Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 01 : Introduction and Background

1. Explain generation of computers.
2. Von Neumann Architecture and Store Program Concept
3. Explain about various logic gates in brief.
4. Discuss about special gates in brief.
5. Discuss about universal gates in brief.
6. What is combinational circuit ? Write the name of some combinational circuit.
7. What do you mean by sequential circuit ? Explain varios types of flip-flop.
8. Draw and explain multiplexer in brief.
9. What is counter ? Explain in details about various types of counters ?
10. Discuss about encoder and decoder in brief.
11. What is flip-flop ? Discuss D flip-flop in details.
12. What do you mean by random access memory ?

Evolution/Generation of Computers

Generation : It means a state of improvement in the product development process.

First Generation(1940-56) : Vacuum Tubes

- First-generation computers used a very large number of vacuum tubes for circuitry and magnetic drums for memory.
- These computers were so big in size that they are often required an entire room to be installed.
- It is very expansive, it consumed a lot of electricity and generate more heat.
- It could be programmed using machine language(low level language) consisting of 1s and 0s.
- These computers could solve only one problem at a time.
- Examples :
  - UNIVAC(Universal Autometic Computer)
  - ENIAC(Electronic Numerical Integrator And Calculator)

Second Generation(1956-63) : Transitor

- Second generation computer manufactured using transitors rather than vacuum tubes.
- Transistors were far superior to vacuum tubes.
- Computers manufactured using transistors were smaller, faster, cheaper, and more energy-efficient and reliable.
- It could be programmed using symbolic or assembly language

Third Generation(1964-71) : Integrated Circuit(IC)

- Third generation computer manufactured using Integrated circuit rather than transitors.
- Integrated Circuit - Several electronic components such as transistors, resisters and capacitors are minimized and placed on silicon chips called integrated chips.

- Integratd chips were smaller, less expansive, more reliable and faster in operation.
- It could be programmed using high level language like COBOL and FORTRAN.

Fourth Generation(1971-89) : Microprocessor

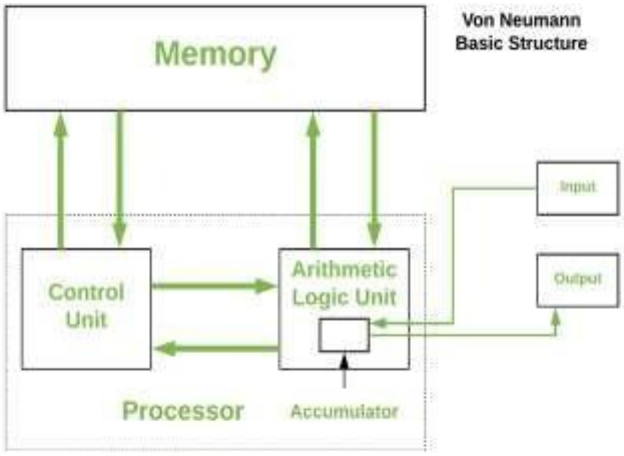
- Fourth generation computer manufactured using microprocessor rather than Integrated circuit.
- Microprocessor - Thousands of integrated circuits built onto a single silicon chip.
- During this period, many new operating system were developed including MS-DOS, MS Windows, UNIX and Apple.

Fifth Generation(Present & Beyond) : Artificial Intelligence(AI)

- Fifth generation computers are completely based on the new concept of artificial intelligence(AI).
- During this period, computers are still in development, there are certain applications such as voice recognition.
- AI touches the following areas :
  - **Expert System :** Computers are programmed to take decisions in real world situations.
  - **Natural language :** Computers are programmed to understand and respond to natural human language.
  - **Neural networks :** Systems are programmed to simulate intelligence by reproducing the physical connections that take place in animal brains.
  - **Robotics :** Computers are programmed to look, listen and react to other sensory stimuli.

Von Neumann Architecture and Store Program Concept

- The Stored Program Concept is the idea that instructions and data are stored together in memory.
- The program is then fetched from memory an instruction at a time and excuted.
- This concept was developed by Jon Von Neumann.



It is also known as IAS **computer and is having three basic units:**

Control Unit(CU)

- A control unit (CU) handles all processor control signals.
- It directs all input and output flow, fetches code for instructions and controlling how data moves around the system.

Arithmetic and Logic Unit (ALU)

- The arithmetic logic unit is that part of the CPU that handles all the calculations the CPU may need, e.g. Addition, Subtraction, Comparisons.
- It performs Logical Operations, Bit Shifting Operations, and Arithmetic Operation.

What is logic gates ? Explain about various logic gates in brief.

Logic Gates

- Logic gates are the basic building blocks of any digital system.
- It is an electronic circuit having one or more than one input and only one output.
- The relationship between the input and the output is based on a certain logic.
- Logic gates are named as AND gate, OR gate, NOT gate etc.

AND Gate

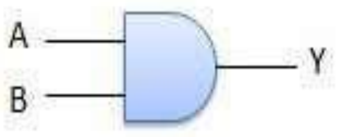
- A circuit which performs an AND operation is shown in figure.
- It has n input (n >= 2) and one output.

Y = A AND B AND C .... N

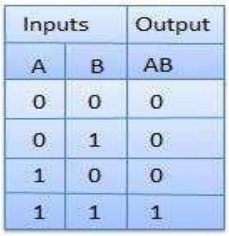
Y = A.B.C .... N

Y = ABC .... N

Logic diagram



Truth Table



Inputs		Output
A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

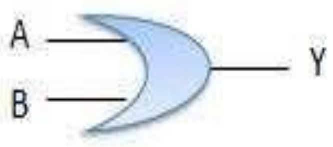
OR Gate

- A circuit which performs an OR operation is shown in figure.
- It has n input (n >= 2) and one output.

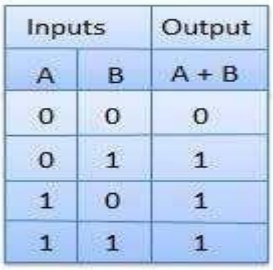
Y = A OR B OR C .... N

Y = A + B + C .... N

Logic diagram



Truth Table



Inputs		Output
A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

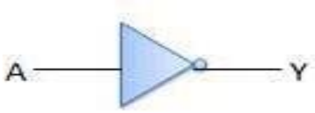
NOT Gate

- NOT gate is also known as **Inverter**.
- It has one input A and one output Y.

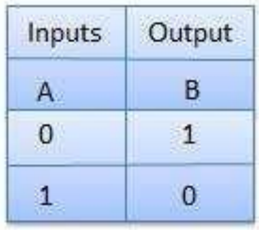
Y = NOT A

Y = $\overline{A}$

Logic diagram



Truth Table



Inputs	Output
A	B
0	1
1	0

Discuss about special gates in brief.

XOR Gate

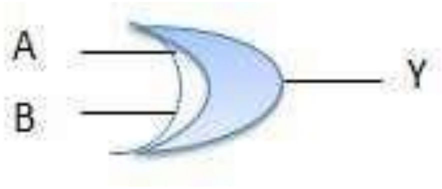
- XOR gate is a special type of gate.
- It can be used in the half adder, full adder and subtractor.
- The exclusive-OR gate is abbreviated as EX-OR gate or sometime as X-OR gate.
- It has n input (n >= 2) and one output.

Y = A XOR B XOR C .... N

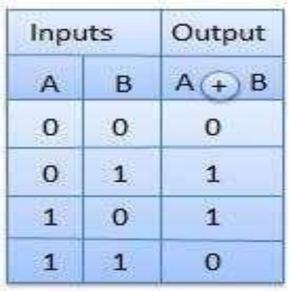
Y = $A \oplus B \oplus C$ .... N

Y = $\overline{A}B + \overline{A}B$

Logic diagram



Truth Table



Inputs		Output
A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

XNOR Gate

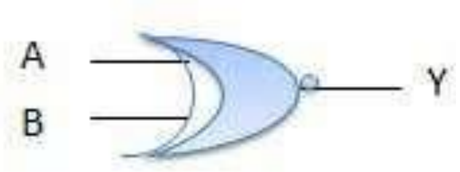
- ➢ XNOR gate is a special type of gate.
- ➢ It can be used in the half adder, full adder and subtractor.
- ➢ The exclusive-NOR gate is abbreviated as EX-NOR gate or sometime as X-NOR gate.
- ➢ It has n input (n >= 2) and one output.

Y = A XOR B XOR C .... N

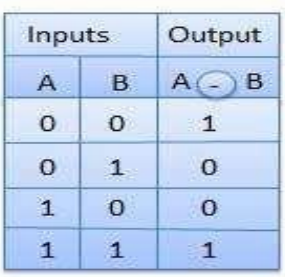
$Y = A \odot B \odot C .... N$

$Y = \overline{A}\,\overline{B} + AB$

Logic diagram



Truth Table



Inputs		Output
A	B	$A \odot B$
0	0	1
0	1	0
1	0	0
1	1	1

Discuss about universal gates in brief.

- A universal logic gate is a logic gate that can be used to construct all other logic gates.
- NAND and NOR gates are widely known to be universal logic gates,

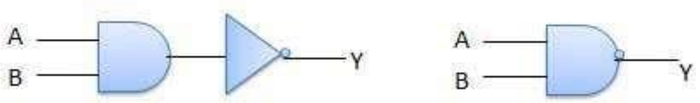
NAND Gate

- ➢ A NOT-AND operation is known as NAND operation.
- ➢ It has n input (n >= 2) and one output.

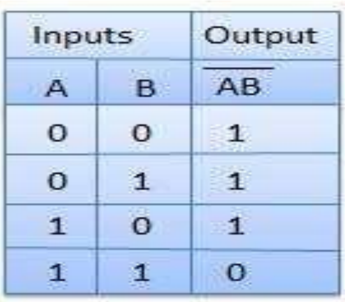
Y = A NOT AND B NOT AND C .... N

Y = A NAND B NAND C .... N

Logic diagram



Truth Table



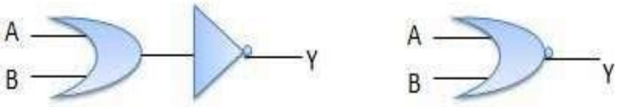
Inputs		Output
A	B	$\overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

NOR Gate

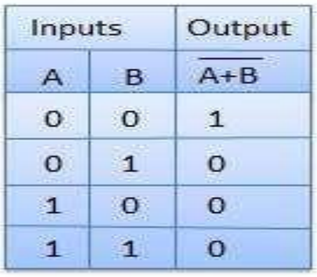
- ➢ A NOT-OR operation is known as NOR operation.
- ➢ It has n input (n >= 2) and one output.

Y = A NOT OR B NOT OR C .... N
Y = A NOR B NOR C .... N

Logic diagram



Truth Table



Inputs		Output
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

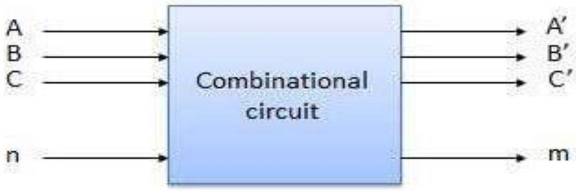
What is combinational circuit ? Write the name of some combinational circuit.

Combinational circuit is a circuit in which we combine the different gates in the circuit, for example encoder, decoder, multiplexer and demultiplexer.

Characteristics of combinational circuits are following −

- The output of combinational circuit at any instant of time, depends only on the levels present at input terminals.
- The combinational circuit do not use any memory.
- The previous state of input does not have any effect on the present state of the circuit.
- A combinational circuit can have an n number of inputs and m number of outputs.

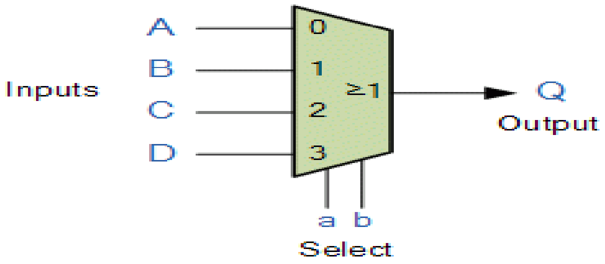
Block diagram



Multiplexers

- It is a combinational circuit which have many data inputs and single output depending on control or select inputs.
- For 2^n input lines, n selection lines are required.
- Multiplexers are also known as
  - ➢ "Data selector,
  - ➢ parallel to serial convertor,
  - ➢ many to one circuit,
  - ➢ universal logic circuit".

Block diagram



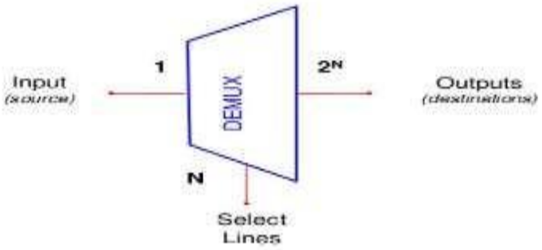
Multiplexers come in multiple variations

- 2 : 1 multiplexer
- 4 : 1 multiplexer
- 16 : 1 multiplexer
- 32 : 1 multiplexer

Demultiplexers

- A demultiplexer performs the reverse operation of a multiplexer.
- It receives one input and distributes it over several outputs.
- It has only one input, n outputs, m select input.
- A Demultiplexer is a data distributor read as DEMUX.

Block diagram



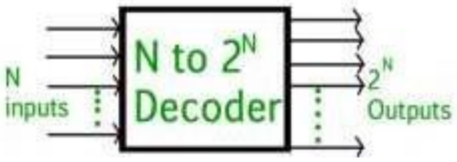
Demultiplexers comes in multiple variations.

- 1 : 2 demultiplexer
- 1 : 4 demultiplexer
- 1 : 16 demultiplexer
- 1 : 32 demultiplexer

Decoder

- A decoder is a combinational circuit.
- It has n input and to a maximum m = 2n outputs.
- Decoder is identical to a demultiplexer without any data input.
- It performs operations which are exactly opposite to those of an encoder.

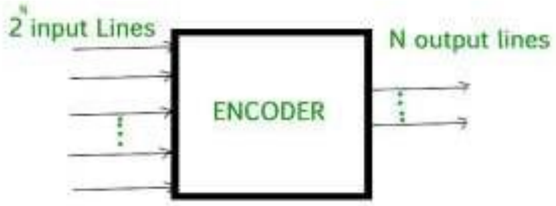
Block diagram



Encoder

- Encoder is a combinational circuit which is designed to perform the inverse operation of the decoder.
- An encoder has n number of input lines and m number of output lines.
- An encoder produces an m bit binary code corresponding to the digital input number.

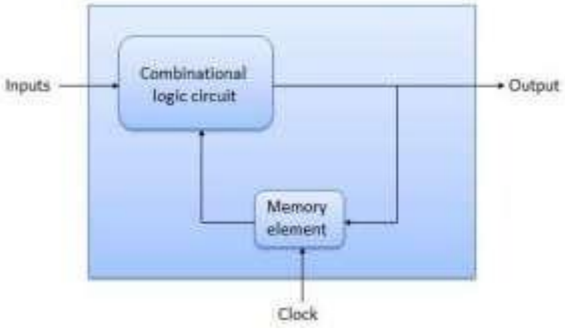
Block diagram



Sequential Circuits

- The combinational circuit does not use any memory.
- Hence the previous state of input does not have any effect on the present state of the circuit.
- But sequential circuit has memory so output can vary based on input.
- In other words, the combinational circuit with memory is called sequential circuit.

Block diagram



Flip-Flop

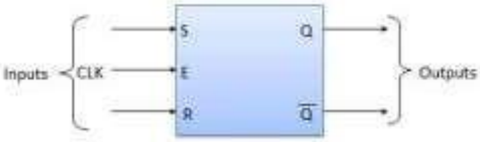
- A flip-flop is a memory element which is capable of storing one bit of information and it is used in clocked sequential circuits.
- A flip-flop has two outputs, one for normal value and other for complement value.
- A flip flop is also known as bistable multivibrator.
- Flip flop is a sequential circuit which generally samples its inputs and changes its outputs only at particular instants of time and not continuously.

Latches	Flip-Flops
A latch doesn’t contain any clock signal.	A flip-flop contains a clock signal.
The structure of Latche is built with logic gates.	FFs are designed with latches by adding an extra clock signal.
Latches are very fast	Flip-Flops (FFs) are very slow
Latches are responsive toward faults on enable pin	FFs are protected toward faults
Latches consume less power A flip-flop can be clocked for all time	FFs consume more power A Latch may be clockless or clocked

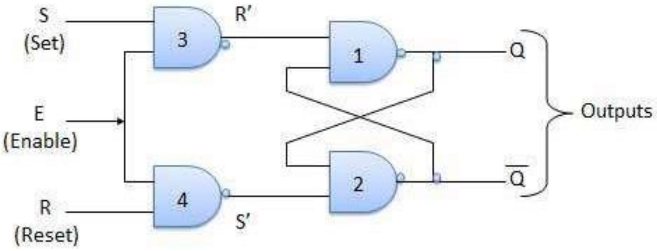
S-R Flip Flop

- It is basically S-R latch using NAND gates with an additional **enable** input.
- It is also called as level triggered SR-FF.
- In short this circuit will operate as an S-R latch if E = 1 but there is no change in the output if E = 0.

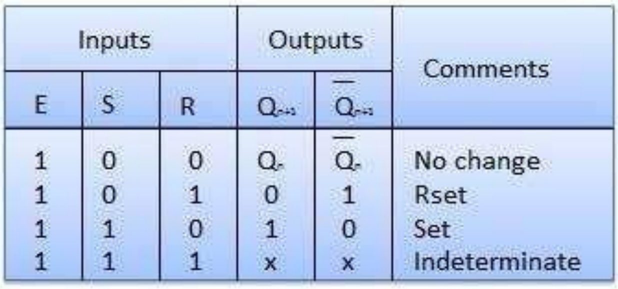
Block Diagram



Circuit Diagram



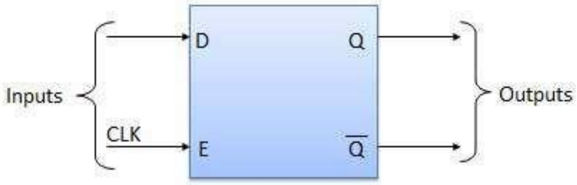
Truth Table



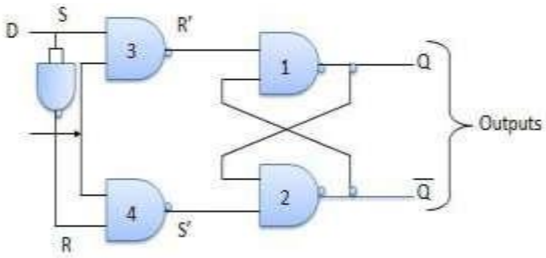
Inputs			Outputs		Comments
E	S	R	Q_{n+1}	$\overline{Q}_{n+1}$	
1	0	0	Q_n	$\overline{Q}_n$	No change
1	0	1	0	1	Rset
1	1	0	1	0	Set
1	1	1	x	x	Indeterminate

D Flip Flop

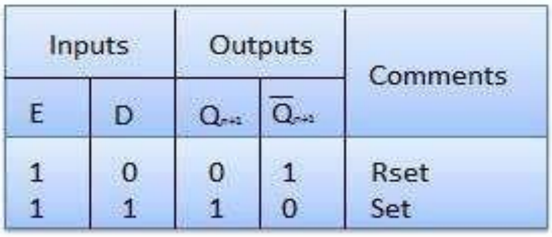
- Delay Flip Flop or D Flip Flop is the simple gated S-R latch with a NAND inverter connected between S and R inputs.
- It has only one input.
- The input data is appearing at the output after some time.
- Due to this data delay between i/p and o/p, it is called delay flip flop. S and R Block Diagram



Circuit Diagram



Truth Table

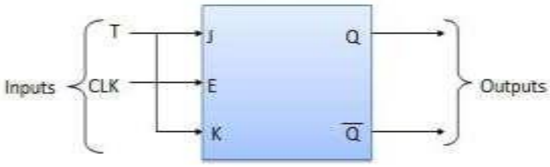


Inputs		Outputs		Comments
E	D	Q_{n+1}	$\overline{Q}_{n+1}$	
1	0	0	1	Rset
1	1	1	0	Set

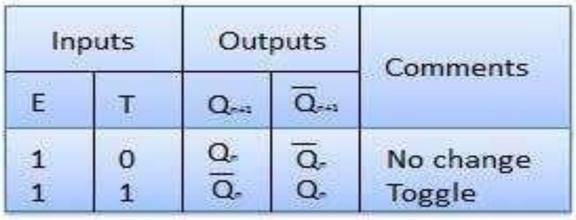
T Flip Flop

- Toggle flip flop is basically a JK flip flop with J and K terminals permanently connected together.
- It has only input denoted by **T** as shown in the Symbol Diagram.

Block Diagram



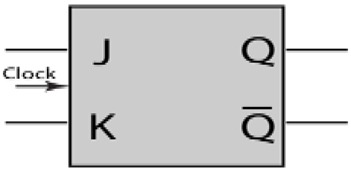
Truth Table



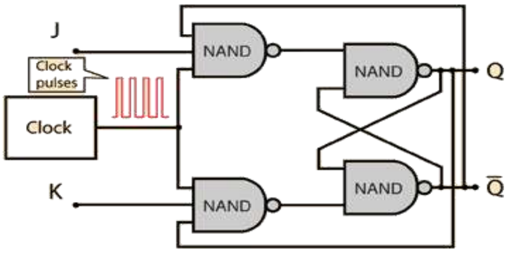
Inputs		Outputs		Comments
E	T	Q_{n+1}	$\overline{Q}_{n+1}$	
1	0	Q_n	$\overline{Q}_n$	No change
1	1	$\overline{Q}_n$	Q_n	Toggle

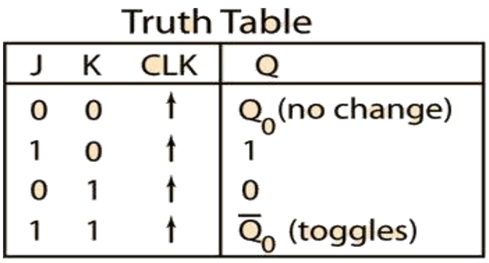
J-K Flip-Flop

- The J-K flip-flop is the most versatile of the basic flip-flops.
- It has the input- following character of the clocked D flip-flop but has two inputs,traditionally labeled J and K.
- If J and K are different then the output Q takes the value of J at the next clock edge.
- The inputs are labeled J and K in honor of the inventor of the device, Jack Kilby.



J-K Type





Truth Table

J	K	CLK	Q
0	0	↑	Q_0(no change)
1	0	↑	1
0	1	↑	0
1	1	↑	$\overline{Q}_0$ (toggles)

Digital Registers

- Flip-flop is a 1 bit memory cell which can be used for storing the digital data.
- To increase the storage capacity in terms of number of bits, we have to use a group of flip-flop.
- Such a group of flip-flop is known as a **Register**.
- The **n-bit register** will consist of **n** number of flip-flop and it is capable of storing an **n-bit** word.

Shift registers

- The binary data in a register can be moved within the register from one flip-flop to another.
- The registers that allow such data transfers are called as **shift registers**.

There are four mode of operations of a shift register.

1. Serial Input Serial Output
2. Serial Input Parallel Output
3. Parallel Input Serial Output
4. Parallel Input Parallel Output

Bidirectional Shift Register

- If a binary number is shifted left by one position then it is equivalent to multiplying the original number by 2.
- Similarly if a binary number is shifted right by one position then it is equivalent to dividing the original number by 2.
- If we should be able to move the data in either left or right direction.
- Such a register is called bi-directional register.

Universal Shift Register

- A shift register which can shift the data in only one direction is called a uni-directional shift register.
- A shift register which can shift the data in both directions is called a bi-directional shift register.
- Applying the same logic, a shift register which can shift the data in both directions as well as load it parallely, is known as a universal shift register.
- The shift register is capable of performing the following operation −
    - Parallel loading
    - Lift shifting
    - Right shifting

Digital Counters

- Counter is a sequential circuit.
- A digital circuit which is used for a counting pulses is known counter.
- Counter is the widest application of flip-flops.
- It is a group of flip-flops with a clock signal applied.

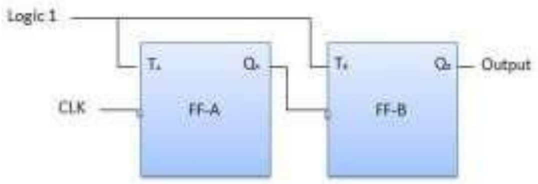
Counters are of two types.

1. **Asynchronous or ripple counters.**
2. **Synchronous counters.**

Asynchronous or ripple counters

- The logic diagram of a 2-bit ripple up counter is shown in figure.
- The toggle (T) flip-flop are being used.
- But we can use the JK flip-flop also with J and K connected permanently to logic 1.
- External clock is applied to the clock input of flip-flop A and Q_A output is applied to the clock input of the next flip-flop i.e. FF-B.

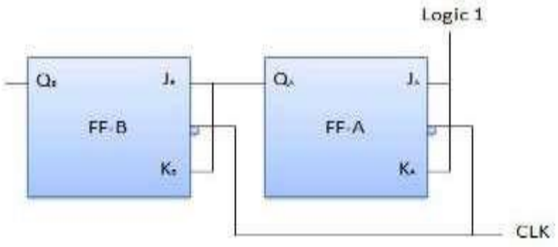
Logical Diagram



Synchronous counters

If the "clock" pulses are applied to all the flip-flops in a counter simultaneously, then such a counter is called as synchronous counter.

Logical Diagram



Random Access Memory(RAM)

- It stands for Random Access Memory.
- RAM is a form of data storage that can be accessed randomly at any time.
- It allows reading and writing.
- RAM is volatile i.e. its contents are lost when the device is powered off.

Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 02 : Register transfer language and micro-operations

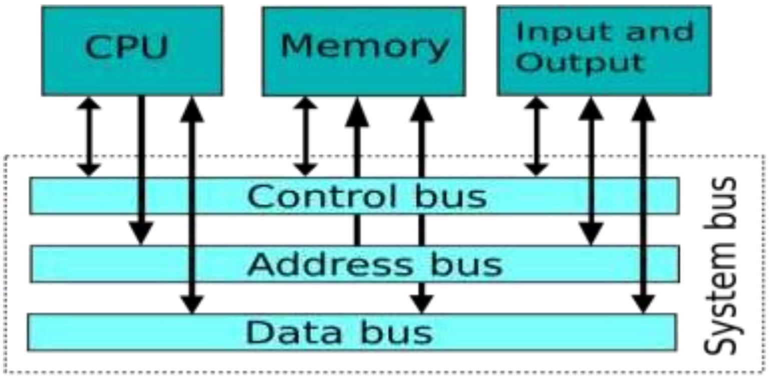
1. What do you mean by bus ? Explain different types of bus.
2. What are the function of following register :
   a. PC
   b. SP
   c. MAR
   d. IR
3. Explain data movement from/to memory.
4. Explain different types of Data Movement Instructions.

What do you mean by bus ? Explain different types of bus.

- A collection of wires through which data is transmitted from one part of a computer to another is known bus.
- A bus as a highway on which data travels within a computer.
- The term bus usually refers to internal bus.
- This is a bus that connects all the internal computer components to the CPU and main memory.

There are three types of bus :--

- **Data bus :** The data bus transfers actual data.
- **Address bus :** The address Bus transfers information about where the data should go.
- **Control bus :** Control bus is used to transmit different commands or control signals from one component to another component.



Universal Serial Bus (USB)

- This is an external bus standard that supports data transfer rates of 12 Mbps.
- A single USB port can be used to connect up to **127 peripheral devices**, such as mice, modems, and keyboards.

We have two versions of USB:-

USB 1X :

- Released in 1996, the original USB 1.0 standard offered data rates of 1.5 Mbps.

USB 2X :

- In 2002 a newer specification USB 2.0, also called Hi-Speed USB 2.0, was introduced.
- It increased the data transfer rate for PC to USB device to 480 Mbps, which is 40 times faster than the USB 1.1 specification.

Some basic register

Accumulator

- The accumulator is an 8-bit register that is a part of ALU.
- This register is used to store 8-bit
- data and to perform arithmetic and logical operations.
- The result of an operation is stored in the accumulator.
- The accumulator is also identified as register A.

Flag register

- The ALU includes five flip-flops, which are set or reset after an operation according to data condition of the result in the accumulator and other registers.
- They are called Zero (Z), Carry (CY), Sign (S), Parity (P) and Auxiliary Carry (AC) flags.
- The microprocessor uses these flags to test data conditions.

Program Counter (PC)

- This 16-bit register deals with sequencing the execution of instructions.
- The function of the program counter is to point to the memory address from which the next byte is to be fetched.
- When a byte is being fetched, the program counter is automatically incremented by one to point to the next memory location.

Stack Pointer (SP)

- The stack pointer is also a 16-bit register, used as a memory pointer.
- It points to a memory location in R/W memory, called stack.
- The beginning of the stack is defined by loading 16-bit address in the stack pointer.

Instruction Register/Decoder

- It is an 8-bit register that temporarily stores the current instruction of a program.
- Latest instruction sent here from memory prior to execution.
- Decoder then takes instruction and decodes or interprets the instruction.
- Decoded instruction then passed to next stage.

Explain Data movement from/to memory

Data movement instructions move data from one place, called the source operand, to another place, called the destination operand. There are six data movement instruction which is used to transfer data.

- MOV R_1, R_2
- MOV M, R
- MOV R, M
- MVI R, Data
- MVI M, Data
- LXI SP, Data

Explain different types of Data Movement Instructions

These instructions provide convenient methods for moving bytes, words, or doublewords of data between memory and the registers of the base architecture. They fall into the following classes:

1. General-purpose data movement instructions.
2. Stack manipulation instructions.
3. Type-conversion instructions.

1. General-Purpose Data Movement Instructions

MOV (Move) transfers a byte, word, or doubleword from the source operand to the destination operand.

1. To a register from memory
2. To memory from a register
3. Between general registers
4. Immediate data to a register
5. Immediate data to a memory

2. Stack Manipulation Instructions

- PUSH (Push) decrements the stack pointer, then transfers the source operand to the top of stack.
- POP (Pop) transfers the word or doubleword at the current top of stack to the destination operand, and then increments ESP to point to the new top of stack.

3. Type Conversion Instructions

- The type conversion instructions convert bytes into words, words into doublewords, and doublewords into 64-bit items (quad-words).
- These instructions are especially useful for converting signed integers, because they automatically fill the extra bits of the larger item with the value of the sign bit of the smaller item.

Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 03 : Architecture of Simple Processor :

1. What do you mean by instruction set ?
2. Define the term micro-instruction ?
3. Mention the different types of instruction format ?
4. Differentiate between RISC and CISC.

What do you mean by instruction set ?

- An instruction set is a group of commands for a CPU in machine language.
- The instruction set, also called instruction set architecture(ISA).
- The instruction set provides commands to the processor, to tell it what it needs to do.
- There are three types of Instructions set.

Arithmetic/Logic Instructions:
These Instructions perform various Arithmetic & Logical operations on one or more operands.

Data Transfer Instructions:
These instructions are responsible for the transfer of instructions from memory to the processor registers and vice versa.

Branch and Jump Instructions:
These instructions are responsible for breaking the sequential flow of instructions and jumping to instructions at various other locations, this is necessary for the implementation of functions and conditional statements.

- **COMPARE** - Compare numbers.
- **IN** - Input information from a device, e.g., keyboard.
- **JUMP** - Jump to designated RAM address.
- **JUMP IF** - Conditional statement that jumps to a designated RAM address.
- **LOAD** - Load information from RAM to the CPU.
- **OUT** - Output information to device, e.g., monitor.
- **STORE** - Store information to RAM.

Define the term micro-instruction ?

- A single instruction in microcode.
- It is the most elementary instruction in the computer, such as moving the contents of a register to the arithmetic logic unit (ALU).
- It takes several microinstructions to carry out one complex machine instruction (CISC).
- Also called a "micro-op" or "µop," microinstructions differ within the same computer family and even the same vendor.

Mention the different types of instruction format ?

- It defines the layout of the bits of an instruction.
- Instruction format describes the internal structures (layout design) of the bits of an instruction.
- An Instruction format must include an opcode, and zero or more operands.
- The address is dependent on an availability of particular operands.
- Designing of an Instruction format is very complex.

General structure of Instruction formate

Mode	Opcode	Operand

Types of Instruction formate :

- Zero-address instruction formate
- One-address instruction formate
- Two-address instruction formate
- Three-address instruction formate

Difference between RISC and CISC

RISC	CISC
It is a Reduced Instruction Set Computer.	It is a Complex Instruction Set Computer.
It emphasizes on software to optimize the instruction set.	It emphasizes on hardware to optimize the instruction set.
It requires multiple register sets to store the instruction.	It requires a single register set to store the instruction.
RISC has simple decoding of instruction.	CISC has complex decoding of instruction.
Uses of the pipeline are simple in RISC.	Uses of the pipeline are difficult in CISC.
The execution time of RISC is very short.	The execution time of CISC is longer.
It has fixed format instruction.	It has variable format instruction.

Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 04 : CPU Organisation :

1. Explain the various types of addressing mode in detail ?
2. What is register ? Explain various mode of register.
3. Discuss the different mapping techniques used in cache memory.

There are five addressing modes in 8085.

1. Immediate Addressing: - An immediate is transferred directly to the register.

Eg: - MVI A, 30H (30H is copied into the register A)
MVI B,40H(40H is copied into the register B).

2. Register Addressing: - Data is copied from one register to another register.

Eg: - MOV B, A (the content of A is copied into the register B)
MOV A, C (the content of C is copied into the register A).

3. Direct Addressing Mode: - Data is directly copied from the given address to the register.

Eg: - LDA 3000H (The content at the location 3000H is copied to the register A).

4. Indirect Addressing Mode: - The data is transferred from the address pointed by the data in a register to other register.

Eg: - MOV A, M (data is transferred from the memory location pointed by the regiser to the accumulator).

5. Implied Addressing Mode: - This mode doesn't require any operand.
The data is specified by opcode itself.

Eg: - RAL
CMP

What is register ? Explain various mode of register.

Digital Registers

- Flip-flop is a 1 bit memory cell which can be used for storing the digital data.
- To increase the storage capacity in terms of number of bits, we have to use a group of flip-flop.
- Such a group of flip-flop is known as a **Register**.
- The **n-bit register** will consist of **n** number of flip-flop and it is capable of storing an **n-bit** word.

Shift registers

- The binary data in a register can be moved within the register from one flip-flop to another.
- The registers that allow such data transfers are called as **shift registers**.

There are four mode of operations of a shift register.

1. Serial Input Serial Output
2. Serial Input Parallel Output
3. Parallel Input Serial Output
4. Parallel Input Parallel Output

Bidirectional Shift Register

- If a binary number is shifted left by one position then it is equivalent to multiplying the original number by 2.
- Similarly if a binary number is shifted right by one position then it is equivalent to dividing the original number by 2.
- If we should be able to move the data in either left or right direction.
- Such a register is called bi-directional register.

Universal Shift Register

- A shift register which can shift the data in only one direction is called a uni-directional shift register.
- A shift register which can shift the data in both directions is called a bi-directional shift register.
- Applying the same logic, a shift register which can shift the data in both directions as well as load it parallely, is known as a universal shift register.
- The shift register is capable of performing the following operation −
    - Parallel loading
    - Lift shifting
    - Right shifting

Discuss the different mapping techniques used in cache memory.

Cache Mapping:

There are three different types of mapping used for the purpose of cache memory which are as follows:

1. Direct mapping,
2. Associative mapping, and
3. Set-Associative mapping.

Direct Mapping

- The simplest technique, known as direct mapping, maps each block of main memory into only one possible cache line.
- An address space is split into two parts index field and a tag field.
- The cache is used to store the tag field whereas the rest is stored in the main memory.
- Direct mapping`s performance is directly proportional to the Hit ratio.

Associative Mapping

- In this type of mapping, the associative memory is used to store content and addresses of the memory word.
- Any block can go into any line of the cache.
- This means that the word id bits are used to identify which word in the block is needed, but the tag becomes all of the remaining bits.
- This enables the placement of any word at any place in the cache memory.
- It is considered to be the fastest and the most flexible mapping form.

Set-associative Mapping

- This form of mapping is an enhanced form of direct mapping where the drawbacks of direct mapping are removed.
- Set associative addresses the problem of possible thrashing in the direct mapping method.
- It does this by saying that instead of having exactly one line that a block can map to in the cache, we will group a few lines together creating a ***set***.
- Then a block in memory can map to any one of the lines of a specific set..
- Set-associative mapping allows that each word that is present in the cache can have two or more words in the main memory for the same index address.
- Set associative cache mapping combines the best of direct and associative cache mapping techniques.

Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 05 : Assembly language programming :

1. What is interrupt ? Explain different types of interrupts.
2. What is subroutine discuss in brief.
3. Discuss in brief pseudo operations.

What is Interrupt? Explain different types of interrupts.

- Interrupt is a mechanism by which an I/O or an instruction can suspend the normal execution of processor and get itself serviced.
- Generally, a particular task is assigned to that interrupt signal.
- In the microprocessor based system the interrupts are used for data transfer between the peripheral devices and the microprocessor.
- An interrupt is a signal sent to the processor that interrupts the current process.
- It may be generated by a hardware device or a software program.
- A hardware interrupt is often created by an input device such as a mouse or keyboard.
- An interrupt is sent to the processor as an interrupt request, or IRQ.

Interrupt Service Routine(ISR)

- A small program or a routine that when executed services the corresponding interrupting source is called as an ISR.

Maskable/Non-Maskable Interrupt

- An interrupt that can be disabled by writing some instruction is known as Maskable Interrupt otherwise it is called Non-Maskable Interrupt.

What is subroutine discuss in brief.

- Subroutine is a block of instruction which carries out a specific & well defined task.
- It is a small program written written & stored separatly.
- It can be called when required.
- A set of Instructions which are used repeatedly in a program can be referred to as Subroutine.
- Only one copy of this Instruction is stored in the memory.
- When a Subroutine is required it can be called many times during the Execution of a Particular program.

X

Discuss in brief pseudo operations.

- Pseudo op stands for "pseudo operation" and is sometimes called "assembler directive".
- These are keywords which do not directly translate to a machine instruction.
- Examples are the ones mentioned in your questions or .data, .def, .desc, .dim, .double, .eject, .else, .elseif, .end and many more.
- A pseudo-operation describes the act of a software sending instructions or code to a hardware device that has not been compiled.
- Afterwards the hardware device translates the instructions into machine language.

Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 06 : Microprogrammed Control Unit

1. Explain micro instruction in details.
2. Explain different types of hazards occurs in a pipeline.
3. What do you mean by instruction cycle, machine cycle and states ?
4. Explain in detail about pipelining processing.

Explain different types of micro instruction in details.

- An instruction that controls data flow and instruction-execution sequencing in a processor at a more fundamental level than machine instructions.
- A series of microinstructions is necessary to perform an individual machine instruction.
- A single instruction in microcode.
- It is the most elementary instruction in the computer, such as moving the contents of a register to the arithmetic logic unit (ALU).
- It takes several micro instructions to carry out one complex machine instruction (CISC).
- Also called a "micro-op" or "µop," micro instructions differ within the same computer.

Types of microinstruction formats:

There are three types of microinstruction formats

1. Horizontal Format, called Horizontal microcode
2. Vertical Format, called Vertical microcode
3. Field-encoded Format

Explain different types of hazards occurs in a pipeline.

- Hazards are problems with the instruction pipeline in CPU microarchitectures when the next instruction cannot execute in the following clock cycle and can potentially lead to incorrect computation results.
- Three common types of hazards are data hazards, structural hazards, and control hazards..

Data hazards

- Data hazards occur when instructions that exhibit data dependence modify data in different stages of a pipeline.
- Ignoring potential data hazards can result in race conditions (also termed race hazards).
- There are three situations in which a data hazard can occur:
    1. read after write (RAW), a true dependency
    2. write after read (WAR), an anti-dependency
    3. write after write (WAW), an output dependency

Structural hazards

- A structural hazard occurs when two (or more) instructions that are already in pipeline need the same resource.
- The result is that instruction must be executed in series rather than parallel for a portion of pipeline.
- Structural hazards are sometime referred to as resource hazards.

Control hazards (branch hazards)[edit]

- To avoid control hazards microarchitectures can:
  - insert a pipeline bubble (discussed above), guaranteed to increase latency, or
  - use branch prediction and essentially make educated guesses about which instructions to insert, in which case a pipeline bubble will only be needed in the case of an incorrect prediction

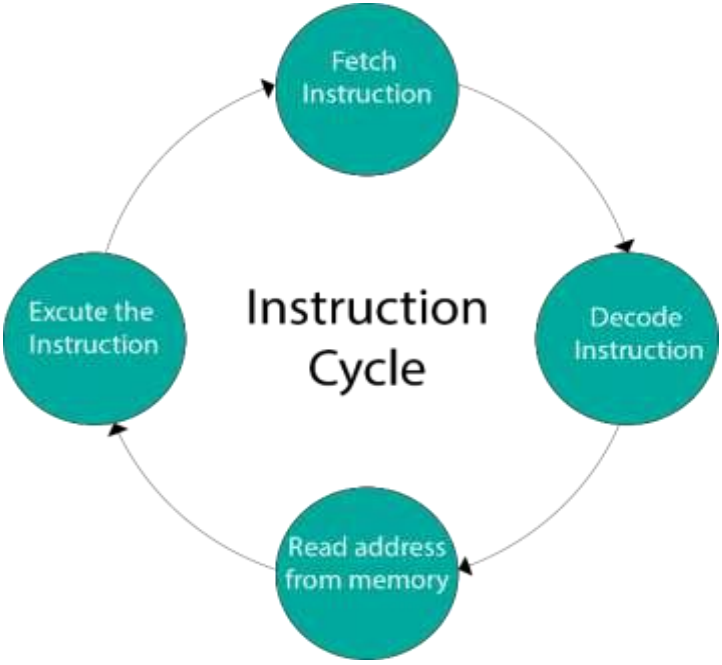
What do you mean by instruction cycle and machine cycle ?

Instruction Cycle

- A program residing in the memory unit of a computer consists of a sequence of instructions.
- These instructions are executed by the processor by going through a cycle for each instruction.

In a basic computer, each instruction cycle consists of the following phases:

1. Fetch instruction from memory.
2. Decode the instruction.
3. Read the effective address from memory.
4. Execute the instruction.



Machine Cycle

- The cycle during which a machine language instruction is executed by the processor of the computer system is known as the machine cycle.
- Machine cycle contains four phases namely fetching, decoding, executing and storing.

The steps of a machine cycle are:

Fetch – The control unit requests instructions from the main memory that is stored at a memory's location as indicated by the program counter.

Decode – Received instructions are decoded in the instruction register. This involves breaking the operand field into its components based on the instruction's operation code (opcode).

Execute – This involves the instruction's opcode as it specifies the CPU operation required. The program counter indicates the instruction sequence for computer. These instructions are arranged into the instructions register and as each are executed, it increments the program counter so that the next instruction is stored in memory.

Explain in detail about pipelining processing.

Pipelining :

- Pipelining is a process of arrangement of hardware elements of the CPU such that its overall performance is increased.
- Simultaneous execution of more than one instruction takes place in a pipelined processor.

Pipeline Stages

RISC processor has 5 stage instruction pipeline to execute all the instructions in the RISC instruction set. Following are the 5 stages of RISC pipeline with their respective operations:

Stage 1 (Instruction Fetch)
In this stage the CPU reads instructions from the address in the memory whose value is present in the program counter.

Stage 2 (Instruction Decode)
In this stage, instruction is decoded and the register file is accessed to get the values from the registers used in the instruction.

Stage 3 (Instruction Execute)
In this stage, ALU operations are performed.

Stage 4 (Memory Access)
In this stage, memory operands are read and written from/to the memory that is present in the instruction.

Stage 5 (Write Back)
In this stage, computed/fetched value is written back to the register present in the instructions.

Computer Organisation & Architecture

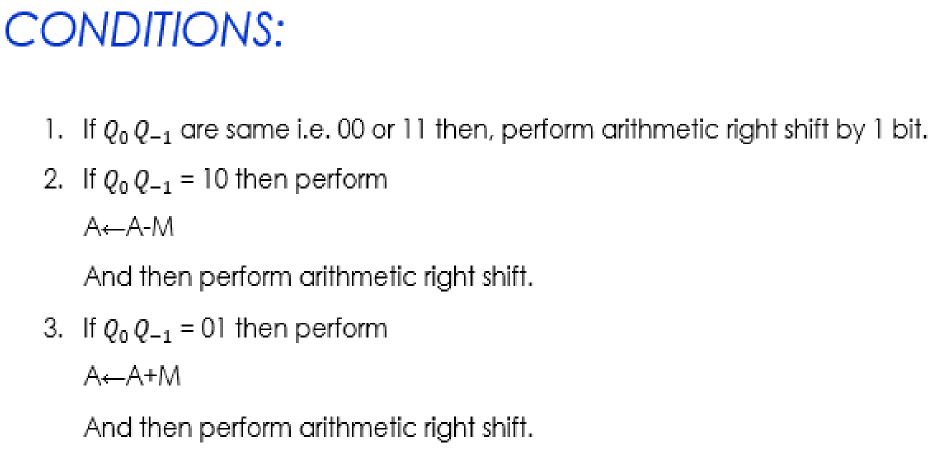
Branch : CSE(3rd Semester)

UNIT – 07 : Arithmetic Algorithm

1. Explain the Booths algorithm for multiplication of two signed numbers with examples.
2. What are the rules to perform 2’s complement subtraction, explain with example ?
3. Using 2’s complement perform the following :
   a. 26 – (-4)
   b. 1 - 7
4. Using booths algorithm solve the following :
   a. (-18) x (-14)
   b. 17 x (-8)

Booth's Multiplication Algorithm

Booth's algorithm is a multiplication algorithm that multiplies two signed binary numbers in 2's compliment notation.



CONDITIONS:

1. If $Q_0\,Q_{-1}$ are same i.e. 00 or 11 then, perform arithmetic right shift by 1 bit.
2. If $Q_0\,Q_{-1}$ = 10 then perform
   A←A-M
   And then perform arithmetic right shift.
3. If $Q_0\,Q_{-1}$ = 01 then perform
   A←A+M
   And then perform arithmetic right shift.

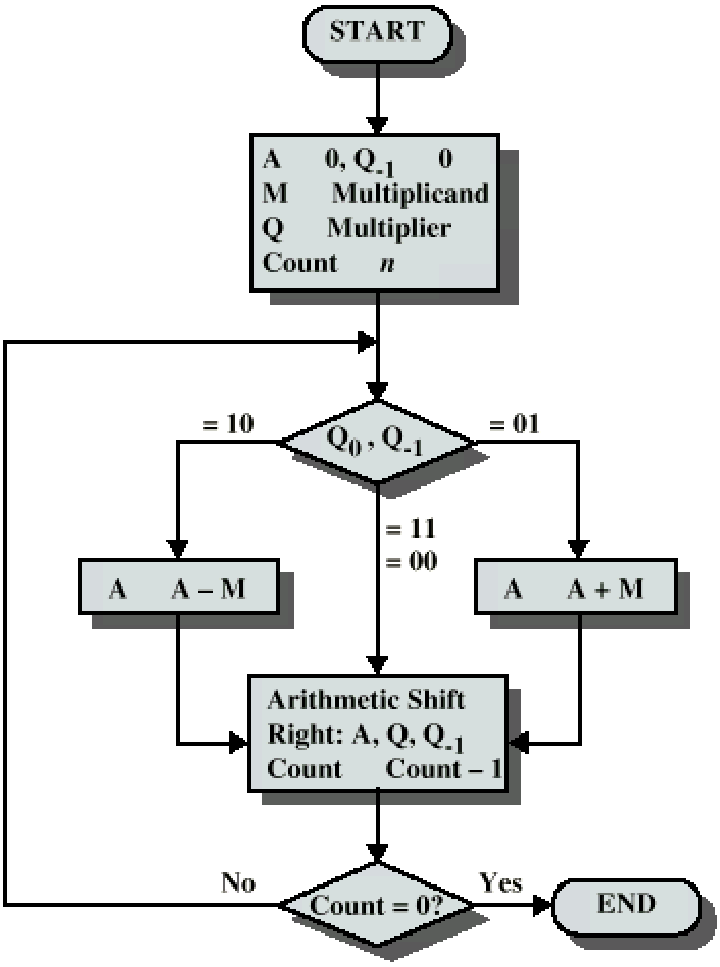


Figure 8.12 Booth's Algorithm for Twos Complement Multiplication

PROCEDURE:

1. Let M is the multiplicand.
2. Let Q is the multiplier.
3. Consider a 1-bit register Q_{-1} and initialize it to 0.
4. Consider a register A and initialize it to 0.

For example:

Consider two numbers 6 and 2 and we have to perform their multiplication by using Booth's algorithm.

Here 6 is multiplicand (M) and 2 is multiplier (Q).

Now write 6 and 2 in binary form.

M = 6 = 0 1 1 0

Q = 2 = 0 0 1 0 (Q_3, Q_2, Q_1, Q_0)

Booth's algorithm calculates the product in n steps where n is the number of bits used to represent the numbers.

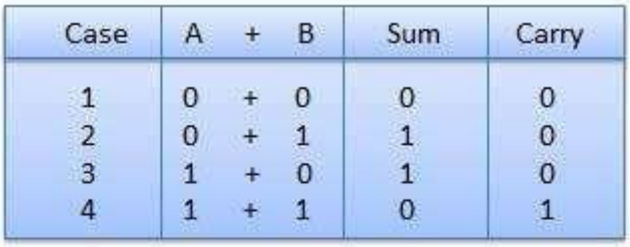
INITIALISE	A	B	Q_{-1}	OPERATIONS
	0 0 0 0 ↓↘↘↘↘	0 0 1 0 ↘↘↘↘	0	
Step 1.	0 0 0 0	0 0 0 1 ↓	0 ↓	Arithmetic right shift
Step 2.	1 0 1 0 ↓↘↘↘↘ 1 1 0 1	0 0 0 1 ↘↘↘↘ 0 0 0 0	0 1	A←A-M Then shift
Step 3.	0 0 1 1 ↓↘↘↘↘ 0 0 0 1 ↓↘↘↘↘	0 0 0 0 ↘↘↘↘ 1 0 0 0 ↘↘↘↘	1 0	A←A+M Then shift
Step 4.	0 0 0 0	1 1 0 0 In binary, 12 = 1100 Hence 3*2 = 12	0	Arithmetic right shift

Binary Arithmetic

Binary arithmetic is essential part of all the digital computers and many other digital system.

Binary Addition

It is a key for binary subtraction, multiplication, division. There are four rules of binary addition.



Case	A	+	B	Sum	Carry
1	0	+	0	0	0
2	0	+	1	1	0
3	1	+	0	1	0
4	1	+	1	0	1

In fourth case, a binary addition is creating a sum of (1 + 1 = 10) i.e. 0 is written in the given column and a carry of 1 over to the next column.

Example − Addition

0011010 + 001100 = 00100110

1 1 carry

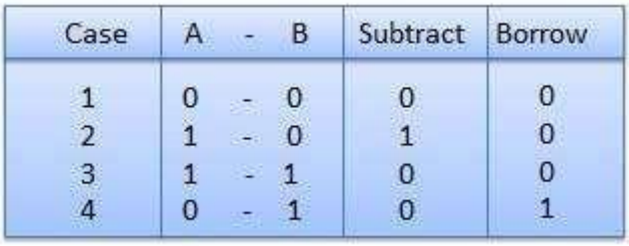
0 0 1 1 0 1 0 = 26_{10}

+0 0 0 1 1 0 0 = 12_{10}

0 1 0 0 1 1 0 = 38_{10}

Binary Subtraction

Subtraction and Borrow, these two words will be used very frequently for the binary subtraction. There are four rules of binary subtraction.



Case	A	-	B	Subtract	Borrow
1	0	-	0	0	0
2	1	-	0	1	0
3	1	-	1	0	0
4	0	-	1	0	1

Example − Subtraction

0011010 - 001100 = 00001110

1 1 borrow

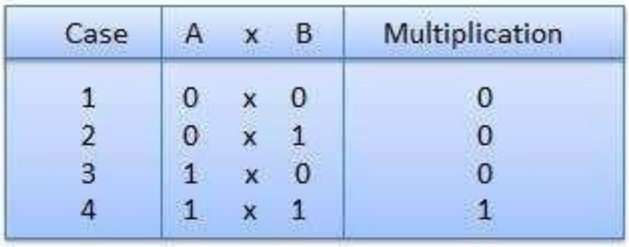
0 0 ~~1 1~~ 0 1 0 = 26_{10}

-0 0 0 1 1 0 0 = 12_{10}

0 0 0 1 1 1 0 = 14_{10}

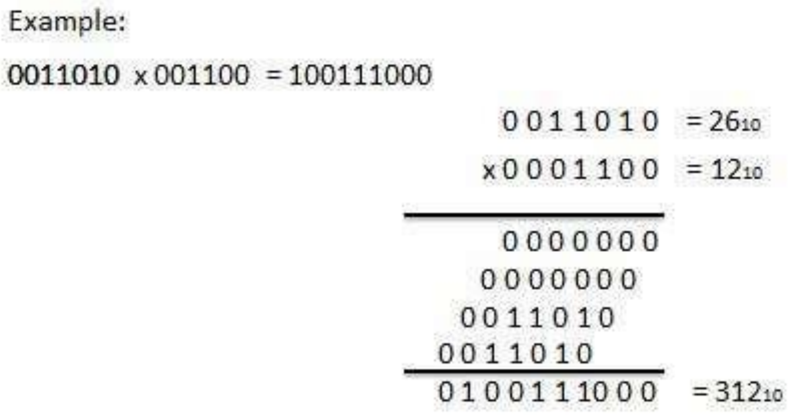
Binary Multiplication

- Binary multiplication is similar to decimal multiplication.
- It is simpler than decimal multiplication because only 0s and 1s are involved.
- There are four rules of binary multiplication.



Case	A x B	Multiplication
1	0 x 0	0
2	0 x 1	0
3	1 x 0	0
4	1 x 1	1

Example – Multiplication



Example:

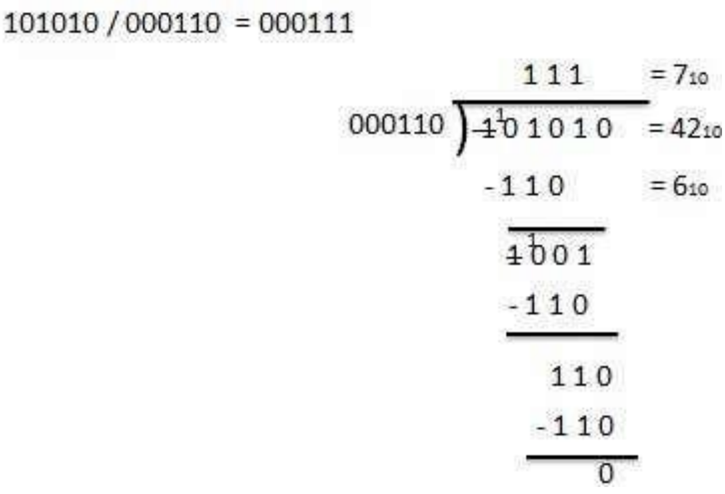
0011010 x 001100 = 100111000

```
          0 0 1 1 0 1 0   = 26₁₀
        x 0 0 0 1 1 0 0   = 12₁₀
      -------------------
          0 0 0 0 0 0 0
        0 0 0 0 0 0 0
      0 0 1 1 0 1 0
    0 0 1 1 0 1 0
      -------------------
    0 1 0 0 1 1 1 0 0 0    = 312₁₀
```

Binary Division

- Binary division is similar to decimal division.
- It is called as the long division procedure.

Example – Division



101010 / 000110 = 000111

```
                 1 1 1      = 7₁₀
000110 ) 1 0 1 0 1 0        = 42₁₀
        -1 1 0              = 6₁₀
        -------
          1 0 0 1
         -1 1 0
         -------
            1 1 0
           -1 1 0
           -------
                0
```

Complement Arithmetic

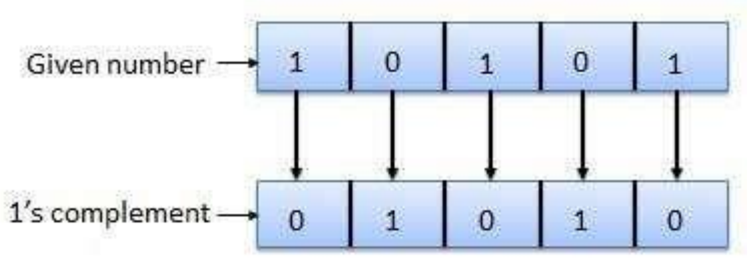
- Complements are used in the digital computers in order to simplify the subtraction operation and for the logical manipulations.
- For each radix-r system (radix r represents base of number system) there are two types of complements.

Binary system complements

As the binary system has base r = 2. So the two types of complements for the binary system are 2's complement and 1's complement.

1's complement

The 1's complement of a number is found by changing all 1's to 0's and all 0's to 1's. This is called as taking complement or 1's complement. Example of 1's Complement is as follows.

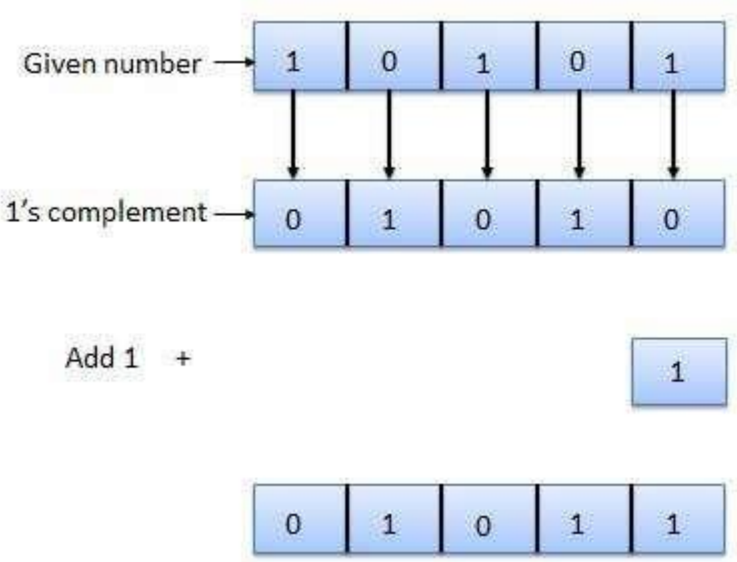


2's complement

The 2's complement of binary number is obtained by adding 1 to the Least Significant Bit (LSB) of 1's complement of the number.

2's complement = 1's complement + 1

Example of 2's Complement is as follows.



Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 08 : I/O Organisation :

1. Discuss about memory mapped I/O.
2. What do you mean by DMA ?
3. What do you mean by DMA controller ?
4. Discuss about associative memory in brief.
5. What do you mean by locality of reference ?

Discuss about memory mapped I/O.

- In this kind of interfacing, we assign a memory address that can be used in the same manner as we use a normal memory location.
- IO devices are accessed like any other memory location.
- They are assigned with 16-bit address values.
- The instruction used are LDA and STA, etc.
- Cycles involved during operation are Memory Read, Memory Write.
- Arithmetic and logical operations are performed directly on the data in the case of Memory Mapped IO.

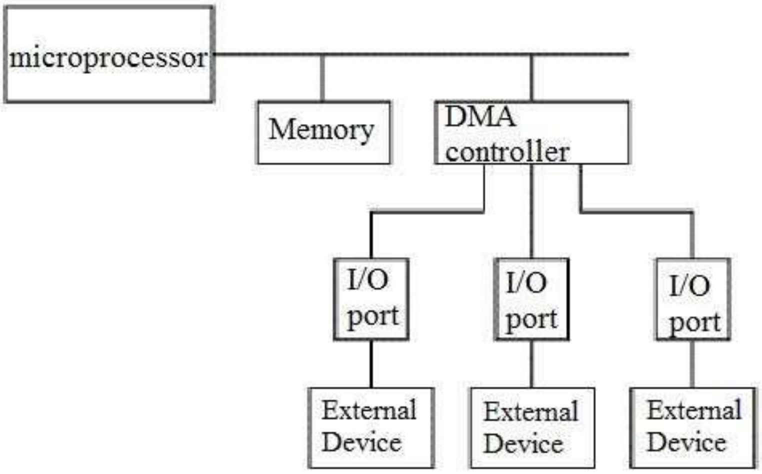
What do you mean by DMA ?

- The term DMA stands for direct memory access.
- Direct memory access(DMA) is a method that allows an input/output device to send or receive data directly to or from memory.
- This method to to send or receive data directly to or from memory increase the memory operations speed.
- These process is managed by a chip known as DMAC.

What is a DMA Controller?

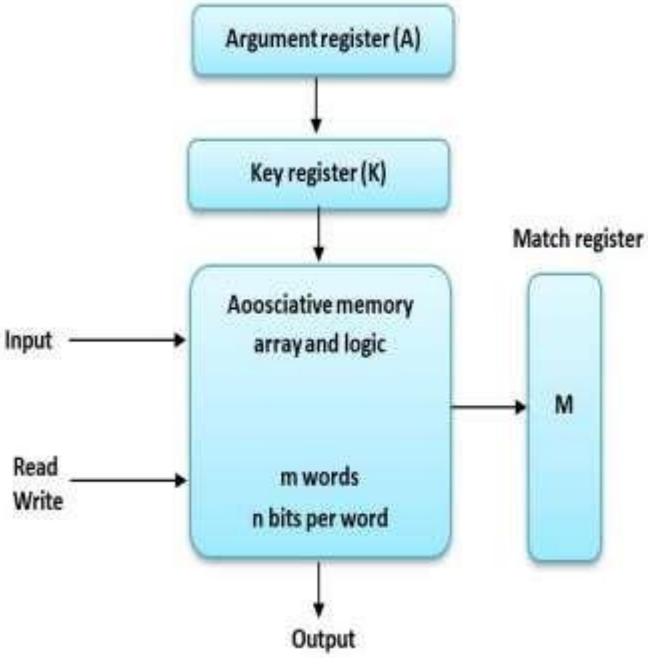
- The term DMA stands for direct memory access.
- The hardware device used for direct memory access is called the DMA controller.
- DMA controller is a control unit, part of I/O device's interface circuit, which can transfer blocks of data between I/O devices and main memory with minimal intervention from the processor.
- DMA controller provides an interface between the bus and the input-output devices.

DMA Controller Diagram in Computer Architecture



Associative memory

- When data is accessed by data content rather than data address, then the memory is called associative memory or content addressable memory(CAM).
- Data is stored at the very first empty location found in memory.
- In Associative memory when data is stored at a particular location but no address is stored along with it.
- When the stored data need to searched then only the key(i.e, data or part of data) is provided.



Locality of reference

- Locality of reference refers to a phenomenon in which a computer program tends to access same set of memory locations for a particular time period.
- In other words, Locality of Reference refers to the tendency of the computer program to access instructions whose addresses are near one another.
- The property of locality of reference is mainly shown by loops and subroutine calls in a program.
- It is also known as "locality in space" and "spatial locality,".

Computer Organisation & Architecture

Branch : CSE(3rd Semester)

UNIT – 09 : Memory Organisation

1. Differentiate between SRAM and DRAM.
2. What do you mean by logical address and physical address space ?
3. Compare paging and segmentation for virtual memory ?
4. What do you mean by cache memory ?
5. What do you mean virtual memeory ?
6. Concider the string 1,3,2,4,2,1,5,1,3,2,6,7,5,4,3,2,4,2,3,1,4. Find the page faults using FIFO and LRU.

Differentiate between SRAM and DRAM.

SRAM	DRAM
It stand for Static Random Access Memory.	It stand for Dynamic Random Access Memory.
Transistors are used to store information in SRAM.	Capacitors are used to store data in DRAM.
Capacitors are not used hence no refreshing is required.	To store information for a longer time, contents of the capacitor needs to be refreshed periodically.
SRAM is faster as compared to DRAM.	DRAM provides slow access speeds.
These are expensive.	These are cheaper.
These are used in cache memories.	These are used in main memories.

Physical and Logical Address Space

Physical Address Space

- Physical address space in a system can be defined as the size of the main memory.
- It is really important to compare the process size with the physical address space.
- The process size must be less than the physical address space.
- Physical Address Space = Size of the Main Memory

Logical Address Space

- Logical address space can be defined as the size of the process.
- The size of the process should be less enough so that it can reside in the main memory.

What are the difference between paging and segmentation.

Paging	Segmentation
In paging, program is divided into fixed or mounted size called pages.	In segmentation, program is divided into variable size called frames.
For paging operating system is accountable.	For segmentation compiler is accountable.
It is faster in the comparison of segmentation.	Segmentation is slow.
Paging could result in internal fragmentation.	Segmentation could result in external fragmentation.
In paging, logical address is split into page number and page offset.	Here, logical address is split into section number and section offset.
In paging, operating system must maintain a free frame list.	In segmentation, operating system maintain a list of holes in main memory.
Paging is invisible to the user.	Segmentation is visible to the user.

What do you mean by cache memory ?

CacheMemory

- Cache memory is a very high speed semiconductor memory.
- It acts as a buffer between the CPU and main memory.
- It is used to hold those parts of data and program which are most frequently used by CPU.
- The parts of data and programs are transferred from disk to cache memory by operating system, from where CPU can access them.

Advantages	Disadvantages
▪ Cache memory is faster than main memory. ▪ It consumes less access time as compared to main memory. ▪ It stores the program that can be executed within a short period of time. ▪ It stores data for temporary use.	▪ Cache memory has limited capacity. ▪ It is very expensive.

What is Virtual Memory?

- Virtual Memory is a storage mechanism which offers user an illusion of having a very big main memory.
- It stores in the form of pages while their execution and only the required pages of processes are loaded into the main memory.
- This technique is useful as large virtual memory is provided for user programs when a very small physical memory is there.
- Main objective of needing is virtual memory is to increase the storage space of running memory, without adding any external memory such as RAM.
- Virtual Memory is managed with two techniques such as Paging and Segmentation.
- Entire memory operations of computer are managed by memory management unit (MMU).