Project Proposal and Feasibility Study

Silence is Golden

Team 6

Matthew Cosnek
Nick Ellens
Jared Heys
Ryan Smith

ENGR 339 Senior Design Project

December 9, 2005
Abstract

The purpose of this design project is to design a system for silencing cellular phones within the physical vicinity of the device. This system shall be constructed in two parts: a client to be run on the cellular phone and a device to activate/deactivate the client within a given proximity. The client shall be responsible for controlling the ringer of the phone. The connection between the phone and base device shall be via Bluetooth™ technology. The device shall be designed to be placed in an entrance/exit. The first pass through shall cause the client to deactivate the ringer of the phone. The second pass through shall cause the client to return the phone settings back to their previous state. This project was determined to be feasible from the standpoint of both hardware and software design.
Table of Contents

1. Introduction .................................................................................................................. 1
2. Objectives ..................................................................................................................... 2
  2.1. Base Station .......................................................................................................... 2
      2.1.1. Hardware .................................................................................................. 2
      2.1.2. Software ................................................................................................. 2
  2.2. Client Software ..................................................................................................... 2
3. Design Norms .............................................................................................................. 3
4. Research ....................................................................................................................... 3
  4.1. Similar Products ..................................................................................................... 3
  4.2. Base Device .......................................................................................................... 4
  4.3. Mobile Phone ......................................................................................................... 6
      4.3.1. Cellular Device ......................................................................................... 6
      4.3.2. SDK Research ......................................................................................... 7
  4.4. Connection Method ............................................................................................... 7
5. Project Management .................................................................................................... 9
  5.1. External Organization ............................................................................................ 9
  5.2. Internal Organization ............................................................................................. 10
  5.3. Project Timeline .................................................................................................... 10
      5.3.1. Prototype development ............................................................................ 11
      5.3.2. Production design development ............................................................... 11
6. Preliminary Feasibility Analysis ................................................................................ 11
  6.1. Base Station .......................................................................................................... 11
      6.1.1. Hardware ................................................................................................. 11
      6.1.2. Software .................................................................................................. 12
  6.2. Client Software ..................................................................................................... 12
      6.2.1. Hardware Control .................................................................................... 12
      6.2.2. Base Recognition ..................................................................................... 12
  6.3. Cost Estimates ........................................................................................................ 12
7. Project Requirements .................................................................................................. 13
  7.1. Base Station .......................................................................................................... 13
      7.1.1. Base Hardware ......................................................................................... 13
      7.1.2. Base Software .......................................................................................... 13
  7.2. Client Software ..................................................................................................... 13
8. System Design ............................................................................................................. 14
  8.1. Base Device .......................................................................................................... 16
      8.1.1. Hardware ................................................................................................. 16
      8.1.2. Software .................................................................................................. 19
  8.2. Client Software ..................................................................................................... 19
9. Conclusion .................................................................................................................... 20
10. References .................................................................................................................. 21
11. Appendices ................................................................................................................. 23
    11.1. Appendix A ...................................................................................................... 23
    11.2. Appendix B ...................................................................................................... 24
11.3. Appendix C

................................................................. 25
Table of Tables

Table 1 Decision matrix for Bluetooth™ IC ................................................................. 5
Table 2 List of possible cellular phones [10] [11] [12] .............................................. 7
Table 3 Connection Method Decision Matrix ......................................................... 8
Table 4 Task List .................................................................................................... 23
Table 5 Preliminary Budget .................................................................................. 24
Table 6 Bill of Materials ...................................................................................... 25
Table of Figures

Figure 1 Simple block diagram of system communication ............................................... 1
Figure 2 The Design Team: Nick Ellens, Jared Heys, Matt Cosnek, and Ryan Smith ..... 1
Figure 3 Client-Base software interaction ........................................................................ 15
Figure 4 Piconet interference ............................................................................................ 16
Figure 5 Hardware Block Diagram ................................................................................... 16
Figure 6 BlueCore2 External block diagram ................................................................. 17
Figure 7 Base station software program flow ................................................................. 19
Figure 8 Client software program flow ............................................................................. 20
## Table of Terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ACL</td>
<td>Asynchronous Connectionless</td>
</tr>
<tr>
<td>2. ACO</td>
<td>Asynchronous Connection-Oriented</td>
</tr>
<tr>
<td>3. Balun</td>
<td>A transformer used to convert an unbalanced signal to a balanced signal</td>
</tr>
<tr>
<td>4. Base Station</td>
<td>Device used to initiate connection with Client Phone via Bluetooth™</td>
</tr>
<tr>
<td>5. BGA</td>
<td>Ball Grid Array</td>
</tr>
<tr>
<td>6. Bluetooth™ Technology</td>
<td>Wireless communication standard for secure short range connections</td>
</tr>
<tr>
<td>7. BOM</td>
<td>Bill of Materials</td>
</tr>
<tr>
<td>8. CAD</td>
<td>Computer Aided Design</td>
</tr>
<tr>
<td>9. Client phone</td>
<td>Phone connected to Base Station</td>
</tr>
<tr>
<td>10. Client program</td>
<td>Program that will run on the Client Phone</td>
</tr>
<tr>
<td>11. CLDC</td>
<td>Connected Limited Device Configuration</td>
</tr>
<tr>
<td>12. C++</td>
<td>Programming language</td>
</tr>
<tr>
<td>13. HCI</td>
<td>Host Controller Interface</td>
</tr>
<tr>
<td>14. HID</td>
<td>Human Interface Device</td>
</tr>
<tr>
<td>15. IC</td>
<td>Integrated Circuit</td>
</tr>
<tr>
<td>16. I/O</td>
<td>Input and Output</td>
</tr>
<tr>
<td>17. IR</td>
<td>Infrared</td>
</tr>
<tr>
<td>18. LED</td>
<td>Light Emitting Diode</td>
</tr>
<tr>
<td>19. LGA</td>
<td>Land Grid Array</td>
</tr>
<tr>
<td>20. PCB</td>
<td>Printed Circuit Board</td>
</tr>
<tr>
<td>21. Piconet</td>
<td>A localized wireless network</td>
</tr>
<tr>
<td>22. JM2E</td>
<td>Java 2 Platform, Micro Edition</td>
</tr>
<tr>
<td>24. Python</td>
<td>Programming language similar to C++</td>
</tr>
<tr>
<td>25. RAM</td>
<td>Random Access Memory</td>
</tr>
<tr>
<td>26. RF</td>
<td>Radio Frequency</td>
</tr>
<tr>
<td>27. RFCOMM</td>
<td>Radio Frequency Communication</td>
</tr>
<tr>
<td>28. RISC</td>
<td>Reduced Instruction Set Computer</td>
</tr>
<tr>
<td>29. SCO</td>
<td>Synchronous Connection-Oriented</td>
</tr>
<tr>
<td>30. SDK</td>
<td>Software Development Kit</td>
</tr>
<tr>
<td>31. SPI</td>
<td>Serial Port Interface</td>
</tr>
<tr>
<td>32. UART</td>
<td>Universal Asynchronous Receiver Transmitter</td>
</tr>
<tr>
<td>33. USB</td>
<td>Universal Serial Bus</td>
</tr>
<tr>
<td>34. VM</td>
<td>Virtual Machine</td>
</tr>
</tbody>
</table>
1. Introduction

Every year cellular phones are becoming more popular, and millions of people are dependent on them every day. As cell phone usage increases, the number of cell phones that ring at inappropriate times is also increasing. Some of the major areas that are experiencing this growing problem are churches, movie theaters, and schools. To combat this issue movie theaters have messages on the screen reminding people to silence their cell phones. Churches display reminders on overhead screens or in bulletins, and schools usually tell the students about cell phone rules once and it is up to the students to remember to silence their phone. These reminders are somewhat helpful, but many people still forget to silence their cellular phone. This design project tries to address this issue by automatically controlling the phone ringing features when entering an area where loud cell phone rings are inappropriate.

The project design will consist of two parts: a client program to be run on the cellular phone and a base device to activate and deactivate the client within a given proximity. Figure 1 shows how the two main parts of the system interact. The client program is responsible for controlling the ring settings of the phone. The connection between the phone and the base device will use Bluetooth™ technology. The base device will be designed to be placed near an entrance/exit to an area where cell phones should be silenced. The first pass by the base device will activate the client program, and when passing by again, the base device will signal the client program to return the phone settings back to their previous state.

The design team for this project is shown in Figure 2. The team consists of four engineering students at Calvin College, and they are in the electrical and computer concentration. Matt Cosnek is from

![Figure 2 The Design Team: Nick Ellens, Jared Heys, Matt Cosnek, and Ryan Smith]
Aliquippa, PA, and he is interested in computer hardware and software. Nick Ellens is from Plymouth, MI and he is interested in computer design and automobiles. Jared Heys is from Grand Rapids, MI and he is interested in analog and digital design. Ryan Smith is from Madison, AL, and he is interested in computer hardware and design.

Calvin College is a liberal arts college located in Grand Rapids, Michigan with about 4000 students currently enrolled. The engineering department consists of four concentrations. These concentrations are civil, mechanical, electrical and computer, and chemical. The senior year for every engineering student contains a year long capstone course called Senior Design. This course is made up of ENGR 339 and ENGR 340. The two parts of this course are each a semester long. The first semester course is ENGR 339 in which a project proposal and feasibility study is completed. During this semester initial research is done on the project to determine the requirements and feasibility of the project. The second semester course is ENGR 340 in which the major goal is an implementation of a prototype. This includes the designing, constructing and testing of a working prototype of the product along with providing documentations and giving formal presentations about the project.

The Project Proposal and Feasibility Study (PPFS) will present the initial research the team has completed along with a description of the feasibility of the project. A list of terms and their definitions that are commonly used throughout the PPFS is given. The objectives for each of the main sections of the project are specified. The Christian aspect of the design processes is given in the design norms section. The results of preliminary research into the major sections and components are presented. The approach to how the team is managing the project to ensure its completion on time is specified in the project management section. The feasibility of each of the major parts of the project is presented. Requirements for the entire project are specified. The preliminary design and methods for designing are given in the design sections. The appendix includes, research materials, project task list, FCC guidelines, Bill of materials, and a preliminary budget.

2. Objectives

The main objective of this project is to design a product and produce a working prototype that implements the major aspects of the full design of a system that will control the ring functions of a cellular phone. This system will consist of a base device that will transmit a signal to cellular phones within range. This signal will query the current settings of the phone and if needed it will activate a preinstalled client program on the phone which will save the current settings and set the phone to silent mode. A second pass near the base station will restore the original setting of the phone.

2.1. Base Station

The base station is comprised of two components: hardware and software. Each component has objectives that will allow the base station to operate properly.

2.1.1. Hardware

The objective for the base device hardware is to transmit a signal based on the software that is written for the base station. The signal will be transmitted using Bluetooth™ technology to a nearby cellular phone. An antenna will be selected and purchased to transmit this signal.

2.1.2. Software

The base device has a software component in which the objective is to provide the proper code to run the Bluetooth™ integrated circuit (IC) and to provide the IC with the code to send the correct signal to the cellular phone. The base software needs to automatically recognize a cellular phone that is in its area that also has the corresponding client software installed on it. It should not affect Bluetooth™ devices that do not have the client program installed on them.

2.2. Client Software

The first objective of the client software is to be able to respond to inputs from the Bluetooth™ features of a cellular phone. Another objective is for the client program to be able to respond to and connect to the base device. The client software needs to manage the settings of the phone and know its current settings. The settings need to be saved by the program and recalled when the program is activated.
again. The program also needs to keep track of its status. This will allow the program to execute the
correct operation when it is activated by the base station. The last objective of the client program is to
manage the phone if it is hovering around or in the area affected by the base station.

3. Design Norms

During the design process, it is important to take into account a set of design norms as guidelines for
the development of a product that will better meet the needs of the user. To do this, a product must be
designed with cultural appropriateness, transparency, stewardship, integrity, justice, caring, and trust in
mind.

When designing a product for use by businesses, the cultural appropriateness of the final design must
be considered. The cultural appropriateness is a consideration to the social environment in which the
product will be used. A product that does not fit with the culture in which it operates will not reach
maximum potential in operation or market.

Transparency, or the ability to fully understand the working of a device, is an important issue to take
into account when one considers the possibility of a product’s future development. To this end, detailed
documentation of all parts of this project shall be generated. These documents will include technical data
so that a trained professional can understand every aspect of our design. In addition, a simple user’s
manual shall be created to aid in understanding the product operation and any necessary set-up
procedures.

In designing this product, being good stewards of resources will keep the product cost low. By
making a design comprised of relatively easy to obtain, “off the shelf” materials, it is hoped that the
amount and cost of resources will be minimized.

The design norm of integrity shall be incorporated into the project by considering the form and
function of the device being built. There is usually no need to make unnecessary aesthetic changes to a
product that is designed to blend into its environment. The product to be designed will aid in the
propagation of human values and relationships by preventing explosive situations that can be caused by
the inappropriate use of a cellular phone ringer in inappropriate venues. The simplicity of use of the
design shall also aid in upholding integrity in the eyes of the user.

Due to the system automatically controlling the cellular phone of the user, the issue of privacy, and
therefore justice to the user, must be taken into consideration. Some individuals are apprehensive about
outside devices controlling their personal possessions, like a cellular phone. It is not the intent of the
design to infringe upon the rights or privacies of the user. Therefore, the client program will be used at
the will of the user.

Closely related to the issue of justice is trust. If a product is to be used willingly, the user must have
confidence in its performance and reliability. This necessitates designs that are robust and predictable.
This also necessitates documentation complete enough to ensure the user is able to gain an adequate
understanding of the operation and reliability of the product. The design team is committed to providing
such assurances.

A sensitive issue in today’s world is that of technological integration. As more technology pervades
the daily lives of the 21st century society, it becomes a scramble to keep up. Taking this into
consideration, the design shall target minimal user interaction. Through personal experience, the design
team has seen that the easier a design is to use, the more accepted it will be.

4. Research

There were several areas of research. First, similar products were researched to determine our
project’s feasibility and to ensure no copyrights were infringed. The two major components of the system
are the base device and the mobile phone. In order to make educated design decisions, research was done
for each component. The base device and the mobile phone need to connect to each other. Therefore,
various connection methods were researched.

4.1. Similar Products

The idea of silencing cellular phones in certain areas is not a new concept. In fact there are other
products that already exist or being developed that are similar to our idea. Two companies in particular
have developed, or are in the process of developing specific products to silence cellular phones in similar ways as our proposed solution.

Cell Block Technologies, Inc. has developed a product (patent-pending as of 2000) that achieves the desired goal, which is to silence cellular phones in a limited area [1]. The way that they have chosen to do this is by using “Quiet Cell” technology which they developed. This consists of a small device about the size of a smoke detector called the “Quiet Cell Control Unit.” This device is recognized by cellular phones at a radius of about 2 meters (about 100 square feet of floor space). Once the device is recognized, it tells the phone to change to a channel that is not active, which keeps the phone from sending or receiving information from its original base station (tower). It is also possible to get control units that cover a larger area, up to 10,000 square feet. This technology, however, is considered in the United States to be in the same category as cell phone jammers, which are illegal. Because of this, Cell Block Technologies are concentrating their efforts outside of the United States, in places where the technology is not considered illegal. This device is not available to small volume retail customers. Their primary market consists of prisons, religious institutions, and foreign embassies [1].

BlueLinx, a North Carolina-based corporation, is also developing a way to silence cellular phones in a confined area [2]. They do this through the use of Q-Zone technology. This consists of a small device that will communicate to cellular phones in the area via a Bluetooth™ connection. Since Bluetooth™ is not on most cellular phones in use today, the device is not commercially available, and only the general concepts are released to the public. The technology has two parts: the base station and the software on the phone. The base station sends out a Bluetooth™ signal to all Bluetooth™-enabled devices in the range. The range is not strictly specified, but they claim to be able to cover sizes from small conference rooms to a multiple screen movie theatre. This area is called the Quiet Zone. When a Bluetooth™ device enters this zone, the Q-Zone device initiates a brief communication with the Bluetooth™ device telling it to be quiet. The Bluetooth™ device will then shift to quiet mode, based on the type of device (i.e. PDA, cellular phone, etc.). For example, one phone might turn to vibrate mode, while another would just turn down the volume to a very low level. As long as the device is in the Quiet Zone, these settings will hold, but upon leaving the area, the original settings will be restored. The goal behind the Q-Zone technology is not to interrupt service to the phones, but to reduce or eliminate the distractions that could be caused by these devices. Since the communication is through Bluetooth™, only devices with this technology and the software can be controlled. The software is currently only available on a royalty licensing to handset manufacturers.

BlueLinx also holds a patent for “politeness zones for wireless communication devices.” [3] This patent seemed to be exactly what the Silence is Golden team was planning on implementing. However, upon closer inspection of the patent we noticed that the “politeness zone transmitter” was defining the politeness zone. Our idea is to make a device that acts similar to a gate; that is when a cellular device passes in proximity of our base device, the settings will be changed to silent until the same device passes through the proximity of our base device for a second time or a time limit has been reached.

4.2. Base Device

Research for the base device primarily consisted of researching a Bluetooth™ IC. Initially a goal of the project was to design as much of the circuit as possible, but research into the operation of Bluetooth™ and the components required implement Bluetooth™ found that it would be nearly impossible to design the circuit at this level. The operation of Bluetooth™ required many components that had to interact in a very precise way at high frequencies. Some of these components are a microcontroller interacting with a memory management system; code from memory has to be executed and sent to radio components that are operating at 2.5GHz. This information shifted the focus to designing the circuit using a Bluetooth™ IC that had most of the essential components included on IC. It was not difficult to find these types of ICs, but most of the manufacturers provided limited datasheets and information.

The BlueCore2 IC was found at a reasonable price of $54 for a quantity of five [4]. BlueCore3 and BlueCore4 ICs are also available at variable prices. The specifications concerning the decision between these ICs are whether the IC has the ability to run software without an external host, package size and price. CSR is the manufacturer of this IC, and their support website provides a lot of helpful information
The site provides in depth datasheets, application notes, example circuit designs, and much more. The team is continuing to research the BlueCore2 to decide whether it should be the IC to be purchased. So far this IC looks like it is the best choice because the datasheet clearly outlines four modes of operation that pertain to software. One mode allows software to be run entirely onboard without the need for an external host. This means that the BlueCore IC can operate as a processor to execute a program that is written by the design team [6]. This is a major reason for considering this IC over the other options.

Other ICs are also being researched to compare to the BlueCore2 IC. A decision matrix is shown in Table 1. The three Bluetooth™ modules selected are the BlueCore2 External, the LMX9820A from National Semiconductor, and the CXN1000 from Sony.

### Table 1 Decision matrix for Bluetooth™ IC

<table>
<thead>
<tr>
<th>Device</th>
<th>Weighting</th>
<th>Package</th>
<th>Features</th>
<th>Price</th>
<th>Availability</th>
<th>Documentation</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weighting</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BlueCore2 External</td>
<td>20</td>
<td>5</td>
<td>12</td>
<td>5</td>
<td>25</td>
<td>35</td>
<td>100</td>
</tr>
<tr>
<td>LMX9820A</td>
<td>2.5</td>
<td>10</td>
<td>14</td>
<td>2</td>
<td>4</td>
<td>10</td>
<td>45.5</td>
</tr>
<tr>
<td>CXN1000</td>
<td>10</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>36</td>
</tr>
</tbody>
</table>

The first criteria in the decision matrix is the package type, size, and pin configuration. The weighting for this criterion received a 20 because a good package will allow the IC to be mounted easily to the circuit board. The BlueCore2 External received a 5 because the package is very small with a 96 pin ball grid array configuration [6]. The LMX9820A received a 2.5 because it is available in a package very similar to the BlueCore2 External, but it has 116 pins [7]. The CXN1000 received a 10 because it is available in a slightly larger package with 36 pins [8].

The second criterion is the features included on the IC. This criterion has a weighting of 15 because more features provide easier design. The BlueCore2 External received a 12 because it includes the essential Bluetooth™ components, and it allows software to execute on it without a host [6]. An 11 is given to the LMX9820A because detailed information of the features was not found, but it is advertised to include all the necessary Bluetooth™ components [7]. The CXN1000 received a 14 because it has all the features of the BlueCore2 External, but it also has a radio filter and voltage regulators included onboard [8].

Price of the IC is the third criterion. This received a weighting of 5 because all the other factors are more important than the price. The BlueCore2 External is available in quantities of five for $54, and this price will fit in the team’s budget. Therefore, it received a 4 in the matrix. The LMX9820A is available at about $35 each from Arrow Electronics. This would also fit in the team’s budget, but this is much more expensive than the BlueCore2 External, so it received a 2. A price for the CXN1000 could not be found and therefore it was given a 0.

The availability of the IC is an important criterion in the decision matrix. If the IC is not available then it cannot be used, so this criterion received a 25. The BlueCore2 External received a 23, because there is no lead time specified at the given price [4]. There is a six week lead time at Arrow Electronics for the LMX9820A. This is not a major problem because if the decision were made to purchase this device the module would still arrive in time to implement. However, if something went wrong and another LMX9820A needed to be purchased later, the lead time could present a problem with the May deadline. For these reasons, the LMX9820A received a 10 for this criterion. Since no price has been found for the CXN1000, the IC does not seem to be available in the US, so it was given a 0.

The last and most important criterion is the documentation provided with the IC. This criterion received a 35 because the documentation provides the team with the information needed to design using the particular IC. The BlueCore2 External’s manufacturer provides a vast amount of information for its products, but there has been little information found that describes the software execution of the IC. These reasons give the BlueCore2 External a 30 for documentation. National Semiconductor does not give a free detailed data sheet for the LMX9820A, but it looks like more information is available to
member of the Simply Blue development page. Since there is probably a lot of information that the team
cannot access without purchasing the IC, the LMX9820A receives a 20 for the documentation. The
CXN1000 has a 24 page datasheet that has some good information, but no other information has been
found relating to this product. Therefore, the CXN1000 receives a 12 for the documentation.

The weighted totals in the decision matrix show that the BlueCore2 is clearly the best choice for the
design team to purchase.

The research for the remaining hardware components revolves around the selection of the
Bluetooth™ IC. Since the BlueCore ICs look like the best choice, research has been done to find the
other components needed to operate the BlueCore IC. Manufacturers are being looked at for voltage
regulators, resistors, capacitors, antennas and filters. These components depend on the final circuit design
and they cannot be confidently selected until the final schematic is completed.

4.3. Mobile Phone

The mobile phone research was focused in two areas: cellular device and software development kit.
The cellular device must be researched to determine if it can connect the base station, and if it has the
capabilities of running the chosen program. This also means it must be compatible with the chosen
programming language. Secondly, available SDK’s must be researched to ensure that the programming
language used can be used on the cellular device.

4.3.1. Cellular Device

In order for a mobile phone to operate properly in the proposed system, it must meet a few important
requirements. Most importantly the phone must have Bluetooth™ Technology in order for it to
communicate with a base device. A programming language that can be used to implement a client
program must also be available on the phone.

There are several programming languages we can use to control the cell phone, and the programming
language that is decided will directly impact the type of phone that can be used. The three programming
language candidates are Java, C++, and Python, which is very similar to C++. The most promising
language is Java, because it is more familiar to the team than Python, and it provides the desired
functionality to implement the client program. Also, if the program is to be optionally downloaded by a
user, then Java has an advantage over the other languages. Java games and other programs can already be
downloaded on Java-enabled phones, so downloading a Java programming to a Java-Enabled phone is
feasible. If the program is written in Java, the phone being used must be Java-enabled. Java-enabled
phones are new, so cost becomes an important factor.

A less powerful alternative is C++. C++ does not provide the same functionality as Java, but it would
not require a Java-enabled phone [9]. However, C++ is the most familiar programming language to the
design team, so it would require a minimal learning process. Another programming language that is
based of C++ is python. Python is unfamiliar and would require extra time to learn the programming
language. Like C++, though, Python would not require a Java-enabled phone, so another option could be
dropped.

Bluetooth™ technology, the type of programming language, and cost are the most critical
requirements when purchasing a cellular phone for testing, but there are other properties that were
researched as well. The amount of free memory available on the phone had to be considered, but most
phones provide plenty of memory for storing a phone book, keeping calendars, etc. The method of
transferring the program to the phone was also an issue. Most phones, like the Java enabled phones, can
transfer programs and information via a Universal Serial Bus (USB) cable. Finally, the program
interaction with the phone was considered. After meeting with our consultant, Tim Theriault from Smiths
Aerospace, it was determined that we should meet with several cell phone vendors to discuss how the
settings on the phone can be saved and how the phone can/will run the client program.

If Java is the decided programming language then the phone must be Java-enabled. Phones that have
Bluetooth™ technology and are Java-enabled are very expensive, and they are a little more difficult to
find because the technology is new. Motorola and Nokia manufacture several phones that meet the major
requirements. The phones are listed in Table 2. The prices in the table assume that a service plan will be
purchased. The cell phone purchased for this project will not include a cell phone plan, so the actual costs
for the cell phones will increase approximately 100 US dollars. The table can still be used as a
comparison for the different cell phone options.

Table 2 List of possible cellular phones [10] [11] [12]

<table>
<thead>
<tr>
<th>Motorola MPx220</th>
<th>269.99</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motorola RAZR V3</td>
<td>250.00</td>
</tr>
<tr>
<td>Motorola V333</td>
<td>149.96</td>
</tr>
<tr>
<td>Motorola V360</td>
<td>279.99</td>
</tr>
<tr>
<td>Motorola V551</td>
<td>74.99</td>
</tr>
<tr>
<td>Nokia 3660</td>
<td>174.99</td>
</tr>
<tr>
<td>Nokia 6256i</td>
<td>149.99</td>
</tr>
</tbody>
</table>

These cell phones will also have to be compatible with the SDK chosen. SDK’s will be discussed in the
next section.

If the programming language for the phone is C++ or Python, then the phone will have to match the
SDK plan being implemented and have Bluetooth™ technology. Interestingly, the same phones from
table 1 would need to be purchased if either of these languages were chosen. Therefore, choosing C++ or
Python as the programming language would save no money.

4.3.2. SDK Research

Researching and deciding on a Software Development Kit (SDK) is directly dependent on the type of
cell phone that is used. A lot of SDKs are designed specifically for one cell phone model. Therefore, it is
important to know what type of cell-phone as well as the SDK that will be used in prototyping.

In the case of Python, the most promising SDK is the Python Series 60 SDK [13]. Only Nokia series
60 phones can be used with this SDK [14]. The SDK will be able to provide all the functionality that is
needed, assuming that the phone is a Nokia Series 60. There will be plenty of memory on the mobile
phone to store the client program because mobile phones have storage space used for saving items like
phonebooks and calendars, and the mobile phone has Bluetooth™ Technology. The Python Series 60
SDK is also free, so that eliminates an item that would have to be purchased.

If C++ is the chosen programming language, then the Symbian OS C++ SDK will be the software
development kit. Like the Python Series 60 SDK, the Symbian OS C++ SDK will provide all the
functionality needed, for the program; however, the Symbian SDK is not free. Funds would have to be
invested in buying an SDK if C++ is the chosen programming language.

Java is the preferred programming language because many phones already allow java programs to be
downloaded onto their phones, like games. Therefore, downloading our client program onto the phone is
feasible. The SDK used will be the Java 2 Standard Edition (J2SE) [15]. Java 2 Platform, Micro Edition
(J2ME) will be used to provide the environment to run J2SE on the cell phone. J2ME is designed to work
on cell phones, plus other packages can be downloaded to implement new technologies like Bluetooth™
[16]. J2ME and J2SE are both free to download.

4.4. Connection Method

In an effort to determine the best method of connection for the design, a decision matrix was made to
weigh the different options against the criteria, see Table 3. The criteria were selected to reflect the most
important aspects being sought in the connection method. The amount of user interaction required is the
most important factor. The connection method should require the least amount of work on the part of the
user. Due to this, the user interaction criterion is weighted at 60 percent. The four other criteria were
considered to be of equal weight because they can all be worked around if the other criteria prove to be
strong. In the decision matrix, higher numbers demonstrate better performance.
Table 3  Connection Method Decision Matrix

<table>
<thead>
<tr>
<th>Device</th>
<th>User Interaction</th>
<th># of Connections</th>
<th>Implementation complexity</th>
<th>Speed</th>
<th>Availability</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weighting</td>
<td>60</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>100</td>
</tr>
<tr>
<td>Infrared</td>
<td>20</td>
<td>2</td>
<td>10</td>
<td>5</td>
<td>10</td>
<td>47</td>
</tr>
<tr>
<td>Docking Station</td>
<td>10</td>
<td>2</td>
<td>5</td>
<td>5</td>
<td>10</td>
<td>32</td>
</tr>
<tr>
<td>Cellular Network</td>
<td>5</td>
<td>10</td>
<td>2</td>
<td>5</td>
<td>10</td>
<td>32</td>
</tr>
<tr>
<td>Bluetooth</td>
<td>60</td>
<td>8</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>86</td>
</tr>
</tbody>
</table>

Infrared (IR): The user interaction of this connection method received a 20 because it requires the user to hold the phone’s IR port in line of sight. IR is capable of only one connection at a time, lending this criterion a two. There is, of course, the possibility of using multiple IR ports, but this would still require too much user interaction for our application. IR is a simple standard of communication that has been used for many years and would be easy to implement into hardware. The complexity of implementation is therefore given a 10. The speed with which this connection can be created is dependent upon the user presenting their phone and properly aligning it. This lengthy process, from the perspective of modern electronics, gives IR speed a value of five. Since IR has been used for many years, the components for its implementation are easily obtained and it is used on many modern phones. The IR method therefore receives a 10 for this category.

Docking Station: The user interaction of this connection method received a 10 because it requires the user to place their phone in a docking cradle – arguably a difficult task. The docking station would be capable of only one connection at a time without becoming cluttered, lending this criterion a two. Designing a docking station that would accept all models of cellular phone would be extremely difficult, as each phone manufacturer has a different connector design. The complexity of implementation is therefore given a five. The speed with which this connection can be created is dependent upon the user presenting their phone and properly inserting it into the docking cradle. This lengthy process, from the perspective of modern electronics, gives the docking cradle speed a value of five. Since docking cradles have been used for many years, the components for implementation are easily obtained as all phones are designed for some form of cradle interface. The docking station method therefore receives a 10 for this category.

Cellular Network: The user interaction of this connection method received a five because it requires the user to provide their personal cellular phone number, as well as accept any call or data transfer sent to the phone from the base. The cellular network is capable of contacting all of the required phones at the same time, lending this criterion a 10. The cellular network is well established technology. This method would require hardware to interface with this network and contact the phones. However, creating the capability to change settings on the phone via this network would present insurmountable challenges, as this form of interaction would infringe upon proprietary phone features. The complexity of implementation is therefore given a two. The speed with which this connection can be created is dependent upon the cellular network and the speed with which the user accepts the incoming call. This lengthy process, from the perspective of modern electronics, gives the cellular network speed a value of five. Since every cellular phone requires a use of the cellular network, availability receives a 10.

Bluetooth™: The user interaction of this connection method received a 60 because it requires no user interaction. Bluetooth™ is capable of up to seven simultaneous client connections, lending this criterion an eight. Bluetooth™ is a relatively new technology, but is also relatively contained. The Bluetooth™ standard is contained within its IC implementations. Due to this, the hardware design needs only to incorporate this IC. The IC packages, however, are difficult to interface without sophisticated component mounting equipment. The complexity of implementation is therefore given a five. The speed with which this connection can be created is dependent upon the paging time of the Bluetooth™ connection – 1.804 seconds. [17] This is the fastest connection speed that can be expected from any of the considered connection methods and is given a value of eight. Since Bluetooth™ is still a relatively new technology, it is not on every cellular phone on the market today. Bluetooth™ has been well received, however, and is being included in most of the new phones being produced. Bluetooth™ has strong potential for the future, but due to its current lack of availability, this criterion receives a five. Based upon the decision matrix, Bluetooth™ was chosen as the connection method.
Bluetooth™ is a newly emerging technology designed for the interconnection of electronic devices without the need for wires. The Bluetooth™ technology was developed by a panel of corporate collaborators including Siemens, Intel, Toshiba, Motorola and Ericsson, and others [18]. The need for this technology has been seen as more and more people begin to connect more and more electronic devices -- such as PDA’s, cell phones, and mice/keyboards -- to one another. In order to circumvent the need for many confusing wiring configurations, Bluetooth™ was developed as a trouble-free method of device connection.

The most important feature of Bluetooth™ technology is that it is automatic via the Bluetooth™ Service Discovery Protocol (SDP) [19]. There exist two modes of Bluetooth™ operation: master and slave. In master mode, a Bluetooth™ device continually scans for other Bluetooth™ devices within range for connection. The device to be connected can be in either master or slave mode. Once such a device is found, the Bluetooth™ protocol handles the communication between the two. If a master and a slave connect, the slave requests connection information from the master and checks that the master is recognized as a trusted device by the slave system. This is a security measure to ensure that Bluetooth™ devices cannot fall prey to any other Bluetooth™ device. If the master is recognized as a trusted device, the slave accepts the connection and the two devices proceed to transfer information between one another. If a master device wishes to connect to another master device, one must take on the role of slave. This is decided during the authentication process. The master device that becomes the slave can still be master to other devices. In this way, a network is formed between trusted Bluetooth™ devices within range of one another. Such a network is referred to as a piconet, due to its limited range between master and slave devices.

In an effort to make Bluetooth™ a more universally applicable standard for short-range, secure wireless communication, the communication frequency was chosen to be 2.45GHz. This is a frequency set aside for world wide for wireless communications. Unfortunately, this frequency is widely used and leads to noise generation from other devices. To overcome this problem, Bluetooth™ utilizes spread-spectrum frequency hopping and switches the frequency of communication at random within 79 variations of the 2.45GHz band 1600 times every second [18]. The next frequency to switch to is decided by the client and master. In this way, Bluetooth™ can function without loss of performance even in high noise environments. This continual frequency shifting also allows multiple Bluetooth™ devices to operate within range of one another without interference from the other devices. To avoid interfering with other wireless devices, the Bluetooth™ protocol transmits with the power of only one milliwatt [18]. This is the main factor limiting the range of Bluetooth™, a range of about 10m [18].

The Bluetooth™ protocol contains two data transfer modes, Synchronous Connection-Oriented (SCO) and Asynchronous Connection-Oriented (ACO). The SCO connection is a full-duplex connection used mainly for audio and video streaming purposes and transmits and receives at 432.6kbps [18][19]. The ACO connection is a half-duplex connection used for non-real-time communication and transmits at 721kbps while receiving at 57.6kbps [18][19]. An Asynchronous Connectionless (ACL) mode also exists in which the master can transmit to either a single slave or all of the slaves to which it is connected [18].

5. Project Management

The design team has two ways of managing the project. The college course that this design is associated with keeps the project moving in the right direction and on the right pace. Within the team, tasks have been divided among the members as the project progresses along a schedule.

5.1. External Organization

Silence is Golden is a team project that is part of a year long engineering capstone course at Calvin College. The purpose of the course is to gather everything learned in a student’s college career and integrate them into a project that will simulate an industrial experience. The senior engineering class has been divided into teams of four or five people to work on a project of their choosing. While this course is a little more open-ended than most courses at Calvin, it still requires about a dozen deliverables to make sure each team stays on track.
Each team is assigned an industrial consultant for analyzing the team’s design work. These consultants have an engineering background, but are not currently affiliated with Calvin College. Consultants provide teams with an outside perspective on how the team is progressing. Silence is Golden has been assigned Tim Theriault from Smiths Aerospace. The team has already met with him once and will meet with him one more time next semester. He was able to provide advice on topics that should be pursued at this point in the design.

Each team is also assigned a professor as an advisor/manager for their project. Silence is Golden’s advisor is Steven VanderLeest. The design team communicates with Professor VanderLeest through weekly status reports, meetings, and various other deliverables.

5.2. Internal Organization

Silence is Golden also has organization within itself. Specific roles have never been formally defined within the team, but each member has voluntarily taken responsibility for certain areas of the project. Nick Ellens has taken the role of the team leader, and keeps the team on the tasks at hand. Nick Ellens and Jared Heys have been working together on the base station design; Ellens is specializing on the software of the base station while Heys is specializing on the hardware. Matt Cosnek is researching the client phone and SDK. Ryan Smith looks over the project as a whole, and brings the two parts together by working in whichever area needs assistance. Smith also performs general tasks such as the bill of materials (BOM) and similar product research.

Currently a rigid schedule for working together has not been implemented. The team does meet every Friday morning to catch up on what has been accomplished for the week. This time is also used to send a weekly status report to the team’s advisor. Tasks that need to be done during the coming week are decided and delegated. Often times, more than one person is delegated a certain task. Those persons decide on a time during the week at which they can meet and work on the task together.

Team conflicts have also been encountered as the design process has progressed. On many occasions, team members have had different ideas on how a certain aspect of the system will work. As conflicts such as these arrive, the team sits down and discusses the matters at hand. Design decisions are discussed until all team members agree.

Sometimes one or more of the team members do not complete the work assigned to them in time. Time delays have been dealt with by shifting tasks and rescheduling assignments. If someone does not complete their work for the week the uncompleted assignment is pushed to the following week along with the newly delegated tasks.

5.3. Project Timeline

There are two different processes to consider with the project: prototype development and production design. The majority of the work will be devoted to designing the base device, and then building and testing a prototype based of the design. The prototype will be used as a proof of concept, which can be used with most cellular phones that are Bluetooth™ enabled.

The goal is to design a device that will be profitable if mass-produced and sold to businesses and organizations. A separate pricing analysis other than prototype will be required, because mass-produced products generally cost less. Also included in the development will be the final appearance of the device. Our prototype must be aesthetically pleasing to improve marketability.

Two programs to help schedule the tasks for our team were used: Microsoft Excel and Microsoft Project. Excel is being used as a way to keep track of how many hours are spent on each task, and to estimate the number of hours it will take to complete any given task. A total of 1406 working hours for our project has been scheduled, which is about 350 hours per person over the course of two semesters. Generous amounts of time were given to each area of the project so that time issues are not encountered later in the second semester. Approximately 300 hours have been logged, which leaves about 1100 hours of work. This seems like a large difference; however, the plan is to continue the project through the holiday seasons. Also a lot of the tasks have been scheduled for the month of January and the second semester.

There are a few major milestones that need to be met during the course of the project. The first major milestone is the composition of the PPFS. The PPFS was allotted 71 hours and was worked on
continuously from the project start date. In total, 113 hours have been logged, which is over budget due to low time estimation for proofreading. Overall system design is scheduled to take 68 hours but only 38.5 hours have been spent. This part of the project is basically complete and is under the budgeted time.

The next milestone to be completed is the research and design of the base device circuit, which was budgeted 240 hours, and to obtain a cellular phone and start writing the code, which was budgeted 150 hours. These will be done simultaneously by splitting the work up among the team. The two major components that need to be purchased are included in this part of the design: cellular phone and associated SDK as well as the Bluetooth™ IC and associated SDK. All of these parts were scheduled to be ordered before the end of the semester. Due to unexpected high cost of some of these products, sponsorship or student discounts must be pursued. These tasks caused delay on deciding on and ordering these components. However, plans are being made to contact various companies over the holiday break and these parts will be ordered in early January. Circuit design and the cellular phone program are planned to be completed by the first of February.

Implementing the base device circuit onto a PCB, producing the PCB either in house or through a company, and then assembling the device should take 170 hours. While this is being done, a test procedure will be developed and test equipment will be acquired. Testing the cellular phone code in isolation from the base device will also commence at this time. It has been budgeted 20 hours. This should be completed by the first or second week of March. The testing of the base station will also take place at this time, followed by full system testing during March and April (~70 hours budgeted). The rest of the budgeted time comes from attending lectures, documenting tests and final design, preparing for presentations, and other miscellaneous tasks. For a more detailed analysis on the budgeted hours, see the Task List in Appendix A.

5.3.1. Prototype development

As stated above, the prototype will be the proof of concept. By the end of spring semester, a working prototype will be complete for senior design project night. Though the prototype needs to be aesthetically, cost needs to be kept low and the prototype needs to be easy to test. For example, not having tiny electrical traces will simplify the testing procedure. To keep costs low, a PCB could be fabricated in house; however, that limits the complexity and the size of the PCB.

The design of a prototype device is in progress. BlueCore2 External has been chosen as the Bluetooth™ IC. This will allow the development of the circuit design to commence.

5.3.2. Production design development

The production design will be the final result of our project, with a prototype to prove that it works. As the base device is developed, the components will determine the size and shape of the production model, but care will be given to keep the design as small as possible. Knowledge of the component costs for the prototype will allow a cost estimate for mass-production of an aesthetically pleasing product for the consumer market. Cost will be kept to a minimum.

6. Preliminary Feasibility Analysis

The design team has had limited experience in many areas of this project, and, therefore, there will be many challenges to overcome. One of the challenges is that the technology being used, despite having been available for some years, is still relatively underdeveloped. Bluetooth™ was released in 1998 and it is still only incorporated on high end mobile and communication devices. Therefore, there is not a lot of information available to aid the team in the design process. The budget is also affecting many areas of the project. The sections below will describe the major problems that are specific to each part of the project and how the design team is dealing with them.

6.1. Base Station

When analyzing the feasibility of the Silence is Golden system, two separate components need to be addressed: Hardware and Software. The hardware analysis evaluates the feasibility of integrating a BlueCore IC, and other components onto a PCB. The software analysis evaluates the feasibility of writing code and executing it for the base station.

6.1.1. Hardware
The Bluetooth™ technology is slowly becoming more available for design. Further research into the BlueCore ICs needs to be done. The application notes and design examples given on the manufacturer’s support webpage are a valuable resource that will provide the team with the information required to implement the Bluetooth™ technology. A thorough read through of the 100 page datasheet for the promising BlueCore2 IC has been done, and a lot has been learned about the operation of Bluetooth™ and the IC. The information and application notes about the BlueCore ICs provide the information needed to design the base station hardware. The team estimates that 20 more team hours will be required to complete the research and to confidently be able to produce a circuit that implements a BlueCore IC.

The requisite ICs, including the BlueCore2, a balun transformer, and a band pass filter, are packaged in a way that may cause some difficulties when trying to mount them to a circuit board. All the BlueCore ICs are only available in ball grid array (BGA) and land grid array (LGA) packages. The other ICs are also only available in surface mount packages, which are difficult to mount to a PCB. The design team has one possible contact that may provide some help with mounting these components. The team is also looking into breakout boards or kits that will allow the implementation of the BlueCore ICs without the need for special component mounting equipment.

Budgeting for the base station has not been found to be a big problem. The major components have been found at a reasonable price. The BlueCore ICs cost between $35 and $70 for a quantity of five [4]. Some of the other mentioned ICs that the team anticipates needing have been found at Digi-Key for about one dollar apiece. Budget concerns were encountered when looking at development kits. These are very expensive, the most expensive being $3000, but it is not yet clear whether such a development kit will be necessary. A development kit would be very helpful, however, and the team is currently attempting to contact the manufacturer to see if student discounts exist.

6.1.2. Software

The main problem with the base station software is determining how the Bluetooth™ IC executes programs. Some limited information has been found with the Bluetooth™ IC data, but no examples or requirements for how the program should be written have yet been found. Research in this area is ongoing in conjunction with the Bluetooth™ IC research. This issue may result in budgeting problems, because programming the Bluetooth™ IC may only be possible using a software development kit. These kits can be very expensive. Once the information is found to implement the base station software, it should be feasible to complete this part of the project.

6.2. Client Software

6.2.1. Hardware Control

Writing and implementing the client software poses a challenge in the area of controlling the settings on the phone. Information on the selected phone will need to be found to determine if this is possible. Initial research into similar products shows that the control of cellular phone features is possible. The BlueLinx product discussed in section 4.1 above uses software on a cellular phone to control its settings [2]. Contact has been made with some potential cellular phone vendors to seek help on access to control the settings of a cellular phone. Once the required information is found, control of the phone hardware by a software program will be feasible for the team.

6.2.2. Base Recognition

The client program also has the challenge of recognizing the base station as a trusted device. Bluetooth™ connections require verification that the device they are connecting to is trusted and secure. This will require the base station software to have a unique identification that the phone can recognize when it is trying to connect to it. This part of the program will be feasible when information is found to describe the Bluetooth™ connection process in depth.

6.3. Cost Estimates

The preliminary cost estimates show that the prototype will be very close to the maximum price of $300. A major cost concern is the price of the cellular phone for use as a development platform. Most of the Bluetooth™ enabled phones are currently the high end phones, so they cost close to $200. Several avenues of phone acquisition are being explored. There are many cell phones that the team should be
able to use for closer to $100 when searching on eBay. Also, the team is contacting cell phone venders to seek a possible student discount. Cost estimates of the base hardware components show that these parts should total less than $100. Some of the parts that the team anticipates it will need are located in section Error! Reference source not found., Bill of Materials. Prices and possible sources are listed as they have been found during this stage of the project. The final Bill of Materials cannot be completed until the final circuit is designed.

7. Project Requirements

This section lists the requirements for the base station and the client software. Each list is laid out arbitrarily.

7.1. Base Station

The base station was broken up into to separate requirements. There are requirements for the base hardware, and there are requirements for the base software.

7.1.1. Base Hardware

1) Base station shall affect a minimum range of a 5 foot radius, and no more than 33 feet
2) The base and client software shall be capable of communication via the Bluetooth™ connection
3) The base production design of the enclosure height, depth, and width shall be limited to 4” x 4” x 2”
4) The base weight shall be limited to 1lb
5) The cost of the production design shall be no more than $100
6) The wattage of the power supply shall be limited to 10 watts for the prototype and two watts for the production design
7) The voltage output from the power supply shall be limited to five volts
8) The base shall meet temperature requirements of ICs
9) The base shall operate within a specified temperature range of 0°C to 50°C
10) The base shall have a power Light Emitting Diode (LED)
11) The base shall meet all FCC requirements for such a device
12) A base unit shall not disrupt operation of another base unit if they are located within range of one another
13) The base shall be capable of withstanding the transient effects of a normal power outages, if not battery operated
14) If the device is power by batteries, the batteries shall last at least one week before needing to be changed

7.1.2. Base Software

1) The base software shall be capable of connecting to 7 clients simultaneously. This is the maximum number of slaves a host can support in a piconet [18]
2) The base software code shall not be accessible to the base user

7.2. Client Software

1) The size (Mb) of the client program shall be no greater than 1Mb, based on the typical free space available on most cellular phones
2) The client shall be capable of storing ring tone settings prior to encountering the base
3) The program shall be capable of restoring the stored ring tone settings when leaving the base area
4) The client shall automatically accept the Bluetooth™ connection from the base as a trusted device
5) The client software shall not require user interaction, but will provide the option for silent or vibrate mode
6) The client software shall be available as a stand-alone program
7) The client software code shall be protected against modification as much as possible
8) The client software shall be responsible for comparing its current status with the base station signal and determining the appropriate course of action.
   a. A phone that does not leave the base station’s range will still be silenced and restored properly
   b. The phone shall not recognize the difference between different base stations at different entrances/ exits. Therefore, the client shall be activated and deactivated properly by any base station it connects to
   c. The user shall be able to manually restore the setting at any time in case the settings fail to restore

8. System Design

The premises behind the operation of this design are the interaction between the base and the client, and the lack of interaction between the base and other bases. Figure 3 shows a composite block diagram of program flow. There are four points of program interaction during the length of the connection:

1. Point of Discovery
2. Device Recognition
3. Client Recognition
4. Toggle Code

The first two are implemented as part of the Bluetooth™ standard and are integrated features of a Bluetooth™ IC. The second two are authentication specific to this design. Detailed operation of the base and client software programs is discussed in sections 8.1.2 below and 8.2 below respectively.

When operating multiple base stations within range of one another, it is important that they do not interfere. Fortunately, this is also a feature that is incorporated in the Bluetooth™ standard and, therefore, into Bluetooth™ IC implementations. By frequency hopping at random, see section 4.4 above, the likelihood of separate piconets encountering the same frequency at the same time is extremely unlikely. In this fashion, the individuality of separate piconets can be maintained even when operating within range of one another. The challenge to the product design comes in creating client software that will not be affected by signals from a different base station. Each base station will appear as every other base station and perform the same connection sequence. This will allow the current program flow of the clients to operate unhindered in an environment containing multiple overlapping piconets. See Figure 4.

Program Interaction Flow

1) Base seeks client
2) Base finds client, requests connection
3) Client authenticates
4) IF trusted GOTO 5 ELSE GOTO 18
5) Connection established
6) Client sends ID
7) Base checks ID
8) IF NOT client GOTO 9 ELSE GOTO 10
9) Base closes connection
10) Base sends toggle code
11) Client checks timer
12) IF timer on GOTO 13 ELSE GOTO 14
13) Client ignores base
14) Client checks flag
15) IF ringer on GOTO 16 ELSE GOTO 17
16) Client stores settings, changes ringer, sets timer, toggles flag
17) Client restores settings, sets timer, toggles flag
18) Client enters passive mode
Figure 3 Client-Base software interaction
8.1. Base Device

The base station design consists of two major parts. The hardware design will be the most time consuming portion of the entire project. It will take the most amounts of research and testing. The second component of the base device design is the development of the software that needs to be loaded and executed by the Bluetooth™ processor.

8.1.1. Hardware

A simple block diagram of the hardware design is shown in Figure 3. Each of the blocks will be explained in the following sections.

Figure 4 Piconet interference

Figure 5 Hardware Block Diagram
8.1.1.1. Bluetooth™ Integration

Bluetooth™ integration of the base device will be accomplished using a BlueCore2 External IC. This IC contains all the major functional blocks required for Bluetooth™ integration. These blocks include baseband and logic, a microcontroller, a RF transmitter, and a RF receiver [6]. Figure 4 below shows a section of the data sheet for the BlueCore2 External. The major blocks that will be explained can be seen in this figure.

![BlueCore2 External block diagram](image)

The baseband and logic contains a memory management unit. This unit provides buffers that hold data in transit between host and the air, and it ensures efficient use of the Random Access Memory (RAM) [6]. The baseband and logic also includes 32Kbytes of onboard RAM, and it contains USB, synchronous Serial Port Interface (SPI), and a Universal Asynchronous Receiver Transmitter (UART) to connect to external devices, such as a host [6].

The microcontroller is a 16-bit Reduced Instruction Set Computer (RISC) that runs the Bluetooth™ software stack. This is firmware that runs on the microcontroller. There are 4 different cases of the software stack that are supported by the BlueCore2 External. These are Host Controller Interface (HCI) stack, RFCOMM stack, Virtual Machine (VM) stack, and Human Interface Device (HID) stack. The design team is most interested in the VM stack because no host is required. This means that application code can run with the BlueCore firmware [6]. Fifteen programmable I/O terminals are also available in the microcontroller section. These terminals are controlled by the firmware and can be programmed to function as a chip select, a sleep mode indicator, or many other things.

The RF transmitter includes a power amplifier that will provide enough power to an antenna to operate as a class 2 or a class 3 Bluetooth™ device. A class 3 device has a maximum range of about five feet, and a class 2 device has a maximum range of about 30 feet. This means that, without an external amplifier, the BlueCore2 External IC can supply enough power to the antenna to have a maximum range of about 30 feet. This range will fulfill the project requirements. The design team is considering the design of a long range device for special applications like in stadiums or other locations that have a large entrance or exit area. This will require an external amplifier to power the antenna.
The RF receiver contains a low noise amplifier. This amplifier has sufficient out-of-band blocking so that this device will be able to operate near cellular devices without limiting its performance. The RF receiver also contains an analog to digital converter.

Using this IC will require few external parts to integrate Bluetooth™. The major external devices will be an oscillator and external flash memory. The amount of external memory needed is uncertain at this point because the design team has not yet acquired specific information about programming for the BlueCore IC. The maximum amount of memory supported is 8Mbit. Since this is the maximum, the software will try to be designed to fit in this amount of memory. Contact is being made with the manufacturer to get this information. Noise filtering capacitors are needed at the power and ground pins of the IC to provide stable operation. One or more of the serial ports will need to be set up to connect to a PC or host that can load the software onto the IC.

8.1.1.2. Antenna

The antenna design consists of three main components. A balun is needed to connect the two transmission ports of the BlueCore IC to a single antenna line. The balun is an impedance matching transformer that will allow this type of connection to be made. Another component needed is a band pass filter. The filter needs to have a bandwidth of about 100MHz and a center frequency of 2.45GHz. The last component needed is the actual antenna. Preliminary research shows that a 1/4 wave antenna should work. Research needs to be done to determine whether a dipole or a monopole antenna will work better for this project.

8.1.1.3. Power Supply

To supply power to the circuit, a transformer and at least two voltage regulators will be needed. A 1V and a 2V regulator will be needed to power the BlueCore2 External and most of the surrounding ICs. The flash memory may need a different supply voltage. The datasheet of the BlueCore2 External recommends the MBM29LV800BA flash memory from Fujitsu Semiconductor. This memory requires a 3V supply. There will have to be noise canceling capacitor connected at the input and output terminals of the voltage regulators to provide stable supply voltages.

The transformer will be determined based on whether the base station is powered by a battery or a wall outlet. A battery powered base station would be ideal because it will allow more freedom for device placement; however, it will require more maintenance to change batteries. A large transformer will be needed for the base station to be powered by a wall outlet. This will provide less maintenance for the device, but it will limit the placement of the device. Research and calculations will need to be made after circuit design is completed to determine the power consumption. This will help the team decide if batteries are a good solution to power the base station.

8.1.1.4. Buttons and Switches

A power switch will be connected to the power supply to turn the circuit on and off. A reset button could also be included. Some of the I/O pins on the BlueCore2 External might be able to be programmed to reset the device from a button.

8.1.1.5. Memory

The maximum amount of flash memory, which was stated above, is 8Mbit [6]. This memory will be used to store the Bluetooth™ software stack and the application software that will be written by the design team. The flash memory needs to support 16-bit data width, 110ns maximum access time, and have independently erasable sectors [6]. There is timing diagrams included in the datasheet that show the timing required by the BlueCore2 External. The MBM29LV800BA from Fujitsu Semiconductor is used in the design example. The design team is using this as a starting point to choose a flash memory device. This particular IC has a good datasheet that is very detailed, but a vender and price is hard to find.

8.1.1.6. Crystal

The crystal oscillator is needed to supply a clock signal to the BlueCore2 External. The specifications for the crystal oscillator are given in the datasheet for the BlueCore2 External. The crystal frequency has to be between 8 and 32 MHz. The TSX-10A from Toyocom is the crystal that the datasheet uses for its example.

8.1.1.7. LEDs
LEDs will be connected to the power supply to indicate that the circuit is running. Some of the programmable I/O pins might be able to connect to LEDs to indicate the status of the BlueCore2 External IC. One known possibility at this time is that one I/O pin can be programmed to indicate whether the BlueCore2 External is in deep sleep mode or if it is actively making connections.

8.1.1.8. Host

The host is an external microprocessor system or computer that supplies code and data to the BlueCore2 External. The host can connect to the BlueCore2 External via USB or one of the other serial connections mentioned in section 8.1.1.1 above. The design team is confident that the application software written for the base station will not exceed the available free space in the flash memory. If this is the case, then the team will not need to design and build the host for this project.

8.1.1.9. Enclosure

The enclosure design will be kept as simple as possible. An enclosure will not be designed for the prototype. This will allow the components to be easily accessible for testing and debugging. A solid plastic or light metal will be used for the final production design. The enclosure will be as small as possible to allow for more freedom of placement of the base device. The design will be modeled using 3D CAD software, and the manufacturing process will be determined by the materials selected.

8.1.2. Software

The base station software for this project will be included on an embedded processor mounted to the circuit board with the Bluetooth™ chip. This software will be designed to interface the Bluetooth™ stack. The language to be used for this base station software will be Java or C/C++, depending upon the capabilities of the microcontroller to be used. The software will contain the necessary information to transmit the ID of the base device that will allow it to be recognized by the client as trusted. The base software will be set up such that it keeps the base in master mode and continually scans for slave devices in the area. Once the base finds another Bluetooth device it will attempt to establish a connection to the device. If the connection is accepted, the base will send out an ID request to the device to ensure that it is communicating with a device running the complimentary client program. If it is not a device running the client program, the base will close the connection. If the device is running the client program, the base will send out a toggle code to the client program. The base will then go back into scanning mode, see Figure 5. This program should not occupy much memory, as it is relatively simple and small, requiring only a small amount of expensive external Flash memory.

**Figure 7 Base station software program flow**

8.2. Client Software

The client program for this project will be downloadable, by a device specific method, to the cellular phone. The software will be programmed using the Java programming language to allow for maximum portability between various cellular phone models. This software will operate in passive mode, not actively seeking a Bluetooth connection, as a slave to the base. When a connection request comes in, the Bluetooth protocol will check the requesting ID to see if it is a trusted device. If the device is not trusted, the program will re-enter passive mode. If the device is trusted, the connection will be established.
client program will then send the ID to the base. Once the client program receives the toggle code, it will proceed to check the timer. If the timer is on, the client will ignore the base station. If the timer is off, the client program will check the ring status flag. If the ringer is currently on, the program will save the current ring settings, change the ringer to silent or vibrate, set the timer, and toggle the flag to silent. If the flag is currently silent when the code is received, the client program will restore the saved ring settings, set the timer, and toggle the flag to ring. When either of these processes is completed, the client program will again enter passive mode, see Figure 6.

![Figure 8 Client software program flow](image)

9. Conclusion

As more information is discovered the complexity of the design grows. With each new discovery come difficulties. For example, it is now apparent that the client phone will have to specifically match the requirements of the chosen SDK, and some parts cost more than the initial budget proposal predicted. Major milestones have been met, however, and progress has been made. Client phone, SDK, and Base Station chip options have been researched and narrowed down to one or two options, and the basic operation of the client program has been developed. Since thorough research was performed, future tasks will be better understood and completed with fewer complications.
10. References


17. Woodings, Ryan, Derek Joos, Trevor Clifton, and Charles Knutson. "Rapid Heterogeneous Connection Establishment." Brigham Young University.


11. Appendices

11.1. Appendix A

Table 4 Task List

<table>
<thead>
<tr>
<th>Task name</th>
<th>Hours Budgeted</th>
<th>Hours Logged</th>
<th>Matt: Hours</th>
<th>Nick: Hours</th>
<th>Jared: Hours</th>
<th>Ryan: Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>System design</td>
<td>68.00</td>
<td>39.50</td>
<td>9.00</td>
<td>13.00</td>
<td>9.00</td>
<td>8.50</td>
</tr>
<tr>
<td>Project Specification</td>
<td>10.00</td>
<td>9.00</td>
<td>2.00</td>
<td>3.00</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Alternative Solutions</td>
<td>10.00</td>
<td>6.00</td>
<td>2.00</td>
<td>2.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Determine Overall Goals</td>
<td>8.00</td>
<td>6.00</td>
<td>2.00</td>
<td>2.00</td>
<td>1.00</td>
<td>2.00</td>
</tr>
<tr>
<td>Design Specifications</td>
<td>20.00</td>
<td>16.00</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
</tr>
<tr>
<td>Patent Research</td>
<td>20.00</td>
<td>2.50</td>
<td>2.00</td>
<td></td>
<td></td>
<td>0.50</td>
</tr>
<tr>
<td>Base Station</td>
<td>463.00</td>
<td>13.50</td>
<td>0.00</td>
<td>3.50</td>
<td>6.00</td>
<td>8.00</td>
</tr>
<tr>
<td>Preliminary Research</td>
<td>91.00</td>
<td>13.50</td>
<td>0.00</td>
<td>3.50</td>
<td>6.00</td>
<td>2.00</td>
</tr>
<tr>
<td>Research Best Communication</td>
<td>5.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Determine/Find Required Electrical Components</td>
<td>50.00</td>
<td>3.00</td>
<td>2.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bluetooth Circuit Research</td>
<td>20.00</td>
<td>9.50</td>
<td>3.50</td>
<td>6.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bill of Materials</td>
<td>16.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td>1.00</td>
</tr>
<tr>
<td>Circuit Design</td>
<td>150.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Antenna</td>
<td>15.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power Supply</td>
<td>35.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bluetooth-processor interface</td>
<td>100.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCB Layout</td>
<td>100.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Learning Software</td>
<td>8.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bluetooth-Processor Interface</td>
<td>79.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power Supply</td>
<td>5.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Antenna</td>
<td>8.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testing</td>
<td>66.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Determine Test Procedure</td>
<td>4.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Determine Test Equipment</td>
<td>2.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Perform Test</td>
<td>40.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Document Test Results</td>
<td>20.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enclosure</td>
<td>16.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Research Enclosure Materials</td>
<td>4.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design Enclosure</td>
<td>12.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assembly</td>
<td>40.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Order and Acquire Components</td>
<td>4.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Build PCB</td>
<td>26.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Build Enclosure</td>
<td>10.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cellular Phone Modification</td>
<td>177.00</td>
<td>8.00</td>
<td>8.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Determine Best Cell Phone to Purchase/Dev Kit</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Program Flow/ISA</td>
<td>25.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learn Programming Language</td>
<td>20.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write Code</td>
<td>100.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test Code</td>
<td>24.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Determine Test Procedure</td>
<td>4.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Perform Test</td>
<td>16.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Document Test Results</td>
<td>4.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Full system testing

<table>
<thead>
<tr>
<th>Task name</th>
<th>Hours Budgeted</th>
<th>Hours Logged</th>
<th>Matt: Hours</th>
<th>Nick: Hours</th>
<th>Jared: Hours</th>
<th>Ryan: Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>PPFS</td>
<td>71.00</td>
<td>84.00</td>
<td>16.00</td>
<td>24.50</td>
<td>30.50</td>
<td>13.00</td>
</tr>
<tr>
<td>Outline</td>
<td>8.00</td>
<td>8.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
</tr>
<tr>
<td>Introduction</td>
<td>3.00</td>
<td>2.50</td>
<td>0.50</td>
<td>2.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>1.00</td>
<td>1.50</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design Norms</td>
<td>1.00</td>
<td>3.00</td>
<td>3.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Research</td>
<td>7.00</td>
<td>15.50</td>
<td>6.00</td>
<td>1.00</td>
<td>6.50</td>
<td>2.00</td>
</tr>
<tr>
<td>Project Management</td>
<td>4.00</td>
<td>2.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Preliminary Feasibility Analysis</td>
<td>8.00</td>
<td>9.00</td>
<td>6.00</td>
<td>3.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Requirements</td>
<td>2.00</td>
<td>3.00</td>
<td>3.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Base Device</td>
<td>15.00</td>
<td>9.50</td>
<td>3.00</td>
<td>6.50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Client Program</td>
<td>15.00</td>
<td>3.50</td>
<td>3.50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conclusion</td>
<td>2.00</td>
<td>1.50</td>
<td>1.50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appendix</td>
<td>4.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proofread</td>
<td>12.00</td>
<td>24.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
</tr>
<tr>
<td>Local Documentation</td>
<td>105.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Base Station Hardware</td>
<td>33.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Base Station Software</td>
<td>33.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Client Code</td>
<td>33.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enclosure</td>
<td>6.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Presentations</td>
<td>90.00</td>
<td>14.00</td>
<td>2.50</td>
<td>7.00</td>
<td>2.00</td>
<td>2.50</td>
</tr>
<tr>
<td>Write Presentation</td>
<td>24.00</td>
<td>8.50</td>
<td>2.00</td>
<td>3.50</td>
<td>1.50</td>
<td>1.50</td>
</tr>
<tr>
<td>Practice Presentation</td>
<td>56.00</td>
<td>3.50</td>
<td>0.50</td>
<td>1.50</td>
<td>0.50</td>
<td>1.00</td>
</tr>
<tr>
<td>Executive Summaries</td>
<td>10.00</td>
<td>2.00</td>
<td>2.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Report</td>
<td>36.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Assemble and Compose Documents</td>
<td>24.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proofread</td>
<td>12.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 5 Preliminary Budget

<table>
<thead>
<tr>
<th>Device</th>
<th>Quantity</th>
<th>Estimated Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell Phone</td>
<td>1</td>
<td>$150.00</td>
</tr>
<tr>
<td>Cell Phone Data Cable</td>
<td>1</td>
<td>$30.00</td>
</tr>
<tr>
<td>Bluetooth Chip</td>
<td>1</td>
<td>$55.00</td>
</tr>
<tr>
<td>Bluetooth Chip Socket</td>
<td>1</td>
<td>$5.00</td>
</tr>
<tr>
<td>SDK</td>
<td>1</td>
<td>$50.00</td>
</tr>
<tr>
<td>Processor</td>
<td>1</td>
<td>$10.00</td>
</tr>
<tr>
<td>Processor Socket</td>
<td>1</td>
<td>$5.00</td>
</tr>
<tr>
<td>Assorted Electrical Components</td>
<td></td>
<td>$20.00</td>
</tr>
<tr>
<td>Enclosure</td>
<td>1</td>
<td>$10.00</td>
</tr>
<tr>
<td>Antenna</td>
<td>1</td>
<td>$10.00</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>$345.00</strong></td>
</tr>
</tbody>
</table>
### Table 6 Bill of Materials

#### Base Station

<table>
<thead>
<tr>
<th>Part #</th>
<th>Description</th>
<th>Vendor</th>
<th>Quantity Needed</th>
<th>Price per unit</th>
<th>Minimum Order</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>BC212015C0N</td>
<td>BlueCore 2 External - Single Chip Bluetooth system</td>
<td>ST Designer</td>
<td>1</td>
<td>$10.80</td>
<td>5</td>
<td>$54.00</td>
</tr>
<tr>
<td>MBM26LV000BA-BUP3T</td>
<td>Flash memory</td>
<td></td>
<td>2</td>
<td></td>
<td>30.00</td>
<td></td>
</tr>
<tr>
<td>SHM-1517</td>
<td>16MHz crystal oscillator</td>
<td>AEL</td>
<td>1</td>
<td></td>
<td>1</td>
<td>$0.00</td>
</tr>
<tr>
<td>X05204B102MR</td>
<td>Voltage Regulator</td>
<td>Digikey</td>
<td>1</td>
<td>$2.05</td>
<td>1</td>
<td>$2.05</td>
</tr>
<tr>
<td>MDR241F</td>
<td>ceramic band pass filter</td>
<td>Soshin</td>
<td>1</td>
<td>$10.00</td>
<td>1</td>
<td>$10.00</td>
</tr>
<tr>
<td></td>
<td>about 3 resistors</td>
<td></td>
<td></td>
<td></td>
<td>10.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>about 3 inductors</td>
<td></td>
<td></td>
<td></td>
<td>10.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>about 1 capacitor</td>
<td></td>
<td></td>
<td></td>
<td>10.00</td>
<td></td>
</tr>
<tr>
<td>Antenna</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10.00</td>
<td></td>
</tr>
<tr>
<td>Power Supply</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enclosure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$67.05</td>
</tr>
</tbody>
</table>

#### Cellphone Related

<table>
<thead>
<tr>
<th>Part #</th>
<th>Description</th>
<th>Vendor</th>
<th>Quantity Needed</th>
<th>Price per unit</th>
<th>Minimum Order</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cellular Phone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data Cable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bluetooth Access Point</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>