THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 62, No. 4, April 1983
*Printed in U.S.A.*

# A Circuit That Changes the Word Rate of Pulse Code Modulated Signals

By J. C. CANDY and O. J. BENJAMIN

(Manuscript received November 2, 1982)

*In this paper we describe a circuit that accepts pulse code modulated signals sampled at about 8 kHz and resamples them at any desired rate up to 512 kHz. When the sampling satisfies Nyquist's criterion, the distortion introduced is at least 35 dB below the signal level. The circuit uses a digital low-pass filter to interpolate sample values, and it may be integrated as about 2500 gates on a 5 mm*$^2$ *chip.*

## I. INTRODUCTION

It often is impractical to synchronize all of the clocks of an extensive digital network. Consequently, data will arrive at connections out of synchronism and special circuits are needed to bring them into time with the local clock. For irregular bursts of data, synchronism can be easily obtained using buffer memories, but for continuous streams of data, such methods are useful only for very small discrepancies in clock frequencies. A case of particular interest in telephone networks is the transmission and processing of pulse code modulation (PCM). Changing the word rate of such data can introduce objectionable noise into the signal. We describe a circuit that uses digital filters to contain this noise.

## II. RESAMPLING

We know that when an analog signal, $x(t)$, having spectral density $X(\omega)$ and bandwidth $\omega_0$ is sampled at regular intervals, $\tau$, the spectral density of the sampled signal can be expressed as the sum of images

$$X'(\omega) = \sum_{-\infty}^{\infty} X\left(\omega + \frac{2\pi n}{\tau}\right). \tag{1}$$

When $\omega_0 \tau < \pi$, the original signal can be recovered by filtering out the images for $n \neq 0$.

If we were to change the sampling rate[1] by first holding each sample value constant throughout its period and resampling at the new period, $\tau_1 < \pi/\omega_0$, the spectral density of the resampled signal can be expressed as

$$X''(\omega) = \sum_n \sum_k X\left(\omega + \frac{2\pi n}{\tau} + \frac{2\pi k}{\tau_1}\right) H\left(\omega + \frac{2\pi n}{\tau}\right), \tag{2}$$

where

$$H(\omega) = \frac{\sin\left(\frac{\omega\tau}{2}\right)}{\frac{\omega\tau}{2}} = \text{sinc}\,(f\tau). \tag{3}$$

When we try to recover the original signal from $X''(\omega)$ by filtering out the components for $n \neq k \neq 0$, it is contaminated by cross modulation products for which

$$0 < \left(\frac{n}{\tau} + \frac{k}{\tau_1}\right) < f_0 = \frac{\omega_0}{2\pi}. \tag{4}$$

Notice that when $\tau$ and $\tau_1$ are integer multiples of one another, this condition never holds and reflected noise is absent from the baseband.

An obvious means of eliminating in-band cross-modulation products from eq. (2) is to smooth out the high-frequency components of the sampled signal $x'(t)$ before resampling it. This has been accomplished by replacing the sample and hold represented by $H(\omega)$ in eq. (3) by a better low-pass filter that interpolates new sample values from the old ones.[1,2] Another implementation of this method is demodulation with a digital/analog (D/A) converter, analog smoothing, and remodulation with an analog/digital (A/D) converter. The method[1] presented here raises the sample rate to a high multiple of the original rate, smooths the sample values with digital filters, holds the smoothed samples, and resamples them at the desired rate.

## III. A CIRCUIT FOR RESAMPLING 8-kHz PCM

We will describe a circuit for resampling a 3.5-kHz signal that has been pulse code modulated at a nominal 8-kHz rate using 16 bits per word. The new sampling can be at any rate up to 512 kHz and even higher rates can be accommodated by minor modification of the circuits. The technique first raises the sampling rate 16 times to 128 kHz using digital interpolating filters to smooth out all unnecessary images of the signal, leaving only those that are adjacent to the new rate and its harmonics. The high-frequency code is placed in a holding register from which the output is gated at the desired rate. We will see that the filter action of this hold, sinc $(f/128)$, provides more than 30-dB attenuation of the cross-modulation products that fold into baseband.

Because the complexity of digital filters increases sharply with increasing sampling rate, it pays to raise the rate in stages.[1] Figure 1 illustrates the process. Our first stage raises the sampling rate four times from 8 to 32 kHz, employing a low-pass filter that cuts off sharply to attenuate spectral images between 4 and 28 kHz. The second stage raises the rate to 128 kHz by simple linear interpolation, and the third stage is a holding register. Most of the circuits used originally were

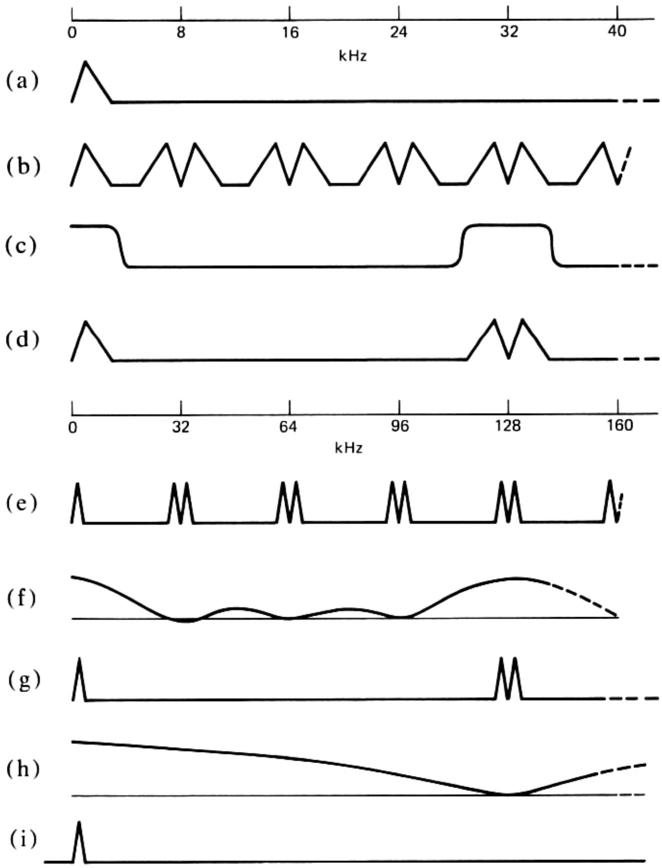


Fig. 1—An illustration of the signal's spectrum at various stages of the conversion: (a) The original signal. (b) The signal sampled at 8 kHz. (c) The response of the low-pass filter. (d) and (e) The output from the low-pass filter. (f) The response of the linear interpolator. (g) The output of the interpolator. (h) The response of the holding circuit. (i) The held signal.

designed to be part of an oversampled coder/decoder (codec).[3] We shall summarize their relevant properties in the following sections.

## IV. THE LOW-PASS FILTER

The low-pass filter shown in Fig. 2 processes data at 32 kilowords per second. Its $z$-transform response is given by

$$F(z) = \frac{1}{32}\frac{1 - z^{-4}}{1 - z^{-1}} \cdot \frac{1 - \frac{5}{4}z^{-1} + z^{-2}}{1 - \frac{19}{16}z^{-1} + \frac{31}{64}z^{-2}} \cdot \frac{1 - \frac{3}{4}z^{-1} + z^{-2}}{1 - \frac{23}{16}z^{-1} + \frac{55}{64}z^{-2}}. \tag{5}$$

Each word of its 8-kHz input signal is repeated four times and fed into two second-order sections. This filter attenuates the images of the signal in the range 4 to 28 kHz by more than 34 dB. Its output is a good approximation of pulse code modulation at 32 kHz. The spectral response of the entire resampler is shown in Fig. 3. The zeros at 4.5 and 6 kHz are introduced by the second-order sections, and those at 8 and 16 kHz by the repetition of input words.

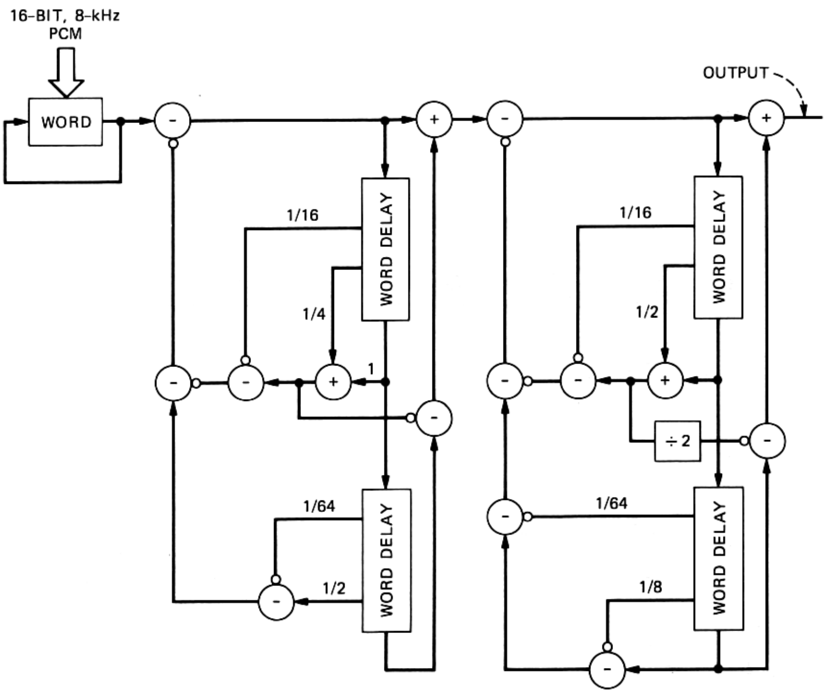


Fig. 2—An outline of the low-pass filter, clocked at 32 kilowords per second, that is used to raise the sampling rate from 8 to 32 kHz.

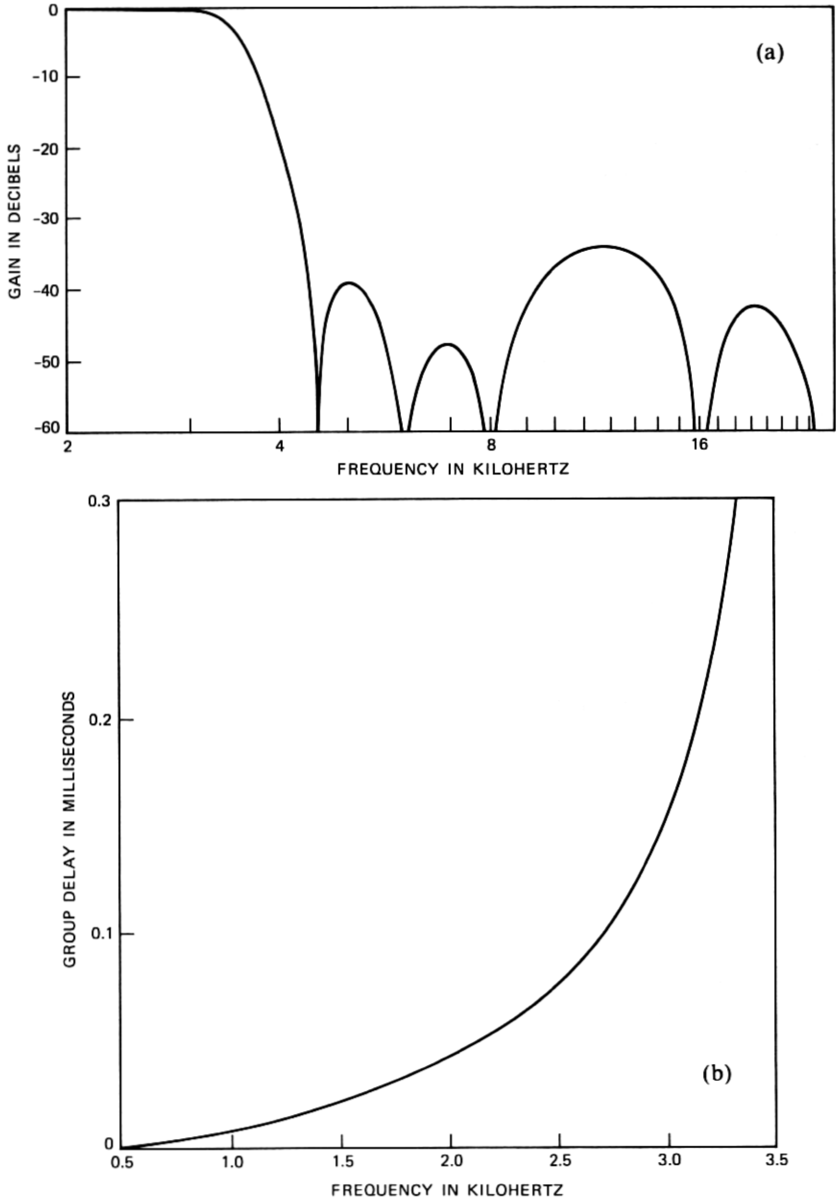


Fig. 3—Spectral response of the resampler. (a) The calculated gain of the cascaded filters used in the resampler. (b) The group delay of the filters.

## V. LINEAR INTERPOLATION

Simple linear interpolation of three sample values increases the sampling rate from 32 to 128 kHz. This process has frequency responses that can be expressed as

$$I(\omega) = \left( \frac{\text{sinc } (f/32)}{\text{sinc } (f/128)} \right)^2 . \tag{6}$$

Its attenuation of spectral images in the range 28 to 36, 60 to 68, and 92 to 100 kHz exceeds 40 dB. The small amount of droop introduced into baseband is compensated in the low-pass filter so that the entire circuit has inband gain in the range −0.41 to −0.57 dB. The circuit implementation shown as Fig. 4 is based on the results

$$y(n\tau) = y[(n-1)\tau] + \tfrac{1}{4}\,\Delta x(n\tau)$$

$$\Delta x(n\tau) = x(4n\tau) - x[4(n-1)\tau]$$

and

$$y(4n\tau) = x(4n\tau). \tag{7}$$

After each new input sample enters register $R_1$, the output, held in register $R_2$, increments four times to make its value equal to the input.

## VI. RESAMPLING

Figure 5 shows the circuit that is used to resample the signal at the desired output rate without causing conflict with the internal clock. Here the output from the linear interpolator is loaded into register $R_3$ in time with the internal clock running at 1 MHz. This loading is inhibited by the presence of the output clock, which after a short delay loads $R_4$ from $R_3$. The frequency response associated with the holding action of this circuit may be expressed as

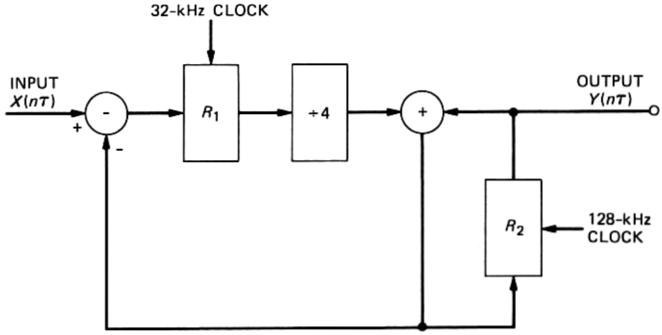


Fig. 4—An outline of the linear interpolator used to raise the sampling rate from 32 to 128 kHz.

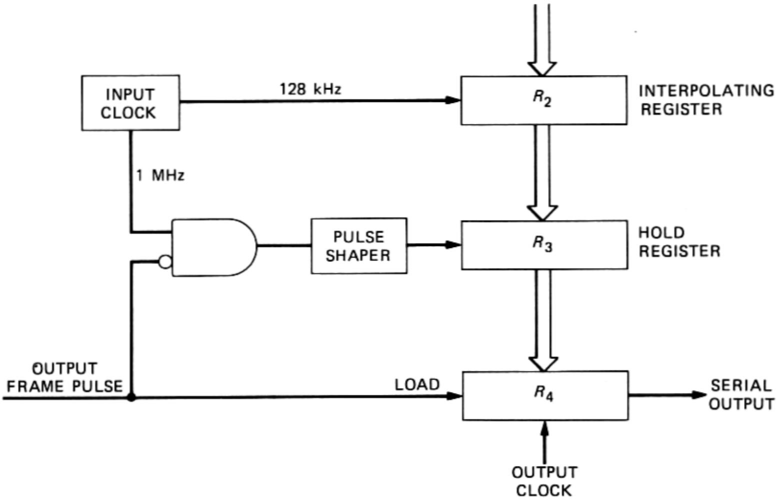


Fig. 5—The circuit used to sample the output. Register $R_2$ is part of the linear interpolator, $R_3$ is loaded from $R_2$ at 1 MHz. An output demand inhibits loading of $R_3$, and then loads $R_4$ from $R_3$.

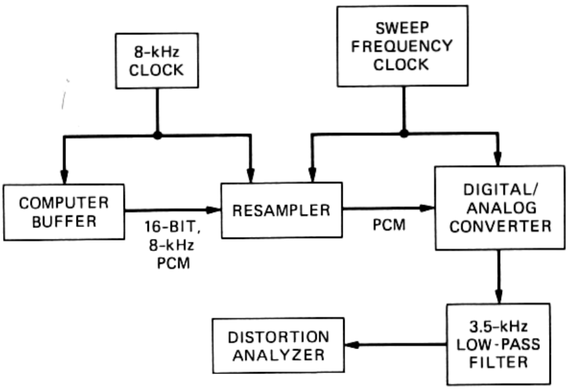


Fig. 6—The test circuit.

$$R(\omega) = \text{sinc}\,(f/128).$$

Images of the signal in the range 124 to 132 kHz are thereby attenuated by more than 30 dB.

## VII. TESTING THE CIRCUIT

A version of the circuit was built of standard digital circuit components and tested in the setup shown in Fig. 6. The net gain of the circuit and its signal-to-noise ratio were measured as the sampling rate was varied continuously from 0 to 256 kHz. For a 1.02-kHz input signal sampled at 8 kHz, the gain remained constant within ±.05 dB and the signal-to-noise ratio always exceeded 40 dB. Input amplitudes were varied in the range 0 to −60 dB. At lower amplitudes quantization noise inherent in the 16-bit input word was significant.

## VIII. CONCLUDING REMARKS

Experience obtained while designing filters for a version of a codec enables us to estimate that the circuits described here can be implemented on about 5 $mm^2$ of silicon in a standard technology. The performance of the circuit is good enough that imperfections introduced by resampling would be insignificant compared with those normally obtained from $\mu$-255 encoding of the signal.


## IX. ACKNOWLEDGMENT

We thank H. S. McDonald for suggesting this use for our circuits, and J. L. Henry for providing information on the integrated circuit implementation.


## REFERENCES


1. D. J. Goodman and M. J. Carey, "Nine Digital Filters for Decimation and Interpolation," IEEE Trans. Acoust., Speech, Signal Processing, *ASSP-25,* 2(April 1977), pp. 121–6.
2. R. E. Crochiere and L. R. Rabiner, "Interpolation and Decimation of Digital Signals—A Tutorial Review," 53 references, Proc. IEEE, *69,* pp. 300–31.
3. J. C. Candy, B. A. Wooley, and O. J. Benjamin, "A Voiceband Codec with Digital Filtering," IEEE Trans. Commun., *COM-29* (June 1981), pp. 815–30.