THESIS

ANALYSIS AND CHARACTERIZATION OF WIRELESS SMART POWER METER

Submitted by

Sachin Soman

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Summer 2014

Master's Committee:

Advisor: Peter Young

Daniel Zimmerle

Sudeep Pasricha

ABSTRACT

ANALYSIS AND CHARACTERIZATION OF WIRELESS SMART POWER METER

Recent increases in the demand for and price of electricity has stimulated interest in monitoring energy usage and improving efficiency. This research work supports development of a low-cost wireless smart power meter capable of measuring V_{RMS}, I_{RMS}, real power, and reactive power. The proposed smart power meter features include matching by-device rate of consumption and usage patterns to assist users in monitoring the connected devices. The meter also includes condition monitoring to detect harmonics of interest in the connected circuits which can give vital clues about the defects in machines connected to the circuits. This research work focuses on estimating communicational and computational requirements of the smart power meter and optimization of the system based on the estimated communication and computational requirements.

The wireless communication capabilities investigated here are limited to existing wireless technologies in the environment where the power meters will be deployed. Field tests are performed to measure the performance of selected wireless standard in the deployment environment. The test results are used to understand the distance over which the smart power meters can communicate and where it is necessary to utilize repeaters or range extenders to reduce the data loss.

Computational requirements included analysis of smart meter front-end sampling of analog data from both current and voltage sensors. Digitized samples stored in a buffer which is further processed by a microcontroller for all the desired results from the power meter. The various stages for processing the data require computational bandwidth and memory dependent on the

size of the data stream and calculations involved in the particular stage. A Simulink-based system model of the power meter was developed to report a statistic of computational bandwidth demanded by each stage of data processing.

The developed smart meter works in an environment with other wireless devices which include Wi-Fi and Bluetooth. The data loss caused when the smart power meter transmits the data depends on the architecture of the wireless network and also pre-existing wireless technology working in the same environment and while operating in the same frequency band. The best approach in developing a wireless network should reduce the hardware cost of the network and to reduce the data loss in the wireless network. A wireless sensor network is simulated in OMNET++ platform to measure the performance of wireless standard used in smart power meters. Scenarios involving the number of routers in the network and varying throughput between devices are considered to measure the performance of wireless power meters.

Supplementary documents provided with the electronic version of this thesis contain program codes which were developed in Simulink and OMNET++.

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank all the people who helped me to complete this project. First and foremost, I would like to express my sincere gratitude to my advisor-Dr. Peter Young for his continuous support in my research. Next, I would like to thank Professor Daniel Zimmerle for his advice and encouragement throughout the project. His expertise, understanding and patience added considerably to my experience in graduate school. I would also like to acknowledge Dr. Sudeep Pasricha for his role as a member of my defense examination committee. Additionally, I would like to thank Jerry Duggan for his suggestions on various phases of the project and I wish to thank all the staff in Engines and Energy Conversion Laboratory for their kind help during various stages of my graduate program.

I would also like to acknowledge the financial support for this work provided by Colorado State University *Clean Energy Supercluster* grant, "Development of Low-Cost Conditioning Monitoring using Wireless Microcontroller Electricity Measurements," and support provided by InGreenium, LLC under Department of Energy grant DE-SC0009210 from program *SBIR/STTR FY2012 Phase 1 (Release 3) Application Topic 3h*.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1 Overview

This chapter introduces the importance of measuring power and surveys electric power meters available in the current market. This is followed by proposed implementation and features of the smart power meter. The computational and communication bandwidth required for the proposed smart meter are also estimated. The final section of this chapter outlines the remaining sections of this thesis.

1.1 Need for Power Metering

Electricity pricing can motivate organizations to reduce energy use. The key to reducing electricity bills is to provide customers with a better understanding of when and where electricity is consumed, a key benefit of smart power meters. The energy usage data from these devices can also be used to improve the facility and limit the loss of energy [1.1]. The installation of power meters helps residential customers to have an understanding of energy usage in home. Wireless displays in these power meters can display real time costs of energy usage and can help customers to budget the energy usage [1.5].

Organizations can choose among many ways to reduce energy costs. Using power meters to detect power transients occurring in an event of start or stop of a device is relevant in this study of smart power meters. Start and stop information is used to calculate the operation time and duration. This data can be used in for scheduling of heating, ventilation and air conditioning (HVAC) to optimize energy usage [1.3] of the organization.

Beyond energy cost savings, power monitoring can enhance operational effectiveness by enhancing equipment maintenance capabilities. The energy consumption history of generators

and motors can be used to predict or detect the faults in the machine. The faults introduce identifiable harmonic components in the stator current of machines [1.6]. When detected, these harmonic components can give vital clues about the type of fault in the machine which includes bearing, eccentricity and gearbox faults. This information can help diagnose failures to speed repairs or to proactively manage maintenance to avoid downtime.

1.2 Types of Power Meters

Power meters are available in various configurations ranging from simple energy meters to complex meters which support power quality analysis and remote power management functions. Power meters can be broadly classified based on their functionality as described in this section.

1.2.1 Utility Power Meters

Utility power meters are installed in residential, commercial or industrial electric systems by utility companies for the primary purpose of billing. Recent innovations include built-in communication schemes to integrate with a central monitoring system. The communication schemes of these power meters include Power Line Communications (PLC), MODBUS over IP adapters etc. [1.7]. A utility power meter also detects sags, swells, transients and flickers in the electric line. The advanced version of utility power meters by Quadlogic [1.7] measures advanced metrics which includes power factor per phase, Total Harmonic Distortion (THD), voltage transients and current transients. Smart meters used by utilities cost between $250 and $500 [1.9] which does not include labor costs.

1.2.2 Analog Meters

The Analog power meter works on the principle of electro-mechanic induction. This meter operates by making an electrically conductive metal disc rotate at a speed proportional to

the power flowing through the meter. Typically, these traditional utility meters are not equipped with communications technology and the utility company sends an employee to record the power flow at regular intervals. Analog meters are inaccurate because of power consumed by rotating coils and friction between the rotating coils.

1.2.3 Power Quality Meters

Power quality meters are the advanced type of power meters compared to the two classifications listed above. Power quality meters measures single phase or 3 phase power in kWh and have advanced features which includes measurement of harmonic distortions (harmonic analysis through as high as the 63^{rd} harmonic) and power quality metrics and can record user defined events [1.10]. These meters operate in wide voltage range. Power quality meters also have protection mechanisms to cope against voltage, current and frequency imbalances [1.10]. As an example, an Eaton Xpert Meter [1.8] has features which include a connection to a web server to analyze waveforms, trends and harmonics. Eaton Xpert meters are installed in large industries and are list- priced near $10,000.

1.2.4 Low Cost Power Meter

Low-cost power meters are priced in the range $50-$100 [1.20]. These basic power meters take in three phase 120/240V 3-wire 100 Amp line and displays real power in an integrated LCD display.

1.3 The Proposed Smart Meter

The proposed smart power meter measures V_{RMS}. I_{RMS}, real power and reactive power. The power measurements are used in advance features of the smart power meter which includes detecting harmonic components and tracking load time. These features are similar to those available in utility and power quality meters discussed in Section 1.2.1 and Section 1.2.3. The

proposed smart meter is similar to the Powerhouse Dynamics SiteSage and competing products, priced under $1000 [1.21] [1.22].

1.3.1 Subpanel Implementation

The proposed power meter is implemented in two configurations.

1.3.1.1 Configuration I

The proposed power meter is deployed in a subpanel and monitors 12 circuits in the subpanel.

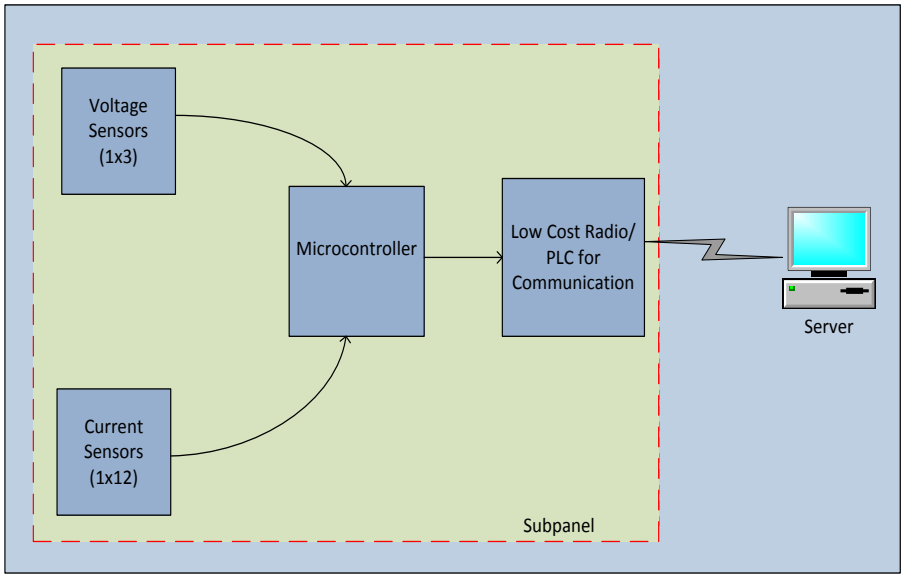

Figure 1.1: Configuration – I

Figure 1.1 shows the implementation of a power meter in a subpanel which has 12 circuits attached to it. The three phase voltage is common for all circuits and there are 12 lines to monitor the current. The calculation of power in these 12 circuits requires three phase voltage value and 12 values of current. Thus, a single power meter can perform all the promised functions in 12 lines. The current sensors are clip-on current transformers attached to the wire exiting the subpanel to each circuit from the subpanel. A power supply is also tapped from voltage sensors to power all the components of the smart power meter. In this implementation,

each subpanel system requires a microcontroller and radio to measure and report power consumption of subpanel circuits.

1.3.1.2 Configuration II

The Figure 1.2 shows the implementation of the proposed smart power meter in configuration II.

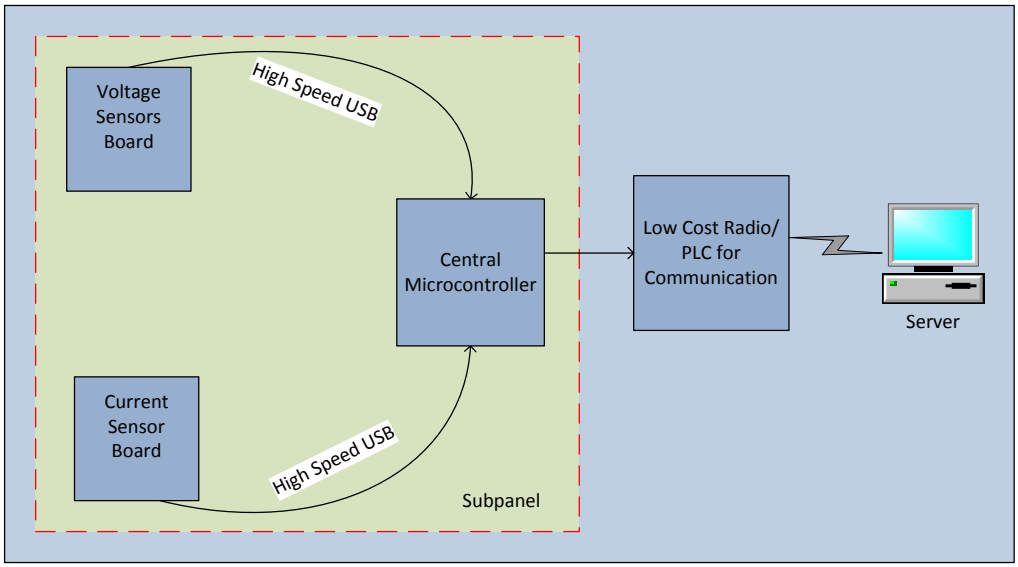

Figure 1.2: Configuration II

In this configuration voltage and current sensors are integrated into a board with a microcontroller. These boards are connected to a central microcontroller (common for a subpanel) via high speed USB connection. The central microcontroller is responsible for executing algorithms required to satisfy the promised functions of the smart power meter which are discussed in Section 1.4.2. This central microcontroller is high-power microcontroller with a dedicated USB memory to store data points from voltage and current sensor boards. The central microcontroller monitors up to 42 circuits.

1.4 Functions of Proposed Smart Power Meter

The smart meter has the following functions:

1.4.1 Root Mean Square (RMS) Measurement

The RMS measures 'heating' potential of a signal. The Smart power meter measures RMS of current and voltage signals. RMS measurement of continuous signal is as follows [1.12]:

$$V_{RMS} = \sqrt{\frac{1}{T_m}\int_0^{T_m} v^2\,(t)dt} \qquad 1$$

$$I_{RMS} = \sqrt{\frac{1}{T_m}\int_0^{T_m} i^2\,(t)dt} \qquad 2$$

v(t) and *i(t)* are the instantaneous values of voltage and current. T_m is the length of a single period of the measured analog signal.

RMS measurement of discrete signal is as follows [1.8]:

$$V_{RMS} = \sqrt{\frac{1}{N}\sum_N v^2\,(n)} \qquad 3$$

$$I_{RMS} = \sqrt{\frac{1}{N}\sum_N i^2\,(n)} \qquad 4$$

$v(n)$ and $i(n)$ are the corresponding discrete values of voltage and current. *N* is the number of samples in a single period of the discrete signal. The sampling frequency of current and voltage signals should be an integral multiple of the fundamental component (60 Hz) to avoid any chance of aliasing.

Figure 1.3 describes the algorithm for RMS calculation implemented in the microcontroller.

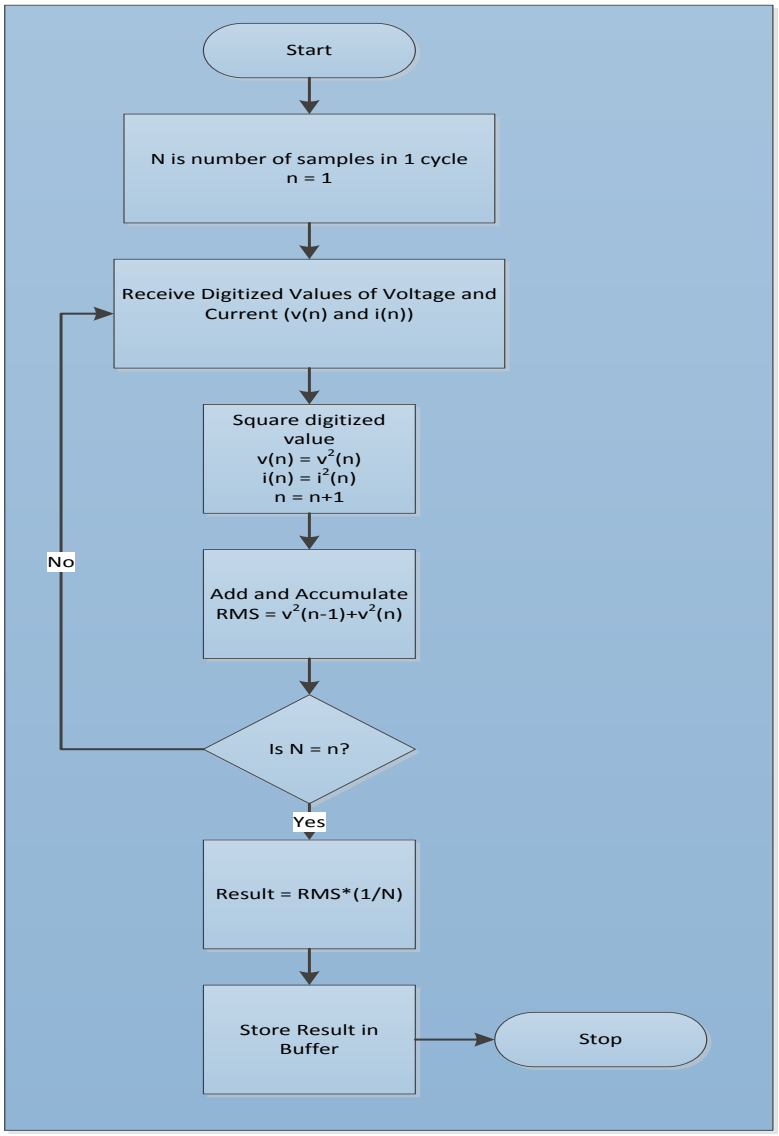

Figure 1.3: Algorithm for RMS Calculation

In the first step, instantaneous values of voltage and current are received from Analog to Digital Converters (ADC). These digital values are multiplied with itself (Squared). The resulting values are accumulated. The accumulated value is multiplied by a stored number (1/N) to produce the final result from microcontroller. This value is stored in buffer and is sent to the server where square root operation is performed. Square root operation is performed in the server

to reduce the computational steps performed for RMS calculation in the microcontroller. In this algorithm, calculation of a single RMS value requires N additions and 1 multiplication.

1.4.2 Real Power Measurement

The Real power or true power denoted by *P*, measured in watts and is calculated as follows [1.12]:

$$p = v \times i \tag{5}$$

Real power of a continuous time signal is measured as follows [1.12]:

$$p = \frac{1}{T_m}\int_0^{T_m} v\,(t) \times i(t)dt \tag{6}$$

$v(t)$ and $i(t)$ are instantaneous values of current and voltage. Real power is measured in discrete time signal as follows [1.12]:

$$p = \frac{1}{N}\sum_{N} v(n) \times i(n) \tag{7}$$

$v(n)$ and $i(n)$ are instantaneous discrete values of current and voltage. N is the number of samples in a single cycle of fundamental component (60 Hz). The sampling frequency of current and voltage signals should be integral multiples of the fundamental component to avoid any chance of aliasing.

Figure 1.4 is the algorithm for real power calculation. In the first step, instantaneous values of V and I are received and are multiplied to calculate instantaneous power. Instantaneous power values are accumulated for a single cycle of the fundamental component. The resulting value is multiplied by 1/N where N is number of samples in a single cycle of the fundamental component.

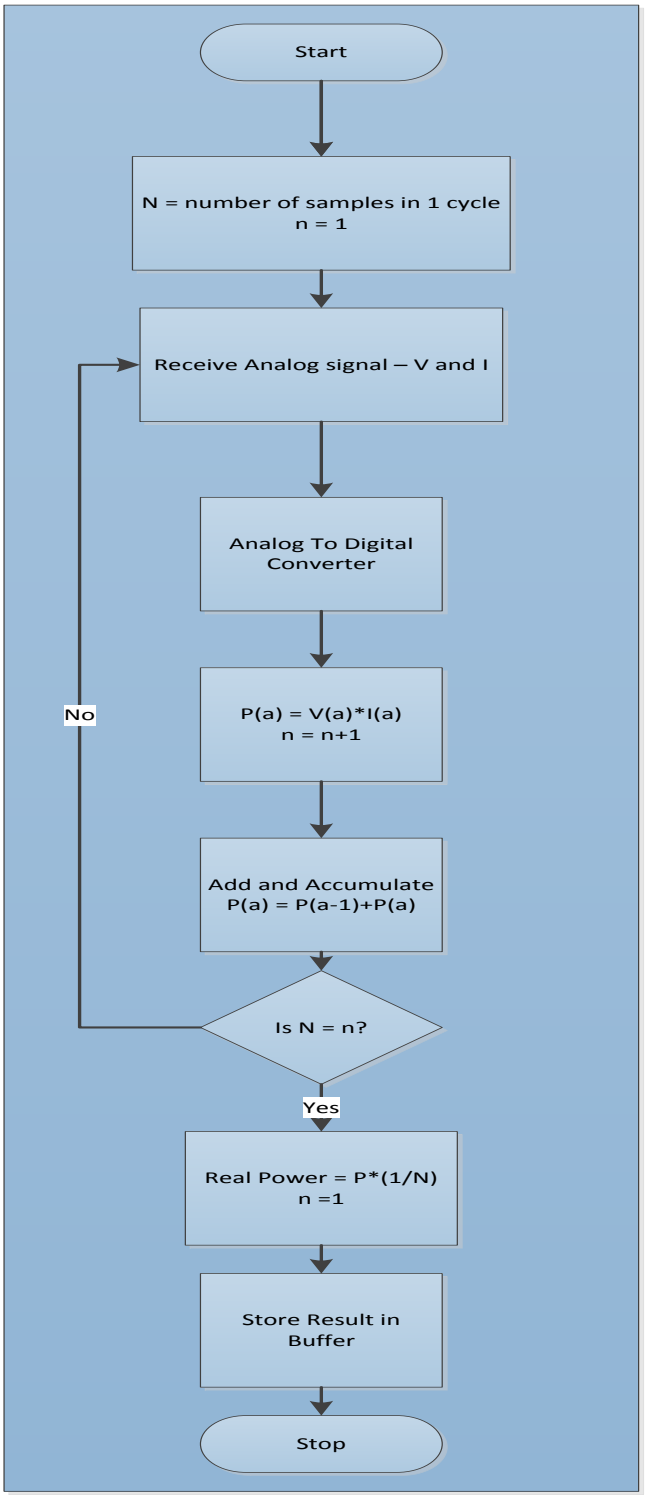

Figure 1.4: Algorithm for Real Power Calculation

1.4.3 Reactive Power Measurement

The magnitude of the reactive power (S) is calculated as follows [1.12]:

$$S = VIsin\theta \quad (8)$$

The Reactive power or apparent power measured in volt-amperes (VA) is calculated by implementing a time delay between voltage and current signals. This is based on the assumption that implementing a phase shift of $\frac{\pi}{4}$ in either v or i and multiplying it produces the same result as in equation 8. This method is accurate only if signals v and i contain only the fundamental component (60 Hz). The phase shift is implemented by shifting the voltage signal by a quarter cycle.

$$S = \frac{1}{T}\int_0^T v(t) \times i(t + \frac{T}{4})dt \quad (9)$$

The reactive power algorithm is implemented by delaying voltage signals by a pre-defined number of samples equal to quarter cycle of the fundamental component (60 Hz) of the current signal. If the data is sampled at 3600 Hz and the fundamental component (60 Hz), then one cycle contains 60 samples. Hence, a quarter cycle of the fundamental component contains 15 samples. In the implementation, a data array is used to store 15 samples of voltage signals. The current samples are sequentially multiplied by stored voltage samples, buffered, and then sent to the server.

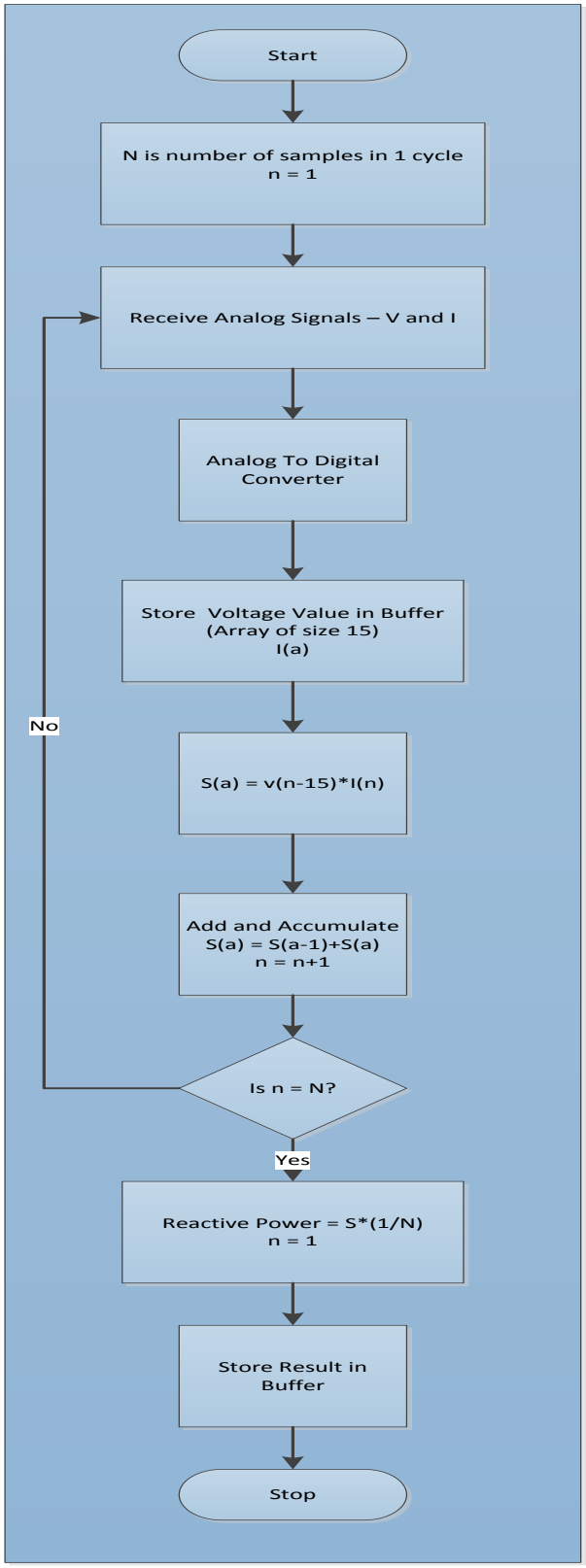

Figure 1.5: Algorithm for Reactive Power Calculation

Figure 1.5 describes the algorithm for reactive power calculation. In the first step, instantaneous values of current and voltage are received. The voltage value is multiplied by a delayed current value. The current value is delayed by quarter cycle of the fundamental component. The power values are accumulated for a single cycle of the fundamental component. The resulting value is multiplied by 1/N where N is the number of samples in a single cycle of the fundamental component.

1.4.4 Condition Monitoring/Frequency Analysis

The proposed power meter uses Discrete Fourier Transform (DFT) [1.10] to check the presence of harmonics. DFT is the mathematical transformation which gives both the amplitude and phase information of the desired frequency. A DFT is computed by using following equation:

$$X_h = \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi hn}{N}} \quad (10)$$

where:

N is the block size that controls the resolved frequency in DFT

x(n) is the signal at point n.

X_h is the Fourier vector of h^{th} harmonic of the input signal.

$$X_h = X_{hr} + jX_{hi} \quad (11)$$

where X_{hr} and X_{hi} are real and imaginary parts of X_h.

Amplitude of harmonic component $|X_h|$ is given by,

$$|X_h| = \sqrt{{X_{hr}}^2 + {X_{hi}}^2} \quad (12)$$

Phase of the harmonic component ϕ_n is given by,

$$\phi_n = arctan\frac{X_{hi}}{X_{hr}} \quad (13)$$

1.4.5 Pattern Matching

Electric machines are often expensive to repair after a failure [1.19]. If a machine failure can be detected early, the cost of downtime, service and repair can be reduced. One of the features of the proposed smart meter is to detect the event of start/stop of the load. Changes in load run times or power drawn during operation can be signs of machine failure and detecting these events can help to schedule proactive maintenance. This project utilizes algorithms for detecting start and stop events described in [1.19].

The key computation for pattern matching is the calculation of several high-order statistical measures which includes the bimodality coefficient.

$$Bimoadility\ coefficient = \frac{skew^2 + 1}{kurtosis + \frac{3(n-1)^2}{(n-2)(n-3)}} \quad (14)$$

where,

$$Skew = \frac{\sum_{i=1}^{n}(x_i - \mu)^3}{\sigma^3} \quad (15)$$

$$Kurtosis = \frac{\sum_{i=1}^{n}(x_i - \mu)^4}{\sigma^4} \quad (16)$$

In equations 14, 15 and 16, μ is the sample mean and σ is the sample standard deviation. The bimodality coefficient is calculated after each calculation of real power value. The computational requirements of other equations in pattern matching [1.19] depend on the frequency of usage for loads which may depend on the time of day.

1.5 Computational and Data Communication Requirements of the Proposed Smart Power Meter

This section explains the computational and communicational requirements of features in the proposed smart power meter.

1.5.1 Per-Cycle Measurements

The proposed meter measures per-cycle values of RMS current, RMS voltage, real power and reactive power. The fundamental component is 60 Hz. The data rate for per-cycle measurements are in Table 1.1

Table 1.1: Computations for RMS and Power Calculation

Measured Parameter	Number of Multiplication	Number of Additions
V_{RMS}	N+1	N
I_{RMS}	N+1	N
Real Power	N+1	N
Reactive Power	N+1	N

Table 1.1 shows the computations handled by microcontroller for per-cycle measurement as determined by the implemented algorithms in Section 1.4. Each cycle measurement takes 4N additions and 4 multiplications for each per cycle measurement.

In configuration I (Section 1.3.1.1), there are 12 circuits. Therefore the power meter is monitoring 12 current signals and three voltage signals. Thus it takes 39N additions and 39N+39 multiplications to complete each per-cycle measurement of a sub-panel. The payload size of per-cycle measurements is 78 bytes. Thus, the data rate of per-cycle measurements generated in the microcontroller is 4875 bytes per second.

In configuration II (Section 1.3.1.2), the power meter is monitoring 42 circuits connected to a single three phase supply. In this case, it takes 129N additions and 129N+129

multiplications to complete per-cycle measurements discussed in Section 1.4.1 , 1.4.2 and 1.4.3. The total payload of per-cycle measurements generated in this configuration is 258 bytes. The data rate of per-cycle measurements generated in the central microcontroller is 16125 bytes per second.

1.5.2 Condition Monitoring

This process makes use of DFT (explained in Section 1.4.4). Block size, N, controls the frequency which is resolved from DFT. The calculation of the complex exponential in equation 10 is computationally intensive and hence is therefore pre-computed and stored in the memory of the microcontroller. The number of multiplications required for a single DFT operation is N. The DFT operation also requires N additions. The payload of DFT results contains a time stamp (32 bits), ID specific to a frequency (16 bits) and channel number (16 bits). The total size of a single DFT result is 8 bytes.

Computation of a DFT is controlled by external processes; an example of this external control is user settings which will require a DFT to be calculated each time a device is turned on or off. Therefore, the number of DFT computations can be modified to fit within the available bandwidth.

1.5.3 Pattern Matching

The calculations performed by this algorithm are discussed in Section1.4.5. Equations 14, 15 and 16 show that these calculations have terms to the power of 2, 3 and 4 which are computationally intensive. The bimodality coefficient is calculated after each calculation of real power (60 Hz).

The payload of pattern matching algorithm contains a time stamp (32 bits), identification number specific to the load (16 bits) and channel number (16 bits). The total size of pattern

matching algorithm result is 8 bytes. The data rates of the results depend on the frequency and number of devices monitored in the circuit.

1.6 Communication Features of Smart Meter

The Section 1.5 discussed the hardware computational overhead and rate of data produced by features promised by the smart power meter. In architecture I (Section 1.3.1.1), the proposed power meter requires a minimum of 39 Kbytes per second of bandwidth to communicate the results to a data server. This data rate is for per-cycle measurements discussed in Section 1.5. The data rates of condition monitoring and pattern matching features will depend on the real time activity in the electric line.

Table 1.2: Data rates of popular communication schemes

Communication Scheme	Frequency of Operation (MHz)	Data Rate (Mbits/s)
Wi-Fi (IEEE 802.11 b/g/n)	2400	54
Wi-Fi (IEEE 802.11 ac)	5000	866.7
Power Line Communication (PLC)	0.5	0.195
Power Line Communication (PLC)	2	1.00
Bluetooth v3.0	2400	24.0
Zigbee 802.15.4	2400	0.244

Table 1.2 shows the data rates for some of the popular data communication technologies. The comparison of required data rate of power meters with Table 1.2 shows that required data rates can be provided by Wi-Fi, Bluetooth and PLC with 2 MHz bandwidth. Bluetooth is not viable because the data server for power meters is not in line of sight with the sensor nodes in

practical implementations and the distance between the sensor and data server usually exceeds the 1 m Bluetooth maximum theoretical range [1.13].

Wi-Fi technology is also not appropriate for the implementation of smart meters. Deployment costs for Wi-Fi nodes are low because Wi-Fi networks are popular and have penetrated most housing, industrial and organizational facilities. However, deployment of power meter nodes under the umbrella of existing Wi-Fi networks brings up administrative issues for common wireless channels with existing applications. In addition, Wi-Fi cards for embedded system applications are relatively costly [1.14].

Zigbee offers a great solution as the price of a 2.4 GHz radio from Digi International is $9-$15 [1.15] which is far less expensive than most of the current Wi-Fi transceivers in the market. The maximum range of a 2.4 GHz Zigbee transceiver is 90m, which is adequate for many building applications. In addition to Zigbee, IEEE 802.15.4 defines several schemes by which the Zigbee range can be increased. Schemes include recommendation to use intermediate devices called Zigbee routers which act as data concentrators collecting data from sensors. It is necessary to consider increased network latency and data rate restrictions of this type of multi-hop Zigbee network.

The restrictions in the bandwidth of Zigbee can be addressed by using a suitable data compression technique [1.16] to reduce the required data rate (described in Section 1.4.3). The data compression techniques are broadly classified into – Lossless and Lossy. Lossy data compression schemes (Example – JPEG and MPEG) are used in audio and video data compression. By comparison, the modern and efficient lossless data compression techniques use probabilistic models which are computationally intensive [1.18].

A simple method which demands less calculation is selected for the smart power meter. The power meter produces output of real power, reactive power, V_{RMS} and $I_{RMS.}$ These values are produced every $1/60^{th}$ of second. These results are q15 fixed point numbers. The compression technique, Run Length Encoding (RLE) reduces the redundancy in the data points. For example, reactive power values of 100 watts, 100 watts, 100 watts can be compressed to one value. This is a basic example of run-length encoding. RLE encodes to match a pre-defined value or the values which are repeated. Noise in monitored electric lines causes fluctuation in the power measured by the power meter. Thus a dead banding is introduced to improve compression technique. In this technique, the values are not transmitted if the corresponding value after comparison with the value last transmitted is in the dead band. This compression technique comes with a cost that signal changes in the dead band are lost. These changes are assumed to be due to the noise component in the monitored signal.

1.7 Hardware Platform of Smart Power Meter

The proposed smart meter has computationally intensive features which demand a microcontroller with good performance. However, performance of the smart power meter must be balanced against cost constrains. The important and non-compromising features considered in the microcontroller selection are memory and operating frequency. The memory of the microcontroller is vital for pattern matching and condition monitoring features as they require temporary storage of some of the associated parameters which are discussed in Section 1.4.2. The operating frequency is vital as it decides the upper cap of the total number of computations which can be performed by the microcontroller. The price of a microcontroller heavily depends on the above factors. Other features of the microcontroller are GPIO (General Purpose Input-

Output) pins, hardware timers, and advanced I/O capabilities, such as USB, that are fairly common across available microcontroller options.

1.8 Smart Power Meter – Architecture

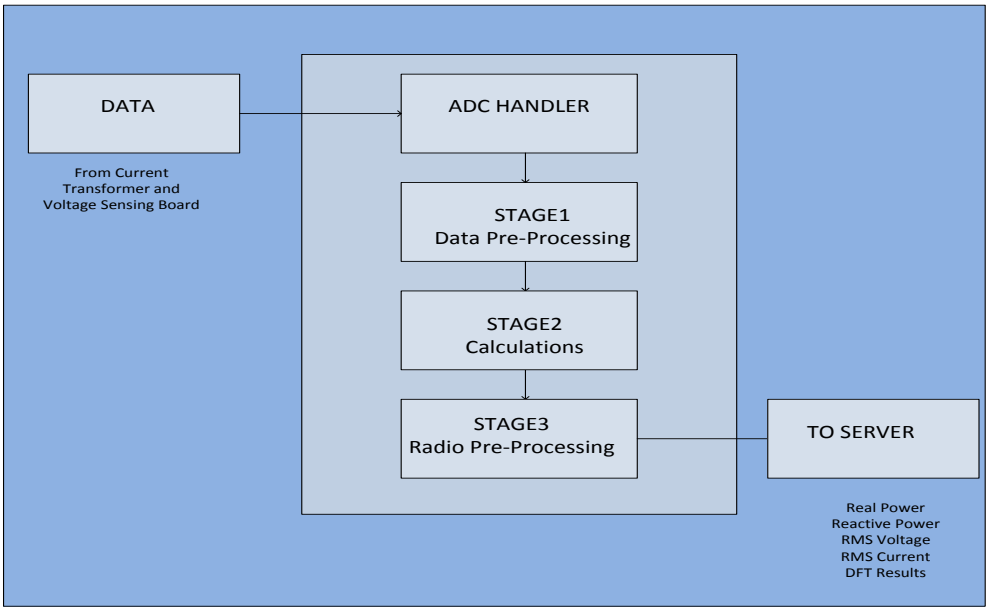

Figure 1.6: Smart Power Meter – Software Architecture

Inside smart power meter node, analog data from voltage and current sensors are sampled by Analog to Digital Converters (ADC). The sampled data is stored in a buffer. Two buffers are used in the microcontroller and they are used alternatively. The digital data is processed as follows:

1.8.1 Stage 1

The data from ADC buffer is transferred to stage 1. The data from ADC is interleaved data which contains data from 15 channels. Stage 1 performs the function of un-interleaving data and converting data into q15 fixed-point format. The un-interleaved data is arranged into 2D matrix. The columns represent the channel and rows represent the type of measured parameter.

1.8.2 Stage 2

Stage 2 calculates RMS voltage, RMS current, real power and reactive power from un-interleaved data. The data from stage 2 is a 2D matrix which contains the calculated results. The columns represent the type of data and rows represent channels. Stage 2 also performs pattern matching and frequency scan and stores the results accordingly.

1.8.3 Stage 3

Stage 3 stores all results from stage 2 in a buffer. The data is subjected to Run Length Encoding (RLE) described in Section 1.6. The resulting data is transferred into the radio buffer when the buffer allocated to stage 3 is full.

1.9 Component Decisions for Smart Power Meter

The computational and communicational capabilities of the proposed smart power meter are discussed in Sections 1.5 and 1.6. Due to cost constrains in this project, communication bandwidth is constrained to 250 kBits per second. The product development team selected ATMEL SAM3SD8 processor which promises a clock frequency of 64 MHz and a flash memory of 512 Kbytes. The features of the microcontroller also include 64 Kbytes of SRAM memory with 79 GPIO (General Purpose Input-Output) pins. The team also has selected Zigbee radio to satisfy the bandwidth requirements of the smart power meters. The performance analysis of the same is discussed in Chapter 2 and Chapter 5.

1.10 Thesis Overview

This thesis covers a series of experiments and modeling steps to develop the smart power meter by optimizing the computational and communication bandwidth required for features in Section 1.4.

Chapter 2 explains an experiment conducted for performance analysis of communication in Zigbee network. Chapter 3 discusses an approach to improving the frequency response of the power meter by oversampling and then downsampling measurements in the analog front end and subsequent signal processing. Chapter 4 discusses a system level model developed in Simulink to measure computational performance characteristics of the selected ATMEL SAM3S based processor after implementation of smart power meter features discussed in Section 1.4.2. Chapter 5 discusses the communication performance of a network of the proposed wireless smart power meters.

1.11 References

[1.1] "Monitoring Energy Use: The Power of Information" [Online]. Available: http://www.schneider-electric.com/solutions/ww/en/med/4665025/application/pdf/1256_monitoring_energy_use.pdf. [Accessed: 09-Dec-2013].

[1.2] "Guide to Energy Monitoring." [Online]. Available: http://efergy.com/us/guide-energy-monitoring/. [Accessed: 10-Dec-2013].

[1.3] "An Overview of Energy Use and Energy Efficiency Oppurtunities." [Online]. Available: http://www.energystar.gov/ia/business/challenge/learn_more/Schools.pdf .[Accessed: 10-Dec-2013].

[1.4] "Big Savings for Texas City with Energy Monitoring Sensors," *Energy.gov*. [Online]. Available: http://energy.gov/articles/big-savings-texas-city-energy-monitoring-sensors. [Accessed: 10-Dec-2013].

[1.5] "Wattson: Monitor Your Home's Energy Usage," *TreeHugger*. [Online]. Available: http://www.treehugger.com/clean-technology/wattson-monitor-your-homes-energy-usage.html. [Accessed: 10-Dec-2013].

[1.6] J. Li, M. Sumner, J. Arellano-Padilla, and G. Asher, "Operating limits for drive condition monitoring using supply current signature analysis," in *Electric Machines Drives Conference (IEMDC), 2011 IEEE International*, 2011, pp. 412–417.

[1.7] "Quadlogic Power Meter - Low Cost, Accurate, Utility-Grade Electric Submeters, Tenant Billing." [Online]. Available: http://www.quadlogic.com/products.html. [Accessed: 11-Dec-2013].

[1.8] “Power Xpert Meter 4000/6000/8000.” [Online]. Available: http://www.eaton.com/Eaton/ProductsServices/Electrical/ProductsandServices/PowerQualityandMonitoring/PowerandEnergyMeters/PowerXpertMeter400060008000/index.htm. [Accessed: 11-Dec-2013].

[1.9] “Smart Meter, Dumb Idea?” [Online]. Available: http://online.wsj.com/news/articles/SB124050416142448555. [Accessed: 11-Dec-2013].

[1.10] “PQMII Power Quality Meter.” [Online]. Available: http://www.gedigitalenergy.com/multilin/catalog/pqmii.htm. [Accessed: 11-Dec-2013].

ASD

[1.11] “ATSAM3SD8C.” [Online]. Available: http://www.atmel.com/devices/SAM3SD8C.aspx. [Accessed: 11-Dec-2013].

[1.12] “IEEE Standard Definitions for the Measurement of Electric Power Quantities Under Sinusoidal, Nonsinusoidal, Balanced, or Unbalanced Conditions,” *IEEE Std 1459-2010 (Revision of IEEE Std 1459-2000)*, pp. 1–50, 2010.

[1.13] “Bluetooth - Wikipedia, the free encyclopedia.” [Online]. Available: http://en.wikipedia.org/wiki/Bluetooth. [Accessed: 12-Dec-2013].

[1.14] “Digi Connect Wi-Wave - 802.11b/g radio card for embedded system - Digi International.” [Online]. Available: https://www.digi.com/products/wireless-wired-embedded-solutions/satellite-wifi-cryptographic/wifi-connectivity/digi-connect-wi-wave. [Accessed: 12-Dec-2013].

[1.15] “DigiKey Electronics - Electronic Components Distributor.” [Online]. Available: http://www.digikey.com/. [Accessed: 12-Dec-2013].

[1.16] “Data compression,” *Wikipedia, the free encyclopedia*. 06-Dec-2013.

[1.17] “Run-Length Encoding.” [Online]. Available: http://www.dspguide.com/ch27/2.htm. [Accessed: 09-Dec-2013].

[1.18] “Burrows-Wheeler Transform” [Online]. Available: http://www.cs.cmu.edu/~ckingsf/bioinfo-lectures/bwt.pdf. [Accessed: 09-Dec-2013].

[1.19] Matt O’Connell, M.S. thesis (Under preparation), ME Dept., Colorado State University, Fort Collins, CO, 2014.

[1.20] “Basic kWh Meter 100A 120/240-volt, 3-wire, 60Hz EKM-25IDS.” [Online]. Available: http://www.ekmmetering.com/ekm-metering-products/electric-meters-kwh-meters/basic-kwh-meter-100a-120-240-volt-3-wire-60hz-ekm-25ids.html?gclid=CIi8gZ-QuLsCFcsRMwod_nAAGQ. [Accessed: 17-Dec-2013].

[1.21] “SiteStage Features” [Online]. Available: http://powerhousedynamics.com/about-sitesage/features1. [Accessed: 04-May-2014]

[1.22] “SiteStage Energy Monitor” [Online]. Available: http://www.amazon.com/SiteSage-24h-formerly-eMonitor4-24-Powerhouse-Dynamics/dp/B006Z5OKW8. [Accessed: 04-May-2014]

Chapter 2

Zigbee Radio Range Testing

2 Overview

This chapter describes an experiment performed to measure the performance of Zigbee network in a test environment where the proposed smart power meter will be deployed. Section 2.1 introduces the Zigbee protocol with a comparison of Zigbee protocol with the popular communication protocols in the current market. This section also introduces the co-existence of Wi-Fi with Zigbee schemes.

Section 2.2 explains test setup to measure the performance of Zigbee based wireless communication in a clean environment and an environment with several Wi-Fi radios. The Data Loss is selected as a metric to measure the performance. Section 2.3 presents the results of the test.

2.1 Introduction

Since its invention, Zigbee has aimed in devices such as switches, energy meters and sensor systems functioning in commercial and industrial buildings which were overlooked by wireless radio industries. Zigbee alliance members created a wireless standard offering easy installation and a mechanism for coexistence of Zigbee with other wireless systems working in the same operation frequency of Zigbee devices. Zigbee is built using the Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 [2.11]. Though low powered, Zigbee devices can achieve long distances by passing data through intermediate Zigbee radios, forming a mesh network. The Zigbee networks are secured by 128 bit encryption keys. Zigbee communication protocol claims to achieve data transmission rates of up to 250Kilobits per second when operating in the frequency of 2.4 GHz.

Zigbee is primarily used in applications which require a low data rate, prolonged battery life and a secure network. The Zigbee protocol supports star, mesh and tree type networks. Every Zigbee network has a coordinator node which is responsible for control of network parameters and basic maintenance of network.

2.1.1 Zigbee Protocol Stack

The Zigbee protocol stack is based on IEEE 802.15.4 standard [2.11]. This standard gives special emphasis for low cost communication with no underlying infrastructure so that users can exploit this to achieve lowest power consumption demanded by their respective application. The main features of IEEE 802.15.4 are real-time reservation of time slots for transmission nodes and collision avoidance through CSMA/CA. Features also include functions for smart power management in Zigbee radios.

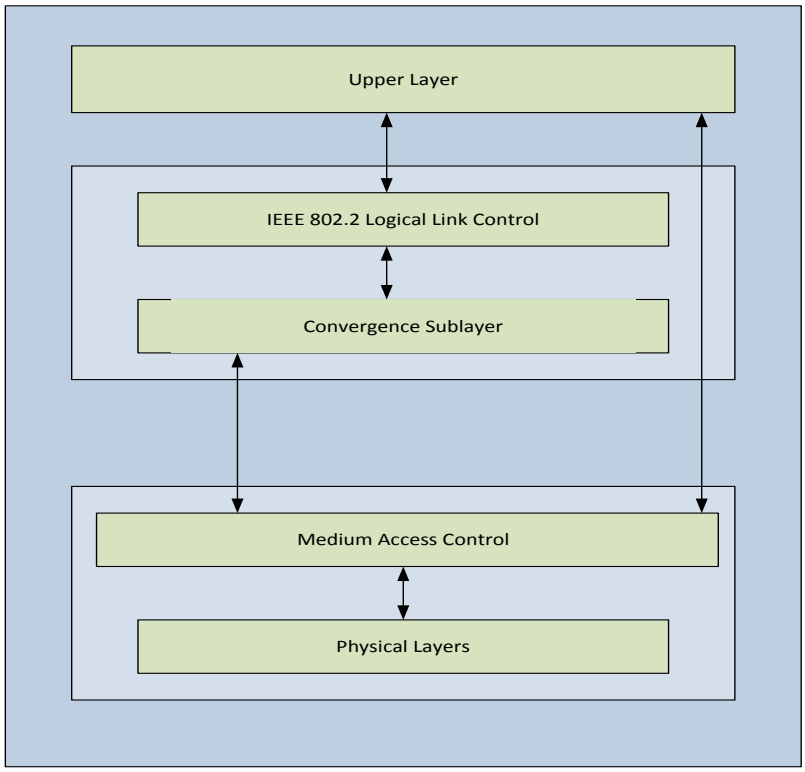

Figure 2.1: IEEE 802.15.4 Protocol Stack [2.1]

IEEE has defined only lower layers of OSI model in IEEE 802.15.4 standard. Protocols for the transfer of data to and from upper layer are defined in IEEE 802.2 standard. The lower physical layer performs channel selection and energy management functions. The MAC layer enables the transmission of data frames from application layer to physical layer. The MAC layer also performs functions such as network beaconing, time slot reservation and frame validation.

2.1.2 Zigbee Frequency of Operation

Zigbee operates in license-free Industrial, Medical and Scientific (ISM) band. Zigbee operates in three ISM bands – 2.4 GHz (Worldwide), 915 MHz (North America and Australia) and 868 MHz (Europe).

2.1.3 Zigbee Range Comparison

Table 2.1: Zigbee Comparison with Wireless Standards [2.5]

	Zigbee	802.11	Bluetooth	Ultra- Wide Band	Wireless USB
Data Rate	250 Kb/s	11/54 MB/s	1 Mb/s	100-500 Mb/s	63 Kb/s
Range	10-100 meters	50-100 meters	10 meters	0-10 meters	10 meters
Network Topologies Supported	Peer to peer, ad-hoc, star and mesh	Point to hub, ad-hoc	Point to point	Point to point	Point to point
Operating Frequency	2.4 GHz (Worldwide)	2.4 GHz and 5 GHz	2.4 GHz	3.1 to 10.6 GHz	2.4 GHz
Power Consumption	Low	High	High	Low	Low
Security	128 AES	None	64 and 128 bit encryption	None	None

Table 2.1 and Figure 2.2 presents a comparison of popular wireless Personal Area Networks (PAN) networks in terms of data rate, range, network topologies, operating frequency, power consumption and security features. Zigbee offers data rates of 250 Kbits per second while

Wi-Fi promises more than 10 times the bandwidth and range of Zigbee. Bluetooth promises higher data rate but lesser range compared to Zigbee.

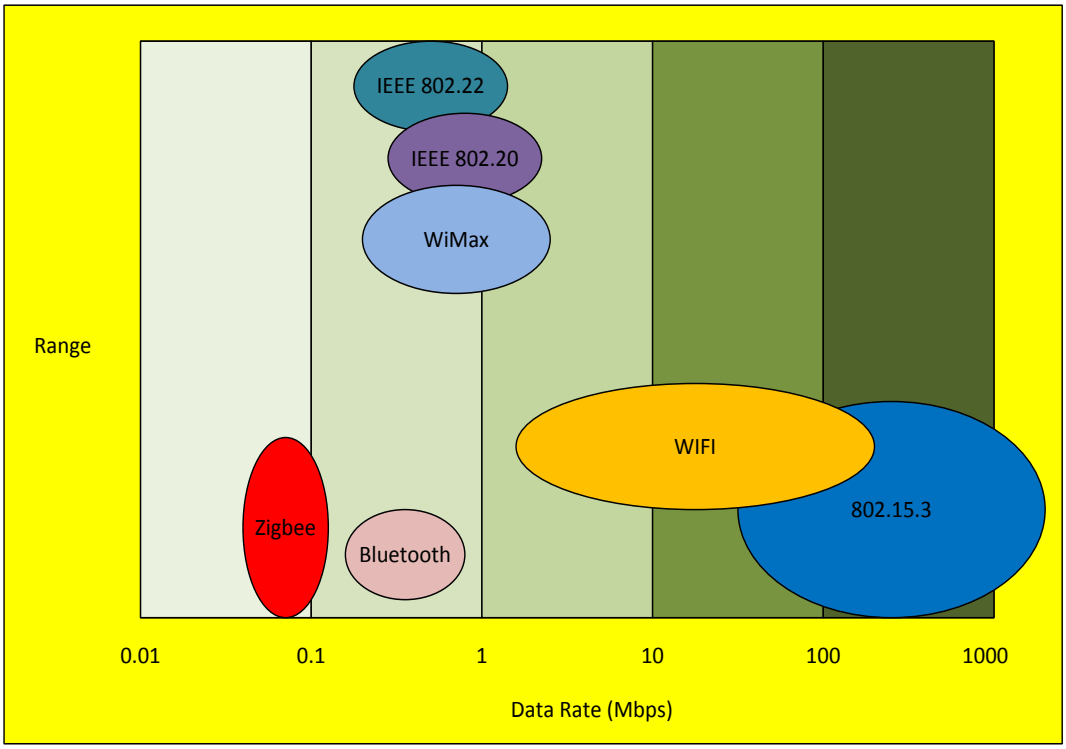

Figure 2.2: Zigbee Protocol Comparison (Range Vs. Data Rate) [2.5]

Section 1.6 described communication schemes and the practicality of using these schemes with the smart power meter. The choice of wireless communication technology for the smart power meter was narrowed down to Zigbee or Wi-Fi in Section 1.6. Wi-Fi has a data rate almost 1000 times more than Zigbee. Wi-Fi also has better range compared to Zigbee. The Zigbee scheme is selected because Zigbee transceivers are priced less than Wi-Fi transceivers and they are within the project budget. The IEEE 802.15.4 standard recommends schemes to increase the range by using intermediate device called Zigbee routers. The router also helps to control the data flow in the network by buffering the data from the sensor nodes.

2.1.4 Coexistence with Wi-Fi

The Zigbee operates in license-free industrial scientific and medical (ISM) bands. The following wireless systems are the main users of 2.4 GHz ISM band:

- 802.11 b/g/n networks
- Bluetooth networks
- Cordless phones
- Microwave Ovens
- WiMax Systems

The Zigbee radio, which operates in the frequency of 2.4 GHz, shares its frequency of operation with the above wireless systems and hence they are susceptible to interference from other wireless systems. The table below shows the wireless channels of operation of Zigbee radios as specified in IEEE 802.15.4 when they are operated in the frequency of 2.4 GHz.

Table 2.2: Zigbee Channels [2.6]

Channel No	Center Frequency (GHz)
11	2.405
12	2.410
13	2.415
14	2.420
15	2.425
16	2.430
17	2.435
18	2.440
19	2.445
20	2.450
21	2.455
22	2.460
23	2.465
24	2.470
25	2.475
26	2.480

The main source of interference for Zigbee signals are Wi-Fi signals. Wi-Fi signals are the most dominant contributor of interference compared to other wireless technology working in the same operational frequency. This is due to the high penetration of Wi-Fi networks in

commercial and residential applications. This penetration causes significant Wi-Fi network traffic in industrial, commercial and housing establishments which can interfere with Zigbee transmissions operating in the same frequency. The Wi-Fi devices used in US operate in 11 wireless channels in 2.4 GHz band. Each channel is evenly spaced at 5 MHz separation. Channel 1 operates at 2.412 GHz and channel 11 operates at 2.462 GHz. Wi-Fi protocol also demands a channel separation of 16.25 MHz to 22 MHz between its channels. Wi-Fi communication channels overlaps with the channels allocated for 802.15.4 protocol. This can reduce the performance of Zigbee radio as Zigbee operates in much lower power compared to Wi-Fi systems (WIFI signals are transmitted at 1 watt [2.7] compared to Zigbee pro which transmits at 63 mW [2.3]).

The performance of wireless power meters depends on existing Wi-Fi networks in the environment. Zigbee signals can also interfere with cordless phone and Bluetooth network signals which are operating in 2.4 GHz bandwidth. The measurement of performance is necessary before the deployment of sensor nodes to make necessary changes to architecture and the transmission time of the nodes.

2.2 Test Setup – Zigbee Performance Measurement

Packet loss was selected as the metric to measure the performance of the Zigbee network. Two Zigbee nodes were used in the test – One was configured as a client (Zigbee end device) and other as server. Packets were transmitted from the client to server. The test was conducted with Regular Zigbee Radio (900 MHz bandwidth, 10mW transmission power) and Pro Zigbee Radio (2.4 GHz bandwidth, 63mW transmission power). The effect of processor power (Clock frequency of the processor) to process the received data is also investigated by using Zigbee radios with a personal computer (2GHz clock frequency Intel Celeron processor) and with an

Arduino board (16MHz microcontroller clock frequency). The test was also conducted in two environments – Wi-Fi environment and Non Wi-Fi environment to characterize the effects of Wi-Fi devices in the operating environment.

The data packets were generated in client. Each data packet consists of a time stamp, network address, source address, packet length, protocol version, counter (incremented with each packet as generated by the client), device time stamp and receive time stamp. The receiver Zigbee node monitored the counter values to detect data loss. Time stamps were used to estimate delay in data.

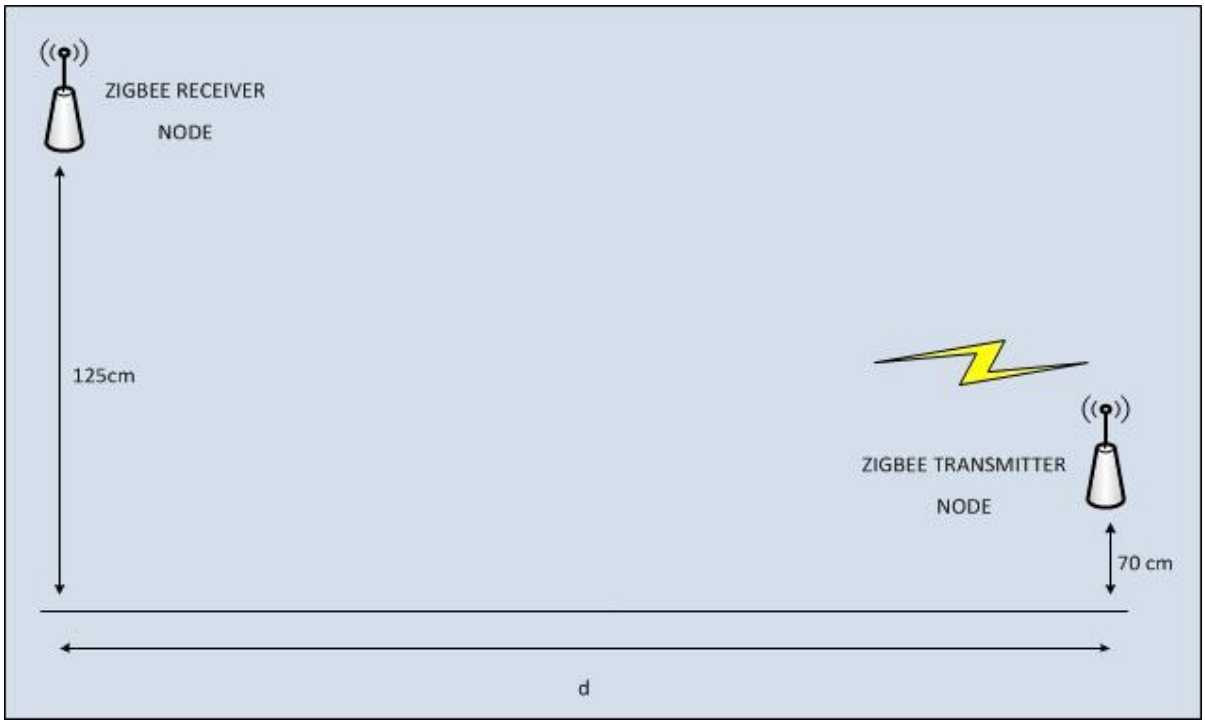

Figure 2.3: Zigbee Performance Test Setup

The test was performed in Engines and Energy Conversion Laboratory (EECL) at Colorado State University (Wi-Fi environment) and at City Park, Fort Collins (Non Wi-Fi environment). EECL had 3 WIFI networks. In the Wi-Fi environment, traffic was highest in Colorado State University Wi-Fi network (Center Frequency at 2.422 GHz). The Zigbee nodes

were configured to operate in channel 14 (Center frequency at 2.420 GHz). This channel was selected to measure the performance of the network when the Zigbee radios operate in the same frequency as Wi-Fi radios.

2.2.1 Experiment 1

The experiment was performed in Engines and Energy Conversion Lab at CSU. Figure 2.4 and Figure 2.5 shows the screenshot of 2.4 GHz spectrum recorded by channel analyzer. The top half of the diagram shows discrete plots of signal energy and bottom half shows a flow diagram in which amplitude (X-Axis) is plotted with time (Y-Axis). The color of the dots in plot depicts the intensity of signal in the frequency band. In Figure 2.4, A and B shows the frequency range where high power Wi-Fi transmission is recorded. These Zigbee radios are operated in this frequency band as described in Section 2.2.

Spectrum recordings of the 2.4 GHz spectrum showed significant wireless activity in Wi-Fi channel 2 (2.410 GHz), channel 3 (2.415 GHz) and channel 4(2.420 GHz). The Zigbee radio was operated in channel 13 (2.415 GHz). The test was performed with two configurations:

- UART connection between Zigbee receiver and PC (INTEL processor with 2.4 GHz) which records the received data.
- The received data was processed by Arduino stalker (ATMega328P microcontroller with 20 MHz).

The test was performed in two configurations to quantize the effect of clock frequency of the chip responsible for transmitting/receiving data by using Zigbee radios in the test environment. The test was also performed with regular Zigbee (900 MHz) and pro-Zigbee (2.4 GHz) to quantify the effect of frequency of operation of Zigbee radios.

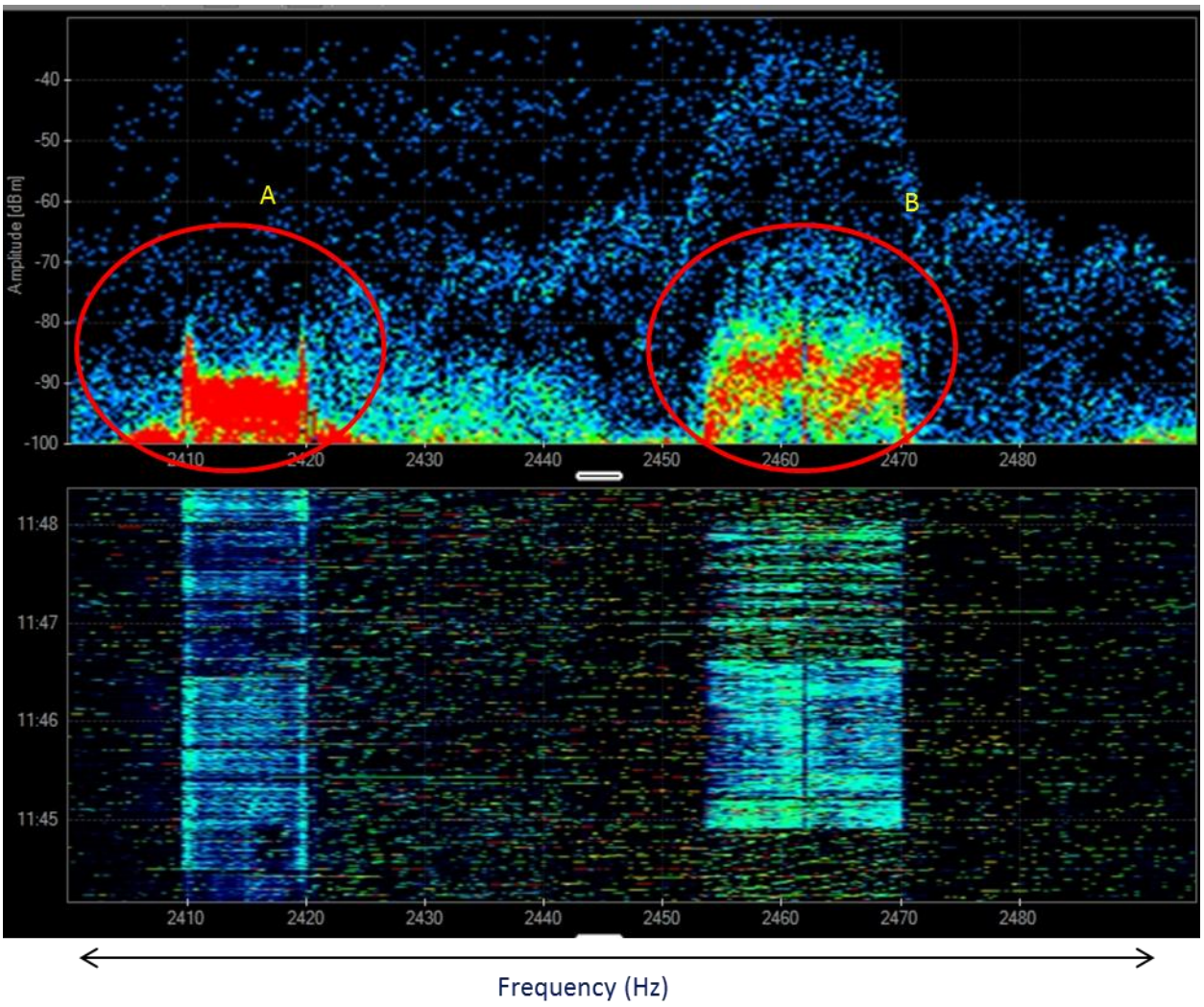

Figure 2.4: 2.4 GHz Spectrum (WIFI Environment)

2.2.2 Experiment 2

In experiment 2, all radio parameters were the same as experiment 1. The experiment 2 was conducted in an environment without Wi-Fi signals. Figure 2.5 shows 2.4 GHz spectrum in the scenario. The scenario showed no wireless signal activity in 2.4 GHz spectrum.

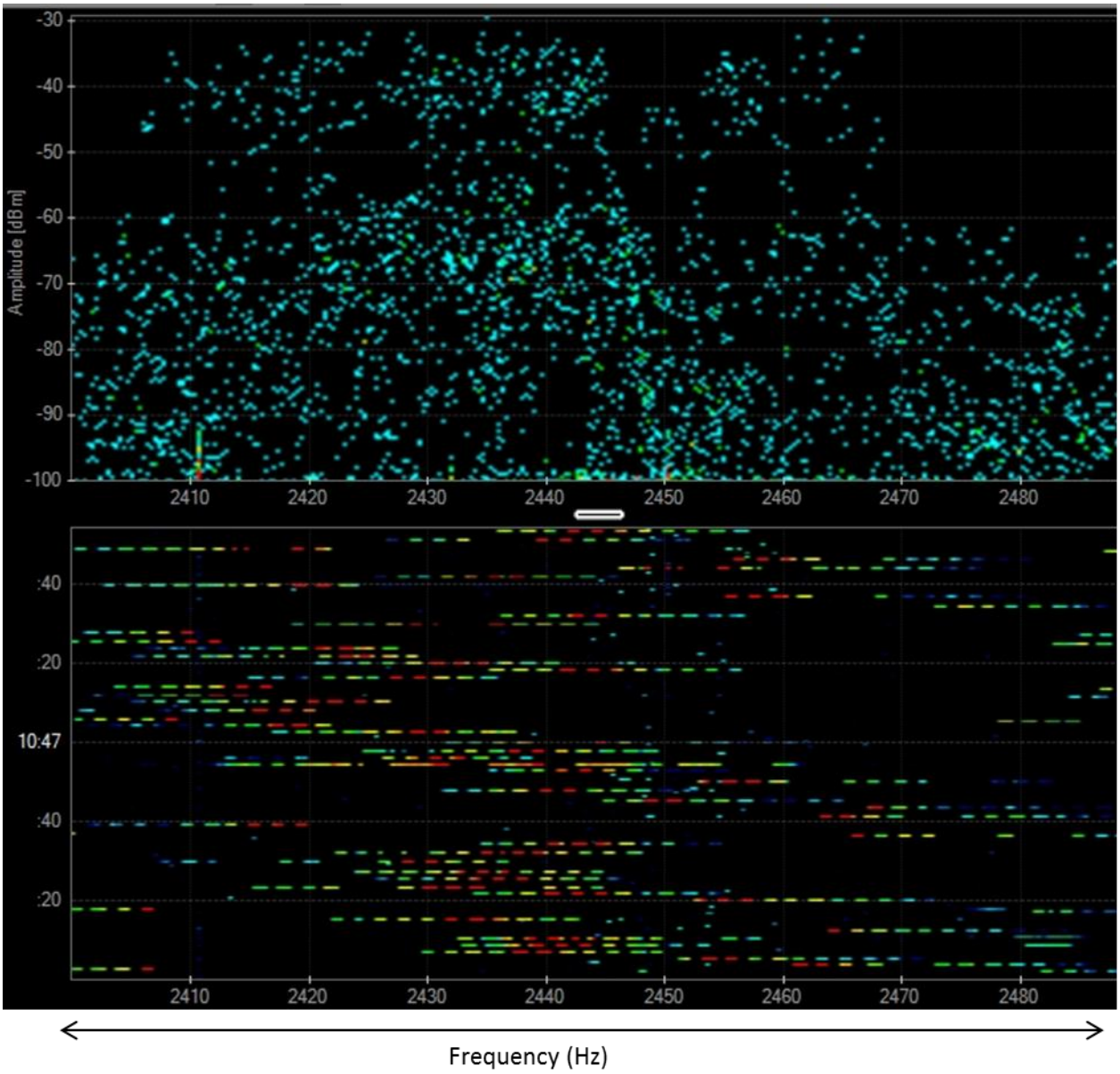

Figure 2.5: 2.4 GHz Spectrum (Experiment 2)

2.3 Test Results

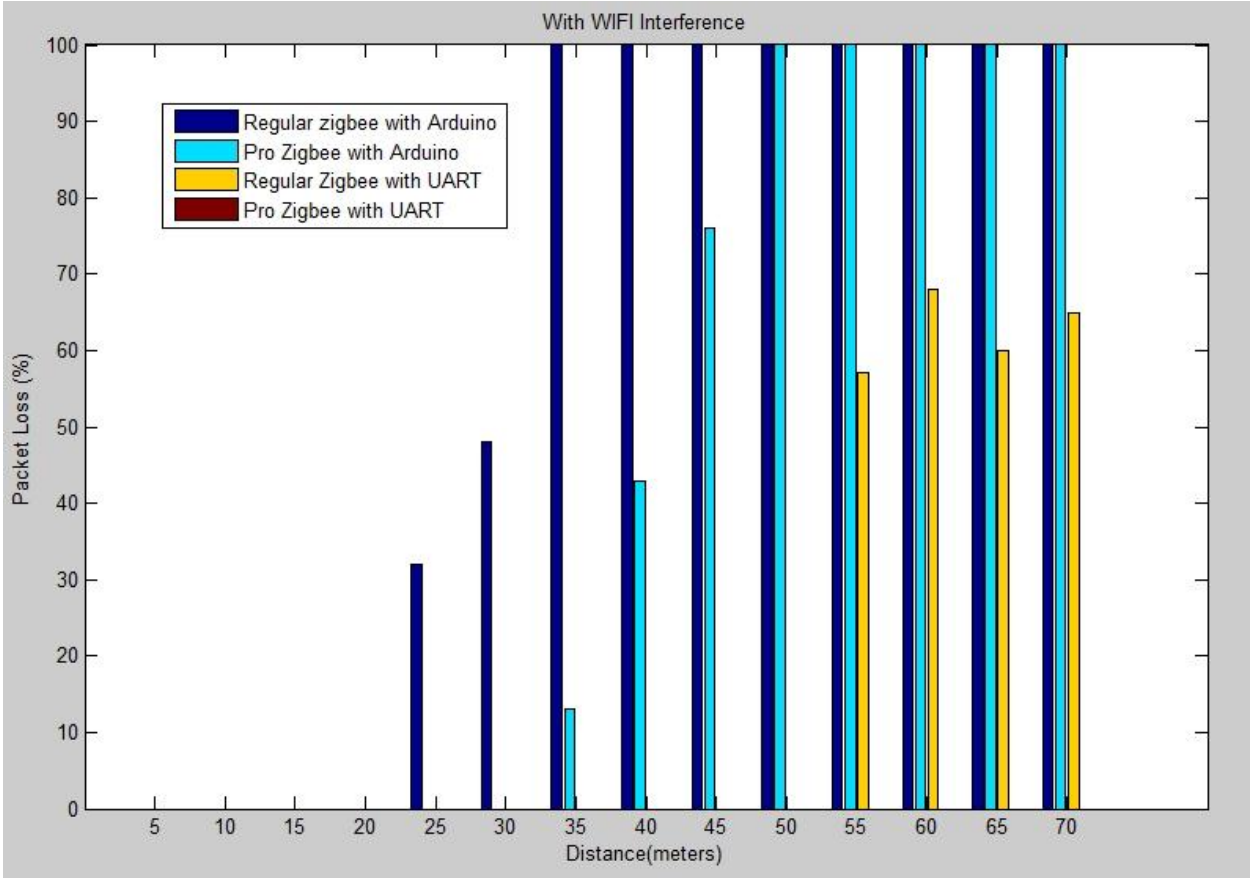

Figure 2.6: Zigbee Radio Performance with WIFI Interference (Packet Loss Vs. Distance (meters))

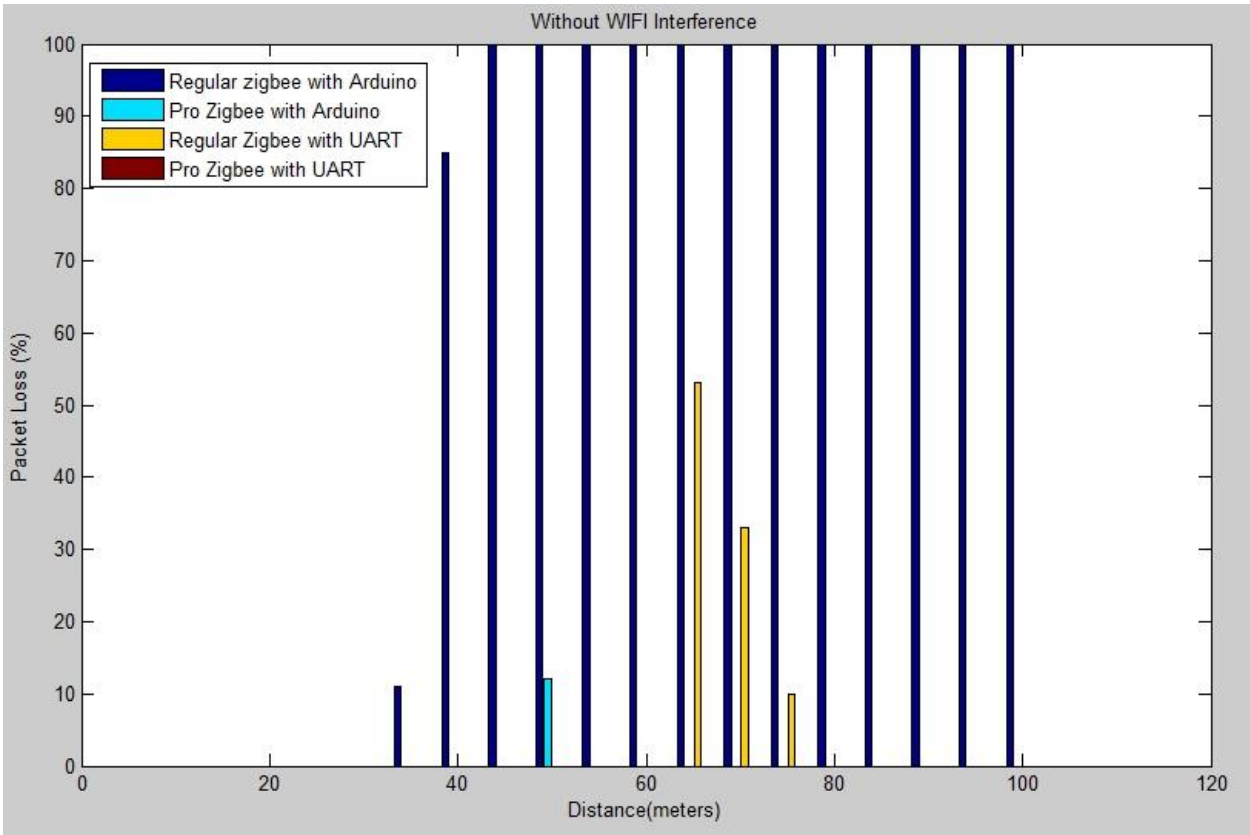

Figure 2.7: Zigbee Radio Performance in Environment Without WIFI (Packet Loss Vs. Distance (meters))

Figure 2.6 and Figure 2.7 show the performance of Zigbee radios working in an environment with Wi-Fi signals and without Wi-Fi signals. The comparison between Figure 2.6 and Figure 2.7 shows that Zigbee radios used with Arduino based processors perform better in terms of range in an environment without Wi-Fi. The test results show that when the Zigbee radio is used in Arduino board with an ATMEL microcontroller (ATMega328P), their range is limited to 30m (900 MHz Zigbee radio). In an environment with Wi-Fi activity (Figure 2.6), the Zigbee shows a performance improvement when used with a faster processor (The range is improved to 70m). Thus a faster backend processor in Zigbee end node improves the communication efficiency of Zigbee networks. The comparison of test results from experiments in Section 2.2.1 and Section 2.2.2 shows that in an environment with Wi-Fi, the packet loss

reaches 100% at 35 meters whereas the same radios when used in environment without Wi-Fi the 100% packet loss is recorded at 45 meters. Therefore, the performances of Zigbee modules are limited by Wi-Fi signals in the environment. In Figure 2.6, the comparison between performance of Zigbee radio used with Arduino boards (Blue bar) and Zigbee radio used with Intel processor connected via UART (Yellow bar) shows that the Zigbee radio used with an Intel processor shows a packet loss of 60% at 55 meters and the Zigbee radio with Arduino based processor showed a 100% loss of packets at 55 meters. Therefore, performance of Zigbee radio is also limited by type of processor used with Zigbee modules.

2.4 Conclusion

The performance test results reveal that Zigbee module performance is significantly limited when they operate in channels which overlap with Wi-Fi channels operating in the same frequency. The recorded range is much less than the range predicted by Digi International, the developer of the Zigbee modules utilized here. Digi International states that range for 900 MHz and 2.4 GHz Zigbee radio are 610m [2.8] and 90m [2.9] respectively. It is recommended that Zigbee channels are selected by considering existing WIFI networks in the deployment environment.

The packet loss in Zigbee based communication can also be due to ground reflection [2.10]. Ground reflected signals can play a prominent role in performance of low power devices. The wireless signals also reflect from PCB (Printed Circuit Board) where Zigbee radios are mounted. The Figure 2.8 shows Arduino board with Zigbee radio mounting pins. In this board, the mounted Zigbee radio is only few inches above PCB board which increases the reflected wireless signals from the board.

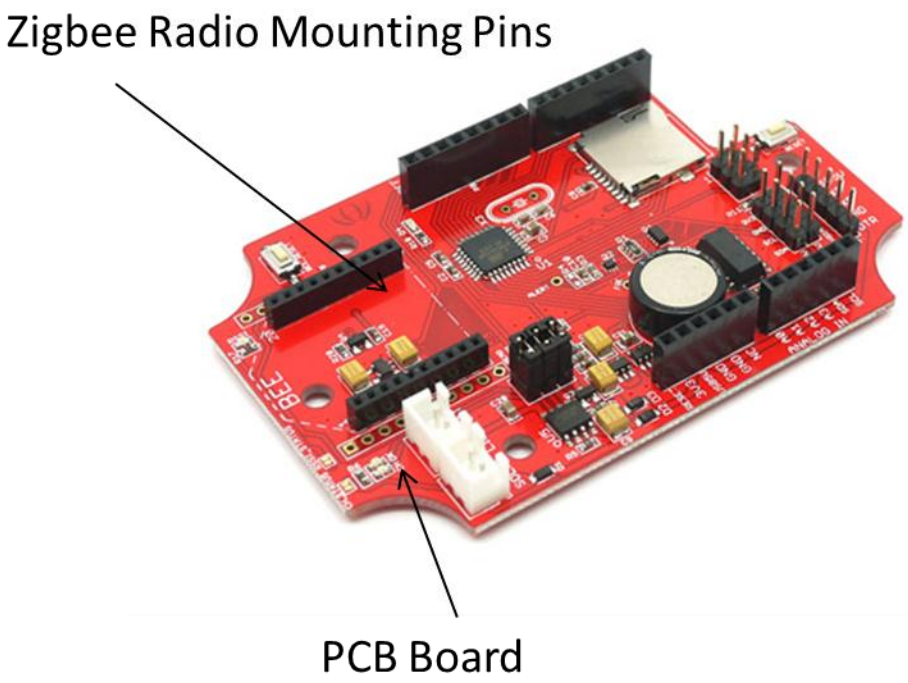

Figure 2.8: Example of PCB board with Zigbee radio mounting pins

Performance degradation due to ground based reflection can be avoided by design improvements in the system. One of suggested improvements is to place the radio in an independent PCB and place it higher or at a distance with respect to PCB board containing microcontroller and other circuitry. This will reduce the reflection of Zigbee radio signals from the circuit board.

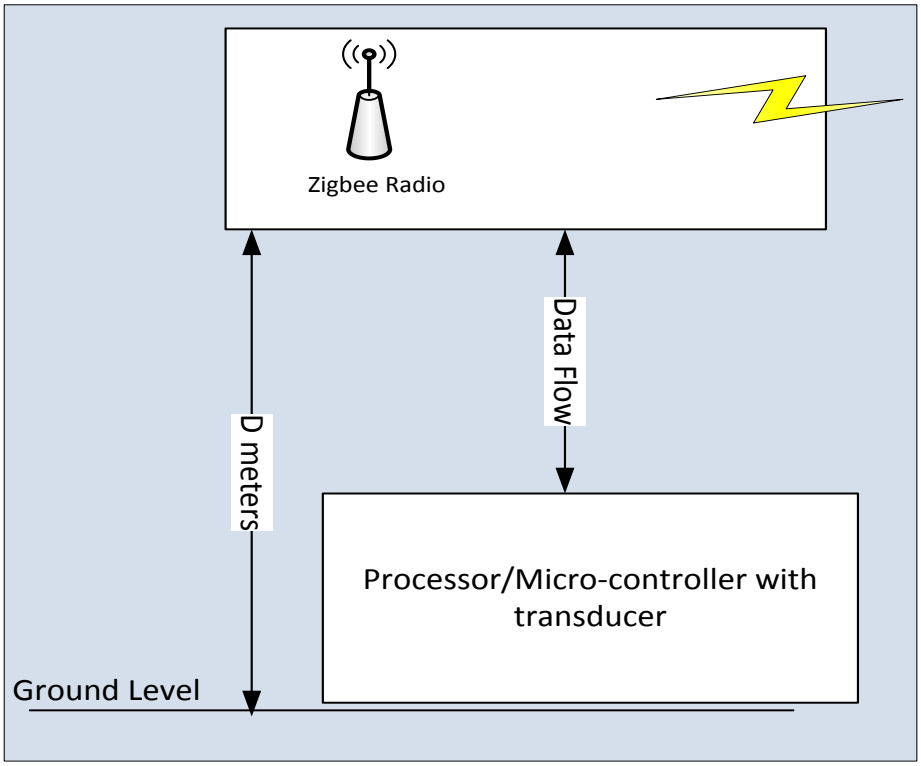

Figure 2.9: Proposed Performance Improvement Scheme for Zigbee Module

2.5 References

[2.1] “Zigbee”, [Online], Available: http://en.wikipedia.org/wiki/Zigbee, Last Accessed: 11/07/2013

[2.2] “Hirose U.FL”, [Online], Available: http://en.wikipedia.org/wiki/Hirose_U.FL, Last Accessed: 11/07/2013

[2.3] “Xbee 802.15.4”, [Online], Available: http://www.digi.com/products/wireless-wired-embedded-solutions/Zigbee-rf-modules/Zigbee-mesh-module/ , Last Accessed: 11/07/2013

[2.4] “Zigbee and Wireless Radio Frequency Coexistence”, [Online], Available: https://docs.Zigbee.org/Zigbee-docs/dcn/07-5219.PDF, Last Accessed: 11/07/2013

[2.5] “How does Zigbee compare with other wireless standards?”, [Online], Available: http://www.stg.com/wireless/Zigbee_comp.html , Last Accessed: 11/07/2013

[2.6] “Zigbee Channels”, [Online], Available: http://www.metageek.net/control4/Zigbee-channels/, Last Accessed: 11/07/2013

[2.7] “FCC rules for operation in ISM bands”, [Online], Available: http://www.afar.net/tutorials/fcc-rules/, Last Accessed: 11/07/2013

[2.8] “Xbee-PRO 900HP”, [Online], Available: http://www.digi.com/products/wireless-wired-embedded-solutions/Zigbee-rf-modules/point-multipoint-rfmodules/xbee-pro-900hp#specs , Last Accessed: 11/08/2013

[2.9] “Xbee 802.15.4”, [Online], Available: http://www.digi.com/products/wireless-wired-embedded-solutions/Zigbee-rf-modules/point-multipoint-rfmodules/xbee-series1-module#specs, Available: 11/08/2013

[2.10] O. Musikanon and W. Chongburee, “Zigbee Propagations and Performance Analysis in Last Mile Network,” *International Journal of Innovation, Management and Technology*, vol. 3, no. 4, 2012.

[2.11] “IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs),” *IEEE Std 802.15.4-2003*, pp. 0_1–670, 2003.

Chapter 3

Digital Filter Overhead Estimation

3 Overview

This chapter explains the estimation of computational overhead in the sampled data of the smart power meter. In the proposed smart power meter, Analog to Digital Converter (ADC) digitizes the analog data received from the voltage and current sensors. The ADC samples at a sampling rate which is almost 20 times Nyquist frequency. The high sampling frequency avoids aliasing and feeds high frequency data to algorithms. This sampling rate would demand a high computation power resource. Hence, a downsampling process is required before transferring the data into calculation stages described in Section 1.8. A typical downsampler system contains a digital low pass filter in the front-end to avoid effects of aliasing after the downsampling process. The implementation of digital filter requires a set of computation steps which depends on the type and number of poles of the filter. The computational load incurred by downsampling process depends upon the frequency of the data supplied to it. The computational load also depends on the type of arithmetic used for filter implementation. Thus, it is more efficient to do the downsampling of the data in stages. This chapter models the calculation of net computation overhead and overhead in each stage of downsampling process. This is followed by estimating the overhead for two proposed configurations.

3.1 Introduction

The implementation of filters in a microcontroller can be carried out in two ways – fixed point and floating point. The fundamental difference between these two is in their respective representation of numerical data.

The digital filters using floating point arithmetic to represent digital data offers much greater numerical bandwidth compared to the digital filters using fixed point arithmetic. However, floating point filters require complex computations compared to fixed point filters. Therefore, implementation of floating point filters typically require a microcontroller platform which has a built-in floating point unit to handle the complex floating point computations. This requirement can increase the component costs of the proposed smart power meter.

In order to understand the effect of the type of arithmetic on the performance of filters for the smart power meter, fixed and floating point capabilities are compared. In digital filters working in floating point arithmetic, each time a processor generates a new fixed point number, the number is rounded to nearest value so that number can be stored in a defined number of bits. Rounding and truncating noise are introduced in the signal which leads to significant deviation between the theoretically correct result and results produced by the software implementation of the smart power meter. Typically, this deviation is much larger in fixed point than in floating point systems.

3.1.1 Factors to Consider for Filter Selection

The following are the key factors to consider before selection of fixed point or floating point based implementation [3.2]:

Processor Cost: The ability to lower cost can significantly impact product profitability. Fixed point computations can be performed on microcontrollers without floating point units (FPU) or co-processors, and are typically less expensive than microcontrollers with floating point capability.

Ease of Development: The easier the development, the faster the product can be released to the market. Floating point algorithms may often be implemented with only minor

consideration of round-off errors in accumulators and similar software features. Fixed point algorithms must often be modified significantly from "textbook descriptions" to avoid truncation errors. The additional algorithmic development effort can make fixed-point development significantly more expensive.

Performance: The performance is a metric which shows the speed with which the tasks are performed by the processor. This metric largely depends on processor in the application. It is possible to use fixed point algorithm in a floating point processor and vice versa.

3.1.2 Implementation in Smart Power Meter

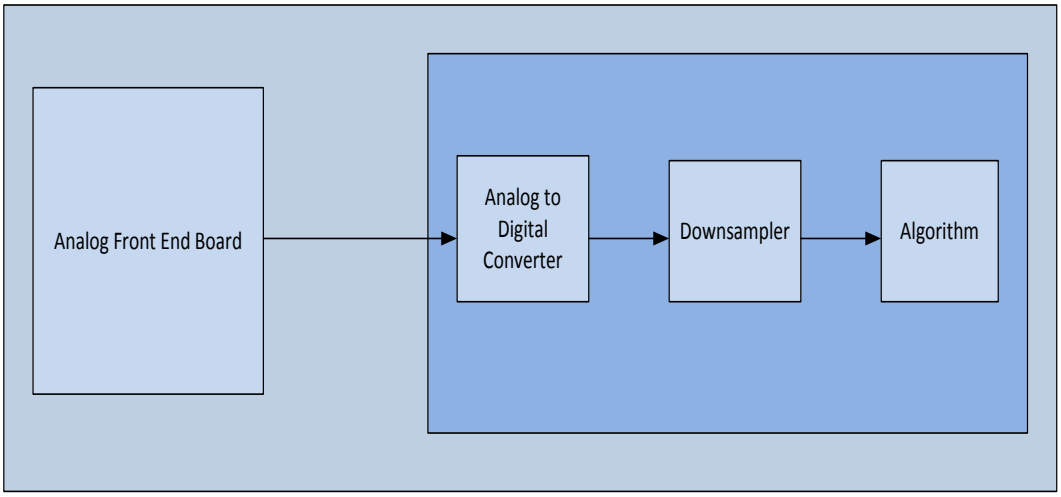

Figure 3.1: Front End System of Data Flow

The data from the analog sensors which capture voltage and current signals are sent to the Analog to Digital Converters (ADC) in the smart power meter. The ADC samples the data at frequencies much higher than the Nyquist limit to avoid aliasing. Directly processing this high frequency data would be computationally intensive and hence downsampling is used to lower the frequency of data before the data is processed by the algorithms in the smart power meter. This

downsampling process should designed to avoid loss of the key signal components required for the algorithms implemented in the proposed smart power meter.

3.1.2.1 Downsampling

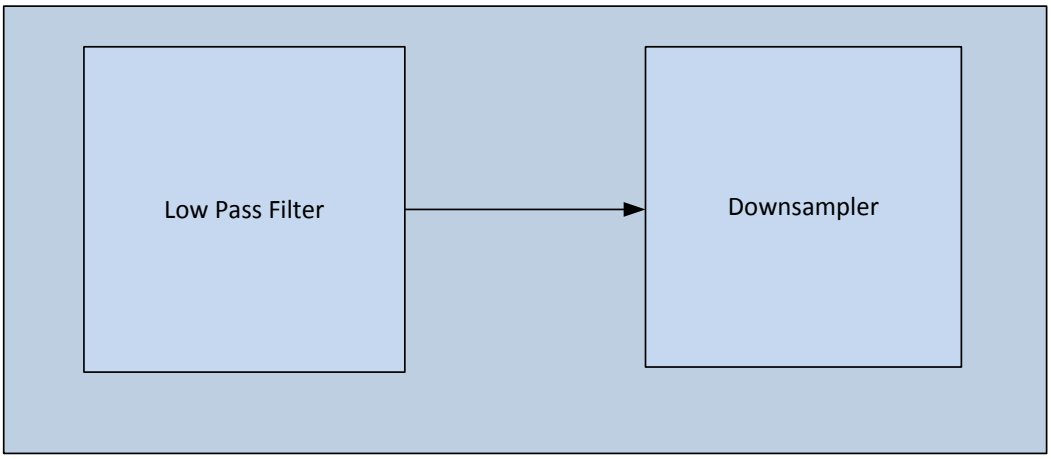

Figure 3.2: A Typical Downsampler

Downsampling is a process of reducing the sampling rate of a signal. The extent of downsampling called downsampling factor should be an integer greater than unity. As an example, if a signal sampled at 2 KHz is downsampled at 1 KHz; the signal is downsampled by a factor of 2. The downsampling factor is denoted by letter M.

In the implementation of downsampler, care should be taken to avoid aliasing. The frequency of signal fed into downsampler should be limited to $\frac{f_s}{2m}$ where f_s is the frequency of the baseband signal. The more the signals spectrum contains signal components whose frequency is greater than$\frac{f_s}{2m}$, the greater the extent of aliasing. Hence, the implementation of a downsampler requires a low pass filter implemented in its front- end to remove the signal components greater than $\frac{f_s}{2m}$ where f_s is the frequency of the sampled signal.

The downsampling of frequencies can be done in single stage or multiple stages. Multi-stage downsampling has an advantage that higher-frequency signals can be routed accordingly to the algorithms requiring higher frequency data, while other algorithms can receive lower-frequency signals tapped from the respective stage of downsampling. Multi-stage downsampling can also reduce the computational overhead because the computational load required by a particular stage depends on the sampling frequency of input data to that stage. A higher order downsampler requires Low Pass Filter (LPF) with sharp cut-off frequency to avoid aliasing. The roll-off of the cut off frequency in the designed LPF depends on the number of poles of the filter. Increasing the number of poles increases the number of multiplication and additions per input sample. Hence, it is computationally efficient to implement the downsampling process in stages where the stages with lower downsampling factors are subjected to high frequency data and the lower frequency data from it are subjected to downsamplers with higher downsampling factors.

In a multi-stage downsampler (Figure 3.3), the multiple downsamplers (D_1,D_2 ,.. D_n) are used to reduce the frequency of the data before the data is sent to computational algorithms (discussed in Section 1.4). The input to D_1 is at the highest frequency and is more computational intensive than D_2, which receives downsampled data from D_1 at a lower frequency. Therefore, the analysis of computations in the proposed architecture of downsampler is necessary to estimate the computations needed in every section of design. This approach can also help in better implementation of algorithms. As an example, the condition monitoring process described in Section 1.4.4 uses Discrete Fourier Transforms (DFT) and the resolution of the signal by the DFT depends on the sampling frequency of the signal.

The Infinite Impulse response (IIR) filter was selected to implement the downsampler as it takes lesser memory than Finite Impulse Response filter. An IIR filter requires fewer

coefficients than an FIR filter, and therefore an IIR filter has requires fewer computations per cycle than an FIR filter [3.6].

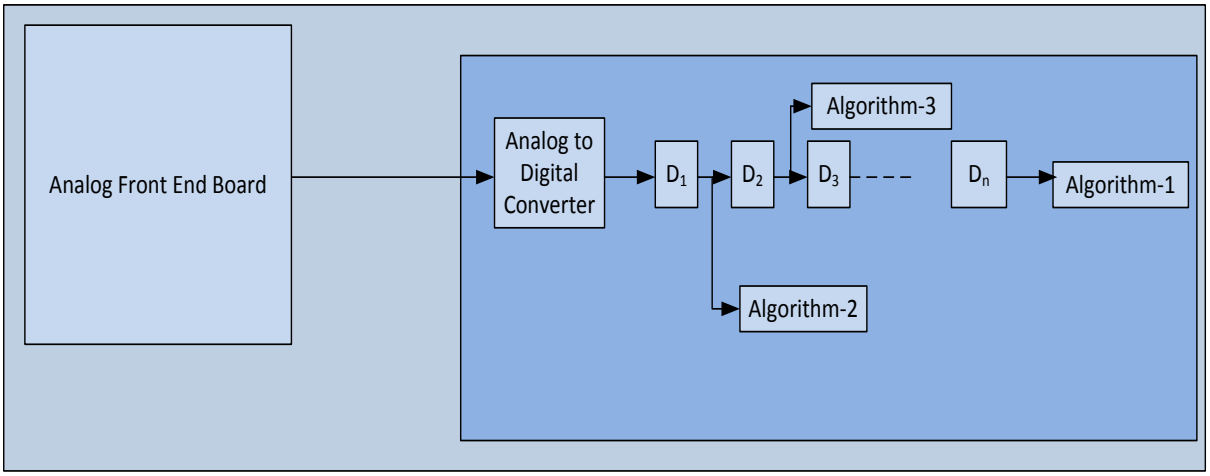

Figure 3.3: Downsampler Implementation in Smart Power Meter

3.2 Estimation of Computation Overhead

A mathematical model was created in Microsoft Excel to estimate the computational overhead of a chain of downsampling stages. The model assumes that, the whole system uses q15 fixed point arithmetic [3.7] to represent the digital data. The selected microcontroller as in Section 0 requires 2 clock cycles for multiplication and a single clock cycle for addition. The number of multipliers and additions per input sample required for an Infinite Impulse Response (IIR) Low Pass Filter (LPF) was calculated from digital filter toolbox in Matlab

Table 3.1: Filter Computations [3.1]

Poles	Multiplication per Input Sample	Addition per Input Sample
16	33	32
15	31	30
14	29	28
13	27	26
12	25	24
11	23	22
10	21	20
9	19	18
8	17	16
7	15	14
6	13	12
5	11	10
4	9	8
3	7	6
2	5	4

The developed model requires the number of poles of filter implemented in each stage. The model gives number of calculations and time to execute these calculations in every stage of downsampling process. The performance parameter includes time for computation per sample (μs), Sampling frequency (Hz) of input data in each stage and computations per second.

3.3 Test Cases

A seven stage downsampler is implemented in the two test cases with each stage having different number of poles. The sampling frequency of the data is reduced in each stage. The reduction in the sampling frequency of the data depends on the configured downsampling factor of each stage. The filters used in all the stages are Infinite Impulse Response (IIR) Low pass filter (LPF).

The following are the test cases implemented in the model:

3.3.1 Test Case-1

The Test case 1 implements a system with higher order pole filters in initial stages followed by lower order filter stages.

Table 3.2: Test Case 1 Implementation of Downsampler System

Stage	Type of Filter	No of poles	Downsampling Factor
1	IIR-LPF	16	2
2	IIR-LPF	10	3
3	IIR-LPF	16	3
4	IIR-LPF	11	5
5	IIR-LPF	3	4
6	IIR-LPF	2	3
7	IIR-LPF	2	2

3.3.2 Test Case-2

The Test case 2 implements a system with lower order pole filters in initial stages followed by higher order filter stages.

Table 3.3: Test Case 2 Implementation of Downsampler System

Stage	Type of Filter	No of poles	Downsampling Factor
1	IIR-LPF	2	2
2	IIR-LPF	3	3
3	IIR-LPF	3	3
4	IIR-LPF	8	5
5	IIR-LPF	7	4
6	IIR-LPF	15	3
7	IIR-LPF	15	2

3.4 Results

Results show MAC (Multiplication and Accumulation) per input sample of each stage, time for computation/input sample of each stage, computations per second of each stage and total processor overhead (ATMEL SAM3SD8 Micro-controller) from the system implemented.

Results:	MAC/Input Sample	Time for Computation/Sample(µs)	Downsampling Factor	Sampling Frequency (Before Computation)	Computations per Second
Stage 1 Computation	65	3.25	2	50000	3250000
Stage 2 Computation	37	1.85	3	25000	925000
Stage 3 Computation	65	3.25	3	8333.333333	541666.6667
Stage 4 Computation	45	2.25	5	2777.777778	125000
Stage 5 Computation	13	0.65	4	555.5555556	7222.222222
Stage 6 Computation	13	0.65	3	138.8888889	1805.555556
Stage 7 Computation	13	0.65	2	46.2962963	601.8518519
					4851296.296
Total		12.55		Processor Overhead	8.09%

Figure 3.4: Results of Downsampler Implementation – Test Case 1

Results:	MAC/Input Sample	Time for Computation/Sample(µs)	Downsampling Factor	Sampling Frequency (Before Computation)	Computations per Second
Stage 1 Computation	9	0.45	2	50000	450000
Stage 2 Computation	13	0.65	3	25000	325000
Stage 3 Computation	13	0.65	3	8333.333333	108333.3333
Stage 4 Computation	33	1.65	5	2777.777778	91666.66667
Stage 5 Computation	29	1.45	4	555.5555556	16111.11111
Stage 6 Computation	57	2.85	3	138.8888889	7916.666667
Stage 7 Computation	57	2.85	2	46.2962963	2638.888889
					1001666.667
Total		10.55		Processor Overhead	1.67%

Figure 3.5: Results of Downsampler Implementation – Test Case 2

The results show that overhead in implementation of test case 2 is $1/8^{th}$ of test case 1. The processor overhead computed for test case 1 and test case 2 are 8.09% and 1.67% respectively.

3.5 Conclusion

The results indicate that the method to improve the computational efficiency of the system is to implement filters with higher number of poles in the final stages of analog data processing. In this implementation, computations per second of each stage will be lower for initial stages where sampling frequency of input signal is high. The efficient approach will also be to implement lower order filters with low downsampling factor in the front end to reduce the computational overhead caused by the later stages of analog data processing.

The implementation of downsampling by using multi-stage filters can also be advantageous to the system where each implemented algorithm in the proposed smart power meter needs data at a pre-defined sampling frequency. As an example, the condition monitoring feature described in Section 1.4.4 uses Discrete Fourier Transform (DFT) to resolve the frequencies of interest. The resolution of the frequencies in DFT depends on sampling frequency of the signal and data block size described in Section 1.4.4.

3.6 References

[3.1] “Filter Design in Matlab”, [Online], Available: http://www.mathworks.com/discovery/filter-design.html , Last Accessed: 11/06/2013

[3.2] “Fixed Point Vs. Floating Point Digital Signal Processing”, [Online], Available: http://www.analog.com/en/content/fixed-point_vs_floating-point_dsp/fca.html, Last Accessed: 11/06/2013

[3.3] C. M. Rader and B. Gold, “Digital filter design techniques in the frequency domain,” *Proceedings of the IEEE,* vol. 55, no. 2, pp. 149–171, 1967.

[3.4] P. Vandewalle, L. Sbaiz, J. Vandewalle, and M. Vetterli, “How to take advantage of aliasing in bandlimited signals,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP ’04)*, 2004, vol. 3, pp. iii–948–51 vol.3.

[3.5] R. A. Gopinath and C. S. Burrus, “On upsampling, downsampling, and rational sampling rate filter banks,” *IEEE Transactions on Signal Processing*, vol. 42, no. 4, pp. 812–824, 1994.

[3.6] “FIR vs. IIR Filter”, [Online], Available: http://www3.spsc.tugraz.at/courses/vwa/papergr6, Last Accessed: 13/18/2013

[3.7] “Introduction to Fixed Point Representation | Shawn’s DSP Tutorials.” [Online]. Available: http://sestevenson.wordpress.com/introduction-to-fixed-point-representation/. [Accessed: 26-Dec-2013].

Chapter 4

System Modeling – Smart Power Meter

4 Overview

A Simulink model was developed to quantify the time it takes for the microcontroller to finish the computations required for the proposed features of smart power meter described in Section 1.4. This time depends mainly on the size of the data and the number of circuits monitored by the smart power meter. This chapter discusses the structure of developed model, the test run and the results obtained from test run. The model gives timing statistics of the operation of the microcontroller in the smart power meter. Timing statistics includes time spent on computations for each of the features described in Section 1.4. The model also gives statistics on the event queue implemented in the microcontroller and data lost in the system.

4.1 Introduction

A microcontroller is defined as a single-chip computer system which contains functional blocks (memory, RAM, ROM etc.) performing different functions.

Microcontrollers operate in highly uncertain environment which depends on the application which in-turn introduces unpredictable flow of data [4.1]. In most cases, the data handled by microcontrollers is not uniform and their sizes grow with the memory and word length of respective microcontroller architectures. In these uncertain environments, if the microcontroller is not configured to the scenario, it can result in data loss caused by insufficient memory in data buffers and unpredicted number of computations performed by the microcontroller as a result of unpredicted flow of data from uncertain environments.

The performance of a microcontroller can be evaluated using formula-based calculation, or through hardware prototyping and measurements. The formula based analysis has the

advantage that it gives detailed insights on relationship between system performances. These are quite useful in the early stages of development but in the later stages of development it is difficult to calculate the performance of complex communication systems using analysis. The performance measurement of microcontroller using hardware prototypes has a disadvantage that hardware prototypes are costly and they are not flexible to the application.

A System-level modeling approach models the individual components of the system in a simulation platform. This approach requires the operational parameters of the individual components from their corresponding data sheets and computational steps from the firmware to be implemented in the system. System level modeling for performance analysis has the advantage that systems can be modeled with any desired level of complexity demanded by the application. The simulation based approach also incorporates the use of formulas and the results from hardware test of the microcontrollers.

One of the most widely used tools for simulating digital systems is Simulink, an extension of Matlab. Simulink's block diagram-based representation of the system model contributes to ease of use.

The objective of system-level modeling of smart power meters is to provide an insight into system performance of the microcontroller with the algorithms implemented as described in Section 1.4 . The performance aspects of the microcontroller are:

- Time spent on computations for each smart power meter features as described in Section 1.4.
- Data lost in the system

The data lost in the system is primarily due to excessive microcontroller computations which then prohibit completion of all computations before the data in the buffer is overwritten by

ADC. Two buffers of pre-defined memory are reserved to receive the data from ADC. At any given point of time, ADC fills data in one of buffer while the algorithms implemented in the smart power meter processes the data in the other buffer.

The model developed yields statistics for the time spent on computations required for the features performed by the smart power meter over the simulation run time. The statistics also includes the data lost in the system.

The test results can be used to modify the algorithms to optimize the time spent on each of stage of data processing as described in Section 1.8. As an example, the computations can be split into stages and these stages can be distributed across time to accommodate more features and avoid data loss in the power meter. The splitting of computations should be designed to allow microcontroller processing of the digitized data in the buffer before the buffer is overwritten.

4.2 The Simulink Model

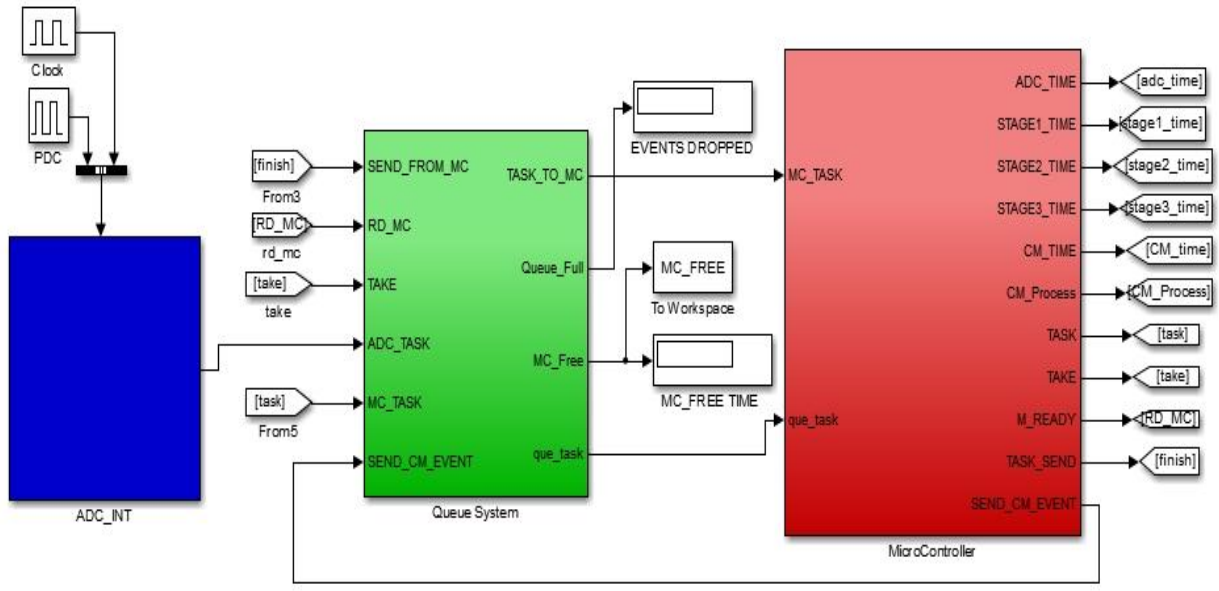

Figure 4.1: Simulink Model

Figure 4.1 shows the primary modules in the Simulink model developed for this thesis.

4.2.1 ADC Interrupt Generator

In Figure 4.1, blue block denotes ADC Interrupt Generator. This block has an event generator with a delay equal to sampling frequency of ADC. This block simulates the sampling interval of analog data in the microcontroller.

4.2.2 Queue System

In Figure 4.1, the green block denotes the Queue System. The smart power meter processes the digitized data from ADC in stages as described in Section 1.8. These stages schedule events in a finite length First in First out (FIFO) queue for the data to be processed by the subsequent stages. The events scheduled by the individual stages of smart power meter are stored in this module and are dispatched to microcontroller module accordingly.

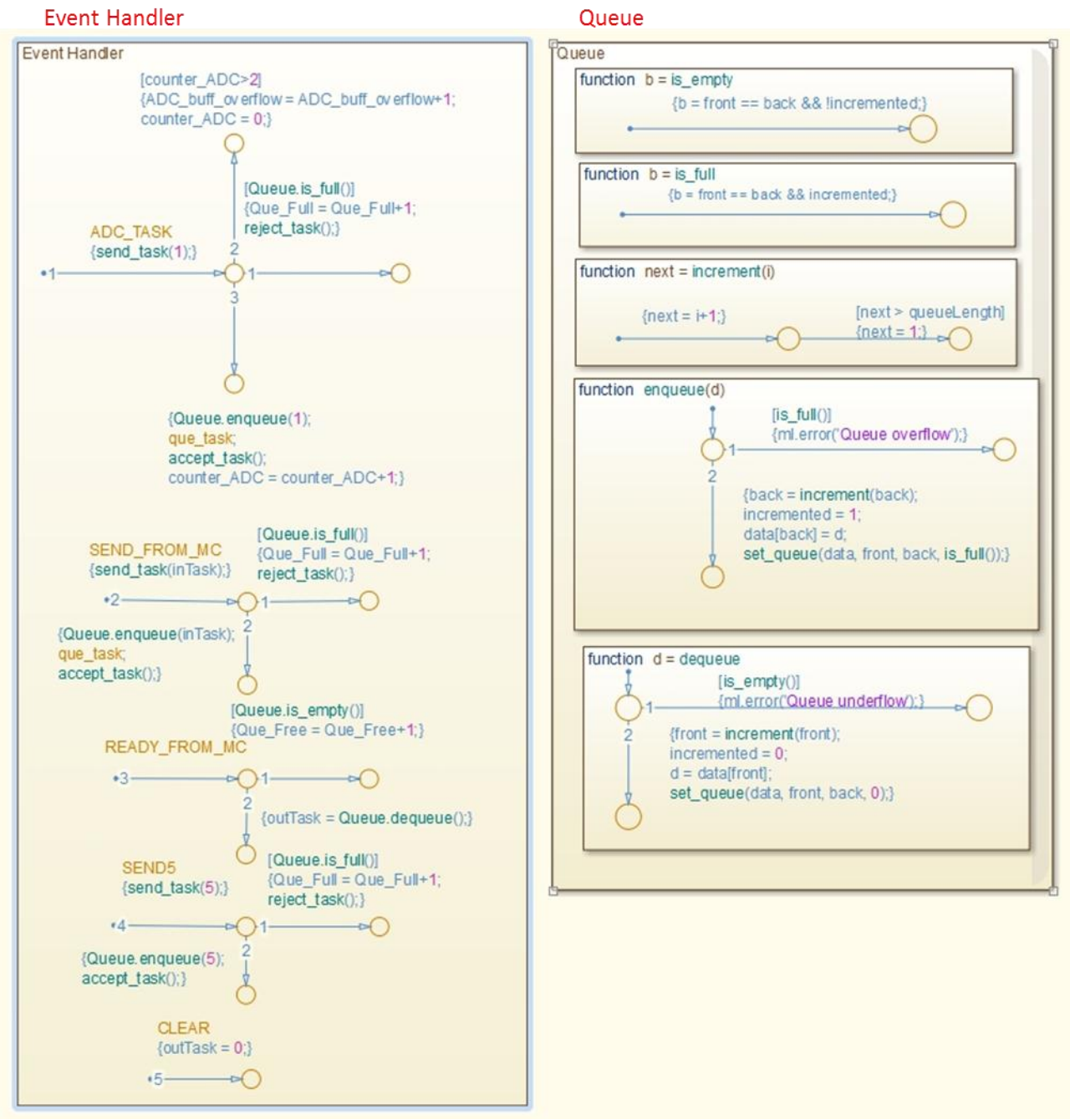

Figure 4.2: Event Queue in Simulink Model

Figure 4.2 shows the FIFO event queue implemented inside queue system block. The Event Handler block performs the task of queuing the events from ADC handler and microcontroller module described in Section 4.2.3.

The following are the events monitored in this module:

- Queue Overflow Event – Queue Overflow Event is the event which occurs when the modules (ADC handler and Microcontroller discussed in Section 4.2.3) in the developed model fails to queue an event. This event occurs when the queue length becomes equal to the number of events in the queue.
- Queue Free Event – Queue Free Event is the event which occurs when the queue does not have any events scheduled. This event shows the time in which the microcontroller was waiting for events to be queued.
- Multiple ADC Interrupt Event – Multiple ADC Interrupt Event is the event which occurs when ADC handler queues an event and the queue has an un-serviced ADC event in queue. This denotes an event in which the data is overwritten in the buffers allocated to ADC.

4.2.3 Microcontroller

In Figure 4.1, the red block denotes the microcontroller. This module forms the core of the system and handles all the events dispatched from the queue. A delay is implemented for each event which corresponds to time required to process the data in each stage as described in section 1.8. The experiment performed to measure this time is discussed in Section 4.2.6.

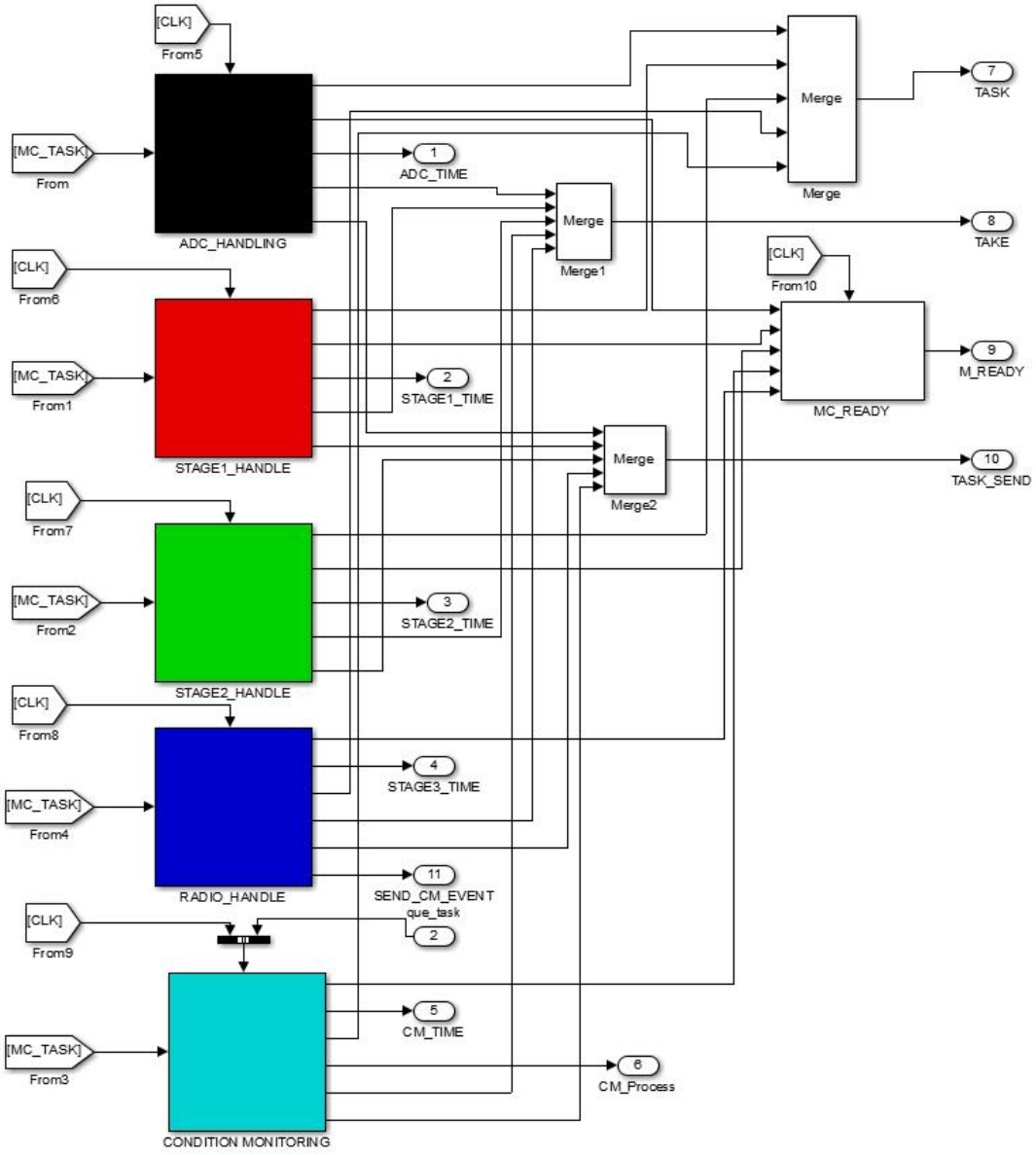

Figure 4.3: Microcontroller in Simulink Model

Figure 4.3 shows the sub-modules inside the Microcontroller Module.

4.2.3.1 ADC Handler

ADC Handler is shown as black block in Figure 4.3. This block receives the event generated by ADC interrupt generator (discussed in Section 0). After a time delay which denotes the processing of data by ADC handler in the firmware, this block generates an event for stage 1.

The different stages of data processing are described in Section 1.8. This models the process of servicing the interrupt generated by ADC in the microcontroller.

4.2.3.2 Stage 1 Handler

Stage 1 Handler is shown as red block in Figure 4.3. This module simulates a time delay for stage 1. After the configured time delay, this block generates an event for stage 2.

4.2.3.3 Stage 2 Handler

Stage 2 Handler is shown as green block in Figure 4.3. This module simulates the time corresponding to time required for calculating V_{RMS}, $I_{RMS,}$ real power and reactive power. After this delay, this block generates an event for radio handler and condition monitoring handler described in Section 4.2.3.4 and Section 4.2.3.5 respectively.

4.2.3.4 Radio Handler

Radio Handler is shown as blue block in Figure 4.3. This module simulates the time delay corresponding to the time required to compress the data before transferring the data into the Zigbee radio buffer. The data compression techniques are described in Section 1.6.

4.2.3.5 Condition Monitoring/Pattern Matching Handler

Condition monitoring handler is shown as cyan block in Figure 4.3. This module simulates a time delay which is the time required by the microcontroller to complete computations for condition monitoring and pattern matching features of the smart power meter. The computations required for these features are described in Section 1.5.2 and Section 1.5.3 respectively. The firmware is implemented in such a way that the microcontroller drops the processing of condition monitoring/pattern matching algorithm if it detects that the buffers allocated to ADC are full. This module also records the event when the condition monitoring/pattern matching is dropped if an event is scheduled by ADC handler.

This algorithm for condition monitoring and pattern matching are described in section 1.4.4 and section 1.4.5. The first stage in pattern matching algorithm is triggered after every power calculation. The rest of the algorithm is executed only if a start/stop transient is detected.

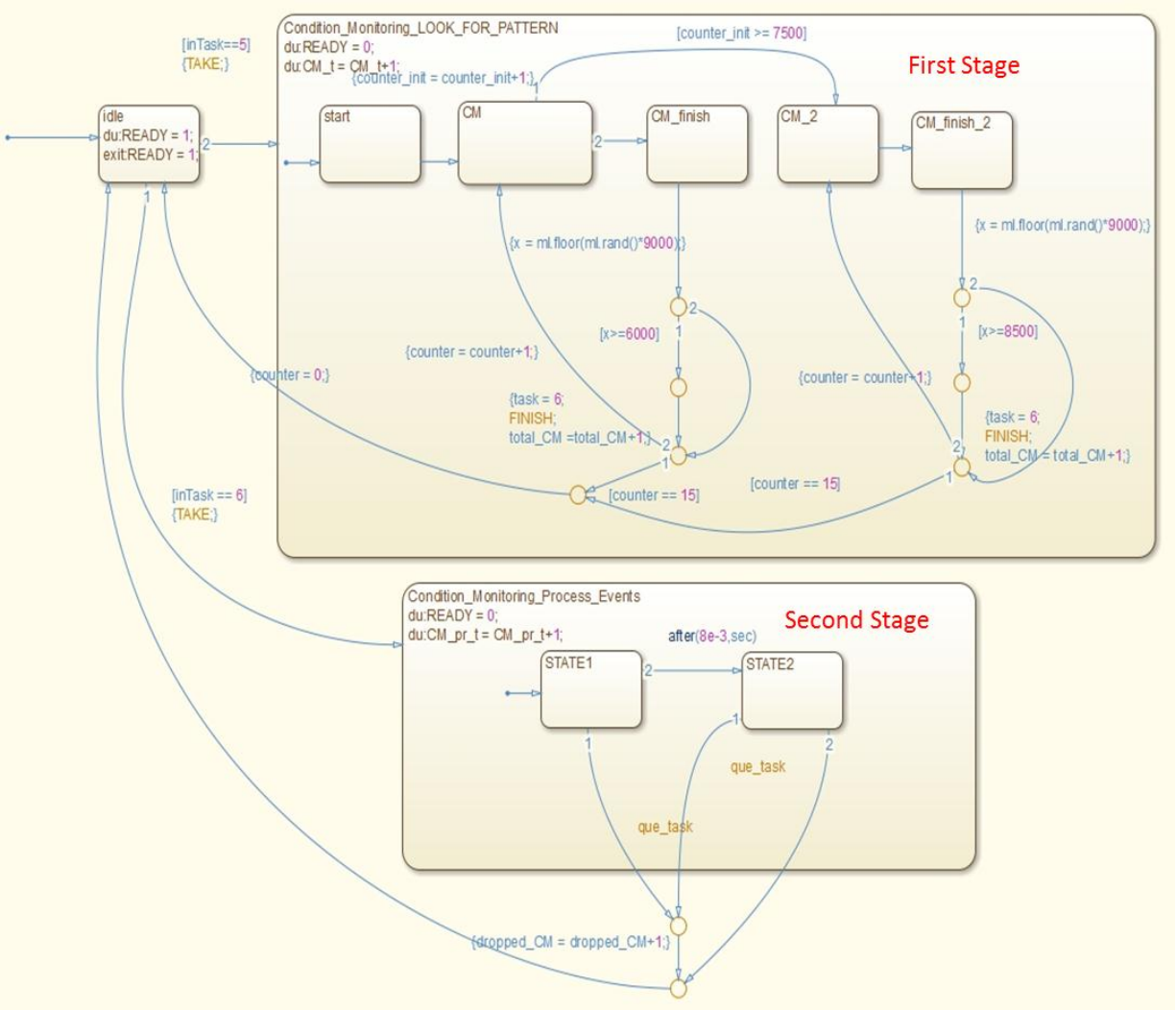

Figure 4.4: Condition Monitoring/Patter Matching Handler

Figure 4.4 shows the condition monitoring handler implementation in the developed model. The handler is implemented in two stages with different time delays corresponding to first and second stage of algorithm in hardware implementation of smart power meter. The first stage is triggered after every power calculation. This stage is triggered by stage 2 module

described in section 4.2.3.3. The second stage is triggered based on probabilistic model to simulate the occurrence of start/stop transients in the electric line being monitored. The developed model triggers the second stage based on simulation run time. In first 0.2% of simulation time, probability to trigger second stage of algorithm over the number of results from second stage is 0.2 thereafter, the probability drops to 0.005. This will simulate a real world scenario where devices are turned on/off in a particular time band of the day creating start/stop transients in the electric line which is monitored by the smart power meter.

4.2.3.6 Events Generated in Microcontroller Module

The following are the events generated by this module:

- Stage 1 Event – Denotes an event from ADC handler for data to be processed by Stage 1 (Section 1.8).
- Stage 2 Event - Denotes an event for data to be processed by Stage 2 (Section 1.8).
- Condition Monitoring Event – Denotes an event for the data to be processed by the condition monitoring/pattern matching algorithm discussed in Section 1.4.
- Stage 3 Event – Denotes an event for the data to be processed before transmitting to the server.

4.2.4 Functioning of Developed Model

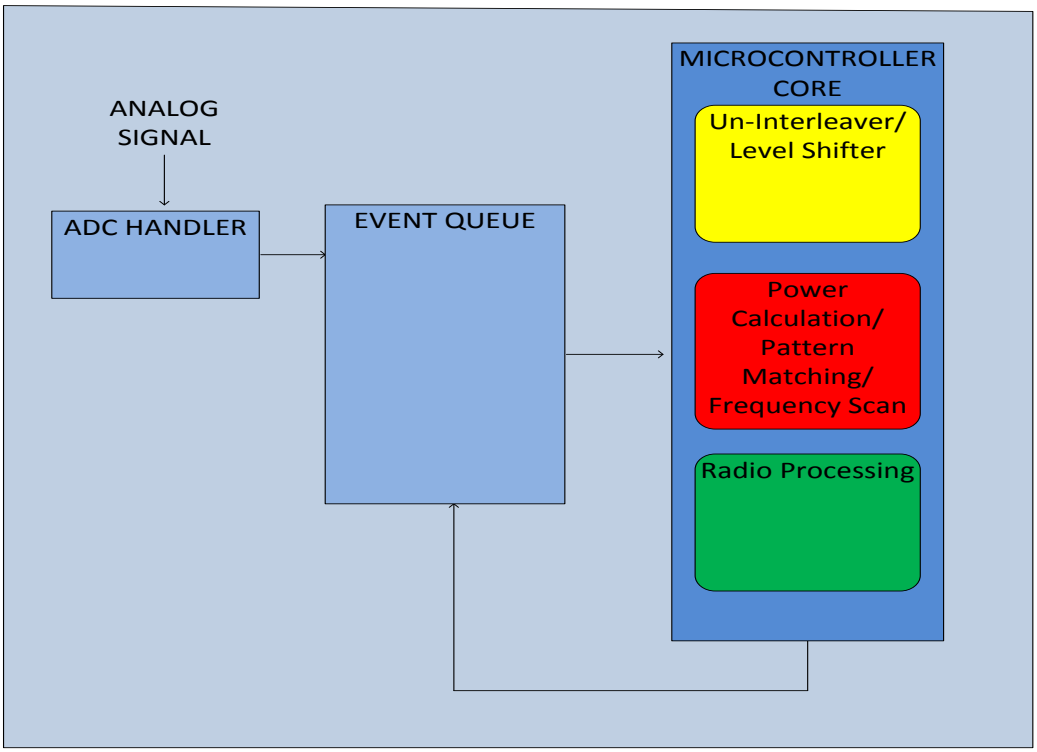

Figure 4.5: Event Flow in Simulink Model

The Figure 4.5 shows broad architecture of the Simulink model which is developed with reference to software architecture of the proposed smart power meter (section 1.8). The yellow, red and green blocks in Figure 4.5 are the sub-modules of the microcontroller described in Section 4.2.3. The event queue and ADC handler blocks in Figure 4.5 are described in Section 4.2.2 and Section 4.2.2 respectively. ADC handler generates events at a pre-defined rate, simulating the sampling interval in the microcontroller. This event is queued in the event queue. Stage 1 (Un-Interleaver/Level Shifter) simulates a process in the microcontroller where the data is pre-processed before power calculation. The pre-processing of the data is described in Section 1.8.1. Stage 2 (Power calculations, pattern matching and condition monitoring) simulates the process of power calculation, pattern matching and condition monitoring performed by the microcontroller. Stage 2 creates an event for stage 3 after a pre-configured number of power

calculations. This simulates the process of transferring the data to data compression algorithms. Stage 1, Stage 2 and Stage 3 in the developed model receives their corresponding events and waits for a pre-configured time before scheduling an event depicting the processing of data in the next step. The experiment performed to measure the time required to process the data in stage 1, stage 2 and stage 3 is described in Section 4.2.6.

4.2.5 Test Parameters

The Simulink model uses the time to perform the computations for the corresponding features of the proposed smart power meter as inputs. This time depends on the buffer size allocated to ADC. The following are the test parameters in the developed simulation:

- Sampling frequency: The frequency at which analog signal is sampled in ADC.
- ADC Buffer Size: The size of buffer reserved to receive sampled digital values.
- Event Queue Length: The maximum number of events supported by the implemented queue in the system.
- ADC Handler Time: The time to fill in ADC buffer. This time depends on ADC buffer size and sampling frequency.
- Stage 1 Time: The measured time to process the data from ADC buffer. Stage 1 performs the function of un-interleaving data with level shifting the voltage and current values.
- Stage 2 Time: Measured time to process the data from stage 1. Stage 2 performs the function of calculating V_{rms}, I_{rms}, Real Power and Reactive Power.
- Radio Time: Measured time to transfer the data from microcontroller to radio buffer.
- Condition Monitoring/pattern matching Time: Measured time to implement the condition monitoring/pattern matching algorithm to the data from stage 2.

4.2.6 Measurement of Test Parameters

Sampling frequency and buffer size allocated to ADC are considered primary parameters. The parameters excluding sampling frequency and buffer size listed in Section 4.2.5 are considered secondary parameters. The power meter is given pseudo-analog data on its 15 input channels to measure time required to complete the computations for the features (Section 1.4) of the smart power meter.

The primary parameters are fixed in the firmware of the proposed smart power meter. The firmware is executed and the secondary parameters are measured using logic analyzer [4.3]. As the microcontroller starts a stage of computation, a GPIO pin in the SAM3S evaluation board is raised to high. The pin is set to logic 0 when the microcontroller exits the particular stage. The resulting square wave in GPIO pin is captured by logic analyzer and analyzed. The duty cycle of the captured square wave is the time spent by the microcontroller to process the data in that particular stage.

4.2.7 Simulation Parameters

The Table 4.1 shows the fixed primary parameters. Table 4.2 shows the measured secondary parameters after the implementation of the firmware in the selected microcontroller (Atmel SAM3SD8C) in Section 0

Table 4.1: Primary Configuration Parameters

Simulation Run-Time	10 minutes
Sampling Frequency	3840 Hz
ADC Buffer Size	1920 Bytes

Table 4.2: Secondary Configuration Parameters

ADC Handler Time	3.84×10^{-4} seconds
Stage 1 Handler Time	1.17 ms
Stage 2 Handler Time	4.23 ms
Radio Handler Time	1.00 ms
Condition Monitoring/Pattern Matching Handler Time	8.00 ms

4.3 Results

Table 4.3: Test Results

Parameter	Result
Queue Overflow Event	0
Total Events from ADC Handler	37500
Multiple ADC Interrupt Event	1852
Probability of ADC buffer being overwritten	0.04
Total no of Condition Monitoring events queued	6.18×10^4
Condition Monitoring events dropped	1.14×10^4
Probability of Condition Monitoring event to be dropped	0.18

The results reveal that no events were dropped from event queue which implies that there was no overflow in queue and the queue size configured was sufficient to handle all the generated events. This result supports the hypothesis that an event queue of 15 is ideal for the proposed configuration. The probability of ADC buffer over-write event over the total number of ADC events scheduled was 0.04. There is a 4% chance that ADC buffer will be overwritten because microcontroller was performing other tasks such as stage1, stage2 or stage3 and it could not free the buffer before the data was overwritten. This leads to loss of un-processed data from analog sensors. The probability of the condition monitoring/pattern matching algorithm quitting before completing all condition monitoring/pattern matching events were generated was 0.18.

There is 18% chance that microcontroller got an ADC interrupt when it was processing data. In this case, microcontroller dropped data it was processing and was servicing ADC interrupt event. This event can lead to loss of information expected from condition monitoring algorithm.

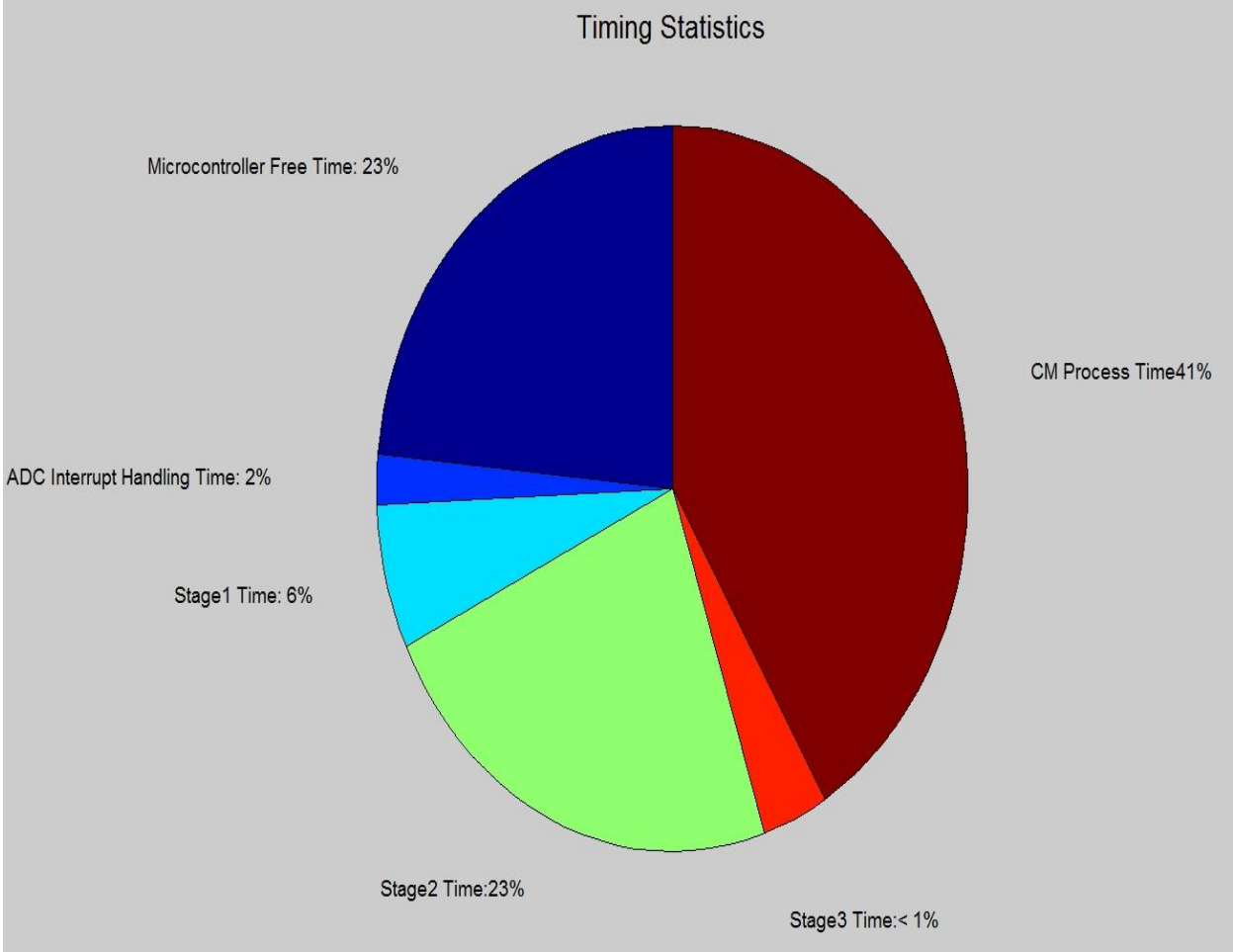

Figure 4.6: Timing Distribution (Stages in Algorithm)

The timing distribution diagram shows the percentage of time spent in each handler of the Simulink model. The distribution also shows percentage of idle time of the micro controller. This idle time can be allocated to prospective functions of smart power meter.

4.4 Conclusion

The developed model predicts critical information about the system which includes statistics on data and results loss. The model also gives timing statistics of different handlers in the power meter. These results can be used to distribute or prioritize algorithms such that data is never lost as it is processed through the system.

4.5 References

[4.1] Prof. Dr. Ir. Joost-Pieter Katoen “Software Modeling and Verification”, [Online], Available: http://www-i2.informatik.rwth-aachen.de/i2/erlang0/, Date Accessed: 11/11/13

[4.2] Renesas DevCon “Simulation: Expert Insights into Modeling Microcontrollers”, [Online], Available: https://www.semiwiki.com/forum/content/1770-simulation-expert-insights-into-modeling-microcontrollers-renesas-devcon.html, Date Accessed: 11/11/13

[4.3] “Logic 16”, [Online], Available: http://www.saleae.com/logic16, Date Accessed: 11/11/13

[4.4] “SAM3S ARM Cortex-M3 Microcontrollers”, [Online], Available: http://www.atmel.com/products/microcontrollers/arm/sam3s.aspx, Date Accessed: 11/11/13

[4.5] E. Bilhan, P. C. Estrada-Gutierrez, A. Y. Valero-Lopez, and F. Maloberti, “Behavioral model of pipeline ADC by using SIMULINK(R),” in *2001 Southwest Symposium on Mixed-Signal Design, 2001. SSMSD*, 2001, pp. 147–151.

[4.6] M. Kezunovic, E. Soljanin, B. Perunicic, and S. Levi, “New approach to the design of digital algorithms for electric power measurements,” *IEEE Transactions on Power Delivery*, vol. 6, no. 2, pp. 516–523, 1991.

Chapter 5

Wireless Sensor Network Simulation - Zigbee Network

5 Overview

This chapter presents a wireless sensor network simulation in which Zigbee sensor nodes communicate to a single server responsible for recording the results from the smart power meter. Section 5.1 introduces Zigbee device types, an overview of Zigbee based energy metering, and the need for discrete event simulation to simulate a Zigbee network. This is followed by section detailing Zigbee frame structure and the steps in frame transmission. Section 5.2 discusses the structure of the proposed wireless network for power meters followed by test scenarios and results from the test scenarios.

5.1 Introduction

Zigbee is built using IEEE 802.15.4 standard. The IEEE 802 working group evaluates the standards and continuously makes changes to the standard to ensure reliable communication in Zigbee networks. Zigbee works on 2.4 GHz, 915 MHz and 868 MHz frequency bands. The 2.4 GHz band is for global use and has 16 channels [5.1]. The 915 MHz band is mainly used in North America and Australia, and has 10 channels. The 868 Hz band is used in European Union and has 1 channel.

Zigbee frequency bands of operation are in ISM (Industrial Scientific and Medical) bands. These bands are license-free and designers and implementers of Zigbee should take precautions to avoid interference with other wireless devices operating in the same frequency band. The major users and possible interferers to Zigbee networks operating in 2.4 GHz bandwidths are 802.11b/g/n networks (Wi-Fi), Bluetooth, cordless phones, microwave ovens and WiMax networks.

5.1.1 Zigbee Device Types

The Zigbee device types [5.1] are described in the following subsections:

5.1.1.1 Zigbee End Device

Zigbee End Devices are attached to sensors collecting data and their basic function is to send the data to a destination node. These devices are typically in sleep mode when not transmitting data thus reducing power consumption in the wireless nodes.

5.1.1.2 Zigbee Router

Zigbee routers routes the data through a Zigbee network to a destination node. The Zigbee router receives the data from Zigbee end devices or other Zigbee routers. The routes are calculated by using distance vector algorithm. The distance vector algorithm calculates the route by selecting a route which has least number of hops (routers) between sender and receiver node. Zigbee routers can buffer the data received from Zigbee end devices before transmission which saves significant power in the Zigbee end device. The Zigbee router may have memory to support data buffering and hence it becomes expensive as the memory increases. The Zigbee routers can also act as range extenders for the Zigbee network.

5.1.1.3 Zigbee Coordinator

The Zigbee coordinator forms the root of the network and performs the function to select a wireless channel of operation in the event of interference with other wireless systems operating in the same frequency as wireless nodes in a Zigbee network. The Zigbee coordinator also acts as a bridge between Zigbee networks for routing the packets between networks.

5.1.2 Discrete Event Simulation

Software simulation of wireless sensor networks is gaining wide acceptance because hardware simulation of wireless systems for optimizing the performance of the network is costly.

Reproducibility of results from wireless sensor network simulation is difficult. This is because many wireless sensor network parameters which include cross talk, interference with other communication devices and ground reflection vary unpredictably. Using a test bed also has the disadvantage that it cannot simulate a very wide deployment of sensors.

Discrete event simulation is widely used to simulate large wireless sensor networks. In discrete event simulation, each event occurs at a particular time and denotes a change in state of system. Discrete event simulation assumes that there is no change in the system between consecutive events. Thus time in the simulation jumps from the time of one event to the time of the next event without considering inter-event times.

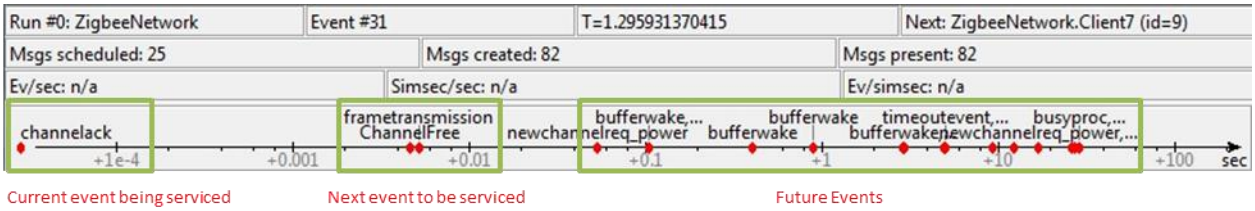

Figure 5.1: Event Queue (OMNET++)

The simulation software lists pending events to ease debugging. Global variables can be declared and are used to record bandwidth, throughput and packet loss of the entire wireless network. Breakpoints/debugging breaks can be used to inspect the code without disrupting the execution of the program. The following are some of the popular discrete event simulators:

5.1.2.1 NS-2

NS-2 is abbreviation of Network Simulator version two. NS is a variant of REAL network simulator developed by DARPA in 1989. NS-2's key feature is that it can support a wide range protocols from physical to application layer. NS-2 is open source and saves significant software cost for characterizing sensor networks.

The NS-2 also has limitations. NS-2 runs only in Linux operating systems. The user has to be proficient with writing scripting language. Tool Command Language used in NS-2 is difficult to debug. NS-2 has no graphical user interface. NS-2 is a general network simulator and cannot model specific functions of sensor networks such as bandwidth, throughput or energy usage.

5.1.2.2 Matlab

Matlab is widely used for wireless sensor network simulation. Matlab/Simulink is developed by Mathworks and supports many platforms including MAC OSX, Windows and Linux. Matlab supports the detailed simulation of nodes and architecture in the nodes and has rich and diverse libraries to model channels, noise schemes, interference, propagation delay and distance between nodes.

Limitations of Matlab become pronounced in large wireless sensor network simulation. Simulation time increases with complexity and number of sensor nodes. Matlab license fees can add to cost for developing a wireless sensor network simulation.

5.1.2.3 TOSSIM

TOSSIM is an open source emulator which runs on TinyOS. TOSSIM was developed at UC-Berkley and is built on Python, which many users find easier to code than C. TOSSIM can support very large number of sensor nodes and can accurately simulate real world scenario.

Limitation of TOSSIM is that it runs only within TinyOS platform.

5.1.2.4 OMNET++

OMNeT++ is a C++ based library which is primarily used for building network simulators. Network development in OMNeT++ is based on GNU Eclipse IDE, a graphical runtime environment. Advantages of OMNeT++ include [5.2]:

- Enables large scale simulation with the simulation components built from reusable blocks sub-classed from library.
- OMNeT++ has an advantage that it reduces the problem of long runtime and debugging periods common to Matlab.

The Limitations of OMNeT++ include support of limited number of protocols which in turn limits its applications. Wireless network simulation for Zigbee network discussed in this chapter uses OMNET++ platform to simulate a network of the smart power meters.

5.1.3 Zigbee based Energy Metering

Zigbee is widely employed in industry and home automation systems. The industrial application of Zigbee radios are electric power meters and fault monitoring systems. The Zigbee also finds extensive application in home automation and energy metering. The prominent advantage of Zigbee over Wi-Fi is the power saving features in Zigbee sensor nodes, which enables the nodes to work for years on battery power [5.2]. Zigbee also introduces low-power, low-cost, hardware for energy metering.

One of the common applications popular in industrial and household wireless sensor network is used to monitor power consumed by all electric machines (Section 1.2). The smart power meters developed in this project are deployed in subpanel which transmits V_{RMS}, I_{RMS}, real power and reactive power to Zigbee node which acts as a server to collect the measurement parameters of the smart meter. The sensor nodes may detect conditions in electric circuits that cause large amounts of data to be transferred through the network to server. This will cause congestion in the network which in turn will cause the Zigbee nodes to back-off from transmission and an attempt will be made to gain channel access (discussed in Section 5.1.5.1). This will cause unpredictable data flow in the network. The information from the sensors which

detects an event such as possible fault in an electric machine is critical. Loosing this information may lead to missing diagnosis of the system which can lead to complete damage or costly repairs of the machine. Thus, the packet loss in the network should be avoided.

5.1.4 Zigbee Data Communication

The Zigbee communication protocol is built on IEEE 802.15.4 standard. The physical layer and MAC (Medium Access Control) specifications are specified in IEEE 802.15.4 [5.3]. The standard also provides protocols for network formation, node addressing and transmission scheduling among Zigbee nodes. The preamble and start delimiter segments are attached to data frames by physical layer. The standard limits the maximum data frame coming from physical layer to 133 bytes. The preamble and delimiter segments are 6 bytes long; this limits the data frames coming from MAC layer to 127 bytes. The MAC header along with the network layer header is 40 bytes long limiting the maximum size of payload to 87 bytes.

5.1.5 Zigbee Packet Transmission

The first step in Zigbee packet transmission is to access a wireless channel to transmit the data. The nodes use Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) in order to access a wireless channel. The time to complete CSMA-CA process increases with increase in channel activity. Assuming, that the packets are transmitted without data packets errors, the data frame is transmitted from the client to destination node after an acknowledgement signal is acquired from CSMA-CA. After receiving the data frame, the destination node will send an acknowledgment packet. This packet is sent after a turn-around time discussed in Section 5.1.5.3. Subsequent frames are sent (Time between frame transmission is discussed in Section 5.1.5.5) after the client receives the acknowledgment packet from the destination.

5.1.5.1 CSMA-CA

The Zigbee nodes use Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) to access a wireless channel in the deployed environment. In CSMA-CA, if the node finds the channel is not available for transmission, a subsequent attempt for channel access is carried out after a time interval. This time interval is random number of symbol periods. The symbol rate of Zigbee radio is fixed at 63500 symbols/second [5.3]. The symbol period is 320μs. The random number is between 0 and (2^{BE}-1). The BE (Back-off Exponent) is fixed at 31 [5.3].

5.1.5.2 Data Transmission

This stage is executed if the wireless channel is acquired by CSMA-CA process. Assuming that the transmitted data does not contain any errors, the time to transmit a data only depends on the size of the data. Maximum size of a data frame in Zigbee networks is 133 bytes. The throughput for Zigbee network is 256 kb/s. Hence, the time to transmit a single frame is calculated as 4.15 ms.

5.1.5.3 ACK Turn-Around Time

After the last octet is received in a destination node, the destination node radio sends an acknowledge packet indicating the successful reception of the data packet. In order to send this acknowledgement packet, the Zigbee transceiver must switch from receive to transmit mode. The time required for a node to send an acknowledgement packet to sender node is defined in IEEE 802.15.4 [5.3] as TurnaroundTime which is equal to 12 symbol periods or$185\ \mu s$.

5.1.5.4 ACK Transmission Time

After receiving a data frame from the Zigbee node attached to the smart power meter, the server sends an acknowledgement packet to the corresponding node. The acknowledgement packet in the simulated scenario is 11 bytes long and requires $343\ \mu s$ for transmission.

5.1.5.5 Time between Data Frames

The time between transmissions of subsequent data frame (if size of payload is greater than 133 bytes, the maximum size of Zigbee data frame calculated in Section 5.1.4) is defined in IEEE 802.15.4 as Inter-frame Spacing (IFS). The IFS value can take a value between two constants aMinLIFSPeriod and aMinSIFSPeriod. The time between data frames in the developed simulation is 40 symbol periods or 0.61 ms (Maximum IFS period in defined in IEEE 802.15.4).

5.2 The Proposed Wireless Sensor Network

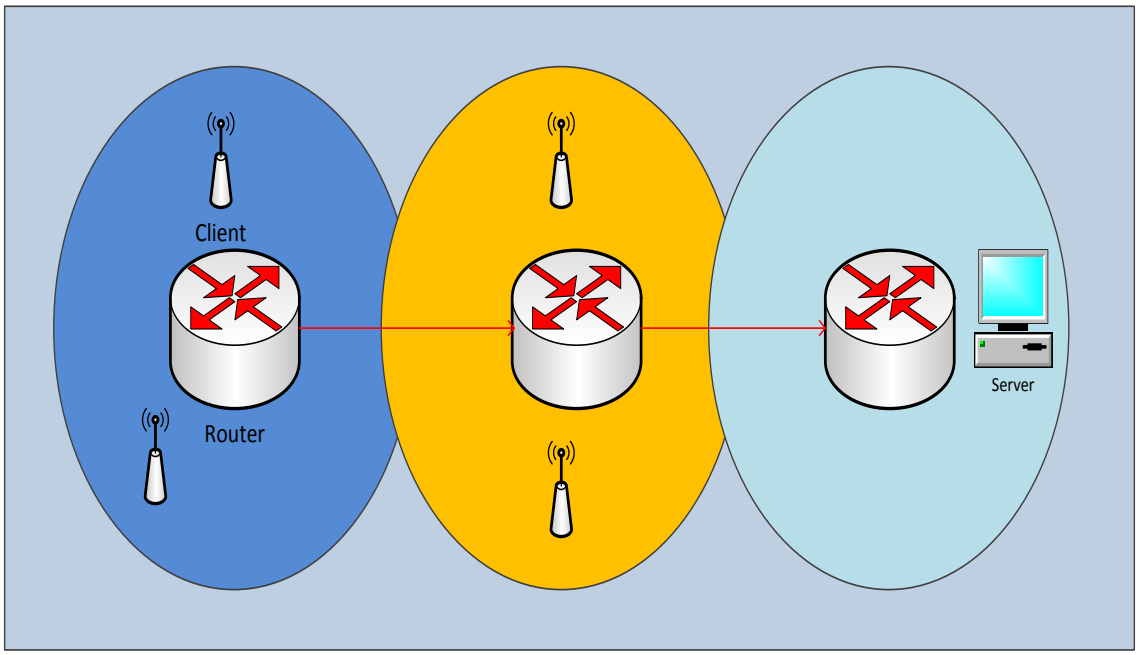

Figure 5.2: Proposed Zigbee Network

The Figure 5.2 shows the proposed Zigbee network which consists of a set of clients attached to the proposed smart power meter. These clients transmit the data from the smart power meter to a server which collects the data from all the smart power meters in the network. The clients are deployed in a subpanel. The subpanel implementation of the smart power meters are described in Section 1.3.1. The server is a workstation with a Zigbee end node attached to it

to receive the data from the power meters. The blue, orange and cyan spheres show the range of three Zigbee routers in the scenario. A single data packet contains real power, reactive power, RMS values of current and voltage from electric circuit concerned with the client. The clients send the data packets to routers placed in their range of operation which forwards the packet to the data server. The routers have memory to buffer the data received from the Zigbee end nodes. The routers transfer the data to the server or another router in a pre-defined time interval. In the process of data transfer between routers, the router shifts its frequency of operation to the frequency of operation of destination router.

In the first step of sending the data in Zigbee network, the sending device (client/router) uses Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) in order to gain access to a wireless channel. CSMA-CA recommended in IEEE 802.15.4 specifies a maximum of 5 attempts to access a wireless channel [5.3]. After 5 attempts, the transmission attempt is cancelled and the data is lost. The routers can retain the data after cancellation of transmission if their buffer is free to accept the data but the clients will lose the data as soon as the data is over-written in the buffer. The use of Zigbee routers can also help in increasing the range of network. The data transfer between routers is in pre-defined intervals. The downside of using routers in a network is that it increases the latency of transmitted data packets in the network. The buffered data in the routers is only transmitted at a pre-defined interval thus increasing the latency of the data packets in the network.

In order to reduce data loss and to get the best throughput of the network, buffer size and inter-transmission time of routers should be optimized.

5.2.1 Assumptions

The following assumptions were used to simulate the proposed Zigbee network:

- Packet error rate is zero in the duration of simulation.
- Size of data packet and frame is constant (1680 bytes) in the duration of simulation.
- Routing tables in the routers are pre-compiled. So, the time to compute the shortest route using Distance Vector (DV) [5.3] algorithm is neglected.
- Channel activity or effect of interference/fading in network throughput is neglected in the simulation.
- Unless specified, throughput between all devices is 250kb/s.

5.2.2 The Simulation

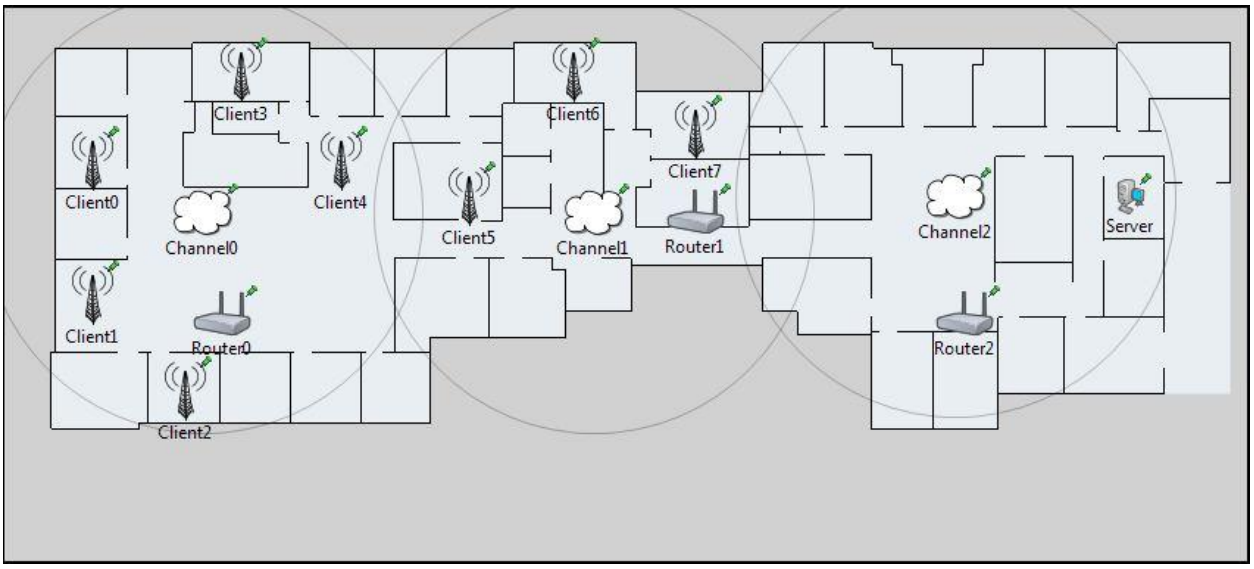

Figure 5.3: The Zigbee Network (OMNET++ Environment)

A Zigbee network of the proposed smart power meters used in a building energy monitoring system was simulated in OMNET++. The following are the main modules in the developed model:

5.2.2.1 Client Module

The client modules are Zigbee end nodes attached to the proposed smart power meters. Each client module transmits the data to a router module installed in range of the client module. Operational ranges of the two routers are shown by circle in Figure 5.3. The clients generate the data packets which are Run Length (RL) encoded in accordance with a probabilistic distribution which in-turn depends on the elapsed simulation time (T). The probabilistic distributions used for the data packet generation in client modules are shown in Table 5.1. The client modules also record the number of data frames lost. This frame loss is due to failure of CSMA-CA process to access a wireless channel.

Table 5.1: Probabilistic Distribution for Power Packet Generation

Time (Seconds)	Distribution
0 – T/4	Uniform(3s,0)
T/4 – T/2	Uniform(30s,0)
T/2 – 3/4 T	Uniform(3s,0)
3/4 T – T	Uniform(30s,0)

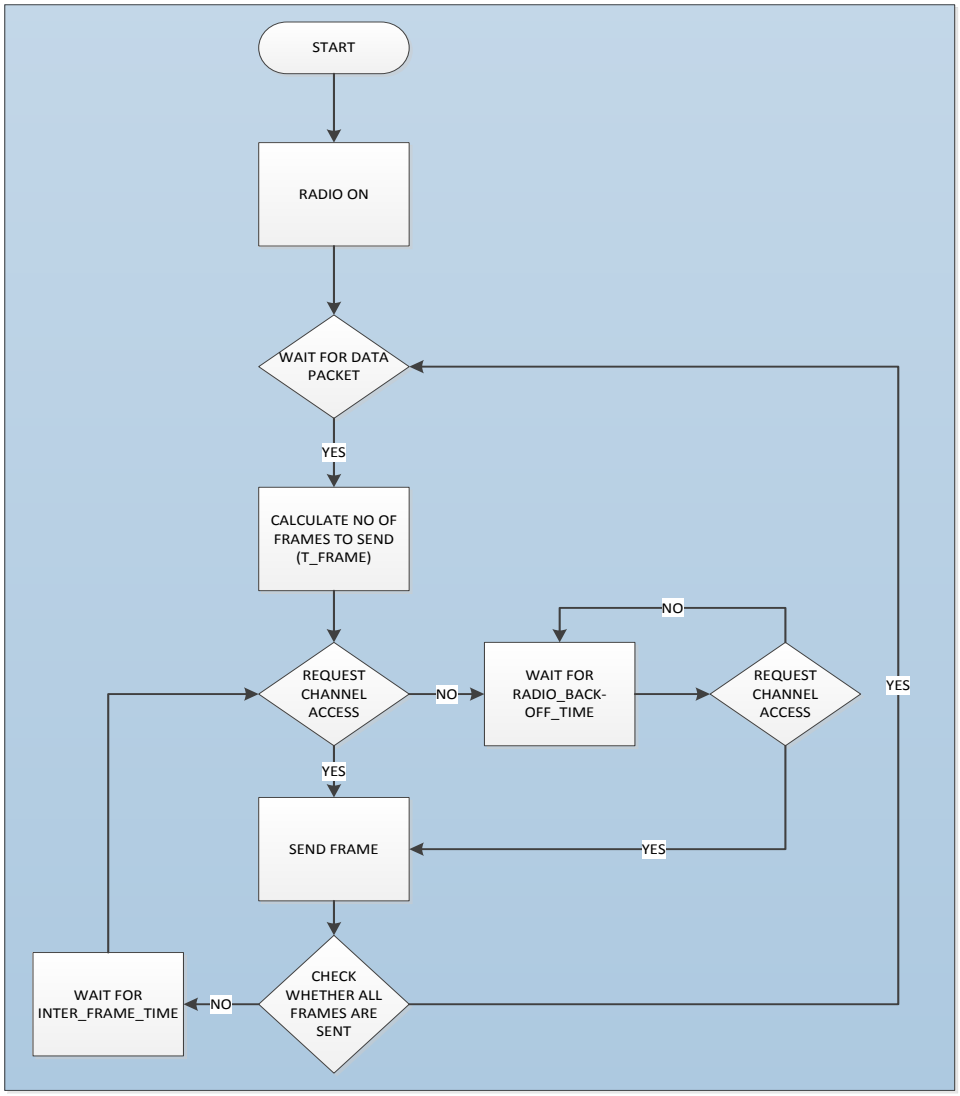

Figure 5.4: State Diagram – Client

Client waits for the generation of data packet. Once the data packet is ready for transmission, request for channel is sent to channel module described in Section 5.2.2.2. If request for channel is denied by the channel module, client waits for a configured time (random time in CSMA-CA described in Section 5.1.5.1) before it retries for a channel. If retry attempt for channel access fails 5 times, the packet is dropped in accordance with the CSMA-CA protocol enlisted in IEEE 802.15.4. Once the client gets the access to channel, it starts sending

data as frames. In the case for sending multiple frames, client waits for a configured time (inter_frame_period) before it sends subsequent frames.

5.2.2.2 Channel Module

The channel module in Figure 5.3 simulates the channel of communication between devices. This module controls the channel access time of the client and router modules (described in Section 5.2.2.4). The access time allocated to a Zigbee device is according to the size of payload from the Zigbee device.

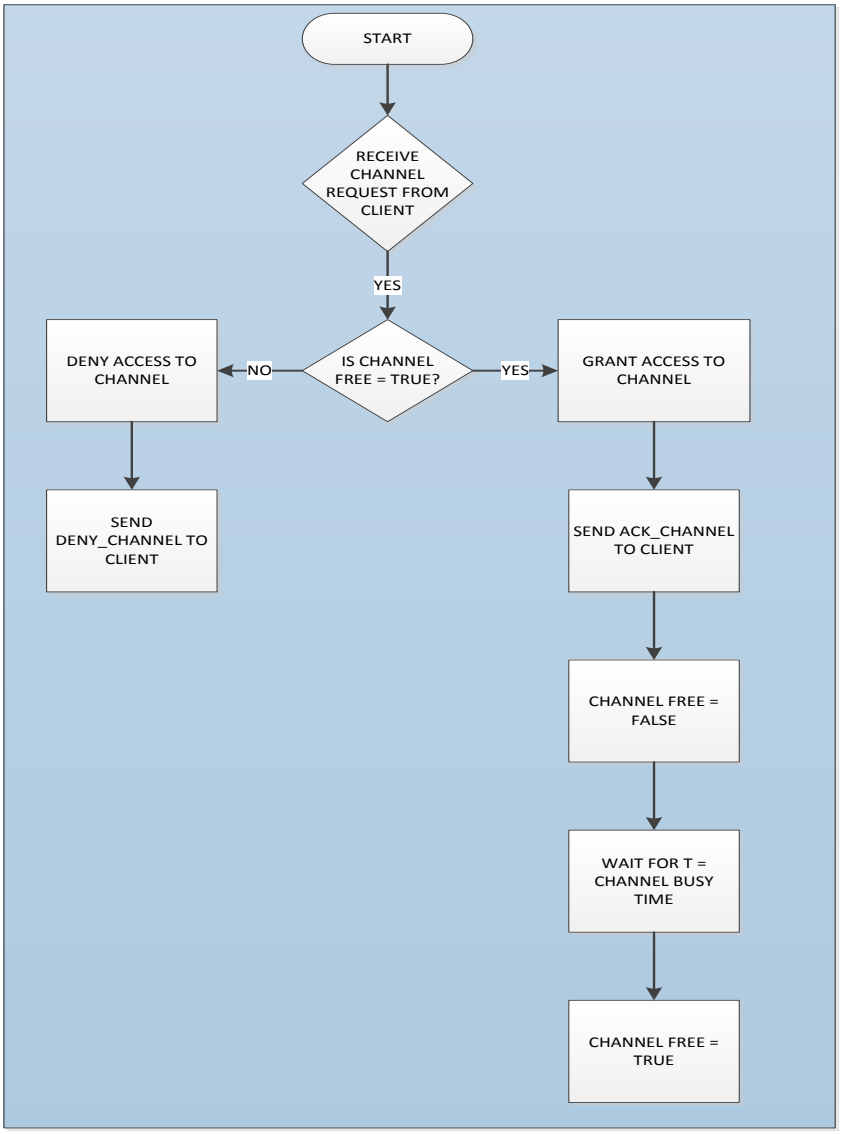

Figure 5.5: State Diagram – Channel

Channel module waits for 'request channel' message from client module. When the channel module receives the request message from client module, it checks for an active allocation of the wireless channel to any device. If channel is free, it returns a message 'Channel ACK' to the client and client starts transmitting the packet. In the case of busy channel, 'Channel Reject' message is sent to the client.

5.2.2.3 Server Module

The server module receives payload generated by client modules and is working under Router3 as shown in Figure 5.3. After receiving data frame from client modules, the server module sends an acknowledgement packet (described in Section 5.1.5.4) to the corresponding client module.

5.2.2.4 Router Module

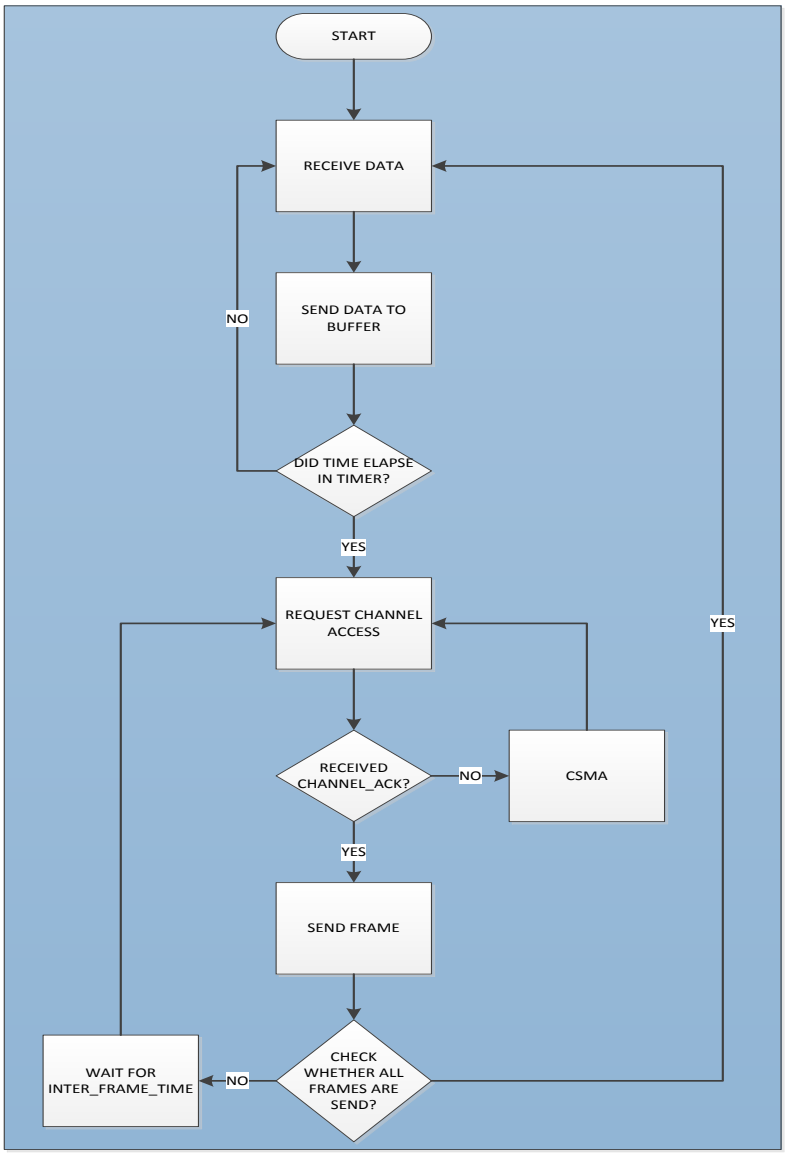

Figure 5.6: State Diagram - Router

The router module has a timer which keeps track of time elapsed. The timer is configured to inter-transmission time of routers (configured according to test scenario). The router buffers the data frames from the client and acknowledgement packets from the server. After the configured inter-transmission time in routers has elapsed, the router attempts a channel access to forward the buffered data to their corresponding destination.

5.2.2.5 Test Scenarios

The rate of generation of data in client module is fixed across all test scenarios and is described in Section 5.2.2.1. In all the scenarios the server sends an acknowledgement packet described in Section 5.1.5.5 of size 11 bytes to the client after it receives a data frame from a client module. The following results are collected in all the test scenarios:

- Mean and maximum length of queues in all the routers inside the network.
- Frame loss in the network.
- Throughput of the network.

The simulation is repeated and results are collected for the following test scenarios:

Scenario 1: Router to router transmission time of 0.2s, 0.5s and 1.5s.

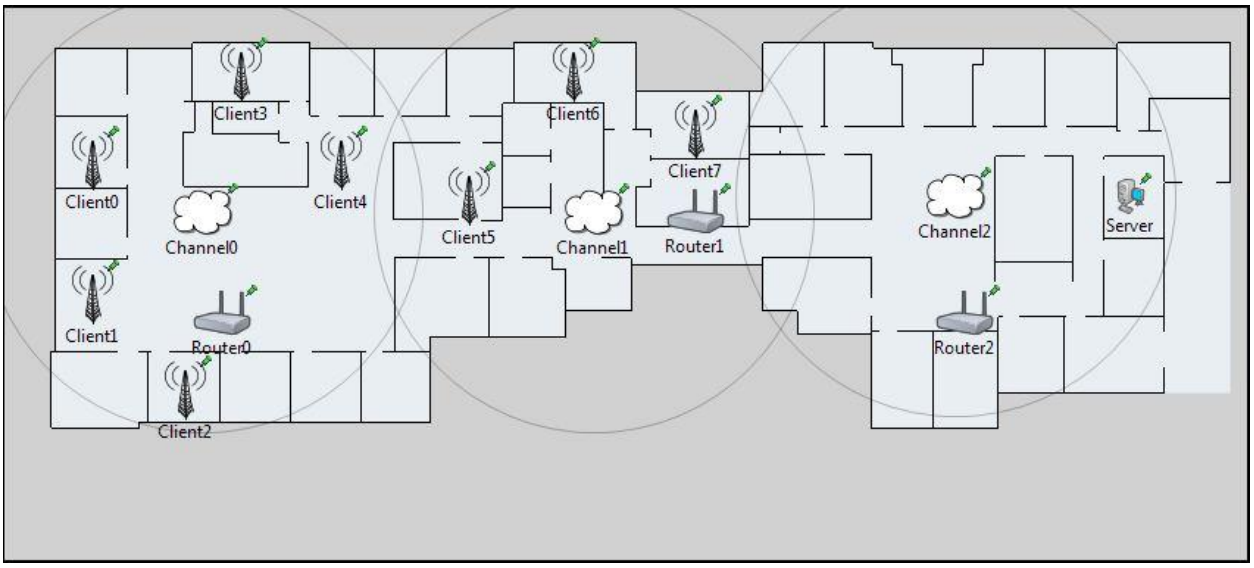

Figure 5.7: The Zigbee Network (Scenario 1)

Scenario 2: Change in Network Topology (Adding extra routers)

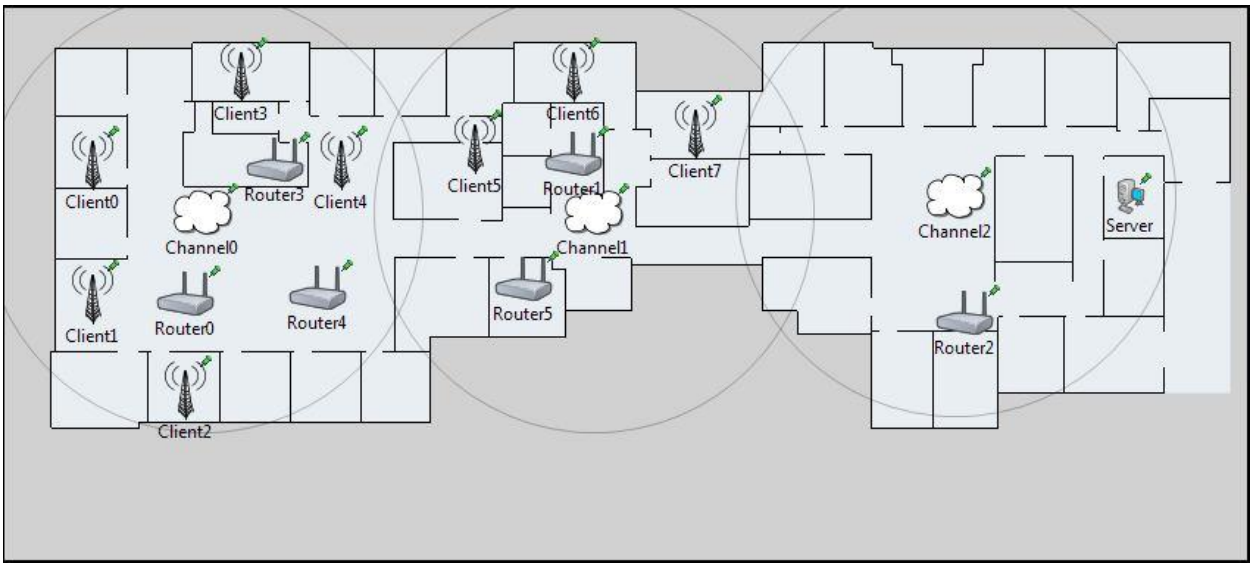

Figure 5.8: The Zigbee Network (Scenario 2)

Table 5.2 describes the architecture of the network in Figure 5.8.

Table 5.2: Network (Device and Clients under device)

Device	Clients
Router 0	Client0, Client1 and Client2
Router 3	Client3 and Client4
Router 4	Router0 and Router3
Router 1	Client5, Client6 and Client7
Router 5	Router1
Router2	Server

The wireless network simulated in scenario 2 is shown in Figure 5.8. The topology of the network is described in Table 5.2. The clients are distributed among Router0, Router3 and Router1. The Router 0, Router 3 and Router 1 are itself clients of Router 5 and Router 4. Router0, Router 3 and Router1 buffer the data from clients and transmit to the routers up in tree in a pre-defined interval of 0.2s. The Router4 and Router5 transmit the data packets to Router2 in a pre-defined interval of 1s.

Scenario 3: Packet loss is simulated between Zigbee devices in accordance to Zigbee performance test results in chapter 2.

Table 5.3: Throughput between devices

Link	Packet Loss (%)
Client to Router	0
Router to Router	50

The clients are placed in line of sight of routers. The packet loss up to 20 meters distance between nodes is 0%. The routers are assumed to be kept at a distance of 40 meters from each other. The packet loss reported at 40 meters is 50%.

5.2.3 Results

The results of test scenarios are presented in this section. The analyses of the results are described in Section 5.2.3.1.

Scenario 1:

Table 5.4: Results (Scenario 1)

Router to Router Transmission Time (seconds)	Queue Length(R0)		Queue Length(R1)		Queue Length(R2)		Frame Loss (Client)	Throughput (kb/s)
	Mean	Max	Mean	Max	Mean	Max		
0.2	13.83	63	20	112	18	80	2714	26.448
0.5	27.84	141	42.23	195	38	178	2537	27.329
1.5	71.95	252	104	347	106	468	2522	27.732

Scenario 2:

Table 5.5: Results (Scenario 2)

Queue Length (R0)		Queue Length (R1)		Queue Length (R2)		Queue Length (R3)		Queue Length (R4)		Queue Length (R5)	
Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
10	54	9	48	182	783	7.8	41	44	155	37.9	98

Scenario 3:

Table 5.6: Results (Scenario 3)

Queue Length (R0)		Queue Length (R1)		Queue Length (R2)		Queue Length (R3)		Queue Length (R4)		Queue Length (R5)	
Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
8.9	37	9.6	44	37.09	224	7.8	37	13.9	73	28.7	202

5.2.3.1 Analysis

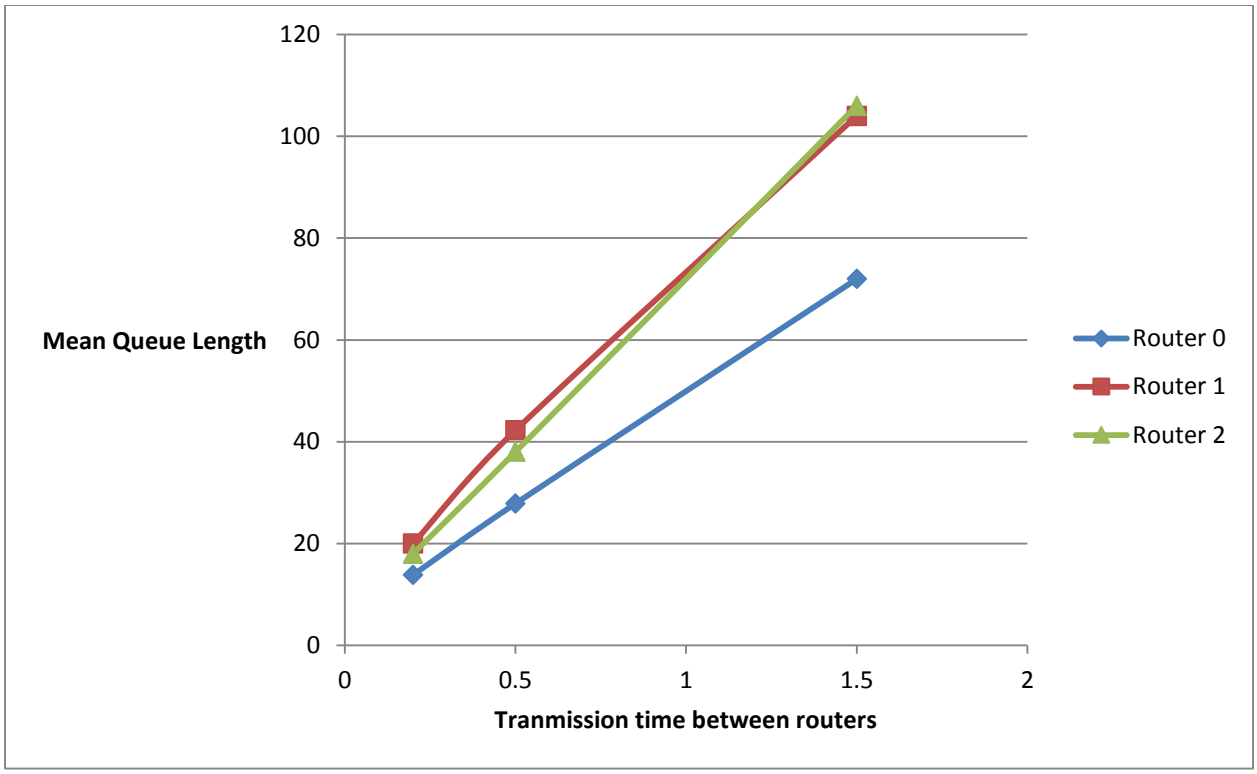

Figure 5.9: Queue Length (Scenario 1)

Figure 5.9 shows that queue length depends on router-router transmission time. As the router-router transmission time is increased the router demands larger queue which increases the cost of network deployment. However Table 5.4 shows that higher router-router transmission time can decrease the frame loss in the network. This can also help in supporting more clients as the routers have sufficient memory to store the results of each client before time elapses for the router to forward the data. It should be noted that frame loss in clients is caused by failure of transmission in CSMA-CA.

Table 5.5 shows the results of test scenario 2 in which the number of routers in the network is increased. The frame loss reported in scenario 2 is 3137 whereas 2817 is the reported frame loss in scenario 1. The frame loss is increased in this scenario compared to scenario 1

because increasing the number of routers increases the network activity. The throughput reported in scenario 2 is 23 kbps

In scenario 3, 50% packet loss in communication is induced in router to router transmission link. This decreases the throughput of overall network. This loss in throughput can only be improved by decreasing router-router distance. Frame loss in clients is low compared to scenario 2. This is because 50% of packets originated do not cross routers decreasing the network activity nearer to server. Throughput reported in scenario 3 is 4.579kbps.

Acknowledge packet send by server is 11 bytes long and send as a single frame. The loss of acknowledgement packet in all three scenarios is high and is in the range of 10000-15000 acknowledgement packets. This causes timeout in clients which triggers re-transmission increasing the network activity. Hence, it is efficient not to send an acknowledgement packet for every packet received in the server. Instead, a re-transmit request can be sent if the server did not receive a particular frame.

The packet loss of a Zigbee network can be controlled by optimizing the parameters – number of routers, buffer size of routers and time interval of router to router transmission. Thus, in the deployment of smart power meters, number of routers in the wireless network should be increased to reduce the data loss and to support more smart power meters. Zigbee routers are available with various buffer sizes and cost from $4 [5.10] to $69 [5.11]. The price of the Zigbee radio varies with buffer sizes and architecture of Zigbee routers. Xbee-PRO Wall Router [5.11] is a plug-and-play model and comes in a casing with pins to connect to wall outlets whereas Texas Instruments (TI) CC2520 [5.10] is a chip with microcontroller support for memory. The ideal solution to reduce the packet loss is increasing the buffer size and number of routers but the same comes with a higher hardware cost.

5.3 Future Work

The next step would be to perform the test scenarios in hardware test platform and verify the result from the simulation. The simulation can further be improved to determine latency of packets in the network.

The simulation could also be improved by using INET module of the OMNET++ software to model the 7 OSI layers in Zigbee protocol stack. The INET framework supports Zigbee protocol, IPv4, IPv6, TCP, SCTP, UDP and several application models. This new framework has models to implement signal fading caused by propagation loss of wireless signals. The modeling of Zigbee network by using INET framework will increase the accuracy of generated results. The new framework also has modules to support tree and mesh type networks in Zigbee.

A TDM (Time Division Multiplexing) approach for router to router communication can also be investigated. In this approach, a particular transmission time is reserved for each router in the network. The router will buffer the data from its clients and transmit the data to server or nearest router towards the server in its allocated time. The TDM approach can reduce the buffer memory in routers because all the devices communicate only in their allocated time slots thus reducing hardware costs in Zigbee routers. However, the TDM approach may increase the packet loss because results from features such as pattern matching and condition monitoring (section 1.4) implemented in smart power meter can cause un-predictable data flow in the network. This can be controlled by allocating buffer in the smart power meter to store these results before transmission.

5.4 References

[5.1] “Zigbee,” *Wikipedia, the free encyclopedia*. 22-Nov-2013.

[5.2] “Zigbee-based Home Area Networks Enable Smarter Energy Management” [Online]. Available: http://www.silabs.com/Support%20Documents/TechnicalDocs/Zigbee-based-HANs-for-Energy-Management.pdf. [Accessed: 25-Nov-2013].

[5.3] “IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs),” *IEEE Std 802.15.4-2003*, pp. 0_1–670, 2003.

[5.4] “OMNeT++ Manual” [Online]. Available: http://www.omnetpp.org/doc/omnetpp/manual/usman.html. [Accessed: 03-Dec-2013].

[5.5] J.-S. Lee and Y.-M. Wang, “Experimental Evaluation of Zigbee-Based Wireless Networks in Indoor Environments,” *Journal of Engineering*, vol. 2013, Feb. 2013.

[5.6] W. S. Jeong and S.-H. Cho, “Congestion Control for Efficient Transmission in Zigbee Networks,” in *5th International Conference on Wireless Communications, Networking and Mobile Computing, 2009. WiCom ’09*, 2009, pp. 1–4.

[5.7] G. Yang and Y. Yu, “Zigbee networks performance under WLAN 802.11b/g interference,” in *4th International Symposium on Wireless Pervasive Computing, 2009. ISWPC 2009*, 2009, pp. 1–4.

[5.8] “PRISM - Case Studies - IEEE 802.15.4 CSMA-CA Protocol (Zigbee).” [Online]. Available: http://www.prismmodelchecker.org/casestudies/Zigbee.php. [Accessed: 27-Nov-2013].

[5.9] C.-C. Song, C.-F. Feng, C.-H. Wang, and D.-C. Liaw, “Simulation and experimental analysis of a Zigbee sensor network with fault detection and reconfiguration mechanism,” in *Control Conference (ASCC), 2011 8th Asian*, 2011, pp. 659–664.

[5.10] “CC2520RHDT Texas Instruments | 296-22996-2-ND | DigiKey.” [Online]. Available: http://www.digikey.com/product-detail/en/CC2520RHDT/296-22996-2-ND/1770309. [Accessed: 26-Dec-2013].

[5.11] “XR-Z14-CW1P1 Digi International | 602-1149-ND | DigiKey.” [Online]. Available: http://www.digikey.com/product-detail/en/XR-Z14-CW1P1/602-1149-ND/2199607?WT.mc_id=IQ_7595_G_pla2199607&wt.srch=1&wt.medium=cpc. [Accessed: 26-Dec-2013].