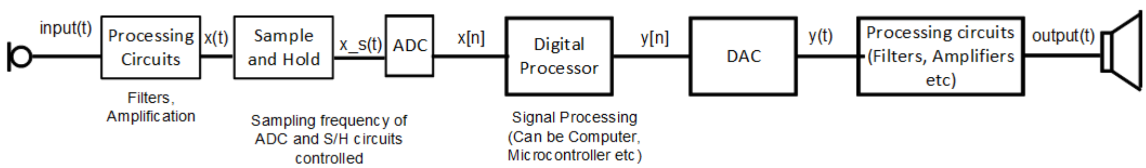
Notes:

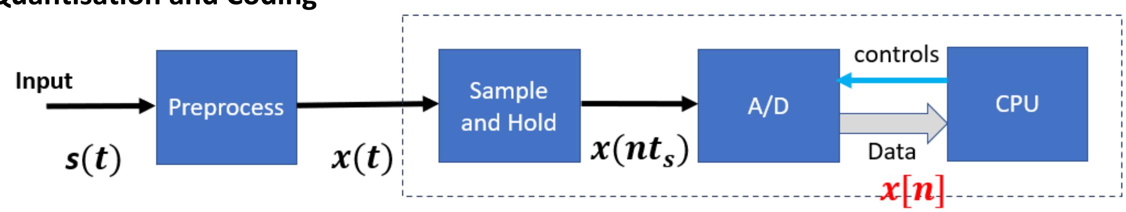
a) **Use plot() for analog signal plots and stem() for discrete signal plot.**
b) **Every plot must have labels for X-axis, Y-axis and title.**
c) **A few conditions adapted for MATLAB Auto-grader**

1 Objectives/Outcome:

Typical Audio/Video Processing System



Quantisation and Coding



- Sampling is a process of converting a continuous time signal (analog signal) x(t) into a discrete time signal x[n], which is represented as a sequence of numbers - A/D converter
- Converting x[n] back into analog (resulting in the process of reconstruction.) (D/A converter)
- For the reconstructed signal to be closer to x(t), sampling theorem is used to get x(n) from x(t).
- The sampling frequency fs determines the spacing between samples.
- Aliasing - A high frequency signal is converted to a lower frequency, results due to under sampling. Though it is undesirable in ADCs, it finds practical applications in stroboscope and sampling oscilloscopes.

Two scenarios have been explained below to understand aliasing and mathematical formulation:

<u>Scenario1: Aliasing of two signals with different frequency inputs to same discrete signal when sampled</u>

$$x_1(t) = \cos(2\pi F_1 t + \phi)$$
$$x_k(t) = \cos(2\pi F_k t + \phi)$$

Discretising $x(t)$ sampling at F_s =>

$$x_1[n] = \cos\left(2\pi F_1 \left(\frac{n}{F_s}\right) + \phi\right) = \cos(2\pi\ (F_1/F_s)\ n + \phi)$$

Discretising $x_k(t)$ sampling at F_s =>

$$x_k[n] = \cos\left(2\pi F_k \left(\frac{n}{F_s}\right) + \phi\right) = \cos(2\pi\ (F_k/F_s)\ n + \phi)$$

$x_k[n] = x_1[k]$ when $\boldsymbol{2\pi \left(\frac{F_k}{F_s}\right) n \ = \ 2\pi \left(\frac{F_1}{F_s}\right) n \pm 2\pi\ k.n} \quad n, k \ \in \ \mathbb{Z}^+$

$$\boldsymbol{F_k = \ F_1 \ \pm \ k\, F_s}$$

Scenario2: Alaising of an input signal when sampled at two different sampling rates

$$x_1(t) = \cos(2\pi F_1 t + \phi)$$

Discretising $x(t)$ sampling at F_{s1} =>

$$x_1[n] = \cos\left(2\pi F_1 \left(\frac{n}{F_{s1}}\right) + \phi\right) = \cos(2\pi\ (F_1/F_{s1})\ n + \phi)$$

Discretising $x_k(t)$ sampling at F_{s2} =>

$$x_k[n] = \cos(2\pi F_1\ (n/F_{s2}) + \phi) = \cos(2\pi\ (F_1/F_{s2})\ n + \phi)$$

$x_k[n] = x_1[k]$ when $\boldsymbol{2\pi \left(\frac{F_1}{F_{s2}}\right) n \ = \ 2\pi \left(\frac{F_1}{F_{s1}}\right) n \pm 2\pi\ k.n} \quad n, k \ \in \ \mathbb{Z}^+$

$$\boldsymbol{\frac{1}{F_{s2}} = \frac{1}{F_{s1}} \pm \frac{k}{F_1}}$$

$$\boldsymbol{F_{s2} = \frac{1}{\frac{1}{F_{s1}} \pm \frac{k}{F_1}}}$$

2 Objectives/Outcome:

1) To be able to represent the discretised signal from analogue signal
2) To understand the minimum and maximum frequency possible for discrete signal
3) To verify sampling theorem for signal of known frequency with reconstruction of signal using different sampling rates – Nyquist sampling rate, under sampling and oversampling
4) To understand and check the aliasing frequencies

3 Tasks

3.1 Task1: Min and max frequency plots for Discrete signal

Consider signal $x(n)$ = 2 cos(ωn). Plot 30 samples of $x(n)$ for different values of ω starting from 0 to 3π (both inclusive) at span of $\pi/4$.

Condition: Variable names

Assign array 'omega' for ω as per the description. On each iteration of the values for $x(n)$ corresponding to ω, print the discrete values and plot the same using stem() and paste the diagrams to draw your interpretations. Keep the signal values for different ω as a two-dimensional array.

$xn(1,:)$ – signal values for $\omega = 0$

$xn(2,:)$ – signal values for $\omega = \frac{\pi}{4}$

…….

Condition2:

Display value of xn for each value of ω.

Note: Display of xn should be as below:

```
Omega=0, xn=[2 2 2…………………………..2]
Omega=3.14 xn=[2 1.414……………………..-1.414]
…..
Omega=9.42 xn=[2.0, -2…………………2]
```

Interpret the results for your understanding $cos(\omega n)$ with different values of ω as against $cos\ (\omega t)$ for different values of ω.

3.2 Task2: Plotting discretised signal

Consider signal $x(t) = (3\cos(2\pi t) + 10\sin(4\pi t) - \cos(5\pi t))$.

a) Plot $x(t)$ as an analogue signal.

b) Generate the discrete time signal $x[n] = x(nT_s)$ => xn1, xn2, xn3 for three different sampling times $T_s = 0.08s$, $0.2s$ and $0.4s$ respectively. For each of these signals overlap xn1 with $x(t)$, xn2 with $x(t)$ and xn3 with $x(t)$

   Use MATLAB to plot the analogue signal and the resulting discrete-time signals as using 4 subplots:

   1) Subplot 1 to plot() the input analogue signal xt for duration time of 1.5s). Use appropriate time sampling to get it like analogue input. Ts=0.001s

2) Subplot 2: Superimpose the following: xn1 using stem, xn1 using plot(), xt using plot()
3) Subplot 3: Superimpose the following: xn2 using stem, xn2 using plot(), xt using plot()
4) Subplot 4: Superimpose the following: xn3 using stem, xn3 using plot(), xt using plot()

Note: All plot/stem in each subplot to have the same time scale to have superimposition properly

Conditions:

1) Simulate discretized signal to have samples taken for duration of 1.5s
2) Represent discrete values as xn1, xn2, xn3 for sampling times 0.08s, 0.2s and 0.4s respectively
3) Display values of xn1, xn2 xn3

c) Determine for which value of Ts, the discrete-time signal has lost the original information in the analogue signal. Task 3: Aliased frequency

3.3 Task 3: Aliased frequency

Consider signal $x(t) = 2 * \cos(2\pi t) + 3\sin(3\pi t)$.

a) For any sampling frequency less than Nyquist frequency plot the discrete signal output xn1 and take two aliased frequencies F_{s1} and F_{s2} that give the signals xn2 and xn3 respectively to be same signal as xn1. Plot all of them.

Conditions

i) Consider 20 samples on input so that the discretized signal is for n=0:19 for xn1, xn2, xn3

ii) Represent the discrete signals as xn for sampling frequency less than Nyquist frequency. Initialise the discrete signals to xn1, xn2 for the two aliased frequencies considered as input for sampling at less than Nyquist frequency

Mathematical formulation for above aliasing

Approach1:

$$x(t) = 2 * \cos(2\pi t) + 3\sin(3\pi t)$$

Discretising $x(t)$, sampling at F_{s1} =>

$$\begin{aligned} x_1[n] &= 2\cos\left(2\pi\left(\frac{n}{F_{s1}}\right)\right) + 3\cos\left(3\pi\left(\frac{n}{F_{s1}}\right)\right) \\ &= 2\cos(2\pi\, n/F_{s1}) + 3\cos(3\pi\, n/F_{s1}) \end{aligned}$$

Discretising $x(t)$, sampling at F_{sk} =>

$$x_k[n] = 2\cos\left(2\pi\ \left(\frac{n}{F_{sk}}\right)\right) + 3\cos\left(3\pi\ \left(\frac{n}{F_{sk}}\right)\right)$$
$$= 2\cos(2\pi\ n/F_{sk}) + \ 3\cos(3\pi\ n/F_{sk})$$

$x_k[n] = x_1[k]$
when $\quad 2\cos(2\pi\ n/F_{sk}) + \ 3\cos(3\pi\ n/F_{sk}) = 2\cos(2\pi\ n/F_{s1}) + \ 3\cos(3\pi\ n/F_{s1})$
$n, k\ \in\ \mathbb{Z}^+$

Approach2:

To analyse this, we break it into parts considering Fs = 1.5=3/2 Hz

1) $\cos(2\pi t)$ is 1Hz signal.

$$\boldsymbol{F_{s2} = \frac{1}{\frac{1}{F_{s1}} \pm \frac{k}{F_1}}}$$

$F_{s2} = \frac{1}{\frac{2}{3}\pm\frac{k}{1}} \ = \frac{3}{2\pm 3k}$

k=1 => $F_{s2} = \frac{3}{5}Hz$

k=2 => $F_{s2} = \frac{3}{8}Hz$

k=3 => $F_{s2} = \frac{3}{11}Hz$

k=4 => $F_{s2} = \frac{3}{14}Hz$

2) $\cos(3\pi t)$ is 1.5Hz signal.

$$\boldsymbol{F_{s2} = \frac{1}{\frac{1}{F_{s1}} + \frac{k}{F_1}}}$$

$F_{s2} = \frac{1}{\frac{2}{3}+\frac{k}{1.5}} \ = \frac{1}{\frac{2}{3}+\frac{2k}{3}} = \frac{3}{2+2k}$

k=1 => $F_{s2} = \frac{3}{4}Hz$

k=2 => $F_{s2} = \frac{3}{6}Hz$

k=3 => $F_{s2} = \frac{3}{8}Hz$

k=4 => $F_{s2} = \frac{3}{10}Hz$

k=5 => $F_{s2} = \frac{3}{12}Hz$

k=6 => $F_{s2} = \frac{3}{14}Hz$

Common Values from these can be picked up that alias both input frequencies