```matlab
%{
    Matlab Practical 4 by Zach Vig
%}

clear;
clearvars;

%Part 1:
%q1.1
Fs = 512;

%q1.2
t = 1/Fs : 1/Fs : 2;

%q1.3
figure(1);subplot(2,1,1);
yfreq = 30;
y = sin(2*pi*yfreq.*t);
plot(t,y);title('q1.3');
xlabel('Time (s)');ylabel('Signal');

%q1.4
%{
    I would probably figure out how long one period of the sine wave takes,
    then take the inverse of that number to get the frequency.
%}

%q1.5
figure(2);
yft = fft(y)./length(t);
P = yft.*conj(yft);
plot(t,P);title('q1.5');

%q1.6
n = 0:length(y)-1;
w = n./(range(t));

%q1.7
figure(1);subplot(2,1,2);
set(gca);
hold on;
plot(w,P,'DisplayName','PowerSpectrum','color','k');
xlabel('Frequency (Hz)');ylabel('Power');

%q1.8
h = spectrum.periodogram;
ps = msspectrum(h,y,'Fs',Fs);
plot(ps.Frequencies,ps.Data,'DisplayName','Periodogram','color','r');
legend();

%q1.9
%{
```

```matlab
    The power spectrum peaks at both 30 and at 482 Hz, which correspond to
the input frequency and to the sampling frequency - the input frequency. The
Mean Squared spectrum only peaks at 30 Hz, but with twice the intensity.
%}

%q1.10
yfreq2 = 40;
y2 = sin(2*pi*yfreq2.*t);
figure(3);subplot(2,1,1);
plot(t,y2);title('q1.10');
xlabel('Time (s)');ylabel('Signal');

subplot(2,1,2);
hold on;
y2ft = fft(y2)./length(t);
P = y2ft.*conj(y2ft);
plot(w,P);title('q1.10 Power')
xlabel('Frequency (Hz)');ylabel('Power');

h2 = spectrum.periodogram;
ps2 = msspectrum(h2,y2,'Fs',Fs);
plot(ps2.Frequencies,ps2.Data,'r--');

%q.11
figure(4);subplot(2,1,1);
y3 = y+y2;
plot(t,y3);title('q1.11')
xlabel('Time (s)');ylabel('Signal');

subplot(2,1,2);
hold on;
y3ft = fft(y3)./length(t);
P = y3ft.*conj(y3ft);
plot(w,P);
xlabel('Frequency (Hz)');ylabel('Power');

h3 = spectrum.periodogram;
ps3 = msspectrum(h3,y3,'Fs',Fs);
plot(ps3.Frequencies,ps3.Data,'r--');

%Part 2:
%q2.1
%The nyquist frequency is 512/2 = 256 Hz.

%q2.2
y255 = sin(2*pi*255 .* t);
figure(5); subplot(2,1,1);
plot(t,y255);title('q2.2');
xlabel('Time (s)');ylabel('Signal')
y255ft = fft(y255)./length(t);
P255 = y255ft .* conj(y255ft);
subplot(2,1,2);
plot(w,P255);
xlabel('Frequency (Hz)'); ylabel('Power');
```

```matlab
%The power spectrum peaks at both 255 Hz and 257 Hz. The peak at 255 Hz
is due to the frequency of the signal, and the peak at 257 is the complex
conjugate value, since 512-255 = 257 Hz.

%q2.3
y300 = sin(2*pi*300 .* t);
figure(6); subplot(2,1,1);
plot(t,y300);title('q2.3');
xlabel('Time (s)');ylabel('Signal')
y300ft = fft(y300)./length(t);
P300 = y300ft .* conj(y300ft);
subplot(2,1,2);
plot(w,P300);
xlabel('Frequency (Hz)'); ylabel('Power');
%Similar to the sub-nyquist frequency case, we see a peak at the actual
frequency, 300 Hz,  and at the sampling minus actual differential frequency,
212 Hz, but since the signal frequency is above the nyquist frequency, the
relative positions of these peaks are flipped, with the actual frequency
being higher than its complex conjugate.

%q2.4
y450 = sin(2*pi*450 .* t);
figure(7); subplot(2,1,1);
plot(t,y450);title('q2.4');
xlabel('Time (s)');ylabel('Signal')
y450ft = fft(y450)./length(t);
P450 = y450ft .* conj(y450ft);
subplot(2,1,2);
plot(w,P450);
xlabel('Frequency (Hz)'); ylabel('Power');
%Here, we see a similar phenomenon as question 5, with two peaks at 450 Hz
and 62 Hz.

%q2.5
y800 = sin(2*pi*800 .* t);
figure(8); subplot(2,1,1);
plot(t,y800);title('q2.5');
xlabel('Time (s)');ylabel('Signal')
y800ft = fft(y800)./length(t);
P800 = y800ft .* conj(y800ft);
subplot(2,1,2);
plot(w,P800);
xlabel('Frequency (Hz)'); ylabel('Power');
%As the signal frequency continues to increase past the sampling frequency
itself, the aliasing effect changes. We no longer have a peak at the actual
frequency, but at the absolute value of the sample frequency minus the
signal frequency, 288 Hz, and at the sampling frequency minus this value,
224 Hz. In practice, this effect is as if the signal frequency was, in fact,
the signal frequency minus the sampling frequency.

%q2.6
%A higher-frequency signal will look exactly like a lower-frequency signal
when analyzing it in the frequency domain if aliasing is present. The exact
relationship between the aliased signal and the actual signal is some kind
```

```matlab
of difference between the sampling frequency and the signal frequency. At
frequencies greater than the sampling frequency, the relationship becomes
slightly more complicated, but still persists in masquerading the high
frequency signal.

%Part 3:
%q3.1
%See Above

%q3.2
y = sin(2*pi*5 .* t);
figure(9);subplot(2,1,1);
plot(t,y,'color','b','DisplayName','No Noise');
xlabel('Time (s)'); ylabel('Signal')
subplot(2,1,2);
yft = fft(y);
P = yft .* conj(yft);
plot(w,P,'color','b','DisplayName','No Noise');
xlabel('Frequency (Hz)'); ylabel('Power');
%The power spectrum peaks at the expected values of 5 Hz and 507 Hz, due to
the reasons mentioned above (e.g. q2.2).

%q3.3
n = normrnd(0,0.1,1,length(t));
yn = y + n;
figure(9);subplot(2,1,1);hold on;
j1 = plot(t,yn,'color','r','DisplayName','Low Noise');
uistack(j1,'bottom');

ynft = fft(yn);
Pn = ynft .* conj(ynft);
figure(9);subplot(2,1,2);hold on;
plot(w,Pn,'color','r','DisplayName','Low Noise');

%q3.4
figure(9);subplot(2,1,1);hold on;
n2 = normrnd(0,2,1,length(t));
yn2 = y + n2;
j1 = plot(t,yn2,'color','g','DisplayName','High Noise');
uistack(j1,'bottom');

yn2ft = fft(yn2);
Pn2 = yn2ft .* conj(yn2ft);
figure(9);subplot(2,1,2);hold on;
plot(w,Pn2,'color','g','DisplayName','High Noise');
%The power spectrum still peaks at the correct values, but there is a
considerable amount of noise between the peak frequencies.


%q3.5
figure(9);subplot(2,1,1);hold on;
n_amplitude = 5;
n3 = normrnd(0,n_amplitude,1,length(t));
yn3 = y + n3;
```

```matlab
j1 = plot(t,yn3,'color','magenta','DisplayName','Very High Noise');
uistack(j1,'bottom');
legend();

figure(9);subplot(2,1,2);hold on;
yn3ft = fft(yn3);
Pn3 = yn3ft .* conj(yn3ft);
plot(w,Pn3,'color','magenta','DisplayName','Very High Noise');
legend();

%a. Between noise levels of about 5-6, it becomes impossible to pick out the
periodicity of the sine wave in the time domain.
%b. The peaks in the power spectrum become indistinguishable from background
noise at around the same noise level that the periodicity disappers in the
time domain.
```

```
                ALim: {}
            ALimMode: {'auto'  'manual'}
          AlphaScale: {'linear'  'log'}
            Alphamap: {}
    AmbientLightColor: {1x0 cell}
                 Box: {[on]  [off]}
            BoxStyle: {'full'  'back'}
           BusyAction: {'queue'  'cancel'}
        ButtonDownFcn: {}
                CLim: {}
            CLimMode: {'auto'  'manual'}
      CameraPosition: {}
   CameraPositionMode: {'auto'  'manual'}
        CameraTarget: {}
      CameraTargetMode: {'auto'  'manual'}
      CameraUpVector: {}
    CameraUpVectorMode: {'auto'  'manual'}
      CameraViewAngle: {}
  CameraViewAngleMode: {'auto'  'manual'}
             Children: {}
             Clipping: {[on]  [off]}
         ClippingStyle: {'rectangle'  '3dbox'}
                Color: {1x0 cell}
            ColorOrder: {}
       ColorOrderIndex: {}
           ColorScale: {'linear'  'log'}
             Colormap: {}
          ContextMenu: {}
            CreateFcn: {}
        DataAspectRatio: {}
     DataAspectRatioMode: {'auto'  'manual'}
            DeleteFcn: {}
            FontAngle: {'normal'  'italic'}
             FontName: {}
             FontSize: {}
         FontSizeMode: {'auto'  'manual'}
       FontSmoothing: {[on]  [off]}
             FontUnits: {1x5 cell}
```

```
               FontWeight: {'normal'  'bold'}
                 GridAlpha: {}
             GridAlphaMode: {'auto'  'manual'}
                 GridColor: {1x0 cell}
             GridColorMode: {'auto'  'manual'}
             GridLineStyle: {'-'  '--'  ':'  '-.'  'none'}
             GridLineWidth: {}
         GridLineWidthMode: {'auto'  'manual'}
          HandleVisibility: {'on'  'callback'  'off'}
                   HitTest: {[on]  [off]}
             InnerPosition: {}
        InteractionOptions: {}
              Interactions: {}
             Interruptible: {[on]  [off]}
    LabelFontSizeMultiplier: {}
                     Layer: {'bottom'  'top'}
                    Layout: {}
     LineStyleCyclingMethod: {'aftercolor'  'beforecolor'  'withcolor'}
            LineStyleOrder: {}
       LineStyleOrderIndex: {}
                 LineWidth: {}
            MinorGridAlpha: {}
        MinorGridAlphaMode: {'auto'  'manual'}
            MinorGridColor: {1x0 cell}
        MinorGridColorMode: {'auto'  'manual'}
        MinorGridLineStyle: {'-'  '--'  ':'  '-.'  'none'}
        MinorGridLineWidth: {}
    MinorGridLineWidthMode: {'auto'  'manual'}
                  NextPlot: {1x4 cell}
             OuterPosition: {}
                    Parent: {}
             PickableParts: {'visible'  'none'  'all'}
         PlotBoxAspectRatio: {}
     PlotBoxAspectRatioMode: {'auto'  'manual'}
                  Position: {}
        PositionConstraint: {'innerposition'  'outerposition'}
                Projection: {'orthographic'  'perspective'}
                  Selected: {[on]  [off]}
        SelectionHighlight: {[on]  [off]}
                SortMethod: {'depth'  'childorder'}
                  Subtitle: {}
        SubtitleFontWeight: {'normal'  'bold'}
                       Tag: {}
                   TickDir: {'in'  'out'  'both'  'none'}
               TickDirMode: {'auto'  'manual'}
       TickLabelInterpreter: {'none'  'tex'  'latex'}
                TickLength: {}
                     Title: {}
    TitleFontSizeMultiplier: {}
           TitleFontWeight: {'normal'  'bold'}
   TitleHorizontalAlignment: {'left'  'center'  'right'}
                   Toolbar: {}
                     Units: {1x6 cell}
                  UserData: {}
```

```
                    View: {}
                 Visible: {[on]  [off]}
                   XAxis: {}
           XAxisLocation: {'bottom'  'top'  'origin'}
                  XColor: {1x0 cell}
              XColorMode: {'auto'  'manual'}
                    XDir: {'normal'  'reverse'}
                   XGrid: {[on]  [off]}
                  XLabel: {}
                    XLim: {}
                XLimMode: {'auto'  'manual'}
            XLimitMethod: {'tickaligned'  'tight'  'padded'}
              XMinorGrid: {[on]  [off]}
              XMinorTick: {[on]  [off]}
                  XScale: {'linear'  'log'}
                   XTick: {}
              XTickLabel: {}
          XTickLabelMode: {'auto'  'manual'}
       XTickLabelRotation: {}
   XTickLabelRotationMode: {'auto'  'manual'}
               XTickMode: {'auto'  'manual'}
           YAxisLocation: {'left'  'right'  'origin'}
                  YColor: {1x0 cell}
              YColorMode: {'auto'  'manual'}
                    YDir: {'normal'  'reverse'}
                   YGrid: {[on]  [off]}
                  YLabel: {}
                    YLim: {}
                YLimMode: {'auto'  'manual'}
            YLimitMethod: {'tickaligned'  'tight'  'padded'}
              YMinorGrid: {[on]  [off]}
              YMinorTick: {[on]  [off]}
                  YScale: {'linear'  'log'}
                   YTick: {}
              YTickLabel: {}
          YTickLabelMode: {'auto'  'manual'}
       YTickLabelRotation: {}
   YTickLabelRotationMode: {'auto'  'manual'}
               YTickMode: {'auto'  'manual'}
                   ZAxis: {}
                  ZColor: {1x0 cell}
              ZColorMode: {'auto'  'manual'}
                    ZDir: {'normal'  'reverse'}
                   ZGrid: {[on]  [off]}
                  ZLabel: {}
                    ZLim: {}
                ZLimMode: {'auto'  'manual'}
            ZLimitMethod: {'tickaligned'  'tight'  'padded'}
              ZMinorGrid: {[on]  [off]}
              ZMinorTick: {[on]  [off]}
                  ZScale: {'linear'  'log'}
                   ZTick: {}
              ZTickLabel: {}
          ZTickLabelMode: {'auto'  'manual'}
```
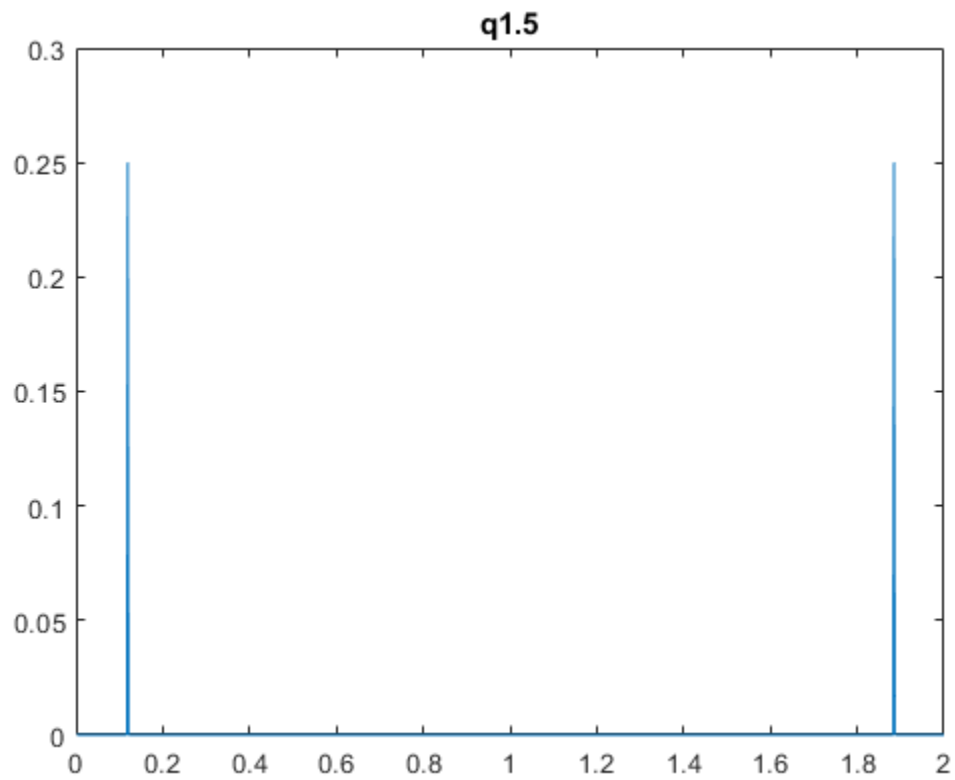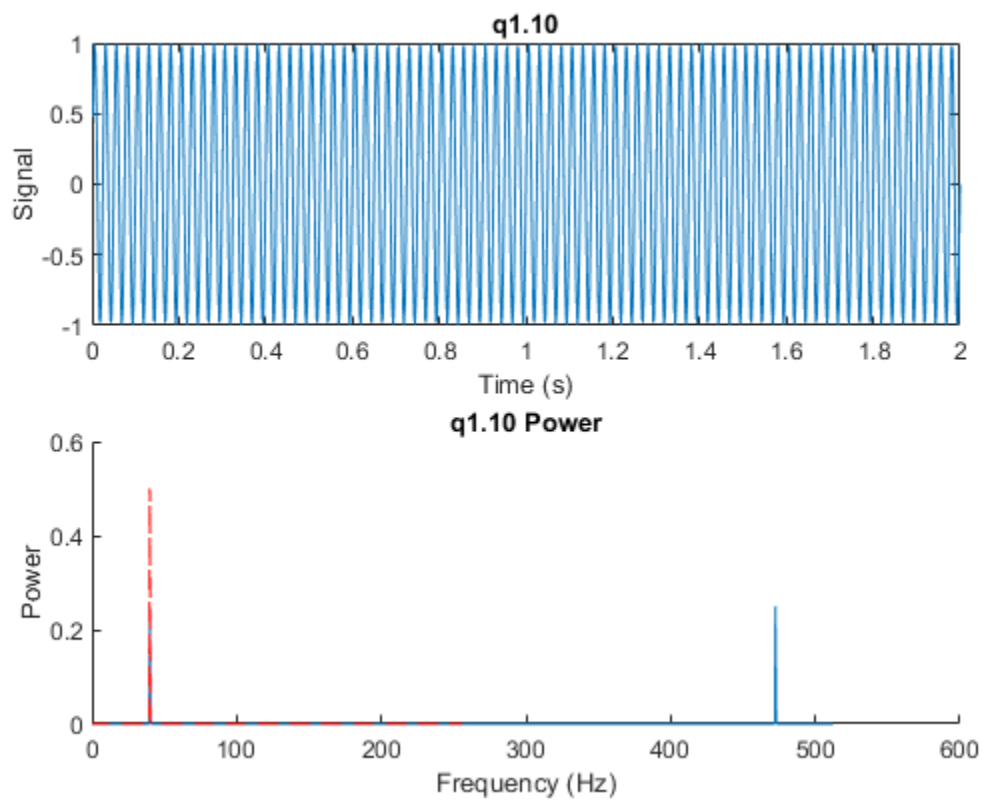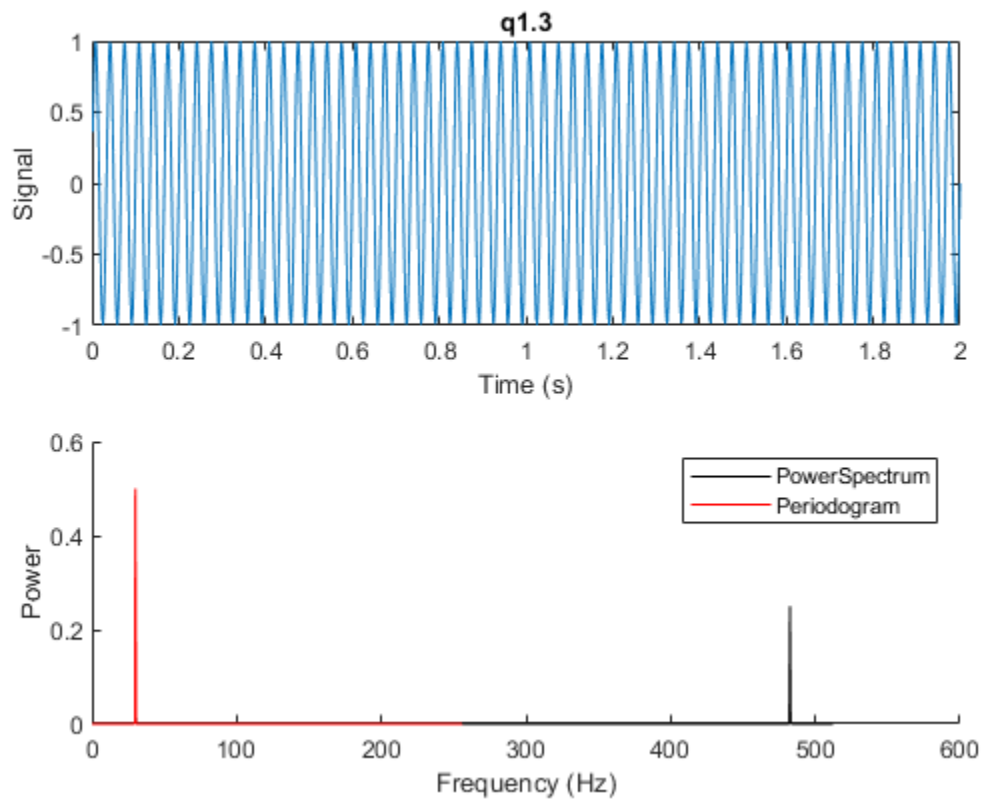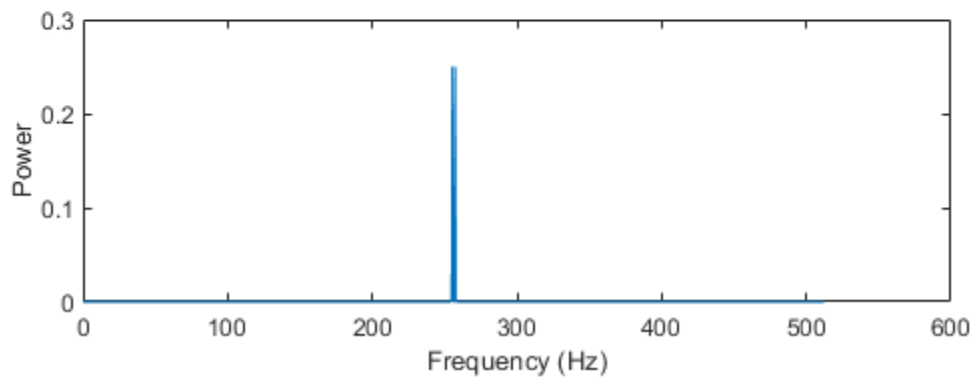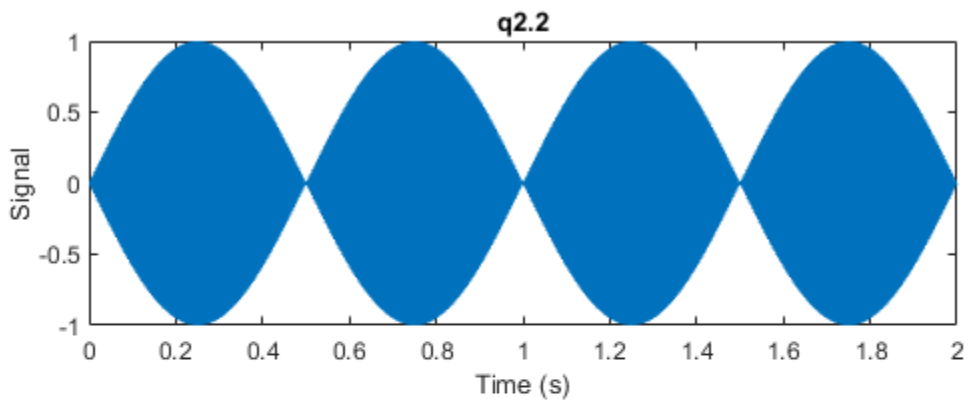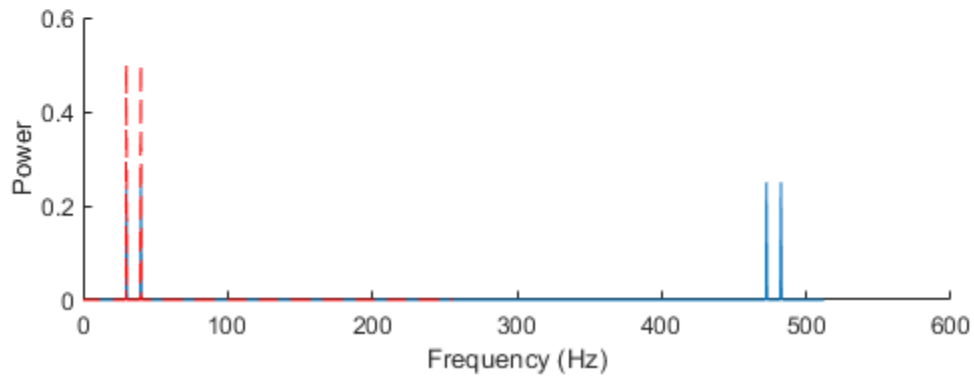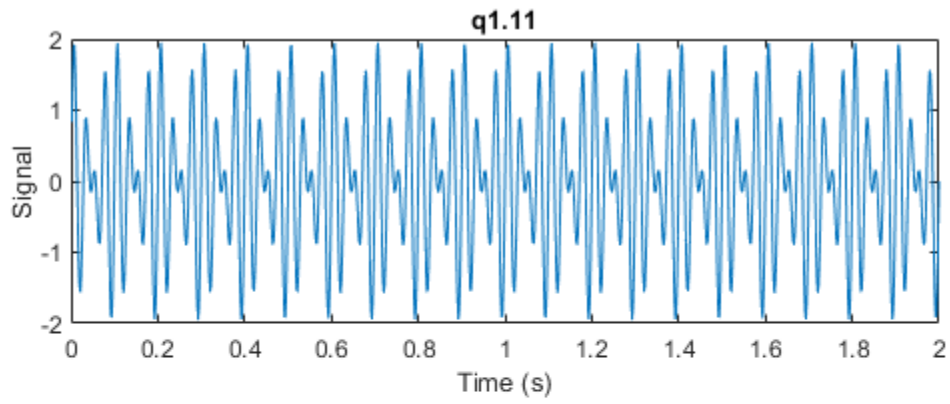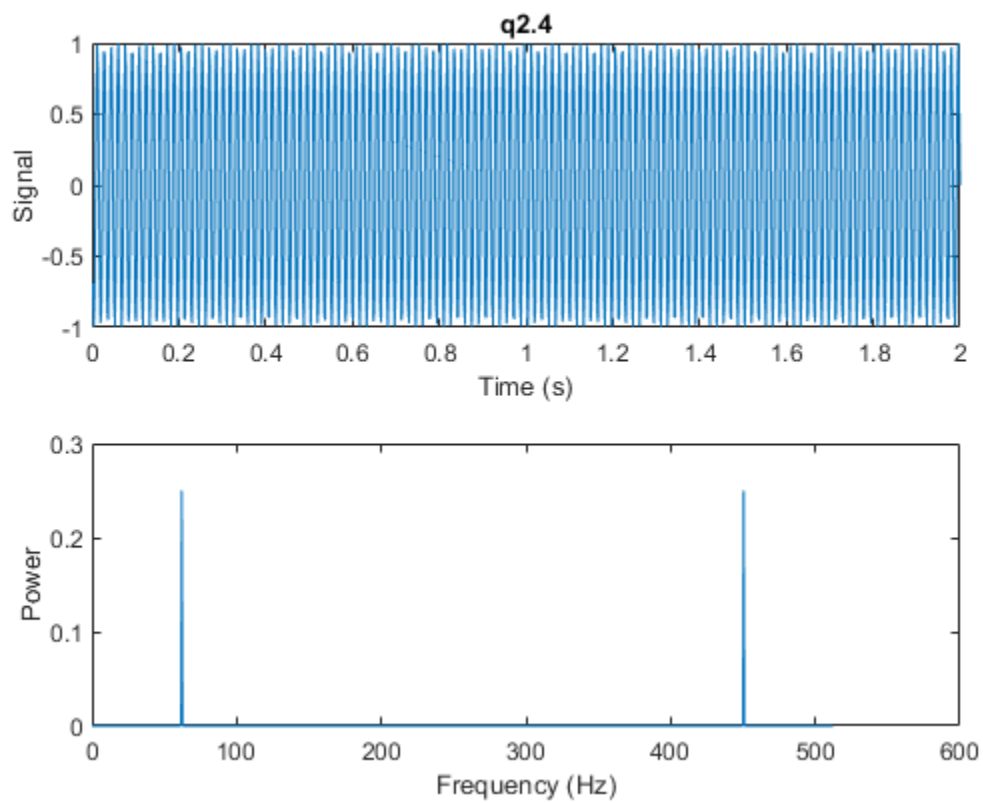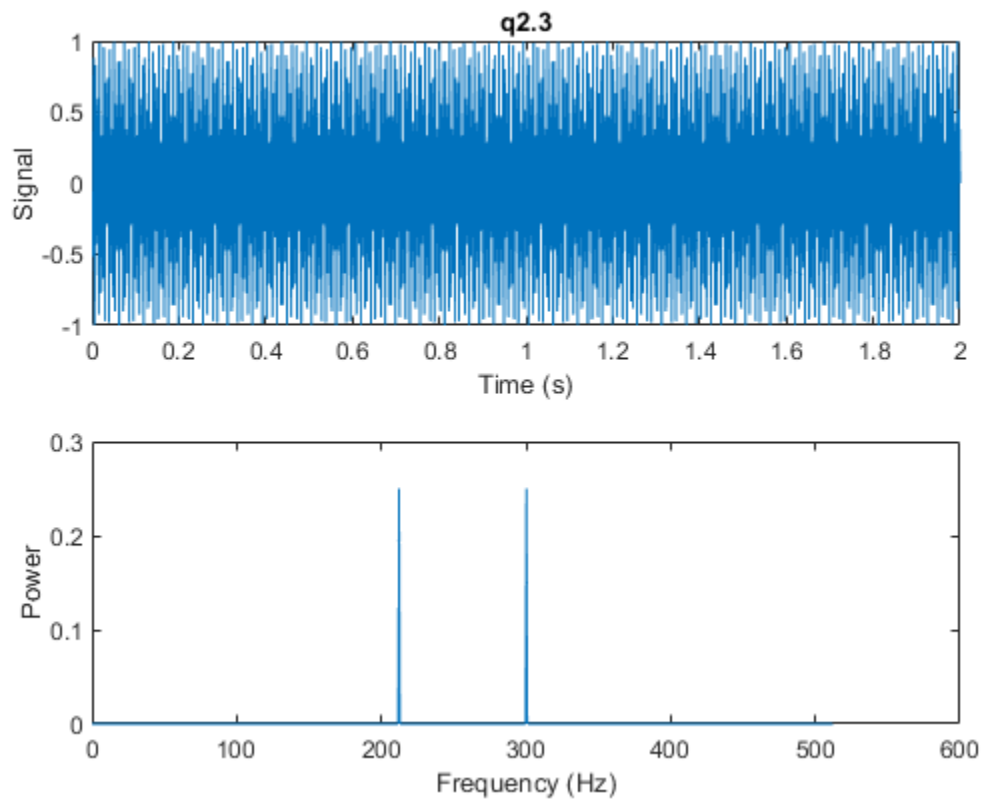
```
            ZTickLabelRotation: {}
    ZTickLabelRotationMode: {'auto'    'manual'}
                 ZTickMode: {'auto'    'manual'}
```

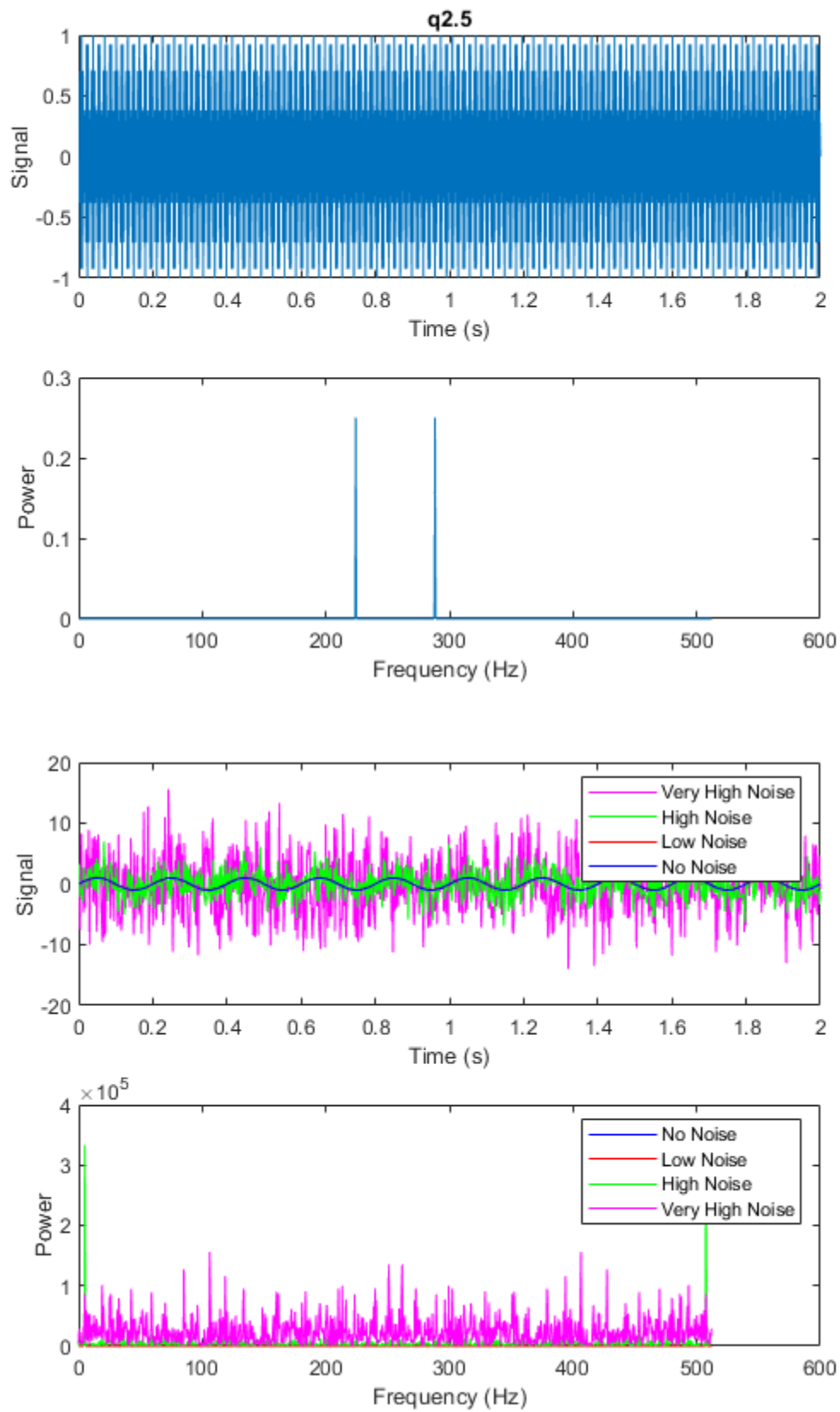**q1.5**