

۱- تبدیل فوریه گسسته سیگنال های زیر را به دست آورید.

$$\begin{aligned} \text{a) } x_1[n] &= 2^n u[-n] & \text{c) } x_3[n] &= \sum_{k=-\infty}^{+\infty} \left(\frac{1}{4}\right)^n \delta[n-3k] \\ \text{b) } x_2[n] &= n \left(\frac{1}{2}\right)^{|n|} & \text{d) } x_4[n] &= \left[\frac{\sin \frac{\pi}{2} n}{n\pi}\right] \times \left[\frac{\sin \frac{\pi}{4} n}{n\pi}\right] \end{aligned}$$

۲- تبدیل معکوس سیگنال های زیر را به دست آورید.

$$\begin{aligned} \text{a) } x(e^{j\omega}) &= \cos^2(\omega) & \text{b) } x(e^{j\omega}) &= \frac{e^{-j\omega}}{1 + \frac{1}{6}e^{-j\omega} - \frac{1}{6}e^{-j2\omega}} \end{aligned}$$

۳- می خواهیم یک سیستم LTI گسسته زمان طرح کنیم که به ازای ورودی $x[n]$ خروجی $y[n]$ را ایجاد کند.

$$\begin{aligned} x[n] &= \left(\frac{1}{2}\right)^n u[n] - \left(\frac{1}{4}\right)\left(\frac{1}{2}\right)^{n-1} u[n-1]. \\ y[n] &= \left(\frac{1}{3}\right)^n u[n] \end{aligned}$$

الف) پاسخ ضربه و پاسخ فرکانسی سیستم LTI دارای مشخصات بالا را پیدا کنید.

ب) معادله دیفرانس ارتباط دهنده ورودی - خروجی را بنویسید.

4- $y[n]$ پاسخ یک سیستم گسسته در زمان به ورودی $x[n]$ است. تبدیل فوریه این سیگنال به صورت زیر بهم مرتبط هستند.

$$Y(e^{j\omega}) = 2X(e^{j\omega}) + e^{-j\omega}X(e^{j\omega}) - \frac{dX(e^{j\omega})}{d\omega}$$

الف) آیا سیستم خطی است؟ استدلال نمایید.

ب) آیا سیستم تغییر ناپذیر با زمان است؟ استدلال نمایید.

ج) پاسخ سیستم به ورودی ضربه را بدست آورید.

5- اگر $x[n]$ به فرم روبرو باشد. بدون محاسبه مستقیم:

الف) سیگنالی را در حوزه زمان بیابید که تبدیل فوریه آن برابر با $\operatorname{Re}\{X(j\omega)\}$ باشد.

ب) سیگنالی را در حوزه زمان بیابید که تبدیل فوریه آن برابر با $\operatorname{Im}\{X(j\omega)\}$ (بخش موهومی) باشد.

۶- تمرین: Matlab

فایل صوتی `Multipletones.wav` را به کمک تابع `audioread()` شامل چند مولفه فرکانسی است:

الف) فایل صوتی را توسط `MediaPlayer` بشنوید.

ب) دامنه تبدیل فوریه گسسته زمان سیگنال فوق را بدست آورید و نمایش دهید.

ج) در طیف فرکانسی چند مولفه فرکانسی و در چه فرکانس هایی مشاهده می شود؟

د) به کمک فیلترهای پایین‌گذر، میان‌گذر و بالاگذر تک‌تک مولفه‌های فرکانسی را از سیگنال اولیه جدا کنید و در فایل‌های `Tone1.wav`، `Tone2.wav` و غیره را به کمک تابع `audiowrite()` ذخیره نمایید. به صدای فایل‌های ایجاد شده گوش دهید.