Developing a practical wireless monitoring solution for a size-constrained, low-power, biomechanical, sports telemetry system

Jeffery Ray King
Purdue University

Follow this and additional works at: https://docs.lib.purdue.edu/open_access_theses

Part of the Biology Commons, Computer Engineering Commons, Electrical and Electronics Commons, and the Neuroscience and Neurobiology Commons

Recommended Citation
King, Jeffery Ray, "Developing a practical wireless monitoring solution for a size-constrained, low-power, biomechanical, sports telemetry system" (2014). Open Access Theses. 1028.
https://docs.lib.purdue.edu/open_access_theses/1028

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.
PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By  Jeffery Ray King III

Entitled
Developing a Practical Wireless Monitoring Solution for a Size-constrained, Low-power, Biomechanical, Sports Telemetry System

For the degree of Master of Science in Electrical and Computer Engineering

Is approved by the final examining committee:

THOMAS M. TALAVAGE, Co-Chair

ERIC A. NAUMAN, Co-Chair

DAVID J. LOVE

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University’s “Policy on Integrity in Research” and the use of copyrighted material.

THOMAS M. TALAVAGE, Co-Chair

Approved by Major Professor(s):

Approved by: V. Balakrishnan 12/02/2014

Head of the Department Graduate Program Date
DEVELOPING A PRACTICAL WIRELESS MONITORING SOLUTION FOR A SIZE-CONSTRAINED, LOW-POWER, BIOMECHANICAL, SPORTS TELEMETRY SYSTEM

A Thesis

Submitted to the Faculty

of

Purdue University

by

Jeffery Ray King III

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

December 2014

Purdue University

West Lafayette, Indiana
ACKNOWLEDGEMENTS

Special thanks to my advisor, Tom Talavage, for believing in my skills and motivating me to exceed all expectations. Thanks to my other committee members, Eric Nauman and David Love, for their time to review this work. I would also like to thank all of my labmates who help row the same boat we are all in. A very special thanks to labmates/roommates/friends, Aditya and Blaine, who helped make grad school much more enjoyable.
TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF TABLES</td>
<td></td>
<td>v</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td></td>
<td>vi</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td></td>
<td>vii</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1.1 Background</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1.2 Scope</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>1.3 Outline</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>2. SELECTION OF A WIRELESS TRANSCEIVER</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>2.1 Wireless Transceiver Requirements</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>2.1.1 Fast Data Rate</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>2.1.2 Low Power</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>2.1.3 Strong signal given low profile</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>2.1.4 Large Contributor Community</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>2.1.5 Computationally Light</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>2.1.6 Minimal Signal Interference</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>2.1.7 Multiple Node Communication</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>2.2 Wireless Transceiver Selection</td>
<td></td>
<td>7</td>
</tr>
<tr>
<td>3. DESIGN OF A SENSOR WIRELESS PROCESS</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>3.1 Prototype Stage</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>3.2 Sensor Design</td>
<td></td>
<td>11</td>
</tr>
<tr>
<td>3.2.1 Main Loop Protocol</td>
<td></td>
<td>13</td>
</tr>
<tr>
<td>4. DESIGN OF A BASE STATION WIRELESS PROCESS</td>
<td></td>
<td>17</td>
</tr>
<tr>
<td>4.1 Specific Criteria</td>
<td></td>
<td>17</td>
</tr>
<tr>
<td>4.2 Development</td>
<td></td>
<td>18</td>
</tr>
<tr>
<td>4.2.1 Medium Access Control Implementation</td>
<td></td>
<td>18</td>
</tr>
</tbody>
</table>
4.2.2 Bluetooth Wireless Algorithm ............................................................... 21
4.2.3 The Bluetooth Commands ................................................................. 22

5. ANALYSIS OF WIRELESS ECOSYSTEM .................................................. 24
   5.1 Range of Sensor to Base Station ........................................................... 24

6. CONCLUSION .............................................................................................. 26
   6.1 Summary ............................................................................................... 26
   6.2 Future Work ......................................................................................... 26

LIST OF REFERENCES ..................................................................................... 27
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1. Comparison of Wireless Transceivers</td>
<td>7</td>
</tr>
<tr>
<td>3.1. Sensor Wireless Commands and Descriptions</td>
<td>13</td>
</tr>
<tr>
<td>4.1. Android Application Commands and Descriptions to Base Station</td>
<td>22</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure                                                                 Page

3.1. SparkFun breakout board (red) of Nordic wireless chip attached to BTE_HITS_V02 ................................................................. 9
3.3. PNG Payload Formats: Normal and Confirmation Operations ................................ 12
3.4. Timing Diagram Illustrating Tasks Performed By MSP430 During Sample Period ........................................................................ 14
3.5. PNG Wireless Process Flow Chart ........................................................................ 16
4.1. Illustration of MAC Protocol for Single Nordic Module........................................ 19
4.2. Illustration of MAC Protocol on Base Station with Three Nordic Modules .......... 20
ABSTRACT


As sport-related concussions become more prevalent, the ability to quickly and reliably assess brain injury risk is increasingly essential. Commercially-available systems exist with the goal of assessing the risk of traumatic brain injury in athletes in real-time. These systems utilize a pre-determined acceleration threshold, discarding all captured information below this arbitrary threshold. The use of an event-based model to assess the risk of traumatic brain injury has been shown to be inadequate. Therefore, these systems falsely promote “accurate” real-time communication of risk. Research conducted by the Purdue Neurotrauma Group (PNG) seeks to advance the field by developing a biomechanical sports telemetry system capable of monitoring and continually storing all captured data, while providing accurate real-time communication of risk. This project focuses on the development of a wireless monitoring solution for real-time communication from sensors designed by the PNG. Analysis of commercially-available systems is conducted to illustrate the drawbacks of current wireless solutions. Problem constraints and potential solutions for wireless monitoring are detailed along with the embedded, modular approach utilized by the PNG system. An ecosystem was created with the sensors and a base station using compatible wireless software similar to the sensor software framework. Challenges in implementing the solution are discussed from both hardware and software perspectives, along with recommendations for future work regarding the deployment and expansion of the wireless monitoring solution.
1. INTRODUCTION

1.1 Background

A significant and increasing amount of attention has been directed to sports-related traumatic brain injuries (TBIs), specifically concussions. While most athletes who experience a concussion are able to recover within a short period of time, there are some athletes who experience sub-concussive hits that go undiagnosed and consequently begin developing long-term issues. Approximately 17% of these cases develop chronic traumatic encephalopathy (CTE), of which the exact cause is unknown [1]. Recent research suggests repetitive sub-concussive impacts are likely to cause CTE and other neurodegenerative diseases [2].

The most appropriate solution to combat this issue of sub-concussive hits going undetected is to collect as much head impact data as possible. The collected data will create a history of exposure to head impact events while also developing a predictive model-based strategy to enhance prevention.

Commercially available sports telemetry systems are attempting to provide the end user the assurance of detecting critical head impacts to athletes wearing the system. The Purdue Neurotrauma Group (PNG) has been collecting data using the Riddell HIT System (HITS) for several years with the goal of gathering enough head impact data to design predictive models in order to better conduct risk analysis. However, the PNG has determined the HITS does not accurately record impact data and present that information in real-time to the end-user. The HITS algorithm is event-based using a threshold trigger and a preset data collection window. The system only records 8 milliseconds pre-trigger and 32 milliseconds post-trigger which has been found insufficient with regards to collecting an appropriate amount of impact data [3]. The HITS also falsely promotes
real-time transmission of data from the head units to the base station. Through excessive use of the system by the PNG, impact events were observed before the HITS transmitted the data to the base station and informed the end-user. Lastly, the onboard memory of the HITS head unit is capable of storing 100 impact events [3]. If the HITS head unit is not in range of the base station and the onboard storage is full, potentially significant impacts will not be recorded.

The event-based system of the Riddell HIT System is not suitable for accurately monitoring the head impacts in athletes or providing sufficient amount of data for understanding TBIs. A continuous-time telemetry system must be used in order to develop proper predictive models and to understand the cause of TBI.

The PNG has been developing a biomedical device to fulfill the requirement of a continuous-time telemetry system to monitor athletes during activity. After learning the limitations of commercially available telemetry systems, the PNG has delivered a fully operational sensor system that continuously records raw data from accelerometers and a gyroscope.

1.2 Scope

The PNG has developed both the hardware and software for a continuous-time telemetry system. The hardware is robust and flexible for various types of packaging. The software has been developed to sample the available sensors and record all raw data to an on-board removable storage device. Currently, the current power consumption of the PNG sensor allows about two weeks of data collection.

The primary stages of the device did not include the ability to communicatewirelessly given the steep design challenges. Accessing the information on the device entails removing the device from the packaging and pulling the removable storage to download the data. To combat this time consuming issue, the PNG decided to move forward with adding a wireless component to the sensor.
As mentioned above, there were noticeable design challenges with incorporating a wireless solution to the sensor. The addition could not significantly increase the overall size of the sensor to not be able to have flexible packaging options. The wireless solutions should not draw an excessive amount of current as to exceed the maximum current draw set forth by the PNG.

Overall, there are three stages of implementation for the PNG wireless solution. The initial stage of the wireless functionality will transmit a “heartbeat”, indicating the sensor is powered on, and a count of observed events that exceed a programmed threshold within a one second time frame. The second stage will allow “on-demand” transmission of episodic data of the specific events by providing data covering a specific amount of time. The final stage of wireless development includes the development of additional independent sensors positioned over the body of the player and the creation of a mesh network for gathering detailed monitoring of the players actions.

The initial stage of wireless functionality is the primary focus of this thesis detailing the constraints and solutions while utilizing the embedded, modular approach of the PNG system. The development of a base station allowed for an ecosystem to be using compatible wireless software.

1.3 Outline

Section two discusses the selection process of a wireless transceiver. Two devices are compared using quantified information along with determining the simplicity of integrating such a device into the PNG system. Requirements for future development of the wireless solution are also considered.

Section three illustrates the design and development of the wireless solution using the Nordic nRF24L01+. Problems are discussed regarding the transition of the wireless software from a prototype stage to the PNG sensor. Different software protocols are discussed for interacting with the wireless component and a final software solution is described.
Section four details the design criteria of the base station and the process integrated into the software. A solution is noted regarding the challenge to incorporate real-time wireless communication within the PNG ecosystem. A Bluetooth module is also discussed as an option for interfacing the PNG system with an end-user.

Section five provides analysis of the performance of the PNG sensor and the base station. Characteristics of the PNG ecosystem are described including the capable range and the timing of wireless transmission.
2. SELECTION OF A WIRELESS TRANSCEIVER

2.1 Wireless Transceiver Requirements

2.1.1 Fast Data Rate

In order to classify a wireless system as real-time communication, the need for high data rates is essential. Maximally, all raw data recorded should be capable of being transferred to a centralized base station in an appropriate amount of time. Raw data is being recorded on the PNG sensor at a rate of 2000 Hz with a sample size of 17 bytes. Therefore, every second 33 Kilobytes will be collected and stored to the local memory. A minimum of 264 kbps\(^1\) for the air data rate is required to meet the sampling threshold. This rate does not account for other information needed for communication, such as possible error codes, device identification, or commands. Another factor to consider is the loss of packets during transmission which would begin to delay the buffer needing to be sent to the base station. With a faster air data rate, multiple attempts to transfer data can be made before an influx of data overcomes the system.

2.1.2 Low Power

One major hardware requirement of the PNG sensor system is low power draw. Aditya Balasubramanian describes in his thesis the need to contain power consumption to a minimum [4]. To adhere to the design constraint he discussed, the wireless module used in the PNG sensor system needs to draw as little current as possible while not restricting the other design criteria of the PNG system. In order to quantify, if the system utilizes a 1000 milliamp-hour battery, the wireless transceiver should not exceed 20 milliamps to be considered low power.

\(^1\) kbps = kilobit per second
2.1.3 **Strong signal given low profile**

The physical location of the PNG sensor will be underneath multiple layers of silicone and thick plastic of a helmet. The signal of the wireless module must be powerful enough to penetrate the padding materials and traverse a long distance to the base station. The wireless transceiver should emit enough transmitting power to travel the distance of a typical American high school football field using a chip antenna.

2.1.4 **Large Contributor Community**

The need to develop and implement a wireless solution in a short amount of time is imperative to the PNG study. Having a large community of contributors is highly valued when attempting to develop new technologies. The MSP430 microcontroller was chosen for the broad support community that Texas Instruments established which allowed the PNG production to operate at a rapid pace.

2.1.5 **Computationally Light**

The software required to operate the wireless transceiver should require little computation. To be precise, the wireless transceiver should have firmware preprogrammed to conduct the majority of the wireless transfer of communication and the packaging of data to be transferred. Offloading the low level functions of the wireless transceiver will allow the microcontroller to maximize the time of each clock cycle.

2.1.6 **Minimal Signal Interference**

One major fault with the competitor’s sensor is the interference of the communications with legacy coaching systems that operate at 900 Mhz. The requirement set forth by the PNG is the use of the 2.4 Ghz frequency. The use of the 2.4 Ghz frequency range quadruples the bandwidth available for communication among the sensors.

2.1.7 **Multiple Node Communication**

The extended goal of the wireless solution is to establish a framework for mesh networking. A mesh network will allow communication among all available sensors within the field of
play. This will also allow research to be conducted for localization and begin detected the probability of a player attacking another player from a blind spot.

### 2.2 Wireless Transceiver Selection

Based upon the above requirements, the Nordic nRF24L01+ was selected for the PNG wireless selection. The CC2500 transceiver developed by Texas Instruments was also considered but found to be inferior to the Nordic chip.

The major selection criteria for selecting the Nordic nRF24L01+ was the size of the support community. Several different sources provided thorough explanations and guides for interfacing with the Nordic transceiver. The Nordic chip was determined to be much simpler to operate and included functionality for multiple connections. Table 2.1 summarizes the comparison between the Nordic nRF24L01+ and the Texas Instruments CC2500 transceivers.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sleep Current</td>
<td>400 nA</td>
<td>900 nA</td>
</tr>
<tr>
<td>RX Current</td>
<td>13.0 – 19.0 mA</td>
<td>12.6 – 13.5 mA</td>
</tr>
<tr>
<td>TX Current</td>
<td>11.0 – 22.0 mA</td>
<td>7.0 – 11.0 mA</td>
</tr>
<tr>
<td>RX Speed</td>
<td>2.4 – 500 kbps</td>
<td>250 – 2000 kbps</td>
</tr>
<tr>
<td>TX Power</td>
<td>-12 dBm – +1 dBm</td>
<td>-18 dBm – 0 dBm</td>
</tr>
</tbody>
</table>
3. DESIGN OF A SENSOR WIRELESS PROCESS

3.1 Prototype Stage

In the initial stages of sensor development, several sources for prototyping the Nordic nRF24L01+ were researched. The Arduino Community was found to be a good source of open source libraries for interfacing with a variety of peripherals. Specifically, the Arduino Community contains several libraries associated with the Nordic wireless chip [7]. The Mirf Arduino Library was selected given the functions included closely resembled the procedures illustrated in the nRF24L01+ datasheet [6].

In order to test the feasibility of using the Nordic chip in the PNG sensor design, several Nordic nRF24L01+ development boards were purchased from SparkFun. The boards contained all necessary components for the Nordic IC’s to properly function once powered. The major benefit for purchasing the SparkFun modules was the inclusion of a chip antenna and its simple circuit layout. Given the inexperience of the PNG team with antenna layout, purchasing breakout boards allowed the prototyping phase to move along quickly.

The first test setup included two Arduino Uno boards using the Atmel ATmega168 each connected to one SparkFun Nordic module. A test sample program was found online that demonstrated a server and client ping program using the two Arduino Uno boards. Successfully, the server sent a ping command to the client and the client returned an acknowledgment command to the server. The server then performed simple calculations on information received by the client. The simple ping test demonstrated the operational standards required to properly control the functionality of the Nordic modules.
The next stage of prototyping involved programming a Texas Instruments MSP430F5659 breakout board with code similar to the Arduino client side. The Texas Instruments MSP430F5659, hereby called MSP430, is the microcontroller implemented on the PNG sensor, described in the thesis written by Aditya Balasubramanian [4]. Once programmed as the client, the MSP430 was capable of communicating with the Arduino server.

The next step in developing the wireless was to integrate the Nordic chip into the PNG sensor. The PNG sensors were designed to allow for the SparkFun breakout boards of the Nordic wireless to be plugged in for development, as shown in Figure 3.1. The framework for testing the Arduino code was essentially a barebones version of the software developed at the time for the sensor. That is, all system clocks and initializations were established which allowed the wireless integration to not fall behind schedule.

![Figure 3.1. SparkFun breakout board (red) of Nordic wireless chip attached to BTE_HITS_V02](image)

Developing the library and main loop function for the MSP430 in conjunction with the Nordic wireless was straightforward. The Arduino code structure is very similar to the code structure required for the PNG sensor software, which is C. When testing the operation, the Nordic chip did not operate as it did when connected to the Arduino. After debugging the code and finding no errors, analysis of the main MSP430 code development was performed. The problem was discovered to be the difference of clock speeds between the Arduino and the PNG sensor.
When prototyping on the Arduino boards, the timing issue was not found to be an issue given the clock speed of an Arduino using an Atmel ATmega168 is 8 Mhz [8]. The library developed for the Arduino does not have any timing delays to allow for settling period of the transmitter or receiver of the Nordic chip. Since the clock speed of the MSP430 is operating at 20 Mhz, software delays were added into the new library to adhere to the Nordic nRF24L01+ radio control state diagram timing as shown in Figure 3.2. With the addition of the software delays, the barebones PNG sensor began to successfully operate with the Nordic chip.

Figure 3.2. Nordic nRF24L01+ Radio Control State Diagram [6]
3.2 Sensor Design

The current goal of the PNG sensor is to allow the end user to know if the sensor is powered on and collecting data. This includes a more robust and complete pinging process, shutdown feature, and calendar option. There are several procedures required to make sure the base station does not skip any information and that the information received is valid.

Immediately when the sensor is powered on, the sensor will continuously transmit a ping signal for about ten seconds. The idea behind this process is to provide the base station ample time to record at least one ping from that sensor. For example, if the sensor is turned on and the player immediately runs onto the field, the base station should be able to record a ping before becoming out-of-range. The first ping is essential to knowing if the sensor is operating correctly. If the base station does not receive any ping during the first ten seconds of the sensor turning, the sensor can still inform the base station of being powered up but at a much slower rate.

Once the ten second frame has ended, the sensor will enter a slow transmission rate of one ping per second. Much slower rates were considered to decrease power consumption but were not chosen because of the early development stage of the project and the necessity for debugging the circuit while in the field.

The package of the ping used for each ping consists of the sensor’s unique serial number (USN) and one extra byte which makes the ping payload size be five bytes. Although the Nordic wireless payload maximum is 32 bytes, the PNG decided to have a smaller payload for quicker transactions during the early stages of development [6]. The USN was packaged in the payload to provide the base station the capability to check if the information received is valid. The Nordic wireless has a built in auto acknowledgement which assists with the transmission between two Nordic chips. However, this auto acknowledgement does not verify that data sent by the sensor is valid for the PNG ecosystem. That is to say, the base station needs to verify the sensor did not scramble the payload package before transmitting.

The last byte packaged in the payload has two purposes. The main purpose of the last byte is to inform the base station that an error code has been triggered. The error codes have
been established to inform the user of failures with the PNG system, primarily focusing on the storage system. If an error is not fatal to the system, the wireless process will package the error code and transmit. However, a fatal error code will immediately halt the system subsequently stopping the wireless process from transmitting. The second purpose of the last byte is to allow for an acknowledge byte. This acknowledgement byte provides the PNG ecosystem the ability to confirm transactions between the sensor and the base station. Further discussion of the acknowledgement byte within the payload is discussed below.

There are three commands implemented on the PNG sensor for communication with the base station. The first two commands correspond to the calendar feature while the last command is associated with shutting down the sensor.

The calendar real-time clock (RTC) setup on the MSP430 requires five bytes of data to properly set. Only 38 bits are utilized in the five bytes of data therefore the command codes can only be two bits. The command code for setting the calendar RTC is 0x01 while accessing the current calendar on the sensor is 0x02. The codes and their respective commands are presented in Table 3.1.
Table 3.1. Sensor Wireless Commands and Descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Hex Value (Bits 7,6)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set Calendar</td>
<td>0x40</td>
<td>Base station sends 5 bytes of RTC data to sensor</td>
</tr>
<tr>
<td>Get Calendar</td>
<td>0x80</td>
<td>Base station requests 2 bytes of RTC from sensor</td>
</tr>
<tr>
<td>Shutdown</td>
<td>0xC0</td>
<td>Base station sends 0xC0 to power down sensor</td>
</tr>
</tbody>
</table>

Once the sensor receives the calendar RTC data and performs the require operations, the sensor will reply to the base station with the normal payload pack of the USN but instead of using the error codes, an acknowledgement byte is concatenated to the USN buffer. This confirmation byte is 0xA4 and is transmitted five times before the sensor returns to normal operating conditions. The confirmation byte allows the base station to verify the sensor had received the calendar RTC data and has successfully processed it.

The command for shutting down the sensor has an associated confirmation byte of 0xB5. When the base station requests the sensor to power down and the sensor receives 0xC0, the sensor will continuously transmit the USN and 0xB5, similar to the fast transmit start mentioned above. An important issue to note is the requirement for the command byte should never be 0xFF in order to restrict false shutdowns to the sensors. If the value 0xFF is received by the sensor, the command will simply be ignored assuming an error in transmission has occurred.

3.2.1 Main Loop Protocol

After the addition of the commands and verification took place using a barebones PNG sensor board, the next step was to begin implementing the wireless code into the main code used to program the PNG sensor.

As mentioned above, the code developed for the wireless code was first written on an Arduino and then adjusted to operate on the barebones PNG sensor board. When integrating the wireless code into the main loop of the PNG sensor, several fatal problems became apparent.
The wireless code was developed using long recursive delay functions that took up a significant amount of time from the main loop. Since the main code was integrated with error features, the newly added wireless code halted the system and displayed an error code associated with skipping a sample cycle.

The main code loop is designed to operate at 2000 Hz and collect sensor data every cycle. Figure 3.4 provides a timing diagram illustrating the duration of the data collection and storage during the sample frame. The analog conversion and digital query of the sensors operate jointly and begin each sample frame. The next step is to convert the collected sensor data into a storage format. During this conversion time, other operations can be performed by the MSP430. The final step within the sample period is to store the collected data to the microSD card which takes the majority of the remaining time.

![Figure 3.4. Timing Diagram Illustrating Tasks Performed By MSP430 During Sample Period](image)

In order to combat the strict timing requirement, a redesign was needed for the wireless code process. Given the time during the data conversion and the downtime throughout the sample period, additional activities to be performed by the MSP430 is estimated to be 150µs. The wireless code previously developed needed to be less computationally heavy in order to have the microcontroller provide ample processing time to the other higher priority segments. Instead of using large functions and IF statements, a switch-case control mechanism was used to allow the compiler to maximize optimization techniques. It is a proven fact that switch-case statements do not rely on earlier cases, such as IF statements, which allows the compilers to optimize the structure. By splitting that larger functions used within the wireless process into smaller, quicker operations, the main code of the PNG
sensor can operate normally without lag while the wireless meets the communication requirements set by PNG.

The wireless control process is complex and the use of a switch-case control mechanism alleviates some of the computation requirements. The overall wireless process contains thirteen cases that are executed in a semi-random fashion. Therefore, to keep track of the progression through the process, flags are implemented to direct flow through the following flow chart shown in Figure 3.5.

As mentioned above, the Nordic nRF24L01+ requires precise timing of delays in order to operate. Throughout the prototyping phase, delays were incorporated to facilitate the timing requirements. However, the delay function pauses all operations of the processor. This method is not ideal because the storage process uses a timer and when the delay is activated, the timer is paused which makes the system throw an error for missing a sample period.

A solution for the delay function was the ability to utilize the same clock the other timer processes are using within the PNG system. Since the framework was already established for the timer functionality, it was not difficult to implement the new wireless timer. The use of a switch-case mechanism allowed the timer for the wireless process to easily integrate. For example, when the timer triggers and stops then a flag is set to inform the wireless process to move onto the next stage.
Figure 3.5. PNG Wireless Process Flow Chart
4. DESIGN OF A BASE STATION WIRELESS PROCESS

4.1 Specific Criteria

The Base Station software is compatible with the software framework implemented on the PNG sensors. Because of this, the wireless hardware of the base station can be similar to that of the sensor system mentioned above. Since the size constraints of the base station is negligible compared to the sensor, the wireless module was exchanged with a similar version with more enhanced power amplification. Also, a Bluetooth module was included in the base station design to allow the end user to communicate with the sensor remotely.

Prefabricated modules were purchased utilizing the Nordic nRF24L01+ chip with two additional chips to amplify the output signal. This module includes a Low Noise Amplifier (LNA) and a Power Amplifier (PA) allowing the output signal to be amplified by 30 dB compared to the standalone Nordic chip[9]. The prefabricated module also includes a SubMiniature version A (SMA) connector allowing different sizes of antennas to be used. From a preliminary search, SMA antennas were found to have a gain of up to 9 dBi and possibly higher. The combination of amplifier chips and an expandable antenna theoretically allows the base station to cover a large range of distance in order to communicate with more sensors in the field.

As mentioned above, the base station is not size constrained allowing the packaging to be large enough for a large battery. A larger battery allows the base station to maximize the peripheral components in terms of signal range at the cost of power consumption. Therefore, the Bluetooth module was chosen to have a long range which, consequently, increased the current draw. The typical Bluetooth radio found in consumer electronics is the class 2 radios that have a range of 33 feet [10].
The PNG chose to maximize the range of the Bluetooth radio by utilizing a class 1 radio which has a range up to 300 feet. The Bluetooth module RN41 by Roving Networks consists of a class 1 radio and is compatible with most devices with Bluetooth. The long range of the Bluetooth radio allows the end user to power on the Base Station and walk around the field as needed without losing connection between the two devices.

4.2 Development

The goal of the base station is to continuously scan for available sensors and report information to the end user via Bluetooth. In order to achieve this goal, multiple enhanced Nordic modules were connected to the MSP430F5659 to alleviate some of the computation needed to communicate with several dozen PNG sensors. Only three wireless modules were connected to the MSP430 given the limiting factor of available DMA channels.

The DMA channels were implemented into the base station code to allow the Bluetooth to have priority when looping through the main code structure. The Bluetooth module communicates with the MSP430 using UART and an interrupt-based routine. Using UART on the MSP430 has a limitation of having a buffer size of only one byte. An interrupt-based routine allows the MSP430 to jump in and out normal operations to store one byte of data into a larger buffer for future processing.

The Bluetooth process uses a case-switch operation within the interrupt to quickly determine what information is being received. Flags are implemented to allow the program to preemptively know what information should be incoming and how the main loop should interact with the information once fully received.

4.2.1 Medium Access Control Implementation

In order to communicate with all sixty sensors within a reasonable amount of time, the base station will need to process the sensors rapidly. A Media Access Control (MAC) protocol was designed to regulate the sensors connected to the base station.

The premise behind development of the MAC protocol was the fact that the Nordic chip required the radio to be powered down when altering the receiver address. However, when
changing the channel number the radio uses to communicate to the specific address the
Nordic radio does not require a power cycle.

The MAC protocol takes advantage of the Nordic five pipe capability by establishing a set
of five permanent addresses. These five addresses are programmed equally among all
sensors while each group of five are associated with a different channel. To clarify, if sixty
sensors are connected then twelve sensors will have the same address but different channels.
Figure 4.1 illustrates the concept of utilizing a single Nordic module on the base station
with the MAC protocol.

![Diagram of MAC Protocol](image)

Figure 4.1. Illustration of MAC Protocol for Single Nordic Module

Expansion of the MAC protocol to operate among the three Nordic modules on the base
station allowed for the rapid communication to all sensors. As shown in Figure 4.2 each
Nordic module is assigned to a specific set of channel numbers. The number of channels is not limited and is expandable along with the scheme of assigning the channels to each Nordic module. The base station will use a scheme similar to round-robin to retrieve the data from the sensors on each channel. The base station determines which sensor is currently communicating by unpacking the payload and obtaining the Unique Serial Number. If a sensor is not in range during the scanning of the specific channel then after a predetermined time the base station will move on to the next channel for that Nordic module. Each Nordic module is scanning the respective channel at a rate independent of the other Nordic modules.

![Figure 4.2. Illustration of MAC Protocol on Base Station with Three Nordic Modules](image)

The method of implementing a MAC protocol allowed for a faster retrieval of data from the sensors but the base station requires less computational power. The storage of sixty
individual 5-byte addresses is not required for communication. The base station simply needs to keep track of which channel the Nordic is communicating and through the use a matrix type protocol, the sensors and base station can be linked.

### 4.2.2 Bluetooth Wireless Algorithm

Given that only three wireless modules can be connected to the MSP430, the process of communicating with available sensors in the field was difficult. Assuming at most sixty sensors powered on during one session, the base station will need to communicate with all sensors in a reasonable amount of time to represent real-time.

Given each wireless module has five pipes, the wireless module can communicate with at most five sensors. Multiple solutions were considered when designing the base station communication scheme.

The first algorithm considered for communicating with the maximum number of sensors in the field of play was similar to round-robin scheduling. Round-robin scheduling is an algorithm used to allocate equal time to all processes. With regards to this wireless solution, the round-robin scheduling would be implemented by allowing each sensor to have ample time to communicate with the base station using one of the three available wireless modules. Since the sensors transmit a ping every second, the base station would be required to have a sample time larger than one second to assure receipt of the ping. The time required would be too much to illustrate a real-time system to the end user.

Another fault with the round-robin type algorithm is the need to change the address for each incoming sensor ping to an available base station wireless module. The base station would have to take time to power down the receiver of the wireless module, change the address, and then begin receiving again. This protocol is not ideal because of the time required to change the address. The amount of storage required for such a feat would be too large. Also, the required computation for looking up the next address would be too heavy for the system resulting in an inaccurate real-time solution.
Another algorithm was considered that would be computationally lighter and require less storage space. Since the wireless modules can communicate with up to five sensors using the same channel, the solution considered keeping the addresses constant while changing the listening channel. Therefore, only five addresses would need to be stored on the MSP430 and an algorithm would be implemented to understand which sensor ping was received.

Using the channel algorithm, a timer would still need to be implemented to allow for the sensor to transmit the ping or return into range of the base station. If the base station does not receive a ping in the allotted time frame, the base station will zero out the corresponding bit in a 60 bit buffer. This buffer is read from the Bluetooth process on the MSP430 in order to transmit to the end user via an Android application (App).

4.2.3 The Bluetooth Commands

Several commands have been implemented on the base station for communication with the Android application. The commands allow the user to know the state of the sensors, turn off the sensors and update the time on the base station using the Android clock. Table provides a list of commands the android application sends to the base station.

<table>
<thead>
<tr>
<th>Command</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pull Status of Sensors</td>
<td>0x11</td>
<td>Android App requests number of sensors powered on</td>
</tr>
<tr>
<td>Send Update Time</td>
<td>0x13</td>
<td>Android App sends current time to base station</td>
</tr>
<tr>
<td>Send Shutdown</td>
<td>0x15</td>
<td>Android App sends command to shutdown sensor</td>
</tr>
<tr>
<td>Pull Shutdown Count</td>
<td>0x25</td>
<td>Android App requests number of sensors powered off</td>
</tr>
</tbody>
</table>

The status of the sensors is not only on or off. Since the sensors have the potential of being out of range of the base station, the status of that sensor would be unknown; neither on nor off. Therefore, an out-of-range state is required to know if the sensors have been turned on
but its current status is unknown. The base station will require the use of three 60 bit buffers to know if the sensor has been turned on, if the sensor is pinging, and if the sensor has been turned off. The base station will transmit only the on and out-of-range buffers during normal use. The off buffer will only be requested from the Android application when the user transmits the sensor off command.

Along with the three status commands, the Android application has the ability to send the time and date to the base station. For the overall PNG sensor ecosystem, the calendar buffer is predetermined for the use in the calendar library. Therefore, in order to streamline the synchronization of the time and date, the Android application assembles the information in a specific form before transmitting. Once the calendar buffer has been transmitted to the base station, the base station can then use the calendar sync command to update each sensor.
5. ANALYSIS OF WIRELESS ECOSYSTEM

5.1 Range of Sensor to Base Station

Members of the Purdue Neurotrauma Group tested the range of the sensors in two different phases using a prototype of the base station, specifically the Arduino test bed. Primary testing of the range using a prototype design of the wireless software was performed inside a long hallway. This test resulted in a range between the sensor and the base station to be about 50 yards. Given the amount of interference within the building, the result was promising with regards to the capabilities of the final product.

A more representative testing was performed to determine the range of the sensors. For this phase, the sensor was packaged in a form factor similar to the final package layout but without the shrink wrap encompassing the sensor package. Four different tests were performed – with and without a helmet and placing the base station at two different heights. Throughout the test, the sensor was consistently positioned at a height of about 4 feet.

The first test consisted of placing the base station on the ground and testing the range of the sensor without a helmet. This test resulted in the sensor reaching a maximum of 50 yards before losing communication with the base station.

The second test involved positioning the base station at the 50 yard line and at a height of 6 feet. Without a helmet, the sensor was able to continue communicating with the base station until the far distance of the field at the zero yard line. The resulting range of communication is roughly 71 yards straight to the base station from the sensor.

The third test was similar to the first subtest by placing the base station on the ground again and measuring the range of the sensor. The helmet was used for this test to
determine the effects of the hard shell to the range of the sensor. The sensor began to lose connection with the base station around the far hash marks at the 20 yard line. The resultant range is roundly 45 yards straight to the base station.

The final test was a combination of the helmet and positioning the base station at a height of 6 feet. While in line-of-sight, the sensor continued to communicate with the base station at the far corner of the end-zone. The range between the sensor and the base station is now about 80 yards.

During implementation of the Nordic wireless, the distance was noted to be about 20 to 30 yards between two sensors using chip antennas on the sensor. The tests described above prove the effect of using the enhanced Nordic wireless module for the base station to be a significant increase in coverage. The PNG attempted to have a range of roughly 53 yards to cover the width of the football field in order to receive information once the players cross the midpoint of the field. Now, the range has exceeded expectations and is capable of reaching both far end-zones during game play. One major note to mention is the loss of signal when the sensor is blocked by an object. In other words, direct visual line-of-sight is recommended when using the sensors installed into helmets.
6. CONCLUSION

6.1 Summary
The work presented in this thesis describes the development of a wireless ecosystem using an embedded, modular approach. The Nordic wireless transceiver was selected for the long term development of the PNG wireless solution. The addition of the wireless solution resulted in a slight increase in size and power consumption while allowing full wireless integration.

6.2 Future Work
As mentioned in the scope of work, the future work of the wireless solution includes the expansion of transferable data to the base station. The PNG is determined to transmit and receive episodic data using the framework similar to the current wireless ecosystem. The current integration of the Nordic wireless chip has the potential to deliver a large amount of data. However, real-time communication of data using the current system is unknown and further research and development is recommended.

The base station also requires further research and development. The current design was quickly prototyped in order to deploy at the same time as the sensors. Therefore, the wireless solution for the base station is not desirable because of the need for multiple antennas and Nordic chips. Ideally, the base station would have a single antenna to communicate with all available sensors. Using a single antenna with a more complex processor, the base station could potentially accomplish the real-time communication goal set by the PNG. Future research is recommended to understand the technology needed to achieve a more robust base station using a single antenna system.
LIST OF REFERENCES
LIST OF REFERENCES


