CPU ORGANIZATION

Prepared By- Mr. Himmatrao Mane

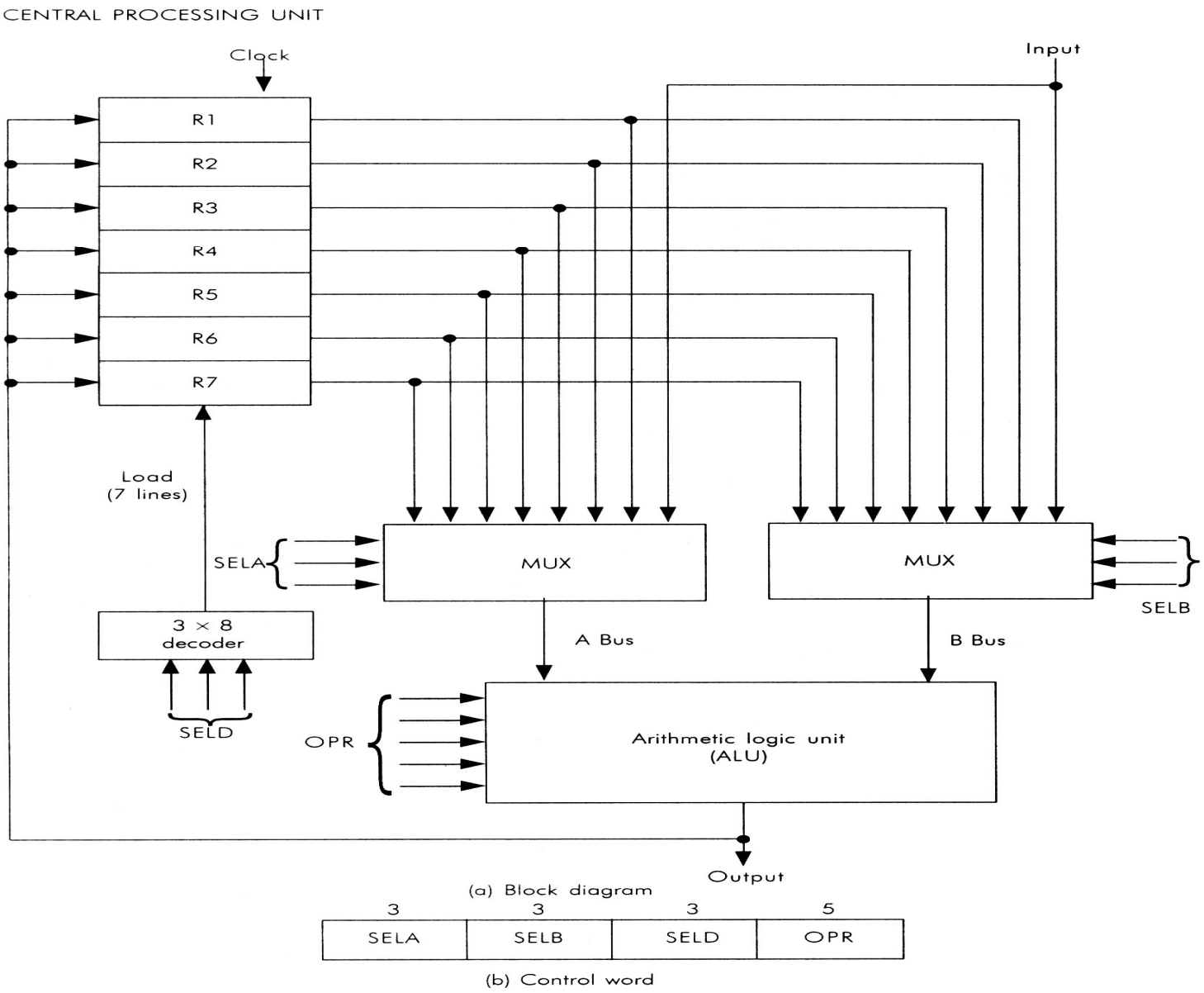
INTRODUCTION

- Intermediate data is stored in the register set during the execution of the instructions.
- The microoperations required for executing the instructions are performed by the arithmetic logic unit whereas the control unit takes care of transfer of information among the registers and guides the ALU.

GENERAL REGISTER ORGANIZATION

- The need for memory locations arises for storing pointers, counters, return address, temporary results and partial products. Memory access consumes the most of the time off an operation in a computer. It is more convenient and more efficient to store these intermediate values in processor registers.

- A common bus system is employed to contact registers that are included in the CPU in a large number. Communications between registers is not only for direct data transfer but also for performing various micro-operations. A bus organization for such CPU register shown in Figure 3.2, is connected to two multiplexers (MUX) to form two buses A and B. The selected lines in each multiplexers select one register of the input data for the particular bus.



Prepared By- Mr. Himmatrao Mane

STACK ORGANIZATION

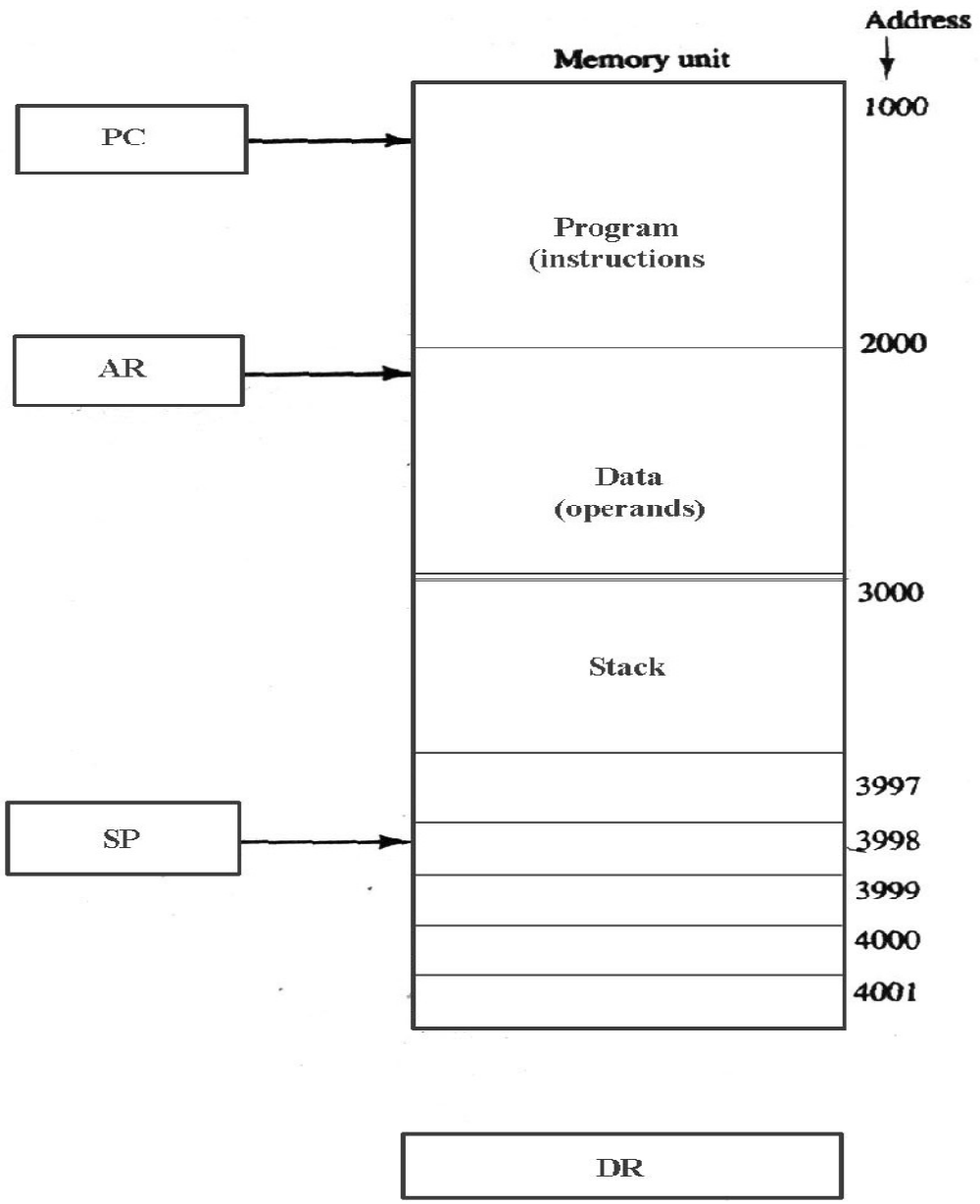
- The CPU of most computers comprises of a stack or last-in-first-out (LIFO) list wherein information is stored in such a manner that the item stored last is the first to be retrieved. The operation of a stack can be compared to a stack of trays. The last tray placed on top of the stack is the first to be taken off.
- The stack in digital computers is essentially a memory unit with an address register that can count only (after an initial value is loaded into it). A Stack Pointer (SP) is the register where the address for the stack is held because its value always points at the top item in the stack. The physical registers of a stack are always available for reading or writing unlike a stack of trays where the tray itself may be taken out or inserted because it is the content of the word that is inserted or deleted.

REGISTER STACK

- There are two ways to place a stack. Either it can be placed in a portion of a large memory or it can be organized as a collection of a finite number of memory words or registers. The organization of a 64-word register stack is exhibited in figure 3.3. A binary number whose value is equal to the address of the word that is currently on top of the stack is contained by the stack pointer register.

MEMORY STACK

- As shown, stack can exist as a stand-alone unit or can be executed in a random-access memory attached to a CPU. The implementation of a stack in the CPU is done by assigning a portion of memory. A portion of memory is assigned to a stack operation and a processor register is used as a stack pointer to execute stack in the CPU. Figure 3.4 shows a portion of computer memory partitioned into three segments -program, data, and stack. The address of the next instruction in the program is located by the program counter PC while an array of data is pointed by address register AR.


Address
Memory unit
PC
1000
Program
(instructions
AR
2000
Data
(operands)
3000
Stack
3997
SP
3998
3999
4000
4001
DR