I. Executive Summary

Senior Project is the capstone course of the computer engineering program offered by the College of Technology at the University of Houston. In this course, students are treated as employees working for a company in the industry. Students are placed in teams of four and are given a semester-long project to invent a prototype that utilizes all the knowledge and skills they have learned throughout the years in the program. The unique thing about the course is that the teams are not assigned specific projects. Instead, they are given requirements and are expected to come up with their own idea and develop their prototype from the early stages of conception, through to the final presentation.

Team 1 has created a device that can revolutionize the way the medical and emergency service industries respond to emergency situations of motorists. The goal of the project was to integrate a human vital-sign monitor into a vehicle with emergency GPS and data-forwarding capabilities. The Emergency Response Unit is ultimately a device that incorporates a medical sensor to obtain specific vital-sign data, a GPS module to obtain geographical coordinates for physical location, and a mobile phone to transmit information wirelessly through SMS text messaging. The combination of these components provide for a great potential to revolutionize the medical and emergency service industries.

The Emergency Response Unit will communicate with the driver through the use of an LCD screen, displaying specific warnings and helpful information. The Emergency Response Unit utilizes an ear-clip pulse-oximeter sensor that a driver will clip onto their ear lobe which will gather both their pulse rate and oxygen saturation readings. This information is saved to a microcontroller and displayed on a LCD in real-time. The GPS module will constantly track geographical coordinates, and the mobile phone will remain on standby until a state of emergency arises. The device will automatically contact emergency services though the SMS text messaging capability of the mobile phone if it detects a critical situation based on the data collected. The driver will also have the ability to override the automatic warning system and either cancel, or request emergency services at any time. In the event of an emergency, the device will transmit the time, the data collected up until the time of an emergency, and the physical location of the vehicle to an emergency service center.

With the information transmitted, the life saving potential of the Emergency Response Unit proves invaluable. Not only because emergency services will know exactly where the driver is located, but because of the fact that they are able to receive the vital sign information before ever reaching them. Time is critical in every emergency, and the difference between life and death can rely on seconds. The Emergency Response Unit will allow emergency services to have a head start in preparation for necessary procedures which results in saved time in determining how to treat the patient, and ultimately resulting in saved lives.

II. Newsletter

As medicine and technology continue to merge and grow exponentially, there exists a desire and need to bring this advantageous union into the lives of the public. Technology now

provides a means to gather and share information on a scale once thought impossible. Gone are the days that a heart rate is intermittently measured by a nurse and a clock on the wall. Within hospitals, individuals can now be constantly monitored through the use of non-invasive electronic sensors, providing doctors and nurses with real-time vital sign information that can be used to examine trends, prevent emergencies, and efficiently log and store data. So are the times when hospital staff awaited an incoming trauma patient whose condition was not known until their arrival. Communication between emergency paramedics and hospitals has been improved to provide critical information about a patient before they ever reach their destination. In all, the use of technology to gather and share specific information regarding a patient's condition and physiological state has increased efficiency, helped save critical time, and ultimately saved lives.

But there is a piece of this puzzle that is still missing. While information sharing has been vastly improved within and between hospitals and emergency responders, this is not the case between hospitals and citizens. But by taking this one step further and creating a direct communication channel between hospitals and the public, invaluable patient information could be gathered. By providing hospitals and emergency paramedics with specific vital-sign data before they ever reach a person, they would have a better understanding of the trauma, more reliable information, and be better prepared and equipped to handle the situation.

By creating a device that is capable of gathering specific vital-sign information from a person, and in the event of an emergency, transmit this information to a hospital along with their GPS coordinates, the puzzle would be complete. From this, the idea for the Emergency Response Unit was created. In its current form, the device is intended for use within motor vehicles. Using a small sensor placed onto a driver's ear, the device can measure both the heart rate and blood/oxygen saturation. It has the ability to detect potentially unsafe levels from either of the two readings, and either automatically or manually send an emergency request for help. Data gathered before, during, and after an event would also be transmitted, along with their physical location. In this sense, the user of the device would be communicating directly with the hospital and paramedics. The information sent by the Emergency Response Unit could help paramedics determine if the person is still breathing, whether or not their heart is still beating, and provide clues to many other ailments. The data could also be used to analyze the sequence of events that led up to such an emergency and further the understanding and prevention of such conditions. By expanding the use of medical technology outside of the hospital setting, much more can be learned, and many more lives can be saved.

III. Project Report

A. Introduction

The Emergency Response Unit prototype created by Team 1 is a device that has the ability to read human vital signs. More specifically, it can read the pulse-rate and blood oxygen saturation. The device collects vital sign data from a person, stores it, and can detect possible states of emergencies. In the event of an emergency, the time, the vital sign data collected from the sensor, and the physical location of the vehicle will be transmitted to an emergency service center. The Emergency Response Unit employs three main functions. This includes real-time

vital sign monitoring, emergency sequence execution, and optional dialing. Vital sign monitoring consists of measuring heart rate and oxygen saturation of the blood. The emergency sequence consists of retrieving the vital sign data which is stored onto the microcontroller, retrieving the GPS coordinates which are constantly fed to the microcontroller by the GPS module, and packing this information together along with a time stamp into a SMS text message to be received by emergency services. The optional dialing allows for the driver to send all this information elsewhere to another cellular device rather than to emergency services. The optional dialing is a useful function since it gives the user flexibility in logging the information. The users can save it for themselves or forward it to a doctor rather than alerting 911.

The entire project was approved and funded by Dr. Attarzadeh, and the College of Technology. Team 1 consisted of four undergraduates, and the workload was divided amongst team members. Zade Faraj worked to interface the medical sensor, Bradley Hemmi worked with the GPS module, Hoang Tran worked with the mobile phone, and Ali Givmanesh tied all the working parts together to ensure that the information being collected was properly transmitted between the microcontroller and cellular phone, and across the wireless network through SMS. With the completion of the project, Team 1 had designed an innovative system that can revolutionize much of the medical and emergency service industries.

The Emergency Response Unit functions as specified:

- Provides constant monitoring of a person's vital-signs.
- Provides one-touch access to emergency services.
- Decides and automatically contacts emergency services.
- Detects and transmits specific vital-sign data to professional medical personnel.
- Provides even more data; including information, before, during, and after the emergency event.
- Determines and transmits the exact GPS location of the person.
- Is easy to use, and non-intrusive.

B. Background

The idea of the ERU was born as a culmination of other project ideas the group had envisioned at the beginning of the semester. The team wanted to do something related to the medical field because this is an area that is always on the forefront of technology and is perhaps the most necessary in terms of improving the quality of life. One of the initial and perhaps most influential ideas in inspiring the ERU was the Steering Wheel EKG Monitor. Using sensors embedded in the steering wheel of a vehicle, the group set out to monitor a person's EKG readings in hopes to predict a heart attack, or similar problem. After discussing with a cardiologist how one would determine and detect whether a person had suffered a heart attack, the group realized it would take much more than an EKG reading to reach such a conclusion.

Although, there are a few projects that share characteristics with the ERU, such as Harvard's Code Blue, in terms of functionality, they differ completely. Briefly put, Code Blue is meant for use in a disaster where medical personnel have already responded, and need to monitor the vital signs of various patients. However, the ERU is a device that is constantly monitoring

the person, and in the situation of a medical emergency, first responders will be contacted automatically. Furthermore, Code Blue is a perfect example of how medical sensor networks can benefit a place such as a hospital or assisted living center, but this is only after the system has been setup to properly function in its' intended environment. Unlike Code Blue, the ERU is a simple device in terms of how the user interacts with it, and would take very little training to use. To have a device with the ability to save seconds, in a situation where seconds can be the difference between life and death is really all there needs to be said concerning the necessity of the ERU.

As stated earlier, the medical field is one of the industries that is always on the cutting edge of technology, and for good reason, it saves lives. The ERU comes at a time when the technology needed to make a convenient and compact device is readily available. For instance, countries all around the world have integrated citywide wireless networks that provide, in most cases, free access for it's' citizens. Improvements in wireless technology have increased the distance these networks cover, and over time, these networks will be available to every inhabited place around the world. The idea of a person having their vital signs constantly monitored and in communication with emergency personnel at all times is very feasible. Combining this constant monitoring with a GPS (Global Positioning System) greatly increases the chance a person will be found in the event they have an emergency out of plain sight. When testing the GPS used in the ERU, the group found that it was accurate to within about five meters or so. This distance is acceptable, but there are systems accurate to within two meters on the market today. Due to cost, the group was limited on the amount of money they were able to spend on the GPS unit. With wireless technology constantly improving and GPS devices becoming commonplace in most everyone's lives, a device such as the ERU makes perfect sense.

C. Product Requirements

The ERU will provide the user with multiple functions and features that are very simple to operate. Below is the list of all the functions that the ERU will carry out, along with the features accessible to the user.

ERU Functions:

- Constant monitoring of vital signs
    - Acquire the heart rate
    - Acquire the oxygen saturation of the blood
- Log the acquired data
- Organize the vital sign data
- Display the heart rate and oxygen saturation on an LCD in real time
- Acquire GPS information
    - Time
    - North coordinates
    - West coordinates
- Transmit an emergency text message via a mobile phone with the following information:

    - Time of the emergency
    - Location of the user
    - Organized, collected data from the sensor
- Provide an optional dialing choice for the user
    - Choose a specific number to send the text message to.
- Retrieve and perform functions based on user input

ERU Features:

- Automatic decision-making to warn emergency services
    - In case of no response from the user
- Warning system displayed on LCD:
    - Manual override of automatic-decision query
    - Out of threshold bounds
    - Sensor not connected / functioning correctly
    - Unintelligible data
    - Sending data / GPS to emergency services
- Manual overriding capability to:
    - Transmit an emergency signal at any time
    - Cancel the automated signal
    - Change the phone number



Figure 1: Emergency Response Unit

A picture of the finished product is shown in Figure 1. The entire project is enclosed within a box which provides the user with a connection for the cell phone and sensor.

The code for this project was written in C language. The software required to write and debug this code is Micro-IDE. After compiling and building the code, it can be downloaded onto the MINI-MAX 8051 located inside the project box.

A list of the components that were used in the project is provided below. The total cost of the project added up to $575.00, including the shipping of parts and any applicable local taxes. The allotted budget set forth by the senior project class totals $350.00. The teams total project cost is more than the budget provided, extending an extra $255.00.

- MINI-MAX/51-C2
- Nonin OEM III Module
- Nonin Ear Clip Sensor
- USGlobalSat EM-406a GPS Module
- GSM Mobile Phone
- Bipom LCD
- Bipom Keypad

D. Design Alternatives

The greatest challenge encountered with the design of the ERU came about when the group had to decide how each piece of hardware would be interfaced with the microcontroller. The three main modules: the GPS, the Nonin OEM III, and the Sony Ericsson T610 are all serial devices. Because the microcontroller has only one RS232 serial port, the group purchased a UART-1 expansion board from Bipom that contained two additional RS232 serial ports. Although this expansion board essentially doubled the physical size of the microcontroller, it also provided the system with enough serial ports to interface all three devices simultaneously.

After spending some time interfacing the devices with the UART-1, the group came to the conclusion that three serial ports were in fact not necessary to achieve the results needed. By using the general purpose pins located on the 8051, the group could interface both the cell phone and GPS unit directly to the microcontroller. In total, four of the general purpose pins were used: Rx, Tx, Vcc, and GND. By using these pins, the group was able to remove the UART-1 expansion board from the project which decreased the size of the ERU, reduced the price, and allowed for a more simple design. A reduction in size and price needs no explanation, but the idea of a more simple design can be elaborated.

Because the phone and GPS are both serial devices they needed to be converted to TTL to properly communicate with the microcontroller. Had the group continued to use the UART-1, the phone and GPS would have to have had to go through an extra voltage level conversion explained in the following sequence: TTL to RS232 to TTL. Without the UART-1, voltage level conversion was no longer necessary and the devices could be connected directly to the microcontroller. The more conversions the data has to go through, the higher the chance of data becoming corrupted along the way. Also, debugging and testing the devices using a single conversion as opposed to two made it much easier and less time consuming.

Another part of the design process the group had to make a decision on was the type of mobile phone to use. After researching other projects that used phones interfaced with a microcontroller, the group quickly realized it would not be a simple task. Different manufacturers use particular programming languages when communicating to a mobile phone. The group discovered the popular choice for these types of projects were Sony Ericsson branded

phones because the company readily provides technical manuals for all of their devices. After obtaining a Sony Ericsson T610 phone the group downloaded the White Papers, also known as the developer's manual. The White Papers contained a library of all the AT commands, which are used to communicate with and program the phone. The AT commands are extremely intuitive and easily integrate with the C programming language which was used to develop the software for the ERU. The group was able to successfully communicate with the phone in a matter of minutes, which allowed a lot of time to be used for other portions of the project.

The final design phase involved the physical construction of the prototype. The group wanted the ERU to look organized, and have a distinctive, professional look. A plastic project box was purchased from RadioShack to contain all the hardware, aside from the phone and ear clip sensor; these two pieces connect via two serial ports cut and fitted for easy access from outside the box. Holes were also cut to provide a clean, tight fit for the external LCD and keypad which are available for the user to interact with the ERU. Inside, the remaining hardware was positioned in a logical manner to allow easy removal in the event a defective part may need to be replaced. Overall, the ERU is a very professional piece of hardware that is proudly displayed by its creators, Team 1.

E. Design Specifications

Several individual modular blocks were used to design this project. The combination and integration of these blocks can be used to create the device. The overall hardware block diagram of the ERU can be seen in Figure 2.

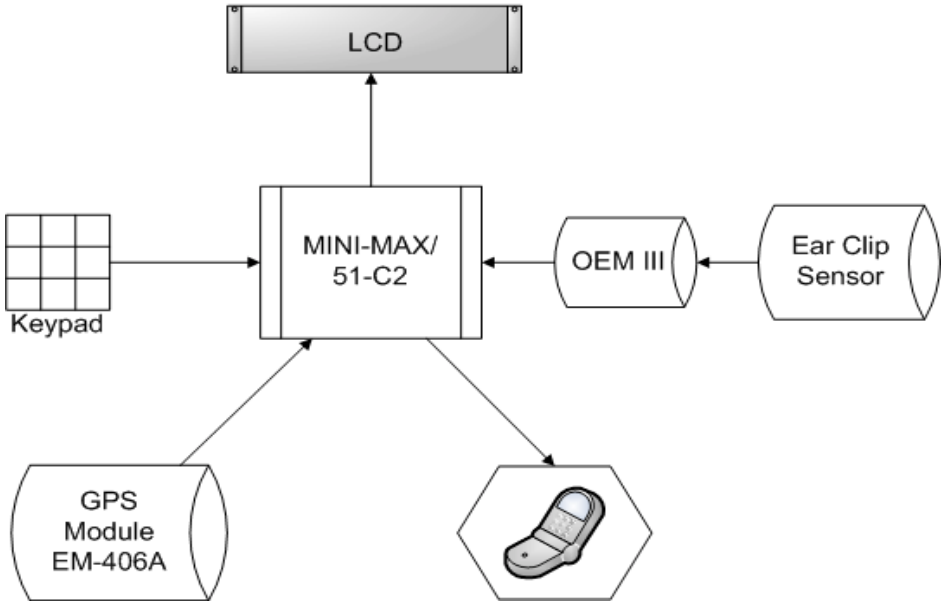


Figure 2: Block Diagram

As shown in Figure 2, the microcontroller is the heart and brain of the ERU, and everything except the ear clip sensor is connected directly to it.

- The LCD and keypad have their own special pins on the microcontroller that are easily fitted.
- The GPS module is connected to the general purpose pins (J4) of the microcontroller to transmit information. It specifically uses pin 18 (P3.2) on the J4 port of the microcontroller. The data transmission occurs at a baud rate of 4800.
- The mobile phone also uses the J4 port of the microcontroller, using pin 14 (P3.6) in order to receive the AT commands needed to send a text message. The software is set to transmit commands to the cell phone at a baud rate of 4800.
- The ear clip sensor is connected to the OEM III prototype board through Nonin's proprietary cable. The OEM III board is then connected to the microcontroller through an RS-232 connection which allows a serial transmission of data at 9600 bps.

The complete flow chart which details the software routine for ERU is shown in Figure 3 on the following page. There are three main functions of the device identified in the flowchart that are listed below and will be discussed individually in more detail.

1. Vital Sign Monitoring Function
2. Emergency Sequence Function
3. Optional Dialing Function

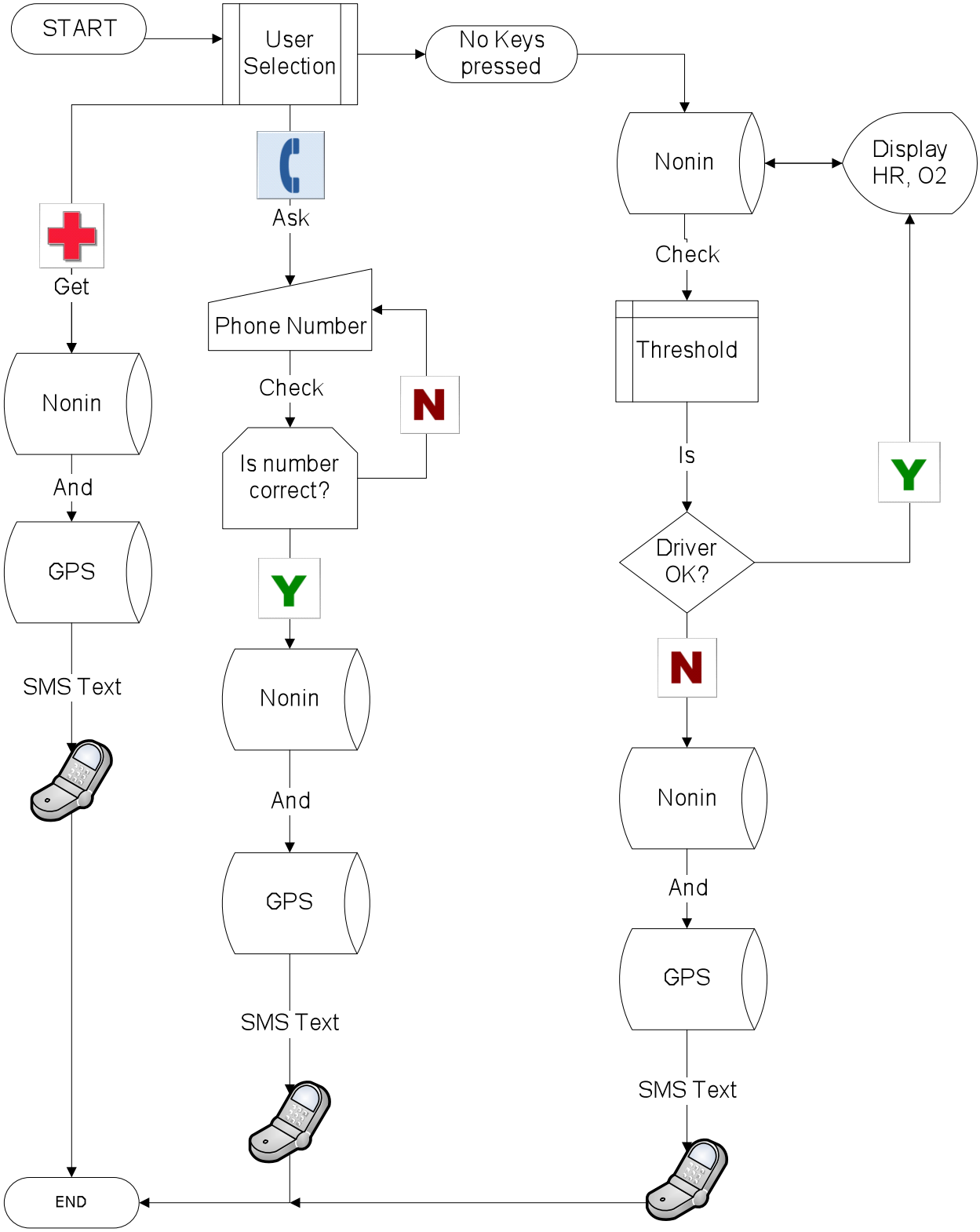


Figure 3: Main Flowchart

1. Vital Sign Monitoring Function

The Team used C language as well as Micro-IDE in order to write the software code expressed in the flowcharts. The first function which provides constant monitoring is shown in Figure 4.

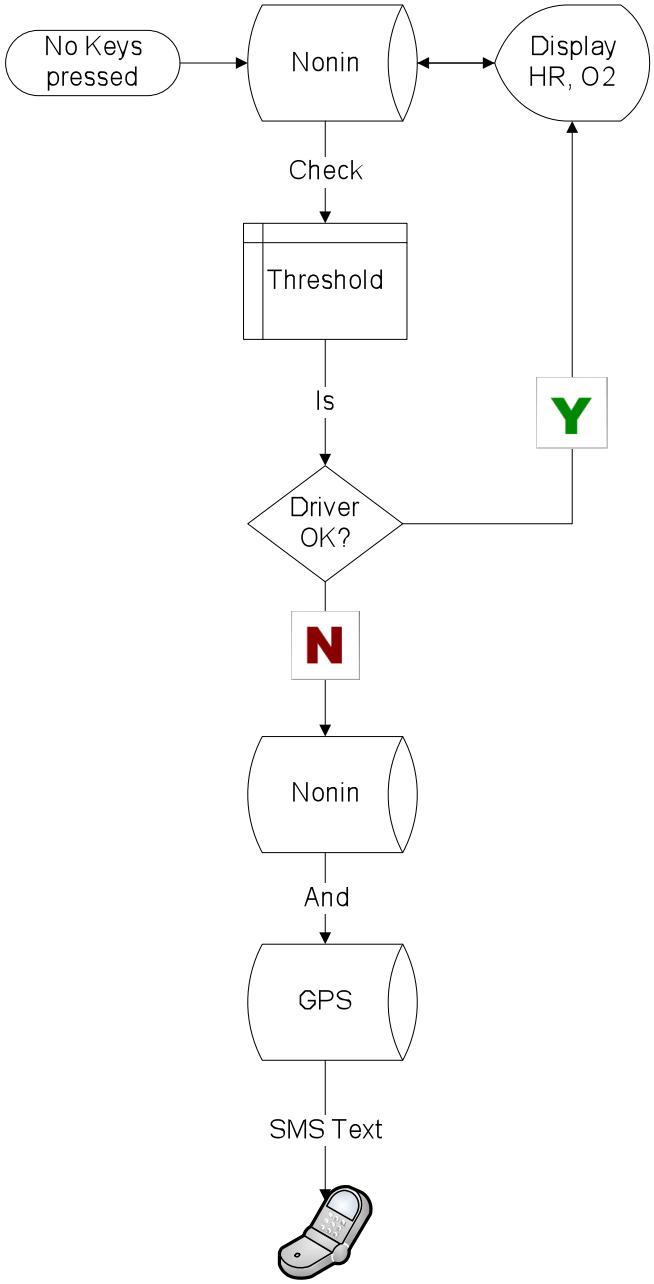


Figure 4: Vital Sign Monitoring Function

If no key is pressed by the user, the program will automatically go through the monitoring sequence in which it will get the data from the sensor and display the heart rate and oxygen saturation on the LCD. It will also check to make sure the data is within the set thresholds. If the data is within the threshold, it will repeat the same function over again. In case of an emergency where one of the vital signs goes out of the acceptable range, the program will prompt the user with this message: "Your HR is too high, are you OK?". If the user responds

with “yes”, the program will return to gathering more data, while temporarily ignoring any sequential emergency situations detected. The user may choose the “no” option, or in the case of no response, the emergency sequence function will be executed. Below is the pseudo code used to provide the vital sign monitoring function.

```
while(1)
        {
                check=User_Override();
                if(check==1)
                {
                        Emergency_Sequence();
                }

                if(check==2)
                {
                        Send_Data();
                }

                Nonin_Get_Data();
                check=Nonin_Check_Data();

                if(check==0)
                {
                        Nonin_Store_Data();
                        Print_Data();
                        check=Nonin_Check_Emergency();
                        if(check==1)
                        {
                                check=User_Emergency_Option();
                                if(check==1)
                                {
                                        Emergency_Sequence();
                                }
                        }
                }
        }
```

Because the Nonin OEM III module uses the serial port provided by the microcontroller, the team was able to use a simple getchr( ) library function provided by Bipom, and the SCON register to receive the three bytes of data emitted from the OEM III. The function is shown below:

```
while(i!=1)
        {

                if((SCON&&0x01)==0x01) // Wait for RI flag
                {
                        SCON=(SCON&0xFE);

                        temp = getchr(); //get a char serially

                        if((temp&&0x80)==0x80)
                        {
                                arraytemp[0]=temp;
```

```
                        if((SCON&&0x01)==0x01) // Wait for RI flag
                        {
                                SCON=(SCON&0xFE);

                                arraytemp[1] = getchr();       }

                        if((SCON&&0x01)==0x01) // Wait for RI flag
                        {
                                SCON=(SCON&0xFE);

                                arraytemp[2] = getchr();       }
                        i=1;
                }
        }
}
```

2. Emergency Sequence Function

Figure 5 displays the logical steps for the emergency sequence function. This function is reusable and can be called in three different parts of the program.

1. If the user is in critical condition and unable to respond.
2. If the user manually seeks emergency help.
3. If the emergency is detected by ERU and confirmed by the user.

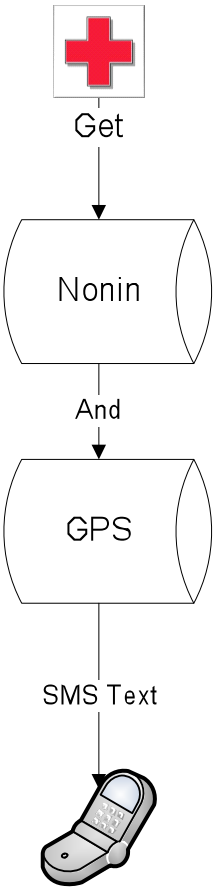


Figure 5: Emergency Sequence Function

If for any reason the program enters this function, five standard steps are taken before returning to original position in the code.

1. Get the data acquired by the sensor
2. Convert the BCD data into ASCII characters
3. Get the data from the GPS module
4. Format the data to show time as well as north and west coordinates
5. Transmit the right AT commands in order the send a text message with all the gathered data

The function that converts the BCD data into ASCII characters is shown below.

```
void Nonin_Convert_Binary_To_ASCII (int Binary,int index)
{

    unsigned char one, two, three;
    int BCD;
    int i,j;
    BCD = 0x00;

    one=0x30;
```

```
    if (Binary >= 0xC8)
    {
        Binary = (Binary - 0xC8);
        one = 0x32;
    }

    if (Binary >= 0x64)
    {
        Binary = (Binary - 0x64);
        one = 0x31;
    }

    for ( i=0 ; i<8; i++ )
    {

        if ((Binary & 0x80) == 0x80)
        {
            BCD = BCD + 0x01;
        }

        if (i<7)
        {
            if (((BCD & 0x05) == 0x05) | ((BCD & 0x06) == 0x06) | ((BCD &
0x08) == 0x08))
            {
                BCD = BCD + 0x03;
            }
            BCD = (BCD << 1);
        }

        Binary = (Binary << 1);
    }

    two = BCD;
    two = (two >> 4);
    two = (two + 0x30);

    three = BCD;
    three = (three & 0x0F);
    three = (three + 0x30);
}
```

3. Optional Dialing Function

Figure 6 displays the optional dialing function which allows the user to change the phone number. Sometimes drivers may feel that a particular situation is not critical or they don't need immediate help. In that case with this function they may send the data to their doctor or to a phone number of their choosing.

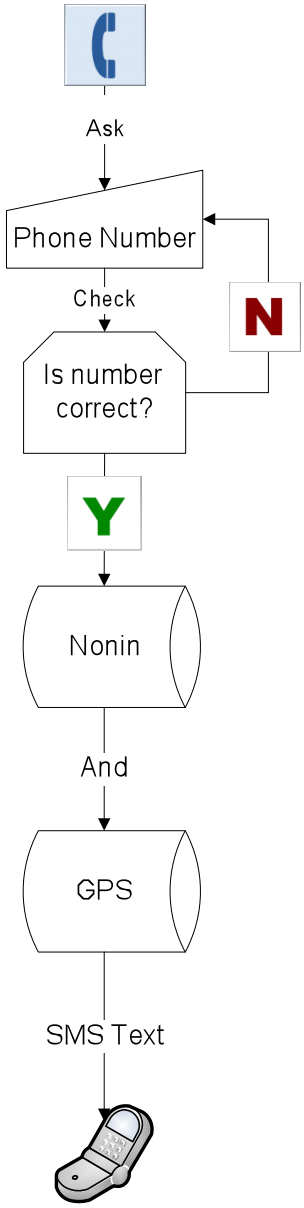


Figure 6: Optional Dialing Function

As it shown in Figure 6, there is an error checking sequence that is executed before the program sends the data. The code for that particular part of the program is provided below. The entire source code of the project is provided in the Appendix of this report.

```
int Phone_Check()
{
        int i;
        char key;
        LCD_WriteCtrl(0x01);
        LCD_WriteCtrl(0x02);
        delay(50);
```

```
        LCD_printf("Phone #: ");
    for(i=1;i<11;i++)
    {
        LCD_printf("%c",data4[i]);

        if(i==3)
        {
                LCD_printf("-");
        }
        if(i==6)
        {
                LCD_printf("-");
        }
    }

        LCD_WriteCtrl(0xC0);
        LCD_printf("Is this correct: (Y/N)");
        while(1)
        {
                key = ScanKeypad();
                if(key)
                {
                        if (key == 'A')
                        {
                                return 1;
                        }
                        return 0;
                }
        }
}
```

F. Design Description

As is shown in the block diagram in Figure 7, the ERU consists of modular blocks. In this section, functions that are provided by each component will be discussed in more detail.

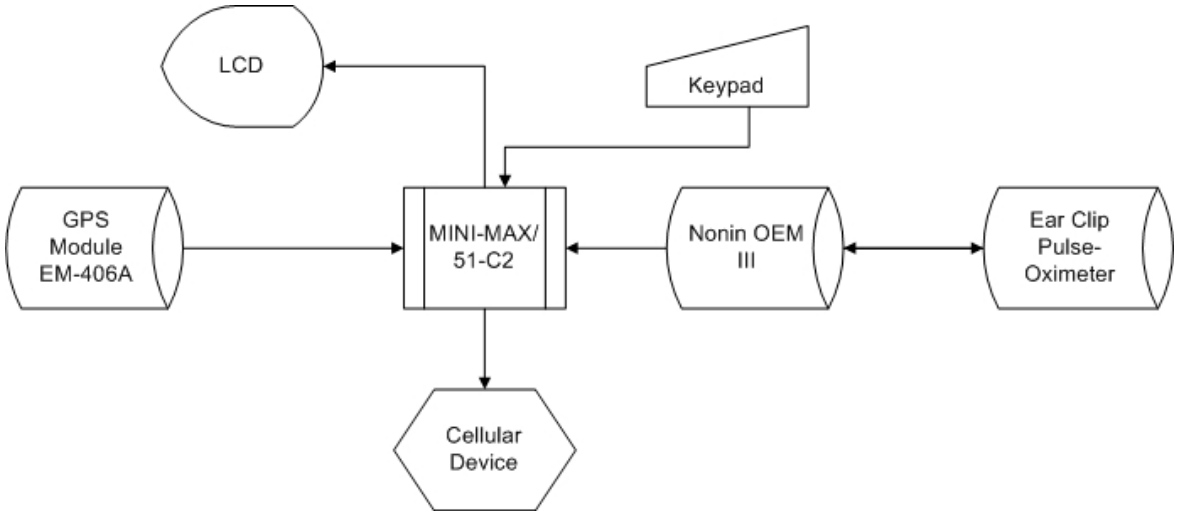


Figure 7: Block Diagram

1. MINI-MAX (51-C2)

The MINI-MAX microcontroller is the heart and brain of the project. It is an 8051 compatible single board computer based on the Atmel 89S family. It is used to interface between every other individual device in the ERU (Emergency Response Unit), and make critical decisions based upon the data it collects.

The microcontroller receives data from the Nonin OEM III module, the GPS module, and the keypad. The collected data is temporarily stored within the microcontroller's main memory and retrieved during an emergency situation. During the operation of the system, the microcontroller's software constantly checks the data to determine what functions, described in design specification section, need to be performed. The microcontroller is also used to transmit data to the LCD to be displayed and to the cellular device for transmission.

2. Ear Clip Pulse-Oximeter and Nonin OEM III module

The Ear Clip Pulse-Oximeter is a medical sensor that indirectly measures both the heart-rate and the oxygen saturation of a person's blood. Using a small clip, the sensor is placed over a patient's earlobe and is directly connected to the Nonin OEM III module.

The pulse-oximeter sensor is crucial to the ERU's operation, providing the critical data used to determine a person's physiological state. The sensor has the extra ability to provide not only the blood/oxygen saturation, but also the heart-rate. This extra piece of information is very important in diagnosing any patient, and is a great feature of the device that comes at no extra cost.

The ear clip sensor is interfaced directly with the Nonin OEM III module. Because it is a direct plug and play system, the device does not have to be configured or calibrated to function correctly. The sensor communicates directly with the OEM III module, both transmitting and receiving datum.

Because of the limitations imposed on the project due to time, cost, and application, the sensor chosen for this project could not be wireless. Given a larger budget and more time, the team would have had the ability to interface a wireless Bluetooth pulse oximeter to a separate wireless module. Also, because the ERU is intended to be used within a vehicle, the sensor the group chose had to be practical. Using the earlobe as the site of data collection was determined to be the least invasive and most convenient method for drivers.

The Nonin OEM III module provides a simple way for the team to perform the pulse-oximetry task of this project. It is a compact device that is power efficient yet robust. Its task is to interpret the information of the pulse-oximeter sensor, condition the readings, and provide a simple serial output of the binary collected data.

The module itself provides many different features that were essential to the team's software and the overall functioning of the ERU. It also allowed the team to gauge when specific warnings needed to be provided to the driver. The OEM III provides three different serial outputting format options; each of which contains the heart-rate and SpO2 (oxygen saturation of the blood). Where the formats differ is in the framing of the outputted packets, and the specific data relative to the operation of the sensor and the accuracy of the data. The OEM III is connected to the RS-232 port of the microcontroller and it transmits the data serially to the microcontroller.

3. GPS Module EM-406a

The GPS module used in this project is known as the EM-406a, manufactured by USGlobalSat Incorporated. To further the functionality of the device, the GPS unit was used to supply the exact coordinates of the patient. Therefore during an emergency, the location of the patient along with their vital-signs can be transmitted directly to paramedics. Because the ERU will be used in a vehicle, having the GPS feature incorporated into the ERU makes it much more practical.

Similarly to the Nonin OEM III module, the EM-406a is interfaced to the microcontroller using a single serial port line. The device outputs a binary stream of data that are collected over the ports of the microcontroller and saved to its memory. In order to perform simple testing and analysis of the EM-406a, a separate device known as a RS-232 shifter was used to interface it with a PC. This was needed to shift the voltage levels of the GPS module so that they were suitable for reading by the computer. Because the GPS module is quite small and compact, it does not contain a larger RS-232 port. This forced the team to make an additional purchase of the shifter to carry out simple testing.

4. Cellular Device

A cellular device is used in this project to transmit information collected by the sensor and GPS unit to an emergency service center. The data is forwarded to the phone from the microcontroller and sent using SMS (Short Message Service). The device serves as the main method of communicating vital-sign information and geographic coordinates to the outside world.

The use of the cellular device, like most other parts in this ERU, is critical to the functionality of the device. Without the communications segment, the ERU would be rendered devoid of its life-saving capabilities.

The mobile phone is interfaced to the microcontroller so that the data stored in the memory of the microcontroller may be passed to the phone for transmission. The cellular device is connected to pin 14 (P3.6) on port J4 of the microcontrollers general purpose pins.

5. Keypad and LCD

The keypad used in this project is the BiPOM matrix keypad. Four buttons are utilized from the keypad; one button to override/cancel any automatic decision made by the microcontroller, a second button to immediately contact medical services, and the third and fourth reserved for "Yes" and "No" answers. The keypad will be within reach of the user at all times and will help them interact with the system.

The minimalist approach of the keypad is indicative of how simple and straightforward the ERU is. The input generated by the user is interpreted by the software encoded in the microcontroller, which helps determine whether or not a critical emergency has occurred.

The keypad manufactured by BiPOM is easily integrated and interfaced with the MINI-MAX microcontroller using the supporting libraries provided. BiPOM also supplies an LCD that was easily integrated for use with the MINI-MAX microcontroller. The LCD in this project is used to convey specific vital-sign information to the driver and also display any warnings that may require attention.

G. Construction Details

The construction of the ERU is divided into separate hardware modules which will be presented and explained.

Hardware necessary to complete the construction of the prototype is as follows:

1. Bipom MINI-MAX/51-C2
2. Bipom LCD and Keypad
3. Nonin OEM III Development Kit
    - OEM III Module
    - Prototype board
    - Ear-clip sensor
4. USGlobalSat EM-406a
5. Sony Ericsson T610 mobile phone
6. Sony Ericsson DRS-11 serial data cable
7. RS-232 Shifter SMD
8. 2x RS-232 Gender Converters

Hardware modules include interfacing the following:

1. Nonin OEM III module to Microcontroller
2. USGlobalSat EM-406a GPS module to Microcontroller
3. GSM mobile phone to Microcontroller
4. LCD and Keypad to Microcontroller

1. <u>Nonin OEM III module to Microcontroller</u>

In order to retrieve vital sign data from the individual wearing the sensor, the Nonin OEM III module must be interfaced with the microcontroller. This allows serial data sent from the OEM III module to be collected and stored on the microcontroller for further processing.

In order to connect to the microcontroller, the Nonin OEM III module must be attached to the Nonin prototype board. The prototype board provides an RS-232 serial port which can be directly connected to the microcontroller through the use of a single RS-232 gender converter. By connecting the two in this fashion, a connection cable is not necessary and the two boards will be joined as seen in Figure 8

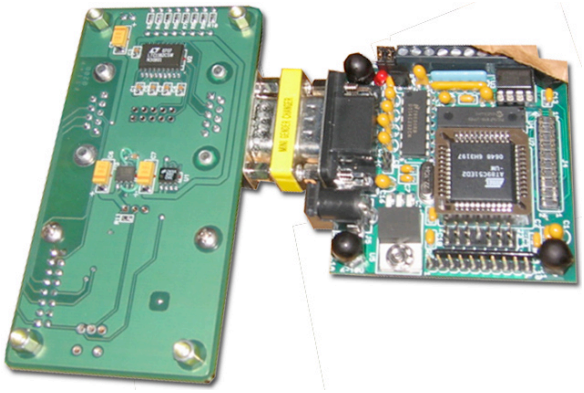


Figure 8: Nonin Prototype Board to Microcontroller

The Nonin OEM III module provides data output in three different formats as noted in the data sheet in the Appendix. The data format used in this project is "Serial Data Format #1". In order to produce this output, physical selection pins on the prototype board must be set in an exact position. To achieve this, the left-most pin must be placed in the on position, while all others are switched off.

In order to acquire data, the ear-clip sensor is connected to the remaining serial port on the Nonin prototype board. A 9V battery is supplied with Nonin OEM III Development Kit that will be used to supply power to the device.

2. <u>USGlobalSat EM-406a GPS module to Microcontroller</u>

In order to obtain the GPS coordinates of a patient using the device, an EM-406 GPS module from USGlobalSat is used to provide this information to the microcontroller. The EM-406a is a TTL device, operating at 5V, and providing serial data between 0 and 5V. Because the microcontroller also operates at TTL logic levels, the two devices can be interfaced directly. Also, the microcontroller provides a power source of 5V which can be used to directly power the EM-406a.



Figure 9: EM-406a GPS Module

As seen in Figure 9, the EM-406a comes with a very small interface cable that is neither suitable nor large enough to fit on the microcontroller. It is recommended that the cables are either spliced and attached to separate wires, or wires are inserted into the pins in order to elongate the interface cable.

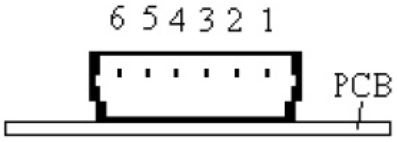


Figure 10: EM-406a output connections

Connections between the EM-406a and Microcontroller are provided in Table 1.

Table 1: EM-406a to Microcontroller connections

	EM-406a pins (Figure 10)	**Microcontroller J4 pins**
Vcc (+5V)	2	1
GND	1,5	3
Rx		18 (3.2)
Tx	4	

3. GSM mobile phone to Microcontroller

To simulate transmission between the device and an emergency response center, a Sony Ericsson T610 mobile phone is used. The mobile phone must be connected to the microcontroller so that data collected from the sensors can be formatted and passed to the phone for transmission. Software will be used to control the phone and implement the transmission sequence.

The mobile phone is connected to the microcontroller through the use of a Sony Ericsson DRS-11 data cable and RS-232 logic level shifter. The DRS-11 serial cable connects directly to the mobile phone. Because the DRS-11 cable contains circuitry that automatically converts the phones 2.7V output to RS-232 +-12V, the RS-232 voltage levels must be shifted back down to 5V in order to connect to the microcontroller. The RS-232 Shifter SMD circuit is used to perform just this.



Figure 11: Sony Ericsson T610 Phone



Figure 12: Sony Ericsson DRS-11 Cable



Figure 13: RS-232 SMD Shifter

The Sony Ericsson phone pictured in Figure 11 is connected to the DRS-11 data cable in Figure 12. The DRS-11 data cable is then connected directly to the RS-232 SMD shifter pictured in Figure 13. through the use of a RS-232 gender converter. Because the Shifter does not come manufactured with wires to interface to the microcontroller, they must be soldered onto the board itself. The shifter is the connected to the microcontroller.

Connections between the RS-232 Shifter and Microcontroller are provided in Table 2.

Table 2: RS-232 Shifter SMD to Microcontroller connections	
RS-232 Shifter	**Microcontroller J4 pins**
Vcc (+5V)	1
GND	3
Tx	
Rx	14 (3.6)

4. LCD and Keypad to Microcontroller

To display information to the user and allow them to interact with the device, a Bipom LCD and keypad are used. The LCD and keypad are connected to the ports labeled J1 and J3 on the MINI-MAX/51-C2 microcontroller.

H. Costs

The cost of the ERU exceeded the allotted budget of $350.00. The medical sensor used in the project alone cost $308.00; this coupled with the microcontroller already put the team over the budget.

Table 3: Project Cost		
Item	**Est. Cost**	**Actual Cost**
MINI-MAX/51-C2	$69.00	$69.00
LCD242	$24.00	$24.00
Keypad (KP1- 4x4)	$24.00	$24.00
Nonin OEM III Ear Clip Pulse-Oximeter	$300.00	$308.00
Mobile Phone	$20.0	$35.0
GPS Module - EM-406A	$80.00	$80.00
RS232 Shifter SMD Electronic Components	$20.00	$35.0
~~UART-1~~	~~$ 49.00~~	~~$49.00~~
Totals	$537.00	$575.00

The group used the formula: **Total Salary = Salary/hr * 2.5 * hours worked**, to calculate an estimated cost of labor.

Table 4: Labor Cost

Team Member	Total Project Est.	Total Project Act.	Calculated Pay
Zade Faraj	142	165	$8250.00
Ali Givmanesh	145	163	$8150.00
Bradley Hemmi	130	150	$7500.00
Hoang Tran	130	150	$7500.00
Totals	**453**	**628**	**$31,400.00**

A number of tools used by the students were provided by the lab. Table 5 is an estimated cost of these tools.

Table 5: Lab Equipment

Item	Cost
Computer	$800.00
DC Power Supply	$550.00
Digital Multi-meter	$300.00
Miscellaneous Tools	$200.00
Shop Service	$200.00
Totals	**$2,050.00**

With a salary of $20.00/hr the estimated total cost of this project comes in at $34,025.00.

Table 6: Grand Total

Category	Cost
Project	$575.00
Labor	$31,400.00
Lab Equipment	$2050.00
Totals	**$34,025.00**

I. Conclusions

Given the allotted time during the senior project course, the team was able to successfully build and satisfy the goals of the Emergency Response Unit. Upon completion, the device was able to gather human vital sign data, store and log the information, and transmit the data via a mobile phone along with GPS coordinates during an emergency situation. The team also met all the objectives set forth in the proposal, adding additional functionality during the build process.

The outcome of the project proves that such a device is not only possible, but also feasible. The idea that a person's physiological state can be monitored, and at a moments notice, their physical location and vital sign information can be relayed to paramedics is completely possible given today's technology. The team feels that the idea of integrating a device of this sort into the lives of the general public is inevitable. Whether this idea is integrated today or tomorrow, it will happen, and the Emergency Response Unit has confirmed that.

Although the concept of the device was realized with the successful build of the Emergency Response Unit, the team would like to further the functionality and more closely align the project to the ideal vision, whereby the device would be much smaller, more functional and more marketable overall. The team agrees that there will always be a demand for personal transportation, and as such, a device of this sort will certainly find use. Given an increased budget and more time to work on the project, the team would have focused more on the hardware portion of the device.

Currently, the E.R.U. uses a wired ear-clip sensor that connects the Nonin OEM III module before being passed on to the microcontroller for processing. There exists wireless sensors based on Bluetooth technology that would allow the wired sensor to be replaced, allowing greater flexibility for the user and a providing a less intrusive means of data acquisition. Also, the Nonin OEM III module used in the project is currently attached to a much larger Nonin prototype board that is connected to the microcontroller. The OEM III module is completely capable of functioning without the use of the prototype board, and at $1/10^{th}$ the size it would greatly minimize the overall. There are many more aspects of the project that could be refined and reconfigured to create a device that is more practical and functional. In all, the team set forth to produce a device that achieved a set of objectives, and the Emergency Response Unit is the product of those objectives.

IV. User Instructions

A. Getting Started

1. Connect medical sensor to the port labeled, “sensor”.
2. Connect phone using the data cable provided to the port labeled, “phone”.
3. Power on the mobile phone.
4. Lift top off of ERU and carefully set it aside so as not to disconnect any cables.
5. Attach a 9V battery to the 9V battery connector.
6. Attach four AA batteries to the AA battery pack.

B. Operating the ERU

1. Attach ear clip sensor to the ear lobe; be sure to have the clip set fully on the lobe to ensure full contact between flesh and sensor.
    - If sensor is placed properly on ear lobe, the LCD should begin displaying heart rate and blood oxygen saturation information.
    - If the LCD displays the message, “Sensor Disconnected”, wipe the ear lobe and the ear clip on the side that makes contact with the skin, and then reattach to lobe.
2. To contact emergency services, press the button with the red cross icon.
    - LCD will display, “Message Sent”, upon successful transmission.
3. To send vital sign information and GPS coordinates to a different phone number, press the button with the blue telephone icon.
    - This will prompt for a phone number to be entered.
    - Using the keypad, enter the 10 digit phone number starting with the area code.
    - Confirm that the number is correct by pressing “Y” for yes or “N” for no.
    - LCD will display, “Message Sent”, upon successful transmission.

C. In Event of Emergency

1. LCD will ask, “Are you okay?”
    - Respond by using keypad and press “Y” for yes or “N” for no.
        - Choosing “Y” will avoid contacting emergency services and continue to monitor vital signs for the time being.
        - Choosing “N” will immediately contact emergency services.

- Pressing the red cross icon at any moment will automatically contact emergency services.

Project Schedule

The group tossed around various ideas during the first couple of weeks of the semester, and later used different aspects of those ideas to create the ERU. The following few weeks the group began researching information involved with the technologies used in the ERU. The group had to find out what hardware would be needed to make the ERU, and also, what would be needed to get these different pieces of hardware to work together?

After the hardware was purchased and received, the group began testing the communication with the individual devices using basic programming. Once the individual modules were working properly, the group used the following three weeks to began interfacing them to the microcontroller. Afterwards, a weeklong testing period ensured that the ERU was functioning properly, and allowed the group to make any necessary changes in the software.

The last week the group used a customized project box to store the ERU. All aspects of the project were completed by May 2, 2008.

References

[1] Yuan, Jane. ISGRIN Research Projects. Engineering Tech. Dept. University of Houston. 20 Feb. 2008.10 Feb. 2008 <http://www.tech.uh.edu/College/Labs/isgrin/ResearchProjects.htm>.

[2] "The World Health Organization's ranking of the world's health systems". Geographic.org. 29 Feb 2007. 09 Feb. 2008 <http://www.photius.com/rankings/healthranks.html>.

[3] "Top 10 Causes of Death among Adults over Age 65". Senior Health. About.com. 21 Jun. 2007. 09 Feb. 2008 <http://seniorhealth.about.com/od/deathanddying/tp/cause_death.htm>.

[4] Hitti, Miranda. "Heart Disease Kills Every 34 Seconds in U.S.". Fox News. WebMD. 27 Dec. 2004. 15 Feb. 2008 <http://www.foxnews.com/story/0,2933,142436,00.html>.

[5] "Code Blue: Wireless Sensor Networks for Medical Care" School of Engineering and Applied Sciences. 26 Sep. 2006. Harvard University. 13 Feb. 2008

<http://www.eecs.harvard.edu/~mdw/proj/codeblue/>.

[6] lifealert.com. Medical Protection. 13 Feb. 2008 < http://www.lifealert.com >.

[7] "OnStar Explained". onstar.com. GM. 13 Feb. 2008
<http://www.onstar.com/us_english/jsp/explore/index.jsp>.

[8] Corenman, James, and William New, Jr. "Pulse oximeter monitor" US Patent 4653498. 12 Feb. 2007.