GESTURE SUPERVISE MACHINE

A Project report submitted in partial fulfillment of the requirements for

the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by

K.SURYA PAVAN (317126512142) B.GOVARDHAN (317126512125)

S. JITHENDRA (317126512168) V.SAI ROSHINI (317126512175)

Under the guidance of

MURUGAPANDIYAN P

M.E,PhD

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, bheemili mandal, visakhapatnam dist.(A.P) 2020-2021

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, Bheemili mandal, Visakhapatnam dist.(A.P)

PRAGNANAM BRAHMA

ANITS

CERTIFICATE

This is to certify that the project report entitled **"GESTURE SUPERVISE MACHINE "** submitted by **KUNCHAM SURYA PAVAN (317126512142), BEVARA GOVARDHAN (317126512125),SINGAMPALLI JITENDRA (317126512168),VAVILAPALLI SAI ROSHINI (317126512175)** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Electronics & Communication Engineering** of Andhra University, Visakhapatnam is a record of bonafide work carried out under my guidance and supervision.

15/07/21

Project Guide

Dr.P.MURUGAPANDIYAN
Associate Professor
Department of E.C.E

ANITS

Associate Professor
Department of E C E
Anil Neerukonda
Institute of Technology & Sciences
Sangivalasa, Visakhapatnam-531 162

Head of the Department

Dr. V.Rajyalakshmi
Professor&HOD
Department of E.C.E

ANITS

Head of the Department
Department of E C E
Anil Neerukonda Institute of Technology & Sciences
Sangivalasa - 531 162

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our project guide MURUGAPANDIYAN P M.E,PhD, Department of Electronics and Communication Engineering, ANITS, for his/her guidance with unsurpassed knowledge and immense encouragement. We are grateful to **Dr. V. Rajyalakshmi**, Head of the Department, Electronics and Communication Engineering, for providing us with the required facilities for the completion of the project work.

We are very much thankful to the **Principal and Management, ANITS, Sangivalasa,** for their encouragement and cooperation to carry out this work.

We express our thanks to all **teaching faculty** of Department of ECE, whose suggestions during reviews helped us in accomplishment of our project. We would like to thank **all non-teaching staff** of the Department of ECE, ANITS for providing great assistance in accomplishment of our project.

We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

PROJECT STUDENTS

K.SURYA PAVAN (313126512142),
B.GOVARDHAN (313126512125),
S.JITENDRA (313126512168),
V.SAI ROSHINI (313126512175),

ABSTRACT

.

Now a days people are in the verge of making innovative and highly sophisticated devices inorder to make life style much simpular.The devices which are being introduced now a days are having superior features which make the people work much simpular and comfortable. Gesture supervise machine is also one such type of typical machine which is made with the help of ultrasonic sensors attached to personal computer.ultrasonic sensors as usually worked as a transducer for collecting analog signals form surrounding environment and convert into electrical signals. Arduino is used to control the ultrasonic sonic waves and counting the distance between the obstacle and the sensor .Python IDE is used to perform various tasks that user want to achieve with the help of PC using different python inbuilt modules.Based on the obstacle distance calculated by ultrasonics ,a certain required commands transfer from Arduino to python IDE with certain baud rate.Using python modules ,the particular commands ,task will be completed.

voice assistance is also used in the project.we can command voice assistance to performance some functions or tasks such as answering the questions.But it is also necessary to educate the voice assistance to execute tasks accordingly.The main advantage of having voice assistance system is it enables us to personalize the system as per the user requirement or need .In this gesture supervise machine a voice assistant is built to control all IOT devices ,such as youtube status,getting mail notification ,global weather reports and so many other functions.

CONTENTS

LIST OF SYMBOLS viii

LIST OF FIGURES ix

LIST OF TABLES x

LIST OF ABBREVATIONS xi

CHAPTER 1 INTRODUCTION 01

1.1 Project Objective 01

1.2 The speed of sound 01

1.3 Arduino ultrasonic distance 02

1.4 Serial communication 03

1.4.1 ASCII Code 05

1.4.2 Serial communication working 06

1.5 Analysis on two arduino serial communications 07

1.6 Ultrasonic range finder setup for serial monitor output 11

1.6.1 Explanation of the code 12

CHAPTER 2 Ultrasonic sensors **13**

2.1 Introduction 13

2.2 Transducers 13

2.2.1 Piezo electric ultrasonic transducers 14

2.2.2 ultrasonic measuring principals with piezo elements 16

2.2.3 Applications 17

2.3 Types of ultrasonic sensors 18

2.4 HC-SRO4 ultrasonic sensors 20

2.4.1 HC -SRO4 pin configuration 21

2.4.2 HC -SRO4 ultrasonic sensor with arduino board 21

2.4.3 How to use HC - SR04 ultrasonic sensor 22

2.4.4 Features of HC -SRO4 22

2.4.5 Applications 23

2.4.6 Limitations 23

2.5 Timing diagram of ultrasonic sensor 25
2.6 Applications of ultrasonic sensors 26
2.7 Advantages of ultrasonic sensors 27
2.8 Limitations of ultrasonic sensors 27

CHAPTER 3 Artificial intelligence - based voice assistant 29
3.1 Introduction 29
3.2 Background 29
3.2.1 History of voice assistance 29
3.2.2 Future applications 30
3.2.3 Aim of the study 30
3.3 Proposed design 30
3.4 Tasks performed by the voice assistant 31
3.5 Methodology 32
3.5.1 Working of voice assistant 33
3.6 The voice - activated digital assistance as participants of communication 35
3.6.1 Methods of research 36
3.7 Typology 37
3.7.1 speaker dependent speech recognition system 37
3.7.1 speaker independent speech recognition system 37
3.8 Voice assistant 37
3.8.1 Microphone 39
3.8.2 Voice commands generation 40
3.8.3 Activation commands generations 40
3.8.4 General control commands generation 40
3.8.5 Evaluation 40
3.9 Overview of speech recognition 41
3.9.1 Feature extraction 41
3.9.2 Mel frequency cepstrum coefficient 42
3.9.3 Acoustic modeling (hidden markov model) 42
3.9.4 Language modelling 43
3.9.5 speech recognition forges and ahead 44
3.9.6 Battle of digital life assistance 44

3.9.7 How our voice assistant is useful 44

3.9.8 Results 45

CHAPTER 4 Automation python modules 49

4.1 Pyautogui module 49

4.1.1 Moving the mouse 51

4.1.2 Automation of mouse 52

4.1.3 Automation of key board 53

4.2 PYTTSX3 53

4.2.1 Supported synthesizers 54

4.2.2 Using PYTTSX3 55

4.2.3 The engine factory 55

4.2.4 The engine interface 55

4.3 Speech recognition 56

4.3.1 Neural networks 57

4.3.2 End- two-End automatic speech recognition 58

4.4 Introduction to pywhatkit modules 59

4.5 Google NEWS 60

4.5.1 How to use the google news API with python 61

4.6 SMTP 61

CONCLUSIONS 64

FUTURE WORK 65

REFERENCES 65

PAPER PUBLICATION DETAILS 66

LIST OF SYMBOLS

C	Speed of sound in meter per sound (m/s)
T	Temperature in ^{0}C
H	Humidity (relative humidity)
d	distance between sensor and palm
v	speed of sound in air (340m/s) at 20^0 c
t	time taken by the wave to hit obstacle(palm) and again to reach the sensor.
υ	speed of sound in a substance, m/s
d	plate thickness.
λ	wavelength
Δt	transmit time difference between emitting and receiving the sound waves
r	distance between the sound source and the object
Δf	shift of frequency between the reflected wavefront
f_0	frequency of ultrasonic waves
v_S	flow velocity of the particles
θ	The angle between the direction of the emitted ultrasonic pulse and the measuring path

LIST OF FIGURES

Figure no	Title	Page no
Fig 1.1	Distance calculated by ultrasonic sensor	03
Fig 1.2	Serial link between two arduino	04
Fig 1.3	Data transfer between two arduino flow chat	07
Fig 1.4	Digital signal for string data of “E” on (a) TX Arduino Uno 1 and (b) (b) RX Arduino Uno 2 from oscilloscope	08
Fig 1.5	Same caption as Fig. 1.4, but for string data of “E L”	09
Fig 1.6	Same caption as Fig. 1.5, but for string data of “E L”	09
Fig 1.7	Same caption as Fig. 1.7, but for string data of “ENGINEER	10
Fig 1.8	Digital signal for string data of “E” on (a) TX Arduino Uno 1 and (b) (b) RX Arduino Uno 2 for string data of “ENGINEER	10
Fig 2.1	Runtime measuring principle	16
Fig 2.2	Doppler effect	17
Fig 2.3	(a) Object Detected , (b) Object Not Detected	18
Fig 2.4	Ranging Measurement	18
Fig 2.5	Target angle	19
Fig 2.6	Beam Spread	19
Fig 2.7	Ultrasonic sensor module HC SR04	21
Fig 2.8	Ultrasonic-sensor-with-Arduino-board	21
Fig 2.9	Object and the sensor schematics	24
Fig 2.10	Object and the sensor at reflection schematics	24
Fig 2.11	Object and the sensor at reflection schematics	24
Fig 2.12	Condition when object is having irregular surface	25
Fig 2.13	Timing diagram of ultrasonic sensor	25
Fig 3.1	Olden days method matching the word with the pattern present in the dictionary.	33
Fig 3.2	Block diagram represents the working of voice assistant	34
Fig 3.3	An illustration on the modulated tone traversing the	38

signal pathway of a voice capture device in terms of FFT

Fig 3.4 An illustration on the modulated tone traversing the signal pathway of a voice capture device in terms of FFT 39

Fig 3.5 Outline of speech recognition system 41

Fig 3.6 Output screen of performing google search 45

Fig 3.7 Output screen of Playing song on YouTube 46

Fig 3.8 Output screen of searching location 46

Fig 3.9 Output screen of Showing Current News 46

Fig 4.1 The coordinates of a computer screen with 1920×1080 Resolution 50

LIST OF TABLES

Table no	Title	Page no
Table 1.1	ASCII CODE	05

LIST OF ABBREVATIONS

PLC	programmable logic control
USB	universal serial bus
MSB	most significant bit
MUTs	micro-machined ultrasonic transducers
OMUS	optical micro-machined ultrasound sensor
CMUT	capacitive micro-machined ultrasonic transducer
PMUT	Piezoelectric micro-machined ultrasonic transducer
MCU/MPU	microcontroller unit
GTTS	Google text to speech
VUI	Voice User interface
ECMs	electret condenser microphones
MEMS	microelectromechanical system
LPFs	lowpass filters
ADC	Analog to digital converter
TTS	Text to-Speech
MFCC	Mel Frequency Cepstrum Coefficient
PLP	Perceptual Linear Prediction
LPC	Linear Predictive Codes
HMM	Hidden Markov Model
DARPA	Defense Advanced Research Projects Agency

I. INTRODUCTION

So many innovative works are being introduced day by day into the world to reduce peoples work time.It is very sophisticated and enthusiastic when system ask the person the task that it need to achieved instead the person ask it before.

1.1 project objective :

As we already know that for any type of electronic devices that are introduced into the market there will be a software which act as backbone for its successful working.In the similar manner gesture supervise machine needs a software behind. As the technology is growing up new set of programming languages are introduced day by day into the market.python is one among them which contains many inbuilt modules which are already predefined.these modules can be used for automation of a devices.for example PC's microphone can be controlled by the python itself using speech recognition module .Along with python second software Arduino IDE is used to control arduino.Arduino is interfaced with two ultrasonic sensors HC-SRO4 for controlling sonic waves and obstacle distance calculation.for that a algorithm is implemented in the arduino IDE and based on the object distance certain commands are generated in binary pattern with stop and start bits.Ultrasonic sensor act as transducer which collect the analog signals and convert into electrical form .It will send the trigger pulse with certain frequency and collect the same pulse when it hit an obstacle in the form of an echo , based on the time and speed of sound in air the distance between the obstacle and the sensor is calculated.the farmulae which is used for distance calculation is

1.2 The speed of sound:

Ultrasonic range finders measure distance by emitting a pulse of ultrasonic sound that travels through the air until it hits an object. When that pulse of sound hits an object, it's reflected off the object and travels back to the ultrasonic range finder. The ultrasonic range finder measures how long it takes the sound pulse to travel in its round trip journey from the sensor and back. It then sends a signal to the Arduino with information about how long it took for the sonic pulse to travel.Knowing the time it takes the ultrasonic pulse to travel back and forth to the object, and also knowing the speed of sound, the Arduino can calculate the distance to the object. The formula relating the speed of sound, distance, and time traveled is

$$\text{Speed} = \frac{\text{distance}}{\text{time}}$$

Rearranging this formula, we get the formula used to calculate distance:

$$\text{Distance} = \text{Speed} * \text{Time}$$

The time variable is the time it takes for the ultrasonic pulse to leave the sensor, bounce off the object, and return to the sensor. We actually divide this time in half since we only need to measure the distance to the object, not the distance to the object *and* back to the sensor. The speed variable is the speed at which sound travels through air.

The speed of sound in air changes with temperature and humidity. Therefore, in order to accurately calculate distance, we'll need to consider the ambient temperature and humidity. The formula for the speed of sound in air with temperature and humidity accounted for is:

$$\text{C} = 331.4 + (0.606 * \text{T}) + (0.0124 * \text{H})$$

C : Speed of sound in meter per sound (m/s)

331.4 : Speed of sound (in m/s) at 0^0 C and 0% humidity

T: Temperature in ^{0}C

H : % Humidity (relative humidity)

For example, at 20°C and 50% humidity, sound travels at a speed of:

$$\text{C} = 331.4 + (0.606 * 20) + (0.0124 * 50)$$

C= 344.02 m/s

1.3 Arduino ultrasonics distance :

Ultrasonic sensors are great tools to measure distance and detect objects without any actual contact with the physical world. It is used in several applications, like in measuring liquid level, checking proximity and even more popularly in automobiles to assist in self-parking or anti-collision systems. Previously we have build many Ultrasonic Sensor projects like water level detecting, Ultrasonic Radar etc . This is an efficient way to measure small distances precisely. In this project, we have used the **HC-SR04 Ultrasonic Sensor with Arduino** to determine the distance of an obstacle from the sensor. The basic principle of ultrasonic distance measurement is based on ECHO. When sound waves are transmitted in the environment then waves return back to the origin as ECHO after striking on the obstacle. So we only need to calculate the traveling time of both sounds means outgoing time and returning time to origin after striking on the obstacle. As the speed of the sound is known to us, after some calculation we can calculate the distance. We are going to use this same technique for this **Arduino distance measurement** project,

$$d = (2 * v)/t \qquad (1)$$

Where d= distance between sensor and palm

v=speed of sound in air (340m/s) at 20^0 c

t=time taken by the wave to hit obstacle(palm)

and again to reach the sensor.

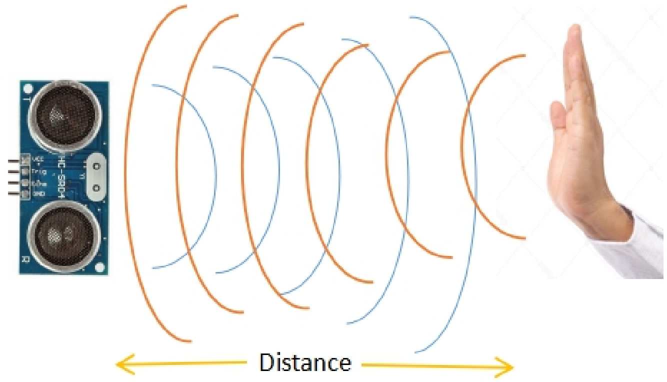


Figure 1.1 Distance calculated by ultrasonic sensor

1.4 Serial communication :

Serial communication is a method used to transfer data between two microcontrollers. This type of communication cannot send data at the same time. It will transfer data alternately. The data sent between microcontrollers will be translated into binary numbers and can be transformed to characters with an ASCII table relief.

In existing research, there are several researches conducting similar experiment, such as communication between programmable logic control (PLC) and arduino. The study explains how to set up the communication between PLC and arduino based on modbus protocol. In this process, PLC was a master and arduino was a slave. Additionally, communication used half-duplex between arduino and PLC. On the other research, a microcontroller based line differential protection using fiber optic communication are employed. It has a focus on the design of a system comprising the microcontroller based line differential protection using fiber optics communication. A working model was designed employing a microcontroller and fiber optic communication for the differential protection of line. In the implemented scheme, received signal was compared with a reference signal using microcontroller . Hardware and software design between microcontroller and computer based on universal serial bus (USB) interface. The study is regarding how to use the USB interface to achieve communication between Sunplus 61A MCU and common personal computer. All the program was written using C++. Furthermore, simplecharacter signal transmission between microcontroller and computer was done. Information could be transmitted successfully using USB protocol.

Microcontroller is a system consist of microprocessor, memory, and input / output devices that are integrated. Therefore, to execute it required a series of programs. The ability of arduino can communicate with each other through serial communication. To control or transfer data for processing arduino on microcontroller, the simplest way is via a serial connection. A serial link is a bidirectional communication process by which information sent over one bit at a time. Usually, to set up a serial link communication utilizes two wires. One will be a transmitter (TX) and the other is a receiver (RX). To connect two Arduino Uno devices in a serial connection the first Arduino Uno’s TX must be connected to the RX of the second Arduino Uno and the RX of the first Arduino Uno to the TX of the second Arduino Uno. To monitor the process with one computer, the program shall to be uploaded alternately one by one before connect the wire. After wires are connected, using the [Serial.print()] function in the first Arduino Uno and it will show up in the serial monitor of the second Arduino Uno and the opposite is also true.

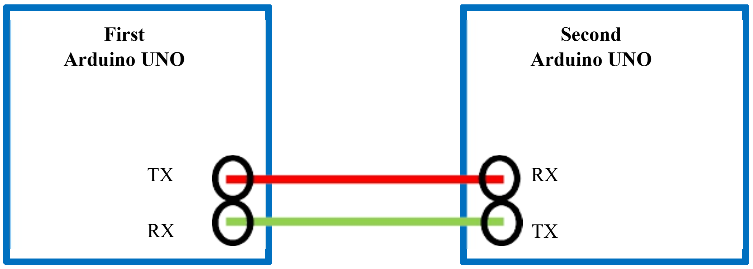


Figure 1.2 Serial link between two Arduino Uno.

1.4.1 ASCII CODE :

There are many classes of ASCII code, such as symbols, letters, numbers, and special characters. Arduino Uno works in binary code. Information is coded using 0s and 1s. Each 0 or 1 is called a bit. Every computer now days uses ASCII code to applied the binary system. It is the standard code that encoded numerically. Generally, the characters between 0 and 31 are not printable (control characters, etc). A 32 is the space character. In addition, there are only 128 ASCII characters. This means, to represent an ASCII character only 7 bits are required. However, since a byte is the smallest size representation on most computers, thus it uses eight bits to represent each letter, number, and symbol. A set of binary code with eight digits can be stored in one byte of Arduino Uno memory. Part of the ASCII code is given in Table .As can be seen, the uppercase letters of A-Z have different a binary number set from lowercase letters of a-z. This pattern indicates that the binary number for each character is not the same, even though with similar letter.

Table 1.1 ASCII CODE

Character	Decimal Number	Binary Number	Character	Decimal Number	Binary Number
[Space]	32	0010 0000	M	77	0100 1101
!	33	0010 0001	N	78	0100 1110
“	34	0010 0010	O	79	0100 1111
#	35	0010 0011	P	80	0101 0000
$	36	0010 0100	Q	81	0101 0001
%	37	0010 0101	R	82	0101 0010
&	38	0010 0110	X	88	0101 1000
‘	39	0010 0111	Y	89	0101 1001
(	40	0010 1000	Z	90	0101 1010
)	41	0010 1001	[	91	0101 1011
*	42	0010 1010	\	92	0101 1100
+	43	0010 1011	]	93	0101 1101
,	44	0010 1100	^	94	0101 1110
-	45	0010 1101	_	95	0101 1111
.	46	0010 1110	`	96	0110 0000
/	47	0010 1111	a	97	0110 0001
0	48	0011 0000	b	98	0110 0010

1	49	0011 0001	c	99	0110 0011
2	50	0011 0010	d	100	0110 0100
3	51	0011 0011	e	101	0110 0101
4	52	ZW0011 0100	f	102	0110 0110
5	53	0011 0101	g	103	0110 0111
6	54	0011 0110	h	104	0110 1000
7	55	0011 0111	i	105	0110 1001
8	56	0011 1000	j	106	0110 1010
9	57	0011 1001	k	107	0110 1011
:	58	0011 1010	l	108	0110 1100
;	59	0011 1011	m	109	0110 1101
<	60	0011 1100	n	110	0110 1110
=	61	0011 1101	o	111	0110 1111
>	62	0011 1110	p	112	0111 0000
?	63	0011 1111	q	113	0111 0001
@	64	0100 0000	r	114	0111 0010
A	65	0100 0001	s	115	0111 0011
B	66	0100 0010	t	116	0111 0100
C	67	0100 0011	u	117	0111 0101
D	68	0100 0100	v	118	0111 0110
E	69	0100 0101	w	119	0111 0111
F	70	0100 0110	x	120	0111 1000
G	71	0100 0111	y	121	0111 1001
H	72	0100 1000	z	122	0111 1010
I	73	0100 1001	{	123	0111 1011
J	74	0100 1010	|	124	0111 1100
K	75	0100 1011	}	125	0111 1101
L	76	0100 1100	~	126	0111 1110

1.4.2 Serial communication working :

The serial function is very important in Arduino because it helps to debug the code .Two pins are used for serial communication pin0 is RX and pin1is TX pin .The Rx pin is use to receive the data from the PC.Tx pin is used to send data to PC. These TX and RX pin in the Arduino are internally connected to TX RX pin of USB hub type B.Baud rate is the speed of transferring data to receiver from transmitter in the form of bits / second. Some of the standard baud rates are 1200, 2400, 4800, 9600, 57600.Framing shows the chunk of bits that transfer from host device to the receiver After selecting the8-bit data chunk, receiver and transmitter must bind to the communication protocols.Transmitter appends synchronization bits (1 start bit and 1 or 2 stop bit) to the original data frame. bits synchronization help the receiver to identify the start and end of the data transfer. This process is known as asynchronous data transfer. Data security loss might happen due to external noise at the receiver end. The only solution to get the stable output is to identify the parity bits.if even number of 1's are present it is known as even parity and the Parity bit is set to 1. If odd number of 1'sare present itis known as odd parity and then parity bit is set to 0.

Asynchronous serial protocols is used for the longer distances and for reliable data transfer.One wire is similar to I2c protocol. the difference is one wired uses single data line and ground. . It provides half duplex communication.thus the commands for the operation of particular task posses serial communication and send to python with certain baud rate and get analyzed in the python interpreter.

1.5 Analysis on two arduino serial communication :

Serial communication speed is not as fast as parallel communication, but serial communication is chosen due to it is more practical in connecting between two Arduino Unos and saving cable. In serial communication, the data bit is sent and then followed by the other data bits on the same path.

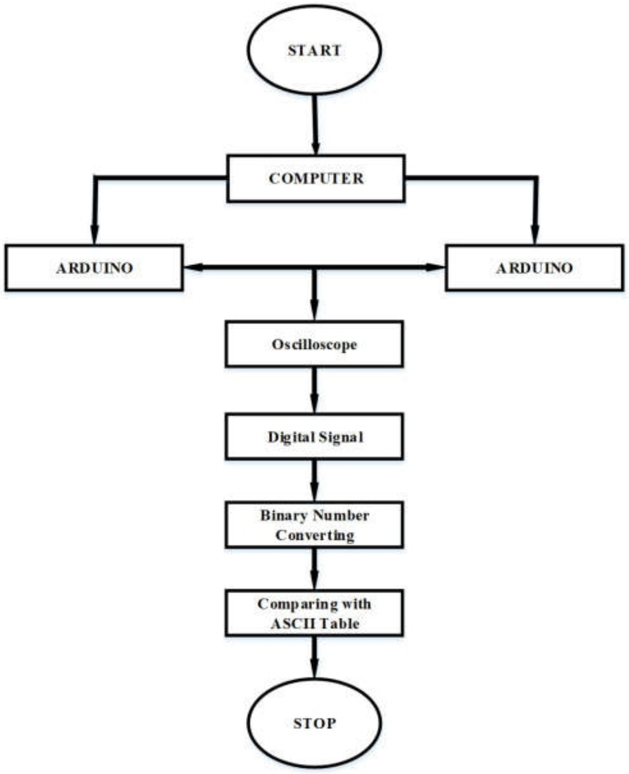


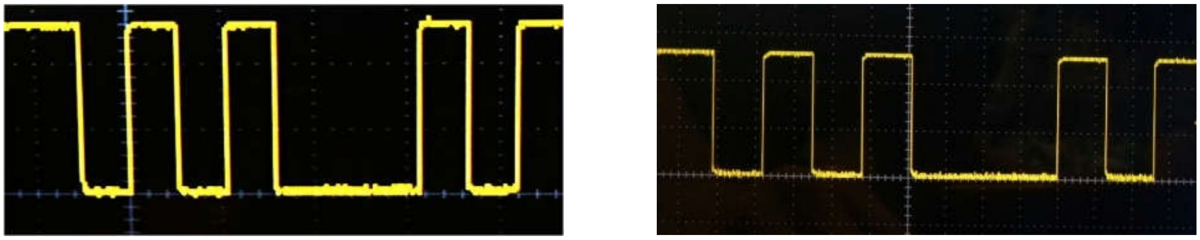
Figure 1.3 Data transfer between two arduinos flowchart.

There are three parameters that build this serial communication, namely baud rate, synchronization bits and frame data. The baud rate used in serial communication is 9600 bits per second (bps). The greater baud rate then the higher data transfer speed, but due to this communication uses electrical signals and data synchronization process, thus the greater in potential for an error would be occurred. Therefore, there is a standard speed for choosing the 9600 bps due to the data sought does not require too fast as a communication speed. The maximum speed in this experiment is recommended not to exceed 115200 bps.To find out the binary number of characters, it can be determined from digital signals obtained. Through oscilloscope of Hantek DSO5120P, the digital signal is assured, as shown in From the encoding process, we yielded the binary number based on digital signal.

In arduino Uno, data is sent via serial communication at 8 bits (1 byte). The synchronization bit consists of the start and stop bits including one packet of data sent. Serial communication has an idle position, specifically 1. The number with the green highlight is called start bit, besides the number with red highlight is called stop bit, as shown in Start bit is indicated by the transition from idle state, which is from 1 to 0, while the stop bit is indicated as the transition back to idle, from 0 to 1.

For data framing, big endian is used since it has a higher speed in string operations and integer data than store strings in the same order. From Table 1.1, we know the binary number of “E” is 01000101. The binary number from encoding processes are inversed from the binary number on ASCII table. Therefore, we discovered the most significant bit (MSB) that was sent first. In other to determine whether the data transfer has been successful or not, we need to compare

The digital signal on the TX Arduino Uno 1 with the digital signal on the RX Arduino Uno 2. From ,we obtain the binary number 01000101 is similar with binary number as result on encoding process for TX on Arduino Uno 1.



(a)

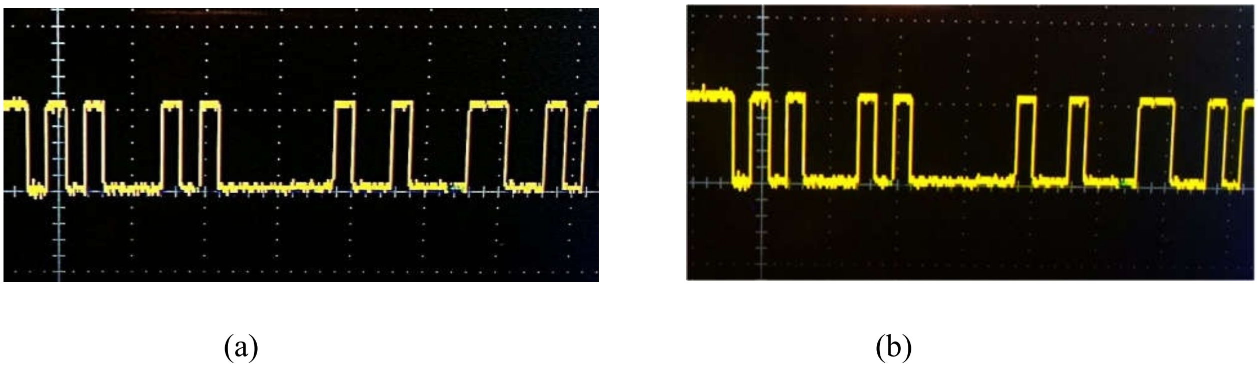
(b)

Figure 1.4 Digital signal for string data of “E” on (a) TX Arduino Uno 1 and (b) RX Arduino Uno 2 from oscilloscope.

To convert binary numbers to decimal number can be used the following method

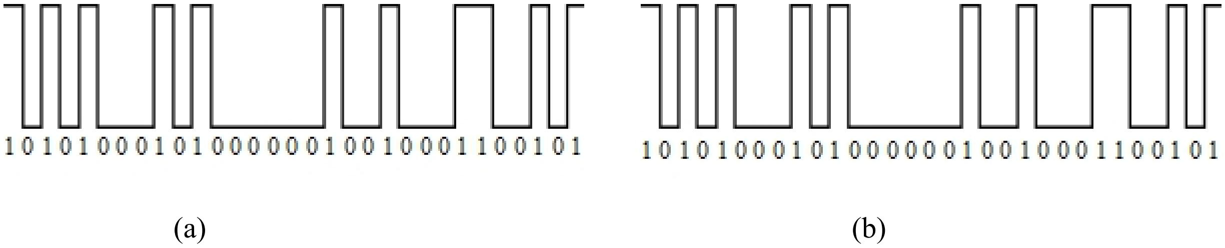
$$\begin{array}{lllllllll}
 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
 & 0x2^7 + & 1x2^6 + & 0x2^5 + & 0x2^4 + & 0x2^3 + & 1x2^2 + & 0x2^1 + & 1x2^0 \\
= & 0 + & 64 + & 0 + & 0 + & 0 + & 4 + & 0 + & 1 \\
= & 69. & & & & & & &
\end{array}$$

The decimal number with a value of 69 is the character of "E".In the second experiment, a string data of ”E L” (E space) is sent to second Arduino Uno from first Arduino Uno. The digital signal obtained from oscilloscope


(a)

(b)

Figure 1.5. Same caption as Fig. 1.4, but for string data of “E L”.



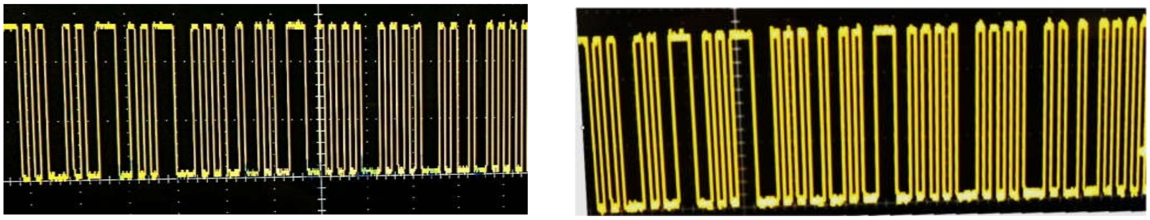
(a)

(b)

Figure 1.6 . Same caption as Fig. 1.5, but for string data of “E L”.

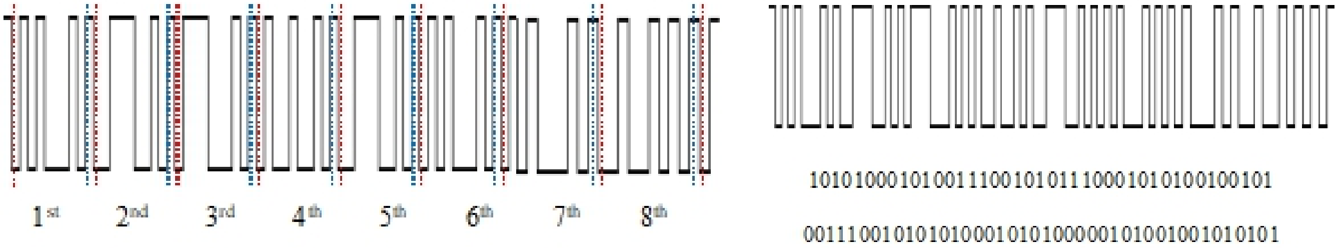
From Fig.1.6, we obtain the digital signal from string data of “E L” on TX Arduino Uno 1 and RX Arduino Uno 2. Both signal had a same form. The first data sent in binary number of 01001100 had a decimal number form of 64 that means “L”. The second data that sent in binary number of 00100000 has a decimal number form of 32 that means “Space”. The last data sent in binary number is 01000101 had a decimal number form of 69 representing “E”.

In the next experiment, we would like to send the "ENGINEER" string data. Moreover, we obtain a digital signal, as shown in Fig.1.7



(a) (b)

Figure 1.7 Same caption as Fig. 1.7, but for string data of “ENGINEER”.



(a) (b)

Figure 1.8 Digital signal for string data of “E” on (a) TX Arduino Uno 1 and (b) RX Arduino Uno 2 for string data of “ENGINEER”

The digital signal should be separated every 8 bit between start bit and stop bit as shown is Fig.1.9.Every digital signal for each character was separated by stop bit and start bit. The blue square indicates the start bit and the red square expresses the stop bit. The first digital signal, it could be described as 10001010 and had decimal value of 69. Therefore, it defined as “E”. The second digital signal presented 00100001 which is defined decimal number of 78 and character of “N”. The third digital signal had binary number as 01000111 and in decimal form it is equal 71 that presented character of “G”. The next digital signal described as 01001001 that is equal to 72 in decimal value and “I” in character. The fourth digital signal had the same digital signal present with the second digital signal. The sixth and seventh digital signal also had the same digital signal with the first digital signal. The last digital signal indicated 01010010 in binary number. It is equal to 82 in decimal and “R” in character. If all parts of digital signal are combined, we find a string data of “ENGINEER”.

1.6 Ultrasonic range finder setup for Serial monitor output:

```
#define trigPin 10
 #define echoPin 13
 void setup()
 {
 Serial.begin (9600);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
 }
 void loop()
 {
float duration, distance;
 digitalWrite(trigPin, LOW);
delayMicroseconds(2);
 digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
 duration = pulseIn(echoPin, HIGH);
distance = (duration / 2) * 0.0344;
if (distance >= 400 || distance <= 2)
{
 Serial.print("Distance = ");
Serial.println("Out of range");
 }
Else
 {
```

```
Serial.print("Distance = ");

Serial.print(distance);

Serial.println(" cm");

 delay(500);

}

delay(500);

 }
```

1.6.1 Explanation of the Code:

- **Line 11:** Declares the variables duration and distance.
- **Lines 12 and 13:** Sends a 2 µs LOW signal to the trigPin to make sure it’s turned off at the beginning of the program loop.
- **Lines 15-17:** Sends a 10 µs HIGH signal to the trigPin to initiate the sequence of eight 40 KHz ultrasonic pulses sent from the transmitting transducer.
- **Line 19:** Defines the duration variable as the length (in µs) of any HIGH input signal detected at the echoPin. The Echo pin output is equal to the time it takes the emitted ultrasonic pulse to travel to the object and back to the sensor.
- **Line 20:** Defines the distance variable as the duration (time in d = s x t) multiplied by the speed of sound converted from meters per second to centimeters per µs (0.0344 cm/µs).
- **Lines 22-24:** If the distance is greater than or equal to 400 cm, or less than or equal to 2 cm, display “Distance = Out of range” on the serial monitor.
- **Lines 26-30:** If the distance measurement is not out of range, display the distance calculated in line 20 on the serial monitor for 500 ms.

II. ULTRASONIC SENSORS

Today's the developing world shows various adventures in every field. In each field the small requirements are very essential to develop big calculations. By using different sources we can modify it as our requirements and implement in various field. In earlier days the measurements are generally occur through measuring devices. But now a day's digitalization as is on height. Therefore we use a proper display unit for measurement of distance. We can use sources such as sound waves which are known as ultrasonic waves using ultrasonic sensors and convert this sound wave for the measurement of various units such as distance, speed. This technique of distance measurement using ultrasonic in air includes continuous pulse echo method, a burst of pulse is sent for transmission medium and is reflected by an object kept at specific distance. The time taken for the sound wave to propogate from transmitter to receiver is proportional to the distance of the object. In this distance measurement system we had ultrasonic sensor HC-SR04 interfaced with arduino UnoR3.

2.1 INTRODUCTION :

The study of ultrasonics is the investigation of the effects of propagation, interaction with matter, and the application of a particular form of energy—sound waves—at frequencies above the limits of human perception. Ultrasonics includes the basic science of the energy–matter interaction, the associated technologies for generation and detection, and an increasingly diverse range of applications, which are now encountered in almost every field of engineering, many of the sciences and in medicine.

Ultrasonics, which is a specific branch of acoustics, deals with vibratory waves in solids, liquids, and gases at frequencies above those within the hearing range of the average person, that is, at frequencies above 16 kHz (16,000 cycles per second). Often, the hearing range of a young person extends above 20 kHz, so the setting of the lower limit for the ultrasonic range is somewhat arbitrary. The acoustic frequency scale covers the range from below 15 Hz to above 1 THz (103 GHz)

2.2 TRANSDUCERS :

Ultrasonic transducers convert AC into ultrasound, as well as the reverse. Ultrasonics, typically refers to piezoelectric transducers or capacitive transducers. Piezoelectric crystals change size and shape when a voltage is applied; AC voltage makes them oscillate at the same frequency and produce ultrasonic sound. Capacitive transducers use electrostatic fields between a conductive diaphragm and a backing plate.

The beam pattern of a transducer can be determined by the active transducer area and shape, the ultrasound wavelength, and the sound velocity of the propagation medium. The diagrams show the sound fields of an unfocused and a focusing ultrasonic transducer in water, plainly at differing energy levels.

Since piezoelectric materials generate a voltage when force is applied to them, they can also work as ultrasonic detectors. Some systems use separate transmitters and receivers, while others combine both functions into a single piezoelectric transceiver.Ultrasound transmitters can also use non-piezoelectric principles. such as magnetostriction. Materials with this property change size slightly when exposed to a magnetic field, and make practical transducers.A capacitor ("condenser") microphone has a thin diaphragm that responds to ultrasound waves. Changes in the electric field between the diaphragm and a closely spaced backing plate convert sound signals to electric currents, which can be amplified.

The diaphragm (or membrane) principle is also used in the relatively new micro-machined ultrasonic transducers (MUTs). These devices are fabricated using silicon micro-machining technology (MEMS technology), which is particularly useful for the fabrication of transducer arrays. The vibration of the diaphragm may be measured or induced electronically using the capacitance between the diaphragm and a closely spaced backing plate (CMUT), or by adding a thin layer of piezo-electric material on diaphragm (PMUT). Alternatively, recent research showed that the vibration of the diaphragm may be measured by a tiny optical ring resonator integrated inside the diaphragm (OMUS). Ultrasonic Transducers are also used in acoustic levitation.

2.2.1 PIEZOELECTRIC ULTRASONIC TRANSDUCERS

As the emitter and receiver of ultrasonic vibrations, ultrasonic transducers are used. The main structural element of tis transducers are piezoelectric elements [2]. Piezoelectric materials (piezoelectrics) have a domain structure. Their behavior in an external electric field can be compared with the reaction of ferromagnets to a magnetic field. The unit cell of a piezoelectric crystal at temperatures above the critical (Curie point) has a symmetrical shape. If the temperature is below critical, then the cell is distorted towards one of the edges. As a result, the distances between positive and negatively charged ions, i.e. a dipole moment arises. The domain structure leads to repeated repetition of this effect within the macro region. When a strong external electric field is applied, the dielectric becomes polarized, leading to priority orientation of the dipole moments of all domains in the field. To facilitate the process of domain orientation, polarization is usually carried out at a temperature slightly below the Curie temperature. With subsequent cooling, the ordered state remains stable. Thus, the piezoelectric element is a dielectric sample polarized in a given direction, on two opposite faces of which there is a surface bound charge of opposite signs. Mechanical tension or compression acting on the piezoelectric element in the direction of polarization leads to deformation of all cells, a change in the dipole moment of the sample as a whole, and, consequently, a change in the surface bound charge. This is a direct piezoelectric effect. If, instead of mechanical action, a polarized sample is placed in an external electric field, the sizes of the unit cells will change in the same direction, which will lead to a change in size along the entire field of the sample. This means a change in the total dipole moment (polarization), and, consequently, the surface bound charge. This is the inverse piezoelectric effect. The principle of action of an ultrasonic emitter is based on the inverse piezoelectric effect, which consists in the mechanical deformation of a crystal under the influence of an electric field. The ultrasound receiver uses the direct piezoelectric effect when the surface bound charge changes as a result of mechanical deformation. Structurally, the piezoelectric element is a piezoceramic with deposited electrodes. The natural frequency f of the plate is determined by its thickness:

$$\boldsymbol{f = \upsilon/2d}$$

where υ is the speed of sound in a substance, m/s *d is* the plate thickness.

On the market there are ultrasonic emitters with a frequency of ≥ 20 kHz. When the ultrasonic wave propagates in air, we set υ=330 m/s. Then the possible wavelengths are $\lambda \leq 16.5$ mm. The wavelength uniquely determines the minimum size of the emitter and the minimum distance between the source and receiver. So, for a frequency of 25 kHz, the diameter of the emitter is 16 mm. The sensitivity range based on such rangefinder transducers (UM0008, HCSR04, TS601, etc.) starts from 0.2 m.The transducer field is divided into two zones: near and far. The near zone is the area in front of the emitter, where series of amplitude maxima and minima are observed. To create a rangefinder, it is necessary to ensure workability in the far zone. Optimal for this application will be emitters with a frequency of 40 – 60 kHz. In the process of choosing an ultrasonic transducer, the following pairs were used, including a receiver and a transmitter: AW8R40-10OA00 / AW8T40-10OA00, AW8R40-12OA01 / AW8T40-12OA01 (manufacturer AUDIOWELL), TCT40-10R1 / TCT40-10T1 (manufacturer MEC) and UCE10T-40R- WA / UCE10T-40T-WA. All of them have a resonant frequency of 40 kHz (wavelength 8.25 mm). The transducer specifications also include the following characteristics: sound pressure level SPL, sensitivity, bandwidth, divergence angle, capacitance, maximum applied voltage, operating temperatures. Sound pressure – alternating overpressure that occurs in an elastic medium when a sound wave passes through it.The sound pressure level of 100 dB corresponds to an overpressure of at least p = 2 Pa, 110 dB – 6 Pa, 115 dB – 11 Pa. The emitter specification provides the sound pressure level determined at a power of 1 W at a distance of 1 m. Similarly, the sensitivity of the receiver is determined. In this case, 0 dB corresponds to the response of the 1V receiver to a pressure of 0.1 Pa. Thus, the −≥ 65 dB sensitivity means that at an overpressure of 0.1 Pa. Features of translating the sensitivity of receivers using special scales are considered in . The width of the frequency band and the angle of discrepancy are determined by halving the radiated power (– 6 dB).

2.2.2 Ultrasonic Measuring Principles with Piezo Elements

Piezo ultrasonic sensors offer high precision and reliability over large measuring ranges as well as long-term stability and are compact. They do not require optical transparency. Basically, a distinction is made between two measuring principles:

1. Runtime measuring :

The piezoceramic element serves as transmitter and receiver during runtime measuring whether measuring the gap, detecting objects or for **Measuring Flow.**

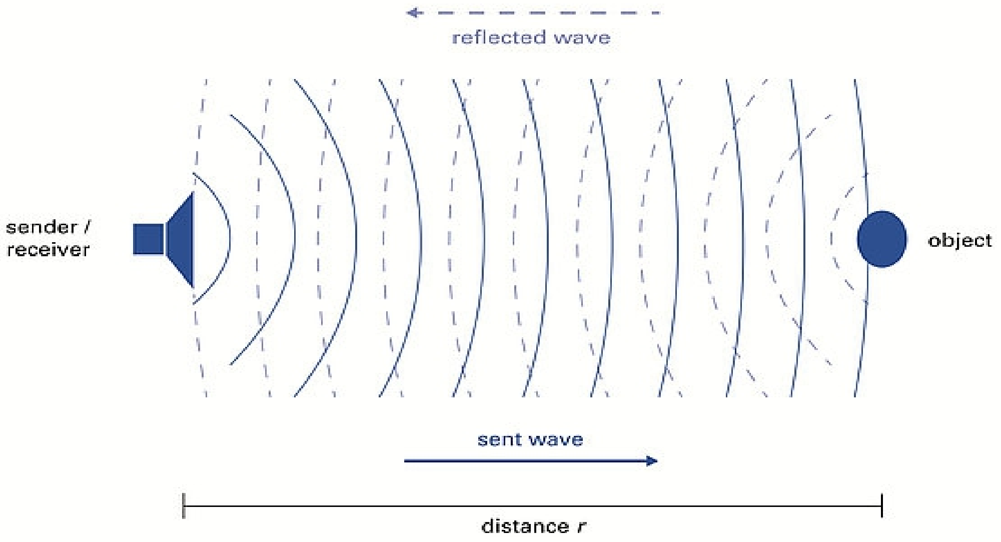


Figure 2.1 Runtime measuring principle

The piezoceramic element emits an ultrasonic pulse. The sound waves triggered by this propagate and and then hit an object. They are then reflected and partially absorbed. The same piezo element receives the reflected waves. The transmit time difference Δ*t* between emitting and receiving the sound waves provides information on the distance *r* between the sound source and the object. Once the speed of the sound *c* in the surrounding medium is known, it is possible to calculate the gap *r*:

$$2 \cdot r = c \cdot \Delta t$$

2. Doppler effect

The principle of the Doppler effect is used to measure the flow rates or flow velocities of contaminated media, e.g., suspended particles or air bubbles. After emitting an ultrasonic pulse, the ultrasonic waves (f_0) are scattered or reflected by liquid particles. The resulting shift of frequency Δ*f* between the reflected wavefront that was emitted and received by the same piezo transducer is proportional to the flow velocity *v_S* of the particles. The angle θ between the direction of the emitted ultrasonic pulse and the measuring path must be taken into account:

$$\Delta f = \frac{2f}{c} \cdot v_S \cdot cos\theta$$

The direction of flow can also be determined by the frequency change. When the liquid particles approach the sensor, the wavelength of the sound shortens and the frequency increases (f_b) because the sound waves are pushed in front of the particles and compresses them. Conversely, the wavelength increases and the frequency of the sound decreases as the particles move away from the sensor (f_a). This frequency change Δf of the sound waves can be detected and compared with the sound frequency of the emitted ultrasonic pulse.

Applications for this include building services engineering for determining the consumption of water or heating energy as well as in the medical field for recording blood flow velocity and direction.

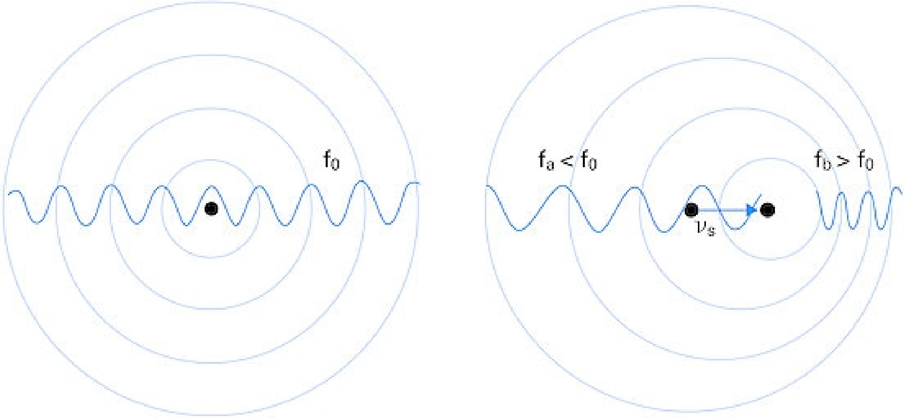


Figure 2.2 Doppler effect

Principle of the Doppler effect: a) Sound waves propagate around the stationary transmitter, b) higher or lower frequencies can be detected depending on the position of the observer when the transmitter is in motion.

2.2.3 Applications

- Process automation and industrial measuring technology, e.g :- **measuring gaps and levels**, **measuring flowrate and detecting bubbles**
- Nondestructive testing

- Medical imaging
- Materials processing with high-performance ultrasound, e.g., welding, drilling, cutting
- **Ultrasonic cleaning in industry**
- **Shockwave lithotripsy** and **aerosol production** in medical technology
- **Sonar technology and hydroacoustics**

2.3 TYPES OF ULTRASONIC SENSORS

The following diagrams summarize the distinctions between proximity and ranging ultrasonic sensors:

Proximity Detection :

An object passing anywhere within the preset range will be detected and generate an output signal. The detect point is independent of target size, material, or degree of reflectivity.

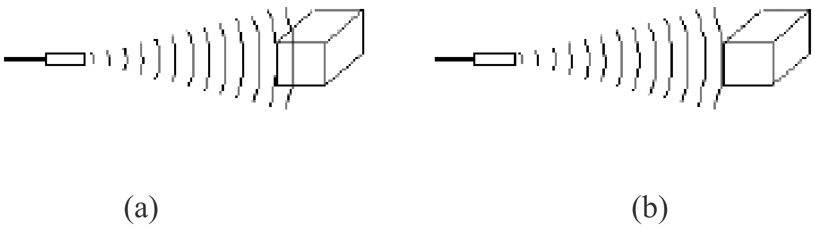

(a) (b)

Figure 2.3 (a) Object Detected , (b) Object Not Detected

Ranging Measurement:

Precise distance(s) of an object moving to and from the sensor are measured via time intervals between transmitted and reflected bursts of ultrasonic sound. The example shows a target detected at six inches from sensor and moving to 10 inches. The distance change is continuously calculated and outputted.

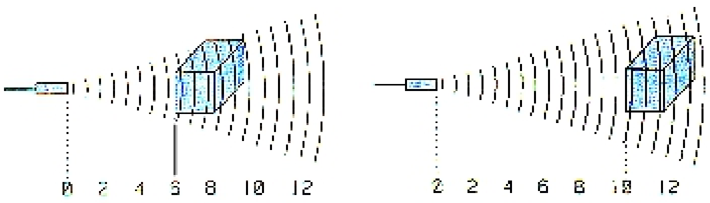


Figure 2.4 Ranging Measurement

Target Angle :

This term refers to the "tilt response" limitations of a given sensor. Since ultrasonic sound waves reflect off the target object, target angles indicate acceptable amounts of tilt for a given sensor. If an application requires a target angle beyond the capabilities of a single sensor, two sensors can be teamed up to provide an even broader angle of tilt.

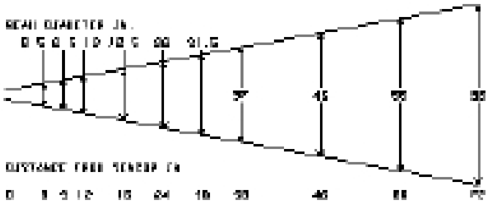


Figure 2.5 Target Angle

Beam Spread:

This term is defined as the area in which a round wand will be sensed if passed through the target area. This is the maximum spreading of the ultrasonic sound as it leaves the transducer.

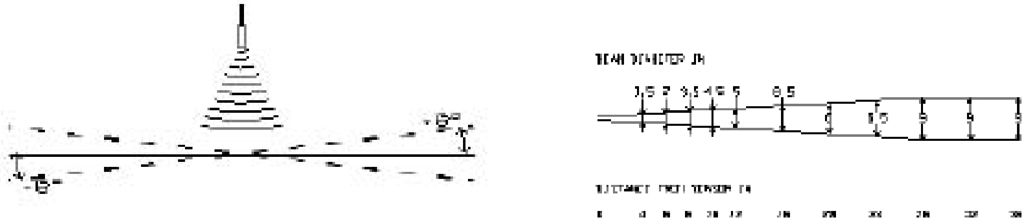


Figure 2.6 Beam Spread

Distance Hysteresis Option :

Some standard proximity sensors can be furnished with a "hysteresis control" capability. Many hysteresis sensors have a fixed detect point equal to the sensor's minimum range, and an adjustable turn off point up to the sensor's maximum range . Consult factory for hysteresis option.

Thru-Beam Option:

The TBT/TBR-600-40 is primarily a Thru-Beam sensor consisting of one transmit and one receive transducer, in separate and self-contained housings.

High Gain Option:

The RPS-300 and 325 can be furnished with a high gain capability. This is denoted by the letter (G) in the part number. (Example: RPS-300G-14)

Analog Output Option:

Analog outputs of 0-10 VDC and 4-20 mA are available in a variety of models. See the selection card and appropriate data sheet for details. Note: In some models, analog output is an option indicated by adding (-500) to the part number. (Example: RP S-100-14-500).

Analog Ranging Card:

For converting proximity sensors into RANGING sensors.When connected to any RPS-100/300/325/326 proximity sensor, an RPS-500 ANALOG RANGING CARD measures discrete distances to a target within a selected range, then outputs one or more signals to a controller or computer.

2.4 HC-SR04 ULTRASONIC SENSOR

- HC-SR04 is an ultrasonic sensor mainly used to determine the distance of the target object.
- It measures accurate distance using a non-contact technology - A technology that involves no physical contact between sensor and object.
- Transmitter and receiver are two main parts of the sensor where former converts an electrical signal to ultrasonic waves while later converts that ultrasonic signals back to electrical signals.
- These ultrasonic waves are nothing but sound signals that can be measured and displayed at the receiving end.
- It gives precise measurement details and comes with accuracy (resolution) around 3mm, terming there might be a slight difference in the calculated distance from the object and the actual distance.

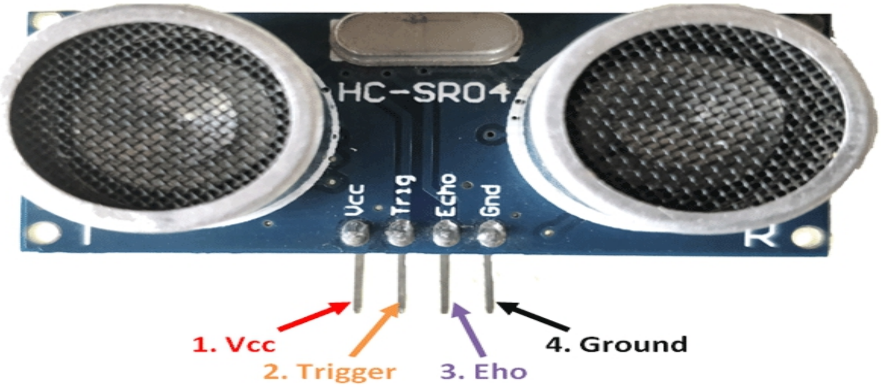


Figure 2.7 Ultrasonic sensor module HC SR04

2.4.1 HC-SR04 PIN CONFIGURATION :

This sensor includes four pins and the pin configuration of this sensor is discussed below.

- **Pin1 (Vcc):** This pin provides a +5V power supply to the sensor.
- **Pin2 (Trigger):** This is an input pin, used to initialize measurement by transmitting ultrasonic waves by keeping this pin high for 10us.
- **Pin3 (Echo):** This is an output pin, which goes high for a specific time period and it will be equivalent to the duration of the time for the wave to return back to the sensor.
- **Pin4 (Ground):** This is a GND pin used to connect to the GND of the system

2.4.2 HC-SR04 Ultrasonic Sensor with Arduino Board

Here we are giving an example for the HC-SR04 ultrasonic sensor using the Arduino board. This sensor is interfaced with an Arduino board.

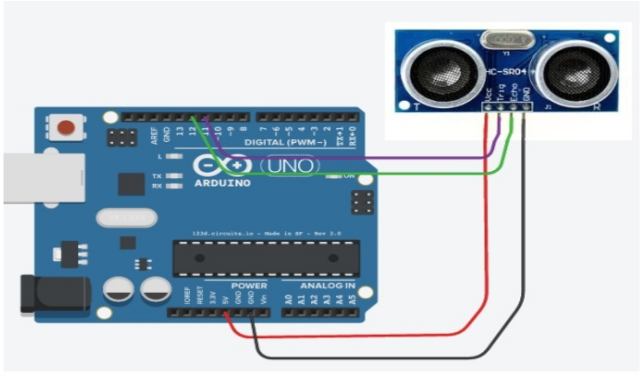


Figure 2.8 Ultrasonic-sensor-with-Arduino-board

The required components of this project mainly include the Arduino UNO board, HC-SR04 ultrasonic sensor, breadboard, and jumper wires. The connections of this project are very simple like the following.

- The VCC pin of the sensor is connected to 5V of the Arduino
- The Trig pin of the sensor is connected to Pin 11 in the Arduino
- The Echo pin of the sensor is connected to Pin 12 in the Arduino
- The GND pin of the sensor is connected to GND pin in the Arduino

2.4.3 How to use the HC-SR04 Ultrasonic Sensor :

HC-SR04 distance sensor is commonly used with both microcontroller and microprocessor platforms like Arduino, ARM, PIC, Raspberry Pie etc. The following guide is universally since it has to be followed irrespective of the type of computational device used.

Power the Sensor using a regulated +5V through the Vcc ad Ground pins of the sensor. The current consumed by the sensor is less than 15mA and hence can be directly powered by the on board 5V pins (If available). The Trigger and the Echo pins are both I/O pins and hence they can be connected to I/O pins of the microcontroller. To start the measurement, the trigger pin has to be made high for 10uS and then turned off. This action will trigger an ultrasonic wave at frequency of 40Hz from the transmitter and the receiver will wait for the wave to return. Once the wave is returned after it getting reflected by any object the Echo pin goes high for a particular amount of time which will be equal to the time taken for the wave to return back to the sensor.

The amount of time during which the Echo pin stays high is measured by the MCU/MPU as it gives the information about the time taken for the wave to return back to the Sensor.

2.4.4 FEATURES

The **features of the HC-SR04 sensor** include the following

- The power supply used for this sensor is +5V DC
- Dimension is 45mm x 20mm x 15mm
- Quiescent current used for this sensor is <2mA
- The input pulse width of trigger is10uS
- Operating current is 15mA
- Measuring angle is 30 degrees

- The distance range is 2cm to 800 cm
- Resolution is 0.3 cm
- Effectual Angle is <15°
- Operating frequency range is 40Hz
- Accuracy is 3mm

2.4.5 APPLICATIONS

The **applications of HC-SR04 sensor** include the following,

- This sensor is used to measure speed as well as the direction between two objects
- It is used in wireless charging
- Medical ultrasonography
- This is used to detect objects & avoid obstacles using robots such as biped, pathfinding, obstacle avoidance, etc.
- Depth measurement
- Humidifiers
- This sensor is used to plot the objects nearby the sensor by revolving it
- Non-destructive testing
- By using this sensor depth of pits, wells can be measured by transmitting the waves through water.
- Embedded system
- Burglar alarms

2.4.6 LIMITATIONS

In terms of accuracy and overall usefulness, HC-SR04 ultrasonic distance sensor is really great, especially compared to other low-cost distance detection sensors. That doesn’t mean that the HC-SR04 sensor is capable of measuring “everything”. Following diagrams shows a few situations that the HC-SR04 is not designed to measure

a) The distance between the sensor and the object/obstacle is more than 13 feet.

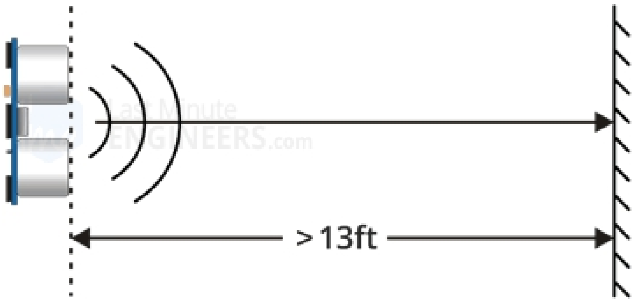


Figure 2.9 object and the sensor schematics

b) The object has its reflective surface at a shallow angle so that sound will not be reflected back towards the sensor.

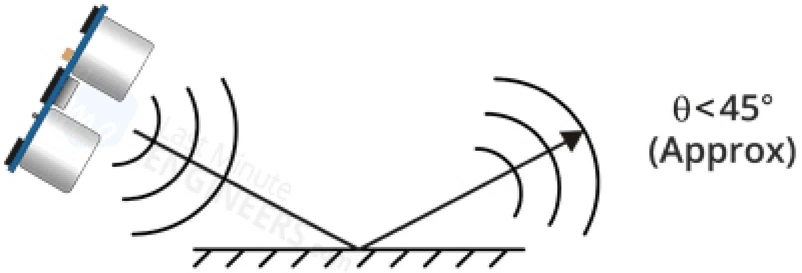


Figure 2.10 object and the sensor at reflection schematics

c) The object is too small to reflect enough sound back to the sensor. In addition, if your HC-SR04 sensor is mounted low on your device, you may detect sound reflecting off of the floor.

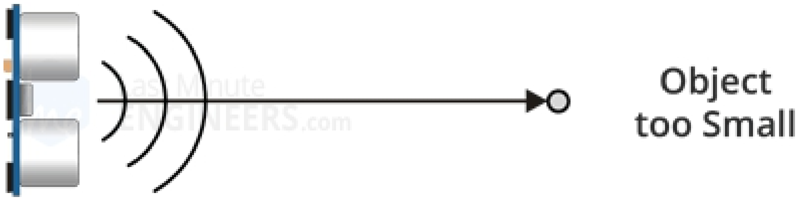


Figure 2.11 condition when object is too small for sensor

d) While experimenting with the sensor, we discovered that some objects with soft, irregular surfaces (such as stuffed animals) absorb rather than reflect sound and therefore can be difficult for the HC-SR04 sensor to detect.

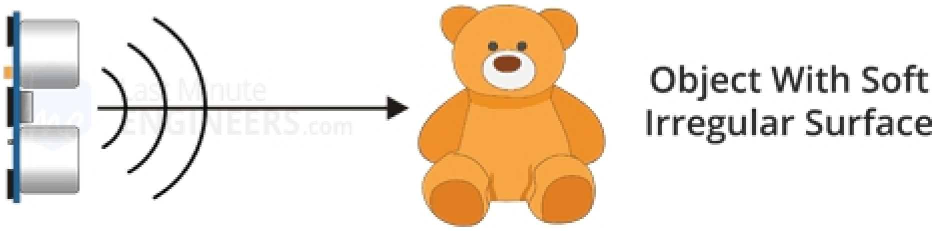


Figure 2.12 condition when object is having irregular surface

2.5 Timing Diagram of Ultrasonic Sensor :

In order to generate the ultrasound you need to set the Trig on a High State for 10 µs. That will send out an 8 cycle sonic burst which will travel at the speed sound and it will be received in the Echo pin. The Echo pin will output the time in microseconds the sound wave traveled.

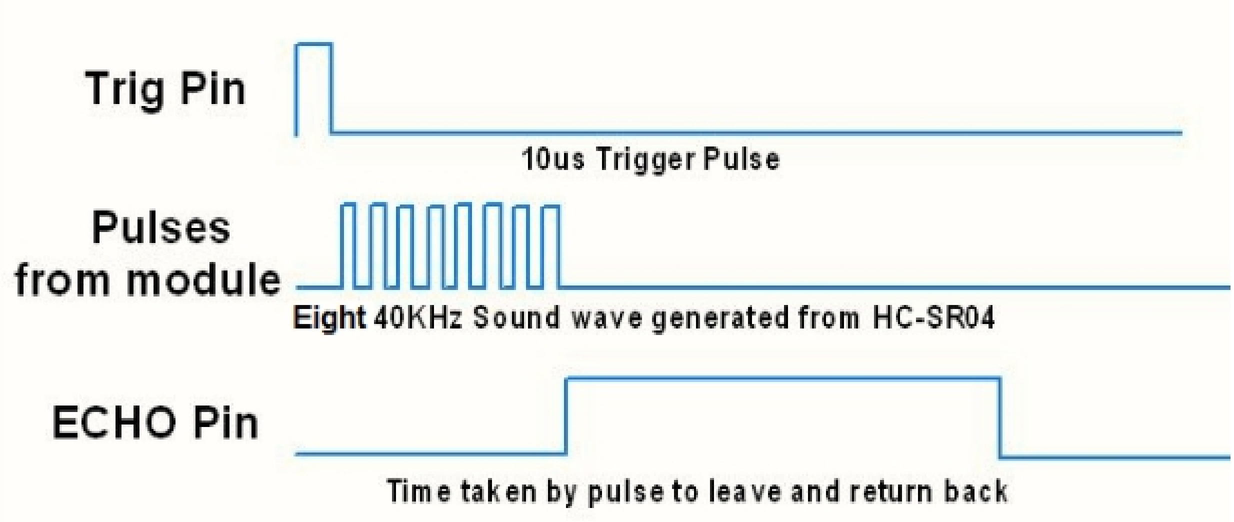

Trig Pin
10us Trigger Pulse
Pulses from module
Eight 40KHz Sound wave generated from HC-SR04
ECHO Pin
Time taken by pulse to leave and return back

Figure 2.13 Timing Diagram of Ultrasonic sensor

1. First, need to transmit trigger pulse of at least 10 us to the HC-SR04 Trig Pin.
2. Then the HC-SR04 automatically sends Eight 40 kHz sound wave and wait for rising edge output at Echo pin.
3. When the rising edge capture occurs at Echo pin, start the Timer and wait for a falling edge on Echo pin.
4. As soon as the falling edge captures at the Echo pin, read the count of the Timer. This time count is the time required by the sensor to detect an object and return back from an object.

2.6 Applications of Ultrasonic Sensors

Migatron has been using advanced technology to solve difficult sensing and control problems for nearly 40 years across a broad range of of industries. With Ultrasonic Sensing's unique advantages over conventional sensors and the rapidly increasing range of applications, ultrasonic sensors are becoming widely accepted as an industry standard across the board.

Tank Level :

Liquid level sensors are integral to process control and inventory management in many industries.At Migatron, we engineer two types of level sensors, point level sensors (proximity sensors) and continuous level sensors (analog sensors). The type of sensor appropriate for your liquid level measurement depends on the application.

Production Line Sensors:

Ultrasonic Sensors can be applied to the manufacturing process for automated process control on the factory floor while also being an indispensable tool for companies to maximize efficiency through precise measurement and control.Ultrasonic sensors can streamline the production processes.

Distance Measurement

Ultrasonic sensors can measure the distance to a wide range of objects regardless of shape, color or surface texture. They are also able to measure an approaching or receding object.

Some of the many ultrasonic sensor applications which use Migatron sensors are :

- Robotic sensing
- Stacking height control
- Loop control
- Liquid level control
- Full detection
- Counting people/people detection
- Presence detection
- Detecting breaks in threads or wires
- Box sorting
- Contouring or profiling

2.7 Advantages of Ultrasonic Sensors:

These are a few ultrasonic sensor advantages that help clarify what applications our sensors are suited for.Not affected by color or transparency of objects.Ultrasonic sensors reflect sound off of objects, so the color or transparency have no effect on the sensor’s reading.Can be used in dark environments.Unlike proximity sensors using light or cameras, dark environments have no effect on an ultrasonic sensor’s detection ability.

Low-cost option :

Our sensors start at $29.95. They come fully calibrated and ready to use. We strive to give a low cost, high-quality product suited for specific needs.

Not highly affected by dust, dirt, or high-moisture environments :

Although our sensors work well in these environments, they can still give incorrect readings with a heavy build-up of dirt or water, especially in extreme conditions. However, our **SCXL-MaxSonar-WR** line is self-cleaning and can help decrease the effects of things like condensation.

Some more advantages are :

- They have greater accuracy than many other methods at measuring thickness and distance to a parallel surface
- Their high frequency, sensitivity, and penetrating power make it easy to detect external or deep objects
- Our **SCXL- MaxSonar- WR** Product line is self-cleaning. Which allows for continuous running and less downtime
- Our ultrasonic sensors are easy to use and not dangerous during operation to nearby objects, people or equipment
- Our sensors easily interface with microcontrollers or any type of controller

2.8 Limitations of Ultrasonic Sensors

Although we fully believe in the capability of our sensors, we understand that ultrasonics are not suited for every application. Below we go into the *limitations of our sensors*, and how we have overcome some of these problems.

Cannot work in a vacuum :

Because ultrasonic sensors operate using sound, they are completely nonfunctional in a vacuum as there is no air for the sound to travel through.

Not designed for underwater use :

Our sensors have not been properly tested in this environment, so underwater use voids our warranty. This being said, we do supply documentation for customers who would still like to test our sensors underwater. If you are interested in underwater applications with ultrasonics, check out our articles on **Water Depth Sensing with Ultrasonics** and **Underwater Ranging** for more information.

Sensing accuracy affected by soft materials :

Objects covered in a very soft fabric absorb more sound waves making it hard for the sensor to see the target.

Sensing accuracy affected by changes in temperature of 5-10 degrees or more :

Although this is true, we have a variety of temperature compensated sensors available that either calibrate upon start-up or before every range reading depending on the sensor model. During this time is when the sensor will calibrate with any change in temperature, voltage, etc. This dramatically decreases this problem.

Have a limited detection range :

At the moment, our **longest range sensors** have a maximum range of 10 meters, now our cargo sensor detects up to 16.5m. While this is a disadvantage in certain applications, our sensors have great mid-range capabilities and are still suited for many applications.

III. Artificial intelligence-based voice assistant

Voice control is a major growing feature that changes the way people can live. The voice assistant is commonly being used in smartphones and laptops. AI-based voice assistants are operating systems that can recognize the human voice and respond via integrated voices. This voice assistant will gather the audio from the microphone and then convert that into text, later it is sent through GTTS (Google text to speech). GTTS engine will convert text into an audio file in English language, then that audio is played using play sound package of python programming Language

3.1 INTRODUCTION :

In recent times only in the Virtual Assistants, we can experience the major changes, the way a user interacts, and the experience of the user. We are already using them for many tasks like switching on/off lights, playing music through streaming apps like Wynk Music, Spotify, etc., This is the new method of interacting with the technical devices that makes lexical communication as a new ally to this technology.The concept of virtual assistants in earlier days is to describe the professionals who provide ancillary services on the web.

The job of a voice is defined in three stages: Text to speech; Text to Intention; Intention to action; Voice assistant will be fully developed to improve the current range.[6] Voice assistants are not befuddled with virtual assistants, which are people, who work casually and can therefore handle all kinds of tasks. Voice Assistants anticipate our every need and it takes action, Thanks to AI-based Voice Assistants.

AI-based voice assistants can be useful in many fields such as IT Helpdesk, Home automation, HR-related tasks, voice-based search, etc., and the voice-based search is going to be the future for next-generation people where users are all most dependent on voice assistants for every need. In this proposal, we have built an AI-based voice assistant which can do all of these tasks without the inconvenience

3.2 BACKGROUND

3.2.1 History of Voice Assistants :

If we come to the history of Voice Assistants the first voice-activated to be released is ‘radio rex’ in the year 1911. The Foundation of smart virtual assistants was laid by IBM Simon in the year 1994 as we know it today. Digital speech recognition technology has become an aspect of personal computers in the 90s with Microsoft, Apple, Philips, etc., After various researches, Siri was introduced as the first modern digital Voice Assistant as the feature in iPhone 4S in 2011. Many companies have used oral dialog systems to design such system devices like Amazon Alexa, Microsoft Cortana, apple's Siri, Google Assistant, etc. General dialogue systems have six components which include Voice recognition, voice language apprehension, dialog manager, natural language generation, text to speech converter, and knowledge base.

3.2.2 Future Applications :

In the future voice, assistants can be used for two developments: First quality of dialogue recognition will increase because broadband allows more complex data processing in powerful data centres. Second, from the user's perspective, VAs aid for interaction. In the companies, voice assistants can be used to automate repetitive tasks, for example, Amazon's Alexa can open video conferencing and book meeting rooms, etc

3.2.3. Aim of this Study :

The main aim of our project is that we have created a function, Intelligent Personal Assistant which can perform mental tasks like turning on/off smartphone applications with the help of Voice User interface (VUI) which is used to listen and process audio commands.

3.3 PROPOSED DESIGN :

The project will give a fair knowledge about the intelligent assistant which is capable of understanding the commands given by the user. Our assistant can easily understand the commands given by the user through vocal media and responds as required. Our assistant performs the most frequently asked requests from the user and makes their task easier. Our voice assistant listens to the command given by the user through the microphone. After listening it will say "done listening" and displays what the users said and acts accordingly.

In our project, we have installed gTTS engine package to make the voice assistant speak like a normal human being. we have defined a function called 'voice assistant speak' as explained. The gTTS will analyze the command given by the user through the microphone and search in the browser the required response and convert that response into text.

```
tts = gTTS(text=audio_string, lang='en')
```

gTTS is basically used to convert the audio string into text. This audio string is nothing but the response which the voice assistant is supposed to give the user. The language of the text is chosen to be English, the code for English is 'en'. We save this entire function into tts. We are saving this text, that is the audio file with the '.mp3' extension. Each audio file is given a random number from 1 to 20000000. The random number can be generated using the command 'random.randint()'.This whole '.mp3' extension file is saved under the name 'audio file' .Finally to save this audio file we have used the command as mentioned.

tts. save(audio file)

This command saves the audio file in the system. (ex-'audio24854.mp3').

3.4 TASKS PERFORMED BY THE VOICE ASSISTANT

1) Can remember any person's name till the usage session.

2) Voice assistant names can also be changed unlike in other voice assistants.

3) Play/download a song or video from YouTube. When the user asked 'can you play/download me a song', 'play movie' the assistant opens YouTube and plays the required content for the user or download the requested video/song.

4) Searches anything from google and tells the required content. If asked 'google search' the assistant searches the content asked from google and opens the required content in the browser.

5) Opens the maps and tells the exact location the user asked for. When asked for 'find the location' or 'google maps' the assistant ask for the location the user wants and opens the google maps and highlight the location user asked for.

6) Tells the accurate weather of the location the user asks for. When asked for 'current weather in' the assistant tells the exact weather of the desired location of both maximum and minimum in degree Celsius.

7) Takes a screenshot of the display. When asked for "capture", "capture my screen", "my screen", "screenshot", "take screenshot", the assistant captures the display the user is using and stores it in the path specified.

8) Gives the live news around the world. When asked 'news for today',' tell me the news, 'what's the news, 'news', the assistant reads the first 5 updated news headlines from the website.

9) Can able to tell whether the password has been hacked or not.

10) If any person is in danger, our voice assistant can able to send the user's location to the police or the relatives by giving the command as "I'm in danger"

11) It sends a mail to the username specified by the user. when told "send mail," the assistant asks to whom the mail has to be sent and it will send a mail according to that.

12) It can translate the words the user speaks into any language and displays the words of that language which is specified by the user.

13) Can able to shut down or restart the system by just user command etc.

These are some features we have added to our AI-based voice assistant as of now and we are working on many more features to embed into this assistant

3.5 METHODOLOGY

Voice assistants are all written in programming languages, which listen to the verbal commands and respond according to the user's requests. In this project, we have used Python Programming language to build the AI-based Voice assistant. A user can say "play me a song" or "open facebook.com" the voice assistant will respond with the results by playing that particular song or by opening the Facebook website. The Voice assistant waits for a pause to know that users have finished their request, then the voice assistant sends users' requests to its database to search for the request.

1) The request asked by the user gets split into separate commands so that our voice assistant can able to understand.

2) Once within the commands list, our request is searched and compared with the other requests.

3) The commands list then sends these commands back to the Voice assistant.

4)Once the voice assistant receives those commands, then it knows what to do next.

5) The voice assistant would even ask a question if the request is not clear enough to process it, in other words, to make sure it understands what we would like to receive.

6) If it thinks, it understands enough to process it, the voice assistant will perform the task which the user has asked for.

3.5.1 working of voice assistant:

The voice recognition actually works whenever the user asks the voice assistant what is the date and time. Then It will analyze the user's question and answers according to it. For example, let's Consider two persons standing each other. When one person speaks his vocal cords present in his throat will create a voice. The voice created by the first-person voice will reach the second person ear. This voice will be in the form of signals (analog signal) that travels through the air where air act as the medium. The fundamental concept which happened in human beings same will get repeated in any devices for voice recognition system between humans and the machines . Voice recognition is actually introduced because In older days desktops are introduced only for textual-based writings. When the user needs to send some larger texts to other persons through the mail it became a tedious job. so then people came up with a solution to this problem by using a voice assistant. Speech recognition will convert the user audio into text automatically. Speech recognition algorithm captures the user voice by activating the microphone. most of the PCs are embedded with a microphone. These microphones will pick the user command in individual words. These individual words will get matched with

the patterns that present in the dictionary (as in Fig. 3.1) which were already saved on the desktop. let's consider an example if the user says HELLO it will take every word and it will match with the pattern present in the dictionary, the algorithm works by matching the indexes, first it will go to the H index, search about it and take the hello word and place it. Basically, the algorithm which was used is the same replicated in the olden days for speech recognition.

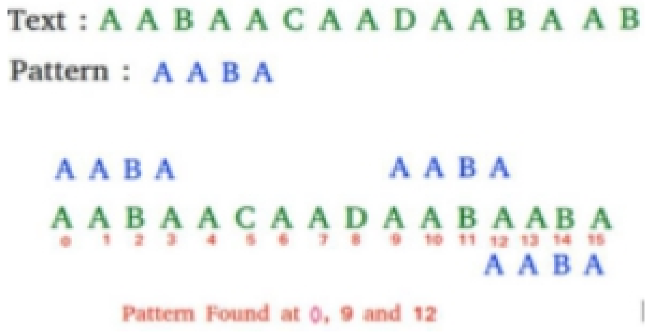

Text : A A B A A C A A D A A B A A B
Pattern : A A B A
A A B A
A A B A
A A B A A C A A D A A B A A B A
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
A A B A
Pattern Found at 0, 9 and 12

Figure 3.1 Olden days method matching the word with the pattern present in the dictionary

Later with the advent of the internet amount of data that people want to process or the number of questions that they want to ask to devices has started increasing day by day. The information that the people want from the devices through speech recognition systems cannot be stored on the desktops. for Example what is the recent trending news? This is an instant question .but all the current information or current happenings cannot be stored on a desktop. That is why the concept of data centers came into existence. Integration of data centers with the voice recognition system results in the concept of a virtual assistant. They are many virtual assistants is present till today. But they should always be connected to the internet because PC can access the data from the data center only through the internet. In order to understand how these tasks are carried out, it is necessary to know the basic block diagram (as in Fig 3.2) and working of voice Assistance.

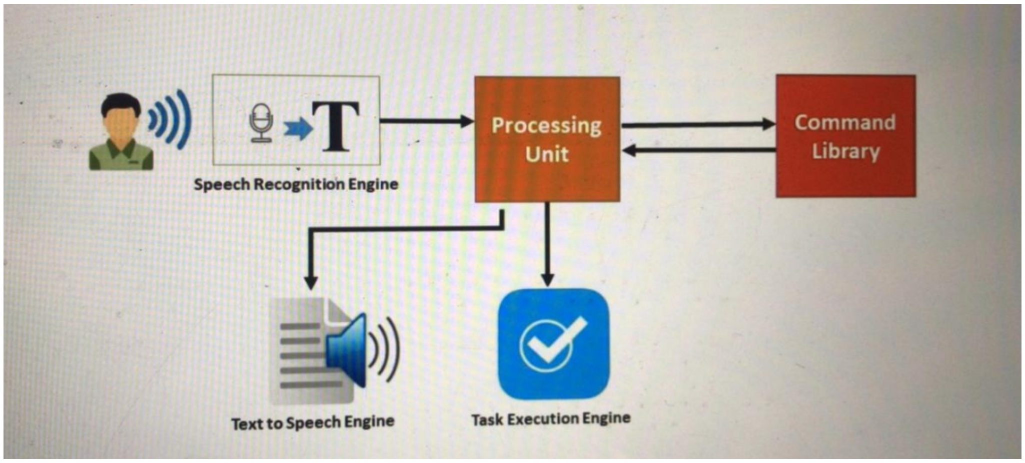


Figure 3.2   Block diagram represents the working of voice assistant

A. **Voice command**:This represents user voice i.e task given to the machine by the humans .

B**. Speech recognition**:The sound is generally transmitted in form of waves.

But the personal computer can only understand binary language thus, it is necessary to convert analog signal to digital form. As the sound waves are 1D signal waves.The analog voltage values of the signal changes at every point of time based on height of the wave.To convert analog form of wave into digital forms,the height of the wave is recorded sampling process will be carried out.

C.Sampling:

The process of converting continous time signal into its equivalent discrete time signal is called sampling .It takes thousand words a record for recording a number representing the height of sound wave at particular point of time.The digital.signals are broken into pieces after recording then these pieces of information is campared with 'phonemes' the smallest words which are present in the database so that it will automatically correct the mis spelt word.

For example:'HELLO'

In the above example the word 'HELLO' the heights of the sound wave at differnt point of times are -1274,1160,-986

Now they are converted into digital signal in binary form and then these information is broken into piece for further process

D. Command library:

This consists of inbuilt library function with which the speech recognition engine compares the given text .If suppose the text is “PLAY” then it will check whether it us present in command library or not if the word is present then the further process is carried out by speech recognition engine

E. Task execution engine:

It will perform the task after receiving the response from speech recognition engine.now the analog is converted into digital form and the binary number then converted into audio by using text to speech engine

For example: we would like to know the present whether condition , a voice command is given “what is the whether ?”. The speech recognition engine capture our voice and converts it into text . The text will be transferred to processing unit.Now the processing unit gets the text and compare against command library to check whether the command is valid (or) not if suppose the command is valid the processing unit will perform associated task by using task and execution engine and finally output is produced from execution engine.Then the particular details get converted into audio using text to speech engine and can listen current weather details

3.6 THE VOICE-ACTIVATED DIGITAL ASSISTANTS AS PARTICIPANTS OF COMMUNICATION

Interaction with voice assistants displays a significant difference from interpersonal communication. This is due to the fact that, as a software product created by a person, the voice assistant, despite the possibilities of self-training, is limited by its functionality created by programmers, architects, designers etc. As a part of the study, the voice assistants’ market can be divided into:

1) assistants integrated into products produced by specific companies – Alisa, Siri, Google Assistant, Alexa.

2) voice bots for support services – Megafon, Beeline, VTB.

3) personal banking assistant – Tinkoff, Sberbank.

4)) ecosystem-integrated assistants – Yandex Dialogues

In natural communication a person has many resources to identify the interlocutor. In interaction with voice assistants, the interlocutor is reduced to a voice (in some cases, to a voice and an accompanying text, which either duplicates information or is a response to a user's request). Interaction with such voice assistants as Siri and Alisa is characterized by imitation of everyday conversation using the methods and forms inherent in interpersonal communication (flirting, jokes, insults). The efficiency of such communication depends solely on the goals of the person and his/her psychological and emotional state, as well as the neural network proficiency. Interacting with voice assistants, a person realizes himself/herself as "ego", and the program – as "the other". But this "other" is not a person, he/she is not capable of representing his/her own "ego", consequently, this communication is purely individualized by the discursive strategies of a person interacting with a voice assistant. For example, the query "weather in Samara" and "Alisa, please, tell me, what is the weather like today in Samara?" reveals the identity of a speaker. The interface does not care how the question is asked.Within a context of a person's communication with voice assistants as brand intermediaries, the latter act as both "subject of speaking" and "subject of listening". In this type of communication, the phenomenon of "understanding" comes to the fore. For a person, understanding is existential in nature; the main function of communication is realized via it. For voice assistants, understanding is a utilitarian functionality, which determines receiving a satisfying result by the person. T.Z. Adamyants notes that there are two mechanisms of understanding in communication. The first involves collecting new information or promoting one's own meanings based on the interpreted material. The task in this case is not to comprehend the true intention of an author. For perceiving consciousness, the creation of a new particular meaning is important. The second method is associated with a desire to realize and comprehend the true intention of an author, taking into account his/her perceived goals and motives, which are not always perceived. In the perceiving consciousness, both mechanisms can be involved simultaneously. The voice assistants have limited access only to the first mechanism of understanding. The problem of teaching voice assistants to understand refers to the areas of natural language modeling, modeling of self-awareness, understanding in automatic text processing, recognizing emotions and speech and mastering creative thinking. Each of these aspects gives rise to many other problems, the solution of which lies in the study of human intelligence.

3.6.1 METHODS OF RESEARCH

The main research methods in this work are the survey and diary method. The survey involved 237 people, segmented by gender and age. The answers were completely anonymous and the survey structure included 15 questions. The questionnaire was designed in such a way that the negative answer to the security question "Have you ever used voice assistants?" did not complete the survey, but opened a branch of additional ones that allowed minimizing the error of false negative answers due to consumer ignorance. The key questions also referred to additional contacts with brands through voice assistants and satisfaction with communication. The respondents were asked to determine the effectiveness of interaction independently. In addition, the diary research method involved 10 people, who recorded all interactions with the voice assistants within 7 days. 50% of the respondents submitted their reports and agreed to go through a final interview. In addition to the survey and the diary method, the authors of the paper conducted an indepth interview with the developers of a neural network involved in modeling a voice based on emotional coloring. The study adopted an axiom, according to which any interaction with voice assistants is communication with the brand.

3.7 Typology

3.7.1 Speaker Dependent speech Recognition System:

A speaker dependent system is one which operates upon the voice command of a single person. The computer is trained to recognize the speech and voice quality of a particular individual. The voice qualities diversify from human to human. To build an artificial intelligence system that could recognize anybody's voice, is not an easy task. The system would require a very large database which is again a difficult task to accomplish. Therefore, speaker dependent voice recognition systems are developed comparatively with greater ease, are economically more feasible and are simple and reliable

3.7.2 Speaker Independent speech Recognition System:

Even though the voice quality differs from to a great extent from one speaker to another, speaker independent voice recognition systems can be used by any individual and can perform upon any voice. In these kinds of systems, user doesn't needs to train the system . Therefore speaker independent voice recognition systems are less cost effective and more complicated. Also, they possess limited vocabularies.

3.8 voice assistant :

A typical voice assistant (VA) system consists of three main sub-systems: voice capture, speech recognition, and command execution, as illustrated in below Fig a). The voice capture subsystem records ambient voices, which are amplified, filtered, and digitized, before being passed into the speech recognition subsystem. Then, the raw captured digital signals are first pre-processed to remove frequencies that are beyond the audible sound range and to discard signal segments that contain sounds too weak to be identified. Next, the processed signals enter the speech recognition system. Typically, a speech recognition (SR) system works in two phases: activation and recognition. During the activation phase, the system cannot accept arbitrary voice inputs, but it waits to be activated. To activate the system, a user has to either say pre-defined wake words or press a special button. For instance, Amazon echo takes "Alexa" as the activation wake word. Apple Siri can be activated by pressing and holding the home button for about one second or by “Hey Siri” if the ‘Allow Hey Siri’ feature is enabled. To recognize the wake words, the microphones continue recording ambient sounds until a voice is collected. Then, the systems will use either speaker-dependent or speakerindependent speech recognition algorithm to recognize the voice. For instance, the Amazon Echo exploits speakerindependent algorithms and accepts “Alexa” spoken by any one as long as the voice is clear and loud. In comparison, Apple Siri is speaker dependent. Siri requires to be trained by a user and only accepts “Hey Siri” from the same person. Once activated, the SR system enters the recognition phase and will typically use speaker-independent algorithms to convert voices into texts, i.e., commands in our cases. Note that a speaker-dependent SR is typically performed locally and a speaker-independent SR is performed via a cloud service. To use the cloud service, the processed signals are sent to the servers, which will extract features (typically Mel-frequency cepstral coefficients) and recognize commands via machine learning algorithms (e.g., the Hidden Markov Models or neural networks). Finally, the commands are sent back. Given a recognized command, the command execution system will launch the corresponding application or execute an operation. The acceptable commands and corresponding actions are system dependent and defined beforehand. Popular voice assistants have been built on smartphones, wearable devices, smart home devices, and automobiles. Smartphones allow users to perform a wide range of operation via voice commands, such as dialing a phone number, sending short messages, opening a webpage, setting the phone to the airplane mode, etc. Modern automobiles accept an elaborate set of voice commands to activate and control a few in-car features, such as navigation, the entertainment system, the environmental controls, and mobile phones. For instance, if “Call 1234567890” is recognized, an automobile or a smartphone may start dialing the phone number 1234567890. Many security studies on voice assistants focus on attacking either the speech recognition algorithms or command execution environment (e.g., malware). This paper aims at the voice capturing subsystem, which will be detailed in the next subsection

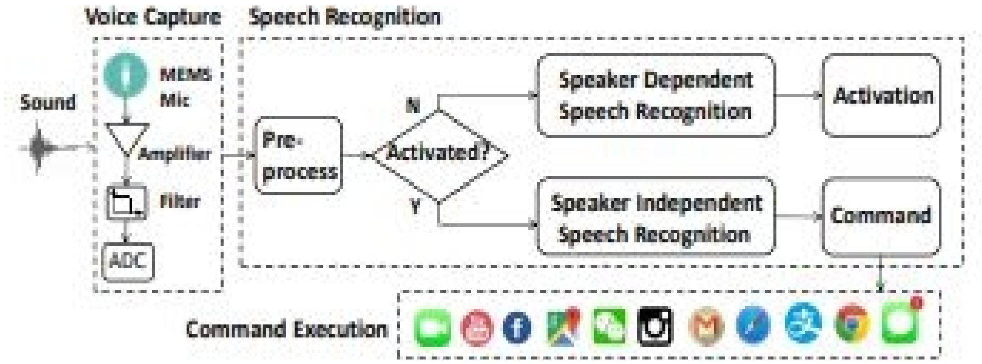


Figure 3.3 An illustration on the modulated tone traversing the signal pathway of a voice capture device in terms of FFT

3.8.1 Microphone:

A voice capture subsystem records audible sounds mainly by a microphone, which is a transducer that converts airborne acoustic waves (i.e., sounds) to electrical signals. A majority of microphones are condenser microphones, and two types of condenser microphones are used on voice controllable devices: electret condenser microphones (ECMs) and microelectromechanical system (MEMS) microphones. Due to the miniature package sizes, lower power consumption and excellent temperature characteristics, MEMS microphones dominate mobile devices, including smartphones and wearables. Nevertheless, ECMs and MEMS microphones work similarly. As shown in Fig. b), condenser microphones are air-gapped capacitors that contain a movable membrane and a fixed electrode (electret for ECMs). In the presence of a sound wave, the air pressure caused by the sound wave reaches the membrane, which flexes in response to changes in air pressure, while the other electrode remains stationary. The movement of the membrane creates a change in the amount of capacitance between the membrane and the fixed electrode. Since a nearly constant charge is maintained on the capacitor, the capacitance changes will produce an AC signal. In this way, air pressure is converted into an electrical signal. Designed to capture audible sounds, microphones, lowpass filters (LPFs), and ADC in the voice capture subsystem are all designed to suppress signals out of the frequency range of audible sounds (i.e., 20 Hz to 20 kHz). According to datasheets, the sensitivity spectrum of microphones is between 20 Hz to 20 kHz. Ideally, even if a signal higher than 20 kHz is recorded by a microphone, it will be removed by the LPF. Finally, the sample rate of the ADC is typically 44.1 kHz, and the digitized signal's frequency is limited below 22 kHz according to the Nyquist Sampling Theorem.

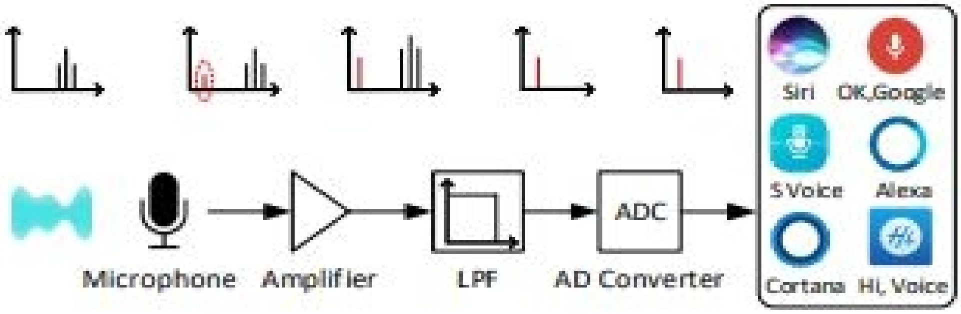


Figure 3.4 An illustration on the modulated tone traversing the signal pathway of a voice capture device in terms of FFT

3.8.2 Voice Commands Generation:

Siri works in two phases: activation and recognition. It requires activation before accepting voice commands, and thus we generate two types of voice commands: activation commands and general control commands. To control a VA, DolphinAttack has to generate activation commands before injecting general control commands.

3.8.3 Activation Commands Generation :

A successful activation command has to satisfy two requirements: (a) containing the wake words "Hey Siri", and (b) toning to the specific voice of the user that was trained for Siri. We design two methods to generate activation commands for two scenarios respectively: (a) an attacker cannot identify the owner of Siri (e.g., attacking random users), and (b) an attacker can obtain a few recordings of the owner's voice. (1) TTS-Based Brute Force. The recent advance in Textto-Speech (TTS) technology makes it easy to convert texts to voices. The only challenge of generating activation commands with TTS is how to match the timbre of TTS voice with that of a human user. We observed that two users with similar vocal tones can activate the other's Siri. Thus, as long as there is one TTS voice that is close enough to the owner's, it suffices to activate Siri. In DolphinAttack, we prepare a set of activation commands in various tones and timbres with the help of existing TTS systems (summarized in Tab. 1), which include Selvy Speech, Baidu, Google, etc. In total, we obtain 90 types of TTS voices. (2) Concatenative Synthesis. When an attacker can record a few recordings from the owner of the Siri which do not include "Hey Siri", we propose to synthesize a desired voice command by searching for relevant phonemes (i.e., HH, EY, S, IH, R) from other words in the available recordings

3.8.4 General Control Commands Generation :

General control commands can be any commands that launch applications (e.g., "call 911", "open google.com") or configure the devices (e.g., "turn on airplane mode"). Unlike the activation commands, SR systems generally do not authenticate the control commands. Thus, an attacker can choose the text of any control commands and utilize TTS systems to generate them.

3.8.5 Evaluation :

We test both activation and control commands. Without loss of generality, we generate both activation and control commands with the TTS systems summarized in Tab. 1. In particular, we prepare two voice commands: "Hey Siri" and "Call 1234567890". For activation, we use the voices from the Google TTS system to train Siri, and the rest for testing. We play the voice commands with an iPhone 6 Plus and the benchtop devices in Fig, and test on an iPhone 4S. The results of both activation and recognition commands are summarized in Tab. 1, which show that the control commands from all of the TTS systems can be recognized by the SR system, and 35 out of 89 types of activation commands can activate Siri, resulting in a success rate of 39 percent. The results are similar when Siri is trained by a human user.

3.9 Overview of speech Recognition:

The major processes of speech recognition include feature extraction, acoustic modeling, pronunciation modeling and decoder. The end user gets through the application by means of an applicable input device such as a microphone. Sound waves travel in form of analog signal thus the recognizer first accepts them as analog signal and converts them into digital signal .speech signal then takes form of electrical signals. Feature extraction eliminates various sources of information; it eliminates periodicity of pitch, amplitude of excitation signal and fundamental frequency etc. by combining and optimizing information, decoder performs the actual decision regard recognition of a speech utterance. Figure a) shows the block diagram of speech recognition process.

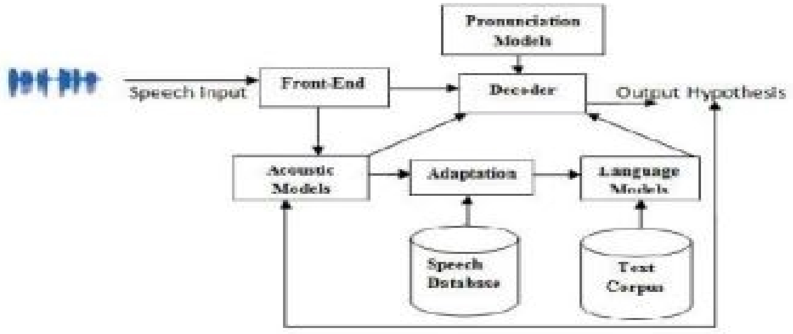


Figure 3.5 Outline of speech recognition system

3.9.1 Feature Extraction:

In automatic speech recognition system, feature extraction is the process by means of which unwanted and redundant information is discarded while retaining the useful information at the same time.The process also involves transformation of signal into an appropriate form for the models used for classification. Feature extraction involves conversion of signals into digital form (known as signal conditioning), measurement of energy or frequency response of signals (known as signal measurement), signal parameterization and forming observation vectors.Various techniques could be used to extract features. The most popular feature extraction technique being used is Mel Frequency Cepstrum Coefficient (MFCC) while other significant techniques are Perceptual Linear Prediction (PLP) and LPC (Linear Predictive Codes).

3.9.2 Mel Frequency Cepstrum Coefficient (MFCC):

The cepstrum may be defined as spectrum of a spectrum. A spectrum provides information regarding the frequency components of a signal whereas a cepstrum gives information about how those frequency changes. Framing, windowing, DFTH, Mel filter bank algorithm, computation of inverse DFT are the basic computational steps of MFCC . The most common application of MFCC is their property by virtue of which system automatically recognizes the numbers spoken into a telephone; also it is used in speaker recognition which is the task of recognizing different speaker from their voices

Perceptual Linear Prediction (PLP): PLP is based on short term spectrum of speech. In feature extraction process, PLP model represents the Physcophysics of human auditory in a much more accurate manner. The method goes vulnerable when frequency response of communication channel modifies the short term spectral values. Through the frequency band, PLP claims to avoid uniform weighting

Linear Predictive Code (LPC): At low bit rates, linear predictive coding is assumed to be one of the most powerful speech analysis technique which provides extremely accurate parameters of speech. LPC assumes that speech signal is produced by means of a buzzer at the end of a tube (known as voiced sounds).Space between the vocal folds (known as glottis) is responsible for buzz production. The tube is formed by vocal tracts (the mouth and throat). Resonance in vocal tracts produces enhanced frequency bands in sound produced. The LPC family of methods finds their uses from the basic telephony to military communications. Some sample based music synthesizers make use of LPC algorithm for compression of waveform ROM.

Decoding: In speech recognition process, decoding may be viewed as crucial process. It is done in order to determine the sequence of words that would match with the acoustic signal. Feature vectors are responsible for generation of the acoustic signal . The essential information sources required for decoding process are

1) An acoustic model with an HMM for each unit.

2) dictionary or a list of words and the phoneme sequences they consist of

3) language model consisting of word sequence.

3.9.3 Acoustic Modeling (Hidden Markov Model):

Acoustic model is implemented using Hidden Markov Models (HMM's). HMM is a tool that is used for representation of probability distributions over a sequence of observations. On underlying principles of

Markov process, HMM is a double stochastic model. The semantics of the model is encapsulated in hidden part i.e. the sequence of states is hidden from observer and he could view only the output symbol sequence and therefore the model is known to be as Hidden Markov Model. In the process of Automatic Speech Recognition, HMM is used to imitate a word, where phoneme represents the hidden part of the word and the statistical characteristics of corresponding acoustic events are accounted by the observable part in the given feature space. HMM can be mathematically expressed as

- A set S of N states, S = {S1, S2, S3….Sn} that are discrete and definite values. It is also assumed that hidden stochastic process could occur .

- An basic state probability division,

pi={DL(Ej | t=0),Ej ɈE}

where t is called as discrete time index.

- Another probability division which allows transition between the states, where our transition probabilities are time independent t

ajk={D(Ek=Rt+1|Ej=Rt)}

- A set of output probability distributions that gives statistical properties of the model:

cj(x)={D(ot=vx|Rt=i)}.

3.9.4 Language Modeling:

Language models are used for probability estimation of each hypothesized sequence of words. This probability is useful in guiding the search towards linguistically probable.sequence of words as they are assigned higher probabilities as compared to unlikely sequence of words. The acoustic score is combined with language model score so as to determine how probable the hypothesized sequence of word. pronunciation modeling: A pronunciation model links acoustic model and the language model. This model compares the words generated by acoustic model with the words present in dictionary and produce’s string of words which is our final output. It contains information about which words are known to system or how they may be pronounced (known as phonetic representation)

3.9.5 Speech Recognition Forges Ahead:

Human beings have been able to recognize the voice of other human's since ages when languages just began. Soon an era would arrive when we could speak directly to our mobiles, tablets and laptop and could instruct them. Then it would send a mail on your voice commands or would format what you said to what you need-a report, a letter, a memo or whatever. Currently three research teams of Defense Advanced Research Projects Agency (DARPA) are functioning on Global Autonomous Language Exploitation (GALE), this program is alleged to accept streams of information from newscast and newspapers of foreign countries and translate them.DARPA claims that the software could quickly translate two languages with an expected accuracy of 90 percent [10].Sony, IBM, Olympus, Lernout, Norcom, Dragon Systems, Phillips, and Grover Industries are a few names that are following the thrill of voice recognition by developing software and portable devices with higher feats of accuracy

3.9.6 Battle of Digital Life Assistants:

Without a footnote of Apple's Siri, Google Now, Microsoft Cortana and Samsung's S Voice discussion on digital life assistant could never end. They are voice respondent systems for mobile applications. These systems make use of traditional Natural Language Processing and Speech Recognition System . However if State System could be applied to these applications response time and user experience could be enhanced to much greater extent. The culture was started by Apple's Siri for ios devices, Google Now soon accompanied next with almost same ingredient for android platform and Microsoft Cortana has just joined the race for windows 8.1 Phone users. All these services are web based-speech recognition gets invoked as soon we talk to the application. The words are then parsed in the form of queries which are sent to some web based service and answer is displayed or spoken out by the device

3.9.7. HOW OUR VOICE ASSISTANT IS USEFUL :

In this section we explain a set of situational events and we can tell how our voice assistant is useful.for eg if we want to go to a particular location instead of opening google maps and typing the destination takes a lot of time instead of that with the help of our voice assistant just by command to find the particular location it opens the map and highlight the particular location.In the modern voice assistant playing a song on youtube is just a feature,but they can't download it.for eg if you want to download the song "give me some sunshine",in other voice assistant we can't download it but in our voice assistant ,the song is searched in the youtube database and video ID is noted ,with this the video is downloaded.a

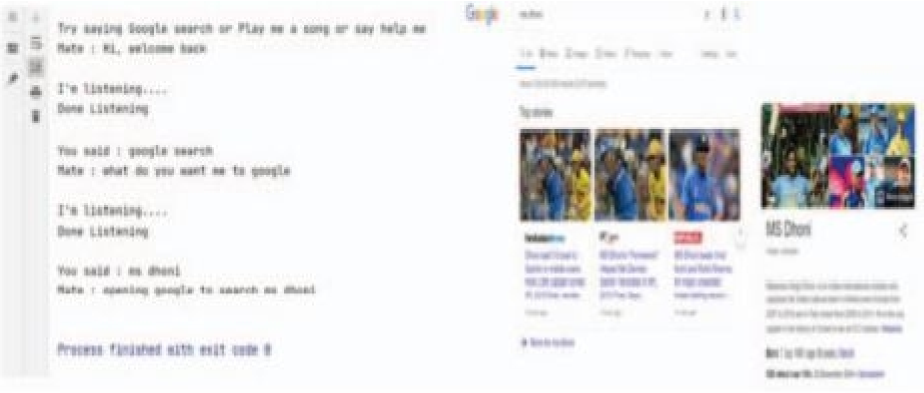
If we want to get any information,we have to open google and search for it.But in voice assistant by just giving it in the form of the command for example “what are super computers”?it collects the best information available on the wikipedia and gives us and if we want to know about what is happening in the world by giving this command “what are the top five news of the day?”it grabs the information and tells us .from this we can gain knowledge .

3.9.8 RESULTS :

The required packages of Python programming language has been installed and the code was implemented using PyCharm Integrated development environment (IDE) and the python code we have developed runs in both Python 2.7 and Python 3.x, and below are the few outputs which we have received in our AI-based voice assistant.

A) Google Search Output :

As shown in below fig 3.6 when we ask the voice assistant to search ‘MS DHONI”it receives the request and performs the action by searching google.

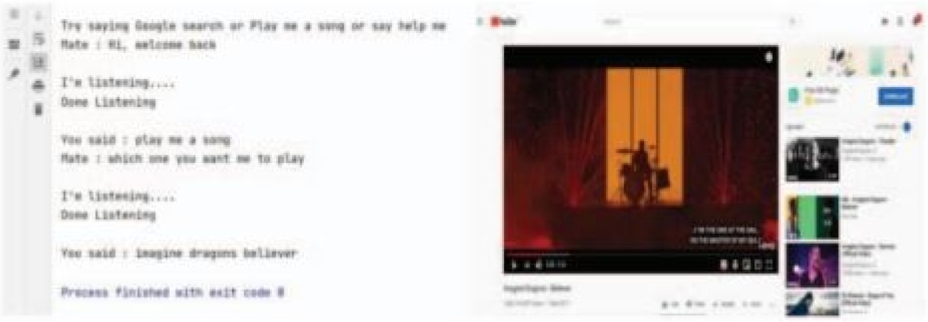


```
Try saying Google search or Play me a song or say help me
Mate : Hi, welcome back

I'm listening....
Done Listening

You said : google search
Mate : what do you want me to google

I'm listening....
Done Listening

You said : ms dhoni
Mate : opening google to search ms dhoni


Process finished with exit code 0
```

Fig 3.6 output screen of performing google search

B) Playing song/video on YouTube

As shown in below fig 3.7 when we ask the voice assistant to “play me a song “it responds by saying “which song you want me to play” and we ask the voice assistant a particular song/video then the voice assistant will perform the task by playing song/video on youtube .

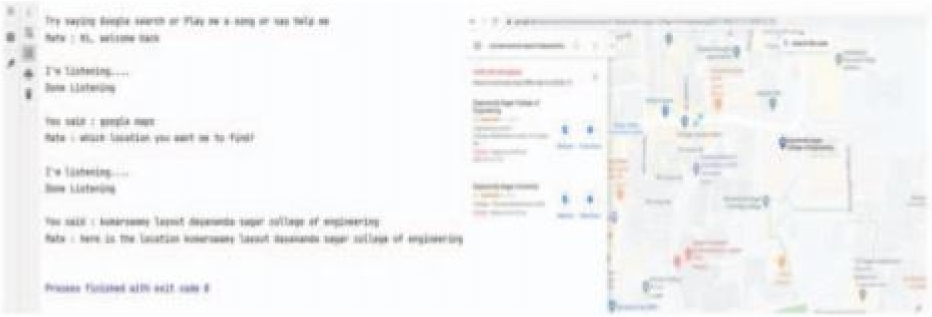


```
Try saying Google search or Play me a song or say help me
Mate : Hi, welcome back

I'm listening....
Done Listening

You said : play me a song
Mate : which one you want me to play

I'm listening....
Done Listening

You said : imagine dragons believer

Process finished with exit code 0
```

Figure 3.7 Output screen of Playing song on YouTube

C) Searches any location :

As shown in below fig 3.8 when we ask the voice assistant to search any location ,it receives the request and it searches that particular location using google maps.

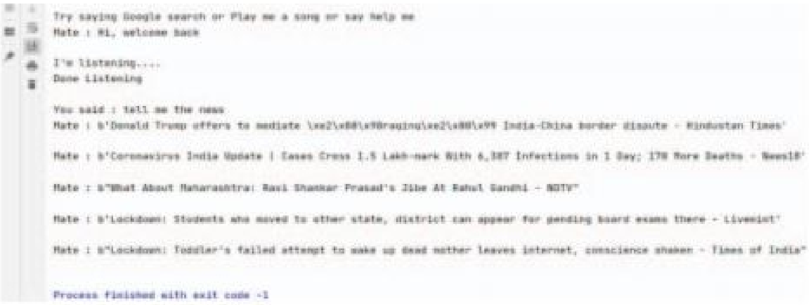


```
Try saying Google search or Play me a song or say help me
Mate : Hi, welcome back

I'm listening....
Done Listening

You said : google maps
Mate : which location you want me to find?

I'm listening....
Done Listening

You said : kumaraswamy layout dayananda sagar college of engineering
Mate : here is the location kumaraswamy layout dayananda sagar college of engineering


Process finished with exit code 0
```

Figure 3.8 output screen of searching location

D) Current news

As shown in figure in fig 3.9 when we ask current news to the voice assistant it receives the request and responds back by giving top 5 news for that day along with website name



```
Try saying Google search or Play me a song or say help me
Mate : Hi, welcome back

I'm listening....
Done Listening

You said : tell me the news
Mate : b'Donald Trump offers to mediate \xe2\x80\x98raging\xe2\x80\x99 India-China border dispute - Hindustan Times'

Mate : b'Coronavirus India Update | Cases Cross 1.5 Lakh-mark With 6,387 Infections in 1 Day; 170 More Deaths - News18'

Mate : b"What About Maharashtra: Ravi Shankar Prasad's Jibe At Rahul Gandhi - NDTV"

Mate : b'Lockdown: Students who moved to other state, district can appear for pending board exams there - Livemint'

Mate : b"Lockdown: Toddler's failed attempt to wake up dead mother leaves internet, conscience shaken - Times of India"

Process finished with exit code -1
```

Figure 3.9 Output screen of Showing Current News

According to the survey, 90% of the respondents at least once encountered and used voice assistants; 5% of them did not realize that they interacted with a robot or a neural network. Most often, respondents used Siri and Alisa, Google Assistant was rarely used. One person, residing permanently outside the Russian Federation, regularly uses the Amazon smart column and Alexa voice assistant integrated into it. Each survey participant encountered robots and support services' voice assistants at least once a year. They were basically used to solve a problem associated with company's services. One person (Evgenia, 24 years old), who provided a diary, contacted Beeline support service during the investigation week. After listening to the voice menu and performing a series of actions, the respondent received an answer to the question "Why doesn't the Internet work?" without using help of an operator. In the final interview, evaluating the effectiveness of communication, Evgenia noted that she "did not have to step over herself and communicate with a stranger. And I found out that the Internet would be repaired within a day". Another participant, who used the diary method (Alexey, 31 years old), has paid attention to the fact that if you need to solve a problem urgently, you still have to listen to information from the robot, and switching to a human being does not always happen after "contact the operator" voice command. Most often, the participants use voice assistants to find out about the weather, to set an alarm, to set a route for the navigator, to set a reminder etc. Every day only 2% of respondents use them. The reasons why voice assistants are rarely used or not used at all are the following:

- delayed activation or activation at the wrong time.

- erroneous recognition of speech

inability to complete the task fully.

E.g., one of the authors of the paper tried to order food using Siri and Alisa. In the first case, the voice assistant produced a list of stores within a radius of five kilometers, and in the second case it opened a recipe book. It is worth noting that Alisa, originally created to speak and understand Russian, recognizes the queries better and more accurately.

The last question for all respondents concerned evaluating the effectiveness of communication. The overwhelming majority of participants noted that voice assistants only help with solving functional problems and are completely unable to solve problems beyond the prescribed functionality. This, according to the participants, often complicates the use of technology, annoys the user, and the desire to use voice assistants disappears. The analysis of the survey revealed two fundamental problems of the voice assistants' market:

a) in marketing campaigns, the brands create false expectations for consumers.

b) the consumer is poorly informed of the essence of technology and is inclined to ascribe certain features to voice assistants that they cannot have. In an in-depth interview with the developers of a neural network, involved in modeling a voice based on emotional coloring, we failed to learn about the goals of programming and the development prospects of the voice assistants' market. The developers noted that presently voice assistants use voices devoid of emotional coloring; they do not know how to change the timbre and intonation, thus they sound artificially. The developments in the field of modeling a voice, which resembles a spontaneous human speech, will allow the users of "smart" gadgets to stop perceiving voice assistants as "something alien and useless speaking in a box". It is also worth noting that the developers do not pay attention to the fact that inefficiency of communication with voice assistants lies precisely in the fact that the consumer of brand services and products perceives software products as a full-fledged subject of communication. This gives rise to false expectations and makes communication inefficient. Therefore, the solution of the problem does not lie in imitating the subjective capabilities of voice assistants, but in the choice of communicative strategies that clearly indicate "who is who

IV. Automation python modules

Knowing various Python modules for editing spreadsheets, downloading files, and launching programs is useful, but sometimes there just aren't any modules for the applications you need to work with. The ultimate tools for automating tasks on your computer are programs you write that directly control the keyboard and mouse. These programs can control other applications by sending them virtual keystrokes and mouse clicks, just as if you were sitting at your computer and interacting with the applications yourself. This technique is known as graphical user interface automation, or GUI automation for short. With GUI automation, your programs can do anything that a human user sitting at the computer can do, except spill coffee on the keyboard. Think of GUI automation as programming a robotic arm. You can program the robotic arm to type at your keyboard and move your mouse for you. This technique is particularly useful for tasks that involve a lot of mindless clicking or filling out of forms.

Some companies sell innovative (and pricey) "automation solutions," usually marketed as robotic process automation (RPA). These products are effectively no different than the Python scripts you can make yourself with the pyautogui module, which has functions for simulating mouse movements, button clicks, and mouse wheel scrolls. This chapter covers only a subset of PyAutoGUI's features; you can find the full documentation at https://pyautogui.readthedocs.io/.

4.1 Pyautogui Module :

The pyautogui module can send virtual keypresses and mouse clicks to Windows, macOS, and Linux. Windows and macOS users can simply use pip to install PyAutoGUI. However, Linux users will first have to install some software that PyAutoGUI depends on. Open a terminal window and enter the following commands:

```
sudo apt-get install scrot

sudo apt-get install python3-tk

sudo apt-get install python3-dev
```

To install PyAutoGUI, run pip install --user pyautogui. Don't use sudo with pip; you may install modules to the Python installation that the operating system uses, causing conflicts with any scripts that rely on its original configuration. However, you should use the sudo command when installing applications with apt-get.

Appendix A has complete information on installing third-party modules. To test whether PyAutoGUI has been installed correctly, run import pyautogui from the interactive shell and check for any error messages.

Controlling Mouse Movement

In this section, you'll learn how to move the mouse and track its position on the screen using PyAutoGUI, but first you need to understand how PyAutoGUI works with coordinates.

The mouse functions of PyAutoGUI use x- and y-coordinates. Figure 20-1 shows the coordinate system for the computer screen; it's similar to the coordinate system used for images, discussed in Chapter 19. The origin, where x and y are both zero, is at the upper-left corner of the screen. The x-coordinates increase going to the right, and the y-coordinates increase going down. All coordinates are positive integers; there are no negative coordinates.

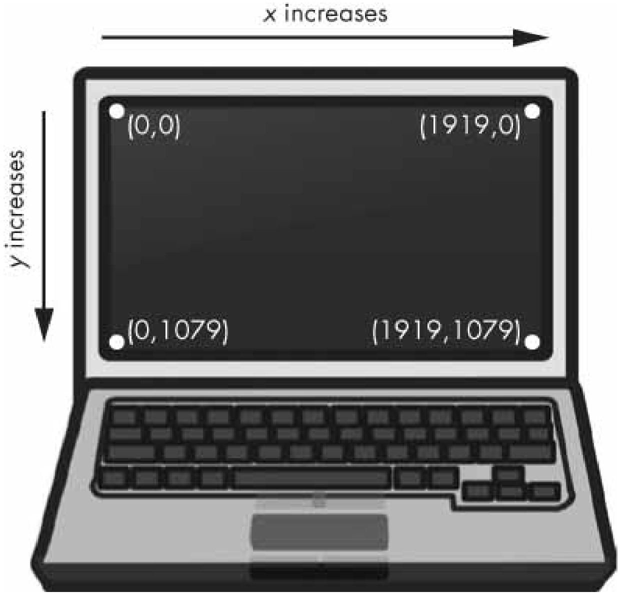


Figure 4.1 The coordinates of a computer screen with 1920×1080 resolution

Your resolution is how many pixels wide and tall your screen is. If your screen's resolution is set to 1920×1080, then the coordinate for the upper-left corner will be (0, 0), and the coordinate for the bottom-right corner will be (1919, 1079).

The pyautogui.size() function returns a two-integer tuple of the screen's width and height in pixels. Enter the following into the interactive shell:

```
>>> import pyautogui
>>> wh = pyautogui.size() # Obtain the screen resolution.
>>> wh
Size(width=1920, height=1080)
>>> wh[0]
1920
>>> wh.width
1920
```

The pyautogui.size() function returns (1920, 1080) on a computer with a 1920×1080 resolution; depending on your screen’s resolution, your return value may be different. The Size object returned by size() is a named tuple. Named tuples have numeric indexes, like regular tuples, and attribute names, like objects: both wh[0] and wh.width evaluate to the width of the screen. (Named tuples are beyond the scope of this book. Just remember that you can use them the same way you use tuples.)

4.1.1 Moving the Mouse :

Now that you understand screen coordinates, let’s move the mouse. The pyautogui.moveTo() function will instantly move the mouse cursor to a specified position on the screen. Integer values for the x- and y-coordinates make up the function’s first and second arguments, respectively. An optional duration integer or float keyword argument specifies the number of seconds it should take to move the mouse to the destination. If you leave it out, the default is 0 for instantaneous movement. (All of the duration keyword arguments in PyAutoGUI functions are optional.) Enter the following into the interactive shell:

```
>>> import pyautogui
>>> for i in range(10): # Move mouse in a square.
...     pyautogui.moveTo(100, 100, duration=0.25)
...     pyautogui.moveTo(200, 100, duration=0.25)
...
```

This example moves the mouse cursor clockwise in a square pattern among the four coordinates provided a total of 10 times. Each movement takes a quarter of a second, as specified by the duration=0.25 keyword argument. If you hadn’t passed a third argument to any of the pyautogui.moveTo() calls, the mouse cursor would have instantly teleported from point to point.

The pyautogui.move() function moves the mouse cursor relative to its current position. The following example moves the mouse in the same square pattern, except it begins the square from wherever the mouse happens to be on the screen when the code starts running:

```
>>> import pyautogui
>>> for i in range(10):
...       pyautogui.move(100, 0, duration=0.25)   # right
...       pyautogui.move(0, 100, duration=0.25)   # down
```

...   The pyautogui.move() function also takes three arguments: how many pixels to move horizontally to the right, how many pixels to move vertically downward, and (optionally) how long it should take to complete the movement. A negative integer for the first or second argument will cause the mouse to move left or upward, respectively.

4.1.2 Automation of Mouse:

This consists of some commands which are helpful to automate the functions such as drag, click, and another mouse .operations that are normally performed by hand. and these operations can be performed by controlling the position of the coordinates X and Y. Before performing any mouse operations it is very necessary to know the size of the screen of the computer.

Command :pyautogui.screen size()   (3)

This command (3) is used to know the size of the screen of the personal computer. so that it is easy to locate the position of the coordinates.

Command :pyautogui.position()   (4)

This command (4) is used to identify the position of the mouse pointer that means it will identify the values of the coordinates of x and y.

Command: pyautogui.move to()   (5)

This command (5) drags the mouse pointer irrespective of its current position. for this, it is necessary to give the values of the coordinates X, Y, and duration to drag the pointer to the specific location.

Command :pyautogui.click() and pyautogui.doubleclick()   (6)

This command (6) is used for basic click operation. click can accept five parameters. i.e the values of coordinates and the last one are the number of clicks to be done. and instead of mentioning the value of the number of clicks DoubleClick() command can be used.

4.1.3 Automation of Keyboard:

pyautogui provides some built-in to perform the functions of the keyboard automatically.

Command : pyautogui.press() (7)

This command (7) is used to press the particularly given key on the keyboard.

Example:Command:pyautogui.press('k') (8)

In the above example (8), k is pressed automatically.

NOTE: It may not work well for some functions such as volume up, volume down, etc. for that the function pyautogui.keyword_keys can be used.

Command :pyautogui.typewrite() (9)

This command (9) is used for pressing multiple buttons one by one based on the given time interval.

Command : pyautogui.typewrite("hello",interval=0.25) (10)

In brackets, it takes two arguments first one is what the user wants python to write. and the second parameter is the time between the characters that python takes (10). so in this case hello has five characters. so there are four stops in between each. and it takes 0.25 seconds gap to type each character. so python takes one second to type the word HELLO.

Command : pyautogui.hotkey('shift','m','o','p'). (11)

Thus by using the above commands (11) if user predefined different distances of ultrasonic sensors with different youtube video functions like increasing volume, decreasing volume, forward, and rewind. The user can able to control youtube just with hand position

4.2 Python Text to Speech by using pyttsx3

pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 and 3. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3. Engine instance. it is a very easy to use tool which converts the entered text into speech.

The pyttsx3 module supports two voices first is female and the second is male which is provided by “sapi5” for windows.

It supports three TTS engines:

1)*sapi5* – SAPI5 on Windows

2)*nsss* – NSSpeechSynthesizer on Mac OS X

3)*espeak* – eSpeak on every other platform

Installation:

To install the pyttsx3 module, first of all, you have to open the terminal and write

```
pip install pyttsx3
```

If you receive errors such as No module named win32com.client, No module named win32, or No module named win32api, you will need to additionally install pypiwin32.

It can work on any platform. Now we are all set to write a program for conversion of text to speech.

4.2.1 Supported synthesizers

Version 2.6 of pyttsx3 includes drivers for the following text-to-speech synthesizers. Only operating systems on which a driver is tested and known to work are listed. The drivers may work on other systems.

• SAPI5 on Windows XP and Windows Vista and Windows 8,8.1 , 10

• NSSpeechSynthesizer on Mac OS X 10.5 (Leopard) and 10.6 (Snow Leopard) • espeak on Ubuntu Desktop Edition 8.10 (Intrepid), 9.04 (Jaunty), and 9.10 (Karmic)

The pyttsx3.init() documentation explains how to select a specific synthesizer by name as well as the default for each platform

4.2.2 Using pyttsx3 :

An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3.Engine instance. During construction, the engine initializes a pyttsx3.driver.DriverProxy object responsible for loading a speech engine driver implementation from the pyttsx3.drivers module. After construction, an application uses the engine object to register and unregister event callbacks; produce and stop speech; get and set speech engine properties; and start and stop event loops.

4.2.3 The Engine factory:

pyttsx3.init([driverName : string, debug : bool]) → pyttsx3.Engine .

Gets a reference to an engine instance that will use the given driver. If the requested driver is already in use by another engine instance, that engine is returned. Otherwise, a new engine is created.

Parameters

• **driverName** – Name of the pyttsx3.drivers module to load and use. Defaults to the best available driver for the platform, currently:

– sapi5 - SAPI5 on Windows –

nsss - NSSpeechSynthesizer on Mac OS X

–espeak - eSpeak on every other platform

debug – Enable debug output or not.

Raises

• **ImportError** – When the requested driver is not found

• **RuntimeError** – When the driver fails to initialize

4.2.4 The Engine interface

class pyttsx3.engine.**Engine**

Provides application access to text-to-speech synthesis.

connect(topic : string, cb : callable) → dict

Registers a callback for notifications on the given topic

Parameters

• **topic** – Name of the event to subscribe to.

- **cb** – Function to invoke when the event fires.

Returns A token that the caller can use to unsubscribe the callback later.

The following are the valid topics and their callback signatures

started-utterance

Fired when the engine begins speaking an utterance. The associated callback must have the folowing signature

onStartUtterance(name : string) → None

Parameters name – Name associated with the utterance

Started-word

Fired when the engine begins speaking a word.The associated callback must have the folowing signature

onStartWord(name : string, location : integer, length : integer)

Parameters name – Name associated with the utterance.

Finished-utterance

Fired when the engine finishes speaking an utterance. The associated callback must have the folowing signature

onFinishUtterance(name : string, completed : bool) → None

Parameters

- name – Name associated with the utterance.
- completed – True if the utterance was output in its entirety or not.

4.3 speech recognition

Speech recognition is an interdisciplinary subfield of computer science and computational linguistics that develops methodologies and technologies that enable the recognition and translation of spoken language into text by computers. It is also known as automatic speech recognition (ASR), computer speech recognition or speech to text (STT). It incorporates knowledge and research in the computer science, linguistics and computer engineering fields.

Some speech recognition systems require "training" (also called "enrollment") where an individual speaker reads text or isolated vocabulary into the system. The system analyzes the person's specific voice and uses it to fine-tune the recognition of that person's speech, resulting in increased accuracy. Systems that do not use training are called "speaker independent"systems. Systems that use training are called "speaker dependent".

Speech recognition applications include voice user interfaces such as voice dialing (e.g. "call home"), call routing (e.g. "I would like to make a collect call"), domotic appliance control, search key words (e.g. find a podcast where particular words were spoken), simple data entry (e.g., entering a credit card number), preparation of structured documents (e.g. a radiology report), determining speaker characteristics, speech-to-text processing (e.g., word processors or emails), and aircraft (usually termed direct voice input).

The term voice recognitionor speaker identification refers to identifying the speaker, rather than what they are saying. Recognizing the speaker can simplify the task of translating speech in systems that have been trained on a specific person's voice or it can be used to authenticate or verify the identity of a speaker as part of a security process.

From the technology perspective, speech recognition has a long history with several waves of major innovations. Most recently, the field has benefited from advances in deep learning and big data. The advances are evidenced not only by the surge of academic papers published in the field, but more importantly by the worldwide industry adoption of a variety of deep learning methods in designing and deploying speech recognition systems.

4.3.1 Neural networks

Neural networks emerged as an attractive acoustic modeling approach in ASR in the late 1980s. Since then, neural networks have been used in many aspects of speech recognition such as phoneme classification,phoneme classification through multi-objective evolutionary algorithms, isolated word recognition, audiovisual speech recognition, audiovisual speaker recognition and speaker adaptation.

Neural networks make fewer explicit assumptions about feature statistical properties than HMMs and have several qualities making them attractive recognition models for speech recognition. When used to estimate the probabilities of a speech feature segment, neural networks allow discriminative training in a natural and efficient manner. However, in spite of their effectiveness in classifying short-time units such as individual phonemes and isolated words, early neural networks were rarely successful for continuous recognition tasks because of their limited ability to model temporal dependencies.

One approach to this limitation was to use neural networks as a pre-processing, feature transformation or dimensionality reduction,step prior to HMM based recognition. However, more recently, LSTM and related recurrent neural networks (RNNs)and Time Delay Neural Networks(TDNN's) have demonstrated improved performance in this area.

4.3.2 End-to-end automatic speech recognition

Since 2014, there has been much research interest in "end-to-end" ASR. Traditional phonetic-based (i.e., all HMM-based model) approaches required separate components and training for the pronunciation, acoustic and language model. End-to-end models jointly learn all the components of the speech recognizer. This is valuable since it simplifies the training process and deployment process. For example, a n-gram language model is required for all HMM-based systems, and a typical n-gram language model often takes several gigabytes in memory making them impractical to deploy on mobile devices. Consequently, modern commercial ASR systems from Google and Apple (as of 2017) are deployed on the cloud and require a network connection as opposed to the device locally.

The first attempt at end-to-end ASR was with Connectionist Temporal Classification (CTC)-based systems introduced by Alex Graves of Google DeepMind and Navdeep Jaitly of the University of Toronto in 2014. The model consisted of recurrent neural networks and a CTC layer. Jointly, the RNN-CTC model learns the pronunciation and acoustic model together, however it is incapable of learning the language due to conditional independence assumptions similar to a HMM. Consequently, CTC models can directly learn to map speech acoustics to English characters, but the models make many common spelling mistakes and must rely on a separate language model to clean up the transcripts. Later, Baidu expanded on the work with extremely large datasets and demonstrated some commercial success in Chinese Mandarin and English. In 2016, University of Oxford presented LipNet, the first end-to-end sentence-level lipreading model, using spatiotemporal convolutions coupled with an RNN-CTC architecture, surpassing human-level performance in a restricted grammar dataset. A large-scale CNN-RNN-CTC architecture was presented in 2018 by Google DeepMind achieving 6 times better performance than human experts.

An alternative approach to CTC-based models are attention-based models. Attention-based ASR models were introduced simultaneously by Chan et al. of Carnegie Mellon University and Google Brain and Bahdanau et al. of the University of Montreal in 2016. The model named "Listen, Attend and Spell" (LAS), literally "listens" to the acoustic signal, pays "attention" to different parts of the signal and "spells" out the transcript one character at a time. Unlike CTC-based models, attention-based models do not have conditional-independence assumptions and can learn all the components of a speech recognizer including the pronunciation, acoustic and language model directly. This means, during deployment, there is no need to carry around a language model making it very practical for applications with limited memory. By the end of 2016, the attention-based models have seen considerable success including outperforming the CTC models (with or without an external language model). Various extensions have been proposed since the original LAS model.

4.4 Introduction to pywhatkit module

Python offers numerous inbuilt libraries to ease our work. Among them pywhatkit is a Python library for sending WhatsApp messages at a certain time, it has several other features too.

Following are some features of pywhatkit module:

1. Send WhatsApp messages.
2. Play a YouTube video.

In Python3 pywhatkit module will not come pre-installed, so you can install it by using the command:

```
pip install pywhatkit
```

1. Send Whatsapp Messages:

Here, we will learn the simplest way of using pywhatkit module which utilises the WhatsApp webpage to automate messages sending to any number on WhatsApp. But make sure that you have logged into your WhatsApp in your browser.

Syntax : pywhatkit.sendmsg(“receiver mobile number”,”message”,hours,minutes)

The parameters are

phone_num (required) - Phone number of target with country code
message (required) - Message that you want to sendwhatmsg
time_hour (required) - Hours at which you want to send message in 24 hour format
time_min (required) - Minutes at which you want to send message
wait_time (optional, val=20) - Seconds after which the message will be sent after opening the web
print_waitTime (optional, val=True) - Will print the remaining time if set to true

2. Play a YouTube video:

Function pywhatkit.playonyt(), opens the YouTube in your default browser and plays the video you mentioned in the function. If you pass the topic name as parameter, it plays the random video on that topic. On passing the URL of the video as the parameter, it open that exact video.

Syntax :

```
 pywhatkit.playonyt(“url/topic name”)
```

Parameters:

URL/Topic Name : Mention the topic name or URL of the YouTube video

Some common errors

Video not opening - Make sure the topic exists or you have provided proper spelling.

3. Perform Google Search:

You can perform a Google search using the following simple command. It opens your browser and searches for the topic you have given in your code.

Syntax : pywhatkit.search("Keyword")

Parameters:

Keyword : It open your browser and performs search operation.

4.5 GOOGLE NEWS :

The first task is to install GoogleNews and newspaper3k which can be accomplished with the following statement.

```
!pip install GoogleNews
```

```
!pip install newspaper3k
```

It's time to import some libraries to get a list of the articles that we will now fetch.

```
from GoogleNews import GoogleNews
from newspaper import Article
import pandas as pd
```

The next task is to fetch the data. Given the time period and query on which you want to search the news articles, we will get a list which will contain date, title, media, description, link where the article is published and link of the image under the img attribute.You will see in the following code, that googlenews.result() will return a list of everything we discussed above

```
googlenews=GoogleNews(start='05/01/2020',end='05/31/2020')
googlenews.search('Coronavirus')
result=googlenews.result()
df=pd.DataFrame(result)
print(df.head())
```

4.5.1 How to use the Google News API with Python :

Google News is a service that we can use to take a pulse of a popular topic. Currently, there is a presidential election happening in the United States. With this event, we have an opportunity for news data analysis. In the realm of marketing, there is a concept of Effective Frequency. This refers to how many times you need to expose people to a message or idea before making a buy decision. In an election, the buy is a vote, and the message is which candidate to vote for. In this article, we will walk through how to use the Google News API with Python. This will allow us to capture data over time and analyze it.

The idea we will test is about media exposure for presidential candidates. Who will have their name appear more often in the news for a few days leading up to the election? Yes, that does not guarantee exposure to the winning candidate more often. News articles are not the only source of media. And it does not ensure that the names are being used in a positive light. But it should give us some signs of the public awareness of the candidates

4.6 Simple Mail Transfer Protocol (SMTP) :

Email is emerging as one of the most valuable services on the internet today. Most of the internet systems use SMTP as a method to transfer mail from one user to another. SMTP is a push protocol and is used to send the mail whereas POP (post office protocol) or IMAP (internet message access protocol) are used to retrieve those mails at the receiver’s side.

SMTP Fundamentals

SMTP is an application layer protocol. The client who wants to send the mail opens a TCP connection to the SMTP server and then sends the mail across the connection. The SMTP server is always on listening mode. As soon as it listens for a TCP connection from any client, the SMTP process initiates a connection on that port (25). After successfully establishing the TCP connection the client process sends the mail instantly.

SMTP Protocol

The SMTP model is of two type :

1. End-to- end method
2. Store-and- forward method

The end to end model is used to communicate between different organizations whereas the store and forward method are used within an organization. A SMTP client who wants to send the mail will contact the destination's host SMTP directly in order to send the mail to the destination. The SMTP server will keep the mail to itself until it is successfully copied to the receiver's SMTP.
The client SMTP is the one which initiates the session let us call it as the client- SMTP and the server SMTP is the one which responds to the session request and let us call it as receiver-SMTP. The client- SMTP will start the session and the receiver-SMTP will respond to the request.

Some SMTP Commands:

HELO - It's the first SMTP command: is starts the conversation identifying the sender server and is generally followed by its domain name.

EHLO - An alternative command to start the conversation, underlying that the server is using the Extended SMTP protocol.

MAIL FROM - With this SMTP command the operations begin: the sender states the source email address in the "From" field and actually starts the email transfer.

RCPT TO - It identifies the recipient of the email; if there are more than one, the command is simply repeated address by address.

DATA - With the DATA command the email content begins to be transferred; it's generally followed by a 354 reply code given by the server, giving the permission to start the actual transmission.

Working of SMTP :

1. **Composition of Mail:** A user sends an e-mail by composing an electronic mail message using a Mail User Agent (MUA). Mail User Agent is a program which is used to send and receive mail. The message contains two parts: body and header. The body is the main part of the message while the header includes information such as the sender and recipient address. The header also includes descriptive information such as the subject of the message. In this case, the message body is like a letter and header is like an envelope that contains the recipient's address.
2. **Submission of Mail:** After composing an email, the mail client then submits the completed e-mail to the SMTP server by using SMTP on TCP port 25.
3. **Delivery of Mail:** E-mail addresses contain two parts: username of the recipient and domain name. For example, vivek@gmail.com, where "vivek" is the username of the recipient and "gmail.com" is the domain name.

If the domain name of the recipient's email address is different from the sender's domain name, then MSA will send the mail to the Mail Transfer Agent (MTA). To relay the email, the MTA will find the target domain. It checks the MX record from Domain Name System to obtain the target domain. The MX record contains the domain name and IP address of the recipient's domain. Once the record is located, MTA connects to the exchange server to relay the message.

4. **Receipt and Processing of Mail:** Once the incoming message is received, the exchange server delivers it to the incoming server (Mail Delivery Agent) which stores the e-mail where it waits for the user to retrieve it.

5. **Access and Retrieval of Mail:** The stored email in MDA can be retrieved by using MUA (Mail User Agent). MUA can be accessed by using login and password.

Python Sending Email using SMTP

Python provides a **smtplib** module, which defines an the SMTP client session object used to send emails to an internet machine. For this purpose, we have to import the **smtplib** module using the import statement.

```
$ import smtplib
```

The SMTP object is used for the email transfer. The following syntax is used to create the smtplib object.

```
import smtplib
smtpObj = smtplib.SMTP(host, port, local_hostname)
```

It accepts the following parameters. b

- **host:** It is the hostname of the machine which is running your SMTP server. Here, we can specify the IP address of the server like or localhost. It is an optional parameter.
- **port:** It is the port number on which the host machine is listening to the SMTP connections. It is 25 by default.
- **local_hostname:** If the SMTP server is running on your local machine, we can mention the hostname of the local machine.

V CONCLUSIONS

Human hand gesture is highly sophisticated and human-computer interacting system using this user can automate the email,whatsapp[3], youtube and browsing features in PC or desktops in the verge of reducing time consumption in doing them manually .the paper described most of the required components and modules which are required to build such type of machine with less cost and power-efficient.In the coming era, automation will become much trending, and utilization of these machines will make future generations much simpler and basic for updating or inventing other new innovations.

Ultrasonic sensors are non-intrusive in that they do not require physical contact with their target, and can detect certain clear or shiny targets otherwise obscured to some vision-based sensors. On the other hand, their measurements are very sensitive to temperature and to the angle of the target. Temperature and humidity affect the speed of sound in air. Therefore, range finders may need to be recalibrated to make accurate measurements in a new environment. Temperature variations and air currents can create invisible boundaries that will reflect ultrasonic waves, so care must be taken to avoid these. For the transmitted wave to echo back to the receiver, the target surface must be perpendicular to the transmitter. Round objects are therefore most easily sensed since they always show some perpendicular face. When targeting a flat object, care must be taken to ensure that its angle with respect to the sensor does not exceed a particular range.In our project we have implemented many things compared to other assistants. Now a days it is very useful in human life because it is a hands-free application. It is a very simple application. As well as it is used in a business field also for example in laboratory, the person wears gloves and body suits for their safety purpose so it is difficult to type, through voice assistant they can get any information so that their work becomes easy. Voice assistants are useful in many fields such as education, daily life application, home appliances, etc. and the voice assistant is also useful for the illiterate people they can get any information just by saying to the assistant, luxury is available for people, thanks to AI-based voice assistants. Voice assistant is developing more and more in daily life. Many companies of voice assistant trying to improve interaction and more features to the next level and many of the youth started using a voice assistant in daily life and from many sources, the result showing very good feedback. Compared to the last 2 years voice assistants have been developed more and more.Speech Recognition is an excellent application of artificial intelligence to work with since speech is one of the most basic activities of humans. Computers would quickly arrive with preinstalled automatic speech recognition systems. Equipment and devices with this technology would make the lives of the blind, the deaf, and other physically challenged people by providing them access to computers without the click of buttons. Voice recognition would soon be a feature of every small to big device ranging from ticket selling devices to washing machines etc.

VI. FUTURE WORK

Gesture supervise machine is a highly sophisticated machine which had many future concern. here we are uniting many inbuilt python modules in the project in which each of them having different application .Using this project as a base idea we could able to implement the patients beds inclination systems automatically just by the doctors/nurse hand gesture or voice command. During pandemic time hospital beds monitoring system is very important ,by using this project as a base ,user can able to interact with voice assistant to know the number of vacant beds.Water management in future will become more significant , by deploying ultrasonic sensor over the water tank,level of the water can be known through voice assistance.In the future with good technology people can integrate this technology/project into television sets , so that without touching the remote control the user can control the TV just by changing hand position and voice commands. In doing these type of things .people's time can be saved and things will become much sophisticated.

REFERENCES

[1] A. Giyantara, V. Mudeng, H. S. Natiand and M. Izzudin Abdillah Afif, "Microcontroller Serial Communication to Analyze Bit Characters," 2018 2nd Borneo International Conference on Applied Mathematics and Engineering (BICAME), 2018, pp. 223-227, doi: 10.1109/BICAME45512.2018.1570509845.

[2] F. Zhou, X. Li and Z. Wang, "Efficiently User-Independent Ultrasonic-Based Gesture Recognition Algorithm," 2019 IEEE SENSORS, 2019, pp. 1-4, doi: 10.1109/SENSORS43011.2019.8956774

[3] L. G. Tyutelova, V. N. Lisovitskaya, E. S. Shevchenko and V. D. Shevchenko, "The Voice-Activated Digital Assistants in the Brand-Consumer Communication," 2020 IEEE Communication Strategies in Digital Society Seminar (ComSDS), 2020, pp. 137-140, doi: 10.1109/ComSDS49898.2020.9101244.

[4] M. Al-Hammadi et al., "Deep Learning-Based Approach for Sign Language Gesture Recognition With Efficient Hand Gesture Representation," in IEEE Access, vol. 8, pp. 192527-192542, 2020, doi: 10.1109/ACCESS.2020.3032140.

[5] O. V. Biryukova and I. V. Koretskaya, "Selection of Ultrasonic Sensors for Distance Monitoring," 2020 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO), 2020, pp. 1-4,doi: .1109/SYNCHROINFO49631.2020.9166089.

[6] R. Golovanov, D. Vorotnev and D. Kalina, "Combining Hand Detection and Gesture Recognition Algorithms for Minimizing Computational Cost," 2020 22th International Conference on Digital Signal Processing and its Applications (DSPA), 2020, pp. 1-4, doi:.1109/DSPA48919.2020.9213273.

[7]] O. V. Biryukova and I. V. Koretskaya, "Using an Ultrasonic Sensor to Monitor Position, Transfer Control Commands and Work Information," 2019 Systems of Signals Generating and Processing in the Field of on Board Communications, 2019, pp. 1-5, doi: 10.1109/SOSG.2019.8706796

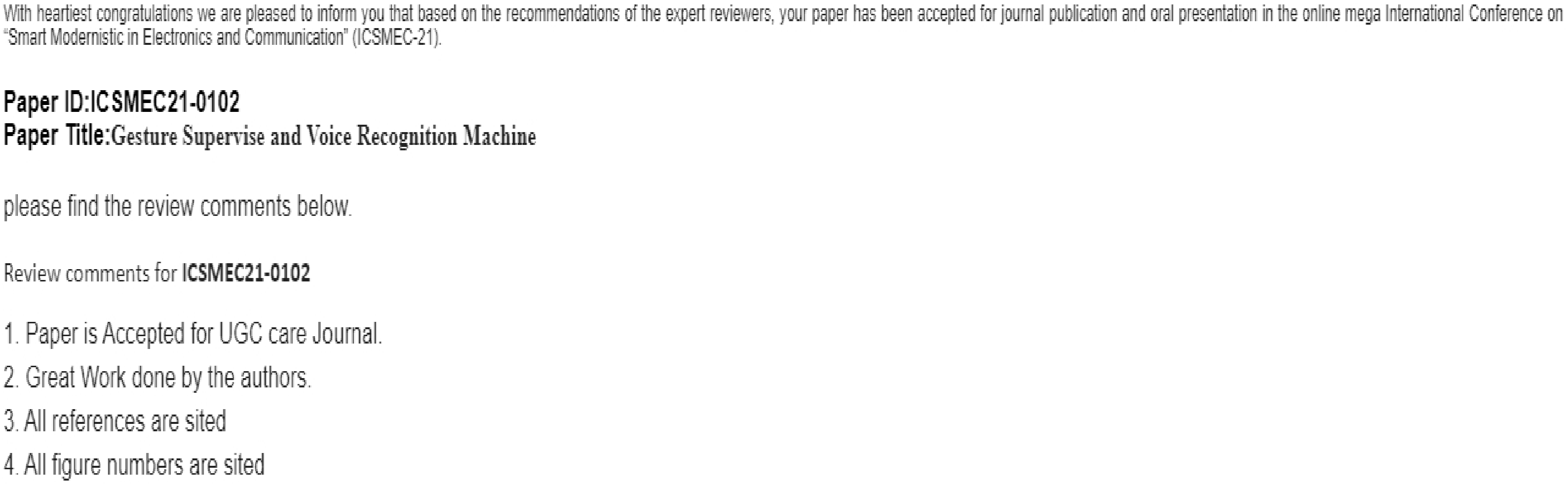
[8] R. Sureswaran, H. A. Bazar, O. Abouabdalla, A. M. Manasrah and H. El-Taj, "Active e-mail system SMTP protocol monitoring algorithm," 2009 2nd IEEE International Conference on Broadband Network & Multimedia Technology, 2009, pp. 257-260, doi: 10.1109/ICBNMT.2009.5348490.

PAPER PUBLICATION DETAILS

The paper has been accepted for the journal publication and oral publication in the international conference on "Smart Modernistic in Electronics and Communication" (ICSMEC -21).

Paper ID: ICSMEC21-0102

Paper title: Gesture supervise and voice recognition machine



With heartiest congratulations we are pleased to inform you that based on the recommendations of the expert reviewers, your paper has been accepted for journal publication and oral presentation in the online mega International Conference on "Smart Modernistic in Electronics and Communication" (ICSMEC-21).

Paper ID:ICSMEC21-0102
Paper Title:Gesture Supervise and Voice Recognition Machine

please find the review comments below.

Review comments for **ICSMEC21-0102**

1. Paper is Accepted for UGC care Journal.
2. Great Work done by the authors.
3. All references are sited
4. All figure numbers are sited