Signals and Systems

MATLAB Homework 3

B11901110 電機二 陳璿吉

Part I.

(a) Use the MATLAB function `plot` to plot $x[n]$ vs n, where $x[n]$ is given by

$$x[n] = \cos(2\pi(n-1)T_s), \qquad n = 1, 2, \ldots, 100 \tag{1}$$

and T_s is the reciprocal of the sampling frequency $f_s = 20$ Hz.

The signal is given by Figure 1.

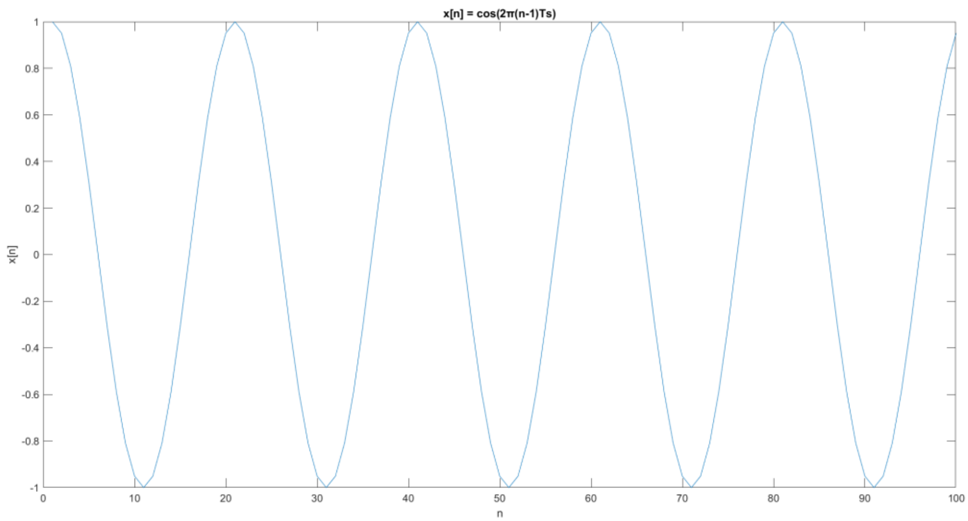

Figure 1. $x[n] = \cos(2\pi(n-1)T_s)$

(b) Obtain a Butterworth lowpass digital filter with frequency response $H(e^{j\omega})$ by using the MATLAB function `butter` with the following specifications:

Filter order: $L = 3$

Normalized cutoff frequency: $f_c = 0.05$

Please write down the transfer function $H(e^{j\omega})$ of the filter in your report, and use the MATLAB function `plot` to plot the magnitude response (in dB) vs ω (in interval $[0, \pi]$) and the phase response (in degree) vs ω of this filter. In addition, use the MATLAB function plot to plot the output signal $y[n]$ vs n when inputting $x[n]$ into the filter $H(e^{j\omega})$. There will be 3 figures in total in this problem.

The transfer function $H(e^{j\omega})$ is given by:

$$H(e^{j\omega}) = \frac{0.00041655 + 0.0012e^{-j\omega} + 0.0012e^{-j2\omega} + 0.00041655e^{-j3\omega}}{1 - 2.6862e^{-j\omega} + 2.4197e^{-j2\omega} - 0.7302e^{-j3\omega}} \qquad (2)$$

Figure 2, Figure 3 and Figure 4 illustrate the magnitude response, phase response of $H(e^{j\omega})$, and the output signal $y[n]$, respectively:

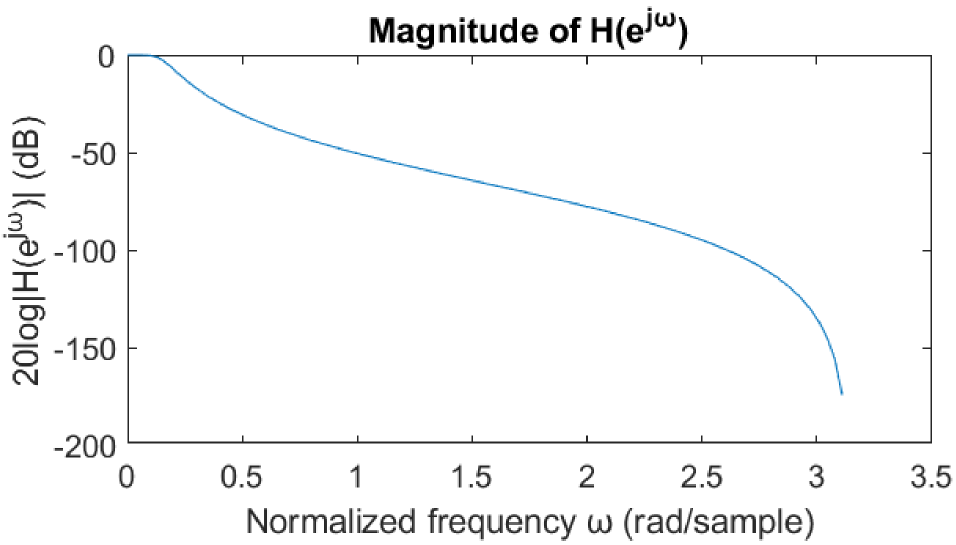

Figure 2. Magnitude response of $H(e^{j\omega})$

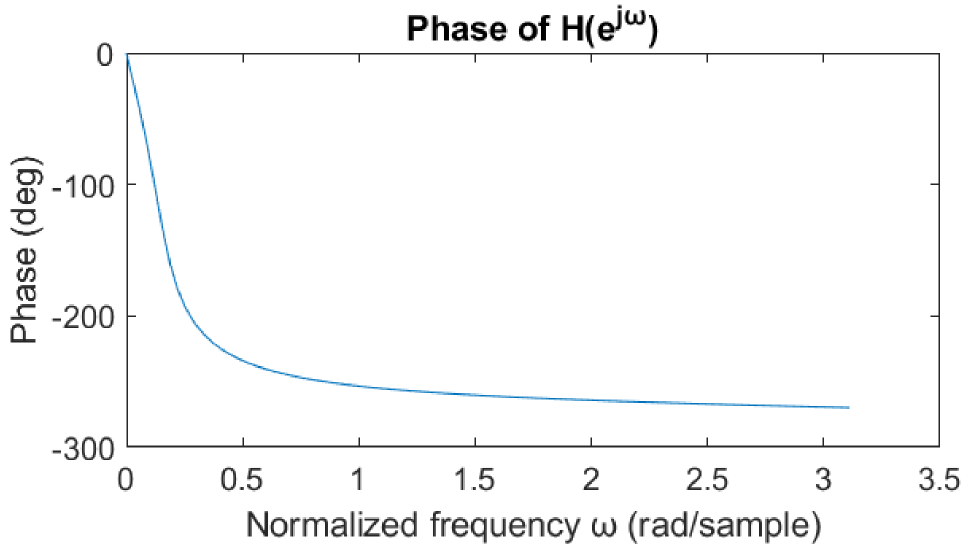

Figure 3. Phase response of $H(e^{j\omega})$

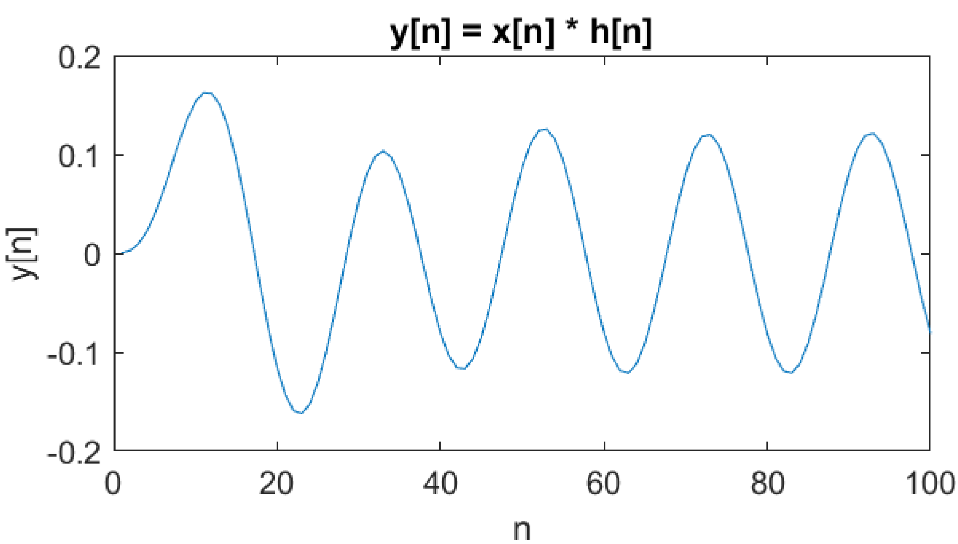

Figure 4. The output signal $y[n]$

(c) Please repeat problem (b) with $L = 7$, $f_c = 0.05$ and $f_s = 20$ Hz.

Since the order of this Butterworth filter is 7, the transfer function $H(e^{j\omega})$ is given by:

$$H(e^{j\omega}) = \frac{\sum_{k=0}^{7} b_k e^{-jk\omega}}{\sum_{k=0}^{7} a_k e^{-jk\omega}} \tag{3}$$

where the coefficients are listed in the following matrices (index from low to high):

$$\begin{aligned} b = [&1.3134 \times 10^{-8}, 9.1939 \times 10^{-8}, 2.7582 \times 10^{-7}, 4.5969 \times 10^{-7}, \\ &4.5969 \times 10^{-7}, 2.7582 \times 10^{-7}, 9.1939 \times 10^{-8}, 1.3134 \times 10^{-8}] \end{aligned} \tag{4}$$

$$a = [1, -6.2942, 17.0111, -25.5884, 23.1343, -12.5702, 3.8005, -0.4932] \tag{5}$$

Figure 5, Figure 6 and Figure 7 illustrate the magnitude response, phase response of $H(e^{j\omega})$, and the output signal $y[n]$, respectively:

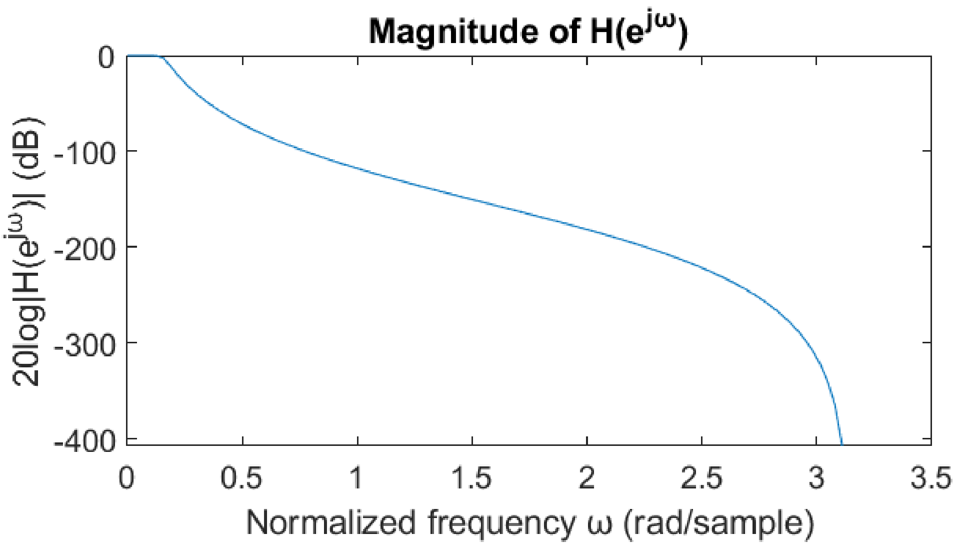

Figure 5. Magnitude response of $H(e^{j\omega})$

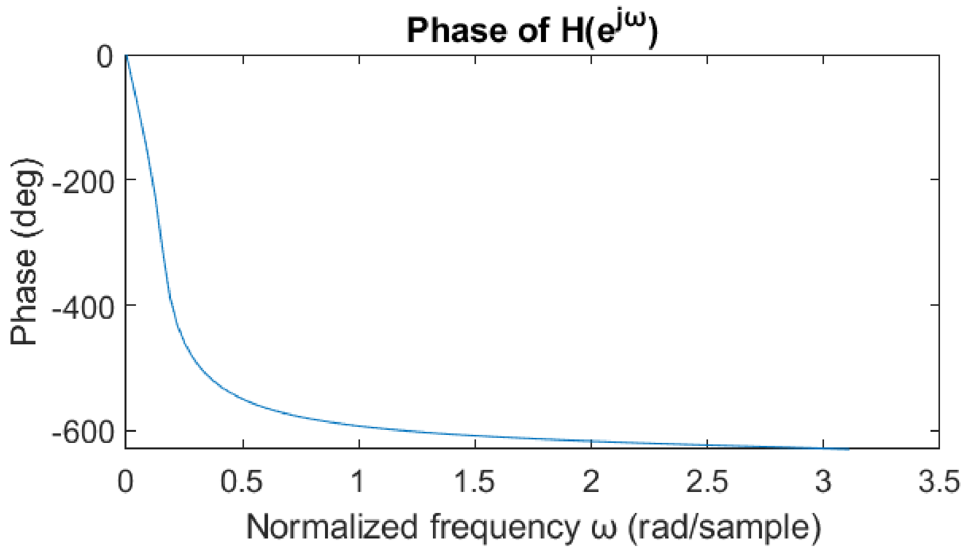

Figure 6. Phase response of $H(e^{j\omega})$

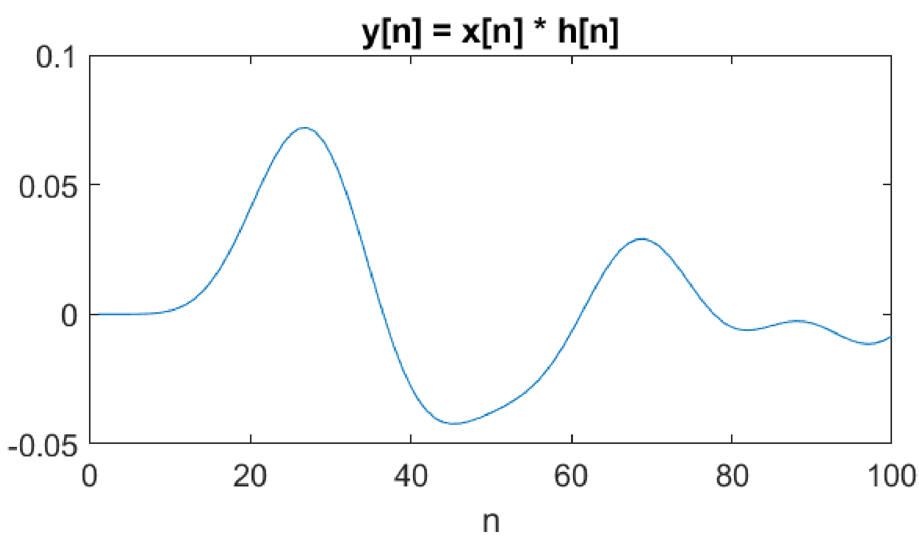

Figure 7. The output signal $y[n]$

(d) Please repeat problem (b) with $L = 3$, $f_c = 0.5$ and $f_s = 20$ Hz.

The transfer function $H(e^{j\omega})$ is given by:

$$H(e^{j\omega}) = \frac{\sum_{k=0}^{3} b_k e^{-jk\omega}}{\sum_{k=0}^{3} a_k e^{-jk\omega}} \tag{6}$$

where the coefficients are listed in the following matrices (index from low to high):

$$b = [0.1667, 0.5000, 0.5000, 0.1667] \tag{7}$$

$$a = [1, -4.9960 \times 10^{-16}, 0.3333, -1.8504 \times 10^{-17}] \tag{8}$$

Figure 8, Figure 9 and Figure 10 illustrate the magnitude response, phase response of $H(e^{j\omega})$, and the output signal $y[n]$, respectively:

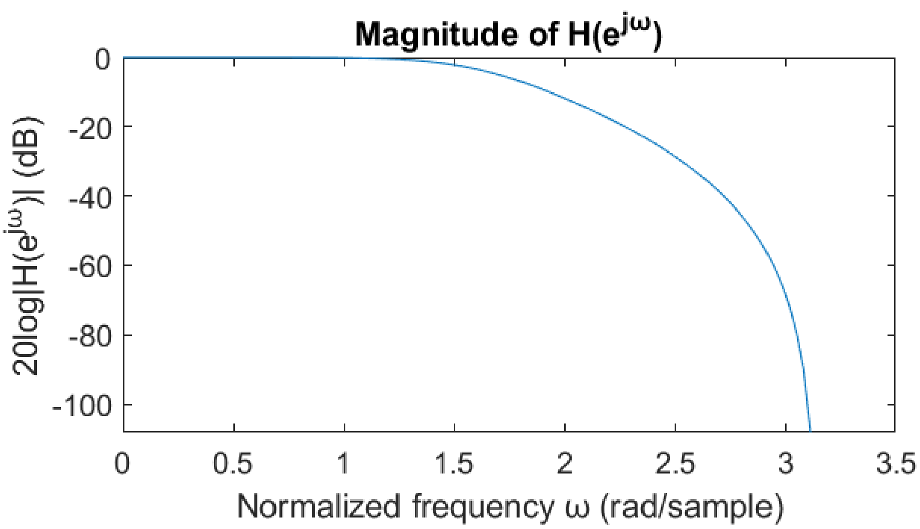

Figure 8. Magnitude response of $H(e^{j\omega})$

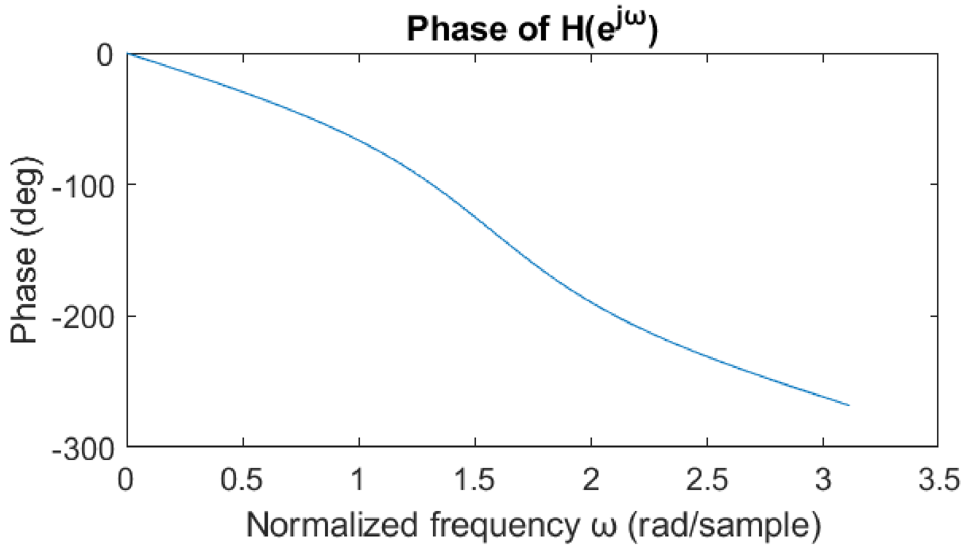

Figure 9. Phase response of $H(e^{j\omega})$

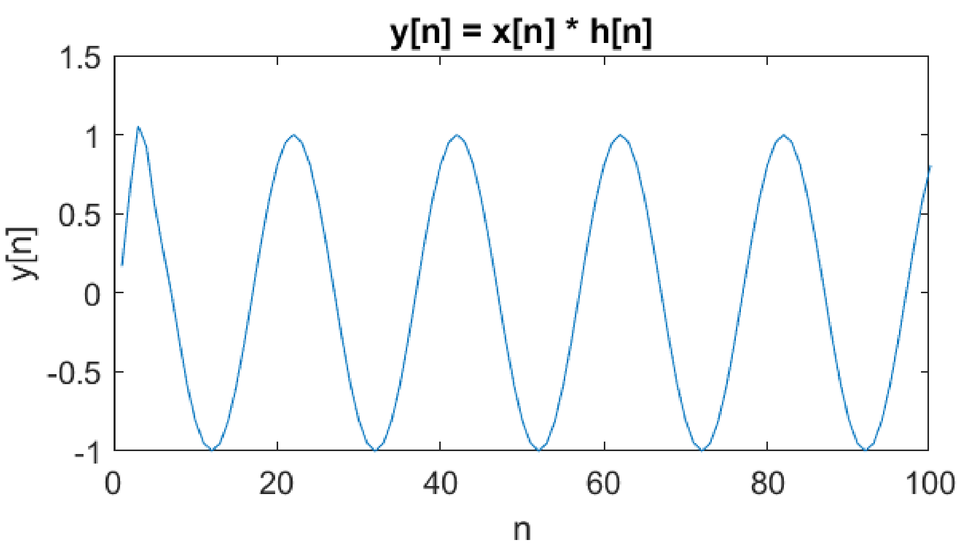

Figure 10. The output signal $y[n]$

(e) What is the effect of increasing L? What about increasing f_c? Please give some explanation in your report.

The frequency response of Butterworth filter is given by

$$|B(j\omega)|^2 = \frac{1}{1 + (j\omega/j\omega_c)^{2N}} \tag{9}$$

where N represents order and ω_c represents cutoff frequency. Although formula (9) applies to continuous signals, we can give a qualitative explanation for discrete time signals. When we increase the order L of a Butterworth filter, the attenuation of signals is more significant, as shown in Figure 2 and Figure 5. As for the phase lag of the output signal, from Figure 3 and Figure 6, we can see that increasing L contributes to more phase lag. The output signal in Figure 4 (before increase order) looks similar to the original input signal, while Figure 7 (after increase order) shows distortion.

On the other hand, if we increase the cutoff frequency f_c, we allow more frequency spectrum to be saved by the filter. Hence, from Figure 2 and Figure 8, we observe that Figure 8, which corresponds to larger f_c, begins to drop at higher frequency, compared to Figure 2. The output signal after increasing f_c is visually identical to the input signal.

Part II.

(a) Use the MATLAB function `plot` to plot $x[n]$ vs n, where $x[n]$ is given by

$$x[n] = \cos(2\pi(n-1)T_s) + 2\cos(2\pi f_1(n-1)T_s)\,, n = 1, 2, \dots, M \tag{10}$$

and $T_s = 0.002$, $f_1 = 100$ and $M = 1000$.

The signal is given in Figure 11.

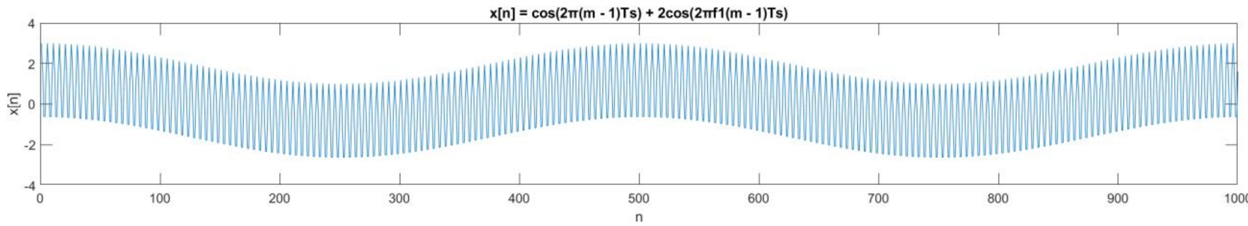

Figure 11. $x[n] = \cos(2\pi(n-1)T_s) + 2\cos(2\pi f_1(n-1)T_s)$

(b) Obtain a 16-order Butterworth <u>lowpass</u> digital filter by using the MATLAB function `butter` such that the output

$$y[n] \approx \cos(2\pi(n-1)T_s)\,, n = 1, 2, \dots, M$$

when inputting $x[n]$ into the filter.

Please write down the transfer function $H(e^{j\omega})$ of this filter and the cutoff frequency in your report, and use the MATLAB function `plot` to plot the output signal $y[n]$ vs n.

Since we want to filter out the higher frequency component, the frequency $f_1 = 100$ Hz should be larger than the cutoff frequency f_c. The sampling frequency f_s of the signal is given by $f_s = 1/T_s$, which is 500 Hz. If we design $f_c = 50$ Hz, then the normalized frequency to be used in the program is

$$f_c' = \frac{f_c}{0.5 f_s} = 0.2 \tag{11}$$

The transfer function $H(e^{j\omega})$ obtained by such design is given by

$$H(e^{j\omega}) = \frac{\sum_{k=0}^{16} b_k e^{-jk\omega}}{\sum_{k=0}^{16} a_k e^{-jk\omega}} \tag{12}$$

where the coefficients are listed in the following matrices (index from low to high):

$$b = 10^{-5}[0.0001, 0.0009, 0.0070, 0.0326, 0.1060, 0.2544, 0.4664, 0.6663, \\ 0.7496, 0.6663, 0.4664, 0.2544, 0.1060, 0.0326, 0.0070, 0.0009, 0.0001] \tag{13}$$

$$a = [1, -9.5922, 43.9955, -127.7924, 262.6519, -404.4528, 482.1181, \\ -453.3463, 339.5554, -203.1005, 96.6268, -36.1596, 10.4286, -2.2398 \\ 0.3377, -0.0319, 0.0014] \tag{14}$$

With this design, we obtain the output in Figure 12.

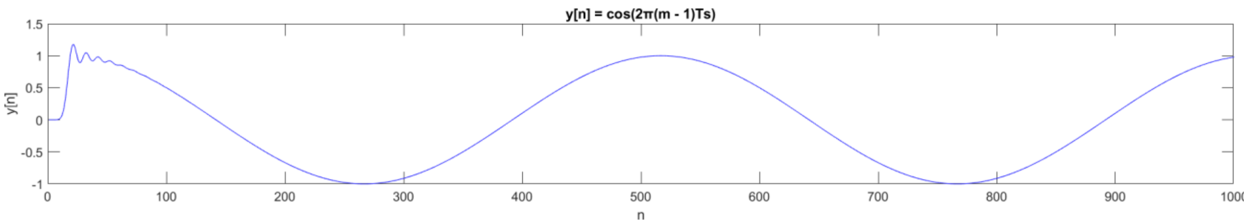

Figure 12. The output signal $y[n]$ of the input signal to the filter

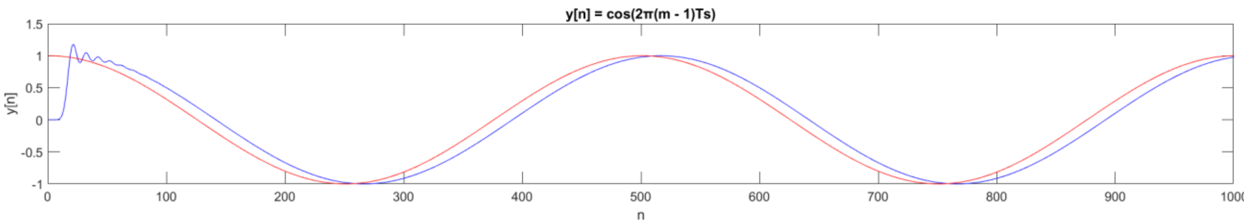

Figure 13. Comparison of $y[n]$ compared to the desired signal

Except for the early spindles and the slight phase difference, the output is similar to the desired signal $\cos(2\pi(n-1)T_s)$.

(c) Obtain a 16-order Butterworth <u>bandpass</u> digital filter by using the MATLAB function `butter` such that the output

$$y[n] \approx 2\cos(2\pi f_1(n-1)T_s), n = 1, 2, \ldots, M$$

when inputting $x[n]$ into the filter.

Please write down the transfer function $H(e^{j\omega})$ of this filter and the cutoff frequency in your report, and use the MATLAB function `plot` to plot the output signal $y[n]$ vs n.

This time, we want to block lower frequency components. Therefore, we have to make sure that the 1 Hz component of $\cos(2\pi(n-1)T_s)$ is lower than the first cutoff frequency f_{c1}. Furthermore, f_1 should be within the two cutoff frequencies, which means $f_{c1} < f_1 < f_2$.

From (11), we design the normalized frequency to be $f'_{c1} = f'_c/\sqrt{2}$ and $f'_{c2} = \sqrt{2}f_c{}'$. In this case, the transfer function of the filter is given by

$$H\left(e^{j\omega}\right) = \frac{\sum_{k=0}^{32} b_k e^{-jk\omega}}{\sum_{k=0}^{32} a_k e^{-jk\omega}} \tag{15}$$

where the coefficients are listed in the following matrices (index from low to high):

$$\begin{gathered} b = 10^{-4}[0.00061, 0, -0.0097, 0, 0.073, 0, -0.34, 0, 1.10, 0, -2.64, 0, \\ 4.85, 0, -6.92, 0, 7.78, 0, -6.92, 0, 4.85, 0, -2.64, 0, 1.10, 0, -0.34, 0 \\ 0.073, 0, -0.0097, 0, 0.00061] \end{gathered} \tag{16}$$

$$\begin{gathered} a = [1, -5.9936, 23.8773, -69.6818, 167.74, -342.32, 614.05, -984.30, \\ 1415.9, -1862.1, 2249.8, -2506.1, 2587.0, -2479.0, -1838.1, 1425.1, -1029.9 \\ , 694.11, -435.38, 253.97, -137.31, 68.66, -31.57, 13.30, -5.09, 1.75, \\ -0.54, 0.14, -0.03, 0.006, -0.00082, 0.000076] \end{gathered} \tag{17}$$

With this design, we obtain the output in Figure 14.

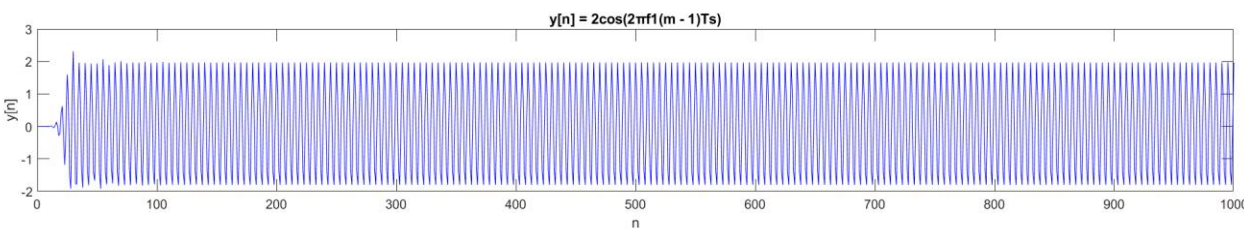

Figure 14. The output signal $y[n]$ of the input signal to the filter

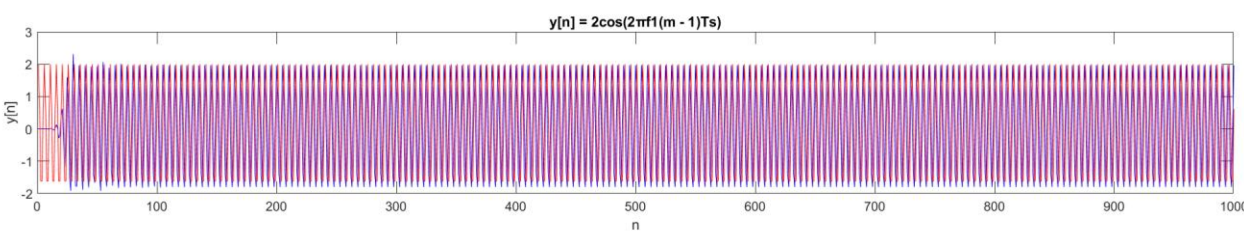

Figure 15. Comparison of $y[n]$ compared to the desired signal

Except for the early spindles and slight phase difference, the output signal is very similar to the desired signal $2\cos(2\pi f_1(n-1)T_s)$.