Undent Solutions
Simon Fraser University . Burnaby, BC . V5A 1S6

April 28, 2010

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

RE: ENSC 440 Post Mortem for the Vehicle Lock-Out Prevention System

Dear Dr. Rawicz,

It has taken us about four months, and now our project has reached its final stages of completion. Countless hours of hard work have gone into our final version of the Vehicle Lock-Out Prevention System. Attached is the Post Mortem document for our product.

The document will explain the current state of our product, deviations from the plans, future plans, budgetary and time constraints, and each person's reflection on their experiences working on the project.

Undent Solutions is composed of Marissa Hun, Daphne Mui, Dona Patikiriarachchi, and Elisa (Xuan) Lu. If you have any questions or comments regarding our Post Mortem, you can contact us through email at mmh2@sfu.ca.

Sincerely,

Daphne Mui

Daphne Mui
CEO
Undent Solutions

Enclosure: Post Mortem for the Vehicle Lock-Out Prevention System

Vehicle Lock-Out Prevention System Post Mortem Report

Project Team:
Daphne Mui
Marissa Hun
Dona Patikiriarachchi
Elisa (Xuan) Lu

Contact Person:
Marissa Hun
mmh2@sfu.ca

Submitted to:
Dr. Andrew Rawicz (ENSC 440)
Steve Whitmore (ENSC 305)
School of Engineering Science
Simon Fraser University

Date: April 28, 2010
Revision: 1.0

TABLE OF CONTENTS

1. INTRODUCTION

The Vehicle Lock-Out Prevention System (VLOPS) had been under development for the past four months and the project has now come to an end. This document reviews the current state of our system and makes note of the deviations from our original planning stages. We also discuss future plans and observe our budget and time constraints. Finally, each member of the team reflects on their technical and inter-personal experiences.

2. CURRENT STATE OF THE DEVICE

As explained in the project proposal, The Vehicle Lock-Out Prevention System by Undent Solutions alerts the user of possible lockouts and unlocks the main door in the case of a lockout. The system overview of our product is shown in Figure 1 below:

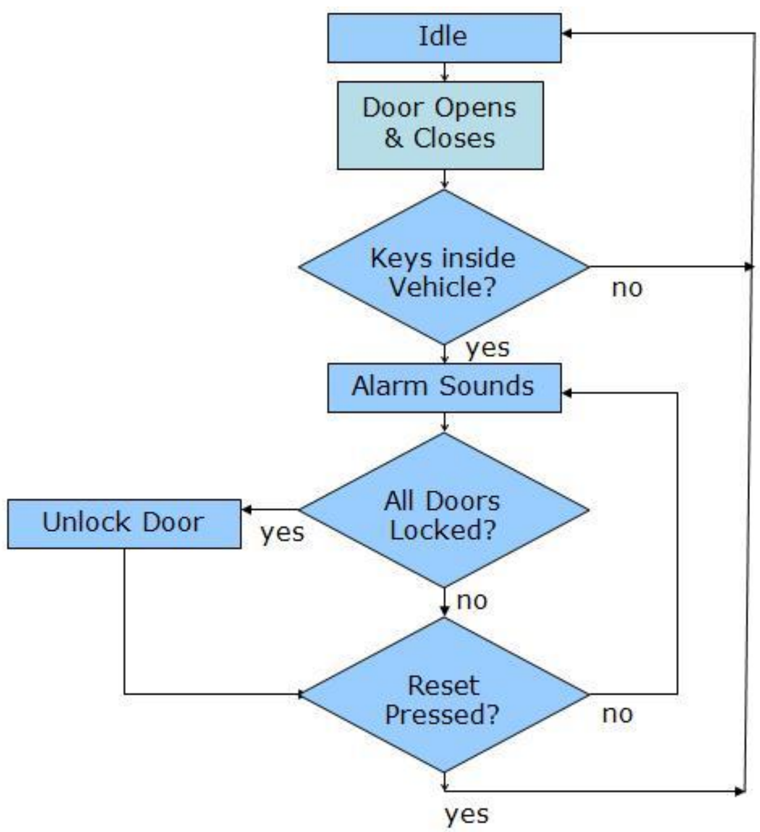

Figure 1: System flowchart

The current state of the device will be explained by examining all the stages outlined in Figure 1.

2.1 Microcontroller

At first, all doors are closed and the system is in idle state. Once a door opens, the push button sensor attached to the door frame is released and this creates a software interrupt. The ISR tentatively stops the current execution stage of the microcontroller and waits until all the open doors are closed. Once all open doors are closed, the microcontroller powers up the two RFID transceivers.

Upon power up, the two RFID transceivers begin initialization and starts looking for nearby authorized RFID tags. Once the RFID transceiver locates and authorizes the RFID tag, the RFID tag sends signal strength data to the RFID transceiver. The RFID transceiver then processes this data and determines if the RFID tag is inside the vehicle. If the RFID tag is detected inside the vehicle, the RFID transceiver sends a digital high signal to the microcontroller otherwise, the transceiver sends a digital low signal.

If the microcontroller received a digital high signal from any of the transceivers, it means an RFID tag attached to the keys is detected, and the buzzer goes off immediately. If no tags are detected within 6 seconds, the microcontroller goes back to the idle state and waits for the next time a door opens.

Once a registered tag is detected and the buzzer goes off, the microcontroller checks the status of the door lock sensors. If all doors are locked, the microcontroller powers up the door lock actuator which would unlock the driver's side door and goes back to the idle state. If all doors are not locked and the reset button is not pressed/a door is not opened, the microcontroller keeps checking for locked doors. This is to prevent the case where some of the newer cars lock the doors automatically when left unlocked for about 1min.

We noticed that the push buttons we were using for the door sensors and door lock sensors are bouncy and therefore trigger false alarms. In order to minimize noise, we averaged the inputs from the sensors, added capacitors to drain noise and added delays to stabilize the signals.

After many weeks of coding and testing each individual part of the system such as the buzzer, sensors, RFID transceiver and tags, door lock actuator, etc. with the microcontroller, we integrated all the parts and tested the program. We ran multiple test scenarios and noted that the entire process takes less than 10sec to complete.

2.2 Unlocking mechanism

The unlocking mechanism is controlled by an H-bridge circuit which allows it to spin in either direction. The H-bridge also allows the actuator to be powered by a separate power source. We found that a 9V battery can be used to power the mechanism.

2.3 Sensors

We used momentary push buttons for all the sensors in our system, such as the door and door lock sensors. In order to detect the door open-closed sequence, we mounted the push buttons on the bottom of the car door where it makes contact with the car door frame. The second set of push buttons are used to detect whether the door is locked. This set of push buttons are only mounted on the two front doors.

2.4 Low battery indicator circuit

The low battery indicator circuit is used to detect the battery level of the microcontroller. The microcontroller uses a 9V battery and performance decreases when the voltage level drops below 5V. When the battery level of the microcontroller drops below 5V, the circuit turns on an LED warning the user the battery is low and needs to be changed. The LED is conveniently located on top of the system unit box.

3. DEVIATION OF THE DEVICE

3.1 Overall System

The functionality of our Vehicle Lockout Prevention System adheres to how it was described in the Functional Specifications document. Some minor software and hardware modifications were made to make the system more efficient and easier to use. The modifications are outlined in the following sections.

3.2 Microcontroller programming

Programming of the microcontroller has slightly changed from that described in the functional specifications document. We used 2 analog pins as the inputs from the door sensors instead of digital pins because the digital signal received from the buttons were too noisy and set off false alarms. We also found out that the microcontroller already does Analog to Digital (A/D) conversion. Therefore, we used 10 digital pins instead of 12 as described in the functional specification document. We also made use of an ‘OR gate’ to OR the two signals received from the door sensors that would trigger an interrupt to the microcontroller whenever a door opens.

The Override function was renamed to ‘Reset’. The microcontroller will no longer check for override. Whenever the reset button is pressed, the microcontroller will turn off the buzzer if it is on and jump to the beginning of the program (idle state shown in Figure 1). The reason for the name change is to better suit the purpose of this function.

The RFID transceivers are powered by the microcontroller to prevent them from being on all the time. This saves energy and simplifies the User Interface (UI) as there will no longer be an additional battery level indicator for the 2 RFID transceivers.

We did not make use of any Pulse Width Modulation (PWM) signals as the motor could be programmed with ordinary digital signals. The motor and the gear train were replaced with an actuator that was already available on the market. Two additional signals for the actuator were used for an H-bridge which controls the actuator motor and allows it to be powered by a separate power source.

With the above changes, the microcontroller had 12 Input/Output (I/O) pins instead of 13 as described in the functional specification.

We completely removed the scheduler from the microcontroller program. The main process was used to decide which function to call through condition checks and flags. The reason for this change is to simplify the code. An Interrupt Service Routine (ISR) was created to trigger an interrupt whenever a door opens. Therefore, the ISR was given the highest priority instead of the scheduler.

The microcontroller no longer serially communicates with RFID transceivers. Instead, it will wait for a digital ‘1’ or ‘0’ on its dedicated input pins. A ‘1’, digital high, means a tag is found and ‘0’, digital low, means no tag is found within 5sec of transceiver activation.

The buzzer will only play one tone when a tag is detected inside the car. We could add a sequence of tones, but due to time constraints we used one distinct tone for the buzzer. The buzzer can be stopped not only by hitting the reset (previously override) button, but also by opening a door. This additional feature was introduced to increase the ‘smartness’ of the system since if a door is opened, it means the user has access to the keys that are inside the car and therefore beeping the buzzer is futile and irritating.

The process flow of the Vehicle Lockout Prevention System remains the same except the microcontroller no longer checks for an override/buzzer off signal. Therefore, this state was ignored and the process flowchart was modified as shown in Figure 1.

3.3 RFID transceiver and tags

During integration and testing with the microcontroller we found that the RFID tags can connect with only one RFID transceiver at a time. To connect to the other RFID transceiver, the RFID tag and both transceivers must be powered cycled. After the RFID transceivers and tag have been power cycled, they will re-initialize and the RFID tag can join another RFID transceiver. This was a problem to our initial design because we needed the RFID tag to be continuously active and sending data back to both RFID transceivers. The workaround for this problem was to power cycle both RFID transceivers every time the microcontroller detects the open-closed sequence of the door. The RFID transceiver will initialize and start looking for a nearby RFID tag. Once the RFID tag is located and authorized by the RFID transceiver, the RFID tag will send its signal strength to the RFID transceiver for processing. Moreover, as in our initial design, the RFID transceiver will send a digital high signal to the microcontroller if it detects the RFID tag inside the vehicle and a digital low signal otherwise. Once the microcontroller receives this information, the RFID transceivers will be powered down and the RFID tag will reset. At the next open-closed door sequence, the RFID tags will be actively searching for a nearby RFID transceiver to join and thus, continuing the cycle.

3.4 Sensors

Momentary push buttons were part of our original design for both the door and door lock sensors. Through testing and integration, we found out its sensitive enough and can be easily programmed with the microcontroller. The location of the push buttons that detect whether the door is locked was slightly modified. The original location was smaller than we anticipated and the button did not fit properly. To account for this, we relocated the push buttons and mounted it below a clamp that connects the actuator used for the unlocking mechanism and a metal rod that is connected to the door lock. When the door is locked the clamp will push down on the push button.

3.5 Unlocking mechanism

We decided that it was more cost efficient to buy a pre-made unlocking mechanism rather than to create our own. After experimenting with several small motors, we found that the cost of a more powerful small motor would be considerably higher than simply purchasing a third party door lock actuator. The durability of the third party motor will need to be more thoroughly tested before taking the system to market.

4. FUTURE PLANS

The Vehicle Lockout Prevention System by Undent Solutions can be improved to best fit the needs of its customers. As we look back on this already built system, we suggest the following for future development.

4.1 Overall System

- *Reduce packaging size.*

  Currently, the microcontroller unit, the UI, the H-bridge circuit for controlling the actuator, the batteries and the low battery indicator circuit are packaged in a 12.5cm x 6cm x 10.5cm plastic box. By rearranging the components to minimize feature size, we hope to further compact the package so it will not take too much space on the dashboard and will be less intrusive to the user.

- *Use car battery to supply power to the microcontroller.*

  We can add the option of powering up the microcontroller with the car battery instead of an external power supply. This would further reduce size of packaging and eliminate the need to change batteries. A car battery outputs 12V DC and the microcontroller's voltage input (V_{in}) range is 5-12V. The microcontroller can be either connected directly to the car battery or passed through a Buck Converter to slightly step down the voltage.

- *Eliminate false alarms.*

  We have tried very hard to reduce noisy signals and minimize false alarms caused by the sensors and RFID transceivers. Although the system is very stable, there is still room for improvement. By modifying the code with conditions to filter out noise and adding appropriate delays, we will be able to eliminate false alarms.

4.2 Sensors

- *Utilize the sensors already installed in the car.*

  Currently, we are using our own door and door lock sensors. In the future, we can tap into the door sensors and door lock sensors already placed in the car and route these signals to the microcontroller. This will reduce the cost implementation and minimize wiring.

4.3 RFID transceivers

- *Use a different RFID transceiver.*

  The RFID transceiver and tags we chose do not have an enable pin that can be programmed for turning the transceiver on/off. This option would come in handy for our system because the microcontroller would not need to power the transceiver on/off.

4.4 Buzzer

- *Add a tone.*

  The buzzer outputs one distinct tone to warn the user of a possible lockout. In the future, we can add a sequence of tones to make the warning less irritating. We can further add the option of choosing from a set of pre-programmed tunes so the user has more of a selection.

4.5 Actuator

- *Tap into the actuator of the door lock.*

  This will only apply to cars with power locks. If our system is installed into a car with power locks, we can tap into its actuator and feed the signals to the microcontroller and power supply to perform the unlocking mechanism. This will reduce the cost of implementation and minimize interference inside the car door.

- *Use car battery to supply power to the actuator.*

  If our system is installed into a car without power locks, we must use an actuator to automatically unlock the main door in the case of a lockout. Currently, we are using a 9V battery to supply power to the motor inside the actuator. However, the motor can also be powered by a 12V battery. Since this is the case, we hope to connect the actuator to the 12V car battery instead so that we can reduce the size of packaging and increase efficiency.

5. BUDGETARY AND TIME CONSTRAINTS

The following table outlines the estimated and actual costs of our project as of April 27, 2010:

Table 1: Estimated and Actual costs of the Vehicle Lock-Out Prevention System

Component	Estimated Costs	Actual Costs
Microcontroller	$12	$37
RFID transceiver	$518	$57
RFID tags	$6	$50
RFID USB debugger	-	Borrowed
Actuator	$15	$30
Car door	$50	$20
Car lock	-	$0
Miscellaneous	$10	$151
Total	**$611**	**$345**

The miscellaneous costs include components such as the batteries, H-bridge component, LEDs, OR gates, wiring, PCB board, buttons and door frame supplies. Our miscellaneous actual costs are higher than our estimated costs because of unexpected ordering of extra parts.

Our actual costs are much lower than our projected costs because we were able to borrow and obtain many of our components free of charge. We received $400 from ESSEF funding, leaving $55 of surplus remaining funds.

The following figure outlines the Gantt Chart we proposed at the beginning of the semester.

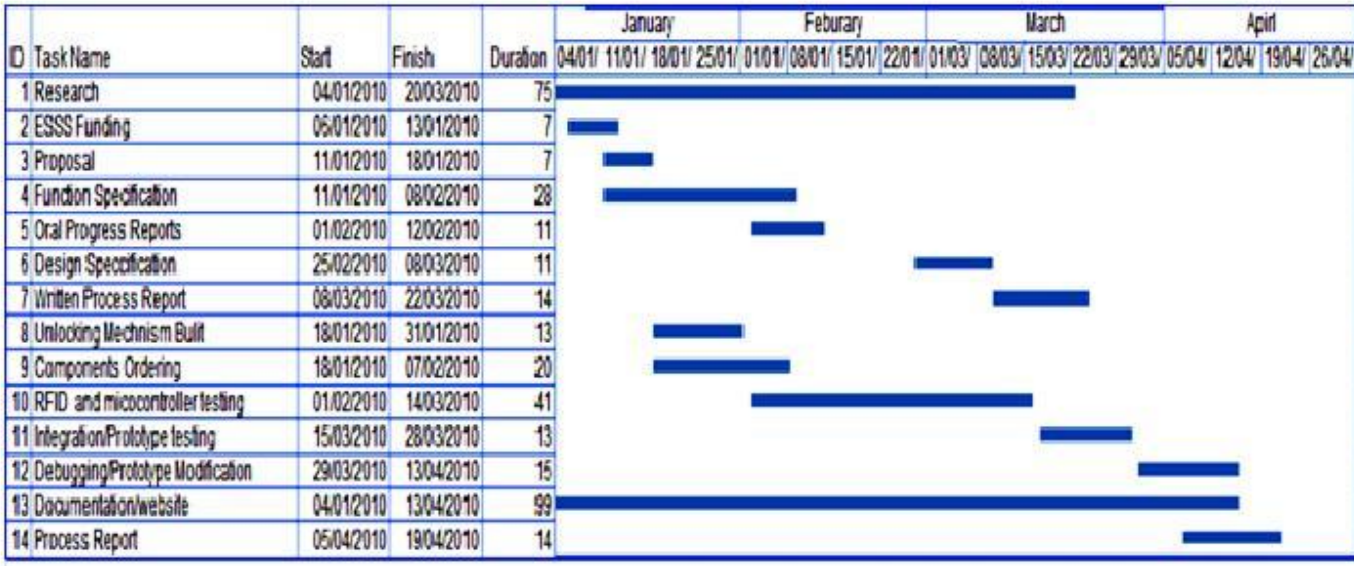

Figure 2: Gantt Chart

The expected duration for the written documents and research was accurate. However, the unlocking mechanism was not built as early on as we anticipated and was instead built at the end of February due to researching and locating appropriate gears and motors. The component ordering was done throughout the semester as needed, but we ordered our most important parts in late January to avoid unexpected delays. This was the case for the RFID transceiver and tag orders and because of this delay, we were fortunate enough to borrow another previous 440 group's RFID USB debugger. Thus, our RFID transceiver and tag programming and testing were on schedule. The microcontroller programming was done in early March. Integration of the individual components to the microcontroller took longer than expected to complete. This was due to the amount of time spent on debugging both the RFID transceiver and tag and microcontroller software. We anticipated that the bulk of our time would be spent on integration and debugging but, the tasks that we expected to be completed easily ironically took the longest time to complete.

Despite these small delays, we were still able to adhere to the schedule mainly because of our weekly meetings. In our weekly meetings, we discussed actions to be completed and followed up at the next scheduled weekly meeting. We also discussed the current progress of the project and took into account potential delays to the project due to midterms and other course projects.

6. INTER-PERSONAL AND TECHNICAL EXPERIENCES

Daphne Mui - CEO

For me, the purpose of this project was entirely for learning. Probably the most important thing I learned was to expect the unexpected. Everything will take at least three times more time than expected. The "easiest" jobs will turn out to be the hardest. Parts are cheap and easy to find, unless you really need them. Things that should happen in theory often do not happen in practice.

Although I expected to do more programming, my role in the team turned out to be more about finding flaws in design ideas and encouraging my teammates. They are all very capable engineers, and I had a lot of fun working alongside them. As the CEO, I gained better leadership skills. It was an odd position to be in, considering I've been friends with the members of my group since first or second year. I learned to be confident in my decision making and more efficient in time management. I believe that my communication skills have also improved. Being able to communicate what I expected from my teammates was vital to getting work done efficiently.

I spent much of my time working on the unlocking mechanism that we did not actually end up using (due to cost effectiveness and pure performance). Although it seems like a big waste of time, I learned a lot from this. A lot of the things we spend a lot of time on may not end up in the final product, but it is sometimes necessary to spend time on things that don't work in order to discover things that *do* work. Besides my own part of the system, I also spent time learning about everyone else's parts of the system. This helped with the integration of the overall system and kept everyone on track.

Marissa Hun - CFO

After working on this project for the past 16 weeks, I've learned overall the amount of complexity, planning and time that goes into designing and building a product. I learned extensively how RFID transceiver and tags operate. I dealt mainly with the software of the RFID transceiver and tags. From the software programming, I have improved on my C++ coding and debugging skills.

I have worked with my team members before in previous project oriented courses so, I am already aware of what their strengths and weaknesses are. Throughout the duration of this project course and long and countless hours in the lab, I learned that communication is the key to a successful and productive team. Each team member brought their own expertise and suggestions to the project. Before this course, we already got along very well and I can safely say that we are all still friends. In the future, I would not hesitate to form a group with my fellow team members as they are all hardworking, responsible and easy to get along with.

From this project course, I learned that discussing issues and problems as a group and in person rather than individually through email or instant messenger is more effective and efficient. We were able to solve and troubleshoot most of the problems through strictly scheduled weekly meetings. It is also very important to listen to everyone's different ideas and approaches to the design and methodology. Furthermore, I learned that the more in-depth research and time spent on perfecting the design in the early stages pays off later on. This is because any potential problems that might have taken only a couple of hours of re-designing could easily take a couple of days of working around later on down the road.

Dona Patikiriarachchi - VP Operations

My idea of ENSC 440 was sacrificing life and fun and living in the lab for 13 weeks. However, I soon realized that I was terribly mistaken. I got to work with 3 amazing individuals who are intelligent, dedicated and hard working. I was able to put my software and hardware skills into practice, try out my own ways of defining problems and coming up with solutions as well as realizing the importance of communication.

I took the responsibility of programming the microcontroller unit. At first, I was slow at programming as I had to learn the Arduino software and the use of its Application Programming Interface (API) functions. I spent many hours browsing through examples and started automating each individual component of the Vehicle Lockout Prevention System. I started with blinking Light Emitting Diodes (LEDs), then, beeping the buzzer and testing LEDs and buzzer with button pushes. I spent a lot of time figuring out how to implement the door open/close sequence. After much research, I decided to use interrupts to indicate that a door is open. One big mistake I made was writing so much code without actually testing to see if it works. After completing the ISR and the rest of the program, I was disappointed to see that nothing works. At that point, I decided to break the code into smaller parts and test which sections are working and which are not. I had to reprogram the door sensor buttons and use analog pins instead of digital as the signal strength of the digital pins were not strong enough for the microcontroller to detect. I joined the Arduino online community and started asking questions on interrupt handling. I was overwhelmed by the number of responses I received with so much advice and suggestions. Taking the advice I received through Arduino community forums and my own imagination, I reprogrammed the ISR. I wrote functions for each state of the process such as buzzer operation, door lock sensors, communicating with the RFID transceivers, etc.

Upon successful completion of automating individual components, I finally put everything together and started testing. Although not smooth at first, I was delighted to see progress. After countless hours of debugging and button pushing to the point my fingers were numb, I was able to successfully implement the microcontroller software.

Working as a team was a crucial aspect of our project. I realized the importance of listening to my team members and asking for help when help is needed. My group has been incredibly supportive and reliable. The weekly meetings gave us the chance to discuss what everyone has been up to and what is to be done. I am happy to say that we are still friends and I am more than willing to work with them in the future as well.

Through ENSC 440, I was able to improve my programming skills, time management as well as communication and interpersonal skills. I was also able to put the knowledge I gained from previous courses, especially electric circuits courses and previous co-op experience into practice. All in all, it was a challenging, yet, very rewarding experience.

Elisa (Xuan) Lu – VP Marketing

I originally thought I would be having a stressful and terrible life living in lab1 for 13 weeks. In fact, thanks to all my team members, we had an enjoyable and memorable time in the lab instead. We’ve worked together for other projects before and we know the personality and skills of each other very well. My role is rather flexible in this project: I was in charge of programming the communication between microcontroller and RFID at first, and then working on the motor and microcontroller communication.

First of all, I would like to thank all my team members for their dedication to the project and their on-going support and help throughout the semester. Without them, it is unlikely for our team to operate as smoothly as it did. Like most senior students, we all have our own things to tend to and are busy with other classes. So it is very important to have your team mates stand by your side and lend you a hand once in a while. As a team, we are all willing to help others and make this project a challenging, rewarding and memorable time.

7. CONCLUSION

Our work on the Vehicle Lock-Out Prevention System has been a good educational experience for the team. As discussed, there have been some changes to the system design since the initial planning stages. We received a lot of unexpected help gathering parts which helped minimize costs. However, we also found unexpected problems which consumed extra time. Overall, we consider the project a success.