# Filtering PWM Signals

Jim Wagner October, 2009 Rev 3



## Table of Contents



Many microprocessor system designers, and especially those with a software background, tend to have a real problem figuring out what to do with a pulse-width modulated (PWM) signal. This tutorial attempts to provide a soundly-based set of "rules of thumb" and a design procedure that will help any designer make these choices.

## A. Fundamentals of PWM Signals

Before we start discussing how filter PWM signals, it may be worthwhile to review the basics of the PWM technique, why it is so useful, and what it means when we say “modulation”.

Lets begin with some definitions. For those definitions, please refer to Figure 1.

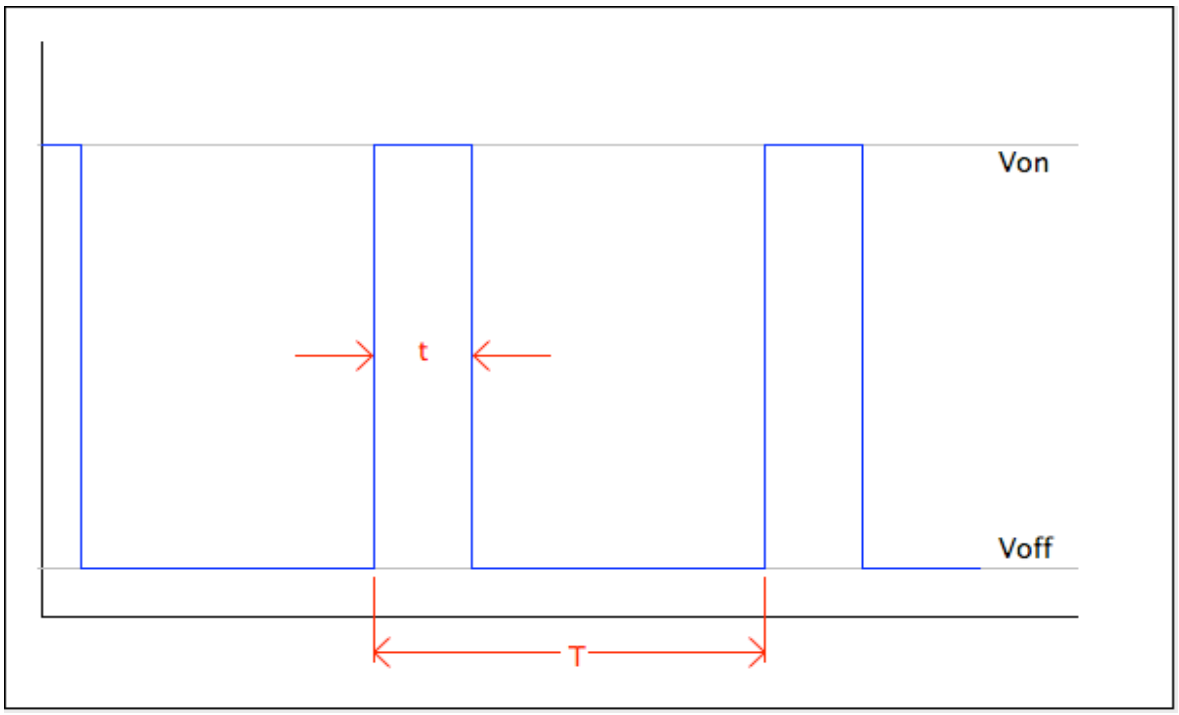


*Figure 1 – PWM Parameters*

• PWM Period is the time duration of one PWM cycle. It is denoted by the parameter “T” in Figure 1. In most applications, the period remains constant, though it does not have to.

• PWM Frequency is the repetition frequency of the PWM cycle. It is 1/T, and can be given in units such as hz (Hertz) or khz (kiloHertz). Please note that this is NOT the clock frequency of the PWM hardware or software; it is the repetition frequency of the PWM output signal.

• Pulse Width is the time during which one PWM cycle is “ON”. Whether or not ON is logic high or logic low depends on the circuit application. In Figure 1, logic high is assumed to be the ON state and the time duration of this interval is marked as “t”.

• Duty Cycle is the ratio of the ON time to the period (t/T). It is often given the symbol D. D can vary from 0 to 1; 0 indicates that t=0 or that there is no ON time while 1 indicates t=T or that it is always ON.

Our next question to consider is why anybody would be interested in this sort of signal. There really are several strong reasons for interest in PWM signals and there are several applications where several of these reasons come together. Here are at least some of those reasons for PWM interest:

1. Average Operation - many circuits are able to "average" an on-off signal to control their operation. It is straight-forward to determine the required duty cycle to achieve a desired averaged value from a signal with given high and low levels. Examples include LEDs that are viewed by humans and inductive loads such as motors and solenoids. If you don't know the relationship, it is given by

   $$Vavg = D*Von + (D-1)*Voff$$

   where Von is the logic voltage level during time "t" (shown in Figure 1) and Voff is the logic level during the remainder of the cycle.

2. Reduced Power Loss – switched circuits tend to have lower power consumption because the switching devices are almost always off (low current means low power) or hard-on (low voltage drop means low power). Common circuits that utilize this feature include switched-mode power supplies, Class D audio power amplifiers, and motor drivers. Frequently, these circuits use semi-analog techniques (ramps and comparators) rather than digital techniques, but the advantages still hold.

3. Easy to Generate – PWM signals are quite easy to generate. Many modern microcontrollers include PWM hardware within the chip; using this hardware often takes very little attention from the microprocessor and it can run in the background without interfering with executing code. On the other hand, PWM signals are also quite easy top create directly from code, often requiring only counting and comparing operations.

4. Digital to Analog Conversion – pulse width modulation can function effectively, as a digital to analog converter, particularly combined with appropriate filtering. The fact that the duty cycle of a PWM signal can be accurately controlled by simple counting procedures is one of the reasons why PWM signals can be used to accomplish digital-to-analog conversion.

When the term "PWM" is used, a key element is "modulated", the "M" in "PWM". In this setting, modulate means "to vary or change". This is because few PWM

applications set the duty cycle to some fixed value and never change it. Usually, the duty cycle is shifted over time to control the motor current or the output voltage or what ever is being controlled. The change may occur infrequently or it may happen continuously, but it is still “modulation”. Figure 2 shows a PWM signal that is coarsely modulated with a ramp waveform (its modulation); it is coarse because the duty cycle change in every PWM cycle is very large (just to make it visible in a small graph – in “real life”, such a ramp frequency would be too close to the PWM repetition frequency to be useful).

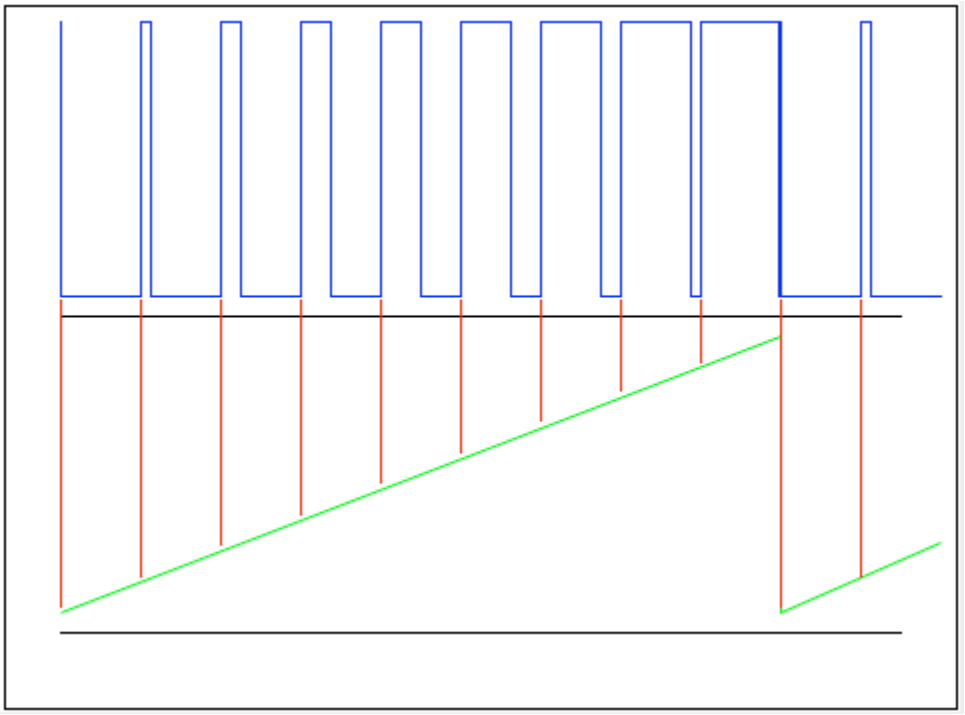

*Figure 2 – Ramp Modulated PWM Signal*

Figure 2 includes a pulse-width modulated (PWM) square wave signal in blue and the modulation (a ramp signal) in green. The vertical red marks indicate the start of each PWM cycle and the corresponding ramp value that determines the width of the next pulse.

In many applications, the modulation, itself, has a frequency. This is particularly true if the modulation represents a single tone. It is also true if the modulation is simply periodic, as is the case with the ramp in Figure 2. One needs to be a bit careful, however, because because a periodic signal that is not a pure “sine” or “cosine” is actually made up of a number of harmonics. Then, one should

probably speak of the “modulation bandwidth” which is the range of frequencies needed to accurately reproduce that signal..

An important concept is that the PWM signal is really a combination of two signals. One is the obvious pulse signal and one is the (somewhat hidden) modulation. This idea will be discussed in much more detail later.

## B. Why is a Filter Needed and When is a Filter Needed?

The main reason to filter a PWM signal is to extract the modulation from the combined signal.

There are a FEW cases where filtering is really not needed. One of those is drive for human-observed LEDs. The observer's eye provides all of the filtering needed if the PWM repetition frequency is above about 30Hz.

Likewise, motor drives often need no filtering. This is the case because the averaging effect of the inertia of the rotating part of the motor plus its load and inductance of the motor combine to provide effective filtering.

Another case where filtering is used very little is the "Class D" audio amplifier. This is a circuit that converts an analog signal into a PWM signal to drive a loudspeaker. Here, the PWM is designed to be at a high enough frequency that the listener does not detect its presence. Also, the inertia of the speaker diaphragm adds some effective filtering.

## C. What Kind of Filter is Needed?

The best answer to this question is: "It all depends". It depends on how the result is to be used and it depends on the nature of the source PWM signal.

The filter choice generally DOES NOT depend (very much) on the kind of circuit or microprocessor used to generate the source signal. Thus, what is described in the following text does not depend on whether the source is a DSP, a CPLD, an ARM processor, a Freescale processor, an Atmel processor, a MicroChip processor, or even discrete logic.

## D. Some Basic Terminology

Design of filters is a discipline with deep theoretical underpinnings. As a result, there is an extensive and frequently obscure set of descriptive terms that are associated with filters. This short exposition cannot do justice to all of this, especially when your background is software. But, there are a few terms that we really need to bring up because it is so difficult to discuss even the most basic filter without using them.

Low Pass Filter This is a filter that reduces (attenuates) high frequency signals and lets low frequency signals through with little modification.

Gain: Gain is a measure of the increase in signal amplitude. Formally, it is the ratio of the output amplitude to the input amplitude. Gain is expected to be greater than 1. Gain less than 1 is really attenuation.

Attenuation: Attenuation is a measure of how much a signal is reduced. Formally, it is the ratio of the output amplitude to the input amplitude, with the expectation that the ratio is less than 1. Gain and attenuation are two ways of looking at exactly the same thing.

Corner Frequency: For a low pass filter, this is the frequency at which the attenuation begins to increase with increasing frequency. With some filter types, this is defined as the point where the attenuation is 3db (see next definition) greater than it is at low frequencies. However, this definition is not consistent for all filter types.

Decibels: This is a system that is often used to measure attenuation and gain. To most of us, it seems rather abstract. For voltage amplitudes, it is defined as

$$db = 20 * LOG(Vout/Vin)$$

Most of us are not familiar with logarithms any more, if we ever were. So, here are some useful values:

| Vout/Vin | db |
|---|---|
| 1 | 0 |
| 10 | 20 |
| 100 | 40 |
| 0.1 | -20 |
| 0.01 | -40 |
| 1.414 | 3 |
| 2 | 6 |
| 0.707 | -3 |
| 0.5 | -6 |

Note that that quirks of gain and attenuation leave us with an odd situation. If you are talking about gain, positive decibel values denote actual gain and negative decibel values represent attenuation. But, if you are talking about attenuation, real attenuation is a positive decibel value and gain is represented as a negative decibel value. That is, a gain of 3db is an attenuation of -3db and an attenuation of 3db is a gain of -3db.

Active/Passive Filters: A filter that contains only resistors, inductors, capacitors, and transformers is generally considered "passive". Note that this does not specify whether or not the output voltage is, or is not, greater in magnitude than the input. However, it DOES imply that there is no increase in the available signal power. An active filter usually contains an amplifier, but could contain other devices, such as tunnel diodes, which CAN (but does not have to) result in an increase in signal power level.

Filter Order or Number of "Poles": This might be described better using examples since it is a fairly abstract concept.

A filter with one capacitor or one inductor but maybe other components is a one pole or first order filter.

A filter with two capacitors, or two inductors, or one capacitor and one inductor is a second order or two pole filter.

A combination of 3 inductors and or capacitors is a third order or three pole filter.

This is important because the number of poles controls the shape of the frequency response of the filter.

Ripple: Ripple is the amplitude of the source PWM signal that "leaks" through the filter.

Step Response: Step response describes how a filter responds to a step input. It is a time description rather than a frequency description. Every filter has BOTH a frequency response AND a step response.

Steady State: This describes the time response when each output cycle is exactly like the one before it, in terms of voltage. That is, any two points, one PWM period apart, are exactly the same voltage.

## E. Normalization

Before we start looking at filters in any detail, we need to take care of an important fact. That fact is that both the PWM signals, themselves, as well as the filters, are “linear systems”. While this designation carries a lot of “baggage”, it tells us an important fact.

This fact is that we can halve or double amplitudes and everything will look the same. Whether the PWM signal has a peak to peak amplitude of 1V or 1000V is not important as far as responses go. If a filter has a certain frequency response with a 1V signal, it will have the same response with a 1000V signal.

This property has to be tempered, however, with some real-world common sense. Filters become non-linear if amplifiers are not able to handle the full signal amplitude. Filters also become non-linear if inductors saturate from high current or if capacitors break down from high voltage.

In the following discussion, filters will always be assumed linear. It will be up to you to choose appropriate parts and, in the case of active filters, power supplies and amplifiers.

PWM signals will generally be shown with a 1V amplitude, understanding that the amplitude can be scaled up or down, accordingly.

## F. Step Response of a Filter

One of the keys to choosing a filter for a PWM signal is step response. This turns out to be a more useful starting point than the traditional frequency response concepts.

Every filter HAS a step response. The step response of a two pole filter can be very, very, different from that of a one pole filter, but both have a step response. Lets start by looking at a one pole filter that consists of one resistor and one capacitor. As a side note, all of the examples and illustrations for this exposition are generated using free "LTSpice" (for a copy, go to http://www.linear.com/designtools/software/ltspice.jsp)

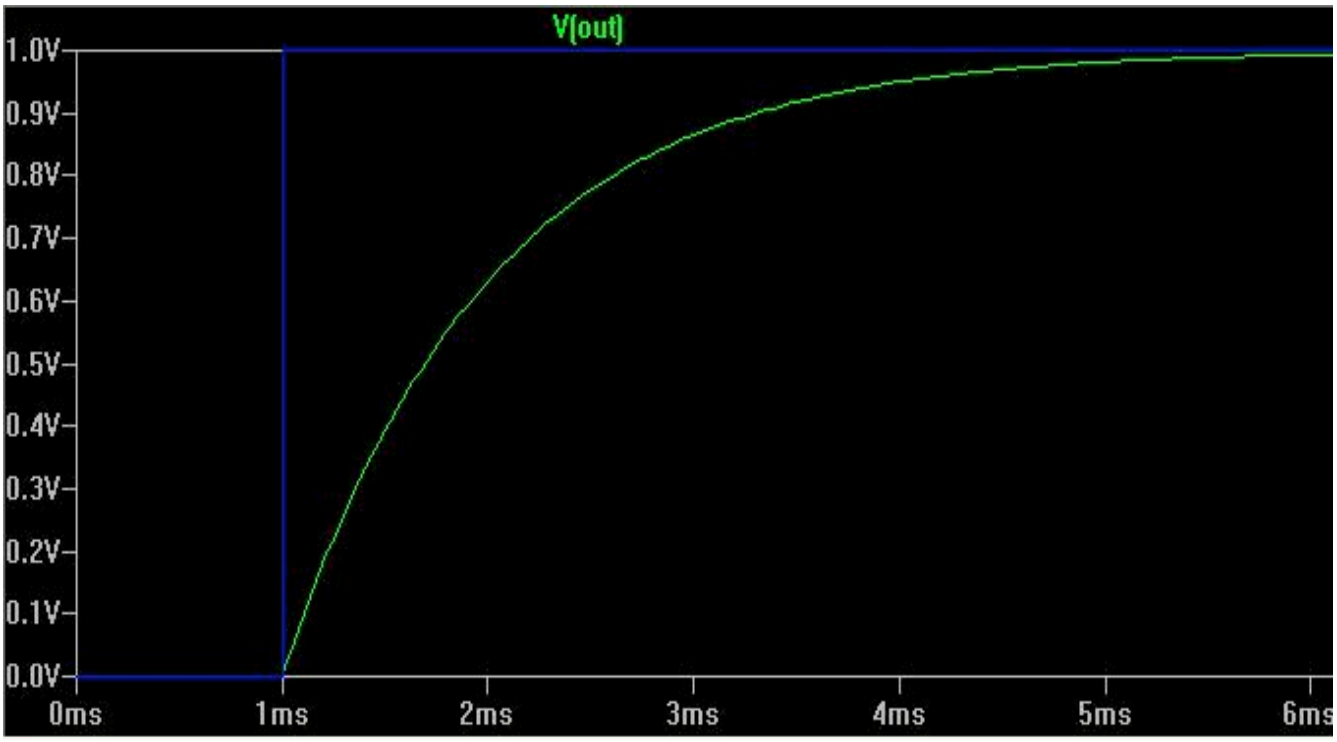


*Figure 3 - Step Response of One Pole Low-Pass Filter*

In this figure, the blue trace is the step input and the green one is the filter output. A single pole low-pass filter has a step response that is described by the following equation:

$$Vout = Vo + Vstep * [1 - e^{**}(-t/\tau)]$$

where "**" denotes "exponentiation", e denotes the base of the natural logarithm (e = 2.718281828), $\tau$ denotes the filter "time constant", t denotes time, Vo is the output voltage at t = 0, and Vstep is the difference between the "final value" and Vo. For a simple RC low-pass filter, $\tau$ = RC.

This time response has a very important property. If Vo is taken to be the output at some new reckoning time and Vstep is the difference between the new Vo and the final value; nothing else changes. It is as if there is no "memory" of what happened before; you can take any point as a new starting point and the response will continue based only on what the value was at that starting point.

Next, lets take a look at the step response with a two pole filter. Here, the step response no longer follows the simple exponential function that was previously used. In fact, depending on the actual values of the inductors, capacitors, resistors, and amplifier gains (if any), the step response can vary from a smooth rise rather like Figure 3 to one that seems to vary quite wildly. See Figure 4 for some of the possibilities.

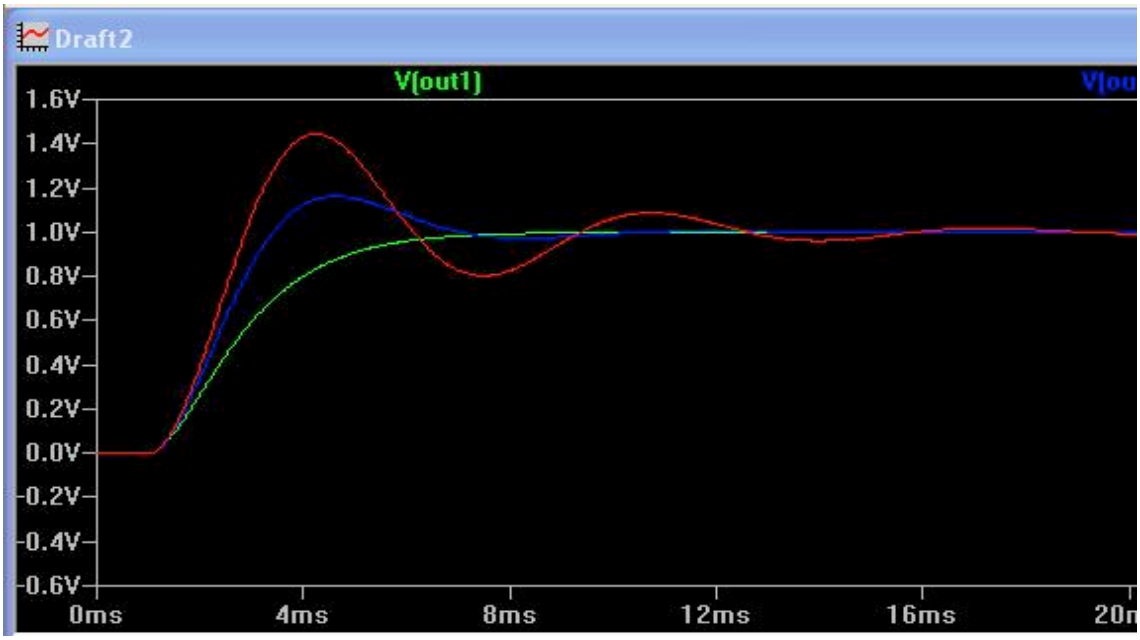


*Figure 4 - Step Responses of Two Pole Low-Pass Filter*

The red and blue traces exhibit features known as "overshoot" and "ringing". The green trace shows neither.

The filter used to create Figure 4 has a series R and L with a shunt C to ground at the output. Components were chosen so that the corner frequency is nearly the same as the circuit used in Figure 3.

A measure that is often used to describe this circuit is "Q". The higher the Q, the more "ringing" there is at the top of the step response. If Q is ½ (green trace), it will be "critically damped", giving the fastest rise without any ringing. Q=1 (blue) and Q=2 (red) are both “under-damped). But the other side of the equation is that the higher the Q, the faster the frequency drops off at high frequencies. We will discuss this in more detail, later.

## G. Pulse Response of a Filter

Next, lets take a look at what happens with narrow pulse as the input to a filter. We will try one case where the output of the filter has had a chance to reach a stable output value before the change happens. Then we will look at what happens when the second edge of the pulse happens before the filter output has had a chance to become stable.

Figure 5 shows the first case, where the pulse (blue trace) is wide enough that the output has approached its final value (that is, reached stead state) quite closely before the second edge occurs. There should be few surprises; the second transition is simply a mirror image of the first.

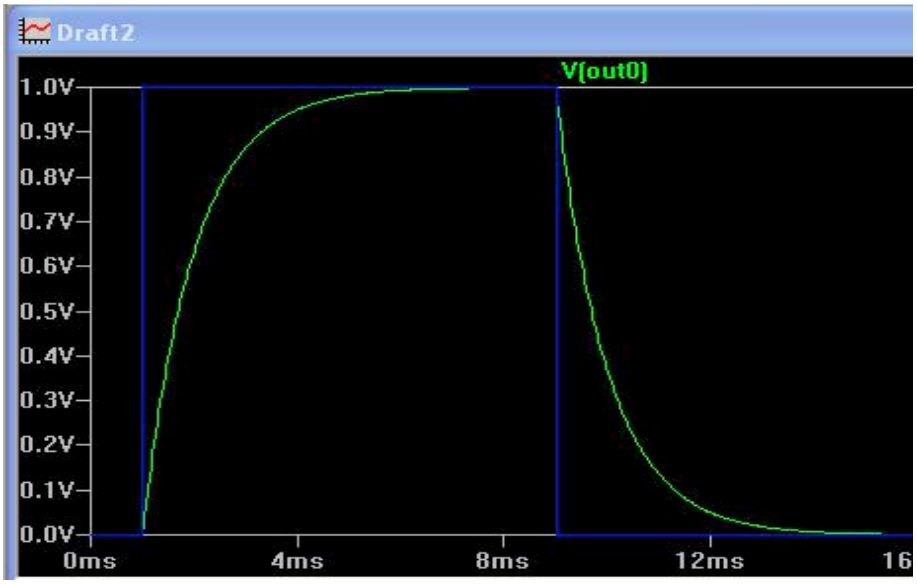


*Figure 5 - One pole Low-pass with Wide Pulse*

The next figure might be a bit more puzzling. In it, the input pulse ends before the output ever has a chance to reach anything close to its final value. However, there is a common theme to these two cases. In both, the downward slope of the output begins with the "initial conditions" that were present at the instant the input changed. In one case, it was from a 1V level and in the second it was near 0.63V. In both cases, the filter output followed the same mathematical function and ended up at 0V.

This behavior is key to how the filter behaves with a PWM signal, which is our next step.

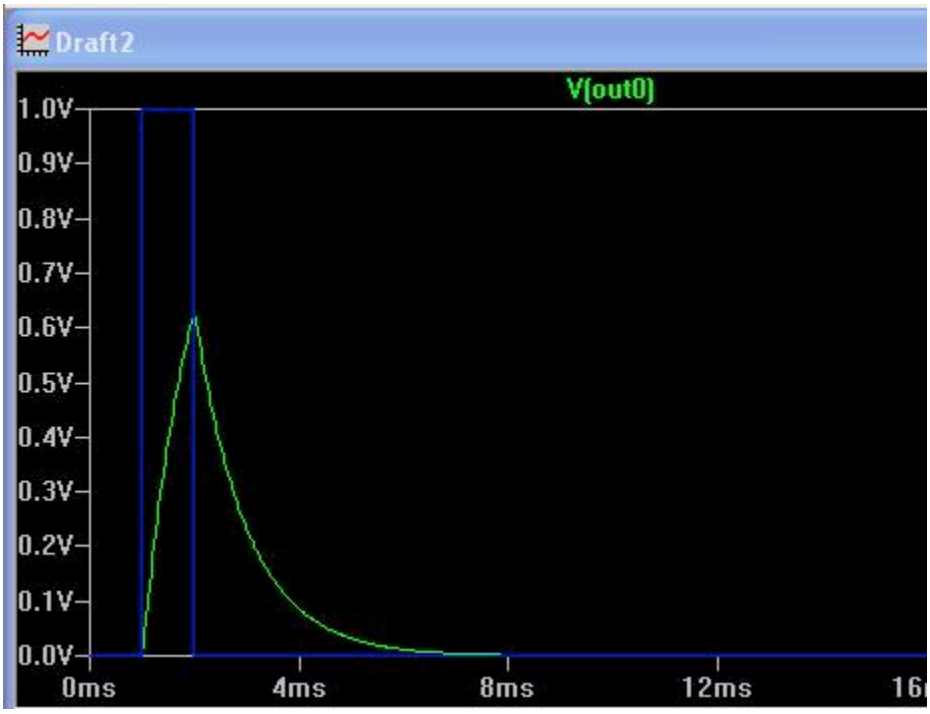


*Figure 6 - One pole Low-pass with Narrow Pulse*

## H. How A Filter Responds to a PWM Signal

Here, we will investigate what happens when there is a whole "train" of pulses, then what happens when the "on" time of the pulse changes while the period is kept constant.

Look back at Figure 6 for a moment and imagine what would happen if a second rising edge happen 1ms after the first falling edge. At that point, the output voltage is around 0.23V (you will have to take my word for it, since it is hard to

read from Figure 6). That second rising input edge should cause the output to begin rising (but from 0.23V, not 0V), and the pattern should continue with each edge. Now, take a look at Figure 7 and see if the behavior is anything like you thought it would be.

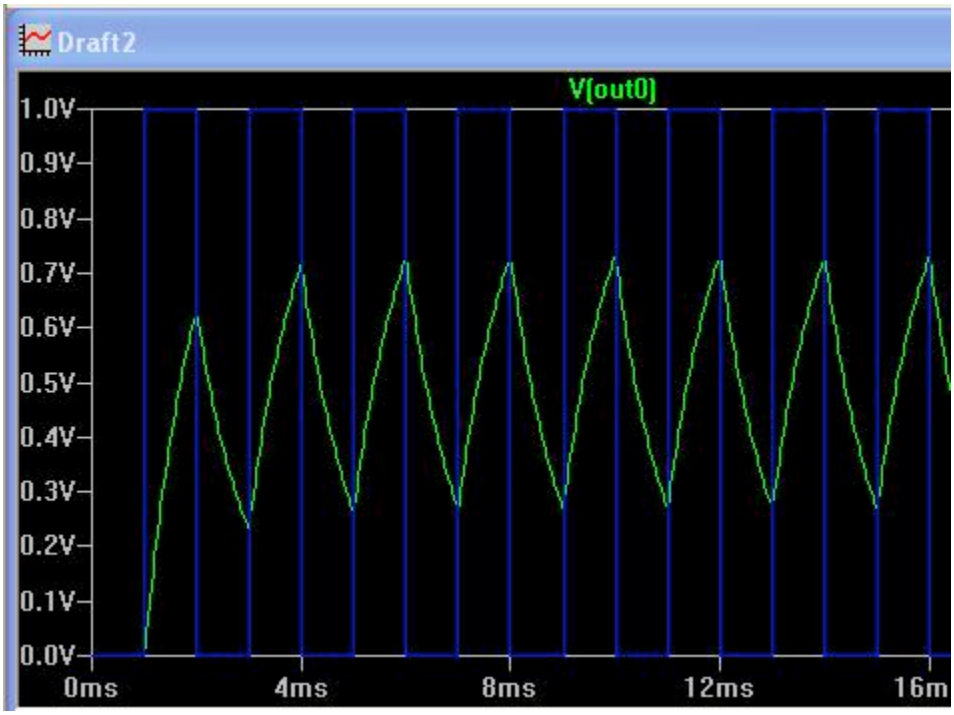


*Figure 7 - One pole Low-pass with Continuous Pulse Train*

Note several things about Figure 7. In particular, where does the filter output end up at? Ignoring the ripple (the triangular up and down at the PWM frequency on the filter output), a careful look will show that it is very close to 0.5V. Now, note that the pulse train logic levels are 0V and 1V and it is on for 1ms and off for 1ms - that is, it has a 2ms period which gives a 50% duty cycle.

Thus, after the filter has had time to "settle", it assumes the value that is the average of the input waveform. Lets check and see if that holds true for other duty cycles. The next figure uses the same pulse train but on for 1.5ms and off for 0.5ms. The total period is unchanged at 2.0ms, but it now has a duty cycle of 1.5ms/2ms = 0.75.

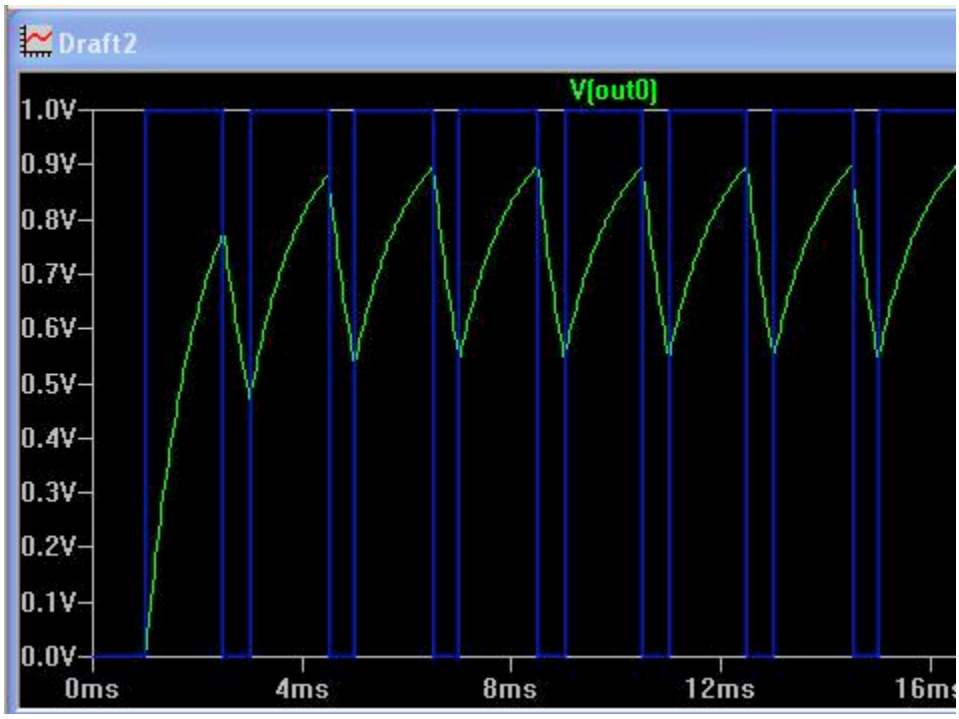


*Figure 8 - One pole Low-pass with Continuous D=0.75 Pulse Train*

And, we have what was probably expected. The output of the filter is about 0.75V. This leads us to the first rule of thumb about PWM filters:

> **Rule of Thumb #1**. Provided that the “PWM frequency is high enough", and provided the filter is given sufficient time to settle, the filter output will be very close to the average of the PWM signal.

Before going on, something needs to be said about the phrase “PWM frequency is high enough”. The meaning that is useful in this report is that the peak-peak amplitude of the ripple is appreciably smaller than the peak-peak amplitude of the input PWM signal. A somewhat equivalent statement is that corner frequency of the filter is lower than the PWM frequency.

Lets now try a little experiment that you might not think to try. We will compare the filter output from Figure 8 to the output of the same filter, but with a simple step input having an amplitude of 0.75V. We will compare them by superimposing the two filter outputs.

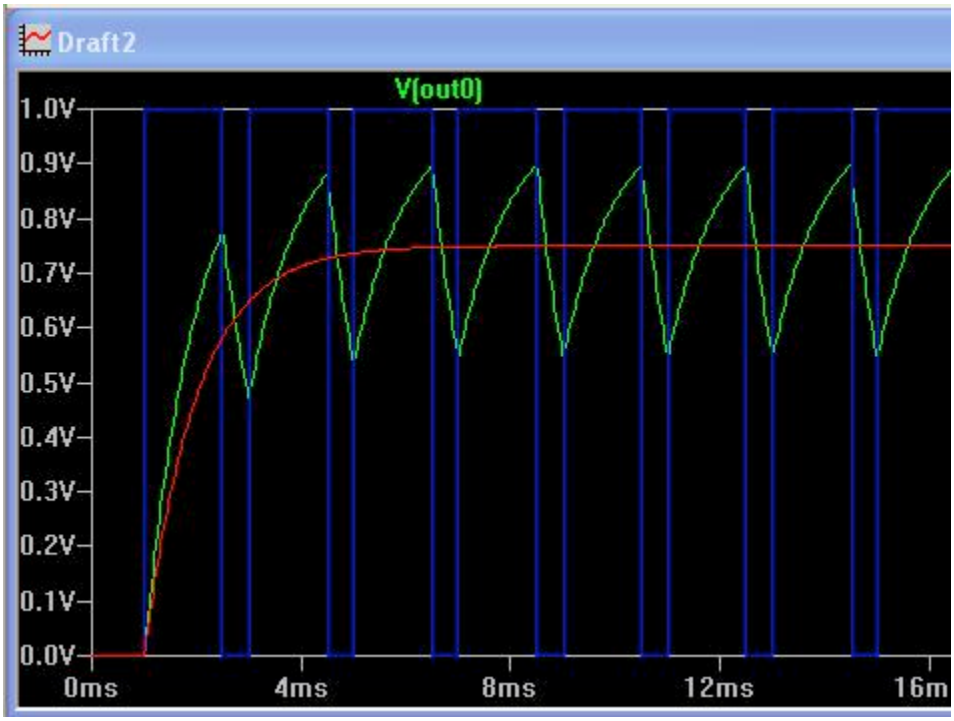


*Figure 9 - One pole Low-pass with Continuous D=0.75 Pulse Train and a second with a 0.75V step.*

Now, take a close look at Figure 9 and compare the two filter outputs. The one driven by the PWM signal has ripple and the other does not - that is to be expected. But, what about their shape and amplitude, ignoring ripple? Right, they are the SAME! And, as a general rule, they will be for any low pass filter with any number of poles. This provides the second rule of thumb:

> **Rule of Thumb #2**. Provided that the PWM frequency is high enough, the startup transient filter output, ignoring ripple, will be exactly the same as if the same filter were driven by a simple step having an amplitude equal to the PWM average.

## I. Modulation Response of a Filter

This section builds on the observations that lead to Rule #2. In particular, lets suppose that we have a PWM signal that has been running with a duty cycle of D1. Lets then suppose that we suddenly change the duty cycle to D2. And, since we had some success from comparing the PWM filter output to the same

filter driven by a simple step, lets try the same comparison. In particular, lets provide a DC voltage level of D1 to the input of the same kind of filter, then suddenly step the input to a voltage level of D2. In our example, which is shown in Figure 10, D1 is set to 0.25 and D2 is set to 0.75. In Figure 10, the PWM signal is shown in blue, the filter output is yellow-green, and the comparison filter output is shown in red.

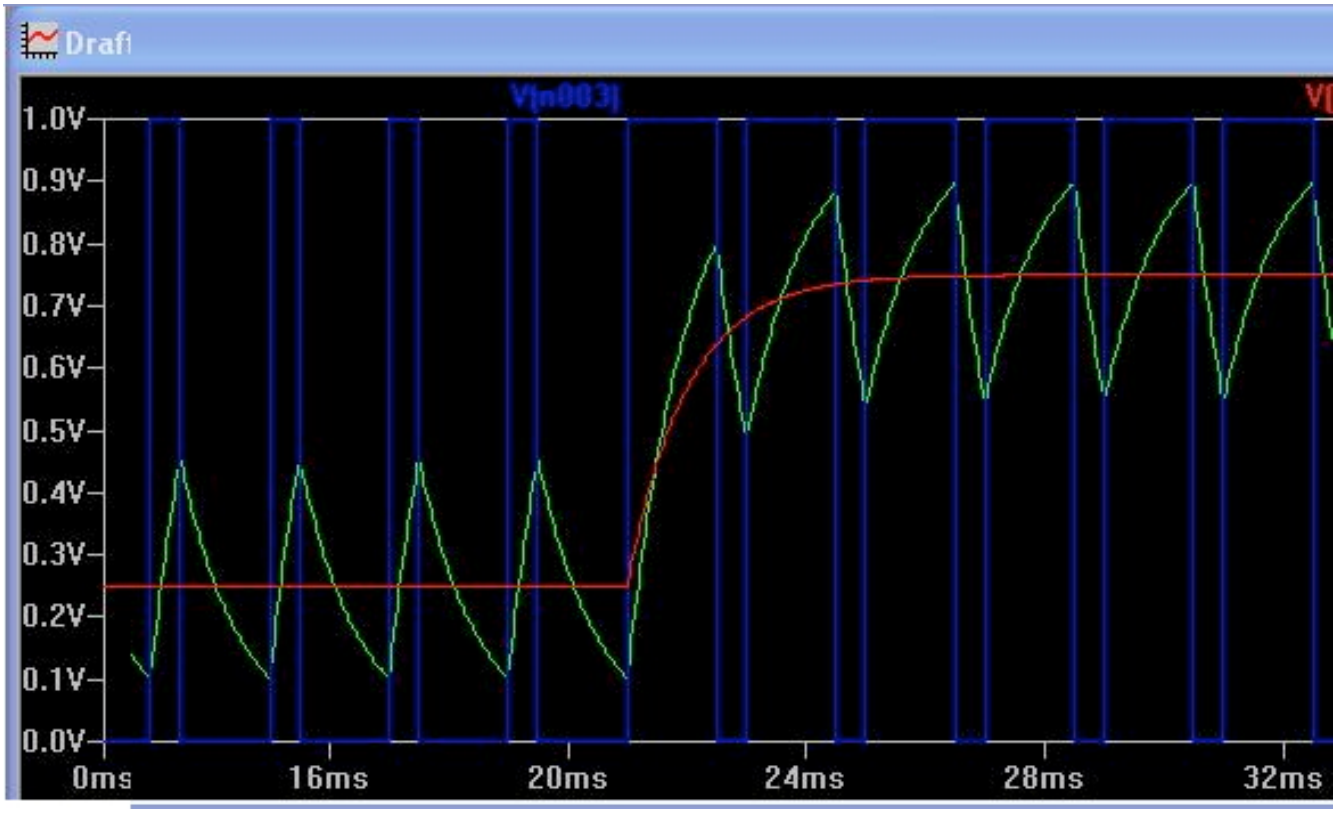


*Figure 10 - One pole Low-pass with PWM Step Change from D=0.25 to D = 0.75 compared to a second filter with an input step from 0.25V to 0.75V.*

Maybe there is a little less surprise this time, but, again, the two filter outputs match (discounting ripple).

This leads us to the third rule of thumb:

> **Rule of Thumb #3**. Provided that the PWM frequency is "high enough", the transient filter output, ignoring ripple, when suddenly changing duty cycle, will be exactly the same as if the same filter

were driven by a simple step having an initial level equal to the PWM initial average and a final level equal to the PWM final average.

From this not so simple rule of thumb, we can take a huge leap. This leap is possible because there is a precise relationship between the frequency response and the step (time) response of any filter. We won't make the effort to derive this result, but it IS derivable.

This leap gives us the following rule of thumb:

**Rule of Thumb #4.** The PWM filter will treat PWM modulation exactly the same as if the modulation (extracted from the PWM signal) were applied directly to the filter.

How are we to interpret this rather remarkable conclusion? Suppose that a PWM signal is modulated so that the duty cycle follows the pattern associated with a 100Hz sine wave. The rule of thumb tells us that the filter will alter the amplitude of the "extracted" 100Hz output in exactly the same way as if a 100Hz sine wave is directly connected to the filter input.

As an example, lets suppose that the filter attenuates 100Hz signals by 0.1db. Then, suppose that our PWM signal is modulated by varying the duty cycle sinusoidally at a 100Hz frequency. Then, we know that the 100Hz filter output, after extracting the 100Hz signal from the PWM waveform, will also be reduced by 0.1db.

Thus, if we know the frequency response of the PWM filter, we can immediately determine what the amplitude of the extracted signal will be at any frequency.

Note that this is TOTALLY independent of the PWM signal and its frequency, again provide that the PWM signal frequency is high enough. This independence leads us to another rule of thumb:

**Rule of Thumb #5.** If the PWM frequency is high enough that the peak-peak ripple is less than the peak-peak amplitude of the PWM input signal, then the only thing that is effected by the PWM frequency is the ripple frequency and ripple amplitude.

In other words, changing the PWM frequency does not, somehow, change frequency responses nor extracted signal amplitudes.

## J. Filter Ripple

It is finally time to begin looking at the relationship between the frequency response of the filter and the ripple amplitude. However, we need to state another rule of thumb which will make this task easier:

> **Rule of Thumb #6.** Ripple amplitude is highest when the duty cycle is 50%. Ripple amplitude is zero at duty cycles of 0 & 1. Ripple amplitude varies smoothly between these values and the value at 50% duty cycle without any peaks or dips.

This rule is independent of the PWM frequency and the frequency response of the filter, so long as the PWM frequency is "sufficiently high". As a result of this rule, we need look only at the 50% case as the one-and-only "worst case" condition.

To proceed from here, we will rely on some information from "Fourier Analysis". It tells us that any periodic signal is made up of series of "harmonics" - that is sines at integer multiple frequencies of the fundamental frequency of the analyzed signal.

For most "well behaved" signals, the harmonics decrease with harmonic number, and this is certainly the case for square and triangle waves, our main focus. However, odd harmonics often follow a different trend than do even harmonics; sometimes all even harmonics are zero.

For a 50% duty cycle square wave with peak-peak amplitude of 1, the Fourier components have an amplitude of $2/n\pi$, where n = 1,3,5,... and are zero for even values of n. The important number is that the first component has an amplitude of $2/\pi = 0.63$. Remember that sine "amplitude" is the peak value. So, the first Fourier component of a 1V peak-peak square wave is a sine with a peak-peak amplitude of 2*0.63 = 1.26.

Why is this important? Any well-chosen low-pass filter will reduce the harmonics of the PWM signal more than the fundamental. If the filter removed ALL of the harmonics and left the fundamental, the ripple would be sinusoidal and would be even larger, in amplitude, than the original PWM signal!

If the filter happens to have an attenuation of A at the fundamental PWM frequency (and, certainly, even more attenuation at higher frequencies), then the first Fourier component will have a peak-peak amplitude of S*A*2/π = S*A*1.26, where S is the peak-peak amplitude of the PWM signal.

Thus, if the PWM signal is 0V to 5V at a frequency of 64KHz, and the filter happens to have an attenuation of 20db (that is, gain = 1/10) at 64KHz, the fundamental Fourier component will have an amplitude of 5V*(0.1)*1.26 = 0.63V peak-peak. The ripple will be AT LEAST this large.

Lets test it with a simulation. We will actually try it with a 5V, 64KHz square wave feeding a filter that has 20db of attenuation at 64KHz. For a single-pole low-pass filter, this translates into a filter with a corner frequency (more about this in a bit) of 6.4KHz and an RC time constant of 24.8us. Check Figure 11 for the results.

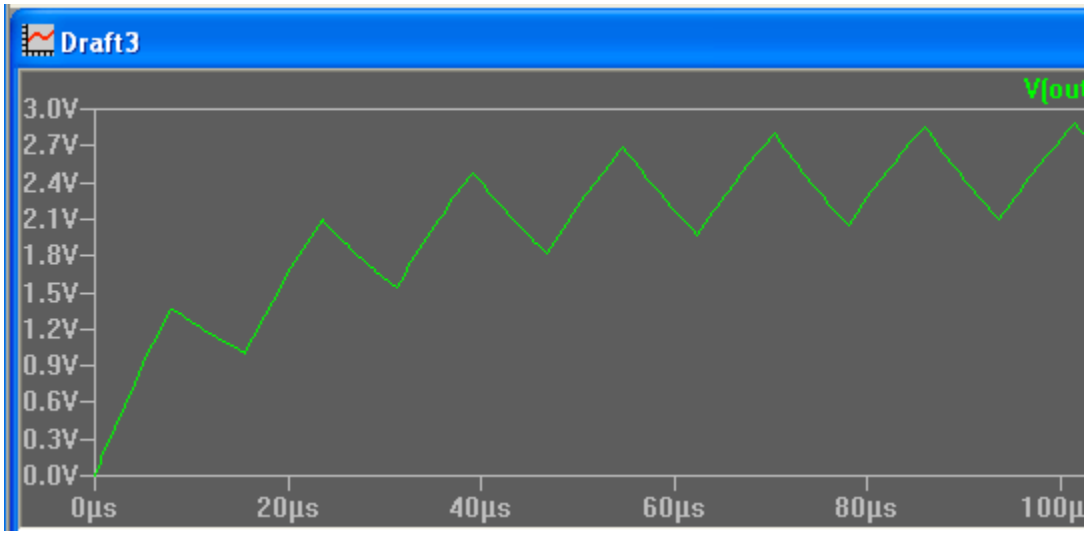


*Figure 11 - One pole Low-pass (Fc=6.4KHz) with 5V, 64KHz Square Wave.*

The last positive peak in Figure 11 is at 2.889V and negative peak just preceding that is 2.131V for a ripple peak-peak amplitude of 0.758V. Our prediction, above, was that the ripple would be at least 0.63V. From one test, it would appear to be accurate as far as it goes.

Our next observation is that the ripple output of the filter, for adequately high PWM frequency, is essentially a triangle. It might be useful, then, to do a similar Fourier analysis on the triangle waveform. That analysis says that the peak-

peak amplitude of the first harmonic sine component of a triangle (with a peak-peak value of 1) has a value of 8/π**2 = 0.81. Thus, if the first order sine component of the triangle IS the attenuated first order sine of the incoming square-wave, then we might expect this triangle to have a peak-peak amplitude of 0.63V/0.81 = 0.78. This is quite close to our simulation value!

Following this logic, we might make the following estimate:

A = filter attenuation at fundamental PWM frequency
S = PWM signal peak-peak amplitude
V = Peak-peak ripple amplitude

$$V \simeq S * A * (4/\pi) / [8/\pi^{**}2]$$

$$V \simeq S * A * [\pi^{**}2/8] * (4/\pi)$$

$$V \simeq S * A * \pi / 2$$

However, this estimate might be a bit high if a filter with much higher attenuation at high frequencies is used. Then, the ripple could be closer to the earlier minimum estimate of

$$V = S * A * 4/\pi$$

We might make this our 7th rule of thumb:

**Rule of Thumb #7.** The maximum ripple amplitude, if the PWM frequency is high enough that the ripple is essentially triangular, is given by $V \simeq S * A * \pi / 2$, where S is the peak-peak amplitude of the PWM signal and A is the attenuation (actually "gain") of the filter at the PWM fundamental frequency. If the filter roll-off at high frequencies is steeper, then the amplitude could be closer to $V = S * A * 4 / \pi$

This is probably most useful if we turn it around, asking how much attenuation is required to give a maximum desired ripple amplitude?

## K. Choosing a Useful FIlter

We have finally reached a point where it is possible to narrow the selection of useful filters. At least as far as PWM filtering is concerned, there is usually a huge selection to choose from. The problem is basically "under-specified" and there may not be any one "right" filter.

There are four specifications that most of us will have to work with. Some, who just want to get a signal out may not be able to pin down all four.

1. PWM repetition frequency may be the easiest spec to determine. In most hardware, you, the designer, set this. If it is created solely by software, it may be pretty low. If it is created solely in hardware, it may be higher. Some devices now offer a PLL clock multiplier to generate a PWM clock that is faster than the system clock. The PWM frequency is often the system clock divided by the full-scale (one period) count. For example, an 8-bit PWM generator clocked at 8MHz could have a repetition frequency of 8MHz/256 = 31.25KHz.

2. Modulation Frequency range should be known, also. If you are creating a 1200Hz and 2200Hz tone pair to implement a Bell-202 modem, the output frequencies will be 1200Hz and 2200Hz. To create telephone-grade voice requires a bandwidth from 300Hz to about 3400Hz. Communications-grade voice uses less upper-end bandwidth.

3. Maximum Ripple Amplitude may be hard to pin down. Many of us will say "as little as possible"; some may say "I don't want any ripple, at all". Unfortunately, starting from a PWM source, zero ripple is NOT an option.

4. Passband Flatness is probably the hardest specification of the four to pin down. For voice, human listeners will be hard pressed to detect an amplitude variation over frequency of even 3db. On the other hand, modem tones may require a high degree of amplitude matching. Many instrumentation applications also require close control of amplitude. If the application is essentially DC, then choose some upper frequency depending on how rapidly your are likely to want to change it (3.3ms settling time corresponds to 100Hz upper frequency limit and 33ms corresponds to 10Hz). You are going to have to determine what this is, depending on your application.

Next, a simple graphical procedure will be described for the purpose of pointing out what the key filter characteristics are for your application.

1. Construct a simple rectangular graph with frequency on the horizontal axis and decibels (db) on the vertical axis. The horizontal axis can be linear or logarithmic, though filter characteristics may look more familiar if logarithmic is used. Extend the horizontal axis, on the right, to at least the PWM repetition frequency and, on the left, down at least to the lower end of your modulation passband. If your lowest frequency is "DC", then extend it down to perhaps 10Hz or 100Hz. On the vertical axis, choose some arbitrary point near the top of the axis as "0db". The vertical axis should span at least 40db. A sample graph to accommodate up to 64KHz PWM frequency is shown in Figure 12; you are welcome to use it.

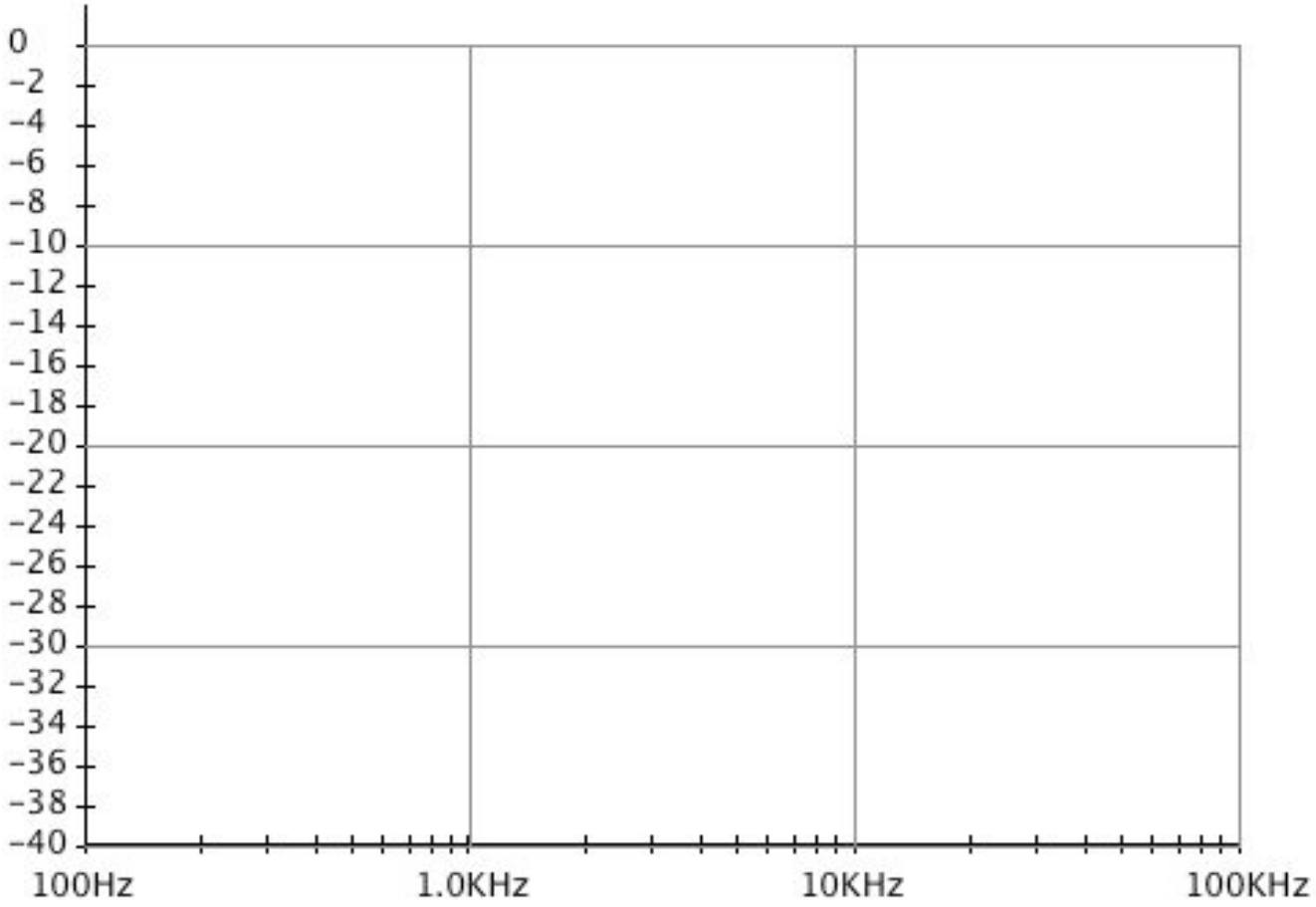


*Figure 12 - Graph Template*

2. Mark the PWM frequency with a vertical line.

3. Figure the required attenuation using the equations in Rule of Thumb #7, solving for a maximum and minimum value for A. Here, where a first order filter is being tested, the ripple will be triangular, the relationship $V \simeq S * A * \pi / 2$ is most appropriate. Convert the attenuation values to decibels using db = 20*Log(A). Mark the vertical PWM frequency line with a mark corresponding to the decibel value. Figure 13 shows the template filled in for a 62KHz PWM signal and 100mV of ripple from a 5V PWM source. The value of A is .0126 which corresponds to about -38db. This design is based on the assumption that later amplifiers will attenuate the ripple more than it is, here.

4. Mark the required output signal response. To do this, mark a horizontal line along the 0db level from the minimum frequency to the maximum frequency. Then, add a second line immediately below the first, spaced down by the number of db of variation that you can tolerate. You should then have a rectangular space where you want your output amplitude to fall. Figure 13 shows what you would have if you try to generate a standard DTMF tone pair (low group is 697Hz to 941Hz and high group is 1209Hz to 1633Hz and the allowed amplitude difference between the low group and the high group is 3db).

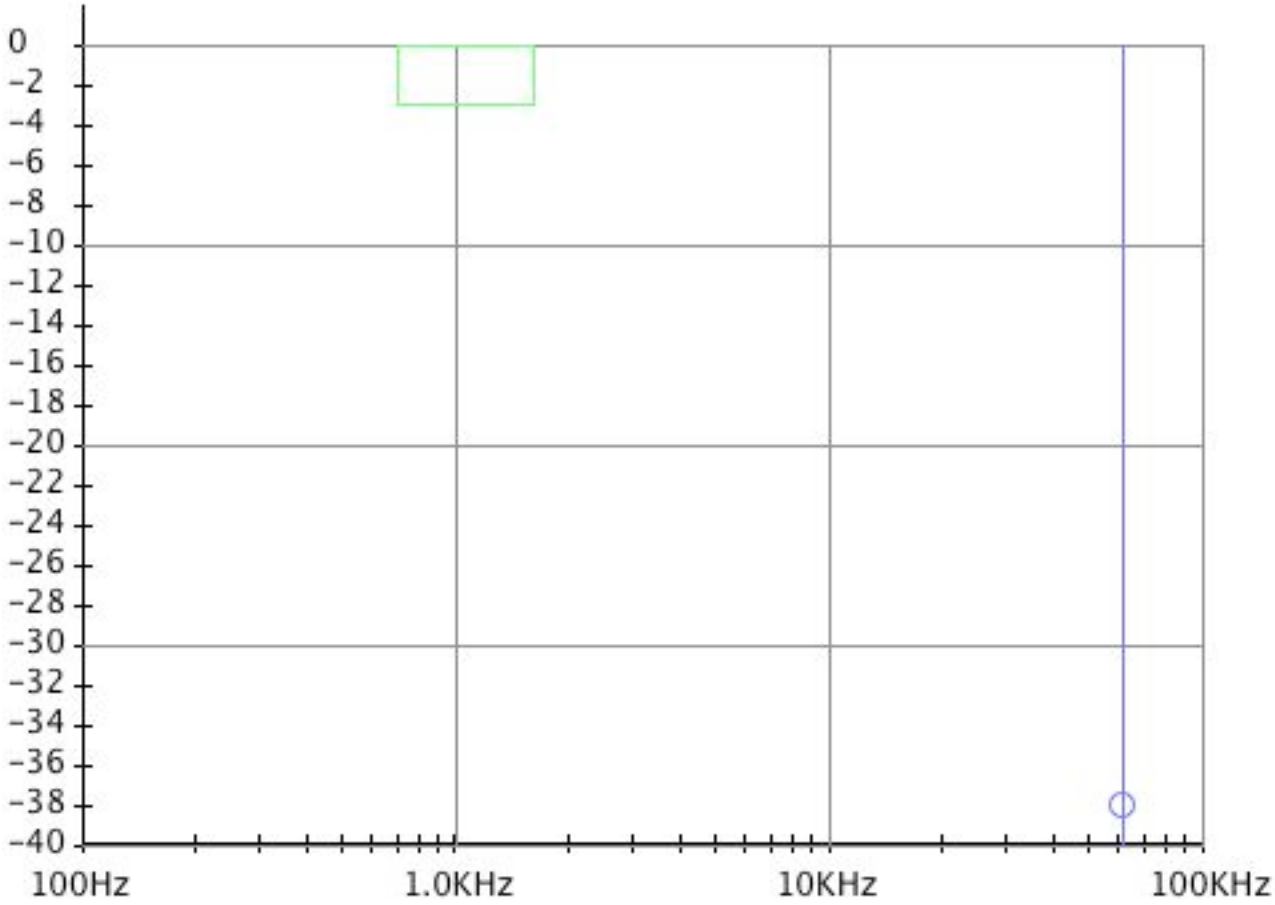


*Figure 13 - Graph with 62KHz PWM signal & DTMF tone specification*

5. Check for first order filter fit. Draw a line with a slope of 20db per decade through the required attenuation point, sloping upward to the left. This line will cross the 0db line at the frequency A*Fpwm (where A is a decimal number, not decibels). In the example case, this frequency will be 0.0126 * 32KHz = 430Hz. Then, continue this line along the 0db line down to the minimum frequency. The result for this example is shown in Figure 14.

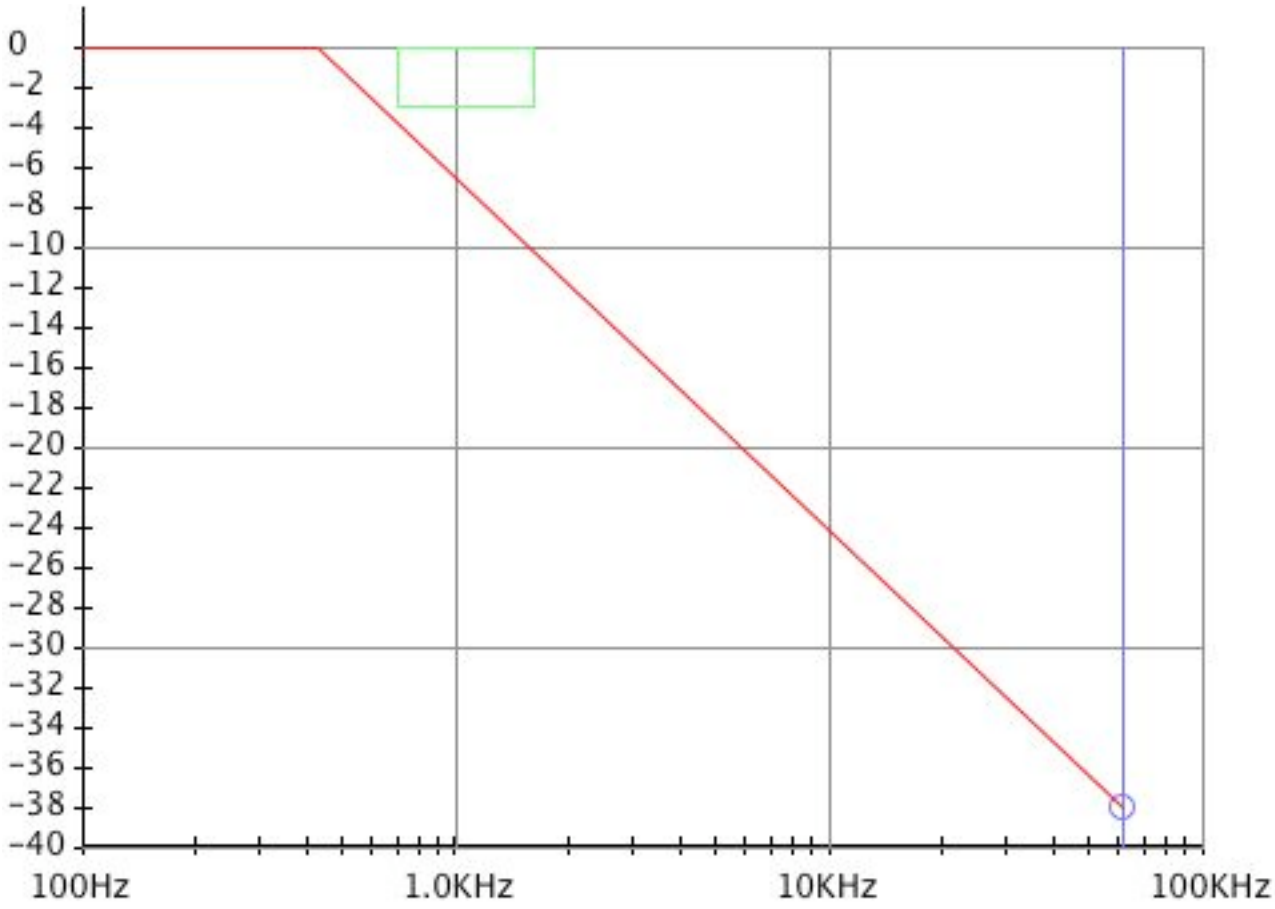


*Figure 14 - Graph with one-pole filter response*

6. Evaluate first order filter fit. There is one question to consider in this step. Is there any frequency for which both the top and bottom edges of the signal amplitude box are above the filter response? If so, the filter has too much attenuation at the signal frequency. In the example, ALL of the signal box is above the filter response, so the specification cannot be met with a single pole filter.

If everything is OK, then the next step is to design the filter which will be discussed in section "L". If everything is NOT ok, then there are some options. One is to allow more ripple. Another is to allow more attenuation, usually frequency dependent, for the signal. A third choice is to go to the next higher order filter with 40db per decade high frequency attenuation. To check, this, draw a new filter response line between the attenuation at Fpwm and the 0db line at Fpwm * sqrt(A). In this case, the frequency is

62KHz * √0.0126 = 6.9KHz. Figure 15 shows this result.

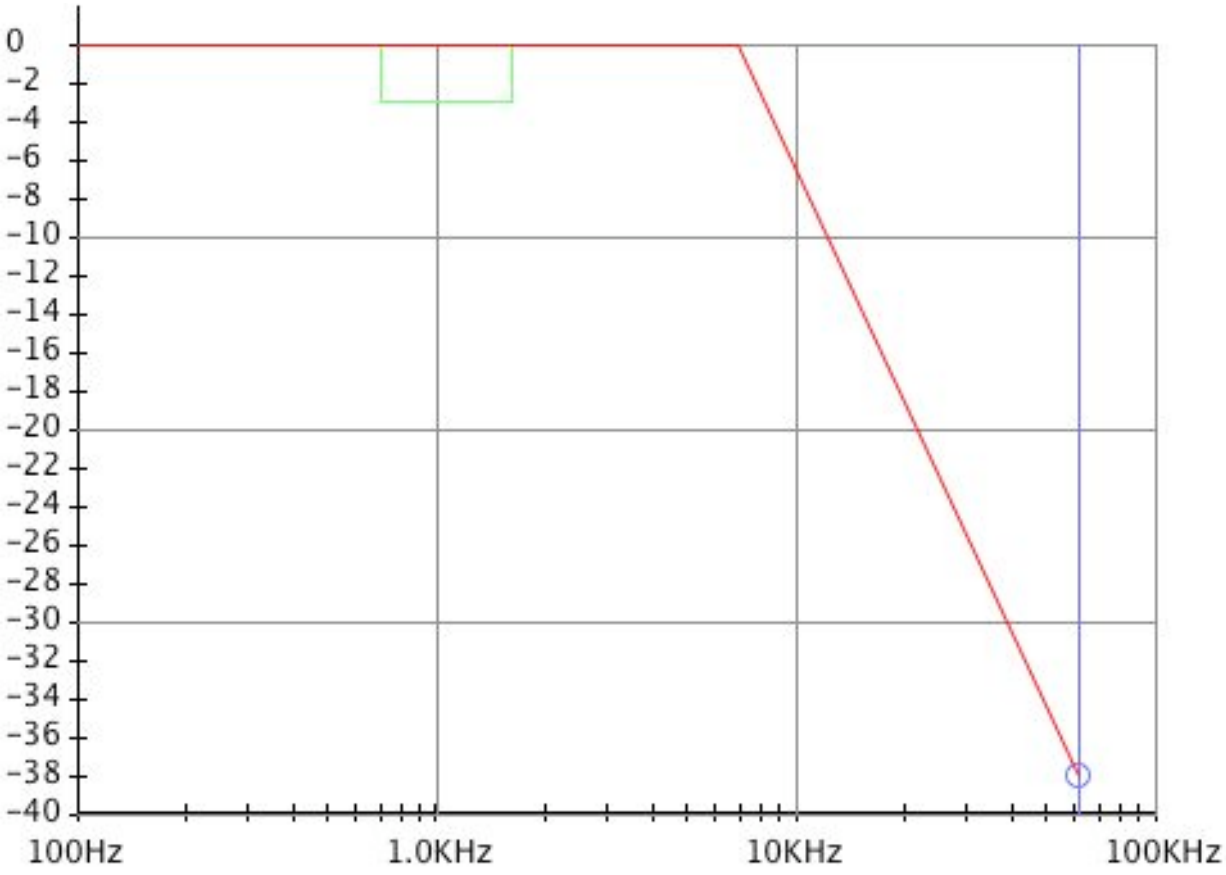


*Figure 15 - Graph with two-pole filter response*

Here, we see that the specifications can easily be met, even ignoring the fact that the real filter does not exactly follow the two straight lines AT the corner frequency. Also note that the filter corner frequency can be shifted to the left, resulting in less ripple.

If a two pole filter is not sufficient, then check with a three pole filter. Here, the filter response is 60db per decade at high frequencies and the point where this line intercepts the 0db line is at Fpwm*$\sqrt[3]{A}$.

Clearly, the closer the PWM frequency and the highest signal frequency are to each other, the more difficult the filter is to create.

7. Signal-Noise Ratio. There is an alternate way of specifying the required ripple attenuation. That is by means of “signal to noise ratio”. It is important to note that, according to the classic definition, ripple is not “noise”. But, it is certainly an undesired component of the output signal. The maximum allowable ripple amplitude is simply the signal amplitude divided by the signal to noise ratio. If you express the signal-noise ratio in db, then the mark you place in Step 3 is down from the 0db line by the minus 1 times the signal-noise ratio.

## L. "Designing" a Useful FIlter

If the previous graphical process has been carried out, successfully, we know how many poles are needed in the filter and we know what the corner frequency is. So, how do you use this to actually create a filter?

1. One Pole. A single-pole filter is certainly the easiest. It can be made from a single resistor and a single capacitor. There are caveats, however, and those will be discussed in a bit. To start with, read off the corner frequency from the graph that you made, above, or compute it from F = A*Fpwm. It does not have to be very precise because most of the components you will deal with have 5% or 10% tolerance ratings. Then compute the resistor and capacitor product, RC, from

$$RC = 1/2\pi F$$

At this point, there is a LOT of latitude. You can, theoretically, choose any combination of R values and C values that have a product given in the previous paragraph. However, practically, things are a bit more limited than that. I prefer to use a minimum R of about 1000 ohms (1.0K) because of the inherent output resistance of most logic devices, which seems to be in the 100 ohm area when the logic is powered from 5V (and a bit higher at lower supply voltages). I also prefer a minimum C of about 220 pf because the stray capacitances on a circuit board can often add up to 10-20 pf. In fact, I generally prefer to use a resistor value near 1.0K with whatever standard capacitor value is available. There is a reason for this.

The larger you make the value of R, the more sensitive the system will be to load resistance. In fact, it will behave as a voltage source with a series resistance (source resistance) of R. Because of this, the signal amplitude will

be reduced to Rload/(R + Rload) of the expected amplitude. There is also a second effect: load resistance will shift the corner frequency UP, making the ripple filtering less effective. One way to reduce both of these effects is to use a buffer amplifier with high input impedance and low output impedance between the filter and the load.

While PWM filters are generally not highly demanding, some care is warranted. In this sort of application, you generally want to avoid electrolytic capacitors. Some ceramic capacitors are good while some others are not, usually because they vary too much with temperature. The preferred types are usually COG or NPO, X7R, X7S, and X5R. The types generally avoided are Y5V, and Z5U. Film capacitors are excellent but costly. Make certain that the capacitor voltage rating is sufficient for the highest signal voltage.

2. Two Pole. A two pole filter is a bit more of a challenge. Any good two pole filter, including the type presumed with the 40db per decade line described in the graphical procedure, needs either a resistor-inductor-capacitor (RLC) combination or it needs an operational amplifier with two capacitors.

While an RLC filter is relatively simple and takes no extra power, the physical inductor size needed for the low frequencies of a PWM filter almost always make them impractical. On top of this, inductors in this size range are expensive, tend to be lower tolerance (often 20% or worse), and are not very "ideal". The latter usually means that special design techniques need to be used to account for the parasitic characteristics. These special techniques are not simple to describe or implement.

Some will tell you "Just put two RC low-pass filters in series". While that is easy, it simply does not provide the performance you are likely to need if a single pole filter is inadequate. On the other hand, if you are only trying to produce a DC output with lower ripple, two RC filters in series may be adequate. You do need to understand, however, that the effective voltage source now has a source resistance equal to the sum of the two resistors in the two filters.

To why two RC filters in series don't have the performance of a well designed two-pole filter, check the next figure. It compares a "proper" two-pole filter with a corner at 2.5KHz (green trace) against two single-pole 2.5KHz RC filters in series (blue trace). The simple fact is that the two single-pole filters have a very "mushy" roll-off (a highly technical term referring to the

sharpness of the corner in the frequency response). This, in turn, means that you cannot get the PWM frequency and the top signal output frequency as close together as you can with a proper two-pole filter. For completeness, the two-pole filter that is plotted is a "Butterworth" type that exhibits no low frequency amplitude "ripple" in the response.

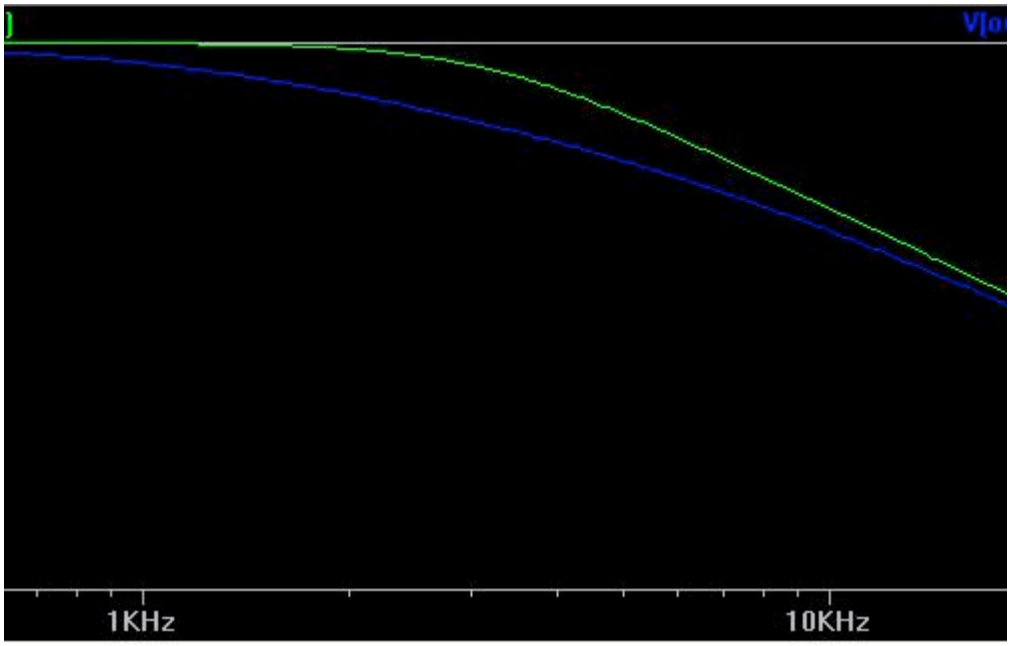


*Figure 16 - Comparing a "good" Two-Pole Filter to a "not-so-good" Two-Pole Filter*

As a matter of fact, the two single-pole filters have been adjusted so that the responses match at high frequencies. The series connection of two one-pole filters results in a lower corner frequency than you would expect from component values.

Having established, hopefully, that there is a major benefit to a good two-pole filter, how does one design such a filter?

The generally preferred two pole filter is an active filter of the "Sallen and Key" type. There is an excellent tutorial available from Texas Instruments (go to http://focus.ti.com/lit/an/sloa024b/sloa024b.pdf) and Wikipedia has a good entry (Sallen–Key topology).

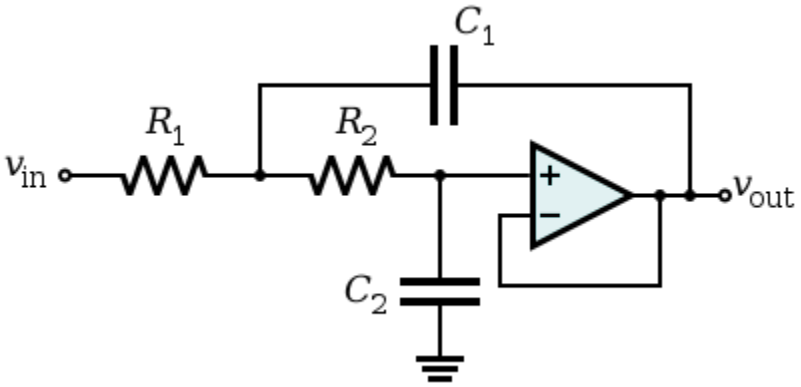


*Figure 17 - Sallen and Key Low Pass Filter*
*(Courtesy Wikipedia)*

For this circuit, the following relationships hold, where Fc is the filter corner frequency and Q is the "quality factor" of the filter:

$$Fc = 1 / 2\pi\sqrt{R1C1R2C2}$$

$$Q = \sqrt{R1C1R2C2} / C2(R1+R2)$$

For a smooth filter response with the "sharpest" possible corner, we want Q = 1/2. This happens to be known as the Two Pole Butterworth response.

A common simplification is to assume that C1 = C2 = C. It is also common to assume that C is chosen to be some readily available value and then pick the resistors to fit.

If we assume that R1 = m R2 and that R2 = R, we have

$$Q = \sqrt{m} / (m + 1)$$

Thus, the resistor ratio is completely specified as the solution to the quadratic equation

$$m^2 + (2 - 1/Q^2)m + 1 = 0$$

Then, for the desired value of Q = 1/2, the equation becomes

$$m^2 - 2m + 1 = 0$$

$$(m - 1)(m + 1) = 0$$

The only "real" solution is m = 1 (or R1 = R2). Then, the corner frequency is given by the simple relationship

$$Fc = 1 / 2\pi RC$$

As a result of this, you pick C to be some convenient and readily available value, then pick a 1% resistor value to suit. All of the comments about "good" capacitors made in the one-pole section apply here, also.

Some special comments are warranted with respect to the operational amplifier (op-amp). It is generally desirable to pick a "rail to rail" type for both input and output. This will allow you to generate a signal that goes within about 50mv of ground and within about 50mv of the positive power supply. That is, a PWM signal that swings between 0V and 5V will have to be limited to a duty cycle no smaller than 50mV/5V = .01 and no greater than (5V-50mv)/5V = 4.95V/5V = 0.99; if the duty cycle is allowed to go all the way to 1 or 0, clipping or limiting will occur in the filter. You CANNOT go all the way to zero with a single supply active filter!

3. 3-Pole. A three pole filter can be readily built using a Sallen and Key two-pole active filter followed by a single pole RC filter. In this application, the filter needs to be designed for a higher Q value of 1. In this case, the relationship between resistor ratio and Q in the Sallen and Key filter using the previous assumptions becomes:

$$m^2 + (2 - 1/Q^2)m + 1 = 0$$

$$m^2 + (2 - 1)m + 1 = 0$$

$$m^2 + m + 1 = 0$$

which has the "nasty" solution of

$$m = (-1 \mp \sqrt{-3}) / 2$$

This tells us that Q=1 is unrealizable with the previous assumptions. Instead, we will have two choices: non-equal capacitors or an op-amp circuit with

more gain than the unity-gain configuration shown in Figure 17.

While an op-amp with gain seems like a simple solution, it raises some real problems. If it is designed with a gain of 2, then the recovered signal will be amplified by 2. Because the op-amp output swing can be no greater than its power supply voltage, the input amplitude would have to be limited to half that. Thus, if the op-amp is powered by the same supply that sets the PWM logic amplitude, then the PWM range would have to be limited to the lower half of the possible values and you loose 50% of the PWM resolution.
So, lets consider the non-equal capacitor option. If we say C1 = n C2 and C2 = C, then the frequency equation becomes:

$$Fc = 1 / 2\pi\sqrt{R1C1R2C2}$$

$$Fc = 1 / 2\pi RC\sqrt{mn}$$

And, for Q:

$$Q = \sqrt{R1C1R2C2} / C2(R1+R2)$$

$$Q = RC\sqrt{mn} / nC(m+1)R$$

$$Q = \sqrt{mn} / n(m+1)$$

How does one approach the problem of using this Q relationship? We can start by converting the Q equation into a second order polynomial in m:

$$m^2 + (2 - 1/Q^2n)m + 1 = 0$$

The quadratic equation tells us that this has real roots only if

$$n \leq 1/4Q^2$$

For the third order Butterworth filter with Q=1, we then find that

$$n \leq 1/4$$

Lets try for n = 1/10 = 0.1 since that allows pairs such as 100nf and 10nf, or 22nf and 2.2nf. Then, our equation for m becomes

$$m^2 + (2 - 10)m + 1 = 0$$

$$m^2 - 8m + 1 = 0$$

$$m = (8 \mp \sqrt{60})/2$$

$$m = 4 \mp \sqrt{15}$$

$$m = 7.87, 0.127$$

Note that 1/7.87 = 0.127 so you end up with the same resistor ratio, only which one is larger and which is smaller. One might give a more useful set of resistor values than the other.

Finally, we have the equation for the corner frequency

$$Fc = 1 / 2\pi RC\sqrt{mn}$$

$$Fc = 1 / 2\pi RC\sqrt{.787} = 1 / 1.77*\pi RC \quad (m=7.87)$$

$$Fc = 1 / 2\pi RC\sqrt{.0127} = 1 / 0.225*\pi RC \quad (m=0.127)$$

Then, pick a convenient value for C (such that 0.1C is larger than 100pf, or so). Choose one of the two values for m, and determine the corresponding value for R based on the desired corner frequency. This specifies all four components in the active filter.

Another convenient value for n is 1/4. This would let you connect 4 capacitors of the value used for C2 in series to make up C1. With this choice,

$$m^2 + (2 - 1/Q^2n)m + 1 = 0$$

$$m^2 - 2m + 1 = 0$$

$$(m - 1)(m + 1) = 0$$

Then, the only useful value for m is m = 1 (that is, R1 = R2).

You will next want to add a single RC filter (series R, shunt C) connected to the op-amp output for which

$$Fc = 1 / 2\pi RC$$

Fc is the SAME as the active part of the filter. For convenience, you might want to choose a value for this C that is the same as one of the two capacitors in the active part of the circuit.

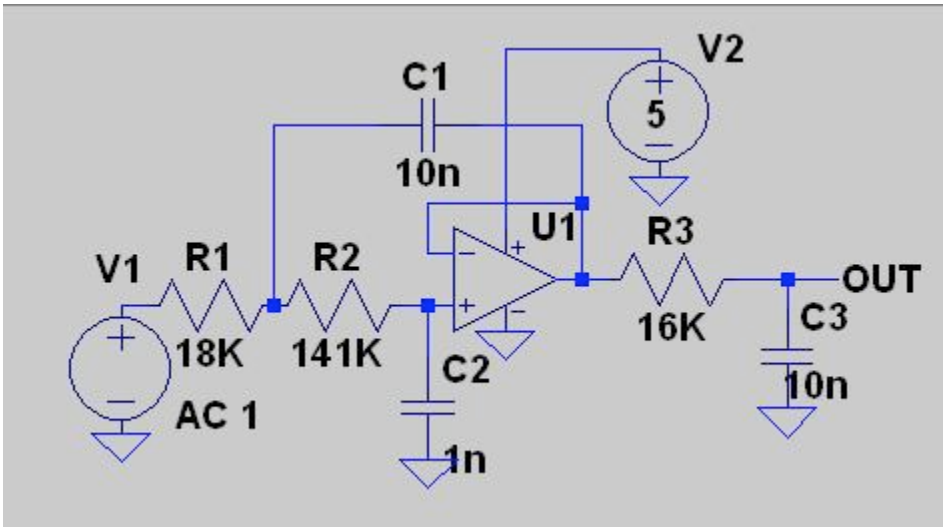


*Figure 18 - Third Order 1KHz Filter*

The circuit to implement a 1KHz third-order filter with m = 0.1 is shown in Figure 18. In that figure, V1 represents the input signal. The filter frequency response is shown in Figure 19.

Note that this filter can be scaled to any other frequency by choosing other capacitors and/or other resistors. ALL resistors have to be scaled by the same factor if resistors are used to shift to a different frequency. Likewise, if capacitors are used to shift to a different frequency, then all have to be scaled by the same factor.

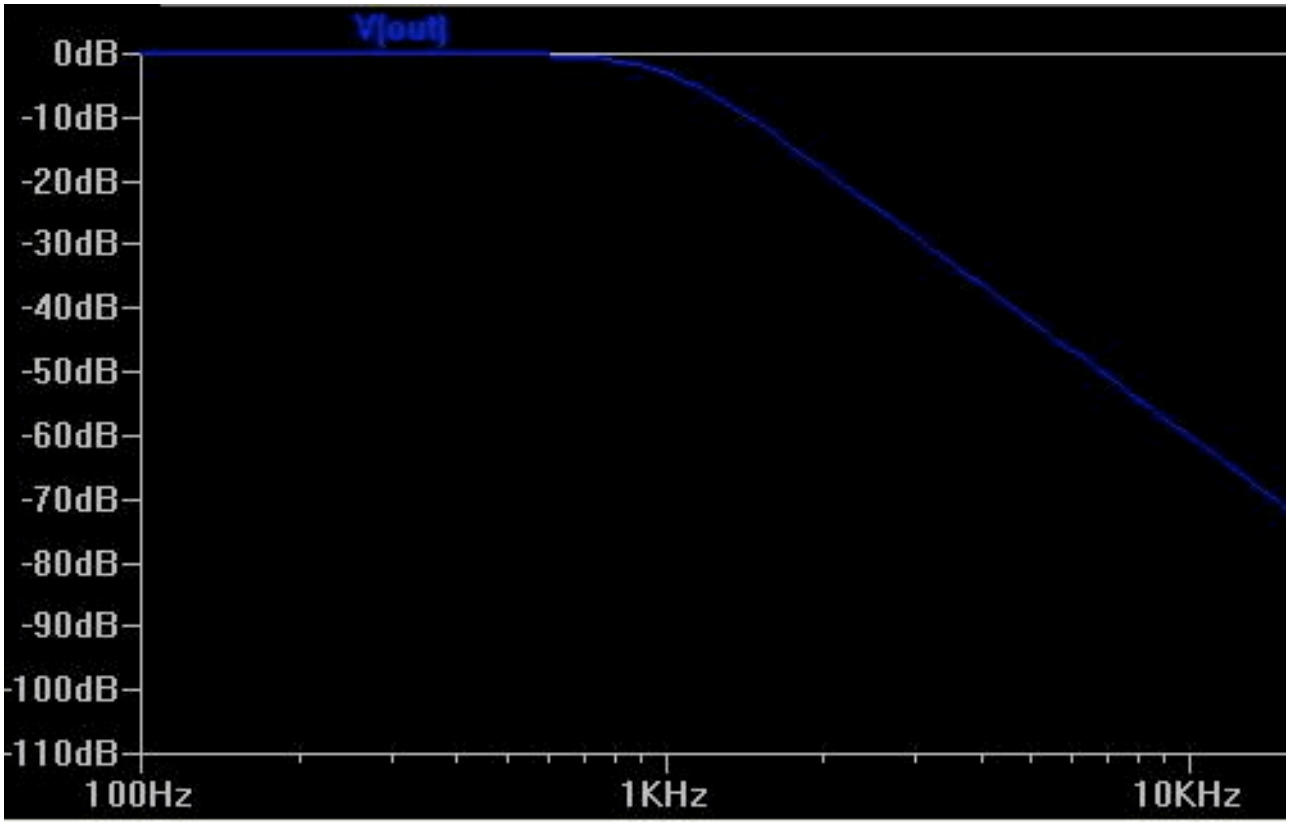


*Figure 19 - Third Order 1KHz Frequency Response*

4. Higher Orders. If you need more than a third order filter to satisfy the system requirements, you have a real challenge. Each step increase in order adds tighter tolerance requirements to the components. Also, you will need more op-amps which increases cost, board area, and power consumption.

At this point, it may become economically attractive to use a "real" digital/analog converter (DAC) simply because of the greatly reduced filtering requirements. In cases where you might have to use a 6th order filter with a PWM "DAC", a real DAC might need only a single RC smoothing filter for the same performance.

If you are determined to continue with PWM, then the author's recommendation is one of the higher order continuous or switched capacitor filters that are available from Linear Technology (go to http://www.linear.com and follow the Signal Conditioning>>Filters>>Low Pass Filters links) and Maxim (go to http://www.maximic.com and follow the Filters (Analog) link).

## M. Non-Butterworth Filters

Through this exposition, the “Butterworth” type filter has been the general focus. There are several reasons for this: they are easy to describe, they are relatively non-critical to build, they provide good performance, and they have no variation in the response below the corner frequency. Further, for a second order filter, there is no ringing or overshoot after a step change in the input since Q = ½.

To be fair, there are other filter formulations. These include Chebyshev, Cauer (also known as elliptic), Bessel, Gausian, and others. Most of these will allow a closer spacing between the highest signal frequency and the PWM frequency. There are, however, several “costs”: generally, the filters are harder to design and require tighter tolerance components, they generally have passband response variation with frequency; this variation may make it harder to meet your signal specifications, and there is more ringing in the step response.

In this report, we really cannot go into the designs of other filter types. However, if you are knowledgeable about filters, there is no reason why you cannot do it, yourself.

## N. Digital Filtering

At this point, someone (usually without much hardware experience) often asks “But, isn't it possible to use digital filtering inside the source hardware instead of an external filter that uses all those parts?”

The answer should be an obvious “NO” and here is why.

The entire reason for filtering a PWM signal is to separate the modulation from the square wave that carries it. So long as your output is a modulated square wave, you need some kind of hardware filter to separate the two.

It IS important, however, to note that there is a possible role for some sort of digital filtering in this process. IF you use a filter that adversely effects your signal, you MIGHT be able to compensate by “correcting” the modulation prior to applying it to the PWM signal. Again, however, the techniques for doing this are far beyond the scope of this report. But, if you are knowledgeable in the art, you might be able to do it.

## O. Conclusion

This short paper has described a set of 7 "Rules of Thumb" which can be used to understand the relationships between PWM repetition frequency, signal frequency, and required filter characteristics. It then describes a design procedure by which the filter order can be determined (up to 3rd order, but extendable to any order) and it describes how practical filters, up to third order, can be constructed.

The author solicits comments about this paper, particularly in matters of clarity, how easy it is to understand or use, and accuracy. Contact may be directed to wagnerj@proaxis.com The author is owner of Oregon Research Electronics and will consult on filtering of PWM signals for a fee.