



**YILDIZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

34732

**UYARLAMALI FARKSAL  
DARBE KOD MODÜLASYONU  
VE  
UYGULAMALARI**

**Elekt. ve Hab. Müh. Aydın AKCEN**

**F.B.E. Elektronik ve Haberleşme Mühendisliği Anabilim Dalında  
hazırlanan**

**YÜKSEK LİSANS TEZİ**

**Tez Danışmanı : Prof.Dr. Günsel DURUSOY**

**İSTANBUL, 1994**

## İÇİNDEKİLER

1.	GİRİŞ.....	1
2.	DARBE KOD MODÜLASYONU İLGİLİ TEMEL BİLGİLER .....	6
2.1	Darbe Kod Modülasyonu (PCM).....	6
2.2	Kuantalama İşlemi .....	7
2.3	Kuantalama Gürültüsü .....	8
2.4	İşaret-Kuantalama Gürültüsü Oranı .....	9
2.5	Boş Kanal Gürültüsü .....	10
2.6	Birörnek Kodlanmış PCM .....	11
2.7	Sıkıştırma Yöntemi .....	13
2.8	A-kanunuyla Sıkıştırma .....	14
2.9	Kodlama .....	15
3.	DELTA VE FARKSAL DARBE KOD MODÜLASYONU .....	18
3.1	Delta modülasyonu (DM).....	18
3.2	Farksal Darbe Kod Modülasyonu (DPCM).....	19
4.	UYARLAMALI FARKSAL DARBE KOD MODÜLASYONU .....	23
4.1	Uyarlamalı Farksal PCM (ADPCM) .....	23
4.2	Adım Boyu Çarpanlarının Dizaynı .....	26
4.3	PCM ile ADPCM'in Randıman Karşılaştırması ...	27
4.3.1	SNR Karşılaştırması .....	27
4.3.2	Gürültü Spektrogramları .....	27
5.	ADPCM'İN SAYISAL İŞARET İŞLEMCİSİYLE GERÇEKLEŞTİRİLMESİ .....	29
5.1	Uçtan Uça Konfigürasyon .....	29
5.2	DSP ile Gerçekleştirilmiş ADPCM Algoritması ....	30
5.3	Uyarlamalı Kuantalayıcı .....	30
5.4	Birinci Dereceden Tahmin Edici .....	34
5.5	Uyarlamalı Adım Boyu Kontrolü .....	34
5.6	Adım Boyu Tabloları için Parametre Seçimi ...	37
5.7	ADPCM için Kod Çözücü .....	37

6.	ADPCM İLE KODLANMIŞ KONUŞMA İŞARETİNİN VARLIĞININ BELİRLENMESİ...	39
6.1	Uç-noktaları Belirleme Algoritması .....	40
7.	ALTBAND KODLAMA .....	46
8.	64 KBİT/SN'DE 7 KHZ İŞİTSEL KODLAMA .....	50
8.1	64 Kbit/sn'de İşitsel Kodlama .....	50
8.2	İşitsel Kısımların İşlevsel Tanımı .....	52
8.3	Muhtemel Modlar ve Veri İlave Etme Cihazı ...	53
8.4	Altband-ADPCM Kodlayıcısının İşlevsel Tanımı ..	54
8.4.1	Verici Dik Ayna Filtreleri (QMF) .....	54
8.4.2	Düşük Altband-ADPCM Kodlayıcı .....	55
8.4.3	Yüksek Altband-ADPCM Kodlayıcı .....	56
8.4.4	Çoğullayıcı .....	57
8.5	Altband-ADPCM Kod çözücünün İşlevsel Tanımı .	58
8.5.1	Ters Çoğullayıcı .....	58
8.5.2	Düşük Altband-ADPCM Kod Çözücü .....	60
8.5.3	Yüksek Altband-ADPCM Kod Çözücü .....	60
8.5.4	Alıcı Dik Ayna Filtresi .....	60
9.	KONUŞMA KODLAMA İÇİN ÇOK KUANTALAYICILI ADPCM .....	62
9.1	Çok Kuantalayıcılı ADPCM (MQ-ADPCM) .....	62
	SONUÇ .....	67
	KAYNAKLAR .....	69

Bu tezin hazırlanmasında, sürekli yardımlarıyla büyük emeđi geen, Sayın Hocam Prof. Dr. Günsel Durusoy'a teŖekkür ederim.

## TÜRKÇE ÖZET

Bu tezde sayısal iletimde güçlü ve etkili bir sistem olan uyarlamalı farksal darbe kod modülasyonu (ADPCM) incelenmiştir. ADPCM, PCM ile DPCM'in geliştirilmiş hali olduğu için bu iki yöntem hakkında temel bilgiler verilmiştir.

ADPCM, DPCM ile karşılaştırıldığında benzer bir yapıda olduğu görülür. Ancak uyarlamalı DPCM'de birörnek kuantalayıcı çıkışına göre adım boyu ayarlanır. Örneğin birönceki kod sözcüğünün değeri büyük ise adım boyu artırılırken, bir önceki kod sözcüğünün değeri küçük ise adım boyu azaltılır. Böylece işareti kuantalamak için daha az bite ihtiyaç duyulur ve bit hızı 32 kbit/s'ye düşer. 32 kbit/s'lik bit hızı kanal band genişliğinde PCM'e göre iki misli tasarruf anlamına gelir.

Uç noktaları belirleme algoritması, 16 kbit/s alt-band kodlayıcı, 64 kbit/s 7 kHz işitsel kodlayıcı ve çok kuantalayıcılı ADPCM (MQ-ADPCM) ADPCM'in uygulamaları olarak ayrı ayrı bölümlerde incelenmiştir.

## SUMMARY

In this thesis, adaptation differential pulse code modulation which provides a robust and efficient technique in the digital transmission is investigated. In addition, basic information is given about PCM and DPCM, because ADPCM improved case of these techniques.

The basic ADPCM system is a differential PCM system with adaptive quantizer which adapts the step size on the basis of the previous quantizer output. For example if the magnitude of the previous code word corresponds to one of the extreme levels, the step size is increased, whereas if the magnitude of the previous code word corresponds to one of the lowest levels, the step size is decreased. So less bit is need to quantize the input signal and the bit rate are decrease to 32 kbit/s.

16 kbit/s sub-band coder, 64 kbit/s (7 kHz) audio coder and ADPCM with a Multiquantizer (MQ-ADPCM) is separately investigated as application of ADPCM.

## 1. GİRİŞ

Tüm dünyada telekomünikasyon alanında yeni teknolojilerin hızla gelişmesi ve kullanıcılar tarafından yüksek hızlı veri, video, daha çok kanallı sistemler gibi yeni hizmetlere duyulan gereksinim, telekomünikasyonla uğraşan bilim adamlarını bu yeni hizmetlere duyulan gereksinimi karşılamak amacıyla yeni sistemler bulmaya itmiştir.

Tümleşik devre teknolojisindeki son gelişmelerin sayısal devre tasarımına katkısı, sayısal işaret işleme tekniklerindeki ilerlemelerin sayısal işaretlerin daha verimli bir biçimde kullanımını sağlaması ve haberleşme ile bilgisayar teknolojilerinin gitgide birbirinin bağımlı duruma gelmesi, sayısal haberleşmeye yönelişi kaçınılmaz kılmaktadır.

Analog ve sayısal sistemler teorik olarak birbirinin aynıdır. Fakat gerçek uygulamada, sayısal sistemlerin lehine büyük farklar ortaya çıkar. Bu farkı ortaya çıkaran etken, haberleşme tekniğinin en büyük düşmanı olan gürültü ve zayıflamadır.

Sayısal iletişim teknikleri, iletim olanaklarının kapasitesini artırmada son derece ekonomiktir. Özellikle jonksiyon kabloları, darbe kod modülasyonu (PCM) kullanıldığında analog iletme kıyasla çok daha fazla eşzamanlı ses işareti taşıyabilmektedir. Bu da yeni kablo döşenmesi sorununu ortadan kaldırır.

Sayısal işaretlerin yeniden üretilmelerinin kolay olması, uzun ve gürültülü haberleşme kanallarından yapılan iletimde analog işaretlere göre üstünlük sağlamaktadır. Ayrıca sayısal işaretlerin çoğullanması daha basit ve daha ucuz olarak gerçekleştirilebilmektedir.

Sayısallaştırılmış ve kodlanmış işaretlerin iletimini kapsayan sistemler darbe kod modülasyonu (PCM) diye adlandırılır. İkili sayısal sistemler PCM'in en sık rastlanan biçimini oluşturur. Darbe kod modülasyonunun bazı üstünlükleri şunlardır:

- İşaretler iletim süresince düzenli olarak yeniden üretilebilirler.
- Sistem baştan başa sayısal olarak gerçekleştirilebilir.
- İşaretler sayısal olarak işlenebilir.
- Gürültü minimize edilebilir.

PCM'de karşılaşılan önemli bir sorun kuantalama gürültüsüdür. Kuantalama gürültüsü genellikle işaretin kuantalamasından ve fazla yükleme gürültüsünden kaynaklanır. Kuantalama gürültüsü tümüyle kaldırılamaz, ancak çeşitli yollarla azaltılabilir. Kuantalama gürültüsünü azaltmak için kuantalama düzeylerinin sayısı artırılmalıdır, bu ise PCM işaretinin band genişliğinin artmasına neden olur. Öte yandan band genişliği kısıtlaması nedeniyle kuantalama düzeylerinin sayısı rastgele artırılamaz. Çoklanan işaret sayısının azaltarak, band genişliğini artırmadan ikili dizilerin uzunluklarının büyütülmesi olasıdır. Ancak çoklanan işaret sayısının azaltılması da istenmeyen bir durumdur.

PCM'in geliştirilmiş hali farksal darbe kod modülasyonudur (DPCM). Farksal kodlama güncel örnek değeriyle önceki giriş değerleri arasındaki farkı kodlama ile gerçekleştirilir. Kodlayıcı girişinde fark işaretini oluşturmak için güncel işaret tahmin değerlerinden çıkarılır. Tahmin etme işlemi geriye uyarlama işlemidir ve kod çözücüyle kodlayıcı arasındaki senkronizasyon için ek bir bilgi gerektirmez. Açıkça belli ki tahmin edicinin yapacağı en doğru tahmin güncel işaretin kendisidir. Farksal işlem kuantalayıcıya uygulanacak



işaret ölçüsünü indirger ve böylece kuantalayıcıda daha az bit kullanımına olanak tanır. Dolayısıyla standart PCM'e göre bit hızı düşer.

DPCM'in geliştirilmiş hali ise uyarlamalı farksal darbe kod modülasyonudur (ADPCM). Uyarlamalı farksal darbe kod modülasyonu Cummskey et al (1973) tarafından tasarlanmıştır. ADPCM'in kuantalayıcısının standart PCM kuantalayıcısından iki önemli farkı vardır. Birincisi her örnekleme anında, bütün kuantalama seviyeleri eşit aralıklarla birörnek ölçeklenir, logaritmik olarak ölçeklenmez. İkincisi giriş işareti, kuantalama aralığının limitlerine geldikçe adım boyu artırılır, eğer aralığın merkezine doğru kuantalanırsa adım boyu azaltılır. Bir konuşmacının konuşma işaretleri, bir sözcükten diğerine önemli bir değişiklik göstermediği için, kuantalayıcı güncel işaretin aralığına çabuk uyum sağlar ve daha sonra konuşmacı susana kadar mükemmel biçimde ölçeklemeye devam eder.

Sayısal iletim sistemlerinde ortalama bit hızını indirmek için konuşma ile sessizlik arasındaki farkı otomatik olarak ayırt etmek gereklidir. Çünkü konuşmadaki düşük seviyeli işaretler ile sessizlik sırasındaki boş kanal gürültüsü aynı kuantalama düzeyleriyle kuantalanabilir ve kod çözücüde hatalı çıkış elde edilebilir. Bunu önlemenin yolu konuşma ifadesinin başını ve sonunu belirleyip, sessizlik sırasında kuantalamayı kesmektir. Rosenthal et al (1974) bunu sağlayan algoritmayı tasarlamışlardır.

16 kbit/s alt-band kodlayıcı, R.E. Crochiere (1977) tarafından 16 kbit/s'li bit hızında iyi nitelikli sayısal konuşma elde edilebilir bir teknik olarak sunulmuştur. Bu nitelik 24 kbit/s'li ADPCM ile sübjektif olarak karşılaştırılabilir.

Her bandı ayrı ayrı kodlama birçok nedenden dolayı avantajlıdır: Birincisi her band için ayrı ayarlama kullanarak kuantalayıcıların adım boyları her bandaki enerji seviyelerine göre ayarlanır. İkinci avantajı ise her bantta tespit edilen bit hızının, her bandın algısal kriterine göre optimize edilebilmesidir.

CCITT G.711 tavsiyesinde, analog/sayısal şebekeleri birbirine bağlamak için PCM kodlama kanunları (A- veya M-kanunları) 1970'lerin başında tanımlanmıştır. Sonuç olarak, PCM kodlama kanunları sayısal şebekelere uygulandığı zaman kuantalama gürültüsü azalır. Diğer yandan iletilen band genişliği 300-3400 Hz'de (bütün analog şebekelerde olduğu gibi) sınırlandırılır. Bu da konuşma niteliğini azaltır. CCITT'nin uzman grubu, bu durumu düzeltmek için 64 kbit/s 7 kHz işitsel kodlama algoritmasını tasarlamışlardır (G.722).

Fark işaretini kuantalamak için 2 bit/örnek veya daha düşük bir değer kullanıldığı zaman ADPCM'in randımanı düşer. Bu düşüşün üstesinden gelmek için çok kuantalayıcılı ADPCM (ADPCM-MQ) Taniguchi et al (1988) tarafından sunulmuştur. ADPCM-MQ kodlama algoritmasında en iyi konuşma niteliğine sahip optimum ADPCM kodlayıcı her bir çerçeve için dinamik olarak seçilir.

Bu tezde ayarlamalı farksal darbe kod modülasyonu ve uygulamaları incelenmiştir. ADPCM sistemiyle birlikte, 64 kbit/s PCM işareti 32 kbit/s'ye indirgenir. Ayrıca ADPCM PCM'e göre 8 dB'lik bir iyileşme sağlar. Bu iyileşmenin 4 dB'i farksal kodlayıcıdan ve geri kalan 4 dB ise ayarlamalı kuantalayıcıdan kaynaklanır.

Tezin ikinci bölümünde darbe kod modülasyonu ile ilgili temel bilgiler verilmiş; sıkıştırma yönteminin bir örnek PCM'e getirmiş olduğu avantajlar belirtilmiştir.

Tezin üçüncü bölümünde delta modülasyonu ve farksal darbe kod modülasyonu incelenmiş; DPCM'in delta modülasyonuna göre üstünlükleri belirtilmiştir.

Tezin dördüncü bölümünde uyarlamalı farksal darbe kod modülasyonu incelenmiştir. PCM ile ADPCM'in randımanlarının karşılaştırılması yapılmıştır.

Tezin beşinci bölümünde sayısal işaret işlemcisiyle gerçekleştirilmiş ADPCM algoritması incelenmiştir.

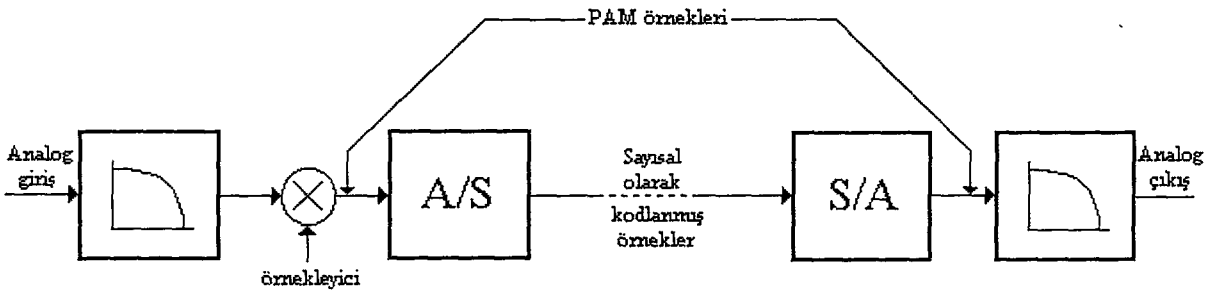
Tezin altıncı bölümünde uç-noktaları belirleme algoritması, yedinci bölümünde 16 kbit/s alt-band kodlayıcı, sekizinci bölümünde 64 kbit/sn'de 7 kHz işitsel kodlama ve dokuzuncu bölümde çok kuantalayıcılı ADPCM algoritması üzerinde durulmuştur.

## 2. DARBE KOD MODÜLASYONU İLE İLGİLİ TEMEL BİLGİLER

### 2.1 Darbe Kod Modülasyonu (PCM)

Darbe kod modülasyonu darbe genlik modülasyonunun (PAM) genişletilmiş halidir. Şekil 2.1'de tipik bir PCM sistemi gösterilmektedir. PCM'de bilgi işareti  $m(t)$ , alıcıda yeniden elde edilebileceği bir hızda örneklenir. Daha sonra A/S dönüştürücüde bu örnek değerler kuantalanır, yani her örnek değere önceden belirlenmiş düzeylerden bu değere en yakın olan ile yaklaştırma yapılır. Her düzey bir kod sözcüğü ile kodlanır. Böylece A/S dönüştürücüsü çıkışında PCM işareti elde edilmiş olur.

PCM işareti ya doğrudan kablolar üzerinden ya da örneksel modülasyon yöntemleri yardımıyla iletilir. Alıcıda kod sözcüğü elde edildikten sonra, kodu çözülür ve kuantalanmış örnek değerler bulunur. Ancak kuantalama tersinir bir işlem olmadığından, gerçek örnek değerlerinin yeniden elde edilmesi olanaksızdır. Gerçek örnek değerleri elde edilse bile, bilgi işareti  $m(t)$  tam olarak elde edilemez. Çünkü kuantalamadan kaynaklanan bir bozulma vardır. Bu bozulma kuantalama gürültüsü diye adlandırılır.



Şekil 2.1

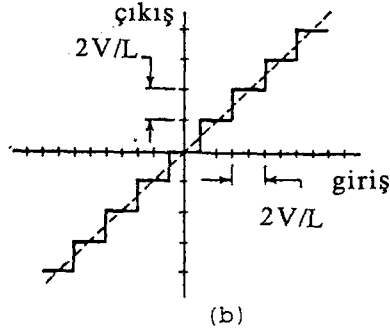
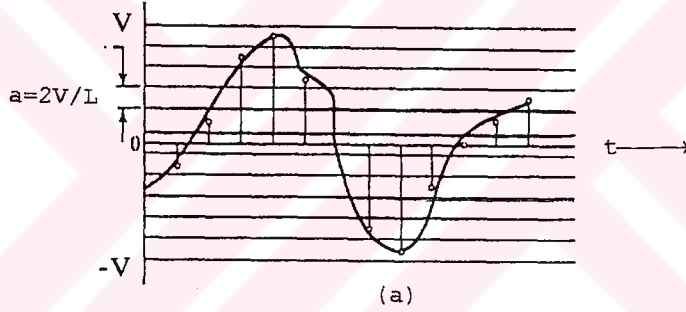
Darbe kod modülasyonu

## 2.2 Kuantalama İşlemi

Şekil 2.2a'da kuantalama işlemi gösterilmiştir.  $m(t)$  işaretinin  $[-V, V]^*$  bölgesinde değerler aldığı ve  $L$  tane kuantalama düzeyinin kullanıldığını varsayalım. Kuantalama,  $[-V, V]$  bölgesinin  $L$  düzeye bölünmesiyle gerçekleştirilir. Kuantalama düzeyleri arasındaki uzaklığa adım boyu denir ve aşağıdaki gibi tanımlanır:

$$a=2V/L \quad (2.1)$$

Kuantalama aralığına isabet eden tüm örnek değerleri, kuantalama aralığının merkezindeki kuantalama düzeyine karşı düşer. Kuantalayıcının giriş-çıkış karakteristiği Şekil 2.2b'de gösterilmiştir.



Şekil 2.2

a) Kuantalama işlemi b) Kuantalama karakteristiği

(\*):  $V$ ,  $m(t)$  işaretinin maksimum genliğidir.

Yukarıdaki bilgilerin ışığında, kuantalama yanılığının adım boyunun yarısı olan  $V/L$ 'yi geçemeyeceğini söyleyebiliriz. Kuantalama yanılığını azaltmak için kuantalama düzeylerinin sayısı artırılmalıdır, bu ise PCM işaretinin bant genişliğinin artmasına neden olur. Bu nedenle kuantalama düzeylerinin sayısı rastgele artırılmaz.

### 2.3 Kuantalama Gürültüsü

Kanalda PCM işaretine eklenen gürültü kod simgelerinin tanımlanmasında yanılığlara neden olur. PCM işaretinin sezimi, bir darbenin varlığını ya da yokluğunu saptamaktan ibaret olduğundan, işaret gücünün artırılması toplamsal gürültünün neden olduğu yanılığ miktarının azalmasına neden olur. Ancak işaret gücü rastgele artırılamayacağından, etkisi azaltılmasına karşın kanaldaki toplamsal gürültü yine de bir yanılığ kaynağı olur.

PCM'de bundan başka bir yanılığ kaynağı daha vardır. Bu PCM'de kullanılan sistemin yapısı gereği oluşan kuantalama yanılığsıdır. Kuantalanmış değerler kullanılarak üretilen sürekli işaret, özgün bilgi işaretinden farklıdır. Kuantalamanın neden olduğu bu fark, bilgi işaretine eklenmiş bir gürültü gibi düşünülebilir. Bu nedenle bu yanılığ kuantalama gürültüsü diye adlandırılır.  $m(t)$  işaretinin genliği dinamik erimi boyunca düzgün dağılım gösteriyorsa, kuantalama gürültüsü  $n_q(t)$ 'nin olasılık yoğunluğu,  $a$  aralığı boyunca düzgün dağılır.  $n_q(t)$ 'nin olasılık yoğunluğu aşağıdaki biçimdedir:

$$f(n_q) = \begin{cases} 1/a & -a/2 \leq n_q \leq a/2 \\ 0 & \text{dışında} \end{cases} \quad (2.2)$$

$n_q(t)$ 'nin beklenen değeri sıfırdır:

$$E[n_q(t)] = \int_{-a/2}^{a/2} \left(\frac{1}{a}\right) n_q \, dn_q = 0 \quad (2.3)$$

$n_q(t)$  durağan varsayılmak üzere, ortalama gücü ya da karesel beklenen değeri:

$$N_q = E[n_q^2(t)] = \int_{-a/2}^{a/2} \left(\frac{1}{a}\right) n_q^2 \, dn_q = \frac{a^2}{12} \quad (2.4)$$

(2.4)'de verilen değer örnekleme anındaki kuantalama gürültüsünün ortalama gücüdür. Bu ifade de  $N_q$ 'nin adım boyunun karesi ile orantılı olduğu görülmektedir.  $a=0$  için (sonsuz kuantum düzeyi) kuantalama hatası sıfırdır. Ancak kuantalama düzeylerinin sayısını artırmak için ya bant genişliğini artırmak ya da kod alfabesinin büyüklüğünü artırmak gerekir. Bant genişliğini artırmak için bilinen kısıtlamalar vardır. Eğer kod alfabesinin büyüklüğü artırılırsa PCM işaretinin toplamsal kanal gürültüsüne karşı olan bağışıklığı azalır. Bu nedenlerle kuantalama gürültüsü istenildiği gibi azaltılamaz.

#### 2.4 İşaret/Kuantalama Gürültüsü Oranı (SNR)

İşaret gücünün kuantalama gürültüsünün ortalama gücüne oranı SNR (signal-to quantizing noise ratio) ile gösterilir.  $m(t)$  işaretinin kuantalanmış gösterimi olan  $m_q(t)$  sırayla  $+a/2, +3a/2, \dots, +(L-1)(a/2)$  değerlerini alır. Bu  $m_i$  değerlerini alabilen  $m_q(t)$  işaretinin karesel ortalama değeri aşağıdaki ifadeden hesaplanır:

$$S_q = [m_q^2(t)] = \sum_i m_i^2 p[m_q(kT_s) = m_i] \quad (2.5)$$

Düzgün dağılmış ve eşit olasılıklı  $L$  düzeye sahip  $m(t)$  işaretinde, her  $i$  için  $m_i$ 'nin olasılığı:

$$p[m_Q(kT_s) = m_i] = \frac{1}{L} \quad (2.6)$$

olarak hesaplanır. O halde işaretin ortalama gücü:

$$S_Q = \left(\frac{1}{L}\right) \sum_{-(L-1)/2}^{(L-1)/2} (ia)^2 = (a^2 / 12)(L - 1)(L + 1) \quad (2.7)$$

olarak bulunur. Bu ifadeyi (2.4)'de hesaplanan kuantalama gürültüsüne oranlı olarak  $SNR$ 'yi elde ederiz:

$$SNR = \frac{S_Q}{N_Q} = \frac{(a^2 / 12)(L - 1)(L + 1)}{(a^2 / 12)} = (L - 1)(L + 1) = (L^2 - 1) \quad (2.8)$$

Yukarıdaki ifadenin dB birimli gösterimi aşağıdaki gibidir:

$$SNR(dB) = 10.8 + 20 \log_{10} L = 10.8 + 20 \log_{10} \left(\frac{V}{a}\right) \quad (2.9)$$

Elde edilen sonuçlar birörnek kuantalama için geçerlidir.

## 2.5 Boş Kanal Gürültüsü

(2.9) eşitliğini incelersek  $SNR$ 'nin küçük örnekler için düşük bir değer aldığını görürüz. Gerçekte, Şekil 2.3'de gösterildiği gibi, örnek değerleri ilk kuantalama aralığındayken gürültü işaretten daha büyük olabilir. Özellikle konuşma duraklamalarında görülen bu gürültüye boş kanal gürültüsü (idle channel noise) denir. Şekil 2.4'de boş kanal gürültüsünü indirgeme metodlarından biri gösterilmiştir. Burada merkezdeki kuantalama aralığına düşen bütün örnek değerleri

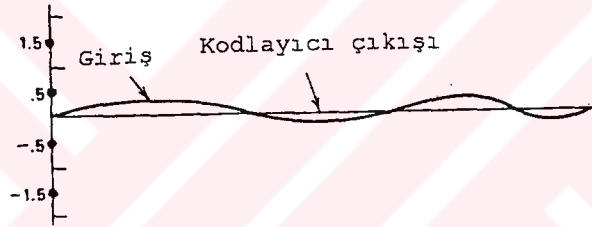


için kod çözücü çıkışı sabit sıfır değerini alır. Böylece boş kanal gürültüsü önlenmiş olur.



Şekil 2.3

Sıfır çıkış seviyesiz kuantalayıcı tarafından üretilen boş kanal gürültüsü



Şekil 2.4

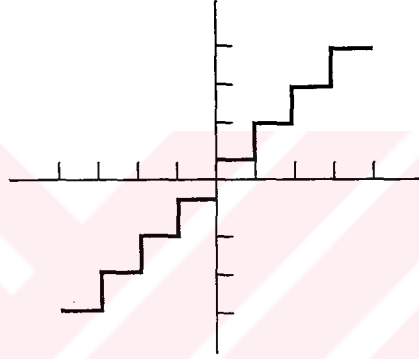
Sıfır çıkış seviyeli kuantalayıcı tarafından boş kanal gürültüsünün yok edilmesi

Şekil 2.3 ve 2.4'de gösterilen dalga şekillerini üretmek için gerekli kuantalama karakteristikleri sırayla Şekil 2.5 ve 2.6'da gösterilmiştir. Birinci karakteristikte sıfır çıkış seviyesi yoktur. İkinci karakteristik düşük seviyeli işaretleri sıfır seviyeli sabit çıkışlar biçiminde çözer.

## 2.6 Birörnek Kodlanmış PCM

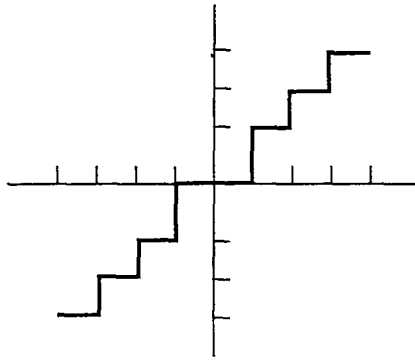
Bütün örnekler için eşit uzunluklu kuantalama aralıkları kullanan kodlayıcı analog örnek değerleriyle doğrusal orantılı kod sözcükleri üretir. Yani her bir kuantalanmış örnek değeri,

kendisine karşı düşen kod sözcüğünün sayısal eşdeğeriyle orantılıdır. Birörnek PCM sisteminde kodlama için, her bir kuantalama aralığı en düşük işaret seviyesi için gerekli olan SNR'ye göre belirlenir. Yüksek genlikli işaretlerde dahi aynı kuantalama aralıkları kullanılır. Böylece birörnek PCM sistemi yüksek genlikler için gereksiz bir kalite sağlar. Üstelik yüksek genlikli işaretlerin meydana gelme olasılığı da çok düşüktür. Bu nedenle bu sistemde kod uzayı çok etkisiz kullanılmış olur.



**Şekil 2.5**

Sıfır çıkış seviyesiz (mid-riser) kuantalayıcı karakteristiği



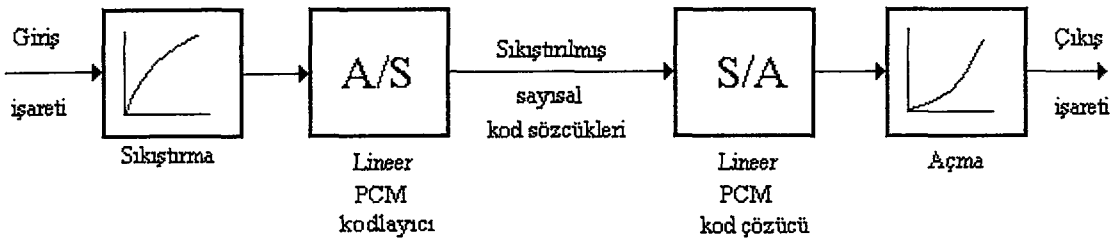
**Şekil 2.6**

Sıfır çıkış seviyeli (mid-tread) kuantalayıcı karakteristiği

## 2.7 Sıkıştırma Yöntemi (Companding)

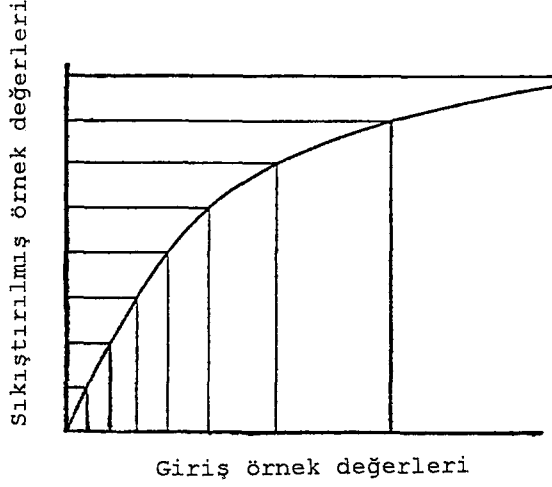
Bir örnek kodlamaya göre daha etkili bir kodlama prosedürü olan sıkıştırma yöntemimde kuantalama aralıkları bir örnek olmadan artırılır. Yani kod sözcükleriyle örnekler arasında lineer olmayan bir ilişki vardır. Kuantalama aralıkları doğrudan doğruya örnek değerleriyle orantılı olduğu zaman, işaret-gürültü oranı bütün işaret seviyeleri için sabit olur. Bu teknikle birlikte örnek başına daha az bit düşer. Böylece küçük genlikli işaretlere daha çok kuantalama aralığı tahsis edilir ve yüksek genlikli işaretler için gereken dinamik aralık sağlanır.

Temel işlem Şekil 2.7'de gösterilmiştir. Burada ilk önce giriş örneği sıkıştırılır ve daha sonra bir örnek kuantalama aralıklarıyla kuantalanır. Sıkıştırma işleminin etkisi Şekil 2.8'de gösterilmiştir. Ard arda gelen geniş giriş işaret aralıklarınının, sabit uzunluklu kuantalama aralıkları içine sıkıştırıldığına dikkat ediniz. Böylece yüksek genlikli örnek değerleri kodlamadan önce daha çok sıkıştırılır. Şekil 2.7'de görüldüğü gibi, kod çözücünde sıkıştırılmış değerler, sıkıştırma karakteristiğinin tersi olan açma karakteristiğiyle açılır. İşareti önce sıkıştırma ve daha sonra açma işlemine "sıkıştırma" (companding) denir.



Şekil 2.7

Sıkıştırılmış PCM



Şekil 2.8

Tipik sıkıştırma karakteristiği

Birçok sıkıştırma yöntemi içinden iki tanesi CCITT tarafından standartlaştırılmıştır. Bunlar Kuzey Amerika ve Japonya'da kullanılan  $\mu$ -kanunu ve Avrupa'da ve dünyanın geriye kalan birçok ülkesinde kullanılan A-kanunu karakteristiktir.

## 2.8 A-kanunuyla Sıkıştırma

A-kanunu karakteristiği aşağıdaki gibi tanımlanmıştır:

$$y = \frac{Ax}{(1 + \log A)} \quad 0 \leq v \leq \frac{V}{A} \quad (2.10)$$

$$y = \frac{1 + \log(Ax)}{(1 + \log A)} \quad \frac{V}{A} \leq v \leq V \quad (2.11)$$

Bu ifade de:

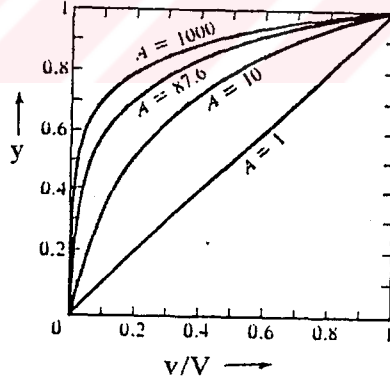
$$x = v / V$$

$$Y = b_i / B$$

Burada  $v$  giriş voltajı,  $V$  doruk sınırlamasının olmaması durumundaki maksimum giriş voltajı,  $b_i$  kuantalama aralığınının merkezinden  $i$ . satıra kadar olan adım boyu sayısı ve  $B$  aralığın her iki tarafında bulunan adım boyu sayısıdır. (CCITT Rec. G.711).

Sıkıştırma parametresi  $A$  ile, işaret gürültü oranı kısmen sabit olan bir dinamik aralık belirlenir. CCITT tarafından standartlaştırılmış olan  $A=87.6$  değeriyle 40 dB'lik bir dinamik aralık sağlanır. Bu değer için ve diğer bir çok  $A$  değeri ( $A=1$  değeri birörnek kuantalamaya karşı düşer) için  $A$ -kanunu karakteristikleri Şekil 2.9'da gösterilmiştir.

Pratik PCM sistemlerinde sıkıştırma devresi, logaritmik eğrinin istenen kopyasını sağlamıyor. Devre eğri parçaları kullanarak yaklaşık bir eşitlik sağlıyor. Bu parçaların her biri lineerdir. Eğri parçası ne kadar çok olursa o kadar çok istenen eğriye yaklaşılr.



Şekil 2.9

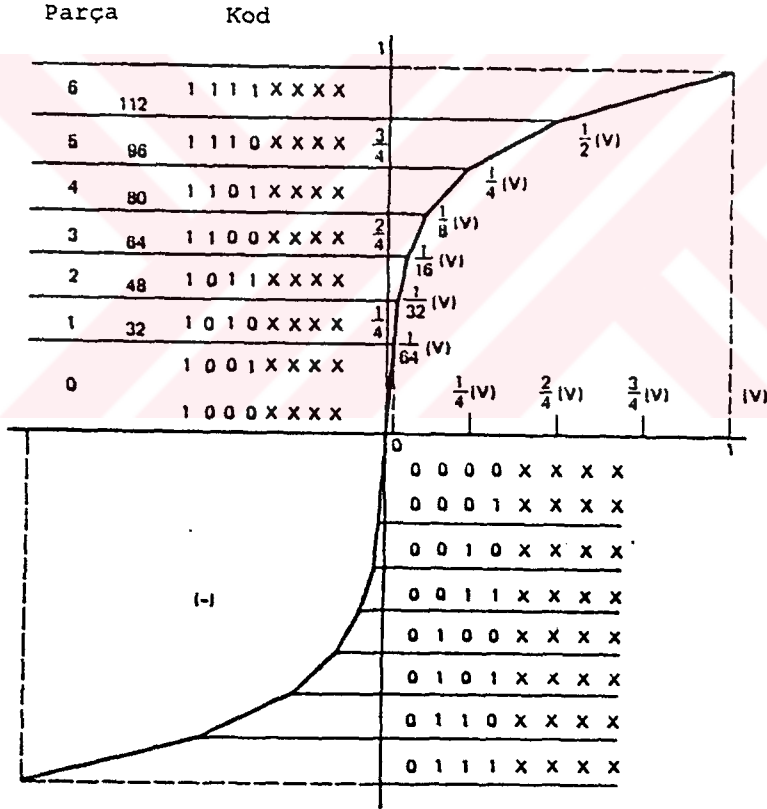
A-kanunu karakteristikleri

## 2.9 Kodlama

Kodlama işlemi kuantalamayla yakından ilgilidir. Yukarıda söz edildiği gibi, pratik sistemlerdeki kuantalama işleminde

sıkıştırma eğrisinin parçalardan oluşan eşdeğeri (Şekil 2.10) kullanılır. Parçalama işlemi kodlamaya yardımcı olur.

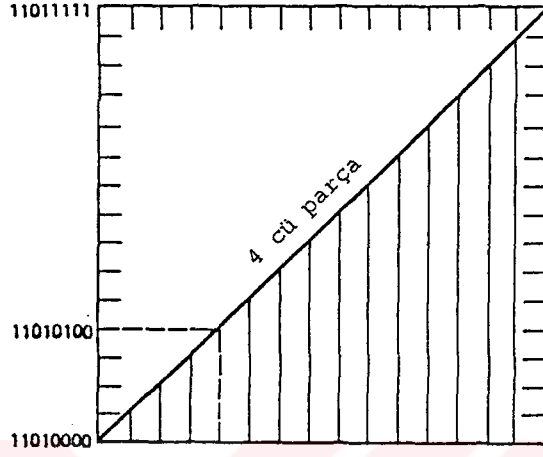
Avrupa'da kullanılan ve 13 parçalı eğriye sahip, A-kanunuyla sıkıştırılmış PCM sistemini inceleyelim (Şekil 2.10). Kodun birinci biti, kuantum adımının orijinin yukarısında veya aşağısında olup olmadığını gösterir. Kodun takip eden 3 biti ise parçayı belirler. Örneğin dördüncü pozitif parçanın ilk dört biti 1101 dir. Birinci "1" parçanın yatay eksenin üstünde olduğunu gösterirken, diğer üç bit ise dördüncü parçayı belirler.



Şekil 2.10

13 parçalı eğriye sahip A-kanunuyla sıkıştırılmış PCM sisteminde kuantalama ve kodlama

Şekil 2.11'de bir örnek kuantalamanın ayrıntılı gösterimi ve dördüncü parçanın açık binary kodlaması gösterilmiştir. Böylece kodlama işlemi tamamlanmış olur.



Şekil 2.11

Şekil 2.10'daki 4. (pozitif) parçanın kodlanması

### 3. DELTA VE FARKSAL DARBE KOD MODÜLASYONU

Analogdan sayısala dönüşümün alternatif yöntemleri gerekli band genişliğini indirgemek, randımanı iyileştirmek veya maliyeti düşürmek için önerilmiştir. Delta modülasyonu ve farksal darbe kod modülasyonu (differential pulse code modulation [DPCM]) iki belirgin örnektir. Özellikle CCITT uluslararası standart olarak 64 kbit/s PCM'in yerine tasarlanan 32 kbit/s uyarlamalı DPCM'i (bkz. Bölüm 4) onaylamaktadır. İki DPCM işareti aynı band genişliğinde iki misli kapasiteyle bir PCM işaretinin yerine geçmektedir. DPCM tekniğinin diğer bir uygulaması da 64 kbit/s işaretleme hızında geniş bantlı (7 kHz) konuşma işaretlerinin iletimi için olan CCITT standardıdır (bkz. Bölüm 8).

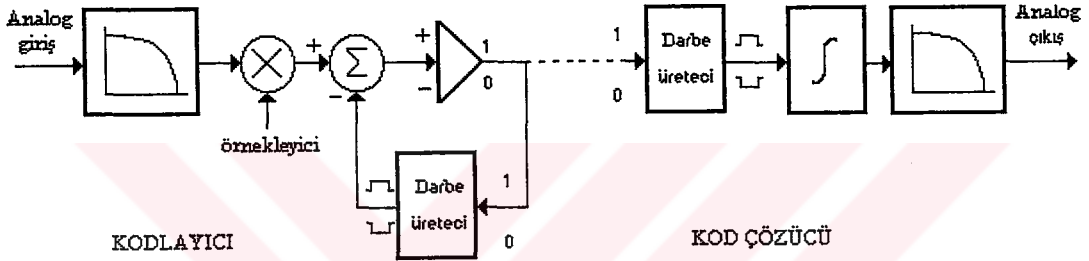
Bu iki uygulama ve bunlara benzer olanlar konuşma iletimini kapsarlar. Delta modülasyonu ve DPCM ile resim iletimi de iyi bir şekilde gerçekleştirilmiştir. Bu uygulamalarda çok iyi tahmin yapılması gerekmektedir. Çünkü iletilen bilgide artış söz konusudur.

#### 3.1 Delta Modülasyonu

Delta modülasyonunun (DM) blok diyagramı Şekil 3.1'de gösterilmiştir.  $m(t)$  analog işareti periyodik olarak örneklenir. Örneklenmiş giriş işaretinden tahmin işareti çıkarılır ve elde edilen fark işareti karşılaştırıcıya uygulanır. Burada karşılaştırıcı iki seviyeli kuantalayıcı gibi çalışır ve çıkışı fark işaretine bağlı olarak 0 veya 1 olur. Böylece fark işaretini göstermek için sadece bir bit kullanılır. Tahmin işareti geri beslemedeki darbe üreticisinde elde edilir. Darbe üreticisi, girişine "1" gelirse pozitif darbe, "0" gelirse negatif darbe üretir. Alıcıdaki darbe üreticisi de aynı mantıkla çalışır ve çıkışındaki tümleyici analog çıkış işaretini oluşturur.



Delta modölatörü ne kadar iyi çalıřır? Bunu genel olarak belirlemek oldukça zordur. Çünkü randıman, giriř iřaretinin türüne, örnekleme hızına, kuantalayıcı seviyelerine ve kullanılan tahmin edicinin biçimine baęlıdır. Cihaz teorik olarak basit olmasına karřın kendi içindeki geribeslemeden dolayı analiz etmek zordur. Son yıllarda delta modölatörüyle ilgili birçok çalıřma yapılmıřtır. Bunlardan bir tanesi uyarlamalı delta modölasyonudur.



řekil 3.1

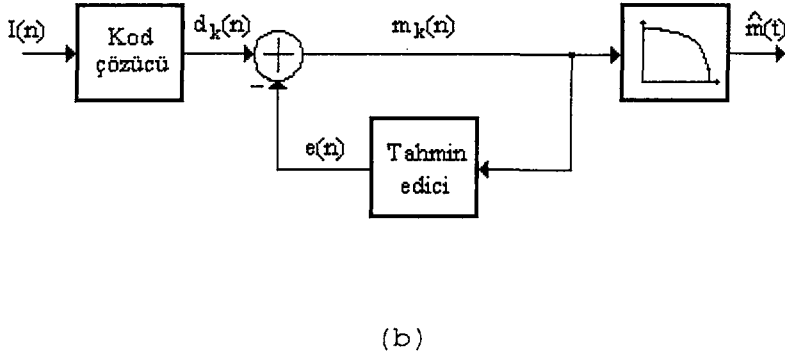
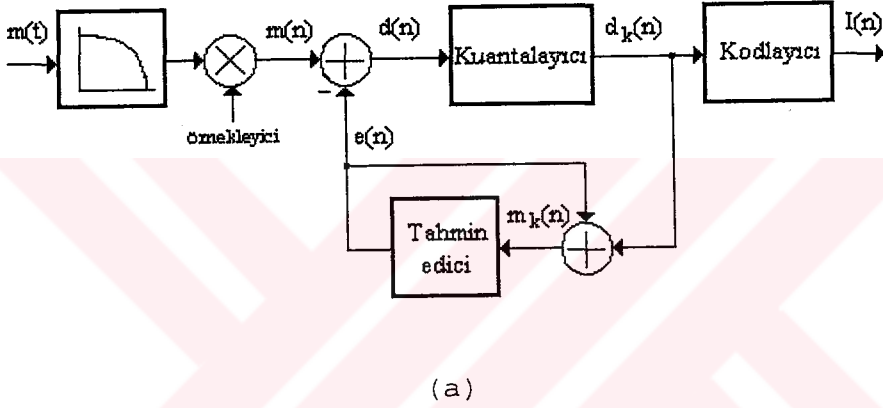
Delta modölasyonu kodlayıcısı ve kod çözücüsü

Delta modölasyonunda iřaretin iki seviyeye kuantalanmasından dolayı gürültü oluşur. Bu gürültü PCM'de olduęu gibi daha çok kuantalama seviyesi kullanılarak azaltılabilir. Bu da DPCM yöntemiyle gerçekleştirilir. Bir bit delta modölatörü kullanılırsa gürültü ancak daha sık örneklemeyle indirgenebilir. Bu ise bant genişliğinin artmasına neden olur.

### 3.2 Farksal Darbe Kod Modölasyonu (DPCM)

Yukarıda söylendięi gibi daha iyi bir randıman için kuantalayıcıda ikiden fazla seviye kullanmak gerekir. Bu daha genel yöntemle farksal darbe kod modölasyonu (DPCM) denir. DPCM'deki "farksal" terimi kodlayıcının güncel örnekle, onun

tahmini olan bir önceki örneğin arasındaki farkı kodlamasından gelir. Böylece PCM'e oranla işareti kuantalamak için daha az bit gerekir. DPCM'in blok diyagramı Şekil 3.2'de gösterilmiştir. Burada  $d(n)$  fark işareti,  $e(n)$  tahmin işareti,  $d_k(n)$  kuantalanmış fark işareti,  $m_k(n)$  yeniden oluşturulmuş giriş işareti ve  $I(n)$  çıkış kod sözcüğüdür.



Şekil 3.2

Farksal darbe kod modülasyonu  
(a)Kodlayıcı (b) Kod çözücü

DPCM sisteminin geribesleme işareti, bir sonraki örnek değerinin birinci dereceden tahminini gösterir ve örnek farkı tahmin hatasıdır. Bu görüş altında DPCM kavramı, tahmin edicinin içine birden fazla geçmiş örnek değeri dahil etmek için genişletilebilir. Genellikle kullanılan tahmin edici, geçmiş örnek tahminlerinin birkaçının ağırlaştırılmış toplamını kullanır. Lineer tahmin edicinin genel formu şöyle tanımlanır:

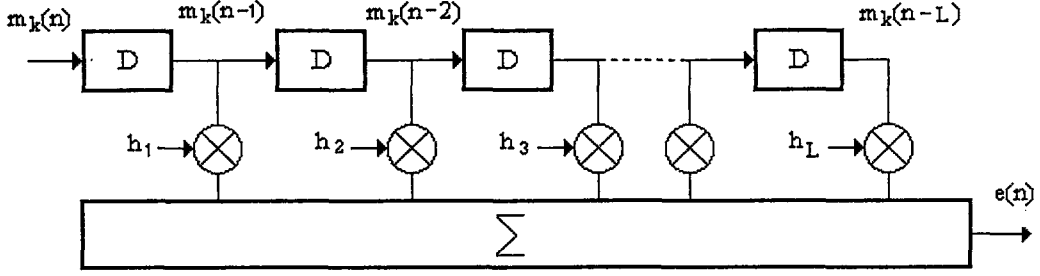
$$e(n) = \sum_{i=1}^L h_i m_k(n-1) \quad (3.1)$$

Burada  $h_i$  katsayıları ağırlaştırma çarpanları,  $m_k(n-1)$  ise bir önceki örnekleme anındaki  $m(n)$  işaretinin yeniden oluşturulmuş halidir. En basit tahmin edici bir önceki örneği, güncel örneğin tahmininin yerine kullanandır. Bu durum şu şekilde tanımlanır:

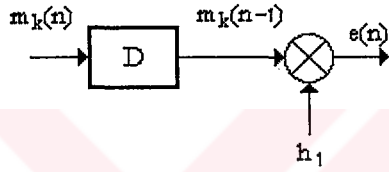
$$e(n) = h_1 m_k(n-1) \quad (3.2)$$

$h_i$  katsayıları ( $i=1,2,\dots,L$ ), tahmin hatalarını düşük bir değere indirgeyecek şekilde seçilirler.

(3.1) ifadesini gerçekleştiren tahmin edici, rekürsif olmayan sayısal filtrenin bir örneğidir. Bu shif-regiſter elemanları kullanılarak kolayca gerçekleştirilir. Őekil 3.3'de teorik blok diyagram görölmektedir. D öbekleri geçikme elemanlarıdır. Bunlar örnekleme hızıyla gelen örnekleri geciktirirler. (3.2)'de tanımlanan birinci dereceden tahmin edicinin blok diyagramı Őekil 3.3'de gösterilmektedir.



(a)



(b)

### Şekil 3.3

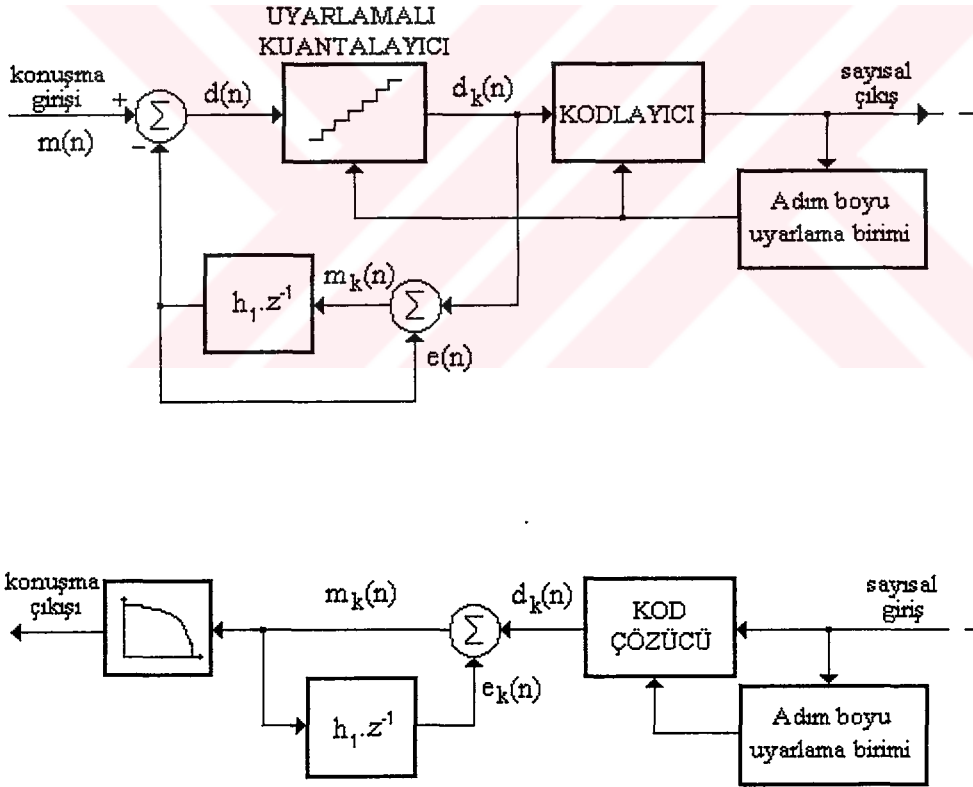
DPCM için tahmin edici

(a) genel durum (b) Birinci derece tahmin edici

#### 4. UYARLAMALI FARKSAL DARBE KOD MODÜLASYONU

##### 4.1 Uyarlamalı Farksal PCM (ADPCM)

Şekil 4.1'de uyarlamalı farksal PCM'in (adaptation differential PCM) temel blok diyagramı gösterilmiştir. Bir önceki bölümde anlatılan farksal PCM ile karşılaştırıldığında benzer yapıda olduğu görülür. Bununla birlikte uyarlamalı DPCM, kuantalayıcı çıkışında adım boyunu uyarlayan *adım boyu uyarlama birimine* sahiptir.



Şekil 4.1

Uyarlamalı farksal PCM'in blok diyagramı

Örneklenmiş giriş işareti  $m(n)$ 'den tahmin işareti  $e(n)$  çıkarılarak fark işareti  $d(n)$  elde edilir:

$$d(n) = m(n) - e(n) \quad (4.1)$$

Eğer tahmin doğru yapılmışsa fark işareti  $d(n)$  giriş işareti  $m(n)$ 'den daha küçük olacaktır. Böylece mevcut kuantalama düzeylerine göre  $d(n)$ 'yi daha düşük bir gürültüyle kuantalamak mümkün olacaktır. Kuantalanmış fark işareti  $d_k(n)$ 'ne tahmin işareti  $e(n)$  eklenerek giriş işaretinin yaklaşığı olan  $m_k(n)$  elde edilir:

$$m_k(n) = e(n) + d_k(n) \quad (4.2)$$

Böylece (4.1) ve (4.2) ifadeleri karşılaştırıldığında  $m_k(n)$ 'nin  $m(n)$ 'den farklı olduğunu görmek mümkündür. Bunun nedeni  $d(n)$ 'nin kuantalanmasından dolayı oluşan hatadır. Tahmin işareti  $e(n)$  lineer birinci derece tahmin ile, giriş işaretinin yeniden oluşturulmuş biçimi olan  $m(n)$ 'den elde edilir (3.2). Tahmin işaretiyle ilgili daha ayrıntılı bilgi bir önceki bölümde verilmiştir.

Kuantalama gürültüsü kuantalama aralığının işaretin tepe değerine göre ayarlanmasıyla azaltılabilir. Bu yüzden adım boyu, düşük seviyeli işaretler (fricative) için küçük, büyük seviyeli işaretler için ise büyük olmalıdır.

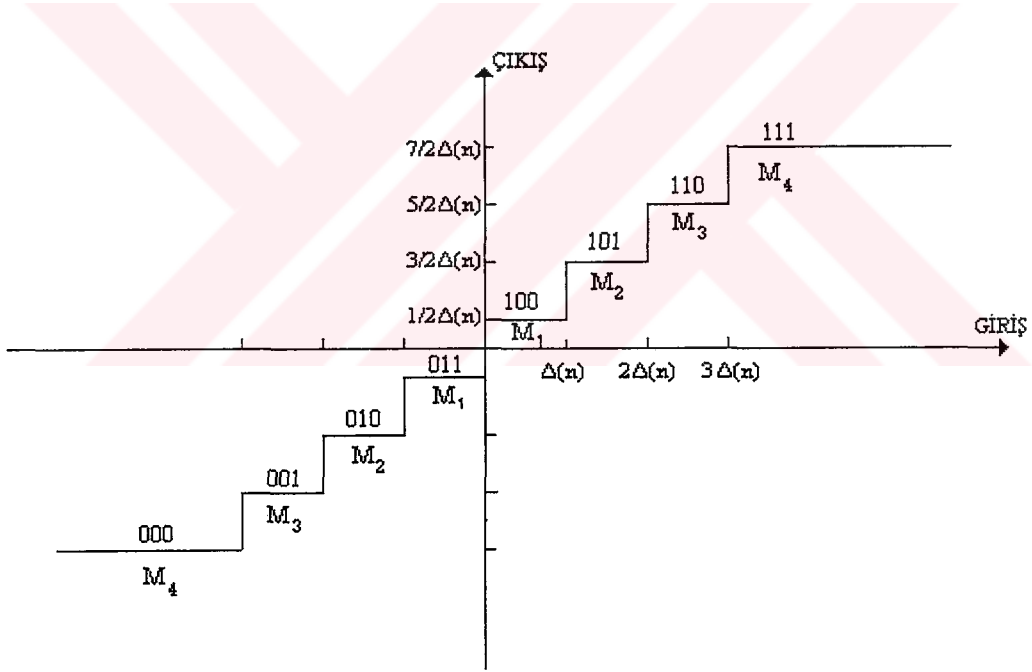
Şekil 4.1'deki adım boyu uyarlama birimi, kodlanmış çıkışı izler ve bir önceki kuantalayıcı çıkışına göre adım boyunu ayarlar. Örneğin bir önceki kod sözcüğünün değeri büyük ise  $\Delta(n)$  artırılırken, bir önceki kod sözcüğünün değeri küçük ise  $\Delta(n)$  azaltılır. Bu aşağıdaki eşitlikle başarılır:

$$\Delta(n) = \Delta(n-1)R(|I(n-1)|) \quad (4.3)$$

Burada  $R$  bir önceki kod sözcüğü  $I(n-1)$ 'in fonksiyonudur.  $|I(n-1)|$  değeri küçük olursa  $\Delta(n)$ 'nin azaltılması için  $l < 1$  seçilir ve  $|I(n-1)|$  değeri büyük olursa  $\Delta(n)$ 'nin artırılması için  $l > 1$  seçilir.

3-bit bir örnek kuantalayıcı için adım boyu çarpanları Şekil 4.2'de gösterilmiştir. Sadece dört çarpan olduğuna dikkat ediniz. Çünkü kuantalayıcı çıkışının kutbiyeti uyarlama mantığında kullanılmaz. Bu dört çarpan arasında aşağıdaki bağıntı vardır:

$$M_1 \leq M_2 \leq M_3 \leq M_4 \quad (4.4)$$



Şekil 4.2

3-bit ADPCM kodlayıcı için kuantalama düzeyleriyle ilişkilendirilmiş uyarlama çarpanları

Adım boyunu uyarlama konuşma genliğini sıkıştırmaya eşdeğerdir. Böylece kuantalayıcı düşük düzeyli işaretleride, yüksek seviyeli işaretlerle hemen hemen aynı nitelikte işler. Uyarlamanın stratejisi bütün kuantalama düzeylerinin yaklaşık eşit bir yüzde ile kullanmaktır. Bununla birlikte işaret genliği sıfıra yaklaştığında, adım boyu da sıfıra doğru gider. Bu nedenle ve pratik uygulamaların getirdiği kısıtlamalardan dolayı adım boyu aşağıdaki sınırlandırmaya göre belirlenir:

$$\Delta_{\min} \leq \Delta(n) \leq \Delta_{\max} \quad (4.5)$$

İşaret seviyesi birkaç milisaniyelik zaman periyodu süresince çok küçük olursa adım boyu minimum değerine düşer ve bu seviyede kalır. Böylece fark işareti çok düşük kuantalama düzeylerinin sınırları içinde kalacaktır. Bu yüzden konuşma olmadığı zaman kod sözcükleri örnekten örneğe sadece biraz değişecektir ve kod sözcüklerinin değeri küçük olacaktır. ADPCM kodlamanın bu özelliği konuşmanın varlığını belirlemeyi kolaylaştırır.

#### 4.2 Adım Boyu Çarpanlarının Tasarımı

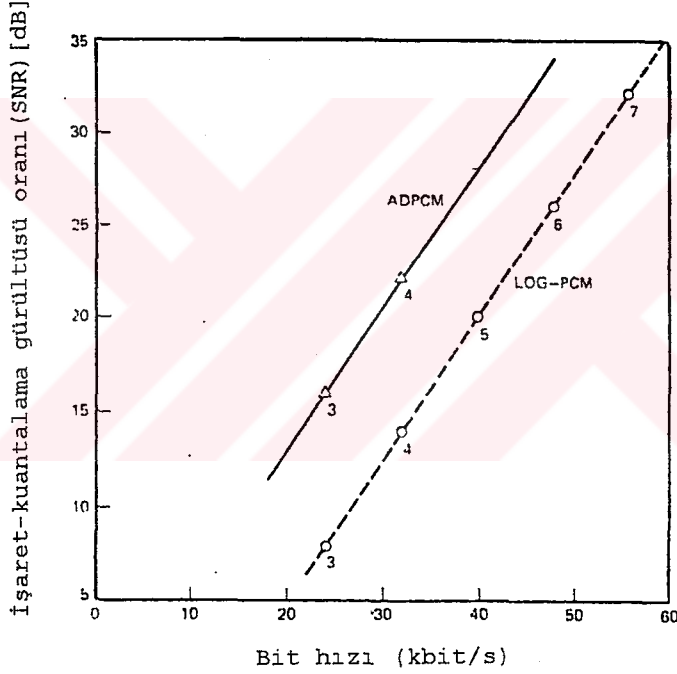
Adım boyu çarpanlarını dizayn ederken iki zıt şartla karşılaşılır. Birinci şart, giriş varyansının ani değişikliklerine çabuk cevap vermedir (3 bit için  $M_4 \gg 1$  ve  $M_1 \ll 1$  kullanılması önerilir). İkinci şart sabit veya kararlı durumda aşırı adım boyu değişikliklerini önlemektir. Bu nedenle uzlaşma değerleri giriş işareti veya giriş işaretleri sınıfı için önerilir. Konuşma örneklerine en uygun çarpan değerlerini belirlemek için bilgisayar simülasyonu kullanılır. Çarpan değerleri işaret kuantalama gürültüsü oranını (SNR) artıracak şekilde seçilir.



### 4.3 PCM ile ADPCM'in Randıman Karşılaştırması

#### 4.3.1 İşaret-Kuantalama Gürültüsü Oranı (SNR) Karşılaştırması

Konuşma işaretini işleyen bilgisayar simülasyonları 3-bit ADPCM kodlayıcı için 16 dB'lik bir SNR gösterir. Böylece PCM'e göre 8 dB'lik bir iyileşme sağlanır. Bu iyileşmenin 4 dB'li farksal kodlayıcıdan ve geri kalan 4 dB ise kuantalayıcının uyarlamasından kaynaklanır. Şekil 4.3'de, ADPCM ve log-PCM için SNR karşılaştırması yapılmıştır. Buradan ADPCM sistemiyle daha iyi bir SNR elde edildiği görülmektedir.



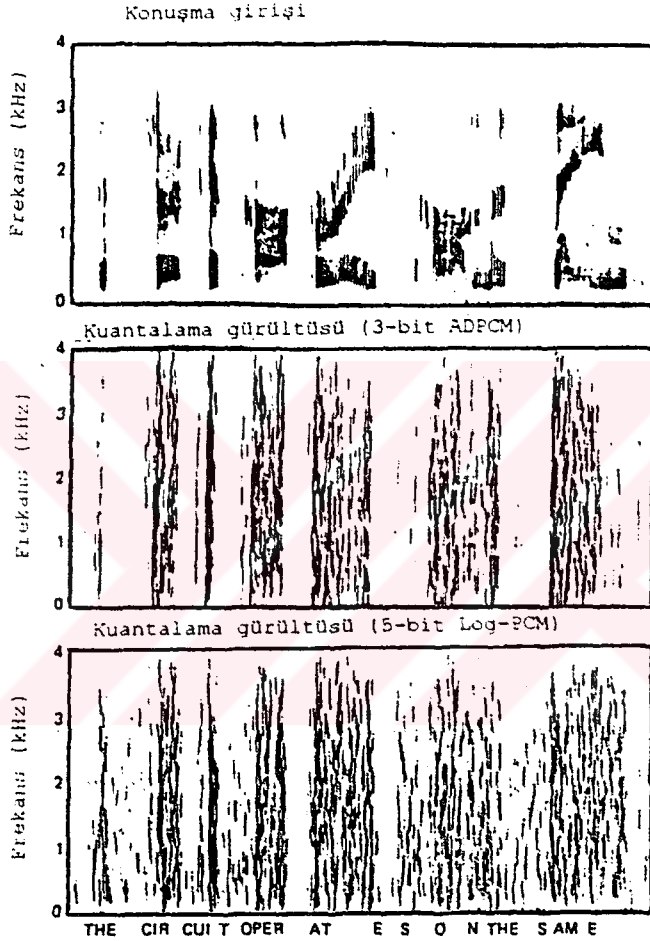
Şekil 4.3

Konuşma işaretleri için işaret-kuantalama gürültüsü oranları

#### 4.3.2 Gürültü spektrogramları

Şekil 4.4.'de konuşma girişinin spektrogramları ve 3-bit ADPCM ve 5-bit log PCM'e göre ilişkilendirilmiş kuantalama

gürültüsü spektrogramları gösterilmiştir. ADPCM'in konuşmanın sessiz aralıkları esnasında, hatırı sayılır bir azlıkta gürültü yaptığı dikkat çekicidir. Bu ADPCM'deki uyarlamalı kuantalamanın PCM'de kullanılan logaritmik sıkıştırıcıdan daha geniş bir dinamik aralık sağladığını gösterir.



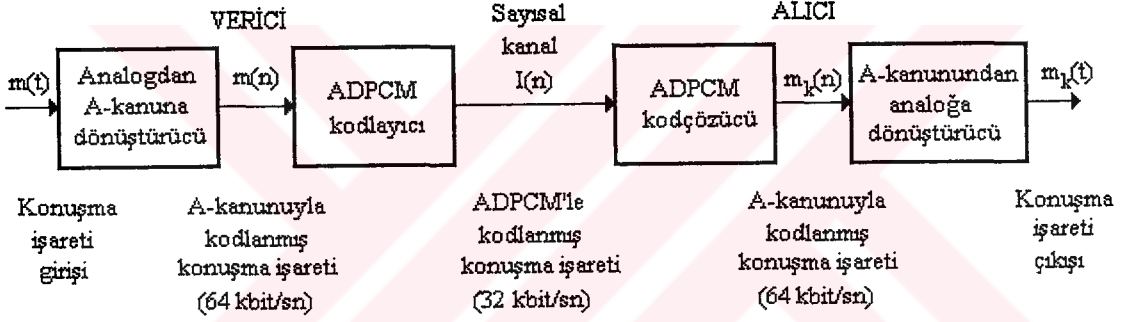
Şekil 4.4

Konuşma ve kuantalama hatasının spektrogramları

## 5. ADPCM'İN SAYISAL İŞARET İŞLEMCİSİYLE GERÇEKLEŞTİRİLMESİ

### 5.1 Uçtan Uça Konfigürasyon

Şekil 5.1'de ADPCM'in sayısal işaret işlemcisiyle (DSP) gerçekleştirilmiş temel konfigürasyonu gösterilmiştir. Giriş analog işareti  $m(t)$  örneklenir ve örneklenmiş  $m(n)$  (burada  $n$  ayrık zaman indeksidir) veri işaretini üretmek için A-kanunuyla sıkıştırılmış PCM tarzına dönüştürülür. Bu işlemler sayısal işaret işlemcisine kadar yapılır. Daha sonra DSP'de  $m(n)$ , A-kanunu biçimden lineer PCM biçimine dönüştürülür.



Şekil 5.1

DSP'li ADPCM kodlayıcı ve kodçözücünün blok diyagramı

ADPCM kodlama tamamen verici DSP içinde yapılır. DSP'nin çıkışında kod sözcüğü  $I(n)$  (3 bit) elde edilir. Alıcıda  $I(n)$ 'nin kodunu, A-kanunu biçimine dönüştürmek için ikinci bir DSP kullanılır. Daha sonra bu işaret harici S/A dönüştürücüsüyle analog işarete dönüştürülür. Saat (clock) ve çerçeveleme işaretleri alıcı ve verici arasında ayrı hatlarda taşınır. Verilen uygulama için iletim protokolü gerçekleştirilen sistemdeki veri kanalına bağlıdır.

## 5.2 DSP İle Gerçekleştirilmiş ADPCM Algoritması

Şekil 5.2'de DSP ile gerçekleştirilmiş ADPCM algoritmasının blok diyagramı gösterilmiştir. Verici Şekil 5.2a'da ve alıcı da Şekil 5.2b'de gösterilmiştir. Şekil 5.2'den görüleceği gibi ADPCM algoritması üç temel kısma ayrılabilir: uyarlamalı PCM kuantalayıcı, farksal tahmin edici döngüsü ve kuantalayıcı için uyarlamalı adım boyu kontrolü. Bu işlemler aşağıdaki paragraflarda sırayla incelenecektir.

## 5.3 Uyarlamalı Kuantalayıcı

İlk önce tahmin işareti  $e(n)$ , fark işareti  $d(n)$ 'yi elde etmek için giriş işareti  $m(n)$ 'den çıkarılır:

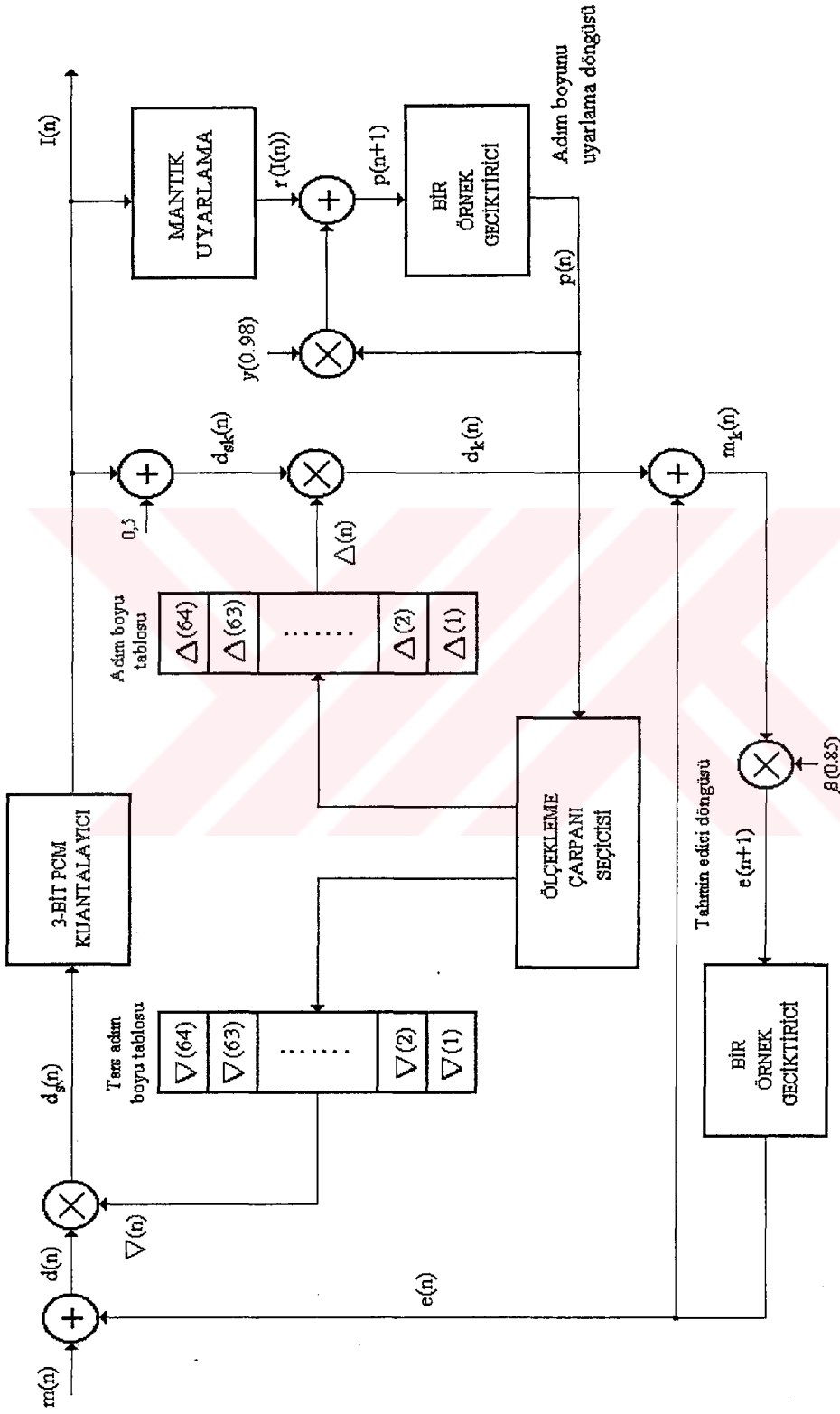
$$d(n) = m(n) - e(n) \quad (5.1)$$

$e(n)$  işareti tahmin edicide elde edilir ve bir önceki örnekleme anı  $(n-1)$ 'de yapılan hesaplamalarla belirlenir. Daha sonra fark işareti  $d(n)$ , ADPCM kod sözcüğü  $I(n)$ 'yi üretmek için uyarlamalı kuantalayıcıda kuantalanır. Bu uyarlamalı kuantalama işleminde 3 bit adım boylu kuantalayıcının değişimine göre  $d(n)$  ölçeklenir. Bu ölçeklenmiş işaret şöyle tanımlanır:

$$d_s(n) = \nabla(n) \cdot d(n) \quad (5.2)$$

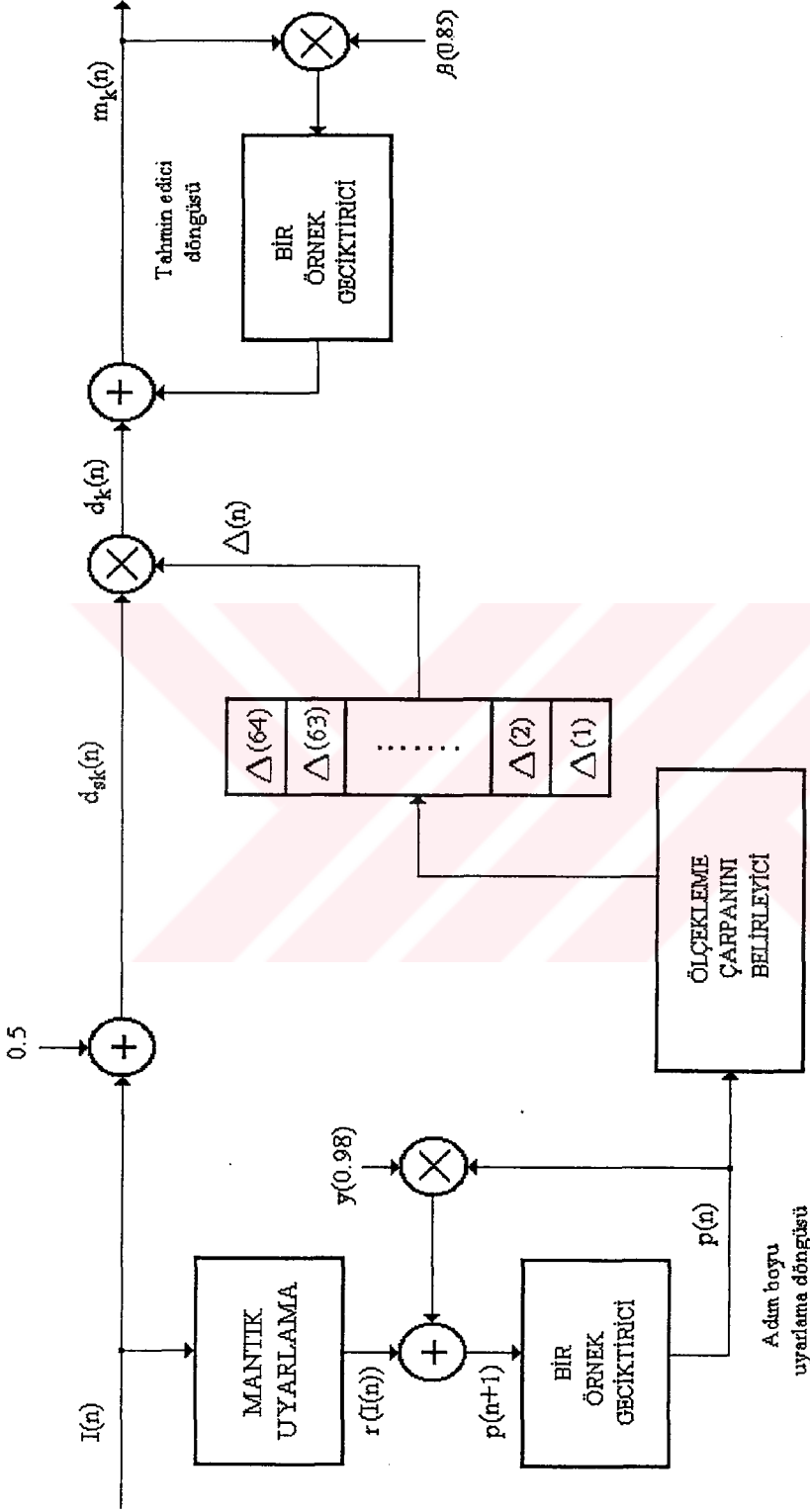
Burada  $\nabla(n)$  bir önceki örnekleme anında mevcut veriden belirlenen ölçekleme çarpanıdır. Sabit adım boylu kuantalayıcıyı  $\nabla(n)$ 'e göre ölçekleme, uyarlamalı kuantalayıcıya eşdeğerdir.

Şekil 5.3'de 3 bit (8 seviye) sabit adım boylu kuantalayıcının karakteristiği gösterilmiştir. Kuantalayıcının girişindeki  $d_s(n)$  işareti (kesirli sayıdan oluşur),  $(-4,+3)$  aralığındaki tamsayı kod sözcüklerinin  $(I(n))$  sırayla karşılığı



Şekil 5.2a

DSP ile gerçekleştirilmiş ADPCM kodlayıcısının blok diyagramı



Şekil 5.2b

DSP ile gerçekleştirilmiş ADPCM kod çözücüsünün blok diyagramı

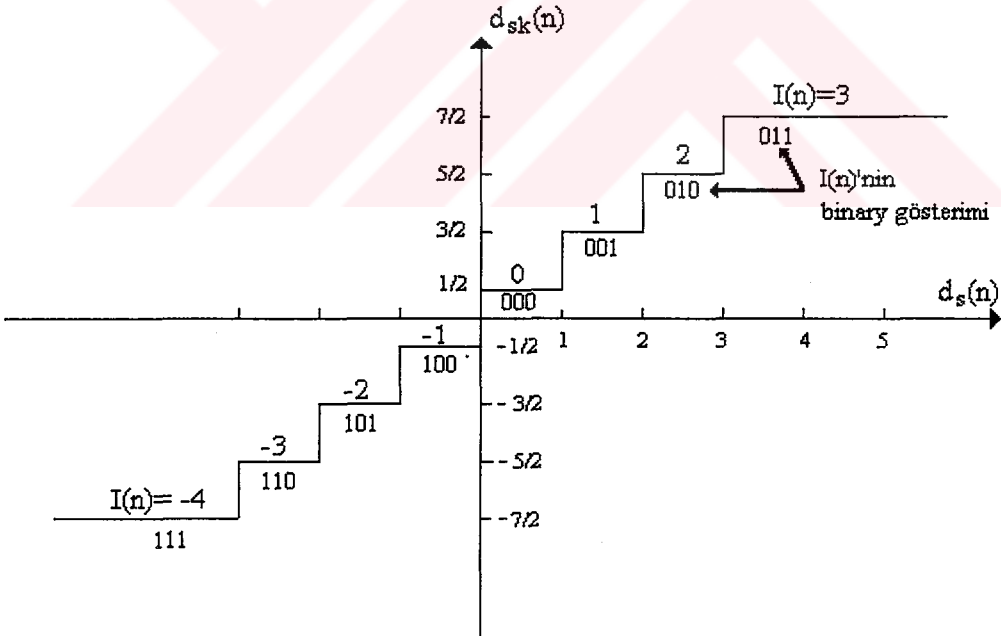
olan  $(-7/2, +7/2)$  aralığındaki 8 kuantalama seviyesinden birine dönüştürülür.  $d_{sk}(n)$  olarak gösterilen kuantalanmış çıkış işareti, kod sözcüğü  $I(n)$ 'yle şu şekilde bağıntılıdır:

$$d_{sk}(n) = I(n) + 0,5 \quad (5.3)$$

Ölçeklenmemiş ve kodu çözülmemiş fark işareti  $d_k(n)$ ,  $d_{sk}(n)$ 'nin  $\Delta(n)$ 'e göre yeniden ölçeklenmesiyle elde edilebilir. Burada  $\Delta(n), \nabla(n)$ 'nin ters bağıntısıdır ve kuantalayıcının adım boyuyla doğrudan orantılıdır. Böylece,

$$d_k(n) = \Delta(n) \cdot d_{sk}(n) \quad (5.4)$$

ifadesi fark işareti  $d(n)$ 'nin uyarlamalı olarak kuantalanmış gösterimidir. Bu işlem ile birlikte algoritmanın uyarlamalı kuantalama kısmı tamamlanır.



Şekil 5.3

3-bit ADPCM kuantalayıcının karakteristiği

#### 5.4 Birinci Dereceden Tahmin Edici

$e(n)$  ve  $d_k(n)$ 'nin toplamı, giriş işaretinin uyarlamalı olarak kuantalandıktan sonra yeniden elde edilmiş gösterimi olur. (Şekil 5.2a'ya bakınız):

$$m_k(n) = e(n) + d_k(n) \quad (5.5)$$

Bununla birlikte tahmin edici değeri  $e(n+1)$  ve kuantalayıcı ölçek çarpanları  $\Delta(n+1)$  ve  $\nabla(n+1)$  ancak bir sonraki örnekleme anı  $(n+1)$ 'de güncel olabilmektedir.

Yeni tahmin edici değeri şu şekilde hesaplanır:

$$e(n+1) = \beta \cdot m_k(n) \quad (5.6)$$

Burada  $\beta$  ağırlaştırma çarpanıdır. Genellikle konuşma kodlama için  $\beta=0.85$  değeri kullanılır. Daha sonra  $e(n+1)$ 'in değeri bir sonraki örnekleme anı  $(n+1)$ 'de kullanılmak üzere depolanır.

#### 5.5 Uyarlamalı Adım Boyu Kontrolü

Bir sonraki örnekleme anı  $(n+1)$  için  $\Delta(n+1)$  ve  $\nabla(n+1)$ 'in yeni değerini hesaplama daha karışıktır. Burada kullanılan algoritma, kuvvetli (robust) adım boyu uyarlama yöntemine dayanır. Yukarıda anılan adım boyunu uyarlama yönteminde, kuantalayıcı düzeyleri ölçeklenir. Böylece kuantalayıcı düzeyleri daha verimli kullanılır. Bu ölçekleme işlemi  $\Delta(n)$ 'le orantılı olduğu için, bundan sonra  $\Delta(n)$  adım boyu diye adlandırılacaktır.

Kuvvetli uyarlama algoritmasında, adım boyu  $\Delta(n+1)$  aşağıdaki gibi seçilir:

$$\Delta(n+1) = (\Delta(n))^Y R(I(n)) \quad (5.7)$$



Bu ifadede,  $n$  örnekleme anındaki adım boyu  $\Delta(n)$ 'nin  $\gamma$ 'e göre ( $0 < \gamma \leq 1$ , genellikle  $\gamma = 0.98$ ) kuvveti alınır ve  $I(n)$  kod sözcüğünün fonksiyonu olan  $R(\cdot)$  çarpanıyla ölçeklenir.  $I(n)$  kod sözcüğü, kuantalayıcının üst değerlerinden biri ise (örneğin  $I(n) = -4, -3, -2$  veya  $3$ ; Şekil 5.3'e bakınız),  $R(\cdot)$ 'in değeri bir sonraki örnekleme anında kuantalayıcının adım boyunu artırmak için birden büyük seçilir.  $I(n)$  kod sözcüğü kuantalayıcının düşük değerlerinden biri ise (örneğin Şekil 5.3'de  $I(n) = -2, -1, 0, 1$ ),  $R(\cdot)$ 'in değeri bir sonraki örnekleme anında, kuantalayıcının adım boyunu azaltmak için birden küçük seçilir. Bu yolla, algoritma uyarlamalı kuantalayıcının dinamik aralığını  $d(n)$  işaretinin aralığına uymasını sağlayacak şekilde adım boyunu sürekli olarak uydurmaya çalışır.

Bu algoritmayı gerçekleştirmek için daha doğrudan bir yaklaşım, (5.7) eşitliğinin logaritmik formda gösterilmesiyle sağlanır:

$$p(n) = \log(\Delta(n)) \quad (5.8)$$

ve

$$r(I(n)) = \log(R(I(n))) \quad (5.9)$$

burada logaritmanın tabanı kodlayıcıdaki parametrelerin seçimine göre belirlenir. Böylece (5.7) eşitliği aşağıdaki eşitliğe dönüşür:

$$p(n+1) = \gamma \cdot d(n) + r(I(n)) \quad (5.10)$$

Bu ifade birinci dereceden fark denklemi biçimine sahiptir ve Şekil 5.2a'daki blok diyagramın sağ üst köşesinde gerçekleştirilen eşitliktir.

$r(I(n))$ ,  $I(n)$  kod sözcüğünün fonksiyonudur ve adım boyu uyarlama algoritmasıyla belirlenir. Bu algoritma şu şekilde dir:

$$r(I(n)) = \begin{cases} |I(n) + 0.5| \geq 2.5 & \text{ise } 8 \\ |I(n) + 0.5| < 2.5 & \text{ise } -3 \end{cases} \quad (5.11)$$

Adım boyunu uyarlama döngüsünün çıkışı, logaritmik formda olan  $\rho(n)$ 'dir. Bu nedenle  $\Delta(n)$ 'i elde etmek için (ve  $\nabla(n)$ 'i),

$$\Delta(n) = \exp(\rho(n)) \quad (5.12)$$

ve

$$\nabla(n) = \frac{1}{\Delta(n)} = \exp(-\rho(n)) \quad (5.13)$$

eşitliklerini gerçekleştirmeye ihtiyacımız vardır. (5.8)'den (5.13)'e kadar olan eşitliklerde logaritmaların ve exponansiyellerin tabanları daha önce belirtildiği gibi, kodlayıcıdaki parametrelerin seçimine göre belirlenir.

(5.12) ve (5.13)'deki exponansiyeller ROM programda yüklenmiş olan arama tabloları kullanılarak hesaplanır.  $\rho(n)$ 'nin tamsayı değeri alınır ve 64'lü tablo ölçüsüne göre  $-32 \leq [\rho(n)] < 32$  aralığında sınırlandırılır.  $\rho(n)=0$  değeri tablodaki merkez parametreler olan  $\Delta[32]$  ve  $\nabla[32]$ 'ye karşı düşer. Giriş genliğinin yüksek bir değeri için 32 ile 64 arasındaki indislerden, düşük bir değeri için ise 0 ile 32 arasındaki indislerden uygun olanı seçilir. Şekil 5.2a'daki tablolarda parantez içine alınmış büyüklüklerin ( $\Delta[.]$  ve  $\nabla[.]$ ),  $[.]$  ile adreslenmiş tablo yerlerinin içeriklerine ait olduğuna dikkat ediniz.

Yukarıdaki algoritma  $\Delta(n)$  ve  $\nabla(n)$ 'nin 64 farklı adım boyu değerini kullanır. Tablo değerleri giriş işaretinin istenen

dinamik deęişimini karřılayacak řekilde seilirler. Tablo adres yerleri, bir sonraki rnekleme anında  $\Delta(n+1)$  ve  $\nabla(n+1)$ 'e eriřmek iin depolanır.

### 5.6 Adım Boyu Tabloları İin Parametre Seimi

Adım boyu tablosunun parametreleri, maksimum-minimum adım boyu oranı 256 olacak řekilde seilir:

$$\Delta[64]/\Delta[1] = 256 = 2^8$$

ve

$$\nabla[1]/\nabla[64] = 256$$

Bu konuřma kodlama iin tahsis edilen 48 dB(8 bit)'li adım boyu uyarlama aralıęını verir. Her bir tabloda 64 tane logaritmik olarak leklenmiř adım boyu deęeri vardır. Bu 0.75 dB'lik bir netlięe karřı dūřer. Ařaęıdaki eřitliklerden grleceęi gibi adım boyları, 1.0902'nin oranıyla artmaktadır:

$$\Delta[i] = \Delta[1] \cdot (1.0902)^{i-1}$$

$$\nabla[i] = \nabla[1] \cdot (1.0902)^{i+1}$$

Maksimum adım boyu,  $d(n)$ 'nin maksimum deęerinin kuantalayıcının maksimum aralıęına gre leklenmesi iin seilir. Bu dinamik aralıęın st ucunda fazla yklemeyi nler. 3-bitli kuantalayıcıda en iyi dinamik aralık randımanı iin  $\Delta[.]$  ikiye gre artırılırken,  $\nabla[.]$  ise ikiye gre azaltılmalıdır.

### 5.7 ADPCM İin Kod zc

řekil 5.2b'de ADPCM kod zc iin benzer bir blok diyagram gsterilmiřtir. Giriř kod szcę  $I(n)$ 'ye 0.5 eklenir (řekil 5.3'e bakınız) ve sonu  $\Delta(n)$ 'gre leklenerek, kodu

özölmüş fark işareti biçimine dönüştürölür. Daha sonra kodu özölmüş ıkış işareti  $m_k(n)$ , kodlayıcıdakine benzer tarzda, tahmin edici öbeğinde  $d_k(n)$  değerlerinden elde edilir.



## 6. ADPCM İLE KODLANMIŞ KONUŞMA İŞARETİNİN VARLIĞININ BELİRLENMESİ

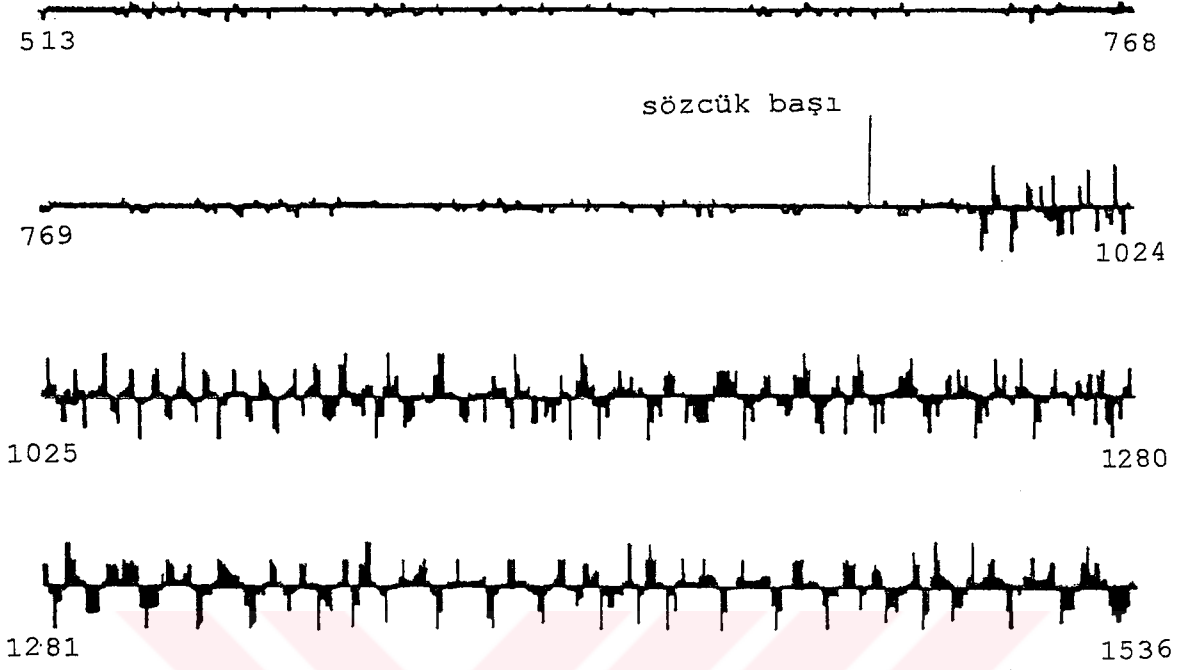
Konuşma işaretini işleyen sistemlerin bir çoğunun girişinde, konuşmanın varlığını belirlemeye ihtiyaç vardır. Örneğin sayısal iletim sistemlerinde ortalama bit hızını indirmek için konuşma işaretiyle sessizlik arasındaki farkı otomatik olarak ayırt etmek gereklidir.

Bu algoritmanın başka bir yararı bilgisayarlı sesli yanıt verme sistemleri için sözcük hafızasının hazırlanması sırasında görülür. Bu sistemlerde ilk önce sözcükler ve cümleler -sayısal hafızaya verimli bir yükleme için- ADPCM sistemiyle sayısal forma dönüştürülür. Bu temel öğeleri uzun mesajlar biçimine dönüştürürken en doğal bağlamayı üretmek için, her bir sözcük veya cümlenin başının ve sonunun doğru olarak belirlenmesi gereklidir. Aksi takdirde ard arda bağlanmış işaretle gereksiz boşluklar oluşur.

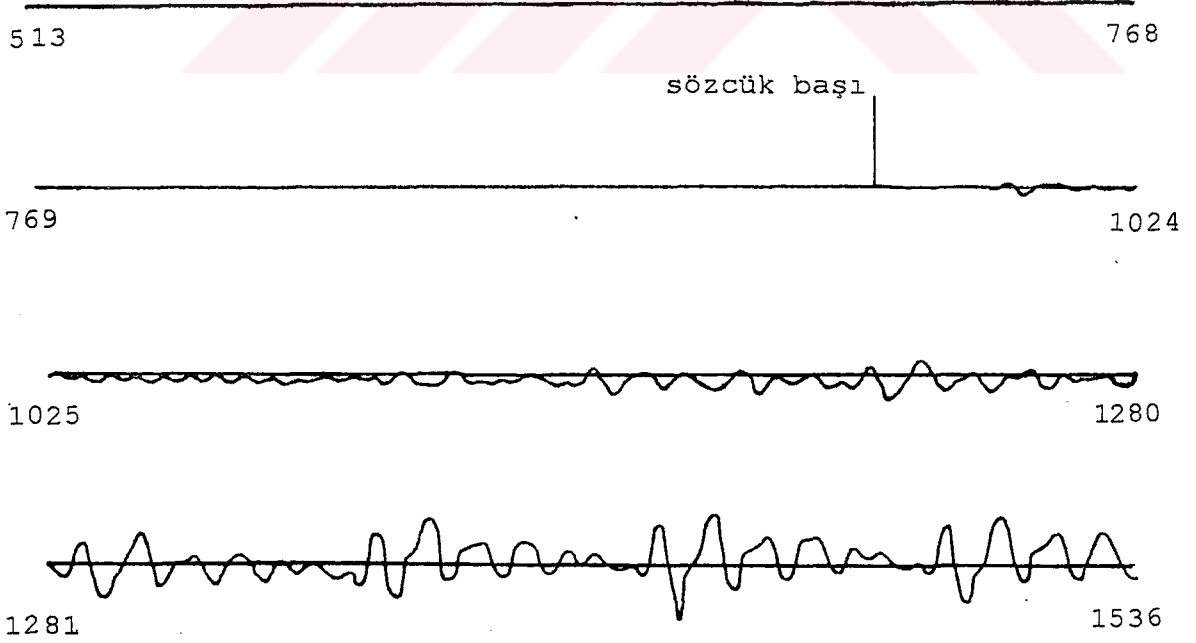
Burada kullanılan algoritma, ADPCM ile kodlanmış kod sözcüklerini kullanarak, kelimelerin veya cümlelerin başını ve sonunu kolaylıkla belirlemektedir.

### 6.1 Uçnoktaları Belirleme Algoritması

Şekil 6.1'de tipik bir kod sözcük dizisi başlangıcından itibaren gösterilmiştir. Örnekleme hızı 6 kHz'dir ve her bir satırda 256 örnek vardır. Yani her bir satır işaretin 40 milisaniyesine karşı düşer. Birinci satırın tamamına ve ikinci satırın ilk kısmına dikkat edersek, kod sözcüklerinin ilk bir kaç pozitif ve negatif değerler içinde değişmediğini görürüz. Dizinin bu ilk kısmı sessizliğe karşı düşer. Bununla birlikte, ikinci satırın sonunda ve daha sonraki iki satırda, kod sözcük dizisi daha hızlı ve yüksek genlikli dalgalanır. Bu da konuşmanın başladığını gösterir. Ayrıca bu kısım,



Şekil 6.1  
/o/ sesli harfinin kod-sözcük dizisi.



Şekil 6.2  
Şekil 6.1'deki kod sözcük dizisinin kodu çözülmüş  
dalga şekli.

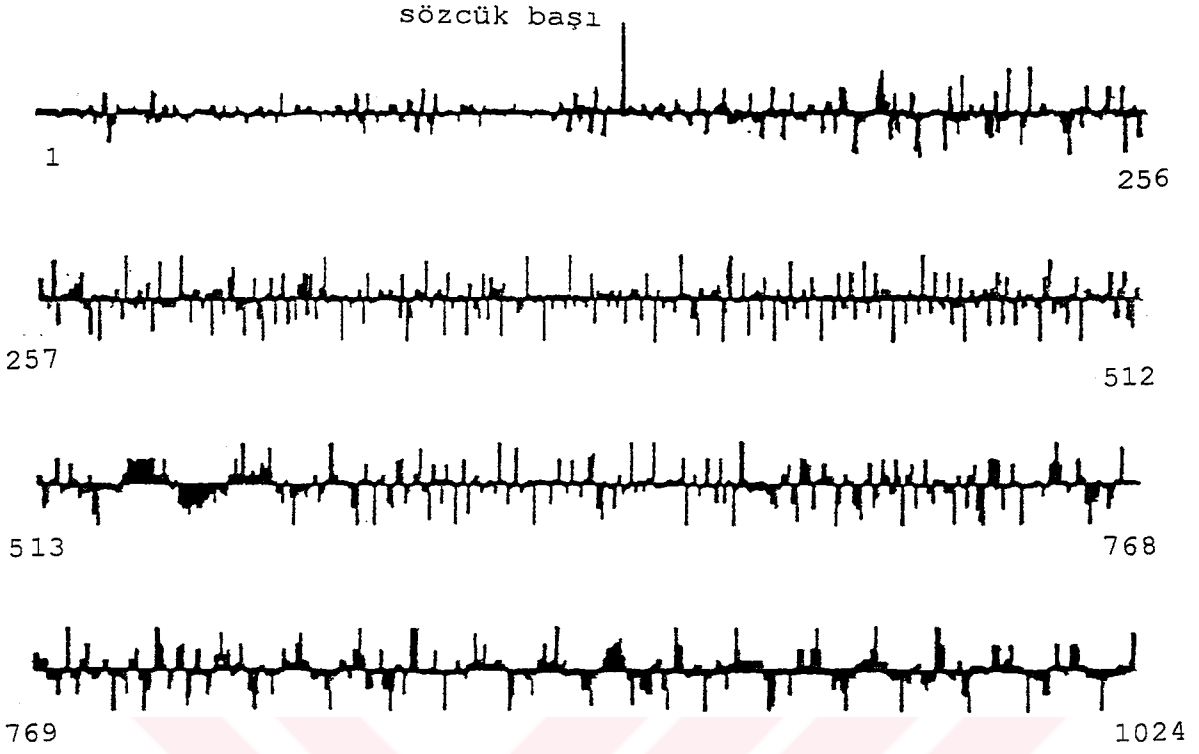
Şekil 6.1'deki kod sözcük dizisinin kodu çözülmüş konuşma işaretini gösteren Şekil 6.2'de de göze çarpar.

Konuşma işaretinin ADPCM ile gösteriminin bu özellikleri, 101 tane kod sözcüğünün karelerinin toplamı şeklinde belirlenen kod sözcük enerjisine yansır. Kod sözcük enerjisi  $E(n)$  aşağıdaki gibi hesaplanır:

$$E(n) = \sum_{i=n-50}^{n+50} [C(i)-7.5]^2 \quad (6.1)$$

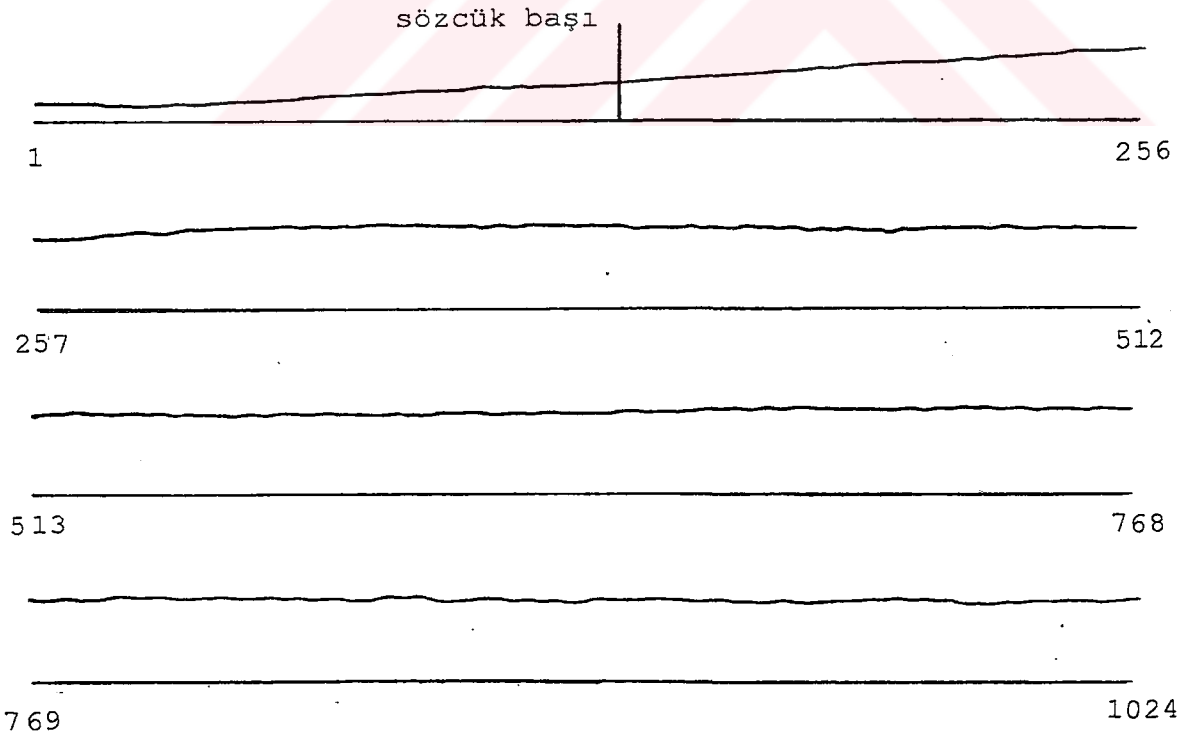
Konuşma işaretini saptama algoritmasında kod sözcük enerjisi her bir örnekte hesaplanır ve bir eşik ile karşılaştırılır. Bu eşik, sessizliğin enerjisiyle konuşma işaretinin ortalama enerjisinin farkının yarısı olarak belirlenir. Kod sözcük enerjisi ardışık 300 örnek veya 50 milisaniye için bu eşiği aşarsa, enerjinin eşiği aştığı ilk nokta ifadenin başıymış gibi kaydedilir. Enerji karşılaştırmasına devam edilir ve kod sözcük enerjisinin 1000 ardışık örnek veya 160 milisaniye için eşiğin altına düştüğü ilk nokta ifadenin sonu olarak kaydedilir. Sözcük veya cümle içinde durak uyumunu sağlayan 160 milisaniye kriteri ifade sonu yanılığını önler.

Yukarıdaki algoritmanın çalışmasının örneği Şekil 6.3'den 6.8'e kadar olan şekillerde gösterilmiştir. Şekil 5.3 |three| sözcüğünün başlangıcından itibaren, kod sözcüklerinin dizisini göstermektedir. Birinci satırın sol yarısında çok az kod sözcük varyasyonu vardır ve burası düşük seviyeli gürültü bandına karşı düşer. İlk |th| düşük seviyesine (fricative) karşı düşen birinci satırın sağ yarısı ve diğer iki satırda belirgin bir şekilde daha büyük varyasyonlar (sesin başlangıcına karşı düşen son satırda olduğu gibi) görülmektedir. Birinci, satırın ortasındaki belirteç, az önce anlatılan algoritmaya göre yeri belirlenen başlangıç noktasıdır. Şekil 6.4'de zamanın



Şekil 6.3

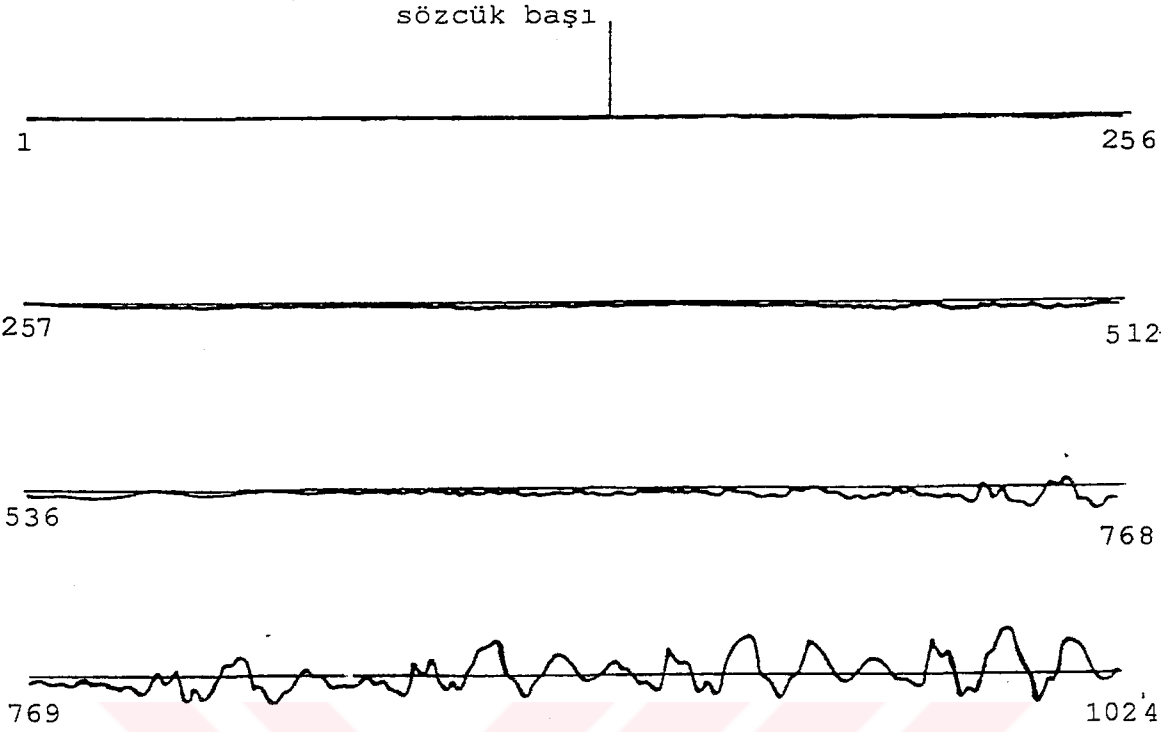
/three/ sözcüğünün başının kod-sözcük dizisi.



Şekil 6.4

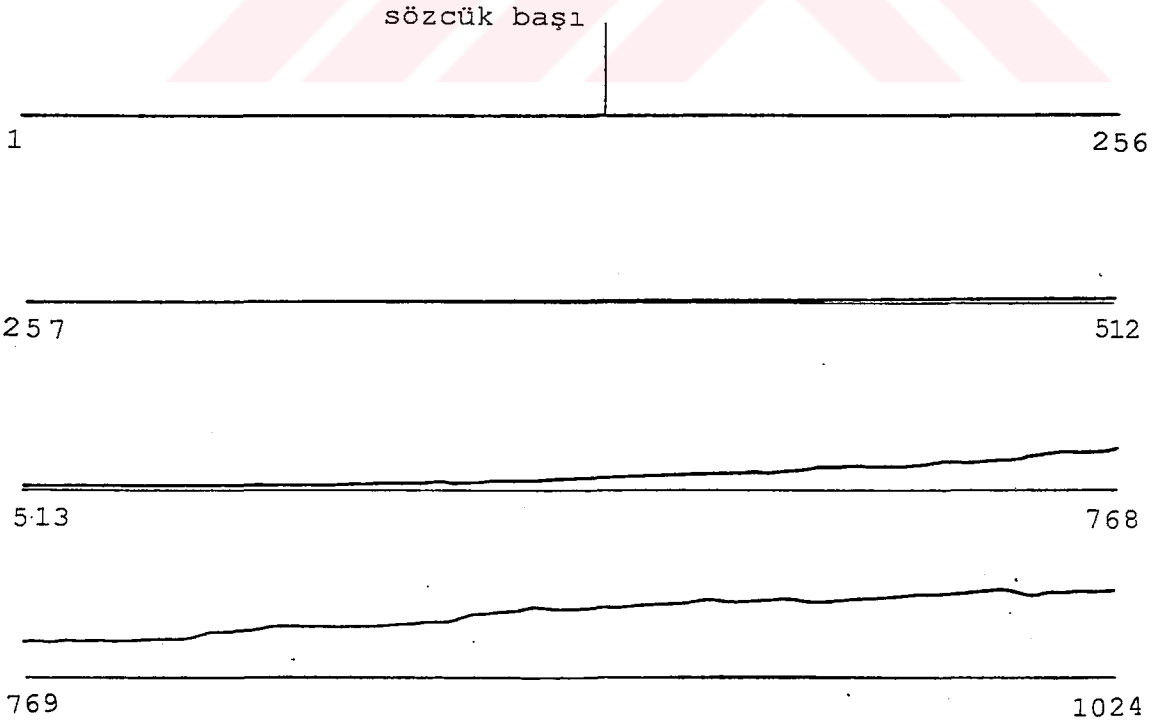
Şekil 6.3'deki kod sözcük dizisinin kod sözcük enerjisi.





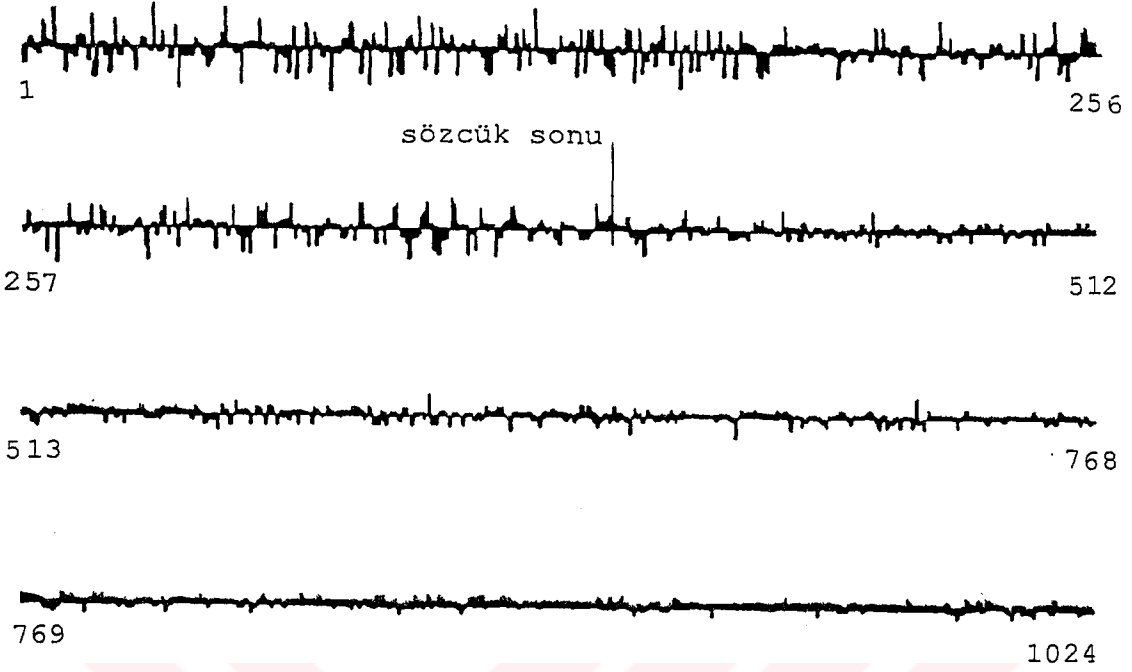
Şekil 6.5

Şekil 6.3'deki kod sözcük dizisinin kodu çözülmüş dalga şekli.



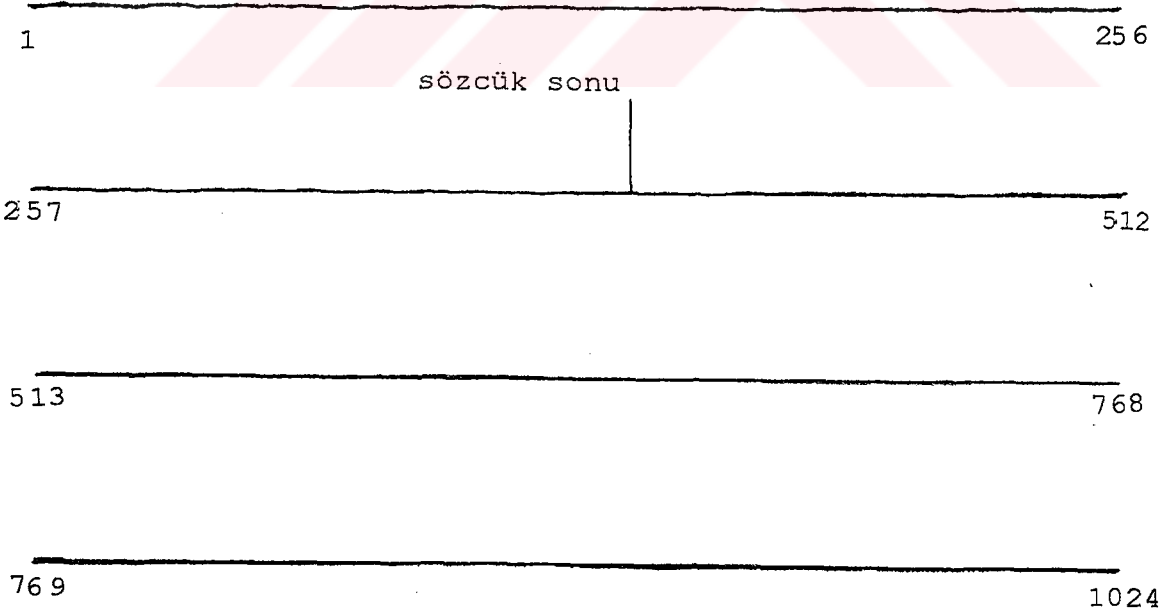
Şekil 6.6

Şekil 6.5'deki dalga şeklinin enerjisi.



Şekil 6.7

/three/ sözcüğünün sonunun kod-sözcük dizisi.



Şekil 6.8

Şekil 6.7'deki kod-sözcük dizisinin kodu çözülmüş dalga şekli.

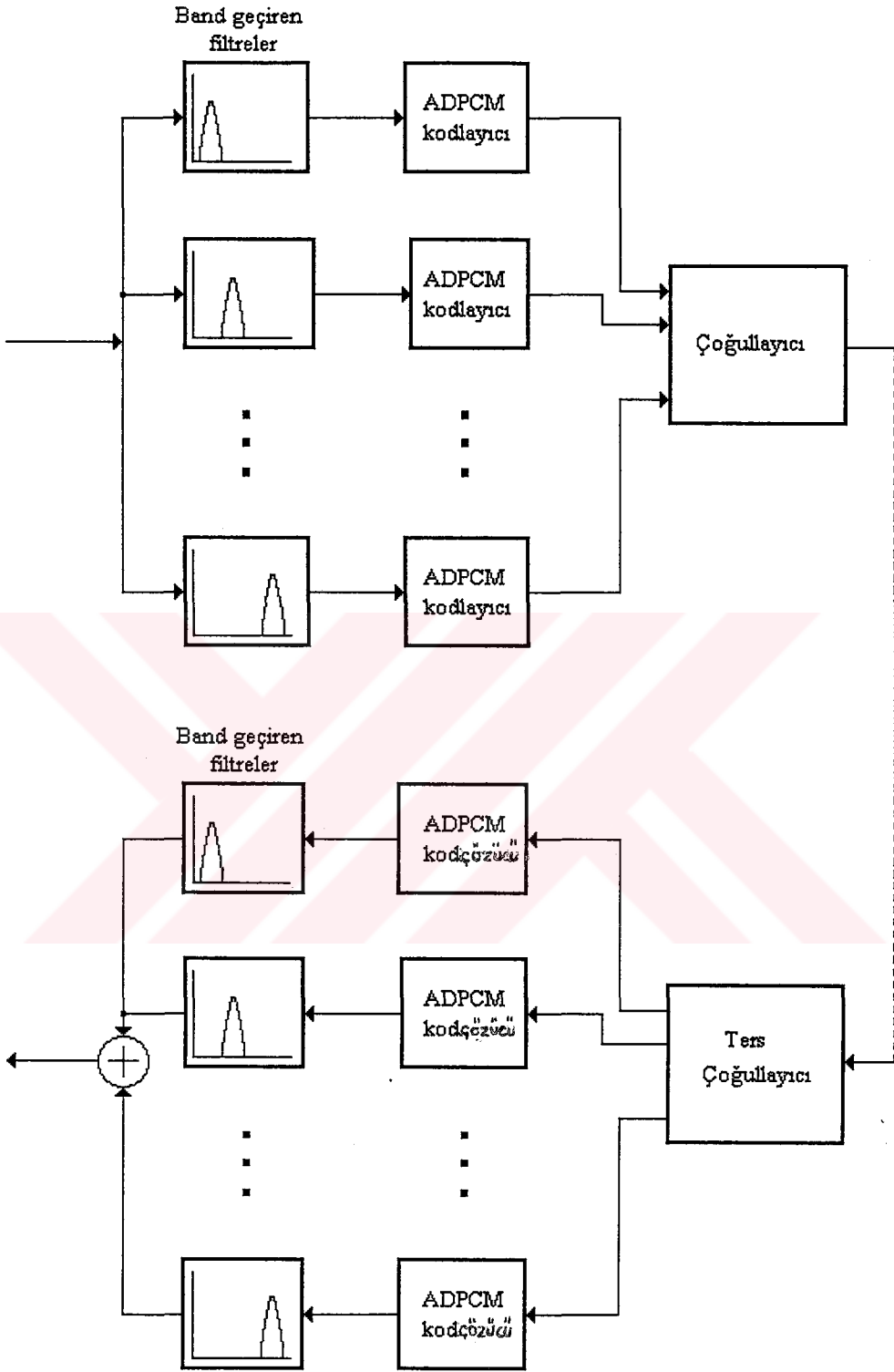
fonksiyonuna göre kod sözcük enerjisi gösterilmiştir. Ayrıca Şekil 6.4'deki belirteç enerjinin eşiği aştığı ve en az 50 milisaniye boyunca değişmeden kaldığı noktayı gösterir. Kod sözcük enerjisinin, konuşma olmadığı zaman oldukça düşük olmasına karşın, hem sesli hem de sessiz harflerin olduğu kısımlarda yaklaşık olarak aynı kaldığı dikkat çekicidir. Bu sav Şekil 6.3'deki kod sözcük dizisinin konuşma dalgaşeklini gösteren Şekil 6.5'de doğrulanır. Burada sessiz harfli kısmın başlangıcı ve bunu izleyen sesli kısım görülmektedir. Şekil 6.5'deki konuşma işaretinin enerjisini gösteren Şekil 6.6'dan, bu algoritmanın konuşma işaretinin kodlanmamış örneklerini işlerken geçerli olamayacağı görülür. Şekil 6.7'de |three| sözcüğünün kod sözcük dizisinin sonu ve otomatik olarak belirlenen son nokta (yani kod sözcük enerjinin eşiğin altına ilk düştüğü ve 160 milisaniye boyunca aşağıda kaldığı nokta) görülmektedir. Şekil 6.8'de ise kodu çözülmüş konuşma dalgaşeklini gösterilmiştir. Kod sözcük dizisinde açıkça göze çarpan son nokta konuşma işaretinin kendisinde daha az belirgindir.

## 7. ALTBAND KODLAMA

Altband kodlama, konuşma bandının band geçiren filtrelerle altbandlara bölünerek kodlandığı kodlama tekniğidir. Bu teknikte giriş işaretinin zaman domeni analizi yerine frekans domeni analizi yapılır. Kodlayıcı, ilk önce giriş spektrumunu band geçiren filtreler kullanarak ayrı bandlara ayırır. Daha sonra her bir altband ADPCM kodlayıcılarla sayısal olarak kodlanır. Konuşma bandının altbandlara bölme işlemiyle her bir altband, bu bandların algısal kriterlerine göre tercihli olarak kodlanabilir. Yeniden oluşturma durumunda, altband işaretlerinin kodu çözülür ve orjinal bandlarına dönüştürülürler. Daha sonra orjinal konuşma işaretini elde etmek için toplanırlar.

Altband kodlayıcının blok diyagramı Şekil 7.1'de gösterilmiştir. Alıcı ve vericideki bandgeçiren filtreler, konuşma girişini altbandlara bölmeye yarar. Kodlayıcılar ve kod çözücüler altband işaretlerinin kodlanması için kullanılır. Çoğullayıcı, sayısal kanalda iletim için bu sayısal işaretleri tekil bit akışı biçiminde birleştirir. Ayrıca çoğullayıcı, vericiyle alıcının senkronize çalışması için bit akışının içine senkronize bitlerini sokar.

Tablo 7.1'de 5 altbandlı kodlayıcı için bandların seçimini ve 16 kbit/s'de kullanılan bit tahsisi gösterilmiştir. Buradaki ikinci sütun her bir altband için frekans aralığını gösterir. Bit tahsisi, her bir altbanddaki kodlayıcılarda kullanılan bit/örnek sayısına aittir. Tablodan görüldüğü gibi düşük band kodlaması için daha çok bit tahsis edilir. Bunun nedeni düşük banddaki enerji seviyelerinin düşük olmasıdır.



Şekil 7.1  
Altband kodlayıcı

**Tablo 7.1**  
16 kbit/sn'de 5 bantlı altband kodlayıcı

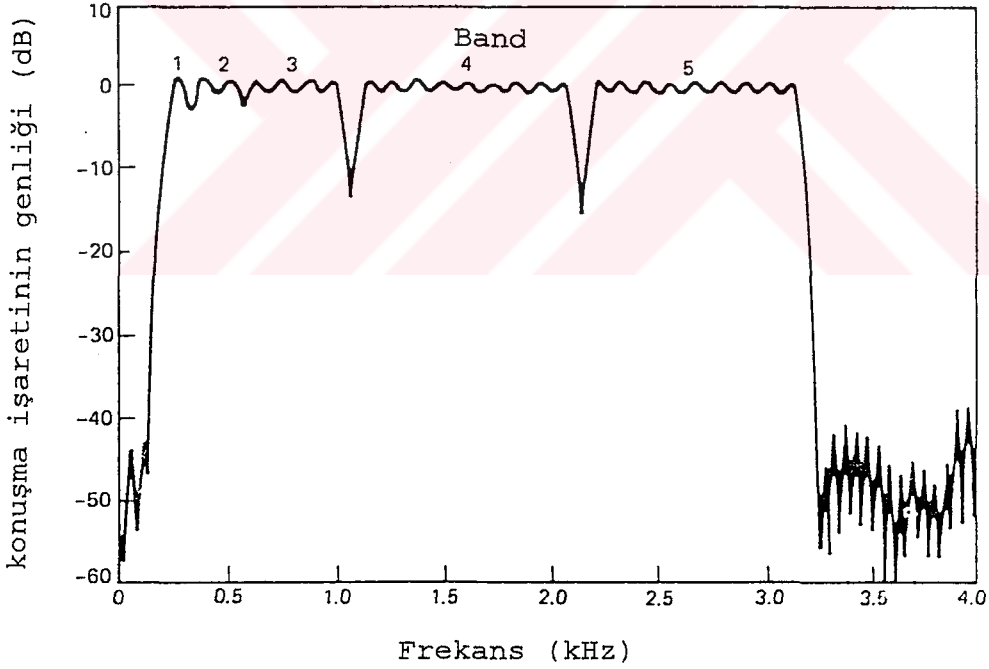
Band	Band Aralıkları (Hz)	Örnekleme Frekansı (Hz)	Tahsis Edilen Bit Sayısı	Kbit/sn
1	178-356	356	4	1.42
2	296-593	593	4	2.37
3	533-1067	1067	3	3.20
4	1067-2133	2133	2	4.27
5	2133-3200	2133	2	4.27
senkronizma biti				0.47
				+ 16.00

Kodlayıcının frekans aralığı 200-3200 Hz arasındadır. Bu frekans yanıtının grafiği Şekil 7.2'de gösterilmiştir. Bu şekilden görüldüğü gibi, 1067 ve 2133 Hz'deki frekans yanıtında iki küçük kerkik görülmektedir. Bu kerkikler dördüncü ve beşinci bantlardaki süzgeçlerin çevirme bantlarından ileri gelir. Sübjektif olarak bu kerkiklerin etkisi fazla belirgin değildir. Düşük frekanslarda, bu gibi kerkiklerden kaçınmak için, birden üçe kadar olan bantlar çakıştırılır.

Her alt bandı ayrı ayrı kodlama bir çok nedenden dolayı avantajlıdır. Birincisi her band için ayrı uyarlama kullanarak kuantalayıcıların adım boyları her bandaki enerji seviyelerine göre ayarlanır. Böylece yüksek seviyeli bantlar daha kaba kuantalamayla kodlanırken, düşük seviyeli bantlar da daha ayrıntılı kuantalamayla kodlanır. Alt bandın ikinci avantajı her bantta tespit edilen bit hızının, her bandın algısal önemine göre optimize edilebilmesidir. Özellikle düşük frekanslar için daha çok bit tahsisi yapılabilir. Bununla

birlikte yüksek frekanslar için örnek başına daha az bit kullanılır.

Altband kodlayıcı, uyarlamalı delta modülasyonu ve uyarlamalı farksal PCM'e göre önemli ölçüde bit hızlarını düşürür. Özellikle 16 kbit/sn altband kodlayıcı 22 kbit/sn ADPCM kodlayıcıya eşdeğerdir. Altband kodlamaya ait önemli bir örnek geniş bantlı konuşma kodlama için CCITT Recom. G.722'dir. Bu standart 64 kbit/sn'de 7 kHz konuşma bandının kodlamasını sağlar ve böylece uçtan uca sayısal kanallarda ses doğruluğunu iyileştirir. Bu standart bir sonraki bölümde anlatılacaktır.



Şekil 7.2

Tablo 7.1'deki 5 bandlı kodlayıcının frekans yanıtı

## 8. 64 KBIT/SN'DE 7 KHZ İŞİTSEL KODLAMA

### 8.1 64 Kbit/sn'de İşitsel Kodlama

Bu bölümde yüksek nitelikli işitsel (50 ile 7000 Hz arası) kodlama sisteminin karakteristikleri anlatılacaktır. Bu kodlama sisteminde 64 kbit/sn bit hızlı altband uyarlamalı farksal darbe kod modülasyonu kullanılır. Bundan sonra bu sistem 64 kbit/sn (7 kHz) işitsel kodlayıcı diye adlandırılacaktır. Kullanılan altband-ADPCM tekniğinde frekans bandı iki alt banda (yüksek ve alçak) ayrılır ve her iki alt bandda işaretler ADPCM ile kodlanır. Sistemin üç çalışma modu vardır. Bu çalışma modları 7 kHz işitsel kodlama için kullanılan bit hızlarına karşı düşer: 64, 56 ve 48 kbit/sn. Son iki mod 8 ve 16 kbit/sn'li yardımcı veri kanalına olanak tanır.

Şekil 8.1'de gösterilen 64 kbit/sn (7 kHz) işitsel kodlayıcının temel işlevsel kısımları aşağıda açıklanmıştır:

i) 64 kbit/sn (7 kHz) kodlayıcı:

- işitsel işareti 16 kHz'e örnekleddikten sonra 14 bit kullanarak birörnek sayısal işarete dönüştüren işitsel verici;

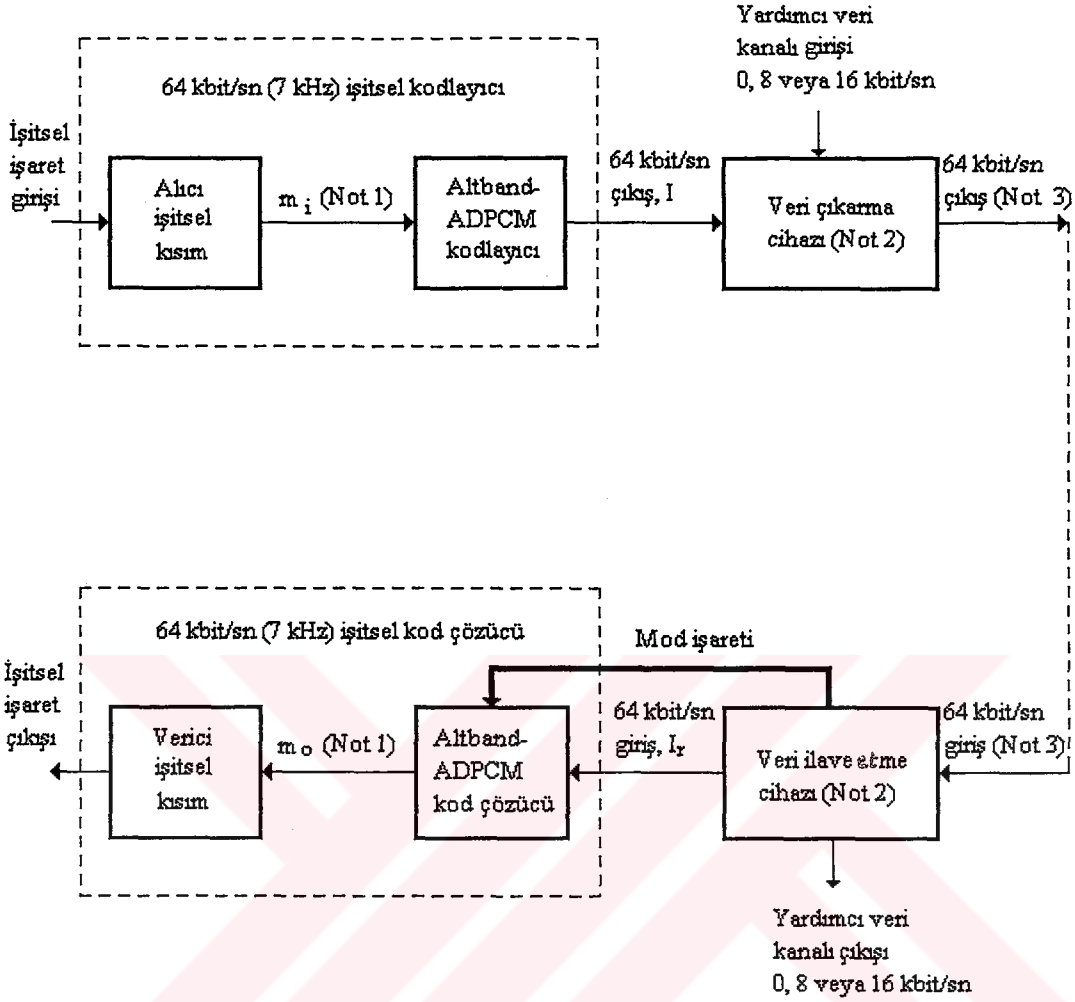
- bit hızını 64 kbit/sn'ye indirgeyen altband-ADPCM kodlayıcıdan oluşur.

ii) 64 kbit/sn (7 kHz) kod çözücü:

- kodlayıcının tersi gibi çalışan altband-ADPCM kod çözücü (çalışma moduna bağlı olarak 64, 56, 48 kbit/sn olabilen kod çözücünün girişindeki işitsel kodlama bit hızlarına dikkat ediniz);

- 16 kHz örneklemeyle 14 bit kullanarak, kodlanmış birörnek sayısal işareten işitsel işareti oluşturan işitsel alıcıdan oluşur.





Not 1 -  $m_i$  ve  $m_o$  14 bit ve 16 kHz örnekleme birörnek kodlanmış sayısal işaretlerdir.

Not 2 - Bu cihaz sadece 64 kbit/sn'li yardımcı veri kanalı uygulamaları için gereklidir.

Not 3 - İşitsel kodlama için 64, 56 veya 48 kbit/sn ve data için 0, 8, veya 16 kbit/sn içerir.

Not 4 - Çalışma moduna bağlı olan işitsel kodlama için 64, 56, veya 48 kbit/sn içeren 64 kbit/sn'lik işarettir.

### ŞEKİL 8.1

64 kbit/sn işitsel kodlayıcının basitleştirilmiş işlevsel blok diyagramı

Şekil 8.1'de verici çıkışıındaki ve alıcı girişindeki cihazlar yardımcı veri kanalı gerektiren uygulamalar içindir;

- verici çıkışıındaki veri ilave etme cihazının işlevi, gerektiğinde çalışma moduna bağlı olarak oktet başına düşen 1 veya 2 işitsel bitin yerine veri bitlerini koymaktır. Böylece 8 veya 16 kbit/sn'li yardımcı veri kanalı sağlanır.

- Alıcı uçtaki veri çıkarma cihazı ise mod kontrol stratejisine göre çalışma modunu belirler ve uygun bir şekilde veri bitlerini çıkarır.

## 8.2 İşitsel Kısımların İşlevsel Tanımı:

Şekil 8.2'de işitsel kısmın muhtemel yapısı gösterilmiştir. Mikrofon, ön kuvvetlendirici, güç anfisi ve kulaklık işitsel kısmın içeriğini belirtmek için basitçe gösterilmiştir.

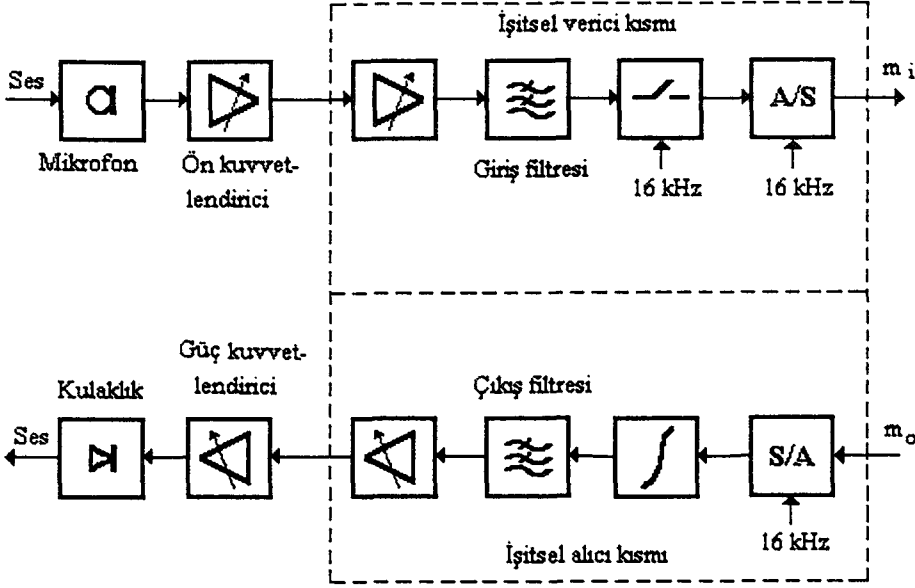
Alıcı ve vericideki işitsel kısımlar aşağıdaki birimleri içerir:

### i) verici:

- giriş seviyesini uyarlama cihazı,
- giriş filtresi,
- 16 kHz'de çalışan örnekleme cihazı,
- 16 kHz örneklemeli 14 bitli analog-birörnek sayısal dönüştürücü;

### ii) alıcı:

- 14 bit birörnek sayısal-analog dönüştürücü,
- yeniden oluşturma filtresi,
- çıkış filtresi,
- çıkış seviyesini ayarlama cihazı.



ŞEKİL 8.2

İşitsel kısımların muhtemel yapısı

### 8.3 Muhtemel Modlar Ve Veri İlave Etme Cihazı

Kod çözücünün girişine gelen değişik bit hızlarınının 3 temel modu Tablo 8.1'de gösterilmiştir.

64 kbit/sn işitsel kodlayıcı, çalışma modundan bağımsız olarak işitsel kodlama için sürekli 64 kbit/sn bit hızında çalışır. İşitsel kodlama algoritması, kodlayıcıya herhangi bir işaret göndermeden düşük altbandın en düşük anlamlı bir veya iki bitinin yerine yardımcı veri kanalı bitlerini yerleştirmek için 64 kbit/sn'den daha düşük bit hızlarını kullanabilir. Bununla birlikte verilen çalışma modunda işitsel randımanı maksimumda tutmak için, 64 kbit/sn işitsel kod çözücü, işitsel kodlama için mevcut olan bit hızlarını optimize etmiş olmalıdır. Bu nedenle altband-ADPCM kod çözücüsünde 3 ayrı bit hızı kullanılır ve yardımcı veri kanalı gerektiren uygulamalarda, kod çözücüde uygun çalışma modunu seçmek için bir mod işareti gönderilir. Şekil 8.1'den kod çözücünün

girişinin daima 64 kbit/sn olduğu görülmektedir. Fakat işitsel kodlama için çalışma moduna bağlı olarak 64, 56 veya 48 kbit/sn bit hızlarını içerir. Altband-ADPCM kod çözücünde kullanılan değişken, iletim süresince her bir oktet için değişebilir. Kod çözücüye çalışma moduyla ilgili işaret gönderilmezse mod 1'e karşı düşen değişken kullanılır.

**Tablo 8.1**

Çalışmanın temel muhtemel modları.

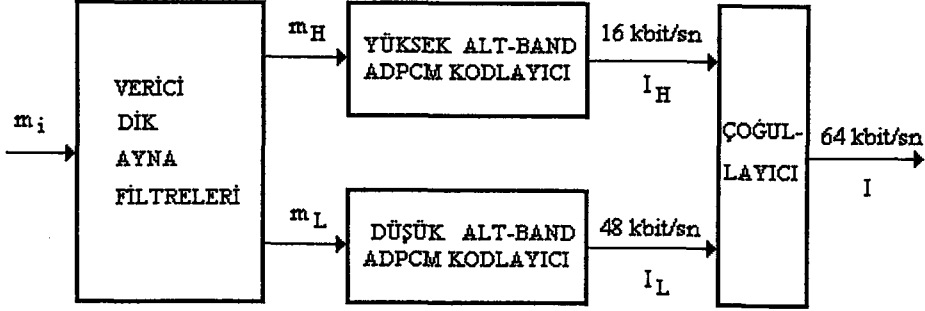
Mod	7 kHz işitsel kodlama için kullanılan bit hızı	yardımcı veri kanalı için kullanılan bit hızı
1	64 kbit/sn	0 kbit/sn
2	56 kbit/sn	8 kbit/sn
3	48 kbit/sn	16 kbit/sn

#### 8.4 Altband-ADPCM Kodlayıcının İşlevsel Tanımı

Şekil 8.3'de Altband-ADPCM kodlayıcının blok diyagramı gösterilmiştir. Her bir öbeğin işlevsel tanımı aşağıda verilen 8.4.1 ile 8.4.8 arasındaki paragraflarda yapılmıştır.

##### 8.4.1 Verici Dik Ayna Filtreleri (QMF)

Verici dik ayna filtreleri (QMF), 0-8000 Hz'lik frekans bandını, düşük altband (0-4000 Hz) ve yüksek altband (4000-8000 Hz) olmak üzere iki ayrı altbanda ayıran rekürsif olmayan filtrelerdir. Verici QMF'nin girişi  $m_i$ , vericinin işitsel kısmından gelir. Alçak ve yüksek altbandlar için olan çıkışlar ( $m_L$  ve  $m_H$ ) 8 kHz'de örneklenir.



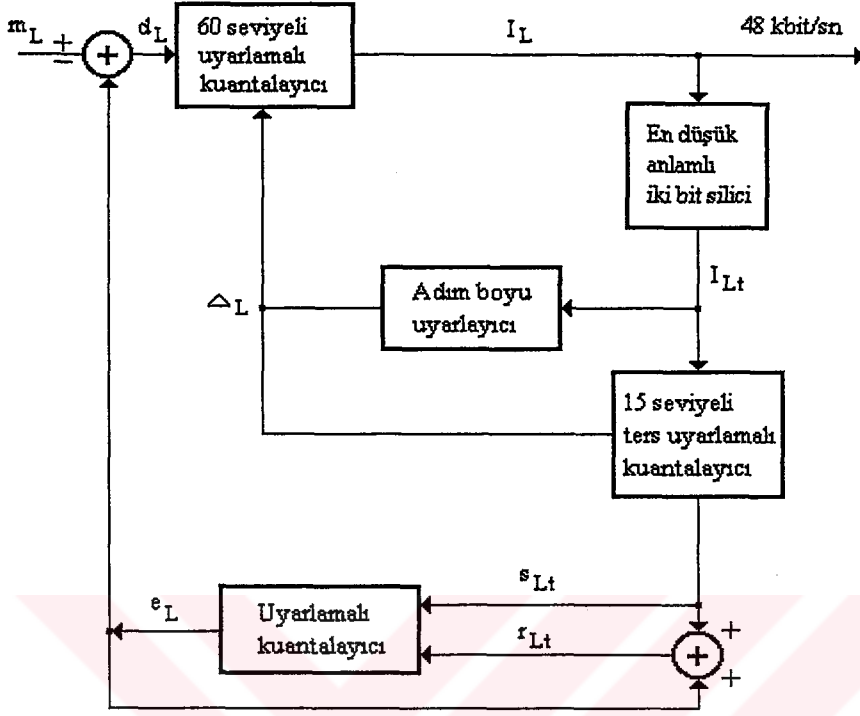
ŞEKİL 8.3

Altband-ADPCM kodlayıcının blok diyagramı

#### 8.4.2 Düşük Altband-ADPCM Kodlayıcı

Şekil 8.4 düşük Altband-ADPCM kodlayıcının blok diyagramıdır. Düşük altband giriş işareti  $m_L$ 'den tahmini olan  $e_L$  çıkarılarak  $d_L$  fark işareti üretilir. Uyarlamalı 60 seviyeli lineer olmayan kuantalayıcı, 48 kbit/sn  $I_L$  işaretini üretmek için fark işaretinin değerini, 6 basamaklı binary sayı olarak saptar.

Geri besleme öbeğinde,  $I_L$ 'nin en küçük anlamlı iki biti 4-bit  $I_{Lt}$  işaretini üretmek için silinir. Bu  $I_{Lt}$  işareti, kuantalayıcının ölçekleme çarpanını belirleyen adım boyu uyarlama öbeğinde kullanılır ve kuantalanmış fark işareti  $d_{Lt}$ 'yi üretmek için 15 seviyeli ters uyarlamalı kuantalayıcıya uygulanır. Tahmin işareti  $e_L$ , düşük altband giriş işaretinin yeniden üretilmiş versiyonu olan  $I_{Lt}$ 'yi üretmek için kuantalanmış fark işaretine eklenir. Hem yeniden oluşturulmuş hem de kuantalanmış fark işareti giriş işaretinin tahmini  $e_L$ 'yi üreten uyarlamalı tahmin edicide işlenir.



ŞEKİL 8.4

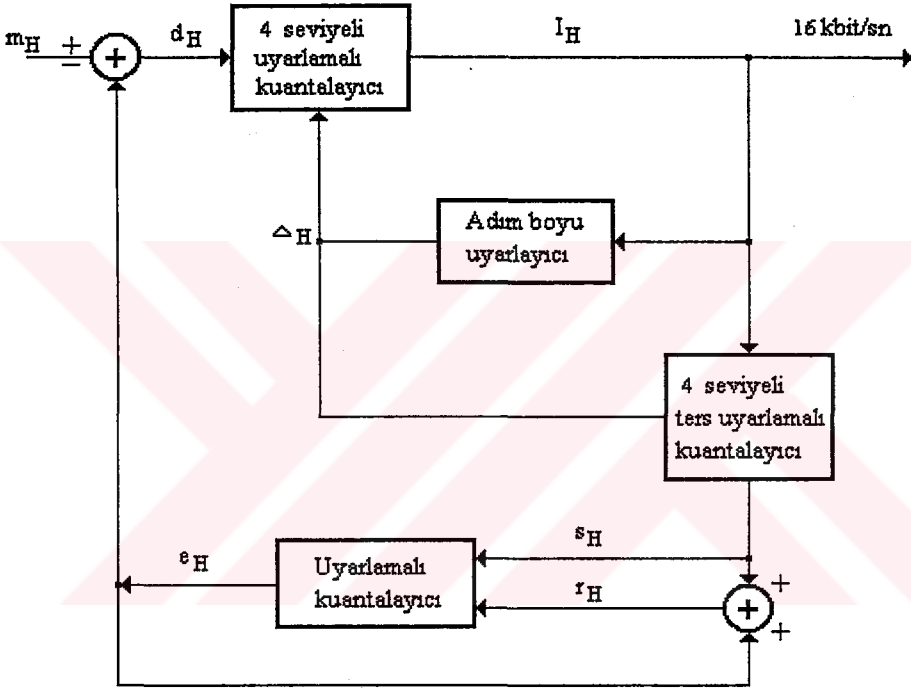
Düşük altband-ADPCM kodlayıcısının blok diyagramı

Düşük Altband-ADPCM kodlayıcısının ve kod çözücünün geribesleme öbeklerinde 6 bit yerine 4 bit ile çalışılması, en düşük anlamlı iki bite verinin ilave edilmesine olanak verir.

### 8.4.3 Yüksek Altband-ADPCM Kodlayıcı

Şekil 8.5 düşük altband kodlayıcısının yapısına benzer olan yüksek altband-ADPCM kodlayıcısının blok diyagramıdır. Yüksek altband giriş işareti  $m_H$ 'dan tahmin işareti  $e_H$  çıkarılarak fark işareti  $d_H$  elde edilir. Uyarlamalı lineer olmayan 4 seviyeli kuantalayıcı, 16 kbit/sn  $I_H$  işaretini üretmek için fark işaretinin değerini 2 basamaklı binary sayısı olarak belirler.

Ters uyarlamalı kuantalayıcı bu binary basamaklarından, kuantalanmış fark işareti  $d_H$ 'ı üretir. Tahmin işareti  $e_H$  yüksek altband giriş işaretinin yeniden üretilmiş versiyonu  $r_H$ 'ı üretmek için kuantalanmış fark işaretine eklenir. Yeniden oluşturulmuş  $r_H$  işaretiyle kuantalanmış fark işareti, giriş işaretinin tahminini üreten uyarlamalı tahmin edicide, işlenir. Böylece geribesleme döngüsü tamamlanır.



ŞEKİL 8.5

Yüksek altband-ADPCM kodlayıcısının blok diyagramı

#### 8.4.4 Çoğullayıcı

Şekil 8.3'de gösterilen çoğullayıcı alçak ve yüksek altband ADPCM kodlayıcılarından gelen  $I_L$  ve  $I_H$  işaretlerini, 64 kbit/sn'li bir kod sözcük işareti biçiminde birleştirmek için kullanılır.

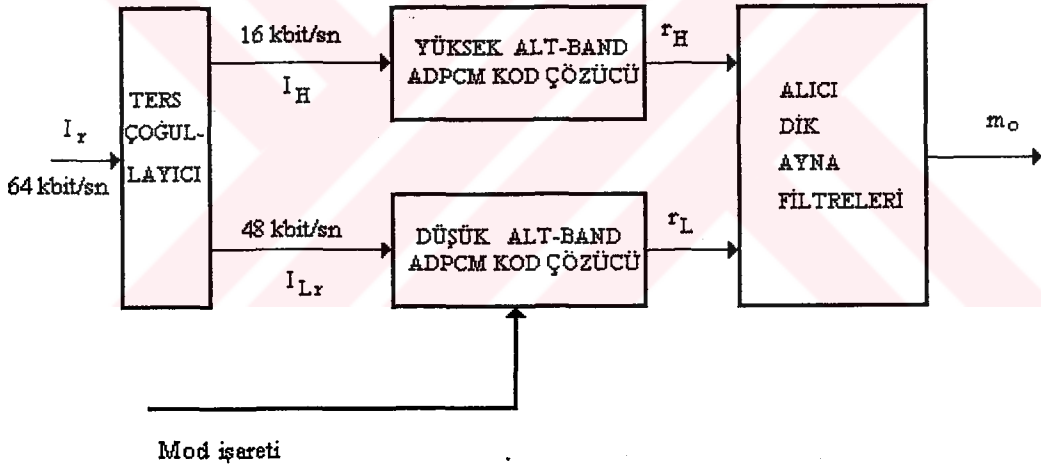
Çoğullamadan sonra elde edilen 8'li format çıkışı aşağıdaki gibidir:

$$I_{H1} \ I_{H2} \ I_{L1} \ I_{L2} \ I_{L3} \ I_{L4} \ I_{L5} \ I_{L6}$$

burada  $I_{H1}$  iletilen ilk bittir.  $I_{H1}$  ile  $I_{L1}$ , sırayla  $I_H$  ile  $I_L$ 'nin en ağırlıklı bitleri iken  $I_{H2}$  ile  $I_{L6}$ ,  $I_H$  ve  $I_L$ 'nin en küçük ağırlıklı bitleridir.

### 8.5 Altband-ADPCM Kod Çözücüsünün İşlevsel Tanımı

Şekil 8.6 altband-ADPCM kod çözücüsünün blok diyagramıdır. Her bir öbeğin işlevsel tanımı aşağıdaki paragraflarda yapılmıştır.



ŞEKİL 8.6

Altband-ADPCM kod çözücüsünün blok diyagramı

#### 8.5.1 Ters Çoğullayıcı

Ters çoğullayıcı 8'li formatlanmış  $I_r$  işaretini, düşük ve yüksek altband ADPCM kod sözcükleri biçiminde olan  $I_{Lr}$  ve  $I_H$  işaretlerine ayırır.





### 8.5.2 Düşük Altband-ADPCM Kod Çözücü

Şekil 8.7 düşük altband-ADPCM kod çözücüsünün blok diyagramıdır. Bu kod çözücü, çalışma moduna bağlı olarak üç farklı olasılığın herhangi birinde çalışabilir. Çalışma modu veri çıkarma cihazından gönderilen mod işaretine göre belirlenir.

Giriş işaretinden,  $e_L$  tahmin işaretini üretmenin yolu 8.4.2 paragrafında açıklanan düşük altband-ADPCM kodlayıcının geribesleme kısmındaki gibidir. Yeniden üretilmiş  $r_L$  işareti, çalışma moduna göre seçilen olası üç kuantalanmış fark işaretleri  $d_{L,6}$ ,  $d_{L,5}$  ve  $d_{L,4}$ 'den birinin, tahmin işaretine eklenmesi suretiyle elde edilir. Tablo 8.2 her bir mod için seçilen kuantalanmış fark işaretini, kullanılan ters uyarlamalı kuantalayıcıyı ve giriş kod sözcüğünden silinmiş en düşük anlamlı bitlerin sayısını gösterir.

### 8.5.3 Yüksek Altband-ADPCM Kod Çözücü

Şekil 8.8 yüksek altband-ADPCM kod çözücüsünün blok diyagramıdır. Bu kod çözücü 8.4.3 paragrafında açıklanan yüksek altband-ADPCM kodlayıcının geribesleme kısmına benzerdir. Çıkışı yeniden oluşturulmuş  $r_H$  işaretidir.

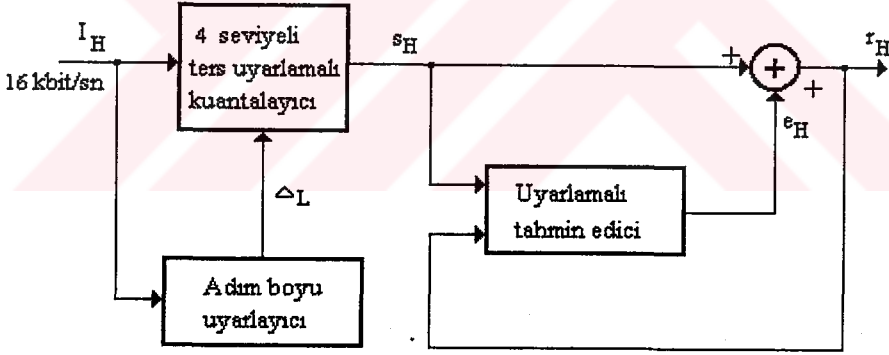
### 8.5.4 Alıcı Dik Ayna Filtresi (QMF)

Şekil 8.6'da gösterilen alıcı QMF, rekürsif olmayan iki lineer sayısal filtredir. Bu filtreler düşük ve yüksek altband-ADPCM kod çözücü çıkışlarını birleştirerek örnekleme frekansını 8 kHz'den 16 kHz'e çıkarır ve böylece alıcının işitsel kısmına uygun formda  $m_o$  çıkışını üretirler.

TABLO 8.2

Düşük altband-ADPCM kod çözücü değişkenleri

Alıcıya gelen çalışma modu	Seçilen kuantalanmış fark işareti	Kullanılan ters uyarlamalı kuantalayıcı	Giriş kod sözcüğü $I_{Lr}$ 'den silinen en küçük anlamlı bitlerin sayısı
Mod 1	$d_{L,6}$	60-seviye	0
Mod 2	$d_{L,5}$	30-seviye	1
Mod 3	$d_{L,4}$	15-seviye	2



ŞEKİL 8.8

Yüksek altband-ADPCM kod çözücünün blok diyagramı

## 9. KONUŞMA KODLAMA İÇİN ÇOK KUANTALAYICILI ADPCM

Bu bölümde az karmaşıklığa ve kısa işleme geçikmesine sahip bir konuşma kodlama algoritması anlatılacaktır. ADPCM kodlama yönteminde fark işaretini kuantalamak için 2 bit/örnek veya daha düşük bir değer kullanıldığı zaman randıman düşer. Bu düşüşün üstesinden gelmek için çok kuantalayıcılı ADPCM (ADPCM-MQ) algoritması Tomohiko Taniguchi [11] tarafından sunulmuştur. Bu algoritmada ilk önce, giriş işareti farklı kuantalama karakteristiklerine sahip ADPCM kodlayıcılarda işlenir. Bu kodlayıcılar birbirine paralel olarak konumlandırılmıştır. Daha sonra her bir çerçeve için minimum hata gücüne sahip optimum ADPCM kodlayıcı dinamik olarak seçilir. Optimum ADPCM kodlayıcının seçimi, giriş işaretiyle her biri yerel olarak kodlanmış işaretler arasındaki farkın (hata işareti) gücü değerlendirilerek yapılır. Bu değerlendirme işleminin nispeten kısa çerçeveler üzerinde yapılmasıyla işleme geçikmesi minimize edilebilir ve basit donanımlı kodlayıcı konfigürasyonu oluşturulabilir.

### 9.1 Çok Kuantalayıcılı ADPCM (ADPCM-MQ)

Genel olarak ADPCM lineer tahminli kodlama (LPC) temel algoritmasındaki gibi karmaşık bir sisteme sahip değildir ve uyarlama işlemi örnekten örneğe gerçekleştirilir. Böylece işleme geçikmesi kısa olur ve donanımının basit olması umulabilir. Fakat ADPCM 16 kbit/sn'de çalıştığı zaman yeterli nitelik sağlayamıyabilir. Bunun nedeni fark işaretini kuantalamak için sadece 2 bit/örnek veya daha azının tahsis edilmesidir. Bu problemi çözmek için Tomohiko Taniguchi ADPCM'in örnekten örneğe uyarlama işlemine çerçeveden çerçeveye geçiktirilmiş karar şemasını dahil etmiştir. Sonuç çok boyutlu kodlama biçimindedir. Bu algoritma çok kuantalayıcılı ADPCM (ADPCM-MQ) diye adlandırılır.

ADPCM-MQ kodlama algoritmasının blok diyagramı Şekil 9.1'de gösterilmiştir. Giriş işareti  $m(n)$  farklı kuantalama karakteristiklerine sahip ADPCM kodlayıcılarda kodlanır. Giriş işareti ile her biri yerel olarak kodlanmış işaretler arasındaki farklardan hata işaretlerinin ( $E_1$ 'den  $E_m$ 'e kadar) dizileri elde edilir. Daha sonra her bir hata işaretinin gücü hesaplanır ve minimum hata gücüne sahip ADPCM kodlayıcı (ADPCM-n) çerçeveden çerçeveye belirlenir. Bu değerlendirme işlemi 2.5 ms'lik her bir kısa çerçevede yapılır. Böylece işleme geçikmesi LPC analizi kullanan diğer kodlama algoritmalarındaki kadar büyük olmaz. ADPCM-MQ kodlama işlemi optimum kodlayıcı indeksine ( $n$ ) göre anahtarlamalı kuantalayıcılar içerir. Daha sonra kuantalanmış çıkış ( $I_n$ ) ve optimum ADPCM kodlayıcı indeksi ( $n$ ) çoğullanır ve iletilir. Kod çözücünde kod çözme işlemi, iletilen indekse ( $n$ ) göre tayin edilen ters kuantalayıcıyı seçme ile gerçekleştirilir.

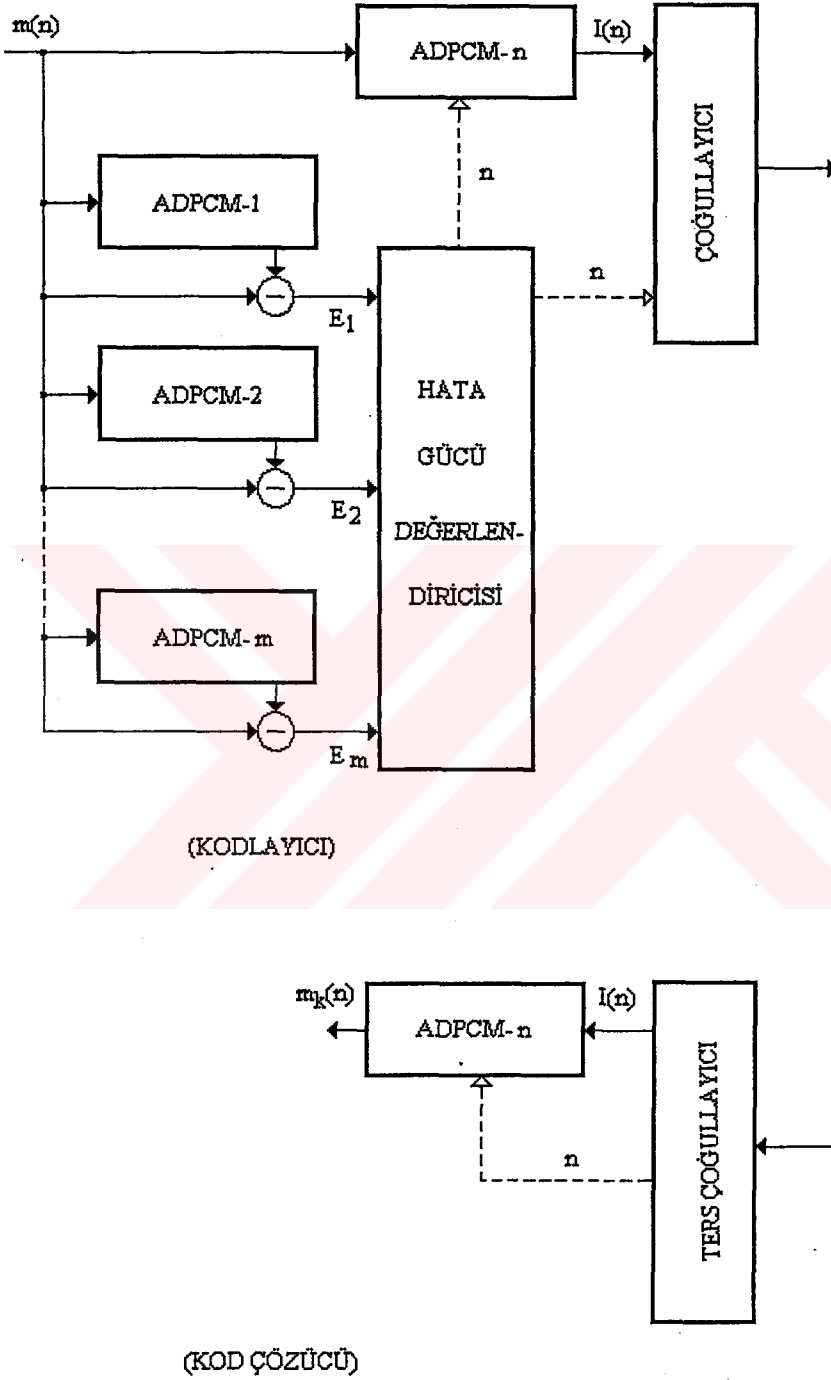
Alelade ADPCM yönteminde, uyarlamalı kuantalayıcının adım boyu (5.7) ifadesine göre kontrol edilir. ADPCM-MQ kodlama algoritmasında ise, her bir ADPCM kodlayıcı farklı uyarlama hızına göre karakterize edilen uyarlamalı kuantalayıcıya sahiptir ve aşağıdaki ifade de gösterildiği gibi birbirinden bağımsız olarak güncellenir:

$$\Delta_m(n+1) = \Delta_m(n)^\gamma R_m(I_m(n)) \quad (9.1)$$

burada

- $\Delta$ : Adım boyu
- $R$ :  $\Delta$ 'nin uyarlama hızı
- $I$ : Kuantalanmış çıkış
- $\gamma$ : Ağırlaştırma çarpanı

Yüksek uyarlama hızlı kuantalayıcı fazla yükleme distorsiyonunu önlemek için kullanılırken, düşük uyarlama hızlı kuantalayıcı kuantalama gürültüsünü bastırmak için kullanılır.



Şekil 9.1

ADPCM-MQ kodlayıcının blok diyagramı

Optimum ADPCM'i belirleyen değerlendirme fonksiyonu, giriş işareti  $m(n)$  ile herbirinin yerel olarak kodu çözülmüş işaretler ( $m_m(n)$ ) arasındaki hata işaretinin gücünü ( $P_m$ ) üretir:

$$P_m = \sum_{i=1}^k (E_m(i))^2$$

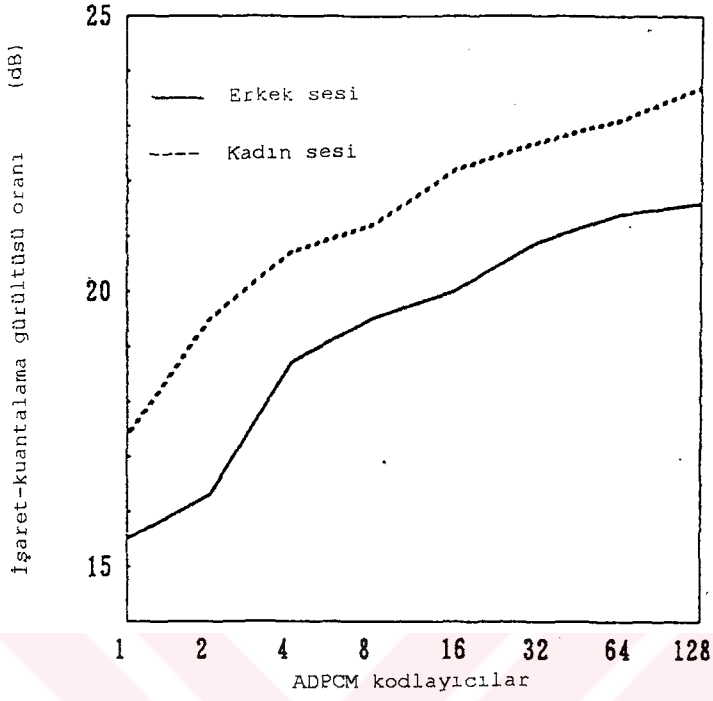
(9.2)

$$E_m = m(n) - m_m(n)$$

burada

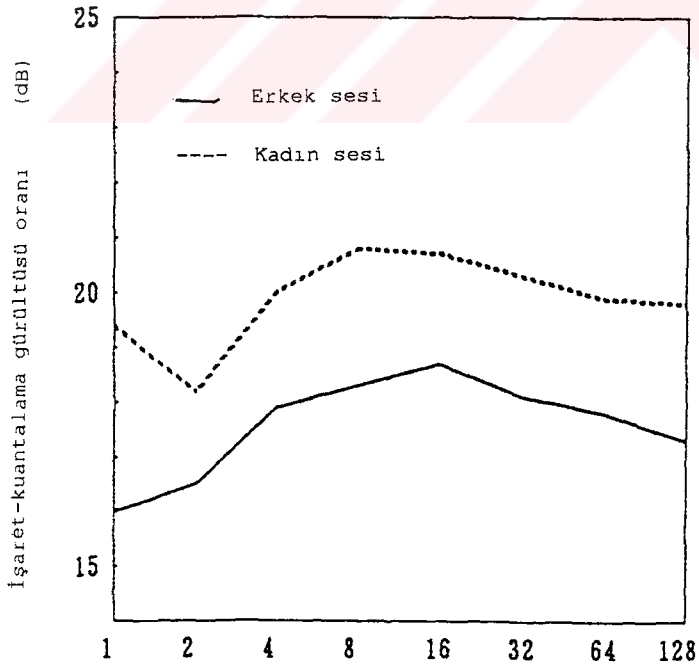
$P_m$ :	Hata işaretinin gücü
$E_m$ :	Hata işareti
$m_m(n)$ :	Yerel olarak kodlanmış işaret
$m$ :	ADPCM indeksi
$k$ :	Çerçeve uzunluğu

Şekil 9.2 ve 9.3, 2 bit/örnek ADPCM-MQ kodlama şemasına göre elde edilen SNR'nin simülasyon sonuçlarını gösterir. Şekil 9.2 ADPCM kodlayıcıların sayısı ile SNR arasındaki ilişkiyi gösterir. ADPCM kodlayıcıların sayısı artıkça optimizasyon gelişir ve en iyi karakteristik elde edilir. Kabul edilebilen konuşma niteliğiyle iletilen yan bilginin artışı arasındaki uzlaşma gözönünde tutulduğunda 4 ADPCM kodlayıcı kullanmak yeterlidir. Çerçeve uzunluğu uygun kodlama geçikmesi elde edilebilecek şekilde seçilir. Şekil 9.3 çerçeve uzunluğu ile 4 ADPCM kodlayıcının kullanıldığı durum arasındaki SNR ilişkisini gösterir. Optimum karakteristikler 8-16 örnekli çerçeve uzunluğunda elde edilir.



Şekil 9.2

ADPCM kodlayıcı sayısına göre SNR



Şekil 9.3

Çerçeve uzunluğuna göre SNR



**SONUÇ:**

ADPCM sistemindeki farksal işlem kuantalayıcıya uygulanacak işaret ölçüsünü indirger ve, kuantalayıcıda daha az bit kullanımına olanak tanır. Böylece standart PCM'e göre bit hızı düşer. Adım boyunu uyarlamayla birlikte konuşma genliği sıkıştırılır. Böylece kuantalayıcı düşük düzeyli işaretleri de, yüksek seviyeli işaretlerle hemen hemen aynı nitelikte işler.

ADPCM sistemiyle birlikte, 64 kbit/s PCM işareti 32 kbit/s'ye indirgenir. Ayrıca ADPCM, PCM'e göre 8 dB'lik bir iyileşme sağlar. Bu iyileşmenin 4 dB'i farksal kodlayıcıdan ve geri kalan 4 dB'i ise uyarlamalı kuantalayıcıdan kaynaklanır.

ADPCM'de elde edilen kod sözcükleri örnekten örneğe sadece biraz değişir ve kod sözcüklerinin değeri küçük olur. ADPCM kodlamanın bu özelliği konuşma işaretinin varlığını belirlemeyi kolaylaştırır. Buna dayanarak ADPCM ile kodlanmış konuşma işaretinin varlığını belirleme algoritmaları tasarlanabilir.

Altband kodlama 16 kbit/s bit hızında iyi nitelikli sayısal konuşma işaretinin elde edildiği bir tekniktir. Bu nitelik 24 kbit/s'li ADPCM ile sübjektif olarak karşılaştırılabilir.

64 kbit/s 7 kHz işitsel kodlama algoritması 4 kHz band genişlikli telefon sistemlerinden geniş bantlı sistemlere (7 kHz) geçişi sağlar. Kodlayıcının önüne yerleştirilen dik ayna filtreleri (QMF) kodlama niteliğini yükseltir.

ADPCM-MQ kodlayıcısı çoklu ADPCM kodlayıcıların kullanıldığı ve her bir çerçeve için optimum kodlayıcının seçildiği basit bir yapıya sahiptir. Bu kodlayıcıda ADPCM kodlayıcıların sayısı artıka optimizasyon gelişir ve en iyi

karakteristik elde edilir. Kabul edilebilen konuşma niteliğiyle iletilen yan bilginin artışı arasındaki uzlaşma göz önünde tutulduğunda 4 ADPCM kodlayıcı kullanmak yeterlidir. Çerçeve uzunluğu ile 4 ADPCM kodlayıcının kullanıldığı durum arasındaki SNR ilişkisinde optimum karakteristik 8-16 örnekli çerçeve uzunluğunda elde edilir.

Sonuç olarak, tümleşik devre teknolojisindeki gelişmelere bağlı olarak ADPCM'e dayanan birçok sistem tasarlanmıştır ve tasarlanmaya devam edilmektedir. Bu da ADPCM'in iyi nitelikli bir kodlama sistemi olduğunu gösterir.



**KAYNAKLAR**

- [1] Boddie J.R., Johnston J.D., McGonegal C.A., Upton J.W., Berkley D.A., Crochiere R.E., and Flanagan J.L., "Adaptive Differential Pulse Code Modulation Coding," B.S.T.J., Vol.60, No.7, pp.1546-1561, September 1981.
- [2] CCITT Recommendation G.721, "32 kbit/s Adaptive Differential Pulse Code Modulation," CCITT Red Book, 1984.
- [3] CCITT Recommendation G.722, "7 kHz Audio Coding Within 64 kbit/s," CCITT Red Book, 1986.
- [4] Chu P.C., "Quadrature Mirror Filter Design for an Arbitrary Number of Equal Bandwidth Channels" in IEEE Trans. ASSP-33, No.1, pp. 203-215, February 1985.
- [5] Crochiere R.E., Webber S.A., and Flanagan J.L., "Digital Coding of Speech in Sub-bands," B.S.T.J., Vol.55, pp.1069-1085, October 1976
- [6] Crochiere R.E., "A Mid-Rise/Mid-Tread Quantizer Switch for Improved Idle-Channel Performance in Adaptive Coders," B.S.T.J., Vol.57, No.8, pp.2953-2955, October 1978.
- [7] Crochiere R.E., "An Analysis of 16 kbit/s Sub-Band Coder Performance: Dynamic Range, Tandem Connections, and Channel Errors," B.S.T.J., Vol.57, No.8, pp. 2927-2954, October 1978.
- [8] Cummiskey P., Jayant N.S., and Flanagan J.L., "Adaptive Quantization in Differential PCM Coding of Speech," B.S.T.J., Vol.52, No.7, pp.1105-1118, September 1973.

[9] Flanagan J.L., Schroeder M.R., Atal B.S., Crochiere R.E., Jayant N.S., and Tribolet J.M., "Speech Coding" in IEEE Trans. Commun., vol.COM-27, NO.4, pp.710-737, April 1979.

[10] Goodman D.J. and Wilkinson R.M., "A Robust Adaptive Quantizer" in IEEE Trans. Commun., pp.1362-1365, November 1975.

[11] Janardhanan E., "Differential PCM Systems" in IEEE Trans. Commun., vol.COM-27, No.1, pp.82-93, January 1979.

[12] Jayant N.S. and Rosenberg A.E., "The Preference of Slope Overload to Granularity in the Delta Modulation of Speech," B.S.T.J., pp.3117-3125, March 1967.

[13] Jayant N.S., "Digital Coding of Speech Waveforms: PCM, DPCM and DM Quantizers," in IEEE Proc., Vol. 62, pp.611-632, May 1974.

[14] Maitre X., "7 kHz Audio Coding Within 64 kbit/s," in IEEE Journal on Selected Areas Commun., Vol.6, No.2, pp.283-297, February 1988.

[15] McDermott B., Scagliola C., and Goodman D., "Perceptual and Objective Evaluation of Speech Processed by Adaptive Differential PCM," B.S.T.J., Vol.57, No.5, pp.1597-1618, June 1978.

[16] Nann H., Straube H.M., and Villars C.P., "A Companded Coder for an Experimental PCM Terminal," B.S.T.J., pp.173-226, January 1962.

[17] O'Neal J.B. and Stroh R.W., "Differential PCM for Speech and Data Signals," IEEE Trans. Commun., Vol. COM-20, pp.900-912, October 1972.

[18] Protonotarios E.N., "Slope Overload Noise In Differential Pulse Code Modulation System," B.S.T.J., Vol.46, pp.2119-2161, November 1967.

[19] Rosenthal L.H., Schafer R.W., and Rabiner L.R., "An Algorithm for Locating the Beginning and End of an Utterance Using ADPCM Coded Speech," B.S.T.J., Vol.53, No.6, pp.1127-1135, July-August 1974.

[20] Scafer R.W., Jackson K., Dubnowski J.J., and Rabiner L.R., "Detecting the Presence of Speech Using ADPCM Coding," in IEEE Trans. Commun., pp.563-567, May 1976.

[21] Taniguchi T., Unagami S., Iseda K., and Tominaga S., "ADPCM with a Multiquantizer for Speech Coding," in IEEE Journal on Selected Areas Commun., vol.6, no.2, pp.410-423, February 1988.

**ÖZGEÇMİŞ**

AYDIN AKCEN

25 Ekim 1967 tarihinde İstanbul'da doğdu. İlkokulu Bağdat'ta Petite Ecole'de tamamladıktan sonra 1986 yılında Fenerbahçe Lisesinden mezun oldu. Daha sonra Yıldız Üniversitesi Kocaeli Mühendislik Fakültesini, Elektronik ve Haberleşme Mühendisi olarak tamamladı. Halen Yıldız Teknik Üniversitesinde Telekomünikasyon bilim dalında yüksek lisans öğrencisi.

