DEVELOPMENT OF ALEXA VOICE SERVICES SOFTWARE DEVELOPMENT KIT FOR SPEECH RECOGNITION ENGINE FOR INTERNET OF THINGS

A Thesis Submitted in partial Fulfillment of the Requirement for the Award of the Degree of

MASTER OF ENGINEERING

In

Electronics and Communication

Submitted By

NIDHI SHARMA

Roll. No.: 801661014

Under Supervision of

DR. HEMDUTT JOSHI

Associate Professor

ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT

THAPAR INSTITUTE OF ENGINEERING &

TECHNOLOGY (A DEEMED TO BE UNIVERSITY),

PATIALA, PUNJAB INTEL TECHNOLOGY INDIA PVT.

LTD. BANGALORE- 560103, KARNATAKA

JULY, 2018

DECLARATION

I, Nidhi Sharma hereby declare that the work presented in this thesis entitled **"Development of Alexa Voice Services Software Development Kit for Speech Recognition Engine for Internet of Things"** in fulfillment of the requirement for the award of degree of Master of Engineering (ECE) submitted at Electronics and Communication Engineering Department, Thapar Institute of Engineering & Technology (Deemed to be University), Patiala is an authentic record of work carried out under supervision of **Dr. Hemdutt Joshi (Associate Professor, ECED)** from July 2017 to June 2018. The matter presented in this has not been submitted either in part or full to any other university or institute for the award of any other degree.

The numbers, facts and figures in this thesis report are not Intel prescribed. These are achieved by me during my own experiments at Intel.

Date: 12/7/18

Nidhi Sharma.

Nidhi Sharma
801661014

Date: 12/7/18

Dr. Hemdutt Joshi
Associate Professor
Electronics And Communication Engineering Department
Thapar Institute Of Engineering & Technology
(A Deemed To Be University), Patiala, Punjab

CERTIFICATE

This is to certify that **Nidhi Sharma (801661014),** a student of M.E. (ECE), Thapar Institute of Engineering and Technology, Patiala, has successfully completed one-year (August 2017 – July 2018) internship program in **Intel Technology India Pvt. Ltd., Bangalore**. Her title of dissertation is **"Development of Alexa Voice Services Software Development Kit for Speech Recognition Engine for Internet of Things".** During the period of her internship program, she was punctual and hardworking.

I wish her every success in life.

Date: July 10, 2018

Vishnu Balraj

Engineering Manager

IOTG Intel Technology India Pvt. Ltd.

Bangalore, India

ACKNOWLEDGEMENT

I would like to convey my deep sense of gratitude to my project guide, **Dr. Hemdutt Joshi, Associate Professor, ECED** who is a constant source of motivation and firm support in carrying out this project. The support and supervision that he gave has helped me to progress in the project. His co-operation is highly appreciated and I highly oblige to him for his valuable comments and moral support during this research period.

I would also like to thank my mentor, **Sitanshu Nanavati, Software Engineer, Intel Technology India Private Limited, Bangalore** and my manager **Vishnu Balraj, Engineering Manager, Intel Technology India Private Limited, Bangalore** for their esteemed guidance, valuable suggestions and time throughout my internship. Their guidance and vast knowledge directed me to accomplish critical tasks smoothly.

Also, my special gratitude to my family for their constant support and motivation.

Date: ……..

Place: …… Nidhi Sharma

ABSTRACT

In 1990s, Internet connectivity began. Internet established a link between people through PC. With the growth in area of Electronics, Communication, Information Technology and Computer Science in terms of technology, reliability, efficiency and availability IoT also started growing rapidly. Cisco predicts that the total number of connected devices in IoT will reach up to 50 Billion (approx.) by year 2020. Google Home, Amazon Echo, Apple Siri have been added to the journey of IoT with the evolution of new technology and algorithms in a decade and this journey will continue over many decades. Speech Recognition provides a capability to user to interact with device by Speech Interface.
People with hearing disability can use speech recognition to convert a telephone caller's speech into text format.

In future, advancement and expansion of Speech Recognition techniques and Artificial Intelligence, Nueral Networks, Echo Cancellation, Beam Forming and Noise Reduction techniques will enhance the quality and efficiency of Speech Recognition Engines. Finally result shows that by improving some key features and parameters such as Wake Word False Rejection Rate, Wake Word Detection Delay, Wake Word Detection Delay, Multi Room
Music (MRM) playback, Response Accuracy Rate and Audio Voice Quality of Speech Recognition Engine better user experience can be achieved.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

AVS	Amazon Voice Service
dB	Decibel
dBA	dB A-weighted
dBC	dB C-weighted
DUT	Device Under Test
FAR	False Acceptance Rate

FRR	False Rejection Rate
RAR	Response Accuracy Rate
SNR	Signal to Noise Ratio
SPL	Sound Pressure Level
TTS	Text-To-Speech
WW	Wake Word
WWDD	Wake Word Detection Delay
WWE	Wake Word Engine
WWDD	Wake Word Detection Delay
FFRS	Far-Field Reference Solution
IoT	Internet of Things
MRM	Multi Room Music
SDK	Software Development Kit
ACL	Alexa Communication Library
ADSL	Alexa Directive Sequencer Library
HTTP	Hyper Text Transport Protocol
TLS	Transport Layer Security
Wi-Fi	Wireless Fidelity
API	Application Programmed Interface
PWM	Pulse Width Modulation
GPIO	General Purpose Input Output
DSP	Digital Signal Processor

SRAM	Static Random Access Memory
UART	Universal Asynchronous Receiver Transmitter
SPI	Serial Peripheral Interface
NNA	Neural Network Accelerator
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
RAM	Random Access Memory
RTOS	Real Time Operating System
RFID	Radio Frequency Identification
WSN	Wireless Sensor Networks
GUI	Graphical User Interface
NLMS	Normalized Least Mean Square
LCMV	Linearly Constrained Minimum Variance
GSLC	Generalized Side Lobe Canceller
SNR	Signal to Noise Ratio
FIR	Finite Impulse Response
IDFT	Inverse Discrete Fourier Transform
ALE	Adaptive Line Enhancer
WF	Wiener Filter
LogMMSE	Log Minimum Mean Square Error
ASR	Automatic Speech Recognition
RNN	Recurrent Neural Networks
DNN	Deep Feedforward Neural Networks

RNN	Recurrent Neural Networks
ANN	Artificial Neural Networks
HMM	Hidden Markov Model
DTW	Dynamic Time Warping
LPC	Linear Predictive Coding
M2M	Machine-to-Machine

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Internet of Thing (IoT) is network of physical devices connected in to share and exchange data among themselves. Over a network, these connected devices can collect and transfer the data with help of different communication and networking technologies and then have ability to exchange the data to produce some useful result and action. Machine to Machine (M2M) is part of IoT, which uses wireless networks to connect the devices over internet with minimal human intervention. IoT networks can include devices of different size and types, from a smart watch to a large smart building and from animals to people, small home appliances to heavy industrial machinery. Each things or objects in IoT having unique address can interact with each other to reach some common or individual goal. Research and development in area of IoT is very vast and challenging.

IoT is mixture of various networking, wired/wireless communication, hardware, software technologies. IoT includes Machine to Machine (M2M), Wireless Sensor Networks(WSN), WiFi, GPS, 2G/3G/4G, BLE/Bluetooth, Ethernet, ZigBee, IPv6, Microcontroller, Microprocessors, Embedded Systems, Cloud Computing, Real Time operation/monitoring and many more. IoT is based on collecting, storing, retrieving and processing the data, which is accomplished by communication between connected devices. Connectivity provided by IoT will enhance the quality of life in terms of security, automation, real time operation, remote monitoring, data sharing, efficiency, independence from human intervention, education and healthcare.

1.2 EVOLUTION OF INTERNET OF THINGS

Kevin Ashton first introduced the term “Internet of Things” in 1999 at Auto-ID Centre at MIT. In 1990s, Internet connectivity began. Internet act as link between people through PC. Then in decade 2000s banking, public services, government, and education sectors raised a demanded connection to internet. So in fulfilling these demands internet migrated from internet to “internet of services” which includes e banking, e-campus, e-commerce, e-government any many more.
Then real time positioning, monitoring and tracking came into demand rapidly in early 2010s.

Gradually internet services became more efficient and reliable with new technologies. With this growth in area of Electronics, Communication, Information Technology and Computer Science in terms of technology, reliability, efficiency and availability IoT also started growing rapidly.

Existing demand in various fields as, but not limited to healthcare, home/industrial automation, education, smart city to smart vehicle and smart building to smart industry, energy, agriculture, environment will drive the rapid development of IoT in upcoming decades. Cisco predicts that the total number of connected devices in IoT will reach up to 50 Billion (approx.) by year 2020.

1.3 COMPONENTS OF INTERNET OF THINGS ARCHITECTURE

IoT architecture can be divided into 4 main parts:

1. Sensors, Actuators

2. Gateways

3. Internet Connectivity 4. Cloud Services, Applications

Layers in IoT:

1. Perception Layer: (RFID tags, WSN, sensors, actuators)

Perception layer is lowest layer of IoT, which is responsible for receiving or collecting the data from environment through sensors, actuators, RFID tags and other connected devices in IoT architecture.

2. Network Layer: (Gateways, wireless and wired communication network, 3G/4G/5G, LAN, ZigBee, Bluetooth, Wi-Fi)

Network layer is middle layer, which is responsible for receiving data from perception layer and transmitting this data to application layer. This layer requires large storage capacity for storage of large amount of data collected from sensors, RFID tags and actuators.

3. Application Layer: (User Interface, application)

Application layer top most layer of IoT, which is responsible for effective utilization of the data collected and delivering various interfaces, applications to user. Applications are delivery point of

any services to user in IoT. Application can be of type logistics, industrial, healthcare, automation, transportation, environment monitoring.

1.4 APPLICATIONS OF INTERNET OF THINGS

A. ***Smart Home:*** Smart home is one in which devices can interact with other and to their surroundings. It can include light control system, smart lock and many more. It provides security, real time monitoring and control, efficient energy management, automation.

B. ***Wearable:*** Wearables are for healthcare, fitness and entertainment. These are small size devices with ultra-low power consumption. In healthcare, these can be used to track sleep, heartbeat, blood pressure, workout and many more.

C. ***Retails:*** IoT will be helpful in retails by providing automation and monitoring to notify shortage of supply, for tracking goods, real time information exchange about goods and services among suppliers and retailers. It can play vital role in SCM (Source Chain Management).

D. ***Smart City:*** To provide automated transportation, smart surveillance, environmental monitoring, efficient energy management, smart waste and recycling management system, smart streetlights, vehicle-parking management, smart water distribution and security, traffic management.

E. ***Health Care:*** IoT will provide real time remote patient health monitoring, optimize surgical workflow, medication adherence, quality and satisfactory medical care to patient. It will make use of wearables and different sensors and networks to monitor heartbeat, blood pressure, glucose level, ECG, respiratory rate, humidity, temperature and will provide appropriate advice to patient remotely.

F. ***Agriculture:*** IoT based smart farming will enable the farmers to increase productivity and reduce waste. It will use different sensors to measure temperature, humidity, Sun light, soil moisture for remote monitoring of crop fields. It will provide the method for efficient and optimized use of water, fertilizers and other resources.

G. ***Transportation and Logistics:*** IoT will improve accuracy in shipping, delivery, receiving and inventory for logistics. It will make use of IoT enabled mobiles and devices to track monitor equipment, vehicles, inventory data and other physical assets.

H. Industrial Automation: Machine-to-Machine (M2M) communication, predictable and fault tolerant real time closed loop control systems. Industrial automation will leverage the IoT technologies like RFID, WSN to monitor, control and automate the industrial production & operation.

I. Energy Management: It will include smart power grids for automatic collection of data to analyze behavior of electricity consumption and supply to enhance the efficient and effective use of electricity. Energy management will leverage IoT technologies for efficient management of conventional as well as non-conventional energy resources.

J. Digital Oil field: IoT will provide digitalization of oilfields to automate the operation and control in order to increase the production Oil and gas companies will leverage the IoT along with data analytics to improve and optimize the production and efficiency.

K. Smart Environment: It will include forest fire detection, earthquake early detection, snow level monitoring, air pollution monitoring & control, humidity and temperature monitoring, pollution level in sea monitoring & control detection of water leakages in pipelines, water level in rivers and reservoirs.

1.5 INTRODUCTION TO SPEECH RECOGNITION

Speech recognition is a process in which a machine recognizes the spoken words and phrases and then converts these into machine-readable format. It transforms the spoken words into text. Conversion of spoken words or audio into text using speech recognition enables the user to control the digital devices through voice interaction/speaking instead of using graphical user interface (GUI), keyboard, buttons and touch screen etc. Speech Recognition Engine performs speech recognition process.

Following are basic definitions to understand speech recognition:

1. ***Utterance:*** Utterance is stream of speech between two periods of silence. Utterance can be a single word, multi word, phrase, complete or multiple sentence.
2. ***Speaker Dependent:*** Speaker dependence is degree up to which speech recognition engine requires knowledge of voice characteristics of speaker's individual voice. Speaker dependent system is designed around specific speaker. These systems are usually more accurate for

specific speaker but less accurate for other speakers. In these systems user need to train the system to his voice.

3. ***Speaker Independent:*** Speaker independent system is designed around variety of speakers. In these systems user does not need to train the system to his voice.

4. ***Accuracy:*** Most important measurement of accuracy is whether the desired/expected result has occurred. Accuracy can also be measured as the ability of recognition engine to recognize the utterance exactly as spoken. Acceptable accuracy of a system will depend upon its application.

5. ***Vocabularies:*** Vocabulary is collection of words that can be recognized by speech recognition system. Each entry in vocabulary does not have to be a single word, it can be multiple words, phrase or complete sentence. We can use the grammar to specify particular syntax and to set some rules that would be used in speech recognition process by speech recognition engine. Speech recognition engine will compare the utterance against the words in vocabulary, if this does not match or present in grammar then engine will not recognize them correctly.

6. ***Acceptance and Rejection:*** Acceptance and rejections can be the two possible results of any utterance processed by speech recognition engine. An utterance is accepted if engine recognized it correctly, otherwise it is rejected.

7. ***Training:*** Speech recognizing engine can be trained if it has ability to adapt to a speaker. Training to speech recognition engine will increase its accuracy.

1.6 EVOLUTION OF SPEECH RECOGNITION ENGINE

In 1952, the first speech recognizing system "Audrey" was developed at Bell laboratories. It was capable to recognize and understand the digits (0-9). Then in 1962, IBM designed a speech recognition machine "Shoebox", which could understand and recognize 16 words spoken in English. Every individual has different voice and pronunciation can be very inconsistent. Spoken language does not have high level of standardization unlike text. Therefore, there can be great variation in spoken words based on emphasis, speed, gender, region etc. Therefore, this was a greatest challenge for speech recognition engine. Then labs in different countries like Japan, England, and United States started working on speech recognition engines to expand it to recognize

9 consonant & 4 vowels. Then in 1970s a speech recognition engine was introduced named "Harpy", which was able to understand 1000 words approximately.

In 1980s, potential of speech recognition jumped from 1000 words to several thousands of words. A new model was introduced named "Hidden Markov Model" allowed flexibility and prediction of words based on recently used patterns. In 1997, world's first Continuous Speech Recognizer was introduced named "Dragon's System", which could understand 100 words per minute. In 2008, "Google Voice Search App" came into picture. Then Apple's "Siri" came into market of Speech recognition products. After this other companies also launched their speech recognition engine based products in market like Microsoft's "Cortana" and Amazon's "Alexa"

Google Home, Amazon Echo, Apple Siri have been added to this journey with the evolution of new technology and algorithms in a decade and this journey will continue over many decades.

1.7 WORKING OF SPEECH RECOGNITION ENGINE

To explain working of Speech Recognizer we will divide it into some basic blocks:

- Grammar
- Speech Recognition Engine
- Acoustic Model
- Audio Input
- Recognized Text

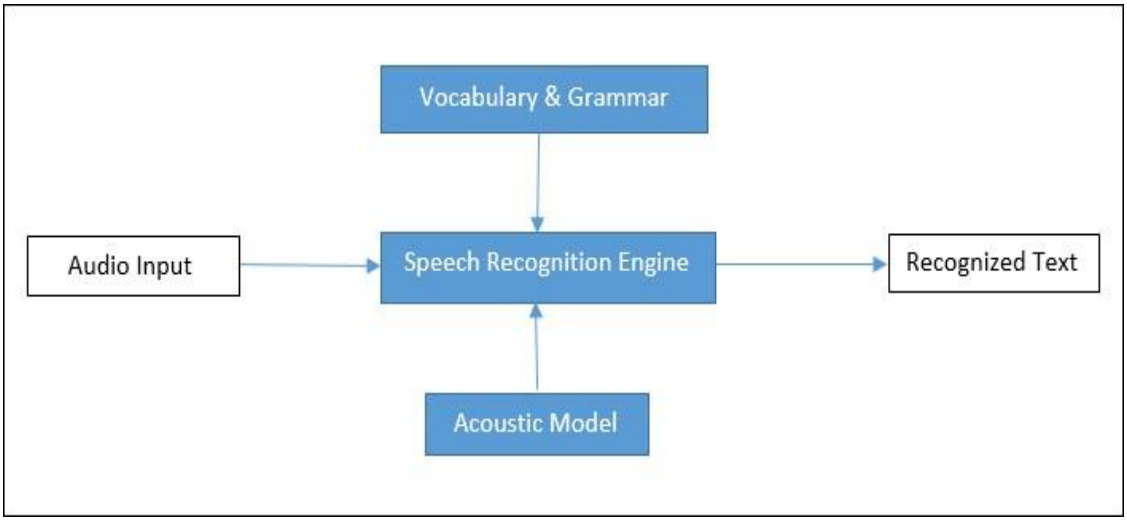

Figure 1.1: Blocks in Speech Recognition Engine

Audio input will contain the speech data/ user utterance along with some background noise. Speech recognition engine should be able to handle the noisy conditions by using various signalprocessing algorithms like noise reduction, beam formation and echo cancellation. After processing the audio input, speech recognition engine will convert the audio data into text format. For conversion of audio data into text format, it will apply data and software algorithms.

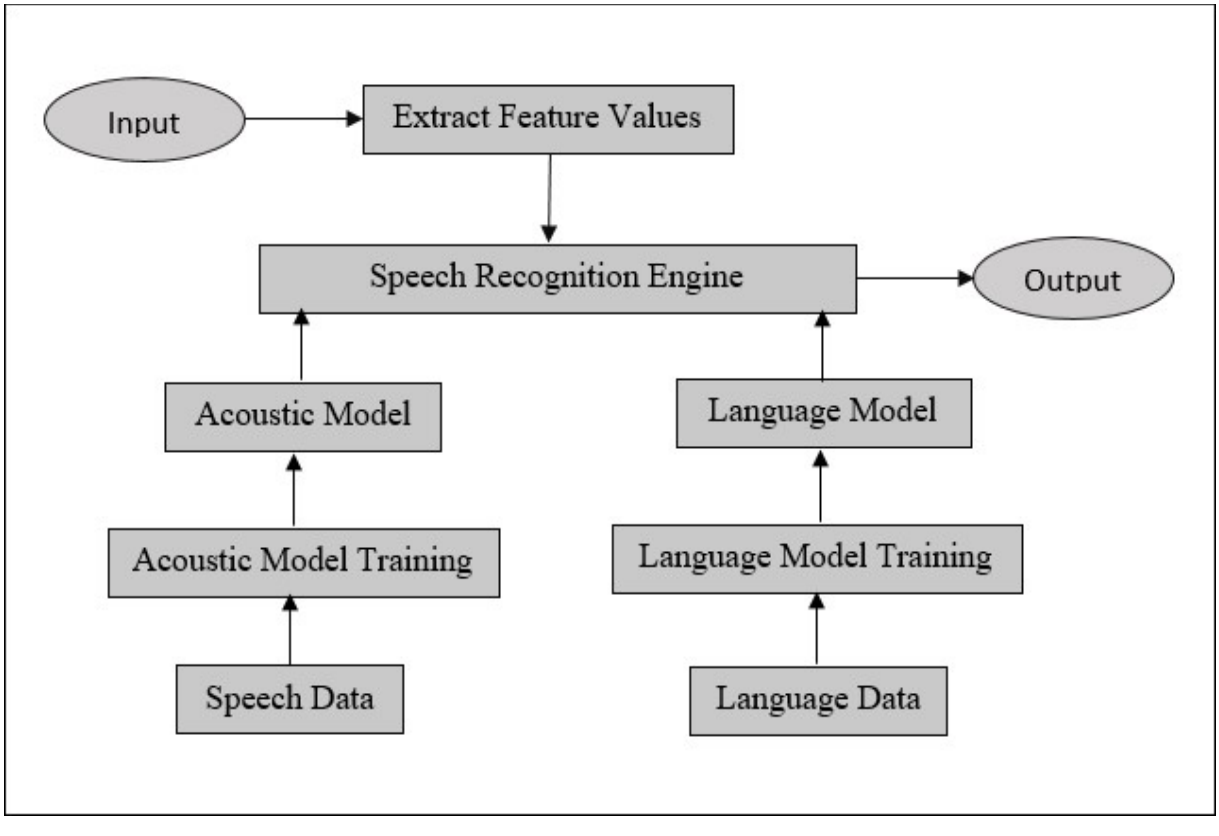

Figure 1.2: Basic Structure of Speech Recognition System

Once the audio converted into text format, recognition engine will search for the best match from vocabulary, grammar along with its knowledge of environment. Here knowledge of environment represented as acoustic model. Engine will try to give best match. Generally, speech recognition engines are based on **Hidden Markov Modal (HMM).** HMM are simple to use and can be trained automatically. Some speech recognition systems uses **Dynamic Time Warping (DTW**) algorithm for speech recognition. Neural Networks has also been used in **Automatic Speech Recognition (ASR)** from last few decades.

1.8 APPLICATIONS SPEECH RECOGNITION ENGINE

1. ***Health Care/Medical:*** Speech recognition engine can be used j medical documentation.

People with hearing disability can use speech recognition to convert a telephone caller's speech into text format.

2. ***Command & Control:*** Automatic speech recognition (ASR) systems are used to control the device by voice commands.

3. ***Dictation:*** Dictation is very common application of speech recognition. It can be useful in different area like medical documentation, business, legal transcript or word processing with help of speech to text conversion.

4. ***Military and Aircraft*****:** US, France, Canada many other many countries have been use speech recognition systems in their aircrafts. Some other examples are Puma helicopter from France, Eurofighter Typhoon**.**

5. ***Telephony:*** To send voice mail by allowing a sender to speak the command and messages rather than typing and pressing the buttons. Will be useful to develop user interface to drive a smart phone.

6. ***Virtual Assistant:*** Apple's Siri, Microsoft' Cortana, Amazon's Alexa, Google Home are the few examples of speech recognitions in virtual assistant for PC, smart phones, smart speakers.

7. ***Home Automation:*** Smart home will control the lights, alarms and other electronic appliance by giving voice command.

8. ***Telecommunication:*** Can be used for automation of operator services like voice recognition call processing, automated alternate billing systems developed by AT&T.

1.9 SPEECH RECOGNITIONON ENGINE

Speech recognition process include following algorithms:

1. Hidden Markov Model (HMM)

2. Dynamic Time Warping (DTW)

3. Artificial Neural Networks (ANN)

1. Hidden Markov Model: Hidden Markov Model is a based on statistical Markov Model in which system being considered is assumed to a Markov process with hidden (unobserved) states. States in HMM are connected by transition. and starts from initial state. After discrete time interval state transition will take place and new state will generate one output symbol. State transition and generated output symbols are random and both will be governed by probability distribution theory. HMM can be considered as a black box, where the sequence of output signal generated over discrete time intervals is observable but sequence of state transition over discrete time intervals can not directly observable and is completely hidden. This is the reason it is called Hidden Markov Model. HMM is popular model to implement speech recognition system.

Speech recognition algorithm contains two major parts: Acoustic Front End and Search Algorithm. Acoustic front end converts the raw speech data to observation vectors which represents the existing events in probability space. Search algorithm will search and fine the most probable event sequence among these events.

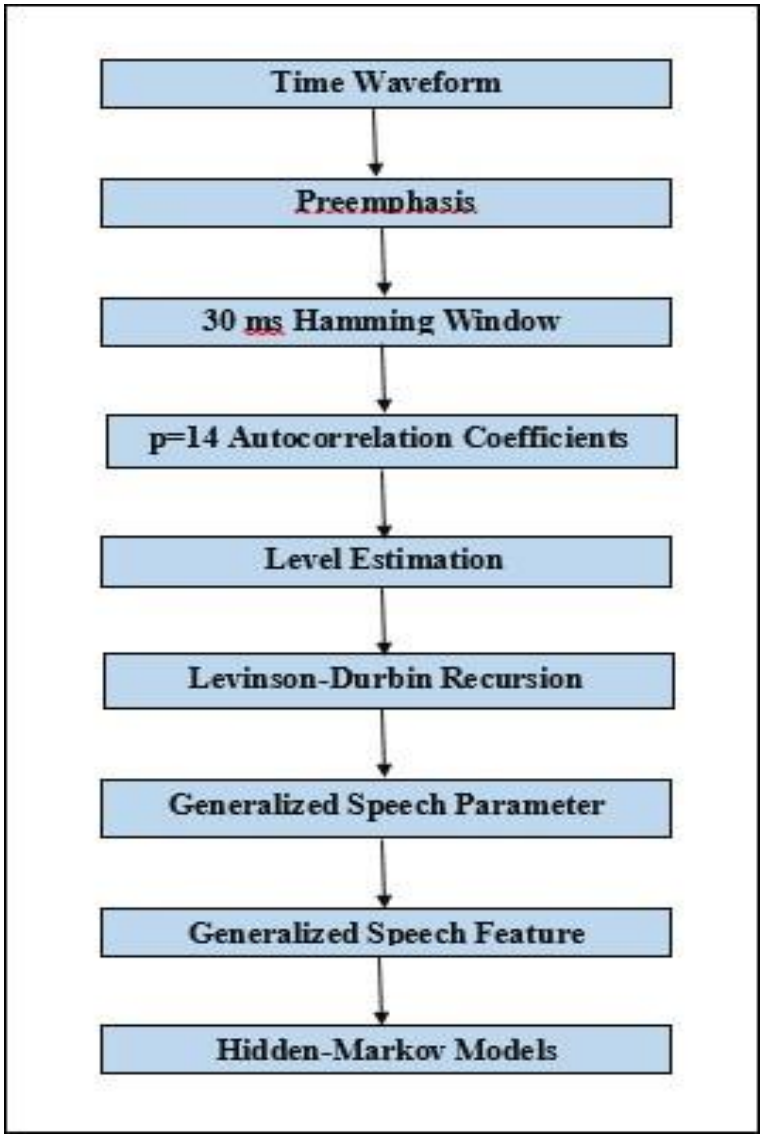

Figure 1.3: Block Diagram of Speech-Recognition System

At very first stage of speech recognition acoustic front-end algorithm, raw speech is converted into digital signal with the help of Analog to Digital Converter (ADC) by sampling. Then on the digital signal, spectral shaping is performed with the help of Finite Impulse Response (FIR) filter to emphasize the more important frequency components in the speech digital signal. This filter is called audio pre-emphasis filter. Sampled signal is represented in form of frames of finite time slice. Then windowing technique is used to allow the portion of sample which is near to the center of window to become more heavily weighted than the portion of sample which is far away from the center of window. Windowing is used to minimize spectral leakage. Then Linear Predictive Coding (LPC) algorithm is used for spectral analysis.

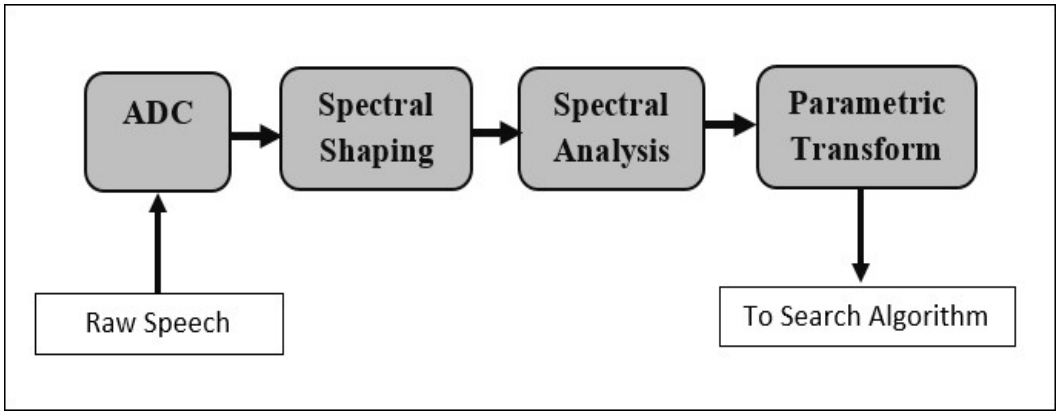

Figure 1.4: An overview of Acoustic Front-End Signal Processor

HMM uses 3 algorithms: Forward Algorithm for isolated word recognition, Forward-Backward algorithm for training a Hidden Markov Model, Viterbi algorithm for continuous speech recognition. Hidden Markov Model has some limitations, which includes requirement amount of large training data, constant observations of frames; output probability density function is restricted.

2. Dynamic Time Warping: This algorithm measures similarity between two time & space varying sequences. This technique compares the words with some reference word in vocabulary. Dynamic time warping is an old approach for speech recognition, which has been replaced by HMM. It is best method to recognize single word and then match it against the all words in vocabulary to find the best match. Rate of speech may vary throughout the words, so optimal alignment will be non-linear in speech samples. In this situation DTW is the best and efficient method to determine the optimal nonlinear alignment in speech. DTW can be applied to audio, video and graphics or can analyze any data that can be represented linearly. DTW is less complex and has fast search with poor speech recognition rate. It is good to use this algorithm for the systems being used is very less noisy environment.

3. Artificial Neural Network: Artificial neural Networks (ANN) are the networks based on neural structure of human brain. Human brain is collection of interconnected nerve cells, which are responsible for basic information processing in human brain, also called neurons. Human brain is made up of approximately 10 billion neurons and trillions of connections between them. If multiple neurons work together then human brain can perform the calculation and processing much faster than the fastest existing computer today. Neural networks do not pre assume any feature

statistical property for speech recognitions**.** Artificial Neural Networks consist of hundreds of processing units. Each processing unit act as a real human brain neuron, which is capable of sending & receiving the signal among different units to form a complex communication network. ANN is non-linear model that is why it is easily understandable as compared to statistical model. ANN are being widely used in image compression, speech recognition, stock market prediction. ANN has following features and capabilities:

(a) ANN supports adaptive learning i.e. they are able to learn how to perform a task based on provided data for training.

(b) ANN also supports real time operation i.e. they have ability to do parallel computation and processing for real time.

(c) ANN is self-organized and can organize and manage the data received during training process.

(d) ANN has fault tolerance capability and they use Redundant Information coding for fault tolerance.

Neural networks can be divided into different types like Recurrent Neural Networks (RNN), Deep Feedforward Neural Networks (DNN). Feed-forward Neural Network is a type of artificial neural networks in which connections between units do not form a direct cycle. In this information or speech data flow only n forward direction from input node to output node via intermediate hidden nodes.

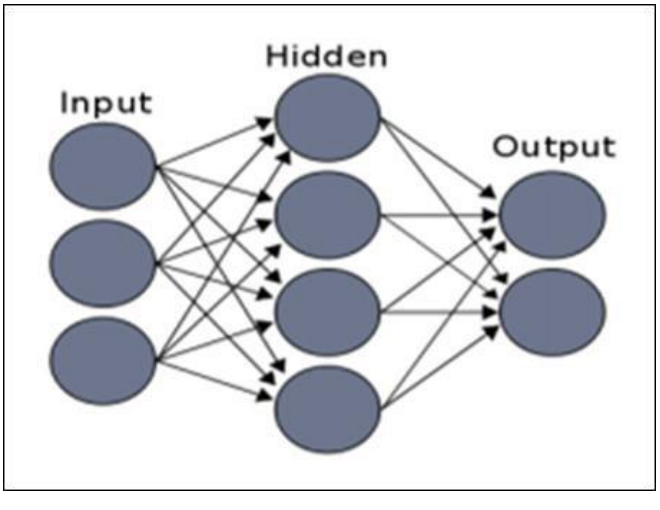

Figure 1.5: Feed Forward Neural Network with one Hidden Layer and one Output Layer

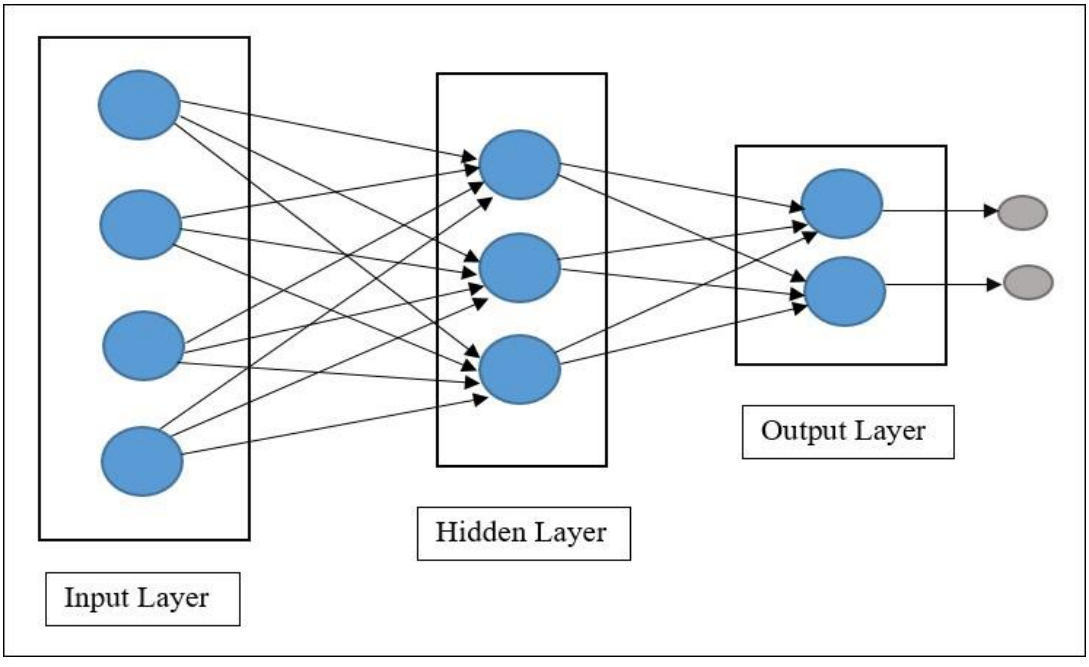

Figure 1.6: Recurrent Neural Network

Recurrent Neural Networks is another type of artificial neural network in which connections between units form direct cycle. RNN has good speech recognition rate but complex training algorithm.

1.10. SPEECH PROCESSING ALGORITHM

1.10.1 Noise Reduction

Noise is one of the most important factor needs to be focused to enhance the performance of any Automatic Speech Recognition (ASR) system due to its great influence in speech recognition process. Noise generated due to environment or channel can greatly degrade the performance of any ASR system. It may result in corruption of speech signal and information contained in it. Motive of noise reduction is to eliminate noise (unwanted signal) from speech signal to increase Signal-to-Noise ratio (SNR) and preserving the shape, energy level and other characteristics of original speech signal. Noise can be classified into different categories like random noise, additive noise, background noise, exhibition hall noise, non- stationary noise, white Gaussian noise, natural

noise etc. Mainly it can be classified into four main categories: Additive, Echo, reverberation and Interference. Noise reduction will depend upon the domain time, frequency and time-frequency.

There are four types of noise reduction methods for speech processing: Subspace, Wiener type, Spectral subtractive and Statistical model based. These method will use some popular algorithms like **Log Minimum Mean Square Error** (LogMMSE), **Least Mean Square** (LMS), **Error Non-linearity LMS** (EN-LMS), **Normalized LMS** (N-LMS), **Wiener Filter** (WF), **Adaptive Noise Cancellation** (ANC).

Additive White Gaussian Noise is most reasonable noise in all communication channels. Amount of improvement in Signal to noise ratio (SNR) is measure of the efficiency of any noise reduction technique. In noise reduction technique underlying methodology is eliminate the inline noise in silence interval from speech signal. Adaptive filters working on speech spectrum are most widely used in noise reduction techniques. Let us discuss ideal noise canceller from which noise reduction are derived.

1. Adaptive Wiener Filtering: This noise reduction technique is two input technique. As shown in figure AWF has one separate input for noise and can provide complete noise cancellation through adaptive Wiener filtering. Filter design and performance can vary depending upon the speed and complexity of adaptive filtering algorithm used. Therefore, performance of filtering will depend on the speed of adaption and design complexity of AWF.

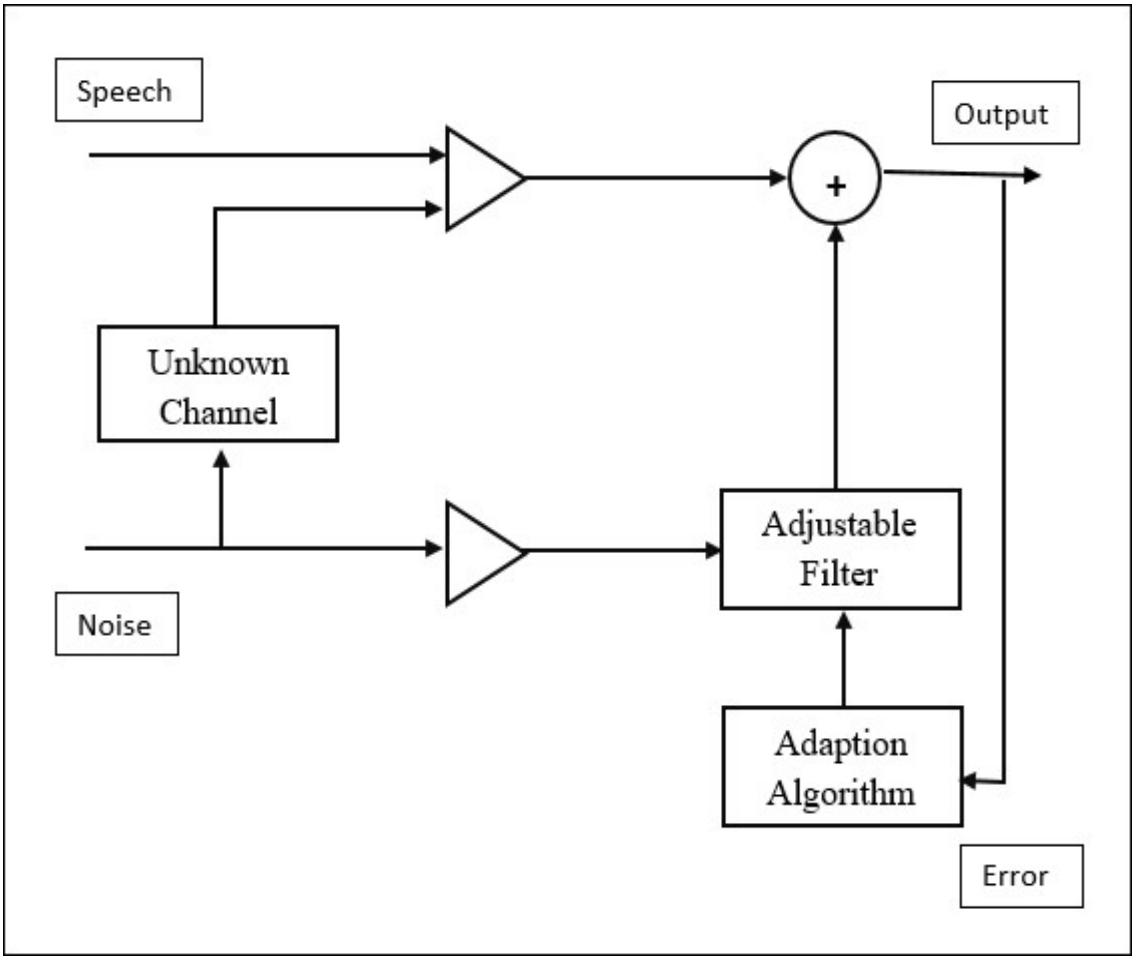

Figure 1.7: Adaptive Wiener Filter Noise Cancellation

2. Adaptive Line Enhancer: Adaptive Line Enhancer (ALE) is modified model of Standard Line Enhancer that is capable of abstracting the narrow band noise from a broadband signal. Their designs consist of a pitch detector to generate reference signal for voiced sound. It will extract the noise signal as an error signal from speech signal, by applying delay of single pitch period. Processed speech output signal will be output of adjustable filter of ALE.

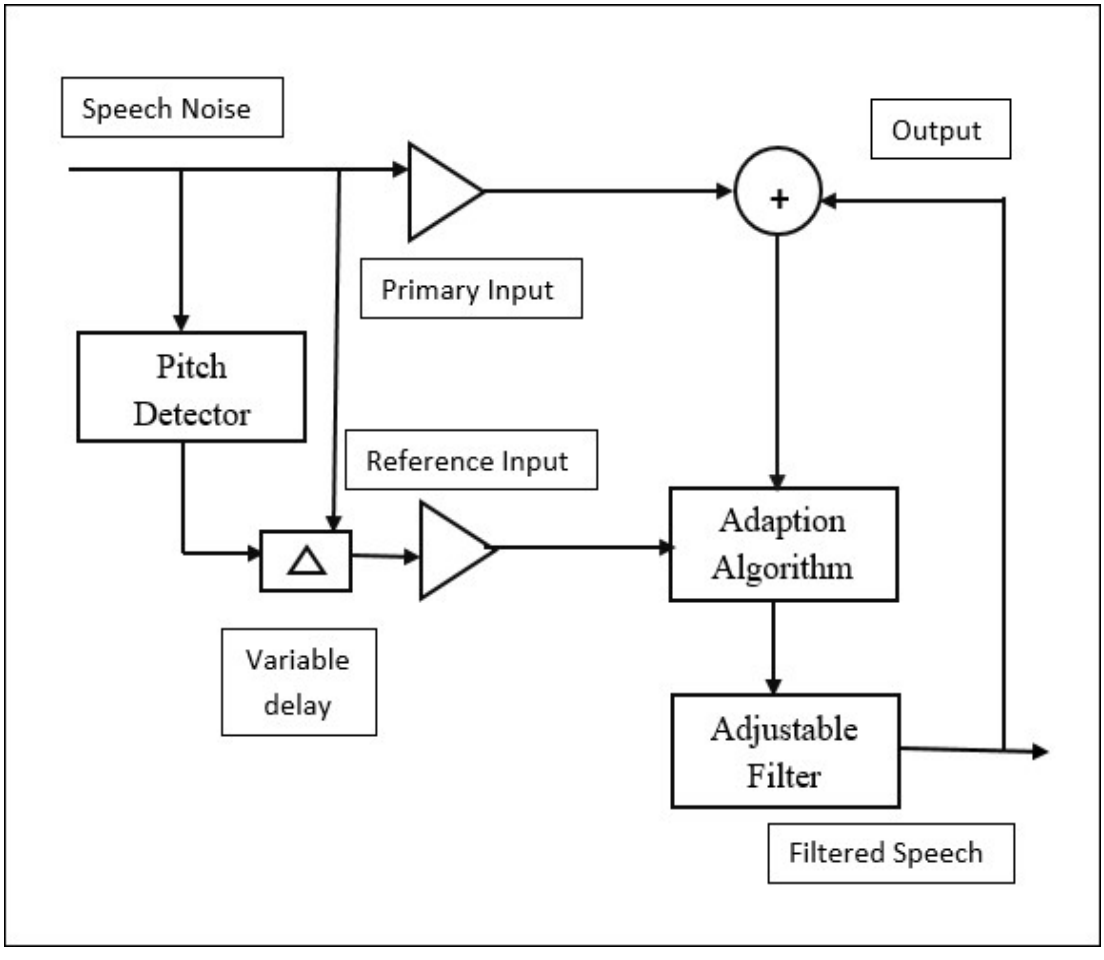

Figure 1.8: Adaptive Line Enhancer

3. Spectral Subtraction: This is the simple and easily implementable method for noise reduction. In speech signal, it estimates the spectral magnitude of the gaps or no speech period and then subtracts this spectral magnitude estimate from the original speech signal. There are some limitations of this SS method. One of the limitation is that resultant magnitude cannot be less than zero. Second limitation is does not consider the statistical distribution of speech signal. This technique generates short duration narrow bands, which sounds like "musical" tones. These narrow bands required to be eliminated from speech signal. This technique includes Discrete Fourier Transform (DFT) and Inverse Discrete Fourier Transform (IDFT) i.e. conversion from time domain to frequency domain and vice versa, which makes it computationally expensive compared to other noise reduction techniques.

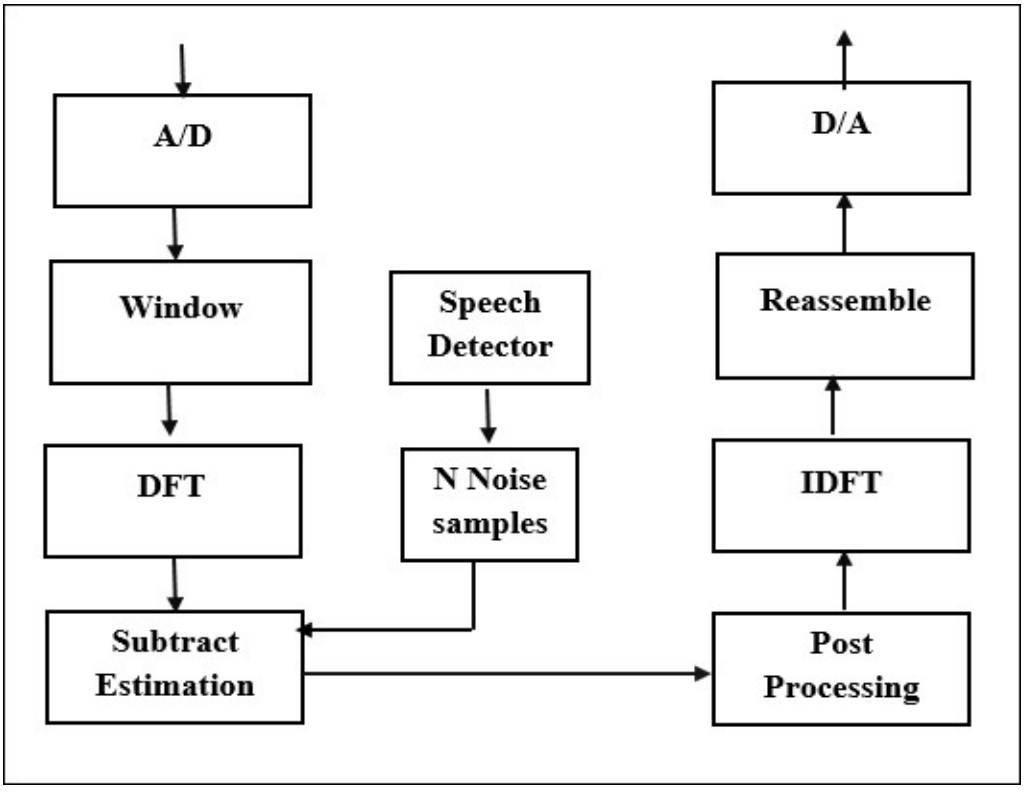

Figure 1.9: Spectral Subtraction

1.10.2 Beam forming

Beam forming is an array processing technique. This technique uses multiple microphones to get high Signal to Noise (SNR) by using directional signal reception. In beam forming technique speech can be received from multiple desired direction. This method steers the sensor in array toward target signal according to some algorithms. This desired direction is known as look direction. These array of sensors act like microphone and will provide the beam forming of speech signal. These microphone array form a spatial filter which can receive signal from desired direction and eliminate the contamination of signal from undesired direction. Beam formation technique includes implementation of spatial-temporal filter, which used to process the output of microphone arrays in time and frequency domain. For processing a speech signal in time domain, microphone signal needs to be filtered through a Finite Impulse Response (FIR) filter then these filtered microphones output signals are combined to get complete beam former output.

Beam former can be of two types:

(A) Data Independent Beam Former: Data independent beam former are also known as fixed beam former. In this algorithm, parameters will remain fix throughout the process. In fixed beamforming both noise source and signal source has some fixed location with respect to microphone array. Fixed Beam Former category includes Delay-and-Sum, Filter-and-Sum, and Weighted-Sum.

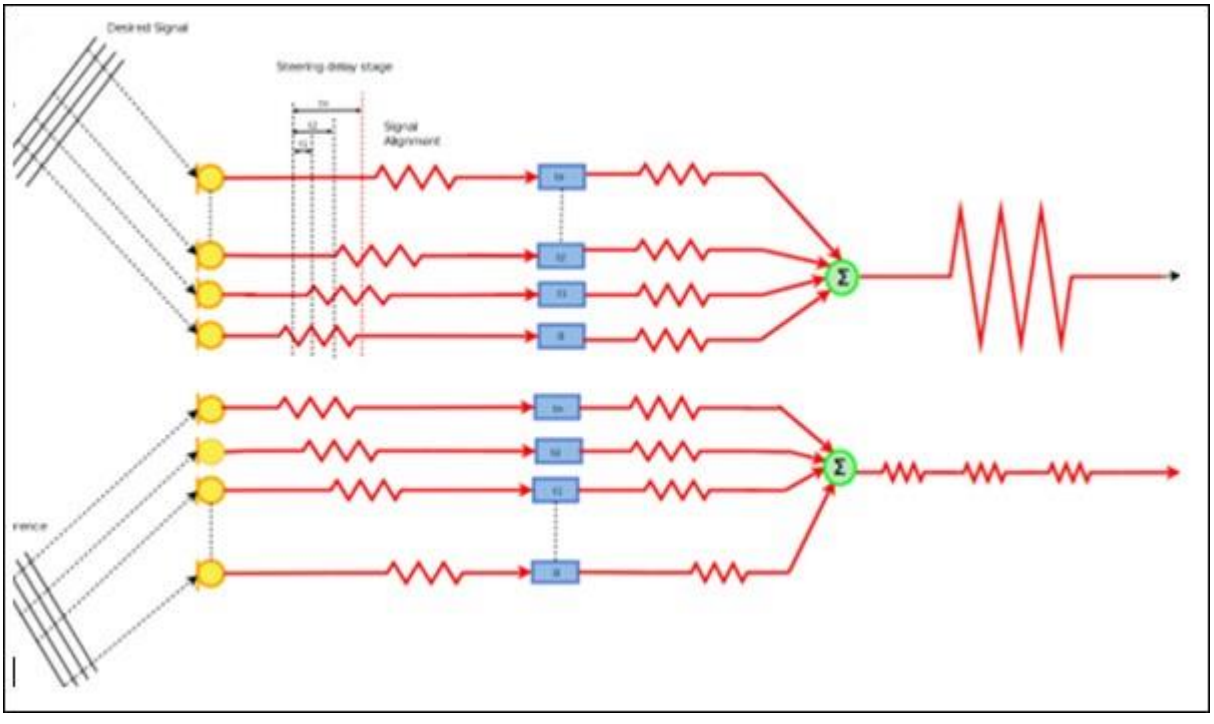

Figure 1.10: Delay and Sum (Fixed) Beamforming

Delay and Sum beam former is simplest type of beam former. It is designed assuming that if microphone array is linear equidistance then output generated by them will be same but delayed by different amount. Sum of delayed output from microphone array will generate desired signal with lower noise and interference. The Signal-to-Noise ratio (SNR) of combined desired output signal will be higher than the SNR of individual microphone in array. Delay in each microphone is calculated based on inter-element distance in microphone array. So geometric arrangement of microphones and their weight plays vital role in generating desired output signal. This method requires large number of microphones to improve the SNR.

(B) Data Dependent Beam Former: Data dependent beam former are also known as adaptive beam former. In this parameters will vary according to signal input. In adaptive beamforming noise source and signal source need not be fixed, the can move. These types of beam formers are useful

when beam former are desired to be adaptable and steer in the direction of desired signal to avoid and eliminate noise from undesired directions. Adaptive Beam Former category includes Generalized Side Lobe Canceller (GSLC), In Situ Calibrated Microphone Array (ICMA), and Linearly Constrained Minimum Variance (LCMV).

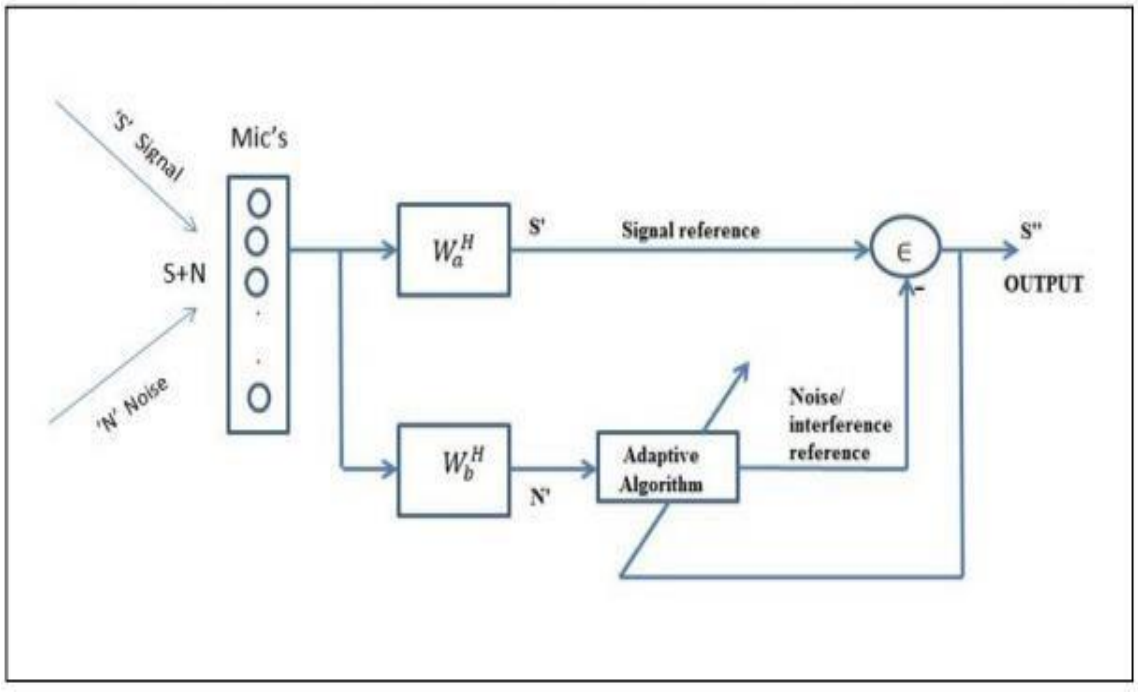

Figure 1.11: Adaptive Beam Former

1.10.3 Echo Cancellation

In context of audio signals, echo is defined as the sound caused by reflection of the original sound from a surface, which reaches to listener with some delay after original sound has stopped. The amount of delay in echo i.e. reflected sound depends on the distance of reflecting surface from source and listener. Strength of echo is measured in dB Sound Pressure Level (dB SPL). Echo canceller is desired to be capable of handling echo or double talk in real time operation for voice-controlled devices. In voice controlled systems echo canceller is responsible to enhance quality (signal-to-echo ratio) of the user speech by suppressing the echo. Adaptive filters cancels echo by identifying the echo path. Quality of Acoustic Echo Canceller (AEC) depends on speed of convergence and accuracy of adaptive filter.

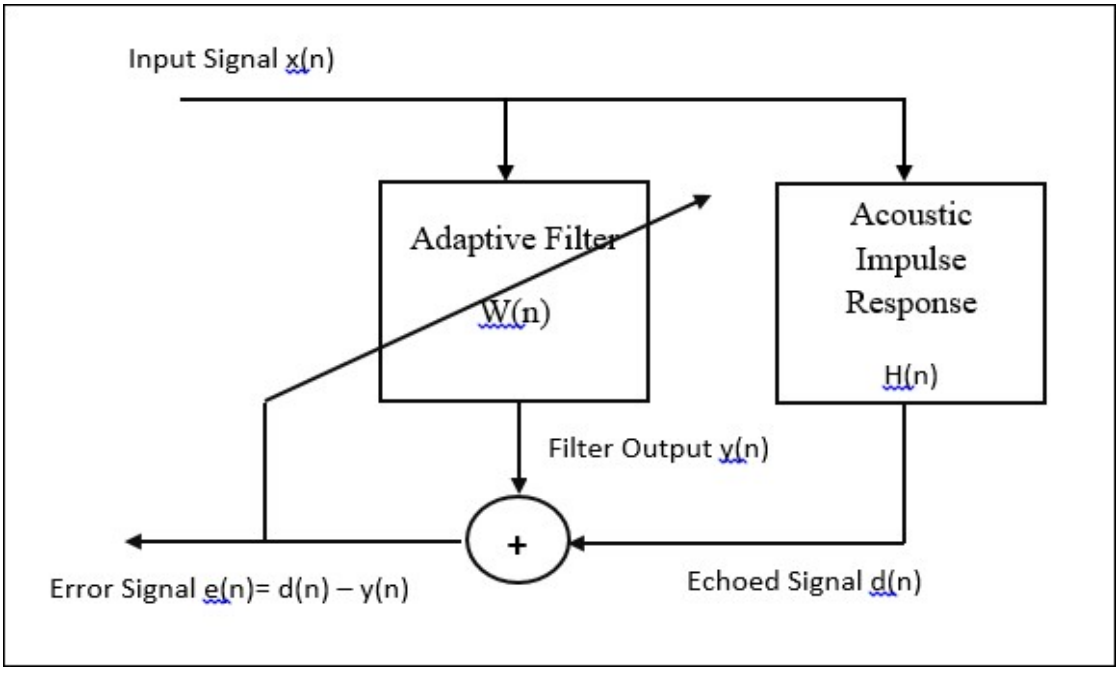

Figure 1.12: Adaptive Echo Cancellation System

Acoustic echo signal is generated when an audio or speech signal is reverberated in real time environment. Echo canceller need to be adaptive as the characteristics and parameters of near end microphone and speaker are unknown. Adaptive filters are linear filters of which transfer function is controlled by variable parameters so these can alter their parameters to get an optimal desired output using some optimization algorithm. Closed loop adaptive filters modify its transfer function by using feedback in from of error signal. The function of difference between the desired output d(n) and actual output y(n) of adaptive filter is known as cost function. In echo cancellation, the adaptive filter aims to minimize the cost function by applying adaptive algorithms. In the above figure, H(n) is impulse response of acoustic environment and W(n) represents the response of adaptive filter used for echo cancellation. Aim of adaptive filter is to make its output y(n) equals to desired output d(n). So error signal e(n) is equal to the difference between y (n) and d(n) i.e.

$$e(n) = d(n) – y(n)$$

Error signal e(n) is feedback to the adaptive filter then adaptive filter will adjust its transfer function according to error signal to minimize the difference between desired output signal d(n) and actual adaptive filter output signal y(n). When adaptive filter output y(n) becomes equal to desired output d(n) then error signal e(n) = 0. This is the desired situation in which the echo is completely get

cancelled and user will not hear any echo (delayed reflection of his own voice) during interaction process. Adaptive filter make use of Least Mean Square (LMS) algorithm Normalized Least Mean Square (NLMS) algorithm, Variable Step Size Normalized Least Square (VSS-NLMS) algorithm for adaptive filtering.

1.11 PROBLEM DEFINITION

Speech recognition provides an interface to interact with a system by making use of voice commands. Therefore, it eliminates the requirement of Graphical User Interface (GUI), Touch Interface, keyboards, and mouse. Speech input can be given by user from distance, making it the preferred mode for situations in which the hand and eye may be busy. Speech interface has proven a boon for people with severe physical or neuromotor disabilities by providing them an alternative input method. Speech enabled systems are very cost effective and often more efficient in real world applications that need to command and control by reducing human efforts.

Therefore, purpose of this thesis work is to test the proposed & developed Speech Recognition Engine and to identify the flaws and scope of improvement.

CHAPTER 2

LITERATUE SURVEY

This section involves the work done by the various researchers in the field of Speech recognition for Internet of Things.

Mohammad Abdur Razzaque [1] worked on Middleware for Internet of Things. He presented the importance of middleware and future challenges in IoT. He explained Internet of Things (IoT) characteristics, machine-to-machine (M2M) communication, middleware requirements, RF identification (RFID), wireless sensor networks (WSNs) and supervisory control and data acquisition (SCADA) briefly. He discussed about Characteristics of IoT Infrastructure like Resource-constrained, Spontaneous interaction, Ultra-large-scale network and large number of events, Dynamic network and no infrastructure, Context-aware, Intelligence and Characteristics of IoT Applications like Diverse applications, Real time, Increased security attack-surface, Privacy leakage. He did survey of state of the art technological requirement for middleware.

Ms. Yogita Pundir et al. [2] worked on Challenges and Future Directions of IoT. They briefly discussed technology, functionality and challenges for present IoT based solutions. They explained RFID, WSN, and IPv6 in context of IoT. They discussed some of the future challenges in order to build the IoT like Standards and interoperability, security, trust and privacy, complexity, system architectures and networking protocol. They also discussed about some basic applications that can be offered by IoT in the area of Smart Homes, Smart Environment, Smart Enterprises and Smart Wearable, and some future impact of IoT.

Anupma Kaushik [3] has given an overview of Internet of Things. She described IoT Communication model, Device to Gateway model, Device to Cloud Communication and Backend Data-sharing model. She explained how a vast amount of research is to be done if we really want IoT to be a reality. In this context, she mentioned terms like Massive Scaling, Big Data, Architecture & Dependencies, Robustness, Security and Privacy.

Govinda K. [4] briefly explained basic concepts, IoT enabling technologies like RFID, Wireless Sensor Networks and Artificial Intelligence. He reviewed potential applications of IoT enabling technologies and the major research issues related to IoT enabling technologies. He described Wireless Sensor Network and Layered architecture of Internet of Things. It is believed that in the near future the achievement of the vision of "from anytime, anyplace connectivity for anyone, connectivity for anything" should depend on cross-discipline and cooperative efforts in related fields.

Sean Dieter Tebje Kelly et al. [5] worked on implementation of Environmental Condition Monitoring in Smart Homes in context of IoT. They proposed an effective IoT implementation to monitor regular domestic conditions by with efficient and low cost sensing system. For the reliable measurement of parameters by smart sensors, they also described the integrated network architecture and inter-connecting mechanisms and transmission of data via internet**.** The developed system has greater control over routing of packets (security and customization) and along with ability to adapt to other WSN.

Shanzhi Chen [6] worked on applications, future opportunities and challenges in IoT with China Perspective. They highlighted some challenges regarding technologies, standardization and applications. To meet the architecture challenge, they have proposed an open and general IoT architecture that includes three platforms. He has also mentioned the challenges like low power nodes and computing, identification and positioning technologies, low cost and low latency communication, self-organized distributed systems technology, and distributed intelligence.

John A. Stankovic [7] worked on research directions for IoT. He briefly highlighted a vision for a smart world. In this research paper author highlighted a various significant research requirements for future IoT systems along with the work done across different research communities. In his research, he highlighted eight areas: creating knowledge and big data, system robustness & openness, security & privacy, architecture & dependencies, massive scaling and human-in-the-loop. His discussion focuses on future challenges that arise for IoT solutions.

Ala Al-Fuqaha [8] has given an overview of IoT with emphasis on IoT enabling technologies, protocols, challenges and application. He has provided a horizontal overview of IoT solutions. He has also briefly discussed the key challenges for future IoT sysytems in context of work presented in the recent literature and research. To illustrate how the different protocols presented in the paper fit together to deliver desired IoT services, he has provided the detailed service usecases. He introduced the basic IoT elements and implementation technologies & tools that are required for realization an IoT system. He discussed different protocols like MQTT & CoAP and concluded that MQTT is suitable for resource-constrained devices that use unreliable or low bandwidth links while Constrained Application Protocol CoAP is reliable, secure protocol for IoT.

Tiago Duarte et al. [9] have provided an overview of speech recognition for voice-based machine translation. They discussed the various technologies and work done in area of machine translation

(MT). In this document they covered available technologies for speech recognition like Microsoft Speech API, Microsoft Server related technologies, Microsoft Unified Communication API, Sphinx open source framework for speech recognition, HTK speech recognition framework, Julius decoder system, Java speech API, Google Web speech API, Nuance Dragon SDK for desktop and mobile application in speech recognition. They concluded that there is need for speech recognition systems that would be capable of supporting multiple languages and speech-to-speech translation to overcome the language barrier worldwide.

Inge Gavat et al. [10] have discussed about the role of Deep Learning in Automatic Speech Recognition and Understanding (ASRU). They briefly explained and evaluated some the deep learning algorithms like Restricted Boltzmann Machine (RBM), Deep Belief Network (DBN), Auto Encoder (AE) and Convolution Neural Network (CNN). They also highlighted deep learning structures for Phone Recognition and Large Vocabulary Continuous Speech Recognition (LVCSR). They concluded that states of art techniques are outperforming and almost successful, but still there is need to solve the problems regarding training to achieve a level of perfectness.

Oliver Hahm et al. [11] discussed the different Operating Systems (OS) for low-end IoT devices in detail. Low-end IoT devices have resources constraint like limited computational capability, limited memory and limited power supply. Therefore, these low-end IoT devices do not use traditional operating systems like Windows and Linux. In this paper, they have discussed about the requirements for operating system (OS) in context of low-end IoT device. These requirements include small memory footprint, support for heterogeneous hardware, low power consumption for network connectivity, real time capability, high security & privacy and energy efficiency. They also provided an overview of key OS design alternatives or choices like general architecture & modularity (e.g.: microkernel and monolithic approach), scheduling model (e.g.: preemptive and non-preemptive or cooperative scheduling mechanism), memory allocation, network buffer management, programming models (event-driven or multithreaded), programming language and debugging tools. They briefly reviewed various open source operating systems (OS) as RIOT, FreeRTOS, TinyOS, OpenWSN, uCLinux, mbedOS and closed source operating systems (OS) like ThreadX, QNX, VxWorks, LiteOS Huawei for IoT. This paper also includes the case study for Contiki, RIOT and FreeRTOS. They concluded that FreeRTOS is the most reliable, efficient and prominent Real Time Operating System (RTOS) for IoT.

Soudeh A et al. [12] discussed the Microphone Array Processing Strategies for distance based Automatic Speech Recognition (ASR). In this paper, they explained how to leverage method like single-level combination using beam forming in Automatic Speech Recognition (ASR) and word hypothesis-level combination using several speech recognition in Recognizer Output Voting Error Reduction (ROVER). They have explained that how negative effect of Speaker Adaption (SA) reduced in ROVER and hence improved the front-end enhancement techniques for Distant Speech Recognition (DSR). He also briefly discussed about different microphone array configurations for DSR that includes Beamforming (BF), Recognizer Output Voting Error Reduction (ROVER) and Multiple Channel Training (MCT). They concluded that MCT provides moderate Word Error Rate (WER) and is simple, time-consuming structure on other hand ROVER provides better WER but includes computational complexity. BF has less computational complexity than WCT and ROVER but in terms of WER, it is worst compared to these two. Therefore, they concluded that combination of these three methods WCT, BF and ROVER would provide best WER with less computational complexity.

Hyunji Chung et al. [13] have discussed about the reliability of Intelligent Virtual assistant (IVA) like Apple's Siri, Amazons Alexa and Google Home. They mentioned the privacy and security risks in IVA like malicious voice commands, wiretapping IVA ecosystem, unintentional voice recording and compromised IVA enabled devices. They have also highlighted some incident regarding vulnerabilities in Intelligent Virtual Assistants. They concluded that there is a need to get a solution that can improve the trustworthiness on IVA systems.

Chan Zhen Yue et al. [14] proposed the low cost, microprocessor based design and implementation of voice activated smart home. The system design they have proposed will leverage the Amazon Alexa Voice Services (AVS) and smart phone and Amazon Developer Console to provide a voice control over smart home. The proposed design consists of Alexa Skills Kit, Raspberry Pi 3 Model B, sensors and smart phone applications as main components. They tested the system setup designed and results were same as expected. They concluded that proposed low cost design is capable of leveraging AVS Alexa skills for smart home and smart phone systems have a wide scope and potential of improvement in future.

Eugenio Rubio et al. [15] explained the Human-Device interaction in natural language through voice commands in Internet of Things (IoT). Purpose of their research work is to decrease the complexity of Natural Language Processing for voice recognition systems in IoT. In this paper,

they proposed a solution that allows the IoT devices to offload the Natural Language Processing to a system in order to improve Natural Language Processing usage and to reduce the requirement of remembering certain words or phrases to activate the device. They have tested and examined the feasibility and reliability of proposed design implementation in home environment. They also discussed the Natural Language Processing briefly. They validated the proof-of-concept implementation and found 92% of the processing of query/order/command responded correctly without any prior training to the proposed system. Therefore, they concluded that the success rate of understanding the command by system is 92%. The future includes the Context-Awareness (CA) recognition and to develop ability to learn ambiguous command or queries in order to improve the performance and reliability of proposed system.

Anoja Rajalakshmi et al. [16] discussed the Internet of Things using Alexa Voice Services. They proposed a system that can connect & control many Internet of Things devices by voice interface. They mentioned the Node-Red tool developed by IBM that can connect the hardware device, Application Program Interfaces (API) and web, cloud services easily and with fewer efforts in context of IoT. They have proposed a low cost, minimal power consumption system for IoT. Proposed system hardware includes Raspberry Pi3 with Intel Edison board, ESP8266 and ESP32 along with AWS components like AWS IoT, AWS Lambda, and Amazon Alexa Voice Services (AVS) and uses MQTT protocol for communication. They have considered different scenarios such as MQTT communication with hardware, weather forecasting, iPhone monitoring and AWS & Alexa. They concluded that proposed system provides a solution to easily connect and control the IoT devices. By the time Node-Red runs on Linux and Raspberry Pi and Linux machine, but still there is scope of running the Node-Red on cloud by making use of IBM Bluemix platform in future for extension of proposed system.

Venton Kepuska et al. [17] discussed about Virtual Personal Assistants (VPA) from Amazon Alexa, Apple Siri, Google Home and Microsoft Cortana. In this paper, they have proposed a Virtual Personal Assistant (VPA) system that supports multi modal dialog by taking image, gesture, speech, video and body movements as user input. Proposed system methodology can be used in home automation, robotics, vehicles, medical, education and security access control systems. Proposed system includes input model, output model, ASR model, online & offline knowledge base, gesture model, graph model and interaction model. They performed different stages of testing for the systems such as testing and comparing of Automatic Speech Recognition (ASR) model with Microsoft API, Amazon API, Google API and testing of Live Speech

Translation, Wakeup-word system, and dialog system tested with Google Cloud & Amazon Web Services and Graph model tested with machine learning techniques like Neural Networks, Deep Learning. After integration testing of complete proposed system they found that, the proposed model is best solution for next-generation Virtual Personal Assistant (VPA). They concluded the proposed system would enhance the user-machine interaction by leveraging new techniques like gesture recognition, speech recognition and image & video recognition for various applications in different fields.

K.F.C Yiu et al. [18] proposed a solution for echo cancellation in Voice Control Devices. The proposed system hardware developed on Xilinx XUP V2P platform in FPGA fabric round a PowerPC. They have illustrated the echo path in speech recognition system and echo cancellation algorithms. They proposed a robust adaptive algorithm that is hybrid of Least Square algorithm and Least Absolute Distance algorithm to get fast convergence. The proposed system hardware consists of Virtex 2 Pro FPGA, DDR SDRAM Controller, 2 PowerPC 405 processor and compact flash. The result shows that system is capable to handle compromise between robustness during echo and efficiency in tracing echo path variations. They concluded that the proposed system is adaptable against tracking echo path variation with high speed and double talk.

Bhushan C. Kamble [19] has given an overview of speech recognition using Artificial Neural Networks (ANN). He focused on different methodologies for Artificial Neural Networks (ANN) and comparison among them. He has introduced speech recognition process and Artificial Neural Network (ANN) in context of speech recognition. He has discussed the different types of Artificial Neural Networks (ANN) such as Feed Forward Network, Recurrent Neural Network, Modular Neural Network and Kohonen Self-Organizing Maps. He also highlighted the advantages of ANN, which includes ability to learn, self to organize, pattern recognition, flexible and adaptable to changing environment. He concluded that Recurrent Neural Network has achieved better speech recognition rate than Modular Neural Network. However, Recurrent Neural Network is dynamically sensitive and requires complex algorithms.

Jianguo Ma [20] has summarized evolution and challenges for Internet of Things (IoT). He has discussed some key technologies in IoT such as smart-sensing, smart-storing, smart-exchanging and ultra-low power wireless technology, and demand of IoT in various fields such as safety monitoring, healthcare, smart manufacturing, smart buildings etc. He has highlighted few terms like Radio Frequency Identification (RFID), Wireless Sensor Networks (WSN) and difference

between internet and IoT. He also discussed the challenges and existing demands for internet of Things. He concluded that there is requirement of low power wireless technology, which will be capable to support future features and applications and new challenges in IoT.

CHAPTER 3

SYSTEM MODEL FOR SPEECH RECOGNBITION ENGINE

3.1 SYSTEM REQUIREMENTS

3.1.1 Hardware Requirements

Hardware requirements are as follows:

- Intel Quark S1000 processor
- ESP32 Microcontroller as host processor
- One FTDI cable
- DC power supply
- USB cable

3.1.2 Software Requirements

Software requirements are as follows:

- Ubuntu Operating System
- Free Real Time Operating System (FreeRTOS)
- Amazon Alexa Voice Services Cloud
- Bug Tracking Tool

3.1.3 Connectivity Requirements

Good Internet connectivity is required to connect with cloud services.

3.2 REQUIREMENT ANALYSIS

The requirement analysis contains all the requirements of the system. It consists of the details like the functionality of the system, requirements like the scalability, maintenance that are non-functional requirements. It also consists of the Dataset considered, the assumptions and constraints if any.

3.2.1 Functional Requirements

Prior knowledge of functional requirements helps the developer to meet customer expectations. Functional requirements are as follows:

Product shall:

- Provide ability to user for audio input (i.e. capturing user speech via more than one microphones).
- Provide ability to user for audio output (e.g. speaker, headphones, Bluetooth or line out).
- Provide physical controls for adjusting volume and for manually initiating an interaction with Alexa.
- Provide ability to user for interrupting an Alexa-initiated output (e.g. music playback and voice response from cloud) using voice interface or physical control.
- Support multi-turn interactions of user with Alexa.
- Support silencing alerts, adjusting volume, enabling/disabling microphones and stopping media during unavailability of internet connectivity.
- Support cloud-based wake word verification.
- Automatic activation of microphones without waiting for a touch interaction in multi-turn situations.

3.2.2 Non-Functional Requirements

Prior knowledge of non-functional or design requirements helps the developer to avoid issues during development, prototype of the product. design requirements are as follows:

- Offline playback of Amazon Music content is not permitted.
- AVS Product must not cache more than 20 seconds of Amazon Music content. Content must not ever be cached on non-volatile storage (e.g., only in dynamic RAM, not written to disk or static RAM). After playback, cached audio data must be flushed from memory.

3.2.3 Use Cases:

Following [21] are the different uses cases that includes user interaction profile – far-field, handsfree and tap-to-talk:

	Far-Field	Hand Free	Tap-to-talk	Push-to-talk
User Interaction	User wakes up the device by uttering Wake Word and cloud instruct the device to stop listening when user stops speaking.	User wakes up the device by uttering Wake Word and cloud instruct the device to stop listening when user stops speaking.	User taps a button to open the microphone and cloud instructs the device to stop listening when user stops speaking.	User pushes and holds the button to keep the microphone open until entire speech is captured.
Use Cases	Here voice is primary user interface in different noise environmental noise conditions. Device will listen across the room. Use case: Hall, room.	Here voice is primary user interface in different noise environmental noise conditions. Device will listen within some specified length. Use case: Kitchen, bedroom.	User will be in close proximity of device. Here tapping is required before voice interface comes into play. Use case: medium ambient environment.	User will be in close proximity of device. Here tapping is required before voice interface comes into play. Use case: high ambient environment.
Wake Word	Yes	Yes	No	No

3.3 SYSTEM ARCHITECTURE

3.3.1 Architectural Diagram

ESP32 AS HOST PROCESSOR:

ESP32 [26] is ultra-low power, 40 nm technology chip and integrated solution for Bluetooth and Wi-Fi. ESP32 has following key features and specifications:

- ***Wi-Fi:*** 802.11 b/g/n (2.4 GHz), up to 150 Mbps.

- ***Bluetooth:*** v4.2 BR/EDR and BLE, Bluetooth Pico net & Scatter net.
- ***CPU:*** Xtensa® dual-core 32-bit LX6 Microprocessor, up to 600 DMIPS.
- ***Memory:*** 448 kB ROM, 520 kB SRAM.
- ***Clocks and Timers:*** Internal and external crystal oscillators, 64-bit timers and watchdog.
- ***Peripheral Interface:*** 34 GPIO, ADC, DAC, SPI, I2S, I2C, UART and PWM.

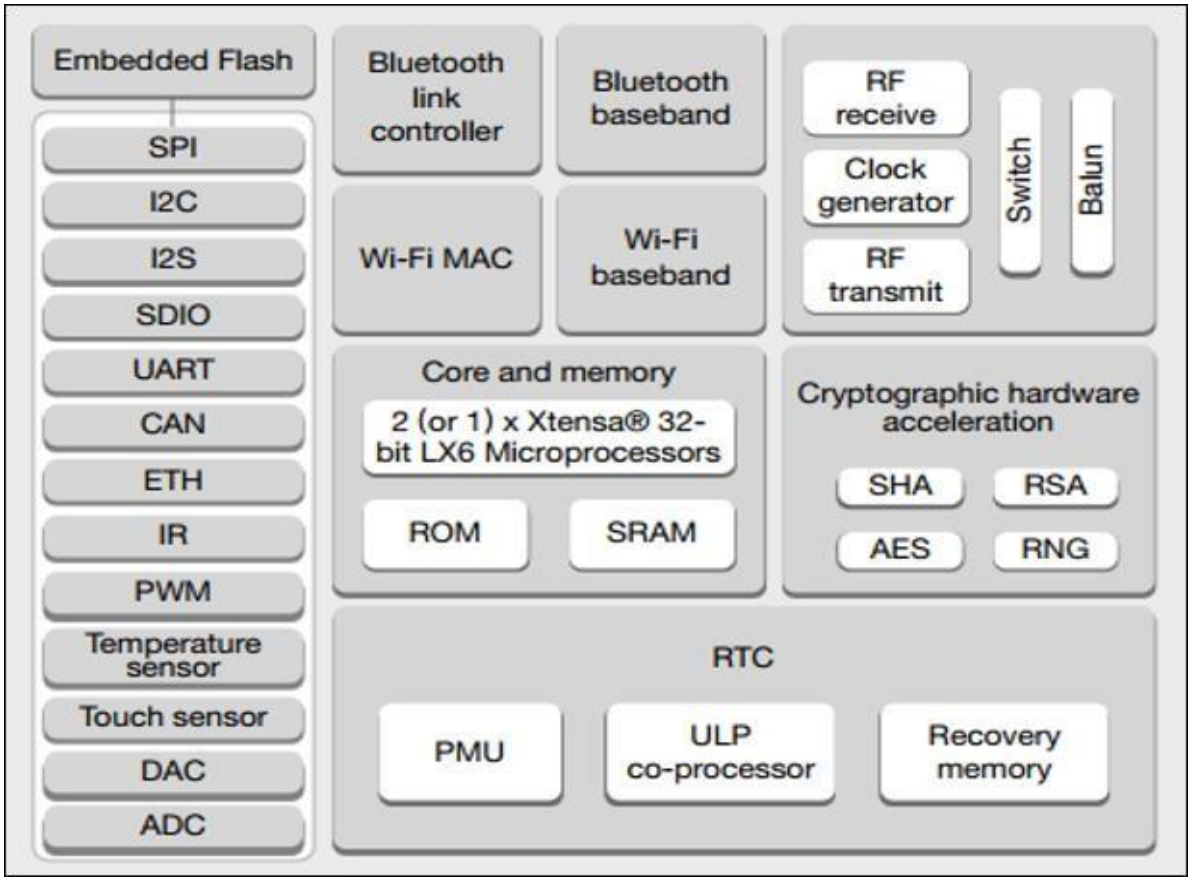

Figure 3.1: ESP32 Block Diagram [26]

INTEL QUARK S1000 AS SPEECH UNDERSTANDING ENGINE:

Intel has designed Speech Enabling Development Kit [22] for developers to create Amazon Alexa integrated products for smart home, which provide a capability to have control over the connected devices by speech recognition. Additional sensors (environmental monitoring) can be integrated in the extensible hardware kit by using flexible interfaces. This kit have inbuilt 8-mic circular array and is powered by Intel's dual Digital Signal Processor (DSP). This kit makes use of Intel's speech

recognition algorithms to detect “Alexa” wake word and to capture the audio for cloud-based Amazon AVS. It consist of :

- 8 Digital Microphones (DMIC) Circular Array
- Connector Cable for Raspberry Pi3
- On-Chip Wake Word Detection Engine
- Speech processing algorithms: Beam Forming, Noise Reduction and Acoustic Echo Cancellation (AEC)

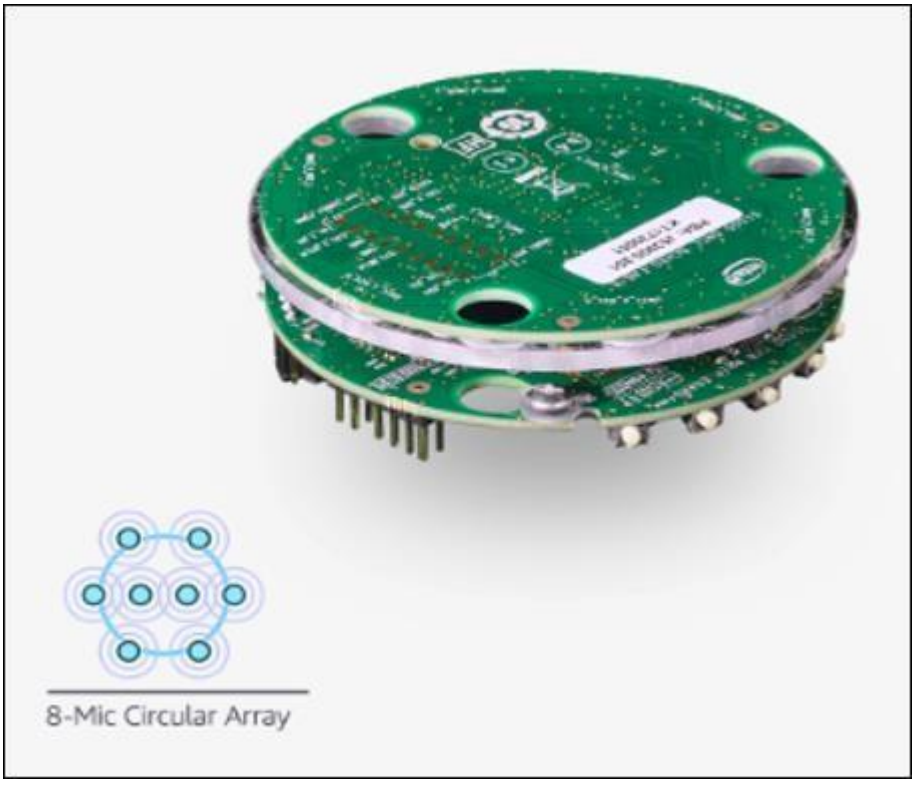

Figure 3.2: Intel® Speech Enabling Development Kit for Amazon Alexa Voice Services [22]

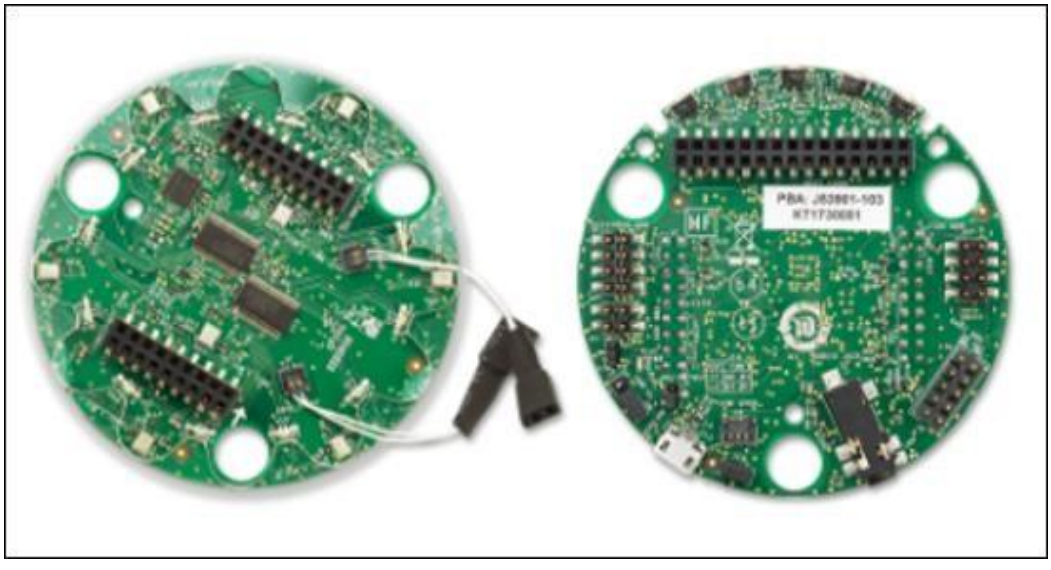

Figure 3.3: Intel® Speech Enabling Development Kit - 8 Digital Mic (DMIC) Board [22]

Intel Quark S1000 [24] has following key features and specifications:

- ***Digital Signal Processors:*** Dual Ten silica LX6 cores @ 400 MHz with HiFi3 DSP.
- ***Speech Accelerators:*** GMM (Gaussian Mixture Model) and Neural Network Accelerator (NNA).
- ***Internal Memory:*** 4MB SRAM.
- ***External Memory Interfaces:*** 8MB 16-bit PSRAM and 128MB SPI flash.
- ***Input/output Interfaces:*** SPI for command & control, I2S for streaming audio and USB 2.0 HS device, Digital Microphone, Speaker, UART, GPIO, PWM outputs.
- ***Power Consumption:*** Low power idle, < 20 mW voice activity detection, < 250 mW full active.
- ***Temperature Range:*** Commercial: 0 to 70 °C and industrial: -40 to +85 °C.

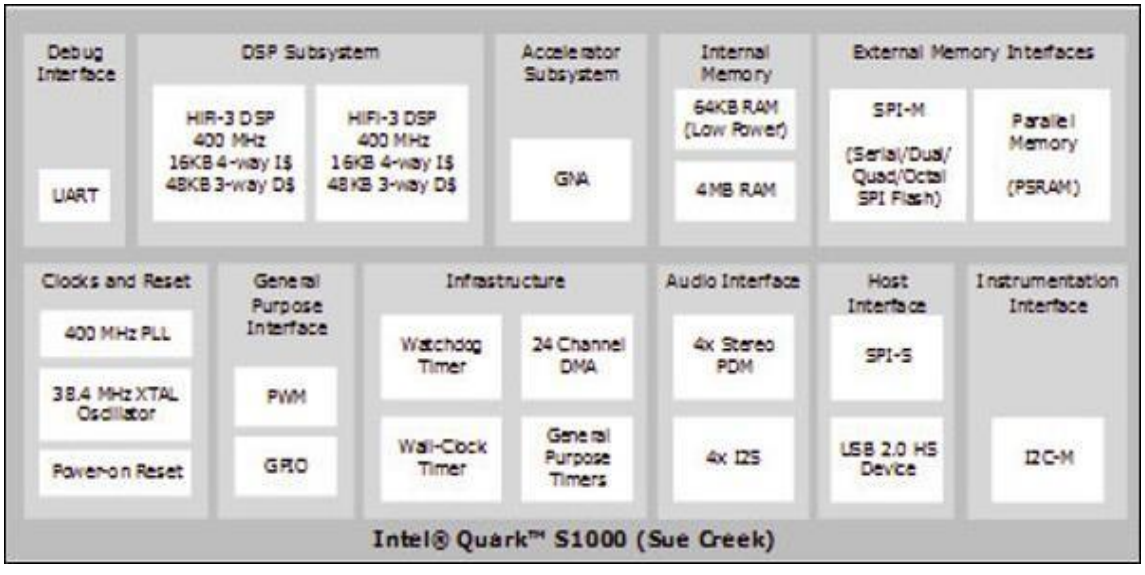

Figure 3.4: Intel® Quark S1000 Block Diagram [24]

3.3.2 AVS SDK Functional Design

Alexa Voice Services (AVS) SDK provides libraries to connect and interact with AVS. SDK provide interface for connecting to AVS, Audio capture, sending events, handling directives and play audio data. Followings are building blocks of AVS SDK [28]:

(A) Alexa Communication Library (ACL):

Serves as communication channel between client device and Alexa Voice Services (AVS). Two functionalities of ACL are:

- Establish and maintain connection with AVS.
- Provides message sending and receiving capabilities.

Main components of ACL are TLS transport, HTTP2 transport [29] and AVS connection manager.

- ***TLS Transport***

  Creation generic functions to connect, write data and read data from a server or host using mbedtls driver APIs.

- ***HTTP2 Transport***

  Creating generic functions to connect send data and read data from a server or host using http2 driver APIs.

- ***AVS Connection Manager***

  Manages the connection with Alexa. Initializes the authorization library, which will get the access token using refresh token. Connects to Alexa server-using http2 connect. Creates down channel and synchronize device states with Alexa. In addition, provide functionality to send message to Alexa and disconnect from Alexa server.

(B) Authorization Library:

This component [30] connects to Amazon server to get the access token using refresh token. Connects to amazon API server. Form http post request using client details (client details like client id, client secret and refresh token are stored in config.json file) and send the request using TLS write. Reads and parse the response to get the access token.

- Amazon API Server or Host : "api.amazon.com"
- Amazon API Server port: "443"

(C) Wi-Fi Library:

Provide functionality to configure, initialize, start and maintain the Wi-Fi connection.
Steps in Wi-Fi initialize:

- Initializes TCP-IP adapter.
- Initializes Wi-Fi with default configurations.
- Set storage and set Wi-Fi mode.
- Set configurations like SSID and password.
- Check certificates and
- Start Wi-Fi.
- Wait for connection to establish

(D) Alexa Directive Sequencer Library (ADSL):

ADSL have four components- Message Interpreter, Directive Sequencer, Directive Processor and Directive Router. This library parses the http2 multipart response and route the directives to corresponding capability agent and audio data to Audio Player.

➢

Message Interpreter

Message interpreter, parse the multipart http2 response using multipart parser library (open source code). It receives http response and parses the response using the boundary line. It separates directives, meta data and audio data. Directives are sent to Directive Sequencer Library, audio data is sent to Audio player via capability agents.

➢ ***Directive Sequencer***

Directive sequencer receives directive from message interpreter checks for "Dialog Request ID" if present then send the directive-to-directive processor else handles it immediately.

➢ ***Directive Processor***

Directive processor maintains a task with infinite loop to process the directives in sequential manner.

Directive processor receives the directive with dialog request id, adds it to queue, and notifies the processor loop. Queue is maintained by common AVS directive library. Processor loop, upon notification:

- Gets the first element of queue.
- Make call to router handle directive.
- Remove directive from queue after processing.
- In addition, checks and processes the directives in queue until the queue is empty.

➢ ***Directive Router***

Directive router receives the directive from directive processor and route it to corresponding capability agent. Provides provision to add the directive handler, and maintains an array of pointer to handlers.

(E) Capability Agent or Alexa Interfaces:

➢ ***Speech Recognizer***

Responsible for forming recognize event and handle stop capture, expected speech directives.

Speech Synthesizer

- Handles speak directive, prepare audio player configurations and send events for playback started and playback finished.

(F) Audio Capture:

This component provides generic API for audio capture. These API will call project specific APIs for audio capture.

(G) Audio Playback:

This component provides generic API for audio playback. These API will call project specific APIs for audio playback.

3.3.3 AVS SDK Data Flow Diagram

Following diagram [23] shows data flow in Amazon AVS SDK:

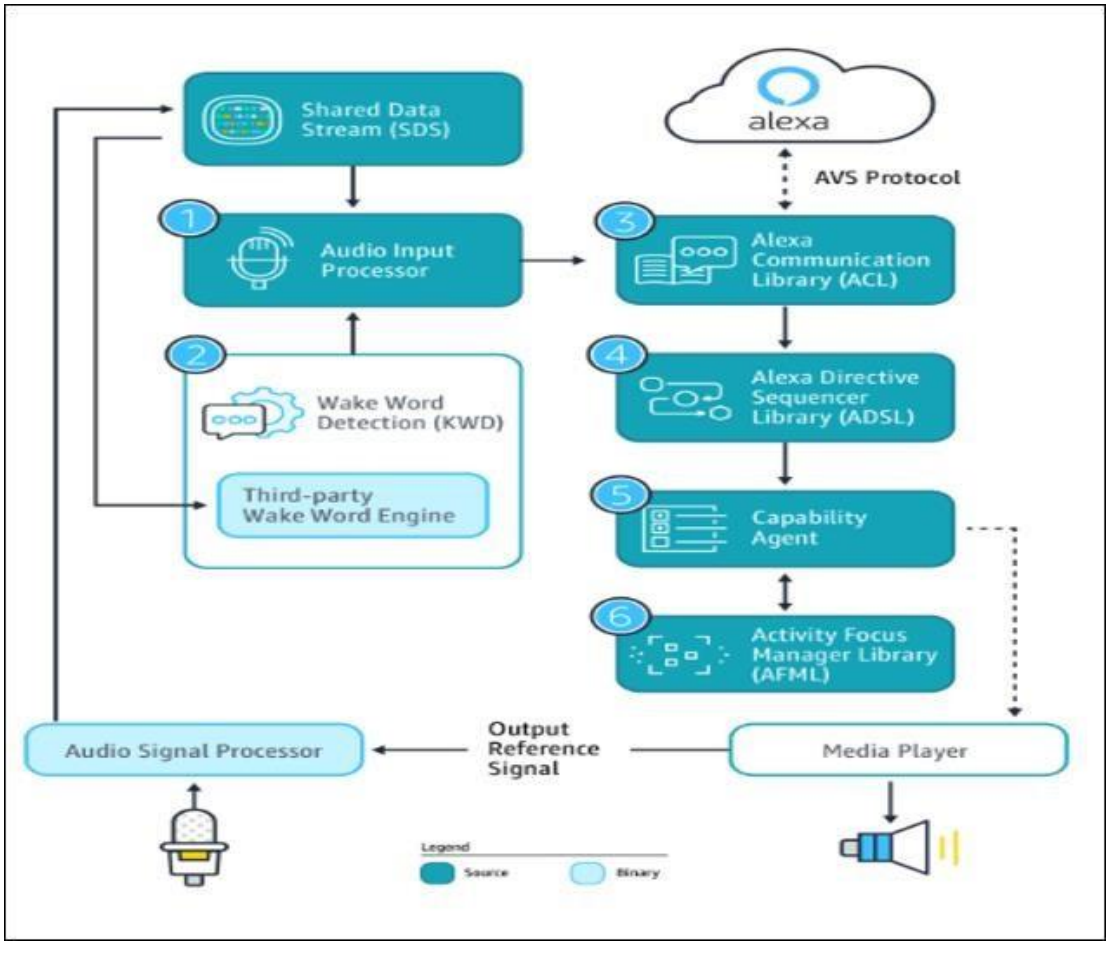

Figure 3.5: Amazon AVS SDK Data Flow Diagram [23]

3.4 SOFTWARE DEVELOPMENT KIT

3.4.1 Code Base Structure

The codebase structure for AVS SDK is organized in a systematic way as shown below:

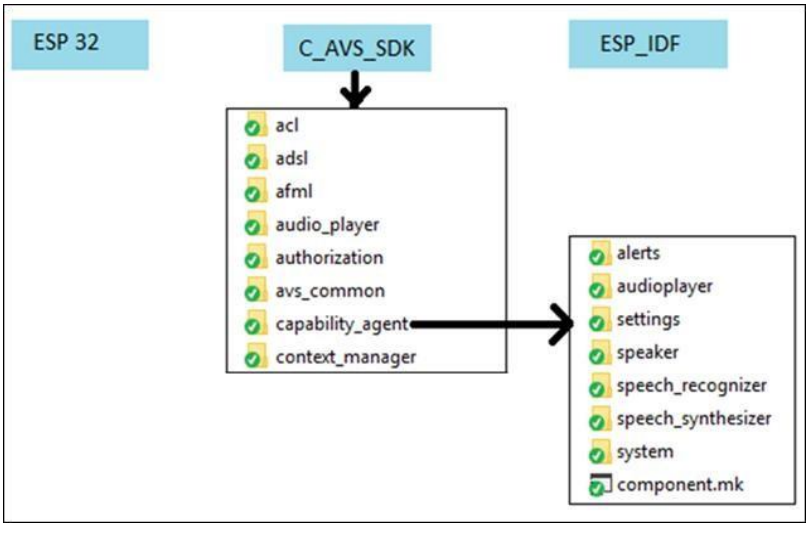

Figure 3.6: Code Base Structure for AVS SDK

3.4.2 Coding Guidelines Used

1. ***Doxygen Style:*** Use Doxygen style [27]. Doxygen is used for phrasing the code and the extracting commands to build documentation out of it.

2. ***Indentation:*** Use 4 spaces for each indentation level. Use of tabs is not allowed. Set configuration in the editor to emit 4 spaces for tab key.

3. ***Vertical Space:*** Place one empty line between two functions.

4. ***Horizontal Space:*** Put single space after loop and conditional keyword. Put single space around binary operators but no space is required around unary operators. No space is required around `.` and `->` operator. Never use tab for horizontal alignment. Do not use trailing whitespace at the end of line.

5. ***Braces:*** Use braces on separate line for function definition. Place opening braces on the same line within a function, for conditional and loop statement.

6. ***Comments:*** Use `//` for single line and `/* */` for multi-line comments.

CHAPTER 4

SYSTEM TESTING

4.1 SYSTEM TEST SPECIFICATIONS

All Alexa enabled devices must be Amazon for certification of media services prior to commercial distribution. The figure given below shows the flow of testing process for the product evaluation by Amazon:

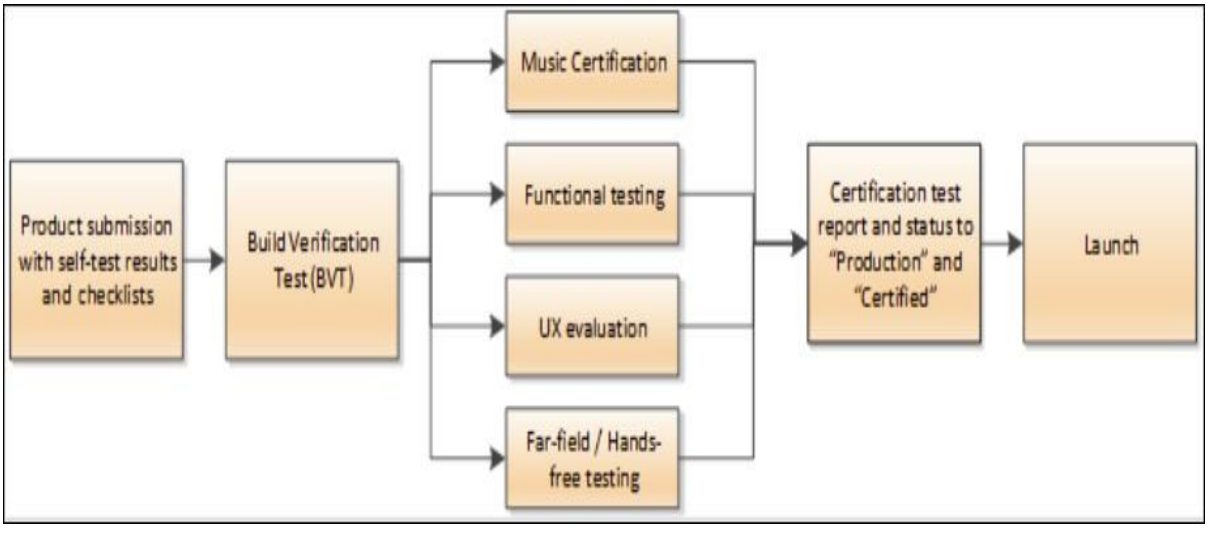

Figure 4.1: Flow of testing process for the product evaluation by Amazon

Device Under Test (DUT) is tested for **Wake Word False Rejection Ratio (FRR)** and **Response Accuracy Rate (RAR)** in various testing scenarios whereby a standardized set of pre-recorded speech is played from speaker to DUT under different noise conditions. Following are some terms used in testing:

- ***Wake Word (WW):*** Wake word is defined as the word that triggers the DUT to 'wake up'- "ALEXA"**.**
- ***Wake Word False Rejection Rate (FRR):*** Wake Word False Rejection Rate is calculated as (number of missed Wake words) / (number of Wake Word spoken).
- For example, if 10 requests are given and the device wakes up 9/10 times, and misses 1/10 times the FRR would be 1 missed wake-word / 10 wake-words spoken = 10%.
- ***Response Accuracy Rate (RAR):*** Response Accuracy Rate is calculated as (number of successful Alexa responses) / (number of Alexa requests).

- For example, if 10 requests are given and Alexa responds with a good response (defined in Section: Annotation guidelines) 9/10 times and a failed response 1/10 times, the RAR would be 9 correct responses / 10 requests = 90%.
- Additionally, in separate testing conditions the device is tested for Wake Word Detection Delay (WWDD) (aka Wake Word Delay in the context of the Reference Solutions Test) and Wake Word False Alarm Rate (FAR).
- ***Wake Word Detection Delay (WWDD):*** Wake Word Detection Delay is measured as the number of seconds required between the Wake-word and response for the device to provide the intended response ~100% of the time.
- ***Wake Word False Alarm Rate (FAR):*** Wake Word False Alarm Rate is calculated as the number of times the DUT wakes up in a 24-hour duration in the presence of audio playback from an external source.

4.2 TEST ENVIRONMENT

4.2.1 Hardware Environment

For setting up the hardware test environment to test the **"Device Under Test (DUT)"**, **"Amazon Far-Field Reference Solution (FFRS) Test"** has been followed. The purpose of the FFRS is to gather an assessment of the far field performance of the DUT.

Materials (Devices and Equipment): This section outlines the devices and equipment used for the Reference Solutions Test. All devices and equipment are used in each Testing Scenario unless specified otherwise.

- ***Device(s)***
    - ***One AVS DUT:*** DUT for AVS far field-testing. ➢ ***Equipment***
    - ***One Speech Speaker:*** Speaker for playing speech through an electronically balanced speaker.
    - ***One Noise Speaker:*** Speaker for playing noise (required for Music Noise, Kitchen Noise and FAR test).

- ***Two Speaker Stands (adjustable elevation; optionally three if needed for DUT):*** One for placing Speech Speaker on and one for placing Noise Speaker.
- ***Two Audio Line-out Cables (depends on setup):*** One cable for connecting Speech Speaker to Speech Laptop or Sound Card and one for connecting Noise Speaker to Noise Laptop or Sound Card (required for Music Noise, Kitchen Noise and FAR test). Ensure cables and adapters are of good quality and fit well.
- ***One Computer System:*** Needed to control the speech playback, noise playback and for annotating the results on the Reference Solutions Test scoresheet.
- ***One Sound Card:*** An external low noise sound card is required.
- ***Sound Pressure Level/SPL Meter:*** The SPL meter is used to measure the loudness of a sound source and is used in the Reference Solutions Test to measure the loudness level of speech and noise.
- ***Protractor/Angle Measuring Device:*** Protractor or other angle-measuring device for measuring azimuth angle for placement of Speech Speaker and Noise Speaker/Noise Laptop stands.
- ***Tape Measure:*** For measuring distance and elevation.
- ***Miscellaneous:*** Table, chair, power supply/extension cords.

4.2.2 Software Environment

- ***Audacity Software:*** To create a more repeatable setup we will use pink noise instead of section of a speech file. The amplitude shall be 0.1. This can be generated in audio editing software (like Audacity, which is free) but is intended to be provided in this test files. Set laptop volume to 50, external speaker knob to 5 (50%). Adjusted Audacity gain setting to meet 62dBC with playing Pink noise (Normalized to -32dB).

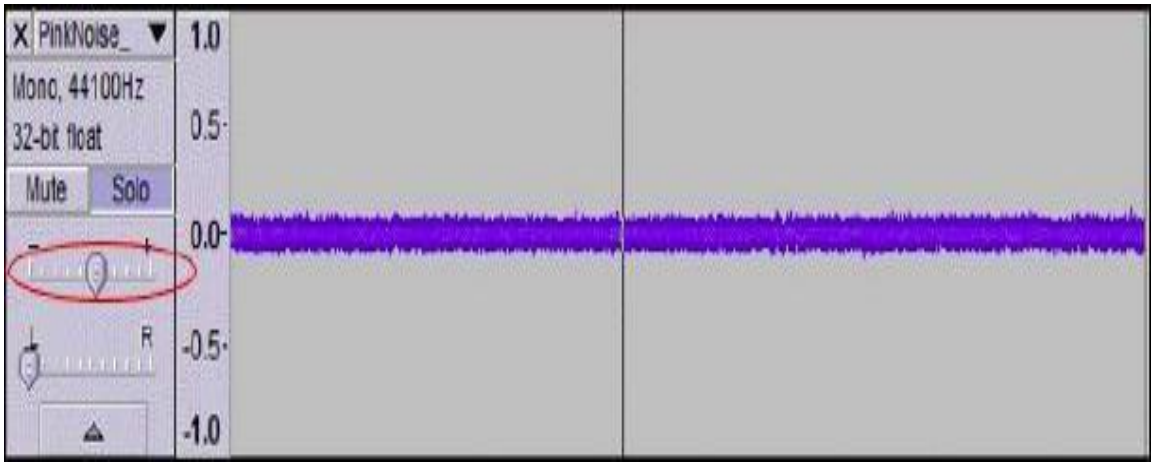

Figure 4.2: Gain Settings adjustment in Audacity Software

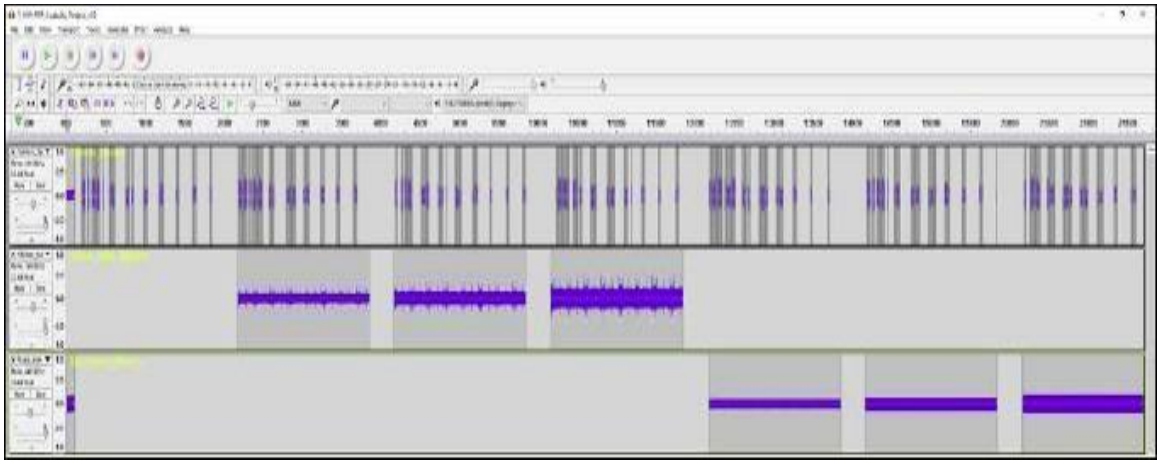

Figure 4.3: Audacity gain setting to meet 62dBC with playing Pink noise (Normalized to -32dB)

4.3 TEST PROCEDURE

4.3.1 Setting Up the Test Environment:

DUT, Noise Speaker and Speech Speaker were placed as shown above, by using protractor to measure Azimuthal Angle. For each Noise Condition, the following Speech Speaker azimuth angle and distance configurations are used:

 Speech Speaker Location 1: Distance 6 feet; Azimuth angle 90 degrees – All Scenarios.

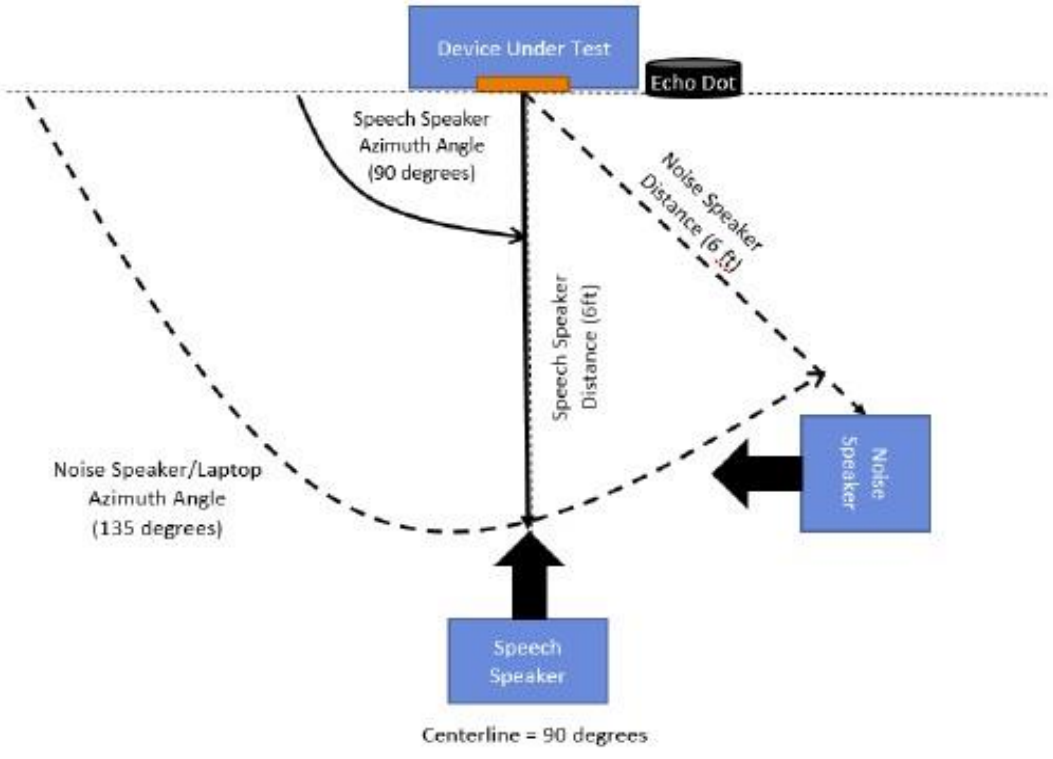

Figure 4.4: Aerial view diagram of a testing environment illustrating azimuthal angle and distances of Speech Speaker and Noise Speaker positions in relation to the AVS DUT

The Noise Speaker (used for Kitchen Noise Condition and FAR test) when used has a location of an azimuth 135 degrees at a distance of 6ft. The AVS DUT should be placed at an elevation/height that you determine to be a typical use case scenario. When comparing the two, the DUT and the Echo device should be on the same horizontal plane. The Speech Speaker and Noise Speaker are expected to be on the same horizontal plane.

4.3.2 Sample Test Cases

Following are test scenarios/ test cases [23] to test the DUT and expected response from the DUT:

Question	Expected Response from DUT
Alexa, what time is in Las Vegas in Nevada?	The time in Las Vegas, Nevada is x:xx.
Alexa, what is the capital of Cuba?	Cuba's capital city is Havana.
Alexa, remind me to buy Panasonic earbud headphones.	When should I remind you? Or Reminder is not currently supported on this device.
Alexa, who wrote the Portrait of Lady?	The Portrait of Lady's author is Henry James.
Alexa, how do you spell HAS?	"Has" is spelled: H.A.S.
Alexa, what time is in Harrisburg?	The time in Harrisburg is xx:xx.
Alexa, what is the capital of Egypt?	Egypt's capital city is Cairo.
Alexa, remind me to buy diapers.	When should I remind you? Or Reminder is not currently supported on this device.
Alexa, who wrote Great Expectations?	Great Expectations' author is Charles Dickens.
Alexa, how do you spell HOW?	"How" is spelled: H.O.W.
Alexa, what time is in Saint Paul?	The time in Saint Paul, Minnesota is xx:xx.
Alexa, what is the capital of South Korea?	South Korea's capital city is Seoul.
Alexa, remind me to buy Rubbermaid food storage container.	When should I remind you? Or Reminder is not currently supported on this device.
Alexa, who wrote Crime and Punishment?	Crime and Punishment's author is Fyodor Dostoyevsky.
Alexa, how do you spell USE?	"Use" is spelled: U.S.E.
Alexa, what time is in Hartford?	The time in Harford, Connecticut is xx:xx.
Alexa, what is the capital of El Salvador?	El Salvador's capital city is San Salvador.
Alexa, remind me to buy a brown ladder laptop bag.	When should I remind you? Or Reminder is not currently supported on this device.
Alexa, who wrote Alice in Wonderland ?	Alice's Adventures in Wonderland's author is Louis Carroll.
Alexa, how do you spell INK?	"Ink" is spelled: I.N.K.
Alexa, what time is it in San Luis Obispo California?	The time in San Luis Obispo, California is xx:xx.
Alexa, what is the capital of Canada?	Canada's capital city is Ottawa.
Alexa, remind me to buy waterproof jacket for men.	When should I remind you? Or Reminder is not currently supported on this device.
Alexa, who wrote Twenty Thousand Leagues Under the Sea?	Twenty Thousand Leagues Under the Sea's author is Jules Verne.
Alexa, how do you spell EAT?	"Eat" is spelled: E.A.T.
Alexa, what time is in Phoenix?	The time in Phoenix, Arizona is xx:xx.
Alexa, what is the capital of Jamaica?	Jamaica's capital city is Kingston.
Alexa, remind me to buy summer beach dress.	When should I remind you? Or Reminder is not currently supported on this device.
Alexa, who wrote War and Peace?	War and Peace's author is Leo Tolstoy.
Alexa, how do you spell PUT?	"Put" is spelled: P.U.T.

After executing each test case during testing, the results shall be recorded as follows:

Highlighted cells to be filled in by tester	
1	Complete success (Wake and complete phrase understood / task completed)
i	Device woke and understood the intent (like spelling) but had wrong word
w	Device woke but didn't get intent or any response
0	Device didn't wake up

CHAPTER 5

RESULTS AND DISCUSSIONS

Discussion:

For this device, the tester started with the file with 0.2 sec delay. For each of 10 trials of playing the same utterance, 10/10 times returned good responses. Next, the tester played 10 trials of the file with the silence duration of 0.15 seconds and got 100. The same is the case for 0.1 seconds even. At 0.05 seconds we see the device performance deteriorating as the Device Under Test only returns the correct response in full 7/10 times or 70% of the time, while at 0 seconds worse at 40% even still.

	Device Under Test									
	Trial #									
time (ms)	1	2	3	4	5	6	7	8	9	10
0	w	1	1	i	i	1	1	i	i	i
50	i	1	1	1	i	1	1	i	i	1
100	i	1	1	1	1	1	1	1	1	1
150	1	1	1	1	1	1	1	1	1	1
200	1	1	1	1	1	1	1	1	1	1
250	1	1	1	1	1	1	1	1	1	1
300										
350										
400										
450										
500										

Figure 5.1: Sample Wake Word Delay Test Score Sheet

The following table test case to test wake word detection, speech recognizer, speech synthesizer and response from cloud:

	Test Scenarios	WW detected & Play/Alert dir received	Music/Alert Playback happened	WW detected & Speak dir received	Dialog/Alert/Music playback happened	Music/ Alert resumed	WW detected & 2ndSpeak received	Dialog playback happened	Comments/observation
2	Alexa Set an Alarm	yes	yes						GOOD: Alarm working.
3	Alexa Set an Timer	no							No Alert directive received, Speak directive received : Timer starts now.
4	Alexa Set an Reminder	no							Ask for time/reminder for what , expected speech not working.
5	Alexa Set an Alarm + Q1(Query1	yes	yes	yes	yes	yes	yes	yes	GOOD: Alarm playback, dialog playback, alarm resumed, dialog playback, alrm resumed ha
6	Alexa Set an Timer + Q1	no							No Alert directive received, Speak directive received : Timer startss now.
7	Alexa Set an Reminder + Q1	no							Ask for time/reminder for what , expected speech not working.
8	Alexa Sing a song	yes	no						ISSUE 1 : Got play directive, connected to server, then 257 error. No playback. Get stuck at th
9	Alexa Sing a song	yes	yes						ISSUE 2 : Got play directive, started playback, then after playing for ~15 sec playback stopp
10	Alexa Sing a song	yes	no						ISSUE 4: Got play directive, connected to server, then W: retry to stop decoder, Audio_Sal: er
11	Alexa Sing a song	yes	yes						ISSUE 5 : Got play directive, Started playback then after playing for 5 sec in middle playbac
12	Alexa Sing a song + Q1	yes	yes	yes	yes	yes			Good: Played dialog + music resumed.
13	Alexa Sing a song + Q1	yes	yes	yes	no				ISSUE 3 : Got play directive, started playback, then after getting speak directive, playback st
14	Alexa Sing a song + Q1	yes	yes	yes	no				ISSUE 2 : Got play directive, started playback, then after asking for Query playback stopped
15	Alexa Sing a song + Q1	yes	yes	yes	no				ISSUE 2 : Got play directive, started playback, then after asking for Query playback stopped
16	Alexa Sing a song + Q1 + Q2	yes	yes	yes	yes	yes	no		ISSUE 6: during second query error E: spi= 257, cap timed out xxxxxx
17	Alexa Sing a song + Q1 + Q2	yes	yes	yes	no				ISSUE 3 : Got play directive, started playback, then after getting speak directive, playback st
18	Alexa Sing a song + Q1 + Q2	yes	yes	yes	no				ISSUE 2 : Got play directive, started playback, then after asking for Query playback stopped
19	Alexa Sing a song + Q1 + Q2	yes	yes	yes	no				ISSUE 2 : Got play directive, started playback, then after asking for Query playback stopped
20	Alexa Sing a song + Q1 + Q2	yes	yes	yes	no				ISSUE 2 : Got play directive, started playback, then after asking for Query playback stopped
21	Alexa some query	yes	no						ISSUE 7 : Received StopCapture directive, Then Error : Corrupt heap, multi heap detected. Cra
22	Alexa Sing a song + Alarm	yes	yes	yes	yes	yes	yes	yes	GOOD: Music+ Alarm + Dialog All are working simultaneously. Focus manager working fine.

Figure 5.2: Test results for Wake Word False Rejection Rate (FRR) and Response Accuracy Rate (RAR)

Form the above test results it is found that sometimes DUT has not been triggered on Wake Word "Alexa". **4/22** times DUT got triggered but did not send Play/Alert directive. Then **18/22** times correct directive is received by DUT from AVS cloud. And **13/18** times the correct and expected action was performed by DUT. **11/13** times DUT detected Wake Word, captured query and sent speak directive in response to query. Then **4/11** times DUT performed correct and expected action.

```
I (2946283) HTTP2: submit nghttp2 ping
I (2978240) AudDrv-IPC: Phrase Detected Pri = 0x1b040001, ext = 0x00000000, notif_type = 0x4
I (2978241) AUDIO_SAL: Capture Audio Stream Got WOV!
I (2978245) AUDIO_SAL: exit wov_callback_function!
I (2978245) audio_buffer_manager: start to capture buffers
I (2978245) RECOGNIZER: is RECOGNIZING
I (2978261) RECOGNIZER: recognize event dialogue request ID = RecDialogRequestId-2
I (2978274) RECOGNIZER: Send AVS speech recognize
E (4864126) TLS_TRANSPORT: mbedtls_ssl_write returned -0x4e
W (4864126) HTTP2: Fatal error: The user callback function failed
I (4864128) HTTP2: POST DONE
E (4864130) TLS_TRANSPORT: mbedtls_ssl_write returned -0x4e
W (4864138) HTTP2: Fatal error: The user callback function failed
E (4864145) TLS_TRANSPORT: mbedtls_ssl_read returned -0x4c
E (4864151) HTTP2: tls read returned -76
W (4864156) HTTP2: Out of http2 main loop
I (678414) MES INTERPRETER: directive detected
I (678416) MES INTERPRETER: Directive: {"directive":{"header":{"namespace":"AudioPlayer","name":"Play","messageId":"85c63312-2c10-47c8-b14f-904016a1a5d7"
I (678634) DIRECTIVE SEQUENCER: Dialog request id present in directive
I (678639) MEDIA_PLAYER: Entry:[audioplayer_handle_directive]

I (678641) RECOGNIZER: is IDLE

I (678649) AUDIO_SAL: Capture initializing
I (678654) AUDIO_SAL: Capture: Audio Stream create Success!
I (678660) AUDIO_SAL: Capture: Audio Stream Set Config Success!
I (678666) AUDIO_SAL: Capture: Audio Stream Prepare Success!
I (678674) FOCUS_MANAGER: Acquiring channel Content
I (678679) MEDIA_PLAYER: Play directive received
I (678686) MEDIA_PLAYER: requesting [GET /ella_v1_220c8b5e5dec2cd7b6ac__Songs_en_au_ella_love_song.mp3?Expires=1525449230&Signature=XUNfG2Ho-n4oEJGTR3PP0
Host: dvi8md6049w6b.cloudfront.net:443
Connection: Keep-Alive
User-Agent: AVSSDK_Player/1.1.1
Accept: */*
```

Figure 5.3: Testing Logs

Results:

As a result, because the device requires 150 milliseconds of delay we say for the **Wake Word Detection Delay (WWDD)** of this device to be **150 milliseconds**.

After various phases of testing **Wake Word False Rejection Rate (FRR)** for the DUT is calculated as **8/10** i.e. **80 %.** So we can conclude that if user utters the Wake Word “Alexa” 10 times, then **8 times out of 10** DUT will detect and trigger (or wake up).

After various phases of testing **Response Accuracy Rate (RAR)** for the DUT is calculated as **8.1/10** i.e. **81 %**. If cloud send directive as response to user query or command 10 times then, **~8 times out of 10** DUT will perform correct and expected action.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

The objectives of thesis have been successfully achieved. From the Literature Survey, various observations, latest technologies in Speech Recognition and future scope were reported. Alexa Voice Services Software Development Kit (AVS SDK) for Speech Recognition Engine has been successfully developed. Which provides a capability to user to interact with device by Speech Interface. The results clearly highlight that performance of the developed AVS SDK can be improved to get better Wake Word False Rejection Rate (FRR) , and better Response Accuracy Rate (RAR) and minimum Wake Word Detection Delay (WWDD). In future, advancement and expansion of Speech Recognition techniques and Artificial Intelligence, Nueral Networks, Echo Cancellation, Beam Forming and Noise Reduction techniques will enhance the quality and efficiency of Speech Recognition Engines.

The work can be furthur extended by improving follwing features and parameters in context of speech recognition:

- High Wake Word False Rejection Rate (FRR)
- High Response Accuracy Rate (RAR)
- Minimum Wake Word Detection Delay (WWDD)
- Enhanced Audio Voice Quality
- Supporting Multi Room Music (MRM) playback
- Enhanced Eco Spatial Perception

REFERENCES

[1] Abdur Razzaque Mohammad, Milojevic-Jevric Marija and Palade Andrei, Middleware for Internet of Things: A Survey, *IEEE Internet of Things Journal*, 3(1), 70-95.

[2] Pundir Yogita, Sharma Nancy and Singh Yaduvir, Internet of Things (IoT): Challenges and Future Directions, *International Journal of Advanced Research in Computer and Communication Engineering*, 5(3), 960-964.

[3] Kaushik Anupama**,** IOT-An Overview, *International Journal of Advanced Research in Computer and Communication Engineering*, 5(3), 1098-1100.

[4] K. Govinda and Saravanaguru R.A.K, Review on IOT Technologies, *International Journal of Applied Engineering Research*, 11(4), 2848-2853.

[5] Tebje Kelly Sean Dieter, Suryadevara Nagender Kumar and Mukhopadhyay Subhas Chandra, Towards the Implementation of IoT for Environmental Condition Monitoring in *Homes, IEEE Sensors Journal*, 13(10), 3846-3853.

[6] Chen Shanzhi, Xu Hui, Liu Dake and Wang Hucheng, A Vision of IoT: Applications, Challenges, and Opportunities With China Perspective, *IEEE Internet of Things Journal*, 1(4), 249-259.

[7] Stankovic John A., Research Directions for the Internet of Things, *IEEE Internet of Things Journal*, 1(1), 3-9.

[8] Al-Fuqaha Ala, Guizani Mohsen and Mohammadi Mehdi, Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications, *IEEE Communication Surveys & Tutorials*, 17(4), 2347-2376.

[9] Duarte Tiago, Prikladnicki Rafael and Calefato Fabio, Speech Recognition for Voice-Based Machine Translation, published by *The IEEE Computer Society*, February 2014, 26-31.

[10] Gavat Inge, Militaru Diana, Deep Learning in Acoustic Modeling for Automatic, Speech Recognition and Understanding: An Overview, *IEEE, 2015*.

[11] Hahm Oliver, Baccelli Emmanuel and Petersen Hauke, Operating Systems for Low-End Devices in the Internet of Things: A Survey, *IEEE Internet of Things Journal*, 3(5), 720-734.

[12] Khoubrouy Soudeh A. and Hansen John H. L., Microphone Array Processing Strategies for Distant-Based Automatic Speech Recognition, *IEEE Signal Processing Letters*, 23(10), 13441348.

[13] Chung Hyunji, Iorga Michaela and Voas Jeffrey, "Alexa Can I Trust You?", published by *IEEE Computer Society*, 2017, 100-104.

[14] Yue Chan Zhen and Ping Shum, Voice Activated Smart Home Design and Implementation, 2017, *2nd International Conference on Frontiers of Sensors Technologies*, 489-492.

[15] Rubio-Drosdov Eugenio, Díaz-Sánchez Daniel and Almenárez Florina, Seamless Human Device Interaction in the Internet of Things, *IEEE Transactions on Consumer Electronics*, 63(4), November 2017, 490-498.

[16] Rajalakshmi Anoja and Shahnasser Hamid, Internet of Things using Node-Red and Alexa, 2017, *17th International Symposium on Communications and Information Technologies (ISCIT).*

[17] Këpuska Veton and Bohouta Gamal, Next-Generation of Virtual Personal Assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home), *2018* IEEE, 99-103.

[18] Yiu K.F.C., Ho C.H. and Huo J.Q., An echo cancellation solution for voice control devices, *The 13th IEEE International Symposium on Consumer Electronics (ISCE2009)*, 32-33.

[19] Kamble Bhushan C., Speech Recognition Using Artificial Neural Network-A Review, *International Journal of Computing, Communications & Instrumentation Engg. (IJCCIE)*, 3(1), 1-4.

[20] Ma Jianguo, Internet-of-Things: Technology Evolution and Challenges, *2014 IEEE.*

[21] https://developer.amazon.com/alexa-voice-service/design

[22] https://developer.amazon.com/alexa-voice-service/dev-kits/intel-speech-enabling/

[23] https://developer.amazon.com/alexa-voice-service/sdk

[24] https://www.cnx-software.com/2017/06/19/intel-quark-s1000-sue-creek-processor-to-support-on-chip-speech-recognition/

[25] https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf [26] https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

[27] https://esp-idf.readthedocs.io/en/v1.0/style-guide.html

[28] https://developer.amazon.com/docs/alexa-voice-service/api-overview.html

[29] https://developer.amazon.com/docs/alexa-voice-service/manage-http2-connection.html [30] https://developer.amazon.com/docs/alexa-voice-service/authorize-companion-site.html

[31] https://www.freertos.org/

Turnitin Originality Report

NIDHI SHARMA

Roll No. 801661014 M.E.

E.C.E.

Submission date: 12-Jul-2018 06:44PM (UTC+0530)
Submission ID: 982068958
File name: ThesisReport_NidhiSharmaV3.pdf (1.32M)
Word count: 10924
Character count: 61731

DEVELOPMENT OF ALEXA VOICE SERVICES SOFTWARE DEVELOPMENT KIT FOR SPEECH RECOGNITION FOR INTERNET OF THINGS

A Thesis Submitted in partial Fulfillment of the Requirement for the Award of the Degree of

MASTER OF ENGINEERING

In

Electronics and Communication

Submitted By

NIDHI SHARMA

Roll. No. 801661014

Under Supervision of

DR. HEMDUTT JOSHI

Associate Professor

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY

(A DEEMED TO BE UNIVERSITY), PATIALA, PUNJAB

INTEL TECHNOLOGY INDIA PVT. LTD.

BANGALORE- 560103, KARNATAKA

JULY, 2018

thesis

ORIGINALITY REPORT

10%	6%	6%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijedr.org Internet Source	1%
2	developer.amazon.com Internet Source	1%
3	www.cnx-software.com Internet Source	1%
4	cse.iitkgp.ac.in Internet Source	<1%
5	Submitted to The University of Manchester Student Paper	<1%
6	Kanika Garg, Goonjan Jain. "A comparative study of noise reduction techniques for automatic speech recognition systems", 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2016 Publication	<1%
7	www.bvmengineering.ac.in Internet Source	<1%