Electrical Engineering
Washington University in St.Louis
School of Engineering & Applied Science

Novel Numerical Representations for Low-Power Audio Signal Processing

Roger Chamberlain*†, Yen Hsiang Chew*, Varuna DeAlwis*, Eric Hemmeter*,
John Lockwood*, Robert Morley*, Ed Richter*, Jason White*, and Huakai Zhang*
*School of Engineering and Applied Science, Washington University, St. Louis, MO
†Hearing Emulations, LLC and BECS Technology, Inc., St. Louis, MO

Abstract

One of the major technical issues facing the designers of modern digital hearing aids is the need to minimize the power consumption of the system to prolong battery life. As new signal processing techniques are proposed, the computational requirements invariably grow, putting additional pressure on power consumption. In this work, we investigate the use of non-standard numerical representations for the audio signals being processed, showing how the power consumption can be lowered for audio signal processing while maintaining (and even improving) overall signal quality.

Standard numerical representations for general computation include fixed-point representations (typically 16 bits) and floating-point representations (either 32- or 64-bit IEEE standard). In this study, we compare the power consumption of a 16-bit linear representation with several different floating-point representations (4- to 6-bit exponent and 4- to 6-bit mantissa) and a 9-bit logarithmic notation. Each representation is tailored to provide a dynamic range of approximately 100 dB and a signal-to-quantization-noise ratio of approximately 30-35 dB (i.e., optimized for understanding of speech signals). The power consumption is investigated while computing a series of multiply-accumulate operations. The multiply-accumulate (MAC) is the most common computation in audio signal processing.

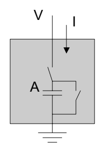
For each representation, we design a hardware MAC unit in the VHDL language and perform a standard-cell synthesis, layout, and place-and-route targeting the AMI Semiconductor 0.5 micron VLSI integrated circuit process. The resulting design is simulated using the Mentor Graphics MACH-PA power analysis tool, with input vectors modeling a 21-tap finite impulse response band-pass filter. The simulation output both verifies correct operation of the circuit and provides information on power consumption.

The results show a significant power savings using both the floating-point representations and the logarithmic representation. This is primarily due to the ability to either eliminate (in the case of the logarithmic representation) or significantly reduce the size of the hardware multiplier required as part of the MAC unit. We will present both novel techniques for implementing the accumulation function with a logarithmic representation as well as the power consumption associated with each numeric representation.

This material is based upon work supported by the NIH under grant 1R4-3DC04028-02, J. Goldstein, PI.

Power Is Used Charging Capacitance

Every time a bit is flipped capacitance is either charged or discharged. Each bit can be thought of as a switch. V is the voltage across the bit, I is current charging the capacitor, and f is the frequency at which the bit can be flipped.



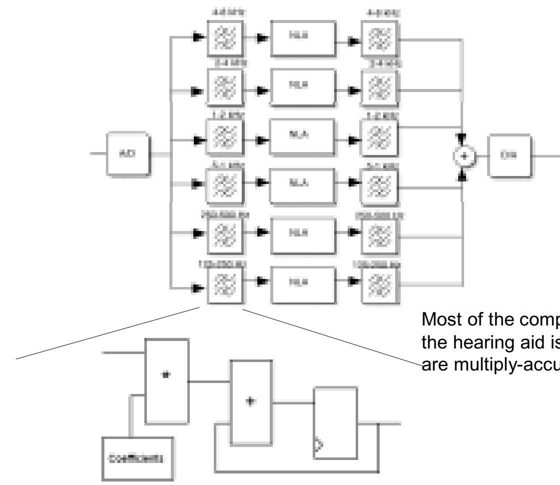
Charge on plate A, Q=CV
Current, I=fCV
Power, P=IV
P=fCV^2

Determined by process and Requirements

Controlled by the number of bits switching
More bits = more power

We are investigating different number representations in order to decrease the number of bits that need to be flipped, thus saving power.

Hearing Aid Architecture



Most of the computations the hearing aid is performing are multiply-accumulates.

Multiplication vs. Addition

We can see from these examples that many more bits are needed to perform a multiplication than to perform an addition.

```
      10101
   x  11001
      10101
     00000_
    00000__
   10101___
  10101____
 1000001101
```

```
  10101
+ 11001
 101110
```

Using a logarithmic representation of the audio signals allows us to take advantage of this difference in the effort to reduce power needed.

By using a logarithmic representation we can replace the multipliers with adders, because

log(A x B) = log(A) + log(B)

However, then the adder must be modified as well. Lets start with this:

$$\log\left(1+\frac{B}{A}\right)=\log\left(\frac{A+B}{A}\right)=\log(A+B)-\log(A)$$

We want log(A+B), so:

$$\log(A+B)=\log(A)+\log\left(1+\frac{B}{A}\right)$$

Since this would be harder to calculate than the original multiplication and addition, this function is performed by a look-up table. To eliminate even more unnecessary calculations, the look-up table is only accessed if the two inputs to the “adder”, i.e., the log multiplier, are significant with respect to each other. In other words, if one of the values is small enough to be negligible then the look-up is skipped and the larger number passed through, thus saving more power.

Floating Point Representations

Our floating-point representations are named based on the number of bits in the exponent and mantissa (each representation has an additional sign bit, as well). For example, a 4-5 representation has four exponent bits and five mantissa bits. In bit form they are expressed like this:

1	n	m
sign (s)	exponent (exp)	mantissa (mant)

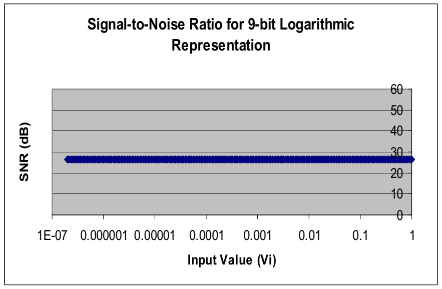
Mathematically, the numbers are then expressed, with the exponent in excess notation, in this form: $(-1)^s \cdot \frac{mant}{2^m} \cdot 2^{\left(\exp-\left(2^{n-1}-1\right)\right)}$

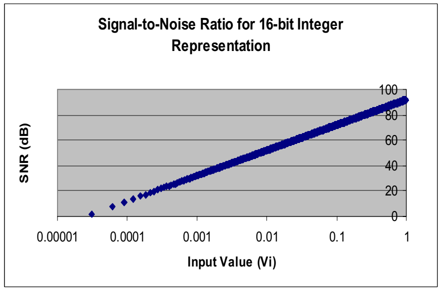
For example, the number 1100110101 is the 4-5 number 1 1001 10101. Using this example, we can see the relationship between the binary and decimal versions of the numbers. Note: 1001 is the unsigned binary equivalent 9 and 10101 corresponds to 21.

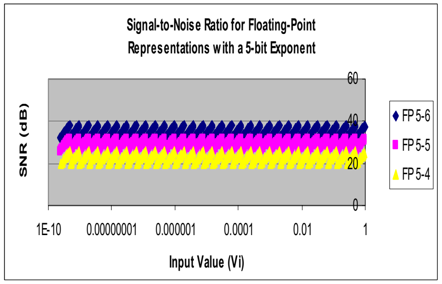
$$(-1)^1 \cdot \frac{21}{2^5} \cdot 2^{9-\left(2^3-1\right)}$$
$$= -1 \cdot \frac{21}{32} \cdot 2^{9-7}$$
$$= -\frac{21}{32} \cdot 2^2$$
$$= -2.625$$

These floating-point representations have some benefits from both the linear representation and the logarithmic representation. By changing the number of bits in the exponent and mantissa the signal-to-noise ratio and dynamic range can be varied to meet desired specifications.

Signal-to-Noise Ratio (SNR) and Dynamic Range

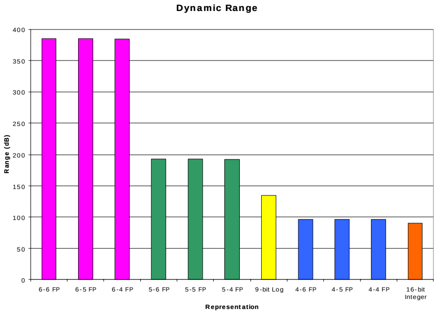






$$\text{SNR (dB)} = 20 \cdot \log_{10}\left(\frac{\frac{V_i}{2\sqrt{2}}}{\frac{V_{i+1}-V_i}{\sqrt{12}}}\right)$$

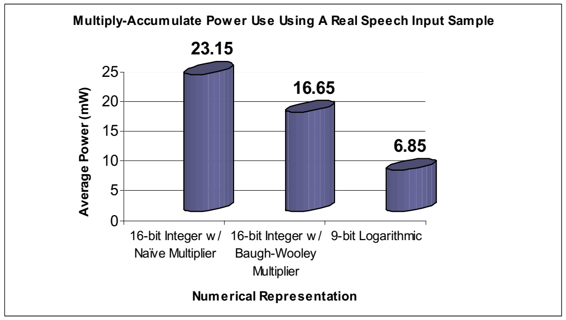
$$\text{Dynamic Range (dB)} = 20 \cdot \log_{10}\left(\frac{V_{\max}}{V_{\min}}\right)$$

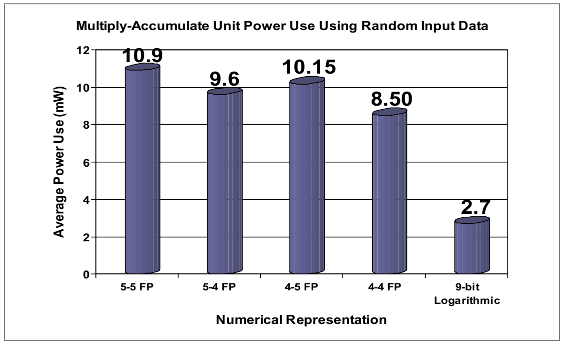


In logarithmic representations, the base of the log sets the SNR, while the number of bits used determines the dynamic range. In floating-point representations, the SNR is determined by the number of bits in the mantissa, while the dynamic range is determined by the number of bits in the exponent. Thus, the SNR and dynamic range can be set independently and the representation used can be chosen based on the desired specifications.

Power Results

The following power analysis results were obtaining using Mentor Graphics Mach-PA spice-like simulator. The simulator provided current usage numbers, thus using the equation P=IV with V=5 Volts, we calculated the power results below.





We believe that the logarithmic representation uses less average power when using random input data, as opposed to real speech data, since it can more frequently bypass the look-up table with random data.

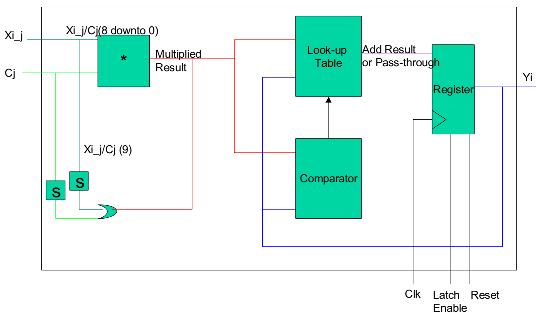
The logarithmic representation uses less power than both the integer and the floating-point representation. It appears that the floating-point representation uses less power than the integer representation as well.

Multiply-Accumulate Units

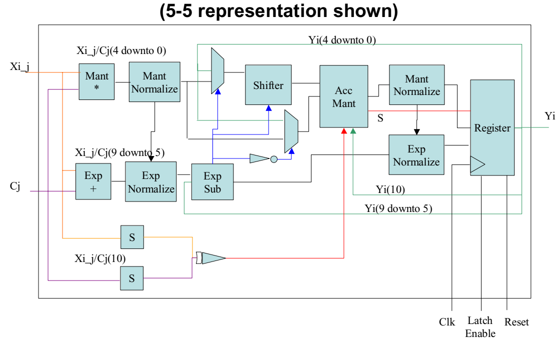
The multiply-accumulate (MAC) units are used to perform the convolution in the 21-tap finite impulse response (FIR) filter. The convolution is given by the equation below, where Xi is the current input, Yi is the current output, and Cj is the coefficient.

$$Y_i = \sum_{j=0}^{20} X_{i-j} \cdot C_j$$

Functional Block Diagram of the 9-bit Logarithmic MAC



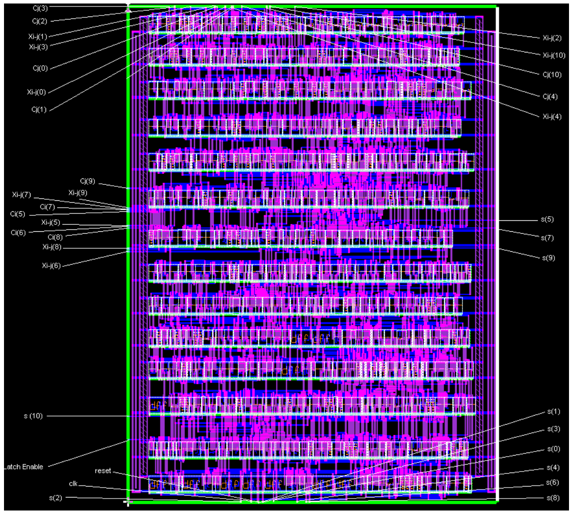
Functional Block Diagram of the Floating-Point MACs. (5-5 representation shown)



Layout

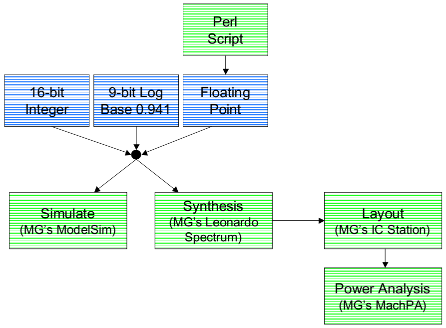
The layout for each representation of the multiply-accumulate unit was done using Mentor Graphics IC Station. The layout targeted an AMI 0.5 micron process and used the Standard Cell Library provided by the Mentor Graphics Higher Education Program. A typical layout is shown below.

Mentor Graphics IC Station Layout for the 5-5 Floating-Point Representation

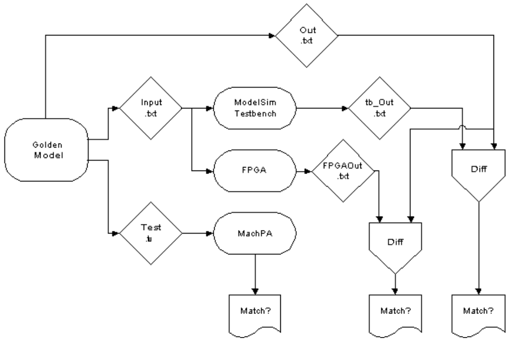


Flows

Design Flow



Verification Flow



For Further Information

http://www.ccrc.wustl.edu/hearingaid/

hearingaid@ccrc.wustl.edu