RANGE FINDER AND SCORE DATABASE FOR GOLF APPLICATIONS

A thesis submitted in partial fulfillment of the requirement for the degree of

Bachelor of technology

In

Electronics & Instrumentation Engineering

by

Subhasis Chand

Roll No - 109EI0263

&

Asutosh Patro

Roll No – 109EI0567

Under the guidance of

Prof. S. K. Das

CERTIFICATE

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

This is to certify that the project report titled "Range Finder and score Database for Golf Applications" submitted by Subhasis Chand (Roll No: 109EI0263) & Asutosh Patro (Roll No: 109EI0567) in the partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Electronics and Instrumentation Engineering during session 2009-2013 at National Institute of Technology, Rourkela, is an authentic work carried out by him under my supervision and guidance.

Prof S. K. Das

Department of Electronics and Communication Engg.

Date National Institute of Technology

Rourkela - 769008

ACKNOWLEDGEMENT

We would like to take this opportunity to express our gratitude and sincere thanks to our respected supervisor Prof. S. K. DAS for his guidance, insight and support he has provided throughout the course of this work. We would like to articulate our profound gratitude and indebtedness to Prof. K. K. MAHAPATRA, Prof. A. K. SWAIN and Prof. D. P. ACHARYA for teaching “Embedded Systems” in such depth and setting up a strong fundamental base in Embedded Systems that enabled us achieve this success. We would like to thank all faculty members and staff of the Department of Electronics and Communication Engineering, N.I.T. Rourkela for their extreme help throughout course.

An assemblage of this nature could never have been attempted without reference to and inspiration from the works of others whose details are mentioned in the reference section. We acknowledge our indebtedness to all of them.

Last but not the least our sincere thanks to all our friends, who have patiently extended all sorts of help for accomplishing this undertaking.

Date: 8th May, 2013 **SUBHASIS CHAND (109EI0263)**

Rourkela **ASUTOSH PATRO (109EI0567)**

Abstract

The objective of this project is to develop a device capable of calculating distance between two points on surface of earth. This device finds its application in professional golf courses where the members need to change their stick depending upon distance between flag and the person. There are a number of devices available in the market for this purpose. But the methodology and components used by them are costly and takes a lot of time. The methodology they use is that they get connected to Google earth (i.e. internet). Prior to that this device is very user friendly, and stores all the recorded data and scores onto a remote server by wireless communication. This feature is not supported by any device related to this in market. This reduces work of the person in charge of the golf course. The database can be accessed only at that server (i.e. PC) by means of a GUI. The database and GUI are password protected and can be accessed by both players and administrators. The technology used behind this is use of GPS modem to extract GPS location of two points and calculating distance between those two points using an algorithm. The GPS location consists of latitude and longitude. Any point on surface of earth can be represented by these two values. And the device is named as "Range Finder for Golf Application".

CONTENTS

LIST OF FIGURES

CHAPTER.1

INTRODUCTION

1.1 RANGE FINDER

Range finder is a device usually used to determine the range between two points. In this thesis the range finder is used to determine the distance of a golf player from his/her target i.e. the flag. According to the distance measured the player can choose his club/stick for a perfect shot. The range finders can be built using laser, sonar or GPS. In this project the range finder is based on GPS, which is very accurate in practical outdoor applications

1.2 GLOBAL POSITIONING SYSTEM (GPS)

The **Global Positioning System** (**GPS**) is a space-based satellite navigation system. Regardless of any weather condition time and location can be obtained from the GPS device anywhere on or near the Earth provided that it can get signals from at least four GPS satellites. It has many applications in civil, military and day to day travelling. The GPS satellites are launched by the United States government. But anyone having a GPS device can access to the signals as it is open for common people.

1.3 PROCESSING UNIT

Like most of the embedded system development, in this device a microcontroller (ATmega32) is used as the central processing unit. It controls all the inputs, outputs and interrupts. All electronics hardwares are controlled by it.

1.4 TRANSMISSION UNIT

For transmission of data (here the score after the game) is done using Zigbee wireless module. The score is transmitted from the device held by the player to the central database.

CHAPTER. 2

LITERATURE SURVEY

The previously existing devices are slow and expensive. Most of them do not use a touch screen. Bushnell Yardage Pro is one such a device. It only displays current position of the player and his distance from the flag. It does not store the score or use a database. The project described here overcomes all the above drawbacks. It is fast, uses a touch screen for user friendly.

Another thing of importance in a golf match is keeping track of score of players. In professional golf courses, score is generally no. of strokes played to pocket a ball. So using this device the score can be generated from touch interface on the display and is transmitted to a central data base by Zigbee modem which range with antenna is up to 1 kilo meter in line of sight. The central database stores the score of all the players and only admin can update it. It is a password protected secured interface.

CHAPTER. 3

DEVELOPMENT

3.1 HARDWARES USED

3.1.1 ATMEGA32 MICROCONTROLLER AND DEVELOPMENT BOARD

The Microcontroller used for development of this device is AVR Atmega 32 which is manufactured by Atmel Corporation. The ATmega32 is a low-power CMOS device. It is an 8-bit microcontroller based on enhanced RISC architecture. It can execute most of the instructions in single clock cycle time period. So the power consumption and processing speed can be optimized by developer and programmer.

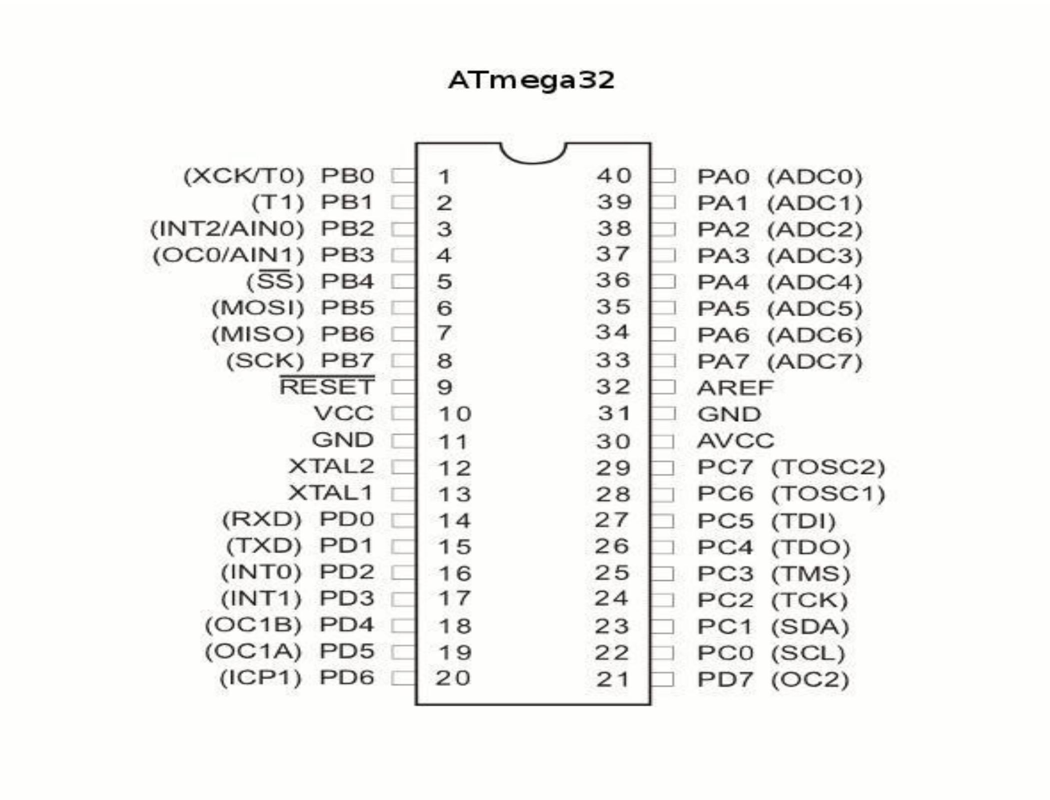


Fig 3.1: Atmega 32 Pin Configuration

Specifications of ATmega32 are

- Advanced RISC Architecture
- 131 Powerful Instructions – Most Single-clock Cycle Execution
- It has32 numbers of 8bit General Purpose Registers
- The 32KB of flash memory is inbuilt and programmable
- 1024 Bytes EEPROM
- 2K Byte Internal SRAM
- Write Cycles and erase cycle: 10,000 for Flash / 100,000 for in built EEPROM
- Data can be stored for 20 years at 85°C and for 100 years in at room temperature.

Interface

▪ Peripheral Features

- High-performance, Low-power AVR 8-bit Microcontroller
- 32 Programmable I/O Lines
- It has two 8-bit Timer/Counters. They have Separate Presales and Compare Modes
- 8-channel, 10-bit ADC 8 Single-ended Channels
- Duplex serial transmission with one byte at a time
- Programmable Serial USART
- Master/Slave SPI Serial Interface
- On-chip Analog Comparator
- 4.5 - 5.5V operating voltage
- 16K Bytes of In-System Self-programmable Flash program memory
- 512 Bytes EEPROM
- 1K Byte Internal SRAM

▪ Speed Grades

- 0 - 8 MHz for ATmega32 without external oscillator
- 0 - 16 MHz for ATmega32 with external oscillator

For this device the complete AVR ATmega 32 board has been chosen. The ATmega 32 development board from Nex- Robotics is displayed in Figure 3.2.

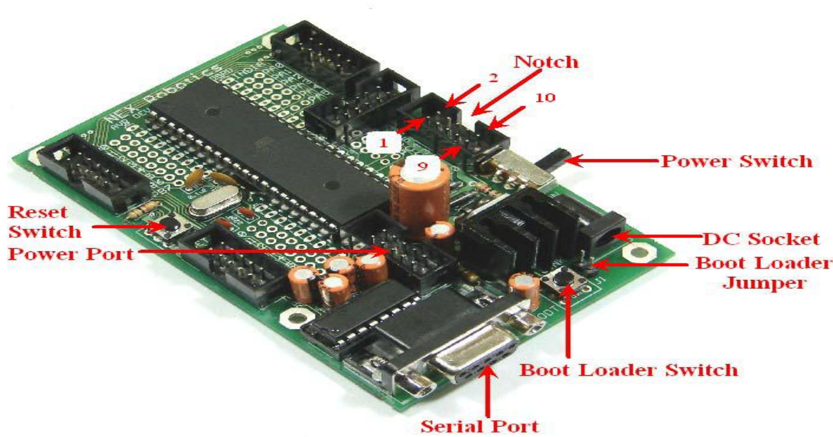


Fig 3.2: Atmega 32 Development board by Nex robotics

Specifications of development board are

- Microcontroller: ATMEGA 32. Crystal frequency is 14.7456MHz.
- The board is double sided high quality PTH PCB.
- Power handling capability: 7 to 15 volts.
- Reverse polarity protected.
- Switches: Boot, Reset, Power.

- RS232 serial interface.
- 10 pin FRC connectors and soldering pads on all ports.
- It is a general purpose development board stackable design.

3.1.2 GPS MODULE

The GPS Module used in this device is GPS Receiver MT3318 Module. GPS Receiver MT3318 Module uses the MediaTek MTK MT3318 chipset. Cirocomm provides active patch antenna for it. It has capability of tracking 51 satellites at a time. The PCB includes GPS receiver, 3.3v voltage regulator power and transmit, receive indicating LED. Schmitt trigger based buffer for 5V to 3.3V logic level conversion.

This GPS receiver gives data output in standard NMEA format with update rate of 1 second at 9600 bps. Receiver has onboard battery for memory backup for quicker acquisition of GPS satellites. It can be directly interfaced with the 5V TTL / CMOS logic. Figure 3.2 shows GPS Module.

Fig 3.3: GPS Module

Pin Description:

5V (Supply Voltage): Give 5V supply to this pin.

GND (Ground): Connect to supply ground

Rx (Receive): This pin is for serial communication. It gives serial data from external UART to GPS module.

module to receive the NEMA commands.

Tx (Transmit): This pin is for serial data transmission.Serial data from GPS module goes to external devices.

UART to transmit the NEMA commands.

3.1.3 RESISTIVE TOUCH SCREEN

A four-wire resistive touchscreen is a sensor consisting of two transparent resistive plates, ideally of uniform resistivity; those are separated by insulating spacer material. The metalized contacts of the "x" layer run along the y-direction and thus the resistance is measured between the two x-direction ends. Similarly, the "y" layer has metalized contacts that run in the x-direction so that the resistance is measured along the y-axis (see Figure 3.3).

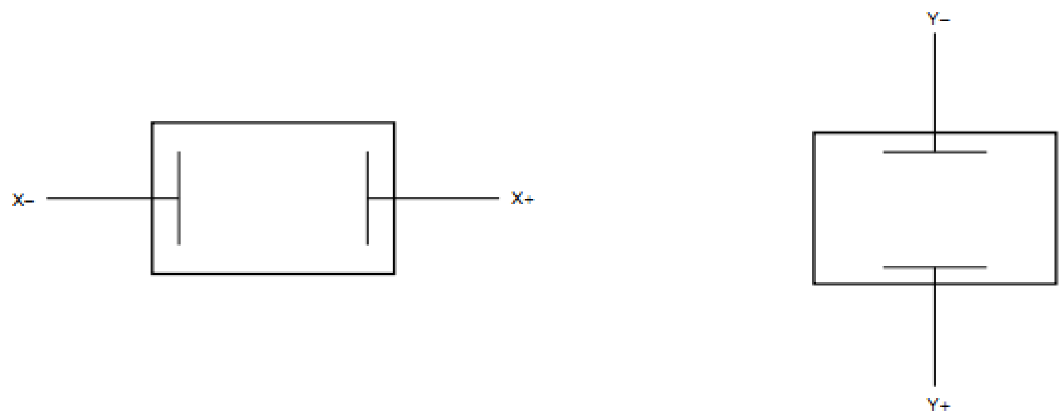


Fig 3.4: Structure of Resistive Touchscreen

When touched with sufficient pressure, the bottom plate is touched by the deformed top plate. At the point of contact, the top layer is divided into two resistors in series by touching the bottom layer, it is similar to the wiper on a potentiometer divides the potentiometer into two series resistors. Same thing happen to the bottom layer. And it is effectively divided into two resistors at the point of contact with the top layer. Each plate is analogous to the two ends of a potentiometer where the other plate serves as the wiper. (see Figure 3.4). With proper biasing, each plate functions as a potential divider where the output or the wiper analog voltage represents the rectangular coordinate of the point of contact.

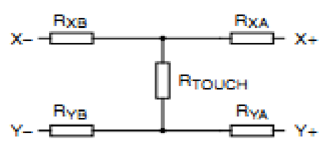


Fig 3.5: Resistive divider form by touching the screen

Biasing the x-axis allows us to use the y-axis to measure the tap on the x-axis. In a similar manner, biasing the y-axis allows us to use the x-axis to measure the tap on the y-axis (Fig 3.4). Biasing both axis can be used to have the hardware detect when the screen has been touched and generate an interrupt (see Figure 3.5).

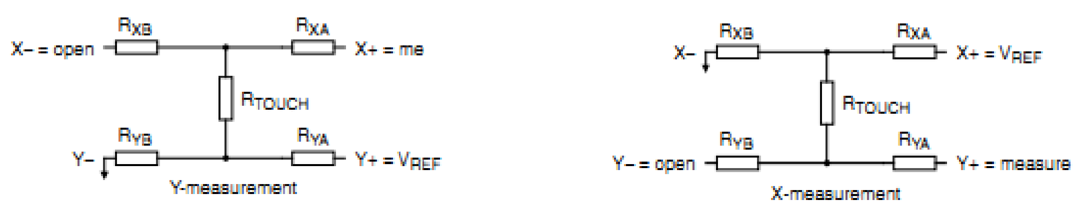


Fig 3.6: Touchscreen biasing for detecting ‘x’ and ‘y’ voltages

This touch screen is interfaced using the inbuilt ADC channels of ATmega32. When the analog voltage comes to the channel it is converted into digital values, which can be used for further calculations.

3.1.4 PROGRAMMER

NEX AVR USB ISP STK500V2 is a high speed USB powered STK500V2 compatible In-System USB programmer for AVR family of microcontrollers. It is used on Windows XP platform with AVR Studio. For Windows7 it is be used in SPI communication mode with SinaProg GUI as programming interface. For lower clock speeds the fuse bits are burn into the microcontroller by the SinaProg. The programmer does not need any external power supply as it is powered directly from a USB port. But the target boards need external supply as the current from burner is limited to 100mA.

Features

- Low cost USB compatible (No legacy RS232 required)
- Compatible with STK500V2Supported by AVR Studio as STK500 programmer
- Supported by AVRdude on Win7/XP/Vista
- Jumper adjustable programming clock speeds for low clock speed microcontrollers. Speed is adjustable from 32KHZ to 8MHz.
- Programs almost all AVR microcontrollers (Refer Table below)
- Jumper selectable HID/CDC mode.
- USB powered and no external power required
- Jumper enabled 5V power supply for target boards
- Standard 10 pin (5x2) programming connector
- The on board LEDs indicate power and programming activities

Fig 3.7: USB Programmer and its pin configuration

3.1.5 GRAPHICAL LCD

The graphical LCD interfaced here has following features

- It has 64 vertical pixel and 128 horizontal pixel resolution.
- Controlled based on KS0108B
- Parallel 8bit interface
- On board graphic memory.
- LED backlight.
- 20 PIN linear connection.

3.1.6 ZIGBEE WIRELESS MODULE

ZigBee is a specification for a suite of high level communication protocols used to create personal area networks. It is built from small, low-power digital radios. IEEE 802 standard is assigned to Zigbee. Even if it consumes low power, ZigBee devices can transmit data over longer distances by passing data through intermediate devices. So a mesh network can be established without any external control or high power. With high gain antenna wireless ad-hoc networks make them suitable for applications where a central node can't be relied upon.

ZigBee can be used in the application which require a low data rate, long battery life, and secure networking. ZigBee has a defined rate of 250 kbit/s, best suited for periodic or intermittent data or a single signal transmission from a sensor or input device. Applications include wireless light switches, electrical meters with in-home-displays, traffic management systems, and other consumer and industrial equipment that requires short-range wireless transfer of data at relatively low rates. The technology defined by the ZigBee specification is intended to be simpler and less expensive than other WPANs, such as Bluetooth or Wi-Fi. Zigbee networks are secured by 128 bit encryption keys.

The Zigbee module used in this project is a XBee 802.15.4. It is used for embedded solutions providing wireless end-point connectivity to devices. This is perfect for robot to pc communication. In indoor condition its range is up to 100 meter. Microcontroller can be interfaced with ZigBee directly by serial port.

Specifications

- Model code: XB24-ACI-001
- Operating Frequency: ISM 2.4 GHz
- Antenna type: Chip antenna
- In indoor condition its range is 30 m
- In outdoor condition its range is or90 m

- It can be interfaced serially with baud rates 1200-115200 bps
- Supply Voltage: 2.8 – 3.4 V
- Transmit Current (typical) 45mA (@ 3.3 V)
- Idle / Receive Current (typical) 50mA (@ 3.3 V)
- In centimeter its dimension (2.438cm x 2.761cm)
- Operating Temperature: -40 to 85º C (industrial)

Features

- All Network Topologies supported.
- Number of Channels: (software selectable) 16 Direct Sequence Channels
- Addressing Options: PAN ID, Channel and Addresses
- Channel capacity: 16 Channels
- 16bit address line. Hence 65000 addresses available for each channel
- Module parameters can be set using AT and API Command
- Extensive command set
- Small form factor

3.2 SOFTWARES USED

3.2.1 AVR STUDIO 4

To work with the Atmel AVR microcontroller using the C programming language, two tools are necessary: AVR Studio is available online and WinAVR. tool is free.

An editor, an assembler, HEX file downloader and a microcontroller emulator is included in AVR integrated development environment that includes.

WinAVR is for a GCC-based compiler and is a plug-in in AVR Studio. In WinAVR Programmer's Notepad programs can be edited and compiled in C , independent of AVR Studio softwareopen.

The major stages are :

- WINAVR is installed on the PC
- AVR Studio 4.1.7 is installed on PC
- AVR Studio is opened and a new project is created
- While creating a new project, its type is "AVR GCC", debug platform is "AVR Simulator" and device type is selected to "Atmega 32"
- Then program was written.
- To compile and generate hex file build option in build menu is clicked.
- Hex file generated is the bit file that has to be dumped in a microcontroller; this burning is done using SPI bus provided in microcontroller. SPI bus of microcontroller is connected to the programmer that is used to burn the hex file from PC to microcontroller in AVR Development Board. The GUI used to burn this hex file is SinaProg.

3.2.2 SINAPROG

It is the user interface to burn the hex file to the microcontroller. It has functions like programming the microcontroller, programming fuse bits, chip erase etc. the fig 3.7 shows the GUI of SinaProg.

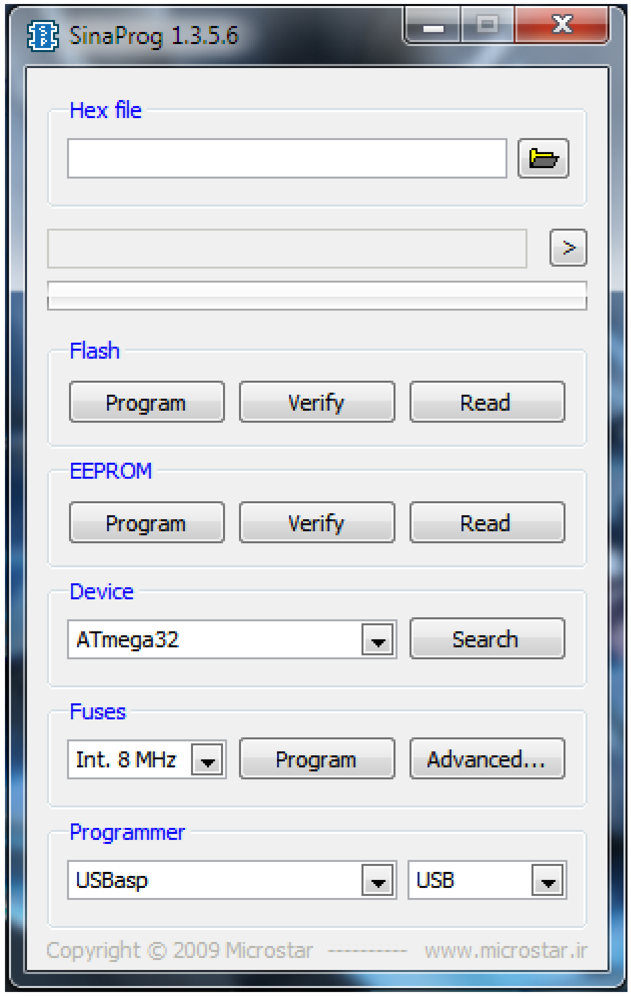


Fig 3.8: GUI of SinaProg

Writing a program

Steps to be followed to create and program a hex file into a microcontroller are

- WINAVR is installed in the PC
- AVR Studio 4.1.7 is installed on PC
- AVR Studio is opened and a new project is created
- While creating a new project, its type is "AVR GCC", debug platform is "AVR Simulator" and device type is selected to "Atmega 32"
- Then program was written
- To compile and generate hex file build option in build menu is clicked

3.3 DRIVERS USED

The drivers used are:

- FT232 usb to serial driver

  An USB to Serial converter is used for interfacing the PC with the Microcontroller serially. It uses the chip called FT232 which requires proper drivers to operate on pc.

- UsbAsp

  The USBAsp driver is required to detect the USBAsp programmer. This is used while programming the microcontroller.

3.4 LIBRARIES USED

3.4.1 PRO GFX GRAPHICS LIBRARY

ProGfx graphics library consists of all the functions required for programming the Graphical LCD. It has functions for drawing lines, circles, rectangles etc. and also it provides functions to display characters or integers.

3.4.2 WINAVR

WinAVR is a library which includes all the header files and .dll files to program an Atmel AVR series of microcontroller. These microcontrollers are of RISC architechture and hosted on Windows platform. C and C++ can also be compiled as it uses GNU-GCC compiler.

WinAVR contains all the tools for developing on the AVR. This includes avr-gcc (compiler), avrdude (programmer), avr-gdb (debugger), and more. WinAVR is widely used by programmers as it is user friendly and simple. Many open source projects can be found online which includes this.

CHAPTER. 4
HARDWARE ASSEMBLY

4.1 GRAPHICAL LCD

PIN Description

PIN	Name	Function	Connection with AVR PIN
1	Vss	Ground	
2	Vcc	+5v Supply in	
3	V0	Contrast Adjust	
4	RS	Instruction/Data Register Select	PD3
5	R/W	READ/WRITE SELECTION	PD6
6	E	ENABLE SIGNAL	PB4
7	DB0	DATA IN/OUT	PC0
8	DB1	DATA IN/OUT	PC1
9	DB2	DATA IN/OUT	PC2
10	DB3	DATA IN/OUT	PC3
11	DB4	DATA IN/OUT	PC4
12	DB5	DATA IN/OUT	PC5
13	DB6	DATA IN/OUT	PC6
14	DB7	DATA IN/OUT	PC7
15	CS1	Chip Select 1	PB0
16	CS2	Chip Select 2	PB1
17	RST	RESET SIGNAL	RESET
18	VEE	NEGATIVE 10V OUT	
19	LED+	LED BACKLIGHT	
20	LED-	LED BACKLIGHT	

Fig 4.1: Pin configuration of Graphical LCD

The 20 linear pins of the graphical LCD are connected to the microcontroller as shown in the figure.

4.2 GPS MODULE

Connect GPS module with the PC using Serial port via MAX232 or equivalent TTL to RS232 converter. Following are the steps to be followed while interfacing GPS Module with PC.

- Install the terminal software called Real Term on the PC.
- In serial terminal setting column the com port has to be selected.
- Set Baud rate to 9600.
- No parity, 8 data bits 1 stop bit.
- connect button is clicked for connection

Then it automatically sends the NMEA protocol strings to the pc which can be shown using any terminal software.

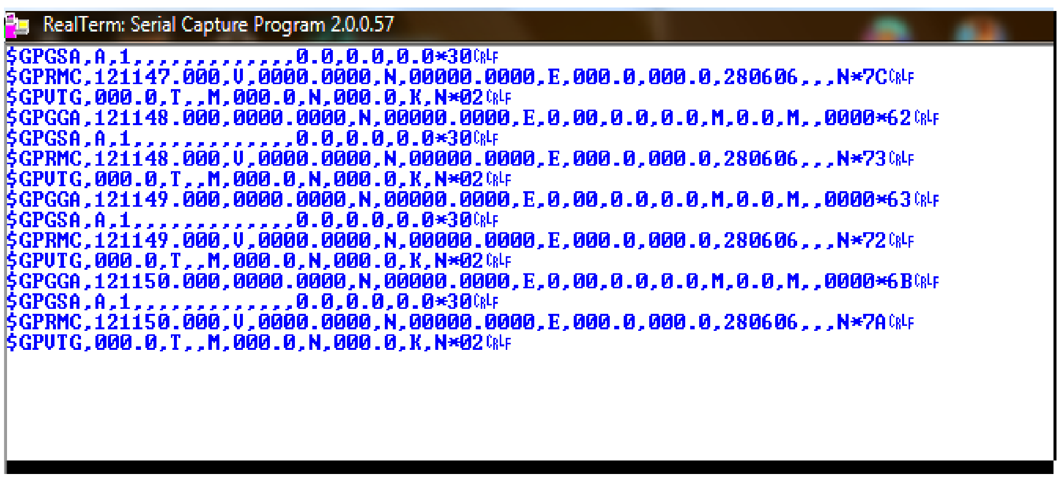


Fig 4.2 : NMEA strings shown on terminal software

The USART interfacing with microcontroller is described below:

Like many microcontrollers AVR also has a dedicated hardware for serial communication this part is called the USART – Universal Synchronous Asynchronous Receiver Transmitter.

This is easy to program. As the serial transmission occurs at speed like 9600 or 19200bps etc it is slower than AVR's CPU speed. The advantage is only message byte has to be put in the data register then other tasks can be done.

The USART of AVR is connected to CPU by the following six regisers:

- UDR – USART data register: this is not one but two registers. When data is read it is stored in receive buffer ant when data is sent it is stored in transmit buffer.
- UCSRA, UCSRB, UCSRC – USART Control and Status Register A,B,C: As the name suggests it is used to configure the USART.
- UBRRH and UBRRL: This is USART baud rate register. It is 16 bit wide so UBRRH is the high byte and UBRRL is the low byte. But by using 'C' language it is directly available as UBRR and the compiler manages to use it as 16bit register.

Through USART the GPS sends the NMEA strings. Those have to be cut down in order to get latitude and longitude up to millisecond accuracy.

Decode of selected position sentences

The NMEA string which provides the current Fix data, or latitude and longitudeincludes GGA. Gps sentences information, and the GSA which provides the Satellite status data.

GGA - essential fix data which provide 3D location and accuracy data.

$GPGGA,123456,7890.123,N,05678.901,E,2,34,5.6,789.0,M,12.3,M,,*45

Where:

GGA	: Global Positioning System Fix Data
123456	: Fix taken at 12:34:56 UTC
7890.123,N	: Latitude 78 deg 90.123' N

05678.901,E : Longitude 56 deg 78.901' E

1:Fix quality: 0 = invalid

1 = GPS fix (SPS)

2 = DGPS fix

3 = PPS fix

4 = Real Time Kinematic

5 = Float RTK

6 = estimated feature

7 = Manual input mode

8 = Simulation mode

08 = Number of satellites from which signal is received.

0.9 : Horizontal dilution of position

545.4, M : Altitude or height (In meters)

46.9, M : Height of geoid (mean sea level) above WGS8 ellipsoid

*47 the checksum data, always begins with *

Of all these data we need only latitude and longitude for our purpose. Those are extracted using string functions.

Distance Calculation:

This uses the ‘**haversine**’ formula to calculate the great-circle distance between two points – that is, the shortest distance over the earth’s surface – giving an ‘as-the-crow-flies’ distance between the points (ignoring any hills, of course!).

Harversine Formula

$$a = \sin^2(\Delta\varphi/2) + \cos(\varphi_1).\cos(\varphi_2).\sin^2(\Delta\lambda/2)$$

$$c = 2.\mathrm{atan2}(\sqrt{a}, \sqrt{(1-a)})$$

$$d = R.c$$

where φ is latitude, λ is longitude, R is earth's radius (mean radius = 6,371km)

The angles need to be in radians to pass to trigonometric functions.

The latitude and longitude of the flag and the player are saved in the microcontroller flash memory and the distance is then calculated using Harversine Formula.

4.3 RESISTIVE TOUCH SCREEN

The inbuilt ADC of AVR MCU is used to get analog values from the resistive touch screen. The ADC is multiplexed with PORTA that means it is shared with the I/O pins of PORTA. It can be operated in single conversion or free running mode. In single conversion mode the ADC does one conversion and then stops. But in free running mode it continuously takes analog values and converts it. It does one conversion and then starts the next conversion immediately after that.

There are the registers which are used in ADC conversion.

- ADMUX: ADC Multiplexer register

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

This register selects the ADC reference values and enables the ADC configuration of PORTA.

- ADCSRA: ADC Status Register

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

It enables the ADC and sets the flags to show the status of the conversion.

- ADCH: ADC data register

Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	–	–	ADC9	ADC8	ADCH
	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

Bit	15	14	13	12	11	10	9	8	
	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADCH
	ADC1	ADC0	–	–	–	–	–	–	ADCL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

When ADC conversion is complete the data is stored in these two registers. It uses different channels.

- SFIOR: Special FunctionIO register

Bit	7	6	5	4	3	2	1	0
	ADTS2	ADTS1	ADTS0	–	ACME	PUD	PSR2	PSR10
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

It selects different modes of ADC like free running mode or comparator mode etc. depending on the values given to the bits

4.4 ZIGBEE: The Transceiver

The Zigbee module used in this project is a XBee 802.15.4. It is used for embedded solutions providing wireless end-point connectivity to devices. This is an ideal module for robots to PC or robots to robots communication. This module can give range of 30 meters indoor or 100 meters outdoor. This XBee wireless device can be directly connected to the serial port (at 3.3V level) of your microcontroller. By using a logic level translator it can also be interfaced to 5V logic (TTL) devices having serial interface. This module supports data rates of up to 115kbps.

The Zigbee module used in this project is a XBee 802.15.4. It is used for embedded solutions providing wireless end-point connectivity to devices. This is perfect for robot to pc communication. In indoor condition its range is up to 100 meter. Microcontroller can be interfaced with ZigBee directly by serial port.The Zigbee module communicates through USART to the microcontroller. The receiver Zigbee which has same PAN Id will receive the data automatically. If the data is valid then it will be transmitted to the server again through serial communication and will be stored against the user name.The initial set up is done by X-ctu, a GUI provided by digi.com. The com port is opened by X-CTU and then in Modem Configuration tab the channel Id and PAN Id is set for both transmitter and receiver end Zigbee. Both the Ids should be same for both the Zigbee. Then only the transmission is possible.

CHAPTER. 5

DATABASE MANAGEMENT

The entire Graphical User Interface on PC is designed using Microsoft Visual Studio 10. It consist of five Windows based Forms and a Database. A Window Form consists of a user controls that makes application highly interactive and database stores the data.

The scores transmitted by the Zigbee is received at the database side. Then it is fed to the PC through serial communication. The score is saved against the user name by admin.

Admin can add new members and update their scores. The user can only change their own information and see others score. But he cannot change the score or upgrade it.

FORM CONNECTIVITY

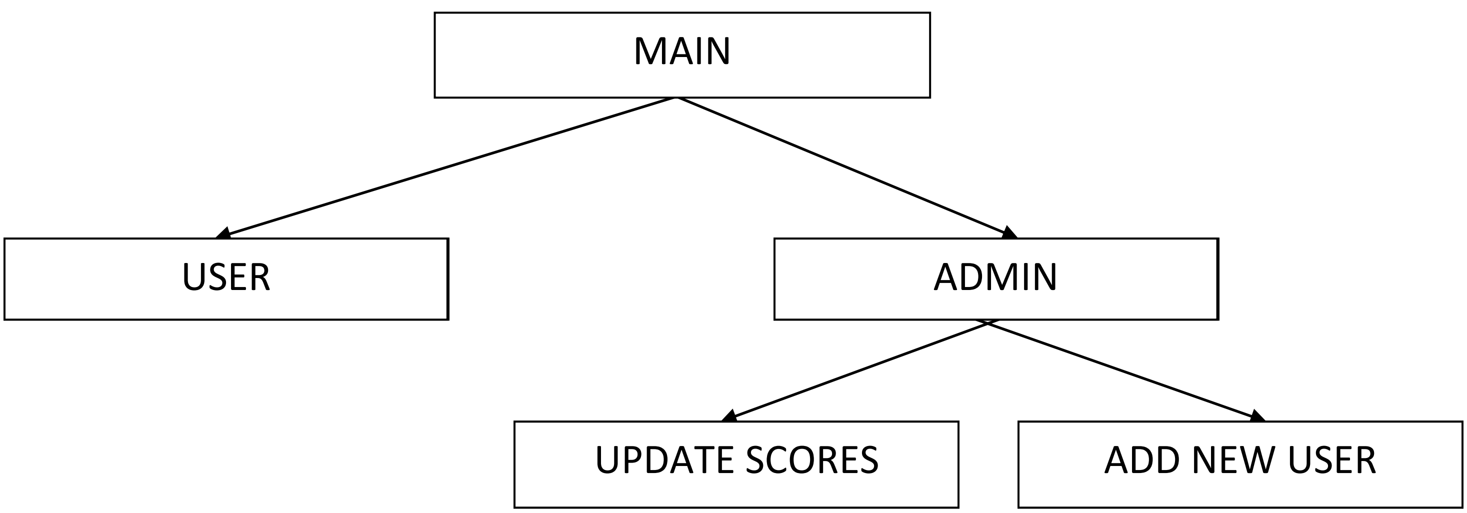


Fig 5.1 : It shows how the forms are connected to each others.

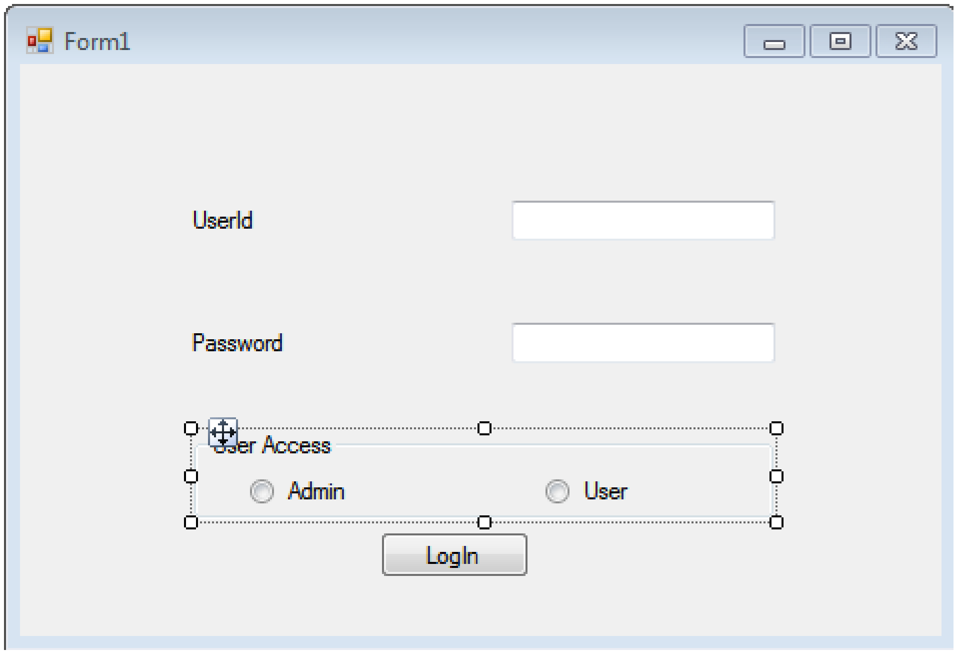


Fig 5.2 : Snapshot of the forms.

Tables used in Database

Table 1: Members

Attributes	Data Types	Property
Name	nvarchar(20)	Not Null
User Id	nvarchar(20)	Primary Key
Password	nvarchar(20)	Not Null
Phone	nvarchar(13)	Not Null, Unique
Address	nvarchar(50)	Not Null

Table 2: Score

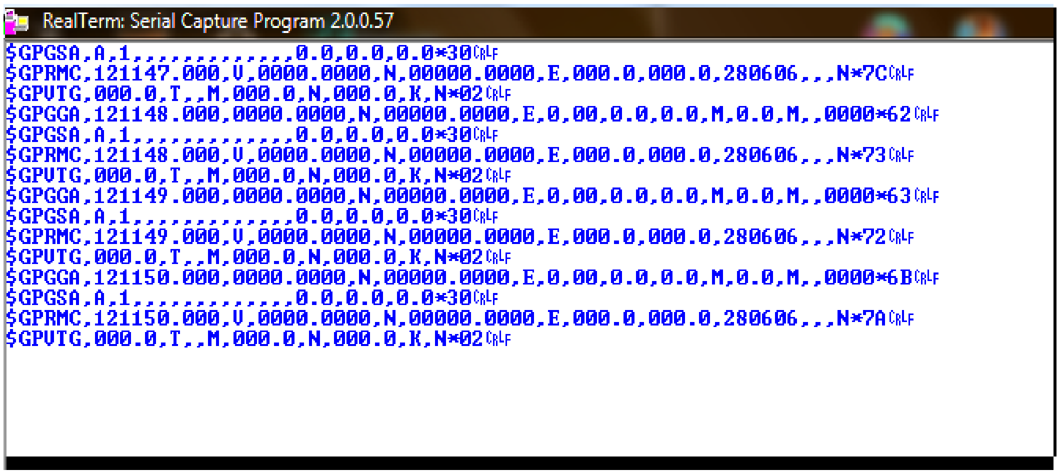
Attributes	Data Types	Property
User Id	nvarchar(20)	Unique, Not Null
Date	Date	Not Null
Score	Numeric	Not Null

CHAPTER. 6

RESULTS

6.1 GPS COORDINATES ON PC

First the gps was checked on PC to confirm whether it works properly or not. It should show the NMEA strings on the terminal software.



RealTerm: Serial Capture Program 2.0.0.57

```
$GPGSA,A,1,,,,,,,,,,,,,0.0,0.0,0.0*30
$GPRMC,121147.000,V,0000.0000,N,00000.0000,E,000.0,000.0,280606,,,N*7C
$GPVTG,000.0,T,,M,000.0,N,000.0,K,N*02
$GPGGA,121148.000,0000.0000,N,00000.0000,E,0,00,0.0,0.0,M,0.0,M,,0000*62
$GPGSA,A,1,,,,,,,,,,,,,0.0,0.0,0.0*30
$GPRMC,121148.000,V,0000.0000,N,00000.0000,E,000.0,000.0,280606,,,N*73
$GPVTG,000.0,T,,M,000.0,N,000.0,K,N*02
$GPGGA,121149.000,0000.0000,N,00000.0000,E,0,00,0.0,0.0,M,0.0,M,,0000*63
$GPGSA,A,1,,,,,,,,,,,,,0.0,0.0,0.0*30
$GPRMC,121149.000,V,0000.0000,N,00000.0000,E,000.0,000.0,280606,,,N*72
$GPVTG,000.0,T,,M,000.0,N,000.0,K,N*02
$GPGGA,121150.000,0000.0000,N,00000.0000,E,0,00,0.0,0.0,M,0.0,M,,0000*6B
$GPGSA,A,1,,,,,,,,,,,,,0.0,0.0,0.0*30
$GPRMC,121150.000,V,0000.0000,N,00000.0000,E,000.0,000.0,280606,,,N*7A
$GPVTG,000.0,T,,M,000.0,N,000.0,K,N*02
```

Fig 6.1 : NMEA protocol strings shown on RealTerm terminal software.

As the GPS is kept in indoor conditions so the values are default values which are not the coordinates of the place.

6.2 GPS COORDINATES ON ALPHANUMERIC LCD

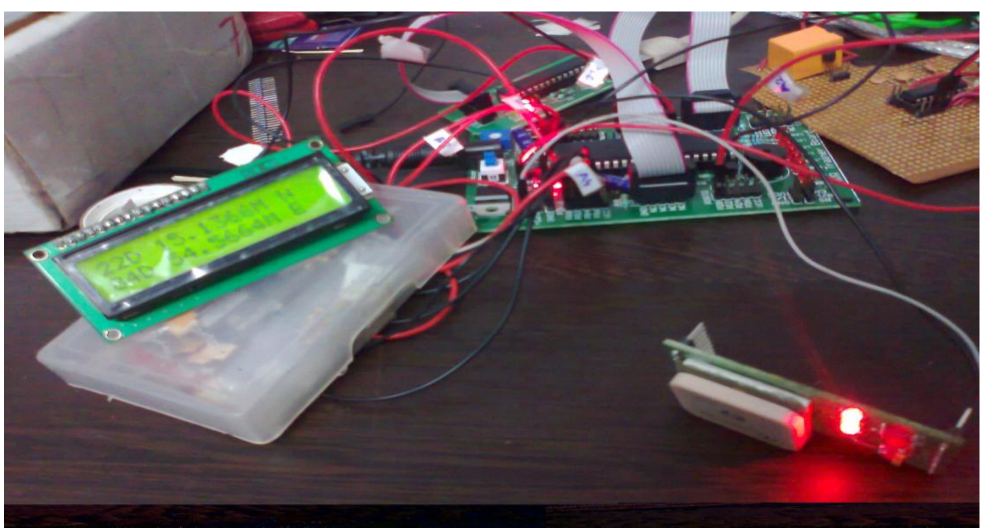


Fig 6.2 : GPS coordinates shown on alphanumeric LCD

6.3 GPS COORDINATES ON GRAPHICAL LCD

The GPS coordinates are shown on GLCD for better visualization.

Fig 6.3 : GPS coordinates of Hall-8 shown on Graphical LCD

Latitude	Longitude
22.25232	84.90990

Fig 6.4 : GPS coordinates of hall-8 taken from Google Maps

6.4 TOUCH SCREEN CALIBRATION

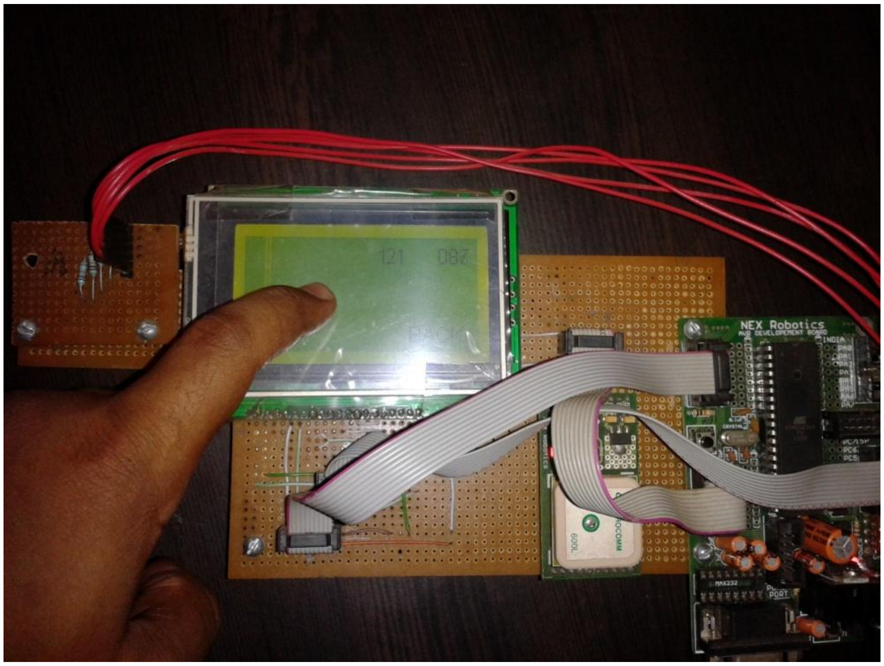


Fig 6.5 : The X and Y coordinates of the touch screen are shown on the GLCD.

6.5 KEY BOARD ON GRAPHICAL LCD

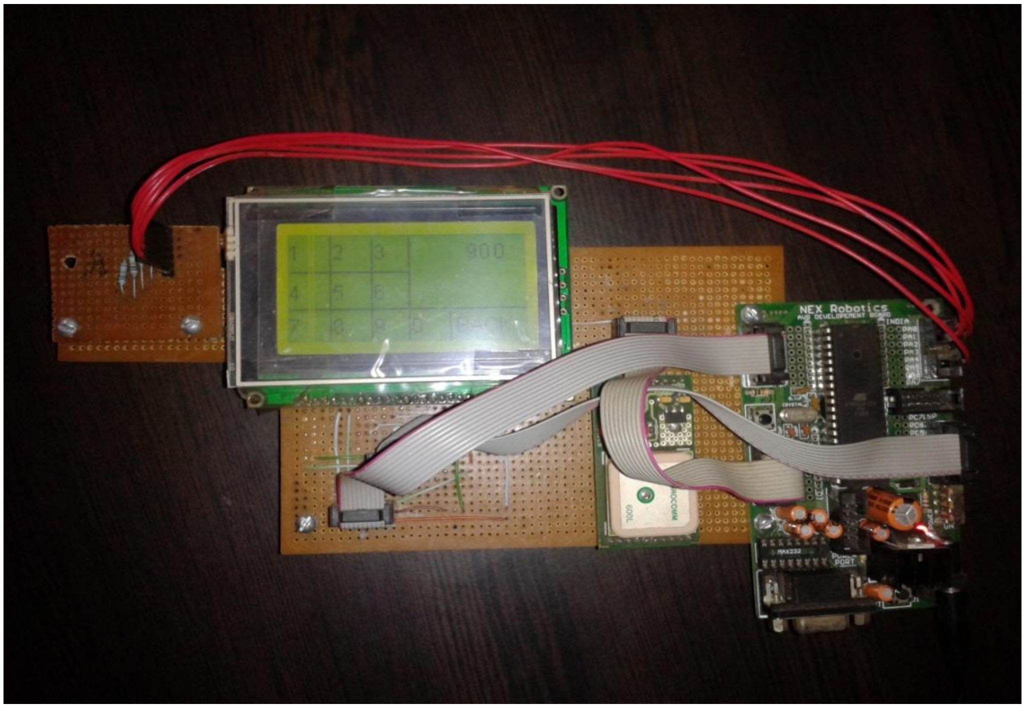


Fig 6.6 : The key board for entering the scores is shown on the GLD.

6.6 ZIGBEE MODEM CONFIGURATION

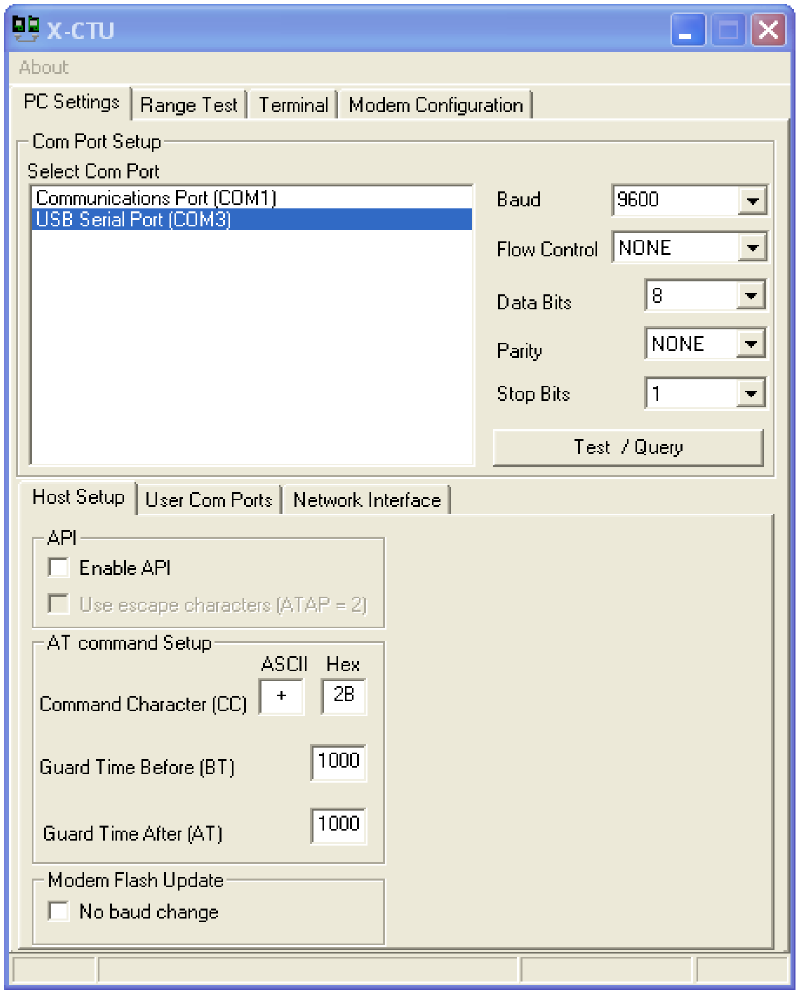


Fig 6.7 : Opening com port through X-CTU.

The zigbee wireless modem is opened through X-CTU and then the PAN Id and channel number is assigned. They are same for both transmitter and receiver.

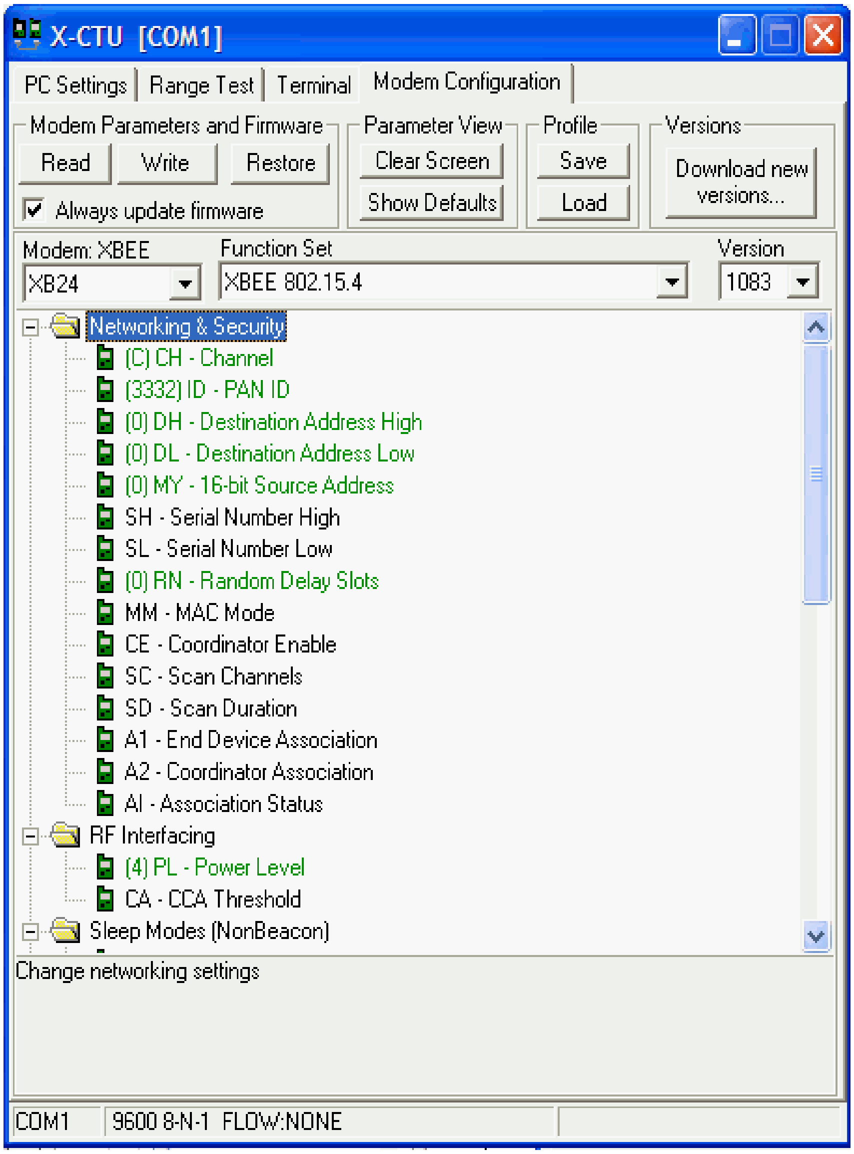


Fig 6.8 : Configuring the Zigbee modem

6.7 FINAL ASSEMBLY

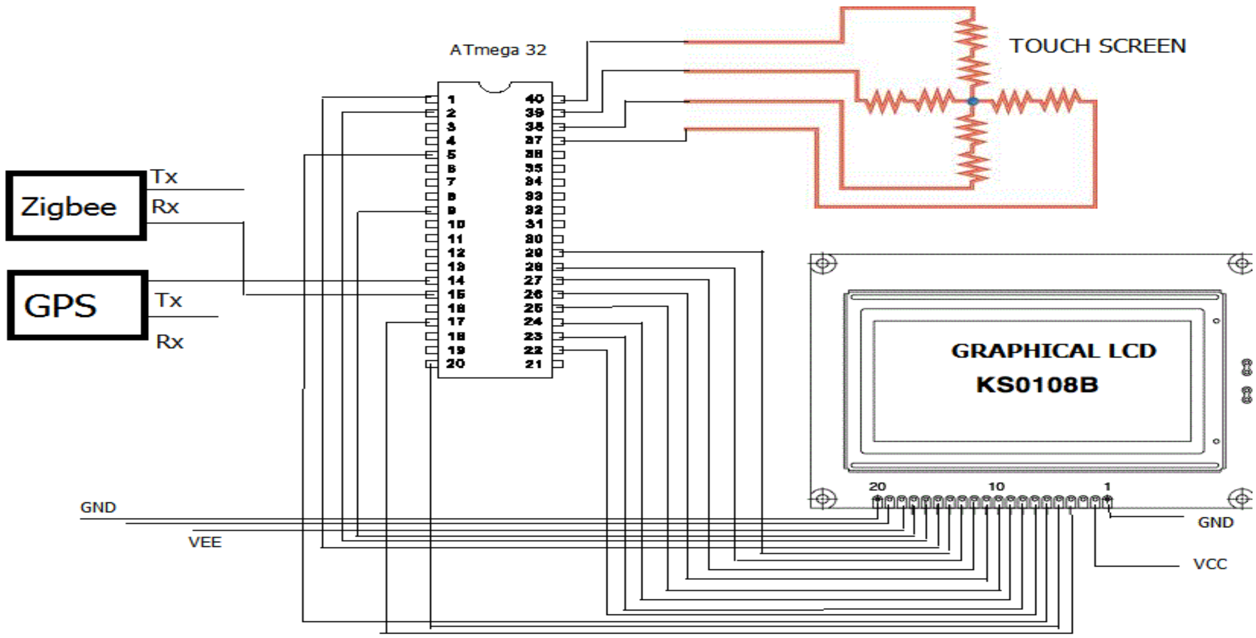


Fig6.9: Complete Circuit Diagram

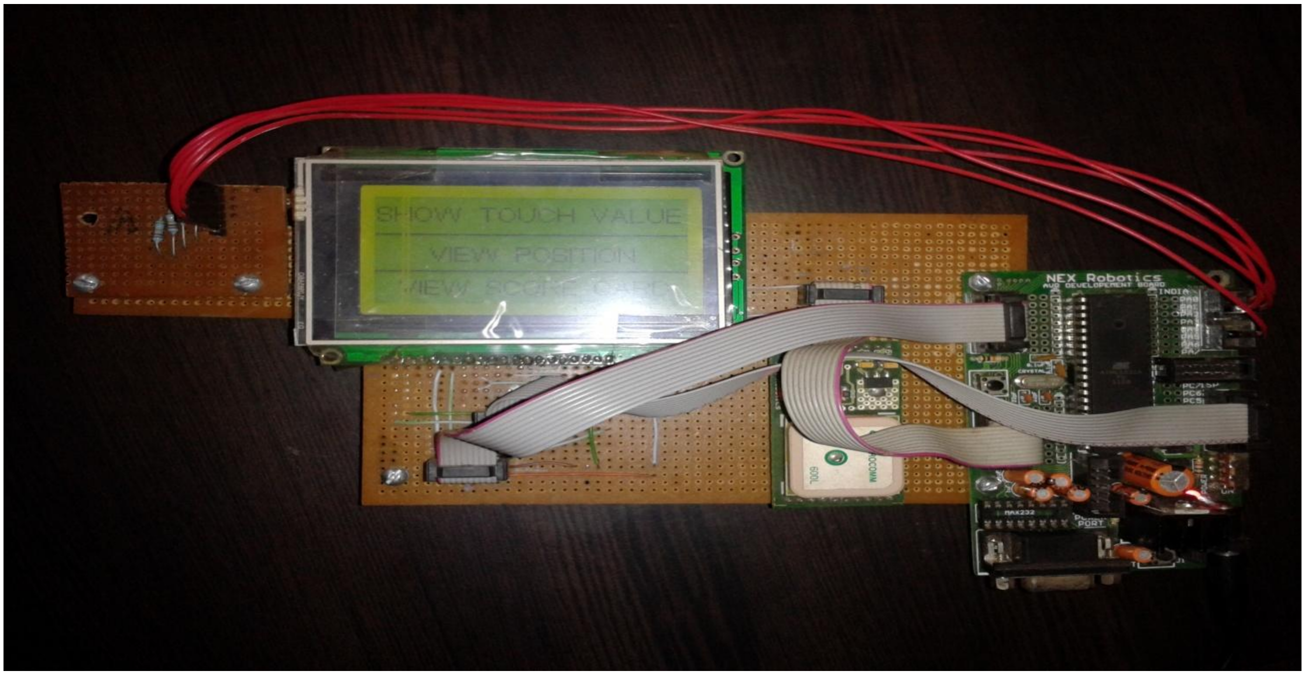


Fig 6.10: The first screen of the final assembly

All the functions are integrated in the final assembly. Those functions include:

- Viewing the X and Y coordinates of the touch screen.
- Determining the coordinates of the point on earth.
- Calculating the distance between two points on earth and entering scores.

CHAPTER. 7

CONLUSION

The distance was calculated manually using coordinates obtained from the GPS Modem output was well in accurate to the original distance. Hence the Haversine Relation obtained above is the best suited algorithm to achieve this purpose. On the other hand GUI and Database was created and score as well as player info could be stored manually.

REFERENCES

1 http://www.nexrobotics.com/images/downloads/ATMEGA32%20Developement%20Board.pdf

2 http://www.nexrobotics.com/images/downloads/avr%20usb%20programmer%20manualv2.pdf

3 http://www.nexrobotics.com/images/downloads/GPS%20V2%20TTL%20module.pdf

4 www.nxp.com/documents/application_note/AN10675.pdf

5 http://extremeelectronics.co.in/avr-tutorials/interfacing-ks0108-based-128x64-graphical-lcd-with-avr-mcu/

6 http://progfx.org/version1.01/

7 http://sodoityourself.com/max232-serial-level-converter/