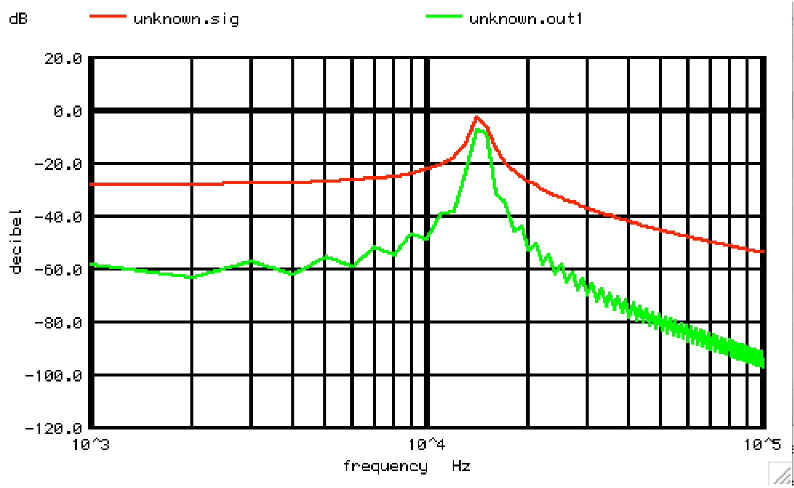
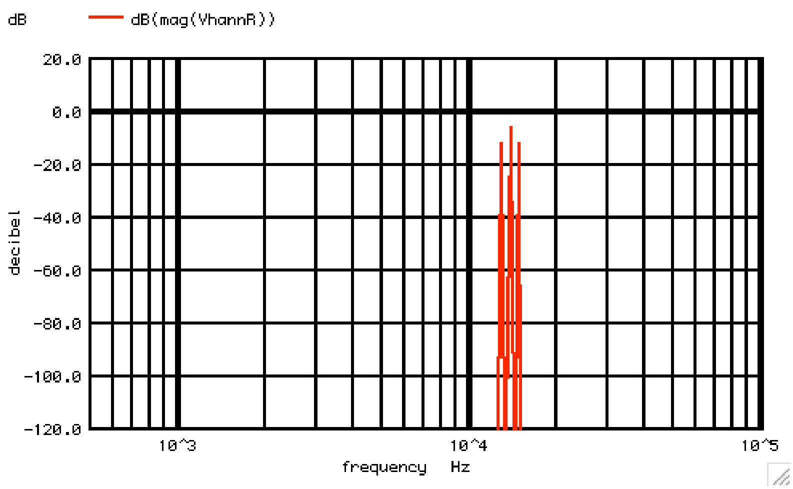
*=============**How_Does_Windowing_Reduce_Leakage?**============



Using just a triangle window can greatly reduce the spectrum leakage of a signal. So how does it do it?

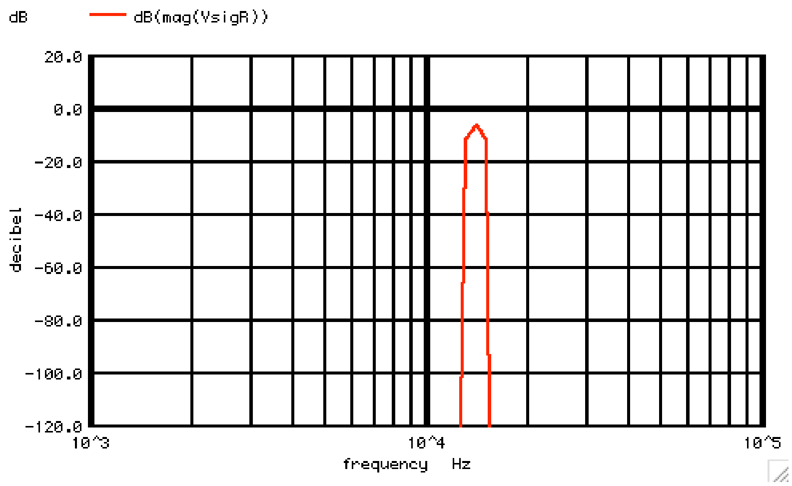
==



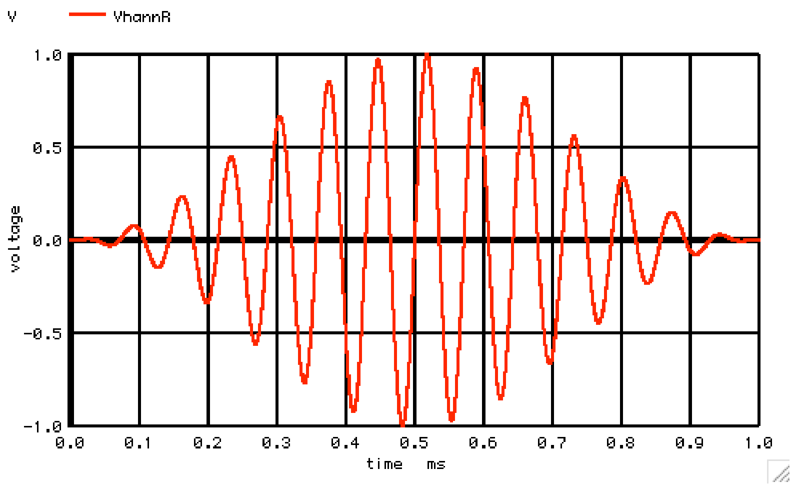
The Hanning window is really just a Classical 100% AM signal. The input signal itself is acting like a carrier. The window function is modulating this carrier. For a simple 14KHz sine-wave, the output is the signal with two -6dB carriers.

But there is another way to look at it.

==



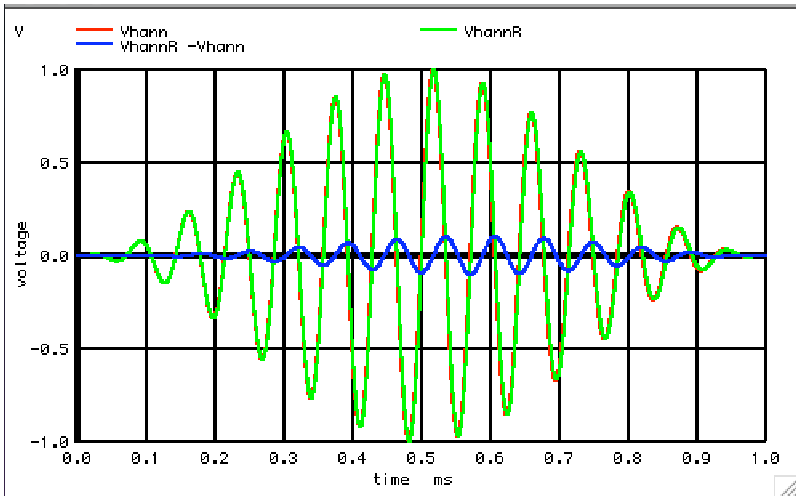
Think of the input signal in this case distorting the window cosine with a 13th harmonic, a 14th harmonic, and a 15th harmonic. In other word the window cosine has been replace with distortion harmonics, and no carrier.



For a 14KHz signal, the AM spectrum, and the harmonic distortion of the window cosine, are really just the same thing.

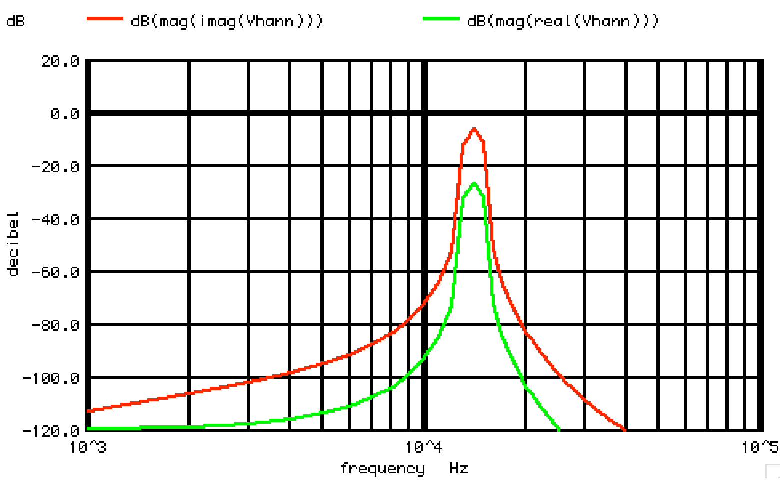
Now watch what happens when the input signal is no longer a perfect 14th kHz signal. Two signals are being generated to see what happens. A reference 14kHz signal will be hanning windowed to generate a reference hanning signal VhannR, which will be compared to a off frequency 14.03Khz signal, to generate the hanning signal Vhann.

```
======================================================================
Vsig      Vsig    0      DC     0      SIN(   0      1      14.03k     )
VsigR     VsigR   0      DC     0      SIN(   0      1      14k        )
VCos1     VCos1   0      DC     0      SIN(   0      1      1k      -.25m )
Bhann     Vhann   0      V =    V(Vsig)*(.5-.5*v(VCos1))
BhannR    VhannR  0      V =    V(VsigR)*(.5-.5*v(VCos1))

.control
set       pensize = 2
tran      .1u       1m      0       .1u
plot      Vhann     VhannR VhannR   -Vhann
set       specwindow=    "rectangular"
spec      1k        100k    1k      v(Vhann)
plot      dB(mag(imag(Vhann)))     dB(mag(real(Vhann)))   vs frequency xlog ylimit -120 20

.endc
.end
```



Since VhannR and Vhann are so close, the difference between them, (the error signal), actually looks like the VhannR signal, only phase shifted and attenuated .

===

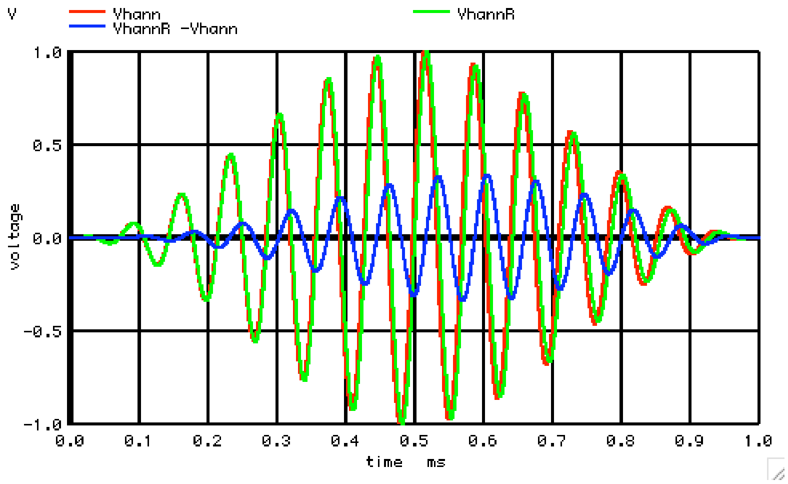


The input 14.03KHz signal is a sine wave. One would expect most of it's spectrum to be imaginary. There is also a little phase shift in this spectrum.

So what happens with a 14.1KHz signal?

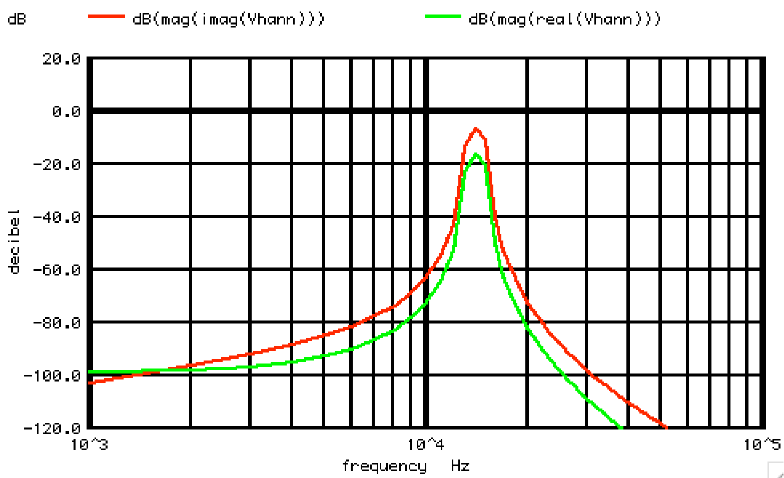
===

```
Vsig      Vsig     0    DC    0    SIN(   0    1    14.1k      )
VsigR     VsigR    0    DC    0    SIN(   0    1    14k       )
VCos1     VCos1    0    DC    0    SIN(   0    1    1k     -.25m )
Bhann     Vhann    0    V =   V(Vsig)*(.5-.5*v(VCos1))
BhannR    VhannR   0    V =   V(VsigR)*(.5-.5*v(VCos1))

.control
set       pensize = 2
tran      .1u       1m      0      .1u
plot      Vhann     VhannR  VhannR  -Vhann
set       specwindow=       "rectangular"
spec      1k        100k    1k     v(Vhann)
plot      dB(mag(imag(Vhann)))     dB(mag(real(Vhann)))  vs frequency xlog ylimit -120 20

.endc
.end
```



The difference between VhannR and Vhann is greater. And still the difference between them, or error signal, actually looks like the VhannR signal, only phase shifted and attenuated. Notice how the error signal on the right is approaching the end of the sample period, without a symmetrical error signal appearing on the left. This could explain the left over Spectrum Leakage.

==

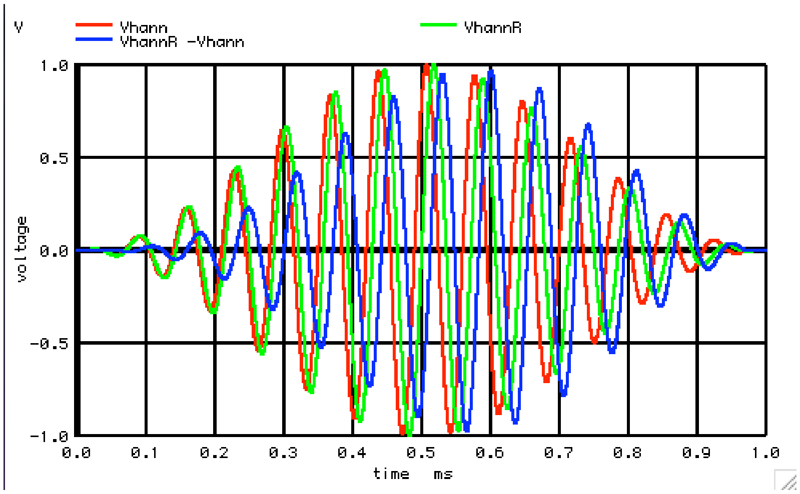


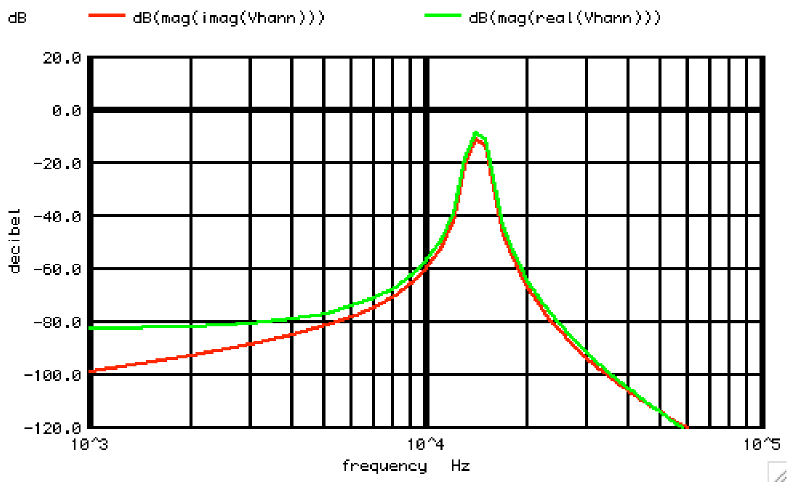
Now there is definitely more phase shift. The 14.1KHz signal is also starting to generate asymmetrical sidebands as well.

So what happens with a 14.3KHz signal?

==

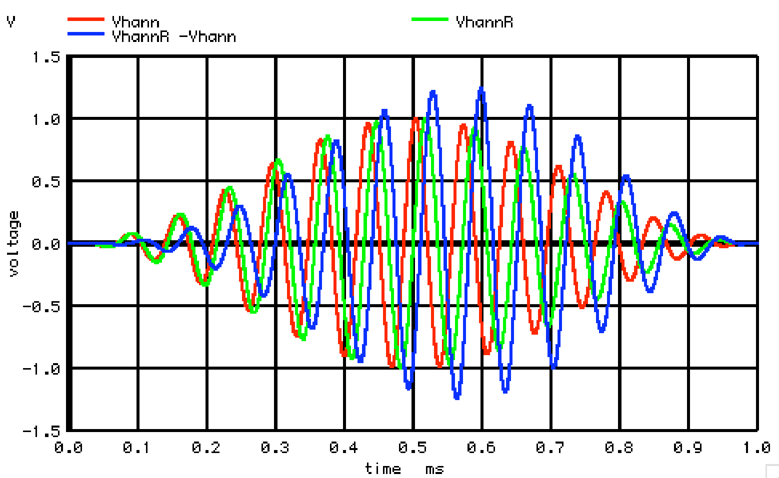
```
Vsig      Vsig     0     DC     0     SIN(   0     1     14.3k      )
VsigR     VsigR    0     DC     0     SIN(   0     1     14k       )
VCos1     VCos1    0     DC     0     SIN(   0     1     1k     -.25m )
Bhann     Vhann    0     V =    V(Vsig)*(.5-.5*v(VCos1))
BhannR    VhannR   0     V =    V(VsigR)*(.5-.5*v(VCos1))

.control
set       pensize = 2
tran      .1u       1m      0      .1u
plot      Vhann     VhannR  VhannR  -Vhann
set       specwindow=       "rectangular"
spec      1k        100k    1k     v(Vhann)
plot      dB(mag(imag(Vhann)))     dB(mag(real(Vhann)))   vs frequency xlog ylimit -120 20

.endc
.end
```

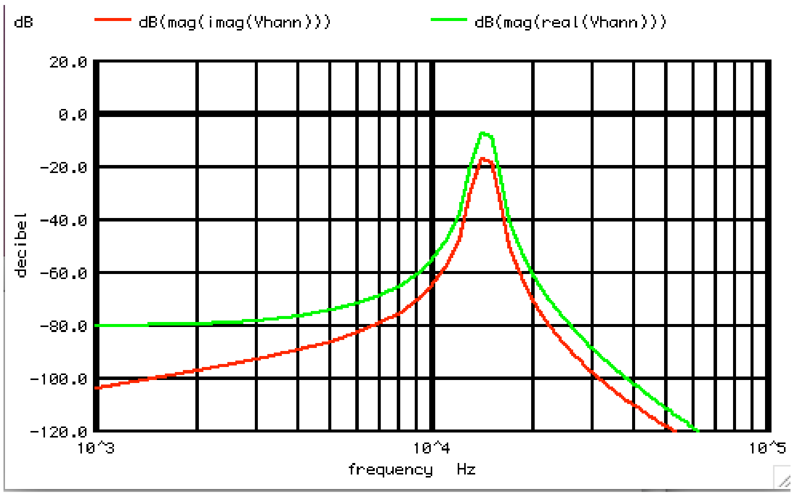




It looks like this is near the 45 degree point. The 13th harmonic is almost gone.

So what happens with a 14.4KHz signal?

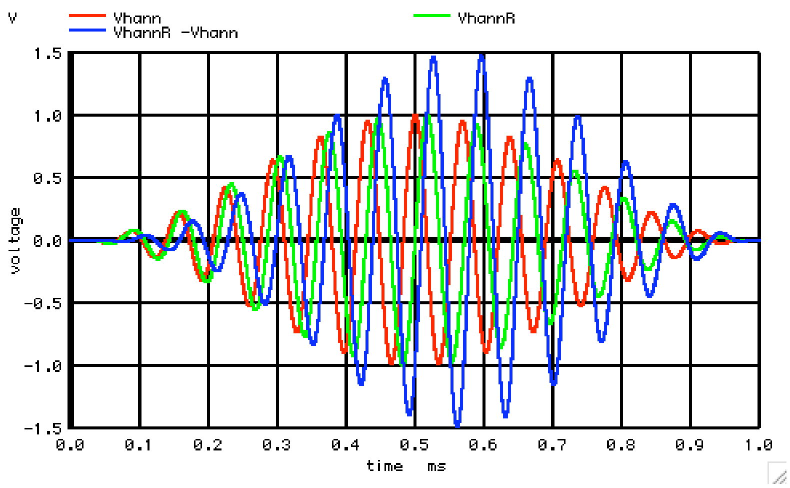


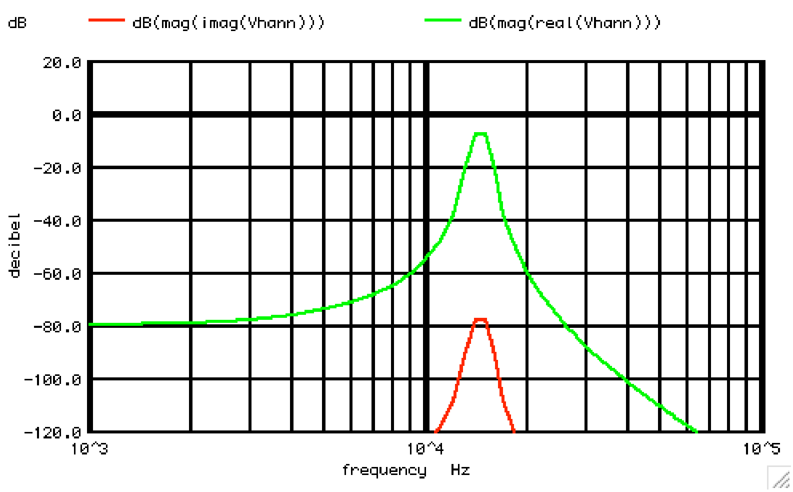


It looks like what was a 14.4KHz sine-wave, is turning into a cosine wave.

So what happens with a 14.499KHz signal?

===





It looks like what was a 14.4999KHz sine-wave has completely turned into cosine elements. Thinking in terms of the distortion harmonics of the 1KHz window cosine, it looks like the 14th and 15th harmonic are at equal magnitudes.

So apparently this hanning window function is translating a non-precise 14KHz+ signal into its distortion harmonics at the 13th, 14th, and 15th levels. From 14.00KHz to 14.50Khz, it is dividing the energy between those three harmonics with the addition of phase shift. And it does the mirror image of that from 14.50KHz to 15KHz, except it is now using the 16th harmonic instead of the 13th.

So that is how it does it. The window function will always be aligned to the sample period. In the case of a Hanning window, the amplitude modulation process can be thought of alternatively as distorting the input window cosine. And the phase and magnitude of those distortion harmonics will adjust accordingly to the input signal frequency.

===========Full_Netlist_For_Copy_Paste=======================

```
Better_Look_At_Hanning
.Option srcsteps = 1 set Gmin = 1.0000E-02
*=======Circuit_Netlist========================
*V_SIN#   NODE_P  NODE_N DC     VALUE  SIN(   V_DC   AC_MAG FREQ   DELAY  FDamp)
Vsig      Vsig    0      DC     0      SIN(   0      1      14.4999k      )
VsigR     VsigR   0      DC     0      SIN(   0      1      14k      )
VCos1     VCos1   0      DC     0      SIN(   0      1      1k     -.25m )
Bhann     Vhann   0      V =    V(Vsig)*(.5-.5*v(VCos1))
BhannR    VhannR  0      V =    V(VsigR)*(.5-.5*v(VCos1))

.control
set       pensize = 2

tran    .1u       1m    0      .1u
plot     Vhann    VhannR  VhannR  -Vhann
set      specwindow=   "rectangular"
spec     1k        100k   1k     v(Vhann)
plot     dB(mag(imag(Vhann)))  dB(mag(real(Vhann)))  vs frequency xlog ylimit -120 20

tran     .1u       1m    0      .1u
set      specwindow=   "hanning"
spec     1k        100k   1k     v(Vsig)
plot     dB(mag(imag(Vsig)))  dB(mag(real(Vsig)))  vs frequency xlog ylimit -120 20

.endc
.end

tran    .1u       1m    0      .1u
set       specwindow=   "rectangular"
spec      1k        100k   1k     v(Vsig)
plot      dB(mag(v(Vsig)))    xlog ylimit -120 20
```

7.29.10_12.02PM
dsauersanjose@aol.com
Don Sauer
http://www.idea2ic.com/