32-K BIT/S DELTAMODULATION CODECS IMPLEMENTED BY TMS32010 DSP

Ibrahim El-Abd, Mazhar Tayel and Galal A. Fahmy
Department of Electrical Engineering, Faculty of Engineering,
Alexandria University, Alexandria, Egypt.

ABSTRACT

This paper presents two DM codecs implemented by the TMS32010 Digital Signal Processor (DSP) and developed by a proposed development system [1]. The first codec is a 32-k bit/s linear DM codec, and the second is a 32-k bit/s adaptive DM codec. The two codecs are presented as design examples that demonstrate the capabilities of the proposed development system and explain the development procedure and the recommended layout of the program under development.

INTRODUCTION

A development system for the TMS32010 DSP was proposed [1]. The system is based on the "ZX=Spectrum" microcomputer, it has the following features:

It receives analog input signal of level one millivolt (max.), filters it to the band (200-3400 c/s) and converts it to 8-bit digital samples at rate of 8000 or 32000 samples/s.

It converts the processed digital samples into an analog signal.

It stores 12000 input samples in the "ZX-Spectrum micro-computer, and 12000 intermediate (or final) output samples.

An external RAM of 1K-word is used to store the program under development, so that, the program can be easily corrected or altered.

16-bit Input/Output latches are used to interface "ZX-Spectrum" microcomputer with the DSP.

It controls the operation of the TMS32010 DSP by switching ON/OFF its D.C. supply and by controlling its INT and RESET lines. Also, the system feeds the DSP by input samples at a real-time rate.

The operation of the TMS32010 DSP is monitored once every sampling period, i.e. the "ZX-Spectrum" microcomputer recieves from the DSP an intermediate of final result during every sampling period. The TMS32010 operates with its full speed, i.e. 20 MHZ clock.

RECOMMENDATIONS ON PROGRAM LAYOUT:

The designer of any program, (that will be developed by the introduced development system), must take into account the following points:

1- The total processing time for each i/p sample must be less than the time intervals between samples.
2- The TMS32010 must be ENABLED to respond to INT input signal after every processing cycle.
3- The system sends an INT pulse to the TM 32010 dsp as soon as a new sample is stored into the most significant byte of the output latch.
4- The ADC and that are used in the system deal with samples represented by Natural Binary Code (NBC).

Design Example 1:

A 32-K BIT/S Linear Deltamodulation (LDM) Codec:

The theory of DM is presented in many references, such as [3-8]. The theoretical background of the two presented examples is derived from [8].

Figure (1) shows a block-diagram for LDM encoder/decodec. A program is to be developed for the TMS32010 DPS to perform this codec. The output of the encoder will be fed to the input of the decoder to obtain an output y (n) which is equal to the input x (x) as shown in Figure (2). The algorithm that will be executed by the TMS32010 is described by the following equations:

ENCODER

Symbol	Description
x_e (n)	Estimated signal
d (n)	Difference signal
d_{qe} (n)	Quantized difference signal
x_r (n)	Reconstructed signal
b_e (n)	Single bit encoded output
h	coefficient of a first order predictor.

$$x_e(0) = 0 \quad (1)$$
$$d(n) = x(n) - x_e(n) \quad (2)$$
$$b_e(n) = \text{sign } d(n) \quad (3)$$
$$d_{qe}(n) = b_e(n) \times \delta \quad (4)$$
$$x_r(n) = d_{qe}(n) + x_e(n) \quad (5)$$
$$x_e(n+1) = h \times x_r(n) \quad (6)$$

DECODER

$$y_e(0) = 0 \quad (7)$$
$$d_{dq}(n) = b_d(n) \times \delta \quad (8)$$
$$y(n) = d_{qd}(n) + y_e(n) \quad (9)$$
$$y_e(n+1) = y(n) \times h \quad (10)$$

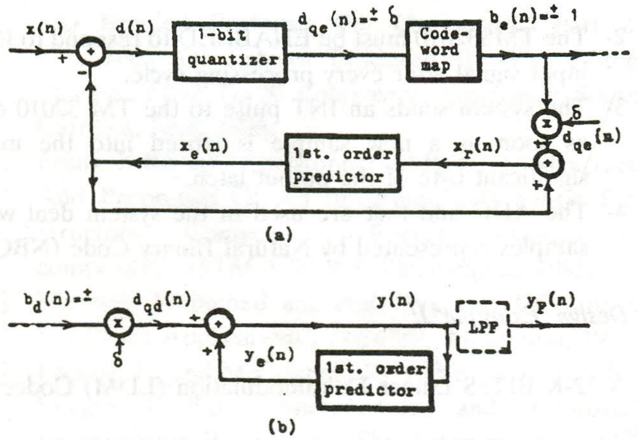

Figure 1. Block diagram for LDM codec; (a) Encoder; (b) Decoder. (N.B. LPF is excluded from the design).

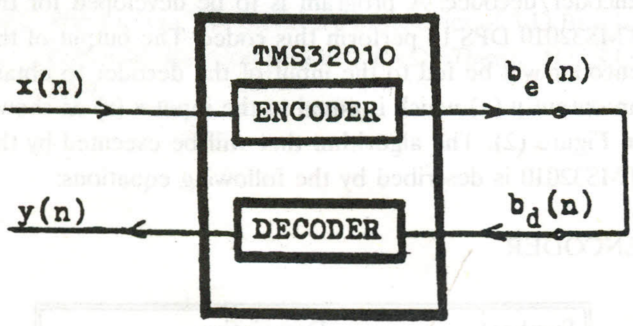

Figure 2. Configuration used for the encode/decode operation.

The theoretical bases for selecting "h" and "δ" are presented in appendix (A).

Figure (3) shows the input and output waveforms as obtained from the non real-time BASIC program, while Figure (4) shows the input and output waveforms as obtained from the real-time program. The output waveforms in both figures are very closely similar, which indicates that the algorithm is correctly implemented by the TMS32010 DSP.

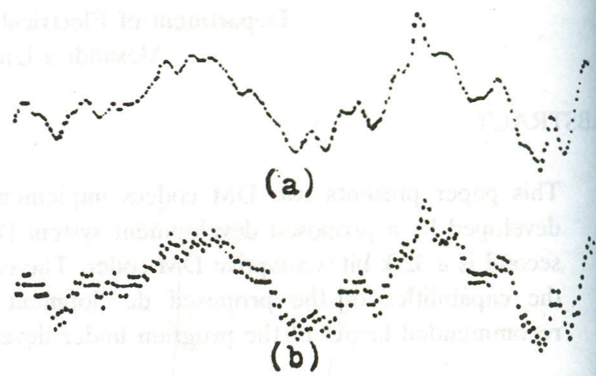

Figure 3. Result of LDM codec, implemented by a BASIC program, run on ZX-spectrum microcomputer; (a) input waveform, sampled at a rate of 32 K sampled/s; (b) output waveform ($h=.95$, $\delta=16$).

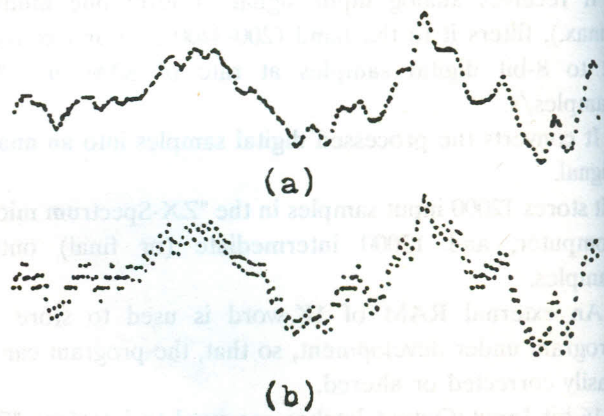

Figure 4. Results of LDM codec algorithm, implemented by TMS32010, in real time, at a sampling rate of 32 K sample/s. (a) input waveform; (b) output waveform ($h=.95$, $\delta=16$).

Design Example 2:

A 32-K BIT/S ADAPTIVE DELTAMODULATION (ADM) CODEC:

The block diagram of the ADM codec is shown in Figure (5). The selected adaptation for the steep size is represented by

$$\delta(n) = \beta\, \delta(n-1) + \alpha(n)\, \delta_o\ ; \qquad (11)$$

where, β equals $1-\varepsilon^2$ with $\varepsilon^2 \to 0$; and $\alpha(n) = 1$, if the last three bits are of equal sign and equals 0 otherwise.

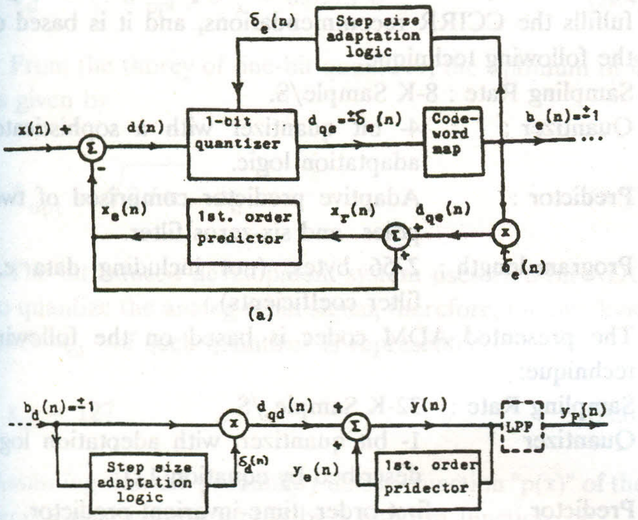

Figure 5. Block diagram of ADM codec; (a) encoder, (b) decoder.

Thus, the step size decays by a factor of "β" every sampling period, and increases by an amount of δ_o if $\alpha(n) = 1$. The time constant "τ" for the step size decay is given by

$$\tau = [f_s \log_e (1/\beta)]^{-1} \qquad (12)$$

where, f_s is the sampling frequency.

The code system is described by the following equations:

ENCODER

$$x_e(0) = 1 \qquad (13)$$
$$\{b_e(-3), b_e(-2), b_e(-1)\} = \{1,-1,1\} \qquad (14)$$
$$\delta_e(0) = 1 \qquad (15)$$
$$d(n) = x(n) - x_e(n) \qquad (16)$$
$$b_e(n) = \text{Sign } d(n) \qquad (17)$$
$$d_{eq}(n) = b_e(n) * \delta_e(n) \qquad (18)$$
$$x_r(n) = x_e(n) + d_{qe}(n) \qquad (19)$$
$$x_e(n+1) = x_r(n) * h \qquad (20)$$
$$\delta_e(n+1) = \beta * \delta_e(n) + \alpha_e(n) * \delta_o \qquad (21)$$

where,

$$\alpha_e(n) = 1 \text{ if } b_e(n-2) = b_e(n-1) = b_e(n) \qquad (22)$$
$$= 0 \text{ otherwise}$$

Decoder

$$y_e(0) = 1 \qquad (23)$$
$$\{d_d(-3), b_d(-2) b_d(-1)\} = \{1,-1,1\} \qquad (24)$$
$$\delta_q(0) = 1 \qquad (25)$$
$$d_{qd}(n) = b_d(n) * \delta_d(n) \qquad (26)$$
$$y(n) = d_{qd}(n) + y_e(n) \qquad (27)$$
$$y_e(n+1) = y(n) * h \qquad (28)$$
$$\delta_d(n+1) = \beta * \delta_d(n) + \alpha_d(n) * \delta_o \qquad (29)$$

Where

$$\alpha_d(n) = 1 \text{ if } d_d(n-2) = d_d(n-1) = b_d(n) \qquad (30)$$
$$= 0 \text{ Otherwise.}$$

SELECTING THE VALUES OF "δ_o" and "β":

A LDM codec of optimum step size δ_{opt}, can follow a change in the input signal from o to x_{ol} in "n" steps given by

$$n = \text{Int}\, [x_{ol}/\delta_{opt}] + 1 \qquad (31)$$

Thus from (A.6) and (A.9) of appendix (A) we have

$$n = 8 \qquad (32)$$

An ADM codec, based on (11), can get comparative performance if

$$\sum_{m=1}^{n} \sum_{i=o}^{m-1} \beta^i \delta_o \geq n\, \delta_{opt} \qquad (33\text{-a})$$

if β^{n-1} is close to 1, then (33-a) can be approximated by

$$\sum_{m=1}^{n} m\, \delta_o \geq n\ \ \delta_{opt} \qquad (33\text{-b})$$

from which we get

$$\delta_o \geq 4 \qquad (34)$$

the lowest integer value is to be selected, i.e. $\delta_o = 4$.

The reported typical value of "τ" is 5ms. The program was run using different values for "τ" that are close to 5 ms. It was found that, for $\tau = 2$ ms, the output waveforms are very illustrative for the adaptation process, the corresponding value of "β" is 0.9845.

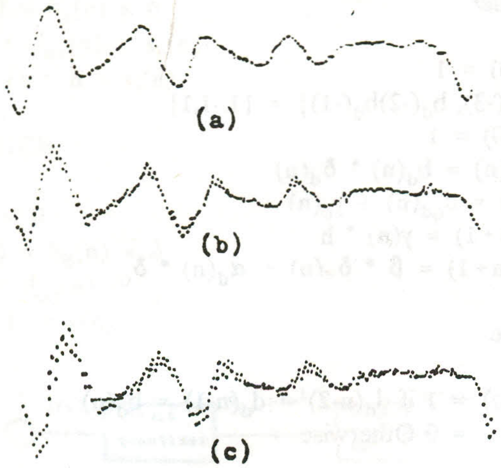

Figure 6. Results of ADM Codec, Implemented by a Basic program run on ZX-Spectrum microcomputer; (a) Input waveform, sampled at a rate 32K sample/sec; (b) Output waveform for $h=0.95$, $\beta=0.9844$, $\delta_o=2$; (c) Output waveform for $h=0.95$, $\beta=0.9844$, $\delta_o=4$;

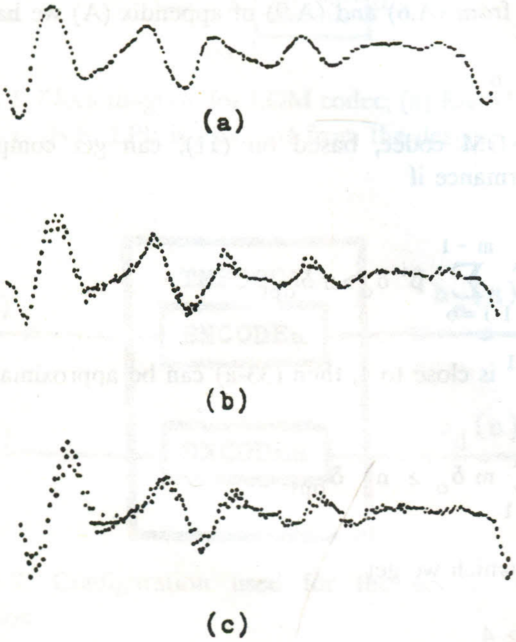

Figure 7. Results of ADM codec algorithm, implemented by TMS32010 DSP, in real-time, at sampling rate of 32K sample/sec; (a) Input waveform; (b) Output waveform for $h=0.95$, $\beta=0.9844$, $\delta_o=2$; (c) Output waveform for $h=0.95$, $\beta=0.9844$, $\delta_o=4$.

Figure (6) shows the results obtained from a non-realtime BASIC program while Figure (7) shows the results of the real-time program that is implemented by the TMS32010 DSP. The two results are very close which confirm correct implantation of the algorithm, they also illustrate the effect of δ_o on the output waveform.

A 32-k bit /S ADPCM codec was presented in [9], it fulfills the CCIRR recommendations, and it is based on the following technique:

Sampling Rate : 8-K Sample/S.

Quantizer : 4- bit quantizer with a sophisticated adaptation logic.

Predictor : Adaptive predictor comprised of two-poles and six-zeros filter.

Program length : 2756 bytes, (not including data e.g. filter coefficients).

The presented ADM codec is based on the following technique:

Sampling Rate : 32-K Sample /S.

Quantizer 1- bit quantizer, with adeptation logic described by equation(11).

Predictor : first order, time-invariant predictor.

Program length: 258 bytes only (including data).

It is clear that the presented ADM Codec algorithm is very simple, this advantage is due to the over sampling technique, which increases the correlation between adjacent samples. The development system offered this facility since it is designed such that it can Control/Monitor the operation of the TMS32010 while it is processing an input signal sampled at a rate of 32-K Samples/S.

Appendix (A):

NOTE ON OPTIMUM VALUES OF "H" AND :δ":

From the theory of DPCM of first order predictor, the variance of the difference signal "σ_d^2" is given by

$$\sigma^{d2} = (1 + h^2 - 2\,\rho_1\, h)\,\sigma^2_x \tag{A.1}$$

where σ^2_x is the variance of input signal, and ρ_1 is the adjacent sample normalized correlation function.

Setting $\partial\sigma^2_d / \partial h = 0$ yields the optimum "h_{opt}"

$$h_{opt} = \rho_1 \tag{A.2}$$

The value of ρ_1 approachs 1 as the sampling rate is increased, the selected value for ρ_1 is 0.95, i.e.

$$\rho_1 = .95 = h_{opt} = h \quad (A.3)$$

From (A.1) and (A.3) we get.

$$\sigma_d^2 = (1-h^2_{opt})\,\sigma^2_x = 0.0975\,\sigma^2_x \quad (A.4)$$

From the thorey of one-bit quantizer, the optimum of δ is given by

$$\delta_{opt} = \sqrt{2/\pi}\quad \sigma_d \quad (A.5)$$

The introduced development system uses an 8-bit ADC to quantize the analog input signal, therefore, the overload level "x_{ol} " of such quantizer is represented by

$$x_{ol} = 127 \quad (A.6)$$

Assuming that the probability density function "p(x)" of the input signal is represented by a uniform function, i.e.

$$p(x) = \frac{1}{2\,x_{ol}}\ ;\ -x_{ol} \le x \le x_{ol}$$

$$= 0 \quad \text{, otherwise} \quad (A.7)$$

Then

$$\sigma_x^2 = \int_{-x_{ol}}^{x_{ol}} x^2 p(x)dx = \frac{x_{ol}^2}{3} \quad (A.8)$$

From (A.4) to (A.8)

$$\delta_{opt}\ (\text{for uniform pdf}) \approx 18 \quad (A.9)$$

CONCLUSION:

Two different DM codecs were implemented by the TMS32010 DSP. A proposed development system was used in developing the programs of the two codecs. The system successfully controlled the operation of the TMS32010, and monitored the data and the results.

The presented design examples demonstrate the capabilities of the proposed development system, explain the development procedure and the recommended layout of the program to be developed.

It is concluded that the introduced development system is a useful, economical, simple and available tool for students and designers working in the field of digital signal processing.

REFERENCES

[1] M. Galal A. Fahmy, "Signal Processing in Electronic Devices", *Ph.D. Thesis*, Electrical Engineering Department, Faculty of Engineering, Alexandria University, 1990.

[2] TMS32010 User's Guide, Texas Instruments, 1985.

[3] J.E. Abate, "Linear and Adaptive Detlamodulation", *Proc. IEEE*, Vol. 55, pp. 298-308, March 1967.

[4] J.A. Greefkes and K. Reimens, "Code Modulation with Digitally Controlled Companding for Speech Transmission", *Philips Tech. Rev.*, pp. 335-353, 1970.

[5] N.S. Jayant (ed.), *Waveform Quantization and Coding*, IEEE press, New York, 1976.

[6] A.L. Auger, M.W. Glancy, M.M. Goutmann and A.L. Kirsch, "The Space shuttle space terminal Deltamodulation system", *IEEE Trans. on Communications*, vol. 26, pp. 1660-1670, November 1978.

[7] C.K. Un and H.S. Lee, "A Study of the Compparative Performance of Adaptive Deltamodulation Systems", *IEEE Trans. on Communications*, vol. 28, pp. 96-101, January 1980.

[8] N.S. Jayant, Peter Noll, *Digital Coding of Waveforms*, Prentice-Hall, 1984.

[9] Jay Reimer, Mike McMahan and Masoud Arjamand, "32-K bit/s ADPCM with TMS32010". Digital Signal Processing Applications, Texas Instruments, 1986.