باسمه تعالی

سیستم های چندرسانه‌ای (۳۴۲-۴۰)
دانشکده مهندسی کامپیوتر
ترم پائیز ۱۳۸۵
دکتر حمیدرضا ربیعی

تکلیف شماره۲: فشرده سازی سیگنال صوتی/تصویری: گفتار و صوت

۱- مقدمه

آنچه توزیع سیگنالهای صوت و گفتار را بدون نیاز به اختصاص پهنای باند وسیعی برای انتقال و ذخیره حجم زیادی از سیگنالهای صوتی ممکن می سازد، تکنیک فشرده سازی یا به بیان دیگر کد کردن است. این تکنیک مقدار داده موردنیاز برای انتقال و ذخیره صوت نمونه برداری شده دیجیتالی را هم در طول مرحله تبدیل آنالوگ – به – دیجیتال و هم پس از ذخیره فایل خام به صورت دیجیتالی، کاهش می دهد. فشرده سازی و عکس فشرده سازی بوسیله الگوریتمهای متعددی قابل پیاده سازی است که می تواند در کاربردهای نرم افزاری یا کاربردهای خاص مدارهای مجتمع (تراشه ها) بکار گرفته شود.

استانداردهای متعدد جهانی برای کد کردن ویدیو و صوت پایه ریزی شده اند. برخی از این استانداردها شامل MPEG-۱، MPEG-۲ و MPEG-٤ می باشند. البته استانداردهای متعددی برای فشرده سازی و عکس عمل فشرده سازی شکل موجهای صوت و گفتار برای کاربردهای desktop چند رسانه ای وجود دارد. بخش های بعدی به الگوریتمهای معمول و انواع گوناگونی از روشهای فشرده سازی برای صوت و گفتار اختصاص دارد.

۲- تئوریها و طرحها

ما قبلاً تئوری نمونه برداری را در تکلیف ۱ مورد بررسی قرار داده ایم. همچنین نشان دادیم که نمونه های یک سیگنال آنالوگ نمایش منحصر به فردی از سیگنال اولیه می باشند به شرطی که پهنای باند محدود داشته باشند و نرخ نمونه برداری حداقل دوبرابر فرکانس سیگنال باشد. از آنجایی که ما با نمایش دیجیتالی صوت و گفتار سر و کار داریم، احتیاج به دانستن خصوصیات طیفی صدا و گفتار داریم. به راحتی مشاهده می شود که برای صداهای واکدار، دامنه طیف فرکانسی سیگنال در فرکانسهای بالای ٤KHz، ٤٠dB پایین تر از قله طیفی سیگنال است. از طرف دیگر، در سیگنالهای صوتی، طیف سیگنال حتی در فرکانسهای بالای ۸KHz به طور قابل ملاحظه ای افت نمی کند. علاوه بر این، در کاربردهای کامپیوتری برای نمایش سیگنال نمونه برداری شده، مقادیر ممکن یک نمونه که در محدودهٔ پیوسته ای تغییر می کنند باید به تعداد محدودی از مقادیر گسسته تبدیل شود. این پروسه کوانتیزاسیون نامیده می شود.

۲-۱- کوانتیزاسیون سیگنالهای نمونه برداری شده

محدوده ها و سطوح کوانتیزاسیون ممکن است به صورتهای متعددی انتخاب شوند که بستگی به کاربردهای از پیش تعیین شدهٔ نمایش دیجیتالی آن دارد. با کوانتیزاسیون یکنواخت، محدودهٔ دینامیک (حداقل تا حداکثر) سیگنال R، به W بازه با طول یکسان Δ تقسیم می شود. ما Δ را پلهٔ کوانتیزاسیون می نامیم. رابطه ورودی و مقدار کوانتیزه نشده، و خروجی (مقدار کوانتیزه شده) برای کوانتیزاسیون یکنواخت در شکل۱ نشان داده شده است. که در آن x_i، محدودهٔ راست بازهٔ i و $\hat{x}_i$ سطح کوانتیزه شدهٔ این بازه است که شرط های زیر را برآورده می سازد.

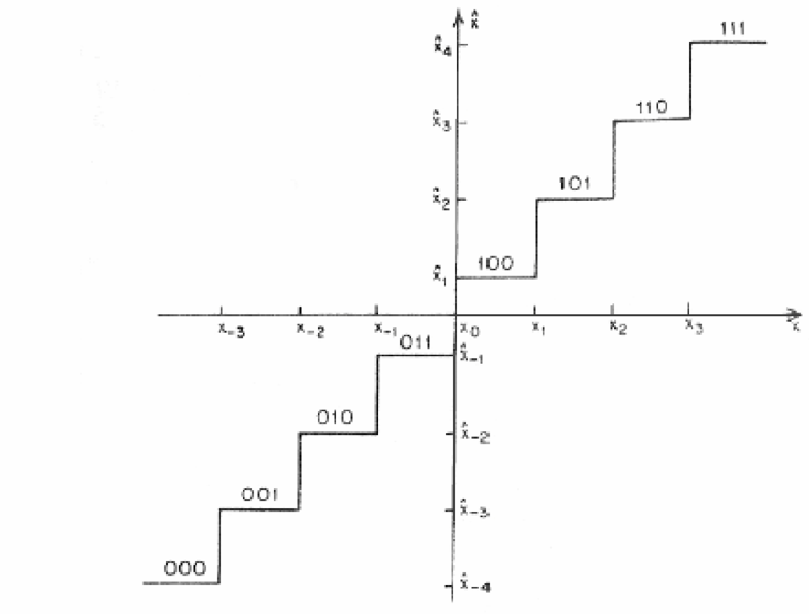
$$x_i - x_{i-1} = \Delta \qquad (۱-۳)$$

$$\widehat{x}_i - \widehat{x}_{i-1} = \Delta. \quad (۲-۳)$$

هر مقدار در محدودهٔ iام به مقدار میانی این محدوده نگاشت می‌شود.

$$Q(x) = \widehat{x}_i = X_{min} + (i-1)\,\Delta + \Delta/2, \text{ if } x_{i-1} <= x < x_i \quad (۳-۳)$$

در کامپیوتر، هر سطح با یک کلمه کد باینری نشان داده می شود. با W سطح کوانیز شده، هر سطح می‌تواند با $B = [\log_۲(L)]$ بیت نشان داده شود (شکل ۱).



شکل۱- خصوصیات ورودی – خروجی یک کوانیزه کنندهٔ ۳ بیتی

اگر محدودهٔ سیگنال R باشد، یک کوانیزه کننده یکنواخت فقط یک پارامتر دارد: تعداد سطوح N یا اندازه کوانتیزاسیون Δ، که هر دو با رابطهٔ زیر به هم ارتباط دارند.

$$\Delta = R / N. \quad (٤-۳)$$

تعداد سطوح N معمولاً به گونه ای انتخاب می شوند که به صورت ۲^B باشند تا بهترین استفاده از کلمهٔ کد B بیتی شود. اگر سیگنال تابع چگالی احتمال متقارن باشد به این ترتیب که $|x(n)| \leq x_{max}$ یا $R=۲x_{max}$ باشد،آنگاه باید مقادیر زیر تنظیم شوند.

$$2\,X_{\max} = \Delta\, 2^B \ \text{ or } \ \Delta = \frac{2\,X_{\max}}{2^B}. \quad (٥-۳)$$

در بحث تاثیرات کوانتیزاسیون مفید به نظر می رسد که مقادیر نمونه برداری شده $\widehat{x}(n)$ را به صورت زیر نمایش دهیم

$$\widehat{x}(n) = x(n) + e(n) \quad (٦-۳)$$

که $x(n)$ نمونه کوانتیزه نشده، $e(n)$ خطای کوانتیزه یا نویز کوانتیزه است. از شکل ۱ دیده می شود که اگر Δ و B مانند رابطهٔ (۳-٥) انتخاب شوند، آنگاه

$$-\frac{\Delta}{2} \leq e(n) \leq \frac{\Delta}{2} \quad (۷-۳)$$

نسبت سیگنال به نویز کوانتیزه در رابطه زیر بیان شده است.

$$SNR=\frac{\sigma_x^2}{\sigma_e^2}=\frac{E[x^2(n)]}{E[e^2(n)]}=\frac{\sum_n x^2(n)}{\sum_n e^2(n)}. \tag{۸-۳}$$

به یاد بیاوریم که برای سیگنال با توزیع یکنواخت در محدودهٔ R، واریانس برابر $\frac{R^2}{12}$ است. اگر توزیع دامنه نویز در بازهٔ $(-\frac{\Delta}{2},\frac{\Delta}{2})$ یکنواخت باشد، رابطه زیر برای نویز نتیجه می شود.

$$\sigma_e^2=\frac{\Delta^2}{12}=\frac{x_{\max}^2}{(3)2^{2B}} \tag{۹-۳}$$

با جایگزینی رابطهٔ (۳-۹) در رابطهٔ (۳-۸):

$$SNR=\frac{(3)2^{2B}}{\left[\frac{X_{\max}}{\sigma_x}\right]^2} \tag{۱۰-۳}$$

یا بیان خطای کوانتیزاسیون در واحد dB

$$SNR=10\log_{10}\left[\frac{\sigma_x^2}{\sigma_e^2}\right]=6B+4.77-20\log_{10}\left[\frac{X_{\max}}{\sigma_x}\right]. \tag{۱۱-۳}$$

اگر محدودهٔ کوانتیزاسیون را $x_{\max}=4\sigma_x$ فرض کنیم سپس رابطهٔ (۳-۱۱) به صورت زیر در می آید.

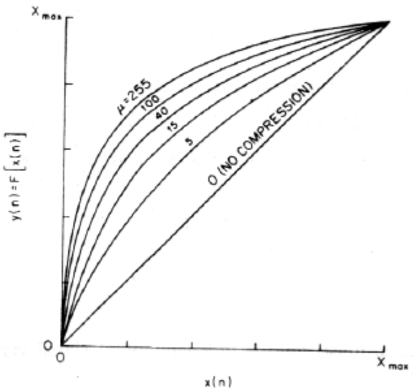
$$SNR=6B-7.2(dB)\,. \tag{۱۲-۳}$$

این رابطه بیان می کند که هر بیت اضافی، ٦dB به بهبود SNR کمک می کند. برای حفظ اعتبار کوانتیزه کردن یکنواخت، لازم است تا تعداد بیت بیشتری نسبت به آنالیز قبلی که در آن سیگنال ایستان و دارای توزیع متقارن فرض می شد و $X_{\max}=4\sigma_x$ بود، اختصاص یابد. به عنوان مثال، در حالی که رابطهٔ (۳-۱۲) تعداد بیت ها (B) را برابر۷ قرار می دهد تا SNR (حدود ٣٦dB) کیفیت قابل قبولی را در اغلب سیستمهای مخابراتی تأمین کند، به طور معمول تعداد بیت های موردنیاز برای تأمین کیفیت بالای سیگنال گفتار با کوانتیزاسیون یکنواخت، ۱۱ بیت است.

۲-۱-۲- μ – law

کوانتیزه کردن یکنواخت تنها برای سیگنالهای با توزیع یکنواخت بهینه است. برای سیگنالهایی که نزدیک مقادیر کوچک دامنه تجمع دارند، به عنوان مثال توزیع گوسی با میانگین صفر، بهتر است که دامنه های کوچک با دقت بیشتری کوانتیزه شوند. برای تحقق این امر ابتدا باید نگاشتی به سیگنال کرد به طوری که مقادیر کوچک را تقویت کند و سپس یک کوانتیزه کنندهٔ یکنواخت به سیگنال نگاشت شده اعمال کرد. یکی از نگاشت‌ها به صورت زیر است.

$$\begin{aligned} y(n) &= F[x(n)] \\ &= X_{\max}\,\frac{\log\left[1+\mu\frac{|x(n)|}{X_{\max}}\right]}{\log[1+\mu]}\,.\sin[x(n)] \end{aligned} \tag{۱۳-۳}$$



شکل۲- رابطه ورودی – خروجی برای یک مشخصه $\mu - \text{law}$ (اقتباس از [۲]smith)

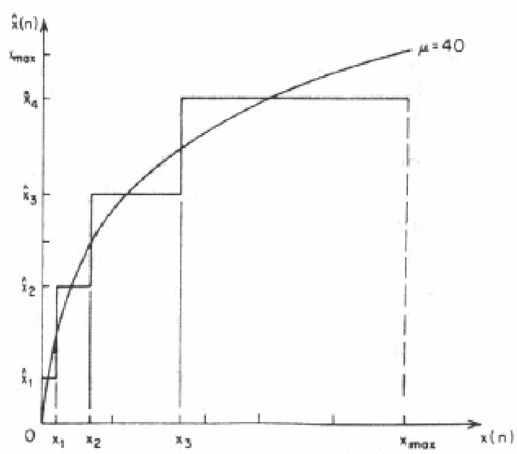
شکل۲، یک خانواده از منحنی های y(n) بر حسب x(n) را برای مقادیر متفاوت μ نشان می دهد. واضح است که با استفاده از تابع (۳-۱۳) دامنه های ورودی کوچک تقویت می شوند. شکل۳ توزیع سطوح کوانتیزاسیون را برای حالتی که μ=۴۰ و N=۸ است، نشان می دهد. اگر μ=۰ باشد، معادلهٔ (۳-۱۳) به معادلهٔ y(n)=x(n) خلاصه می شود، سطوح کوانتیزاسیون با فاصله‌های یکنواخت تقسیم شده اند، با این وجود بری مقادیر بزرگ μ و برای |x(n)| های بزرگ:

$$|y(n)| \approx X_{\max} \log\left|\frac{x(n)}{X_{\max}}\right|$$

$$\text{or} \qquad |x(n)| \approx X_{\max} 10^{\frac{|y(n)|}{X_{\max}}}$$

(۳-۱٤)

بنابراین به جز دامنه های بسیار کوچک، سطوح کوانتیزاسیون به طور نمایی با اندیس کوانتیزاسیون افزایش می یابند. این کوانتیزاسیون، کوانتیزاسیون $\mu - \text{law}$ نامیده می شود و اولین بار توسط smith ارایه شد [۲].



شکل۳- توزیع سطوح کوانتیزاسیون برای کوانتیزه کنندهٔ ۳ بیتی μ-law با μ=۴۰ از [۱]

با بکارگیری همان فرضیات که برای آنالیز کوانتیزاسیون یکنواخت استفاده شد، [۲]smith رابطه زیر برای نسبت سیگنال بهنویز کوانتیزه شده در کوانتیزه کنندهٔ $\mu - \text{law}$ بدست آورد.

$$SNR = 6B + 4.77 - 20\log_{10}[\ln(1+\mu)] - 10\log_{10}\left[1+\left[\frac{X_{max}}{\mu\ \sigma_x}\right]^2+\sqrt{2}\left[\frac{X_{max}}{\mu\ \sigma_x}\right]\right]$$
(۱۵-۳)

این معادله بستگی کمتر SNR به مقدار ($\frac{x_{max}}{\sigma_x}$) را که به تویع سیگنال بستگی دارد را در مقایسه با معادله (۳-۱۲) نشان می دهد.

مشاهده می‌شود که با افزایش μ، SNR به تغییرات ($\frac{x_{max}}{\sigma_x}$) کمتر بستگی پیدا می کند، یعنی با وجود اینکه ترم $-20\log10[\ln(1+\mu)]$ ، SNR را کاهش می دهد، محدوده ای از ($\frac{x_{max}}{\sigma_x}$) که در آن SNR ثابت است با μ افزایش می یابد.

بنابراین با استفاده از یک μ بزرگ، کارآیی کوانتیزه کننده به آمارگان سیگنال وابستگی کمتری پیدا می کند.

۲-۲- کد کردن پیشگویانه (Predictive Coding)

در یک شکل موج صوت معمولی، نمونه های متوالی بجز در گذارهای بین آواهای متفاوت، مقادیر مشابهی دارند. یک راه برای بهره گیری از این همبستگی استفاده از کد کردن به روش پیشگویی خطی است. ابتدا نمونه فعلی x(n) از روی ترکیب خطی نمونه های قبلی ساخته شده $\hat{x}(n-k)$ تخمین زده می شود تا

$$x_p(n) = \sum a_k\ \hat{x}(n-k)$$

سپس خطای بین مقدار نمونه اصلی و مقدار پیش بینی شده

$$d(n) = x(n) - x_p(n)$$

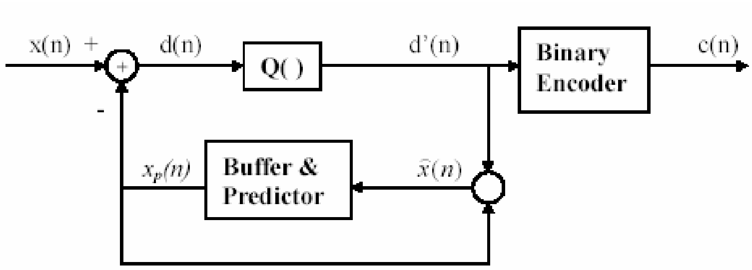
به d(n) کوانتیزه می شود و بوسیلهٔ کلمهٔ کد c(n)، کد می شود.

در دی کد کننده، ابتدا همان مقدار پیشگویی شده از روی نمونه های قبلی دی کد

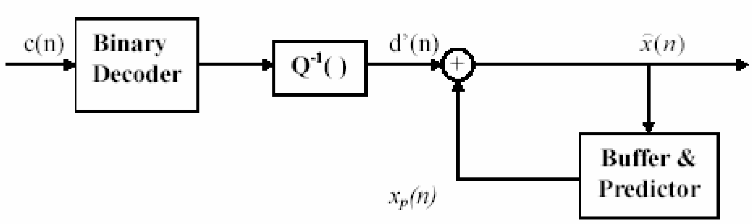
شده ساخته می شود. این مقدار سپس به مقدار خطای دی کد و کوانتیزه شده اضافه می شود تا مقدار کوانتیزه شده برای نمونه فعلی بدست آید. یعنی:

$$\hat{x}(n) = x_p(n) + \hat{e}(n).$$

بلوک دیاگرام کد کننده و دی کد کننده یک سیستم کد کنندهٔ پیشگو در شکل٤ نشان داده شده است. سیستم کد کننده پیشگو معمولاً به مدولاسیون کد شده سیگنال تفاضلی یا "DPCM" شناخته می شود. کلمهٔ «تفاضلی» به این موضوع اشاره می کند که سیگنال خطای پیشگویی کد می‌شود و "PCM" به طرح مدولاسیون اشاره می کند که در آن هر بیت کد شده یک سمبل است که بوسیله یک پالس (با دامنه صفر یا یک) نشان داده می شود. کوانتیزه کردن مستقیم یک نمونهٔ اولیه با طول ثابت در کد کردن، "PCM" نامیده می شود.



(a)



(b)

شکل٤- کد کردن پیشگویانه (الف) کد کننده (ب) دی کد کننده

۲-۲-۱- مدولاسیون دلتا

یک سیستم سادهٔ پیشگویانه، سیستم مدولاسیون دلتا (DM) است که در شکل٥ نشان داده شده است. در این سیستم، کوانتیزه کنندهٔ خطای پیشگویی فقط دو سطح دارد و طول پله «کوانتیزاسیون» ثابت است. سطح کوانتیزه مثبت با c(n)=۰ و منفی با c(n)=۱ مشخص می شود. بنابراین d(n) به صورت زیر تعریف می شود.

$$\begin{cases} \hat{d}(n) = \Delta & or \quad c(n) = 0 & if \quad d(n) \geq 0 \\ \hat{d}(n) = -\Delta & or \quad c(n) = 1 & if \quad d(n) < 0 \end{cases}$$

(۳-۱٦)

که از یک پیشگویی خطی درجه اول استفاده شده است، یعنی $x_p(n) = x(n-1)$ می توان از شکل٥-الف مشاهده کرد که معمولاً x(n) در معادله تفاضلی زیر صدق می کند.

$$\hat{x}(n) = \alpha\, \hat{x}(n-1) + \hat{d}(n)$$

(۳-۱۷)

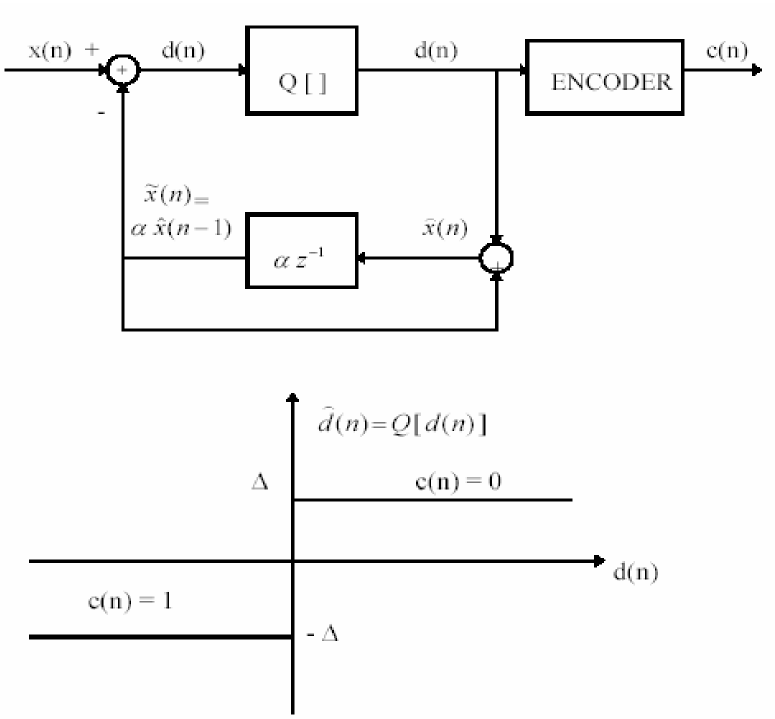
با $\alpha = 1$، این معادله، معادل دیجیتالی انتگرال است. همچنین باید دقت کرد که ورودی کوانتیزه کننده

$$d(n) = x(n) - \hat{x}(n-1) = x(n) - x(n-1) - e(n-1)$$

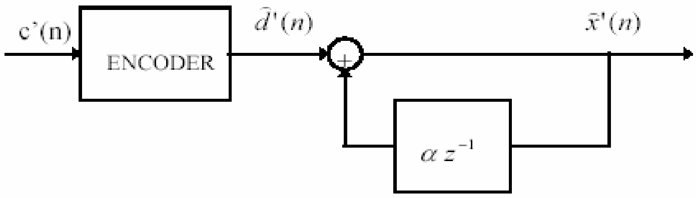
(۳-۱۸)

می باشد. بنابراین به جز خطای کوانتیزه شده در $\hat{x}(n-1)$، d(n) یک تفاضلی برگشتی درجه اول از x(n) است که می تواند به عنوان تقریب دیجیتالی برای مشتقات ورودی در نظر گرفته شود، معکوس پروسهٔ انتگرال دیجیتالی.

از آنجا که خطای کوانتیزاسیون فقط دو سطح دارد، مدولاسیون دلتا دارای نرخ بیتی برابر ۱bit/sample است. اگر به دنباله ١٦bits/sample اعمال شود، آنگاه به نرخ فشرده سازی (CR) برابر ١٦ می رسد.



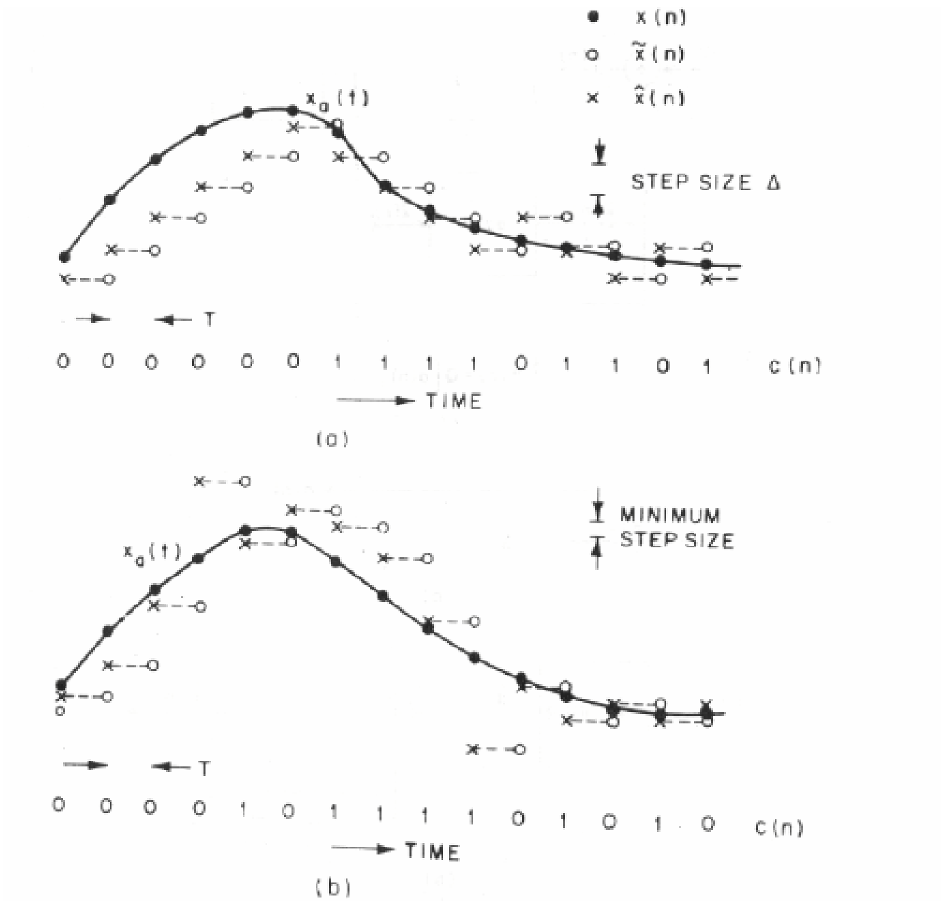
(a)



(b)

شکل۵- بلوک دیاگرام سیستم مدولاسیون دلتا الف) کد کننده ب) دی کد کننده

برای اینکه مدولاسیون دلتا به خوبی کار کند، اندازه پله باید طوری انتخاب شود که تغییرات سیگنال را دنبال کند. تحقق این امر مشکل است زیرا مشخصات سیگنال از یک tone به tone دیگر تغییر می کند. شکل۶-الف پروسه کوانتیزاسیون مدولاسیون دلتا را با طول پله متناسب و دقیق نشان می دهد. می توان مشاهده کرد که طول پله در ابتدا بسیار کوچک است که باعث می‌شود سیگنال کوانتیزه شده زیر دامنه سیگنال اصلی آهسته تر حرکت کند. از طرف دیگر اگر طول پله را خیلی بزرگ بگیریم، باعث می شود که سیگنال کوانتیزه شده در حول و حوش سیگنال اصلی نوسان کند. برای کارآیی بهتر، طول پله باید به طور وفقی باشد که موضوع بخش آینده است.



شکل٦-نمایش مدولاسیون دلتا الف) استفاده از یک پله با طول ثابت ب) استفاده از طول پله وفقی

۲-۲-۲- مدولاسیون دلتای وفقی

طرحهای مدولاسیون دلتای وفقی (ADM) متعددی پیشنهاد شده اند. بیشتر این طرحها از نوع برگشتی هستند که در آنها طول پله برای کوانتیزه کنندهٔ دو سطحی بر مبنای کلمات کد در خروجی بهینه می شود. سیستمی که ما در زیر پیشنهاد کرده ایم بوسیلهٔ Jayant[۳] طراحی شده است. طول پله در الگوریتم Jayant از قانون زیر پیروی می کند.

$$\Delta(n) = M\,\Delta(n-1)$$ (۳-۱۹-الف)

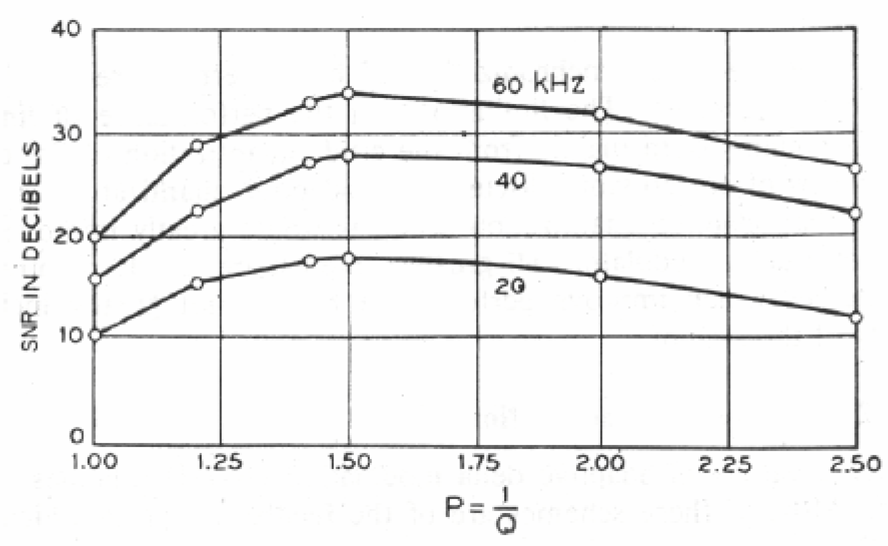
$$\Delta_{\min} \le \Delta(n) \le \Delta_{\max}$$ (۳-۱۹-ب)

الگوریتم برای تعیین طول پله به صورت زیر است.

$$\begin{aligned} M = P > 1 \qquad & \text{if } c(n) = c(n-1) \\ M = Q < 1 \qquad & \text{if } c(n) \ne c(n-1) \end{aligned}$$ (۳-۲۰)

شکل٦-ب- نشان می دهد که چگونه شکل موج شکل٦-الف می تواند توسط یک مدولاسیون دلتای وفقی که در رابطه (۳-۱۸) و (۳-۲۰) بیان شد کوانتیزه شود. برای راحتی، پارامترهای سیستم در $p = ۲$ و $\alpha = 1$ تنظیم می شوند و حداقل طول پله در شکل نشان داده شده است. می توان مشاهده کرد که نواحی با شیب مثبت زیاد هنوز یک دنباله از صفر تولید می کنند اما در این حالت طول پله آنقدر افزایش می یابد تا ازدیاد شیب شکل موج را دنبال کند. نواحی دانه دانه ای در سمت راست شکل دوباره بوسیله یک دنباله

از صفر و یک های متناسب کوانتیزه می‌شوند، اما در این حالت طول پله سریعاً به مقدار حداقل (Δ_{min}) کاهش می یابد و تا وقتی که شیب کم باشد در این مقدار می ماند.



شکل۷- نسبت های سیگنال به نویز از یک مدولاتور دلتای وفقی بر حسب توابع p

شکل۷، نتایج شبیه سازی را برای سیگنال گفتار با PQ = ۱ برای سه نرخ نمونه برداری متفاوت نشان می دهد.

مشاهده می شود که حداکثر SNR برای P = ۱/۵ بدست می آید، با این وجود، قلهٔ منحنی بسیار پهن است و SNR چند dB بالاتر و پایین تر از مقدار حداکثر برای ۱/۲۵<p<۲ قرار دارد. توجه کنید برای اینکه مدولاسیون دلتا خوب کار کند باید نمونه برداری از سیگنال در فرکانسی بالاتر از آنچه تئوری نایکوئیست تحمیل می کند، انجام شود تا تغییرات بین نمونه های متوالی کوچک باشد. این پدیده در حقیقت مصالحه بین دقت نمونه برداری و دقت دامنه را آشکار می کند. یعنی برای کاهش دقت دامنه (یک بیت در مدولاسیون دلتا) باید دقت نمونه برداری را افزایش داد.

در این آزمایش شما با مدولاسیون دلتا در حالت طول پله ثابت و وفقی کار خواهید کرد.

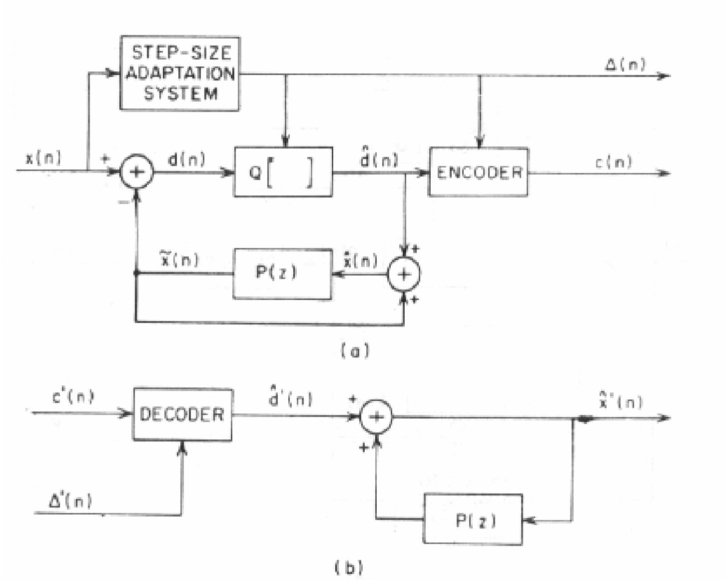
۲-۲-۳- DPCM های مرتبه بالاتر

مدولاتورهای دلتا، همانطور که در بخش قبلی بیان شد، می توانند سیستمهای DPCM یک بیتی نامیده شوند. به طور کلی، می توان از بیشتر از یک نمونه قبلی برای تخمین نمونه فعلی استفاده کرد. همچنین، می توان از یک کوانتیزه کننده با بیشتر از دو سطح استفاده کرد. برای درک بهتر از چگونگی تعیین ضرایب پیشگو و طراحی کوانتیزه کننده بهینه به مرجع [۱] مراجعه کنید. عموماً، DPCM برای سیستمهای کوانتیزه تفاضلی که در آن کوانتیزه کننده بیشتر از دو سطح دارد، معکوس می شود. سیستمهای DPCM با پیشگوهای ثابت می توانند از ٤ تا ۱۱ dB بهبود در کوانتیزاسیون مستقیم (PCM) ایجاد کنند. بیشترین بهبود در مرحله تغییر از حالت بدون پیشگویی به حالت پیشگویی درجه اول رخ می دهد. این بهبود در گذر به پیشگوهای درجه ٤ یا ۵ کمتر محسوس می‌باشد. در گفتار از پیشگوهای با درجه بالاتر از ۱۰ استفاده می شود، زیرا سیگنال گفتار با درجات بالاتر بهتر می تواند مدل شود. بهره SNR نشان می دهد که در یک سیستم DPCM، بدست آوردن SNR خواسته شده با استفاده از بیت های کمتر از بیت های موردنیاز، هنگامی که همان کوانتیزه کننده به طور مستقیم روی سیگنال گفتار اثر می کند، عملی است. به یادآورید که هنگام کوانتیزه کردن مستقیم یک سیگنال، هر بیت اضافی به بهرهٔ ٦dB منجر می شد، بنابراین اگر سیستم DPCM بتواند به بهره پیشگویی ٦dB برسد به این معنی است که یک بیت کمتر نسبت به حالتی که سیستم PCM وجود داشته باشد، موردنیاز است تا به همان کیفیت از سیگنال برسد.

ADPCM -۴-۲-۲

دو طرح عمده برای DPCM وفقی یا ADPCM وجود دارد. یکی از آنها DPCM با کوانتیزه وفقی و دیگری DPCM با پیشگویی وفقی است.

برای DPCM با کوانتیزه وفقی، طول پله کوانتیزه کننده متناسب با واریانس ورودی کوانتیزه کننده تغییر می کند. با این وجود، از آنجایی که سیگنال تفاضل d(n) متناسب با ورودی است، معقول است که تنظیم طول پله از روی سیگنال ورودی x(n) انجام شود که در شکل۸ نشان داده شده است. رویه های وفقی متعددی برای تنظیم طول پله در گذشته ارائه شده اند. نتایج نشان می دهد که این قبیل رویه های وفقی می توانند در حدود ۵ dB در SNR نسبت به حالت غیر وفقی μ – law در PCM بهبود ایجاد کنند. این بهبود می تواند با ۶ dB بهبود که از وضعیت تفاضلی با پیشگویی ثابت بدست می آید، ترکیب شده و ADPCM با پیشگویی وفقی رو به جلو، بهبود SNRای برابر ۱۰-۱۱dB نسبت به PCM با همان تعداد سطوح، نتیجه دهد.



شکل۸- سیستم ADPCM با کوانتیزه وفقی رو به جلو الف) کد کننده ب) دی کد کننده

برای DPCM با پیشگویی وفقی، ضرایب پیشگویی کننده بستگی به زمان دارند، بنابراین مقادیر پیشگویی شده به صورت زیر هستند.

$$\tilde{x}(n)=\sum_{k=1}^{p}\alpha_k(n)\,\hat{x}(n-k)$$

(۲۱-۳)

در تطبیق ضرایب پیشگویی $\alpha_k(n)$، معمول است که فرض کنیم خصوصیات آماری سیگنال در طول یک بازه کوتاه زمانی ثابت می مانند. ضرایب پیشگو به گونه ای انتخاب می شوند تا میانگین مربع خطای پیشگویی در هر پنجره کوچک زمانی حداقل شود. برای آشنایی بیشتر با نحوهٔ انتخاب بهینهٔ ضرایب پیشگویی خطی به مرجع [1] مراجعه کنید.

۳-۳- استانداردهای کد کردن گفتار

استانداردهای جهانی متعددی برای کد کردن سیگنالهای گفتار وجود دارند. تعدادی از این استانداردها در لیست پایین آمده اند. به جز استاندارد G.۷۱۱ همگن از نوعی ADPCM استفاده می کنند.

1) CCITT G. 711 (A-LAW and μ-LAW)

2) CCITT G. 721 (ADPCM at 32 kbits/ sec)

3) CCITT G. 723 (CELP Based 5.3 and 6.3 kbits/sec)

4) GSM 06.10 is speech encoding used in Europe (13 kbits/sec)

5) U.S. Federal Standard :

۳- آزمایشات

۱) برنامه MATLAB موجود در Appendix، "demo-quant," کوانتیزاسیون یکنواختی روی سیگنال صوت انجام می دهد. برنامه را با دقت بخوانید تا متوجه شوید چگونه کار می کند. برنامه را برای یک فایل گفتار ضبط شده در ۸ $\frac{\text{bits}}{\text{sample}}$ و یک فایل موسیقی ضبط شده در ۱۶ $\frac{\text{bits}}{\text{sample}}$ مقایسه کنید. برای هر مورد، اغتشاش (Distortion) در هر شکل موج، کیفیت صدا را با تغییر سطوح کوانتیازسیون (N) مقایسه کنید. برای هر مورد (گفتار و موسیقی)، N موردنیاز برای داشتن یک صدای با کیفیت خوب کدام است؟ شکلهای تولید شده بوسیله انتخاب های متفاوت را پرینت کنید.

۲) برنامه سه نمونه بالا را برای کوانتزاسیون $\mu - \text{law}$ (به جای کوانتیزاسیون یکنواخت) تکرار کنید. شما باید قادر به تنظیم پارامتر μ علاوه بر تعداد سطوح کوانتزاسیون، N، باشید. نتایج بدست آمده با μ و N های متفاوت را مقایسه کنید. برای یک μ انتخاب شده، تعداد بیتهای لازم برای بدست آوردن کیفیت قابل قبولی از گفتار و موسیقی چه می باشد؟ این مقادیر را با بیت های موردنیاز در کوانتیزاسیون یکنواخت، مقایسه کنید.

راهنمایی: شما باید $\mu - \text{law}$ را به مقدار نمونه اولیه اعمال کنید، مقدار تبدیل یافته را با استفاده از کوانتیزاسیون یکنواخت، کوانتیزه کنید، سپس عکس $\mu - \text{law}$ را به مقدار کوانتیزه شده اعمال کنید تا مقدار کوانتیزه شده در فضای سیگنال اولیه بدست آید. برای عکس $\mu - \text{law}$ گرفتن، شما احتیاج به تعیین x(n) از y(n) دارید (معادلهٔ ۱۳-۳).

۳- برنامهٔ مطلب "sinadm.m" و "sindm.m" را در Appendix بخوانید که DM و ADM را برای یک سیگنال سینوسی پیاده سازی می کند. نتایج بدست آمده با dmax, dmin, xmean, p, Q های مختلف را مقایسه کنید. اثرات تغییر Q به مقادیر بسیار کوچک و بسیار بزرگ را بررسی کنید. مشابه آن اثرات P را وقتی بسیار کوچک یا بزرگ است مورد بررسی قرار دهید. کدام دسته پارامترها بهترین نتایج را در این حالت می دهد؟ شکل های تولید شده با انتخاب های متفاوت را پرینت کنید.

٤- "sindm.m" را به گونه ای تغییر دهید که سیگنال صوت را پردازش کند. برنامه باید قادر به:

الف) خواندن یک فایل ورودی به فرمت wav.

ب) اعمال مدولاسیون دلتا به هر نمونه

ج) بازسازی نمونه بعد از کوانتزاسیون

د) نمایش شکل موج سیگنال های اصلی و کوانتیزه شده: تا به شما هر تغییر کوچکی را در دامنه نمونه ها نشان دهد. شما باید یک قسمت کوچک از شکل موج را در هنگام نمایش در نظر بگیرید تا نمونه های مجزا را واضح ببینید.

هـ) فایل بازسازی شده را به فرمت فایل wav. ذخیره کنید.

و) فایل اولیه و بازسازی شده را پخش کنید تا کیفیت صوت آنها را مقایسه کنید.

۵) برنامه را به سیگنالهای گفتار نمونه برداری شده در ۱۱KHz و ۲۲KHz و ٤٤KHz و کوانتیزه شده به ۸ بیت اعمال کنید. برای هر فایل ورودی، DM را با استفاده از برنامهٔ MATLAB اعمال کنید. شما باید طول پله را به گونه ای تنظیم کنید که بهترین کیفیت

ممکن در هر مورد بدست آید. سعی کنید از هیستوگرام اختلاف نمونه ها برای تعیین طول پله استفاده کنید. کیفیت صدا و تغییرات دامنه را در هر مورد مشاهده کنید. در چه فرکانس نمونه برداری، سیگنال فشرده شدۀ DM کیفیت قابل مقایسه ای با سیگنال اصلی ۸ بیتی و ۱۱KHz فراهم می کند؟ برای هر نرخ نمونه برداری فرکانسهای داده اصلی و داده بعد از DM را برای هر نرخ نمونه برداری بدست آورید (دلخواه).

توجه کنید که با برنامه MATLAB، هر چند خطای پیشگویی به $1\,\frac{\text{bit}}{\text{sample}}$ کوانتیزه می شود، سیگنال بازسازی شده دقت double را دارد و هنگامی که به یک فایل wav. تبدیل می شود، هر نمونه ۸ یا ۱٦ بیت جا می گیرد. بنابراین اندازه فایل wav. که شما ساخته اید، نمایش درستی از اندازه فایل فشرده شده حقیقی نیست. یک برنامه فشرده سازی واقعی خطای پیشگویی را با $1\,\frac{\text{bit}}{\text{sample}}$ ذخیره می کند.

(٤) و (٥)و (٦) را با استفده از ADM تکرار کنید. در این حالت شما باید پارامترهای P و xmean و dmin و dmax را به طور مناسب انتخاب کنید. سعی کنید که از هیستوگرام اختلاف نمونه ها برای تعیین این پارامترها استفاده کنید. کیفیت ADM را با DM در کوانتیزاسیون یکنواخت مقایسه کنید.

۷) اختیاری : یک برنامه MATLAB برای تحقق فشرده سازی" $ADM + \mu - law$ " در یک سیگنال ورودی داده شده بنویسید. آنرا به سیگنال موسیقی که قبلاً استفاده کرده اید اعمال کنید. آیا شما به بیت های کمتری برای رسیدن به همان کیفیت احتیاج دارید؟

راهنمایی: شما احتیاج به تولید خطای پیشگویی در هر نمونه دارید. $\mu - \text{law}$ را اعمال کنید. مقدار تبدیل شده را با استفاده از الگوریتم ADM که در آن مقدار طول پله تعیین می شود کوانتیزه کنید. سپس مقدار کوانتیزه شده را با اعمال معکوس $\mu - \text{law}$ به حوزه اصلی برگردانید. ما سرانجام این مقدار بازسازی شده را به مقدار پیشگویی شده اضافه می کنیم.

۵- گزارش

برنامه های MATLAB و شکل ها (plots) را تحویل دهید. هر پدیده ای که مشاهده کرده اید. توضیح دهید. روی کیفیت صدا با تنظیم پارامترهای متفاوت نظر دهید و سوالات خواسته شده در آزمایش را پاسخ دهید.

٦- مراجع

[۱]. L.R.Rabiner and R.W.Schafer, *Digital Processing of Speech Signals*, Prentice Hall ۱۹۷۸

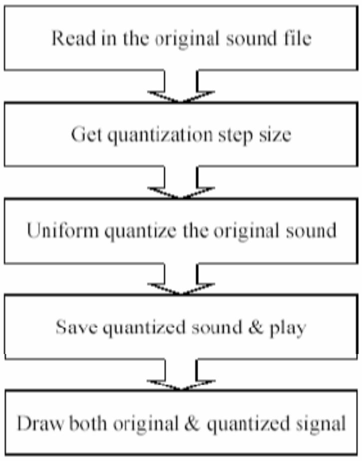
[۲]. B.Smith, "Instantaneous Companding of Quantized Signals", *Bell System Tech. J.*, Vol.۳٦, No.۳, pp.٦۵۳-۷۰۹, May ۱۹۵۷.

[۳]. N.S.Jayant, "Adaptive Quantization with a One Word Memory", *Bell System Tech. J.,* pp. ۱۱۱۹-۱۱٤٤, September ۱۹۷۳.

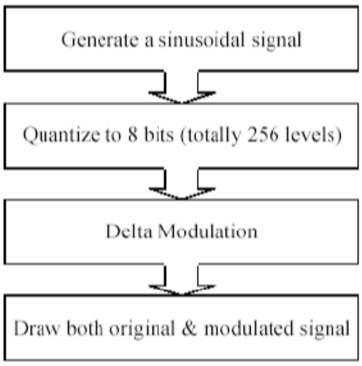
[٤]. Guido van. Rossum, "FAQ: Audio File Formats", http://www.cis.ohio_state.edu.

APPENDIX Sample Matlab Programs

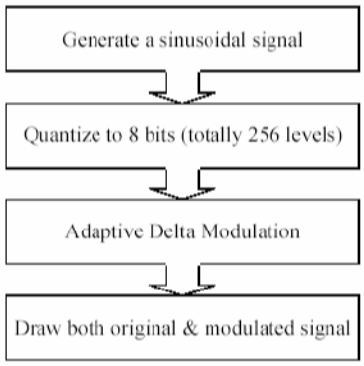
1. Flow charts for sample programs.
2. Matlab scripts for quantization, Delta Modulation(DM) and Adaptive Delta Modulation(ADM).



a. Block diagram for quantization



b. Block diagram for delta modulation



c. Block diagram for adaptive delta modulation

****

* MATLAB program for applying uniform quantization on a sound signal *

****

```
%quantizing a sound signal
function []=demo_quant(inname,outname, N);
if nargin < 3
   disp('Usage: sampl_quant(inname,outname, N)');
   disp('inname: input .wav file name');
   disp('outname: output .wav file name');
   disp('N: quantization level, N>1');
   return;
end;

%read in input  signal
[x,fs,N0]=wavread(inname);
xmin=min(x); xmax=max(x);
Q=(xmax-xmin)/N;
disp('N0,xmin,xmax,N,Q');
disp([N0,xmin,xmax,N,Q]);

%apply uniform quantization on each sample
xq=sign(x).*(floor((abs(x)+Q/2)/Q)*Q);

%compare sound quality
wavwrite(xq,fs,N0,outname);
sound(x,fs);
pause;
sound(xq,fs);

%plot waveforms over the entire period
t=1:length(x);
figure; plot(t,x,'r:');
hold on; plot(t,xq,'b-');
axis tight; grid on;

%plot waveform over a selected period
t=5000:5100;
figure; plot(t,x(5000:5100),'r:');
hold on; plot(t,xq(5000:5100),'b-');
axis tight; grid on;
```

```
***********************************************************
* MATLAB program for Delta Modulation on a sinusoidal signal*
***********************************************************
function [t,x,xx]=sindm(Q, xmean);
if nargin < 1
   disp('Usage: sindm(Q, xmean)');
   disp('Q: stepsize');
   disp('xmean: mean value of the signal');
   return;
end;


% construct a quantized sinusoid wave signal using 8-bit quantizer
% { ranged at (0,255) } for sampling time interval t=(0,1).

% given a sampling frequency.
fs=33;
t=[0:1/fs:1];
L=length(t);
f=2;
x=(sin(2*pi*f*t)+1.0)/2*255;
x=round(x); %the round operation essentially quantizes to 8 bits,
because the range of x is 0-255.

% Delta Modulation
D=Q*ones(L); % fixed stepsize=30
%xmean=128;

% given the initial condition.
d(1)=x(1); %difference signal
c(1) =0; %coded signal
dd(1)=D(1); %quantized difference signal
xx(1)=xmean+dd(1); %reconstructed signal
%sindm.m
% calculate the delta modulation.
for i = 2:L,
   d(i)=x(i)-xx(i-1);
      if d(i) > 0
         c(i) = 0;
         dd(i)=D(i);
      else
         c(i) = 1;
         dd(i)=(-1)*D(i);
      end
   xx(i)=xx(i-1)+dd(i);
end

figure;
t1=[0:1/(100*fs):1];
x1=(sin(2*pi*f*t1)+1.0)/2*255;
plot(t1,x1,t,x,'*',t,xx,'x',t(2:L),xx(1:(L-1)),'o');
title('Illustration of the linear delta modulation');
legend('original signal', 'original sample', ...
      'reconstructed value','predicted value');
```

```
*******************************************************************
* MATLAB Script file for Adaptive Delta Modulation with sin as input signal *
*******************************************************************
function [t,x,xx]=sinadm(P,xmean,dmin,dmax)

if nargin < 1
   disp('Usage: sinadm(P,xmean,dmin,dmax)');
   disp('P: Adaptation Parameter, 1<=P<=3');
   disp('xmean: mean value of x');
   disp('dmin, dmax: min and max of stepsize');
   return;
end;

% given a sampling frequency.
fs=33;

% construct a quantized sinusoid wave signal using 8-bit quantizer
% { ranged at (0,255) } for sampling time interval t=(0,1).
t=[0:1/fs:1];
L=length(t);
f=2;
x=(sin(2*pi*f*t)+1.0)/2*255;
x=round(x);

% Adaptive Delta Modulation
%P=1.8;
%dmin=2;
%dmax=40;
%xmean=128;
Q=1.0/P;

% given the initial condition.
d(1)=x(1);
c(1) =0;
dd(1)=(dmin+dmax)/2;
xx(1)=xmean+dd(1);

% calculate the adaptive delta modulation.
for i = 2:L,
   d(i)=x(i)-xx(i-1);
      if d(i) > 0
         c(i) = 0;
      else
         c(i) = 1;
      end
      if c(i) == c(i-1)
         M=P;
      else
         M=Q;
      end
   dd(i)=M*dd(i-1);
   %dd(i)=round(dd(i));
      if dd(i) < dmin
         dd(i)=dmin;
      elseif dd(i) > dmax
         dd(i)=dmax;
```

```
        end
        if c(i) == 0
           xx(i)=xx(i-1)+dd(i);
        elseif c(i) == 1
           xx(i)=xx(i-1)-dd(i);
        end
end

% graph of the fixed stepsize delta modulation of a sinusoid wave
signal.
figure;
t1=[0:1/(100*fs):1];
x1=(sin(2*pi*f*t1)+1.0)/2*255;
plot(t1,x1,'-',t,x,'*',t,xx,'x',t(2:L),xx(1:(L-1)),'o');
title('Illustration of the adaptive delta modulation');
legend('original signal', 'original sample', ...
   'reconstructed value','predicted value');
```