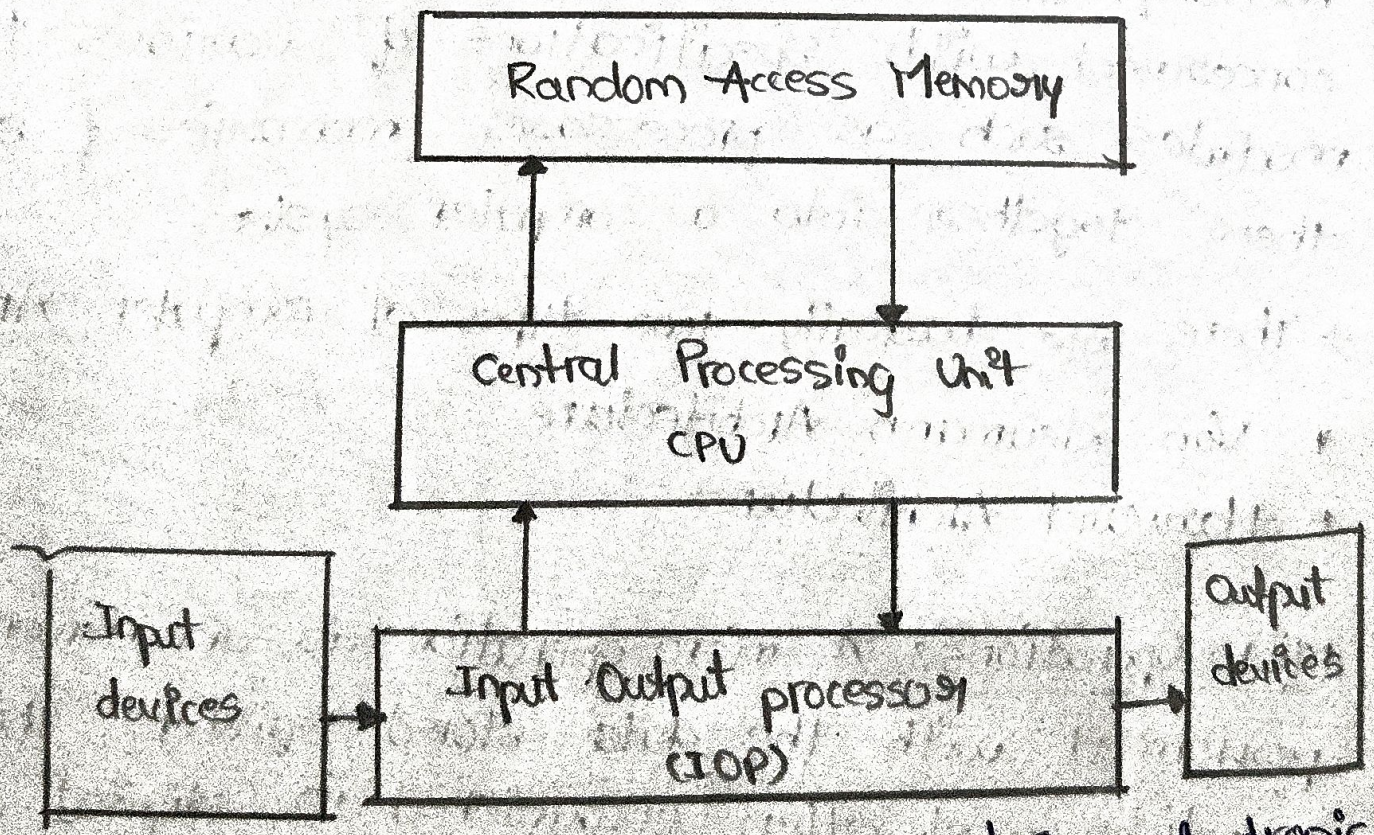
3a) A computer system is sometimes subdivided into 2 functional entities; hardware & software. The hardware of the computer is usually divided into 3 major parts. They are the CPU, arithmetic logic unit for data manipulation, a no of registers for storing data & for fetching and executing instructions. The memory of a computer contains storage for instruction & data. It is called the RAM because CPU can access any location in memory & retrive binary information within a fixed interval of Time.



The input & output processors contains electronic circuit for communications & controlling the transfer of information this computer & outside. The input & output devices are connected to computers including keyboards, disk drives, etc.

1b) Computer Organization: Computer Organization is a concern with the way the hardware components operate & the way they are connected together to form a computer system. The various components are assumed to be in place & the task is to investigate the organizational structure to verify that the computer parts operate as intended.

Computer Architecture: Computer Architecture is concerned with the structure & behavior of the computer as seen by user. It includes the information formats, the instruction set, techniques for address memory. The Architecture of the computer system is concerned with specifications of various functional modules such as processors, memories & structuring them together into a computer system.

→ There are basically two types of computer Architecture:

* Von Neumann Architecture
* Hoorward Architecture.

2a) Micro Operations: A micro operation is an elementary operation performed with the data stored registers the micro operations most often encountered in digital computers are classified into 4 categories.

→ Register transfer micro-operations transfer binary information from one register to another.

→ Arthimatic micro-operations perform arithmetic operation on numeric data stored in registers.

→ Logic micro-operations perform bit manipulation operations on non numeric data stored in registers.

→ Shift micro-operations perform shift operations on data stored in registers.

The another type of micro-operation is the register transfer micro operation. The basic micro operations are add, shift, subtraction, increment & decrement.

A set of micro operations together is called a Micro Instruction

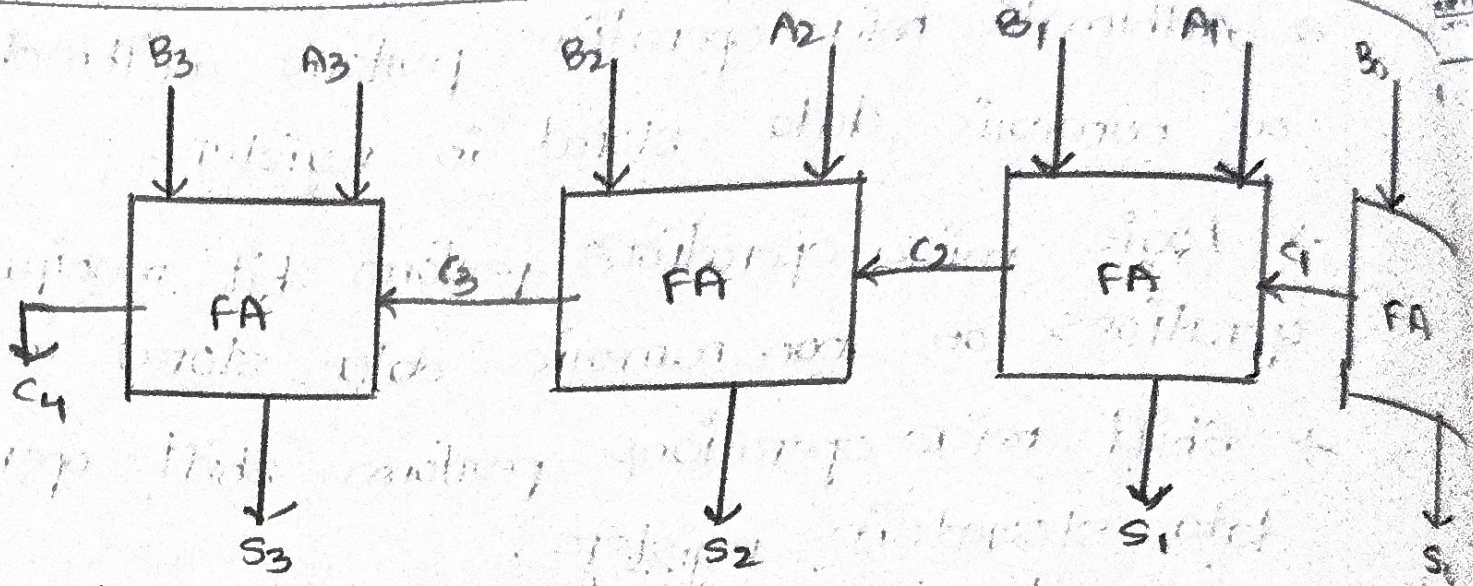
2b) Micro-Operation:- The operation performed on the register stored value is called a Micro Operation.

Types of Micro Operations:

Add Micro operation: The Add micro-operation states that the contents of register R1 are added to the contents of register R2 & the sum transfered to register R3. To implement this we need 8 registers & the digital component that perform addition operation.

$$R3 \leftarrow R1 + R2$$

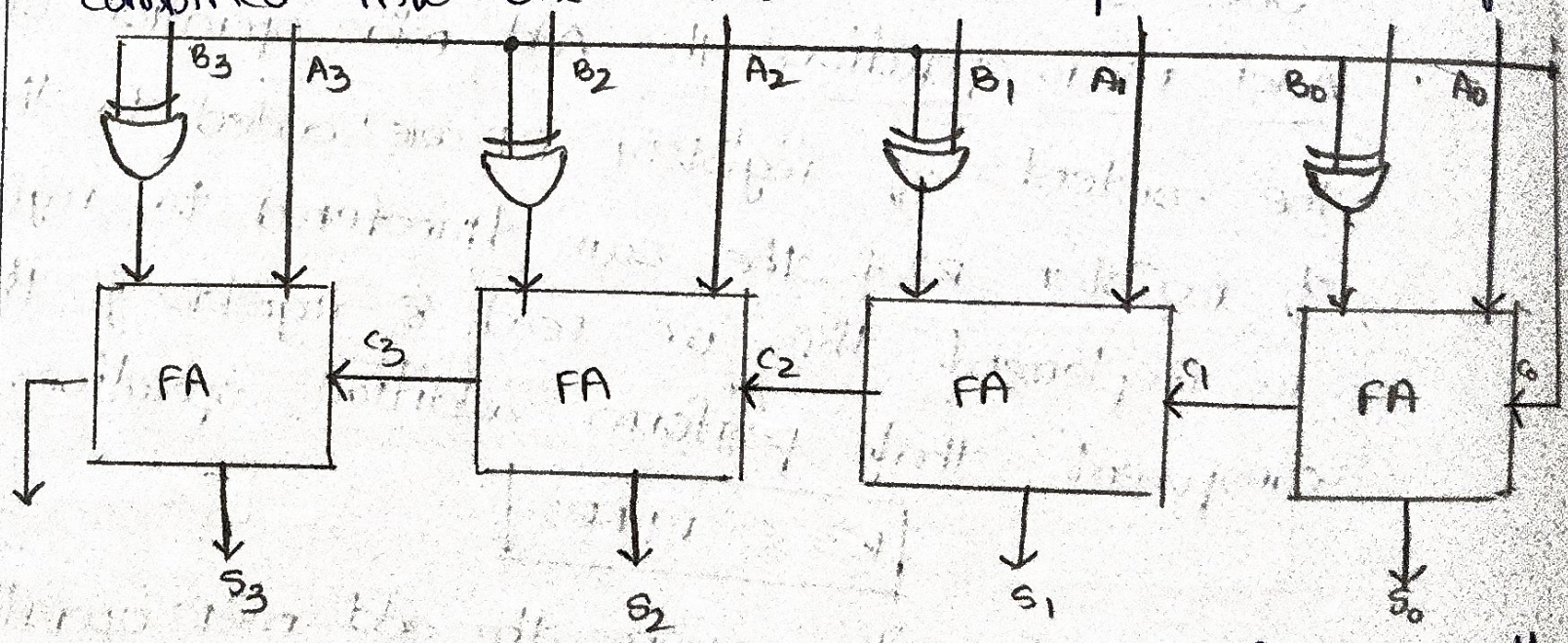
Binary Adder: To perform the add micro operation with hardware, we need to make registers hold the data. The digital circuit that performs the arithmetic sum of 2 bits & a previous carry is called a full-adder.



Subtract Micro Operations: It is most often implemented through completion of addition. Instead of using the minus operator, we can specify by following statement

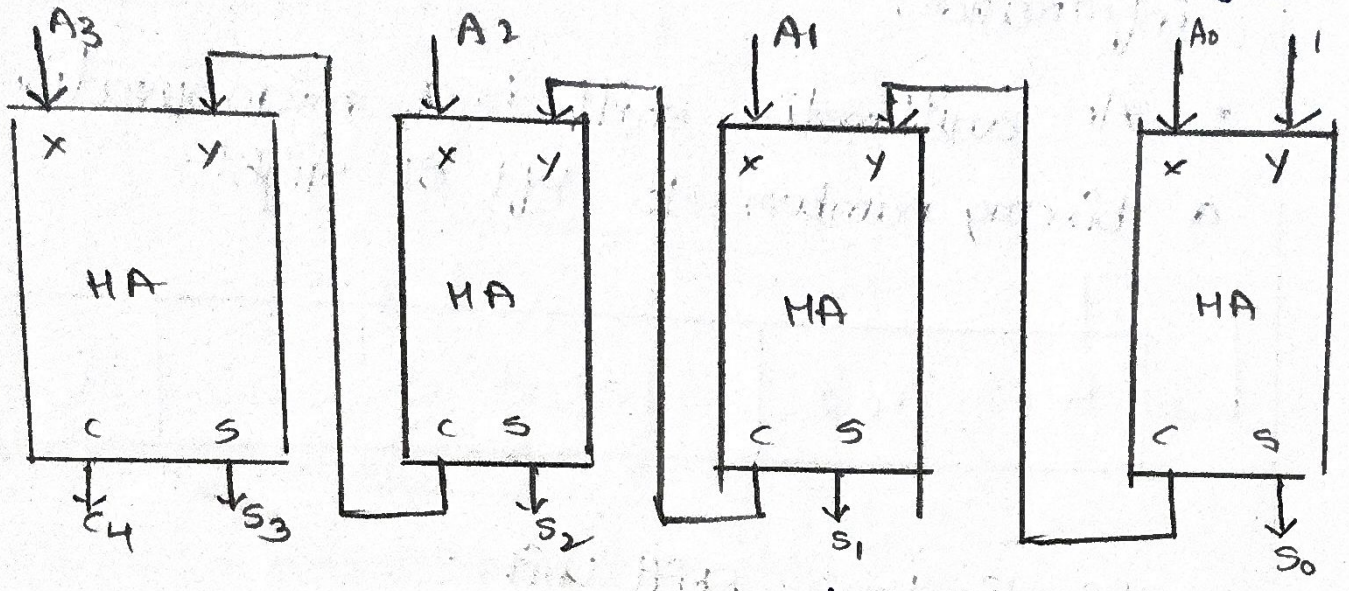
$$R3 \leftarrow R1 + \overline{R2} + 1$$

Binary adder substractor: The subtraction of binary numbers can be done most conveniently by means of complement. The addition & subtraction operations can combined into one common circuit by exclusive OR gate



where the Mode input M controls the operation. Here the B inputs are all complemented & 1 is added through the input carry. The circuit performs the operation A plus the 2's complement of B

Binary Incrementer: The increment micro operation adds one to a number in a register. If this operation has 4-bit value 0110. It will increment to 0111. This micro-operation is implemented by binary counter. The clock pulse transition increments the content of register by one.



The diagram of 4-bit incrementer has 4-full adder circuits that has 4 multiplexers. We will have 4 bit inputs A & B & 4 bit output D. The output is calculated by $D = A + y + C_{in}$

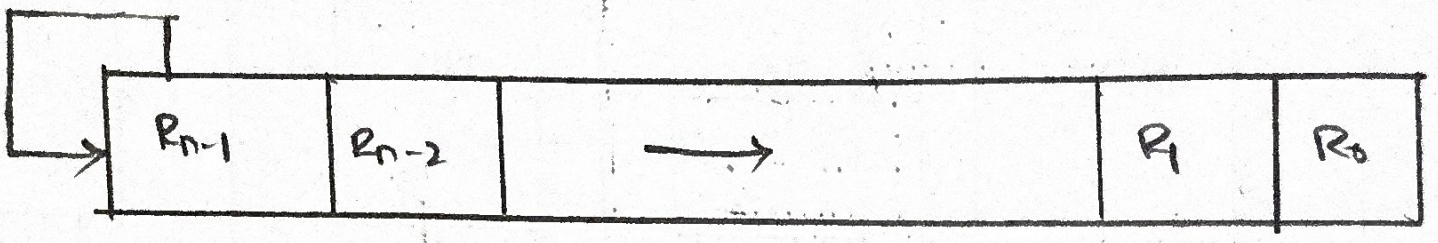
Logic Micro Operations:

Logic Micro operations specify binary operations for strings of bits stored in registers.

$$R_1 \leftarrow R_1 \oplus R_2$$

Shift Micro operation:- Shift micro-operations are used for serial transfer of data. They are also used in conjunction with arithmetic, logic & other data. Processing operations. The contents of a register can be shifted to the left or right.

- A logical shift is one that transfer through the serial input for example $R_1 \leftarrow shl\ R_1$
  $R_2 \leftarrow shl\ R_2$
- The circular shift circulates the bits of the regist around the two shift ends without loss of information.
- An arithmetic shift is a micro operation that shift a binary number to left or right.

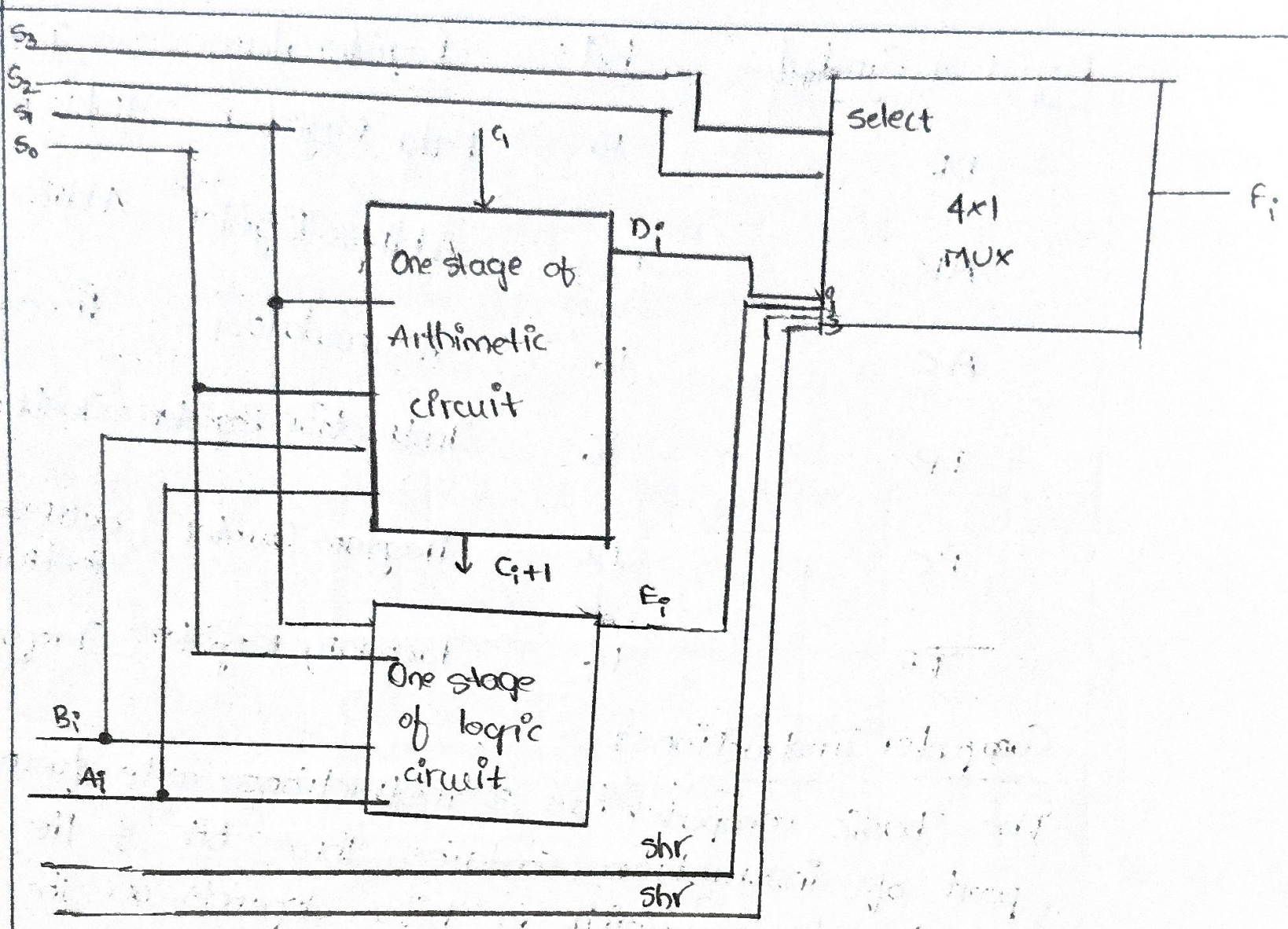


3a) Arithmetic Logic Shift Unit:-

Computer system employes a no. of storage registers connected to a common operational unit called Arithmetic logic shift unit (ALU).

To perform a micro-operation, the contents of specified registers are placed in the inputs of common ALU. The ALU performs an operation & the result of the operation is then transferred to destination register The ALU is a combinational circuit so that the entire register transfer operation from source register through the ALU & into the destination register can be performed during on clock pulse period. The arithmetic, logic shift circuits can be combined into one ALU with common selection variable.

CMREC



The above circuit is specified that it provides eight arithmetic operation, four logic operations & two shift operations.

3b) Computer Registers:-

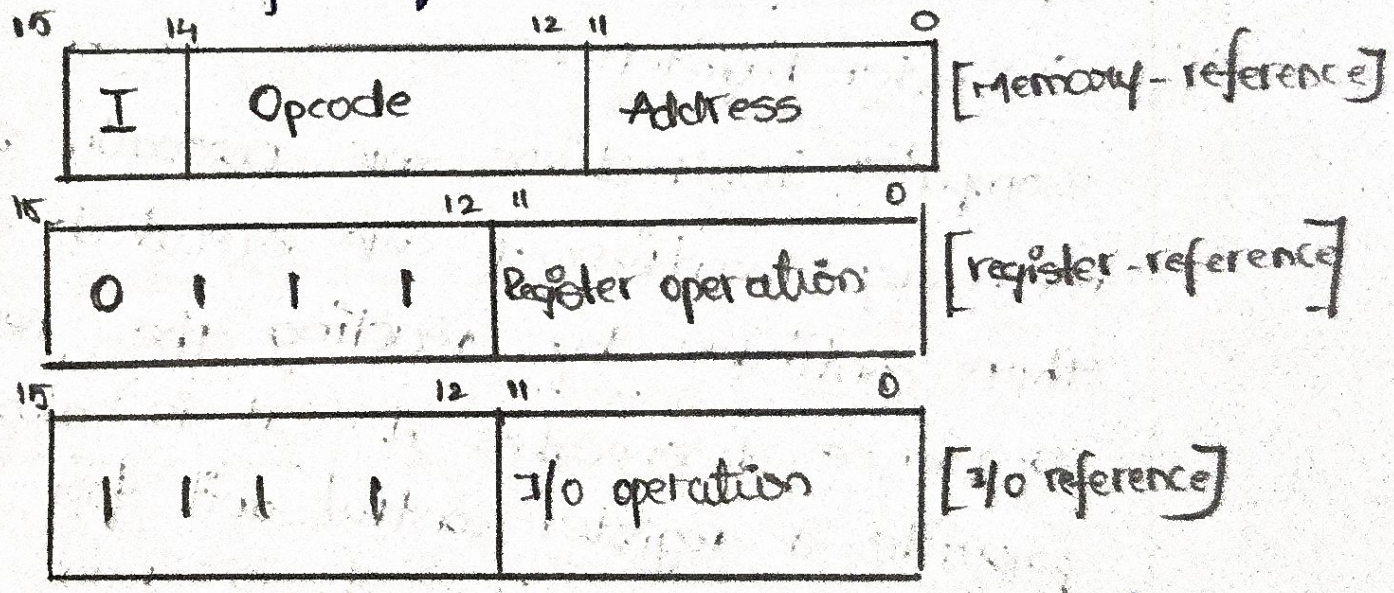
Computer Instructions are normally stored in consecutive memory locations & are executed one at a time. It then continues by reading the next instructions in sequence & executes it. It is also necessary to provide a register control unit for storing the instruction code after it is read from memory. The computer needs processor registers for manipulating data & a register for holding a memory address.

Register Symbol	Bits	Register Name	Function
DR	16	Data Register	Holds memory op
AR	12	Address Register	Address for mem
AC	16	Accumulator	Processor regist
IR	16	Instruction Register	Instruction co
PC	12	Program Counter	address of next instructor
TR	16	Temporary Register	Temporary stora

4

Computer Instructions:

The basic computer has 3 instruction code formate part of instruction. Contains three bits of the meaning of remaining 13 bits depends on the operation code encountered. A memory reference uses 12 bits & 1 bit is reserved for mode. If I is equal to 0 for direct address & 1 for indirect address



Similarly the I/O instruction does not need a reference the memory & is recognized by the operation code 111 with a 1 in left most bit of the instruction. A

register reference instruction specifies an operation on or test of the AC register. An operand from memory is not needed, therefore the 12 bits are used to specify the operations to be executed.

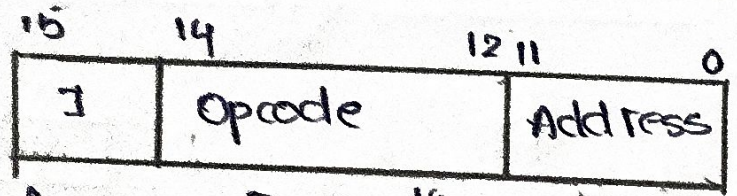
4- Instruction format:-

There are 3 types of Instruction format

i) Memory- Reference instruction
ii) Register - Reference Instruction
iii) Input- Output Reference Instruction

1)→ Memory- Reference Instruction:-

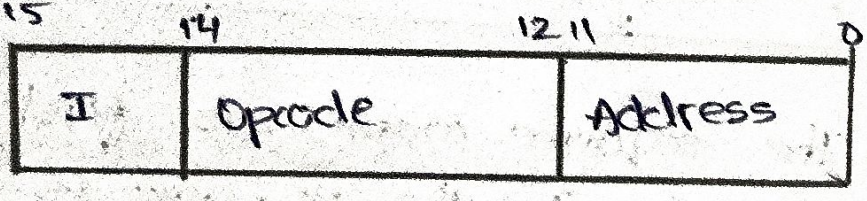
A memory reference instruction uses 12 bits to specify an address & one bit to specify the addressing mode. If 15th bit is 1 then indirect address if other direct address.



(op code =000-110)

2)→ Register - Reference Instruction:-

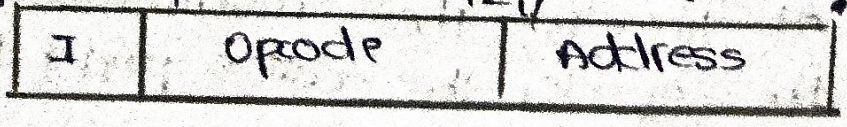
The register reference instructions are recognized by the op code 111 with a 0 in the 15th bit of instruction.



(opcode=111, I=0)

3)→ Input - Output Reference Instruction:-

The input-output reference instruction is recognised by the op code 111 with 1 in the leftmost bit of the instruction.

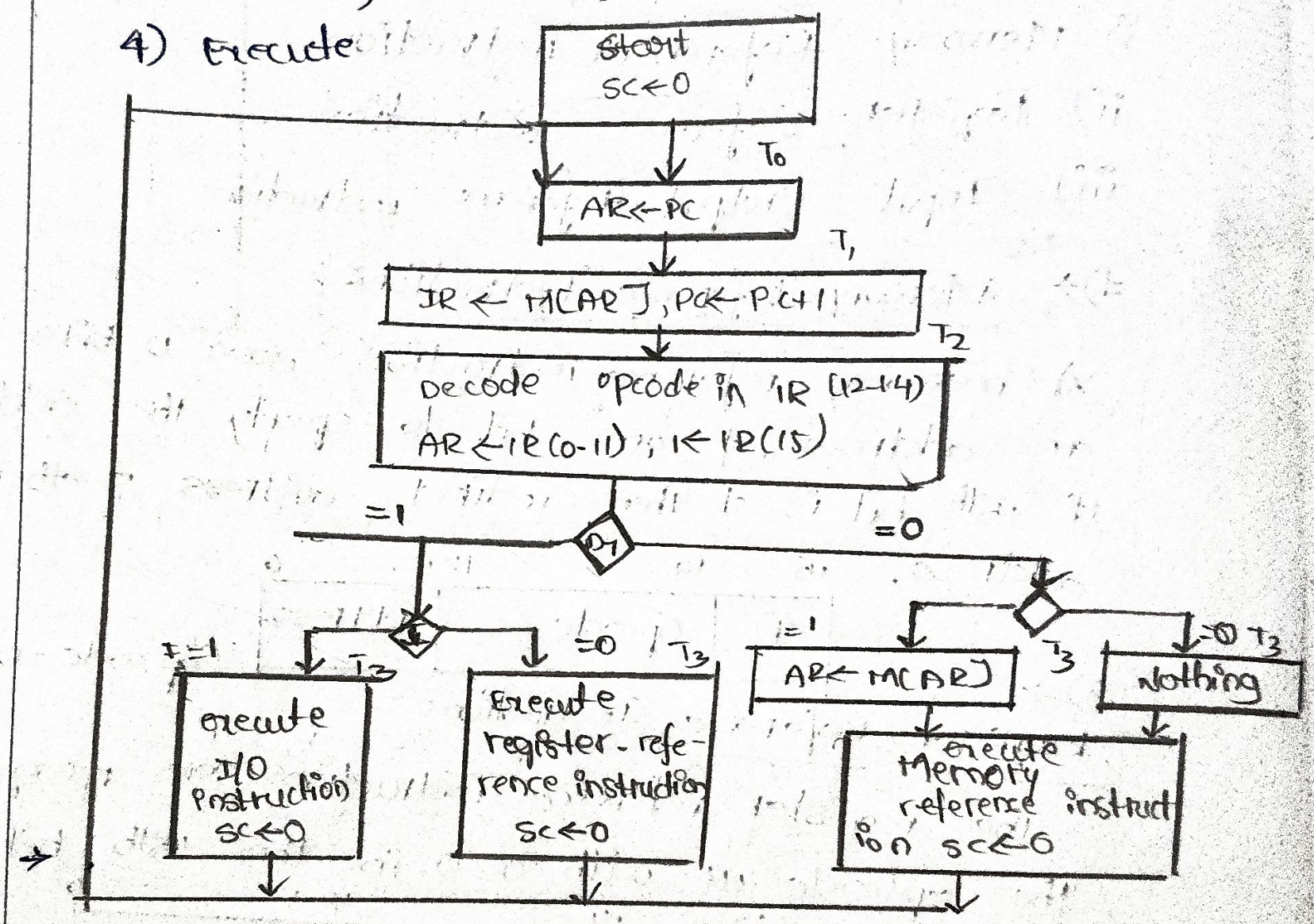


(opcode=111, I=1)

CMREC

Instruction Cycle:-

- The program executed in the computer by going a cycle for each instruction.
- The process of fetching, decoding & executing the instruction is called instruction cycle.
- The instruction cycle consists of following phases:
  1) Fetch 2) Decode 3) Read effective-address
  4) Execute



→ The sequence counter is first initialized to 0.
→ At time T_0, the PC value is loaded into the AR
→ At time T_1, the instruction is fetched from memory into the IR & the PC is incremented.
→ At time T_2, the instruction is decoded to identify the opcode, the address part & the indirect bit.

→ If the D7 is equal to 1, the instruction non-memory instruction.

→ If the I/O bit is 0, a register-reference instruction is executed & the SC is cleared to 0.

→ If the indirect bit is 1, the effective address is obtained from memory into AR.

→ If the indirect bit is 0, the direct address is already available in the AR.

→ Finally, the memory-reference instruction is executed at time T_3 & the SC is cleared to 0.

5a) Addressing Modes:-

→ They are used in order to find out "effective address".

→ The operation field of an instruction specifies the operation to be performed

→ The addressing mode specifies a rule for interpreting/modifying the address field of the instruction

→ The CPU of a computer is designed to through an instruction cycle.

1) Fetch 2) Decode 3) Execute

Types of Addressing Modes:-

- Implied mode
- Immediate mode
- Register mode
- Auto increment & Auto decrement
- Direct address mode
- Indirect address mode

- ~~Indirect~~ Indexing addressing mode
- Relative address mode
- Base register addressing mode.

- Implied mode:-

The operand is implicitly available in the definitions instruction.

→ There is no need to specify operand explicitly

- Immediate mode:-

→ An immediate mode instruction has an operand field rather than a address field.

→ Immediate mode instructions are useful for initialize register to constant value.

- Register Addressing Mode:-

→ The corresponding, register contains the content of the accumulator.

→ The particular register is selected from an n-bit field to specifies ~~x~~ register.

- Register Indirect mode:-

→ Register contains an effective address.

→ the effective address contain address of the operand in memory.

- Auto Increment & Auto decrement:-

→ Similar to register indirect mode, it follows post increment, 1st the content will be accessed & after it's increment.

→ It's similar to register indirect address, it follows pre decrement, 1st decrementation after the content will be accessed

Direct Address mode:-

→ the instruction specifies the memory address where the ~~effective~~ address of the operand is located

Indirect address mode:-

→ the instruction gives a memory address that contains the effective address of operand.

• Index addressing mode:-

The operand's address is calculated by adding a constant to a base register. The content of index register is added to the address part of the instruction to obtain effective address.

• Relative addressing mode:-

The effective address is obtained by addressing the effect to program counter.

• Base Register Address mode:-

The operand is stored in a CPU register mentioned in the instruction.

The content of a base register is added to address part of the instruction to obtain the effective address.

5b) Address Sequencer:-

The basic components of a micro-programmed control unit are the central memory & the circuits that select the next address. the address selection part is known as address sequencing.

→ The central memory stores the micro instructions that define control signals for the CPU.

→ the CAR holds the address of next micro instruction to be executed.

→ the incrementer increases the CAR value to a point to the next sequential micro instruction.

→ MUX1 selects the next address for the CAR from multiple sources such as increment of SBR.

→ the SBR stores return address for microprogram subroutines.

→ the load signal allows a value from SBR or external i/p to loaded into the CAR

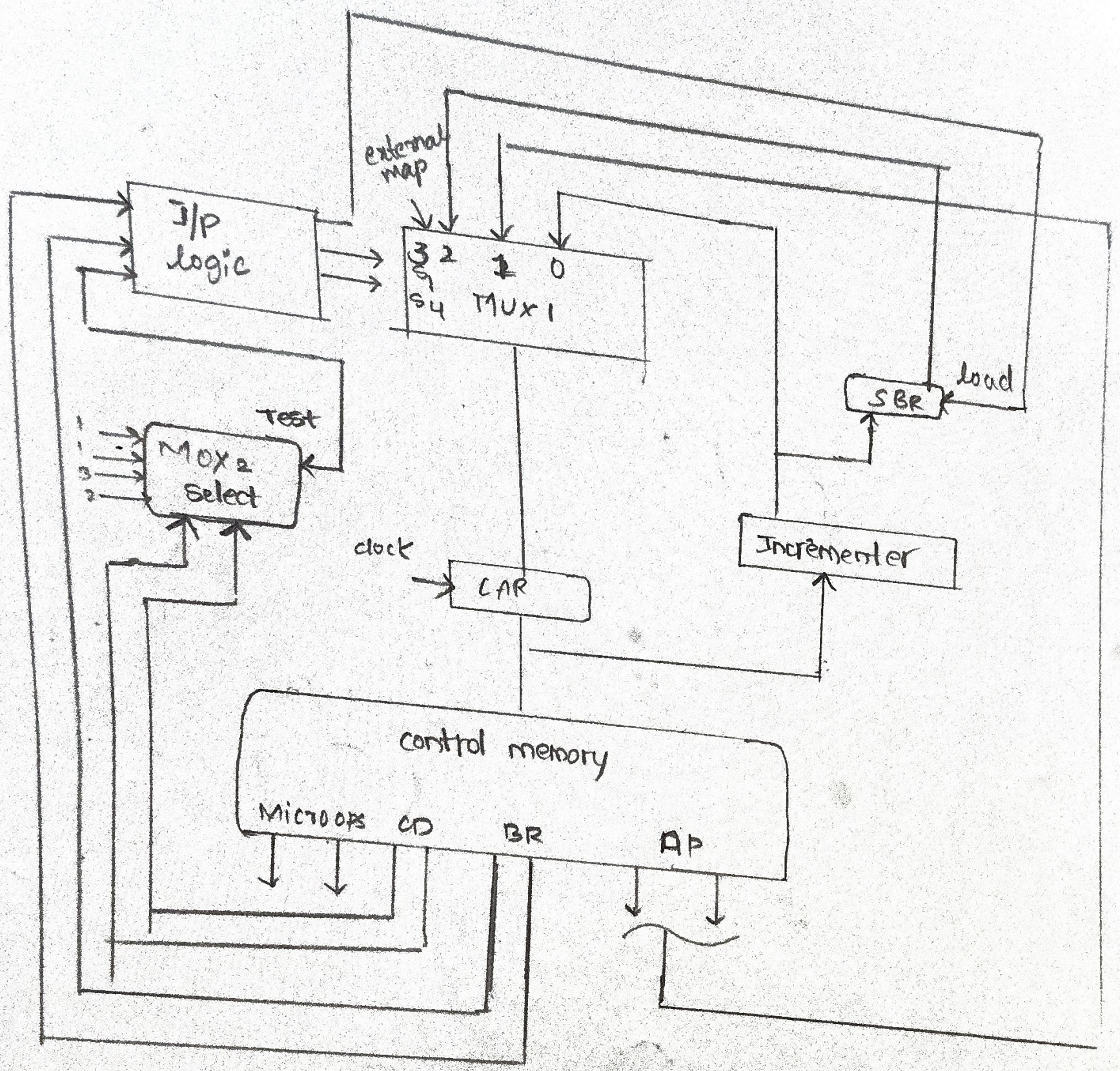
→ MUX2 selects the condition for branching based on status flag such as ST2

→ the test logic checks these condition flags & provides the o/p to control MUX1.

→ the i/p logic generates control signals based on the current instruction & external i/p.

→ the micro-operations from the control memory are sent as control signals to execute CPU functions.

CMREC



Address Sequencer