Course Contents

Unit-01: Introduction to Microprocessor

Unit-02: Intel 8085

Unit-03: Microoperations

Unit-04: Control Unit and Central Processing Unit

Unit-05: Fixed point Computer Arithmetic

Unit-06: Input and Output Organization

Unit-07: Memory Organization

Unit-08: Pipelining

Course Contents

Unit-01: Introduction to Microprocessor

- Components of a Microprocessor: Registers, ALU, Control and Timing, System Buses,
- Microprocessor Systems with Bus Organization,
- Introduction to SAP1 and SAP2

Course Contents

Unit 1 Introduction to Microprocessor 6 Hours

1. Definition of Microprocessor Components : Registers, ALU, Control and Timing, System Buses (Address, Data, Control), Microprocessor System with Bus Organization 1.5 Hours
2. SAP-1 Architecture: Block Diagram, and Function of each Block SAP-1 Instructions :LDA, ADD, SUB, OUT, HLT Fetch and Execution Cycle of SAP-1 Instructions with Timing Diagram • Fetch Cycle: Address State, Increment State, Memory State • Execution Cycle of LDA only 3 Hour
3. SAP-2 Architecture: Block Diagram and Functions of each Block, Architectural Differences with SAP-1
   - ➢ Bidirectional Registers
   - ➢ Flags 1.5 Hour

Course Contents

Definition of microprocessor and its application

- Basic Concepts of Microprocessors

Differences between:

- ➢ Microcomputer – a computer with a microprocessor as its CPU. Includes memory, I/O etc.
- ➢ Microprocessor – silicon chip which includes ALU, register circuits & control circuits
- ➢ Microcontroller – silicon chip which includes microprocessor, memory & I/O in a single package.

Course Contents

What is a Microprocessor?

The word comes from the combination micro and processor.

- Processor means a device that processes whatever. In this context processor means a device that processes numbers, specifically binary numbers, 0's and 1's.
- To process means to manipulate. It is a general term that describes all manipulation. Again in this content, it means to perform certain operations on the numbers that depend on the microprocessor's design.

Course Contents

What about micro?

Micro is a new addition.

- In the late 1960's, processors were built using discrete elements.
  - These devices performed the required operation, but were too large and too slow.
- In the early 1970's the microchip was invented. All of the components that made up the processor were now placed on a single piece of silicon. The size became several thousand times smaller and the speed became several hundred times faster. The "Micro"Processor was born.

Course Contents

Definition of the Microprocessor.

The microprocessor is a programmable device that takes in numbers, performs on them arithmetic or logical operations according to the program stored in memory and then produces other numbers as a result.

Numbers: The microprocessor only understands binary numbers. A binary digit is called a bit (which comes from binary digit). The microprocessor recognizes and processes a group of bits together. This group of bits is called a "word". The number of bits in a Microprocessor's word, is a measure of its "abilities".

Course Contents

Lets expand each of the underlined words:

Instructions: Each microprocessor is designed to execute a specific group of operations. This group of operations is called an instruction set. This instruction set defines what the microprocessor can and cannot do.

Takes in: The data that the microprocessor manipulates must come from somewhere.

- It comes from what is called "input devices".
- These are devices that bring data into the system from the outside world.
- These represent devices such as a keyboard, a mouse, switches, and the like.

Course Contents

Definition of the Microprocessor.

Words, Bytes, etc.: The earliest microprocessor (the Intel 8088 and Motorola's 6800) recognized 8-bit words.

They processed information 8-bits at a time. That's why they are called "8-bit processors". They can handle large numbers, but in order to process these numbers, they broke them into 8-bit pieces and processed each group of 8-bits separately.

Later microprocessors (8086 and 68000) were designed with 16-bit words.

A group of 8-bits were referred to as a "half-word" or "byte". A group of 4 bits is called a "nibble".

Also, 32 bit groups were given the name "long word".

Today, all processors manipulate at least 32 bits at a time and there exists microprocessors that can process 64, 80, 128 bits

Course Contents

Definition of the Microprocessor.

Arithmetic and Logic Operations: Every microprocessor has arithmetic operations such as add and subtract as part of its instruction set.

- Most microprocessors will have operations such as multiply and divide. Some of the newer ones will have complex operations such as square root.
- In addition, microprocessors have logic operations as well. Such as AND, OR, XOR, shift left, shift right, etc.
- Again, the number and types of operations define the microprocessor's instruction set and depends on the specific microprocessor.

Course Contents

Definition of the Microprocessor.

Stored in memory: First, what is memory?

- Memory is the location where information is kept while not in current use.
- Memory is a collection of storage devices. Usually, each storage device holds one bit. Also, in most kinds of memory, these storage devices are grouped into groups of 8. These 8 storage locations can only be accessed together. So, one can only read or write in terms of bytes to and form memory.
- Memory is usually measured by the number of bytes it can hold. It is measured in Kilos, Megas and lately Gigas. A Kilo in computer language is 210 =1024. So, a KB (KiloByte) is 1024 bytes. Mega is 1024 Kilos and Giga is 1024 Mega.

Course Contents

Definition of the Microprocessor.

Stored in memory: First, what is memory?

- When a program is entered into a computer, it is stored in memory. Then as the microprocessor starts to execute the instructions, it brings the instructions from memory one at a time.
- Memory is also used to hold the data.
- The microprocessor reads (brings in) the data from memory when it needs it and writes (stores) the results into memory when it is done.

Course Contents

Definition of the Microprocessor.

Produces: For the user to see the result of the execution of the program, the results must be presented in a human readable form.

- The results must be presented on an output device.
- This can be the monitor, a paper from the printer, a simple LED or many other forms.

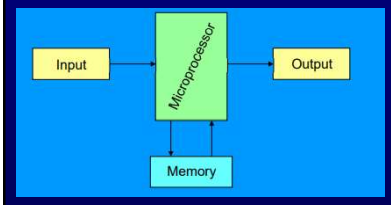
Introduction to Microprocessor

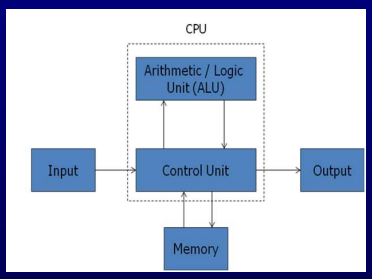
Components of a Microprocessor: Registers, ALU, Control and Timing, System Buses,

Course Contents

Definition of the Microprocessor.

Block diagram to represent a microprocessor-based system:





Inside The Microprocessor

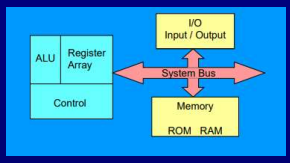
Internally, the microprocessor is made up of 3 main units.

- The Arithmetic/Logic Unit (ALU)
- The Control Unit.
- An array of registers for holding data while it is being manipulated.

Course Contents

Definition of the Microprocessor.

Organization of a microprocessor based system:



Memory: Memory stores information such as instructions and data in binary format (0 and 1). It provides this information to the microprocessor whenever it is needed.

Usually, there is a memory "sub-system" in a microprocessor-based system. This sub-system includes:

- The registers inside the microprocessor
- Read Only Memory (ROM) - used to store information that does not change.
- Random Access Memory (RAM) (also known as Read/Write Memory) - used to store information supplied by the user. Such as programs and data.

Course Contents

Definition of the Microprocessor.

Organization of a microprocessor based system:

Memory:

To execute a program:

- the user enters its instructions in binary format into the memory.
- The microprocessor then reads these instructions and whatever data is needed from memory, executes the instructions and places the results either in memory or produces it on an output device.

Course Contents

Definition of the Microprocessor.

Organization of a microprocessor based system:

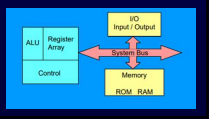
The three cycle instruction execution model

- To execute a program, the microprocessor "reads" each instruction from memory, "interprets" it, then "executes" it.
- To use the right names for the cycles: – The microprocessor fetches each instruction, decodes it, then executes it.
- This sequence is continued until all instructions are performed.

Microprocessor system with bus organization

Microprocessor system with bus organization:

- Bus is a common channel through which bits from any sources can be transferred to the destination.
- A typical digital computer has many registers and paths must be provided to transfer instructions from one register to another.
- The number of wires will be excessive if separate lines are used between each register and all other registers in the system.
- A more efficient scheme for transferring information between registers in a multiple register configuration is a common bus system.
- A bus structure consists of a set of common lines, one for each bit of a register, through which binary information is transferred one at a time.
- Control signals determine which register is selected by the bus during each particular register transfer.



Microprocessor system with bus organization

Microprocessor system with bus organization:

- The data lines provide a path for moving data between system modules. These lines are collectively called data bus.
- The address lines are used to designate the source/destination of data on data bus.
- The control lines are used to control the access to and the use of the data and address lines. Because data and address lines are shared by all components, there must be a means of controlling their use. Control signals transmit both command and timing signals indicate the validity of data and address information. Command signals specify operations to be performed. Control lines include memory read/write, I/O read/write, bus request/grant, clock, reset, interrupt request/acknowledge etc.

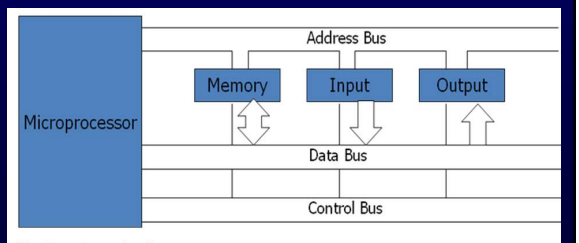


Fig: Bus Organization

Microprocessor architecture and operation

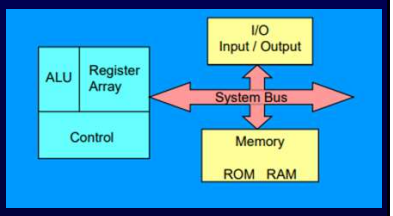
Microprocessor architecture and operation:

What is a Microprocessor Architecture?

- The microprocessor is a programmable logic device, designed with registers, flip-flops. The microprocessor has a set of instructions designed internally, to manipulate data and communicate with peripherals. This process of data manipulation and communication is determined by the logic design of the microprocessor, called the architecture.

The function of microprocessor can be classified in 3 general categories:

1. Microprocessor-initiated operations
2. Internal data operations
3. Peripheral (or externally) initiated operations.



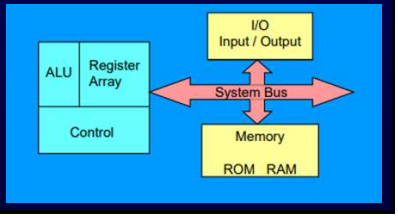
3. Microprocessor architecture and operation

Microprocessor architecture and operation:

1. Microprocessor-initiated operations and 8085/8080A Bus Organization

The MPU performs primarily four operations.

1. Memory Read: Reads data from memory.
2. Memory Write: Writes data into memory.
3. I/O read: Accepts data from input devices.
4. I/O Write: Sends data to output devices.



3. Microprocessor architecture and operation

Microprocessor architecture and operation:

All these operations are part of the communication process between the MPU and peripheral devices (including memory). To communicate with a I/O devices or a memory, the MPU needs to perform the following steps:

Step 1 : Identify the peripheral or the memory location (with its address).

Step 2 : Transfer data

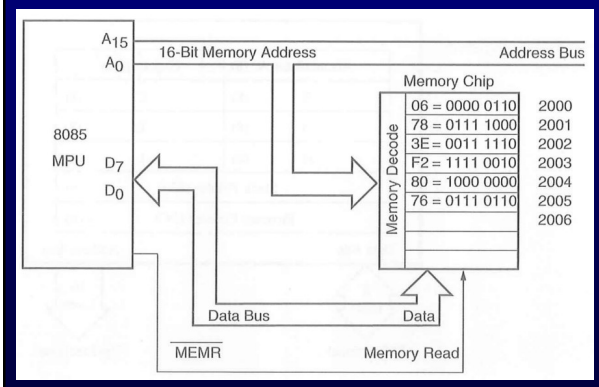
Step 3 : Provide timing or synchronization signals.

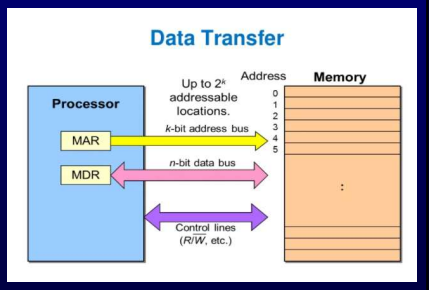
The 8085/8080A MPU performs these functions using three sets of communication lines called buses: The address bus, the data bus, and the control bus.

3. Microprocessor architecture and operation

Microprocessor architecture and operation:

To communication with a memory, for example to read instruction from memory location:





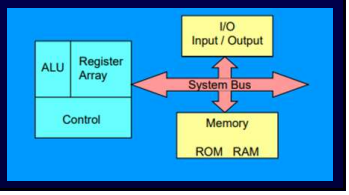
3. Microprocessor architecture and operation

Microprocessor architecture and operation:

2. Internal Data Operations

The internal architecture of the microprocessor determines how and what operations can be performed with the data. These operations are

- Store data.
- Perform arithmetic and logical operations.
- Test for conditions.
- Sequence the execution of instructions.
- Store data temporarily during execution in the defined R/W memory locations called the stack.



3. Microprocessor architecture and operation

Microprocessor architecture and operation:

2. Peripheral or Externally Initiated Operations:

External devices (or signals) can initiate the following operation for which individual pins on MP chip are assigned: Reset, Interrupt, Ready, Hold.

A) Reset: when reset is activated all internal operations are suspended and the program counter is cleared.

B) Interrupt: the MP can be interrupted from normal execution and asked to execute other instructions called "service routine" (emergency), MP resumes its operation after that.

C) Ready: 8085 has pin called ready, if the signal is low MP enters into wait state, this signal used to synchronize slower peripherals with MP.

D) Hold: when hold pin activated by external signal MP relinquishes control buses and allows the external peripheral to use the. For example: Hold signal is used in direct memory access data transfer.

Course Contents

SAP-1 Architecture: Block Diagram, and Function of each Block

SAP-1 Instructions :LDA, ADD, SUB, OUT, HLT

Fetch and Execution Cycle of SAP-1 Instructions with Timing Diagram

Fetch Cycle: Address State, Increment State, Memory State

Execution Cycle of LDA only

SAP-2 Architecture: Block Diagram and Functions of each Block, Architectural Differences with SAP-1

- ➢ Bidirectional Registers
- ➢ Flags

Introduction to Microprocessor

SAP-1 Architecture:

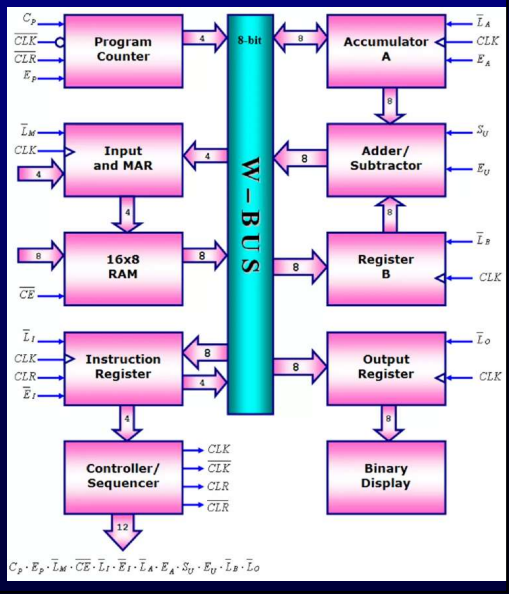
- SAP is **Simple As Possible** Computer. The SAP-1 computer is specially designed for the academic purpose and nothing has to do with the commercial use. The architecture is 8 bits and comprises of 16 X 8 memory i.e., 16 memory location with 8 bits in each location, therefore, need 4 address lines which either comes from the PC (Program Counter which may be called instruction pointer) during computer run phase. All instructions (5 only) stores in this memory. It means SAP can't store program having more than 16 instructions.
- SAP can only perform addition and subtraction and no logical operation. These arithmetic operations are performed by an adder/subtractor unit.
- There is one general purpose register (**B register**) used to hold one operand of the arithmetic operation while another is kept by the accumulator register of the SAP-1. In addition, there are 8 LEDs, work as output unit and connected with the 8-bit output register.
- All timely moment of data or activities are performed by the controller/sequencer part of SAP-1.

Introduction to Microprocessor

SAP-1 Architecture: Block Diagram, and Function of each Block:

The features in SAP-1 computer are

- W bus - A single 8-bit bus for address and data transfer.
- 16 Bytes memory (RAM)
- Registers are accumulator(A) and B-register each of 8 bits.
- Program counter – initializes from 00H(0d) to 0FH (15d) during program execution.
- Memory Address Register (MAR) to store memory addresses.
- Adder/Subtractor for addition and subtraction instructions.
- A Control Unit
- A Simple Output (BINARY DISPLAY).
- 6 machine cycles (T_1 to T_6) reserved for each instruction



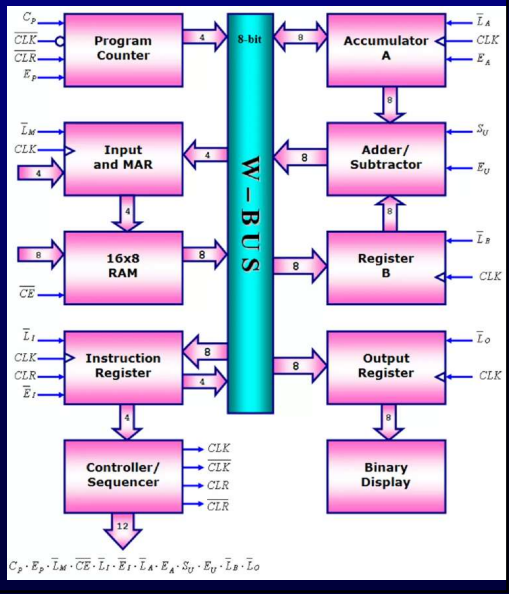
Introduction to Microprocessor

SAP-1 Architecture: Block Diagram, and Function of each Block:

Architecture

The architecture of SAP1 consists of following components:

1. Program counter
2. Input MAR
3. The RAM
4. Instruction Register
5. Controller Sequencer
6. Accumulator
7. The Adder/subtractor
8. B Register
9. Output Register
10. Binary Display



Introduction to Microprocessor

SAP-1 Architecture: Block Diagram, and Function of each Block:

1. Program Counter (PC)

- It counts from 0000 to 1111 and it signals the memory address of next instruction to be fetched and executed

2. Input and MAR (MAR)

- During a computer run, the address in PC is latched into Memory Address Register (MAR).

3. RAM

- The program code to be executed and data for SAP1 computer is stored here.
- During a computer run, the RAM receives 4-bit addresses from MAR and a read operation is performed. Hence, the instruction or data word stored in RAM is placed on the W bus for use by some other part of the computer.
- It is asynchronous RAM, which means that the output data is available as soon as valid address and control signal are applied.

Introduction to Microprocessor

SAP-1 Architecture: Block Diagram, and Function of each Block:

4. Instruction Register (IR)

- IR contains the instruction (composed of OPCODE+ADDRESS) to be executed by SAP1 computer.

5. Controller- Sequencer

- It generates the control signals for each block so that actions occur in desired sequence.
- CLK signal is used to synchronize the overall operation of the SAP1 computer.
- A 12-bit word comes out of the Controller-Sequencer block. This control word determines how the registers will react to the next positive CLK edge.

6. Accumulator

- It is a 8-bit buffer register that stores intermediate results during a computer run.
- It is always one of the operands of ADD, SUB and OUT instructions.

Introduction to Microprocessor

SAP-1 Architecture: Block Diagram, and Function of each Block:

7. Adder-Subtractor

- It is a 2's complement adder-subtractor
- This module is asynchronous (un-clocked), which means that its contents can change as soon as the input words change

8. B Register

- It is 8 bit buffer register which is primarily used to hold the other operand (one operand is always accumulator) of mathematical operations.

9. Output Register

- These registers hold the output of OUT instruction.

10. Binary Display

- It is a row of eight LEDs to show the contents of output register.

Introduction to Microprocessor

SAP-1 Instructions :LDA, ADD, SUB, OUT, HLT:

SAP-1 instruction set consists of following instructions

Mnemonic	Operation	OPCODE
LDA	Load addressed memory contents into accumulator	0000
ADD	Add addressed memory contents to accumulator	0001
SUB	Subtract addressed memory contents from accumulator	0010
OUT	Load accumulator data into output register	1110
HLT	Stop processing	1111

The instruction format of SAP-1 Computer is (XXXX) (XXXX)

The first four bits make the op-code while the last four bits make the operand (address).

Example LDA 8H

Example: if 0000 1000 is stored at memory location 0000 of RAM then SAP1 computer interprets it as "The data from 08 address will load in the accumulator".

Introduction to Microprocessor

SAP-1 Instructions :LDA, ADD, SUB, OUT, HLT:

Mnemonic	Operation	OPCODE
LDA	Load addressed memory contents into accumulator	0000
ADD	Add addressed memory contents to accumulator	0001
SUB	Subtract addressed memory contents from accumulator	0010
OUT	Load accumulator data into output register	1110
HLT	Stop processing	1111

Address	Content
0H	LDA 9H
1H	ADD AH
2H	ADD BH
3H	SUB CH
4H	OUT
5H	HLT
6H	
7H	
8H	
9H	8
AH	9
BH	7
CH	4
DH	
EH	
FH	

What will be in the output if values 8, 9, 7 and 4 are stored at memory locations 9H, AH, BH and CH respectively?

LDA 9H
ADD AH
ADD BH
SUB CH
OUT
HLT

IR (8-bit)	
OPCODE	ADDRESS

Introduction to Microprocessor

Problems:

1. What will be in the output if values 8, 9, 7 and 4 are stored at memory locations 9H, AH, BH and CH respectively?

   LDA 9H
   ADD AH
   ADD BH
   SUB CH
   OUT
   HLT

2. Write a SAP-1 program using mnemonics that will display the result of 5 + 4 – 6, Use addresses D_H, E_H. and F_H for the data.

3. Convert the assembly language of above problem into SAP-1 machine language. Show the answer in binary form and in hexadecimal form.

4. Write an assembly-language program that performs this operation: 8 + 4 - 3 + 5 – 2, Use addresses B_H to F_H for the data.

Introduction to Microprocessor

Machine cycle and Instruction cycle:

SAP1 has six T-states (three fetch and three execute cycles) reserved for each instruction. Not all instructions require all the six T-states for execution. The unused T- state is marked as No Operation (NOP) cycle. Each T-state is called a machine cycle for SAP1. A ring counter is used to generate a T-state at every falling edge of clock pulse. The ring counter output is reset after the 6th T-state.

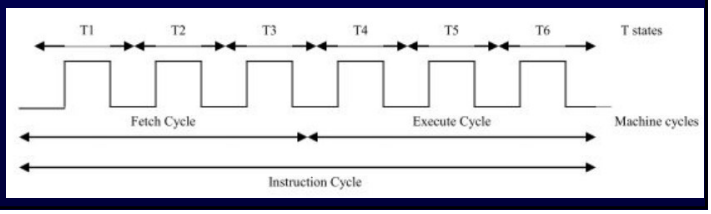
FETCH CYCLE – T1, T2, T3 machine cycle

EXECUTE CYCLE - T4, T5, T6 machine cycle

Introduction to Microprocessor

Machine cycle and Instruction cycle:

- Complete code includes opcode and operand
- One instruction is executed in one instruction cycle
- Instruction cycle may consist of many machine cycles
- For SAP-1, Instruction cycle = Machine cycle
- Instruction cycle = Fetch cycle + Execution cycle
- Fetch cycle is generally same for all instructions
- Complete code includes opcode and operand
  Like LDA 04H (0000 0100)
- One instruction is executed in one instruction cycle



Introduction to Microprocessor

Fetch and Execution Cycle of SAP-1 instructions:

- SAP-1 instruction cycle: 3 clock cycles to fetch and decode phase, 3 clock cycles to execute
- The first three states are:
  1. address
  2. increment
  3. memory
- Controller has a 6-bit ring counter which continuously cycles from 000001 up to 100000 then resets (must be set to 000001 when we initialize the computer)
- Ring counter is clocked on clock high-to-low transition, most of the other circuits in the computer on clock low-to-high transition

Introduction to Microprocessor

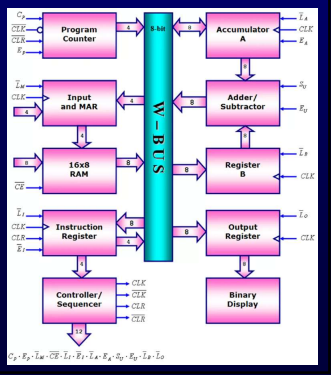
Fetch Cycle: LDA 9H

- Address state: enable PC to bus three-state output, MAR load line
- Increment state: enable PC increment (and perhaps wait for memory access time)
- Memory state: enable memory CE, IR load line
- IR is loaded on the low-to-high clock transition, so stabilizes before state 4 is entered

t1: MAR ← PC
t2: PC ← PC +1
t3: IR ← RAM

Mnemonic	Operation	OPCODE
LDA	Load addressed memory contents into accumulator	0000
ADD	Add addressed memory contents to accumulator	0001
SUB	Subtract addressed memory contents from accumulator	0010
OUT	Load accumulator data into output register	1110
HLT	Stop processing	1111

Address	Content
0H	LDA 9H
1H	
2H	
3H	
4H	
5H	
6H	



Introduction to Microprocessor

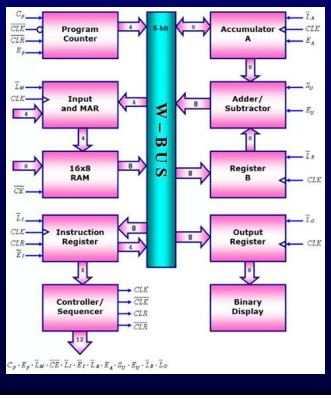
Execution Cycle – LDA 9H:

- state 4: enable IR to bus three-state output, MAR load line
- state 5: enable memory CE, accumulator load line
- state 6: enable nothing

  t4: MAR ← (IR (Address of operand))

  t5: Accumulator ← RAM

  t6: nothing



Address	Content
0H	LDA 9H
1H	
2H	
3H	
4H	
5H	
6H	
7H	
8H	
9H	
AH	
BH	
CH	
DH	
EH	
FH	

Introduction to Microprocessor

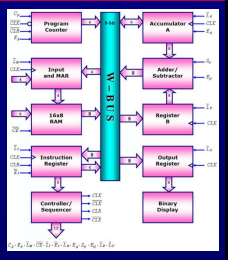
Execution Cycle – LDA 9H:

- state 4: enable IR to bus three-state output, MAR load line
- state 5: enable memory CE, accumulator load line
- state 6: enable nothing

  t4: MAR ← (IR (Address of operand))

  t5: Accumulator ← RAM

  t6: nothing



LDA 9H process:

IR register = 0 0 0 0 1 0 0 1 after T_3 States.

During T4 States, the instruction field 0000 goes to the controller sequencer where it is decoded and address field 1001 is loaded into the MAR

$\overline{Ei}$ and $\overline{Lm}$ are active while all other control bits are inactive.

During T5 States, $\overline{CE}$ and $\overline{La}$ go low and the addressed data word in the RAM will be loaded into the accumulator on the positive clock edge.

T6 is non operation state for LDA 9H

Introduction to Microprocessor

Execution Cycle – ADD B_H:

- state 4: enable IR to bus three-state output, MAR load line
- state 5: enable memory CE, register B load line
- state 6: enable add, ALU to bus three-state output, accumulator load line

t4: MAR ← IR (Address:1011= B is the address of operand
t5: B ← RAM (Content from memory address B is transferred to B-Register)
t6: Accumulator ← Accumulator + B

Introduction to Microprocessor

Example: cycle-by-cycle operation of the instruction "add":

PART 1: FETCH

t1: MAR ← PC
t2: PC ← PC +1
t3: IR ← RAM

Step 1 (T1)- (address state): The first thing we need is an address to fetch the instruction from. That address is stored in the PC, so we route the value in PC to the MAR, getting ready to address the memory. mar <- pc.

Step 2 (T2)- (Increment state): Since the PC should always point at the next instruction to execute, we need to increment the PC at this point, so we add 1 to it. pc <- pc + 1.

Step 3 (T3)- (memory state): We now use the MAR to address the memory, putting the resulting instruction in the instruction register (IR) where we will use its various pieces.

Introduction to Microprocessor

Example: cycle-by-cycle operation of the instruction "add":

PART 2: EXECUTE

t4: MAR ← (IR (Address of B))
t5: B ← RAM
t6: Accumulator ← Accumulator + B

Step 4 (T4): The instruction tells us we are adding, so we need to get the data from the memory at the address stored in the instruction, and put that data into the B register. Before we do that, we need to put the address from the instruction into the MAR so the memory has access to it. mar <- ir(3:0)

Step 5 (T5): Now that the memory has access to the address, we use that address to get the data from memory, and put it into B. B <- M(mar)

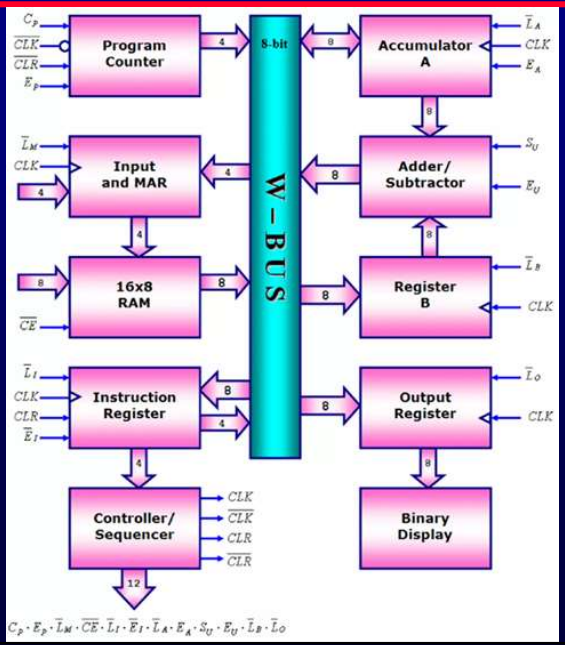
Step 6 (T6): Now in this state, add what's in B to what's in A, and put the result back in A. A <- A+B.

Introduction to Microprocessor

Some additional notes on the SAP-1: control point descriptions

- C_P: Count PC. Increment the value of PC by 1: pc <- pc + 1
- E_P: Enable PC. Put value of PC onto the bus: bus (3:0) <- pc
- L_I: Load IR. Instruction Register gets value on the bus: ir <- bus
- E_I: Enable IR. Put value of lower 4 bits of IR (the address) onto the bus: bus(4:0) <- ir(3:0)
- L_M: Load MAR. Memory address register gets value on the low 4 bits of the bus: mar <- bus(3:0)
- C_E: Chip Enable. Memory chip puts word stored at the address onto the bus: bus <- M(mar)
- L_A: Load A. Accumulator Register gets value on the bus: A <- bus
- E_A: Enable A. Put value of A onto the bus: bus <- A
- S_U: Subtract Unit. Tell the adder/subtractor unit to subtract instead of Add.
- E_U: Enable Unit. Put value of the sum (or difference) from the adder/subtractor unit onto the bus: bus <- Z
- L_B: Load B. B Register gets value on the bus: B <- bus
- L_O: Load Output. Output Register gets value on the bus: O <- bus

Introduction to Microprocessor



Introduction to Microprocessor

Machine cycle and Instruction cycle:

- SAP1 has six T-states (three Fetch and three Execute cycles) reserved for each instruction. Not all instructions require all the six T-states for execution. The unused T-state is marked as No Operation (NOP) cycle. Each T-state is called a machine cycle for SAP1. A ring counter is used to generate a T-state at every falling edge of clock pulse. The ring counter output is reset after the 6th T-state.

FETCH CYCLE	–	T1, T2, T3 machine cycle
EXECUTE CYCLE	-	T4, T5, T6 machine cycle

- SAP-1 executes the instruction in two cycle called Fetch and Execute cycle. The control unit called controller/sequencer has 12 output lines/signals. This unit generates the control words that fetch and execute each instruction. While each instruction is fetched and execute, the computer passes through defined timing states (T-States), periods during which register contents change.
- Fetch cycle uses 3 T-States T1, T2 & T3 and Execute cycles uses another 3-States T4, T5 & T6.
- Fetch cycles states T1, T2, & T3 is called Address State, Increment State and memory State.

Introduction to Microprocessor

Machine cycle and Instruction cycle:

In address state (T1-State), the address in the program counter (PC) is transferred to the Memory Address Register (MAR). During the address state, E_p and $\overline{L}_m$ are active and all other control bits are inactive. So, the control sequencer is sending out a control word as

Control Word =	Cp	Ep	$\overline{Lm}$	$\overline{CE}$	$\overline{Li}$	$\overline{Ei}$	$\overline{La}$	E_A	Su	Eu	$\overline{Lb}$	$\overline{Lo}$
	0	1	0	1	1	1	1	0	0	0	1	1

During increment State, the program counter (PC) is incremented and the controller sequencer is producing a control word as

Control Word =	Cp	Ep	$\overline{Lm}$	$\overline{CE}$	$\overline{Li}$	$\overline{Ei}$	$\overline{La}$	E_A	Su	Eu	$\overline{Lb}$	$\overline{Lo}$
	1	0	1	1	1	1	1	0	0	0	1	1

In T3-State called Memory State, addressed RAM instruction is transferred from the memory to the instruction register (IR) and $\overline{CE}$ and $\overline{Li}$ are active. So, control word is

Control Word =	Cp	Ep	$\overline{Lm}$	$\overline{CE}$	$\overline{Li}$	$\overline{Ei}$	$\overline{La}$	E_A	Su	Eu	$\overline{Lb}$	$\overline{Lo}$
	0	0	1	0	0	1	1	0	0	0	1	1

So, Address, Increment and Memory states are called the fetch cycle of SAP-1.

Introduction to Microprocessor

Machine cycle and Instruction cycle:

Execution Cycle:

- The next 3-States (T4, T5 & T6) are the execution cycle of SAP-1. The register transfer during the execution cycle depends on the particular instruction being executed.
- For example: LDA 9_H requires different register transfer than ADD B_H

Introduction to Microprocessor

Control words and Memory:

Macro	State	Cp	Ep	Lm'	CE'	Li'	Ei'	La'	Ea	Su	Eu	Lb'	Lo'	Active	CON
Fetch	T1	0	1	0	1	1	1	1	0	0	0	1	1	Ep,Lm'	5E3H
	T2	1	0	1	1	1	1	1	0	0	0	1	1	Cp	BE3H
	T3	0	0	1	0	0	1	1	0	0	0	1	1	CE',Li'	263H
LDA	T4	0	0	0	1	1	0	1	0	0	0	1	1	Lm',EI'	1A3H
	T5	0	0	1	0	1	1	0	0	0	0	1	1	CE',La'	2C3H
	T6	0	0	1	1	1	1	1	0	0	0	1	1	None	3E3H
ADD	T4	0	0	0	1	1	0	1	0	0	0	1	1	Lm',EI'	1A3H
	T5	0	0	1	0	1	1	1	0	0	0	0	1	CE',Lb'	2E1H
	T6	0	0	1	1	1	1	0	0	0	1	1	1	La',Eu	3C7H
SUB	T4	0	0	0	1	1	0	1	0	0	0	1	1	Lm',EI'	1A3H
	T5	0	0	1	0	1	1	1	0	0	0	0	1	CE',Lb'	2E1H
	T6	0	0	1	1	1	1	0	0	1	1	1	1	La',Su,Eu	3CFH
OUT	T4	0	0	1	1	1	1	1	1	0	0	1	0	Ea,Lo'	3F2H
	T5	0	0	1	1	1	1	1	0	0	0	1	1	None	3E3H
	T6	0	0	1	1	1	1	1	0	0	0	1	1	None	3E3H

Address	CON	Routine
0H	5E3H	Fetch
1H	BE3H	
2H	263H	
3H	1A3H	LDA
4H	2C3H	
5H	3E3H	
6H	1A3H	ADD
7H	2E1H	
8H	3C7H	
9H	1A3H	SUB
AH	2E1H	
BH	3CFH	
CH	3F2H	OUT
DH	3E3H	
EH	3E3H	
FH		

Introduction to Microprocessor

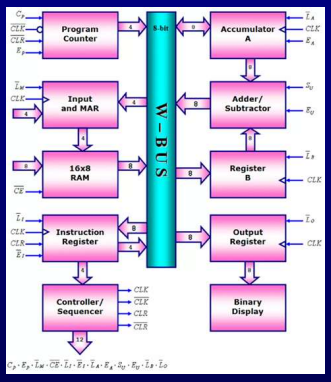
Problems:

1. Value 8H is stored at memory locations 7H. State the operations and control signals at T- states t1, t2, t3, t4, t5 and t6 of instruction ADD 8H.

t1: MAR ← PC ; Ep, Lm'

t2: PC ← PC +1 ;

t3: IR ← RAM



Address	Content
0H	LDA 9H
1H	
2H	
3H	
4H	
5H	
6H	
7H	08H
8H	
9H	
AH	
BH	
CH	
DH	
EH	
FH	

Introduction to Microprocessor

Problems:

1. Value 8H is stored at memory locations 7H. State the operations and control signals at T- states t1, t2, t3, t4, t5 and t6 of instruction ADD 8H.

Address	Content
0H	LDA 9H
1H	ADD 7H
2H	SUB 8H
3H	OUT
4H	HLT
5H	
6H	
7H	08H
8H	04H
9H	06H
AH	
BH	
CH	
DH	
EH	
FH	

Solution:

ADD 7H

t1: MAR ← PC ; E_P, L_M'

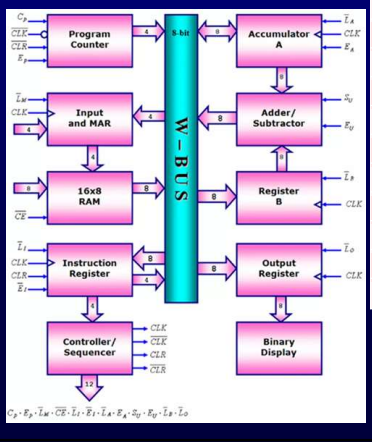
t2: PC ← PC +1 ; C_P

t3: IR ← RAM ; C_E' L_I'

t4: MAR ← IR ; L_M' E_I'

t5: B ← RAM ; C_E' L_B'

t6: A ← A + B ; L_A' E_U



Introduction to Microprocessor

Problems:

1. Values in memory contents are shown in memory. State the operations and control signals at T- states t1, t2, t3, t4, t5 and t6 of instruction LDA 9H, SUB 8H and OUT.

Address	Content
0H	LDA 9H
1H	ADD 7H
2H	SUB 8H
3H	OUT
4H	HLT
5H	
6H	
7H	08H
8H	04H
9H	06H
AH	
BH	
CH	
DH	
EH	
FH	

Solution:

Instruction:……

Clock State & Operation Control Signals

t1: …… ; …….

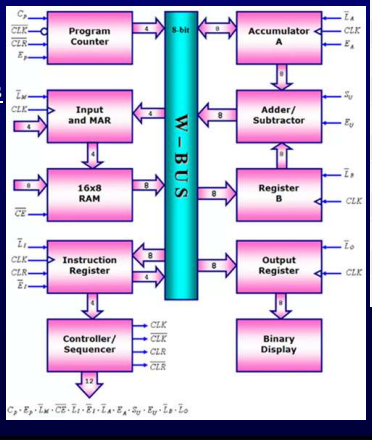
t2: …… ; …….

t3: …… ; …….

t4: …… ; …….

t5: …… ; …….

t6: …… ; …….



Introduction to Microprocessor

SAP-2 Architecture:

Block Diagram and Functions of each Block,

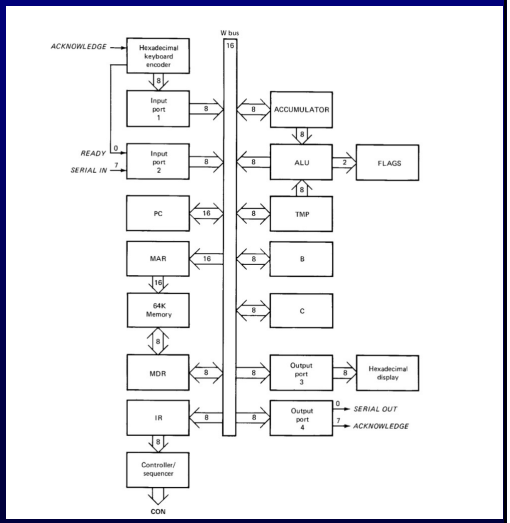
Architectural Differences with SAP-1

- **Bidirectional Registers**
- **Flags:**

Introduction to Microprocessor

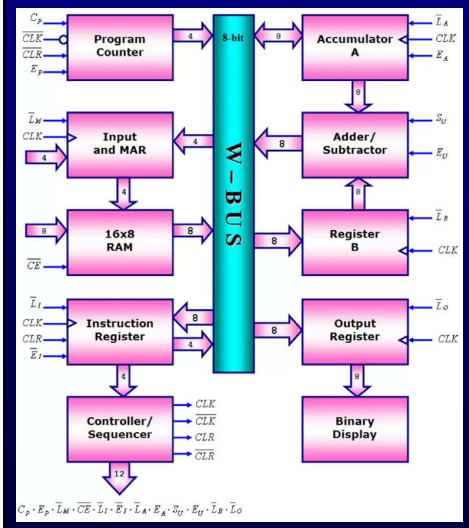
SAP-2 Architecture: Block Diagram and Functions of each Block:

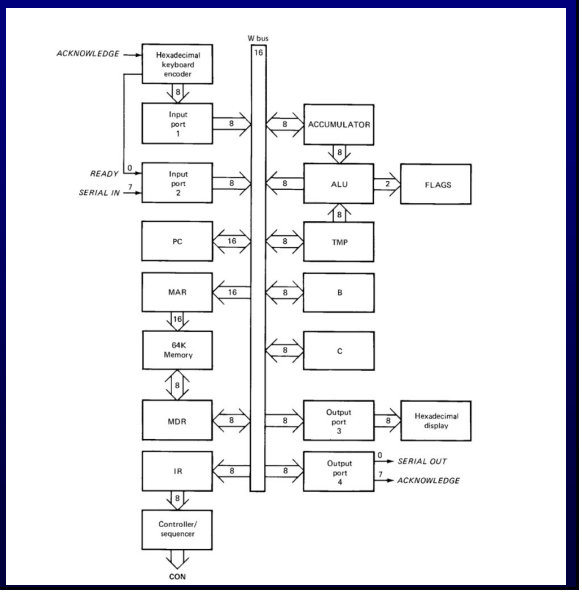
- SAP-2 is the enhanced version of SAP-1 and provides better computing capabilities.
- Many new features are added in SAP-2
- RAM in the case of SAP-2 is 64k. The bus for SAP-2 is 16 bits. Control sequencer is also bigger in comparison to SAP-1. Memory data register in SAP-2 helps to retain the result of a memory access. B register is also now known as TMP in SAP-2. The unit of Adder/ Subtractor is also known as Arithmetic/Logic unit as it also carries out logical tasks rather than just subtraction and addition. In SAP-2 the instruction can occur in variable length.



Introduction to Microprocessor

SAP-2 Architecture: Block Diagram and Functions of each Block:





Introduction to Microprocessor

SAP-2 Architecture: Block Diagram and Functions of each Block:

- **Hexadecimal Keyboard Encoder:** The hexadecimal keyboard encoder receives the data from outer environment and converts it into hexadecimal form. The system can understand and send them to the input port.
- **Input Ports:** The SAP-2 contains two input ports which input the data in the system in the most convenient way.
- **PC:** PC is the program counter that holds the address of the next instruction to be fetched. It initializes from 0000H to 1111H during the execution.
- **MAR:** MAR is the memory address register that stores the complete format of the address sent by the program counter. It stores the final address of the memory word that needs some computations.
- **64K Memory:** It contains 64 K memory where data and instruction reside. All the computations are performed relative to the memory.

Introduction to Microprocessor

SAP-2 Architecture: Block Diagram and Functions of each Block:

- **MDR:** MDR is the Memory Data Register which stores the data or operand that is fetched from the memory which is needed for computation.
- **IR:** The IR is the Instruction Register that holds the complete format of the Instruction that is to be executed.
- **Control Sequencer:** It provides necessary timing signals like T0, T1, T2, ….. and control signals providing the direction for executing the program.
- **Accumulator:** The result of all the mathematical operations is stored in accumulator. It is one of the operand of ADD, OUT, SUB instruction. It is also known as processor register.
- **ALU and Flag:** The ALU perform all the arithmetic and logical calculations. The flag reflect the intermediate changes on the values during execution.

Introduction to Microprocessor

SAP-2 Architecture: Block Diagram and Functions of each Block:

- **Temporary register, B, C:** They are the second operand of the mathematical operations. The register B and C is accessible to the programmer.
- **Output Ports:** It consists of two output ports to show the result of OUT instruction.
- **Hexadecimal Display:** Unlike SAP-1 which has binary display, SAP-2 has a hexadecimal display to show outputs in the LEDs.

Introduction to Microprocessor

Features of SAP-2 Architecture:

- Jump Instructions: loadable PC
- 16-bit program counter
- 8-bit op-code, 42 instructions
- 2 input ports, 2 output ports
- 2K ROM, up to 64K RAM (16-bit addresses) with read and write
- Memory data register (MDR) buffers reads and writes
- Accumulator(AC) can write to bus
- Temporary, B and C registers
- 16 arithmetic and logic operations in ALU
- Sign and zero flag

Introduction to Microprocessor

Features of SAP-2 Architecture:

Flags

- 2 flip flops, sign flag and zero flag
- set during arithmetic and logic operations to reflect final accumulator contents
- JM jumps only if the sign flag is set (minus result)
- JZ jumps only if the zero flag is set (zero result)
- JNZ jumps if the zero flag is clear (non-zero result)

Introduction to Microprocessor

SAP-2 Instruction Sets:

- Same fetch cycle (T1, T2, T3) as SAP-1
- Memory reference instructions: LDA, STA (3 bytes, lower byte before higher byte)
- Immediate instructions: MVI reg, value (2 bytes)
- Register instructions: MOV, ADD and SUB, INR and DCR, ANA, ORA, XRA, CMA (1 byte), ANI, ORI, XRI (2 bytes)
- Jump and Call instructions: JMP, JM, JZ, JNZ, CALL (3 bytes), RET (1 byte)
- CALL saves return address in memory FFFEH and FFFFH
- NOP, HLT, RAL, RAR (1 byte), IN, OUT (2 bytes)

Introduction to Microprocessor

Comparison between Sap 1 and Sap 2:

Description	SAP 1	SAP 2
Full Form	Simple As Possible 1	Simple as Possible 2
Logical operations	Cannot be handled	Can be handled
Addresses	8 bits	16 bits
Opcodes	4 bits	8 bits
Instructions	Fixed length	Variable length.
Adder/subtractor unit	Carries out simple operations like addition and subtraction	Carries out logical operations too and therefore also known as ARITHMETIC/LOGIC UNIT
Size of control sequencer	Smaller	Bigger due to extra registers
Number of instructions	Comparatively lesser	Comparatively greater
Register to hold the data being added or subtracted from the accumulator	B	TMP
Jump instructions	Not available	JM, JZ and JNZ – conditional jumps JMP – unconditional jumps
Instructions	Includes instruction – LDA, ADD, SUB, OUT and HLT	Includes instructions like ADD, SUB, INR, JMP, DCR, ANA, ORA, XRA, ANI, ORI, XRI, etc.
Flag	No Flag	sign and zero flag
Display to show outputs in the LEDs	Binary	Hexadecimal

Introduction to Microprocessor

All 42 SAP-2 Instruction Sets:

Instruction	Op Code	*T* States	Flags	Addressing	Bytes
ADD B	80	4	S, Z	Register	1
ADD C	81	4	S, Z	Register	1
ANA B	A0	4	S, Z	Register	1
ANA C	A1	4	S, Z	Register	1
ANI byte	E6	7	S, Z	Immediate	2
CALL address	CD	18	None	Immediate	3
CMA	2F	4	None	Implied	1
DCR A	3D	4	S, Z	Register	1
DCR B	05	4	S, Z	Register	1
DCR C	0D	4	S, Z	Register	1
HLT	76	5	None	—	1
IN byte	DB	10	None	Direct	2
INR A	3C	4	S, Z	Register	1
INR B	04	4	S, Z	Register	1
INR C	0C	4	S, Z	Register	1
JM address	FA	10/7	None	Immediate	3
JMP address	C3	10	None	Immediate	3
JNZ address	C2	10/7	None	Immediate	3
JZ address	CA	10/7	None	Immediate	3
LDA address	3A	13	None	Direct	3
MOV A,B	78	4	None	Register	1
MOV A,C	79	4	None	Register	1
MOV B,A	47	4	None	Register	1
MOV B,C	41	4	None	Register	1
MOV C,A	4F	4	None	Register	1
MOV C,B	48	4	None	Register	1
MVI A,byte	3E	7	None	Immediate	2
MVI B,byte	06	7	None	Immediate	2
MVI C,byte	0E	7	None	Immediate	2
NOP	00	4	None	—	1
ORA B	B0	4	S, Z	Register	1
ORA C	B1	4	S, Z	Register	1
ORI byte	F6	7	S, Z	Immediate	2
OUT byte	D3	10	None	Direct	2
RAL	17	4	None	Implied	1
RAR	1F	4	None	Implied	1
RET	C9	10	None	Implied	1
STA address	32	13	None	Direct	3
SUB B	90	4	S, Z	Register	1
SUB C	91	4	S, Z	Register	1
XRA B	A8	4	S, Z	Register	1
XRA C	A9	4	S, Z	Register	1
XRI byte	EE	7	S, Z	Immediate	2

Introduction to Microprocessor

Questions:

1. Explain the different states of Fetch and Execution Cycle of SAP-1 instructions.
2. Draw the block diagram of SAP-1 and explain the function of each block diagram in brief.
3. Explain the step by step operations (Fetch and Execute cycle) of LDA 9_H of SAP-1 instruction.
4. Explain the difference between SAP-1 and SAP-2 in brief.
5. Write a SAP-1 program using mnemonics that will display the result of 5 + 4 – 6, Use addresses DH, EH. and FH for the data.
6. Explain the difference of address bus and RAM memory of SAP-1 and SAP-2 computer architecture.

Introduction to Microprocessor

Questions:

1. What is Microprocessor? Explain the difference between Microcomputer, Microprocessor and Microcontroller.
2. What is Microprocessor system bus. Explain the function of different types of system buses.
3. What are the 3 general categories of the function of microprocessor? Explain.