UNIT V -DSP PROCESSORS

Outcome

- Able to understand fixed point and floating point architecture
- Apply the Pipelining techniques

Introduction

- DSP processors can be divided into two broad categories:
  - » General Purpose
  - » Special Purpose
- Further DSP processors include
  - » Fixed point devices
  - » Floating point devices

Examples

- 32-BIT FLOATING POINT PROCESSORS
  - TI TMS320C3X, TMS320C67xx
  - AT&T DSP32C
  - ANALOG DEVICES ADSP21xxx
  - Hitachi SH-4
- 16-BIT FIXED POINT PROCESSORS
  - TI TMS320C2X, TMS320C62xx
  - Infineon TC1xxx (TriCore1)
  - MOTOROLA DSP568xx, MSC810x
  - ANALOG DEVICES ADSP21xx
  - Agere Systems DSP16xxx, Starpro2000
  - LSI Logic LSI140x (ZPS400)
  - Hitachi SH3-DSP
  - StarCore SC110, SC140

Fixed Vs Floating Point Processors

- fixed point processor are :
  - cheaper
  - smaller
  - less power consuming
  - Harder to program
    - – Watch for errors: truncation, overflow, rounding
  - Limited dynamic range
  - Used in 95% of consumer products
- floating point processors
  - have larger accuracy
  - are much easier to program
  - can access larger memory

Fixed Point Vs Floating Point

- Floating Point
- Applications

•Modems

•Digital Subscriber Line (DSL)

•Wireless Basestations

•Central Office Switches

•Private Branch Exchange (PBX)

•Digital Imaging

•3D Graphics

•Speech Recognition

•Voice over IP

•FixedPoint

Applications

•Portable Products

•2G, 2.5G and 3G Cell Phones

•Digital Audio Players

•Digital Still Cameras

•Electronic Books

•Voice Recognition

•GPS Receivers

•Headsets

•Biometrics

•Fingerprint Recognition

Features of DSP Processors

- DSP processors should have multiple registers so that data exchange from register to register to fast.
- It requires multiple operands simultaneously. Hence DSP processors should have multiple operand fetch capacity.
- DSP processors should have circular buffers to support circular shift operations.
- It should be able to perform multiply and accumulate operations very fast.
- It should have multiple pointers to support multiple operands, jumps and shifts.
- To support the DSP operations fast, the DSP processors should have on chip memory.
- For real time applications, interrupts and timers are required. Hence DSP processors should have powerful interrupt structure and timers.

PIPELINING

- A technique which allows two or more operations to overlap during execution.
- It is used extensively in DSP to increase speed.
- The simultaneous functions going on are:
- **Fetch:** In this phase, any instruction is fetched from the memory.
- **Decode:** In this phase, an instruction is decoded.
- **Read:** An operand required for the instruction is fetched from the data memory.
- **Execute:** The operation is executed and results are stored at appropriate place.

instruction execution without pipeline

Value of T	Fetch	Decode	Read	Execute
1	I 1			
2		I 1		
3			I 1	
4				I 1
5	I 2			
6		I 2		
7			I 2	
8				I 2

instruction execution with pipeline

Value of T	Fetch	Decode	Read	Execute
1	I 1			
2	I 2	I 1		
3	I 3	I 2	I 1	
4	I 4	I 3	I 2	I 1
5	I 5	I 4	I 3	I 2
6		I 5	I 4	I 3
7			I 5	I 4
8				I 5

Parameters used in pipelining

- Throughput is determined by the number of instructions through the pipe per unit time.

- In a perfect pipeline the average time per instruction is given by(Hennesy and Patterson,1990)

$$\text{time per instruction(non-pipeline)} \;=\; \frac{\text{time per instruction(non-pipeline)}}{\text{number of pipe stages}}$$

- Speedup $= \dfrac{\text{Average instruction time (nonpipeline)}}{\text{Average instruction time(pipeline)}}$

Summary

- Fixed Point AND Floating point architecture
- Pipelining