IIR FM pre-emphasis filter investigations and a digital implementation.

Initial ramblings.

So, like I was, you're wanting to make a compact digital recursive pre-emphasis filter. Well, this is how I got on.

A pre-emphasis filter, is a filter that boosts the high frequencies. An IIR implementation is a infinite impulse response implementation, they are recursive in nature. IIR filters represent what can be made with electronic components. A digital implementation approximates some sort of analog design.

First of all you need to know one or two things. The most important being transfer functions.

A transfer function is a function that relates an input signal with an output signal and is usually denoted with a *H*. If you were to multiply *H* by an input signal you would get what the filter would output. Here's an example, figure 1 shows a voltage divider network. The input voltage and the output voltage are related through the function $V_{out} = V_{in} \frac{R_2}{R_1+R_2}$. This means the transfer function is equal to $H(s) = \frac{R_2}{R_1+R_2}$. The s in the equation is a "complex frequency" $s = \sigma + i\omega$, where ω is angular frequency in radians ($\omega = 2\pi f$) and σ the rate of exponential change (see http://en.wikipedia.org/wiki/S_plane or http://www.dspguide.com/CH32.PDF for more info). As the voltage divider network is not affected by frequency, *s* has no effect in its transfer function in this case.

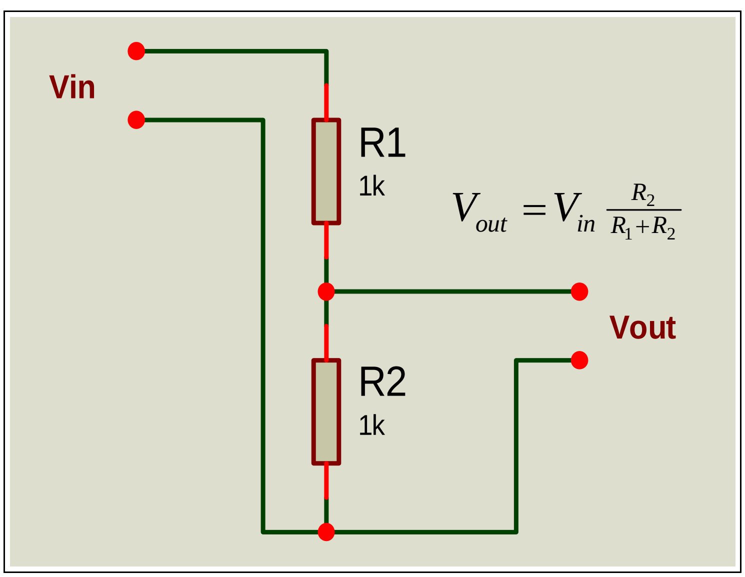


Figure 1: Voltage divider network.

Now what's the big deal with transfer functions? The big deal is you can see how the filter will respond when you change the frequency. If you plot $|H(i2\pi f)|$ you will get a plot of linear gain versus frequency in hertz, whilst, $\arg ument(H(i2\pi f))$ will give you phase response. Of course for our present example $|H(i2\pi f)|$ produces a straight line at $\frac{R_2}{R_1+R_2}$, and $\arg ument(H(i2\pi f))$ is zero. So it's on to more interesting transfer functions.

Now, my goal was to make a pre-emphasis filter for the wideband FM at 88-108 MHz. What is the official standard? In all my looking on the Internet I could not find one place that would give me an official straight answer. All I could find were more like rumors, no official documents. This really grinds my gears. Why can't there be an easily found a website that contains all the official standards for all countries of the world. This is what I found, but as I couldn't find any official documents it is best to take this paragraph with a pinch of salt. It was very common for people to quote the 3 dB corner which rumor has it is 75 µs only for the USA and 50 µs for the rest of the world. A few places would also say that after the 3 dB corner the gain would increase at 6 dB per octave. No one even mentioned a transfer function. It also leaves a lot of questions unanswered; what does the gain do between 0 dB and 3 dB, and if the decibel gain with respect to octaves is not linear at 6 dB per octave everywhere, then how does the gain change when it's not moving at 6 dB per octave? Also, what is the phase response? Unfortunately this is all I had to go on. A simple transfer function and its coefficients given would have been all I needed, and would have told me everything possible about the filter that I was after. Instead this is what I had to do.

Preliminary investigations.

First I needed some simple transfer function to investigate. This would represent some analog electronics. The simplest one I could think of was a linear one.

$$H_a(s) = as + c \qquad (1)$$

Formula 1 is just such a function. From what we know of pre-emphasis, when we put in 0 Hz we would like to get a linear gain of 1. This gain we define as our 0 dB power mark. This being the case implies that $c = 1$.

$$H_a(s) = as + 1 \qquad (2)$$

As we are mainly interested in s where $\sigma = 0$ we can rewrite formula 2 for this special case as

$$H_a(i\omega) = ai\omega + 1 = \sqrt{a^2\omega^2 + 1}\, e^{i\theta} \qquad (3)$$

Where θ is the phase response of the filter. As power is proportional to linear gain square, we can derive the following equations of the power gain response of the filter.

$$P_{lin}(\omega) = \beta\left|H_a(i\omega)\right|^2 = \beta\left(a^2\omega^2 + 1\right) \text{ By formula 3} \quad (4)$$

$$P_{dB}(\omega) = 10\log_{10}\left(\frac{P_{lin}(\omega)}{P_{lin}(0)}\right) \text{ By formula 4}$$

$$\Rightarrow P_{dB}(\omega) = 10\log_{10}\left(a^2\omega^2 + 1\right) \qquad (5)$$

With pre-emphasis we have the time constant τ, which as I said earlier is apparently 75 µs for the US and 50 µs for the rest of the world. As I live in the rest of the world, from now on I'm treating it as being 50 µs. The so-called pre-emphasis time constant is the reciprocal of the angular frequency when the power gain is 3 dB. This means the following.

$$P_{dB}(1/\tau) = 3dB \text{ Definition of time constant.}$$

$$\Rightarrow \frac{a^2}{\tau^2} + 1 = 2 \qquad \text{By formula 5}$$

$$\Rightarrow a = \tau \qquad (6)$$

$$\Rightarrow H_a(s) = \tau\, s + 1 \quad \text{By formula 2} \qquad (7)$$

That solves the unknowns for our transfer function formula 1. However, we haven't looked into the rate of change of the frequency response; remember, some websites have said that the rate of change of the gain would increase at 6 dB per octave after the 3 dB corner. This brings me to another one of my pet peeves, the decibel scale. First we need to create some sort of octave scale, so let $\omega = 2^{oct}$. This does the job where *oct* is the octave. Increase *oct* by one and the frequency doubles i.e. an octave.

$$\frac{d}{d\omega}P_{dB} = \frac{10}{\ln(10)}\frac{\left(2a^2\omega\right)}{\left(a^2\omega^2 + 1\right)} \text{ by formula 5.} \quad (8)$$

$$\Rightarrow \frac{d}{doct}P_{dB} = \left(\frac{d}{d\omega}P_{dB}\right)\frac{d\omega}{doct}$$

$$= \frac{20}{\ln(10)}\frac{\left(a^2\omega\right)}{\left(a^2\omega^2 + 1\right)}\frac{d}{doct}2^{oct} \text{ by formula 8.}$$

$$= \frac{20}{\ln(10)}\frac{\left(a^2\omega\right)}{\left(a^2\omega^2 + 1\right)}\ln(2)2^{oct}$$

$$= \frac{20\ln(2)}{\ln(10)}\frac{\left(a^2\omega^2\right)}{\left(a^2\omega^2 + 1\right)} \quad (9)$$

Clearly formula 9 is not linear with respect to frequency. But when $\omega >> 1/a \Rightarrow a\omega >> 1$, we can make an approximation which is the following.

$$\frac{d}{doct}P_{dB} \approx \frac{20\ln(2)}{\ln(10)}\frac{\left(a^2\omega^2\right)}{\left(a^2\omega^2\right)} \text{ When } \omega >> 1/a \text{ by formula 9.}$$

$$\Rightarrow \frac{d}{doct}P_{dB} \approx \frac{20\ln(2)}{\ln(10)} \quad \text{When } \omega >> 1/a$$

$$\Rightarrow \frac{d}{doct}P_{dB} \approx 6.02\ dB/octave \quad \text{When } \omega >> 1/a \qquad (10)$$

We can also see formula 9 is a monotonic increasing function. This means that this $6.02\ dB/octave$ is the biggest the decibels per octave power gain can have and it happens at a frequency of infinity. This I presume is where the $6\,dB/octave$ comes from that some websites mention. Now our transfer function pretty much matches what websites mention about FM pre-emphasis filters (really this is not hard because they don't tell us much).

Now let's graph the frequency response and phase response of this transfer function.

```
> tau:=50*10^(-6);
> PdB:=omega->10*log10(tau^2*omega^2+1);
>plot(PdB(2*Pi*f),f=0..17000,labels=[Frequency_Hz,Power_gain_dB],labeldirections=[HORIZONTAL,VERTICAL],title=Power_gain_verses_freque
ncy);
> arg:=omega->argument(tau*(I*omega)+1);
> plot((180/
Pi)*arg(2*Pi*f),f=0..17000,labels=[Frequency_Hz,Phase_degrees],labeldirections=[HORIZONTAL,VERTICAL],title=Phase_verses_frequency);
```

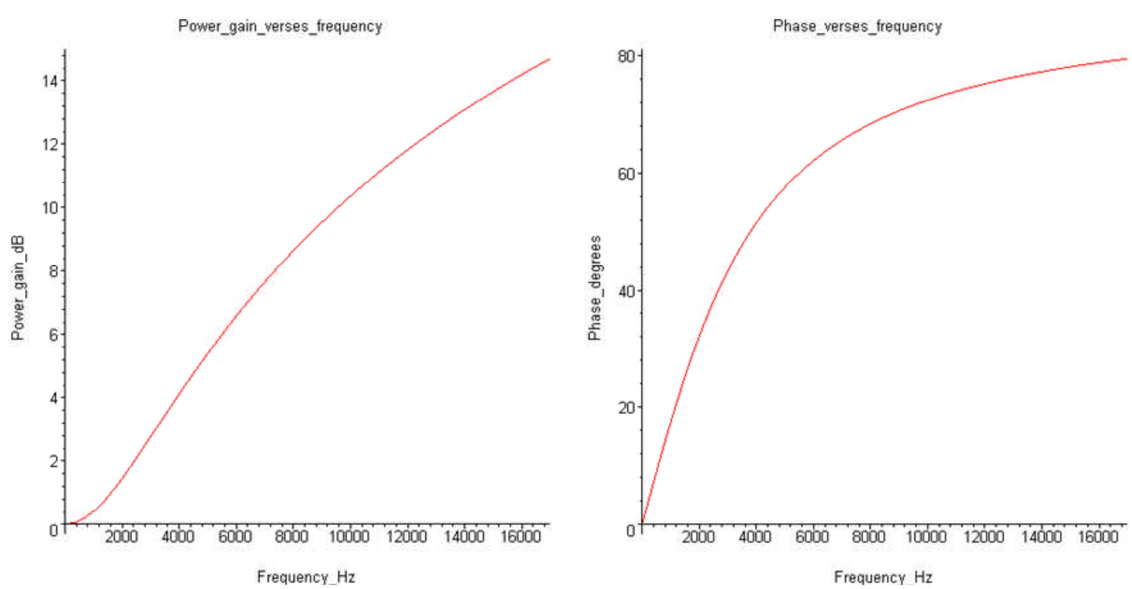


Figure 2: Gain and phase response of formula 7.

So what does the electronic network that represents this transfer function look like? The answer is figure 3.

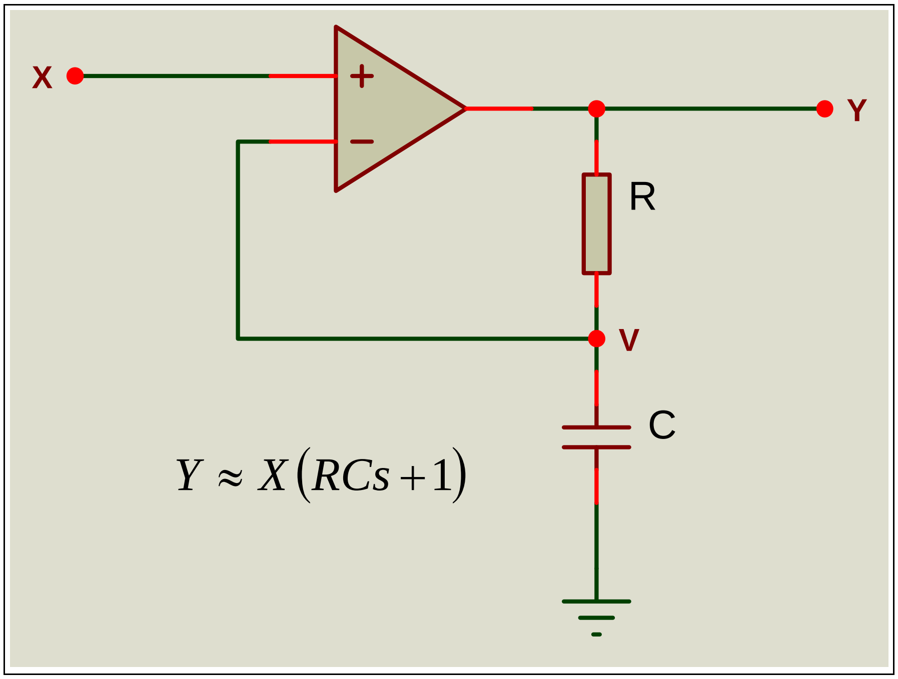


Figure 3: Schematic representation of formula 2 and 7.

To see that figure 3 is related to formula 2, let X, Y and V be rms voltages at their respective positions. Calculate the transfer function of each component, the input being the current through the component and the output being voltage across the component. For the resistor R this is simply the resistor value as $R = V_R / I_R$. For the capacitor C we need to calculate the Laplace transformation of the voltage over the capacitor and divide this by the Laplace transformation of the current through the capacitor. The differential equation that relates current and voltage with the capacitor is $i(t) = C \frac{d}{dt} v(t)$. Using the differentiation property of the Laplace transformation this implies that $I_C(s) = CsV_C(s)$, and hence $H_C(s) = 1/sC$. Once again for more details see http://www.dspguide.com/CH32.PDF.

Then, we can calculate how the voltage Y is related to voltage X through the following.

$Y = G(X - V)$ What an op-amp does. (11)

$V = Y \frac{1/sC}{R + 1/sC}$ R C voltage divider network.

$\Rightarrow V = Y \frac{1}{RCs+1}$ (12)

$Y = G\left(X - Y \frac{1}{RCs+1} \right)$ By formula 11 and 12.

$$\Rightarrow \frac{Y}{G} + Y \frac{1}{RCs+1} = X$$

$$\Rightarrow Y\left(\frac{1}{G}+\frac{1}{RCs+1}\right)=X$$

$$\Rightarrow Y\left(\frac{1}{RCs+1}\right)\approx X$$ Assuming that the gain of the op amp is very large.

$$\Rightarrow \frac{Y}{X}\approx RCs+1$$

$$\Rightarrow H_a(s)\approx RCs+1 \qquad (13)$$

This is just formula 7 with $\tau = RC$.

The fly in the ointment.

There is a problem with the transfer function we have derived. The problem is as the frequency increases so does the gain, and with this transfer function this never stops. This means at an infinite frequency we have an infinite gain. This can be solved but first we need to have a look around the s-plane.

So far we have not looked into anything more than the points lying on the imaginary axis of the s-plane. Now let's look over this s-plane more. Figure 4 shows a contour plot of the magnitude of the transfer function we have derived so far. Think of color as height, purples and blues are low in height while yellows and reds a high in height. The point at $(-1/\tau, 0)$ (purple in color) has a magnitude of zero and is called not surprisingly a "zero". The vertical black line is the frequency response of the filter.

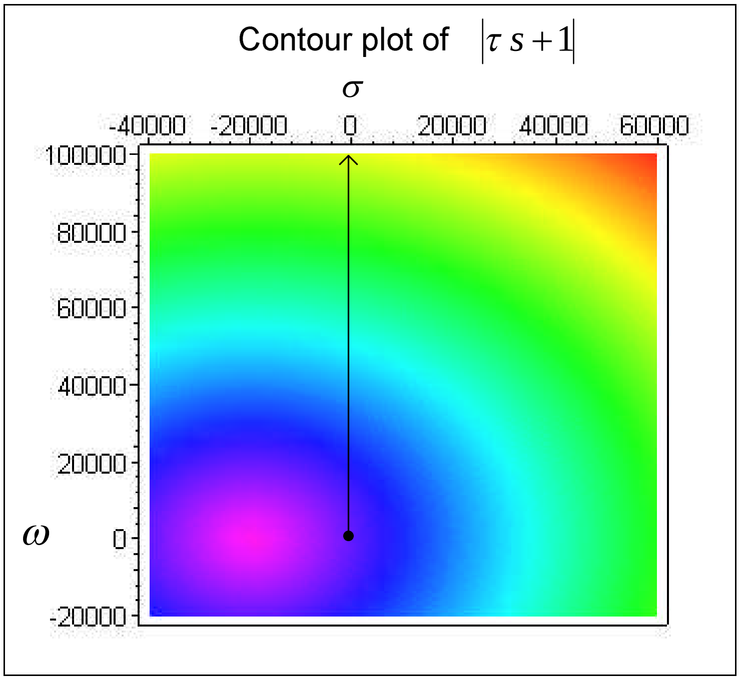


Figure 4: contour plot of the magnitude of the transfer function so far.

Besides being a pretty picture what does figure 4 show us? Well, it shows us if we are near a zero and move away from it, our gain goes up.

But there's more to life than just zeros. There's something called a pole. A pole is a beastie that goes up instead of down like a zero as you approach it. In fact mathematically a pole is just the reciprocal of a zero. This means if you're standing at a pole, you're at infinity, and when you go away from it, you'll gain goes down.

So if we were to put a pole slightly to the left of the zero we have in figure 4, as we stand on the black line at a point with a very high frequency (a long way from both the pole and the zero), the pole and the zero will seem to be more or less the same distance from us, and hence will cancel each other out so our gain stops increasing. But, at low frequencies to zero still be much closer to us than the pole, meaning the zero dominates the frequency response, and we should still have a reasonably similar frequency response as to that in figure 2 for low frequencies. That's the idea anyway, let's see what really happens.

Pole dancing.

Adding a pole to the transfer function in formula 2 gives us the new transfer function of formula 14.

$$H_a(s) = \frac{a\,s+1}{b\,s+1} \qquad (14)$$

This one will work, but lets take it step-by-step first. As before we calculate the decibel power gain of the filter.

$$H_a(i\omega) = \sqrt{\frac{a^2\,\omega^2+1}{b^2\,\omega^2+1}}e^{i\theta}$$
$$\Rightarrow P_{dB}(\omega) = 10\log_{10}\left(\frac{a^2\,\omega^2+1}{b^2\,\omega^2+1}\right) \qquad (15)$$

Then as before we derive a formula that relates the 3 dB corner and the time constant τ together.

$$P_{dB}(1/\tau) = 3dB \quad \text{Definition of time constant.}$$
$$\Rightarrow \frac{a^2\,(1/\tau)^2+1}{b^2\,(1/\tau)^2+1} = 2 \qquad \text{By formula 15.}$$
$$\Rightarrow a = \sqrt{2b^2+\tau^2} \qquad (16)$$

In fact because b must be small, formula 16 is approximately equal to formula 6. That gets *a* out of the way, but what do we do about *b*? Well, as I see it, I can think of a few ways of doing it. The first way would be setting the maximum decibel power gain minus 3 dB to some frequency like 20 kHz (this is above the maximum frequency we can transmit audio. The second way would be to stipulate the maximum rate of change of the power gain of the filter and set *b* accordingly. Thirdly you could stipulate a maximum power gain and set *b* accordingly. I'll start off by doing it the first way. Let δ be the reciprocal of the angular frequency that produces 3 dB less than the maximum power gain.

$$P_{dB}(1/\delta) = 10\log_{10}\left(\frac{a^2(1/\delta)^2+1}{b^2(1/\delta)^2+1}\right) \quad \text{By formula 15.}$$

$$\Rightarrow 10\log_{10}\left(\frac{a^2(1/\delta)^2+1}{b^2(1/\delta)^2+1}\right) = 10\log_{10}\left(\frac{a^2}{b^2}\right) - 3 \quad \text{As this is 3 dB less than maximum power.}$$

$$\Rightarrow 10\log_{10}\left(\frac{b^2}{a^2}\left(\frac{a^2(1/\delta)^2+1}{b^2(1/\delta)^2+1}\right)\right) = -3$$

$$\Rightarrow \frac{b^2}{a^2}\left(\frac{a^2(1/\delta)^2+1}{b^2(1/\delta)^2+1}\right) = 1/2$$

$$\Rightarrow \frac{b^2a^2+b^2\delta^2}{a^2b^2+a^2\delta^2} = 1/2$$

$$\Rightarrow a^2 = \frac{2b^2\delta^2}{\delta^2-b^2}$$

$$\Rightarrow 2b^2+\tau^2 = \frac{2b^2\delta^2}{\delta^2-b^2} \quad \text{By formula 16.}$$

$$\Rightarrow b = \frac{\sqrt{-\tau^2+\sqrt{\tau^4+8\tau^2\delta^2}}}{2} \qquad (17)$$

$$\Rightarrow b \approx \frac{\sqrt{-\tau^2+\sqrt{16\delta^4+\tau^4+8\tau^2\delta^2}}}{2} \quad \text{As } \delta << \tau$$

$$\Rightarrow b \approx \frac{\sqrt{-\tau^2+\sqrt{(4\delta^4+\tau^4)^2}}}{2}$$

$$\Rightarrow b \approx \frac{\sqrt{-\tau^2+4\delta^4+\tau^4}}{2}$$

$$\Rightarrow b \approx \delta \qquad (18)$$

For the second method equations get kind of messy, but there's nothing really difficult about them. As in the first transfer function we looked at, we calculate the rate of change of gain of power in dB with respect to octaves as below.

$$\frac{d}{doct}P_{dB} = \frac{20\ln(2)\left(b^2+\tau^2\right)\omega^2}{\ln(10)\left(2b^2\omega^2+\tau^2\omega^2+1\right)\left(b^2\omega^2+1\right)}$$ By formula 15. (19)

Differentiate with respect to ω and set to zero to get the maximum rate of change than solve for *b*.

$$\frac{d}{d\omega}\left[\frac{d}{doct}P_{dB}\right] = \frac{-40\ln(2)\left(b^2+\tau^2\right)\omega\left(2b^4\omega^4+\tau^2\omega^4b^2-1\right)}{\ln(10)\left(2b^2\omega^2+\tau^2\omega^2+1\right)^2\left(b^2\omega^2+1\right)^2}$$

$$\frac{d}{d\omega}\left[\frac{d}{doct}P_{dB}\right] = 0$$

$$\Rightarrow \omega_{\max} = \left(b^2\tau^2+2b^4\right)^{-1/4}$$

$$\Rightarrow b = \frac{\sqrt{-\tau^2+\sqrt{\tau^4+8\omega_{\max}^{-4}}}}{2} \qquad (20)$$

The third method is really easy because looking at formula 15 the maximum power is the following.

$$P_{dBMAX} = 10\log_{10}\left(\frac{a^2}{b^2}\right)$$ By formula 15. (21)

Then solve for *b*.

$$a^2 = b^2 10^{P_{dBMAX}/10}$$

$$\Rightarrow b = \frac{\tau}{\sqrt{10^{P_{dBMAX}/10}-2}}$$ By formula 16. (22)

Now we can calculate some values for *a* and *b*.

Evaluating A and B.

In FM stereo radio, at 19 kHz there is a pilot. Because of this pilot you must limit the audio frequencies to somewhat less. 15 kHz seems to be a common stopping point. As I said before for my calculations I'm going to use 50 µs pre-emphasis time constant.

Using the first method described above (formula 17) I need $1/\delta$ something higher than the frequencies I'm interested in. This being the case let the frequency be say 20 kHz.

$$1/\delta = 2\pi \times 20000$$

$$\Rightarrow \delta = \left(1/\left(2\pi \times 20000\right)\right)$$

$$\Rightarrow \delta \approx 7.958 \times 10^{-6}\ s/rad$$
$$\tau = 50 \times 10^{-6}\ s/rad$$
$\Rightarrow b = 7.772 \times 10^{-6}$ By formula 17.
$\& \Rightarrow a = 51.19 \times 10^{-6}$ By formula 16.

As formula 18 says, $b \approx \delta$ and the comment just under formula 16 that $a \approx \tau$. It's always interesting when things turn out to be so simple.

For the second method we need to choose a frequency where we want the steepest change of power in dB with respect to octaves to happen. As our first transfer function could do no better than 6 dB an octave, we will do worse with this transfer function because of the added pole. So let's select the frequency to be somewhere in the middle, say 8 kHz.

$$\omega_{\max} = 2\pi 8000\ rad/s$$
$$\tau = 50 \times 10^{-6}\ s/rad$$
$\Rightarrow b = 7.733 \times 10^{-6}$ By formula 20.
$\& \Rightarrow a = 51.18 \times 10^{-6}$ By formula 16.
$\& \Rightarrow \left[\frac{d}{doct} P_{dB}\right]_{MAX} = 4.44\ dB/octave$ By formula 19.

Notice we get more or less the same values using this method as the last.

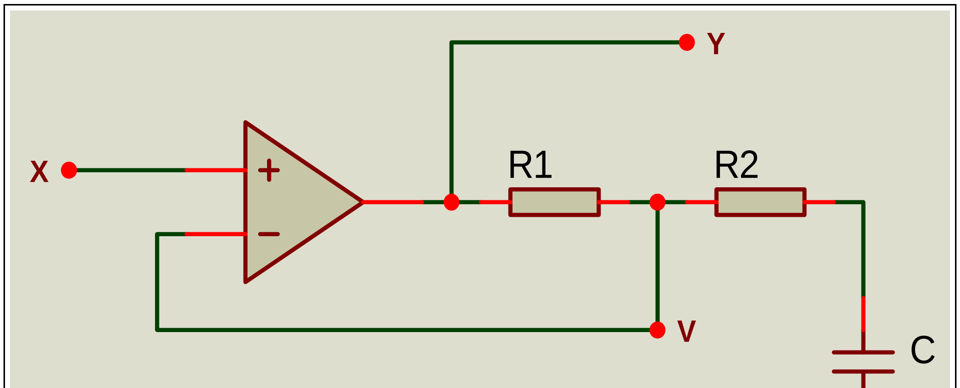
And finally for the last method we choose the maximum gain that we are prepared to cope with, let's say 17 dB.

$$P_{dBMAX} = 17\,dB$$
$\Rightarrow b = 7.208 \times 10^{-6}$ By formula 22.
$\& \Rightarrow a = 51.03 \times 10^{-6}$ By formula 16.

So there we have it, three different ways of evaluating these pesky constants. Method one is probably the most useful because of the approximations that can be made. However, I'm sure the other two methods have their own merit.

The electronics of this transfer function.

Now comes the time to have a look at what electronics this transfer function relates to. With a bit of thinking a possible solution is figure 5.



$$Y \approx X \frac{(C(R_1 + R_2))s + 1}{(CR_2)s + 1}$$

Figure 5: schematic representation of our new transfer function.

I calculated the transfer function the same way as I did in formulas 11 to 13.

$Y = G(X - V)$ What an op-amp does. (23)

$V = Y \frac{R_2 + 1/sC}{R_1 + R_2 + 1/sC}$ Voltage divider network.

$\Rightarrow V = Y \frac{R_2 Cs + 1}{(R_1 + R_2)Cs + 1}$ (24)

$Y = G\left(X - Y \frac{R_2 Cs + 1}{(R_1 + R_2)Cs + 1} \right)$ By formula 23 and 24.

$$\Rightarrow \frac{Y}{G} + Y \frac{R_2 Cs + 1}{(R_1 + R_2)Cs + 1} = X$$

$$\Rightarrow Y\left(\frac{1}{G} + \frac{R_2 Cs + 1}{(R_1 + R_2)Cs + 1} \right) = X$$

$\Rightarrow Y\left(\frac{R_2 Cs + 1}{(R_1 + R_2)Cs + 1} \right) \approx X$ Assuming that the gain of the op amp is very large.

$$\Rightarrow \frac{Y}{X} \approx \frac{(R_1 + R_2)Cs + 1}{R_2 Cs + 1}$$

$\Rightarrow H_a(s) \approx \frac{(C(R_1 + R_2))s + 1}{(CR_2)s + 1}$ (25)

This is formula 14 with $a = C(R_1 + R_2)$, and $b = CR_2$. As my main goal is to implement this transfer function digitally, the analog implementation is not of great concern to me. However, while we here we might as well model it on the computer.

Choosing the capacitor $C = 4.7nF$ and a and b from the first method described previously, I get $R_1 \approx 9.2K$ and $R_2 \approx 1.6K$. As I know 10 K and 1.5 K resistors exist I'll use these as approximations. Using the following component values and a general-purpose audio opamp, I get the plot in figure 6.

$C = 4.7nF$
$R_1 = 10K$
$R_2 = 1.5K$

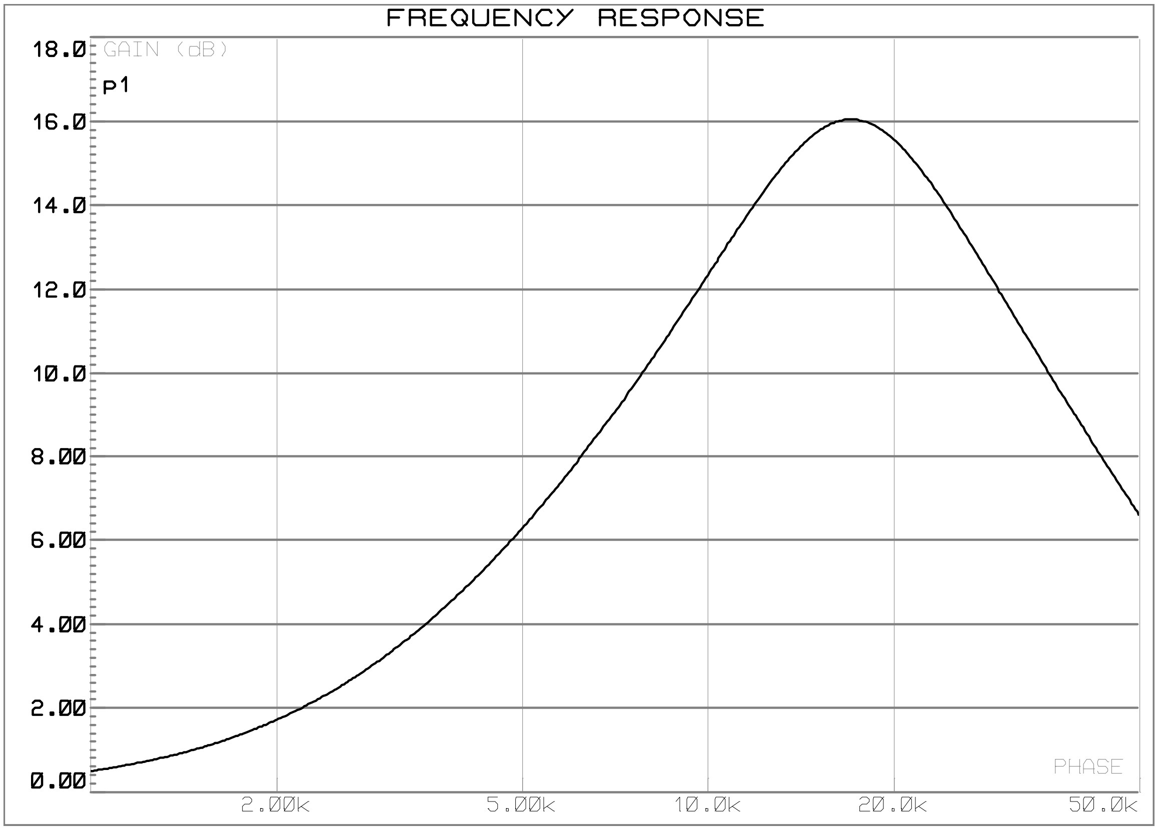


Figure 6: Computer simulated frequency response of the schematic in figure 5.

This looks reasonably acceptable. Well, enough of this analog stuff let's convert what we know about this filter into a digital implementation.

From S plane to Z plane. The great divide.

So far we have been dealing with the S plane and the Laplace transformation. For discrete systems the equivalent is the Z plane and its transformation. The Z plane is polar whilst S plane is Cartesian. http://www.dspguide.com/CH33.PDF gives a good insight about the Z plane, also try http://en.wikipedia.org/wiki/Z-transform. Figure 7 shows how the two planes are related.

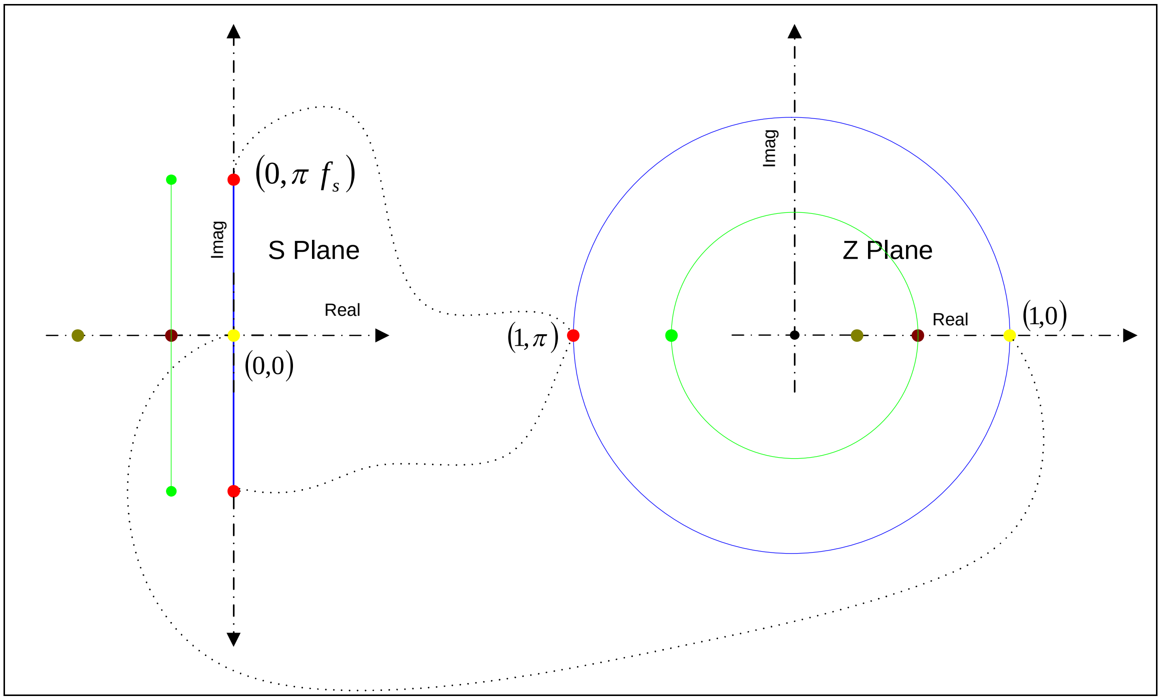


Figure 7: How the S and the Z plane are related
with respect to the Laplace and Z transformations respectively.

Relationship between the points are $z = e^{sT}$ where T is sample period. What I mean by the relationship between the two planes, is the following. If we have a time continuous function in the time domain and perform the Laplace transform on it, we get the points of the S plane. Now, if we sampled this same time continuous function in the time domain at intervals $T = 1/f_s$ and performed the Z transformation on this, we would get the points in the Z plane. This is what I mean by the points are related by $z = e^{sT}$. This relationship can be shown by the following. First we calculate the Laplace transform of a sampled function $x(t)$. Once sampled the function becomes $\Delta_T(t)x(t)$. Where $\Delta_T(t)$ is a Dirac comb that plucks out the sampled values in $x(t)$.

$$\Delta_T(t) = \sum_{n=0}^{\infty} \delta(t - nT) \quad \text{Dirac comb.}$$

$$X(s) = \int_{-\infty}^{\infty} \Delta_T(t)x(t)e^{-st}\,dt \quad \text{By definition the Laplace transform of the sampled function.}$$

$$X(s) = \int_{-\infty}^{\infty} \sum_{n=0}^{\infty} \delta(t - nT)x(t)e^{-st}\,dt \quad \text{By def of Dirac comb.}$$

$$X(s) = \sum_{n=0}^{\infty} \left(\int_{-\infty}^{\infty} \delta(t - nT)x(t)e^{-st}\,dt \right)$$

$X(s) = \sum_{n=0}^{\infty} x(nT) e^{-snT}$ Dirac delta plucking out one value.

$X(s) = \sum_{n=0}^{\infty} x[n] e^{-snT}$ Defining $x[n] = x(nT)$ discrete set.

$$X(s) = \sum_{n=0}^{\infty} x[n] \left(e^{sT}\right)^{-n}$$

$X(z) = \sum_{n=0}^{\infty} x[n] z^{-n}$ By definition of the Z transform.

$$\therefore z = e^{sT}$$

And the relationship falls out. Also note the following.

$$z = e^{sT}$$
$$\Rightarrow z = e^{\sigma T} e^{i\omega T}$$
$$\Rightarrow \theta = \omega T$$
$$\& \Rightarrow r = e^{\sigma T}$$

Where θ is the angle the point *z* in the Z plane makes with the real axis, and *r* the distance from the center of the circle.

The most important line is the imaginary axis line in the S plane and the unit circle in the Z plane. This line or the circle contain everything you need to know about the frequency response of the filter. Also note frequencies like $\theta + \pi f_s, \theta + 2\pi f_s$, etc radians get mapped to the same point in the Z plane.

The idea now is to transform our transfer function from the S plane to the Z plane, then perform the inverse Z transform to get a discrete function. This is the function that actually performs the filtering on the computer.

We have a problem though; how do we transform our transfer function? There is no single method. If you look around on the net you will see names like bilinear transform, matched Z, invariant impulse, step invariant, and others. Shortly it will become clear that we need after this transformation to have a transfer function that is a rational function in the variable *z*. That rules out using $s = (1/T)\ln(z)$ directly, but still allows to use approximations for it. That is the method I am going to choose to do it.

$z = e^{sT}$ Known relationship.

$$\Rightarrow z = e^{sT/2} e^{sT/2}$$
$$\Rightarrow z = \frac{e^{sT/2}}{e^{-sT/2}}$$

$$\Rightarrow z = \frac{1 + sT/2 + \ldots}{1 + (-sT/2) + \ldots}$$ Taylor expansion.

$$\Rightarrow z \approx \frac{1 + sT/2}{1 - sT/2}$$ First order approximation.

$$\Rightarrow z - zsT/2 \approx 1 + sT/2$$
$$\Rightarrow z - 1 \approx zsT/2 + sT/2$$
$$\Rightarrow \frac{2}{T} z - 1 \approx s(z+1)$$
$$\Rightarrow s \approx \frac{2}{T}\frac{z-1}{z+1}$$
$$s := \frac{2}{T}\frac{z-1}{z+1}$$ Bilinear transformation (http://en.wikipedia.org/wiki/Bilinear_transform)

This approximation is just what we need. It's called the bilinear transform. Firstly it's a relatively simple approximation of $(1/T)\ln(z)$ with nothing but a couple of linear polynomials. Secondly, clearly any point in the Z plane only comes from one point in the S plane, unlike $s = (1/T)\ln(z)$.
Let's have a look how frequencies now appear in each plane.

For probing frequencies under the Z transform we use $e^{i\omega T}$ whilst probing frequencies under the Laplace transform we use $i\omega$. Now, how does our newly acquired digital filter respond to a frequency of ω if given such a signal? Take the point z in the Z plane that lies on the unit circle and has an angle of ωT, i.e. $e^{i\omega T}$. To determine where this point came from before the bilinear transformation we do the following.

Let $s = \frac{2}{T}\frac{z-1}{z+1}$ Making the bilinear transformation approximation.

$$s = \frac{2}{T}\frac{e^{i\omega T} - 1}{e^{i\omega T} + 1}$$ Evaluating on the unit circle.

$$s = \frac{2}{T}\frac{(e^{i\omega T} - 1)e^{-i\omega T/2}}{(e^{i\omega T} + 1)e^{-i\omega T/2}}$$
$$s = \frac{2}{T}\frac{(e^{i\omega T/2} - e^{-i\omega T/2})}{(e^{i\omega T/2} + e^{-i\omega T/2})}$$
$$s = \frac{2}{T}\frac{(e^{i\omega T/2} - e^{-i\omega T/2})2i}{(e^{i\omega T/2} + e^{-i\omega T/2})2i}$$
$$s = i\frac{2}{T}\frac{(e^{i\omega T/2} - e^{-i\omega T/2})2}{(e^{i\omega T/2} + e^{-i\omega T/2})2i}$$

$$s = i\frac{2}{T}\tan\left(\frac{\omega T}{2}\right)$$

$$\Rightarrow \omega_a = \frac{2}{T}\tan\left(\frac{\omega T}{2}\right)$$ As was in the form of probing the analog frequency. (26)

$$\Rightarrow \omega = \frac{2}{T}\tan^{-1}\left(\frac{T\omega_a}{2}\right)$$

This means under this transformation the digital filter responds at ω as the analog filter responded at ω_a. Because of the tangent function, low frequencies will tend to be fairly linear but the high frequencies will be compressed and hence distorted and warped. We can correct for this at frequencies that we can control in the design off the analog filter. It's called pre-warping. Figure 8 shows in an intuitive way how we do this.

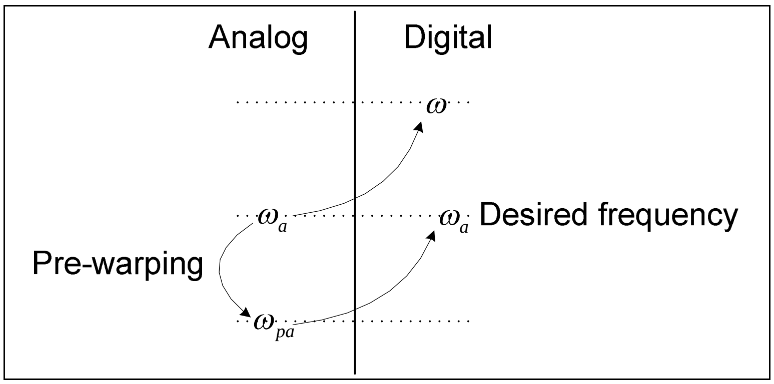


Figure 8: Intuitive idea how pre-warping works.

So to select any frequency in our analog filter we do not choose it directly, but use formula 26 and plug in ω for our desired frequency and use ω_a in our analog filter design.

Now we perform the bilinear transform on our transfer function formula 14.

$$H_a(s) = \frac{a\,s+1}{b\,s+1}$$ From formula 14.

$$\Rightarrow H_d(z) = \frac{a\left(\frac{2}{T}\frac{z-1}{z+1}\right)+1}{b\left(\frac{2}{T}\frac{z-1}{z+1}\right)+1}$$ Bilinear transform.

$$\Rightarrow H_d(z) = \frac{2a(z-1)+T(z+1)}{2b(z-1)+T(z+1)}$$

$$\Rightarrow H_d(z) = \frac{2az-2a+Tz+T}{2bz-2b+Tz+T}$$

$$\Rightarrow H_d(z) = \frac{(2a+T)z + (T-2a)}{(2b+T)z + (T-2b)}$$

$$\Rightarrow H_d(z) = \frac{(2a+T) + (T-2a)z^{-1}}{(2b+T) + (T-2b)z^{-1}} \qquad (27)$$

This is our transfer function of our digital filter. Using the Z transform's timeshifting property, and definition of the transfer function, we can write the digital filter in a recursion relation by performing an inverse Z transform.

$$\Rightarrow \frac{Y(z)}{X(z)} = \frac{(2a+T) + (T-2a)z^{-1}}{(2b+T) + (T-2b)z^{-1}} \quad \text{From formula 27.}$$

$$\Rightarrow Y(z)(2b+T) + z^{-1}Y(z)(T-2b) = X(z)(2a+T) + z^{-1}X(z)(T-2a)$$

$$\Rightarrow y[n](2b+T) + y[n-1](T-2b) = x[n](2a+T) + x[n-1](T-2a) \quad \text{Inverse Z transform.}$$

$$\Rightarrow y[n] = \frac{(2a+T)}{(2b+T)}x[n] + \frac{(T-2a)}{(2b+T)}x[n-1] + \frac{(2b-T)}{(2b+T)}y[n-1] \qquad (28)$$

$$\Rightarrow y[n] = a_0 x[n] + a_1 x[n-1] + b_1 y[n-1] \qquad (29)$$

$$\Rightarrow H_d(z) = \frac{a_0 + a_1 z^{-1}}{1 - b_1 z^{-1}}$$

Formula 28 (or 29) is the recursion relation that actually does the filtering on the computer.

Putting it all together.

$$\tau_P = \frac{T}{2}\cot\left(\frac{T}{2\tau}\right) \quad \text{Pre-warping } \tau .$$

$$\delta_P = \frac{T}{2}\cot\left(\frac{T}{2\delta}\right) \quad \text{Pre-warping } \delta .$$

$$\Rightarrow b_P = \frac{\sqrt{-\tau_P^2 + \sqrt{\tau_P^4 + 8\tau_P^2\delta_P^2}}}{2} \quad \text{By formula 17.}$$

$$\& \Rightarrow a_P = \sqrt{2b_P^2 + \tau_P^2} \quad \text{By formula 16.}$$

$$\Rightarrow a_0 = \frac{(2a_P + T)}{(2b_P + T)} \quad \text{By formula 28 \& 29.}$$

$$\& \Rightarrow a_1 = \frac{(T - 2a_P)}{(2b_P + T)} \quad \text{By formula 28 \& 29.}$$

$$\& \Rightarrow b_1 = \frac{(2b_P - T)}{(2b_P + T)} \quad \text{By formula 28 \& 29.}$$

For my implementation on the computer, I will be using a soundcard sampling at 192000 times a second. I think that $1/\delta$ corresponding to 20 kHz should be acceptable in my setup. Plugging in the numbers, table 1 is what I get.

f_s	$192{,}000\ samples/s$
τ	$50\times10^{-6}\ s/rad$
δ	$7.96\times10^{-6}\ s/rad$
T	$5.21\times10^{-6}\ s/sample$
τ_P	$49.95\times10^{-6}\ s/rad$
δ_P	$7.67\times10^{-6}\ s/rad$
b_P	7.50×10^{-6}
a_P	51.07×10^{-6}
a_0	5.30986
a_1	-4.79461
b_1	0.48475

Table 1: Calculated values for a time constant of 50 µs, a $1/\delta$ corresponding to 20 kHz, and a sample rate of 192,000 samples a second.

Figure 9 shows the gain and phase response of the digital filter.

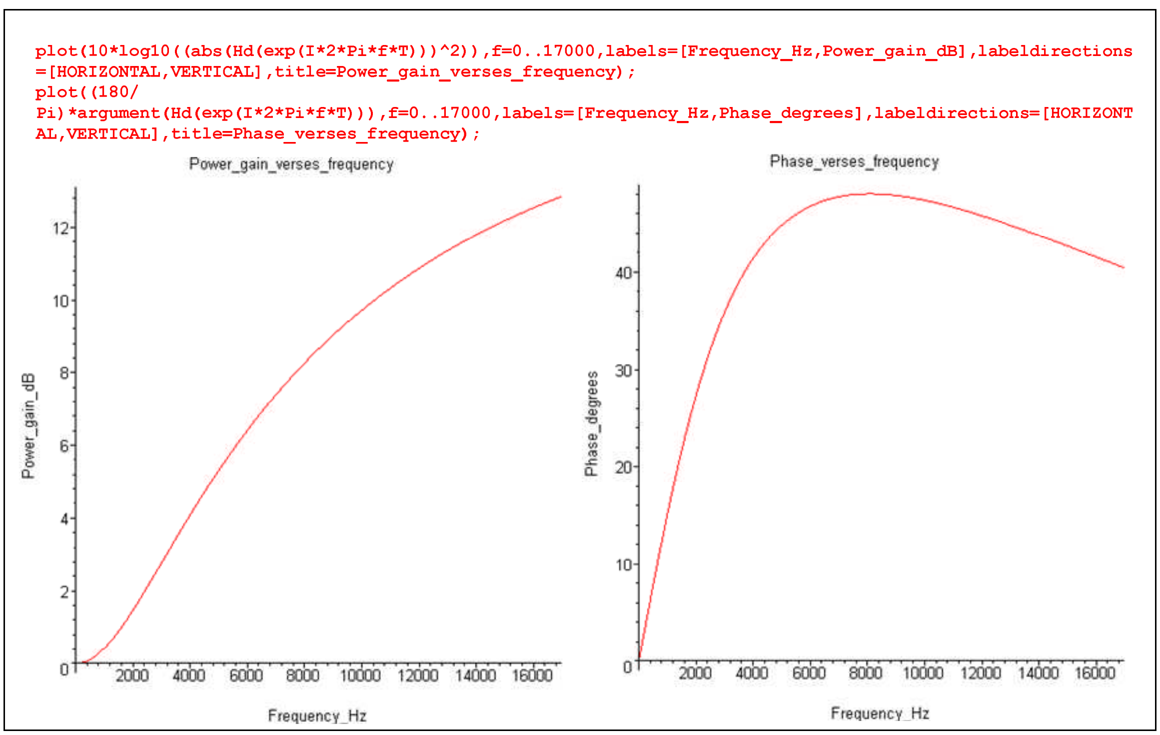


Figure 9: Gain and phase response of the digital filter.

Figure 10 shows the contour plot of the magnitude of the digital transfer function. You can see the zero as in figure 4, but now we also have a pole close to the left of the zero.

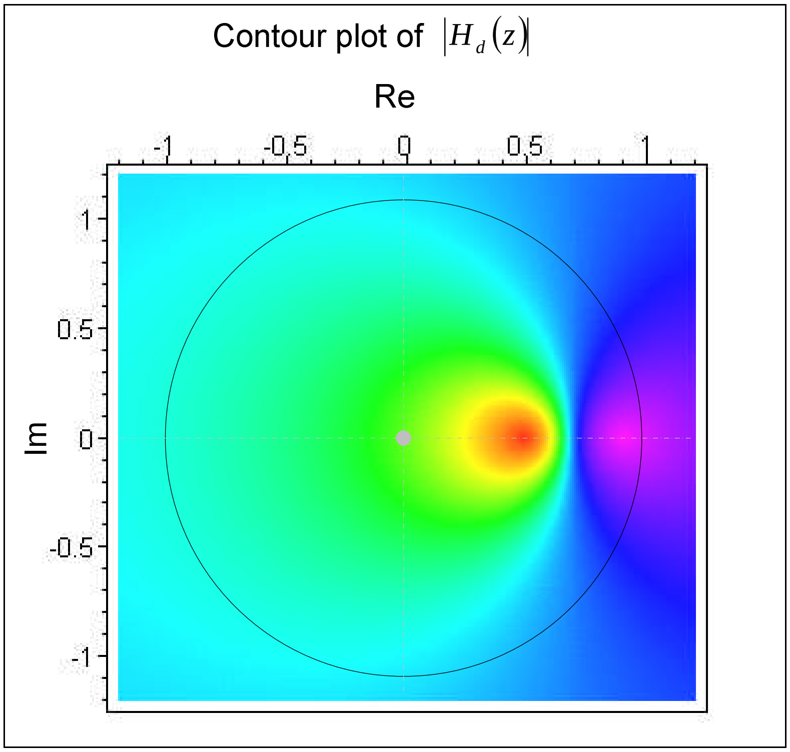


Figure 10: Contour plot of the magnitude of the digital transfer function.

All that remains is to write a routine to implement this. Listing 1 shows a C++ program code snippet for doing the filtering. The update part of the code must be run once a sample.

```
void Init()
{
	a[0]=5.309858008;
	a[1]=-4.794606188;
	b[1]=.4847481783;
	x=0;
	y=0;
}

double Update(double signal)
{
	y=a[0]*signal+a[1]*x+b[1]*y;
	x=signal;
	return y;
}
```

Listing 1: C++ code snippet of filter implementation.

Amazing, all that hard work for code that only is a few lines. Implementing this code and supplying the soundcard with white noise and using another soundcard to listen to the response we get figure 11. There is a low pass FIR filter with a cutoff frequency of 17 kHz insisted just prior to the IIR pre-emphasis filter (hence the role of around 16 kHz). This is an actual real-life power gain response to the C++ code in listing one.

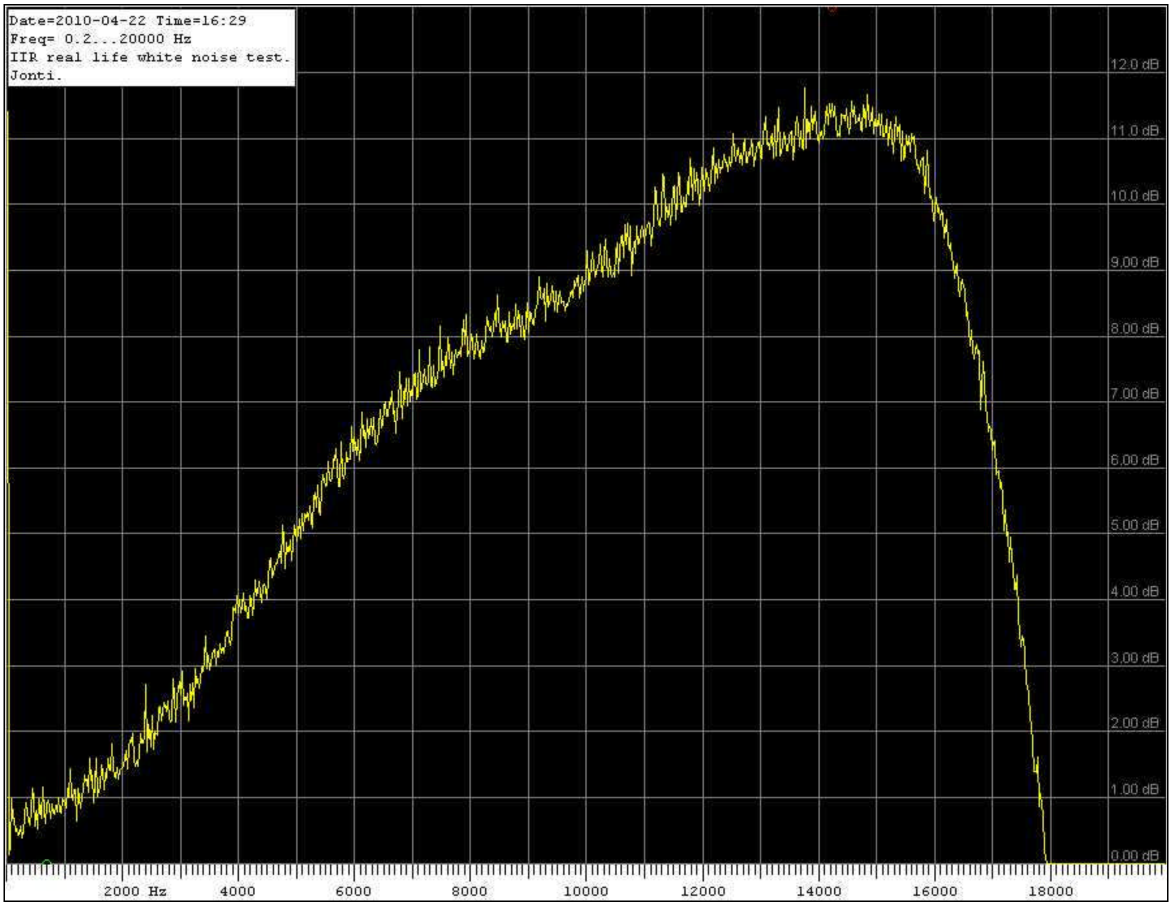


Figure 11: Real-life white noise test of the IIR pre-emphasis filter.

All things being equal figure 11 is a relatively good match with figure 9. As always with an implementation, when actual results correlate well with theoretical models, then, that's good.

A few final words.

Of course there are a myriad of ways that you could create a pre-emphasis filter. This method is just one, and as far as I can tell it's acceptable. FIR filters could be used, but are large and slow. IIR filters provide an advantage in that, most likely at the receiving end there are real-life electronic components doing the de-emphasis, and IIR filters are equivalent to real-life electronic components so it makes sense to pursue the IIR route in the hope that the pre-emphasis filter and the de-emphasis filter have a better match. Also IIR filters are very quick. Other options would be to investigate the use of more poles and zeros as well as do more real-life testing with radio transmitters and receivers.

This filter is used in my MPX stereo encoder software for stereo transmission on the 88-108 MHz band. See http://wwwjontio.zapto.org/mpxencoder for this program.

Jonti.
22/4/10