45th IEEE Midwest Symposium on Circuits and Systems Proceedings (MWSCAS), Tulsa, Oklahoma, August 4-7, 2002
Novel Circuits and Architecture Lecture session WAM1L-214.

A HYBRID INFINITE/FINITE IMPULSE RESPONSE (HIR) FILTER

Michael Weeks and Baoan Wang
Georgia State University
Atlanta, Georgia 30303
e-mail: mweeks@cs.gsu.edu

Abstract

Filters are used to implement signal transforms, and filter design is a very important part of the signal processing field. There are currently 2 types of filters available: the finite impulse response (FIR) filter, and the infinite impulse response (IIR) filter. While FIR filters have desirable traits such as stability (an absence of poles), IIR filters are able to generate outputs more efficiently. This paper combines the two types of filters by introducing a zero-counter, to make a hybrid impulse response (HIR) filter with desirable traits from both FIR and IIR.

This work is protected by a provisional patent.

1. Introduction

Filtering techniques are widely used in digital signal processing as well as in daily life, for example, a noisy audio signal can be filtered to remove the noise that is above or below the frequencies of interest, through a bandpass filter. This can be implemented on a digital signal with convolution.

There are two types of filters in common use: the finite impulse response (FIR) filter, and the infinite impulse response filter (IIR). The FIR filter is stable, while the IIR filter is more efficient, since it uses feedback. This implies that the IIR filter has a longer-lasting effect on a signal, but also that there is no cut-off point where it stops responding. In this paper, we propose an improvement on the IIR filter, where it selectively uses feedback. As a result, the filter we propose is neither an IIR nor an FIR filter, but something in between. Therefore, we call it a hybrid impulse response filter, or HIR.

The next section explains the FIR and IIR filters in more detail. The third section presents the HIR filter, and how it relates to the other filter types. The simulation and results appear in the fourth section, and the fifth section concludes this paper.

2. FIR and IIR Filters

The basic filter equation is given below, where X[n] are the input samples, and Y[n] are the output samples. The b_k values specify the filter coefficients, and can be altered to give different characteristics to the filter.

$$Y[n] = \sum_{k=0}^{M} b_k X[n-k]$$

This type of filter is called a Finite Impulse Response (FIR) filter [1], so named because an impulse provided to the filter will generate only a finite amount of non-zero results. This filter has many nice properties, shown in the following transfer function (a characterization of how the filter will respond).

$$H(z) = \sum_{k=0}^{M} b_k e^{-jkw}$$

This transfer function indicates that the FIR filter may have zeroes, but no poles. This means that its output will be zero under certain circumstances, but will never give an infinite response. Therefore, it is stable, and a desirable type of filter to use.

A second type of filter also exists, the Infinite Impulse Response (IIR) filter [1]. This filter uses feedback, allowing it to use previously computed results to influence the current output. The result is that it may be able to give the same output as the FIR filter, but use fewer filter coefficients, resulting in a more efficient implementation. That is, the size of the processing unit can be made smaller (translating directly into a cost savings for manufacturer as well as consumer), or faster. The following equation gives the output for IIR filters. Notice that the current output is partially based upon the previous outputs, i.e. Y[] appears on both sides of the equation.

$$Y[n] = \sum_{k=0}^{M} b_k X[n-k] + \sum_{k=0}^{N} a_k Y[n-k]$$

The problem with IIR filters is that they can generate an infinite response based on certain input conditions. The transfer function reveals why this is the case:

$$H(z) = \frac{\sum_{k=0}^{M} b_k e^{-jk\omega}}{(1 - \sum_{k=0}^{N} a_k e^{-jk\omega})}$$

The filter coefficients a_k can be positive or negative. They are assumed to be negative to attempt to make a stable filter. If the feedback part of the filter results in a value of 1, then the denominator becomes 0, resulting in an infinite response, called a pole. This is a problem with IIR filters.

3. Hybrid Infinite/Finite Impulse Response Filter

The problem with the FIR filter is that it is less efficient compared to the IIR filter. The problem with the IIR filter is that its output can approach infinity. To reap the benefits of both FIR and IIR filters, a combination of the two types of filters can be made. This is the Hybrid (Finite/Infinite) Impulse Response filter [2], or HIR for short. To combine these filters, a "zero counter" circuit will be introduced, figure 1. It counts the number of zero values in the input stream, and when it reaches the maximum amount, set by the filter designer, it causes all outputs to become zero by resetting all registers within the device. When non-zero input values arrive, they will "re-start" the filter by propagating through the memory registers. The filter architecture is shown in figure 2. It is much like the IIR filter, except for the zero-counter and memory clearing capabilities.

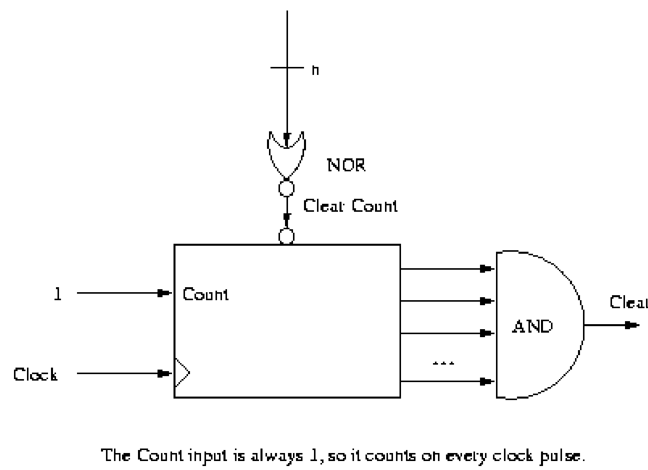

Figure 1 - the Zero Counter

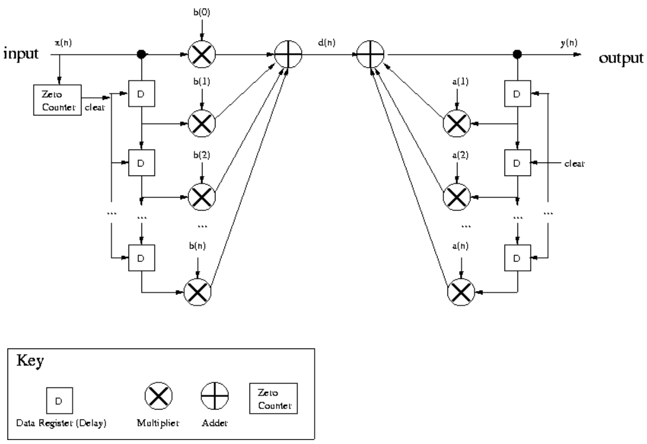

Figure 2 - the HIR filter[1]

4. Simulation of the HIR filter

To simulate the HIR filter, we add another feedback mechanism. Whenever the output signal gets larger in magnitude, it may be approaching infinity. Therefore, anytime the output grows and the input remains constant, the zero counter increments. On the other hand, when the output is not approaching infinity, the zero counter is reset to 0. We then simulate the proposed HIR filter by processing a short period of sound signal (Figure 3A and 3B). An arbitrary filter is selected whose feed forward coefficients are:

$$b_k = [\ 0.25, 0.5, 0.25\ ]$$

and feedback coefficients are:

$$a_k = [\ 1.194, -1.436\ ]$$

First, figure 3C shows that the filtered sound signal looks much like the original. In fact, it is a smoothed version of it, since the feed-forward coefficients used in the FIR filter correspond to a low-pass filter. Figure 3E shows the time-domain impulse response of an IIR filter with these coefficients. The problem shown here is that the filter's output gets extremely large, in the range of 10^6. Finally, figure 3F shows the IIR's frequency response.

[1] Compare to the IIR filter in [1], page 222, from which this figure is adapted.

How the HIR filter responds is shown in figure 4. First, we set the zero counter to be very low, so that all the feedback values (y[n-k], k>0) are cleared to zero, making

$$y[n] = d[n]$$

i.e. effectively disabling the right-hand-side of figure 2. This simulation (Figure 4A and 4B) will therefore act like the FIR filter (Figure 3C and 3D), allowing us to verify the simulation. The results were equivalent to the FIR filter, as expected. Next, by setting the zero-count very high (i.e. as high as the number of data samples), the simulation (Figure 4C and 4D), should act like the IIR filter (Figure 3E and 3F), since the zero-count circuit will never get the chance to clear the filter's memory. In this way, the simulation is compared to IIR filters as well. Finally, the zero-count can be set to an arbitrary value between the low and high, and the true response of the proposed filter will become apparent (Figure 4E and 4F). The actual number that the zero-count should be set at depends on the desired effects that the filter should have. In other words, raising or lowering it determines how much like an FIR or IIR filter it behaves. As figure 4E shows, the HIR filter can give a reasonable response to an input, without poles (compare to figure 3C). Note that the filter's response shown in figure 4E is not meant to provide the same smoothing function that is shown in 3C, since the HIR filter reduces to an FIR filter when the a_k values are set to zero.

The simulation results in figures 3 and 4 show that the HIR filter can act as an FIR filter (figure 3C versus 4A, and 3D versus 4B), or as an IIR filter (Figure 3E versus 4C, and 3F versus 4D). Also, it can give a response that is neither FIR nor IIR (Figure 4E and 4F).

45th IEEE Midwest Symposium on Circuits and Systems Proceedings (MWSCAS), Tulsa, Oklahoma, August 4-7, 2002
Novel Circuits and Architecture Lecture session WAM1L-214.

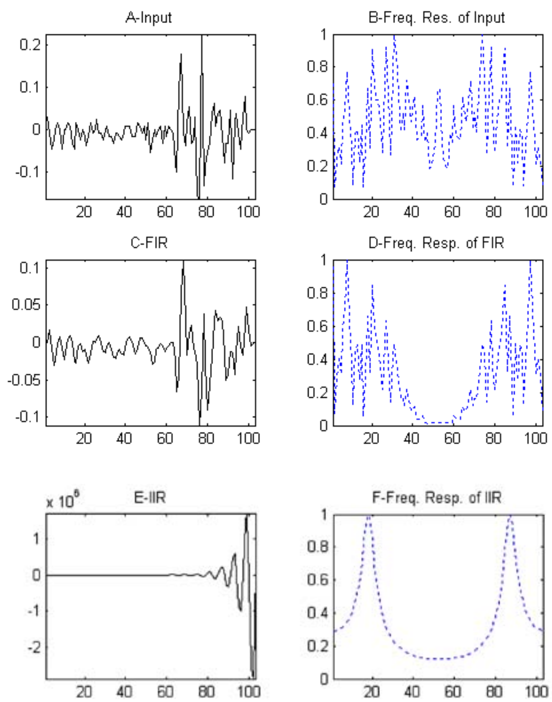

Figure 3 – Results of an FIR and IIR filter

A: Original Signal
B: Frequency response of the original signal
C: Time-domain impulse response of FIR filter
D: Frequency response of the FIR filter
E: Time-domain impulse response of IIR filter
F: Frequency response of the IIR filter

5. Conclusions

In this paper, a novel filtering technique, the Hybrid Impulse Response filter, is introduced. We examine its properties by simulation, and show that it behaves like an FIR filter in certain circumstances, and like an IIR filter in others. However, questions remain to be answered, such as how this new filter could compare to applications performed with FIR filters in image processing [3], noise-tolerant (lower voltage) DSP [4], and video processing. It is likely that the new filter can use less precision among the filter coefficients to achieve the same results as an FIR filter, resulting in smaller/cheaper processors.

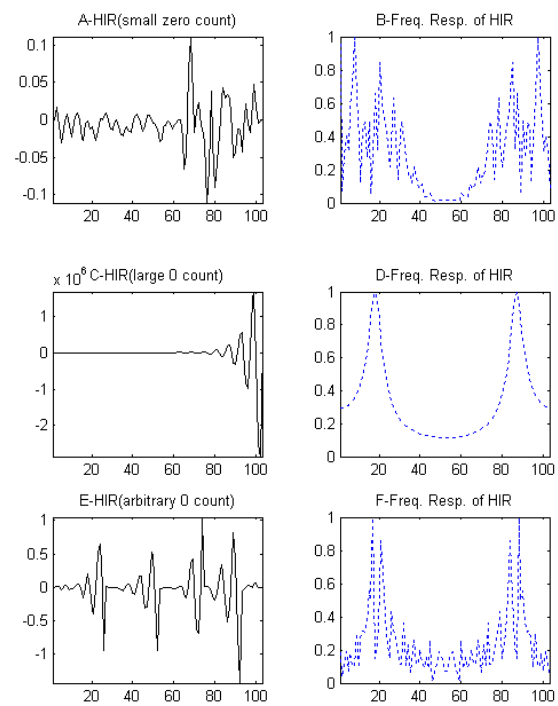

Figure 4 – Simulation results of our HIR filter

A: Time-domain impulse response of HIR filter when the zero-count is set low
B: Frequency response of HIR filter when the zero-count is set low
C: Time-domain impulse response of HIR filter when the zero-count is set high
D: Frequency response of HIR filter when the zero-count is set high
E: Time-domain impulse response of HIR filter when the zero-count is set at an arbitrary value
F: Frequency response of HIR filter when the zero-count is set at an arbitrary value

References:

[1] Richard Lyons, *Understanding Digital Signal Processing*, Addison-Wesley, 1997.
[2] Michael Weeks and Baoan Wang, "The Hybrid FIR/IIR Filter," poster, *State of Georgia's Yamacraw-Project Conference*, April 17, 2002.
[3] Michael Weeks, Jimmy Limqueco, Magdy Bayoumi, "On Block Architectures for Discrete Wavelet Transform," *32nd Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, November 1-4, 1998.
[4] Rajamohana Hegde and Naresh R. Shanbhag, "Soft Digital Signal Processing," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Volume 9, Number 6, December 2001.