FIR FILTER HARDWARE REDUCTION WITH ADAPTIVE DELTA-MODULATION

William F. Lawrence

The Bendix Corporation
Communications Division
Baltimore, Maryland 21204

Robert W. Newcomb

Dept. of Electrical Engineering
University of Maryland
College Park, Maryland 20742

ABSTRACT

This paper presents a new type of digital system which makes use of adaptive delta-modulation for input digitization and is particularly suited to the hardware implementation of small, low power, programmable digital filters. An experimental prototype was built which demonstrated a finite impulse response (FIR) digital filter. This filter has no requirement for hardware multipliers and has a significant reduction in input sample storage capacity when compared to equivalent pulse code modulation (PCM) filters. Quantitative as well as qualitative results show that this new filter is especially useful with voice inputs.

INTRODUCTION

Traditionally, digital filter designs have been based upon PCM digitization techniques. Certainly, this is the obvious choice for filters implemented on a general purpose digital computer. Direct hardware implementations also tend to lean towards PCM since each sample is represented by a stand alone digital word whose accuracy is easily increased by increasing the word size.

In hardware designs where physical size and power consumption are of prime importance, PCM designs suffer several problems. This is especially true when one tries to apply digital signal processing techniques to hearing-aids. If N input samples are stored in B bit words, then NB storage bits are required. Most PCM filters also require multipliers which are costly, slow and power hungry. In addition, PCM analog-to-digital converters are usually large and relatively expensive.

The unique characteristics of delta-modulation offer an attractive alternative to normal FIR PCM digital filters in low power, small size applications. Linear delta-modulation filters have already been described by (1) and (2). This paper describes adaptive delta-modulation FIR filter which offers good performance at lower bit rates than filters using linear delta-modulation. Only N bits of memory are required to store the input samples for an Nth order filter. In addition, if an adaptation scheme such as (3) is used where step sizes are powers of two, no hardware multipliers are required.

An experimental prototype of such an adaptive delta-modulation filter (ADMF) was built and tested. The results proved that the concept is sound and works very well with input signals such as speech, where the average spectral density falls off with increasing frequency.

FILTER DESCRIPTION

It is well known that the general form of an Nth order FIR digital filter can be expressed as follows:

$$H(z)= \sum_{k=0}^{N-1} a_k z^{-k} \qquad (1)$$

where a_k are the filter coefficients and z^{-1} represents a delay of T seconds. Equation (1) can be equivalently represented in terms of an input sequence x(nT) and an output sequence y(nT) as follows:

$$y(nT)= \sum_{k=0}^{N-1} a_k \, x(nT-kT) \qquad (2)$$

To simplify the notation, let $y(nT)= y_n$ and $x(nT-kT) = x_{n-k}$. Equation (2) can also be written in terms of differences

CH1374-7/79/0000-0849$00.75 © 1979 IEEE

from one input sample to the next as follows:

$$y_n = y_{n-1} + \sum_{k=0}^{N-1} a_k \Delta x_{n-k} \qquad (3)$$

where $\Delta x_{n-k} = x_{n-k} - x_{n-k-1}$.

Equation (3) shows that an equivalent filter to that represented by equation (2) can be built which works on input differences. This is where delta-modulation can be used to advantage.

Various adaptive delta-modulation schemes are described throughout the literature. Of particular concern here is the technique described by (3). In general, adaptive delta-modulators encode differences in input analog signals and reconstruct by integrating a weighted version of the digital sequence. The input signal x(t) is compared to the reconstructed signal $\hat{x}(t)$ and the error is quantitized to either +1 or -1. Figure 1 shows a particular implementation of Winkler's algorithm.

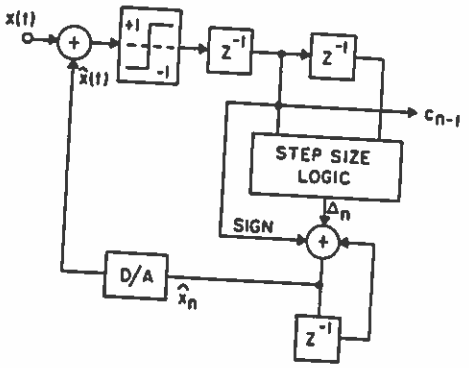

Figure 1: Adaptive Delta-Modulator

Note in Figure 1 that c_{n-1} is the modulator's output with

$$c_n = \mathrm{sgn}(x_n - \hat{x}_n) = \pm 1 \qquad (4)$$

and

$$\hat{x}_n = \hat{x}_{n-1} + \Delta_n \qquad (5)$$

In our case we choose

$$\Delta_n = 2^{\ell_n} \Delta^\circ \, c_{n-1}, \quad \Delta^\circ = \text{minimum step size} \qquad (6)$$

and

$$\ell_n = \begin{cases} \ell^{max} & \text{if } c_{n-1} = c_{n-2} = \pm 1,\ \ell_{n-1} = \ell^{max} \\ \ell_{n-1} + 1 & \text{if } c_{n-1} = c_{n-2} = \pm 1,\ 0 \le \ell_{n-1} < \ell^{max} \\ \ell_{n-1} - 1 & \text{if } c_{n-1} = -c_{n-2} = \pm 1,\ 0 < \ell_{n-1} \le \ell^{max} \\ 0 & \text{if } c_{n-1} = -c_{n-2} = \pm 1,\ \ell_{n-1} = 0 \end{cases} \qquad (7)$$

Now assume

$$\Delta x_n = x_n - x_{n-1} = \hat{x}_n - \hat{x}_{n-1} = \Delta_n. \qquad (8)$$

Under this assumption, equations (3), (6) and (8) can be combined to give the following:

$$y_n = y_{n-1} + \Delta^\circ \sum_{k=0}^{N-1} a_k c_{n-k-1} \, 2^{\ell_{n-k}} \qquad (9)$$

Equation (9) is thus the defining relation for a FIR digital filter which uses adaptive delta-modulation for input quantization. Note that $c_{n-k-1} = \pm 1$ and is, therefore, only a sign term. If binary logic circuits are used to implement equation (9), then multiplication by $2^{\ell_{n-k}}$ is equivalent to shifting the multiplicand by ℓ_{n-k} places. This filter thus has the nice property of requiring no hardware multiplications; simple shifts, sign changes, and additions of the coefficients a_k suffice.

The last term in equation (9) can be thought of as y_n and the output of the filter would be simply written as:

$$y_n = y_{n-1} + \Delta y_n. \qquad (10)$$

Thus, the output y(t) would be the digital to analog (D/A) converted version of the output of a first order recursive digital filter on y_n with a unity feedback coefficient. In practice, this can easily be unstable. One solution to this problem would be to use a feedback coefficient close to but less than one. Although this would solve the problem, it would increase circuit complexity by the addition of a multiplier. A simpler solution to this problem of possible insta-

bility would involve the D/A conversion of y_n followed by an analog lossy integrator with a long time constant. Such a configuration is shown in Figure 2. The digital processor provides storage for the N, B bit filter coefficients and N single bit input samples.

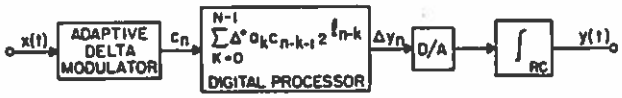

Figure 2: Adaptive Delta-Modulation Filter

The coefficients (a_k) used in the ADMF of Figure 2 are exactly those of the well-known FIR filters which use PCM for input digitization. Any synthesis techniques for PCM filter coefficients will apply equally well to the ADMF.

As one might expect, the quality of the ADMF's output depends heavily upon the quality of the adaptive delta-modulator (ADM). Standard sources of noise normally found in digital filters are present in the ADMF. These include coefficient and arithmetic roundoff along with quantization noise and are analyzed elsewhere by (4). The biggest source of error and noise is due to the ADM. It can be shown (5) that the error introduced into the output of the ADMF is basically the sum of two errors. One is introduced by coefficient roundoff. The other is a component that looks like the ADM errors after being passed through a perfect FIR digital filter whose coefficients are the same as those in the ADMF.

EXPERIMENTAL RESULTS

A detailed analysis of the ADMF was performed and can be found elsewhere (5). This analysis was useful in providing some intuition as to the sources of error and soundness of the ADMF concept but was not able to provide a closed-form representation of the noise behavior. For this reason, an experimental prototype was constructed and some of the results are described here.

Figure 3 is a block diagram of the experimental test configuration for the ADMF. The input was either an audio tone or voice. This signal was next passed through an automatic gain control (AGC) to provide an optimum level for the ADM. To reduce input alias distortion, a three-pole low pass filter was inserted between the AGC and the ADMF. The ADM was essentially that of Figure 1 with four possible step sizes ($\pm\Delta^{\circ}$, $\pm 2\Delta^{\circ}$, $\pm 4\Delta^{\circ}$, and $\pm 8\Delta^{\circ}$). The digital processor performed all the storage, timing, control and arithmetic to compute y_n. Here N = 64 and each coefficient was represented by an 8-bit (including sign) binary word. Accumulators were taken to 12 bits to reduce overflow problems.

Since circuit complexity was to be kept to a minimum, serial processing was used. This meant that the maximum input sample rate was limited to 1/N times the maximum clock rate of the digital processor. CMOS logic was used for the experiment and the input sampling rate was thereby limited to 16 kHz. Higher sampling rates would have been desirable, but it was felt that 16 kHz was certainly adequate for a qualitative analysis.

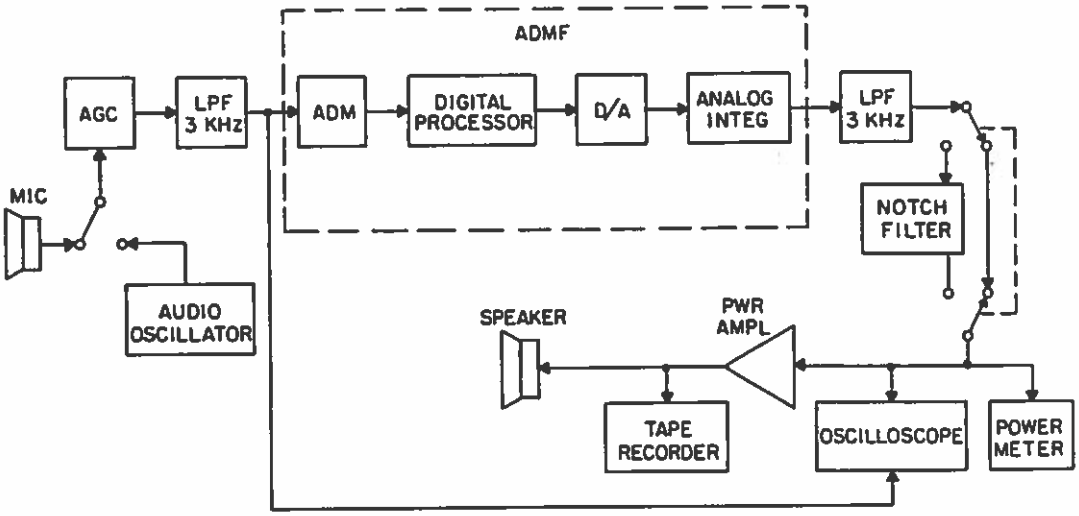

Figure 3: Experimental Test Configuration

The signal output from the digital processor was next passed through an 8-bit R-2R D/A converter and there integrated by a simple RC filter with a 30 Hz cutoff. After another three-pole low pass filter, a variable notch filter could be inserted or deleted to allow signal-plus-noise-to-noise measurements.

Subjective tests with voice inputs tended to sound very good. Since the ADMF filters the ADM noise just as it would any input signal, one would expect the subjective signal to noise ratio to be much better for low pass filters. This was certainly the case. The lower the cutoff frequency of the ADMF, the better the subjective signal-to-noise ratio. In any event, the output always sounded better with the ADMF than with an ADM alone.

It was found that pure sinusoidal inputs tended to give poor output responses. The frequency behavior was very good and agreed well with theory but the output noise level was high. This high noise level was found to be entirely due to the ADM which would adapt slightly differently on each cycle of the sinusoid and produce a large amount of amplitude modulation (AM). Figure 4 shows a typical frequency response for a low pass FIR filter implemented with the ADMF. The measured response agreed well with the theoretical response until the signal fell below the noise floor. The random nature of voice signals tended to eliminate this AM noise and thereby increased the effective dynamic range of the ADMF.

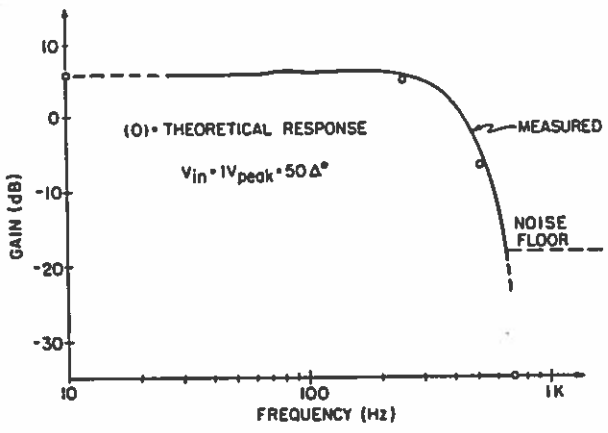

Figure 4: Response of Filter

CONCLUSIONS

A new type of system has been presented which makes use of adaptive delta modulation. This system was found to be well suited to the processing of speech signals. By eliminating hardware multipliers and the majority of the normally required input sample storage space, this system should prove useful for applications such as programmable hearing aids where small size and low power are of particular importance.

An experimental prototype of this ADMF was built, tested, and evaluated. It was found that it performed as expected and filter function synthesis can use already proven design techniques for conventional FIR digital filters.

References

(1) G.B. Lockhart, "Digital Encoding and Filtering Using Delta Modulation", RADIO AND ELECTRONIC ENGINEER Vol. 42, No. 12, pp. 547-551: Dec. 1972.

(2) A. Peled, B. Liu, "A New Approach to the Realization of Non-Recursive Digital Filters", IEEE TRANSACTIONS ON AUDIO AND ELECTROACOUSTICS Vol. AU-21, No. 6, pp. 477-484: Dec. 1978.

(3) M.R. Winkler, "High Information Delta Modulation", 1963 IEEE INTERNATIONAL CONVENTION RECORD, pt. 8, pp. 260-265.

(4) B. Gold, C.M. Rader, DIGITAL PROCESSING OF SIGNALS: McGraw Hill, New York, New York, 1969.

(5) W.F. Lawrence, "An Adaptive Delta Modulation System with Applications to Programmable Hearing-Aids": Ph.D. Dissertation, Dept. of Electrical Engineering, University of Maryland, College Park, 1978.

ICASSP 79

1979 IEEE International Conference on Acoustics, Speech & Signal Processing

Held at the
INTERNATIONAL INN
WASHINGTON, D.C.
APRIL 2–4, 1979

RECORD

79CH1379-7 ASSP