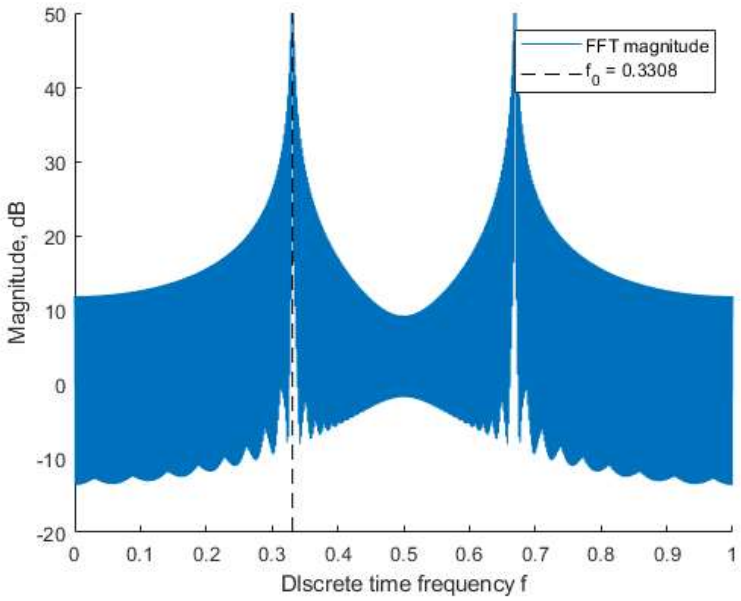
## Contents



```
A = 3.7; %amplitude in V
f0 = 0.3308;
N = 512; %block size
NFFT = 32768; %FFT points

n = linspace(0,N-1,N); %time indices
x_w = A * cos(2*pi()*f0*n); %sample signal as function of time indices

fft_x_w = fft(x_w,NFFT); %dtft calculation from FFT
ffw_x_w_dB = 20 * log10(abs(fft_x_w) + 0.0000000000001); %magnitude of FFT, small positive value in log10() argument to avoid log10(0)

f = linspace(0,1,NFFT); %frequency from 0 to 1, NFFT indices

figure;
hold on;
plot(f,ffw_x_w_dB,'DisplayName','FFT magnitude'); %plots discrete time frequency 'f' vs magnitude
ylim([-20 50]);
plot([f0 f0], ylim,'--k', 'DisplayName','f_0 = 0.3308'); %shows f0
xlabel('DIscrete time frequency f');
ylabel('Magnitude, dB');
legend('show');

hold off;
```

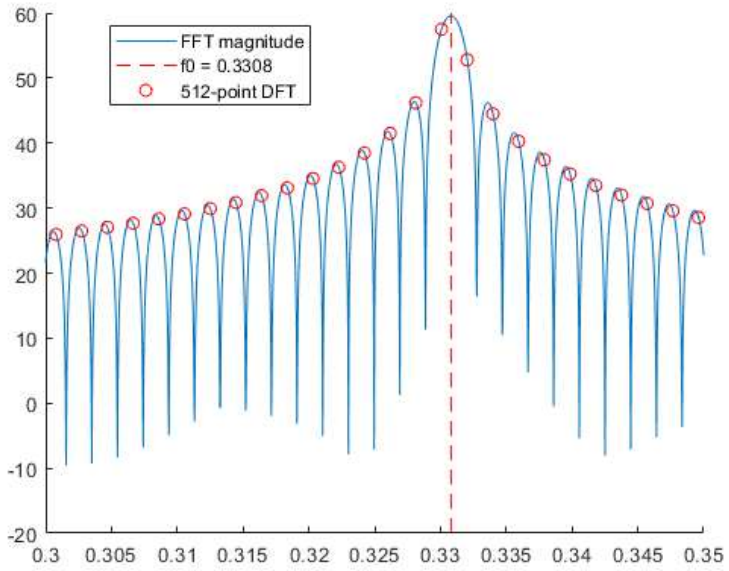



## 3b, copied 3a code

```
fprintf('NEW INSTANCE \n');

A = 3.7;
f0 = 0.3308;
N = 512;
NFFT = 32768;
n = linspace(0,N-1,N);
x_w = A * cos(2*pi()*f0*n);
fft_x_w = fft(x_w,NFFT);
ffw_x_w_dB = 20 * log10(abs(fft_x_w) + 0.0000000000001);
f = linspace(0,1,NFFT);
figure;
hold on;
plot(f,ffw_x_w_dB,'DisplayName','FFT magnitude');
ylim([-20 60]);
xlim([0.30 0.35]);
```

```
plot([f0 f0], ylim,'--r','DisplayName','f0 = 0.3308'); %shows f0
%no hold-off here, is instead below to include DFT on single plot

%end of 3a code

ffw_x_w_DFT = fft(x_w); %512-point DFT calculation
fft_x_w_DFT_dB = 20 * log10(abs(ffw_x_w_DFT) + 0.00000000000000000001); %compute 32768 point FFT


f_DFT = linspace(0,1,N+1);%aligns 512 DFT points to 32768 point DFT curve


plot(f_DFT(1:N),fft_x_w_DFT_dB,'ro','MarkerSize',6,'DisplayName','512-point DFT');

legend('show', 'Location','best');




ylim([-20 60]); %dB only has so much of a range
xlim([0.30 0.35]);%discrete time frequency range specified in problem






hold off;
```

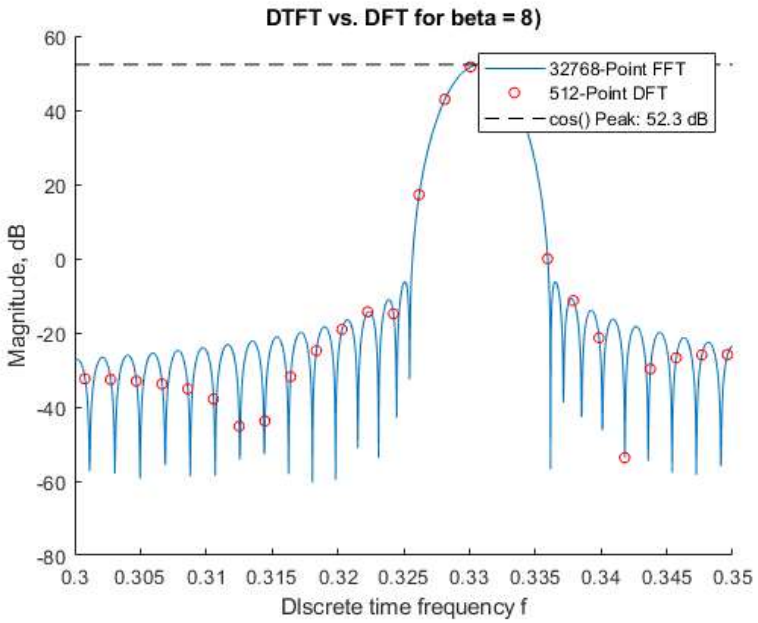
NEW INSTANCE




### 3c

```
w = kaiser(N,8); %N = 512, 8 = beta

x_w = x .* w'; %applies kaiser window to x[n]

%compute DTFT
DTFT_x_w = fft(x_w, NFFT);
DTFT_x_w_dB = 20 * log10(abs(DTFT_x_w) + 0.000000000000001);

%compute 512-point DFT
DFT_x_w = fft(x_w);
DFT_x_w_dB = 20 * log10(abs(DFT_x_w) + 0.000000000000001);

%defining frequency axes
DTFT_f = linspace(0,1,NFFT);
DFT_f = linspace(0,(N-1)/N,N);
```

```
%calculating cosine peak
peak = A * (sum(w)/N) *N/2; %amplitude of cos() times average value of window function times N/2 (since positive and negative frequencies split)
peak_dB = 20 * log10(peak); %'peak' is nonzero so no correction needed

figure;

hold on;

plot(DTFT_f,DTFT_x_w_dB,'DisplayName','32768-Point FFT'); %DTFT on plot
plot(DFT_f,DFT_x_w_dB,'ro','MarkerSize',5,'DisplayName','512-Point DFT'); %DFT on plot
plot([0.30 0.35], [peak_dB peak_dB], '--k','DisplayName',sprintf('cos() Peak: %.1f dB', peak_dB)); %cos() amplitude in dB on plot

xlim([0.3 0.35]);
xlabel('DIscrete time frequency f');
ylabel('Magnitude, dB');
title('DTFT vs. DFT for beta = 8)');
legend('show');


hold off;
```




*Published with MATLAB® R2024b*