The Audio Group

tag-340@sfu.ca

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

December 18, 2000

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

Re: ENSC 340 Process Report for an MP3 to Stereo Gateway

Dear Dr. Rawicz,

Attached you will find The Audio Group's *Process Report for an MP3 to Stereo Gatewa*y.

Our report outlines our project and our technical implementation, followed by the details of the difficulties we faced throughout the semester. Finally, each group member has included a brief summary of what they learned, loved, or hated in ENSC 340.

The members of TAG are still actively engaged in debug process, and will likely be working throughout the week. We are: Tom Halford, Aaron Kaiway, Jeff Robinson, Ross Tulloch, and Ross Wightman. Please feel free to query us with any questions, Aaron Kaiway is our contact person. He can be reached at 925-6168 or at akaiway@sfu.ca. We look forward to greeting you at our demonstration on Thursday, December 21st at 10:30am.

Sincerely,

Aaron Kaiway

Aaron Kaiway,
TAG-340 Project Group

Enclosure: *Process Report for an MP3 to Stereo Gateway.*

The Audio Group

tag-340@sfu.ca

Audio Gateway Project Process Report

Project Team: **Tom Halford**
Aaron Kaiway
Jeff Robinson
Ross Tulloch
Ross Wightman

Contact Person: **Aaron Kaiway**
akaiway@sfu.ca

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

Submission date: **December 18, 2000**
Revision: **1.0**

Table of Contents

1 PROJECT OVERVIEW

The recent explosion of Napster from a relatively unknown service frequented by college students into the international media spotlight demonstrates the public's interest in receiving music over the Internet. While Napster provides music for free in a chaotic fashion, it is only a small jump to reason that this method of distribution may be tried on a more organized commercial basis. If such a service were even fractionally as popular as the Napster example, such a project would be commercially viable. Subscription or per-use oriented business models for music distribution could change the music industry permanently. There are signs that the music industry is considering these models. Universal Music and Sony Music have recently made announcements concerning the release of digital catalogues. BMG has signed a partnership with Napster.

Although the use of MP3s has grown exponentially, playback methods have been largely limited to portable devices or to computers. If digital music distribution is to gain wide acceptance, playback devices must become widespread and not be limited to the current options.

The Audio Group set out in July to design and build a prototype device that integrates MP3 playback into the ubiquitous home stereo system. Our solution is the Internet-to-stereo Audio Gateway. This document chronicles the progress of the Audio Gateway project over the past three months. A high level introduction to the device is first presented, followed by a technical summary of its design. The document proceeds to present a description of the problems encountered along with the solutions we devised. Finally, we present a summary of what each group member learned from the Audio Gateway experience and recommendations for further work.

Figure 1 illustrates in a block diagram form the high-level design solution that we developed in order to build the Audio Gateway. We envisioned a set of functional blocks that work together in order to transform and transfer encoded music data from a remote server into an audio signal in the users home stereo system.

The following describes the function of each of these blocks as data travels a remote server to the users home stereo:

1. *Power & clock blocks*

The power block provides power to all of the other components. Both 3.3V and 5V lines are required. The clock provides the signals that synchronize the other components. The clock and power blocks are the foundation upon which the rest of the Audio Gateway is built.

2. *Physical layer interface*

Packet data from remote servers is received at the physical layer interface. We chose the Ethernet standard for this interface as it is becoming the standard for ubiquitous home

networks. If a user has an existed DSL or cable Internet connection it is relatively simple to create an Ethernet network to which the Audio Gateway can interface.

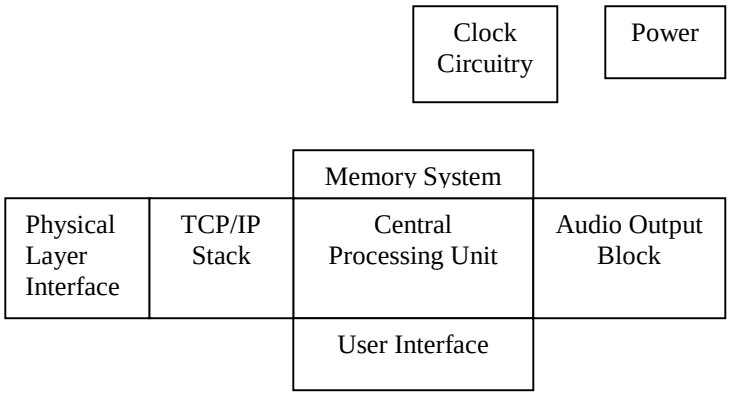

Figure 1: High-level block description of the Audio Gateway device

3. TCP/IP stack

The data packets received over the Internet connection enter a TCP/IP stack where they wait to be handled by the processing unit.

4. Central Processing Unit

The central processing unit takes packet data from the stack and stores it temporarily in memory. This block also receives user data through its connection to the user interface block.

5. User Interface

The user interface (UI) consists of a liquid crystal display (LCD) and a set of buttons that allow the user to control play by stopping, pausing and choosing tracks for play. The LCD-button UI was chosen rather than, for example, a touch screen display to be consistent with the interfaces of other stereo equipment.

6. Memory Block

The memory block temporarily stores music data. Because data is received faster than it can be played, a cache is required.

7. Audio Output Block

The audio output block converts cached music data from the memory block into a signal that is sent, via a cable, to the users stereo amplifier.

2 TECHNICAL SOLUTION

The functional blocks described in Section 1 were implemented using a three-part solution.

2.1 Hardware Solution

Figure 2 illustrates the hardware blocks used to build the Audio Gateway. The hardware is found on two boards: the Microplex print server (the bottom block of Figure 2) and a separate daughter that we designed and built to house additional hardware (the top block of Figure 2). The purpose of each component is summarized.

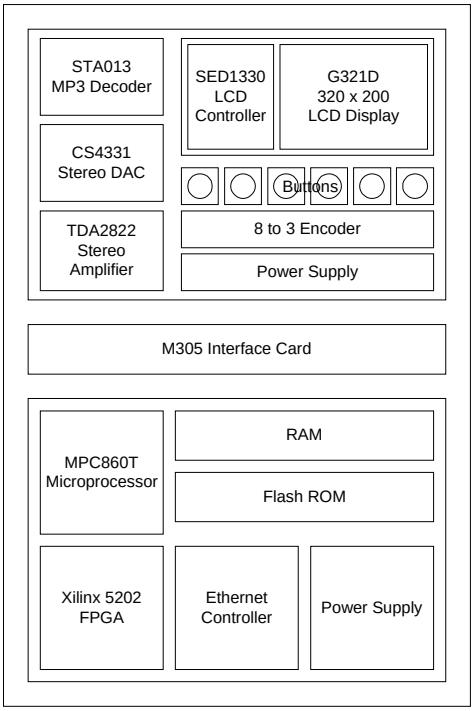

Figure 2: Audio Gateway Hardware Diagram

1. MPC 860

The MPC 860 is the brain of the Audio Gateway. It manages the network connection at a high level, runs application code to interact with the user and communicates with the hardware daughtercard via the XC5202.

2. XC5202

The XC5202 is a field programmable logic array (FPGA) used to implement an interface between the microprocessor and the hardware daughtercard.

3. Quality Semi QS6612 10/100 Base TX MII Transceiver

This Quality transceiver provides Ethernet connectivity (refer to the physical layer block of Figure 1).

4. Micron SDRAM 16 Megabytes

This memory chip provides sufficient memory to run an embedded operating system, buffer network data and run application code.

5. AMD 16 Megabits Flash Memory

This memory chip provides sufficient memory for storage of the operating system, application code and stored user data.

6. STA013 MP3 Decoder

The MP3 decoder chip takes encoded music data from the FPGA and converts it to a digital audio signal.

7. CS4331 Stereo DAC

The DAC is used to convert the digital audio signal produced by the MP3 decoder into the analog signal that is required by the user's home stereo.

8. TDA2822 Stereo Amplifier

The amplifier is used to amplify the DAC output so that a set of headphones can be connected to the device.

9. SED1330 LCD Controller & Display

The LCD display provides a 320 x 200 pixel screen upon which user data is displayed. The LCD is manipulated by the application software through the FPGA.

10. Buttons

The user controls the device via a set of six buttons. The buttons are mounted along the short side of the LCD screen so that icons corresponding to the buttons functions can be displayed on the screen.

2.2 Hardware-Software Connectivity Solution

Nearly all of the communication between the network processor and the daughtercard is routed through the onboard Xilinx 5202 FPGA of the Microplex print server. This includes the LCD display, button inputs and the MP3 data stream to the decoder. The logic in the FPGA will simplified the hardware design on the daughtercard since it includes button debounce circuitry and is able to route multiple hardware interrupts into the single available interrupt line of the MPC 860. The FPGA also provides a buffer for the serial data stream, easing time constraints on the software tasks running in the MPC 860. The FPGA is configured using VHDL code that is loaded from the MPC 860's read-only memory (ROM) at system power-up.

2.3 Software Solution

The software solution is twofold:

1. An operating system to run the application software; and,
2. The application software.

We chose eCos as our operating system. Traditionally, Microplex print servers have run a customized real-time operating system that is very expensive to license. eCos is an **e**mbedded, **c**onfigurable **o**perating **s**ystem which functions at a high level like Linux and has the advantage of being freely available as it is open source. Because eCos is in its early stages of development, it does not fully support the MPC 860 board. Thus, part of the software solution was writing and configuring eCos code so that it worked with our hardware.

The application software was built using four main modules:

1. A communications model that acts like an file transfer protocol (FTP) client retrieving music data from remote servers;
2. A file handling module that keeps track of all the files currently available on a selection of servers;
3. A memory module that controllers the buffering of data as per the memory block of Figure 1; and,
4. A user interface module that coordinates the first three modules in response to user commands.

The application software was built using the open source C++ compiler *gcc*, open source linker *ld* and open source debugger *gdb*. By using an open source operating system and development tools we ensured that our financial resources could be focused on hardware (and of course, sex, drugs, and rock & roll).

3 ISSUES ENCOUNTERED

3.1 Technical Issues

3.1.1 Operating System

By using eCos, a very new operating system that is not yet fully supported for PowerPC microchips by its vendor RedHat, we were asking, nay begging, for technical issues. Fortunately, Aaron was able to write most of the support routines without problem. The major issue that held back our project for the entire month of November was the eCos Ethernet driver. Aaron was required to write the software that interfaces the Ethernet port on the board with the TCP/IP stack in the eCos operating system. The few hundred lines of code required to perform this task took a daunting five months to complete. (As a comparison, in the same time span, approximately ten thousand lines of application code were written.) This issue was not a result of technical inexperience but a simple result of sailing into uncharted waters, coupled with a lack of time to focus on the project. By being the first group to develop these routines, we have contributed to the eCos source tree.

3.1.2 Hardware

Choosing and sourcing parts presented some difficulties and delays. Jeff was constantly tracking packages that were not sent, but our foresight of ordering parts early meant that parts arrived in September, as opposed to November. Jeff ran into more problems constructing the board. The fine pitch of the pins of the chosen components, coupled with the high cost of sockets for such chips, prevented a breadboard solution. Jeff was thus forced to use a vector board to build the daughter card.

A second issue was encountered late in the project development cycle. We discovered that a number of lines in a chip that we thought were accessible via pin outs are in fact hidden under a layer of plastic casing. This error was the result of our negligent assumption. Fortunately, an engineer at Microplex was able to access the internal pins with a very creative soldering job.

3.1.3 ASIC Implementation

The major hurdle encountered by Ross Tulloch in the ASIC design and implementation was the small capacity of the FPGA chip. The Xilinx has 256 flip-flops but its routing resources are very limited, and unlike many smaller FPGA's, its look-up tables cannot be configured as RAM. So, in order to meet both the routing constraint and a 40 MHz clock cycle, we had to reduce the size of the MP3 buffer in the FPGA. The buffer acts as an intermediate storage facility between the application code and the MP3 decoder. A large buffer was planned so that a pause in transmission from a remote file server wouldn’t result in a pause, or skip, in the music data sent out of the Audio Gateway to a user's stereo. We thus lost the safety net provided by the buffer.

3.1.4 Application Software

Two major application software issues were encountered. First, the hardware and software were developed concurrently making testing difficult. This issue was resolved by first developing the software under the Linux operating system and performing as much manual testing as possible using a debugger. The Linux code was then ported to the similar eCos operating system, and tested on the board.

The second application software issue lay is Tom's inexperience with low level programming. Initially, it was assumed that Tom would develop the high level code as well as the ftp client used to obtain files from remote site and that low level tasks, such as interfacing with the LCD driver, would be left to Aaron or Ross Wightman. When Ross and Aaron became entangled in the Ethernet driver code, Tom was forced to write LCD code with little help because other group members were extremely busy. The initial results were dismal: Tom mistook an 8 bit register for an 8 byte register and wasted six hours in the process. For the record, however, Tom would like to state that he received an A+ in both ENSC 151 and 250 - the very courses that teach digital design principles.

3.2 Financial Issues

Our initial project plan involved developing the Audio Gateway using the Cirrus EP7212 evaluation board (EVB). This board had the perceived advantage of having nearly all of the hardware required for this project pre-assembled and pre-configured. We foresaw the project thus focusing on developing an intuitive interface, professional casing and polished application software alongside the construction of a sophisticated remote control. Unfortunately, the Cirrus EVB is priced in the $1000US - $2000US range and beyond our budget.

We resolved this issue by partnering with Microplex, a local print server manufacturer. Microplex donated a number of their print server boards as well as the use of office space and testing equipment. Included in the Microplex boards are the microprocessor, FPGA chip and Ethernet socket required by the Audio Gateway. In exchange for the hardware and the use of office space and equipment, we agreed to supply Microplex with all of the code that was written to allow eCos, our chosen embedded operating system, to run on the print server board. With much of the project hardware provided free of charge, we were able to focus our resources on the purchase of a project server to act as a secure code depository and test bed.

3.3 Scheduling Issues

The Audio Group was formed in March of this year. We brainstormed project ideas for three months and began working on the Audio Gateway project in earnest in July. By September, we had received all of our parts, written nearly one half of the application code, designed the daughtercard and completed the high level design for all other aspects of the project. Yet, despite our diligence, we found our studying for final exams compromised by the demands of this project, which continued into late December.

We see two reasons for this situation. First, we may have taken on too big a project. We reject this for two reasons: a catastrophically large project would not have been approved and given the support provided by Microplex, we were in a good position from the start to resolve large conflicts.

In the spirit of Sherlock Holmes, were are left with our second conclusion: the course scheduling this semester left us with little time to do our projects in the months from mid-October to mid-November. Quite frankly, the lack of foresight on the part of the architects of the new engineering curriculum is appalling. Furthermore, the fact that ENSC 340 is assigned only 3 credit hours defies all reason. Tom's experience this semester demonstrates the validity of our complaints. In the past, Tom - one of the top students in our year - has been able to maintain a high GPA, compete on the SFU Varsity Swim Team and lead the lifestyle of a lush. This semester, however, he was forced to step down as the men's team captain and reduce his socializing to nearly null in order to meet the demands of the course schedule.

Clearly something must be done to address this problem to prevent future years from encountering the same problems. We suggest changing the ENSC 340 vector to five or six credit hours. If we could repeat this semester, none of us would have enrolled in ENSC 383 this semester. As a group with no Systems option student in its ranks, we could have afforded to drop this course en masse. Hindsight is, of course, an exact science. Jeff and Aaron are very active members of the student society and have brought this issue to both the Engineering Faculty and the Dean of Applied Sciences, it is hoped that this issue will be resolved.

3.4 Team Dynamics Issues

We were lucky in that we only faced three team dynamics issues. From the outset it was clear that Aaron was the natural leader and other roles seemed to fall into place. As we possess complementary skills we functioned fairly smoothly. The first issue was encountered in September when Ross Tulloch and Jeff Robinson disputed over the amount of work the former was contributing to the project. Aaron stepped in and sent an email to the entire group detailing the responsibilities of each member so that the equality of work was clear. The issue was quickly resolved.

At the beginning of October we met to define our individual work for that month. We then neglected to meet as a group for four weeks because of the demands of midterms and labs in other courses. By the beginning of November we had each completed most of our assigned work yet the entire project was out of step because certain development areas had outpaced others resulting in wasted time as members waited for the work of others. Aaron again stepped in to solve this issue by scheduling weekly status meetings for the rest of the semester to ensure synchronicity within the group.

The final team dynamic issue was faced in December when it became clear that the project would not be finished before the 16th - the day of Tom's flight home to Toronto for the holidays. Tom's flight could not be re-scheduled and he was thus forced to write documents whilst keeping contact with the team in Burnaby as they debugged his software.

4 PROPOSED SOLUTIONS

Section 3 of this document addresses the solutions to many of the problems we encountered, however, in retrospect there are a number of areas where we could have used different solutions.

The biggest issues encountered were the result of our choice to base our hardware development on the Microplex print server. For fiscal reasons, we would follow this path again. We saved a few thousand dollars by using Microplex hardware and an open source operating system. A number of changes could have been made, however, in how we approached the Microplex hardware.

The development of a functioning Ethernet stack was not made a priority early enough. As team leader, Aaron was the chief architect of the first project documents and in charge of organizing the team. Had we known the challenges he would encounter with the Ethernet code earlier, we would have put all of his resources into that task and left documentation and organization to Tom, whose application code was dependent on the completion of all of the other project components.

Jeff spent many late hours in the lab soldering the fine pitch pins to vector board sockets. In retrospect, this was wasted time that should have gone into completing and testing the hardware by an earlier date. A better solution would have been to buy a few hundred dollars worth of sockets with the money that we saved by using Microplex hardware.

Hardware issues aside, we should have been more vigilant in our scheduling of meetings. Too often we relied on casual conversations over lunch and before class to gauge the progress of our project. While such casual encounters were generally fruitful in solving technical issues, they were not coupled with organized meeting to enforce project milestones.

5 INDIVIDUAL EXPERIENCES

The following is a summary of what each group member has learned while participating in the project.

5.1 Aaron Kaiway - Project Manager

I've learned a diverse set of lessons over the course of this project. These include lessons in the areas of team management, project planning, and course loads.

I was expected to take a leadership role in this project, in positions of both technical advisory and people management. Many of the lessons I learned were not technical in nature, but rather were managerial in nature.

Every engineer has a different set of talents and skills. Since this project required such a broad range of tasks to be performed in such a short amount of time, each team member was assigned a specific focus area. When work in certain areas bogged down, it was next to impossible for team members to transfer to a task outside of their focus area without a considerable amount of ramp-up time (5-10 hours of study and familiarization). This made it difficult to allocate resources and people to work on problem areas. The amount of work expected from each team member contributed to lessening the distribution of knowledge. Some problems that took single group members hours in trying to resolve were fixed in a matter of minutes after a team member with more knowledge was introduced to the problem. If the problem set were narrowed, problem solving time conceivably would have decreased.

The diverse range of tasks that were required highlighted the necessity of a particular area of project work – communication. Group communication, both on an informal level and a formal level are essential. Periodic meetings with the group and documentation detailing precisely what each designer was performing allowed for every team member to be informed of the project status. Documentation also allowed every team member to set goals and strive to achieve them. Documents also allowed questions to be answered in a timely fashion when consulting group members with the necessary knowledge would have taken hours. Informal communication, mostly via lunch-time chats or email discussions were essential in clearing up small details, troubleshooting, and communicating deadlines. The most important aspect of communication was that it allowed every group member to feel involved in the project and take ownership in the collective effort.

Finally, I learned that I should have never taken more than twelve credits in the semester when I took Ensc340. The sixteen credits that I took, though beneath the recommended department course load of 19 credits, forced me to work between seventy and eighty hours per week from October to December. This amount of time was:

1. not sufficient to finish my coursework in a fashion that I deem acceptable.

2. Allowed me to go home for dinner seven times over the course of the aforementioned three months.
3. Disenfranchised a large number of students in my year.
4. Showed that while faculty members recognize that engineering requires a broad range of knowledge, some fail to recognize the limits of human endurance.
5. Made me wonder repeatedly why no one showed up with a shotgun one morning and started shooting.
6. Has made me seriously question why I want to be a professional engineer.

5.2 Tom Halford - Embedded Software Design Lead

In addition to our group experience regarding this semester's curriculum as detailed in section 3.3, I learned a few things about team dynamics, software design, and my future career path.

Our team worked well together, but as the others it's clear that our parallel development required better planning when it came to system integration. We put so much time into our own efforts that we hardly considered integrating software and hardware until exams were already upon us. Given the chance to go back, we should have made system integration a priority from the beginning.

My experience with the application software was invigorating. I wrote TCP/IP client software and, thanks to this project, I claim that I can network program (although some group members may disagree after looking at my code). The project was also a great way to review theory of my previous courses. Of course I knew there's 8 bits in a byte!

I also realized that I have little interest in entrepreneurial projects in engineering. Designing products for people simply doesn't excite me very much, actually I prefer syrup to such projects. This course showed me that I should focus my efforts where they are most valued, in research and development or academia. Theoretical work comes more naturally for me, and with less effort and sacrifice. Thus, I am thankful to ENSC 340 for setting me helping me define my career path.

5.3 Jeff Robinson - Hardware Design Lead

Over the course of my ENSC 340 semester I learned several important lessons. I have distilled these lessons into the following 2 pieces of wisdom:

Never underestimate the worth of a finalized design before construction

Trying to create a pin mapping with hot soldering iron in hand is never a good idea. Often I found myself trying to figure out an unanticipated design problem midway through construction. After frustrating attempts to solve the problem in parallel with construction I discovered that it is always best to stop construction and develop a coherent solution. It is also vital that this solution be fully articulated in writing, so that future verification will have a solid plan to work from.

Projects are never seem too ambitious in September

Our project idea was really cool when we first thought of it. A consumer grade information appliance, neat right? Sure we'd have to build some hardware using surface mount technology, port an operating system to an embedded platform and write application code for use in real time; but we're engineers, we can handle it.

The brutal reality is that midway through December we realized that five students working sporadically do not make a professional development team.

Of course there is a fine line between manageable and challenging, and this is what we should be aiming for. The ideal solution would have been to select a project with a core set of trivial deliverables that could be expanded upon as the project progressed.

5.4 Ross Tulloch - ASIC Design Lead

ENSC 305/340 has definitely been the most worthwhile and stimulating course this semester. Our group anticipated the rigors of this course and began planning our nontrivial design, of a streaming MP3 device, during the summer. We designated a leader and partitioned sections of the design to each group member. I was given the relatively simple but very integral task of designing the hardware for our FPGA primarily because I just finished a coop designing digital circuits with VHDL. The technical work certainly presented challenges, but this course presented more important lessons in team dynamics and organization.

All our group members worked very well together and we did our best to properly set up the project with infrastructure to enable proper code development and provide easy access to information. As a result, we had little difficulty putting together quality documentation for ENSC 305 since we had already created similar documents.

Our partitioning of the design was quite useful in the early stages of the project since it enabled us to work around our own schedules. But, as the project progressed, critical paths developed and unfortunately we were not able to make full group efforts to solve arising issues. This is because the complexity of the project warranted enough distinct tasks such that each member essentially became a specialist in just their task.

My work with the FPGA served as an excellent follow up to my Coop last semester in Rick Hobson's VLSI Research Lab. I spent the summer learning the Synopsys design tools and developing a well-structured VHDL coding style. So when I went back for this project I was able to step right in where I left off. But this task definitely had some new challenges such as: how to design with an FPGA's very limited resources in mind, learning the Xilinx Foundation development software, and also how to determine detailed interface characteristics from various data sheets. I am grateful to this course and project for enabling me to enhance my digital design skills, since I will indubitably use them in the future.

Given the chance to defy the Relativistic laws of physics and return to the start of the semester, I think we would probably still use our concurrent design methodology, but create fewer overall tasks and have at least 2 people per task. Also, investing in an

evaluation board could have alleviated some hardware difficulties. And of course, if we held beer drinking parties throughout the semester instead of just at the beginning, and hopefully the end, we would definitely have finished on schedule, but that's just my opinion, I could be wrong.

5.5 Ross Wightman - Firmware Support Lead

I have learned a great many things over the duration of this semester both directly, and indirectly related to ENSC 340. Most of what I have learned, such as team dynamics, project planning, problem solving, and technical knowledge will be an asset to me as I continue along my path as an engineer. Unfortunately, a significant portion of what I have learned this semester, issues of dealing with an extreme course/project load, should have been apparent to those with more experience and wisdom than ourselves (the professors and individuals who designed our program) and thus not needlessly endured by us.

To digress away from the negative, I must admit that most of what I have learned during ENSC 340 is very useful. My understanding of the benefits and pitfalls of team dynamics has increased substantially. I now realize that constant communication is the key to success even if it means time taken away from the technical tasks at hand. As a whole, our group had good communication, whether in formal meetings or through informal banter. One area of communication that our group could have improved on arose due the division of tasks within our group based upon technical expertise. Each person in our group worked on a specific area that was based upon that person's skill strengths. However, all tasks in the project eventually had to interact and coexist and this brought about some confusion, misunderstanding and hold ups. This was not due to group members' inability to communicate, but our inability to take the time in this hectic semester to keep ourselves familiarized with the other areas of the project. I constantly found myself wrapped up in my section of the project, other courses, and with little idea of the technical workings of other areas of the project. Only necessity forced me to take a step back, consult the other members of my group, and allow me to look at the bigger picture.

In addition to communication, a well-defined project plan is essential. While we did have a respectable plan for our project months before the semester started, our inexperience with large undertakings such as ENSC 340 meant that our plan couldn't and didn't account for everything. We encountered, and are still dealing with, a large number of technical pitfalls that we weren't aware of until we actually started working on the project. I expect that the foresight required for project planning will come with experience.

Since our project unifies many of today's hottest technological areas, I also picked up a significant amount of technical know how in areas such as real time operating system architecture, power PC microprocessor programming and architecture, Ethernet and TCP/IP protocol, and Xilinx programmable logic devices.

6 THE NEXT STEP

During the course of ENSC 340 and 305, The Audio Group has developed a proof of concept for the Audio Gateway. The next logical step for our team is to improve upon the design in order to present it at the Western Engineering Conference & Competition (WECC). We see WECC as a proving ground for our device. Enthusiasm from our peers will provide us with the motivation to bring our product to market. In order to prepare for WECC we will effect a number of cosmetic changes:

1. Build a custom case so that the housing looks professional;
2. Complete the power function block so that the device can be plugged it to a standard wall socket- currently the device runs off a lab power supply;
3. Develop a more graphical, less text-based, used interface; and,
4. Optimize the application software for performance.

If we were to bring this product to market, a number of major changes would have to be effected. These include reducing the part count, improving and expanding upon the user interface, allowing for network connections other than Ethernet, and improving the robustness of the device's firmware. In addition, a through consideration of our target markets would have to be performed.

A reduction in part count for our device would significantly reduce the cost to produce our device. Our current design solution includes six major ICs. The are ICs available which offer combined functionality. (ie, mp3 playback and CPU) Using fewer ICs would allow us to reduce costs for both our ICs and our surrounding support circuitry.

Our current design solution has a single user interface - the graphical LCD display. In order to make user interface easier, interface options such as PDA-based control, remote control, and television-based control should be considered.

We currently connect to a network using 10/100 Base-T Ethernet. To increase the appeal of the device, a modular system additional networking options such as homePNA, and wireless connections should be designed.

Finally, this project has been developed with a consumer market target. Due to the size and competition or the consumer market, companies participating in this market require large capital investments. It is likely that an Audio Gateway related product would have to be 'co-branded' by a larger company in order to gain widespread exposure. Alternatives to the consumer market, such as private institution-specific (ie, hotels, private homes) markets should be explored.