School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, BC
V5A 1S6

May 6, 2009

Mr. Steve Whitmore
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
V5A 1S6

RE: ENSC 440 Post Mortem for Watchbird™ Home Monitoring System.

Dear Mr. Whitmore,

"The Watchbird™ Remote Peace of Mind System: Post Mortem", attached to this letter, details the wrap-up comments for the Watchbird™ system.

The goal for the Watchbird™ prototype was to create a "proof of concept" peace-of-mind home monitoring system to allow users to remotely check the status of various sensors in their homes, as well as lock or unlock their door remotely. Chickadee Tech has successfully built this prototype and demonstrated its functionality.

This document first presents the manner in which the realized Watchbird™ prototype's design differs from that presented in "The Watchbird™ Remote Peace of Mind System: Design Specifications". It then presents some ideas for future work on the system to prepare for production level manufacturing.

Please let me know if you have any questions, comments, or concerns regarding this document. I can be contacted at 604.837.4009 or by email (smg2@sfu.ca). Thank you very much for your time in reviewing the Watchbird™ post mortem.

Regards,

Samantha Grist
President
Chickadee Tech

Chickadee Tech

The Watchbird™ Remote Peace of Mind System:

Post-Mortem

Samantha Grist
Derek Lowes
Andra St. Quintin
Shiou-Min (Eric) Shen

May 6, 2009

Table Of Contents

List of Figures

List of Tables

1 Introduction

Through hard work and careful planning, Chickadee Tech was able to bring the idea of a cell phone operated home monitoring system to realization. The changes not documented in previous specifications will be described in this report, as well as improvements and features this project could expand upon. To conclude this report, each member will reflect on their experiences throughout this project.

2 Current Status

At the concluding point in the prototype development, Chickadee Tech has successfully implemented the home monitoring system proposed in January 2009. Using cell phone text messages, we are able to lock and unlock a door as well as query the status of three sensor locations. This communication has been implemented using a server running on a company laptop in conjunction with a base station in the user's home. We have also implemented a web-based user interface, which allows users to configure their systems. The overall system design as currently implemented is shown in Figure 1.

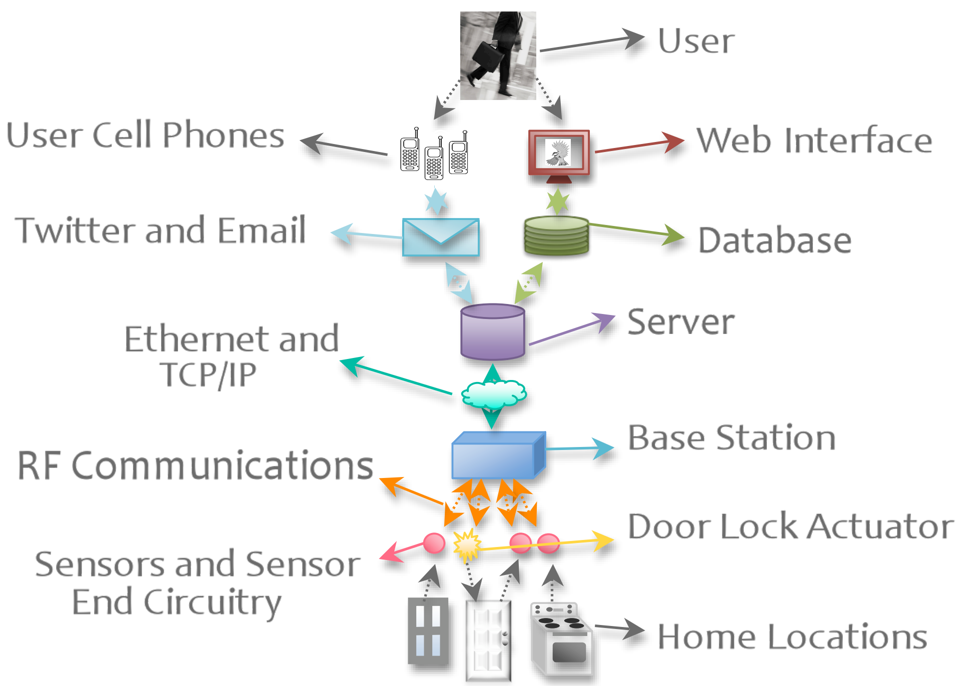

Figure 1: System Overview

Communication from cell phones is achieved using Twitter, which converts text messages into emails that are then in turn read by the server. Communication to the cell phone is implemented directly from the server, whereby emails sent by the server are automatically converted to text messages by the user's service provider. The web interface and database were constructed to store user information and to allow the server to access the appropriate user information when processing requests.

In the user's home, the base station is connected to an Ethernet port to enable communication with Chickadee Tech's server. It then coordinates communication with all sensors in the user's home. RF signals for actuating or querying are sent to each sensor in turn, using unique address bits to identify each sensor. The base station then compiles the status of all sensors and responds to the server.

We have demonstrated that the sensor-end enclosures and the sensors themselves can be mounted to walls and appliances in the user's home with essentially no damage to these locations. The contact switches proved to be versatile for mounting on door locks, ovens, and windows; and we are confident that they could be used in a wide variety of other locations.

3 Deviation from Specifications

The Watchbird™ prototype did not exactly follow the design specifications in some areas. These changes from the specifications are outlined in this section.

3.1 Timing Design

Due to the problems we experienced with the RF receivers and transmitters, we increased the number of tries of the query or lock request to 4. This increased the worst-case time for a query (per-sensor) from 2 seconds to approximately 4 seconds. For prototyping purposes, this delay was negligible compared to the text message delays (and the maximum number of tries was rarely used). This change increases the worst-case time for a query from 31s to 51s and the worst-case time for a lock/unlock request from 35s to 59s.

3.2 Cell Phone – Server Communication Protocol

For the cell phone to server communication protocol, there have been no deviations from the specifications for the user to server communication. For the server to user responses, however, there have been a few changes.

Firstly, the lock/unlock confirmation messages and the incorrect password messages are no longer used. Instead of sending a lock or unlock confirmation message, the server simply sends a status update message for any successful user request. Since this message contains the new status of all sensors anyway, the user can immediately tell that their door was locked or unlocked successfully. The incorrect password message is no longer used because with the current design, the prototype is not able to support it. This problem arises because to generate a text message reply, the server needs to know the user's service provider. Without a valid cell phone number and password pair, the server will not read any user information from the database.

Finally, there are now some additional error messages the server can send as a reply to the user request. These are shown in Table 1.

Table 1: List of new types of text message responses with examples.

Message Type	Example Message
base station disconnected	Base station not connected.
base station busy	Base station currently busy with another request. Please try again in a minute.
error reading reply	Invalid reply from base station.
wrong message type	Invalid reply from base station.
invalid sensor name	Actuator name not found.

3.3 Server

3.3.1 Hardware

There were no deviations to the server hardware. As intended in the design specifications, we used an existing company laptop to help avoid any unnecessary costs in the prototype.

3.3.2 Software

The server software has been simplified by integrating the user configuration into the query and lock/unlock requests rather than having additional configuration messages. Instead of continually checking the database for any changes, the server only reads the user configuration data when it receives a request from that user.

3.4 Server/Base Station Communication Protocol

Due to the changes to the server design, the sensor configuration request and sensor configuration confirmation messages are no longer used. Instead, whenever a query or lock/unlock request is generated from the server, that request will include a list of all currently connected sensors.

3.5 Base Station

The PICDEM.net 2 development board was used as our base station hardware. Only a few minor changes to the hardware and software of the system occurred.

3.5.1 Hardware

The Linx LR series receiver and transmitter were indeed used at the base station end. They were, however, connected to a 3.3 V power supply rather than the 5 V power supply proposed in the specifications. This change was easy to implement and required less components than the original design. Furthermore, a digital I/O pin was used as input from the receiver, instead of the analog one outlined in the design specifications, so the RSSI pin of the receiver was not used in the final implementation. Through testing, it was found that the receiver output was suitable for digital input to the microcontroller, which simplified the design of the system. The revised interface is shown in Figure 2.

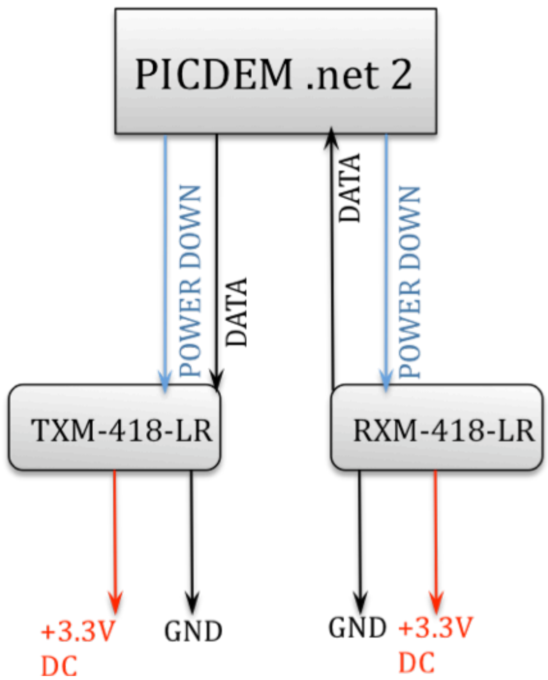

Figure 2: Revised base station-RF interface

Additionally, slightly different circuitry was used at the base station compared to that presented in the design specifications (a result of the change in power supply voltage). This circuitry is shown in Figure 3.

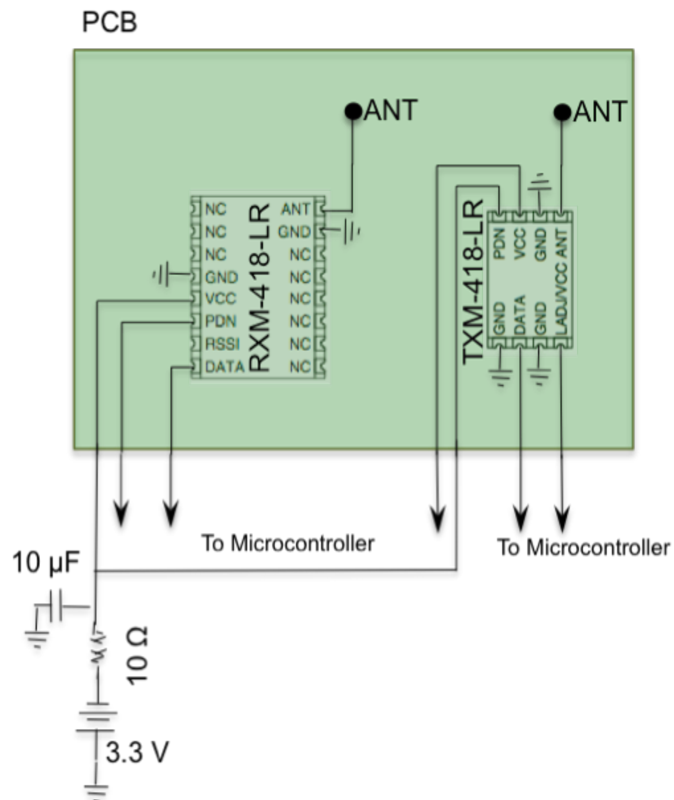

Figure 3: Revised RF circuitry at base station.

3.5.2 Software

The only significant change to the base station software was that the receiver is only powered up after transmitting has completed. This was chosen both to help conserve power as well as to avoid any damage to the receiver's sensitivity that might occur from having full-strength transmission occur at such a close proximity.

3.6 RF Transceivers

As stated in the design specifications, we implemented our design using a Linx LR transmitter and receiver at the base station, and Linx KH2 transmitters and receivers at the sensor locations. However, we were not able to attain the full range expected from these modules, as is discussed further in the section for future plans.

3.7 RF Transceiver/Sensor/Actuator Circuitry

A few changes were made to the design of the circuitry at the sensor locations. The most significant was the elimination of the AND gate chips and inverter from our delay circuit. We originally understood that the 555 timer chip required its input to be generally held high, and triggered by a short negative pulse. We understood that it would not work to have the input to the timer generally held low, and that the falling edge of a short positive pulse would not correctly trigger the timer. As a result, we required an inverter at the input, as well as an AND gate at the output to ensure there was no overlap between the received and transmitted pulses.

However, while testing our circuits, we discovered that it was actually possible to trigger the timer with a positive pulse, and thus eliminate the AND gates and inverter in the circuit. The revised circuit is shown in Figure 4.

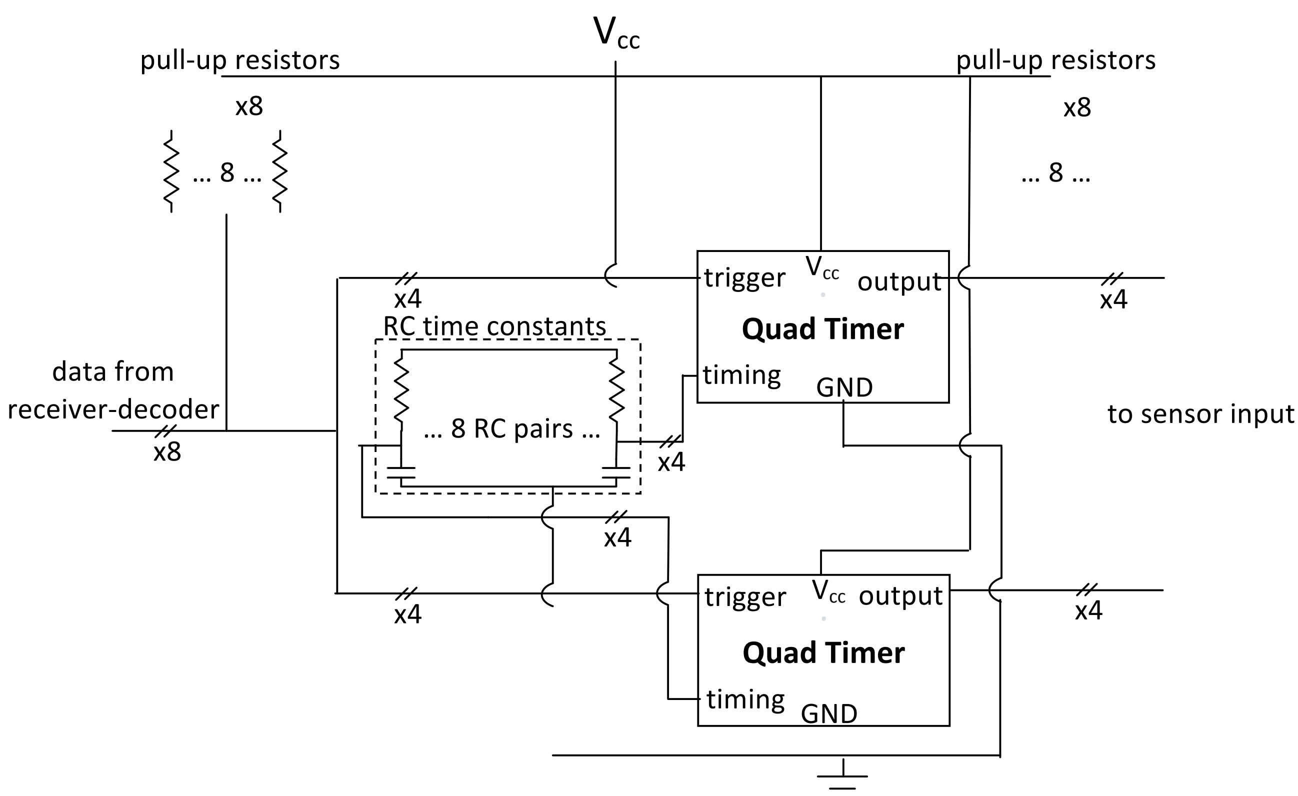

Figure 4: Revised time delay circuit.

This change reduced the size and complexity of our circuit, as well as improving its performance. The increase in performance is attributed to the fact that now the transmitter will always transmit for the amount of time specified by the timer circuit. Previously, the transmitter transmitted for this amount of time, *less* the time the receiver was receiving data for, as shown in Figure 5. In our testing we noticed that even when the base station transmitted for a constant amount of time, the amount of time the receiver-decoder's output pins remained valid varied significantly, which we attribute to the timing within the receiver-decoder chip. As a result, when the receiver's output pins were valid for longer than expected, the transmitter did not have as long as it needed to respond to the base station, as shown in Figure 6. With our improved design, the response from sensor locations is much more consistent, as shown in Figure 7.

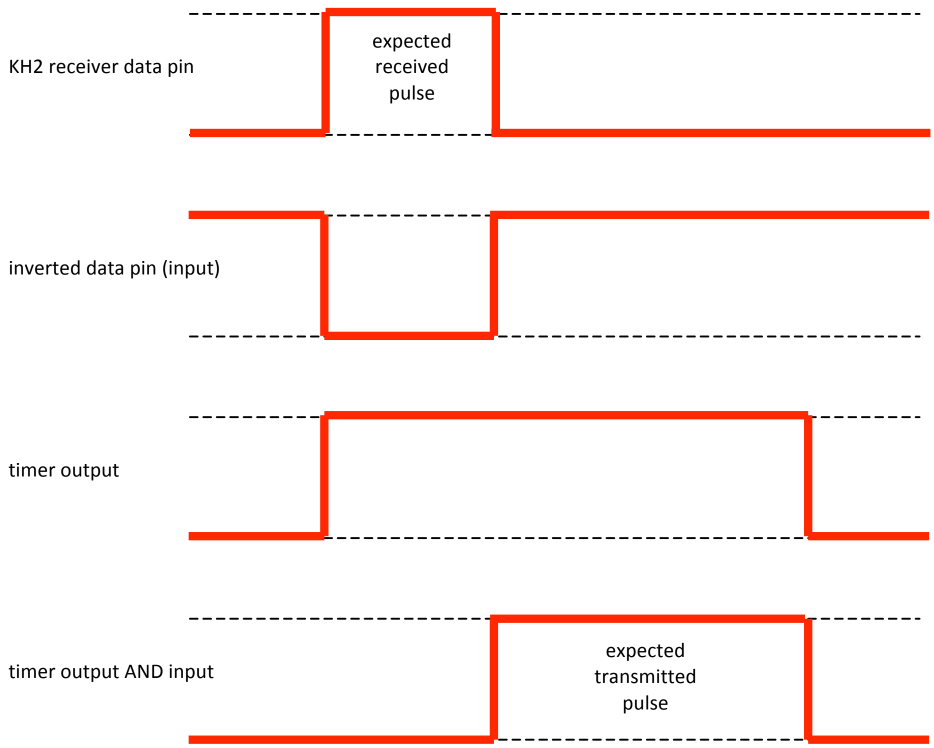

Figure 5: Originally expected pulses with original circuit design.

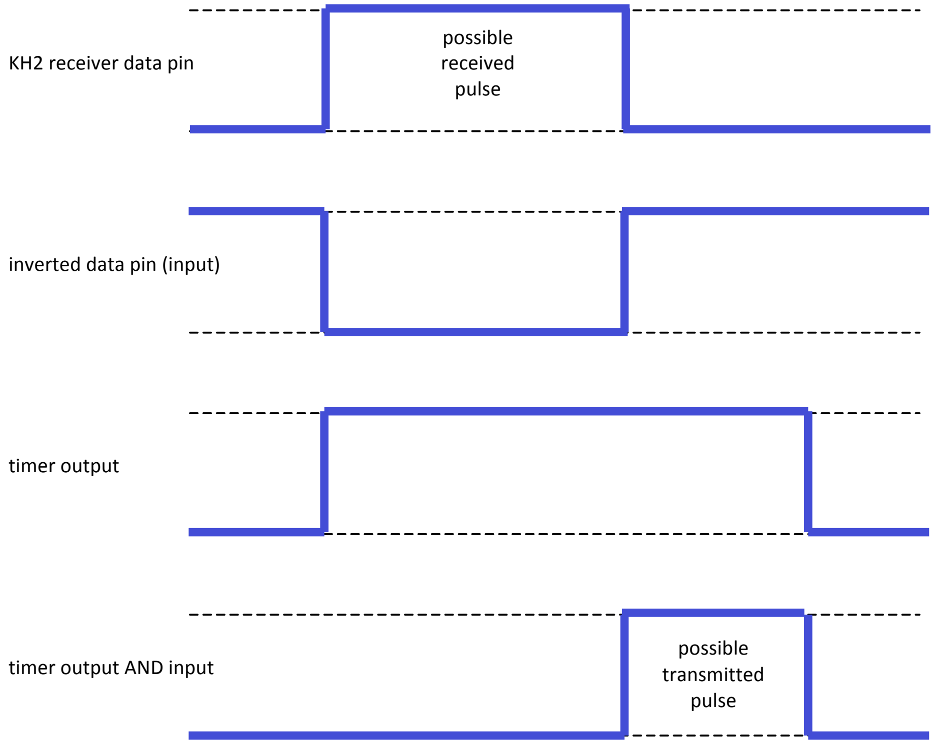

Figure 6: Pulses sometimes observed with original circuit design: the transmitted pulse is no longer long enough.

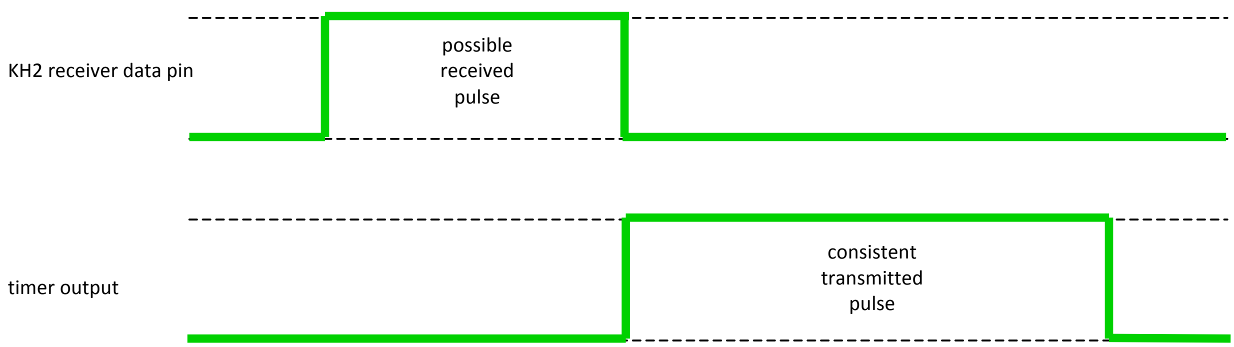

Figure 7: Pulses consistently observed with modified circuit design: the transmitted pulse is a constant length regardless of the received pulse.

Secondly, we had intended to power down the majority of the circuit for the majority of the time, and leave only the receiver turned on. Our intent was that when the receiver got a message intended for the sensor, it would activate the rest of the circuit. However, this goal proved to be more difficult than expected, because a large portion of the circuit had to be activated to process the originally received pulse correctly. When this problem was presented to Jason Lee, he suggested that our current consumption of 15 mA was quite acceptable for prototyping purposes, and so we did not make further attempts in this regard.

Also pertaining to the power of the circuit, we added a voltage regulator to the system. We had originally planned to run the circuit using two AA batteries, which would provide 3 V to the system. However, we discovered that when the batteries slightly degraded, they no longer provided enough voltage for the receiver to operate correctly. Therefore we used four AA batteries, and their 6 V was stepped down to 3.5 V using a voltage regulator. We found that this change made the system far less sensitive to decreases in battery voltage.

In our design specification, we had indicated that we would use the valid transmission line from the receiver, and propagate it through our circuit to the transmit enable line at the transmitter. However, we realized that we had designed our communication codes such that the first data line, D0, was always 1. As a result, it was redundant to also use the valid transmission line. By eliminating the use of the valid transmission signal, we were able to eliminate a third of the necessary chips from the timing delay portion of the circuit. (Although we only reduced the number of data lines from 9 to 8, each chip was able to service 4 data lines, meaning we only needed 2 of each chip rather than 3.)

3.8 Sensor Modules

The sensor modules worked very well with the initial design as proposed in the design specification, and little to no changes were applied to the design.

3.9 Lock Actuator

The lock-unlock actuator was largely implemented as outlined in the design specifications. One change was the addition of a level shifter, which was necessary because while most of our circuit was running at 3.5 V, the lock required 6 V to operate. We also needed to add a second set of four batteries exclusively for locking or unlocking the door. We discovered that when the door lock was actuated, it drew so much power from the batteries that there was not enough power to activate the remainder of the circuit, and thus the sensor location was unable to respond properly to the base station. Ideally, this second set of batteries would have been the four batteries already included with the keypad door lock. However, it would have been difficult to connect wires to the existing battery case, and so we felt that using another set was the best solution for the prototype.

The other change was that we originally intended the sensor circuit to only respond to the base station once it determined (at the comparators) that it had received a query, lock or unlock request. However, we decided that the key factor was that the sensor location acknowledged it had received the base station's request. Therefore, we eliminated a comparator for the query code, as well as an OR gate. The sensor location now always sends back a response, regardless of the message type. If the message was a query, the return message will be the sensor's status. However, if it is a lock or unlock request, the return message will simply be an acknowledgement of the received message.

Based on the contents of the message, the base station will be able to determine if the sensor location correctly interpreted a lock or unlock request.

3.10 Setup and User Interface

The user interface implementation deviated from the design specification due to changes in the base station software and server program. Because the list of sensors is no longer stored on the base station and is only obtained when the user sends a query, and the sensor slots are always a fixed number, adding and removing sensors became unnecessary. Instead, the user will always have a full list of available sensor slots, and the user will mark active sensor number designations as available by placing a checkmark on the corresponding checkbox.

Additionally, the number of sensor slots was changed from 10 to 5 in the prototype so the sensor list reflected total available sensors for the demo.

4 Future Plans

Chickadee Tech has devised some ideas for project expansion and future work. These ideas are presented in this section.

4.1 Full System

For the production level full system, we would like to increase the number of locations in the home that may be monitored. To do this, we will add other types of sensors, such as Hall Effect proximity sensors, temperature sensors, water sensors, and power usage sensors. To add in some of these sensors in a useful fashion, we also plan on adding functionality to the system so that the user may be alerted if an event occurs (such as flooding). Adding this functionality will help ensure peace-of-mind for the user as well as decrease potential damage that may occur from accidents in the home.

4.2 Cell Phone – Server Communication Protocol

For the production level, we will phase out the current use of Twitter by including functionality in the server to send and receive text messages directly. This will also mean that the user does not need to indicate their service provider. Finally, since the server will now be able to send reply text messages with only the knowledge of the incoming cell phone number and will not require user configuration information to do so, we will add the incorrect password error message back into the system.

4.3 Server

4.3.1 Hardware

In the production level system, rather than using an existing company laptop to run the server, we will purchase a dedicated server machine optimized for running continuously, and it will be connected to a phone line to allow for sending and receiving text messages directly.

4.3.2 Software

Once a dedicated phone line is connected, the server software will be modified to send and receive text messages directly rather than connecting via email. This will require some new code to be written, but it will also allow for the removal of the code that handles sending and receiving emails.

The server software will also need to be modified so that the user may be alerted if an emergency such as fire or flooding occurs in their home, or if a low battery state occurs in one or more of their sensors. This will require adding functionality that allows the base station to contact the server if these events occur, as well as new message formats for these alerts.

4.4 Server/Base Station Communication Protocol

To help minimize bandwidth usage, the actual format of the messages between the server and base station will be revised for the production level. Currently, the message consists of 1 byte to indicate message type, 2 bytes to indicate length, and 1 byte per sensor listed. However, the sensor list, for example, can be revised to use each bit to represent whether a particular sensor is connected or not, allowing up to 8 sensors per byte. Through simplifications like this, the message size can be reduced by a significant fraction.

4.5 Base Station

4.5.1 Hardware

For the production level hardware, we will use only the microcontroller itself with accompanying circuitry on a Watchbird™ base station board rather than the development board. To accomplish this, a new board for the base station will need to be designed.

4.5.2 Software

For the production level, the base station software will need to be modified to support the encrypted data transfer to both the server and the lock actuator unit.

Additionally, some of the server's functionality (such as the setup web interface) may be moved to the base station to reduce load on the server. The base station should be able to support this functionality.

Finally, the base station software will need to be modified to support the alert messages discussed above. For these messages, the base station will need to receive the alert from the sensor in question and contact the server to notify the user.

4.6 RF Transceivers

We still plan to use the Linx LR transmitters and receivers at the base station, and the Linx KH2 transmitters and receivers at the sensor locations. However, we would like to take full advantage of the range offered by these modules. In order to do so, we would place the RC power supply filters directly next to the power connections on the module, which we did not do in our original PCB design. We anticipate that this change would make the received data considerably cleaner, and therefore improve the reception range.

In door lock locations, we plan to use a microcontroller to improve data transmission security. As a result, the encoder and decoder capabilities of the KH2 series chips would no longer be necessary, and we could switch to the cheaper and smaller LR chips that we are currently using at the base station.

4.7 RF Transceiver/Sensor/Actuator Circuitry

As discussed in Section 3.7, the number of chips used in the sensor circuitry has already been decreased from our original design. For a production level model, we would plan to use surface mount components for the entire circuit, and we would have all components on a PCB. Many surface mount components are considerably smaller than the components we used during prototyping, and a PCB would also make it more feasible to place components closer together. As a result, we are confident that our production level design would be significantly smaller than the prototype.

We also hope to improve the power consumption of our production level model, increasing the time between battery changes from approximately 5 days to a couple of months. This rather drastic improvement would be possible if most of the circuit was powered down for the vast majority of the time. We understand that it is even possible to power down the receiver by using the received signal strength indicator (RSSI). The receiver continues to monitor received data in a low power state, which means its RSSI pin will continue to be active. Additional circuitry could be designed such that when the RSSI pin passed a certain threshold (indicative of a valid transmission), the receiver and the rest of the circuit would turn on.

We would also implement a battery monitoring feature in the sensor end circuitry, so that the user would be notified when the batteries in a unit were almost dead. This improvement would mean a user would not encounter "sensor nonresponsive" replies when the batteries on a unit died. Instead, they would be able to act ahead of time, and thus maintain constant sensor functionality.

4.8 Sensor Modules

The types of sensor design could be expanded to accommodate different needs that a press switch could not accommodate in a household:

- Proximity Hall Effect sensor to detect object distance where actual contact is not possible;
- Temperature sensor to determine if room temperature is above or below desired level;
- Water sensor to determine water level in a water tank, or flooding or leakage from a roof or tank;
- Power sensor to determine if a digitally operated appliance is consuming above desired power level.

In all above cases, a contact switch cannot monitor the situation and the new sensor designs are needed to accomplish the tasks. In order to achieve this improved flexibility, the sensor circuitry could adopt sockets to switch between sensors so users could easily buy a new type of sensor and use it right away after the new sensor is plugged in.

4.9 Lock Actuator

For the lock actuator, our primary future goal is to consolidate its power supply into a single set of four AA batteries. A production level model would integrate the keypad lock circuitry with the RF circuitry we designed, which would make it more practical for all power to come from one set of batteries. Furthermore, we suspect that further voltage regulation, and/or modifications to the timing circuitry would prevent the high power required by the door lock from interfering with the transmitter's operation.

We would also like to improve the security of data transfer to the door lock. While this is not a priority at simple sensor locations (where the information about one's oven is not of particular interest to thieves, and the status of a window being open is fairly obvious to someone who can see the house), security at a door lock is critical. Therefore we would investigate the use of a microcontroller at the lock actuator locations to implement some type of encrypted wireless data transfer. This microcontroller would also be used for the keypad functions, so two microcontrollers would not be necessary.

Lastly, although not related to a lock actuator directly, we would like to pursue the implementation of actuators in other locations. There would be a great increase in utility if we could design unobtrusive actuators to perform tasks such as closing windows, turning off ovens, or adjusting thermostats.

4.10 Setup and User Interface

There are many possible areas where the user interface could be expanded. Secure data transfer and data encryption are definitely essential elements in the production level user interface design. Secure Sockets Layer must be applied to user log in and user registration menus, as well as the Watchbird™ configuration menu. The website will also need to encrypt all the data stored on SQL database so if the database was breached, the data could not be deciphered easily.

In addition to security enhancements, an internet query implementation has also been considered for user convenience. The current system design only allows the user to query the Watchbird™ system through cell phone text messages, which are usually charged per text message. This cost could become a burden to the user if the system is queried frequently. To offer users an alternative method to query their system, we will implement the ability to query the Watchbird™ system through the user interface. With this functionality, the internet becomes another option to obtain the status of or issue commands to the Watchbird™ system.

5 Budget and Timeline Overview

5.1 Budget

Our budget overview is presented in Table 2.

Table 2: Budget Overview

Item	Estimate Cost	Actual Cost	Amount Over Budget
Embedded PC or Microcontroller	$200	$213.68	$13.68
PC/Microcontroller Components (RAM, Power Supply/Programmer)	$100	$238.27	$138.27
Display	$60	$0.00	$0.00
Keypad	$4	$0.00	-$4.00
RF Transmitters/Receivers	$11	$303.64	$292.64
RF/Keypad Door Lock	$150	$110.88	-$39.12
Sensor End/Door Lock Circuitry, PCBs	$0	$269.47	$269.47
Sensors	$30	$12.60	-$17.40
Integration Components	$100	$101.64	$1.64
Cases	$0	$94.77	$94.77
Miscellaneous Costs (sockets, protoboard, cables, battery cases etc.)	$100	$21.16	-$78.84
Demo Materials (mounting materials, door, window, batteries)	$0	$108.93	$108.93
Total Costs	**$755**	**$1,475.04**	**$720.04**

It is evident that we were significantly over budget in the development of the Watchbird™ prototype. This was mostly due to the fact that we had not decided on a final implementation at the time that the budget was estimated. We ended up spending more money than expected on a microcontroller programmer (nearly $200), and much more than expected on the RF and sensor end circuitry. Additionally, a few of our receivers and transmitters were damaged in the course of development and needed to be replaced. Finally, we did not budget for professional cases, and this as well added significantly to our actual cost.

However, we did secure $700 in funding from the ESSEF, which made the remaining costs acceptable when split between the four group members.

5.2 Timeline Overview

Our timeline overview is presented in Figure 8.

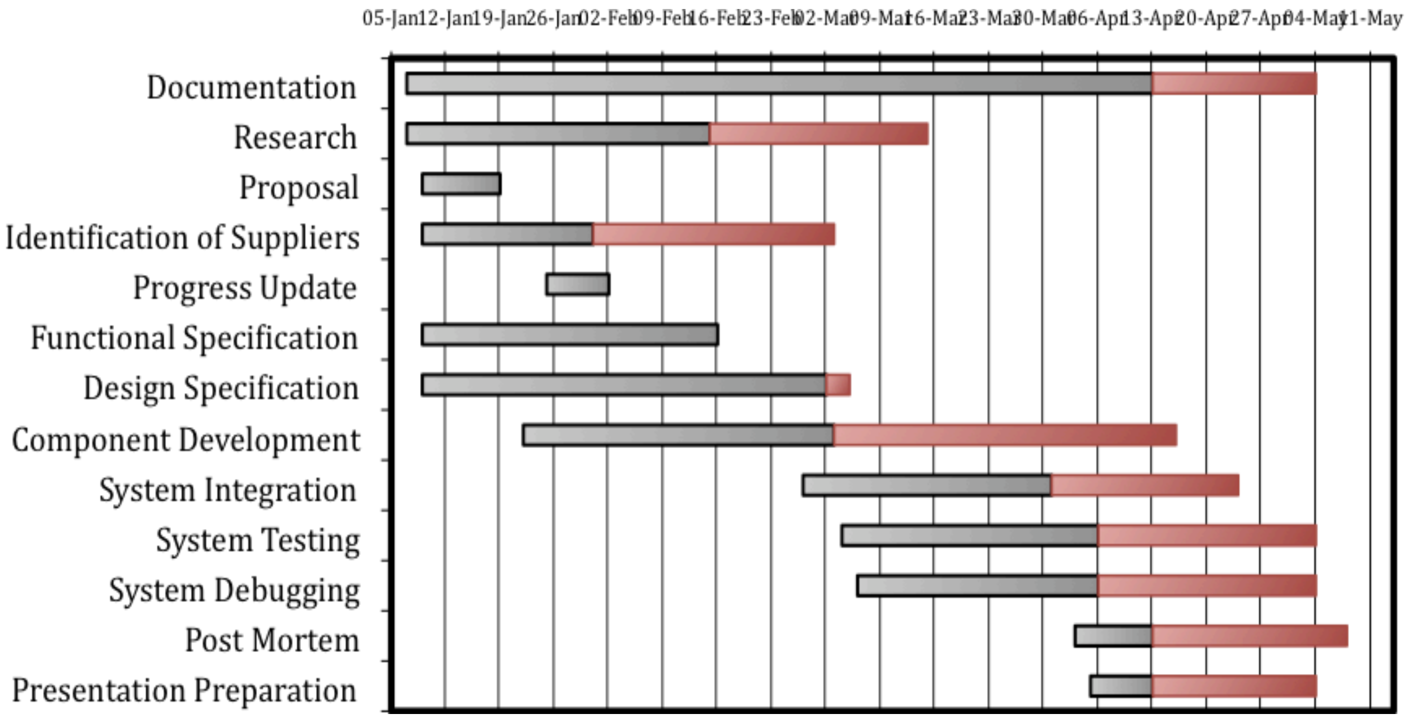

Figure 8: Timeline Overview

The red bars indicate the amount that each component went overtime. There were several reasons for the development being behind schedule.

Firstly, we spent a significant amount of time at the beginning of the semester trying to decide the best way to implement the product. We considered microcontrollers vs. embedded PCs, ZigBee vs. Bluetooth vs. simple RF transmission, microcontrollers at all sensor locations, and many other options. This put us behind schedule for the remainder of development.

Additionally, we encountered many issues with the RF section which required significant troubleshooting, which again put us behind schedule. We integrated, debugged, and tested the system as development progressed, however, so development was only slowed down by this problem rather than completely held up.

6 Group Reflections

6.1 Samantha Grist

I feel that carrying out this project has allowed me to grow immensely as both an engineer and a leader.

As an individual engineer, I was very lucky to have been able to work on several, diverse areas of the system. I really enjoyed integrating the microcontroller software with the RF chips, as well as designing and fabricating circuits and helping to integrate the door lock actuator into our system. The project gave me the opportunity to implement some low-level software, which I found very interesting because it blurred the line between what I had previously considered software and hardware; I had previously had very little experience with using software to control individual pins at such a low level. I also gained experience in designing PCBs and ordering parts, which really emphasized good planning and double-checking. Finally, fabricating and debugging circuits allowed me to really take pride in my work, do it carefully (but quickly!), and learn the steps necessary to fix things if something went wrong. By the end of the semester, I felt that I had gained a significant amount of technical knowledge and problem solving skills. Furthermore, I really came to appreciate the individual talents that each of my other group members brought to the project, and learned from them as well.

As a leader, this project was very interesting for me as it was the first time I had acted as a leader in a project. I am generally a quiet or shy individual in groups, so this was really a big step for me. It was eased by the fact that we rotated the chair of each meeting, and all at some time contributed some sort of leadership to the group, but I really feel that it helped me grow as a leader. One problem that I encountered stemmed from the fact that I sometimes have very high standards with regards to my work, and at times during the semester, I found that I had to catch myself before I tried to force these standards onto my group members. I realized that there needs to be a compromise between quality and time and so tried not to force my personal standards onto others, but at times it was a little bit frustrating for me when others did not seem to take as much pride in their work as I try to (and I'm sure others probably felt a similar frustration towards me!). I really learned that when working in groups it is necessary to compromise between what each member feels is acceptable quality and just try to work together to get things done.

On the whole, however, I think that we divided up the work well and worked effectively as a team. We started off by each taking a part of the project, and each contributed wherever necessary after our parts had been completed. I think that the contribution to parts of the project which we had not originally chosen was very important in this time-limited project, and allowed us to work together and solve problems as quickly as possible as they appeared.

Above everything else, I really believe that each of us has contributed a lot to this project, and has accomplished something incredible. I am amazed at both what each group member has completed and at what we were able to achieve by working together.

6.2 Eric Shen

This project exposed me to various area of an engineering project and I learned a lot in the last 4 months. Since the first week of the semester we have been meeting frequently to discuss the project designs and documentations. I found splitting up the sections of the specifications and then compiling and editing by each group member is extremely efficient. All of us not only gives finishing touch on the document but also learn more about each other's design through editing. The weekly group meeting with rotating chairman is a refreshing experience for me because I experienced holding a meeting for the first time. It forced me out of the role of a follower and puts me in charge, and rotating chairman also allows me to observe different meeting formats from different person each week.

When the group first split the design into sections, I chose 2 sections, a section that I had some experience with, and a section that I have never done before. I was able to quickly come up with a sensor design that is easy to use, easy to implement, flexible and low cost within a short period of time. The initial design was tested and worked very well and I was able to deliver a component that can be attached to the overall sensor end circuitry easily. However, my process of designing the user interface and the data storage could be described as venturing into the unknown. I had a general idea of how I wanted to approach the problem, but I was missing the knowledge to carry out the design. Had I not stumbled into a friend who suggested me to use ASP.NET, I would not have been able to build the user interface hosted on a server. I not only gained valuable knowledge on building dynamic website, but also on database as well. Because I was able to finish my tasks early, I was able to help out with the making of the 3rd sensor end circuit and the making the demo materials. I enjoyed the hands on long hour doing soldering and drilling in machine shop; it was a lot more fun than long hours of debugging staring at the computer screen.

For group dynamics, we were able to communicate efficiently through emails and msn messenger. Meetings were held regularly and every attended. Each of us contributed to the project and completed individual assigned parts. In short, we were able to complete the project as a team.

Starting a project of this scale and building from the ground up is a huge amount of work that can barely squeezed into a 4 month period. If I am to do a similar project in a course like ENSC 440, I would want to make sure the system functions are defined and system designs are ready, as well as some actual circuit built and working.

6.3 Derek Lowes

This semester has been an excellent experience for me. Developing a project of this scale from bare wisps of ideas to a working prototype has been a lot of stress at times, but I have definitely learned a lot. I gained experience in areas I'd never worked with before, like TCP/IP communication protocols, or IMAP and SMTP protocols for working with emails.

If I were to do this again, the main thing I would do differently is to take into account just how much time and effort something like this takes. For example, try to be decisive with design decisions to maximize the time for actually putting the project together, and when scheduling, no matter how much contingency time you allow for, it probably isn't enough.

As to group dynamics, I feel our group worked very well together. For the documentation, we divided it into individual parts, did our write-ups, and then put it together and did multiple rounds of editing. I feel this did an excellent job of getting the initial copy of the document together promptly, and then having plenty of time to polish it up into a good finished product. As to the project itself, we again divided it into sections and worked on it separately at first, and then started working together more and more as the various sections began to merge.

One thing I think we did well was by basically treating the connections between the different sections as components themselves. For example, while developing the server and base station, we were simultaneously developing the method for the server and base station to communicate with each other. This made integration an ongoing process right from the beginning, and helped us avoid having to revamp any sections due to incompatibilities.

All in all, while this was a big project, and took a lot of time and effort, it was an excellent learning experience, and the sense of accomplishment after completing something this major is great. I learned a lot, and it was definitely worth doing.

6.4 Andra St. Quintin

From a technical perspective, I have learned a great deal in the process of designing and implementing the sensor-end circuitry of our devices. With regards to radio-frequency (RF) communications, I learned about the intricacies and details this technology requires: a clean power supply, care in antenna choice and positioning, static-safe handing of components, and precision in timing the signals. Designing the printed circuit board for the RF devices familiarized me with the steps required to go from an idea to a software depiction to a physical realization. Although I had been introduced to PCB software in ENSC 204, the need to produce a physically correct board without having to pay for a second attempt was a valuable experience! Laying out the circuit on the prototyping board taught me the importance of careful planning and double-checking; it certainly was better to measure twice and cut (or solder) once, rather than trying to discover and correct a problem later. I also significantly improved my soldering skills and my circuit debugging skills throughout the implementation process. Troubleshooting circuit problems at the end of the semester was definitely easier than at the beginning!

In terms of group dynamics, I experienced and learned from the different approaches people have to completing a project. What may seem to be a perfectly acceptable job to one person may appear to be lacking is some regard by another. Similarly, something that might seem like a waste of time in the pursuit of perfection by one person may seem to be an essential component to another. Having experienced both sides of this situation throughout the semester, I have come to realize that there cannot be one "right" opinion in this regard. In a group situation, I found that strategies for dealing with these differences included focusing on the goals that we ultimately intend to achieve; objectively assessing how the work at hand did or did not accomplish to these goals; delivering and receiving criticism in a fair, calm, and professional manner; helping each other out to minimize individual weaknesses and maximize individual strengths; and respecting the fact that each members had different opinions on the details of implementing our project.

I also learned that while it is often difficult, it is important to find the right balance between the two important adages of "divide and conquer" and "two heads are better than one." In a large project like this with multiple components, it was sensible to make effective use of time by having each person working on something different, and at times I wished it was possible to take further advantage of this division of labour. But at the same time, it was valuable to have multiple thoughts on how to implement an idea, or how to troubleshoot a problem.

Throughout this project, I came to fully realize the importance of self-initiative on behalf of all group members. It was immensely helpful when someone was able to say "I think we need to get task 'X' done, and I am going to tackle it," rather than saying "What needs doing next?" As I was not directly involved in the software side of our system, all I really saw was that the software worked very well. So even though I did not witness it, I was very impressed with the amount of self-initiative, attention to detail, coordination, and troubleshooting that it took for the system to run flawlessly.

All in all, I gained valuable technical knowledge and important group work experience. I am proud of what I learned as an individual, and proud of what we were able to accomplish as a team.

7 Conclusion

The Watchbird™ system was a complex and challenging project. Over the past 4 months we developed a viable, low cost, text message based home-monitoring system. We are confident it that can be used for a wide variety of home-monitoring applications, and that it is a marketable product that will provide peace-of-mind for users every day.