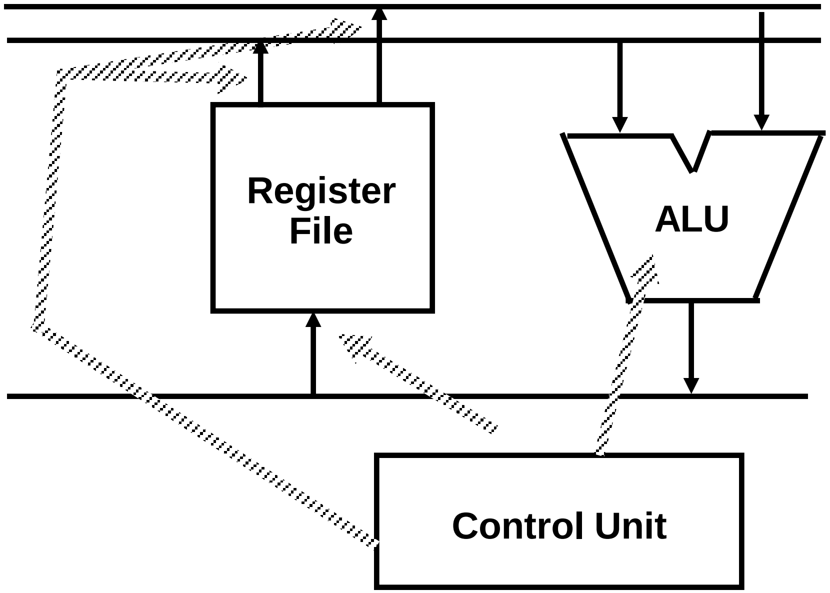
Unit-1

Processor Basics

CPU Organization

- The part of the computer that performs the bulk of data processing task is called Central Processing Unit or it is also referred to as CPU.

- The CPU is made up of 3 major parts:
    - Register Set
    - ALU
    - Control Unit


Register File
ALU
Control Unit

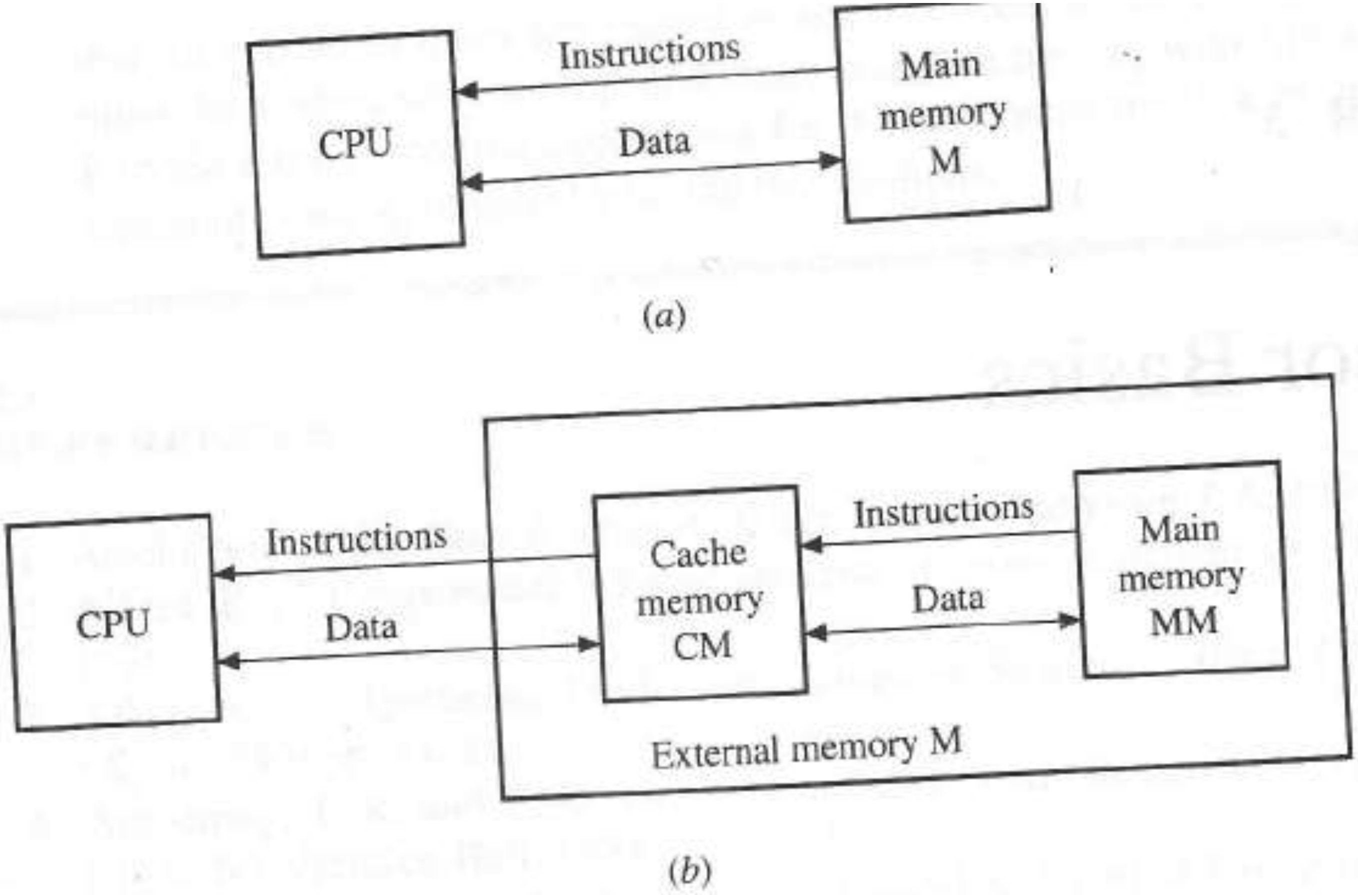
- Register set is used to store the intermediate data during the execution of the instruction.
- ALU is used to perform micro-operation that are required to execute the instruction.
- Control Unit supervises the transfer of data among the register as well as it instructs the ALU as to which operation to perform.

How CPU works

- First it fetches the instruction.
- After fetching the instruction, it decodes the instruction.
- After decoding the instruction, the CPU comes to know what operation is to be performed and it also comes to know that whether the data is stored in the memory or in the register.

Transfer of Data between Main Memory & CPU

- Without Cache
- With Cache

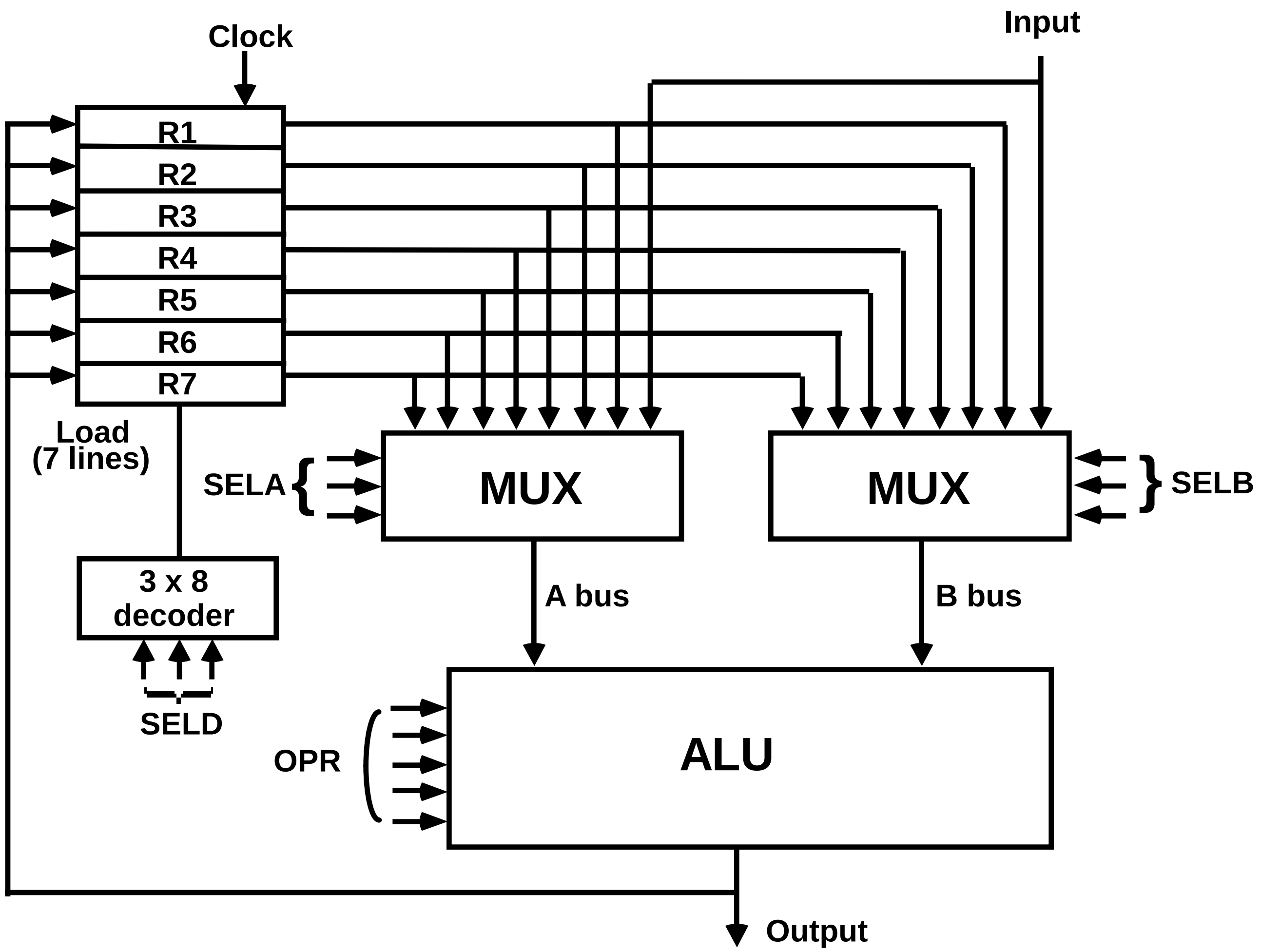

CPU
Instructions
Data
Main
memory
M
CPU
Instructions
Data
Cache
memory
CM
Instructions
Data
Main
memory
MM
External memory M

(a)

(b)

While executing an instruction if we are fetching the data from the memory then it is a time consuming process. It is more convenient & more efficient to store these intermediate value to registers and when large no of registers are used in the CPU, it is most efficient to connect them through a common BUS system.

Register Set With Common ALU



OPR

Select	Operation	Symbol
00000	Transfer A	TSFA
00001	Increment A	INCA
00010	ADD A + B	ADD
00101	Subtract A - B	SUB
00110	Decrement A	DECA
01000	AND A and B	AND
01010	OR A and B	OR
01100	XOR A and B	XOR
01110	Complement A	COMA
10000	Shift right A	SHRA
11000	Shift left A	SHLA

- A useful feature that is included in the CPU of most computers is a stack.

- A stack is a storage device that stores the information in such a manner that the item stored last is the first item retrieved.

- The two operations of a stack are insertion & deletion.

- For Push

```
SP ← SP+1
M[SP] ← DR
If (SP=0) then (FULL ← 1)
EMTY ← 0
```

- For POP

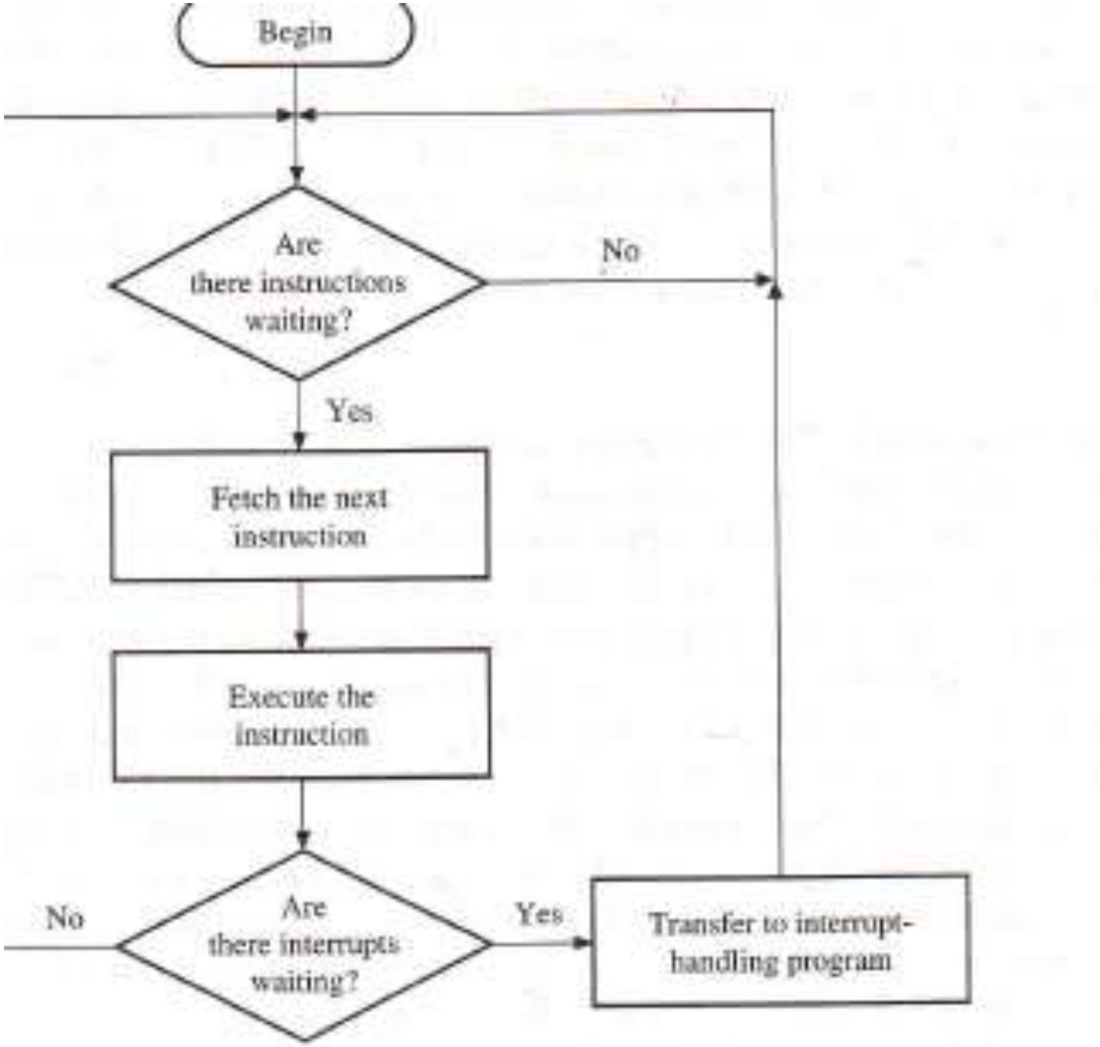
```
DR ← M[SP]
SP ← SP-1
If (SP=0) then (EMTY ← 1)
FULL ← 0
```

Some Important Points Related To CPU Are:

- The CPU communicates with I/O devices in the same ways as it communicates with the external memory. The I/O devices are associated with the addressable register called I/O ports. There are two types of I/O configuration:
  - Isolated I/O
  - Memory Mapped I/O

- The programs executed by general purpose computer fall into 2 broad categories:
  - User Program
  - Supervisor Program

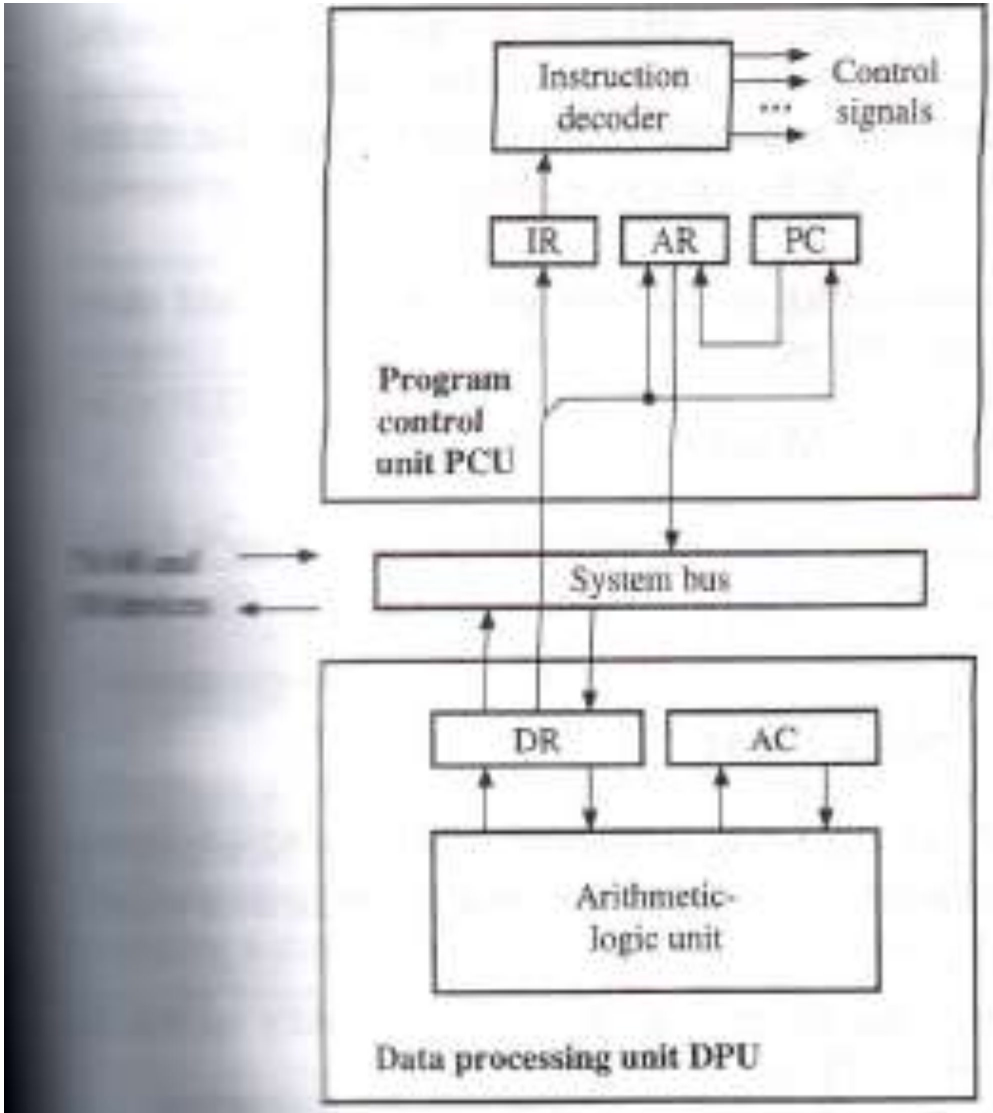
Sequence of The Instruction Executed


Begin
Are there instructions waiting?
No
Yes
Fetch the next instruction
Execute the instruction
No
Are there interrupts waiting?
Yes
Transfer to interrupt-handling program

Accumulator Based CPU

- A Small Accumulator Based CPU consist of:
  - Program Control Unit-IR, AR, PC, Decoder
  - Data Processing Unit- DR, AC, ALU

Accumulator Based CPU Contd…



Legend

Program control unit PCU

AR:	Address register
IR:	Instruction register
PC:	Program counter

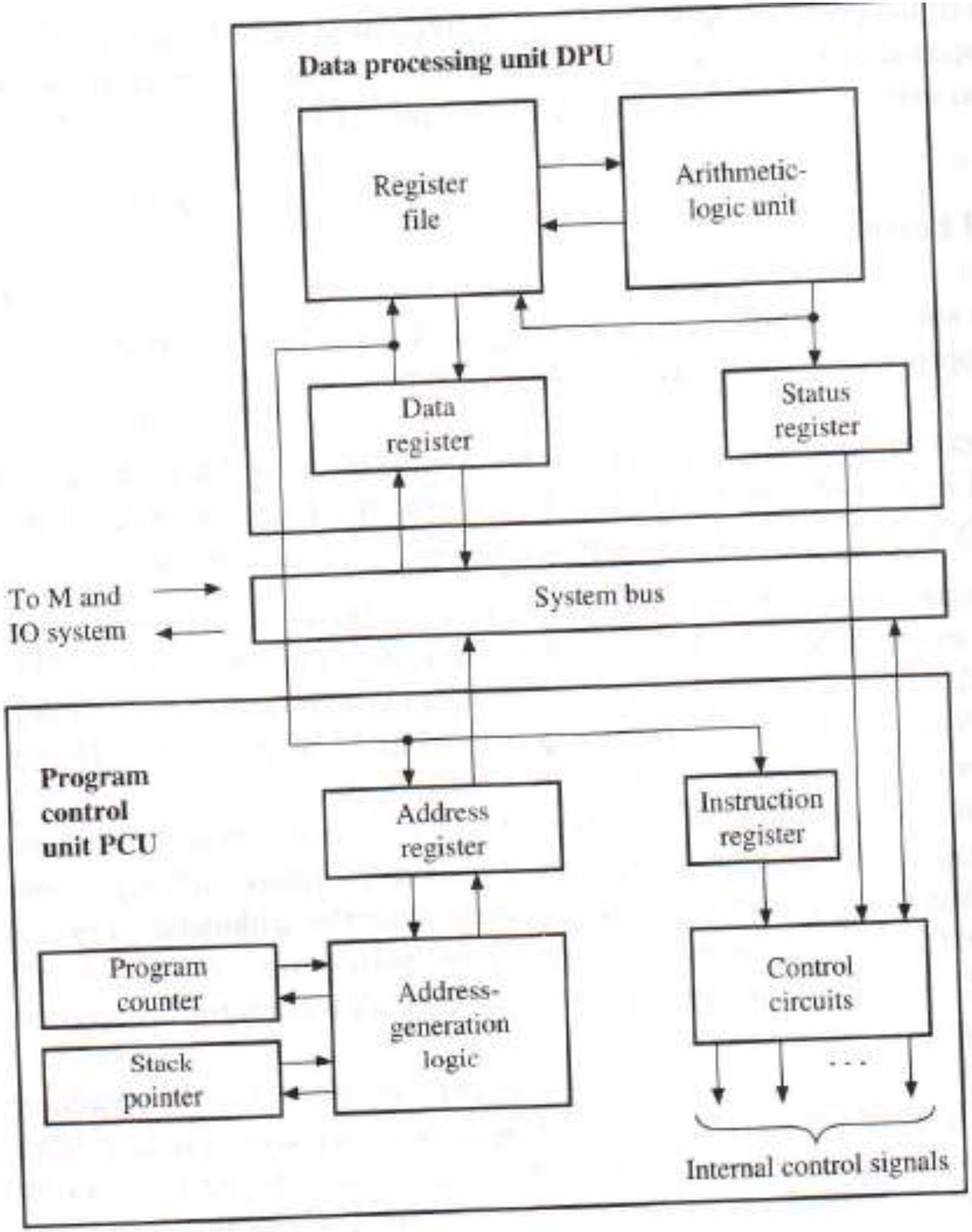
Data processing unit DPU

AC:	Accumulator register
DR:	Data register

Additional Features

There are many ways in which the basic design of Small Accumulator Based CPU can be improved. Most recent CPU contain the following extension.

- Multipurpose Register set for storing Data & Addresses
- Additional Data, Instruction & Address Type
- Register To indicate computation status
- Program Control Stack


Data processing unit DPU
Register file
Arithmetic-logic unit
Data register
Status register
To M and IO system
System bus
Program control unit PCU
Address register
Instruction register
Program counter
Address-generation logic
Control circuits
Stack pointer
. . .
Internal control signals

Pipelining

- Parallel processing can be done by using the techniques of pipeline processing which is an implementation technique where arithmetic suboperation or the phases of a computer instruction cycle overlap in execution.

- Pipelining is a technique of decomposing a sequential process into suboperation, with each subprocess being executed in a special dedicated segment that operates concurrently with all the other segments.

Pipelining Contd…..

- The pipeline organization will be demonstrated by means of a simple example:
  - Ai*Bi+Ci   for i=1,2,3,…….,7

    - R1 ← Ai   R2 ← Bi
    - R3 ← R1*R2   R4 ← Ci
    - R5 ← R3+R4

    - Hardware implementation & Clock pulses

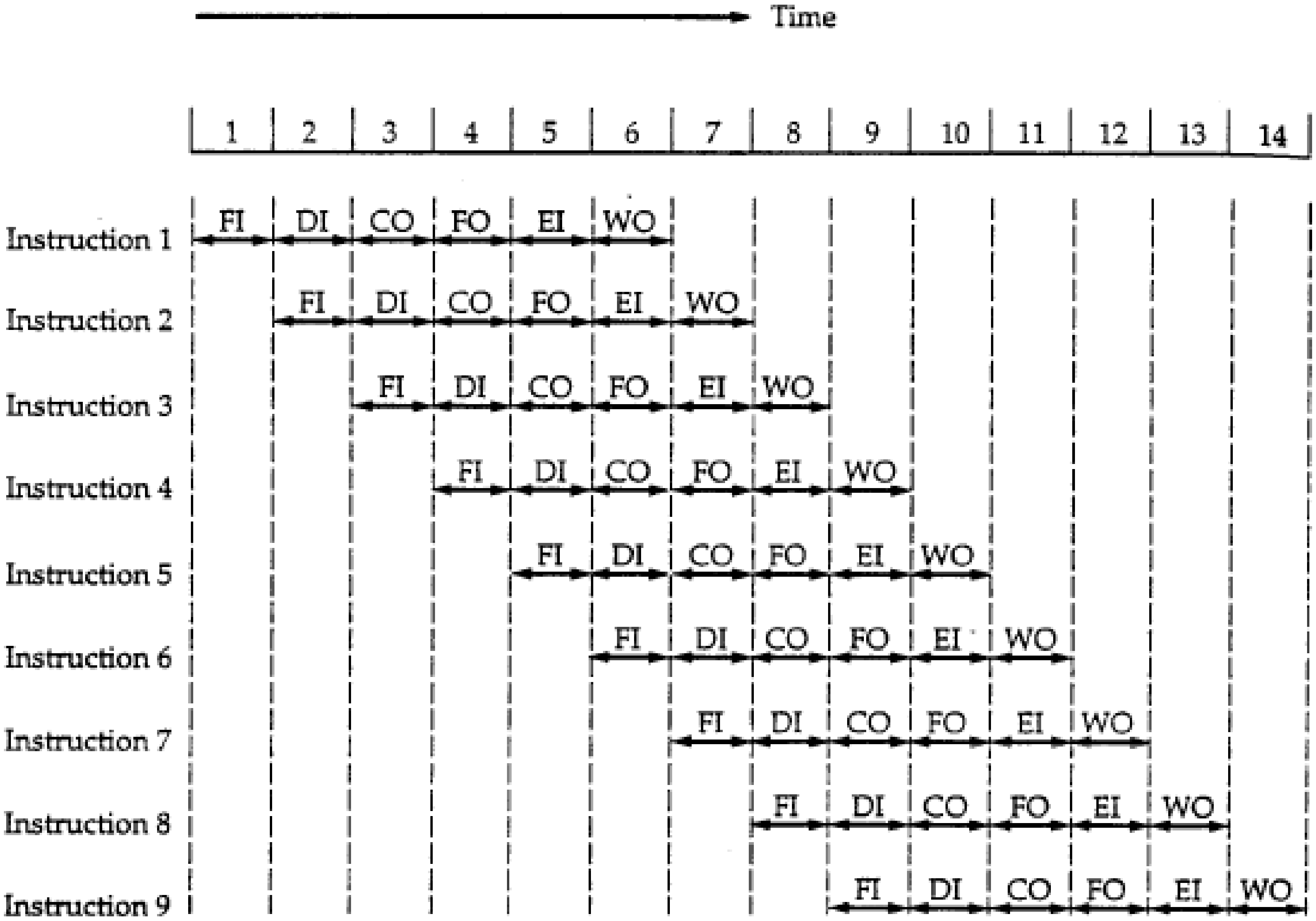
- There are two areas of computer design where the pipeline organization is applicable:
  - An Arithmetic Pipeline
  - An Instruction Pipeline

- Arithmetic Pipeline: An arithmetic pipeline divides an arithmetic operation in suboperations for execution in the pipeline segments.
- The floating point addition and subtraction can be performed in 4 segments are:
  - Compare the exponent
  - Align the mantissas
  - Add or subtract the mantissas
  - Normalize the result

- Instruction Pipeline: An instruction pipeline reads consecutive instructions from memory while previous instructions are being executed in other segments.
- The instructions are inserted into the FIFO buffer.
- The instruction cycle consist of:

Finer division of the instruction cycle: use a 6-stage pipeline

- Instruction fetch
- Decode opcode
- Calculate operand address(es)
- Fetch operands
- Perform execution
- Write (store) result



Time →

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Instruction 1	FI	DI	CO	FO	EI	WO								
Instruction 2		FI	DI	CO	FO	EI	WO							
Instruction 3			FI	DI	CO	FO	EI	WO						
Instruction 4				FI	DI	CO	FO	EI	WO					
Instruction 5					FI	DI	CO	FO	EI	WO				
Instruction 6						FI	DI	CO	FO	EI	WO			
Instruction 7							FI	DI	CO	FO	EI	WO		
Instruction 8								FI	DI	CO	FO	EI	WO	
Instruction 9									FI	DI	CO	FO	EI	WO

ARM6 Microprocessor

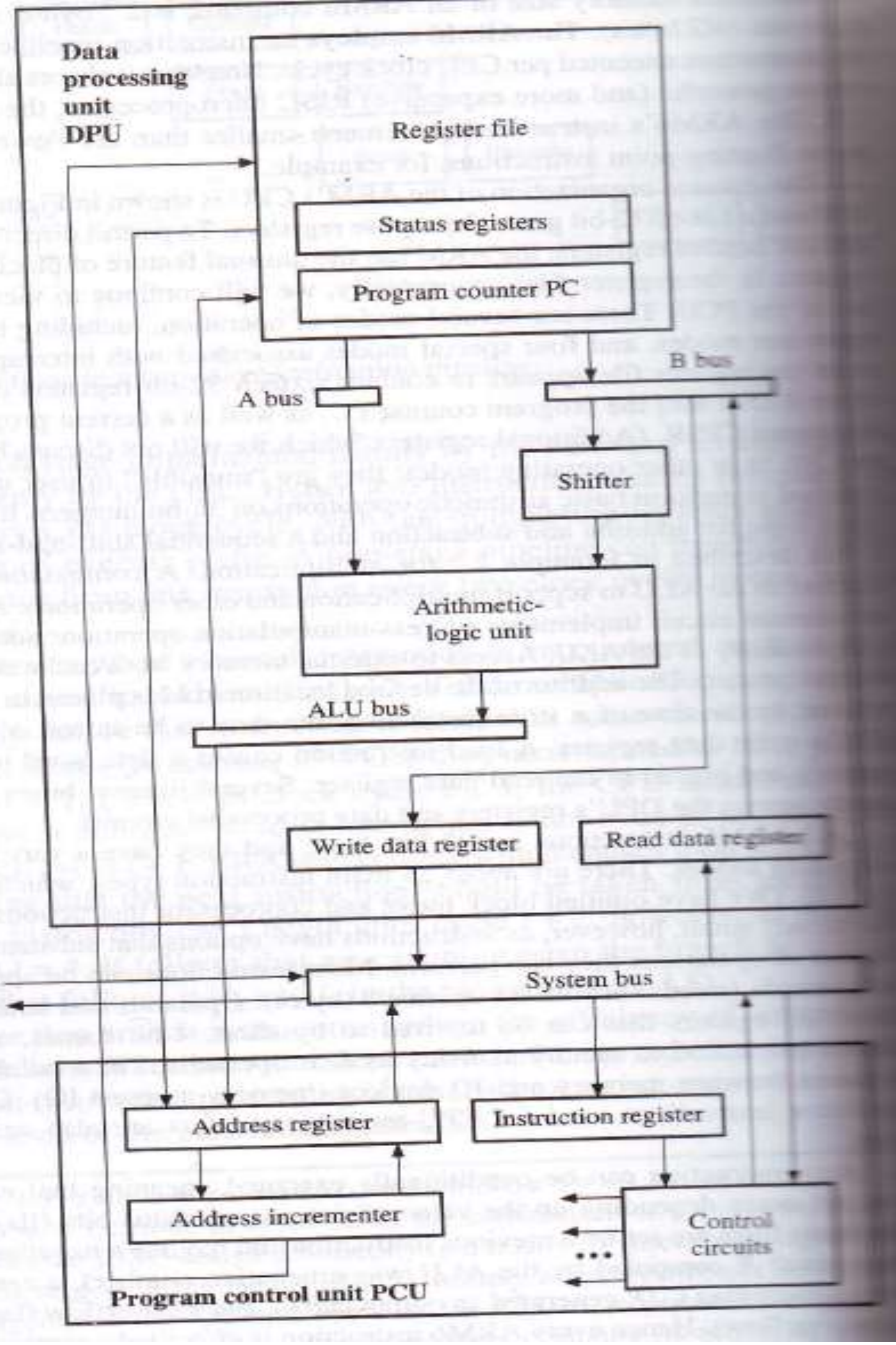
- Here ARM stands for Acorn RISC Machine or Advanced RISC Machine.
- RISC stands for Reduced Instruction Set Computer.
- In early computers, we have to use fewer instructions with simple constructs so they can be executed much faster within the CPU without having to use memory as often.

The Major Characteristics of RISC Are:

- Relatively few instructions
- Relatively few addressing modes
- Memory access limited to load and store instruction.
- All operations done with in the registers of the CPU.
- Fixed length instruction format.
- Single cycle instruction execution.
- Hardwired control rather than microprogrammed control.

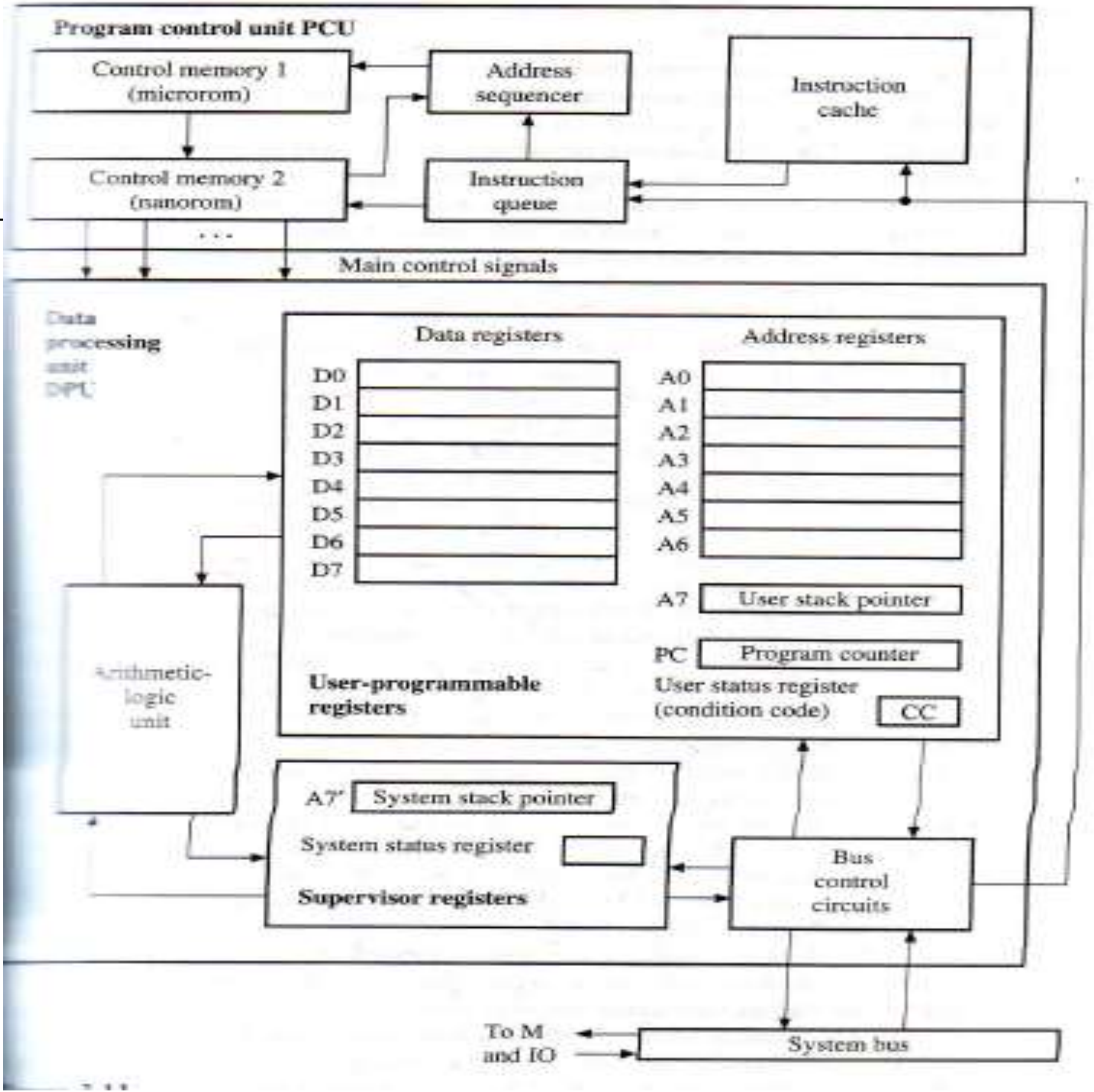
- The ARM6 has 32 bit ALU and a register file of 32 bit general purpose registers.
- In this, there are several modes of operation that includes
  - User mode
  - Supervisor Mode
  - Four Special Modes which is associated with interrupt handling

- Here the ALU employs combinational logic for addition and subtraction and a sequential shift and add method for multiplication.
- A separate address incrementer circuits implements the address manipulation operations.
- There are about 25 instructions


Data processing unit DPU
Register file
Status registers
Program counter PC
B bus
A bus
Shifter
Arithmetic-logic unit
ALU bus
Write data register
Read data register
System bus
To M and I/O
Address register
Instruction register
Address incrementer
Control circuits
...
Program control unit PCU

A CISC Machine (68020)

- A computer with a large number of instructions is classified as a Complex Instruction Set Computer.
- The major characteristics of CISC architecture are:
  - A large no of instructions-from 100 to 250 instructions.
  - Some instructions that perform specialized task.
  - A large variety of addrressing modes
  - Instructions that manipulate operands in memory.


Program control unit PCU
Control memory 1 (microrom)
Address sequencer
Instruction cache
Control memory 2 (nanorom)
Instruction queue
. . .
Main control signals
Data processing unit DPU
Data registers
Address registers
D0
D1
D2
D3
D4
D5
D6
D7
A0
A1
A2
A3
A4
A5
A6
A7
User stack pointer
PC
Program counter
User-programmable registers
User status register (condition code)
CC
Arithmetic-logic unit
A7'
System stack pointer
System status register
Supervisor registers
Bus control circuits
To M and IO
System bus

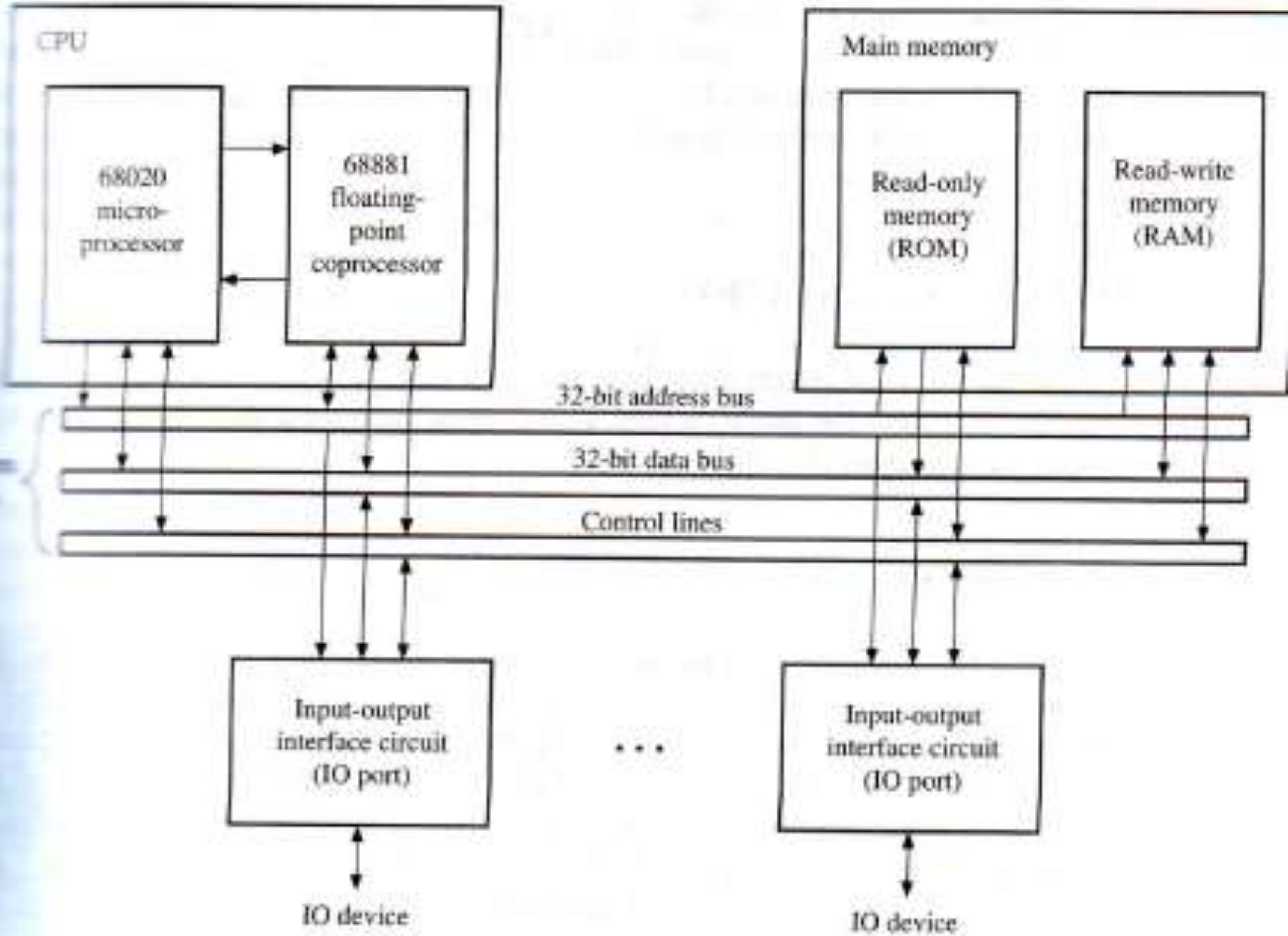
Coprocessors

A secondary processor which is used to speed up operations by taking over a specific part of the main processor’s work. The most common type of coprocessor is the floating point coprocessor which is designed to manage arithmetic calculations many times faster than the main processor.

Coprocessors Contd….

- A coprocessor 68881, a floating point processor from Motorola designed for use with the 68000 & 68020 chips.
- The 68881 contains a set of eight 80 bit registers for storing floating point numbers of various format including 34 & 64 bit numbers.

- The command executed by the 68881 include the basic arithmetic operations (Add, Subtract, Multiply, Divide), square root, logarithms & trignometric functions.


CPU
68020 micro-processor
68881 floating-point coprocessor
Main memory
Read-only memory (ROM)
Read-write memory (RAM)
32-bit address bus
32-bit data bus
Control lines
Input-output interface circuit (IO port)
. . .
Input-output interface circuit (IO port)
IO device
IO device

Data Representation

- Binary information in digital computers is stored in memory or processor register.
- Register contain either data or control information.
- Information is again divided into 2 parts:
  - Instruction
  - Data

- Information
    - Instruction
    - Data
        - Numeric
            - Fixed Point
                - Binary
                - Decimal
            - Floating Point
                - Binary
                - Decimal
        - Non-Numeric

Important Points in Data Representation:

- Word Length
- Storage Order
- Tag
- Error Detection & Correction

Word Length

- Information is represented in digital computer by means of binary words.
- Word is a unit of information of some fixed length n.
- Word size is typically a multiple of 8.
- In Motorola 680X0 the term word is restricted to 4 bytes

Instruction Format of 680X0

- 1 Bit - Status Flag
- 8 Byte - Smallest Addressable Memory Item
- 16 Halfword -Short Fixed Point Number
- 32 Word - Fixed or Floating Point number
- 64 Double Word -Long Instruction

Storage Order

- One of the important aspect of data representation is the way in which the words are indexed.
- The most important convention is that the low order indexes corresponds to the numerically less significant bits and the high order indexes corresponds to the numerically more significant bits.
- Suppose each word consist of 4 byte.

- Basically there are two types of storage method:
  - Big Endian Byte Order
  - Little Endian Byter Order
- When the most significant byte of a word is stored in the lowest address and the least significant byte is assigned to the highest addresses the this method is known as Big Endian Byte Order.
- When the lowest address is assigned to byte 0, Little Endian Byte Order.

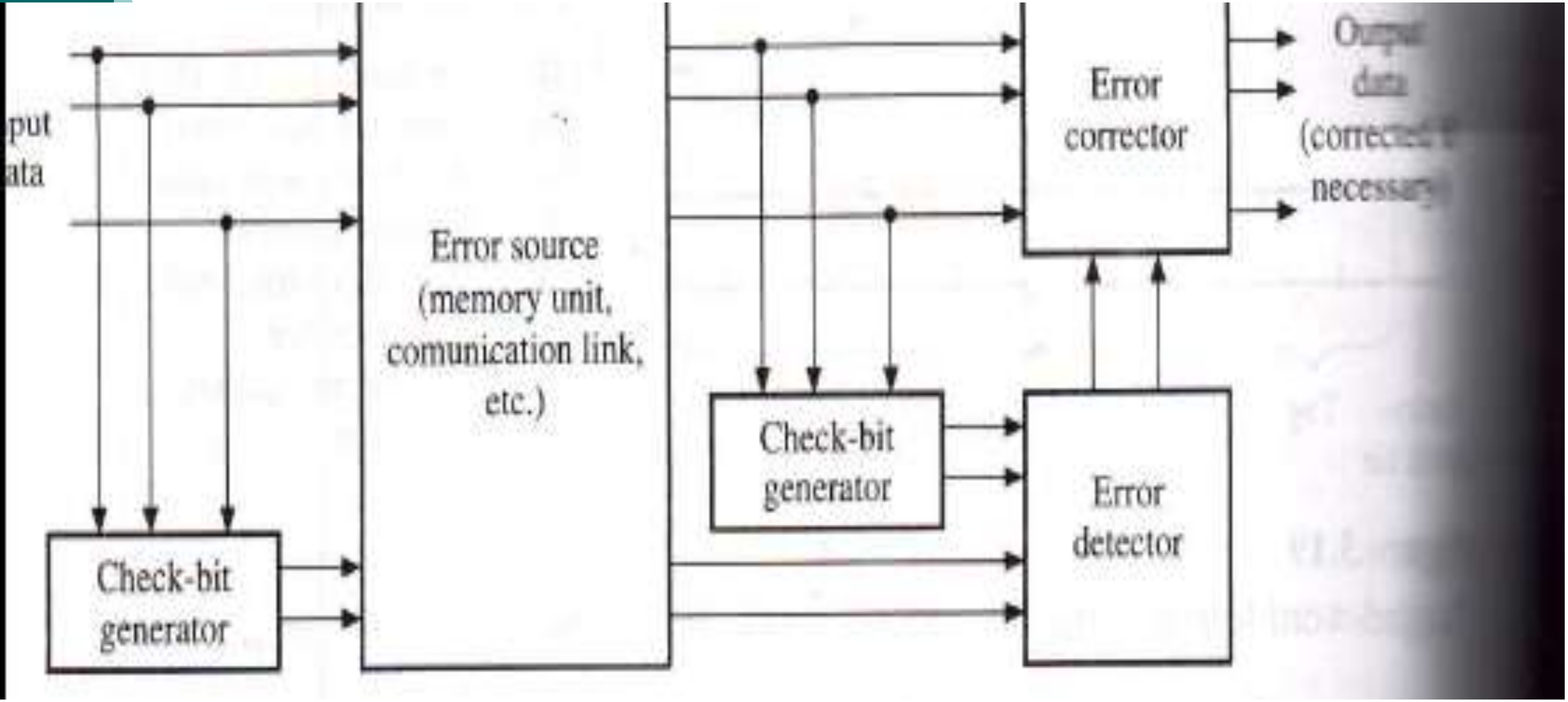
Tag

- Tag identifies the type of word.
- Tagging simplifies the instruction specification.
- The Tag field consist of 3 bits in B6500/7500 series.

Tag Contd…

- 000 - Single Precision Number
- 001 –Indirect Reference Word
- 010 – Double Precision Number
- 011 – Segment Descriptor
- 100 – Step-Index Control Word
- 101 – Data Descriptor
- 110 – Uninitialized Operand
- 111-Instruction

Error Detection & Correction



Number Representation

- While selecting a number representation to be used in a computer, the following factors should be taken into account:
  - The number types to be represented.
  - The range of values likely to be encountered.
  - The cost of hardware required to store & process the numbers.

- The two most common number format are:
  - Fixed Point Representation
  - Floating Point Representation

Fixed Point Representation

- While a number is represented the two most important things are:

- Because of hardware limitation the computer must represent every thing in 1’s & 0’s, even the sign of a number is also represented by 1 & 0. If it is 0 then the no is positive, if it is 1 then the no is negative.

Some Points Related To Fixed Point Representation:

- Signed Number
- Exceptional Conditions
- Decimal Number

Signed Numbers

- When a no is negative then it is represented in one of the 3 possible ways:
  - Signed Magnitude Representation
  - Signed 1’s Complement Representation
  - Signed 2’s Complement Representation

Exceptional Conditions

- When 2 numbers of n digits are added and the sum occupies n+1 digits, we say that an overflow is occurred.
- An overflow is a problem in digital computer because the width of a register is finite.
- An overflow cannot occur when we operate on a +ve & a –ve number

- An overflow may occur if the two numbers added are both positive or negative.
- An overflow condition can be detected by observing the carry into the sign bit position & the carry out of the sign bit position. If these two carries are not equal then an overflow is occurred.

Decimal Numbers

- Decimal numbers are stored in computer after converting it into binary, for this decimal to binar conversion is used or there are certain binary codes that helps you in converting the decimal number to binary. Some of the Codes are:
    - BCD (8421, 2421)
    - ASCII
    - EBCDIC
    - Gray
    - Excess-3

Floating Point Representation

- In addition to the sign of a number , a number may have decimal point . There are two ways of specifying the position of the binary point in a register.
  - By giving it a fixed position.
  - By employing a floating point representation.

Floating Point Representation Contd…

- A floating point number consist of two parts:
- The first part represents a signed, a fixed point number called the mantissa.
- The second part designates the position of the decimal point which is called the exponent.

Some Imp Points For This Representation is:

- Normalization
- Biasing

Instruction Format

- The purpose of an instruction is to specify both an operation to be carried out by a CPU or other processor and the set of operands or data to be used in the operation.

- The most common fields found in instruction format are:
  - Opcode
  - Mode
  - Address Field

Different Types of CPU Organization:

- Single Accumulator Register
- General Register Organization
- Stack Organization

Single Accumulator Register

- All operations are performed with an implied accumulator register.

- For Eg:- ADD X

  $AC \leftarrow AC+M[X]$

General Register Organization

- The instruction format of this type of computer needs three register address fields.

- The no of address fields in the instruction can be reduced from 3 to 2 if the destination register is same as that of the source register.

- ADD R1, R2, R3
    - R1 ← R2+R3
- ADD R1, R2
    - R1 ← R1+R2
- ADD R1, X
    - R1 ← R1+M[X]

Stack Organization

- Computers with stack organization would have PUSH & POP instructions which require an address field.

Types of Instruction Format

- Three Address Instruction
- Two Address Instruction
- One Address Instruction
- Zero Address Instruction

A Selection of Instruction Format of the Motorola 680X0 series:

- Opcode
  - F1
- Opcode R1 Opcode
  - F2
- Opcode R1 Opcode R2
  - F3

- Opcode/Register-Immediate Operand (Short)--------→32 bits
  - F4
- Opcode/Register-Memory Address ADR1 (Short)--------→32 bits
  - F5
- Opcode/Register-Immediate Operand (Long)--------→47 bits
  - F6

- Opcode/Register-Memory Address ADR1 --------→47 bits
  - F7
- Opcode/Register-Memory Address ADR1 – Memory Address ADR2--------→79 bits
  - F8

- For Eg:
  - ADD.L D1, D2
  - ADD.L ADR1, D2
  - Move.B ADR1, ADR2

Imp Points Related To Instruction Set:

- RISC Format
- Operand Extension

Addressing Modes

- Implied Addressing Mode:- In this mode the operand is implicitly specified in the instruction. For Eg: CLR.

- Immediate Addressing Mode:- In this mode the operand is specified in the instruction, it means that the instruction itself contain the operand. There is no need of accessing the memory to fetch the operand. For Eg: Load #3, MOV R1, #4.

- Direct Addressing Mode: In this mode the address part contain the address of the operand.

- Indirect Addressing Mode:- In this mode the address part of the instruction contain the address of the effective address. For Eg: LOAD @A

○ Register Addressing Mode:- Direct & Indirect addressing mode is used in case of memory reference instruction and if the operand is stored in register then register addressing mode is used.

○ Register Indirect Addressing Mode:- The address part of the instruction contain the address of the register where the address of the operand is stored.

- AutoIncrement & AutoDecreement Mode
- Displacement Addressing Mode
  - Relative Addressing Mode
  - Indexed Addressing Mode
  - Base Addressing Mode