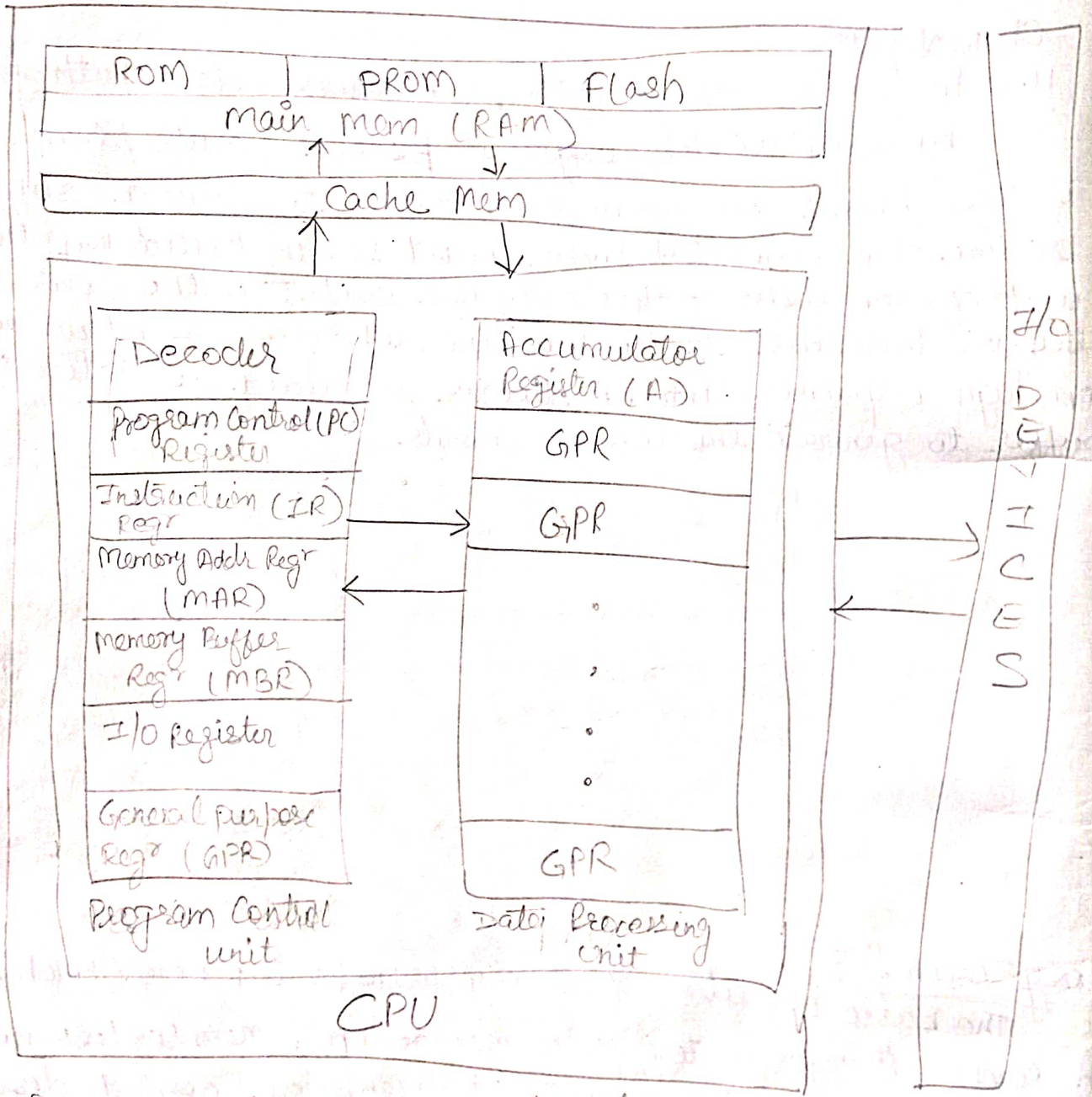
Processor Organization

The processor which is a main part of computer. The execution unit contains a set of registers for storing data and an arithmatic and logic unit for execution of arithmatic and logic operations. The processor can be designed to fulfil the requirements of a set of data paths for specific applications.

The processor has three main parts:

i) Registers ii) ALU iii) Control unit.



Processor & memory architecture of a computer system.

1) Control Unit :- The control unit of the CPU selects and interprets program instructions and then sees that they are executed. Some special purpose registers IR and PC hold the current & next instruction respectively. Decoder is used to decode & interpret the meaning of each and every instruction supported by the CPU. Each instruction is accompanied by microcode which tell the CPU how to execute the instruction.

2) ALU :- The ALU of the CPU is the place where the actual execution of the instruction takes place during the data processing operation.

3) Registers :- Registers are used to hold information on a temporary basis and are part of the CPU (not main memory). The length of a register equals the number of bits it can store & process in a given period, as 8, 16, 32, 64 bits. The size of the register is sometimes called the word size, indicates the amount of data, which can computer can process in a given period.

There are some registers that are common to all computers. The functions of these registors are as follows:

i) Program Counter Register (PC) :- holds the address of the next instruction.

ii) Instruction Register (IR) :- holds the current instruction that is being executed. In IR operation part and address part are seperated. Address part is sent to MAR. Operation part is sent to control section where it is

iii) Memory Address Register (MAR) :- holds the address of active memory loc.

iv) Memory Buffer Register (MBR) :- holds the content of the memory word read from or written in memory.

v) Input / output Register :- Communicates with I/O devices.

vi) Accumulator Register :- holds the initial data to be operated upon, the intermediate results and also the final results of processing opn.

vii) Accumulator Register :-

viii) General purpose Register :- needed for storing pointers, counters, return addresses, temporary results, and partial products during MULT.

ix) Accumulator Register (A) :- It is commonly used to perform Arithmetic & logic operations, the result of an ALL operation is stored in Accumulator

Working of Processor organization

1) CU takes the addr of next program instruction from PC.
2) IR is initated with current instruction comes from PC.
3) IR sends address to MAR and data to MBR.
4) Now CU sends the operation part to the decoder.
5) Decoder interprets the instruction and accordingly CU sends a signal to the ALU.
6) The ALU performs the necessary operation on the data and signals the CU as soon as it has finished.
7) As instruction is executed, next instruction get loaded into PC for execution automatically., and steps 1 to 7 get repeated.

(2)

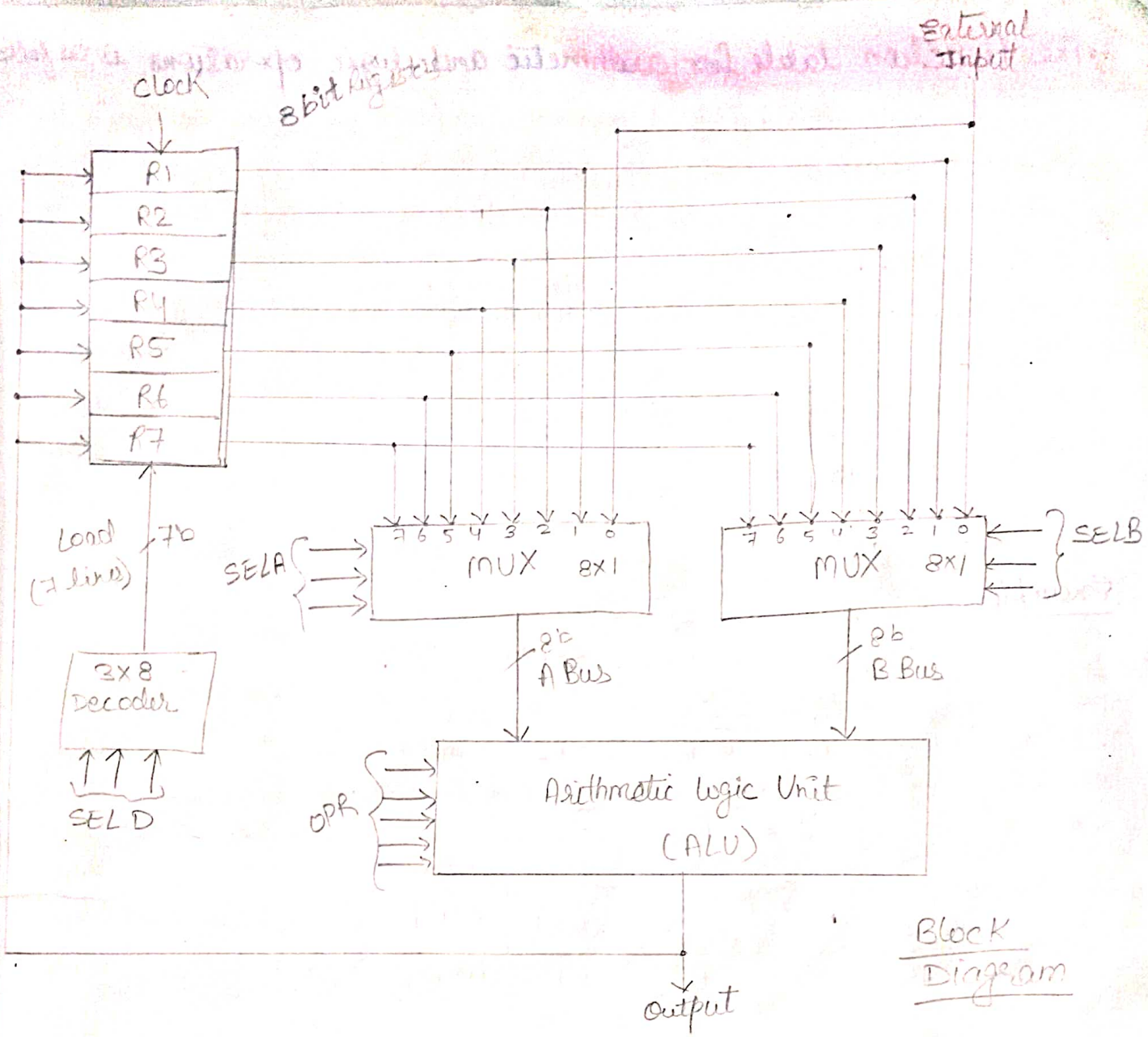
General Register Organization

In this organization, the registers communicate with each other not only for direct data transfers, but also while performing various microoperations: When a large number of registers are included in the CPU, it is most efficient to connect them through a common bus system. Hence it is necessary to provide a common unit that can perform all arithmetic, logic and shift microoperations in the processor.

For example, 7 registers are used for general purpose. Now following addition needs to be performed:

$$R3 \leftarrow R1 + R2$$

i) R1, R2 → source register
ii) R3 → destination register.
iii) MUX A selector (SEL A): to place the contents of R1 into Bus A.
iv) MUX B selector (SEL B): to place the content of R2 into Bus B.
v) ALU operation selector (OPR): to provide the arithmetic addition A+B
vi) Decoder destination selector (SEL D): to transfer the contents of the output bus into R3.



Block Diagram

Control word :- A control word is formatted to perform a specified operation. The control word consist of four fields. Three fields contain three bits each, and one field has four bits. The 13 bit control word when applied to the selection inputs of ALU specify a particular micro operation. General format of control word is as follows:-

3b	3b	3b	4b	= 13 bits.
SELA	SEL B	SEL D	OPR	

Encoding of Register selection is as shown in the following table:-

Binary Code	SELA	SEL B	SEL D
000	Ext. Input	Ext. Inp	None
001	R1	R1	R1
010	R2	R2	R2
011	R3	R3	R3
100	R4	R4	R4
101	R5	R5	R5
110	R6	R6	R6
111	R7	R7	R7

The function table for arithmetic and logic operations is as follows.

OPR Select	operation	Symbol
0 0000	Transfer A	TSF A
1 0001	Increament A	INC A
2 0010	Add A+B	ADD
3 0011	Add with carry A+B+1	ADC
4 0100	Subtract with borrow A-B-1	SBB
5 0101	Subtract A-B	SUB
6 0110	Decreament A(A-1)	DEC A
7 0111	Transfer A	TSF A
8 1000	OR(A∨B)	OR
9 1001	XOR (A⊕B)	XOR
10 1010	AND (A·B)	AND
11 1011	complement A (or) Not A	COM A

Example.

① $R3 \leftarrow R1 + R2$

SELA	SELB	SELD	OPR
001	010	011	0010
for R1	for R2	for R3	for addition (+)

② $R1 \leftarrow R2 - R3$

SELA	SELB	SELD	OPR.
010	011	001	0101
R2	R3	R1	for subtration (-)

③

Stack Organization

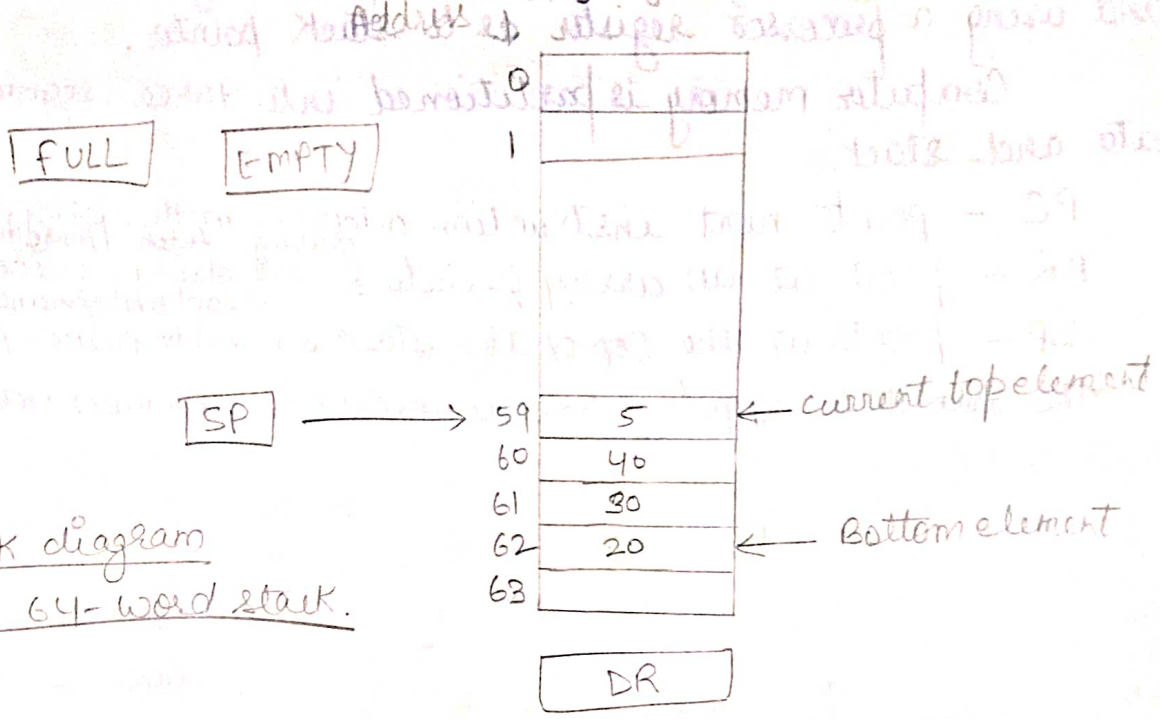
A stack is a storage device that stores information in such a manner that the item stored last is the first item retrieval. Stack is a temporary storage of binary information during the execution of a program. The register that holds the address for the stack is called a stack pointer (SP) because its value always points at the top item in the stack.

Data can be stored on stack by using PUSH operation and remove read from the stack by using POP. According to operations SP increaments or decreaments.

Register Stack :-

A stack can be placed in a portion of logic memory or it can be organized as a collection of finite number of memory words or registers.

Data stored in the main memory of a computer can be organized as a stack, with successive elements occupying successive memory locations.



Block diagram of 64-word stack.

For example, in a 64-word stack, the SP contains 6 bits because $2^6 = 64$. Since SP has only 6 bits, it cannot exceed a number greater than 63 (111111 in binary).

The one bit register FULL = 1 when stack is full and 1b reg^r is EMPTY = 1 when stack is empty of items. DR is the data register that holds the binary data to be written into or read out of the stack.

Initially SP → 0 (Address), EMPTY = 0, FULL = 0.

PUSH operation is implemented with the following sequence of microoperations:

```
SP ← SP+1        Increament SP
M[SP] ← DR       write item on top of the stack
If (SP=0) then (FULL←1)   check if stack is full.
EMPTY ← 0        mark the stack not empty.
```

POP operation consists of the following sequence of microoperations:

```
DR ← M[SP]       read item from the top of stack
SP ← SP+1        Decreament SP
If (SP=0) then (EMPTY←1)   check if stack is empty
FULL ← 0         mark the stack not full.
```

Memory Stack:- The implementation of a stack in the CPU is done by assigning a portion of memory to a stack operation and using a processor register as a stack pointer.

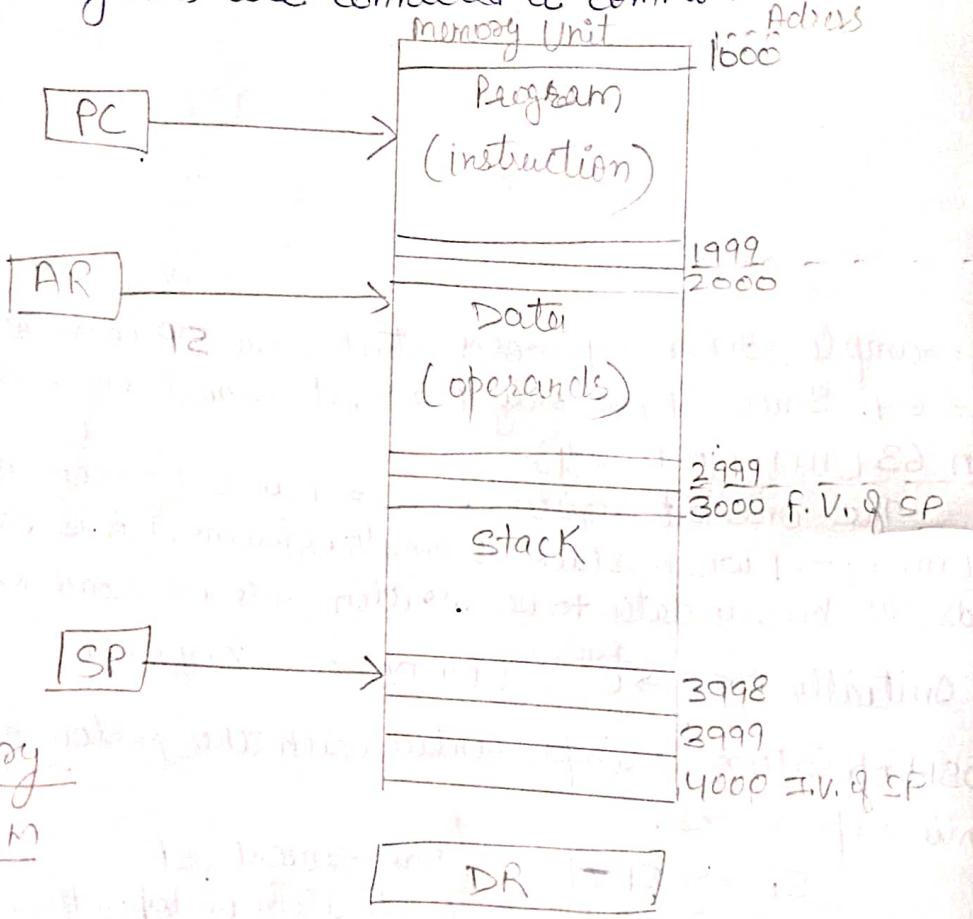
Computer memory is partitioned into three segments: program, data and stack.

PC – points next instruction address in the program & used during fetch phase to read an instruction

AR – point at an array of data & used during execution phase to read an operand

SP – points at the top of the stack & used to push or pop items.

The three registers are connected a common address bus.



Computer Memory Block Diagram

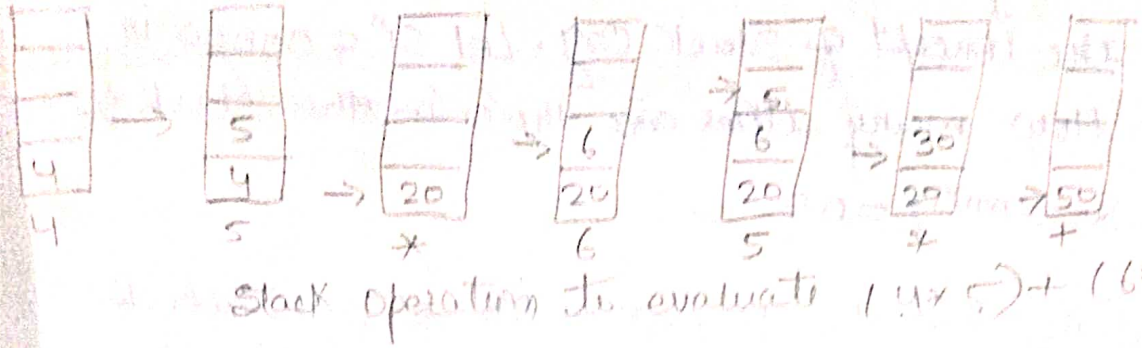
Reverse Polish Notation:- ~~The stack is a reverse operation.~~ Arithmetic operations are ~~implemented~~ represented in three notations:

$w + x$	Infix
$+ wx$	Prefix or Polish notation
$wx +$	postfix or reverse polish notation.

The stack is a reverse operation so using postfix notation.

$(w * x) + (y * z)$	Infix
$wx * yz * +$	postfix
$+ * wx * yz$	prefix

Example: $(4*5)+(6*5)$



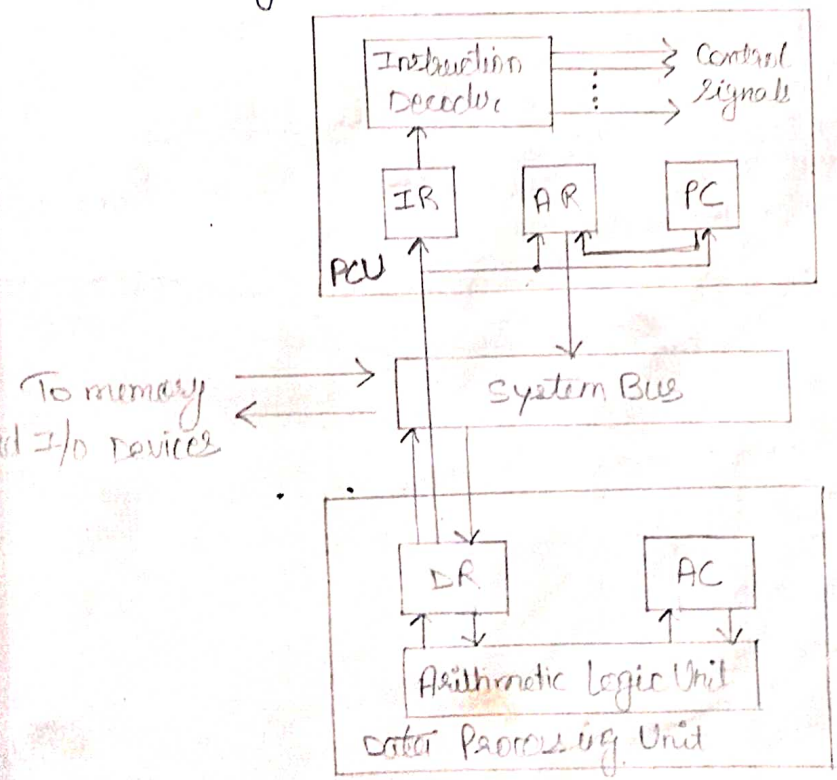
Stack operations to evaluate $(4*5)+(6*5)$.

(1) Single Accumulator organization

In early design, the accumulator was one of the processor registers which is used implicitly in all operations performed by the processor. It has the feature of data storage and addition i.e. any quantity is transferred to an accumulator was automatically added to its previous content.

Instructions are fetch by the program control unit (PCU) using program counter (PC). These instructions are executed in the data processing unit which contains an n-bit arithmetic logic unit (ALU) and accumulator register (AC) and a data register (DR).

There may be more than one such register.



Single accumulator Based Register