

Spring 5-25-2019

Development of a Prototype System for Distributed Spectrum Sensing via Wireless Sensor Networks

Christopher R. Barber

**DEVELOPMENT OF A PROTOTYPE SYSTEM FOR
DISTRIBUTED SPECTRUM SENSING VIA
WIRELESS SENSOR NETWORKS**

by

Christopher Reineke Barber, B.S.

A Thesis Presented in Partial Fulfillment
of the Requirements of the Degree
Master of Science

COLLEGE OF ENGINEERING AND SCIENCE
LOUISIANA TECH UNIVERSITY

MAY 2019

LOUISIANA TECH UNIVERSITY

GRADUATE SCHOOL

March 15, 2015

Date of thesis defense

We hereby recommend that the thesis prepared by

Christopher Reineke Barber

entitled **Development of a Prototype System for Distributed Spectrum**

Sensing VIA Wireless Sensor Networks

be accepted in partial fulfillment of the requirements for the degree of

Master of Science in Engineering, Electrical Engineering Concentration

Dr. Miguel Gates, Supervisor of Thesis Research

Dr. Chester Wilson,
Head of Electrical Engineering

Members of the Thesis Committee:

Dr. Rastko Selmic
Dr. Jinko Kanno

Approved:

Hisham Hegab
Dean of Engineering & Science

Approved:

Ramu Ramachandran
Dean of the Graduate School

ABSTRACT

This thesis describes the research and development results of designing a system that can detect an unknown transmitter that is operating within a wideband spectrum covering 240-960MHz. The system design, development, and integration of a Prototype Spectrum Sensing Platform for the MEMSIC IRIS wireless sensor network platform [1] can be used for detection, monitoring, and localization of emitters in diverse RF propagation environments. Decoupling the dependency of the Received Signal Strength (RSS) measurements from the primary transceiver by the addition of a secondary receiver provides for continuous real-time analysis of the RF environment signal levels. The goal is to provide a simple, distributed 3-D spectrum analysis in real-time that is energy efficient, has a small form-factor and does not degrade the omni-directional operation of the primary node's antenna by utilizing mobile sensor nodes. Such distributed spectrum sensing will be able to provide accurate RSS information about non-coherent signals present in the sensing environment without the overhead of having to handle bidirectional communication to other nodes. The platform has been designed, fabricated, tested, and employed on static MEMSIC IRIS nodes for evaluation of RF sensing performance.

APPROVAL FOR SCHOLARLY DISSEMINATION

The author grants to the Prescott Memorial Library of Louisiana Tech University the right to reproduce, by appropriate methods, upon request, any or all portions of this Thesis. It is understood that “proper request” consists of the agreement, on the part of the requesting party, that said reproduction is for his personal use and that subsequent reproduction will not occur without written approval of the author of this Thesis. Further, any portions of the Thesis used in books, papers, and other works must be appropriately referenced to this Thesis.

Finally, the author of this Thesis reserves the right to publish freely, in the literature, at any time, any or all portions of this Thesis.

Author _____

Date _____

DEDICATION

I would like to dedicate this thesis to my grandfather and grandmother, Mr. Harry W. Reineke Jr. and Mrs. Eleanor McGarry Reineke. It was from the inspiration and stories about my grandfather's life as an engineer that helped foster my love for physics and engineering. Finally, if it was not for my grandmother's love and support, I do not think I would have had the amazing life and great wisdom that she imparted to me.

TABLE OF CONTENTS

ABSTRACT	iii
DEDICATION	v
LIST OF FIGURES	ix
LIST OF TABLES	xi
ACKNOWLEDGMENTS	xii
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.2 Research Objectives	2
1.3 Motivation	3
1.4 Organization	3
CHAPTER 2 BACKGROUND	5
2.1 Wireless Sensor Networks	5
2.1.1 Architecture of a Wireless Sensor Node	6
2.1.2 Microcontroller Unit (MCU)	7
2.1.3 Transceiver	8
2.1.4 Sensor Applications	8
2.1.5 Software	10
2.1.6 Wireless Networks	12
2.2 Localization Methods for Wireless Sensor Networks	13
2.2.1 Range-Based Localization Methods	14
2.2.1.1 Time Difference of Arrival (TDOA)	15

2.2.1.2	Angle of Arrival (AOA)	15
2.2.1.3	Received Signal Strength Indicator (RSSI)	15
CHAPTER 3 SYSTEM DESIGN		17
3.1	System Design Parameters.....	17
3.2	Circuit Level Design	18
3.2.1	Si1000 Wireless MCU	21
3.2.2	RF Frontend	22
3.2.2.1	PMA-545+ Low Noise Amplifier.....	24
3.2.2.2	PE42440 SP4T RF Switch.....	25
3.2.2.3	TC4-19+ RF Transformer.....	26
3.2.3	IRIS Node Interface	27
CHAPTER 4 SYSTEM INTEGRATION AND TESTING		28
4.1	PDSSP Code Development.....	28
4.1.1	PDSSP System Configuration and Operation.....	28
4.1.2	Spectrum Sensing Operation.....	29
4.2	IRIS Code Development.....	31
4.2.1	IRIS to Si1000 Serial Control Payload Development.....	32
4.2.2	IRIS to Si1000 Preliminary Integration Testing	36
CHAPTER 5 EXPERIMENTAL RESULTS		39
5.1	Operational System Testing and Diagnostics	39
5.1.1	PDSSP Preliminary Hardware Diagnostics	39
5.1.2	PDSSP Operational System Functionality Tests	39
5.2	PDSSP Deployment via Point-To-Point WSN	40
5.2.1	PDSSP Integration with MEMSIC IRIS Node	40
5.2.2	Point-To-Point Network Spectrum Sensing Experiment.....	43

5.2.3 Emitter Detection using Spectrum Sensing Experiment.....	46
CHAPTER 6 CONCLUSIONS AND FUTURE WORK.....	49
6.1 Conclusions.....	49
6.2 Future Work.....	51
BIBLIOGRAPHY.....	53

LIST OF FIGURES

Figure 2-1: WSN Applications	6
Figure 2-2: Architecture of a Wireless Sensor Node	7
Figure 2-3: Wireless Sensor Nodes	9
Figure 2-4: Basic Network Topologies.....	13
Figure 3-1: PDSSP Block Diagram	18
Figure 3-2: PDSSP PCB Layout	19
Figure 3-3: PCB Layer Thickness and Isolation Parameters	20
Figure 3-4: Si1000 Block Diagram.....	21
Figure 4-1: EZRadioPRO State Transitions During Frequency Scanning	30
Figure 4-2: TinyOS Serial Packet Format	31
Figure 4-3: MIB510 Programming Board Connected to IRIS Node.....	33
Figure 4-4: MIB510 Pass-Through to UART Output Pins.....	34
Figure 4-5: Si1000 Control Message Parameters	34
Figure 4-6: Si1000 Data Message Format	36
Figure 4-7: MIB510 UART Connection to Si1000 Development Board.....	37
Figure 4-8: Si1000 UART and Power Connections to IRIS Node	38
Figure 5-1: IRIS Node Integration with PDSSP for Initial Testing.....	41
Figure 5-2: IRIS Node Base Station with RS-232 to USB Connection to PC.....	43
Figure 5-3: Spectrum Data Collected from Scanning 500-933MHz	45
Figure 5-4: 400-933MHz Spectrum Data from the PDSSP.....	45

Figure 5-5: Emitter Not Present in the 900-933MHz Spectrum.....	46
Figure 5-6: Emitter Present at 920MHz.....	47

LIST OF TABLES

Table 2-1: Sensors Based on Application	10
Table 2-2: ITU-R Spectrum Allocations for the ISM Bands	13
Table 3-1: Parameters Used to Calculate the Characteristic Impedance of the RF Traces	23
Table 3-2: PMA-545+ Device Specifications	24
Table 3-3: PE42440 Port Selection Truth Table	25
Table 3-4: PE42440 Device Specifications	26
Table 3-5: Si1000 Differential Input Impedance Range	27
Table 4-1: EZRadioPRO IDLE State Operational Modes	29
Table 5-1: PDSSP Component Energy Consumption.....	42
Table 5-2: PDSSP Operational Energy	43

ACKNOWLEDGMENTS

I would like to give my sincere thanks and appreciation to my advisor, Dr. Rastko Selmic, for all his support and guidance. Next, I would like to thank my wife, Mary Barber, and my family for all their love, support and inspiration during the many hours I have spent working on the research and documentation for this thesis. Additionally, I would like to recognize Jason Porter for his help with the board layout improvements he assisted with on my design. Finally, I would like to thank Dr. Nathan Wallace and Dr. Miguel Gates, both of whom have been great friends and colleagues to me during my years at Louisiana Tech, for reviewing my thesis and providing support during the creation of my thesis.

CHAPTER 1

INTRODUCTION

This chapter presents an overview of the scope of this thesis and its focus on developing the Prototype Distributed Spectrum Sensing Platform (PDSSP). The PDSSP was developed to act as an integrated sensor for providing wideband spectrum sensing via a deployed wireless sensor network (WSN). Moreover, the PDSSP design provides a hardware-based solution, which will advance hidden emitter detection accuracy using the Position Adaptive Direction Finding (PADF) method [2].

1.1 Background

Wireless Sensor Networks (WSNs) have become more prominent as the advancement of technology has given rise to the development of small scale, low power wireless systems that can be used to provide high-resolution spatial and temporal environmental measurements for various applications. A WSN is an infrastructure of wirelessly enabled embedded devices sometimes referred to by researchers as “motes” [3]. Moreover, wireless sensor nodes communicate using routable network topologies to provide an end user the aptness to observe and respond to developments within a certain environmental domain. The applications in which WSNs can be utilized to provide enhanced environmental feedback to systems are ever expanding.

However, as a basic example, we describe the application of a WSN in battlefield and urban warfare situational awareness. Many hazards are present in today's clandestine and often urban battlefield scenarios that involve the wireless control of explosive devices. These devices can be controlled either by current modern technologies that employ the use of the cellular communications infrastructure, or very crude wireless communication technologies. Therefore, the deployment of a distributed network of sensor nodes outfitted with specialized sensors that can provide spectrum analysis and source localization is critical to the warfighter.

1.2 Research Objectives

The research objectives of this thesis are to develop the Prototype Distributed Spectrum Sensing Platform (PDSSP) into a fully functional device that could be paired with a MEMSIC IRIS [1] wireless sensor node. Thereby, the PDSSP will operate as a sensor network that could be used to collect and measure RSSI over a wideband of frequencies. The process to create this system involved the selection of the components, all of which had to be able to operate via a DC supply of 3.3V, with the primary component being a System on Chip (SoC). The SoC was a basic MCU integrated with a wideband transceiver. This SoC would need to be able to provide UART based communication with the MEMSIC IRIS node for control and transfer of sensed data. Finally, this SoC would need to be controllable via the MEMSIC IRIS sensor node, thus creating a controllable sensor platform that can be integrated into a deployable WSN and utilized for spectrum sensing applications. Similar research to this design was used for expanding the capabilities of the Versatile Sensor Node (VSN) to perform spectrum

sensing of the 868MHz Industrial, Scientific and Medical (ISM) band [4]. However, this design was not integrated and deployed as a sensor device using a pre-existing WSN.

1.3 Motivation

The focus of this thesis has applications within the area of range-based localization using WSNs. Specifically, this thesis investigates applications where the use of a hardware-based platform can be used for spectrum sensing to help aid the use of the PADF method for localizing a hidden emitter. The PADF localization method was first presented in [5] and uses the application of RSSI-based localization within a framework of dynamic receivers. However, this research advances the PADF method to be able to utilize information collected from the dynamic spectrum sensing of the RF signals present in the environment. In contrast, the previous implementation of the PADF method was restricted to the use of singular frequencies for the receivers and the hidden emitter [2].

A majority of research involving spectrum sensing applications using WSNs is focused on the use of WSNs as a secondary network. In this scenario the WSN is used specifically for spectrum hole detection. This information is then transferred to the primary cognitive radio network to determine which parts of the spectrum are unoccupied. Finally, this information is used to move the members of the primary network to the uncongested spectrum regions to improve spectrum usage [6] [7].

1.4 Organization

Chapter 2 presents the background information and research in relation to WSNs, localization methods and the research conducted in this thesis. Chapter 3 covers the

details of all the processes involved in designing the PDSSP. This provides the details of the circuit development, fabrication, and software development that were necessary to take the PDSSP from a concept to a functional prototype. Chapter 4 presents the process of software development and hardware integration between the PDSSP and the MEMSIC IRIS node for operational testing. In Chapter 5, all operational metrics of the PDSSP's performance while performing spectrum sensing have been evaluated. Tests have been performed with the PDSSP isolated to determine baseline measurements. Next, the PDSSP is connected to MEMSIC IRIS node. This test is used to verify network-based operation and control using a Point-To-Point (PTP) network. Any discrepancies of the system performance or future enhancements are included in Chapter 6.

CHAPTER 2

BACKGROUND

2.1 Wireless Sensor Networks

The author of [3] presents the formal definition of the components that comprise a sensor network. Therefore, one can visualize a sensor network as an infrastructure comprised of sensing (measuring), computing, and communication elements that gives an operator the ability to instrument, observe, and react to events and phenomena in a specified environment. Furthermore, [3] describes the four major components of a sensor network, which can be described as a collection of distributed sensors connected through a network of a specific topology to a centralized point where data aggregation and analysis is performed.

Wireless Sensor Networks (WSNs) have become a common fixture in the advancement of distributed sensing as well as acting as actuators that provide command and control of a diverse variety of automated systems. WSNs can provide a method of distributed data collection of environmental data for smart sensing applications. Smart sensing applications can require the need for specific or high-resolution data. Deployment coverage for WSNs can vary between small areas of coverage (i.e. monitoring a building) to large areas of coverage (i.e. monitoring a battlefield) as shown

in Figure 2-1. In summary, sensing applications for WSNs include industrial, home, municipal, atmospheric, space, and maritime and military applications in nature.



Figure 2-1: WSN Applications

2.1.1 Architecture of a Wireless Sensor Node

Wireless sensor nodes are designed based on the following criteria, which includes such fundamental components as: 1) Microprocessor for system control and signal processing; 2) Wireless transceiver for network communication and data transfer; 3) Power source to provide energy for proper system operation; 4) Singular or combination of sensors, actuators or both types of interfaces. However, the primary factor for the design of a wireless sensor node is the total combined energy consumption of the system. The basic architecture of a wireless sensor node can be seen in Figure 2-2.

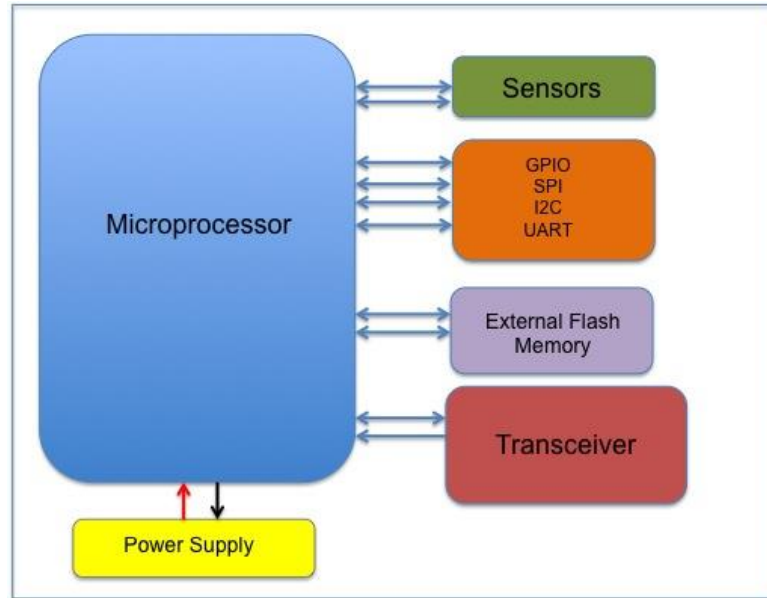


Figure 2-2: Architecture of a Wireless Sensor Node

2.1.2 Microcontroller Unit (MCU)

MCUs act as the primary control and signal-processing unit for a sensor node. They provide the primary interface to the system's sensors, actuators and wireless transceiver. The majority workload of the collection, processing and storage of environmental data is tasked to the system's MCU. The MCU controls the system's wireless transceiver usually by way of communication through a Serial Peripheral Interface (SPI) connection.

Additionally, most node platforms have onboard external Flash RAM, which provides additional memory for logging applications and is accessible via the MCU. The node's MCU also provides analog and digital GPIOs, I2C and UART pins that can be used for interfacing with sensors or provide hardware-based communication for custom applications. Two primary MCUs used in popular wireless sensor node platforms are the Texas Instruments MSP430 and the Atmel ATmega1281 [1].

2.1.3 Transceiver

IEEE 802.15.4 is a wireless communication standard that was developed by the IEEE and the ZigBee Alliance [8]. This standard provides a communication protocol that is ideally suited for WSNs, where large networks of nodes are used for control and monitoring applications. ZigBee can be implemented with a low system cost per node while providing efficient and secure wireless communication between members of a WSN. ZigBee transceivers are optimized for low power and data rate applications where the primary system power is provided by the batteries.

The two main layers of the IEEE 802.15.2 standard used in ZigBee are the Physical Layer (PHY) and the Medium Access Control (MAC). The PHY layer consists of the RF-IC, its frequency range and modulation scheme. The MAC layer provides a software-based protocol stack that is very diverse in functionality. This layer provides access control and dependable data exchange for all members of the network.

The MAC layer utilizes Carrier Sense Multiple Access with Collision Detection (CSMA/CD) to mitigate collisions between wireless packets traversing the network. Additionally, the MAC layer supports the star and mesh-based network topologies [9] [1]. The two layers that are implemented above the MAC layer were developed by the Zigbee Alliance to provide network functionality and security mechanisms. In conclusion, we can see how the versatile, robust and low power design of the IEEE 802.15.4 ZigBee protocol is ideally suited for WSNs.

2.1.4 Sensor Applications

The ability of wireless sensor nodes to be outfitted with sensors provides a distributed feedback system that is capable of providing high-resolution information about

one or several environmental variables. Primary platforms used for wireless sensor networks include the following shown in Figure 2-3. Sensors integrated into the design or attached to any of the digital interface ports can range in size, power consumption, sampling rate and required memory allocation needed to store collected data.



Figure 2-3: Wireless Sensor Nodes

Depending on the available power source for the node, the operational lifetime expectancy could be greatly reduced if careful consideration of the above parameters is not considered. Results in [10], [3] and [6] provide an extensive amount of background and examples of typical sensors and applications involving WSNs. From this literature a synopsis of the applications and associated sensors is provided in Table 2-1.

Table 2-1: Sensors Based on Application

Application	Environment	Sensors
Medical	Hospital/Patient	Blood Pressure EKG Body Temperature
Military	Battlefield	Motion Radiation Spectrum
Meteorology	Ground-level Atmospheric Layers	Wind Speed Temperature Pressure
Industrial	Plant/Pipeline	Pressure Stress/Strain Chemical
Residential	Personal living space	Carbon Monoxide Water

2.1.5 Software

TinyOS, a component based embedded operating system for WSNs, was developed and is maintained by the TinyOS Alliance [11]. TinyOS is written in the NesC programming language, which is a variant of the C programming language. NesC was developed to optimize code size to conserve system RAM, therefore providing powerful, portable and optimized code for WSNs. Software components make up TinyOS to provide hardware abstractions to low level MCU peripherals. The size of TinyOS by itself is only 400 bytes, which makes it ideal for low power MCU applications where System-On-a-Chip (SOC) devices are designed with limited RAM for power conservation.

The nature of TinyOS's event-driven concurrency, which includes the use of pairing of software and hardware split-phase operations, is optimized for memory conservation. Split-Phase operations are those in which a process is executed by a request and then executes its function immediately. Once the execution is completed, this process will signal its completion to the operating system. This is the solution to the problem of the RAM intensive nature of using threaded processes on embedded platforms. Split-phase operations remove the problem that process threading produces, which is the need for a private stack for each thread that is initialized. Therefore, a TinyOS program is an interconnection of individual components in which each provides specific interfaces for system operation and control. Components can be viewed as a set of three types of abstractions: commands, events, and tasks. The execution of a TinyOS application starts when a command is issued. Then this command becomes a request that is provided to the component to initiate a service. Once the command completes its computation, it will provide an event, which notifies the system that the service has completed the command that was issued. The author of [12] explains this operation: "Commands and events cannot block: rather, a request for service is split phase in that the request for service (the command) and the completion signal (the corresponding event) are decoupled. The command returns immediately and the event signals completion at a later time." [13] [12] also state that "The current version of TinyOS provides a large number of components to applications developers, including abstractions for sensors, single-hop networking, ad-hoc routing, power management, times, and non-volatile storage." The versatility and flexibility of TinyOS stems from the programming scheme provided by the NesC

language, which in turn provides a relationship focused toward RAM preservation for low power embedded systems.

2.1.6 Wireless Networks

The direct implementation of WSNs started with the development of the ALOHA networks [10], which were based on the premise that a node can transmit data at any time. If another member of the network receives the data transmitted by the node, then an acknowledgement is sent from the node that received the data. However, if the transmitting node does not receive any acknowledgement, then it will wait for a timeframe of random length and retransmit the data again.

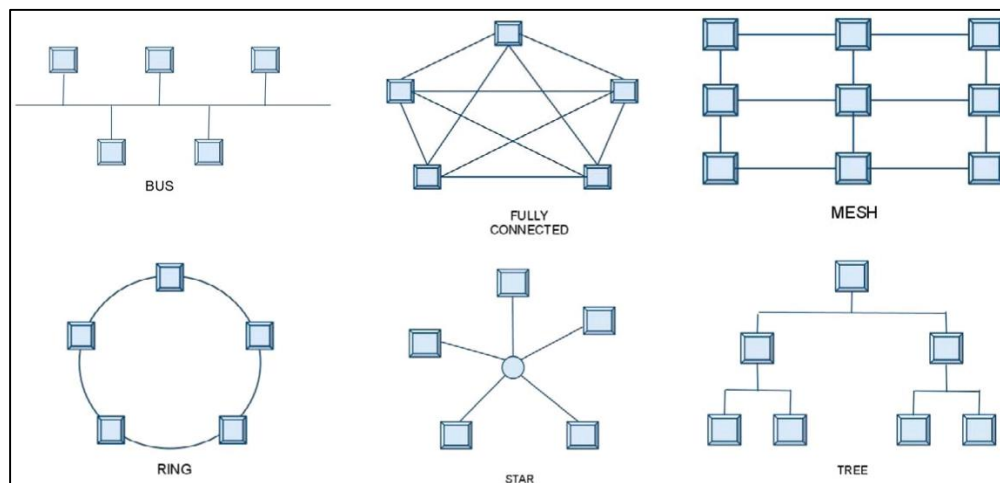
With the advancement of wireless communication technologies, the problem of loss of information due to simultaneous transmission between members of a network resulted in what is commonly known as “packet collisions”. This in turn gave rise to the development and implementation of transmission schemes, which helped to reduce the likelihood of packet collisions between members of a network. Solutions to this issue gave rise to the development of Frequency Division Multiple Access (FDMA), Code Division Multiple Access (CDMA) and Time Division Multiple Access (TDMA).

WSNs can be deployed using various wireless communications, most of which are chosen based on the data size and power requirements of the network. Most commercially available products utilize the Industrial, Scientific and Medical (ISM) frequencies that are defined by the ITU-R in 5.138 [13]. In Table 2-2 the spectrum allocations are shown.

Table 2-2: ITU-R Spectrum Allocations for the ISM Bands

Frequency Range (MHz)	Bandwidth (MHz)	Center Frequency (MHz)	Channels
433.050-434.790	1.74	433.920	6.5
902.000-928.000	26	915	10
2400-2500	100	2450	5.7

Depending on the application, WSNs can be configured in several different network topologies each of which will be directly dependent on the application. Shown in Figure 2-4 are the basic network topologies used for WSNs.

**Figure 2-4: Basic Network Topologies**

2.2 Localization Methods for Wireless Sensor Networks

Localization is one of the important tasks that WSNs perform. This involves the ability of members of the network to systematically discover and localize other members of a deployed network. Localization by itself is still an intense area of research in WSNs. In the following sections, the primary methods of localization that are utilized in WSN

applications will be discussed. Localization techniques can be categorized into two primary categories: 1.) Range-Based Localization; and 2.) Range-Free Localization. Range-Based techniques include: Time Difference of Arrival (TDOA) [14], [15] Angle of Arrival (AOA) [14], [16] and Received Signal Strength Indication (RSSI) [14]. Range-Free techniques include: Centroid [16], APIT [17], and SeRLoc [18]. However, the techniques described above are just some of many other standard and hybrid techniques used for localization applications within wireless communication system infrastructures. The focus of this research directly relates to the area of Range-Based localization techniques, specifically the use of RSSI based localization in the context of the Position Adaptive Direction Finding (PADF) algorithm.

2.2.1 Range-Based Localization Methods

Range-Based localization techniques are dependent on the information that can be extracted from the intensity of the power being received between nodes. This signal intensity information is used to provide an estimate of the distance between nodes in the deployed network. In [19] the author categorizes nodes into two subtypes: *Anchors*, which are nodes in which their physical location is known or determined by GPS, and *Unlocalized* nodes that are members of the network, but no localization information is available. In [20] the authors discuss how Range-Based localization techniques can be seen to operate in two phases, “Ranging and position computation”. Methods that are considered to be included in the ranging process are: TDOA, AOA and RSSI. Where algorithms such as Least Squares (LS), Weighted Least Squares (WLS) and Maximum Likelihood (ML) are used for position computation and estimation.

2.2.1.1 Time Difference of Arrival (TDOA)

Time Difference of Arrival (TDOA) is an active and highly developed method of localization used in WSN. This distance-related method uses signal measurements taken at N known receivers to estimate the position of an unlocalized transmitter. Once the transmitter's signal is detected by the receivers, this information is then used to compute the integration of the lag-product of the signals or cross-correlation. However, this method is only accurate at estimating the position when the time interval over which the integration performed is of considerable length. This method also requires that the time bases of all the receivers be synchronized.

2.2.1.2 Angle of Arrival (AOA)

Angle of Arrival (AOA) based localization uses the known position and orientation of one or more receivers in a linear configuration to estimate the direction from which a signal was transmitted. In [14] the author provides additional detail showing that this technique can be broken down in to two subclasses: 1.) Measurements based on the amplitude response of the receiver's antenna; and 2.) Measurements based on the phase response of the receiver's antenna. Factors that are critical to AOA based estimation include the use of anisotropic antennas, accurate signal measurement and that the receivers should have a sufficient spatial separation between them.

2.2.1.3 Received Signal Strength Indicator (RSSI)

RSSI is one of the simplest localization methods that can be employed for estimating the position of a transmitter. This method is directly dependent on the measurement of the non-coherent characteristics of a signal. RSSI is a standard feature found in a majority of wireless communication devices. Measuring RSSI requires no

additional hardware beyond the basic transceiver. It uses the basic understanding that during the time of flight of a signal, its power level will be reduced as the inverse square of the distance between the source of the signal and the receiver that detects this signal.

Therefore, RSSI based localization involves the initial collection of detected signal power levels at multiple receivers, then the use of a propagation model to estimate channel propagation conditions to account for channel losses, which is then applied to the signal measurements to determine an estimate of the transmitter's location. The propagation model accounts for the additional factors that will reduce the signal strength due to large and small-scale fading, reflections and refractions of the signal as it travels in free space.

CHAPTER 3

SYSTEM DESIGN

3.1 System Design Parameters

The design parameters for the Prototype Distributed Spectrum Sensing Platform (PDSSP) were developed to advance the hardware aspects involved with the experiments that included RSSI based localization using PADF. This design decoupled the dependency of the received signal strength (RSS) measurements from the primary transceiver by the addition of a secondary receiver. Therefore, it provided a distributed and continuous real-time analysis of signals in the RF environment. The challenge of this design was to provide a distributed, 3-D spectrum analysis in real time that is energy efficient, has small a form-factor and does not degrade the operation of the nodes communication system.

The block diagram shown in Figure 3-1 depicts the system design concept of the PDSSP. The connections depicted show the RF and digital signal paths between components that are connected to the Si1000. Once a signal is received at the antenna port it is then processed by a PMA-545+ Low Noise Amplifier (LNA), which is used to increase the signal to noise ratio of the RF signal.

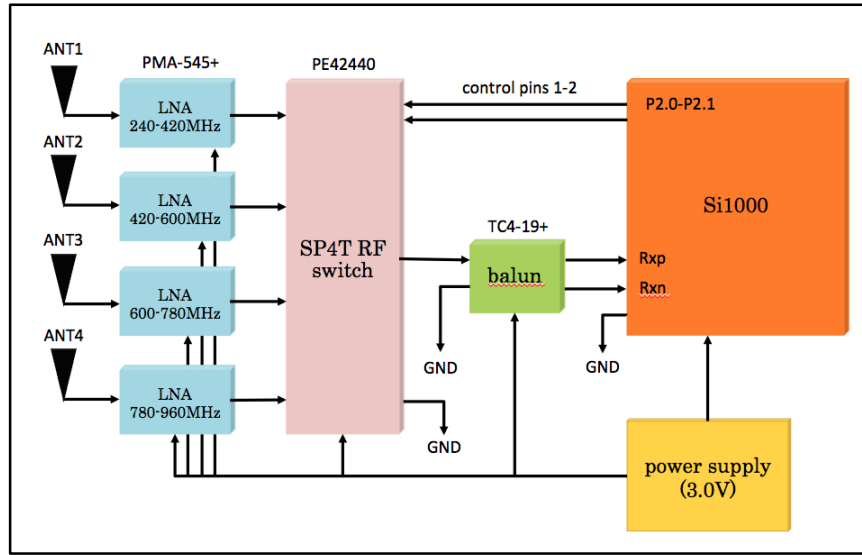


Figure 3-1: PDSSP Block Diagram

This serves two purposes: (i) improves weak signal reception; (ii) overcomes any losses in the RF path between the antenna and the Si1000. Four antennas were used to subdivide the 720MHz wide operational range of the EZRadioPRO into four 180MHz wide sub-bands. The subdivision of the operational frequencies was used to eliminate the need for complex matching networks. The TC4-19+ wideband balun was used to match the single-ended output from the PE42440 SP4T RF switch to the differential input of the Si1000. Therefore, this system concept provided the component level framework from which the PDSSP was designed using the CadSoft EAGLE software.

3.2 Circuit Level Design

The circuit level design of the PDSSP was done in CadSoft's Eagle PCB Design software. This software was used to create the schematic of all the components needed to create the PDSSP. Once the schematic was created, the PCB layout of the board was developed and the components were positioned on the board. The board can be viewed as two sections: the RF frontend of the system located at the top of the board and the

MCU section of the board located at the lower half. This separation of the two sections was critical to reduce inductive interference from the traces carrying RF signals.

Figure 3-2 shows the PCB layout of the hardware components on the circuit board. The RF frontend and antennas are distanced from the MCU section of the board. This separation provides the necessary isolation of the RF section from the MCU section.

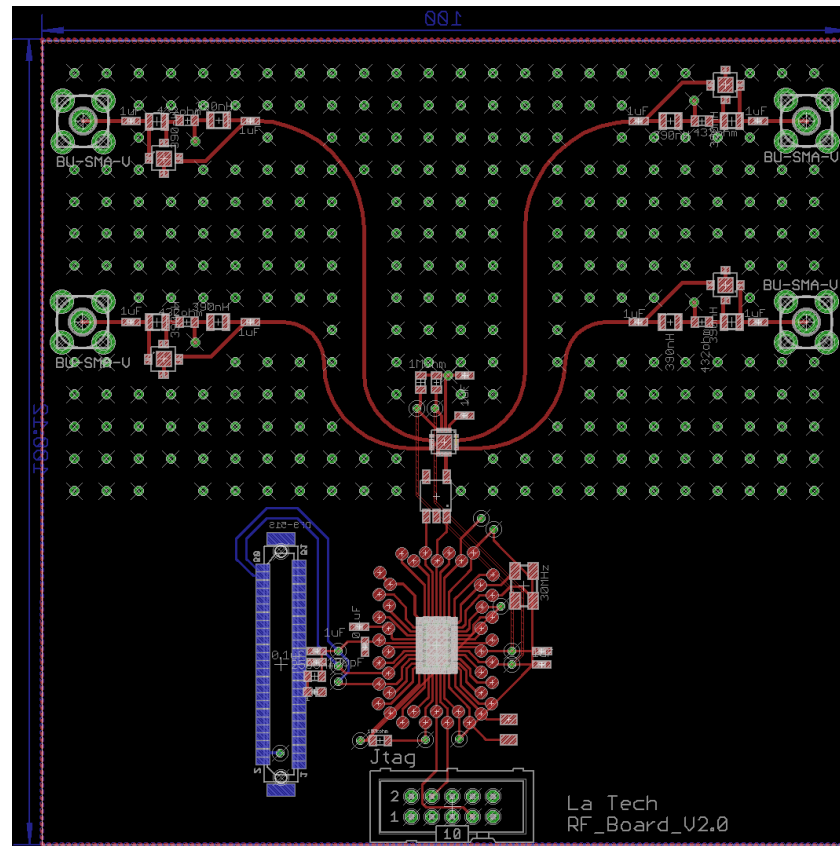
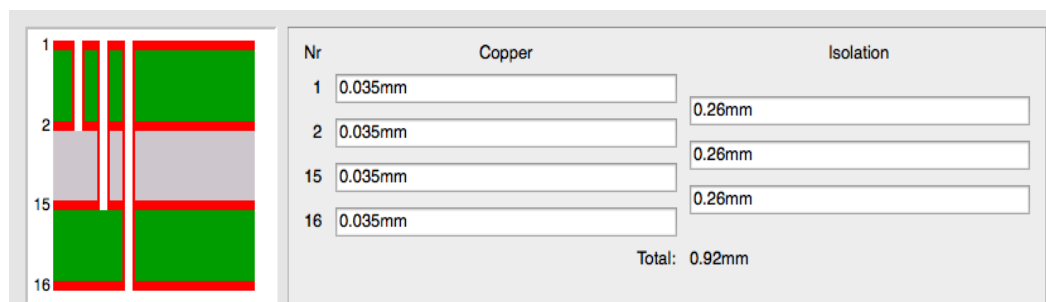


Figure 3-2: PDSSP PCB Layout

The parameters shown in Figure 3-3 were used to determine the layer and isolation thickness between layers for each of the four layers of the board.



Nr	Copper	Isolation
1	0.035mm	0.26mm
2	0.035mm	0.26mm
15	0.035mm	0.26mm
16	0.035mm	
Total:		0.92mm

Figure 3-3: PCB Layer Thickness and Isolation Parameters

The PDSSP is composed of four conductive layers. The first layer is where all the electronic components are located. This layer also contains all of the RF signal traces, which are isolated to the top half of the board. The second layer acts as a ground plane that provides isolation between the first layer of the board and the third layer, which is the power plane that provides power to all components on the board. The third layer also has two isolated traces that are connected between two of the Si1000's digital pins and the PE42440's digital CMOS inputs. These traces are used for digital logic control of the PE42440 to switch between each of the four antennas. The third layer contains the power plane for providing power to all components on the PDSSP. This layer is isolated from the first layer by the second layer ground plane. This is done to isolate the RF signals from the DC voltages present in the third layer. During the design process, several factors had to be taken into consideration to account for the resonances that occur with multi-layer PCB design that transfer the RF signals. With the use of ground planes to isolate the RF signal layer from the power layer, this created resonances between layers that had to be eliminated using vias. These vias short the two ground layers and are used to increase the TEM (Transverse Electric Magnetic) resonance to frequencies higher than the 10th-12th harmonics of the radio signals being captured.

3.2.1 Si1000 Wireless MCU

The Silicon Labs Si1000 [21], shown in Figure 3-4, was chosen for this design for its wide band coverage to allow for a broad range of operation for dynamic spectrum sensing for various applications such as geo-location, RF propagation environment monitoring and other situations where distributed spectrum awareness is required.

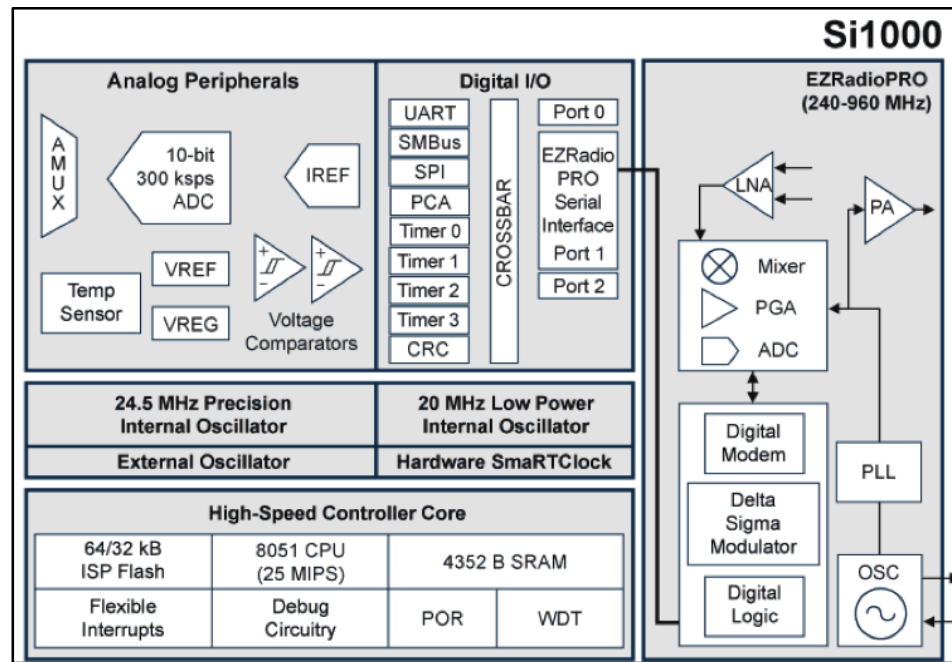


Figure 3-4: Si1000 Block Diagram

The Si1000 wireless MCU platform [21] combines a high-speed 8051 MCU core with an EZRadioPRO transceiver. This System-On-a-Chip (SoC) device provides a versatile wireless platform for applications that require wideband ISM operation. The Si1000 8051 core operates with a pipelined instruction architecture, enabling 70% of instruction execution within one to two system clock cycles. Therefore, it provides up to 25 MIPS operation with a 25MHz clock. Total memory capacity is 4352 bytes from

internal RAM (256 + 4096) with an additional 64 kB of 1024 byte programmable sectors. Additional device communication capabilities are provided with the usage of UART, SPI, and I2C.

The Si1000 also includes an EZRadioPRO transceiver, which is paired with the 8051 MCU core within the Si1000's single chip package. This transceiver is controlled via an internal SPI connection to the 8051 MCU. This transceiver has operational capabilities to receive and transmit data over the frequency range from 240-960MHz, thereby providing coverage over a large portion of the Industrial, Scientific and Medical (ISM) [13] allocated spectrum. The receive sensitivity of this transceiver is -121 dBm providing improved reception of weak signals. More importantly, during receive operation only 18.5 mA of current is required. Moreover, the implementations of the Si1000 for this design only utilizes the EZRadioPRO's receiving capabilities.

3.2.2 RF Frontend

The PDSSP has a specially designed RF frontend section. The RF frontend section of the board has four separate independent RF pathways. Each pathway consists of a PMA-545+ Low Noise Amplifier (LNA) with all of the pathways converging to the PE42440 RF switch. From the PE42440 the selected port transfers the RF signal to the TC4-19+ balun, which transforms the single-ended connection into a differential connection that connects to the Si1000.

Each antenna connection is used to sub-divide the full range of receive frequencies so the Si1000 can operate into four separate frequency bands. Each of the four sub-bands has a bandwidth of 180MHz. This allows the use of specific antennas that are resonant for each of the four sub-bands. The selection of which specific antenna will

be active during spectrum sensing is controlled by the variation of logic states from two pins that are connected between the Si1000 and the PE42440 RF switch.

The RF traces from the Si1000, through the RF switch, low-noise amplifier and other supporting circuitry to the antennas are all impedance-matched to reduce loss. These components are connected to the RF switch through the thick, curved traces. The traces need to be curved; otherwise, the traces would cause signal reflections, which could induce an impedance mismatch. The impedance for each of the four RF traces was calculated using **Eq. 0-1** using the parameters in **Table 3-1**.

$$Z_0 = \frac{87}{\sqrt{\epsilon_r + 1.41}} \ln \left(\frac{5.98H}{0.8W + T} \right) \text{ ohms} \quad \text{Eq. 0-1}$$

The traces each have specific width, height, and thickness that creates the correct characteristic impedance for matching the 50 ohm input and output pin impedances used on the PMA-545+ and PE42440.

Table 3-1: Parameters Used to Calculate the Characteristic Impedance of the RF Traces

Parameter	Value
Height (H)	0.26 mm
Width (W)	0.45 mm
Thickness (T)	0.035 mm
Relative Permittivity (ϵ_r)	4.8 (FR4 dialectic material)
Characteristic Impedance	47.8 ohms

Therefore, each RF trace has a characteristic impedance of 47.8 ohms, which creates an acceptable match for the RF trace to be connected to the 50-ohm impedance of the PMA-545+ and PE42440.

3.2.2.1 *PMA-545+ Low Noise Amplifier*

The PMA-545+ low noise monolithic amplifier from Mini Circuits [22] provides the initial RF signal amplification which increases the Signal-To-Noise Ratio (SNR) of any signal that is being transferred from the antenna. This is critical when signals of interest are weak or have been attenuated by channel losses. Additionally, the amplification of the signal also helps to compensate for any additional losses between the PMA-545+ and the PE42440 RF Switch. Operational specifications for the PMA-545+ are shown in Table 3-2.

Table 3-2: PMA-545+ Device Specifications

Parameter	Typical	Max	Units
Operational Frequency Range	0.05	6	GHz
Power Supply	3.0	5.0	VDC
Current Draw	80	160	mA
Receive Sensitivity	5.6	10	mA
Gain	0.05@26.1 0.5@23.3 1.0@19.4	15.6	GHz@dBm
Power Dissipation	0.24	0.8	mW

3.2.2.2 PE42440 SP4T RF Switch

The PE42440 SP4T RF switch [23] provides a mechanism for digitally controlling the selection of which antenna will be active during spectrum sensing. Antenna selection is controlled by the on-board CMOS control interface on the PE42440. The Si1000 is connected to the CMOS input pins on the PE42440 via two of its digital I/O pins. The selection of which antenna port is used can be performed by varying the logic states of two digital I/O pins on the Si1000 that are connected to the PE42440. The maximum rate at which the switching between antennas can occur is 25kHz. Table 3-3 shows the logic states, which are used to control the PE42440.

Table 3-3: PE42440 Port Selection Truth Table

Path	V2	V1
Antenna 1	0	0
Antenna 2	1	0
Antenna 3	0	1
Antenna 4	1	1

During spectrum sensing the Si1000 will switch the logic states in accordance with the values for V2 and V1 to change the signal path to the correct antenna for the frequencies that will be scanned. The device parameters of the PE42440 critical to the PDSSP design are shown in Table 3-4.

Table 3-4: PE42440 Device Specifications

Parameter	Min	Typ	Max	Units
Operational Frequency Range	50		3000	MHz
Supply Voltage	2.65	2.75	3.3	VDC
Current Draw		13	50	μ A
Insertion Loss (50-1000MHz)		0.45	0.65	dB
RF Input Power (50-500MHz) (500-3000MHz)			+28 +33	dBm
Digital Logic Control Voltage	0.4		1.4	VDC
Power Dissipation		0.35	0.39	mW

3.2.2.3 TC4-19+ RF Transformer

The TC4-19+ RF transformer [24] creates a 50 Ω match between the PE42440 and the Si1000's differential inputs used for signal reception. Impedance matching between the two unmatched sections of the RF pathway is critical to the proper transfer of RF energy from the antennas to the Si1000. This component was necessary due to the differential input on the Si1000.

The TC4-19+ provides the necessary impedance match between the 50-ohm impedance of the PE42440 RF switch and the differential input of the Si1000 RX-P and RX-N pins. Table 3-5 shows the variance of impedance over the operational range of the Si1000's inputs over specific frequencies within the coverage range.

Table 3-5: Si1000 Differential Input Impedance Range

Center Frequency (MHz)	Impedance (Ω)
315	107-137j
433	89-110j
868	54-63j
915	51-60j

3.2.3 IRIS Node Interface

The 51-pin connector provides a direct interface to the PDSSP from which power, control, and data are transferred between the IRIS node and the PDSSP. Voltage from the IRIS node's 3.3VDC batteries is transferred to the PDSSP and then distributed across the board to provide power to the Si1000, PE42440, and all four of the PMA-545+ preamps. Control parameters used for configuring the Si1000 for spectrum sensing operation and collected signal data are transmitted through the connected RX and TX UART pins on the 51-pin connector.

CHAPTER 4

SYSTEM INTEGRATION AND TESTING

4.1 PDSSP Code Development

Code development for the PDSSP involved developing software that enabled the Si1000 to be controlled by commands sent over a UART connection, dynamically scan through various frequencies at predetermined steps, collect RSSI data for each frequency while scanning, then transmit this collected RSSI data via a UART connection to the IRIS wireless sensor node. Therefore, with this level of functionality the PDSSP can operate as a low-power compact versatile platform that can be coupled with wireless sensor nodes for distributed dynamic spectrum sensing and monitoring.

4.1.1 PDSSP System Configuration and Operation

During initial system power up of the PDSSP, the Si1000's 8051 MCU configures the EZRadioPRO using SPI commands to perform operational state transitions. These commands are used to configure a set of 8-bit registers that are used to operate and control the behavior of the EZRadioPRO.

The *EZRADIO_OPERATING_AND_FUNCTION_CONTROL_1* register is used to change between IDLE state operational modes of the EZRadioPRO. Table 4-1 shows the IDLE state operational modes and the current consumption each mode uses. During the initial system power up, the EZRadioPRO is set to the IDLE-STANDBY operational

mode. This mode has the lowest current consumption and utilizes the low power digital regulated supply (LPLDO) to maintain the register values while in this mode.

Table 4-1: EZRadioPRO IDLE State Operational Modes

Operational Mode	Current Consumption
SHUTDOWN	15 nA
STANDBY	450 nA
SLEEP	1 μ A
READY	600 μ A
TUNING	8.5 mA
RECEIVE	18.5 mA

4.1.2 Spectrum Sensing Operation

The primary function of the PDSSP is to perform the dynamic spectrum sensing by utilizing the Si1000 to scan user specific frequency ranges for a specific step size. RSSI data is then collected and averaged at each frequency determined by the step size over the scan range. Only when the Si1000 receives a serial control packet from the IRIS node to initiate spectrum sensing does the Si1000 execute this process. The execution of the *Scan_Frequencies()* function is the primary set of directives being performed until the operation is terminated. Once the spectrum sensing operation is complete, the Si1000 configures the EZRadioPRO back into STANDBY mode.

Frequency scanning can be done either as a “single-shot” or a “continuous” operation. Single-Shot operation only performs one iteration of the *Scan_Frequencies()* function for the specified range of frequencies. Continuous operation will execute the

Scan_Frequencies() function for n iterations, where n is the iteration index specified by the user. Operational state changes are shown in Figure 4-1 that occur during the execution of the *Scan_Frequencies()* function. During active frequency scanning, the EZRadioPRO must be in the TUNE state for the frequency to be modified.

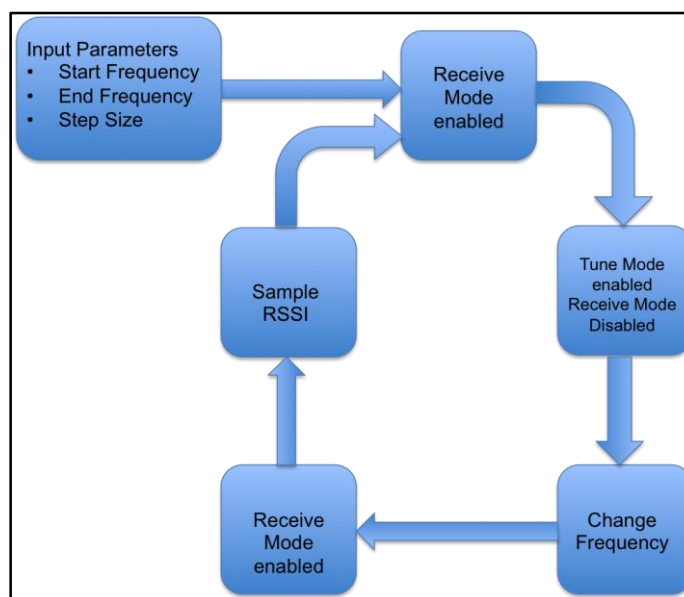


Figure 4-1: EZRadioPRO State Transitions During Frequency Scanning

Additionally, if a user specified range of frequencies to be scanned is below or overlapping between the distinct frequency bands, then each of the four antennas the PE42440 RF switch will be reconfigured to the correct port for that band.

Currently, the functioning code takes an average of the 10 sampled RSSI values and discards the lowest and highest values to remove erroneous data. However, depending on the application, the number of RSSI samples can be adjusted by the end-point controller. However, due to the variable representing the number of RSSI samples being an unsigned 8-bit integer, the number of samples that can be requested is limited to values between 0 and 255. One must also acknowledge that as the number of samples

increase, so will the number of clock cycles required to calculate this average, thereby adding a delay to the time it takes to calculate the average RSSI and then transmitting this data back to the IRIS node.

4.2 IRIS Code Development

Once the code development for the Si1000 was completed, the next set of development tasks involved developing a software that would: (i) enable a MEMSIC IRIS node to control operation of the Si1000 over a serial connection; (ii) the IRIS node would receive a TinyOS formatted packet containing the current frequency and RSSI data from the Si1000 as it performed spectrum sensing; and (iii) the IRIS node would transmit this data wirelessly to the base station. The first development task involved reverse engineering the packet structure used for serial communication within the TinyOS framework. The TinyOS packet format for serial communication can be seen in Figure 4-2.

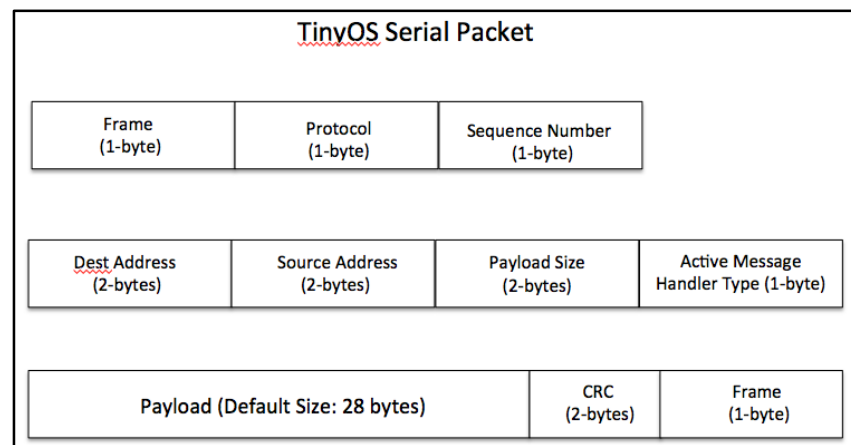


Figure 4-2: TinyOS Serial Packet Format

Custom payloads were then developed for passing control parameters to the Si1000, transferring RSSI data from the Si1000 to the IRIS node, and then relaying this RSSI information wirelessly back to the network's base station. A detailed explanation of the custom payload is presented in Section 4.2.1. Finally, using the example BaseStation *nesC* program I was able to establish a wireless to serial communication interface using an IRIS node. Then the IRIS node is connected to a PDSSP board via the UART pins, therefore providing a serial-based communication that can be used to transfer spectrum-sensing data to the IRIS node. The end product of this integration enables wireless transfer of the commands and data between the sensing node and a base station. The base station is the centralized point of control and data aggregation for the distributed network of nodes that are interfaced with PDSSP boards. Therefore, with the use of multiple sensing nodes (i.e. WSN node interfaced with a PDSSP board), one can deploy this system for applications requiring the need for distributed dynamic spectrum sensing over limited areas. With the current implementation network communication is limited to single-hop network topologies.

4.2.1 IRIS to Si1000 Serial Control Payload Development

Development of the custom payload control and communication protocol is one of the critical elements to enabling the PDSSP attached to a WSN node to perform the environmental RF sensing. To achieve this goal three different payload structures were required: (i) a payload to transfer control parameters from the base station to the IRIS node, then Si1000 on the PDSSP; (ii) a payload to transfer RSSI and frequency data from the Si1000 on to the IRIS node; and (iii) a payload to transfer the sensed data from the Si1000 and the node ID from the IRIS node to the base station.

For initial system development and communication implementation testing, a modified 51-pin connector has been used with jumpers attached to act as a direct connection to the Si1000 development board. Currently, the MIB510 programming board shown in Figure 4-3 is acting as the testing interface for developing and testing the full serial communication protocol, which enables the IRIS node to control the Si1000-based prototype spectrum sensing board.

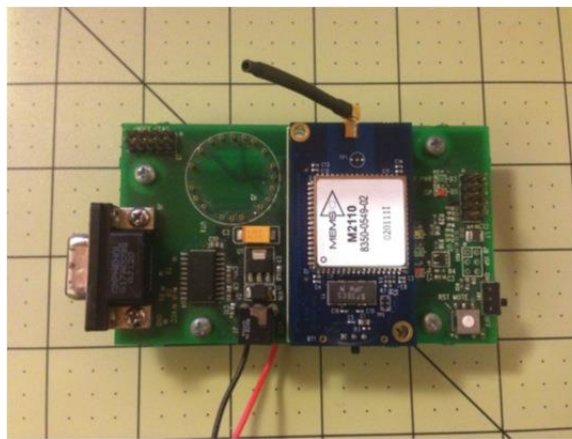


Figure 4-3: MIB510 Programming Board Connected to IRIS Node

To determine and verify the correct structure of the serial packets used for communication using the TinyOS software an IRIS node connected to a MIB510 programming board was loaded with a basic PC-Node serial communication program. The use of a Saleae Logic 16 [25] logic analyzer connected to the UART pins on the MIB510's 51-pin pass-through the connector is shown in Figure 4-4. Serial packets being transmitted between the IRIS node and the PC were captured and decoded.

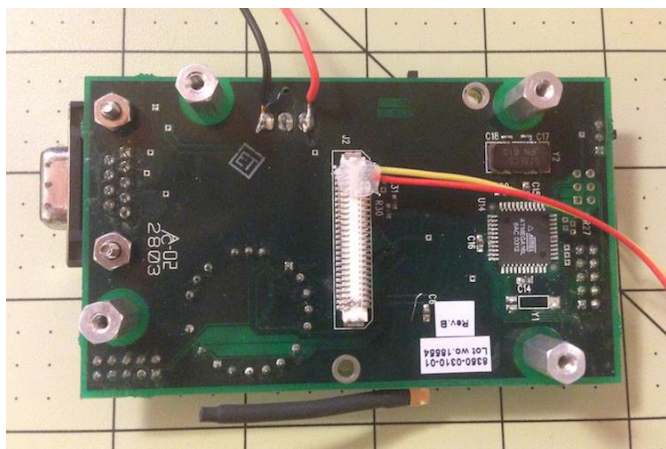


Figure 4-4: MIB510 Pass-Through to UART Output Pins

Communication between the IRIS node and the Si1000 is performed via a custom serial payload packet structure. Figure 4-5 shows an example of this packet payload, which is used for passing the control parameters to the Si1000 for frequency scanning.

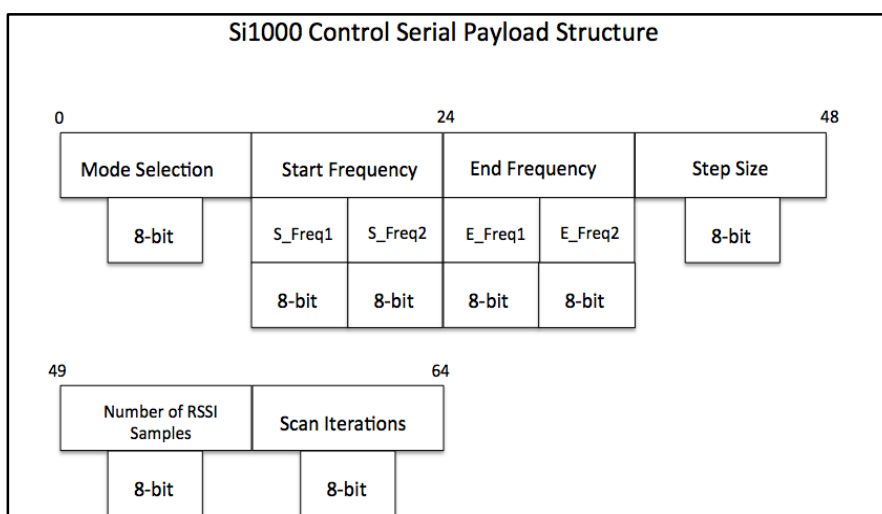


Figure 4-5: Si1000 Control Message Parameters

This packet format shown in Figure 4-5 is communicated from the IRIS node to the Si1000 after the IRIS node has received a scanning command packet from the network's base station. The "Mode Selection" bit of the payload is used to select either

“Single-Shot” or “Continuous” frequency scanning operations. Additional control parameters contained inside the payload provide the ability for a user to specify the scanning range, which is composed of a start and end frequency. The start and end frequencies are processed as 16-bit unsigned integers, but they have to be represented as two hexadecimal values for the serial packet format.

The step size parameter represents the increment that the frequencies will be changed for each frequency change. The final 16-bits of the payload are composed of the number of RSSI samples and iterations. The number of RSSI samples is the parameter a user can utilize to increase or decrease the resolution of RSSI measurements, thereby varying the number of RSSI samples that are collected and averaged for each frequency. However, if this parameter is set to a large value, it will increase the time it takes for a scanning operation to complete. The final 8-bits of the payload are reserved for the number of scanning iterations a user can select only if the mode selection parameter is configured for “Continuous” mode.

Figure 4-6 shows the format of the payload that is transferred from the Si1000 to the IRIS node through the serial connection during each frequency scanning iteration. During frequency scanning, RSSI values are being collected and averaged. This payload contains the frequency and the associated averaged RSSI value.

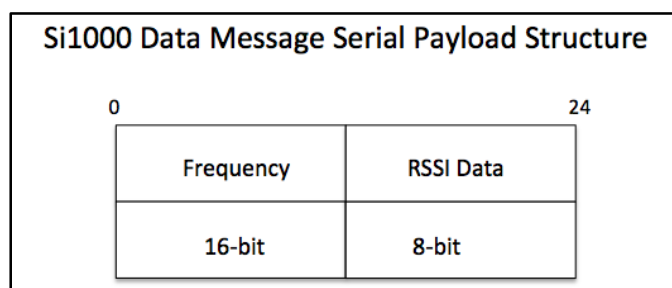


Figure 4-6: Si1000 Data Message Format

Finally, the payload shown in Figure 4-6 is inserted into a packet that is then transmitted wirelessly from the IRIS node to the network's base station. This packet adds the node identification information and provides a method of identifying which node is reporting spectrum sensing information from its unique ID. Therefore, this ID will be correlated with a database of position information for nodes in the network. Location information for each node is acquired during the manual deployment of nodes in the network over the area which is to be monitored.

4.2.2 IRIS to Si1000 Preliminary Integration Testing

Once the code development for the Si1000 and the IRIS node was completed, the next phase of the PDSSP development was the initial communication integration testing of the serial communication payloads between the Si1000 and the IRIS node. Testing was conducted in the following stages: (i) verification of serial packets generated from the Si1000 and the IRIS node using a serial connection with a PC running a terminal program shown in Figure 4-7, (ii) verification of proper serial communication between the Si1000 and IRIS node using a direct connection shown in Figure 4-8, and (iii) verification of

proper transfer of serial packets across the wired interface to the IRIS node which then transmits the data wirelessly to another IRIS node which acts as a base station connected to a PC.

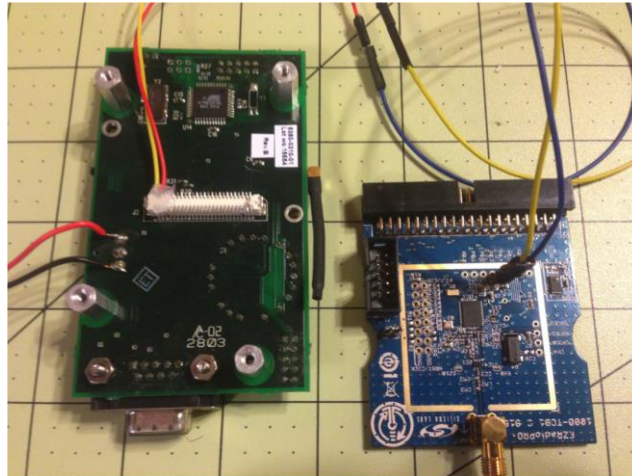


Figure 4-7: MIB510 UART Connection to Si1000 Development Board

The process of integration testing used a modular based method of system testing of the software that was developed, thereby verifying the software's development processes between the Si1000 and the IRIS node was successful in accomplishing the primary goal of this thesis from the perspective of the PDSSP's software and device communication.

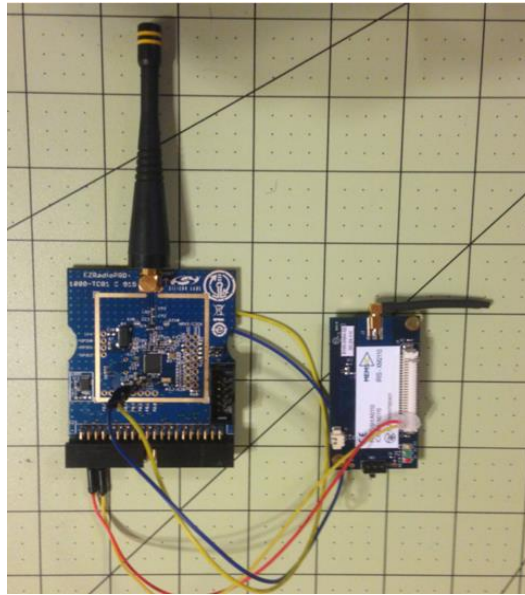


Figure 4-8: Si1000 UART and Power Connections to IRIS Node

In conclusion the design, fabrication, and software development process for the PDSSP involved several areas of engineering design and innovation. However, through an iterative process of the design review, fabrication, software creation and testing the PDSSP is fully operational. Therefore, Chapter 5 provides the experimental results of the PDSSP being deployed within a WSN. This real-world example will be utilized to investigate the capabilities of the PDSSP being used for experimental spectrum analysis.

CHAPTER 5

EXPERIMENTAL RESULTS

5.1 Operational System Testing and Diagnostics

5.1.1 PDSSP Preliminary Hardware Diagnostics

Once the PDSSP had been fabricated, an evaluation of all the electrical connections between components and layers was performed to verify that there were no errors in the fabrication process. This involved a systematic process of checking that all power distribution points from the power plan to each of the primary components was correct in addition to the RF signal paths and digital IO connections between the Si1000 and the PE42440. Once this was verified, power was applied to the PDSSP to verify that the proper voltages were being distributed to the Si1000 and the RF front end components. Next the Silicon Labs software and JTAG programmer was connected to verify that software could correctly communicate with the C2 pins, which are used to read and write to the Si1000's flash memory.

5.1.2 PDSSP Operational System Functionality Tests

The PDSSP operational system tests provided a mechanism for the verification and debugging of the developed software using the Si1000 development boards. These tests were used to confirm that all critical electrical connections from digital pins on the

Si1000 to other components were not misconfigured or damaged during the fabrication process. The first step in this verification process involved making sure that code could be flashed to the Si1000 on the PDSSP board using the Silicon Labs software and JTAG programmer. The code was flashed to the PDSSP via the JTAG pins, which are connected directly to the C2 pins of the Si1000. The C2 pins are used to read and write data to the 8051 MCU registers and load program code into flash memory. Thereby, register values could be viewed using the Silicon Labs development software to show that the system was operational and there were no faults in the initialization of the system clocks, DIO, or additional peripherals that are required for stable operation during code execution.

The primary system components that needed to be checked for proper operation and configuration were the system clock of the 8051 MCU and the initialization of the internally connected SPI pins to provide communication with the EZRadioPRO. To initialize the EZRadioPRO, the SDN pin is required to be put into a logic low state. Once this is performed, a series of commands are sent to the EZRadioPRO over the internal SPI connection to read and write parameters to the EZRadioPRO's memory registers. These registers control the state of the EZRadioPRO's Voltage Controlled Oscillator (VCO), Automatic Gain Control (AGC), filters and digital modem settings.

5.2 PDSSP Deployment via Point-To-Point WSN

5.2.1 PDSSP Integration with MEMSIC IRIS Node

The test configuration provided a method to make measurements of the power consumption of the PDSSP while connected to the IRIS node. Additionally, the serial

communication was monitored during communication tests to verify that the correct packet format and payload data values were being transmitted with our delays or errors.

The integration between the MEMSIC IRIS node and the PDSSP was tested with the two systems connected using a test connector created from a standard 51 pin connector. This interface provides power and a bidirectional UART connection from the IRIS node to the PDSSP. Power was provided to the IRIS node from the Si1000-DK motherboard which supplied 3.3VDC to the IRIS node's battery bypass connections.

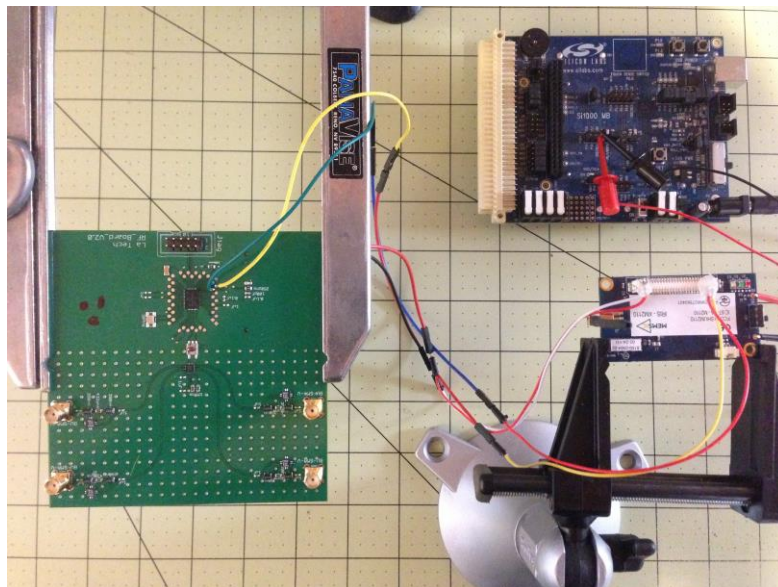


Figure 5-1: IRIS Node Integration with PDSSP for Initial Testing

Once the system was connected and power was applied to the IRIS node, the example Base Station code provided by the TinyOS development framework was used to test the communication between the IRIS node and the PDSSP. This software example is used to create a wire to wireless data communication interface between a PC and a node. Therefore, it can be used to transmit serial data being sent from the PC over the wireless network to one or more nodes.

The next task for evaluating the overall performance of the PDSSP was to determine the total energy consumption of the design. With the PDSSP being designed to operate specifically from the 3.0VDC power source provided by an IRIS node, it was critical to evaluate and measure the operational performance against the device specifications. Presented in **Table 5-1** below are the manufacturer's performance metrics relating to energy consumption.

Table 5-1: PDSSP Component Energy Consumption

Device	Voltage (V)	Current (mA)
Si1000 (8051 MCU +EZRadioPRO)	3.0	22.6
PE42440 SP4T RF Switch	3.0	0.05
PMA-545+ Low Noise Amplifiers	3.0	$80 \times 4 = 320$

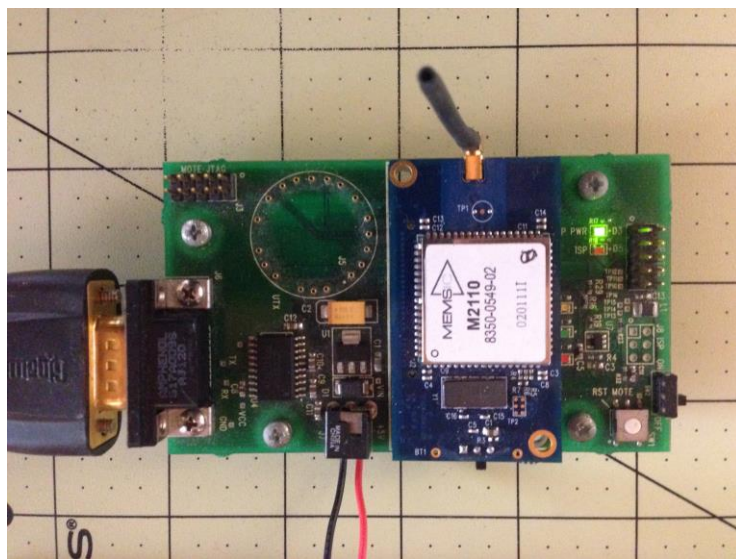
In Table 5-2 the measured performances of the PDSSP while in the initial powered on state as well as during the spectrum sensing and data transfer operational state are presented. We can see that from these metrics that the design of the PDSSP is within the operational energy requirements that can be provided by an IRIS node. However, future advancements of the PDSSP should help to lower the total energy consumption with the addition of transistor-based switching for providing selective power to each PMA-545+.

Table 5-2: PDSSP Operational Energy

Operational Mode	Voltage (V)	Current (A)
Powered On	2.910	0.28
Spectrum Sensing and Data Transfer	2.908	0.28

5.2.2 Point-To-Point Network Spectrum Sensing Experiment

To fully demonstrate the concept of using the PDSSP for distributed spectrum sensing, we performed an experiment where a Point-To-Point Ad Hoc network was deployed using an IRIS node connected to the PDSSP, which acted as the sensing node. The other point in the network was an IRIS node that preformed the actions of a Base Station. The Base Station provided a method to send commands and received data to the sensing node and was connected directly to a PC via a serial-to-USB adapter.

**Figure 5-2: IRIS Node Base Station with RS-232 to USB Connection to PC**

A custom Python script was developed that allows a user to issue parameters to the sensing node. These parameters are used to configure the PDSSP to perform spectrum sensing in either a “Single-Shot” or “Continuous” mode. The input parameters are: start frequency, end frequency, step size interval, and number of RSSI samples for a “Single-Shot” scan. The continuous scan mode requires an additional parameter for “scan iterations” which controls the number of iterations for which a defined range of frequencies will be scanned in a repetitive succession.

Once the input parameters have been passed to the script, they are converted into a payload that is inserted into a properly formatted TinyOS serial packet. Once the packet is created, it is sent over the serial connection to the base station node. Next, the base station node transfers this packet wirelessly to the sensing node. Once the PDSSP has received the packet, it will begin spectrum sensing. The specifics of what occurs during this process are provided in detail in section 4.1.2. Finally, the base station will start receiving packets from the sensing node and transferring them to the PC. The script then decodes these packets extracting the RSSI and frequency values. It then converts the RSSI values to signal power in dBm using the equation shown in Eq. 5-1.

$$\text{Signal Power} = R \left(\frac{10}{19} \right) - 127 \quad \text{Eq. 5-1}$$

This equation is provided in the manufacture’s documentation [21] as the correct conversion from the values transferred from the RSSI registers. Then the signal strength is plotted versus frequency using the Python module matplotlib shown in Figure 5-3.

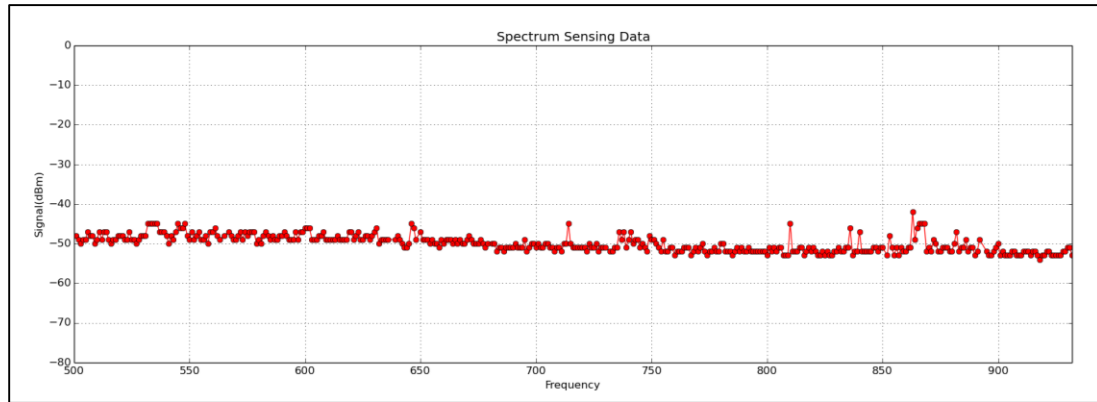


Figure 5-3: Spectrum Data Collected from Scanning 500-933MHz

To verify that the design of the PDSSP is functional for providing spectrum data across the entire operational range of the EZRadioPRO's operational bandwidth we performed scans that gradually increased the selected bandwidth that was analyzed. The scan data presented in Figure 5-4 shows how the PDSSP was able to actively scan from 400-933MHz.

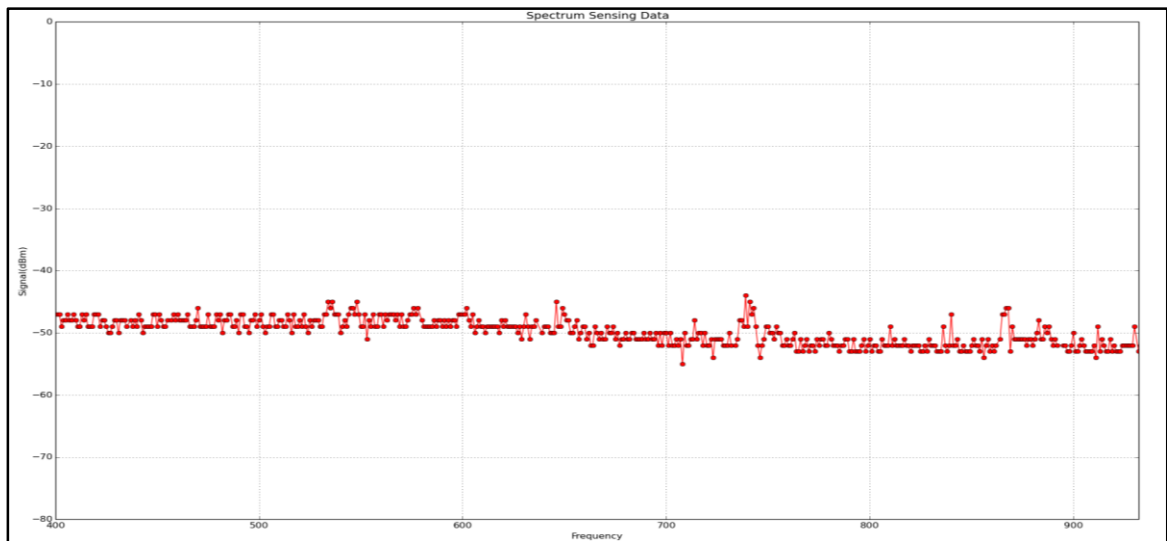


Figure 5-4: 400-933MHz Spectrum Data from the PDSSP

5.2.3 Emitter Detection using Spectrum Sensing Experiment

We verify the ability of the PDSSP to not only perform spectrum sensing, but also to be able to use the spectral data being collected to detect a known emitter within an allocated spectrum which was an additional focus of this thesis. Therefore, we used one of the Si1000 development boards loaded with the code that transmitted a GFSK signal at 920MHz. The PDSSP data was then used to verify that detection through visual inspection was possible. Figure 5-5 shows the results of the spectrum scan with the emitter not active and no signal present at 920MHz.

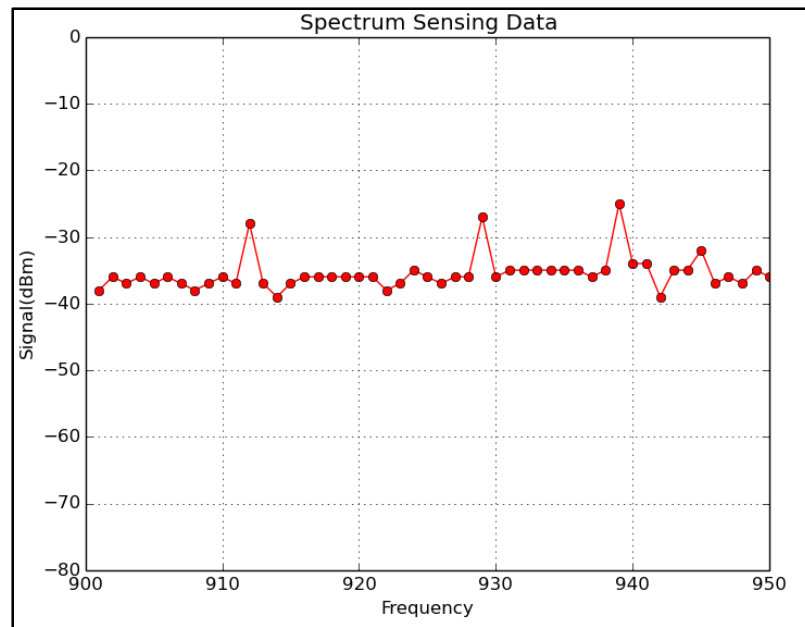


Figure 5-5: Emitter Not Present in the 900-933MHz Spectrum

Figure 5-6 shows the results of scanning the same frequency range with the emitter active and transmitting at 920MHz. The emitter's signal is very strong with a received signal strength of -10 dBm.

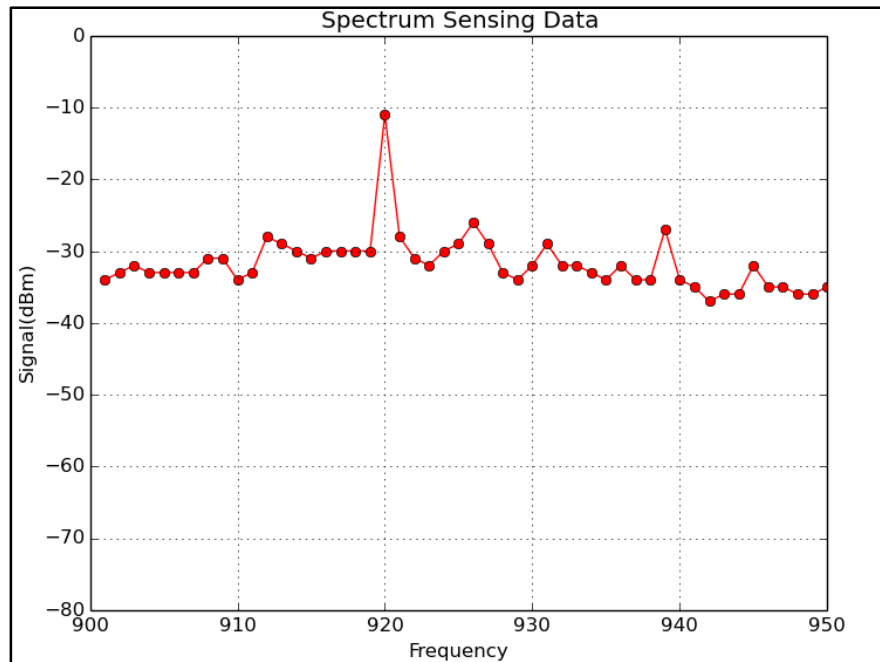


Figure 5-6: Emitter Present at 920MHz

Therefore, due to the restrictions of the Si1000, the development board was only capable of transmitting signals in the 915MHz band. Therefore, we were the only able to use this for specific signal experimental emitter detection. However, with additional test equipment, other signals of interest could be created and used to test the ability of the PDSSP for emitter detection.

Once this experimental setup was carried out, it provided additional information that helps to classify the functionality and operation of the PDSSP while it is being used as a sensor. In this role it enables a small or large collection of wireless sensor nodes to act as a network of discrete spectrum sensing systems. Therefore, this creates the overall combined power to measure the spectrum of an environmental domain using a low-power

and low-cost system. The applications can be used for monitoring spectrum allocations for usage and congestion, hidden node localization, or even specialized applications in the field of electronic warfare.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

Wireless Sensor Networks (WSNs) have been used as the primary platform for deploying distributed sensing technologies. The use of distributed sensing using a WSN for monitoring environmental conditions can increase the data resolution over an industrial or geographical domain of interest. Moreover, this domain-based information can be correlated to provide a feedback to additional systems for overall optimized awareness. Therefore, the need for distributed sensing will become ever more necessary to increase the resolution in measurements across engineering and scientific areas of research.

The use of WSNs member-based localization for network management and topology analysis is used to assess the operation and health of deployed network. During this process, the WSN uses channel and measured signal properties to determine the location of the members in a GPS free environment. The use of signal measurement has been further extended into the realm of cognitive radio with applications focused on spectrum sensing [4] where the task of spectrum sensing is applied to a secondary network of wireless sensor nodes. These nodes provide spectrum information to the primary cognitive radio network. The cognitive radio network uses this information to determine if the congestion of the signals will cause harmful interference to its operation. This need

for real-time spectrum monitoring will be key to the optimal use of allocated regions of the spectrum where congestion is commonplace between primary and secondary users.

The research, design and development conducted for this thesis provide support and further advancement for WSNs used for detection, monitoring, and localization roles in RSSI-based sensing research [5] [2] [18] [20]. The primary challenge of this thesis was to create a low-power spectrum sensing device that could be deployed and act as a sensor. This sensor is designed to provide RF signal measurements within several sub-GHz frequency bands. During the device's operation, band specific RSSI information is collected across a range of frequencies separated by a specific interval. This signal information can be used to create various visualizations of spectrum and channel conditions in almost real-time for channel analysis.

The primary focus of this project was to develop a novel distributed spectrum sensing system. This system required the development of a hardware platform that through a developed software module provided an inter-device communication stack. This communication stack enabled the control and communication between the platform and a wireless sensor node.

The performance of the hardware platform's proper operation has been debugged, tested and verified using the software module. Moreover, the software module enabled full integration of the platform with a MEMSIC IRIS node, which created the basic component of the distributed spectrum sensing network. Experiments were conducted using two wireless sensor nodes in a point-to-point network. This network demonstrated the operational and functional capabilities of the system under deployment to perform distributed spectrum sensing.

6.2 Future Work

The future work related to this thesis will include further enhancements of the current design. Such enhancements will consist of various modifications to the physical dimension and electrical connections from the current PDSSP design. Subsequently, this will involve the reduction of the design's overall size and power distribution.

The physical dimensions will need to be minimized to provide an optimal balance in size so that the integration with a wireless sensor node will not be too large to be cumbersome while connected. The optimization of the design into a smaller form-factor will enhance the ability of the PDSSP to be properly mated with an MEMSIC IRIS node.

However, this will entail other possible enhancements in the way the antennas are placed and fabricated, requiring antennas to be on the RF signal level and embedded within the PCB layers. This will further reduce the profile of the PDSSP and also streamline its compact ability to be placed in various locations without having any obstructions from the platform's enclosure.

The power distribution to the frontend components including PE42440 Low-Noise Amplifiers (LNAs) is another area that needs to be redesigned to reduce power consumption. This will require the use of a transistor-based relay that will be able to power on and off the LNA as needed. The current design provides the LNAs with constant power unless the PDSSP is powered off. Consequently, this increases the current consumption of the design and ultimately requires more power than can be provided by current power sources utilized by basic wireless sensor nodes. The solution to this problem is that when a specific RF pathway is selected, the transistor-based enhancement could be used to power on the LNA. Once the LNA is powered on, it will be able to

amplify the incoming signal. The PDSSP's operational parameters and system losses should be evaluated further using specific testing and measurement techniques to optimize the design further for low-power applications. This will provide a more optimal design that will be more energy efficient for spectrum sensing applications.

Finally, the PDSSP's frequency limits for spectrum sensing will be increased to provide coverage from sub-GHz to 5GHz or above this limit in future designs. This will provide a design that is versatile, low powered, and has an expansive coverage to operate across many of the very active telecommunication spectrums.

BIBLIOGRAPHY

- [1] 2014. [Online]. *Available:*
http://www.memsic.com/userfiles/files/datasheets/wsn/iris_datasheet.pdf.
- [2] M. Gates, C. Barber, R. Selmic, H. Al-Issa, R. Ordonez and A. Mitra, "PADF RF Localization Criteria for Multi-Model Scattering Enviroments," in *SPIE DSS*, Orlando, 2011.
- [3] K. Sohraby, D. Minoli and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*, Hoboken: John Wiley & Sons, Inc., 2007.
- [4] M. Smolnikar, M. Mihelin, G. Berke, G. Kandus and M. Mohorcic, "ISM bands spectrum sensing based on Versatile Sensor Node platform," in *3rd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*, 2010.
- [5] A. Mitra, "Position-Adaptive UAV Radar for Urban Enviroments," *Proc. of the IEEE International Radar Conference*, 2003.
- [6] B. Mercier, V. Fodor, R. Thobaben, M. Skoglund, V. Koivunen, S. Lindfors, J. Ryynanen, E. Larsson, C. Petrioli, G. Bongiovanni, O. Grondalen, K. Kansanen, G. Oien, T. Ekman, A. Hayar, R. Knopp and B. Lozano, "Sensor Networks for Cognitive Radio: Theory and System Design," in *ICT-Mobile Summit*, Stockholm, 2008.
- [7] L. Pescosolido and C. Petrioli, "Wireless sensor networks for spectrum sensing to support opportunistic spectrum access networks: Protocol design and fundamental trade-offs," in *Wireless Communications and Networking Conference (WCNC)*, 2011.
- [8] Zigbee Alliance, "Zigbee Standards Overview," 2014. [Online]. *Available:*
<http://www.zigbee.org/Standards/Overview.aspx>.

- [9] S. Safaric and K. Malaric, "ZigBee wireless standard," in 48th International Symposium ELMAR-2006, Zadar, 2006.
- [10] F. Lewis, "Wireless Sensor Networks", Smart Environments: Technologies, Protocols, and Applications, D. a. D. S. Cook, Ed., New York: John Wiley, 2004.
- [11] TinyOS Alliance, [Online]. Available: <http://www.tinyos.net/tinyos-2.x/doc/html/tep120.html>.
- [12] J. Hill, R. Szewczyk, A. Woo, P. Levis, S. Madden, C. Whitehouse, J. Polastre, D. Gay, C. Sharp, M. Welsh, E. Brewer and D. Culler, "TinyOS: An Operating System for Sensor Networks," *Ambient intelligence*, pp. 115-148, 2005.
- [13] International Telecommunication Union, 8 3 2007. [Online]. Available: <http://www.itu.int/ITU-R/terrestrial/faq/>.
- [14] G. Mao, B. Fidan and B. D. Anderson, "Wireless sensor network localization techniques," *Computer networks*, vol. 51, no. 10, pp. 2529-2553, 2007.
- [15] R. Peng and L. S. Mihail, "Angle of arrival localization for wireless sensor networks," in SECON'06. 2006 3rd Annual IEEE Communications Society, 2006.
- [16] N. Bulusu, J. Heidemann and D. Estrin, "Adaptive beacon placement.," in International Conference on Distributed Computing Systems (ICDCS), 2001.
- [17] T. He, C. Huang, B. Blum, J. A. Stankovic and T. Abdelzaher, "Range-Free localization schemes in large scale sensor networks," in *ACM International Conference on Mobile Computing and Networking (Mobicom)*, 2003.
- [18] L. Lazos and R. Poovendran, "SeRLoc: Secure range-independent localization for wireless sensor networks," in ACM Workshop on Wireless Security (WiSe), 2004.
- [19] K. K. Almuzaini and A. Gulliver, "Range-Based Localization in Wireless Networks Using Density-Based Outlier Detection," *Wireless Sensor Network*, vol. 2, pp. 807-814, November 2010.
- [20] H. Tao, Z. Chen, F. Xia, C. Jin and L. Li, "A Practical Localization Algorithm Based on Wireless Sensor Networks," in *Green Computing and Communications (GreenCom), IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*, 2010.
- [21] Silicon Labs, [Online]. Available: <http://www.silabs.com/support%20documents/technicaldocs/si1000.pdf>.

- [22] Mini Circuits, [Online]. Available: <http://www.minicircuits.com/pdfs/PMA-545+.pdf>.
- [23] Peregrine Semiconductor, [Online]. Available: www.psemi.com/pdf/datasheets/pe42440ds.pdf.
- [24] Mini Circuits, [Online]. Available: <http://www.minicircuits.com/pdfs/TC4-19+.pdf>.
- [25] "Saleae," [Online]. Available: <https://www.saleae.com/logic16>.
- [26] O. S. Oguejiofor, A. N. Andiedu, H. C. Ejiofor and A. U. Okolibe, "Trilateration Based localization Algorithm for *Wireless Sensor Network*," *International Journal of Innovative Science and Modern Engineering(TM)*, vol. 1, no. 10, pp. 21-27, September 2013.