**TABLE OF CONTENTS**

[ABSTRACT 3](#_Toc104909525)

[LIST OF FIGURES 4](#_Toc104909526)

[1 INTRODUCTION 5](#_Toc104909527)

[1.1 Introduction 5](#_Toc104909528)

[1.2 Project Overview 5](#_Toc104909529)

[1.3 Problem Statement 6](#_Toc104909530)

[1.4 Aim and Objective 6](#_Toc104909531)

[1.5 Thesis: 7](#_Toc104909532)

[2 LITERATURE SURVEY 8](#_Toc104909533)

[3 HARDWARE DESCRIPTION 9](#_Toc104909534)

[3.1 Introduction 9](#_Toc104909535)

[3.2 Microcontroller 10](#_Toc104909536)

[3.2.1 Introduction 10](#_Toc104909537)

[3.2.2 Peripheral Features 11](#_Toc104909538)

[3.2.3 CMOS Technology 11](#_Toc104909539)

[3.2.4 Blocks of PIC Microcontroller 12](#_Toc104909540)

[3.2.5 Pin description 13](#_Toc104909541)

[3.2.6 Memory organization 15](#_Toc104909542)

[3.2.7 Applications 15](#_Toc104909543)

[3.3 Regulated Power Supply 16](#_Toc104909544)

[3.3.1 Introduction 16](#_Toc104909545)

[3.3.2 Block Diagram 17](#_Toc104909546)

[3.4 LoRa (Long Range) module 17](#_Toc104909547)

[3.4.1 LoRa SX1276 18](#_Toc104909548)

[3.4.2 LoRa SX1276 Pin Description 19](#_Toc104909549)

[3.4.3 Features of LoRa SX1276 19](#_Toc104909550)

[3.5 Buzzer 19](#_Toc104909551)

[3.6 HC-05 Bluetooth 20](#_Toc104909552)

[3.6.1 Hardware Features 21](#_Toc104909553)

[3.6.2 Software Features 21](#_Toc104909554)

[3.6.3 Pin Description 21](#_Toc104909555)

[3.7 LCD Display 22](#_Toc104909556)

[3.7.1 LCD Background 22](#_Toc104909557)

[3.7.2 Basic 16x 2 Characters LCD 23](#_Toc104909558)

[3.7.3 Pin description: 23](#_Toc104909559)

[4 SOFTWARE DESCRIPTION 25](#_Toc104909560)

[4.1 PIC Compiler 25](#_Toc104909561)

[4.2 Proteus 26](#_Toc104909562)

[4.3 Procedural steps for compilation, simulation and dumping 26](#_Toc104909563)

[4.3.1 Compilation and simulation steps 26](#_Toc104909564)

[4.3.2 Dumping steps 27](#_Toc104909565)

[5 METHODOLOGY 34](#_Toc104909566)

[6 ADVANTAGES AND DISADVANTAGES 38](#_Toc104909567)

[6.1 Advantages 38](#_Toc104909568)

[6.2 Disadvantages 38](#_Toc104909569)

[6.3 Applications 38](#_Toc104909570)

[7 RESULTS 39](#_Toc104909571)

[7.1 Transmitter Section 39](#_Toc104909572)

[7.2 Receiver Section 40](#_Toc104909573)

[7.3 BT Voice Control App 40](#_Toc104909574)

[7.4 Result 41](#_Toc104909575)

[8 CONCLUSION AND FUTURE SCOPE 42](#_Toc104909576)

[8.1 Conclusion 42](#_Toc104909577)

[8.2 Future Scope 42](#_Toc104909578)

[REFERENCES 43](#_Toc104909579)

# ABSTRACT

The LoRa is noticed as the most widespread wireless technology in terms of open structure and unlicensed band. LoRa stands for Long Range Radio and is mainly targeted for M2M and IoT networks. For the purpose of technology supporting and development there was incorporated LoRa Alliance. Transceivers use different types of modulations to send and receive data. The network coverage and data capacity are highly dependent on the frequency and type of modulation used. By using Chirp spread spectrum modulation, data can be sent to long distances. LoRa provides long-range, low-power consumption, a low data rate, and secure data transmission.CSS uses wideband linear frequency modulated chirp pulses to encode information. LoRa can be used with public, private, or hybrid networks to achieve a greater range than cellular networks. LoRa technology can easily integrate with existing networks and enables low-cost, battery-operated Internet of Things (IoT) applications. The main intention of the project is to convert speech to text using a LoRa module. LoRa module is available as transmitter receiver pair. The sender uses a Mobile phone to contact the receiver. At the receiver end, the speech will get converted into text and get displayed on the mobile screen to which the sender has contacted. Initially while transmitting the data, the distance between the sender and the receiver is maintained at 3km. The receiver is able to decode the received speech properly. This project is helpful in situations where the receiver end person would be able to simply see the text instead of talking.

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **FIGURE NO.** | **NAME OF THE FIGURE** | **PAGE NO.** |
| 3.1 | Transmitter Block Diagram |  |
| 3.2 | Receiver Block Diagram |  |
| 3.3 | Pin diagram of PIC16F72 |  |
| 3.4 | Block Diagram of Regulated Power Supply |  |
| 3.5 | LoRa SX1276 |  |
| 3.6 | Picture of Buzzer |  |
| 3.7 | HC-05 Bluetooth Module |  |
| 3.8 | 16x2 LCD Pin Diagram |  |
| 4.1 | Program Dumper Window |  |
| 4.2 | Checking Before Dumping into Microcontroller |  |
| 4.3 | Connecting Dumper to Microcontroller |  |
| 4.4 | Dumper Recognition to Microcontroller |  |
| 4.5 | Program Importing into Microcontroller |  |
| 4.6 | After Dumping Program into Microcontroller |  |
| 5.1 | Circuit Diagram of Transmitter Section |  |
| 5.2 | Circuit Diagram of Receiver Section |  |
| 7.1 | Bluetooth Module |  |
| 7.2 | LoRa Module |  |
| 7.3 | Receiver Circuit |  |
| 7.4 | BT Voice Control App |  |
| 7.5 | Speech Signal given to Mobile Unit |  |
| 7.6 | Text on LCD Display |  |

# INTRODUCTION

## Introduction

The project aims at implementing a LoRa based speech-to-text conversion using Bluetooth, PIC microcontroller, Mobile phone, LCD display.

This project contains onboard computer, which consists of number of input and output ports. These onboard computers are commonly termed as micro controllers. The input and output port of the controller are interfaced with different input and output modules depending on the requirements. In other words micro controller acts as a communication medium for all the modules involved in the project.

In this project, there are two sections such as transmitter and receiver. The user can give the voice through Bluetooth application using Mobile unit. This voice is transmited by Lora module via Bluetooth. In receiver section, Lora module receives the transmitted voice and fed as same to the PIC microcontroller. Microcontroller will convert that voice into text and displays on LCD module along with Buzzer.

The major features of this project are:

* Voice to text conversion.
* Using LORA wireless technology.
* Visible alerts using LCD display.
* Audible alerts using Buzzer.

## Project Overview

An embedded system is a combination of software and hardware to perform a dedicated task*.* Some of the main devices used in embedded products are Microprocessors and Microcontrollers.

Microprocessors are commonly referred to as general purpose processors as they simply accept the inputs, process it and give the output. In contrast, a microcontroller not only accepts the data as inputs but also manipulates it, interfaces the data with various devices, controls the data and thus finally gives the result.

The “Speech to text using LoRa” using PIC16F72 microcontroller is an exclusive project which is used to converts the user voice into text wirelessly using LoRa module. This system uses Bluetooth application to give the voice input by using mobile.

## Problem Statement

The problem with the existing technologies like Wi-Fi, Bluetooth and GSM, when used for information transfer, is low range and high power consumption. LoRa overcomes such challenges.

## Aim and Objective

The project aims in implementing a LoRa based speech to text conversion using PIC controller, Bluetooth module, LCD display, Mobile phone.

So in order to achieve this LoRa module is used instead of GSM and Wi-Fi.

The objectives of the project are:

* To achieve Long range wireless communication.
* To communicate with low power consumption.
* To obtain efficient transmission.

## Thesis:

The thesis explains the implementation of “**Speech to text using LoRa”** using PIC16F72 microcontroller. The organization of the thesis is explained here with:

**Chapter 1** Presents introduction to the overall thesis and the overview of the project. In the project overview, a brief introduction of **Speech to text using LoRa** and its applications are discussed.

**Chapter 2** Presents the Literature survey of the project.

**Chapter 3** Presents the hardware description. It deals with the block diagram of the project and explains the purpose of each block.

**Chapter 4**  Presents the software description. It explains the implementation of the project using PIC C Compiler software.

**Chapter 5** Presents the project description

**Chapter 6** Presents the advantages, disadvantages and applications of the project.

**Chapter 7** Presents the results, conclusion and future scope of the project.

**Chapter 8** Presents the conclusion and future scope of the project.

# LITERATURE SURVEY

LORA is the new communication technology under the Low Power Wide Area Network (LPWAN). It emphasizes on the long-range communication with the high receiving sensitivity ability which allows it to work under the noise interference or noise floor

Effectively and concluded that LoRa technology has the advantages of working under noise interference and allow long-distance communication thanks to its modulation technique. In this paper, the performance and coverage result of the LoRa indoor and outdoor deployment is presented using a selected combination of Spreading Factor and Bandwidth setting.[1]

LoRa is low power wide area wireless network (LPWAN) protocol for Internet of Things (IoTs) applications. Effective cost, long range and energy efficiency of LPWANs make them most suitable candidates for smart city applications and concluded that This paper focused on one of the most prominent LPWAN technology: LoRa; Analyzed the performance of LoRa, based on its three basic parameters: code rate, spreading factor and bandwidth.[2]

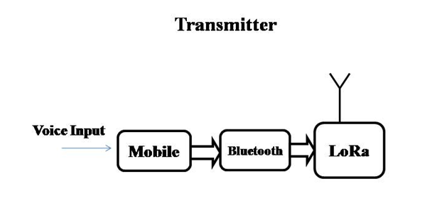
Long range (LoRa) is emerging modulation technique in the family of low power wide area network (LPWAN) which enables long distance and power efficient wireless communication. LoRa is best elegance for communication of internet of things (IoT) devices. By keeping simple network structures and easy management LoRa ensures good connectivity in outdoor IoT applications and concluded that LoRa technology has the potential of longer distance communication than normally used short range IoT protocols. Major benefit is the robustness to interferences by simply changing the SF parameter, the communication distance is greatly increased to the affliction of the data transfer rate. Hence LoRa transceivers provide best choice for IoT application.[3]

The LoRa technology addresses these needs of a battery operated embedded device. The LoRa technology is a long range low power technology. This paper discuses about the advantages of LoRa over the existing technologies used in IoT and concluded that The LoRa technology is preferred due to various advantages such as the battery lifetime, the long range, the security, robustness to interferences and more. The technology is a package in itself. [4]

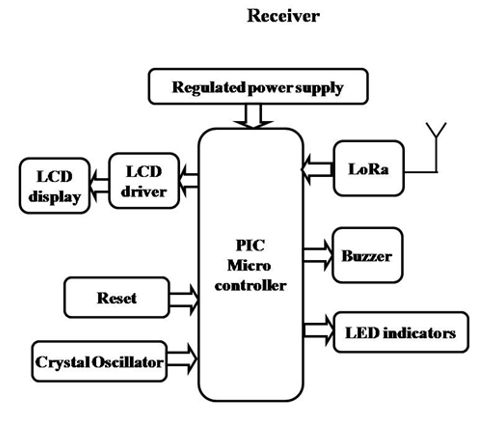
# HARDWARE DESCRIPTION

## Introduction

In this chapter the block diagram of the project and design aspect of independent modules are discussed. Block diagram is shown in fig: 3.1:



**Fig 3.1** Transmitter block diagram



**Fig 3.2** Receiver block diagram

The main blocks of this project are:

* Regulated Power Supply.
* PIC Microcontroller.
* LED indicators.
* Reset.
* LoRa.
* Crystal oscillator.
* LCD display.
* Bluetooth.
* Buzzer.

## Microcontroller

A Microcontroller is a programmable digital processor with necessary peripherals. Both microcontrollers and microprocessors are complex sequential digital circuits meant to carry out job according to the program / instructions. Sometimes analog input/output interface makes a part of microcontroller circuit of mixed mode (both analog and digital nature). It can be considered as a smaller computer with on-chip RAM, ROM, I/O ports.

Example: Motorola’s 6811, Intel’s 8051, Zilog’s Z8 and PIC 16X

### Introduction

• High performance RISC CPU

• Only 35 single word instructions to learn

• All single cycle instructions except for program branches which are two-cycle

• Operating speed: DC - 20 MHz clock input DC - 200 ns instruction cycle

• 2K x 14 words of Program Memory

• 128 x 8 bytes of Data Memory (RAM)

• Pin out compatible to the PIC16C72/72A and PIC16F872

• Interrupt capability

• Eight level deep hardware stack

• Direct, Indirect and Relative Addressing modes

### Peripheral Features

• Timer0: 8-bit timer/counter with 8-bit prescaler

• Timer1: 16-bit timer/counter with prescaler, can be incremented during SLEEP via

external crystal/clock

• Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler

• Capture, Compare, PWM(CCP) module

- Capture is 16-bit, max resolution is 12.5 ns

- Compare is 16-bit, max resolution is 200 ns

- PWM max resolution is 10-bit

• 8-bit, 5-channel Analog-to-Digital converter

• Synchronous Serial Port (SSP) with SPI (Master mode) and I2C (Slave)

• Heat sink/Source Current: 25 mA

• Brown-out detection circuitry for Brown-out Reset (BOR)

### CMOS Technology

• Low power, high speed CMOS FLASH technology

• Fully static design

• Wide operating voltage range: 2.0V to 5.5V

• Industrial temperature range

• Low power consumption:

- < 0.6 mA typical @ 3V, 4 MHz

- 20 μA typical @ 3V, 32 kHz

### Blocks of PIC Microcontroller

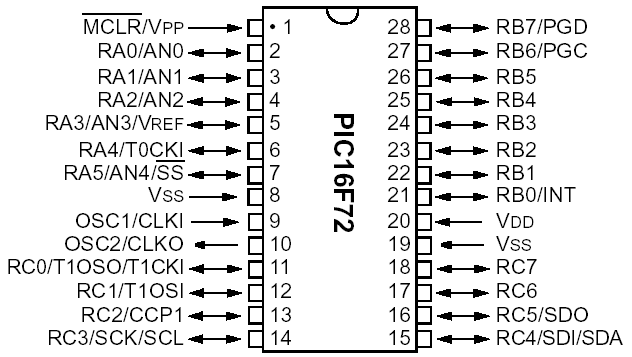
**Program memory** (FLASH) is used for storing a written program.   
Since memory made in FLASH technology can be programmed and cleared more than once, it makes this microcontroller suitable for device development.   
  
**EEPROM** - data memory that needs to be saved when there is no supply.  
It is usually used for storing important data that must not be lost if power supply suddenly stops. For instance, one such data is an assigned temperature in temperature regulators. If during a loss of power supply this data was lost, we would have to make the adjustment once again upon return of supply. Thus our device looses on self-reliance.   
  
**RAM**- Data memory used by a program during its execution. In RAM are stored all inter-results or temporary data during run-time. PORTS are physical connections between the microcontroller and the outside world. PIC16F72 has 22 I/O.

**FREE-RUN TIMER** is an 8-bit register inside a microcontroller that works independently of the program. On every fourth clock of the oscillator it increments its value until it reaches the maximum (255), and then it starts counting over again from zero. As we know the exact timing between each two increments of the timer contents, timer can be used for measuring time which is very useful with some devices.   
  
**Crystal oscillator:** The crystal oscillator speed that can be connected to the PIC microcontroller range from DC to 20Mhz. Using the CCS C compiler normally 20Mhz oscillator will be used and the price is very cheap. The 20 MHz crystal oscillator should be connected with about 22pF capacitor. Please refer to my circuit schematic.

There are 5 input/output ports on PIC microcontroller namely port A, port B, port C, port D and port E. Each port has different function. Most of them can be used as I/O port.

### Pin description

PIC16F72 has a total of 28 pins. It is most frequently found in a DIP28 type of case but can also be found in SMD case which is smaller from a DIP. DIP is an abbreviation for Dual In Package. SMD is an abbreviation for Surface Mount Devices suggesting that holes for pins to go through when mounting aren't necessary in soldering this type of a component.



**Fig 3.3** Pin diagram of PIC16F72

Pins on PIC16F72 microcontroller have the following meaning:  
  
There are 28 pins on PIC16F72. Most of them can be used as an IO pin. Others are already for specific functions. These are the pin functions:

1. MCLR – to reset the PIC  
2. RA0 – port A pin 0  
3. RA1 – port A pin 1  
4. RA2 – port A pin 2  
5. RA3 – port A pin 3  
6. RA4 – port A pin 4  
7. RA5 – port A pin 5  
8. VSS – ground  
9. OSC1 – connect to oscillator  
10. OSC2 – connect to oscillator  
11. RC0 – port C pin 0 VDD – power supply  
12. RC1 – port C pin 1  
13. RC2 – port C pin 2  
14. RC3 – port C pin 3  
15. RC4 - port C pin 4  
16. RC5 - port C pin 5  
17. RC6 - port C pin 6  
18. RC7 - port C pin 7  
19. VSS - ground  
20. VDD – power supply  
21. RB0 - port B pin 0  
22. RB1 - port B pin 1  
23. RB2 - port B pin 2  
24. RB3 - port B pin 3  
25. RB4 - port B pin 4  
26. RB5 - port B pin 5  
27. RB6 - port B pin 6  
28. RB7 - port B pin 7  
  
By utilizing all of this pin so many application can be done such as:  
1. LCD – connect to Port B pin.  
2. LED – connect to any pin declared as output.  
3. Relay and Motor - connect to any pin declared as output.  
4. External EEPROM – connect to I2C interface pin – RC3 and RC4 (SCL and SDA)   
5. LDR, Potentiometer and sensor – connect to analogue input pin such as RA0.  
6. GSM modem dial up modem – connect to RC6 and RC7 – the serial communication interface using RS232 protocol.

### Memory organization

PIC16F72 has two separate memory blocks, one for data and the other for program. EEPROM memory with GPR and SFR registers in RAM memory make up the data block, while FLASH memory makes up the program block.

**Program memory**

Program memory has been carried out in FLASH technology which makes it possible to program a microcontroller many times before it's installed into a device, and even after its installment if eventual changes in program or process parameters should occur. The size of program memory is 1024 locations with 14 bits width where locations zero and four are reserved for reset and interrupt vector.

**Data memory**

Data memory consists of EEPROM and RAM memories. EEPROM memory consists of 256 eight bit locations whose contents are not lost during loosing of power supply. EEPROM is not directly addressable, but is accessed indirectly through EEADR and EEDATA registers. As EEPROM memory usually serves for storing important parameters (for example, of a given temperature in temperature regulators) , there is a strict procedure for writing in EEPROM which must be followed in order to avoid accidental writing. RAM memory for data occupies space on a memory map from location 0x0C to 0x4F which comes to 68 locations. Locations of RAM memory are also called GPR registers which is an abbreviation for *General Purpose Registers*. GPR registers can be accessed regardless of which bank is selected at the moment.

### Applications

PIC16F72 perfectly fits many uses, from automotive industries and controlling home appliances to industrial instruments, remote sensors, electrical door locks and safety devices. It is also ideal for smart cards as well as for battery supplied devices because of its low consumption.

EEPROM memory makes it easier to apply microcontrollers to devices where permanent storage of various parameters is needed (codes for transmitters, motor speed, receiver frequencies, etc.). Low cost, low consumption, easy handling and flexibility make PIC16F72 applicable even in areas where microcontrollers had not previously been considered (example: timer functions, interface replacement in larger systems, coprocessor applications, etc.).

In System Programmability of this chip (along with using only two pins in data transfer) makes possible the flexibility of a product, after assembling and testing have been completed. This capability can be used to create assembly-line production, to store calibration data available only after final testing, or it can be used to improve programs on finished products.

## Regulated Power Supply

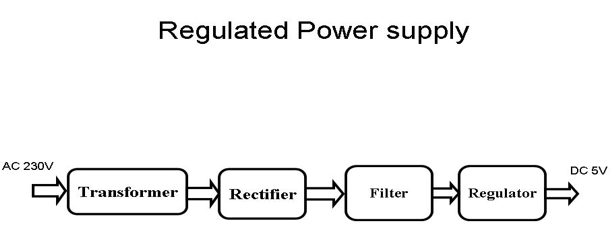
### Introduction

Power supply is a supply of electrical power. A device or system that supplies electrical or other types of energy to an output load or group of loads is called a power supply unit or PSU. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others.

A power supply may include a power distribution system as well as primary or secondary sources of energy such as

* Conversion of one form of electrical power to another desired form and voltage, typically involving converting AC line voltage to a well-regulated lower-voltage DC for electronic devices. Low voltage, low power DC power supply units are commonly integrated with the devices they supply, such as computers and household electronics.
* Batteries.
* Chemical fuel cells and other forms of energy storage systems.
* Solar power.
* Generators or alternators.

### Block Diagram

****

**Fig 3.4** Block Diagram of Regulated Power Supply

## LoRa (Long Range) module

LoRa is the physical layer or the wireless modulation utilized to create the long range communication link. Many legacy wireless systems use frequency shifting keying (FSK) modulation as the physical layer because it is a very efficient modulation for achieving low power. LoRa® is based on chirp spread spectrum modulation, which maintains the same low power characteristics as FSK modulation but significantly increases the communication range. Chirp spread spectrum has been used in military and space communication for decades due to the long communication distances that can be achieved and robustness to interference, but LoRa® is the first low cost implementation for commercial usage.

The advantage of LoRa is in the technology’s long range capability. A single gateway or base station using LoRa can cover wide range. Range highly depends on the environment or obstructions in a given location, but LoRa and LoRaWAN have a link budget greater than any other standardized communication technology. The link budget, typically given in decibels (dB), is the primary factor in determining the range in a given environment. With a minimal amount of infrastructure, entire countries can easily be covered.

One technology cannot serve all of the projected applications and volumes for IoT. Wi-Fi and BTLE are widely adopted standards and serve the applications related to communicating personal devices quite well. Cellular technology is a great fit for applications that need high data throughput and have a power source. LPWAN offers multi-year battery lifetime and is designed for sensors and applications that need to send small amounts of data over long distances a few times per hour from varying environments

The most critical factors in a LPWAN are:

• Network architecture

• Communication range

• Battery lifetime or low power

• Robustness to interference

• Network capacity (maximum number of nodes in a network)

• Network security

• One-way vs two-way communication

### LoRa SX1276

DRF1276DM is a type of long range low data rate data radio modem based on SX1276 from Semtech.

The module can work in two different modes: Standard mode and Star network mode. In the standard mode DRF1276DM acts as transparent data radio modem which it communicates with the host at the preset data format without encoding/decoding needed. In star network mode, one module will be the configured to the central module and other modules are set to node modules.

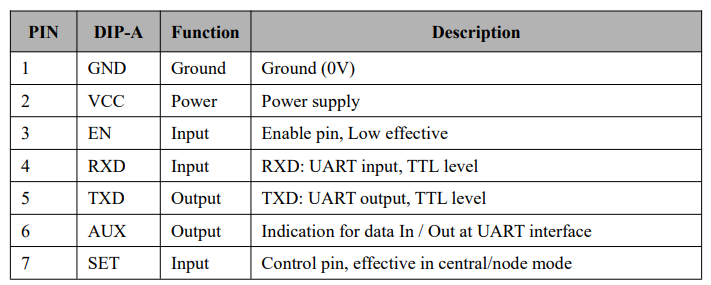
The communication between the central module and node module are bidirectional but the node modules cannot talk with each other.

****

**Fig 3.5** LoRa SX1276

### LoRa SX1276 Pin Description

### 



### Features of LoRa SX1276

* 868/915MHz ISM frequency band
* -138dBm receiver sensitivity
* 20dBm Max. output power
* Serial port wake-up
* Wireless wake-up
* Star networking ability
* Supply voltage 3.4-5.5V

## Buzzer

Basically, the sound source of a piezoelectric sound component is a piezoelectric diaphragm. A piezoelectric diaphragm consists of a piezoelectric ceramic plate which has electrodes on both sides and a metal plate (brass or stainless steel, etc.). A piezoelectric ceramic plate is attached to a metal plate with adhesives. Applying D.C. voltage between electrodes of a piezoelectric diaphragm causes mechanical distortion due to the piezoelectric effect. For a misshaped piezoelectric element, the distortion of the piezoelectric element expands in a radial direction. And the piezoelectric diaphragm bends toward the direction. The metal plate bonded to the piezoelectric element does not expand. Conversely, when the piezoelectric element shrinks, the piezoelectric diaphragm bends in the direction Thus, when AC voltage is applied across electrodes, the bending is repeated, producing sound waves in the air.

To interface a buzzer the standard transistor interfacing circuit is used. Note that if a different power supply is used for the buzzer, the 0V rails of each power supply must be connected to provide a common reference.

If a battery is used as the power supply, it is worth remembering that piezo sounders  
draw much less current than buzzers. Buzzers also just have one ‘tone’, whereas a  
piezo sounder is able to create sounds of many different tones.

To switch on buzzer -high 1

To switch off buzzer -low 1



**Fig 3.6** Picture of buzzer

## HC-05 Bluetooth

HC‐05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. The HC-05 Bluetooth Module can be used in a Master or Slave configuration, making it a great solution for wireless communication. This serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04‐External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature).

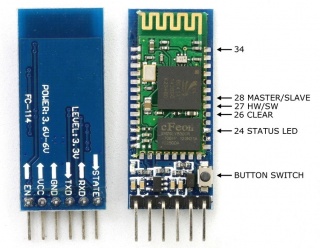
The Bluetooth module HC-05 is a MASTER/SLAVE module. By default the factory setting is SLAVE. The Role of the module (Master or Slave) can be configured only by AT COMMANDS. The slave modules cannot initiate a connection to another Bluetooth device, but can accept connections. Master module can initiate a connection to other devices. The user can use it simply for a serial port replacement to establish connection between MCU and GPS, PC to your embedded project, etc.

### Hardware Features

* Typical ‐80dBm sensitivity.
* Up to +4dBm RF transmit power.
* 3.3 to 5 V I/O.
* PIO(Programmable Input/Output) control.
* UART interface with programmable baud rate.
* With integrated antenna.
* With edge connector.

### Software Features

* Slave default Baud rate: 9600, Data bits:8, Stop bit:1,Parity:No parity.
* Auto‐connect to the last device on power as default.
* Permit pairing device to connect as default.
* Auto‐pairing PINCODE:”1234” as default.

[](https://wiki.eprolabs.com/index.php?title=File:FC-114.jpg)

**Fig 3.7** Bluetooth module

### Pin Description

The HC-05 Bluetooth Module has 6pins. They are as follows:

**ENABLE**: When enable is pulled LOW, the module is disabled which means the module will not turn on and it fails to communicate.When enable is left open or connected to 3.3V, the module is enabled i.e the module remains on and communication also takes place.

**Vcc**: Supply Voltage 3.3V to 5V

**GND:** Ground pin

**TXD & RXD**: These two pins acts as an UART interface for communication

**STATE**: It acts as a status indicator. When the module is not connected to / paired with any other Bluetooth device, signal goes Low. At this low state, the led flashes continuously which denotes that the module is not paired with other device. When this module is connected to/paired with any other Bluetooth device, the signal goes High. At this high state, the led blinks with a constant delay say for example 2s delay which indicates that the module is paired.

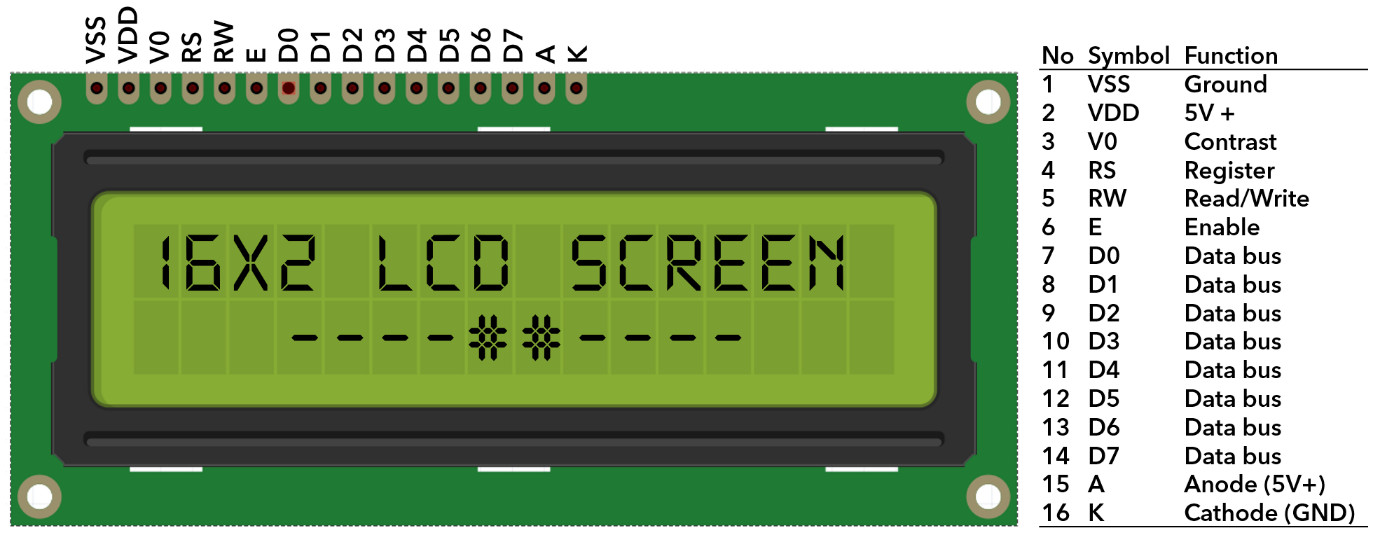
**BUTTON SWITCH**: This is used to switch the module into AT command mode. To enable AT command mode, press the button switch for a second. With the help of AT commands, the user can change the parameters of this module but only when the module is not paired with any other BT device. If the module is connected to any other Bluetooth device, it starts to communicate with that device and fails to work in AT command mode.

## LCD Display

### LCD Background

One of the most common devices attached to a micro controller is an LCD display. Some of the most common LCD’s connected to the many microcontrollers are 16x2 and 20x2 displays. This means 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

### Basic 16x 2 Characters LCD



**Fig 3.8** LCD block diagram

### Pin description:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr. No** | **Pin No.** | **Pin Name** | **Pin Type** | **Pin Description** | **Pin Connection** |
| 1 | Pin 1 | Ground | Source Pin | This is a ground pin of LCD | Connected to the ground of the MCU/ Power source |
| 2 | Pin 2 | VCC | Source Pin | This is the supply voltage pin of LCD | Connected to the supply pin of Power source |
| 3 | Pin 3 | V0/VEE | Control Pin | Adjusts the contrast of the LCD. | Connected to a variable POT that can source 0-5V |
| 4 | Pin 4 | Register Select | Control Pin | Toggles between Command/Data Register | Connected to a MCU pin and gets either 0 or 1.  0 -> Command Mode  1-> Data Mode |
| 5 | Pin 5 | Read/Write | Control Pin | Toggles the LCD between Read/Write Operation | Connected to a MCU pin and gets either 0 or 1.  0 -> Write Operation  1-> Read Operation |
| 6 | Pin 6 | Enable | Control Pin | Must be held high to perform Read/Write Operation | Connected to MCU and always held high. |
| 7 | Pin 7-14 | Data Bits (0-7) | Data/Command Pin | Pins used to send Command or data to the LCD. | In 4-Wire Mode  Only 4 pins (0-3) is connected to MCU  In 8-Wire Mode  All 8 pins(0-7) are connected to MCU |
| 8 | Pin 15 | LED Positive | LED Pin | Normal LED like operation to illuminate the LCD | Connected to +5V |
| 9 | Pin 16 | LED Negative | LED Pin | Normal LED like operation to illuminate the LCD connected with GND. | Connected to ground |

# SOFTWARE DESCRIPTION

This project is implemented using following software’s:

* PIC C compiler - for compilation part
* Proteus 7 (Embedded C) – for simulation part

## PIC Compiler

PIC compiler is software used where the machine language code is written and compiled. After compilation, the machine source code is converted into hex code which is to be dumped into the microcontroller for further processing. PIC compiler also supports C language code.

It’s important that you know C language for microcontroller which is commonly known as Embedded C. As we are going to use PIC Compiler, hence we also call it PIC C. The PCB, PCM, and PCH are separate compilers. PCB is for 12-bit opcodes, PCM is for 14-bitopcodes, and PCH is for 16-bit opcode PIC microcontrollers. Due to many similarities, all three compilers are covered in this reference manual. Features and limitations that apply to only specific microcontrollers are indicated within. These compilers are specifically designed to meet the unique needs of the PIC microcontroller. This allows developers to quickly design applications software in a more readable, high-level language. When compared to a more traditional C compiler, PCB, PCM, and PCH have some limitations. As an example of the limitations, function recursion is not allowed.

This is due to the fact that the PIC has no stack to push variables onto, and also because of the way the compilers optimize the code. The compilers can efficiently implement normal C constructs, input/output operations, and bit twiddling operations. All normal C data types are supported along with pointers to constant arrays, fixed point decimal, and arrays of bits.

PIC C is not much different from a normal C program. If you know assembly, writing a C program is not a crisis. In PIC, we will have a main function, in which all your application specific work will be defined. In case of embedded C, you do not have any operating system running in there. So you have to make sure that your program or main file should never exit. This can be done with the help of simple while (1) or for (;;) loop as they are going to run infinitely.

We have to add header file for controller you are using, otherwise you will not be able to access registers related to peripherals.

## Proteus

Proteus is software which accepts only hex files. Once the machine code is converted into hex code, that hex code has to be dumped into the microcontroller and this is done by the Proteus. Proteus is a programmer which itself contains a microcontroller in it other than the one which is to be programmed. This microcontroller has a program in it written in such a way that it accepts the hex file from the pic compiler and dumps this hex file into the microcontroller which is to be programmed. As the Proteus programmer requires power supply to be operated, this power supply is given from the power supply circuit designed and connected to the microcontroller in proteus. The program which is to be dumped in to the microcontroller is edited in proteus and is compiled and executed to check any errors and hence after the successful compilation of the program the program is dumped in to the microcontroller using a dumper.

## Procedural steps for compilation, simulation and dumping

### Compilation and simulation steps

For PIC microcontroller, PIC C compiler is used for compilation. The compilation steps are as follows:

* Open PIC C compiler.
* You will be prompted to choose a name for the new project, so create a separate folder where all the files of your project will be stored, choose a name and click save.
* Click **Project, New**, and something the box named 'Text1' is where your code should be written later.
* Now you have to click 'File, Save as' and choose a file name for your source code ending with the letter '.c'. You can name as 'project.c' for example and click save. Then you have to add this file to your project work.
* You can then start to write the source code in the window titled 'project.c' then before testing your source code; you have to compile your source code, and correct eventual syntax errors.
* By clicking on compile option .hex file is generated automatically.
* This is how we compile a program for checking errors and hence the compiled program is saved in the file where we initiated the program.

After compilation, next step is simulation. Here first circuit is designed in Express PCB using Proteus 7 software and then simulation takes place followed by dumping. The simulation steps are as follows:

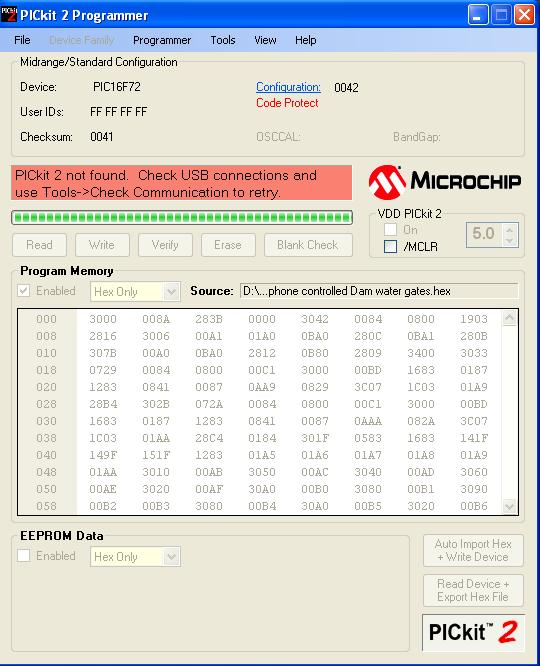
* Open Proteus 7 and click on IS1S6.
* Now it displays PCB where circuit is designed using microcontroller. To design circuit components are required. So click on component option.

10. Now click on letter ’p’, then under that select PIC16F877A ,other components related to the project and click OK. The PIC 16F877A will be called your “*'*Target device*”,* which is the final destination of your source code.

### Dumping steps

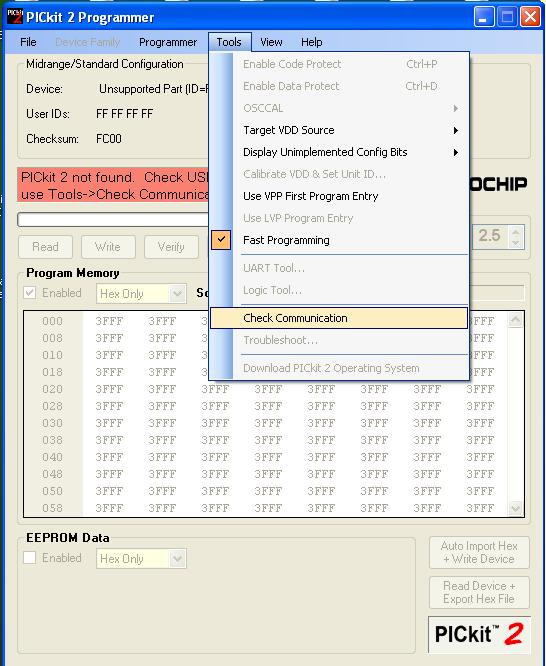
The steps involved in dumping the program edited in proteus 7 to microcontroller are shown below:

1. Initially before connecting the program dumper to the microcontroller kit the window is appeared as shown below:



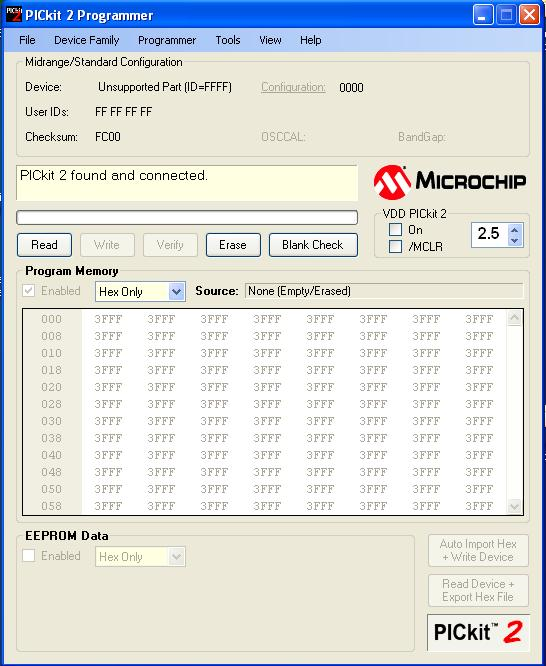
**Fig 4.1** Program dumper window

1. Select Tools option and click on Check Communication for establishing a connection as shown in below window



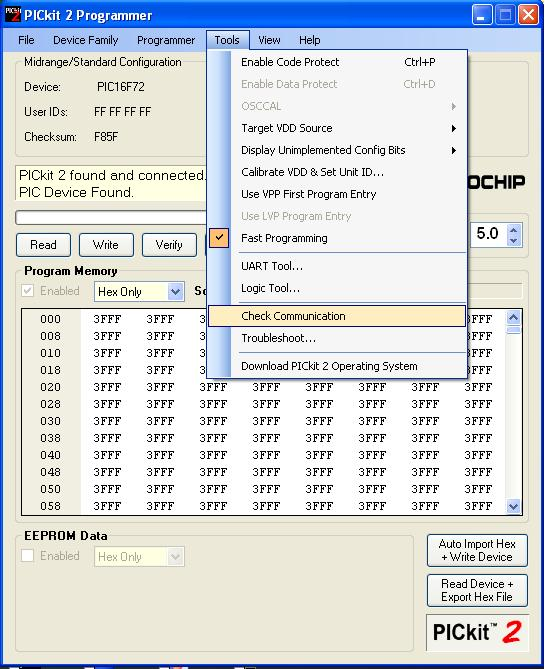
**Fig 4.2** Checking communications before dumping program into microcontroller

3. After connecting the dumper properly to the microcontroller kit the window is appeared as shown below.



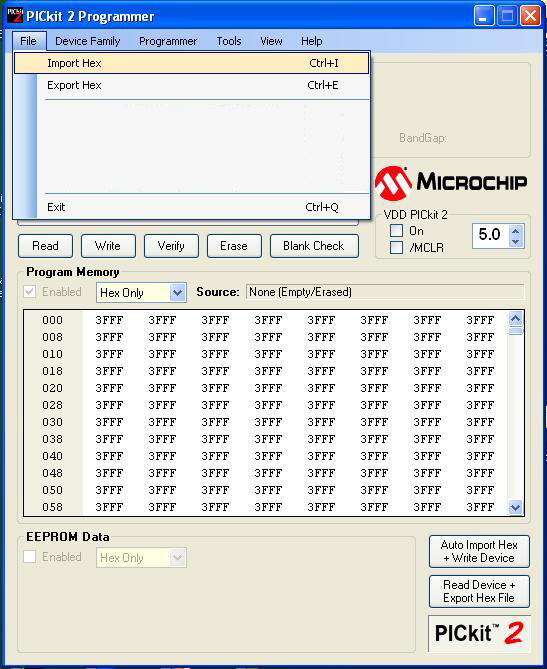
**Fig 4.3** After connecting the dumper to microcontroller

4. Again by selecting the Tools option and clicking on Check Communication the microcontroller gets recognized by the dumper and hence the window is as shown below.



**Fig 4.4** Dumper recognition to microcontroller

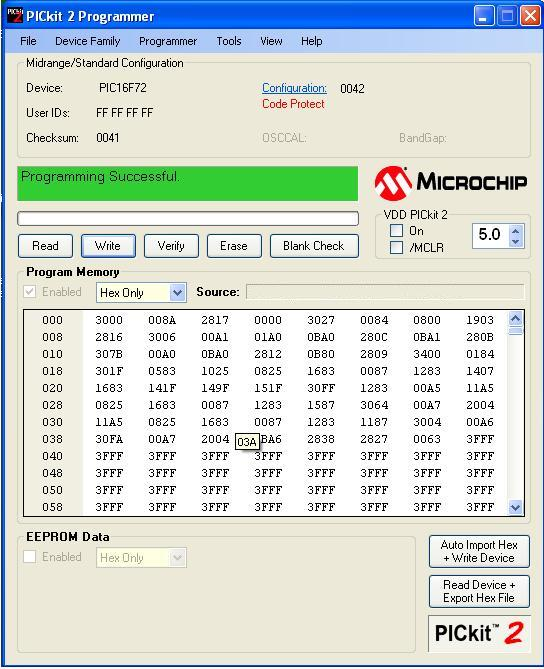
5. Import the program which is ‘.hex’ file from the saved location by selecting File option and clicking on ‘Import Hex’ as shown in below window.

****

**Fig 4.5** Program importing into the microcontroller

6. After clicking on ‘Import Hex’ option we need to browse the location of our program and click the ‘prog.hex’ and click on ‘open’ for dumping the program into the microcontroller.

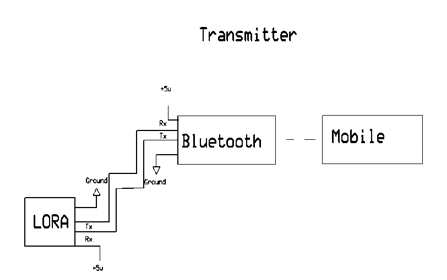
7. After the successful dumping of program the window is as shown below.

****

**Fig 4.6** After program dumped into the microcontroller

# METHODOLOGY

In this chapter, schematic diagram and interfacing of PIC16F72 microcontroller with each module is considered**.**

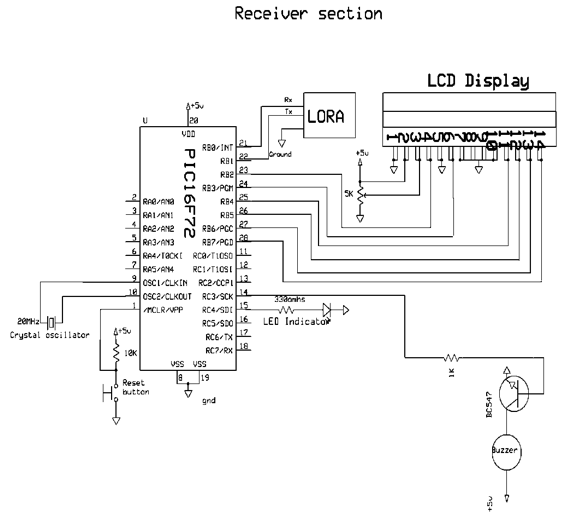


**Fig 5.1** Circuit Diagram of Transmitter Section

In the transmitter section the components which we are using are LORA transmitter , one Bluetooth module, and a mobile phone to give voice signals. So at first we give voice on the mobile and that is converted to text signal via an app called BT voice control next this text is transmitted to the LORA using Bluetooth module. We are converting the voice to text here itself because the bitrate of LORA is less so it becomes difficult to transmit voice over it.

**ABOUT BT VOICE CONTROL APP**

Android Meets Robots : Voice Recognition  
Uses android mobiles internal voice recognition to pass voice commands to your robot  
Pairs with Bluetooth Serial Modules and sends in the recognized voice as a string  
for example if you say Hello the android phone will return a sting \*Hello# to your bluetooth module \*and # indicate the start and stop bits  
Can Be used with any micro controller which can handle strings  
Examples Platforms : Arduino , ARM , PICAXE , MSP430 , 8051 based and many other processors and controllers

****

**Fig 5.2** Circuit Diagram of Receiver Section

The above schematic diagram **Speech to text using LoRa** explains the interfacing section of each component with micro controller. Crystal oscillator connected to 9th and 10th pins of micro controller and regulated power supply is also connected to micro controller and LED’s also connected to micro controller through resistors.

The source code for displaying the message on the LCD display is

#include <16F72.h>

#use delay (clock=20000000)

#use rs232 (baud = 9600, xmit=PIN\_B1,rcv=PIN\_B0,stream=BT) //rs232 standard for serial communication

//New RFID Reader baudrate

#include <flex\_lcd.c>

int i;

char data[17];

char ch;

void main()

{

lcd\_init();

lcd\_putc('\f'); //Clear LCD

lcd\_gotoxy(1,1);

printf(lcd\_putc," LoRa Data");

lcd\_gotoxy(1,2);

printf(lcd\_putc,"Transmission");

output\_high(PIN\_C4);

output\_high(PIN\_C3);

delay\_ms(500);

output\_low(PIN\_C4);

output\_low(PIN\_C3);

delay\_ms(500);

output\_high(PIN\_C4);

output\_high(PIN\_C3);

delay\_ms(500);

output\_low(PIN\_C4);

output\_low(PIN\_C3);

fprintf(BT," \*message# to enter new message\r\n");

while(1)

{

ch = fgetc(BT);

if(ch == '\*')

{

i = 0;

while(1)

{

ch = fgetc(BT);

data[i] = ch;

if(ch == '#')

break;

i++;

}

data[i] = 0;

output\_high(PIN\_C4);

output\_high(PIN\_C3);

delay\_ms(2000);

lcd\_putc('\f'); //Clear LCD

lcd\_gotoxy(1,1);

printf(lcd\_putc,"Received:");

lcd\_gotoxy(1,2);

printf(lcd\_putc,"%s",data);

output\_low(PIN\_C4);

output\_low(PIN\_C3);

}

delay\_ms(70);

}

}

# ADVANTAGES AND DISADVANTAGES

## 6.1 Advantages

* LoRa module is simple in design and user friendly. It is more efficient than other communication technologies like GSM, Wi-Fi.
* Lora module consumes low power.It can run on battery for years
* Buzzer is present which notifies by giving audible alerts. It also gives visible alerts using LCD display
* LoRa modules are pocket friendly. It uses constant envelope modulation that brings lower cost and higher efficiency to the power amplifier

## 6.2 Disadvantages

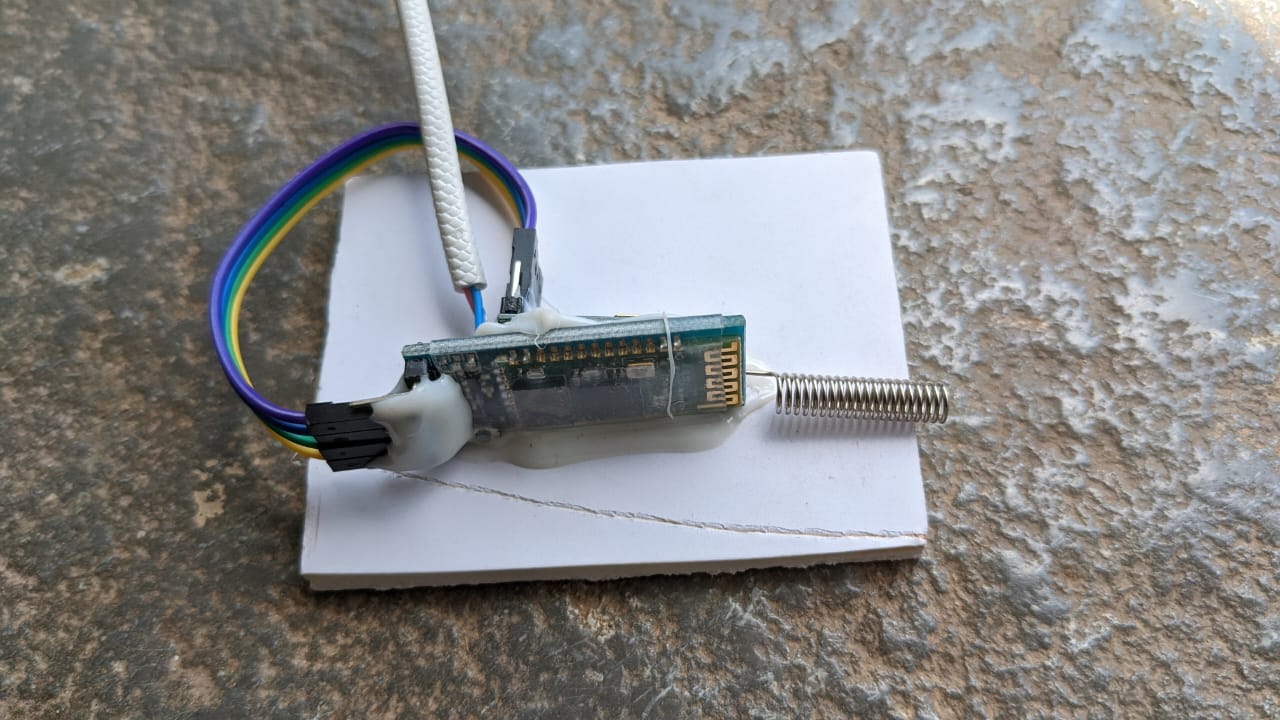
* Interfacing LoRa with microcontroller is highly sensitive.
* Lower data rate of LORA is a problem . It occurs due to long distance transmission
* LORA also has the problem of lower bandwidth
* LoRa network size is limited based on parameter called as duty cycle. It is defined as the percentage of time during which the channel can be occupied

## 6.3 Applications

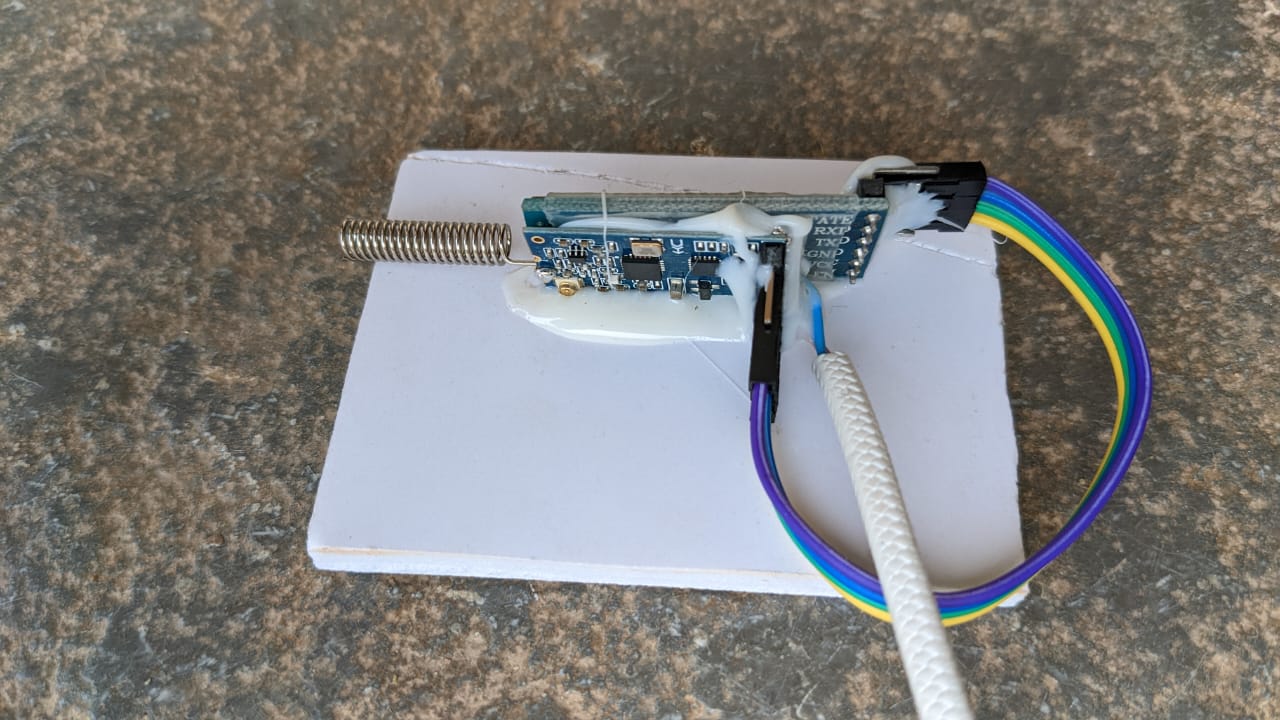
* This system can be practically implemented in real time at defense and army places.
* Communication of disabled people in hospitals.
* Helpful for people who have some urgency like meetings etc.

# RESULTS

## Transmitter Section

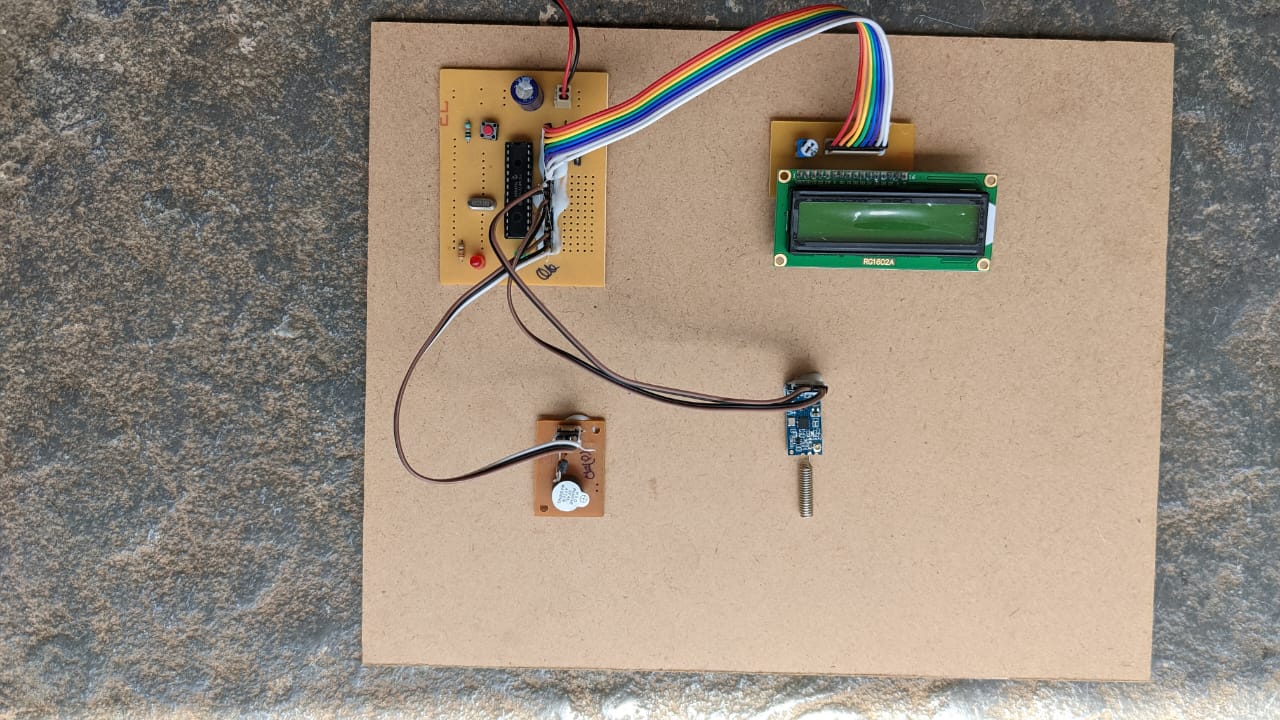


**Fig 7.1** Bluetooth module



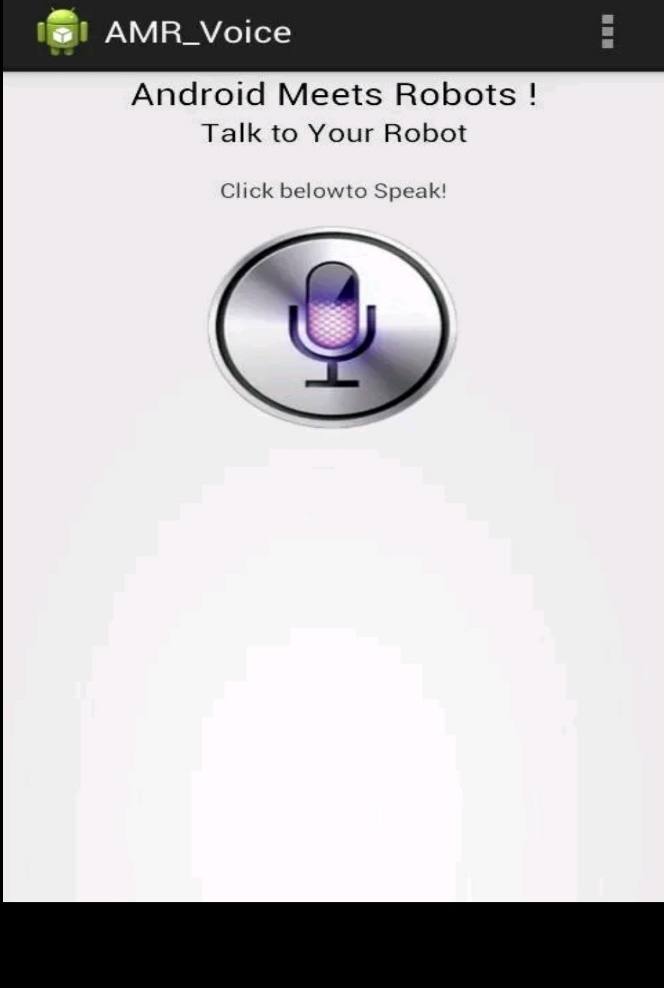
**Fig 7.2** LoRa Module

## Receiver Section



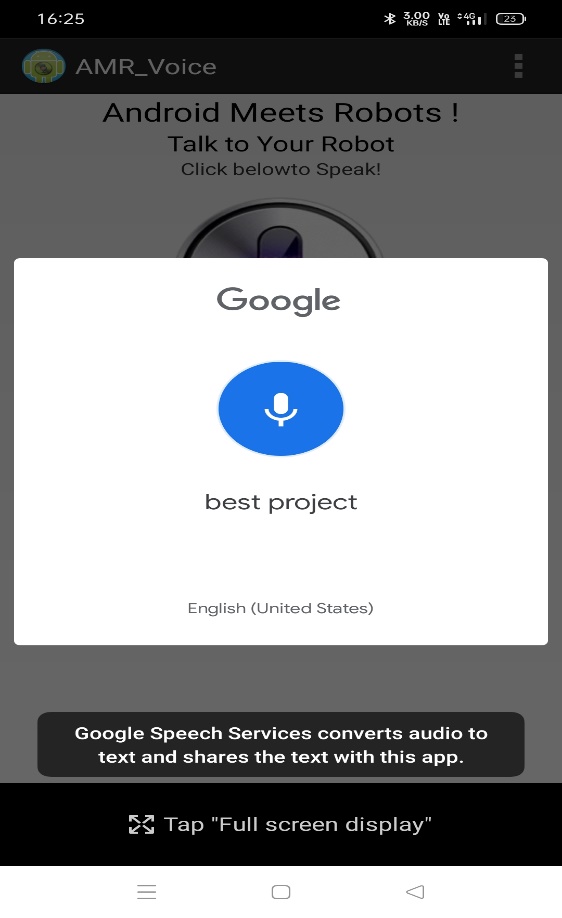
**Fig 7.3** Receiver Figure

## BT Voice Control App



**Fig 7.4** BT voice control

## Result



**Fig 7.5** Voice signal given to mobile phone



**Fig 7.6** Received text on LCD display

# CONCLUSION AND FUTURE SCOPE

## Conclusion

Communication for long range is a challenging issue with power consumption, noise and range concerns. This could be achieved by using LoRa technology. With LoRa as a medium for communication, exchange of information for long distances with low power consumption is made possible.

Presence of every module has been reasoned out and placed carefully, thus contributing to the best working of the unit. Secondly, using highly advanced IC’s with the help of growing technology, the project has been successfully implemented. Thus the project has been successfully designed and tested.

## Future Scope

* Addition of voice module at receiver section to give the voice alerts.
* Addition of multiple channels to transmit the parallel data.
* By using a better version of LoRa module, better communication can be established.

# REFERENCES

[1] Alireza Zourmand, Chan Wai Hung, Andrew Lai Kun Hing, Mohammad AbdulRehman , “Internet of Things (IoT) using LoRa technology”, 2019 IEEE International Conference on Automatic Control and Intelligent Systems.

[2] Umber Noreen, Ahcene Bounceur, Laurent Clavier, “A Study of LoRa Low Power and Wide Area Network Technology”, 3rd International Conference on Advanced Technologies for Signal and Image Processing - ATSIP'2017, May 22-24, 2017, Fez, Morroco.

[3] Prof. Sunil Somani, Smita Patil, Bhushan Vidhate, “LoRa: An Emerging Star In IoT”, Second International Conference on Intelligent Computing and Control Systems (ICICCS 2018) IEEE Xplore Compliant Part Number: CFP18K74-ART; ISBN:978-1-5386-2842-3.

[4] Shilpa Devalal, A. Karthikeyan, “LoRa technology-an overview”, 2nd International conference on Electronics, Communication and Aerospace Technology (ICECA 2018) IEEE Conference Record # 42487; IEEE Xplore ISBN:978-1-5386-0965-1

[5] T. Bouguera, J.-F. Diouris, J.-J. Chaillout, R. Jaouadi, and G. Andrieux, ‘‘Energy consumption model for sensor nodes based on LoRa and LoRaWAN,’’ Sensors, vol. 18, no. 7, p. 2104, Jun. 2018, doi: 10.33