

## Unit 8: Interfacing and Applications of DSP Processor

**8.1 Introduction:** In the case of parallel peripheral interface, the data word will be transferred with all the bits together. In addition to parallel peripheral interface, there is a need for interfacing serial peripherals. DSP has provision of interfacing serial devices too.

**8.2 Synchronous Serial Interface:** There are certain I/O devices which handle transfer of one bit at a time. Such devices are referred to as serial I/O devices or peripherals. Communication with serial peripherals can be synchronous, with processor clock as reference or it can be asynchronous. Synchronous serial interface (SSI) makes communication a fast serial communication and asynchronous mode of communication is slow serial communication. However, in comparison with parallel peripheral interface, the SSI is slow. The time taken depends on the number of bits in the data word.

**8.3 CODEC Interface Circuit:** CODEC, a coder-decoder is an example for synchronous serial I/O. It has analog input-output, ADC and DAC. The signals in SSI generated by the DSP are DX: Data Transmit to CODEC, DR: Data Receive from CODEC, CLKX: Transmit data with this clock reference, CLKR: Receive data with this clock reference, FSX: Frame sync signal for transmit, FSR: Frame sync signal for receive, First bit, during transmission or reception, is in sync with these signals, RRDY: indicator for receiving all bits of data and XRDY: indicator for transmitting all bits of data.

Similarly, on the CODEC side, signals are FS\*: Frame sync signal, DIN: Data Receive from DSP, DOUT: Data Transmit to DSP and SCLK: Tx / Rx data with this clock reference. The block diagram depicting the interface between TMS320C54xx and CODEC is shown in fig. 8.1. As only one signal each is available on CODEC for clock and frame synchronization, the related DSP side signals are connected together to clock and frame sync signals on CODEC. Fig. 8.2 and fig. 8.3 show the timings for receive and transmit in SSI, respectively.

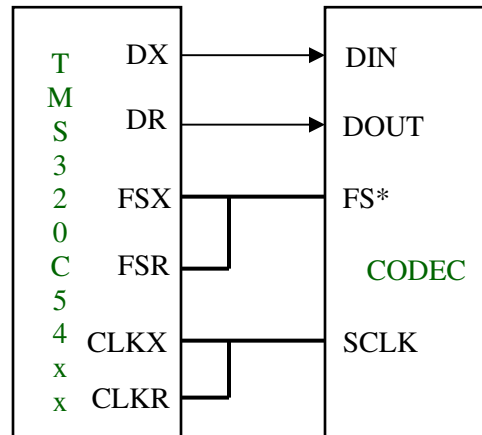


Fig. 8.1: SSI between DSP & CODEC

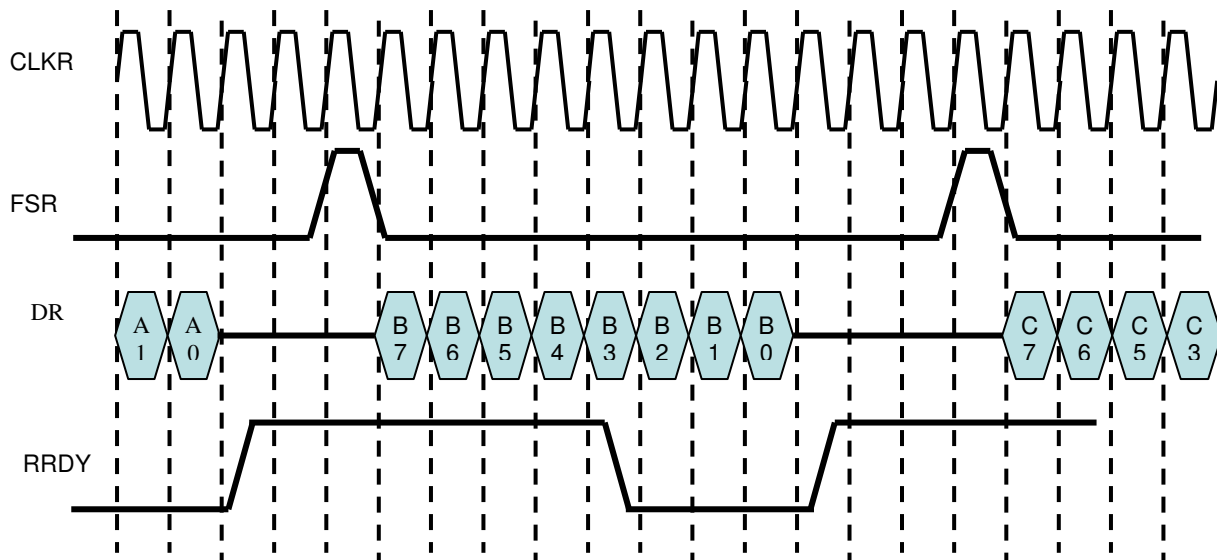


Fig. 8.2: Receive Timing for SSI

As shown, the receiving or transmit activity is initiated at the rising edge of clock, CLKR / CLKX. Reception / Transfer starts after FSR / FSX remains high for one clock cycle. RRDY / XRDY is initially high, goes LOW to HIGH after the completion of data transfer. Each transfer of bit requires one clock cycle. Thus, time required to transfer / receive data word depends on the number of bits in the data word. An example of data word of 8 bits is shown in the fig. 8.2 and fig. 8.3.

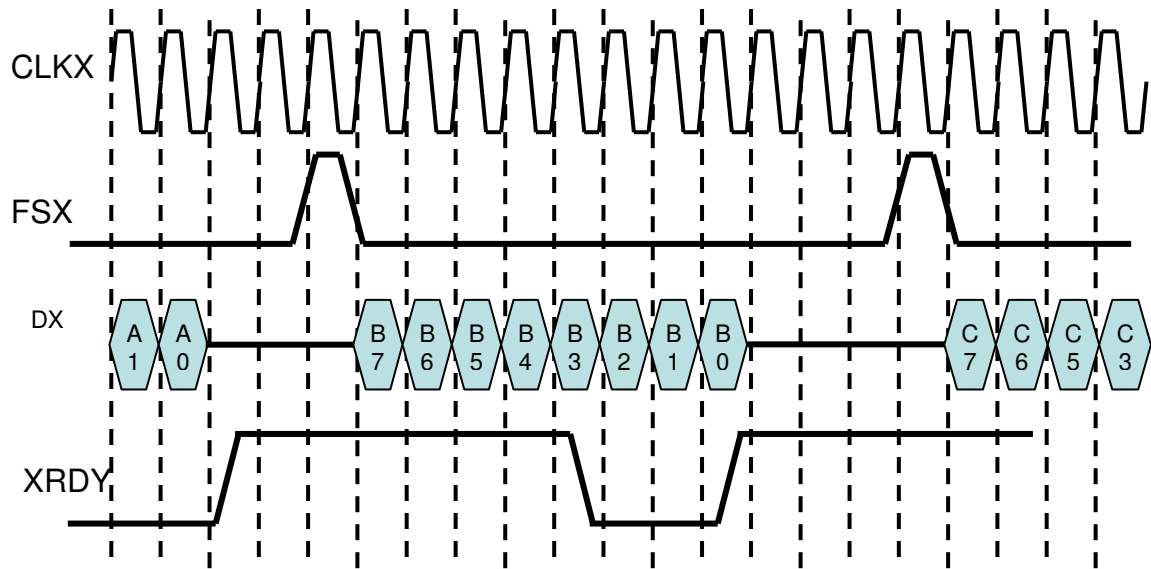


Fig 8.3: Transmit Timing for SSI

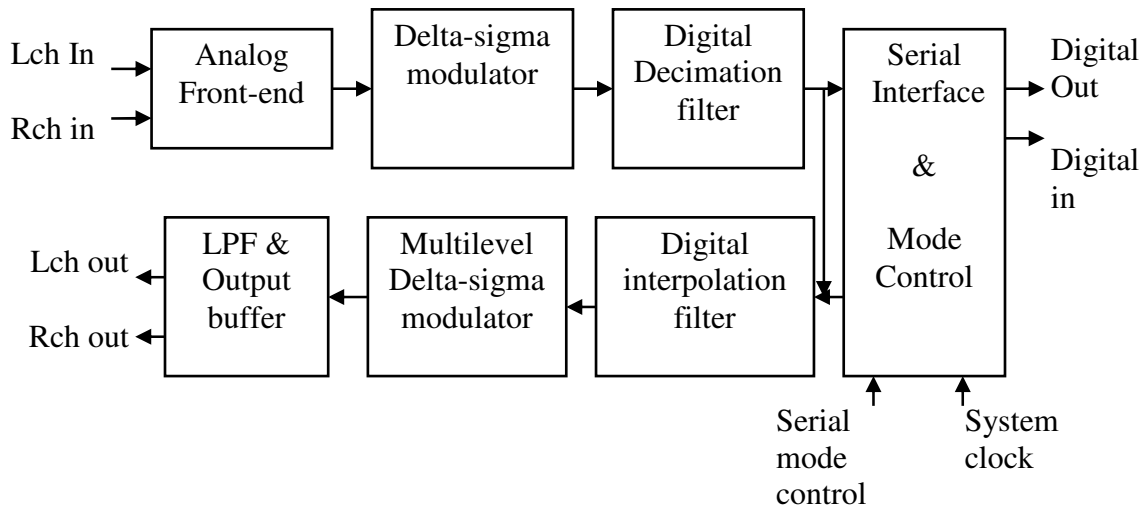


Fig. 8.4: Block diagram for CODEC PCM3002

Fig. 8.4 shows the block diagram of PCM3002 CODEC. Analog front end samples signal at 64X over sampling rate. It eliminates need for sample-and-hold circuit and simplifies need for anti aliasing filter. ADC is based on Delta-sigma modulator to convert analog signal to digital form. Decimation filter reduces the sampling rate and thus processing does not need high speed devices. DAC is Delta-sigma modulator, converts digital signal

to analog signal. Interpolation increases the sampling rate back to original value. LPF smoothens the analog reconstructed signal by removing high frequency components.

The Serial Interface monitors serial data transfer. It accepts built-in ADC output and converts to serial data and transmits the same on DOUT. It also accepts serial data on DIN & gives the same to DAC. The serial interface works in synchronization with BCLKIN & LRCIN. The Mode Control initializes the serial data transfer. It sets all the desired modes, the number of bits and the mode Control Signals, MD, MC and ML. MD carries Mode Word. MC is the mode Clock Signal. MD to be loaded is sent with reference to this clock. ML is the mode Load Signal. It defines start and end of latching bits into CODEC device.

Figure 8.5 shows interfacing of PCM3002 to DSP in DSK. DSP is connected to PCM3002 through McBSP2. The same port can be connected to HPI. Mux selects one among these two based on CPLD signal. CPLD in Interface also provides system clock for DSP and for CODEC, Mode control signals for CODEC. CPLD generates BCLKIN and LRCIN signals required for serial interface.

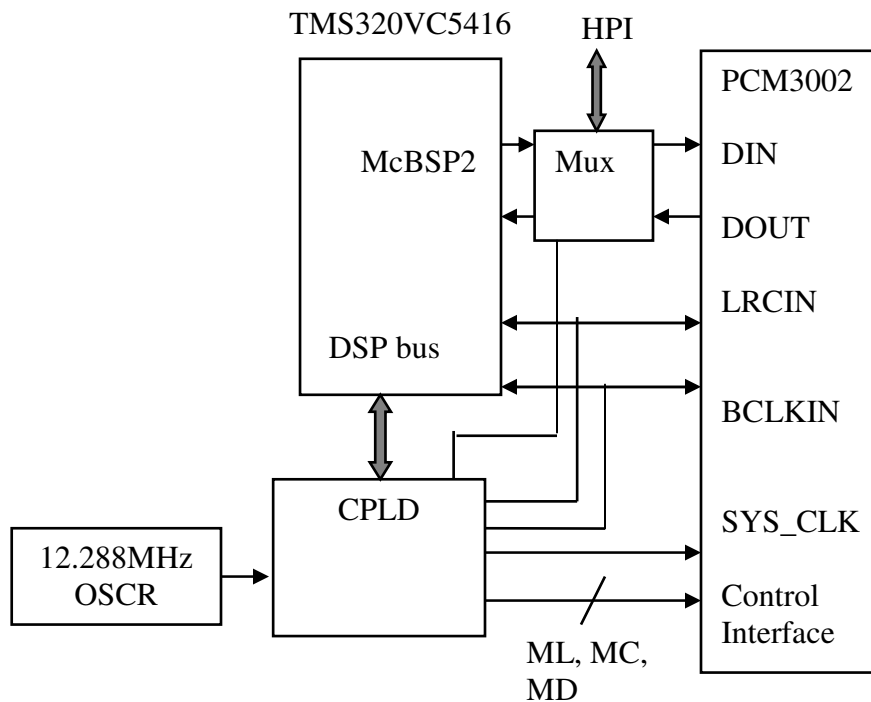


Fig. 8.5: PCM3003 Interface to DSP in DSK

PCM3002 CODEC handles data size of 16 / 20 bits. It has 64x over-sampling, delta-sigma ADC & DAC. It has two channels, called left and right. The CODEC is programmable for digital de-emphasis, digital attenuation, soft mute, digital loop back, power-down mode. System clock, SYSCLK of CODEC can be 256fs, 384fs or 512fs. Internal clock is always 256fs for converters, digital filters. DIN, DOUT are the single line data lines to carry the data into the CODEC and from CODEC. Another signal BCLKIN is data bit clock, the default value of which is CODEC SYSCLK / 4. LRCIN is frame sync signal for Left and Right Channels. The frequency of this signal is same as the sampling frequency. The default divide factor can be 2, 4, 6 and 8. Thus, sampling rate is minimum of 6 KHz and maximum of 48 KHz.

**Problem P8.1:** A PCM3002 is programmed for the 12 KHz sampling rate. Determine the divisor N that should be written to the CPLD of the DSK and the various clock frequencies for the set up.

Solution: CPLD input Clock=12.288MHz (known)  
 Sampling rate  $f_s = \text{CODEC\_SYSCLK} / 256 = 12\text{KHz}$  (given)  
 CPLD output clock,  $\text{CODEC\_SYSCLK} = 12.288 \times 106 / N$   
 Thus,  $\text{CODEC\_SYSCLK} = 256 \times 12 \text{ KHz}$   
 $\& N = 12.288 \times 106 / (256 \times 12 \times 103)$   
 $= 4$

**Problem P8.2:** Determine the timing parameters for a 20 bit data communication at 8KHz.

Solution: CPLD input Clock=12.288MHz (known)  
 Sampling rate  $f_s = 8\text{KHz}$  (given)  
 Proceeding as in previous problem,  $N=6$   
 Thus,  $\text{CODEC\_SYSCLK} = 2.048\text{MHz}$   
 $\text{BCLKIN}$ , Bit clock rate  $= \text{CODEC\_SYSCLK} / 4 = 512\text{KHz}$   
 $\text{LRCIN} = f_s = 8\text{KHz}$

**Problem P8.3:** Frame Sync is generated by dividing the 8.192MHz clock by 256 for the serial communication. Determine the sampling rate and the time a 16 bit sample takes when transmitted on the data line.

Solution:  $\text{LRCIN}$ , Frame Sync  $= 8.192 \times 106 / 256 = 32 \text{ KHz}$   
 Sampling rate  $f_s =$  frequency of  $\text{LRCIN} = 32 \text{ KHz}$   
 $\text{BCLKIN}$ , Bit clock rate  $= \text{CODEC\_SYSCLK} / 4 = 8.192 \times 106 / 4 = 2.048\text{MHz}$

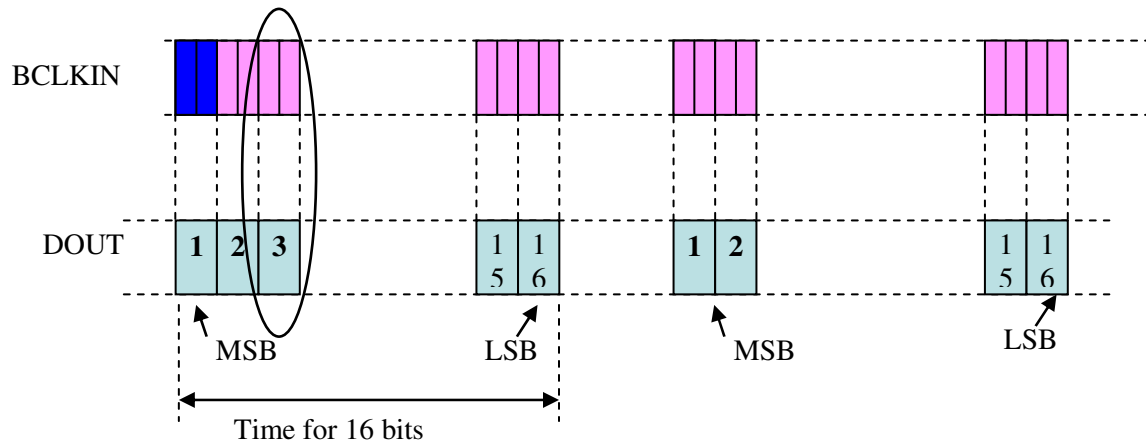


Fig. P8.3: Time for Data Transmission

LRCIN, Frame Sync =  $8.192 \times 10^6 / 256 = 32$  KHz

Sampling rate  $f_s$  = frequency of LRCIN = 32 KHz

BCLKIN, Bit clock rate = CODEC\_SYSCLK / 4 =  $8.192 \times 10^6 / 4 = 2.048$  MHz

Bit clock period =  $1 / 2.048 \times 10^6 = 0.488 \times 10^{-6}$  s

Time for transmitting 16 bits =  $0.488 \times 10^{-6} \times 16 = 7.8125 \times 10^{-6}$  s (refer fig. P8.3)

The CODEC PCM3002 supports four data formats as listed in table 8.1. The four data formats depend on the number of bits in the data word, if the data is right justified or left justified with respect to LRCIN and if it is I<sup>2</sup>S (Integrated Inter-chip Sound) format.

Table 8.1: Data formats of CODEC		
Format	DAC	ADC
Format 0	16 bit, MSB first, right justified	16 bit, MSB first, left justified
Format 1	20 bit, MSB first, right justified	20 bit, MSB first, left justified
Format 2	20 bit, MSB first, left justified	20 bit, MSB first, left justified
Format 3	20 bit, MSB first, I <sup>2</sup> S	20 bit, MSB first, I <sup>2</sup> S

Figure 8.6 and fig. 8.7 depicts the data transaction for CODEC PCM3002. As shown in fig. 8.6, DIN (/ DOUT) carries the data. BCLKIN is the reference for transfer. When LRCIN is high, left channel inputs (/ outputs) the data and when LRCIN is low, right

channel inputs (/ outputs) the data. The data bits at the end (/ beginning) of the LRCIN thus Right (/ left) justified.

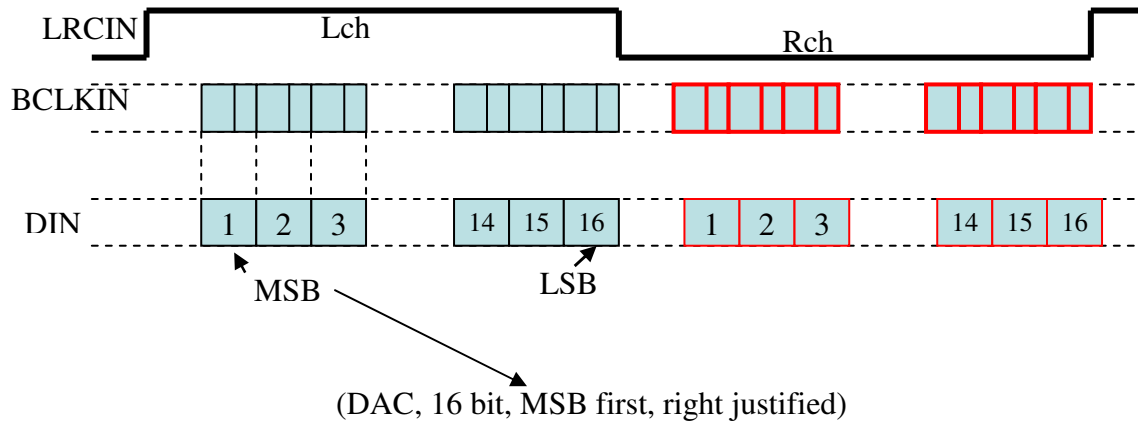


Fig. 8.6: Data Formats for PCM3002

Another data format handled by PCM3002 is I<sup>2</sup>S (Integrated Inter-chip Sound). It is used for transferring PCM between CD transport & DAC in CD player. LRCIN is low for left channel and high for right channel in this mode of transfer. During the first BCLKIN, there is no transmission by ADC. During 2<sup>nd</sup> BCLKIN onwards, there is transmission with MSB first and LSB last. Left channel data is handled first followed by right channel data.

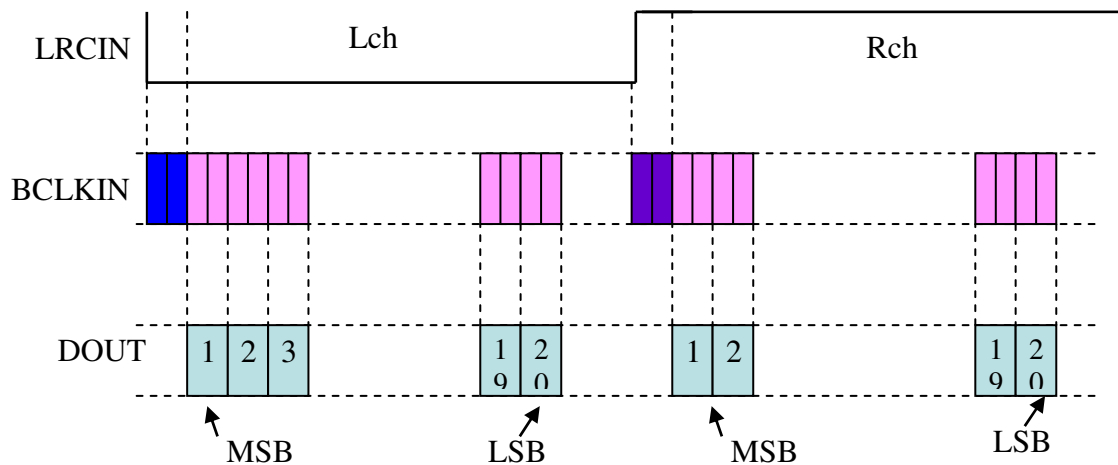


Fig. 8.7: ADC 20 bit, MSB first, I<sup>2</sup>S format

**8.4 DSP Based Bio-telemetry Receiver:** Biotelemetry involves transfer of physiological information from one remote place to another for the purpose of obtaining experts opinion. The receiver uses radio Frequency links. The schematic diagram of bio-telemetry receiver is shown in fig. 8.8. The biological signals may be single dimensional signals such as ECG and EEG or two dimensional signals such as an image, i.e., X-ray. Signal can even be multi dimensional signal i.e., 3D picture. The signals at source are encoded, modulated and transmitted. The signals at destination are decoded, demodulated and analyzed.

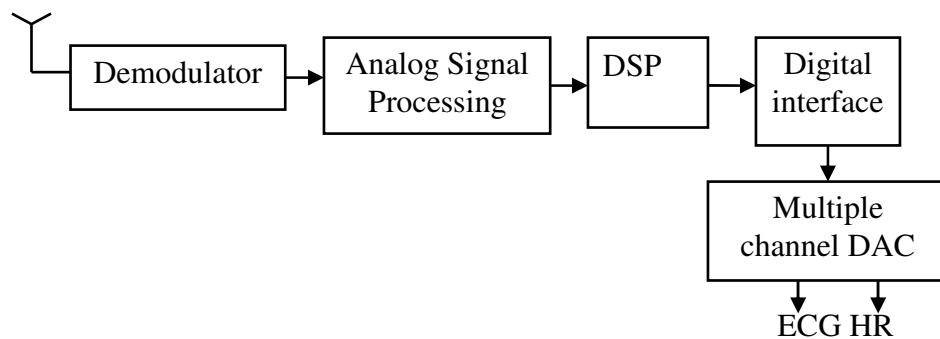


Fig. 8.8: Bio-telemetry Receiver

An example of processing ECG signal is considered. The scheme involves modulation of ECG signal by employing Pulse Position Modulation (PPM). At the receiving end, it is demodulated. This is followed by determination of Heart beat Rate (HR). PPM Signal either encodes single or multiple signals. The principle of modulation being that the position of pulse decides the sample value.

The PPM signal with two ECG signals encoded is shown in fig. 8.9. The transmission requires a sync signal which has 2 pulses of equal interval to mark beginning of a cycle. The sync pulses are followed by certain time gap based on the amplitude of the sample of 1<sup>st</sup> signal to be transmitted. At the end of this time interval there is another pulse. This is again followed by time gap based on the amplitude of the sample of the 2<sup>nd</sup> signal to be transmitted. After encoding all the samples, there is a compensation time gap followed by sync pulses to mark the beginning of next set of samples. Third signal may be encoded in either of the intervals of 1<sup>st</sup> or 2<sup>nd</sup> signal. With two signals encoded and the pulse width as  $t_p$ , the total time duration is  $5t_p$ .





A DSP based PPM signal decoding is shown in fig. 8.11. PPM signal interface generates the interrupt for DSP. DSP entertains the interrupt and starts a timer. When it receives another interrupt, it stops the timer and the count is treated as the digital equivalent of the sample value. The process repeats. Dual DAC converts two signals encoded into analog signals. And heart rate is determined referring to the ECG obtained by decoding.

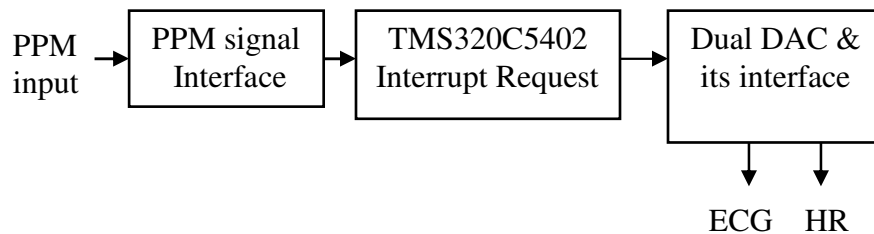


Fig. 8.11: DSP based biotelemetry Receiver Implementation

Heart Rate (HR) is a measure of time interval between QRS complexes in ECG signal. QRS complex in ECG is an important segment representing the heart beat. There is periodicity in its appearance indicating the heart rate. The algorithm is based on 1<sup>st</sup> and 2<sup>nd</sup> order absolute derivatives of the ECG signal. Since absolute value of derivative is taken, the filter will be a nonlinear filtering.

Let the 1<sup>st</sup> order derivative be  $y1(n) = |x(n) - x(n-1)|$

And let the 2<sup>nd</sup> order derivative be  $y2(n) = |x(n-2) - 2x(n-1) + x(n)|$

The 1<sup>st</sup> order derivative is obtained as the difference between the two adjacent samples, the present sample and the previous sample. In a similar way, 2<sup>nd</sup> order derivative is obtained by finding the derivative of 1<sup>st</sup> order derivative. The  $y1(n)$  and  $y2(n)$  are summed. &  $y3(n) = y1(n) + y2(n)$

High frequency components are removed from  $y3(n)$  by passing the same through a LPF to get  $y4(n) = \alpha(y3(n) - y4(n-1)) + y4(n-1)$

Mean of half of peak amplitudes is determined, which is threshold for detection of QRS complex. QRS interval is then the time interval between two such peaks. Time Interval between two peaks is determined using internal timer of DSP. Heart Rate, heart beat per

minute is computed using the relation  $HR = \text{Sampling rate} \times 60 / \text{QRS interval}$ . The signals at various stages are shown in fig. 8.12.

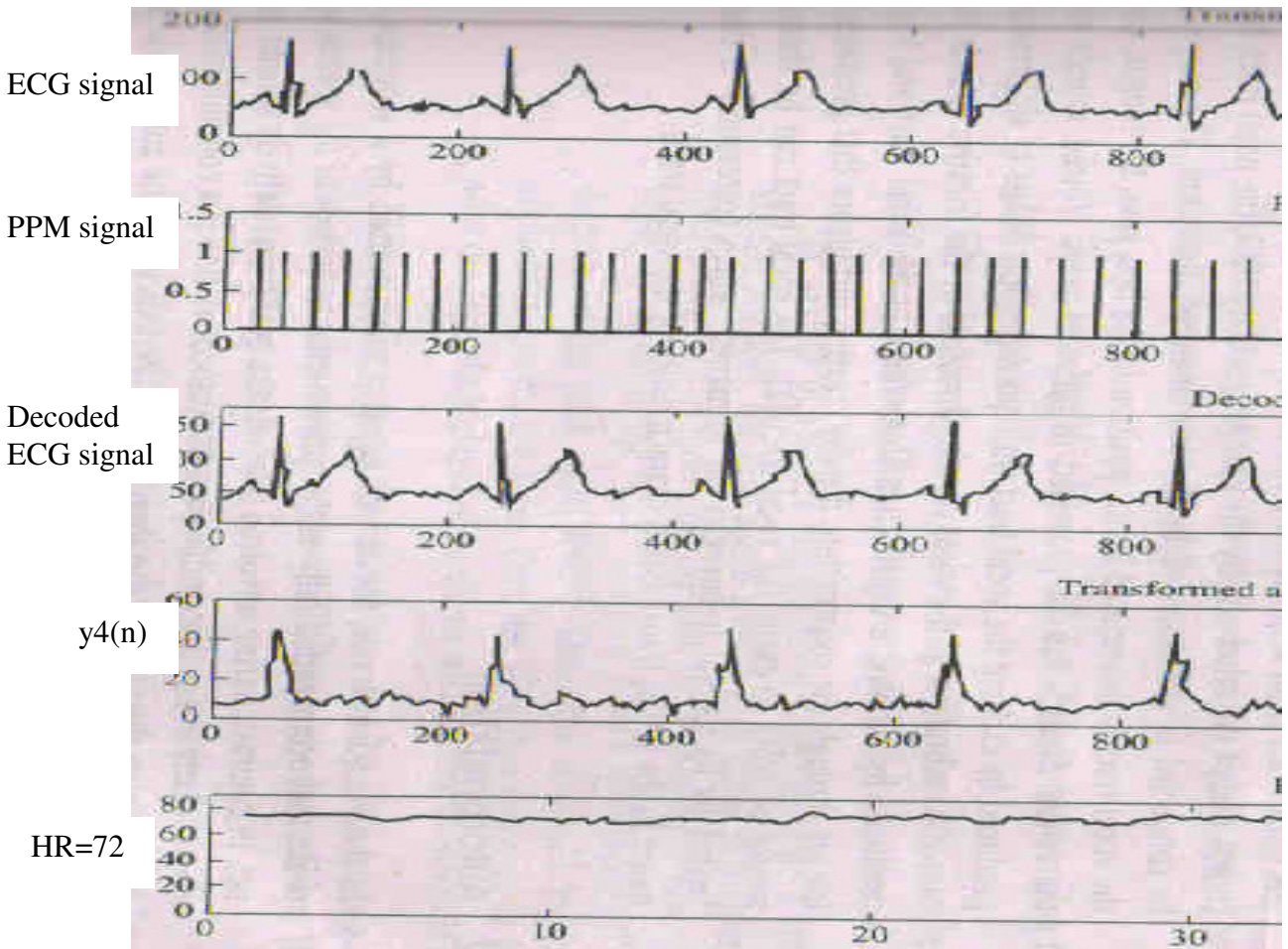


Fig. 8.12: Signals in determination of HR

**8.5 A Speech Processing System:** The purpose of speech processing is for analysis, transmission or reception as in the case of radio / TV / phone, denoising, compression and so on. There are various applications of speech processing which include identification and verification of speaker, speech synthesis, voice to text conversion and vice versa and so on. A speech processing system has a vocoder, a voice coding / decoding circuit. Schematic of speech production is shown in fig. 8.13. The vocal tract has vocal cord at one end and mouth at the other end. The shape of the vocal tract depends on position of lips, jaws, tongue and the velum. It decides the sound that is produced. There is another tract, nasal tract. Movement of velum connects or disconnects nasal tract. The overall voice that sounds depends on both, the vocal tract and nasal tract.

Two types of speech are voiced sound and unvoiced sound. Vocal tract is excited with quasi periodic pulses of air pressure caused by vibration of vocal cords resulting in voiced sound. Unvoiced sound is produced by forcing air through the constriction, formed somewhere in the vocal tract and creating turbulence that produces source of noise to excite the vocal tract.

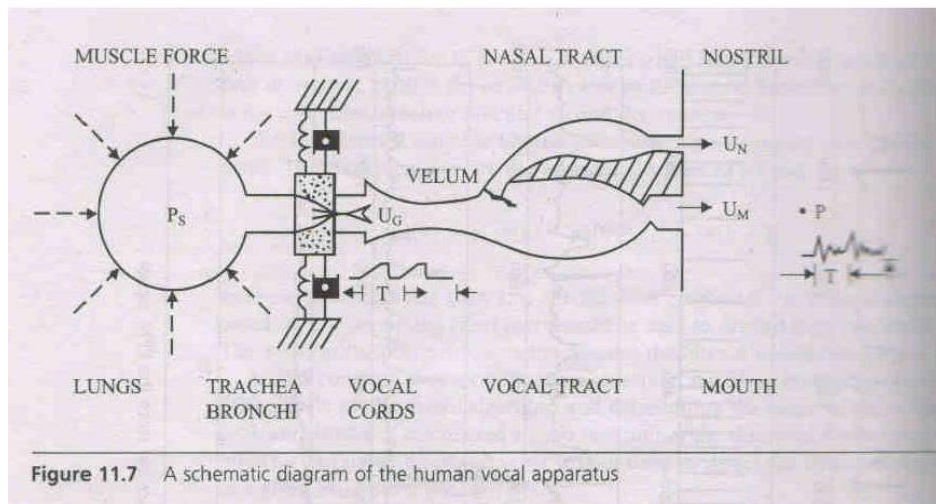


Fig. 8.13: Speech Production Mechanism

By the understanding of speech production mechanism, a speech production model representing the same is shown in fig. 8.14. Pulse train generator generates periodic pulse train. Thus it represents the voiced speech signal. Noise generator represents unvoiced speech. Vocal tract system is supplied either with periodic pulse train or noise. The final output is the synthesized speech signal.

Sequence of peaks occurs periodically in voiced speech and it is the fundamental frequency of speech. The fundamental frequency of speech differs from person to person and hence sound of speech differs from person to person. Speech is a nonstationary signal. However, it can be considered to be relatively stationary in the intervals of 20ms. Fundamental frequency of speech can be determined by autocorrelation method. In other words, it is a method of determination of pitch period. Periodicity in autocorrelation is because of the fundamental frequency of speech. A three level clipping scheme is discussed here to measure the fundamental frequency of speech. The block diagram for the same is shown in fig. 8.15.

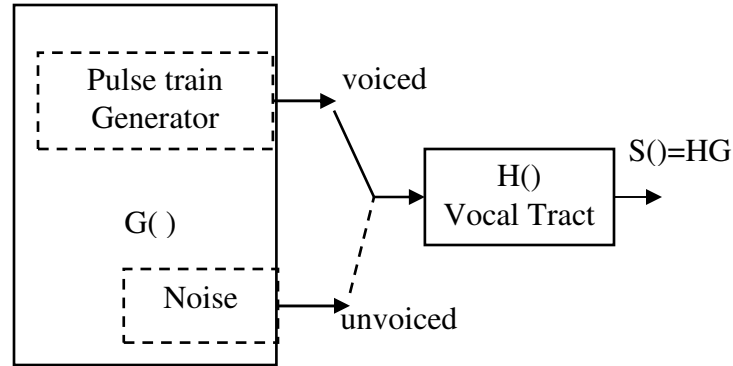


Fig. 8.14: Speech Production Model

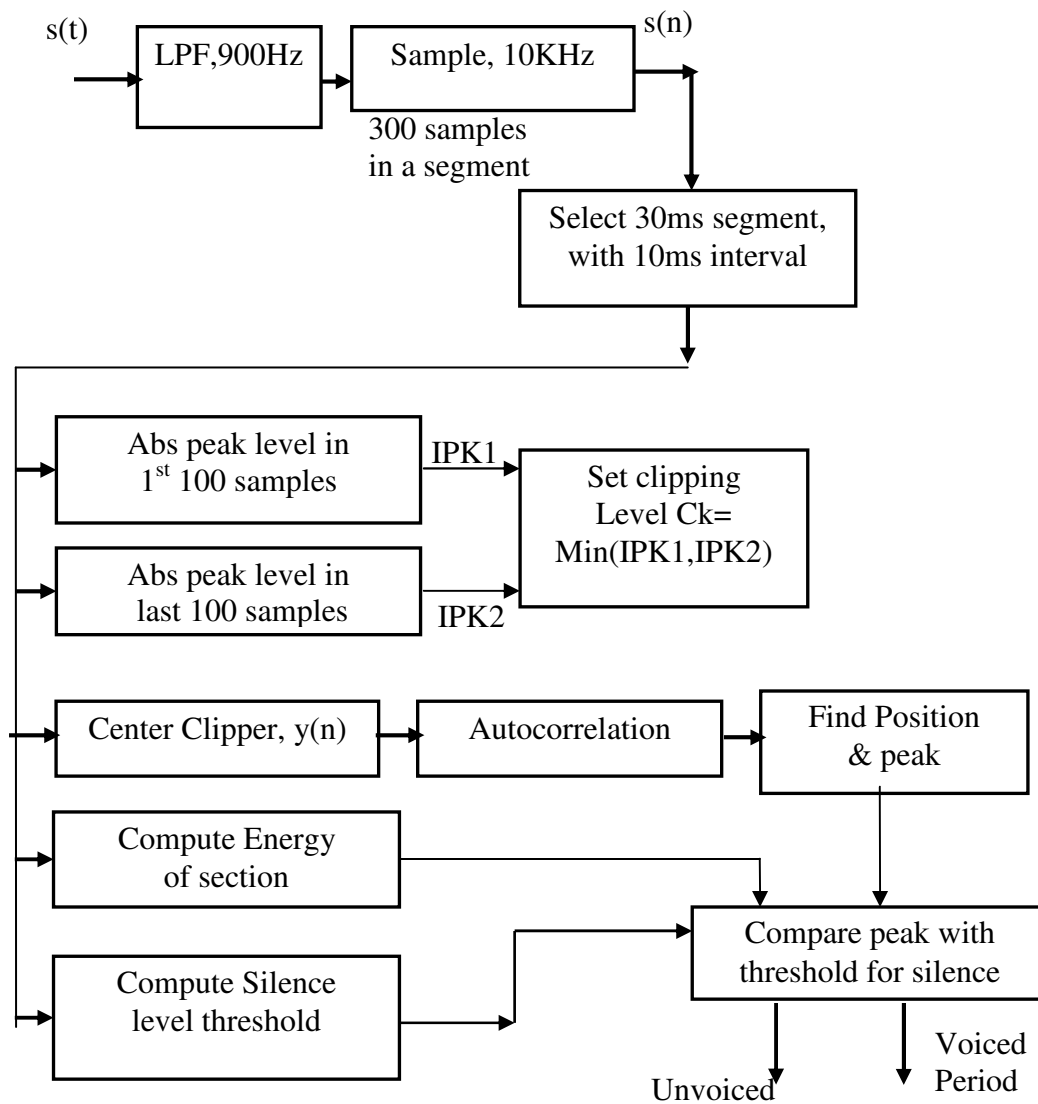


Fig. 8.15: Block Diagram of Clipping Autocorrelation Pitch Detector

The speech signal  $s(t)$  is filtered to retain frequencies up to 900Hz and sampled using ADC to get  $s(n)$ . The sampled signal is processed by dividing it into set of samples of 30ms duration with 20ms overlap of the windows. The same is shown in fig. 8.16.

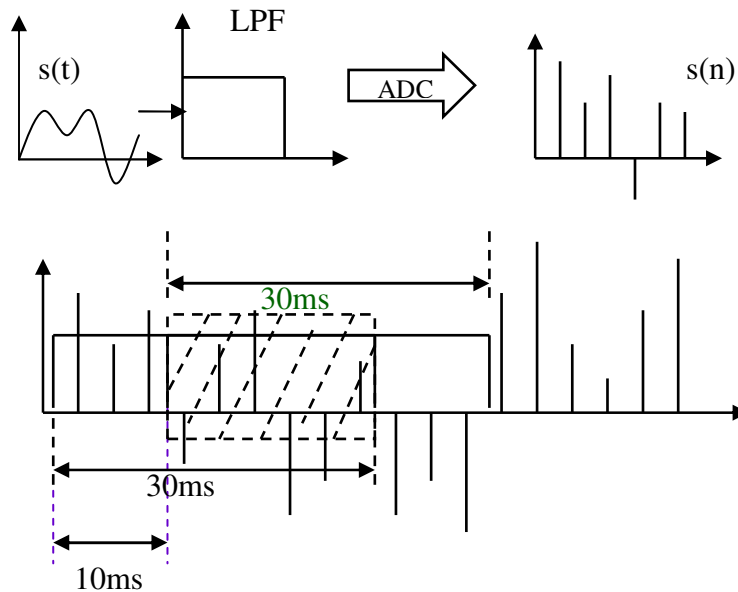
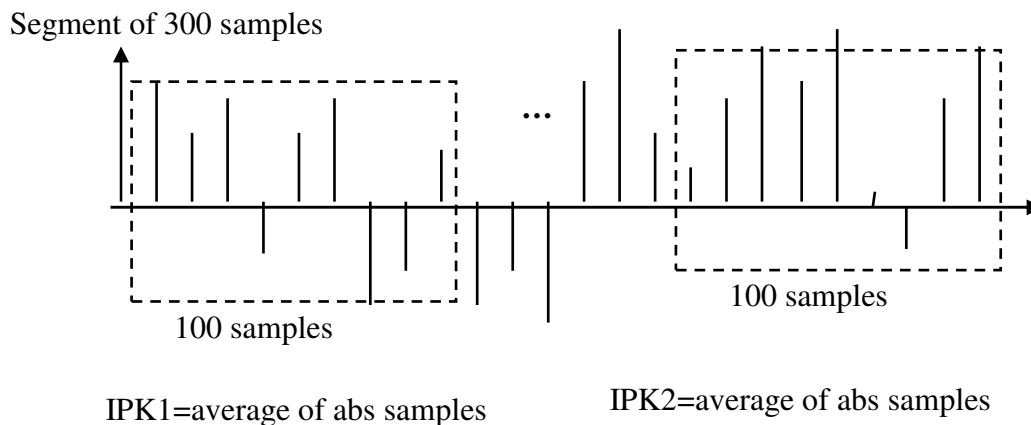


Fig. 8.16: LPF, ADC and windowing

A threshold is set for three level clipping by computing minimum of average of absolute values of 1<sup>st</sup> 100 samples and last 100 samples. The scheme is shown in fig. 8.17.



$$\text{Threshold for Clipping } C_k = \text{Min}(\text{IPK1}, \text{IPK2}) \times 70\%$$

Fig. 8.17: Setting threshold for Clipping

The transfer characteristics of three level clipping circuit is shown in fig. 8.18. If the sample value is greater than +CL, the output  $y(n)$  of the clipper is set to 1. If the sample value is more negative than -CL, the output  $y(n)$  of the clipper is set to -1. If the sample value is between  $-CL$  and  $+CL$ , the output  $y(n)$  of the clipper is set to 0.

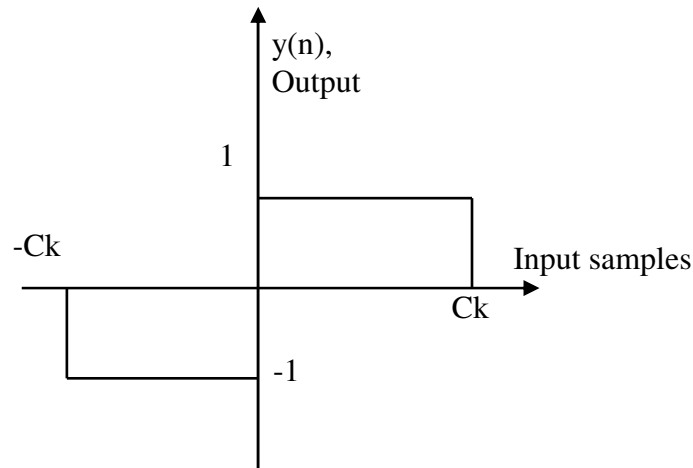


Fig. 8.18: Center Clipper

The autocorrelation of  $y(n)$  is computed which will be 0,1 or -1 as defined by eq (1). The largest peak in autocorrelation is found and the peak value is compared to a fixed threshold. If the peak value is below threshold, the segment of  $s(n)$  is classified as unvoiced segment. If the peak value is above threshold, the segment of  $s(n)$  is classified as voiced segment. The functioning of autocorrelation is shown in fig. 8.19.

$$R_n(k) = \sum_{m=0}^{N-1-k} y(n+m)y(n+m+k)$$

$$y(n+m)y(n+m+k) = \begin{cases} 0 & \text{if } y(n+m) = 0 \text{ or } y(n+m+k) = 0 \\ +1 & \text{if } y(n+m) = y(n+m+k) \\ -1 & \text{if } y(n+m) \neq y(n+m+k) \end{cases} \quad (1)$$

As shown in fig. 8.19, A is a sample sequence  $y(n)$ . B is a window of samples of length  $N$  and it is compared with the  $N$  samples of  $y(n)$ . There is maximum match. As the window is moved further, say to a position C the match reduces. When window is moved further say to a position D, again there is maximum match. Thus, sequence  $y(n)$  is periodic. The period of repetition can be measured by locating the peaks and finding the time gap between them.

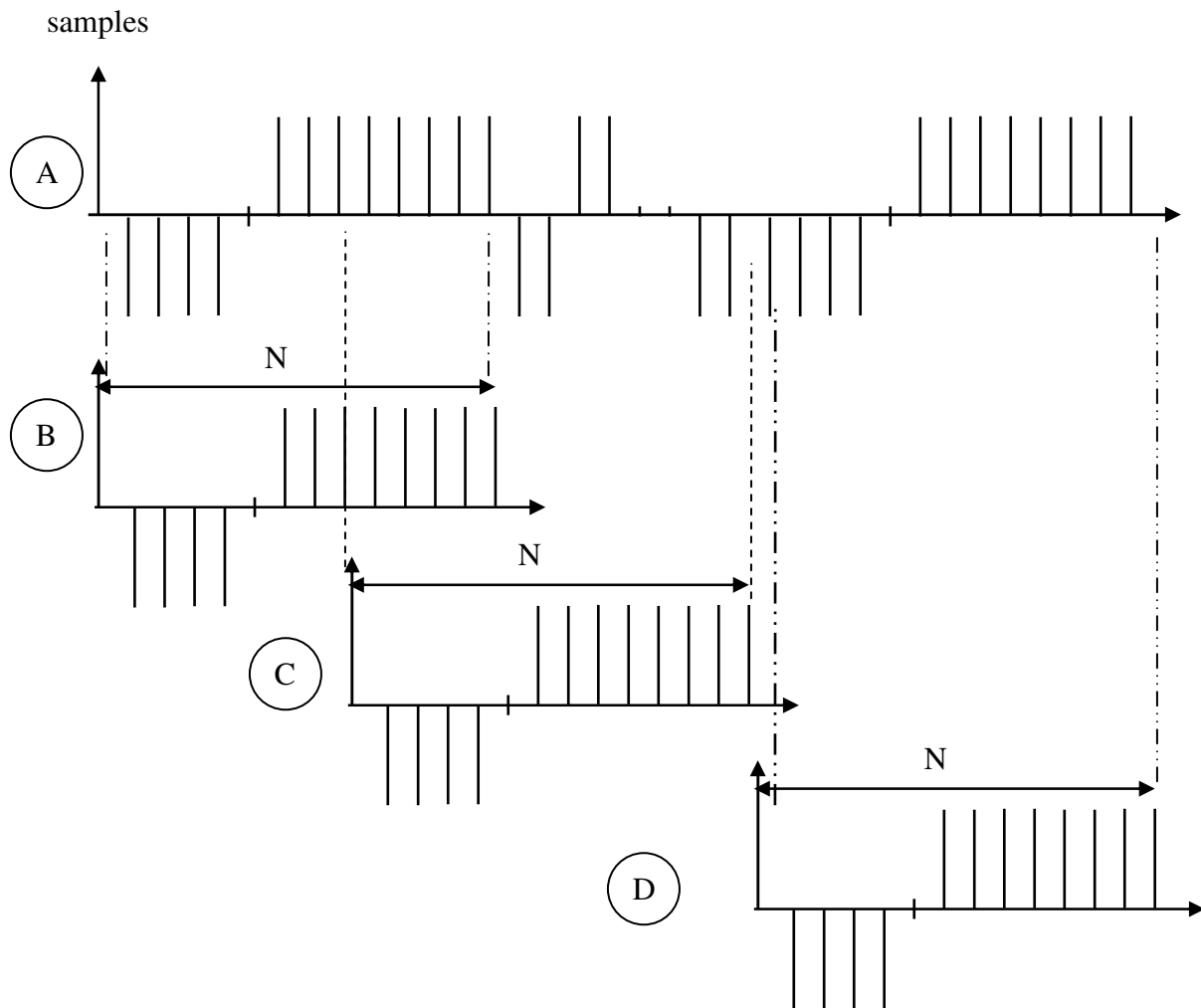


Fig. 8.19: Autocorrelation functioning

**8.5 An Image Processing System:** In comparison with the ECG or speech signal considered so far, image has entirely different requirements. It is a two dimensional



signal. It can be a color or gray image. A color image requires 3 matrices to be maintained for three primary colors-red, green and blue. A gray image requires only one matrix, maintaining the gray information of each pixel (picture cell). Image is a signal with large amount of data. Of the many processing, enhancement, restoration, etc., image compression is one important processing because of the large amount of data in image. To reduce the storage requirement and also to reduce the time and band width required to transmit the image, it has to be compressed. Data compression of the order of factor 50 is sometimes preferred. JPEG, a standard for image compression employs lossy compression technique. It is based on discrete cosine transform (DCT). Transform domain compression separates the image signal into low frequency components and high frequency components. Low frequency components are retained because they represent major variations. High frequency components are ignored because they represent minute variations and our eye is not sensitive to minute variations.

Image is divided into blocks of 8 x 8. DCT is applied to each block. Low frequency coefficients are of higher value and hence they are retained. The amount of high frequency components to be retained is decided by the desirable quality of reconstructed image. Forward DCT is given by eq (2).

$$f_{v,u} = \frac{1}{4} c_v c_u \sum_{x=0}^7 \sum_{y=0}^7 f_{x,y} \cos\left(\frac{(2x+1)u\pi}{16}\right) \cos\left(\frac{(2y+1)v\pi}{16}\right) \quad (2)$$

Since the coefficients values may vary with a large range, they are quantized. As already noted low frequency coefficients are significant and high frequency coefficients are insignificant, they are allotted varying number of bits. Significant coefficients are quantized precisely, with more bits and insignificant coefficients are quantized coarsely, with fewer bits. To achieve this, a quantization table as shown in fig. 8.20 is employed. The contents of Quantization Table indicate the step size for quantization. An entry as smaller value implies smaller step size, leading to more bits for the coefficients and vice versa.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Fig. 8.20: Matrix used for quantization & dequantization

The quantized coefficients are coded using Huffman coding. It is a variable length coding Huffman Encoding. Shorter codes are allotted for frequently occurring long sequence of 1's & 0's. Decoding requires Huffman table and dequantization table. Inverse DCT is taken employing eq (3). The data blocks so obtained are combined to form complete image. The schematic of encoding and decoding is shown in fig. 8.21.

$$f_{x,y} = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 c_u c_v f_{u,v} \cos\left(\frac{(2x+1)u\pi}{16}\right) \cos\left(\frac{(2y+1)v\pi}{16}\right) \quad \text{eq(3)}$$

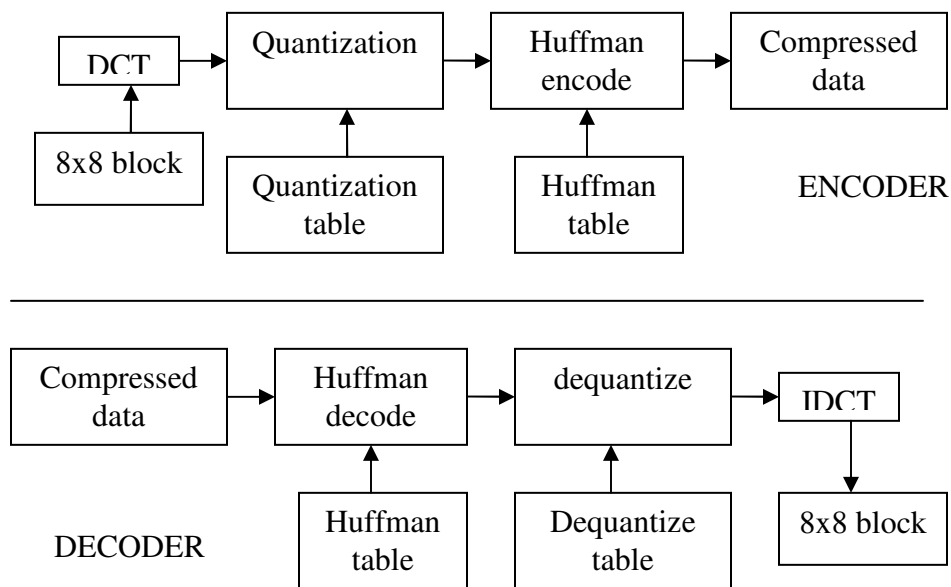


Fig. 8.21: JPEG Encoder & Decoder