CHAPTER-II

Processing of digital signals in (+1, -1) binary system

Digital signal processing in (+1, -1) system has been studied in detail. Effects of number representation on quantization and truncation or rounding have been discussed. First order and second order filters have been investigated. Differential pulse code modulation method of coding is being studied. It is found that this system is well suited for VLSI design. The hardware realisation of different components of DPCM is easy and straight. The positive and negative signals can be processed in a unified way. It gives better performance in comparison of two's complement representation of conventional binary system. The effects of limit cycles, stability and round off noise in first and second order filter have been discussed too. In digital signal processing the fractional signed numbers are required. Fixed point arithmetic numbers are employed. In fixed point arithmetic two's complements representation are used for signed fractional numbers. Two's complement number suffers due to some drawbacks. To overcome the drawbacks, Pekmestzi[10] pointed out that in digital signal processes, (+1, -1) binary system is very usefull and suitable in VLSI circuit design. In digital signal processing, realisation of digital filters of 1st and 2nd order either recursive or non recursive are important. In the design of digital filters finite word length including round off noise, coefficient quantization noise, limit cycles and stability of filter design are essential.
Quantization effects in digital filters have been extensively studied for different type of digital filters [45-47]. Differential pulse code modulation is very useful in digitizing the signals. The use of DPCM in digital filter has been extensively studied by various workers [48-55] in both recursive and non-recursive type. Linear predictive coding has been studied in (+1, -1) system [56].

In present paper we are mainly interested in digital signal processing using (+1, -1) binary system. Digital signal or sequences in this system is related to signum function which is related to unit step function. The quantization characteristic has been discussed. The paper contains rounding and truncation effects. The limit cycle effects and the effect on stability due to rounding and quantization on first and second order recursive digital filter have been studied. The results have been compared with two's complement representation of conventional binary system. Effect of prediction coefficient on mean square error is presented. Finally, differential pulse code modulation has been presented for digitizing signals.

2.1 Digital Sequence:

The unit sample sequence in (+1, -1) is defined by signum function.

\[
\text{Sgn}(n) = \begin{cases} 
+1, & n > 0 \\
-1, & n < 0 
\end{cases} 
\]  \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (1)

The unit sample sequence is shown as in fig. (1a) and (1b).

A sequence of number \( x \) can be written as

\[
x = \begin{cases} 
{x(n)} & \text{for } n > 0 \\
{-x(n)} & \text{for } n < 0 
\end{cases} 
\]  \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (2)\]
where \( x(n) \) is the \( n^{th} \) sample of sequence. The product and sum of sequence \( x \) and \( y \) are defined as

\[
x \cdot y = \{ (\pm x(n)) (\pm y(n)) \} \quad \ldots \ldots (3)
\]

or, \( x \cdot y = \{ + x(n) \ y(n) \} \) or \( \{ + x(n) - y(n) \} \)

or \( \{ - x(n) \ y(n) \} \) and or \( \{ - x(n) - y(n) \} \) \ldots \ldots (4)

\[
x + y = \{ + x(n) + y(n) \}, \quad \text{or} \quad \{ - x(n) + y(n) \}
\]

or \( \{ + x(n) + y(n) \} \) or \( \{ - x(n) + y(n) \} \) \ldots \ldots (5)

Multiplication of a sequence \( x \) by a number \( \alpha \) is defined as

\[
\alpha \cdot x = \{ \alpha (\pm x(n)) \} \quad \ldots \ldots (6)
\]

Any arbitrary sequence in \((+1, -1)\) system is defined by

\[
x(n) = \sum_{k=-\infty}^{\infty} x(k) \ Sgn(n-k) \quad \ldots \ldots (7)
\]

Where \( \text{Sgn}(n-k) \) is delayed unit sample.

The unit sample response in this system is defined by

\[
h(n) = \begin{cases} 
  a^n & \text{for } n > 0 \\
  -a^n & \text{for } n < 0 
\end{cases} \quad \ldots \ldots (8)
\]

or equivalently, by \( h(n) = a^n \ Sgn(n) \)

The stability of linear time invariant system is computed by the sum.

\[
S = \sum_{k=-\infty}^{\infty} |h(k)| = \sum_{k=-\infty}^{\infty} |\pm a|^k \quad \ldots \ldots (9)
\]

When \( |\pm a| < 1 \), the geometric series for \( S \) is given by

\[
S = \frac{1}{1 - |(\pm a)|} \quad \ldots \ldots (10)
\]

The system is stable when \( |(\pm a)| < 1 \). Causal sequence is not possible in the system.
Quantizer

Quantizer characteristic in (+1, -1) system is shown in Fig. (2). Here, b is the number of bits used for digital representation of sampled signals. The quantizer saturates output either at 1 - 2^{-b} or - (1 - 2^{-b}). Δ is the step size and is given by 2^{-b+1}. This quantizer characteristic if compared with 2's complement quantizer characteristic, one finds that in 2's complement representation, b+1 bits left portion of decimal point represent the sign bit and right portion of decimal point represents the magnitude of the number but in (+1, -1) system sign bits are not required. The range of saturation is 1 - 2^{-b+1} and -1 in conventional binary system and the step size is 2^{-b}.

Thus, the saturation points are not symmetrical in positive and negative side. The negative side contains more steps. At the same time the quantizer characteristic is symmetric in our system. The step size in our system is double that of two's complement quantizer. So, the number of steps are halved in our system.

Signal to noise ratio (SNR) for this system is given by (3x2^{2b}) \sigma_f^2. Whereas in 2's complement, it is given by (12x2^{2b})\sigma_f^2. In (0, 1) system, if the sign of unquantized signal is unknown, the variance is given by 2^{-2b}/3. For 1st order filter, noise variance due to quantization is given by \sigma_f^2 = \frac{2^{-2b}}{3(1 - c^2)} = \frac{2^{-2b}}{3x2\epsilon} = \frac{2^{-2b}}{6\epsilon}

where, c is the amplitude.

If the pole is very close to the unit circle i.e. c = 1 - \epsilon \text{ where } \epsilon << 1

c^2 = (1-\epsilon)(1-\epsilon) = 1 + \epsilon^2 - 2\epsilon = 1 - \epsilon(2 - \epsilon) \text{ so } c^2 \approx 2\epsilon \text{ , } \epsilon \text{ is the distance of the pole from the unit circle.}
In conventional binary system of $\sigma^2 = 2^{2h}/2 - c^2 \equiv 2^{2h}/24\varepsilon$ and one extra bit for sign is required in two's complement representation.

In our system variance of quantization noise is $2^{2h}/3$ whereas in 2's complement representation, it is $2^{2h}/12$.

2.2 TRUNCATION AND ROUNDING:

The process of the truncation of numbers in this binary system can be explained by an example. Let us take fractional number (0.84375) represented by (.1111) where complement of 1 is $\bar{1}$ which is equal to -1. The rightmost bit in our example is 1 and assuming it to be an extra bit, it has to be truncated. After truncation the number becomes. 1111 = 13/16 = 0.8125. The error of truncation thereby is 1/32. In case we have the last bit as $\bar{1}$ for a number say (.1111) = 29/32 = 0.90625 then on truncating, we obtain the number as 30/32 = 0.9375. One observes that depending upon the number to be truncated, the value after truncation may be greater or smaller than the original number.

Next, we consider the process of rounding off a number in (+1, -1) representation. A number which is to be rounded, the last bit is removed and necessary quantity is added to the number. Using the process of rounding repeatedly, one can round off any number of digits. In the case of (+1, -1) system, a full adder is necessary therefore we need three digits for addition. The rounding process can be achieved in the following manner.
Let us take an example: Let the number to be rounded off be (1 1 1 1 1). The value of this number is (0.84375). The third binary number we take as the complement of the last bit (whose value in our case is 1/32) and equals -1/32 which will be written as (1 1 1 1 1). Now we get on adding the three numbers

\[
\begin{array}{c}
.11111 \\
1 \\
1111 \\
1.11111
\end{array}
\]

→ 3rd number
→ Carry number
→ Original number

The result of the addition has an leftmost digit (1). The next to leftmost (1) is replaced by the bit (1) and one gets the sum as .1 1 1 1 1 which is equal to 27/32. Since we have used an additional third number, we remove that value from the result and then we obtain 27/32 - 1/32 = 26/32 = 13/16, which is equal to (1 1 1 1). Thus, we find that truncation and rounding produce the same result.

In two's complement representation the truncation and rounding produces different results. The magnitude of the number after truncation is always less than the magnitude before truncation. In our system, the magnitude of the number after truncation is either smaller or greater than the number before truncation. The rounding and truncation produce the same result.

The truncation error in two's complement for positive and negative numbers are given by the range \(-2^b \leq E_T \leq 0\) and \(0 \leq E_T \leq 2^b\) respectively. The rounding error for positive as well as negative numbers is given by \(\frac{1}{2} \leq E_R \leq \frac{1}{2} \cdot 2^b\).
In our system the rounding or truncation error is given by the relation

\[-2^b < E_R < 2^b.\]

**LIMIT CYCLE EFFECTS:**

The limit cycle behaviour for first order filter can be realised as

\[p(n) = Q(n) + CP(n-1) \] \hspace{1cm} \ldots (11)

Let us assume that \( C = 1/8 \) and the register length for storing the coefficient \( C \), the input \( Q(n) \) and filter node variable \( P(n-1) \) are three bits respectively. Because of finite length registers, the product \( CP(n-1) \) is rounded or truncated to three bits before addition to \( Q(n) \). Let us consider the rounding or truncation of the product. The actual output \( R(n) \) satisfies the nonlinear difference equation.

\[ R(n) = Q[CR(n-1)] + Q(n) \] \hspace{1cm} \ldots (12)

where \( Q[\cdot] \) represents the rounding or truncation operation.

Now \( C = 1/8 = .1 \ 1 \ 1 \)

The amplitude of unit sample \( = 7/8 = .1 \ 1 \ 1 \)

From equation (12) for \( n = 0 \)

\( R(0) = .1 \ 1 \ 1 \) and for \( n = 1 \)

\[ R(1) = CR(0) \text{ when the input i.e. } Q(n) = 0 \]

\[ = 7/64 = .1 \ 1 \ 1 \ 1 \ 1 \ 1 \]

After truncation \( R(1) = .1 \ 1 \ 1 = 1/8 \)

Now, for \( n = 2 \)

\[ R(2) = CR(1) = 1/64 = .1 \ 1 \ 1 \ 1 \ 1 \ 1 \]
After truncation

\[ R(2) = \frac{1}{1} = \frac{1}{8} \]

Thus for \( n \geq 2 \), all the values of \( R \) are the same and equal to \( 1/8 \)

For \( C = -1/8 \) in similar fashion,

we get

\[ R(1) = -1/8 \]
\[ R(2) = +1/8 \]
\[ R(3) = -1/8 \]

Thus, periodic steady state solution between \(+1/8\) and \(-1/8\) is obtained. for \( C = -1/8 \)

Following the method of Jackson [12] the condition for dead bands is given by

\[ |R(n - 1)| \leq \frac{2^{-b}}{1 - |C|} \]

\[ ............ \ (13) \]

For \( |C| = 1/8 \), \( R(n-1) \leq 8/7 \times 2^{-b} \approx 2^{-b} \)

Thus \( |C| = 1/8 \) gives the correct value of dead band.

In two's complement representation, the register length is limited to four bits. The method of rounding determines the limit cycles and requires lengthy process. For \( C = 1/2 \), the correct value for the dead band is achieved. Thus, we find that limit cycle calculation in \((+1, -1)\) representation is easier in comparison of two's complement representation. In \((+1, -1)\) system, the dead bands are achieved using truncation and less number of bits and less number of \( n \) in comparison of conventional binary system.

For second order filter, the non linear difference equation is given by

\[ R(n) = Q(n) + Q[A Q(n-1)] + Q[B Q(n-2)] - Q[CR(n-1)] - Q[DR(n-2)] \]

\[ ............ \ (14) \]
In case of limit cycle

\[ R(n) = -Q[CR(n-1) - Q[DR(n-2)] \]

\[ \text{.......... (15)} \]

Using truncation relation

\[ |Q[DR(n-2)] - DR(n-2)| \leq 2^b \]

\[ \text{.......... (16)} \]

Since \( Q[DR(n-2)] = R(n-2) \)

So \( |R(n-2) - DR(n-2)| \leq 2^b \)

i.e. \( |R(n-2)| \leq \frac{2^{-b}}{1-|D|} \)

\[ \text{.......... (17)} \]

The value of \( D \) should be such that poles of the system are on the unit circle.

**Effect of quantization error on stability**

Step size between quantized level is given by

\[ \Delta = 2^{b+1} \]

Where \( b \) is the number of bits for coefficient \( C \).

For stable filter \(-1 < C < 1\)

If \( \varepsilon = 1 - C \) is the distance of the pole to the unit circle.

the smallest value of \( \varepsilon = 2^{b+1} \) when it is either truncated or rounded.

Now \( \log_{10} \varepsilon = \log_{10}(2^{-(b-1)}) = -(b-1) \log_{10}2 \)

\[ \text{.......... (18)} \]

\[ \frac{\log_{10}\varepsilon}{\log_{10}2} = -(b - 1) \]

\[ \text{.......... (19)} \]

So, \( b = \frac{-\log_{10}(1-C)}{\log_{10}2} + 1 \)

\[ \text{.......... (20)} \]

In case of two's complement representation in \((0, 1)\) system for truncation.
\[ b = - \frac{\log_{10}(1 - C)}{\log_{10} 2} - 2 \quad \text{and for rounding,} \quad b = - \frac{\log_{10}(1 - C)}{\log_{10} 2} - 1 \quad \ldots \ldots \quad (21) \]

For second order filter, the difference eqn. is given by

\[ Y_k = A_1 y_{k-1} + A_2 y_{k-2} + B_1 x_k + B_2 x_{k-1} \quad \ldots \ldots \quad (22) \]

Let the poles be \( p_1 \) and \( p_2 \). \( A_1, A_2, B_1 \) and \( B_2 \) are coefficients with respect to pole position.

Sensitivity of coefficient is given by the relation.

\[ \frac{\partial p_1}{\partial A_1} = \frac{p_1}{p_1 - p_2} \quad \frac{\partial p_2}{\partial A_1} = \frac{p_2}{p_2 - p_1}, \quad \frac{\partial p_2}{\partial A_2} = \frac{1}{p_2 - p_1} \]

and \[ \frac{\partial p_1}{\partial A_2} = \frac{1}{p_1 - p_2} \quad \ldots \ldots \quad (23) \]

Total variation in pole position is given by

\[ \Delta p_1 = \frac{1}{p_1 - p_2} (p_1 \Delta A_1 + \Delta A_2) \quad \ldots \ldots \quad (24) \]

and

\[ \Delta p_2 = \frac{1}{p_2 - p_1} (p_2 \Delta A_1 + \Delta A_2) \quad \ldots \ldots \quad (25) \]

For stabilities \(-2 < A_1 < 2 \) and \(-1 < A_2 < 1 \).

For truncation or rounding,

\[ \Delta A_1 = \Delta_{1/2} = \frac{2^{-\Delta+2}}{2} = 2^{b+1} \quad \text{where} \quad \Delta_1 = 2^{b+2} \quad \ldots \ldots \quad (26) \]

\[ \Delta A_2 = \Delta_{2/4} = \frac{2^{-\Delta+1}}{4} = 2^{b-1} \quad \text{where} \quad \Delta_2 = 2^{b+1} \quad \ldots \ldots \quad (27) \]

From (24), (25), (26) and (27) and also using
\( p_1 = 0.98 \) and \( p_2 = 0.94 \), we get

\[
\Delta p_1 = \frac{1}{0.04} (0.98 \times 2 + 0.5) 2^b
\]

\[
= \frac{1}{0.04} (1.96 + 0.5) 2^b
\]

\[
= \frac{2.46}{0.04} \times 2^b
\]

........ (28)

or,

\[
\Delta p_1 = 1 - p_1 = 0.02 = \frac{2.46}{0.04} \times 2^b
\]

........ (29)

so,

\[
2^b = \frac{0.04 \times 0.02}{2.46}
\]

........ (30)

\[
2^b = \frac{2.46}{0.04 \times 0.02} = \frac{24600}{0.0008} = \frac{24600}{8} = 3075
\]

........ (31)

\( b = 12 \) bits for the minimum register length.

In case of 2's complement number for rounding, we get

\[
\Delta p_1 = \frac{1}{0.04} (0.98 + 0.5) 2^b
\]

........ (32)

\[
0.02 = \frac{1.48}{0.04} \times 2^b
\]

\[
2^b = \frac{1.48}{0.0008} = \frac{14800}{8} = 1850
\]

........ (33)

\( b = 11 \) bits for the minimum register length.

and for truncation, we get

\[
\Delta A_1 = \frac{1}{0.04} (2 \times 0.98 \times 1.0) 2^b
\]

........ (34)

\[
0.02 = 74 \times 2^b \quad \text{or} \quad 2^b = 3700
\]

........ (35)
b = 12 bits for minimum register length.

Thus, minimum register length in both the systems are approximately the same.

2.3 MULTIPLIER:

Let us take two fractional numbers A and B. In (+1, -1) system these numbers can be written as

\[ A = \sum_{k=1}^{m} (a_k)2^{-k} \]
\[ B = \sum_{l=1}^{n} (b_l)2^{-l} \]

where \( a_k, b_l = +1 \) or \(-1 \) and \( m \) and \( n \) are the number of bits. To clarify the multiplication process, we take an example.

Let \( A = (1 \ 1 \ 1 \ 1) = 1/8 \) and \( B = (1 \ 1 \ 1 \ 1) = -5/8 \)

then \( A \times B = -5/64 = 0. \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \). The multiplication process can be understood from figure (3).

In figure (3), the value of \( C_3 \ C_2 \ C_1 \) is taken as \( 1 \ 1 \ 1 \) which in general will be \( C_n \ C_{n-1} \)

\( \ldots \ldots \ C_2 \ C_1 \) as \( \ 1 \ \ldots \ldots \ldots \ 1 \ 1 \). The values of \( S_3 \ S_2 \ S_1 \) is complement of \( C_3 \ C_2 \ C_1 \) and hence it is \( \ 1 \ 1 \ 1 \) and in general, it will be \( S_n \ S_{n-1} \ldots \ldots \ldots \ S_2 \ S_1 \) as \( \ 1 \ 1 \ldots \ldots \ldots \ 1 \ 1 \).

The input sum \((S_3 \ S_2 \ S_1)\) and input carry \((C_3 \ C_2 \ C_1)\) are introduced in the partial product because in this system only full addition is possible. Introduction of input sums and input carries do not affect the result because they are of complementary nature.
C₁ and S₁ are added to the partial product marked as a rectangle to produce a sum S and a carry C in the figure (3). Since S has nothing else to be added, it is carried towards final total sum as T₁. The generated carry C and the input sum S₂ are added to the partial product T₁ to give a generated carry and a generated sum. These generated carry and sum will be subsequently added to partial products. The last generated carry is however, taken as the generated sum for the diagonally adjacent partial product as shown by arrow.

THE PREDICTION FILTER:

The first order prediction filter is given by the expression

\[ P(n) = Q(n) + C \times P(n-1) \]  

...... (36)

where Q(n) is real data and C.P. (n-1) is real data with additional bits.

Let us take an example

Let \( Q(n) = -3/16 = 0.1111 \)

and \( C \times P(n-1) = +59/128 = 0.1111111 \)

substituting these values in equation (19), we get

\[
\begin{align*}
\text{Q(n)} & = 0.1111 \\
\text{CP (n-1)} & = 0.1111111 \\
\text{Sum} & = 1.1111111 \\
\text{So, } P(n) & = 0.1111 = +5/16 \\
\text{The rounding error} & = 111 = -5/128
\end{align*}
\]
In CP(n-1) the bits to be truncated are \(1 \bar{1} \bar{1} = 3/128\). The MSB of truncated value is taken as a carry bit for the sum and the addition is done. In the result, if the carry bit and the MSB are opposite, the MSB is replaced by carry bit and the carry is ignored. This is due to fact that when two fractional numbers are added, an additional bit is always generated. If the result is in the same range as the two numbers, the last two bits \(S_0\) and \(S_1\) will be of opposite sign and the sum is contracted by replacing \(S_0\) by \(S_1\) because of the relation

\[(S_0) + (S_1) 2^{-1} = (S_0) 2^{-1} \text{ when } (S_0) = -(S_1).\]

The carry bit, used for the sum does not produce any error in the result which can be explained as.

The truncated value \(1 \bar{1} \bar{1} = 3/128\) can be written as \(1/16-5/128\), which implies the addition of \(1/16\) as carry in the addition does not change the value of the rounding error.

Next, we consider the second order prediction filter. The second order filter is given by the expression

\[
p(n) = Q(n) + AQ(n-1) + BQ(n-2)\]

\[
- CP(n-1) - DP(n-2) \quad \ldots \quad (37)
\]

We take an example to illustrate the above filter,

Let \(Q(n) = 0.1 \bar{1} \bar{1} \bar{1} = 1/16,\)

\(AQ(-1) = 0.1 \bar{1} \bar{1} \bar{1} \bar{1} \bar{1} = 3/16 + 1/128\)
\[ BQ\ (n-1) = \begin{array}{cccc}
0 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 \\
\end{array} = 5/16 + 1/64 \]

\[ CP(n-2) = \begin{array}{cccc}
0 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 \\
\end{array} = -1/16 - 3/128 \]

\[ DP\ (n-2) = \begin{array}{cccc}
0 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 \\
\end{array} = -3/16 - 1/64 \]

Using the same method as done in first order filter, we obtain

\[ p(n) = \begin{array}{cccc}
0 & 1 & 1 & 1 \\
\end{array} = 5/16 \]

and the truncated value = -1/64 which equals the round off error.

In a similar way the third order filter can be written as

\[ p(n) = Q(n) + AQ\ (n-1) + BQ\ (n-2) + CQ\ (n-3) \]

\[ -DP\ (n-1) - EP\ (n-1) - FP(n-3) \]

\[ \text{.......... (38)} \]

Higher order filters can be similarly generated.

Next, we discuss the evaluation of the prediction coefficients by using the Lavinson and Durbin algorithm.

The autocorrelation function for sample \( (x_a) \) is given by the relation

\[ \phi(n) = (1/N \sum_{i=1}^{N-1} x_i x_{i+1} \text{ for } n = 0, 1, 2, \text{.........p} \]

\[ \text{.......... (39)} \]

For minimum mean square error (MSE) with respect to the prediction coefficient \( (a_i) \), we have the relation for the set of linear equations

\[ \sum_{i=1}^{p} a_i \phi(i - j) = \phi(j) \text{ for } j = 1, 2, 3, \text{.........p} \]

\[ \text{.......... (40)} \]

Equation (23) can be written in matrix form as

\[ \Phi a = \phi \]

\[ \text{.......... (41)} \]

Where \( \Phi \) is a \( p \times p \) matrix with elements
\[ \phi_{ij} = \phi(i-j), \] is a \((p \times 1)\) column vector of prediction coefficients and \(\phi\) is a \((p \times 1)\) column vector with elements \(\phi(i)\), for \(i = 0, 1, 2, 3, \ldots, p\).

The matrix \(\Phi\) has equal diagonal elements.

In this system the samples \(\{x_n\}\) are always odd fraction numbers. For a 5 bit quantizer the input to the quantizer will be \(2/32, 4/32, 6/32, \ldots, 30/32\). Thus, there will be 15 samples. The values of the prediction coefficients from equation (8). The values of the prediction coefficients and the results are shown in figure (4).

From the figure (4), we observe that prediction coefficients sharply changes from a maximum value \(a_{11} = 0.4\) to the minimum value \(a_{22} = -0.167\) when the order of the filter is changed from 1 to 2. However for \(p = 2\) to 9 the values of filter coefficients increase exponentially and finally for \(p = 9\) to 12, the filter coefficients show some variations. We find that the changes in the values of prediction coefficients with order of prediction filter show a damped oscillatory behaviour.

The mean square error for first order filter is given by

\[ e_1 = \phi(0) (1 - a_{11}^2) \]

for \(m\)th order filter, mean square error is given by

\[ e_m = e_{m-1} (1 - a_{mm}^2) \quad \text{and} \]

\[ m = 2, 3, \ldots, p \]

\(a_{mm}\) is the prediction coefficients of \(m\)th order filter. From equation (22), we can easily compute the values of MSE for different order of filters. Variation of MSE with \(p\) has been shown in figure (5). The values of mean square error decreases with increase of \(p\). Firstly, it decreases rapidly and then slowly and finally it
becomes almost constant. Thus the necessary and sufficient condition \( (e_m \leq e_{m-1} \leq \ldots \leq e_1) \) and \( (|a_{nm}| < 1) \) is satisfied in this system.

2.4 DIFFERENTIAL PULSE CODE MODULATION (DPCM):

The current sample for the source can be obtained by weighted linear combination of past samples. Hence, we can write

\[
y_q(n) = \sum_{i=1}^{r} a_i y_{n-i}
\]

where \( y_q(n) \) is the predicted value of \( y_n \) which is the current sample. \( r \) is the number of samples and \( a_i \) represents the coefficients of the past samples. The equation (26) can be rewritten as

\[
y_q(n) = a_1 y_{n-1} + a_2 y_{n-2} + a_3 y_{n-3} + \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots
\]

\( a_1 y_{n-1} \) represents first order prediction filter, \( a_2 y_{n-2} \) represents second order filter and so on. The prediction filter of different orders have been discussed earlier.

In order to explain the concept of the DPCM we take an example. Let the four samples be \(-2/16, +2/16, +4/16\) and \(+6/16\). The quantized values of these are \(-3/16, +3/16, +5/16\), and \(+7/16\) respectively. The predictor output of these samples is given by using equation (44).

\[
y_q(n) = 3/16 + (-3/16) + 5/16 + 7/16 = 13/16 = .1111
\]

whose truncated value = \(-3/128\).

The current value of the incoming signal \( y_n = +8/16 \). The difference of the input to the quantizer and the predicted value is given by
\[ e_n = y_q (n) - y_n + q_e (n) \]  \hspace{1cm} (45)

where \( q_e (n) \) is the quantization error.

Substituting the values in the above equation (45), we get \( e_n = 6/16 \) when \( q_e(n) = 1/16 \).

The quantized value of error is given by the relation

\[ y'_n = y_q - e_q (n) + q_e (n) \]  \hspace{1cm} (46)

\[ = 13/16 - 7/16 + 1/16 \]

\[ = 7/16 \]

If we add the quantization noise to the prediction filter input, we obtain the current signal. Thus

\[ y'_n + q_r (n) = y_n = 8/16 \]  \hspace{1cm} (47)

Hence one finds that the quantization error is independent of the prediction filter employed.

The DPCM transmitter and receiver are shown in figure 6(a) and 6(b) respectively.

The average power of the message sequence of length \( N \) is given by

\[ P_m = \frac{1}{N} \sum_{n=0}^{N-1} y_n^2 \]  \hspace{1cm} (48)

The average power of the quantizing error of message sequence is however given by

\[ P_q = \frac{1}{N} \sum_{n=0}^{N-1} q_r^2(n) \]  \hspace{1cm} (49)

and the average power of the error sequence is given by
\[ P_e = (1/N) \sum_{n=0}^{N-1} e_n^2 \] 

........ (50)

The output signal to quantization noise ratio

\[ (\text{SNR})_0 = P_m / P_q \] 

........ (51)

The signal to quantizing noise ratio

\[ (\text{SNR})_q = P_e / P_q \] 

........ (52)

and the prediction gain is given by

\[ G_p = P_m / P_e \] 

........ (53)

The values of (SNR)_0 and G_p for different values of N are shown in table 1.

The results have been compared with (0,1) system [15]

2.5 CONCLUSIONS

Signal can be processed in efficient way in (+1,-1) binary system. Digital signals can be represented by using signum function. Sequences are evenly spaced on time scale in positive as well as negative direction. The amplitudes of sequences are positive as well as negative. Sign is inherent in this system. The quantizer is symmetric and step size is twice. Signal to noise ratio is small. Rounding and truncation are equivalent. Therefore, the limit cycle calculation becomes very simple. Dead bands are achieved in simple way and also rapidly. The number of bits to represent the coefficients for stabilities is less than two's complement representation for 1st order filter.
The proposed system is well suited for differential pulse code modulation. Each component of DPCM can be constructed in modular form. Positive and negative signals can be handled in a unified way. The internal symmetry of the system is reflected from the result that the truncation and rounding errors are equal. The internal symmetry of system results in a highly modular circuits. The unified circuit design leads to reduction in circuit complexity and delay time. The modularity of the circuit is yet another important factor for VLSI design [16,17]. The addition and multiplication for positive and negative signals can be performed with the help of a single basic cell. An array of binary adders is also possible. Carry-save type, the cellular array multipliers and the iterative multipliers can be constructed with the help of similar cells. The prediction gain in this system is very high and hence the average power of error sequence is small for a given baseband signal with fixed average power of message sequence. Thus, a prediction filter with minimum error power can be designed. The values of prediction coefficients are positive as well as negative in this system. The prediction filter is very sensitive upto third order and the predicted value decreases very sharply. The predicted values increases after third order. It first increases reapidly up to 6th order and then increases slowly up to 10th order. This type of behaviour of the filter is expected due to uniform quantization of positive and negative signals and ordering realisation of filter. The behaviour of filter after third order shows resemblance with conventional binary but the predicted values are different. Output signal to noise ratio increases with prediction filter coefficients.
Quantization noise ratio is found to be 0.25 for different number of samples. It is independent of filter coefficients. Thus, we conclude that the (+1, -1) is suitable for VLSI design and also for fast communication whenever positive and negative signals are transmitted simultaneously.
Recursive DPCM filter in (+1, -1) System

A recursive filter using differential pulse code modulation in (+1, -1) system has been proposed. Different parts of the filter have been discussed in detail. Results have been compared with the conventional binary system. The effect of round off error and limit cycles in the proposed filter have been analysed. In PCM each sample of the wave form is encoded independently of all the other samples. However, most source signals sampled at the Nyquist rate or faster exhibit significant correlation between successive samples. In other words, the average change in amplitude between successive samples is relatively small. Consequently, an encoding scheme that exploits the redundancy in the samples, will result in a lower bit rate for the source input. A relatively simple solution is to encode the difference between successive samples rather than, the samples themselves. Since differences between samples are expected to be smaller than the actual sample amplitudes, fewer bits are required to represent the difference. Differential Pulse Code Modulation (DPCM) has been extensively studied by the authors. The use of DPCM in non-recursive filter is being discussed by Goldstein and Liu. A recursive digital filter using DPCM is being studied by Thong and Liu.

Effect of finite word lengths in digital filters has been studied in detail [52, 47, 53, 61, 62]. Digital signal processing in (+1,-1) system is being discussed by Pekmestzi [10]. Linear predictive coding in (+1,-1) system is being discussed by Tiwari et al. [63].
In this paper, the structure of DPCM filter is being studied in detail. It contains the study of the different components of filter. Effect of round off noise and limit cycle is analysed. The results and hardwares have been compared with the conventional binary system. Hardware realisation of second order filter has been discussed. Advantages of filter is also presented.

2.6 RECURSIVE DPCM FILTER

The input analog signal \( x(t) \) is digitized into DPCM sequence \( \{v_n\} \) and quantization error \( q_n \). \( v_n \) is a \( b \) bit word. The generation of \( v_n \) with \( q_n \) can be explained in brief which is as follows.

The current sample for the source can be obtained by weighted linear combination of past samples

\[
X_q(n) = \sum_{i=1}^{r} a_i x_{n-i}
\]

Where \( r \) is the number of sample, \( a_i \)'s are the coefficients of past samples and \( x_q(n) \) is the predicted value of \( x(n) \) which is the current sample. The detail of prediction filter in \( (+1, -1) \) system has been discussed by authors [63, 64]. Fig. (7) shows encoding and processing of DPCM and fig. (8) shows decoding and Fig. (9) shows the quantizer characteristic. From the analysis of Fig. (7), it is obvious that

\[
d_n = x_n - x_q^n + q_n \quad \ldots \quad (54)
\]

\[
q_n = v_n - d_n \quad \ldots \quad (55)
\]

\( x_q^n \) is the predictor output.
\[ x_n' = x_n^q + q_n + v_n \] \hspace{1cm} \text{...... (56)}

where \( x_n' \) is the predictor input.

The current sample is obtained by adding quantisation noise to predictor input i.e.

\[ x_n = x_n' + q_n \] \hspace{1cm} \text{...... (57)}

Samples are related to the sequence \( v_n \) and \( q_n \) by the relation.

\[ x_n = x_{n-1} + \Delta 2^{b-2} v_n + q_n \] \hspace{1cm} \text{...... (58)}

where \( \Delta \) is the step size of the quantizer.

The input-output relationship of recursive DPCM filter is given by

\[ w_n = \sum_{i=0}^{N} a_i (v_{n-i} + q_{n-i}) - \sum_{i=1}^{N} b_i w_{n-i} \] \hspace{1cm} \text{...... (59)}

The sequence \( \{w_n\} \) and the output \( \{u_n\} \) is related recursively as

\[ u_n = w_n + u_{n-1} + q_{n-1} \] \hspace{1cm} \text{...... (60)}

In general, the recursive digital filter is defined by the following difference equation

\[ y_n = \sum_{i=0}^{N} a_i x_{n-i} + \sum_{i=1}^{N} b_i y_{n-i} \] \hspace{1cm} \text{...... (61)}

and hence,

\[ y_{n-1} = \sum_{i=0}^{N} a_i x_{n-1-i} + \sum_{i=1}^{N} b_i y_{n-1-i} \] \hspace{1cm} \text{...... (62)}

Subtracting eqn (62) from (61), we have

\[ y_n - y_{n-1} = a_i (x_{n-i} - x_{n-i-1}) - \sum_{i=1}^{N} b_i (y_{n-i} - y_{n-i-1} - q_{n-i-1}) \] \hspace{1cm} \text{...... (63)}

where \( q_{n-1} \) is the bias error introduced during subtraction
Substituting equation (58) into (63), we get

\[ y_n - y_{n-1} - q_{n-1} = \sum_{i=0}^{N} a_i q_{n-i} - \sum_{i=1}^{N} b_i q_{n-i-1} \]

\[ = \sum_{i=0}^{N} a_i \Delta 2^{b-2} v_{n-i} - \sum_{i=1}^{N} b_i (y_{n-i} - y_{n-i-1}) \]  \hspace{1cm} ...... (64)

Using equations (59) and (60) in equation (64), and multiplying by \( \Delta 2^{b-2} \), we get

\[ \Delta 2^{b-2} u_n - \Delta 2^{b-2} u_{n-1} - q_{n-1} = \sum_{i=0}^{N} a_i q_{n-i} - \sum_{i=1}^{N} b_i q_{n-i-1} \]

\[ = \sum_{i=0}^{N} a_i \Delta 2^{b-2} v_{n-i} - \sum_{i=1}^{N} b_i \Delta 2^{b-2} (u_{n-i} - u_{n-i-1}) \]  \hspace{1cm} .... (65)

where \( \Delta 2^{b-2} q_{n-1} = q_{n-1} \)

Comparing equations (11) and (12), we have

\[ y_n = \Delta 2^{b-2} u_n \]  \hspace{1cm} ...... (66)

The proposed scheme is divided into three parts. The first part converts analog signal \( x(t) \) into DPCM sequence \( v_n \) with quantization errors \( q_n \). Second part is digital processor which converts \( v_n \) into the sequence \( w_n \) and finally, the third part that produces the required output \( u_n \) which is related to \( y_n \).

The second order recursive DPCM can be represented by the equation

\[ A_n = \sum_{i=0}^{N} a_i (v_{n-i} + q_{n-i}) - \sum_{i=1}^{N} b_i w'_{n-i} + A_{n-1} - w'_{n-1} \]  \hspace{1cm} ..... (67)

where \( w'_{n} = A_{n} + r_{n} \)

\( r_{n} \) is the rounding error and \(-2^h < r_n < +2^h\)
Here, the input sample \( v_n \) is a \( b \)-1 bit word and the coefficient \( a \) is a \( c \) bit word. So, the product \( a_0 v_n, a_1 v_{n-1} \) and \( a_2 v_{n-2} \) will contain \((b+c-2)\) bit words. \((b+c-2)\) bit word is rounded or truncated to \( b \) bit word using DPCM-DPCM converter and then it is fed back to the arithmetic unit. Here, truncation and rounding are equivalent in our system. The block and respectively detailed diagram of second order recursive DPCM filter is shown in fig (3a) and (3-b). It contains multipliers, adders and DPCM-DPCM converter. Multipliers, adders and DPCM have been discussed by authors[62, 10, 63].

The \( \phi \) DPCM-DPCM converter is shown in fig. (10). The reconstructed output \( u_n \) due to second order recursive DPCM filter in (+1, -1) and conventional binary are shown in fig. (11) for 100 samples and 7 bits.

2.7 EFFECTS OF FINITE WORD LENGTH

1. Round off noise

From equations (67), we have

\[
\begin{align*}
\hat{w}_n &= r_n + a_0 v_n + a_1 v_{n-1} + a_2 v_{n-2} + \ldots + a_N v_{n-N} \\
&\quad + a_0 q_0 + a_1 q_{n-1} + a_2 q_{n-2} + \ldots + a_N q_{n-N} \\
&\quad - b_1 \hat{w}_{n-1} - b_2 \hat{w}_{n-2} - b_3 \hat{w}_{n-3} - \ldots - b_N \hat{w}_{n-N} - r_{n-1} \\
&= w_n - u_n - q_n - \sum_{k=1}^{N} b_k \hat{w}_{n-k} - r_{n-1} \\
&= u_n - u_{n-1} - q_n \\
\text{and} \quad v_n &= \frac{1}{\Delta^2_{b-1}} \{ x_n' - x_{n-1}' \} 
\end{align*}
\]

where \( x_n' \) is the approximate value of \( n \)th sample \( x_n \) i.e.
\[ x_n^* = x_n - \delta_n \] .......... (71)

where \( \delta_n \) is the random variable. Substituting equations (69) and (70) into (68), we get

\[
u_n - u_{n-1} - q_n + b_2 \{ u_{n-2} - u_{n-3} - q_{n-2} \} + b_N \{ u_{n-N} - u_{n-N-1} - q_{n-N} \} +
\]

\[
= r_n + \frac{1}{\Delta 2^{b-1}} \{ a_0 (x_n^* - x_{n-1}^*) + a_1 (x_{n-1}^* - x_{n-2}^*) + a_2 (x_{n-2}^* - x_{n-3}^*) + \ldots + a_N (x_{n-N}^* - x_{n-N-1}^*) \} + r_{n-1}.
\] ........ (72)

we define

\[ \text{Output error } e_n = u_n - u_n - q_n \] .......... (73)

Where \( q_n \) is the bias error.

Substituting equations (71) and (73) into eqn. (72) and solving, we obtain

\[
e_n + \sum_{i=1}^{N} b_i e_{n-i} - (e_{n-1} + \sum_{i=1}^{N} b_i e_{n-1-i}) + (u_n + \sum_{i=1}^{N} b_i u_{n-i}) - (u_{n-1} + \sum_{i=1}^{N} b_i u_{n-i-2}) -
\]

\[
q_n - \sum_{i=1}^{N} b_i q_{n-i} = r_n + \frac{1}{\Delta 2^{b-1}} \sum_{i=0}^{N} a_i x_{n-i} - \frac{1}{\Delta 2^{b-1}} \sum_{i=0}^{N} a_i x_{n-1-i} - \frac{1}{\Delta 2^{b-1}} \sum_{i=0}^{N} a_i \delta_{n-i} - r_{n-1} - r_{n-1} \]

\[
= \sum_{i=0}^{N} a_i \delta_{n-i} + \frac{1}{\Delta 2^{b-1}} \sum_{i=1}^{N} a_i \delta_{n-1-i} - r_{n-1}.
\] ....... (74)

Comparing \( n^{th} \) term from both sides, we have

\[
e_n + \sum_{i=1}^{N} b_i e_{n-i} - q_n - \sum_{i=1}^{N} b_i q_{n-i} = r_n + \sum_{i=0}^{N} a_i \delta_{n-i}
\] ....... (75)

Now using the z-transform of eqn. (75), and solving we find
\[
Z \left[ \sum_{i=1}^{N} a_{i} \partial_{z_{i}} \right] = N(z) \phi_{cC}(Z)
\]

(76)

Where \( N(Z) = \sum_{i=0}^{N} a_{i} z^{i} \).

So, from equation (75), we get

\[
D(Z) \phi_{cC}(Z) = \phi_{rr}(Z) + D(Z) q_{mm}(Z) + N(Z) \phi_{ss}(Z)
\]

(77)

Where \( D(Z) = 1 + \sum_{i=1}^{N} b_{i} z^{-i} \)

\[
M(Z) = \sum_{i=0}^{N} a_{i} z^{-i}
\]

\( \phi_{rr}^{(2)}, q_{mm}^{(2)} \text{ and } \phi_{ss}^{(2)} \) are power spectral densities of \( \{r_{n}\} \), \( \{i_{n}\} \) and \( \{\delta_{n}\} \) respectively.

The correlation function in the Z-transform \( \phi_{cC}(Z) \) is related to \( \phi_{yy}(Z) \) by the relation.

\[
\phi_{yy}(Z) = H(Z) H(Z^{-1}) \phi_{cC}(Z)
\]

(78)

Using the relation (78) in eqn. (77), we get

\[
\phi_{cC}(Z) = \frac{\phi_{rr}(Z)}{D(Z) D(Z^{-1})} + q_{mm}(Z) + \frac{N(Z)}{D(Z) D(Z^{-1})}
\]

(79)

Here, \( \phi_{rr}(Z) = \frac{2^{-2b+2}}{3} \)

and \( \phi_{yy}(Z) = 2^{-2b} \)

The mean square value of \( e_{n} \) is given by

\[
E \{ e_{n}^{2} \} = \frac{1}{2\pi j} \int_{|z|=1} \phi_{cC}(Z) \frac{dZ}{Z}
\]

\[
= \frac{1}{2\pi j} \int_{|z|=1} 2^{-2b+2} \frac{1}{3} \frac{1}{1 + \sum_{i=1}^{N} b_{i} z^{-i}} dZ
\]
\[
+ \frac{1}{2\pi j} \int_{|Z|=1} 2^{-b} \frac{dZ}{Z} + \frac{2^{-2b+2}}{1 + \sum_{i=1}^{N} b_i} \] \quad \ldots \ldots (80)

Mean square error \( E \{ e_n^2 \} \) is evaluated and found to be \( 50.38 \times 10^{-6} \) for \( b = 9 \) with \( b_1 = -0.6171875 \) and \( b_2 = 0.234375 \). It is more than four times of conventional binary \( (11.68 \times 10^{-6}) \)\cite{54}. The reason is that the step size is twice in our system compared with conventional binary system.

**Limit Cycles**

\( r_n \) can be expressed as.

\[
r_n = \{-\frac{1}{2} + e_n\} \cdot 2^{-b} \quad \ldots \ldots (81)
\]

Where \( e_n \) is the white noise.

Now, putting equation (81) into equation (68) for \( n = k \), we get

\[
w'_k = \varepsilon_k - b_1 w'_{n-1} - b_2 w'_{n-2} - \varepsilon_{k-1} \quad \ldots \ldots (82)
\]

where \( v = 0 \), \( q_n = 0 \) due to zero input condition.

\( \varepsilon_k \) is the error introduced by the quantizer in DPCM-DPCM converter at time \( k \).

Now, \( 2^{-b} > \varepsilon_k \geq 0 \).

Since \( w_k \) and \( \varepsilon_k \) have only a finite number of states and \( \varepsilon_k \) must be periodic.

\( \varepsilon'_n = \varepsilon_{n+p} \) for all \( n \), where \( p \) is the period.

Now, we assume that a limit cycle of period \( p \) exists i.e.

\[
w'_n = w'_n + p \quad \text{for all \( n \)}
\]

It can be re-written as
(1 + b_1 + b_2) \sum_{k=n+1}^{n+p} w_k = 0 \quad \ldots \ldots(83)

where \( \varepsilon_n = \varepsilon_0 + p \)

\[ w_n = w_{n+p}, \quad w_{n-1} = w_{n-1+p}, \]

for a stable filter

\[ (1 + b_1 + b_2) \neq 0 \]

Hence, \( \sum_{k=n+1}^{n+p} w_k = 0. \) \quad \ldots \ldots(84)

From eqn. (84), it is clear that for \( p = 1 \), the condition is not satisfied.

For period 2, the limit cycle is given by

\[ l_1 - \varepsilon_1 = -b_1 l_2 - b_2 l_1 - \varepsilon_2 \] \quad \ldots \ldots(85)

\[ l_2 - \varepsilon_2 = -b_1 l_1 - b_2 l_2 - \varepsilon_1 \] \quad \ldots \ldots(86)

From eqn. (85), we get

\[ l_1 = -l_2 = I \]

From eqn. (85) and (86), we have

\[ I = \frac{\varepsilon_2 - \varepsilon_1}{1 - b_1 + b_2} \] \quad \ldots \ldots(87)

Since, \( I = i \cdot 2^h \),

where \( i \) is an odd integer and \( b \) is the word length of signal \( \omega_k \).

Since, \( \varepsilon_2 - \varepsilon_2 < 2^h \),

So, \( \frac{1}{1 - b_1 + b_2} > i \)

or \( 1 - b_1 + b_2 < i \) \quad \ldots \ldots(88)

This is the required condition for stability.
2.8 HARDWARE REALIZATION

Hardware realization of second order DPCM filter is shown in fig. (12)

\( \phi \) is defined as.

\[ \phi (x^1 x^2 x^3 x^4 x^5) = a_0 x^1 + a_1 x^2 + a_3 x^3 - b_1 x^4 - b_2 x^5 \] ......... (89)

For second order filter

\[ w_n = a_0 v_n + a_1 v_{n-1} + a_2 v_{n-2} - b_1 w_{n-1} - b_2 w_{n-2} \] ......... (90)

in terms of \( \phi \), \( w_n \) may be expressed as,

\[ w_n = \sum_{j=1}^{b} \phi (v_n^j v_{n-1}^j v_{n-2}^j w_{n-1}^j w_{n-2}^j) \] ......... (91)

The function \( \phi \) has 32 values \( \phi \) can be realised by programmed read only memory (PROM) addressed by binary vectors. The table (1) shows a program for a ROM for \( b=7 \) bits, organised by 32 x 7 words for \( a_1 = 0.95 \), \( a_2 = -0.1665478 \), \( a_3 = 0.095 \), \( b_1 = -1.8080353 \) and \( b_2 = 0.9129197 \).

The five columns of the memory address correspond to five binary arguments of the \( \phi \), \( \phi \) is scaled down by 4. The sequence \( v_n \) and \( w_n \) enter serially into shift registers \( RV_1, RV_2, RW_1 \) and \( RW_2 \). After a shift a new vector appears at the input of circuit realising \( \phi \). The output register is connected to switch. First of all, it is connected to switch at position one and at \( (b+1) \) st addition cycle, the switch is connected to position two which connects to register \( RW_1 \). For first \( b \) addition cycle \( RV_1, RV_2, RW_1 \) and \( RW_2 \) are shifted \( RW_1 \) is the DPCM-DPCM converter register. Each register contains \( b \) bits which goes to switch position two.
2.9 CONCLUSIONS

In digital signal processing, the input analog signal is normalised to a fractional number with negative sign. In conventional binary system two's complement representation is utilized. Although it is an efficient representation but it suffers with certain drawbacks. It requires extra sign bits for representation. Extra hardwares are needed for bits conversion. The multiplication process in two's complement is a complex. In general Booth's algorithm is employed in which as each step of multiplication sign bit is defected. The addition and subtraction depend on the nature of sign bit. Finally, arithmetic sum is required. It is very difficult to obtain modular circuits. The quantizer characteristic is asymmetric and requires large number of quantization steps. Hence, a large number of bits are required during quantization process.

Truncation and rounding are different in this representation and produces different results. In general rounding is preferred because it gives out accurate result.

In (+1, -1) system the size is inherent and no extra bit is required. There is no need of extra hardware for conversion. The multiplication process is simple and straight. It requires only add shift technique. The multiplication process is faster and simpler than two's complement representation. The quantizer characteristic is symmetrical. The quantizer step is twice in comparison of two's complement. Hence less number of quantisation levels are required for signal in comparison of two's
complement. So, less number of bits are required to process the signal. Modular circuits can be achieved in single way. A single multiplier cell is possible.

Truncation and rounding produce the some results, therefore to reduce the coefficient bit after multiplication, the truncation process is used which is single and fast. The positive and negative signals are represented in unified way.

The (+1, -1) system is well suited for DPCM filter. Quantization error is the integral part of filter derivation. Round off noise is very large in comparison of conventional binary. Quantization is symmetric and it lies between limits (1-2^b) and (1-2^b). Two’s complement quantizer is assymmetric and lies between -1 and 1-2^{b+1}. In this system, no extra bit for sign is required. The addition process is very simple because the positive and negative numbers are represented in a unified way. In 2’s complement, the special care for sign and carry bits are taken during addition process. In (0, 1) system intermediate overflows results, so overflow detection circuit is required. In our system, there is no intermediate overflows. If an overflow takes place at the last result, it can be detected by simply XNOR gate at the last two bits when they are of the same sign. If the last two bits are of opposite nature, they are contracted by replacing last sum bit by carry bit.

The register length is minimised. The hardware realisation of DPCM filter is minimised. It requires only four registers for RV_1, RV_2, RW_1 and RW_2, one adder, and one ROM. In 2's complement representation, the same DPCM filter requires extra registers RW_{1B}, RW_{1C}, one bit full adder and one overflow detector circuit. In our
system, the realisation of DPCM filter is straight and simple. Thus it is more economical in terms of power consumption and hardware realisation.

The (+1, -1) system is applicable in the design of filters, in DFT and FFT butterfly computation and coding like PCM, DPCM, DM and LPC.
Table 1 (SNR)$_0$ and $G_p$ for different values of $P$.

<table>
<thead>
<tr>
<th>$P$</th>
<th>(SNR)$_0$ (db)</th>
<th>$G_p$ (db)</th>
<th>(SNR)$_0$ (db)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.87</td>
<td>5.23</td>
<td>5.4</td>
</tr>
<tr>
<td>3</td>
<td>4.98</td>
<td>10.22</td>
<td>8.4</td>
</tr>
<tr>
<td>5</td>
<td>8.43</td>
<td>13.66</td>
<td>8.6</td>
</tr>
<tr>
<td>7</td>
<td>10.83</td>
<td>16.77</td>
<td>10.5</td>
</tr>
<tr>
<td>Memory address</td>
<td>Contents (ϕ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0. 1 1 1 1 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \ddot{1} \dot{1} \ddot{1} 1 )</td>
<td>( 0.1 \dddot{1} \ddot{1} \ddot{1} \ddot{1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \dddot{1} \ddot{1} \ddot{1} \dot{1} )</td>
<td>( 0. \dddot{1} \dddot{1} \ddot{1} \dddot{1} \ddot{1} \dot{1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \dddot{1} \ddot{1} \ddot{1} )</td>
<td>( 0. \dddot{1} \ddot{1} \ddot{1} \dddot{1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \dddot{1} \ddot{1} \ddot{1} )</td>
<td>( 0. \dddot{1} \dddot{1} \ddot{1} \ddot{1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \dddot{1} \ddot{1} \dot{1} )</td>
<td>( 0. \dddot{1} \dddot{1} \ddot{1} \ddot{1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \dddot{1} \ddot{1} \ddot{1} )</td>
<td>( 0. \dddot{1} \dddot{1} \ddot{1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \dddot{1} \ddot{1} \ddot{1} )</td>
<td>( 0. \dddot{1} \dddot{1} \dddot{1} \ddot{1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \dddot{1} \ddot{1} \ddot{1} \ddot{1} )</td>
<td>( 0. \dddot{1} \dddot{1} \ddot{1} \ddot{1} )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Where \( \ddot{1} = -1 \)
Fig. 1.
Fig. (2)
Three-bit multiplier

Fig. 3.
Prediction order against their values

Fig. 4.
Variation of MSE with orders of filter

Fig. 5.
Block diagram of DPCM transmitter

Block diagram of DPCM receiver

Fig. 6.
Fig. (7)
Fig (3)
Fig. (9)

Fig. 11
Fig. (10):
Captions to figures

Fig. 1. (a) Representation of digital sequence.
(b) Representation of digital step sequence.

Fig. 2. Characteristic of quantizer.

Fig. 3. Three bit multiplication process.

Fig. 4. Variation of the predicted values with prediction coefficients.

Fig. 5. Mean Square Error and the orders of filter coefficients.

Fig. 6. (a) Block diagram of DPCM transmitter.
(b) Block diagram of DPCM receiver.
CAPTIONS TO FIGURES

Fig. 7: DPCM digital processor system.

Fig. (8): Quantizer characteristics.

Fig. (9): Second order recursive DPCM digital filter.

Fig. (10): Response of Seven bit DPCM high pass filter.

Fig. 11: Hardware realization of Second order DPCM filter.