

YILDIZ TEKNİK ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

Ses Çözümlemesi ve Bireşimi

Burç Bıyıklı

Yüksek Lisans Tezi

YILDIZ TEKNİK ÜNİVERSİTESİ
KÜTÜPHANE DOKÜMANTASYON
DAİRE BAŞKANLIĞI

R 152
152

Kot :

Alındığı Yer : Fen Bil Ens.

Tarih : 10.4.95

Fatura :

Fiyatı : 186.00.-

Ayniyat No : 1-6

Kayıt No : 50980

UDC :

Ek :

YILDIZ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



SES ÇÖZÜMLEMESİ VE BİREŞİMİ



YÜKSEK LİSANS TEZİ

BURÇ BİYİKLİ

ELEKTRONİK VE HABERLEŞME MÜH.

İSTANBUL 1989

.....	IV
.....	V
.....	1
.....	4
.....	12
.....	12
.....	12
.....	12

Araştırma ve çalışmalarım sırasında beni yönlendiren,
düşünceleri ile yardımcı olan tez yöneticim Sayın Doç. Dr.
Sezgin Alsan'a teşekkürü bir borç bilirim. 14

Haziran 1989

Burç BİYİKLİ

İstanbul

.....	17
.....	18
.....	19
.....	20
.....	20
.....	22
.....	23
.....	24
.....	28

İÇİNDEKİLER	Sayfa
ÖZET	IV
ABSTRACT	V
BÖLÜM 1. GİRİŞ	1
BÖLÜM 2. KONUŞMA ÜRETİM MODELİ	4
BÖLÜM 3. KONUŞMA ÇÖZÜMLEMESİ	12
3.1 Doğrusal Öngörüm	12
3.1.1 Giriş	12
3.1.1.1 Genel Bakış	12
3.1.1.2 Doğrusal Öngörüm	12
3.1.2 Parametre Kestirimi	14
3.1.2.1 Tüm-Kutup Modeli	14
3.1.2.2 En Küçük Kareler Yöntemi	16
1) Deterministik İm	16
a) Özilişki Yöntemi	17
b) Çaprazdeğişinti	
Yöntemi	18
2) Rastgele İm	19
a) Duragan Hal	20
b) Duragan Olmayan Hal ...	20
3) Kazanc Hesabı	22
a) Dürtü Girişi	23
b) Beyaz Gürültü Girişi ..	24
3.1.2.3 Öngörücü Parametrelerinin	
Hesabı	25

	<u>Sayfa</u>
1) Tek Adımda Çözüm	25
2) Sonuca adım-adım yaklaşan	29
3) Süzgeç Kararlılığı	30
3.1.2.4 En İyi Kutup Sayısı	31
3.2 Atım Bulma	34
3.2.1 Giriş	34
3.2.1.1 Atım Bulmadaki Sorunlar	34
3.2.1.2 Atım Bulucu Türleri	35
3.2.2 Atım Bulma Algoritmaları	37
3.2.3 Değiştirilmiş Özilişki Yöntemi	
(AUTOC)	37
3.3 Konuşma Uç Noktalarının Belirlenmesi	43
3.3.1 Giriş	43
3.3.2 Uç Noktalarının Belirlenmesindeki sorunlar	45
3.3.3 Uç Noktaları-Belirleme Algoritması	45
BÖLÜM 4. KONUŞMA BİREŞİMİ	55
BÖLÜM 5. GÖRÜNGESEL ÇÖZÜMLEME	56
BÖLÜM 6. ÇÖZÜMLEME-BİREŞİMLEME DİZGESİNİN BENZETİMİ	57
6.1 Giriş	57
6.2 Örneklem ve Çıkış için	
Kullanılan Donanım	57
6.3 Benzetim Ana Yönergesi	59

	<u>Sayfa</u>
6.4 Deneysel Çalışmalar ve Sonuçlar.....	62
BÖLÜM 7. SONUÇ	63
EK-A DONANIMI SÜREN 8088 MAKİNA DİLİ YÖNERGE	64
EK-B BASIC DİLİ BENZETİM ANA YÖNERGESİ	68
EK-C FORTRAN DİLİ FIR TASARIM YÖNERGESİ	82
EK-D UYGULAMA SONUÇLARI	96
KAYNAKLAR	104
ÖZGEÇMİŞ	106

ÖZET

Bu çalışmada konuşma çözümlemesi ve bireşimlemesi konusuna bir giriş yapılması amaçlanmıştır. Bu amaçla önce kuramsal bilgi verilmiş ardından en yalın uygulamalardan biri sayısal bir bilgi işlemci üzerinde benzetimlenmiştir.

Yapılan uygulamada konuşma imi 100-4000 Hz'lik bir BGS'den geçirildikten sonra 3 s'lik bir süre boyunca 10 KHz'lik bir sıklıkla örneklenerek kayıt edilmiştir. Kayıdın başlangıç ve sonundaki sessizlik bölgeleri bir uç-noktaları belirleme algoritması ile atılmıştır. Elde edilen dizi üzerinde bir atım bulma algoritması uygulanarak hem sesli/sessiz/sessizlik ayırımı yapılmış hem de sesli bölümler için atım dönemi belirlenmiştir. Bundan sonra sesli ve sessiz olan bölümler için doğrusal öngörüm yöntemiyle çözümleme yapılmıştır. Bireşim aşamasında uç-noktaları bulunmuş dizi, atım bulma ve çözümlemede elde edilen bilgiler kullanılarak bireşimlenmiştir.

Uygulamanın başarımını inceleyebilmek için görüngesel çözümleme yapılmış ve özgün im ile bireşimlenmiş imin görüngeleri karşılaştırılmıştır. Ayrıca bireşimlenmiş im, girişinde 4000 Hz'lik bir AGS bulunan bir yükseltici yoluyla bir sesyayardan çıkılmış ve böylece öznel bir değerlendirme de yapılmıştır.

ABSTRACT

Purpose of this study is to introduce the speech analysis and synthesis. Therefore, the theoretical informations was first given and then one of the most simple applications was simulated on a digital computer.

In the application the speech signal is first band-pass filtered from 100 Hz to 4000 Hz and then sampled at a frequency of 10 KHz for a duration of 3 seconds. Silences at the begining and end of the record are removed by an algorithm for determining the endpoints of isolated utterances. An algorithm for pitch detection is performed and thus both voiced/unvoiced/silence segments is determined and pitch periods are found for voiced segments. Voiced/unvoiced segments are analyzed by linear prediction. In the synthesis, the sequence that its endpoints were detected is synthesized by using informations that were found in pitch detection and analysis.

In order to compare between original and synthesized spectrums, spectral analysis was executed. Thus performance of the simulation was measured. Also, synthesized sequence was transmitted to a loudspeaker through a LPF and an amplifier. Thus a subjective test was carried out.

BÖLÜM 1. GİRİŞ

Sayısal im işleme yöntemlerinin en önemli uygulamalarından bazıları konuşma işleme alanındadır. Gerçekte, sayısal im işlemenin kuramsal tabanının büyük bir kısmı, konuşma ile ilgili çalışmalardan kaynaklanmıştır.

Konuşma haberleşmenin en doğal biçimidir. Konuşma imi dinleyiciye haberin katkısız içeriğinden çok daha fazla ve gereksiz bilgi taşır. Bu nedenle, konuşma belki de şimdiye kadar bilinen en "sıkıştırılabilir" bilgi kaynağıdır.

Konuşma imlerinin, sayıları olabildiğince az olan ve zamanla yavaş bir şekilde değişen parametreler ile etkin gösterimi konuşma araştırmalarındaki en önemli problemlerden biridir. Konuşma çözümlemesi için geliştirilmiş olan yöntemlerin çoğunda, ilk önce konuşma dalgasının kısa-sürelili bir Fourier çözümlemesi yapılarak akustik veri görüngenel biçime dönüştürülür. Her ne kadar, görüngenel çözümleme imlerle çalışmak için iyi bilinen bir yöntem ise de, konuşma imlerine uygulanması durumunda, konuşma dalgasının dönemli-benzeri özelliklerinden ve durağan olmamasından kaynaklanan çok önemli kısıtlamalarla karşılaşılmaktadır. Sonuç olarak, görüngenel çözümlemeye dayanan yöntemler konuşma çözümlemede çoğu zaman yetersiz kalırlar. Bu çalışmada konuşma çözümleme ve bireşimleme için daha sonraları geliştirilmiş olan değişik bir yöntem kullanılmıştır. Bu yöntemde konuşma dalga biçimi doğrudan doğruya ses dizgesinin geçiş işlevi ve kaynak işlevinin

özellikleri ile ilişkili olan, zamanla değişen parametrelere dayanılarak gösterimlenir. Konuşma dalgasının, görüngesi yerine kendisini modellemekle frekans tanım kümesi yöntemlerinin doğasında bulunan sorunlardan kurtulmuş oluruz. Örneğin, alışılmış Fourier çözümleme yöntemleri, yeterli görüngesel ayrışımı sağlamak için uzunca bir konuşma bölümünü gerektirirler. Bunun sonucunda da hızla değişen konuşma olayları doğru olarak izlenemez. Ayrıca, sesli konuşmaların dönemli doğası nedeniyle, atım harmonikleri arasındaki görünge ile ilgili çok az bilgi bulunmaktadır; bu yüzden frekans tanım kümesi yöntemleri, kadın ve çocuk sesleri gibi atım frekansı yüksek olan sesler için doyurucu olarak uygulanamazlar. Her ne kadar atım-eşzamanlı çözümleme-bireşimleme yöntemleri yukarıda anlatılan zorluklara kısmi bir çözüm sağlayabilirse de, bu gibi yöntemler en gelişmiş sayısal bilgi işlemciler için bile son derecede hantal ve zaman alıcıdır. Bu yüzden büyük tutarlardaki konuşma verilerinin otomatik işlenmesi için uygun değildirler. Bu çalışmada ele alınan yöntemlerin kullanılmasıyla bu sorunlardan bütünüyle kurtulmak olasıdır.

Burada kullanılan konuşma çözümleme-bireşimleme yöntemi konuşma üretimi ve tanınmasındaki araştırma problemlerinin büyük bir çoğunluğuna uygulanabilir. Bu çalışmanın ana hedeflerinden birisi konuşmanın doğal insan konuşmasından ayırt edilemeyen bireşimini gerçekleştirmektir. Konuşma iminin özelliklerini seçime bağlı olarak değiştirmek yoluyla

konuşmanın bilgi taşıyan yapısı hakkında çok şey öğrenilebilir. Bu yüzden burada incelenen yöntemler, verilen bir konuşma iminin akustik özelliklerini, konuşma niteliğini düşürmeden değiştirmek için bir takım gibi kullanılabilir. Bu yöntemlerin diğer bazı olası uygulamaları; konuşmanın etkin saklanması ve iletimi, otomatik formant ve atım çıkarımı, ile konuşmacı ve konuşma tanıma alanlarındadır.

Sonraki bölümlerden; 2.'de konuşma iminin zaman tanım kümesinde gösterimi için parametrik bir modelin tanımlanması, 3.'de model parametrelerinin elde edilmesi için konuşma imi çözümleme yöntemleri, 4.'de çözümlemede bulunan model parametrelerinden konuşma iminin bireşimlenmesi, 5.'de görüngesel çözümleme, 6.'da çözümleme-bireşimleme dizgesinin bilgi işlemci benzetimi, yapılan deneysel çalışmalar ve elde edilen sonuçlar, ve 7.'de sonuçlar ve ileriki çalışmalar için öneriler yer almaktadır.

BÖLÜM 2. KONUŞMA ÜRETİM MODELİ

Modern im işleme yöntemlerindeki im çözümleme işlemleri incelenmekte olan im hakkında önceden elde edilebilen bütün bilgilerin kullanılabilmesini sağlarlar. Bu yüzden im çözümlemedeki ilk adım imin bir modelini oluşturmaktır.

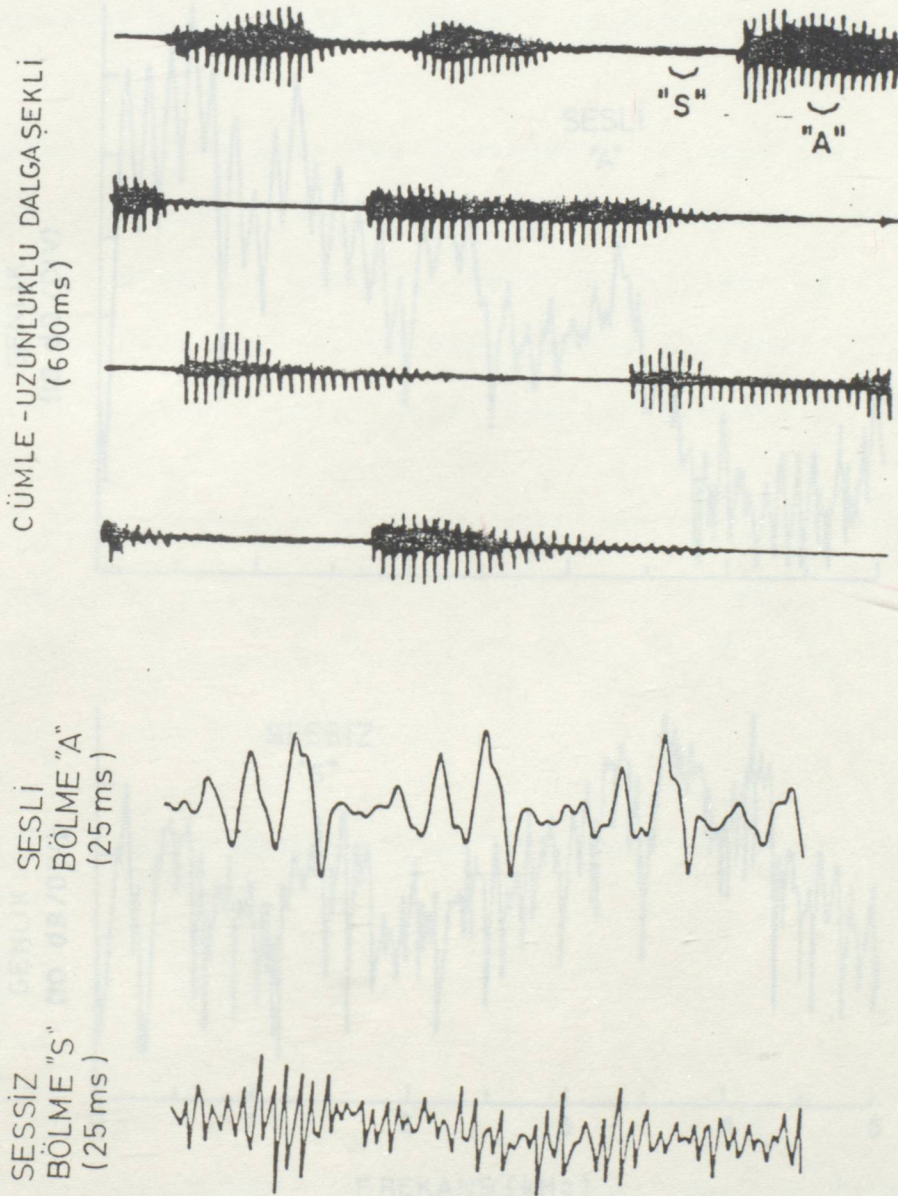
Konuşma sesleri insanın, gırtlğından burun ve dudaklarına kadar uzanan ses dizgesinin akustik uyarımının bir sonucu olarak üretilirler. Uzun süreli bir konuşma dalga biçimi kaydı (Şekil 2.1) incelendiğinde iki temel yapı göze çarpar:

- a) Yüksek enerjili, dönemli-benzeri bölümler
- b) Düşük enerjili, gürültü-benzeri bölümler

Sesli sesler denilen yüksek enerjili bölümlerin üretimi sırasında, ses dizgesi ses tellerinin atım frekansı denilen bir frekansta dönemli-benzeri titreşmesiyle üretilen hava darbeleri dizisi ile uyarılır. Atım frekansı erkekler için 50-200 Hz, kadın ve çocuklar için 200-400 Hz arasındadır. Sessiz sesler olarak adlandırılan düşük enerjili bölümler hiç bir dönemlilik göstermez. Bu bölümlerin üretimi sırasında ses telleri titreşmez ve ses dizgesinin uyarımı, sesiz sese bağlı olarak iki değişik şekilde gerçekleşir: ya akciğerlerden gelen hava ses dizgesindeki bir yerde bir kısıtlama içerisinden geçmeye zorlanarak gürültü-benzeri, çalkantılı bir hava akıntısı oluşturulur (snow kelimesindeki s gibi), yada ses dizgesindeki bir tam kapanma noktası arkasındaki bir basınç birikiminin ardından kısa süreli, geçici bir uyarım

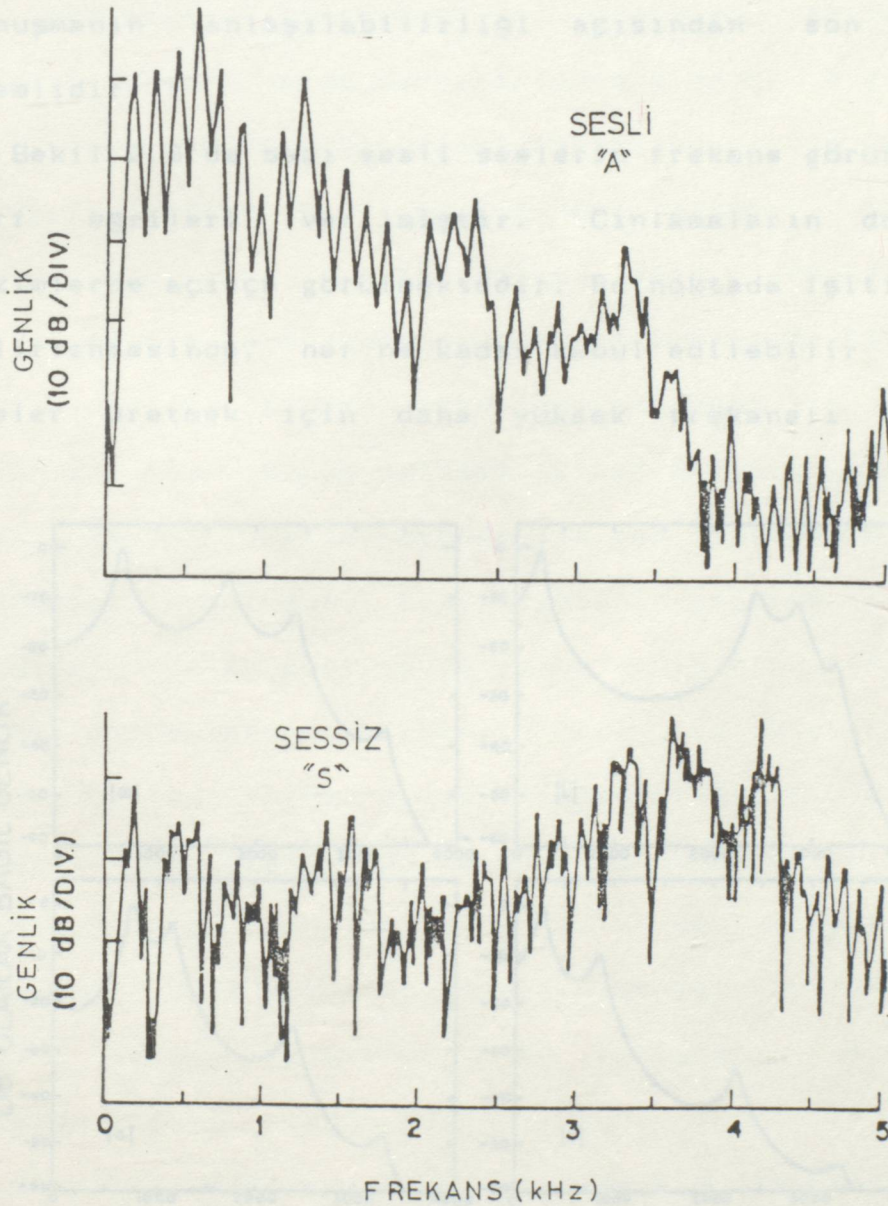
meydana gelir (pit kelimesindeki p gibi).

Ses dizgesi, yukarıda açıklanan uyarım tiplerinden biri ile uyarıldığında, birbiçimsiz bir akustik tüp gibi davranır ve çıkış im görüncesini biçimlendirir. Sesli "a" ve sessiz "s" 'nin kısa süreli dalga biçimi bölümleri



Şekil 2.1 Uzun - Süreli (cümle uzunluklu) ve kısa - süreli konuşma bölümlerinin zaman dalga biçimleri.

Şekil 2.1'de ve görüngeleri Şekil 2.2'de görülmektedir. Dikkat edilirse "s" 'nin yüksek frekanslı içeriği ağırlıklıdır. "a" içinse tam ters durum geçerlidir. Her iki ses için de görüngenin genel biçimi ses dizgesi tarafından belirlenir, "a" için ince yapının daha düzenli

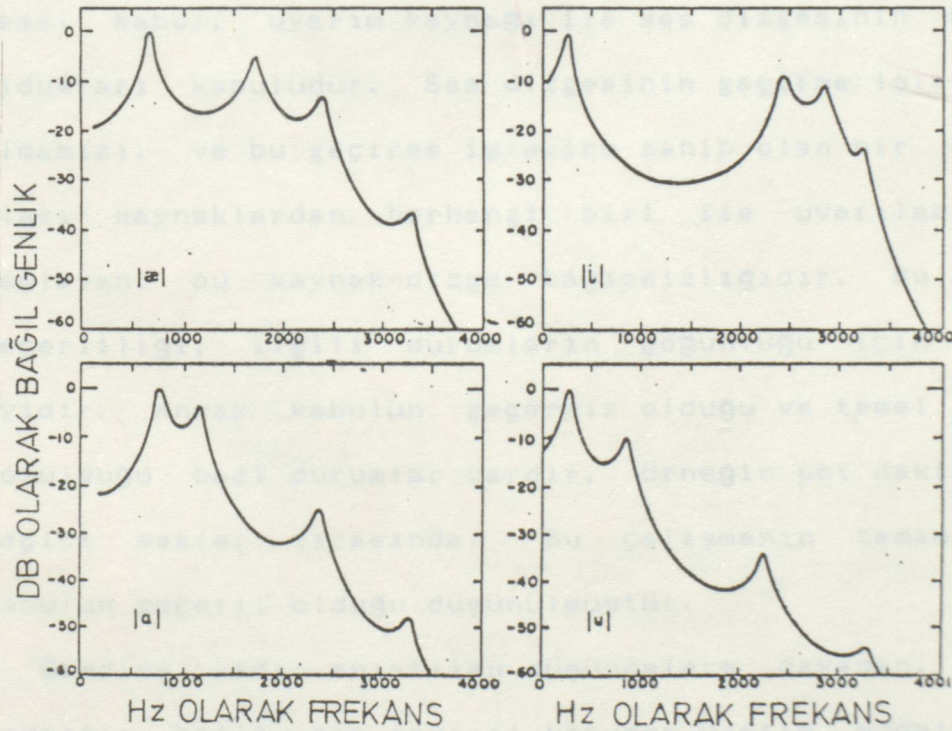


Şekil 2.2 Tipik sesli ve sessiz bölümler için kısa.sürelî frekans görüngeleri.

olmasının nedeni atım frekansıdır.

Ses dizgesi, tıpkı borulu bir orgda olduğu gibi, "formantlar" denilen çınlama frekanslarına sahiptir. Ses dizgesinin her değişik biçimi, değişik bir formant frekansları takımına karşılık gelir. Formantların zamanla değişimleri kadar bant genişlikleri ve frekansları da konuşmanın anlaşılabilirliği açısından son derecede önemlidir.

Şekil 2.3'de bazı sesli seslerin frekans görüngelerinin zarf eğrileri verilmiştir. Çınlamaların doğası bu çizimlerde açıkça görülmektedir. Bu noktada işitilen sesin belirlenmesinde, her ne kadar kabul edilebilir nitelikde sesler üretmek için daha yüksek frekanslı formantlar



Şekil 2.3 Bazı sesliler için frekans görüngesi zarf eğrileri [6].

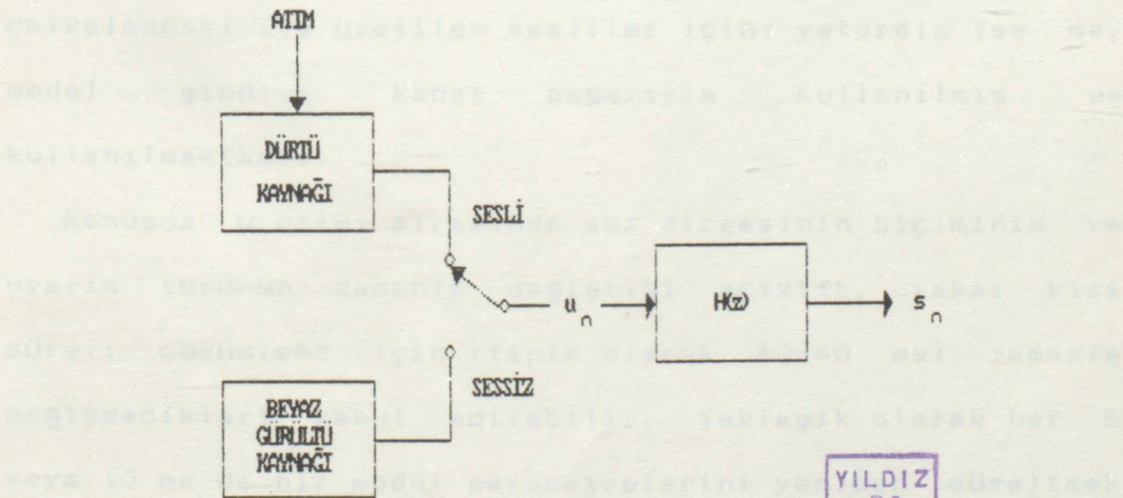
gerekli ise de, algısal açıdan sadece ilk üç formantın önemli olduğuna işaret etmek gerekir. Bu bulgu çeşitli konuşma bant genişliği sıkıştırma dizgelerinde kullanılan temel ilkedir.

Daha önce anlatıldığı gibi, ses dizgesi uyarımı için üç ana mod vardır. Sesli sesler için kaynak gırtlaktır ve ses tellerinin titreşmesi ile üretilen geniş bantlı, dönemli-benzeri hava darbelerinden oluşur. S gibi sessiz sesler için, kaynak kısıtlama noktasındadır ve çalkantılı, rastgele-benzeri hava akıtısından oluşur. Son olarak pop'daki p gibi sessiz sesler için, kaynak kapanma noktasındadır ve bu kapanma noktası arkasında biriktirilen hava basıncının aniden serbest bırakılması ile oluşur.

Hemen hemen bütün konuşma işleme dizgelerinde yapılan temel kabul, uyarım kaynağı ile ses dizgesinin bağımsız oldukları kabulüdür. Ses dizgesinin geçirme işlevini ele almamızı, ve bu geçirme işlevine sahip olan bir dizgenin olası kaynaklardan herhangi biri ile uyarılabilmesini sağlayan, bu kaynak-dizge bağımsızlığıdır. Bu kabulün geçerliliği, ilgili durumların çoğunluğu için oldukça iyidir. Ancak kabulün geçersiz olduğu ve temel modelin bozulduğu bazı durumlar vardır, örneğin pot'daki p gibi geçici sesler sırasında. Bu çalışmanın tamamında bu kabulün geçerli olduğu düşünülmüştür.

Şimdiye kadar anlatılan düşüncelere dayanan, oldukça başarılı basit bir sayısal konuşma üretim modeli Şekil 2.4'de görülmektedir. Bu model hala yaygın olarak

kullanılan en eski konuşma üretim modelidir. Model ses dizgesinin uyarım biçimlerini benzetimleyen iki uyarım kaynağı ile ses dizgesini benzetimleyen bir süzgeçten oluşur. Sesli seslerin üretimi için gerekli olan dönemli-benzeri uyarım, atım dönemi imi ile denetlenen bir dönemli dürtü üretici tarafından sağlanır. Sessiz seslerin üretiminde gerekli gürültü-benzeri uyarım ise bir rastgele sayı veya gürültü (beyaz gürültü) üretici ile sağlanır. Bu gürültü üreticinin çıkışı sessiz sesler için hem rastgele-benzeri çalkantıyı hem de basınç birikim dalga biçimini benzetimler. Uygun uyarım türü sesli/sessiz açkısı ile seçilir. Modeldeki $H(z)$ süzgeci ayrık, zamanla değişen doğrusal bir süzgeçtir. Bu süzgeç ses dizgesini benzetimlediğinden, süzgeç katsayıları sürekli konuşma sırasında bir bakıma ses dizgesini zamanın bir işlevi olarak ifade eder. Ses dizgesi biçiminin zamanla değişimlerinin bir durağan şekiller dizisi ile yeterli doğrulukla yaklaşımlanabildiğini kabul edersek, ses



Şekil 24 Ayrık konuşma üretim modeli.



dizgesi için karmaşık z tanım kümesinde bir geçiş işlevi tanımlamak olasıdır. Bilindiği gibi doğrusal bir dizgenin geçiş işlevi her zaman kutup ve sıfırları ile gösterilebilir. Burunsal olmayan sesli konuşma sesleri için ses dizgesinin geçiş işlevinin sıfırlarının olmadığı iyi bilinmektedir. Ses dizgesinin sessiz ve burunsal sesler için benzetimi çoğunlukla ses dizgesinin kutupları kadar sıfırlarını da içerir. Sessiz ve burunsal sesler için ses dizgesinin geçiş işlevinin sıfırları, z düzleminde birim çember içerisinde bulunduğundan, geçiş işlevinin payındaki her çarpan paydasındaki katlı kutuplar ile yaklaşımlanabilir. Ayrıca bir kutubun yeri algısal açıdan bir sıfırın yerinden daha fazla öneme sahiptir. Bu nedenlerle ses dizgesi bir tüm-kutuplu (özbağlaşımlı) süzgeç ile yeterli olarak benzetimlenebilir. Bu süzgecin kararlı olması için, kutupları birim çemberin içerisinde olmalıdır. Her ne kadar bu basit sesli/sessiz düzenleme bir çok sesler için (örneğin "v" ve "z" gibi hava çalkalanması ile üretilen sesliler için) yetersiz ise de, model şimdiye kadar başarıyla kullanılmış ve kullanılmaktadır.

Konuşma üretimi sırasında ses dizgesinin biçiminin ve uyarım türünün zamanla değiştiği açıktır, fakat kısa süreli çözümleme için (tipik olarak 10-40 ms) zamanla değişmedikleri kabul edilebilir. Yaklaşık olarak her 5 veya 10 ms'de bir model parametrelerini yeniden düzeltmek yeterlidir.

Yukarıda anlatılan modeli denetlemek için zamanın bir işlevi olarak uygun model parametrelerinin (atım dönemi, açkı konumu ve süzgeç katsayıları) bilinmesi gerekir. Bu, yani gerçek konuşmadan uygun model parametrelerini kestirmek hemen hemen bütün konuşma çözümleme dizgelerinin amacıdır. Çoğu konuşma bireşimleme dizgelerinin amacı ise, herhangi bir uygun yöntem ile elde edilen bu parametreleri, algısal olarak gerçek imden ayırt edilemeyen bireşimlenmiş bir konuşma imi elde etmek için kullanmaktır. Konuşma çözümleme-bireşimleme dizgelerinde bu iki problem şu iki amaçla birleşmiş durumdadır:

1) Etkinlik, yani bireşim dizgesinin bit oranını, alışılmış dalga biçimi gösterimleri için gerekli olanın altına indirmek ve böylece iletimdeki bant genişliğini ve saklama dizgelerindeki belleği küçük tutmak,

2) Esneklik, yani model parametrelerinin değiştirilmesi yoluyla, konuşmanın bazı istenilen tarzlarda nitelendirilmesini ve değiştirilmesini olası kılmak.

BÖLÜM 3. KONUŞMA ÇÖZÜMLEMESİ

3.1 Doğrusal Öngörüm

3.1.1 Giriş

3.1.1.1 Genel Bakış

Genel dinamik dizgelerin davranışlarının matamatiksel olarak çözümlenmesi, bu yüzyılın başlangıcından beri bir ilgi alanıdır. Son yirmi yıldır, elektronik sayısal bilgisayarlar ortaya çıktığından beri problem giderek artan bir kuvvetle araştırılmaya başlanmıştır. Dinamik dizgelerin çıkışlarının çözümlenmesi, temel olarak istatistik, ekonomi ve iletişim alanlarında geliştirilmiş olan "zaman dizileri çözümülemesi" ne oldukça uygundur.

Bu kısımda zaman dizileri çözümüleme yöntemlerinden biri olan doğrusal öngörüm, matamatiksel katılıktan çok sezgisel bir yaklaşımla incelenmiştir.

3.1.1.2 Doğrusal Öngörüm

Zaman dizileri çözümülemesi uygulamada kullanılır iken, çözümlenmekte olan dizgenin çıkışındaki $s(t)$ sürekli-zaman imi ayrık-zamanlı bir $s(nT)$ imi elde etmek için örneklenir. $s(nT)$ imi zaman dizisi olarak da bilinir. Burada n bir tamsayı değişken ve T örnekleme aralığıdır. Buna göre örnekleme frekansı $f_s = 1/T$ 'dir. (Bundan sonraki kısımlarda $s(nT)$ 'yi s_n ile göstereceğiz.)

Zaman dizileri çözümülemesi şimdiye kadar çoğunlukla güç görüngesi, çapraz-görünge, tutarlık işlevleri, özilişki ve çapraz-ilişki işlevleri kestiriminde kullanılmıştır. Son zamanlarda dizge modellemeye olan ilgi daha fazladır. Eger

verilen bir imin davranışı için parametrik bir model geliştirilebilirse o zaman bulunan modelin öngörüm veya tahmin, denetim ve veri sıkıştırma gibi değişik uygulamalar için kullanılabileceği açıktır.

Hala kullanılan en güçlü modellerden birisi, verilen bir s_n iminin, girişine bilinmeyen bir u_n imi uygulanan ve girişi ile çıkışı arasında:

$$s_n = - \sum_{k=1}^p a_k s_{n-k} + G \sum_{l=0}^q b_l u_{n-l}, \quad b_0 = 1 \quad (1)$$

bağıntısı bulunan bir dizgenin çıkışı olarak düşünüldüğü modeldir. Burada $a_k, 1 \leq k \leq p, b_l, 1 \leq l \leq q,$ ve G kazancı varsayılan dizgenin parametreleridir. (1) eşitliği s_n "çıkış" ının, geçmişteki çıkışlar ile şimdiki ve geçmişteki girişlerin doğrusal bir işlevi olduğunu ifade eder. Yani s_n imi geçmişteki çıkışlar ile girişin şimdiki ve geçmişteki değerlerinin doğrusal bileşimlerinden öngörülebilir. "Doğrusal Öngörüm" adının kaynağı da budur. (1) eşitliği, her iki tarafının z dönüşümü alınmak yoluyla frekans tanım kümesinde de ifade edilebilir. Şekil 2.4'de olduğu gibi dizgenin geçiş işlevi $H(z)$ ise, o zaman (1)'den:

$$H(z) = \frac{S(z)}{U(z)} = G \frac{1 + \sum_{l=1}^q b_l z^{-l}}{1 + \sum_{k=1}^p a_k z^{-k}} \quad (2)$$

buluruz. Burada

$$S(z) = \sum_{n=-\infty}^{\infty} s_n z^{-n} \quad (3)$$

s_n 'in z dönüşümü, ve $U(z)$ de u_n 'in z dönüşümüdür. (2)'deki $H(z)$ genel kutup-sıfır modelidir. Pay ve payda çokterimlilerinin kökleri, sırasıyla modelin sıfır ve kutuplarıdır.

Modelin oldukça önemli olan iki özel durumu vardır:

1) tüm-sıfır modeli: $a_k = 0, 1 \leq k \leq p$

2) tüm-kutup modeli: $b_l = 0, 1 \leq l \leq q$

Tüm-sıfır modeli istatistik literatüründe "moving average" (MA) modeli, ve tüm-kutup modeli de "autoregressive" (AR) modeli olarak bilinir. Bunlara bağlı olarak kutup-sıfır modeli de "autoregressive moving average" (ARMA) modeli olarak bilinir.

Bize sadece tüm-kutup modeli gerekli olduğundan diğer modeller ile ilgilenmeyeceğiz.

3.1.2 Parametre Kestirimi

3.1.2.1 Tüm-Kutup Modeli

Tüm-kutup modelinde s_n iminin, kendisinin geçmişteki değerleri ile bir u_n girişinin doğrusal bir bileşimi olarak verildiği kabul edilir:

$$s_n = - \sum_{k=1}^p a_k s_{n-k} + G u_n \quad (4)$$

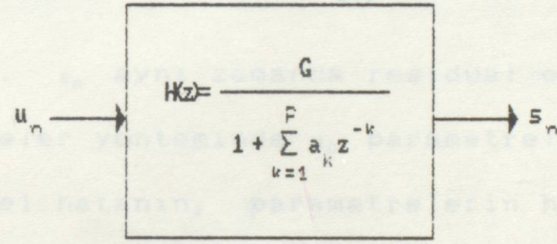
burada G bir kazanç çarpanıdır. Bu model Şekil 3.1.2'de zaman ve frekans tanım kümelerinde görülmektedir. (2)'deki $H(z)$ geçiş işlevi bu durumda bir tüm-kutup geçiş işlevine indirgenir.

3.1.2.2 En Küçük Kareler Yöntemi

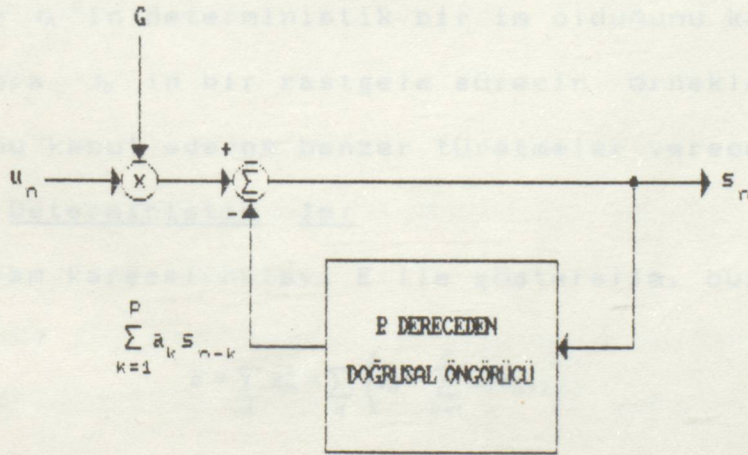
$$H(z) = \frac{G}{1 + \sum_{k=1}^P a_k z^{-k}} \quad (5)$$

Verilen belirli bir s_n imi için çözülmesi gereken problem a_k öngörücü katsayılarını ve G kazancını bir yöntem ile belirlemektir.

Türetmeler s_n 'in önce deterministik bir im olduğu ve sonra bir rastgele sürecin örneklerinden biri olduğu kabul edilip, sezgisel bir en küçük kareler yöntemi kullanılarak verilecektir.



(a)



(b)

Şekil 3.1.1 Ayırık tüm-kutup modeli

(a) Frekans tanım kümesinde

(b) Zaman tanım kümesinde.

3.1.2.2 En Küçük Kareler Yöntemi

Burada u_n girişinin bir çok uygulamalarda, örneğin EEG çözümlemesinde, olduğu gibi hiç bilinmediğini kabul edeceğiz. Bu yüzden s_n imi geçmişteki örneklerin doğrusal olarak ağırlıklandırılmış bir toplamından sadece yaklaşık olarak öngörülebilir. s_n 'in bu yaklaşımı $\tilde{s}_n$ olsun, burada

$$\tilde{s}_n = - \sum_{k=1}^p a_k s_{n-k}. \quad (6)$$

dır. Bu durumda gerçek değer s_n ile öngörülen değer $\tilde{s}_n$ arasındaki hata

$$e_n = s_n - \tilde{s}_n = s_n + \sum_{k=1}^p a_k s_{n-k}. \quad (7)$$

ile verilir. e_n aynı zamanda residual olarak da bilinir. En küçük kareler yönteminde a_k parametreleri ortalama veya toplam karesel hatanın, parametrelerin her birine göre en küçültülmesinin bir sonucu olarak elde edilirler.

Önce s_n 'in deterministik bir im olduğunu kabul edeceğiz ve sonra s_n 'in bir rastgele sürecin örneklerinden biri olduğunu kabul ederek benzer türetmeler vereceğiz.

1) Deterministik İm:

Toplam karesel hatayı E ile gösterelim, burada

$$E = \sum_n e_n^2 = \sum_n \left(s_n + \sum_{k=1}^p a_k s_{n-k} \right)^2. \quad (8)$$

dır. (8)'deki toplamın sınırları ve s_n 'in bu sınırlar arasındaki tanımı önemlidir. Bununla beraber önce E'yi toplamın sınırlarını belirtmeden en küçültelim. E,

$$\frac{\partial E}{\partial a_i} = 0, \quad 1 \leq i \leq p. \quad (9)$$

denklemleri kurularak en küçültülür. (8) ve (9)'dan:

$$\sum_{k=1}^p a_k \sum_n s_{n-k} s_{n-i} = - \sum_n s_n s_{n-i}, \quad 1 \leq i \leq p. \quad (10)$$

eşitlik takımını elde ederiz. (10) eşitlikleri en küçük kareler terminolojisinde "normal eşitlikler" olarak bilinir. s_n iminin herhangi bir tanımı için, (10) bağıntısı, (8)'deki E'yi en küçülten öngörücü katsayıları $\{a_k, 1 \leq k \leq p\}$ için çözülebilecek olan p bilinmeyenli p eşitlikten meydana gelen bir takımı oluşturur.

E_p ile göstereceğimiz en küçük toplam karesel hata, (8)'i açmak ve (10)'u (8)'de yerine koymakla elde edilir. Sonucun

$$E_p = \sum_n s_n^2 + \sum_{k=1}^p a_k \sum_n s_n s_{n-k}. \quad (11)$$

olduğu gösterilebilir.

Şimdi (8), (10), ve (11)'deki n boyunca toplamaların sınırlarını belirleyeceğiz. Parametrelerin kestirimi için iki ayrı yönteme ulaşan iki önemli durum vardır.

a) Özilişki Yöntemi:

Burada (8)'deki hatanın sonsuz süre boyunca, $-\infty < n < \infty$, en küçültüldüğünü kabul edeceğiz. Bu durumda (10) ve (11) eşitlikleri

$$\sum_{k=1}^p a_k R(i-k) = -R(i), \quad 1 \leq i \leq p \quad (12)$$

$$E_p = R(0) + \sum_{k=1}^p a_k R(k) \quad (13)$$

eşitliklerine indirgenir, burada

$$R(i) = \sum_{n=-\infty}^{\infty} s_n s_{n+i} \quad (14)$$

s_n iminin özilişki işlevidir. $R(i)$ 'nin i 'nin bir çift işlevi, yani

$$R(-i) = R(i) \quad (15)$$

olduğuna dikkat edin. $R(i-k)$ katsayıları bir özilişki matrisi oluşturduklarından dolayı bu yönteme özilişki yöntemi diyeceğiz. Bir özilişki matrisi simetrik bir Toeplitz matrisidir. (Toeplitz matrisi her diyagonalı boyunca bütün elemanları eşit olan bir matristir.)

Uygulamada s_n imi sadece sonlu bir süre boyunca bilinir, veya im ile sadece sonlu bir süre boyunca ilgilenilir. Yaygın bir yöntem s_n imini, $0 \leq n \leq N-1$ aralığı dışında sıfır olan başka bir im, s'_n , elde etmek için bir w_n pencere işleviyle çarpmaktır:

$$s'_n = \begin{cases} s_n w_n, & 0 \leq n \leq N-1 \\ 0, & \text{aksi halde} \end{cases} \quad (16)$$

Bu durumda özilişki işlevi

$$R(i) = \sum_{n=0}^{N-1-i} s'_n s'_{n+i}, \quad i \geq 0. \quad (17)$$

ile verilir. Pencere işlevi w_n 'in biçimi önemli olabilir. Bu konuda ayrıntılı bilgi [2] ve [6]'da bulunabilir.

b) Çapraz Değişinti Yöntemi:

Özilişki yöntemine zıt olarak, burada (8)'deki E

hatasının sonlu bir süre boyunca, örneğin $0 \leq n \leq N-1$ için en küçültüldüğünü kabul edeceğiz. Bu durumda (10) ve (11) eşitlikleri

$$\sum_{k=1}^p a_k \varphi_{ki} = -\varphi_{0i}, \quad 1 \leq i \leq p \quad (18)$$

$$E_p = \varphi_{00} + \sum_{k=1}^p a_k \varphi_{0k} \quad (19)$$

eşitliklerine indirgenir, burada

$$\varphi_{ik} = \sum_{n=0}^{N-1} s_{n-i} s_{n-k} \quad (20)$$

s_n iminin verilen aralıktaki çaprazdeğişintisidir. (18)'deki φ_{ki} katsayıları bir çaprazdeğişinti matrisi oluşturur ve bu yüzden bu yöntem çaprazdeğişinti yöntemi diyeceğiz. φ_{ik} çaprazdeğişinti matrisinin simetrik, yani $\varphi_{ik} = \varphi_{ki}$ olduğu (20)'den kolaylıkla görülebilir. Bununla beraber özilişki matrisindeki gibi her diyagonal boyunca terimler eşit değildir. Bunun böyle olduğu (20)'den

$$\varphi_{i+1,k+1} = \varphi_{ik} + s_{-i-1} s_{-k-1} - s_{N-1-i} s_{N-1-k} \quad (21)$$

yazmakla görülebilir. Ayrıca (21)'e dikkat edilirse s_n iminin $-p \leq n \leq N-1$ için değerlerinin, yani toplam $p+N$ örneğin bilinmesi gerektiği anlaşılır. Çaprazdeğişinti yöntemi, aralığın sınırları sonsuza doğru götürüldüğünde özilişki yöntemine indirgenir.

2) Rastgele İm:

Eğer s_n iminin bir rastgele sürecin örneklerinden biri olduğu kabul edilirse, bu durumda (7)'deki e_n hatası da bir rastgele sürecin örneklerinden biri olur. En küçük

kareler yönteminde hatanın karesinin beklenen değerini en küçültürüz. Buna göre

$$E = \mathbb{E}(e_n^2) = \mathbb{E} \left(s_n + \sum_{k=1}^p a_k s_{n-k} \right)^2. \quad (22)$$

dir. (9)'u (22)'ye uygulayarak normal eşitlikler elde ederiz:

$$\sum_{k=1}^p a_k \mathbb{E}(s_{n-k} s_{n-i}) = -\mathbb{E}(s_n s_{n-i}), \quad 1 \leq i \leq p. \quad (23)$$

Bu durumda en küçük ortalama hata

$$E_p = \mathbb{E}(s_n^2) + \sum_{k=1}^p a_k \mathbb{E}(s_n s_{n-k}). \quad (24)$$

ile verilir. (23) ve (24)'deki beklenen değerlerin alınması s_n sürecinin durağan olup olmadığına bağlıdır.

a) Durağan Hal:

Durağan bir s_n süreci için

$$\mathbb{E}(s_{n-k} s_{n-i}) = R(i - k) \quad (25)$$

alabiliriz, burada $R(i)$ sürecin özilişkisi. Bu durumda (23) ve (24) eşitlikleri sırasıyla (12) ve (13)'e özdeş eşitliklere indirgenir. Buradaki tek fark özilişkinin, deterministik bir imin yerine durağan bir sürecin özilişkisi olmasıdır. Durağan (ve ergodik) bir süreç için özilişki bir zaman ortalaması olarak hesaplanabilir. $R(i)$ 'nin bilinen sonlu bir s_n iminden kestirimi için literatürde değişik yaklaşımlar önerilmektedir.

b) Durağan Olmayan Hal:

Durağan olmayan bir s_n süreci için

$$\mathbb{E}(s_{n-k} s_{n-i}) = R(n - k, n - i) \quad (26)$$

alabiliriz, burada $R(t, t')$, t ile t' anları arasındaki duran olmayan özilişkidir. $R(n-k, n-i)$, zaman indeksi n 'in bir işlevidir. Genel olarak a_k parametrelerinin kestirimiyle $n=0$ anında ilgilendiğimizi kabul edeceğiz. Bu durumda (23) ve (24) eşitlikleri

$$\sum_{k=1}^p a_k R(-k, -i) = -R(0, -i) \quad (27)$$

$$E'_p = R(0, 0) + \sum_{k=1}^p a_k R(0, k). \quad (28)$$

eşitliklerine indirgenir. s_n iminden duran olmayan özilişki katsayılarının kestiriminde, duran olmayan süreçlerin ergodik olmadığına ve bu yüzden bütün ortalamanın bir zaman ortalaması ile yer değiştiremeyeceğine dikkat ediniz. Bununla beraber duran olmayan süreçlerin, bölgesel duran süreçler diye bilinen belli bir sınıfı için özilişki işlevini, zamandaki bir noktaya göre kısa-sürelili bir ortalama olarak kestirmek mantıklıdır. Bölgesel duran olduğu kabul edilebilen duran olmayan süreç örnekleri konuşma ve EEG imleridir.

Duran haldekine benzer bir tarzda, $R(-k, -i)$ 'yi (20)'deki φ_{ik} ile kestiririz. Duran olmayan özilişki için bu yaklaşımı kullanmak, (27)'deki a_k parametreleri için deterministik haldeki çaprazdeğişinti yönteminde (18) ile verilene eşdeğer bir çözüm verir.

Duran bir im için: $R(t, t') = R(t - t')$ olduğundan (27) ve (28) normal eşitlikleri (12) ve (13)'e indirgenir.

3) Kazanç Hesabı:

En küçük kareler yönteminde girişin bilinmediğini kabul ettiğimizden dolayı G kazancı için bir değer belirlemek fazla anlamlı değildir. Bununla birlikte, yapılabilecek bazı ilginç incelemeler vardır.

(7) eşitliği

$$s_n = - \sum_{k=1}^p u_k s_{n-k} + e_n. \quad (29)$$

şeklinde yeniden yazılabilir. (4) ile (29)'u karşılaştıırırsak, çıkış olarak s_n imini verecek olan tek un giriş iminin $Gu_n = e_n$ eşitliğini sağlayan u_n giriş imi olduğunu görürüz. Yani giriş imi hata imi ile orantılıdır. Başka herhangi bir u_n girişi için Şekil 3.1.1'deki $H(z)$ süzgecinin çıkışı s_n 'den farklı olacaktır. Bununla birlikte eğer, u_n girişi ne olursa olsun çıkış iminin enerjisinin özgün s_n imininkine eşit olması gerektiği koşulunu getirirsek o zaman hiç olmazsa giriş imindeki toplam enerjiyi belirleyebiliriz. Buna göre, $H(z)$ süzgeci sabit olduğundan Gu_n giriş imindeki toplam enerjinin, kullanılan yöntemle bağlı olarak (13) veya (19)'daki E_p ile verilen hata imindeki toplam enerjiye eşit olması gerektiği açıktır.

Özellikle önemli olan iki giriş tipi: deterministik dürtü ve durağan beyaz gürültüdür. Bu iki girişin her biri için $H(z)$ süzgecinin tepkesini sinayarak tüm-kutup modelinin zaman tanım kümesi özelliklerini daha iyi kavrayacağız. Daha sonra giriş kazancı bir özilişki

çözümlemesinin bir yan ürünü olarak belirlenecektir.

a) Dürtü Girişi:

Tüm-kutuplu $H(z)$ süzgecine giriş olarak $n=0$ 'daki bir dürtü veya birim örnek uygulansın, yani $u_n = \delta_{n0}$ olsun, burada δ_{nm} Kronecker delta'sıdır. Bu durumda $H(z)$ süzgecinin çıkışı h_n dürtü tepkesidir, burada

$$h_n = - \sum_{k=1}^p a_k h_{n-k} + G \delta_{n0}. \quad (30)$$

dır.

h_n dürtü tepkesinin özilişkisi $\hat{R}(i)$ ile s_n iminin özilişkisi $R(i)$ arasında ilginç bir bağıntı vardır. (30)'u h_{n-1} ile çarparak tüm n boyunca toplamını alalım. Sonucun:

$$\hat{R}(i) = - \sum_{k=1}^p a_k \hat{R}(i-k), \quad 1 \leq |i| \leq \infty \quad (31)$$

$$\hat{R}(0) = - \sum_{k=1}^p a_k \hat{R}(k) + G^2. \quad (32)$$

olduğu gösterilebilir. Yukarıda getirdiğimiz koşula göre h_n 'deki toplam enerji s_n 'dekine eşit olmalıdır. buna göre sıfırıncı özilişki katsayısı imdeki toplam enerjiye eşit olduğundan

$$\hat{R}(0) = R(0) \quad (33)$$

almalıyız. (33)'den ve (12) ile (31) arasındaki benzerlikden

$$\hat{R}(i) = R(i), \quad 0 \leq i \leq p. \quad (34)$$

sonucuna ulaşırız. Bu sonuç $H(z)$ 'in dürtü tepkesinin ilk $p+1$ özilişki katsayısının, imin karşılık gelen özilişki katsayılarına özdeş olduğunu ifade eder. Buna göre

özilişki yöntemiyle doğrusal öngörüm problemi, yeni bir tarzda şöyle ifade edilebilir: (5)'deki $H(z)$ 'in biçiminde öyle bir süzgeç bulun ki dürtü tepkesinin özilişkisinin ilk $p+1$ değeri, im özilişkisinin ilk $p+1$ değerine eşit olsun ve (31)'i sağlasın.

(32), (34), ve (13)'e göre kazanç için

$$G^2 = E_p = R(0) + \sum_{k=1}^p a_k R(k) \quad (35)$$

yazılabilir, burada G^2 , $G\delta_{n0}$ giriş imindeki toplam enerjidir.

b) Beyaz Gürültü Girişi:

Burada u_n girişinin, sıfır ortalamalı ve birim değişintili, yani tüm n değerleri için $E(u_n) = 0$ ve $E(u_n u_m) = \delta_{nm}$ olan, ilişkisiz örneklerden bir dizi (beyaz gürültü) olduğunu kabul edeceğiz. Süzgecin çıkışını s_n ile gösterelim. Sabit bir $H(z)$ süzgeci için s_n çıkışı durağan bir rastgele süreci oluşturur:

$$\hat{s}_n = - \sum_{k=1}^p a_k \hat{s}_{n-k} + u_n. \quad (36)$$

(36)'yı $\hat{s}_{n-i}$ ile çarparak beklenen değerleri alalım. u_n ile $\hat{s}_{n-i}$ 'nin $i > 0$ için ilişkisiz olduklarını göz önüne alarak sonucun (31) ile (32)'ye özdeş olduğu gösterilebilir, burada $\hat{R}(i) = E(\hat{s}_n \hat{s}_{n-i})$, $\hat{s}_n$ çıkışının özilişkisi. Bu yüzden (31) ile (32) tüm-kutuplu bir rastgele süreci de tam olarak belirler. (31) eşitlikleri istatistik literatüründe Yule-Walker eşitlikleri olarak bilinir.

Rastgele hal için sn çıkışının ortalama enerjisi (veya değişintisi) özgün sn iminin değişintisine eşit olmalıdır veya sıfır-ortalamalı bir rastgele sürecin sıfırncı özilişkisi değişintisine eşit olduğundan $\hat{R}(0) = R(0)$ olmalıdır. Önceki kısımda verilene benzer bir düşünce tarzı ile (34) ile (35)'in rastgele hal için de geçerli oldukları sonucuna ulaşırız.

Yukarıda gösterilenlerden, tüm-kutuplu bir süzgecin çıkışının özilişki katsayıları arasındaki bağıntıların, giriş tek bir dürtü de olsa beyaz gürültü de olsa aynı olduğunu anlarız. Bu, her iki tip giriş de özdeş özilişkilere ve tabi özdeş düz görüneye sahip olduklarından beklenen bir sonuçtur. Deterministik dürtü ile istatiksel beyaz gürültü arasındaki bu ikilik şaşırtıcıdır. Bu ikiliğin kullanışlılığı konuşma sürecinin Şekil 2.4'deki gibi modellenmesinde çarpıcı bir şekilde ortaya çıkar. Bu modellemede hem beyaz gürültü hem de birim dürtüler konuşma bireşimleme için gerçekten kullanılır.

3.1.2.3 Öngörücü Parametrelerinin Hesabı

1) Tek Adımda Çözüm Yöntemleri:

Doğrusal öngörümün önceki bölümde verilen her iki formülasyonunda $a_k, 1 \leq k \leq p$ öngörücü katsayıları, p bilinmeyenli p eşitlikten oluşan bir denklem takımını çözerek hesaplanabilir. Bu eşitlikler özilişki yöntemi (duragan hal) için (12) ve çaprazdeğişinti yöntemi (duragan olmayan hal) için (18)'dir. Gerekli hesaplamaları

yapmak için çeşitli standart yöntemler vardır, örneğin Gauss veya Crout indirgeme yöntemleri gibi. Bununla beraber (12) ile (18)'e dikkat edilirse her iki haldeki katsayılar matrisi birer çaprazdeğişinti matrisidir. Çaprazdeğişinti matrisleri simetrik ve, uygulamada çoğunlukla pozitif belirli oldukları halde genelde pozitif yarıbelirlilidirler. Bu yüzden (12) ve (18) denklem takımları, karekök veya Cholesky ayrışım yöntemiyle daha etkin olarak çözülebilir. Bu yöntem, genel yöntemlerdeki hesaplama ve saklama belleği gereksinimlerinin yaklaşık yarısına gerek gösterir. Bu yöntemin sayısal kararlılık özellikleri iyi bilinmektedir; yöntemin bütünüyle kararlı olduğu kabul edilebilir.

(12)'deki özilişki normal eşitliklerinin çözülmesinde, bu eşitliklerin özel biçimlerinden dolayı, saklama belleği ve hesaplama zamanında daha fazla indirgeme yapmak olanaklıdır. (12) eşitliği matris biçiminde

$$\begin{bmatrix} R_0 & R_1 & R_2 & \cdots & R_{p-1} \\ R_1 & R_0 & R_1 & \cdots & R_{p-2} \\ R_2 & R_1 & R_0 & \cdots & R_{p-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{p-1} & R_{p-2} & R_{p-3} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_p \end{bmatrix} \quad (37)$$

olarak açılabilir. $p \times p$ 'lik özilişki matrisinin simetrik ve her diyagonal boyunca elemanların aynı olduklarına (yani bir Toeplitz matrisi olduğuna) dikkat edin. Levinson bu

tip bir eşitliği çözmek için üstün bir yinelemeli işlem türetmiştir. Bu işlem daha sonra Robinson tarafından yeniden formüle edilmiştir. Levinsonun yöntemi (37)'nin sağ tarafındaki sütun vektörün genel bir sütun vektör olduğunu kabul eder. Bu sütun vektörün, özilişki matrisindekiler ile aynı elemanları içerdigi gerçeğinin kullanımının sağlanmasıyla, Durbin'e atfedilen, Levinsonunkinin iki katı kadar hızlı başka bir yöntem ortaya çıkar. Durbin'in yinelemeli işlemi aşağıdaki gibi tanımlanabilir:

$$E_0 = R(0) \quad (38a)$$

$$k_i = - \left[R(i) + \sum_{j=1}^{i-1} a_j^{(i-1)} R(i-j) \right] / E_{i-1} \quad (38b)$$

$$a_i^{(i)} = k_i \quad (38c)$$

$$a_j^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)}, \quad 1 \leq j \leq i-1$$

$$E_i = (1 - k_i^2) E_{i-1}. \quad (38d)$$

(38b)-(38d) eşitlikleri $i=1,2,\dots,p$ için yinelenerek çözülür. Son çözüm

$$a_j = a_j^{(p)}, \quad 1 \leq j \leq p. \quad (38e)$$

ile verilir. p . dereceden bir öngörücü için çözümün elde edilmesinde, gerçekte derecesi p 'den küçük olan bütün öngörücüler için de çözümlerin hesaplandığına dikkat edilmelidir.

Bütün özilişki katsayıları bir sabit ile çarpılırsa (37)'nin çözümü değişmez. Özellikle bütün $R(i)$ 'ler $R(0)$ ile bölünerek normalize edilirse, normalize edilmiş

özilişki katsayıları olarak bilinen $r(i)$:

$$r(i) = \frac{R(i)}{R(0)} \quad (39)$$

leri elde ederiz. Bu katsayılar $|r(i)| \leq 1$ özelliğine sahiptir.

(38)'deki çözümün bir yan ürünü, her adımda en küçük toplam hata E_i 'nin hesaplanmasıdır. En küçük hata E_i 'nin öngörücünün derecesi artarken azaldığı (veya aynı kaldığı) kolaylıkla gösterilebilir. Doğal olarak, E_i karesel bir hata olduğundan hiç bir zaman negatif olmaz. Bu yüzden,

$$0 \leq E_i \leq E_{i-1}, \quad E_0 = R(0). \quad (40)$$

almalıyız. Eğer özilişki katsayıları (39)'daki gibi normalize edilirse o zaman en küçük hata E_i de $R(0)$ ile bölünmüş olur. Böylece ortaya çıkan büyüklüğe normalize edilmiş hata V_i :

$$V_i = \frac{E_i}{R(0)} = 1 + \sum_{k=1}^i a_k r(k). \quad (41)$$

diyeceğiz. (40)'a göre

$$0 \leq V_i \leq 1, \quad i \geq 0. \quad (42)$$

olduğu açıktır.

Ayrıca (38d) ile (41)'e göre, son normalize edilmiş hata V_p

$$V_p = \prod_{i=1}^p (1 - k_i^2). \quad (43)$$

dir. $k_i, 1 \leq i \leq p$ ara büyüklükleri yansımalar katsayıları olarak bilinir. İstatistik literatüründe ise kısmi ilişki katsayıları olarak bilinirler. k_i, s_n ile, $s_{n+1}, \dots, s_{n+i-1}$

sabitlerini içeren $sn+1$ arasındaki (negatif) kısmi ilişki olarak yorumlanabilir. "Yansıma katsayısı" ifadesinin kullanımı iletim hatları kuramından gelir. Bu kurama göre ki, empedansları Z_i ve Z_{i+1} olan iki bölüm arasında bulunan sınırdaki yansıma katsayıları olarak düşünülebilir. Bu durumda ki

$$k_i = \frac{Z_{i+1} - Z_i}{Z_{i+1} + Z_i} \quad (44)$$

ile verilir. Buna göre $H(z)$ geçiş işlevinin, empedans oranları (44)'e göre

$$\frac{Z_{i+1}}{Z_i} = \frac{1+k_i}{1-k_i}, \quad 1 \leq i \leq p. \quad (45)$$

olan bu bölümlerden oluşan bir dizinin geçiş işleviymiş gibi düşünülebilir. Aynı açıklama, farklı empedanslı bölümler veya kalın dilimlerden oluşan bir diziden meydana gelen bir ortamda normal doğrultusunda düzlem dalga iletiminin olduğu herhangi bir yer için de verilebilir. Eşit kalınlıklı p bölmeden oluşan akustik bir tüp için, empedans oranları ardışık kesit alanlarının ters orantısına indirgenir. Bu gerçek konuşma çözümlemesinde son zamanlarda kullanılmaya başlanmıştır.

2) Sonuca Adım-Adım Yaklaşan Çözümler:

Takım halindeki doğrusal eşitliklerin çözümlenmesi için, tek adımda çözüm yöntemlerinden başka sonuca adım-adım yaklaşan bir kaç yöntem vardır. Bu yöntemlerde çözümlemeye, çözüm için bir ilk tahmin yapılarak başlanır. Bundan sonra çözüm, çoğunlukla bazı hata ölçütlerinin

değişim ölçülerine dayanan bir düzeltme terimi eklenerek güncellenir. Genel olarak, adım-adım yaklaşmalı yöntemler, istenilen bir yakınsaklık derecesini gerçekleştirmek için tek adımda çözüm yöntemlerinden daha fazla hesaplama gerektirirler. Bununla beraber bazı uygulamalarda çoğu kez, sadece bir kaç adımda çözüme ulaşabilen iyi bir ilk tahmin vardır. Bu, eğer eşitlik sayısı fazla ise tek adımda çözüm yöntemlerine göre büyük bir tutum sağlayabilir. Adım-adım yaklaşmalı yöntemlerin bazıları değişim ölçüsü yöntemi, en dik iniş yöntemi, Newton yöntemi, birleşik değişim ölçüsü yöntemi ve stokastik yaklaşım yöntemidir.

Şimdiye kadar, çözümlenecek olan imin bütünüyle elimizde olduğunu kabul ettik. Bazı gerçek-zamanlı uygulamalar için hesaplamaları im geliyor iken yapmayı mümkün kılmak yararlıdır. Çözümün, imin her yeni gözlemine dayanılarak güncellendiği uyarlamalı düzenler vardır. Bu düzenlerde güncelleme, çoğunlukla yeni gözlem ile o anki çözümün verdiği öngörölmüş değer arasındaki fark ile orantılıdır. Uyarlamalı işlemler için diğer bir uygulama, çözümün bütün veri çözümlenmeden önce kısa sürede yakınsayabileceği çok uzun veri kayıtlarının işlenmesindedir.

3) Süzgeç Kararlılığı:

$H(z)$ 'in kararlılığını sınamak için kullanılabilecek bir yöntem, bütün ardışık hataların pozitif olup olmadıklarını denetlemektir. Gerçekte $Ei > 0, 1 < i < p$ koşulu $H(z)$ 'in

kararlılığı için gerek ve yeter koşuldur. (38d) ile (40)'a göre $H(z)$ 'in kararlılığı için eşdeğer bir koşulun

$$|k_i| < 1, \quad 1 \leq i \leq p. \quad (46)$$

olduğu açıktır. Bu yüzden (38)'deki yinelemeli işlem $H(z)$ süzgecinin kararlılığının denetlenmesini de kolaylaştırır. Kararlılık ile ilgili daha ayrıntılı bilgiler [2]'de bulunabilir.

3.1.2.4 En İyi Kutup Sayısı:

Tüm-kutuplu modellerin uygunlaştırılmasında, çoğunlukla kutup sayısı için bir "en iyi" değerin belirlenmesi gerekir. Buradaki "en iyi" kelimesini tanımlamak önemsiz bir iş değildir. Modelin uygunluğu, kutupların sayısı p artarken "iyileşir". Problem nerede durulacağıdır. Hem hesaplamayı azaltmak hem de kötü-şartlanma [2], olasılığını en küçültmek için p 'nin en küçük değerini istediğimiz açıktır.

Eğer im görüngesi p_0 kutuplu bir tüm-kutuplu görünge ise, o zaman $V_p = V_{p_0}$, $p > p_0$ ve $k_p = 0$, $p > p_0$ 'dır, yani hata eğrisi $p > p_0$ için düz kalır. Bu yüzden eğer im görüngesinin tüm-kutuplu bir görünge olmasını bekliyorsak, en iyi p 'yi elde etmek için basit bir yol, hata eğrisinin düz olduğu anı gözlemlemektir. Ancak modellenen im , beyaz gürültü ile uyarılan p_0 -kutuplu bir süzgecin çıkışı ise, o zaman önerilen yol işe yaramayacaktır, çünkü model parametrelerinin kestirimi veri noktalarının sonlu bir sayısına dayandığından hata eğrisi $p > p_0$ için düz kalmayacaktır. Bununla birlikte uygulamada hata eğrisi

$p > p_0$ için hemen hemen düz kalır. Bu, aşağıdaki eşik sinamasının kullanılması düşüncesini verir

$$1 - \frac{V_{p+1}}{V_p} < \delta. \quad (47)$$

Bu sinama hata eğrisinin gerçekten düzelmiş olduğundan emin olmadan önce çeşitli ardışık değerler için gerçekleştirilmelidir.

En iyi p 'nin belirlenmesi için literatürde önerilen diğer bir yol, ortalama log-olasılığın kestiriminde kullanılan bir enformasyon kuramı ölçütüne dayanmaktadır. Bu ölçüt

$$I(p) = -2 \log(\text{en büyük olasılık}) + 2p \quad (48)$$

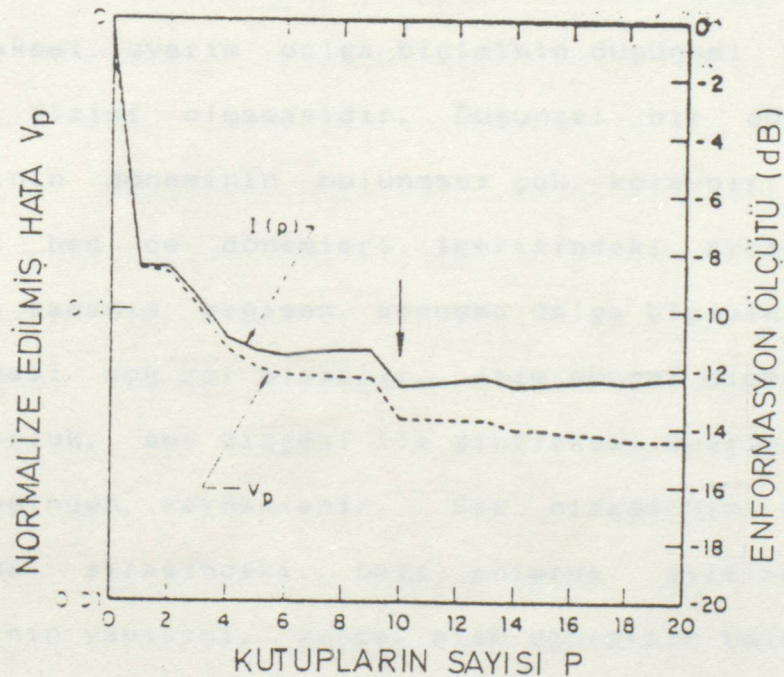
ile verilir. p 'nin $I(p)$ 'yi en küçük yapan değeri en iyi değer olarak alınabilir. Tüm-kutuplu modelleme probleminde, eğer imin Gaussian olasılık dağılımına sahip olduğu kabul edilirse, o zaman (48) eşitliği (toplamsal sabitler ihmal edilip N ile bölünerek)

$$I(p) = \log V_p + \frac{2p}{N_e} \quad (49)$$

eşitliğine indirgenir. Bu eşitlikteki N_e imdeki veri noktalarının etkin sayısıdır. "Etkin" kelimesi, pencereleme yapılmış olması halinde dengeleme yapılması gerektiğini belirtmek için kullanılmıştır. Bir pencerenin etkin genişliği, o pencerenin enerjisinin aynı genişlikteki bir dikdörtgen pencerenin enerjisine olan oranı ile belirlenir. Örneğin Hamming penceresi için $N_e = 0.4 N$ 'dir.

Dikkat edilirse (49)'daki ilk terim p 'nin bir işlevi olarak azalırken ikinci terim artar. Bu yüzden $l(p)$ 'nin en küçük olduğu bir p değeri vardır. Uygulamada, çoğunlukla bir kaç yerel en küçük ortaya çıkar. Bu durumda $l(p)$ 'nin mutlak en küçüğüne karşılık gelen p değeri en iyi değer olarak alınır. Genellikle $l(p)$ ilgili en büyük değere kadar (güvenli bir kestirim için $p_{max} < 3N/2$ alınması önerilmektedir) hesaplanır ve $l(p)$ 'nin en küçüğü bu bölge içinde bulunur.

Şekil 3.1.2'de (48)'deki enformasyon ölçütü ile yapılmış bir uygulama örneği görülmektedir [2]. Noktalı eğri alışılmış hata eğrisidir. Birleşik eğri ise, (49)'daki $l(p)$ 'nin, $10 \log_{10} e$ ile çarpılarak DB'ye çevirildikten sonraki çizimidir. Ok ile işaretli $p=10$ noktası, $l(p)$ 'nin en küçük olduğu en iyi p değeridir.



Şekil 3.1.2 (49)'daki $l(P)$ 'nin p ile değişimi [2].

3.2 Atım Bulma

3.2.1 Giriş

Atım Bulucu, konuşma işleme dizgelerinin temel bileşenlerinden birisidir. Konuşmanın atım yapısı, konuşma üretimi için uyarım kaynağının doğası ile ilgili çok önemli bilgiler sağlamasından başka konuşmacı tanıma ve işitme engellilerin konuşma ile eğitimi için çok kullanışlıdır. Ayrıca konuşmanın atım yapısının bilinmesi, hemen hemen bütün konuşma çözümleme-bireşimleme (vocoder) dizgelerinde bir gerekliliktir. Atım bulmanın öneminden dolayı, konuşma işleme literatüründe çok çeşitli atım bulma algoritmaları geliştirilmiştir.

3.2.1.1 Atım Bulmadaki Sorunlar

Konuşma iminin atım döneminin akustik dalga biçiminden doğru ve güvenli olarak ölçümü, çeşitli nedenlerle, çoğu zaman son derecede zordur. Bu nedenlerden birisi gırtlaksal uyarım dalga biçiminin düşünsel bir dönemli darbe dizisi olmamasıdır. Düşünsel bir dönemli dalga biçiminin döneminin bulunması çok kolaydır, ancak hem dönemi hem de dönemleri içerisindeki ayrıntılı dalga biçimi zamanla değişen, konuşma dalga biçiminin döneminin ölçülmesi çok zor olabilir. Atım dönemi ölçmedeki ikinci bir zorluk, ses dizgesi ile gırtlaksal uyarım arasındaki etkileşimden kaynaklanır. Ses dizgesinin formantları, konuşma sırasındaki bazı anlarda gırtlaksal dalga biçiminin yapısını, gerçek atım döneminin bulunmasını çok zorlaştıracak şekilde, aşırı olarak değiştirebilir. Böyle

etkileşimler genellikle en çok, konuşma belirleyicilerin, formantların da hızla değişmesine neden olan hızlı hareketleri sırasında atım bulunmasını kötü yönde etkiler. Güvenilir atım ölçümündeki üçüncü bir sorun, sesli konuşma bölümleri sırasındaki her atım döneminin tam başlangıç ve bitişinin tanımlanmasındaki doğal zorluktur. Atım döneminin tam başlangıç ve bitiş yerlerinin seçimi çoğu kez bütünüyle keyfidir. Atım bulmadaki dördüncü bir zorluk sessiz konuşma ile düşük-seviyeli sesli konuşma arasındaki ayırımın güçlüğüdür. Çoğu hallerde, sessiz konuşma bölümleri ile düşük-seviyeli sesli konuşma bölümleri arasındaki geçişler çok belirsizdir ve bu yüzden yerlerini kesin olarak belirlemek çok zordur.

3.2.1.2 Atım Bulucu Türleri

Atım ölçmelerindeki sayısız zorlukların bir sonucu olarak, çok sayıda karmaşık atım bulma yöntemleri geliştirilmiştir. Atım bulucu, temel olarak sesli-sessiz ayırımı yapan, ve sesli konuşma dönemleri sırasında atım döneminin ölçümünü veren bir aygıttır. Bununla beraber bazı atım bulma algoritmaları sadece, sesli konuşma bölümleri sırasında atım belirler ve sesli-sessiz ayırımını diğer bazı yöntemlere bırakırlar. Atım bulma algoritmaları, kabaca aşağıdaki üç yaygın sınıfa ayrılabilir.

- 1) Konuşma imlerinin, çoğunlukla zaman tanım kümesi özelliklerini kullananlar.
- 2) Konuşma imlerinin, çoğunlukla frekans tanım kümesi

özelliklerini kullananlar.

3) Konuşma imlerinin hem zaman hem de frekans tanım kümesi özelliklerini kullananlar.

Zaman tanım kümesi atım bulucuları, atım dönemini kestirmek için doğrudan doğruya konuşma dalga biçimi üzerinde işlem yaparlar. Bu atım bulucuları için en sık yapılan ölçümler tepe ve çukur ölçümleri, sıfır-geçiş ölçümleri, ve özilişki ölçümleridir. Bütün bu hallerde yapılan temel kabul, eğer dönemli-benzeri bir im, formant yapısının etkilerini en küçültmek için uygun bir şekilde işlenmiş ise, bu durumda basit zaman tanım kümesi ölçümlerinin atım döneminin iyi bir kestirimini vereceği kabulüdür.

Frekans tanım kümesi sınıfı atım bulucuları, imin zaman tanım kümesinde dönemli olması halinde frekans görüngesinin, ana frekans ve harmoniklerindeki dürtülerden oluşan bir diziden meydana gelmesi özelliğini kullanırlar. Bu yüzden imin atım dönemini kestirmek için frekans görüngesi (veya kepsral atım bulucuda olduğu gibi frekans görüngesinin doğrusal olmayan dönüştürülmüş bir türü) üzerinde basit ölçümler yapılabilir.

Karma atım bulucuları sınıfı zaman tanım kümesi ve frekans tanım kümesi atım bulma yaklaşımlarının özelliklerini birleştirir. Örneğin karma bir atım bulucu görüngesel olarak düzeltilmiş bir zaman dalga biçimi sağlamak için frekans tanım kümesi yöntemlerini, ve daha sonra atım dönemini kestirmek için de özilişki ölçümlerini

kullanabilir. Şekil 3.2.1 de Sondhi'nin

3.2.2 Atım Bulma Algoritmaları

Yaygın olarak kullanılan atım bulma algoritmalarından bazıları şöyle sıralanabilir:

- 1) Kenetleme kullanan değiştirilmiş özilişki yöntemi (AUTOC).
- 2) Kepstrum yöntemi (CEP).
- 3) Basitleştirilmiş ters süzme yöntemi (SIFT).
- 4) Veri indirgeme yöntemi (DARD).
- 5) Koşut işleme yöntemi (PPROC).
- 6) Newton dönüşümü kullanan LPC görüngesel eşitleme yöntemi (LPC).
- 7) Ortalama büyüklük fark işlevi (AMDF).

Bu çalışmada atım bulma için AUTOC yöntemi kullanılmıştır. Bu yüzden sadece bu yöntemi incelemekle yetineceğiz.

3.2.3 Değiştirilmiş Özilişki Yöntemi (AUTOC)

Daha önce de söylendiği gibi, atım döneminin bulunmasındaki en büyük sorunlardan birisi, gırtlaksal uyarım ile ses dizgesi arasındaki etkileşimden kaynaklanır. Bu yüzden güvenli bir atım bulma için formantların etkisinin çok zayıflatılması veya olanaklı ise bütünüyle giderilmesi sağlanmalıdır. Bu işleme görüngesel düzeltme adı verilir. Sondhi [5], bu işlemi gerçekleştirmek için iki yöntem tasarlamıştır: Süzgeç topluluğu yöntemi ve merkez kenetleme yöntemi.

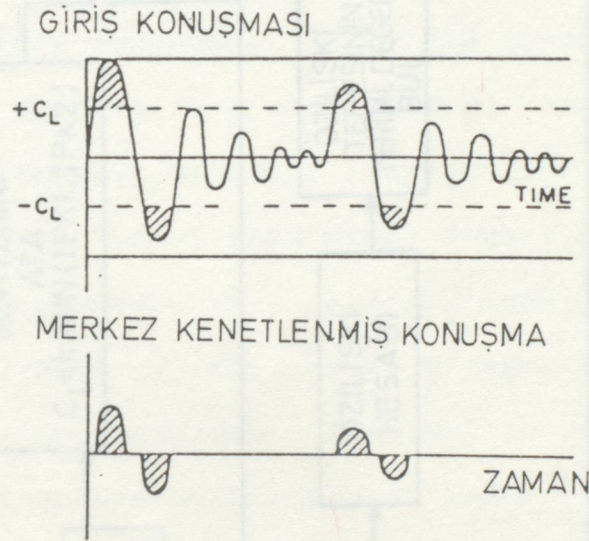
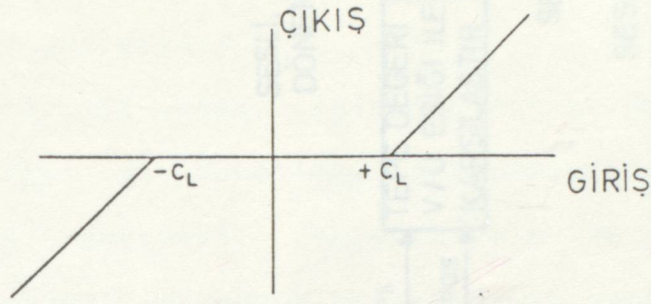
Değiştirilmiş özilişki atım bulucusu Sondhi'nin merkez

kenetleme yöntemine dayanır. Şekil 3.2.1'de Sondhi'nin yönteminde kullanılan merkez kenetleyicinin giriş-çıkış işlevi ve tipik çalışma örneği görülmektedir. Bu kenetleyici girişindeki imin, kenetleme seviyesinin altında kalan değerlerini sıfıra eşitlemekte ve kenetleme seviyesinin üzerinde kalan değerlerini ise kenetleme seviyesi kadar sıfıra doğru kaydurmaktadır. Şekil 3.2.1'den de görülebileceği gibi, eğer kenetleme seviyesi uygun olarak ayarlanırsa, konuşma dalga biçiminde formantların etkisiyle ortaya çıkan bileşenlerin bütünüyle giderilmesi olanaklıdır. Bu yüzden merkez kenetleyici, çıkışında görüngesel olarak etkin bir şekilde düzeltilmiş bir im verir.

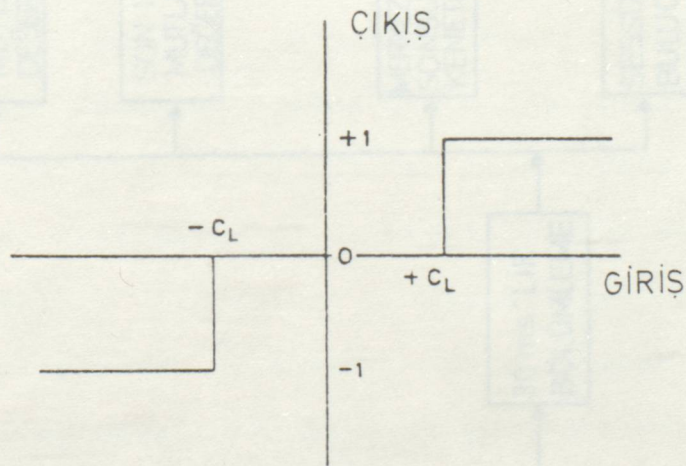
Değiştirilmiş özilişki yönteminde, özilişki hesaplarını kolaylaştırmak için merkez kenetlemesiyle birlikte sonsuz tepe kenetlemesi de kullanılır. Böylece ortaya çıkan kenetleyicinin giriş-çıkış işlevi Şekil 3.2.2'de verilmiştir.

Değiştirilmiş özilişki algoritmasının blok çizgesi Şekil 3.2.3'de görülmektedir. Bu yöntem, girişine uygulanan konuşma iminin 900 Hz'lik bir AGS'den geçirilmesini gerektirir. (AGS olarak 99-noktalı, doğrusal fazlı, FIR tipi bir sayısal süzgeç kullanılır.)

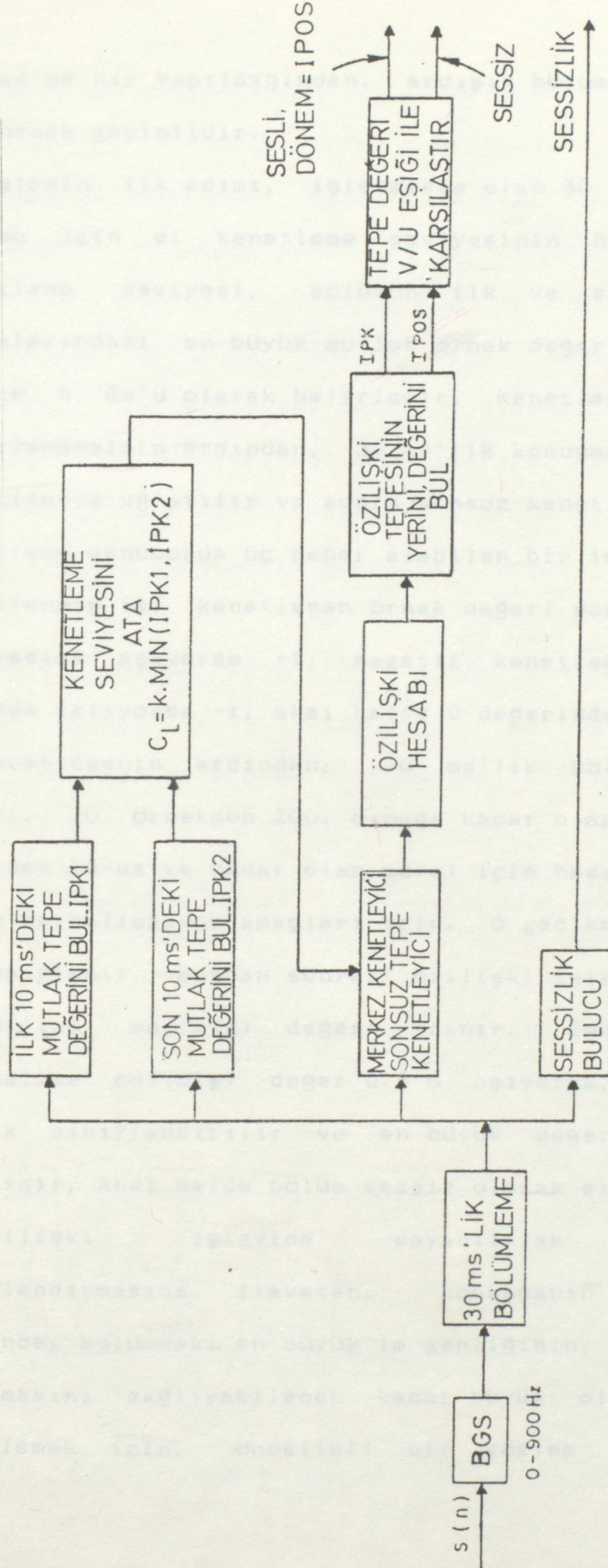
AGS'den geçirilen konuşma imi, 10 kHz'lik örnekleme oranında sayısallaştırılır ve işlenmek üzere 30-ms'lik (300 örneklik) geçişli bölümlere ayrılır. Atım dönemi hesabı bütün atım bulucularında saniyede 100 kez, yani her



Şekil 3.2.1 Merkez kenetleyicinin giriş-çıkış karakteristiği ve tipik çalışması [4], [5].



Şekil 3.2.2 Merkez kenetleyici, sonsuz tepe kenetleyici bileşiminin giriş-çıkış karakteristiği [4].



Şekil 3.2.3 AUTOC atım bulucunun blok çizgesi [3], [4]

10 ms'de bir yapıldığından, ardışık bölümler 20 ms veya 200 örnek geçişlidir.

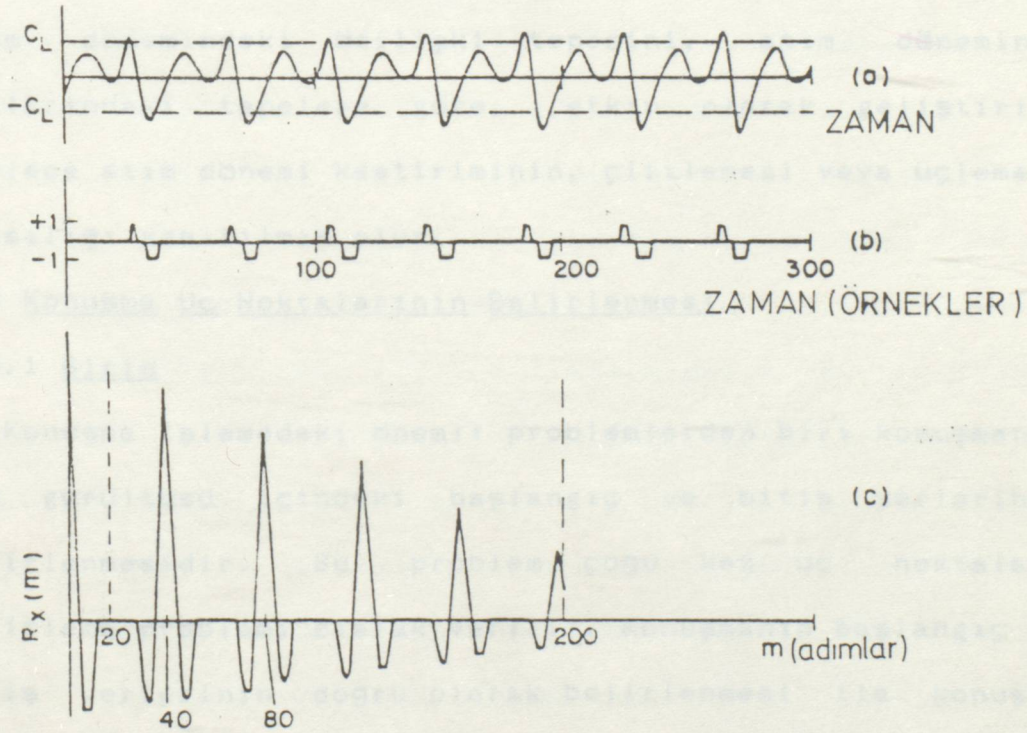
İşlemin ilk adımı, işlenmekte olan 30 ms'lik konuşma bölümü için c1 kenetleme seviyesinin hesaplanmasıdır. Kenetleme seviyesi, bölümün ilk ve son 10 ms'lik kısımlarındaki en büyük mutlak örnek değerlerinden küçük olanın % 64'ü olarak belirlenir. Kenetleme seviyesinin belirlenmesinin ardından, 30 ms'lik konuşma bölümü merkez kenetlemeye ugratılır ve sonra sonsuz kenetleme uygulanır. Kenetleme sonucunda üç değer alabilen bir im ortaya çıkar. Kenetlenmiş im, kenetlenen örnek değeri pozitif kenetleme seviyesini aşıyorsa +1, negatif kenetleme seviyesinin altında kalıyorsa -1, aksi halde 0 değerindedir.

Kenetlemenin ardından, 30 ms'lik bölümün özilişki işlevi, 20. örnekten 200. örneğe kadar olan adımlar (yani 2-ms'den 20-ms'ye kadar olan süre) için hesaplanır. Ayrıca uygun normalizasyon amaçları için, 0 gecikmedeki özilişki de hesaplanır. Bundan sonra, özilişki işlevinin en büyük (normalize edilmiş) değeri aranır. Eğer en büyük (normalize edilmiş) değer 0.3'ü aşıyorsa, bölüm sesli olarak sınıflandırılır ve en büyük değerin yeri atım dönemidir. Aksi halde bölüm sessiz olarak sınıflandırılır.

Özilişki işlevine dayanılarak sesli-sessiz sınıflandırmasına ilaveten, konuşmanın her bölümü üzerinde, bölümdeki en büyük im genliğinin, atım hesabının yapılmasını sağlayabilecek kadar büyük olup olmadığını belirlemek için, öncelikli bir gözlem yapılır. Eğer

bölümdeki en büyük im genliği, verilen bir eşeğin altında ise, bölüm sessizlik olarak sınıflandırılır ve hiç bir atım hesabı uygulanmaz. Sessizlik eşeği olarak, genellikle konuşmadaki en büyük mutlak im değerin 1/15'i seçilir. Sessizlik eşeği, 50 veya 100 ms'lik bir sessizlik kaydındaki en büyük mutlak örnek değeri olarak da alınabilir. Bu çalışmada, sezgisel olarak daha güvenli bir sessizlik eşeği vereceği düşüncesiyle, sessizlik eşeği için yukarıdaki iki yoldan iki ayrı değer bulunmuş ve sonra bunların aritmetik ortalaması sessizlik eşeği olarak kullanılmıştır.

Şekil 3.2.4'de tipik bir 30 ms'lik konuşma bölümü için işleme örneği verilmiştir [4]. En üstte, AGS'den geçirilmiş dalga biçimi ve bu örnek için hesaplanmış +-CL



Şekil 3.2.4 Kenetlenmiş konuşma örneği ve özilişki işlevi [4].

kenetleme seviyeleri görülmektedir. Ortada kenetlenmiş konuşma, ve en altda da kenetlenmiş konuşmanın özilişki işlevi görülmektedir. Atım döneminin çoğunlukla bulunduğu saha, $m=20$ ile $m=200$ 'deki noktalı düşey hatlar ile gösterilmiştir. Bu örnek için, bölümün atım dönemi yaklaşık 40 örnek, veya 4 ms'dir. Bu da 250 Hz'lik bir atım frekansına karşılık gelir.

Özilişki işlevinin hesaplanmasında, işlenmekte olan 30 ms'lik bölümün dışındaki örneklerin sıfır kabul edildiğine dikkat edilmelidir. Bu kabul, özilişki işlevinin, $m=0$ 'da 1'den başlayıp $m=300$ 'de 0'a giden, doğrusal bir azalma ile etkin bir şekilde ağırlıklandırılmasına neden olur. Bu etki, özilişki işlevinin tepelerinin doğrusal olarak sifıra doğru azaldığı Şekil 3.2.4'de açıkça görülmektedir. Özilişki işlevi üzerinde doğrusal bir azalmanın kullanımı, atım dönemindeki özilişki tepesini, atım döneminin katlarındaki tepelere göre, etkin olarak geliştirir. Böylece atım dönemi kestiriminin, çiftlemesi veya üçlemesi olasılığı azaltılmış olur.

3.3 Konuşma Uç Noktalarının Belirlenmesi

3.3.1 Giriş

Konuşma işlemedeki önemli problemlerden biri konuşmanın fon gürültüsü içindeki başlangıç ve bitiş yerlerinin belirlenmesidir. Bu problem çoğu kez uç noktaları belirleme problemi olarak verilir. Konuşmanın başlangıç ve bitiş yerlerinin doğru olarak belirlenmesi ile konuşma verisi işleme tutarı, önemli değerlerde azaltılabilir.

Ayrıca konuşma tanıma gibi, bazı konuşma işleme uygulamaları, tanımları gereği uç noktaları probleminin çözülmesini gerektirirler.

Konuşmayı fon gürültüsünden ayırma işi, yüksek nitelikli kayıtların yapılabilirdiği son derecede yüksek im-gürültü oranlı, örneğin yankısız bir bölme veya ses geçirmeyen bir oda gibi akustik çevreler dışında önemsiz bir iş değildir. Böyle yüksek im-gürültü oranlı çevrelerde, en düşük seviyeli konuşma seslerinin (örneğin, güçsüz sessizler, düşük seviyeli sesli kısımlar, v.b. gibi) enerjisi bile fon gürültüsü enerjisini aştığından, uç noktalarının belirlenmesi için basit bir enerji ölçümü yeterli olur. Bununla birlikte, böyle düşünsel kayıt koşulları, konuşma işleme dizgelerinin gerçekteki uygulamaları için kullanışlı değildir. Bu yüzden basit enerji ölçümleri, güçsüz sessizleri (örneğin "four" 'daki /f/ gibi) fon gürültüsünden ayırmak için yeterli değildir. Bu çalışmada konuşma başlangıç ve bitişini belirlemek için, im-gürültü oranı en az 30 DB olan hemen hemen her çevrede iyi sonuç veren, oldukça basit bir algoritma [7], kullanılmıştır. Bu algoritma konuşma üzerinde yapılan iki ölçüme dayanmaktadır. Bu ölçümler kısa-süreli enerji ve sıfır geçiş oranı ölçümleridir. Algoritma, karar verme ölçütleriyle ilgili olan bütün eşik değerlerini, doğrudan doğruya kayıt edilen süre üzerinde yapılan ölçümlerden elde ettiğinden, akustik çevreye bir dereceye kadar kendi kendini uydurma

özelliğine sahiptir.

3.3.2 Uç Noktalarının Belirlenmesindeki Sorunlar

Uç noktalarının belirlenmesinde zorluklar ile karşılaşıl原因 durumları aşağıdaki gibi sınıflandırabiliriz.

- (1) Konuşmanın başlangıç veya bitişindeki güçsüz sessizler (/f,th,h/).
- (2) Güçsüz patlayıcı (veya basınç birikimli) sesler (/p,b,k/).
- (3) Konuşma sonlarındaki burunsallar.
- (4) Kelime bitişlerindeki seslendirilmemiş sesli sessizler.
- (5) Bazı sesli seslerin, kelime bitişlerindeki kesimleri, örneğin "three" (/th-r-i/) veya "binary" (/b-ai-n-e-r-i/) kelimelerindeki sondaki /i/ sesi bazen sessiz olur.

Konuyla ilgili açıklayıcı bilgiler [7]'de bulunabilir.

3.3.3 Uç Noktaları Belirleme Algoritması

Bu çalışmada kullanılan algoritma iki basit ölçüme dayanmaktadır. Bu ölçümler enerji ve sıfır geçiş oranı ölçümleridir. Enerji ve sıfır geçiş oranı ölçümleri, hem kolaydır hem de hızlı olarak yapılabilir. Bu iki ölçüm konuşmanın varlığı veya yokluğu ile ilgili oldukça doğru (aynı zamanda da korunumlu) bilgiler verebilir. Algoritmayı açıklamaya geçmeden önce, enerji ve sıfır geçiş oranının nasıl ölçüldüğünü tanımlayacağız. Konuşma "enerjisi" $E(n)$, konuşmanın 10 ms'lik bölümlerindeki örnek büyüklüklerinin, ölçüm aralığına merkezlenen toplamı

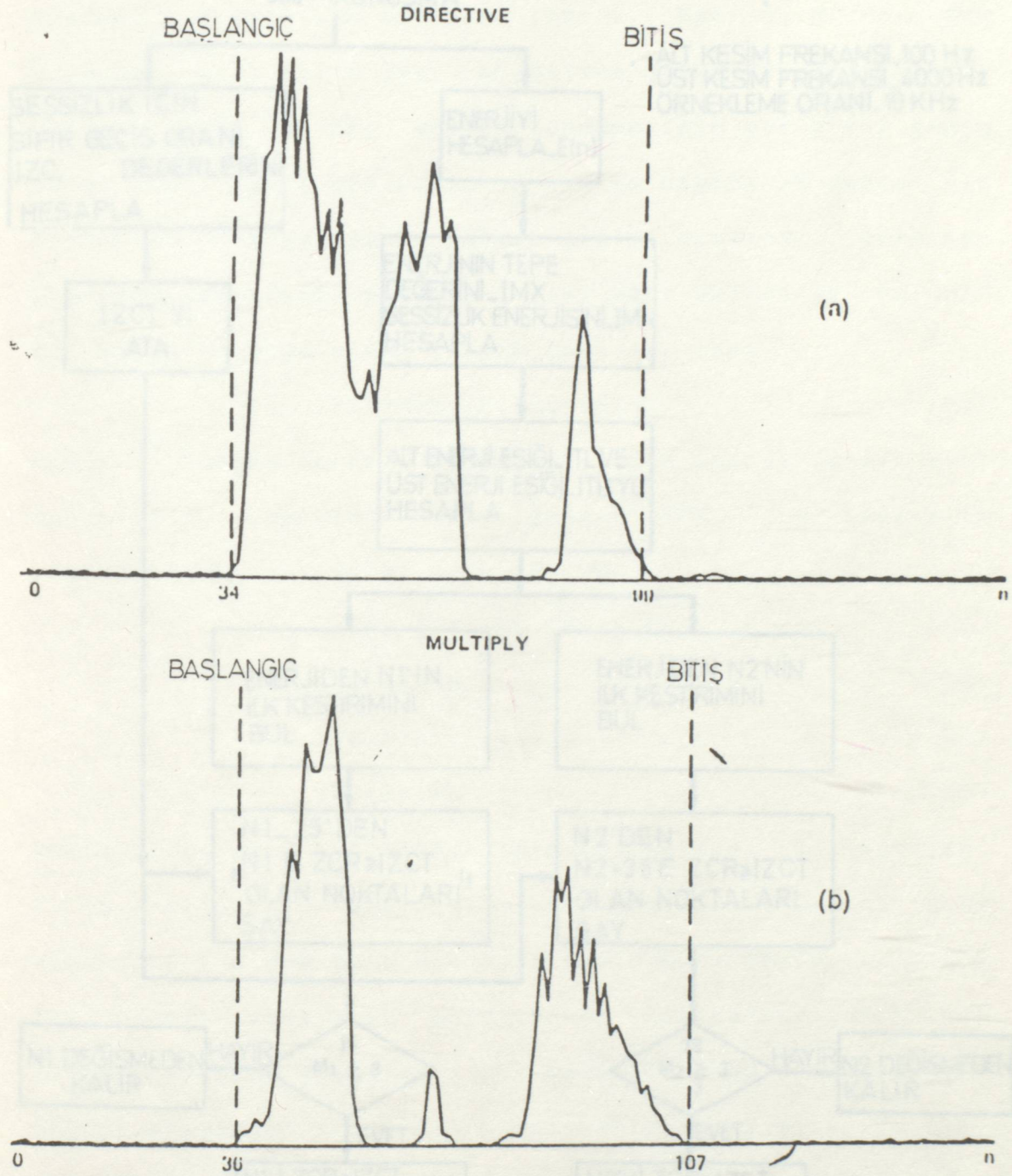
olarak tanımlanır, yani

$$E(n) = \sum_{i=-60}^{60} |s(n+i)|, \quad (50)$$

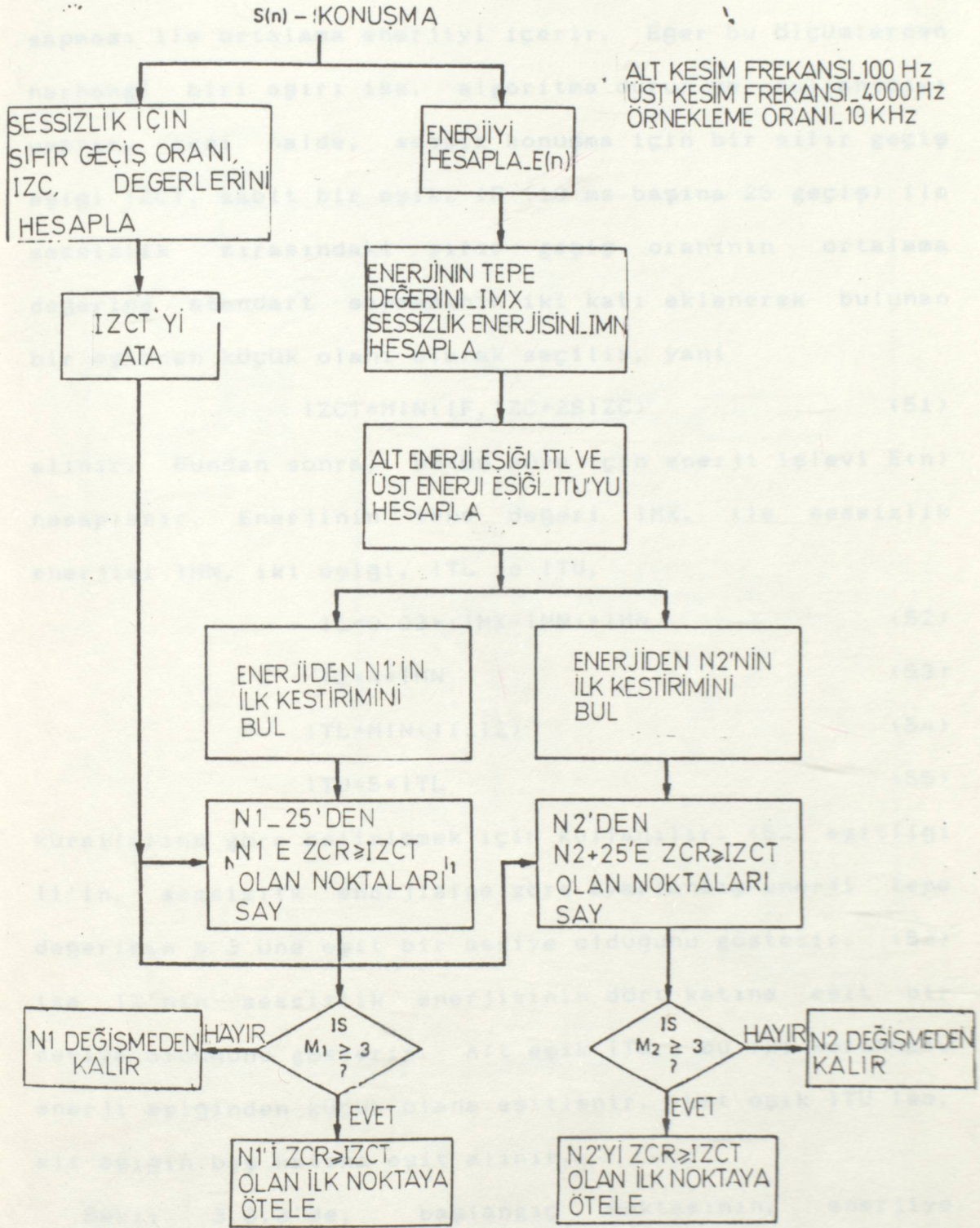
dir. Burada $s(n)$, 10 kHz'de örneklendiği düşünülen konuşma örnekleridir. Enerji hesabı için 10 ms'lik bir pencerenin alınmasının ve karesel büyüklüğün yerine doğrudan büyüklüğün kullanılmasının nedeni, hesaplamaların tamsayılarla yapılmasını olası kılmak ve böylece hesaplama hızını arttırmaktır. Ayrıca, doğrudan büyüklüğün kullanılması, büyük-genlikli konuşma değişimlerinin önemini azaltır ve daha düzeltilmiş bir enerji işlevi verir. Şekil 3.3.1'de "directive" ve "multiply" kelimeleri için tipik enerji işlevleri görülmektedir [7]. (Bu kelimelerin başlangıç ve bitişi bu enerji çizimleri üzerinde işaretlenmiştir.) Bu örnek için enerji işlevi her 10 ms'de bir, veya saniyede 100 kez hesaplanmıştır.

Konuşmanın sıfır (seviye) geçiş oranı $z(n)$, 10 ms süre başına sıfır (seviye) geçiş sayısı olarak tanımlanır. Sıfır geçiş oranı 50-Hz'lik hum gürültüsü, dc kayma, v.b.'den çok etkilendiği halde, sessiz seslerin varlığı veya yokluğu için çoğu kez oldukça iyi bir ölçüdür.

Şekil 3.3.2'de uç noktaları belirleme algoritmasının akış çizeneği görülmektedir. Konuşma dalga biçimi 10 kHz'de örneklenmeden önce, alt kesim frekansı 100 Hz, üst kesim frekansı 4000 Hz. ve kenar eğimi 48 DB/octave olan bir AGS'den geçirilir. Kayıt süresinin ilk 100 ms'si sırasında konuşmanın olmadığı kabul edilir. Bu yüzden, bu



Şekil 3.3.1 "Directive" ve "Multiply" kelimeleri için tipik enerji eğrileri [7].



Şekil 3.3.2 Uç noktaları bulma algoritmasının akış çizeneği [7].

süre sırasında fon gürültüsünün istatistikleri ölçülür. Bu ölçümler, sıfır geçiş oranının ortalama ve standart sapması ile ortalama enerjiyi içerir. Eğer bu ölçümlerden herhangi biri aşırı ise, algoritma durur ve kullanıcıyı uyarır. Aksi halde, sessiz konuşma için bir sıfır geçiş eşiği IZCT, sabit bir eşik, IF (10 ms başına 25 geçiş) ile sessizlik sırasındaki sıfır geçiş oranının ortalama değerine standart sapmasının iki katı eklenerek bulunan bir eşikten küçük olanı olarak seçilir, yani

$$IZCT = \min(IF, IZC + 2SIZC) \quad (51)$$

alınır. Bundan sonra, bütün süre için enerji işlevi $E(n)$ hesaplanır. Enerjinin tepe değeri IMX, ile sessizlik enerjisi IMN, iki eşiği, ITL ve ITU,

$$I1 = 0.03 * (IMX - IMN) + IMN \quad (52)$$

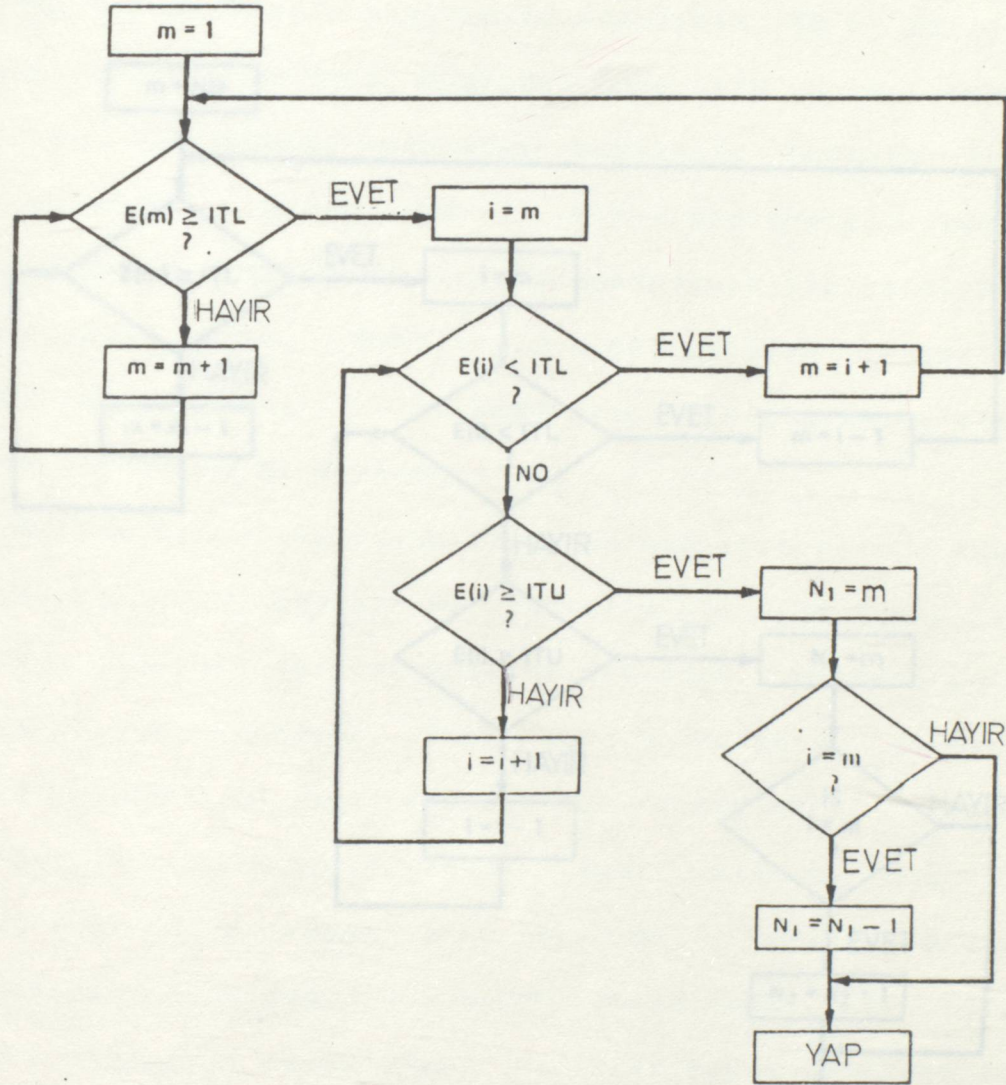
$$I2 = 4 * IMN \quad (53)$$

$$ITL = \min(I1, I2) \quad (54)$$

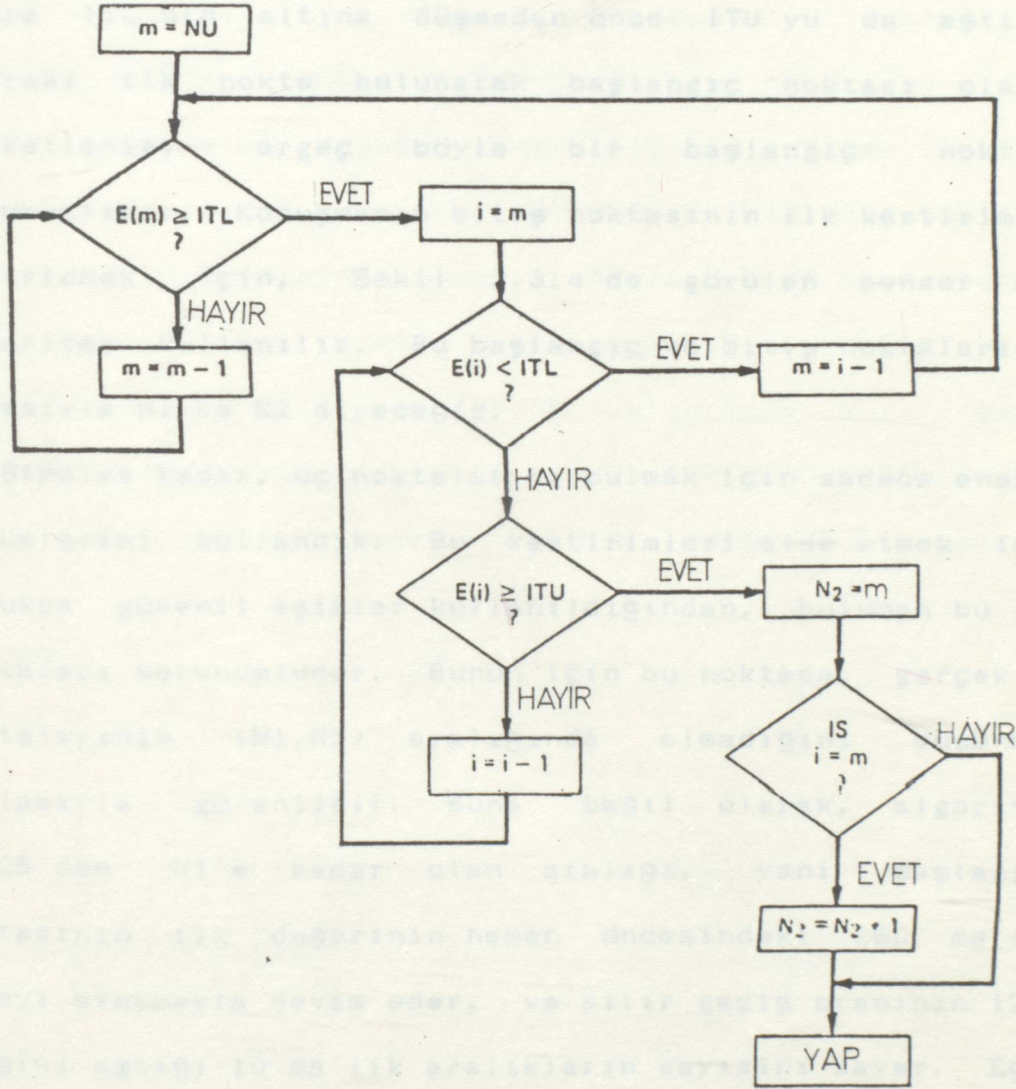
$$ITU = 5 * ITL \quad (55)$$

kurallarına göre belirlemek için kullanılır. (52) eşitliği I1'in, sessizlik enerjisine göre ayarlanmış enerji tepe değerinin % 3'üne eşit bir seviye olduğunu gösterir. (53) ise I2'nin sessizlik enerjisinin dört katına eşit bir seviye olduğunu gösterir. Alt eşik ITL, bu iki korunumlu enerji eşiğinden küçük olana eşitlenir. Üst eşik ITU ise, alt eşiğin beş katına eşit alınır.

Şekil 3.3.3'de, başlangıç noktasının, enerjiye dayanılarak ilk kestirimi için bir algoritma görülmektedir. Bu algoritma, enerji işlevinin, kayıt



Şekil 3.3.3 Enerjiye dayanılarak başlangıç noktasının ilk kestirimi için akış çizeneği [7].



Sekil 3.3.4 Enerjiye dayanılarak bitiş noktasının ilk kestirimi için akış çizeneği (7).

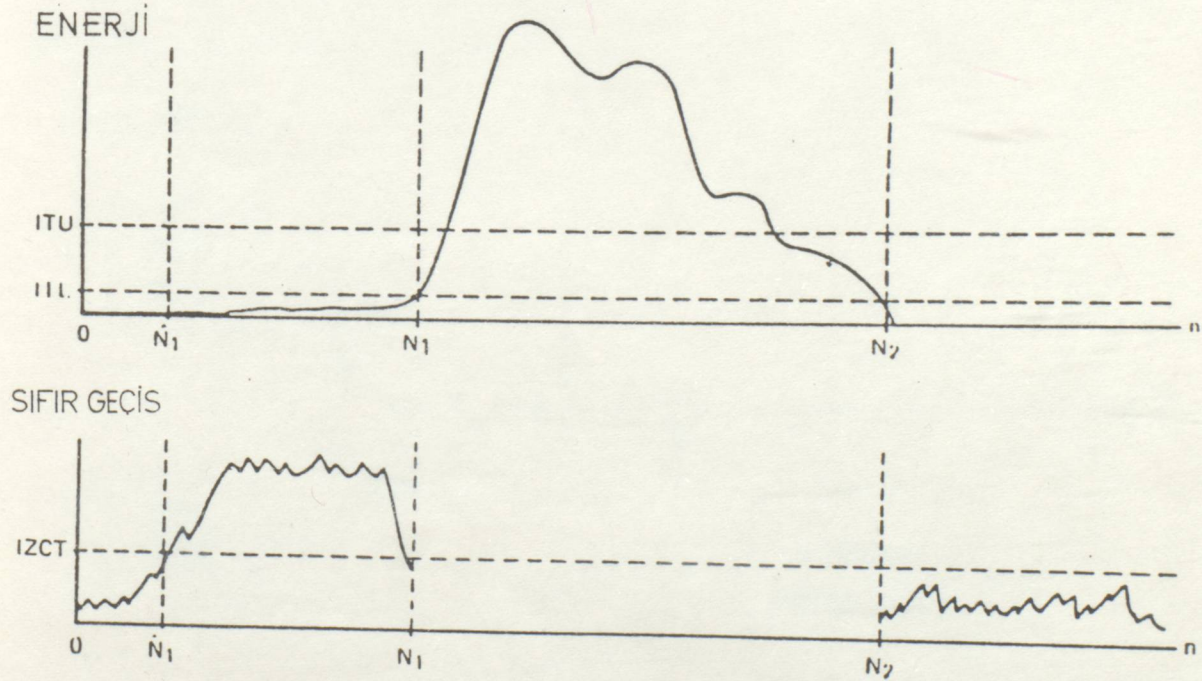
süresinin başlangıcından itibaren alt enerji eşiği aşılanaya kadar araştırılması ile başlar. Enerjinin ITL'yi aştığı ilk nokta, enerji ITU'nun üzerine çıkmadan önce tekrar ITL'nin altına düşmüyorsa konuşma başlangıcının ilk değeri olarak etiketlenir. Aksi halde enerjinin ITL'yi aştığı ve sonra ITL'nin altına düşmeden önce ITU'yu da aştığı, sonraki ilk nokta bulunarak başlangıç noktası olarak etiketlenir; ergeç böyle bir başlangıç noktası bulunmalıdır. Konuşmanın bitiş noktasının ilk kestirimini belirlemek için, Şekil 3.3.4'de görülen benzer bir algoritma kullanılır. Bu başlangıç ve bitiş noktalarına, sırasıyla N1 ve N2 diyeceğiz.

Şimdiye kadar, uç noktalarını bulmak için sadece enerji ölçümlerini kullandık. Bu kestirimleri elde etmek için oldukça güvenli eşikler kullanıldığından, bulunan bu uç noktaları korunumludur. Bunun için bu noktada, gerçek uç noktalarının (N1,N2) aralığında olmadığını düşünmek fazlasıyla güvenlidir. Buna bağlı olarak, algoritma N1-25'den N1'e kadar olan aralığı, yani başlangıç noktasının ilk değerinin hemen öncesindeki 250 ms'lik süreyi sınamakla devam eder, ve sıfır geçiş oranının 1ZCT eşiğini aştığı 10 ms'lik aralıkların sayısını sayar. Eğer 1ZCT eşiğinin aşıldığı aralıkların sayısı üç veya daha fazla ise, başlangıç noktası eşiğin aşıldığı (zamandaki) ilk noktaya ötelenir. Aksi halde, başlangıç noktası N1 olarak bırakılır. Bu uygulamanın arkasındaki mantık, ilgili bütün durumlar için sıfır geçiş oranının güvenli

bir eşiği aşmasının sessiz enerjinin kuvvetli bir göstergesi olmasıdır.

Benzer bir araştırma işlemi, N_2 'den N_2+25 'e kadar olan aralıkta sessiz enerji olup olmadığını belirlemek için konuşmanın bitişiinde kullanılır.

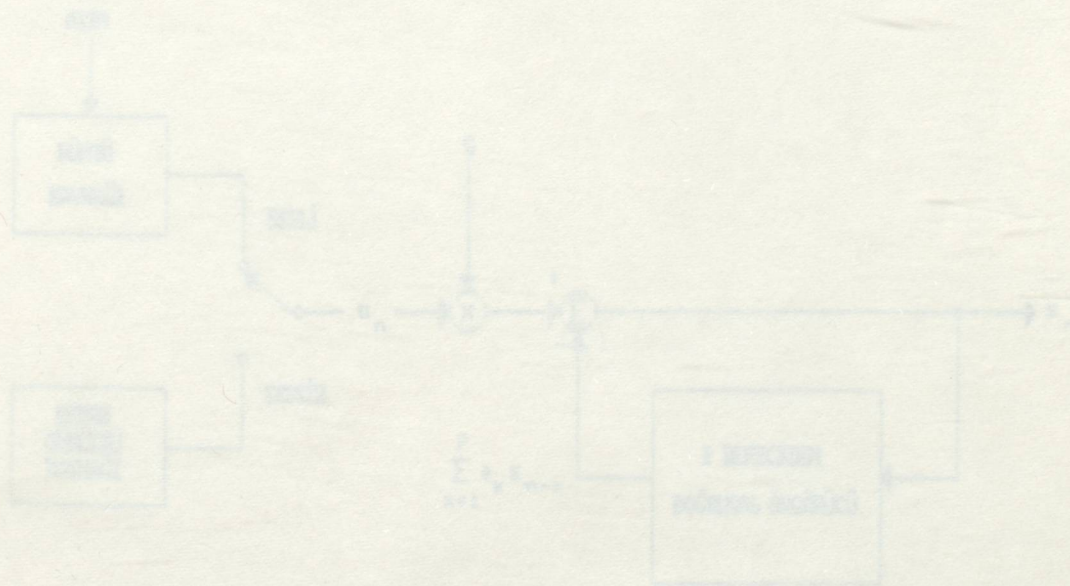
Uç noktaları belirleme algoritmasının kullanılmasını açıklamak için, Şekil 3.3.5'de konuşma için enerji ve sıfır geçişlerin yapısı örnek olarak gösterilmiştir. Algoritma, sadece enerji ölçütü kullanıldığında N_1 noktasını konuşmanın başlangıcı olarak ve N_2 noktasını da konuşmanın bitişi olarak seçer. N_1-25 'den N_1 'e kadar olanın sürenin araştırılmasıyla, algoritma sıfır geçiş oranının eşiği aştığı çok sayıda aralık bulur; bu yüzden başlangıç noktası sıfır geçiş eşiğinin aşıldığı



Şekil 3.3.5 Güçlü bir frikatif ile başlayan bir kelime için enerji ve sıfır geçiş verisinin tipik örneği (7).

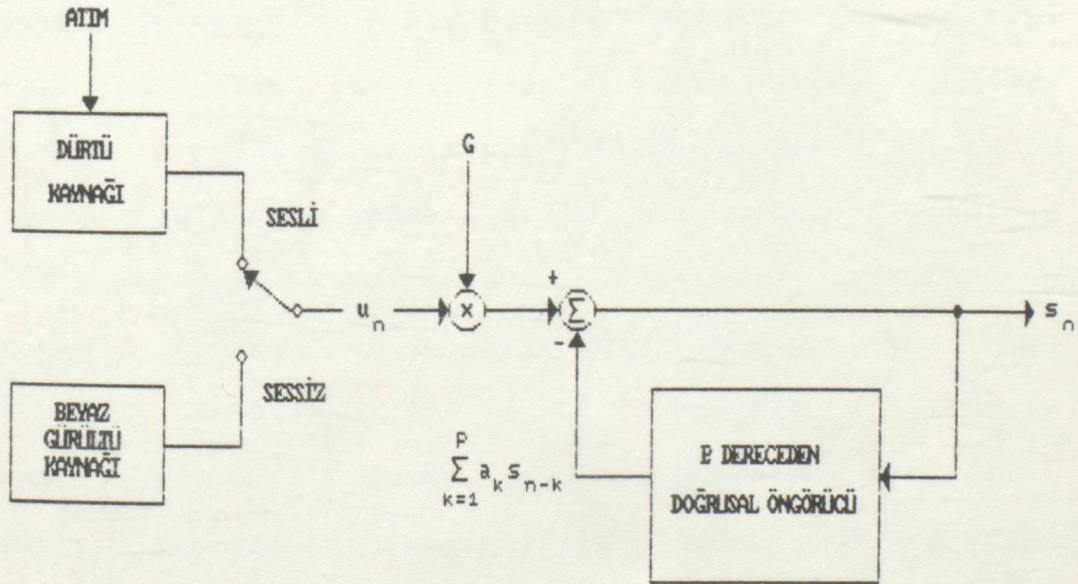
(zamandaki) ilk nokta olan N1'e taşınır. N2'den N2+25'e kadar olan sürenin benzer sınılanması, sıfır geçiş oranı eşigini aşan hiç bir aralık bulunmadığını gösterir; buyüzden, N2 noktası konuşmanın bitiş noktası olarak bırakılır.

İkiye bağlanan denetim parametreleri alın döner. Sesli/sessiz açığının konumu veya konuya bölünmüş. Sesli. G kazanç gerçeği ve ak öngörüde katabilirler. Dürü kaynağı her alın dönerinin başlangıcında birim genlikli bir dürü üretir. Beyaz dürü kaynağı ise her örnekize anında ilgisiz. Birlikli değiller. Standart sapması 1 olan rasgele örnekler üretir. Kaynak seçer: sesli/sessiz açığı ile yapılır. Sesli zaman genliği G kazanç ile ayarlanır. Konuşma süresinin değersizler öngörülen en değeri, birleşimlerle alın N. Örneğini oluşturarak işlemler ile bu ile toplanır.



BÖLÜM 4. KONUŞMA BİREŞİMİ

Konuşma imi çözümlemede kullanılan parametrik model ile bireşimlenir. Doğrusal öngörüne dayanan konuşma bireşimleyicinin blok çizgesi Şekil 4.1'de görülmektedir. Bireşimleyiciye sağlanan denetim parametreleri atım dönemi, sesli-sessiz açkısının konumu veya konuşma bölümünün sınıfı, G kazanç çarpanı ve ak öngörücü katsayılarıdır. Dürtü kaynağı her atım döneminin başlangıcında birim genlikli bir dürtü üretir. Beyaz gürültü kaynağı ise her örnekleme anında ilişkisiz, birbiçimli dağılımlı, standart sapması 1 olan rastgele örnekler üretir. Kaynak seçimi sesli/sessiz açkısı ile yapılır. Uyarım iminin genliği G kazancı ile ayarlanır. Konuşma iminin doğrusalolarak öngörülen s_n değeri, bireşimlenmiş imin n . örneğini oluşturmak için uyarım imi u_n ile toplanır.



Şekil 4.1 Konuşma bireşimleyicinin blok çizgesi

BÖLÜM 5. GÖRÜNGESEL ÇÖZÜMLEME DİZGESİNİN BENZETİMİ

Daha önce de söylenildiği gibi, bu çalışmada yapılan konuşma çözümleme-bireşimleme uygulamasının başarımını inceleyebilmek için, işitme yoluyla öznel bir değerlendirmeden başka görüngenel çözümleme de yapılmıştır.

Görüngenel çözümleme için, [6]'da verilmiş olan bir iki-tabanlı FFT fortran programı önce BASIC diline aktarılmış ve daha sonra bu çalışmanın amaçlarına uygun olarak genelleştirilmiştir. Böylece ortaya çıkan yönerge yapısı, ana benzetim yönergesine bir seçenek olarak eklenmiştir.

Uygulama sonucunda elde edilen bazı görüngenel örnekleri EK-D'de verilmiştir.

BÖLÜM 6. ÇÖZÜMLEME-BİREŞİMLEME DİZGESİNİN BENZETİMİ

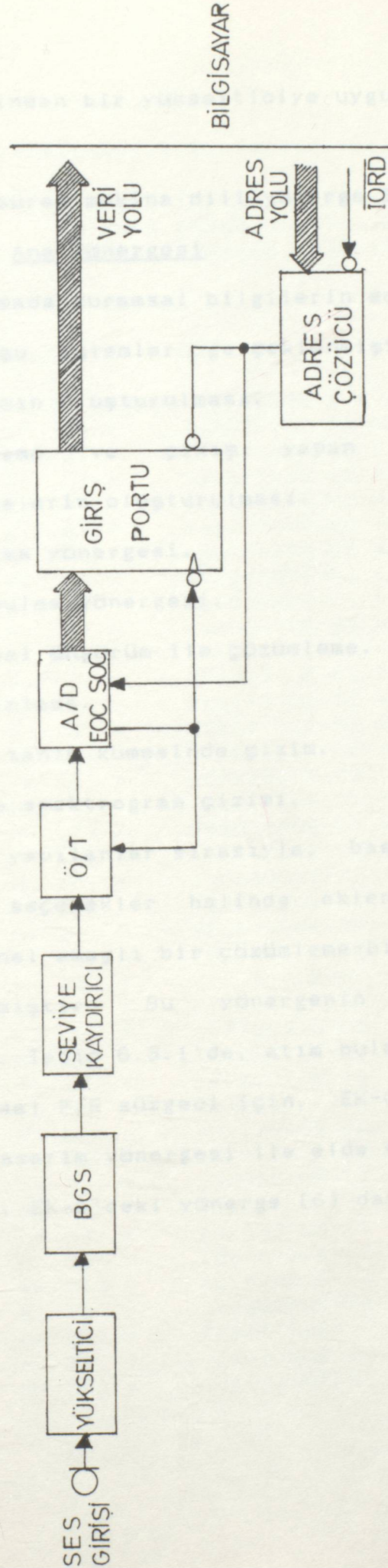
6.1 Giriş

Bu bölümde, şimdiye kadar anlatılan kuramsal bilgilere dayanılarak, konuşma çözümüleme-bireşimleme dizgesinin sayısal bir bilgisayar üzerinde gerçekleştirilen benzetimi ve elde edilen sonuçlar verilecektir.

6.2 Örneklemeye ve Çıkış İçin Kullanılan Donanım

Örneklemeye ve çıkış için kullanılan donanımın blok çizgesi Şekil 6.1.1'de görülmektedir. Örneklemeye için konuşma imi, bir mikrofon ve yükseltici üzerinden, alt kesim frekansı 100 Hz, üst kesim frekansı 4000 Hz ve kenar eğimi 60 DB/dec. olan bir BGS'e uygulanır. BGS çıkışındaki im, kullanılan A/D'nin sadece 0-10 V'luk girişleri sayısallaştırması nedeniyle 5 V'luk bir DC seviye kaydırıcı üzerinden A/D girişindeki örneklemeye-tutma bloğuna aktarılır. Bu örneklemeye-tutma bloğunun örnekle/tut denetim girişi ile giriş portunun input enable girişi A/D'nin EOC çıkışı ile denetlenmektedir. Giriş portunun output enable denetimi ve A/D'nin SOC girişi ise adres çözücü çıkışı ile denetlenir. Böylece uygun zamanlama bağlantıları gerçekleştirilmiş olur. Örneklemeye frekansı bu donanımı yürüten makina dili bir yönerge tarafından belirlenir. Dikkat edilirse örneklemeye işleminin başlangıcında boş bir okuma yapılması gerektiği anlaşılır.

Çıkış için printer portu kullanılmıştır. Bu porttan gelen sayısal örnek dizisi bir D/A ile analog işarete çevirildikten sonra, kesim frekansı 4000 Hz olan bir



Şekil 6.2.1 Donanım Blok Çizgesi

süzgeç üzerinden bir yükselticiye uygulanarak sesyayardan çıkılır.

Donanımı süren makina dili yönerge EK-A'da verilmiştir.

6.3 Benzetim Ana Yönergesi

Bu çalışmada kuramsal bilgilerin edinilmesinden sonra sırasıyla şu işlemler gerçekleştirilmiştir:

- 1) Donanımın oluşturulması.
- 2) Örneklem ve çıkış yapan makina dili alt yönergelerin oluşturulması.
- 3) Uç bulma yönergesi.
- 4) Atım bulma yönergesi.
- 5) Doğrusal öngörüm ile çözümleme.
- 6) Bireşimleme.
- 7) Zaman tanım kümesinde çizim.
- 8) FFT ve spektrogram çizimi.

Bütün bu yapılanlar sırasıyla, basic dilindeki bir ana yönergeye seçenekler halinde eklenmiştir. Böylelikle sonuçta genel amaçlı bir çözümleme-bireşimleme yönergesi elde edilmiştir. Bu yönergenin listesi EK-B' de verilmiştir. Tablo 6.3.1'de, atım bulmada gerekli olan 900 Hz'lik sayısal FIR süzgeci için, EK-C'de verilen FORTRAN dili FIR tasarım yönergesi ile elde edilen dürtü tepkisi verilmiştir. EK-C'deki yönerge [6]'dan güncellenmiştir.

TABLO 6.3.1

FINITE IMPULSE RESPONSE (FIR)
LINEAR PHASE DIGITAL FILTER DESIGN
REMEZ EXCHANGE ALGORITHM

BANDPASS FILTER

FILTER LENGTH = 99

* * * * * IMPULSE RESPONSE * * * * *

H(1) =	-.14937570E-02	= H(99)
H(2) =	.65197920E-03	= H(98)
H(3) =	.16629740E-02	= H(97)
H(4) =	.30231610E-02	= H(96)
H(5) =	.42878080E-02	= H(95)
H(6) =	.49578030E-02	= H(94)
H(7) =	.46404380E-02	= H(93)
H(8) =	.32183990E-02	= H(92)
H(9) =	.94986470E-03	= H(91)
H(10) =	-.15556580E-02	= H(90)
H(11) =	-.35032310E-02	= H(89)
H(12) =	-.41815520E-02	= H(88)
H(13) =	-.32452730E-02	= H(87)
H(14) =	-.89991110E-03	= H(86)
H(15) =	.20972390E-02	= H(85)
H(16) =	.46497480E-02	= H(84)
H(17) =	.56980890E-02	= H(83)
H(18) =	.46354200E-02	= H(82)
H(19) =	.16033020E-02	= H(81)
H(20) =	-.24652280E-02	= H(80)
H(21) =	-.61067460E-02	= H(79)
H(22) =	-.78308870E-02	= H(78)
H(23) =	-.67015100E-02	= H(77)
H(24) =	-.27772630E-02	= H(76)
H(25) =	.27775230E-02	= H(75)
H(26) =	.79985580E-02	= H(74)
H(27) =	.10781570E-01	= H(73)
H(28) =	.96885690E-02	= H(72)
H(29) =	.46015010E-02	= H(71)
H(30) =	-.30501690E-02	= H(70)
H(31) =	-.10626800E-01	= H(69)
H(32) =	-.15123610E-01	= H(68)
H(33) =	-.14283370E-01	= H(67)
H(34) =	-.75928790E-02	= H(66)
H(35) =	.32742850E-02	= H(65)
H(36) =	.14730080E-01	= H(64)
H(37) =	.22337790E-01	= H(63)
H(38) =	.22343740E-01	= H(62)
H(39) =	.13204420E-01	= H(61)

TABLO 6.3.1 (DEVAMI)

H(40) = -.34362940E-02 = H(60)
H(41) = -.22725650E-01 = H(59)
H(42) = -.37671230E-01 = H(58)
H(43) = -.41068880E-01 = H(57)
H(44) = -.27750400E-01 = H(56)
H(45) = .35391410E-02 = H(55)
H(46) = .49258260E-01 = H(54)
H(47) = .10153690E+00 = H(53)
H(48) = .14994540E+00 = H(52)
H(49) = .18410800E+00 = H(51)
H(50) = .19642400E+00 = H(50)

	BAND 1	BAND 2
LOWER BAND EDGE	.000000000	.110000000
UPPER BAND EDGE	.090000000	.500000000
DESIRED VALUE	1.000000000	.000000000
WEIGHTING	1.000000000	10.000000000
DEVIATION	.032647320	.003264732
DEVIATION IN DB	-29.723050000	-49.723050000

EXTREMAL FREQUENCIES

.0000000	.0112500	.0225000	.0331250	.0443750
.0556250	.0662500	.0768749	.0856249	.0900000
.1100000	.1125000	.1187500	.1268750	.1356250
.1450000	.1543750	.1643749	.1743749	.1843749
.1943749	.2043749	.2143749	.2249999	.2349999
.2449999	.2550000	.2656252	.2756254	.2856257
.2962509	.3062511	.3162514	.3262516	.3368768
.3468771	.3568773	.3675025	.3775028	.3875030
.3981282	.4081285	.4181287	.4287539	.4387542
.4487544	.4593796	.4693799	.4793801	.4900053
.5000000				

6.4 Deneysel Çalışmalar ve Sonuçlar

Geliştirilen benzetim yönergesi ile üç ayrı kayıt yapılarak bu kayıtlar için sırasıyla bütün çözümleme işlemleri uygulanmıştır. Bu kayıtlar "alo", "ses", kelimeleri ile "konuşma çözümülemesi" tümcesinden oluşmaktadır. "Alo" kelimesi uygulamanın daha çok sesli sesler için davranışını, "ses" kelimesi ise sessiz sesler için davranışını sınamak için seçilmiştir. "Konuşma çözümülemesi" tümcesi ise uygulamanın tümce uzunluklu bir kayıt için davranışını incelemek amacıyla keyfi olarak seçilmiştir. Seçilen bu kayıtlar için elde edilen sonuçlar EK-D'de toplu olarak verilmiştir. Bu sonuçlar uç noktaları bulma için yapılan çizimler, özgün imin görüngesi ve bireşimlenmiş imin görüngesinden oluşurlar. Ayrıca "alo" sesinin, atım bulma öncesinde 900 Hz'lik sayısal süzgeçten geçirilmesi ile elde edilen imin görüngesi de verilmiştir.

BÖLÜM 7. SONUÇLAR

Daha önce de söylendiği gibi bu çalışmanın asıl amacı konuşma çözümlemesine bir giriş yapmak ve bu konudaki temel bilgileri edinmekdi. Bu amaçla önce kuramsal bilgiler edinilmiş ardından basit bir uygulama gerçekleştirilmiştir.

EK-D'de verilen özgün görüngeler ile bireşimlenmiş görüngeler karşılaştırılırsa, yapılan uygulamanın başarımı hakkında bir fikir edinilebilir. Ayrıca yapılan öznel değerlendirmelerin oldukça iyi sonuç verdiği de belirtilmelidir.

Sonuç olarak amaçlanan hedeflere ulaşıldığı söylenebilir.

PAGE 60,132

TITLE SRC - Sampling, Reproducing and Clearing

```

;-----
CODESG SEGMENT PARA 'CODE'
    ASSUME CS:CODESG,DS:CODESG
; * * * * *
;      Sampling:
; * * * * *
SMP    PROC    FAR
    PUSH    DS
    PUSH    ES
    PUSH    SS
    PUSH    BP
    MOV     AX,CODESG
    MOV     DS,AX
    CALL    Q10CLR      ;Clear screen
    MOV     RPRLEN,SMLEN ;Set RPRLEN to SMLEN
    LEA     DX,WFSMSG    ;Display wait message
    CALL    X10DSP
    CLI     ;Clear interrupt enable
    MOV     AL,00H       ;Reset ENABLE NMI
    OUT     OAOH,AL      ; signal
    MOV     DX,2F8H      ;Input for first
    IN      AL,DX        ; SOC
    MOV     CX,SMLEN     ;Set counter
    MOV     DI,00H       ;Set index reg. DI to 00
SNXT:  MOV     BX,0EH     ;Delay for 10 KHz sampling
SBCK:  DEC     BX
    NOP     ;Dummy command
    JNZ     SBCK
    OR      AX,AX        ;Equalize to RPR
    IN      AL,DX        ;Input for reading last conversion and next SOC
    MOV     TABLE[DI],AL ;Store into table
    INC     DI           ;Inc. index reg.
    LOOP    SNXT         ;Fly - back for next conversion
    MOV     AL,80H       ;Set ENABLE NMI
    OUT     OAOH,AL      ; signal
    STI     ;Set interrupt enable
    POP     BP
    POP     SS
    POP     ES
    POP     DS
    RET
SMP    ENDP
; * * * * *
;      Reproducing:
; * * * * *
RPR    PROC    FAR
    PUSH    DS
    PUSH    ES
    PUSH    SS
    PUSH    BP
    MOV     AX,CODESG

```



```

MOV     DS,AX
CALL    Q10CLR      ;Clear screen
LEA     DX,WFRMSG   ;Display wait message
CALL    X10DSP
CLI     ;Clear interrupt enable
MOV     AL,00H      ;Reset ENABLE NMI
OUT     OAOH,AL     ; signal
MOV     CX,RPRLEN   ;Set counter
MOV     DX,03BCH    ;Set prt. port addr.
MOV     DI,00H      ;Set index reg. DI to 00
RNXT:   MOV     BX,0EH ;Delay for 10 KHz reproducing
RBCK:   DEC     BX
NOP     ;Dummy command
JNZ     RBCK        ;
OR      AX,00H      ;Equalize to SMP
MOV     AL,TABLE[DI] ;Load AL with sample from table
INC     DI          ;Inc. index reg.
OUT     DX,AL       ;Output sample
LOOP    RNXT        ;Fly-back for next output
MOV     AL,80H      ;Set ENABLE NMI
OUT     OAOH,AL     ; signal
STI     ;Set interrupt enable
POP     BP
POP     SS
POP     ES
POP     DS
RET
RPR     ENDP
; * * * * *
;
;      Clear TABLE:
; * * * * *
CTB     PROC     FAR
        PUSH    DS
        PUSH    ES
        PUSH    SS
        PUSH    BP
        MOV     AX,CODESEG
        MOV     DS,AX
        CALL    Q10CLR      ;Clear screen
        MOV     RPRLEN,SMLEN ;Set RPRLEN to SMLEN
        LEA     DX,WFCMSG   ;Display wait message
        CALL    X10DSP
        CLI     ;Clear interrupt enable
        MOV     AL,00H      ;Reset ENABLE NMI
        OUT     OAOH,AL     ; signal
        MOV     CX,SMLEN    ;Set CX to SMLEN
        MOV     DI,00H      ;Set index reg. DI to 00
FRW:    MOV     TABLE[DI],00H ;Clear TABLE
        INC     DI
        LOOP    FRW
        MOV     AL,80H      ;Set ENABLE NMI
        OUT     OAOH,AL     ; signal
        STI     ;Set interrupt enable

```



```

        POP    BP
        POP    SS
        POP    ES
        POP    DS
        RET
CTB     ENDP
;
;      Clear screen:
;      -----
Q10CLR  PROC    NEAR
        MOV    AX,0600H      ;Request scrool
        MOV    BH,07
        MOV    CX,0000
        MOV    DX,184FH
        INT    10H
        MOV    DX,0000      ;Set cursor to 00,00
        CALL   Q20CURS
        RET
Q10CLR  ENDP
;
;      Set cursor acc'g to DX:
;      -----
Q20CURS PROC    NEAR
        MOV    AH,02      ;Request set
        MOV    BH,00      ; cursor
        INT    10H
        RET
Q20CURS ENDP
;
;      Display routine:
;      -----
X10DSP  PROC    NEAR
        MOV    AH,09      ;DX contains address
        INT    21H        ; of message
        RET
X10DSP  ENDP
WFSMSG  DB      '* ORNEKLEME YAPILIYOR ... *','$'
WFRMSG  DB      '* CIKIS YAPILIYOR ... *','$'
WFCMSG  DB      '* BELLEK Siliniyor ... *','$'
SMPLN   EQU     30000
RPRLEN  DW      7530H
TABLE   DB      SMPLN DUP (?)
CODESG  ENDS
        END
```



```

10 CLS:PRINT "KODU OKU"
20 DEFINT =A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
30 PI=3.14159265358979323846264338327950288419716939937510582097494
40 DIM D(255),L(255)
50 D(0)=0:FOR I=1 TO 255:GOTO 100
60 GOTO 100,200,300,400,500,600,700,800,900,1000,1100,1200,1300,1400
70 GOTO 1500,1600,1700,1800,1900,2000,2100,2200,2300,2400
80 GOTO 2500,2600,2700,2800,2900,3000,3100,3200,3300,3400
90 GOTO 3500,3600,3700,3800,3900,4000,4100,4200,4300,4400
100 GOTO 4500,4600,4700,4800,4900,5000,5100,5200,5300,5400
110 GOTO 5500,5600,5700,5800,5900,6000,6100,6200,6300,6400
120 GOTO 6500,6600,6700,6800,6900,7000,7100,7200,7300,7400
130 GOTO 7500,7600,7700,7800,7900,8000,8100,8200,8300,8400
140 GOTO 8500,8600,8700,8800,8900,9000,9100,9200,9300,9400
150 GOTO 9500,9600,9700,9800,9900,10000,10100,10200,10300,10400
160 GOTO 10500,10600,10700,10800,10900,11000,11100,11200,11300,11400
170 GOTO 11500,11600,11700,11800,11900,12000,12100,12200,12300,12400
180 GOTO 12500,12600,12700,12800,12900,13000,13100,13200,13300,13400
190 GOTO 13500,13600,13700,13800,13900,14000,14100,14200,14300,14400
200 GOTO 14500,14600,14700,14800,14900,15000,15100,15200,15300,15400
210 GOTO 15500,15600,15700,15800,15900,16000,16100,16200,16300,16400
220 GOTO 16500,16600,16700,16800,16900,17000,17100,17200,17300,17400
230 GOTO 17500,17600,17700,17800,17900,18000,18100,18200,18300,18400
240 GOTO 18500,18600,18700,18800,18900,19000,19100,19200,19300,19400
250 GOTO 19500,19600,19700,19800,19900,20000,20100,20200,20300,20400
260 GOTO 20500,20600,20700,20800,20900,21000,21100,21200,21300,21400
270 GOTO 21500,21600,21700,21800,21900,22000,22100,22200,22300,22400
280 GOTO 22500,22600,22700,22800,22900,23000,23100,23200,23300,23400
290 GOTO 23500,23600,23700,23800,23900,24000,24100,24200,24300,24400
300 GOTO 24500,24600,24700,24800,24900,25000,25100,25200,25300,25400
310 GOTO 25500,25600,25700,25800,25900,26000,26100,26200,26300,26400
320 GOTO 26500,26600,26700,26800,26900,27000,27100,27200,27300,27400
330 GOTO 27500,27600,27700,27800,27900,28000,28100,28200,28300,28400
340 GOTO 28500,28600,28700,28800,28900,29000,29100,29200,29300,29400
350 GOTO 29500,29600,29700,29800,29900,30000,30100,30200,30300,30400
360 GOTO 30500,30600,30700,30800,30900,31000,31100,31200,31300,31400
370 GOTO 31500,31600,31700,31800,31900,32000,32100,32200,32300,32400
380 GOTO 32500,32600,32700,32800,32900,33000,33100,33200,33300,33400
390 GOTO 33500,33600,33700,33800,33900,34000,34100,34200,34300,34400
400 GOTO 34500,34600,34700,34800,34900,35000,35100,35200,35300,35400
410 GOTO 35500,35600,35700,35800,35900,36000,36100,36200,36300,36400
420 GOTO 36500,36600,36700,36800,36900,37000,37100,37200,37300,37400
430 GOTO 37500,37600,37700,37800,37900,38000,38100,38200,38300,38400
440 GOTO 38500,38600,38700,38800,38900,39000,39100,39200,39300,39400
450 GOTO 39500,39600,39700,39800,39900,40000,40100,40200,40300,40400
460 GOTO 40500,40600,40700,40800,40900,41000,41100,41200,41300,41400
470 GOTO 41500,41600,41700,41800,41900,42000,42100,42200,42300,42400
480 GOTO 42500,42600,42700,42800,42900,43000,43100,43200,43300,43400
490 GOTO 43500,43600,43700,43800,43900,44000,44100,44200,44300,44400
500 GOTO 44500,44600,44700,44800,44900,45000,45100,45200,45300,45400
510 GOTO 45500,45600,45700,45800,45900,46000,46100,46200,46300,46400
520 GOTO 46500,46600,46700,46800,46900,47000,47100,47200,47300,47400
530 GOTO 47500,47600,47700,47800,47900,48000,48100,48200,48300,48400
540 GOTO 48500,48600,48700,48800,48900,49000,49100,49200,49300,49400
550 GOTO 49500,49600,49700,49800,49900,50000,50100,50200,50300,50400
560 GOTO 50500,50600,50700,50800,50900,51000,51100,51200,51300,51400
570 GOTO 51500,51600,51700,51800,51900,52000,52100,52200,52300,52400
580 GOTO 52500,52600,52700,52800,52900,53000,53100,53200,53300,53400
590 GOTO 53500,53600,53700,53800,53900,54000,54100,54200,54300,54400
600 GOTO 54500,54600,54700,54800,54900,55000,55100,55200,55300,55400
610 GOTO 55500,55600,55700,55800,55900,56000,56100,56200,56300,56400
620 GOTO 56500,56600,56700,56800,56900,57000,57100,57200,57300,57400
630 GOTO 57500,57600,57700,57800,57900,58000,58100,58200,58300,58400
640 GOTO 58500,58600,58700,58800,58900,59000,59100,59200,59300,59400
650 GOTO 59500,59600,59700,59800,59900,60000,60100,60200,60300,60400
660 GOTO 60500,60600,60700,60800,60900,61000,61100,61200,61300,61400
670 GOTO 61500,61600,61700,61800,61900,62000,62100,62200,62300,62400
680 GOTO 62500,62600,62700,62800,62900,63000,63100,63200,63300,63400
690 GOTO 63500,63600,63700,63800,63900,64000,64100,64200,64300,64400
700 GOTO 64500,64600,64700,64800,64900,65000,65100,65200,65300,65400
710 GOTO 65500,65600,65700,65800,65900,66000,66100,66200,66300,66400
720 GOTO 66500,66600,66700,66800,66900,67000,67100,67200,67300,67400
730 GOTO 67500,67600,67700,67800,67900,68000,68100,68200,68300,68400
740 GOTO 68500,68600,68700,68800,68900,69000,69100,69200,69300,69400
750 GOTO 69500,69600,69700,69800,69900,70000,70100,70200,70300,70400
760 GOTO 70500,70600,70700,70800,70900,71000,71100,71200,71300,71400
770 GOTO 71500,71600,71700,71800,71900,72000,72100,72200,72300,72400
780 GOTO 72500,72600,72700,72800,72900,73000,73100,73200,73300,73400
790 GOTO 73500,73600,73700,73800,73900,74000,74100,74200,74300,74400
800 GOTO 74500,74600,74700,74800,74900,75000,75100,75200,75300,75400
810 GOTO 75500,75600,75700,75800,75900,76000,76100,76200,76300,76400
820 GOTO 76500,76600,76700,76800,76900,77000,77100,77200,77300,77400
830 GOTO 77500,77600,77700,77800,77900,78000,78100,78200,78300,78400
840 GOTO 78500,78600,78700,78800,78900,79000,79100,79200,79300,79400
850 GOTO 79500,79600,79700,79800,79900,80000,80100,80200,80300,80400
860 GOTO 80500,80600,80700,80800,80900,81000,81100,81200,81300,81400
870 GOTO 81500,81600,81700,81800,81900,82000,82100,82200,82300,82400
880 GOTO 82500,82600,82700,82800,82900,83000,83100,83200,83300,83400
890 GOTO 83500,83600,83700,83800,83900,84000,84100,84200,84300,84400
900 GOTO 84500,84600,84700,84800,84900,85000,85100,85200,85300,85400
910 GOTO 85500,85600,85700,85800,85900,86000,86100,86200,86300,86400
920 GOTO 86500,86600,86700,86800,86900,87000,87100,87200,87300,87400
930 GOTO 87500,87600,87700,87800,87900,88000,88100,88200,88300,88400
940 GOTO 88500,88600,88700,88800,88900,89000,89100,89200,89300,89400
950 GOTO 89500,89600,89700,89800,89900,90000,90100,90200,90300,90400
960 GOTO 90500,90600,90700,90800,90900,91000,91100,91200,91300,91400
970 GOTO 91500,91600,91700,91800,91900,92000,92100,92200,92300,92400
980 GOTO 92500,92600,92700,92800,92900,93000,93100,93200,93300,93400
990 GOTO 93500,93600,93700,93800,93900,94000,94100,94200,94300,94400
1000 GOTO 94500,94600,94700,94800,94900,95000,95100,95200,95300,95400
1010 GOTO 95500,95600,95700,95800,95900,96000,96100,96200,96300,96400
1020 GOTO 96500,96600,96700,96800,96900,97000,97100,97200,97300,97400
1030 GOTO 97500,97600,97700,97800,97900,98000,98100,98200,98300,98400
1040 GOTO 98500,98600,98700,98800,98900,99000,99100,99200,99300,99400
1050 GOTO 99500,99600,99700,99800,99900,100000,100100,100200,100300,100400
1060 GOTO 100500,100600,100700,100800,100900,101000,101100,101200,101300,101400
1070 GOTO 101500,101600,101700,101800,101900,102000,102100,102200,102300,102400
1080 GOTO 102500,102600,102700,102800,102900,103000,103100,103200,103300,103400
1090 GOTO 103500,103600,103700,103800,103900,104000,104100,104200,104300,104400
1100 GOTO 104500,104600,104700,104800,104900,105000,105100,105200,105300,105400
1110 GOTO 105500,105600,105700,105800,105900,106000,106100,106200,106300,106400
1120 GOTO 106500,106600,106700,106800,106900,107000,107100,107200,107300,107400
1130 GOTO 107500,107600,107700,107800,107900,108000,108100,108200,108300,108400
1140 GOTO 108500,108600,108700,108800,108900,109000,109100,109200,109300,109400
1150 GOTO 109500,109600,109700,109800,109900,110000,110100,110200,110300,110400
1160 GOTO 110500,110600,110700,110800,110900,111000,111100,111200,111300,111400
1170 GOTO 111500,111600,111700,111800,111900,112000,112100,112200,112300,112400
1180 GOTO 112500,112600,112700,112800,112900,113000,113100,113200,113300,113400
1190 GOTO 113500,113600,113700,113800,113900,114000,114100,114200,114300,114400
1200 GOTO 114500,114600,114700,114800,114900,115000,115100,115200,115300,115400
1210 GOTO 115500,115600,115700,115800,115900,116000,116100,116200,116300,116400
1220 GOTO 116500,116600,116700,116800,116900,117000,117100,117200,117300,117400
1230 GOTO 117500,117600,117700,117800,117900,118000,118100,118200,118300,118400
1240 GOTO 118500,118600,118700,118800,118900,119000,119100,119200,119300,119400
1250 GOTO 119500,119600,119700,119800,119900,120000,120100,120200,120300,120400
1260 GOTO 120500,120600,120700,120800,120900,121000,121100,121200,121300,121400
1270 GOTO 121500,121600,121700,121800,121900,122000,122100,122200,122300,122400
1280 GOTO 122500,122600,122700,122800,122900,123000,123100,123200,123300,123400
1290 GOTO 123500,123600,123700,123800,123900,124000,124100,124200,124300,124400
1300 GOTO 124500,124600,124700,124800,124900,125000,125100,125200,125300,125400
1310 GOTO 125500,125600,125700,125800,125900,126000,126100,126200,126300,126400
1320 GOTO 126500,126600,126700,126800,126900,127000,127100,127200,127300,127400
1330 GOTO 127500,127600,127700,127800,127900,128000,128100,128200,128300,128400
1340 GOTO 128500,128600,128700,128800,128900,129000,129100,129200,129300,129400
1350 GOTO 129500,129600,129700,129800,129900,130000,130100,130200,130300,130400
1360 GOTO 130500,130600,130700,130800,130900,131000,131100,131200,131300,131400
1370 GOTO 131500,131600,131700,131800,131900,132000,132100,132200,132300,132400
1380 GOTO 132500,132600,132700,132800,132900,133000,133100,133200,133300,133400
1390 GOTO 133500,133600,133700,133800,133900,134000,134100,134200,134300,134400
1400 GOTO 134500,134600,134700,134800,134900,135000,135100,135200,135300,135400
1410 GOTO 135500,135600,135700,135800,135900,136000,136100,136200,136300,136400
1420 GOTO 136500,136600,136700,136800,136900,137000,137100,137200,137300,137400
1430 GOTO 137500,137600,137700,137800,137900,138000,138100,138200,138300,138400
1440 GOTO 138500,138600,138700,138800,138900,139000,139100,139200,139300,139400
1450 GOTO 139500,139600,139700,139800,139900,140000,140100,140200,140300,140400
1460 GOTO 140500,140600,140700,140800,140900,141000,141100,141200,141300,141400
1470 GOTO 141500,141600,141700,141800,141900,142000,142100,142200,142300,142400
1480 GOTO 142500,142600,142700,142800,142900,143000,143100,143200,143300,143400
1490 GOTO 143500,143600,143700,143800,143900,144000,144100,144200,144300,144400
1500 GOTO 144500,144600,144700,144800,144900,145000,145100,145200,145300,145400
1510 GOTO 145500,145600,145700,145800,145900,146000,146100,146200,146300,146400
1520 GOTO 146500,146600,146700,146800,146900,147000,147100,147200,147300,147400
1530 GOTO 147500,147600,147700,147800,147900,148000,148100,148200,148300,148400
1540 GOTO 148500,148600,148700,148800,148900,149000,149100,149200,149300,149400
1550 GOTO 149500,149600,149700,149800,149900,150000,150100,150200,150300,150400
1560 GOTO 150500,150600,150700,150800,150900,151000,151100,151200,151300,151400
1570 GOTO 151500,151600,151700,151800,151900,152000,152100,152200,152300,152400
1580 GOTO 152500,152600,152700,152800,152900,153000,153100,153200,153300,153400
1590 GOTO 153500,153600,153700,153800,153900,154000,154100,154200,154300,154400
1600 GOTO 154500,154600,154700,154800,154900,155000,155100,155200,155300,155400
1610 GOTO 155500,155600,155700,155800,155900,156000,156100,156200,156300,156400
1620 GOTO 156500,156600,156700,156800,156900,157000,157100,157200,157300,157400
1630 GOTO 157500,157600,157700,157800,157900,158000,158100,158200,158300,158400
1640 GOTO 158500,158600,158700,158800,158900,159000,159100,159200,159300,159400
1650 GOTO 159500,159600,159700,159800,159900,160000,160100,160200,160300,160400
1660 GOTO 160500,160600,160700,160800,160900,161000,161100,161200,161300,161400
1670 GOTO 161500,161600,161700,161800,161900,162000,162100,162200,162300,162400
1680 GOTO 162500,162600,162700,162800,162900,163000,163100,163200,163300,163400
1690 GOTO 163500,163600,163700,163800,163900,164000,164100,164200,164300,164400
1700 GOTO 164500,164600,164700,164800,164900,165000,165100,165200,165300,165400
1710 GOTO 165500,165600,165700,165800,165900,166000,166100,166200,166300,166400
1720 GOTO 166500,166600,166700,166800,166900,167000,167100,167200,167300,167400
1730 GOTO 167500,167600,167700,167800,167900,168000,168100,168200,168300,168400
1740 GOTO 168500,168600,168700,168800,168900,169000,169100,169200,169300,169400
1750 GOTO 169500,169600,169700,169800,169900,170000,170100,170200,170300,170400
1760 GOTO 170500,170600,170700,170800,170900,171000,171100,171200,171300,171400
1770 GOTO 171500,171600,171700,171800,171900,172000,172100,172200,172300,172400
1780 GOTO 172500,172600,172700,172800,172900,173000,173100,173200,173300,173400
1790 GOTO 173500,1736
```



```
10 CLS:SCREEN 0:KEY OFF
20 DEFINT H-N:DEF SEG=&H7880:TB%=&H129:SS%=0:RS%=&H44:CTBS%=&H82
30 P1=3.141592653589793#:DP1=2*P1:SDB=10/LOG(10):DDB=2*SDB
40 DIM ZR%(300),E(300)
50 DIM H%(98),T%(300),R(200),CSC%(299)
60 DIM A(30,30),K!(30),V(30),I!(30),U(99)
70 DIM AR(512),AI(512)
80 DIM I1(4),I2(4),I3(4),I4(4),I5(4),I6(4),I7(4),I8(4)
90 BLOAD "SRC.SBR"
100 CLS:GOTO 230
110 CLS:SCREEN 0
120 PRINT"1-) DOSYA ADI GİRME 6-) UC NOKTALARI BULMA "
130 PRINT"2-) YUKLEME 7-) ATIM BULMA "
140 PRINT"3-) KAYIT 8-) COZUMLEME "
150 PRINT"4-) ORNEKLEME 9-) BIRESIM "
160 PRINT"5-) HP'E ÇIKIS 0-) TD/FD ÇİZİM "
170 PRINT
180 PRINT"DOSYA = ";F$+"."+E$
190 A$=INKEY$:IF A$="" THEN 190
200 IF ASC(A$)<48 OR ASC(A$)>57 THEN 190 ELSE CLS
210 ON ASC(A$)-47 GOTO 4580,230,370,470,530,560,610,2110,3050,3940
220 ' * * * DOSYA ADI GİRME * * *
230 INPUT"DOSYA ADI = ";F$
240 F$=LEFT$(F$,8):PRINT
250 IF F$<>"" THEN 270
260 CLS:GOTO 230
270 INPUT"SOYADI = ";E$
280 CLS
290 E$=LEFT$(E$,3)
300 IF E$<>"REC" THEN 320
310 CALL CTBS%:GOTO 110
320 IF E$="ORG" OR E$="LPF" OR E$="WND" OR E$="SNT" THEN 380
330 IF E$="OFS" OR E$="LFS" OR E$="WFS" OR E$="SFS" THEN 6100
340 PRINT"* SOYADI REC,ORG,LPF,WND,SNT,OFS,LFS,WFS, VEYA SFS OLMALI !.*"
350 PRINT:GOTO 270
360 ' * * * YUKLEME * * *
370 IF RIGHT$(E$,1)="S" THEN 440
380 PRINT"* YUKLENİYOR ... *"
390 BLOAD F$+"."+E$,TB%
400 OPEN "A",#1,F$+"."+E$:L=LOF(1)-15
410 LH=INT(L/256):POKE TB%-1,LH
420 LL=L-256*LH:POKE TB%-2,LL
430 CLOSE #1:GOTO 110
440 PRINT"* BU DOSYA İÇİN UYGULANAMAZ !.*"
450 A$=INKEY$:IF A$="" THEN 450 ELSE 110
460 ' * * * KAYIT * * *
470 IF RIGHT$(E$,1)="S" THEN 500
480 PRINT"* KAYIT YAPILIYOR ... *"
490 BSAVE F$+"."+E$,TB%,256*PEEK(TB%-1)+PEEK(TB%-2):GOTO 110
500 PRINT"* BU DOSYA İÇİN UYGULANAMAZ !.*"
510 A$=INKEY$:IF A$="" THEN 510 ELSE 110
520 ' * * * ORNEKLEME * * *
530 E$="REC"
```



```
540 CALL SS%:GOTO 110
550 ' * * * HP'E ÇIKIS * * *
560 IF RIGHT$(E$,1)="S" THEN 580
570 CALL RS%:GOTO 110
580 PRINT"* BU DOSYA İÇİN UYGULANAMAZ !.*"
590 A$=INKEY$:IF A$="" THEN 590 ELSE 110
600 ' * * * UC NOKTALARI BULMA * * *
610 IF E$="ORG" THEN 1470
620 IF E$="REC" THEN 650
630 PRINT"* BU DOSYA İÇİN UYGULANAMAZ !.*"
640 A$=INKEY$:IF A$="" THEN 640 ELSE 110
650 PRINT"* UC NOKTALARI BULUNUYOR ... *"
660 AVS=0:AZRS%=0:ASES=0:VZRS=0:SL=0
670 FOR I=0 TO 999
680 AVS=AVS+PEEK(TB%+I)
690 NEXT
700 AV=AVS/1000
710 ST1=ABS(PEEK(TB%)-AV)
720 FOR I=1 TO 999
730 AS=ABS(PEEK(TB%+I)-AV)
740 IF ST1<AS THEN ST1=AS
750 NEXT
760 '* ZR%(n) VE E(n) 'NİN HESAPLANMASI *
770 FOR N=1 TO 300
780 ZC%=0:ES=0:I=-1:HA=TB%+(N-1)*100:GOTO 810
790 IF I=99 THEN 910 ELSE I=I+1
800 SL=PEEK(HA+I)-AV
810 IF SL=0 THEN 790
820 IF I>-1 THEN ES=ABS(SL)
830 I=I+1
840 IF I=100 THEN 910
850 SR=PEEK(HA+I)-AV
860 ON SGN(SL*SR)+1 GOTO 830,880
870 ZC%=ZC%+1
880 SL=SR
890 ES=ES+ABS(SR)
900 GOTO 830
910 ZR%(N)=ZC%
920 E(N)=ES
930 NEXT
940 '* * * * *
950 FOR N=1 TO 10
960 AZRS%=AZRS%+ZR%(N)
970 ASES=ASES+E(N)
980 NEXT
990 AZR=AZRS%/10
1000 IMN!=ASES/10
1010 FOR N=1 TO 10
1020 VZRS=VZRS+(ZR%(N)-AZR)^2
1030 NEXT
1040 VZR=VZRS/10
1050 SDZR=VZR .5
1060 IMX!=E(1):I2RX=ZR%(1)
```



```

1070 FOR N=2 TO 300
1080 IF IMX!<E(N) THEN IMX!=E(N)
1090 IF IZRX<ZR%(N) THEN IZRX=ZR%(N)
1100 NEXT
1110 IZCT!=AZR+2*SDZR
1120 IF IZCT!>25 THEN IZCT!=25
1130 I1!=.03*(IMX!-IMN!)+IMN!
1140 I2!=4*IMN!
1150 IF I2!=0 OR I1!<I2! THEN ITL!=I1! ELSE ITL!=I2!
1160 ITU!=5*ITL!
1170 M=1:H=1:ID=25:GOSUB 1810
1180 N1=N
1190 M=300:H=-1:ID=-25:GOSUB 1810
1200 N2=N
1210 IF N1>10 AND N2<300 THEN 1240
1220 BEEP:PRINT"KAYIT COK UZUN !"
1230 A$=INKEY$:IF A$="" THEN 1230 ELSE 110
1240 N1=N1-1:N2=N2+1:NSM=N2-N1+1
1250 BA%=TB%+(N1-1)*100:EA%=TB%+N2*100-1
1260 PU=ABS(PEEK(BA%)-AV)
1270 FOR SA%=BA%+1 TO EA%
1280 AS=ABS(PEEK(SA%)-AV)
1290 IF PU<AS THEN PU=AS
1300 ST2=PU/15
1310 NEXT
1320 ST=(ST1+ST2)/2
1330 BSAVE F$+".ORG",BA%,NSM*100
1340 OPEN "O",#1,F$+".EPA"
1350 PRINT #1,NSM,AV,ST,AZR,SDZR,IMN!,IMX!,IZRX,IZCT!,ITL!,ITU!
1360 FOR I=1 TO 300
1370 PRINT #1,E(I),ZR%(I)
1380 NEXT
1390 CLOSE #1
1400 E$="ORG"
1410 BLOAD F$+".ORG",TB%
1420 OPEN "A",#1,F$+".ORG":L=LOF(1)-15
1430 LH=INT(L/256):POKE TB%-1,LH
1440 LL=L-256*LH:POKE TB%-2,LL
1450 CLOSE #1
1460 '* EPA Cizim *
1470 CLS:PRINT"* BEKLEYİN ... *"
1480 OPEN "I",#1,F$+".EPA"
1490 INPUT #1,NSM,AV,ST,AZR,SDZR,IMN!,IMX!,IZRX,IZCT!,ITL!,ITU!
1500 FOR I=1 TO 300
1510 INPUT #1,E(I),ZR%(I)
1520 NEXT
1530 CLOSE #1
1540 SCREEN 2
1550 LOCATE 1,31:PRINT"DOSYA = ";F$+".EPA"
1560 LOCATE 1,6:PRINT"ERKE":LOCATE 1,79:PRINT CHR$(26);"t"
1570 LOCATE 14,6:PRINT"SIFIR GECİS":LOCATE 14,79:PRINT CHR$(26);"t"
1580 LINE (39,87)-(639,87):LINE (39,191)-(639,191)
1590 FOR I=0 TO 3

```



```

1600 IF I<3 THEN K=2*10~1
1610 FOR J=0 TO 600 STEP K
1620 H=39+J:PSET (H,88+I):PSET (H,192+I)
1630 NEXT J,I
1640 H=43:YE=87-79*E(1)/IMX!:LINE (39,YE)-(41,YE)
1650 FOR N=2 TO 300
1660 YE=87-79*E(N)/IMX!:LINE -(H-2,YE):LINE -(H,YE)
1670 H=H+2
1680 NEXT
1690 H=43:YZR=191-79*ZR%(1)/IZRX:LINE (39,YZR)-(41,YZR)
1700 FOR N=2 TO 300
1710 YZR=191-79*ZR%(N)/IZRX:LINE -(H-2,YZR):LINE -(H,YZR)
1720 H=H+2
1730 NEXT
1740 YTU=87-79*ITU!/IMX!:LINE (39,YTU)-(639,YTU),,,&HAAAA
1750 YTL=87-79*ITL!/IMX!:LINE (39,YTL)-(639,YTL),,,&HAAAA
1760 YZCT=191-79*IZCT!/IZRX:LINE (39,YZCT)-(639,YZCT),,,&HAAAA
1770 BEEP
1780 A$=INKEY$:IF A$="" THEN 1780 ELSE 110
1790 '* EPA ALTPROGRAM *
1800 '* E(n)'NIN TARANMASI *
1810 IF E(M)>ITL! THEN 1860
1820 M=M+H
1830 IF M>300 OR M=0 THEN PRINT"KAYIT YOK !" ELSE 1810
1840 BEEP
1850 A$=INKEY$:IF A$="" THEN 1850 ELSE 110
1860 N=M
1870 IF E(M)<ITU! THEN M=M+H ELSE 1910
1880 IF M>300 THEN PRINT"N11>300 !" ELSE IF M=0 THEN PRINT"N21<1 !" ELSE IF E(M)
<ITL! THEN 1820 ELSE 1870
1890 BEEP
1900 A$=INKEY$:IF A$="" THEN 1900 ELSE 110
1910 IF N=M THEN N=N-H
1920 IF N<=35 THEN PRINT"N11<36 !" ELSE IF N>275 THEN PRINT"N21>275 !" ELSE 1960
1930 BEEP
1940 A$=INKEY$:IF A$="" THEN 1940 ELSE 110
1950 '* ZR%(n)'iN TARANMASI *
1960 I=N-ID
1970 IF ZR%(I)>=IZCT! THEN 1990 ELSE I=I+H
1980 IF I=N THEN 2080 ELSE 1970
1990 NT=I:M=0
2000 M=M+1
2010 IF M=3 THEN N=NT ELSE 2030
2020 RETURN
2030 I=I+H
2040 IF I=N THEN 2060 ELSE IF ZR%(I)<IZCT! THEN 2030 ELSE 2000
2050 N=I
2060 I=I-H
2070 IF ZR%(I)>=IZCT! THEN 2050
2080 RETURN
2090 '* * * * *
2100 '* * * ATIM BULMA * * *
2110 IF E$="ORG" THEN 2140
2120 PRINT"* BU DOSTA iCIN UTGULANAMAZ !.*"

```



```
2130 A$=INKEY$:IF A$="" THEN 2130 ELSE 110
2140 PRINT" * ATIM DONEMLERI BULUNUYOR ... * "
2150 OPEN "I",#1,F$+".EPA"
2160 INPUT #1,NSM,AV,ST
2170 DAV=2*AV
2180 OPEN "I",#2,"LPF.IRP"
2190 FOR I=0 TO 98
2200 INPUT #2,H$(I)
2210 NEXT
2220 FOR N=NSM*100-1 TO 98 STEP -1
2230 Y#=AV:SA%=TB%+N
2240 FOR I=0 TO 48
2250 Y#=Y#+H$(I)*(PEEK(SA%-I)+PEEK(SA%-98+I)-DAV)
2260 NEXT I
2270 Y#=Y#+H$(49)*(PEEK(SA%-49)-AV)
2280 IF Y#>255 THEN Y#=255 ELSE IF Y#<0 THEN Y#=0
2290 POKE SA%,Y#
2300 NEXT N
2310 FOR N=97 TO 0 STEP -1
2320 Y#=AV:SA%=TB%+N
2330 FOR I=0 TO N
2340 Y#=Y#+H$(I)*(PEEK(SA%-I)-AV)
2350 NEXT I
2360 IF Y#>255 THEN Y#=255 ELSE IF Y#<0 THEN Y#=0
2370 POKE SA%,Y#
2380 NEXT N
2390 FOR N=0 TO NSM*100-50
2400 TA%=TB%+N
2410 POKE TA%,PEEK(TA%+49)
2420 NEXT
2430 FOR N=NSM*100-49 TO NSM*100-1
2440 POKE TB%+N,AV
2450 NEXT
2460 BSAVE F$+".LPF",TB%,NSM*100
2470 OPEN "R",#3,F$+".SUV",1
2480 FIELD #3,1 AS C$
2490 FOR NSC=1 TO NSM
2500 SCA%=TB%+(NSC-1)*100
2510 PAS=ABS(PEEK(SCA%)-AV)
2520 FOR N=1 TO 99
2530 FAS=ABS(PEEK(SCA%+N)-AV)
2540 IF PAS<FAS THEN PAS=FAS
2550 NEXT N
2560 T%(NSC)=PAS
2570 NEXT NSC
2580 LSET C$=CHR$(0)
2590 PUT #3
2600 FOR NSC=2 TO NSM-1
2610 PAS=T%(NSC-1)
2620 FOR I=NSC TO NSC+1
2630 IF PAS<T%(I) THEN PAS=T%(I)
2640 NEXT I
2650 IF PAS>ST THEN 2670
```



```
2660 C%=0:GOTO 2920
2670 IF T%(NSC-1)<T%(NSC+1) THEN CL=.64*T%(NSC-1) ELSE CL=.64*T%(NSC+1)
2680 SCA%=TB%+(NSC-2)*100
2690 FOR N=0 TO 299
2700 FS=PEEK(SCA%+N)-AV
2710 IF ABS(FS)<CL THEN CSC%(N)=0 ELSE IF FS>0 THEN CSC%(N)=1 ELSE CSC%(N)=-1
2720 NEXT N
2730 R(0)=0
2740 FOR N=0 TO 299
2750 IF CSC%(N)=0 THEN 2770
2760 R(0)=R(0)+1
2770 NEXT N
2780 FOR M=20 TO 200
2790 R(M)=0
2800 FOR N=0 TO 299-M
2810 IF CSC%(N)=0 OR CSC%(N+M)=0 THEN 2830
2820 IF CSC%(N)=CSC%(N+M) THEN R(M)=R(M)+1 ELSE R(M)=R(M)-1
2830 NEXT N
2840 R(M)=R(M)/R(0)
2850 NEXT M
2860 IPK!=R(20):IPOS=20
2870 FOR M=21 TO 200
2880 IF IPK!=R(M) THEN 2900
2890 IPK!=R(M):IPOS=M
2900 NEXT M
2910 IF IPK!<.3 THEN C%=1 ELSE C%=IPOS
2920 LSET C%=CHR$(C%)
2930 PUT #3
2940 NEXT NSC
2950 LSET C%=CHR$(0)
2960 PUT #3
2970 CLOSE #1,#2,#3
2980 BLOAD F$+".ORG",TB%
2990 OPEN "A",#1,F$+".ORG":L=LOF(1)-15
3000 LH=INT(L/256):POKE TB%-1,LH
3010 LL=L-256*LH:POKE TB%-2,LL
3020 CLOSE #1
3030 BEEP:GOTO 110
3040 ' * * COZUMLEME * * *
3050 IF E$="ORG" THEN 3080
3060 PRINT"* BU DOSYA iCin UYGULANAMAZ !.*"
3070 A$=INKEY$:IF A$="" THEN 3070 ELSE 110
3080 PRINT"* COZUMLEME YAPILIYOR ... *"
3090 OPEN "I",#1,F$+".EPA"
3100 INPUT #1,NSM,AV
3110 CLOSE #1
3120 OPEN "R",#1,F$+".SUV",1
3130 FIELD #1,1 AS C$
3140 OPEN "R",#2,F$+".EPF",1
3150 FIELD #2,1 AS I$
3160 OPEN "R",#3,F$+".PGP",4
3170 FIELD #3,4 AS S$
3180 CL%=0:N=100:PM%=30
```



```
3190 FOR NSC=1 TO NSM
3200 GET #1
3210 C%=ASC(C#)
3220 IF C%=0 THEN 3870
3230 SCA%=TB%+(NSC-1)*100
3240 IF C%<=1 THEN 3330
3250 MS!=ABS(PEEK(SCA%)-AV):E%=0
3260 FOR K=1 TO C%-1
3270 S=ABS(PEEK(SCA%+K)-AV)
3280 IF MS!>=S THEN 3300
3290 MS!=S:E%=K
3300 NEXT
3310 LSET I$=CHR$(E%)
3320 PUT #2
3330 FOR K=0 TO N-1
3340 SCSA%=SCA%+K
3350 POKE SCSA%, (.5+.46*COS(DPI*(K-(N-1)/2)/N))*(PEEK(SCSA%)-AV)+AV
3360 NEXT
3370 FOR I=0 TO PM%
3380 R(I)=0
3390 FOR K=0 TO (N-1)-I
3400 SCSA%=SCA%+K
3410 R(I)=R(I)+(PEEK(SCSA%)-AV)*(PEEK(SCSA%+I)-AV)
3420 NEXT K, I
3430 E(0)=R(0)
3440 FOR I=1 TO PM%
3450 TS=0
3460 FOR J=1 TO I-1
3470 TS=TS+A(J, I-1)*R(I-J)
3480 NEXT J
3490 K!(I)=- (R(I)+TS)/E(I-1)
3500 FOR J=1 TO I-1
3510 A(J, I)=A(J, I-1)+K!(I)*A(I-J, I-1)
3520 NEXT J
3530 A(I, I)=K!(I)
3540 E(I)=(1-K!(I)^2)*E(I-1)
3550 NEXT I
3560 FOR I=1 TO PM%
3570 IF ABS(K!(I))<=1 THEN 3620
3580 BEEP:PRINT "SUZGEC KARARSIZ !"
3590 CLOSE #1, #2, #3
3600 KILL F$+".EPF":KILL F$+".PGP"
3610 A$=INKEY$:IF A$="" THEN 3610 ELSE 110
3620 NEXT
3630 FOR I=1 TO PM%
3640 V(I)=E(I)/R(0)
3650 NEXT
3660 FOR P%=1 TO PM%
3670 I!(P%)=SDB*(LOG(V(P%))+5*P%/N)
3680 NEXT
3690 AMI=I!(1):PO%=1
3700 FOR P%=2 TO PM%
3710 IF AMI<=I!(P%) THEN 3730
```



```
3720 AMI=I!(P%):PO%=P%
3730 NEXT
3740 LSET I$=CHR$(PO%)
3750 PUT #2
3760 IF CL%>0 THEN 3810
3770 FOR K=0 TO PO%-1
3780 LSET I$=CHR$(PEEK(SCA%+K))
3790 PUT #2
3800 NEXT
3810 FOR J=1 TO PO%
3820 LSET S$=MKS$(A(J,PO%))
3830 PUT #3
3840 NEXT
3850 LSET S$=MKS$(SQR(E(PO%)))
3860 PUT #3
3870 CL%=C%
3880 NEXT NSC
3890 CLOSE #1,#2,#3
3900 E$="WND"
3910 BSAVE F$+"WND",TB%,NSM*100
3920 BEEP:GOTO 110
3930 '* * * BIRESIM * * *
3940 PRINT"* BIRESIM YAPILIYOR ... *"
3950 OPEN "I",#1,F$+".EPA"
3960 INPUT #1,NSM,AV
3970 CLOSE #1
3980 OPEN "R",#1,F$+".SUV",1
3990 FIELD #1,1 AS C$
4000 OPEN "R",#2,F$+".EPF",1
4010 FIELD #2,1 AS I$
4020 OPEN "R",#3,F$+".PGP",4
4030 FIELD #3,4 AS S$
4040 CL%=0:N=100
4050 FOR NSC=1 TO NSM
4060 SCA%=TB%+(NSC-1)*100
4070 GET #1
4080 C%=ASC(C$)
4090 IF C%>0 THEN 4140
4100 FOR K=0 TO N-1
4110 POKE SCA%+K,AV
4120 NEXT
4130 GOTO 4510
4140 IF C%>1 THEN 4220
4150 FOR K=0 TO N-2 STEP 2
4160 Y=SQR(-2*LOG(RND))
4170 Z=DFI*RND
4180 U(K)=Y*COS(Z)
4190 U(K+1)=Y*SIN(Z)
4200 NEXT
4210 GOTO 4270
4220 GET #2
4230 E%=ASC(I$)
4240 FOR K=0 TO N-1
```



```
4250 IF (K-E%)-C%*FIX((K-E%)/C%)=0 THEN U(K)=1 ELSE U(K)=0
4260 NEXT
4270 GET #2
4280 PO%=ASC(I$)
4290 SP%=0
4300 IF CL%>0 THEN 4360
4310 FOR K=0 TO PO%-1
4320 GET #2
4330 POKE SCA%+K,ASC(I$)
4340 NEXT
4350 SP%=PO%
4360 FOR J=1 TO PO%
4370 GET #3
4380 A(J,PO%)=CVS(S$)
4390 NEXT
4400 GET #3
4410 G=CVS(S$)
4420 FOR I=SP% TO N-1
4430 S=AV:SCSA%=SCA%+I
4440 FOR K=1 TO PO%
4450 S=S-A(K,PO%)*(PEEK(SCSA%-K)-AV)
4460 NEXT K
4470 S=S+G*U(I)
4480 IF S>255 THEN S=255 ELSE IF S<0 THEN S=0
4490 POKE SCSA%,S
4500 NEXT I
4510 CL%=C%
4520 NEXT NSC
4530 CLOSE #1,#2,#3
4540 E$="SNT"
4550 BSAVE F$+".SNT",TB%,NSM*100
4560 BEEP:GOTO 110
4570 '* * * TD/FD Cizim * * *
4580 IF E$="REC" THEN 4640
4590 IF RIGHT$(E$,1)="S" THEN 6100
4600 PRINT"FD VEYA TD (I'E VEYA BASKA BIR TUSA BAS) ?"
4610 A$=INKEY$:IF A$="" THEN 4610
4620 IF A$="1" THEN 5350 ELSE 4710
4630 '* TD Cizim *
4640 CLS:PRINT"* D.C. BILESEN HESAPLANIYOR ... *"
4650 AVS=0
4660 FOR I=0 TO 999
4670 AVS=AVS+PEEK(TB%+I)
4680 NEXT
4690 AV=AVS/1000
4700 GOTO 4740
4710 OPEN "1",#1,F$+".EPA"
4720 INPUT #1,NSM,AV
4730 CLOSE #1
4740 SCREEN 2
4750 SMN%=256*PEEK(TB%-1)+PEEK(TB%-2):SCN%=INT(SMN%/600)+1
4760 DAV=2*AV:SCC%=1
4770 GOTO 4990
```



```
4780 A$=INKEY$:IF A$="" THEN 4780
4790 IF ASC(A$)=13 THEN 110 ELSE IF LEN(A$)=1 THEN 4780
4800 B$=RIGHT$(A$,1)
4810 IF B$(">"M" THEN 4840
4820 IF SCC%=SCN% THEN SCC%=1 ELSE SCC%=SCC%+1
4830 GOTO 4990
4840 IF B$(">"K" THEN 4870
4850 IF SCC%=1 THEN SCC%=SCN% ELSE SCC%=SCC%-1
4860 GOTO 4990
4870 IF B$(">"P" THEN 4960
4880 CLS:PRINT"* EVIRME YAPILIYOR ... *"
4890 FOR I=0 TO SMN%-1
4900 SA%=TB%+I
4910 S=DAV-PEEK(SA%)
4920 IF S>255 THEN S=255 ELSE IF S<0 THEN S=0
4930 POKE SA%,S
4940 NEXT
4950 GOTO 4990
4960 IF B$(">"H" THEN 4780
4970 CLS:PRINT"SAYFA (>0,<":LOCATE 1,12:PRINT USING"###";SCN%+1
4980 LOCATE 1,14:INPUT") =";SCC%:IF SCC%<1 OR SCC%>SCN% THEN 4970
4990 CLS
5000 LOCATE 1,6:PRINT"DOSYA=";F$+"."+E$
5010 LOCATE 1,25:PRINT"SAYFA=";USING"## ";SCC%
5020 LOCATE 1,33:IF SCC%=SCN% THEN PRINT"(SON)"
5030 LOCATE 1,39:PRINT"D.C.=";USING"### ";AV
5040 LOCATE 1,48:PRINT CHR$(26);"=iLERi "
5050 LOCATE 1,56:PRINT CHR$(27);"=GERi "
5060 LOCATE 1,63:PRINT CHR$(24);"=ATLA "
5070 LOCATE 1,70:PRINT CHR$(25);"=EViR "
5080 LOCATE 1,79:PRINT CHR$(26);"t"
5090 IF SCC%>1 THEN 5150
5100 LOCATE 2,3:PRINT"V";CHR$(24)
5110 LINE(39,8)-(39,199)
5120 LINE(36,196)-(38,196)
5130 Y1=196-188*100/255:LINE(36,Y1)-(38,Y1)
5140 Y2=196-188*200/255:LINE(36,Y2)-(38,Y2)
5150 IF SCC%<SCN% THEN 5190
5160 IM=SMN%-(SCC%-1)*600
5170 IF IM<600 THEN LINE (40+IM,8)-(40+IM,199)
5180 GOTO 5200
5190 IM=600
5200 LINE (39,196)-(39+IM,196)
5210 FOR J=0 TO IM STEP 10
5220 PSET(39+J,197)
5230 NEXT
5240 FOR J=0 TO IM STEP 100
5250 K=39+J:PSET(K,198):PSET(K,199)
5260 NEXT
5270 YAV=196-188*AV/255:LINE (39,YAV)-(39+IM,YAV),, &HAAAA
5280 SCA%=TB%+(SCC%-1)*600
5290 PSET(40,196-188*PEEK(SCA%)/255)
5300 FOR I=1 TO IM-1
```



```
5310 LINE -(40+1,196-198*PEEK(SCA%+1))/255)
5320 NEXT
5330 BEEP:GOTO 4780
5340 '* FFT VE FD Cizim *
5350 CLS
5360 PRINT "* FFT UYGULANIYOR ... *"
5370 E$=LEFT$(E$,1)+"FS"
5380 OPEN "I",#1,F$+".EPA"
5390 INPUT #1,NSM,AV
5400 OPEN "R",#2,F$+".SUV",1
5410 FIELD #2,1 AS C$
5420 OPEN "R",#3,F$+". "+E$,1
5430 FIELD #3,1 AS I$
5440 FOR NSC=2 TO NSM-1
5450 GET #2
5460 C%=ASC(C$)
5470 IF C%=0 THEN 6070
5480 SCA%=TB%+(NSC-2)*100
5490 FOR K=0 TO 299
5500 AR(K+1)=(.54+.46*COS(DPI*(K-299/2)/300))*(PEEK(SCA%+K)-AV)
5510 AI(K+1)=0
5520 NEXT
5530 FOR K=301 TO 512
5540 AR(K)=0
5550 AI(K)=0
5560 NEXT
5570 '* FFT *
5580 M=9
5590 N=2^M
5600 NV2=N/2
5610 NM1=N-1
5620 J=1
5630 FOR I=1 TO NM1
5640 IF I>J THEN 5680
5650 TR=AR(J)
5660 AR(J)=AR(I)
5670 AR(I)=TR
5680 K=NV2
5690 IF K>J THEN 5730
5700 J=J-K
5710 K=K/2
5720 GOTO 5690
5730 J=J+K
5740 NEXT I
5750 FOR L=1 TO M
5760 LE=2^L
5770 LE1=LE/2
5780 UR=1!
5790 UI=0!
5800 WR=COS(PI/LE1)
5810 WI=-SIN(PI/LE1)
5820 FOR J=1 TO LE1
5830 FOR I=J TO N STEP LE
```



```

5840 IP=I+LE1
5850 TR=AR(IP)*UR-A1(IP)*UI
5860 TI=AR(IP)*UI+A1(IP)*UR
5870 AR(IP)=AR(I)-TR
5880 A1(IP)=A1(I)-TI
5890 AR(I)=AR(I)+TR
5900 A1(I)=A1(I)+TI
5910 NEXT I
5920 UR=UR*WR-UI*WI
5930 UI=UR*WI+UI*WR
5940 NEXT J,L
5950 '* * * *
5960 FSM=0
5970 FOR I=1 TO 256 STEP 4
5980 T=SQR(AR(I)^2+A1(I)^2)
5990 IF T>1 THEN AR(I)=DDB*LOG(T) ELSE AR(I)=0
6000 IF FSM<AR(I) THEN FSM=AR(I)
6010 NEXT
6020 IF FSM=0 THEN 6070
6030 FOR I=1 TO 256 STEP 4
6040 LSET I$=CHR$(CINT(8*AR(I)/FSM))
6050 PUT #3
6060 NEXT
6070 NEXT NSC
6080 CLOSE #1,#2,#3
6090 '* FD Cizim *
6100 OPEN "I",#1,F$+".EPA"
6110 INPUT #1,NSM
6120 OPEN "R",#2,F$+".SUV",1
6130 FIELD #2,1 AS C$
6140 OPEN "R",#3,F$+". "+E$,1
6150 FIELD #3,1 AS I$
6160 SCREEN 2
6170 DATA &HFF00,&HFF00,&HFF00
6180 DATA &HAA00,&HFF00,&HAA00
6190 DATA &HAA00,&HDD00,&HAA00
6200 DATA &HAA00,&HAA00,&HDD00
6210 DATA &HAA00,&HAA00,&HAA00
6220 DATA &HAA00,&H0000,&HAA00
6230 DATA &H8800,&H2200,&H0000
6240 DATA &H8800,&H0000,&H2200
6250 FOR Y%=0 TO 35 STEP 5
6260 READ L1,L2,L3
6270 LINE (0,Y%)-(4,Y%),,L1
6280 LINE (0,Y%+1)-(4,Y%+1),,L2
6290 LINE (0,Y%+2)-(4,Y%+2),,L3
6300 LINE (0,Y%+3)-(4,Y%+3),,L2
6310 LINE (0,Y%+4)-(4,Y%+4),,L1
6320 NEXT
6330 GET (0,0)-(4,4),18:GET (0,5)-(4,9),17
6340 GET (0,10)-(4,14),16:GET (0,15)-(4,19),15
6350 GET (0,20)-(4,24),14:GET (0,25)-(4,29),13
6360 GET (0,30)-(4,34),12:GET (0,35)-(4,39),11

```



```
6370 RESTORE
6380 PN%=INT(NSM/31)
6390 IF NSM-PN%*31>0 THEN PN%=PN%+1
6400 FOR P%=1 TO PN%
6410 CLS
6420 LOCATE 25,8:PRINT "DOSİA = ";F$+"."+E$
6430 LOCATE 25,30:PRINT "SAYFA = ";LOCATE 25,38:PRINT USING"###";P%
6440 LOCATE 25,40:IF P%=PN% THEN PRINT "(SON)"
6450 IF P%=1 THEN YS%=3 ELSE YS%=0
6460 IF P%>1 OR PN%=1 THEN 6480
6470 YF%=183:GOTO 6490
6480 IF P%=PN% THEN YF%=(NSM-(PN%-1)*31-1)*6+YS% ELSE YF%=180
6490 LOCATE 25,2:PRINT CHR$(25);"t"
6500 IF P%>1 THEN 6570
6510 LOCATE 2,50:PRINT "5 KHz"
6520 LOCATE 3,50:PRINT CHR$(26);"f"
6530 LINE (0,1)-(431,1)
6540 FOR X%=8 TO 380 STEP 6
6550 PSET (X%,0)
6560 NEXT
6570 LINE (1,0)-(1,199):LINE (387,0)-(387,199)
6580 FOR Y%=YS% TO YF% STEP 6
6590 PSET (0,Y%+5):PSET (388,Y%+5)
6600 GET #2
6610 C%=ASC(C$)
6620 IF C%=0 THEN 6760
6630 FOR X%=3 TO 381 STEP 6
6640 GET #3
6650 ON ASC(I$) GOTO 6670,6680,6690,6700,6710,6720,6730,6740
6660 GOTO 6750
6670 PUT (X%,Y%),11:GOTO 6750
6680 PUT (X%,Y%),12:GOTO 6750
6690 PUT (X%,Y%),13:GOTO 6750
6700 PUT (X%,Y%),14:GOTO 6750
6710 PUT (X%,Y%),15:GOTO 6750
6720 PUT (X%,Y%),16:GOTO 6750
6730 PUT (X%,Y%),17:GOTO 6750
6740 PUT (X%,Y%),18
6750 NEXT X%
6760 NEXT Y%
6770 BEEP
6780 A$=INKEY$:IF A$="" THEN 6780 ELSE NEXT P%
6790 CLOSE #1,#2,#3
6800 GOTO 110
```



```

C * * * FIR Design Program * * *
C
C PROGRAM FOR THE DESIGN OF LINEAR PHASE FINITE IMPULSE
C RESPONSE (FIR) FILTERS USING THE Remez EXCHANGE ALGORITHM
C JIM MCCALLAN, RICE UNIVERSITY, APRIL 13, 1973
C THREE TYPES OF FILTERS ARE INCLUDED--BANDPASS FILTERS
C DIFFERENTIATORS, AND WILBERT TRANSFORM FILTERS
C
C EK-C FORTRAN DILI FIR TASARIM YONERGESI
C
C THE INPUT DATA CONSISTS OF A FILE. THERE ARE FOUR AREAS
C IN THE FILE.
C
C REC. 1--FILTER LENGTH, TYPE OF FILTER, 1-BANDPASS
C PASSBAND/STOPBAND, 2-DIFFERENTIATOR, 3-WILBERT TRANSFORM
C FILTER. NUMBER OF BANDS, FILE WRITE DESIGN, AND GRID
C DENSITY.
C
C REC. 2--BAND EDGES, LOWER AND UPPER EDGES FOR EACH BAND
C WITH A MAXIMUM OF 32 BANDS.
C
C REC. 3--DESIRED FUNCTION AND DESIRED SLOPE IF A
C DIFFERENTIATOR FOR EACH BAND.
C
C REC. 4--WEIGHT FUNCTION IN EACH BAND. FOR A
C DIFFERENTIATOR, THE WEIGHT FUNCTION IS INVERSELY
C PROPORTIONAL TO F.
C
C THE FOLLOWING INPUT DATA SPECIFIES A LENGTH 32 BANDPASS
C FILTER WITH STOPBANDS 0 TO 0.1 AND 0.425 TO 0.5 AND
C PASSBAND FROM 0.1 TO 0.35 WITH WEIGHTING OF 10 IN THE
C STOPBANDS AND 1 IN THE PASSBAND. THE IMPULSE RESPONSE
C WILL BE WRITTEN A FILE AND THE GRID DENSITY IS 32.
C
C SAMPLE INPUT DATA SETUP
C 32,1,1,1,32
C 0,0.1,0.1,0.35,0.425,0.5
C 0.1,0
C 10,1/10
C
C THE FOLLOWING INPUT DATA SPECIFIES A LENGTH 32 WILBERT
C DIFFERENTIATOR WITH SLOPE 1 AND HEIGHTING OF 1.0. THE
C IMPULSE RESPONSE WILL BE WRITTEN A FILE AND THE GRID
C DENSITY IS ASSUMED TO BE 16.
C
C 32,3,1,0,0
C 0,0,0
C 1,0
C 1,0
C
C CHARACTER 1001,11,100100,10,1001,11
C COMMON P/2,AL,DIG,1,1,GRID,DIG,AT,ALPHA,TEXT,INCH,INCH
C DIMENSION TETTA(100),ALPHA(100),ALPHA(100),1,000,1,000

```



```

C
C *** FIR Design Program ***
C
C PROGRAM FOR THE DESIGN OF LINEAR PHASE FINITE IMPULSE
C RESPONSE (FIR) FILTERS USING THE REMEZ EXCHANGE ALGORITHM
C JIM MCCLELLAN, RICE UNIVERSITY, APRIL 13, 1973
C THREE TYPES OF FILTERS ARE INCLUDED--BANDPASS FILTERS
C DIFFERENTIATORS, AND HILBERT TRANSFORM FILTERS
C
C THE INPUT DATA CONSISTS OF A FILE. THERE ARE FOUR RECORD
C IN THE FILE.
C
C REC. 1--FILTER LENGTH, TYPE OF FILTER. 1-MULTIPLE
C PASSBAND/STOPBAND, 2-DIFFERENTIATOR, 3-HILBERT TRANSFORM
C FILTER. NUMBER OF BANDS, FILE WRITE DESIRED, AND GRID
C DENSITY.
C
C REC. 2--BANDEDGES, LOWER AND UPPER EDGES FOR EACH BAND
C WITH A MAXIMUM OF 10 BANDS.
C
C REC. 3--DESIRED FUNCTION (OR DESIRED SLOPE IF A
C DIFFERENTIATOR) FOR EACH BAND.
C
C REC. 4--WEIGHT FUNCTION IN EACH BAND. FOR A
C DIFFERENTIATOR, THE WEIGHT FUNCTION IS INVERSELY
C PROPORTIONAL TO F.
C
C THE FOLLOWING INPUT DATA SPECIFIES A LENGTH 32 BANDPASS
C FILTER WITH STOPBANDS 0 TO 0.1 AND 0.425 TO 0.5, AND
C PASSBAND FROM 0.2 TO 0.35 WITH WEIGHTING OF 10 IN THE
C STOPBANDS AND 1 IN THE PASSBAND. THE IMPULSE RESPONSE
C WILL BE WRITE A FILE AND THE GRID DENSITY IS 32.
C
C SAMPLE INPUT DATA SETUP
C 32,1,3,1,32
C 0,0.1,0.2,0.35,0.425,0.5
C 0,1,0
C 10,1,10
C
C THE FOLLOWING INPUT DATA SPECIFIES A LENGTH 32 WIDEBAND
C DIFFERENTIATOR WITH SLOPE 1 AND WEIGHTING OF 1/F. THE
C IMPULSE RESPONSE WILL NOT BE WRITE A FILE AND THE GRID
C DENSITY IS ASSUMED TO BE 16.
C
C 32,2,1,0,0
C 0,0.5
C 1.0
C 1.0
C
C CHARACTER PRFL *12,VSTATUS *3,VANF *1
C COMMON P12,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
C DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)

```



```
DIMENSION H(66)
DIMENSION DES(1045),GRID(1045),WT(1045)
DIMENSION EDGE(20),FX(10),WTX(10),DEVIAT(10)
DOUBLE PRECISION PI2,PI
DOUBLE PRECISION AD,DEV,X,Y
PI2=6.283185307179586
PI=3.141592653589793

C
C THE PROGRAM IS SET UP FOR A MAXIMUM LENGTH OF 128, BUT
C THIS UPPER LIMIT CAN BE CHANGED BY REDIMENSIONING THE
C ARRAYS IEXT, AD, ALPHA, X, Y, H TO BE NFMAX/2 + 2.
C THE ARRAYS DES, GRID, AND WT MUST DIMENSIONED
C 16(NFMAX/2 + 2).
C
      NFMAX=128
100 CONTINUE
      JTYPE=0

C
C PROGRAM INPUT SECTION
C
      WRITE (*,*) 'Type in FILE , STATUS (NEW/OLD).'
      READ (*,*) PRFL,VSTATUS
      IF(VSTATUS.EQ.'OLD') GO TO 110
      WRITE (*,*) 'Type in NFILT , JTYPE , NBANDS , JPUNCH , LGRID.'
      READ (*,*) NFILT,JTYPE,NBANDS,JPUNCH,LGRID
      JB=2*NBANDS
      WRITE (*,*) 'Type in EDGE(J).'
      READ (*,*) (EDGE(J),J=1,JB)
      WRITE (*,*) 'Type in FX(J).'
      READ (*,*) (FX(J),J=1,NBANDS)
      WRITE (*,*) 'Type in WTX(J).'
      READ (*,*) (WTX(J),J=1,NBANDS)
      WRITE (*,*) 'OPEN A NEW FILE (Y/N) ?'
      READ (*,*) VANF
      IF(VANF.NE.'Y') GO TO 120

C
      OPEN (10,FILE=PRFL,STATUS='NEW',FORM='UNFORMATTED')
      WRITE (10) NFILT,JTYPE,NBANDS,JPUNCH,LGRID
      WRITE (10) (EDGE(J),J=1,JB)
      WRITE (10) (FX(J),J=1,NBANDS)
      WRITE (10) (WTX(J),J=1,NBANDS)
      CLOSE (10)
      GO TO 120

C
110 OPEN (10,FILE=PRFL,STATUS='OLD',FORM='UNFORMATTED')
      READ (10) NFILT,JTYPE,NBANDS,JPUNCH,LGRID
      JB=2*NBANDS
      READ (10) (EDGE(J),J=1,JB)
      READ (10) (FX(J),J=1,NBANDS)
      READ (10) (WTX(J),J=1,NBANDS)
      CLOSE (10)

C
120 IF(NFILT.GT.NFMAX.OR.NFILT.LT.3) CALL ERROR
```



```

      IF(NBANDS.LE.0) NBANDS=1
C
C   GRID DENSITY IS ASSUMED TO BE 16 UNLESS SPECIFIED
C   OTHERWISE
C
      IF(LGRID.LE.0) LGRID=16
      IF(JTYPE.EQ.0) CALL ERROR
      NEG=1
      IF(JTYPE.EQ.1) NEG=0
      NODD=NFILT/2
      NODD=NFILT-2*NODD
      NFCNS=NFILT/2
      IF(NODD.EQ.1.AND.NEG.EQ.0) NFCNS=NFCNS+1
C
C   SET UP THE DENSE GRID.  THE NUMBER OF POINTS IN THE GRID
C   IS (FILTER LENGTH + 1) *GRID DENSITY/2
C
      GRID(1)=EDGE(1)
      DELF=LGRID*NFCNS
      DELF=0.5/DELF
      IF(NEG.EQ.0) GO TO 135
      IF(EDGE(1).LT.DEF) GRID(1)=DELF
135  CONTINUE
      J=1
      L=1
      LBAND=1
140  FUP=EDGE(L+1)
145  TEMP=GRID(J)
C
C   CALCULATE THE DESIRED MAGNITUDE RESPONSE AND THE WEIGHT
C   FUNCTION ON THE GRID
C
      DES(J)=EFF(TEMP,FX,WTX,LBAND,JTYPE)
      WT(J)=WATE(TEMP,FX,WTX,LBAND,JTYPE)
      J=J+1
      GRID(J)=TEMP+DELF
      IF(GRID(J).GT.FUP) GO TO 150
      GO TO 145
150  GRID(J-1)=FUP
      DES(J-1)=EFF(FUP,FX,WTX,LBAND,JTYPE)
      WT(J-1)=WATE(FUP,FX,WTX,LBAND,JTYPE)
      LBAND=LBAND+1
      L=L+2
      IF(LBAND.GT.NBANDS) GO TO 160
      GRID(J)=EDGE(L)
      GO TO 140
160  NGRID=J-1
      IF(NEG.NE.NODD) GO TO 165
      IF(GRID(NGRID).GT.(0.5-DELF)) NGRID=NGRID-1
165  CONTINUE
C
C   SET UP A NEW APPROXIMATION PROBLEM WHICH IS EQUIVALENT
C   TO THE ORIGINAL PROBLEM

```



```

C
  IF(NEG) 170,170,180
170 IF(NODD.EQ.1) GO TO 200
    DO 175 J=1,NGRID
      CHANGE=DCOS(PI*GRID(J))
      DES(J)=DES(J)/CHANGE
175 WT(J)=WT(J)*CHANGE
    GO TO 200
180 IF(NODD.EQ.1) GO TO 190
    DO 185 J=1,NGRID
      CHANGE=DSIN(PI*GRID(J))
      DES(J)=DES(J)/CHANGE
185 WT(J)=WT(J)*CHANGE
    GO TO 200
190 DO 195 J=1,NGRID
      CHANGE=DSIN(PI2*GRID(J))
      DES(J)=DES(J)/CHANGE
195 WT(J)=WT(J)*CHANGE
C
C INITIAL GUESS FOR THE EXTREMAL FREQUENCIES--EQUALLY
C SPACED ALONG THE GRID
C
200 TEMP=FLOAT(NGRID-1)/FLOAT(NFCNS)
    DO 210 J=1,NFCNS
210 IEXT(J)=(J-1)*TEMP+1
    IEXT(NFCNS+1)=NGRID
    NM1=NFCNS-1
    NZ=NFCNS+1
C
C CALL THE REMEZ EXCHANGE ALGORITHM TO DO THE APPROXIMATION
C PROBLEM
C
    CALL REMEZ(EDGE,NBANDS)
C
C CALCULATE THE IMPULSE RESPONSE.
C
  IF(NEG) 300,300,320
300 IF(NODD.EQ.0) GO TO 310
    DO 305 J=1,NM1
305 H(J)=0.5*ALPHA(NZ-J)
    H(NFCNS)=ALPHA(1)
    GO TO 350
310 H(1)=0.25*ALPHA(NFCNS)
    DO 315 J=2,NM1
315 H(J)=0.25*(ALPHA(NZ-J)+ALPHA(NFCNS+2-J))
    H(NFCNS)=0.5*ALPHA(1)+0.25*ALPHA(2)
    GO TO 350
320 IF(NODD.EQ.0) GO TO 330
    H(1)=0.25*ALPHA(NFCNS)
    H(2)=0.25*ALPHA(NM1)
    DO 325 J=3,NM1
325 H(J)=0.25*(ALPHA(NZ-J)-ALPHA(NFCNS+3-J))
    H(NFCNS)=0.5*ALPHA(1)-0.25*ALPHA(3)

```



```

      H(NZ)=0.0
      GO TO 350
330  H(1)=0.25*ALPHA(NFCNS)
      DO 335 J=2,NM1
335  H(J)=0.25*(ALPHA(NZ-J)-ALPHA(NFCNS+2-J))
      H(NFCNS)=0.5*ALPHA(1)-0.25*ALPHA(2)
C
C  PROGRAM OUTPUT SECTION.
C
350  WRITE (*,*) 'VDU , PRN or a FILE (CON/LPT1 or 2 /FILE) ?'
      READ (*,*) PRFL
      WRITE (*,*) 'Type in STATUS (NEW/OLD) ?'
      READ (*,*) VSTATUS
      IF(VSTATUS.EQ.'NEW') THEN
      OPEN (20,FILE=PRFL,STATUS='NEW')
      ELSE
      OPEN (20,FILE=PRFL)
      END IF
355  WRITE (20,360)
360  FORMAT(70(1H*)//25X,'FINITE IMPULSE RESPONSE (FIR)'/
+ 25X,'LINEAR PHASE DIGITAL FILTER DESIGN'/
+ 25X,'REMEZ EXCHANGE ALGORITHM'/)
      IF(JTYPE.EQ.1) WRITE (20,365)
365  FORMAT(25X,'BANDPASS FILTER'/)
      IF(JTYPE.EQ.2) WRITE (20,370)
370  FORMAT(25X,'DIFFERENTIATOR'/)
      IF(JTYPE.EQ.3) WRITE (20,375)
375  FORMAT(25X,'HILBERT TRANSFORMER'/)
      WRITE (20,378) NFILT
378  FORMAT(15X,'FILTER LENGTH = ',13/)
      WRITE (20,380)
380  FORMAT(15X,'* * * * * IMPULSE RESPONSE * * * * *')
      DO 381 J=1,NFCNS
      K=NFILT+1-J
      IF(NEG.EQ.0) WRITE (20,382) J,H(J),K
      IF(NEG.EQ.1) WRITE (20,383) J,H(J),K
381  CONTINUE
382  FORMAT(20X,'H(',13,') = ',E15.8,' = H(',14,')')
383  FORMAT(20X,'H(',13,') = ',E15.8,' = -H(',14,')')
      IF(NEG.EQ.1.AND.NODD.EQ.1) WRITE (20,384) NZ
384  FORMAT(20X,'H(',13,') = 0.0')
      DO 450 K=1,NBANDS,4
      KUP=K+3
      IF(KUP.GT.NBANDS) KUP=NBANDS
      WRITE (20,385) ('BAND',J,J=K,KUP)
385  FORMAT(/24X,4(A4,13,8X))
      WRITE (20,390) (EDGE(2*J-1),J=K,KUP)
390  FORMAT(2X,'LOWER BAND EDGE',5F15.9)
      WRITE (20,395) (EDGE(2*J),J=K,KUP)
395  FORMAT(2X,'UPPER BAND EDGE',5F15.9)
      IF(JTYPE.NE.2) WRITE (20,400) (FX(J),J=K,KUP)
400  FORMAT(2X,'DESIRED VALUE',2X,5F15.9)
      IF(JTYPE.EQ.2) WRITE (20,405) (FX(J),J=K,KUP)

```



```

405 FORMAT(2X,'DESIRED SLOPE',2X,5F15.9)
    WRITE (20,410) (WTX(J),J=K,KUP)
410 FORMAT(2X,'WEIGHTING',6X,5F15.9)
    DO 420 J=K,KUP
420 DEVIAT(J)=DEV/WTX(J)
    WRITE (20,425) (DEVIAT(J),J=K,KUP)
425 FORMAT(2X,'DEVIATION',6X,5F15.9)
    IF(JTYPE.NE.1) GO TO 450
    DO 430 J=K,KUP
430 DEVIAT(J)=20.0*ALOG10(DEVIAT(J))
    WRITE (20,435) (DEVIAT(J),J=K,KUP)
435 FORMAT(2X,'DEVIATION IN DB',5F15.9)
450 CONTINUE
    WRITE (20,455) (GRID(IEXT(J)),J=1,NZ)
455 FORMAT(/2X,'EXTREMAL FREQUENCIES'/(2X,5F12.7))
    WRITE (20,460)
460 FORMAT(/,70(1H*))
    CLOSE (20)
    IF(JPUNCH.EQ.0) GO TO 480
    NFCNS=NFCNS/2
    NZ=NFCNS+1
    WRITE (*,*) 'Type in H(n) output FILE.'
    READ (*,*) PRFL
    OPEN (30,FILE=PRFL,STATUS='NEW')
    WRITE (30,490) (H(J),J=1,NFCNS)
    IF(NODD.EQ.0) GO TO 470
    WRITE (30,490) H(NZ)
470 IF(NEG.NE.0) H(J)=-H(J)
    WRITE (30,490) (H(J),J=NFCNS,1,-1)
490 FORMAT(E15.8)
    CLOSE (30)
480 STOP
    END

C
    FUNCTION EFF(TEMP,FX,WTX,LBAND,JTYPE)
C
C  FUNCTION TO CALCULATE THE DESIRED MAGNITUDE RESPONSE
C  AS A FUNCTION OF FREQUENCY.
C
    DIMENSION FX(5),WTX(5)
    IF(JTYPE.EQ.2) GO TO 1
    EFF=FX(LBAND)
    RETURN
1  EFF=FX(LBAND)*TEMP
    RETURN
    END

C
    FUNCTION WATE(TEMP,FX,WTX,LBAND,JTYPE)
C
C  FUNCTION TO CALCULATE THE WEIGHT FUNCTION AS A FUNCTION
C  OF FREQUENCY.
C
    DIMENSION FX(5),WTX(5)

```



```

      IF(JTYPE.EQ.2) GO TO 1
      WATE=WTX(LBAND)
      RETURN
1    IF(FX(LBAND).LT.0.0001) GO TO 2
      WATE=WTX(LBAND)/TEMP
      RETURN
2    WATE=WTX(LBAND)
      RETURN
      END
C
      SUBROUTINE ERROR
      WRITE (*,1)
1    FORMAT(' * * * * * ERROR IN INPUT DATA * * * * *')
      STOP
      END
C
      SUBROUTINE REMEZ(EDGE,NBANDS)
C    THIS SUBROUTINE IMPLEMENTS THE REMEZ EXCHANGE ALGORITHM
C    FOR THE WEIGHTED CHEBYCHEV APPROXIMATION OF A CONTINUOUS
C    FUNCTION WITH A SUM OF COSINES.  INPUTS TO THE SUBROUTINE
C    ARE A DENSE GRID WHICH REPLACES THE FREQUENCY AXIS, THE
C    DESIRED FUNCTION ON THIS GRID, THE WEIGHT FUNCTION ON THE
C    GRID, THE NUMBER OF COSINES, AND AN INITIAL GUESS OF THE
C    EXTREMAL FREQUENCIES.  THE PROGRAM MINIMIZES THE CHEBYCHEV
C    ERROR BY DETERMINING THE BEST LOCATION OF THE EXTREMAL
C    FREQUENCIES (POINTS OF MAXIMUM ERROR) AND THEN CALCULATES
C    THE COEFFICIENTS OF THE BEST APPROXIMATION.
C
      COMMON PI2,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
      DIMENSION EDGE(20)
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION DES(1045),GRID(1045),WT(1045)
      DIMENSION A(66),P(65),Q(65)
      DOUBLE PRECISION PI2,DNUM,DDEN,DTEMP,A,P,Q
      DOUBLE PRECISION AD,DEV,X,Y
C
C    THE PROGRAM ALLOWS A MAXIMUM NUMBER OF ITERATIONS OF 25
C
      ITRMAX=25
      DEVL=-1.0
      NZ=NFCNS+1
      NZZ=NFCNS+2
      NITER=0
100  CONTINUE
      IEXT(NZZ)=NGRID+1
      NITER=NITER+1
      IF(NITER.GT.ITRMAX) GO TO 400
      DO 110 J=1,NZ
      DTEMP=GRID(IEXT(J))
      DTEMP=DCOS(DTEMP*PI2)
110  X(J)=DTEMP
      JET=(NFCNS-1)/15+1
      DO 120 J=1,NZ

```



```

120 AD(J)=D(J,NZ,JET)
    DNUM=0.0
    DDEN=0.0
    K=1
    DO 130 J=1,NZ
        L=IEXT(J)
        DTEMP=AD(J)*DES(L)
        DNUM=DNUM+DTEMP
        DTEMP=K*AD(J)/WT(L)
        DDEN=DDEN+DTEMP
130 K=-K
    DEV=DNUM/DDEN
    NU=1
    IF(DEV.GT.0.0) NU=-1
    DEV=-NU*DEV
    K=NU
    DO 140 J=1,NZ
        L=IEXT(J)
        DTEMP=K*DEV/WT(L)
        Y(J)=DES(L)+DTEMP
140 K=-K
    IF(DEV.GE.DEVL) GO TO 150
    CALL OUCH
    GO TO 400
150 DEVL=DEV
    JCHNGE=0
    K1=IEXT(1)
    KNZ=IEXT(NZ)
    KLOW=0
    NUT=-NU
    J=1
C
C SEARCH FOR THE EXTREMAL FREQUENCIES OF THE BEST
C APPROXIMATION
C
200 IF(J.EQ.NZZ) YNZ=COMP
    IF(J.GE.NZZ) GO TO 300
    KUP=IEXT(J+1)
    L=IEXT(J)+1
    NUT=-NUT
    IF(J.EQ.2) Y1=COMP
    COMP=DEV
    IF(L.GE.KUP) GO TO 220
    ERR=GEE(L,NZ)
    ERR=(ERR-DES(L))*WT(L)
    DTEMP=NUT*ERR-COMP
    IF(DTEMP.LE.0.0) GO TO 220
    COMP=NUT*ERR
210 L=L+1
    IF(L.GE.KUP) GO TO 215
    ERR=GEE(L,NZ)
    ERR=(ERR-DES(L))*WT(L)
    DTEMP=NUT*ERR-COMP

```



```

      IF(DTEMP.LE.0.0) GO TO 215
      COMP=NUT*ERR
      GO TO 210
215  IEXT(J)=L-1
      J=J+1
      KLOW=L-1
      JCHNGE=JCHNGE+1
      GO TO 200
220  L=L-1
225  L=L-1
      IF(L.LE.KLOW) GO TO 250
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.GT.0.0) GO TO 230
      IF(JCHNGE.LE.0) GO TO 225
      GO TO 260
230  COMP=NUT*ERR
235  L=L-1
      IF(L.LE.KLOW) GO TO 240
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 240
      COMP=NUT*ERR
      GO TO 235
240  KLOW=IEXT(J)
      IEXT(J)=L+1
      J=J+1
      JCHNGE=JCHNGE+1
      GO TO 200
250  L=IEXT(J)+1
      IF(JCHNGE.GT.0) GO TO 215
255  L=L+1
      IF(L.GE.KUP) GO TO 260
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 255
      COMP=NUT*ERR
      GO TO 210
260  KLOW=IEXT(J)
      J=J+1
      GO TO 200
300  IF(J.GT.NZZ) GO TO 320
      IF(K1.GT.IEXT(1)) K1=IEXT(1)
      IF(KNZ.LT.IEXT(NZ)) KNZ=IEXT(NZ)
      NUT1=NUT
      NUT=-NU
      L=0
      KUP=K1
      COMP=YNZ*(1.00001)
      LUCK=1

```



```

310 L=L+1
    IF(L.GE.KUP) GO TO 315
    ERR=GEE(L,NZ)
    ERR=(ERR-DES(L))*WT(L)
    DTEMP=NUT*ERR-COMP
    IF(DTEMP.LE.0.0) GO TO 310
    COMP=NUT*ERR
    J=NZZ
    GO TO 210
315 LUCK=6
    GO TO 325
320 IF(LUCK.GT.9) GO TO 350
    IF(COMP.GT.Y1) Y1=COMP
    K1=IEXT(NZZ)
325 L=NGRID+1
    KLOW=KNZ
    NUT=-NUT1
    COMP=Y1*(1.00001)
330 L=L-1
    IF(L.LE.KLOW) GO TO 340
    ERR=GEE(L,NZ)
    ERR=(ERR-DES(L))*WT(L)
    DTEMP=NUT*ERR-COMP
    IF(DTEMP.LE.0.0) GO TO 330
    J=NZZ
    COMP=NUT*ERR
    LUCK=LUCK+10
    GO TO 235
340 IF(LUCK.EQ.6) GO TO 370
    DO 345 J=1,NFCNS
345 IEXT(NZZ-J)=IEXT(NZ-J)
    IEXT(1)=K1
    GO TO 100
350 KN=IEXT(NZZ)
    DO 360 J=1,NFCNS
360 IEXT(J)=IEXT(J+1)
    IEXT(NZ)=KN
    GO TO 100
370 IF(JCHNGE.GT.0) GO TO 100

```

C
C CALCULATION OF THE COEFFICIENTS OF THE BEST APPROXIMATION
C USING THE INVERSE DISCRETE FOURIER TRANSFORM

C
400 CONTINUE
 NM1=NFCNS-1
 FSH=1.0E-06
 GTEMP=GRID(1)
 X(NZZ)=-2.0
 CN=2*NFCNS-1
 DELF=1.0/CN
 L=1
 KKK=0
 IF(EDGE(1).EQ.0.0.AND.EDGE(2*NBANDS).EQ.0.5) KKK=1


```

IF(NFCNS.LE.3) KKK=1
IF(KKK.EQ.1) GO TO 405
DTEMP=DCOS(PI2*GRID(1))
DNUM=DCOS(PI2*GRID(NGRID))
AA=2.0/(DTEMP-DNUM)
BB=-(DTEMP+DNUM)/(DTEMP-DNUM)
405 CONTINUE
DO 430 J=1,NFCNS
FT=(J-1)*DELF
XT=DCOS(PI2*FT)
IF(KKK.EQ.1) GO TO 410
XT=(XT-BB)/AA
FT=ACOS(XT)/PI2
410 XE=X(L)
IF(XT.GT.XE) GO TO 420
IF((XE-XT).LT.FSH) GO TO 415
L=L+1
GO TO 410
415 A(J)=Y(L)
GO TO 425
420 IF((XT-XE).LT.FSH) GO TO 415
GRID(1)=FT
A(J)=GEE(1,NZ)
425 CONTINUE
IF(L.GT.1) L=L-1
430 CONTINUE
GRID(1)=GTEMP
DDEN=PI2/CN
DO 510 J=1,NFCNS
DTEMP=0.0
DNUM=(J-1)*DDEN
IF(NM1.LT.1) GO TO 505
DO 500 K=1,NM1
500 DTEMP=DTEMP+A(K+1)*DCOS(DNUM*K)
505 DTEMP=2.0*DTEMP+A(1)
510 ALPHA(J)=DTEMP
DO 550 J=2,NFCNS
550 ALPHA(J)=2*ALPHA(J)/CN
ALPHA(1)=ALPHA(1)/CN
IF(KKK.EQ.1) GO TO 545
P(1)=2.0*ALPHA(NFCNS)*BB+ALPHA(NM1)
P(2)=2.0*AA*ALPHA(NFCNS)
Q(1)=ALPHA(NFCNS-2)-ALPHA(NFCNS)
DO 540 J=2,NM1
540 IF(J.LT.NM1) GO TO 515
AA=0.5*AA
BB=0.5*BB
515 CONTINUE
P(J+1)=0.0
DO 520 K=1,J
520 A(K)=P(K)
P(K)=2.0*BB*A(K)
P(2)=P(2)+A(1)*2.0*AA

```



```

      JH1=J-1
      DO 525 K=1,JH1
525  P(K)=P(K)+Q(K)+AA*A(K+1)
      JP1=J+1
      DO 530 K=3,JP1
530  P(K)=P(K)+AA*A(K-1)
      IF(J.EQ.NM1) GO TO 540
      DO 535 K=1,J
535  Q(K)=-A(K)
      Q(1)=Q(1)+ALPHA(NFCNS-1-J)
540  CONTINUE
      DO 543 J=1,NFCNS
543  ALPHA(J)=P(J)
545  CONTINUE
      IF(NFCNS.GT.3) RETURN
      ALPHA(NFCNS+1)=0.0
      ALPHA(NFCNS+2)=0.0
      RETURN
      END
C
      DOUBLE PRECISION FUNCTION D(K,N,M)
C
C  FUNCTION TO CALCULATE THE LAGRANGE INTERPOLATION
C  COEFFICIENTS FOR USE IN THE FUNCTION THE GEE.
C
      COMMON P12,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION DES(1045),GRID(1045),WT(1045)
      DOUBLE PRECISION AD,DEV,X,Y
      DOUBLE PRECISION Q
      DOUBLE PRECISION P12
      D=1.0
      Q=X(K)
      DO 3 L=1,M
      DO 2 J=L,N,M
      IF(J-K)1,2,1
1  D=2.0*D*(Q-X(J))
2  CONTINUE
3  CONTINUE
      D=1.0/D
      RETURN
      END
C
      DOUBLE PRECISION FUNCTION GEE(K,N)
C
C  FUNCTION TO EVALUATE THE FREQUENCY RESPONSE USING THE
C  LAGRANGE INTERPOLATION FORMULA IN THE BARYCENTRIC FORM
C
      COMMON P12,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION DES(1045),GRID(1045),WT(1045)
      DOUBLE PRECISION P,C,D,XF
      DOUBLE PRECISION P12

```



```
DOUBLE PRECISION AD,DEV,X,Y
P=0.0
XF=GRID(K)
XF=DCOS(PI2*XF)
D=0.0
DO 1 J=1,N
C=XF-X(J)
C=AD(J)/C
D=D+C
1 P=P+C*Y(J)
GEE=P/D
RETURN
END
```

C

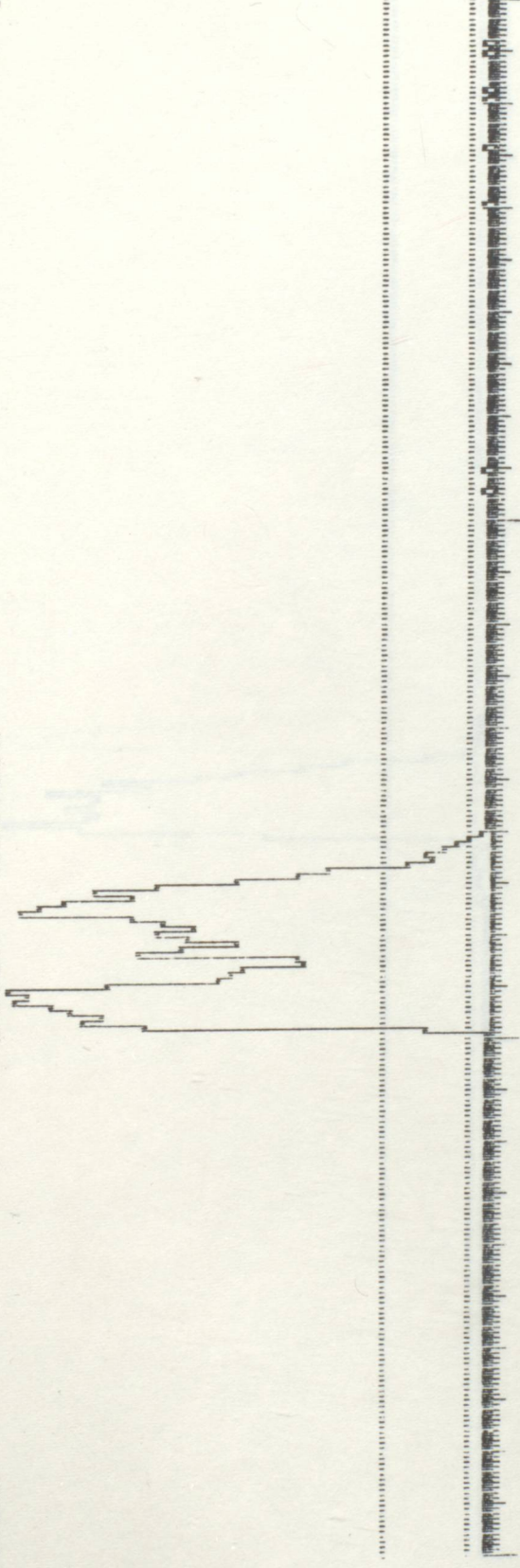
```
SUBROUTINE OUCH
WRITE (*,1)
1 FORMAT(' * * * * * FAILURE TO CONVERGE * * * * *' /
+ 'OPROBABLE CAUSE IS MACHINE ROUNDING ERROR' /
+ 'OTHE IMPULSE RESPONSE MAY BE CORRECT' ,
+ 'OCHECK WITH A FREQUENCY RESPONSE')
RETURN
END
```


EK-D UYGULAMA SONUÇLARI

ERKE

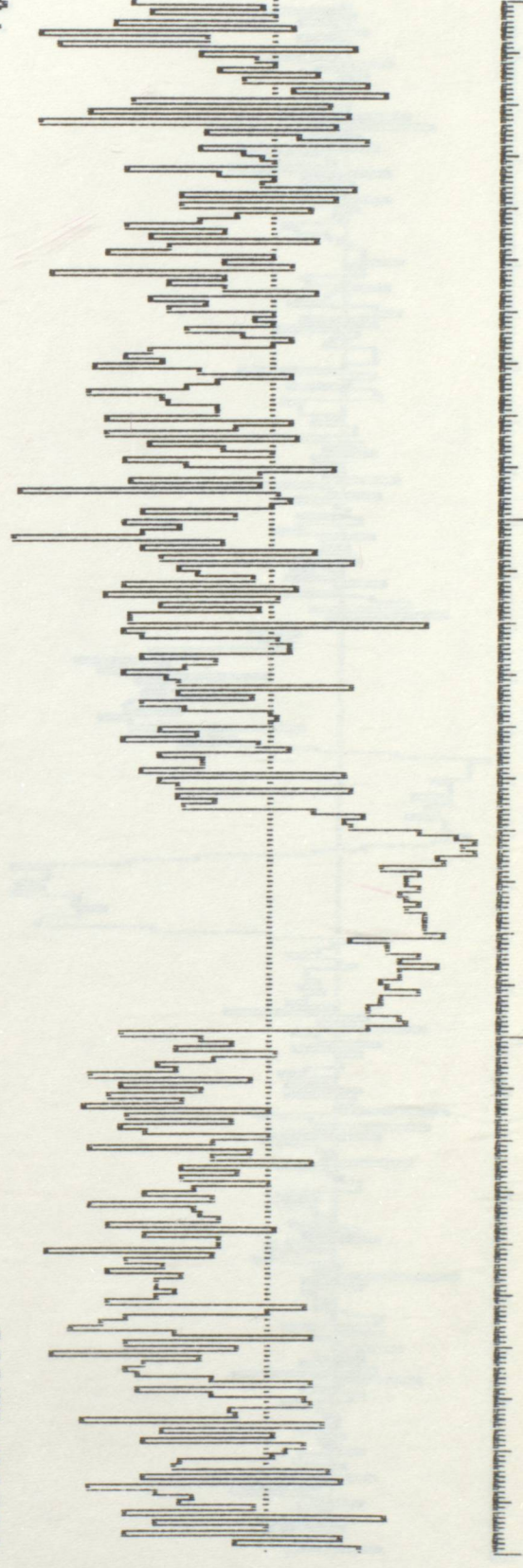
DOSYA = ALO.EPA

→t



SIFIR GECİS

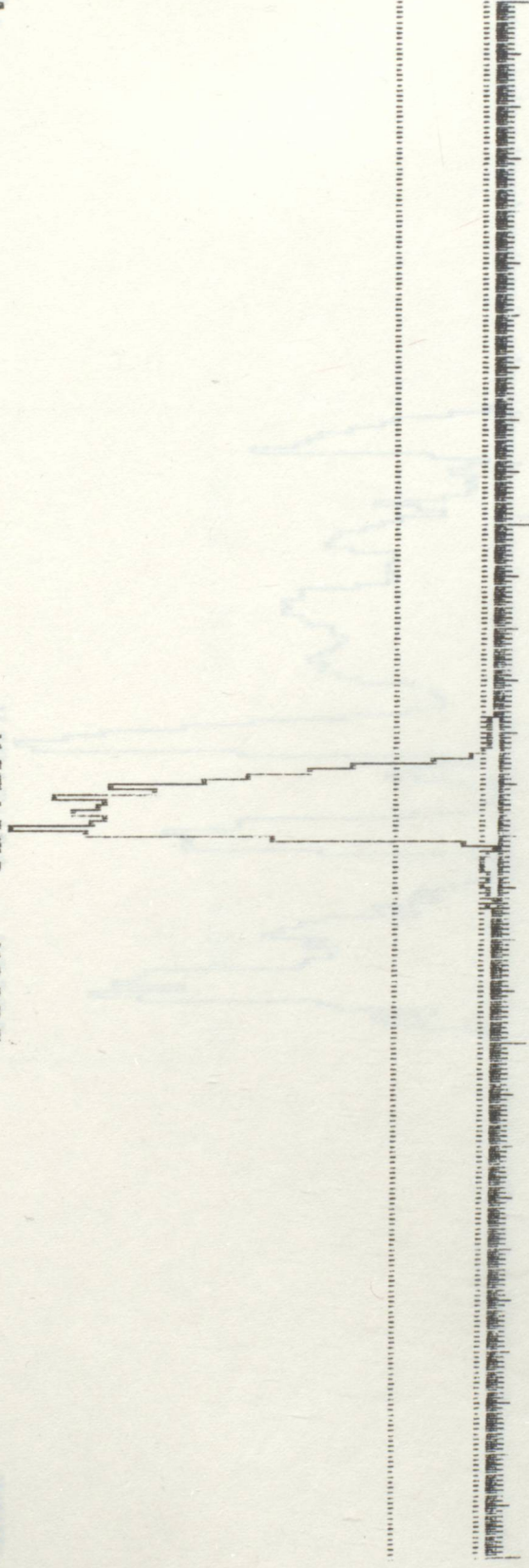
→t



ERKE

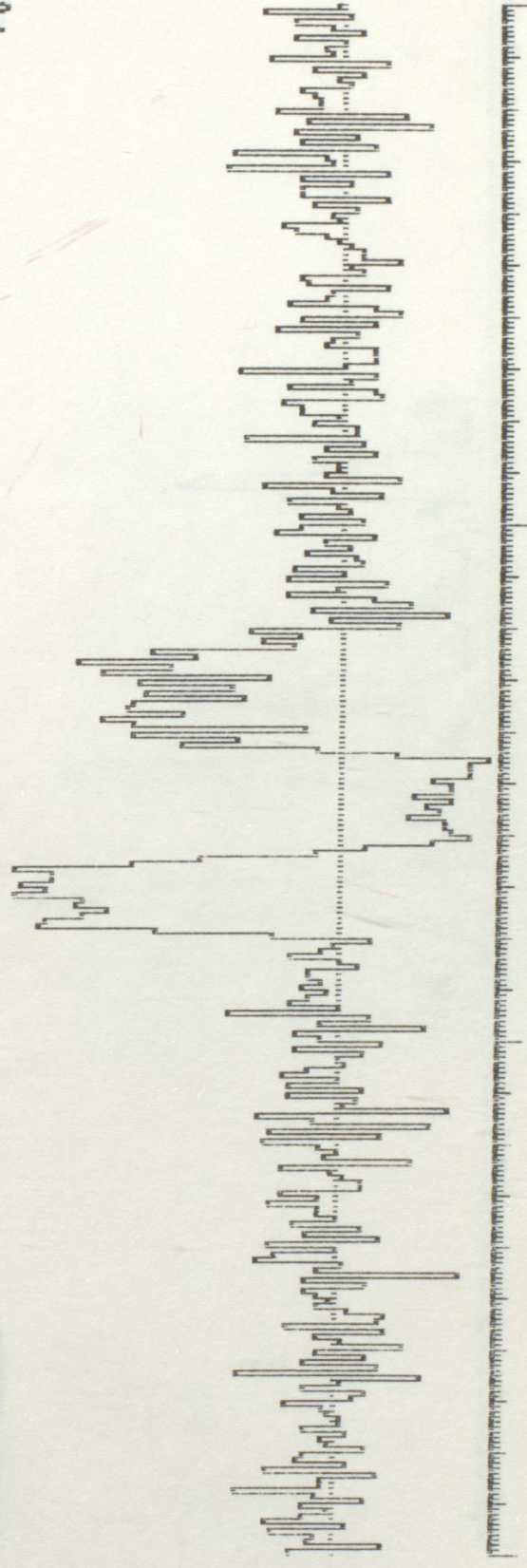
DOSYA = SES.EPA

→t



SIFIR GECİS

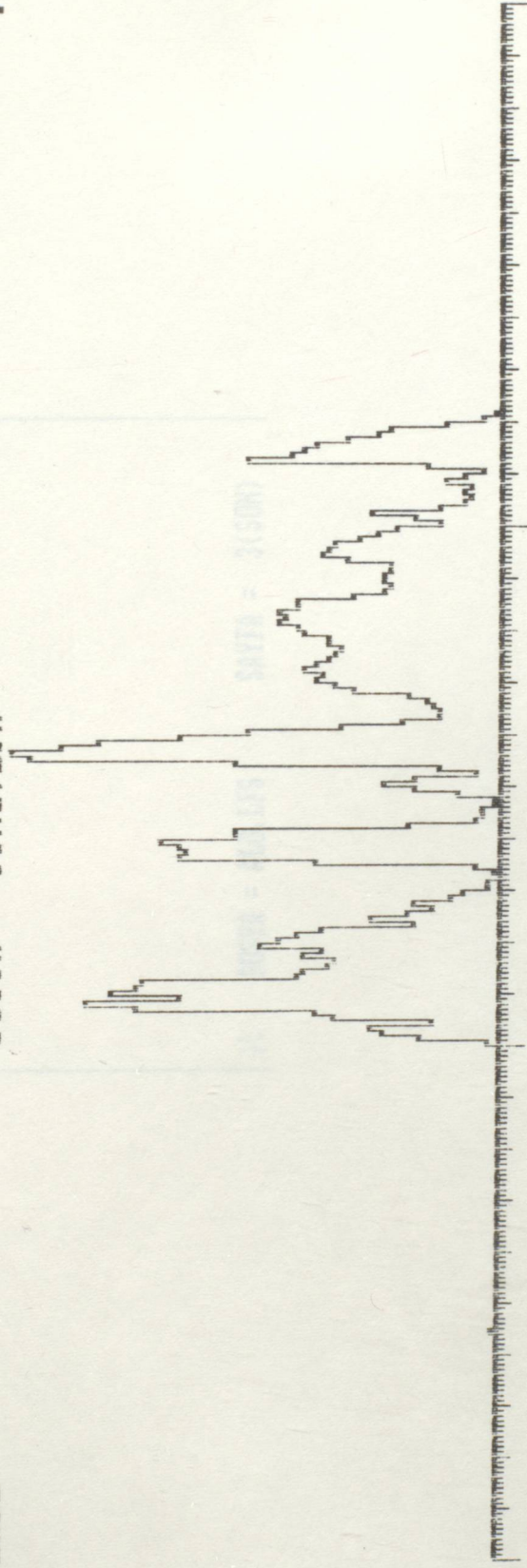
→t



ERKE

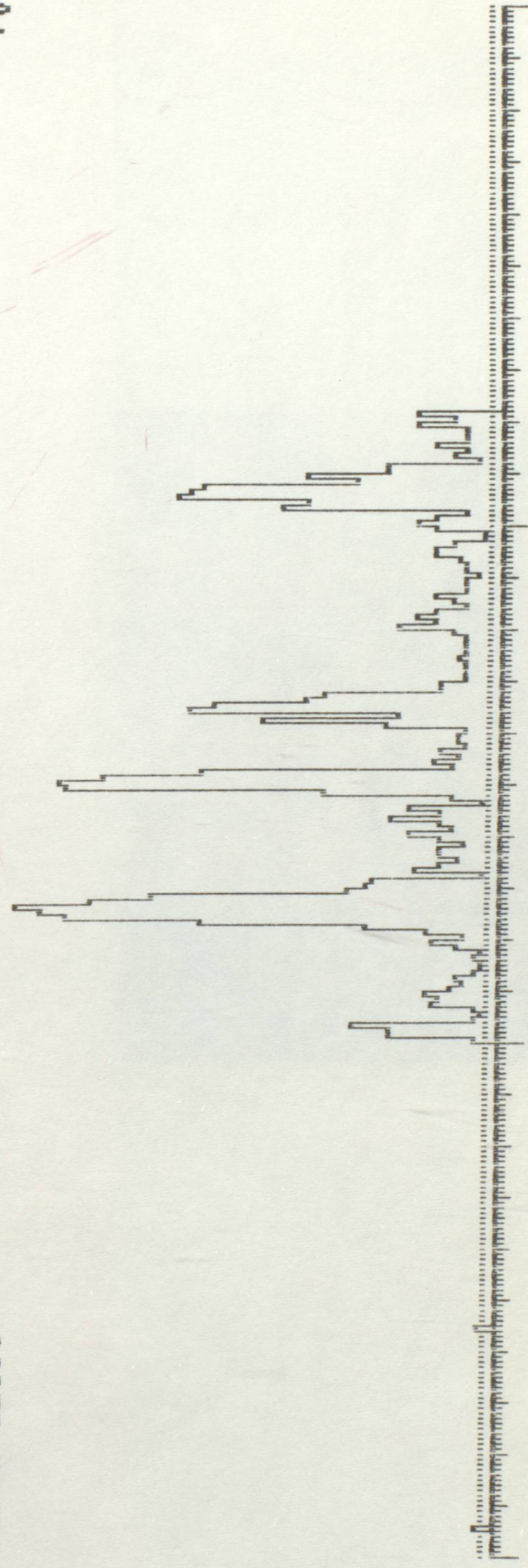
DOSYA = TUM1.EPA

→t

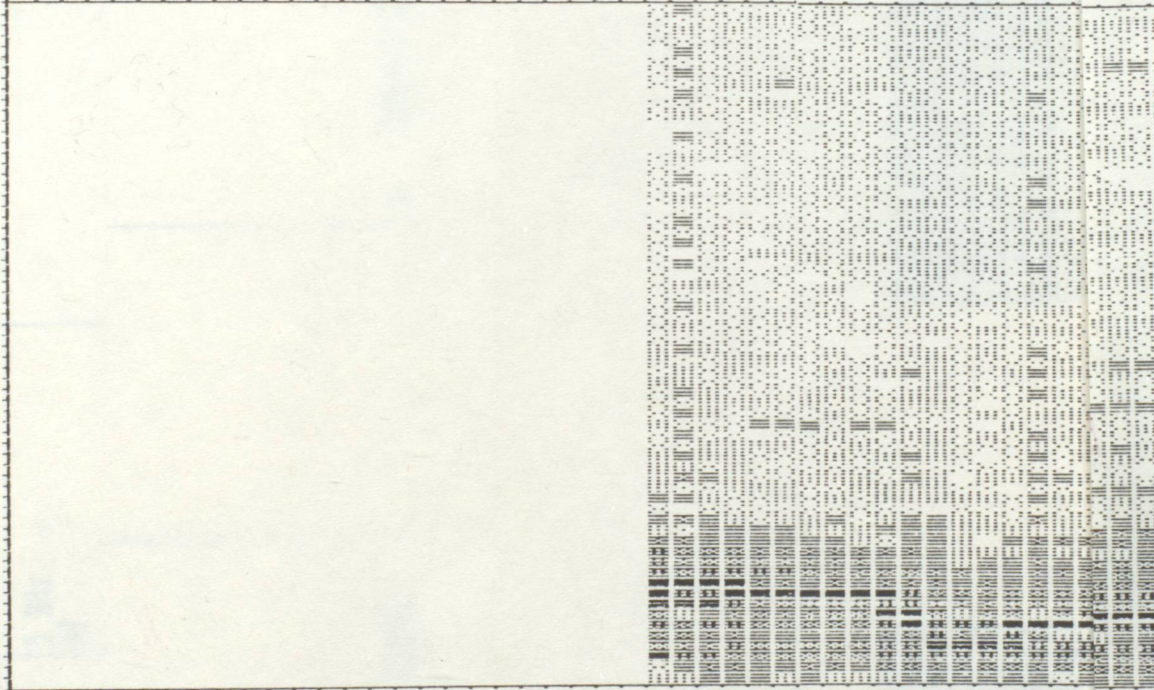


SIFIR GECİS

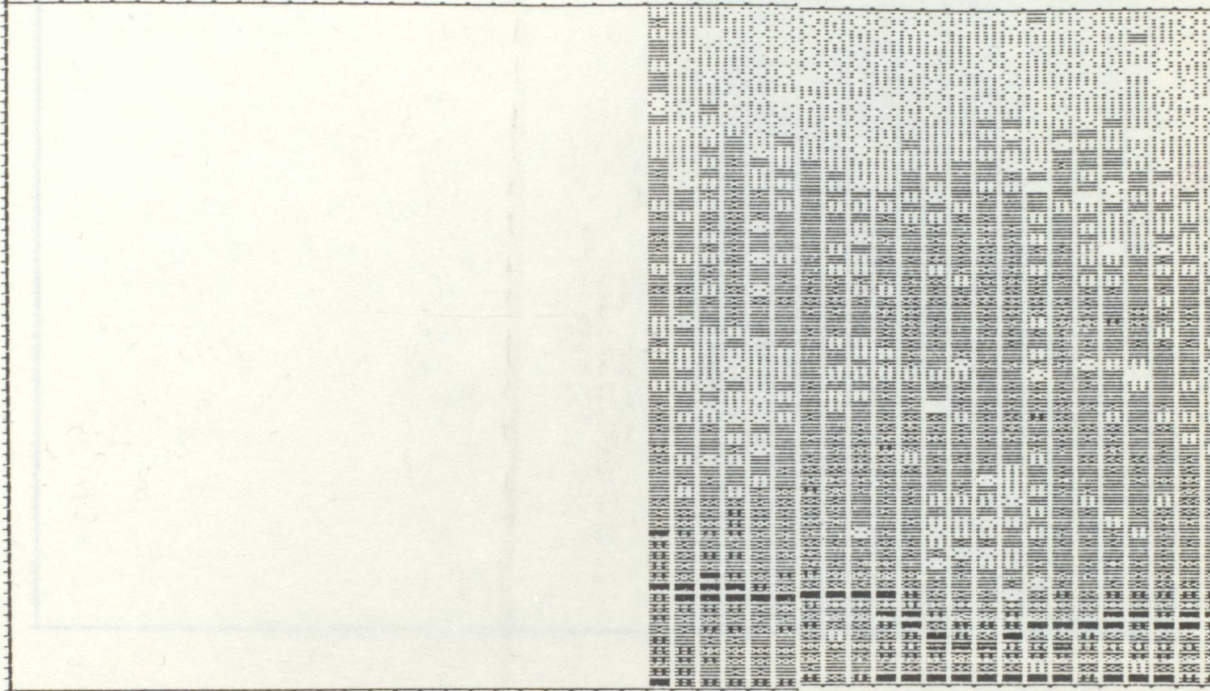
→t



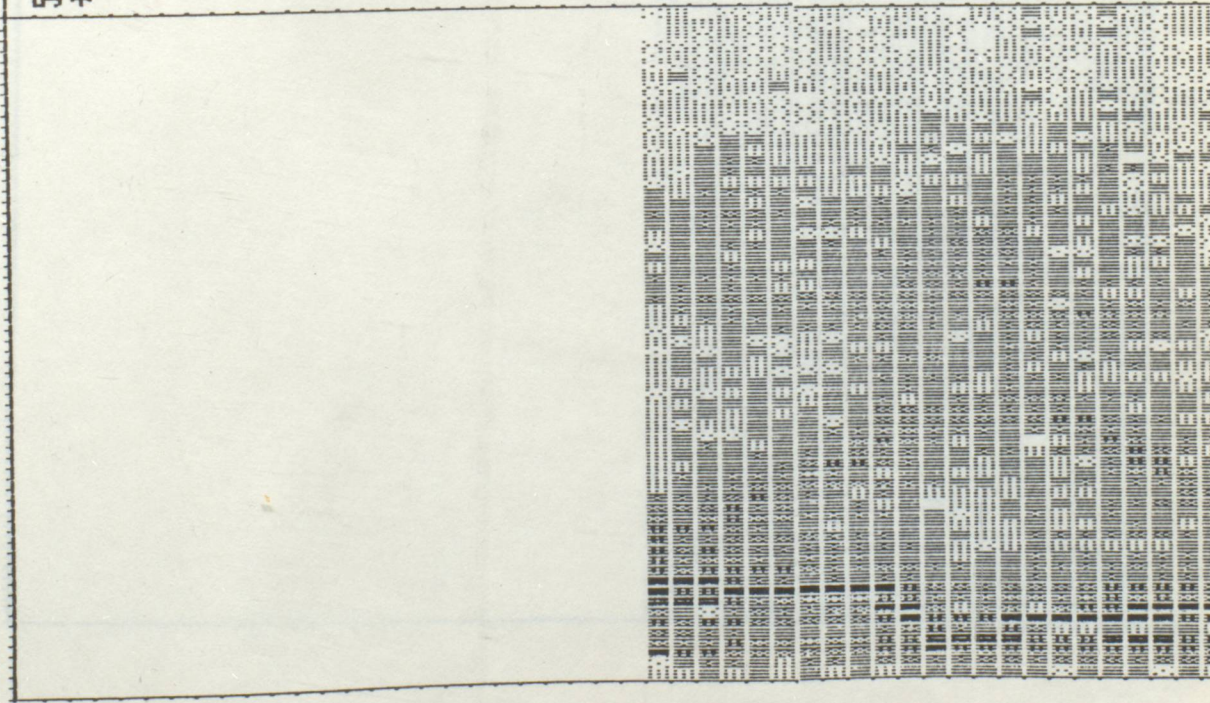
5 kHz
+f



5 KHz
→f



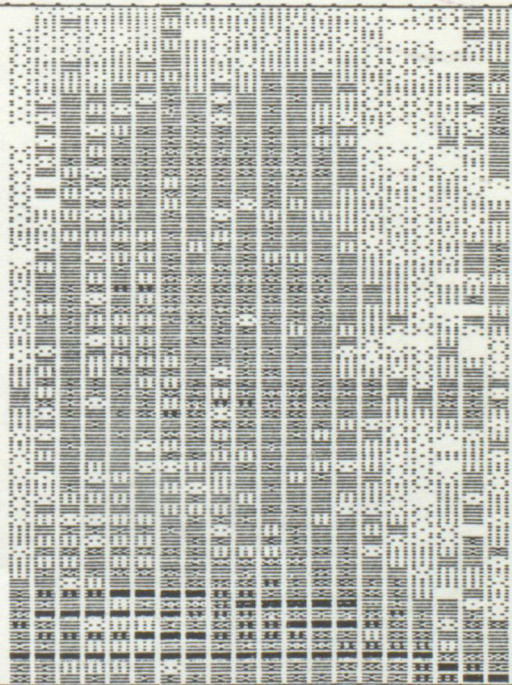
5 KHz
→f



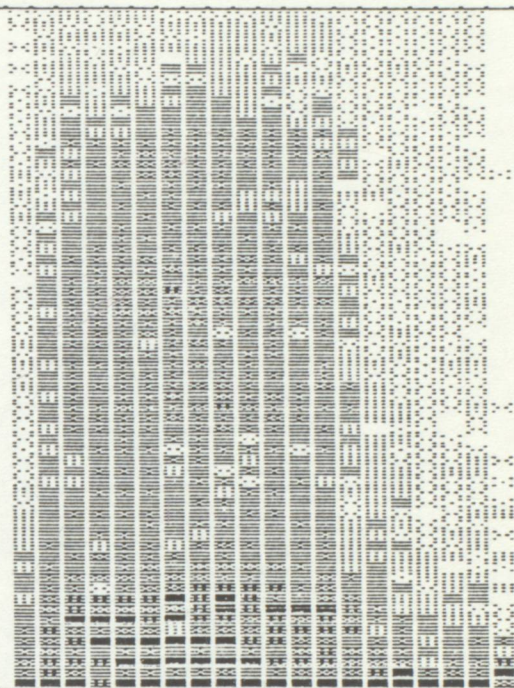
↓↑ DOSYA = ALO.OFS SAYFA = 3(SON)

↓↑ DOSYA = ALO.SFS SAYFA = 3(SON)

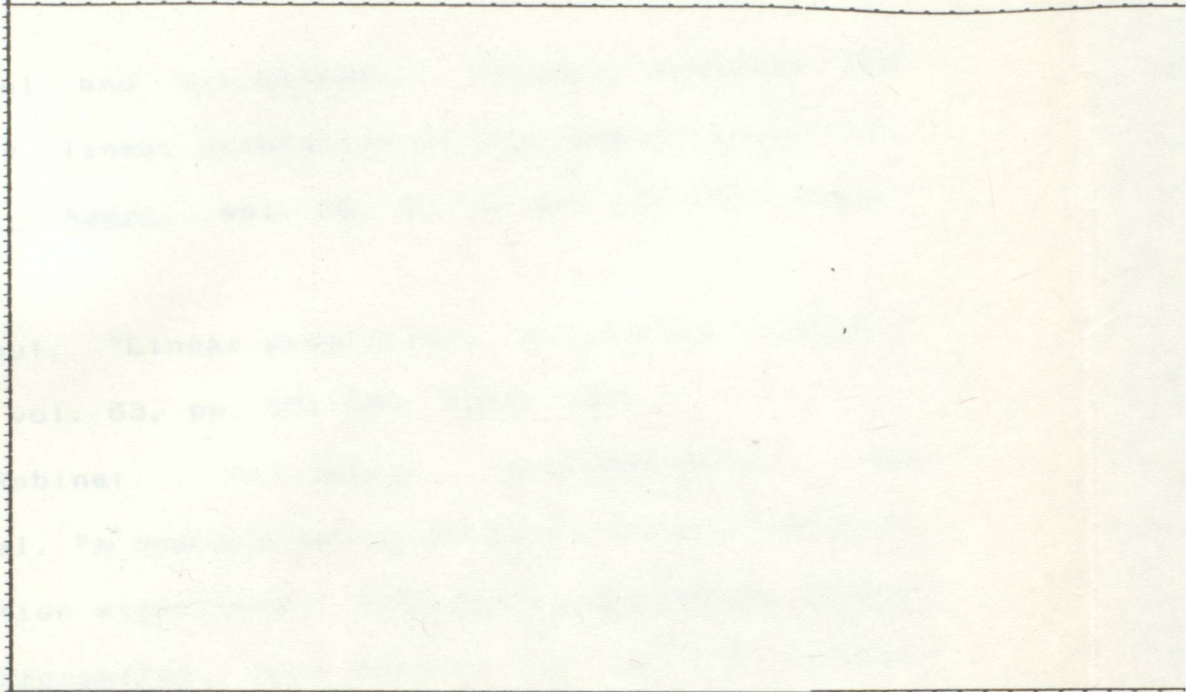
5 KHz
+f



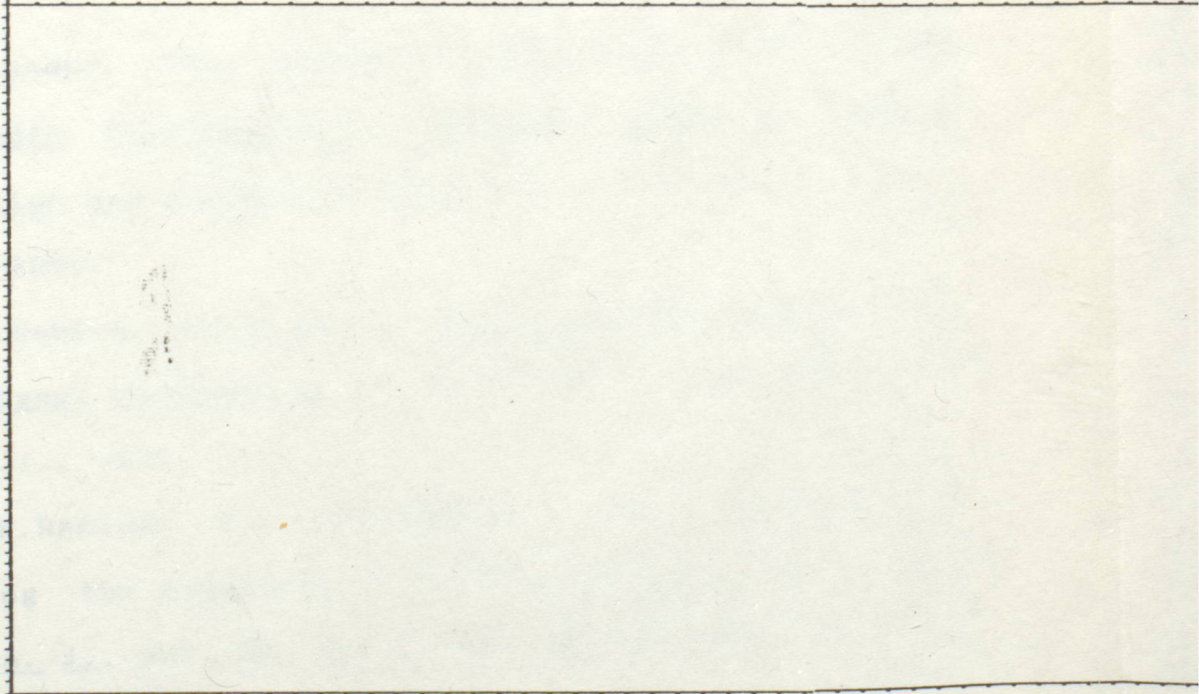
5 KHz
+f

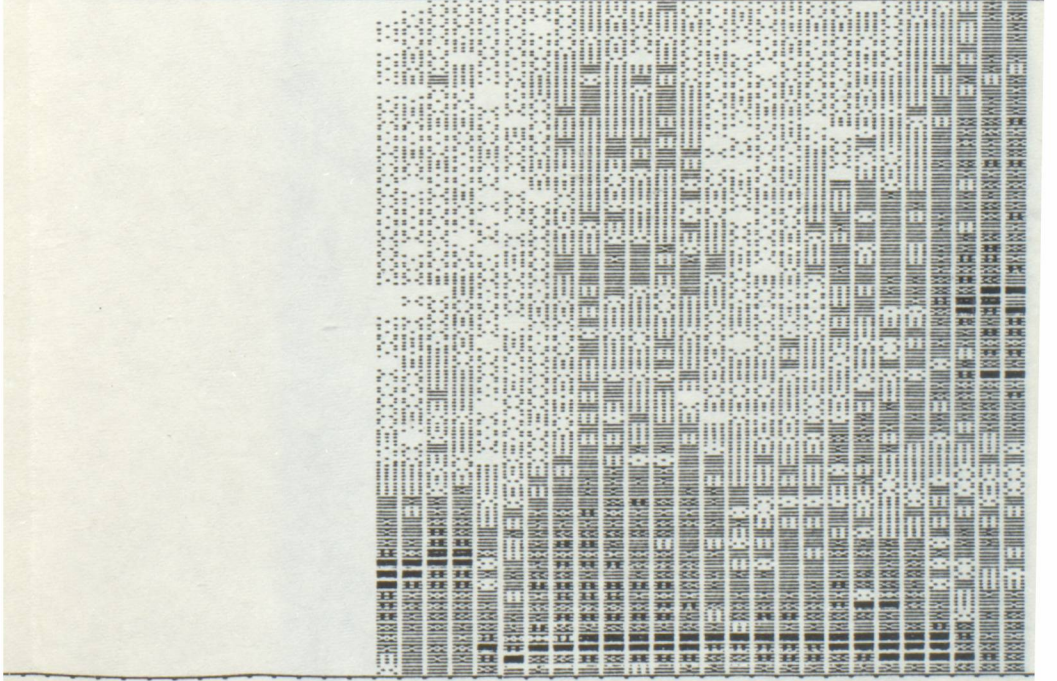
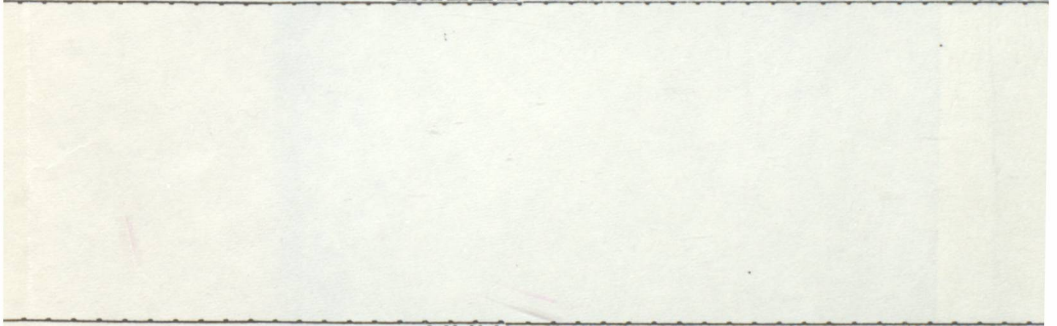
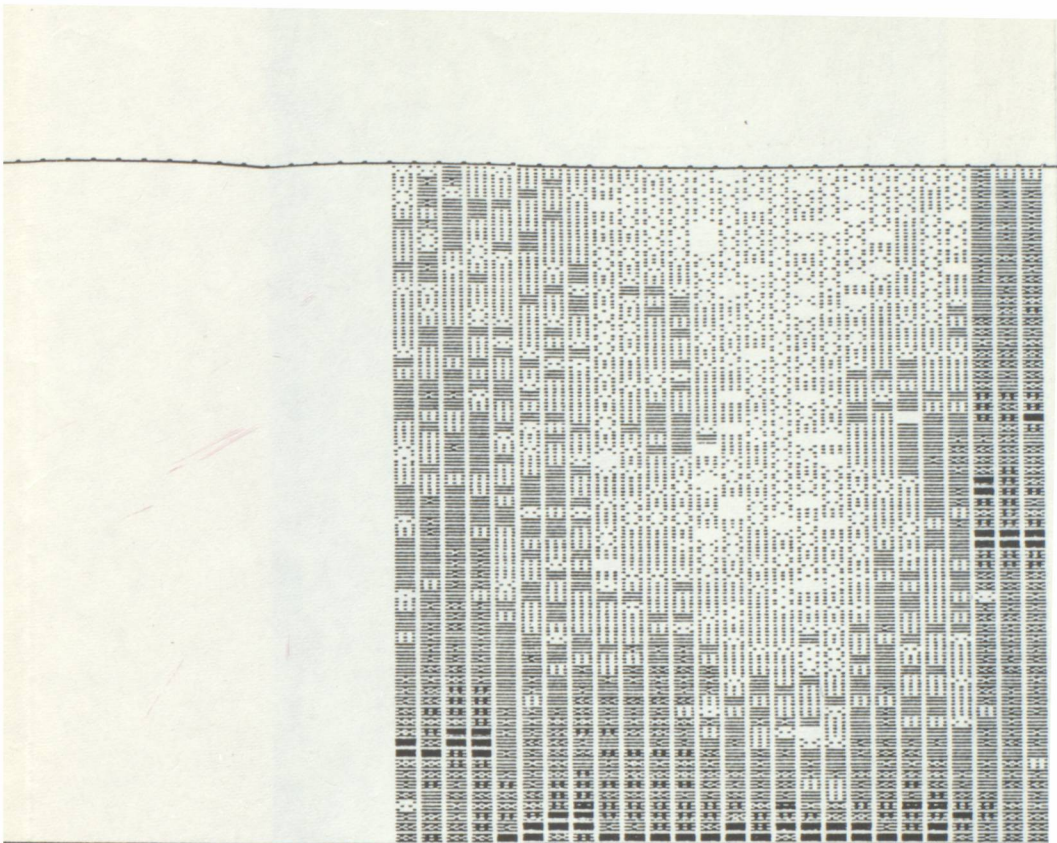


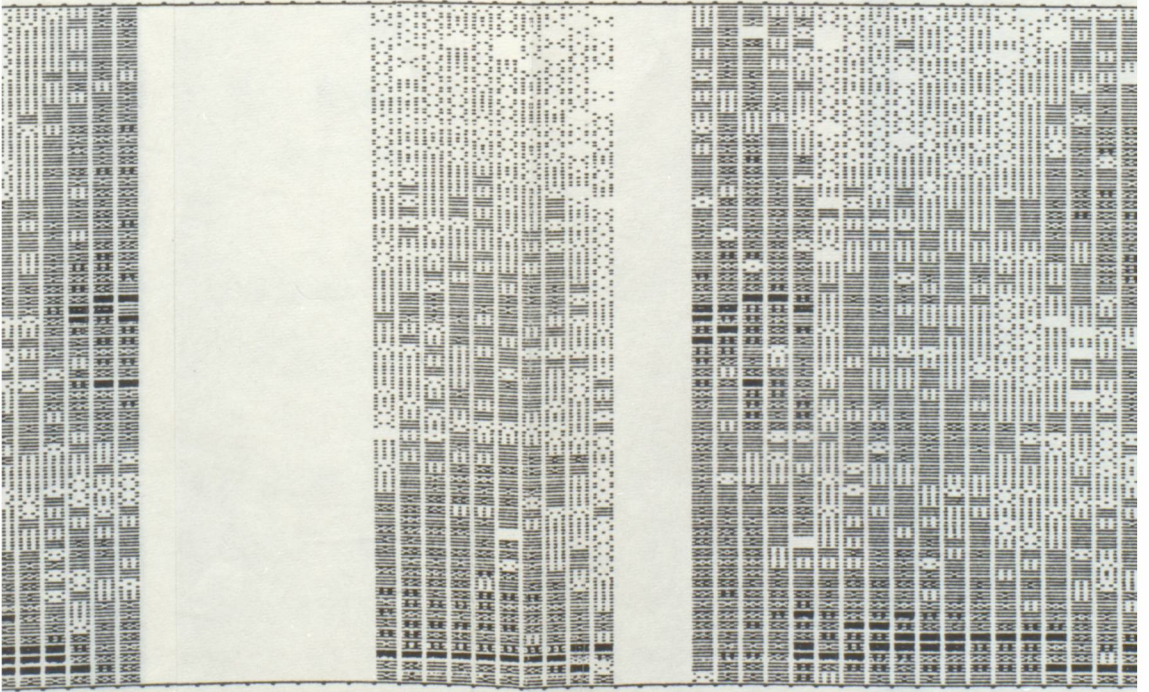
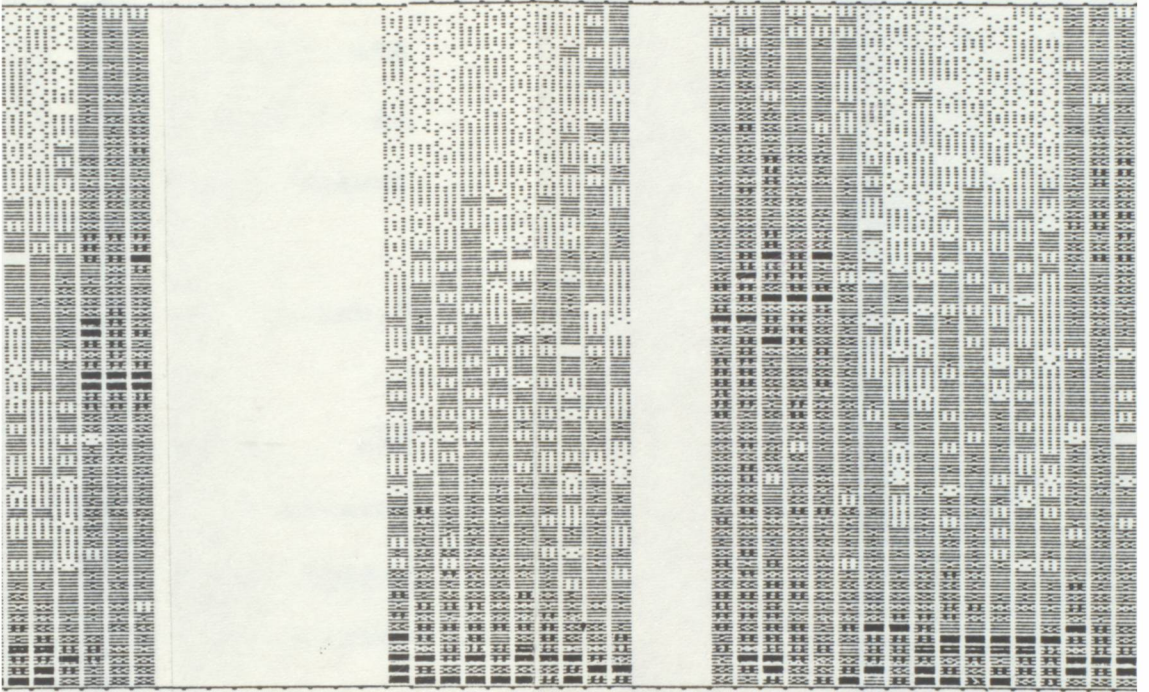
5 KHz
+f



5 KHz
+f







1. The first part of the document is a list of names and addresses, which are arranged in a table format. The names are listed in the first column, and the addresses are listed in the second column. The names are: John Doe, Jane Smith, and Bob Johnson. The addresses are: 123 Main St, 456 Elm St, and 789 Oak St.

Name	Address
John Doe	123 Main St
Jane Smith	456 Elm St
Bob Johnson	789 Oak St

Name	Address
John Doe	123 Main St
Jane Smith	456 Elm St
Bob Johnson	789 Oak St

14 DOSYA = TUM1.OFS SAYFA = 6(SON)

14 DOSYA = TUM1.SFS SAYFA = 6(SON)

KAYNAKLAR

- [1] B.S.Atal and S.L.Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave," J. Acoust. Soc. Amer., vol. 50, no. 2, pp. 637-655, August 1971.
- [2] J.Makhoul, "Linear prediction: A tutorial review," Proc. IEEE, vol. 63, pp. 561-580, April 1975.
- [3] L.R.Rabiner, M.J.Cheng, A.E.Rosenberg, and C.A.McGonegal, "A comparative performance study of several pitch detection algorithms," IEEE Trans. Acoustics, Speech and Signal Processing, vol. ASSP-24, pp. 399-418, October 1976.
- [4] J.J.Dubnowski, R.W.Schafer, and L.R.Rabiner, "Real-time digital hardware pitch detector," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-24, pp. 2-8, Feb. 1976.
- [5] M.M.Sondhi, "New methods of pitch extraction," IEEE Trans. Audio Electroacoust. (Special Issue on Speech Communication and Processing-Part II) vol. AU-16, pp. 262-266, June 1968.
- [6] L.R.Rabiner and B.Gold, Theory and Applications of Digital Signal Processing, Prentice Hall, Inc., Englewood Cliffs, N.J., 1975.
- [7] L.R.Rabiner and M.R.Sambur, "An algorithm for determining the endpoints of isolated utterances," Bell Syst. Tech. J., vol. 54, No. 2, pp. 297-315, Feb. 1975

[8] H.F.Silverman and N.R.Dixon, "A Parametrically Controlled Spectral Analysis System for Speech," IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-22, No. 5 (October 1974), pp. 362-381.

DZGEÇMİŞ

Yazar 1964 senesinin agustos ayında Aydın'da doğmuştur. İlk ve orta öğrenimini İstanbul'da tamamlamış ve 1982 senesinde Yıldız Üniversitesi Kocaeli Mühendislik Fakültesi Elektronik ve Haberleşme Bölümünü kazanarak 1986 senesinde mezun olmuştur. Aynı yıl Yıldız Üniversitesi Mühendislik Fakültesi Elektronik ve Haberleşme Bölümü Yüksek Lisans Sınavını kazanmıştır ve öğrenimine devam etmektedir. Ayrıca 1987 senesinden beri Yıldız Üniversitesi Mühendislik Fakültesi Elektronik ve Haberleşme Bölümü Devreler ve Sistemler Anabilim Dalında araştırma görevlisi olarak çalışmaktadır.



