Constructor University Bremen

Prelab: Sampling
Fall Semester 2024

Author: **Idriz Pelaj**

Experiment Conducted By:
Mr. Idriz Pelaj, Mr. Getuar Rexhepi

Conducted on: **October 18, 2023**

1. Prelab

1.1 Problem 1: The Sampling Theorem

1. **Analog signals are usually passed through a low-pass filter prior to sampling. Why is this necessary?**

   The low-pass filter is used to remove noise, and to remove frequency components that may be higher than the Nyquist frequency. This is necessary to prevent not accounting for the higher frequency components, which would result in aliasing.

2. **What is the minimum sampling frequency for a pure sine wave input at 3KHz? Assume that the signal can be completely reconstructed.**

   The minimum sampling frequency for a pure sine wave input at 3KHz would be $f_s > 2 * f_{Nyquist} = 6$KHz

3. **What is the Nyquist frequency?**

   The Nyquist frequency is the highest frequency that can be represented in a sampled signal. It is half the sampling frequency and defined as $f_{Nyquist} = \frac{f_s}{2}$

4. **What are the resulting frequencies for the following input sinusoids 500Hz, 2.5KHz, 5KHz and 5.5KHz if the signals are sampled by a sampling frequency of 5KHz?**

   The frequencies would be:

   (a) 500Hz: 500Hz
   (b) 2.5Khz: 0KHz (DC component)
   (c) 5KHz: 0KHz (Aliased)
   (d) 5.5KHz: 0.5KHz (Aliased, by $f - f_{sampling}$)

5. **Mention three frequencies of signal that alias to a 7Hz signal. The signal is sampled by a constant 30 Hz sampling frequency.**

   The frequencies would be: 37Hz, 67Hz, 97Hz.

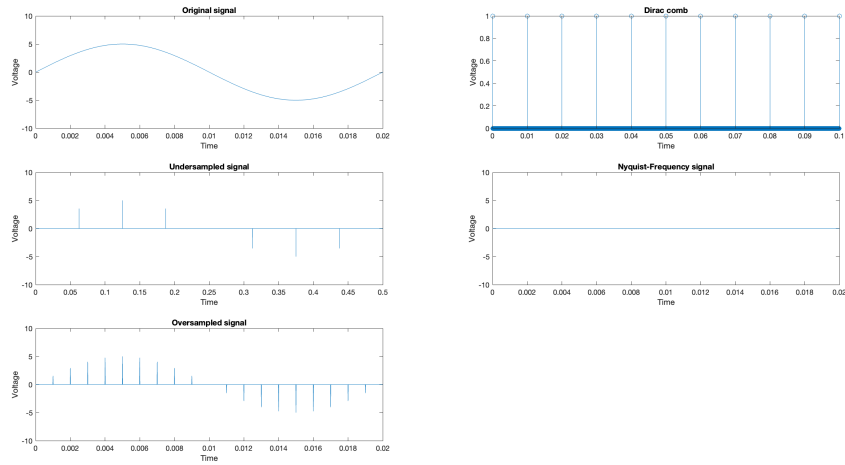## 1.2 Problem 2: Impulse Train Sampling and Real Sampling
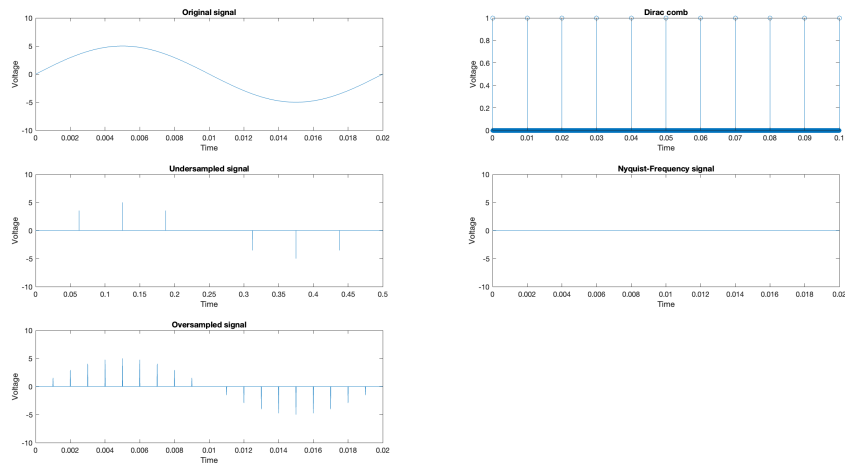

Figure 1.1: Impulse train sampling

Figure 1.2: Rectangular Pulse Train Sampling

The code used to perform the sampling is given below.

```
%% Prelab:

% first part:

f = 50; % frequency, in hertz
f_n = 2*f; % nyquist frequency, in hertz

t = 0:1E-5:1; % smaller iterations lead to closer to continuous behavior

V_pp = 5;
w = 2*pi*f;
x = V_pp * sin(w .* t);

% Original signal:
subplot(3, 2, 1);
plot(t, x);
ylabel("Voltage")
xlabel("Time")
title("Original signal");
ylim([-10, 10]);
xlim([0, 1/f]);

% Dirac comb
T = 1/f_n;
subplot(3, 2, 2);
p = zeros(size(t));
p(mod(t, T) == 0) = 1;
stem(t, p);
ylabel("Voltage")
xlabel("Time")
title("Dirac comb");
ylim([0, 1]);
xlim([0, 10*T]);

% Undersampled signal:
fs = 48; % In hertz
```

```
T = 1/fs;

p = zeros(size(t));
p(mod(t, T) == 0) = 1;

subplot(3, 2, 3);
plot(t, p .* x);
ylabel("Voltage")
xlabel("Time")
title("Undersampled signal");
ylim([-10, 10]);
xlim([0, 1/2]);

% Nyquist-Frequency sampled signal
fs = 100; % In hertz
T = 1/fs;

p = zeros(size(t));
p(mod(t, T) == 0) = 1;

subplot(3, 2, 4);
plot(t, p .* x);
ylabel("Voltage")
xlabel("Time")
title("Nyquist-Frequency signal");

ylim([-10, 10]);
xlim([0, 1/50]);


% Oversampled signal
fs = 1000; % In hertz
T = 1/fs;

p = zeros(size(t));
p(mod(t, T) == 0) = 1;

subplot(3, 2, 5);
```

```matlab
plot(t, p .* x);
ylabel("Voltage")
xlabel("Time")
title("Oversampled signal");

ylim([-10, 10]);
xlim([0, 1/50]);

%%

clear
close all

f = 50;
t = 0:1E-5:1;
V_pp = 5;
w = 2*pi*f;
x = V_pp * sin(w .* t);

% Underdamped signal:
fs = 48; % In hertz
T = 1/fs;
T0 = 0.5 * 1/fs;
p = gen_pulse_train(t, T0, T);

subplot(3, 2, 1);
plot(t, p);
title("Rectangular Pulse Train(Undersampled)");
ylabel("Voltage")
xlabel("Time")
xlim([0, 1/2]);
ylim([-0.5, 2]);

subplot(3, 2, 2);
plot(t, p .* x);
ylabel("Voltage")
xlabel("Time")
title("Undersampled Signal")
```

```matlab
xlim([0, 1/2]);
ylim([-6, 6]);


% Nyquist Damped signal:
fs = 100; % In hertz
T = 1/fs;
T0 = 0.5 * 1/fs;
p = gen_pulse_train(t, T0, T);

subplot(3, 2, 3);
plot(t, p .* x);
ylabel("Voltage")
xlabel("Time")
title("Nyquist Sampled Signal")
xlim([0, 1/50]);
ylim([-6, 6]);

% Nyquist Sampled signal:
fs = 1000; % In hertz
T = 1/fs;
T0 = 0.5 * 1/fs;
p = gen_pulse_train(t, T0, T);

subplot(3, 2, 4);
plot(t, p .* x);
ylabel("Voltage")
xlabel("Time")
title("Oversampled Signal")
xlim([0, 1/50]);
ylim([-6, 6]);

subplot(3, 2, [5 6]);
plot(t, x);
ylabel("Voltage")
xlabel("Time")
title("Original Signal")
xlim([0, 1/50]);
```

```
ylim([-6, 6]);

% Function to generate pulse train
function [p] = gen_pulse_train(t, T0, T)
      p = zeros(size(t));
      for k=1:length(t)
            if mod(t(k), T) == 0
            left = t(k) - T0;
            right = t(k) + T0;
            p((t > left) & (t < right)) = 1;
            else
            p(k) = 0;
            end
      end
end
```

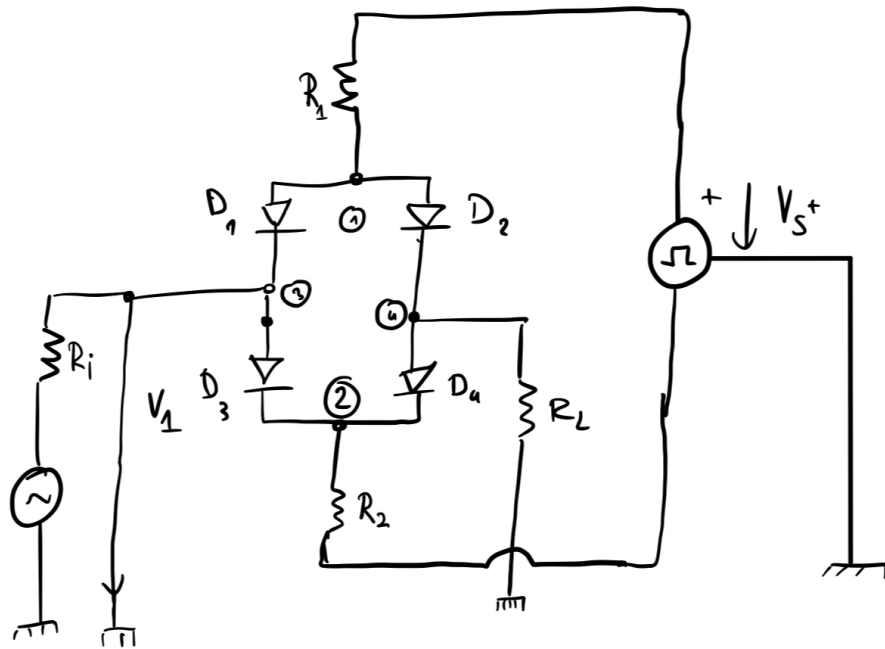## 1.3 Problem 3: Sampling using a Sampling bridge

Figure 1.3: The modified Sampling Bridge circuit.

The reason this sampling bridge circuit works is due to the polarity of the two different voltage sources given in the original circuit being practically the same, with the potential being the same for both sources, we can simply

omit one and lead both currents into the same node, which is what we did in the modified circuit.