



به نام خدا



دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر

پردازش سیگنال‌های دیجیتال

تمرین شماره ۴

طراح: نیما کیا حیرتی

زمان تحويل: ۴ اردیبهشت ۱۴۰۳

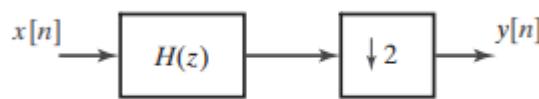
نیمسال دوم ۱۴۰۲-۱۴۰۳

سوال ۱

سیستم شکل ۱ را در نظر بگیرید که در آن دنبال فیلتر ($H(z)$) یک فشرده ساز قرار دارد. فرض کنید پاسخ صربه ($H(z)$) به صورت زیر است:

$$h[n] = \begin{cases} \left(\frac{1}{2}\right)^n & 0 \leq n \leq 11 \\ 0 & \text{otherwise} \end{cases}$$

برای افزایش کارایی این سیستم میتوان فیلتر ($H(z)$) و فشرده ساز را با استفاده از پولی فاز ساخت. یک ساختار پولی فاز موثر برای این سیستم رسم کرده، فیلترهای مورد استفاده را تعیین کنید.



شکل ۱

سوال ۲

سیستم شکل ۲ را در نظر بگیرید که در آن یک باز کننده و فیلتر ($H(z)$) (همان فیلتر سوال قبل) قرار دارد. برای افزایش کارایی این سیستم میتوان فیلتر ($H(z)$) و فشرده ساز را با استفاده از پولی فاز ساخت. یک ساختار پولی فاز موثر برای این سیستم رسم کرده، فیلترهای مورد استفاده را تعیین کنید.



شکل ۲

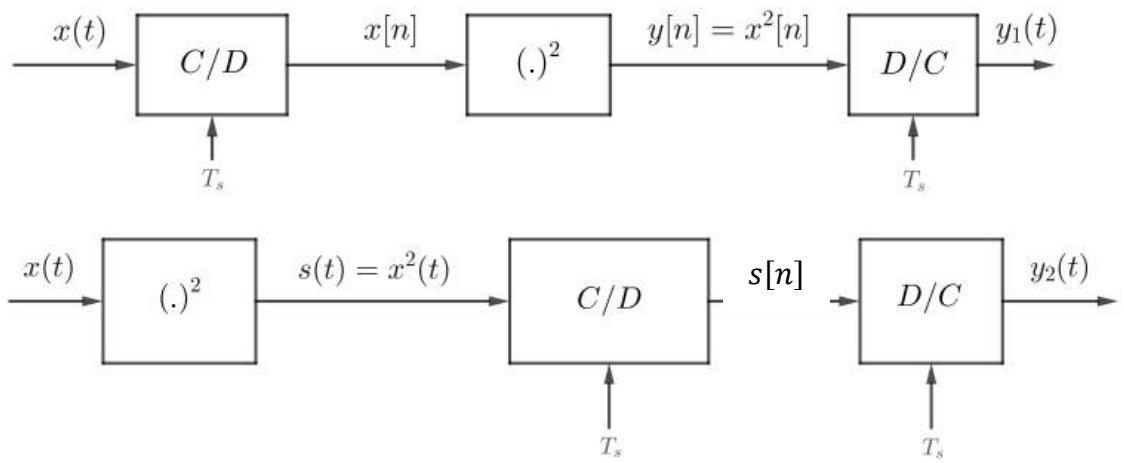
سوال ۳

دو سیستم که در شکل ۳ قابل مشاهده هستند و سیگنال پیوسته باند محدود ($x(t)$) مفروض هستند. تبدیل فوریه ($X(f)$) در شکل ۴ رسم شده است. فرض کنید نرخ نایکوئیست سیگنال ($x_s(t)$) باشد.

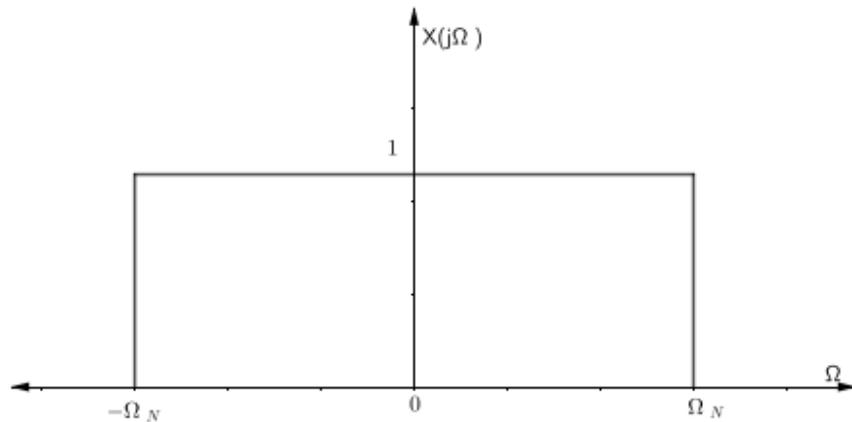
$$\left(\Omega_N \ll \frac{\pi}{T_s}, T_s = \frac{2\pi}{\Omega_s} \right)$$

الف) تبدیل فوریه تمام سیگنال های مشخص شده در شکل ۳ را رسم کنید.

ب) سپس ($y_1(t)$ و $y_2(t)$) را بدست آورده و رابطه این دو سیگنال با ($x(t)$) را بدست آورید.



شکل ۳



شکل ۴

سوال ۴ شبیه سازی بلوک های مختلف پردازش سیگنال

لازم به ذکر است در هیچ یک از قسمت های این بخش از توابع آماده مانند ...
نمی توانید استفاده کنید. همچنین با توجه به اینکه سیگنال های آنalog را نمی توان در متلب شبیه سازی کرد در صورت نیاز برای شبیه سازی سیگنال های سیگنال زمان با یک فرکانس نمونه برداری به اندازه کافی بزرگ (f_s) از آن نمونه برداری کنید و از تقریب رابطه زیر استفاده کنید (هرچه قدر فرکانس نمونه برداری بیشتر باشد، دقت این تقریب بیشتر است):

$$x(t) \approx x(nT_s)$$

الف) مطابق مطالب گفته شده در درس، تبدیل فوریه گسسته زمان $[n]x[n]$ با استفاده از رابطه زیر بدست می آید:

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

تابعی به نام DTFT بنویسید که سیگنال $x[n]$ را در بازه $[n_1, n_2]$ به عنوان ورودی دریافت کند و خروجی $X(e^{j\omega})$ را در بازه $[-\pi, \pi]$ به عنوان خروجی بازگرداند (تصویر کد مربوط به این تابع را در گزارش قرار دهید). با استفاده از این تابع تبدیل فوریه سیگنال های $x_1[n]$ و $x_2[n]$ که در روابط زیر تعریف شده اند را بدست آورده و اندازه و فاز آن را در بازه $[-\pi, \pi]$ رسم کنید.

$$x_1[n] = (0.8)^n \quad -10 \leq n \leq 20$$

$$x_2[n] = 1 \quad 0 \leq n \leq 40$$

ب) بلوک فشرده ساز (Compressor) با ضریب M را در غالب یک تابع با نام Compressor شبیه سازی کنید. ورودی این تابع $x[n]$ و خروجی آن $y[n]$ است به طوری که رابطه زیر برقرار باشد (تصویری از کد متلب مربوط به این تابع در گزارش خود قرار دهید).

$$y[n] = x[nM]$$

سیگنال پیوسته $x_c(t) = \text{sinc}^2(t)$ را در بازه $[-4,4]$ ثانیه با فرکانس 10Hz نمونه برداری کنید تا سیگنال گسسته $x_d[n]$ حاصل گردد. سپس $x_d[n]$ به ازای $M \in \{2,4\}$ به کمک نابع Compressor فشرده کنید و با استفاده از تابع DTFT پاسخ فرکانسی سیگنال خروجی را به ازای هر M به همراه پاسخ فرکانسی سیگنال $x_d[n]$ را در یک نمودار رسم کنید و نتایج را در حوزه فرکانس تحلیل کنید.

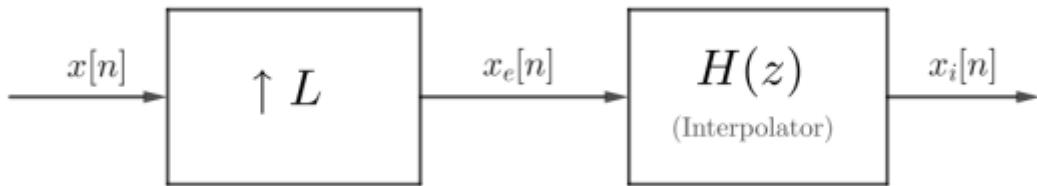
ج) بلوک (Expander) با ضریب L را که رابطه بین ورودی $x[n]$ و خروجی $y[n]$ آن به صورت رابطه زیر می باشد را در قالب یک تابع به نام Expander شبیه سازی کنید(تصویری از کد متلب مربوط به این تابع را در گزارش خود قرار دهید).

$$y[n] = \begin{cases} x\left[\frac{n}{L}\right] & n = 0, \pm L, \pm 2L, \dots \\ 0 & \text{otherwise} \end{cases}$$

سیگنال پیوسته $x_c(t) = \text{sinc}^2(t)$ را در بازه $[-4,4]$ ثانیه با فرکانس 10Hz نمونه برداری کنید تا سیگنال گسسته $x_d[n]$ حاصل گردد. سپس $x_d[n]$ به ازای $L \in \{2,4\}$ به کمک نابع Expander باز کنید و با استفاده از تابع DTFT پاسخ فرکانسی سیگنال خروجی را به ازای هر L به همراه پاسخ فرکانسی سیگنال $x_d[n]$ را در یک نمودار رسم کنید و نتایج را در حوزه فرکانس تحلیل کنید.

د) در این قسمت عمل درونیابی (Interpolation) را به سه روش مختلف بررسی میکنیم. درونیاب نوعی فیلتر با پاسخ ضربه (z) است که معمولاً در بلوک های A/D یا پس از بلوک Expander استفاده میشد.

فیلتر درونیاب در شکل ۱۱ قابل مشاهده است (تصویری از کد مربوط به این تابع در گزارش خود قرار دهید).



شکل ۱۱

برای شبیه سازی این فیلتر تابعی به نام *Interpolate* بنویسید که ورودی آن سیگنال $x[n]$ و متغیری به نام $mode = 1$ باشد. اگر $mode = 1$ باشد باید درونیابی به صورت ایده آل شکل بگیرد:

$$H_{ideal}(e^{j\omega}) = L \quad |\omega| \leq \frac{\pi}{L}$$

اگر $mode = 2$ باشد باید درونیابی به صورت خطی شکل بگیرد:

$$h_{lin}[n] = \begin{cases} 1 - \frac{|n|}{L} & |n| \leq L \\ 0 & Otherwise \end{cases}$$

و در صورتی که $mode = 3$ باشد باید درونیابی به صورت اسپلاین مکعبی شکل بگیرد (در این حالت میتوانید از تابع *spline* مطلب استفاده کنید).

- ۱) در رابطه با سه درونیاب ذکر شده و چگونگی عملکرد آنها به صورت مختصر توضیح دهید.
- ۲) توضیح دهید چرا نمیتوان در عمل از درونیاب ایده آل استفاده کرد.
- ۳) با فرض $f_s = 10Hz$ از سیگنال $s(t) = \cos(2\pi f_c t)$ در بازه $t \in [0,1]sec$ با فرکانس $F_s = 20Hz$ نمونه برداری کنید تا سیگنال $s_d[n]$ را بدست آورید و آن را رسم کنید. حال با استفاده از توابع *Interpolate* و *Expander* تبدیل کنید و آن را به همراه نمونه های $s_d[n]$ رسم کنید. طبق نمودار های رسم شده توضیح دهید آیا $s(t)$ با تقریب مناسبی بازیابی شده است؟