Tutorial for the FMPlot Package
FM Spectral Plots in Mathematica

Craig Stuart Sapp
craig@ccrma.stanford.edu

23 March 1995

Getting Started

First, load in the FMPlot package so that it can be used in a *Mathematica* notebook. Specify the filename of the package source code to be loaded below:

```
<< "FMPlot.m"
```

To see all of the functions defined in FMPlot:

```
? FMPlot‘*
```

```
AbsPositiveFMSpectrum      PositiveFMSpectrum
CombineSpectra             RemovePhase
Deviation                  SpectralAmp
FMPlot                     SpectralFreq
FMSpectrum                 SpectralLines
Index                      SpectralListQ
NormalizeSpectrum          SpectralPlot
PlotAbsPositiveFMSpectrum
```

To get help on a particular function:

```
? FMPlot
```

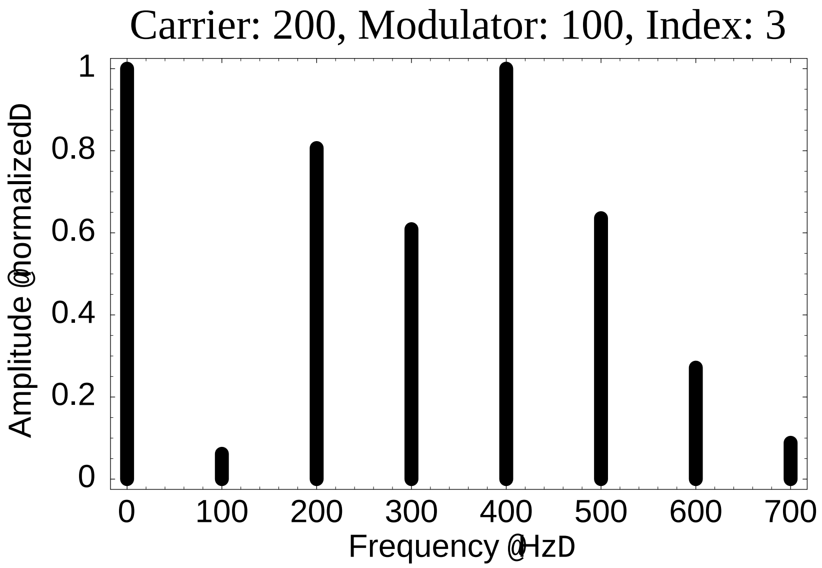
```
FMPlot@carrierFrequency, modulatorFrequency, index, minimumAmplitude:
  0.045, width:0.02, plotOptionsD plots a frequency spectra of the
  given carrier ê modulator ê index; where width controls the
  size of the spectral lines, and lowering the minimumAmplitude
  may extend the range of the plot to include less important sidebands.
```

Double click on the right side of each heading below to open/close a section of this notebook:

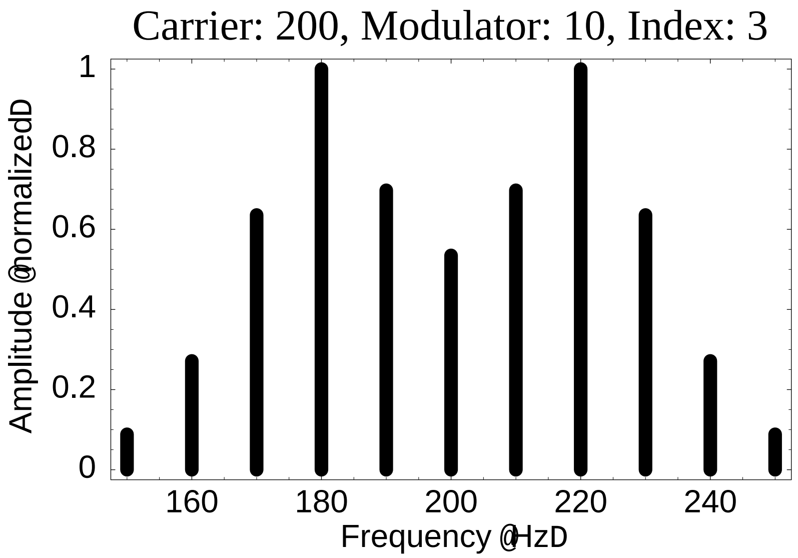
Plotting FM Spectra

The main function for this package is `FMPlot`, which plots the FM spectrum of a given carrier / modulator pair as we hear it. The FMPlot function plots spectral lines and gives a title which lists the three variables of carrier frequency, modulator frequency, and index. All frequencies are positive for this plot since the negative frequencies are folded over and added or subtracted from the positive spectrum as necessary.

```
FMPlot@200, 100, 3D;
```

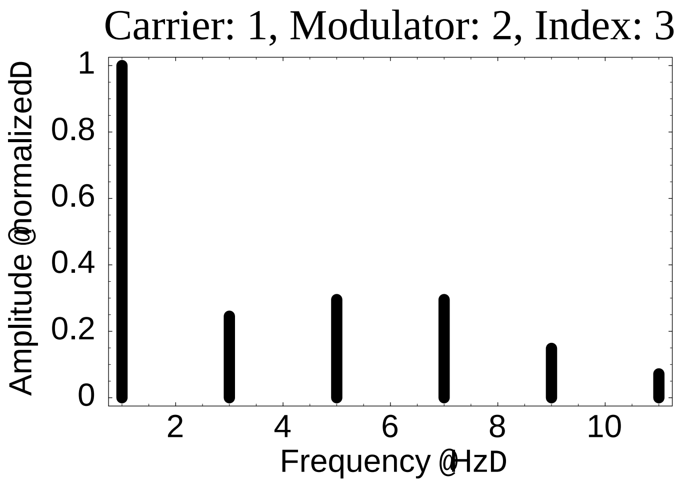


```
FMPlot@200, 10, 3D;
```



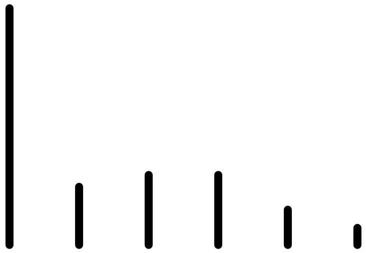
The default size of the plots may be slightly smaller than I would like, depending on the version of *Mathematica* which you are using. Since I use a larger font size than the default, you may have to increase the size of the plot by clicking on the plot and then dragging one of the corners.

```
FMPlot@1, 2, 3D;
```



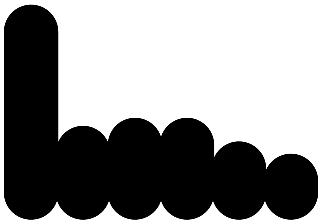
You can also draw just the spectral lines without any text:

```
Show@SpectralLines@AbsPositiveFMSpectrum@1, 2, 3DDD;
```

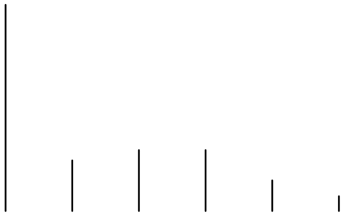


You can change the thickness of the spectral lines as well. The default is 0.02, and the range of values is [0..1]:

```
Show@SpectralLines@AbsPositiveFMSpectrum@1, 2, 3D, 0.2D, PlotRange → AllD;
```

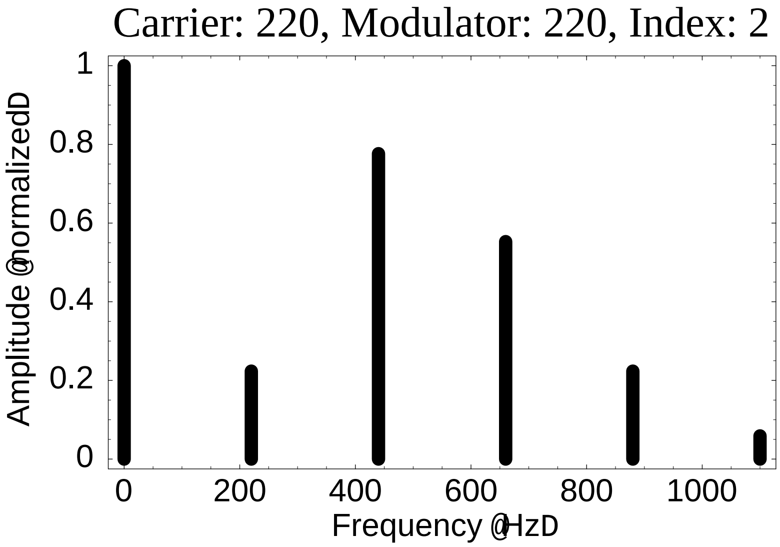


```
Show@SpectralLines@AbsPositiveFMSpectrum@1, 2, 3D, 0.005D, PlotRange → AllD;
```

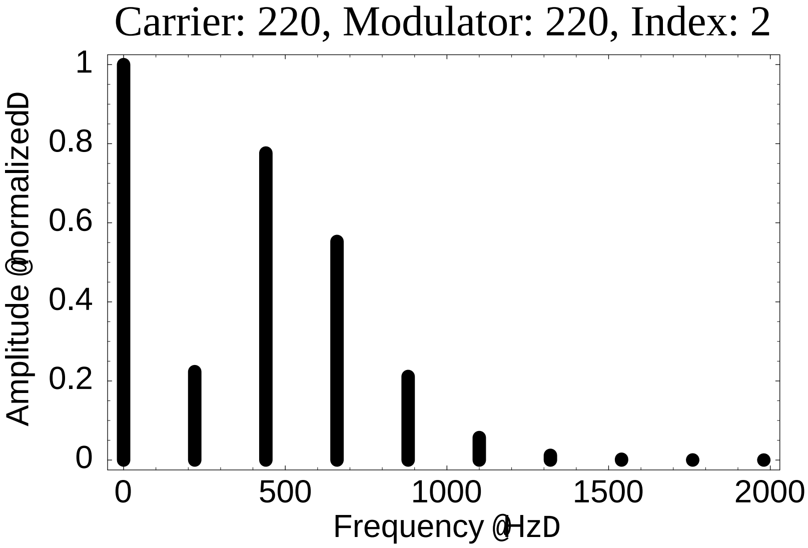


In `FMPlot`, you can specify the lower threshold for plotted amplitude of the spectral lines. The main spectral frequencies extend (carrier ± k * modulator), where k = index + 2. These lines will always be plotted no matter what, but you can extend the outer limits of the plotted spectrum to include additional insignificant harmonics which are present, but in very small quantities. The default minimum amplitude is 0.045, and lowering this parameter may add frequencies that are normally ignored. This amplitude is the value of the Bessel function, which can range from 0 to 1, but the maximum unnormalized value for any carrier/modulator/index amplitude is not always (nearly never) 1.

```
FMPlot@220, 220, 2D;
```

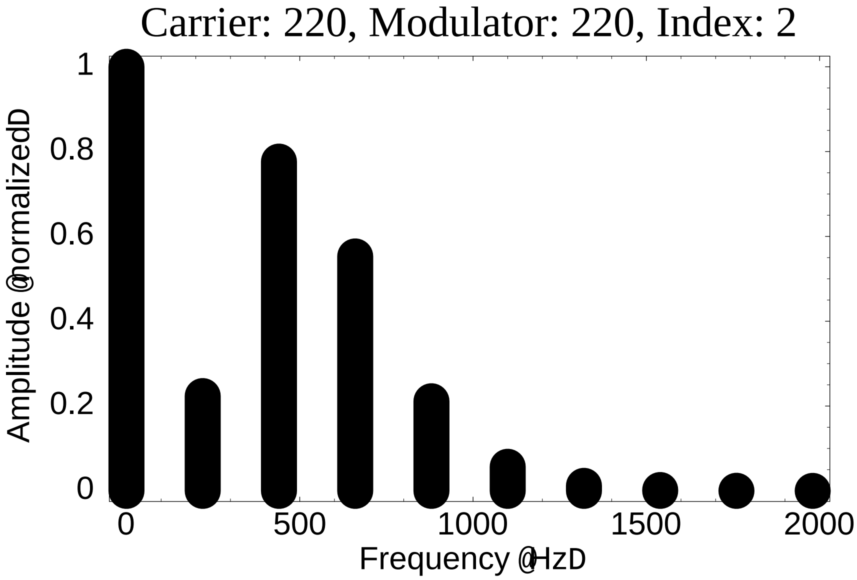


```
FMPlot@220, 220, 2, 0.00001D;
```



Changing the widths of the spectral lines in the plots:

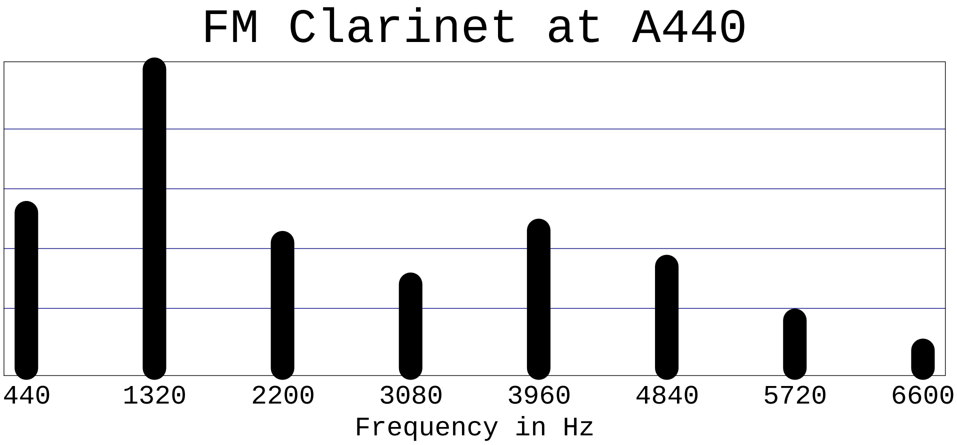
```
FMPlot@220, 220, 2, 0.00001, 0.05D;
```



You can tailor the plot style to your likening by using the same options available with `ListPlot`:

```
Options@ListPlotD
```

```
                  1
9AspectRatio Æ ÅÅÅÅÅÅÅÅÅÅÅ,
               GoldenRatio
 Axes Æ Automatic, AxesLabel Æ None, AxesOrigin Æ Automatic,
 AxesStyle Æ Automatic, Background Æ Automatic, ColorOutput Æ Automatic,
 DefaultColor Æ Automatic, Epilog Æ 8<, Frame Æ False,
 FrameLabel Æ None, FrameStyle Æ Automatic, FrameTicks Æ Automatic,
 GridLines Æ None, ImageSize Æ Automatic, PlotJoined Æ False,
 PlotLabel Æ None, PlotRange Æ Automatic, PlotRegion Æ Automatic,
 PlotStyle Æ Automatic, Prolog Æ 8<, RotateLabel Æ True, Ticks Æ Automatic,
 DefaultFont ¶ $DefaultFont, DisplayFunction ¶ $DisplayFunction,
 FormatType ¶ $FormatType, TextStyle ¶ $TextStyle=
FMPlot@440 3, 440 2, 4, 1, 0.025,
 DefaultFont → 8"Courier", 10<,
 FrameLabel → 8"Frequency in Hz",
   "", FontForm@"FM Clarinet at A440", 8"Courier", 18<D, ""<,
 FrameTicks → 8Table@440 i, 8i, 1, 17, 2<D, None, None, None<,
 GridLines → 8None, Automatic<,
 AspectRatio → 1 ê 3D;
```



Spectral Manipulation

Creating FM Spectra

You can also manipulate FM spectra in the form of listsÅ. The example below shows the spectral amplitude data used to create a previous plot. Spectra are stored in lists, with each element of a list being it self a list of two numbers – first, the frequcency and second, the relative amplitude:

```
TableForm@Reverse@FMSpectrum@220, 220, 2, 0.00001DDD
```

```
1980      0.0000221796
1760      0.000174944
1540      0.00120243
1320      0.00703963
1100      0.0339957
880       0.128943
660       0.352834
440       0.576725
220       0.223891
0         0.576725
-220      0.352834
-440      0.128943
-660      0.0339957
-880      0.00703963
-1100     0.00120243
-1320     0.000174944
-1540     0.0000221796
```

Notice that there are negative frequencies. These frequencies are the mathematical representations of positive frequencies with a phase difference of 180° , or π radians from the phase of the positive frequencies.

```
TableForm@Reverse@PositiveFMSpectrum@220, 220, 2, 0.00001DDD
```

```
1980     0.0000221796
1760     0.000174944
1540     0.00118025
1320     0.00686469
1100     0.0327933
880      0.121904
660      0.318838
440      0.447782
220      -0.128943
0        0.576725
```

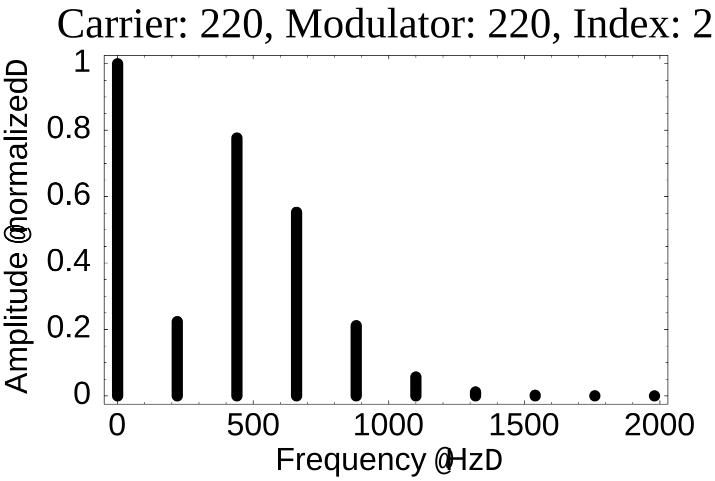
Now all of the frequencies listed in the first column are positive. Note, however, that 220 Hz has a negative amplitude which means that it is 180° out of phase relative to the rest of the harmonics. This phase difference is not audible (in almost all cases) to human ears, so we can look at just the absolute value of the amplitudes:

```
TableForm@Reverse@AbsPositiveFMSpectrum@220, 220, 2, 0.00001DDD
```

```
1980     0.0000221796
1760     0.000174944
1540     0.00118025
1320     0.00686469
1100     0.0327933
880      0.121904
660      0.318838
440      0.447782
220      0.128943
0        0.576725
```

This is the data which is plotted in the plot below; however, notice that the maximum amplitude in this data is 0.576725, and in the plot, the maximum is 1. Since the absolute amplitudes of the spectrum are not important and the relative amplitudes are, the loudest frequency is set to an amplitude of 1 in the plots. This can be done to spectral lists with the function NormalizeSpectrum.

```
FMPlot@220, 220, 2, 0.00001D;
```



```
TableForm@
 Reverse@NormalizeSpectrum@AbsPositiveFMSpectrum@220, 220, 2, 0.00001DDDD
```

```
1980     0.0000384578
1760     0.000303341
1540     0.00204647
1320     0.0119029
1100     0.0568612
880      0.211372
660      0.552843
440      0.776422
220      0.223578
0        1.
```

Or if you prefer to see the relative amplitudes in terms of percentages of the maximum amplitude:

```
spectrum = AbsPositiveFMSpectrum@220, 220, 2, 0.00001D;
TableForm@Reverse@NormalizeSpectrum@spectrum, 100DDD
```

```
1980     0.00384578
1760     0.0303341
1540     0.204647
1320     1.19029
1100     5.68612
880      21.1372
660      55.2843
440      77.6422
220      22.3578
0        100.
```

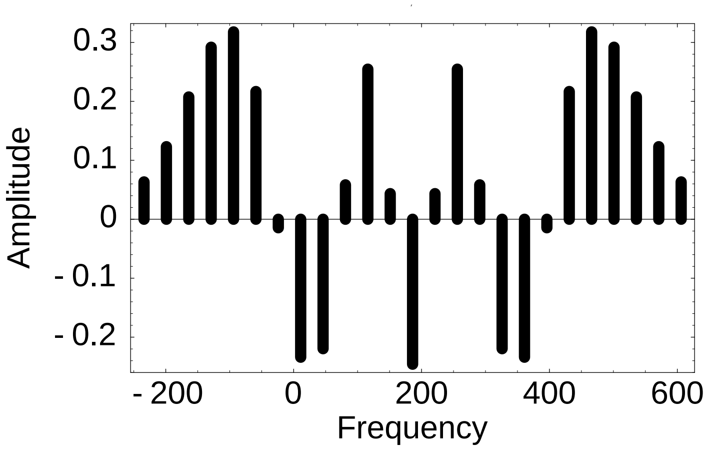
```
spectrum = AbsPositiveFMSpectrum@220, 220, 2, 0.00001D;
8frequencies, amplitudes< =
 Transpose@Reverse@NormalizeSpectrum@spectrum, 100DDD;
amplitudes = Round@amplitudesD;
spectrum = Transpose@8frequencies, amplitudes<D;
TableForm@spectrum, TableAlignments → 8Center, Center<,
 TableHeadings → 8None, 8"Frequency HHzL", "Strength H% of MaxL"<<,
 TableSpacing → 80.5, 4<D
```

```
Frequency HHzL       Strength H% of MaxL
     1980                     0
     1760                     0
     1540                     0
     1320                     1
     1100                     6
      880                    21
      660                    55
      440                    78
      220                    22
       0                     100
```

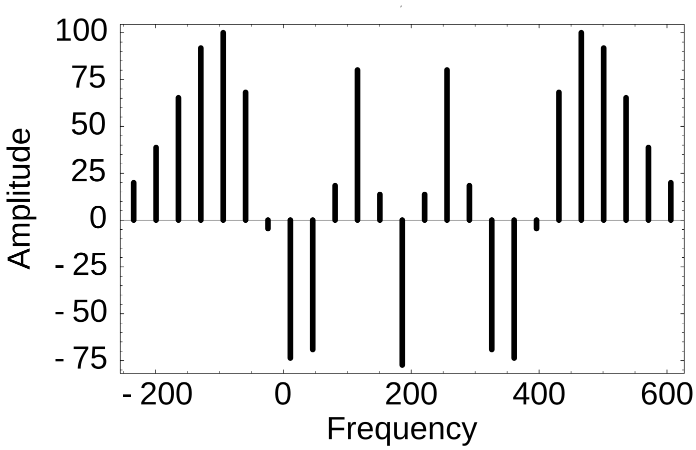
Generalized Spectral Plotting

The function `SpectralPlot` is similar to `FMPlot`, but you can manipulate the spectrum before you plot it:

```
spectrum = FMSpectrum@186, 35, 10D;
SpectralPlot@spectrumD;
```

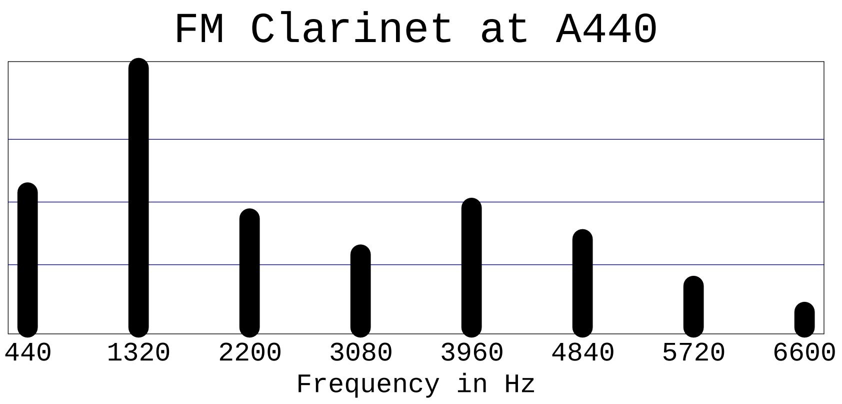


```
SpectralPlot@NormalizeSpectrum@spectrum, 100D, 0.01D;
```



Options can also be used as in `FMPlot`:

```
x = AbsPositiveFMSpectrum@440 3, 440 2, 4, 1D;
SpectralPlotA
 x, 0.025, DefaultFont → 8"Courier", 10<, FrameLabel → 8"Frequency in Hz",
   "", FontForm@"FM Clarinet at A440", 8"Courier-Bold", 16<D, ""<,
 FrameTicks → 8SpectralFreq@xD, None, None, None<,
 GridLines → 8None, Automatic<, AspectRatio → 1/3E;
```



Combining Spectra

CombineSpectra allows you to mix any number of spectra together:

```
x = PositiveFMSpectrum@100, 100, 1D
```

```
880, 0.440051<, 8100, 0.650294<, 8200, 0.420487<, 8300, 0.114903<,
 8400, 0.0195634<<
```

```
y = PositiveFMSpectrum@600, 100, 1D
```

```
88300, 0.0195634<, 8400, 0.114903<, 8500, 0.440051<, 8600, 0.765198<,
 8700, 0.440051<, 8800, 0.114903<, 8900, 0.0195634<<
```

```
z = PositiveFMSpectrum@900, 50, 3D
```

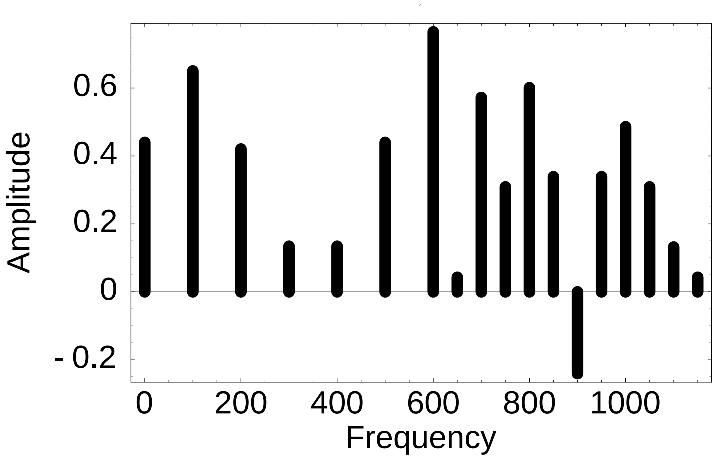
```
88650, 0.0430284<, 8700, 0.132034<, 8750, 0.309063<, 8800, 0.486091<,
 8850, 0.339059<, 8900, -0.260052<, 8950, 0.339059<, 81000, 0.486091<,
 81050, 0.309063<, 81100, 0.132034<, 81150, 0.0430284<<
```

```
CombineSpectra@x, yD
```

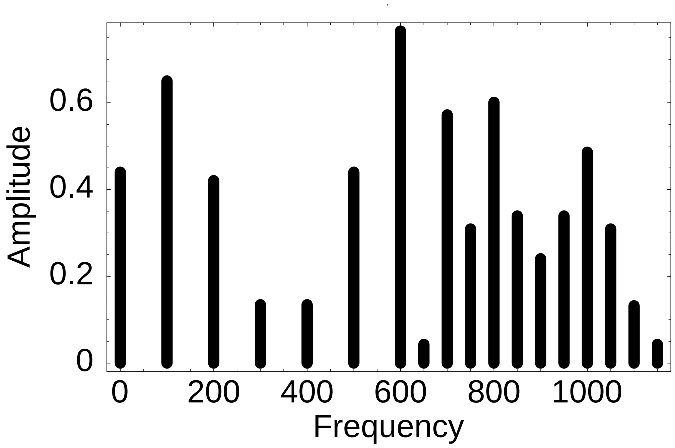
```
880, 0.440051<, 8100, 0.650294<, 8200, 0.420487<, 8300, 0.134467<,
 8400, 0.134467<, 8500, 0.440051<, 8600, 0.765198<, 8700, 0.440051<,
 8800, 0.114903<, 8900, 0.0195634<<
```

Once you have created a combined spectral list which contains phase information, you can remove the phase by using `RemovePhase`:

```
SpectralPlot@CombineSpectra@x, y, zDD;
```



```
SpectralPlot@RemovePhase@CombineSpectra@x, y, zDDD;
```



Note that if you remove the phase information from the spectra before you combine the spectra, you will not be able to accurately combine the spectra.

Also, you can combine duplicate frequency listings in a single spectral list:

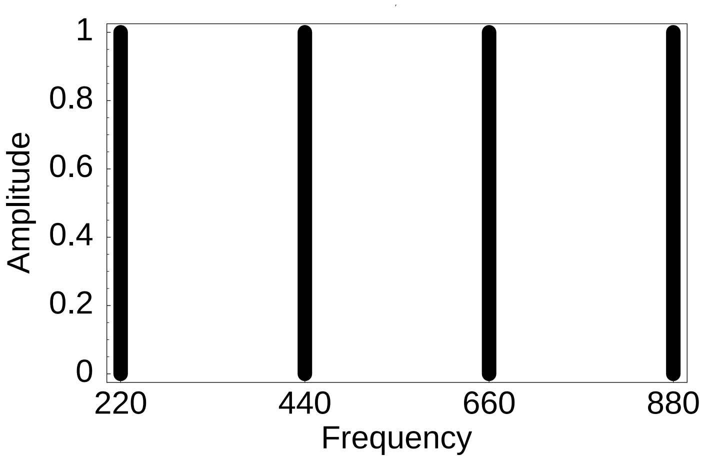
```
CombineSpectra@881, 1<, 83, 3<, 81, -2<<D
```

```
881, -1<, 83, 3<<
```

Additive Synthesis Uses

You can create your own spectra and plot it with FMPlot. A spectral list is a list of lists, where the inner lists contain two numbers. The first of these numbers is the frequency, and the second of these numbers is the amplitude of that frequency.

```
spectrum = 88220, 1<, 8440, 1<, 8660, 1<, 8880, 1<<;
SpectralPlot@spectrum, 0.025,
 FrameTicks → 8SpectralFreq@spectrumD, Automatic, None, None<D;
```



You are able to test your spectrum to see if it is a well-formed spectral list by using `SpectralListQ`:

```
SpectralListQ@spectrumD
```

```
True
```

```
SpectralListQ@80, 1<D
```

```
False
```

```
SpectralListQ@17D
```

```
False
```

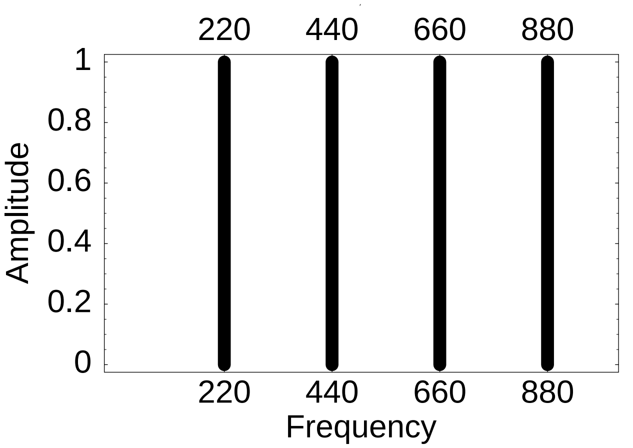
!!! Special Feature of SpectrumPlot !!!

Control of the plotting domain is possible by creating a zero amplitude frequency on the outer ends of the spectral list:

```
newSpectrum = Join@880, 0<<, spectrum, 881000, 0<<D
```

```
880, 0<, 8220, 1<, 8440, 1<, 8660, 1<, 8880, 1<, 81000, 0<<
```

```
SpectralPlot@newSpectrum, 0.025,
 FrameTicks → 8SpectralFreq@spectrumD, Automatic<D;
```

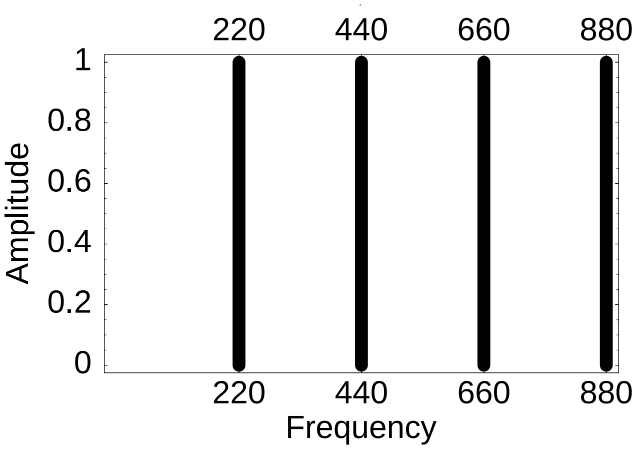


There can be a zero frequency on just one side of the spectral list as well:

```
notherSpectrum = Drop@newSpectrum, -1D
```

```
880, 0<, 8220, 1<, 8440, 1<, 8660, 1<, 8880, 1<<
```

```
SpectralPlot@notherSpectrum, 0.025,
 FrameTicks → 8Transpose@spectrumDP1T, Automatic<D;
```

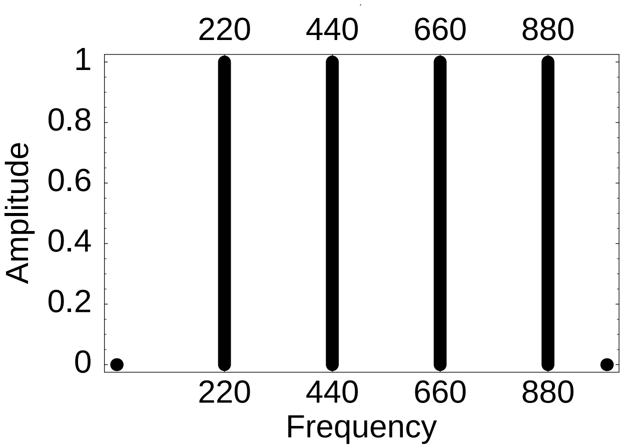


Disable this feature by making sure amplitudes are real, *i.e.*, they have a decimal point after them: 0. or 0.0 .

```
newSpectrum = Join@880, 0.<<, spectrum, 881000, 0.<<D
```

```
880, 0.<, 8220, 1<, 8440, 1<, 8660, 1<, 8880, 1<, 81000, 0.<<
```

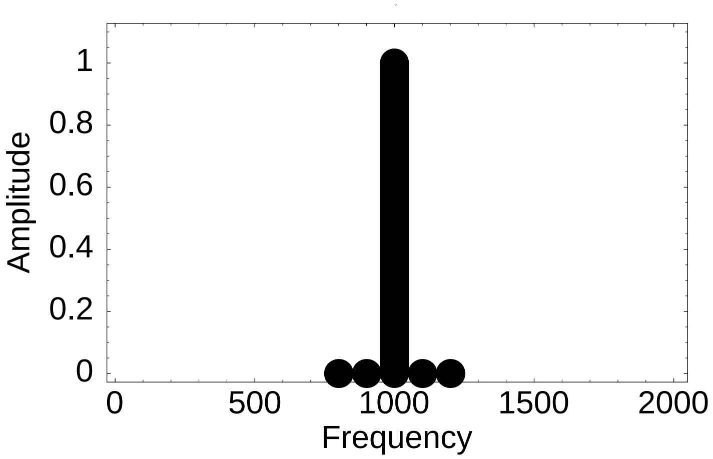
```
SpectralPlot@newSpectrum, 0.025,
 FrameTicks → 8Transpose@spectrumDP1T, Automatic<D;
```



Animations of a dynamic index of modulation are possible using this plot-range feature. Double click on the picture cell below to start the animation of an increasing index of modulation. Once you start the animation, buttons appear at the bottom of the window that allow you to control the speed of the animation and how to loop the animation. Try circular looping. This animation starts at an index of 0 and ends at an index of 6.

```
x = TableAJoinA8820, 0<<, AbsPositiveFMSpectrumA
                                 2 π i    π
    1000, 100, NA3 ij1 + SinA ——————— - ——EyzEE, 882000, 0<, 82000, 1.1<<E,
                   k               36     2  {
  8i, 0, 17<E;
```

```
HSpectralPlot@#1, 0.05D&L ê@ x
```



Description of Functions in the FMPlot Package

A *spectral list* is a list of this form: {{frequency1,amplitude1},...,{frequencyN,AmplitudeN}}

AbsPositiveFMSpectrum

```
? AbsPositiveFMSpectrum
```

```
AbsPositiveSpectrum@carrierFrequency, modulatorFrequency, index,
  minimumAmplitude:0.045D returns the positive frequencies created
  by the particular FM combination with an absolute value for
  the amplitudes.
```

`AbsPositiveFMSpectrum` creates a spectral list of positive frequencies for the specifies carrier/modulator combination with no phase information in the amplitues; so, both the frequencies and amplitudes in the spectral list are positive. Any negative frequencies are first folded over onto the positive frequencies, then the phase of the negative frequency is reversed and added to the amplitude of any existing positive frequency component that

CombineSpectra

```
? CombineSpectra
```

```
CombineSpectra@Spectrum1, Spectrum2, Spectrum3, ...D adds the
  given spectra into one spectal list which is returned.
```

`CombineSpectra` combines different spectra into a single spectral list where there all frequencies are unique. Any number of spectral lists can be passed as an argument to `CombineSpectra`. Also, if there are two frequencies in a single spectral list, `CombineSpectra` will add those two spectral elements together. Note that if you remove the phase information from the spectra before you combine the spectra, you will not be able to accurately combine the spectra.

Deviation

```
? Deviation
```

```
Deviation@modulatorFrequency, indexD returns the deviation given
  the modulatorFrequency and index.
```

The deviation (Δ*f*) is the maximum frequency deviation in Hertz from the carrier frequency (the average frequency). The deviation is related to the amplitude of the modulator frequency – the larger the amplitude of the modulator, the greater the deviation will be.

FMPlot

```
? FMPlot
```

```
FMPlot@carrierFrequency, modulatorFrequency, index, minimumAmplitude:
  0.045, width:0.02, plotOptionsD plots a frequency spectra of the
  given carrier ê modulator ê index; where width controls the
  size of the spectral lines, and lowering the minimumAmplitude
  may extend the range of the plot to include less important sidebands.
```

FMPlot plots the FM spectrum of a given carrier / modulator pair as we hear it. This function plots the spectral lines and gives a title that lists the three variables of carrier frequency, modulator frequency and index. All frequencies are positive for this plot since the negative frequencies are folded over and added or subtracted from the positive spectrum as necessary.

FMSpectrum

```
? FMSpectrum
```

```
FMSpectrum@carrierFrequency, modulatorFrequency, index,
  minimumAmplitude:0.045D returns the partials created by the
  particular FM combination.
```

`FMSpectrum` returns a spectral list of all the sidebands created along with the carrier frequency, as well as their relative amplitudes. Frequencies and amplitudes can be either positive or negative.

Index

```
?Index
```

```
Index@modulatorFrequency, deviationD returns the index given
  the modulatorFrequency and deviation.
```

The index in FM synthesis is the ratio of the deviation to the modulator frequency ($\Delta f / f_m$) . The index must always be greater or equal to zero.

NormalizeSpectrum

```
?NormalizeSpectrum
```

```
NormalizeSpectrum@list, max:1D normalizes the second number in
  a list of number pairs to the value of max which is defaulted to 1.0
```

PlotAbsPositiveFMSpectrum

```
?PlotAbsPositiveFMSpectrum
```

```
PlotAbsPositiveFMSpectrum@carrierFrequency,
  modulatorFrequency,    index, minimumAmplitude:0.045, width:
  0.02, plotOptionsD plots a frequency spectra of the given
  carrier ê modulator ê index; where width controls the size
  of the spectral lines, and lowering the minimumAmplitude may
  extend the range of the plot to include less important sidebands.
```

PlotAbsPositiveFMSpectrum is identical to FMPlot,which plots the FM spectrum of a given carrier / modulator pair as we hear it. This function plots the spectral lines and gives a title that lists the three variables of carrier frequency, modulator frequency and index. All frequencies are positive for this plot since the negative frequencies are folded over and added or subtracted from the positive spectrum as necessary.

PositiveFMSpectrum

```
?PositiveFMSpectrum
```

```
PositiveFMSpectrum@carrierFrequency, modulatorFrequency, index,
  minimumAmplitude:0.045D returns the positive frequencies
  created by the particular FM combination.
```

`PositiveFMSpectrum` is similar to `FMSpectrum`, but any negative frequencies are converted into positive frequencies and combined with any positive spectral frequencies in the spectral list.

RemovePhase

```
?RemovePhase
```

```
RemovePhase@spectralListD removes phase information from the
  spectralList.
```

`RemovePhase` converts a spectral list into another spectral list that contains only real frequencies and amplitudes that are greater or equal to 0. Negative frequencies are assumed to be 180° out of phase from positive frequencies.

SpectralAmp

```
?SpectralAmp
```

```
SpectralAmp@spectralListD returns a list of only the amplitudes
  in the spectral list.
```

`SpectralAmp` returns a list of the amplitudes in a spectral list. Useful if you want to find the maximum or minimum amplitude in a spectral list.

SpectralFreq

```
?SpectralFreq
```

```
SpectralFreq@spectralListD returns a list of only the frequencies
  in the spectral list.
```

`SpectralFreq` returns a list of the frequencies in a spectral list. Useful if you want to label the individual spectral frequencies in a plot.

SpectralLines

```
?SpectralLines
```

```
SpectralLines@spectralList, width:0.02D gives the graphs for a
  set of spectral lines of width thickness.
```

`SpectralLines` returns a Graphics list of the spectral lines and can be mixed with other graphics or plots if desired.

SpectralListQ

```
?SpectralListQ
```

```
SpectralListQ@spectralListD tests whether the argument is a well-
  formed spectral list.
```

`SpectralListQ` returns true if the given object is a list that contains at least one element and each of the elements in the list is a list of two elements. It is assumed that if you can make that much of the spectral list correctly that you will be nice and put only numbers inside the inner lists, since `SpectralListQ` does not check the actual contents of the inner lists.

SpectralPlot

```
?SpectralPlot
```

```
SpectralPlot@spectralList, lineWidth_:0.02D plots a frequency
  spectra of the given list of spectral lines of the form
  88frequency ,amplitude<,...<.
```

`SpectralPlot` plots a spectral list using spectral lines. All frequencies and amplitudes are allowable. Most options valid for `Plot` can also be used. You can also control the plotting domain by creating a zero amplitude frequency on the outter ends of the spectral list: There can also be a zero frequency on just one side of the spectral list: You can disable this feature by making sure amplitudes are real, *i.e., they have a decimal point after them: 0. or 0.0* .