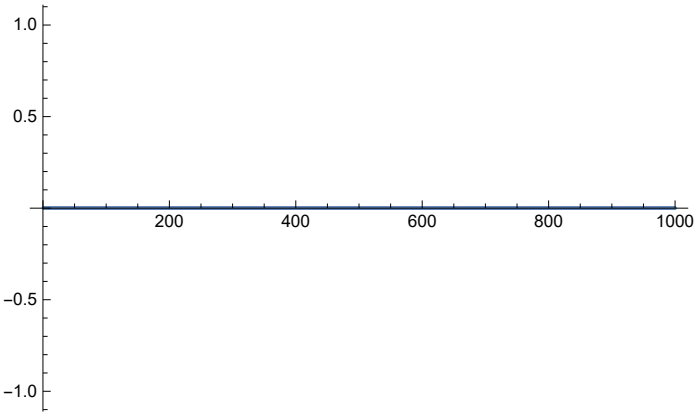
In[1]:=
```
constantSignalValue = 0.3;
(* from here we assume sample count of 1000 *)
sampleCount = 1000;
originalConstantSignal = constantSignalValue & /@ Range[sampleCount];
(* 1 bit quantization of constant signal of 1000 samples *)
quantizedSignal = Round[constantSignalValue] & /@ Range[sampleCount];
SetDirectory@NotebookDirectory[];
```

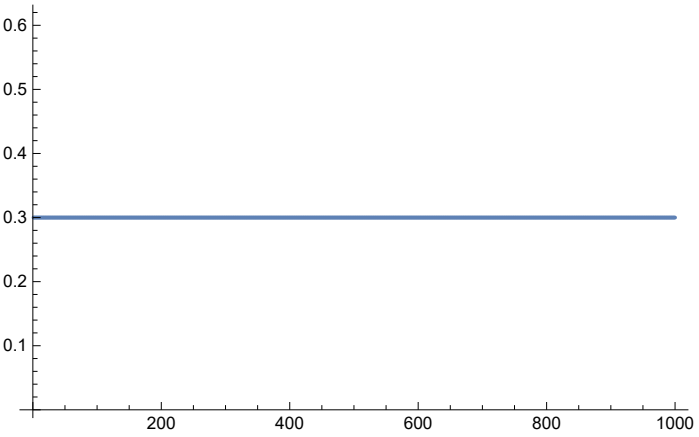
In[6]:=
```
signalError = originalConstantSignal - quantizedSignal;
```

In[7]:=
```
ListPlot[quantizedSignal]
ListPlot[signalError]
(* Plot of quantized signal *)
```

Out[7]=



Out[8]=



In[9]:=
```
(* Plot of quantization error *)
```

In[10]:=
```
(*As expected, average of signal error is 0.3 *)
Mean[signalError]
```

Out[10]= 0.3

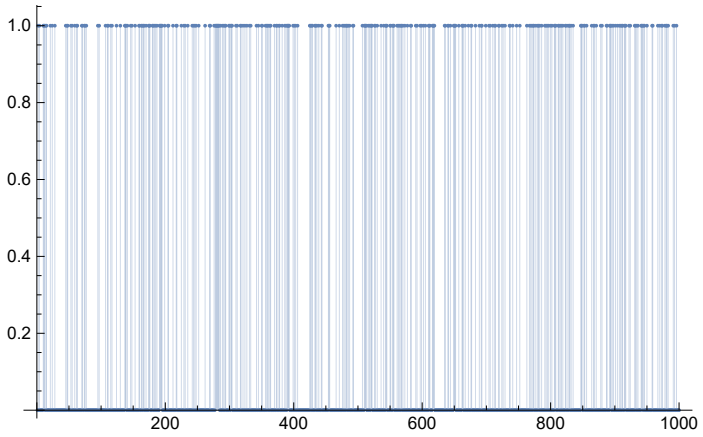
In[11]:=
```
(* Same qantization, but with dithering in range -0.5, 0.5 *)
```

In[12]:=
```
quantizedDitheredSignal =
  Round[constantSignalValue + RandomReal[] - 0.5] & /@ Range[sampleCount];
```

In[13]:=
```
(* Plot of quantized dithered signal *)
```

In[14]:=
```
ListPlot[quantizedDitheredSignal, Filling → Axis]
```

Out[14]=



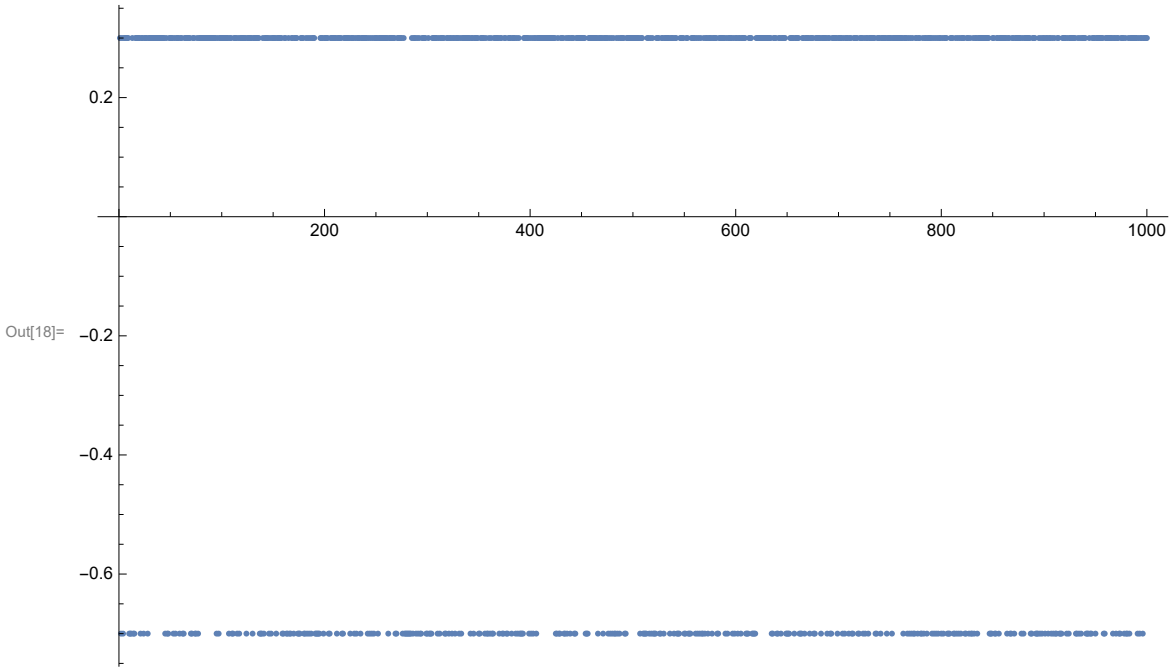
In[15]:=
```
Image[Join[{quantizedDitheredSignal}, {quantizedDitheredSignal},
  {quantizedDitheredSignal}, {quantizedDitheredSignal}, {quantizedDitheredSignal},
  {quantizedDitheredSignal}, {quantizedDitheredSignal}, {quantizedDitheredSignal},
  {quantizedDitheredSignal}, {quantizedDitheredSignal}, {quantizedDitheredSignal},
  {quantizedDitheredSignal}, {quantizedDitheredSignal}, {quantizedDitheredSignal},
  {quantizedDitheredSignal}, {quantizedDitheredSignal}, 1]]
```

Out[15]=

In[16]:=
```
(*Dithered signal error *)
```

In[17]:=
```
ditheredSignalError = originalConstantSignal - quantizedDitheredSignal;
ListPlot[ditheredSignalError]
```

Out[18]=



In[19]:=
```
Mean[ditheredSignalError]
```

Out[19]= −0.022

In[20]:=
```
0.02800000000000019
```

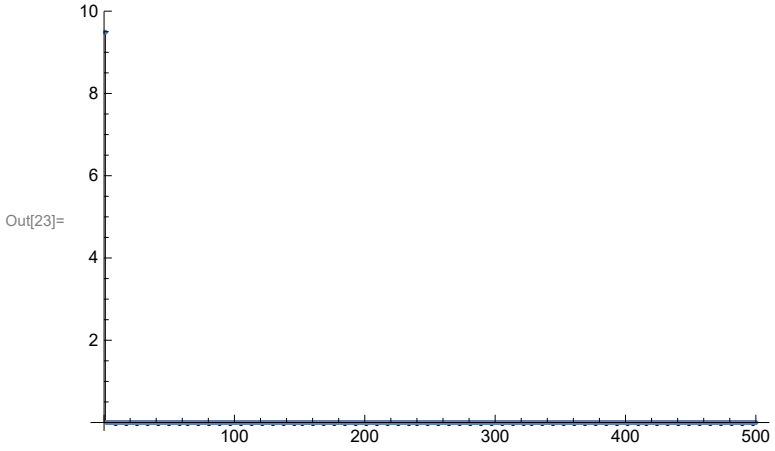
Out[20]= 0.028

In[21]:=
```
MyPeriodogram[x_, col_: Black, range_: Full] :=
 ListPlot[Abs@Fourier[x][[ ;; sampleCount / 2]], PlotRange → range,
  Filling → Axis, FillingStyle → {Thickness[0.05], col}]
```

In[22]:=
```
(*Frequency plot of error signal*)
```

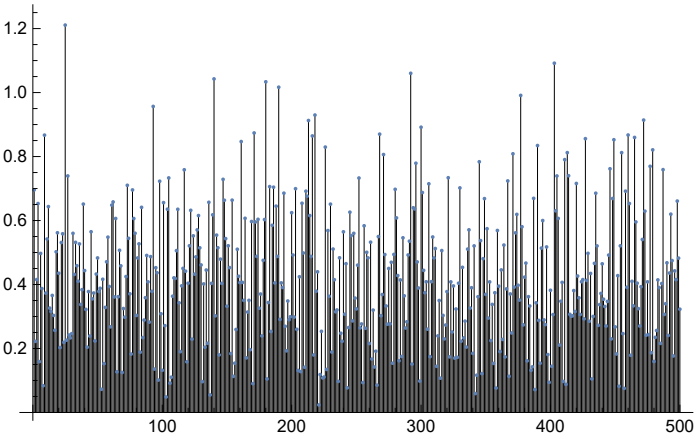
In[23]:=
```
MyPeriodogram[signalError]
```

Out[23]=



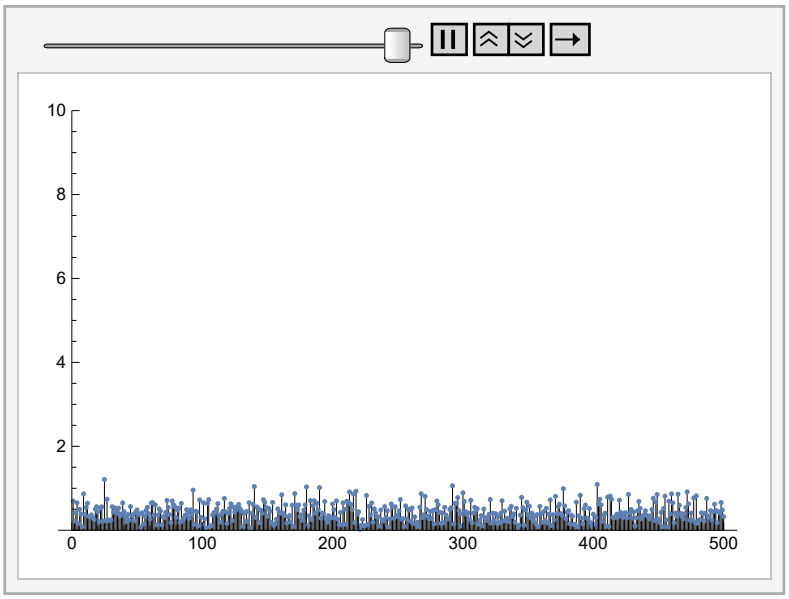
In[24]:=
```
MyPeriodogram[ditheredSignalError]
```

Out[24]=



In[25]:=
```
ListAnimate[{MyPeriodogram[signalError, Red, {0, 10}],
  MyPeriodogram[ditheredSignalError, Black, {0, 10}]}]
```

Out[25]=



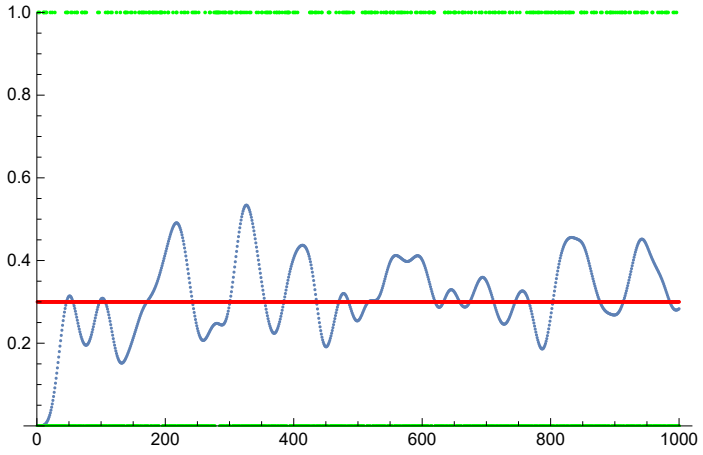
In[26]:=
```
Export["spectrum_quantization_noise_comparison.gif",
  {MyPeriodogram[signalError, Red, {0, 10}],
   MyPeriodogram[ditheredSignalError, Black, {0, 10}]}, "DisplayDurations" → 2];
```

In[27]:=
```
MyFilter[x_] := RecurrenceFilter[
  ToDiscreteTimeModel[ButterworthFilterModel[{5, 0.1}], 1], x, Padding → "Fixed"]
```

In[28]:=
```
(* Plot low pass filter applied on quantized signal and its error *)
```

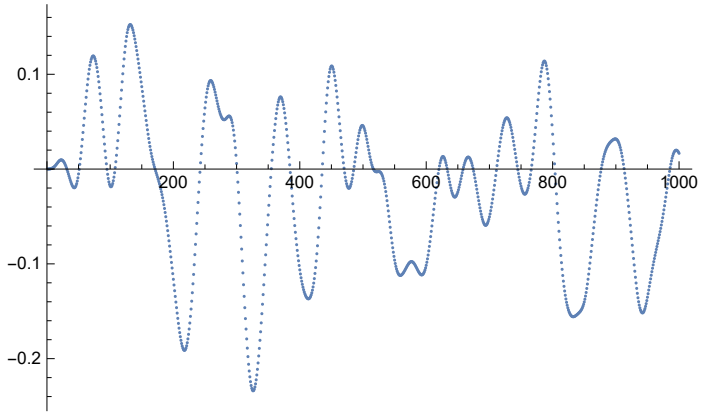
In[29]:=
```
Show[ListPlot[MyFilter[quantizedDitheredSignal], PlotRange → {0, 1}],
 ListPlot[originalConstantSignal, PlotStyle → Red],
 ListPlot[quantizedDitheredSignal, PlotStyle → Green]]
```

Out[29]=



In[30]:=
```
ListPlot[MyFilter[ditheredSignalError]]
```

Out[30]=



In[31]:=
```
(*Let's try a different dithering noise function*)
```

In[32]:=
```
quantizedDitheredSignalGolden =
  Round[constantSignalValue + FractionalPart[GoldenRatio * #1] - 0.5] & /@
   Range[sampleCount];
```

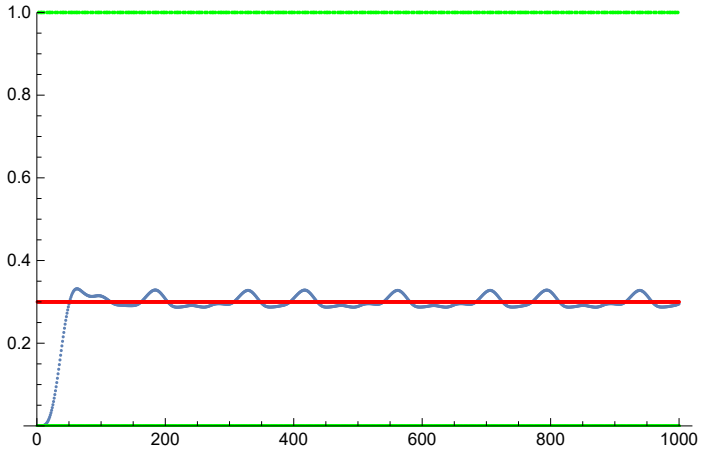
In[33]:=
```
ditheredSignalErrorGolden = originalConstantSignal - quantizedDitheredSignalGolden;
```

In[34]:=
```
Mean[ditheredSignalErrorGolden]
```

Out[34]=
```
-0.001
```

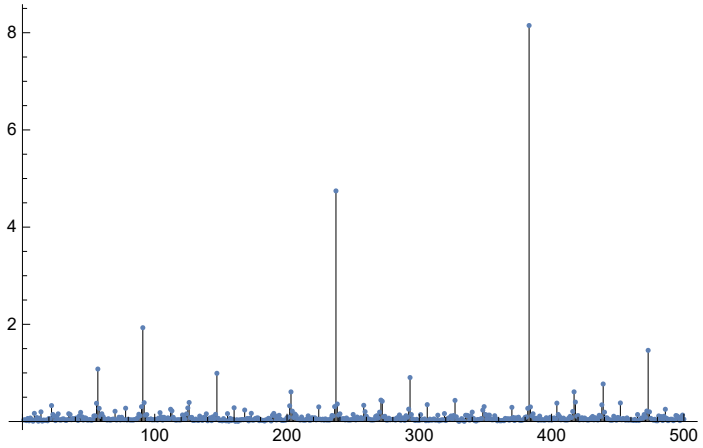
In[35]:=
```
Show[ListPlot[MyFilter[quantizedDitheredSignalGolden], PlotRange → {0, 1}],
 ListPlot[originalConstantSignal, PlotStyle → Red],
 ListPlot[quantizedDitheredSignalGolden, PlotStyle → Green]]
```

Out[35]=



In[36]:=
```
MyPeriodogram[ditheredSignalErrorGolden]
```

Out[36]=



In[37]:=
```
Image[Join[{quantizedDitheredSignalGolden}, {quantizedDitheredSignalGolden},
  {quantizedDitheredSignalGolden}, {quantizedDitheredSignalGolden},
  {quantizedDitheredSignalGolden}, {quantizedDitheredSignalGolden},
  {quantizedDitheredSignalGolden}, {quantizedDitheredSignalGolden},
  {quantizedDitheredSignalGolden}, {quantizedDitheredSignalGolden},
  {quantizedDitheredSignalGolden}, {quantizedDitheredSignalGolden}, 1]]
```

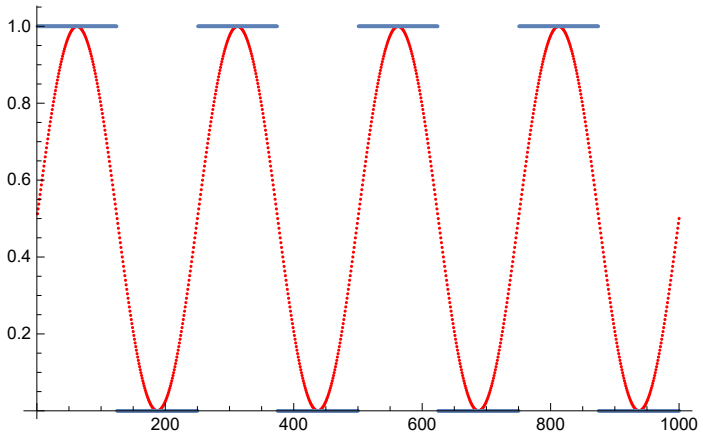
Out[37]=

In[38]:=
```
sineFunction[x_] := 0.5 + Sin[x / 250 * 2 * π] * 0.5;

originalSineSignal = sineFunction[#1] & /@ Range[sampleCount];
(* 1 bit quantization of sine signal*)
quantizedSineSignal = Round[sineFunction[#1]] & /@ Range[sampleCount];
```

In[41]:=
```
Show[ListPlot[originalSineSignal, PlotStyle → Red],
 ListPlot[quantizedSineSignal]]
```

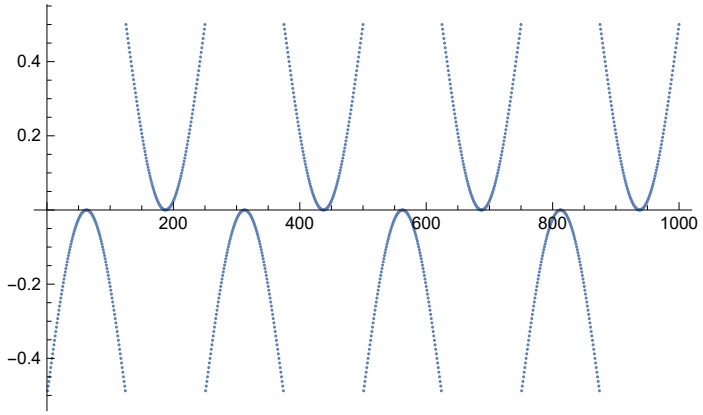
Out[41]=



In[42]:=
```
sineSignalError = originalSineSignal - quantizedSineSignal;
```

In[43]:=
```
ListPlot[sineSignalError]
```

Out[43]=



In[44]:=
```
(*Average error... not bad, huh?*)
```

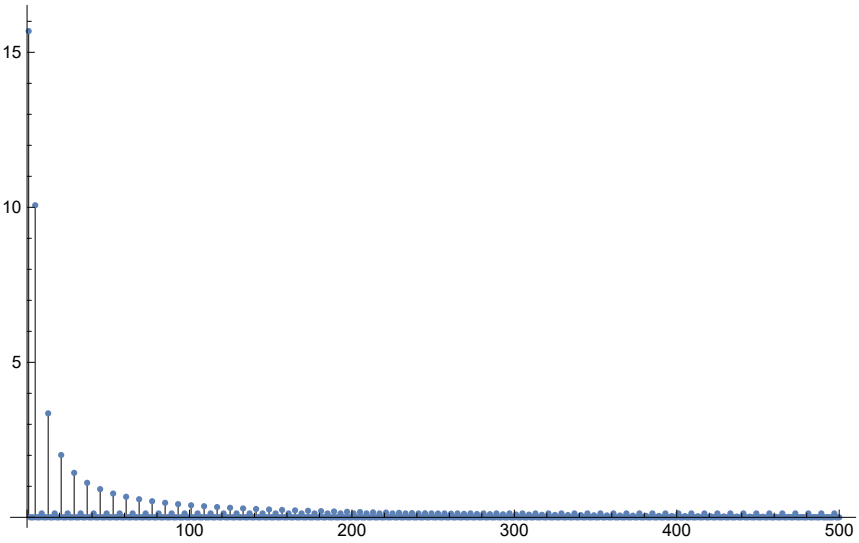
In[45]:=
```
Mean[sineSignalError]
```

Out[45]= 0.004

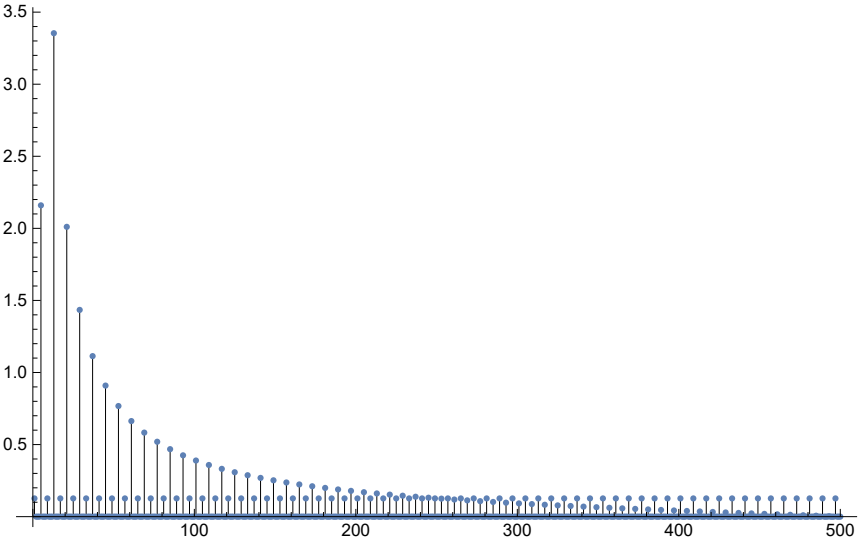
In[46]:=
```
(*But error has lots of low and high frequencies*)
```

In[47]:=
```
MyPeriodogram[quantizedSineSignal]
MyPeriodogram[sineSignalError]
```

Out[47]=

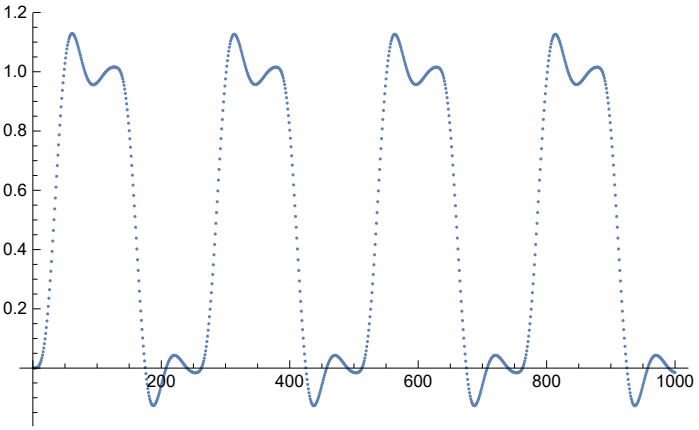


Out[48]=



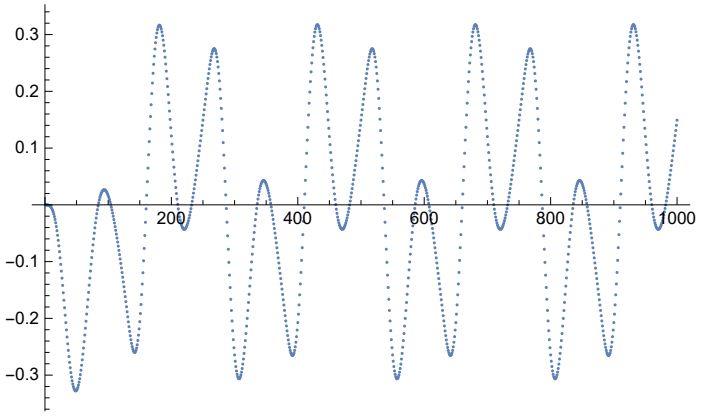
In[49]:=
```
ListPlot[MyFilter[quantizedSineSignal]]
```

Out[49]=



In[50]:=
```
ListPlot[MyFilter[sineSignalError]]
```

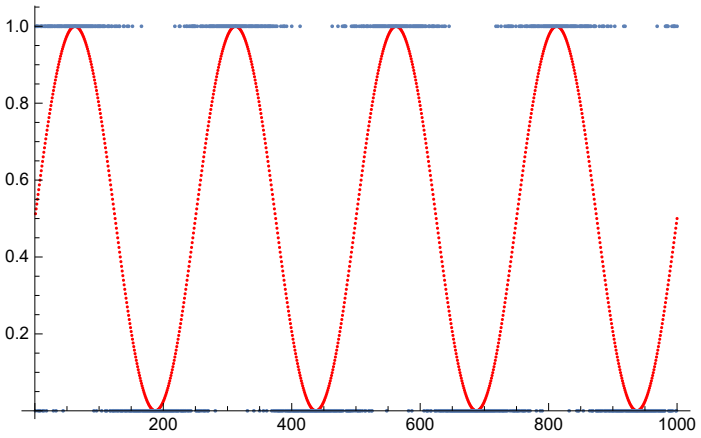
Out[50]=



In[51]:=
```
(* 1 bit quantization of dithered sine signal*)
quantizedDitheredSineSignal =
  Round[sineFunction[#1] + RandomReal[] - 0.5] & /@ Range[sampleCount];
```

In[52]:=
```
Show[ListPlot[originalSineSignal, PlotStyle → Red],
 ListPlot[quantizedDitheredSineSignal]]
```

Out[52]=

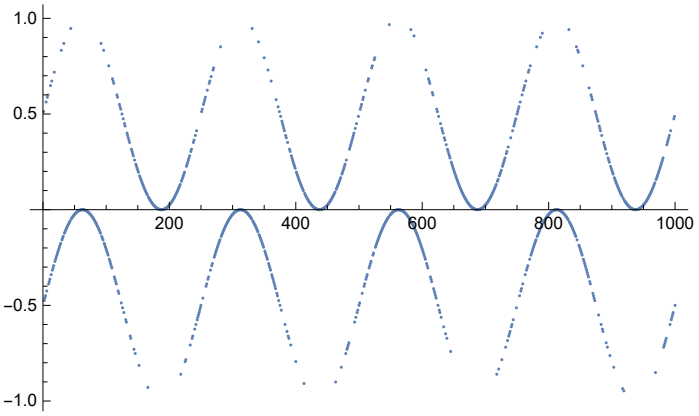


In[53]:=
```
Image[Join[{quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal}, 1]]
ditheredSineSignalError = originalSineSignal - quantizedDitheredSineSignal;
```

Out[53]=

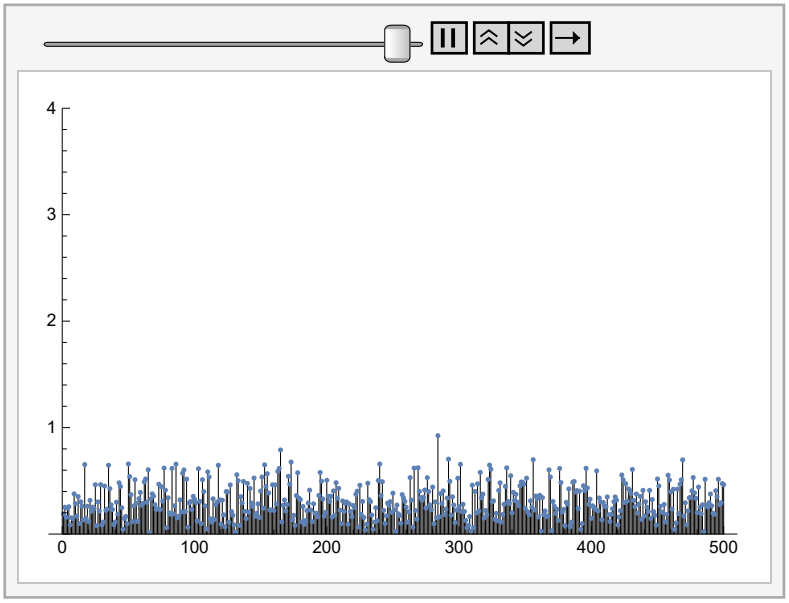
In[55]:=
```
ListPlot[ditheredSineSignalError]
```

Out[55]=



In[56]:=
```
ListAnimate[{MyPeriodogram[sineSignalError, Red, {0, 4}],
  MyPeriodogram[ditheredSineSignalError, Black, {0, 4}]}]
```

Out[56]=



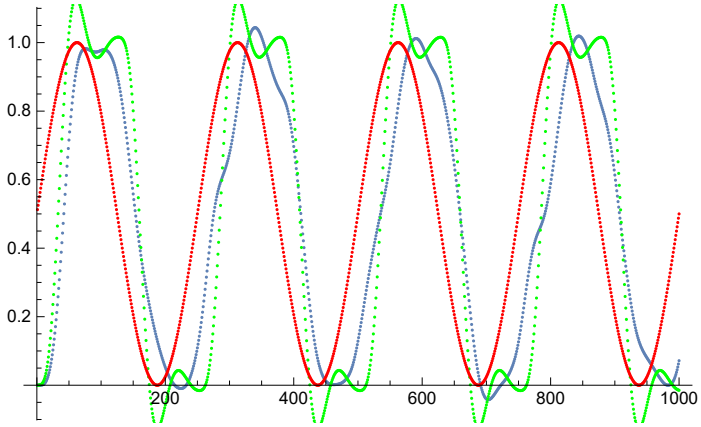
In[57]:=
```
Export["spectrum_quantization_noise_comparison_sine.gif",
  {MyPeriodogram[sineSignalError, Red, {0, 4}],
   MyPeriodogram[ditheredSineSignalError, Black, {0, 4}]}, "DisplayDurations" → 2];
```

In[58]:=
```
Mean[ditheredSineSignalError]
```

Out[58]= 0.006

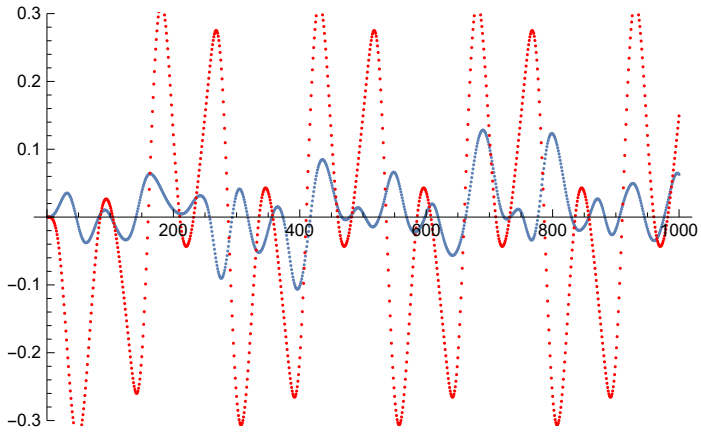
In[59]:=
```
Show[ListPlot[MyFilter[quantizedDitheredSineSignal]],
 ListPlot[MyFilter[quantizedSineSignal], PlotStyle → Green],
 ListPlot[originalSineSignal, PlotStyle → Red]]
```

Out[59]=



In[60]:=
```
Show[ListPlot[MyFilter[ditheredSineSignalError], PlotRange → {-0.3, 0.3}],
 ListPlot[MyFilter[sineSignalError], PlotStyle → Red]]
```

Out[60]=



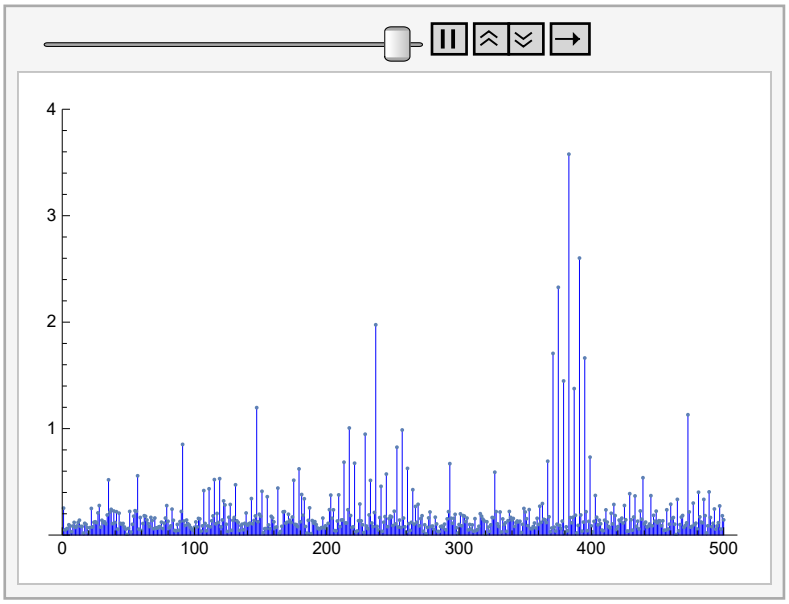
In[61]:=
```
quantizedDitheredSineSignalGolden =
  Round[sineFunction[#1] + FractionalPart[GoldenRatio * #1] - 0.5] & /@
   Range[sampleCount];
Image[Join[{quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden}, 1]]
```

Out[62]=

In[63]:=
```
ditheredSineSignalGoldenError =
  originalSineSignal - quantizedDitheredSineSignalGolden;
```

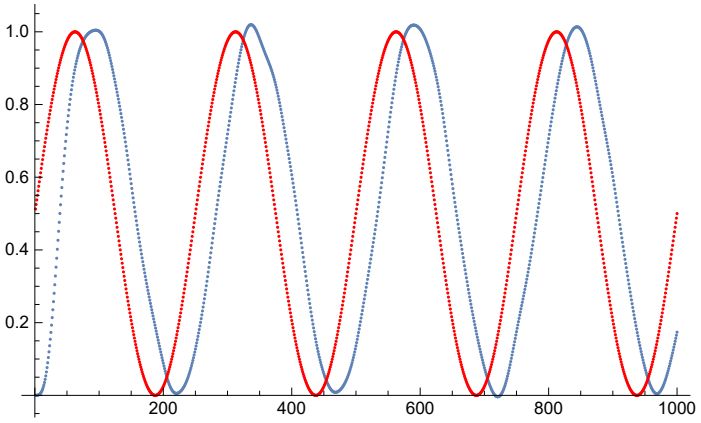
In[64]:=
```
ListAnimate[{MyPeriodogram[sineSignalError, Red, {0, 4}],
  MyPeriodogram[ditheredSineSignalError, Black, {0, 4}],
  MyPeriodogram[ditheredSineSignalGoldenError, Blue, {0, 4}]}]
Export["spectrum_quantization_noise_golden__comparison_sine.gif",
  {MyPeriodogram[sineSignalError, Red, {0, 4}],
   MyPeriodogram[ditheredSineSignalError, Black, {0, 4}], MyPeriodogram[
    ditheredSineSignalGoldenError, Blue, {0, 4}]}, "DisplayDurations" → 2];
```

Out[64]=



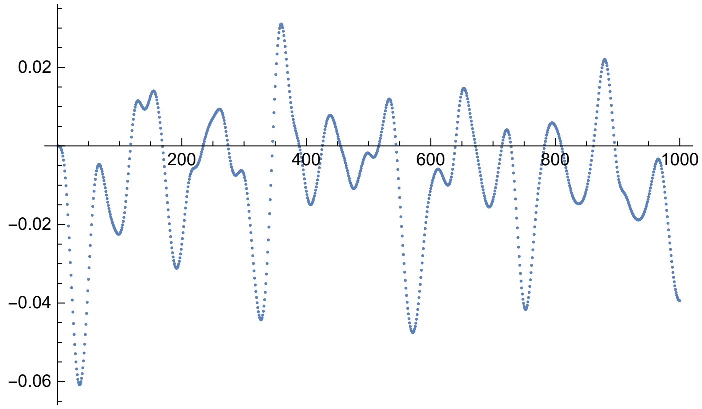
In[66]:=
```
Show[ListPlot[MyFilter[quantizedDitheredSineSignalGolden]],
 ListPlot[originalSineSignal, PlotStyle → Red]]
```

Out[66]=



In[67]:=
```
ListPlot[MyFilter[ditheredSineSignalGoldenError]]
```

Out[67]=



In[68]:=
```
Image[Join[{quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignal}, {quantizedDitheredSineSignal},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden},
  {quantizedDitheredSineSignalGolden}, {quantizedDitheredSineSignalGolden}, 1]]
```

Out[68]=