

Assignment :- 2

Q.1. Difference between IIR & FIR filters			
→ Characteristics		IIR Filter	FIR Filter
1. No of necessary multiplications		Least	Most
2. Sensitivity of filter co-efficient		Can be high for very low.	
3. Probability of overflow errors		Can be high for very low.	
4. Stability		Depends on system design	Guaranteed.
5. Linear Phase		No	Guaranteed.
6. Can simulate prototype analog filters.		Yes	No.
7. Required b/w memory		Least	Most
8. H/W filter control complexity		Moderate	Simple.
9. Availability of design software		Good	Very good.
10. Ease of designer Complexity of design.		Moderate complicated	Simple.
11. Difficulty of quantization noise analysis.		Complicated	Least complicated
12. Supports adaptive filtering.		Yes	Yes.

Q.2 Applications of DSP in Radar Systems.

- As a matter of fact, radar is used for detecting stationary & moving objects. Fig. shows the block diagram of a modern radar system. Here, in the transmitter side, the signals are generated & transmitted with the help of an

antenna.

- When these signals strike the target object, a portion of the signal is echoed back. This echoed signal is received by the radar system. Depending upon the time duration between the transmitted and received signals, the distance at which the target is located can be identified.
- The main parts of a radar system are the antenna, the tracking computer & the signal processor. In fact, the tracking computer is called the brain of the system.

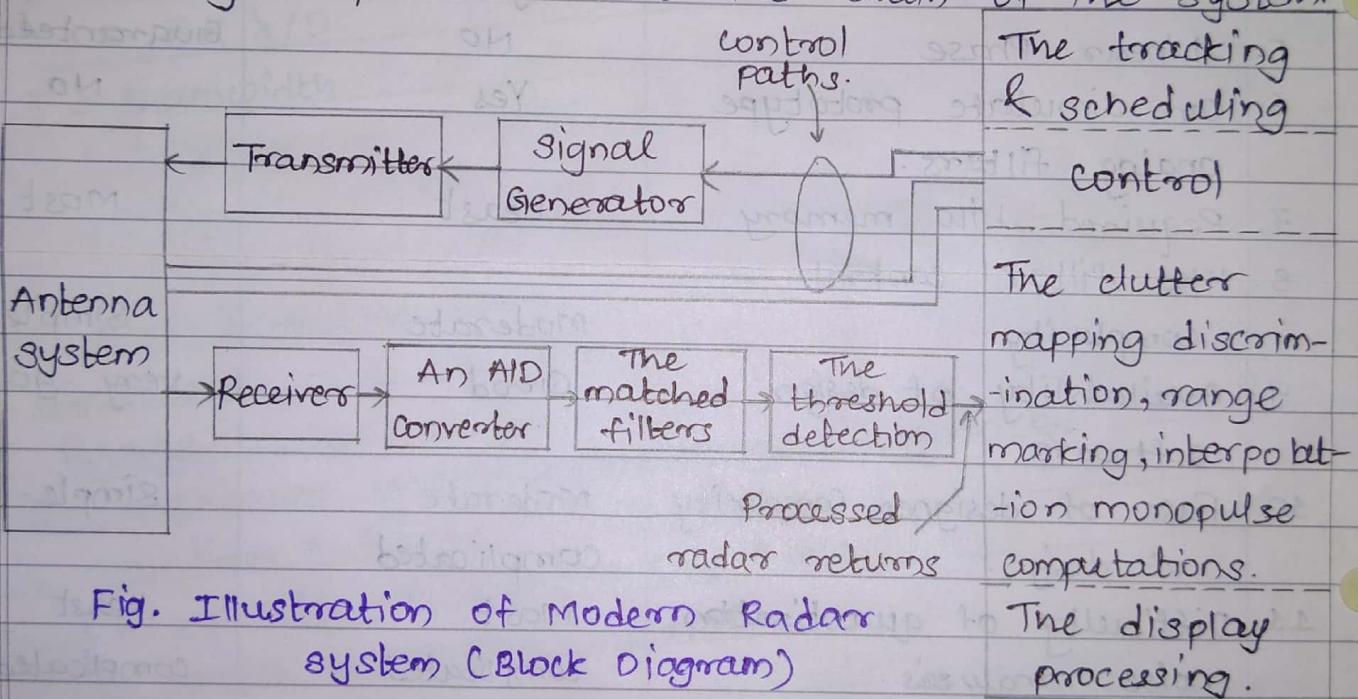


Fig. Illustration of Modern Radar system (Block Diagram)

- The tracking computer serves the following main functions:-

- It schedules the appropriate antenna positions & transmitted signal as a function of time.
- It seeks track of important targets.
- It runs the display.

The major functions of signal processors can be listed as under:-

(ii) It provides matched filtering.

(iii) It provides removal of useless information - threshold detection.

The tracking computer controls the entire operation of the radar system. Some important. Thus, we have

$$\Delta f = \frac{2V}{c} f_0$$

The tracking computer controls the entire operation of the radar system. some important radar parameters are discussed in the sections to follow:

1) Antenna Beamwidth:-

The beamwidth of an antennal is given by,

$$\beta \propto \lambda / D$$

here, β = Beamwidth

λ = Wavelength

D = Antenna width

For a pencil beam, the antenna geometry is kept symmetric. Also, β is the same in both the horizontal & vertical dimensions.

2) Radar Range :-

The maximum unambiguous range is given by

$$R_{\max} = cT/2$$

where, c = velocity of light $\approx 3 \times 10^8$ m/s.

T = pulse repetition interval.

3) Radio Range Resolution:-

In case when, two targets are present near each other, then the ability of the radar to detect these targets can be measured by the range resolution ΔR . If the signal is having constant frequency, the ΔR is determined by the pulse width. If the pulse width is narrowed, then range resolution can be improved but the maximum range is reduced by decreasing the average power.

4) Doppler Filtering.

As a matter of fact, moving targets can be identified by using the Doppler effect. When a continuous sine wave of frequency f_0 is transmitted & the target is moving with a constant velocity, then the received echo signal frequency will be $f_0 + \Delta f$.

Thus, we have,

$$\Delta f = \frac{2v}{c} f_0 = \frac{2v}{\lambda}$$

here, f_0 = carrier frequency.

v = Target velocity.

λ = Wavelength.

It may be noted that pulsed Doppler signals may be used to get range & velocity resolution.

5) Signal Designing :-

Transmitting narrow pulse provides good range but at the same time, it provides poor velocity measurement. In fact, a wide pulse of single frequency provides good velocity but bad range information.

- Let us consider the radar model as shown in fig. Let the signal be generated digitally & transmitted through an analog filter.

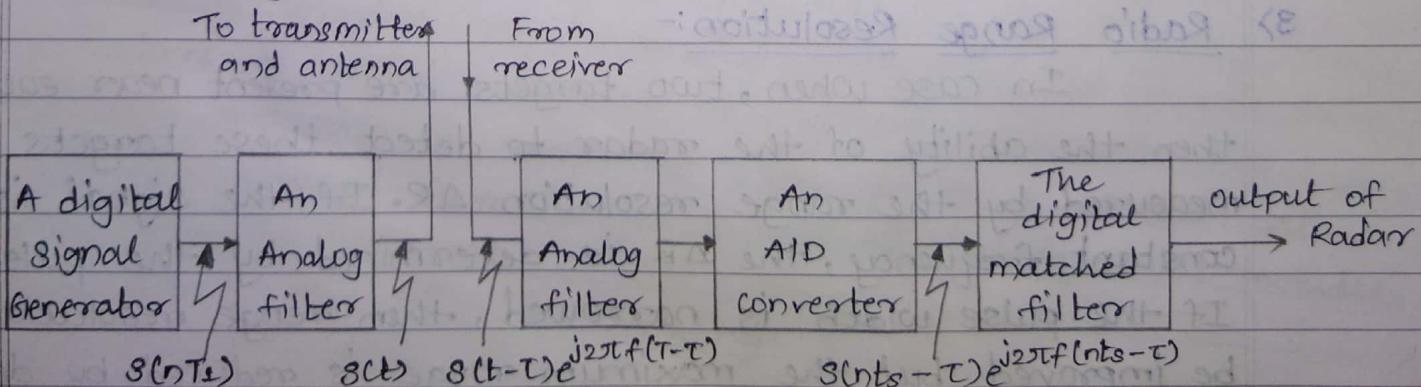


Fig. Illustration of block diagram of a radar model.

- The transmitted signal is $s(t)$. The received signal is $s(t-\tau)e^{j2\pi f(t-\tau)}$ which is delayed & frequency shifted.
- The received signal is made to pass through an analog filter, A/D converter and then through a digital matched filter.
- The input signal to the matched filter will be $s(nT_s - \tau) e^{j2\pi f(nT_s - \tau)}$

Q.3. Write applications of DSP in Image Processing.

- As a matter of fact, any two dimensional information-bearing function is known as an image. Images are specified by arrays of real or complex numbers represented by finite number of bits.
- Image signals are two-dimensional & speech signals are one-dimensional signals. Smallest element of an image is known as Picture Element or Pixel.
- Image processing is referred as the manipulation of two-dimensional signals with the help of a digital computer.
- The purpose of image processing is to improve the visual appearance of image for human viewing & perception & to prepare images for measuring of their various features.
- Image processing has found application in the following areas: Medical imagine, remote sensing, processing of images from Radars & sonars, storage of business documents, etc.
- Image processing is concerned with processing of electrical signals extracted from images by digital techniques. It includes the following topics:
 - (i) Image Formation & Recording
 - (ii) Image compression.
 - (iii) Image restoration.
 - (iv) Image enhancement.
- Above operations are possible because of high-tech digital computers, artificial intelligence & advanced version of software.

Q. 4. Compare DSP processors & General Purpose microprocessors.

Parameter	DSP Processors	Gen. Purp. Microprocessors
1. Instruction cycle.	Instructions are executed in single cycle of the clock i.e. total instruction cycle.	Multiple clock cycles are required for execution of one instruction.
2. Instruction execution.	Parallel execution is possible.	Execution of instructions is always sequential.
3. Operand fetch from memory.	Multiple operands are fetched simultaneously.	Operands are fetched sequentially.
4. Memories.	Separate program and data memories.	Normally no such separate memories are present.
5. On-chip / off-chip memories	Program sequencer and data memories are present on-chip & extendable off chip.	Normally on-chip cache memory is present. Main memory is offchip.
6. Program flow control.	Program sequencer & instruction register & instruction cache.	Program counter maintains the flow of execution.
7. Queuing / pipelining	Queuing is implicit through instruction register & instruction cache.	Queuing is performed explicitly by que registers for pipelining of instructions.
8. Address generation.	Addresses are generated combinably by DAC's and program sequencer.	Program counter is incremented sequentially to generate addresses.
9. Address / data bus multiplexing	Address & data buses are not multiplexed. They are separate onchip as well as off chip.	Address / data buses can be separate on the chip usually multiplexed offchip.
10. Computational units.	Three separate computational units ALU, multiplier, shifter.	ALU is the main computational unit.

	-ional units: ALU, MAC and shifter.	-ional unit.
11. on-chip address	separate address & data buses. Address & data buses are	for program memories and the two buses on the chip.
	data buses.	data memories & result bus, i.e. PMA, DMA, PMD, DMD &
	R-bus.	

12. Addressing modes. Direct & Indirect addressing. Direct, indirect, register, register is supported. indirect, immediate, etc. addressing modes are supported.

13. Suitable for. Array Processing General purpose processing.

Q.5. Write a short note on following with reference to P-DSP's :

1. Von Neumann Architecture & 2. Harvard Architecture.

→ It may be noted that the MAC operation with the move (ie. the MACD instruction) requires four memory accesses per instruction cycle. (An instruction cycle is the time that elapses since an instruction is fetched till the particular instruction completes execution including the time taken for writing the result into a register or memory. Many of the instructions in P-DSP's including the MACD instruction require only one processor clock period / instruction cycle. In the conventional microprocessors one instruction cycle corresponds to several clock periods). The four memory access / clock period required for the MACD instructions are as follows :

1. Fetch the MACD instruction from the program memory.
2. Fetch one of the operands from the program memory.
3. Fetch the second operand from the data memory.

4. Write the content of the data memory with address dm_1 into the location with the address dm_{avg+1} .

The relatively static impulse response coefficients are stored in the program memory & the samples of the input data are stored in the data memory. If the MACD instruction is to be executed in a machine with von Neumann architecture, it requires four clock cycles. This is because in the von Neumann architecture in fig. 1. there is a single address bus & a single data bus for accessing the program as well as data memory area. One of the ways by which the number of clock cycles required for the memory access can be reduced is to use more than one bus for both address & data. For example in the Harvard architecture shown in fig. 2.

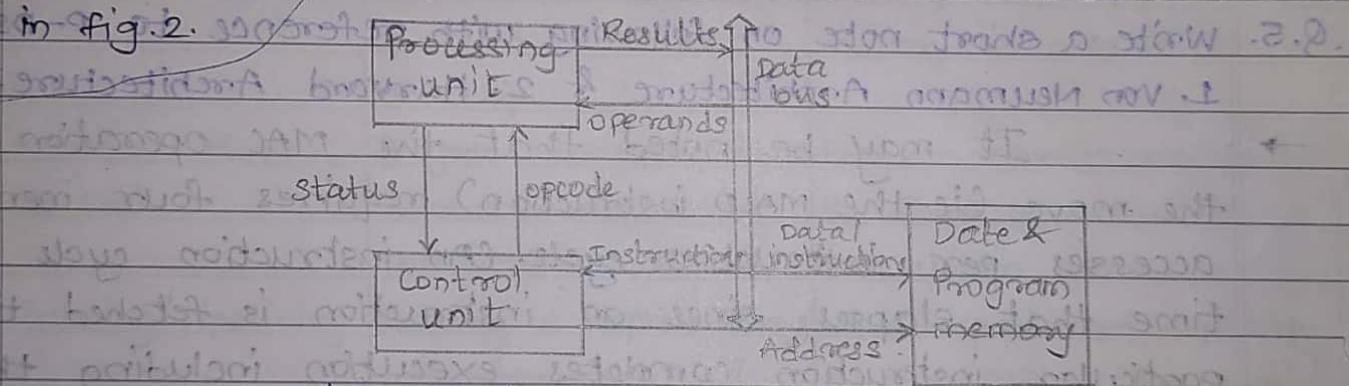


Fig. 1. Von Neumann Architecture

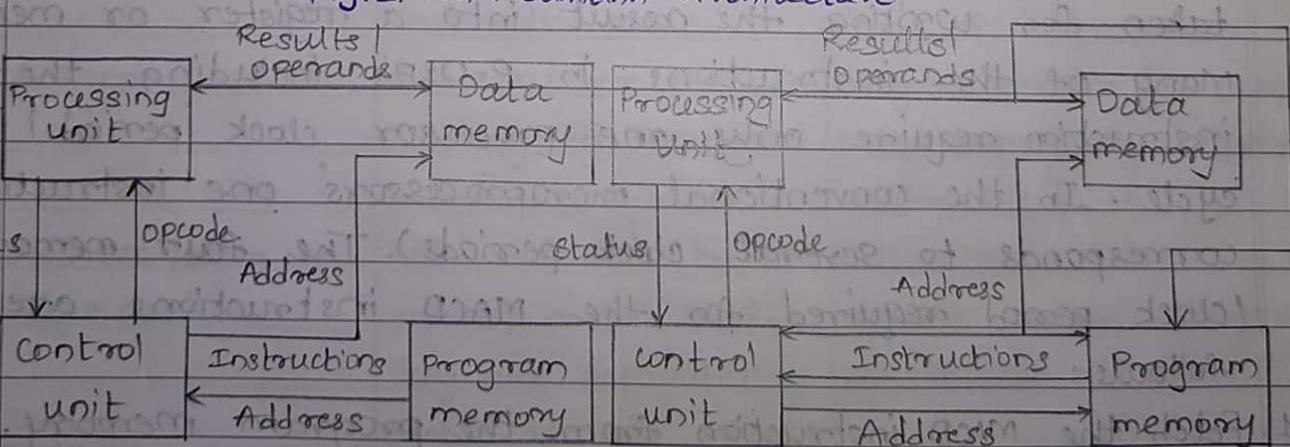


Fig. 2. Harvard Architecture

Fig. 3. Modified Harvard Architecture.

there are two separate buses for the program & data memory. Hence the content of program memory & data memory can be accessed in parallel. The instruction code can be fed from the program memory to the control unit while the operand is fed to the processing unit from the data memory. The processing unit consisting of the registers & processing elements such as MAC units, multiplier, ALU, shifter, etc., are also referred to as data path. The P-DSP's follow the modified harvard architecture shown in Fig.3. One set of bus is used to access a memory that has both program & data & another that has data alone. Data can also be transferred from one memory to another. The modified Harvard architecture is used in several P-DSP's, for example P-DSP's from Texas Instruments & Analog devices.

With the Harvard architecture, the number of memory accesses /clock cycle was shown to be two. This can be increased further by using more number of buses. For example, by using three separate address & data buses, the number of memory accessed clock cycle can be increased to three. Motorola DSP5600X, DSP96002, etc. have three separate buses. TMS320CS4X has four address buses.

Since the cost of an IC increases with the number of pins in the IC, extending a number of buses outside the chip would unduly increase the price. Hence the P-DSP's use multiple buses only for connecting the on-chip memory to the control unit & data path. For accessing off-chip memory only a single bus is used for accessing both the program memory & data memory. Because of this, any operation that involves an off-chip memory is slow compared to that using the on-chip memory.

Q.6. Explain how a higher throughput is obtained through VLIW architecture. Give an example of a DSP that uses VLIW architecture

→ Another architecture used for P-DSP's, for example in TMS320C6X, is the very long instruction word (VLIW) architecture. These P-DSP's have a number of processing units (data paths). In other words, they have a number of ALU's, MAC units, shifters, etc. The VLIW is accessed from memory & its used to specify the operands & operations to be performed by each of the data paths. As shown in fig., the multiple functional units share common multiplexed register file for fetching the operands & storing the results. Parallel random access by the functional units to the register file is facilitated by the read / write cross bar. Execution of the operations in the functional units is carried out concurrently with the load / store operation of data.

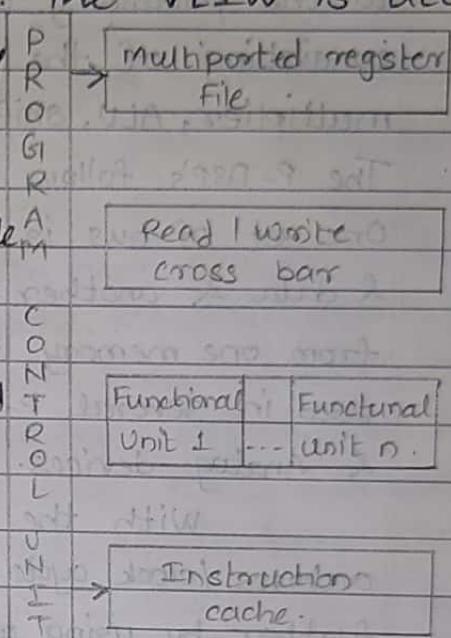


Fig. Block Diagram of the VLIW architecture

The performance gains that can be achieved with VLIW architecture depends on the degree of parallelism in the algorithm selected for a DSP application & the number of functional units. The throughput will be higher only if the algorithm involves execution of independent operations. For example in fig. by using eight functional units, the time required for convolution can be reduced by a factor of 8 compared to the case where a single functional unit is used.

However, it may not always be possible to have independent stream of data for processing. Further the number of functional units is also limited by hardware cost for the multiplexed register file & cross bar switch.

Q.7. Explain what is meant by instruction pipelining. Explain with one example how pipelining increases throughput efficiency.

One of the approaches adopted for increasing the efficiency of the advanced microprocessors as well as P-DSP's is instruction pipelining. An instruction cycle starting with the fetching of an instruction & ending with the execution of the instruction including the time storage of the results can be split into a number of microinstructions. Execution of each of the microinstructions is also referred to as one phase of an instruction. For example, an instruction cycle requiring four microinstructions can be said to be in four phases as well follows:-

1. Fetch phase in which the instruction is fetched from the program memory.
2. Decode phase in which the instruction is decoded.
3. Memory read phase in which the operand required for the execution of the instruction may be read from the data memory.
4. Execution phase in which execution as well as the storage of the results in either one of the registers or memory is carried out.

Value of T	Fetch	Decode	Read	Execute
1	I1			
2		I1		
3			I1	
4				I1
5	I2			
6		I2		
7			I2	
8				I2
9	I3			
10		I3		
11			I3	
12				I3

Fig. 1. Instruction cycles of processor with no pipelining

Value of T	Fetch	Decode	Read	Execute
1	I1			
2	I2	I1		
3	I3	I2	I1	
4	I4	I3	I2	I1
5	I5	I4	I3	I2
6	I6	I5	I4	I3
7	I7	I6	I5	I4
8	I8	I7	I6	I5
9	I9	I8	I7	I6
10		I9	I8	I7
11			I9	I8
12				I9

Fig. 2. Instruction cycle of a processor with pipelining.

Each of the above microinstructions may be carried out separately by four functional units. Let us assume that each of the above four phases take equal time for completion. In this case in a conventional microprocessor with no pipelining, each of the functional units is busy only 25% of the time. This is because only one instruction is processed at the CPU at a time. Fig. 1. shows when each of the functional unit is busy when a program containing three instructions I_1, I_2, I_3 is executed.

The functional units can be kept busy almost all the time by processing a number of instructions simultaneously in the CPU. For example, in a machine with four functional units, four instructions I_1, I_2, I_3 & I_4 can be processed simultaneously as shown in fig. 2. When I_1 enters the decode phase I_2 can enter the opcode fetch phase. When I_1 enters the operand read phase I_2 enters the decode phase & I_3 enters the opcode fetch phase. When I_1 enters the execute phase I_2 enters the operand read phase I_3 enters the decode phase & I_4 enters the opcode fetch phase. The pipeline is fully loaded now & all the functional units have useful work to do. The instructions that follow I_4 keep the functional units busy till the program is exited. Let T denote the time required for each phase of the instruction. One clock cycle of the processor corresponds to T . In a period of $12T$ only three instructions can be executed in a machine without pipelining. In the same period nine instructions can be executed as shown in fig. 2. Hence the throughput is increased by a factor of 3 in this case.

It may be noted that the initial latency of a machine with four phases is $4T$. Hence for executing a program with N instructions, the time required for execution is $(N+4)T$. With a non-pipelined machine, the time required for executing N instruction is $4NT$.

Instruction pipeline shown in fig. 2. corresponds to a highly optimistic case. In the case of processors requiring single clock cycle for execution for each of the instructions in the program, the throughput shown in fig. 2. can be achieved. This is normally achieved with restricted instruction set computers (RISC). However in complex instruction set computers (CISC), there are also instructions with multiple word requiring multiple clock cycles for execution. In this case all the functional units cannot be kept busy all the time. For example, in the case of call & branch instructions of a P-DSP, four phases or T states are required for the call/branch instruction to exit execution phase. By that time - two more single word instructions or one double instruction enters the instruction pipeline. These instructions should not be executed. Hence two words have to be flushed out of the instruction pipeline before the instructions are fetched starting from the new program address.

Q.8. What are the on-chip peripherals connected to DSP processors?

→ The DSP processors TMS320C54X ('54X) devices have different on-chip peripherals as:-

a) General purpose I/O pins :-

Each 54X device has 2 general purpose I/O pins :- BIO & XF. BIO is an input pin used to monitor the status of external devices. XF is a software controlled output pin that allows to signal external devices.

b) Programmable Bank switching logic. wait-state Generator:- The software programmable wait-state generator extends external bus cycles upto seven machine

cycles to interface with slower external memory & I/O devices. The wait-state generator is incorporated without external hardware.

c) Programmable Bank switching Logic:

The programmable bank switching logic can automatically insert one cycle when an access crosses memory bank boundaries inside program memory or data memory. This extra cycle prevents bus contention by allowing memory devices to release the bus before other devices start acquiring it.

d) Host Port Interface.

The host port interface is an 8-bit parallel port that provides an interface to a host processor. The information is exchanged between '54x & the host through the '54x on-chip memory.

e) Hardware Timer

The '54x features a 16-bit timing circuit with a 4-bit prescalar. The timer can be stopped, restarted, reset or disabled by specified status bits.

f) Clock Generator

Clock generator consists of an internal oscillator & a PLL circuit. Internal crystal resonator or external clock source can be used to drive the clock generator. The PLL circuit is used to generate an internal CPU clock.

g) Synchronous Serial Ports:-

The synchronous serial ports are high-speed full-duplexed serial ports used for direct communication with serial devices like codecs, ADC, etc. Each synchronous serial port can operate at upto 1/4th the machine cycle rate.

The transmitter & receiver for this port are double buffered & individually controlled by maskable external interrupt signals.

h) Buffered Serial Ports:-

* **TDM** The buffered serial port is enhanced with an autobuffering unit & is clocked at full machine cycle rate. It is fully-duplexed & double-buffered to offer a flexible data stream length.

i) TDM Serial Ports:-

A TDM serial port is a synchronous port that allows time-division multiplexing of the data. It is often used for multiprocessor applications.

Q.9. Explain different Addressing Modes in DSP processors.

→ The DSP processors belonging to TMS320C3X series support 6 addressing modes:-

a) Register Addressing.

The syntax of register addressing mode is as follows :-

"mnemonic src, dst"

The 'mnemonic' can be any assembly instruction code that supports register addressing.

'src' is source register & 'dst' is the destination

register. The TMS320C3X processor register file contains eight extended precision registers (R0-R7), which can be used.

b) Direct Addressing.

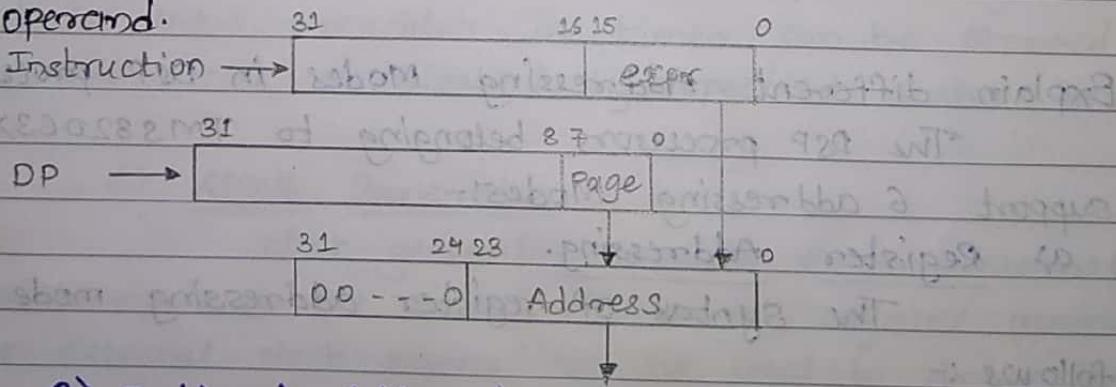
The no. of address bits of TMS320C3X processor is 32 bits & addressable data space with 32 bits is 16M words. The data space is organized in 256 pages, each page of 64K words. Data - page pointer (DP), a 32-bit register holds the data page value.

The location of operand in a specific page is given in the instruction code. The data page pointer is to be loaded first before the data access using direct addressing.

In direct addressing, the least significant 16-bits of instruction code & the least significant 8-bits of OP are combined to form a 24-bit address of the data operand present in data space.

The syntax for direct addressing is:-
"mnemonic @ expr, dst"

where expr is the LSB 16-bit value of the operand.



c) Indirect Addressing.

In indirect addressing the address of an operand in memory is specified through the content of an Auxiliary Register (AR). There are 8 AR's (AR0-AR7)

in 'C3X processors. Each AR is of 32 bit, the LSB 24 bits are used to specify the address. The rest of the bits are loaded to AR, are not modified by the instruction.

The instruction code format is shown below.

MSB	MOD	Arn	Disp	LSB
	5bit	8bit	01518 bit	

The 5bit MOD field specifies the type of indirect addressing. The 8-bit Arn specifies the content of the AR (i.e. Operand Address) & the Disp field is used to modify the address value after the memory access.

d) short-immediate Addressing mode.

In immediate addressing the operand can be given directly in the instruction. The syntax is:-

"mnemonic expr"

The expr field can be a 16-bit or 24-bit value.

In short immediate addressing, the operand is a 16-bit value contained in the 16 LSB of the instruction word (expr). It can be a 2's complement integer, an unsigned integer or a floating-point number.

e) Long-immediate Addressing.

The syntax is :-

"mnemonic expr"

In long-immediate addressing, the operand is a 24-bit immediate value contained in the 24 LSB's of the instruction word.

f) PC- Relative Addressing.

The program counter (PC)- relative addressing

is used for branching. It loads 16-bit or 24-bit LSB's of the instruction word into the PC.

The syntax is :- "mnemonic src".

The 'mnemonic' can be branch, call and repeat instructions codes & 'src' is a label or address.

Q. 10. Describe speech Processing system with the help of a

Block Diagram :-

→ Speech Recognition: Speech recognition involves in putting of information into a computer using human voice & then computer quantizes this information & recognizes human speech.

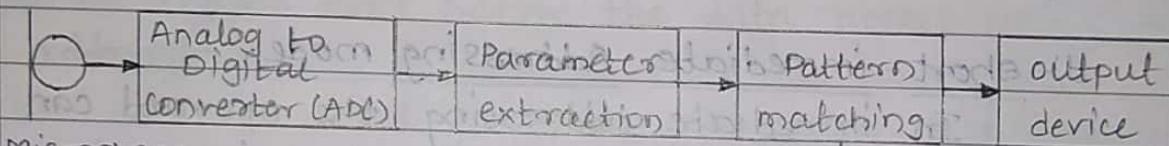
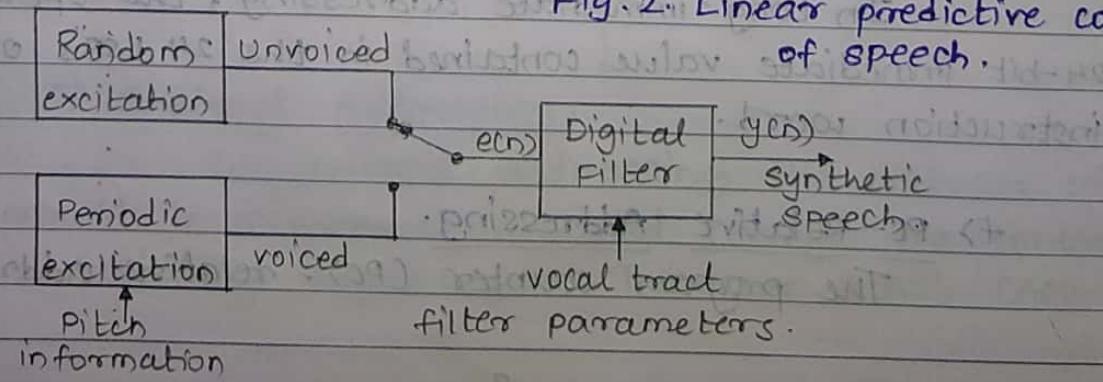


Fig. 1. Block Diagram of
Speech Recognition system.

Template
memory

As shown in fig. 1, speech voice is input through microphone. The input analog signal is digitized using A/D converters. This digitized output is stored in memory. Recognition process starts after storage. Then here again the spoken word is digitized & its template is compared with template of memory. As soon as the match occurs, this indicates that the word has been recognized & the same is informed to the user.

Fig. 2. Linear predictive coding
of speech.



DSP processor plays an important role in extracting an important parameter called template from spoken word. The whole operation is done accurately by DSP processors.

Following performance parameters that greatly affect the above system too are overcome by DSP processor:-

(i) Noise.

(ii) Microphone characteristics.

(iii) Pause taken between two words.

(iv) correct pronunciation of word.

- Speech Synthesis:-

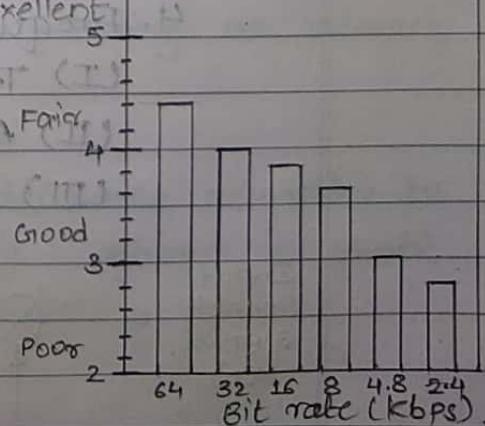
In this system, human voice is produced. To achieve this, we use Linear Predictive Coding (LPC) technique.

In human speech, voiced sound & unvoiced sound are present, voiced sound is nothing but the vowels. These represent the air flow through vocal cords. While unvoiced sound represents the noise created by forcing air in the vocal tract.

Unvoiced sound is generated by random excitation while voiced sound is generated by periodic excitation. Digital filters perfectly simulates the behaviour of the vocal tract. Thus, we get human speech.

- Training Phase & coding:-

Coding algorithms seek to minimize the bit rate in the digital representation of a signal without an objectionable loss of signal quality in the process. Quality of speech refers to how natural it sounds as compared to the words uttered by a human being. High quality is attained at low bit rates by exploiting signal



redundancy as well as the knowledge that certain types of coding distortion are imperceptible because they are masked by the signal. Intelligibility & quality of speech are measured in terms of a performance index called Mean Opinion Score (MOS). To determine the MOS of a speech source, a large number of listeners are requested to rate the given sample of speech & allot marks as shown in fig. 3.

These numerical values, after averaging, yield the MOS score for the coder. For a high quality coder, the MOS values range between 4.0 & 4.5.

* Methods of Speech Coding :-

Various methods of speech coding may be classified as under:-

(i) Waveform Coding :-

1. Pulse- coding Modulation (PCM).

2. Adaptive Pulse- code Modulation (APCM).

3. Linear Predictive coding :-

(I) Differential Pulse- code Modulation (DPCM).

(II) Adaptive Differential Pulse- code Modulation (ADPCM).

(III) Delta Modulation (DM).

(IV) Adaptive Delta Modulation (ADM).

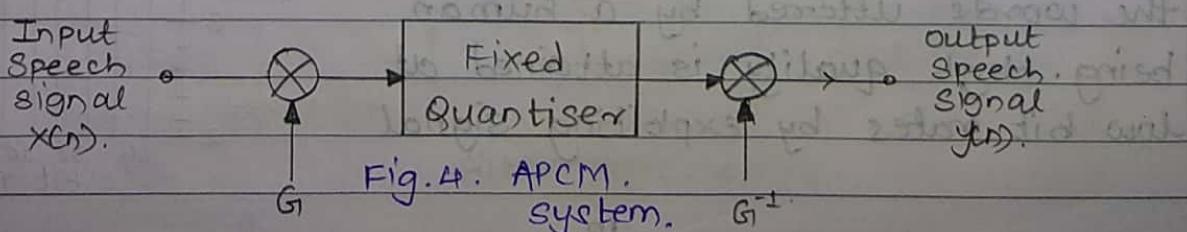
(V) Continuously variable-slope Delta modulation.

(C) Frequency-Domain Coding :-

(I) Transfer coding (TC).

(II) Adaptive Transform Coding (ATC).

(III) Subband Coding (SBC).



(ii) Baseband coding (BBC).

(iii) Narrowband coding.

(I) Pitch- Excited coder,

(II) Optimal scalar Quantisation (OSQ).

(III) vector Quantisation (VQ).

(IV) Segment Quantisation (SQ).

Q.11 Explain in detail :- Methods of speech coding.

→ Waveform coding aims at reproducing the speech waveform as faithfully as possible. The waveform coders are able to produce high-quality speech at high enough bit rates. Waveform coding includes:-

1. Pulse - code Modulation (PCM).

PCM is the simplest coding system, a memoryless quantizer & provides essentially transparent coding of telephone speech at 64 kbps. This waveform coding scheme employs uniform quantisation in which the range of signal value of $x(n)$ is subdivided into small equal bins, each of width A , the step size & all signal values falling within a bin are decoded to one value in that particular bin.

Dithering is used to weaken the signal dependent character of quantisation noise. In Dithering, a pseudorandom sequence is added to the signal prior to its quantisation & subsequently at the receiver the same sequence is subtracted from the decoder output yielding an almost white quantising error.

2. Adaptive Pulse - code Modulation (APCM)

Here, the quantiser adjusts the step size according to the short-term amplitude of the signal. Fig. gives a schematic illustration of the Adaptive Pulse - code Modulation system.

3. Linear Predictive Coding (LPC)

In this scheme, the waveform coding is based on a linear prediction model of speech at any instant as a predicted value plus an error term. For rates of 16 kbps & lower, high speech quality is achieved by using LPC.

The most familiar examples of LPC are as under:

(i) Differential Pulse- code Modulation (DPCM).

In this system, predictor has the simple form as,

$$a(z) = 1 + a_1 z^{-1}$$

where a_1 is a fixed negative number calculated by determining the long-term average of the signal spectrum & computing the optimal $a(z)$ for $P=1$. Fixed predictors of higher order, $P > 1$, can also be used but beyond $P=4$, the approach has not much utility.

(ii) Adaptive Differential Pulse- code Modulation (ADPCM).

In ADPCM, the predictor changes or modifies itself according to the requirement of the signal. ADPCM provides high quality speech at 32 kbps. The speech quality is slightly inferior to that of 64 kbps PCM. ADPCM at 32 kbps is widely used for expanding the number of speech channels by a factor of two, particularly, in private networks & international circuits. It is also the basis of low-complexity speech coding in several proposals for personal communication networks.

(iii) Delta Modulation (DM).

In DM, the waveform sampling rate is much larger than the Nyquist rate to increase the adjacent sample correlation & the quantisation of the residual or difference signal into a one-bit or two-level strategy enabling the transmission of the difference only. Here, bit rate is equal to the sampling rate.

(iv) Adaptive Delta Modulation (ADM)

In ADM, variable step size is used depending on the quantizer outputs. In a two-level quantizer, observation of a single sample of quantiser output does not indicate slope overload distortion or granularity. Therefore, a sequence of quantiser outputs is required for the desired step size adaptation.

(v) Continuously - variable slope- Delta (CVSD) Modulation.

CVSD modulation technique includes smooth adaptation of the step size with a time constant 5-10 ms. CVSD coders are more tolerant to channel errors & decrease the granular noise in output speech at the expense of a high slope overload distortion. CVSD coders give a clean-sounding speech at bit rates less than 24 Kbps.

4. Frequency Domain coding.

Frequency domain coders uses the non-flat short-term spectral characteristics of the speech signal, constituting the redundancy, & the signals at a level lower than the psychoacoustic threshold, which are irrelevant to the ear & are called the Irrelevancy.

Redundancy & irrelevancy removal in frequency domain coders is accomplished by decomposing the source spectrum into frequency bands containing uncorrelated spectral components of the signal. These components are quantized separately. At the receiver, the components are subjected to an inverse transformation to obtain the reconstructed signal.

There are two main forms of speech coding in the frequency domain as follows.

(i) Transform Coding (TC).

TC is a frequency domain technique in which a block of input samples is taken & linearly transformed using DFT or DCT computation via a FFT algorithm. Then, the transformation of the

Signal is efficiently coded by assigning bits to transform coefficient. At receiver, an inverse transformation is used to reconstruct the speech signal.

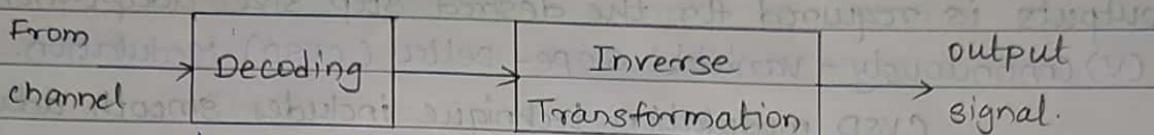
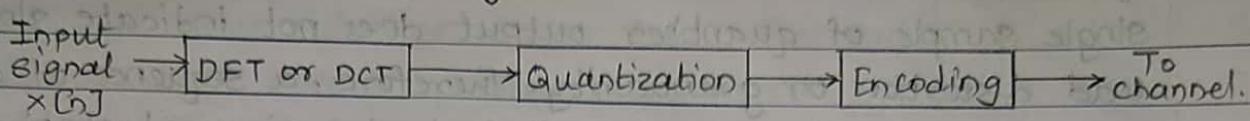


Fig. 1: Transform coding system.

In TC, total number of bits available to quantize the transform coefficients remains constant whereas in adaptive transform coding, the bit allocation to each coefficient changes from frame to frame. This dynamic bit allocation is controlled by the time varying characteristics of speech, which have to be transmitted as side information. The side information is also used to determine the step sizes of the various coefficient quantizers. The number of bits assigned to each transform coefficient is proportional to its corresponding spectral energy value.

(ii) Adaptive Transform Coding (ATC).

In ATC, important thing is the determination of bit assignment or the number of bits used for quantising individual transform coefficient. In the presence of channel errors, the received side information may yield inaccurate bit assignment. This may lead to wrong decoding of the received bit stream into transform coefficients which spoils the quality of speech. Side information should be safeguarded against channel noise.

(iii) Subband Coding.

The speech signal is applied to an analysis filterbank consisting of a set of band pass filters. This digital filtration divides the speech signal into a number of non-overlapping frequency band. By additive recombination of the set of subband

Signals without quantisation as shown in fig, we can generate the

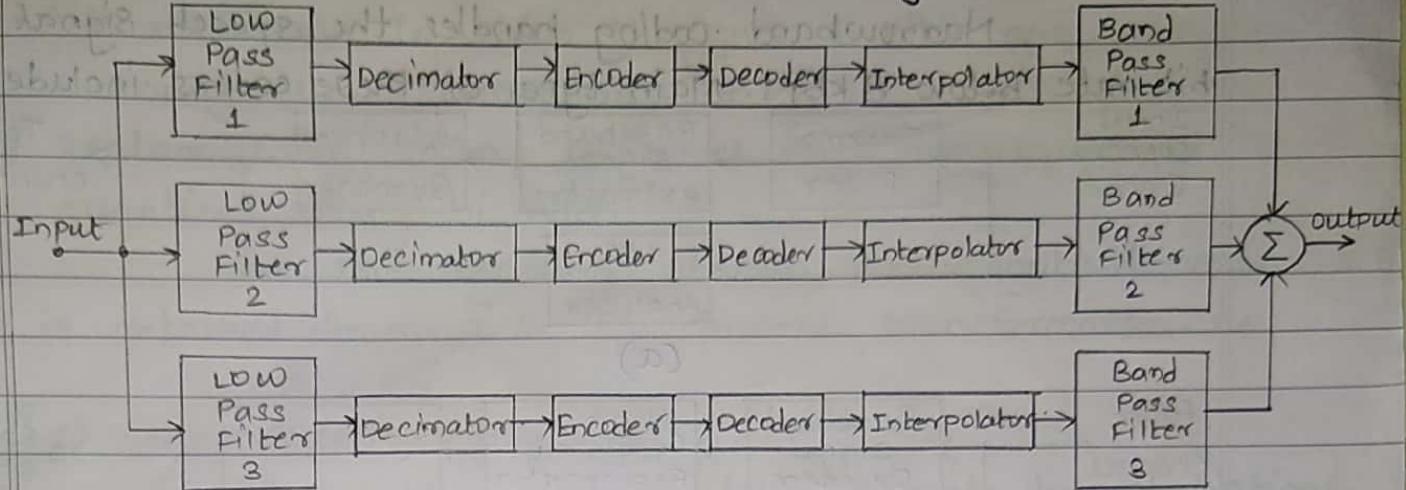


Fig.2 Subband coder

original speech signal.

Each band of non-overlapping frequency is passed through a low-pass filter (LPF) & decimated. Each band is separately quantized & coded using PCM, DPCM etc. & ~~inserting~~ transmitted. In the receiver, the sampling rate of each band is raised to that of the source signal by inserting proper number of zero samples. Then, subband samples appear at the bandpass outputs of the synthesis filter bank. The sum of bandpass outputs is equal to source signal without quantisation as shown in fig.2.

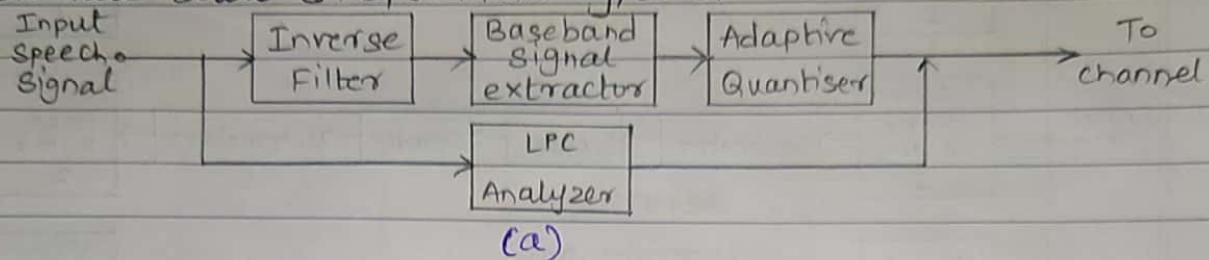
(iii) Baseband coding.

At bit rates $< 12 \text{ kbps}$, the speech quality of waveform encoders falls down. If we wish to achieve higher speech quality at lower bit rates, we must consider input/output signal matching characteristics instead of quality resemblance of output speech with input speech.

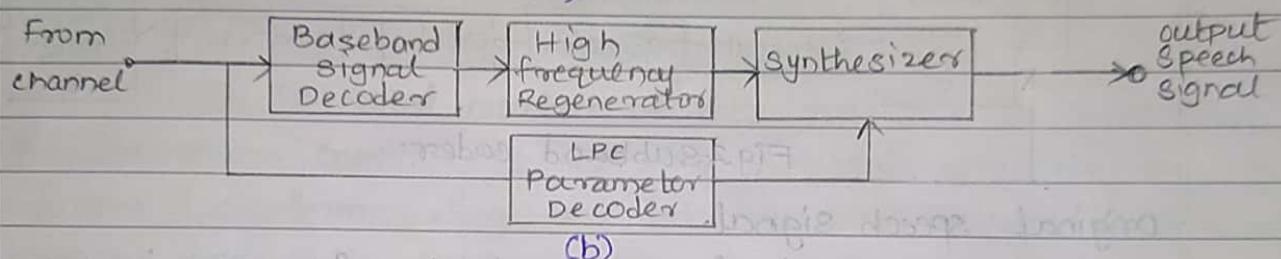
According to fig., in baseband coding, the input signal is inverse filtered to extract the baseband signal. The full signal is not transmitted, only a portion is sent for which a lower transmission rate suffices. In the receiver, the baseband signal is decoded.

(iv) Narrowband Coding.

Narrowband coding handles the speech signal with bit rate below 8 kbps. Main types of these coders include.



(a)



(b)

Fig. 3. Baseband coding (BBC) system (a) Transmitter (b) Receiver

a) Pitch-Excited coder.

This method is based on a mixed source model of speech as shown in fig. 4. Here, a pulse source is used for exciting low frequencies.

A white noise source, excites the high frequencies. The energy level of each frame is decided by a gain parameter. Generally, pitch is quantized to six bits & gain to five bits.

b) Optimal Scalar Quantization.

This method exploits the correlation among a set of P parameters represented by a random vector \vec{x} to be quantized to b bits, to decrease the bit rate. It is a three-step process,

- **Parameter Decorrelation** :- In which a new parameter vector \vec{y} is formed using a matrix Q whose columns are the eigen vectors of the covariance matrix of random vector

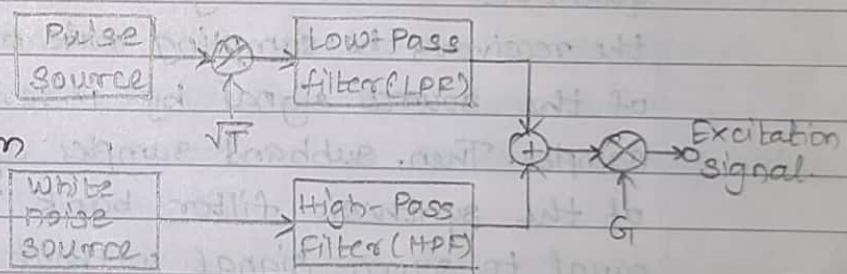


Fig. 4.. Pitch-Excited model.

\vec{x} , as $\vec{y} = Q\vec{x}$

- Bit allocation:- In which the given b bits are assigned among the P components of \vec{y} .
- Scalar Quantisation:- In which each component y_i is quantized using b_i bits.

At the receiving station, the quantised vector \vec{x} is obtained from \vec{y} by the inverse transformation, i.e.

$$\vec{x} = Q^{-1}\vec{y}$$

c) Vector Quantization:-

This method reduces the bit rate by statistical dependence among parameters beyond correlation. It is based on clustering procedure. Vector quantisers consume large amount of memory & computational space. They are used at low bit rates, one bit per parameter in the region where scalar quantisers are insufficient.

d) Segment Quantization:-

Segment quantisation utilizes interdependence between frames. Segments of a suitable size are obtained by dividing the speech into phoneme boundaries. A diphone is the region separating the equilibrium state of one phoneme from that of the next. By warping, two segments can be made to possess an equivalent number of frames. While in Dynamic Time Warping, the distance between adjacent segments is minimized. By dynamic programming, the computational load is decreased.

Q. 12.

Applications of DSP Processors:-

- C1X, C2X, C2XX, C5X, C54X :- Toys, Hard disk drives, modems, cellular phones & active car suspensions.
- C3X:- Filters, Analysers, Hi-Fi systems, voice mail, Imaging, Bar-code readers, Motor control, 3D graphics or scientific processing.

C4X :- Parallel-processing clusters in virtual reality, Image recognition, telecom routing & parallel processing systems.

C6X :- Wireless Base stations, Pooled modems, remote-access servers, digital subscriber loop systems, cable modems & multichannel telephone systems.

C8X :- Video telephony, 3D computer graphics, virtual reality & a number of multimedia applications.

Q. 13. Internal Architecture of DSP processors.

→ Architecture of TMS320CSX DSP's :- The block diagram of the internal architecture of C5X is shown in Fig. 1. The 320CSX DSP's are said to have advanced Harvard architecture because they have separate memory bus structures for program & data & have instructions that enable data transfer between the program & data memory area.

Bus Structures:-

Separate program & data buses allow simultaneous access to program instructions & data, providing a high degree of parallelism. For example, while data is multiplied, a previous product can be loaded into, added to or subtracted from the accumulator & at the same time, a new address can be generated. Such parallelism supports a powerful set of arithmetic, logic & bit-manipulation operations that can all be performed in a single machine cycle. In addition, the 'CSX includes the control mechanisms to manage interrupts, repeated operations & function calling. The 'CSX architecture has four buses & their functions are as follows:-

- Program Bus (PB) :- It carries the instruction code & immediate operands from program memory space to the CPU.
- Program Address Bus (PAB) :- It provides addresses to program memory space for both reads & writes.

- Data Read Bus (DB) :- It interconnects various elements of the CPU to data memory space.
- Data Read Address Bus (DAB) :- It provides the address to access the data memory space. The program & data buses can work together to transfer data from on-chip data memory & internal or external program memory to the multiplier forms for single-cycle multiply/accumulate operations.

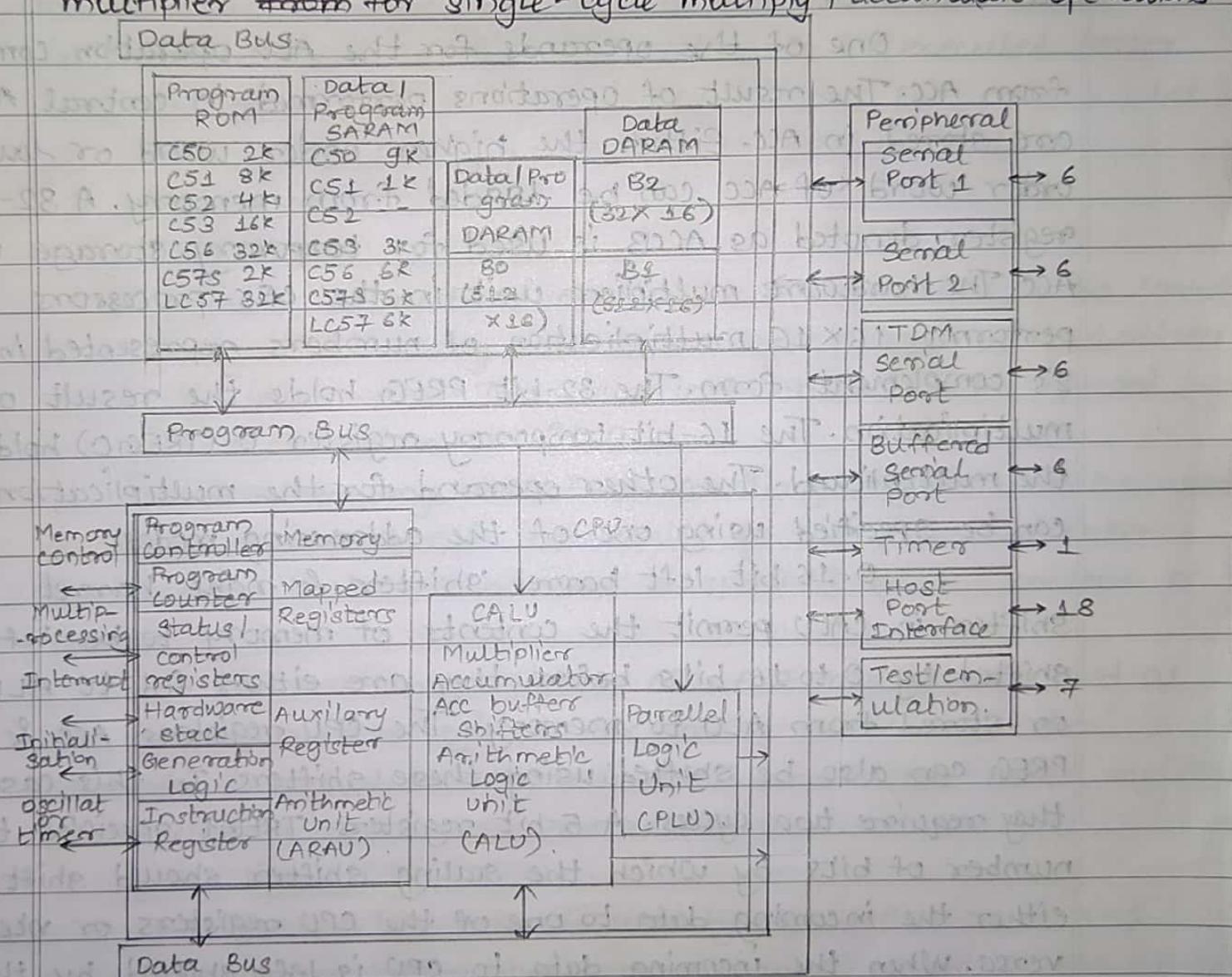


Fig. 1. Internal Architecture of CSX.

CPU registers (except ST0 & ST1), Peripheral registers & I/O ports occupy data memory space. Some of the registers & execution units in the CPU of CSX DSP processors & their

functions are as follows.

Central Arithmetic Logic Unit (CALU).

It consists of the following elements : (16X16) - bit parallel multiplier, arithmetic logic unit (ALU), accumulator (ACC), Accumulator Buffer (ACCB), Product Register (PREG1) each with 32 bits & 0-16-bit left barrel shifter & right barrel shifter.

One of the operands for the ALU operation comes from ACC. The result of operations performed in central ALU are stored in ACC. Either the higher order word or lower order word of ACC can be loaded from memory. A 32-bit register denoted as ACCB is used for temporary storage of ACC. The hardware multiplier unit in the CSX processors performs 16×16 multiplication of numbers represented in 2's complement form. The 32-bit PREG1 holds the result of multiplication. The 16-bit temporary register 0 (TREG10) holds the multiplicand. The other operand for the multiplication can be specified using one of the addressing modes.

0-16 bit left barrel shifter & right barrel shifter in CALU permit the contents of memory to be left shifted by 0 to 16 bits before they are either fed to ALU or stored from ALU to memory. The CPU registers ACC & PREG1 can also be shifted using these shifters. In this case they require two cycles. A 5-bit register TREG1 specifies the number of bits by which the scaling shifter should shift either the incoming data to one of the CPU registers or vice versa. When the incoming data to CPU is left shifted by the scaling shifter the LSB's are filled with 0.

Auxiliary Register ALU (CARAU).

It consists of eight 16-bit auxiliary registers (ARS) ARO-ART, a 3-bit auxiliary register pointer (ARP) &

an unsigned 16-bit ALU. ARAU calculates indirect addresses by using inputs from ARs, 16-bit index register (INDX) & auxiliary register compare register (ARCR). The ARAU can autoindex the current AR while the data memory location is being addressed & can index either by ± 1 or by the contents of the INDX. As a result, accessing data does not require the CALU for address manipulation; therefore, the CALU is free for other operations in parallel. This makes the instructions to be executed faster compared to the conventional microprocessors. For example, let us consider the following sequence of 8085 instructions:-

MOV A, M₆₀

INX H

These instructions enable the accumulator to be loaded using indirect addressing mode & HL register used as the address pointer is incremented. These two instructions can be replaced by a single 5x instruction LACC*, 0.

Further, any one of the auxiliary registers can be used as the address pointer & incremented by the above instruction. The register that will be used is specified by the content of the ARP.

The auxiliary registers ARO-ART may also be used as the general purpose registers for holding the operands for arithmetic & logical operations in CALU. Some of the other registers of ARAU & their functions are as follows:-

Index Register (INDX) :-

The 16-bit INDX is used by the ARAU as a step value (addition or subtraction by more than 1) to modify the address in the AR's during indirect addressing. For example, when the ARAU steps across a row of a matrix, the indirect address is incremented by 1. However, when the ARAU steps down a column, the address is incremented by the dimension of the matrix. The ARAU can add or

subtract the value stored in the INDX from the current AR as part of the indirect address operation. INDX can also map the dimension of the address block used for bit-reversal addressing.

Auxiliary Register Compare Register (ARCR):

The 16-bit ARCR is used for address boundary comparison. The CMPP instruction compares the ARCR to the selected AR & places the result of the compare in the TC bit of ST1.

Block Move Address Register (BMAR):

The 16-bit BMAR holds an address value to be used with block moves & multiply accumulate operations. This register provides the 16-bit address for an indirect-addressed second operand.

Block Repeat Registers (RPTC, BRCR, PASR, PAER):

All these registers are 16-bit wide. Repeat counter register (RPTC) holds the repeat count in a repeat single-instruction operation & is loaded by the RPT & RPTZ instructions. Block Repeat counter register (BRCR) holds the count value for the block repeat feature. This value is loaded before a block repeat operation is initiated. Block repeat program address start register (PASR) indicates the 16-bit address where the repeated block of code starts. The block repeat program address end register (PAER) indicates the 16-bit address where the repeated block of code ends. The PASR & PAER are loaded by the RPTB instruction.

Parallel Logic Unit (PLU):

It performs Boolean operations or the bit manipulations required of high speed controllers. The PLU can set, clear, test or toggle bits in a status register control register or any data memory location. The PLU allows logic operations to be performed on data memory values directly without affecting the contents

of the ACC or PREG. Results of a PLU function are written back to the original data memory location.

Memory-Mapped Registers

The 'c5x has 96 registers mapped into page 0 of the data memory space. All 'c5x DSP's have 28 CPU registers & 16 Input/Output (I/O) port registers but have different numbers of peripheral & reserved registers. Since the memory-mapped registers are a component of the data memory space, they can be written to & read from in the same way as any other data memory location. The memory-mapped registers are used for indirect data address pointers, temporary storage, CPU status & control or integer arithmetic processing through the ARDV.

Program Controller

The program controller contains logic circuitry that decodes the instructions, manages the CPU pipeline, stores the status of CPU operations & decodes the conditional operations. Parallelism of architecture lets the 'c5x perform three concurrent memory operations in any given machine cycle: fetch an instruction, read an operand & write an operand. The program controller consists of the following elements:-

- 16-bit Program Counter (PC).
- 16-bit status registers ST0, ST1, Processor mode status register (PMST) & circular buffer control register (CBCR).
- (8x16)-bit hardware stack.
- Address generation logic.
- Instruction register.
- Interrupt flag register & Interrupt mask register.

Q.14. Write short note on Multirate DSP Processors.

→ There are several applications where the signal of a given sampling rate (i.e. sampling frequency) requires to be converted

into an equivalent signal with a different sampling rate. Multirate digital signal processing is required in digital systems when more than one sampling rate is required. For example, in digital audio, the different sampling rates used are 82 kHz for broadcasting, 44.1 kHz for compact disc & 48 kHz for audio tape. In digital video, the sampling rate for composite video signals are 14.3181818 MHz & 17.734475 MHz for NTSC & PAL respectively. But the sampling rates for digital component of video signals are 11.5 MHz & 8.75 MHz for luminance & colour difference signal. As a matter of fact, different sampling rates can be obtained using an up-sampler & down-sampler. The basic operations in multirate processing to achieve this are decimation & interpolation. Decimation is used for reducing the sampling rate & interpolation used for increasing the sampling rate. Also in digital transmission systems like teletype, facsimile, low bit rate speech where data has to be handled in different rates, multirate signal processing is used.

Application of MDSP:-

In fact, multirate digital signal processing finds its applications in :-

- (i) sub-band coding (i.e. speech, image)
- (ii) voice privacy using analog phone lines.
- (iii) signal compression by subsampling.
- (iv) A/D, D/A converters, etc.

There are various areas in which multirate signal processing is used. Few are :-

- (i) Speech & Audio processing systems.
- (ii) Antenna systems.
- (iii) communication systems.
- (iv) Radar systems.

Advantages of MDSP :-

The various advantages of multirate digital signal processing are as under:-

- (i) computational requirements are less.
- (ii) storage for filter coefficients are less.
- (iii) Finite arithmetic effects are less.
- (iv) filter order required in multirate application are low.
- (v) sensitivity to filter coefficient lengths are less.

At the time of designing multirate systems, effects of aliasing for decimation & pseudoimages for interpolators must avoided.

Q.15. Explain Decimation by Factor 'D'.

- Decimation is a process in which sampling rate is reduced. It is also called as down sampling.
- consider a discrete time signal $x(n)$.
- The sampling rate of $x(n)$ can be reduced by a factor 'D' by taking every D^{th} sample of signal. This can be explained with the help of block diagram & waveforms shown in fig. 1 & fig. 2.

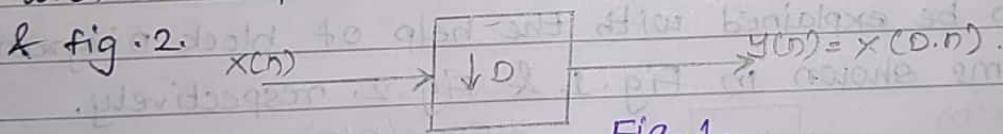


Fig. 1.

- The output $y(n)$ is called decimated signal or down sampled signal. The waveforms are shown in Fig. 2.
- Here, let $D=3$. Then as shown in the waveform (Fig. 2), decimated signal can be obtained by simply keeping every third sample in the output & removing in between.
 $D-1 = 2$ samples.
- That means ; 0th sample is available in output
1st sample is discarded
2nd sample is discarded
3rd sample is available in output

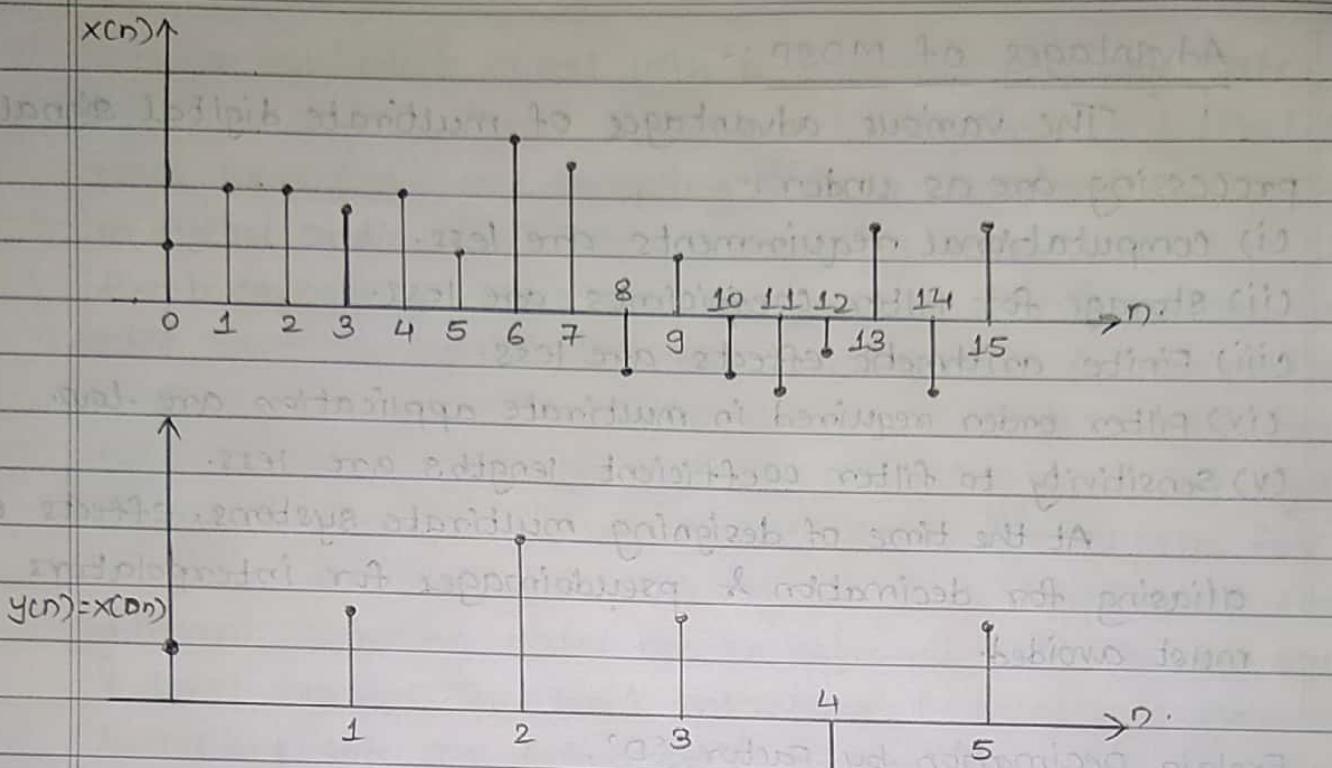


Fig. 2.

- Q.16. Explain Interpolation by factor 'I'.
- The sampling rate of a discrete time signal processing can be increased by factor I by placing $I-1$ equally spaced zeros between successive samples.
 - This can be explained with the help of block diagram & waveforms shown in Fig. 1. & Fig. 2. respectively.

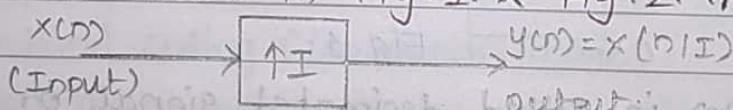


Fig. 1.

- This can be explained mathematically as,
$$y(n) = x(n/I) \text{ for } n = 0, \pm I, \pm 2I, \dots$$

$$= 0 \text{ otherwise} \quad \text{①}$$
- The Z transform of signal $y(n)$ is,
$$Y(z) = \sum_{n=-\infty}^{\infty} y(n) z^{-n} \quad \text{②}$$
- But $y(n) = x(n/I)$

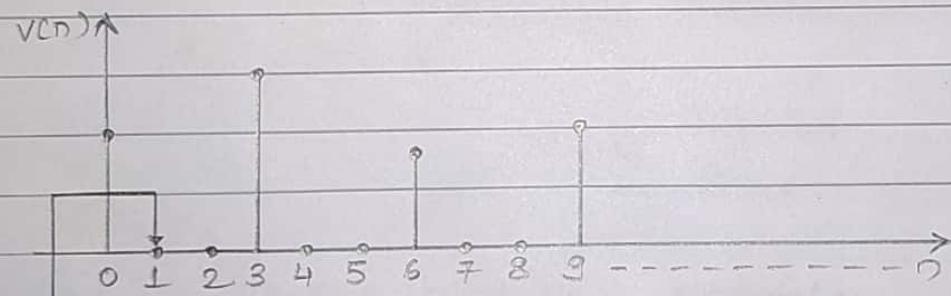
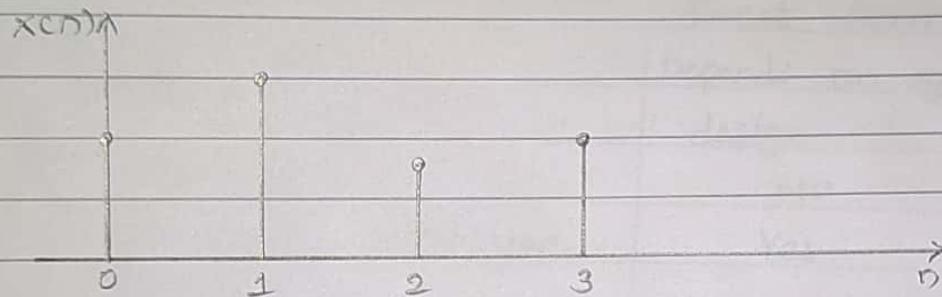
$$\therefore Y(z) = \sum_{n=-\infty}^{\infty} x(nI) z^{-n} \quad \dots \textcircled{3}$$

• Similar to the derivation of decimation we can write,

$$Y(z) = \sum_{n=-\infty}^{\infty} x(nI) \cdot z^{-I \cdot n}$$

$$= \sum_{n=-\infty}^{\infty} x(nI) \cdot (z^I)^{-n}$$

$$Y(z) = X(z^I). \quad \dots \textcircled{4}$$



Equally spaced
zeros

Fig. 2.