Russ Bielawski
jbielaws@umich.edu

CryptoKeys: End-to-End Encrypted SMS-ish Messaging Device for GSM Phones

What if you could text anyone, anywhere in the world without worrying about anyone in between reading your message? Perhaps this doesn't actually sound like much to someone who doesn't feel oppressed. But, as recently demonstrated during the Arab spring, such a device could have a wide purpose in the modern world. The possibility of a low-cost add on to any mobile phone to enable end-to-end encrypted real-time SMS-like messaging is not mere fantasy.

Details

I envision CryptoKeys, a simple keyboard and display with an audio jack. This audio jack could be plugged into any GSM phone, and after initiating a phone call, could be used to send messages in two directions, like SMS. This translation of cell phone call into SMS machine is accomplished using a tried and true method, audio modulation via a modem. In my research I have shown the ability to modulate and demodulate data into audio for transmission across a GSM phone call using a cheap microcontroller SOC.

The tradeoff for complete end-to-end realtime encrypted personal messaging is twofold. The most obvious tradeoff is the lack of support. Two parties need to each own a CryptoKeys device in order to communicate securely. Contrast that to the near ubiquity of cell phones in most parts of the world. The second tradeoff is speed. At this point in the technologies development, a full 160 8-Byte character SMS message would take roughly 2 minutes. This isn't quite the convenience we're used to from traditional SMS, but strong end-to-end encryption might just be worth the tradeoff.

To use CryptoKeys, before or after initiating a GSM phone call, each party plugs their CryptoKeys device into the audio jack of any GSM cell phone. The two devices, sharing an AES key, would then be able to communicate securely. Key exchange would not necessarily done by the device. Asynchronous crypto (a la SSL) could be used to do so on the device, or an unspecified side channel could be used.

Schedule

9/12-10/1 – Research and Planning

Creating planning documents for the 498 project and for the success of the CryptoKeys happens at this stage. Major parts search and design feasibility must start now (things like keyboard, LCD, enclosure, integration plan) to ensure the feasibility of the project. Full project should be shown to be reasonable at this point, and design fleshed out. Good documentation is required for success.

10/1-10/7 – Initial Design Phase

An initial design must be created and made solid. This is taking the fleshy, but incomplete, design from the previous stage and talking it into implementation.

10/7-11/25 – Design and Implementation

Design continues with implementation. The project is broken up into parts, so that work can be serialized among the team. Roles could be split into things like integration and physical parts responsible, low-level protocol concerns responsible, high-level protocol concerns responsible, hardware responsible, software responsible. Test responsibilities must not be neglected.

The integration responsible must work on getting a nice enclosure for the CryptoKeys device. This person must design mockups, and ensure that we can a reasonable device. The low-level protocol concerns responsible needs to get the modem / channel interface working and optimize it. This is open-ended, and the better the system is, great. The high-level protocol responsible should work with encryption and a software interface for transmitting messages. The hardware & software responsibles handle the basic parts of hardware and software on the device. Testing responsibilities can be shared (as could many of these repsonsibilities).

10/25-11/30 – Testing and Robustness

This phase must run concurrent with the implementation phase. The system needs to work as expected, or at least show good improvement towards doing so. This involves tests of all the things being built from physical to software to hardware to protocol robustness to security.

11/30-12/3 – Final Push

At this phase, everyone works frantically to work out bugs and polish the product.

Budget

The budget for this project is mostly based upon belief that the parts can be purchased cheaply enough. It is expected that an LCD and Keyboard could be purchased for under $100. Custom work is not my expertise, but I am following cues in the course that this will work within the budget.

Optional Stretch Goal

Using asynchronous crypto to do key exchange would be really awesome, but add a lot of work.

A Mobile, Programmable Tactical Targeting System on the Cheap

Despite other appearances, many law enforcement officers (LEOs) do not receive thorough sidearm training. Although LEOs do receive time at the shooting range, the bar for acceptable is very low. And, working with moving targets is minimal. The main reason for this is, obviously, cost.

The purpose of this project would be to build a fully programmable (free 2-D motion) targeting system. The system would be able to move one target around the field freely. Although this system won't be dirt cheap, it must be cheaper than the competition and take advantage of modern technologies for efficiency.

Obviously, the system must be able to withstand a barrage of bullets. One single target will be able to move freely in 2-D space. The system shall be programmable, ideally by an iOS/Android app so that commodity devices can be used to program the system.

Schedule

9/12-9/24 – Research and Planning

In this phase, research into major parts and cables and things must progress. Major design documents are started but are very high-level.

9/24-10/1 – Finalization of Initial Design

Feasibility should be determined. An initial design for this large system must be finalized. The team should be comfortable with the types of parts that will be used in the mechanical design, where to get them, and how to evaluate them. Software and hardware designs must be close to finished.

10/1-10/21 – Prototyping and Beginning Building Mechanical Components

This system might need prototyping (building a smaller version). The mechanical team must get cracking on this stuff. Those working on the prototype must work with the hardware/software team.

10/1-10/21 – Hardware and Software Implementation

The hardware and software team(s) must get off and running to design the control interfaces and algorithms.

10/21-11/25 – Full System Implementation

Implementation continues, but with more focus on the full product. The system must start coming together.

11/25-12/3 – Final Push

The system's loose ends must be tied up. It should work, even if the user interface (for programming) isn't beautiful. But, it should be functional and usable.

Budget

This almost certainly blows the project budget!

Major Issue

- This needs to be a big robust thing. It requires LOTS of mechanical design at which most students in the class will not be skilled.

Optional Stretch Goals

Mobile Device for Programming

It would be ideal to develop iOS and/or Android apps for programming this thing. This would take advantage of modern advances and allow use of nice, cheap commodity devices for programmers. This also requires and interface to provision the shooting range via a network.

Multiple Targets

The ability for multiple targets to share the space would be really cool, but would create many more problems / design constraints.

Stabilization Algorithms

It's possible that the target stands straight up from the positioning system. This is a classical controls problem of stabilizing the upright agent. It would be cool to tackle, but probably well too much work for the schedule.

First Proposal: Robotic Line Painter for Sporting Fields and Parking Lots

Summary:

Painting lines on the ground is used for several different applications. Painted lines are used for marking sports fields, parking lots, and roads. Currently the only way to paint these lines is to use a push cart that has a sprayer on the front. This technique requires extensive planning and surveying beforehand. There is also the potential for a lot of human error because the straightness of the line is dependent on how straight the operator pushes the cart.

A possible alternative for this would be to use a robotic line painter. With a robotic line painter you would be able to remove human error from the operation. Line patterns could be designed in CAD software and then transmitted to the robotic line painter to carry out the work. With this software you could design parking lot layouts more efficiently resulting in more spots and potentially more revenue. Sporting fields could be preprogrammed in the device requiring the user to simply choose the type of field and let the robot do its thing. Along with lines the robot could paint images and logos. For instance a school logo could be painted in the middle of the school's Football field, something that would normally require an experienced artist.

Details:

The line painting robot would be a three wheeled robot with two motorized wheels and a third wheel for balance. The paint will be dispensed by a pressurized system on the robot. For this project a simple aerosol can can be used that will be activated by a small actuator. The robot will be powered by a car battery. The car battery will provide plenty of power to the robot considering the surfaces the robot will be operating on will generally be flat. In order for the robot to be accurate, a positioning system will need to be used. GPS is not nearly accurate enough for this application. A more feasible system would be to use a laser triangular positioning system. This could be implemented using a laser rangefinder, a rotating mirror and three reflective beacons. The laser rangefinder would be pointed completely upwards and defected off the rotating mirror. While the mirror would rotate a full 360 degrees it would point the rangefinder beam to each of the three reflective beacons giving the robot three distances. With these three distances the robot would be able to determine its location. An angle sensor would also have to be used on the mirror in order to determine which direction the robot is pointing. With this constant feed of information the robot would be able to adjust its course if need so.

Schedule:

9-12 to 10-1: Planning

For the first two weeks the group will spend its time coming up with a simplified design for the robot. A basic wiring diagram should be figured out as well as a component layout of the robot. The key components should be decided upon and by the end of the two weeks ordered. Key components include microcontroller, motors, motor controllers, rangefinder, angle sensor.

10-1 to 10-15: Testing interfacing of key components

As key components start to show up the group will start trying to interface with them and figure out the basic electrical interaction between them all. By the end of the two weeks we should be able to send and receive information to each one of the components so that we fully understand how they should be connected to each other. A crude prototype should be the result of these two weeks, basically a microcontroller connected to freestanding motors and sensors.

10-15 to 10-29: PCB Design

The group at this point should split up into two teams. One team will be responsible for the construction of the robot chassis and the other team will work on the PCB design. With the knowledge gained from the previous two weeks we should be able to design the PCB layout. The PCB will include the microcontroller, the motor controller and power supply for the motor. There will also be connections that will go the rangefinder, angle sensor and sprayer actuator. A data storage device might also need to be included in the design to store the patterns the robot will paint. The other team should have the chassis done so that wheels and motors can be mounted.

10-29 to 11-12: Construction/Software Development

Once the PCBs are ordered the build team will work towards finishing the chassis of the robot so that the only thing missing from the robot will be the circuitry. The rangefinder and mirror should be mounted and ready for testing. The group that was previously designing the PCB boards should start developing software that can control the robots movement.

11-12 to 11-19: PCB Installation and Rangefinder integration

The build team should have the robot completely constructed by the end of this week, including all the circuitry. The software team should finish the integration of the range finder to the existing software.

11-19 to 12-3: Calibrate robot and fix bugs

The last two weeks will be a full team effort to fix software bugs and calibration issues. A couple people will spend time programming the demonstration pattern into the robot while the rest of group will make sure that all the sensors are giving accurate data and make necessary adjustments.

Budget:

Item	**Quantity**	**Unit Cost**	**Total Cost**
Netduino Plus	1	$60	$60
Motors	2	$50	$100
Chassis/Parts	1	$300	$300
Rangefinder	1	$400	$400
PCB	1	$50	$50
Paint sprayer	1	$50	$50
Angle Sensor/Actuator	1	$25	$25
Battery	1	$50	$50
Miscellaneous	1	$100	$100
Total			$1135

Issues:

1. Budget
2. Rangefinder/will it work

The two main issues revolve around the cost and complexity of a laser rangefinder/ positioning system. The cheapest rangefinder I could find was around $400 and to get a proven one you are going to need to spend twice that. There are a couple of ways around this. The first is to rely on the angle sensor alone to do positioning. If you get the angles between the three beacons you still calculate your position. This requires a very accurate angle sensor and it still needs a laser sensor to detect the reflector. Other options and ideas may exist, like active beacons, but they would have to be researched.

Wacky Idea: Quad rotor Flying Movie Camera

Summary:

Robotically controlled cameras have become more and more common in movies today. This is because computer generated scenes require consistent camera movement that cannot easily be carried out by a camera operator. An example would be if you have a pan of the actors overlooking a landscape but the landscape is not actually there. The robot would film the actors first and then later film the landscape using the exact movement so that when the two clips are added together the scene would look seamless. There are many other possible effects that would require the same technique. The problem is that these movements are usually grounded and cannot be used for an aerial shot. The only way to get aerial shot would be to use a helicopter or plane with a pilot.

An Idea for a project would be to build a quad rotor that would be able to pull off these spectacular aerial movie scenes over and over again consistently by programming their movement beforehand. This camera would also save movie studios money by allowing them to ditch the helicopter rental and use one of the robotic cameras. Also scenes where it would be hard to use a full sized helicopter could use the much smaller quad rotor camera.

Details:

The flying camera would take previous quad rotor designs but add a movie camera to it and the ability for the camera to pivot vertically. Since the cameras used in feature films are heavy the quad rotor would have to be quite sizeable. A minimum size would probably be around four by four feet. Smaller cameras could be used but might result in a diminished image quality. The microcontroller on board would have several things to keep track of. It would have to control the four motors separately for controlled flight and also control the cameras angle. The flight would be controlled by a person on the ground using a standard RC transmitter setup. In order to pre-program the flight you would have to have either a record function on the transmitter or the ability to load a program into the transmitter.

Budget:

Building a larger quad rotor from scratch would not be possible in the time given and would require a good amount of knowledge in aerodynamics. A more reasonable approach would be to buy an existing quad rotor helicopter and modifying it. The helicopter alone would eat up most of the budget. As a result you would have a small amount to use on PCBs and camera control. The camera being used would most likely need to be borrowed or mocked up.

Schedule:

As said before the only way this would be possible to do the in allotted time is use a quad rotor kit. If time becomes more of an issue the flight control could be giving to a controller that has already been developed for these type of flying machines.

Issues:

The two main issues with this project are the budget and skill required to fly a quad rotor RC helicopter. Budget issues could be solved by team members donating more to the project. As far as flying the camera, practice with smaller quad rotors might be needed so that the project helicopter is not destroyed.

Andrew Curtin
EECS 473
9/10/2012

Homework 1: Project Pre-Proposal

Idea 1: Space Balloon data logger

Intro

Space is pretty awesome. Learning about space is fun, but actually doing experiments yourself is even better. This project would be an amateur space balloon that would be launched into space (or near space) and take pictures as well as log data. The project could be summarized as a data logger, but it's a bit more than that. It needs to be rather resilient, as space is quite cold. It also needs to be reliable, because it would be a large disappointment if it were launched into space and part (or all) of it failed.

Details

This project would essentially be a data logger. We would try to get as many different sensors as we think would be interesting or useful. There would also be a parachute system to deploy the parachute when it is close to the ground. The sensors would include cameras, temperature, pressure, maybe a Geiger counter, accelerometer, gyroscope, maybe a magnetometer, and a GPS. Some sort of radio transmitter would also be required to locate the device after it lands. If a suitable radio can be found, it may allow for streaming of the data in real time, and possibly even remote control (stretch goal). Care would need to be taken to ensure all components used can withstand the low temperatures in space. This will most likely mean using higher-grade parts where available.

9/12- 10/1	Planning and Research In the three weeks after the project groups are formed, we will need to carefully define our project, research and select parts, get a rough design and write it all down in our proposal. Prototype plans will be solid.
10/1- 10/12	Design and part selection It is expected that the first 1.5-2 weeks will be spent further fleshing out the idea, selecting parts and designing the communication algorithms. During this time a few different wireless radios will be tested. A GPS must also be chosen as well as a microcontroller. As soon as a part is selected it will be ordered so as to reduce the lull between ordering parts and being able to start on the design (thus the overlap with the next step).
10/8- 10/29	Breadboard prototype and PCB design Once parts are selected and ordered a breadboard design will be implemented. Once we are certain the hardware design is correct PCBs will be ordered. This is expected to take about 3 weeks. Getting the PCB order out by 10/29 puts us 1.5 weeks ahead of the deadline.
10/29- 11/12	Software development and testing on prototype While we are waiting for the PCBs to come back, software will be refined and a test plan will be developed. Once the boards arrive we will assemble them (in 2- 3 days) and ideally have things working by Nov 12th.
11/12- 11/26	Software development and testing on PCB-based system Testing will be done to insure that rapid association is working. Back-end communication schemes (cellular, Wi-Fi or whatever) will first be extensively used. Accuracy of GPS will be tested. Bugs will be found and fixed. PCB issues resolved. If needed PCB revisions will be made. Any additional features/stretch goals will be researched decided upon by this time.
11/27- 12/3	Final testing, additional features, poster and final report Targeting project completion at 11/27. At this point we'll do all of our writing and continue to fix any remaining minor bugs.

Budget

At this point in time, a budget is rather difficult. Rather than have a details budget, the following is a possible breakdown of how funds could be allocated.

Part	cost
processor / embedded linux platform	$130
GPS	$40
Radio	$50
other sensors	$100
PCB	$30
battery	$40
balloon and other non electronic parts	$200
total	**$590**

Issues

The main concerns are:

1. Ensuring reliability
2. Determining what radio or radios to use for communication
3. Battery tolerance to low temperatures

Issue 1 is the most difficult, and may not be easily testable within our means. Issue 2 should be solvable with a bit of research. Issue 3 is also a research issue, but may also by testable.

Additional stretch goals

Add some motors and rotors to allowed the balloon to be controlled, both during ascent and descent (this should make it much easier to find as well). Battery life would be a concern with this though.

Idea 2: Chiptune player

Intro

This project will combine a few different synthesizer chips used in various game systems with a midi interface to allow for easy chiptune reproduction that is true to the original systems' sounds.

Details

This project would have a midi interface for connecting a midi keyboard (or multiple midi keyboards) to allow easy playing of music. This would also allow a computer with a midi interface to connect to the device and play prerecorded chiptunes as well. The main processor will interface with a number of synthesizer chips. This will require lots of IO, and the more powerful the processor, the better, to allow for more different notes (and chips) to be used at one time. This project is all about interfacing.

Issues

The main issues will be:

1. Sourcing the old chips
2. Dealing with any supply voltage mismatches
3. Find documentation for these old chips, and possibly dealing with esoteric interface schema.

These issues all stem from the chips being old, and the requirements that specific chips be used.

Automated Home Energy Saving System

In today's world there is a lot of wasted energy just due to the lights, appliances, and utilities left on when people leave their homes and businesses. While more energy is wasted the world is also becoming busier at an increasing pace. This causes people to have less time to worry about and take care of this problem, which is wasting money and resources. According to http://physics.fau.edu/observatory/lightpol-econ.html, it is estimated that about one third of the lighting in the United States is wasted, which amounts to about $2 billion. If there was a way for a home/building to automate the saving of energy it would solve this problem without causing people to spend time out of their busy lives turning off lights and other appliances in their homes and businesses.

There are many systems in place for home automation of light control, however most of those involve motion sensors, timers, or a remote. Motion detectors can become annoying since they do not always work correctly and can turn off lights even if you are in a room but not moving directly in front of a sensor. Timers by themselves do not solve the problem and remotes require extra work for someone to turn everything off and also become one extra piece of hardware that need to be carried around. This is why I instead propose a smarter system that knows when to turn the lights off.

Details

The automated home energy saving system will involve wireless communication between several devices and control of electrical outlets. In order to sense if someone is home I propose a system that has a central unit that communicates with other smaller devices and knows when they are within its proximity. There are two options for the proximity device in order to make this system work. The first option is to use a smartphone to communicate with a central device. With the increasing popularity of smartphones, this makes it easier to implement since it would eliminate one piece of hardware. An app could instead be written that simply detects when the smartphone is near a central device. A second option is to create a simple device that could be wired inside a vehicle. Most American households own a vehicle so this also allows for this system to be used in most homes.

As mentioned in the previous paragraph, there would be communication between the smartphone or device in a vehicle that would determine how far away someone is from the home and whether or not to turn on electrical appliances and lights. There would then also have to be communication between the electrical outlets and the central device. This would allow for the central device to tell the outlets to be turned on when it senses a proximity device is near, or off it is far away or out of range. The devices on the electrical outlets would be simple switches and could be implemented with wireless technology so that they are easy to install. The wireless technology allows for this system to be implemented without rewiring an entire house.

Overall the system would include 3 types of devices: A central device that determines how close someone is and whether the electricity should be turned on or off, a proximity device which can either be a smartphone or small sensing device wired inside of vehicle, and simple switches with wireless capability connected to electrical outlets that allow for the electricity to be turned off. Both the "proximity devices" (other than the smartphones) and the outlet controllers will have microcontrollers and the central device will have an FPGA. This system would also have the ability to be disabled in case there was a need for the electricity to be on when the proximity devices were out of range.

Schedule

9/12/2012 – 9/24/2012: Project Planning

The details of the project will need to be ironed out before it can be decided what parts need to be purchased and how the different portions of the system will be designed.

9/24/2012 – 10/1/2012: Write Proposal and select Parts

Once the project has been defined and details ironed out, we will write our proposal and decide what parts to use in the system. We will test out different ways to implement the wireless communication within our system as well as what sort of operating system should be used for the central device.

10/1/2012 – 10/5/2012: Order parts

During these few days we will order the parts that we need for the project

10/5/2012 – 10/18/2012: Breadboard Prototyping

Before we can design our PCB for the different components we will need to design 3 different prototype circuits on breadboards for the central device, the proximity device, and the outlet controllers.

10/18/2012 – 11/1/2012: PCB Design

Once the breadboard prototypes have been designed and tested we will create a PCB design in EAGLE. This will give us a week before the deadline of ordering the PCB's.

10/18/2012 – 11/15/2012: Software Development

At the same time the PCB is being designed, we will begin development of the software that will run on all three of the systems. This will give us about a month to write the code for the software and work out any bugs before we try to integrate all of the parts. We will have a lot of time to focus on this while the PCB are being made and will be able to test it out on the breadboard prototype.

11/15/2012 – 11/22/2012: PCB Testing

Once we receive the PCB we will be able to test it out and solder to it any components needed for it to work with our code. We can also test out our software with it and decide if any changes need to be made to the software to make it work with the PCB.

11/15/2012 – 11/25/2012: Integration of Parts

When all of the individual parts have been designed, built, and have the bugs worked out, we will integrate them together. This will be a very time consuming part and will require a lot of debugging in order to get each of the parts working with each other seamlessly.

11/25/2012 – 11/29/2012: Finalizing of System

After everything has been integrated together and is working, we will have a few days to polish up the system. We can start writing our final report during this time. This will also give us a few extra days in case we were unable to work out all of the bugs before this deadline.

11/29/2012 – 12/4/2012: Preparation for Demo and Design Expo

During these few days we will prepare everything for the demo and Design Expo. We will practice so that our presentation looks very professional for both the in-class demo and the design expo.

12/5/2012: In-class Demo

We will demonstrate our finished project to our entire EECS 498 class. We will use any feedback to improve our presentation for the Design Expo.

12/6/2012: Design Expo

We will demonstrate our final project to all interested faculty and students somewhere in one of the buildings on North Campus.

Budget

It is very difficult to come up with an exact budget at this time since the project has not yet been designed. Since it will not be feasible to implement this in an actual home or building given the constraints of this project, we will instead design this on a smaller scale. We will build 3 outlets, 1 central device, 1 device that can be wired into a car, and use preferably 2 smartphones.

Given that at least two people in a group have smartphones that will not have to be added to the cost of the project (this will probably be a requirement given the budgetary constraints). The price for the microcontrollers was taken from the ATmega128RFA1, which is priced at about $9.46 on Digikey for a unit with zigbee capability. This would probably be on the higher end for a microcontroller that gives us the needed wireless capability. The price for the FPGA for the Central Device comes from Digikey as well. The A2F200M3F-1FGG256 from Microsemi costs $35 and should fulfill our needs of being able to communicate with multiple wireless components. However, I budgeted $50 to give us a little extra room on choosing that device. Overall the budget on the next page keeps us well within the $250 per person range.

Part:	Quanity:	Cost per Unit:	Total Cost:
FPGA – Central Device	1	$50	$50
Microcontrollers	4	$10	$40
PCB	5	$15.00	$75
Wireless Components	5	$20	$100
Shipping	1	$100	$100
Miscellaneous Parts	1	$200	$200
Total:			$565

Issues

1. Developing a method that will easily detect if person is gone without disturbing them
2. Wireless communication between central device and proximity detectors
3. Creating detection device that could be used in place of smartphone
4. Creating central system interface

Stretch Goals

One stretch goal is to also include control of heat and air conditioning as well. Many people are gone for the same exact hours everyday however they leave the air conditioning and heat on set at the same temperature, which is also a waste in energy. Both heating and cooling systems could be set at a lower threshold during the day and then calculate how long it would take to return to that higher threshold so that homeowners would not even notice the difference in temperature when they return. This would require one extra device that could interface with heating and cooling systems and communicate with the central device.

Automatic Administration of Medication

Elderly people accidentally take the wrong medication all the time, which hurts them in their fight against disease and other effects of time. It can be easy enough for someone without dementia to confuse what medication they need to take when they have to take 5 to 6 pills a day, let alone someone who has it. If there was a way for people who have to take a lot of medication to have this automatically administered to their body instead of memorizing what pills to take on which days it could save and prolong people's lives.

Solution

This device would have to be something that a person could comfortably wear so that it would actually be worn, since otherwise it is useless. It would also have to be programmable so that it could be loaded with certain medication as well as when to administer a specific dose of medication. A user friendly interface would have to be developed in order for a pharmacist (most likely someone with no programming experience) to select all of the options for how the device would work.

The device would also be able to monitor the status of the patient so that if anything was going wrong, paramedics could be called immediately. The device could help save someone if they were having a heart attack since it would essentially call 911 when they are unable to. The device would therefore have to communicate with a cell phone modem in order to place the call or at least send the information over a call. It would also have a GPS in order to send the location of the person in trouble to the paramedics. A microcontroller would be used to control all of these devices.

Overall this would allow for elderly and sick people to become more independent since they wouldn't have to worry about the different medications to take. Elderly people and their families could have the peace of mind knowing they would be taken care of in the case of an emergency.

Schedule

9/12/2012 – 9/24/2012: Project Planning
9/24/2012 – 10/1/2012: Write Proposal and select Parts
10/1/2012 – 10/5/2012: Order parts
10/5/2012 – 10/18/2012: Breadboard Prototyping
10/18/2012 – 11/1/2012: PCB Design
10/18/2012 – 11/15/2012: Software Development
11/15/2012 – 11/22/2012: PCB Testing
11/15/2012 – 11/25/2012: Integration of Parts
11/25/2012 – 11/29/2012: Finalizing of System
11/29/2012 – 12/4/2012: Preparation for Demo and Design Expo
12/5/2012: In-class Demo

12/6/2012: Design Expo

Budget

Part:	Cost per Unit:	Quantity:	Total Cost:
GPS	$50	1	$50
Microcontrollers	$10	1	$10
PCB	$15	1	$15
Modem	$50	1	$50
Miscellaneous Parts	$200	1	$200
Shipping	$100	1	$100
Total			$425

Issues

1. Creating a device that someone can comfortably wear
2. Configuring the device to make a 911 call or contact paramedics
3. Designing a PCB to interface between different microcontrollers, a GPS, and a modem
4. Designing an effective and easy way to automatically administer medication to person's body

Stretch Goal

While being able to create a useable system above is already somewhat of a stretch, an additional goal would be to add extra monitoring capability besides just the heart rate so that disease could be caught earlier. If additional monitoring capability was added, a central monitoring unit could also be built that could interact with the device the patient was wearing.

Rebecca Doebler
Project Proposal – Homework 1

Automatic Home Window Shutter

During the hot and humid months of the summer, in an effort to reduce the cost of the heating/cooling bill, many people will opt to keep their windows open to allow for a cross breeze rather than run an air conditioner. In fact, providing fresh air flow into a house is a low-cost and potentially energy-efficient way to cool a home and maintain good indoor air quality[1]. However, it is important to make sure windows are closed when they need to be, such as in storms and blowing rain.[1]

The issue of rain and storms can cause potential problems if one chooses to leave his or her windows open during the day or even during the night. If the windows are left open during the day and a sudden rainstorm appears, things inside the home can get wet and cause possible damage. In this case, having a device that can detect when it is raining and close the windows partially or all of the way could come in handy.

Similar ideas like this have been proposed on a larger scale for things such as skylights but have not been proposed in the smaller scale[2].

Details

In order to implement this project, a rain sensor would be necessary outside of the user's home/apartment that can communicate wirelessly to a microcontroller connected to the windows of the home.

To make implementing this project easier, an optical rain sensor can be used outside of the home since it is completely sealed and can get wet. This optical rain sensor would transmit data wirelessly using the XBee wireless transmitter. On the inside of the home the microcontroller would receive the message that is raining, or it is not raining and using a mechanical device will close the window if it is open and it is raining. The window will have a pressure sensor on it that it can transmit to the microcontroller that tells it if the window is already closed.

To control the mechanical device, a DC motor can be used with an H-bridge and connected to a lever that is attached to the window which will move forward or backwards based on the output of the microcontroller.

[1]http://www.epa.gov/greenhomes/ReduceEnergy.htm
[2] http://www.rainsensors.com/its_raining.php

Schedule

9/12-10/1	**Planning** In these first few weeks, we will need to define what we want our project to consist of how we want to carry out the details
10/1 – 10/12	**Design and part selection** Decide the best method for controlling how to open and close the window. Test and decide which wireless communicator will be best for this project. Decide which microcontroller will be best and research and select which rain sensor will be best suited for this project.
10/8 – 10/29	**Breadboard prototype and PCB design** This will be especially important for the mechanical portion of the project. Once the best method is selected for controlling how we open and close the window, the breadboard will be created and tested. Once we are sure that the hardware design is correct we will order the PCB.
10/29 – 11/12	**Software development and testing on prototype** Test and refine the software while waiting on the PCB to arrive. Make sure that wireless communication is working properly and make sure that rain sensor is outputting data correctly. Once the board arrives, assemble and begin testing to ensure things are working correctly.
11/12 – 11/26	**Software development and testing on PCB-based system** Testing will be done to ensure that mechanical device is working properly. Wireless systems are communicating properly and correct data is being sent to the microcontroller. Resolve any PCB issues during this time.
11/27 – 12/3	**Final testing, additional features, poster and final project** Finish up testing. Add some additional features if there is time.

Budget

A rough estimate for this project:

Optical Rain Sensor: $59.00 – 1
ATmega128RFA1: $7.00 – 2
DC Motors: $15.00 – 2
PCB: $15.00 – 2
Horizontal slider window: $60 – 1
Misc: $50

Total: $243

Issues:

Largest concerns: Figuring out the best method for opening and closing the window and power consumption of the rain sensor (How long will it last battery operated?)

Automatic Bottle Opener

The basic idea of this project is to open up a variety of bottles using a robotic arm which can detect what sort of bottle needs to be opened. There are several different types of bottles: twist off, cork, and bottle caps. The bottle will be placed on a platform that contains a pressure sensor. Once the pressure sensor is triggered, the robotic arm will use a camera that will scan the bottle, send the data to the microcontroller and through calculation, determine its type.

Once the type of the bottle is determined, the microcontroller will send a signal to the robotic arm to grab to open the bottle in the manner that it needs to be opened. This automatic bottle opener could be used in bars or as something fun to use at parties. This would be a fun project but would be tricky due to mechanical issues that may arise.

Budget

Arm gripper with servo motors: $177.74
ATmega128RFA1: $7.00
DC Motors: $15.00 – 2
Serial camera with video - $42.00
Misc parts: $50

Total: $306.74

Outstanding issues:

One large issue with this project would be figuring out how it would move mechanically. The arm would have to have several different pivot points in order to move properly. In addition to these pivot points, figuring out the method to actually open the bottles will be difficult as three potential hands on the arm may be necessary to carry out that step.

Another issue that may arise is figuring out an algorithm to detect what sort of bottle needs to be opened. The camera will be useful to get an image of the bottle, but figuring out how to use that image to determine the bottle might be a challenge.

RFID based Transportation Violation Detection System Design for Intersection

Introduction

In recent years, with the rapid growth of the number of motor vehicles, the existing traffic management system is facing tough challenges. In major cities of China, most of the traffic accidents are caused due to illegal driving. Based on the resent technologies, general traffic monitoring system is using the electronic camera to determine the illegal acts of the vehicle at the crossroads of important sections. However, it is a little hard for the image processing when the weather is bad, such as rainstorm and fog. Another way to test the transportation violation is to use sensors which are buried under the road. However, the wires are always broken due to the heavy vehicles and road construction.

Thus, a new technology should be found in order to help efficiently control the transportation condition and punish the violated vehicles. As we all know, RFID (Radio Frequency Identification) are widely used in product tracking applications, yard management, shipping and freight and distribution centers[1]. It should also be used to track a car. Based on this idea, we can design a transportation violation detection system for the intersection area.

Design Schematic and Details

In order to detect the transportation violation, we need to build a small intersection, which can be shown in Figure 1.

Then we need to design an embedded system in order to control and implement this system. The system structure diagram is shown in Figure 2. And the RFID reader and vehicle information collection diagram is shown in Figure 3.

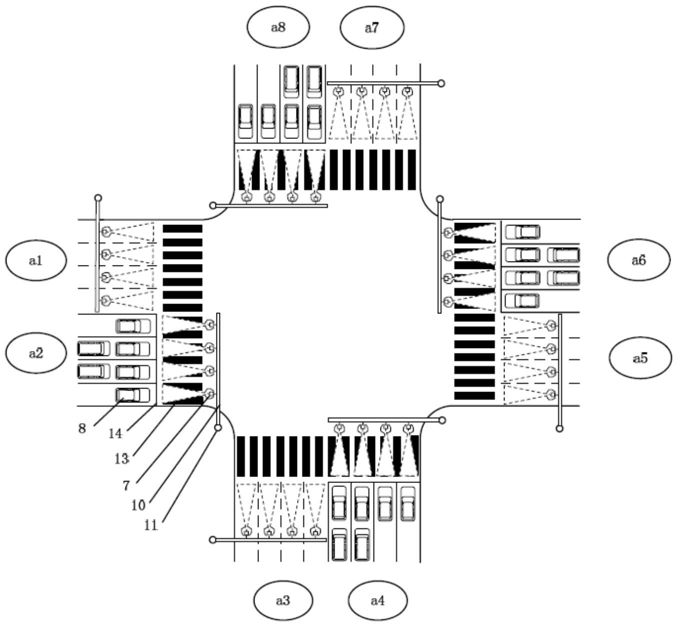


Figure 1

From the Figure 1, we know that each road will have its own RFID reader and each vehicle will have a RFID emitter. Then what we want to do is to use the reader to detect the when and which road a vehicle is travelling and in order to test if this vehicle is violation or not.

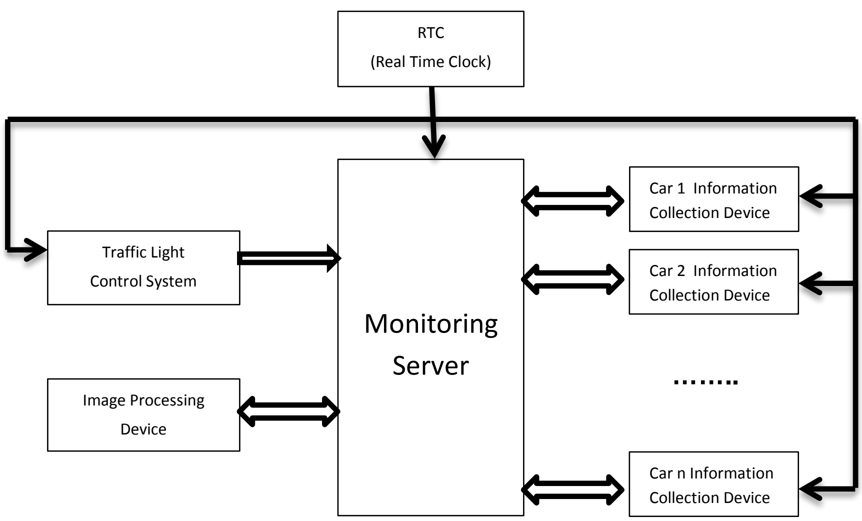


Figure 2

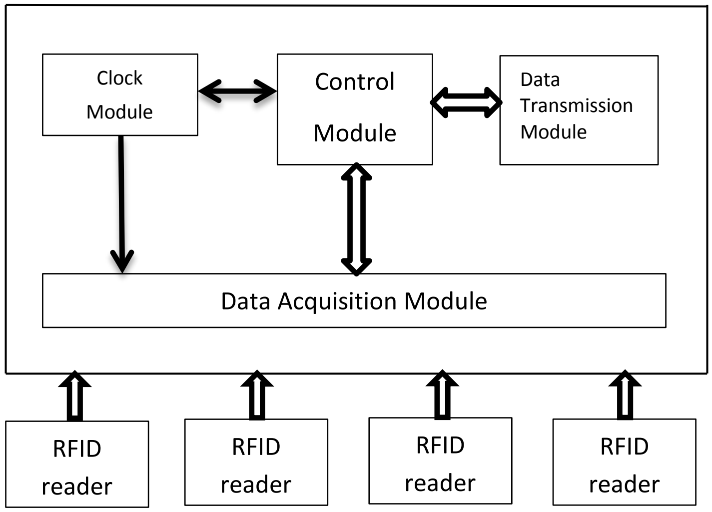


Figure 3

Schedule

9/12-9/24	Planning In these two weeks, we need to figure out how to design our transportation system. Because this is not a real transportation, we can only use small cars and virtual streetlight and road. After finish discussing this part, we need to write a demo by using CAD or 3D-max.
9/24-10/15	Embedded System Design and part selection During this time, we need to get together to decide how to design this embedded system in order to test the vehicle violation. After making sure all the part function in figure 2 and 3. We need to select our microcontroller and also need to divide our work into several parts.
10/15-10/25	PCB design We need to order our PCB. Getting the PCB order our by 10/25 puts us 2 weeks ahead of the deadline.
10/25-11/15	Simulation development and testing on prototype In order to rebuild a virtual transportation system in computer, we can use VISSIM or Matlab to set up a simulation module in order to test our algorithm.
11/15-12/25	Testing on PCB-based real system During this time, if our simulation works well, we can try to test our real transportation system. Accuracy of RFID will be tested, because it contains the DAQ and measurement error. If encounter any testing problem by using RFID, try to modify our detection algorithm.
12/25-12/05	Final testing, poster and final report writing Finish all the paperwork, write the final report, design our poster, fix small bugs and prepare for the expo!

The dates selected are mostly “lines in the sand” and will almost certainly not be exact though they should be close.

Budget

It is very difficult to write down all the budge. However, we can definitely write down some things that we need to buy. If this proposal can pass, we can search on the Internet to find how much and what type of the equipment and electron device we need to use.

What	Quantity	Cost per	Cost total
Toy Car	10		
RFID chip	10		
RFID reader	64		
PCB-base			
PCB-vehicle			
Microcontroller			
Total			

Possible problems

During the RFID detection, based on our knowledge, this will definitely contain testing problem, which means we cannot catch the car information correctly and timely. The solving method and some detail reason can be found on the Internet.

A design for a car game implementation

Idea:

We will try to build a car driving system by using a microcontroller or microprocessor. According to our basic knowledge, most of real world vehicle have an ACC (adaptive cruise control system), and several kinds of cars have emergency avoidance system. We can try to rebuild this system by design a mechanical part and then the embedded system part. Then we can use an interface board and driving our cars on a virtual environment just like the need for speed. Furthermore, we can build more cars and using more control buttons to race them!

Mechanical Design:

In this part, we can use a wheel as the steering wheel. We can also attach an encoder on this wheel in order to know the exactly position of our steering wheel. The haptic wheel will serve as the steering wheel to your simulated vehicle. The position of the wheel will determine the angle of the front wheels of your car and the steering forces will be affected back to the wheel. By turning this wheel, it just likes to drive a real car. For the accelerator part, we can use a potentiometer which is designed on our embedded system; the volt could be proportion to our car speed.

Embedded system design:

We can try to build our own embedded system, which contains some dip-switches, LEDs and CAN communication. Then we can use a tool kit called 3D realm in Matlab to build our virtual racing track. After all this have done. We can write down the control C code in order to drive them on the computer or on the interface screen!

To model the vehicle motion, you will implement the differential equations describing the motion of the vehicle model. We also need to write code to interface with the hardware (GPIO, PWM, QADC, FQD, and CAN).

Details for ACC

The ACC system will be activated/deactivated by toggling one of the GPIO bits with the dip switch. When the ACC system is inactive, the vehicle speed will be manually controlled with the potentiometer connected to the QADC. When the ACC system is enabled, the potentiometer input is ignored and the speed set point in m/s will be given by the binary value of the lower eight GPIO bits divided by four.

Automatic Pediatric Blood Pressure Home Monitor

Screening for pediatric prehypertension and hypertension is frequently not provided by physicians.[1] With the increasing percentage of overweight or obese children and young adults – two conditions that increase risk factors of prehypertension and hypertension – this screening is becoming more important for physicians to perform on pediatric patents. Any patent diagnosed with either prehypertension or hypertension has a greater risk for additional associated health problems and upon diagnosis will require additional screenings for any correlated or underlying disorders. Prehypertension and hypertension both have associated symptoms and both have well-accepted methods of treatment that can be prescribed upon diagnosis.[2] The early diagnosis of prehypertension for a patient moving toward hypertension can help prevent the more serious and potentially irreversible symptoms of hypertension. This early diagnosis is especially useful when diagnosing pediatric patients because the lifestyle changes required as part of the treatment of both prehypertension and hypertension are more likely to be permanent if introduced to the patient in their formative years.

The physicians that do not screen pediatric patients for prehypertension and hypertension often decline to screen because it necessitates adjusting the systolic and diastolic blood pressure readings taken with a standard blood pressure machine to the patients' age, weight and sex. Only after this adjustment can the physician determine the percentile of the pediatric patient's blood pressure and make a subsequent diagnosis. This adjustment is performed manually by the physician by using charts published by the National Institute of Health, which list the percentiles of blood pressure of boys and girls from age 1 to 17 by percentiles of height.[3] This adjustment is a tedious process that can easily be over looked by a busy physician.

The proposed automatic pediatric blood pressure monitor would increase a physician's ease in screening a pediatric patient for prehypertension and hypertension by removing much of the manual process required to adjust a blood pressure reading to that patient's age, weight and sex, facilitating more screening tests on pediatric patients. A physician using the monitor would be required to type in, using an intuitive and simple touch screen and physical keypad interface, a patient's age, sex and weight. In addition, the proposed monitor is economical enough to be used by the patient in their home for daily monitoring, a suggested aid to treatment for both

---

[1] RIiley, Margaret. "High Blood Pressure in Children and Adolescents." *American Family Physician*. 85.7 (2012): n. page. Print.
[2] Rodriguez-Cruz, Edwin. "Pediatric Hypertension Treatment & Management." *Medscape Referance*. NOVARTIS Grant/research funds INVESTIGATOR. 2011. Web. 9 Sep 2012. <http://emedicine.medscape.com/article/889877-treatment>.
[3] U.S. Department of Health and Human Services, National Institutes of Health, National Heart, Lung, and Blood Institute, National High Blood Pressure Education Program. The Fourth Report on the Diagnosis, Evaluation, and Treatment of High Blood Pressure in Children and Adolescents. NIH Publication No. 05-5268. Bethesda, MD: National Heart, Lung, and Blood Institute. Revised May 2005, pp 8-15.

prehypertension and hyptertension.[4] The monitor would fill a substantial hole in the current home and professional blood pressure market by providing a method to screen with greater ease pediatric patients for prehypertension and hypertension, an important and necessary screening in the determination of a patient's health.

Details

The proposed machine will be battery powered and roughly the same size as currently existing home blood pressure monitor systems that have a base station and an attachable cuff. The machine will be able to run for at least six hours active, many days idle before depleting its batteries. The machine will be able to connect to a computer over Bluetooth to upload past measurements to a software logging application that will be developed in conjunction with the machine. This software will be able to do some basic trend analysis on the data that it has logged.

The machine will be housed in a plastic case and will contain a roughly two inch color TFT display paired with a capacitive touch screen. This screen will display the patients' age, sex and weight as well as the original and adjusted systolic, diastolic readings from the patient. In addition, the screen will display the patients pulse and will indicate whether the machine is connected over Bluetooth to the logging application previously mentioned. The machine will also contain a physical sixteen keypad which will be used to enter the patients' age, sex and weight.

The machine will be based around an AVR atmega328 microcontroller. This microcontroller was selected because of its large number of GPIO pins, its moderate level of processing power which is required for the TFT screen, for its low cost and for its ease of use.

Because this machine will be used as a diagnostic tool accuracy and reliability will be two primary design concerns. The machine will be developed to be able to be manufactured at a large scale in a way that is economical enough for an average consumer.

The machine will be designed with ease of use as an additional primary design consideration. Because of this, as much information as possible will be abstracted away from the user. It is a design goal to get as close to a single button operation as possible.

Schedule

This schedule is preliminary and is should not be taken as a proposal of strict deadlines. It is organized by week with week 1 being the first week of class fall semester 2012 and with weeks overlapping.

Week 1 to Week 3	*Planning* During the first two weeks we will form teams

[4] Rodriguez-Cruz, Edwin. "Pediatric Hypertension Treatment & Management." *Medscape Referance*. NOVARTIS Grant/research funds INVESTIGATOR. 2011. Web. 9 Sep 2012. <http://emedicine.medscape.com/article/889877-treatment>.

	and solidify high level and then low level design decisions of the blood pressure machine. We will draft requirements to meet and start planning how we will attempt to meet them.
Week 3 to Week 4	*Design and part selection* During this week we will use the requirements defined for the machine to select appropriate parts.
Week 4 to Week 7	*Breadboard prototype, PCB design, Enclosure Design* During these three weeks we will both develop the breadboard prototype and layout the PCB based on determinations and lessons learned from the bread boarding experience. We will order the enclosures at this point.
Week 7 to Week 10	*Software development and testing on breadboard prototype* During these three weeks we will continue testing the bread board prototype and will develop the pairing software application.
Week 10 to Week 12	*Software development and testing on PCB system* During these two weeks we will assemble the PCB and finish developing the paired software. We will also receive the enclosures and mount the final system. We will then perform physical reliability tests.
Week 12 to Week 15	*Final Testing, additional features, poster and final report* We will verify that our design met our original requirements by completing our testing. We will summarize our results and prepare and present our final presentations.

Budget

This budget should be understood to be preliminary. The below items allow for a breadboard prototype and two PCB prototypes. These preliminary estimates bring the project well below budget of a team of five.

What	*Quantity*	*Cost Per*	*Cost Total*
AVR atmega328-AU-ND	2	2.77	5.54
AVR atmega328-PU-	1	2.88	2.88

ND			
2.4 TFT color display, touch screen	3	18.50	55.50
16 key pad	3	3.50	10.50
Bluetooth Module	3	12.00	36.00
PCB	2	50.00	100.00
Pump	3	12.00	36.00
Pediatric Blood Pressure Cuff	2	50.00	10.00
CNC enclosure	2	150.00	300.00
Misc Parts	1	100.00	100.00
Shipping	1	80.00	80.00
Total cost			826.42

Concerns

This project has five major implementation concerns. These are listed below in order of concern.

1. Insuring reliability
2. Developing the enclosure
3. Developing algorithm to take blood pressure readings to perform the adjustment with 100 percent accuracy
4. Soldering surface mount components

Concern 1 will be addressed by testing components individually and by paying extra consideration to physical mounting and layout of the machine. Reliability will be insured through testing the machine in real life situations and repairing any weak link components or aspects of design. Concern 2 will be addressed by making sure that one member of the team is experienced in some regard with 3D modeling software that can output CAM files. If this is not the case, a team member will seek out this education and become proficient. Concern 3 will be addressed by consulting medical texts and through careful unit testing. Research will be performed on popular techniques of how other blood pressure machines calculate systolic and diastolic readings as not to reinvent tried and true techniques. Concern 4 will be addressed by making sure one team member is comfortable with soldering surface mount components by the point when the PCB’s arrive.

Testing

The machine will be tested for three main functionalities.

1. Physical Reliability

2. Accuracy

3. Usability

Physical reliability will be tested by building the machine and testing for weakest link parts. In addition, the machine will be subjected to above normal harsh operating environments.

Software accuracy will be tested through a unit test suite. Hardware accuracy will be tested through volunteer use and through identifying and forcing edge case behaviors such as low battery life.

Usability will be tested through regular volunteer test patients. If possible these patients should range from very low to extremely high weight and in sex and age.

Each team member will be responsible for at least one aspect to test and will be responsible on reporting this tests status to the rest of the team on a regular basis.

A successful design will be able to be used by anyone above the age of 8 years old and will be accurate and usable for extended periods of time in a variety of environmental conditions.

Stretch Goals

If time permits it would better the design to add a lithium ion battery that can be recharged over a 120 volt wall circuit. This would be determined before the PCB design process. It would also be good to add a feature that allows the tables used to perform the adjustment algorithm to be updated over Bluetooth. In addition it would be helpful to make the machine interface with common blood pressure machines found in hospitals.

Separate but Shaking

As international business continues as standard practice, companies must develop methods of performing their common functions across different cultures. Trust is both one of the hardest to foster and most fundamental components of conducting good strong business.[1] One method that the business community has developed to demonstrate trust in a business relationship is the hand shake. A hand shake has come to become an important part of both an informal and formal contract and signifies to all the parties involved that the proposed deal will commence with good will and pure intentions.

With the advent of certain communication technologies, many business deals are conducted without the players of the deal being in the same physical location. While this saves time and money in travel costs, because of a simple matter of arm length, this physical separation does not allow for the deal to be concluded with the standard reassuring shaking of hands. It is clear that it would not be practical in all cases to fly to a location where a hand shake could occur as this would lead to higher transaction costs and would invalidate much of the money saved by the use of the before mentioned communication technologies.

The Separate but Shaking allows business partners separated by an inconvenient distance the next best thing to shaking hands. By shaking either a standalone Separate but Shaking unit or by wearing a “techno glove,” an optional expansion for the businessman on the go, any number of additional Separate but Shaking units – paired to your Separate but Shaking or “techno glove” using our simple and fun to use app – will mimic your grip and motion, translating it perfectly onto your business partners hand from anywhere you might be in the world.[2] In addition, the Separate but Shaking will be available at a competitive cost allowing for units to easily be provided to lower level management so that they can participate in the “handshake” using an optional feature of “conference shakes,” a moral building feature that gives everyone the tactile feedback they need to know that business is going well.

Details

A Separate but Shaking will look very much like a human hand, except it will be mounted on a sleek titanium pole. It will be consist of a fully animatronic five fingered robotic hand with each finger capable of matching the grip strength of a strong human. It will also be able to measure how hard it is being gripped and shaken using force sensors in each finger and an accelerometer in the palm, so that a user’s specific handshake can be replicated onto any

[1] Green, Charles. "Trust in Business: The Core Concepts."*http://trustedadvisor.com/*. N.p., April 29, 2007 . Web. 10 Sep 2012. <http://trustedadvisor.com/articles/trust-in-business-the-core-concepts>.
[2] With an internet connection

paired units. Each finger will be forged from an alloy of tin and copper. There will be four servo motors on each finger and one on the wrist allowing for a full range of human motion.

The Separate but Shaking unit will be controlled by an AVR atmega328 microcontroller which has more than the required amount of processing power and GPIO pins. It will connect to the internet using a Gainspan WIFI chip.

The “techno glove” attachment will be manufactured out of cotton and flex sensors. It will consist of a discrete flex sensor on each joint and a AVR atmega168 microcontroller. It will connect to the internet using a Gainspan WIFI chip. The “techno glove” will not be able to provide tactile feedback to the wearer.

Concerns

This project has three primary concerns:

1. Grip malfunction or “Death Grip”
2. Inhumane feeling fingers
3. Excessive servo noise

Concern 1 will be mitigated by installing analog failsafe’s separate from the digital control logic to shut down the motors above an acceptable level of force. Concern 2 will be addressed by attempting to replicate the motion of the human hand as directly as possible. This will be achieved by reading and consulting with experts in the fields of biomechanics. Concern 3 will be addressed by testing users to see what an acceptable level of noise would be. Various servos will be auditioned.

LCD Twitter Notification Board

By Shaun Levin

How often has it happened where a student is left standing outside of a professor's closed office door wondering when the professor will return? People need better methods to post notices than dry erase boards mounted on doors. Ideally, one would be able to update a short status from a smartphone that gets pushed directly to a small noticeboard. An LCD Twitter Notification Board would solve this issue. The professor could remotely tweet a short message that is then displayed on his office door.

If this solution can be made inexpensive and portable, the LCD Twitter Notification Board could be used for offices, classrooms, and even dorm rooms. With the wide adoption of twitter, it could be used for purposes besides room notifications. Users could configure the device to post updates from sports twitter feeds or even the most recent thoughts of prominent politicians.

Professor Brehob has previously spoken on this idea and is optimistic in its adoption if it is implemented well.

Details

The proposal contains three modules: a display module and a base module. The base module connects to the internet via Ethernet, downloads the most recent tweet from a specified Twitter account, and sends it wirelessly to the display module. The display module receives the tweet from the base module and displays it on an LCD.

The display module requires a microcontroller, LCD, wireless radio, and battery. The battery must be able to power the module for one workday (10 hours). It must be lightweight and small enough to be mounted on a wall or door and portable enough to move to a power adapter for charging. Since few inexpensive and lightweight LCDs can display 140 characters at once, the microcontroller will need to constantly send a rotating message to the display. In order to conserve power, the microcontroller will pull new tweets from the base module once every 5 minutes using a low-power wireless protocol.

The base module requires a microcontroller, Ethernet, and wireless radio. It will use an AC wall outlet for power, so power consumption and size of the device are unimportant. The module will connect to a computer via a serial connection in order to configure the Twitter account handle. The module will download the most current tweet from the set account every 2 minutes. When it receives a pull request from the display module, it will send the most current tweet. This approach is better than downloading the most current tweet upon a pull request because it reduces the time and power required for the display module's wireless radio to be enabled.

Schedule

The following schedule is preliminary and subject to change as details are finalized.

9/12 – 9/26	**Planning** 2 week to plan and finalize all features of the project. We will organize teams and milestone goals. We will also create a rough design and create our proposal.
9/26 – 10/10	**Part Selection** 2 weeks to research and pick a microcontroller for each module, wireless communication components between modules, and a display. Components will be ordered as soon as size and power restrictions for all parts fall within guidelines.
10/10 – 10/29	**Breadboard prototype and PCB design** 3 weeks to create a working prototype of the two modules on a breadboard using a power supply. Once the prototype works, create the PCB design and send it out. The PCB is expected to take 3 weeks, so this puts us ahead of schedule.
10/29 – 11/12	**Write software and test scheme** 2 weeks to write the software running on the microcontrollers. The base module has to connect to Twitter, and the display module has to regularly pull data and rotate the display. Also create a test scheme for when the PCB arrives.
11/12 – 11/26	**Assemble and Test on PCB** 2 weeks to assemble the modules on the PCB and test them for correctness.
11/27 – 12/3	**Final testing and presentation** 1 week for final testing and preparing for the presentation. Rehearse how the presentation is supposed to go. Create the poster board and final report.

Budget

It is difficult to create a realistic budget when the specific components have not yet been selected, so the budget below will consist of prices for general components.

The proposal requires 1 microcontroller with Ethernet, 1 microcontroller without ethernet, 2 wireless radios, 1 power adapter, 1 LCD, 1 battery, 1 battery charger, and 2 PCBs. For simplicity's sake, we will assume that we are using the Arduino platform.

Our total costs end up being well below the cost of two people. The costs can likely be reduced further upon finding cheaper alternatives to these components.

Item	Quantity	Cost per item	Cost total
Arduino Ethernet	1	$60	$60
Arduino Duemilanove	1	$25	$25
Xbee 1mW with shield	2	$45	$90
LCD 20x4 characters	1	$20	$20
Lithium Ion battery 6Ah	1	$40	$40
Lithium Ion battery charger	1	$25	$25
AC power adapter	1	$6	$6
PCBs	2	$15	$30
Total			$302

Issues

The main issues are as follows.

1. Powering the display module's microcontroller, wireless module, and LCD for one working day (10 hours)
2. Making the display module lightweight and small enough to be mounted on a wall or door
3. Having two different systems/boards

Issue 1 is the main concern. Finding a LCD and wireless module that can be powered for 10 hours with a battery that does not violate Issue 2 will be difficult. Issue 3 is the most difficult because it means having work for two microcontroller systems. It is two PCBs.

Additional stretch goals

Ideally, the display module would be able to be powered for one full workweek (50 hours) instead of one workday (10 hours). This could be accomplished by finding low-power components or high-density batteries

Constant Brightness Lamp System

By Shaun Levin

A traditional room is lighted by the sun during the days and by lamps during the evenings. However, what lights the room on cloudy days and while the sun sets? If one illuminates a lamp too early, the room becomes too bright. If one waits until the sun is fully gone, the room becomes too dark. There needs to be a system where the room can be partially lit by lamps and partially by sunlight. This system needs to be automatic and intelligent.

This proposal suggests a lamp system that maintains a constant level of brightness in a room. By utilizing several light sensors, the system would be able to determine who bright each lamp in the room would need to be. The system would then adjust each lamp's brightness as the sunlight shifts in order to maintain a constant luminosity throughout the room.

Details

The system would require four lamp controllers, four light sensors, and one base station. Each of the controllers and light sensors would communicate with the base station wirelessly. The lamp controllers would have potentiometers to control how much brightness the lamp would output. The light sensors would read in luminosity from different spots in the room to know where more light is needed. The base station would use its microprocessor to control everything else.

The lamp controllers would be AC powered and would connect directly into the power for each lamp. The base station would be AC powered and stored in a corner of the room. The light sensors would have to be portable and battery powered so that they could be placed around a room. In order to make batteries easier, we will use alkaline AA batteries.

Schedule

The following schedule is preliminary and subject to change as details are finalized.

9/12 – 9/26	**Planning** 2 week to plan and finalize all features of the project. We will organize teams and milestone goals. We will also create a rough design and create our proposal.
9/26 – 10/10	**Part Selection** 2 weeks to research and pick a microcontroller for the base station, the wireless communication components, the light sensors, and the potentiometers. Components will be ordered as soon as possible.
10/10 – 10/29	**Breadboard prototype and PCB design** 3 weeks to create a working prototype of the base station, one light

	sensor, and one lamp controller. Once the prototype works, create a PCB design and send it out. The PCB is expected to take 3 weeks, so this puts us ahead of schedule.
10/29 – 11/12	**Write software and test scheme** 2 weeks to write the code that allows the base station to control four different lamps and use four different light sensors for information. Also create a test scheme for when the PCB arrives.
11/12 – 11/26	**Assemble and Test on PCB** 2 weeks to assemble the components on the PCB and test them for correctness.
11/27 – 12/3	**Final testing and presentation** 1 week for final testing and preparing for the presentation. Rehearse how the presentation is supposed to go. Create the poster board and final report.

Budget

The following is a general budget for common components. The budget is well within the costs of a three to four person team.

Item	**Quantity**	**Cost per item**	**Cost total**
Arduino Duemilanove	1	$25	$25
Xbee 1mW with shield	9	$45	$405
Digital potentiometer	4	$2	$8
Light sensor	4	$8	$32
Alkaline battery (AA)	8	$0.50	4
AC power adapter	5	$6	$30
PCBs	9	$15	$135
Total			$639

Issues

The main issues are as follows.

1. Finding light sensors able to differentiate minute differences in lighting
2. Writing software that can light lamps in order to maintain constant brightness
3. Having three different types of PCBs to work with

Issue 1 will just take some careful searching of components. Issue 2 will take some serious testing, but it should be achievable in the allotted time. Issue 3 just adds more work, but it should also be achievable.

John Marcoux

<u>Project Idea #1: Rubik's Cube Solver</u>

High-level Description: The Pitch

The name really says it all. It would be a complete, robust, Rubik's cube solving machine that would be 100% portable and be able to solve a Rubik's cube 3x3 in the ballpark of 30 seconds. It would use six stepper motors to turn each side, mounted on an already created rig. In EECS 373, for our final project, our group was able to get this project to work, but not in a manner which was fitting for transportation of any sort. It was, much like Doctor Brehob mention in class, a mess of wires which could fall apart at any time. It would also use, ideally, a camera to read in all of the sides prior to solve, so they could then be process and an, ideally, optimal solve. Although it has already been done, there is still so much that could be done to make it really something special, which is why I am proposing it again. To see how it looks, here is a video of it working from our project last year. As you will see in the video, there could be a lot of improvements.

http://www-personal.umich.edu/~briankir/rubiks/results.htm

A website outlining our process can also be found at:

http://www-personal.umich.edu/~briankir/rubiks/index.htm

The budget would defiantly be doable with a group of 5. The outer portion itself is already constructed, and all we would need would be 6 stepper motors (one for each face), and possibly one more depending on how we wanted to engineer the camera aspect of it. There would also be the cost of the PCB, which may or may not need to be fairly advanced depending on how we handle the aspect of the solve. Also there would need to be a new cube, as the one on there currently is not that great.

The reliability would defiantly be the reason for doing it, as Dr. Brehob has said before that there was interest for this project by the department. It would be completely portable, with no issues of construction upon arrival. Plus I've always thought that it would be a great thing to show to younger kids, like at the Expo, and show them the types of things you can really do here.

The only real safety concern would possibly be the power supplies we would need to power the rig, as it requires at least 2 different voltage levels (probably) and a lot of current. There would need to be surface mounting on the PCB, and possibly welding on the frame to modify it if we thought it was needed to fit the new, probably larger, steppers to replace the old servo motors. There would also be some soldering on the rig itself, on the various wires and such.

Implementation issues

This would be something that, if properly done could be finished by the expo as there has already had work been done on it. Keeping in mind we would be completely changing almost every aspect, it wouldn't be that easy of a task either. If the group could be broken up, some could work purely on the hardware portion of it (getting everything working on the steppers) and the software portion (the PCB design, and possibly the implementation of the solve). This could be done in a variety of ways, depending on what language you plan to have your code for the solve in. Some of them would require a Linux box to run, so that would be an extra expense to factor in. All in all, I think that it is completely possible and could have a fun impact at the expo, and for years to come.

One implementation issue (mentioned earlier) was all of the interference on all of the wires. I suppose there would be less than we originally had (due to not having light sensors) but there would still be issues. There are a few options for this though, one of them being differential signaling. This however, would probably be harder than the other option of somehow using the PCB to eliminate this issue.

Evaluation and or testing with regard to real world issues

This project really doesn't solve a real, everyday problem. Not unless you chose to try to do a Rubik's cube every day; Then in fact, it would and it would probably do it close to as fast if not much faster. It has a large real world application as it would take complicated synergy of hardware and software to do perfectly. It would also show that our team could take a complicated task and produce a very robust, and quick machine which could, hopefully, have the solve in the neighborhood of 30 seconds. It would do this every time, in almost the exact speed.

<u>Project Idea #2: Breakfast Machine</u>

High-level Description: The Pitch

If any of you have seen the classic movie 'Back to the Future', then you may remember the opening scene where Marty Mcfly enters Doc Brown's home before school to feed his dog while the Doc is away. The camera pans around his home showing an elaborate breakfast making machine, which is doing everything from cooking breakfast for the doctor as well as toast, making juice, and everything. This would be, in an essence, what this project would be. It would be a (much smaller) version of this machine, all controlled by a remote (or wirelessly) to make your breakfast while you are getting ready so you can come down to enjoy a great, healthy, warm breakfast prepared for you. You would be able to choose the crispiness of the toast, the amount of time the eggs were cooked for, and the amount of pulp in your squeezed orange juice. This machine would exist to be a larger appliance, probably occupying a fairly large counter space or table. It would have a burner for the egg pan, a toaster for the toast, a juice squeezer for the juice, and probably a food setter portion

Implementation issues

The idea is listed under the wacky section for a reason, it would be a somewhat strange task to complete, as you would need to interface with some pretty strange peripherals (kitchen appliances). These could be accessed by 'hacking' them and opening them up and using their internal circuitry, or simply we could engineer some switch flickers or button pushers, which could also be accomplished. It would also require developing a large procedure for doing all of this efficiently, and compiling it in a way that would be…. Eatable. There also could be a power issue, considering there is probably no way it could be portable (or even run by being plugged into one regular outlet).

Evaluation and or testing with regard to real world issues

This would complete a very specific novel task, making your breakfast every morning for you so you could come down to your kitchen and find it prepared. It would be difficult, but very fun and honestly if it were to work, I couldn't see why I wouldn't want one of these for myself. It is a "first world problem" solver, but I've seen it done too many times on television (family guy) and such to not want to have one of these myself.

Home Security System

Nowadays, home security is always a hot topic. Ideas like designing or improving the security system have been proposed several times.

In my point of view, the most important thing for a home security system is not blocking strangers outside, but providing the host with an interface that they can actually keep track of the indoor information wherever they are. I think a simple private webpage would help. The page would show the condition of lights (on or off) and doors (locked or unlocked). My idea is to design a simple system like this.

Details

First of all, I will need to have some sensors that can capture the condition on lights and doors. All the data will be transmitted wirelessly to the microcontroller board. The design for the system on the microcontroller is much more complicated. It has to receive and process the data from different channels.

After that, the data needs to be sent to the computer through TCP/IP bus. I'm not sure what kind of protocol should I use, maybe a Modbus protocol. Anyway, the computer end needs a simple server and create a webpage with an IP address that only the host can have the access to. The information on the webpage will be updated in real time and also have a list of log history for the user.

With this system built, as long as the host has the internet access, he can always keep track of the indoor condition of his house. If something unexpected happens, he can take measures as soon as possible.

Schedule

The dates listed may not be exact through the semester, but we will try to follow that as much as we can.

9/13 – 10/2	Planning Once the project group is formed, we will get together and make a complete plan for the project. In the first three weeks, we will need to do some research, select boards, get a rough design and modify our proposal.
10/3 – 10/12	Design and part selection The next 1.5 weeks will be spent on selecting parts. The most important part will be the sensor and microcontroller. We also need to run some basic

	tests on wireless modules.
10/12 – 10/21	Breadboard prototype and PCB design Once parts are selected and ordered, a breadboard design will be implemented. The PCB design also needs to be completed and start ordering the board.
10/22	Milestone 1.
10/23 – 11/11	Software development and testing While waiting for the PCB board, the software development for the microcontroller will start. It's a big challenge to build the webpage and we expect to have that done before milestone 2.
11/12	Milestone2.
11/12 – 11/30	Software testing on PCB board Once we have the PCB board, we will start building the whole system. Bugs will be found and fixed. Any additional features will be researched and added depending on the time.
12/1 – 12/5	Final testing, additional features and poster We will continue fixing any remaining bugs and get the poster done. Also, we'll do a preparation for demos.
12/6	Design expo!
12/7 – 12/10	Finish the final report.

Budget

At this point, I couldn't give a proper estimate on the total budget. With a team of 5, we will have a budget over 1000 dollars. I think that's enough for the whole system.

Issues

The main concerns are

1. The sensor used for door and lights
2. Wireless modules
3. PCB design and microcontroller programming
4. Building the webpage

Additional stretch goal

If time allows, I would like to develop some extra features such that when some special cases happen, the host can get notification from the web.

Auto Page-turning Machine

Many people today do lots of reading during free time. But when you hold a book for a long time, you may feel bothered that you always have to turn pages. For people who play musical instruments like piano, they have to stop playing when the book turns to a new page which interrupts the performance.

If there's a machine that can turn the page automatically without using hands, that's really awesome and will do great help to many readers. So I'm planning to design a system with a simple machine to place the book and turn pages just by people shaking the head!

Details

The page-turning system can be implemented in two ways. The first idea is to use an embedded camera that can detect the moving of the head. This is a little bit complicated because the algorithm is difficult to implement and needs hardware support. Another idea is to use a sensor of gravity acceleration. But this device needs to be worn on the ear and transmit signal wirelessly to the machine.

The most challenging part for this project would be the design of the page-turning machine. This is more than a simple microcontroller but should be able to turn the page under certain command. I still have no idea how to realize this because it looks like a mechanical issue.

The software programming part I believe would also be time consuming. For the hardware, all parts need to be selected properly.

Issues

Overall, this is an interesting project with lots of challenges. I couldn't come up with a very detailed proposal at this time and I'm even not sure whether this is doable. Therefore, I wish if someone else is interested in this or any similar project, he could have some ideas that can help.

Home Appliance Automation and Energy Use Optimization

Energy usage at home is something that if left unchecked, can get to be expensive pretty quickly. High power devices such as electric ovens, space heaters, air conditioning units, and desktop computers built for gaming or other applications that require massive amounts of graphics processing power can contribute to an expensive energy bill if left on overnight or when no one is at home to use said devices. In the case of the electric heat sources like a stove or space heater, leaving any of them on while unattended for an extended period of time can be a dangerous and expensive mistake.

High energy bills are frustrating and environments with irresponsible or forgetful roommates contributing to the high energy cost only compound the frustration when the statement arrives. Home automation systems exist to help reduce this, but they can be expensive, require installation of a central control panel, or still require the use of a remote control in order to function. With irresponsible, messy, or forgetful roommates, the remote control is just another thing that can be lost easily.

Project Details

To minimize unnecessary energy usage, most devices found in the average home or apartment can be turned off when nobody is home or all residents are asleep. To implement this kind of logic in an automated system, the control system will be able to learn these patterns over time through the use of various sensors and a real time clock. All devices controlled by the system will have a wirelessly controlled switch at the outlet that will communicate with a base station when the outlet needs to be enabled or disabled based on occupancy patterns learned by the base station. Each outlet module can be paired with a base station for automated control. The benefit to this approach is that the entire control system can be implemented with no additional hardware or wiring, no tools, and very minimal setup.

Project Timeline

9/12-9/28 (2 weeks)	**Research**	• **Decide what types of components will be necessary to make a base station and four outlet modules** • **Research systems for providing power to each module** • **Research appropriate wireless communication systems based on what source of power will be used** • **Begin research and development of a machine learning algorithm that will learn occupancy patterns within a reasonable amount of time** • **Decide if manual pattern programming should be done using computer software or if should all be done on the base station**
9/19-10/3 (2 weeks)	**Component Selection**	• **Base station: decide which sensors should be used to detect no occupants and when occupant are asleep** • **Outlet module: decide which components should be used to break an AC circuit and how toggle current flow between the wall outlet and the output of the module** • **Pick wireless radios for each module type**

10/3-10/26 (3 weeks)	**Design & Prototyping**	• **Design the outlet and base station circuitry on breadboards and start developing software** • **Once the breadboard circuitry is working correctly, design and order a PCB for each module** • **Design simple enclosures for each module that can be manufactured relatively easily and at a low cost**
10/26 -11/23 (3 weeks)	**Fabrication and Software Development**	• **Build the circuitry on the PCBs and verify that the hardware wiring is correct** • **Develop and test the software that will be running on the final product** • **Manufacture the module enclosures**
11/23 – 12/3	**Software Refinement and Project Report**	• **Fix any bugs that may be remaining at this point** • **Incorporate any additional software features that can ultimately make the system easier to use or more reliable** • **Prepare a poster for project presentation** • **Write up final project report**

Cost Estimates

Providing an accurate cost estimation without knowing which parts will be used might not be the most accurate estimation, but gives a starting point to work with if anything else can be included or if components need to be cut from the project

Base Station Cost

Microcontroller Board(1)	<$60.00
Wireless Radio(1)	$40.00
Character Display LCD for system status(1)	$15.00
Keypad for system input(1)	$4.00
Ambient Light Sensor(1)	$5.00
Motion Sensor(1)	$10.00

Outlet Module Cost

Microcontroller (1)	<$20.00
Wireless Radio (1)	$40.00
Outlet Switch(1)	$8.00
Solenoid(1)	$5.00

The base station will cost about $134, allowing $40 for a Bluetooth modem and up to $60 on the microcontroller. These estimates are most likely far above what will actually be spent on the base station components. Each outlet module will cost about $73, using the same Bluetooth modem as the one used for the base station and allowing up to $20 for a microcontroller. Again, these estimates are probably much higher than what will actually be spent on the outlet module components. The microcontroller cost is highly variable based on capabilities and the radios chosen can also be swapped out with more cost effective alternatives. The $40 estimate is based on a modem found on SparkFun. In total, One base station will be built and four outlet modules, for a total cost of $436, not including shipping or miscellaneous component costs

Potential Implementation Issues

There are a number of stumbling blocks that will arise during the development of this project, the most significant probably being radio interference. Other problematic aspects of the project might include the algorithm development and learning the patterns relatively quickly or how to handle modules that aren't responding to the base station's commands. The control system functions over the course of an entire day, so testing the algorithm or any other time based aspect might not be very straightforward.

Additional System Features

If development of the core system functionality proceeds well and there is enough time to spare, other features that could be added include the ability to manually program patterns or manage connected modules from a program on the user's computer over a USB connection or even over a local network. With the proper setup, it would be possible to remotely access the base station and view or change the states of connected modules from any device with an internet connection.

Robot Guitarist

A lot of people play a musical instrument, and the guitar is a popular choice because it is very versatile and can be incorporated into any musical work. It is also portable and guitar players can provide entertainment for themselves and others nearby with minimal setup. Learning to play a guitar takes a lot of time, practice, patience, a fair amount of dexterity, and some level of musical inclination. Not everyone can learn to play a guitar and their reasons may be any or all of the above, or some other reason, but many people enjoy the sound of guitar music played live. To provide a way to listen to live guitar music at any time without knowing how to actually play one is a task that can be automated due to the guitar's simple structure

Project Details

The goal of this project is to provide a mechanism to automate guitar playing for those who appreciate the sound of a live guitar without being able to play one themselves. Such a device would be mounted on the neck of the guitar and would be able to press all the frets and strum any or all of the strings for any given chord. An assembly of solenoids would move up and down the fret board to press each fret and another assembly would be at the body of the guitar to strum each string as needed (a pick assembly, so to speak). The device would be able to load and play tablatures from an SD card and would have a small display and controls for browsing the SD card, as well as other device options.

Project Timeline

9/12-9/28 (2 weeks)	**Research**	• **Devise an efficient method to mechanically imitate a guitarist's finger motions on the fret board and while strumming.** • **Develop an algorithm to parse tablature files and possibly a device specific tablature file format that would be easy to create**
9/19-10/3 (2 weeks)	**Component Selection**	• **Decide what components will have fast enough response times and low noise output in order to not interfere too much with the guitar sound**
10/3-10/26 (3 weeks)	**Design & Prototyping**	• **Design a mounting mechanism** • **Design the circuitry on a breadboard and route wires to the solenoids and motors in such a way that they do not interfere with the device functionality**
10/26 -11/23 (3 weeks)	**Fabrication and Software Development**	• **Build and verify the correctness of the circuitry on the PCB** • **Develop and test the software that will be running on the final product** • **Manufacture the mounting mechanism and each moving assembly**
11/23 – 12/3	**Software Refinement and Project Report**	• **Fix any bugs that may be remaining at this point** • **Prepare a poster for project presentation** • **Write up final project report**

Dedicated Handheld Device for Encrypted SMS Traffic

1. Overview

People in today's society who are concerned about their privacy may feel that carrying today's cell phone may leak more information about their personal lives than they wish to expose. For example, there is no guarantee that the microphone on a cell phone may not be remotely turned on and used to illegally monitor a person's surroundings. In addition, voice and Short Message Service (SMS) data that is sent on a phone may be encrypted between a phone and a cell phone tower, but not between the phones of two individual people. This may lead to warrantless monitoring of a customer's cell phone usage (either voluntary, or enforced by a government), a highly controversial policy.

In addition to privacy concerns, old working cell phones are constantly being replaced with new ones for their new features or designs, leading to millions of discarded cell phones every year. If people had a greater monitary incentive to recycle these phones, the amount of cell phone waste may dramatically drop.

The proposed product would help to satify both of these markets by implementing a handheld device that is capable of end-to-end SMS encryption, as well as reusing perfectly good parts from recycled old cell phones.

2. Details

The proposed device would only support communicating over SMS. No audio components will be included in the device, which should help to keep costs low. To take avantage of old recycled cell phones, all well-working and suitable hardware components will be used from old phones, including the antenna and any chips necessary for interfacing with the antenna. The device must also include a battery capable of powering the device for at least a full day of usage. Public key encryption will be used to provide end-to-end encryption of the SMS messages, and additional storage will be necessary to save contact phone numbers and public encryption keys, as well as the device's own private encryption key. An LCD will be necessary to compose and display messages, as well as a key pad for input.

The whole device should be small and light enough to carry in one's pocket comfortably. In this manner, it should be just as compact and light, if not more, than the average cell phone on the market today.

The device should also have a method to generate and reset (or support multiple) public and private key pairs. Ideally it would also include a way to backup encrypted and/or decrypted messages and private keys to one's personal computer.

3. Schedule

These dates are general ideas and when particular parts of the project should be completed and are not final.

9/12 - 10/1	**Planning** In the three weeks after project groups are formed, we will need to define specifics capabilities of the prototype device, carefully research and select parts, create a preliminary design of the product, and compile all of this information together into our proposal. Any show-stoppers for the project should be found during this time.
10/1 - 10/12	**Part Selection and Design** It is expected that during this time inexpensive pay-as-you-go phones will be purchased and used to begin designing the communication process between these recycled cell phone parts and other parts necessary for the device (such as storage media). Other parts such as the processor and storage should be chosen during this time frame. A high-level design should be created to ease the development of PCBs later.
10/12 - 11/2	**Breadboard prototype and PCB design** Once parts have been selected a breadboard design will be implemented for the device. Testing will be done to insure the hardware design is correct, and after this we will order PCBs. Development and shipment of the PCBs is expected to take three weeks.
11/2 - 11/26	**Software development and testing on PCB-based system** After the hardware design is finalized we will be able to start development of the software. Once the boards arrive we will assemble them and begin testing to discover any remaining hardware issues (such as, but not limited to any possible power problems due to the extra overhead of encryption). A working product should be finished by November 26.
11/27 - 12/3	**Final testing and report** Targeting project completion by 11/27. Any remaining minor bugs should be fixed during this time.

4. Budget

One of the goals of this project is to reuse as much technology from old phones as possible. For this project, that means purchasing inexpensive pay-as-you-go phones to dissamble and reuse the antenna and any additional necessary chips.

For reliable testing and demonstration of multiple devices, no fewer than three devices should be built. In addition, a small cell phone plan will need to be purchased (this could be included as part of the recycled phones).

While a realistic budget is difficult to determine at this point in time, the following table should generally describe how much it would cost to implement this design. With a team of five people, we would be about $800.00 under budget.

What	**Quantity**	**Cost per item**	**Cost total**
Cell Phone	3	$10.00	$30.00
Cell Phone Usage	3	$20.00	$60.00
Processor	3	$20.00	$60.00
PCB	3	$15.00	$45.00
Keypad	3	$10.00	$30.00
LCD	3	$20.00	$60.00
Misc Parts	1	$100.00	$100.00
Shipping	1	$75.00	$75.00
Total			$460.00

5. Issues

The largest concerns at this time are

1. Interfacing with the chips and antenna from the recycled cell phone
2. Implementing the protocol and encryption scheme
3. Dealing with power issues

Issue 1 is a complete unknown (this will probably be the most difficult part of the project) but should be possible if the hardware parts from the phone are all standard and well documented. Issue 2 shouldn't be conceptionally difficult for anyone who has experience with encryption, but it may be difficult to make the protocol work within the character limit restrictions of SMS. Issue 3 may turn out to be a major problem if the encryption causes a significant amount of extra power to be drawn from the battery.

6. Stretch goal

If time allows we will see if it would be possible to interface the device with a personal computer in some way to backup encryption keys and saved messages.

Self-locking House

1. Overview

The home is often the most important place for any family, and holds the most cherrished items and possessions. It is always a tragedy when a burglar is able to easily take these possessions if the house doors were accidentally left unlocked. The proposed product would prevent this issue from ever happening by automatically locking every door in the house when the last person leaves.

Each person would carry a small device that wirelessly communicates to a base station in a house. When all devices have left the range of this device, and if there are doors that were left unlocked, this system would automatically lock them.

2. Details

The base station would be wired to every door in the house. Once every wireless key has been removed from the wireless range of the base station (assumming that there is no one left in the house), the base station will send out a request to every door to lock itself. Each door would have its own physical locking device. Manually unlocking a door with a physical key will have the door's device send a notification to the base station to put the house in manual mode, as the house should not lock itself as soon as it was manually unlocked.

Each wireless key should last for about a month at a time on battery power to avoid the nuisance of constant recharging and/or replacing of the battery.

The project would involve developing the base station, a protocol to communicate between the base station and wireless keys, another protocol to communicate between the base station and door locking devices, and the design and building of the mechanical door locking devices.

3. Schedule

These dates are general ideas and are not final.

9/12 - 10/1	**Planning and Part Selection** In the weeks after project groups have formed, the protocols to communicate between all devices will need to be developed. Parts will need to be chosen for the base station, wireless keys, and door locking devices.
10/1 - 10/12	**Design** After the parts have been chosen, the design of each of the components will need to be completed. A high-level schematic should be created to ease the development of PCBs later.

10/12 - 11/2	**Breadboard prototype and PCB design** Once parts have been chosen and the design of each component has been finalized, PCBs shall be created and ordered. Development and shipment of PCBs is expected to take three weeks.
10/12 - 11/2	**Door locking device design** At the same time as the PCBs are developed and shipped, the physical locking mechanism and controller for each door must be designed and created. This part would likely best be completed by a separate team in the group who is more familiar with designing the physical design of the locking device.
11/2 - 11/26	**Software development and testing on PCB-based systems** During this time the software shall be finished and the whole system will be tested.
11/27 - 12/3	**Final testing and report** Targeting project completion by 11/27. Any remaining minor bugs should be fixed during this time.

4. Budget

A realistic budget for developing this project is difficult to determine at this point in time. Due to the large scope of this project, it is likely that six people would end up working on it. At a budget $250/person, this would lead to $1,500 of money available.

The following table is a rough estimate on how the cost would break down.

What	Quantity	Cost per item	Cost total
Processor (Base)	1	$10.00	$10.00
Processors (Door)	1	$10.00	$10.00
Processors (Key)	1	$10.00	$10.00
PCB (Base)	1	$15.00	$15.00
PCB (Base)	1	$15.00	$15.00
PCB (Key)	1	$15.00	$15.00
Misc Parts	1	$600.00	$600.00
Shipping	1	$75.00	$75.00
Total			$750.00

5. Issues

The largest concerns at this time are

1. Designing the PCB and additional hardware for each individual component (base

station, wireless key, and door locking device)

2. Developing the protocol to communicate between each individual component

3. Designing and implementing each door's physical locking mechanism

4. Solving the issue of having the house lock yourself out if you have no access to any keys and walk away from the house

Issues 1 and 2 seem reasonable to complete, but are just a lot of work, perhaps even for a team with six people. Issue 3 is much more of a mechanical issue and requires much further design on the exact implementation. Issue 4 may require some additional thought for the best solution, but adding an additional interface to the base station, as described in the Stretch goal section below may provide a solution.

6. Stretch goal

Integrate the base system into a home's wireless network to set additional configuration options, such as times of day that the system should be active, and to manually override any automatic locks.

University of Michigan
Joe Romeo
EECS 498
Advamced Embedded Systems

RACE

09/09/2012

Project Proposal

R.A.C.E.

"BEEP! BEEP! BEEP! BEEP!". It is a cold winter morning in Michigan. A young engineer wakes up to the sound of an alarm after only a few brief hours of sleep… THWAP!! The alarm is silenced and the young engineer drifts back into a slumber.

The sun is now beating through the window. The young engineer wakes again to the warmth of the sun. He reorients himself, trying to make sense of the numbers on his alarm clock. The numbers finally come into focus reading "12:00 PM". OH NO, the young engineer just slept through his final exam!

This is a problem that has plagued people for decades. Alarm clocks are just too easy to turn off! They are often conveniently located on a nearby desk or shelf, and often don't require a person to get off the bed. Even those people who strategically place their alarm clocks in a far side of the room can navigate to the alarm clock on auto-pilot to turn it off and return to bed.

This is why I propose R.A.C.E as a solution to our problems. R.A.C.E. stands for Runaway Alarm Clock Energizes! Instead of having a stationery alarm clock, we will put the alarm clock on wheels and have it run away from the user. This will not only get the user out of bed, but it will also fill him with energy as they must reorient themselves and focus to catch and turn off the clock. Unlike a clock across the bedroom, this one will be in a different spot everyday, and will therefore force the person to leave autopilot.

Details

R.A.C.E.'s final price(in mass production) should cost between $50-70 to produce. It will about the size of the 498 lab cars, but less than half the weight. It should also drive faster than walking pace (to make it more difficult to catch). The sound output for the alarm should be equivalent to a similarly priced alarm clock. It should have a display of the current time. There will be multiple buttons for setting the current time, as well as setting the alarm. The alarm clock car will "race" around the room and will use a various assortment of sensors(mostly infrared as they are cheap, reliable and low power) to keep from running into objects. To keep track of the time, a basic microcontroller timer will be used. The alarm clock must run on rechargeable batteries(it will be recharged while not sleeping). The car will be a hacked off the shelf remote control car.

Implementation Issues

1. How will we keep timing accurate, so the user will not have to reset the time often? Possible solutions: 1) microcontrollers with extremely accurate timers. 2) Daily Internet Time Sync.

2. Finding sensors and a remote control car that match the end budget.

Budget

Price per	Qty	Part
$50.00	1	Remote Control Car
$15.00	5	PCB
$1.00	5	MSP430
$5.00	5	Wiznet SPI to Ethernet Chip(maybe)
$20	1	Speakers
$20	1	LCD Display
$60		Misc. parts (caps, res, sensors, ect.)

Evaluation

1. The sound level should be comparable to a normal off the shelf alarm clock.
2. The car should be faster than the average morning stumble pace of a person(tbd by testing).
3. The clock should keep accurate time over at least the course of a week(should be longer, but a week should actually be testable in terms of this class and grading).
4. Robot should run around and make annoying noises for at least long enough to wake up the average person before it needs a recharge(5-10 minutes?).

Issues

1. Outputting loud audio using a microcontroller.
2. Keeping accurate time.
3. Avoiding objects while running away from user.
4. Having batteries that don't need to be replaced more than once a year.

Are the Issues Solvable?

1. Even small 8-bit microcontrollers like the atmega328p can output audio(it's even an example sketchbook on arduino).
2. Here is a pdf showing how to set up timers for RTC(Real Time Clock) applications on the MSP430, one application of which they say can be a watch or a clock. (http://www.gaw.ru/pdf/TI/app/msp430/slaa076a.pdf)
3. Avoiding objects is a fairly simple task and can be achieved by using a few cheap infrared detectors.
4. I'm not quite sure, but it looks like most rechargeable batteries can recharge at least a couple hundred times.

Schedule

Weeks 1-3	Research different microcontroller, rechargeable batteries, speakers, LCD Display and remote control car and see which ones look like they will work best for the application. Start work on proposal
Weeks 3-5	Buy parts that we've researched, and begin testing to make sure parts are adequate. Half of the team will test to see if the batteries will last long enough while driving the car + audio + LCD (testing can probably be done via arduino) display given our project requirements. The other part of the group will begin prototyping and working on the PCB design.
Weeks 5-8	Finish figuring out how to interface microcontroller with peripherals and finalize PCB design. Send out PCB and continue to work on software(ie, make sure timing is accurate, the car avoids walls well, and audio is playable). Write software tests to check PCB design.
Weeks 8-10	Receive PCB and begin testing. Software before this point should have been solid on the prototype, so things should be working and we should just be running the created software tests.

University of Michigan
Joe Romeo
EECS 498
Advamced Embedded Systems

NO KEYS

09/09/2012

Whacky Project Proposal

You are the Key!

It's time for basic home security to enter the 21st century. Keys are burdensome and easy to lose, and a quick google search shows at least a dozen techniques for breaking into a home with normal locks quickly and quietly. Also, locks must be replaced often in rented places because keys are easy to duplicate. Biometric fingerprint locks are a great solution to these problems as they are easily reprogrammable for new people and it's hard to duplicate fingerprints, but these types of locks tend to be very expensive.

This is why I propose to create a system that uses facial recognition to unlock a door. With the increase in open source facial recognition software in operating systems, C++/java/C/ect APIs, and web API's, facial recognition can be cheap and easily implemented.

Details

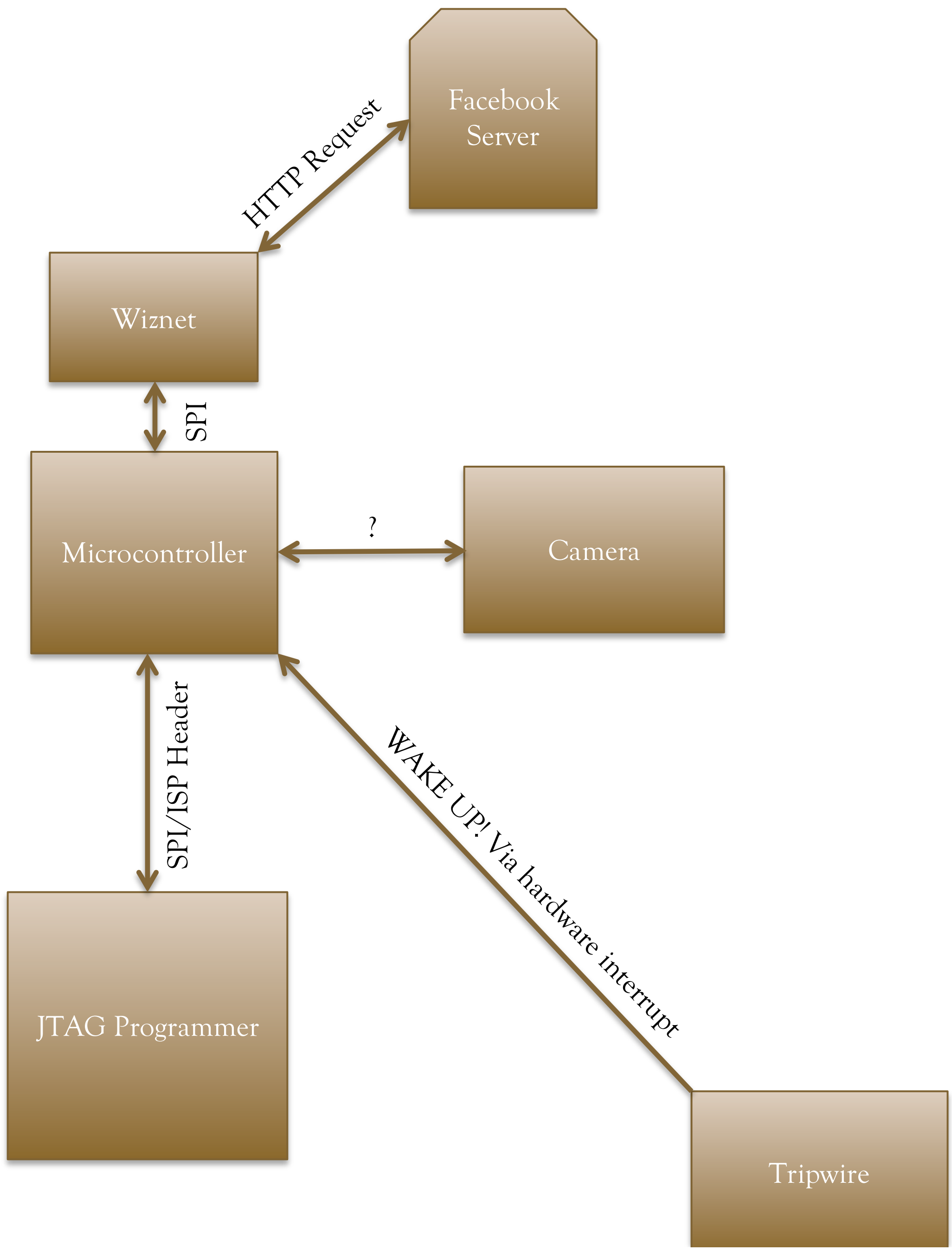
A tripwire sensor near the door will trigger the facial recognition system. The tripwire will send a signal to the camera to turn the camera on (that way the camera isn't wasting energy all day). The user will then know to put his face up to the camera so the camera can snap a picture. Using a microcontroller and a chip that converts SPI to ethernet, allowing for easy internet connection with low power microcontrollers, the picture will be uploaded to an API provided by facebook that allows a user to find out who the picture is of, if the person exists in their database. The Facebook server will send us the name of the person in the picture, and then microcontroller can then decide whether access should be granted or not.

Budget

Price	Qty	Part
$5.00	4	Atmel Microcontroller
$15.00	5	PCB
$20.00	1	Programmer
$5.00	4	Wiznet SPI to Ethernet Chip
$50-100?	1	Camera
$40	na	Misc. parts (caps, resistors, regulators, oscillators, ect.)

The previous budget is fairly conservative. Money will definitely not be the issue with this project.

High Level Block Diagram



Walking Aid for the Blind

Being visually impaired in today's world is difficult. Not being able to perform everyday tasks dramatically reduces quality of life. One main issue is the inability to be able to move. As of now, the standard for walking aids is a stick that person slides along the ground. The user uses feedback from the stick to gauge whether or not they are able to move forward. This system is still not very advanced, nor is it a reliable way to move.

The device that would be created here would improve blind people's ability to move around in dynamic environments. It would be some sort of handheld device that takes in information about the surrounding environment, and transmits that information in a useful manner to the user.

Currently there aren't many solutions for making life easier for the blind. This is most likely due to the fact that blindness is not a very prominent problem. Nevertheless, a device such as this has the possibility of dramatically improving people's lives.

Details

The device would need to have many features in order to be considered a viable alternative to the current stick. The main feature is the ability to detect objects that pose a risk to a walking person.

A device like this would have two parts. The detection system, and the notification system where the detection aspect would analyze the user's surroundings, and the notification part would signal to the user the device's findings.

There are many possible ways to implement the detection system. One possible way is to use a small camera that would work in tandem with computer vision algorithms to gain information about the environment. Another possible method is to use a sonic distance sensor which would provide feedback about the distance between the use and the objects in front of him. Using this method would mean the system is dealing with less information. To deal with this, there probably would need to be some sort of scanning motion to get enough information to get a decent picture of the environment. This could involve either a physical scanning motion that the user performs, or a built in scanning of the distance sensor.

Given that this device is for the blind, the notification about the environment will have to be non-visual. Considering it seems most likely to be a handheld device, vibration would seem to make sense in that regard. A simple method would be to use a more violent vibration for warnings about upcoming objects, and little/no vibration meaning the coast is clear.

Schedule

Reasonable schedule is as follows.

3 weeks – Planning. Choosing the correct way to implement the device is very critical. Research must be done to ensure that the parts and algorithm choices are the best possible. Once past this date, changing overall design would be difficult. Ordering these parts occurs once they are selected.

3 weeks – Development of the PCB and hardware design. These two parts go hand in hand. Ideally, we want to build a prototype of the design along with PCB design so we know what works best.

2 weeks – Algorithm design. While the PCBs are on the way, we would use the prototype to begin the design of the software used.

2 weeks – Testing. Once the PCBs come, we can use them to improve the algorithms and make sure it all works as best as possible.

1 week – Tech writing and final bug fixes. Basically make it perfect.

Budget

With a couple of design possibilities, the budget will have a range. Going the route of a camera would require a small cheap camera, and a fairly powerful processor that would be able to handle the vision algorithms. Small and powerful enough cameras are available at around $20 and a decent processor might even be something like a Raspberry Pi, at least for prototyping, for around $30. If you went with the distance sensor, the main components would be the sensor, such as a PING sensor for around $20 but with a much lighter processor, something along the lines of an ATmega for around $7. Tacking on the cost of a PCB and a container, the cost of this device could be anywhere from $50 to $100. This may seem like a lot, but considering this would be a useful device for a long time to come, it would be well worth it.

Issues

Like all projects, there are many possible issues that could come up.

One issue is with low lighting environments. The use of a camera in the detection system would mean that a light would probably be needed. This may or may not be a problem except for the vision algorithms used because a light may be helpful to alert others of the user's existence too. A distance sensor would not have this issue.

Using a distance sensor poses another large issue about how to get relevant information from just a single sensor. Interpretation of the data would have to be really well done if this device were to be as useful as it could be.

The last issue is about the notification to the user. It is pretty tough to transfer detailed information just by a vibration. If this is the case, a simple detection system might be sufficient to use. On the other hand, it might be better to have a simple output of safe/unsafe, even if it involves very detailed environment scanning.

Cheap Home Security System

College town safety is an important issue. Home thefts seem to be increasing in recent times. The problem is that current home safety solutions are expensive and fairly permanent. This project would involve cheap sensing devices that would allow for decent protection of your home.

Details

The main device would be some sort of tiny box that can be attached to and movable object such as a window or a door. It would most likely involve some sort of accelerometer that would trigger a notification to the people who live in the house if they have the house armed.

Many options for notifications exist. Email or SMS based notifications seem like simple and efficient solutions. People check both of those constantly.

Another point to think about is how to make capturing the event useful. Coming back to a home where a suspected robbery is taking place is dangerous, and calling the police on a false alarm is also undesirable. Putting some sort of camera in the device that would record when triggered would be nice, but that could raise the cost but quite a bit more.

Schedule

The main steps of building this system are:

2 weeks - Deciding on the parts. Which ones to use and if you would want to incorporate a camera.

2 weeks - PCB design.

2 weeks - Software design. Fairly minimal since is a reaction system, but could be complicated if you are dealing with video. It might be better to do the main notification in a central location so that the device only has to notify a trigger. This part might take some time.

2 weeks - Usefulness testing. Does the device work to detect unwanted intrusions. This is the most important part since it is expected to work and have practical implications. This involves tweaking the notifications and figuring a physical setup that works.

1 week - Report writing.

Budget

Cheap accelerometer from SparkFun -- ~$25
Wifi connector from SparkFun -- $35
ATmega controller -- ~$7

Total -- ~$67 per device.

Issues

Cost seems to be the biggest issue. With a house having a bunch of entry points, a device would be needed on each one. At $67 per, that can add up very quickly.

Another problem could be telling when the system is armed. In a college environment, people living in a house leave and enter at often and random time periods. Granted, this means that a robbery would be difficult during these times, but being protected is still a necessity.

Plug 'N Play TV Trivia Game

In the United States, the market for toys and video games is huge. The toy industry's annual total impact is nearly 80.9 Billion dollars.[1] My idea for a project is to build a battery powered trivia game that can be plugged straight into a TV and played with up to four players. The unit itself would be fairly small, lightweight and aesthetically pleasing. It would have four "buzzer" type controllers that plug into the device that the players will use to buzz in to answer a question and a single keyboard to type the answers in. It will connect to the TV via some sort of graphical output, preferably HDMI so that we could get audio out of it as well. The cost should be fairly cheap in order to entertain a target market. It should be very reliable in every aspect of its design. It would be very frustrating if you were in the middle of a game with your friends and the device just crashed. Safety isn't too much of a concern with this device, but it isn't something that should be overlooked. I can't think of any safety issues at the moment, but we should be constantly thinking about them and dealing with them as they arise in the design process.

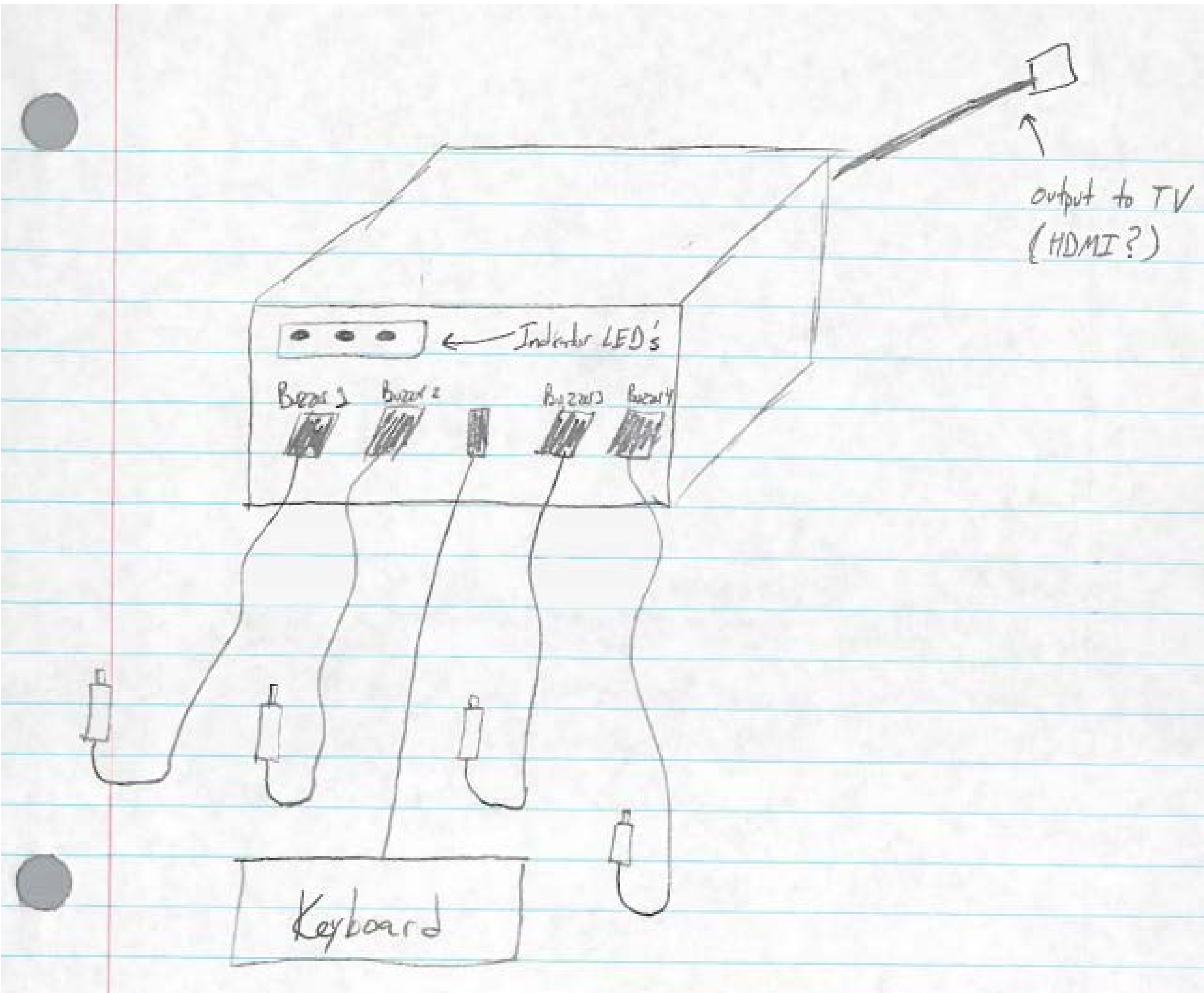
Details

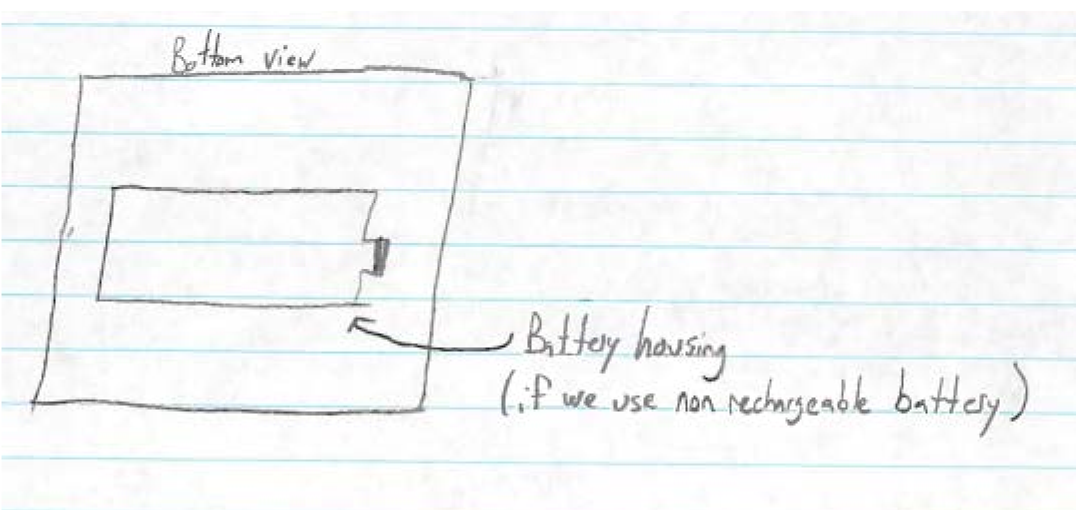
To build a fully functional trivia game, we would have to design and program the user interface for the game. This includes coming up with a game structure and format, designing the graphics, and integrating audio. This can be a challenge for someone that has no experience in this area. Along with that, we would have to integrate the physical inputs with the user interface. An issue that I want to deal with is the timing of the buzzers when answering a question. I've played different trivia games on consoles that gave a clear advantage to one player over others, which isn't fun for the players that have the disadvantage. This also has to be low power so that it can be run off standard batteries that a consumer can buy at the store. Another feature we could look at is putting a rechargeable battery in there. Design issues with that might be cost and complexity.

A Raspberry Pi[2] could be a good platform to use for this, it has everything we would need at a low cost and small design. The only issue with this is that it might take too long to get ahold of one.

[1] http://www.toyassociation.org/AM/Template.cfm?Section=Industry_Statistics
[2] http://www.raspberrypi.org

(very) Rough Design Drawing





Schedule

These dates are a rough idea of what we should be working on at a given time. They are not hard deadlines but we should try to follow them as closely as possible. Any input to improve this schedule is welcomed.

9/12 – 9/26	Planning Two weeks after we get into a group we should have all the research done on what chip(s) or devices we want to use as well as a format for the trivia game and a rough idea of what the interface will look like.
9/26 – 10/3	By this week we should get the parts that we want to use and start programming the user interface. We should also design a housing for the device.
10/3 – 10/24	Breadboard and PCB design At this point we should be building a breadboard design that we can use to design and test our implementation. Getting the circuit designed should be a top priority so that we can order a PCB as soon as possible.
10/24 – 11/14	More testing Given three weeks for the PCB to arrive, we will continue to work on testing and debugging of the integration of the user interface with the user inputs. We should be ready to put our boards together once they arrive.
11/14 – 11/27	More testing of the system once the PBC arrives. Testing of the timing will be an issue. Bugs will be fixed, PCB issues found and fixed, anything else that we might want to add or remove will be decided at this time.
11/27 – 12/3	Goal to have the project completed by this week. Final testing, poster for design expo, and final design report will all be done in this week.

Budget

Our budget should be as small as possible, in order to make the device cheap to produce to maximize profits. The budget is something I would like to discuss as a group, so I'm not going to put any hard numbers on anything, but keeping it as cheap as possible is a main goal of this project.

Issues

Our largest issues at this time are:

Choosing the correct parts in order to meet project goals

Designing and developing a user interface

Designing the physical inputs for the user interface

Integrating the inputs and the user interface together

Timing of the physical inputs in order to not create an advantage to any player.

Whole Song Digital Tuner

When playing a musical instrument, playing in tune is one of many things that a musician has to be aware of when playing a piece of music. Talented and well-practiced musicians can listen to any pitch and tell be ear whether it is in tune or not. For other musicians this can be a difficult task, and can be aided with constant practice and the use of a digital tuner. To use today's tuners you play one note and watch the digital readout as the needle tells you whether you're too flat or too sharp. This method works great for long sustained notes, but for shorter consecutive notes, the digital readout is often lags behind in trying to listen to a note and give you output on it.

Details

The idea is to build a tuner that will listen to a musician playing a whole song and give a report at the end of how in tune the song was, areas that can be worked on, and other statistics about the song session. For the tuner to know what song you are playing there would have to be some way for a song to be programmed into the tuner. One possibility would be a MIDI file stored on a removable memory stick. This would allow for quick changing of songs and programmability on another platform such as a PC. You could also plug the device into a PC where you can download the data for a "session" that you have already played or program a new song into the memory card. This project would most likely involve a lot of signal processing, but is simple enough that it could be done in one semester.

The design would be lightweight and battery powered. Small enough to fit in your pocket and/or the case of an average sized musical instrument. It will have a digital readout with an easy enough interface as to not confuse a beginner to electronics. The budget for a design like this would come mostly from parts and the PCB.

Issues

Finding a microphone that works with this design.
Programming the PC software and getting it to interface with our device.
Getting the correct timing to work with a song, how the notes change over time.

The Embedded Vehicle Driving Position Monitoring System

Pitched by Shili Xu {stevenxu@umich.edu}

Driving in an improper physical condition, like being exhausted or extremely sleepy, can be lethal; and an important omen for the possible accidents is driving out of the lane that the vehicle is in. In some states in the US, like California, there are some tiny blocks on the boundary line between two adjacent lanes, thus when driving across the lanes, the blocks can cause significant noise and oscillation to warn the driver. This setting, though effective if fully equipped on all roads, is very costly and can be quite annoying, as every time the driver merge to another lane, the noise and oscillation will occur.

Thus, an automatic monitoring system embedded in the vehicle directly can greatly reduce the annoyingness these facilities could bring out while ensure the safety of the driving vehicle on the road, while at the same time reduce the expense of the government on these on-road warning facilities.

Implementation Details

The system is based on the visual system directly, without using other fancy sensors, such as the infra lasers. Currently, there is a great increase in the use of the visual system in the rear end of the vehicle to help driver better back it off. And this lane position monitoring system can be a good application for a vehicle to make use of the frontal camera.

In terms of the algorithms used in the system, there are a lot of lane tracking methods existing in the computer vision literature, for example, [1] provides a good way to detect the lane by estimating vanishing point in the frame, we can compare the algorithms together and find an efficient and effective method to implement on our embedded system. Thus, from algorithm level, an on-desktop development of the algorithm and a convert to the MCU stage should be accounted.

According to the specification above, the system is composed of the wide-vision frontal camera, and a microcontroller and is fully embedded into the vehicle.

Schedule

9.12 – 9.17	Keep research for the suitable lane tracking algorithm for an embedded system; specify the design requirement for the system; assign the task for each group individual
9.17 – 10.1	Apart from the algorithm level search, further research and select the necessary components for the embedded system, come up with a rough design; write down all the specifications and form the final proposal.

[1] **M. Nieto**, "Detection and tracking of vanishing points in dynamic environments," **PhD** Thesis, Universidad Politécnica de Madrid, 2010.

10.1 – 10.14	In the first two weeks of the execution, divide the team into two groups. First group try to test the lane tracking algorithm on the desktop and try the think about a way to modify it, making it more memory efficient to better adjust into the embedded system; the other team will look into the candidate microcontrollers for the system, research on their parameters and decide which one to be selected; moreover, come up with a brief design for the PCB suitable for this MC.
10.15 – 10.29	In these two weeks, all the members will focus on the PCB design part. Some preliminary test on low-level microcontroller, if find out the pre-selected one is not powerful enough for the task, higher level processor is then chosen; verify the design of PCB and make the order.
10.30 – 11.12	The following two weeks will be about the prototyping on the desktop for the lane tracking algorithm; at this stage, the algorithm should have a relative mature performance and can be adjusted to the embedded system. Meanwhile, once the PCB is ready for use, it will be tested before we port the software into it.
11.13 – 11.28	Implementing the algorithm into the embedded system, and test will be taken especially for two aspects, computing power exploitation and lane tracking accuracy. Trying the find all the possible bugs, get the revision done if needed.
11.29 – 12.5	Final testing, fix minor bugs; try adding better demo affect, like some visual warning signs for the system to make it better suitable for a vehicle driver; prepare for the expo.

Budget

As this embedded monitor system mainly composed of a camera and a well-suited microcontroller, the system will not be too costly. As for the camera, a normal webcam can be used here in our developing stage, but if to be set up in the vehicle, a smaller and surely more expensive one should be used. As for the microcontroller, the system requires enough memory storage for a real color image from the camera, thus making the microcontroller higher level than normal. The budget for the two items will be as following:

Item	Cost($)
Microcontroller(try using Arduino used in lab first)	80
PCB	75
Camera (WebCam)	50
Misc Parts	50
Shipping	50
Total	305

[1] **M. Nieto**, "Detection and tracking of vanishing points in dynamic environments," **PhD** Thesis, Universidad Politécnica de Madrid, 2010.

Issues

The concerns about the project:

1. Can the algorithm be simplified enough to port the embedded environment?
2. Can the system be robust enough to work under all possible conditions, such as under the dark night with bad illumination?
3. Microcontroller powerful enough to store the whole image and process the input stream frame by frame in real time?
4. Having design difficulty to make whole system compact and stream the input video steadily to the microcontroller.

"Wacky" Version:

The stretch goal for the system is that as the system adds a whole frontal vision utility to the vehicle, they can be used not only for lane monitoring and warning, but also for other applications like pedestrian detection or distance estimation with the vehicles in front. Also, the visual system can be composed of more than one cameras, for examples, two cameras to form a stereo vision system, which can better estimate the frontal environment of the vehicle.

These ideas will be evaluated during the planning stage. And they are mostly the algorithm level modification, and the whole system can be composed simply by cameras and a microcontroller.

[1] **M. Nieto**, "Detection and tracking of vanishing points in dynamic environments," **PhD** Thesis, Universidad Politécnica de Madrid, 2010.