**1.INTRODUCTION**

**1.1Aim of Project**

* To represent the American Standard Sign Language which can be understood by

the deaf and the dumb.

* To bridge the communication and expression gap between the normal people who cannot understand the sign language, and the deaf and dumb who cannot understand the normal speech.
* To provide a software package to convert the speech signal, (which does not have any meaning for the deaf and the dumb) into the sign language.
* To provide a project for deaf and dumb children which will help them in their primary education.

**1.2 SIGN LANGUAGE**

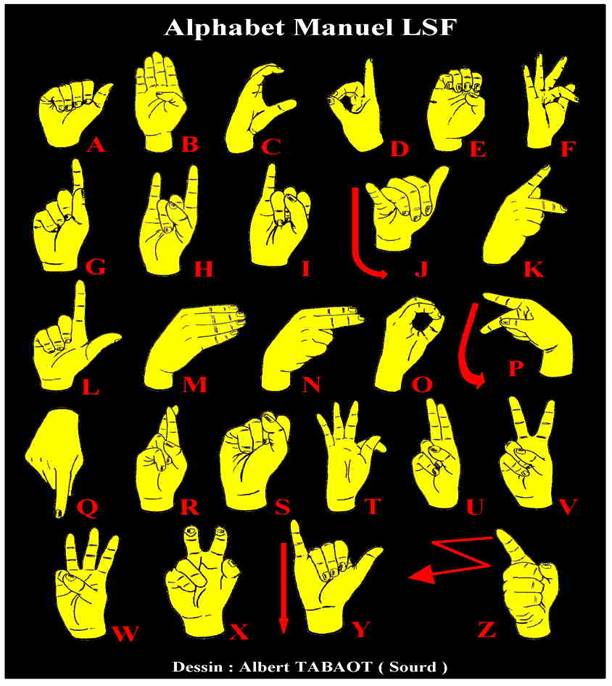
* Sign language is the language used by deaf and mute people. It is a combination of shapes and movements of different parts of the body. These parts include face and hands. The area of performance of the movements may be from well above the head to the belt level. Signs are used in a sign language to communicate words and sentences to audience. A gesture in a sign language is a particular movement of the hands with a specific shape made out of them.
* Facial expressions also count toward the gesture, at the same time. A posture on the other hand, is a static shape of the hand to indicate a sign. A sign language usually provides signs for whole words. It also provides signs of letters to perform words that don’t have a corresponding sign in that sign language. So, although sentences can be made using the signs for letters, performing with signs of words is faster. The sign language chosen for this project is the American Sign Language.

**1.3 AMERICAN STANDARD SIGN LANGUAGE**

* It is the most well documented and most widely used language in the world. American Sign Language (ASL) is a complex visual-spatial language that is

used by the Deaf community in the United States and English-speaking parts of Canada

* It is a linguistically complete, natural language. It is the native language of many Deaf men and women, as well as some hearing children born into Deaf families.
* ASL shares no grammatical similarities to English and should not be considered in any way to be a broken, mimed, or gestural form of English.
* American standard sign language was demonstrated around 1980 in United States and many other countries, to provide education for the people who have problem in speaking and hearing the words (the deaf and the dumb).
* A-S-S-L enables the dumb and def to speak with their hands which represent their tongue.

**1.3The signs are shown below:**  **Fig. 1.3.1 ASL Signs**

**2. Literature Survey**

**2.1 Speech Survey**

* To communicate with deaf and dumb person the existing methods are lip reading, writing down word and finger spelling.
* We are developing such a system that translates spoken language to sign language via hand assembly using speech recognition.
* The current stage of the process focuses on translating speech signal to American sign language.
* Speech is one of the worst analog signals in the world due to its instability and unpredictable
* Speech signal completely depends on the tone of its creature, and the tone of each creature depends on the frequency, age, artifice and so many other factors which make it very variable and instable.
* Speech signal is the result of vibration of larynx and pharynx along with the brain signal, which is different from one creature to another.
* Therefore, recognition of speech signal becomes a very complex task.

**2.2 AVAILABLE METHODS OF RECOGNITION**

* There are many methods available to recognize the identified features and classify them into categories. The most commonly used methods are HMM, ANN, ANFIS and MFCC.
* HMM: Hidden Markov Model
* ANN: Artificial Neural Network
* ANFIS: Adaptive Neuro-Fuzzy Inference System
* MFCC: MELLL Frequency Cepstral Coefficient

Our first task was to find out the way for recognizing speech. There are many methods available for speech recognition. Speech recognition can be done using:

1. Hidden Markov Models
2. Using Neural Networks
3. Dynamic Time Warping
4. Digital Signal Processing
5. Mell Frequency Cepstrum
6. Log –Likelihood
7. ANFIS

* 1. **Why We Choose Servo Motor?**

|  |  |  |  |
| --- | --- | --- | --- |
| **PARAMETER** | **STEPPER** | **SERVO** | **DC** |
| Supply Voltage | 5V | 4.8V | 12V |
| Loop Operation | Opened | Closed | Closed |
| Output Angle | 1.8deg | ≥170° |  |
| Running Current | 1.5A | 0.25A | 2.5A |
| Stall Torque | 1.86Kg/cm | 1.0kg/cm |  |

We choose servo motor because

1. Size is small.

2. Servo motor rotates at particular angle.e.g.

90 degree=1.50ms

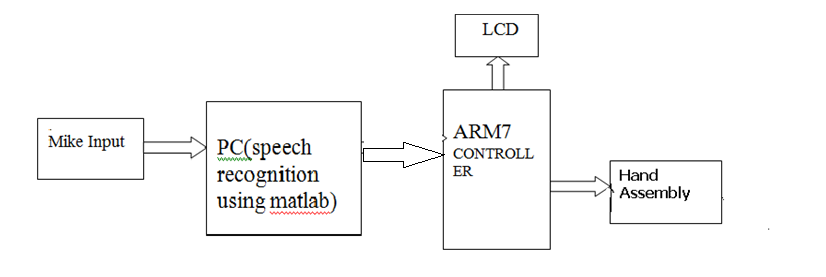
0 degree=1.25ms

180 degree=1.75ms

3. Also stall torque is less. **3. Specifications**

* Input Voice strength
* 3-4 KHz
* Input Duration
* 3 sec
* Recognized Alphabets (2D)
* 10 Alphabets
* A,B,G,H,I,L,S,U,W,X
* Number of words Recognized
* BALL, BUS, SHY, UGLY, WAIL, ALL, GAS, HI, etc
* Power Supply
* 5 Volt
* ARM 7 microcontroller(LPC2148)
* Operating Range 3 to 3.3V
* Frequency Range 0 to12 MHz
* LCD Display(16x2)
* Wooden hand assembly
* Mike

1. **Block Diagram**

****

**Fig. 4.1 Block Diagram**

**4.2Block Diagram Description**

**4.2.1. Mike Input**

We are using i-ball’s headphones to take input from user and to save it on computer. It converts user’s voice into analog signal. With the help of mike we can save user’s voice on computer.

**4.2.2. Computer**

On computer we are having MATLAB software. In MATLAB we have written MFCC program. This program we are using for speech recognition. In this program it will take input through mice and with the help of MFCC algorithm it will recognize spoken word. That recognized word will be displayed on computer and serially transferred to the controller part.

**4.2.3. Controller**

On the controller side we have ARM7. ARM7 receives input from computer serially. This input is displayed on LCD. One letter at a time. The letter displayed on LCD is represented by robotic hand assembly in ASL sign.

**4.2.4. ARM7 (LPC 2138) Controller**

The LP2138 microcontrollers are based on a 32/16 bit ARM7TDMI-S CPU with real-time emulation and embedded trace support, that combines the microcontroller with 32 kB, 64 kB, 128kB, 256 kB and 512 kB of embedded high speed Flash memory. A 128-bit wide memory interface and unique accelerator architecture enable 32-bit code execution at maximum clock rate. For critical code size applications, the alternative 16-bit Thumb mode reduces code by more than 30 % with minimal performance penalty. Due to their tiny size and low power consumption, these microcontrollers are ideal for applications where miniaturization is a key requirement, such as access control and point-of-sale. With a wide range of serial communications interfaces and on-chip SRAM options of 8/16/32 kB, they are very well suited for communication gateways and protocol converters, soft modems, voice recognition and low end imaging, providing both large

buffer size and high processing power. Various 32-bit timers, single or dual 10-bit

8 channel ADC(s), 10-bit DAC, PWM channels and 47 GPIO lines with up to nine edge or

level sensitive external interrupt pins make these microcontrollers particularly suitable for

industrial control and medical systems.

**4.2.5 LCD**

LCD has been used for the debugging purpose. LCD is displaying recognized word’s letter.LCD is getting input from ARM7.We are using 15pin LCD.

**4.2.6 Hand Assembly**

Hand is made up of wood and servo motors. We have chosen servo motor because of its light weight and torque handling capacity. 5 Servo motors are connected for finger movement. As we have used 1 servo motor for each finger, only 180 degree movement of each finger is possible. This 180 degree movement of each finger is in backward and forward direction. Because of this limitation we can only show ASL sings of 10 letters(A,B,G,H,I,L,S,U,W,X). So hand assembly will display only 10 letters n words made up of 10 letters.

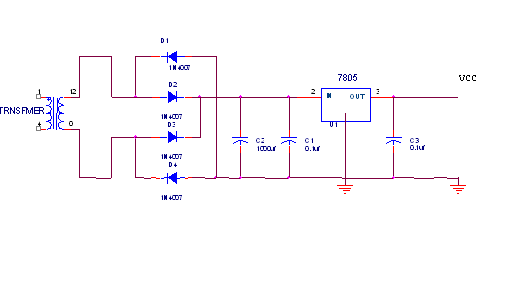
1. **Hardware System Design**

**5.1 Power Supply:**

The basic step in the designing of any system is to design the power supply required for that system. The steps involved in the designing of the power supply are as follows,

1) Determine the total current that the system sinks from the supply.

2) Determine the voltage rating required for the different components.

 **Fig.5.1. Power supply design**

Transformer selection we required 12V for relay.

Min Input for 7805 is,

Min = Drop across IC 7805 + Required Output voltage

= 3 V+ 5V

= 8 V

So at Input of 7805 we required 8 V with margin

Consider drop across diode 0.7V so 2 diode conducts drop is 1.4 V

= 1.4 V +8 V

= 9.4 V

So at secondary we required 10 V

For filter capacitor design

C= (Il \* t1)Vr

Vr = ripple voltage

So unregulated power supply is design for 10 V

Vr = ripple voltage 10% of output voltage

Vr = 1.0 V

Assume Il 100mA

C = Il

(Vr\*f)

= 100mA

(1 V\*100Hz)

= 1000 µF

So we select ~ 1000 µf capacitor

For diode design

PIV = Vm

E0 max=10+0.7

10.7V

E0min=10-0.7

9.3V

Vm = E0 max + 2 Vf

= 10.7 + 1.4 V

= 12.1 V

I0 = Il

2

= 116.2 mA

2

= 58.1 mA

Peak repetitive current

Ifm = 116.2mA( 8.6ms+1.2ms).

1.2ms

=833mA

From above specification diode 1N4007 is selected

PIV =100V

I = 1A

**5.1.1 Reasons for choosing Bridge rectifier:**

1. The TUF is increased to 0.812 as compared the full wave rectifier.
2. The PIV across each diode is the peak voltage across the load =Vm, not 2Vm as in the two diode rectifier

Output of the bridge rectifier is not pure DC and contains some AC some AC ripples in it. To remove these ripples we have used capacitive filter, which smoothens the rippled output that we apply to 7805 regulators IC that gives 5V DC. We preferred to choose capacitor filters since it is cost effective, readily available and not too bulky.

The value of the capacitor filter can be found by following formula

C = ( IL \* t1 )

Vr

**5.2.VOLTAGE REGULATOR (LM 1117):**

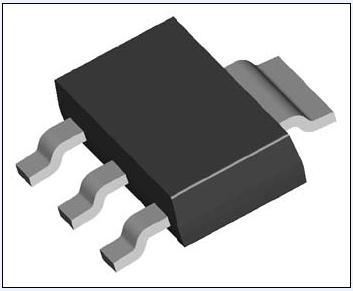
**Functional Description:**

The LM 1117 is a monolithic integrated fixed NPN type voltage regulator that can supply loads up to 1.0 A. The device is housed in the small surface mounted SOT223 package.

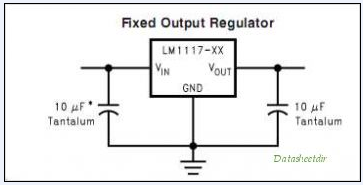
The IC is equipped with additional protection against overload, short circuit and over temperature. The LM 1117 GSV33 supplies a regulated output voltage of 3.3 V (2%). The LM 1117 GSV supplies an output voltage with 2% precision adjustable via an external voltage divider. The input voltage for the IFX 1117 GSV33 ranges from 4.5 V (= *V*Q+*V*DR) to 15 V for a load current of 800 mA, for the maximum load current of 1.0 A a minimum input voltage of 4.7 V is required. The drop voltage *V*DR ranges from 1.1 V to 1.4 V depending on the load current level. The device operates in the temperature range of *T*j = 0 to 125 °C.

**Specifications of LM1117:**

1. Output voltage 3.3 V or adjustable
2. 1.0 A output current
3. Low drop voltage < 1.2 V @ 800 mA
4. Short circuit protected
5. Over temperature protected
6. Operating range up to 15 V
7. Industrial type

****

**Fig.5.2.1.LM 1117**

****

**Fig.5.2.2.LM1117**

**5.3 MAX 232 (DS 14C232):**

The DS14C232 is a low power dual driver/receiver featuring an onboard DC to DC converter, eliminating the need for ±12V power supplies. The device only requires a +5V power supply. ICC is specified at 3.0 mA maximum, making the device ideal for battery and power conscious applications. The drivers’ slew rate is set internally and the receivers feature internal noise filtering, eliminating the need for external slew rate and filter capacitors. The device is designed to interface data terminal equipment (DTE) with data circuit-terminating equipment (DCE). The driver inputs and receiver outputs are TTL and CMOS compatible. DS14C232C driver outputs and receiver inputs meet TIA/EIA-232-E (RS-232) and CCITT V.28 standards.

###### Specifications of MAX 232:

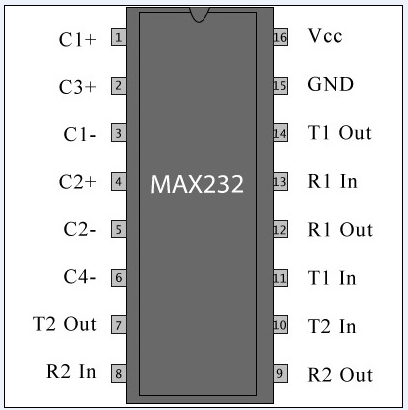
1. Pin compatible with industry standard MAX232, LT1081,

ICL232 and TSC232

1. Single +5V power supply
2. Low power—ICC 3.0 mA maximum
3. DS14C232C meets TIA/EIA-232-E (RS-232) and CCITT

V.28 standards

1. CMOS technology
2. Receiver Noise Filter
3. Package efficiency—2 drivers and 2 receivers
4. Available in Plastic DIP, Narrow and Wide SOIC packages
5. TIA/EIA-232 compatible extended temperature range option:
6. DS14C232T -40°C to +85°C
7. DS14C232E/J: -55°C to +125°C



**Fig.5.3.1. Pin Diagram of MAX232**

**5.4 ARM7(LPC2138):**

The LPC2138 microcontrollers are based on a 32 bit ARM7TDMI-S™ CPU with real-time emulation and embedded trace support, that combines the microcontroller with 512 kB of embedded high speed Flash memory.

Due to their tiny size and low power consumption, these microcontrollers are ideal for applications where miniaturization is a key requirement, such as access control and point-of-sale.

**Specifications:**

1. 32-bit ARM7TDMI-S microcontroller in a tiny LQFP64 package.
2. 32 kB of on-chip static RAM and 512 kB of on-chip Flash program memory. 128 bit wide interface/accelerator enables high speed 60 MHz operation.
3. In-System/In-Application Programming (ISP/IAP) via on-chip boot-loader software. Single Flash sector or full chip erase in 400ms and programming of 256 bytes in 1 ms.
4. Two (LPC2138) 8 channel 10-bit A/D converters provide a total of up to 16 analog inputs, with conversion times as low as 2.44 s per channel.
5. Single 10-bit D/A converter provide variable analog output. (LPC2132/2138 only)
6. Two 32-bit timers/counters (with four capture and four compare channels each), PWM unit (six outputs) and watchdog.
7. Real-time clock equipped with independent power and clock supply permitting extremely low power consumption in power save modes.
8. Multiple serial interfaces including two UARTs (16C550), two Fast I2C (400 Kbit/s), SPI™ and SSP with buffering and variable data length capabilities.
9. Vectored interrupt controller with configurable priorities and vector addresses.
10. Up to 47 of 5 V tolerant general purpose I/O pins in tiny LQFP64 package.
11. Up to nine edge or level sensitive external interrupt pins available.
12. 60 MHz maximum CPU clock available from programmable on-chip Phase-Locked Loop (PLL) with settling time of 100 microseconds.
13. On-chip crystal oscillator with an operating range of 1 MHz to 30 MHz
14. Power saving modes include Idle and Power-down.
15. Individual enable/disable of peripheral functions as well as peripheral clock scaling down for additional power optimization.
16. Processor wake-up from Power-down mode via external interrupt.

**Applications:**

1. Industrial control
2. Medical systems
3. Access control
4. Point-of-sale
5. Communication gateway
6. Embedded soft modem
7. General purpose applications.

**Device Information:**

Device…………………………………LPC2138

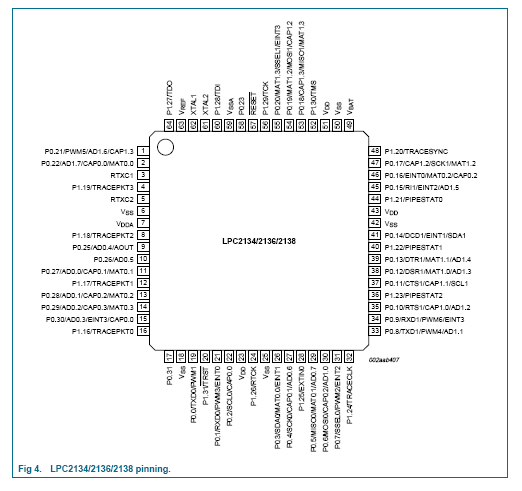
No. of pins………………………….…64

On-chip RAM………………………....32 kB

On-chip FLASH………………………512

No. of 10-bit AD Channels……………16

No. of 10-bit DA Channels…………....

****

**Fig5.4.1.LPC2138 Pinning**

**FunctionalDiagram:**

**Fig.5.4.2.Functional diagram of ARM**

**5.5 LCD (16x2):**

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over [seven segments](http://www.engineersgarage.com/content/seven-segment-display) and other multi segment [LED](http://www.engineersgarage.com/content/led)s. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even [custom characters](http://www.engineersgarage.com/microcontroller/8051projects/create-custom-characters-LCD-AT89C51) (unlike in seven segments), [animations](http://www.engineersgarage.com/microcontroller/8051projects/display-custom-animations-LCD-AT89C51) and so on.



**Fig.5.5.1.LCD Display**

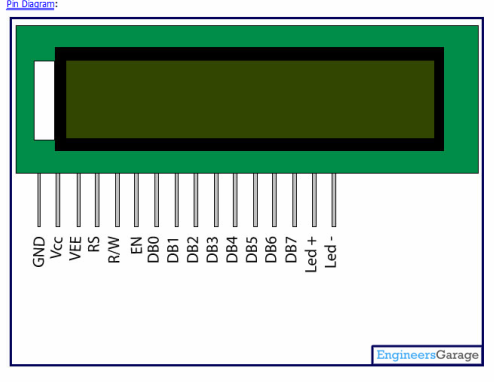
A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD.

**Pin Description:**

|  |  |  |
| --- | --- | --- |
| **Pin No** | **Function** | **Name** |
| 1 | Ground (0V) | Ground |
| 2 | Supply voltage; 5V (4.7V – 5.3V) | Vcc |
| 3 | Contrast adjustment; through a variable resistor | VEE |
| 4 | Selects command register when low; and data register when high | Register Select |
| 5 | Low to write to the register; High to read from the register | Read/write |
| 6 | Sends data to data pins when a high to low pulse is given | Enable |
| 7 | 8-bit data pins | DB0 |
| 8 | DB1 |
| 9 | DB2 |
| 10 | DB3 |
| 11 | DB4 |
| 12 | DB5 |
| 13 | DB6 |
| 14 | DB7 |
| 15 | Backlight VCC (5V) | Led+ |
| 16 | Backlight Ground (0V) | Led- |
|  |  |  |
|  |  |  |
|  |  |  |

**Table 5.5.1 Pin Description of LCD**

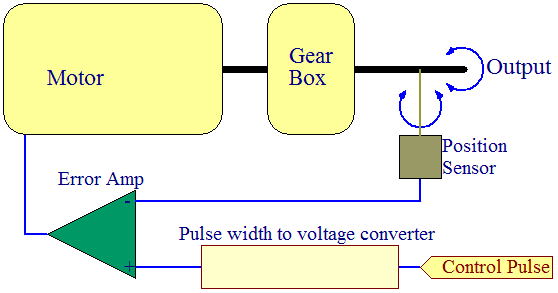
****

**Fig 5.5.2.LCD Pin Description Diagram**

**5.6 Servo motor:**

Servo motor consists of several main parts, the motor and gearbox, a position sensor, an error amplifier and motor driver and a circuit to decode the requested position. Figure 1 contains a block diagram of a typical servo motor unit.

The radio control receiver system (or other controller) generates a pulse of varying length approximately every 20 milliseconds. The pulse is normally between 1 and 2 milliseconds long. The length of the pulse is used by the servo to determine the position it should rotate to.

**** **Fig.5.6.1 Servo Motor Block Diagram**

Starting from the control pulse we will work though each part of the diagram and explain how it all fits together. Once we have gone through how the servo works we will investigate how the control pulses can be generated with a microcontroller.

**Pulse width to voltage converter**

The control pulse is feed to a pulse width to voltage converter. This circuit charges a capacitor at a constant rate while the pulse is high. When the pulse goes low the charge on the capacitor is fed to the output via a suitable buffer amplifier. This essentially produces a voltage related to the length of the applied pulse.

The circuit is tuned to produce a useful voltage over a 1ms to 2ms period. The output voltage is buffered and so does not decay significantly between control pulses so the length of time between pulses is not critical.

**Position Sensor**

The current rotational position of the servo motor output shaft is read by a sensor. This is normally a potentiometer (variable resistor) which produces a voltage that is related to the absolute angle of the output shaft.

The position sensor then feeds its current value into the Error Amplifier which compares the current position with the commanded position from the pulse width to voltage converter.

**Error Amplifier**

The error amplifier is an operational amplifier with negative feedback. It will always try to minimize the difference between the inverting (negative) and non-inverting (positive) inputs by driving its output is the correct direction. The output of the error amplifier is either a negative or positive voltage representing the difference between its inputs. The greater the difference the greater the voltage.

The error amplifier output is used to drive the motor; If it is positive the motor will turn in one direction, if negative the other. This allows the error amplifier to reduce the difference between its inputs (thus closing the negative feedback loop) and so make the servo go to the commanded position.

The servo normally contains a single integrated circuit and a hand full of discreet components to implement the entire control system.

**Controlling a Servo Motor with a Microcontroller**

From the above we can determine that we need to generate a pulse approximately every 20ms although the actual time between pulses is not critical. The pulse width however must be accurate to ensure that we can accurately set the position of the servo.

**PWM modules**

Many microcontrollers are equipped with PWM generators and most people initially consider using these to generate the control signals. Unfortunately they are not really suitable.

The problem is that we need a relatively accurate short pulse then a long delay; and generally you only have one PWM generator share between several servos which would require switching components outside the microcontroller and complicate the hardware.

The PWM generator is designed to generate an accurate pulse between 0% and 100% duty cycle, but we need something in the order of 5% to 10% duty cycle (1ms/20ms to 2ms/20ms). If a typical PWM generator is 8 or 10 bits say, then we can only use a small fraction of the bits to generate the pulse width we need and so we lose a lot of accuracy.

**Specifications of VTS 05A:**

|  |
| --- |
| 1. Control System: Pulse width control, 1500μs neutral 2. Operating Voltage: 4.8V～6.0V (DC) 3. STD Direction: Counter clockwise/pulse traveling 800 to 2200 μsec. 4. Test Voltage: at 4.8V at 6.0V 5. Operating Speed: 0.19sec/60°at no load 0.17sec/60° at no load 6. Stall Torque: ≥1.0kgf.cm(13.89 oz/in) ≥1.2kgf.cm(16.66 oz/in) 7. Running Current: ～0.25A ～0.3A   8. Stall Current: ～0.7A ～0.8 A  9. Output Angle: ≥170°  10.Dead Bank Width: 5μsec  Picture of a servo  :  **fig 5.6.1 Top view of servo motor**    Picture of servo guts **Fig 5.6.2 Servo Motor** |

1. **Software System Design**

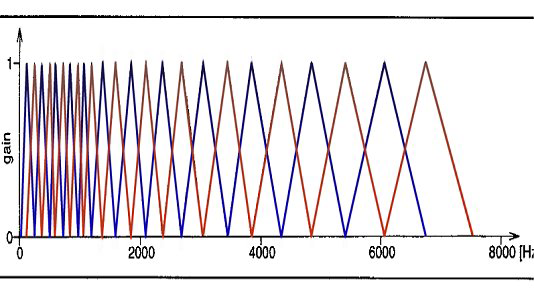
**MATLAB Working for Speech Reorganization Using MFCC**

**6.1 MELLL-FREQUENCY TRANSFORMATION**

* According to psychophysical scientists human hearing perception of frequency contents is not linear. It is linear up-to 1000 Hz and logarithmic over 1000 Hz. This logic can be implemented by using this formula,

Melll-frequency =2595\*log (1+linearfrequency/700)

* Each region on the basilar membrane acts as a filter bank.
* There are more receptors for frequencies from 0-1KHz and the number decrease rapidly there on.
* Melll filter banks are used to simulate this Membrane.

****

**Fig.6.1.1. Filter design**

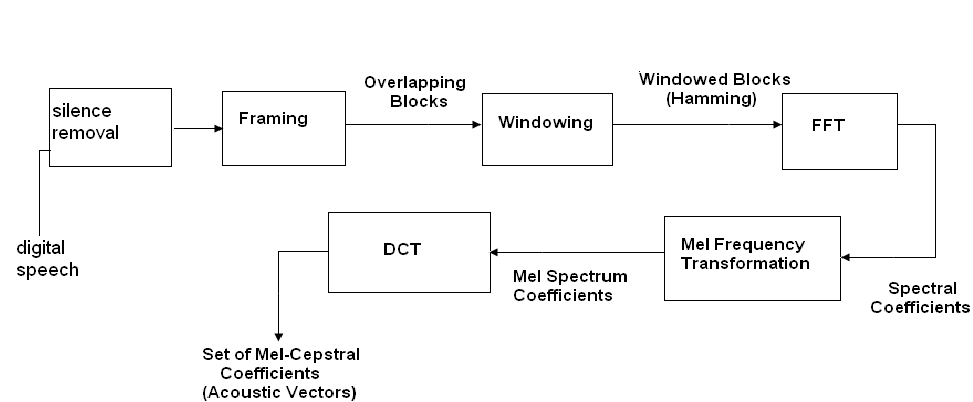
**SPEECH RECOGNITION BY MFCC**

* There are two phase involved in speech recognition process:

1) TRAINING

2) RECOGNITION

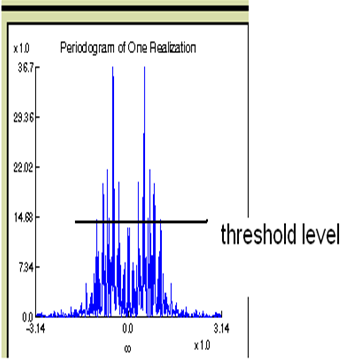
The overview of the MFCC process:

****

**Fig.6.1.2 Feature Extraction**

**6.2 SILENCE REMOVAL PROCESS**

* The silence removal process detects the maximum amplitude of original signal.
* Depending upon the maximum amplitude, the threshold level is decided.
* The samples below the threshold level from both the sides are removed and silence-free signal will be obtained.

****

**Fig.6.2.1. Silence remove**

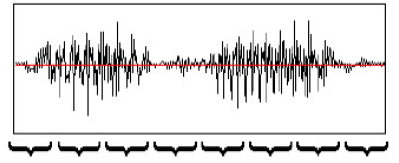
**6.3 FRAMING:**

Framaning process enable to group the samples as per to requirements. Depending on the predecided size of the frames, consequtive samples will groups in packet of frames.

The number of frmes are very important to decide. Based on the number of frames and size of each frame , particular formula will help to deside which sample can place in which frame.

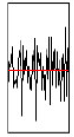
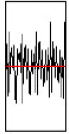
After framing, overlapping of frames helps to have perfect control of each and every sample, and avoid any loss of information in this case.

Framing process properly explained in figure below.



Speech signal with time interval of 10msec



**Fig.6.3.1.Framing**

**6.4 WINDOWING**

* Procedure:

In each frame, multiply the N samples with Hamming window, H(n)

Where

H(n)=0.54-0.46\*cos(2\*pi\*n/(N-1));

Y[n]=x[n]\*H[n]

* Reason:

To decrease discontinuities at the both ends of a frame by approximating to zero. And in the remaining part of the wave bringing regularity by multiplying the wave by Hamming window function.



**Fig.6.4.1.Window shapes**

**6.5 FFT**

* To analyze the spectrum of the speech wave, apply Discrete Fourier Transform (DFT).
* Every complex wave, like a speech wave, is a composition of some simple trigonometric waves.
* To find the simple signals to visualize the spectrum, we are applying DFT.
* FFT is the best available mathematical method to apply DFT.
* Each X[k] is squared since the Fourier transform produces complex numbers.
* A fast Fourier transform (FFT) is an efficient algorithm to compute the [discrete Fourier transform](http://en.wikipedia.org/wiki/Discrete_Fourier_transform) (DFT) and it’s inverse. There are many distinct FFT algorithms involving a wide range of mathematics, from simple [complex-number arithmetic](http://en.wikipedia.org/wiki/Complex_number) to [group theory](http://en.wikipedia.org/wiki/Group_theory) and [number theory](http://en.wikipedia.org/wiki/Number_theory)

An FFT computes the [DFT](http://en.wikipedia.org/wiki/Discrete_Fourier_transform) and produces exactly the same result as evaluating the DFT definition directly; the only difference is that an FFT is much faster. (In the presence of [round-off error](http://en.wikipedia.org/wiki/Round-off_error), many FFT algorithms are also much more accurate than evaluating the DFT definition directly, as discussed below.)

Let *x*0, *xN*-1 be [complex numbers](http://en.wikipedia.org/wiki/Complex_number). The DFT is defined by the formula

 X_k =  \sum_{n=0}^{N-1} x_n e^{-{i 2\pi k \frac{n}{N}}}
\qquad
k = 0,\dots,N-1. 

**6.6 DISCRETE COSINE TRANSFORM (DCT)**

* Usually, the wave should be in time domain. So, to bring back the speech wave into time domain, we apply Inverse Discrete Fourier Transform. Mathematically this can be implemented by DCT.

**6.7 COMPARISON AND DECISION**

* Comparing the input speech with the data base and finally decision is take place base on following concepts :
* VECTOR QUANTIZATION
* EUCLIDIAN DISTANCE

**6.7.1 VECTOR QUANTIZATION**

Vector quantization is a classical [quantization](http://en.wikipedia.org/wiki/Quantization_%28signal_processing%29) technique from [signal processing](http://en.wikipedia.org/wiki/Signal_processing) which allows the modeling of probability density functions by the distribution of prototype vectors.

* Vector quantization: The process of compressing the signal in such a way that the compressed signal has complete feature of original signal.
* Vector quantization is very useful process which helps to minimize the memory requirement and makes the process of comparison very easy with minimum amount of time.
* A simple training algorithm for vector quantization is:

1. Pick a sample point at random
2. Move the nearest quantization vector centroid towards this sample point, by a small fraction of the distance
3. Repeat

A more sophisticated algorithm reduces the bias in the density matching estimation, and ensures that all points are used, by including an extra sensitivity parameter:

1. Increase each centroid's sensitivity by a small amount
2. Pick a sample point at random
3. Find the quantization vector centroid with the smallest <distance-sensitivity>
   1. Move the chosen centroid toward the sample point by a small fraction of the distance
   2. Set the chosen centroid's sensitivity to zero
4. Repeat

**6.7.2 Application of vector quantization:**

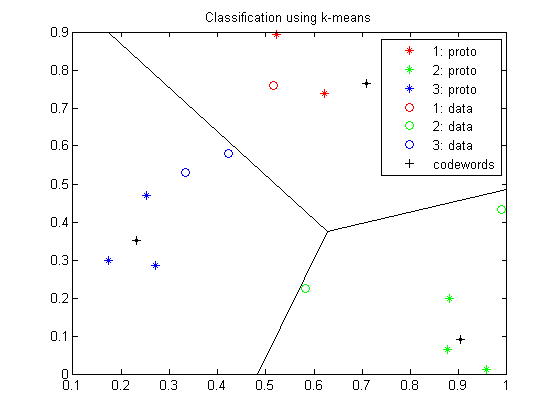
Vector quantization is used for lossy data compression, lossy data correction and density estimation.

Lossy data correction, or prediction, is used to recover data missing from some dimensions. It is done by finding the nearest group with the data dimensions available, and then predicting the result based on the values for the missing dimensions, assuming that they will have the same value as the group's centroid.

For [density estimation](http://en.wikipedia.org/wiki/Density_estimation), the area/volume that is closer to a particular centroid than to any other is inversely proportional to the density (due to the density matching property of the algorithm).

**VECTOR QUANTIZATION**





**Fig. 6.7.2.1.Vector quantization**

**6.7.3 EUCLIDIAN DISTANCE**

* The process of comparison includes the calculation of Euclidian distance for each original data, which has been stored in data base and the input speech during the test.
* The final decision and the output are based on the minimum distance between one of the stored speech and input speech.

**6.8** **Algorithm**

Step 1: Start.

Step 2: Record the word.

Step 3: Save it.

Step 4: Image display – Wave of saved voice.

Step 5: Image display-Wave of saved voice, after removing noise.

Step 6: Press 1 to proceed for further speech detection process or ‘0’ to

stop.

Step 7: After pressing 1 (using MFCC software) it recognizes the word

and display it on PC screen.

Step8: Letters of detected word is transferred to the LCD serially and accordingly

hand assembly will represent it in ASL signs.

Step9: After complete representation of detected word in ASL sign using hand assembly,

hand assembly will come in its original position and stop.

Step 10: Stop.

**6.9Flow chart**

START

**7. PCB Design**

STOP

HAND ASSEMBLY COME TO ITS ORIGINAL POSITION AFTER COMPLETE REPRESENTATION

LETTER DISPLYED ON LCD IS REPRESENED BY HADN ASSEMBLY IN ASL SIGN

DETECTED DATA SERIALLY TRANSFER TO LCD

0PRESSED

1PRESSED

DETECTED WORD OR LETTER DISPLAY ON PC SCREEN

PRESS 1’ OR’0

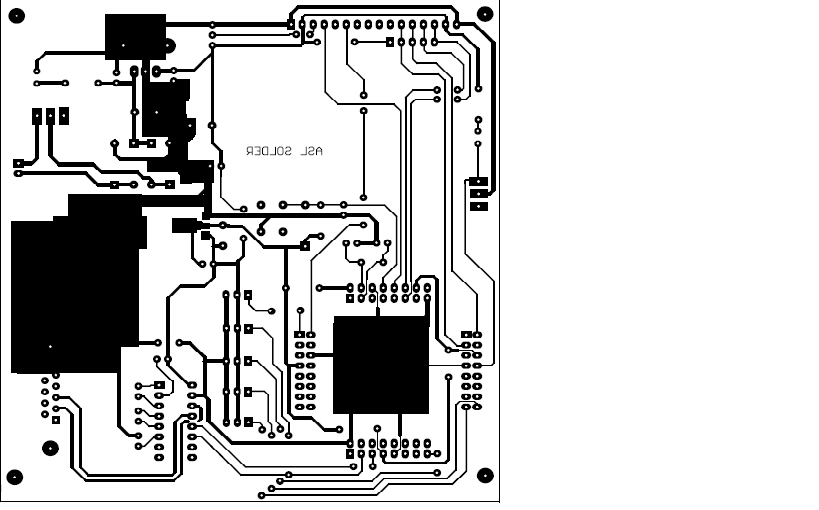
?

DISPLAY IMAGE- WAVE OF INPUT VOICE AFTER REMOVAL OF NOISE

DISPLAY IMAGE –WAVE OF INPUT VOICE

SAVE THE RECORDED VOICE INPUT

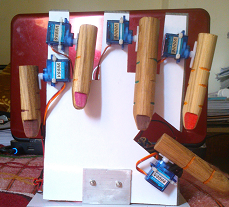
RECORD THE INPUT VOICE



**Fig7.1.PCB Design**

**8. Result and Performance Evaluation**

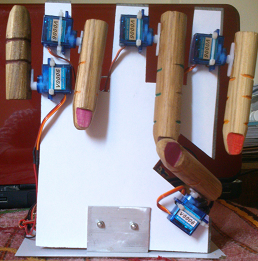
* Software result
* In MFCC feature extraction and matching done.
* After matching character or word pass it serially to hardware.
* Hardware result
  + - LCD display.
    - Hand movement of later displayed on LCD.
* Performance



**fig 8.1 Representation of A**

****

**fig 8.2 Representation of B**



**fig 8.3** **Representation of I**

**9. Application**

* Public places like airports, railway stations and counters of banks, hotels etc. (where there is communication between different people.)
* A mute person can deliver a lecture using it.

(Assuming the fact that we are able to convert whole of American Sign Language into spoken English, we can manufacture a handy and portable hardware device having this translating system built in as a chip.

* In Primary Education of deaf and dumb people. (This will almost bridge the communication gap present between the deaf community and the normal world.)

**FUTURE SCOPE**

* By American Standard Sign Language with speech recognition, it is possible to speak directly with a robot without any physical engagement to communicate with the dumb and deaf easily
* It is easily possible to make a software package to automatically install in video and movie services with the help of set top box and record the speech words and at the time of testing (playing the movie) represent the sign language by animated hand at the corner of screen.

**10. Bill of material**

|  |  |  |
| --- | --- | --- |
| **Component** | **Quantity** | **Cost(Rs)** |
| LPC2138 | 1 | 350 |
| LCD | 1 | 150 |
| Servo motor | 5 | 2500 |
| LM 7805 | 1 | 45 |
| LM 1117 | 1 | 40 |
| MAX 232 | 1 | 40 |
|  | Total | 3125/- |

**11. Bibliography**

* References:
  + **Speaker Identification Using Melll Frequency Cepstral Coefficients**

Md. Rashidul Hasan, Mustafa Jamil, Md. Golam Rabbani Md. Saifur Rahman

28-30 December 2004, Dhaka, Bangladesh

* + Andrew S.Sloss,”Arm system Developer”
* ***PAPERS***
* Ibrahim Patel1, Dr. Y. Srinivasa Rao2**,” AUTOMATED SPEECH**

**RECOGNITION APPROACH TO CONTINUOUSCUE-SYMBOLS**

**GENERATION**”, International Journal of Power Control Signal and

* Computation (IJPCSC) Vol. 2 No. 1.
* Mahdi Shaneh, and Azizollah Taheri “**Voice Command Recognition System Based on MFCC and VQ Algorithms**”, World Academy of Science Engineering and Technology’
* ARM user guide by ***“Andrew Sloss”***
* Embedded systems by ***“Rajkamal”***
* Websites:
  + [www.ieeexplore.com](http://www.ieeexplore.com)
  + [www.datasheet.com](http://www.datasheet.com)

**CONCLUSION**

This project was meant to be a prototype to check the feasibility of recognizing sign languages using speech recognition. More words and sentences can be recognized by developing some complex algorithm.

**12. Data Sheets**