DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

TEXT-TO-SPEECH SYNTHESIS FOR TURKISH USING A DSP BOARD

by
Uğur AYAZ

November, 2016
İZMİR

TEXT-TO-SPEECH SYNTHESIS FOR TURKISH USING A DSP BOARD

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science in Electrical and Electronics Engineering

by
Uğur AYAZ

November, 2016
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"TEXT-TO-SPEECH SYNTHESIS FOR TURKISH USING A DSP BOARD"** completed by **UĞUR AYAZ** under supervision of **ASSOC. PROF. DR. OLCAY AKAY** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Olcay AKAY

Supervisor

Assist. Prof. Dr. Yavuz SENOL

(Jury Member)

Assist. Prof. Dr. Yalçın İŞLER

(Jury Member)

Prof. Dr. E. İlknur CÖCEN
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENT

Firstly, I would like to thank my supervisor, Assoc. Prof. Dr. Olcay Akay, for his support and guidance during my thesis study.

I am grateful to all my theachers who have contributed to my intellectual development during my graduate education.

I also would like to thank my classmates and colleagues for supporting me over the years during the period of my graduate education.

Finally, I am indebted to all my family members for always believing in me and for their support under all circumstances.

Uğur AYAZ

TEXT-TO-SPEECH SYTHESIS FOR TURKISH USING A DSP BOARD

ABSTRACT

Speech synthesis has ben encountered more and more with the development of technology. It is utilized in varios devices and applications in our daily life. In this thesis, it is intended to design a standalone Turkish text-to-speech (TTS) synthesizer by using a digital signal processing (DSP) board. The method of linear predictive coding (LPC) was employed for synthesis of speech. The design work was carried out using the Texas Instrument's TMS320C5535 DSP development board which is a commonly used platform in voice applications.

Turkish text entered via computer is transferred to the DSP board after a linguistic analysis in the form of performing tokenization between words and syllables. Numerical expressions corresponding to each sound that was previously analyzed via the LPC method are stored in the DSP board permanently. Numerical expressions corresponding to each sound are once again converted into audio signals in the DSP board via some mathematical operations. The resulting audio signal is fed out via the onboard AIC3204 audio codec.

Synthesis results have been evaluated by listeners subjectively. Performance of the TTS synthesizer for Turkish language is measured using the method of mean opinion score (MOS). Based on the evaluation results, we conclude that intelligibility and naturalness of the synthesized sounds can be rated as in medium quality.

Keywords: Text-to-speech (TTS) synthesis, linear predictive coding (LPC), digital signal processing (DSP) board.

DSP KARTI KULLANARAK TÜRKÇE METİNDEN KONUŞMA SENTEZLEME

ÖZ

Teknolojinin gelişmesi ile birlikte konuşma sentezleme ile daha çok karşılaşmaktayız. Günlük hayatımızda pek çok cihaz ve uygulamda kullanılmaktadır. Bu tezde, bir DSP kartı kullanılarak metinden Türkçe konuşma sentezleyen bağımsız bir sistemin gerçekleştirilmesi amaçlanmıştır. Ses sentezleme için "doğrusal öngörücü kodlama" (LPC) yöntemi kullanılmıştır. Tasarım çalışması, sesle ilgili uygulamalarda sıklıkla kullanılan bir platform olan Texas Instrument şirketinin TMS320C5535 sayısal sinyal işleme (DSP) geliştirme kartı üzerinde gerçekleştirilmiştir.

Bilgisayardan girilen Türkçe bir metin, heceler ve kelimeler arasına çeşitli işaretler koymak suretiyle dilbilimsel olarak analiz edildikten sonra DSP kartına gönderilir. Daha önce LPC metodu ile analiz edilmiş her bir sese karşılık gelen sayısal ifadeler DSP belleğinde kalıcı olarak depo edilir. Her bir sese ait sayısal ifadeler DSP içerisinde matematiksel işlemler ile yeniden ses sinyaline dönüştürülür. Elde edilen ses sinyali geliştirme kartı üzerinde mevcut olan AIC3204 ses kodlayıcı-kod çözücü işlemci üzerinden dışarıya aktarılır.

Elde edilen sentezleme sonuçları öznel olarak dinleyiciler tarafından değerlendirilmiştir. Türkçe konuşma sentezleyicisinin performansı "ortalama görüş puanı" (MOS) yöntemi kullanılarak ölçümlenmiştir. Bu yöntemin sonuçları ışığında, sentezlenen seslerin anlaşılırlık ve doğallığının ortalama bir derecede olduğu sonucuna varılmıştır.

Anahtar kelimeler: Metinden konuşma sentezleme (TTS), doğrusal öngörücü kodlama (LPC), sayısal sinyal işleme (DSP) kartı.

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

1.1 Introduction

Communication between people has been provided by different means from past to present. Among those, speaking is the most common way of effective communication. Text-to-speech (TTS) synthesis forms an alternative form of communication between people and machines. It allows use of various applications as technology develops and enters our daily life more and more.

Blind people can understand text by using TTS technology. Similarly, speech-impaired people can better express themselves via this technology. Furthermore, TTS is a useful tool for learning new languages because it helps correctly pronounce difficult words.

Speech synthesizers are used in the commercial area as call center automation systems and in phone banking. Public transportation information systems, voiced navigation applications, robotics, and toys form some other application areas.

Figure 1.1 indicates the general concept of TTS production (Lemmetty, 1999). Written text is artificially converted into a speech signal by using different methods. One priority of any TTS application is making the text and linguistic analysis compatible with the rules of the language to be synthesized. Synthetic speech is generated by adding phonetic mapping onto the processed text.

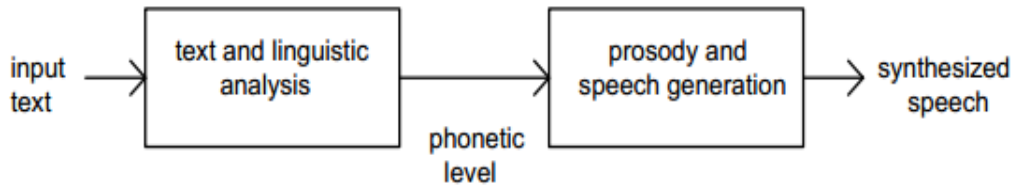


Figure 1.1 Block diagram of TTS production.

Development of a particular TTS method is not independent of the language. This is because pronunciation of the written text varies depending on the selected language (Lemmetty, 1999). Therefore, the linguistic rules should be well known for developing any particular TTS application. Adding linguistic rules into the developed application improves the quality of the synthesized speech.

Intelligibility and naturalness of the synthesized speech are two main goals of a TTS application. Intelligibility can be defined as the ability of listeners to decode a message from the synthesized speech (Taylor, 2009). Synthesized voice should be understandable and similar to human voice as much as possible. In addition, the system should be capable of producing any written input text.

1.2 General Speech Synthesis Methods

Articulatory synthesis, formant synthesis, and concatenative synthesis are three main categories of speech synthesis techniques (Tatham & Morton, 2005).

Articulatory synthesis is a method of speech synthesis by mimicking organs that help produce sound. It is based on the working principles of the articulators such as tongue, lips, jaw, palate, teeth, etc.

Formant synthesis techniques use speech formant parameters which are found by analyzing phonemes. Phoneme segments that are the smallest part of the speech can be modelled by mathematical functions. Speech is formed by assigning appropriate values to these functions.

Concatenative synthesis techniques make use of recorded original speech samples. Speech is generated by concatenating the recorded speech samples that consist of phonemes, diphones, and half-syllables. The most important difficulty with the concatenative synthesis methods is in joining the individual speech units with each other. The point of junctions should not be perceivable by listeners (Tatham & Morton, 2005).

1.3 Outline of the Thesis

As a formant synthesis technique, linear predictive coding (LPC) is utilized in this thesis. Computational load of LPC is overcome by using a particular digital signal processor (DSP) board. The audio codec of the DSP board helps in converting mathematical signals into audio outputs. In this thesis, Turkish speech synthesis is carried out with the limited embedded memory space of the DSP board.

The following chapters of the thesis is organized as follows. In Chapter 2, historical development of speech synthesis, literature review of speech synthesizers for Turkish and other languages, and milestones of speech synthesis are summarized. In addition, general speech synthesis methods are explained in detail. In Chapter 3, the employed DSP board and its features are introduced. Chapter 4 starts by explaining biological speech production and continues with the mathematical basis of the speech production used in this thesis. Explanations of the developed algorithm for speech production are also given in Chapter 4. Results and outputs are discussed in Chapter 5 and conclusions are given in the last chapter.

CHAPTER TWO
BACKGROUND ON TEXT-TO-SPEECH SYNTHESIS (TTS)

2.1 History of Speech Synthesis

Imitating the nature for scientific innovations has been one of the driving forces in human development. Nature is a source of inspiration for science. Artificial speech has attracted the attention of researchers from past to present. In the past, speech synthesis was realized by using mechanical devices, and hence, was low quality. At present, speech synthesis is performed by employing improved electronics technology such as high performance processors, and thus, possesses more intelligibility and naturalness.

Speech synthesis efforts started more than two centuries ago by adopting mechanical resonance to generate vowel pronunciations. Christian Kratzenstein noted the physiological differences between five long vowels /a /, /e /, / i /, / o /, and / u / and produced them artificially. As shown in Figure 2.1, the shapes of resonators differ, causing the generated sounds to be different as well. Kratzenstein's resonators won him the annual prize of the Russian Imperial Academy of Science in 1779 (Schroeder, 1972).

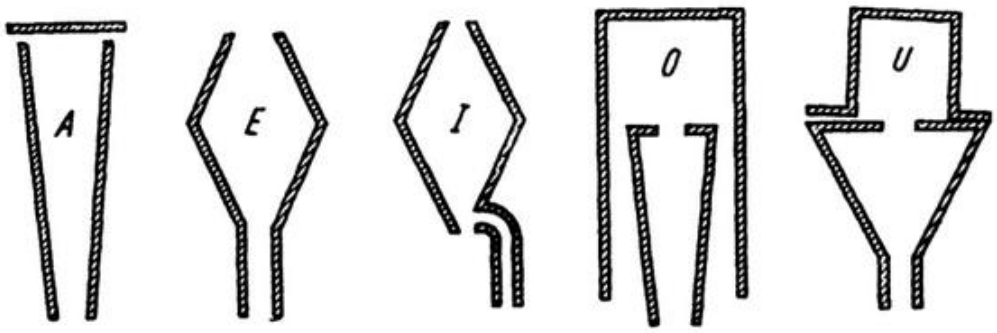


Figure 2.1 Kratzenstein's resonators.

During the same century, Wolfgang von Kempelen, who is a Hungarian author and an inventor, designed an acoustic mechanical speaking machine and published a book on speech sources in 1791. The contents of the book consisted of Kempelen's observations on human speech and his experiments on a speaking machine. The main

part of the machine was an air tank that mimicked human lung. The other parts of the machine were a vibrating metal reed to act as a vocal cord and a pliable leather tube imitating the vocal tract. Many different vowels could be produced by moving the mechanical parts of the machine (Schroeder, 1972). In 1838, Charles Wheatstone reconstructed Kratzenstein's speaking machine as shown in Figure 2.2. He added the theory of multiple resonances (Marschall, 2005).

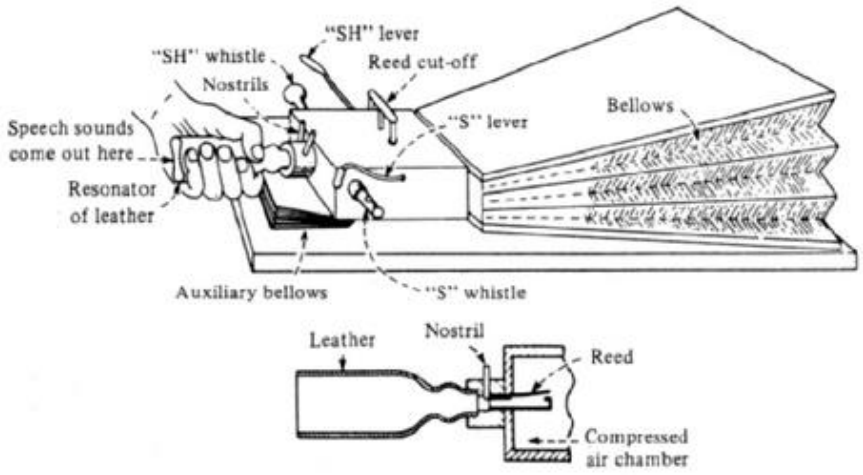


Figure 2.2 Wheatstone's reconstruction of von Kempelen's speaking machine (Flanagan, 1972).

Research efforts for mechanical speech synthesis and related experiments continued until the 1960s. Nevertheless, satisfactory results could not be achieved (Lemmetty, 1999).

During the 1900s, electrical synthesizer systems have gradually started to replace their mechanical counterparts. The first electrical voice synthesizer was developed by Stewart in 1922 (Klatt, 1987). As shown in Figure 2.3, the synthesizer has a buzzer that excites two different resonant circuits consisting of resistors, capacitors, and inductances. Thus, Stewart's synthesizer was able to produce vowel sounds with two formant frequencies.

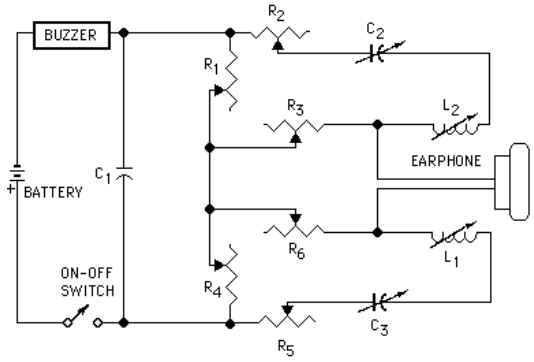


Figure 2.3 Stewart's voice circuit (Haskins Laboratories, n.d).

In 1939, a new device called Vocoder (voice coder) was developed by the employees of Bell Telephone Laboratories. It can be considered as the first true speech synthesizer, because it attempted to produce connected speech. This device analyzed speech into slowly changing acoustic parameters, and then drived a synthesizer to reconstruct an approximation of the original waveform. This led to the new idea of a human controlled Vocoder, which itself was called Voder (Voice Operating DEmonstratoR) (see Figure 2.4). Thus, the synthesizer operator could select either a voice source or a noise source by using a foot pedal in order to control the fundamental frequency of sound vibration. In addition, the source signal passes to the resonance control section that consists of ten band-pass electronic filters controlled by the operator's fingers (Dudley, 1939; Klatt, 1987).

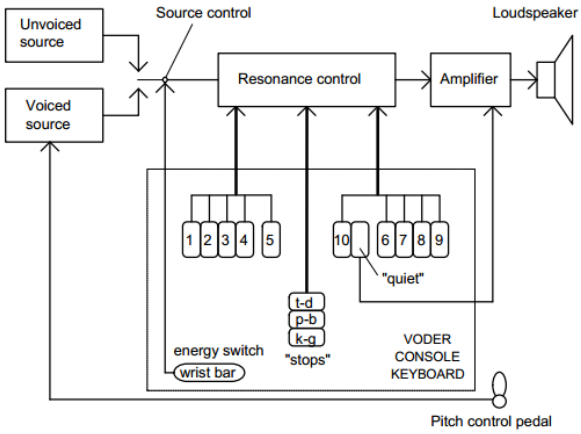


Figure 2.4 Voder speech synthesizer (Dudley, 1939).

In 1951, a pattern playback machine was developed by Franklin Cooper and his associates at Haskins Laboratories. As shown in Figure 2.5, the principle of the machine was to convert the recorded spectrogram pattern into a sound signal with the aid of an optical system. Although using a constant pitch period caused an unnatural sound, intelligibility was more than adequate for their purposes (Klatt, 1987).

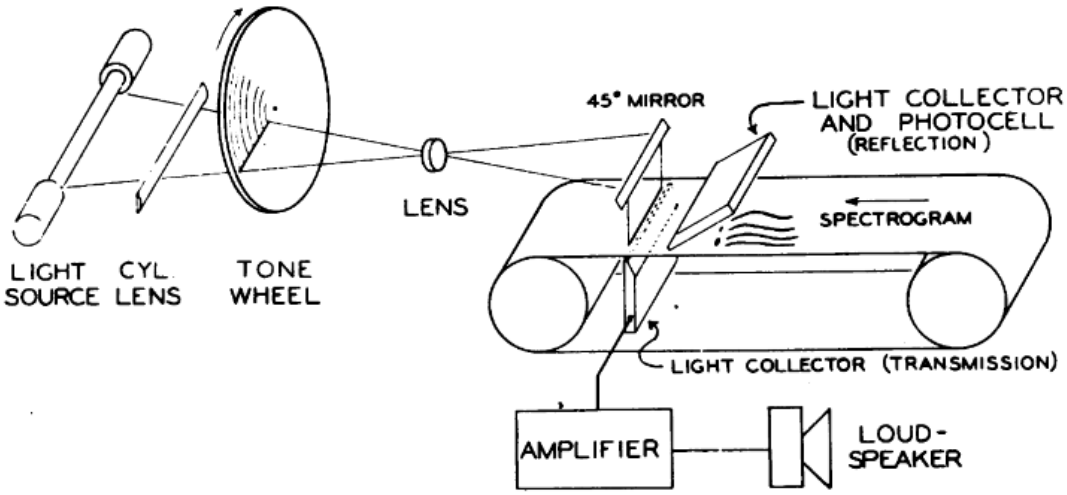


Figure 2.5 Operating principle of pattern playback (Copper et al., 1951).

In 1953, the first formant synthesizer PAT (Parametric Artificial Talker) was developed by Walter Lavrance. PAT had three parallel connected electronic formant resonators whose inputs were a buzz or noise. During the same years, Gunnar Fant developed the first cascade connected formant resonator named OVE (Orator Verbis Electris) I. The amplitude and frequency of the voiced vowel were tuned by potentiometers. In the following years, further improvements were made to create OVE II which possessed increased speech quality. Figure 2.6 shows the structure of OVE II. It consisted of three separate circuits to model transfer functions of vowels, nasals, and obstruent consonants (Klatt, 1987).

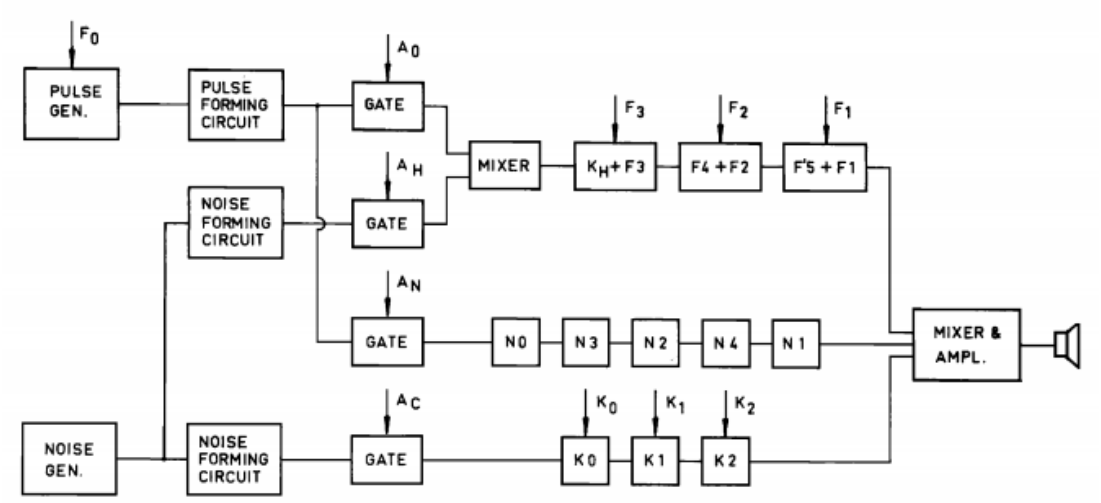


Figure 2.6 Structure of improved version of the OVE synthesizer (OVE II) (Klatt, 1987).

In 1958, the first articulatory model speech synthesizer was developed by George Rosen at the Massachusetts Institute of Technology (MIT). As shown in Figure 2.7, the articulatory synthesizer device DAVO (Dynamic Analog of VOcal tract) had hand adjusted variable inductors and capacitors for each section. To construct the vocal tract, the circuit was excited by a buzz source for voicing and by a noise source for consonants. Later, DAVO was further improved to approximate the nasal tract (Klatt, 1987).

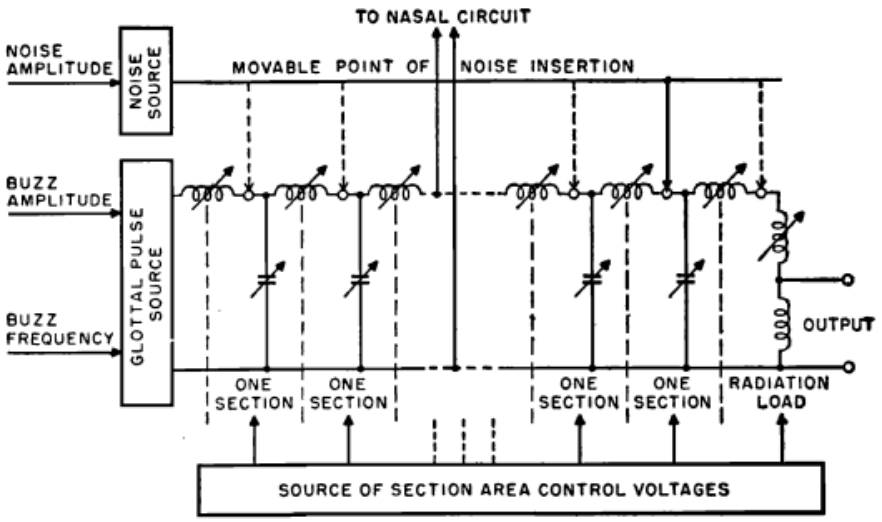


Figure 2.7 First articulatory model speech synthesizer (Klatt, 1987).

Rapid development of computer and microchip technologies enabled the development of TTS systems in a digital manner. After the 1960s, mechanical TTS systems turned into fully electronic systems. Analysis of written text allowed new TTS systems to be implemented. First trials of LPC based speech synthesis were made in the 1960s (Lemmetty, 1999). The first full TTS system was developed for English language in 1968 (Klatt, 1987).

Between the 1970s and 1980s, some commercial and research-based TTS projects were developed. Kurzweil designed a machine for the blind to read multifont written text in 1978. Texas Instruments developed a linear prediction based synthesis chip, TMS-5100. This chip was used inside TI Speak & Spell which was a toy developed for children as a hand-held computer to improve their reading skills. Figure 2.8 gives a chart illustrating the historical timeline of speech synthesis systems.

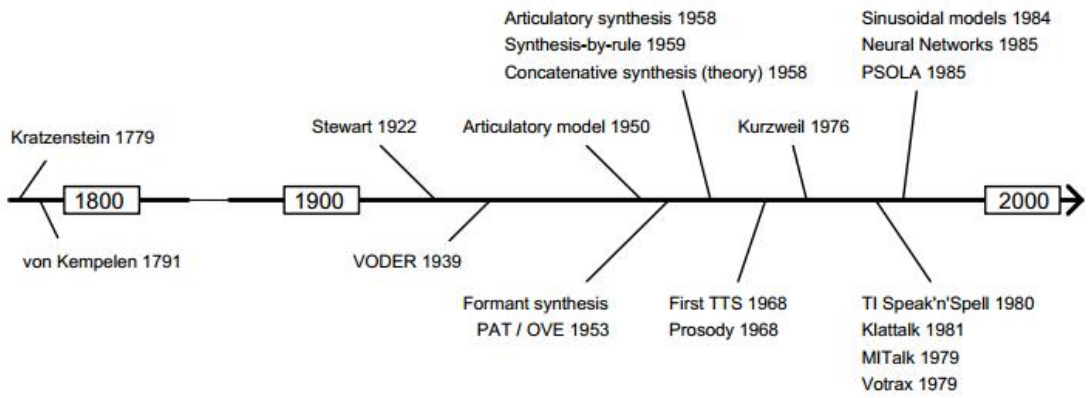


Figure 2.8 Historical timeline of speech synthesis systems (Lemmetty, 1999).

Today, speech synthesis systems have been developed exploiting the advantages of modern technology. Their calculation complexity is overcome by using high performance processors. Depending on the employed method, any needed extra memory can also be provided to improve a platform's storage capability.

After the development of TTS systems for English language, TTS systems for other languages spoken in the world have also started to come into existence. Although the same synthesis techniques are utilized, applications vary in accordance with the linguistic properties of the concerned language.

2.2 Speech Synthesis for Turkish Language

Turkish speech synthesis systems can be divided into two categories as academic and commercial. Compared to speech synthesizer systems for other languages such as English, development of speech synthesis systems for Turkish is rather new. However, discovered new techniques and quality improvements in TTS systems for other languages have been adapted to Turkish speech synthesis systems as well.

As shown in Table 2.1, Alper Gerçek, 1991, developed a speech synthesizer by using a specialized speech synthesizer processor TMS5220 that is produced by Texas Instruments in the 1980s. Table 2.1 also summarizes academical studies on Turkish speech synthesis systems together with the mathematical design methods adopted by them. In the early projects for the Turkish speech synthesis, LPC design method was utilized. However, more recent research projects used the concatenation method as an alternative approach with the aim of achieving more natural speech than LPC based methods.

Table 2.1 Academic studies on Turkish speech synthesis.

Year	Study	Author	Title	Method
1991	M.Sc.	Alper Gerçek	"A TMS5220 based speech synthesis development system"	Based on TMS5220 speech synthesizer processor and LPC
1992	M.Sc.	Karen Büyükkaşıkoğlu	"Analysis and synthesis of speech signals"	LPC
1992	M.Sc.	Enis Sezai Başara	"Yapay ses üretim yöntemleri"	LPC
1992	M.Sc.	Nevin Çizmecioğulları	"Implementation of LPC based voice communication system via DSP 56001"	Based on DSP56001 and LPC

Table 2.1 Academic studies on Turkish speech synthesis (continue).

Year	Study	Author	Title	Method
1993	M.Sc.	İlhan Yaşar Özüm	"A speech synthesis system for Turkish language based on the concatenation of phonemes taken from speaker"	Concatenation
1994	M.Sc.	Murat Servet Erer	"Text-to-speech in Turkish language by using a mixed speech synthesis method"	Concatenation
1994	M.Sc.	Selami Sadıç	"Türkçe ses sentezleyici"	LPC
1994	M.Sc.	Kamil Güven	"PC based speech synthesis for Turkish"	Formant synthesizer
1998	M.Sc.	Nihal Alıcı	"Türk dili için konuşma üretme"	Concatenation
1998	M.Sc.	Kerem Ayhan	"Text to speech synthesizer in Turkish using non parametric techniques"	TD-PSOLA
1999	M.Sc.	Özgür Salor	"Signal processing aspect of text to speech synthesizer in Turkish"	Concatenation
2000	M.Sc.	Ömer Eskidere	"Software based speech synthesizer"	Formant synthesizer
2000	M.Sc.	Barış Bozkurt	"Reading aid for visually impaired (a Turkish text-to-speech system development)"	TD-PSOLA
2001	M.Sc.	Çağla Ömür	"Concatenative speech synthesis based on a sinusoidal speech model"	Concatenation
2002	M.Sc.	Barış Eker	"Turkish text to speech system"	Concatenation
2002	M.Sc.	Şifa Serdar Özen	"Turkish text to speech synthesis"	Concatenation
2004	M.Sc.	Haşim Sak	"A corpus-based concatenative speech synthesis system for Turkish"	Concatenation
2005	M.Sc.	Asude Karlı	"A Turkish text-to-speech synthesizer for a set of sentences"	Concatenation
2007	M.Sc.	İlker Ünaldı	"Turkish text to speech synthesis system for mobile devices"	Concatenation
2009	M.Sc.	Zeliha Görmez	"Implementation of a text-to-speech system with machine learning algorithms in Turkish"	Concatenation

Table 2.1 Academic studies on Turkish speech synthesis (continue).

Year	Study	Author	Title	Method
2009	M.Sc.	Kenan Güldalı	"Turkish text to speech system"	Concatenation
2010	M.Sc.	Cavit Erdemir	"Natural speech synthesis for Turkish text-to-speech conversion"	Concatenation
2010	M.Sc.	Yücel Bicil	"Turkish speech synthesis"	Concatenation
2010	M.Sc.	Tuncay Şentürk	"Turkish text to speech synthesizer"	Concatenation
2011	M.Sc.	Güray Arık	"Enabling the use of computers for the visually impaired, accessibility and development of a Turkish syllable-based speech synthesis system"	Concatenation
2012	Ph.D.	İbrahim Baran Uslu	"Synthesizing natural speech from text using speech processing and linguistic properties of Turkish"	Concatenation
2013	M.Sc.	Ekrem Güner	"A hybrid statistical/unit-selection text-to-speech synthesis system for morphologically rich languages"	Concatenation
2013	M.Sc.	İlhami Sel	"Syllable based text to speech synthesis system for Turkish texts"	Concatenation
2014	M.Sc.	Erdem Erkan	"Developing speech engine for Turkish text"	Concatenation

Works on commercial Turkish speech synthesis have been increasing day by day in parallel to the development of technology and growing needs of people. Personal assistants, navigation systems, and phone banking form the major application areas. Some of the computer programs and Web applications contribute to the development of Turkish speech synthesis systems by supporting the Turkish language.

2.3 Speech Synthesis Methods

There are three essential approaches to speech synthesis. They are called articulatory synthesis, formant synthesis, and concatenative synthesis. The articulatory and formant synthesis methods are rule-based synthesis techniques. Producing speech using the concatenative synthesis method is easier than the other two alternatives

because the concatenative method does not require speech production rules. However, the main challenges of the concatenative method are resolving discontinuities and providing harmonization between speech units (Rashad, Hazem & Mastorakis, 2010).

2.3.1 Articulatory Method

The articulatory speech synthesis method is based on modelling the physical human vocal apparatus. Articulatory speech is mainly made up of two sections. One is the vocal fold model which represents excitation source and the other is the vocal tract model which describes the position of articulators like tongue, jaw, nose, etc. (Levinson, Davis, Slimon & Huang, 2012).

Understanding the articulatory speech synthesis method requires knowledge of human speech production mechanism. Figure 2.9 shows the human speech production organs and the corresponding idealized articulatory speech synthesis model.

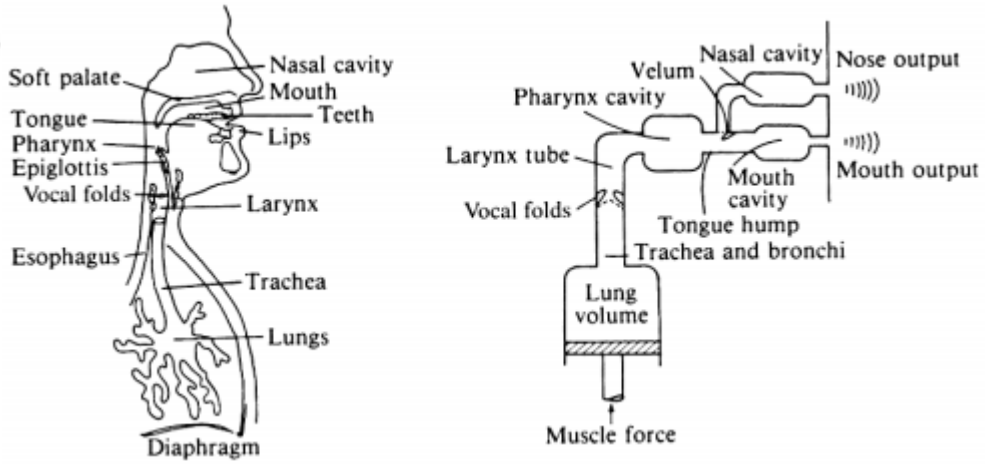


Figure 2.9 Comparison between human and articulatory speech mechanisms (Rossing, Moore & Wheeler, 2002).

The aerodynamic model of the wave propagation inside a tube is used to convert input parameters to sound. Theoretically, the articulatory model is the most effective way of generating natural sound which is similar in quality to human speech. However, many important problems need to be solved. One major problem is the lack of knowledge for the articulatory movement pattern (Levinson et al., 2012).

2.3.2 Formant Synthesis Method

The formant synthesis model provides an approximation to the speech waveform by a simplified set of rules formulated in the acoustic domain (Klatt, 1980). There are two common models for formant synthesizer as parallel and cascade. To attain high quality approximation to human speech, these two models are used together as a hybrid system. The cascade formant resonator is used for synthesizing voiced sounds and the parallel formant resonator is used for generating unvoiced sounds such as fricatives.

The input parameters of the formant synthesizer are calculated using the recorded real speech samples that represent the smallest part of speech. Theoretically, using calculated formant parameters that involve all the sounds of the synthesized specific language allows generation of all the words of the language.

The calculated present speech sample depends on the previous output parameters and the present input values. As a result, formant synthesizers need a fast computation system to work in runtime, even though they neither require large databases nor storage units.

Another rule-based speech synthesis method is called LPC which is very similar to formant synthesis. One notable difference is in the sound generating filter. LPC uses an all-pole filter as opposed to parallel filters that are common in formant synthesis (Taylor, 2009).

2.3.3 Concatenation Method

In the concatenative synthesis, speech is modelled as a sequence of individual sound segments (Tatham & Morton, 2005). Synthesizing speech by using concatenation method requires a database involving all possible sound segments. There are several types of sound segments depending on the size. Phonemes are the smallest sound units. Based on the selected synthesizer design, diphones or triphones can also be chosen as sound segments.

Speech is generated by combining recorded speech units according to particular rules. One important aim of these rules is to eliminate discontinuities between units such that a listener cannot perceive joints (Taylor, 2009). In various methods, the discontinuity problem is resolved using overlap and add operations.

One important trade-off in the concatenative speech synthesis method emerges when forming the database with the optimum pre-recorded unit length. Longer units provide more naturalness; however, more memory is needed for covering all word combinations. On the other hand, shorter length units require less memory space along with less natural sounds.

CHAPTER THREE
DSP DEVELOPMENT PLATFORM

TTS applications need a processor with high processing capability. Especially, speech synthesis via the LPC method requires implementation of intensive mathematical operations. An effective TTS synthesizer should not cause much delay between the input text and the speech output. DSP development platforms are specialized chips whose architecture is optimized for high level signal processing operations.

3.1 TMS320C5535 Development Platform

Texas Instrument's TMS320C5535 eZdsp kit has significant advantages for voice applications. This is because it has a programmable audio codec. Therefore, TMS320C5535 eZdsp development platform was selected for the implementation of this thesis. Figure 3.1 shows some features of the development platform. Further details regarding the DSP board and its processor are given in Appendices 1 and 2, respectively. Some of the key features of the Texas Instrument's TMS320C5535 eZdsp development kit are:

- Texas Instruments TLV320AIC3204 Stereo Codec (stereo in, stereo out)
- Micro SD card connector
- USB 2.0 interface to C5535 process
- I2C OLED display
- 8 Mbytes SPI flash

Figure 3.1 Key features of the development kit from top side (TMS320C5535 eZdsp technical reference, (n.d.)).

As shown in Figure 3.2, stereo in and stereo out terminals are directly connected to AIC3204 that is a flexible low power, low voltage stereo audio codec with programmable input and output features. AIC3204 audio codec connects to C5535 DSP via the I2S bus which is a standard bus to connect digital audio devices with each other. PCM audio data pass over the I2S bus. Key features of AIC3204 audio codec can be seen in the data sheet in Appendix 3.

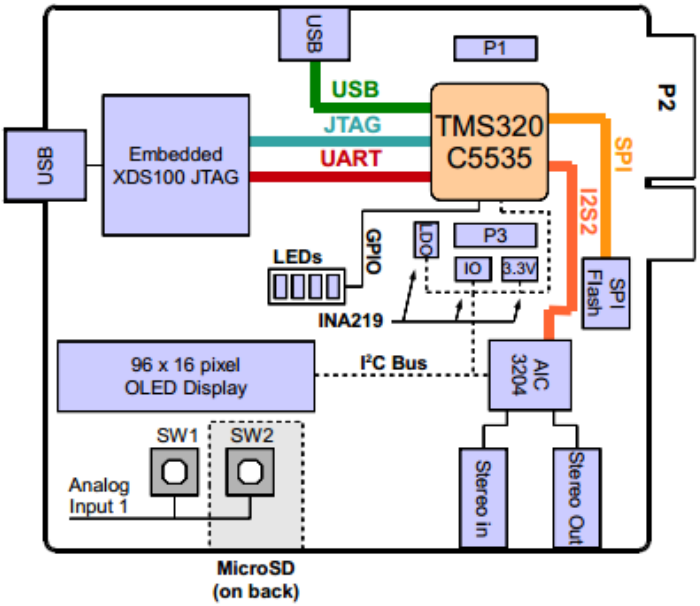


Figure 3.2 Block diagram of the development kit (TMS320C5535 eZdsp technical reference, (n.d.)).

3.2 Connections of the Development Platform

Connections of the developed synthesizer platform are demonstrated in Figure 3.3. Some peripheral devices are needed to be connected to the development kit directly. The text to be synthesized is forwarded to the development kit via the keyboard of the computer. This text is parsed and tokenized by the computer before it is sent to the development platform. Computer is also used to supply the development kit with voltage over the USB terminal.

The synthesized speech signal is converted into audio by AIC3204 audio codec. The development kit has two terminals for voice exchange with outside. A microphone can be connected to the stereo in terminal. In this thesis work, a speaker is connected to the stereo out terminal in order to listen to the synthesized speech.

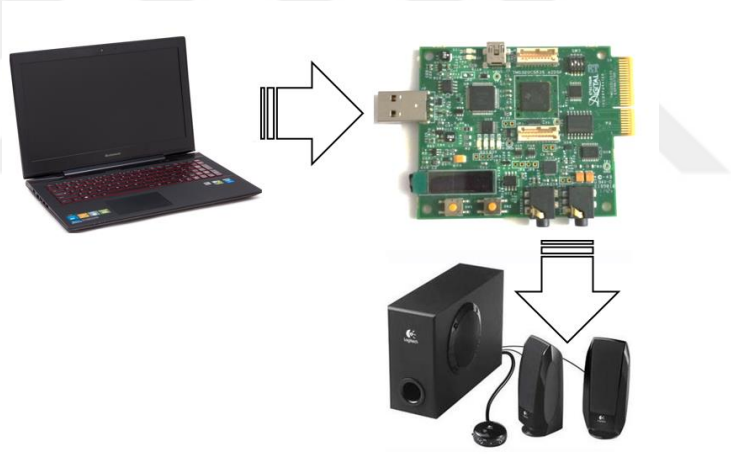


Figure 3.3 Connections of the development platform with peripheral devices.

The development platform was coded using Texas Instrument's Code Composer Studio integrated development environment which is based on C programming language. The text data is transferred from the computer to the development kit using the UART protocol on the USB terminal of the development kit. A MATLAB graphical user interface (GUI) was also designed for performing the pre-analysis of the text.

CHAPTER FOUR
METHODOLOGY

4.1 Human Speech Synthesis

Voice production is a complex process. Figure 4.1 shows the organs of the human speech production. Each organ has a different role. The characteristic and behavior of the organs change from person to person. This causes hearing of slightly distinct tones when the same word is pronounced by different people.

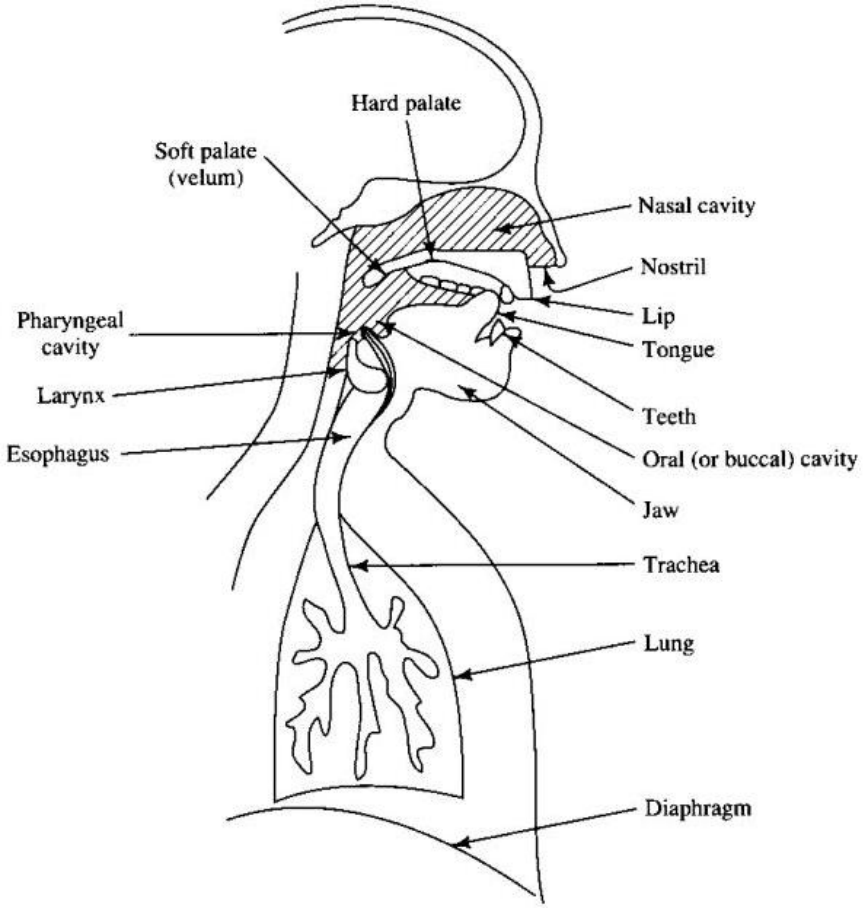


Figure 4.1 Human speech production organs (Biometric speech production, n.d.).

During speech production, the vocal cords vibrate and resist against the air. The main energy for speech production is supplied from the lungs and the diaphragm. Vocal cords play an important role in the generation of voiced and unvoiced sounds. They modulate the flow of air being expelled from the lungs. The fundamental

frequency of a voiced sound depends on the vibration of the vocal cords. There are three main cavities on the human vocal tube: pharyngeal, oral, and nasal. The air flow coming from the vocal cords passes on to the oral and nasal cavities and leaves through the mouth and nose as a spoken sound. Nostrils, teeth, jaw, lips, and tongue also help in the generation of utterances (Lemmetty, 1999).

Sounds are generally classified into two categories as voiced and unvoiced. Voiced sounds consist of a fundamental frequency and its harmonics which are generated by vibration of vocal cords. Thus, voiced sounds can be modelled by a fundamental frequency, F_0, bandwidth, and amplitude. Air flow is affected by vocal organs after passing the vocal cords. This causes a turbulence effect and the modulated air flow loses its periodic nature. White noise is used for modelling unvoiced sounds. Figure 4.2 shows the difference between the signals of voiced and unvoiced sounds. The signal of a voiced sound is quasi periodic and almost repeats itself.

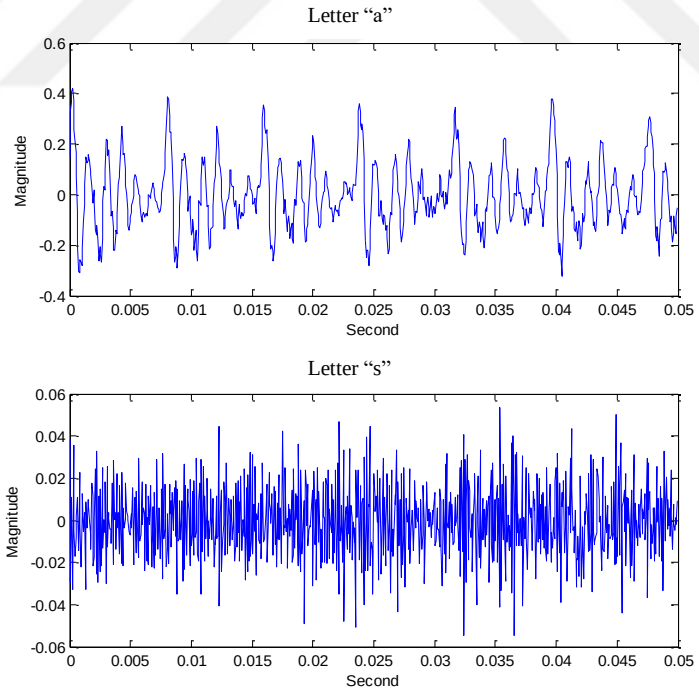


Figure 4.2 Waveforms of voiced “a” sound and unvoiced “s” sound.

4.2 Method of Linear Predictive Coding (LPC)

LPC can be defined as an encoding process by which the present value of a signal can be represented as a linear combination of its past values. LPC was developed with the main aim of encoding human speech. LPC model corresponds to a mathematical approximation of the human vocal tract. LPC is most widely used in speech coding, speech synthesis, speech recognition, and speaker recognition areas.

The procedure of LPC consists of two parts. The first part is called encoding or analysis part and the second part is decoding or synthesis part. In the encoding part, filter coefficients are determined by using a frame of speech. In the decoding part, speech is reproduced by using synthesized filter coefficients. Figure 4.3 shows the synthesis part of the LPC model. The parameters that are found after analyzing the speech frame are used to construct the speech signal. The system is excited either by a periodic pulse generator or white noise generator depending on the sound to be synthesized as voiced or unvoiced. The LPC method combines vocal tract, glottal pulse, and radiation characteristics of voiced speech to produce a sound. The synthesized speech can be modelled as in Equation (4.1) below

$$s[n] = \sum_{k=1}^{p} a_k s[n-k] + Gu[n]. \tag{4.1}$$

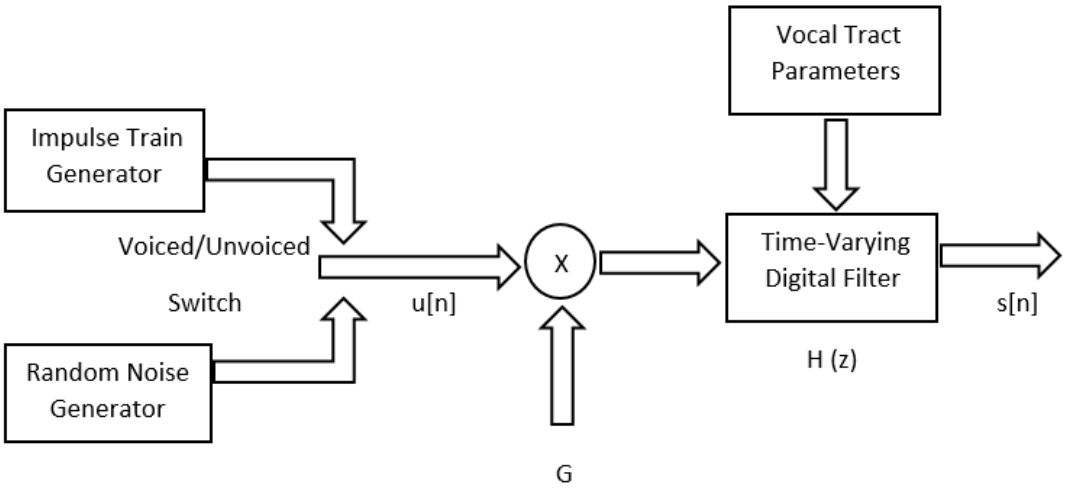


Figure 4.3 Speech generation via LPC.

The essential idea of the LPC is that the present speech sample can be closely approximated as a liner combination of past samples as follows

$$s[n] = \sum_{k=1}^{p} a_k s[n-k]. \tag{4.2}$$

Prediction coefficients, a_k, are determined by minimizing the sum of the squared difference between the referenced speech frame and the linearly predicted speech.

A time-varying digital filter represents human vocal tract. The all-pole filter model performs synthesis of voiced and unvoiced speech samples. Linear prediction estimate of $s[n]$ is obtained by using a p^{th} order prediction filter with transfer function

$$H(z) = \frac{S(z)}{G\mathrm{U(z)}} = \frac{1}{1-\sum_{k=1}^{p} a_k z^{-k}}\,. \tag{4.3}$$

The prediction error can be expressed as the difference between the reference signal and its estimate;

$$\hat{s}[n] = a_1 s[n-1] + a_2 s[n-2] + \cdots + a_p s[n-p] \tag{4.4}$$

$$e[n] = s[n] - \hat{s}[n] \tag{4.5}$$

$$e[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k]. \tag{4.6}$$

Equation (4.7) below indicates energy of the error signal. It must approach zero for the optimum prediction error;

$$\mathrm{E} = \sum_{n=1}^{N} e^2[n] = \sum_{n=1}^{N} (s[n] - \hat{s}[n])^2 \tag{4.7}$$

$$\mathrm{E} = \sum_{n=1}^{N} [s[n] - \sum_{k=1}^{p} a_k s[n-k]]^2 \tag{4.8}$$

$$\frac{\partial \mathrm{E}}{\partial \mathrm{a_i}} = 0, \text{ for } i = 1,2,3 \ldots, p \tag{4.9}$$

$$\frac{\partial \mathrm{E}}{\partial \mathrm{a_i}} = \sum_{n=1}^{N} 2\big[s[n] - \sum_{k=1}^{p} a_k s[n-k]\big][-s[n-i]] = 0 \quad (4.10)$$

$$\sum_{n=1}^{N} s[n]s[n-i] - \sum_{n=1}^{N}\sum_{k=1}^{p} a_k s[n-k]s[n-i] = 0 \quad (4.11)$$

$$\sum_{n=1}^{N} s[n]s[n-i] - \sum_{k=1}^{p} a_k \sum_{n=1}^{N} s[n-k]s[n-i] = 0 \text{ for } i = 1,2,3 \dots, p \quad (4.12)$$

$$\sum_{k=1}^{p} a_k \sum_{n=1}^{N} s[n-k]s[n-i] = \sum_{n=1}^{N} s[n]s[n-i] \text{ for } i = 1,2,3 \dots, p. \quad (4.13)$$

To analyze LPC parameters using autocorrelation method, it is assumed that the signal outside the window frame is zero. Hence, the analysis is made for the windowed speech frame. The autocorrelation terms are defined as

$$\sum_{n=1}^{N} s[n]s[n-i] = r[i]. \quad (4.14)$$

Thus, the set of equations can be written by using autocorrelation terms as

$$\sum_{k=1}^{p} a_k r[k-i] = r[i] \text{ for } i = 1,2,3 \dots, p. \quad (4.15)$$

$$\begin{gathered} a_1 r[0] + a_2 r[1] + a_3 r[2] + \cdots + a_p r[p-1] = r[1] \\ a_1 r[1] + a_2 r[0] + a_3 r[1] + \cdots + a_p r[p-2] = r[2] \\ a_1 r[2] + a_2 r[1] + a_3 r[0] + \cdots + a_p r[p-3] = r[3] \\ \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \\ a_1 r[p-1] + a_2 r[p-2] + a_3 r[p-3] + \cdots + a_p r[0] = r[p] \end{gathered} \quad (4.16)$$

Equation (4.16) can be written in matrix-vector form as follows

$$\begin{bmatrix} r[0] & \cdots & r[p-1] \\ \vdots & \ddots & \vdots \\ r[p-1] & \cdots & r[0] \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r[1] \\ \vdots \\ r[p] \end{bmatrix}. \quad (4.17)$$

Thus, the vector of unknown LPC coefficients, **a**, can be expressed as

$$\mathbf{a} = \mathbf{R^{-1}r} \quad (4.18)$$

To find the LPC coefficients, we have to invert the autocorrelation matrix, **R**. It is known as a Toeplitz matrix; that is, it is a symmetric matrix and its diagonal elements are equal. By using the Levinson-Durbin algorithm Equation (4.18) can be solved iteratively, as described in the following steps:

Set initial condition $E^{(0)} = \mathbf{R}(0)$ and start with i = 1.

1. Calculate $k_i = \frac{\left[\mathbf{R}(i) - \sum_{j=1}^{i-1} a_j^{(i-1)} \mathbf{R}(i-j)\right]}{E^{(i-1)}}$
2. Set $a_i^{(i)} = k_i$ and $a_j^{(i)} = a_j^{(i-1)} - k_i a_{i-j}^{(i-1)}$ for $1 \leq j \leq i$
3. Calculate $E^{(i)} = (1 - k_i^2) E^{(i-1)}$
5. Repeat Steps 2, 3 and 4 until $i = p$ which is the filter order.

4.3 Speech Database

The main component of the speech is sound. Sounds are represented by letters. There are 29 letters in the alphabet of the Turkish language. Letters are classified according to specific criteria. The main criterion of separation for letters is their formant frequencies at which the vocal tract resonates. Table 4.1 shows the vowel and consonant letters in the Turkish alphabet. Vowel sounds are produced by air flow coming from lungs without any resistance from the vocal tract. On the other hand, the air flow is exposed to obstacles for producing consonant sounds (Adalı, 2012).

Table 4.1 Vowel and consonant letters in the Turkish alphabet.

Vowel	a, e, ı, i, o, ö, u, ü
Consonant	b, c, ç, d, f, g, ğ, h, j, k, l, m, n, p, r, s, ş, t, v, y, z

The separation of vowel and consonant letters is used to generate spelling. Turkish is an agglutinative language so that words in the Turkish language are generated via linear combinations of morphemes which are the smallest units of speech bearing any meaning (Kuru & Akın, 1992).

Syllables perform an important role in the investigation of phonetics. In the Turkish language, there are six essential syllable types as shown in Table 4.2 where the

abbreviations “V” and “C” represent the vowel and consonant sounds, respectively. Syllabification is done on the basis of syllable structure of the Turkish language (Delibaş, 2008).

Table 4.2 Syllable structure of the Turkish language.

Pattern	**Samples**
V	a, e, ı, i, o, ö, u, ü
VC	al, er, ık, iş, ok, öm, un, üs
CV	ta, re, cı, di, ko, yö, zu, fü
CVC	kal, tek, kıs, lik, yok, kör, sur, tüm
VCC	alt, ilk, üst, ırk
CVCC	kart, renk, yurt

There are 8 vowel letters in the Turkish alphabet. They can be classified according to positions of the lips, jaw, and tongue while producing the vowel sounds. They are categorized as unrounded or rounded vowels according to lip position, wide or narrow vowels according to jaw position, and back or front vowels according to tongue position. Table 4.3 shows classification of Turkish vowel letters.

Table 4.3 Classification of Turkish vowel letters.

	Unrounded Vowel		**Rounded Vowel**	
	Wide Vowel	**Narrow vowel**	**Wide Vowel**	**Narrow vowel**
Back vowel	a	ı	o	u
Front vowel	e	i	ö	ü

Words are formed by a combination of significant syllables that are uttered in one breath. They do not need to be meaningful. There is a vowel letter in a syllable either as a back or front letter. Figure 4.4 demonstrates the relation between the vowel and consonant letters and joining syllables (Delibaş, 2008).

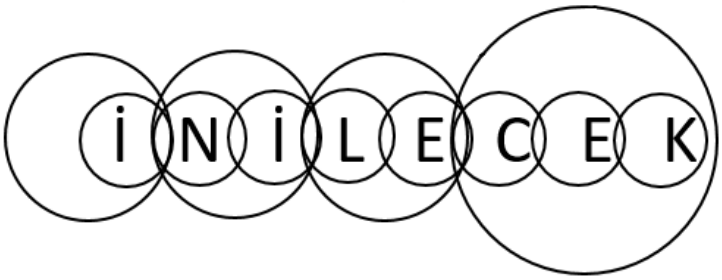


Figure 4.4 Word articulation of the Turkish language.

In this thesis work, 93 different letter samples were used to obtain the interaction between vowel and consonant sounds even though there are only 29 letters in the Turkish alphabet. For example, there are four different “b” sounds in the speech database depending on the pronunciation with the back vowel or front vowel. Hence, they were sampled by using the syllables, “ba”, “ab”, “be”, “eb”. The full list of the members of the employed speech database is given in Appendix 4. All the sounds in the speech database were recorded as mono using the voice of one single person.

4.4 Voiced and Unvoiced Letters

Speech signals can be classified as voiced and unvoiced depending on the vibration of vocal cords. In the LPC method, voiced sounds are generated by creating an impulse train to excite the vocal tract. Unvoiced sounds are produced when the vocal tract is in stationary position, creating turbulence passing through the vocal tract. By investigating the voiced and unvoiced sounds in the time domain, it can be seen that the time domain waveforms of voiced sounds are quasi periodic. The fundamental frequency of a voiced sound is called the pitch frequency. The frequency range depends on the gender. For males, the frequency range is between 50 Hz and 250 Hz, for females, it is between 120 Hz and 500 Hz (Chu, 2003).

There are several methods to parse voiced and unvoiced letters automatically. Zero crossing and energy based methods are two of those methods. In the zero crossing (ZRC) method, the sign changes of the sound signal are counted. As shown in Equation (4.19), the signum function is utilized for this purpose:

$$ZRC = \sum_{n=0}^{N-2} \frac{1-[\mathrm{sgn}(s[n]\mathrm{sgn}(s[n+1])]}{2} \quad (4.19)$$

Voiced and unvoiced letters can be estimated by calculating ZRC values. For intervals of 10 miliseconds, approximate values of *ZRC* are 12 for voiced and 50 for unvoiced sounds (Caruntu, Toderean & Nica, 2005).

Average sum of squared energy in Equation (4.20) below is another separation criterion between voiced and unvoiced letters. Higher energy values indicate voiced sounds. Unvoiced sounds have less energy than their voiced counterparts (Caruntu, Toderean & Nica, 2005).

$$E = \frac{1}{N}\sum_{m=0}^{N-1} \mathrm{s}^2[m] \quad (4.20)$$

Table 4.4 lists different forms of comparisons between voiced and unvoiced sounds.

Table 4.4 Comparison of voiced and unvoiced sounds.

ZRC comparison	**Energy comparison**	**Output**
Low	High	Voiced
High	Low	Unvoiced

Determination of voiced and unvoiced sounds is important for exciting the vocal tract in the LPC method. As shown in Figure 4.3, the vocal tract is excited by an impulse train for voiced sounds because waveforms of voiced sounds are quasi periodic. To synthesize aperiodic unvoiced sounds, on the other hand, vocal tract is excited by a noise like signal.

For the Turkish alphabet, classification of voiced and unvoiced letters is given in Table 4.5 (Türk Dil Kurumu, (n.d.)). All of the vowels and some consonant letters are classifieded as voiced. There are 8 unvoiced letters in the alphabet. While implementing this thesis work, this classification is taken into consideration. After

parsing the input text, a time varying filter is excited based on the synthesized letter type.

Table 4.5 Classification of voiced and unvoiced letters in the Turkish alphabet.

Voiced	a, b, c, d, e, g, ğ, ı, i, j, l, m, n, o, ö, r, u, ü, v, y, z
Unvoiced	ç, f, h, k, p, s, ş, t

4.5 Finding the Pitch Period

For quality of the synthesized speech, it is important to determine the correct pitch period which is used for exciting the time varying filter. The pitch period of the letter "a" is shown in Figure 4.5.

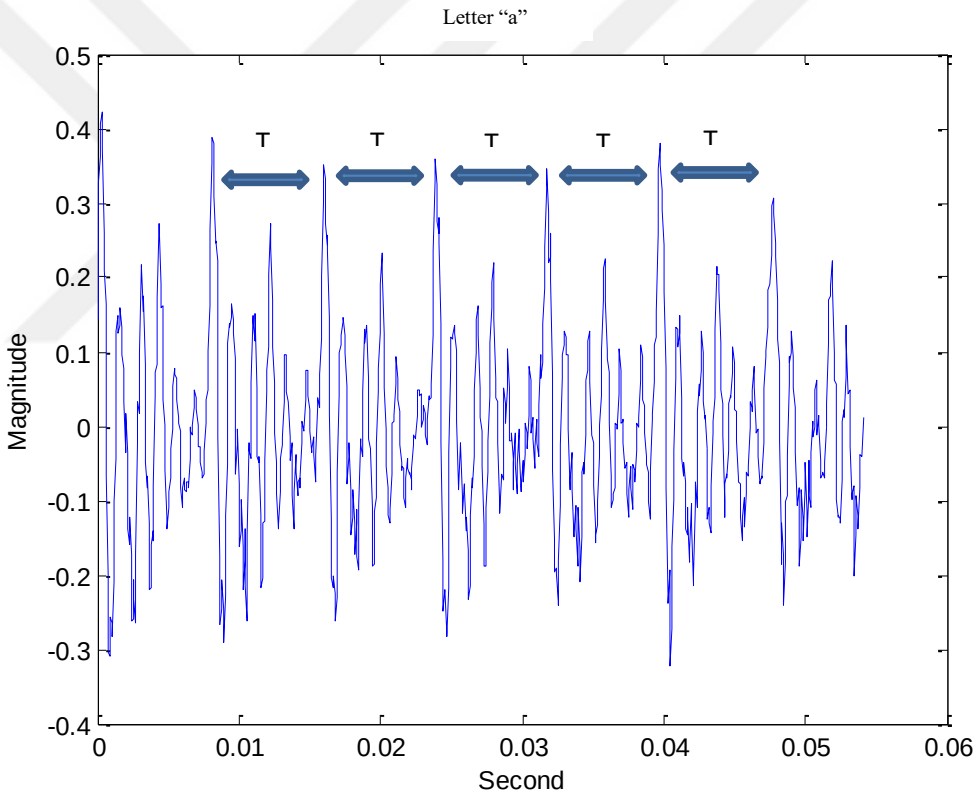


Figure 4.5 Pitch period representation of the letter "a".

When a voiced sound is examined in the time domain, it appears to repeat itself every T seconds as shown in Figure 4.5. Pitch period also corresponds to the inverse of the fundamental frequency. There are several frequency components that are harmonically related in a voiced speech signal. The least one of those harmonically

related frequencies is called the fundamental frequency of the signal (David, 2003). The power spectral density of the letter “a” is sketched in Figure 4.6 where the least frequency component is also indicated. It corresponds to the fundamental frequency of the letter “a”. The fundamental frequency depends on the speaker and his/her characteristic features.

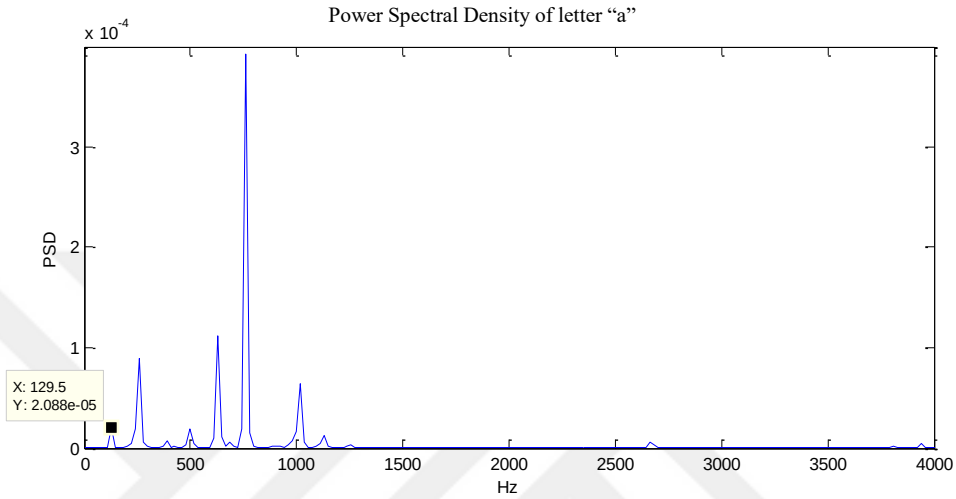


Figure 4.6 Power spectral density of the letter “a”.

Finding the fundamental frequency may be difficult in cases when a speaker’s speech could be stressed. Therefore, extraction of the fundamental frequency should be performed using unstressed speech samples. In the time domain, the fundamental frequency can be calculated manually by determining the distance between the signal samples that repeat themselves periodically. It can also be determined automatically by using the frequency domain analysis tools.

In this thesis, the correlation method has been utilized for estimating the fundamental frequencies of analyzed speech samples. These estimated values were used to determine the periods of the impulse train for exciting the time varying filter which represents the vocal tract. In the correlation method, similarity of the two waveforms is measured by comparing them in specific time intervals. The maximum similarity is expected to be at the zero time lag (David, 2003).

4.6 Flowchart of the Designed TTS Synthesizer

The workflow of the designed synthesizer can be divided into two parts which are performed by the computer and the DSP board, respectively. Figure 4.7 explains the processes realized by the computer. The raw text is transferred into the DSP board after making text parsing and tokenization with the help of MATLAB GUI. Then, the data reach to the serial communication buffer of the DSP board in the form of ASCII characters. When the stop message charter arrives at the DSP board, it finishes off the data transfer and the program jumps into the synthesis process.

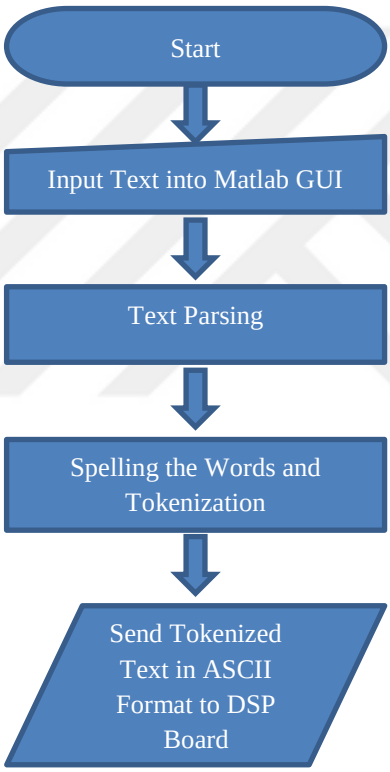


Figure 4.7 First part of the workflow of the TTS synthesizer.

The second part of the synthesizer workflow involves operations performed by the DSP board as demonstrated in Figure 4.8. Those operations start by getting the tokenized text to generate the speech. Speech signal is formed using the LPC method by assigning LPC parameters that are extracted from the speech database. The process of transferring text characters into the text buffer proceeds until reaching the end of the text. The synthesized speech signals are collected in the speech buffer consecutively. Then, the speech signal is sent into AIC3204 audio codec on the I2S

bus which is a specialized serial sound communication bus. AIC3204 has programmable inputs, outputs, and power tune. It allows up to 48 kbps DAC stereo playback and supports up to 100 dB DAC SNR value.

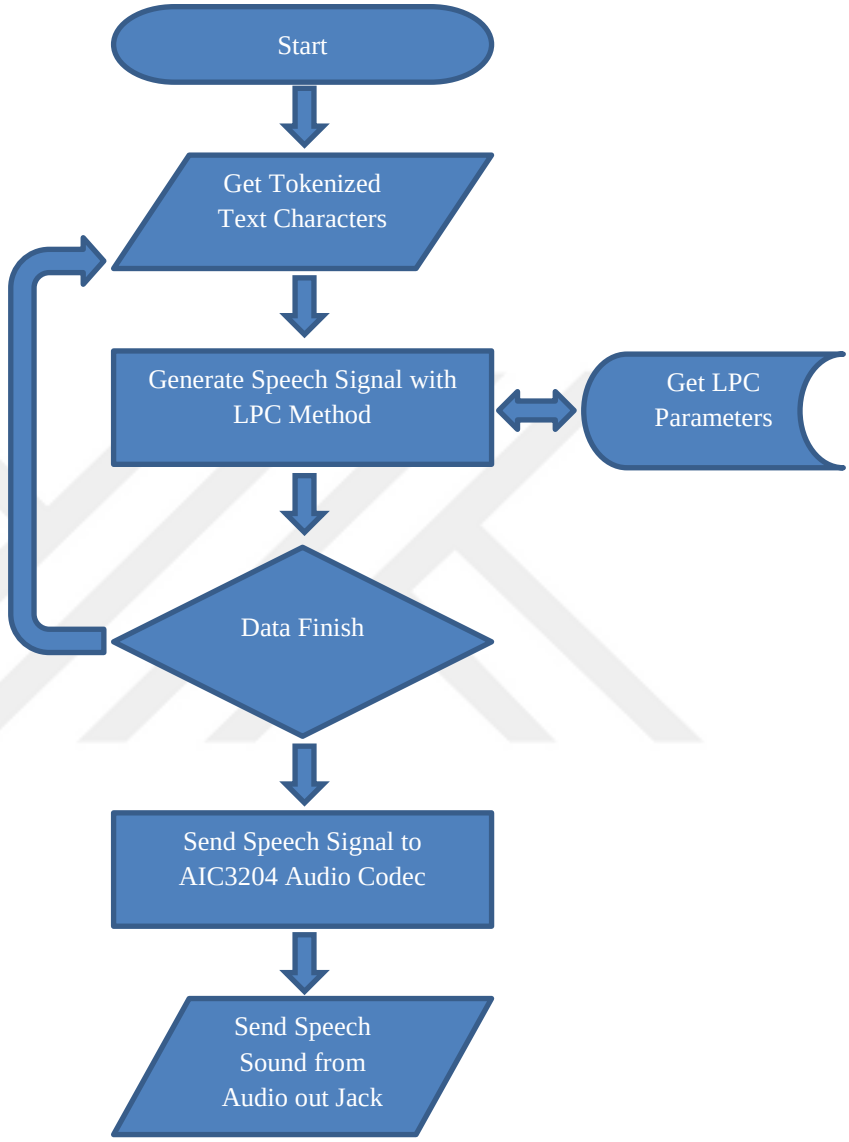


Figure 4.8 Second part of the workflow of the TTS synthesizer.

CHAPTER FIVE
EXPERIMENTS AND RESULTS

In TTS synthesis, it is important to employ proper speech units for analysis in order to obtain the best synthesized speech quality. Noise and environmental sounds should not interfere with the original speech units. Filtering before synthesis can be a solution for removing these undesirable components. Using clean speech units allows obtaining better synthesized speech.

5.1 Choosing LPC Parameters

The LPC method is extensively preferred for applications of speech synthesis, speech coding, speech recognition, and speaker recognition. The reason for extracting the LPC parameters from the actual speech samples is to construct a mathematical model for the speech signal. Error between the actual speech and the resultant synthesized speech determines the performance of the synthesizer. Figure 5.1 contrasts real speech production with the LPC model.

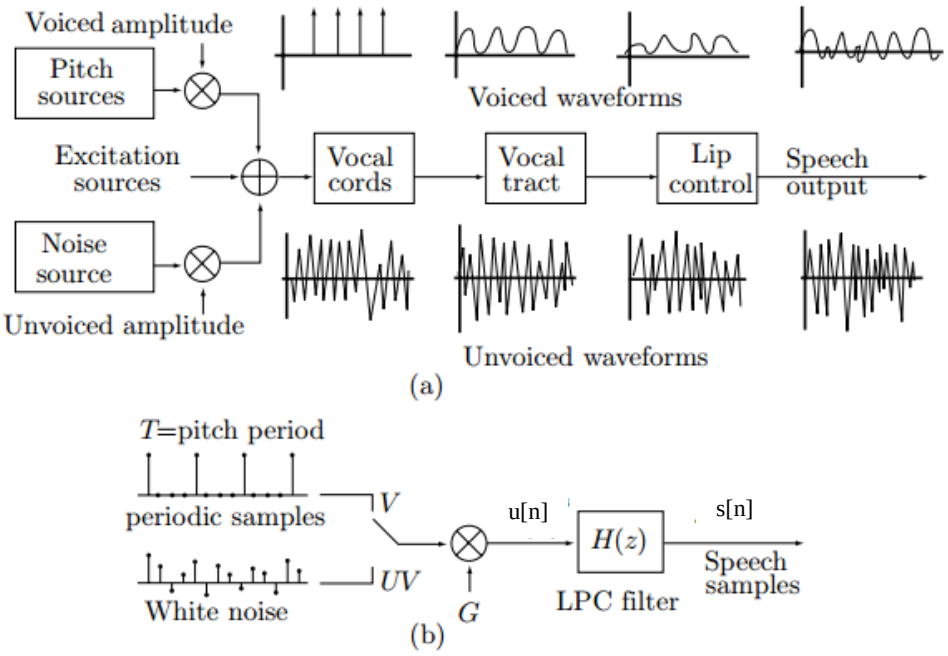


Figure 5.1 (a) Effect of vocal organs on speech production. (b) LPC model representation (Salomon, 2007).

Important LPC parameters can be listed as follows (Salomon, 2007):

- Knowledge of voiced and unvoiced sound information in relation to vibration of vocal cords.
- Pitch period of vocal cord vibration.
- Gain parameter of loudness related to volume of air coming from the lungs.
- LPC filter coefficients corresponding to each speech sample.

5.1.1 Ideal Speech Units for Analysis

Employed speech samples for extracting the LPC coefficients directly affect the quality of the resultant synthesized speech. As an example, Figure 5.2 shows the signal waveform of the letter “o” in our speech database. The waveform displays the quasiperiodic character of the signal with a particular period. Magnitude is rather diminished at the beginning and the end of the waveform. Similarly, it is possible to generate speech waveforms for all sounds.

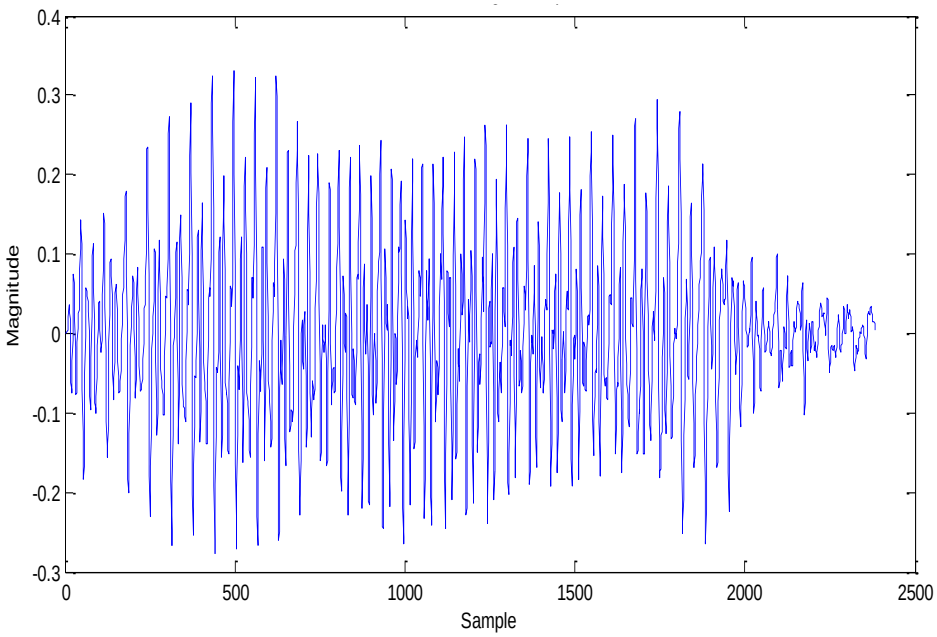


Figure 5.2 Signal waveform of the letter “o”.

Generally, envelopes of the time waveforms for letters can resemble the shape in Figure 5.3 that gives information about loudness of a sound at different time instants. This envelope representation is usually utilized for generating synthetic musical tones. In the attack phase, the sound reaches its peak volume. Decay, sustain, and release phases follow the attack phase. The time between the phases can be changed depending on the analyzed speech sample (Casabona & Frederick, 1987).

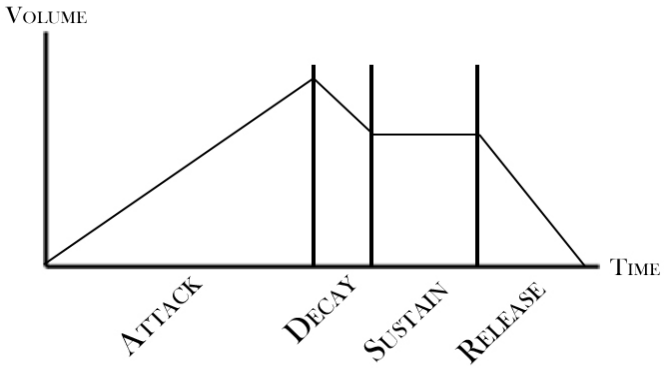
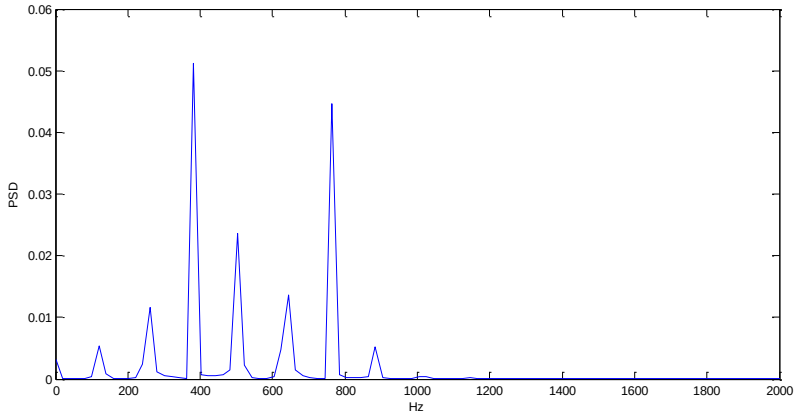


Figure 5.3 Envelope representation of a musical tone used for synthesizing musical instruments.

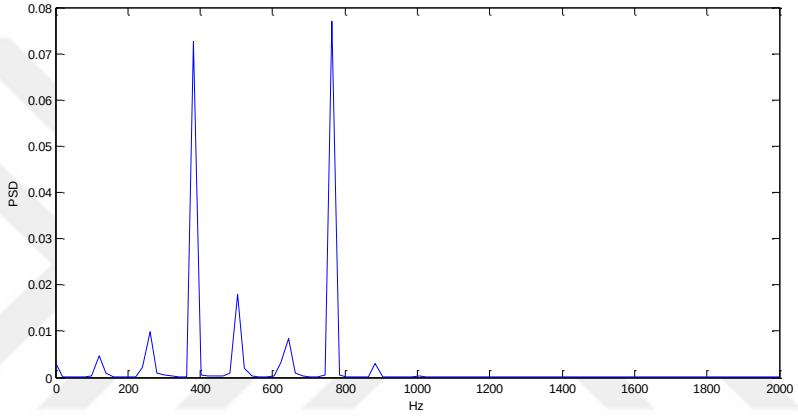
Amplitude of a speech signal can also be divided into four phases as shown in Figure 5.3. In this thesis, the sustain region of speech units was used to extract the LPC parameters to obtain more accurate information about speech units.

5.1.2 Best Filter Order

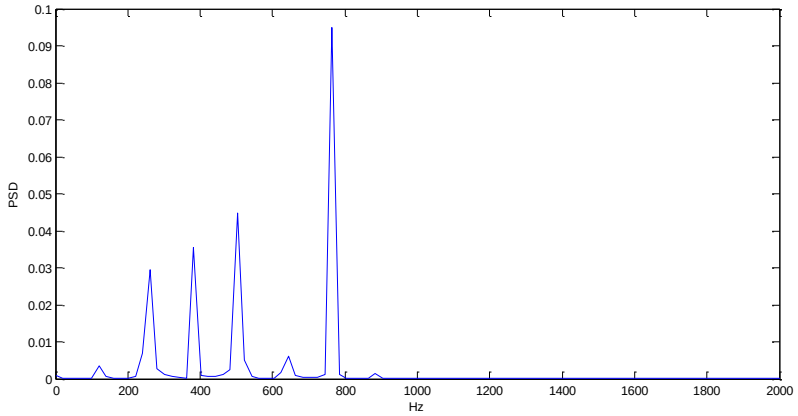
LPC employs a filter that also determines the performance of the resultant speech. Autoregressive (AR) or Autoregressive Moving Average (ARMA) models of IIR filters can be used to determine the second order statistics of input speech data. As for the AR model, which is formed as a denominator polynomial of the transfer function, the filter is commonly used for spectral modelling of the speech signal (Bharitkar & Kryiakakis, 2006). In the all pole filter of LPC, the filter coefficients are found by minimizing an error norm as mentioned in Section 4.2.



(a)



(b)



(c)

Figure 5.4 Power spectral densities (PSDs) of the letter “o” with filter orders (a) 15, (b) 25, and (c) 45.

Advantages of AR model can be listed as follows:

- Provides an excellent spectral representation of the vocal tract for speech signal
- Minimum-phase
- Analytically tractable
- Straightforward to implement in hardware or software
- Works well in all types of speech applications.

Spectral peaks of the sound spectrum are defined as formants. The frequency band of recorded audio speech signals corresponds to the frequency range of 300 Hz to 4 kHz. For intelligibility, formant frequencies should cover all the frequency spectrum of speech (Ballou, 2015). In this thesis work, some trials were performed to obtain the filter order with the best performance as shown in Figure 5.4. After the experiments, use of an LPC filter with order 45 was decided to be employed for all the speech units in the speech database. Figure 5.5 displays the power spectral densities (PSDs) of both the original and synthesized speech waveforms for the letter “o”, respectively. The synthesized waveform was obtained using an LPC all-pole filter of order $p = 45$.

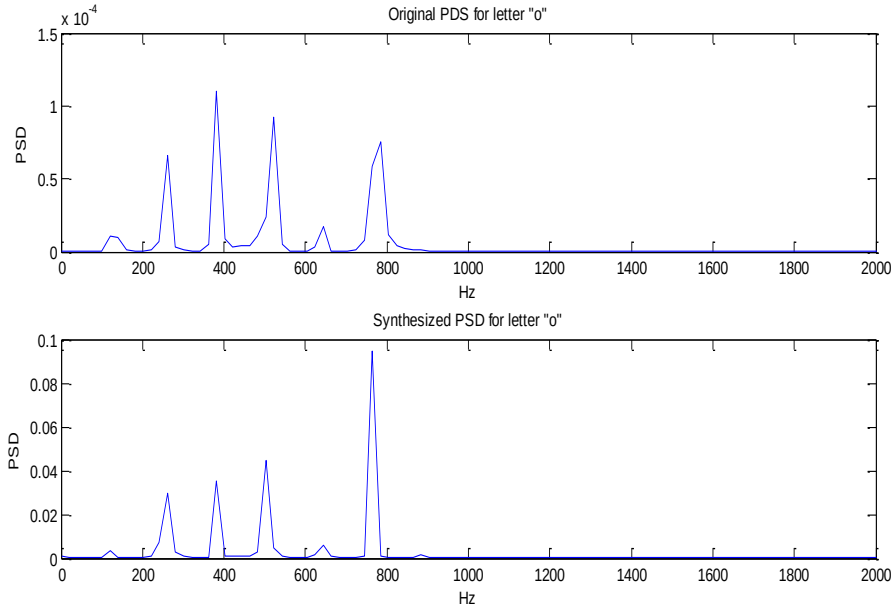


Figure 5.5 Comparison of PSDs of the original and synthesized speech waveforms for the letter “o” using an LPC all-pole filter of order p = 45.

Another important issue about the LPC filter is its stability. To obtain a stable filter, all poles must lie inside the unit circle as also expressed by Equation (5.1) where α and N_p represent the magnitude and the number of poles, respectively. (Krukowski & Kale, 2003). Figure 5.6 represents the positions of the poles on the z-plane for the all-pole filter of letter "o". The LPC filter coefficients belonging to all the speech units in our database also satisfy this stability condition

$$|\alpha_i| < 1 \text{ for } i = 1, .., N. \tag{5.1}$$

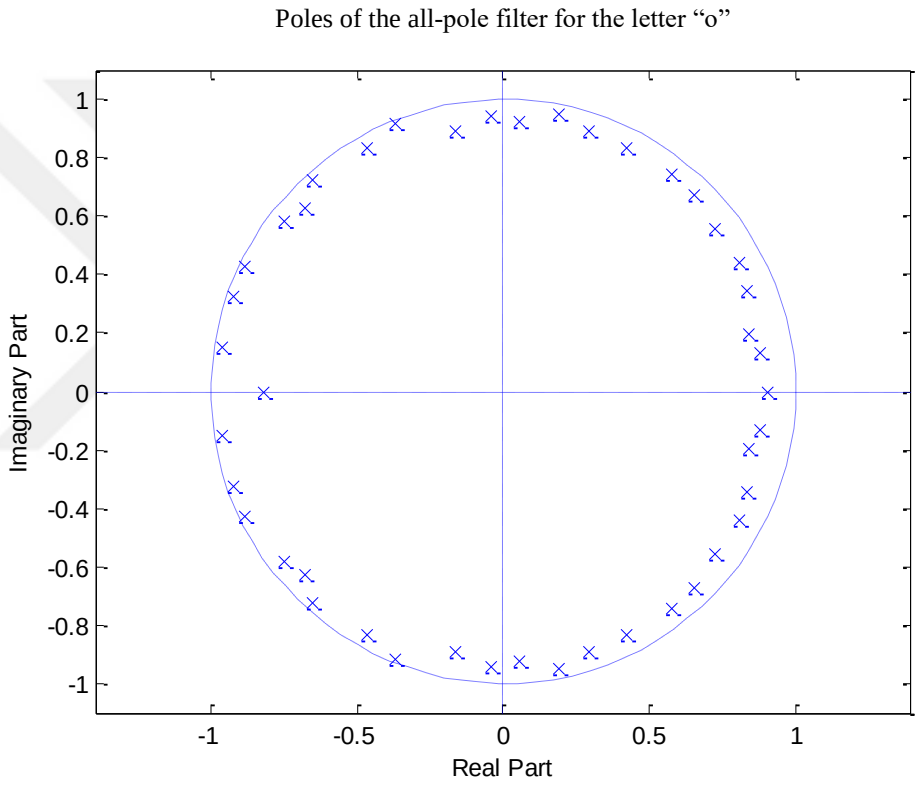


Figure 5.6 Poles of the LPC filter of order 45 for the letter "o". All poles are inside the unit circle indicating a stable filter.

5.1.3 System Sampling Frequency

Determination of the LPC filter order is not independent of the frequency of the analyzed speech. Hence, the frequency content must also be taken into consideration for determining the filter order. In addition, when converting analog signals into digital

form, the sampling rate and the frequency of the obtained digital signal are crucial to obtain high quality speech outputs.

Theoretically, the Nyquist sampling rate which is the minimum sampling rate required to avoid aliasing, is considered for determining the sampling frequency of an analog signal (Shenoi, 2006). Generally, sampling frequency of a standard audio signal is taken as 44.1 kHz. On the other hand, for speech signals, the preferred sampling rate is 22.05 kHz which supports most of the sound cards. That means the frequency content of speech can be as large as 11.025 kHz. Higher frequencies have negligible energy content (Fulop, 2011).

Intelligibility of a speech signal is related to its bandwidth. Intelligibility increases with increasing bandwidth. In classical Public Switched Telephone Networks (PSTN), transmitted speech signals have a frequency content between 300 Hz and 3.4 kHz. Although a lower bandwidth is sufficient, some consonants such as “f” and “s” have higher frequency components up to 14 kHz (Rodman, 2006).

In this thesis, to obtain good quality results, sampling frequencies of 8 kHz and 16 kHz have been tried. Even though the computational cost and system complexity are increased, 16 kHz has been determined as the sampling frequency to obtain better quality results.

5.1.3.1 Bit Quantization

There are various important attributes to pay attention before constructing an LPC synthesizer. Bit rate, delay, complexity, and quality are four of the attributes of a synthesizer. These attributes should be traded off among themselves to design an optimum synthesizer (Bradbury, 2000).

The amount of data that is processed in unit time is called the bit rate or data rate and is measured bits per second (bps) (Mullennix & Stern, 2010). Bit rate affects the quality of the speech output of a synthesizer. High bit rates increase process

complexity and the delay time between the input and the output of the synthesizer. Analog signals are represented using bits in digital domain. Analog signals should be converted into digital form using sufficienty high bit rates to obtain their accurate digital counterparts. Otherwise, a quality loss of a perceptible level might occur.

The LPC method is used in many speech applications like speech coding, speaker recognition, speech enhancement, and speech synthesis. Selecting low bit rates may be sufficient for speech coding applications. The optimal bit rate can be selected depending on the type of application.

Output speech quality, in other words, intelligibility and naturalness of the resultant speech should be of top priority for speech synthesis applications. With the advancement of more complex speech synthesis application environments, using high bit rates has become more common. Because of their improved operation capabilities, common DSP boards allow using high bit rates. Accordingly, in this thesis, some comparison experiments have been performed for obtaining the optimum bit rate of quantized speech samples as either 8 bits or 16 bits. Even if it causes some delays, quantization of speech units with 16 bits was preferred in our applications. The effect of using 16 bits was positively noticeable in the synthesizer output.

5.2 Energy of Speech Units for Joining Speech Samples

In the LPC method, speech is synthesized by means of parameters extracted from the analyzed speech samples. Speech is generated by applying the parameters properly. Any extracted parameter is a factor for enhancing the quality of speech output. The control parameters are pitch period, information of voiced versus unvoiced, predictor coefficients, and the root mean square (RMS) energy value of speech samples (Atal & Hanauer, 1971).

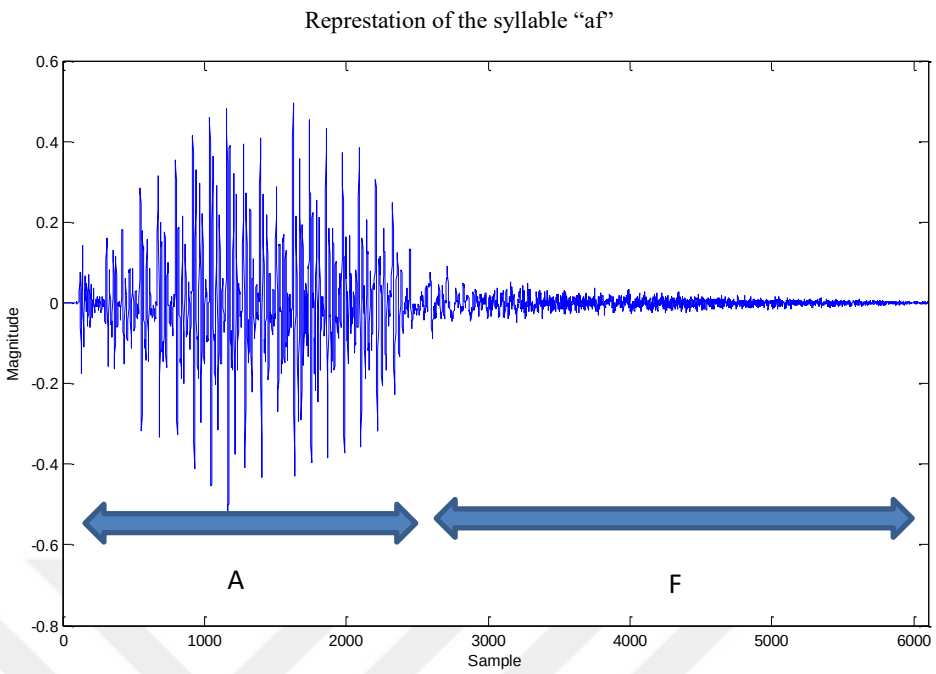


Figure 5.7 Time domain waveform of the syllable “af”. The waveform starts with the voiced letter “a” and ends with the unvoiced letter “f”.

As shown in Figure 5.7, voiced sounds have higher average energy levels and lower frequency values. As opposed to that, unvoiced sounds have less average energy and higher frequency content. Those distinct features are useful for recognizing voiced and unvoiced letters. The energy compatibility between joining letters is an important factor for the quality of the synthesized speech. To overcome the energy incompatibility problem some precautions must be taken. First, the selected speech samples for analysis must be suitable. Speech units to be joined must have the same volume and stress. Second, the gain parameter of the all-pole filter should be arranged by measuring the volume of each synthesized letter.

Let us assume that E_1 and E_2 represent the average energy of two real voiced and unvoiced sounds, $d_1[n]$ and $d_2[n]$, respectively, as shown in Equations (5.2) and (5.3) below

$$E_1 = \frac{1}{K}\sum_{n=1}^{K} d_1^2[n], \tag{5.2}$$

$$E_2 = \frac{1}{L}\sum_{n=1}^{L} d_2^2[n]. \tag{5.3}$$

If γ represents the squared root of the average energy ratio of voiced and unvoiced sounds, then we can write

$$\gamma = \sqrt{\frac{E_1}{E_2}}. \tag{5.4}$$

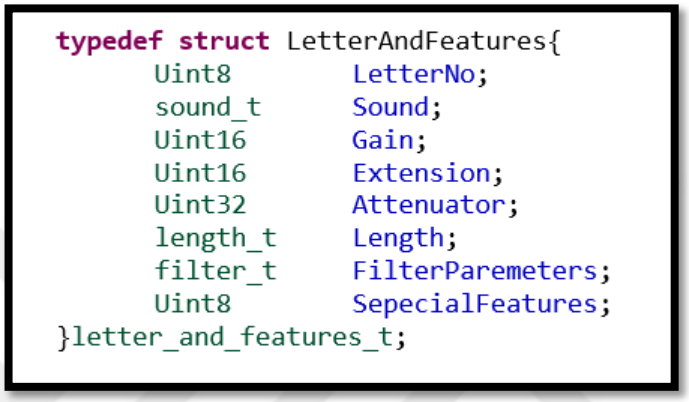
The obtained value of γ is used for changing the value of the gain parameter of the all-pole filter, and thus, helps in arranging the synthesized speech volume of adjoining voiced and unvoiced sound signals.

In this thesis, all the above mentioned issues are taken into consideration. All the speech units were selected so that all had a comparable volume value with each other. The gain parameters were defined for each member of the speech database. Since the DSP board uses fixed-point arithmetic, gain parameters were converted from floating-point into fixed-point with proper proportions. The excitation signal was amplified by a gain value obtained using Equation (5.4) above, before it was applied to the all-pole filter as indicated in Figure 5.1.

The average energy of each adjoining speech unit was calculated using Equations (5.2) and (5.3). Then, the constant γ determining the gain of the all-pole filter was found.

5.3 System Memory Cost

Efficient memory usage and code optimization are both important factors for obtaining high performance from the DSP chip. Performing redundant operations, doing a high number of iterations, and performing multiplications in long format cause latency.



```
typedef struct LetterAndFeatures{
        Uint8          LetterNo;
        sound_t        Sound;
        Uint16         Gain;
        Uint16         Extension;
        Uint32         Attenuator;
        length_t       Length;
        filter_t       FilterParemeters;
        Uint8          SepecialFeatures;
}letter_and_features_t;
```

Figure 5.8 Structure definition of a speech unit.

Figure 5.8 shows the defining parameters of a speech unit in the speech database. These parameters are sufficient for synthesizing a letter from ASCII format to sound wave. "*LetterNo*" defines the number of letters in the speech database. The variable "*Sound"* stores voiced and unvoiced information of letters. The energy of the speech unit is arranged by using the variables "*Gain*", "*Extension*", and "*Attenuator*". Duration of the synthesized speech is assigned via the variable "*Length*". The variable "*FilterParameters*" includes the optimal filter coefficients. Consonant sounds are defined using the variable "*SpecialFeatures*".

Table 5.1 Content and size of the speech database for a speech unit.

Variable Type	**Content**	**Size**
Uint8	• LetterNo	1 Byte
sound_t	• Voiced • Unvoiced • Space	1 Byte
Uint16	• Gain	2 Bytes
Uint16	• Extension	2 Bytes
Uint32	• Attenuator	4 Bytes
length_t	• PitchPeriod • PitchRepeat • Sample	6 Bytes
filter_t	• FilterOrder • FilterCoefficent	91 Bytes
Uint8	• SpecialFeatures	1 Bytes
Total Size		**108 Bytes**

An example for content and size of the speech database is demonstrated for a speech unit in Table 5.1. Data of 108 bytes is stored for converting a letter from ASCII format into a sound signal. Considering that there are 93 members of the speech database, the total memory size is nearly 10 Kbytes.

Beside the size required for the content of the speech database, speech buffers also take up much space in the memory lock. That is because they are needed for synthesizing a sentence or a group of words as a whole. The predicted speech buffer size should be as large as possible. Figure 5.9 demonstrates the memory map of TMS320C5535 DSP board. 64 Kbytes of RAM space is allocated in the code area. In our thesis work, nearly 50 Kbytes of RAM is utilized. Thus, the development platform offers enough code space for our synthesis work. DSP board has 256 Kbytes of memory for code development. Our thesis work took up only one quarter of the whole code memory space of the DSP board.

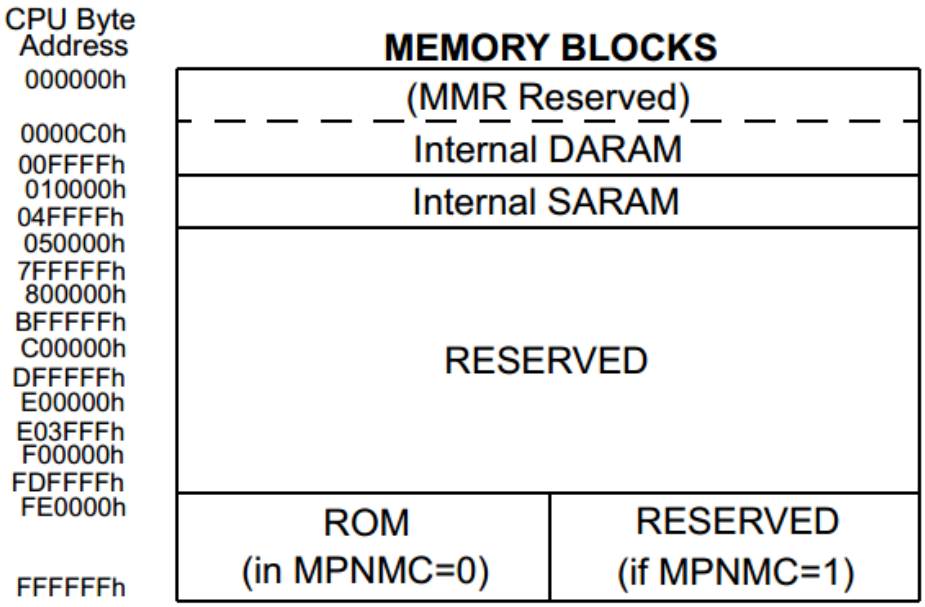


Figure 5.9 Memory map of the TMS320C5535 DSP board.

Having sufficient code and RAM spaces is important to be able to design a high quality synthesizer. Large RAM space paves the way for using high bit rate and quantized high frequency data. This directly impacts the quality of the synthesized speech. In addition, large RAM space allows synthesis of a large number of characters into a single command uninterruptedly. Having all the LPC parameters in the RAM space allows reaching the variables more quickly.

In this thesis work, the speech buffer accumulates nearly 20 tokenized text characters. Then, AIC3204 audio codec operates on the content of the buffer as a mono sound wave.

5.4 System Running Frequency

DSP chips are processors optimized for signal processing applications. Most of the DSP chips have special instruction sets and built-in hardware modules for the multiply and accumulate operations. TMS320C5535 DSP board has two multiply-accumulate (MAC) units each capable of 17-bit X 17-bit multiplication in a single cycle. Additionaly, DSP chips have an arhitecture based on multiple data input and output

buses (Stranneby & Walker, 2004). TMS320C5535 evaluation board has three data read buses and two data write buses. Data read and write buses provide the ability of performing three data reads and two data writes in a single cycle.

Operating frequency is one of the performance criteria to obtain the maximum computational and operating performance from the DSP board. Increasing the system frequency also increases the power consumption of the system. System frequency can even be traded off against the system performance (Piguet, 2006). Power characterization of the DSP chip is defined on average as 0.22 mW/MHz at 1.3 V core voltage and 100 MHz operating frequency.

Digital signal processor was driven with 100 MHz that is the highest frequency value supported by the DSP chip to obtain low latency between the input text and the output speech. This frequency is generated by a built-in crystal using the method of phase-locked loop (PLL).

5.5 Speech Quality and Evaluation

Quantifying synthesizer quality by making objective evaluation of synthesized speech is an important matter. Basic evaluation and comparison criteria of synthesizer outputs are intelligibility, naturalness, and suitability for a particular application. Figure 5.10 summarizes some of the TTS evaluation techniques (Klatt, 1987).

As shown in Figure 5.10, there are quite many evaluation tests. Most of the researchers who are interested in speech applications complain that there are many existing evaluation methods. For this reason, making a comparison is difficult throughout all TTS applications (Lemmetty, 2006). On the other hand, differences amoung the used TTS application languages cause a lack of standardization. This is because distinct languages have different grammatical rules.

Synthesizers are generally designed for a specific goal. Before evaluating the performance of a synthesizer, it is important to know its design purpose. For example,

for an address reader application, a rather low system performance may be deemed sufficient by the user. On the other hand, for multimedia applications, high quality is expected.

Evaluation methods are usually based on subjective listening tests in response to a set of syllables, words, sentences, or other questions. The sought answers are related to intelligibility, naturalness, or other features of speech. The synthesized speech is graded by a listener to form an opinion about the overall synthesizer quality.

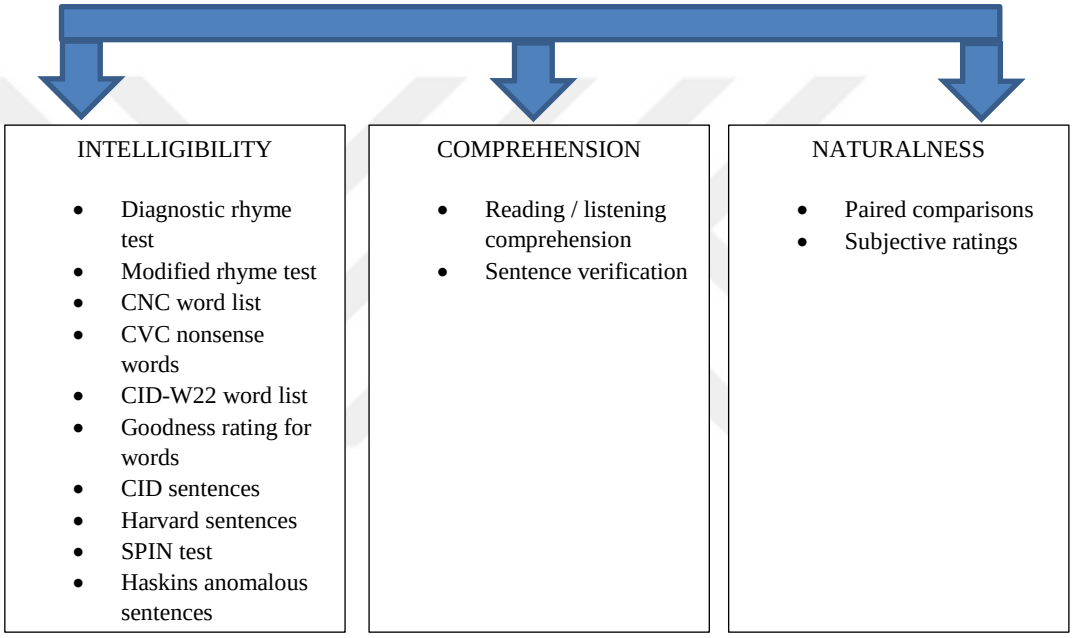


Figure 5.10 Techniques for evaluating TTS system performance.

5.5.1 Intelligibility Tests

In a synthetic speech system, there are three different types of error that may affect intelligibility of speech. Those are incorrect spelling-to-sound rules, computation and production of incorrect or inappropriate suprasegmental information, and use of error-prone phonetic implementation of allophones into a speech waveform (Pisoni, 1997).

The intelligibility test can be applied on a single segment as phoneme intelligibility or as intelligibility of a whole sentence. Intelligibility of synthetic speech is measured simply by the number of correctly identified words compared to all words. Diagnostic information can be given by confusion matrices which give information on how different phonemes are misidentified and help to localize the problematic points for development. The employed method and application language can render selection of a testing method difficult. Hence, a large number of test methods have been developed. Some of those test methods are briefly explained below.

5.5.1.1 Diagnostic Rhyme Test (DRT)

Monosyllabic words are used for diagnostic rhyme test. The test syllables are constructed from consonant-vowel-consonant sound sequence. 96 word pairs that have a single acoustic difference with each other are used in the test. Six phonetic characteristic features that are given in Table 5.2 are evaluated by the listener and the result is generated in the form of percentage error (Limmetry, 2006).

Table 5.2 Phonetic characteristics of diagnostic rhyme test.

Characteristic	**Description**	**Examples**
Voicing	voiced-unvoiced	veal-feel, dense-tense
Nasality	nasal-oral	reed-deed
Sustension	sustained-interrupted	vee-bee, sheat-cheat
Sibilation	sibilated-unsibilated	sing-thing
Graveness	grave-acute	weed-reed
Compactness	compact-diffuse	show-sow

Most of the test procedure is planned for and hence suitable to pronunciation of English and different from the Turkish language. Still, examples can also be translated into Turkish language. For example, the monosyllabic words of “nal” and “sal” are of voiced and unvoiced character such that “n” is a voiced letter and “s” is unvoiced.

5.5.1.2 Modified Rhyme Test (MRT)

Modified rhyme test is similar to diagnostic rhyme test. It uses 50 word sets. Each set has 6 different words which are constructed from a consonant-vowel-consonant sound sequence. The test focuses on initial and final consonants of six different words. The results are evaluated by listening to the word sets similar to DRT. Table 5.3 gives some MRT samples.

Table 5.3 Phonetic characteristics of modified rhyme test.

	A	**B**	**C**	**D**	**E**	**F**
1	went	sent	bent	dent	tent	rent
2	holt	told	cold	fold	sold	gold
3	pat	pad	pan	path	pack	pass
.	.	.	.	.	.	.
.	.	.	.	.	.	.
49	bun	bus	but	bug	buck	buff
50	fun	sun	bun	gun	run	nun

5.5.1.3 Diagnostic Medial Consonant Test (DMCT)

Diagnostic medial consonant test is also similar to diagnostic rhyme test. The test is made up of 96 two syllable word pairs. The only difference between the word pairs is the middle consonant. For example: bobble-bottle, stopper-stocker. The difference is classified into six categories as in DRT. The aim of the test is to choose the correct words from two possible alternatives. The test provides an overall measure of system intelligibility.

5.5.1.4 Harvard Psychoacoustic Sentences

Harvard psychoacoustic sentences are widely used in research for synthetic speech intelligibility in the sentence context. They consist of 100 sentences that are chosen from various segmental phonemes represented in accordance with their frequency of occurrence. Some text sentences are given as follows (Harvard sentences, (n.d.)):

- The birch canoe slid on the smooth planks.
- Glue the sheet to the dark blue background.
- It's easy to tell the depth of a well.
- These days a chicken leg is a rare dish.
- Rice is often served in round bowls.
- The juice of lemons makes fine punch.
- The box was thrown beside the parked truck.

5.5.1.5 Haskins Sentences

Haskins sentences are used for testing intelligibility just like Harvard sentences. One difference is that the sentences are put together with anomalous monosyllable words. As a result, they are meaningless. Some of the Haskins Sentences are given as follows. Fixed sentences give reliable test results when comparing synthesizer quality.

- The wrong shot led the farm.
- The black top ran the spring.
- The great car met the milk.
- The big bank felt the bag.
- The red shop said the yard.
- The full leg shut the score.
- The first car stood the ice.

5.5.1.6 Semantically Unpredictable Sentences (SUS)

Another sentence level intelligibility test is semantically unpredictable sentence test. The sentences are constructed randomly with pre-defined words according to the five grammatical structures described in Table 5.4 (Jekosch, 1993). They are different from Haskins Sentences. The test sentences are changed at every sentence generation attempt. Because of this, SUS test is not as sensible to the previously given sentence contrary to other test methods (Jekosch, 2005).

Table 5.4 Structures of semantically unpredictable sentences.

No	Structure	Example
1	Subject-verb-adverbial	The table walked through the blue truth.
2	Subject-verb-direct object	The strong way drank the day.
3	Adverbial-verb-direct object	Newer draw the house and the fact.
4	Q-word-transitive verb-subject-direct object	How does the day love the bright word?
5	Subject-verb-complex direct object	The plane closed the fish that lived.

5.5.2 Comprehension Test

As opposed to phonemes and single words, comprehensive test is applied on a few sentences or a paragraph and listeners try to answer the questions about the text. It is not important that a word or phoneme % 100 intelligible as long as the meaning of the sentence is understood (Bernstein, & Pisoni, 1980).

5.5.3 Naturalness Tests

Naturalness of synthetic speech can be defined as being similar to human speech. It is often related to the presence of various types of distortions or artifacts in the synthetic speech such as noise, echoes, muffling, and clicking (Chu, 2003). The most common approach for measuring naturalness of synthetic speech is to let some listeners to listen to synthetic speech and ask them to rate what they hear.

Sometimes the naturalness test is used for investigating overall system performance. For this, a few methods have also been developed to evaluate the quality of synthetic speech. These are also the most preferred methods for measuring quality of Turkish TTS systems.

5.5.3.1 Absolute Category Rating (ACR)

A common subjective benchmark for assessing the performance of a speech synthesizer is the absolute category rating method. It is also named as the mean opinion score (MOS) method. MOS test is applied to a group of listeners. It is important to apply the test on a wide range of people to obtain more accurate results (Davidson, Peters & Gracely, 2000).

Table 5.5 MOS test grading.

Points	**Assessments**
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Table 5.5 shows assessment criteria of MOS test and the related grading points. There are five different gradation scales. Listeners grade the synthetic speech according to Table 5.5. Calculation of MOS is realized by averaging the grades of listeners as given in Equation (5.5) where L denotes the number of listeners and s_l is the score assigned by the l^{th} listener,

$$\text{MOS} = \frac{1}{L}\sum_{l=1}^{L} s_l. \tag{5.5}$$

5.5.3.2 Degradation Category Rating (DCR)

Degradation category rating test is similar to MOS test in terms of grading; however, it differs with respect to execution. Each test case involves two samples; the first sample is the original speech sample and the second is its synthesized version (Davis, 2002). Listeners listen to the first sample as a reference before the synthetic speech. Then, they compare the two samples and give a rating according to the amount of degradation perceived. The choices are graded according to the assessment points in Table 5.6 below.

Table 5.6 Degradation category ratings.

Points	**Assessments**
5	Inaudible
4	Audible, but not annoying
3	Slightly annoying
2	Annoying
1	Very annoying

5.5.3.3 Comparison Category Rating (CCR)

Comparison category rating test is another measuring method for investigating overall synthetic speech quality. One of the differences with the DCR test is the hidden reference and the synthesized signal. The speech sample pairs are listened arbitrarily in random order. The elements of the pair can be designated as A and B, respectively. The final result is found by making a comparison between the two signals. Grading is made according to Table 5.7 (Bech, & Zacharov, 2006).

Table 5.7 CCR test grading.

Points	**Assessments**
3	A is much better than B
2	A is better than B
1	A is slightly better than B
0	A is the same as B
-1	B is slightly better than A
-2	B is better than A
-3	B is much better than A

5.5.4 Overall Quality of Synthesizer

The simplest performance criterion for a TTS system is being as close to human speech as possible. For this, naturalness and intelligibility should be at nearly the optimum levels. These two parameters are very important for determining the synthesizer quality. There are many subjective methods for assessing quality of synthetic speech as mentioned in the previous sections. MOS method is commonly

used for determining synthesizer performance. This method is referred to in P.800.1 ITI-I Recommendation (Mean opinion score (MOS) terminology, (n.d.)).

Table 5.8 Turkish sample sentences for MOS test.

Sentence Id	**Sentence**
S1	İzin almanız gerekir.
S2	Gelinen nokta aynı.
S3	Gideceğiniz yer neresi?
S4	Baştan sona kadar.
S5	Ana kapıdan geçti.
S6	Test edilecek.
S7	Artık bizimle oturacak.
S8	İlk defa kabul etti.
S9	Hangi yoldan geçti?
S10	Geçiş için hızlan.

Table 5.8 shows sample test sentences that are used in MOS test for the Turkish language. The test is applied on ten different listeners with ten different sentences. After the synthesized sentences are played to listeners only once, it is asked from them to grade their assessment.

Table 5.9 MOS test ratings.

Sentence Id	**Grading**					
	Excellent	**Good**	**Fair**	**Poor**	**Bad**	**Average**
S1	-	1	5	4	-	2.7
S2	-	-	5	5	-	2.5
S3	-	-	6	4	-	2.6
S4	-	1	4	5	-	2.6
S5	-	2	5	3	-	2.9
S6	-	3	4	3	-	3
S7	-	1	5	4	-	2.7
S8	-	-	5	5	-	2.5
S9	-	-	6	4	-	2.6
S10	-	-	5	4	1	2.4
Sum	**0**	**8**	**50**	**41**	**1**	**2.65**

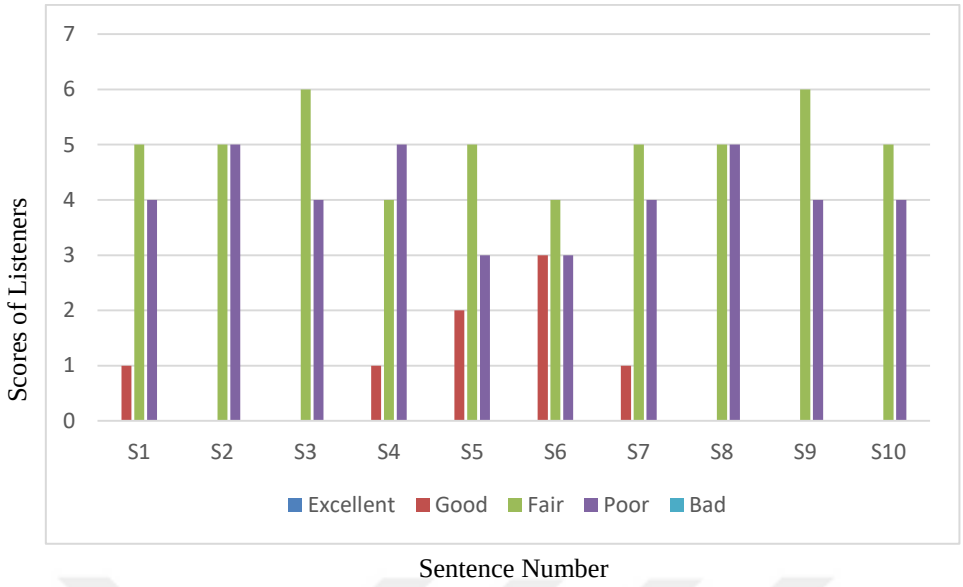


Figure 5.11 MOS test rating graph.

Table 5.9 displays the scores of all the sentences numerically and Figure 5.11 represents them graphically. The assessment measures were given in Table 5.6. The average test results for all the sentences are drawn in Figure 5.12.

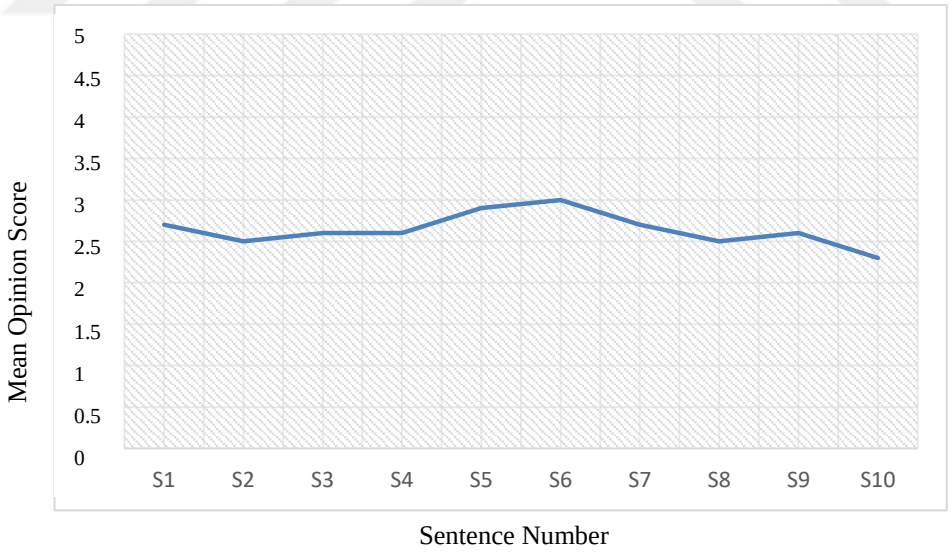


Figure 5.12 Average MOS test results.

A limited number of academical research studies have been performed in the area of Turkish TTS synthesis. Unfortunately, the outcomes of those studies are not suitable for a performance comparison. In fact, there exist no performance evaluations in the

studies which perform Turkish TTS synthesis via the LPC method. Among the studies employing concatenative method, Görmez (Görmez, 2009) reached a MOS rank of 3.42 and Erkan (Erkan, 2014) obtained a MOS rank of 3.72. Considering these results and computational advantages of the LPC method, we believe the obtained MOS rank of 2.65 in this thesis is acceptable.

5.5.5 Suitability for a Particular Application

This thesis work can be divided into two parts; one is the analysis of the input text and the second is the synthesis part. The LPC parameters are stored in the DSP storage field statically. The constant parameters are called from the storage and start forming the synthetic speech. The core of the design work can be migrated to another platform by moving the LPC parameters easily as long as the new platform fulfills synthesizer hardware requirements.

CHAPTER SIX
CONCLUSION

Natural language processing has become a more and more popular application area with the development of technology. Technological capabilities have paved the way for easier implementation of speech applications on electronic devices. It has become possible to fit entire human speech functionality into a single electronic device.

Following the development of speech synthesis, various applications have also come into existence. Some of those applications help disabled people. They also facilitate our daily life via applications such as telephone banking, voice information systems, and learning languages. The developed devices can be used standalone without any database connection.

Languages spoken in the world belong to different origins. This fact brings about different rules in language processing and speech production. Therefore, any designed speech synthesizer must be language-specific. Accordingly, linguistic rules are effective in the design of speech synthesizers. Heeding linguistic rules is a factor which increases the quality of the synthesizer.

Although the first emerged speech synthesis systems were designed mechanically, nowadays formant synthesis and concatenative synthesis techniques are used extensively. There could be significant advantages and disadvantages of any preferred method. More natural sounding speech can be obtained using concatenative methods though discontinuities between two concatenated speech samples can be problematic. Some intonation techniques could overcome this discontinuity problem. On the other hand, more intelligible speech can be produced by using formant synthesis methods even though the produced speech could sound unnatural and robotic. Formant synthesis methods are modifiable allowing control of the fundamental frequency. One of the most popular speech coding methods is called the LPC method that can also be used in speech synthesis applications. The LPC method can be classified under the formant synthesis methods.

Composing the database is a major issue to be considered since there must be an appropriate database accompanying any suitable synthesizer. To minimize any possible disharmony, each member of the database should be chosen carefully. Speech samples must not be affected from environmental noise or background sounds. Considering that hundreds or even thousands of database members could be in a synthesizer, it is hard to fulfill the database quality requirements for each member without professional recording media.

In this thesis, the LPC method has been used for speech synthesis in the Turkish language. The aim of using the LPC method is to obtain a Turkish TTS synthesizer that can fulfill small memory requirements, be realized by embedded systems, and satisfy quality conditions. The LPC parameters and database content are usually smaller than the database of concatenation method. On the other hand, theoretically, all the words in a selected language can be generated with a limited number of LPC parameters. This thesis work allows conversion of the entered text into a speech signal. The existence of the audio codec in the DSP board is an advantage. Besides that, since the software of the DSP platform is based on C programming language, it can be transported into another hardware platform.

One expects the synthesizer to produce more understandable and natural sounding speech as much as possible. One technique for this could be recording the original speech samples as whole words rather than generating sentences by joining recorded speech units. Even if this approach could be a good solution for closed-circuit systems, it is too costly to offer wide range uses. Synthesized speech via the LPC method is less natural sounding although more intelligible than concatenation method. This fact was also observed in the results of this thesis work.

Turkish is a language that is read as written. Because of that speech is synthesized by combining letters. Speech parameters obtained by analyzing the actual speech signal are again turned into a speech signal via an inverse operation. Letters are selected by parsing syllables. Letters forming a word are joined by also considering

the letters which are adjacent to them. This was shown to improve the quality of the synthesizer.

Quality of a TTS application is a relative concept that varies from person to person. In general, speech synthesizers are assessed by listening to the generated sounds which are also scored by an audience. When this test method is applied on our project, the obtained results were deemed of medium quality. This is an expected result for the LPC method. There is a trade-off between the size and quality of a synthesizer system. An optimum balance between them should be struck in accordance with the requirements of the underlying application.

Several methods can be applied to improve the system. Further linguistic rules can be included in sound generation. Sound quality can be enhanced by using less noisy recorded speech samples for analysis of LPC parameters. Even further improvement can be made by adding emphasis and intonation to speech. Scope of the synthesizer can be enlarged via further expanded text analysis.

Despite the fact that the history of TTS has more than a hundred years, studies on synthesis of Turkish are quite recent. Academic studies have come into existence since the 1990s. Compared to other languages, there are only a few professional works. Language is a constantly evolving phenomenon and quality is relative. Accordingly, studies on speech synthesis and natural language processing are expected to continue growing in numbers in the near future.

REFERENCES

Adalı, E. (2012). *Sesbilim*. Retrieved May 24, 2016, from www.adalı.net/wp content/uploads/2012/10/DDI-Kitap-Sesbilim.pdf.

Atal B., & Hanauer, S. L. (1971). Speech analysis and synthesis by linear prediction of the speech wave. *The Journal of Acoustic Society of America*, 637 - 655.

Ballou, G. (2015). *Handbook for sound engineers*. New York: Focal Press.

Bech, S., & Zacharov, N. (2006). *Perceptual audio evaluation: Theory, method, and application*. England: John Wiley & Sons.

Bernstein J., & Pisoni D. (1980). Unlimited text-to-speech system: Description and evaluation of a microprocessor based device. *Proceedings of ICASSP*, 80 (3): 574 579.

Bharitkar, S., & Kryiakakis, C. (2006). *Immersive audio signal processing*. New York: Springer.

Biometrics speech production. (n.d.). Retrived May 14, 2016, from http://www.barcode.ro/tutorials/biometrics/img/speech-production.jpg.

Bradbury, J. (2000). *Linear predictive coding*. Retrieved June 25, 2016, from my.fit.edu/~vkepuska/ece5525/lpc_paper.pdf.

Caruntu, A., & Toderean, G., & Nica, A. (2005). Automatic silence/unvoiced/voiced classification of speech using a modified Teager energy feature. *WSEAS International Conference on Dynamical Systems and Control*, Venice, Italy, November 2-4, 2005, 62-65.

Casabona, H., & Frederick, D. (1987). *Beginning synthesizer: A volume in the keyboard magazine library for electronic musicians*. Sherman Oaks: Alfred.

Chu, C. W. (2003). *Speech coding algorithms: Foundation and evolution of standardized coders*. New Jersey: John Wiley & Sons, Inc.

Copper, S. F., Liberman, M. A. & Borst, M. J. (1951). The interconversion of audible and visible patterns as a basis for research in the perception of speech. *Proceedings of the National Academy of the Sciences of the America*, 37, 318 - 325.

David, G. (2003). *Pitch extraction and fundamental frequency: History and current techniques* (Tech. No. TR-CS 2003-06). Canada, Saskatchewan: Department of Computer Science, University of Regina.

Davidson, J., Peters, J., & Gracely, B. (2000). *Voice over IP fundamentals*. Indianapolis: Cisco Press.

Davis, G. M. (2002). *Noise reduction in speech applications*. Boca Raton: CRC Press.

Delibaş, A. (2008). *Doğal dil işleme ile Türkçe yazım hatalarının denetlenmesi*. M.Sc. Thesis, İstanbul Teknik Üniversitesi, İstanbul.

Dudley, H. (1939). The Vocoder. *Bell Labs Record*, 17, 122-126.

Dudley, H., Riesz R. R., Watkins, S. A. (1939). A synthetic speaker. *Journal of the Franklin Institue*, 227, 739-764.

Erkan, E. (2014). *Türkçe metinler için konuşma motoru geliştirilmesi.* M.Sc. Thesis, Karabük Üniversitesi, Karabük.

Flanagan, J. (1972). *Speech analysis, synthesis, and perception*. Heidelberg: Springer.

Fulop, S. A. (2011). *Speech spectrum analysis*. Berlin: Springer.

Görmez, Z. (2009). *Implementation of a text-to-speech system with machine learning algorithms in Turkish.* M.Sc. Thesis, Fatih University, İstanbul.

Harvard Sentences. (n.d.). Retrieved July 09, 2016, from http://www.cs.columbia.edu/~hgs/audio/harvard.html

Haskins Laboratories, *Stewart*. (n.d). Retrived April, 3, 2016, from http://www.haskins.yale.edu/featured/heads/SIMULACRA/stewart.html.

Jekosch U. (1993). Speech quality assessment and evaluation. *Proceedings of Eurospeech*, 93 (2): 1387-1394.

Jekosch, U. (2005). *Voice and speech quality perception: Assessment and evaluation*. Berlin: Springer.

Klatt, D. (1980). Software for a cascade/parallel formant synthesizer. *Journal of the Acoustical Society of America, JASA*, 67: 971-995.

Klatt, D. (1987). Review of text-to-speech conversion for English. *Journal of the Acoustical Society of America, JASA*, 82 (3), 737-793.

Krukowski, A., & Kale, I. (2003). *DSP system design: Complexity reduced IIR filter implementation for practical applications*. Boston: Kluwer Academic.

Kuru, S. & Akın, H. L. (1992). Spelling checking in Turkish. *DECSYM'92 Latest Trends in Computing* (195-204), Ankara.

Lemmetty, S (1999). *Review of speech synthesis technology*. M.Sc. Thesis, Helsinki University of Technology, Helsinki.

Levinson, S., Davis, D., Slimon, S. & Huang, J. (2012). *Articulatory speech synthesis from the fluid dynamics of the vocal apparatus*. Williston: Morgan & Claypool Publishers.

Marschall, B. P. (2005). Remarks on the history of articulatory-acoustic modelling. *ZAS Papers in Linguistics*, 40, 145-159.

Mullennix, J. W., & Stern, S. (2010). *Computer synthesized speech technologies: Tools for aiding impairment*. Hershey, PA: Medical Information Science Reference.

P.800.1: Mean Opinion Score (MOS) terminology. (n.d.). Retrieved July 12, 2016, from http://www.itu.int/rec/T-REC-P.800.1-200303-S.

Piguet, C. (2006). *Low-power processors and systems on chips*. Boca Raton, FL: CRC Press/Taylor & Francis.

Pisoni, D. B. (1997). *Perception of synthetic speech*. Santen, J. P., Sproad, R. W., Olive, J. P., & Hirschberg, J., (Ed). *Progress in speech synthesis* (541-560). New York: Springer.

Rashad, M. Z., Hazem, M. & Mastorakis, N. (2010). *An overwiev of text-to-speech synthesis tecniques.* N. Mastorakis & V. Mladenov & Z. Bojkovic, (Ed.). *Latest Trends of Communication and Information Technology (84-89)*. Corfu Island: WSEAS Press.

Rodman, J. (2006). *The effect of bandwidth on speech intelligibility*. Retrieved June 19, 2016, from http://docs.polycom.com/global/documents/whitepapers/effect_of_bandwidth_on speech_intelligibility_2.pdf.

Rossing, T. D., Moore, F. R, & Wheeler, P. A. (2002). *The science of sound* (3rd ed.). Boston: Addison Wesley.

Salomon, D. (2007). *Data compression: The complete reference* (4th ed.). London: Springer.

Schroeder, M. R. (1972). *Computer speech recognition, compression, synthesis* (2nd ed.). Berlin: Springer.

Shenoi, B. A. (2006). *Introduction to digital signal processing and filter design*. Hoboken, New Jersey: Wiley-Interscience.

Stranneby, D., & Walker, W. (2004). *Digital signal processing and applications* (2nd ed.). Oxford: Newnes.

Tatham, M., & Morton, K. (2005). *Developments in speech synhesis*. Chichester: John Wiley & Sons Ltd.

Taylor, P. (2009). *Text-to-speech synthesis*. Cambridge: Cambridge University Press.

TMS320C5535 eZdsp technical reference. (n.d.). Retrieved May 7, 2016, from http://support.spectrumdigital.com/boards/ezdsp5535/revc/files/ezdsp5535_TechRef_RevC.pdf.

Türk Dil Kurumu. (n.d.). Retrieved June 04, 2016, from http://tdk.gov.tr/index.php?option=com_content.

APPENDICES

APPENDIX 1: Features of TMS320C5535 Development Kit.

TMS320C5535 eZdsp™ USB Development Kit

TEXAS INSTRUMENTS

Key features and benefits

- Small form factor DSP development kit for the C5535 processor
- TMS320C5535 fixed-point ultra-low-power DSP
- Embedded XDS100 emulator
- 8-MB serial Flash memory
- TLV320AIC3204 programmable low-power stereo audio codec
- USB 2.0 high speed
- Micro SD card slot with 2-GB micro SD card
- Line in/Mic in, headphone out audio jacks
- Earphone with mic
- 60-pin expansion connector
- 96 × 16-pixel OLED display
- Two push buttons
- Includes Code Composer Studio™ IDE 4.0
- Software framework for USB audio class and HID applications
- Out-of-the-box demo software
- Full documentation with source code on CD-ROM

The TMDX5535eZdsp is a small form factor, very-low-cost USB-powered DSP development kit which includes hardware and software needed to evaluate the C553x generation, which is the industry's lowest-cost and lowest-power 16-bit DSP. This ultra-low-cost kit allows quick and easy evaluation of the advanced capabilities of the C5532, C5533, C5534 and C5535 processors. The kit has an on-board XDS100 emulator for full source-level debug capability and supports Code Composer Studio™ (CCStudio) Integrated Development Environment (IDE) version 4.0 and eXpressDSP™ software which includes the DSP/BIOS™ kernel. The full contents of the Development Kit include C5535 eZdsp board, CCStudio IDE Rev. 4.0, a headphone with mic, a 2-GB micro SD card, a free software framework for USB audio class and human interface device (HID) applications and an out-of-the-box comprehensive demo for USB audio class applications.

Technical details

The C5535 eZdsp USB kit simplifies development by providing integrated features including:

- Complete Code Composer Studio v4 IDE for fast code development
- On-board XDS100 v2 emulator provides complete debugging capabilities and visibility inside the processor for algorithm optimization and benchmarking
- On-board audio codec and connectors allow developers to evaluate the C5535 processor and quickly optimize complex DSP algorithms in terms of performance and power consumption across a variety of design scenarios
- Energy-efficient C5535 DSP allows the entire development tool to be powered by the USB port — no other components or cables are needed
- Rich set of features including LCD 96 × 16 monochrome OLED display screen, MicroSD card slot, USB 2.0 port for applications, *Bluetooth*®/Chipcon expansion connector

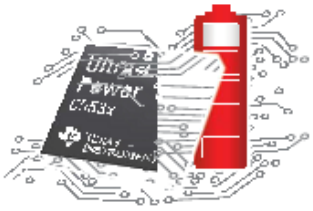
Software

Designers can readily target the TMS320C5532/33/34/35 DSP through TI's robust and comprehensive Code Composer Studio IDE, including:

- A complete Integrated Development Environment, an efficient optimizing C/C++ compiler assembler, linker, debugger, integrated CodeWright editor with CodeSense technology for faster code creation, data visualization, a profiler and a flexible project manager
- DSP/BIOS™ real-time kernel
- Chip Support Library
- Free, integrated software framework for USB audio class and HID applications, including an out-of-the-box demo

Community support

The eZdsp USB kit is supported by TI's online community **e2e.ti.com**. Complete collateral, CCStudio IDE drivers, Chip Support Library (CSL) and all the required production-quality



APPENDIX 2: Features of TMSC5535 Digital Signal Processor.

TMS320C5535, 'C5534, 'C5533, 'C5532 Fixed-Point Digital Signal Processors

Check for Samples: TMS320C5535, TMS320C5534, TMS320C5533, TMS320C5532

1 Fixed-Point Digital Signal Processor

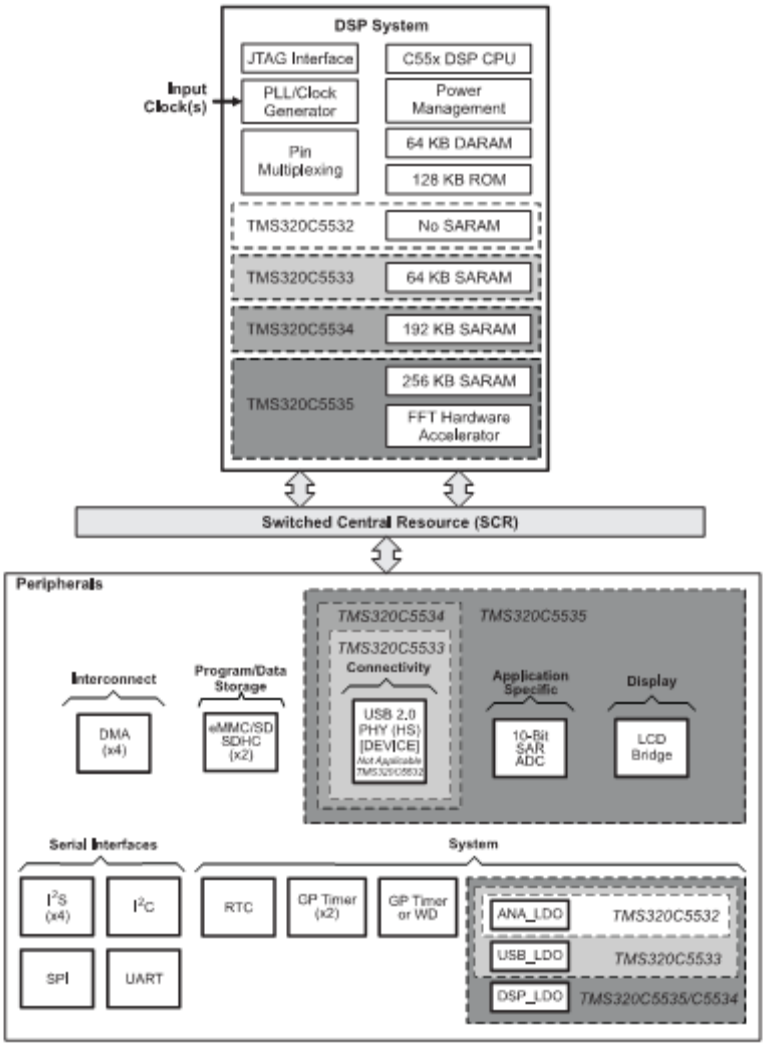
1.1 Features

- CORE:
  - High-Performance, Low-Power, TMS320C55x Fixed-Point Digital Signal Processor
    - 20-, 10-ns Instruction Cycle Time
    - 50-, 100-MHz Clock Rate
    - One/Two Instruction(s) Executed per Cycle
    - Dual Multipliers [Up to 200 Million Multiply-Accumulates per Second (MMACS)]
    - Two Arithmetic/Logic Units (ALUs)
    - Three Internal Data/Operand Read Buses and Two Internal Data/Operand Write Buses
    - Software-Compatible With C55x Devices
    - Industrial Temperature Devices Available
  - 320K Bytes Zero-Wait State On-Chip RAM, Composed of:
    - 64K Bytes of Dual-Access RAM (DARAM), 8 Blocks of 4K x 16-Bit
    - 256K Bytes of Single-Access RAM (SARAM), 32 Blocks of 4K x 16-Bit
  - 128K Bytes of Zero Wait-State On-Chip ROM (4 Blocks of 16K x 16-Bit)
  - Tightly-Coupled FFT Hardware Accelerator
- PERIPHERAL:
  - Direct Memory Access (DMA) Controller
    - Four DMA With 4 Channels Each (16-Channels Total)
  - Three 32-Bit General-Purpose Timers
    - One Selectable as a Watchdog and/or GP
  - Two Embedded Multimedia Card/Secure Digital (eMMC/SD) Interfaces
  - Universal Asynchronous Receiver/Transmitter (UART)
  - Serial-Port Interface (SPI) With Four Chip-Selects
  - Master/Slave Inter-Integrated Circuit (I^2C Bus™)
  - Four Inter-IC Sound (I^2S Bus™) for Data Transport
  - Device USB Port With Integrated 2.0 High-Speed PHY that Supports:
    - USB 2.0 Full- and High-Speed Device
  - LCD Bridge With Asynchronous Interface
  - 10-Bit 4-Input Successive Approximation (SAR) ADC
  - IEEE-1149.1 (JTAG™) Boundary-Scan-Compatible
  - Up to 20 General-Purpose I/O (GPIO) Pins (Multiplexed With Other Device Functions)
- POWER:
  - Four Core Isolated Power Supply Domains: Analog, RTC, CPU and Peripherals, and USB
  - Three I/O Isolated Power Supply Domains: RTC I/O, USB PHY, and DV_{DDIO}
  - Three integrated LDOs (DSP_LDO, ANA_LDO, and USB_LDO) to power the isolated domains: DSP Core, Analog, and USB Core, respectively
  - 1.05-V Core (50 MHz), 1.8-V, 2.5-V, 2.75-V, or 3.3-V I/Os
  - 1.3-V Core (100 MHz), 1.8-V, 2.5-V, 2.75-V, or 3.3-V I/Os
- CLOCK:
  - Real-Time Clock (RTC) With Crystal Input, With Separate Clock Domain, Separate Power Supply
  - Low-Power S/W Programmable Phase-Locked Loop (PLL) Clock Generator
- BOOTLOADER:
  - On-Chip ROM Bootloader (RBL) to Boot From SPI EEPROM, SPI Serial Flash or I2C EEPROM eMMC/SD/SDHC, UART, and USB
- PACKAGE:
  - 144-Terminal Pb-Free Plastic BGA (Ball Grid Array) (ZHH Suffix)

1.4 Functional Block Diagram

Figure 1-1 shows the functional block diagram of the devices.

Figure 1-1. Functional Block Diagram



APPENDIX 3: Features of AIC3204 DSP Onboard Audio Codec.

TEXAS INSTRUMENTS

TLV320AIC3204

SLOS602C – SEPTEMBER 2008 – REVISED NOVEMBER 2014

TLV320AIC3204 Ultra Low Power Stereo Audio Codec

1 Features

- Stereo Audio DAC with 100dB SNR
- 4.1mW Stereo 48ksps DAC Playback
- Stereo Audio ADC with 93dB SNR
- 6.1mW Stereo 48ksps ADC Record
- PowerTune™
- Extensive Signal Processing Options
- Six Single-Ended or 3 Fully-Differential Analog Inputs
- Stereo Analog and Digital Microphone Inputs
- Stereo Headphone Outputs
- Stereo Line Outputs
- Very Low-Noise PGA
- Low Power Analog Bypass Mode
- Programmable Microphone Bias
- Programmable PLL
- Integrated LDO
- 5 mm x 5 mm 32-pin QFN Package

2 Applications

- Portable Navigation Devices (PND)
- Portable Media Player (PMP)
- Mobile Handsets
- Communication
- Portable Computing

3 Description

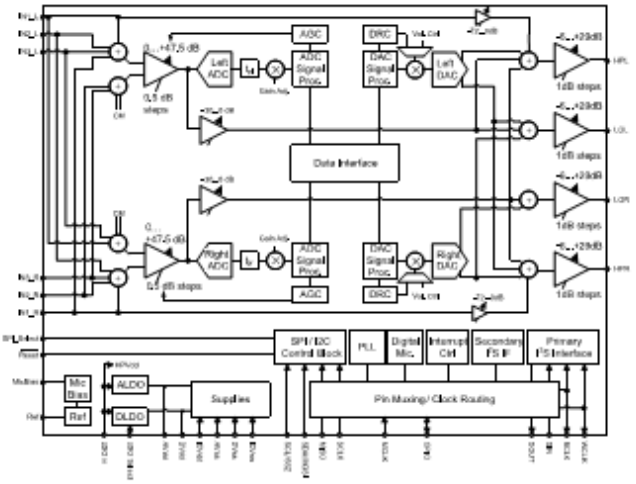
The TLV320AIC3204 (also called the AIC3204) is a flexible, low-power, low-voltage stereo audio codec with programmable inputs and outputs, PowerTune capabilities, fixed predefined and parameterizable signal-processing blocks, integrated PLL, integrated LDOs and flexible digital interfaces.

Device Information(1)

PART NUMBER	PACKAGE	BODY SIZE (NOM)
TLV320AIC3204	VQFN (32)	5.00 mm x 5.00 mm

(1) For all available packages, see the orderable addendum at the end of the datasheet.

4 Simplified Block Diagram



APPENDIX 4: Members of Speech Database.

No	Synthesized Letter	Letter Name	Parsed syllable
1	A	_AA_A	A
2	B	_AB_B	AB
3	B	_BA_B	BA
4	B	_EB_B	EB
5	B	_BE_B	BE
6	C	_AC_C	AC
7	C	_CA_C	CA
8	C	_EC_C	EC
9	C	_CE_C	CE
10	Ç	_AÇ_Ç	AÇ
11	Ç	_ÇA_Ç	ÇA
12	Ç	_EÇ_Ç	EÇ
13	Ç	_ÇE_Ç	ÇE
14	D	_AD_D	AD
15	D	_DA_D	DA
16	D	_ED_D	ED
17	D	_DE_D	DE
18	E	_EE_E	E
19	F	_AF_F	AF
20	F	_FA_F	FA
21	F	_EF_F	EF
22	F	_FE_F	FE
23	G	_AG_G	AG
24	G	_GA_G	GA
25	G	_EG_G	EG
26	G	_GE_G	GE
27	Ğ	_AĞ_Ğ	AĞ
28	Ğ	_ĞA_Ğ	ĞA
29	Ğ	_EĞ_Ğ	EĞ
30	Ğ	_ĞE_Ğ	ĞE
31	H	_AH_H	AH
32	H	_HA_H	HA
33	H	_EH_H	EH
34	H	_HE_H	HE
35	I	_II_I	I

APPENDIX 4: Members of Speech Database (continue).

No	Synthesized Letter	Letter Name	Parsed syllable
36	I	_İİ_İ	İ
37	J	_AJ_J	AJ
38	J	_JA_J	JA
39	J	_EJ_J	EJ
40	J	_JE_J	JE
41	K	_AK_K	AK
42	K	_KA_K	KA
43	K	_EK_K	EK
44	K	_KE_K	KE
45	L	_AL_L	AL
46	L	_LA_L	LA
47	L	_EL_L	EL
48	L	_LE_L	LE
49	M	_AM_M	AM
50	M	_MA_M	MA
51	M	_EM_M	EM
52	M	_ME_M	ME
53	N	_AN_N	AN
54	N	_NA_N	NA
55	N	_EN_N	EN
56	N	_NE_N	NE
57	O	_OO_O	O
58	Ö	_ÖÖ_Ö	Ö
59	P	_AP_P	AP
60	P	_PA_P	PA
61	P	_EP_P	EP
62	P	_PE_P	PE
63	R	_AR_R	AR
64	R	_RA_R	RA
65	R	_ER_R	ER
66	R	_RE_R	RE
68	S	_AS_S	AS
69	S	_SA_S	SA
70	S	_ES_S	ES
71	S	_SE_S	SE

APPENDIX 4: Members of Speech Database (continue).

No	Synthesized Letter	Letter Name	Parsed syllable
72	Ş	_AŞ_Ş	AŞ
73	Ş	_ŞA_Ş	ŞA
74	Ş	_EŞ_Ş	EŞ
75	Ş	_ŞE_Ş	ŞE
76	T	_AT_T	AT
77	T	_TA_T	TA
78	T	_ET_T	ET
79	T	_TE_T	TE
80	U	_UU_U	U
81	Ü	_ÜÜ_Ü	Ü
82	V	_AV_V	AV
83	V	_VA_V	VA
84	V	_EV_V	EV
85	V	_VE_V	VE
86	Y	_AY_Y	AY
87	Y	_YA_Y	YA
88	Y	_EY_Y	EY
89	Y	_YE_Y	YE
90	Z	_AZ_Z	AZ
91	Z	_ZA_Z	ZA
92	Z	_EZ_Z	EZ
93	Z	_ZE_Z	ZE