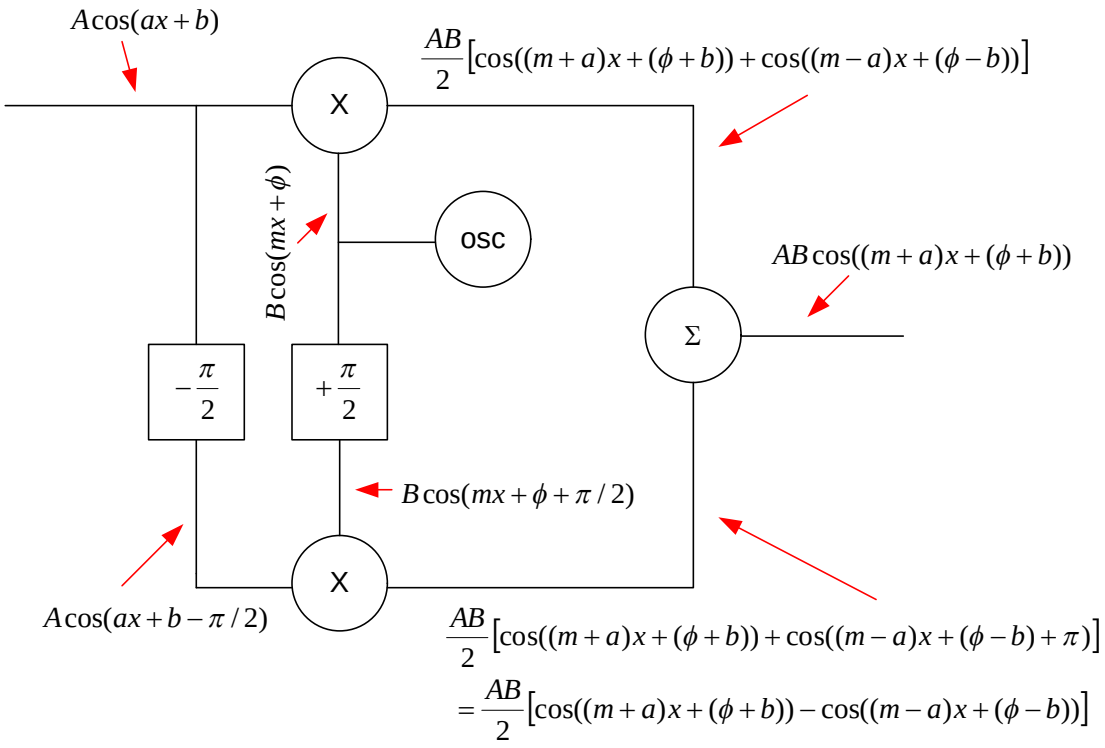
Shifting one frequency up

Say we want to move a tone that we can hear that has a frequency *a* to a frequency of *m+a*. This frequency might be so high that we can't hear it; how might we do this?

The schematic below shows one way of doing this.



Why so complicated? We could have just mixed our frequency of *a* with $B\cos(mx+\phi)$ but doing that creates $\cos((m-a)x+(\phi-b))$ as a term. In RF literature this term is in what is called the lower sideband (LSB). We only want $\cos((m+a)x+(\phi+b))$ and this is in the upper sideband (USB). Basically we want to create a single sideband (ssb) transmitter for just one frequency of *a*.

Cheating with maple

Using maple you can experiment with mixing things.

$$F(f(x))(k)=\frac{\int_{-\infty}^{\infty} f(x)\,e^{I\,k\,x}\,dx}{\pi}$$

While maple can't even calculate the Fourier transform of sine from basic principles it does have a function that will do it called "fourier"

```
int(sin(x)·exp(I·k·x), x = -infinity ..infinity);
```

undefined

With maple (13) performing a Fourier transform of $B\cos(mx+\phi)$ from the time domain to the frequency domain we get the following.

```
with(MTM):
assume(phi > 0): assume(m > 0):
F(B·cos(m·x + phi))(k) = fourier(B·cos(m·x + phi), k)/Pi;
```

$$F(B\cos(m\sim x+\phi\sim))(k)=B\left(e^{I\,\phi\sim}\,\mathrm{Dirac}(k-m\sim)+e^{-I\,\phi\sim}\,\mathrm{Dirac}(k+m\sim)\right)$$

This means $B\cos(mx+\phi)$ has a frequency of m with phase ϕ as well as a frequency of $-m$ with phase $-\phi$, both with amplitude *B*. The positive frequency *m* is probably no surprise but the negative frequency $-m$ might be new to you.

Now we mix two cosines together.

```
assume(phi > 0): assume(m > 0): assume(b > 0): assume(a
    > 0):
F(B·cos(m·x + phi)·A·cos(a·x + b))(k)
    = fourier(B·cos(m·x + phi)·A·cos(a·x + b), k)/Pi;
```

$$F(B\cos(m\sim x+\phi\sim)\,A\cos(a\sim x+b\sim))(k)=\frac{1}{2}\,B\,A\left(e^{-I\,\phi\sim}\,e^{I\,b\sim}\,\mathrm{Dirac}(k+m\sim-a\sim)+e^{I\,\phi\sim}\,e^{-I\,b\sim}\,\mathrm{Dirac}(k-m\sim+a\sim)+e^{-I\,\phi\sim}\,e^{-I\,b\sim}\,\mathrm{Dirac}(k+m\sim+a\sim)+e^{I\,\phi\sim}\,e^{I\,b\sim}\,\mathrm{Dirac}(k-m\sim-a\sim)\right)$$

This time we get four nonzero locations at frequencies $a-m$, $m-a$, $-(m+a)$ and $m+a$, with phases $b-\phi$, $\phi-b$, $-(b+\phi)$ and $b+\phi$ all with amplitudes of $AB/2$.

Usually the negative frequencies are ignored and just the positive frequencies are kept; as *m* is bigger than *a* week get $[AB/2, m+a, \phi+b]$ and $[AB/2, m-a, \phi-b]$ where this time I express amplitude, frequency, and phase as a three dimensional vector.

This means you could use the following notation to describe what has happened.

$$[B,m,\phi]*[A,a,b]=[AB/2,m+a,\phi+b]+[AB/2,m-a,\phi-b]$$

Clearly from this if we advance one phase by 90° and retard the other one by 90° we get the following.

$$[B,m,(\phi+\pi/2)]*[A,a,(b-\pi/2)]$$
$$=[AB/2,m+a,(\phi+\pi/2)+(b-\pi/2)]+[AB/2,m-a,(\phi+\pi/2)-(b-\pi/2)]$$
$$=[AB/2,m+a,\phi+b]+[AB/2,m-a,\phi-b+\pi]$$

How about adding two frequencies that are the same together?

$$F(B\cdot\cos(m\cdot x+\text{phi})\cdot A\cdot\cos(m\cdot x+b))(k)$$
$$=\frac{fourier(B\cdot\cos(m\cdot x+\text{phi})+A\cdot\cos(m\cdot x+b),k)}{\text{Pi}};$$

$$F(B\cos(m\sim x+\phi\sim)\,A\cos(m\sim x+b\sim))(k)=\text{Dirac}(k-m\sim)\,B\,e^{I\phi\sim}+\text{Dirac}(k-m\sim)\,A\,e^{Ib\sim}+\text{Dirac}(k+m\sim)\,B\,e^{-I\phi\sim}+\text{Dirac}(k+m\sim)\,A\,e^{-Ib\sim}$$

In this case we get two terms that appear when k=m so the amplitude and phase are more tricky to work out.

$$\text{polar}(B\,e^{I\phi}+A\,e^{Ib});$$
$$\text{polar}(|B\,e^{I\phi}+A\,e^{Ib}|,\text{argument}(B\,e^{I\phi}+A\,e^{Ib}))$$

So generally we can't say much more than the following two frequencies that are exactly the same.

$$[B,m,\phi]+[A,m,b]=[|Be^{i\phi}+Ae^{ib}|,m,\angle(Be^{i\phi}+Ae^{ib})]$$

However when the amplitudes are also the same we can simplify things.

$$[A,m,\phi]+[A,m,b]=[|Ae^{i\phi}+Ae^{ib}|,m,\angle(Ae^{i\phi}+Ae^{ib})]$$
$$=[A|e^{i\phi}+e^{ib}|,m,\angle(e^{i\phi}+e^{ib})]$$

And if the phases are the same the amplitude doubles

$$[A,m,b]+[A,m,b]=[A|e^{ib}+e^{ib}|,m,\angle(e^{ib}+e^{ib})]$$
$$=[A2|e^{ib}|,m,\angle(2e^{ib})]$$
$$=[2A,m,b]$$

And if the phases differ by 180° the waves are of course going to cancel.

$$[A,m,b]+[A,m,b+\pi]=[A|e^{ib}+e^{ib+i\pi}|,m,\angle(e^{i\phi}+e^{ib+i\pi})]$$
$$=[A|e^{ib}+e^{ib}e^{i\pi}|,m,\angle(e^{i\phi}+e^{ib+i\pi})]$$
$$=[A|e^{ib}-e^{ib}|,m,\angle(e^{i\phi}-e^{ib})]$$

$$= [0, m, 0]$$
$$= \emptyset$$

This means it's critical that as well as the frequency being exactly the same the amplitude has to be exactly the same along with the phase kept steady; a hardware implementation assuring that the amplitude is exactly the same and the phase does not move might be a bit tricky.

So putting it all together we get the following.

$$[B, m, \phi] * [A, a, b] + [B, m, (\phi + \pi/2)] * [A, a, (b - \pi/2)]$$
$$= [AB/2, m + a, \phi + b] + [AB/2, m - a, \phi - b] + [AB/2, m + a, \phi + b] + [AB/2, m - a, \phi - b + \pi]$$
$$= [AB, m + a, \phi + b]$$

This is the same thing as what the schematic says in the very beginning.

Or we could just put the question directly into maple.

$$F\left(B\cdot\cos(m\cdot x + \text{phi})\cdot A\cdot\cos(a\cdot x + b) + B\cdot\cos\left(m\cdot x + \text{phi} + \frac{\text{Pi}}{2}\right)\cdot A\cdot\cos\left(a\cdot x + b - \frac{\text{Pi}}{2}\right)\right)(k) = \frac{1}{\text{Pi}}\left(fourier\left(B\cdot\cos(m\cdot x + \text{phi})\cdot A\cdot\cos(a\cdot x + b) + B\cdot\cos\left(m\cdot x + \text{phi} + \frac{\text{Pi}}{2}\right)\cdot A\cdot\cos\left(a\cdot x + b - \frac{\text{Pi}}{2}\right), k\right)\right);$$

$$F(B\cos(m\sim x + \phi\sim)\, A\cos(a\sim x + b\sim) - B\sin(m\sim x + \phi\sim)\, A\sin(a\sim x + b\sim))(k) = B\,A\,(\text{Dirac}(k - m\sim - a\sim)\, e^{I\,\phi\sim + I\,b\sim} + \text{Dirac}(k + m\sim + a\sim)\, e^{-I\,\phi\sim - I\,b\sim})$$

Giving the same answer as before of $[AB, m + a, \phi + b]$

Extension for many frequencies

At the moment we are only raising one frequency up. What if we wanted to raise up some arbitrary wave that contained many frequencies? To do many frequencies you would have to delay the phase of each frequency you wish to raise by 90°. Lets examine see how we might do this for an arbitrary signal f(t).

First take your arbitrary signal f(t). An arbitrary signal can be written as an infinite series of waves.

$$f(t) = \sum A_n \cos(a_n t + \phi_n)$$

Each of these ways we retard by 90° to create our wanted signal.

$$\hat{f}(t) = \sum A_n \cos(a_n t + \phi_n - \pi/2)$$

We then perform a Fourier transform on this wanted signal.

$$F\left[\hat{f}(t)\right](k) = F\left[\sum A_n \cos(a_n t + \phi_n - \pi/2)\right](k)$$
$$= \sum A_n e^{i(\phi_n - \pi/2)}\delta(k - f_n) + \sum A_n e^{-i(\phi_n - \pi/2)}\delta(k + f_n)$$
$$= -i\sum A_n e^{i\phi_n}\delta(k - f_n) + i\sum A_n e^{-i\phi_n}\delta(k + f_n)$$
$$= -i\,\mathrm{sgn}(k)\sum A_n e^{i\phi_n}\delta(k - f_n) - i\,\mathrm{sgn}(k)\sum A_n e^{-i\phi_n}\delta(k + f_n) \quad \text{as } f_n > 0$$
$$= -i\,\mathrm{sgn}(k)\left[\sum A_n e^{i\phi_n}\delta(k - f_n) + \sum A_n e^{-i\phi_n}\delta(k + f_n)\right]$$
$$= -i\,\mathrm{sgn}(k) \times F[f(t)](k) \quad \text{This is multiplication in the frequency domain}$$
$$= F^{-1}[-i\,\mathrm{sgn}(k)](t) * f(t) \quad \text{This is equivalent to convolution in the time domain}$$

So all we have to do is convolute our arbitrary signal f(t) with the inverse Fourier transform of –isgn(k) to get our wanted signal.

Maple fails to find the inverse fourier transform of sign.

```
ifourier(signum(k), k, t);
```

$$\mathit{invfourier}(\mathrm{signum}(k), k, t)$$

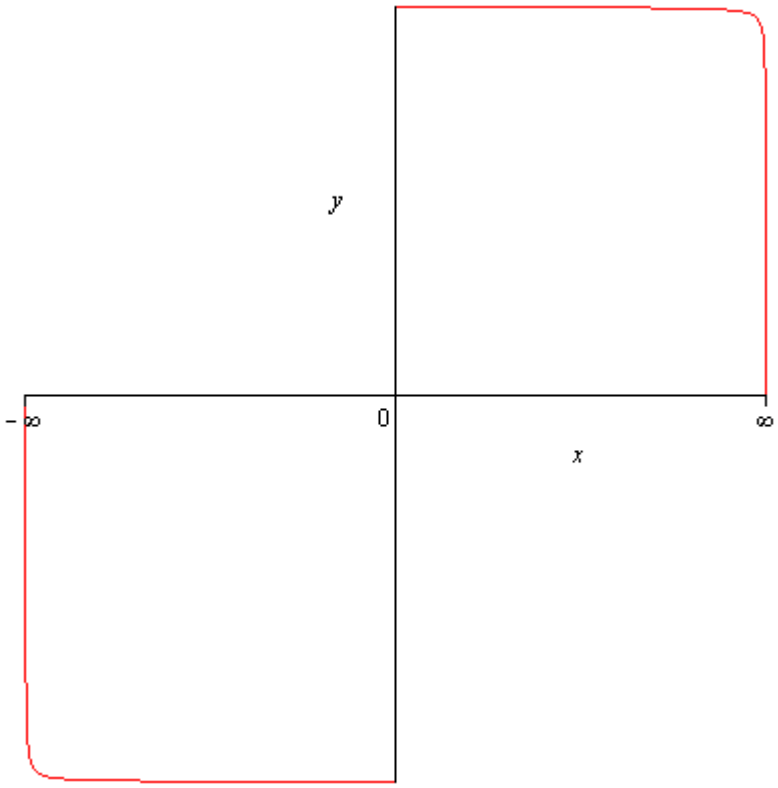
So we go back to basics.

$$F^{-1}[\mathrm{sgn}(k)](t) = \tfrac{1}{2}\int_{-\infty}^{\infty} \mathrm{sgn}(k) e^{ikt}\,dk$$
$$= \tfrac{1}{2}\int_{-\infty}^{\infty} \mathrm{sgn}(k)[\cos(kt) + i\sin(kt)]\,dk$$
$$= \tfrac{1}{2}\int_{-\infty}^{\infty} \mathrm{sgn}(k)\cos(kt)\,dk + i\,\tfrac{1}{2}\int_{-\infty}^{\infty} \mathrm{sgn}(k)\sin(kt)\,dk$$
$$= i\,\tfrac{1}{2}\int_{-\infty}^{\infty} \mathrm{sgn}(k)\sin(kt)\,dk \quad \text{Integral of an odd and even function is zero}$$
$$= i\,\tfrac{1}{2}\left[\int_{-\infty}^{0} \mathrm{sgn}(k)\sin(kt)\,dk + \int_{0}^{\infty} \mathrm{sgn}(k)\sin(kt)\,dk\right]$$
$$= i\,\tfrac{1}{2}\left[\int_{\infty}^{0} \mathrm{sgn}(-k)\sin(-kt)(-dk) + \int_{0}^{\infty} \mathrm{sgn}(k)\sin(kt)\,dk\right]$$
$$= i\,\tfrac{1}{2}\left[\int_{0}^{\infty} \mathrm{sgn}(k)\sin(kt)\,dk + \int_{0}^{\infty} \mathrm{sgn}(k)\sin(kt)\,dk\right]$$
$$= i\int_{0}^{\infty} \mathrm{sgn}(k)\sin(kt)\,dk$$
$$= i\int_{0}^{\infty} \sin(kt)\,dk$$
$$= \left.\frac{-i\cos(kt)}{t}\right|_{0}^{\infty}$$
$$= \left.\frac{-i\cos(kt)}{t}\right|^{\infty} + \frac{i}{t}$$

Problem, can't evaluate as because does not converge at infinity. This is probably the reason Maple can't find the inverse of the sign function; the Fourier transform of the sign function doesn't seem to exist.

We can make a function that looks very similar to the sign function but one that converges to zero at infinity; this will solve the non-convergence problem. The following function is one that will do this.

$$\operatorname{sgn}(k) \approx \begin{cases} e^{-\alpha k} & k > 0 \\ -e^{\alpha k} & k < 0 \end{cases} \quad \textit{for small } \alpha$$



This means we can write the inverse Fourier transform for the sign function as follows.

$$\tfrac{1}{2}\int_{-\infty}^{\infty} \operatorname{sgn}(k) e^{ikt}\,dk \approx i\int_{0}^{\infty} e^{-\alpha k}\sin(kt)\,dk$$

$$= \frac{-ie^{-\alpha k}}{\alpha^2 + t^2}\left(t\cos(kt) + \alpha\sin(kt)\right)\Big|_0^{\infty} \quad \text{maple}$$

$$= \frac{ie^{-\alpha k}}{\alpha^2 + t^2}\left(t\cos(kt) + \alpha\sin(kt)\right)\Big|_0 \quad \text{At infinity exp wins over sine and cosine}$$

$$= \frac{ie^{-\alpha 0}}{\alpha^2 + t^2}\left(t\cos(0t) + \alpha\sin(0t)\right)$$

$$= \frac{it}{\alpha^2 + t^2}$$

$$= \frac{it}{t^2}$$ Take limit of alpha going to zero

$$= \frac{i}{t}$$

Maple can do the Fourier transform of this funnily enough

$$\frac{fourier\left(\frac{\mathrm{I}}{t}\right)}{\mathrm{Pi}}$$

$$-1 + 2\,\mathrm{Heaviside}(w)$$ (this is signum)

So that means the following.

$$F^{-1}\left[-i\,\mathrm{sgn}(k)\right](t) * f(t) = \frac{1}{t} * f(t)$$

And hence

$$\hat{f}(t) = \frac{1}{t} * f(t)$$

This means we just convolute our arbitrary signal by 1/t to get our wanted signal.

$$\hat{f}(t) = \int_{-\infty}^{\infty} \frac{f(t-\tau)}{\tau} d\tau$$

Now we find the discrete version of the above formula.

Define (discrete Fourier transforms) DFT

$$F[y] = \sum_{x=0}^{N-1} f[x] e^{-i 2\pi x y / N}$$ DFT

$$f[x] = \frac{1}{N} \sum_{y=0}^{N-1} F[y] e^{i 2\pi x y / N}$$ Inverse DFT

We start by sampling our signal.

$$f[x] = f(xT) = \sum A_n \cos(a_n x T + \phi_n)$$

Do the DFT on the signal.

$$F[y] = F[f[x]]$$

For the continuous Fourier transform we multiplied the signal by -isgn(k) in the frequency domain ie $-i\,\mathrm{sgn}(k)\times F[f(t)](k)$. Therefore we wish to do the same in the discrete frequency domain, this means the values of y between 1 and N/2-1 inclusive will get multiplied by –i, ones between N/2+1 and N-1 by i (as this is negative frequency), and finally for y=0 and y=N/2 by 0 (we can't decide what the value of sgn(0) is so let's take it be equal to zero).

Therefore we assume N is even then we can say the following.

$$\hat{F}[y]=\begin{cases} -iF[f[x]] & 1\le y<N/2-1 \\ iF[f[x]] & N/2+1\le y<N-1 \\ 0 & y=0 \text{ or } y=N/2 \end{cases}$$

Perform the IDFT on this.

$$F^{-1}[\hat{F}[f[x]]]=\begin{cases} F^{-1}[-iF[f[x]]] & 1\le y<N/2-1 \\ F^{-1}[iF[f[x]]] & N/2+1\le y<N-1 \\ 0 & y=0 \text{ or } y=N/2 \end{cases}$$

This of course is the wanted signal.

$$\hat{f}[x]=\begin{cases} F^{-1}[-iF[f[x]]] & 1\le y<N/2-1 \\ F^{-1}[iF[f[x]]] & N/2+1\le y<N-1 \\ 0 & y=0 \text{ or } y=N/2 \end{cases}$$

This piecewise function can be written as the following.

$$\hat{f}[x]=F^{-1}[(-i\,\mathrm{sgn}(N/2-y)\mathrm{sgn}(y))\times F[f[x]]]$$

And once again as multiplication in the frequency domain is convolution in the time domain we can say the following.

$$\hat{f}[x]=F^{-1}[(-i\,\mathrm{sgn}(N/2-y)\mathrm{sgn}(y))]*f[x]$$

Now we find the inverse of the discrete Fourier transform in the above formula.

$$F^{-1}[(-i\,\mathrm{sgn}(N/2-y)\mathrm{sgn}(y))]=\frac{1}{N}\sum_{y=0}^{N-1}(-i\,\mathrm{sgn}(N/2-y)\mathrm{sgn}(y))e^{i2\pi xy/N}$$
$$=\frac{1}{N}\sum_{y=0}^{N-1}(-i\,\mathrm{sgn}(N/2-y)\mathrm{sgn}(y))(\cos(2\pi xy/N)+i\sin(2\pi xy/N))$$
$$=\frac{1}{N}\sum_{y=0}^{N-1}\mathrm{sgn}(N/2-y)\mathrm{sgn}(y)\sin(2\pi xy/N)-\frac{i}{N}\sum_{y=0}^{N-1}\mathrm{sgn}(N/2-y)\mathrm{sgn}(y)\cos(2\pi xy/N)$$

$$= \frac{1}{N}\sum_{y=0}^{N-1}\operatorname{sgn}(N/2-y)\operatorname{sgn}(y)\sin(2\pi xy/N) \qquad \text{as } \operatorname{sgn}(N/2-y)\operatorname{sgn}(y) \text{ is odd}$$

$$= \frac{1}{N}\sum_{y=1}^{N-1}\operatorname{sgn}(N/2-y)\operatorname{sgn}(y)\sin(2\pi xy/N)$$

$$= \frac{1}{N}\sum_{y=1}^{N/2-1}\operatorname{sgn}(N/2-y)\operatorname{sgn}(y)\sin(2\pi xy/N) + \frac{1}{N}\sum_{y=N/2+1}^{N-1}\operatorname{sgn}(N/2-y)\operatorname{sgn}(y)\sin(2\pi xy/N)$$

$$= \frac{1}{N}\sum_{y=1}^{N/2-1}\sin(2\pi xy/N) - \frac{1}{N}\sum_{y=N/2+1}^{N-1}\sin(2\pi xy/N)$$

$$= \frac{1}{N}\sum_{y=1}^{N/2-1}\sin(2\pi xy/N) - \frac{1}{N}\sum_{y=N/2+1}^{N-1}\sin(2\pi x(v+N/2)/N) \qquad \text{Substituting y=v+N/2}$$

$$= \frac{1}{N}\sum_{y=1}^{N/2-1}\sin(2\pi xy/N) - \frac{1}{N}\sum_{v=1}^{N/2-1}\sin(2\pi x(v+N/2)/N)$$

$$= \frac{1}{N}\left(\sum_{y=1}^{N/2-1}\sin(2\pi xy/N) - \sum_{y=1}^{N/2-1}\sin(2\pi xy/N+\pi x)\right)$$

$$= \frac{1}{N}\left(\sum_{y=1}^{N/2-1}\sin(2\pi xy/N) - \sum_{y=1}^{N/2-1}\left(\sin(2\pi xy/N)\cos(\pi x)+\sin(\pi x)\cos(2\pi xy/N)\right)\right)$$

$$= \frac{1}{N}\left(\sum_{y=1}^{N/2-1}\sin(2\pi xy/N) - \cos(\pi x)\sum_{y=1}^{N/2-1}\sin(2\pi xy/N)\right)$$

$$= \frac{1}{N}\left(1-\cos(\pi x)\right)\sum_{y=1}^{N/2-1}\sin(2\pi xy/N)$$

$$= \frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\sum_{y=1}^{N/2-1}\sin(2\pi xy/N)$$

Now the following equality I was unaware of, I'm not sure what it's called. Maple is quite good with expanding summations of things.

$$Sum(\sin(n\cdot x), y=0..N) = (sum(\sin(n\cdot x), x=0..N));$$

$$\sum_{y=0}^{N}\sin(n\,x) = \frac{1}{2}\,\frac{\sin(n)\,\cos(n\,(N+1))}{\cos(n)-1} - \frac{1}{2}\sin(n\,(N+1)) - \frac{1}{2}\,\frac{\sin(n)}{\cos(n)-1}$$

Wolfram has the same expansion (http://mathworld.wolfram.com/Sine.html) but has written it nicer as

$$\sum_{n=0}^{N}\sin(nx)=\frac{\sin\left(\frac{Nx}{2}\right)\sin\left[\frac{1}{2}(N+1)x\right]}{\sin\left(\frac{1}{2}x\right)}$$

We can use this expansion of the sum of sines to find a closed form.

$$\sum_{y=1}^{N/2-1}\sin\left(\frac{2\pi xy}{N}\right)=\frac{\sin\left(\frac{\pi x}{N}\left(\frac{N}{2}-1\right)\right)\sin\left[\frac{\pi x}{N}\left(\left(\frac{N}{2}-1\right)+1\right)\right]}{\sin\left(\frac{\pi x}{N}\right)}$$

$$=\frac{\sin\left(\frac{\pi x}{N}\left(\frac{N}{2}-1\right)\right)\sin\left(\frac{\pi x}{2}\right)}{\sin\left(\frac{\pi x}{N}\right)}$$

$$\Rightarrow F^{-1}[(-i\,\mathrm{sgn}(N/2-y)\,\mathrm{sgn}(y))]=\frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\frac{\sin\left(\frac{\pi x}{N}\left(\frac{N}{2}-1\right)\right)\sin\left(\frac{\pi x}{2}\right)}{\sin\left(\frac{\pi x}{N}\right)}$$

$$=\frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\frac{\sin\left(\frac{\pi x}{2}-\frac{\pi x}{N}\right)\sin\left(\frac{\pi x}{2}\right)}{\sin\left(\frac{\pi x}{N}\right)}$$

$$=\frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\frac{\left(\sin^2\left(\frac{\pi x}{2}\right)\cos\left(\frac{\pi x}{N}\right)-\sin\left(\frac{\pi x}{N}\right)\cos\left(\frac{\pi x}{2}\right)\sin\left(\frac{\pi x}{2}\right)\right)}{\sin\left(\frac{\pi x}{N}\right)}$$

Now $\cos\left(\frac{\pi x}{2}\right)=0$ for any odd x and $\sin\left(\frac{\pi x}{2}\right)=0$ so

$$\Rightarrow F^{-1}[(-i\,\mathrm{sgn}(N/2-y)\,\mathrm{sgn}(y))]=\frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\frac{\left(\sin^2\left(\frac{\pi x}{2}\right)\cos\left(\frac{\pi x}{N}\right)\right)}{\sin\left(\frac{\pi x}{N}\right)}$$

$$= \frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\sin^2\left(\frac{\pi x}{2}\right)\cot\left(\frac{\pi x}{N}\right)$$

$$= \frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\cot\left(\frac{\pi x}{N}\right)$$

So finally we have a closed form for N being even. (there is also a version when N is odd be we wont go into that here, One solution is good enough)

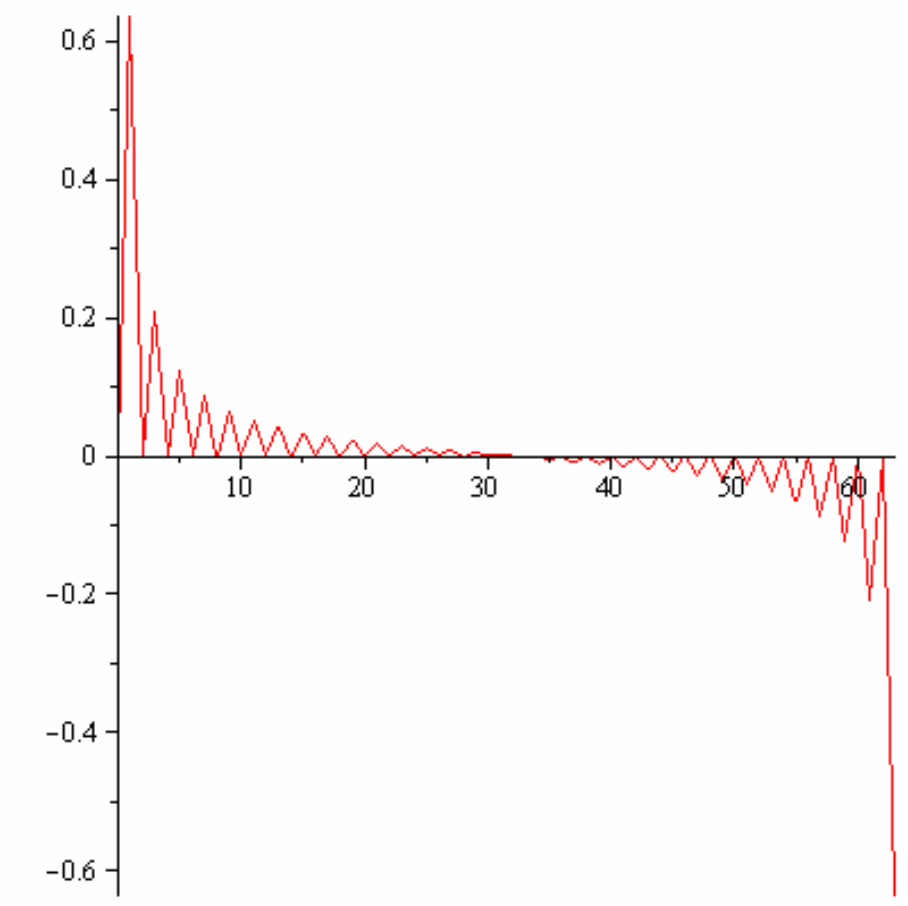
$$F^{-1}[(-i\,\mathrm{sgn}(N/2-y)\,\mathrm{sgn}(y))] = \frac{2}{N}\sin^2\left(\frac{\pi x}{2}\right)\cot\left(\frac{\pi x}{N}\right)$$

Let's first check that the response of this filter does what we want it to do.

```
Digits := 16:
with(DiscreteTransforms):
with(plots):
h := x -> piecewise(x mod N <> 0, 2/N * sin(Pi*x/2)^2 * cot(Pi*x/N))
```

$$x \to \mathit{piecewise}\left(x \bmod N \neq 0, \frac{2\sin\left(\frac{1}{2}\pi x\right)^2\cot\left(\frac{\pi x}{N}\right)}{N}\right)$$

```
N := 64; plot([seq([x, h(x)], x = 0..N - 1)]);
```

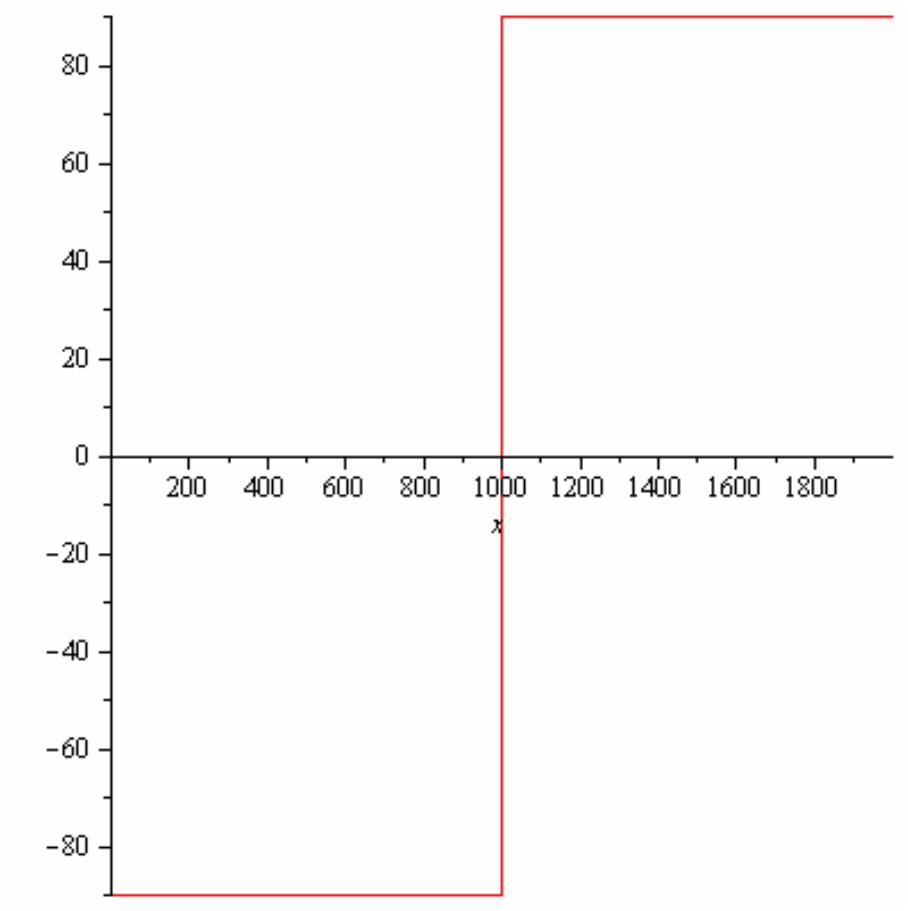


$$Z := Vector\left(\left[seq\left(h(x), \mathrm{x} = 0 .. \frac{\mathrm{N}}{2}\right), seq(0, y = 0 .. 2000 - 1 - N - 1), seq\left(h(x), \mathrm{x} = \frac{\mathrm{N}}{2} + 1 .. N - 1\right)\right]\right):$$

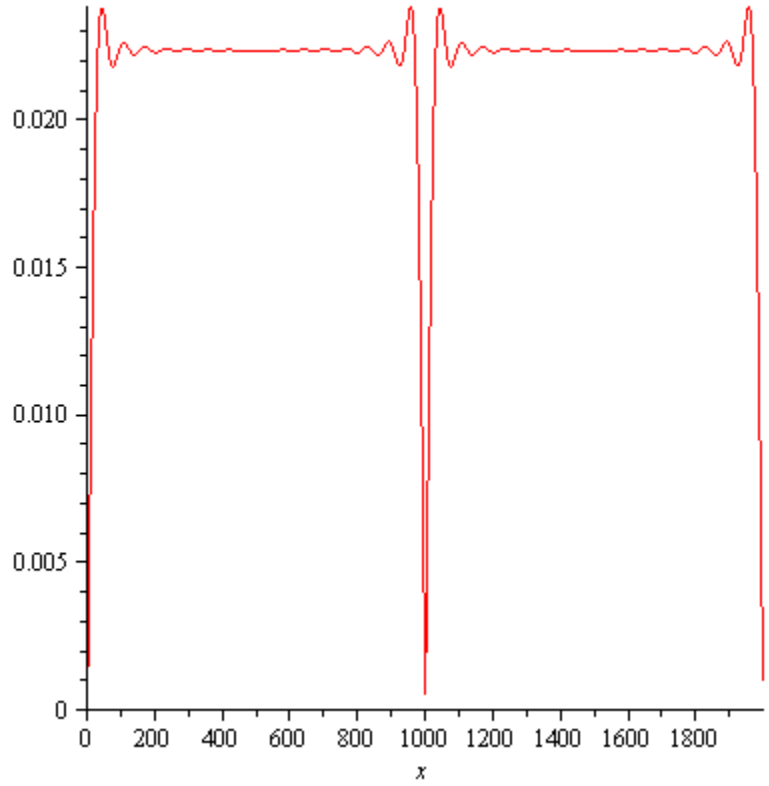
$$X := FourierTransform(Z):$$

$$plot\left(\left\{\left[seq\left(\left[k, \frac{\mathrm{argument}(X[k])}{\mathrm{Pi}} \cdot 180\right], k = 1 .. op(X)[1]\right)\right]\right\}, x = 0 .. op(X)[1] - 1\right);$$

$$plot(\{[seq([k, \mathrm{abs}(X[k])], k = 1 .. op(X)[1])]\}, x = 0 .. op(X)[1] - 1);$$



Phase



Amplitude

So it attenuates the DC signal but all others are shifted by 90° and not attenuated; that's pretty much what we wanted.

We now shift the IDFT result by –N/2 (so as to make the center at N/2 rather than 0) and take account of the x being a multiple of N causing an undefined problem. This results in the filter kernel (impulse response) h[x].

$$h[x]=\begin{cases} 0 & (x-N/2)\bmod 2=1 \\ \frac{2}{N}\cot\left(\pi\left(\frac{x}{N}-\frac{1}{2}\right)\right) & else \end{cases}$$

Formula 1: Hilbert kernel when N is even

With this kernel we can do the following to get the wanted signal. (the -N/2 is to say this filter delays the signal by N/2 samples)

$$\hat{f}[x-N/2]=h*f[x]$$

$$\hat{f}[x-N/2]=\sum_{p=0}^{N-1}h[p]f[x-p]$$

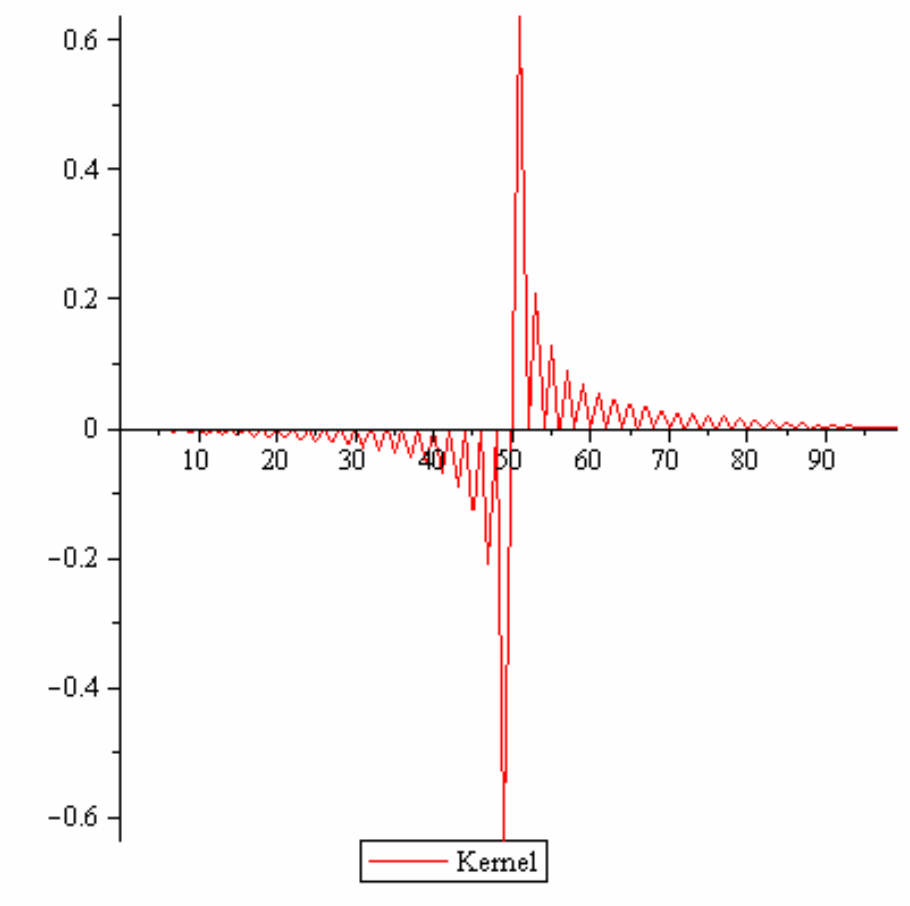
Formula 2: discrete convolution formula

Let's make sure that this filter does what we want to do and check with maple using a sample rate of 48000 and a arbitrary sine wave signal of 1Khz and see if the wave is shifted by 90°.

```
> with(ArrayTools):with(plots):with(DiscreteTransforms):
N := 100:'N'=N;
Digits := 32:
Buffer := convert(Vector[row](N), list):
h := x->piecewise((x - N/2) mod 2 = 1, evalf(2/N*cot(Pi*(x/N
     - 1/2))));
H := ([seq(h(x), x = 0..N - 1)]):
plot([seq([x, Vector([op(H)])(x + 1)], x = 0..N - 1)]);
```

$$N = 100$$

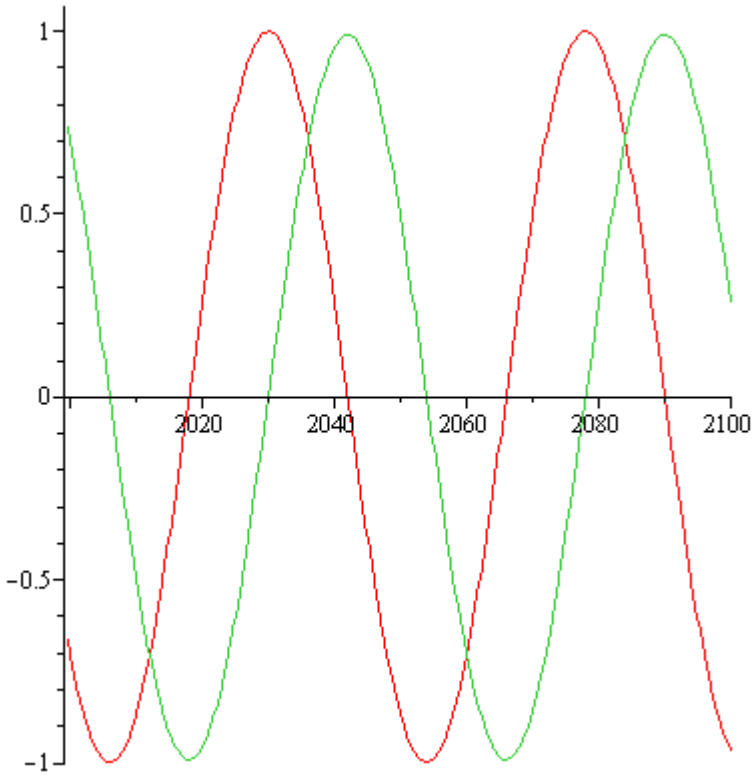
$$x \to piecewise\left(\left(x-\frac{1}{2}N\right) \bmod 2 = 1,\ evalf\left(\frac{2\cot\left(\pi\left(\frac{x}{N}-\frac{1}{2}\right)\right)}{N}\right)\right)$$



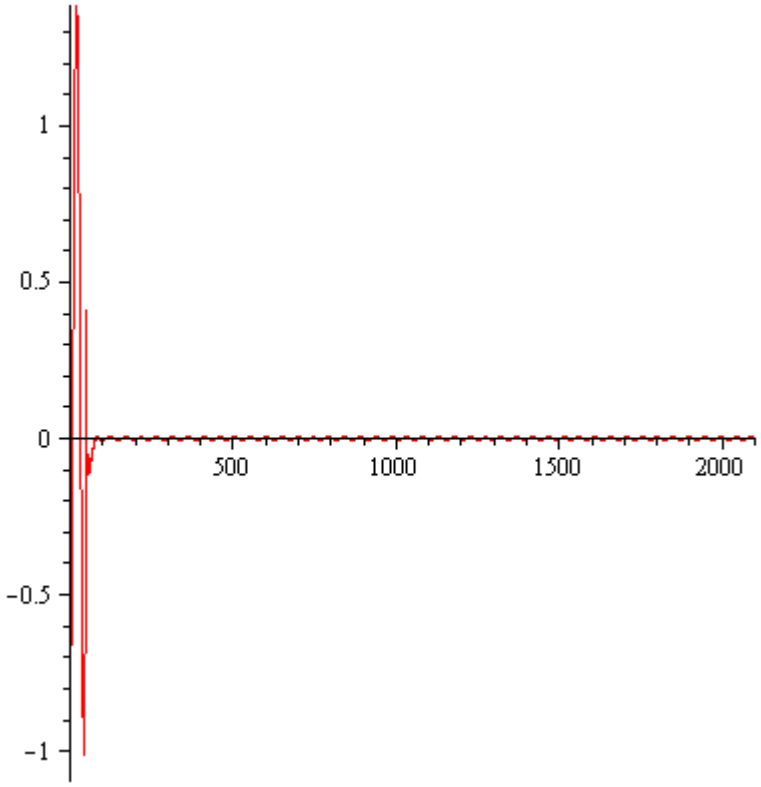

```
circlebuff := proc(f1 :: scalar) :: list
global Buffer, N;
#Buffer := [ op(subsop(1 = NULL, Buffer)), f1 ];
Buffer := [ f1, op(subsop(N = NULL, Buffer)) ];
 end proc:


convol := proc(signal :: scalar) :: scalar
 global Buffer, H, N;
 local p;
circlebuff(signal);
sum('Buffer[p]·H[p]', p = 1..N);
 end proc:
```

$$E := 2100:$$
$$S := 2000:$$
$$F := 1000:$$
$$Y := 'Y': Z := 'Z':$$
$$Z := \left[seq\left(\left[x, convol\left(evalf\left(\sin\left(\frac{x \cdot 2 \cdot \mathrm{Pi}}{48000} \cdot \mathrm{F}\right)\right)\right)\right], x = 0 \,..\, \mathrm{E}\right)\right]:$$
$$Y := \left[seq\left(\left[x,\ evalf\left(\sin\left(\frac{\left(x - \frac{N}{2}\right) \cdot 2 \cdot \mathrm{Pi}}{48000} \cdot \mathrm{F}\right)\right)\right], x = 0 \,..\, \mathrm{E}\right)\right]:$$
$$U := \left[seq\left(\left[x, evalf\left(\sin\left(\frac{\left(x - \frac{N}{2}\right) \cdot 2 \cdot \mathrm{Pi}}{48000} \cdot \mathrm{F} - \frac{\mathrm{Pi}}{2}\right)\right)\right.\right.$$
$$\left.\left. - convol\left(evalf\left(\sin\left(\frac{x \cdot 2 \cdot \mathrm{Pi}}{48000} \cdot \mathrm{F}\right)\right)\right)\right], x = 0 \,..\, \mathrm{E}\right)\right]:$$
$$plot(\{Y, Z\}, x = \mathrm{S} \,..\, \mathrm{E});$$
$$plot(U);$$

Note that a N/2 delay in the original wave is needed because the filter delays any signal by N/2 samples.



90° out.

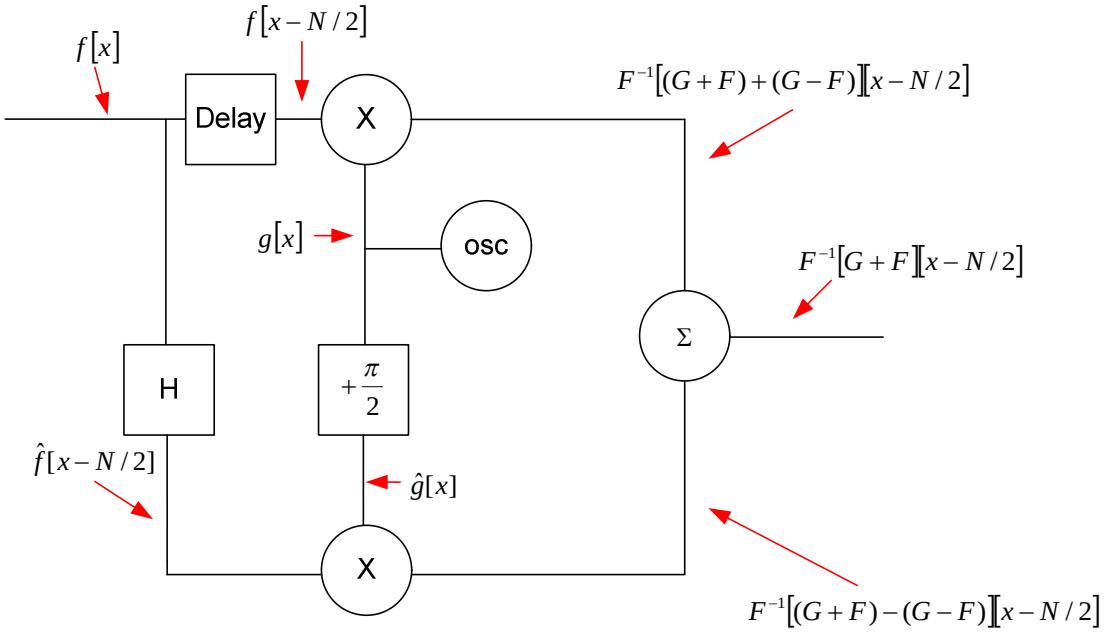


Difference between we wanted and what we got.

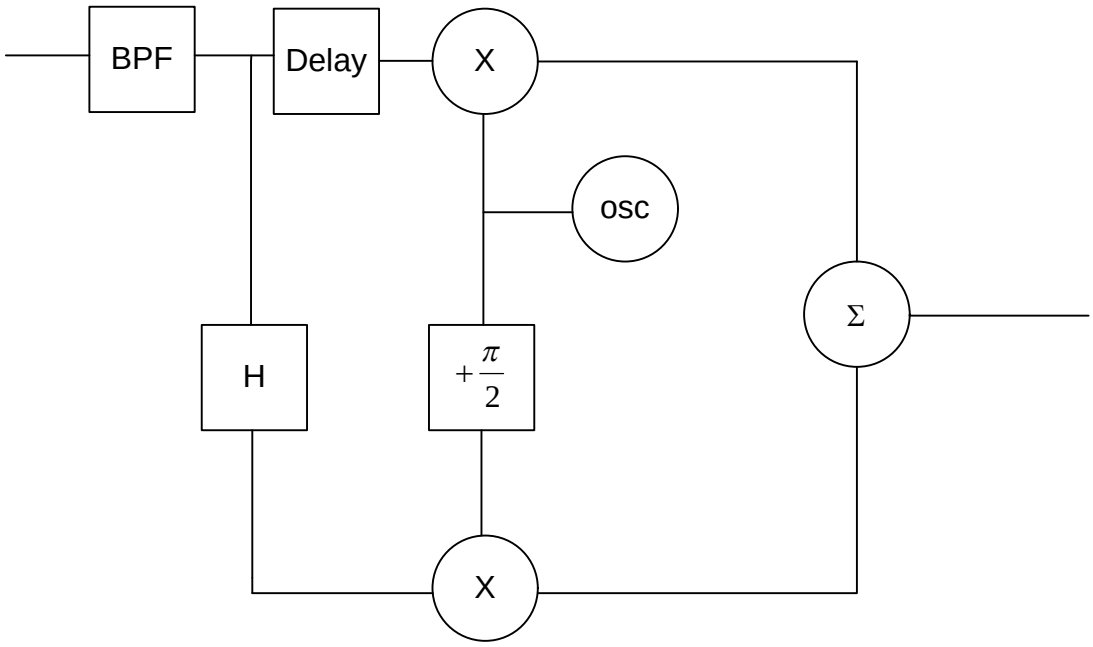
So it does indeed seem to. The amplitudes are the same and the waves are out of sink with each other by 90°.

Putting it altogether

Formula 1 and Formula 2 create what is called a discrete Hilbert transform and denoted by the letter H in the following block diagram. Along with an N/2 delay we can create something that raises all frequencies of an arbitrary wave f[x] up by a frequency G.



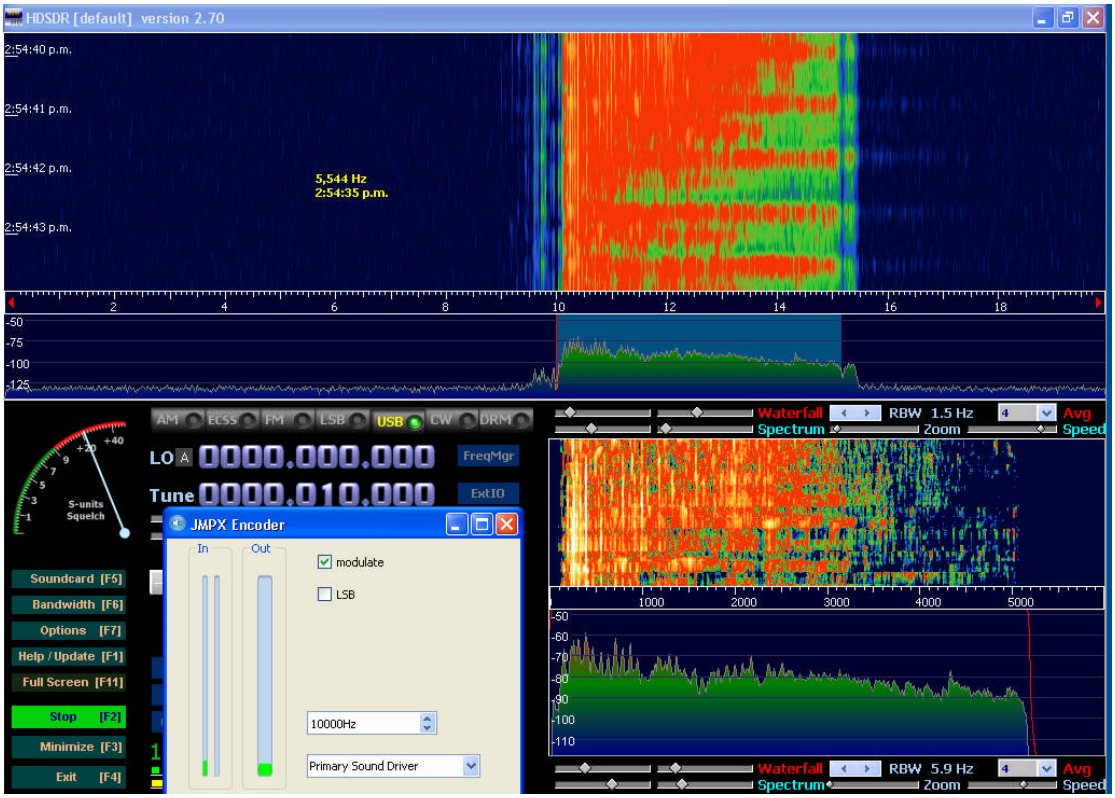
We don't really want to raise every frequency up, frequencies above the hearing range is not as any particular interest anyway. This being the case we also include a band pass filter.



Also if N is divisible by 4 then we can say the following which is even simpler to implement.

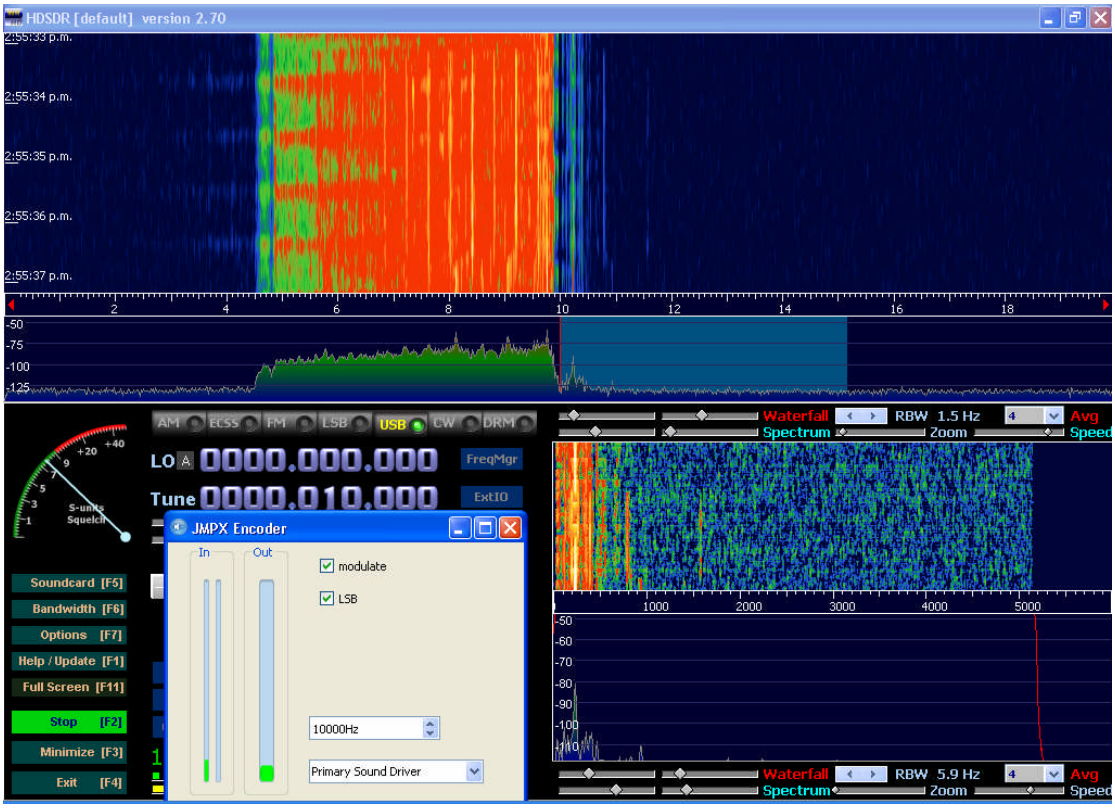
$$h[x] = \begin{cases} 0 & x \bmod 2 = 0 \\ \frac{2}{N}\tan\left(\pi\left(1-\frac{x}{N}\right)\right) & else \end{cases}$$

Implementing these filters as just plain convolution we test it out with the oscillator set to 10 kHz.



The band pass filter is set to only permit frequencies between 300Hz and 5kHz through.

Rather than adding the two final signals together if we subtract the two we end up with the lower side band.

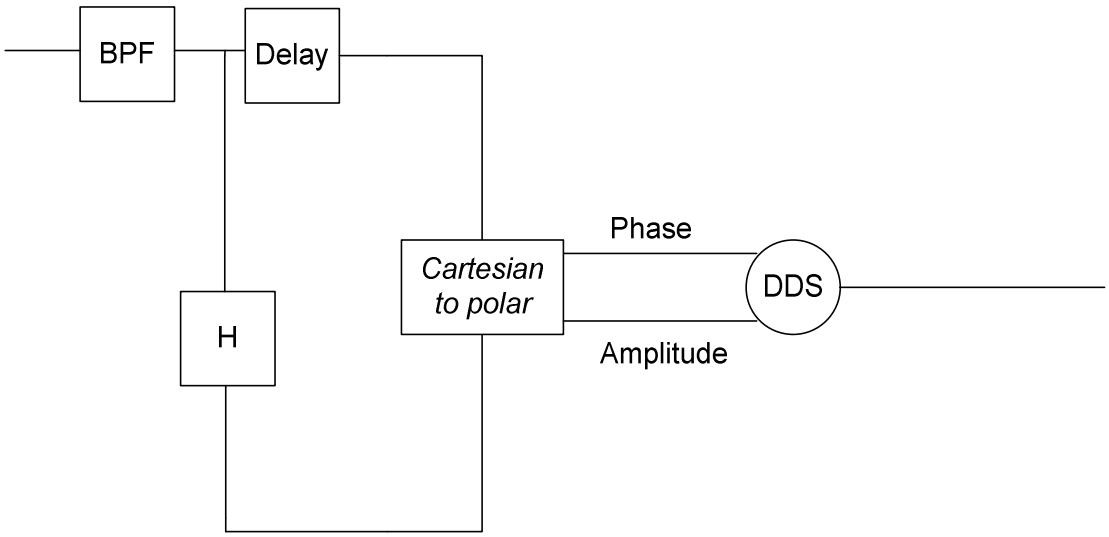


The code was implemented by just hacking the JMPX program. Here http://jontio.zapto.org/hda1/ssbtest.zip is the source code if anyone's interested.

So yes it does work in real life too. The way I've implemented it here is not a particularly good way as regular convolution is very CPU intensive, a better way would be to use FFT convolution this is merely an extension of the current code.

What could be done next?

With a Direct Digital Synthesis (DDS) chip that has the ability to modulate both phase and amplitude you could perform the same idea at RF frequencies. Initially I was thinking of one of those five dollar DDS modules that you can get from ebay at the moment but then I noticed they only have 11° phase resolution and no ability to adjust the amplitude so it's not really suitable. However if there is a super cheap DDS module out there or one is to appear that is suitable (like the AD7008) you could then use a computer to directly modulate it without using a soundcard. The following would do the trick.



Jonti
12/Jan/2014