Secure Solutions

School of Engineering Science, Burnaby, BC V5A 1S6
*http://www.sfu.ca/~chrism/ss * ss-ensc@sfu.ca*

December 16, 2002

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, BC V5A 1S6

Re: ENSC 340 Process Report for a Motion Sensitive Laptop Anti-theft Device

The attached document, *Process Report for a Motion Sensitive Laptop Anti-theft Device*, describes the functions of our entire system for ENSC 340.

Our goal was to design and prototype a laptop anti-theft device that comprises of a pair of programmable units that will alert a laptop owner of any possible theft attempts. The contrivance, called the Infiltrator, features automatic arming and disarming based on the proximity of the owner to his laptop, and enables the owner to screen for false alarms.

The purpose of this report is to explain the current state of the system, any deviations from the design specifications, and future plans for the technology. In addition, we will list budgetary and time constraints we overcame and discuss group dynamics and educational benefits of the project.

Secure Solutions consists of five engineering students: Vincent Au-Yeung, Matt Brown, Steve Lau, Hani Mehrpouyan, and Chris Mitchell. If you have any questions or concerns in, please feel free to contact Chris Mitchell, our contact person, by phone at (604) 984-8771 or by email at chrism@sfu.ca.

Sincerely,

Chris Mitchell

Chris Mitchell
COO
Secure Solutions

Process Report for a

Motion Sensitive Laptop Anti-theft Device

Project Team: Vincent Au-Yeung
Matt Brown
Steve Lau
Hani Mehrpouyan
Chris Mitchell

Contact: Chris Mitchell
chrism@sfu.ca

Submitted to: Dr. Andrew Rawicz – ENSC 340
Steve Whitmore – ENSC 305
School of Engineering Science
Simon Fraser University

Issued: December 16, 2002

Table of Contents

1 Introduction 3
2 Design Synopsis 4
2.1 Deviations 4
2.1.1 Microcontroller 4
2.1.2 Tilt Switches 4
2.1.3 Siren 4
2.1.4 Antenna Specs 4
2.1.5 Power Supply 4
2.1.6 Transceiver to transceiver interface 4
2.2 Problems and Resolutions 5
3 Budgetary and Time Constraints 6
3.1 Finances 6
3.2 Scheduling 7
4 Group Dynamics 8
5 What We Would Do Differently 8
6 Future Plans and Recommendations 9
7 Individual Reflections 9
Appendix - Schematics 14

1 Introduction

The Infiltrator is a dual-unit system that uses RF communication to interact between its Console and Remote to send the status of the owner's laptop to him/her and notify of any possible theft attempts. The entire device will automatically arm and disarm itself based on the proximity of the owner to his/her laptop. Figure 1, below, illustrates a high-level view of our system.

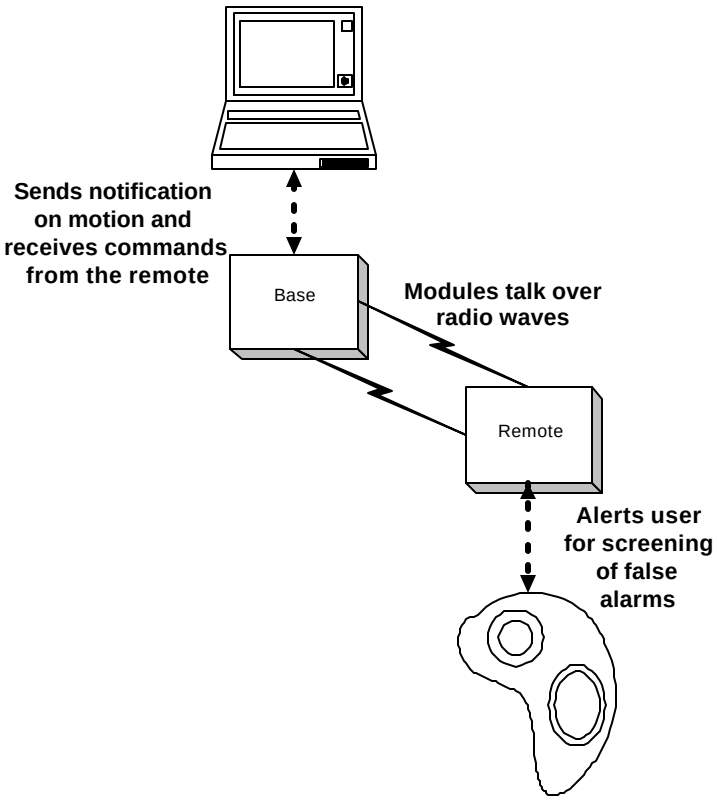

Figure 1 – Dual Unit Anti-theft Device Overview

Theft of laptops continues to be on the rise for business people and others who travel with their notebooks. Over 600,000 laptops were stolen in the United States last year, an increase of 53% from the previous year. Unfortunately, the increasing popularity of laptop computers has spawned substantial black markets in both stolen computers and stolen confidential business data. These facts bring into play our standalone solution to combating laptop theft.

2 Design Synopsis

2.1 Deviations

2.1.1 Microcontroller

The microcontroller used in our prototype is much more efficient than the one stated in our design specifications; we were able to optimize the code to fit on a smaller chip, the Atmel AT90S2313. It is an 8-bit RISC processor with 2K of flash memory and 128 bytes of RAM memory.

2.1.2 Tilt Switches

Our final design uses only a single tilt switch, supplied by American Electronics Components Ltd. During extensive testing periods we found that one tilt switch was sensitive enough to vibrations in all directions to be sufficient for the type of motion detection we're looking for.

2.1.3 Siren

The siren integrated in the prototype is the same as specified in the design specifications, however the pitch (frequency) is performed using analog circuitry as opposed to a PWM signal from the microcontroller.

2.1.4 Antenna Specs

For the antenna, we decided upon a small JJB antenna, coil monopole, because we never made it to the PCB stage of our design so the loop antenna would be more difficult to implement. The JJB is small and provides us with the best results for transmitting distance at low power on the transceiver when compared to several other antenna variations.

2.1.5 Power Supply

For the remote, 3 1.2V AAA rechargeable batteries were used to power the components. The AAA's were chosen over the single 3V battery, as stated in the design specification, because we already had a AAA charger allowing us to recharge the batteries instead of having to purchase handfuls of 3V batteries for our extensive field testing.

For the console, it was suggested to us to use a Buck converter as the voltage regulator, but due to size restraints and future ideas for the siren, we decided to stay with the basic, linear 3-pin voltage regulator as stated in the design specification.

2.1.6 Transceiver to transceiver interface

Data is transmitted through 74 bit packets. Each packet consists of a preamble, a header and data. The header on the packet is used to make the user of the security device unique and the data enables different messages to be transmitted between the Base and Remote unit.

Figure 2 shows the contents of each packet transmitted between the Remote and Base.

50-bit Preamble	16-bit Header	8-bit Data

Figure 2 - Packet Information

The 50-bit preamble is used to synchronize the receiver and transmitter and it is also used by the receiver to lock its internal free running filter. The 16-bit header is used to signal the start of the data packet. It separates the preamble and data and serves as a unique code for each Infiltrator so over 32,000 can operate with no problems at a single frequency. The 8 bits of data are used to transmit different messages between the Remote and Base, corresponding to the different modes of Infiltrator.

2.2 Problems and Resolutions

- Problem: Unable to keep the Remote and Base in synchronization due to slight differences in there respective operating crystals.

✓ Resolution: Need to keep the Base in a longer receive mode (causes us to use more power) after a few minutes to account for these imperfections. This allows us to still keep the 2 systems communicating if the owner goes away for a long time and then returns back to the signal range.

- Problem: Motion sensor interrupt caused the system to behave erratically.

✓ Resolution: We disabled that particular interrupt when the system is disarmed, allowing for lots of motion to happen without problems.

- Problem: Chipcon User Guide contains an error for what bit pattern need to be sent as a preamble.

✓ Resolution: Contacted Chipcon directly and got the proper bits.

- Problem: The Atmel development board seemed to reset for no apparent reason during operation.

✓ Resolution: We realized that we didn't declare the operation of the ports that we were not using. So they were floating, causing internal problems for the microcontroller and caused it to randomly reset. To solve this we set all the pins we weren't using as output pins so they would not affect the operation of the microcontroller.

- Problem: Switching the siren on and off periodically using the microcontroller turned out to be extremely difficult because of the complexity of our timing algorithm.

- ✓ Resolution: We decided to make a square wave generator out of transistors, capacitors and resistors to create the periodic siren output. All the microcontroller had to do now was simply flip a port 'high' to turn on the siren instead of creating new timers and checking for interrupts.

- Problem: The tilt switches weren't sensitive enough no matter what orientation or circuit configuration we put them in.

- ✓ Resolution: Designed a spring-like mounting system to allow a single tilt switch to rock quite a bit for small movements in any direction. This spring mounting solution reduced our final number of tilt switches needed in the prototype as well as increased the motion and vibration sensitivity.

3 Budgetary and Time Constraints

3.1 Finances

Projected	
Equipment	Cost
2 Microcontroller Evaluation Boards	$200
Transceivers: HW/SW	$0 (borrowed)
Tilt Switches/Buttons/Etc.	$20
Alarm/Speakers	$20
Batteries/Power Management	$20
PCBs	$270
Total	**$550**

Actual	
Equipment	Cost
1 Microcontroller Evaluation Board (1 borrowed)	$170
2 CC1000 Plug and Play Modules	$200
Miscellaneous: Parts, shipping etc.	$492.30
Total	**$862.30**

As one can see, the actual cost ended up being substantially more than what we had anticipated in our initial project proposal. There are several reasons for this:

- Many of the parts that we ended up using were not finalized by the time of the project proposal – thus the projection of how much those items would cost was underestimated.
- We decided that the best move was to purchase 2 ready made transceiver modules from Chipcon as a functional reference for our own boards. The cost of these parts was substantial.
- Low volume purchases ended up being much more of a burden than we had expected because of the very large markup on such purchases.
- Not accounting for shipping cost ended up being a very large factor in our underestimation.

Fortunately, we were able to marginally offset those extra costs by borrowing an Atmel STK500 development board that we would have otherwise needed to purchase.

3.2 Scheduling

A Gantt chart summarizing time spent on various tasks is included below. As a comparison, the original time scheduled for each task as per the initial project proposal has been included, each of which is followed by the actual time used for the corresponding task.

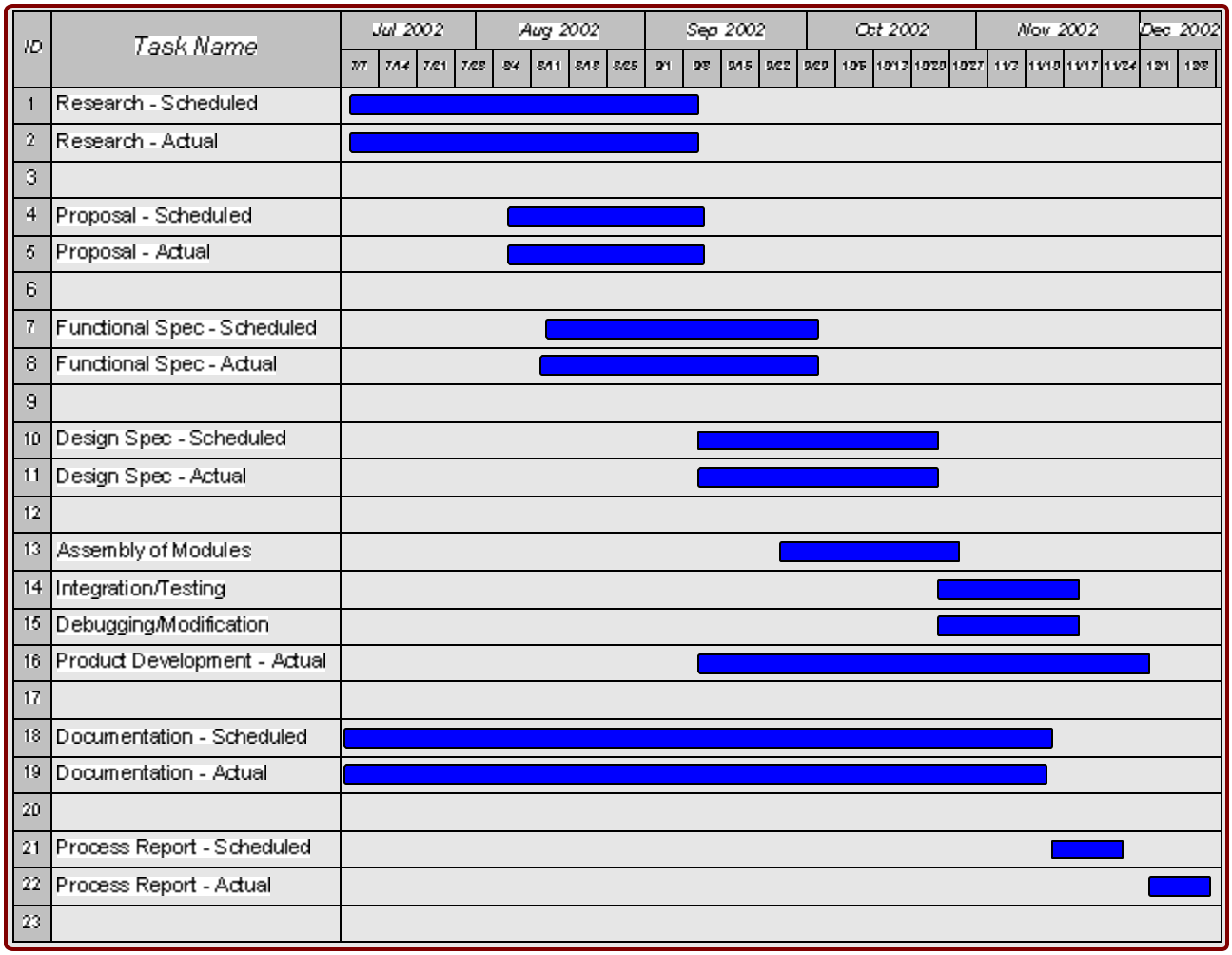

Figure 3 – Gantt Chart Comparing Projected and Actual Schedules

One major departure from our original schedule was the conglomeration of the "assembly of modules", "integration/prototype testing", and "debugging/prototype modification" design tasks into a single task labeled "product development". This was done to reflect changes in our design approach that resulted in all of the aforementioned tasks happening concurrently.

On the whole, the team did an admirable job keeping to schedule. The only deviations in terms of the planned schedule to note was the development time taken at the end, and the change in time scheduled for the process report. However, the extra time taken for development was mostly touch up work, and the change in the process report schedule was simply shifted to accommodate exams.

4 Group Dynamics

Secure Solutions was fortunate enough to have been formed by a group of responsible, motivated individuals. The fact that we stuck to our timeline and completed the project as scheduled was extremely satisfying and could not have been pulled off had we not had the commitment to do so. Our successful implementation of this project, from initial concept to final product, is the best proof of how well our group managed to work cohesively. We are a group of students with varied backgrounds and experiences who have pooled our talents together to strive for a common goal. The result was sweet success.

Nevertheless, no team of 5 students working together for three months could have passed that time without an occasional hiccup. There were times of disagreement over design issues that had led to some interesting shouting and high blood pressure events, but that was a natural process that eventually healed itself with group consensus and even a few chuckles. All in all, it was relatively smooth sailing for us on the group dynamics side. This is in no small part due to each individual's motivation to work hard and complete the project on schedule, regardless of any differences in each members' working habits and availability to spend time on the project.

Fortunately, the main sources of stress were technical problems encountered throughout the development cycle. All such problems were properly and expeditiously addressed because of the unique skills that each member of the team was capable of contributing. We have learned (the easy way) that for any team to be successful in the undertaking of any sort of engineering project, each member must possess some unique skill or knowledge that they can contribute to the team, as this adds depth to the design and implementation process and makes troubleshooting easier. However, by far the best attributes that make for excellent group dynamics is individual motivation, tolerance, and the ability to make compromises. We would never have completed the project on schedule had not everyone shared the same vision and goals.

5 What We Would Do Differently

Overall, the Infiltrator turned out the way we envisioned it from the start. A few areas with the design of the Infiltrator that could have been done differently are the voltage regulation and the siren.

Right now we operate the remote at 3.4 Volts using 3 1.2 Volt AAA batteries in series and use a 9 Volt battery for the console with a simple voltage regulator to cut the 9 Volts to 3 Volts for the CC1000 and the microcontroller. The 9 Volts is used or the console because that is the voltage we need to run our piezoelectric siren. So in fact, the siren we use will determine what kind of voltage (regulator or batteries) we need to use. So using a siren that operates off of 3 Volts would be ideal so then no voltage regulator is needed so no power is wasted with the regulator.

But a siren that runs at 3 Volts is most likely not going to be loud enough for our needs so we will have to look into a more efficient voltage regulator. A more efficient voltage regulation method is to use a Buck converter with complementary circuitry to cut the 9 Volts down to our required 3 Volts.

Our piezoelectric speaker is rated for 96dB at 10V operation but comes up way short of that with our operating circuit, which is the same as the application circuit for that speaker. Plus, when the casing for the console is shut and screwed together, it extremely dampens the sound to a point where the siren is ineffective. So a new operating circuit is needed for the speaker and the casing needs to be adjusted to allow for less damping of the noise.

6 Future Plans and Recommendations

We see a bright future for Secure Solutions and Infiltrator. We have already received a provisional patent from USPTO. Secure Solutions has been chosen to represent SFU engineering in Western Canadian Conference and Competition WECC. We are also in talk with UILO for incorporating or licensing the product. These are a number of achievements that we have made during the past four months and there is a lot more to come.
During the next four months we will try to improve the product and also make a great effort to find customers for Infiltrator. We will first try to get in contact with laptop manufacturers to present the potentials of the Infiltrator. Their interest in our product can help us attract investors to our company and helps us raise money. This money can be used for further research and development. Their requirements will make our product ready for a market niche. This enables us to form and alliance with a laptop manufacturer.
Secure Solutions is committed to provide security solutions for portable devices and we will not stop here. If successful we will move on to provide security systems for briefcases, backpacks, and even kids.

7 Individual Reflections

Vincent Au-Yeung

Working on a project from the brainstorming phase to the creation of a complete, successful, packaged product was the best education I have gotten so far. Being involved in every decision making process and discussion from the start let me share my views with the other group members and allowed me to learn from other people's ideas and experiences.

We were fortunate to have a great team to work with whereby everybody had unique skills to contribute. This worked out very well in helping our team avoid "people problems" so that we could focus 100% on the technical issues. The technical issues, although numerous, were ironed out one by one throughout the course of the design and implementation period and is where the advantage of having a varied team with a large pool of different skills to dip into really shines.

I learned about the need to identify and rank your priorities, otherwise the "feature list" grows to unmanageable proportions and would consequently lead to frustration and much stress for

everybody, further aggravating the situation. As with all engineering activities, nothing ever works the first time around despite your most optimistic predictions. There will always be bugs to work out in any software and unforeseen problems being discovered throughout the design process. One thing that I also learned to understand and accept is that if there is a one in a million chance of something going wrong, it will happen sooner or later, and it is typically the case that it would happen at the worst possible time, such as during a demo. As a result, as engineers, we should never ever sweep anything under the rug and pretend it never existed because out in industry, we have the potential of changing people's lives.

I also found that planning your work out and setting milestones is the only way to keep everybody focussed on the task at hand. For our product, we first setup the microcontroller circuit and the RF transceiver circuit. That was followed by getting the 2 to talk to each other. Once that was solidly implemented, we spent the rest of the time coding our unique algorithm to implement the laptop security device functionality. The great thing with our particular design process was that everyone got a chance to do a bit of everything so that nobody was every left out or given hand-me-down tasks that nobody else wanted to do. As the project was software oriented, I spent a lot of time writing and debugging code, which was suitable for me as I had experience using our particular microcontroller from a previous co-op. I also got a chance to do some soldering and hardware work towards the end of the project as we were moving everything off the breadboard and evaluation boards to a perforated board which we could package into a neat little plastic box and run off of batteries. Being involved at every step of the project, from the ideas scratched out on paper to the coding and the soldering, allowed me to learn about the integration and interfacing between hardware and software.

I feel that our success was entirely a result of our motivation and desire to get this great idea to work. We shared a common goal and worked many long hours to arrive at our final product, which is no trivial accomplishment. I thoroughly enjoyed working with these great people and would do so again.

Matt Brown

After hearing the dreaded stories of past ENSC 340 students, I prepared myself for a grueling 3 months in the lab. We came up with a project that involved RF, which none of us knew too much about. I liked the idea of working with RF because everything is being converted to wireless technology and so working on a project with basic RF is a great learning block.

I helped out a bit with the C programming on the Atmel microcontroller, but my main contributions were designing and implementing the hardware, creating the timing algorithm, and high level coding ideas. With the hardware, I improved upon my surface mount soldering skills in building up the Plug&Play modules for the CC1000. Also, I gained experience with transistors and piezoelectric speakers for low power noise generation. The timing algorithm was an extremely important component of our project. Since our remote and console are periodically sending and receiving data, we wanted to sleep both the microcontroller and CC1000 while we did not need them, so we could minimize our battery consumption.

Looking back, the RF was straightforward for us, since we just had to communicate with the CC1000 Plug&Play module with the microcontroller and set the registers in the CC1000 to make it do what we wanted it to do. The hardest part of our project was implementing the timing algorithm into our project. Numerous hours were spent in the lab, trying to debug and test the operation of the algorithm.

Working in a group on a large project took a lot of communication and group design. We made sure everyone understood and agreed with what we wanted to accomplish for our idea. Since programming was a large area of our project, it was difficult to split it up into sections since everyone programs differently. So we worked on the programming together most of the time to make it easier to understand but slowed the process down quite a bit because 5 guys around 1 computer never works.

Overall, we were able to get our device to do what we wanted it to do from the start. The Infiltrator automatically arms and disarms, senses motion of the console, alerts the user when the laptop is moved, and sounds a siren when it is not a false alarm. The power consumption and decibel output of the siren are 2 areas that I would have like to improve upon if we had more time, but overall I am extremely pleased with the outcome of our project for ENSC 340.

Steve Lau

I think the most valuable thing I gained from this project is the experience in designing a wireless device. There are a lot of intricacies and considerations that aren't immediately apparent for someone undergoing such a project for the first time, especially from the timing and synchronization side of things. It definitely gave me more appreciation for complex wireless applications like cell phone and paging networks. I have been involved in BOTH of these types of networks from previous work terms and I found that I still have a lot to learn, especially at the lower design levels. Besides timing, many other physical issues also needed to be considered such as power management and the unpredictable nature of RF.

In retrospect, it this was a great project for anyone wanting to learn more about wireless. Having to design both the receiver and transmitter meant that we were free to design our own communications protocol. The freedom to do so opened up a whole new can of worms, problems to solve, and experience to be gained that wouldn't have been there if we were to simply comply with a pre-defined standard.

Another thing that valuable was the experience in working with the various bits of hardware in our system, especially in the areas of interfacing, process control and scheduling. In the past when I have work in the industry, things like IPC, timing, and scheduling were already taken care of by the time I joined the project. So now, having to schedule various "pseudo" tasks in a real time type of application using the limited timing resources of the microcontroller while having to deal with internal and external interrupts that were essentially unpredictable became rather – interesting – to say the least. This was another case where building our system essentially from scratch, really helped to increase understanding and appreciation of what happens in larger systems using these types of components.

On the non-technical side of things, I found that this was yet another lesson on "why not to procrastinate", but from the other side of what I'm used to. The non-procrastinating nature of our group really allowed us to finish 340 while keeping the ulcer count to a minimum. I really believe that we probably wouldn't have finished this project had we waited until the last minute because of not only all the time required to acquire parts and such, but also the conceptually challenging aspects of the project really didn't lend themselves well to being worked on by tired minds at 4:14 in the morning.

As always, working in groups has allowed me to learn about a couple new personalities, and learn more about a couple that I already somewhat knew. It really opened my eyes to the necessity of having a well-rounded group not only from the standpoint of engineering skills but also from the aspect of personalities. A conflict in ideas can cause a heated "discussion" (read: argument) between two or more "strong willed" (read: stubborn) people where the misunderstandings just seem to build upon each other. In these types of situations, it becomes imperative the group have one or more objective people who can diffuse such arguments.

For the most part however, our group communicated quite well. Group members always made a point of making sure the other members of the group knew what they were working on via email. Advancements in source code were always sent out to the group mailing list at the end of the day. Overall, we had a variety of personalities on this team, and I'm pleased that they all managed to work well together. I think the strongest asset of the group was the motivation level. The high motivation really brought the stress level down – at least for me, because I knew that we would get the project done.

Hani Mehrpouyan

Let me just say that I had the best time working with my group members and friends on this project. It was a pleasure and a great learning experience working with all of these senior engineering students. However the best part is that we are still a team and looking ahead to the future trying to make a difference in people's lives. All the experiences gained here are going to make Secure Solutions a better company and Infiltrator a better product.

We all worked together and we made sure that everyone is aware of the sate of the project. I think that was one of the biggest reasons that we were successful since at any given day any of our group members could pick up from where we left off the previous day.

Our plans went quite well. We were able to notice the major obstacles in the design and work around them and believe me having four brains thinking at the same time really helps. By tackling the obstacles one by one we were able to stay focused and make consistent progress.

But nothing is as important as having a great team. Everyone put their best effort to make sure we can achieve the best result. Nothing distracted us from our main goal and the final outcome of our project comes to show that hard work pays off.

Let me say congratulations to every single one of my group members.

Chris Mitchell

As sick as it sounds, this was my favourite semester yet. I think our group was so overcome with warnings and fear of not completing our project that we were quite diligent in starting research early in the summer before the course even began. This allowed us all to have a sound understanding of our project and goal and to begin very early in the semester and stay ahead of schedule. It was definitely an excellent experience being able to apply all the theoretical knowledge, we had supposedly gained in the past 4 years, for a real-world solution and then add so much more new understanding as well.

In terms of technical knowledge, I think it worked to our advantage to have all group members be involved in all facets of the development and knowing the state of the system. I think most of all, it will be a great asset to have learned how to build and operate a state-of-the-art transceiver module (Chipcon CC1000) that can be applied to a number of different applications. We became familiar with Atmel's line of microcontrollers to communicate with the transceiver and we had to work with several interrupts (from motion sensors to different clocks and data lines to/from the transceiver to timer interrupts) all in a real-time environment where speed and size are critical. It will also be of great importance to have learned the advantages and disadvantages of working on a rather large software project where people need to integrate individual parts and styles into one final system.

Many other valuable skills were picked up including managerial, interpersonal, and various non-technical skills such as sourcing parts and where to find samples and funding. I think our group worked well together because we have a wide range of differences and thoughts and, when discussed civilly, great ideas were developed.

Appendix – Schematics

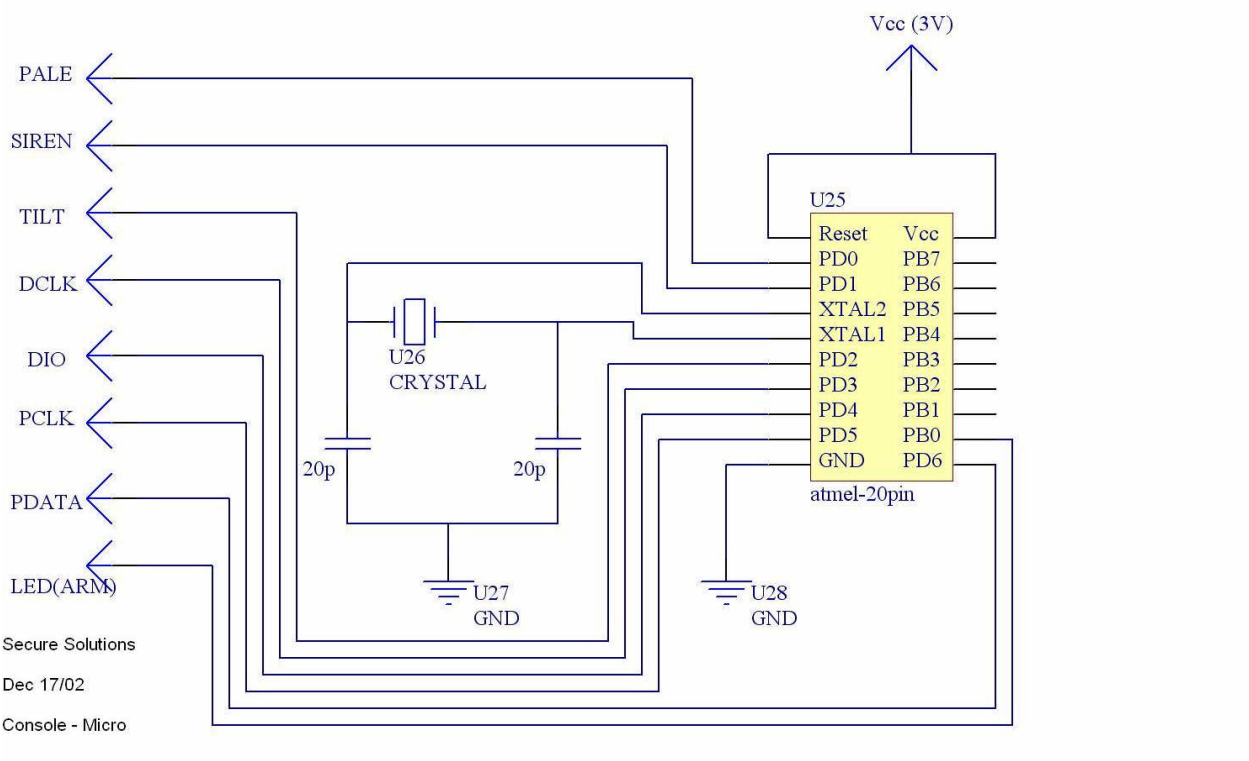

Figure A_1: Base Unit Microcontroller and Transceiver Interface

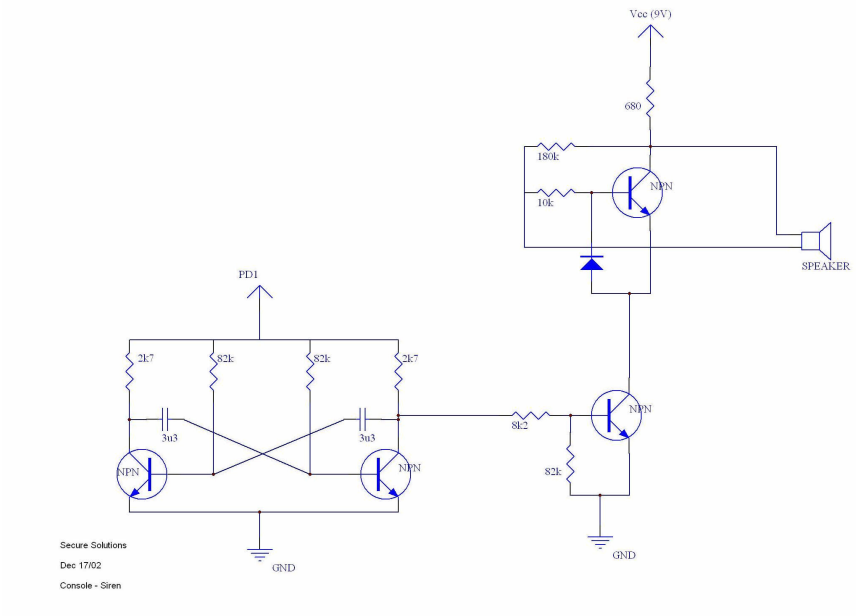

Figure A_2: Base Unit Siren Design

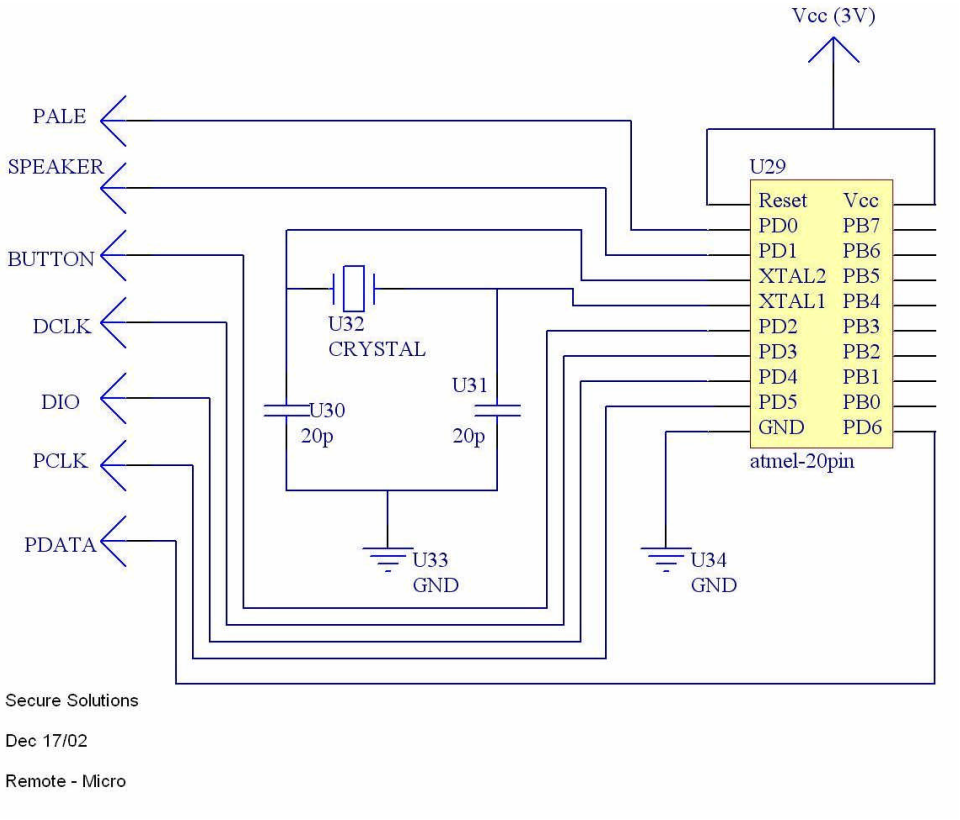

Figure A_3: Remote Unit Microcontroller and Transceiver Interface

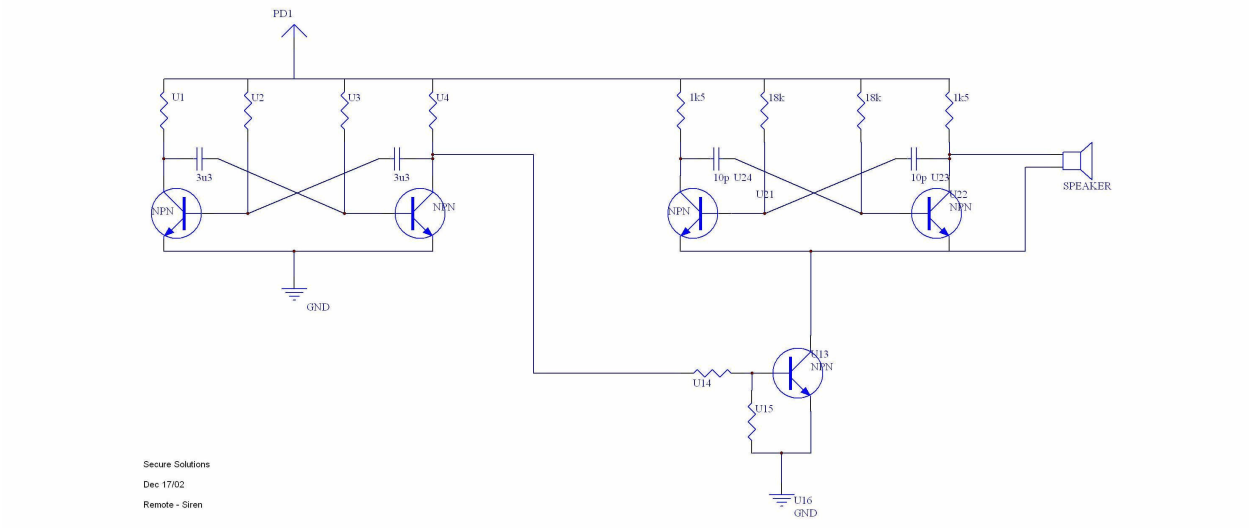

Figure A_4: Remote Unit Siren Design