October 27, 2003

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

Re: Design Specification for an Emergency Response System

Dear Dr. Rawicz

The attached document, *Design Specification for an Emergency Response System*, lists the design specifications for our ENSC 340 project. We are developing a device that will help people with medical conditions obtain assistance under emergency situations.

The purpose of this design specification is to detail the design and testing that our completed *Intelli-Alert*™ will fulfill. The document lists the relevant design information needed for successful project completion in December. Some sections are fairly detailed since group members will be using this document as a reference for design.

GiveLife Systems was formed in September of 2003 by four motivated, innovative, and talented engineering students: Tristen Georgiou, Yang Pan, Hashina Parveen, and Melody Guo. If you have any questions or concerns about our functional specification, please feel free to contact me by phone at (604) 771- 6089 or by e-mail at givelife-systems@sfu.ca.

Sincerely,

Y Pan

Yang Pan
President and CEO
GiveLife Systems, Inc

Enclosure: *Design Specification for an Emergency Response System*

GIVELIFE
Systems

Design Specification for an

Emergency Response System

Project Team: Tristen Georgiou
Yang Pan
Hashina Parveen
Melody Guo

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

Issued date: October 27, 2003

Revision: 1.0

EXECUTIVE SUMMARY

The rapidly growing senior population in Canada suggests that the demand for products designed for senior citizens is likely to be high in the near future. GiveLife Systems' (GLS) goal is to produce *Intelli-Alert* ™, an emergency response device that can also assist the elderly to monitor their heart rate.

Intelli-Alert ™ consists of pulse sensor, RF sub-system and the base station. The pulse sensor detects pulse in the wrist as blood flows beneath the sensor. The RF sub-system, which consists of a RF transmitter, a transmitting antenna, a receiving antenna and a RF receiver, will be responsible for communicating the signal between the pulse sensor and base station. When the microcontroller detects something goes wrong or when the user pushes the buttons on the wristband, the base station will be activated to call the default number stored in its memory.

This document provides an overview of our system and details the design specifications of the system hardware and software. A test plan is also included.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

GiveLife Systems (GLS) will use off-the-shelf sensors, microcontrollers, and wireless technology to develop *Intelli-Alert* ™. The system improves upon existing technology by monitoring and acting upon if necessary, pulse rates of the user. As well, use of the device will not require a subscription service, or central monitoring station. The calls will go direct to the source, saving valuable time, and possibly lives.

1.1 Scope

This document describes the design specifications that must be met by the *Intelli-Alert*™ system. It explains how the product will be designed to meet all the functional requirements as described in the Functional Specification. Note that these design specifications only apply to the proof of concept device. These specifications may change slightly as the design process moves forward to final completion.

1.2 Intended Audience

This document is to be used by all members of the GLS team. The engineers will use it as a manual for product completion. Project leaders will use it to assess project progress. Marketing personnel will use this document to develop initial promotional material. Finally, patent lawyers will use it to identify any intellectual property that should and can be protected.

1.3 Acronyms

ADC	Analog-to-Digital Converter
API	Application Programming Interface
DMM	Digital Multi-Meter
GLS	GiveLife Systems Inc.
GUI	Graphical User Interface
IR	Infrared
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MDB	Linx Master Development Board
MEK	Linx Master Evaluation Kit
PIC	Programmable Integrate Circuit
RF	Radio Frequency
SPST-NO	Single Pole, Single Throw, Normally Open
TAPI	Telephony Application Programming Interface

2. SYSTEM OVERVIEW

Figure 1 shows the system block diagram of the major components required to make *Intelli-Alert* ™. The pulse sensor detects the pulse of the user. When the sensor detects the user's irregular pulse beat or the panic buttons are pressed, the RF transmitter sends a radio signal to the RF receiver at the base station located in the house. The telephony system dials the pre-programmed numbers for help. Once the call is answered, the base station will play a pre-recorded message stating the address where help is needed.

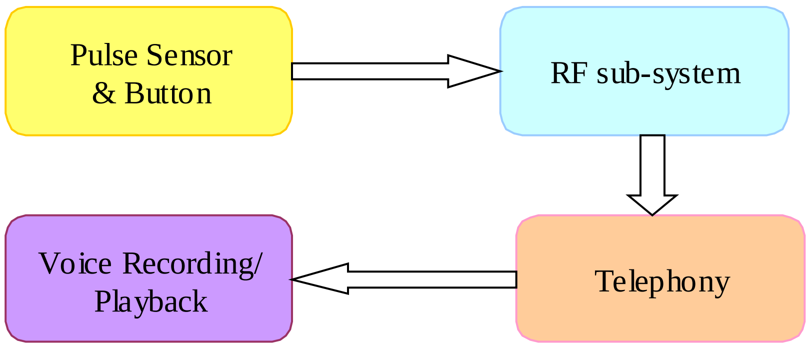


Figure 1 System Block Diagram

3. SYSTEM HARDWARE

3.1 Infrared LEDs and Phototransistors

The IR LED and phototransistor pairs are used to detect a human pulse from the fingertips. This is accomplished by placing the IR LED very close to the fingernail, pointing through the finger. The phototransistor is placed on the opposite side of the finger, pointing directly at the IR LED (see figure below).

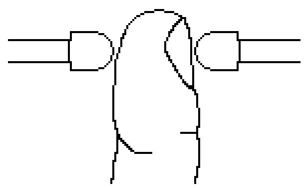


Figure 2 IR LED and Phototransistor Pair

As blood flows to the fingertip, it scatters some of the emitted IR light, and the detector will pass less current. Two different circuits can detect and amplify this small signal; the best and final circuit will be chosen after testing (see section 7, *Test Plan*).

Both circuits will consist of signal acquisition, filtering, and amplification (See figure below).

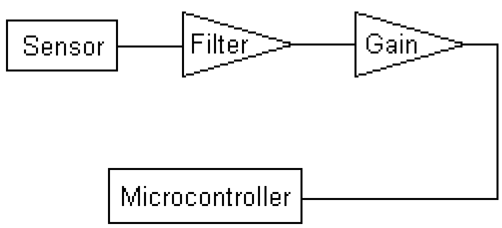


Figure 3 Pulse Sensor Diagram

The main difference between the circuits will be the point of signal acquisition (discussed in next section). Both will consist of the phototransistor and a current-limiting circuit in series, a filter and a gain amplifier (see following circuit diagrams).

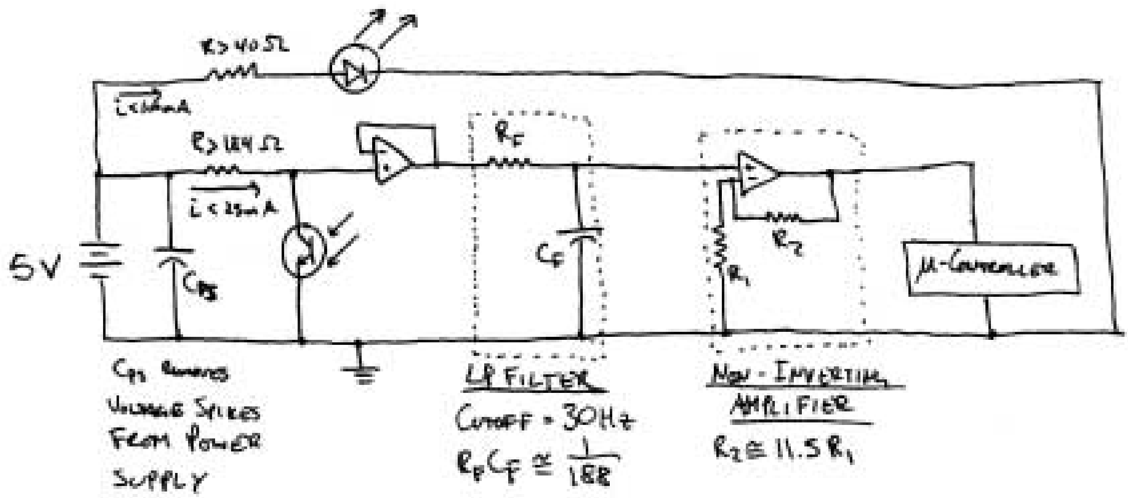


Figure 4 Pulse Sensor Circuit Diagram #1

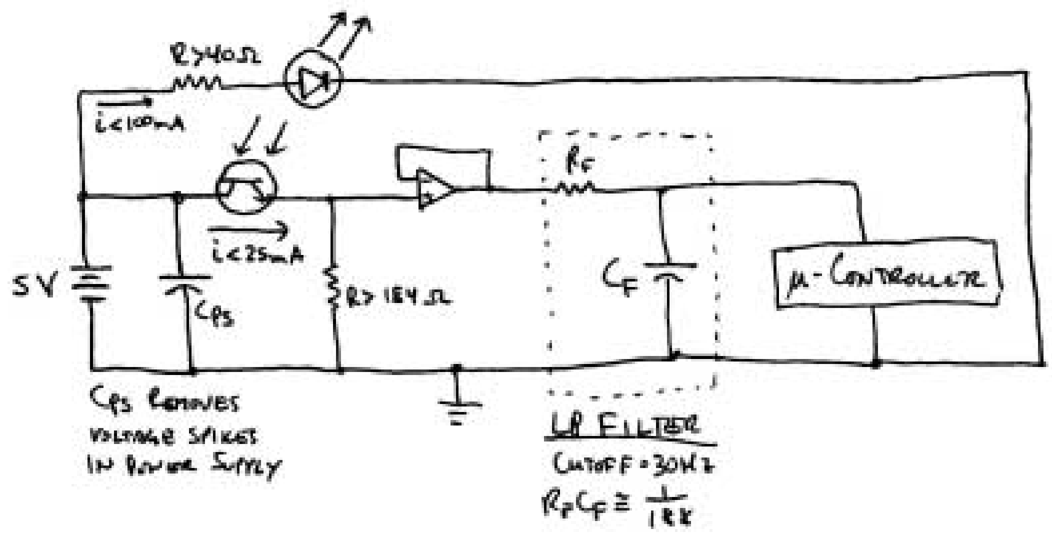


Figure 5 Pulse Sensor Circuit Diagram #2

We chose the infrared LED and phototransistor pair because it is cheap, easy to set-up, efficient and reliable.

- ### 3.1.1 Signal Acquisition

The first method (Circuit #1) involves signal acquisition (a voltage reading) across the phototransistor, and the second (Circuit #2) will be across the current-limiting resistor (also a voltage reading, see circuit diagrams above). Since the voltage drop across the phototransistor is a maximum of 0.4 V when completely saturated with IR light, the small signal of the first method will be no more than 0.4 V. In the second case (assuming a 5 V power source), the maximum expected voltage reading would be approximately 4.6 V. Realistically, the phototransistor will not become completely saturated, so we can expect voltages lower than 0.4 V in the first case, and greater than 4.6V in the second case.

- ### 3.1.2 Filtering

The IR phototransistor is very sensitive to IR light, and ambient conditions must be considered. The device is designed for indoor conditions, so the majority of ambient lighting can be attributed to incandescent and fluorescent lighting. Fortunately, filtering out this component is not difficult because the lighting is powered by a 60 Hz power supply. The IR light emitted from these lights peaks every 1/60 seconds. Since we are only interested in frequencies consistent with the human heartbeat (specifically, no larger than 150 beats/minute, or 2.5 Hz), we can build a filter to significantly attenuate frequencies above 60 Hz.

- ### 3.1.3 Amplification

Ultimately, we want our signal to pulse (or square-wave) at the frequency of the detected pulse, between 0-5 V (not exactly, ranges between 1-4V are also acceptable for the microcontroller) so that we can read the signal through an I/O line on the microcontroller. In the first circuit, this means we will have to amplify the signal by approximately 12.5 to 25 times, and in the second case, we may not need amplification. This will be accomplished using a non-inverting amplifier.

3.2 Push Buttons

Two SPST-NO switches (momentary push buttons) will be placed in series and connected to the microcontroller (See figure below).

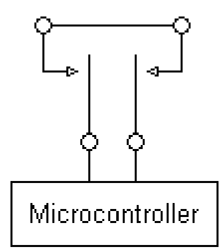


Figure 6 Push Button Circuit Diagram

They will be placed in series such that a connection is made only when both buttons are depressed simultaneously; this is to prevent accidental triggering of the device. This circuit will be connected to a single digital input line on the microcontroller. No current-limiting resistor will be placed in the circuit because the microcontroller's internal pull-up resistors will be enabled.

We chose push buttons because they are readily available, cheap, reliable, and use little power.

3.3 OOPic™ II (B.1.0) Microcontroller Evaluation Kit

The OOPic II microcontroller evaluation kit is a pre-assembled microcontroller built around a MicroChip PIC (PIC16C74b). It is easily programmable from PC and can be programmed in 3 different object-orientated languages: Java, C, or Basic.

We will be writing our code in C. Code (firmware) design will be discussed in section 5, *Firmware Design*.

We chose to use this microcontroller because its easy to program, comes pre-assembled, and has more than enough digital I/O and ADC lines. Although this microcontroller is significantly more costly than if we built our own based on the MicroChip PIC16C74b, it allows us to focus on the other, more important features of our product. Once these features are perfected, we can then focus on designing and programming the product around a more cost-effective microcontroller.

3.4 Link Master Evaluation Kit – HP Series

The Master Evaluation Kit (MEK) model # MDEV-900-HP produced by Linx Technologies will be used in the current phase of our project to implement the RF link between the remote device and the base station.

Included in the MEK are the following components:

- 2 assembled Development Boards w/Prototyping Area and RS-232 connector
- a transmitter module and a receiver module
- 2 Linx CW-series 1/4-wave connectorized whip antennas

We decide to use this kit for the following reasons. The RF modules and antennas are designed to be Plug & Play. An on-board encoder/decoder with buzzer and relay outputs allow range and interference testing in anticipated use environments. A convenient prototyping area with breakout headers and regulated power supply allows for rapid testing and interface. An on-board RS-232 serial interface and demonstration software are also provided.

• 3.4.1 The Master Development Board

A pair of master development boards is included in the MEK, one to be used with the receiver module and the other with the transmitter module. The schematic of the boards are the same, but the final layouts are slightly different. The schematic is shown in the following figure.

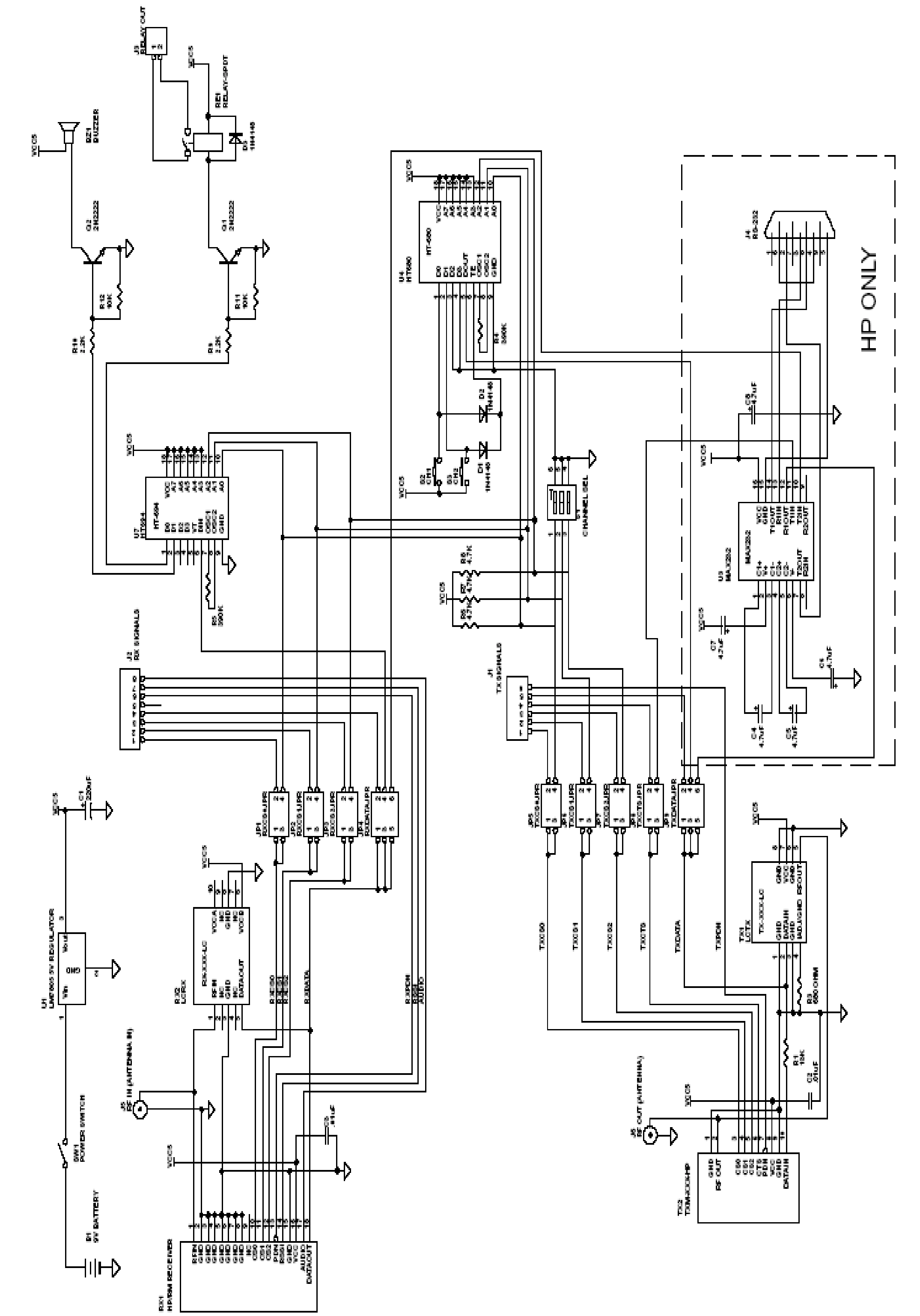


Figure 7 Linx Evaluation Board Schematics

A snapshot of the board set is also presented here for easier reference:

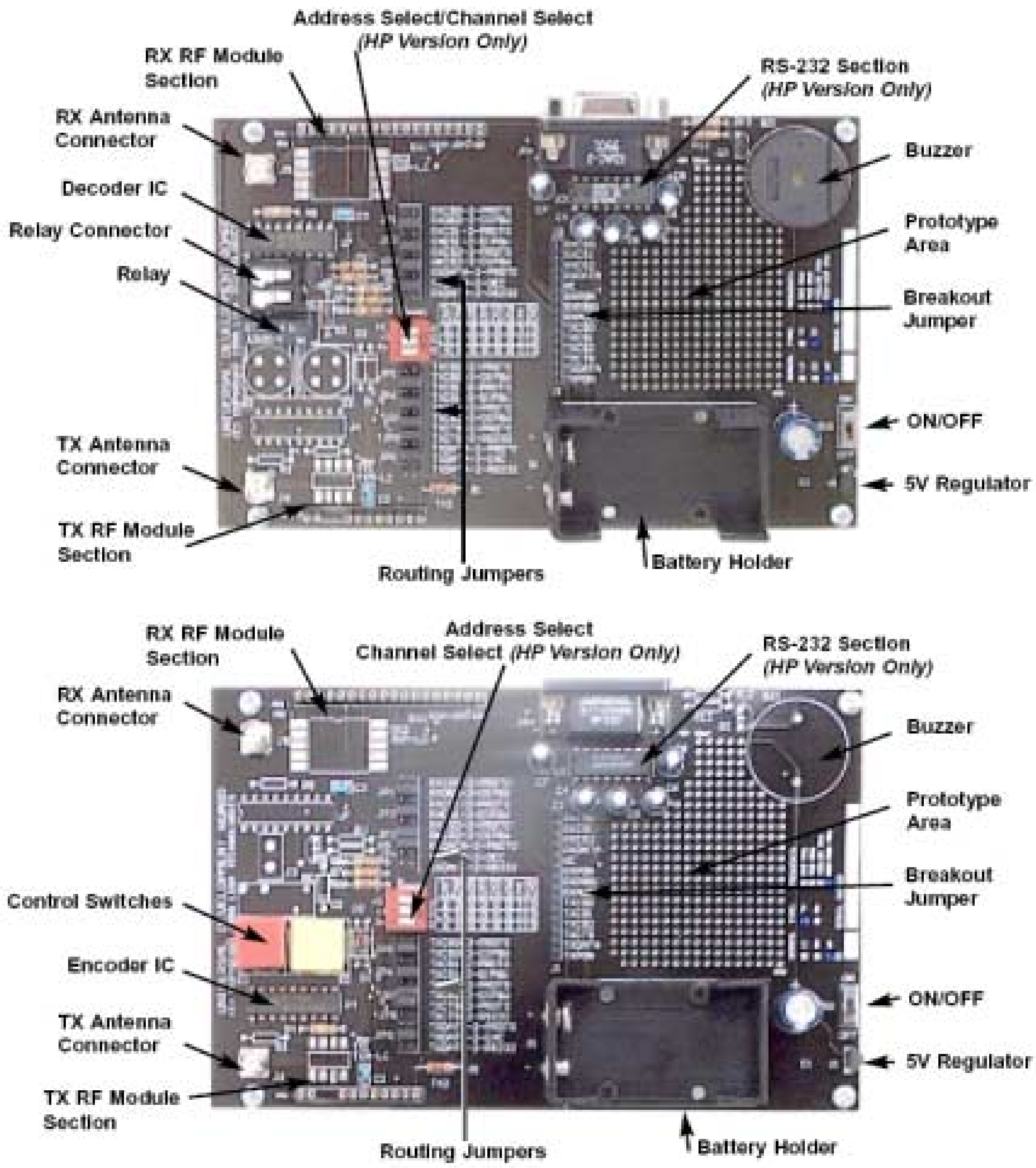


Figure 8 Linx Development Board Set Snapshot

As shown in Figure 3, the board that is to be used with the receiver module has the decoder IC, relay and relay connector, and Buzzer populated, and the transmitter board has the encoder IC and control switches populated.

The functions of the master development board are controlled by a series of 9 jumpers (JP1-JP9) which route the transmitter and receiver signals to the appropriate circuitry. JP1-JP3 are used to determine how the receiver channels are selected. JP4 is used to determine where the receiver's data output is routed. JP5-JP8 are identical to JP1-JP3 except that they apply to the channel-select lines of the transmitter. JP9 is same as JP4 except it routes the transmitter data input signal. They are shown in the following figure with brief explanations.

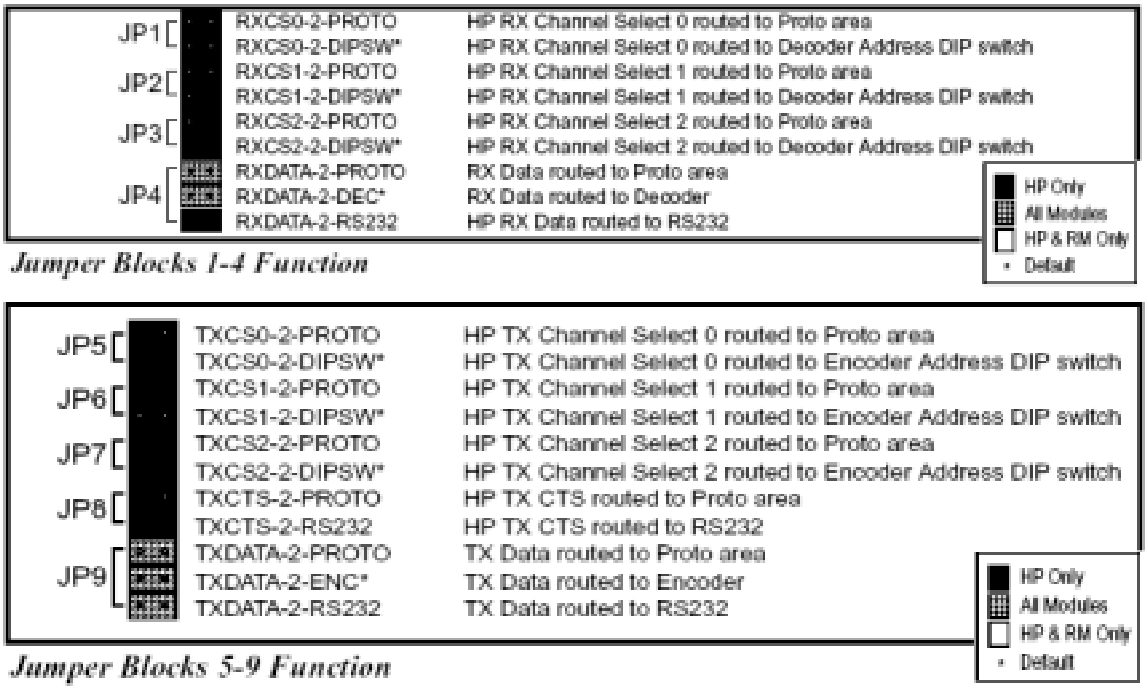


Figure 9 Jumper Functions of the MDB

All signals routed to the prototyping area are made available at the wire-wrap header labeled "J1/J2". J1 designates the section of the header which pertains to the transmitter module, while J2 refers to the header section dealing with the receiver signals. The layout of the breakout jumper is shown in the following figure.

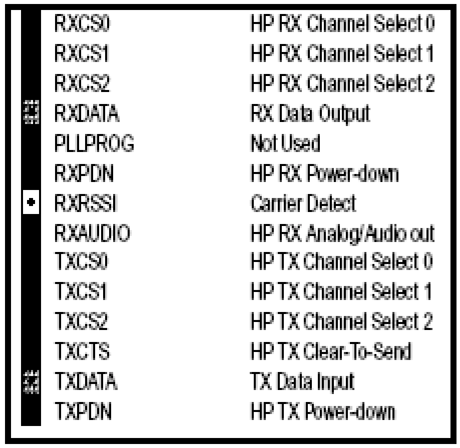


Figure 10 MDB Breakout Jumper Positions

• 3.4.2 The Transmitter Module

A TXM-900-HP-II transmitter module is included in the Linx MEK. This module features:

- 8 Binary Selectable Transmission Frequencies
- FM/FSK Modulation For Noise Immunity
- Precision Synthesized Frequency Reference
- Direct Analog/Serial Interface
- High Data Rate (50Kbps max.)
- Can Be Used To Transmit Analog (Including Audio) or Digital Data
- Wide Supply Range (2.7-16V DC)
- Power-Down & CTS Functions
- No Production Tuning
- No External RF Components Required (Except Antenna)
- FCC Compliant Output Power (0dBm typical)
- Manufacturing-Friendly SIP-Style Packaging

The block diagram of the transmitter module is shown below. Please refer to the *HP Series-II Transmitter Module Design Guide* for more detailed explanation for each individual part.

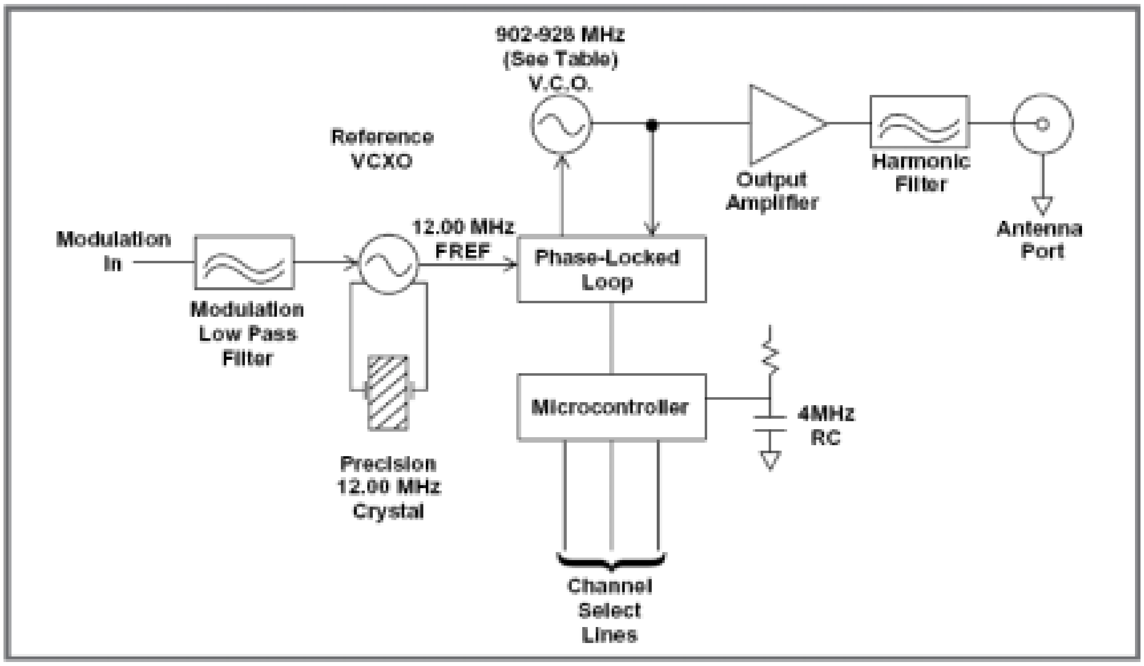
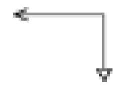
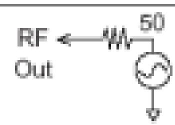
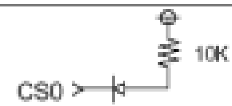
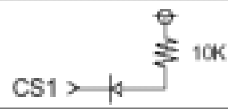
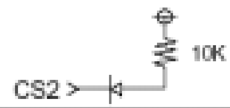
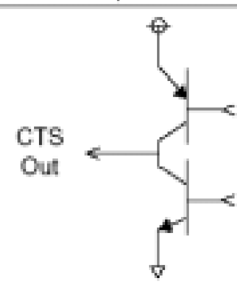
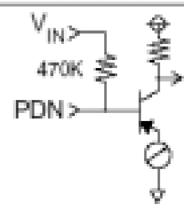
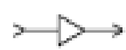
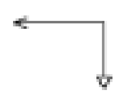
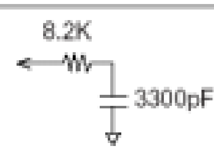


Figure 11 HP Series-II Transmitter Block Diagram

The transmitter is packaged as a hybrid through-hole SIP-style module with 10 pins spaced at .1” intervals with a .3” gap between pins 2 & 3. The following table gives a brief description for each pin.

Table 1 Pin Functions and Equivalent Circuits for the Linx Transmitter Module

PIN#	PIN Name	Equivalent CTK	Description
1	GND		Analog Ground
2	RF/ANT Out		50 Ohm RF Output
3	CS0		Channel Select 0
4	CS1		Channel Select 1
5	CS2		Channel Select 2
6	CTS		Clear-to-Send Output
7	PDN		Power down (Active Low)
8	VCC		Voltage Input 2.7-16V
9	GND		Digital Ground
10	Analog In/Data In		Digital/Analog Input 0-3V 0-5V *See text "Inputting Digital Data"*

The HP-II-TXM is controlled by an onboard microprocessor. When power is applied, a start-up sequence is executed. At the end of the start-up sequence, the transmitter is ready to transmit data. Figure 12 shows the start-up sequence. This sequence is executed when power is applied to the VCC pin or when the PDN pin is cycled from low to high. Once the initial power-on delay has been executed, the on-board microprocessor reads the external channel-selection lines and sets the frequency synthesizer to the appropriate channel. When the frequency synthesizer has locked on to the proper channel frequency, the circuit is ready to accept data. This is acknowledged by the Clear-To-Send (CTS) line transitioning high. The transmitter is then ready to accept modulated data from a user's circuit. The HP-II-TXM can be put into an ultra-low current (50µA) power-down mode by holding the PDN pin low. This removes all power from the transmitter's circuitry. If PDN is left floating or held high, the transmitter will wake up and begin normal operation. No transmitter functions work when PDN is low.

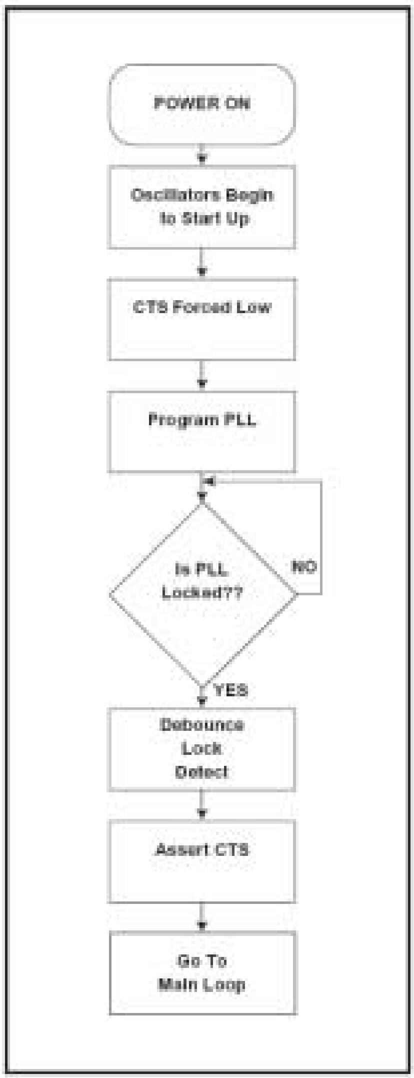


Figure 12 Linx Transmitter Module Start-up Sequence

- ## 3.4.3 The Receiver Module

A RXM-900-HP-II transmitter module is included in the Linx MEK. This module features:

- 8 Binary Selectable Reception Frequencies
- Exceptional Sensitivity (-95dBm @ 10-5 BER typical)
- High Serial Data Rate (50Kbps max.)
- Direct Serial Interface
- Fully Qualified Data Output
- Wide-Range Audio-Capable Analog Output (50Hz-25KHz)
- No External RF Components Required (Except Antenna)
- Manufacturing-Friendly SIP-Style Packaging
- Precision-Synthesized Frequency Reference
- Wide Supply Range (2.7-16V DC)
- Receive Signal Strength (RSSI) and Power down Pins
- No Production Tuning

The block diagram of the receiver module is shown below. Please refer to the *HP Series-II Receiver Module Design Guide* for more detailed explanation for each individual part.

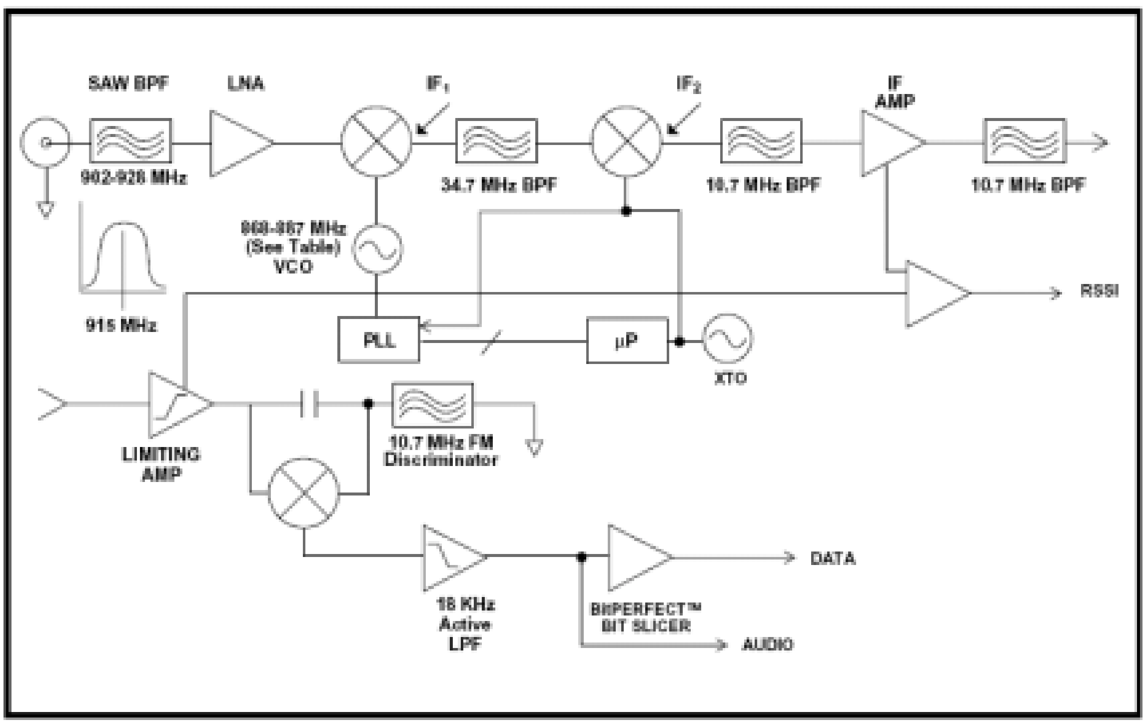
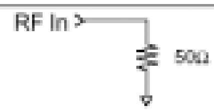
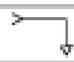
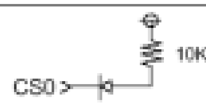
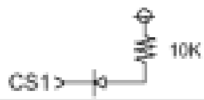
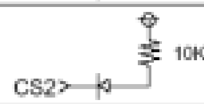
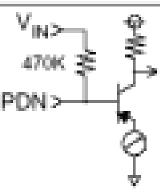
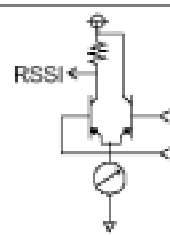
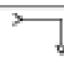
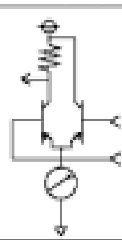
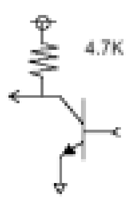


Figure 13 HP Series-II Receiver Block Diagram

The receiver is packaged as a hybrid through-hole SIP-style module with 18 pins spaced at .1” intervals. The following table gives a brief description for each pin.

Table 2 Pin Functions and Equivalent Circuits for the Linx Receiver Module

PIN#	Name	Equivalent CTK	Description
1	RF Input/ Antenna Input		50-ohm RF Input
2-8	Gnd		Analog Ground
9	N/C		No Connection
10	CS0		Channel Select 0
11	CS1		Channel Select 1
12	CS2		Channel Select 2
13	PDN		Power down (Active Low)
14	RSSI		Received Signal Strength Indicator
15	Gnd		Digital Ground
16	VCC		Voltage Input 2.7-16V
17	Analog Out		1V p-p Analog Output
18	Data Out		Digital Data Output

The HP-II-RXM is controlled by an on-board microprocessor. When power is applied, a start-up sequence is executed. At the end of the start-up sequence, the receiver is ready to receive data. Figure 14 shows the start-up sequence. This sequence is executed when power is applied to the VCC pin or when the PDN pin is cycled from low to high. On power-up, the microprocessor reads the external channel-selection lines and sets the frequency synthesizer to the appropriate channel. Once the frequency synthesizer has locked on to the selected channel's frequency, the receiver is ready to accept data.

Like the HP-II-TXM, the HP-II-RXM can be put into an ultra-low current (<50uA) power-down mode by holding the PDN pin low. If the PDN pin is left floating or held high, the receiver will turn fully on. In power down mode, the receiver is completely shut down. Thus, the RSSI circuit CANNOT be used to monitor for channel activity.

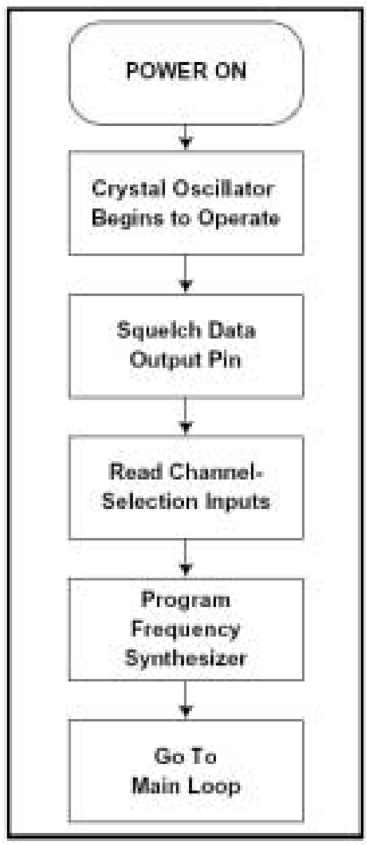


Figure 14 Linx Receiver Module Start-up Sequence

3.5 TAPI-Compliant Voice Modem

In the current phase of the project, we will use a TAPI-compliant voice modem to handle the telephony portion of our product.

TAPI, or Telephony Application Programming Interface, was developed jointly by Microsoft and Intel. TAPI technology assists the integration of computers and the telephone network and is at the heart of Windows Telephony.

Although Widows Telephony is universally known as TAPI this is only half of the story. The other half is TSPI. The two together make Windows Telephony.

- **TAPI:** The Windows Telephony Application Programming Interface (TAPI) is the tool kit to enable programmers to develop applications that provide personal telephony to users of the Microsoft Windows environment.

- **TSPI:** The Windows Service Provider Programming Interface (TSPI) is the tool kit to enable programmers to develop back-end services that handle requests from applications that conform to the Telephony API.

Windows Telephony is a component of the Windows Open Services Architecture. Microsoft supplies a Telephony DLL component called TAPI.DLL. This component resides between the Telephony API that is called by applications and the Telephony SPI that is implemented by Service Providers. TAPI compliant applications call TAPI functions managed by TAPI.DLL and routed to the appropriate service provider for execution. The relationships between these parts are shown in the following diagram.

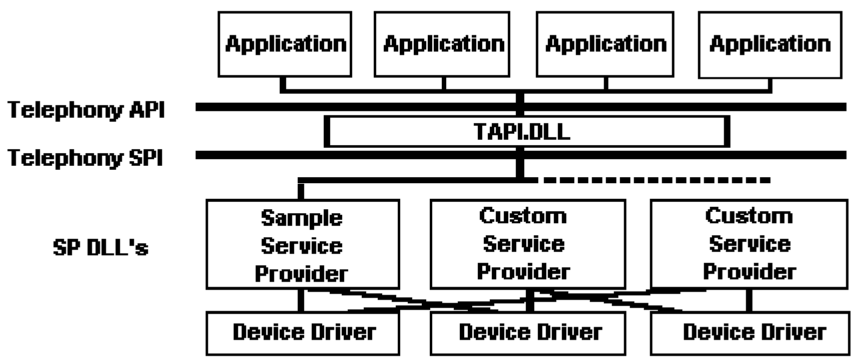


Figure 15 A Illustration of Windows® TAPI and TSPI Layer

TAPI sets up the path for the call but does not interpret the data on the path. The media mode is the form in which data is transmitted on the path. Media modes are speech, facsimile & data. TAPI applications require appropriate media stream API to handle the media stream.

4. SYSTEM INTERFACE

4.1 Interface between Sensor/Push Button and Microcontroller

Both the pulse sensor and push buttons will be interfaced directly into the microcontroller (as shown in the circuit diagrams in the previous section). Each will be connected to one of the microcontroller's digital I/O lines. The OOPIC's internal pull-up resistors will be enabled to prevent damage to the microcontroller. We may also read the voltage across the phototransistor or its current-limiting resistor directly to detect when the finger has been removed. When the finger is removed, the IR light will be unimpeded, resulting in saturation of the phototransistor. We may use an ADC input line for this. This would prevent false alarms (0 pulse rate, cardiac arrest) when the sensor is removed from the finger.

4.2 Interface between Microcontroller and Linx TX MDB

The digital output from the microcontroller is to be directly connected to the data input pin on the Linx MDB with the transmitter module. Set jumper JP9 to 'TXDATA-2-PROTO' position. Because the microcontroller is a 5V source while the Linx module is optimized for serial data that transitions from 0V to 3V, a 2.2K resistor is required to be connected in series with the data pin.

4.3 Interface between Linx RX MDB and PC

The Linx MDB with the receiver module will be connected to a PC via a serial connection. Set jumper JP4 to 'RXDATA-2-PROTO' position.

5. FIRMWARE DESIGN

The firmware required will be simple and efficient. The microcontroller will be monitoring a few different conditions, and then will act upon those conditions. The code will be interrupt driven using the OOPIC's virtual circuits.

The code will simply handle the following two conditions (through interrupts):

- Both buttons depressed simultaneously
- Pulse rates below 50 or above 120 beats per minute

If either of these conditions is detected, one of the digital I/O pins will go high, and the emergency signal will be transmitted.

As well, we may read the voltage signal across the phototransistor or its current-limiting resistor using an ADC input line to detect when the finger has been removed. With finger removed, the reading will be significantly higher than when the finger is in place.

If 0 pulses per minute is detected, the ADC input line will be checked, and if the voltage is above a certain voltage threshold (indicating that the finger has been removed), no alarm will be signalled (although the button alarm signal will still be enabled).

6. APPLICATION SOFTWARE DESIGN

The main purposes of the software are: 1) provide the user an interface to record message and set phone numbers to be dialled under emergency situation. 2) make emergency phone calls when radio trigger is received. The software will be developed using Microsoft™ Visual Basic 6.

6.1 Graphical User Interface Design

Figure 16 shows a mock up of the final GUI. This control panel contains buttons that allow user to perform tasks such as enter and save phone number, record and play voice message, make a emergency phone call and cancel a false alert.

Figure 16 A Mock-up of the Final GUI

On the right hand side of the control panel are the buttons used to enter phone numbers. The display window at the right up corner displays the number the user just entered. In the middle of the control panel are a group of four buttons named “Mem 1” to “Mem 4”. They are used to select the location to save a phone number. The display windows just above each button shows the current phone number stored in this memory location. If the display window is blank, it means no phone number is currently stored in this memory location. Finally, on the left hand side of the control panels are buttons designated to record/play message, make phone call and cancel a false alert. A status window is also present to display the current status of the control software. The red “Shut Down” button at the right bottom corner is used to close this control panel. When clicked, a confirm window will pop up to ask user for confirmation. This is shown in figure 17. Click “Cancel” will bring user back to the control panel, and click “OK” will quit the program.

Figure 17 Confirmation Message Box of the Control Software

6.2 Low Level Software Design

The control software is also responsible for monitoring the serial port for trigger sent by the Linx RX MDB. As soon as the trigger signal is detected, the control software will make emergency phone calls via the TAPI-compliant voice modem. The following figure shows the flow chart of how the software functions. Refer to section 3.2 for more details regarding TAPI.

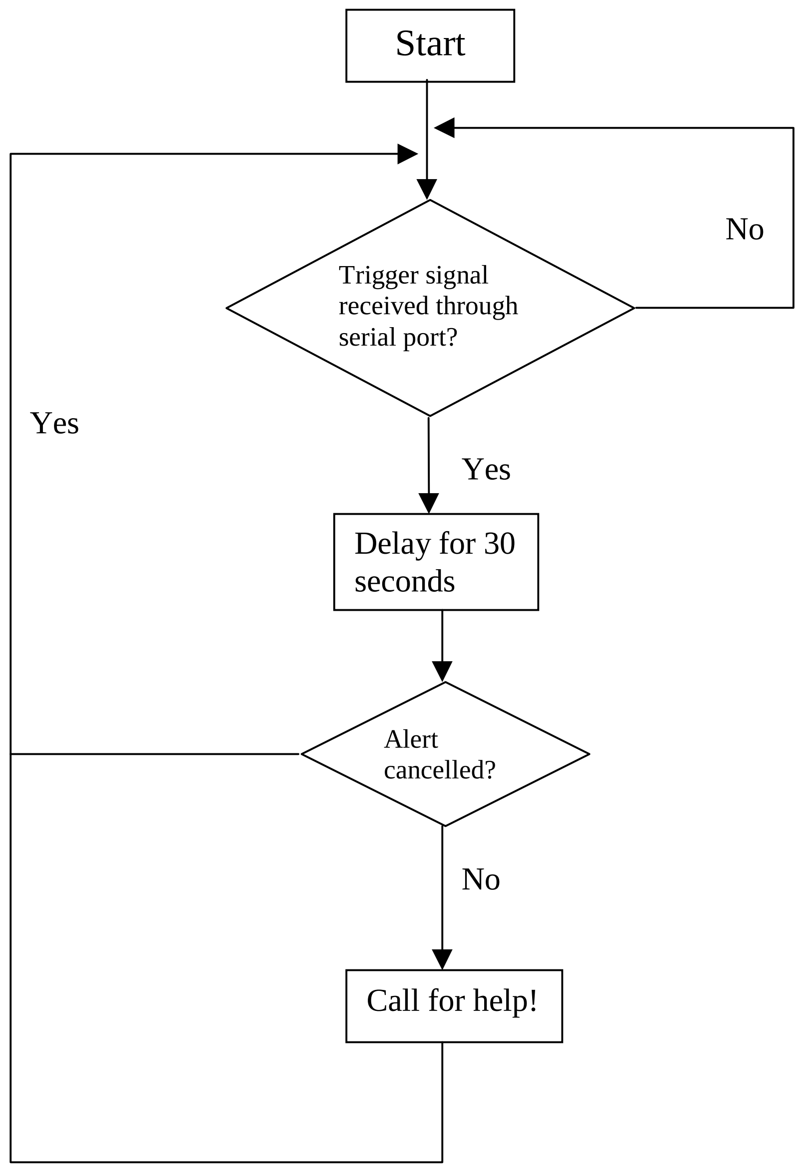


Figure 18 Control Software Flow Chart

7. TEST PLAN

7.1 Hardware Test Plan

- ### 7.1.1 Pulse Sensor

Both proposed pulse sensor circuits will be implemented on a breadboard using the stable lab power supplies and scoped using an oscilloscope at different points in the circuit. These points will be chosen such that we can verify each component is working properly. Once we have verified this, we will scope the output (where the microcontroller will acquire the signal) with a human finger.

Depending on the measured signal, we may modify our resistor and capacitor values, to ensure that our signal is strong and defined. As well, to suit different human fingers, we may find it valuable to insert a potentiometer in place of one of the resistors in the non-inverting amplifier circuit so that we can tune the circuit to get the best signal possible.

Once we have finalized our component values, the components will be soldered onto circuit board and the final output will be scoped to verify that the soldered circuit has been built properly.

At this point, we will connect our battery power supplies and again scope the circuits to ensure that the power supply capacitor is satisfactorily removing any noise and voltage spikes from the battery.

- ### 7.1.2 Push Buttons

Each push button will be tested separately before being implemented into any circuits. A DMM will measure the resistance across the terminals of each button. When the button is not depressed, the DMM should measure an infinite resistance (no reading on the DMM) and when it is depressed, the DMM should display a very small resistance (very small, possibly even read 0 ohms). Once verified, the push button circuit will be built, and tested again in a similar manner (this time the DMM will measure resistance across the buttons in series). The DMM should measure an infinite resistance if no, or only one of the buttons is depressed, and should measure zero resistance if both buttons are depressed.

- ### 7.1.3 OOPic™ II (B.1.0) Microcontroller Evaluation Kit

To verify that the needed digital and ADC I/O lines work properly, simple circuits will be attached to these lines and tested. The final product needs digital input and output as well as ADC input.

For digital output lines, a visible light LED in series with a current-limiting resistor will be attached to the line. Simple code will be written to toggle this line at a set rate (1 Hz), and the LED should blink at this rate.

To test digital input lines, we will attach a working SPST-NO push button to the line, and simple code will be written to read this line (a one bit value, 0 or 1) and display it on a small 2 line LCD attached to the microcontroller. When the button is depressed, the display should change from 0 to 1.

Finally, to test the ADC line, a simple voltage divider circuit will be built, and the ADC line will be connected across the output resistor. A lab power supply will be used to power the circuit. The power supply voltage will be varied by hand such that the range of voltages across the output resistor will be 0 to 5 Volts. Simple code will be written to read the ADC value (one byte, 0 to 255, where 0 corresponds to 0 volts and 255 corresponds to 5 volts) and display it on the LCD.

• 7.1.4 Link Master Evaluation Kit – HP Series

First, we will perform a range test to make sure the hardware is working properly. Plug the transmitter module to the MDB that has 2 push buttons and an onboard encoder. Plug the receiver module to the MDB that has a buzzer and an onboard decoder. Set the address select dip switch to the same setting. The jumpers are to be set as follow:

JP1 – RXCS0-2-DIPSW
JP2 – RXCS1-2-DIPSW
JP3 – RXCS2-2-DIPSW
JP4 – RXDATA-2-DEC
JP5 – TXCS0-2-DIPSW
JP6 – TXCS1-2-DIPSW
JP7 – TXCS2-2-DIPSW
JP8 – not used in range test
JP9 – TXDATA-2-ENC

Power up both boards, the buzzer should sound if the push button is pressed.

Second, we will transmit a square wave over the radio link. Set the jumpers as described above except set JP4 to ‘RXDATA-2-PROTO’ and JP9 to ‘TXDATA-2-PROTO’. Generate a 0V-3V square wave using the function generator. Connect the signal to the ‘TXDATA’ pin of the transmitting board. Monitor the ‘RXDATA’ pin of the receiving board using an oscilloscope. Make sure that the signals are properly grounded since this is critical in successfully transmitting radio signal.

• 7.1.5 TAPI-Compliant Voice Modem

Follow the steps to make sure the modem to be used has voice capability: (assume windows XP® operating system)

1. On desktop, right click on ‘My Computer’ icon and choose ‘Properties’ on the pop-up menu.
2. In the ‘System Properties’ window, click on the ‘Hardware’ tab and then click ‘Device Manager’ button.
3. In the ‘Device Manager’ window, click the ‘+’ sign next to ‘Sound, video and game controllers’ to expand this item and make sure ‘Unimodem Half-Duplex Audio Device’ is present.

7.2 Interface Test Plan

- ### 7.2.1 Interface between Sensor/Push Button and Microcontroller

To test the interface between the pulse sensor and the microcontroller, simple code will be written to display the value of the digital input line where the pulse sensor is being read. With a finger in the sensor, the output should toggle between 0 and 1, in unison with the pulse rate.

A similar method will be used to test the push button interface; the value should read 0 on the LCD if neither or only 1 button is depressed, and a 1 should be displayed when both buttons are depressed.

- ### 7.2.2 Interface between Microcontroller and Linx TX MDB

To test the interface between the microcontroller and the Linx TX MDB, simple code will be written to toggle the value of the microcontroller output pin between ‘0’ and ‘1’. We will then probe the DXDATA pin on the RX board to see if we get a square wave on the oscilloscope.

- ### 7.2.3 Interface between Linx RX MDB and PC

To test the serial interface between the Linx RX MDB and PC, we will use the demo software provided by Linx Technologies. To use this software, both boards are to be connected to two PCs via serial port. We can use the software on the PC that is connected to the TX board to send text string or even pictures to the TX board, which will then send the data over the radio link. After the RX board received the data, it will output to the computer connected to it. We can check on the computer to see if correct data is received.

7.3 Software Test Plan

First, the software will be tested to see if it can respond to user input from the mouse or the keyboard. Then phone numbers will be entered and saved. We must make sure that the phone number is properly saved on disk and can be reloaded next time when the software is loaded.

Second, the software will be tested to see if it can read data from the serial port. Simple data strings will be fed to the serial port, the software must be able to capture the input data stream and display it on screen

Third, we will test the recording and play back capability of the software. A voice message will be recorded using the software. The software must be able to save this voice message and play it when requested.

Last, the software will be tested to see if it can make calls via the modem. Calls will be initiated using the software. We must make sure that calls made by the software can be received on the other end.

8. CONCLUSION

This document contains the design specifications for the prototype version of our complete wireless emergency response system, called *Intelli-Alert*™. The prototype is specified to prove the concept of monitoring pulse rate in an effective and convenient way and communicating wirelessly between modules using RF signal. Given these specifications, we are confident that we will satisfy our minimum requirements and more by the end of December 2003.

9. References

- ENSC305/340 Website (http://www.ensc.sfu.ca/users/whitmore/public_html/courses/305/305.htm)
- Dr. Andrew Rawicz, School of Engineering Science, Simon Fraser University
- 5mm, IR LED, KIE 7304 Specification (http://www.rpelectronics.com/Data/S-KIE-7304.GIF)
- 5mm, IR Phototransistor, KID7407 Specification (http://www.rpelectronics.com/Data/S-KID7407.GIF)
- Linx Technologies (http://www.linxtechnologies.com)
- Wilco Telephony Ltd. (http://www.wilco-telephony.co.uk/index.html)