A Programmable Power-Efficient Decimation Filter for Software Radios

Emad N. Farag[1] Ran-Hong Yan[2] Mohamed I. Elmasry[1]

[1]VLSI Research Group, University of Waterloo, Waterloo, Ontario, Canada.

[2]Bell Labs, Lucent Technologies, Holmdel, New Jersey, USA.

ABSTRACT

Several high-level low-power design techniques have been incorporated in the design of a decimation filter for software radio. These include; operation minimization, multiplier elimination and block deactivation. Analysis and simulation results indicate that these techniques can achieve a 4 times reduction in power dissipation. An interleaved multiplier-accumulator array is used in the lowpass filter. The decimation filter designed has a programmable resolution, that varies from 12 to 20 bits. The entire decimation filter has been designed in a 3.3 Volt $0.5\mu m$ CMOS technology.

1 INTRODUCTION

In the last few years, design for low-power has become an important issue [1]. With the widespread use of portable equipments whether portable computers, cellular phones, pagers, etc, and with the ever increasing consumer demand for new services and better quality, there is need to further reduce the power dissipation.

Software radio [2] is emerging as a radio architecture with increased flexibility and programmability. In software radio, the received/transmitted radio signal is digitized as electrically close to the antenna as possible. The signal processing is done digitally after that, using a general purpose programmable hardware. Performing the radio, IF and baseband functions in a programmable digital hardware increases the flexibility of the transceiver. However, the price paid for this increased flexibility is greater power dissipation.

Software radios require high-speed high-resolution A/D and D/A converters [3]. In this paper, we examine the design of a resolution programmable decimation filter. The resolution can vary from 12 to 20 bits. To reduce the power dissipation several high-level low-power techniques have been applied to the design of a decimation filter used in the analog-to-digital converter. The low-power techniques considered include operation minimization, datapath width optimization, multiplier elimination, and block deactivation.

2 DIGITAL RECEIVER ARCHITECTURE

Figure 1 shows a typical receiver where the digitization is done after the first IF (Intermediate Frequency) stage [4]. The bottle neck of this architecture is the A/D (analog-to-digital) converter. Not only does this A/D operate at a high speed (in the MHz), but it also requires high resolution as well (12 – 20 bits).

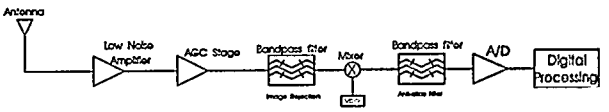

Figure 1: Digital IF receiver architecture

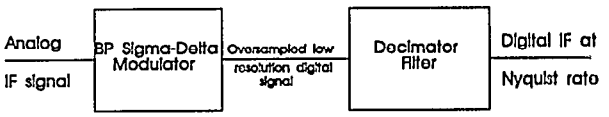

Figure 2: Bandpass Sigma-Delta A/D converter

To achieve a resolution of up to 20 bits a bandpass Sigma-Delta A/D converter [5] is used. The bandpass Sigma-Delta A/D converter incorporates a decimation filter [6] which digitally transforms a low-resolution oversampled signal into a high-resolution Nyquist-rate sampled signal. Figure 2 shows the block diagram of a bandpass Sigma-Delta A/D converter.

The resolution requirement of the A/D and hence the resolution requirement of the decimation filter varies according to the strength of the received signal as well as the background noise and interference. Simulation results, for a system based on the DAMPS standard, indicate that for a digital receiver digitizing an IF signal of bandwidth 0.96 MHz (32 TDMA channels), the maximum required dynamic range for the A/D is 20 bits. Simulation results also indicate that when the input signal is strong enough a dynamic range of 12 bits is sufficient.

The reason for having an A/D with variable resolution is to save power when the lower resolution is sufficient. This leads to the concept of Automatic Resolution Control (ARC) where the resolution of the A/D is varied as opposed to AGC where the gain of the amplifier is controlled by the level of the input signal. In ARC, the required resolution of the A/D converter is determined by the digital stages following it.

3 THE DECIMATION FILTER ARCHITECTURE

The decimation filter (Figure 3) consists of two parts; the Sinc decimator and the lowpass decimation filter (LPDF). The Sinc decimator is characterized by its simple structure, requiring only addition operations which makes it a power efficient structure. The order of the Sinc used is given by [6]:

$$\text{Order of Sinc} = \frac{\text{Order of Bandpass } \Sigma - \Delta}{2} + 1 \qquad (1)$$

For a Sinc filter with order N and a decimation factor $M = 2^m$, the transfer function (before down sampling) is given by:

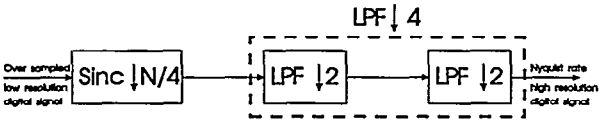

Figure 3: The decimation filter

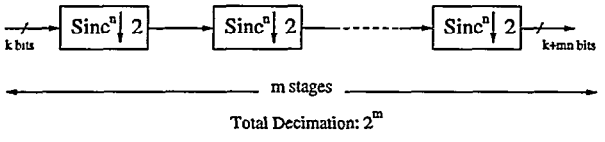

Figure 4: First architecture for a Sinc decimator filter.

$$H(e^{jw}) = (1 + e^{-jw} + e^{-2jw} + \ldots + e^{-(M-1)jw})^N \quad (2)$$

Due to the variable-resolution requirement, the order of the Sinc, N, and the decimation factor, M, can vary. N is designed to be 3 or 4 for a fourth order or a sixth order bandpass $\Sigma - \Delta$ modulator, respectively. M is designed to be 8, 16 or 32.

Due to its gradual transition from the passband to the stopband, the Sinc filter can't be used in the entire decimation process. The last stage of decimation is done using a LPF, which does decimation by a factor of four [7]. The LPF is built as a two stage LPF each doing decimation by a factor of two. Due to the variable-resolution requirement of the A/D, this filter is designed to operate with variable resolution and hence reduce power dissipation when operating at the lower resolution.

4 POWER EFFICIENT SINC ARCHITECTURE

Figures 4 and 5 show two possible architectures for the Sinc decimator. The first architecture is a cascade of m stages, each stage is an nth order Sinc filter that does decimation by a factor of 2. Each Sinc stage has a higher resolution than the preceding stage, but it operates at a lower speed. The second architecture consists of a cascade of n integrators followed by a 2^m down-sampler followed by n differentiators. The integrators operate at high speed and have a high resolution. Hence, the second architecture is less computational efficient than the first one, because it has more redundant computations.

Table 1 gives the number of addition operations required to generate a single output for each implementation. This number is proportional to the computational power dissipation. The first architecture minimizes redundant computations. It requires 3 – 5 times less addition operations than the second architecture.

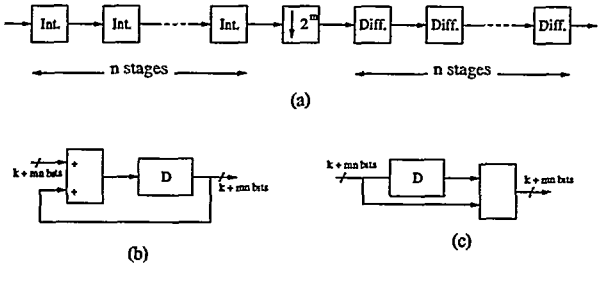

Figure 5: Second architecture for a Sinc decimator filter. (a) Block diagram. (b) Integrator stage. (c) Differentiator stage.

Table 1: Number of additions per output for the Sinc decimators of Figures 4 and 5.

Sinc Filter	Order	Number of additions per output
Figure 4	3	$(4k+14)(2^m-1)-12m$
Figure 4	4	$(5k+24)(2^m-1)-20m$
Figure 5	n	$(2^m+1)(k+nm)n$

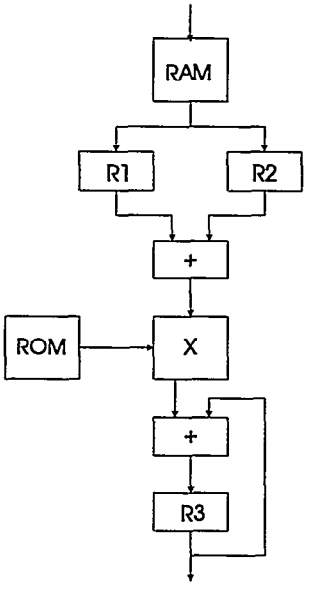

Figure 6: Conventional multiply-accumulate filter architecture.

Reducing the datapath width without degrading the output numerical accuracy plays a central role in achieving a power-efficient Sinc architecture. A Sinc decimator of order $N+1$ is used to decimate the output signal of a bandpass Sigma-Delta modulator of order $2N$. Each octave of oversampling in the Sigma-Delta modulator increases the resolution by $N + 0.5$ bits. However, each octave of decimation by the Sinc decimator increases the datapath width by $N + 1$ bits. Half a bit more than what is required. The computation required by this extra half-a-bit represents irrelevant computation, which can be eliminated.

For a third-order Sinc decimator having a decimation factor of 8, it is possible to reduce the output datapath width from 10 bits to 7 bits, at the expense of a 3 – 4 dB degradation in the SNR (which is equivalent to just over half-a-bit). The saving in computational complexity is 9–14% [8].

5 POWER EFFICIENT LOWPASS FILTER DESIGN

The LPF architecture used is based on the multiply-accumulate architecture. This architecture is shown in Figure 6. The filter designed is a linear phase FIR filter, it has symmetric coefficients. The filter has been modified to take into account the symmetry of the LPF coefficients, by reading two values from the memory adding them together before multiplying them by the filter coefficient. This reduces the number of multiplications by 50%. However, the number of RAM reads remains unchanged. The RAM dissipates 35% of the total LPF power dissipation. Hence, the power saving achieved by eliminating multiplications due to filter symmetry is 33%.

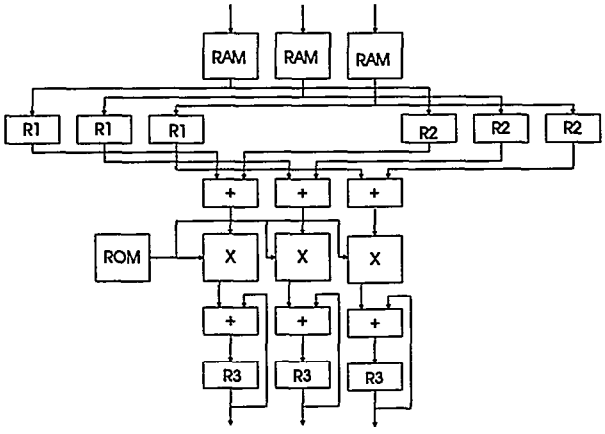

Figure 7: The modified filter architecture.

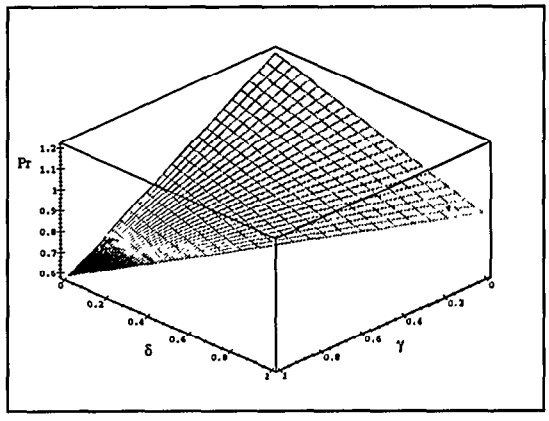

Figure 8: Pr versus γ and δ for a lowpass filter having a 3 parallel-unit datapath.

The number of multiplications is further reduced by using a halfband filter [9]. Halfband filters have half of their coefficients zero. However, this does not reduce the number of multiplications by 50% because halfband filters are usually longer (have more delay units and hence more taps) than non-halfband filters that provide the same out-of-band attenuation.

The halfband filters designed are 25%–30% longer than their corresponding non-halfband filters. However, the designed halfband filters require 30%–35% less multiplications (and RAM reads) than the non-halfband filters. Hence, the power saving achieved by halfband filters is 30%–35%.

Interleaving the adder into the multiplier array reduces redundant computations and decreases the delay of the multiplier accumulator. This achieves a 17% reduction in power dissipation for a 20×20 multiplier accumulator array [10].

5.1 Variable Resolution Lowpass Architecture

To reduce the power dissipation when a lower resolution is sufficient, the datapath is divided into parallel units as shown in Figure 7. The blocks corresponding to the least significant bits are deactivated when a lower resolution is sufficient. In Figure 7, the datapath is shown consisting of 3 parallel units, making its resolution one of three values (12, 16 or 20 bits).

The datapath of the LPF consists of functional blocks as shown in Figure 7. The power dissipation of each functional block F can be divided into two components, the first component is a constant independent of the datapath width, while the second component is proportional to the datapath width:

$$P_F = P_{0F} + P_{1F}N \tag{3}$$

Assume that the power dissipation components, for the entire lowpass filter, are P_{0T} and P_{1T} for the fixed and width-dependent components respectively. If the datapath is not divided into parallel units the total power dissipation is given by:

$$P_T = P_{0T} + P_{1T}N \tag{4}$$

If the datapath is divided into M parallel units. The power dissipation when the full accuracy of the datapath is required is:

$$P_T = P_{0T}M + P_{1T}N \tag{5}$$

Notice that the power dissipation, in this case, has increased by $(M-1)P_{0T}$. When the low resolution is sufficient, the power dissipation is given by:

$$P_T = P_{0T} + P_{1T}N_L \tag{6}$$

In this case, the power dissipation has decreased by $(N-N_L)P_{1T}$. N_L is the datapath width in the low resolution case. The amount of power saving is dependent on the percentage of time the datapath is required to operate at each resolution, as well as the ratio between the two power components, P_{0T} and P_{1T}.

Figure 8 shows the relative power dissipation, for a 3 parallel-unit datapath, versus γ and δ. γ is the probability that a 12 bit resolution is sufficient. δ is the probability that a 16 bit resolution is sufficient.

From Figure 8, it can be seen that the maximum reduction in power dissipation that can be achieved using this technique is 40%. This is achieved when a 12 bit resolution is sufficient for most of the time. When the three resolutions are equi-probable, the reduction in power dissipation is 10%.

In this section, several low-power techniques have been employed in the design of the lowpass filter. The power saving each technique can achieve is shown in Table 2. The total reduction in power dissipation is about 4 times.

Table 2: Power savings for the high-level low-power techniques used in the design of the lowpass filters.

Low power technique	Power saving
Symmetric filter coefficients	33%
Halfband filter	30 – 35%
Interleaved MAC array	17%
Datapath division	10 – 40%
TOTAL	65 – 78%

6 VLSI IMPLEMENTATION OF THE DECIMATION FILTER

The decimation filter is designed to generate a $12-20$ bit resolution sampled signal from a $1-2$ bit resolution oversampled signal. The first stage of the decimation filter is the Sinc decimator. The order of the Sinc decimator, as well as its decimation factor, can be programmed.

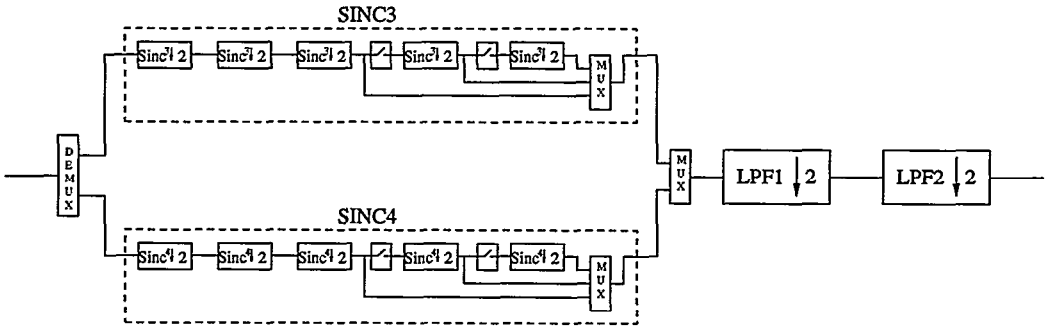

Figure 9: Block diagram of the designed decimation filter.

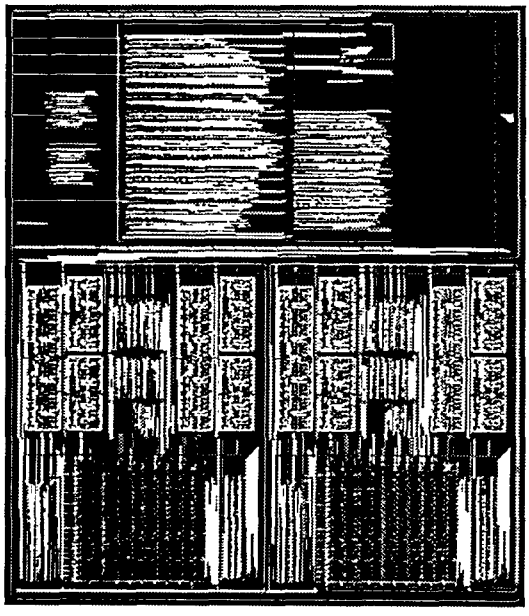

Figure 10: VLSI layout of the decimation filter.

The Sinc decimator can be a third order Sinc decimator. This accepts a one-bit sampled signal from a fourth-order bandpass Sigma-Delta modulator. The Sinc decimator can also be a fourth order Sinc decimator. This accepts a two-bit sampled signal from a sixth order bandpass Sigma-Delta modulator. The decimation factor of the Sinc decimator can be 8, 16 or 32.

The second stage of the decimation filter is the lowpass decimation filter. The lowpass decimation filter consists of two stages in cascade, each stage does decimation by a factor of two. Each lowpass stage is a linear phase stage. Each lowpass filter can be programmed to be halfband filter or a non-halfband filter. The output resolution of each lowpass stage can be programmed to be 12, 16 or 20 bits. The number of taps required in the first LPF stage is 15 taps, while the number of taps required in the second LPF stage is 47 taps.

Figure 9 shows a block diagram of the decimation filter. The decimation filter was designed in a 3.3 Volt $0.5\mu m$ CMOS technology. The total number of transistors required for the decimation filter is 187 K. The total area of the decimation filter is 7.4mm $\times$ 6.4mm. The VLSI layout of the decimation filter is shown in Figure 10.

7 CONCLUSIONS

A decimation filter to be used as part of a Sigma-Delta A/D converter has been designed in a 3.3 Volt $0.5\mu m$ CMOS technology. The filter design incorporates several high-level low-power design techniques, such as operation minimization, datapath width optimization, multiplication elimination and block deactivation. These low-power design techniques, as shown in sections 4 and 5, are able to achieve up to 4 times reduction in power dissipation.

REFERENCES

[1] A. P. Chandrakasan, A. Burstein, and R. W. Brodersen, "A low-power chipset for a portable multimedia I/O terminal," *IEEE Journal of Solid State Circuits*, vol. 29, pp. 1415–1428, Dec. 1994.

[2] J. Mitola, "The software radio architecture," *IEEE Communications Magazine*, pp. 26–38, May 1995.

[3] J. A. Wepman, "Analog-to-digital converters and their applications in radio receivers," *IEEE Communications Magazine*, pp. 39–45, May 1995.

[4] H. Meyr and R. Subramanian, "Advanced digital receiver principles and technologies for PCS," *IEEE Communications Magazine*, pp. 68–78, Jan. 1995.

[5] P. M. Aziz, H. V. Sorensen, and J. V. D. Spiegel, "An overview of sigma-delta converters," *IEEE Signal Processing Magazine*, pp. 61–84, Jan. 1996.

[6] J. C. Candy, "Decimation for sigma delta modulation," *IEEE Transactions on Communications*, vol. COM-34, pp. 72–76, Jan. 1986.

[7] B. P. Brandt and B. A. Wooley, "A low-power area efficient digital filter for decimation and interpolation," *IEEE Journal of Solid State Circuits*, vol. 29, pp. 679–687, June 1994.

[8] E. N. Farag, *VLSI Low-Power Digital Signal Processing*. PhD thesis, University of Waterloo, Waterloo, Ontario, Canada, 1997.

[9] P. P. Vaidyanathan, *Multirate systems and filter banks*. Englewood Cliffs, N.J.: Prentice Hall, 1993.

[10] E. N. Farag, R.-H. Yan, and M. I. Elmasry, "Power-efficient multiplier-accumulator design for FIR filters," in *IEEE Canadian Conference on Electrical and Computer Engineering*, vol. 1, (St. John's, Newfoundland, Canada), pp. 27–30, May 1997.