# Declaration by students

The report has been formed by our deep conceptual understanding with our immense hard work and dedication on the topic “Voice Recognition Robot” and has been submitted for the evaluation purpose

**Name of Students:**


Lakshya Sharma (9912102362)

Khaimchand Sharma (9912102339)

Nitij Agarwal (9912102333)

# ACKNOWLEGDEMENT

We take this opportunity to express our deepest gratitude towards Dr. Vipin Balyan, our project guide, who has been the driving force behind this project and whose guidance and co-operation has been a source of inspiration for us.

We are very much thankful to our professors, colleagues and authors of various publications to which we have been referring to. We express our sincere appreciation and thanks to all those who have guided us directly or indirectly in our project. Also much needed moral support and encouragement was provided on numerous occasions by our whole division

Finally we thank our parents for their immense support.

| | | |
|---|---|---|
| **Lakshya Sharma** | **Khaimchand Sharma** | **Nitij Agarwal** |
| (9912102362) | (9912102339) | (9912102333) |

# Abstract

The term “robot” generally connotes some anthropomorphic (human-like) appearance. Brooks [5] research coined some research issues for developing humanoid robot and one of the significant research issues is to develop machine that have human-like perception. What is human-like perception? - The five classical human sensors - vision, hearing, touch, smell and taste; by which they percept the surrounding world. The main goal of our project is to introduce “hearing” sensor and also the voice synthesis to the Mobile robot such that it is capable to interact with human through Spoken Natural Language. Voice recognition is a prominent technology, which helps us to introduce “hearing” as well as Natural Language interface through Voice for the Human-Robot interaction. So the promise of anthropomorphic robot is starting to become a reality. We have chosen Mobile Robot, because this type of robot is getting popular as a service robot in the social context, where the main challenge is to interact with human. Two type of approach we have chosen for Voice User Interface (VUI) implementation - using a Hardware system and another one, using a Software system. The design and both implementation approaches are presented in this report. We have performed tests with simple words for different types of robotics activities; and also analyzed the test result to find-out the problems and limitations. This report presents all the test results and the findings, which we have achieved through out the project.

# Chapter 1

# INTRODUCTION

Voice is the most used way of communication for people. We born with the skills of speaking, learn it easily during our early childhood and mostly communicate with each other with speech throughout our lives. By the developments of communication technologies in the last era, speech starts to be an important interface for many systems. Instead of using complex different interfaces, speech is easier to communicate with computers.

In this project, it is aimed to control a robot with voice commands. The robot is able to recognize spoken commands to move correctly. To give a direction to robot, first the voice command is send to the voice recognition module using a microphone mounted on it. Firstly, the computer trains the command by voice recognition system through the commander. And then computer converts the voice command to direction command that predefined and recognizable by robot. When the robot gets the direction command, it moves according to spoken command. The whole project is based on the Arduino Uno R3 which is based on ATMEGA328 microcontroller also the result is an initiative development of voice controlled LED(light emitting diode). The whole agenda is based on the idea of human-robot interaction which has a major role in the upcoming future.

Another advantage to this stand-alone speech-recognition circuit (SRC) is its programmability. You can program and train the SRC to recognize the unique words you want recognized. The SRC can be easily interfaced to the robot's CPU.

To control and command an appliance (computer, VCR, TV security system, etc.) by speaking to it, will make it easier, while increasing the efficiency and effectiveness of working with that device. At its most basic level speech recognition allows the user to perform parallel tasks, (i.e. hands and eyes are busy elsewhere) while continuing to work with the computer or appliance.

Robotics is an evolving technology. There are many approaches to building robots, and no one can be sure which method or technology will be used 100 years from now. Like biological systems, robotics is evolving following the Darwinian model of survival of the fittest.

Suppose you want to control a menu driven system. What is the most striking property that you can think of?

Well the first thought that came to our mind is that the range of inputs in a menu driven system is limited. In fact, by using a menu all we are doing is limiting the input domain space. Now, this is one characteristic which can be very useful in implementing the menu in standalone systems. For example think of the pine menu or a washing machine menu. How many distinct commands do they require?

# Chapter 2

# THE TASK

The purpose of this project is to build a robotic wheel which could be controlled using voice commands. Generally these kinds of systems are known as Speech Controlled Automation Systems (SCAS). Our system will be a prototype of the same. We are not aiming to build a robot which can recognize a lot of words. Our basic idea is to develop some sort of menu driven control for our robot, where the menu is going to be voice driven. What we are aiming at is to control the robot using following voice commands. Robot which can do these basic tasks:-

1. Start
2. Move forward
3. Move back
4. Turn right
5. Turn left
6. Stop (stops doing the current job)

Apart from the above stuff the core process involves the use of techniques which has helped in the functioning of the above samples of the voices in the voice recognition commander which has trained commands stored by us via microphone through the voice recognition module. When we talk about the basic stuff then it also involves the rigorous study of the books as well as various research papers through which we could accomplish our task. Moreover, the module also supports the trained commands in various other languages as English, Italian, German, French, Japanese and Spanish.

# Chapter 3

# VOICE RECOGNITION TYPES AND STYLES

Voice enabled devices basically use the principal of speech recognition. It is the process of electronically converting a speech waveform (as the realization of a linguistic expression) into words (as a best-decoded sequence of linguistic units).

Converting a speech waveform into a sequence of words involves several essential steps:

1. A microphone picks up the signal of the speech to be recognized and converts it into an electrical signal. A modern speech recognition system also requires that the electrical signal be represented digitally by means of an analog-to-digital (A/D) conversion process, so that it can be processed with a digital computer or a microprocessor.

2. This speech signal is then analyzed (in the analysis block) to produce a representation consisting of salient features of the speech. The most prevalent feature of speech is derived from its short-time spectrum, measured successively over short-time windows of length 20–30 milliseconds overlapping at intervals of 10–20 Ms. Each short-time spectrum is transformed into a feature vector, and the temporal sequence of such feature vectors thus forms a speech pattern.

3. The speech pattern is then compared to a store of phoneme patterns or models through a dynamic programming process in order to generate a hypothesis (or a number of hypotheses) of the phonemic unit sequence. (A phoneme is a basic unit of speech and a phoneme model is a succinct representation of the signal that corresponds to a phoneme, usually embedded in an utterance.) A speech signal inherently has substantial variations along many dimensions.

Before we understand the design of the project let us first understand speech recognition types and styles. Speech recognition is classified into two categories, speaker dependent and speaker independent.

**Speaker dependent** systems are trained by the individual who will be using the system. These systems are capable of achieving a high command count and better than 95% accuracy for word recognition. The drawback to this approach is that the system only responds accurately only to the individual who trained the system. This is the most common approach employed in software for personal computers.

**Speaker independent** is a system trained to respond to a word regardless of who speaks. Therefore the system must respond to a large variety of speech patterns, inflections and enunciation's of the target word. The command word count is usually lower than the speaker dependent however high accuracy can still be maintain within processing limits. Industrial requirements more often need speaker independent voice systems, such as the AT&T system used in the telephone systems.

A more general form of voice recognition is available through feature analysis and this technique usually leads to "speaker-independent" voice recognition. Instead of trying to find an exact or near-exact match between the actual voice input and a previously stored voice template, this method first processes the voice input using "Fourier transforms" or "linear predictive coding (LPC)", Mel-frequency Cepstral Coefficients(MFCC) then attempts to find characteristic similarities between the expected inputs and the actual digitized voice input. These similarities will be present for a wide range of speakers, and so the system need not be trained by each new user. The types of speech differences that the speaker-independent method can deal with, but which pattern matching would fail to handle, include accents, and varying speed of delivery, pitch, volume, and inflection. Speaker-independent speech recognition has proven to be very difficult, with some of the greatest hurdles being the variety of accents and inflections used by speakers of different nationalities. Recognition accuracy for speaker independent systems is somewhat less than for speaker-dependent systems, usually between 90 and 95 percent. Speaker independent systems do not ask to train the system as an advantage, but perform with lower quality. These systems find applications in telephony communications such as dictating a number or a word where many people are in concern.  However, there is a need for a well training database in speaker independent systems.

**Recognition Style**

Speech recognition systems have another constraint concerning the style of speech they can recognize. They are three styles of speech: isolated, connected and continuous.

**Isolated speech** recognition systems can just handle words that are spoken separately. This is the most common speech recognition systems available today. The user must pause between each word and command spoken. The speech recognition circuit is set up to identify isolated words of .96 second lengths.

**Connected** is a half-way point between isolated word and continuous speech recognition. Allows users to speak multiple words. The Voice recognition module can be set up to identify words or phrases 40 seconds in length. This reduces the word recognition vocabulary number to 20.

**Continuous** is the natural conversational speech we are use to in everyday life. It is extremely difficult for a recognizer to shift through the text as the word tend to merge together. For instance, "Hi, how are you doing?" sounds like "Hi,.howyadoin" Continuous speech recognition systems are on the market and are under continual development.

# Chapter 4

# FEATURE EXTRACTION

The signal, as first captured by the microphone, contains information in a form not suitable for pattern recognition. However, it can be represented by a limited set of features relevant for the task. These features more closely describe the variability of the phonemes (such as vowels and consonants) that constitute each word. There are different techniques to extract the required features such as DPCM, LPC, MFCC, etc. LPC and MFCC are most successful future extraction techniques and mainly one of these techniques is used in the speech recognition projects [2, 3, 4, and 5]. As a comparison of these two techniques, Sahar E. Bou-Ghazale and John H. L. Hansen show these results in their project "A Comparative Study of Traditional and Newly Proposed Features for Recognition of Speech Under Stress"[4] :

| **TECHNIQUES** | | | **SPEAKING STYLES** | | | | **OVERALL RECOGNITION** | |
|---|---|---|---|---|---|---|---|---|
| | NEUTRAL | | ANGRY | | LOAD | | | |
| | | | | | | | | |
| **LPC** | 61.65% | | 37.78% | | 43.89% | | 48.19% | |
| | | | | | | | | |
| **MFCC** | 83.52% | | 58.15% | | 63.89% | | 69.15% | |

Table 4-1 Recognition performance based on feature extraction.

As shown at the tables, MFCC is more successful than LPC. Because of these results, it is decided to implement MFCC in this project.

## 4.1. Mel Frequency Cepstral Coefficients

The speech input is typically recorded at a sampling rate above 10000 Hz. This sampling frequency is chosen to minimize the effects of aliasing in the analog to-digital conversion. These sampled signals can capture all frequencies up to 5 kHz, which cover most energy of sounds that are generated by humans. The main purpose of the MFCC processor is to mimic the behavior of the human ears. In addition, rather than the speech waveforms themselves, MFCC's are shown to be less susceptible to mentioned variations. It is shown below the block diagram of MFCC process [11]:

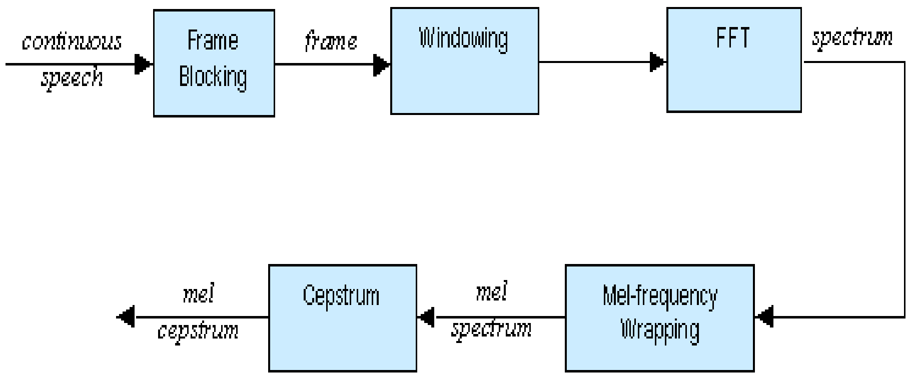


**FIGURE 4.1 Block diagram of MFCC Process [11]**

### 4.1.1. Frame Blocking

In this step the continuous speech signal is blocked into frames of N samples, with adjacent frames being separated by M (M < N). The first frame consists of the first N samples. The second frame begins M samples after the first frame, and overlaps it by N - M samples. Similarly, the third frame begins 2M samples after the first frame (or M samples after the second frame) and overlaps it by N - 2M samples. This process continues until all the speech is accounted for within one or more frames. Typical values for N and M are N = 256 (which is equivalent to ~ 30 msec windowing and facilitate the fast radix-2 FFT) and M = 100 [11].

### 4.1.2. Windowing

The next step in the processing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame. The concept here is to minimize the spectral distortion by using the window to taper the signal to zero at the beginning and end of each frame. If we define the window as, where N is the number of samples in each frame, then the result of windowing is the signal [11]:

Y (n) = x (n) w (n), 0<=n<=N-1……………………………………………equation (1)

Typically the Hamming Window is used, which is of the form:

W (n) = 0.54 – 0.46 cos ((2*3.14*n)/N-1), 0<=n<=N-1……….equation (2)

### 4.1.3. Cepstrum

In this step, we convert the log Mel spectrum back to time. The result is called The Mel frequency Cepstrum coefficients (MFCC). The cepstral representation of the speech spectrum provides a good representation of the local spectral properties of the signal for the given frame analysis. Because the Mel spectrum coefficients (and so their logarithm) are real numbers, we can convert them to the time domain using the Discrete Cosine Transform (DCT). Therefore we can calculate the MFCC's.

$$c = \sum_{k=1}^{k} (\log S_k) \cos\left[\frac{n\left(k-\frac{1}{2}\right)\pi}{k}\right], n = 1,2, \ldots k$$ …………… equation (1)

Where $S_k$ is the Mel Scaled Signal got after wrapping. Cn is the Cepstral Coefficient [11].

# Chapter 5

# NATURE OF PROBLEM

Voice recognition is the process of finding an interpretation of a spoken utterance; typically, this means finding the sequence of words that were spoken.

This involves preprocessing the acoustic signals to parameterize it in a more usable and useful form. The input signal must be matched against a stored pattern and then makes a decision of accepting or rejecting a match. No two utterances of the same word or sentence are likely to give rise to the same digital signal. This obvious point not only underlies the difficulty in speech recognition but also means that we be able to extract more than just a sequence of words from the signal.

The different types of problems we are going to face in our project have been enumerated below: -

**5.1 DIFFERENCES IN THE VOICES OF DIFFERENT PEOPLE**

The voice of a man differs from the voice of a woman that again differs from the voice of a baby. Different speakers have different vocal tracts and source physiology.

Electrically speaking, the difference is in frequency. Women and babies tend to speak at higher frequencies from that of men.

**5.2 DIFFERENCES IN THE LOUDNESS OF SPOKEN WORDS**:-

No two persons speak with the same loudness. One person will constantly go on speaking in a loud manner while another person will speak in a light tone. Even if the same person speaks the same word on two different instants, there is no guarantee that he will speak the word with the same loudness at the different instants. The problem of loudness also depends on the distance the microphone is held from the user's mouth.

Electrically speaking, the problem of difference is reflected in the amplitude of the generated digital signal. Even if the same person speaks the same word at two different instants of time, there is no guarantee that he will speak exactly similarly on both the occasions. Electrically speaking there is a problem of difference in time i.e. indirectly frequency.

## 5.3 DIFFICULTY IN THE HARDWARE IMPLEMENTATION

Since the above difficulties were regarding the software but hardware has way far dominant behavior in comparison. Interfacing the software i.e. Voice recognition commander is a quite complex task, since it contains various trained voice samples which are needed to control the robot.

Also, since we are using Arduino UNO R3 (ATMEGA 328) so we also have to keep in our minds the voltage requirements of Arduino microcontroller (i.e. 5V). Along with the voltage requirements of the robotic wheel (12V).

Hardware implementation is the most complicated task when we talk about the human-robot interaction.

# Chapter 6

# SOLUTIONS TO THE PROBLEMS

After analyzing the problems we come out with the solutions which are listed below.

1. **Amplitude Variation**:-

Amplitude variation of the electrical signal output of microphone may occur mainly due to:

a) Variation of distance between sound source and the transducer.

b) Variation of strength of sound generated by source.

To recognize a spoken word, it does not matter whether it has been spoken loudly or less loudly. This is because characteristic features of a word spoken lies in its frequency & not in its loudness (amplitude). Thus, at a certain stage this amplitude information is suitably normalized.

2. **Recognition of a word**: -

If same word is spoken two times at different time instants, they sound similar to us; question arises what is the similarity in-between them? It is important to note that it does not matter whether one of spoken word was of different loudness than the other. The difference lies in frequency. Hence, any large frequency variation would cause the system not to recognize the word. In speaker independent type of system, some logic can be implemented to take care of frequency variation. A small frequency variation i.e. features variation within tolerable limits is considered to be acceptable.

3. **Noise**:-

Along with the sound source of the speech the other stray sounds also are picked up by the microphone, thus degrading the information contained in the signal.

4. **Microphone response**: -

Two different microphones may not have same response. Hence if microphone is changed, or the system is installed on a new PC due to different response the success rate of recognition may drop.

1. In order our voice is recognized by robot at a distance we will use wireless mic. In case robot does not recognize any word, we will make an arrangement such that robot automatically stops after some time.

2. We will use microphone pre-amplifier circuit. It is in-built in Voice recognition module.

3. We use decoding logic and motor driving circuits so chip and motors are made compatible, thereby solving compatibility problem.

4. One of the important problem which needed to be solved was to provide sufficient current and voltage to entire assembly when interfered together. Since the current drawn from supply was so much that a 9V battery could not last for a longer period, we used current buffer IC. In our application we have used L293D.

# CHAPTER 7

# HARDWARE DESIGN

The most challenging part of the entire system is designing and interfacing various stages together. Our approach was to get the analog voice signal being digitized. The frequency and pitch of words be stored in a memory. These stored words will be used for matching with the words spoken. When the match is found, the system outputs the address of stored words. We used voice recognition module transmitted various frequencies each for right, left forward, backward etc.

**SYSTEM DESIGN**

Arduino Uno R3

Microcontroller (ATMEGA 328)

Voice recognition module

Motor driver circuit

Chassis

Jumper Wires

# BLOCK DIAGRAM

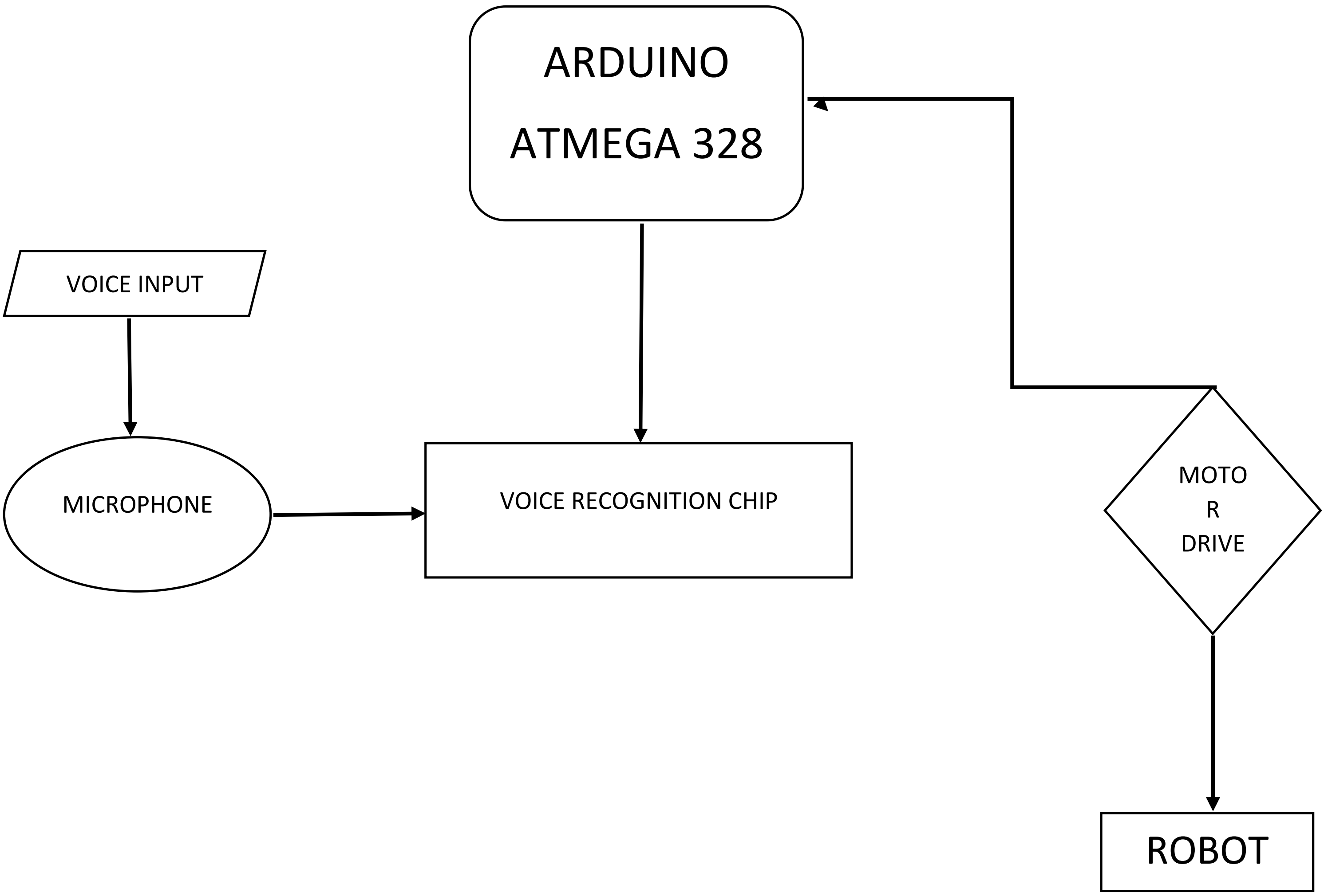


**Figure 7.1. Block diagram of VR bot.**

## 7.1 VOICE RECOGNITION MODULE

VR is a multi-purpose speech recognition module designed to easily add versatile, robust and cost effective speech recognition capabilities to virtually any application. The VR module can be used with any host with an UART interface powered at 3.3V – 5V, such as PIC and Arduino boards. Some application examples include home automation, such as voice controlled light switches, locks or beds, or adding “hearing” to the most popular robots on the market.

### 7.1.1 FEATURES

- A host of built in speaker independent (SI) commands for ready to run basic controls in the following languages:
  - English(US)
  - Italian
  - German
  - French
  - Spanish
  - Japanese

- Supports up to 32 user-defined Speaker Dependent (SD) triggers or commands as well as Voice Passwords. SD custom commands can be spoken in ANY language.
- Sonic Net technology for wireless communications between modules or any other sound source (Audio CD, DVD, MP3 Player).
- DTMF tone generation.
- Easy-to-use and simple Graphical User Interface to program Voice Commands and audio. Module can be used with any host with an UART interface (powered at 3.3V - 5V).
- Simple and robust documented serial protocol to access and program through the host board 3 x GPIO lines (IO1, IO2, and IO3) that can be controlled by new protocol commands. PWM audio output that directly supports 8Ω speakers. Sound playback of up to 9 minutes of recorded sounds or speech.

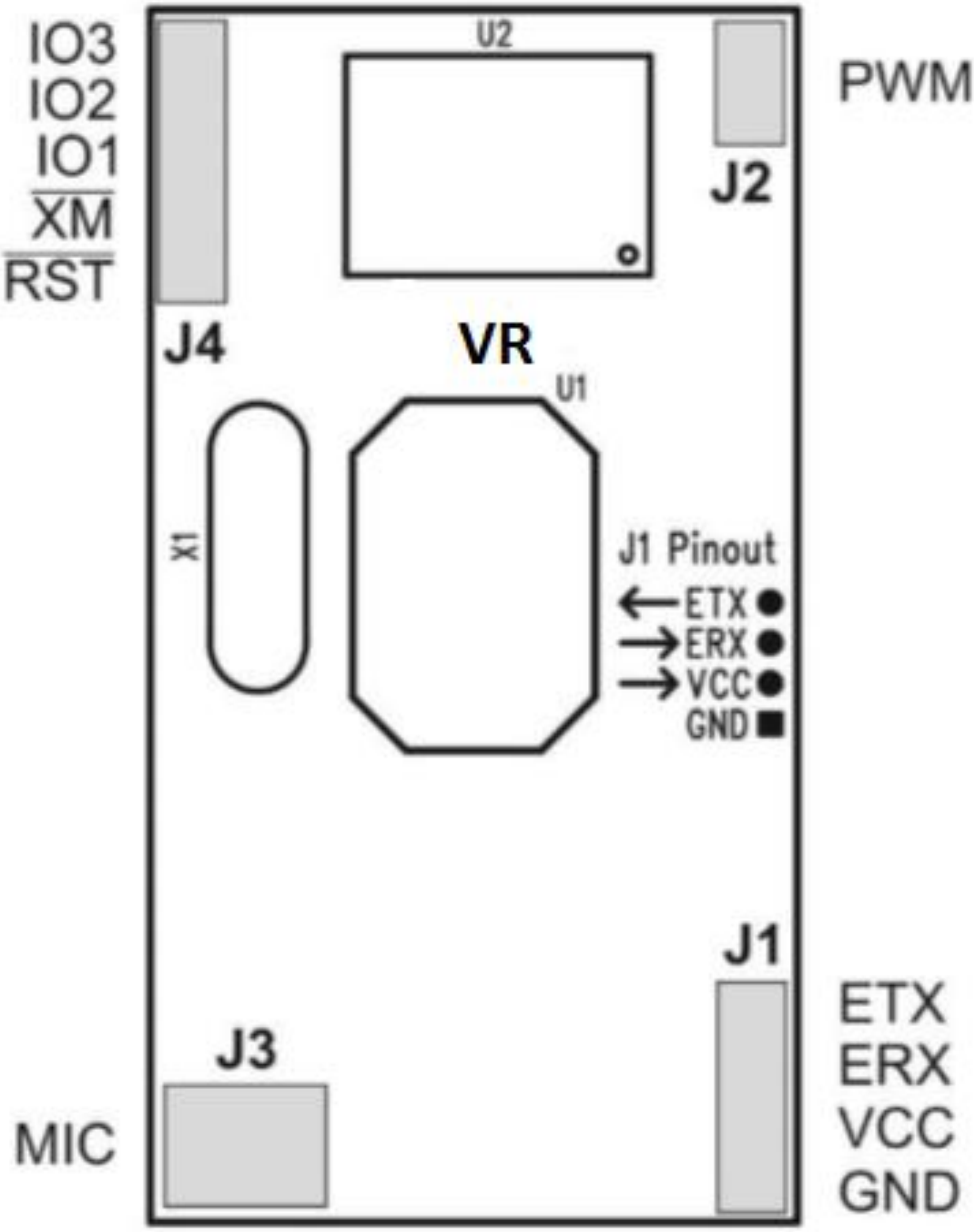


**Figure. 7.2 Pin diagram of VR module**

## 7.1.2 PHYSICAL DIMENSIONS AND PIN ASSIGNMENTS

| Connector | Number | Name | Type | Description |
|---|---|---|---|---|
| J1 | 1 | GND | - | Ground |
| | 2 | VCC | I | Voltage DC input |
| | 3 | ERX | I | Serial Data Receive (TTL level) |
| | 4 | ETX | O | Serial Data Transmit (TTL level) |
| J2 | 1-2 | PWM | O | Differential audio output (can directly drive 8Ω speaker) |
| J3 | 1 | MIC_RET | - | Microphone reference ground |
| | 2 | MIC_IN | I | Microphone input signal |
| J4 | 1 | /RST | I | Active low asynchronous reset (internal 100K pull-up) |
| | 2 | /XM | I | Boot select (internal 1K pull-down) |
| | 3 | IO1 | I/O | General purpose I/O (3.0 VDC TTL level) |
| | 4 | IO2 | I/O | General purpose I/O (3.0 VDC TTL level) |
| | 5 | IO3 | I/O | General purpose I/O (3.0 VDC TTL level) |

**Table 7.1.2 Pin Assignment**

## 7.1.3 RECOMMENDED OPERATING CONDITIONS

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| VCC | Voltage DC Input | 3.3 | 5.0 | 5.5 | V |
| Ta | Ambient Operating Temperature Range | 0 | 25 | 70 | °C |
| ERX | Serial Port Receive Data | 0 | - | VCC | V |
| ETX | Serial Port Transmit Data | 0 | - | VCC | V |

**Table 7.1.3 Operating Conditions for VR module**

## 7.1.4 SERIAL INTERFACE

The VR is a “slave” module communicating via an asynchronous serial interface (commonly known as UART interface), with the following features:

Baud Rate: 9600 (default), 19200, 38700, 57600, and 115200. Frame: 8 Data bits, No parity, 1 Stop bit

The receiver input data line is ERX, while the transmitter output data line is ETX. No handshake lines are used.

Example of a serial data frame representing character “A” (decimal 65 or hexadecimal 41) in figure 7.3 is given below:

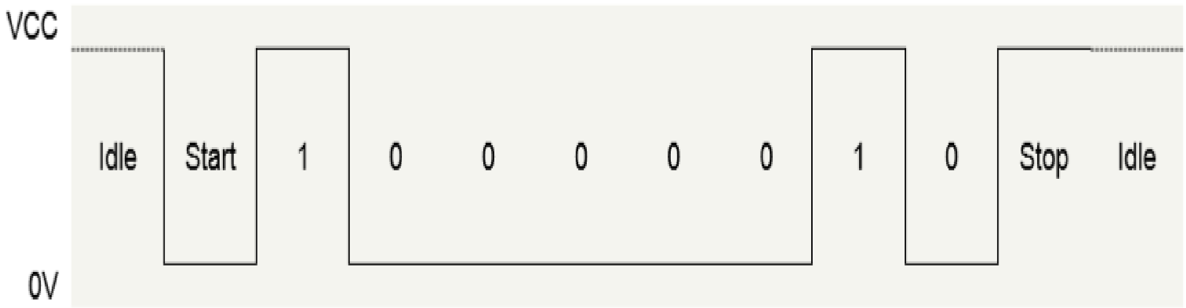


**Figure 7.3 Serial data representation of character A**

## 7.1.5 MICROPHONE

The microphone provided with the VR module is an omnidirectional electret condenser microphone (Horn EM9745P-382):

- Sensitivity -38dB (0dB=1V/Pa @1KHz).
- Load Impedance 2.2K.
- Operating Voltage 3V.
- Almost flat frequency response in the range 100Hz – 20 kHz

If you use a microphone with different specifications the recognition accuracy may be adversely affected. No other kind of microphone is supported by the VR module.

### 7.1.5.1 POSITIONING GUIDELINES

Please note that improper acoustic positioning of the microphone will reduce recognition accuracy. Many mechanical arrangements are possible for the microphone element, and some will work better than others. When mounting the microphone in the final device, keep in mind the following guidelines:

1. **Flush Mounting** - The microphone element should be positioned as close to the mounting surface as possible and should be fully seated in the plastic housing. There must be no airspace between the microphone element and the housing. Having such airspace can lead to acoustic resonance, which can reduce recognition accuracy as seen in figure 7.3 below.

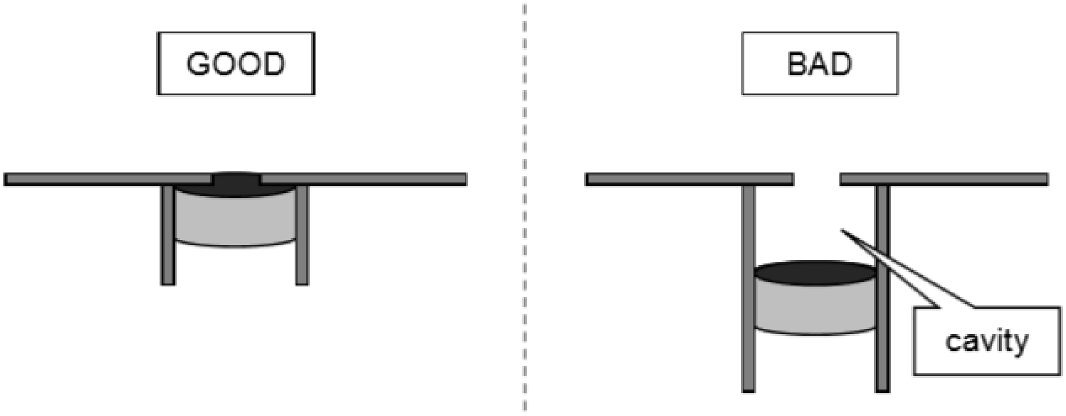


**Figure 7.4 Flush mounting**

2. **No Obstructions, Large Hole** - The area in front of the microphone element must be kept clear of obstructions to avoid interference with recognition. The diameter of the hole in the housing in front of the microphone should be at least 5 mm. Any necessary plastic surface in front of the microphone should be as thin as possible, being no more than 0.7 mm, if possible as shown in the figure below.

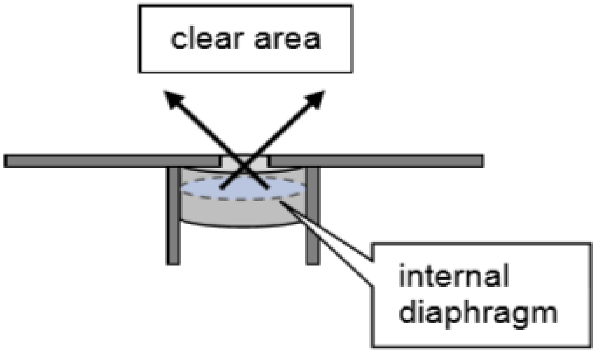


**Figure 7.5 Obstruction in the microphone**

**Insulation** - The microphone should be acoustically isolated from the housing if possible (shown in figure 7.5). This can be accomplished by surrounding the microphone element with a spongy material such as rubber or foam. The provided microphone has this kind of insulating foam. The purpose is to prevent auditory noises produced by handling or jarring the device from being "picked up" by the microphone. Such extraneous noises can reduce recognition accuracy.

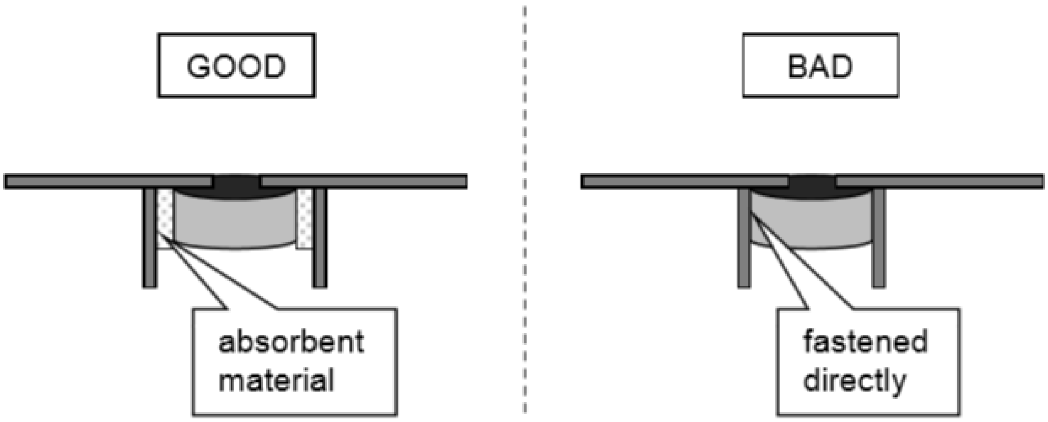


**Figure 7.6 Insulating material**

3. **Distance** - If the microphone is moved from 15 cm to 30 cm from the speaker's mouth, the signal power decreases by a factor of four. The difference between a loud and a soft voice can also be more than a factor of four. Although the internal preamplifier of the VR compensates for a wide dynamic range of input signal strength, if its range is exceeded, the user application can provide feedback to the speaker about the voice volume (see appendix Error codes).

## 7.2 MICROCONTROLLER AND DRIVER CIRCUIT

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.
The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.
This of the board has the following new features:

- 1.0 pin out: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible with both the board that uses the AVR, which operates with 5V and with the Arduino Due that operates with 3.3V. The second one is a not connected pin that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

**Summary**

| | |
|---|---|
| Microcontroller | ATmega328 |
| Operating Voltage | 5V |
| Input Voltage (recommended) | 7-12V |
| Input Voltage (limits) | 6-20V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 40 mA |
| DC Current for 3.3V Pin | 50 mA |
| Flash Memory | 32 KB (ATmega328) of which 0.5 KB used by boot loader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |
| Clock Speed | 16 MHz |

### 7.2.1 Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.
External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and VIN pin headers of the POWER connector.
The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.
The power pins are as follows:

- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND. Ground pins.
- IOREF. This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

### 7.2.2 Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinmode (), digitalWrite () and digitalRead () functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50k ohms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt () function for details.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write () function.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference () function. Additionally, some pins have specialized functionality:
There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference ().
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

### 7.2.3 Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on windows an .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).
A software serial library allows for serial communication on any of the Uno's digital pins.

### 7.2.4 Programming

The Arduino Uno can be programmed with the Arduino software. Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board).
The ATmega328 on the Arduino Uno comes preburned with a boot loader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ().
The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

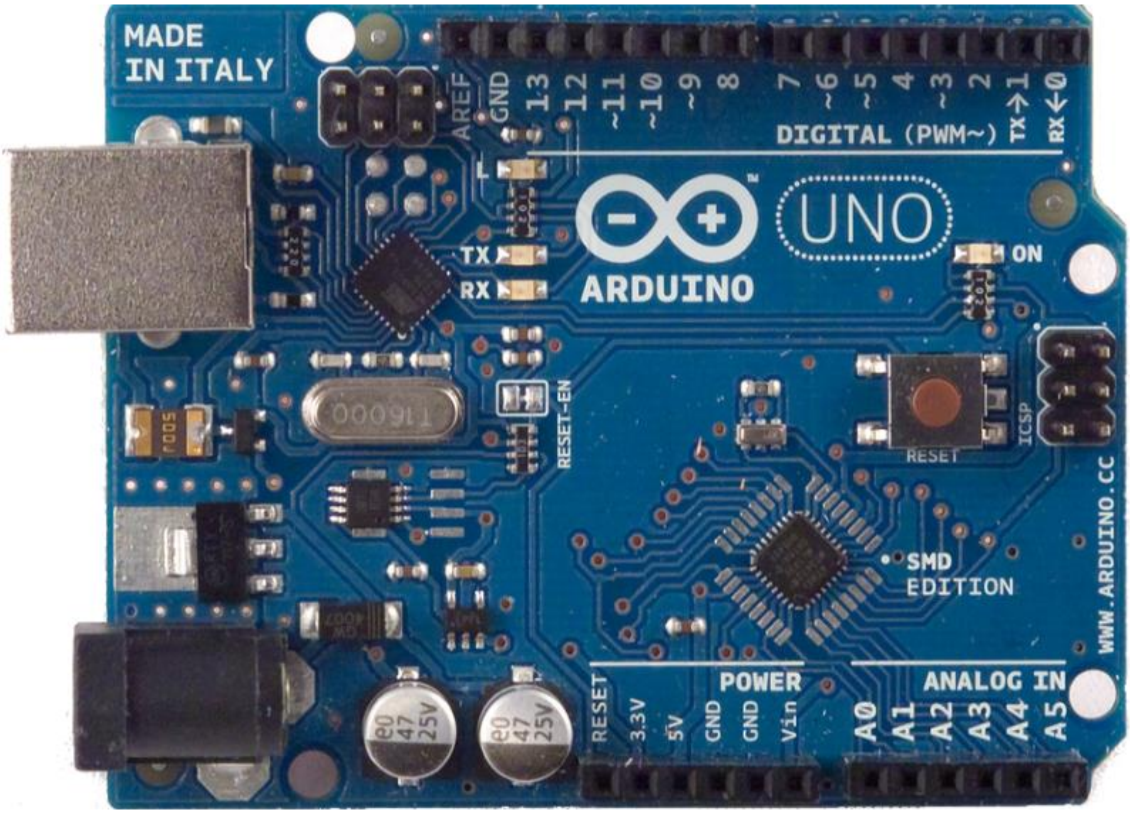


**Figure 7.7 Arduino UNO R3 SMD**

## 7.3 MOTOR DRIVER

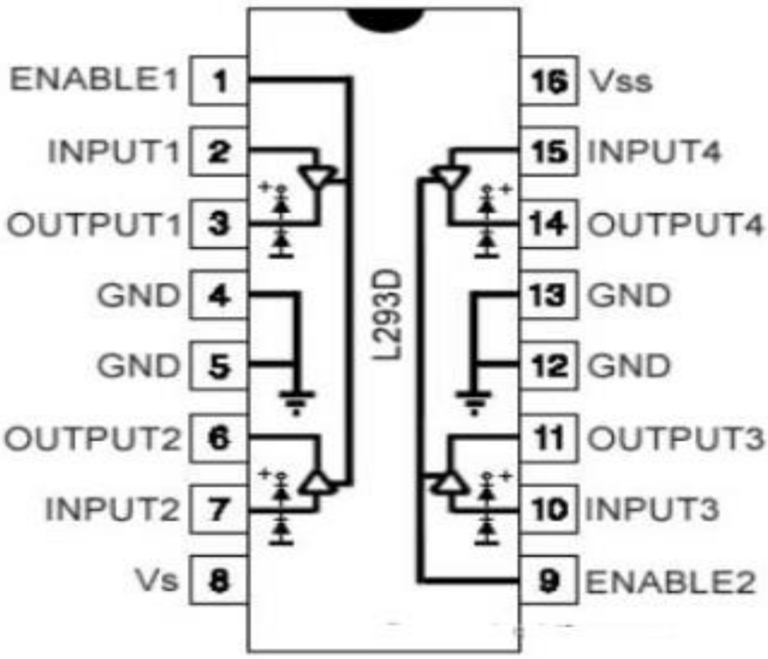


**Figure 7.8 Pin diagram Of L293D**

The L293 and L293D are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications. All inputs are TTL compatible. Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN. When an enable input is high, the associated drivers are enabled, and their outputs are active and in phase with their inputs. When the enable input is low, those drivers are disabled, and their outputs are off and in the high-impedance state. With the proper data inputs, each pair of drivers forms a full-H (or bridge) reversible drive suitable for solenoid or motor applications. On the L293, external high-speed output clamp diodes should be used for inductive transient suppression. A VCC1 terminal, separate from VCC2, is provided for the logic inputs to minimize device power dissipation. The L293and L293D are characterized for operation from 0°C to 70°C.

**BLOCK DIAGRAM**

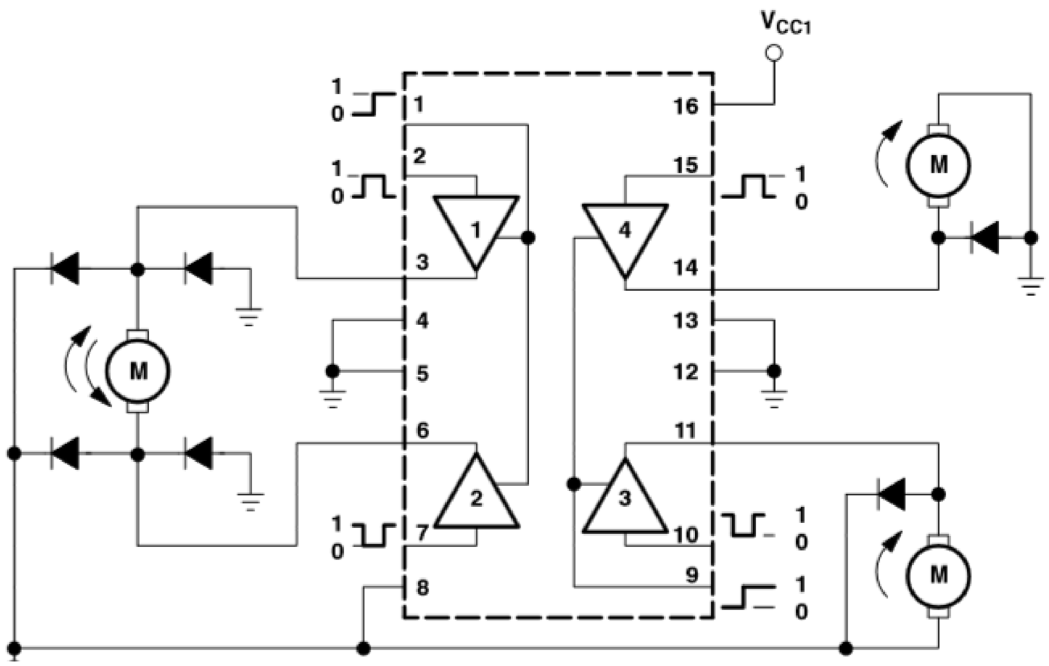


**Figure 7.9 Block diagram of L293D**

## 7.3.1 Operating Conditions

| | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| | Supply voltage | $V_{CC1}$ | 4.5 | 7 | V |
| | | $V_{CC2}$ | $V_{CC1}$ | 36 | |
| $V_{IH}$ | High-level input voltage | $V_{CC1} \leq 7$ V | 2.3 | $V_{CC1}$ | V |
| | | $V_{CC1} \geq 7$ V | 2.3 | 7 | V |
| $V_{IL}$ | Low-level output voltage | | –0.3† | 1.5 | V |
| $T_A$ | Operating free-air temperature | | 0 | 70 | °C |

**Table 7.3.1 Operating Conditions for L293D**

# Chapter 8

# ARDUINO INTERFACING WITH VR MODULE

You can connect the VR module to an Arduino board basically in two ways:
**Bridge mode** – You can control the module using a software serial library and connect to the module with the VR Commander from your PC, with the same pin configuration. This is the preferred connection mode, since it allows simple communication with both the Arduino microcontroller and the PC. All the provided examples for Arduino manage the bridge mode automatically when the VR Commander requests a connection.

**Adapter mode** – You can use the Arduino board as a USB/Serial adapter by holding the microcontroller in reset, but you need to change the connections once you want to control the module from the microcontroller. This connection scheme has the advantage of working with any Arduino board that has an on-board USB/Serial adapter and not needing a spare input pin to enter bridge mode. Also, it does not rely on the AVR microcontroller to do any software bridge between communication pins, so it can be used to check your hardware in case of connection problems.

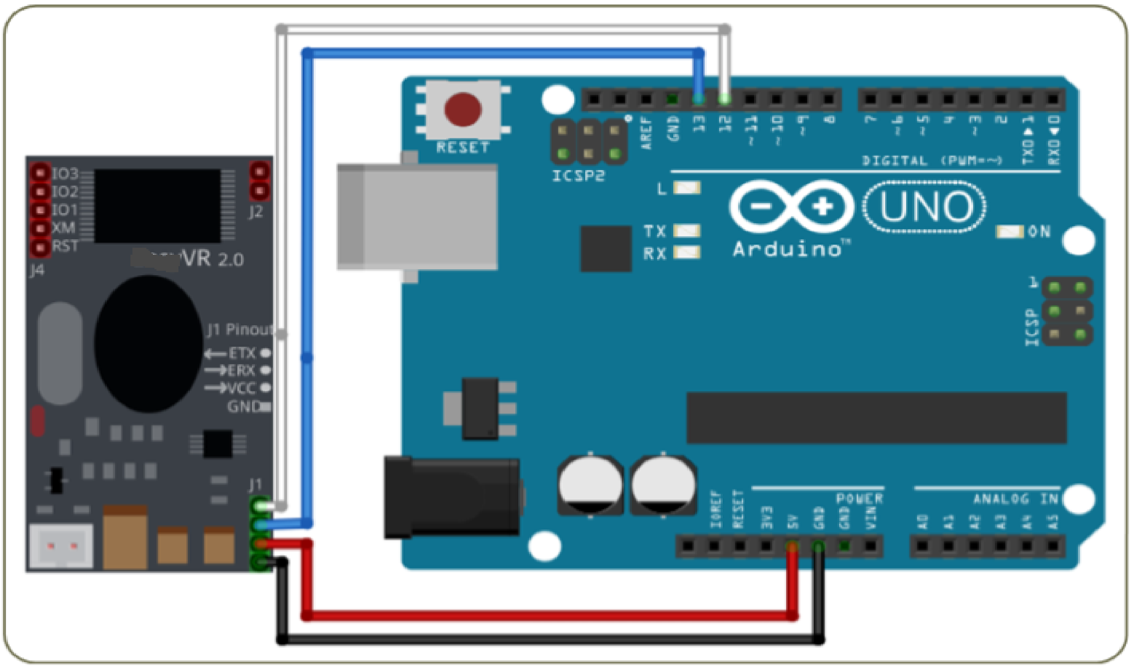


**Figure 8.1 Arduino Interfacing with VR module**

# Chapter 9

# RESULTS

```
Error 11
Say a command in Group 1
Error 11
Say a command in Group 1
Command: 1 = BACK
Say a command in Group 1
Command: 2 = LEFT
Say a command in Group 1
Error 11
Say a command in Group 1
Error 11
Say a command in Group 1
Error 11
Say a command in Group 1
Command: 3 = RIGHT
Say a command in Group 1
```

**Figure 9.1 Output on the serial monitor of Arduino Uno.**

```
Bridge not started!
EasyVR not detected!
EasyVR detected!
Say a command in Group 0
Command: 0 = START
Say a command in Group 1
Error 11
Say a command in Group 1
Command: 1 = BACK
Say a command in Group 1
Error 11
Say a command in Group 1
Timed out, try again...
Say a command in Group 1
Error 11
Say a command in Group 1
Error 11
Say a command in Group 1
Error 11
Say a command in Group 1
Error 11
Say a command in Group 1
Timed out, try again...
Say a command in Group 1
```

**Figure 9.2 Output showing the access of the triggering input.**

```
Bridge not started!
EasyVR detected!
Say a command in Group 0
Command: 0 = START
Say a command in Group 1
Command: 2 = BACK
Say a command in Group 1
Command: 3 = FRONT
Say a command in Group 1
Error 3
Say a command in Group 1
Error 12
Say a command in Group 1
Command: 0 = LEFT
Say a command in Group 1
Timed out, try again...
Say a command in Group 1
Command: 1 = RIGHT
Say a command in Group 1
Timed out, try again...
Say a command in Group 1
Timed out, try again...
Say a command in Group 1
Command: 4 = STOP
Say a command in Group 1
```

**Figure 9.3 Output showing complete commands stored.**

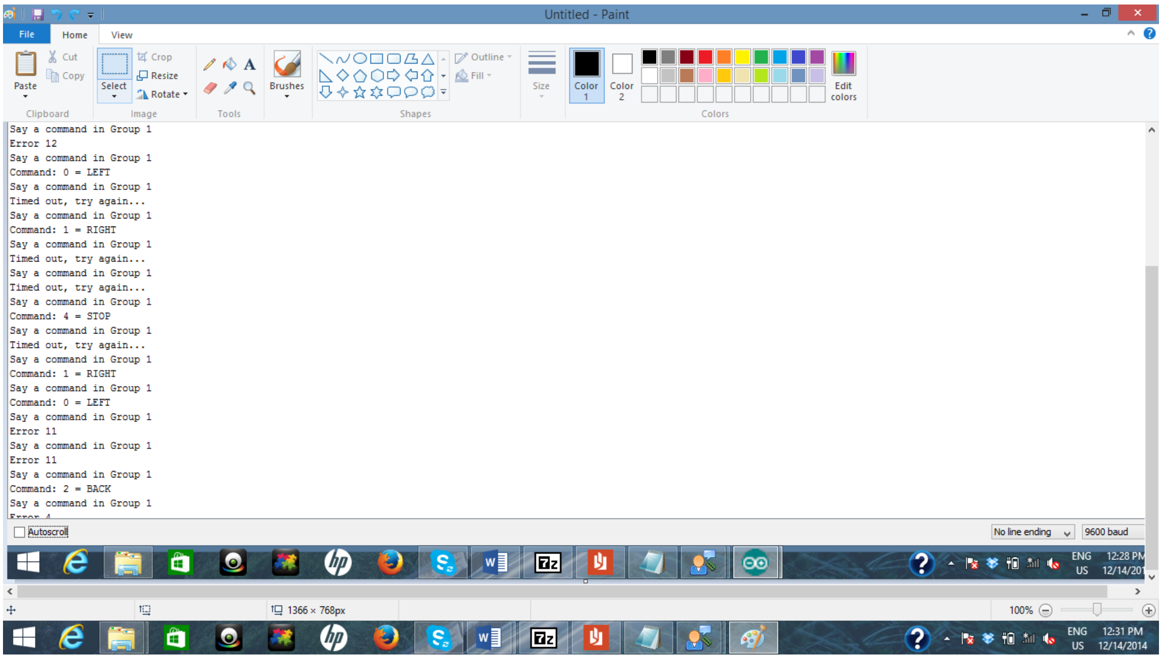


**Figure 9.4 Output showing a screenshot of the outcomes.**

# Chapter 10

# CONCLUSION AND FUTURE SCOPE

Human-Robot interaction is an important, attractive and challenging area. The Service Robot popularity gives the researcher more interest to work with user interface for robots to make it more user friendly to the social context. Voice Recognition technology gives the researcher the opportunity to add Natural language communication with robot in natural and even way. Most of the presented projects in VR interface for robotics emphasize on Mobile Autonomous Service Robot. The working domain of the Service Robot is in the society -to help the people in every day's life and so it should be controlled by the human. In the social context, the most popular humans' communication media is Spoken Natural Language, so to communicate with human the VR interface for Human-Robot interaction is coined.

Main target of our project is to add VR capabilities in the Mobile Robot and investigate the use of a natural language such as English as a user interface for interacting with the Robot also use of other languages is also entertained. We have implemented the VR interface successfully with hardware Speech Recognition device as. We have done the laboratory test with expert users and the real-time test with novice users. After all the implementation and the testing session, we have gained a lot of experience and also found the problems and limitations when introducing VR system as a user interface to robot. From these achieved experiences, we have reached some conclusions. Our first finding is that the hardware VR device is not as matured as the Software PC based VR system. The hardware SR module does not support the complex grammar sentences, which are normal parts of the spoken natural languages. Another thing is that LED is not suitable interface for the user feedback. After testing the system with the novice users in the technical fair, we have found that SR user interface is a promising aid for interaction with robot. It makes them learn quickly to control the robot. We have also found limitation of the Software PC based VR system; the noise factor affects the VR performance of the Voice Recognition Software Program and also the robot performance - means the robot does malfunctioning. Another thing is that when the user is not planning to control the robot; he/she should mute the microphone. The VRSP supports complex sentences; this gives us opportunity try complex sentences to control the robot and we have successfully done this experiment.

## 10.1 Future Scope

We believe such a system would find wide variety of applications. Menu driven systems such as e-mail readers, household appliances like washing machines, microwave ovens, and pagers and mobiles etc. will become voice controlled in future

- The robot is useful in places where humans find difficult to reach but human voice reaches. E.g. in a small pipeline, in a fire-situations, in highly toxic areas.
- The robot can be used as a toy.
- It can be used to bring and place small objects.
- It is the one of the important stage of Humanoid robots.
- Command and control of appliances and equipment
- Telephone assistance systems
- Data entry
- Speech and voice recognition security systems

Also, in gesture based systems voice recognition is of the key importance, can be used as a tool in the concept of Artificial Intelligence based models. Moreover, it can also be used as a bench of reply via an audio output from the robot, can also be used in rob wars, voice controlled wheel chair etc.
This project is a base of the transformation in the field of robotics and technology.

# Chapter 11

# REFERENCES

[1] Stephen Cook, Speech Recognition HOWTO
[2] Fatih AYDINOĞLU, Robot Control System with Voice Command Recognition, Yıldız Technical University, Department of Computer Engineering.
[3] P. M. Grant, Speech recognition techniques
[4] Sahar E. Bou-Ghazale, Member, IEEE, and John H. L. Hansen, Senior Member, IEEE, A Comparative Study of Traditional and Newly Proposed Features for Recognition of Speech Under Stress
[5] Harshavardhana , Varun Ramesh , Sanjana Sundaresh, Vyshak B N, Speaker Recognition System Using MFCC, A Project Work Of 6th Semester Electronics &Communication Engineering, Visvesvaraya Technological University
[6]http://www.ee.ic.ac.uk/hp/staff/dmb/.../voicebox.html
[7] http://www.instructables.com/id/Arduino-Voice-recognition-Basic
[8]http://www.theengineeringprojects.com/2013/04/interfacing-of-easyvr-shield-with-arduino-uno.html
[9] www.timcudesigndys.in
[10]https://www.sparkfun.com/arduino_guide
[11]http://arduino.cc/en/Main/ReleaseNotes

[12]https://www.sparkfun.com/products/11021

[13]http://www.instructables.com/id/Arduino-Voice-recognition-Basic/