

# Real-Time Systems and Application-Hardware

Dr. J. K. Das

School of Electronics Engineering, KIIT DU

# Table of Contents

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

## 1 Processor Architecture

## 2 Instruction Processing

## 3 I/O & Interrupts

## 4 Memory

## 5 Architectural Advances

# Basic Processor Architecture

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

Processor Architecture

Instruction Processing

I/O & Interrupts

Memory

Architectural Advances

The basic processor architecture is of two types:

- Von Neumann Architecture [refer to Fig1]
- Harvard Architecture[refer to Fig2]

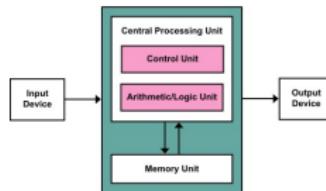


Figure: Von Neumann

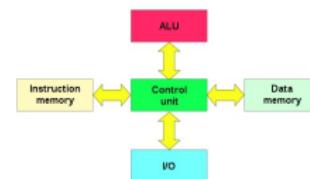


Figure: Harvard

# Basic Processor Architecture *contd...*

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

Processor Architecture

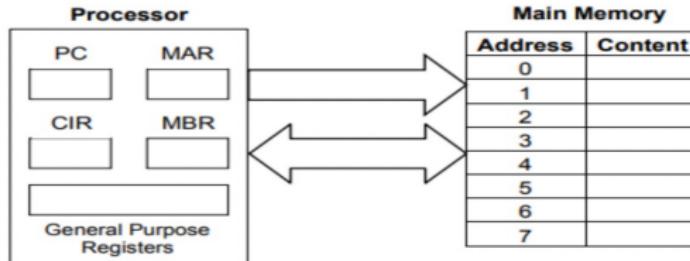
Instruction Processing

I/O & Interrupts

Memory

Architectural Advances

- ☞ The Von Neumann processor consists of 3 fundamental units: **CPU, Memory & System Bus.**
- ☞ The system bus consists of Address bus(unidirectional) and data bus(bidirectional).
- ☞ In these the I/O registers are memory mapped as they are accessed similar to accessing memory.
- ☞ Data transfer may be synchronous or asynchronous depending on instructions used. The internal architecture of CPU is shown below.



# Basic Processor Architecture *contd...*

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

The Harvard Architecture consists of 2 memory elements- one for data Memory and other one is program memory(instruction memory). Processor Architecture is same as that of Von Neumann architecture.

# Instruction Processing

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

Processor  
Architecture

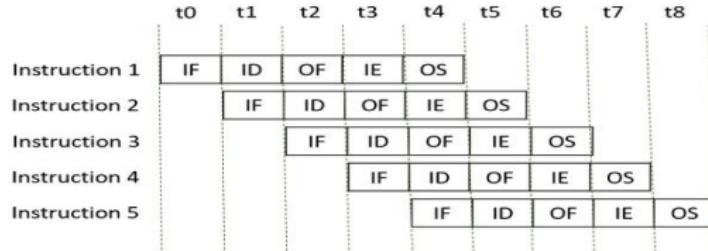
Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

The instruction processing consists of multiple phases taking multiple clock cycle is shown in figure below.



Instruction Pipelining of Five Instructions

IF-Instruction Fetch, ID-Instruction Decode, OF- Operand Fetch, IE- Instruction Execute, OS- Operand Store

# Instruction Processing<sub>contd...</sub>

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

Processor Architecture

Instruction Processing

I/O & Interrupts

Memory

Architectural Advances

Every instruction has its unique binary code stored in memory and a stream of such code forms a machine-language program. The general instruction format is shown below.

**opcode op1, op2, op3**

in the above code op1,op2,op3 are operands may be direct or memory referenced value. opcode represents the operation to be performed by the instruction. For example:

INC R1 ; increments the content of register R1  
ADD R1, R2; add content of R1, R2 and stores it in R1.

# Input/Output and Interrupts

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- Every computer system needs input and output ports to interface different devices in order to get excitation and give response.
- the Von Neumann architecture does not contain any I/O block, rather input and output registers are assumed to exists.
- with memory mapped I/O, I/O ports are considered to be a part of memory and to access them memory operands are used.
- In I/O mapped I/O or programmed I/O, separate address space is provided for I/O and memory.

# Input/Output and Interrupts

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- Every computer system needs input and output ports to interface different devices in order to get excitation and give response.
- the Von Neumann architecture does not contain any I/O block, rather input and output registers are assumed to exists.
- with memory mapped I/O, I/O ports are considered to be a part of memory and to access them memory operands are used.
- In I/O mapped I/O or programmed I/O, separate address space is provided for I/O and memory.

# Input/Output and Interrupts<sub>contd...</sub>

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

## I/O Types used in Real-Time Systems

- Parallel I/O Devices, PCI(a synchronous Parallel) protocol
- Serial communication- UART, USART (can work in half-duplex or full duplex mode)
- other serial communication protocols - I2C, SPI, USB, CAN
- Wireless protocols- IrDA, Bluetooth, Zigbee (IEEE 802.15.4 standard), Wireless LAN(IEEE 802.11) etc.

# Input/Output and Interrupts<sub>contd...</sub>

## Interrupts

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- An interrupt is an external hardware signal that initiates an event and are used to indicate an I/O operation completed or to initiate an I/O operation.
- Hardware Interrupts are used to give prompt service to important events occurring in the Operating environment.
- The interrupts used for different events to occur or completed may be maskable or non-maskable.
- Generally nonmaskable interrupts are reserved critical condition of operating conditions.

# Input/Output and Interrupts<sub>contd...</sub>

## Interrupt Process

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

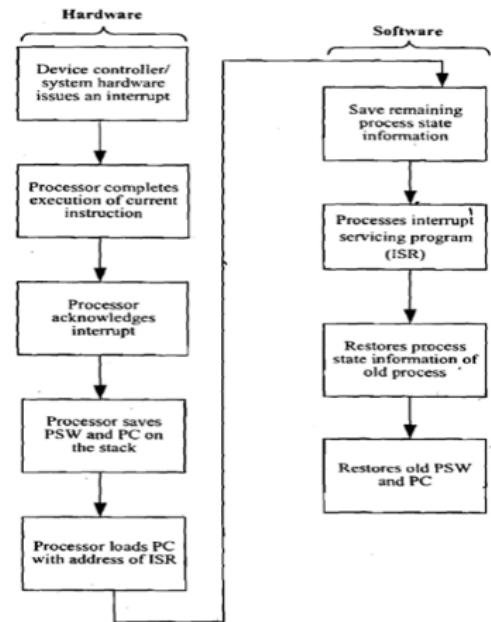
I/O & Interrupts

Memory

Architectural  
Advances

A typical Interrupt service process as follows:

- ☞ Interrupt process is activated on request and latched to CPU hardware.
- ☞ to start, PC is moved to stack after current execution of instruction completed and PC is loaded with a interrupt handler address.
- ☞ After completing interrupt service routine, PC value is popped (restored) from stack.



# Input/Output and Interrupts<sub>contd...</sub>

## Interrupt Sources or Types

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

The interrupt sources may be of 2 types which are

- **Hardware Sources:** this may be internal or external for interrupt of ongoing routines and makes a transit from current execution to new ISR. It is different from processor to processor.
- **Software Sources:** interrupts related to processor, detecting computational errors, or may be due to SWI call.

# Input/Output and Interrupts<sub>contd...</sub>

## Hardware Interrupt Example

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

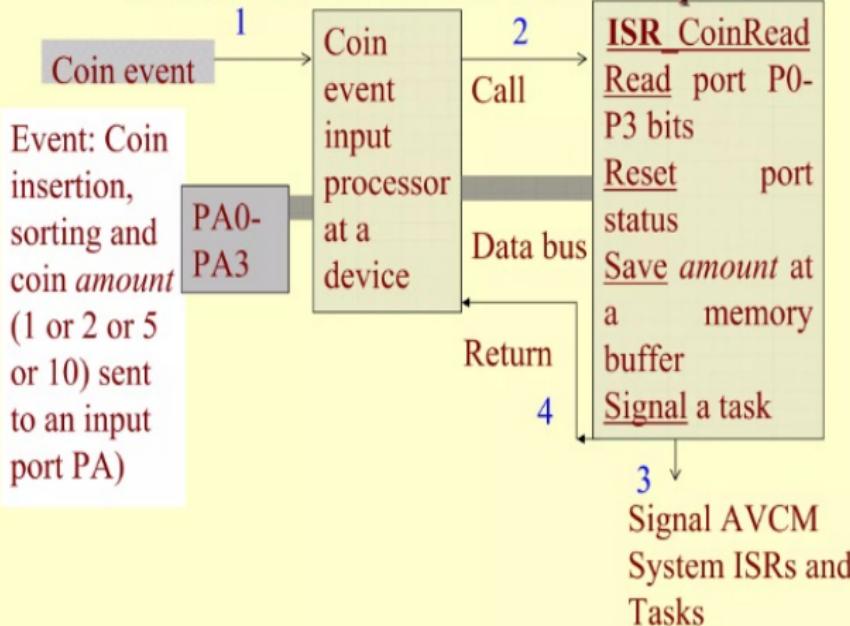
Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

### Use of ISR in the ACVM example



# Memory Technology

## Memory types

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

The memory at the basic level can be classified as

- Processor Memory (Register Array)
- Internal on-chip Memory
- Primary Memory
- Cache Memory
- Secondary Memory

# Memory Technology

## Data Storage

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

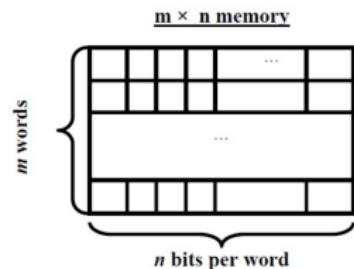
I/O & Interrupts

Memory

Architectural  
Advances

An  $m$  word memory can store  $mxn$  :  $m$  words of  $n$  bits each. One word is located at one address therefore to address  $m$  words we need  $k = \log_2(m)$  address input signals or  $k$  number of address lines can address  $m = 2^k$  words

Example 4,096x8 memory will have 32,768 bits, 12 address input signals and 8 input/output data signals



# Memory Technology

## Common Memory Types

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- Read Only Memory (ROM)- Mask-programmed ROM, OTP ROM: One-time programmable ROM, EPROM: Erasable programmable ROM, EEPROM
- Flash Memory
- RAM: “Random-access” memory
  - SRAM: Static RAM -Memory cell uses flip-flop to store bit, Requires 6 transistors, Holds data as long as power supplied
  - DRAM: Dynamic RAM - Memory cell uses MOS transistor and capacitor to store bit , More compact than SRAM , periodic refresh required, slower to access than SRAM

# Memory Technology

## Memory size

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

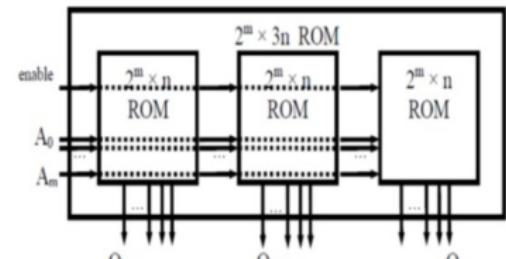
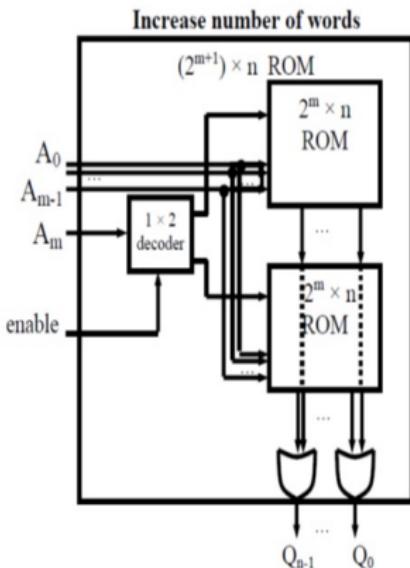
Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

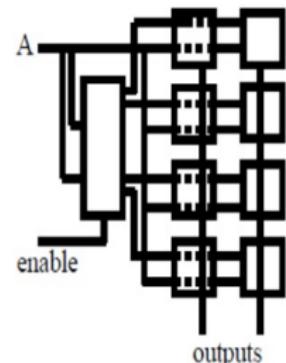
Memory

Architectural  
Advances



INCREASING WORD SIZE

Increase number  
and width of  
words



# Memory Technology

## Memory Hierarchy

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

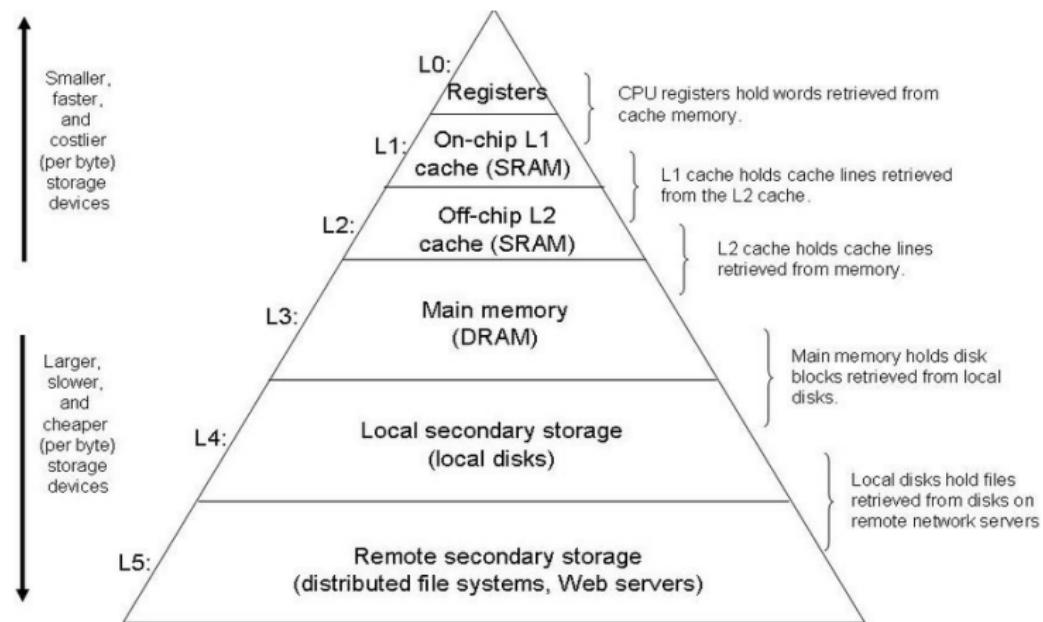
Processor Architecture

Instruction Processing

I/O & Interrupts

Memory

Architectural Advances



# Memory Technology

## Embedded vs External Memory

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

Features	Embedded Memory	External Memory
Type	1. Embedded RAM, ROM, Flash ROM 2. Smaler size	External RAM, ROM, Flash ROM Large size
Features	1. Multi port Features 2. Internal bus used to access 3.May be synchronous or asynchronous access 4.Faster to access	1. Multi port Features 2. External Bus used to access 3.May be synchronous or asynchronous access 4.slower to access

# Memory Technology

## Memory Interface

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

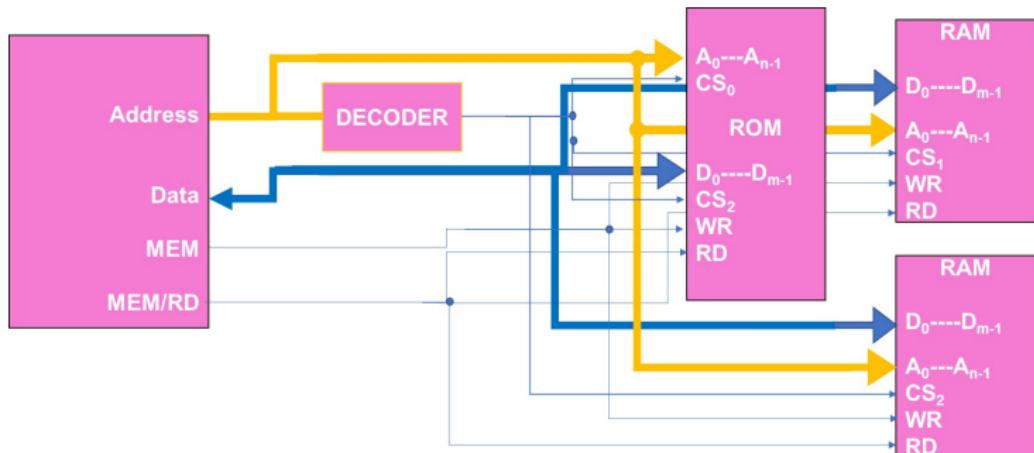
Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances



# Architectural Advances

## Introduction

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- In last few years, CPU architecture have been changed remarkable with advanced features.
- Most of the advanced done on locality based references for high performance and throughput.
- With the advancement of automation and integration technologies, processor gets more functional and performance using parallelism in a single chip.

# Architectural Advances

## Introduction1

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- With the help pipeline architecture, processing speed increases from previous cyclic base instruction processing.
- Also modern architecture uses 2 separate buses(instruction and data) i.e. uses 2 memory(instruction and data memory).
- Also in these CPUs, the bus width may vary with processor design.

# Architectural Advances

## Pipeline Instruction Processing

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

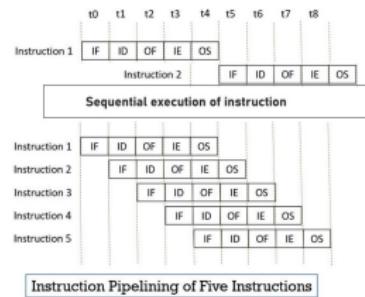
Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- Pipeline processing provides execution of instructions in parallel in different phases which increase instruction throughput as discussed in early.
- In non-pipelined execution of instruction processing done in cyclic basis so power consumption increase as well as throughput decreases. Example take a 5stage pipeline as shown in figure.



# Pipeline Instruction Processing

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

Processor Architecture

Instruction Processing

I/O & Interrupts

Memory

Architectural Advances

- The pipeline architecture requires additional buffer register between different stages.
- Pipeline also degrades performances at certain situations as in case of branch instruction, the succeeding instruction is may not valid in pipeline. This can be avoided by using advanced branch prediction technique.
- Cycle stilling may occur.
- Higher-level pipelines or superscalar architectures, the instruction is decomposed further. With superscalar architecture the clock speed may reach to GHz-level.

# Architectural Advances

## Design Space of Processor

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

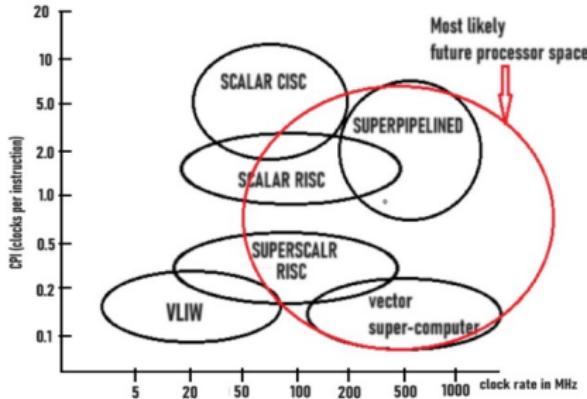
Memory

Architectural  
Advances

V

### Design Space of Processor:

- Various processor families can be mapped onto a coordinated space of clock rate versus CPI.
- As implementation technology evolves rapidly, the clock rate are moving from low to higher speeds.
- Another trend is that processor designers are trying to lower the CPI rate using hardware and software approaches.



# Architectural Advances

## Basic Scalar computer Architecture

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

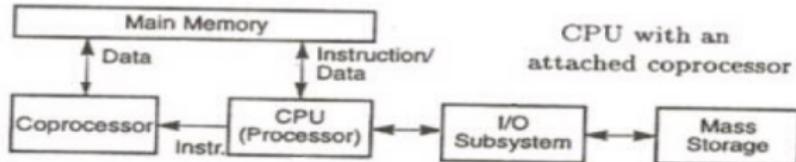
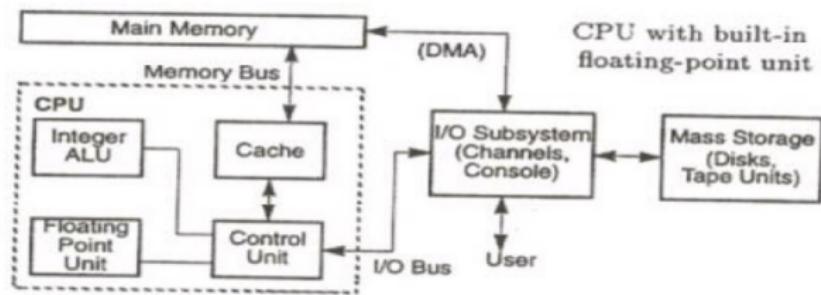
I/O & Interrupts

Memory

Architectural  
Advances

### Base Scalar Computer Architecture:

- The CPU is essentially a scalar processor consists of multiple functional units.
- The floating-point unit can be built on a coprocessor attached to the CPU.



# Architectural Advances

## Basic Scalar computer Architecture

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

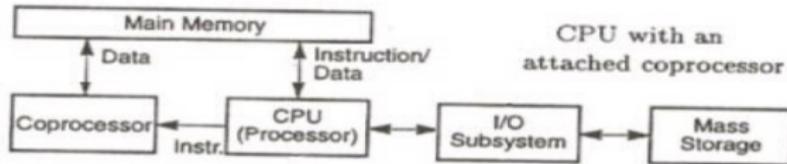
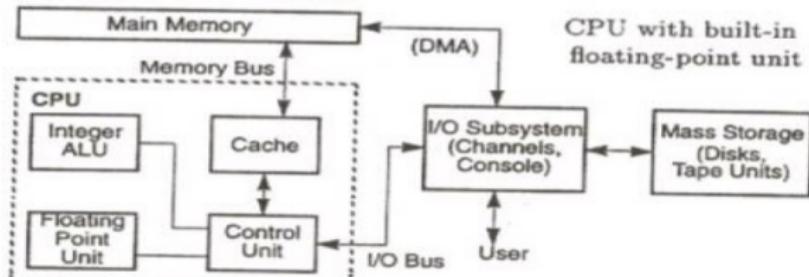
I/O & Interrupts

Memory

Architectural  
Advances

### Base Scalar Computer Architecture:

- The CPU is essentially a scalar processor consists of multiple functional units.
- The floating-point unit can be built on a coprocessor attached to the CPU.



# Architectural Advances

## RISC Scalar Processor

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

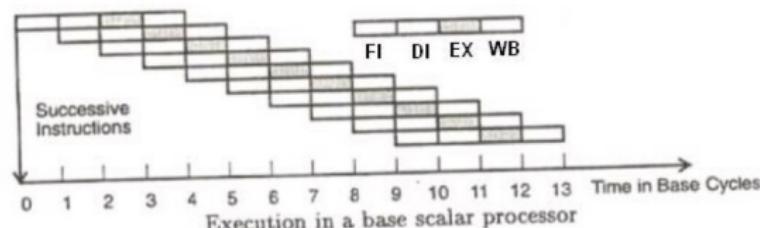
I/O & Interrupts

Memory

Architectural  
Advances

### RISC Scalar Processors:

- Generic RISC processors are called scalar RISC because they are designed to issue one instruction per cycle, similar to the base scalar processor.



- In theory, both RISC and CISC scalar processors should perform about the same if they run with the same clock rate, and with equal program length.
- Without a high clock rate, a low CPI, and good compilation support, neither CISC nor RISC can perform well as designed.

# Architectural Advances

## Super-scalar Architecture

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

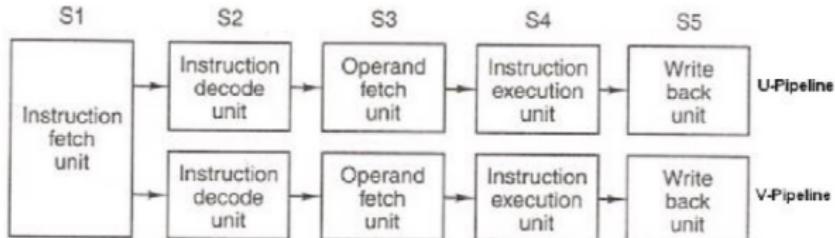
Architectural  
Advances

### Superscalar Architectures:

- If one pipeline is good, then two pipelines are better.

**Example:** Consider the following architecture (as in Pentium);

- Single instruction fetch unit fetches pairs of instructions together and puts each one into its own pipeline, complete with its own ALU for parallel operation.
- The main pipeline (U-Pipeline) could execute an arbitrary Pentium instruction.
- The V-Pipeline could execute only simple integer instructions (and also one simple floating-point instruction).
- If the instructions in a pair were not simple enough or incompatible, only the first one was executed (in U-pipeline). The second one was then held and paired with the instruction following it.



Dual five-stage pipelines with a common instruction fetch unit.

# Architectural Advances

## An example of Super-scalar Processor

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

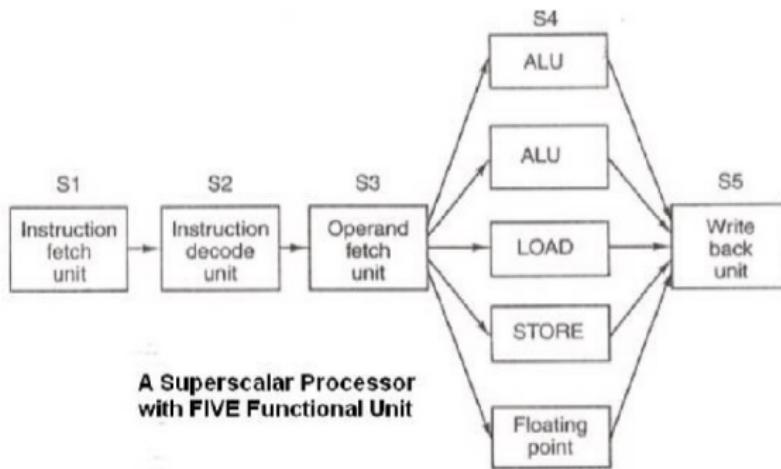
I/O & Interrupts

Memory

Architectural  
Advances

**Example:** A superscalar processor (Pentium II) with 5 functional units:

- It is possible to have multiple ALUs in stage (S4).
- Most of the functional units in stage (S4) take longer than one clock cycle to execute.
- Stage (S3) can issue instructions faster than the S4 stage.



## Architectural Advances

## Super-scalar Processor

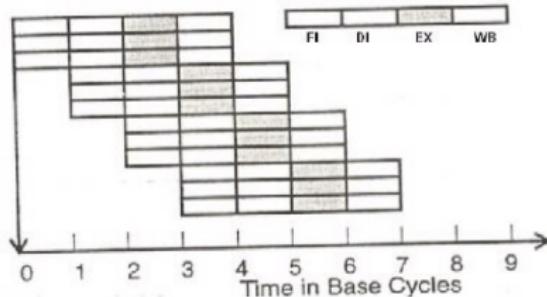
# Real-Time Systems and Application- Hardware

Dr. J. K. Das

## Architectural Advances

## Superscalar Processors:

- In a superscalar processor, multiple instruction pipelines are required. This implies that multiple instructions are issued per cycle and multiple results are generated per cycle.
  - Superscalar processors are designed to exploit more instruction-level parallelism in user programs. Only independent instructions can be executed in parallel without causing a wait state.
  - The instruction-issue degree in a superscalar processor is limited to 2-5 in practice.
  - The effective CPI of a superscalar processor should be lower than that of a generic scalar RISC processors.



A superscalar processor of degree  $m = 3$ .

# Architectural Advances

## A Typical Super-scalar Architecture

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

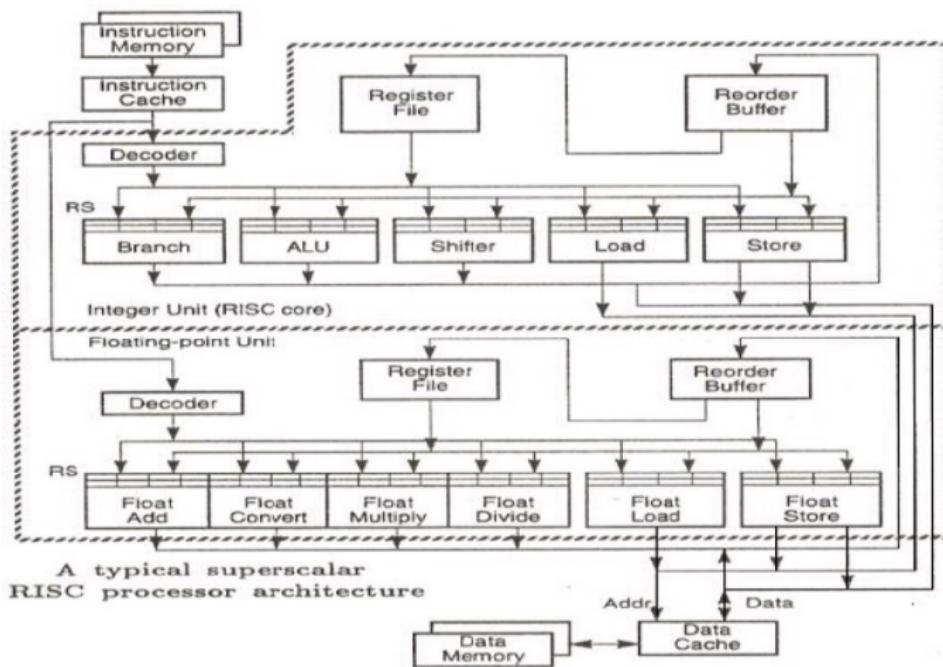
Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances



# Architectural Advances

## Example IBM RS/6000

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

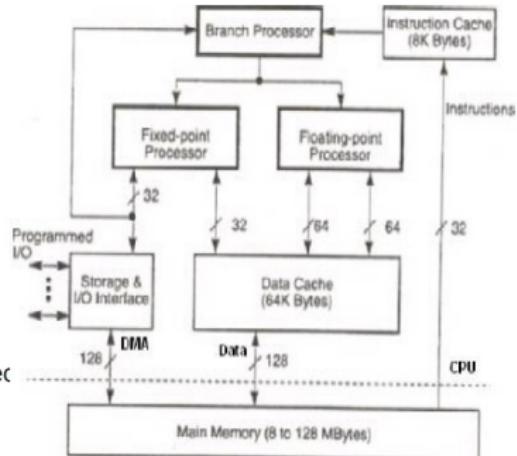
I/O & Interrupts

Memory

Architectural  
Advances

### Example: IBM RS/6000

- IBM announced this superscalar RISC system in 1990.
- There are 3 parallel functional units; branch processor, fixed-point unit, and floating-point unit.
- The branch processor can arrange the execution of up to 5 IPC.
- It is hardwired rather than micro-programmed control unit.
- The system uses a number of wide buses. These will provide the high instruction and data bandwidths required for superscalar implementation.
- This system design is optimized to perform well in numerically intensive scientific and engineering applications.



The POWER architecture of the IBM RISC System/6000 superscalar processor.

# Architectural Advances

## VLIW Architecture

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

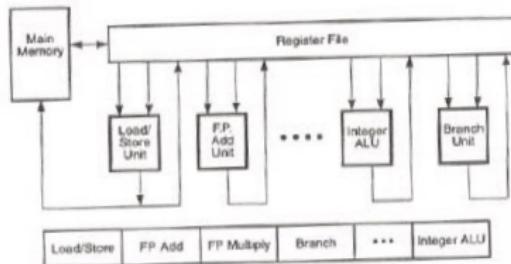
I/O & Interrupts

Memory

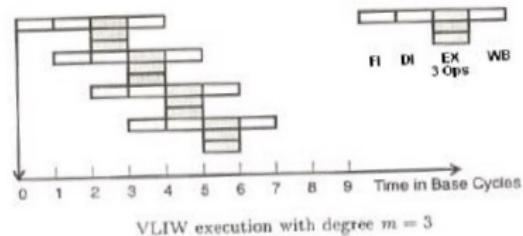
Architectural  
Advances

### The VLIW Architecture:

- Very Long Instruction Word (VLIW) architecture is generalized from two concepts; horizontal microcoding and superscalar processing.
- A typical VLIW machine has;
  - instruction words hundreds of bits in length.
  - Multiple functional units.
  - Common large register file shared by all functional units.



A typical VLIW processor and instruction format



VLIW execution with degree  $m = 3$

# Architectural Advances

## Super-scalar and VLIW Processor

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

### VLIW and Superscalar Processor:

VLIW machines behave much like superscalar machine with 3 differences:

1. The decoding of VLIW instruction is easier than that of superscalar instructions.
2. The code density of the superscalar machine is better than when the available instruction level parallelism is less than that exploitable by the VLIW machine.
3. A superscalar machine can be object-code compatible with a larger family of nonparallel machines. On the contrary, a VLIW machine exploiting different amount of parallelism would require different instruction sets.

### VLIW: Advantages:

- The main advantage of VLIW architecture is its simplicity in hardware structure and instruction set.
- The VLIW processor can potentially perform well in scientific applications where the program behavior (branch predictions) is more predictable.

# Architectural Advances

## Vector Processor

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

### Vector Processors:

- A vector processor is a coprocessor specially designed to perform vector computations.
- A vector instruction involves a large array of operands. The same operation will be performed over a string of data.
- Vector processors are often used in a multipip pipelined supercomputer.
- Vector processors can assume either;
- A register-to-register architecture using shorter instructions and vector register files, or
- A memory-to-memory architecture using memory-based instructions.
- The vector pipelines can be attached to any scalar processor (whether it is superscalar, superpipelined, or both).

# Architectural Advances

## Vector Pipeline

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

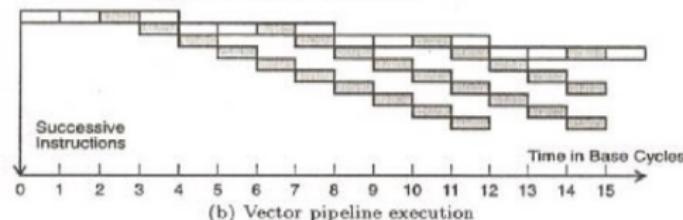
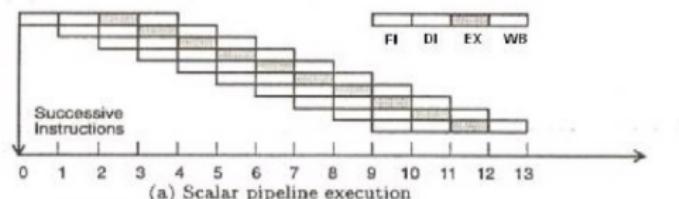
I/O & Interrupts

Memory

Architectural  
Advances

### Vector Pipelines:

- In a scalar processor, each scalar instruction executes only one operation over one data element.
- Each vector instruction executes a string of operations, one for each element in the vector.



Pipelined execution in a base scalar processor  
and in a vector processor

# Architectural Advances

## Array Processor

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

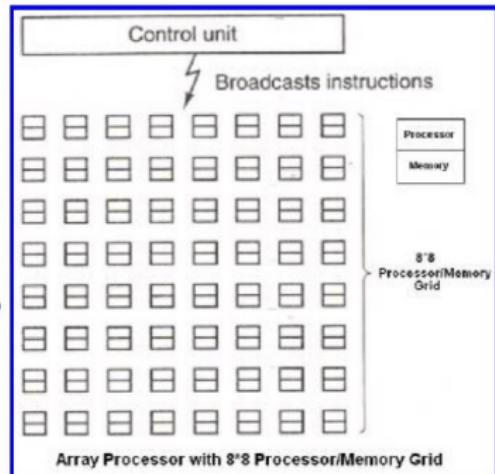
Architectural  
Advances

### Array Processor:

- An array processor consists of a large number of identical processors that perform the same sequence of instructions on different sets of data.

### Example: ILLIAC IV

- Announced by University of Illinois in 1972.
- The original plan was to build a machine consisting of 4 quadrants, each having  $8 \times 8$  square grid of processor/memory elements.
- Only one quadrant was built due to cost. It did achieve a performance of 50 megaflops.



# Peripheral Interfaces

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

Peripherals, sensor and actuators interfaced with Real-Time system are generally slower in nature than memory interface. The peripheral interface can be done in three ways which are

- Polled I/O or programmed I/O
- Interrupt driven I/O
- DMA base I/O interface

# Peripheral Interfaces

## Polled I/O or Programmed I/O

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

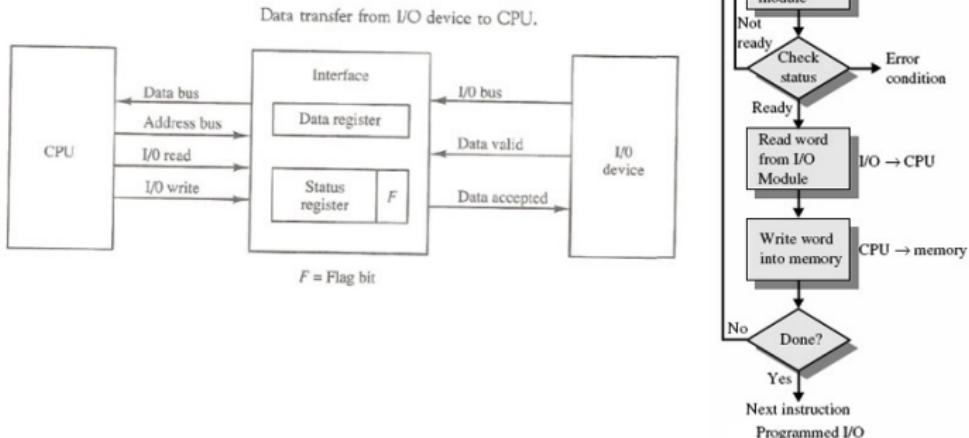
Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

The figure shows Polled or Programmed I/O interface.



# Peripheral Interfaces

## Polled I/O or Programmed I/O-description

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- Polled I/O system checks the status and data register periodically.
- it is a software control for I/O interface.
- It is simple way for I/O interfacing but suffers from unnecessary status requests.
- In minority of status request, leads to either Input or output translations with data register happens.
- this increases worst case latency.
- hence an appropriate polling intervals may increase CPU utilization with reduced I/O latency.
- in some case special PIU (peripheral Interface units) are used.

# Peripheral Interfaces

## I/O interface methods-difference

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

Processor Architecture

Instruction Processing

I/O & Interrupts

Memory

Architectural Advances

I/O-mapped - I/O	Memory-mapped I/O
Different address spaces are used for computer memory and I/O devices. I/O devices have dedicated address space.	Same address space is used for memory and I/O devices.
Separate control unit and control instructions are used in case of I/O devices.	Control units and instructions are same for memory and I/O devices.
More complex and costlier than memory-mapped I/O as more bus are used.	Easier to build and cheap as it's less complex.
Entire address space can be used by memory as I/O devices have separate address space.	Some part of the address space of computer memory is consumed by I/O devices as address space is shared.
Computer memory and I/O devices use different control instructions for read write.	Computer memory and I/O devices can both use same set of read and write instructions.
Separate control bus is used for computer memory and I/O devices. Though same address and data bus are used.	Address, data and control bus are same for memory and I/O devices.

# Peripheral Interfaces

## I/O interface methods-Interrupt Driven

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

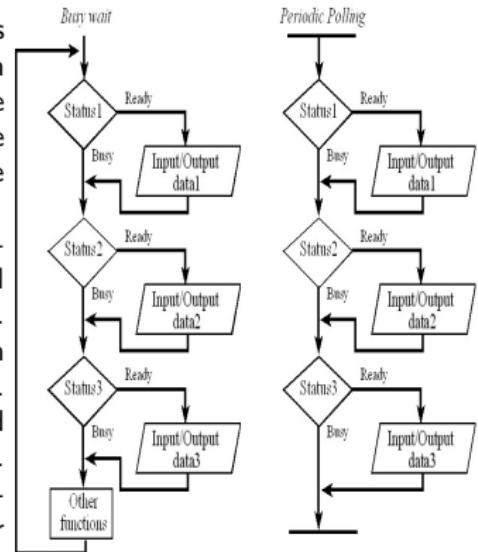
I/O & Interrupts

Memory

Architectural  
Advances

Interrupt driven I/O interface reduces service latency and uncertainty in loaded condition. At any moment one interrupt service will be active. There may exit multiple interrupts which are identified.

In small real-time system, for multiple interrupts, interrupts are identified by polling the status registers all PIUs. The status register contains flags which are set when any interrupt is raised. By this polling, higher priority tasked are first served than low-priority ones. Polling orders can be modified dynamically to provide rotating priorities for instances.



# Peripheral Interfaces

## I/O interface methods-Interrupt Driven

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

Processor Architecture

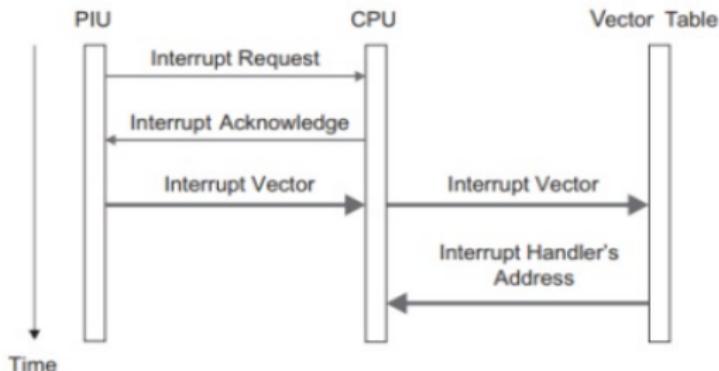
Instruction Processing

I/O & Interrupts

Memory

Architectural Advances

In large interrupt is to be serviced, it is not suitable polling methods to serve interrupt services. Instead, vectored interrupt handling methods is used for identification interrupt using system software. It is shown in figure.



# Peripheral Interfaces

## I/O interface methods-Interrupt Driven

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- Vectored interrupt increases system cost.
- In vectored interrupt method, if no of vectors is 256, then there exists 256 distinguished interrupt sources. When a single vector interrupt is inverted, can lead to system crashes.
- A priority interrupt controller(PIC) is used for prioritizing different interrupts when vectored interrupt is used shown in figure in next slide.
- Although Interrupt-driven I/O is effective technique to handle firm/hard real-time systems but privileged to time-critical system.

# Peripheral Interfaces

## I/O interface methods-Interrupt Driven

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

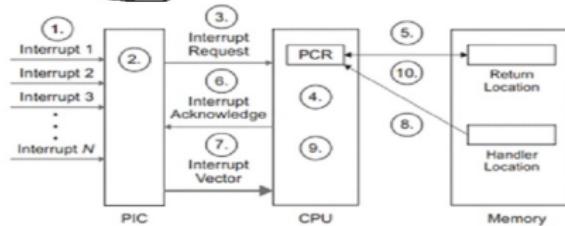
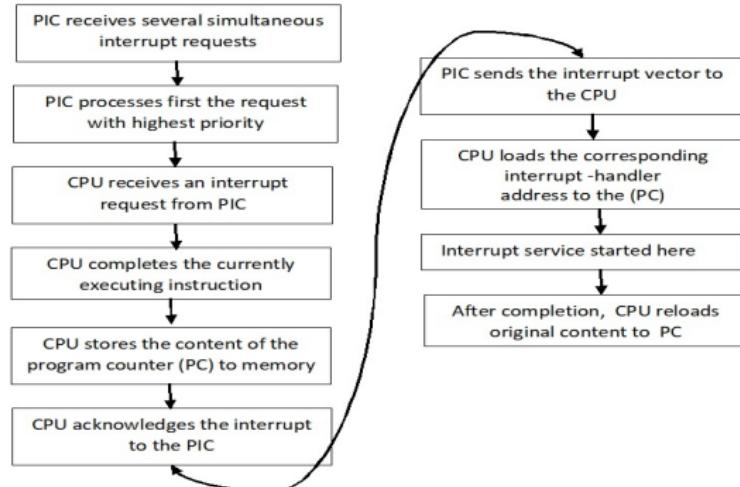
Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances



# Peripheral Interfaces

## I/O interface methods-DMA transfer

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- Interrupt driven I/O transfer is effective when data elements are small but large data it is very difficult to handle with this technique, also time consuming.
- To reduce time, DMA (direct memory access) transfer is used.
- In DMA, access to the computer's memory is given to other devices without intervention of CPU.
- Here a separate DMA controller is needed if DMA is not available inside CPU.

# I/O interface methods-DMA transfer

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

- An I/O device requests DMA transfer by activating a DMA - request signal ( $D\_REQ$ ).
- This makes the DMA controller issue a bus - request signal ( $B\_REQ$ ) for the CPU.
- The CPU finishes its present bus cycle and activates a bus - acknowledgment signal ( $B\_ACK$ ).
- After recognizing the active  $B\_ACK$  signal, the DMA controller activates a DMA - acknowledgment signal ( $D\_ACK$ ), instructing the I/O device to begin data transfer.
- When the transfer is completed, the DMA controller deactivates the  $B\_REQ$  signal, giving buses back to the CPU.

# Interrupt Sources

Real-Time Systems  
and Application-  
Hardware

Dr. J. K. Das

Processor  
Architecture

Instruction  
Processing

I/O & Interrupts

Memory

Architectural  
Advances

There are two types of interrupt sources-  
**Software, Hardware.**

- Software sources for interrupt are related to software detecting computational error or exceptional condition during execution and there up on executing a SWI (software interrupt) instruction, which causes processor interrupt of ongoing routine.
- Hardware sources can be internal or external for interrupt of ongoing routine and thereby diversion to corresponding ISR.
- The internal sources from devices differ in different processor or microcontroller or device and their versions and families.
- External sources and ports also differ in different processors or microcontrollers

# Microprocessor vs Micro-controllers

Real-Time Systems  
and Application-Hardware

Dr. J. K. Das

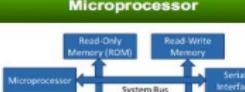
Processor Architecture

Instruction Processing

I/O & Interrupts

Memory

Architectural Advances

Microprocessor	Micro Controller
	
Microprocessor is heart of Computer system.	Micro Controller is a heart of embedded system.
It is just a processor, Memory and I/O components have to be connected externally	Micro controller has external processor along with internal memory and i/O components
Since memory and I/O has to be connected externally, the circuit becomes large.	Since memory and I/O are present internally, the circuit is small.
Cannot be used in compact systems and hence inefficient	Can be used in compact systems and hence it is an efficient technique
Cost of the entire system increases	Cost of the entire system is low
Due to external components, the entire power consumption is high. Hence it is not suitable to used with devices running on stored power like batteries.	Since external components are low, total power consumption is less and can be used with devices running on stored power like batteries.
Most of the microprocessors do not have power saving features.	Most of the micro controllers have power saving modes like idle mode and power saving mode. This helps to reduce power consumption even further.
Since memory and I/O components are all external, each instruction will need external operation, hence it is relatively slower.	Since components are internal, most of the operations are internal instruction, hence speed is fast.
Microprocessor have less number of registers, hence more operations are memory based.	Micro controller have more number of registers, hence the programs are easier to write.
Microprocessors are based on von Neumann model/architecture where program and data are stored in same memory module	Micro controllers are based on Harvard architecture where program memory and Data memory are separate
Mainly used in personal computers	Used mainly in washing machine, MP3 players