PERCEPTUM TECHNOLOGIES

RockIt Post-Mortem

April 16, 2008

Mr. Patrick Leung
School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, British Columbia
V5A 1S6

Re: ENSC 440/305 Design Specification for *RockIt*

Dear Mr. Leung,

The enclosed document is entitled *RockIt Post-Mortem*. The document will summarize the design details of our capstone ENSC 440/305 project based upon motion-controlled guitar effects. Our goal was to incorporate newer technologies with music to inspire guitarists to create.

This document will outline the current state of our product and detail some deviations from the original plan. Also within the post-mortem is a brief breakdown of our team, budget, and timeline. Some future plans for our system will also be discussed.

Perceptum Technologies consists of four engineers: Kyle Huffman, Daniel Galeano, Paul Carriere, Ben Shewan. Each team member is a well-accomplished engineer. If you have any questions or concerns about our product, please contact us through our email address: ensc440-spring08-perceptum@sfu.ca

Sincerely,

Kyle Huffman

Kyle Huffman

Director of QA, Perceptum Technologies

PERCEPTUM TECHNOLOGIES

RockIt Post-Mortem

Project Members: Kyle Huffman
Paul Carriere
Ben Shewan
Daniel Galeano

Contact Person: Kyle Huffman
ensc440-spring08-perceptum@sfu.ca

Submitted To: Steve Whitmore, ENSC 305
Patrick Leung, PEng, ENSC 440
School of Engineering Science
Simon Fraser University

Issued: April 16, 2008

Version: A.1

Table of Contents

List of Figures

1 Introduction

Since mid-January Perceptum has been planning to bring the concept of motion-controlled guitar effects to a reality. Throughout time, technology has made major influences on music. The electric guitar influenced the creation of rock n' roll. Then attributes to the guitar such as the tremolo and wah-pedal helped shape other forms of music. Jimmy Hendrix extensively used the wah-pedal in his ventures in psychedelic rock. Eddie van Halen extensively used his tremolo to change the pitch of notes and chords to produce a very hard rock sound. Technology doesn't end with the guitar and rock music. It extends into many genres. Hip hop might not exist if there was never a drum machine. Today, most new music that is played on the radio has been created through a variety of software. Seeing how technology is influential to music, we hoped to create another piece of equipment to influence music. So we created the *RockIt.*

2 Current System State

A system overview of RockIt is shown in Figure 1. This high-level diagram is composed two subsystems: the sensor processor and the MIDI controller. The sensor processor includes and accelerometer and four force sensitive resistors (FSRs). This module of RockIt also processes the sensor signals in order to send the status of the FSRs and the accelerometer to the MIDI controller. The sensors data sent from the sensor processor is mapped into user selected MIDI commands. These commands correspond to different effects, which are selected in the MIDI controller. The current state of the device is further explained in the following sections by going through the crucial components: accelerometer, FSRs, communications protocol and MIDI controller.

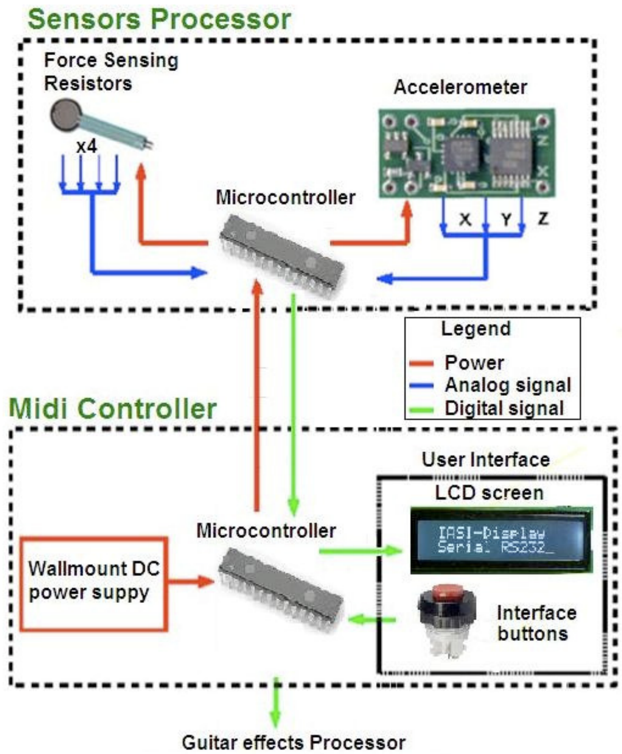

Figure 1 System overview

2.1 Accelerometer

To trigger and control various guitar effects, we use a 3-axis accelerometer mounted on the backside of the guitar head. This accelerometer reads both human and gravity induced acceleration. The three accelerometer signals are then used as two unique control mechanisms: tilt control and bump triggering.

We use the accelerometer and microcontroller to calculate the angle of the guitar with respect to the horizontal, then we convert the angle into a guitar effect like volume or wah-wah. Alternately, the user can bump the guitar neck with the heel of their hand. Once this occurs, the accelerometer/PIC combination can detect that bump and transform it into an appropriate delay effect. Both of these signals undergo filtering and signal processing to ensure that unwanted triggering is minimized.

2.2 Force Sensing Resistors

We needed a method to control the processing of the accelerometer and to quickly toggle sound effects. After cycling through many ideas we settled on using force-sensitive resistors. FSRs were low profile, reasonably-priced, and tunable. We could easily tune the sensitivity of the sensors and if desired, we could use them to control variable effects.

We use the FSRs to simply toggle effects and we have implemented Schmitt-triggering in both hardware and software. This was due to restrictions on available pins on our PIC18F2420. Secondly, we needed an effective way to mount the sensors on the guitar. Ideally, they would be embedded however we don't have the resources to manipulate a guitar. Instead, we created a sleek and low-profile housing using SolidWorks and SFU's 3D plotter. We were also able to implement feedback LEDs for the user. There were four FSRs in total and two were designated to toggle a large array of sound effects. The two other sensors were designated to toggle the tilt and bump control on the accelerometer.

2.3 Communications Protocols - Dan

The MIDI Controller is connected to the Sensor Processor using a serial link implemented with hardware UART functionality. Because this is a one-way link, it is important that there is some type of error handling for cases when there is distortion in the received signal. To provide for this, we send the current state of all FSRs every time an FSR update message is sent. In this way, any errors between the FSR states in the Sensor Processor and the MIDI Controller will be resolved once an error free message is received. Although we provided for this possibility, it should be noted that an erroneous message was never actually observed on this link.

The generation of MIDI signals was handled using the software USART module available with the PIC18F2420. Because MIDI signalling is based on current instead of voltage levels, a small amount of external circuitry was provided to make the necessary changes.

2.4 MIDI Controller

The final aspect of our system is what we call the MIDI Controller. The purpose of this module is to map control signals from the Sensor Processor into valid MIDI messages.

Because there are dozens of different effects that can be used with the PODxt Live, it is necessary to let the user determine those that they wish to control using *RockIt*. To accomplish this, an LCD and three buttons have been configured as a simple user interface.

By navigating a series of menus, the user will be able to dynamically control which effects are mapped to the control signals received from the Sensor Processor. For those who are not familiar with the effects available with the PODxt, we have provided a "preview mode". When the user is selecting their desired mappings, the effects that they are previewing are activated on the POD. In this way, the user can simply cycle through effects and listen to what that effect sounds like. This feature proved to be invaluable as it would be very inconvenient to set the mappings only to discover that you didn't like the sound of a particular effect.

3 Device Deviations

3.1 Usability

Unfortunately , it wasn't until the system was near completion that we realized the need for additional components to enhance system usability. These items include: activity LEDs for the MIDI controller, sensor processor and FSR enclosure, a tilt sensitivity knob to adjust the tilt/midi ratio, and a preview mode for the guitar effects. Although these items were not challenging, they were unforeseen and forced us to revise elements of our design.

3.2 Z-Axis / Bump Detection

Once the tilt control of the accelerometer was complete, it was apparent that we needed a more robust mechanism for z-axis accelerometer detection. We had expected to measure the back and forth motion of the guitar to toggle effects. Unfortunately, human motion is not capable of generating large accelerations(and hence signals) in this fashion. After some experimentation, we realized that we did not have to change the accelerometer, only its triggering method! Instead of moving the guitar, we realized that we could achieve the necessary output signal by simply bumping the guitar with the heel of the hand. Although the z-axis algorithm was virtually unchanged, we were able to design a more robust trigger by taking advantage of the high-acceleration, high frequency impact. This triggering method required us to increase the sampling frequency to 120Hz, as this would allow us to capture the signal peak.

4 Future Plans

One key aspect of *RockIt* is that we designed with room for potential growth. We would like our system to be an open-source product. There is also room for flexibility in how we used our sensors and there is room for improvement. We did revise segments of the product to make it more usable but there is room to make even further changes, as outlined below.

Bump-Control

As of now, we are controlling an effect called 'delay'. This is essentially an echo. The harder the user bumps the guitar, the larger the time interval is between delay pulses. We would like to have the time interval pre-set via a knob and the action of bumping the guitar is only used to activate and de-activate. This was a suggestion from a couple different test-users who tested our product.

Embedding Sensors

Instead of having our sensors external, it would look much better to have the sensors (accelerometer and FSRs) embedded in the body and neck of the guitar. This would require excellent craftsmanship but would also rid of the cabling that is currently hanging from the guitar.

Wireless Data Transmission

Ideally, we would omit our two processors altogether, and wirelessly link the sensors directly to a main processor. This would eliminate extra wires completely.

Improved PIC

There is also the potential of having more advanced motion detection implement with a more advanced microcontroller. We could use more sensors and we could even perform our own DSP on the sound effects instead of interfacing to the Line6 PODxt Live.

Open-Source Production

As mentioned before, we could sell the PCB to customers and allow them the freedom to decide what sensors they use, how they will be used, and where they want to place them. Perhaps this can be done as an attachment to a guitar or to be imbedded in a guitar.

5 Budget and Timeline

5.1 Budget

Table 1 compares the estimated and final cost of our project:

Table 1: Project Budget

Required Equipment	Estimated Costs	Incurred Costs
Accelerometer	$120	$120
FSR Sensors	$45	$70
MIDI-to-USB Bridge	$40	$45
PIC	$0	$25
MidiBox core module	$40	$40
Misc Electronics & enclosures	$55	$185
Total	**$300**	**$485**

The above table shows that we under-estimated the project costs by 60%. This discrepancy can be explained by our decision to design our own midi controller, rushed PIC orders, and our gross under-estimation of the costs of cabling, sockets, and enclosures. Regardless of money lost, the funds were spent in ways that contributed to the final success of our project. The most important and expensive project components, the instruments, were luckily supplied for free.

5.2 Timeline

Table 2 below compares the original and realized project development timelines. The timeline has been split to reflect the major sub-sections of the project.

Table 2: Project Development Timeline

Original Timeline		**Realized Timeline**	
FSR Development			
Finish testing & characterization:	Feb 22	Finish testing & characterization:	Feb 20
Design PIC interface & circuitry:	Feb 29	Design PIC interface & circuitry:	Mar 12
Final assembly:	Mar 14	Final assembly:	Apr 08
System test of FSR/PIC:	Mar 25	System test of FSR/PIC:	Mar 21
Accelerometer Development			
Finish testing & characterization:	Feb 27	Received accelerometers:	Feb 15
ADC interface with PIC:	Feb 29	PIC ADC functional:	Feb 28
Finalize design:	Mar 14	Finalize tilt and bump algorithms:	Mar 10
System test of Accelerometer/PIC:	Mar 25	Finish MATLAB test bed:	Mar 26
		Implement tilt algorithm in PIC:	Mar 17
		System test of tilt with POD:	Mar 28
		Finalize z-axis sensor:	Apr 04
Sensor Processor			
Blinking LED:	Feb 22	Blinking LED:	Mar 05
UART interface:	Feb 29	UART interface:	Mar 16
System test of PIC & Sensors:	Mar 14	System test of PIC & Sensors:	Mar 20
Test of Sensor Processor to MIDI Controller communications:	Mar 25	Test of Sensor Processor to MIDI Controller communications:	Apr 07
MIDI Controller			
Arrival of MIDIBox unit:	Mar 03	Arrival of PIC 18F2420:	Mar 03
PC to POD interface:	Mar 10	PC to POD interface:	Feb 25
Generate MIDI messages:	Mar 17	Generate MIDI messages:	Mar 12
System test of MIDIBox to POD communications:	Mar 22	System test of MIDIBox to POD communications:	Mar 20
		User interface integration:	Apr 04

The most major difference to note in the FSR development is the later than anticipated completion of the final assembly. This discrepancy is due to making of a Solid Works designed model for the FSR enclosure, a feature of the project that was not included in the original plan. The accelerometer development was delayed by the implementation of the Matlab test bed that proved to be crucial for developing the tilt and bump algorithms. The use of the PIC18F2420 instead of the MIDIBox development kit was the most major deviation from the timeline in the MIDI Processor development. The justification for this choice is that we did not wish to be constrained by the pre-existing framework of the development kit; also, we wanted to implement a user interface, an option that the development kit did not afford.

6 Personal Reflections

Paul Carriere

Working independentlyon a large, technical task is much more difficult task that I had originally anticipated. It involves a great deal of planning and foresight because, in the end, all the parts need to work together. As the semester started, we worked relatively independently on individual tasks, developing knowledge and researching possibilities. A good deal of time was spent discussing this information, and exchanging ideas. This process was crucial because it exposed the core elements of project, and allowed for members to begin developing their tasks. Meetings 2 or 3 times a week were not un-common, and kept the group synchronized.

As the semester progressed, and our parts began to arrive, we began to separate into our respective groups and tackle our problems. We transitioned from the meeting room to the lab. Meetings were held less frequently, but it was never difficult to gather the entire group to discuss and solve issues. Knowing who was doing what allowed for effective 'mini-meetings' where 2 or 3 members gathered to solve a program. This was a very effective technique, as it minimized time wasted watching someone debugging code or cutting wires.

I believe the group dynamics were good, especially considering the rather devastating effect of having 3 weeks and months of planning shaved from your semester. Although a bit discouraged, it was great to have people become immediately interested in a follow-up idea. Everyone had an opportunity to excel, which is demonstrated by the creative components of our project.

During the semester, I was responsible for the accelerometers and I developed the procedure for using them in our system. This task included sensor selection, signal filtering, sensor compensation, understanding digital read-out, and soldering countless electronic widgets. I also developed a Matlab-based emulator for the sensors, such that they could be artificially simulated. This allowed me to bring together many topics I have learnt during my education, and further my ability to do advanced analysis using software. It was really rewarding to create something in the digital world, and then watch it work in the real world.

My only regret is that we never had enough time to test and iterate the system. Considering the fine tuning required for a musician to seamlessly use it during a performance, I don't feel like we had everything working as well as we had planned it too. We had gone to create lengths to have a very intuitive system, and I feel like we are almost there. The hardware components worked better than expected and now it's a matter of getting it all 'just right'. But that was a problem that we had little control over, since the semester came to a close.

The most important lesson I learnt during this project is that a independent verification of any system is crucial for its proper operation. What might seem to work one day won't the next, and you must be able to quickly identify why.

In closing, I would like to say that watching other people enjoy something that I helped make was a really inspiring experience.

Ben Shewan

There were many individual and group lessons learned throughout the course of our system development. Before taking ENSC 440, I had never taken on a project of this magnitude and complexity, with such an aggressive timeline. I feel that the course has prepared me for many of the challenges and tribulations that I will face in my professional career. It proved to be a wise decision that I only take two other courses this semester, as this project consumed the vast majority of my time.

I strongly believe that every member of our group gave this project everything that they had. There were many late nights in the lab and I really think that everyone stepped up to the plate when crunch-time came.

I cannot stress the importance of maintaining good relations with group members throughout the project development. It is very easy to become irritated with others when you work in such close proximity with them for an extended period of time. Because our group was comprised of highly dedicated people with very strong personalities, it was inevitable that some friction would ensue. When such situations arise, it is important that you stop and try to take things in perspective. Ultimately, I knew that every one of us wanted the project to be the very best it could be and remembering this helped me to stay level headed when there were differences of opinion.

Besides the lessons in interpersonal skills and group dynamics, there was a vast amount of technical knowledge acquired over the last four months. It was often necessary to take a topic that you knew little about and become in expert in a matter of weeks. For much of this project my group and I were operating outside of our comfort zone in terms of the specific knowledge needed to get a task done. Although there was a steep learning curve, it is at these times that the most personal growth can be realized.

Since our design philosophy was modular in nature, much of the project was geared towards small groups or independent work. Individual members would take ownership over some aspect of the project and see that part to completion. When there were unexpected difficulties the group was flexible enough to come together and tackle the problem. This method was exceptionally effective at producing results in a very short amount of time, although system integration was not always seamless.

Overall my experience in ENSC 440 was a resoundingly positive one. We accomplished all that we set out to do in an exceptionally short amount of time. I think that *RockIt* is a product with real potential and great practical value. I feel nothing but pride for my group and myself that this project was such a success.

Kyle Huffman

The most important lesson I learned throughout this project is that it takes a very long time to perform a task in a technical project. As a group, we discussed heavily the methods we would use to accomplish our goals. Although time constraints pushed us to make decisions, by initially spending time to rigorously analyze our choices for components, we undoubtedly saved time in the long-run.

Technically, I was impressed with our group's resourcefulness. I suppose we've had many years to practice our problem-solving skills in the faculty of engineering and it showed in our execution of our product plan. By assigning each member to be the 'chair' of specific tasks, we were able to allow members to specialize. This allowed a lot of freedom, but also requires a lot of trust. This was also a great way to showcase our skills because we all had to investigate various things that we didn't always know much about. Sometimes, it was necessary that a less experienced member was responsible for tasks that he was least qualified to do. Nonetheless, you can see from our final product these obstacles were overcome.

I have gained many tools from this class; it's safe to say I've learned more practical skills in this course than any other course. By being responsible for planning, prototyping, testing, and refining a design, you gain more than just coding or circuitry skills. Although, I did learn about PIC programming, DSP for human motion, SolidWorks fabrication, and MIDI controls to name a few things. There's also a lot of collaboration that is necessary and sometimes there's nobody to speak with. There's also a shortage on funds available. These restraints can be frustrating but there's an upside. I definitely improved my resourcefulness.

Dan Galeano

The capstone project has been a challenging and a very rewarding experience. We created the group in the Fall of 2007 and immediately realized how difficult it was to find an appropriate idea for this project. There were however several advances in terms of group dynamics, we had several meetings throughout the semester and also created an account in Central Desktop, a wiki that supports small group collaboration features.

Having our first idea rejected after we had already submitted the proposal and started working actively on the project was very discouraging. For me this was probably the most critical moment in the project as we all knew that it would require a significant amount of work and commitment from everyone in the group in order to produce a satisfactory project within the limited time frame. I really liked the approach the group had to choose: define a problem, generate a solution and a corresponding design to address the area of interest; as opposed to constraining ourselves with a method or design to then look for a problem or an application to implement.

Several lessons were learned throughout this experience. One of them is documenting the problems that have been solved in order to avoid making the same mistake over and over. We had situations in which a significant amount of time was spent on debugging problems that we had encountered before.

It is also a good idea to share the problems encountered and their corresponding solutions with the rest of the group. Also, the advantages that come from assigning personal tasks within the group were also essential to identify priorities and responsibilities for an effective design. Taking ownership and responsibility of a task can sometimes be intimidating, especially when it isn't clear how one is going to address and solve the problem. However, this was crucial to complete project successfully within the expected time. I'm very pleased to have tried new skills I hadn't put into practice before such as soldering and embedded programming.

One of the challenges I had throughout the project consisted on identifying how much information and research needed to be done in order to accomplish a task successfully. I regret for example not reading throughout the data sheets of the PIC from the beginning as it would have saved me a significant amount of time when debugging the different problems I encountered. It was also important to know when to ask for help, and especially who was the most appropriate person to answer my questions.

It wasn't until the end of the semester that I realized how everything was starting to integrate into the idea we had initially envisioned. I feel very proud of this project and this group for being able to develop RockIt successfully.

7 Conclusion

We believe that our project has been a tremendous success. Our goal was design an intuitive control mechanism for guitar effects, and the video of four musicians using the system for the first time proves that we were successful. We achieved all the major goals of our project, along with additional goals like custom-designed FSR enclosure and advanced usability features.

Our group was composed of four very skilled engineers, and our collective abilities really shone through during the course of the semester. It was important to have a wide variety of personalities and perspectives, and this kept the project innovative. Also, the fact to that every member was responsible for a specialized component ensured that every component was well fabricated.

The feedback we have received thus far suggests that our idea might have some serious musical applications. People who have used it commented on ways to improve it, and if it would be possible to acquire their own hardware. Also, people who have seen the video are interested in trying RockIt, which is a testament to our achievement.