Aim ↬ Overview

- **Verification of Sampling Theorem in both time and frequency domain.**
- **Determine the energy and power of the following signal.**

$$\boldsymbol{x(t) = e^{-a|t|}}$$

Software Required ↬ MATLAB

Theory ↬

The Sampling Theorem states that a continuous-time signal can be exactly reconstructed from its discrete samples if it is sampled at a frequency greater than twice its highest frequency component. Specifically, if a signal x(t) is band-limited to a maximum frequency f_{max} , it can be perfectly reconstructed if the sampling frequency f_s satisfies $f_s \geq 2f_{max}$.

To verify this theorem in the time domain, we begin with a continuous-time signal, such as $x(t)=\cos(2\pi ft)$, and sample it at a frequency f_s . The sampling frequency must be at least twice the signal frequency f to ensure accurate representation. We then reconstruct the signal using sinc interpolation and compare it with the original signal to confirm that they are equivalent.

In the frequency domain, the verification involves computing the Fourier Transforms of both the original and sampled signals. By analyzing the frequency spectra, we ensure that the sampling frequency is sufficient to prevent aliasing, where high-frequency components overlap due to inadequate sampling.

Energy and Power Calculation for $\boldsymbol{x(t) = e^{-a|t|}}$ ↴

The energy of a continuous-time signal x(t) is defined as

$$E = \int_{-\infty}^{\infty} |x(t)|^2 dt$$

The average power of a signal is given by

$$P = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} |x(t)|^2 dt$$

Code ↬

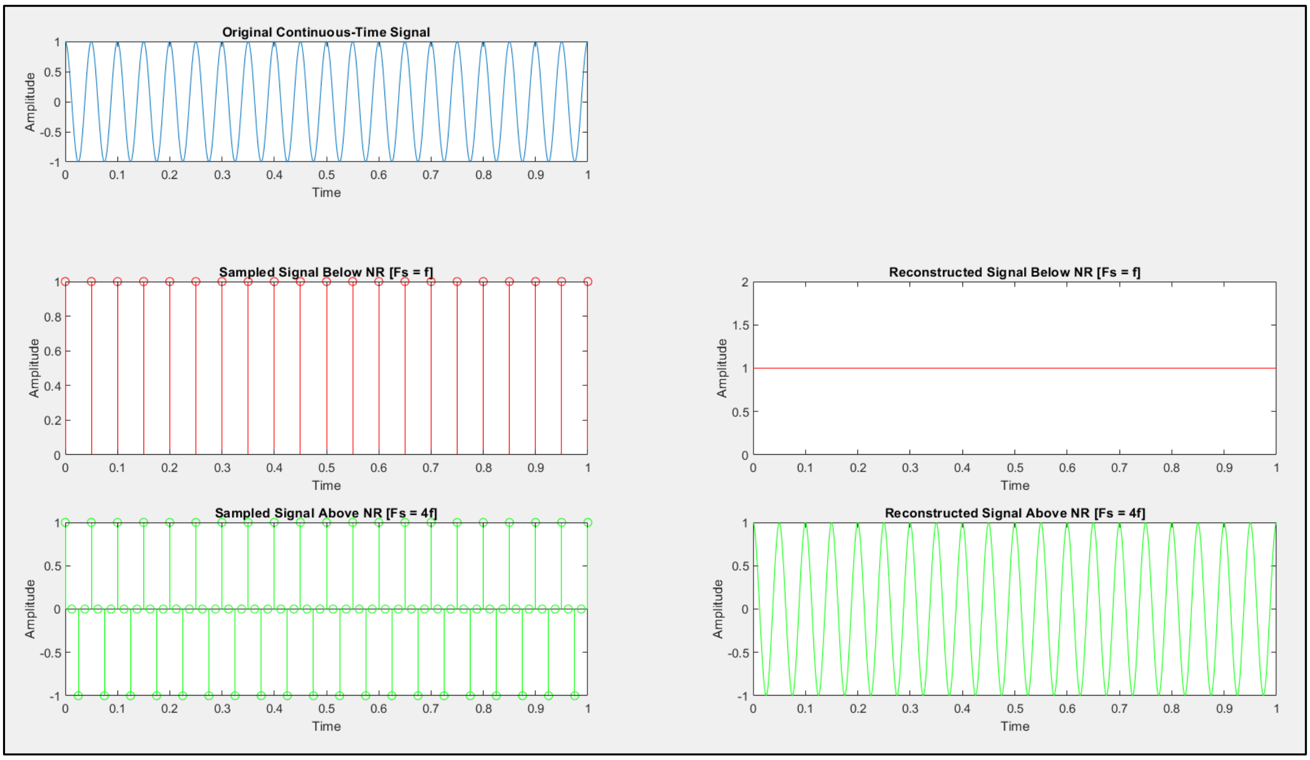
```
%Sampling Theorem - Time & Freq domain
Fs = 1000;
t = 0:1/Fs:1;
f = 20;
```

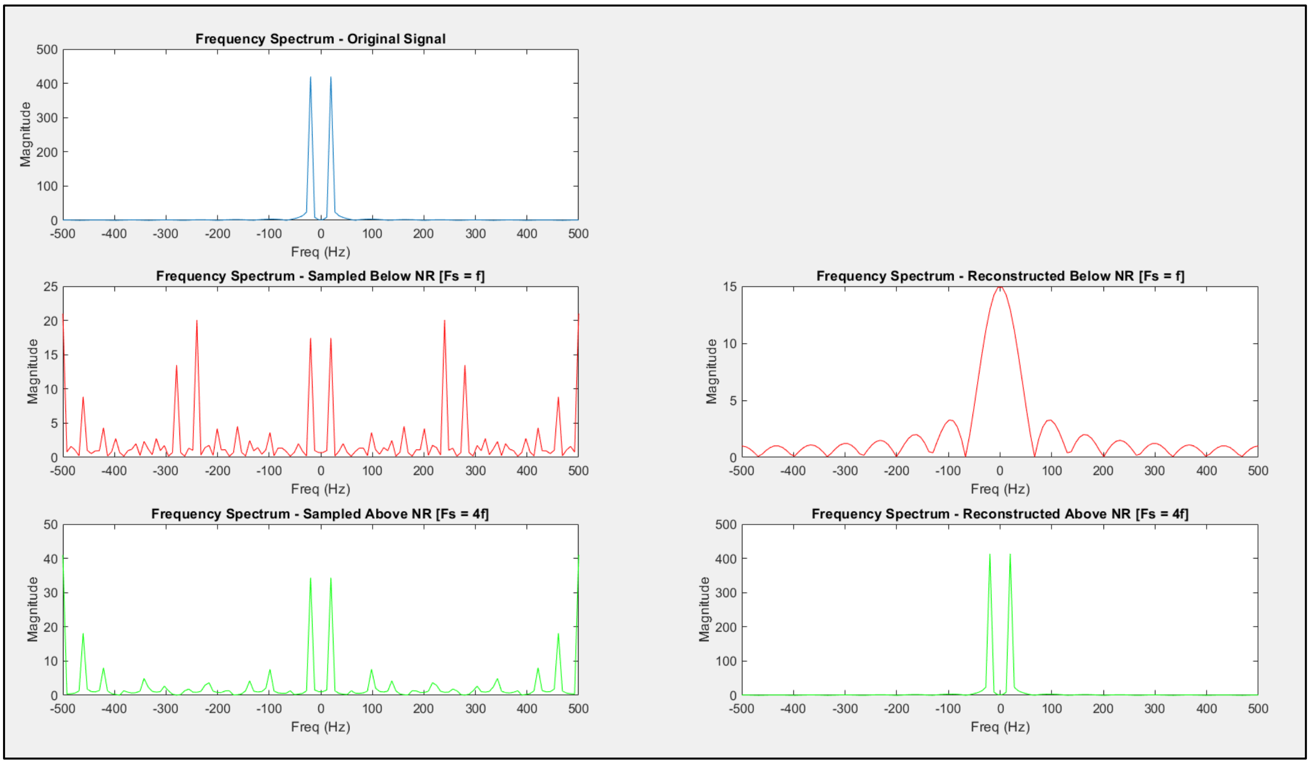
```
x = cos(2*pi*f*t);

subplot(4,2,1);
plot(t, x);
xlabel('Time');
ylabel('Amplitude');
title('Original Continuous-Time Signal');

Fs1 = f;
Fs2 = 4*f;

t1 = 0:1/Fs1:1;
t2 = 0:1/Fs2:1;

x1 = cos(2*pi*f*t1);
x2 = cos(2*pi*f*t2);

subplot(3,2,3);
stem(t1, x1, 'r');
xlabel('Time');
ylabel('Amplitude');
title('Sampled Signal Below NR [Fs = f]');

subplot(3,2,5);
stem(t2, x2, 'g');
xlabel('Time');
ylabel('Amplitude');
title('Sampled Signal Above NR [Fs = 4f]');

t_interp = 0:1/Fs:1;
x1_interp = interp1(t1, x1, t_interp, 'spline');
x2_interp = interp1(t2, x2, t_interp, 'spline');

subplot(3,2,4);
plot(t_interp, x1_interp, 'r');
xlabel('Time');
ylabel('Amplitude');
title('Reconstructed Signal Below NR [Fs = f]');

subplot(3,2,6);
plot(t_interp, x2_interp, 'g');
xlabel('Time');
ylabel('Amplitude');
title('Reconstructed Signal Above NR [Fs = 4f]');
```

```
n = 128;
f_freq = linspace(-Fs/2, Fs/2, n);

X = zeros(1, n);
X1 = zeros(1, n);
X2 = zeros(1, n);
X1_interp = zeros(1, n);
X2_interp = zeros(1, n);

for k = 1:n
    freq = f_freq(k);
    X(k) = sum(x .* exp(-1i * 2 * pi * freq * t));
    X1(k) = sum(x1 .* exp(-1i * 2 * pi * freq * t1));
    X2(k) = sum(x2 .* exp(-1i * 2 * pi * freq * t2));
    X1_interp(k) = sum(x1_interp .* exp(-1i * 2 * pi * freq *
t_interp));
    X2_interp(k) = sum(x2_interp .* exp(-1i * 2 * pi * freq *
t_interp));
end

figure;

subplot(3,2,1);
plot(f_freq, abs(X));
xlabel('Freq (Hz)');
ylabel('Magnitude');
title('Frequency Spectrum - Original Signal');

subplot(3,2,3);
plot(f_freq, abs(X1), 'r');
xlabel('Freq (Hz)');
ylabel('Magnitude');
title('Frequency Spectrum - Sampled Below NR [Fs = f]');

subplot(3,2,4);
plot(f_freq, abs(X1_interp), 'r');
xlabel('Freq (Hz)');
ylabel('Magnitude');
title('Frequency Spectrum - Reconstructed Below NR [Fs = f]');

subplot(3,2,5);
plot(f_freq, abs(X2), 'g');
xlabel('Freq (Hz)');
ylabel('Magnitude');
```

```
title('Frequency Spectrum - Sampled Above NR [Fs = 4f]');

subplot(3,2,6);
plot(f_freq, abs(X2_interp), 'g');
xlabel('Freq (Hz)');
ylabel('Magnitude');
title('Frequency Spectrum - Reconstructed Above NR [Fs = 4f]');


%Energy & Power of x(t) = e^-a|t|
a = 2;
syms t

x_t = exp(-a * abs(t));

energy = int(x_t^2, t, -inf, inf);
energy_value = double(energy);
fprintf('Energy of the signal: %f\n', energy_value);

T = 100000;
power = int(x_t^2, t, -T, T)/(2*T);
power_value = double(power);
fprintf('Power of the signal: %f\n', power_value);
```

Output ↬






```
>> Energy_Power
Energy of the signal: 0.500000
Power of the signal: 0.000003
```

Result ↬

The experiment verified the Sampling Theorem using a cosine signal sampled at various rates. Reconstructed signals and their spectra were analyzed, and the energy and power of an exponential signal were computed.

Conclusion ↬

Sampling at or above the Nyquist rate was confirmed essential to prevent aliasing and ensure accurate signal reconstruction. The Fourier analysis and energy calculations reinforced key signal processing concepts.

Precautions ↬

- Ensure the correct sampling rate to prevent aliasing, in line with the Nyquist criterion.
- Verify the accuracy of signal generation formulas to avoid computational errors.
- Appropriately label and title plots to clearly distinguish between different signals and sampling rates.
- Handle mathematical singularities, such as potential division by zero in sinc functions, with care.