دانشگاه تربیت معلم سبزوار

گزارش پروژه درس :

ADAPTIVE FILTER

عنوان پروژه:

پیاده سازی نرم افزاری ADPCM مطابق توصیه G.726

استاد راهنما:

دکتر هادی صدوقی یزدی

دانشجویان:

سید مصطفی عبدالله پور MOSTAF_83@YAHOO.COM

مصطفی اسماعیلیان MOSTAFA_ESMAEELIAN@YAHOO.COM

WWW.ECA.IR

چکیده

گستردگی و همه گیر بودن شبکه های تلفنی باعث افزایش تقاضا برای ارسال غیر گفتاری بر روی خطوط تلفنی شده است . بنابراین با توجه به محدودیت ظرفیت کانالهای تلفنی ، فشرده سازی صحبت ، لازم و مطلوب است . معمولترین تکنیک برای این کار ADPCM است زیرا در آن مصالحه خوبی بین کیفیت ، نرخ بیت و پیچیدگی انجام گرفته است . استفاده از استاندارد CCITT برای ADPCM باعث سازگاری سیستم با شبکه های بین المللی می شود .

در این پروژه سیستم ADPCM پیشنهادی CCITT را برای نرخ بیت kbps 24(G.726) بررسی کرده و شبیه سازی کامپیوتری آن را انجام داده ایم عملکرد سیستم از لحاظ حفظ شکل موج ورودی نیز خوب است . در مجموع این سیستم کیفیت مخابراتی لازم را برای استفاده در شبکه های تلفنی دارا می باشد و بنابراین می توان بر مبنای آن ترانسکدر PCM 30 کانال به ADPCM 80 کانال را طراحی نمود . شبیه سازی کامپیوتری سیستم فوق در واقع ، گام اول برای ساخت نرم افزاری این ترانسکدر است .

فهرست مطالب

عنوان	صفحه
فصل اول : مقدمه	4
1-1) پیشگفتار	4
1-2) فشرده سازی و کدینگ صوت	5
1-2-1) پارامتر های اساسی در فشرده سازی صوت	5
1-2-2) روشهای فشرده سازی صوت	6
1-3) هدف انجام این پروژه	8
فصل دوم : کوانتیزاسیون	10
2-1) مقدمه	10
2-2) کوانتیزاسیون یکنواخت	11
2-3) کوانتیزاسیون لگاریتمی	13
2-4) کوانتیزاسیون وفقی	13
2-4-1) کوانتیزاسیون وفقی مستقیم (AQF)	14
2-4-2) کوانتیزاسیون وفقی معکوس (AQB)	15
2-4-3) کوانتیزاسیون وفقی مقاوم	16
فصل سوم : سیستمهای کدینگ بر پایه پیشگویی خطی	18
3-1) مقدمه	18
3-2) پیشگویی خطی مرتبه N	19
3-3) سیستم DPCM	20
3-4) پیشگویی وفقی	21

3-4-1) پیشگویی وفقی مستقیم (APF) 21
3-4-2) پیشگویی وفقی معکوس (APB) 22
3-5) مدل مناسب برای پیشگویی سیگنال صحبت 23
3-6) بررسی پیشگویی در سیستم ADPCM 25

فصل چهارم: بررسی سیستم ADPCM مطابق توصیه CCITT-G.726 26
4-1) مقدمه 26
4-2) بررسی فرستنده و گیرنده سیستم ADPCM 27
4-3) بررسی پیشگو یی کننده 30
4-4) بررسی کوانتیزه کننده 35

فصل اول : مقدمه

1-1) پیشگفتار

اصولا ارتباط از طریق گفتار از اصلی ترین و ابتدایی ترین قابلیتهای انسان به شمار می رود که از دیرباز بوسیله وی به کار گرفته شده است . هر چند که سایر حواس انسان اطلاعات زیادی را به وی می رسانند ولی در ارتباط دو طرفه یا چند طرفه گفتار نقش به مراتب ارزنده تری را ایفا می کند . در حقیقت قابلیتهای گفتاری انسان از ویژگی های مهم تمایز وی از حیوانات به شمار می رود . گفتار نه تنها حاوی اطلاعات و مفاهیم مورد نظر گوینده است بلکه در مورد شخصیت ، جنس و حتی حالات احساسی و ویژگی های درونی و اکتسابی او نیز اطلاعات مهمی به شنونده ارائه می دهد . عجین شدن زبان با فرهنگ ، بینش و طرز زندگی مردمان خود دلیلی بر این مدعاست . [1]

امروزه سیستم های ارتباطی روز به روز نقش مهمتری را در زندگی انسان بر عهده می گیرند و شبکه های مخابراتی جز لاینفک جوامع پیشرفته امروزی به شمار می روند . ارتباطات از حالت ساده شبکه تلفنی به یک شبکه عظیم از اطلاعات متنوع صوتی ، تصویری و داده در آمده است ، اما هنوز در این شبکه جز اصلی ارتباط همان گفتار است . جالب اینجاست که با توجه به پیشرفت های روز افزون پردازش صوت ، انتظار می رود که در آینده علاوه بر ارتباط انسانها با هم ، در ارتباط انسان با ماشینهای هوشمند نیز گفتار ، طبیعی ترین نقش را داشته باشد .

پردازش صوت در ابتدا فقط به صورت آنالوگ صورت می گرفت ، اما آنچه امروزه به عنوان پردازش صوت نامیده می شود همان پردازش دیجیتالی صوت است . امروزه اهمیت سیگنالهاو پردازش های دیجیتال بر کسی پوشیده نیست و هر روز بیشتر شاهد جایگزین شدن سیستمهای دیجیتال به جای سیستم های آنالوگ هستیم . مزایای سیستم های دیجیتال چنین است حساسیت سیگنال های دیجیتال به نویز انتقال کمتر است در نتیجه میزان خطای آن خیلی کمتر است . سیگنال دیجیتال را می توان دقیقا بازسازی کرد چون دو سطح 0 و 1 بیشتر ندارد . ذخیره سازی آن آسانتر است . صدا ، موسیقی ، داده ، تصویر و امکانات دیگر را می توان با هم مخلوط یا مالتی پلکس کرد . سرعت انتقال بیشتری را برای خطوط موجود فراهم می کند . رمز نگاری بوسیله آن آسانتر است و از همه مهمتر این است که قیمت سیستمهای دیجیتالی روز بروز کاهش می یابد . پردازش صوت نیز روز به روز در حال پیشرفت است و این پیشرفت های در سه زمینه مختلف صورت می گیرد . [4]

1- پیشرفت تئوری بویژه در حوزه آنالیز صحبت .

2- پیشرفت در تکنیکهای DSP که امکان استفاده از پیشرفت های تئوری را فراهم می کند .

3- پیشرفت سریع در پیاده سازی سخت افزاری با استفاده از تکنولوژی VLSI

مباحث مورد بررسی در پردازش گفتار را می توان به 4 زیر مجموعه تقسیم کرد[5] :

1- کدینگ و انتقال گفتار که ارتباط صوتی انسان با انسان را تحت پوشش قرار می دهد .

2- سنتز گفتار که با ارتباط ماشین با انسان سر و کار دارد .

3- تشخیص گفتار که به ارتباط انسان با ماشین مربوط می شود .

4- تشخیص گوینده به دو بخش شناسیی گوینده و تایید گوینده تقسیم می شود .

1-2) فشرده سازی و کدینگ صوت

با توجه به اهمیت روز افزون مخابرات در توسعه جوامع و این نکته که کانالهای تلفنی سهم عمده ای از کانالهای مخابراتی را تشکیل می دهند لزوم استفاده بهینه از کانالها و پهنای باند موجود مطرح می شود تا با پردازشهای مناسب بتوان استفاده بیشتری از امکانات موجود نموده ، راه را بری امکانات جدیدتر باز نمود . به همین علت مساله کدینگ یا فشرده سازی سیگنال صوتی مطرح می گردد .

اهداف اصلی کدینگ صوت به طور خلاصه چنین است :

1- ارسال اطلاعات با حجم کمتر و استفاده موثر تر از پهنای باند

2- ذخیره سازی و ضبط اطلاعات با حجم کمتر

3- استفاده بخشهای دیگر پردازش صوت از جمله تشخیص و سنتز از اطلاعات کد شده

4- رمز نگاری [1]

1-2-1) پارامتر های اساسی در فشرده سازی صوت

دو پارامتر اساسی در فشرده سازی صوت مطرح است : نرخ بیت[2] و کیفیت . نرخ بیت یا نرخ ارسال از رابطه $I = S.B$ بدست می آید که در آن S فرکانس نمونه برداری و B تعداد بیتهای کوانتیزاسیون است . برای سنجش کیفیت صوت نیز معیاری های متعددی وجود دارد که در جای خود مطرح خواهد شد . در فشرده سازی صوت می خواهیم برای یک نرخ بیت ثابت ، حداکثر کیفیت ممکن را بدست آوریم یا برای یک کیفیت ثابت و مطلوب به حداقل نرخ بیت ممکن برسیم . دو عامل فوق را نمی توان

---

[1] Encription
[2] Bit Rate

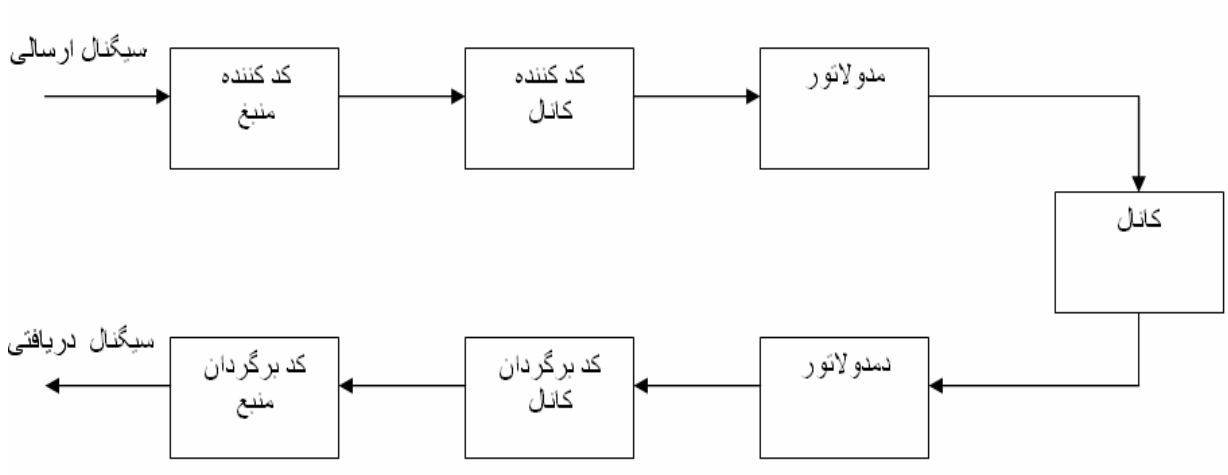
همزمان بهینه نمود و باید بین آنها مصالحه نمود . مثلا در ضبط و پخش موسیقی با کیفیت عالی SNR باید بین 90 تا 96 dB باشد و با توجه به رابطه تقریبی SNR = 6 B ، 16 بیت برای کوانتیزاسیون مورد نیاز می باشد از طرح دیگر برای موسیقی که حداکثر پهنای باند آن به علت محدودیت قوه شنوایی انسان 20kHz است نرخ نمونه برداری باید بیش از 40kHz باشد . استاندارد نمونه برداری در دیسکهای فشرده (CD) و کاستهای دیجیتالی (DAT) به ترتیب 44/1 و 48 کیلوهرتز است . استاندارد تعیین شده برای CD ها و DAT ها ، سیگنال صوتی دیجیتال را با بهترین کیفیت ارائه می کند در نتیجه برای داشتن کیفیت عالی در موسیقی نرخ بیت باید 705/6 kbps باشد (705/6 = 44/1 × 16) . اما ذخیره و انتقال این مقدار نرخ بیت ، بسیار مشکل است به عنوان مثال اگر فرض کنیم هر خط ارتباطی ISDN ظرفیت انتقال 64 kbps را داشته باشد برای انتقال این حجم از اطلاعات ، حداقل به 12 خط ارتباطی ISDN نیاز خواهیم داشت که از نظر اقتصادی قابل قبول و مقرون به صرفه نیست .پس استفاده از روشهای فشرده سازی کاملا ضروری است .

دو عامل دیگر را نیز باید در نظر گرفت و بین آنها نیز باید مصالح ای صورت بگیرد . این دو عامل پیچیدگی و تاخیر هستند . عامل پیچیدگی در حقیقت بیانگر حجم محاسباتی الگوریتم های بکار رفته برای فشرده سازی است و معمولا بر حسب تعداد عملیات حسابی و ظرفیت حافظه مورد نیاز بیان می شود . امروزه تکنولوژی پردازش سیگنالهای دیجیتال روز به روز گسترش بیشتری می یابدو در تراشه های کوچک میلیونها عمل در ثانیه انجام می گیرد . در فاصله 1990 تا 1995 ، سرعت چیپ ها از 25 به 250 mips رسیده و به موازات آن حجم حافظه زیاد شده و سرعت دسترسی به آنها نیز افزایش یافته است . این پیشرفت ها برای فشرده سازی سیگنال که نیازمند محاسبات زیادی است ، کاملا نوید بخش و حیاتی است و باعث می شود که عامل پیچیدگیروز به روز از اهمیت کمتری نسبت به سایر عوامل برخوردار شود . عامل دیگر تاخیر است که با پیچیدگی ارتباط دارد . تاخیر در کاربردهای یک طرفه چندان مهم نیست ولی در شبکه های مخابراتی دو طرفه باید آن را در نظر گرفت تا کمتر از حد مجاز شود.

1-2-2) روشهای فشرده سازی صوت

پارامتر های اساسی در فشرده سازی را مطرح کردیم و گفتیم که مصالحه بین آنها الزامی است بر همین مبنا روشهای مختلفی برای کدینگ یا فشرده سازی صوت ارائه شده است ، اما قبل از اینکه به آنها بپردازیم ذکر این نکته را لازم می دانم که در یک سیستم مخابراتی معمولا کدینگ در دو

مرحلــه صــورت مــی گیــرد : کــدینگ منبــع و کــدینگ کانــال کــه در شــکل (1-2) نشــان داده شده است .



شکل (1-2) مراحل کدینگ در یک سیستم مخابراتی

اطلاعــات ارســالی از طریــق یــک کانــال انتقــال ، غالبــا بواســطه وجــود نــویز دچــار اغتشــاش و خطــا می شــوند بــه همــین دلیــل معمــولا اطلاعــات را قبــل از ارســال بــا اســتفاده از کــدهای ویــژه ای طــوری کــد مــی کننــد کــه در گیرنــده بتــوان اطلاعــات درســت و نادرســت را از هــم متمــایز کــرد و خطاهــای ایجاد شده را تا حد امکان بر طرف نمود . به روش هایی که برای پیشگیری و خنثی کــردن اثــرات مخــرب نویز کانال انتقال بکار برده یم شود ، کدینگ کانال گفتــه مــی شــود کــه خــارج از بحــث ماســت و آنچــه در مورد کدینگ صوت گفته می شود زیر مجموعه ای از کدینگ منبع می باشد .

کد کننده های سیگنال صــوتی را مــی تــوان بــه دو گــروه عمــده تقســیم کــرد . اولــین گــروه مــی کوشــد تا شکل موج ورودی را حفظ کند یعنی سیگنال بازســازی شــده درگیرنــده اساســا هماننــد ســیگنال ورودی باشــد از ایــن رو بــه آنهــا کــد کننــده هــای شــکل مــوج گفتــه مــی شــود . دومــین گــروه مــی کوشــد تــا صــدایی شــبیه گفتــار اولیــه در گیرنــده فــراهم کنــد بــدون اینکــه ضــرورتی بــه حفــظ شــکل مــوج داشــته باشــد . روشــهای ایــن گــروه مبتنــی بــر آنــالیز و ســنتز صــحبت مــی باشــد و بــه آنهــا وکــدر [1] گفته می شــود . در عمــل مــرز دقیقــی بــرای ایــن دو گــروه وجــود نــدارد . دو راه بــرای کاســتن نــرخ بیــت در کد کننده هــای شــکل مــوج وجــود دارد . راه اول کاســتن اثــر مــزاحم نــویز کوانتیزاســیون بــا اســتفاده

[1] VOCODER = Voice Coder

از خواص ادراکی [1] گوش انسان است و راه دوم کم کردن مقدار اطلاعاتی که باید کد شود . این دو تکنیک غالبا با هم ترکیب می شوند اما در وکرد ها از مدل سازی سیستم تولید گفتار در انسان برای کاهش نرخ بیت استفاده می شود . صحبت را اساسا می توانبه دو مولفه تحریک و پاسخ ضربه محفظه صوتی تفکیک کرد . سیگنال تحریک بوسیله ششها تولید و توسط تارهای صوتی شکل داده می شود . محفظه صوتی یک سیستم متغیر با زمان است که مشخصات آن با اوضاع نسبی زبان ، دهان و بینی در هنگام ادای یک آوای خاص تعیین می شود و با عبور سیگنال تحریک از این سیستم متغیر با زمان ، سیگنال صحبت تولید می شود . تغییرات مشخصه سیگنال تحریک و محفظه صوتی معمولا کند است و از این رو می توان آن را در فواصل کوتاه 10 تا 40 میلی ثانیه تقریبا ثابت فرض کرد و بر مبنای همین فرض آنالیز و سنتز سیگنال صحبت در وکدرها صورت می گیرد بدین ترتیب که در آنالیز کننده گفتار ورودی به بخشهای کوتاه تقسیم و در هر بخش پارامتر های توصیف کننده دو مولفه تحریک و مشخصه محفظه صوتی استخراج ، کد و ارسال می شود . به همین دلیل به این گروه از کد کننده ها ، کد کننده های پارامتری نیز گفته می شود . مشخصه محفظه صوتی پوش طیف سیگنال را تعیین می کند و سیگنال تحریک مشخص کننده ساختار جزئی طیف است . بسته به نحوه آنالیز محفظه صوتی و چگونگی مدل سازی سیگنال تحریک ، سیستمهای وکدر گوناگونی پیاده سازی شده اند .

1-3) هدف انجام این پروژه

در کشور هایی مانند ایران که گسترش شبکه تلفنی را از طریق دیجیتالی کردن شبکه دنبال می کنند ، استفاده از تکنیکی که ظرفیت انتقال شبکه تلفنی دیجیتال (PCM) را افزایش دهد ، لازم و مطلوب است .

گستردگی و همه گیر بودن شبکه تلفنی باعث افزایش تقاضا برای ارسال اطلاعات غیر گفتاری (مانند داده ، تصویر ...) بر روی خطوط تلفنی شده است . بنابراین با توجه به محدودیت باند کانالهای تلفنی لازم است سیگنال صحبت را تا حد امکان فشرده کنیم و معمولترین تکنیک برای این کار استفاده از سیستم ADPCM است . به همین علت در سالهای اخیر پیشنهادهایی از سوی CCITT برای تبدیل سیگنال log-PCM با نرخ بیت 64 kbps به سیگنال ADPCM با نرخ بیت های کمتر ارائه شده است . در سال 1984 سیستم ADPCM با نرخ بیت 32 kbps ارائه شد (توصیه G . 721) و در سال 1988 براساس آن ترانسکدر 30 PCM کانالی به 60 ADPCM کانالی ارائه گردید (توصیه G . 726) که می توان

[1] Perceptual

براساس آن ترانسکدر PCM 30 کانالہ بہ ADPCM 80 کانالہ را طراحی مود و ھدف این پروژہ شبیہ سازی توصیہ G. 726 بعنوان گام اول برای ساخت نرم افزاری ترانسکدر PCM 30 کانالہ بہ ADPCM 80 کانالہ است .

البتہ سیستم ADPCM با نرخ بیت 24 kbps فقط برای انتقال سیگنال صحبت آن ھم با کیفیت مخابراتی مناسب است . با این حال کاربرد آن بویژہ در مواردی کہ با کمبود ظرفیت کانالھا مواجہ ھستیم بسیار مطلوب است .

فصل دوم : کوانتیزاسیون

2-1) مقدمه

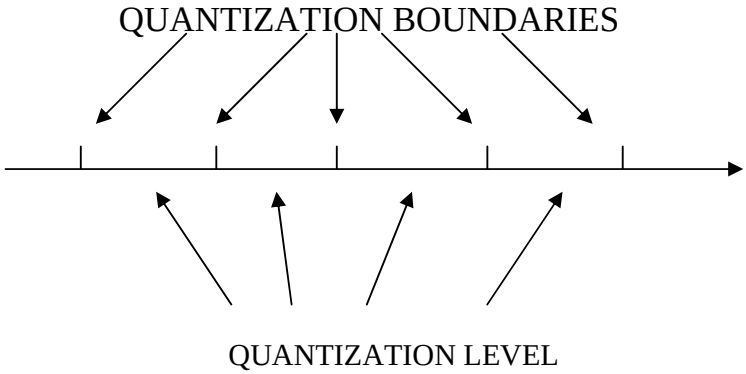
یک شکل موج آنالوگ هم از نظر زمان و هم از نظر دامنه پیوستگی دارد ، بنابراین تبدیل آن به یک سیگنال دیجیتال شامل دو مرحله است[4] :

1- نمونه برداری : که با انجام آن ، پیوستگی زمانی سیگنال از بین رفته و سیگنال حاصل را سیگنال زمان گسسته می گویند . نمونه برداری باید طوری انجام گیرد که بازسازی تقریبی سیگنال اولیه امکان داشته باشد و لازم است برای این کار از پدیده تداخل طیفی جلوگیری شود . قضیه نمونه برداری Nyquist ، فرکانس نمونه برداری مناسب را برای عدم تداخل اراده می کند . این قضیه بیان می کند که فرکانس نمونه برداری باید حداقل دو برابر پهنای باند (یا فرکانس قطع بالا در سیگنالهای باند میانی) سیگنال ورودی باشد [6].

2- کوانتیزه کردن : پس از نمونه برداری ، گسستگی زمانی ایجاد شده اما پیوستگی دامنه هنوز وجود دارد یعنی دامنه هنوز می تواند هر مقدار دلخواهی داشته باشد . با کوانتیزه کردن پیوستگی دامنه نیز از بین می رود . به سیگنال حاصل که هم از نظر زمان و هم از نظر دامنه ، گسسته است. سیگنال دیجیتال می گویند . کوانتیزه کردن در واقع تقریب مقدار یک نمونه است به مقادیر محدودی که مجموعه کدها را تشکیل می دهند . به عبارت دیگر پس از نمونه برداری سیگنال صحبت ، برای اینکه آن را با تعدادی از ارقام (بیتها) بصورت دیجیتالی بیان کنند ، آن را کوانتیزه می کنند . با افزایش و یا کاهش تعداد بیتهای کوانتیزاسیون نرخ بیت و کیفیت سیگنال صحبت افزایش و یا کاهش می یابد . در این فصل چند روش رایج در کوانتیزاسیون را بررسی خواهیم کرد .

2-2) کوانتیزاسیون یکنواخت

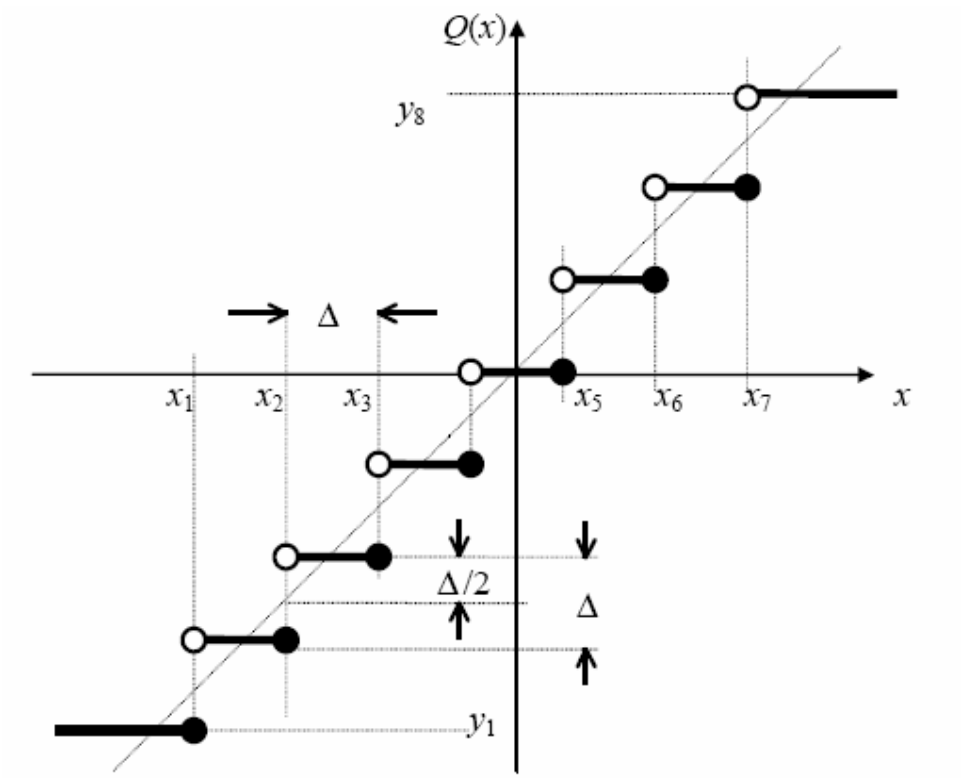
ساده ترین روش کدینگ شکل موج این است که آن را کوانتیزه نموده و سپس لغات کد را به عنوان بر چسب بر نمونه های کوانتیزه شده ، نسبت داده و این اعداد را به گیرنده ارسال نمود . این روش کدینگ که روشی پایه بوده و پیاده سازی آن بسیار سده است ، PCM نامیده می شود . چگونگی کار این کوانتیزه کننده در شکل (2-1) نشان داده شده است .



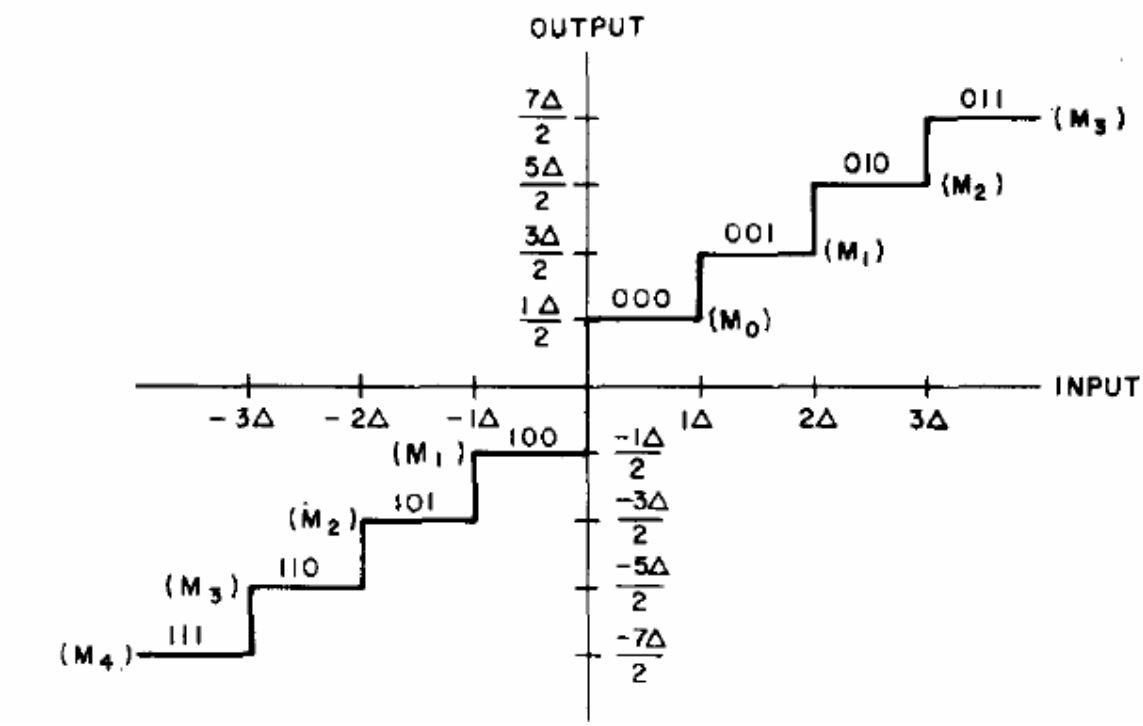
شکل (2-1) مرزها و سطوح کوانتیزاسیون در یک کوانتیزه کننده یکنواخت[2]

همانطور که مشاهده می شود تمام مقادیر بین s_1 و s_2 به s_2 کوانتیزه می شوند . s_1 و s_2 مرزهای کوانتیزاسیون و s_2 یکی از سطوح کوانتیزاسیون است . فاصله بین مرزهای کوانتیزاسیون را گام کوانتیزاسیون می گویند که معمولا با Δ نمایش می دهند . چون در اینجا Δ ثابت استاین روش کوانتیزاسیون یک نواخت نامیده می شود . شکل (2- 2) دو نوع کوانتیزاسیون یک نواخت 8 سطحی را نشان می دهند .

الف)



ب)



شکل (2 – 2) کوانتیزه کننده یکنواخت (الف) Midtread (ب) Midriser

در نوع اول که به آن Midtread گفته می شود یکی از سطوح کوانتیزاسیون صفر است اما در نوع دوم که Midriser نامیده می شود سطح صفر وجود ندارد . برای کد کردن این 8 سطح 3 بیت مورد نیزا است که چگونگی نسبت دادن لغات کد به سطوح دلخواه است اما معمولا از روش مکمل دو استفاده می شود تا در پردازشگر های دیجیتال قابل پردازش باشند . در شکل (2– 2– الف) کد های مکمل دو نشان داده شده اند .

از به منظور مقایسه سیستم های کدینگ شکل موج اغلب از نسبت سیگنال به نویز (SNR) استفاده می شود که طبق تعریف برابر است با نسبت واریانس سیگنال به واریانس نویز .

(2-3) کوانتیزاسیون لگاریتمی

مشکلی که در کوانتیزاسیون یکنواخت وجود دارد این است که چون سیگنال صحبت غیر ایستان است واریانس هر نقطه از آن با واریانس قطعه دیگر می تواند متفاوت باشد . اگر اندازه گام کوانتیزاسیون از روی سیگنالهای قوی محاسبه گردد این اندازه برای سیگنال های ضعیف خیلی بزرگ است و در نتیجه مقدار خطای کوانتیزاسیون برای سیگنالهای ضعیف بسیار زیاد خواهد شد و بر عکس اگر اندازه گام برای سیگنالهای ضعیف محاسبه کنیم ، برای سیگنالهای قوی دقت بیش از اندازه خواهد شد و تعداد بیتهای مورد نیاز خیلی زیاد خواهد شد .

برای حل این مشکل کوانتیزاسیون لگاریتمی پیشنهاد می شود که در آن ابتدا سیگنال صحبت بوسیله یک تبدیل لگاریتمی فشرده می شود تا دامنه های قوی به دامنه های ضعیف نزدیک شوند و سپس روی سیگنال فشرده شده کوانتیزاسیون یکنواخت انجام می گیرد . چون دامنه های سیگنال فشرده شده تفاوت فاحشی با هم ندارند اندازه گام برای تمام بخش های سیگنال مناسب است و بهبود قابل ملاحظه ای در کیفیت و SNR بوجود می آید .

در گیرنده با استفاده از تبدیل معکوس ، سیگنال فشرده شده ، باز می شود . به عمل فشرده سازی Compressing و به عمل باز نمودن Expanding و به این دو فرایند در مجموع کامپندینگ (Companding) گفته می شود .

کوانتیزه کننده لگاریتمی در عمل بسیار بهتر از کوانتیزه کننده یکنواخت کار می کند . مثلا یک کوانتیزه کننده لگاریتمی 8 بیتی ، کیفیتی معادل یک کوانتیزه کننده یکنواخت 13 بیتی دارد .

(2-4) کوانتیزاسیون وفقی

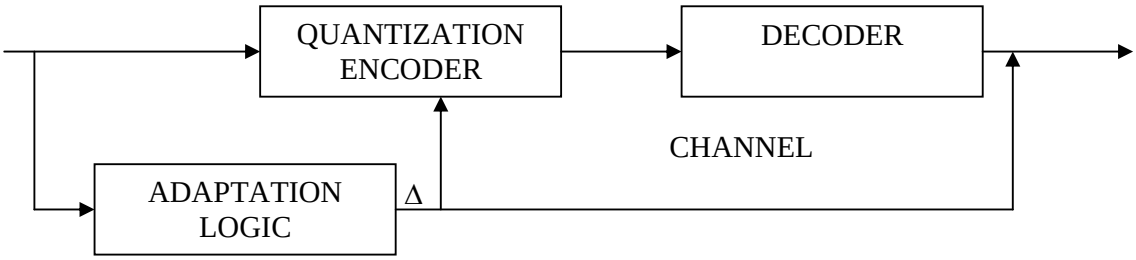
بعلت طبیعت غیر ایستان سیگنال صحبت آمارگان آن با زمان تغییر می کند و لذا برای بهبود عملکرد کوانتیزه کننده بهتر است که اندازه گام کوانتیزاسیون نیز متغیر بوده و مطابق با تغییرات آماری سیگنال محاسبه گردد . به این روش کوانتیزاسیون وفقی می گویند که به دو شکل انجام می گیرد :

کوانتیزاسیون وفقی مستقیم (AQF) که در آن اندازه گام کوانتیزاسیون از هر قطعه سیگنال به قطعه دیگر تغییر نموده و بهنگام می شود .

کوانتیزاسیون وفقی معکوس (AQB) که در آن اندازه گام کوانتیزاسیون از هر نمونه به نمونه ای دیگر از سیگنال بهنگام می شود .

(2-4-1) کوانتیزاسیون وفقی مستقیم (AQF)

شکل (2-3) یک کوانتیزه کننده وفقی مستقیم را نشان می دهد .



شکل (2-3) کوانتیزاسیون وفقی مستقیم[2]

در این روش اندازه گام کوانتیزاسیون در لحظه n از رابطه زیر بدست می آید .

$$\Delta(n) = \Delta.\sigma(n) \qquad (2\text{-}8)$$

که در آن Δ ، گام مناسب برای واریانس واحد است و o^2 (n) واریانس قطعه ای از سیگنال به طول M نمونه است که قرار است کوانتیزه شود و از رابطه زیر محاسبه می شود :

$$\sigma^2(n) = \frac{1}{M}\sum_{m=n}^{n+M-1} s^2(m) \qquad (2\text{-}9)$$

و یا به روش بازگشتی می توان از رابطه زیر آن را محاسبه کرد :

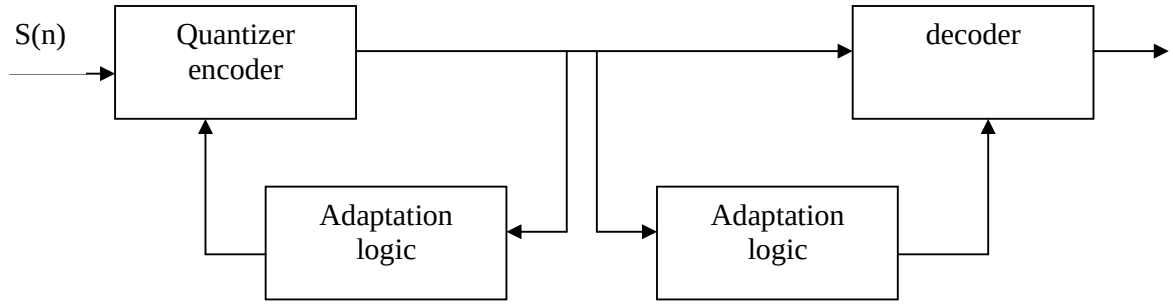
$$\sigma^2(n) = a\sigma^2(n-1) + s^2(n-1) \qquad (2\text{-}10)$$

که در آن $0 < a < 1$ است و نوعا مقدار 0/9 برای آن اختیار می شود . هر قدر a کوچکتر باشد ، کوانتیزه کننده سریعتر می تواند تغییرات سیگنال ورودی را دنبال کند .

همانطور که در شکل (2-3) مشاهده می شود ، برای بازسازی سیگنال از روی مقادیر کد شده ، لازم است که اندازه گام Δ (n) بعنوان اطلاعات جانبی به گیرنده فرستاده شود که این اطلاعات جانبی موجب افزایش نرخ بیت می شود . حسن این روش سهولت و دقت زیاد در محاسبه اندازه گام کوانتیزاسیون است .

(2-4-2) کوانتیزاسیون وفقی معکوس (AQB)

در شکل (2-4) کوانتیزه کننده وفقی معکوس نشان داده شده است :



شکل (2-4) کوانتیزاسیون وفقی معکوس[2]

در این روش اندازه گام از روی مقادیر کوانتیزه شده سیگنال که به صورت فیدبک برگشت داده می شوند ، محاسبه می شوند :

$$\Delta(n) = p.\Delta(n-1) \quad (2\text{-}11)$$

که در آن مقدار P فقط به $|c(n-1)|$ بستگی دارد یعنی اندازه کلمه کد قبلی . جدول (2-3) مقادیر نوعی P را برای یک کوانتیزه کننده به شکل Midriser نشان می دهد . مثلا برای کوانتیزه کننده 2 بیتی (4 سطحی) اگر C(n-1) متناظر با نزدیک ترین سطح کوانتیزاسیون به صفر (مثبت یا منفی) باشد P=0/6 و در غیر این صورت P = 2/2 اختیار می شود . با دقت در جدول (2-1) مشاهده می شود که برای کلمات کد کوچک که مربوط به قسمتهای ضعیف سیگنال هستند از P <1 استفاده می شود تا اندازه گام کوچک شده و در نتیجه ، عمل کوانتیزاسیون با دقت بالایی انجام گیرد برعکس برای کلمه کدهای بزرگ P>1 انتخاب می شود .

CODER	PCM	DPCM
QUANTIZER BITS	p	p
2	2.2 0.6	1.6 0.8
3	1.5 1.0 1.0 0.85	1.75 1.25 0.9 0.9
4	0.8 0.8 0.8 0.8	0.9 0.9 0.9 0.9
	2.4 2.0 1.6 1.2	2.4 2.0 1.6 1.2

جدول (2-1) ضرایب تنظیم گام در AQB

در این روش اندازه گام در گیرنده نیز محاسبه می شود پس دیگر نیازی به ارسال آنها بعنوان اطلاعات جانبی نیست و از این رو نرخ بیت آن از AQF کمتر است . ولی در عوض AQB پیچیده تر از AQF انجام می گیرد . همچنین در AQF انجام می گیرد همچنین در AQF چون نویز کوانتیزاسیون در تخمین گام بی تاثیر است ، تطبیق صحیح تر صورت می گیرد و SNR بیشتری حاصل می شود .

2-4-3) کوانتیزاسیون وفقی مقاوم

اگر کانال بدون خطا باشد ، کوانتیزه کننده های وفقی عملکرد مناسبی دارند ولی اگر کانال نویزی باشد ، به علت این که گامهای کوانتیزه کننده از روی مقادیر سابق محاسبه می گردند ، با مشکل انتشار خطا مواجه هستیم . پس باید کوانتیزه کنندهای را بکار برد که در برابر خطا مقاوم باشد . برای این کار باید سیستم ، اثر یک گام (Δ) را که با خطا مواجه شده است ، به تدریج فراموش کند . در نتیجه از ضریب نشتی استفاده می شود و رابطه بازگشتی (2-11) بدین شکل اصلاح می شود .

$$\Delta(n) = p.\Delta^{\beta}(n-1) \qquad (12\text{-}2)$$

که در آن $\beta < 1$ ضریب نشتی است و باعث می شود با گذشت زمان اثر گام های قبلی کمتر شود و بدین ترتیب انتشار خطا محدود گردد . البته بکار بردن ضریب نشتی عمل تطبیق را کمی کندتر می کند و باعث کاهش کیفیت در کانالهای بدون نویز می شود .

رابطه بازگشتی (2-12) را به شکل لگاریتمی نیز می توان بیان کرد :

$$\nabla(n) = \beta\nabla(n-1) + \log_2 p \qquad (13\text{-}2)$$

که در آن $\nabla(n) = \log\Delta(n)$ است . رابطه بازگشتی (2-13) اساس تنظیم ضرایب در کوانتیزه کننده وفقی سیستم ADPCM مطابق توصیهCCITT می باشد که در بخش (4-4) در مورد آن بیشتر توضیح خواهیم داد .

فصل سوم : سیستمهای کدینگ بر پایه پیشگویی خطی

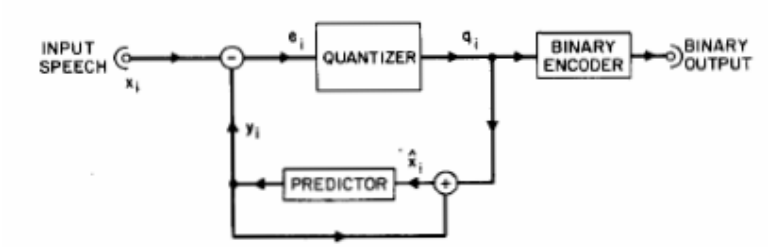
3-1) مقدمه

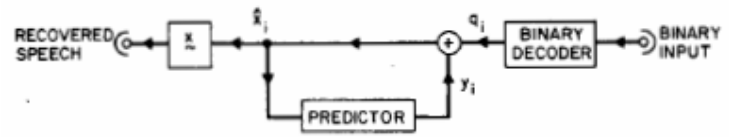
در مقدمه گزارش گفتیم که یکی از راههای فشرده سازی این است که میزان اطلاعاتی را که باید کد شود ، کاهش دهیم . بر همین مبنا سیستمهای کدینگ پیشگویی یا کدینگ تفاضلی ارائه شده اند که در آنها اضافات موجود در شکل موج سیگنال صحبت را در حوزه زمان کاهش می دهند بطوری که به ازای SNR مطلوب ، نرخ بیت کاهش یابد و در نتیجه کیفیت صحبت بازسازی شده در گیرنده ، در حد قابل قبول باقی بماند . با توجه به اینکه بین نمونه های شکل موج سیگنال صحبت همبستگی زیادی وجود دارد ، هر نمونه از شکل موج فوق را می توان از روی چند نمونه قبلی اش تخمین زد و پیشگویی نمود که این مقدار تخمین زده شده اختلاف ناچیزی با مقدار واقعی سیگنال دارد . چون خطای پیشگویی یعنی تفاضل بین نمونه های واقعی و مقادیر نمونه های پیشگویی شده ، محدوده تغییرات و متوسط انرژی کمتری نسبت به سیگنال اولیه صحبت دارد ، تعداد بیت کمتری نیز برای کوانتیزه شده نیاز دارد . پس بجای کوانتیزه نمودن سیگنال اصلی می توان خطای پیشگویی را کوانتیزه ، کد و ارسال نمود . این کار موجب کاهش قابل ملاحظه ای در نرخ بیت ارسالی می شود . این نوع سیستم را DPCM می نامند .

فرض کنید برای پیشگویی x(n)از نمونه های قبلی اش یعنی ...,x(x-2) ,x(x-1)استفاده شود. بهترین پیشگویی x(n) برای رسیدن به کمترین خطای مربعی (MMSE)همان امید ریاضی شرطی است :

$$\hat{X}(n) = E[X(n) \mid X(n-1), X(N-2), \ldots] \qquad (3\text{-}1)$$

اما این پیشگویی به دو دلیل غیر عملی است . نخست اینکه تابع چگالی احتمال (pdf) شرطی (یا مشترک) لازم برای این تخمین معمولا در دسترس نیست به همین دلیل سیستمهای کدینگ پیشگویی از پیشگویی خطی استفاده می کنند که در آن فقط اطلاعات آماری مرتبه دوم (مانند میانگین ، واریانس ، خود همبستگی و ...) مورد نیاز است . دوم اینکه ، برای آنکه فرستنده و گیرنده شکل موج ورودی را همزمان و مشابه بازسازی کنند مطلوب است (ولی لازم نیست)که پیشگویی از روی سیگنال بازسازی شده ، انجام گیرد که اثرات کوانتیزه کننده را در بردارد . بنا به دلایل گفته شده از ساختار شکل (3-1) استفاده می شود .





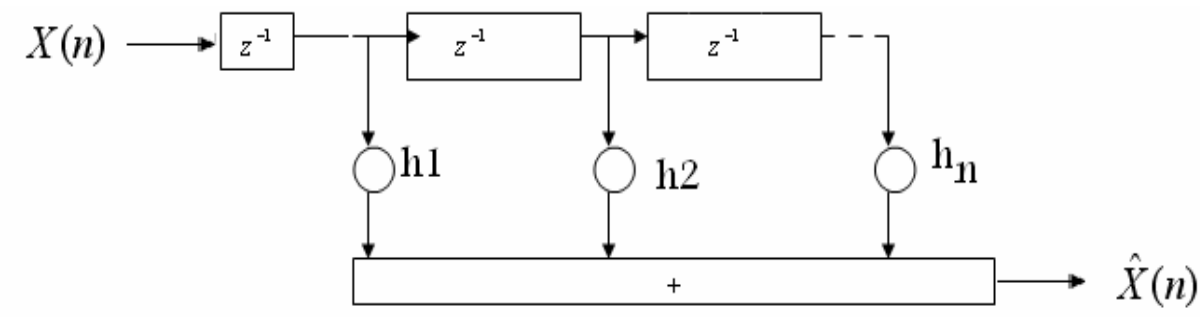
کل (3-1) بلوک دیاگرام فرستنده و گیرنده سیستم DPCM[8]

3-2) پیشگویی خطی مرتبه N

طبق تعریف پیشگویی خطی مرتبه N عبارت است از :

$$\hat{X}(n) = \sum_{J=1}^{N} h_j X(n-j) \qquad (3-2)$$

که در آن x(n) مقدار پیشگویی شده است که از ترکیب خطی N نمونه قبلی بدست آمده است. h_i ها ضرایب پیشگویی خطی نامیده می شوند . شکل (3-2) ساختار یک پیشگویی خطی مرتبه N را نشان می دهد :



شکل (3-2) ساختار یک پیشگویی خطی مرتبه N [8]

تابع تبدیل عبارت است از :

$$H(Z) = \sum_{K=1}^{N} h_k z^{-k} \qquad (3-3)$$

ضرایب پیشگویی hi باید طوری انتخاب شوند که فیلتر H(z) بهینه شود . بدین منظور باید MSE پیشگویی حداقل شود . یعنی :

$$\sigma^2{}_d = E[d^2(n)] = E[(x(n) - \hat{x}(n))^2] \qquad (3-4)$$

$$\frac{\partial \sigma^2{}_d}{\partial h_i} = E[2\{x(n) - \hat{x}(n)\} \frac{\partial}{\partial h_i}\{-\hat{x}(n)\}] = 0 \qquad (3-5)$$

$$\Rightarrow E[\{x(n)-\hat{x}_{opt}(n)\}x(n-i)]=0 \qquad i=1,2,3,...,N \quad (3-6)$$

یعنی کمترین خطای پیشگویی بر تمام اطلاعات بکار رفته برای پیشگویی عمود است که همان اصل تعامد است[1] با بسط (3-4) ضرایب بهینه بدست می آیند :

$$R_{xx}(k)=\sum_{j=1}^{N} h_{j,opt} R_{xx}(k-j); \qquad k=1,2\ldots,N \quad (3-7)$$

$$\Rightarrow r_{xx}=R_{xx}h_{opt} \qquad \Rightarrow h_{opt}=R^{-1}_{xx} r_{xx} \quad (3-8)$$

$$r^{T}_{xx}=\{R_{xx}(i)\} \qquad R_{xx}=\{R_{xx}(|i-j|)\} \qquad i,j=1,2,\ldots,N \quad (3-9)$$

این معادلات را معادلات نرمال ، معادلات پیشگویی Yule – Walker یا معادلات Wiener – Hopf می گویند . چون ماتریس R_{xx} یک ماتریس toeplitz است (یعنی علاوه بر متقارن بودن، عناصر روی قطر های آن برابرند) معکوس آن وجود دارد . بنابراین معادله (3-8) همیشه جواب داشته و پیشگویی کننده H(z) همواره پایدار است .

3-3) سیستم DPCM

در شکل (3-1) فرستنده و گیرنده یک سیستم DPCM نشان داده شده است . طبق تعریف پیشگویی خطی داریم :

$$\hat{x}(n)=\sum_{j=1}^{N} h_j \hat{x}(n-j) \quad (3-10)$$

که بیانگر یک پیشگویی تمام قطب است . در این سیستم ، پیشگویی کننده ثابت است یعنی ضرایب با زمان تغییر نمی کند و ورودی آن سیگنال بازسازی شده است ، در این سیستم مشاهده می شود که اگر انتقال اطلاعات روی کانال بدون خطا صورت گیرد :

$$y(n) = x(n) = x(n) + u(n) = x(n) + d(n) + q(n) = x(n) + q(n) \quad (3-11)$$

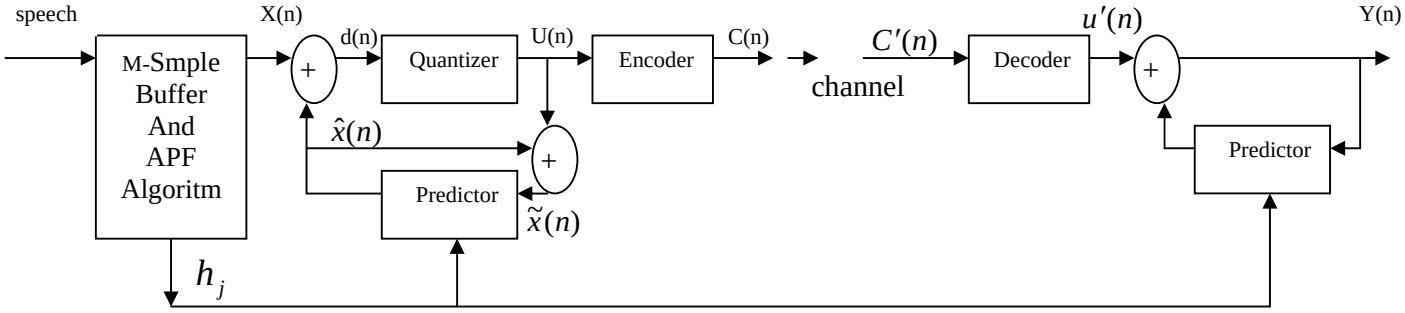
یعنی در صورت عدم وجود خطا در کانال انتقال و عملکرد یکسان پیشگوی فرستنده و گیرنده ، خطای بازسازی همان نویز کونتیزاسیون است . بعبارت دیگر خطای بازسازی در هر لحظه برابر با خطای کوانتیزاسیون در همان لحظه است پس خطا روی هم انباشته نمی شود و این از ویژگی های مهم سیستم حلقه بسته DPCM است .

3-4) پیشگویی وفقی

در سیستم DPCM معمولی ، پیشگویی کننده و کوانتیزه کننده هر دو ثابتند و هیچگونه تطبیقی در آن صورت نمی گیرد . بدیهی است که اگر کوانتیزه کننده و پیشگویی کننده وفقی باشند چون گامهای کوانتیزاسیون و ضرایب پیشگویی دائما به هنگام می شوند ، بهبود قابل ملاحظه ای در عملکرد سیستم بوجود خواهد آمد که این سیستم وفقی را ADPCM می نامند . اما تطبیق در پیشگویی کننده به دو صورت امکان دارد .

3-4-1) پیشگویی وفقی مستقیم (APF)

در روش اول که پیشگویی وفقی مستقیم APF نامیده می شود ، پیشگویی بر اساس نمونه های قبلی سیگنال اصلی انجام می گیرد و ضرایب پیشگویی و ضرایب خود همبستگی (که در محاسبه ضرایب پیشگویی مورد استفاده قرار می گیرند) برای سیگنال صحبت در بخشهای 20 تا 30 میلی ثانیه ای ، که در این فاصله زمانی سیگنال صحبت نسبتا ایستان است ، محاسبه می شوند . در شکل (3-3) ساختار اصلی سیستم ADPCM با پیشگویی تطبیقی مستقیم نشان داده شده است.

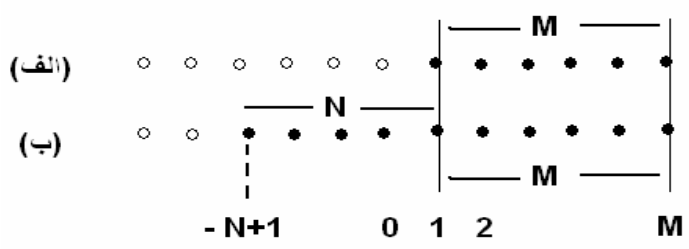


شکل (3-3) ساختار فرستنده و گیرنده در سیستم DPCM – APF[8]

همانطوری که در شکل نشان داده شده است ابتدا M نمونه (معدل 20 -30 میلی ثانیه) ذخیره شده و با استفاده از الگوریتم پیشگویی مستقیم ، ضرایب پیشگویی کننده مرتبه N محاسبه می شوند و پیشگویی خطی مرتبه N بر روی این نمونه ها انجام گرفته و خطای پیشگویی محاسبه و سپس کوانتیزه شده و از طریق کانال به طرف گیرنده ارسال می شود . این N ضریب پیشگویی بهینه لازم است بعنوان اطلاعات جانبی به گیرنده ارسال شود تا در آنجا با استفاده از خطای پیشگویی کوانتیزه شده و سیگنال پیشگویی شده ، سیگنال صحبت بازسازی شود . معیار برای بدست آوردن ضرایب پیشگویی

بهینه طبق معمول ، حداقل کردن قدرت خطای پیشگویی است و برای این کار درو روش وجود دارد : روش خود همبستگی و روش کوواریانس[9] .

در روش خود همبستگی برای محاسبه ضرایب بهینه از همان M نمونه موجود در هر بخش استفاده می شود . اما در روش کوواریانس N نمونه از بخش قبلی نیز مورد استفاده قرار می گیرد که در شکل (3-4) نشان داده شده است .



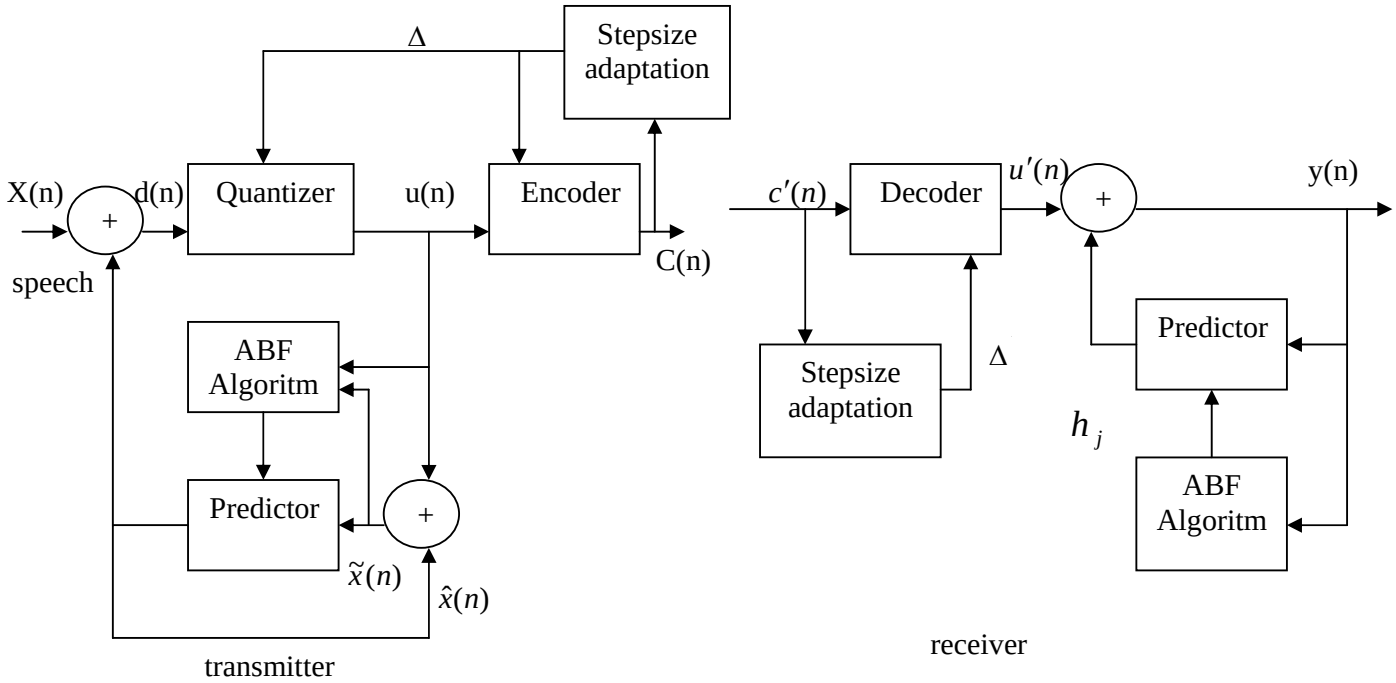
شکل (3-4) طول پنجره لازم در APF : (الف) روش خود همبستگی (ب) روش کوواریانس

در روش خود همبستگی خطای پیشگویی در لبه بخش ها زیاد است ، چون می خواهیم سیگنال را از روی نمونه هایی که خودمان صفر قرار داده ایم پیشگویی کنیم ، در نتیجه لازم است از پنجره هایی استفاده کنیم که اثر لبه ها را کاهش می دهند ، مثل پنجره همینگ . اما در روش کوواریانس به دلیل استفاده از N نمونه اضافی از بخش قبلی نیازی به استفاده از این نوع پنجره نیست. در عوض روش خود هم بستگی همیشه به جواب می رسد اما روش کوواریانس ممکن است به جواب نرسد . برای مقادیر بزرگ M اختلاف عملکرد این دو روش ناچیز است[8] .

4-4-2) پیشگویی وفقی معکوس (APB)

از مهمترین معایب پیشگویی مستقیم یکی تاخیر سیستم به دلیل ذخیره اطلاعات و دیگر ضرورت کانال اضافی برای ارسال اطلاعات جانبی می باشد . برای رفع نقیصه فوق باید به نموی ضرایب بهینه را از مقادیر کوانتیزه شده خروجی استخراج کنیم که این روش را APB می گویند . در این روش پیشگویی از روی نمونه های بازسازی شده انجام می گیرد و ضرایب پیشگویی از هر نمونه به نمونه دیگر بهنگام [1] می شوند . روش تنظیم ضرایب در APB معمولا براساس الگوریتم بازگشتی LMS است که قبلا آن را تشریح کرده ایم . در گیرنده نیز با همان الگوریتم بازگشتی ضرایب پیشگویی محاسبه می شوند بنابراین نیازی به ارسال ضرایب بعنوان اطلاعات جانبی نیست . در شکل (3-6) یک سیستم ADPCM با APB و AQB نمایش داده شده است . ساختار بکار رفته در توصیه CCITT نیز بر همین مبناست که موضوع مورد بحث در فصل بعد می باشد .

[1] Update



شکل (3-6) ساختار فرستنده و گیرنده یک سیستم DPCM– AQB – APB [10]

مقایسه APF و APB : برای یک مرتبه پیشگویی ثابت ، G_p در APB فقط 1dB کمتر از APF است که این مقدار نیز تقریبا با اطلاعات جانبی ارسال شده در APF خنثی می شود در نتیجه برای یک نرخ بیت ثابت ، عملکرد این دو خیلی به هم نزدیک است اما با استفاده از کوانتیزاسیون برداری می توان حجم اطلاعات جانبی را در APF کاهش داد . در نتیجه عملکرد APF در نرخ بیت های پایین بهتر از APB است زیرا در APB بهره پیشگویی را اثرات کوانتیزاسیون محدود می کند . معمولا همراه با APB از AQB استفاده می شود در نتیجه برای عملکرد مناسب در نرخ بیت پایین لازم است که اولا کوانتایزرهای مقاوم مورد استفاده قرار گیرند ثانیا تعداد ضرایب پیشگویی کننده حدود 4 تا 6 باشد از حداکثر بهره پیشگویی استفاده نگردیده است[8] . این نکته مهم در سیستم پیشنهادی CCITT رعایت شده است .

3- 5) مدل مناسب برای پیشگویی سیگنال صحبت

پیشگویی کننده از دو جهت قابل تقسیم بندی است . اول از نظر فرآیند تنظیم ضرایب است که آن را بیان کردیم و دوم از نظر نوع فیلتر بکار رفته برای پیشگویی است (تمام قطب – تمام صفر – قطب و صفر) که در واقع نوع سیگنال بکار رفته در تخمین را معین می کند و در این بخش به آن می پردازیم .

3 مدل معروف برای سیستمهای سیگنال گسسته زمان وجود دارد :

مدل AR : در این مدل خروجی سیستم در هر زمان برابر است با مجموع تعداد محدودی از مقادیر گذشته آن به اضافه نویز سفید . تابع تبدیل این سیستم ، یک فیلتر تمام قطب است و در نتیجه IIR و دارای حافظه طولانی است .

مدل MA : در این مدل ورودی نویز سفید و خروجی ترکیب خطی از مقادیر گذشته آن است . تابع تبدیل آن یک فیلتر تمام صفر است در نتیجه FIR و دارای حافظه کوتاه است . مدل MA چون از ترکیب خطی مقادیر نویز ساخته می شود ، مدل خوبی برای بخش ای بی واک سیگنال صحبت ، که رفتاری شبه نویز دارند ، است .

مدل ARMA : ترکیبی از دو مدل پیشین است یعنی یک فیلتر قطب و صفر . در نتیجه IIR است اما با حافظه محدود تر .

یکی از مهمترین پارامتر هایی که برای انتخاب این سه مدل در پیشگویی صحبت باید در نظر گرفت ، حافظه پیشگویی کننده است . برای بهبود عملکرد پیشگویی کننده باید از تعداد نمونه های بیشتر یا به عبارت دیگر حافظه طولانی استفاده کرد بدیهی است در این صورت پیشگویی دقیق تر و خطای پیشگویی کمتر خواهد بود ولی در صورت بروز خطای انتقال در کانال ، حافظه طولانی باعث انتشار خطا خواهد شد بطوریکه بروز یک خطای انتقال باعث ایجاد خطا در تعداد زیادی از نمونه ای سیگنال بازسازی شد در گیرنده می شود . از این رو مدل AR که دارای حافظه طولانی است برای کانالهای با خطای زیاد مناسب نیست . از طرف دیگر اگر کانال بدون خطا باشد استفاده از مدل MA سبب افزایش خطای پیشگویی می شود زیرا این مدل دارای حافظه کوتاه مدت است و از نمونه های کمتری برای پیشگویی بهره می برد . بنابراین باید مصالحه ای بین بهبود عملکرد پیشگویی کننده در کانال بدون خطا که حافظه طولانی را می طلبد و اثرات زیان آور خطا که حافظه کوتاه مدت را می طلبد ، انجام دهیم . اما پیشگویی کنند قطب و صفر اگرچه IIR است ولی با کاهش تعداد قطبها و افزایش صفر ها می توان طول پاسخ ضربه را کنترل کرد و ترکیبی از دو مدل فوق (ARMA) قادر است هر دو بخش واکدار و بی واک سیگنار صحبت را بخوبی مدل سازی کند . به همین دلیل در سیستمهای کدینگ پیشگویی از پیشگویی کننده قطب و صفر استفاده می شود . از جمله در سیستم پیشنهادی CCITT از پیشگویی کننده قطب و صفر استفاده خواهیم کرد .

6-4) بررسی پیشگویی در سیستم ADPCM

برای اینکه سیستم ADPCM دارای عملکرد مناسب باشد ، باید دو خاصیت زیر را داشته باشد :

1- پیشگویی خوب که به معنای خطای پیشگویی کم است .

2- قدرت تنظیم یا ردیابی که به معنای توانایی الگوریتم در سنکرون کردن فرستنده و گیرنده است . یعنی هر گاه بعلت بروز خطا در کانال ، فرستنده و گیرنده از سنکرون خارج شدند ، گیرنده بتواند سریعا خود را دوباره با فرستنده سنکرون کند تا در گیرنده سیگنال بازسازی شده خوبی داشته باشیم .

اما دو خاصیت فوق عکس همدیگرند . در حقیقت پیشگویی کامل زمانی است که سیگنال خطا ، نویز سفید باشد ولی از طرف دیگر مقداری اطلاعات درباره پارامترهای پیشگویی کننده و کوانتایزر در سیگنال خطا جاداده می شود در نتیجه نباید سفید باشد . بنابراین پیشگویی و تنظیم بهینه بطور همزمان محقق نمی شود و باید مصالحه ای بین آنها انجام گیرد .

رسیدن به خاصیت اول با الگوریتم هایی از خانواده LMS که در بخش های قبل بررسی شده امکان پذیر است اما دستیابی به خاصیت دوم مشکل است زیرا اولا پارامتر های کوانتیزه کننده و پیشگویی کننده مستقیما به گیرنده ارسال نمی شود و ثانیا از یک پیشگویی کننده ARMA که مسائل پایداری را به همراه دارد ، استفاده می شود (بنابر عللی که بیان کردیم) . اما نشان داده می شود که تنظیم گیرنده نسبت به فرستنده حاصل می شود فقط اگر پایداری یکنواخت قسمت AR تامین شود . پایداری یکنواخت به مشخصه وفقی الگوریتم مربوط است و هنگامی حاصل می شود که قطبهای پیشگویی کننده در فاصله ای معین از مرز در داخل دایره واحد باقی بمانند . در سیستم پیشنهادی CCITT این ویژگی حاصل شده است زیرا قسمت AR فیلتر از مرتبه 2 است و تست پایداری آنها آسان است . درباره سیستم پیشنهادی CCITT در فصل بعد مفصل تر بحث خواهیم کرد .

فصل چهارم : بررسی سیستم ADPCM
مطابق توصیه CCITT- G .726[11]

4-1) مقدمه

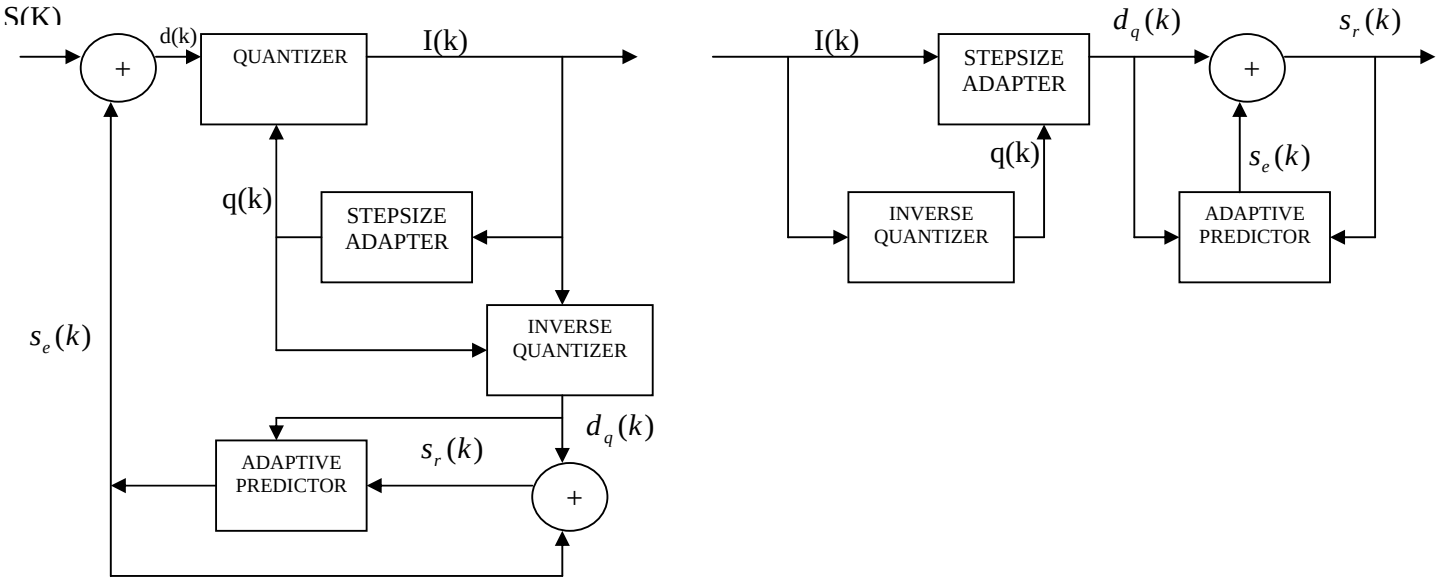
کمیته مشورتی بین المللی برای تلگراف و تلفن (CCITT) یک سازمان معتبر جهانی برای استاندارد کردن تکنیکها و ابزارهای مخابراتی است تا امکان اتصال شبکه های مخابراتی کشورهای مختلف برای ارتباط بین المللی بوجود آید . استانداردهای CCITT برای شبکه های ملی نیز مفید است و امروزه بسیاری از کشورها از آنها ، حتی در شبکه های محلی ، استفاده می کنند[13,5] .

استاندارد G .726 که در آخرین اجلاس CCITT در سال 1992 ارائه شده است ، سیستم ADPCM را با نرخ بیت های 40 ، 32 ، 24 و 16 کیلوبیت در ثانیه (kbps) معرفی کرده است . در نرخ بیت 40 و 32 kbps علاوه بر سیگنال صحبت قادر به ارسال داده و سیگنال تک فرکانس هستیم اما در نرخ بیت 24 و 16 kbps فقط ارسال سیگنال صحبت توصیه شده است . سیستم پیشنهادی CCITT یک سیستم APB – AQB –DPCM است که ساختار آن را در شکل (3-6) قبلا نشان داده ایم و در آن بنابر دلایلی که در بخش 3-5 بیان کردیم از یک پیشگویی کننده قطب و صفر استفاده شده است . کوانتیزه کننده هم نوعی کوانتیزه کننده وفقی مقاوم است که در بخش 2-4-3 درباره آن بحث کردیم . برای تنظیم ضرایب فیلتر های پیشگویی از الگوریتم اصلاح شده LMS استفاده شده است .

در این فصل ابتدا بلوک دیاگرام فرستنده و گیرنده پیشنهادی CCITT را نمایش داده و اجزا آن را شرح می دهیم و سپس در مورد اصول کار دو بلوک اساسی یعنی پیشگویی کننده و کوانتیزه کننده مفصل تر بحث خواهیم کرد .

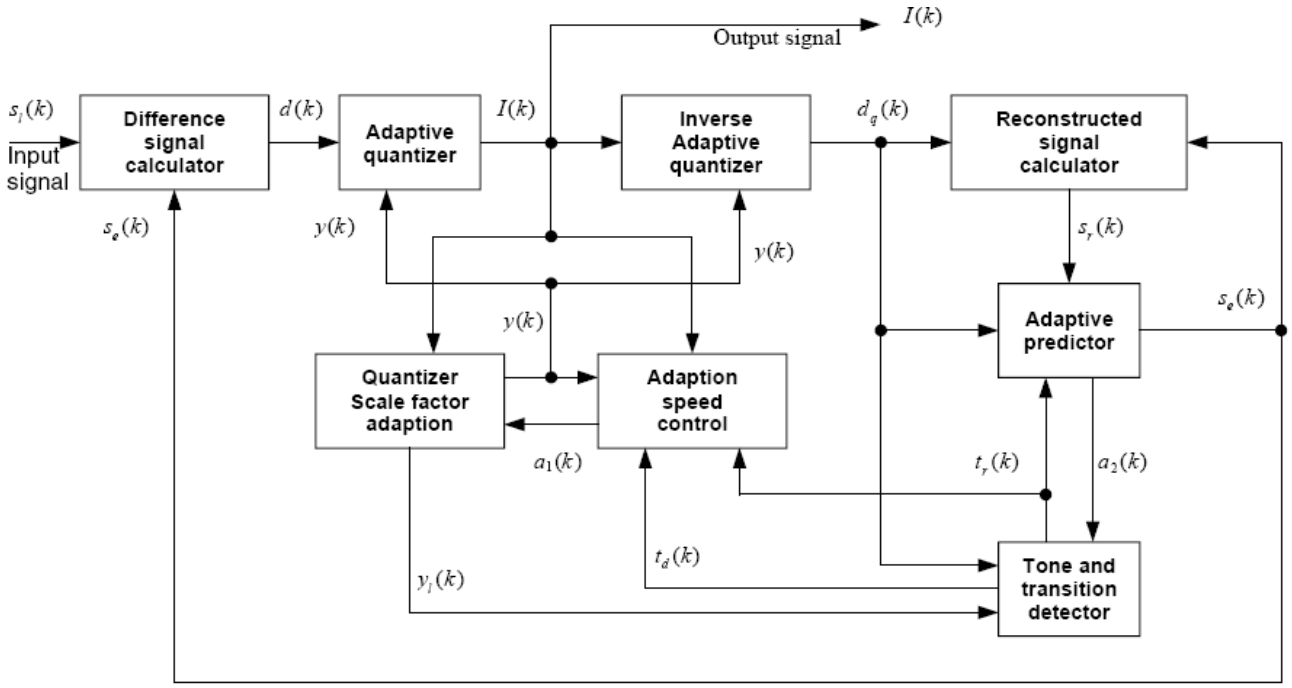
4-2) بررسی فرستنده و گیرنده سیستم ADPCM

در شکل (4-1) بلوک دیاگرام ساده فرستنده و گیرنده ADPCM نشان داده شده است و در شکل های (4-2) و (4-3) ساختار دقیق تر آنها قابل مشاهده است .

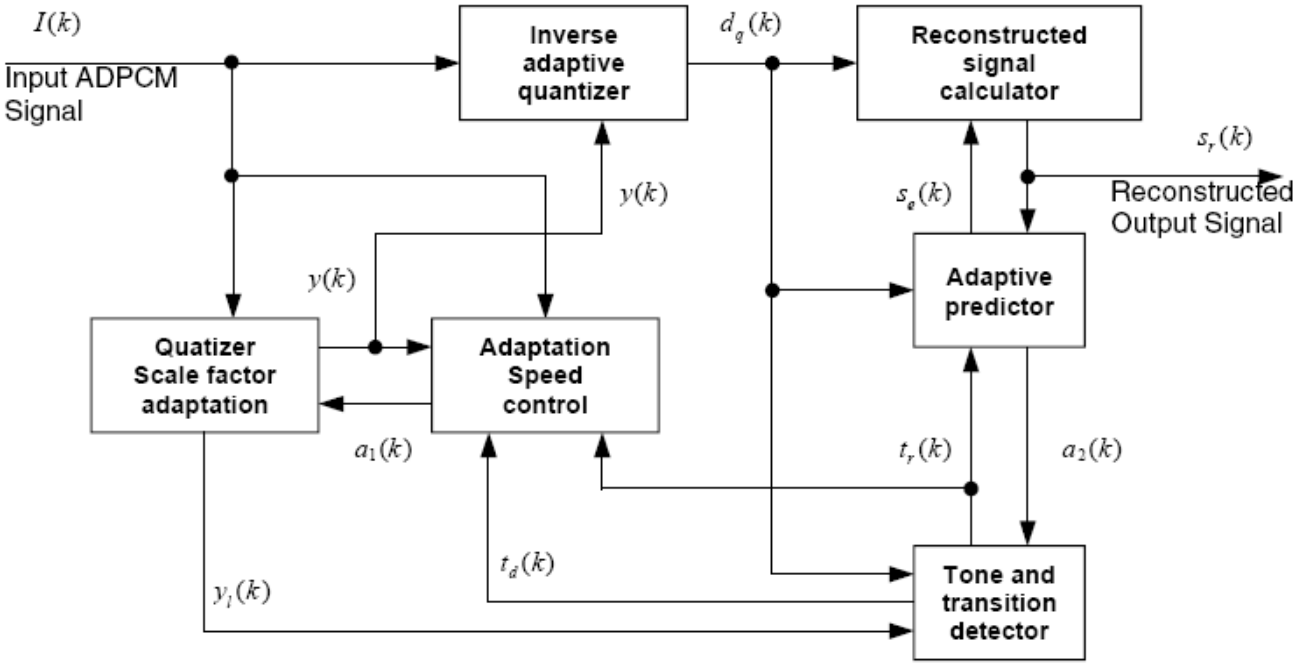


شکل (4-1) ساختار ساده سیستم ADPCM مطابق توصیه CCITT[2]

با دقت در این شکلها دیده می شود که گیرنده و فرستنده دارای ساختاری متقارن هستند به عبارت بهتر فرستنده خود دارای یک گیرنده محلی است و از سیگنالهایی استفاده می کند که در گیرنده نیز موجود است . این تقارن از ویژگی های مهم این سیستم است که امکان ردیابی و تنظیم گیرنده را نسبت به فرستنده فراهم می کند .



شکل (4–2) ساختار فرستنده ADPCM مطابق توصیه CCITT- G .726



شکل (4–3) ساختار گیرنده ADPCM مطابق توصیه CCITT- G .726

ورودی سیستم ، PCM لگاریتمی (log – PCM) است اما چرا ؟ علت این است که در شبکه های تلفن دیجیتال ، ADPCM بعنوان یک سیستم کدینگ پایه مطرح نیست بلکه اساسا یک مبدل است برای تبدیل کدهای log – PCM به کدهای ADPCM در حال حاضر در شبکه های دیجیتال بیشتر از log – PCM برای انتقال اطلاعات استفاده می شود . برای محاسبه سیگنال اختلاف (خطای پیشگویی) لازم است که سیگنال لگاریتمی به یکنواخت تبدیل شود . اگر log – PCM مطابق استاندارد G. 711 از CCITT بدست آمده باشد برای تبدیل آن به PCM یک نواخت نیز باید از همین استاندارد استفاده کرد . سیگنال PCM یکنواخت حاصل 13 بیتی (برای ورودی A - Law) یا 14 بیتی (برای ورودی μ -Law) خواهد بود . از تفاضل این سیگنال و سیگنال پیشگویی شده ، سیگنال تفاضل بدست می آید :

$$d(k) = s_l(k) - s_e(k) \qquad (1\text{–}4)$$

مرحله بعد کوانتیزاسیون وفقی است که در آن ابتدا از سیگنال اختلاف لگاریتم (مبنای 2) گرفته می شود و بوسیله فاکتور تنظیم گام ، y(k) ، نرمالیزه شده و سرانجام نتیجه ، کد شده و ارسال می شود . لگاریتم گرفتن از سیگنال اختلاف سبب افزایش SNR می شود کلمات کد با I(k) نمایش داده شده اند. فاکتور تنظیم از روی مقادیر گذشته نمونه های کد شده ، محاسبه می شود . بلوک کنترل سرعت تطبیق نیز یک تطبیق دو وضعیتی ایجاد می کند . تطبیق سریع برای سیگنالهایی با نوسانات بزرگ دامنه (مانند صحبت) و تطبیق کند برای سیگنالهایی که آهسته تر تغییر می کنند (مانند داده) .

برای بدست آوردن سیگنال بازسازی شده و سیگنال پیشگویی شده ، ابتدا با استفاده از یک کوانتیزه کننده وفقی معکوس ، سیگنال اختلاف متاثر از کوانتیزاسیون ، $d_9(k)$ حاصل می شود . از جمع این سیگنال ، با سیگنال پیشگویی ، سیگنال بازسازی شده ، $s_r(k)$ بدست می آید که در واقع همان خروجی گیرنده است (به شرط عدم خطا در کانال) . با استفاده از سیگنال بازسازی شده و سیگنال اختلاف $d_9(k)$ پیشگویی صورت می گیرد و این سیگنال پیشگویی شده حاصل از نمونه بعدی سیگنال ورودی کم می شود و بدین شکل حلقه فیدبک بسته می شود . بهره گیری از کوانتیزه کننده وفقی معکوس باعث کاهش خطای بازسازی می شود (با توجه به رابطه 3–11) زیرا $d_9(k)$ نویز کوانتیزاسیون کمتری را شامل می شود .

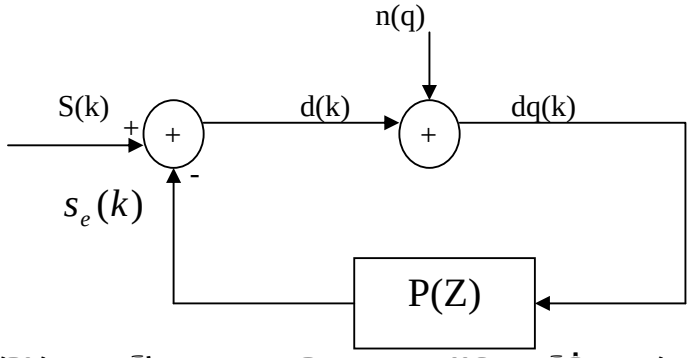
همانطور که گفتیم از یک پیشگویی کننده قطب و صفر استفاده می شود با دو قطب و شش صفر . انتخاب تعداد قطب ها و صفر ها خیلی مهم است . در اینجا با این انتخاب ، تعادل خوبی بین پیشگویی دقیق وحجم محاسبات ، حاصل می شود .

به منظور بهبود عملکرد سیستم برای سیگنالهایی که از مدم های FSK خارج شده اند ، بلوک تشخیص سیگنال تن و انتقال مورد استفاده قرار می گیر . اگر سیگنال ورودی به سیستم ، سیگنالی با باند جزئی [1] (مانند تن) باشد کوانتیزه کنند به سمت وضعیت تطبیق سریع رانده می شود . ضمنا این بلوک حالت انتقال از این نوع سیگنالل را تشخیص داده و در این حالت ضرایب پیشگویی کننده را صفر کرده و کوانتیزه کنند را سریعا به وضعیت تطبیق سریع می برد . از این بلوک دو سیگنال $t_d(k), t_r(k)$ خارج می شوند . اگر سیگنال ورودی سیگنال با باند جزئی باشد ، $t_d(k) = 1$ خواهد شد و در غیر این صورت صفر خوهد ماند . در حالت انتقال هم $t_r(k) = 1$ خواهد شد و در بقیه حالت صفر خواهد بود .

عملکرد گیرنده نیز مشابه فرستنده است . در گیرنده سیگنال یک نواخت خروجی تبدیل به سیگنال لگاریتمی می شود و عملکرد این بلوک درست عکس بلوک ورودی فرستنده است . وظیفه بلوک تنظیم کدینگ همزمان ، جلوگیری از انباشته شده خطا و ایجاد اعوجاج در سیستم های کدینگ متوالی همزمان[2] (ADPCM – PCM – ADPCM …) است به شرط آنکه اولا در کانال انتقال خطا رخ ندهد و ثانیا دنباله های بیت در حین ارسال بوسیله ابزارهای پردازش سیگنا دیجیتال دچار اختلال نگردند . در این بلوک سیگنال log – PCM مجددا به سیگنال ADPCM تبدیل می شود و با سیگنال وارد شده به گیرنده مقایسه می شود و در صورت خطا ، اصلاحات لازم در آن صورت می گیرد . با استفاده از این بلوک ، عملکرد ADPCM در سیستمهای کدینگ متوالی تا حدی بهبود می یابد اما بدیهی است که کیفیت عملکرد آن پایین تر از log – PCM خواهد بود و تعداد طبقات متوالی مجاز آن کمتر خواهد بود .

4–3) بررسی پیشگویی کننده

در فصل گذشته بیان کردیم که چرا برای پیشگویی سیگنال صحبت استفاده از یک پیشگویی کننده صفر و قطب مناسب تر است .



شکل (4–4) ساختار ساده پیشگویی کننده در سیستم ADPCMاگر به ساختار شکل (4–4) به دقت بنگریم مشاهده می کنیم که پیشگویی کننده p(z) در واقع نوعی سنتز کننده سیگنال

[1] Partial Band Signal
[2] Synchronous tandem Codings

صحبت با تحریک سیگنال خطا است چون باید از روی سیگنال خطای پیشگویی ، تخمینی از سیگنال صحبت را ارائه دهد . با این دید ، سیگنال $d_9(k)$ مشابه یک سیگنال تحریک است و فیلتر $p(z)$ نیز باید مشابه تابع تبدیل محفظه صوتی عمل کند . بنابراین یم پیشگویی کنند خوب باید بتواند تغییرات شکل محفظه صوتی را بخوبی دنبال کند .

با در نظر گرفتن ویژگی فوق و مسائل مربوط به پایداری ، انتشار خطا در گیرنده ، سادگی و کم حجم بودن محاسبات الگوریتم وفقی ، CCITT پیشگویی کننده ای با دو قطب و شش صفر را پیشنهاد کرده است تا مصالحه ای مناسب بین تمام پارامتر های فوق برقرار کند . با انتخاب فوق یک بهینگی نسبی [1] در پیشگویی کننده بوجود خواهد آمد .

شکل (4-5) پیشگویی کننده پیشنهادی CCITT را نشان می دهد که در آن دو بخش تمام قطب (AR) و تمام صفر (MA) جداگانه نشان داده شده اند . بخش AR از روی سیگنال بازسازی شده و بخش MA از روی خطا ، پیشگویی می کنند . فیلتر FIR مرتبه 6 را با $B(z)$ و فیلتر IIR مرتبه 2 را با $A(z)$ نمایش می دهیم .

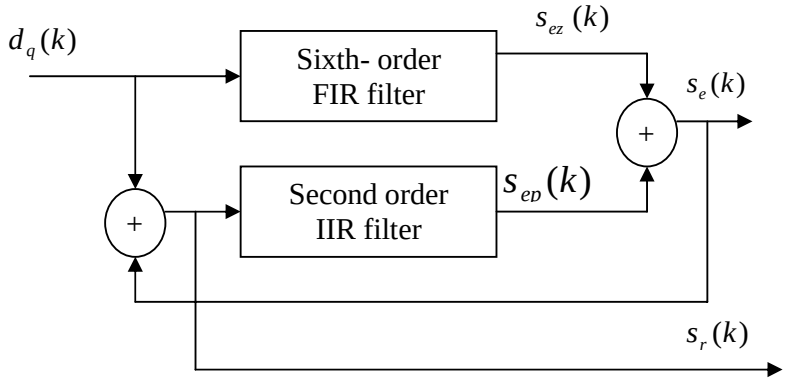
مطابق شکل (5-5) سیگنال پیشگویی شده چنین بدست می آید :

$$s_e(k) = \sum_{i-1}^{2} a_i s_r(k-i) + \sum_{j=1}^{6} b_j d_q(k-j) \qquad (4-2)$$

رابطه (4-2) رابطه اساسی پیشگویی در سیستم ADPCM پیشنهادی CCITT که در آن $\{a_i\}$ و $\{b_j\}$ مجموعه ضرایب پیشگویی هستند و بوسیله الگوریتم های وفقی باید آنها را تنظیم کرد .

$B(z)$ یک فیلتر FIR با ساختار عرضی است بنابراین برای تنظیم ضرایب آن می توان از الگوریتم ساده و خوب LMS بهره برد . اما $A(z)$ یک فیلتر IIR است و در نتیجه تنظیم ضرایب آن مشکل و پیچیده است . پس لازم است که به نموی فیلتر IIR را با فیلتر FIR تقریب بزنیم .

(الف)



[1] Suboptimality

شکل (4-5) ساختار دقیق پیشگویی کننده در سیستم ADPCM پیشنهادی CCITT : (الف) ساختار کامل قطب و صفر (ب) فیلتر تمام صفر مرتبه شش FIR (ج) فیلتر تمام قطب مرتبه دو IIR [2]

مطابق شکل (4-5- الف) تابع تبدیل پیشگویی کننده چنین است :

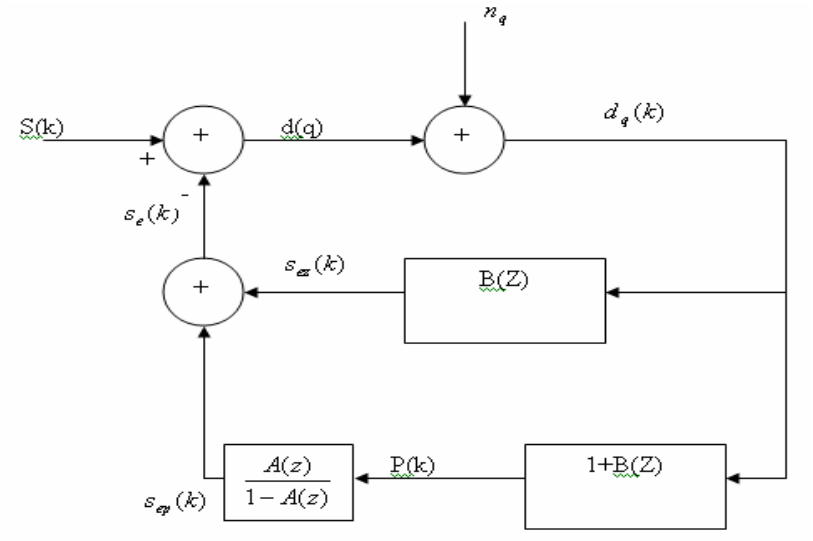
$$P(z)=\frac{S_e(Z)}{D_q(Z)}=\frac{A(Z)+B(Z)}{1-A(Z)} \quad (3\text{-}4)$$

رابطه (4-3) را می توان بدین شکل هم نوشت :

$$P(Z)=\frac{A(Z)+B(Z)}{1-A(Z)}=\frac{A(Z)[1+B(Z)]+B(Z)[1-A(Z)]}{1-A(Z)} \quad (4\text{-}4)$$

$$=[1+B(Z)].\frac{A(Z)}{1-A(Z)}+B(Z)$$

ساختار معادل رابطه (4-4) در شکل (4-6) نشان داده شده است .



شکل (4-6) ساختار بهبود یافته پیشگویی کننده در ADPCM

در این ساختار فیلتر $\frac{A(Z)}{1-A(Z)}$ یک فیلتر IIR است . در توصیه CCITT فیلتر فوق با یک فیلتر FIR مرتبه دوم تقریب زده می شود و با این تقریب تما فیلتر ها FIR خواهند شد و می توان برای تنظیم آنها تنها از الگوریتم های مربوط به فیلتر های FIR استفاده کرد . اما تقریب به این شکل است :

$$\frac{A(Z)}{1-A(Z)}=\frac{a_1z^{-1}+a_2z^{-2}}{1-(a_1z^{-1}+a_2z^{-2})}=a_1z^{-1}+(a_1^{\ 2}+a_2)z^{-2} \quad (5\text{-}4)$$

$$+\frac{a_1^{\ 3}z^{-3}+a_2(a_1^{\ 2}+a_2)z^{-4}}{1-(a_1z^{-1}+a_2z^{-2})}$$

با چشم پوشی از عبارت سوم سمت راست ، فیلتر فوق یک فیلتر FIR خواهد شد :

$$\frac{A(Z)}{1-A(Z)} \approx a_1 z^{-1} + (a_1^2 + a_2) z^{-2} \overset{\Delta}{=} f_1 z^{-1} + f_2 z^{-2} \quad (4-6)$$

پس حالا می توان از الگوریتم LMS برای تنظیم ضرایب f_1 استفاده کرد :

$$f_i(k+1) = f_i(k) + \mu_i p(k) p(k-i) \quad (4-7)$$

با جایگزینی مقادیر f_1 با مقادیر معادل آن در رابطه (4-6) ،رابطه تنظیم ضرایب برای a_1 و a_2 بدست می آید :

$$a_1(k+1) = a_1(k) + \mu_1 p(k) p(k-i) \quad (4-8)$$

$$a_2(k+1) = a_2(k) - [a_1(k+1) + a_1(k)] \mu_1 p(k) p(k-1) + \mu_2 p(k) p(k-2) \quad (4-9)$$

در توصیه CCITT عبارت $a_1(k+1) + a_1(k)$ را بصورت تابعی از $a_1(k)$ تقریب زده اند :

$$a_1(k+1) + a_1(k) = f[a_1(k)] \quad (4-10)$$

که تعریف $f(a_1)$ به این صورت است :

(4-11)

$$f(a1) = \begin{cases} 4a1 & ; |a1| \le 2^{-1} \\ 2\,\mathrm{sgn}\,(a1) & ; |a1| > 2^{-1} \end{cases}$$

سیگنال واسطه $p(k)$ نیز از رابطه زیر بدست می آید : (مطابق شکل 5-6)

$$p(k) = d_q(k) + s_{ez}(k) = d_q(k) + \sum_{i=1}^{6} b_i(k-1) d_q(k-i) \quad (4-12)$$

رابطه تنظیم ضرایب صفرها یعنی $\{b_i\}$ نیز مستقیما از الگوریتم LMS حاصل می شود :

$$b_i(k+1) = b_i(k) + \mu d_q(k) d_q(k-i) \quad (4-13)$$

روابط (4-9) و (4-10) و (4-13) اساس تنظیم ضرایب در توصیه CCITT هستند و روابط نهایی با اصلاحاتی در آنها بدست می آیند که این اصلاحات از این قرار است :

1- برای تنظیم ضرایب بجای خود سیگنالها از تابع علامت آنها (تابع sign) استفاده می شود . این کار باعث سادگی و کاهش بار محاسبات می شود . در نتیجه سرعت تنظیم و تطبیق افزایش می یابد که در کاربردهای بلادرنگ [1] عامل بسیار مهمی است .

[1] Real - Time

2- برای جلوگیری از انتشار خطا در صورت بروز خطا در کانال ارسالی ، از فاکتور نشتی یا ضریب فراموشی استفاده می شود که سبب می شود پس از مدتی اثر خطا فراموش شده و گیرنده و فرستنده دوباره با یکدیگر سنکرون شوند .

با توضیحات فوق روابط نهایی تنظیم ضرایب پیشگویی کننده در توصیه CCITT چنین است :

$$a_1(k) = (1 - 2^{-8})a_1(k-1) + (3 \cdot 2^{-8}) \operatorname{sgn}[p(k)] \operatorname{sgn}[p(k-1)], \quad (14\text{-}4)$$

$$a_2(k) = (1-2^{-7})a_2(k-1) + 2^{-7} \operatorname{sgn}[p(k)] \operatorname{sgn}[p(k-2)] - f[a_1(k-1)] \operatorname{sgn}[p(k)] \operatorname{sgn}[p(k-1)] \quad (15\text{-}4)$$

$$b_i(k) = (1 - 2^{-8})b_i(k-1) + 2^{-7} \operatorname{sgn}[dq(k)] \operatorname{sgn}[dq(k-i)], \quad (16\text{-}4)$$

برای تضمین پایداری ، محدودیتهای زیر به ضرایب اعمال می گردد :

$$|a_2(k)| \le 0.75 \text{ and } |a_1(k)| \le 1 - 2^{-4} - a_2(k). \qquad , \qquad -2 \le b_i(k) \le 2 \quad (17\text{-}4)$$

پیشگویی کننده با الگوریتم فوق عملکرد بسیار خوبی برای سیگنال صحبت ارائه می کند .

با توجه به توضیحات فوق می توان برنامه پیشگویی کننده را مطابق توصیه CCITT به صورت زیر نوشت.

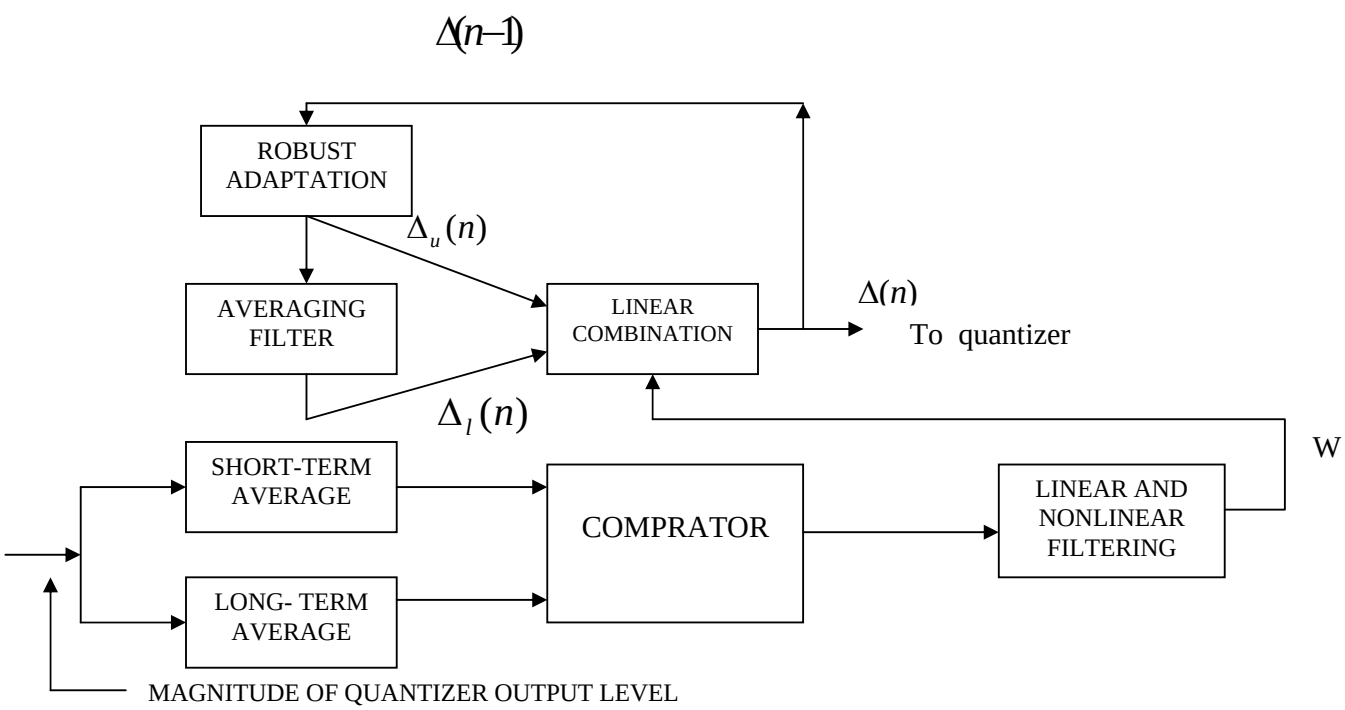
```
b=rand(1,6);
a2=rand(1)*0.7  ;
a1=1-a2-0.05    ;
sr(1:6)=[ 0.5695    0.1593    0.5944    0.3311    0.6586  0.5] ;
daq(1:6)=[ 0.5695    0.1593    0.5944    0.3311    0.6586  0.5];
for k=7:length(sinp)
%-----------------------------
iir= a1*sr(k-1)+a2*sr(k-2  );
for j=1:6
fir=+ b(j)*daq(k-j );
end
se(k)=fir+iir   ;
%----------------------------
daq(1,k)=sinp(k)-se(k );

%-----------------------------
p(k)=daq(k)+fir;
%------------------------------
for i=1:6
b(i)=b(i)+u*daq(k)*daq(k-i  );
end
%------------------------------
a1=a1+u1*p(k)*p(k-1 );
```

```
if abs(a1)<=0.5
f=4*a1   ;
else
f=2*sign(a1 );
end
a2=a2-f*u1*p(k)*p(k-1)+u2*p(k)*p(k-2 );
-------------------------------%
sr(k)=se(k)+daq(k  );
end
```

4-4) بررسی کوانتیزه کننده

کوانتیزه کننده پیشنهادی CCITT ، کوانتیزه کننده دارای قفل دینامیکی (DLQ) نام دارد که به کوانتیزه کننده مقاوم شباهت دارد (رجوع به بخش 3-5-3) تفاوت عمده آنها این است که در DLQ با توجه به ورودی سرعت تنظیم تغییر می کند . در DLQ اگر ورودی یک سیگنال ایستان باشد عمل تطبیق متوقف می شود و مانند یک کوانتیزه کننده غیر وفقی عمل می کند . این ویژگی باعث بهبود کیفیت ADPCM هنگام عمل با داده های باند صوتی می شود . بلوک دیاگرام DLQ در شکل (4-7) نشان داده شده است .

مطابق شکل ابتدا یک متوسط گیری کوتاه مدت و یک متوسط گیری بلند مدت روی دامنه خروجی کوانتزه کننده انجام می گیرد سپس این دو میانگین با هم مقایسه می شوند . اگر به هم نزدیک باشند، سیگنال ورودی ، داده است . تشخیص بین داده و صحبت وظیفه بخش پایین DLQ است که در شکل (4-2) همان بلوک کنترل سرعت تطبیق است . بخش بالای DLQ که همان بلوک تنظیم گام کوانتیزه کننده در شکل (4-2) است ، دارای دو وضعیت قفل شده و قفل نشده است . اگر سیگنال ورودی بعنوان صحبت تشخیص داده شد ، روی وضعیت قفل نشده می رود وم مشابه کوانتیزه کننده مقاوم عمل می کند . ابتدا چند سیگنال بکار رفته را معرفی می کنیم :



شکل (4–7) ساختار یک DLQ

$$y_u(k) = \log_2 \Delta_u(k) \ , \ y_l(k) = \log_2 \Delta_l(k) \ , \ y(k) = \log_2 \Delta(k) \qquad (4\text{–}18)$$

$a_1(k)$ نیز همان پارامتر وزن (w) است که مقداری بین صفر و یک دارد . در این صورت مقدار گـام چنـین خواهد بود :

$$y(k) = a_l(k)\, y_u(k-1) + [1 - a_l(k)]\, y_l(k-1) \qquad (4\text{–}19)$$

همانطور که در (4–18) بیان کردیم $y_u(k)$ و $y_1(k)$ به ترتیب لگاریتم اندازه گام در وضعیت قفـل نشـده و قفل شده هستند و از روابط زیر محاسبه می شوند :

$$y_u(k) = \frac{31}{32} y(k) + \log_2 p \qquad 1.065 \le y_u(k) \le 10 \qquad (4\text{–}20)$$

$$y_l(k) = (1 - 2^{-6})\, y_l(k-1) + 2^{-6}\, y_u(k) \qquad (4\text{–}21)$$

در رابطه (4–20) p همان ضریب گام است که در فصل سوم معرفی کردیم و گفتیم که تابعی از اندازه لغات کد است . در توصیه CCITT نیز مقادیر $\log_2 p$ به صورت تابعی از اندازه لغات که داده شده است . (برای 24kbps) :

|I(k) |	3	2	1	0
W[I(k)]	36.38	8.56	1.88	–0.25

$$\log_2 p = \frac{1}{32} W[I(k)] \qquad (4\text{–}22)$$

اگر ورودی ، سیگنال صحبت باشد ، $a_1(k) = 1$ خواهد بود . در این صورت با ترکیب (4-19) و (4-20) خواهیم داشت :

$$yu(k) = (1 - 2^{-5})\, y(k) + 2^{-5}\, W[I(k)], \quad (4\text{-}23)$$

که دقیقا همان الگوریتم کوانتیزه کننده مقاوم است . اما برای یک ورودی ایستان (مثل داده) $a_1(k)=0$ خواهد شد و با ترکیب روابط (4-19) ، (4-20) و (4-21) خواهیم داشت :

$$y(k) = y_l(k-1) \quad (4\text{-}24)$$

$$y_l(k) = \frac{2047}{2048} y_l(k-1) + \frac{1}{64}\log_2 p = \frac{2047}{2048} y_l(k-1) + \frac{1}{2048} W[I(k)] \approx y_l(k-1) \quad (4\text{-}25)$$

با استفاده از روابط (4-24) و (4-25) نتیجه گرفته می شود که :

$$y(k+1) = y_l(k) \approx y_l(k-1) = y(k) \quad (4\text{-}26)$$

یعنی گام کوانتیزه تقریبا ثابت است پس در این حالت (ورودی ایستان) DLQ مانند یک کوانتیزه کننده یک نواخت عمل می کند . اما برای بدست آوردن $a_1(k)$ همانطور که گفتیم باید 2 نوع متوسط گیری کوتاه مدت و دراز مدت روی کدهای خروجی انجام داد . اما چون هدف مقایسه این دو میانگین است و اندازه مطلق آنها مورد نظر نیست ، در توصیه CCITT به جای متوسط گیری روی خروجی کوانتیزه کننده یعنی $I(K)$ ، روی تابعی از $I(K)$ متوسط گیری انجام می شود که مقادیر این تابع برای نرخ بیت 24kbps چنین است :

$\lvert I(k) \rvert$	3	2	1	0
$F[I(k)]$	7	2	1	0

با این کار قبل از متوسط گیری یک وزن دهی مناسب انجام می گیرد تا اعمال ساده تر شود . برای متوسط گیری کوتاه مدت 32 نمونه و برای متوسط گیری بلند مدت 128 نمونه در نظر گرفته می شود و برای بدست آوردن میانگین باید مجموع نمونه ها را بر تعداد آنها تقسیم کرد . ولی چون می خواهیم با ورود هر نمونه ، گام کوانتیزه کننده را تنظیم کنیم ، روش تحلیلی فوق ، محاسبات زیاد و تاخیر را به دنبال دارد پس باید متوسط گیری را هم به طور بازگشتی انجام داد :

$$d_{ms}(k) = (1 - 2^{-5})\, d_{ms}(k-1) + 2^{-5}\, F[I(k)] \quad (4\text{-}27)$$

$$d_{ml}(k) = (1 - 2^{-7})\, d_{ml}(k-1) + 2^{-7}\, F[I(k)] \quad (4\text{-}28)$$

بنابر این $d_{ms}(k)$ و $d_{ml}(k)$ به ترتیب متوسط نسبی کوتاه مدت و دراز مدت تابع $F[I(K)]$ هستند که در آنها به تدریج اثر نمونه های قبلی کم می شود . در $d_{ms}(k)$ از نمونه 32 به قبل و در $d_{ml}(k)$

از نمونه 128 به قبل را می توان صرف نظر کرد . حال با استفاده از این دو میانگین سیگنال کنترل سرعت ، $a_p(k)$ را محاسبه و از روی آن $a_1(k)$ را که پارامتر وزن یا ضریب کنترل سرعت است ، بدست می آوریم :

$$ap(k) = \begin{cases} (1-2^{-4})ap(k-1) + 2^{-3} & \text{if } |dms(k) - dml(k)| \geq 2^{-3}\, dml(k); \\ (1-2^{-4})ap(k-1) + 2^{-3} & \text{if } td(k) = 1 \\ (1-2^{-4})ap(k-1) + 2^{-3} & \text{if } y(k) < 3; \\ 1 & \text{if } tr(k) = 1; \\ (1-2^{-4})ap(k-1) & \text{otherwise;} \end{cases}$$

(4- 29)

(4- 30)

$$al(k) = \begin{cases} 1 & ;ap(k-1) > 1 \\ ap(k-1) & ;ap(k-1) \leq 1 \end{cases}$$

همانطور که دو رابطه فوق نشان می دهند ، اگر ورودی ، صحبت باشد ، $a_1(k)$ به تدریج افزایش می یابد تا به مقدار ثابت یک برسد ، اما اگر ورودی داده (Data) باشد : $a_p(k)$ متوالیا در $\frac{15}{16}$ ضرب شده و به همراه $a_1(k)$ به سمت صفر میل می کنند .

در رابطه (4- 29) شرط $y(k) < 3$ معادل کانال خنثی است یعنی ورودی ، نویز کم دامنه است . در این حالت نیز DLQ به وضعیت قفل نشده برده می شود تا برای دریافت صحبت آماده باشد یا به عبارت دیگر به ورودی صحبت ، سریع تر پاسخ دهد . علت این کار این است که مکالمات معمولا با سکوت نیز همراه است اما در ارسال داده کمتر وقفه پیش می آید . در مورد سیگنالهای با باند جزئی نیز عملکرد DLQ مشابه سیگنال صحبت است با این تفاوت که در حالت انتقال از این سیگنالها DLQ سریعا به وضعیت تطبیق سریع برده می شود .

DLQ شامل 4 بلوک از بلوک دیاگرام شکل (4-2) است . بلوک کنترل سرعت تطبیق ، بلوک تشخیص سیگنال تن و بلوک تنظیم ضریب گام را شرح دادیم . حال به بررسی بلوک چهارم یعنی کوانتیزه کننده وفقی می پردازیم :

کوانتیزه کننده توصیه CCITT یک کوانتیزه کننده غیر یکنواخت است . جدول (4-1) مشخصه ورودی و خروجی نرمالیزه شده کوانتیزه کننده را برای ADPCM پیشنهادی CCITT با نرخ بیت 24 kbps نشان می دهد .

(جدول 4–1) مشخصه ورودی – خروجی کوانتیزه کننده برای نرخ بیت 24 kbps

Normalized quantizer input range $\log_2 \lvert d(k) \rvert - y(k)$	$\lvert I(k) \rvert$	Normalized quantizer output $\log_2 \lvert d_q(k) \rvert - y(k)$
$[2.58, +\infty)$	3	2.91
$[1.70, 2.58)$	2	2.13
$[0.06, 1.70)$	1	1.05
$(-\infty, -0.06)$	0	$-\infty$

در جدول (4–1) از ستون اول و دوم در کوانتیزه کننده فرستنده و از ستون دوم و سوم در کوانتیزه کننده معکوس فرستنده و گیرنده برای کد کردن و کد برگرداندن استفاده می شود .

بعنوان نمونه این برنامه که یک کوانتیزه کننده ساده است استفاده شده است

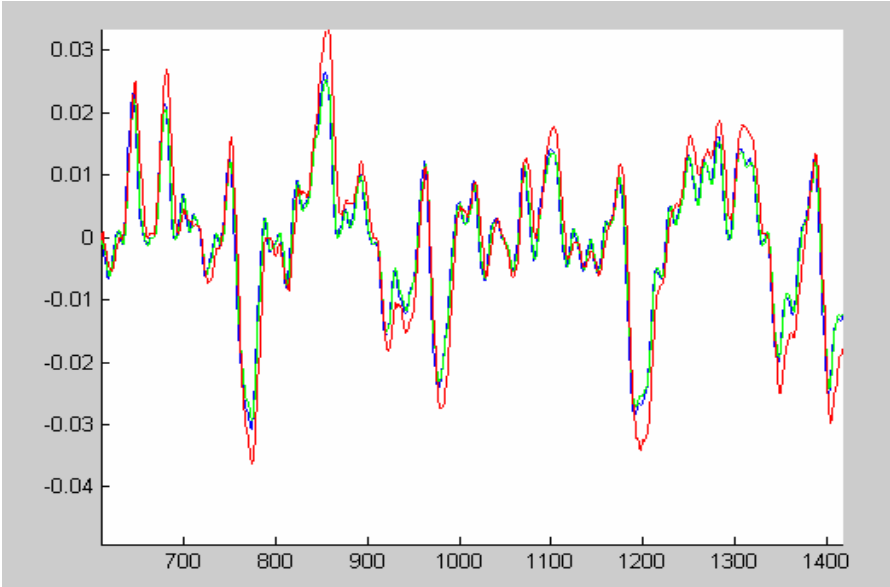
```
y=rand(1 );
  w=[-0.25 1.88 8.56 36.38];
  if daq(k)~=0
  nq=log2(21000*abs(daq(k)))-y    ;
  I=(2.58<nq&nq<inf).*3+(1.7<nq&nq<2.58).*2+(0.06<nq&nq<1.7).*1+
  (-inf<nq&nq<0.06).*0;
  else
  I=0
  end
  q(k)=I*a(k);    %quantzer output
  y=(31/32)*y+(1/32)*w(I+1 )
```

برنامه گیرنده و بازسازی سیگنال کد شده:

```
  function sr=reserver1(i,s)_
  u1=0.55;
  u2=0.55;
  u=0.60;
 p=zeros(1,length(i));
 sr=zeros(1,length(i)) ;
 se=zeros(1,length(i));
  b=zeros(1,6);
```

```
b=rand(1,6);
a2=rand(1)*0.7;
a1=1-a2-0.05;
sr(1:6)=[ 0.5695    0.1593    0.5944    0.3311    0.6586  0.5]
daq(1:6)=[ 0.5695    0.1593    0.5944    0.3311    0.6586  0.5]
y=rand(1)
w=[-0.25 1.88 8.56 36.38]
n=[-inf 1.05 2.0 2.2]
for j=7:length(i )
I=abs(i(j  ));
y=(31/32)*y+(1/32)*w(I+1 );
daq(j)=s(j)*2^(n(I+1)+y)/30000  ;
end
for k=7:length(i)
-------------------------------%
iir= a1*sr(k-1)+a2*sr(k-2 );
for j=1:6
fir=+ b(j)*daq(k-j );
end
se(k)=fir+iir  ;
-----------------------------%
p(k)=daq(k)+fir ;
-----------------------------%
for i=1:6
b(i)=b(i)+u*daq(k)*daq(k-i );
end
-----------------------------%
a1=a1+u1*p(k)*p(k-1 );
if abs(a1)<=0.5
f=4*a1 ;
else
f=2*sign(a1 );
end
a2=a2-f*u1*p(k)*p(k-1)+u2*p(k)*p(k-2 );
-------------------------------%
sr(k)=se(k)+daq(k );
end
```

سیگنال اصلی(آبی) سیگنال پیش بینی شده(سبز) سیگنال باز سازی شده(قرمز)



فهرست مراجع:

[1] Witten,I.H. ;PRINCIPLES OF COMPUTER SPEECH ;ACADEMIC PRESS ,1982

[2] PAPAMICHALIS P.E ; RRACTICAL APPROACHES TO SPEECH CODING;PRENTICE HALL,1987

[3] TANENBAUM A.S ;COMPUTER NETWORKS;PRENTICE HALL,1989

[4] SAITO S.AND K.NAKATA;FUNDAMENTALS OF SPEECH SIGNAL PROCESSING; ACADEMIC PRESS,1985

[5] INCE,A.N; DIGITAL SPEECH PROCESSING ;KLUWER ACADEMIC PUBLISHERS,1992

[6] OPPENHEIM, A.V. AND R.W.SCHAFER; DESCRETE TIME SIGNAL PROCESSING; PRENTICE HALL ,1989

[7] PARSONS ,T.W ;VOICE AND SPEECH PROCESSING;MCGRAW HILL,1987

[8] JAYANT, N.S. AND P. NOLL; DIGITAL CODING OF WAVEFORMS; PRENTICE HALL,1984

[9] RABIER,L.R. AND R.W.SCHAFER; DIGITAL SPEECH PROCESSING; PRENTICE HALL 1978

[10] FURUI, S. ; DIGITAL SPEECH PROCESSING , SYNTHESIS AND RECOGNITION; MARCEL DEKKER INC.,1989

[11] CCITT RECOMMENDATION G.726 1990