Methodology:

A device to be used by people with hearing, speaking and blind disability, so that they can have a two-way communication, has been proposed in this project. There are two main features that the device can do, the first one being able to convert the speech into vibrotactile output, and the other feature is being able to get the input as a Morse using 2 identical touch sensors and then converting it into speech.

The wearable device uses Arduino UNO, as the core component in the overall system. The patient wears the designed device as a watch, which is fixed with 1 vibration motor and 2 identical touch sensors. The system also has a microphone to listen to the input audio and act accordingly. For a conversation between normal and a disabled person, if the normal person is speaking, the audio is captured and the vibration motor is made to generate vibrations in the Morse code sequence. Then this Morse code generated is converted to audio and then fed out using a speaker. Arduino UNO is a popular choice as it is very easy to use, affordable, and it is a programmable board that can be used in a variety of projects. It can be interfaced with other boards, micro controller and computers ,making it a bit ideal for our device. Fig. 1 shows the overall design of the system which uses Ardiuno UNO, which is connected to laptop via wire.

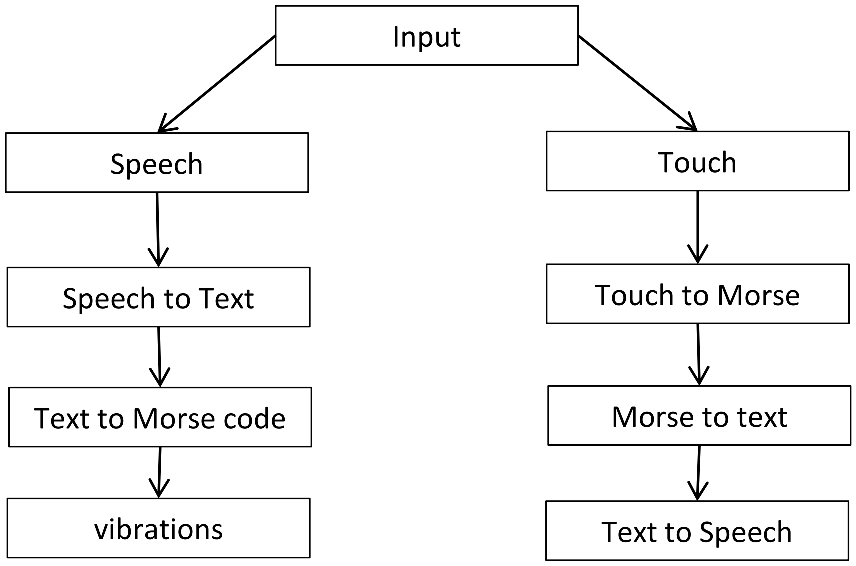


Fig-1

Text-to-Speech (TTS) technology converts written text into spoken words using computational algorithms and language processing techniques. It bridges the gap between written communication and auditory output, enabling applications such as accessibility tools, virtual assistants, and interactive voice
systems. The for converting Text-to-speech our device uses pythons inbuilt text to speech module.

A. Speech to Morse code to vibrotactile output:

To establish a mode of communication between normal person and a person who is deaf, mute and blind, speech needs to be converted into Morse code and then to be transmitted as vibration feedback. This whole procedure requires to steps : (1) Speech to text conversion and (2) Text to Vibration conversion.

1) Speech to Text conversion:

When someone talks to the person wearing the device, the voice is transmitted to the Arduino UNO via a microphone. The Arduino is predefined with codes running inside it which is enables it to receive audio inputs in mp3 format and store in mp3 files. The program uses pythons inbuilt speech recognition to convert the mp3 file into text.

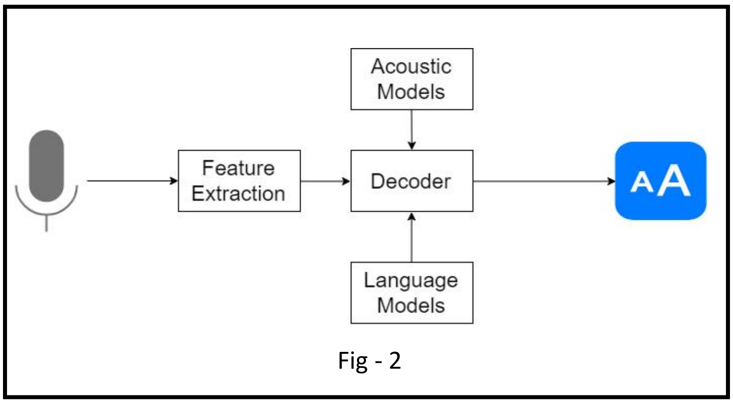


Fig - 2

The audio is sliced into chunks of 20 milliseconds. For each of the audio slice, it tries to figure out the letter that corresponds to the sound. Recurrent neural network is used which has memory that influences future predictions. It can predict the upcoming letter from the recent letter. For example, if the person says “HEL” then it predicts the rest of the word to construct “HELLO”. Therefore, having access to previous memory, the prediction method helps the neural network make more accurate predictions going forward. After the entire audio clip is run through the neural network one chunk at a time, each audio chunk is mapped to the letters most likely spoken in the chunk. The trick is to combine these pronunciation-based predictions with likelihood scores based on large database of written text books, news articles etc.

The audio is captured via microphone, which is then fed to the pythons built in speech to text module, ant then is converted to text. The program can also convert the text to Morse code by taking each letter and white space from the text, which it got as the input, and representing them with dots, dash and slashes. The device has one vibration motor, which is assigned with the task of outputting the Morse code items which are dots, dashes and slashes. The program can differentiate spaces in two ways, one is space between two words and the other is the space between two different lines. Space between two words is represented by a single slash (‘/’) while the space between two different lines is represented by a double slash (‘//’)

2) Text to vibration conversion:

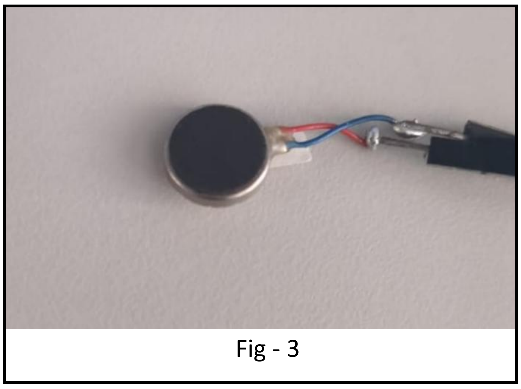
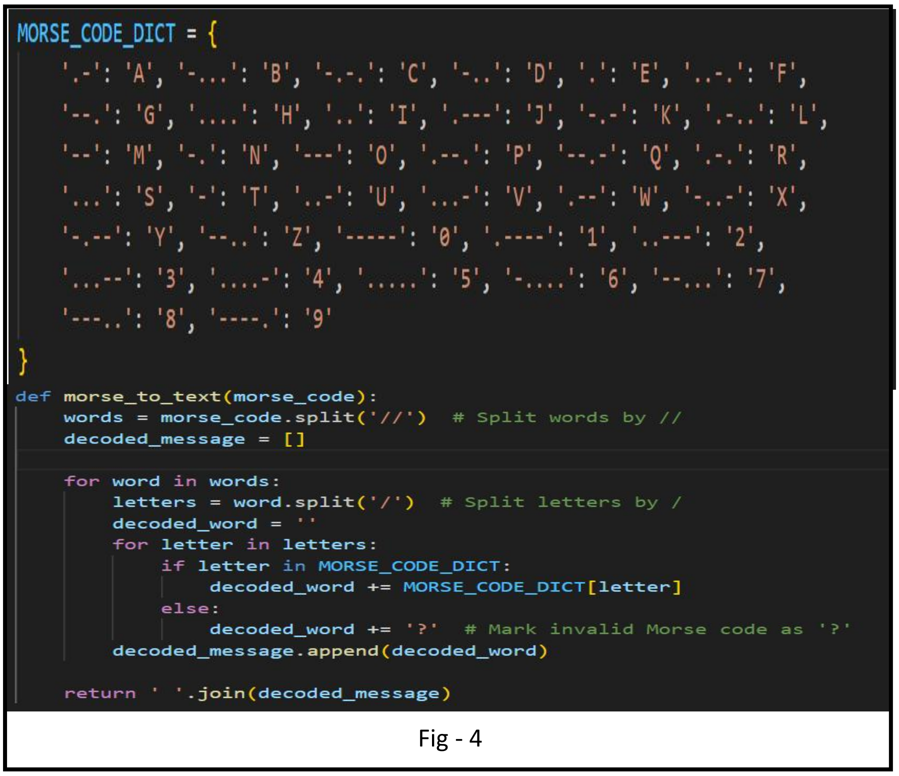


Fig - 3

Fig. 3 shows the type of vibration motors used in the device. The Morse code, dot dashes and slashes where each is converted to vibrations corresponding to 0.5, 1 and 0.25 x2 seconds of vibrations. This makes the user to be able to distinguish between what is dot, what is dash and what is slashes, so that he can understand the conversation. The vibration motors a fit on the sides of a watch strap, which will be worn on the wrist.

3) Code explanation:

Python provides a data structure called dictionary which stores information in the form of key-value pairs which is very convenient for implementing a cipher such as the Morse code. We can save the Morse code chart in a dictionary where (key-value pairs) => (English Characters-Morse Code). The plain-text (English characters) take the place of keys and the cipher text (Morse code) form the values of the corresponding keys. The values of keys can be accessed from the dictionary in the same way we access the values of an array through their index and vice verse. Fig. 4 demonstrate the code implementation of Morse code dictionary.A loop searches for the specific letter and white-space from text file that it receives from pythons inbuilt module. For every letter and space, Morse code is generated using the Morse Code Dictionary and is then stored into a string named decoded_message.



```
MORSE_CODE_DICT = {
    '.-': 'A', '-...': 'B', '-.-.': 'C', '-..': 'D', '.': 'E', '..-.': 'F',
    '--.': 'G', '....': 'H', '..': 'I', '.---': 'J', '-.-': 'K', '.-..': 'L',
    '--': 'M', '-.': 'N', '---': 'O', '.--.': 'P', '--.-': 'Q', '.-.': 'R',
    '...': 'S', '-': 'T', '..-': 'U', '...-': 'V', '.--': 'W', '-..-': 'X',
    '-.--': 'Y', '--..': 'Z', '-----': '0', '.----': '1', '..---': '2',
    '...--': '3', '....-': '4', '.....': '5', '-....': '6', '--...': '7',
    '---..': '8', '----.': '9'
}
def morse_to_text(morse_code):
    words = morse_code.split('//')  # Split words by //
    decoded_message = []

    for word in words:
        letters = word.split('/')  # Split letters by /
        decoded_word = ''
        for letter in letters:
            if letter in MORSE_CODE_DICT:
                decoded_word += MORSE_CODE_DICT[letter]
            else:
                decoded_word += '?'  # Mark invalid Morse code as '?'
        decoded_message.append(decoded_word)

    return ' '.join(decoded_message)
```

Fig - 4

B. Touch input to Morse Code to speech:

Since the device implements a two-way communication, the user can also communicate using a touch sensor. Two touch sensors are connected to the Arduino UNO through the GPIO pins. The user inputs Morse code by touch the sensor. If the touch duration is less than 0.5 sec the Arduino UNO takes dot as the input, if the touch duration is more than 0.5 and less than 1.5 the Arduino UNO takes input as dash. If both the touch sensors are activated at the same time the Arduino UNO understands that the user wants to read his message aloud.

1) Touch to Morse code:

Fig-5 shows the touch sensor used in this device. The design uses a touch sensor placed on the wrist of the user. A **vibrotactile sensor** is a type of sensor designed to detect and measure vibration-related stimuli, often in the context of tactile or touch-based applications. These sensors are commonly used in areas where feedback through vibration is essential, such as haptic, robotics, wearable devices, and biomedical engineering. A **vibration motor** is a compact electric motor designed to generate vibrations by creating an imbalance in the motor's rotation. These motors are widely used in devices where tactile feedback is required, such as mobile phones, game controllers, and wearable.

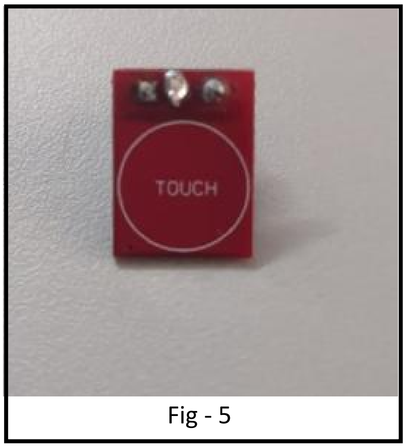


Fig - 5

The working principle of a vibration motor involves the generation of vibrations through an eccentric rotating mass (ERM) or a similar mechanism. The motor has a small weight attached off-center to its rotating shaft, creating an imbalance during rotation. As the motor spins, this imbalance generates centrifugal force, which causes the motor to vibrate. The frequency and intensity of the vibrations are determined by the motor's speed and the mass of the weight. By controlling the motor's power supply, the vibration can be adjusted to suit specific applications, such as producing tactile feedback or encoding patterns like Morse code. Fig. 6 shows a graph on how the vibration motor acts upon receiving the Morse code signal.

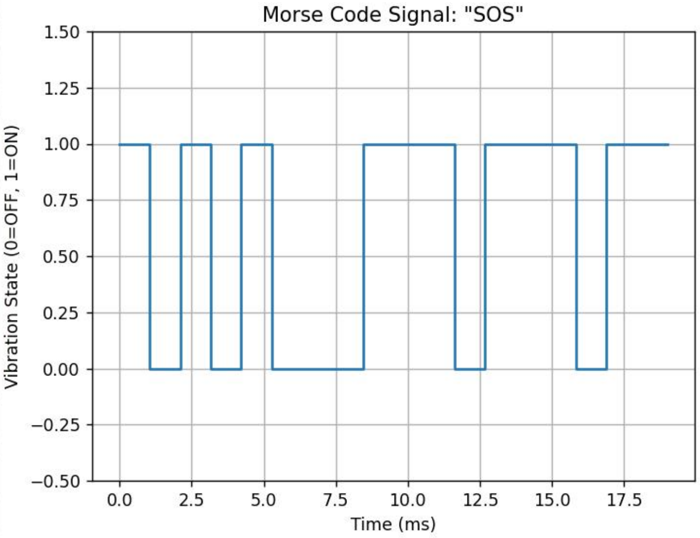
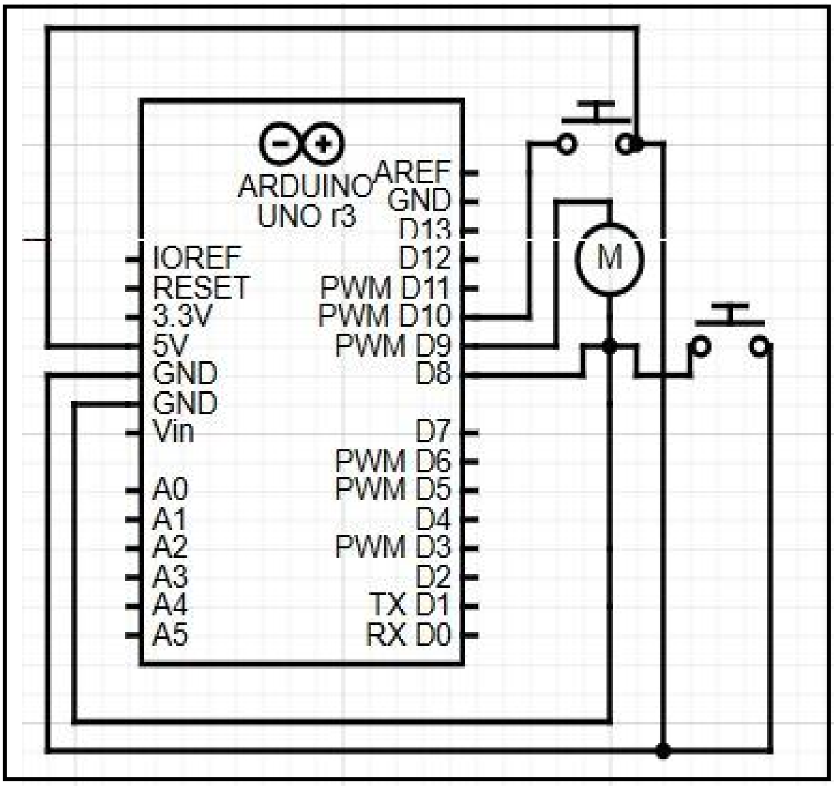


Fig - 6

2) Text to Speech:

For converting the text into speech we used pythons inbuilt module speech_recognition. It takes text as its input and generates an audio stream as a mp3 file.

Experiments:


ARDUINO
UNO r3
AREF
GND
D13
D12
PWM D11
PWM D10
PWM D9
D8
D7
PWM D6
PWM D5
D4
PWM D3
D2
TX D1
RX D0
IOREF
RESET
3.3V
5V
GND
GND
Vin
A0
A1
A2
A3
A4
A5
M