectural changes was on the following units:
  - On the mini-controller it was first undecided to include F/U 27 (the USB flash drive) in its architecture, but because it is the memory of the device it is internally part of the mini-controller, although it is an off the shelf and removable part.
Numerous functional units are the result of performing the detailed design. These are results such as power requirements, stability resistors, specific configurations required by the datasheets, and so on. In the first few iterations of the preliminary design, these requirements are unforeseeable.

- Continuous improvements in the decision making of finding the correct solutions for units. Solutions that required the most attention was the following:
  - If the power supply configuration should drawn power from the USB host or from the mains supply via the power cable. It was found that USB communication will not work as effectively if a lot of current is drawn on the same port.
  - The USB controller responsible for switching the USB communication lines.
  - If we should use a balun or passive circuitry for impedance matching between the radio module and antenna of the RF unit.
  - Connector layout between the different devices.
  - How to make each unit as user friendly as possible.
- Identifying which aspects are appropriate to this or other levels of design.

5.10 Conclusion

The preliminary design of development, as required by the process model, was performed in this chapter. During this design, focus moved from a system view, to a view where the operational outcomes of the three devices are known.

A detailed literature study and analysis was performed on the requirements of the client, technologies involved in the system, operational feasibility outcomes of the devices and government regulations. This study revealed detailed requirements for the second level functional analysis.

This functional analysis again consisted of an operational analysis and functional architecture. Resource allocation tables were also used to relate the requirements to the functional analysis. The operational analysis was used to define how personnel would use these units. These actions and requirements were used in the decisions made to select possible solutions.

During the trade-off and configuration analysis, solutions were developed for each functional unit that forms part of the development scope. These solutions were then incorporated into the developed functional analysis and put together into a second level design synthesis.

Design reviews and feedback was performed throughout this development phase to ensure that the development does not lose focus or deviate from the process model steps. As with the conceptual design, this design phase exceeded expectations regarding the SE development. We were able to successfully break up the conceptual architectures into more detailed architectures, that define each
unit of the system more detail, although we only gave attention to the units that have a direct influence on the design.

One drawback that may have an influence on the developed devices, is the lack of performance requirements by the client. These requirements could have been used during the trade-off analysis to select between configurations, components, design choices and so forth. A decision matrix could have been used for the trade-off analysis using the requirements as a measuring instrument. This said the developed architectures are functionally capable of satisfying the needs and expectations of the client.
Chapter 6

Detail Design

Detailed designs of the entities involved in the development scope are presented in this chapter. Units of the preliminary design synthesis are evaluated further, until solutions are found for them.

6.1 Introduction

Detail design phases are used to design and develop the form of units. In the conceptual design we concentrated how units fit into the system and during the preliminary design we described functionally how units are used in the system. In this chapter, the fit and function requirements are used to design and develop detailed functional analysis of the units. This functional analysis gives the exact solutions for the respected units.

First in this chapter, an analysis of the preliminary design synthesis is completed. This is aimed at establishing how entities are solved: purchase from markets, PCB design and so on. It is an indication of which entities require further attention or have further influence on the design. A detailed functional analysis commences once this is completed, and consists of an analysis on personnel operational tasks, detailed schematic architectures of functional units and lastly, PCB architectures.

A discussion on the two different development models is given. The first is the experimental model and the other the engineering model. Tests used to validate the devices for functional capability is also shown. The chapter concludes with a design review of the detailed design phase.
6.2 Preliminary Design Output Analysis

The design synthesis of the preliminary design, together with technical information on certain components in the form of datasheets and user manuals, serve as inputs for this design phase. Units destined for further analysis are identified based on the following criterion:

1. Only units that form part of the development scope is evaluated further. This is important during the detailed operational analysis.
2. “The wheel will not be reinvented”. If solutions exist at a consumer market, then that unit will be purchased and not designed or manufactured. This is done to save development time. These products are also already assured for effectiveness and performance, and in some cases approved for specific applications (like USB cables).
3. Laboratory equipment is used to test the devices.
4. The results from the trade-off analysis and resource allocation tables in the preliminary design are used to find detailed characteristics and requirements for units.

6.2.1 Personnel Operational Procedures

The development scope requires no development of unit packaging, enclosures and final application firmware. Some operational functions stated in the preliminary design, influence the design of these units and vice versa, but not the design and layout of the PCB boards.

All operational tasks that affect the design and layout of the PCB of each device, needs further detailed evaluation. Therefore, the following units are evaluated further in Section 6.3.1:

- Programming personnel: O/F a2, O/F a3 and O/F a4.
- Inspection personnel: O/F b2 and O/F b3.
- Maintenance personnel: M/F c2.
- Installation personnel: O/F d3 and O/F d4.

6.2.2 Three devices under development

All three devices are subjected to a detailed functional analysis, as required by the development scope. This includes schematic architectures, PCB designs and construction of the XDM and EDM prototypes.
6.2.3 Boundary Cables and Programming Apparatuses

Testing for functional capability is completed in a laboratory and therefore no power cable as described in the preliminary design is used. Laboratory power supplies and cables are used to deliver power to the devices. These power supplies can be adjusted to deliver the right amount of power.

USB cables used for connections between the mini-controller and USB hosts are purchased from consumer markets. Cables required for connections between the programming PC and programming apparatus, as well as the programming apparatus and mini-controller, come with the purchase of the programming apparatus.

6.2.4 Programming Software

All the required programming software comes with the purchase of the programming apparatuses, or can be downloaded from the websites of the respected manufacturers.

6.3 Detailed Functional Analysis

Detailed functional analyses on operational tasks and functional units are performed in this section.

6.3.1 Operational Analysis

Results obtained from performing the operational analysis is used to design products that aid personnel in their everyday tasks, and to do these tasks in accordance with the feasibility criteria of the system. Of significant importance are the reliability, maintainability, usability and testability criterions.

The conceptual design enabled us to identify all personnel involved in the system, and in the preliminary design we identified the first level operational tasks that they must perform. It is now the function of this design phase to depict how each task is accomplished. These tasks are the ideal actions that personnel will perform when interacting with the devices. Results obtained from analysing these tasks are used in the detail design of system units to ensure that the operational feasibility is met, when personnel interact with them.

Note: For the development scope, operational tasks that influence the design and layout of the three devices are required. Other tasks require preliminary design information on the enclosures, packaging and final firmware of the three devices.
A detailed analysis on how personnel interact with the devices is completed to ensure that the feasibility criterion is met in the following sense:

1. Design user-friendly devices with regard to connectors, indicators, buttons, and so forth.
2. To design PCB boards that are easily configurable with other PCB boards (RF unit with mini-controller for example), and yet be strong and durable.
3. To put in place markings on the devices for personnel to easily recognise a device.

6.3.1.1 F/U 16 – Programming Personnel

This functional unit represents the personnel responsible for upgrading devices with new firmware. The purpose of this firmware (testing or operational application) is irrelevant to their actions. Therefore, the same operational tasks account for the developing engineers that need to write firmware to test the device’s functional capability.

Based on the preliminary analysis, the first task of these personnel is to retrieve a package containing a device and its casing (O/F a1) and to inspect it (M/F a2). Inspections are completed to determine if the package, casing and device is present and in good order. After a successful inspection, it will be possible to install the device for programming (O/F a3) and then to program it with the new firmware (O/F a4). The success of upgrading a device with new firmware is assured (M/F a5) where after the device and casing are reassembled into the unit package (O/F a6), and stored again (O/F a7).

M/F a2 – Inspect Device Package

We are able to analyse this operational task, assuming that the device package and casing exists. This operational task has the primary function of determining if the device and casing are present in the storing package and in good working order. Figure 6-1 shows a detailed analysis for this operational function. Each reference to a unit represents both the device and casing.

![Figure 6-1: Detailed Operational Analysis – M/F a2.](image-url)
It will become clear as this detailed operational analysis is completed that these operational tasks are shared between the different personnel. Devices should be marked clearly with their name, hardware version number and software version number. The hardware name and hardware version number can be printed on the PCB board from the beginning, but not the firmware version number. This last version can be indicated by putting stickers on the final device, after programming.

This makes it possible for personnel to easily identify units when retrieving them from the package (O/F a2.2). It will ensure effectiveness in the feasibility criterion of reliability and usability. The system is made more reliable since personnel cannot get confused with which device they are working with.

O/F a2.3 is discussed later in this section under inspection personnel, as it is also a shared operational task. M/F a2.1 required more information on the package and O/F a2.4 cannot be analyzed as the criteria to a successful inspection is not known (not part of development scope).

O/F a3 – Install Unit for Programming

There are three different functional units that require programming with new or updated firmware and the operational tasks that personnel must go about doing this, differs from unit to unit.

The differences exist, because the programming apparatus of each component differs. To program the MSP430F1611 microprocessor on the RF unit a USB-serial converter (if the programming PC does not have a serial port), the MSP430-Gang programming adaptor, a special adaptor cable and a separate power connection is required.

On the mini-controller, the PIC18F6722 microprocessor only requires the Pickit 3 programming adaptor and USB cable that comes with the purchase of the adaptor. It is only required to program the VNC1L unit once via the PC serial port or USB-serial converter, for those PCs without one. This device would usually be programmed before it is mounted on the PCB board during mass-production, but this would be too expensive for evaluation purposes. Thus, the mini-controller needs to be configured by personnel to program either the PIC18F6722 or VNC1L.

The programming adaptors are purchased from the manufacturers of the microprocessors and they already contain markings and other features that will ensure that the system feasibility criterion is met. The only influence that this operational task has on the design is the programming connections between the device and adaptor. It must be easily accessible and configurable. Procedures for installing these to program the devices are given in Figure 6-2.

*Note. To configure the mini-controller for the different programming roles is fully explained in Section 6.3.2.2 under the functional architecture of the PIC18f6722 and VNC1L units.*
Chapter 6 – Detail Design

Development of a modular low power IEEE 802.15.4 wireless sensor module

Figure 6-2: Detailed Operational Analysis – O/F a3.

O/F a4 – Program Unit

The programming software from the respected component manufacturers is used for this purpose and therefore the means by which they affect the feasibility criterion cannot be assured. However, well-known and proven IDEs were chosen from proven manufacturers who have quality assurance procedures for reliability and user-friendliness. No operational procedures can be given for this unit, because the user documentation on these IDEs can be consulted for this.

6.3.1.2 F/U 18 – Inspection Personnel

Regular inspection of installed devices is a mandatory objective of any business, as it aids in the reliability and maintainability aspects of any system. A well-designed interface through which inspection is completed ensures a user-friendly environment. Inspections on devices can identify problems before they can occur (preventive maintenance), to maintain a high operating life cycle for the whole system.

The preliminary analysis for these personnel defined three specific operational tasks. Personnel must first travel to and locate an installed unit (M/F b1), identify it by using means set out by the device usability features (M/F b2) and inspect it for any errors, defects or failures (O/F b3). The last two operational procedures require deeper understanding.

M/F b2 – Identify Installed Unit

The operational procedures that personnel must complete to identify an installed unit is illustrated in Figure 6-3.
It is clear that these operational tasks will reveal the same characteristics for reliability and usability as programming personnel that remove devices from the package. They also must locate and use the marking of the device to accomplish this.

O/F b3 – Inspect Unit

The RF unit and mini-controller devices will be able to diagnose themselves for problems with automated routines for testing interfaces and components. These diagnostic routines can run throughout the life cycle of the device or upon request from personnel. The first is the most appropriate for the following reasons:

- The latter would require extra interface components through which users can trigger it to run.
- New interface components make development costs, and manufacturing costs higher.
- The first will eliminate human interactions and therefore human errors.

Detailed procedures on how personnel should go about completing this task are given in Figure 6-4. The procedure clearly illustrates that the actions of these personnel affect the usability of the device with respect to their visual indicators. All visual indicators should be clearly marked to indicate their purpose, to avoid confusion. The extent of the accessibility and the clarity thereof affects the reliability, maintainability and testability criteria of operational feasibility.

Clear and easily accessible indicators make system inspections easier. Results obtained from device testing are also shown through these indicators. If these indicators are interpreted wrongly or if they are not easily accessible, the reliability and system availability is influenced negatively.
6.3.1.3 F/U 19 – Maintenance Personnel

It is the responsibility of inspection personnel to identify faulty devices, but it is the responsibility of maintenance personnel to correct them. The preliminary analysis of these personnel revealed several operational tasks that must be completed.

Personnel start by gathering a package, marked for repair by inspection personnel, from the depot-and-store facility (M/F c1). This package contains a faulty device, which needs to be identifiable by the personnel (M/F c2). The faulty device then needs to be analyzed to determine exactly what is wrong with it (O/F c3), and the level of maintenance required to fix it is calculated (O/F c4). Once the level of maintenance is known, can the personnel perform maintenance operations on it (M/F c5), reassemble the repaired device into its package (M/F c6), and stored for future use (M/F c7).

The input analyses revealed that only M/F c2 requires further analyses to reveal the detail characteristics thereof. This operational task will however reveal the same characteristics as M/F b2 and M/F a2.

6.3.1.4 F/U 20 – Installation Personnel

Personnel responsible for installing new devices need to ensure installations of a high reliability. Not only will the results from the installation affect the installed unit, but also the reliability and effectiveness of the system.

Top-level operational procedures were defined for installation personnel in the preliminary design. The first task based on this analysis is to retrieve a packaged containing the device that needs to be installed (M/F d1), where after the package is inspected (M/F d2). Upon a successful inspection, the device is removed from the package and installed into the system (O/F d3). The installation is verified by viewing the outputs from the diagnostic routines on the visual indicators of the device (O/F d4), and all items not installed reassembled in the package (O/F d5). This reassembled package is then stored for future use (M/F d5).

O/F d3 – Install Unit

Installing the actual device into the system requires interaction from personnel. They have the responsibility of setting the devices for their respected operational outcomes. Figure 6-5 shows the actions that these personnel must complete to install a unit.
The operation selector (F/U 1.2.4) is used for selecting the possible output configurations on the mini-controller. No interaction is required on the RF unit, as firmware will automatically detect if it is connected to the mini-controller or battery pack.

**O/F d4 – Evaluate Installation**

Automatic start-up procedures will diagnose the installation and output its findings on the visual indicators of the device, once a device is installed correctly. These indicators need to be observed by personnel to determine if installation was successful. It also serves as an indication if the device was able to configure itself into the system correctly via the RF communication.

### 6.3.2 Schematic Functional Architecture

In this section, schematic architectures are developed for each of the three electronic devices. To develop these architectures, the following information is used:

- Results from the trade-off analyses completed in the previous design phase.
- Functional operations derived from the preliminary design requirements analysis.
- Technical information provided by the datasheets of the components used.

#### 6.3.2.1 F/U 1 – RF Unit

Figure 6-6 shows the schematic for F/U 1.1.1. In this schematic, the antenna is represented by \( U1 \) and \( U2 \) represents the balun, which is used for impedance matching between the antenna and the ECU of the RF unit. The capacitor \( C1 \) and inductor \( I1 \) forms a matching circuit across the balanced port and a DC block for biasing with the differential ports, respectively. Refer to [25], [33], [34] and [35] for more technical information and requirements on these two units and their configuration.
Chapter 6 – Detail Design

The antenna and balun configuration connects directly with the radio transceiver module given in Figure 6-7. In this schematic, U3 represents the CC2420 IEEE 802.15.4 compliant wireless radio transceiver.

This transceiver module has a few external components. R1 stabilises the active-low reset of the radio module, R2 is a precision current reference for the module and R3 stabilises the active-high voltage reference enable pin. The crystal and its load capacitors are represented by X1, C2 and C3 respectively. C6, I2 and C7 form a DC filtering and decoupling circuit for the power input. All other capacitors are used for DC filtering and decoupling functions for the output of the internal voltage regulator and power pins of the device. R4 is only a coupling resistor that can be used to split the analogue supply from the digital supply. Please refer to [24] for more information.

Figure 6-6: Detailed Functional Architecture – F/U 1.1.1.

Figure 6-7: Detailed Functional Architecture – F/U 1.1.2.
The transceiver module is controlled by the microprocessor of the RF unit and its detailed schematic is given in Figure 6-8. The microprocessor (MSP430F1611) is represented by $U_4$. $U_5$ is the connector through which the microprocessor is programmed.

The active low reset pin of the microprocessor is held high through $R_5$. The button, represented by $B_1$, enables the user to reset the microprocessor as it pulls the reset pin low when pressed. $C_{13}$, $I_3$, $C_{14}$, $C_{15}$ and $C_{16}$ have the function of filtering and decoupling the DC power delivered to the device. $X_2$, $C_{13}$ and $C_{18}$ represents the crystal of the microprocessor and its loading capacitors. More information on the MSP430F1611 microprocessor is available in [36] and for programming the device, please refer to [31].

The visual indicators, through which internal states of the RF unit are shown to personnel or other users, are shown in Figure 6-9. $L_1$, $L_2$ and $L_3$ shows the required LED indicators and the current that flows through these indicators is limited by $R_6$, $R_7$ and $R_8$. [37][38]

The indicators are also configured as open source pins on the microcontroller. This means that the current is drawn directly from the power supply and not delivered by the microcontroller. [37]
In Figure 6-10, \( U7 \) represents the connector used to interface the RF unit with either the mini-controller or the battery pack. Power is received through pin 8 on the connector, but is returned through pin 6 to tell the mini-controller the RF unit is attached to it. Pins 2, 3 and 4 are communication pins configured for SPI communication. Pins 4 and 5 can be used for communication handshaking.

The schematic architecture of the sensor unit is shown in Figure 6-11. \( U8 \) and \( U9 \) illustrate the temperature and tilt sensors respectively. The capacitor acts as decoupling cap for the power supply of \( U8 \). The two resistors, \( R9 \) and \( R11 \), are current limiting resistors in the event of a short circuit between the I/O pins of the microprocessor. Pull-down resistors, \( R10 \) and \( R12 \), are used to stabilise the outputs of the tilt sensor when no contact is made inside the sensor device. [38]

Please see [27] and [26] for more information on the temperature sensor and tilt sensor respectively.
6.3.2.2 F/U 2 – Mini Controller

The schematic architecture for the power supply unit of the mini-controller is shown in Figure 6-12. In this architecture, a 5V DC regulator and a 3.3 DC regulator exists, represented by U1 and U2 respectively. They are of the same series (LM1117), but are bought with fixed output voltage from suppliers. The capacitors (C1, C2, C3 and C4) seen in the figure all have the function of dc filtering and stabilization. In [39], technical information on the voltage regulator is given. [38]

In Figure 6-13, U3 represents the active-high dip switch, used by personnel, to set the mini-controller for different operational functions. R1 and R2 stabilizes the inputs to the microcontroller to ground (not a floating value), where R32 and R33 are current limiting resistors in the case the I/O pins of the microcontroller short circuit. [37]
Power is connected to the mini-controller via connector U4. This is shown in Figure 6-14. C5 acts as a DC filtering and decoupling capacitor for the DC input and D1 protects the circuitry of the mini-controller in cases when the personnel reverse the polarity of the input power lines. [38]

Flash drives are connected to the mini-controller via a flat type A USB socket as indicated by the schematic architecture shown in Figure 6-15. The connector is represented by U5 and C6 is a capacitor used for DC filtering and stabilization of the power delivered to the flash drive.
Figure 6-16 shows the complete schematic architecture for the flash drive controller. The required crystal frequency and additional loading capacitors are indicated by X1, C7 and C8. The device requires an external phase-lock loop (PLL) filter, which is done with the RC circuit of R3, C9 and C10.

The device has an active high test pin, which is deactivated by R4 (holds the pin at ground level). It is always in reset, because R5 pulls the active-low reset pin to ground. The function of R6 is to deactivate programming wake-up through the active-low programming pin, by pulling the pin high. U16 and U17 are used to manually put the device into reset and firmware programming mode.

New firmware versions can be put on the USB flash drive because the device can detect new updates on the drive, after the initial programming. To put the device into programming mode, jumper U17 must be bridged. The device should then be put into reset and then out of reset, by first bridging jumper U16 and then to remove it. The device will then wake up in programming mode.

C11 through C12 are decoupling capacitors with R7 a link between the analogue supply and digital supply. As with the CC2420, this link can be broken for debugging purposes. The pull-up resistors designated R8 through R14 are required to enable communication modes (SPI in this case). U7 is the programming pins of the device. R16 and R17 are recommended by the technical specifications of the device. R17, R18, R19 and R20 are links between the serial programming connector and the I/O pins.
of the microcontroller (F/U 1.2.10. These resistors should be removed when the device is programmed initially. For more information see [32], [40] and [28].

The visual indicators architecture is shown in Figure 6-17. Like with the RF unit, the indicators are open source devices on the microcontroller. There are six indicators, designated L1 to L6, and for each indicator there is a current limiting resistor R1 to R6. L6 indicates whether the device has power and is therefore connected directly to ground through the resistor. [37][38]

![Figure 6-17: Detailed Functional Architecture – F/U 1.2.8.](image)

In Figure 6-18 the schematic architecture of the USB switching circuit is given. The two USB switching devices are represented by U8 and U9. C16 and C17 serves as decoupling capacitors for the switching units, and R27 and R28 keeps the device in a deactivated state when power is applied, by pulling the active-low output enable pins high. The output switch of the device is controlled and activated by the microcontroller. Refer to [29] for more information on the switching devices. [37]

![Figure 6-18: Detailed Functional Architecture – F/U 1.2.9.](image)
The ECU of the mini-controller is shown in its schematic form in Figure 6-19. The PIC18F722 is indicated by U10 and the crystal together with its loading capacitance, is designated by U11, C24 and C25. R29 keeps the microcontroller awake, by holding its active low reset pin high. Personnel can reset the device by pressing the reset button (B1), because it pulls the reset pin low. [37]

C19, C20, C21, C22 and C23 acts as DC decoupling and filtering devices for the power delivered to the microcontroller. The I/O pin used for detecting the RF unit is kept stable by R30. This resistor holds the pin low, because the RF unit will return a high value if connected. The device is programmed through U11. More information on this microcontroller can be found in [41].

Figure 6-19: Detailed Functional Architecture – F/U 1.2.10.

Figure 6-20 shows the schematic for the connector used to connect other devices to the mini-controller. The connector itself is represented by U12. The mini-controller does not supply power, but ground lines are required. Provisions are made for SPI communication and one digital I/O pin. The SPI will allow multiple devices to be connected to the mini-controller.
Chapter 6 – Detail Design

Figure 6-20: Detailed Functional Architecture – F/U 1.2.11.

U13 in the schematic, shown in Figure 6-21, represents the connector used for USB connections between the mini-controller and USB host. It is a type-B square connector and C26 is for decoupling filtering functions of the power received from the USB host.

Figure 6-21: Detailed Functional Architecture – F/U 1.2.12.

The schematic architecture for the controller that controls the USB communication between the microcontroller and USB host, is shown in Figure 6-22. X3, C28 and C29 symbolizes existence of the crystal and loading capacitors of the device. C27 is a decoupling capacitor for the power input of the device and C30 is required for the USB power input. R31 keeps the device in reset by pulling the active low reset pin low. Refer to [30] for more information on the USB232 (U14) device.
Figure 6-22: Detailed Functional Architecture – F/U 1.2.13.

Figure 6-23 shows the schematic architecture for the connections with the RF unit. The connector used is symbolized by *U15*. The functions of each pin are the same as F/U 1.1.7 except they are on different pins, to line them up with the connector of the RF unit.

Figure 6-23: Detailed Functional Architecture – F/U 1.2.14

### 6.3.2.3 F/U 3 – Battery Pack

The battery pack needs to keep the I/O pins used for communication between the RF unit and the mini-controller stable. The schematic architecture shown in Figure 6-24 indicates that all communication pins are pulled to ground (negative pole of the battery) through *R1*. [37]
$U_1$, in the schematic of Figure 6-25, symbolizes the connector used to connect the battery back to the RF unit. No action is required for pin 4 as it is only a return of the voltage.

The battery holder must house two AAA rechargeable batteries. $U_2$ in the schematic architecture shown in Figure 6-26, represents this battery holder.
6.3.3 PCB Functional Architectures

In our development, we had two major evolutionary steps in PCB layout. The PCB layouts of the manufactured XDM and EDM models are shown in this section.

Note. Please note the PCBs have gone through numerous revisions than just these two. Only those manufactured and tested are given. None of the PCB architectures is according to scale.

6.3.3.1 1st Generation PCBs

The first generation PCB architectures for the RF unit, mini-controller and battery pack are shown in Figure 6-27, Figure 6-28 and Figure 6-29 respectively. Altium Designer was used to develop the PCB boards of the three devices.

![PCB Architectures](Figure%206-27%3A%20Detailed%20Functional%20Architecture%20-%20F%2FU%201.1.6%20-%201st%20Generation.)
Figure 6-28: Detailed Functional Architecture – F/U 1.2.2 – 1st Generation.
6.3.3.2 2\textsuperscript{nd} Generation PCBs

Figure 6-31 and Figure 6-31 show the second generation PCB architectures for the mini-controller, Figure 6-32 and Figure 6-33 for the mini-controller, and Figure 6-34 for the battery pack.
Figure 6-31: Detailed Functional Architecture – F/U 1.1.6 – 2\textsuperscript{nd} Generation (Part 2).

Figure 6-32: Detailed Functional Architecture – F/U 1.2.2 – 2\textsuperscript{nd} Generation (Part 1).
Chapter 6 – Detail Design

Development of a modular low power IEEE 802.15.4 wireless sensor module
6.4 Model Design

All the various differences between features and layouts of the two development models that form part of the development scope are explained in this section. Results from tests completed on the devices are used to define to what end each model validates the overall functional capability. The tests themselves are given in Section 6.5.

6.4.1 Experimental Model (XDM)

The first generation PCBs were used as the experimental model. Experiments on breadboards were not possible as some components were of the Quad Flat No Lead (QFN) package. Therefore, we had to design and develop PCB boards for testing the different units and their interfaces.

All development and design decisions given in this dissertation are for the engineering model or EDM. The EDM has all the features of the XDM, whereas the XDM only has limited features. After testing the XDM, the client requirements changed. The most noticeable features that the XDM does not have are:

- No activation or temperature sensor.
- No connector was chosen for attaching peripherals to the mini-controller.
- Different size constraints for the mini-controller.

Despite the differences in features, all internal interfaces of the RF unit, mini-controller unit and battery pack were tested using the same firmware on both models. The firmware was used to ensure that the devices are tested for functional capability (Section 6.5).

Based on the results, we could not verify the IEEE 802.15.4 wireless communication on the XDM model, as well as the sensor unit. These sensors were not available on the XDM and problems occurred during construction (Section 6.5) that made it very difficult to test the communication interface.

The following observations were made regarding the design and layout of the modules:

- Device identifiers were too small and difficult to read. These identifiers are the markings, indicating the names of the devices and component labels, like OmnipoMote.
- PCB layouts can be improved by using better layer structures.
- The 5-pin dip-switch on the mini-controller was an over design as only two pins were needed.
- Human interface components can be placed better for better user-friendliness.

In Figure 6-35, Figure 6-36 and Figure 6-37 3D representations of all XDMs are shown. The 3D representation was made possible through the Altium Designer package. These representations
allowed us to design and develop devices of high usability and configurability, before manufacturing them. However, changes still needed to be made, after examining the final XDM models. These changes are given in Section 6.4.2.

Figure 6-35: RF unit XDM Model.

Figure 6-36: Mini-controller XDM Model.

Figure 6-37: Battery Pack XDM Model.
6.4.2 Engineering Model (EDM)

The main outcome of the development is the delivery of working EDMs. The EDMs should be functionally capable of satisfying the needs of the client, as they will be used to run test scenarios of the system with firmware versions that are close to the final application firmware.

Results obtained from testing the XDM models showed that the devices are functionally capable in all areas, except for the IEEE 802.15.4 wireless communication interface and the communication interface between the ECU and sensors. The sensor unit was verified on the EDM model. Complete verification of the IEEE 802.15.4 wireless communication depends on the results of a test yet to be performed. The device was however verified for IEEE 802.15.4 compliance and for configuration.

EDM models differ from the XDM models in circuit layout, size and looks. The following major and most noticeable enhancements on the XDMs were made:

- All devices are more user friendly, because components are placed better. Bigger and easier locatable labels were used, and interface connectors are better placed.
- Although more components were required on the RF unit to realize the sensor unit, the PCB board of the device is only 2mm longer than the XDM model.
- New requirements from the client required that the new mini-controller be larger than its XDM equivalent (0.5cm higher and 2cm wider). This is because the client wants the RF unit to be inserted into the casing of the mini-controller - over the mini-controller.
- The components used for developing the XDMs stayed the same on the EDMs except for the use of bigger crystals and a connector for the I/O peripherals on the mini-controller. The bigger crystals were chosen because they were cheaper to buy and there is a lot of additional space due to the larger PCB board.
- Some unnecessary decoupling capacitors were also removed, but more current protection was employed. The power supply of the mini-controller is now also protected against reverse polarity.
- The ground and voltage polygons of all devices are better laid out on the new models. In the XDMs only layer 3 was used to relay voltages, but on the EDMs both layers 3 and 4 (bottom) are used. In addition, the ground plane of the power supply is separate from the digital ground plane on the mini-controller. They are connected via a short piece of circuit.

3D visualizations of the RF unit, mini-controller and battery pack in their EDM form are shown in Figure 6-38, Figure 6-39 and Figure 6-40.
Chapter 6 – Detail Design

Figure 6-38: RF unit EDM Model.

Figure 6-39: Mini-controller EDM Model.

Figure 6-40: Battery Pack EDM Model.
6.5 Model Testing

In this section, both models of the three devices are tested for technical and operational feasibility. The tests are completed to determine if the provided solutions are functionally capable of satisfying the needs of the client.

6.5.1 Technical Feasibility

Here we will determine if the developed solutions are technically feasible by delivering on the needs of the client. To determine this, we developed firmware that will test the top-level functional outcomes of the devices.

The top-level functional outcomes for all three devices can be summarized to the following:

- **RF unit:**
  - Wireless communication using the IEEE 802.15.4 protocol.
  - External wake-up using the tilt sensor.
  - Temperature sensing.
  - Communication with the mini-controller.
  - FFD and RFD functional capabilities.

- **Mini-controller:**
  - Receive power over the cable and convert it to the power requirements of the device.
  - Different operational modes
  - Communicate with a USB host PC.
  - Communicate with a USB flash drive.
  - Enable a USB host to access the USB flash drive.
  - Communicate with the RF unit and deliver power to it.
  - Communicate with unknown peripherals in the future.

- **Other:**
  - RF unit can receive power from either the mini-controller or the battery pack.
  - RF unit and Mini-Controller visual indicators control.

Descriptions on how the above output requirements are tested will now be given. These descriptions include the method and the results obtained.

*Note:* Each test is aimed at verifying the inputs, process and outputs of the unit being tested. Refer to the resource allocation tables to see what requirements are satisfied in each test.

*Note:* Firmware for the tests completed is given in Appendix A.
6.5.1.1 RF unit – IEEE 802.15.4 wireless communication

Verifying the devices for IEEE 802.15.4 communication was the most difficult to do, for reasons that will become clear in this discussion.

One aspect that we did not give attention to, in Section 5.3.3.6, is a method to choose a manufacturer that will assemble the devices with high reliability. This neglect is not a result of the development methodology, but because of the inexperience we had early on in the development.

Unforeseen difficulties in using stencils and heating issues lead to numerous construction problems in one area on the RF unit. This is the area of the CC2420 radio transceiver module, balun and matching circuit. The QFN pads of the CC2420, and the pitch between them, is very small. In addition, the balun is very tiny (2mm X 2mm) with four connection pads underneath it. As a result, we had the following problems on both the XDM and EDM of the RF unit during assembly and construction.

- No mechanism was available to hold the stencil in place on the PCB board when we applied solder paste to it. This we had to be done manually and as a result, the stencil moved when we applied the solder paste. The footprint of the balun was a specific area of concern, because of its small size. There were numerous occasions when the solder paste overflowed over all four pads and instances where too much paste was put down.
- The small size and four pads of the balun was also an influence when the balun needed to be positioned over the solder paste. We were unable to place the balun with high accuracy using the manual methods. The solder paste also overflowed between pads as we pressed too hard on the balun.
- These effects were visible when an assembled device was examined with a magnifying glass.

The problems mentioned above can be avoided using the correct machinery, or to use a company that specializes in assembling PCB boards.

Another point of concern is the heating method to reflow the solder paste. At the start, we used a normal household heat gun for this purpose. It functioned well enough for the bigger components, but we had the following problems regarding its use:

- It was impossible to control the spread and temperature of the heat correctly. This meant that we could not follow a reflow pattern, which may allow solder paste not to flow correctly. If used correctly, the air that blows out of the heat gun does not affect the position of the components, but instances were obtained where the components moved when we lost concentration using the heat gun.
- The excessive heat that the heat gun puts out can damage the components, especially the CC2420 radio transceiver and the balun. They are very sensitive to thermal shock.
• The heat gun was primarily used in assembling the XDM, but for the EDM models, we used a reflow oven at the NWU. This oven did not yet exist for the XDM, but none of the EDM models were functional using the reflow oven.

Stencil usage, problems placing components with high accuracy and heat gun issues impact the XDM negatively as we were unable to manufacture a working device, at this time.

As time went on, we did however obtain more experienced in using the stencil and using our manual construction techniques, we were able to assemble a single working RF unit. On this working device, we were able to verify the following aspects for this output function:

• The MSP430F1611 microprocessor was used to configure the CC2420 via SPI communication. Firmware was written to program the parameters of registers that control the actions of the radio transceiver. The parameters of the registers were read back after each write action, to confirm that we were able to configure the device successfully. We were able to write data to any register and read back the same parameters.

• The internal status parameters of the CC2420 were also read by the microcontroller. These parameters were used to determine if the transceiver was initialized correctly. Three specific parameters gave an indication of whether the external crystal is running, the frequency synthesizer PLL is locked and if transmission is active. The status of the assembled RF unit reported a device ready for operation.

• The CC2420 must be set up to be IEEE 802.15.4 compliant by changing the parameters of the registers. Please see the datasheet of the transceiver for this. We were able to successfully configure the CC2420 as IEEE 802.15.4 compliant.

Two of the three aspects regarding this output function are now verified. The first was to verify communication between the microcontroller and the CC2420, and the second was to verify a device compliant to the IEEE 802.15.4 protocol. The last is to verify that the device is capable of transmitting data via radio frequency. We decided to use the following approaches to do this.

• Assemble one more RF unit and let the two communicate with each other wirelessly.

• Use a network or spectrum analyzer and observe the output of the existing RF unit.

Using the same assembly methods as we did with the XDM were we able to manufacture one more RF unit. We had to use manual construction again as we did not have any other resources. This second RF unit was operational in the sense that we could communicate with the CC2420 by changing the parameters of the registers. The status parameters did however not confirm that the frequency synthesizer PLL of the transceiver module is in lock. This meant that the CC2420 on this second RF unit would not be able to transmit data, because the frequency synthesizer part of the transceiver cannot give a signal to its modulator and demodulator parts.
It became impossible to use bidirectional communication to test the wireless link as a result. We had no more components available to construct additional units and because of time constraints, we could not order new components. This forced us to use a spectrum analyser to test the first RF unit.

The following arguments support our decision to test the wireless link of the functional RF unit using a spectrum analyzer:

- IEEE 802.15.4 compliance is verified if the CC2420 is configured correctly and this was already shown to be true.
- The antenna used is designed for wireless communication at 2.4 GHz. It is manufactured by Fractus and is compliant to industry standards like ZigBee, which uses the IEE 802.15.4 protocol. This means that in our design the antenna should function as required.
- Our balun configuration is the same as the solution provided by Anaren, the manufacturers of the balun. In an application note they demonstrate a small and simple balun solution, specifically for the CC2420. In this application note, performance and other characteristics of the solution are also given. The design of this solution was implemented in our design and as such our design should mimic their results.
- The only problem that can have a negative impact on the performance of our design is the layout of the PCB board, but we copied the design in the Anaren application note as best as possible. Therefore, this should have minimal negative impacts.
- A spectrum analyzer can verify the wireless link, because the design should work based on the above.

The following procedures were used to test the RF unit using the spectrum analyzer:

- The CC2420 can be put into several test modes when configured with the microprocessor, to view the unmodulated carrier or modulated spectrum. We decided to detect both outputs.
- The unmodulated signal will pass through the balun and antenna. The output of the antenna (radiation spread) will be used as an input to the spectrum analyzer.
- This unmodulated signal should be the same as the one provided in the datasheet and if it is found to be true, then the wireless link is verified.

We performed the test and observed the outputs of the spectrum analyzer. The spectrum analyzer was set up so that its parameters are the same as the parameters of the unmodulated carrier output in the datasheet (2.45 GHz centre frequency and 2 MHz spread). No signal was seen on the spectrum analyzer when we turned on the RF unit and configured CC2420 to transmit the unmodulated carrier. The spread parameter of the spectrum analyzer was made larger, but still no signal was found. Next, we decided to change the point where we measure this signal.
Figure 6-41 shows the selected test points for measurement on the RF unit. Point 1 is the place where we tested the radiation spread of the antenna output, but here we found no signal. The antenna was removed and we measured the output of balun, but still no signal was found. Parameters for the spectrum analyzer in this test were the same as when we measured the output at point 1.

![Diagram showing test points on RF unit](image)

**Figure 6-41: Test Points on RF unit for Spectrum Analyzer Test.**

We were not able to measure the output at point 3, because it is a balanced port and the spectrum analyzer measures unbalanced ports. The balun is a balanced to unbalanced transformer and therefore we could measure the output at point 2. A high frequency oscilloscope can be used to measure the output of the CC2420 transceiver, but such an oscilloscope is not in our possession. However, we observed the status and other registers of the CC2420 and were able to draw the following conclusions for point 3:

- The external crystal was running.
- The RF transmission of the CC2420 was active.
- The frequency synthesizer PLL was in lock.
- The registers that control the test mode of the CC2420 returned the correct parameters for it be in the test mode it was configured for.
- From the above we conclude that the CC2420 should be transmitting the unmodulated carrier, based on the datasheet. This means that there should be a signal at point 3, but visual confirmation cannot be done since we do not have high frequency oscilloscope. The register parameters and status register is sufficient enough to assume that there must be a signal.

A problem then must exist with the balun because theoretically a signal exists at point 3, but not at point 2. The design of the balun and antenna circuit should work as it is essentially the same as the one provided in the application notes. We can note the following as causes for the incorrect behaviour of the balun.

- Due to the manufacturing process, the balun may be damaged because we cannot control the heat of the heat gun. Reflow ovens have a specific heating pattern make sure that components are not thermally shocked, but this is not the case when we used heat guns.
- It is also possible that the pads of the balun are shorted, because of excessive solder paste.
We propose that this issue can be solved when new components are bought and a company is used to construct and assemble new RF units. This will eliminate all the problems associated with the manual construction techniques used. Only once these new units are tested, can we truly verify the operational capability of the wireless link. For now, we can state the following for this output function:

- A IEEE 802.15.4 compliant configuration is verified
- The MSP430F1611 can successfully configure the CC2420 radio transceiver.
- Verification of the wireless link depends on results to be obtained when we have tested new RF units. Thus, we cannot give a definitive result for this. We can however state that the design is based on a proven and tested design, provided by the supplier of the balun (see [35] for this).

Note: The effect of the above results on the research is explained in Chapter 7.

### 6.5.1.2 RF unit – External Wake-Up

The MSP430F1611 microprocessor can be put into sleep mode using the correct firmware procedures. Any form of interrupt automatically wakes up the microprocessor from this mode to full active mode as specified in the datasheet. Both the outputs of the tilt sensor are connected to IO port pins that can be configured in software for either high-low or low-high interrupts. [36]

Firmware was written to read the outputs of the tilt sensor. We observed that the microcontroller successfully interpreted high-low and low-high outputs of the tilt switch. Thus, we can conclude that this sensor satisfies the requirements.

### 6.5.1.3 RF unit – Temperature Sensing

Another capability of the MSP430F1611 microcontroller is its ability to do 12-bit analogue to digital conversions (ADC). The output of the external temperature sensor is connected to one of the microcontroller’s IO port pins used for ADC. Conversion itself is controlled in firmware. [36]

Test firmware was written to control the ADC and display the result on a PC. It was observed that when the temperature changed the 12-bit ADC value also changed. The 12bit value can be converted into a temperature value using mathematical algorithms, but it is not part of the development scope. The temperature sensor is now verified to be operational.
6.5.1.4 RF unit – FFD and RFD Functions

Requirements for FFD and RFD operational outcomes are handled by both the CC2420 wireless transceiver module and the MSP430F1611 microprocessor. The CC2420 has to be configured with the MSP430F1611 for it to accept and receive beacons, and for it to be the PAN coordinator. Simple SPI communication is used to access the registers on the CC2420 and to configure it. [24]

The microprocessor itself is used to control the network topology after the configuration. All this is done in firmware. The topology recognition algorithms, FFD and RFD functions of the system are not part of the development scope. All we needed to do is to determine if we can successfully configure the RF unit for FFD or RFD operations. This was done using SPI communication and other interface lines between the microcontroller and transceiver module. Because of this, we can say that MSP430F1611 and CC2420 are capable for both FFD and RFD operations. [24]

6.5.1.5 RF unit and Mini-Controller Communication

The IO port pins used for communication between the two devices were designed for SPI communication, as stated in Chapter 5, although this was not a client requirement.

To successfully test all the IO pins used for this function, we decided to implement a firmware controlled communication algorithm. It is essentially the same as hardware SPI, but controlled in software. This communication interface was also used to view results from other tests completed on the RF unit, like temperature reading. Both devices could successfully communicate with each other.

6.5.1.6 Mini-Controller – Power Input

Power is delivered to the mini-controller via the power cable. This power must be regulated to suitable levels for the components on both the mini-controller and the RF unit. An oscilloscope was used to observe the input and output levels and they were at levels that we expected.

6.5.1.7 Mini-Controller – Operational Modes

A dip switch is used to put the mini-controller into different operating modes. In order to verify that the mini-controller is technically capable of being put into different start-up modes, we had written firmware to interpret the outputs of the dip switches. The microcontroller was able to detect the low and high output levels of the dip switch.
Four start-up possibilities exist, because there are two switches on the dip switch. These two switches allow for four different high and low outputs. Firmware can detect this upon start-up and enter different operating modes.

6.5.1.8 Mini-Controller – USB Host PC Communication

Communication with a USB host PC is controlled by the USB232TM device. The PIC18f6722 needs to be configured to communicate with the USB232TM device via UART. Firmware was written for the microcontroller to communicate with the USB232TM and in effect with a PC. [30]

Commands and data were sent from the PC to the PIC18f6722 to control different output actions, like opening a file on a USB flash drive. In addition, data was sent from the microprocessor to the PC during some of the tests completed, for observation purposes. The mini-controller is thus capable of communicating with a USB host PC.

6.5.1.9 Mini-Controller – USB Flash Drive Communication

As mentioned above the ECU of the mini-controller was given commands to interact with a USB flash drive. However, the ECU itself does not communicate with the flash drive, but with the VNC1L controller that handles communication with the flash drive. [40]

Application firmware was written for the microcontroller to communicate with the VNC1L and to control it. We were able to create or delete directories and files, and append or remove any data from files using the capabilities of the VNC1L controller. USB flash drive control and communication is possible based on the results from this test.

6.5.1.10 Mini-Controller – USB Switch

The purpose of the USB switch is to give a USB host direct access to the USB flash drive without going through the ECU of the mini-controller.

Firmware was written for the PIC18F6722 to control the internal configuration of the two USB switches (TS3USB221). We were able to verify the output functions of the USB switch, because we successfully controlled the switches as required by the analyses given in Section 5.7.2.7. [29]
6.5.1.11 Mini-Controller – Communication with Future Peripherals

Firmware was written to control the IO port pins used for communication with other peripherals. An oscilloscope allowed us to observe the outputs of the ECU on the connector and it was seen that the output levels changed as required. The port pins that were used for this can be configured for SPI or I2C communication using the correct firmware procedures. This test enabled us to verify that the mini-controller is capable of communicating with other devices in the future.

6.5.1.12 Visual Indicator Control

The visual indicators of both devices were used during these tests to display internal states of the devices. It was observed that all indicators responded to our control, which verifies that devices are capable of showing conditions to users.

6.5.1.13 RF unit Power

Both the mini-controller and battery pack must deliver power to the RF unit. This was tested by configuring the RF unit first with the mini-controller and then with the battery pack. A test application was written for the RF unit to blink its indicators when it is running.

It was observed that in both cases the RF unit was able to blink its indicators. This can only be done if it has power connected to it. An oscilloscope was also used to observe the power levels on each component and it was observed that each component had the power level that it required.

6.5.1.14 Other

We cannot verify that the devices will fit into enclosures, because none exist. However, the following provisions were made in order to connect the devices to enclosures later on in the development:

- A 2mm no-zone was created on the edge of the PCB boards where the devices can be connected to enclosures.
- The mini-controller has extra mounting holes because of its bigger size.
6.5.2 Operational Feasibility

Here descriptions are given on how we ensured that the devices are also operationally feasible regarding the criteria set out in Section 5.3.3. The following criteria are not described here for reasons already given:

- Performance criteria, because none were received.
- Economic feasibility aspects are handled by the client.
- Environmental compatibility in the operational environment.
- EMC of the devices will only be completed on the ADM.

6.5.2.1 Reliability

Performance and effectiveness factors are required to determine comprehensively if the devices are reliable or not. However, the following points can be noted on this subject:

- A development methodology is used that enables us to include factors that will affect the reliability of the devices from the beginning.
- Well-known CAD tools were used to develop the PCB boards. The 3D visualizations allowed us to make effective detailed design decisions regarding the devices.
- All design decisions were documented and baselines for each level of design were completed.
- Resource allocation tables allow us to identify key points of concern and concentrate on them.
- The devices are designed to be easily maintainable and user friendly.

6.5.2.2 Maintainability

Both types of maintenance can be carried on these devices. The visual indicators allow personnel to easily see the internal states of the devices.

Automatic diagnostic routines (to be written by the client) will enable personnel to only observe outputs and not to interact physically with the devices. All baselines and resource allocation tables can be used to backtrack errors or failures when problems are identified.

6.5.2.3 Usability

All connectors (except VNC1L and MSP430F1611 programming connections) are placed with a section over the edge of the PCB board. This allows them to be easily accessible even if an enclosure is made
for the devices. The VNC1L and MSP430F1611 will only be programmed once via their programming connections.

During the times when we used the device, we never got confused in identifying the correct connector. Connecting the RF unit with the battery pack is also easy and it is a solid connection. XDM connections between the RF unit and mini-controller are better than the way the EDMs connected, because the PCB board of the mini-controller elevates the RF unit.

The visual indicators of both the RF unit and mini-controller are easily readable because of their placement on the PCB board. They are also sharp in brightness, but not so much that they affected our eyes negatively. Labels used to identify the devices were more readable in 3D visualization than in real life.

Upon receiving the PCB boards, it was found that all the printed labels were blurry. This is because of the machines used to manufacture the boards. It is however not a huge concern as other manufacturers may be used or the current labels can be adjusted to make them more readable.

### 6.5.2.4 Manufacturability

The minimum number of parts was used. During designs revisions unnecessary components were removed. Devices are designed so that all components will be used to their maximum potential.

Single sided PCB boards are used to make assembly easier. Standard parts are used except for the balun. The balun is very small in size and there were difficulties to mount it as we did not have the required equipment to do so, but pick-and-place machinery have it better because the places of the components are designed to be easily accessible.

Only the connectors are through-hole components, which make assembly easier. The components are separated so that they can easily be placed by machinery. However, this is not yet tested, but comparing the distances between the components of our boards with specifications on pick-and-place machinery, we determine that we are within safe in that aspect.

### 6.5.2.5 Firmware/Software Compatibility

Firmware used to test the devices was written in objects and functions. Separate header and source files were used for each component. That is, a source file and header file for the communication between the microcontroller and USB232TM as an example. This construction makes the firmware portable and traceable.
Variable and function names are very descriptive, and comments were used throughout files to make it readable. The client can thus use these objects and functions. Complete firmware and software compatibility can however only be determined once the final application firmware is developed.

### 6.6 Design Review

The detailed design phase had more iterations than all the other design phases combined. This is not due to the difficulty in performing the functions of schematic layout or PCB design. This issue was to do it according to the requirements of the client and that of the previous design phases.

All “design to” requirements and operational feasibility requirements were satisfied. The XDMs have gone through three, seven and two iterations to realize the RF unit, mini-controller and battery pack respectively. On the other hand, the EDMs had to go through two, three and one iterations respectively for their completion.

The most noticeable results from the design review performed during this detail design phase are the results from the XDM and EDM models. The feedback from the tests allowed us to develop devices of better quality.

### 6.7 Conclusion

The detail design phase of our process model was presented in this chapter to develop detailed solutions for units. An analysis completed at the start of the chapter showed which functional units are developed using a detailed functional analysis, and which are bought from a consumer market. The detailed functional analysis consisted of an operational analysis, schematic functional architecture and PCB functional architectures.

The operational analysis described in detail how personnel would interact with the devices. Schematic architectures showed detailed interfaces between all the different components of the three devices and the PCB architectures illustrated how these components fit together. PCB architectures for both the XDM and EDM were shown.

Descriptions and illustrations of both the XDM and EDM were given to show how they were used in the development for functional capability testing. Differences between the XDM and EDM were also mentioned and both models were tested for functional capability. Design reviews and feedback was used to update the design with the new requirements of the client as well as the different iterations on the PCB designs.
The SE methodology allowed us to take a system concept, and develop devices that are functionally capable of satisfying the needs of the client. These devices were designed to fit and function in a system, while the form of the devices satisfies the requirements of set out by the operational feasibility criteria. This would not have been possible if SE concepts were not used.

Device testing of the RF unit was inconclusive whether or not the radio channel works or not, due to difficulties in manufacturing and assembly. However, this can be solved using the correct manufacturing and assembling techniques. This then, is our first fault in the development, since we did not give enough attention to these processes. If we had given attention to this during the preliminary design, using a FMECA analysis to identify these processes as a risk, we could have avoided this inconclusiveness, using the correct techniques.

That said, it is our believe that the RF communication should work, using the correct manufacturing and assembly procedures, based on the conclusions drawn from the tests completed on the IEEE 802.15.4 wireless communication.
Chapter 7

Validation and Verification

Now that the development is finished, becomes it mandatory to validate the product against the requirements of the client and to verify that it was completed correctly. This chapter focuses on these validation and verification aspects.

7.1 Introduction

In this chapter, the validation and verification aspects of this research are presented. Unlike the other chapters, this component of our process model is not a step by step process, but is included throughout the development life-cycle in the form of design reviews, design iterations, requirements allocations, trade-off analysis studies, “design to” specifications and design syntheses.

An analysis is completed on the inputs of this research, before the validation and verification of both the development methodology and product begins. This is done in order to determine the examinations required to validate and verify the research.

7.2 Analysis

To complete this analysis we have to go back to the needs and requirements of the client, as well as the methodology of research.
Our client, ZaPOP, have need for a new advertisement system that requires the development of three custom electronic devices. They also want to use the technology of this system, including the three electronic devices, in future years for more complex and unique marketing solutions. The need and concept for this system is described in Section 1.1. In this section, a boundary line is also given on the development scope of these electronic devices.

The client only requires the physical hardware of the devices and no firmware for the actual realization of the system. Firmware for the devices, that will enable the client to use the devices operationally in the system, is developed by the client. However, after the development it must be validated that the electronic devices can satisfy the needs of the client.

Validation results must show that the electronic devices will satisfy the requirements of the client by being functionally capable of fitting into the system. This can only be achieved when a development methodology is followed that allows developers to include all components and third parties (boundary units, design specification and so on), that have an influence on the design of the devices. This methodology needed to be chosen at the beginning of the research and followed throughout the life cycle of the proposed system.

The above described what we wanted to accomplish with this research. Three questions that came from this description, associated with this research, are now present and needs to be examined and evaluated to successfully verify and validate the research. The three questions are:

- Were the right methods used to chose the development methodology? (Verification)
- Was the development methodology used and implemented correctly? (Verification)
- Are the three devices functionally capable satisfying the needs of the client? (Validation)

### 7.3 Methodology Verification

Verification answers questions aimed at the correct use of a research methodology for the problem. In our research, a unique and custom problem was known from the start for which the main objective was the development of electronic devices for a specific client.

Uncertainty and inexperience in commercial electronic development was a great obstacle that needed to be overcome. To meet this challenge, we had numerous consultations with experts in electronic development when we took the advanced electronic development course at the NWU. During this time, a certain development concept was learned, and all examples showed that this is the way to develop the electronic devices. After completing the course, we set out to develop the three electronic devices based on the concept of systems engineering.
Literature showed that systems engineering is not a methodology by itself, but provides tools for a total and complete development. How the tools are used, in a development, is defined by a process model. The first step in our research was to develop a process model for our development and use the principles of systems engineering in it. This process model was developed using the following methods:

- Knowledge obtained during the attendance of the advanced electronic development course.
- Extended literature studies after completing the course.
- Reviews and iterations to establish a true, stable and correct process model.

The requirement to include all components that may have an effect on the design of the electronic devices is now solved by using the correct development methodology. The following system components were included into the design:

- All system personnel, together with their interfaces and points of interest.
- Boundary units (like cables, display machines, power supplies, and so forth), their interfaces and points of interest.
- Different operational domains like the convenience markets, office buildings, depot-and-storage sites, and so on.
- Validation and verification components as a result of design reviews, design iterations, requirements allocations, trade-off analyses and design syntheses.

Although the methodology allows these components to be included, it does not specify the extent to which these components may affect the design of the devices. These effects were determined through different levels of literature studies, operational feasibility studies, component attribute analysis and requirements analysis. Iterations and design reviews assured that only the required functional units are included into the design synthesis.

Development from need and concept to end product is what systems engineering is all about - a top-to-bottom approach. Progression from a system to a product viewpoint is also a purpose of the process model. This development did progress from a client need and concept, to a system viewpoint that was broken up into smaller areas, until the design of the three electronic devices was completed.

Moving through the different focal points of the design, allowed us to design devices that are functionally capable of solving the needs of the client, but to also satisfying the concept of systems engineering – a complete and total development. Because all interfaces, boundary units, operational domains and so on were designed for from the beginning, we were assured that the devices were designed to function within that structure. This means that from a top-to-bottom approach the devices are functionally capable of satisfying the client needs. This is all true, except for the wireless link. Now, we will examine the effect of the inconclusive results obtained when testing the wireless link on our development methodology.
As we mentioned in Section 6.5.1.1 when we described the tests completed on the wireless link, we neglected to foresee the influence of manual construction on the verification of results. Theoretically, the provided solution should work, but hardware difficulties led us to a point where we cannot state whether the wireless link will operate as required. This could have been avoided if we were able to make the correct decisions in the preliminary design when we examined factors that affect the manufacturing of devices. Together with these factors we should have examined the effects of manual construction and we should have chosen a company to construct and assemble our devices.

The neglect of us not doing this could have been avoided if we implemented one more stage in each design loop, just before the design synthesis stage. This design stage should have been the FMECA analysis. We primarily decided to leave it out, because no performance and effectiveness factors were received to which we can compare and analyse the design. However, we could have used the analyses to determine what the probability would be on the reliability of the devices using manual construction techniques. It would then have been foreseen that the radio transceiver module and the balun would not be reliably assembled using our own techniques. If this was done at the end of the preliminary design, we could have selected a company that could assemble our PCB boards and we would have begun construction earlier. However, our methodology has not failed as a result of this.

All other units are functionally capable of delivering on the needs of the client and there is also no delivery date for the electronic devices. This means that there is still time to manufacture new RF units, using a company to assemble the components, and to incorporate the results from a next iteration of tests on the wireless link into the overall success of the research. The reason why this was not completed before handing in this dissertation is given in Chapter 8.

Our problem discussion and research objective set out in Section 1.2 can be answered, despite the result of tests completed on the wireless link. In the problem discussion we stated that we must find a development methodology capable of developing products that will operate in the structures and operational objectives of the company, and in the research objective that it must be a scientific design and development methodology.

he development based on scientific principles since the SE methodology and process model that we followed. This allowed us to develop devices with a form to fit and function in the defined system. Thus, we conclude that our methodology is correct, although not perfect, since we need to include a FMECA analysis. This verifies the methodology from a research perspective,
7.4 Product Validation

It is known that devices should be designed and developed for functional capability only. This level is not a production ready model, but a model at a level appropriate for the client to use and test in an operational environment. The product is validated using a bottom-to-top approach.

The top-to-bottom approach of the development methodology allows us to identify boundary units, interfaces, operational domains, and so on. To validate the devices for functional capability within this scenario, as required by the development scope, we have to test the technical and operational aspects of the devices. The results from these tests were given in Section 6.5.

All internal and external interfaces as shown in the detailed architectures of the preliminary and detailed design phases were verified for functional capability, except the wireless link. Remember, the RF unit was verified for IEE 802.15.4 compliancy and the results for the wireless link are still pending, since we await the construction of new RF units. However, we can still validate the end product up to where we have been able to verify it.

The solutions and detailed designs of the devices and boundary units, except for F/U 1.1.1, are ensured through the validation of the internal and external interfaces, via the tests performed on them. Tests that verified the internal interfaces do not just validate the communication between units, but also the design decisions taken to derive their solutions. In addition, the verification of external interfaces validates the device’s existence in the system. The detailed internal and external interfaces are derived from higher design level literature studies, requirements analysis and allocation, and results obtained from a trade-off analysis.

To fully validate the functional capability of the three devices (high-level requirement) all low-level requirements must be satisfied, which was done with the tests performed. Therefore, we can state that the devices, except F/U 1.1.1, are functionally capable of satisfying the needs of the client, but since no effectiveness factors exist for the system, we cannot state to what end. Let us look how the results of the tests performed on F/U 1.1.1 influenced the validation of the research.

We remind the reader that there is no definitive result for the tests performed on the wireless link, and all indication is that the fault lies in the assembly method of the RF unit and not the design. Tests to be performed on new RF units will result in one of two possible outcomes. The result will either indicate if the link is active or inactive.

An active link will influence the research positively, because it not only verifies the solution and design of the balun, but it also verifies our conclusion that the manual construction techniques we performed were ineffective and damaged the balun. An inactive link means that our conclusions were wrong and that we have to either correct our design or find a better solution for the antenna unit. This means that
we have to go back and change our decision-making in the preliminary design or detailed design decisions, or both. This is where the effects of SE and our process model really come to an advantage.

There will be no effect on the functioning of the other units if we were to redesign and redevelop the antenna unit, because our approach ensures this. We were able to break a complex system into individual resources and components that make them what they are. The interfaces between the resources and components, and the effect of them on each other, were designed for from the beginning. This is why we can say that each component has an input, a process and an output. Therefore, there will be no negative impact on the functioning of the other components or units if we were to redesign the antenna, knowing its inputs and outputs.

### 7.5 Overall

In [42], the author sets out to give principles on how to validate systems engineering research. This article states that there is a big scope difference between conventional research and systems engineering research. Conventional research tends to be focussed on a small area within a field of study, whereas systems engineering research includes all information in all fields associated with the problem. [42]

The bigger the scope, the more difficult it becomes to validate the research. Four different categories of validation are defined for systems engineering research: technical feasibility, technical value, practical feasibility and practical value. Our development falls fully into the case of the first category and partly into the third – devices that are technically feasible and practical. [42]

All four different methods of verification as defined by the author were included in our own validation and verification assessment. These four methods are observation, reflection, data collection and augmentation. Data for our validation are requirements, functional analyses, requirements allocation tables and trade off analyses, which is augmented by the synthesis stages in each design phase. Augmentation reflects the results from all the observations made and data collected. [42]

Technical feasibility is achieved through the development of a working model. Practical feasibility is more difficult to validate as all external interfaces and influences need to be validated. As entities responsible for this do not yet exist (not part of this first development scope), it is impossible to conduct experiments on them. However, this can be validated by means of a well design research approach. This was done using our development methodology (functional architectures, requirements allocation, design to specifications, technical and operational verification tests, and so on). [42]
7.6 Conclusion

The validation and verification aspects of the research were handled in this chapter. We set out to verify the aspects associated with the development methodology from the point where the problem was presented, to the developed process model, as well as the actual development. Product validation was completed and it was shown that the devices are functionally capable of satisfying the needs and objectives of the client, except for the wireless link.

The impact of the results obtained from testing the wireless link was completed on the methodology and product. On the methodology we found one shortcoming, and that is a FMECA analysis. A FMECA analysis will enhance the methodology as it will serve the function of risk assessment. This assessment will help engineers to identify possible problems in the development before they happen. The methodology is verified, based on the initial problem discussion of the research.

Complete validation of the product was not completed as we were able to validate all aspects of the product, except the wireless link. To complete the tests on the wireless link, we have to fabricate and assembly new RF units with the right procedures. However, it is believed that the link will work when the correct procedures is used, based on the results from tests completed on it and the discussion given in Section 7.4.
Chapter 8

Conclusions

Final thoughts and remarks of the author are given in this chapter. Lessons learned and experience obtained during this research is given, as well as possible improvements for the methodology and devices.

8.1 Introduction

In this concluding chapter, the research presented is discussed from the viewpoint of the author. Firstly, the research problem is examined to see whether the end objective of the research was achieved or not, and to what end. Discussions on the methodology used and the devices themselves are given thereafter. Possible improvements for the methodology and devices are given and contributions to the research community are also given.

8.2 Research Objective

The research objective asked us to use a scientific design and development methodology to develop the three devices of our client. This methodology needed to be verified and the functioning of the devices validated to the extent of the supplied development scope. Please see Sections 1.1 and 1.2 for specific details on the research objective and the development scope.
In the previous chapter, we set out to verify and validate the development. Based on the results from this chapter, we can draw the following conclusions for the research:

1. The methodology followed is based on the science of system engineering, as required by the research objective. One shortcoming was identified in our process model that could have made a positive influence on the overall development. However, it was verified that the methodology was correct and appropriate for our development.

2. The three electronic devices were also validated, except for the wireless link of the RF unit. Both the mini-controller and battery-pack are completely capable of satisfying the needs of the client. The RF unit is IEEE 802.15.4 compliant and the sensors of the unit were also verified, but verification of the wireless link depends on the results of a test yet to be completed.

The advertisement system of the client is at this point not yet functional, but there is no delivery date on the devices. This means that there is time to complete the test on the RF unit and use the results to either validate the RF unit as functionally ready or to complete a third iteration of the design thereof.

Due to limitations in finances and time constraints, were we unable to manufacture new RF units and perform tests on them. However, the results from the test would not add anything new to the results of the research, other than the fact that the wireless is active, making the RF unit functionally capable of satisfying the needs of the client. The current status of the development allowed us to demonstrate the following in addition to the research outcomes:

- How decisions, based on incorrect assumptions and data, can negatively affect the projected outcome of development.
- The positive effect that a FMECA would have had on the research.

The total development is still in its first phase, pending the results of the tests to be completed on new RF units. This research demonstrated that we successfully and scientifically evaluated our problem.

### 8.2.1 Development Methodology

The methodology, and principles behind it, has shed light onto the objectives and outcomes of this research. It paved the way forward as we had little experience and understanding in advanced electronic development.

Through consultation with advanced electronic development experts we became familiar with SE and our knowledge grew through extensive studies completed on the subject. This new found knowledge is great importance towards this research and to the client. The following personal benefits are noted:

- How to evaluate client objectives, needs and requirements, and develop products from them.
• Integration of literature research and practical development.
• Multiple fields of study– wireless communication and design specifications for example.

The personal benefits are beneficial to our client since the engineer will go and work for the client in the near future. Clients that will use the system of ZaPOP will also benefit from this. This research has a commercial contribution due to this. The personal gain was put to use in this research and will also be put to use when working at the client.

A process model was first developed and improved throughout the design and development of the three devices. To reflect on the examination in Section 1.3.1, the process model enabled us to:

• Fully understand the utilization outcomes of the devices within our client’s business structure.
• Understand the different development criteria of the devices - fit, form and function.
• Incorporate projected system entities that will have an influence on the operational outcomes of the devices, like the boundary units of the devices, enclosures and so forth.
• Client inputs and requirements gathered from literature studies were handled together, to design and develop devices that satisfies both client and literature requirements.
• Include several fields of study into the design - WSNs, DSNs, design rules, and so on.

The process model and systems engineering principles also allowed for the validation and verification of the research, because of the following:

• Design review types and feedback.
• Requirements analysis and allocations.
• Design synthesis.
• Development models – XDM and EDM.

The XDM allowed us to test different interfaces and components defined by the design synthesis. Results obtained from these tests and further design reviews were used to develop EDMs. The EDMs of the mini-controller and battery pack are functionally ready for the client to use. However, the EDM of the RF unit requires, at least, one more design iteration. The effectiveness of the methodology is discussed in Section 8.3.1.

8.2.2 Device Overview

During the development and design of the electronic devices, experience and skills were obtained in electronic development, which will benefit the client in the future. The most important skills and experience aspects obtained are:

• Electronic design and development.
• PCB design and development in Altium Designer.
• Electronic component selection and configurations.
• How to use “design to” specifications in the design of a product.
• Firmware development.

Illustrations of the RF unit, mini-controller unit and battery pack is given in Appendix A. It also includes pictures of how the units connect to one another. A full description of the device’s overall function capability is given in Section 8.3.2.

8.2.3 Room for Improvement

The following improvements can be made on the development methodology or the devices.

8.2.4 Improving the Process Model

In the validation and verification chapter, we stated one aspect that could have helped us to determine the effect of certain development decisions on the overall development, but there are two more. The following can be implemented to make the development methodology better:

• An FMECA stage at the end of each design phase, but before the design synthesis. The first phase where this should be implemented is in the phase where the needs and objectives of the client are formulated. It should start at this phase, because it would benefit both the client and the developers to have a common understanding of where the difficulties lie.
• A risk assessment at the beginning of every design phase, after the input analysis, and starting at the conceptual design phase. Here, the results from the FMECA are used to identify mayor risks for the current design level. This can be used by the developing team to put their focus on the aspects that required it the most.
• Methods for client conversations.

Good and meaningful conversations should be established, as information regarding the needs, objectives and concepts are established during that time. In our process model this was the starting point of the development and conversations were used through all the design phases, to formulate new or update existing needs, concepts or requirements. The following points of concern with client conversations were identified:

• Focus difference between the client and developing engineer. The focus of the client tended to drift away from key points that needed to be discussed (unnecessary waste of time and finances). This is a place where an FMECA and a risk assessment could have helped.
• Client requirements versus technical feasibility within the budget and size constraints.
• Increasing and changing development scope. This was probably the most influential point that influenced our development negatively. Written and signed documents can be used to limit the effects of changes, because they would fix down requirements at an early stage.
• Written confirmations of client visits are necessary, because it will eliminate or in most cases minimize the times where one of the parties do not come to an agreed upon meeting.

The above becomes better with more experience, but there was a big age and experience difference between the developing engineer and the client. Based on our experience, age and experience differences play a huge role is developments such as this one.

8.2.5 Improving the Devices

The following improvements on the devices are possible:

• The surface mount indicators can be replaced with elevated through-hole indicators once information on the enclosures becomes available. If this is done, the enclosures can be designed to have holes into which indicators can be fitted.
• All PCB boards have a 2mm zone at the edge. This no-zone can be used to design enclosures into which the RF unit and battery pack are slide into. The no-zone of the mini-controller can be used as a support mechanism for the device in its enclosure. The mini-controller is also fitted with mounting holes, which can be used to hold the device in place within the enclosure.
• The current EDM of the RF unit attaches to the EDM of the mini-controller with a slight incline in its length, because the through-hole components stick out underneath the RF unit and presses down the PCB board of the mini-controller. We suggest that the connection be changed to manner in which the XDMs connect with each other, or the design of the mini-controller PCB board be changed so that the piece of the board underneath the RF unit be cut out.
• If the test, to be completed on the wireless link of the RF unit, requires a new solution be developed for the matching circuit between the radio transceiver and the antenna, then a lumped element implementation can be used and tested. However, this will make the size of the PCB board substantially larger.
• The antenna of the RF unit can be replaced with a smaller one from the same manufacturer. It has the same length as the one used, but only a quarter of its width, with almost the same radiation spread. This will allow us to use a lumped element circuit for impedance matching between the radio transceiver and antenna. It will enforce no change on the size of the RF unit and will also make manufacturing and assembly easier.
8.2.6 Contributions

Projected contributions that this research may have were stated in Chapter 1. We conclude the next as contributions, now that we have completed the research.

The client and engineering community benefits from an engineer that obtained new skills in:

- Systems engineering principles and development.
- Technical understanding and development.
- PCB development in Altium Designer.

This dissertation is an example of a development where a client, from the public sector, requires the design and development of devices with specific needs and objectives. This dissertation is publically available to be studied or referenced by any developing engineer, student or researcher.

ZaPOP, our client, is influenced positively by this research. The skills and experience obtained by the developing engineer will be beneficial to our client. Based on the results of the research, the employees of our client can start to use the mini-controller and battery-pack devices to get comfortable with them while the wireless link of the RF unit is verified.

Employees of our client can use the current versions of the RF unit, mini-controller unit and battery pack to develop the final application firmware that will be used, except for the wireless communication part. The last part of the software can be developed once the wireless link is verified. The client does not lose any time due to this, because the wireless link should be verified by the time the application firmware for the RF unit and mini-controller unit are written.

The system concept of our client will become a reality only when the wireless link is verified and when the final application firmware is written. This will mean that our client will stay in the forefront of the in-store marketing arena.

8.3 Final Comments

In Section 8.2, we addressed the results of Chapter 7 on the research objectives, outcomes and contributions, given in Section 1.3. We now would like to give concluding remarks on the methodology and the developed products, to emphasize their effectiveness and functionality respectively.
8.3.1 Methodology Effectiveness

The methodology used was effective in numerous aspects of the design, except for one. Risk assessment and analysis must form an integrated part of development, which we neglected to use.

SE does allow for a huge variety of risk assessment techniques to be implemented. We did demonstrate what the effect of this would have been on the manufacturing and assembly aspects of the products, even if it is for prototyping only. That said the methodology chosen for this development was effective, especially in the following perspectives:

- It not only allowed us to concentrate on what is important, but also when it becomes important, due to the different phases and stages.
- All lower level objects, resources, requirements, and so forth are related from higher level ones. This means that no unnecessary development occurred.
- Each functional block was handled as an independent, yet integrated entity of the system.
- The fit and function of the devices was known and defined long before any form was developed.

8.3.2 Product Functionality

Theoretically, the devices should be completely functional to fulfil in the needs of the client. We believe however, that if all aspects are not fully tested or validated experimentally, then it does not work, even if all studies indicate that it should work. Therefore, at this point in the development, we can state the following about the overall functionality of the devices:

- The mini-controller fulfils all the requirements of the client – multipurpose USB applications, power outputs, connectors, user friendliness, and so forth.
- The battery pack fulfils all the requirements of the client – batteries to use, connectors, purpose, and so on.
- The RF unit fulfils all the requirements of the client, except for RF communications. This means that the RF unit is IEEE 802.15.4 compliant, capable to operate in WSNs and DSNs, capable of connecting with the mini-controller and battery pack, and so forth are all solved, but wireless operation not.

This does not mean that the client does not get what he asked for, because this is an ongoing development, that will be finished while the engineer works at the client.

To date, we still seek answers on what went wrong regarding the wireless capability of the RF unit. It is not just the fact that the methodology did not include a risk identification and assessment stage. There are more factors, but they cannot be held as research items, because they are external to the research
objectives given in Section 1.3. The following external influences can be held as reasons why the RF unit was not completed up to level of functionality of the min-controller or battery pack:

- **Client discussions.** The most difficult point of all, as the client requirements constantly changed. There was a small window of opportunity for us to get the correct equipment to manufacture and assemble new RF units, but we did not receive any confirmation from the client to go ahead with the purchase.

- **Finance difficulties.** It was very difficult to get finances released for purchases. Most of the time, we used our own finances and then requested reimbursement from the client, but that can only take you so far. This was especially true for the manufacture and assembly equipment.

- **Experience.** One must also keep in mind the fact that in the beginning of this research we did not yet have all the experience and knowledge we now have. If we did have the experience we would have foreseen most of the risks, even without a risk assessment.

Each aspect of the research was a learning curve that opened up new possibilities, not only for us, but also for the client and the research community. We learned more from the failures and mistakes of this research, than its successes. The next step is to put the knowledge we gained into practise, in upcoming developments and research objectives.
Bibliography


Bibliography


Fractus, "Small Chip Antenna for Handset, Headset PDAs and USB-dongles," FR05-S1-N-0-001 chip antenna datasheet, [Revision date unknown].

Active, "Product Data Sheet Tilt Switch," RBS 040200 datasheet, May 2003 [Revised May 2003].


Texas Instruments, "High-Speed USB 2.0 (480-Mbps) 1:2 Multiplexer/Demultiplexer Switch With Single Enable," TS3USB221 datasheet, Nov. 2006 [Revised Nov. 2008].


Fractus, Fractus Reach Xtend™ Bluetooth®, 802.11b/g WLAN Chip Antenna - FR05-S1-N-0-001 User Guide, Apr. 2008 [Revision date unknown].

Anaren, "Ultra Low Profile 0404 Balun 50? to 200? Balanced," BD2425N50200A00 datasheet, [Revision date unknown].


Appendixes

10.1 Appendix A – Data CD

A data CD comes with this dissertation to provide the following information to the reader:

- **Appendix A.1 – Research Documentation**: A replica of this dissertation in pdf-format.
- **Appendix A.2 – Referenced Literature**: Literature referenced in this dissertation in pdf-format.
- **Appendix A.3 – Unreferenced Literature**: Literature not used in this dissertation, but used during the research in pdf-format.
- **Appendix A.4 – OmnipoMote Test Firmware**: The firmware code used to test the functional capability of the OmnipoMote.
- **Appendix A.5 – OmnipoController Test Firmware**: Firmware code that was used to test the functional capability of the OmnipoController.
- **Appendix A.6 – Illustrations of Devices**: Visual illustrations of the devices are given.
1 O Lord our Lord, how excellent is thy name in all the earth! Who hast set thy glory above the heavens.

2 Out of the outh of babes and sucklings hast thou ordained the strength because of thine enemies, that thou mightiest still the enemy and the evenger.

3 When I consider the heavens, the work of thy fingers, the moon and the stars, which thou hast ordained;

4 what is man; that thou art mindful of him? And the of man, that thou visited him?

5 For thou hast made him a little lower that the angels, and hast crowned with glory and honor.

6 Thou madest him to have domininion over the works of thy hands; thou hast put all things under his feet:

7 all sheep and oxen, yea and the beasts of the field;

8 the fowl of the air and the fish of the sea, and whatsoever passeth through the paths of the seas.

9 O Lord our Lord, how excellent is thy name in all the earth.

- Psalms 8