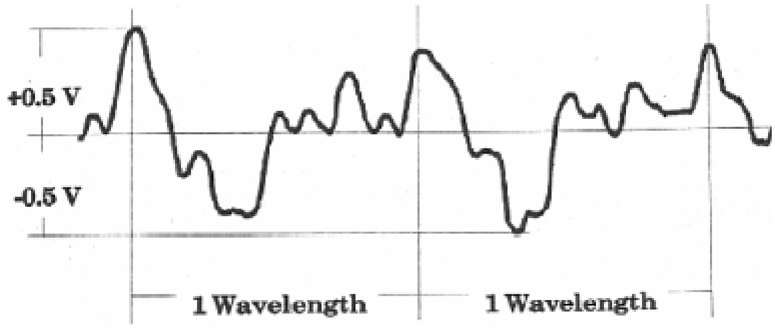
باسمه تعالی

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

تکلیف شماره ۱: دیجیتال کردن صوت و تبدیل نرخ نمونه برداری

۱- مقدمه

در یک میکروفون، امواج فشار صدای فیزیکی به سیگنالهای الکتریکی متناظر با خود، به وسیله مبدلهای آکوستیکی نظیر میکروفون یا Phonograph cartridge تبدیل می شوند. خروجی الکتریکی مبدل یک سیگنال آنالوگ نامیده می‌شود، زیرا سیگنال الکتریکی مشابه الگوی فشار موج صوتی است که آن را بوجود آورده است. سیگنالهای صوت به صورت الگوهای موج دوبعدی می باشند که محور y نشان دهندهٔ شدت یا دامنه و محور x نشان دهندهٔ مسیر زمان هستند، شکل ۱، شکل موج آنالوگ از یک سری موجهای صوت از یک chime را نشان می دهد. این سری موجها به وسیله میکروفون و تقویت کننده به ولتاژ آنالوگ با حداکثر دامنهٔ ± 0.5 ولت (دامنهٔ قله به قله یا V_{PP}) تبدیل شده اند.



شکل۱- شکل موج معمول صوت

فرکانس یک موج به وسیله زمان سپری شده بین تکرارها تعیین می‌شود که طول موج نامیده می‌شود. بیشتر موجهای صوت دقیقاً تکرار نمی شوند اما می توان یک الگوی مشخص در شکل موجی که توسط بیشتر سازهای موسیقی ایجاد می‌شود، مشاهده کرد. طول موج یک صوت الکتریکی، λ، در مقیاس میلی ثانیه یا میکروثانیه بیان می شود. فرکانس،F، که با واحد هرتز اندازه گیری می‌شود (تعداد دور در هر ثانیه) معکوس λ می باشد، یعنی $F = \frac{1}{\lambda}$.

صدای بشر یا صداهای تولید شده به وسیله سازهای موسیقی می‌توانند به یک موج پایه و موجهای متعدد الحاقی دیگر تقسیم شوند. موجهای الحاقی که به موج پایه اعمال شده اند، overtone نامیده می شوند. Overtonها موجهای فرکانس بالاترمی باشند و با فرکانس هایی که ضرایبی از موج پایه می باشند (هارمونیک ها) به صدا، مشخصات یک صوت بشری یا صوت سازهای موسیقی را می بخشند. هنگامی که یک سیگنال صوت به سیگنالهای دیجیتال تبدیل می‌شود، نرخ نمونه برداری موردنیاز بستگی به فرکانسهای overtoneهای موجود در سیگنال دارد.

در این آزمایش شما امکان بازی با سیگنالهای نمونه برداری شده در نرخ های متفاوت نمونه‌برداری را دارید، که کیفیت های مختلفی از صوت را عرضه می کنند.

۲- تئوری

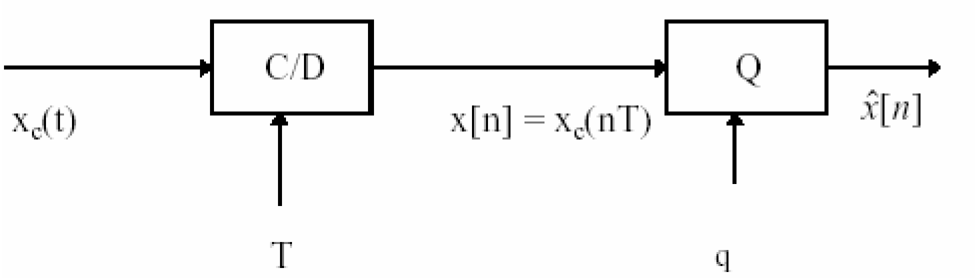
۲-۱- نمونه برداری یک سیگنال پیوسته و تئوری نمونه برداری نایکوئیست

سیگنالهای صوت از نوع سیگنالهای پیوسته (آنالوگ) هستند که به تدریج با نقصان یافتن منبع صدا، افت دامنه پیدا می‌کنند. از سوی دیگر، کامپیوترها، داده های خود را به صورت دیجیتال ذخیره می کنند: یک رشته stream از بیتهای صفر و یک. داده های دیجیتال طبیعتاً گسسته هستند زیرا مقدار "۰" یا "۱" دادهٔ دیجیتال فقط در یک لحظهٔ مشخص معتبر می باشد. بنابراین، سیگنال صوت آنالوگ که پیوسته است باید به فرم دیجیتالی ناپیوسته تبدیل شود تا کامپیوتر توانایی ذخیره یا پردازش صوت را داشته باشد. البته دادهٔ دیجیتال دوباره باید به فرم آنالوگ تبدیل شود تا از طریق یک سیستم صوتی قابل شنیدن باشد. تبدیل دو طرفه بین سیگنالهای آنالوگ و دیجیتال، عملیات اولیه تمام کارتهای adapter و کارتهای صدا می باشد.

۲-۱-۱- نمونه برداری تناوبی و تبدیل آنالوگ به دیجیتال

روش معمول نمایش یک سیگنال زمان – گسسته از یک سیگنال زمان – پیوسته، نمونه برداری متناوب (پریودیک) است که در آن یک دنباله از نمونه های $x[n]$ از یک سیگنال زمان پیوسته $x_c(t)$ مطابق رابطه زیر بدست می آید.

$$x[n] = x_c(nT), \qquad -\infty < n < \infty . \tag{۱-۲}$$



شکل۲- یک مبدل آنالوگ به دیجیتال (A/D) ایده آل

در رابطهٔ (۲-۱)، T تناوب نمونه برداری و معکوس آن، $f_s = \frac{1}{T}$، فرکانس نمونه برداری بر حسب نمونه بر ثانیه که معمولاً بر حسب هرتز (Hz) نمایش داده می شود، می باشند. ما یک سیستم را که رابطهٔ (۲-۱) را به عنوان یک مبدل ایده آل پیوسته – به – گسسته (C/D) عملی می کند در شکل ۲ نشان داده ایم. برای ذخیرهٔ مقادیر نمونه برداری شده توسط کامپیوتر با دقت محدود، مقادیر پیوسته باید به یک سری مقادیر از پیش تعیین شده کوانتیزه شوند. عملیات نمونه برداری و کوانتیزاسیون دقیقاً همان عملیاتی است که در مبدل آنالوگ – به – دیجیتال، یا دیجیتال – به – آنالوگ، به صورت برعکس، صورت می گیرد. بیشتر کارتهای صدا قابلیت ذخیره صدا را هم به صورت ۸ بیتی و هم ۱۶ بیتی، برای کیفیت های بالاتر صوتی دارند.

۲-۱-۲- تئوری نمونه برداری نایکوئیست

تئوری نمونه برداری به ما می گوید که چه اندازه نمونه برداری ما می تواند سریع باشد تا نمایش بهتری از سیگنال اولیه داشته باشیم. می دانیم که اگر یک سیگنال تغییرات خیلی سریع داشته باشد، ما هم باید در فاصله های نزدیکتری نمونه برداری کنیم تا هیچ تغییر میانی را از دست ندهیم. یک مثال خوب، نمایش سهام بورس بر حسب نمایش وضع هوا است. از آنجا که تغییرات بورس بسیار

سریع است، به طور معمول باید هرچند دقیقه یکبار اعلام شود، از طرف دیگر، دربارهٔ وضع هوا، نمایش این تغییرات در هر ساعت کافی خواهد بود.

حالا، نگاهی به تئوری تناوب T می اندازیم که چه اندازه دقیق باید آن را تعیین کرد.

تئوری نمونه برداری نایکوئیست: در نظر بگیرید که $x_c(t)$ یک سیگنال با پهنای باند محدود و $X_c(j\Omega)$ تبدیل فوریه آن است که شرط زیر را برآورده می کند.

$$X_c(j\Omega)=0 \qquad for\,|\Omega|>\Omega_N \;. \tag{۲-۲}$$

پس $x_c(t)$ منحصراً توسط نمونه های $x[n]=x_c(nT)$ $n=0,\pm1,\pm2,\ldots,$ بیان می شود به شرطی که تناوب نمونه برداری آن یا فرکانس نمونه برداری Ω_s شرط زیر را برآورده کند.

$$\Omega_s=\frac{2\pi}{T}>2\Omega_N \;. \tag{۳-۲}$$

نتیجه فوق ابتدا توسط نایکوئیست بدست آمد که به نام تئوری نمونه برداری نایکوئیست مشهور شد. فرکانس $2\Omega_N$ که باید از فرکانس نمونه برداری کوچکتر باشد، نرخ نمونه برداری نایکوئیست نامیده می‌شود. برای اثبات تئوری فوق، $X(e^{jw})$ تبدیل فوریه زمان – گسسته دنبالهٔ $x[n]$ را بر حسب $X_c(j\Omega)$، تبدیل فوریه پیوسته $x_c(t)$، بدست می آوریم.

به همین منظور سیگنال قطار ضربه زیر را در نظر می گیریم.

$$\begin{aligned} x_s(t) &= x_c(t)\cdot\sum_{n=-\infty}^{\infty}\delta(t-nT) \\ &= \sum_{n=-\infty}^{\infty} x_c(nT)\,\delta(t-nT) \;. \end{aligned} \tag{۴-۲}$$

می توان نشان داد که تبدیل فوریه $x_s(t)$ به صورت زیر می باشد:

$$X_s(j\Omega)=\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(j\Omega-k\Omega_s) \;, \tag{۵-۲}$$

با استفاده از تعریف

$$X(e^{j\omega})=\sum_{n=-\infty}^{\infty} x[n]e^{-jn\omega} \tag{۶-۲}$$

$$X_s(j\Omega)=\int_{-\infty}^{\infty} x_s(t)\,e^{-j\Omega t}\,dt=\sum_{n=-\infty}^{\infty} x_c(nT)\,e^{-jn\Omega T} \;, \tag{۷-۲}$$

مشاهده می شود

$$X(e^{j\omega})=X_s(j\Omega)\big|_{\Omega=\frac{\omega}{T}} \;. \tag{۸-۲}$$

از روابط (۲-۵) و (۲-۸) نتیجه می گیریم که

$$X(e^{j\omega})=\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{T}-j\frac{2\pi k}{T}\right) \;. \tag{۹-۲}$$

از رابطه (۲-۹) مشاهده می شود که $X(e^{jw})$ مجموع ترمهای مقیاس بندی شده و شیفت یافتهٔ $X_c(j\Omega)$ می باشد. مقیاس فرکانس توسط $\Omega = \frac{w}{T}$ تعیین می شود، در حالی که شیفت ها برابر با ضرایب فرکانس نمونه برداری $\Omega s = \frac{2\pi}{T}$ می باشند.

۲-۱-۳- بازسازی یک سیگنال باند محدود از نمونه هایش (تبدیل دیجیتال به آنالوگ)

از شکل۵، اگر سیگنال اولیه باند محدود باشد، یعنی $\Omega > \Omega_N$ برای $x_c(j\Omega) = 0$، که $\Omega_N \leq \frac{\Omega s}{2}$، می توان $X_c(j\Omega)$ را دوباره از $X(jw)$ بدست آورد. به طور دقیقتر ابتدا می توانیم سیگنال قطار ضربه $x_s(t)$ را بوجود آوریم.

$$x_s(t) = \sum_{n=-\infty}^{\infty} x[n]\delta(t-nT) \quad .$$

(۱۰-۲)

سپس می توانیم یک فیلتر ایده آل پایین گذر را با فرکانس قطع $\Omega_c = \frac{\Omega_s}{2} = \frac{\pi}{T}$ به $x_s(t)$ اعمال کنیم.

$$H_r(j\Omega) = \begin{cases} T & |\Omega| < \Omega_c \\ 0 & otherwise \end{cases}$$

(۱۱-۲)

اگر $X_c(j\Omega)$ پهنای باند محدود باشد آنگاه سیگنال فیلتر شدهٔ تبدیل فوریه آن دقیقاً برابر $X_c(j\Omega)$ است. یعنی:

در حوزه زمان، فیلتر بازسازی ایده آل $h_r(t)$ به صورت زیر می باشد.

$$h_r(t) = \frac{\sin \pi t / T}{\pi t / T} \quad .$$

(۱۲-۲)

سیگنال بازسازی شده به صورت زیر است

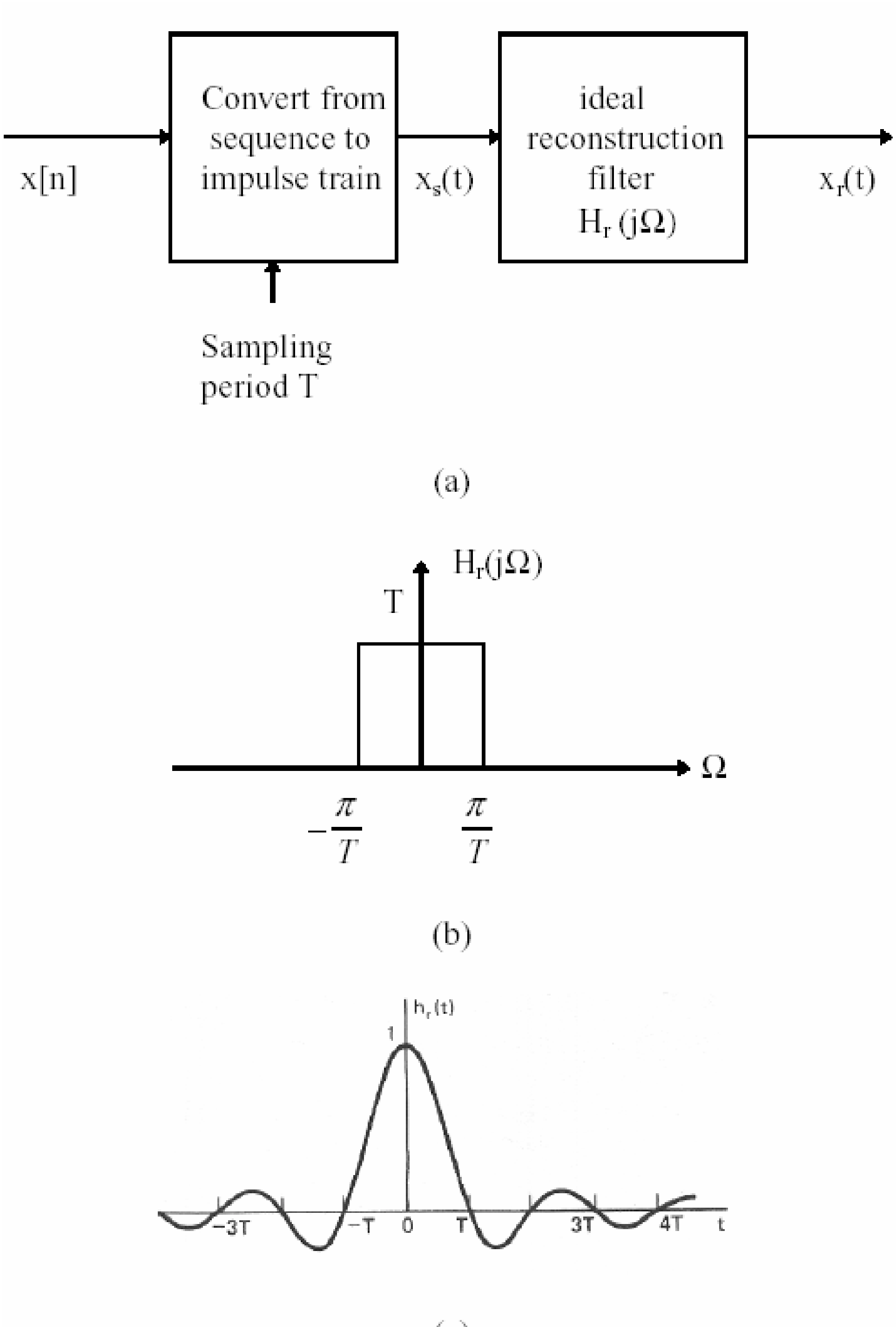
$$x_r(t) = x_s(t) * h_r(t) = \sum_{n=-\infty}^{\infty} x[n] h_r(t-nT) = \sum_{n=-\infty}^{\infty} x[n] \frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T} \quad .$$

(۱۳-۲)

شکل۳، نمودار بلوکی این فرآیند بازسازی را نشان می دهد.

از نتیجه فوق، نمونه های یک سیگنال باند محدود زمان پیوسته که با فرکانس کافی نمونه برداری شده اند (یعنی $\Omega_s > 2\Omega_N$)، برای نمایش سیگنال اصلی کافی هستند پس سیگنال اصلی از روی نمونه ها و دانستن پریود نمونه برداری قابل بازسازی است.

در عمل، فیلتر ایده آل پایین گذر قابل پیاده سازی نیست و ما باید تقریب بزنیم. علاوه بر این، یک سیگنال واقعی ممکن است پهنای باند زیادی داشته باشد که توسط سیستم نمونه برداری قابل اجرا نباشد. بنابراین در عمل باید ابتدا یک پیش فیلتر پایین گذر و با فرکانس قطع $\Omega_c \leq \frac{\Omega_s}{2}$ را اعمال کرد ($\Omega_N \leq \frac{\Omega_s}{2}$).



شکل۳- (الف) یک سیستم بازسازی سیگنال باند محدود ایده آل (ب) پاسخ فرکانسی یک فیلتر بازسازی ایده آل (ج) پاسخ ضربه به یک فیلتر بازسازی ایده آل

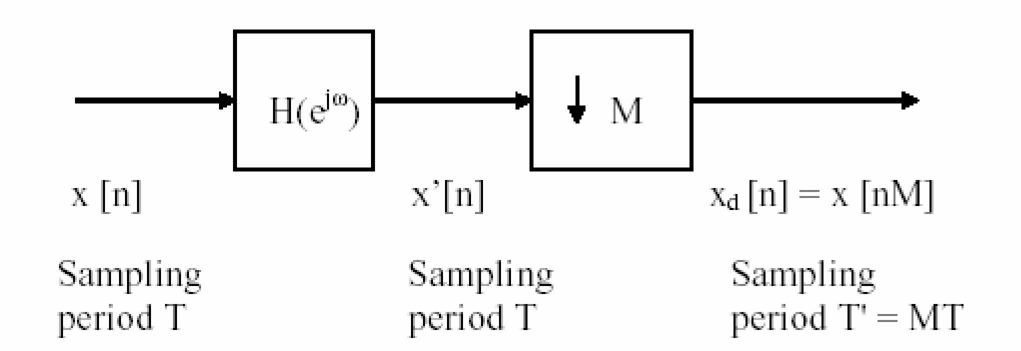
۲-۲- نمونه برداری کاهشی

نرخ نمونه برداری یک دنباله می تواند با نمونه برداری از آن کاهش یابد.

$$x_d[n] = x[nM] = x_c(nMT) \qquad (۲-۱۴)$$

در رابطهٔ (۲-۱۴) مشاهده می شود که $x_d[n]$ را می توان به صورت مستقیم با تناوب نمونه بردار یک $T' = MT$ از سوی $x_c(t)$ بدست آورد. به علاوه، اگر $X_c(j\Omega) = 0$ برای $|\Omega| > \Omega N$ آنگاه، $x_d[n]$ یک نمایش دقیق از $x_c(t)$ اگر

$\Omega_s' = \frac{2\pi}{T'} = \frac{1}{M}\Omega_s > 2\Omega_N$ باشد. بنابراین، نرخ نمونه برداری را می توان با ضریب M کاهش داد اگر نرخ نمونه برداری اولیه حداقل M برابر نرخ نایکوئیست باشد . به طور کلی، برای جلوگیری از تداخل، پهنای باند دنباله ابتدا باید به وسیله فیلتر زمان – گسسته تا M برابر کاهش یابد. نمودار بلوکی یک نمونه بردار – کاهشی در شکل٤ نشان داده شده است.



شکل٤- نمونه برداری کاهشی با ضریب M، که در آن $H(e^{jw})$ یک پیش فیلتر است. در حالت ایده آل $H(e^{jw})$ باید یک فیلتر با فرکانس قطع $\Omega_c = \frac{\pi}{M}$ باشد.

برای تعیین رابطه بین تبدیل فوریه $x[n]$ و $x_d[n]$، ابتدا باید تبدیل فوریه زمان گسسته $x[n]=x_c(nT)$ را به یاد آوریم.

$$X(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{T} - j\frac{2\pi k}{T}\right) \quad . \tag{۱۵-۲}$$

مشابه رابطه بالا، تبدیل فوریه زمان – گسسته $x_d[n]=x[nw]=x_c(nT)$ یا $T' = MT$ به صورت زیر می باشد.

$$X_d(e^{j\omega}) = \frac{1}{T'}\sum_{r=-\infty}^{\infty} X_c\left(j\frac{\omega}{T'} - j\frac{2\pi r}{T'}\right) = \frac{1}{MT}\sum_{r=-\infty}^{\infty} X_c\left(j\frac{\omega}{MT} - j\frac{2\pi r}{MT}\right) \quad . \tag{۱٦-۲}$$

اندیس جمع r در رابطهٔ (۲-١٦) به صورت زیر بیان می شود.

$$r = i + BM \tag{۱۷-۲}$$

که B و i اعداد صحیح می باشند، $-\infty < B < \infty$ و $0 < i < M-1$ است.

واضح است که r هنوز یک عدد صحیح در دامنهٔ $-\infty$ تا $+\infty$ است، حالا معادله (۲-۱۷) را می‌توان به صورت زیر بیان کرد.

$$X_d(e^{j\omega}) = \frac{1}{M}\sum_{i=0}^{M-1}\left[\ \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{MT} - j\frac{2\pi k}{T} - j\frac{2\pi i}{MT}\right)\ \right] \quad . \tag{۱۸-۲}$$

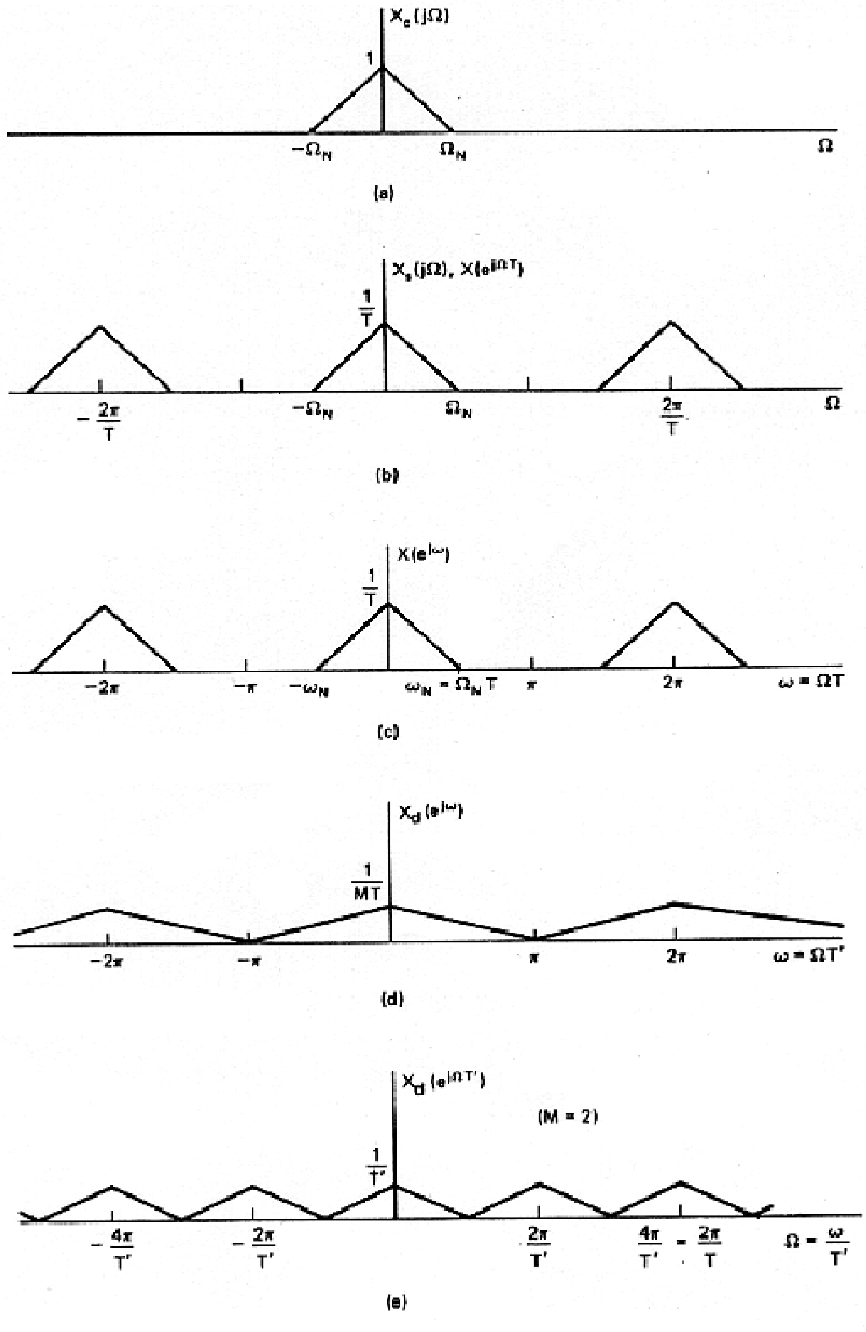
عبارت درون کروشه در رابطهٔ (۲-۱۸) از رابطهٔ (۲-۱۵) قابل جایگزینی است.

$$X(e^{j(\omega-2\pi i)/M}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega-2\pi i}{MT} - j\frac{2\pi k}{T}\right) \quad . \tag{۱۹-۲}$$

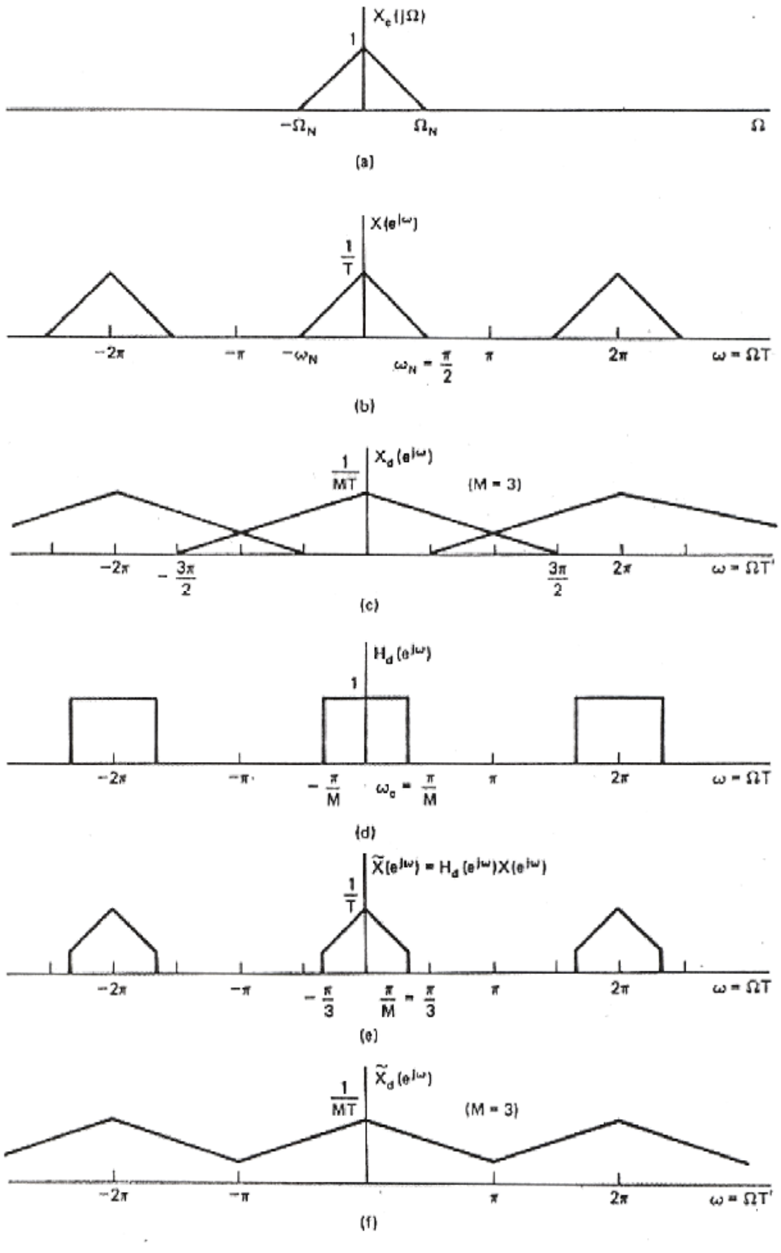
بنابراین ما می توانیم رابطهٔ (۲-۱۸) را به صورت زیر بازنویسی کنیم.

$$X_d(e^{j\omega}) = \frac{1}{M}\sum_{i=0}^{M-1} X(e^{j(\omega-2\pi i)/M}) \quad . \tag{۲۰-۲}$$

که در شکل (٥) برای M=٢ و در شکل (٦) برای M=٣ نمایش داده شده است. می توان مشاهده کرد زمانی که M=٢، نمونه برداری کاهشی باعث همپوشانی طیف سیگنال اولیه نمی‌شود. از طرف دیگر، وقتی M=٣، همپوشانی بین طیفهای تکرار شد (همان تداخل) رخ می‌دهد. برای جلوگیری از تداخل پیش فیلتر کردن قبل از نمونه برداری کاهشی الزامی است.



شکل٥- نمایش حوزه فرکانس در نمونه برداری کاهشی (M=٢)



شکل٦- نمایش حوزه فرکانس در نمونه برداری کاهشی (M=۲)

(a)-(c) بدون پیش فیلتر کردن، بنابراین سیگنال نمونه برداری شده کاهشی تداخل دارد.

(d)-(f) نمونه برداری کاهشی با پیش فیلتر کردن برای جلوگیری از تداخل.

۲-۳- نمونه برداری افزایشی سیگنالهای دیجیتالی

کاهش نرخ نمونه برداری یک سیگنال گسسته – زمان با ضریب صحیح شامل نمونه برداری یک دنباله، مشابه نمونه برداری سیگنال پیوسته می باشد. جای تعجب نیست که افزایش نرخ نمونه برداری نیز با عملیات مشابه تبدیل D/C سر و کار دارد. برای مشاهده این

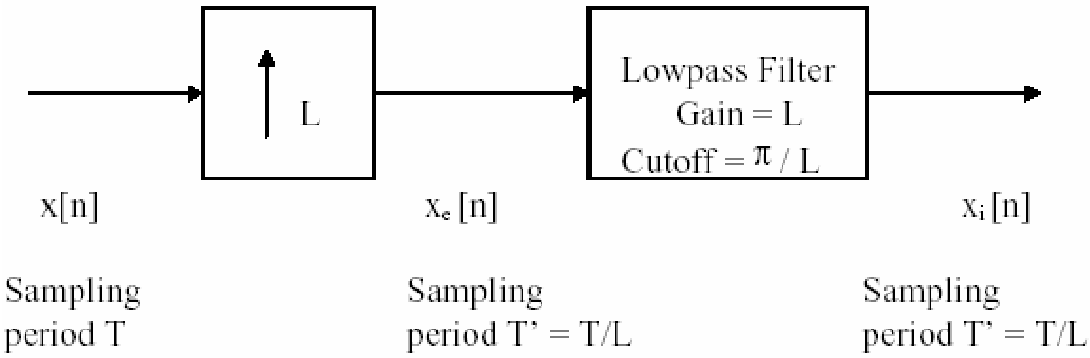
مطلب، سیگنال $x[n]$ را که می خواهیم نرخ نمونه برداری آن را با ضریب L افزایش دهیم، در نظر بگیرید. اگر ما سیگنال پیوسته $x_c(t)$ را در نظر بگیریم هدف بدست آوردن نمونه های

$$x_i[n] = x_c(nT'), \tag{۲-۲۱}$$

$$x[n] = x_c(nT). \tag{۲-۲۲}$$

از نمونه های دنبالۀ ما عملیات افزایش نرخ نمونه برداری را نمونه برداری افزایشی می نامیم.

$$xi[n] = x\left[\frac{n}{L}\right] = x_c\left(\frac{nT}{L}\right), \quad n = 0, \pm L, \pm 2L, \ldots \tag{۲-۲۳}$$



شکل:۷- پروسۀ نمونه برداری افزایشی (درون یابی)

شکل۷- یک سیستم را برای بدست آوردن $x_i[n]$ از $x[n]$ با استفاده از پردازش زمان – گسسته نشان می دهد. سیستم سمت چپ یک افزایندۀ نرخ نمونه برداری یا به طور ساده یک افزاینده نامیده می شود.

خروجی آن به صورت زیر می باشد.

(۲-۲٤)

$$x_e[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \ldots\ldots \\ 0 \quad , & otherwise \end{cases}$$

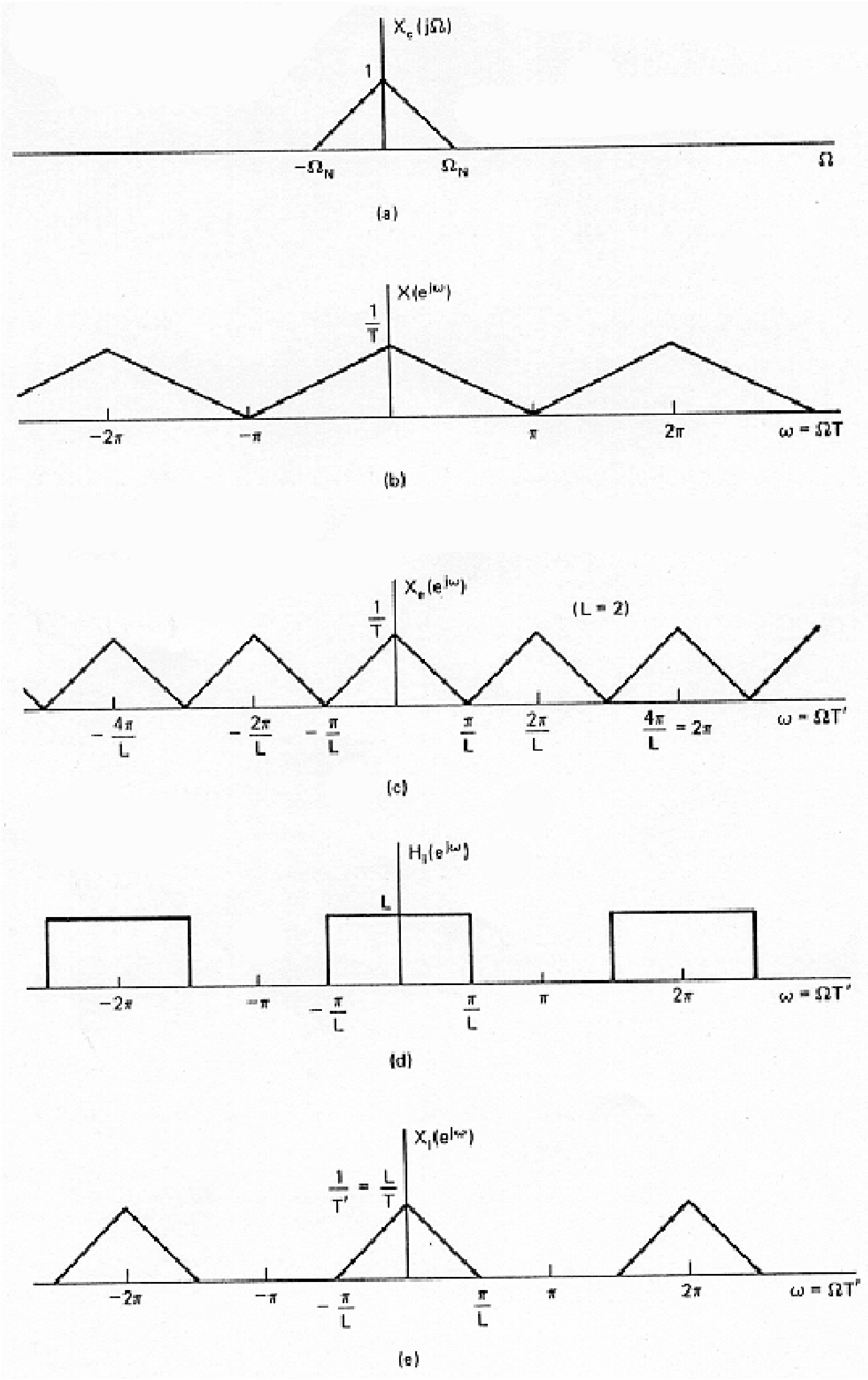
یا به بیان دیگر

$$x_e[n] = \sum_{k=-\infty}^{\infty} x[k]\delta\ [n-kL] \ . \tag{۲-۲۵}$$

تبدیل فوریه $x_e[n]$ می تواند به صورت زیر بیان شود.

$$X_e(e^{j\omega}) = \sum_{n=-\infty}^{\infty}\left(\sum_{k=-\infty}^{\infty} x[k]\delta\ [n-k\ L] \right) e^{-j\omega n} \tag{۲-۲٦}$$

رابطه بالا در شکل ۸-(b) و ۸-(c) نمایش داه شده است. برای بدست آوردن $x_i[n]$ از $x_e[n]$، احتیاج به اعمال یک فیلتر ایده ال پائین گذر با فرکانس قطع $\Omega_c = \frac{\pi}{L}$ و با بهرۀ L داریم (که در شکل ۸-(d),(e) مشاهده می شود)



شکل۸- نمایش دامنه فرکانس نمونه برداری افزایشی

۳- آزمایش ها

۳-۱- نمونه برداری صوت در فرکانسهای متفاوت

در این تجربه، شما صدای خود را ضبط خواهید کرد،

۱) صدای خود را ضبط کنید.

الف) مطمئن شوید که ارتباط میکروفون درست است یعنی میکروفون به "MIC-in" در کارت صدا متصل شده است، در سمت پشت کامپیوتر.

ب) سه پنجره "sound recorder" را باز کنید. ابتدا روی file-properties-convert کلیک کنید.

سپس ۸۰۰۰Hz,۸bit, Mono ۸kB/s را انتخاب کنید. ۵ ثانیه از صدای خود را ضبط کنید و تحت عنوان فایل rec۸.wav در آدرس خود ذخیره کنید.

ث) برای دومین و سومین ضبط صدا از فورمت Mono ۱۱ kB/s و ۱۱۰۲۵Hz,۸Bit را از ۲۲،۰۵۰Hz,۸bit, Mono ۲۲ kB/s استفاده کنید: فایلها را تحت عنوان recll.wav و rec۲۲.wav ذخیره کنید.

ت) صداها را یکی پس از دیگری گوش داده و کیفیت آنها را مقایسه کنید:

۲) صدا را از CD-Rom ضبط کنید.

یک CD صوتی را با استفاده از CD player پخش کنید. پنج ثانیه از صدای CD را در فرکانسهای ۸k و ۲۲k و ٤٤k در ۸bits/sample ضبط کنید. فایلها را به فرمت cd۲۲.wav cd۱۱.wav و cd٤٤.wav در آدرس خود ذخیره کنید.

۳) صدای MIDI را برای کامپیوتر خود ضبط کنید.

فایل MIDI را با استفاده از media player پخش کنید. ۵ ثانیه از موسیقی MIDI را در فرکانسهای ۱۱k و ۲۲k و ٤٤k ضبط کنید. سپس فایلها را تحت عنوان midi۱۱.wav.

Midi۲۲.wav و midi٤٤.wav در آدرس خود ذخیره کنید.

٤- حالا دربارهٔ کیفیت صدا از منابع و فرکانسهای نمونه برداری مختلف نظر دهید.

۵- مراحل ۱ تا ٤ را با تغییر ۱٦bit/sample به ۸bit/sample تکرار کنید.

۳-۲- پردازش صوت با استفاده از MATLAB

در این آزمایش، شما احتیاج به نوشتن برنامه MATLAB برای حل مسأله دارید.

۱- یک سیگنال نمونه برداری شده در ۲۲KHz و ۸ بیتی را در نظر بگیرید. فرکانس نمونه برداری آن را به نصف برسانید(بدون پیش فیلتر کردن) و سپس با استفاده از یک فیلتر درون یابی خطی آن را دوباره به نرخ نمونه برداری سیگنال اولیه برسانید.

۲- طیف سیگنالهای نمونه برداری شده افزایشی و کاهشی را نمایش دهید و سپس آنها را با سیگنال اولیه مقایسه کنید. خطای مربع میانگین (MSE) بین سیگنال اولیه در فرکانس ۲۲KHz و سیگنال بازسازی شده را محاسبه کنید. MSE بین دو سیگنال x(n) و y(n) با طول N به صورت زیر محاسبه می شود.

$$MSE = \sum_{n=1}^{N} [x(n) - y(n)]^2 / N$$

۳- مراحل ۱ و ۲ را برای موارد زیر تکرار کنید.

الف) یک فیلتر میانگین (شما می توانید طول فیلتر میانگین را انتخاب کنید) برای پیش فیلتر کردن.

ب) از تابع "FIR۱" در MATLAB استفاده کنید تا فیلتر بهتری طراحی کنید. از تابع "interp()" برای درون یابی استفاده کنید. طولهای متفاوت برای فیلتردرون یابی استفاده کنید. طیف فیلترهای پیش و پس پردازش را علاوه بر سیگنالهای نمونه برداری شده کاهشی یا افزایشی نمایش دهید.

٤- مراحل ۱ تا ۳ را برای سیگنال نمونه برداری شده در ۱۱KHz تکرار کنید (اختیاری)

٥- تمام نتایج را چاپ کنید.

برای موارد ۱ و ۲، سه نمونه متن MATLAB در Appendix A آورده شده است. (sp.m,sp۱.m, spfilter.m). شما باید قادر باشید تا بقیه موارد را با بهبود این متنها انجام دهید. راهنمایی: فیلتر کردن یک دنبالهٔ x( ) به وسیله یک فیلتری توسط تابع conv( ) انجام می شود. شما می توانید از دستور زیر برای شناخت یک تابع استفاده کنید. به عنوان مثال help conv. به طور خلاصه، شما باید برنامه های Appendix A را بهبود دهید تا موارد زیر را عملی سازند.

الف) بدون پیش فیلتر کردن، نمونه برداری کاهشی با نرخ ۲، درون یابی دوباره و برگشت به اندازه اولیه

ب) فیلتر میانگین، نمونه برداری کاهشی با نرخ ۲، و برگشت به اندازهٔ اولیه

ج) fir۱( )، نمونه برداری کاهشی با نرخ ۲، و برگشت به اندازهٔ اولیه

٤- گزارش

گزارش شما باید شامل m فایلها و فیلترها و شکلهای خروجی باشد. پرسشهای زیر را در گزارش خود پاسخ دهید. تمامی فایل های گزارش را به صورت یک فایل فشرده به آدرس TA e-mail بفرستید. (yashil۱@yahoo.com)

۱) اگر سیگنال ورودی $\sin(2\pi ft)$ است زمانی که f=٦KHz و فرکانس نمونه برداری ۸KHz است، سیگنال نمونه برداری شده چه خواهد بود؟ سیگنال بازسازی شده که از فیلتر پایین گذر با فرکانس قطع ٤KHz استفاده می کند چه خواهد بود؟ (مسئله را بررسی کرده و تمام مراحل آن را در گزارش خود بنویسید)

۲) درباره کیفیت فایلهای نمونه برداری شده در نرخهای نمونه برداری و bits/sample متفاوت نظر دهید. دربارهٔ تفاوت صوت و گفتار در نرخهای نمونه برداری مختلف بحث کنید.

۳) صدای درون یابی شده بعد از نمونه برداری کاهشی با نرخ ۲ را با سیگنال نمونه برداری شده کاهشی و سیگنال اصلی مقایسه کنید. توضیح دهید که چرا سیگنالهای بازسازی شده در بعضی موارد بهتر هستند.

٥- مراجع

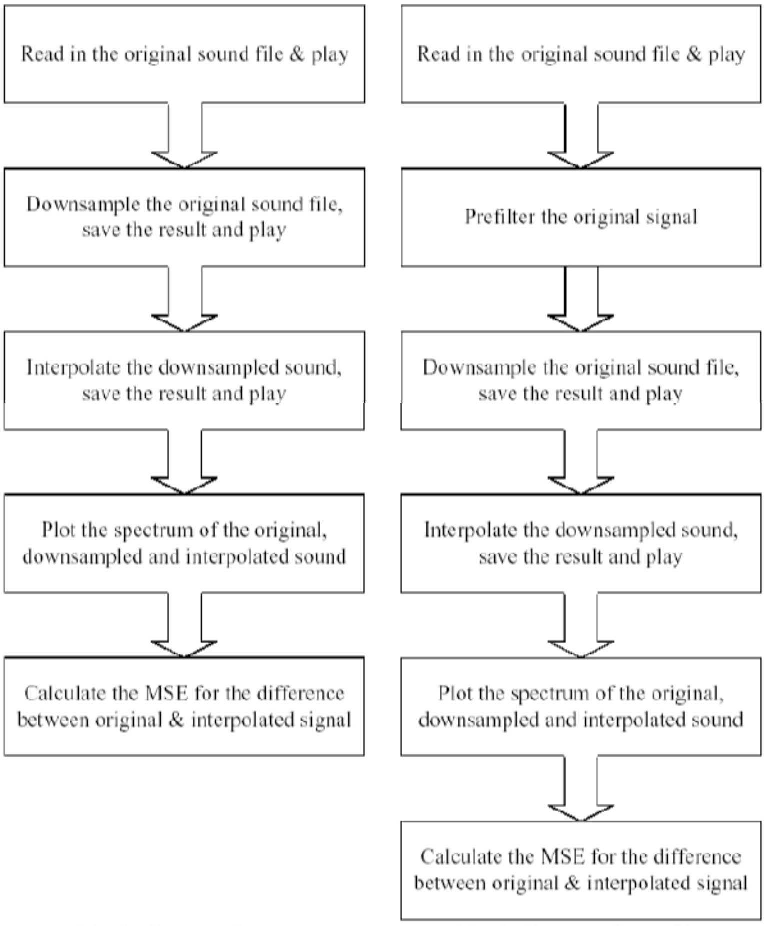
[۱]. The Math Works Inc., Matlab User's Guide, ۱۹۹۳, MATLAB USERS'S GUIDE, ۱۹۹۳.

[۲]. The Math Works Inc., MATLAB REFRENCE GUIDE, ۱۹۹۲.

[۳]. Wilsky and Openheim, Signals & Systems, Chapter ۸.

Appendix A.

1. Flow charts for sample programs.
2. Matlab scripts for down- and up- sampling a sound file.



1. block diagram for sp1.m   2. block diagram for spfilter.m

****

sp1.m (No pre-filtering , Interp() for post-filtering, Normalized output)

```
% Downsample sound file and save the result
% Then Interpolate it back to the original sampling rate
% Display the spectrums
% Usage : sp1('Infile')

function[]=sp(InFilename);

fprintf('\n The original sound \n')
[y,Fs]=wavread(InFilename);
if (rem(length(y), 2) ~=0)
   y = y(1: length(y) - 1);
end
% Play it
sound(y,Fs);
pause;

% Downsample
fprintf('\n The downsampled sound \n')
x=y(1:2:length(y));

sound(x,Fs/2);
pause;

% Save the result as down.wav
wavwrite(x,Fs/2,'down.wav');

%Interpolation
fprintf('\n The interpolated sound \n')
z=interp(x,2);

sound(z,Fs);
pause;

% Save as up.wav
wavwrite(z,Fs,'up.wav');

%Spectrums
py=spectrum(y);
px=spectrum(x);
pz=spectrum(z);
pxx=px(:,1);
```

```
xss=1.0/length(py(:,1)):1.0/length(py(:,1)):1;
l=length(pxx(1:2:length(pxx)));

xss1=1.0/l:1.0/l:1;

subplot(3,1,1),semilogy(xss,py(:,1));
xlabel('Original Signal');
h=axis;

subplot(3,1,2),semilogy(xss1,pxx(1:2:length(pxx)));
xlabel('Downsampled Signal');
axis(h);

subplot(3,1,3),semilogy(xss,pz(:,1));
xlabel('Interpolated Signal');
axis(h);

% Calculate the MSE
D=y-z;
MSE=sum(D.^2)/length(D);
fprintf('\n Mean Square Error = %g\n\n',MSE )
```

****

spfilter.m (Averaging for pre-filtering , Linear for post-filtering)

```
% Downsample sound file and save the result
% Then Interpolate it back to the original sampling rate
% Display the spectrums
% Incorporate pre-filter & post-filter
% Usage : spfilter('Infile')

function[]=spfilter(InFilename);

fprintf('\n The original sound \n')
[y,Fs]=wavread(InFilename);

% Play it
if (rem(length(y), 2) ~=0)
   y = y(1: length(y) - 1);
end

sound(y,Fs);
pause;
```

```
%Pre-filter the original signal
fprintf('\n Pre-filtering ...    &   ')
f=[1 1 1 1 1];
f=f/5;
yy=conv(y,f);

%Prune to same length
n=length(f)/2;
yy=yy(n:length(yy)-n);

% Downsample
fprintf(' The downsampled sound \n')
x=yy(1:2:length(yy));

sound(x,Fs/2);
pause;

% Save the result as down.wav
wavwrite(x,Fs/2,'down.wav');

%Interpolation (Post-filtering)
%A simple average filter

fprintf('\n The interpolated sound \n')
z=zeros(1,2*length(x));
for i=1:length(x);
        z(2*i)=x(i);
        if i<length(x)
                z(2*i+1)=(x(i)+x(i+1))/2;
        end
end


sound(z,Fs);
pause;

% Save as up.wav
wavwrite(z,Fs,'up.wav');

%Spectrums
py=spectrum(y);
px=spectrum(x);
pz=spectrum(z);
pxx=px(:,1);

xss=1.0/length(py(:,1)):1.0/length(py(:,1)):1;
```

```
l=length(pxx(1:2:length(pxx)));

xss1=1.0/l:1.0/l:1;

subplot(3,1,1),semilogy(xss,py(:,1));
xlabel('Original Signal');
h=axis;

subplot(3,1,2),semilogy(xss1,pxx(1:2:length(pxx))
xlabel('Downsampled Signal');
axis(h);

subplot(3,1,3),semilogy(xss,pz(:,1));
xlabel('Interpolated Signal');
axis(h);


% Calculate the MSE

D=y-z';
MSE=sum(D.^2)/length(D);
fprintf('\n Mean Square Error = %g\n\n',MSE )
```