UNIVERSITY OF STRATHCLYDE

Optimal Wireless Technologies for the Internet of Things (IoT)

by

Darshana Thomas

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Centre for Intelligent and Dynamic Communications

Department of Electronic and Electrical Engineering

May 2017

Declaration of Authorship

I, Darshana Thomas, declare that this thesis titled, 'Optimal Wireless Technologies for Internet of Things (IoT)' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Darshana Thomas

Date: 01/06/2017

“Nothing is impossible, the world itself says ‘I’m possible’!”

Audrey Hepburn

UNIVERSITY OF STRATHCLYDE

Abstract

Centre for Intelligent and Dynamic Communications
Department of Electronic and Electrical Engineering

Doctor of Philosophy

by Darshana Thomas

The Internet of Things (IoT) – connection of small smart sensors, actuators and other devices to the Internet – is a key concept within the smart home. To ease deployment, such devices are often wireless and battery powered. An important question is the wireless interface used. As these small sensors are increasing in number, the need to implement these with much more capable and ubiquitous transmission technology is necessary. The ubiquity of Wi-Fi in homes today makes this an attractive option, but the relatively high power requirements of Wi-Fi conflict with the requirement for long battery life and low maintenance. Lower power alternatives, such as Bluetooth and Zigbee, have been proposed, but these have a much smaller installed base. In addition, many Smart Home products are currently available using 433MHz technology.

This thesis considers whether it is possible to reduce Wi-Fi power usage to the point where cheap Wi-Fi based products can be used instead of other protocols. A low cost Wi-Fi inbuilt IoT prototype was developed and tested for the purpose of the experiment carried out for this thesis, part of Treegreen project. The work in this thesis undertakes power analysis of a wireless sensor with a System on Chip (SoC) Wi-Fi module, with and without a separate microcontroller, optimized for low power usage which can be used to control the Wi-Fi module. The Wi-Fi chip used within the prototype is the ESP8266- ESP03. Based on the results, in order to optimize the power consumption of the Wi-Fi chip, an MSP430 microcontroller was added onto the existing device. Finally, the IoT data in LTE network is investigated and compared with the real world IoT data.

Acknowledgements

This work would not have been possible without the support and assistance of many great individuals.

Firstly, I would like to express deepest appreciation to Dr.James Irvine my project supervisor who has been a great support throughout this project. Thank you so much for your valuable suggestions and encouragement from the initial stage of this project. Constant support has enabled the successful completion of this project and writing up of the thesis. Secondly, I would like to thank Tregreen for providing me an opportunity to do the work which has resulted in the completion of this thesis. I would also like to thank my external and internal examiners, Dr. Mario Kolberg and Prof Ivan Andonovic for their valuable suggestions and advice which further helped enhance my thesis.

Last but not least, my sincere thanks goes to my family and friends for their constant support and encouragement, and to my friends Tania Wallace, Ross MacPherson, Edward Wilkie, Chris Harison and Greig Paul for their illuminating discussion which has indirectly contributed towards this work.

Thank you all !

Contents

List of Figures

List of Tables

Abbreviations

1G	1st Generation
2G	2nd Generation
3G	3rd Generation
3GPP	3rd Generation Partnership Project
4G	4th Generation
DL	Downlink
eNB	EnodeB
HSDPA	High Speed Downlink Packet Access
HARQ	Hybrid Automatic Repeat Request
IoT	Internet of Things
IP	Internet Protocol
LTE	Long Term Evolution
MCS	Modulation Coding Scheme
OFDMA	Orthogonal Frequency Division multiple access
PDSCH-	Physical Downlink Shared Channel
PBCH	Physical Broadcast Channel
PCFICH	Physical Control Format Indicator Channel
PDCCH	Physical Downlink Control Channel
PMCH	Physical Multicast Channel
P-SS	Primary Synchronization Signal
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared Channel
PRACH	Physical Random Access Channel
RB	Resource Block

RNTI	Radio Network Temporary Identifier
S-SS	Secondary Synchronization Signal
SSID	Service Set Identifier
SCFDMA	Single Carrier Frequency division multiple access
SoC	System on Chip
TBS	Transport Block Size
TD	Time Division
UE	User Equipment
UL	Uplink

Dedicated to my...

Family especially my parents

Chapter 1

Introduction

1.1 Introduction

The Internet of Things (IoT) is a networking paradigm whereby small sensors, actuators and other devices are connected to the Internet, either directly or through a hub. The devices then become accessible remotely, and so provide flexibility for the user to control them from anywhere with the availability of an Internet connection. Smart Homes are a prime example of an application which utilizes small sensor devices which relay their data to a centralized hub [4]. This information can then be redistributed to users or devices requiring data input. For example, a central heating system would benefit from temperature sensor readings.

System-on-Chip (SoC) technology [5] has allowed very complex wireless modules to be marketed at lower cost, encouraging the take up of the paradigm. Figure 1.1 illustrates some IoT examples; there are not many applications which can not be envisaged. According to Gartner the number of IoT devices will reach 6.4 billion this year. [6].

Wi-Fi is a relatively complex wireless protocol [7], but recently Wi-Fi modules have become available at a low enough cost to facilitate their incorporation into

IoT devices [8]. This offers significant advantages as most homes in the UK [9] are now equipped with Wi-Fi hotspots, and this combined with public access hotspots in city centers, hotels, and transportation means that Wi-Fi coverage is essentially ubiquitous. Using a Wi-Fi module means that the IoT device has direct a connection to the Internet, and can for example use a web service without requiring a hub, simplifying both deployment and software development. However, the complexity of Wi-Fi means that it is far less power efficient than other more specialized wireless technologies designed for low power sensors. While low power Wi-Fi is currently being standardized, cheap Wi-Fi modules making low cost IoT devices use current Wi-Fi technology [10].

Power consumption is a major constraint for IoT devices, since many are likely to depend on batteries. Some IoT devices can be powered from the mains, for example a smart power socket or central heating controller. It is interesting to note that commercial examples of such devices often already incorporate Wi-Fi modules. However, mobile IoT devices, such as remote controls or smart buttons, are powered from batteries and use alternative wireless technology such as Bluetooth Low Energy (BLE) or 433MHz and Zigbee offering alternatives as wireless protocols specifically designed for low power sensor applications. The 433MHz protocol is power efficient and popular compared to other existing protocols. Although 433MHz is effective for short range communication [11], it is incapable of supporting long distance transmission. The range offered by Bluetooth Low Energy (BLE) [12, 13] is greater than 433MHz and capable of supporting up to 20 devices. Although WiFi is ubiquitous with more range and device coverage than BLE and 433MHz, the current power consumption restrains it from being implemented into IoT sensor devices.

The applications displayed in Figure 1.1 represent a subset of a spectrum of IoT applications with much potential to expand. The software development stage is the process in which the developers decide upon the software to be used to control the hardware of the device. Based on the purpose of the device, developers have

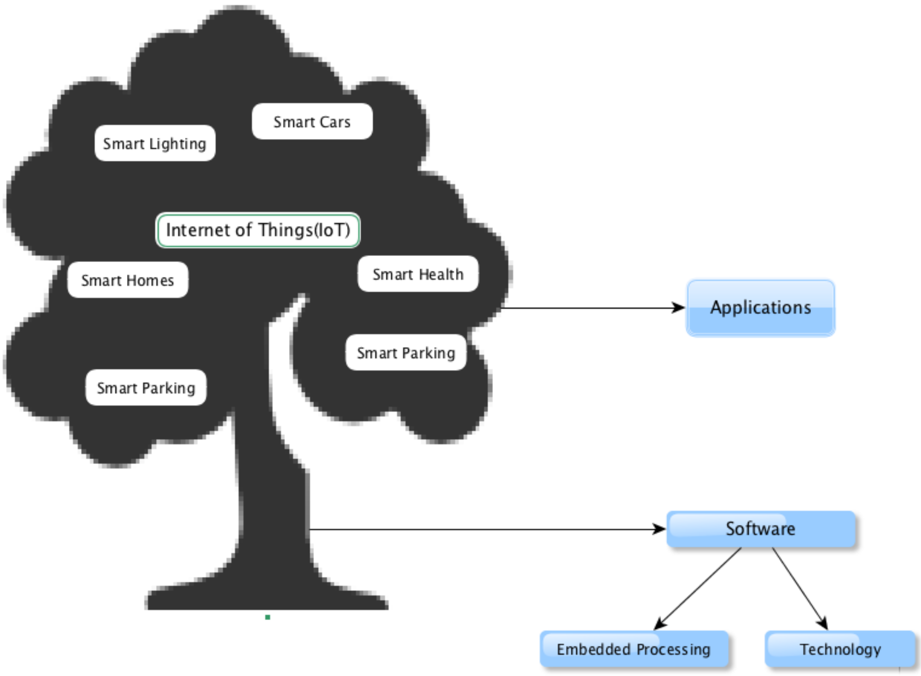

Figure 1.1: Internet of Things Applications

to design specific software for each device. To implement Wi-Fi within a device, the Wi-Fi chip has to be programmed to meet the needs of the application. The processor within the chip is responsible for deciding the functionalities of the chip and the program built on top makes the device intelligent giving the hardware smarter capabilities, to meet the requirements of the application. The thesis, conducted in conjunction with an industry sponsored development for Treegreen, investigates aspects of IoT with particular focus on Wi-Fi enabled IoT devices.

Treegreen is an energy efficiency company based in Glasgow focusing particularly on products such as the energyEgg to improve energy efficiency and reduce electric bills [14]. The company currently focuses on devices for Smart Home particularly integrating Wi-Fi into smart home devices. Based on the discussion with Treegreen, the requirement to develop a low cost, power efficient Wi-Fi inbuilt device without a hub has been raised. Although the concept of Wi-Fi inbuilt IoT without a hub would have been part of the thesis work, Treegreen requirement for a practical device has resulted in the development of the actual device to be commercialized in the future. The concept and the entire development of the device was carried out myself with Treegreens business requirements only providing the motivation.

1.2 Motivation

According to CISCO, 50 billion devices are expected to be connected to the internet by 2020 [15]. This raises the question of how connectivity and the optimization of these devices will be managed. Such a significant rise in connected devices means cogent changes in how to design and organize both devices and networks will be necessary. Figure 1.2 illustrates the initial thought process for this work.

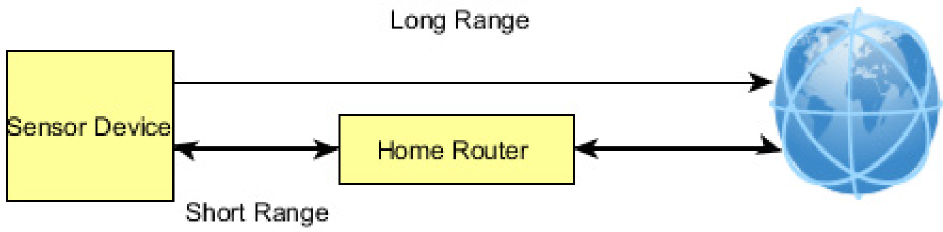

FIGURE 1.2: Design Phase

Based on Figure 1.2 there are two options to connect IoT sensors. Firstly, IoT sensors could be connected directly to the network. LTE offers direct connection of IoT sensors to the network and therefore has been considered in this thesis as the first research question; how well does LTE perform for IoT. Along with the direct connectivity to the network other reasons for choosing LTE for IoT at the beginning of the work is due to the fact that LTE is ubiquitious and can provide good data security as each device will be authenticated by the network. Although LTE can provide a number of benefits as an option for IoT, the complexity of LTE is often acted as a barrier for IoT. An evaluation of IoT sensors in LTE has been carried out in Chapter 3 and based on the results, LTE is not suitable for IoT. An alternative option is to consider 2G for IoT, but this option is restricted for capacity and therefore could not be considered for this research work.

Therefore the second option to connect IoT sensors is to use Wi-Fi, currently been used by 97 percent of UK households [9], with a hub. The disadvantage of using a hub for connectivity is that it requires extra infrastructure increasing the complexity. Power issue is a major constraint for Wi-Fi devices. Therefore the alternative option is to develop Wi-Fi capable device without the extra infrastructure i.e hub and to solve the power constraints of Wi-Fi. This has resulted as the second and third research question of this thesis; can Wi-Fi be considered as an alternative for IoT and is it possible to solve the power problem of Wi-Fi devices. To summarise, the three research questions are:

- **How well does LTE perform for IoT ?**
- **Can Wi-Fi be considered as an alternative for IoT instead of LTE ?**
- **Would power consumption factor of Wi-Fi be solved for IoT ?**

Firstly, the connectivity of IoT devices within Long Term Evolution (LTE) network is researched and recommendations drawn. Based on results, it was concluded that LTE is not an acceptable solution for IoT devices transmitting small amounts of packets. Therefore the utilisation of the ubiquitous Wi-Fi infrastructure was considered for these devices, the second contribution for this thesis. Wi-Fi is power hungry and therefore has to be optimized for power consumption, the third contribution of the work.

The ubiquity of Wi-Fi means that it has secured massive customer acceptance and its ready deployment means that most customers simply switch it on within the home, rather than having to buy additional hubs. Therefore, if Wi-Fi based devices could have acceptable battery life and reduced cost, they would be of interest for consumers. In addition to the ubiquity of Wi-Fi, it has many other attractive features which makes it suitable to develop low power operation. In many applications it has a longer range than alternative technologies, and with a complete Internet stack built in to the module, it offers a plug and play option for service deployment.

1.3 Objectives

This research focuses on developing a practical sensor prototype for IoT application. The second aspect considered is the impact of this data generated within an LTE network. Power consumption of the device is tested and improved with appropriate hardware implementation. The main objectives are as follows:

- Identify the current state of the art within smart home devices
- To investigate the use of LTE for sensor node transmission within the smart home
- Compare sensor node transmission scenarios with the sensor prototype data
- Develop a sensor device with Wi-Fi capabilities suitable for both indoor and outdoor use with reliable connectivity and less infrastructure
- Optimize the Power Consumption of the developed prototype

1.4 Main Contributions

As discussed in the previous section, in order to utilize the full potential of Wi-Fi, power usage had to be reduced for implementations within IoT. The main contributions of this research include:

- IoT data in LTE

  The first contribution of the thesis considered an investigation of LTE as a suitable network for IoT data. Simulation was carried out and compared against the case study included in Chapter 4. The data attained from the prototype was explained in the context of a LTE network and compared with simulation results.

- Low Cost Wi-Fi IoT node - sensor device with Wi-Fi capabilities

  A low cost Wi-Fi module, ESP03, is used within the device along with DS18S20. The design and implementation process is included within Chapter 5 and the results gathered through the device appear in subsequent chapters.

- Comparison with existing protocols

  The power consumption of the device recorded in Chapter 6 is compared against an alternative Wi-Fi device CC3000 for a performance comparison. The results in Chapter 7 are compared with an existing protocol, Bluetooth Low Energy (BLE) and 433MHz.

- Optimized power consumption by coupling the Wi-Fi Inbuilt IoT device with microprocessor

  The third contribution of optimizing power consumption was based on the results of the Wi-Fi inbuilt device. To reduce the power consumption even further and to control the Wi-Fi module an additional microcontroller was integrated into the node. The performance for both, Wi-Fi with and without a processor, were compared before deriving the conclusion.

1.5 Publications

A number of publications have resulted from this work.

[1] D. Thomas, R. McPherson, and J.Irvine. Power analysis of local transmission technologies. In *12th IEEE Conference on PhD Research in Microelectronics and Electronics (PRIME)*, 2016

In this paper the focus was placed on the power consumption aspects of the Wi-Fi IoT device in comparison to the popular protocol based on 433MHz. The paper begins by an introduction into IoT devices followed by a description of the experiment scenarios and sums up with discussion and conclusion. Results illustrated that the developed device is better than 433MHz in terms of power consumption for up to 27 transmissions. The work published in this paper is part of the work in Chapter 4 within the thesis.

[2] D. Thomas, R. McPherson, G. Paul, and J. Irvine. Optimizing power consumption of Wi-Fi for IoT Devices: An MSP430 processor and an ESP-03 chip provide a power-efficient solution. *IEEE Consumer Electronics Magazine*, 5(4):92–100, Oct 2016. ISSN 2162-2248. doi: 10.1109/MCE.2016.2590148

This paper describes the optimisation of the power consumption of the Wi-Fi IoT device by coupling it with a microprocessor - an MSP430. The experiment section of this paper provides an overview of the design of the prototype followed by the discussion of results. Results are compared with an alternative Wi-Fi device CC3000 to provide an indication of the likely system performance of alternative devices. The work in this paper is presented in Chapter 5 of the thesis.

[3] D. Thomas, G.Paul, and J.Irvine. Going beyond the user - the challenges of universal connectivity in IoT. In *Wireless World Research Forum Meeting (WWRF35)*, 2015

The work in this paper explains the challenges of universal connectivity of IoT. The connectivity issues of both cellular and Wi-Fi are discussed in detail. Other aspects discussed are power consumption, security and privacy.

[4] D. Thomas and J. Irvine. Connection and resource allocation of IoT sensors to cellular technology-LTE. In *Ph.D. Research in Microelectronics and Electronics (PRIME), 2015 11th Conference on*, pages 365–368, June 2015. doi: 10.1109/PRIME.2015.7251411

This paper focused on Internet of Things (IoT) sensor data in LTE, as discussed in Chapter 4. The paper begins with introduction of IoT and concepts of LTE followed by a case study and experiment scenario description. With the help of simulation, the transmission of sensor data in LTE is discussed in detail with results followed by a conclusion.

1.6 Thesis Structure

- Chapter 1 - Introduction

  The contents in this chapter provide a brief overview of the topic followed by objectives and motivation. Publications and a short explanation of each published paper is presented, in order to conclude the end of the chapter.

- Chapter 2 -Wireless Technologies for the Smart Home

  This chapter begins with an introduction of the Smart Home concept in the context of small smart sensors. The Internet of Things (IoT) is described, followed by a detailed explanation of different wireless methods of transmission which are currently being used by the IoT devices. Discussion of Wi-Fi and LTE is provided in subsequent sections. Related work carried out on this topic is included towards the end of the chapter followed by a conclusion.

- Chapter 3 - IoT sensors in LTE

  This chapter starts with an introduction followed by sections about wireless sensor network technologies and security. The main work in this chapter consists of evaluating IoT data in LTE. Simulation software is used to gather results. The packet sizes for the simulation is chosen based on a case study at the beginning followed by comparison of real world data packets, based on the results obtained from the developed prototype. The work in this chapter is concluded at the end of the chapter. From the work carried out for this chapter, it is shown that LTE is inefficient for IoT devices transmitting small packets of data. This has motivated to use Wi-Fi for IoT sensor node communication.

- Chapter 4 - Design and Implementation of the Prototype

  The major work leading to the results for the following thesis chapters is due to the implementation of the prototype, which is discussed in detail in this chapter. A brief introduction is provided at the beginning of the chapter followed by a detailed explanation of the components used in the development of the prototype. The design process and the working phase of the product at each stage is explained in detail in the following sections of the chapter with a conclusion summing up the work.

- Chapter 5 - Power Consumption of Wi-Fi Inbuilt IoT Device, ESP03

  The contents in this chapter begins with an introduction of the IoT devices, followed by the power consumption of the developed prototype discussed in Chapter 5. The results are discussed in detail within the first few sections of the chapter. The developed device compared to an alternative Wi-Fi device CC3000 is included with results (Figure 6.6) and implementation details. Dynamic Host Configuration Protocol (DHCP) and Static IP for the device is also considered and discussed towards the end of the chapter followed by a conclusion. The results from this chapter has shown that power consumption of Wi-Fi chip is high and therefore power consumption had to be optimised. This has been a motivation to use an additional microcontroller with the Wi-Fi module to reduce power consumption resulting in the third contribution.

- Chapter 6 - Power Consumption of Wi-Fi Inbuilt IoT Device with an addition of microcontroller, MSP430.

  The first section of the chapter begins with an introduction and based on the results from the previous Chapter, the device is modified to improve power consumption. This is done by coupling the device with an additional microprocessor. The first few sections in this chapter looks into the implementation and working phase of the improvised prototype followed by the results. These results are then compared with the results from the previous chapter. The improvised device results are compared against a popular protocol used within IoT devices, BLE. The following sections is comprised of discussion and results of the improvised device in comparison to BLE and 433MHz. The conclusion at the end of the chapter briefly outlines the work and sums up the chapter. Based on the work carried out for this chapter, it is shown that Wi-Fi chip with the addition of microcontroller has resulted in further power savings compared to the previous chapter. This would enable using ubiquitous Wi-Fi for future IoT devices.

- Chapter 7 - Conclusion

  The conclusion chapter concludes the work carried out for this thesis. It begins with an introduction about IoT followed by the discussion of the research questions solved through this thesis work. The necessity of the work is also discussed within this chapter followed by a future work section in which the work that could be carried out in each of the chapters in this thesis is mentioned.

1.7 Conclusion

This thesis investigates the efficient connection of IoT devices wirelessly. The work investigated LTE before focusing on improving Wi-Fi as a more promising approach. This chapter has introduced the thesis and its content. This thesis will now continue with a description of the smart home concept which will be a driver of IoT.

Chapter 2

Wireless Technologies for the Smart Home

2.1 Introduction

Advances in electronics have brought noticeable differences in our lives where we are surrounded by devices in everyday lives [20], [21]. Whether working on a computer to switching on a washing machine, all includes some sort of processing. The ever-increasing number of devices for tackling the fast paced lifestyles of humans increases the complexity and manageability of these devices. Most people are unaware of how processors work within the device and the thought of it is beyond the comprehension of many people.

However if these devices had to be set-up, the majority of people would be reluctant to use them. Plug and play, off-the-shelf ready to go devices are often more comfortable for people to embrace [22]. In a few years from now, when even more devices penetrate our lives to give us increased flexibility, the issues which will arise are management and power.

Before discussing about wireless technologies it is important to discuss about smart home. This chapter explains the concept of a smart home and the benefits of smart home devices followed by explanation of wireless technologies such as Wi-Fi and LTE. In addition, the work done in this area and the contributions this thesis brings to the topic is also covered.

2.2 Smart Home

The concept of the Smart Home has been emerging in recent years. The introduction of devices into homes has enabled the control of the entire house at the touch of a button. Figure 2.1 illustrates a smart home equipped with multiple devices [1].

The advancement in technology has given users the ability to interact with devices in their home. The home user is given flexibility to control the home with pre installed devices through their smart phone (iPhones or Andriod) via mobile applications [23]. There is also the potential for extension of applications to the medical field for monitoring patients in the home, especially elderly, using these devices.

Sensors alone are incapable of doing what a smart home requires. Other components within these units helps sensors to transmit and receive information from a central unit either with or without a communication interface. The communication interface is used to transmit the data from a smart home through a communication infrastructure such as LTE discussed in Section 2.3.3.

Sensors are devices that detect signals and respond to input on the resultant. Output is gathered electronically for further processing. They form the foundation for many applications such as Medical, Environmental etc. In the context of this thesis the types of sensors used in a smart home are temperature, door sensors,

FIGURE 2.1: Smart Home [1]

etc, which fall under the category of environmental sensing. These sensors implemented within smart homes transmit information with the help of a programmed microprocessor.

Gartner [6] predicts that, by 2022 the number of connected devices in a smart home would be 500. The age of the consumer influences the purchase of smart home devices; based on the maturity of the application and the advancement in technology the number of these devices per home is likely to increase.

2.2.1 Benefits of Smart Home Devices

The benefits smart home devices offer are ease of use, energy savings, interaction and safety [24].

2.2.1.1 Ease of use:

This is probably the most important advantage of providing the ability to control the entire home [25]. Mobile Apps connected to these devices let the user control either a single device in a room or multiple devices simultaneously. A 'Zero configuration' capability reduces the effort from the user. For a one off installation, the unit is set-up by an external contractor and requires no input by the user.

2.2.1.2 Safety:

Before the introduction of smart home devices, if a user forgets to turn off an appliance they have to execute either do manually or ask for help from someone else within the home. However with the introduction of these devices, the user could control the device and turn it off without physically being present. Energy saving is possible and hazardous situations could be avoided.

2.2.1.3 Energy Savings:

As mentioned earlier energy saving [20] is an integral part of the safety. By providing control of multiple or single devices in a home, the user is reassured that the energy is not wasted. If these devices are not available and a light bulb is left turned on in the morning before leaving for work; the chances are the user would have not noticed it and would be turned on until they return home resulting in energy wastage. Energy wastage affects the user as there will be an increase in the power bill due to lack of control of these devices. With the introduction of these devices it is possible for the user to check if any appliances or bulbs are turned on and gives the ability to turn it off remotely if required.

2.2.1.4 Interaction:

Based on the software implementation the device could evolve to additional intelligence e.g. providing the user with feedback at certain intervals. For example if a feedback for an appliance is enabled such automatic updates can be generated. This level of interaction would enhance the experience.

2.2.1.5 Business Perspective

Factors that have to be considered from a business perspective when choosing these devices are cost, resilience and scalability [26]. In terms of cost although consumers are interested to acquire smart home devices some of these devices may not be affordable. Therefore the cost of the device should be low as possible. This aspect has been taken into consideration when developing the prototype, described in Chapter 4. Resilience is the next factor to be considered when purchasing a product. The consumer would like to have multiple functions from the device. It is possible that some deployments of connected devices will wish to ensure connectivity is sustained, even in the event of a local network failure. The third factor is scalability. This is very important if many devices are to be added to the network. Different communication protocols which are currently used for transmission by sensors are discussed in the next section.

2.2.2 Wireless Methods of Data Transmission

The most common methods of wireless transmission used in smart home environments are described below. All can be used to deploy wireless sensor networks, so it is important to consider the system needs before choosing a standard [27].

Several technologies are currently used for smart home environments. Not all technologies currently used were necessarily designed to support IoT principles.

2.2.2.1 ZigBee

The Zigbee (IEEE 802.15.4) [28] standard is an open wireless protocol used for personal area networking (PAN). Low power consumption is a priority, achieved by restricting low bandwidth, short range applications. Range varies greatly depending on the technology and Zigbee can cover up to 50 meters. Zigbee can operate in a mesh networking architecture in which devices relay between themselves until they reach the final target. The Zigbee standard is controlled by Zigbee Alliance [28].

Zigbee uses IEEE (Institute of Electronics and Electrical Engineering) 802.15.4 targeting local area sensor networks. The frequency band it works in is 2.4GHz; as a global standard, Zigbee applications could be deployed anywhere in the world.

Examples of Zigbee applications are security systems, home automation and smart lighting [29].

2.2.2.2 Bluetooth

Bluetooth [30] offers a relatively high data rate (up to 1 Mbps for Bluetooth Smart) that can be used in specific mobile applications. Recent BLE (Bluetooth Low Energy) technologies require reduced power consumption and theoretically allow for unlimited number of devices to be connected in a network. In reality, however Bluetooth is only capable of supporting a limited number of devices. Range can be up to 10 metres or less.

Bluetooth is normally used for short range connections such as printers. Most smartphone devices and automotive media are paired up using this technology to automotive audio systems. Bluetooth offers frequency hopping which makes it a very robust technology. Bluetooth operates between the 802.15.4 standard and Wi-Fi in terms of data rate and power consumption [12, 13]. Several Bluetooth devices would make a wireless personal area network. The Bluetooth standard is controlled by the Bluetooth Special Interest Group [31].

2.2.2.3 433MHz

Amplitude Modulation (AM) at 433 MHz is a commonly used channel for local area communication, due to the low cost of transmission and reception modules. One of the major reasons for this lower cost is that the frequencies within this band are internationally reserved for short range non-specific applications. This means that many manufacturers produce and sell these components, resulting in economies of scale reducing the price of parts. Examples of some of these products are remote controls, garage door fobs and smart LED lights.

Another considerable advantage of using 433 MHz AM for low powered sensors is that the device can be configured so that it doesn't have to join a network before transmitting content, since the receiver is always listening. Having to connect to a network involves keeping the transceiver circuitry powered for longer. However, systems like this rely heavily on having a base station or alternative receiver which means additional equipment has to be provided, powered and maintained for the system to operate. A further limitation of 433 MHz AM is that these devices are designed to operate over small distances, up to 50 meters, usually less without direct line of sight [11]. This might not be able to satisfy all users, particularly users those who reside in expansive environments. Furthermore, basic AM transmitters do not provide any kind of error correction, so the signal interface in devices can introduce signal corruption. This is an existing issue in highly populated areas and can only get worse with more devices operating at the same frequency.

2.2.2.4 Wi-Fi

Wi-Fi or Wireless Fidelity [32] offers very high data rates - theoretically up to 600 Mbps for the most commonly used 802.11n version controlled by Wi-Fi Alliance. A number of different versions are available with different operating frequencies and throughputs. The most widely adopted version currently deployed is 802.11n, which is compatible with early devices, albeit at lower speeds. The latest commercially available version is 802.11ac, offering higher speeds, while retaining the ability to support older devices. While useful for broadband access within the home, in sensor networks, typical Wi-Fi data rates are rarely used to their full capacity. However, the ability to support roaming and send large amounts of information in bursts is ideal for many applications. The range varies on implementation but it can cover up to 200 meters [33].

Wi-Fi offers security, both authentication of devices and encryption of transmitted data. Early versions of the standard were relatively insecure, but current devices implement WPA2 (Wi-Fi Protected Access II), which offers good security, especially given the very low amounts of data transmitted by IoT devices.

There are two types of connectivity for Wi-Fi: connected and disconnected. In the context of connectivity the Wi-Fi device should establish a connection with an access point which would have its own unique SSID and passphrase. After the connection is established an IP address is allocated to the device normally using Dynamic Host Configuration Protocol (DHCP). The access point usually allocates IP address automatically to the user device and therefore after the address has been allocated a full connection is established. These DHCP addresses are leased for certain period of time and therefore the device could use the same address until the lease time expires. For a device to be in the Wi-Fi connected mode the device has to go through all the above mentioned steps whereas in a Wi-Fi disconnected situation the device has to go through each of these procedures prompting the access point for connection each time it tries to get connected resulting in increased power consumption.

2.2.2.5 IEEE 802.11ah

Wi-Fi HaLow or IEEE802.11ah is an extension to the existing Wi-Fi standard [34], [35]. The use of a lower transmission frequency, primarily chosen in order to reduce the energy consumption means that the signal can propagate through walls and doors better than their 2.4GHz counterparts, increasing the range of the network without having to increase the power consumed, more suitable for smart home applications. Wi-Fi HaLow has thus been considered an option for IoT since it allows the support of all IP devices rather than the existing heterogeneous non standardised connections such as Zigbee and Bluetooth.

HaLow was designed with the OSI 7 Layer model in mind so that only the physical and data link layer had to be changed, leaving the remaining layers the same to maintain compatibility - as with previous implementations of Wi-Fi. Changing the electronic circuitry to a new design involves additional cost for the manufacturer, which makes them unwilling to change until compelled to do so by market forces. This combined with the relatively slow uptake for new technologies by average consumers means that although HaLow is effective at reducing the energy consumption, it is unrealistic to expect widespread use for a number of years. Therefore it is still advantageous to investigate ways of reducing energy consumption with already deployed devices.

The gateways and physical layer protocols along with the cost has always raised issues among manufacturers. With IEEE 802.11ah being widely adopted, the need for a gateway could be avoided, if manufacturers include HaLow capabilities within their devices. As Wi-Fi HaLow is at the beginning stage of development for IoT the goal is to create battery operated implementations. The work in this thesis, which started before HaLow was developed, adopts this principle creating devices that operate longer hours on standard Wi-Fi. A comparison table showing the frequencies and estimated range is provided in Table 2.1, a second Table 2.2 details the capacity of each technology.

TABLE 2.1: Comparison of Protocols

	Bluetooth	Zigbee	Wi-Fi	Wi-Fi Halow
Frequencies	2.4GHz	2.4GHz, 868/915MHz	2.4GHz for 802.11b/g/n	900MHz
Range	10m	Upto 100m	250m	10m (expected)
Battery	Days	Years	Hours	Years

TABLE 2.2: Comparison of Protocols

	Bluetooth	Zigbee	Wi-Fi	Wi-Fi Halow
Raw Data Rate	1Mbps	250Kbps(2.4GHz), 40Kbps(915MHz), 20Kbps(868MHz)	11Mbps(2.4GHz), 54Mbps(2.4GHz or 5GHz), 500Mbps(2.4GHz and 5GHz)	18Mbps,150Mbps (estimated)

2.3 Wireless Technologies: Wi-Fi and LTE

Alternative technologies such as the Wi-Fi and LTE has never been considered as a medium of communication for IoT sensors, both are capable of providing better data rate and range than Zigbee and BLE. Zigbee, BLE, other cellular technology such as GSM currently used by IoT does not provide the benefits offered by LTE and Wi-Fi. The subsequent sections looks into two most recent technologies Wi-Fi and LTE for IoT and explains each technology in detail.

2.3.1 IEEE 802.11b - Wi-Fi

Wi-Fi is an IEEE 802.11 standard [7] developed to compliment IEEE 802.3. Wi-Fi offers a full TCP/IP stack when connecting to the Internet. Ever since Wi-Fi was introduced, it has been very popular among users. Almost all of today's technology: laptops, smart phones, tablets, and TVs are equipped with Wi-Fi interfaces. The majority of Wi-Fi networks operate within the 2.4 GHz band. When higher data rates are needed, Wi-Fi is capable of operating in the 5GHz band providing a non distorted signal with more channel space. However, the range of 5 GHz radios is shorter than 2.4GHz which is why the 2.4GHz is preferred to be used within homes. Enterprise applications often favour the 5GHz band over 2.4GHz because it is better at serving multiple access points.

IEEE 802.11b, Wi-Fi, was the first wireless LAN standard to be implemented among computing devices to provide Ethernet level of performance and availability. In July 1999, Wi-Fi was approved by the Institute of Electrical and Electronic Engineers (IEEE) for wireless networking [36]. Based on the approval Wi-Fi implemented products became readily available. Wi-Fi hotspots were deployed in offices, buildings and airports making Internet connectivity easily available while travelling with portable devices.

2.3.2 Wi-Fi Architectures

The architectures and feature of the 802.11b are defined by the standard 802.11. The physical layer is adapted for the 802.11b specification adding more robust connectivity and higher data rates. For example, 802.11b supports up to a maximum of 11Mbps at 2.4 GHz whereas 802.11g supports 54Mbps using a different physical layer and modified MAC layer for backward compatibility to 802.11b.

The main contribution of the 802.11b, addition to the existing WLAN standard, was to use two speeds supported by the physical layer, 5.5Mbps and 11Mbps. Direct Sequence Spread Spectrum (DSSS) [37] is the technique used by the physical layer to support these speeds. The original 802.11 DSSS technique used a Barker Sequence [38], 11 bit chipping, to encode data. The 11 chip sequence is then converted to a waveform called the symbol which can then be sent over wirelessly.

Binary Phase Shift Keying (BPSK) [39] is used to transmit these symbols at a rate of 1MSps (1 million symbols per second). For 2Mbps, an advanced technique Quadrature Phase Shift Keying (QPSK) [40] is used and this doubles the data rate of BPSK. For an increased data rate for the 802.11b advanced coding techniques are employed.

Complementary Code Keying (CCK) is used 802.11b, which uses 8 bit code words consisting of a set of 64. These set code words have mathematical properties that allow encoding of these data at the receiver end regardless of interference and noise from the surroundings. Due to its mathematical properties the receiver could correctly distinguish from one another. CCK is used for the 5.5 Mbps signal to encode 4 bits per carrier whereas 11 Mbps encodes 8 bits per carrier. QPSK modulation and symbol rate of 1.37MSps is used by both the speeds of the 802.11b.

Dynamic Shift Keying (DSK) is also used by the 802.11b at times to support noisy environments and extended range. DSK also allows adjusting the data rate to compensate for the changing nature of the radio channel. Users normally connect up at 11Mbps. If the devices move out of the 11Mbps range then it would automatically be placed at lower speeds at 5.5Mbps, 1Mbps or 2Mbps. Likewise, if the devices moves back within the range then the speed would automatically be increased and would provide the data rate at 11Mbps. The physical layer is responsible for the rate shifting mechanism.

The need for an increased data rate has resulted in amendment to existing standard resulting in IEEE 802.11n [41]. After several round of discussions it was finalized with features such as MIMO and 40 MHz bandwidth. In October 2009 the 802.11n standard was officially released.

The advantage of IEEE 802.11n standard is that it offers higher data rate, increased throughput and range in both the 2.4GHz and 5GHz bands. The data rate extends to 600Mbps from 270Mbps in the PHY and operates to 70m indoors and 250m outdoors.

802.11n uses MIMO - OFDM technique. MIMO (Multiple Input Multiple Output) together with the Orthogonal Frequency Division Multiplexing (OFDM) enables spatial diversity by the use of multiple antennas at the transmitter and receiver end [42]. MIMO supports up to four antennas and four spatial streams compared to the traditional Single Input Single Output (SISO). Spatial streams are bit streams transmitted in multiple spatial dimensions by the multiple antennas used both at the transmitter and receiver end. This in turn increases the diversity gain resulting in higher data rates for the 802.11n standard. In the previous standards 802.11a/b/g only 20MHz was supported whereas 802.11n supports 40MHz.

The main features of the 802.11n standard in addition to MIMO and 40MHz channel operation are Beam Forming [43], Space Time Block Coding (STBC) [44], Short Guard Interval (GI) and low density parity check (LDPC) [45]. All the features mentioned are introduced in the physical layer.

Spatial Division Multiplexing (SDM) increases the throughput by the use of multiple antennas. Multiplexing combines multiple signals into a single transmission channel whereas spatial multiplexing enables the use of several spatial channels to transmit several data bits. STBC ensures signals are received at the receiver end even in difficult environments that could result in loss of signal owing to refraction, reflection and scattering. This technique transmits multiple signals and the receiver combines all copies of the signal resulting in the correct decoding of the signal. Transmit Beam Forming (TBF) is an optional technique used to increase signal strength at the receiver. Spatial mapping matrix generated through channel estimates are used to ensure the signal quality.

Robert G. Gallagher [45] developed the Low Density Parity Check Codes (LDPC) in 1963 but were not implemented in commercial systems until the release of 801.11n. LDPC is mainly used for noise correction in which signals impacted by a noisy channel. It is an error correction code that allows reliable transmission. A Guard Interval (GI) allows seperation between signals without interfering with any other signals. It supports signal protection against reflections and propagation delays. The important 802.11 family specifications are provided in Table 2.3.

Table 2.3: 802.11 Standards

Standard	Description
802.11	Used for wireless LANs and capable of providing 1 to 2Mbps transmission in the band 2.4GHz using Frequency Hopping Spread Spectrum (FHSS) or Direct Sequence Spread Spectrum (DSSS)
802.11a	Enhanced version of 802.11 providing 54Mbps in the band 5GHz and uses orthogonal frequency division multiplexing instead of FHSS and DSSS
802.11b	Also known as Wi-Fi. Provides upto 11Mbps in the band 2.4GHz with fallback to 5.5Mbps,1Mbps and 2Mbps. Uses only DSSS and functionality similar to the Ethernet
802.11g	Used for transmission at 54Mbps over short distances in the band 2.4GHz
802.11n	Addition of 802.11n upon the standard 802.11 to provide Multiple Input Multiple Output (MIMO) capability. Allows increased data throughput by the addition of extra transmitter and receiver antennas and through spatial multiplexing. Range could also be increased by using coding schemes Alamouti coding.
802.11ac	Builds upon the standard 802.11n for increased data rates of upto 433Mbps per spatial stream or 1.3Gbps with three antenns. Works in a 5GHz frequency range and supports wider channels 80MHz and 160MHz
802.11ah	Wi-Fi Halow the first to offer frequency band below 1GHz (900MHz). Range is double than the standard Wi-Fi. Capable of penetrating through walls and barriers providing better connectivity

2.3.3 LTE

Fourth Generation technology, Long Term Evolution (LTE), has been the recent evolution within the industry [46]. It has been standardized by 3rd Generation Partnership Project (3GPP) and has upgraded Universal Mobile Telecommunication Service (UMTS), High Speed Packet Access (HSPA) and Evolved High Speed Packet Access (HSPA+). LTE offers an uplink of 50Mbps, downlink speed of 100Mbps, low latency and high cell edge performance. LTE provides packet switched services, and not circuit switched services, which ensures seamless Internet connection for users. Orthogonal Frequency Division Multiple Access (OFDMA) and Single Carrier Frequency Division Multiple Access (SCFDMA) within LTE has made the whole system capable compared to previous cellular systems. This section gives an overview of LTE. Defining the architecture followed by the uplink and downlink capabilities of the channel.

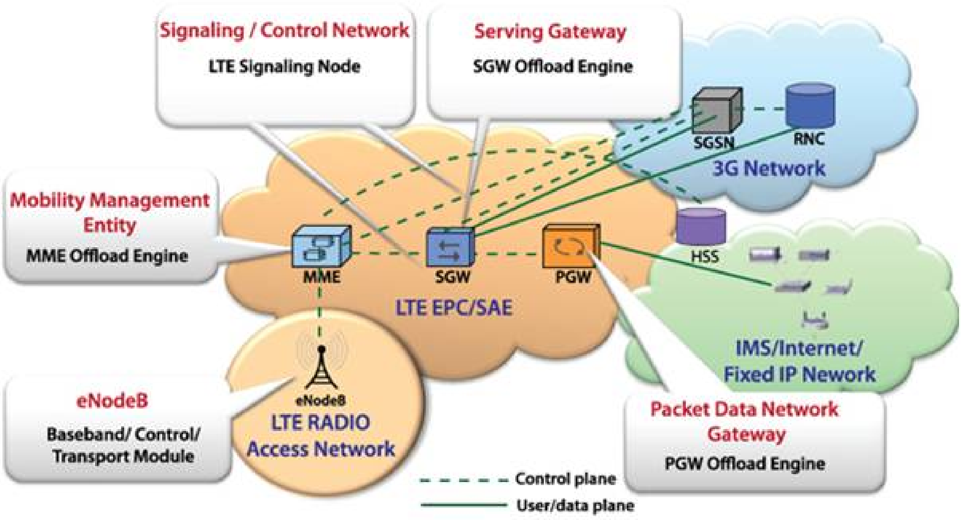

FIGURE 2.2: LTE Architecture [2]

Figure 2.2 displays the LTE architecture divided into two levels: Evolved Packet Core(EPC) (user plane low level) and Enhanced UMTS Terrestrial Radio Access Network (EUTRAN) (control plane high level). Each level comprises different elements, which together form a complete LTE network architecture [47]. There are protocol stacks in each element within the network architecture [48].

2.3.4 LTE Elements

2.3.4.1 E-UTRAN

This level is a user plane level that consists of base stations - Enhanced Node B (eNodeB) - connected between different eNodeBs using the X2 interface and connected to upper level of network using the S1 interface. Base stations are responsible for allocating resource blocks to users within the network [49].

2.3.4.2 Enhanced Node B

The eNode B provides the Radio Resource Control (RRC) for the control plane. It has to schedule and allocate users from service requests. It also has to compress and decompress data for users. Within the eNB protocol stack (Figure 2.3) there are different layers: physical (PHY), medium access control (MAC) [50], radio link control (RLC) and packet data convergence protocol (PDCP). eNode B has

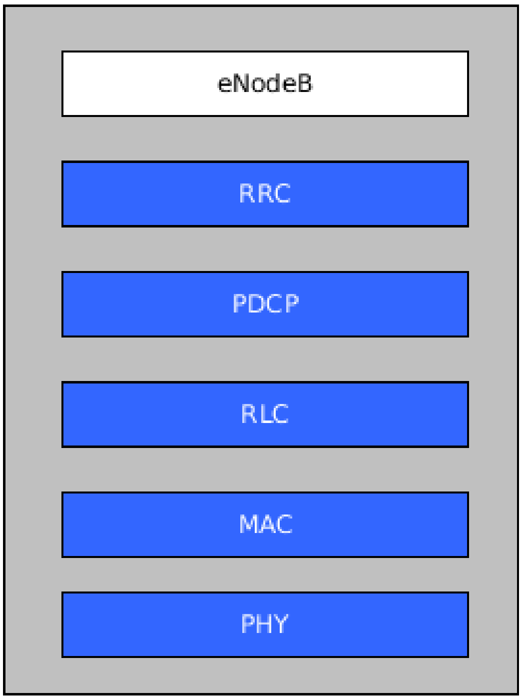

FIGURE 2.3: eNodeB Protocol Stack

to manage multiple cells if the UE is moving from one cell to another; in this case eNode B has to ensure all the user data is transferred to the eNode B where the user is currently using the service. Packet loss should not occur during the handover. In contrast to earlier systems, in LTE the radio controller function is installed within the eNode B to reduce latency and improve handover efficiency. As data is not kept in a centralized node, each time a user moves from one cell to another, the eNode B has to transfer all user data to the new eNode B.

2.3.4.3 Evolved Packet Core (EPC)

This level comprises the mobility management (MME), serving gateway (SGN), packet gateway (PGW) and home subscriber server (HSS). MME and HSS are the core of LTE network architecture, managing the user access to the network. The MME acts on information received from HSS such as authentication, mobility management and security of user information. It is also responsible for the retransmission of data and to identify if the user is in idle mode.

2.3.4.4 SGN

The serving gateway deals with packets received from users. It transfers these packets to the PGW. If the user is in idle mode downlink path can be terminated and use-paging id as data arrives to the user. It is also responsible for inter eNB handovers [51].

2.3.4.5 Packet Gateway

This is the gateway that allows connection between the user and the PDN (Packet Data Network). It filters the packets as they arrive before sending it to the user [52].

2.3.5 Header Compression

Like UMTS, LTE employs a Packet Data Convergence Protocol (PDCP) [53] entity in the upper part of layer 2 to more efficiently handle packet data. When IP packets are being transported, they are delivered from layer 3 to the PDCP, which performs IP header compression. The 20+ byte IP header is replaced with a PDCP header of 1-4 bytes before the resulting PDU is passed to the RLC. RFC2507 [54] is used for header compression, which requires the transmission of the full header periodically for context. An issue with header compression is that while it is effective for continuous data streams at reducing the amount of header data transmitted, where only a few packets are transmitted, such as for sensor nodes transmitting one or two IP packets, the savings will be much less. Therefore, the capacity predicted for IP data streams over LTE when sending large amounts of data will not directly scale when smaller amounts of data are being sent.

2.3.6 Resource Allocation in LTE

Resource allocation in LTE is a key feature enabling users to transmit and receive. Scheduling of resources is carried out by base stations based upon the availability of resources. There are two types of scheduling in LTE: uplink and downlink. In an LTE network, the eNodeB controls scheduling. Base station controlling scheduling ensures quality of service for each user, dealing with overload situation and optimizing overall throughput by reacting to changing radio conditions. The bandwidth slots allocated for LTE are between 1.4MHz and 20 MHz. Each bandwidth slot within this range can only use resources allocated from the network. The higher the bandwidth, the larger the number of resource blocks available [55].

As shown in Figure 2.4, LTE resource allocation is executed in two domains: time and frequency domains. In the time domain, 10 sub frames are available where each sub frame represents 2 slots of 0.5ms duration. Each 1ms slot results in a

- time transmission interval (TTI)- is responsible to transmit every 1ms. TTI reduces delays within a 1ms timescale for transmission. Each 0.5ms slot has 1 resource block made of 12 subcarriers and 6 or 7 OFDM symbols. Even though resource allocation uses resource blocks it is actually the subcarriers within each resource block which are responsible for uplink and downlink transmission. The allocated frequency for each resource block is 180kHz [56].

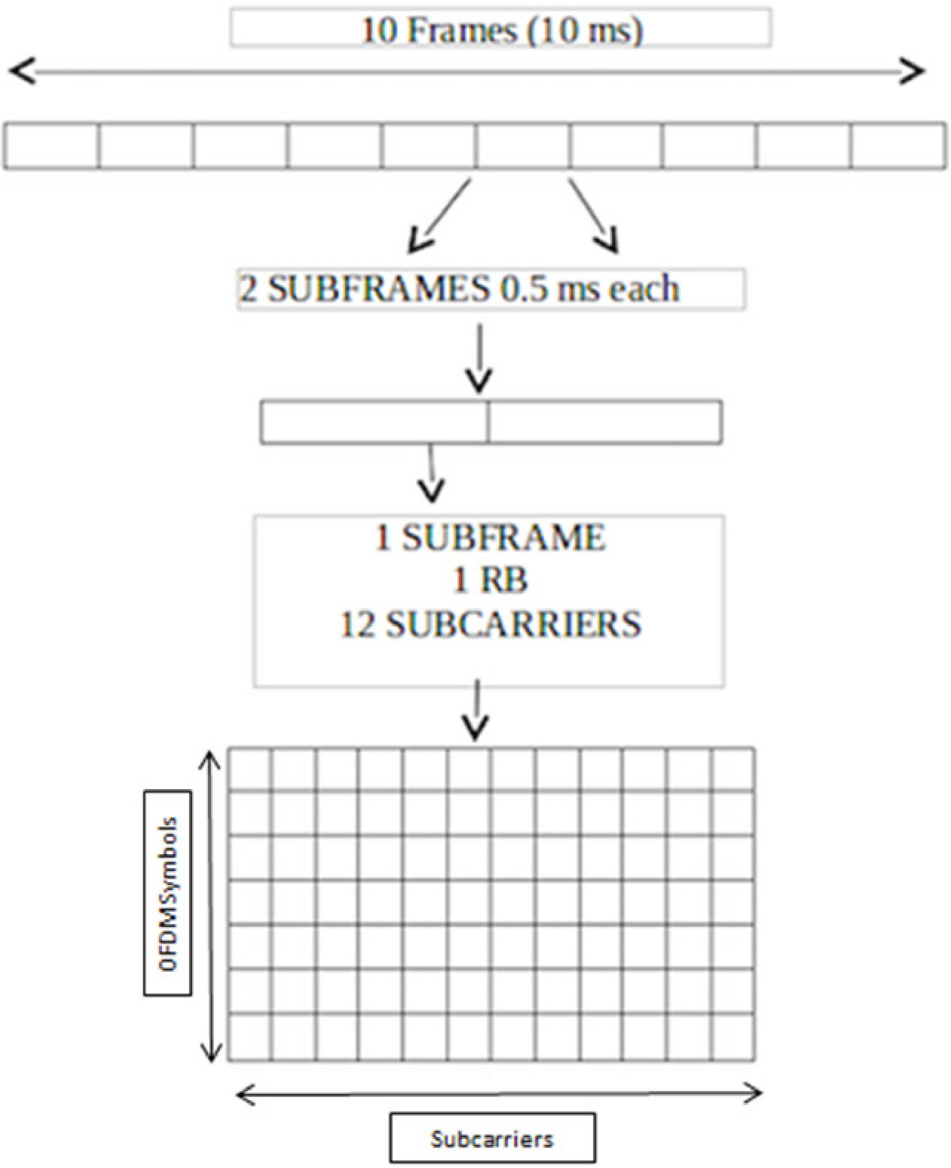

FIGURE 2.4: LTE frame structure

2.3.7 OFDMA for Downlink

A key feature of LTE is OFDM (Orthogonal Frequency Division Multiplexing) used as a single bearer and access techniques such as OFDMA and SCFDMA. For downlink, OFDMA is utilised. The main advantage of OFDMA is that it has twelve subcarriers and can provide radio resources for multiple users. OFDMA uses orthogonal frequency division multiplexing (OFDM) to divide these carriers into subcarriers subsequently modulated using Quadrature Phase Shift Keying (QPSK), 16 Quadrature Amplitude Modulation (QAM) and 64 Quadrature Amplitude Modulation (QAM). OFDMA would use these subcarriers and allocates the bandwidth needed for each user. Any subcarriers not being used are switched off resulting in less interference and reduced power consumption. Even if both OFDMA and OFDM are interconnected it is OFDMA that manages the resources allocated to users [57].

2.3.8 SCFDMA for Uplink

For uplink, Single Carrier Frequency Division Multiple Access (SCFDMA) [58] is used which is a form of OFDM. One of the main reasons for using SCFDMA is that it has a very low PAPR (Peak average power ratio), yielding reduced power consumption. SCFDMA is a pre-coded version. The difference between uplink and downlink is that SCFDMA would allocate each user across a number of subcarriers whereas OFDMA would allocate data in one subcarrier for each user. The reason SCFDMA is not used for the downlink as well as uplink is that as it is more complex than OFDMA, so a more complex receiver is required.

Each lost packet within a cell means more retransmissions to redeliver the packets, resulting in increased latency and more resources being used up which could have been used to accomodate additional user. Practically it is almost impossible to

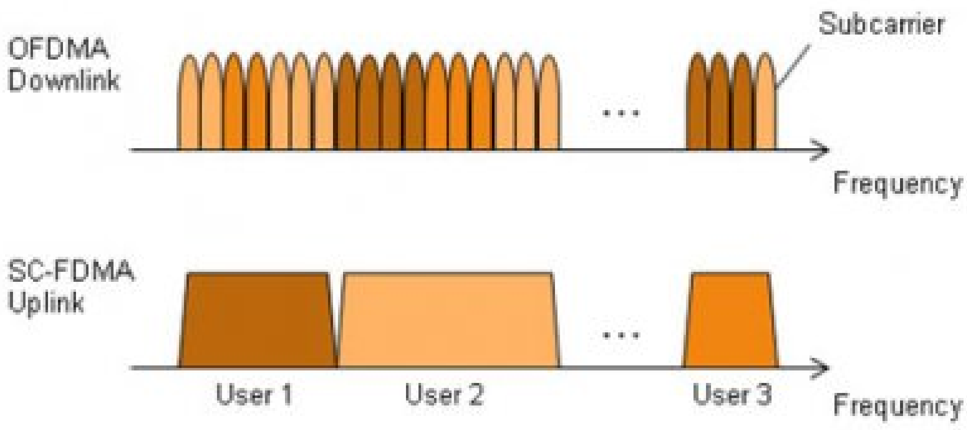

FIGURE 2.5: Subcarriers [3]

avoid retransmission in an LTE network, as transmissions are affected by many factors such as building and environment noise, etc.

2.3.9 Physical Channels and Signals

LTE has various physical channels and signals within its structure for uplink and downlink.

2.3.9.1 Downlink Physical Channels

There are six physical channels for downlink [59]. The first is PDSCH (Physical Downlink Shared Channel) which transfers higher data rates (user data) and shares the resource block allocated with other users. Modulations used are QPSK, 64QAM and 16QAM. The second channel PBCH (Physical Broadcast Channel) sends system identification from each cell at every 40ms governed by the control parameters using QPSK modulation. The third one is Physical Control Format Indicator Channel (PCFICH), which uses values one to three to indicate the OFDM symbols used for transmission of PDCCH (Physical Downlink Control Channel) using QPSK modulation. The next PDCCH, channel matches users with resources, and as indicated for the previous channel, the number of OFDM symbols for PDCCH is decided by PCFICH. The next two channels are PMCH and PHICH.

PMCH (Physical Multicast Channel) sets the types of modulations to be used and would inform users about this information. HARQ (Hybrid Automatic Repeat Request) is applied at the LTE MAC layer, an error correction system which maximizes the throughout by using other transport blocks to transport other processes while waiting for an acknowledgement. PHICH (Physical Hybrid ARQ Indicator Channel) uses acknowledgments (ACK) and negative acknowledgment (NACK) -HARQ mechanism- to ensure that the data is delivered and if not then a retransmission is requested.

2.3.9.2 Downlink Physical Signals

Two types of signals are used, the reference and synchronization [60]. The reference signal gives an indication to the receiver about the downlink channel delivery for demodulation and channel impulse response. The reference on the OFDMA subcarrier could be different for each user. There are two types of synchronization signasl: P-SS (Primary Synchronization Signal) and S-SS (Secondary Synchronization Signal), where P-SS is given 0.5ms slot timing synchronization and S-SS is given frame timing synchronization used for cell search.

2.3.9.3 Uplink Physical Channels

Compared to downlink, the uplink only has three physical channels [59]. The first channel is PUCCH (Physical Uplink Control Channel) containing the control information of the uplink channel, informs the UE about the quality of the channel, HARQ (Hybrid Automatic Repeat Request) procedures and also on scheduling. The next channel is a shared channel that allocates users in each sub frame and in the uplink many users could be allocated into different slots. This shared channel is known as PUSCH (Physical Uplink Shared Channel). The third channel is the PRACH (Physical Random Access Channel) that randomly accesses and transports information from the service provider to the UE.

2.3.9.4 Uplink Physical Signals

There is only one physical signal for the uplink, the reference signal. There are two types of reference signals, dealing with demodulation and with frequency dependent scheduling. These two-reference signals use CAZAC (Constant Amplitude Zero Autocorrelation) sequences.

2.3.10 Scheduling

In an LTE network the base station controls scheduling, ensuring quality of service for each user, manage overload situations and optimize overall throughput by reacting to changing radio conditions. The base station decides the number of resource blocks to be allocated to each user and how many users are to be scheduled. These physical channels and signals are used to carry out uplink [61] and downlink scheduling [62].

2.3.10.1 Downlink Scheduling

Downlink scheduling can be divided into dynamic and semi persistent scheduling. Dynamic Scheduling is used for bursty traffic transmission such as emails, web surfing and video streaming. If certain preferences are granted for a bearer for a particular user then the scheduler has to ensure that these preferences are met. The scheduler would give priority to this user over other users. Bearers with the same priority are not treated equally as this would result in wastage of capacity on air interface. A user at the cell edge will be allocated with a disproportional number of resource blocks due to a weak signal. The scheduler tries to provide best throughput for each user and balance overall cell capacity. First the mobile device would gather information from the PCFICH to find the number of allocated sub frame columns in which the PDCCH sends messages. Based on this information, the mobile device can calculate where search spaces are allocated and decode

PDCCH messages. Each message has a checksum that contains the mobile ID. If a downlink message is found, then resource allocation has to be carried out. There are three different types of resource allocation provided by the base station. Type 0 resource would give a bitmap of the allocated resource block. Resource blocks are allocated based on bandwidth and resource block group size. Type 1 resource allocation would also use bitmap but would not be like Type 0 allocation where resource allocation is executed based on resource block group size which results in wastage of resource blocks. In type 1 a subset is added to the existing procedure and resource blocks are allocated based on subset. Type 2-resource allocation would give a starting point with length in frequency channel. Resources can use up the entire channel. Once resource allocation is carried out the mobile device can decode the areas of the sub frame in which the data is transmitted.

2.3.10.2 Semi-Persistent Scheduling

Semi-Persistent Scheduling is used for real time applications such as voice calls [63]. A transmission pattern is defined for this type of scheduling as data is sent at regular intervals. The benefit of using semi persistent scheduling is that it reduces the assignment overhead. If no transmission is carried out then semi persistent scheduling will be switched off. For the downlink, an RRC (Radio Resource Control) message is needed to cancel semi persistent scheduling. In the uplink, a semi- persistent grant scheme is completely cancelled if no data is transmitted. The network is responsible for deciding when to use semi persistent scheduling.

2.3.10.3 Uplink Scheduling

The mobile device has to send a request to a base station to receive an assignment on the PUSCH. After connecting to PUSCH the mobile device can transmit in the uplink direction. The base station uses the buffer status report in the header of each packet to assign resources in the following sub frames while the mobile device is actively using the uplink channel. If the mobile device is not been allocated with resources on the uplink channel then it has to send bandwidth request through the PUCCH. The base station selects a coding scheme, modulation and resource blocks based on the transmission power of the mobile device. This information can be received using the power headroom reports transmitted by the mobile device periodically in the uplink direction.

The physical channel and signals [59] are required to transport data at the LTE air interface to the higher levels of the LTE protocol. The frame and sub frame structure within LTE is used to carry information between the base station and user equipment. There are two types of frame structure: Type 1 LTE FDD (Frequency division duplex) and Type 2 LTE TDD (Time Division Duplex). Type 1 has a 10ms frame structure divided into 20 individual slots. There are 10 sub frames within a frame. Type 2 consists of two 5ms half frame. Each half frame is divided into five sub frames each 1ms long. The sub frame can then be divided into three small 'special' sub frames for downlink pilot time slot (DwPTS), Guard Period (GP), Uplink Pilot tine slot (UpPTS). All three combined should not exceed 1ms. The advantage of using LTE TDD is that it allows the balancing of the uplink and downlink according to load conditions, since the traffic on the uplink and downlink can be asymmetric. As stated earlier the mobile device uses these sub frames to transmit data.

2.3.10.4 NarrowBand IoT

The Narrowband Iot (NB-IoT) standard is intended for IoT devices and services [64], to reduce power consumption of user devices and increase system capacity and spectrum efficiency. The physical layer and signals for the standard have just been finalised and will be commercialised by 2017. The main focus is to provide extended coverage in rural and deep indoor locations especially for uplink transmissions.

The key feature of the technology is the single tone and multi tone uplink transmission. Single tone transmission preserves battery life in indoor scenarios whereas multi tone tone transmission is aimed for higher data rates in good coverage areas. In [65], [66] and [67] authors discussed in detail the key theoretical features of NB-IoT. The design principles of LTE are used by the NB-IoT radio access [68]. The difference is that it uses new channels and carrier along with random access procedure to meet the increasing demands of IoT to provide extended coverage whilst operating in a narrow spectrum. As NB-IoT follows the same design principles as LTE the connection to the EPC layer enables NB-IoT devices to be supported globally in terms of connection, allowing roaming and flexible charging.

NB-IoT will be supported by 4G devices. Mobile and network security features would also be supported by NB-IoT which would be very beneficial in the long term. According to Ericsson, NB-IoT could support connectivity of billions of devices [68]. As NB-IoT will not be globally commercialized by 2017 [69] the concept could not be considered for the purpose of the work in this chapter. Therefore the focus is on LTE.

2.4 Related Work

2.4.1 Smart Home

There is a significant body of literature on the smart home concept and protocols within smart home. However, since Wi-Fi is normally discounted as requiring too much power, the focus has been on other wireless technologies. Research reported are commented and realted to the work done in this thesis.

Dongmei Yan *et al.* [70] discuss the implementation of Zigbee in smart home products. The approach and implementation detailed in this paper was at the time utilized within China's smart home industry. The proposed approach consisted of Zigbee home module connected to other home equipment to gather data. Aggregator hub unit installed within the home would be responsible to collect all the data and send it out to external network. This work looks into the concept of smart home controlled by a hub unit. Later the concept of IoT within smart homes was researched where the authors addressed power consumption in particular.

Karan Nair *et al.* [71] have discussed power consumption for IoT WSN using BLE. The paper discusses topologies followed by protocols such as Zigbee and Bluetooth and the power consumption of Zigbee and NRF against BLE is compared. The experiments carried out was based on the master/slave concept of these protocols and introduced a system in which a single slave within the master slave network is considered as a sub master node. If the master node dies then the slave node could quickly take up the responsibility of the master and carry out the relevant tasks. Based on the experiments Zigbee and NRF both have long wakeup times resulting in increased power consumption whereas BLE connects faster and the wakeup time is much shorter. The work does not consider real time monitoring.

As an extension to this, work was considered by Artem Dementyev *et al.* [72] investigating the disadvantage of using BLE in a cyclic sleep scenario. The experimental setup transmitted an 8 byte data packet at certain sleep intervals. The power consumption was compared against Zigbee and ANT. Results discussed indicate that Zigbee was more efficient in a cyclic sleep scenario situation. The results comparison was based on the reconnection time. Although BLE provided low power consumption, the time to reconnect after a cyclic sleep is much longer than Zigbee and ANT. This work considered fixed packet sizes which have an effect on results. M.D Prieto *et al.* [73] investigated the impact of variable packet size power consumption; other factors such as processor, processing times and the cost were not taken into consideration.

Rajeev Piyare *et al.* [74] discuss home automation and propose an architecture with switching functionality. The architecture is detailed but no results are given. The architecture consisted of an Arduino ethernet Shield that controlled all the units installed within the home, effectively operating as a hub to control and monitor the other units. A smartphone app is also part of the proposed architecture to monitor the units within the home.

Raja Gopalan *et al.* has discussed the data aggregation techniques in sensor networks [75]. Data is compressed at the device before transmission to the aggregator. This would allow reducing the unwanted packets transmitted to the aggregator and reduces the complexity of the duties to be carried out by the aggregator. Although this technique is effective the system still uses an aggregator to control the nodes within the home. A number of surveys have been carried out on protocols such as Zigbee and BLE for wireless sensor networks [76].

Kuor-Hsin Chang [30] discusses the suitability of BLE for IoT. According to the paper BLE supports star networks and not mesh networks; however further research is required to confirm that BLE could support mesh network. For BLE to be considerd for IoT it has to be compatible with all types of topologies for flexibility.

In [77] Kamlesh Sharma *et al.* discuss utilising an IoT system to reduce power consumption for devices or equipment used within University campuses. The idea is similar to the smart home concept in which Zigbee nodes communicate to a sink node and onto a central server. The power consumption is reduced using this technique. Although power consumption can be reduced a central sink node is still used to control other Zigbee nodes. If the work considered not using a sink node and making the Zigbee nodes capable to talk directly then the overall cost and complexity of the system would be reduced.

Research work has been carried out in smart homes particularly in relation to Zigbee and Bluetooth. Factors that developers consider when implementing these protocols on devices are the cost, power consumption and performance. Wi-Fi although is as efficient as any other protocol is not considered due to overall cost and power consumption. Although Wi-Fi related papers particularly in the IoT topic are available, these papers do not consider implementing IoT devices which are capable to communicate without the requirement of a hub and also with reduced power consumption, a key factor why Wi-Fi is discounted for IoT. Few of the papers published discussed about the concept of reaping the benefits of Wi-Fi for IoT but still did not consider the use of these devices without a hub unit.

In the paper [78], authors discussed about the concept of Wi-Fi in IoT and has mentioned about the cost and power consumption of these devices and the investments put forward for IoT devices in the future. Authors has proposed an Intergrated circuit (IC) design cloud concept by which all the IC devices could be controlled. Although the discussion about the increased cost , power consumption is discussed no practical work or simulation work has been done in the paper.

Majority of the papers in this topic area discusses about the cost and power consumption and proposes new concepts to deal with the problem but no practical or simulation work with results are provided. Few other papers in the topic area which has carried out discussion on the topic area but has not at all considered power consumption aspect of Wi-Fi or avoiding hub units are [79], [80], [81] and [82]

In this thesis Wi-Fi for smart home devices is investigated and discussed with results. Particular focus has been placed on the power consumption aspect of Wi-Fi inbuilt IoT device. Treegreen Project has helped taking this research a step further by taking the developed product to the market.

2.4.2 LTE

Research carried out by other researchers on wireless sensors mainly focuses on energy consumption and security of sensors. Very little work has been done on sensors in LTE, especially network management with the proliferation in the deployment of of IoT sensor devices to the network.

M.Asim *et al.* [83] focuses on a self-configurable architecture for wireless sensor networks. The authors proposed an architecture that efficiently reorganizes the network. The architecture relies on cluster leaders for each cell within a network who manage the clustering of the nodes. The performance of the proposed algorithm was evaluated using GTSNETS. Sensors were randomly deployed within a 150x150 square meter area and the analysis assumed that each sensor would have an initial energy of 2000mJ. Results proved that the proposed algorithm was successful in re organizing the network energy efficiently. Although the proposed algorithm was good, the simulation environment was not sufficiently good enough to make a fair comparison because all nodes were considered similar and assumed the initial energy of each sensor.

In [84] Delgado and Jaumard have proposed two scheduling algorithms: single channel scheduling (SC-SA) and multiple channel scheduling (MC-SA). A multiclass system is taken into account in this paper. For the first algorithm, one resource block is allocated to users with poor signal quality. The second allocates more than one resource block for the request placed with higher throughput requirements. Both were compared through simulation results. Results showed that MC-SA is capable in terms of performance than SC-SA. MC-SA resource allocation was much useful for video users as more than the required resource block is allocated to the user.

In [85] Adibi *et al.* discuss using sensors in mHealth applications and using the LTE network. The main issue discussed is the impact of thousands of sensors on the network. Different sensor link technologies such as Bluetooth, BLE (Bluetooth low energy), Zigbee etc., were compared and conclusions on the best were given. LTE, LTE-A and QoS affecting these devices were also discussed. The conclusion was that LTE and LTE-A are good for transmitting information from phone to sensors and vice versa.

In the thesis [86] by S.Krishnan the main topic is connecting sensors to LTE. The author has introduced a new way to connect sensors to LTE, thereby increasing the battery life of sensors. LTE and its architecture followed by the architecture of wireless sensor networks and its applications were presented. The need for connecting WSNs to LTE was stressed as being the only method of connectivity in remote areas and inhospitable environments, and therefore an efficient method is required to establish connectivity. The new scheme is introduced for periodically transmitting sensors where the sensor does not receive data. Analysis of the manner in power consumption is affected by increasing the report interval; increasing the report interval decreases the power consumption as the user equipment is in idle mode more of the time. A periodic transmission profile signalling scheme was proposed which enables the user equipment to send a message to the network when it needs to be connected, and it would be connected only when the base station has received this message. This would allow the user equipment to wake up, transmit data and then go back to sleep immediately after the procedure. The base station would allocate resources just before the user equipment would wake up and de-allocates it again after the transmission. The benefits of the scheme is that the power consumption is less, as the user equipment does not remain active after the transmissions until informed by the network. The other benefit is that it does not use any resources when the user equipment is not transmitting. Simulation was carried out using Matlab where power consumption was measured for three different types of packet size and different reporting intervals. The results proved that the proposed scheme did improve the battery life for reasonable reporting intervals.

In [87] J Brown *et al.* have developed a new resource management method, which reduces uplink latency from 6ms to 5ms and the mean uplink latency is reduced by 50 percent for certain scenarios. When a user sends a scheduling request, the base station identifies the neighboring devices that would benefit from a predictive resource allocation. Predictive resource allocation by the base station is only triggered only until threshold $x + 1$ sub frames are met and an SR (scheduling request) receipt from the first device is received. $x + 1$ sub frame means devices with an SR opportunity greater than 1 sub frame will only be considered for predictive resource allocation. An algorithm was proposed for the predictive resource allocation where each device only has one predictive resource allocation as a result of some propagating event. If the device does not have uplink data packets when the base station allocates resources then it would be unsuccessful and the base station will not attempt further resource allocation for the current event. Simulation was carried out using OPNET and results proved that latency did improve by the proposed resource allocation method.

Based on the related work in this section, it is evident that work has to be carried out specifically in the area of IoT sensors in LTE with particular focus on the performance and utilisation of the LTE network.

2.5 Conclusion

The concept of smart home is explained within this chapter with reference to related work carried out by other researchers. With reference to the increase in devices in smart home by Gartner, illustrates the importance of work carried out for this thesis. Related work has illustrated the fact that researchers work mostly on power consumption of Zigbee, Bluetooth and 433MHz and neglect Wi-Fi due to reasons such as power consumption and cost.

Along with the benefits of smart home the wireless standards of data transmission currently used within smart home is also discussed in detail within the chapter. The gap identified within the literature for Wi-Fi devices provided a focus for this research.

Along with the focus on smart home concept this chapter has also focused on providing an overview of LTE and Wi-Fi with reference to related work carried out within the field. The chapter begins with a brief introduction followed by description of various aspects of Wi-Fi (IEEE802.11) such as the architectures and the important family specifications of 802.11. Detailed description of 802.11n is provided as the thesis uses this standard for the experimental work later in Chapter 4, Chapter 5 and Chapter 6. The chapter then focus on LTE in detail with description of the elements of the network architecture and other features such as resource allocation and scheduling.

Related work has clearly identified the gaps in the state-of-the-art and explicitly points out the need to work on LTE networks that supports IoT devices. Background information provided here forms the basis of the work carried out in the following Chapters. The next chapter discusses IoT sensors in LTE followed by the design and implementation of Wi-Fi inbuilt IoT device and the power consumption of that device.

Chapter 3

IoT Sensors in LTE

3.1 Introduction

The past few years has witnessed an explosion in the availability of small, programmable, integrated wireless modules which enable small connected networks of sensors [88]. These modules have developed from Bluetooth devices with relatively simple stacks to Wi-Fi modules such as the Texas Instruments CC3000 [89], which for less than 10 dollars in low quantities includes a complete 802.11g subsystem, IP stack and web server, with only a few external capacitors and chip antenna required for a complete design. As modules become more powerful, the inclusion of an IP stack allows rapid development and enables the 'Internet of Things' concepts, where sensors are fully connected IP nodes rather than reliant on bespoke wireless protocols. Sensors are devices that detect signals and respond to input. The outputs arre gathered electronically for further processing. According to a recent Cisco white paper [15] growth of machine to machine communication in everyday lives is increasing. Figure 3.1 shows the migration of M2M devices through 2G, 3G and 4G. In areas beyond Wi-Fi coverage, it is likely that the advantages of rapid development and ubiquity are going to promote the IP route as well. Currently cheap wireless modules are restricted to 2G GPRS technology,

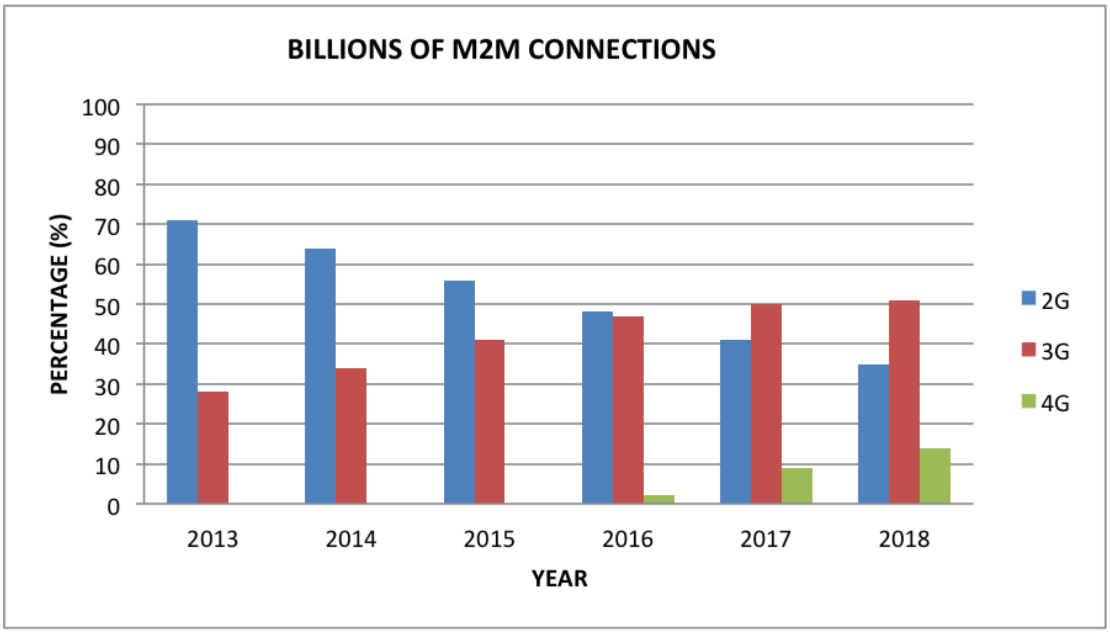

FIGURE 3.1: M2M Connections

but the integration of other wireless technologies is likely to be applied here (albeit that licensing costs form a greater part of cellular technology than WLAN). With 4G currently being deployed, it is anticipated that cellular modules for sensors will leapfrog 3G technology to implement 4G LTE, especially since LTE has a much more flexible resource management and is able to cope with large numbers of sensors more efficiently than previous technologies.

This chapter details the simulation of LTE models created based on different packet sizes generated from sensors. This gives an in-depth insight about the resource allocation in LTE network. The packet sizes used for the simulation are later compared with the packet sizes obtained through a real world sensor prototype developed and described in Chapter 5. The knowledge on IoT data in LTE is useful for the future as more and more sensor devices are deployed. An introduction followed by the security of wireless sensor networks and then the data model created for the simulation are explained in the following sections.

3.2 Wireless Sensor Network Topologies

Usually, it is not possible to spread sensor nodes evenly throughout the area to be monitored. There is varying density and it is important that a network is capable of adapting adequately if a node or a link fails. Since data reliability is essential in all environments, a hop-by-hop approach is often preferred by developers over end-to-end transmission. There are several different self-organizing, self-repairing architectures, also referred to as network topologies, that describe the general behaviour of how the nodes communicate. The most commonly used topologies are discussed below. The following figures all display the flow of information towards the sink node.

3.2.1 Sensor Network Topologies

3.2.1.1 Basic Topology

Basic, or unstructured, is the simplest form of topology. It works by having a source node that will flood, or transmit through all possible nodes (except for the one that transmitted to the source node), as the ultimate goal is for data to reach a sink node (Figure 3.2).

3.2.1.2 Cluster Topology

The cluster topology [90] is composed of three elements - sensor node, base station and a cluster head. Sensor nodes collect data and transmit to cluster heads which, in turn, send the collected data to the base station for processing (Figure 3.3).

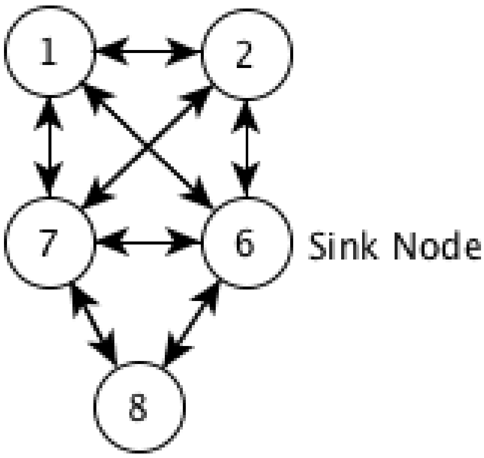

FIGURE 3.2: Basic Topology

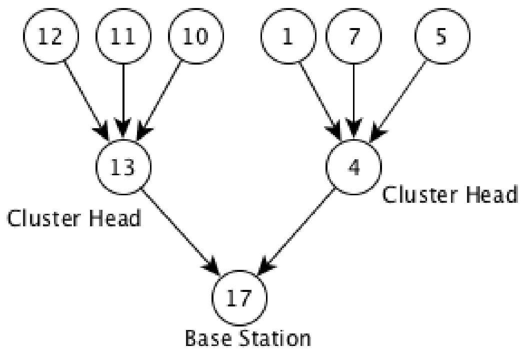

FIGURE 3.3: Cluster Topology

3.2.1.3 Chain Topology

Figure 3.4 shows a Chain topology [91]. The nodes build a path, called a chain, that data follows to reach the sink node. Transmission of information happens in successions, resulting in information delay but is nevertheless energy efficient.

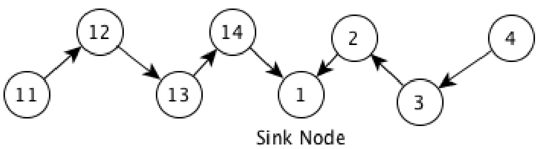

FIGURE 3.4: Chain Topology

3.2.1.4 Tree-based Topology

Figure 3.5 illustrates a Tree-based topology. The composing elements are root, parent and leaf nodes. A leaf node transmits data to its parent which, in turn, passes it to its parent and so on to the root node (which then acts as a sink). Tree topology is power efficient but there is a significant delay [92].

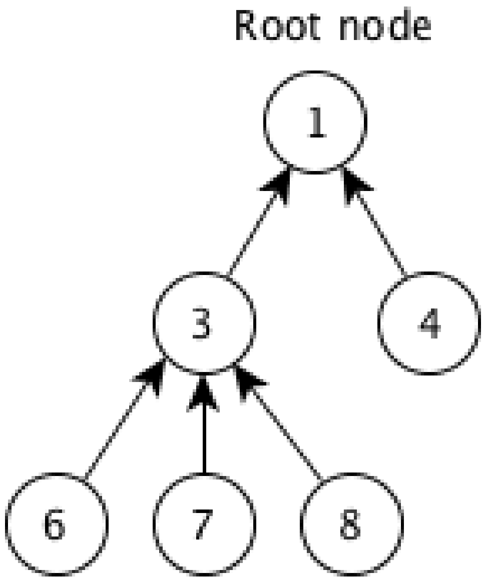

FIGURE 3.5: Tree Topology

3.2.2 Topology Considerations

In the case of a sensor prototype, like the one built for this research, there are specific factors within each topology that must be taken into consideration. The basic topology offers the best reliability because a node failing is not detrimental to the network, as multiple redundant paths are available through flooding. The clustered topology is very dependent on its cluster heads, so its overall reliability is lower than the basic structure. Chain and tree topologies are least reliable as both rely on other nodes. The prototype created in Chapter 5 does not require a sink node or a neighbour node to form a communication chain (Figure 3.8). The downside of having to depend on other nodes or a sink node is the fact that if one of the node dies the whole topology is impacted. The prototype created differentiated by communicating directly to the network which makes it more reliable for future use. The next section discusses the different security attacks towards wireless sensor networks.

3.2.3 Security of wireless sensor networks

Attacks [93] found within sensor networks are described below:

- Denial of Service

  As the name suggests, this type of attack would deny the service requested by the sender nodes [94]. The service is restricted by hackers sending large amount of false information to the targeted network and thereby using up all the allocated resources thus preventing others from using the service. If all the resources are used up then the network is flooded. Figure 3.6 shows an attack flooding the network to overload resources.

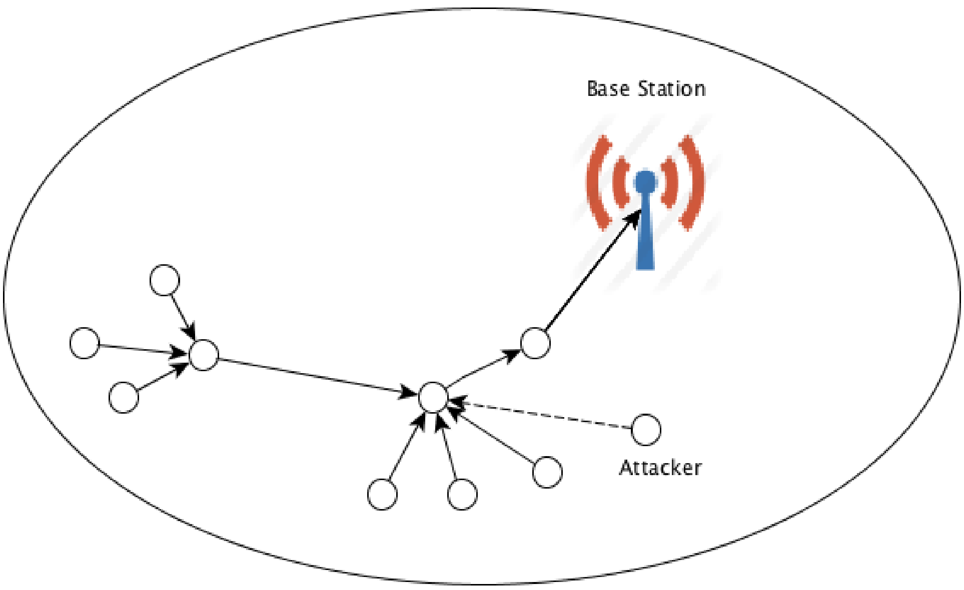

FIGURE 3.6: Denial of Service Attack

- Selective Forwarding

  If the attacker receives the packet to be transmitted to the destination the chances of tampering with the data or extracting the information and sending it with just the routing information would prevent it from reaching the end user [95]. This would result in a lost packet for the sender and therefore the network has to re-request for retransmissions which would consume both time and resources, resulting in further delay for other nodes within the network. If the packet lost contains critical data then the confidentiality of this data has to be protected in order to prevent the attacker from intercepting important information.

- Sybil

  A Sybil attack [96] originates from a single node which uses multiple identities to confuse the network and delay the performance of the network. For example, if five sensor nodes within a group are trying to transmit to the cluster head, another five more of those identical fake nodes would appear within the group trying to transmit and thereby compromise the network.

- Replay

  Here attackers would replay the same information without compromising the integrity of the received packets, making the receiver believe that it is an original packet [97]. By doing this, they could gather important information from the packet. This could be prevented by using proper authentication of packets and discovering from the network whether the packets had been sent by a third party sources.

- Sink Hole

  Sink hole attacks [93] are carried out by the attackers to prevent the base station from receiving the correct data. These sink node would act as a compromised node within the sensor group and provide information because they are closer to the base station and can transmit data. After receiving the data from neighbouring nodes, this data will be either used by the attacker or used to attack the network. Figure 3.7 demonstrates a sink hole attack.

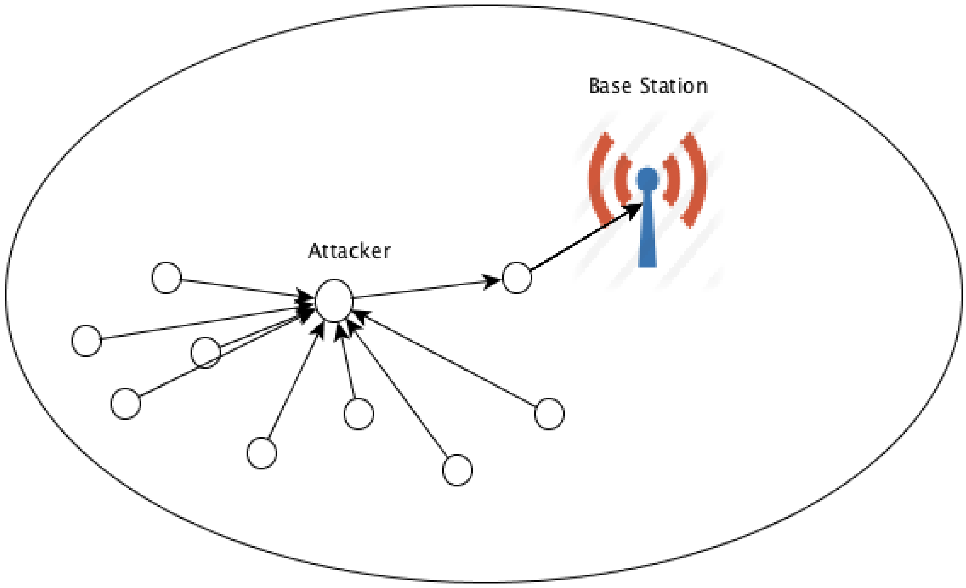

FIGURE 3.7: SinkHole Attack

Security attacks have to be considered when sensor networks are implemented to reduce the risk of attackers hacking these nodes to gain access to the network. The security aspects discussed here have been taken into consideration and has been a motivation to design a node capable of direct connection using LTE. Another reason for choosing LTE connectivity for the nodes is due to the fact that LTE would mitigate the security attacks compared to Wi-Fi. The Subscriber Identity Module (SIM) cards for each device within the LTE network ensures unique identity for each node ensuring additional security through its authentication techniques. The location identification provided through its SIM cards is an additional security feature which adds onto the central management system provided by LTE to identify and protect the node from security attacks. Wi-Fi which does not have a central management system therefore has a higher risk of security attacks compared to an LTE network. The node developed therefore does not require a sink node for connection and so it obviates some of the security issues detailed. The next section discusses the simulations carried out based on wireless sensor networks. Resource allocation is investigated in detail along with the sensor node data transmission.

3.3 Simulation Scenario

This section provides a brief overview of the simulation scenario which is discussed in detail in Section 3.5. The parameters chosen for the simulation is based on a paper [98] which is also referred to and discussed in this section.

An uncoordinated number of sensor nodes are used to connect to the LTE and push data across the network. In the scenarios created for the simulation, sensor nodes are transmitting data across the network.

LTE allocates resources to many users within the network. Utilising the parameters in [98] the number of sensor nodes that can transmit the same amount of data with 55 resource blocks was determined. The simulation set up is as close to real world situations as possible, where sensor nodes would send feedback to the base station. Figure 3.8 shows the scenario for sensor nodes communicating with a base station.

As in [98] the network is allocated 10 MHz bandwidth, 55 resource blocks and 180kHz for each resource block. The simulation is based upon following parameters:

$$\text{CQI} \leftarrow 14$$

$$\text{kHz} \leftarrow 180$$

$$\text{Data} \leftarrow 10\text{Kb}$$

$$\text{Bandwidth} \leftarrow 10\text{MHz}$$

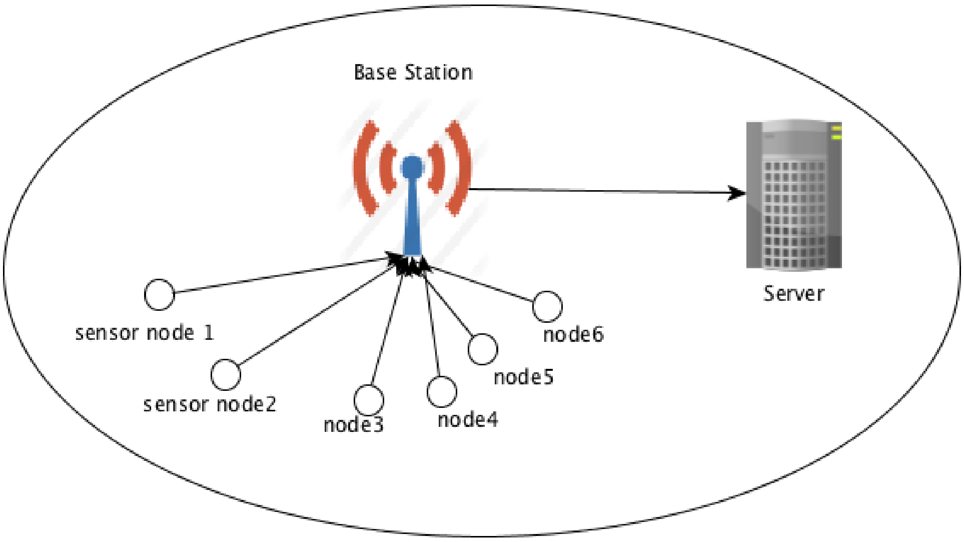

FIGURE 3.8: LTE Sensor Scenario

The assumption is that all 10 Kb of data was allowed to be transmitted by the sensor nodes at once. The analysis assumes that 181 bytes are transferred per Resource Block (RB) as there are only 55 RBs. The simulation carried out in the next section would indicate if it is possible to transmit a total of 10Kb of data with 181 bytes per RB.

3.4 Data Model

The analysis assumes a constant data rate of 10 Kb/s and that the sensor nodes transmit data intermittently. Sensors are likely to transmit very small amounts of data. For example, a temperature sensor may only have to transmit a single temperature value, which could be encoded as a single byte.

Following the examples in [99], and assuming a Spark.IO device is a typical internet connected sensor, the analysis indicated that a minimal set of headers for the HTTP request is in the order of 100 characters, and a minimal body is about 50 characters. However, a more normal (and secure) 'small' packet is in the order of

250 bytes, Section 4.6, and with large packets reporting a number of measurements approaching 1000 bytes. Therefore these values are used in the simulations.

3.5 Simulation

In order to model LTE, SimuLTE [100] was used as an extension for Omnet [101]. In papers [102], [103] the software Omnet was used for simulation purposes of both wired and wireless networks. All models created in Omnet could be combined and reused anytime. It also offers extensive data collection and graphical presentation. SimuLTE models LTE down to the physical layer, including the PDCP, RLC and MAC, status and channel reporting, etc. Physical transmission at a symbol level over OFDM is abstracted [101], and the software implements a channel model which allows us the prediction of whether the packets will be corrupted and require retransmission.

The scenario created was a simple single cell with sensors distributed over a 300m by 300m square area. Since the focus is on the relative requirement for resource blocks, retransmissions were not considered at the application layer, if a packet timed out at the RLC it was considered dropped. Sensors transmitted at random intervals over the course of a second, and all transmissions were completely within that time. Two configurations were considered, sensors transmitting 250 bytes and 1000 bytes. The simulation was run several times to gather results with a 95 percent confidence interval.

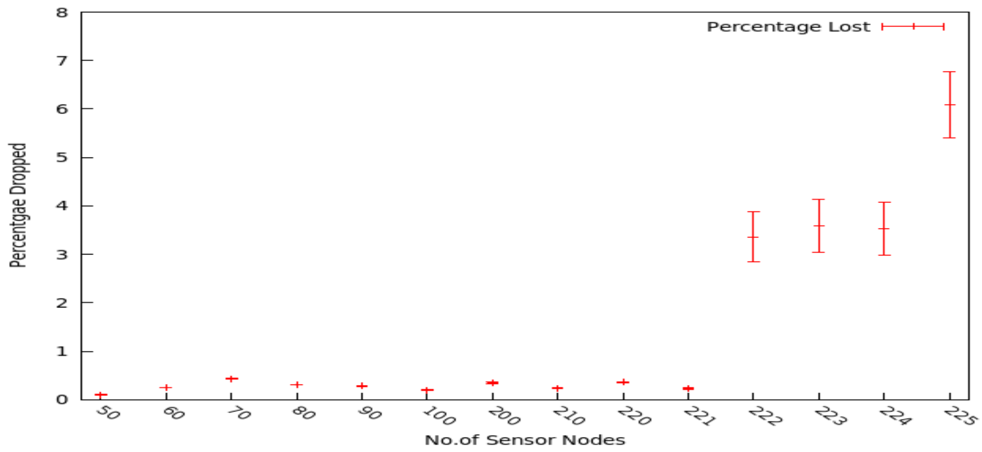

FIGURE 3.9: Percentage dropped packets for nodes transmitting at 300m (250 bytes each)

Figure 3.9 represents the percentage of dropped packets obtained for transmission of data by the sensor nodes at 300m. Each node transmitted 250 bytes. The bandwidth allocated for the system is 10MHz. From Figure 3.9 it can be noted that 221 IoT sensor nodes could transmit with near zero percent delay. The packet lost rate has increased after 221 node transmission. Figure 3.10 is for a distance of 800m to test a different scenario.

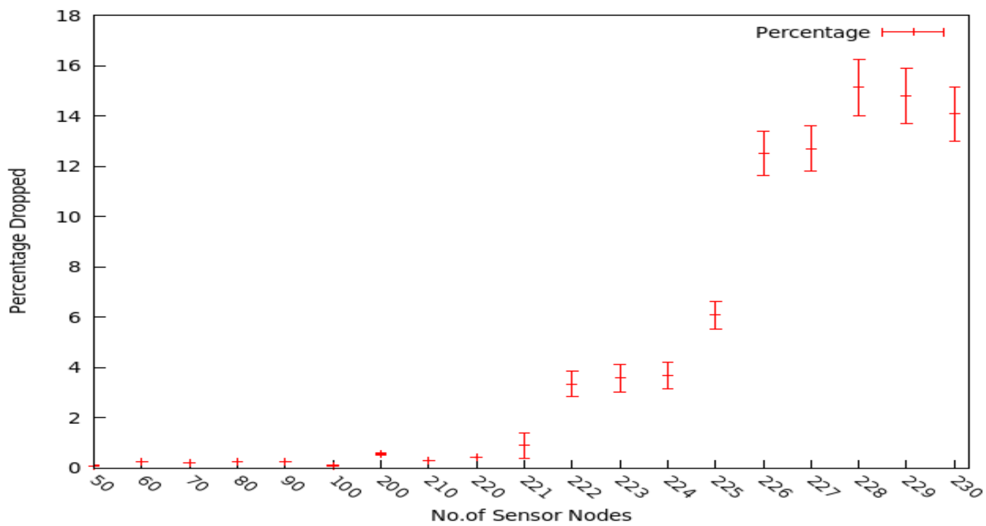

FIGURE 3.10: Percentage dropped packets for nodes transmitting at 800m (250 bytes each)

As expected, in Figure 3.10 it can be seen that there is an increase in propagation delay which resulted in loss of packets for 221 nodes whereas for 300m transmission 221 nodes transmitted at no packet loss. Figure 3.12 and Figure 3.13 show results for nodes transmitting 1000 bytes at 300m and 800m respectively. Figure 3.11 is the same results as Figure 3.12 but with smaller scale for ease of understanding of results.

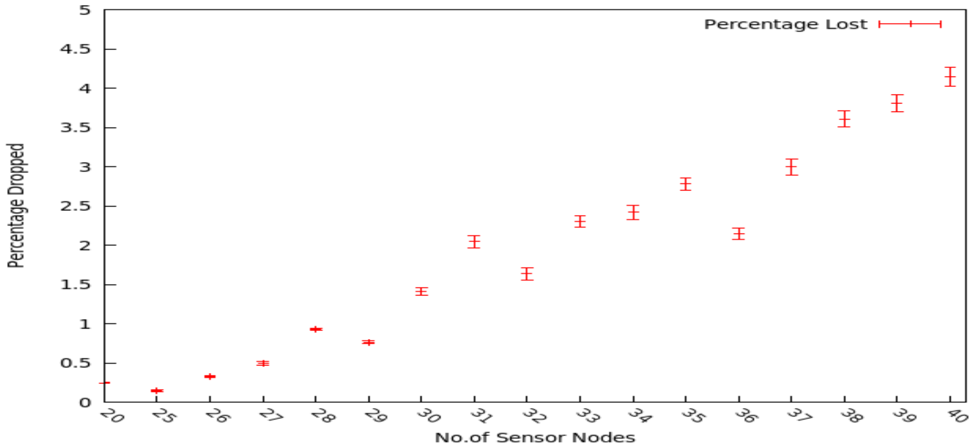

FIGURE 3.11: Percentage Dropped for nodes transmitting at 300m (1000 bytes each)

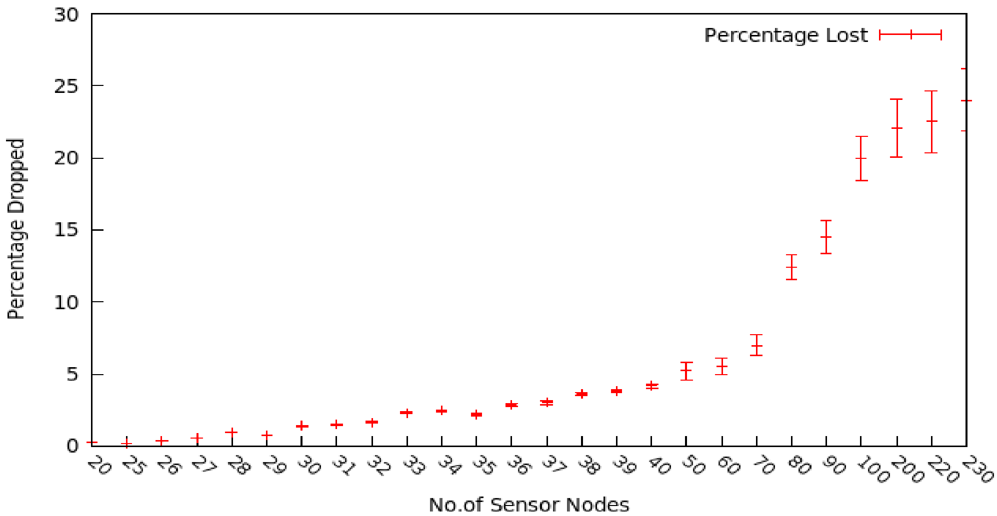

FIGURE 3.12: Percentage dropped packets for nodes transmitting at 300m (1000 bytes each)

The percentage drop is presented with an upper limit and lower limit based on a 95 percent confidence interval. As the number of sensor nodes decreases there is a significant drop in percentage of lost packets. 29 sensor nodes transmitted 2302 bytes with less than one percent packet drop. Figure 3.9 and Figure 3.10 shows

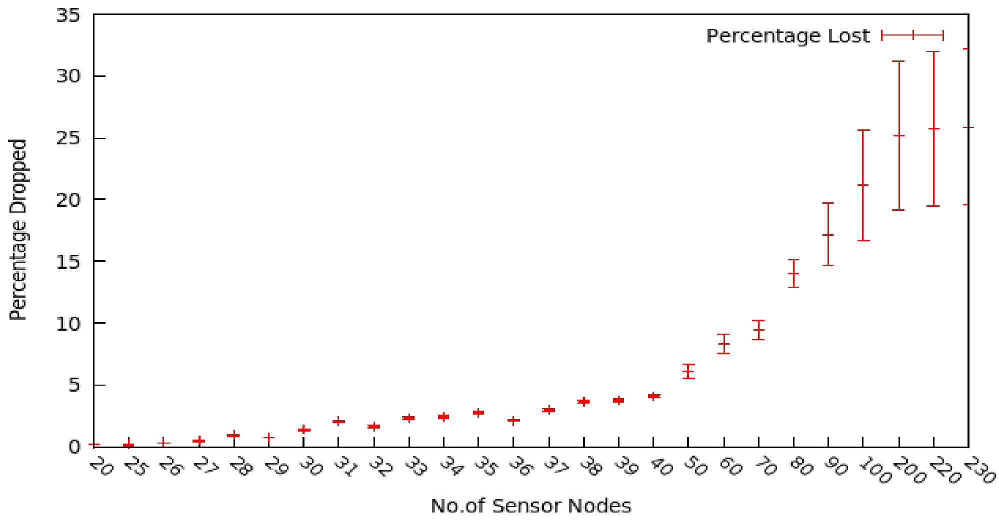

FIGURE 3.13: Percentage dropped packets for nodes transmitting at 800m (1000 bytes each)

the results obtained for sensors transmitting 250 bytes. 100 sensors transmitted with less than 1 percent packet loss from Figure 3.9 and 3.10.

Based on Section 3.3, 10Kb will be transmitted with the allocated 55 resource blocks in which each resource block would send 181 bytes per resource block. However, when transmitting to individual users discontinuously in full simulation with resource allocation, etc, less than 29 users transmitting 1000 bytes can be supported in each second. This corresponds to a usable rate of 21 bytes per resource block, a significant reduction, for a 1 percent packet drop.

These results assume that the sensors transmit infrequently such that resources are not allocated between sessions, likely if the sensors are transmitting only a few times each hour or less.

The LTE simulation above was carried out based on the assumption that a small packet would be in the range of 250 bytes and large packet in the range of 1000 bytes. The results obtained provides information on the number of sensor nodes that could transmit both types of packets and the resource allocation of the network to these nodes.

3.6 Prototype Information

This section focuses on the packet sizes of a sensor node in a real world scenario. For the LTE simulations in the previous section packet sizes were assumed in the range of 250 for small and 1000 for large packets. This section looks into the packets sizes of sensor node at the server to validate the assumption of packet sizes used in LTE simulation.

The data gathered at the server for the prototype is recorded to derive a conclusion of the packet size of an actual sensor node in the real world. Using the developed temperature sensor prototype, the data pushed to and received at the server was recorded. Wireshark was used to capture the traces at the server end. The packet sizes of the node would be larger by the time it goes through the seven layers of the OSI model. The final packet size of such a sensor node at the server would be higher than the seven bytes of data transmitted at the application layer. With the addition of overheads from each layer of the OSI model would result in the increase of the packet size which was seven bytes when starting off from the application layer. Based on the results from Wireshark the total length of packets of a temperature sensor node at the server is 189 bytes which means 182 additional bytes has been added onto the initial seven bytes of data.

Considering a home with multiple devices generating and transmitting data packets would explain the complexity at the server. If five devices are installed in a home then a total of 945 bytes of data is sent to the server aligned with the LTE sensor data models. The average packet size for a data transmission is taken as 250 bytes for small packets and 1000 bytes for bigger packets. Thus 189 bytes falls in the range of ‘small’ packets and therefore the performance of transmitting packets using LTE has been validated by the prototype data.

3.7 Conclusion

LTE was investigated to determine its ability to support the transmission of data from sensor nodes in terms of resource allocation. The packet size chosen based on current trends in IoT applications being deployed. Existing work on mobile users in an LTE network has calculated theoretical numbers of RBs needed to carry 10Kb of data continuously, but this significantly overestimates the practical performance when sensors have to register with the system, request resources, and then transmit. The total resource blocks used for transmission are shown to be much higher than would be suggested by theory. Simulation was carried out using SimuLTE in OMNET using real world parameters for sensor networks in an LTE network. Results show that a maximum of 1 Kb of data can be transmitted by 29 sensor nodes each second with less than 1 percent of packet loss which equates to 21 bytes of data being transmitted per RB, compared to 181 bytes per resource block for continuous transmission.

The packet size used for the simulation was within the range of the packet size obtained through experimental sensor prototype transmission. This validates the results gathered and discussed throughout this chapter.

Based on the results, LTE in its standard form is not viable for sensor networks transmitting small amounts of data. LTE as such is very efficient for mobile users transmitting large amounts of data but when considering it for small sensor nodes transmitting small packets a better option is required. NB-IoT which has just been standardised might be able to address these issues. At the time of the work that standard was not reported. The conclusion based on the results from this chapter has been a motivation to consider an alternative option, Wi-Fi, for IoT sensor node communication as LTE is inefficent for sensor networks. The next chapter discusses using Wi-Fi for IoT and its design and implementation.

Chapter 4

Design and Implementation of the Prototype

4.1 Introduction

With the introduction of devices for multiple purposes within a home the need to understand more about the device is crucial.

Many smart home products available in the market are capable to use Zigbee and BLE for communication. The capability of the device to utilize Wi-Fi has not yet been researched mainly due to significant power consumption and cost. The need to develop a device which is cost effective and consumes less power and could use all the benefits of Wi-Fi is crucial. Therefore one of the major contributions presented through this work is the development of a prototype and investigating the smart home device power consumption and optimization. The need for including Wi-Fi capability for these IoT devices as discussed within Chapter 2 is substantial. The work in this chapter therefore considers including Wi-Fi integrated within the prototype.

This chapter addresses the architecture and implementation of the prototype designed and built for the purpose of this thesis. Developing a prototype was useful to have a full control of the device and to alter the coding aspects to derive an enhanced and efficient smart home unit. The prototype developed was later taken forward for further testing to develop it in a large scale. As part of developing a much more efficient devices for future, the entire prototype was developed during the time of research by putting together low cost but efficient components together. All the coding aspects, implementation and challenges throughout the development of the initial design through to the final design is explained with figures throughout this chapter.

4.2 Architecture

The beginning stage of the design was to choose the components needed to build the device. The challenge faced while choosing the components is to keep it simple, in terms of efficiency and cost being essential as proliferation of devices expands to vast numbers in the future. The capabilities expected from the prototype were: it should be able to have both short range and long range connectivity and should be wireless with no infrastructure required.

With short range the prototype will be capable of being used inside houses connecting via ADSL. This would let the prototype function as a smart home device. In a wider context where the prototype has to be placed outside the home or building it should have the capability to connect to the network via a base station, i.e. the LTE network. Existing IoT devices do not have multi functional capabilities and therefore can only be used for a single purpose. The next question to be addressed, is choosing a communication medium to meet all the expected requirements mentioned above. Devices used in smart homes use 433 or BLE in-built within the device. This medium of communication is good only if this device is used for short range purposes. Above all, these require an infrastructure for proper functionality.

The most suitable option for the prototype to be built is Wi-Fi. This choice is based on the capability Wi-Fi provides such as: short and long range capability, and the ability to use existing infrastructure. Wi-Fi being ubiquitous and available everywhere this is the best option for the prototype. Using this existing infrastructure would provide the user with less complexity in terms of connecting the device.

The type of sensor used for the device is a temperature sensor, commonly used in smart homes to control appliances such as heating. Figure 4.1 illustrates the design phase.

The components used for the prototype are

- Temperature sensor (DS18S20)
- Light Emitting Diode (LED)
- low cost Wi-Fi chip
- FTDI for serial communication (Programming)
- Arduino IDE

4.2.1 Arduino

Arduino Boards are used for development of most of the smart home devices. As it is an open source platform developers prefer to use it. A normal Arduino board cannot be used to develop the smart home device as it has to be Wi-Fi capable.

FIGURE 4.1: Design Phase

An additional chip has to be added onto it to make it Wi-Fi capable. In terms of cost, the single Wi-Fi chip opted for this thesis would do the entire job. This saves money on an additional Arduino Board. In future, a chip like that used in this thesis would be more useful to reduce the cost of the overall device for the consumer. Figure 4.2 illustrates the work carried out using the board. The

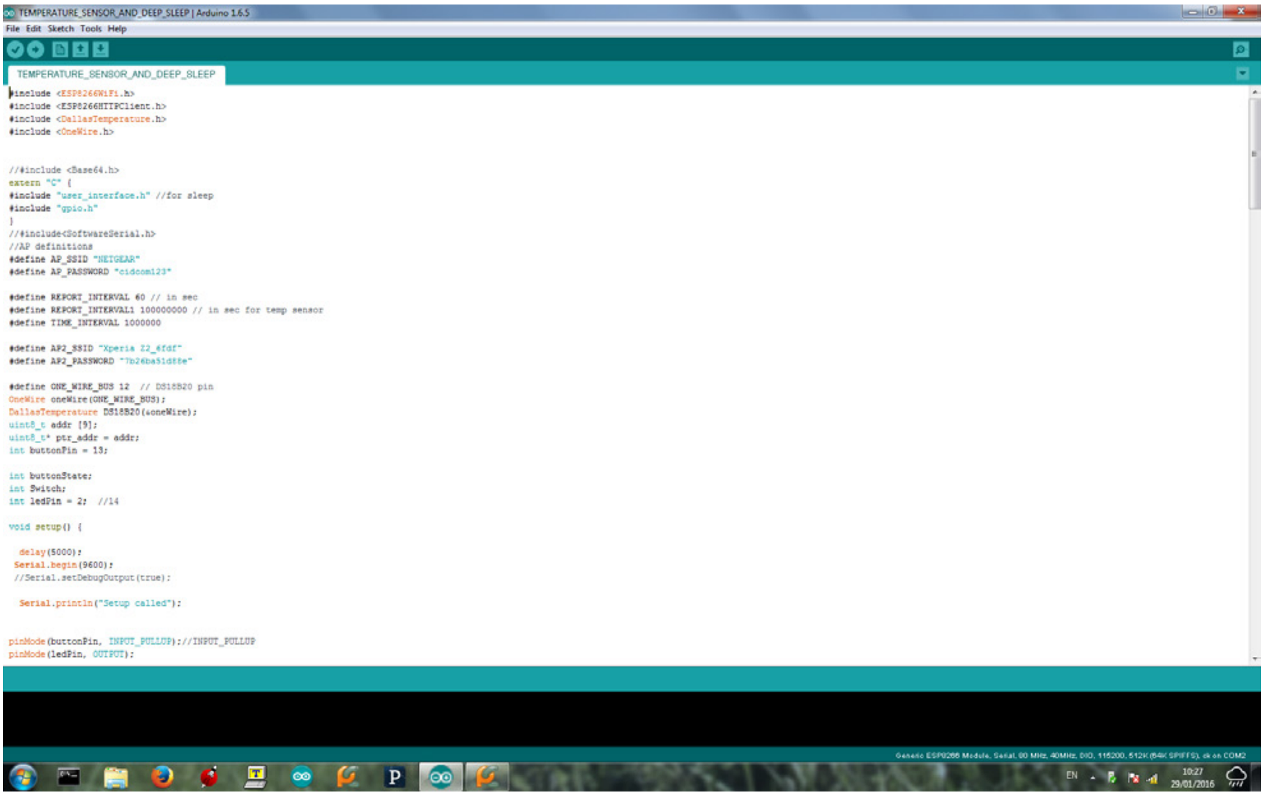

FIGURE 4.2: ESP Code in Arduino IDE

Arduino IDE was used to push the code to the Wi-Fi chip. The programming language used for coding the device is the one supported by the Arduino IDE which is C++. The code is programmed such that the prototype would connect to the router and transmit the data to the server. Wi-Fi chip enables the digital output from the temperature sensor to be transmitted across to the server via the router.

4.2.2 Temperature Sensor DS18S20

Each temperature sensor [104] device has its own unique code and its stored in the 64 bit ROM (Read-Only Memory) of the device. The memory (scratchpad) of the device would hold the 2 bytes of digital output data and would retain the data by providing access to two of its access registers. The scratchpad memory consists of a CRC (Cyclic Redundancy Check) which would verify if the data received is error free. The scratchpad CRC would calculate a value based on the data within the scratchpad and would compare it with the read CRC. If both the values would match then the data received has no error within it. Figure 4.3 illustrates the concept of temperature sensor communication.

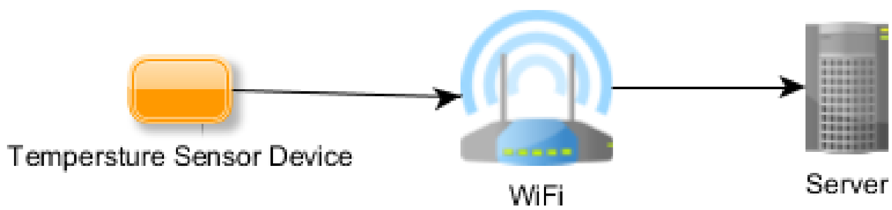

FIGURE 4.3: Temperature Sensor Device Communication

4.2.3 Wi-Fi Chip

The chip with inbuilt Wi-Fi properties used in the prototype is ESP8266, precisely the ESP03. The cost of the chip is low and has all the advantages needed to convert it into a Wi-Fi device by suitable coding pushed onto the chip. The pin layout of the chip is provided in Table 4.1:

Points to be taken into consideration for this chip is that some of the pins within the

TABLE 4.1: Chip Pin Connections

ESP03 PIN	Connections
CHPD	HIGH
URXD	Receiving Pin for programming
UTXD	Transmitting pin for programming
GND	GND
GPIO0	LOW for programming
GPIO15	HIGH
GPIO16	I/O
GPIO2	I/O
GPIO14	I/O
GPIO13	I/O
GPIO12	I/O

chip has to be high and others need to be grounded all the time. The connections of this chip is provided in Table 4.1. The baudrate, the transfer rate of the data to the communication channel, of this chip works at is 115200, anything lower would result in the chip delivering non readable output.

4.3 Implementation

The first process of any hardware development is to verify operation. The Wi-Fi chip was tested to confirm responsivity on AT commands. A telnet terminal was used to push AT commands, instructions used for testing purposes, onto the ESP chip. If the chip responds with an OK message for the AT commands this confirmed that the chip is ready to be tested. To use this chip within an Arduino IDE a library is required, currently available as open source. The header file of the library should be included to get the code opeartional on the Wi-Fi chip. The baudrate of the chip is 115200, the Bus Pirate was used to check the baudrate of the chip. The ESP chip used for the prototype design, was then tested with a basic LED test.

A single line of code was written to activate the LED, based on the chip being high or low. the chip had to be connected to the sensor device to obtain the output and provide information on bytes transferred over to the server. The VCC and ground were tied with the chip and the data pin was connected to GPIO 12 with a resistor to pull up the voltage. The breakdown of connection of the chip is provided in Table 4.2.

Table 4.2: Chip Pin Connections

FTDI Pins	ESP03 Pins	Sensor Pins
3V3	VCC	VCC
GND	GND	GND
TXD	UTXD	-
RXT	URXD	-
-	CHPD (always HIGH)	-
-	GPIO15 (always grounded)	-
-	GPIO12	DATA PIN

With the temperature sensor connected to the chip several issues had to be mastered. Insufficient power going into the chip resulted in no output which was corrected by the addition of a resistor between the data pin and VCC to pull it high. A Voltmeter was used to determine if sufficient power was applied to the sensor. If the Wi-Fi chip does not recognize the sensor then the code should be written such that it would return a 'No more addresses' each time when the sensor is not identified. Connecting the sensor to a wrong input pin result in the chip not recognising the sensor. An alteration to the code with the correct sensor pin resulted in fixing this problem. At the beginning phase of implementing the sleep function within the prototype, the addition of sleep function code onto the existing code did not function as expected. The chip worked as per mentioned in the code except the sleep functionality needed. Light sleep function is not supported by the firmware and had to choose deep sleep function which could be done by soldering together two jumper pins on the Wi-Fi chip. This would ensure the chip would wake up based on the sleep function written within the code. The chip resets itself by default. Figure 4.4 illustrates the initial design process of the device.

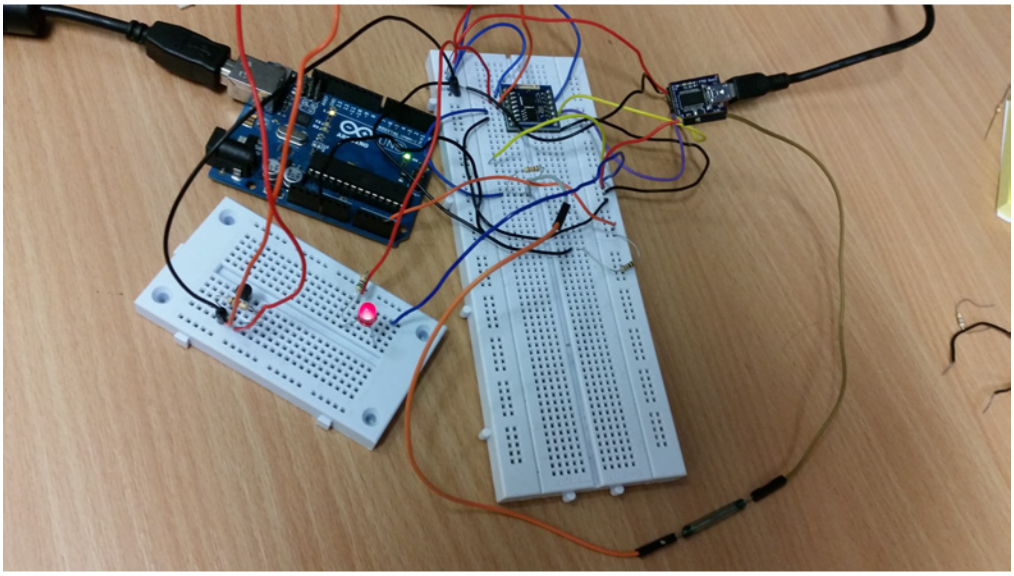

FIGURE 4.4: Initial Design

4.3.1 Server

The second stage of the work is to transfer the reading recorded over to the server via a router. A simple python sever was created to receive the data sent by the sensor via the router-NETGEAR. At the beginning of testing the data transmission from the prototype to the server a simple string message 'Hello' was send across. After the successful transmission of the simple string message the prototype was tested to send the temperature reading to the server. To ensure the server is reliable enough the prototype was left overnight sending data to the server. The server did not receive any reading during the day as it went into auto sleep mode after a certain period of time. The server settings had to be changed to not go into auto sleep mode. Figure 4.5 illustrates the temperature and the MAC address of the chip recorded at the server end. The benefit of including the MAC address of the device along with the data send to the server helps identifying the device. With many devices installed, a reading sent in without its address would result in misinterpreting the device from which the data is received. This is a much more secure way of gathering data. A pre-installed device sending data to the server without its address would not necessarily be from the device. It could be from an unreliable source trying to hack into the device.IP Number of the server along

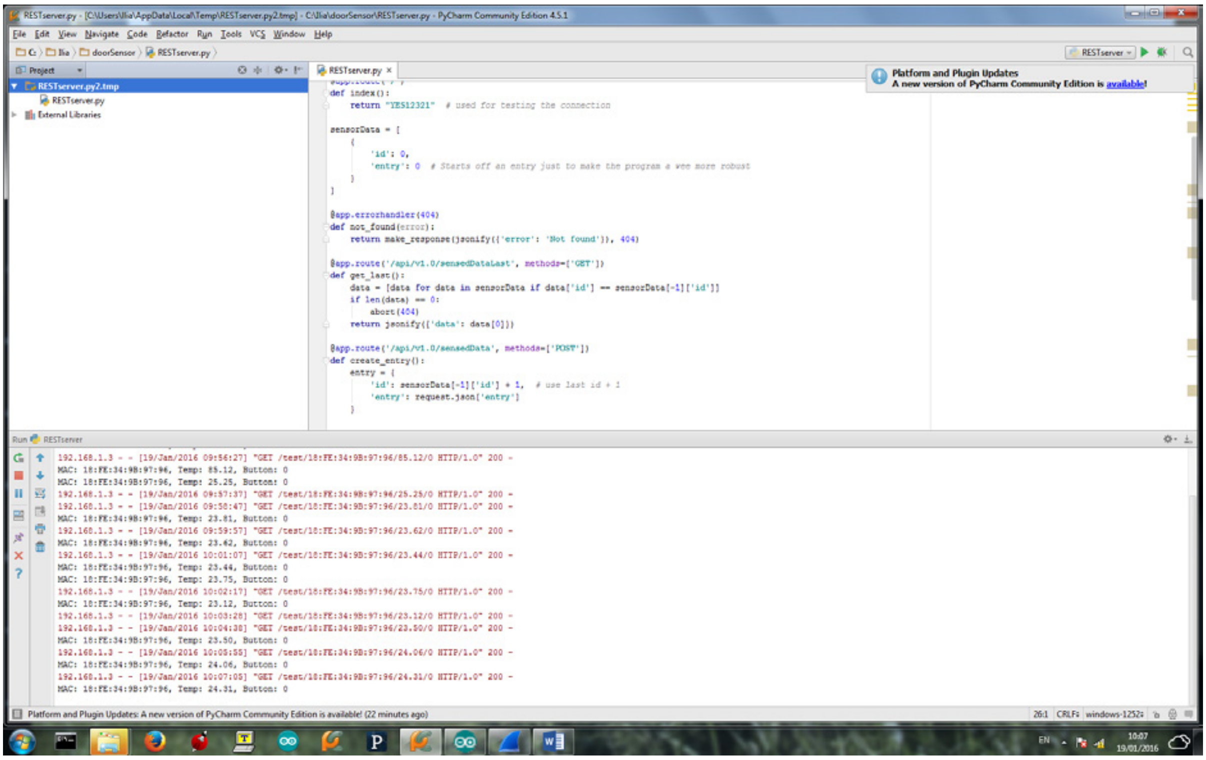

FIGURE 4.5: Server

with the SSID and password has to be connected to deliver the message correctly to the desired server. The code is written such that if the chip could not connect to the server it would send back a message 'connection to server failed'.

4.4 Final Design Process

The final operation to be tested was reliability as it is a key factor for any prototype developed. Reliability tests were carried out to ensure the prototype would work without any delays or crashes. The test was carried out by letting the prototype to run for several days and the server was continually checked to ensure data has been received. The initial prototype design was implemented using a breadboard. After the stage of finalizing that this prototype would work a PCB was created. The schematics and final prototype is provided in Appendix A. Eagle software was used to create the schematic and the board for it.

A delay in receiving the final PCB from production resulted in a delay in gathering results. When developing prototypes the potential delays to receive the final product has to be taken into account if the products are to be made available to the user. The developed code has to be pushed onto the final design to be able to use it as developed in the initial design. The schematics and the board layout - top and bottom - are provided in the Appendix A (Figure A.1, Figure A.2 and Figure A.3)

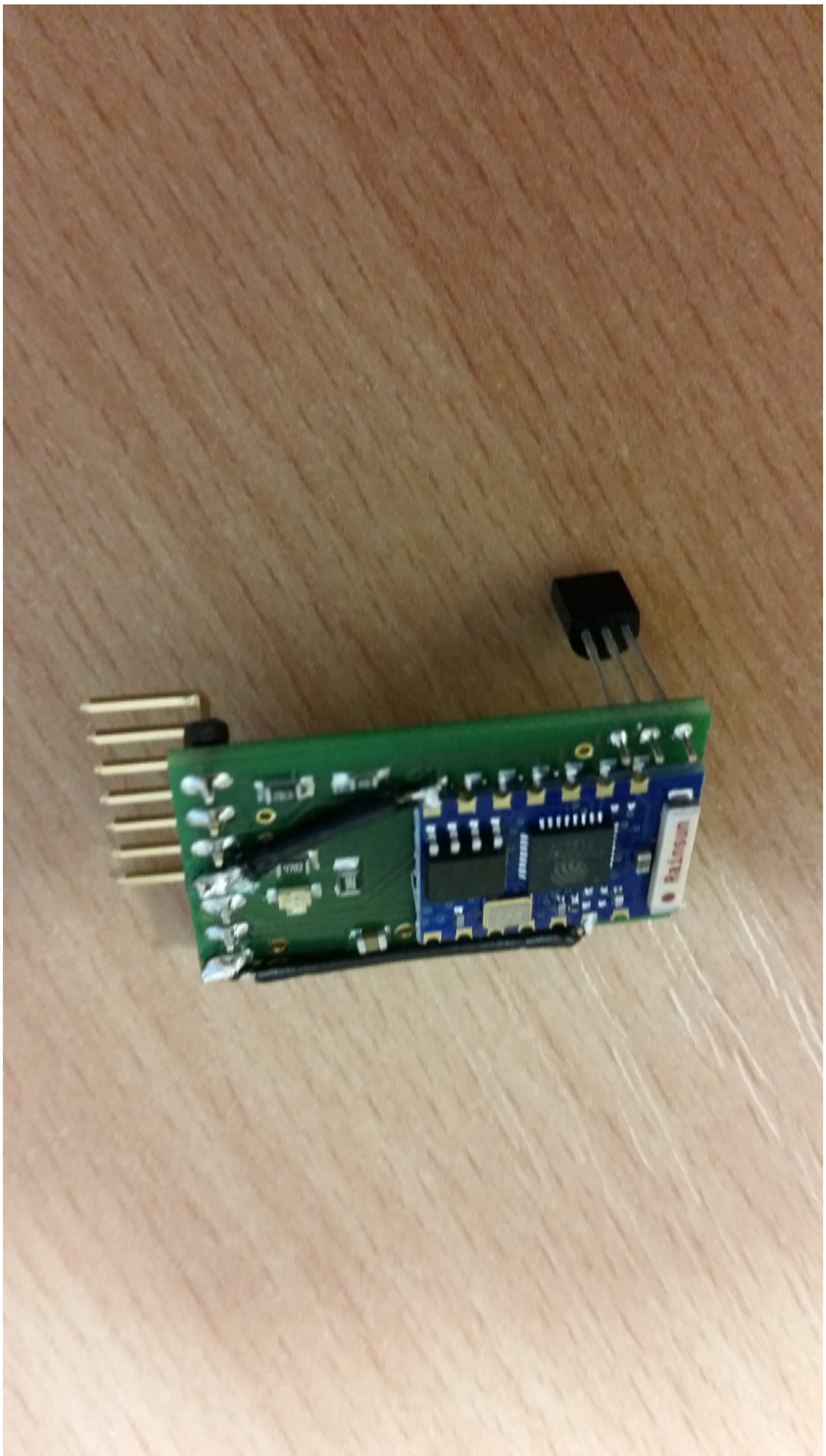

FIGURE 4.6: Version 1 of the Prototype

The prototype when received had to be modified to include a few other aspects which improved the entire design. The coin cell battery was removed for version two as it would not provide sufficient power to yield it working for at-least a year based on the results from Chapter 6. The additional jumper wires which had to be included in version 1, Figure 4.6, of the board was removed for the second version. GPIO0 was not grounded in version 1 and so jumper wires were added for test purposes. An LM1117 3.3V regulator was added in version 2 which would take inputs of up to 12V, so the circuit will work with either a 3V battery -without the regulator- or with an input up to a 12V supply. A mini USB regulator was also added for convenience of powering it via a USB for testing.

Version two, Figure 4.7, of this prototype was developed which included all the points mentioned above. The schematics and board design are included in the Appendix A (Figure A.4, and Figure A.5).

Version three of the prototype was developed with additional features to reduce power consumption, explained in detail in Chapter 7.

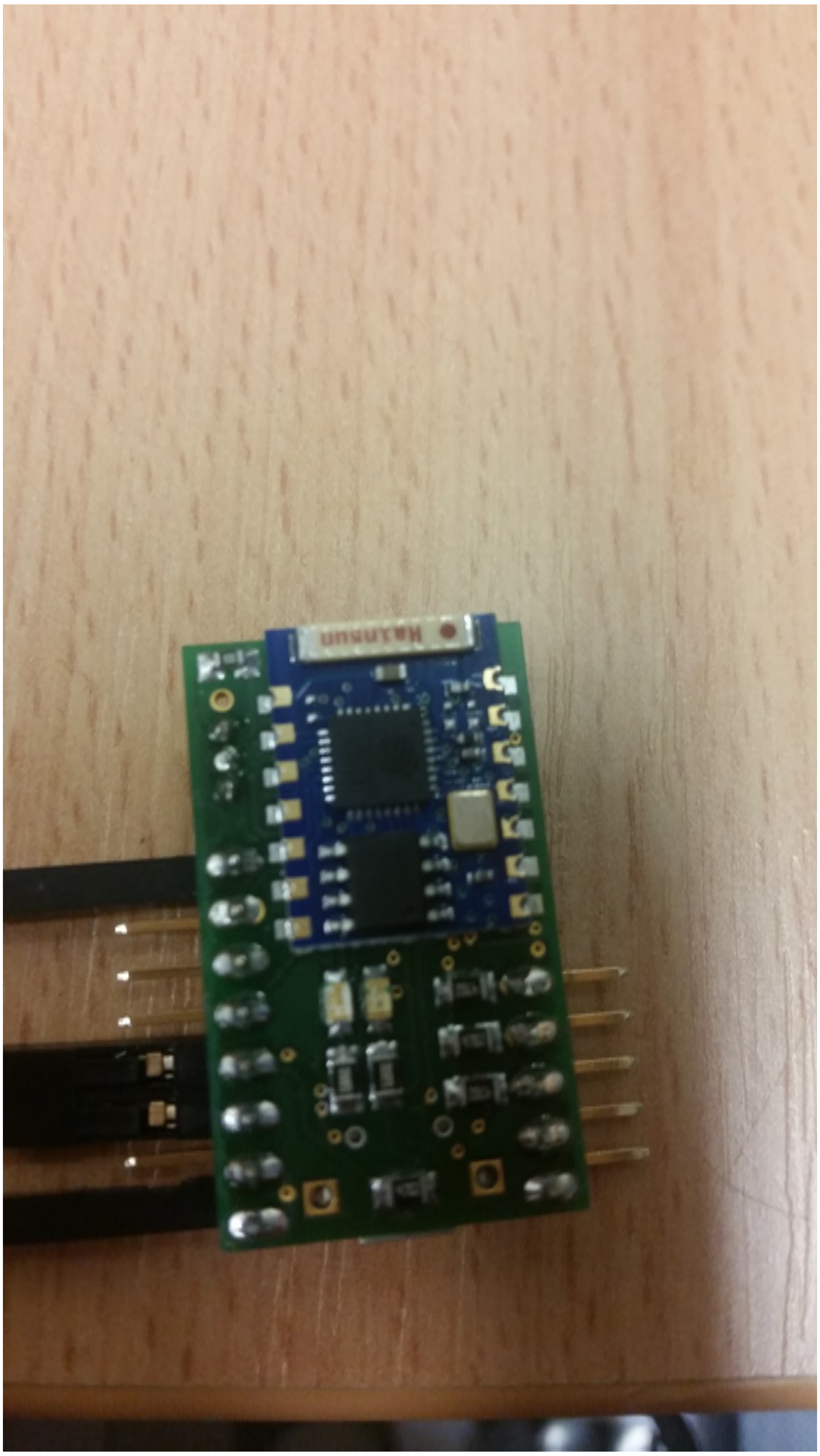

FIGURE 4.7: Version 2 of the Prototype

4.5 Conclusion

In this chapter a temperature sensor prototype is proposed and developed at low cost. The prototype is intended as a model device to be taken forward for use in smart homes.With this knowledge of the technical and coding aspects of the devices used within the prototype and taking into consideration the issues mentioned during the process of the set-up, this will help ease the reproduction of this device. The issues overcome during the implementation will aide further developments. The coding for the set-up can be easily customized to fit to specific requirements.

In addition, all the other aspects related to the prototype such as power consumption are discussed in Chapter 5. The following Chapter 6 discusses the optimization of power consumption of the developed prototype with results.

Chapter 5

Power Consumption of Wi-Fi Inbuilt IoT Device, ESP03

5.1 Introduction

The ever decreasing cost of silicon powered devices has made it possible to have tiny computers monitor ever increasing aspects of our lives, this can be seen with such devices as Fitbit. This is a tiny computer positioned in a wrist band that has the sole purpose of helping us monitor our fitness. This could not have even been imagined 20 years ago, due to the cost and size of computers [105]. This convenience of easily accessible data is becoming more prominent with ever more sensors detecting our movement, vitals & habits. The issue which is currently being faced is that transmitting this sensor data back to a platform [4] that the user can easily access consumes significant power, and therefore most of these sensing devices have to be hardwired for power.

Two important areas of consideration for IoT devices are energy efficiency and cost. The benefit of using Wi-Fi is that it can cover up to 200 meters. Wi-Fi implementation is beneficial for consumers if the device is power efficient and cheap. This chapter investigates both of these aspects to create a low cost, power efficient Wi-Fi capable device.

The approach to reducing power usage in IoT is, for devices to rely on batteries to last for a reasonable amount of time. Then the barrier of having to be near a power source would be removed, allowing greater flexibility and uptake. Previous research has looked into the power consumption aspect of these devices [72],[73]. As a Wi-Fi enabled IoT device is currently not designed, very few researchers have looked into the aspect of power consumption of a Wi-Fi enabled IoT device.

This chapter considers the efficiency of the device in the long-term. For the device to be commercially viable, it was crucial to develop an efficient way to keep the chip operational without running out of power. Power issues tend to be avoided in favour of product features. So for the particular prototype the power consumption became the focus.

5.2 Practical Battery Tests

Discharge current has to be taken into consideration when looking into a battery capacity. Discharge current is different for different battery types. A measure at which the battery discharges is known as the C rate [106]; the C rate is relative to the battery's maximum capacity. For a battery with 1C rate means the current would discharge the battery's entire capacity within an hour. For example a battery with a capacity of 100Amps per hour with a 1C rate would have a discharge current of 100Amps.

For long-term fit-and-forget applications, Lithium Thionyl Chloride such as the LS14250 [107] provides an excellent option with high energy density and very low self-leakage. With a voltage of 3.6 volts and an ability to provide peak currents of tens of milliamps, no regulator is required for the ESP-03 which saves on energy use. Available in a range of capacities, the AA size battery provides between

2200mAh and 2500mAh depending on the manufacturer. Allowing for leakage, such a battery would give three years life when transmitting every 4 hours, or over ten years life averaging one transmission per day. As testing with such a battery is not possible during this thesis experimental work due to the timescale the battery life of ten years mentioned is through calculations. Therefore a smaller battery is used for test purposes and supported with results in subsequent paragraphs and sections.

The battery used for the purpose of testing is a lithium ion battery which has a nominal capacity of 110mAh at a discharge current of 0.2C, which equates to 22mA. Normally 1C is referred to one hour discharge; 0.5C for 2 hours and 0.2C for 5 hour discharge. Batteries with very low discharge currents may not give their rated capacity due to the increased influence of self discharge. Lithium ion batteries are protected from high discharge currents which makes it unsuitable for high wattage transceivers. Normally the discharge rate of a lithium ion battery is limited to a maximum of 2C. Table 5.1 is a representation of the discharge for the lithium battery used for testing.

Table 5.1: Battery Discharge

C-rate	Time
5C	12 minutes
2C	30 minutes
1C	1 hour
0.5C or C/2	2 hour
0.2C or C/5	5 hour
0.1C or C/10	10 hour
0.05C or C/20	20 hour

5.2.1 Testing with Lithium Ion Battery

The normal calculation for calculating the hardware hours of expected runtime is

Battery capacity (mAh) / Average Power consumption (mA) = Hours of expected runtime

At the time of testing a 3.7V lithium ion battery with 110mAh was available. It was connected onto the chip and was left running under different circumstances, Wi-Fi connected and disconnected with and without sleep. The Wi-Fi connected state is when the chip is responsible for a connection all the time even when no transmission occurs whereas a sleep mode would ensure the Wi-Fi only wakes up for transmission and returns back to sleep to save power. Wi-Fi disconnected scenario is when the chip after transmission becomes inactive and would have to start the entire process of connection and authentication before another transmission. Sleep in the Wi-Fi disconnected situation means that the chip still goes into complete inactivity during the sleep time and would still have to go through all the procedures to get connected (Section 2.2.2.4) . Figure 5.1 illustrates the testing of the device with the lithium battery.

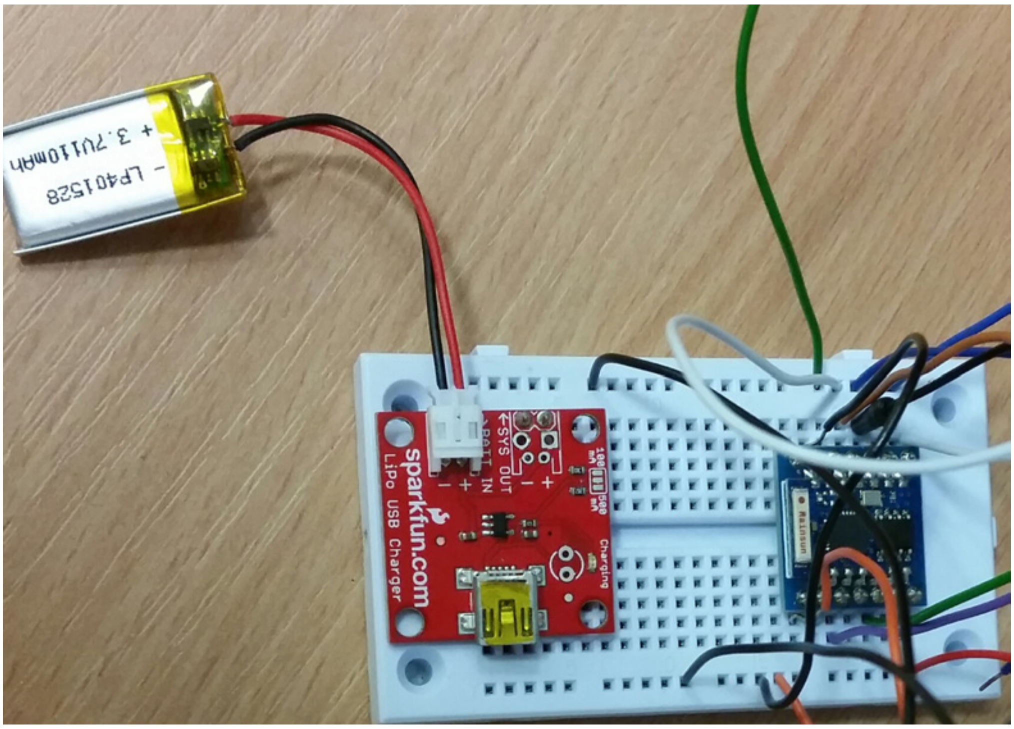

FIGURE 5.1: Testing with battery

The test procedure was then carried out in different circumstances to determine the longest runtime for the chip with different possibilities. The test results are presented in Table 5.2.

Table 5.2: Power Consumption

Wi-Fi Connected	Wi-Fi Disconnected	Sleep (Seconds)	Hours Battery Lasted
Yes	-	-	2hrs
Yes	-	60	5hrs 20 mins
-	Yes	60	4hrs 50mins
Yes	-	600	38hrs
-	Yes	600	32hrs 30mins

Without sleep and Wi-Fi connected, the battery was drained out in close to 2 hours time which would not be beneficial for the consumer. However the battery provided 38hours of runtime with a sleep time of 10minutes between transmission.

The Wi-Fi disconnected gave less time compared to the always Wi-Fi connected situation. This is because every time the chip tries to connect to the Wi-Fi it consumes significant power than if already connected to transmit temperature readings. A sleep time of 10 minutes would not affect the smart home user as the temperature variation within this time would not be high enough. 2 AA (2200mAh) batteries were also used for testing the power consumption. The chip drained out the battery in two weeks but when the transmission is reduced to one or two transmission within 4 hours, the battery life would last for three years. Battery tests were carried out to corroborate the theoretical calculation or power recorded using a device such as portapow [108] Figure 5.2.

These tests at the initial stages of the design illustrated the need to use Wi-Fi connected with sleep implemented to save the battery life of the device. The decision was then made to use sleep intervals for further prototype testing. The next section explains the procedure and the results obtained while testing ESP03 with power measuring device Portapow.

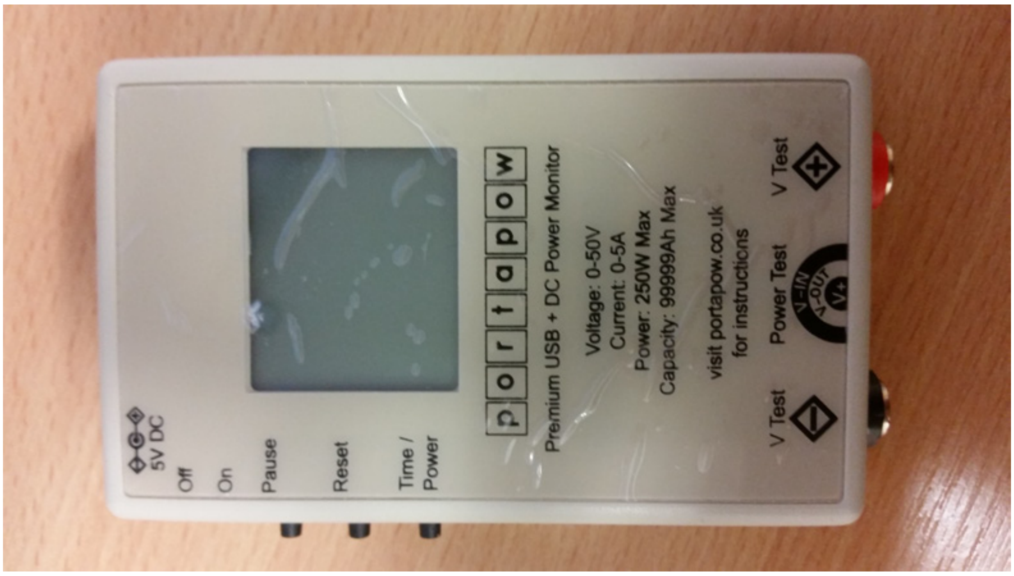

FIGURE 5.2: Portapow measuring device

5.3 ESP power consumption

The aim is to determine the power consumption of the chip to ensure it could last longer if to be used as a sensor device for purposes such as in a smart home. Replacing batteries constantly for the chip would be inconvenient and therefore power consumption tests were carried out on the chip.

ESP chip power measurements were taken using a Portapow power measuring device as displayed in Figure 5.2. The ESP03 prototype described in Chapter 4 was used to record power consumption. During the initial testing phase with batteries it was found that using the chip with Wi-Fi Connected drains less power. The power consumption of the ESP chip recorded over the time of 24 hours is provided in Figure 5.3. The breakdown of results of power consumption of ESP03 is provided in Table 5.3.

The results obtained through battery testing and power consumption through power monitor device portapow both agree on the fact that with deep sleep the ESP works longer compared to the situation without Wi-Fi.

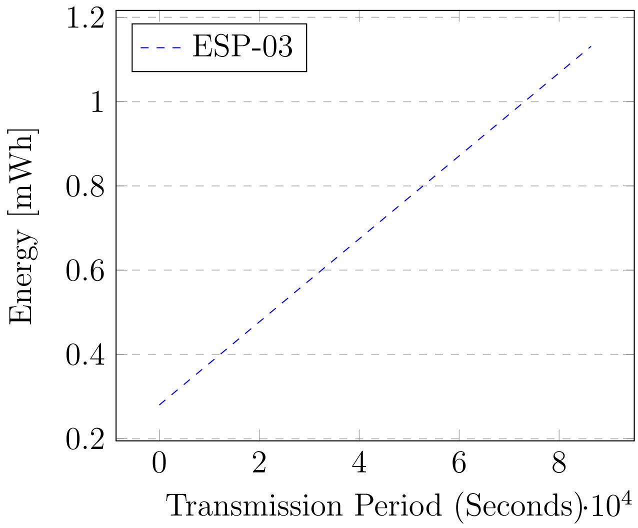

FIGURE 5.3: ESP-03 Power Consumption

TABLE 5.3: Power Measurements of ESP

Time Delay	Power Used by ESP (mWh)
10 minutes	0.285
30 minutes	0.297
1 Hour	0.315
2 Hour	0.350
3 Hour	0.386
4 Hour	0.421
5 Hour	0.457
6 Hour	0.492
7 Hour	0.528
8 Hour	0.563
9 Hour	0.599
10 Hour	0.634
11 Hour	0.670
12 Hour	0.705
13 Hour	0.741
14 Hour	0.776
15 Hour	0.821
16 Hour	0.847
17 Hour	0.883
18 Hour	0.918
19 Hour	0.953
20 Hour	0.989
21 Hour	1.024
22 Hour	1.060
23 Hour	1.095
24 Hour	1.131

5.4 Comparison of ESP03 with CC3000

The CC3000 was one of the first highly integrated Wi-Fi modules to become available. Priced at significantly less than $10, the CC3000 allowed Wi-Fi to be connected to small devices with very little additional design effort. The chip is popular among hardware makers and users due to its cost, easy to use and low power features [109]. This chip was chosen due to its popularity and the cost of the chip compared to other Wi-Fi chip is low [110]. The CC3000 supports 802.11g, but not 802.11n like newer devices such as the ESP. When launched, the manufacturer claimed energy use per transmission down to 3.6 μAh, which is very impressive but is under ideal conditions and does not consider the rest of the circuit, including a controlling microcontroller which is required.

5.4.1 Implementation of CC3000

There are several ways to program the CC3000 device, displayed in Figure 5.4; the most popular way, is to build a project using an Integrated Development Environment (IDE). The two main development environments for working with Texas Instrument devices are: IAR Embedded Workbench and Code Composer Studio (CCS) [10]. These environments are useful for integrating new functionality to the basic WiFi application by allowing the user to develop their own features. Another way to flash the code onto the CC3000 is by executing and running a binary file. A binary file allows the code to be pushed on to the device without the need for an IDE.

The binary File was pushed on to the chip using the terminal. Figure 5.5 illustrates the terminal commands used and the file pushed onto the chip. The HyperTerminal is used to send commands to the CC3000 device when using the basic Wi-Fi application. In order to establish communication between the PC and the device the hyper terminal settings on the computer must be configured to match

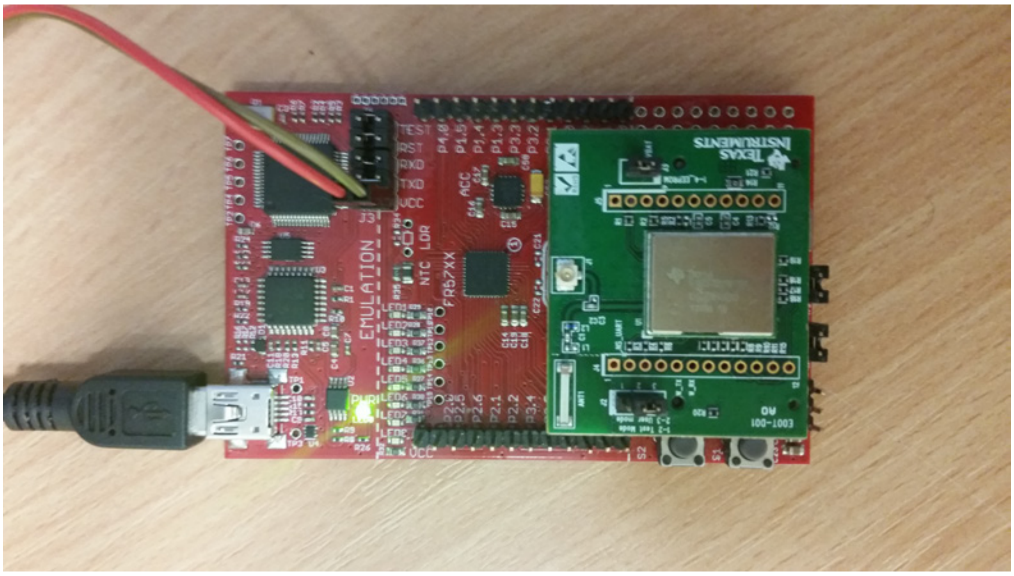

FIGURE 5.4: CC3000 Device

the specifications of the devices hyper terminal. These settings can be configured through the device manager on a windows computer. Figure 5.5 shows the device and its COM port on a Windows 7 PC. In order to configure the device to receive the data from the computer we must make sure that the baud rate [111], [112] settings are the same as that of the device. The baud rate is the rate at which the MSP430 samples the data when receiving and transmitting data through serial communication. If the PC sends data at a different baud rate than the MSP430 samples, then the data received will be out of sync. The baud rate settings can be accessed by selecting the device through the device manager [111], [112].

Since the CC3000 is an older model, it does not support the later IEEE 801.11n wireless networking standard. Most modern routers using 801.11n have greater data rates than what the CC3000 can handle, i.e. greater than 54 Mbits/sec. In order for the CC3000 to connect to the router, the router must be forced down to a previous networking standard (801.11g) which supports 54 Mbits/sec transfer rates. Configured a NETGEAR router, the data rates from 134 to 54 Mbits/sec i.e. 801.11n to 801.11g.

After the connection is established using a teraterm terminal, it has to be opened to transmit data to the destination. The command for the CC3000 to transmit

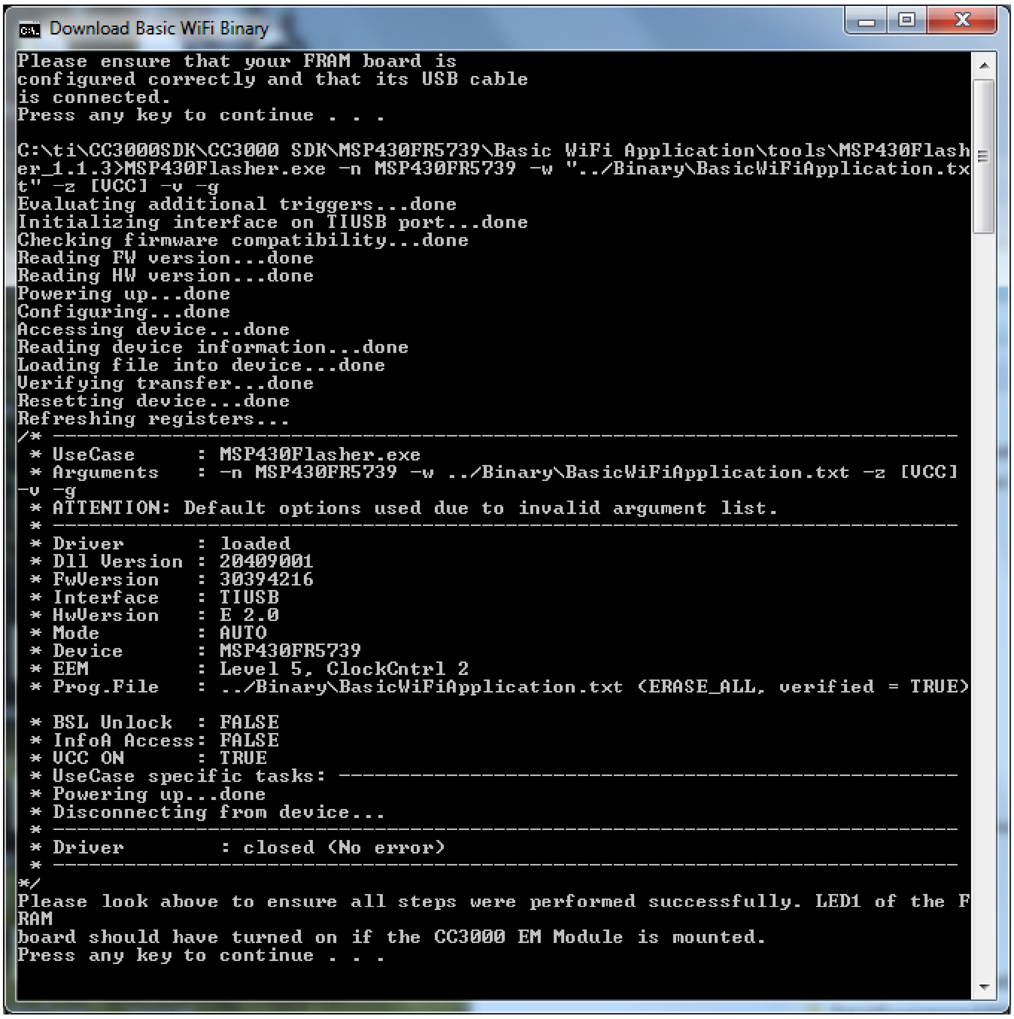

Figure 5.5: Binary File

data should be represented in hexadecimal. The command should follow the socket opening followed by data length, data, port number and IP address [113].

5.4.2 CC3000 Results

The results for CC3000 is compared against the low cost Expressif ESP8266 chipset from the previous section. Figure 5.6 illustrates the results obtained for CC3000 with ESP03 results included within the graph for comparison.

A breakdown of the results from the graph is provided in Table 5.4. It can be seen that in all cases as the transmission interval increases, so does the energy use: this is due to the energy consumed during the sleep cycle between transmission.

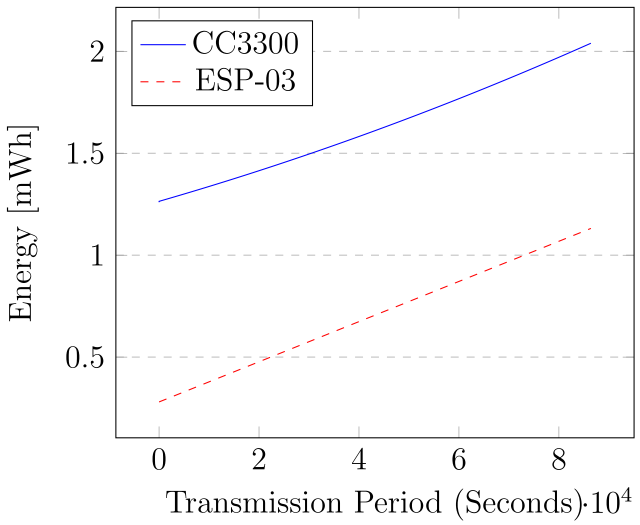

FIGURE 5.6: CC3000 and ESP03 Power Consumption

TABLE 5.4: Energy Use per Transmission

Transmission every	CC3000 (mWh)	ESP (mWh)
10 minutes	1.268	0.285
30 minutes	1.277	0.297
1 Hour	1.289	0.315
2 Hour	1.316	0.350
3 Hour	1.343	0.421
4 Hour	1.370	0.563
1 Day	2.039	1.131

However, the energy usage of the transmission itself dominates. The CC3000 has the poorest performance in terms of energy usage, and significantly higher than the manufacturer's claims. This is due to the fact that energy use of the complete system is considered, and the CC3000 requires quite a significant microcontroller to control it. The ESP-03, which is completely integrated, has lower energy usage.

5.5 DHCP and Static IP

IP allocation is used for all types of network these days to reduce the complexity of management. These IP addresses are allocated using Dynamic Host Configuration Protocol (DHCP) which in a home network scenario often lasts for longer

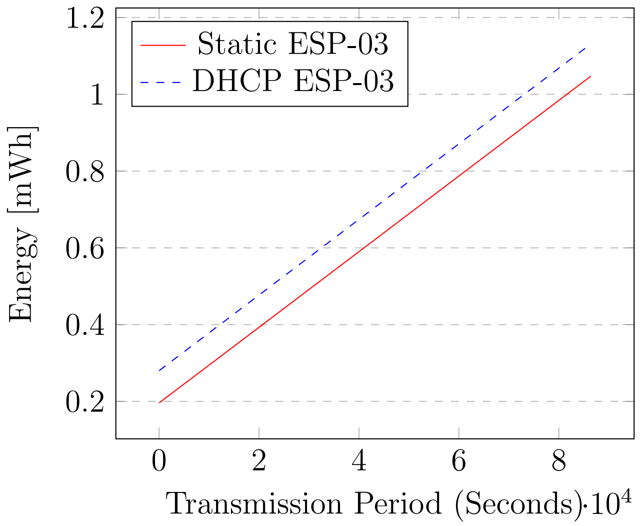

FIGURE 5.7: Energy Analysis of ESP-03 Static Vs DHCP

number of hours. Implementing static IP is a bit more complex than the automatic allocated DHCP. The network has to remember the device and each time during the time of connection the same IP is used over and over again. While using an ESP03 the router it connected to could allocate a static IP. Experiments are carried with ESP03 being allocated both DHCP and static IP. Static IP are used to reduce transmission time whereas considering deployment and simplicity DHCP is considered and used for all the experiments carried out within this thesis. A comparison of these results with the DHCP is shown in Figure 5.7. The results illustrated the power saving obtained through the use of Static IP. From the results it can be concluded that Static IP would reduce the power consumption by 0.1mWh.

In practice, in a home network the DHCP lease time is usually 24 hours. The ESP03 could therefore assume a 'static' IP allocation as long as it transmits again within 24 hours.

5.6 Conclusion

In this chapter, the power consumption of the Wi-Fi module used within the device, the ESP03, was recorded and discussed with results. The results illustrated the fact that ESP03 is efficient only for short periods of time and therefore to save power over longer periods of time the device should reduce the number of transmissions.

The module was then compared with an alternative Wi-Fi device CC3000 to derive a conclusion as to how efficient the ESP03 is in terms of performance. Based on the results the CC3000 consumed more power than the ESP03, therefore indicating that ESP03 is an efficient module for incorporating into IoT devices. To reduce the power consumption of the ESP03, DHCP and Static IP configurations were tested and discussed with results. Static IP address allocation showed a reduction in power consumption compared with the DHCP IP allocation. The next chapter, discusses optimising the power consumption of the Wi-Fi module and compares the results with a popular protocol currently used within these devices, Bluetooth Low Energy.

Chapter 6

Power Consumption with an additional microcontroller

6.1 Introduction

In the previous chapter, a SoC Wi-Fi device was used for the sensor. However, the power consumption recorded by using the ESP03 chip in the previous chapter was high and an alternative approach used to solve this problem was to implement this chip as a modem for a MSP430. The reason for high power drain by the chip is because it reads the data from the sensor and provides voltage to the sensor. Along with this the chip had to use up power to connect to the network and later transmit the received temperature data. This justifies the results obtained in the previous chapter and the need for an alternative solution to the problem. This chapter discusses the alternative chosen in which ESP will act as a modem within MSP430, a microcontroller designed for very low power operation.

6.2 Architecture

The initial stage of developing the concept was to ensure the ESP03 chip is only used for transmission and MSP430 deals with all other processes. Based on the concept the power consumption should be reduced as the ESP03 is only a slave for the MSP430. Figure 6.1 illustrates the design phase of the new architecture. According to the Figure, ESP and the temperature sensor would be completely controlled by the MSP430 and would only wake up when requested by the MSP430. The detailed implementation of the design is discussed in the following Section.

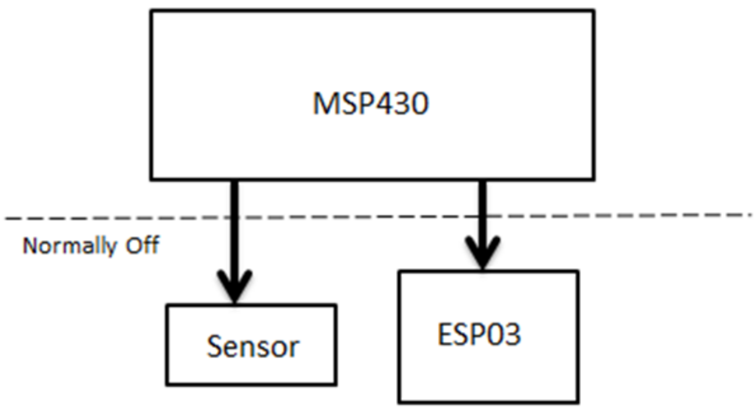

FIGURE 6.1: Design Phase

6.3 Implementation of MSP430 and ESP

The first stage established that the Wi-Fi chip would respond to the commands from the MSP430. A simple program was created where the LED on the MSP430 blinks if the Wi-Fi chip is correctly connected over the UART, a visible indication that both the boards are communicating with each other. The pin connections are stated below:

- P1.1 to TX

- P1.2 to RX

- Common power supplies together

Energia IDE was used to implement code for the MSP430 with the Wi-Fi chip acting as a modem to connect to a router and server. Incorrect jumper pins on the MSP430 would result in no output. Parasite power mode could be used for the sensor to save power but according to the datasheet of the sensor the use of MOSFET is required between conversions acting as a strong pull up. This increases the complexity of the design and used the regular mode of connection for the sensor.

The temperature sensor was connected with the MSP430 to ensure device operation. The temperature sensor was powered by the MSP430 and read by the pin P1.4 on the MSP430, as shown in Table 6.1.

Table 6.1: Pin Connections

MSP430	DS18S20
P1.7	Vdd
GND	GND
P1.4	Data

The Vdd of the DS18S20 connected to MSP430 had a data pull up of 1k resistor or 2k resistor, to maintain the voltage. Using P1.7 to supply voltage to the sensor allows switching the supply similar to a GPIO switch. This reduces the complexity when creating the PCB for the design. Figure 6.2 shows the temperature reading obtained with the MSP430. Figure 6.3 illustrates the working phase of the MSP430 with a temp sensor - DS18S20.

The successful development of the initial test phase gated the evolution of the ESP chip with the additional MSP430 on the same protoboard design. This would enable the MSP430 to work as master/slave wherein the master gathers the data

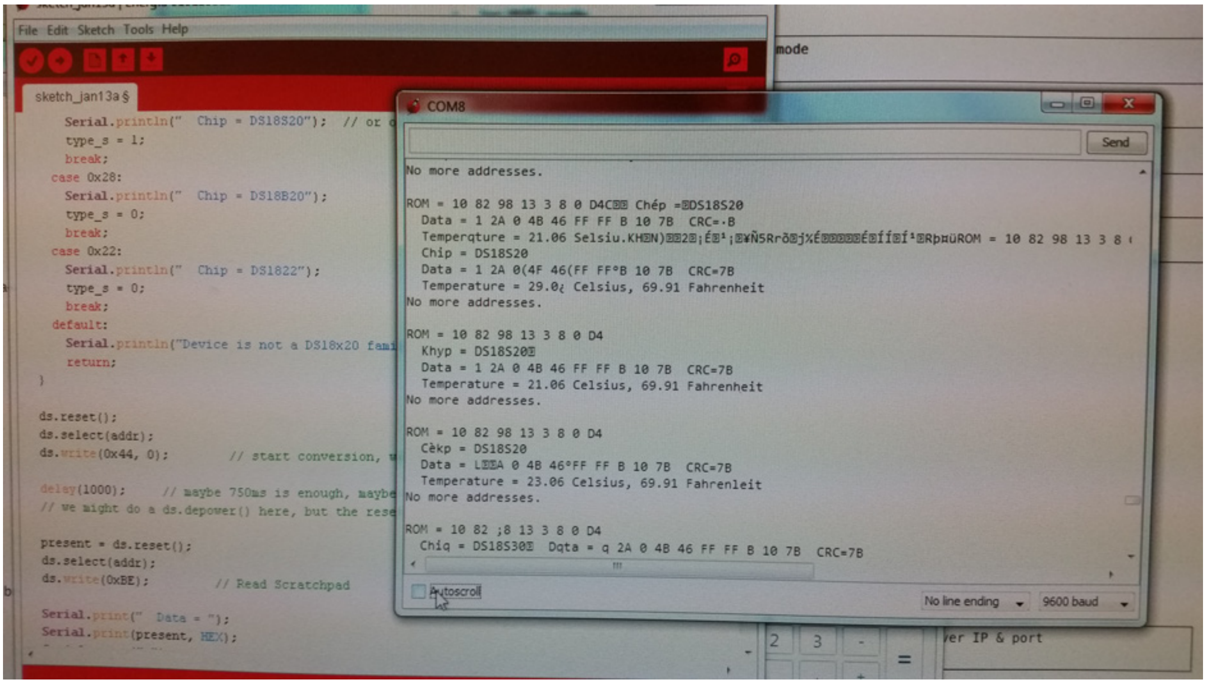

FIGURE 6.2: Energia IDE

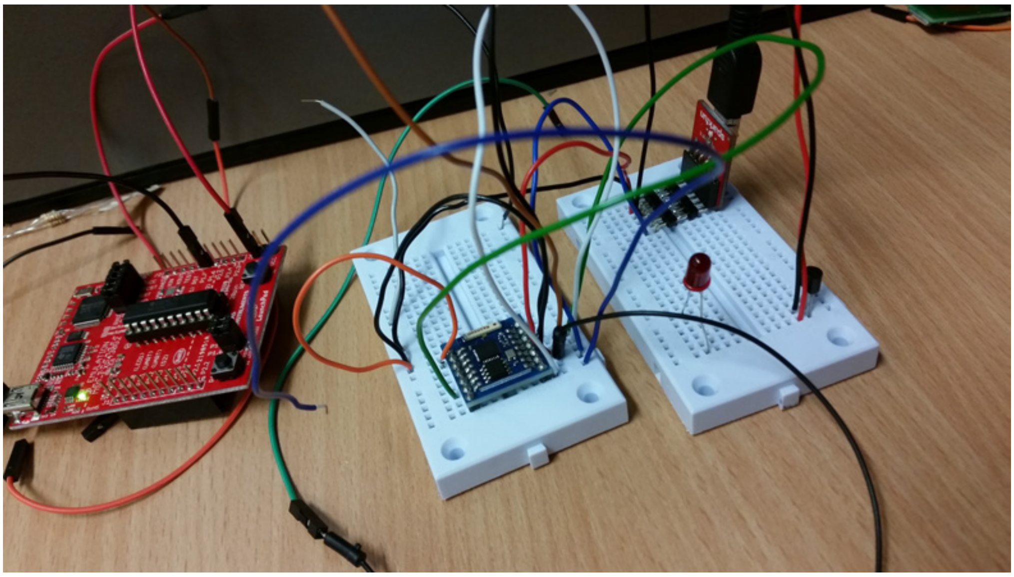

FIGURE 6.3: Initial Phase

from the slave and only turns on the ESP when required to transmit the data over the network. Instead of connecting the temperature sensor to the GPIO pin of the ESP, the sensor is now connected to the GPIO pin of the MSP430.

6.3.1 Testing Phase of ESP with inbuilt MSP430

The test was carried out in the final prototype in which the MSP430 chip is inbuilt on the ESP board. An external MSP430 board was used to push the code onto the protoboard. The MSP430 within the ESP protoboard would be the master. When connecting together both master and slave to communicate each other and gather temperature reading, the ESP was sinking the signal sent to the MSP430 slave. Energia IDE was used to push the code onto the device. The code was pushed and an UART was used to view the output. The output viewed had random characters which was expected as ESP was sinking the signal. The MSP430 output presents the temperature readings on the output screen rather than random characters. Figure 6.4 and Figure 6.5 illustrates the final board and the process of testing.

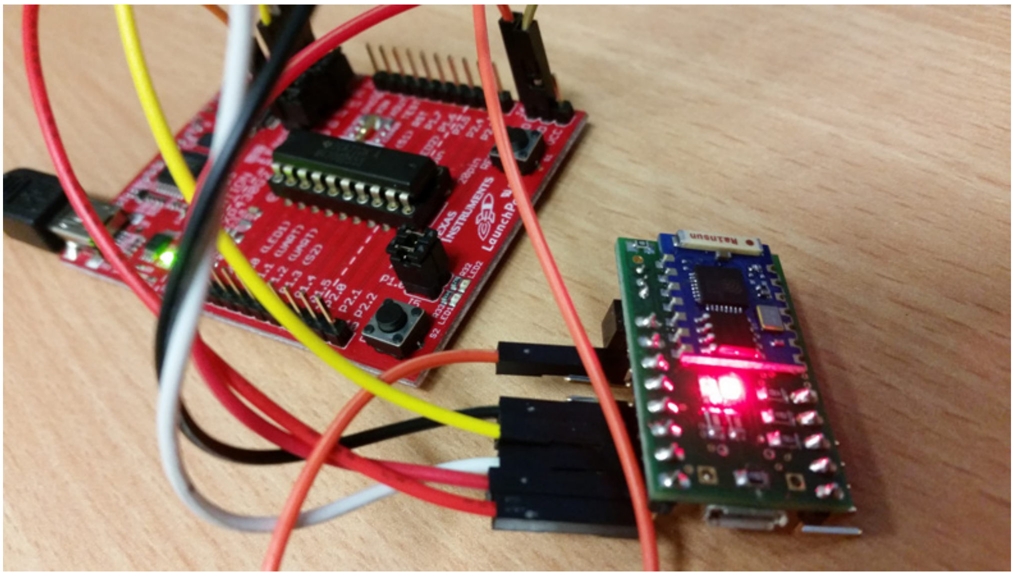

FIGURE 6.4: Testing Phase

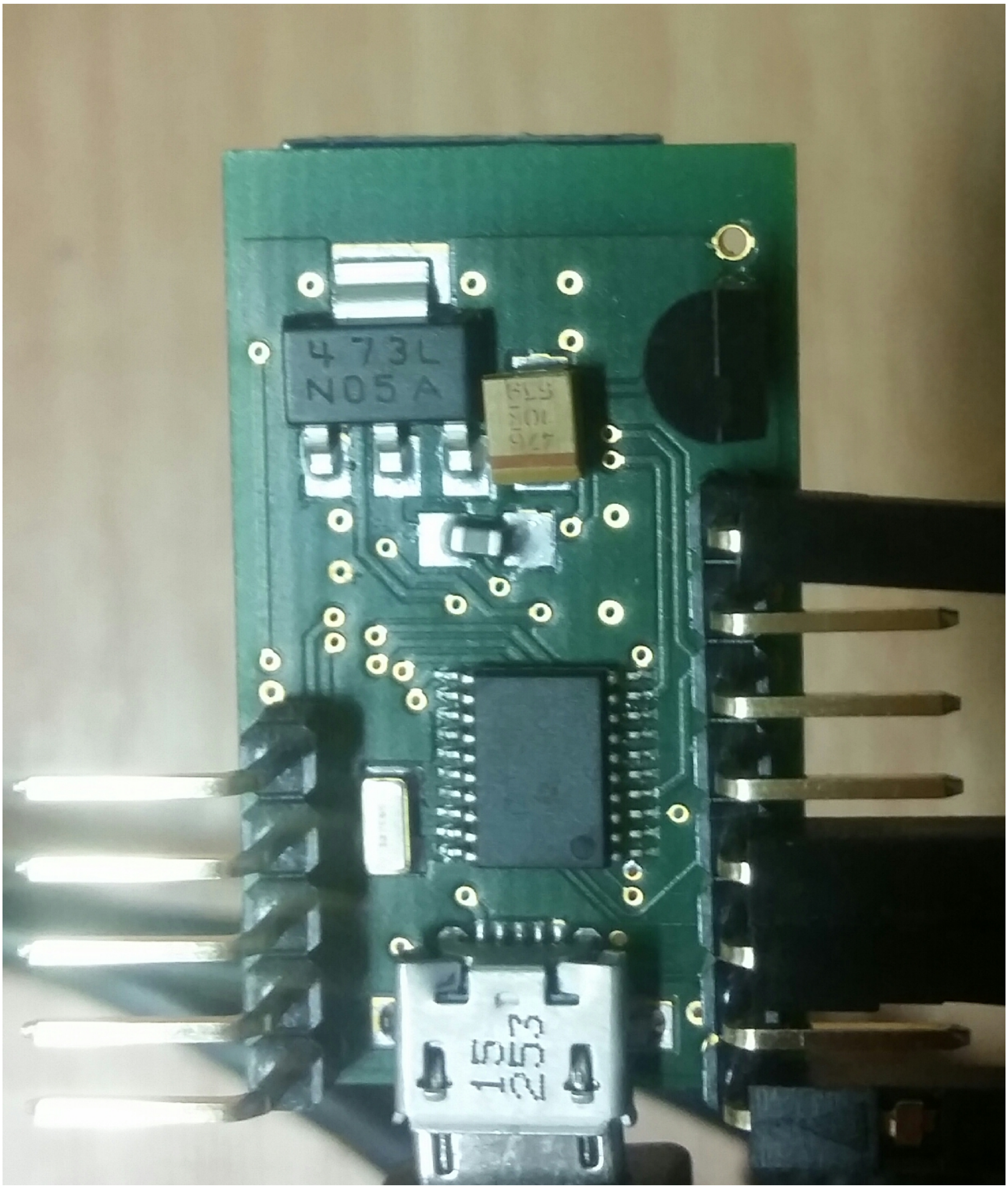

FIGURE 6.5: MSP430 chip on ESP board

The code was modified such that the LED on the protoboard would blink after gathering the temperature data. The LED did blink but the output was not able to be displayed therefore a bus pirate was used to view the output. The connections of the final board, bus pirate and the additional MSP430 board is provided in Table 6.2. The schematics and the board created using Eagle software for the

Table 6.2: Chip Pin Connection

MSP430 (External Board to Push Code)	ESP03 board with inbuild MSP chip	Bus Pirate
VCC	VCC	-
TEST	TEST	-
RESET	RESET	-
GND	GND	GND
-	URXD	MISO

final product is included in Appendix A (Figure A.6, Figure A.7 and Figure A.8).

Few problems were encountered while attempting to connect the ESP and transmit the data read. At the beginning the temperature read by the MSP430 pin could not be transmitted as a STRING as it did with an Arduino IDE. The request to the server also had to be printed in three lines in which one of the lines would include the temperature data variable. This method of avoiding STRING for the variable was send request using few print lines.

```
Serial.print(‘‘GET /test/msp430/ ‘‘);
Serial.print(celsius);
Serial.print(" HTTP/1.1\nHost: 192.168.1.2\nConnection: close\n\n"
);
```

Tests were carried out to ensure any temperature data read would be transmitted. For example, a single digit reading and a double digit reading were tested to ensure any type of reading would be forwarded without crashing the device. The code was modified such that the LED would blink when the ESP is connected to the server and at the end after the data is transmitted the LED would go HIGH and LOW for few seconds.

The power recorded by the device and ESP03 from previous chapter is shown in Figure 6.6, as expected power saving was obtained. If the device has to run for a longer period of time then using ESP03 alone would compromise the power saving. For a relatively short period of time using ESP03 on its own would save power

Table 6.3: Power Measurements of MSP+ESP

Time Delay	Power Used by ESP+MSP430 (mWh)
10 minutes	0.331
30 minutes	0.333
1 Hour	0.337
2 Hour	0.344
3 Hour	0.344
4 Hour	0.352
5 Hour	0.359
6 Hour	0.367
7 Hour	0.374
8 Hour	0.381
9 Hour	0.389
10 Hour	0.396
11 Hour	0.404
12 Hour	0.411
13 Hour	0.419
14 Hour	0.426
15 Hour	0.433
16 Hour	0.441
17 Hour	0.448
18 Hour	0.456
19 Hour	0.463
20 Hour	0.471
21 Hour	0.478
22 Hour	0.485
23 Hour	0.493
24 Hour	0.500

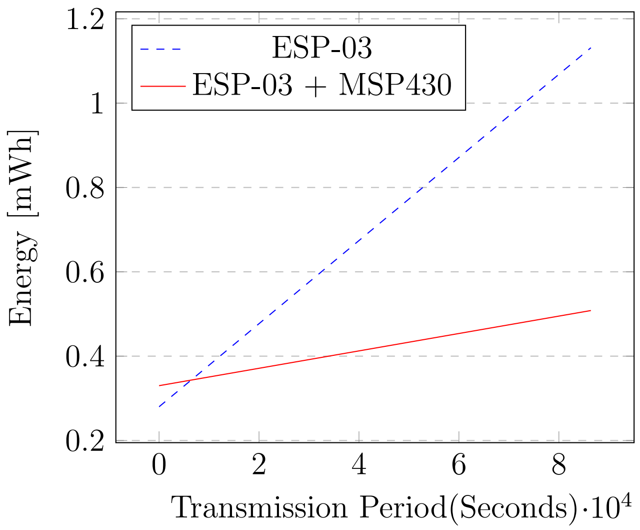

Figure 6.6: ESP03 vs MSP430 Power Consumption

which is evident by comparing the results in the Table 6.3 and Table 5.3. The majority of these IoT devices are installed in homes to operate for a longer period of time therefore the need to have a reliable system is beneficial.

6.4 Comparison of MSP430+ESP with BLE

Based on the results an alternative to reduce the power consumption, MSP430 and ESP, was discussed and proved with results. The results illustrated the fact that using an extra processor such as the MSP430 would reduce the power consumption of the Wi-Fi chip greatly resulting in power savings.

A comparison of the MSP430+ESP with BLE is undertaken. In this scenario the MSP430 [114] microcontroller would receive the temperature data from the DS18S20 sensor and using the BLE module, transmit it to a user's mobile device using Bluetooth. An MSP430 processor is used in this scenario with the BLE used as a slave device.

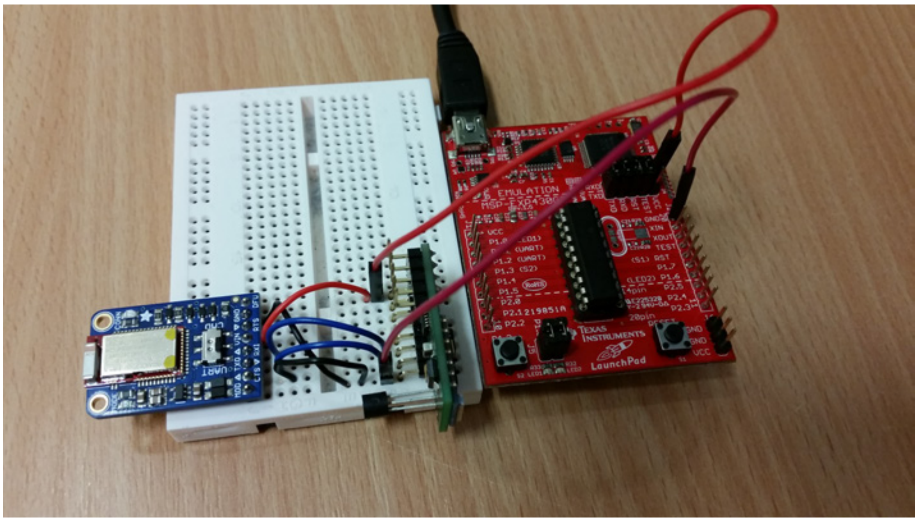

Figure 6.7: Experiment Setup

The code is set up such that the BLE module transmits temperature data to the user when the user inputs a specific character command. This command acts as a flag in order to control when the data is transmitted. After the device has received

TABLE 6.4: Energy Use per Transmission

Transmission every	BLE (mWh)	ESP (mWh)	ESP+MSP (mWh)
10 minutes	0.607	0.285	0.331
30 minutes	0.607	0.297	0.333
1 Hour	0.608	0.315	0.337
2 Hour	0.610	0.350	0.344
3 Hour	0.612	0.421	0.359
4 Hour	0.616	0.563	0.389
1 Day	0.813	1.131	0.508

the character, the temperature data is transmitted by the MSP430 + DS18S20 sensor via Bluetooth to the users mobile device. In order to get as accurate power measurements as possible, the Portapow is used to record to power consumption for each device [108].

If the BLE module is constantly transmitting data, then a clear measurement of the power consumption is difficult. Constantly transmitting data is impractical for IoT devices which solely rely on battery packs as their main source of power because the BLE consumes power at a much higher rate when transmitting. By choosing to only transmit data when the user requires it, can reduce the overall power being consumed thus increasing the life span of the system. This method is used to replicate and compare the power consumption of sleep mode in Wi-Fi modules.

When a Wi-Fi module enters sleep mode, all connections remain established between devices. With BLE the reduction of power consumption over a long period of time can only be acheived if the user disconnects the device and reconnects when the data is needed. When a connection has been established and maintained by the BLE module and device, the BLE module consumes more power than when having no devices attached. In light of this, all devices are disconnected from the BLE module to ensure the module consumes as little power as possible.

The results in Figure 6.8 illustrate the three different power consumption scenarios. With the Wi-Fi only module, the power consumption is higher than the BLE after

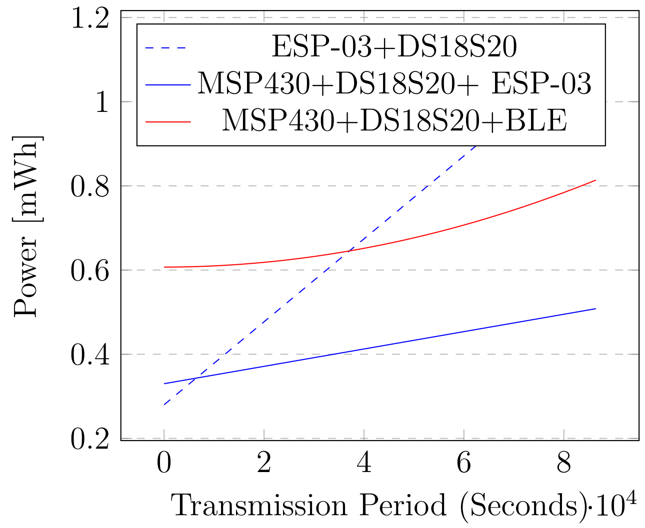

FIGURE 6.8: Comparison Graph

27 transmissions. In the beginning however, the BLE's power consumption is higher. In the long term, the BLE is more efficient in terms of power consumption. While comparing the BLE module to the Wi-Fi inbuilt device developed, it is evident that the power consumption of the BLE is higher in both the short and long term. This is due to the extra power being consumed by the devices, when establishing and de-establishing connections between the BLE when transmitting data. Whereas with our developed device, the Wi-Fi module goes into sleep mode retaining all previous connections with the user. Therefore, each time the Wi-Fi module awakens, a new connection does not need to be established. Overall, this results in less power being consumed by the Wi-Fi inbuilt IoT device. Table 6.4 provides an overview of results over a long period of time.

6.5 Comparison of MSP430 with ESP03 and 433MHz

Power consumption of an MSP430+ESP03 was compared against MSP30+433MHz. MSP430 Chip was programmed to output pulses on a digital GPIO pin. This pin was then connected to the data input on the AM transmitter, along with all appropriate power connections. Portapow was used to record power measurements. Figure 6.9 illustrates the results obtained.

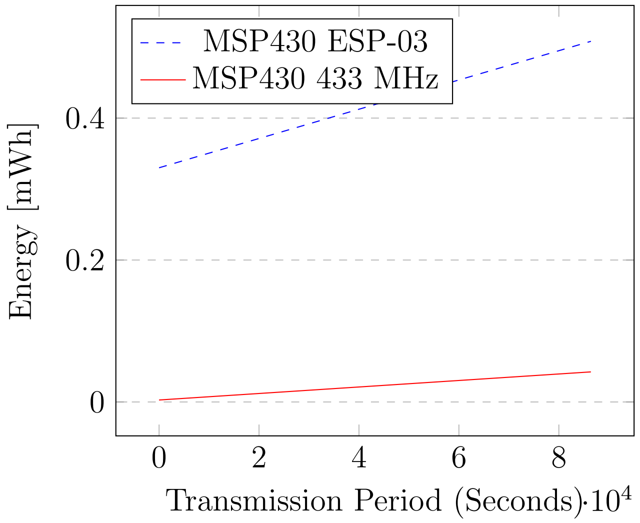

Figure 6.9: Comparison of Wi-Fi vs 433 MHz using MSP430

Although MSP430 + ESP03 uses significantly more energy for an equal number of transmissions over a given period, it is worth noting that due to physical limitations of operating at a higher frequency, 2.4 GHz vs 433 MHz. Coupled with the fact that in order to transmitt data the ESP-03 has to create the whole ISO stack, while the AM system can transmit on the equivalent to the physical layer. The energy requirements for the Wi-Fi system is always going to be higher. But the Wi-Fi system has a significantly higher data rates, so if a system is required a lot of bandwidth to be transmitted it would be more efficient to use the Wi-Fi system. Furthermore since when connecting to a secured wireless network authentication is required which means that only sensors which had been authorized to access and transmit on the network could provide information, therefore providing more security. Where as within a basic AM system any received signal will be interrupted allowing the possibility of false signals.

6.6 IP Address Allocation

Home networks and Wi-Fi networks often use dynamic IP address allocation for ease of management. IP addresses are allocated using DHCP. Since the ESP-03 is switched off to save power between transmissions, an address has to be allocated whenever the device is switched on, i.e., for each transmission. Since the transmissions are short, the DHCP exchange, although short, has an impact on battery life.

In fact, DHCP is a very flexible protocol, and could be used to send measurements without an overlaid Internet service, but the DHCP server would have to be configured to allow this. Since we want to use standard Internet protocols and reduce deployment issues, we would not want to use DHCP in this manner. However, in a home network, IP addresses are normally allocated for an extended period of time (for example, 24 hours). For the MSP430 based device, the MSP430 can be used to store the allocated IP address between transmissions, so when the ESP-03 is switched on it can use a 'Static' address from the MSP430. We have investigated power consumption with 'Static' IP to investigate the reduction in power consumption through the saved DHCP exchange.

Figure 6.10 presents power consumption comparison of static and DHCP IP address allocation on both the standalone ESP-03 and the ESP-03 + MSP430. As shown in the graph there is roughly 30% reduction in energy consumption when using static IP address allocation compared to DHCP.

6.7 Additional Sensors

Although the sensor used within the prototype is a temperature sensor, DS18S20, the prototype is capable to support any type of sensors and not restricted to DS18S20 sensor. The benefits of using a DS18S20 sensor is that the capability of

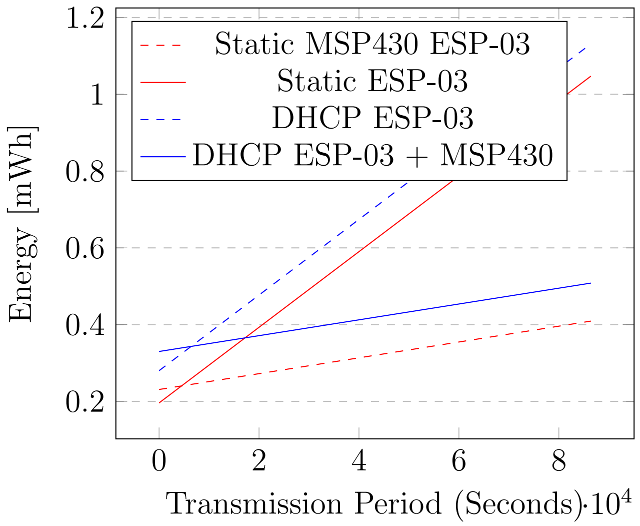

FIGURE 6.10: Comparison of Static IP assignment vs DHCP

such one wire sensors enables the addition of multiple DS18S20 to a single power supply, reducing the complexity of the design and the need for an extra power supply. The temperature reading obtained through two DS18S20 senors would be bundled by the chip to reduce the overhead and therefore the packets sizes would not be increased with the addition of a second sensor. The board is designed with additional I/O pins taking into consideration the multiple device requirements of the smart home users and therefore the developer could add on multiple sensors onto the board.

An alternative sensor used within homes is a door sensor. Door Sensors can added to the existing prototype to control the door openings and that information could be passed onto the home user frequently or infrequently. The operational lifetime is based on the frequency of the information transferred.

6.8 Conclusion

This chapter discussed the power consumption aspects of the Wi-Fi inbuilt IoT device. The chapter progresses with the results obtained for both the prototype device and BLE. Based on the results an alternative to reduce the power consumption, MSP430 and ESP, was characterised.

The results illustrated the fact that using an extra processor such as the MSP430 would reduce the power consumption of the Wi-Fi chip greatly resulting in power savings. The power issues of Wi-Fi connected IoT device has previously been avoided by the research community due to its high power consumption. Through the work carried out for this chapter within this thesis this problem of high power consumption by the Wi-Fi chip is resolved. Achieving power savings for a Wi-Fi inbuilt device increases the possibility of using it in future IoT devices. Hence the advantages provided by the ubiquitous Wi-Fi can be utilized for future devices.

Although the experiments in this thesis focusses on temperature sensor, the platform could use any type of sensors. The I/O pins of the board could be used to add on additional sensors for multiple device requirements. If an additional temperature sensor is added onto the existing platform and both the sensors are transmitting at the same time then the second sensor at the time of transmission would only transmit the data bytes with the temperature reading and would not have any overhead attached to it. This is because both the sensor readings are bundled together by the chip and therefore beneficial as it reduces the number of data packets tranmsitted.

This chapter has also compared the results of the Wi-Fi inbuilt device with Bluetooth. The results illustrated that the power consumption of the Wi-Fi inbuilt device with an additional microcontroller had lower power consumption than Bluetooth. However, the power consumption of BLE was lower when compared against standalone ESP03 chip. Static IP and DHCP IP address allocation was investigated to determine if power consumption could be further reduced for the device. Based on the results static IP was better in terms of power consumption than DHCP. The following Chapter concludes the work in this thesis and suggests future work.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

The Internet of Things (IoT) is currently a vibrant topic of research, especially within the context of Smart Home [115] and Smart Cities [116]. These devices, when implemented with sensors, can be viewed as being 'smart'; the idea behind IoT. IoT will dominate in terms of devices and therefore must be efficient. According to Cisco and Gartner, the number of these small devices deployed in 2016 would reach 16 billion [6] with an expected rise to 50 billion by 2020 [15].

According to Lattice Semiconductor, power efficiency is one of the main challenges that constrains the advancements of IoT devices [117]. Since power efficiency is necessary, a power hungry protocol such as Wi-Fi is undesirable. However, with the majority of homes, cities, transport stations, and airports already supporting Wi-Fi, a low energy solution would be favored over Bluetooth, Zigbee [118] or 433MHz. To make full use of Wi-Fi a low cost efficient and low power consuming device is required to be considered over the existing protocols such as Bluetooth, Zigbee or 433MHz. This thesis focused on this aspect and has developed a low cost, power efficient Wi-Fi enabled IoT device.

This thesis has explored the capabilities of Wi-Fi to be utilized and considered within these small smart devices as an alternative to the commonly used protocols. In Chapter 3, as LTE offers an option for IoT connection this thesis considered IoT data in LTE with the help of simulations based on the assumption of IoT data size. In order to validate these simulations, data obtained through the developed prototype was used to confirm the data size of a real world IoT device. The research question whether LTE is suitable for IoT can now be answered based on the results. LTE is not acceptable for IoT and should consider alternatives for IoT.

The implementation of the prototype described in Chapter 4 is a key contribution within the thesis as the developed prototype would be experimented and the results gathered would be used in the following chapters. In Chapter 5, power consumption aspects of the device were investigated and have been compared with an alternative Wi-Fi device CC3000. Comparison to the alternative device CC3000 enabled deriving to a conclusion of the likely system performance of the Wi-Fi chip. Based on the outcome from this chapter it can be concluded that Wi-Fi is a suitable alternative for IoT instead of LTE which answers the second research question stated in the introduction.

Lastly, in Chapter 6, based on the results obtained in Chapter 5, the device was modified to make it more efficient. An additional processor was added onto the prototype which in turn reduced the power consumption, as is evident through the results. For comparison purposes the results in this chapter was compared against Bluetooth Low Energy (BLE) and 433MHz. Power consumption issue of Wi-Fi has been solved in this chapter by the addition of a microcontroller to the device. Hence the third research question has been answered.

To conclude, the work reported in this thesis constitutes the beginning to the widespread use of Wi-Fi in IoT.

Specific contributions of this thesis are:

- Suitability of LTE for IoT

  The first contribution of the thesis considered the investigation of LTE as a suitable network for IoT sensor data. Simulation was carried out and compared against the case study included in Chapter 3. The data attained from the prototype was explained in the context of LTE network and compared with the simulation results. From the results it was evident that LTE although efficient with large packets of data was inefficient for IoT devices transmitting small packets of data.

- Low Cost Wi-Fi Inbuilt IoT device - sensor device with Wi-Fi capabilities

  Development of a low cost Wi-Fi inbuilt IoT device. A low cost Wi-Fi module, ESP03, is used within the device along with a temperature sensor, DS18S20. The design and implementation process is included within Chapter 4 and the results gathered through the device appear in the following Chapters 5 and Chapter 6.

- Comparison with existing protocols

  The power consumption of the device recorded in Chapter 5 is compared against an alternative Wi-Fi device CC3000 to compare performances of both the devices. The results in Chapter 6 are compared with the existing protocol for these devices, Bluetooth Low Energy (BLE) and 433MHz. Comparison of the Wi-Fi module ESP03 with CC3000 showed that ESP03 is efficient in terms of power consumption.

- Optimized power consumption by coupling the Wi-Fi Inbuilt IoT device with microprocessor

  The third contribution of optimizing power consumption was based on the results of the Wi-Fi inbuilt device. To reduce the power consumption even further and to control the Wi-Fi module an additional microcontroller was added onto the device. The results gathered for both, Wi-Fi with and without a processor, were compared against each other before deriving the conclusion. Based on the results from this chapter the use of an additional microntroller along with the Wi-Fi module resulted in further power savings enabling the device to be much more efficient to be used in future within IoT devices.

7.2 Future Work

The work in this thesis begins the journey for Wi-Fi being inbuilt IoT devices, as a viable option going forward. A number of aspects could now be added to take this potential further. These are discussed in this section.

During the work carried out for this thesis the prototype developed was tested in the lab room. This limits the applicability of results as opposed to results from real world scenarios in homes. Therefore the device after being commercially offered to customers by Treegreen, could be researched further using data obtained through actual homes.

At this point different types of batteries could also be tested along with the devices implemented in real homes. At the time of testing due to lack of availability of different batteries the experiments carried out for this work were limited to the batteries available at the time of testing. Testing with different batteries would conclude which batteries to be implemented in the long run, if the device has to be battery powered. An important aspect to be considered when testing with different batteries is the leakage time. Every battery over the time would drop its energy - self leakage. This often occurs when the battery is in idle mode for longer periods of time. The leakage rate is different for different batteries therefore research has to be done as to which battery is efficient to be used for these devices. Factors such as age, cycling and elevated temperature are often the reasons for self leakage of power within batteries. Batteries such as lithium and alkaline batteries are efficient in retaining the energy whereas nickel based batteries would leak the most and have to be recharged if not used for a few days [119]. When considering to use batteries for these small devices the power leakage should also be taken into consideration before concluding how long the device would work with batteries.

For the IoT data in LTE, this could be carried out using Narrowband IoT [64] in the future as it would be globally commercialized by the year 2017. It allows reduction in power consumption of user devices and increase system capacity and spectrum efficiency. NB-IoT will be supported by all 4G devices. Mobile and network security features would also be supported by NB-IoT which would be very beneficial in the long term. NB-IoT could be tested out for IoT sensor devices particularly for the prototype developed and the results validated with different scenarios. NB-IoT aims to provide extended coverage for deep indoor areas and therefore should be tested in real world applications after both the prototype and the NB-IoT is commercialized.

Security and Privacy is another key factor to be considered when deployed in large scale to customers. Although users are trustworthy there is still potential of attacks from hackers to gather information. Adding on additional features such as password protection would maintain the information securely and prevent hackers from hacking confidential information. To implement security into the developed low processing power device is challenging. The device developed as it is would use up more power if security aspects are also implemented onto the device. Memory and processing power has to be increased when adding on extra features to the device. This would mean considering a more powerful processor if needed or using the existing device but utilize it in an efficient manner to include additional features. If a bigger processor is added more power would be used up resulting in expensive batteries to be used and the overall cost of the device would increase [120]. An alternative is to use the proposed device but optimize its features such that it could work with the existing prototype. Power consumption should be low, even with the addition of security features, for the device to be efficient in the long term.

Lastly, future work should also consider using smart access techniques or sleep cycles to increase the lifetime of the device. Sleep awake cycle would ensure the reduction of power consumption of the device as well as increased lifetime. An algorithm could be developed for the sleep awake cycle and added to increase the lifespan of the device. The algorithm could be beneficial to be implemented on many IoT devices.

Appendix A

Eagle Schematics and Board

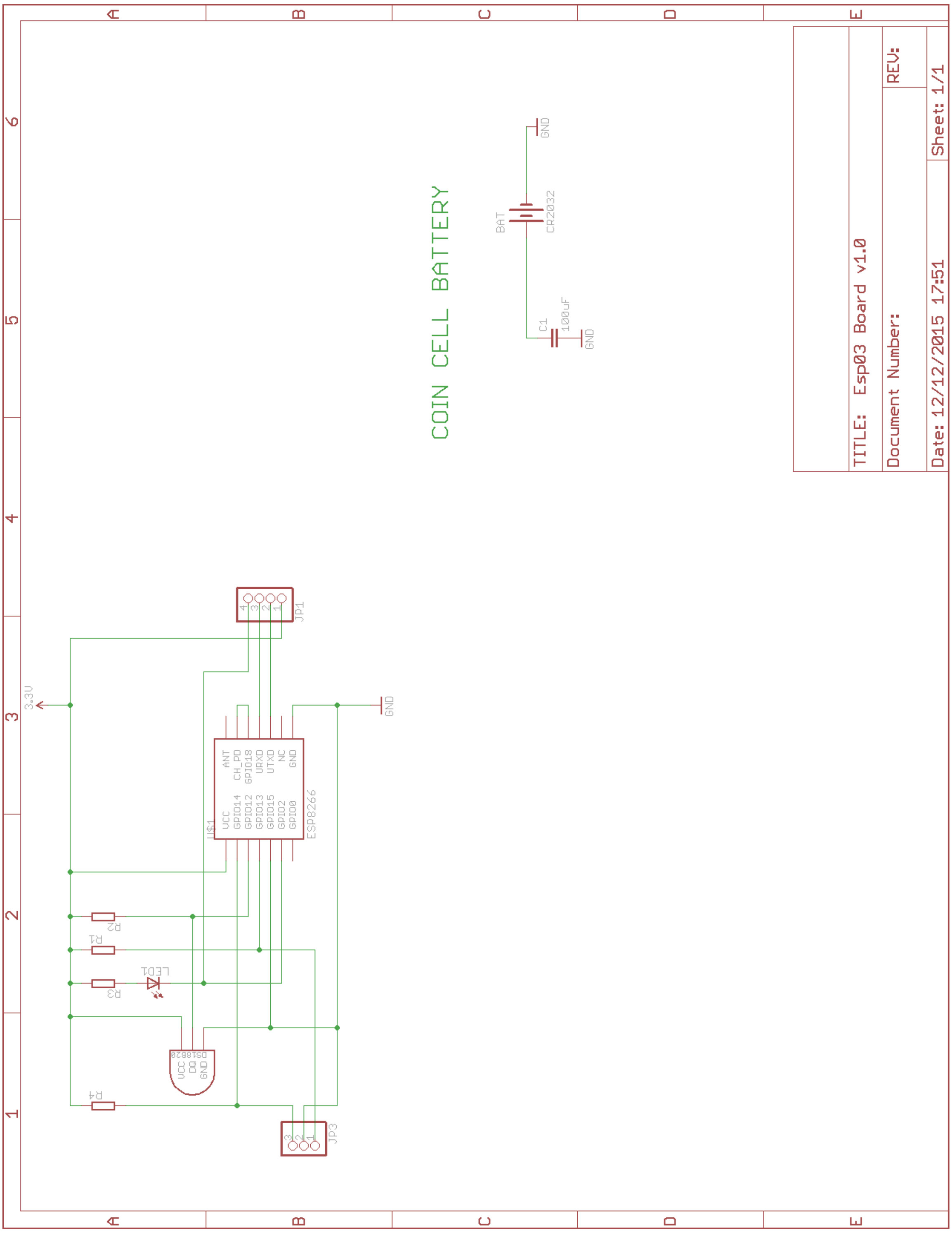

Figure A.1: Eagle Schematic

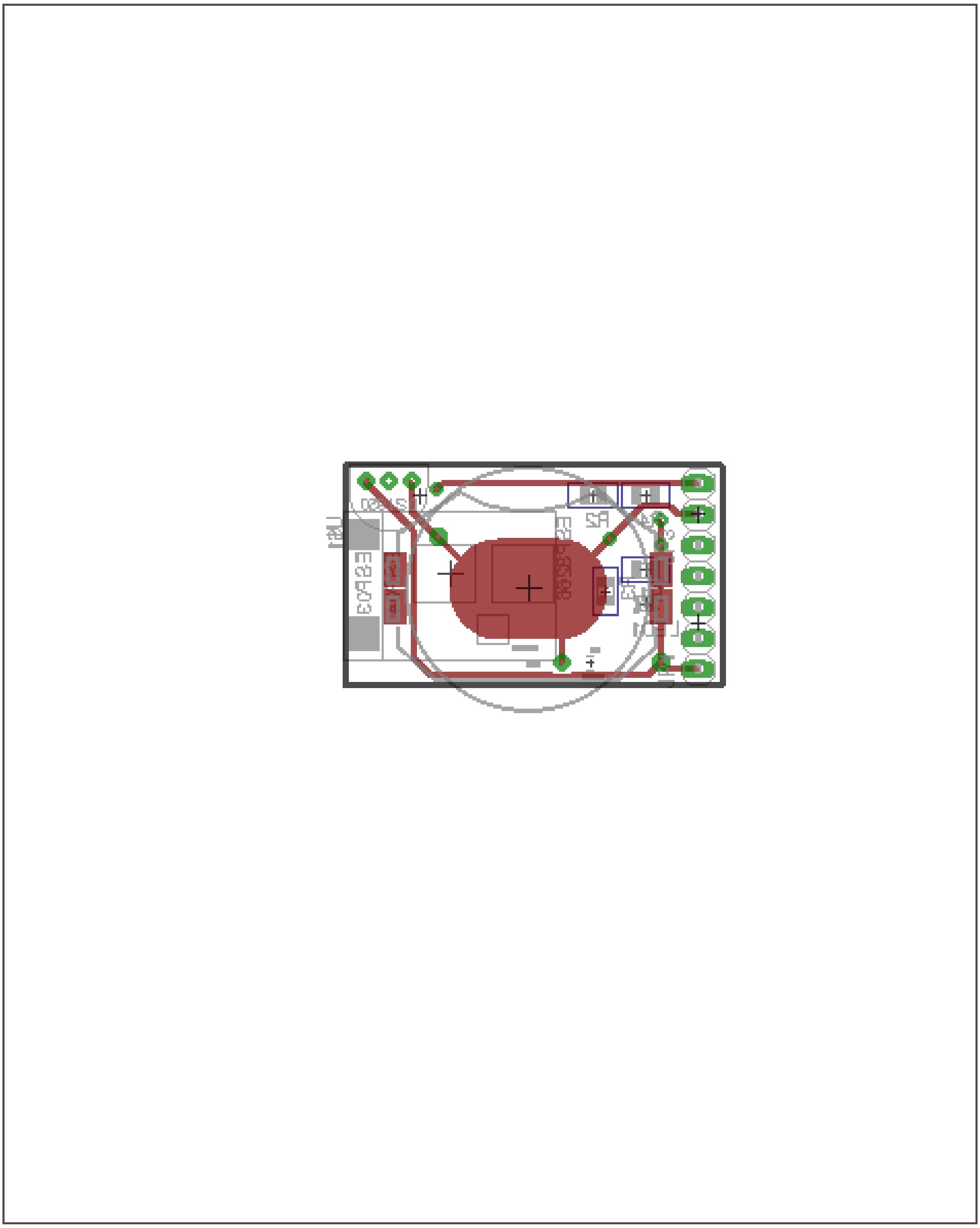

Figure A.2: Version One - Top Part of the Board

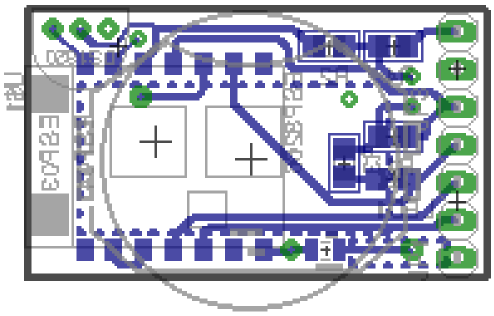

Figure A.3: Version One - Bottom Part of the Board

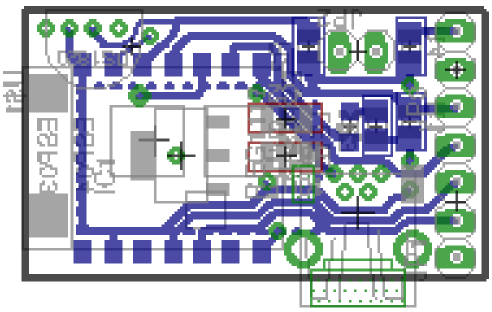

Figure A.4: Version Two - Bottom Part of the Board

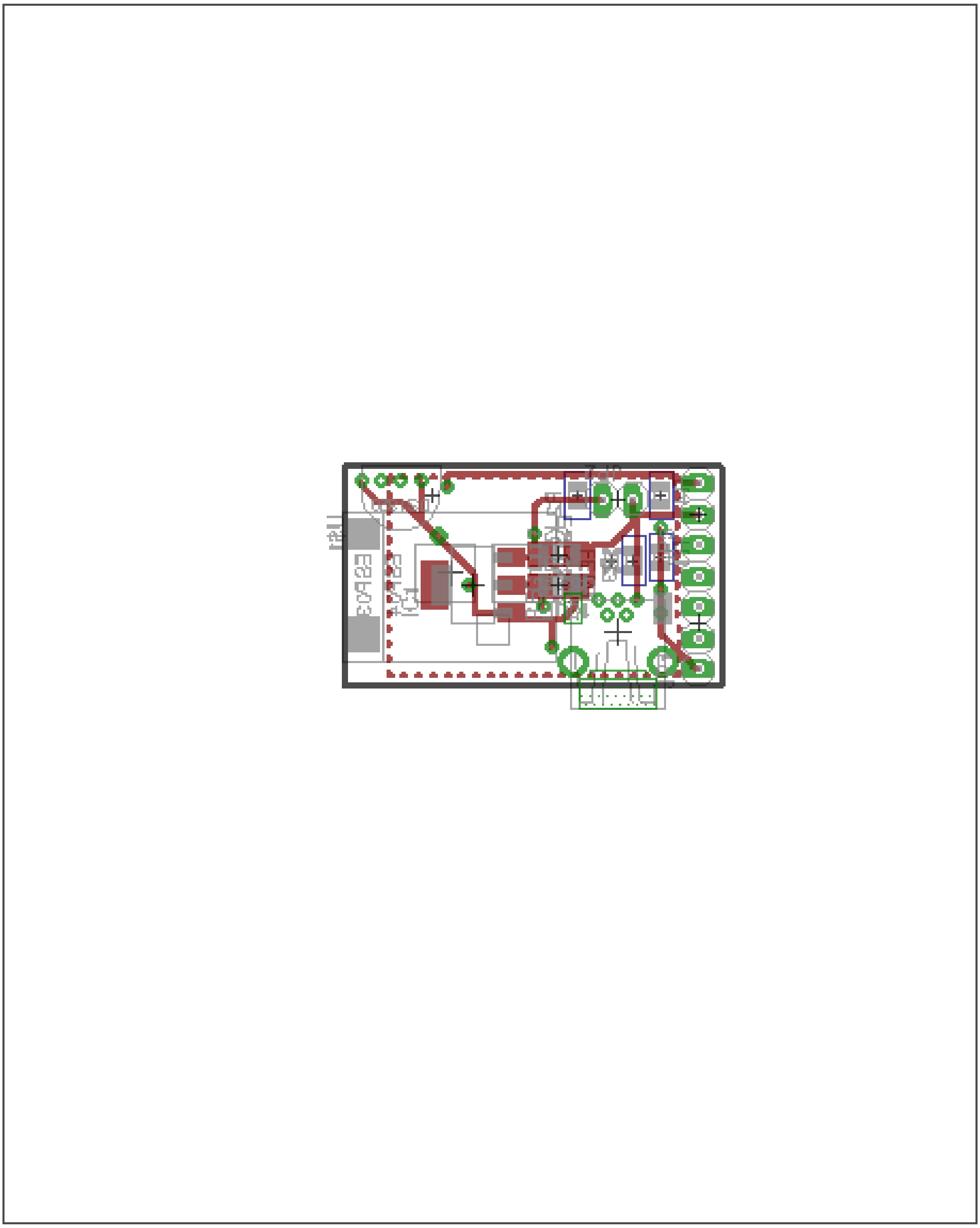

Figure A.5: Version Two - Top Part of the Board

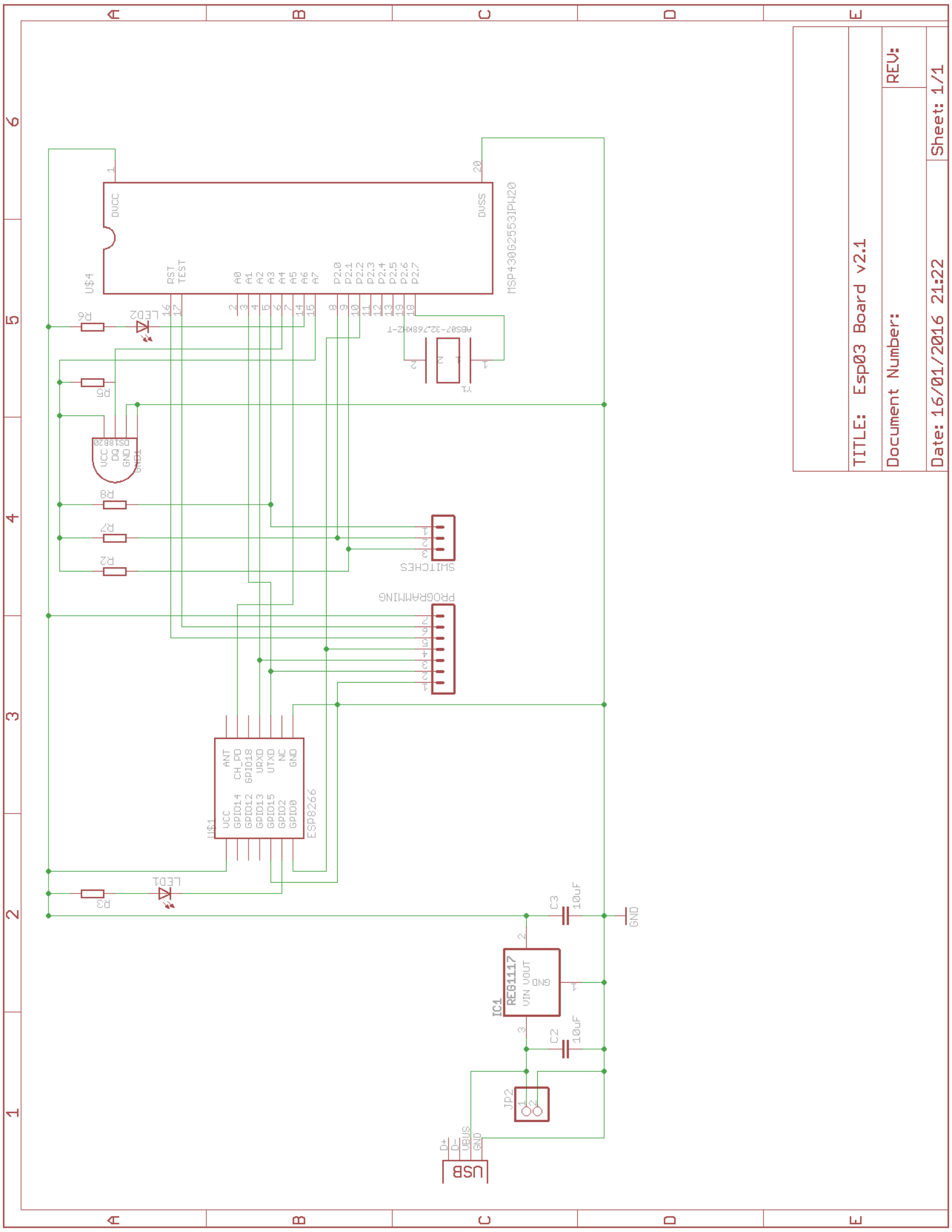

FIGURE A.6: MSP chip on ESP board schematic

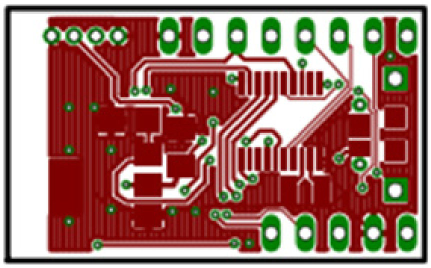

Figure A.7: Bottom Side of the Board

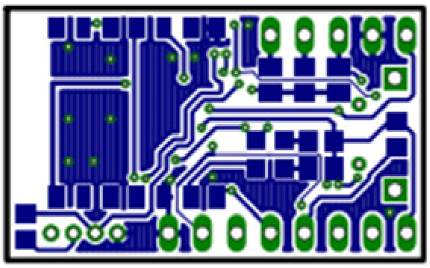

Figure A.8: Top Side of the Board

Bibliography

[1] Uniting smart home technologies: Who will rule the roost? http://icon.asid.org/index.php/2014/11/04/uniting-smart-home-technologies-who-will-rule-the-roost/. Accessed on : 2016.04.10.

[2] LTE Network Architecture Diagram. http://www.rcrwireless.com/20140513/network-infrastructure/lte/lte-network-architecture-diagram. Accessed on: 2015.10.15.

[3] 3GPP LTE - The Mobile Broadband Standard. http://www.3gpp.org/technologies/keywords-acronyms/98-lte. Accessed on: 2014.08.04.

[4] C. Perera, P. Jayaraman, A. Zaslavsky, P. Christen, and D. Georgakopoulos. Dynamic configuration of sensors using mobile sensor hub in internet of things paradigm. In *Intelligent Sensors, Sensor Networks and Information Processing, 2013 IEEE Eighth International Conference on*, pages 473–478, April 2013. doi: 10.1109/ISSNIP.2013.6529836.

[5] Shashank Sharma. System on a Chip: what you need to know about SoCs. http://www.techradar.com/news/computing/pc/system-on-a-chip-what-you-need-to-know-about-socs-1147235. Accessed on: 2017.02.08.

[6] Gartner Says 6.4 Billion Connected "Things" Will Be in Use in 2016, Up 30 Percent From 2015. http://www.gartner.com/newsroom/id/3165317. Accessed on: 2016.04.10.

[7] Radio Electronics. IEEE 802.11b. www.radioelectronics.com/info/wireless/wi-fi. Accessed on: 2016.03.10.

[8] Nikos Vokas. Connecting devices to the Internet of Things with Wi-Fi. http://embedded-computing.com/articles/connecting-devices-to-the-internet-of-things-with-wi-fi/. Accessed on: 2017.02.08.

[9] Wireless router (WiFi) penetration in the United Kingdom (UK) from 1st quarter 2007 to 1st wave 2015. https://www.statista.com/statistics/272754/wireless-router-take-up-in-the-united-kingdom-uk/. Accessed on: 2017.02.23.

[10] CC3000 Basic Wi-Fi example application for MSP430. http://processors.wiki.ti.com/index.php/CC3000. Accessed on:2016.05.10.

[11] Using 433 MHz for wireless connectivity in the Internet of Things. http://www.edn.com/electronics-blogs/eye-on-iot-/4437311/Using-433-MHz-for-wireless-connectivity-in-the-Internet-of-Things. Accessed on: 2017.02.08.

[12] J. Decuir. Introducing Bluetooth Smart: Part 1: A look at both classic and new technologies. *Consumer Electronics Magazine, IEEE*, 3(1):12–18, Jan 2014. ISSN 2162-2248. doi: 10.1109/MCE.2013.2284932.

[13] J. Decuir. Introducing Bluetooth Smart: Part II: Applications and updates. *Consumer Electronics Magazine, IEEE*, 3(2):25–29, April 2014. ISSN 2162-2248. doi: 10.1109/MCE.2013.2297617.

[14] Brian O'Reilly. Smart Sensor EnergyEGG). http://energy-egg.com/our-story/. Accessed on: 2017.02.11.

[15] Robert Pepper. The Internet of Things is Now: M2M Devices Forecast 2013-2018. www.iicom.org/.../555-machine-to-machine-devices-forecast-2013-2018. Accessed on: 2015.06.20.

[16] D. Thomas, R. McPherson, and J.Irvine. Power analysis of local transmission technologies. In *12th IEEE Conference on PhD Research in Microelectronics and Electronics (PRIME)*, 2016.

[17] D. Thomas, R. McPherson, G. Paul, and J. Irvine. Optimizing power consumption of Wi-Fi for IoT Devices: An MSP430 processor and an ESP-03 chip provide a power-efficient solution. *IEEE Consumer Electronics Magazine*, 5(4):92–100, Oct 2016. ISSN 2162-2248. doi: 10.1109/MCE.2016.2590148.

[18] D. Thomas, G.Paul, and J.Irvine. Going beyond the user - the challenges of universal connectivity in IoT. In *Wireless World Research Forum Meeting (WWRF35)*, 2015.

[19] D. Thomas and J. Irvine. Connection and resource allocation of IoT sensors to cellular technology-LTE. In *Ph.D. Research in Microelectronics and Electronics (PRIME), 2015 11th Conference on*, pages 365–368, June 2015. doi: 10.1109/PRIME.2015.7251411.

[20] Wen-Yen K. Lin. Smart Homes Promote Convenience,Energy Savings, Safety, and Wellness). http://www.intel.co.uk/content/dam/www/public/emea/xe/en/pdf/emea-iot-smart-home-ecs-tatung-brief.pdf. Accessed on: 2017.02.11.

[21] Linda. 10 Smart Medical Devices That Are Changing HealthCare 2016. http://www.appcessories.co.uk/connected-smart-medical-devices-that-are-changing-healthcare/. Accessed on: 2017.02.23.

[22] Harikrishnan Unnikrishnan. Five Plug and Play IoT Devices For Your Smarthome. http://electronicsofthings.com/news/five-plug-and-play-iot-devices-for-your-smarthome/. Accessed on: 2017.02.11.

[23] Control and monitor your home from your Android Smartphone or Tablet. http://www.smarthome.com/android-apps.html. Accessed on: 2017.02.24.

[24] Rick Delgado. Top 10 Benefits of Automating your Home. http://freshome.com/2013/01/17/top-10-benefits-of-automating-your-home/. Accessed on: 2017.02.08.

[25] Jim Fulker. Ease of Use: What the Smart Home Needs for Mass Market Adoption). https://www.icontrol.com/blog/ease-of-use-what-the-smart-home-needs-for-mass-market-adoption/. Accessed on: 2017.02.11.

[26] John Raspin. Smart and Connected Homes: Trends and Opportunities). https://www.tekes.fi/globalassets/global/ohjelmat-ja-palvelut/ohjelmat/fiksu-kaupunki/john-raspin-presentation-tekes-connected-homes-190416-pdf.pdf. Accessed on: 2017.02.11.

[27] S. Schultz. Wireless goes process automation - challenges in hazardous areas. In *Electrical and Instrumentation Applications in the Petroleum Chemical Industry, 2007. PCIC Europe 2007. 4th European Conference on*, pages 1–7, June 2007. doi: 10.1109/PCICEUROPE.2007.4354007.

[28] Zigbee Alliance. What is Zigbee. http://www.zigbee.org/what-is-zigbee/. Accessed on: 2016.08.29.

[29] Tarun Agarwal. ZigBee Wireless Technology Architecture and Applications. https://www.elprocus.com/what-is-zigbee-technology-architecture-and-its-applications/. Accessed on: 2017.02.08.

[30] K. H. Chang. Bluetooth: a viable solution for IoT? [industry perspectives]. *IEEE Wireless Communications*, 21(6):6–7, December 2014. ISSN 1536-1284. doi: 10.1109/MWC.2014.7000963.

[31] Bluetooth SIG. https://www.bluetooth.com/what-is-bluetooth-technology. Accessed on: 2017.02.08.

[32] Jacob Andrew. WiFi Transfer Rates and Types. http://techin.oureverydaylife.com/wifi-transfer-rates-types-21881.html. Accessed on: 2016.08.29.

[33] Gil Reiter. Wireless connectivity for the Internet of Things. http://www.ti.com.cn/cn/lit/wp/swry010/swry010.pdf. Accessed on: 2015.06.20.

[34] Wi-Fi Halow. http://www.wi-fi.org/discover-wi-fi/wi-fi-halow. Accessed on: 2016.05.10.

[35] IEEE 802.11ah Wi-Fi HaLow Emerges for IoT . http://www.enterprisenetworkingplanet.com/netsp/ieee-802.11ah-wi-fi-halow-emerges-for-iot.html. Accessed on: 2016.05.10.

[36] Todor Cooklev. *Wireless Communication Standards: A Study of IEEE 802.11, 802.15, 802.16*. IEEE Standards Association, 2004.

[37] Margaret Rouse. Direct sequence spread spectrum (DSSS) or direct sequence code division multiple access (DS-CDMA). http://searchnetworking.techtarget.com/definition/direct-sequence-spread-spectrum. Accessed on: 2017.02.09.

[38] Kenneth MacDonald. *Barker sequences theory and applications*. ProQuest, 2009.

[39] Mathuranathan. BPSK modulation and Demodulation). http://www.gaussianwaves.com/2010/04/bpsk-modulation-and-demodulation-2/. Accessed on: 2017.02.09.

[40] Robert Keim. Understanding Quadrature Phase Shift Keying (QPSK) Modulation). http://www.allaboutcircuits.com/technical-articles/quadrature-phase-shift-keying-qpsk-modulation/. Accessed on: 2017.02.09.

[41] Study and Analysis of wireless networking 802.11n protocol, 2011. http://eureka.lib.teithe.gr:8080/bitstream/handle/10184/11063/Chiotakis-Konstantinos.pdf?sequence1. Accessed on: 2015.10.02.

[42] Ian Poole. IEEE 802.11n Standard), . http://www.radio-electronics.com/info/wireless/wi-fi/ieee-802-11n.php/. Accessed on: 2017.02.09.

[43] Mark Julier. 802.11N Transmit Beamforming TxBF - Explained. http://www.digitalairwireless.com/wireless-blog/t-80211n/transmit-beamforming-txbf-explained.html. Accessed on: 2017.02.11.

[44] 802.11N Surviva Guide. https://www.safaribooksonline.com/library/view/80211n-a-survival/9781449335472/ch04.html. Accessed on: 2017.02.11.

[45] Robert G Gallagher. Low-Density-Parity-Check-Codes. http://www.rle.mit.edu/rgallager/documents/ldpc.pdf. Accessed on: 2017.02.11.

[46] Alcatel Lucent. The LTE Network Architecture. http://www.cse.unt.edu/rdantu/FALL-2013-WIRELESS-NETWORKS/LTE-Alcatel-White-Paper.pdf. Accessed on: 2015.08.05.

[47] Telesystem Innovations. LTE in a Nutshell. *White Paper*, 2010.

[48] Hyung G Myung. Technical overview of 3GPP LTE. *Polytechnic University of New York*, 2008.

[49] Ville Eerola. LTE Network Architecture Evolution. Research Paper, Helsinki University of Technology, 2009.

[50] Tim Godfrey. Long-Term Evolution Protocol: How the Standard Impacts Media Access Control,Freescale Semiconductor Accessed on: 2014.10.07.

[51] Keisuke Suzuki, Takashi Morita, Shin Naraha, and Zhen Miao. Core Network (EPC) for LTE. *NTT DOCOMO Technical Journal*, 13(1):26–32, 2011.

[52] 3GPP. The Evolved Packet Core, . http://www.3gpp.org/technologies/keywords-acronyms/100-the-evolved-packet-core. Accessed on: 2015.10.15.

[53] 3GPP LTE Packet Data Convergence Protocol (PDCP) Sub Layer, 2009. http://www.eventhelix.com/lte/presentations/3GPP-LTE-PDCP.pdf. Accessed on: 2015.11.20.

[54] 3GPP. 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Packet Data Convergence Protocol (PDCP) Specification, . ftp://www.3gpp.org/tsg-ran/TSG-RAN/TSGR-06/Docs/Pdfs/rp-99645.pdf. Accessed on: 2015.10.15.

[55] TS ETSI. 136 101 V10. 5.0 Release 10. January 2012. *LTE Evolved Universal Terrestrial Radio Access (E-UTRA): User Equipment (UE) radio transmission and reception*.

[56] Kan Zheng, Fanglong Hu, Wenbo Wang, Wei Xiang, and Mischa Dohler. Radio resource allocation in LTE-advanced cellular networks with M2M communications. *Communications Magazine, IEEE*, 50(7):184–192, 2012.

[57] Morten Tolstrup. *Indoor Radio Planning: A Practical Guide for Gsm, Dcs, Umts, Hspa and LTE*. John Wiley & Sons, 2011.

[58] Hyung G Myung, Junsung Lim, and David J Goodman. Single carrier fdma for uplink wireless transmission. *IEEE Vehicular Technology Magazine*, 1 (3):30–38, 2006.

[59] Ian Poole. LTE Physical, Logical and Transport Channels, . http://www.radio-electronics.com/info/cellulartelecomms/lte-long-term-evolution/physical-logical-transport-channels.php. Accessed on: 2017.02.17.

[60] Jim Zyren. Overview of the 3gpp long term evolution physical layer.

[61] N. Abu-Ali, A. E. M. Taha, M. Salah, and H. Hassanein. Uplink scheduling in lte and lte-advanced: Tutorial, survey and evaluation framework. *IEEE Communications Surveys Tutorials*, 16(3):1239–1265, Third 2014. ISSN 1553-877X. doi: 10.1109/SURV.2013.1127.00161.

[62] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, and P. Camarda. Downlink packet scheduling in lte cellular networks: Key design issues and a survey. *IEEE Communications Surveys Tutorials*, 15(2):678–700, Second 2013. ISSN 1553-877X. doi: 10.1109/SURV.2012.060912.00100.

[63] D. Jiang, H. Wang, E. Malkamaki, and E. Tuomaala. Principle and performance of semi-persistent scheduling for voip in lte system. In *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, pages 2861–2864, Sept 2007. doi: 10.1109/WICOM.2007.710.

[64] NarrowBand - Internet of Things (NB-IoT). http://www.gsma.com/connectedliving/narrow-band-internet-of-things-nb-iot/. Accessed on: 2016.06.10.

[65] R. Ratasuk, N. Mangalvedhe, Y. Zhang, M. Robert, and J. P. Koskinen. Overview of narrowband iot in lte rel-13. In *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 1–7, Oct 2016. doi: 10.1109/CSCN.2016.7785170.

[66] N. Mangalvedhe, R. Ratasuk, and A. Ghosh. Nb-iot deployment study for low power wide area cellular iot. In *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–6, Sept 2016. doi: 10.1109/PIMRC.2016.7794567.

[67] J. Gozalvez. New 3gpp standard for iot [mobile radio]. *IEEE Vehicular Technology Magazine*, 11(1):14–20, March 2016. ISSN 1556-6072. doi: 10.1109/MVT.2015.2512358.

[68] NB-IOT:A SUSTAINABLE TECHNOLOGY FOR CONNECTING BILLIONS OF DEVICES. https://www.ericsson.com/assets/local/publications/ericsson-technology-review/docs/2016/etr-narrowband-iot.pdf. Accessed on: 2017.02.23.

[69] 3GPP NB-IoT. http://www.3gpp.org/news-events/3gpp-news/1785-nbiotcomplete. Accessed on: 2016.06.10.

[70] Dongmei Yan and Zhiguang Dan. Zigbee-based smart home system design. In *2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)*, volume 2, pages V2–650–V2–653, Aug 2010. doi: 10.1109/ICACTE.2010.5579732.

[71] K. Nair, J. Kulkarni, M. Warde, Z. Dave, V. Rawalgaonkar, G. Gore, and J. Joshi. Optimizing power consumption in IoT based wireless sensor networks using Bluetooth Low Energy. In *Green Computing and Internet of Things (ICGCIoT), 2015 International Conference on*, pages 589–593, Oct 2015. doi: 10.1109/ICGCIoT.2015.7380533.

[72] A. Dementyev, S. Hodges, S. Taylor, and J. Smith. Power consumption analysis of bluetooth low energy, zigbee and ant sensor nodes in a cyclic sleep scenario. In *Wireless Symposium (IWS), 2013 IEEE International*, pages 1–4, April 2013. doi: 10.1109/IEEE-IWS.2013.6616827.

[73] M. D. Prieto, B. Martinez, M. Montn, I. V. Guillen, X. V. Guillen, and J. A. Moreno. Balancing power consumption in iot devices by using variable packet size. In *Complex, Intelligent and Software Intensive Systems (CISIS), 2014 Eighth International Conference on*, pages 170–176, July 2014. doi: 10.1109/CISIS.2014.25.

[74] Rajeev Piyare and Seong Ro Lee. Smart home-control and monitoring system using smart phone. In *International Conference on Convergence and its Applications (ICCA)*, 2013.

[75] Ramesh Rajagopalan and Pramod K Varshney. Data aggregation techniques in sensor networks: A survey. In *Electrical Engineering and Computer Science Paper 22*, 2006.

[76] A. Sharma and G. S. Tewolde. Considerations in low power wireless sensor networks. In *2015 IEEE International Conference on Electro/Information Technology (EIT)*, pages 626–631, May 2015. doi: 10.1109/EIT.2015.7293408.

[77] K. Sharma and T. Suryakanthi. Smart system: IoT for university. In *Green Computing and Internet of Things (ICGCIoT), 2015 International Conference on*, pages 1586–1593, Oct 2015. doi: 10.1109/ICGCIoT.2015.7380720.

[78] K. S. Yeo, M. C. Chian, T. C. Wee Ng, and D. A. Tuan. Internet of things: Trends, challenges and applications. In *2014 International Symposium on Integrated Circuits (ISIC)*, pages 568–571, Dec 2014. doi: 10.1109/ISICIR.2014.7029523.

[79] S. Saloni and A. Hegde. Wifi-aware as a connectivity solution for iot pairing iot with wifi aware technology: Enabling new proximity based services. In *2016 International Conference on Internet of Things and Applications (IOTA)*, pages 137–142, Jan 2016. doi: 10.1109/IOTA.2016.7562710.

[80] N. K. Walia, P. Kalra, and D. Mehrotra. An iot by information retrieval approach: Smart lights controlled using wifi. In *2016 6th International Conference - Cloud System and Big Data Engineering (Confluence)*, pages 708–712, Jan 2016. doi: 10.1109/CONFLUENCE.2016.7508211.

[81] R. K. Kodali, S. Soratkal, and L. Boppana. Iot based control of appliances. In *2016 International Conference on Computing, Communication and Automation (ICCCA)*, pages 1293–1297, April 2016. doi: 10.1109/CCAA.2016.7813918.

[82] R. K. Deore, V. R. Sonawane, and P. H. Satpute. Internet of thing based home appliances control. In *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, pages 898–902, Dec 2015. doi: 10.1109/CICN.2015.177.

[83] M Asim, M Yu, H Mokhtar, and M Merabti. A Self-Configurable Architecture for Wireless Sensor Networks. In *Developments in E-systems Engineering (DESE), 2010*, pages 76–81. IEEE, 2010.

[84] O. Delgado and B. Jaumard. Scheduling and resource allocation for multiclass services in lte uplink systems. In *2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications*, pages 355–360, Oct 2010. doi: 10.1109/WIMOB.2010.5645044.

[85] Sasan Adibi, Amin Mobasher, and Tom Tofigh. LTE networking: extending the reach for sensors in mHealth applications. *Transactions on Emerging Telecommunications Technologies*, 2013.

[86] Suresh Krishnan. *Connecting sensors to LTE*. PhD thesis, Concordia University, 2012.

[87] Jason Brown and Jamil Y Khan. Predictive resource allocation in the LTE uplink for event based M2M applications. In *Communications Workshops (ICC), 2013 IEEE International Conference on*, pages 95–100. IEEE, 2013.

[88] Sarah Perez. *The Number Of Mobile Devices Will Exceed Worlds Population By 2012*, February 2012. http://techcrunch.com/2012/02/14/the-number-of-mobile-devices-will-exceed-worlds-population-by-2012-other-shocking-figures/. Accessed on: 2014.08.18.

[89] Texas Instruments. *CC3000 IEEE 802.11b/g Solution Module*. http://www.ti.com/product/cc3000.

[90] Madan Lal Rasneet Kaur. Wireless Sensor Networks: A Survey. http://www.iject.org/vol4/spl3/c0118.pdf. Accessed on: 2017.02.17.

[91] Quazi Mamun. A qualitative comparison of different logical topologies for wireless sensor networks. *Sensors*, 12(11):14887–14913, 2012.

[92] Quazi Mamun. A qualitative comparison of different logical topologies for wireless sensor networks. *Sensors*, 12(11):14887–14913, 2012.

[93] M. M. Patel and A. Aggarwal. Security attacks in wireless sensor networks: A survey. In *2013 International Conference on Intelligent Systems and Signal Processing (ISSP)*, pages 329–333, March 2013. doi: 10.1109/ISSP.2013.6526929.

[94] Krishna Doddapaneni and Arindam Ghosh. Analysis of denial-of-service attacks on wireless sensor networks using simulation. *IT Security for the Next Generation-European Cup 2011*, 2011.

[95] Preeti Sharma, Monika Saluja, and Krishan Kumar Saluja. A review of selective forwarding attacks in wireless sensor networks. *International journal of Advances Smart Sensor Network Systems (IJASSN)*, 2(3), 2012.

[96] Kuo-Feng Ssu, Wei-Tong Wang, and Wen-Chung Chang. Detecting sybil attacks in wireless sensor networks using neighboring information. *Computer Networks*, 53(18):3042–3056, 2009.

[97] Teodor-Grigore Lupu, I Rudas, M Demiralp, and N Mastorakis. Main types of attacks in wireless sensor networks. In *WSEAS International Conference. Proceedings. Recent Advances in Computer Engineering*, number 9. WSEAS, 2009.

[98] Roke Manor. LTE MAC Scheduler & Radio Resource Scheduling. http://www.docstoc.com/docs/122980686/LTE-MAC-Scheduler-and-Radio-Resource-Scheduling. Accessed on: 2014.09.16.

[99] Spark. *Spark Cloud API.* http://docs.spark.io/api/. Accessed on: 2015.01.29.

[100] *SimuLTE LTE User Plane Simulation Model for INET OMNeT++* . https://github.com/inet-framework/simulte. Accessed on: 2015.01.29.

[101] G. Nardini A. Virdis, G. Stea. SimuLTE A Modular System-level Simulator for LTE/LTE-A Networks based on OMNeT++. 2014.

[102] Andreas Köpke, Michael Swigulski, Karl Wessel, Daniel Willkomm, PT Haneveld, Tom EV Parker, Otto W Visser, Hermann S Lichte, and Stefan Valentin. Simulating wireless and mobile networks in omnet++ the mixim vision. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 71. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[103] C Mallanda, A Suri, V Kunchakarra, SS Iyengar, R Kannan, A Durresi, and S Sastry. Simulating wireless sensor networks with omnet++. *submitted to IEEE Computer*, 2005.

[104] Datasheet. https://datasheets.maximintegrated.com/en/ds/DS18S20.pdf. Accessed on:2015.11.28.

[105] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Comput. Netw.*, 54(15):2787–2805, October 2010. ISSN 1389-1286. doi: 10.1016/j.comnet.2010.05.010. http://dx.doi.org/10.1016/j.comnet.2010.05.010.

[106] What is C-rate?, 2003. http://batteryuniversity.com/learn/article.what-is-the-c-rate. Accessed on: 2015.10.06.

[107] LS14250 Lithium Thionyl Chloride Battery. http://uk.farnell.com/saft/ls14250/battery-lithium-1-2aa-standard/dp/1282760. Accessed on: 2016.10.15.

[108] Portapow. http://www.portablepowersupplies.co.uk/portapow-premium-usb-dc-power-monitor/. Accessed on: 2016.04.10.

[109] Mark Schwartz. Making weather station with Arduino and CC3000 chip. https://openhomeautomation.net/arduino-wifi-cc3000/. Accessed on: 2017.02.11.

[110] Mike Barela. IoT Wi-Fi Modules Comparison Table ESP8266 vs CC3000 vs RN131 vs HDG204. http://www.cnx-software.com/2015/06/15/iot-wi-fi-modules-comparison-table-esp8622-vs-cc3300-vs-rn131-vs-hdg204/. Accessed on: 2017.02.11.

[111] Texas Instruments WiFi Application, 2015. http://processors.wiki.ti.com/index.php/CC3000-Basic-Wi-Fi-example-application-for-MSP430. Accessed on: 2015.06.21.

[112] Sparkfun Tutorials, 2016. https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi. Accessed on: 2015.06.21.

[113] Texas Instruments Smart Config, 2016. http://processors.wiki.ti.com/index.php/CC3000-Smart-Config. Accessed on: 2015.06.21.

[114] Mixed Signal Microcontroller. http://www.ti.com/lit/ds/symlink/msp430g2553.pdf. Accessed on: 2016.04.01.

[115] Y. Jie, J. Y. Pei, L. Jun, G. Yun, and X. Wei. Smart home system based on iot technologies. In *Computational and Information Sciences (ICCIS), 2013 Fifth International Conference on*, pages 1789–1791, June 2013. doi: 10.1109/ICCIS.2013.468.

[116] M. J. Kaur and P. Maheshwari. Building smart cities applications using iot and cloud-based architectures. In *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, pages 1–5, March 2016. doi: 10.1109/ICCSII.2016.7462433.

[117] Lattice Semiconductor. Energy Efficiency: The Common Denominator in the Internet of Things. http://www.latticesemi.com/ /media/LatticeSemi/-Documents/WhitePapers/HM/LatticeSemiconductorIoTWhitePaper.pdf.

[118] C. Gray, R. Ayre, K. Hinton, and R. S. Tucker. Power consumption of IoT access network technologies. In *2015 IEEE International Conference on Communication Workshop (ICCW)*, pages 2818–2823, June 2015. doi: 10.1109/ICCW.2015.7247606.

[119] The Secrets of Battery Runtime. http://batteryuniversity.com/ Accessed on: 2016.09.27.

[120] Insecure Internet of Things. https://www.toptal.com/it/are-we-creating-an-insecure-internet-of-things Accessed on: 2016.09.27.