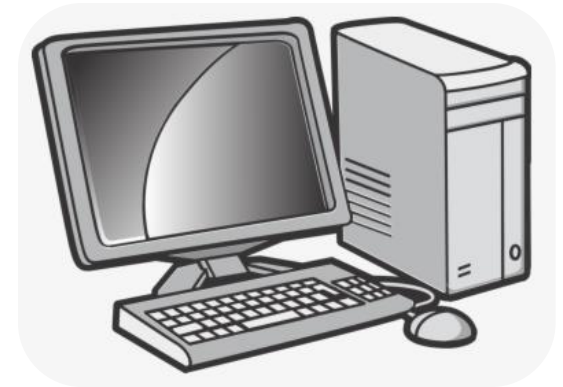
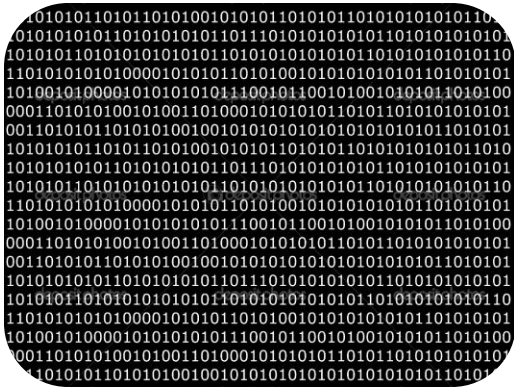


AMIRAJ

COLLEGE OF ENGINEERING & TECHNOLOGY

Unit-2

Basic Computer Organization and Design



Subject:- COA
Code:-3140707

Prepared by:
Asst. Prof. S Y JOSHI
(CSE Department, ACET)

AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

Topics to be covered

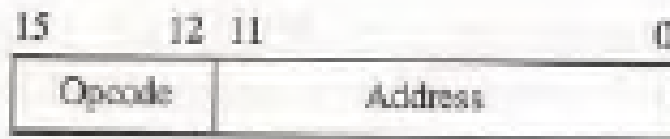
- Instruction codes
- Computer registers
- computer instructions
- Timing and Control
- Instruction cycle
- Memory-Reference Instructions
- Input-output and interrupt
- Complete computer description
- Design of Basic computer
- Design of Accumulator Unit

Instruction Codes

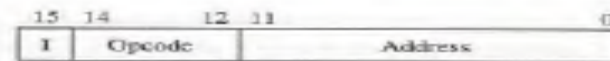
- An instruction code is a group of bits that instruct the computer to perform a specific operation. It is usually divided into parts, each having its own particular interpretation.
- The most basic part of an instruction code is its operation part.
- The operation code of an instruction is a group of bits that define such operations as add, subtract, multiply, shift, and complement.
- The number of bits required for the operation code of an instruction depends on the total number of operations available in the computer.
- The operation code must consist of at least n bits for a given 2^n (or less) distinct operations

Instruction Codes

- Below is shown the instruction format - four bits for the operation code (abbreviated opcode), and 12 bits to specify the address of an operand
- It is sometimes convenient to use the address bits of an instruction code not as an address but as the actual operand. When the second part of an instruction immediate code specifies an operand, the instruction is said to have an immediate instruction operand. When the second part specifies the address of an operand, the instruction is said to have a direct address. This is in contrast to a third possibility called indirect address, where the bits in the second part of the instruction designate an address of a memory word in which the address of the operand is found. One bit of the instruction code can be used to distinguish between a direct and an indirect address. *The I bit is 0, so instruction is direct address instruction, if I is 1 then indirect*



Instruction format



(a) Instruction format

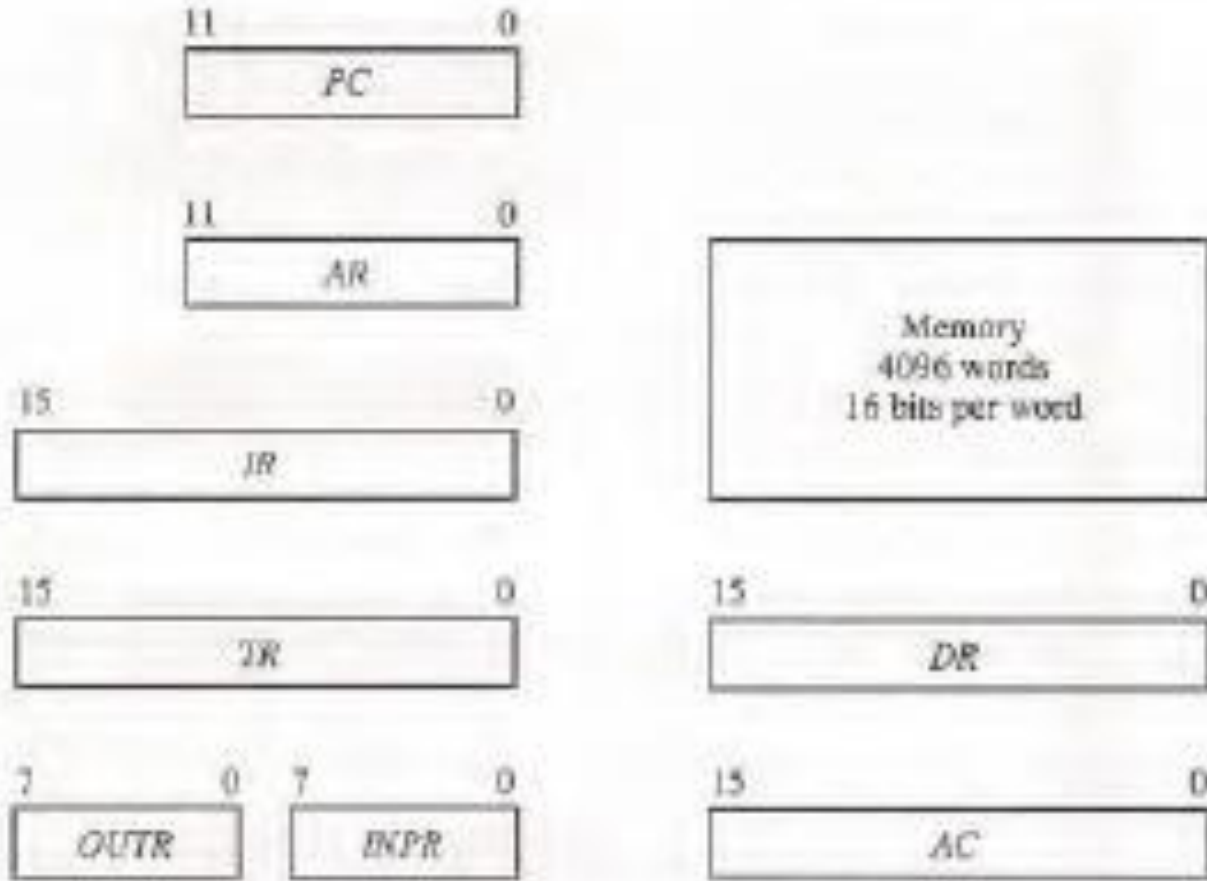
Computer Registers

- It is necessary to provide a register in the control unit for storing the instruction, code after it is read from memory. The computer needs processor registers for manipulating data and a register for holding a memory address.

Register	Number of bits	Register name	Function
DR	16	Data register	Holds memory operand
AR	12	Address register	Holds address for memory
AC	16	Accumulator	Processor register
IR	16	Instruction register	Holds instruction code
PC	12	Program counter	Holds address of instruction
TR	16	Temporary register	Holds temporary data
INPR	8	Input register	Holds input character
OUTR	8	Output register	Holds output character

Computer Registers

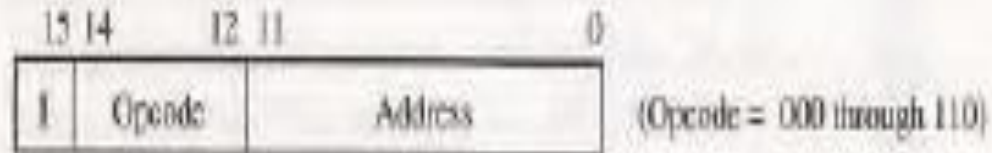
Basic computer registers and memory.



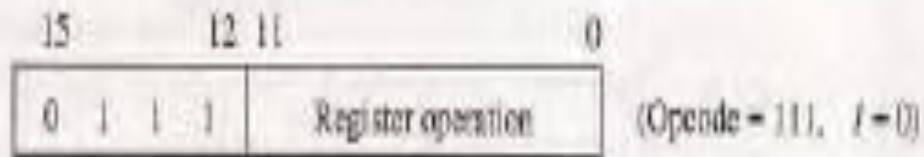
Computer instructions

- The basic computer has three instruction code formats,
- Each format has 16 bits.
- The operation code (opcode) part of the instruction contains three bits and the meaning of the remaining 13 bits depends on the operation code encountered.
- A memory-reference instruction uses 12 bits to specify an address and one bit to specify the addressing code I.
- I is equal to 0 for direct address and to 1 for indirect address.
- The register reference instructions are recognized by the operation code 111 with a 0 in the leftmost bit (bit 15) of the instruction.
- A register-reference instruction specifies an operation on or a test of the AC register. An operand from memory is not needed; therefore, the other 12 bits are used to specify the operation or test to be executed.
- Similarly, an input-output instruction does not need a reference to memory and is recognized by the operation code 111 with a 1 in the leftmost bit of the instruction. The remaining 12 bits are used to specify the type of input-output operation or test performed.

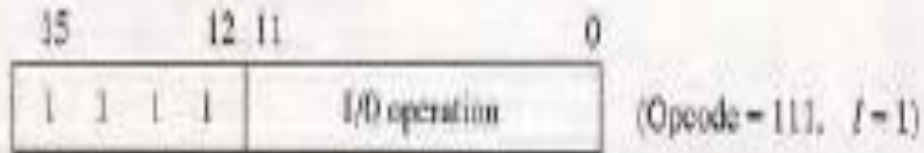
Computer instructions



(a) Memory - reference instruction



(b) Register - reference instruction



(c) Input - output instruction

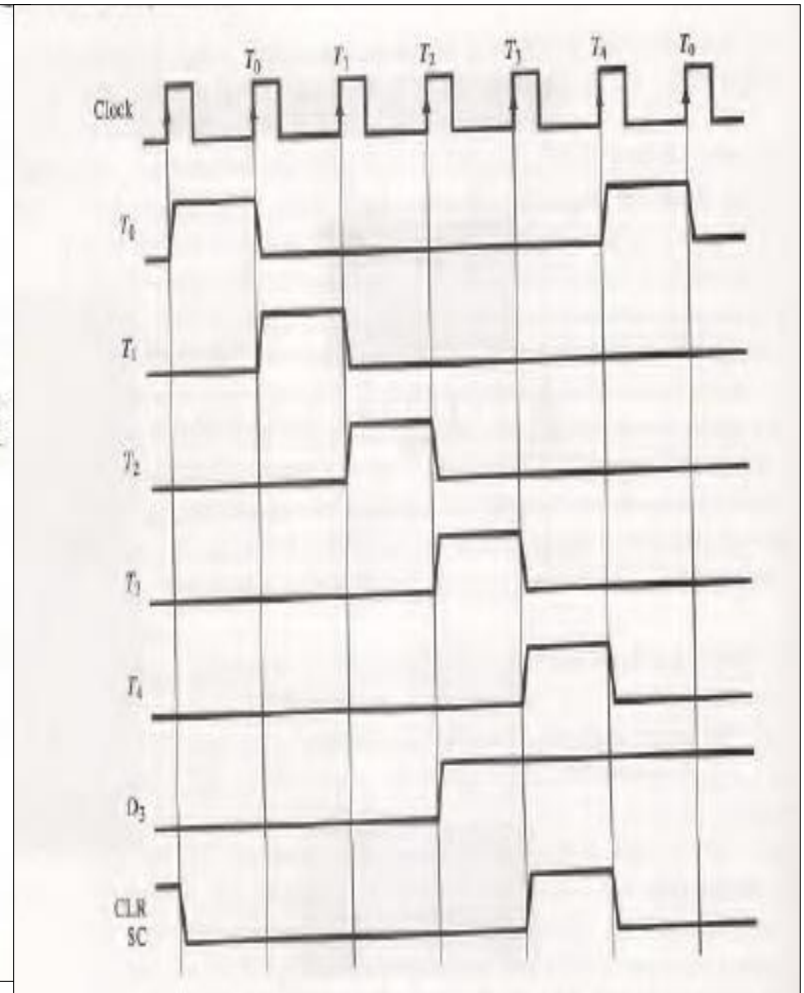
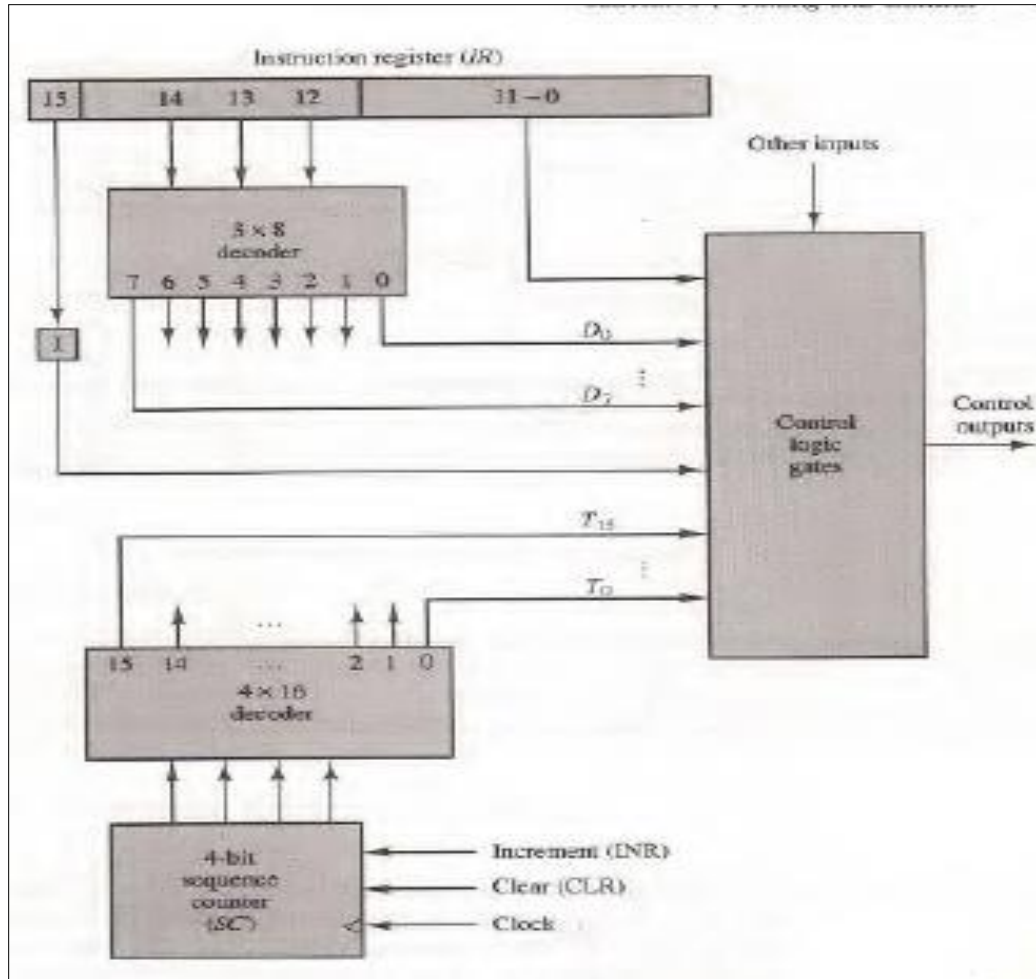
Computer instructions

- The set of instructions are said to be complete if the computer includes a sufficient number of instructions in each of the following categories:
 - 1. Arithmetic, logical, and shift instructions
 - 2. Instructions for moving information to and from memory and processor registers
 - 3. Program control instructions together with instructions that check status conditions
 - 4. Input and output instructions

Timing and Control

- The timing for all registers in the basic computer is controlled by a master clock generator.
- The clock pulses are applied to all flip-flops and registers in the system, including the flip-flops and registers in the control unit.
- The clock pulses do not change the state of a register unless the register is enabled by a control signal.
- The control signals are generated in the control unit and provide control inputs for the multiplexers in the common bus, control inputs in processor registers, and micro operations for the accumulator.
- The outputs of the counter are decoded into 16 timing signals T0 through T15 .REFER FIGURE

Timing and Control



Instruction Cycle

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

1. Fetch an instruction from memory.
2. Decode the instruction.
3. Read the effective address from memory if the instruction has an indirect address.
4. Execute the instruction.

Instruction Cycle

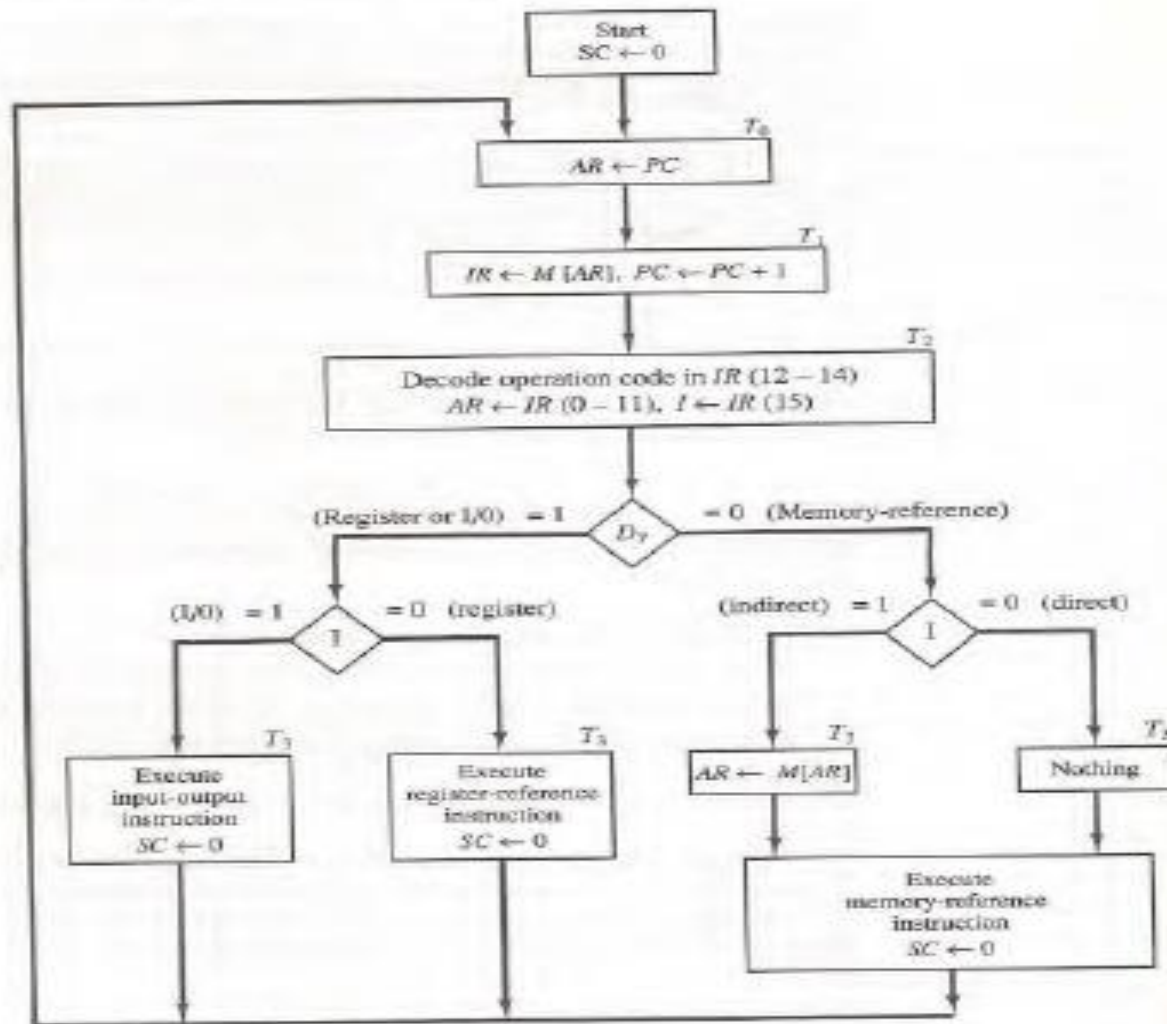


Figure 5-9 Flowchart for instruction cycle (initial configuration).

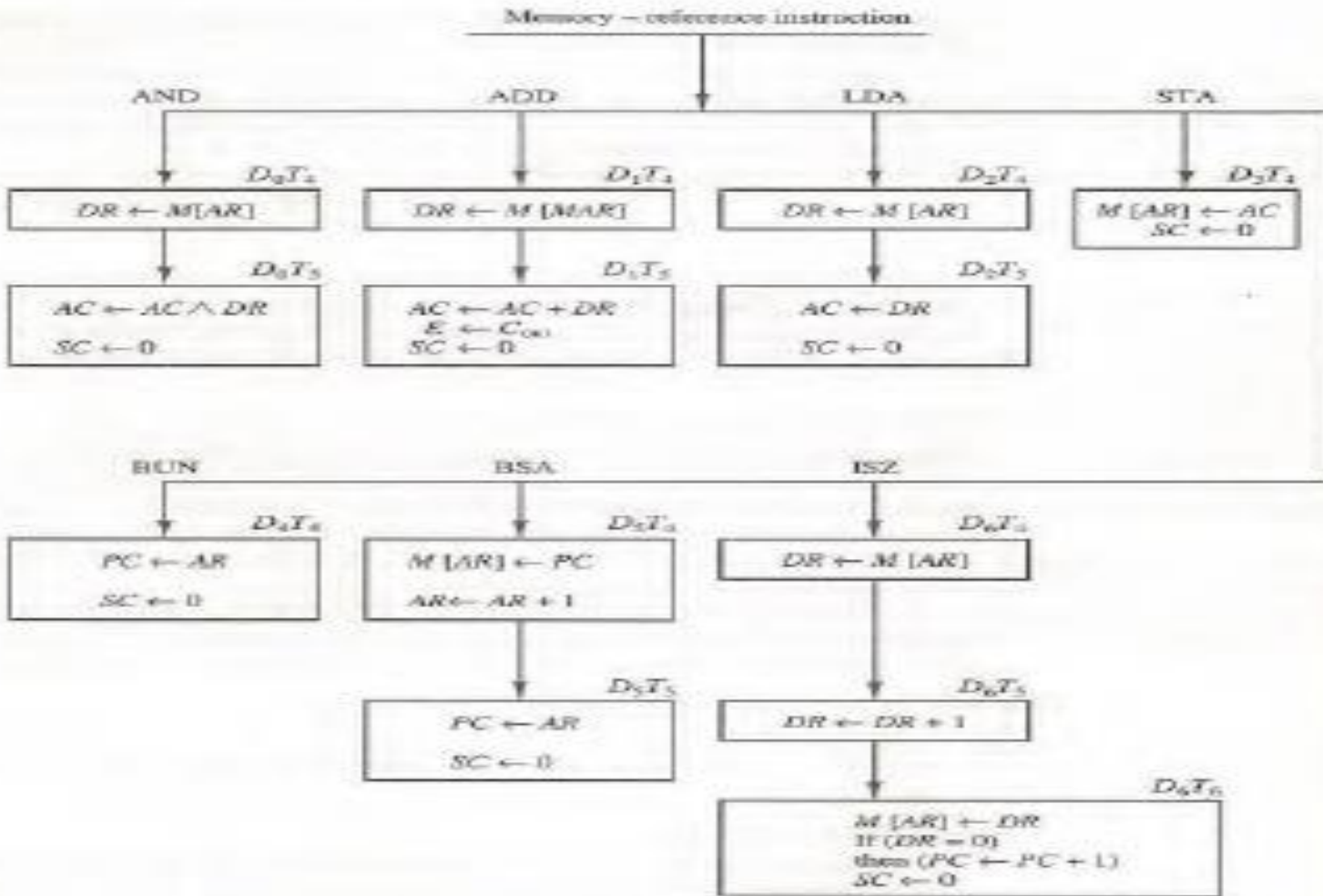
Memory Reference Instructions

Below lists the seven memory-reference instructions

Symbol	Operation decoder	Symbolic description
AND	D0	$AC \leftarrow AC \wedge M[AR]$
ADD	Dr	$AC \leftarrow AC + M[AR], E \leftarrow Cout$
LDA	D2	$AC \leftarrow M[AR],$
STA	D3	$M[AR] \leftarrow AC$
BUN	D4	$PC \leftarrow AR$
BSA	D5	$M[AR] \leftarrow PC, PC \leftarrow AR + 1$
ISZ	D6	$M[AR] \leftarrow M[AR] + 1,$ If $M[AR] + 1 = 0$ then $PC \leftarrow PC + 1$

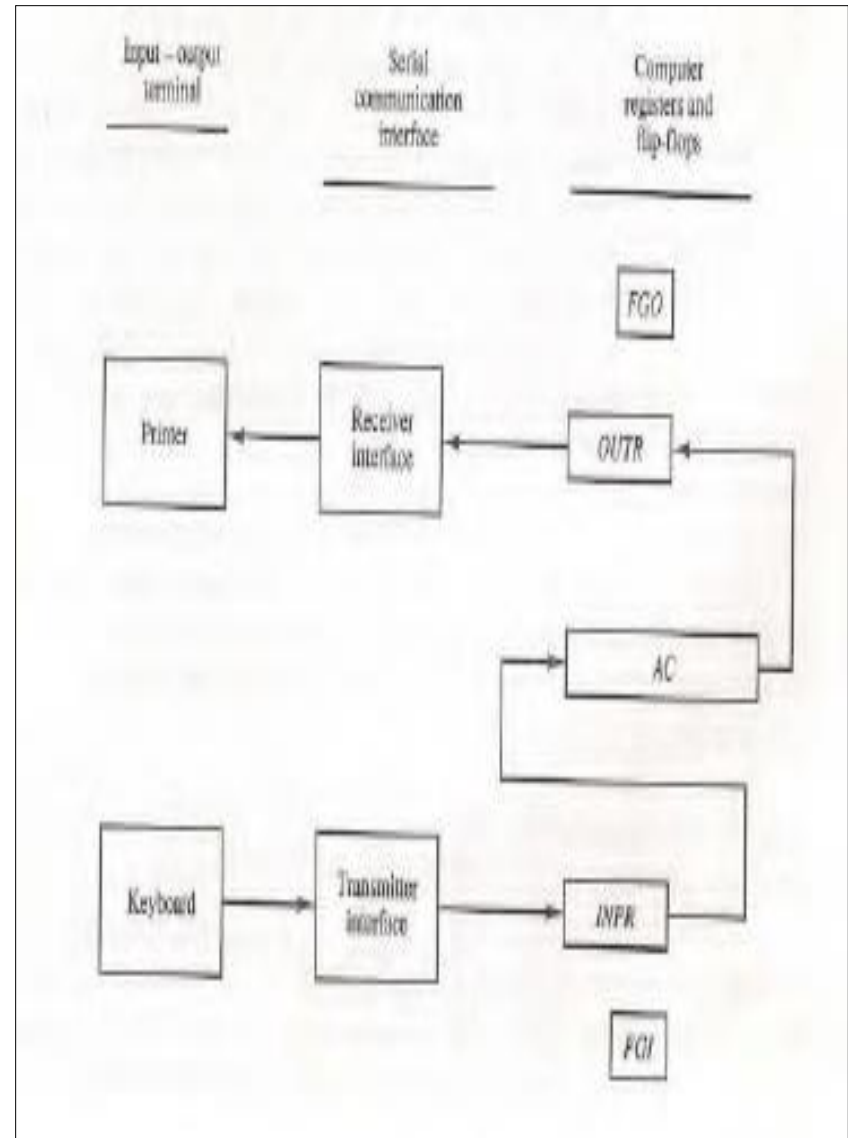
Memory Reference Instructions

Flow chart for memory reference instructions



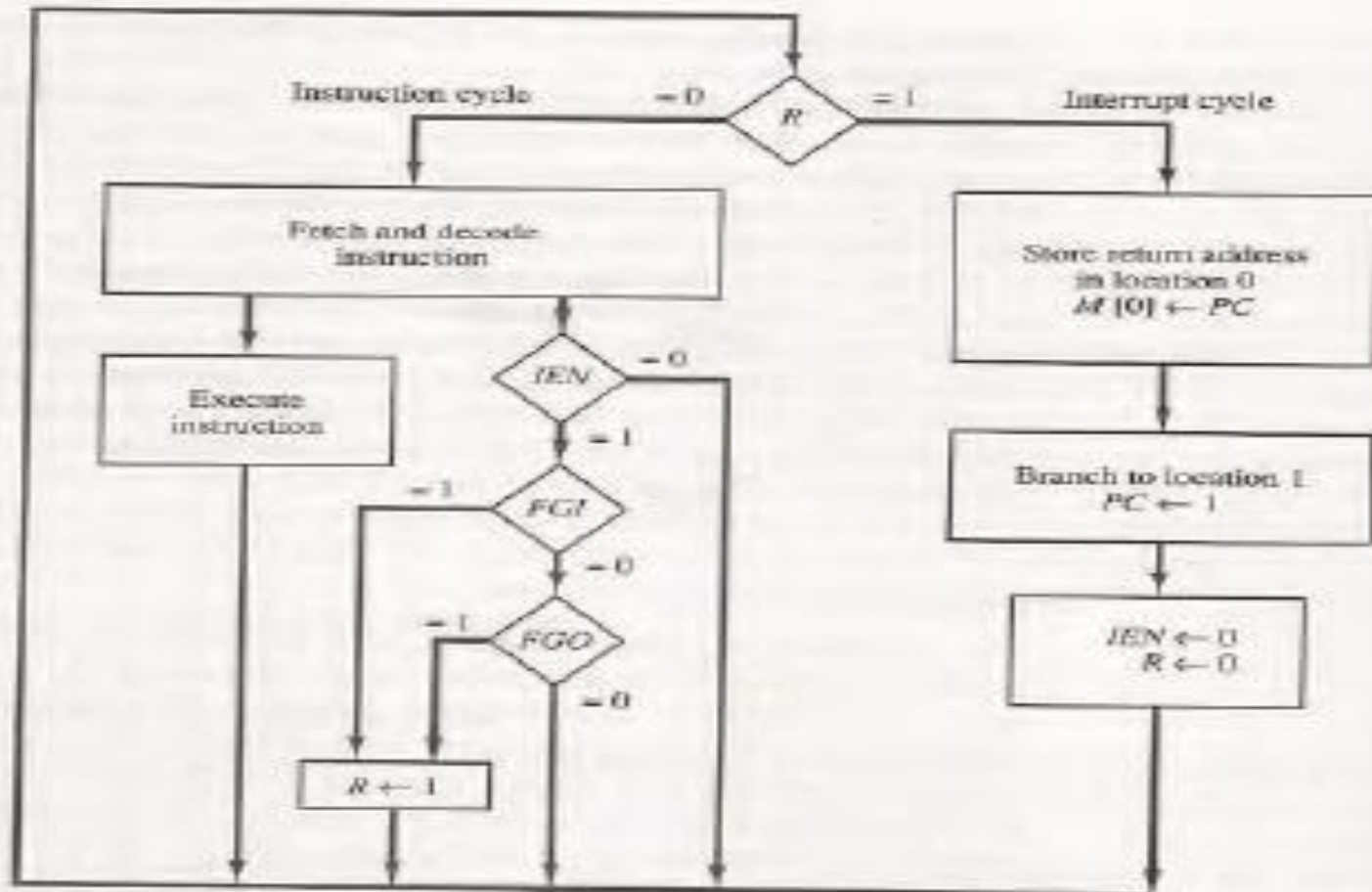
I/O and Interrupts

- A computer can serve no useful purpose unless it communicates with the external environment. Instructions and data stored in memory must come from some input device.
- Computational results must be transmitted to the user through some output device. Input output configuration - shows input output configuration- The terminal sends and receives serial information. Each quantity of information has eight bits of an alphanumeric code. The serial information from the keyboard is shifted into the input register INPR . The serial information for the printer is stored in the output register OUTR. These two registers communicate with a communication interface serially and with the AC in parallel.



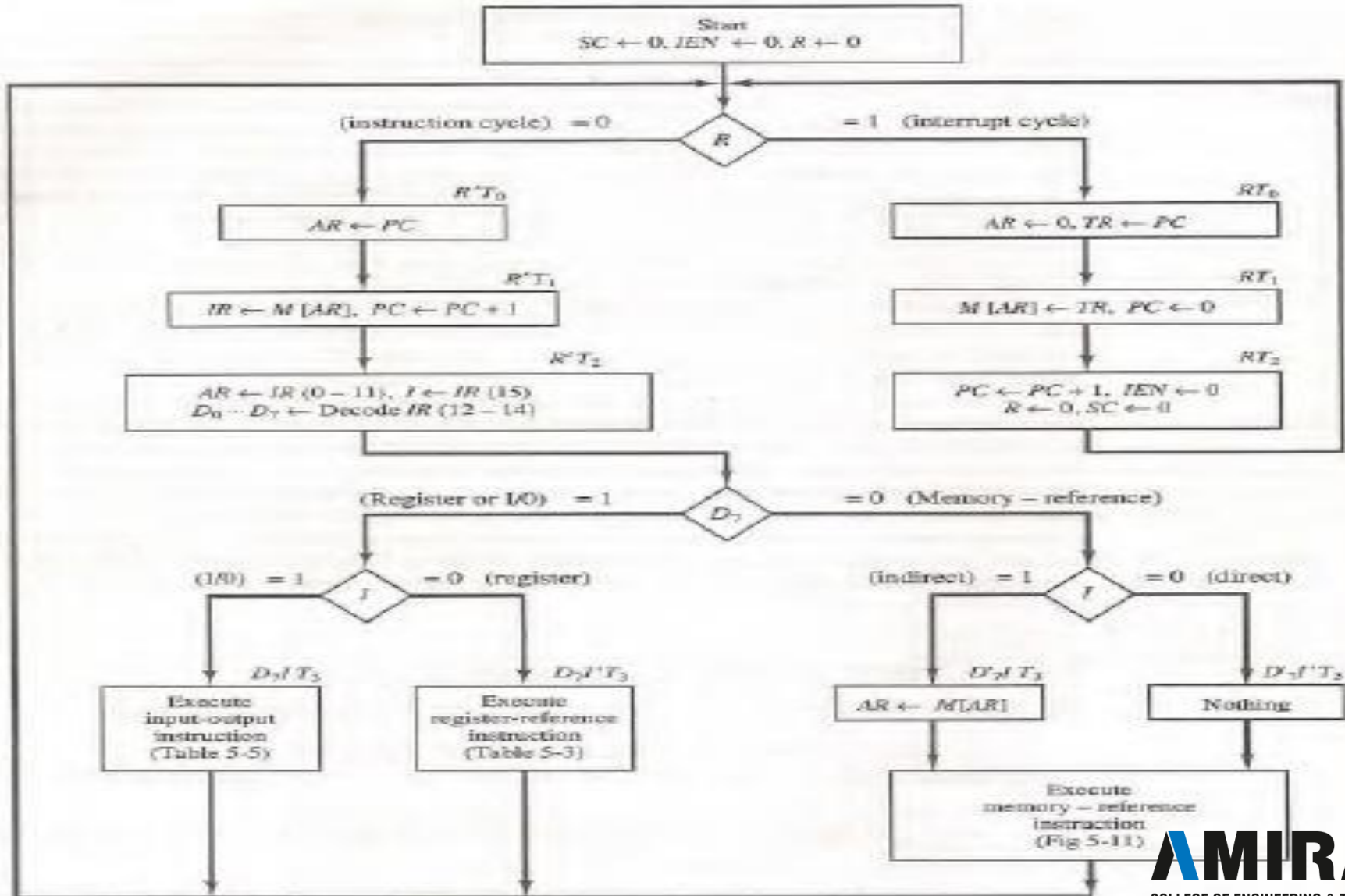
I/O and Interrupts

- Flow chart for interrupt cycle



Complete computer description

Flowchart of Complete computer

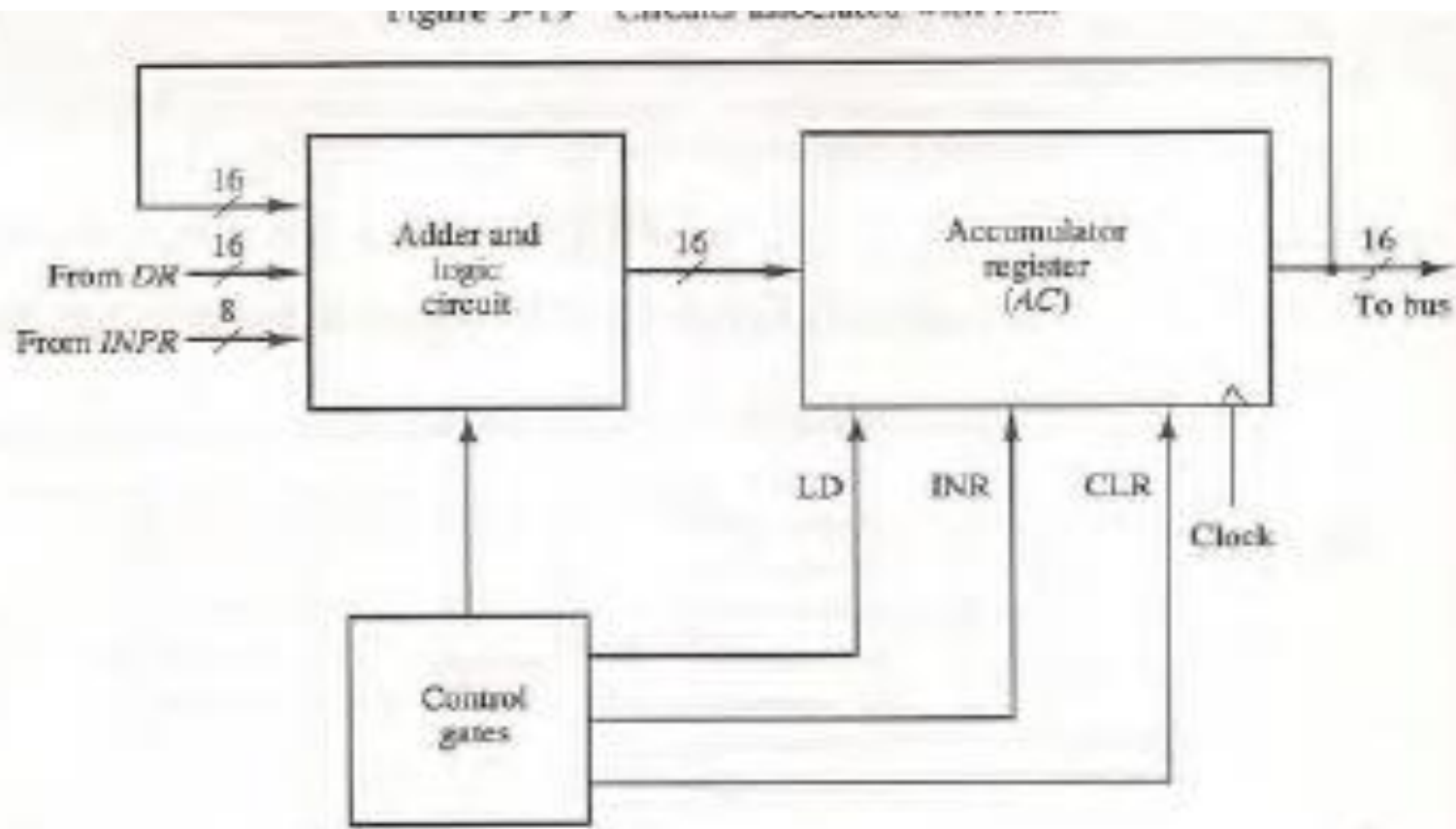


Design of Basic computer

- The basic computer consists of the following hardware components:
 - 1. A memory unit with 4096 words of 16 bits each
 - 2. Nine registers: AR, PC, DR, AC, IR, TR, OUTF, INPR, and SC
 - 3. Seven flip-flops: I, S, E, R, IEN, FGI, and FGO
 - 4. two decoders: a 3 x8 operation decoder and 4x16 timing decoder
 - 5. A 16-bit common bus
 - 6. Control logic gates
 - 7. Adder and logic circuit connected to the input of AC

Design of Accumulator Unit

- The circuits associated with the AC register are shown in Fig.



References

- Images , descriptive Tables , from Computer System Architecture, Morris Mano, 3rd edition Prentice Hall
- **Note: These pdf/ppt notes are for purpose of teaching aids to classroom/online sessions study, and in no case imply for GTU syllabus or GTU exam. For GTU syllabus or exam related preparation, one may, however will need to attend college/online lectures and refer books given by GTU in their official syllabus.**