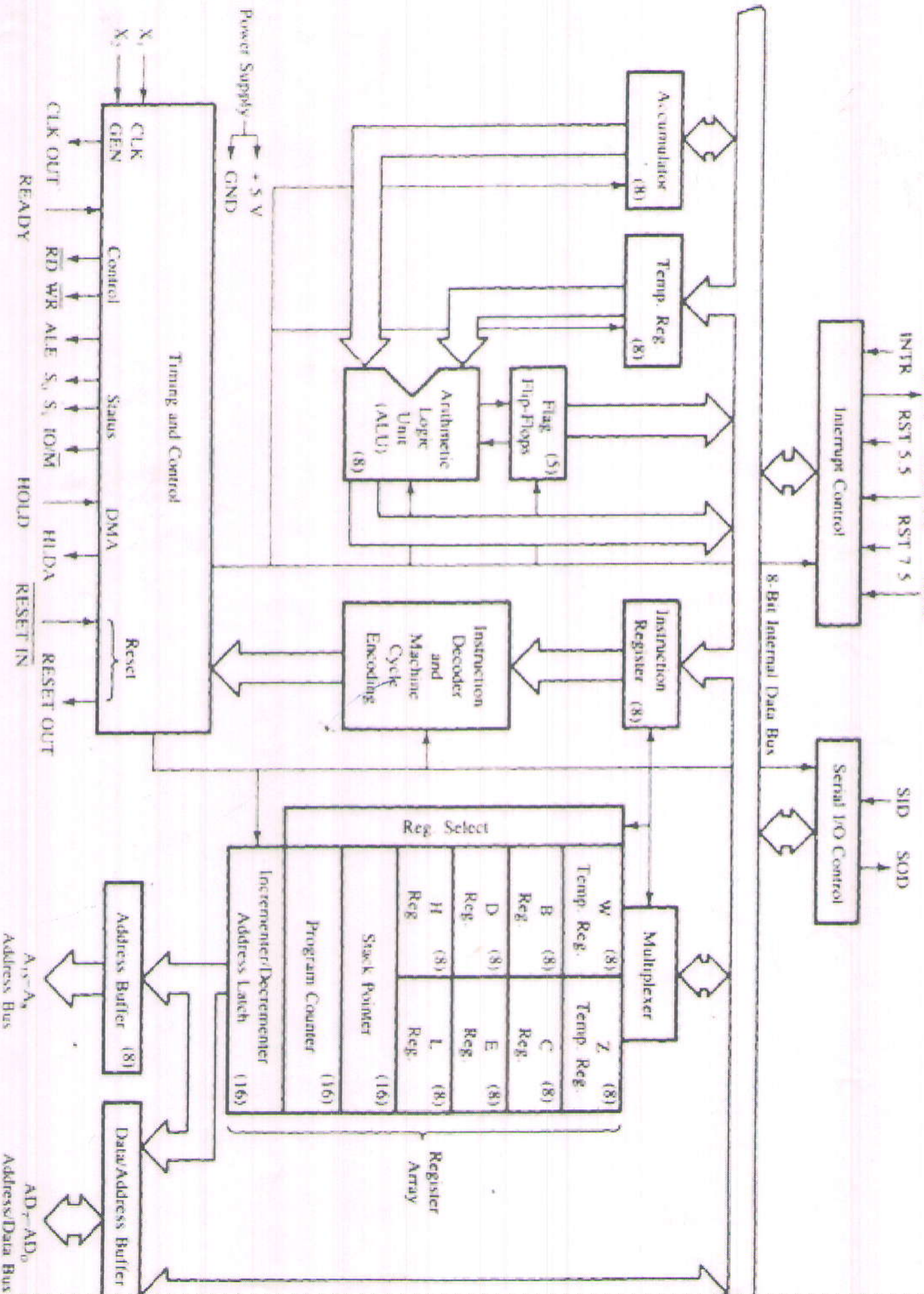
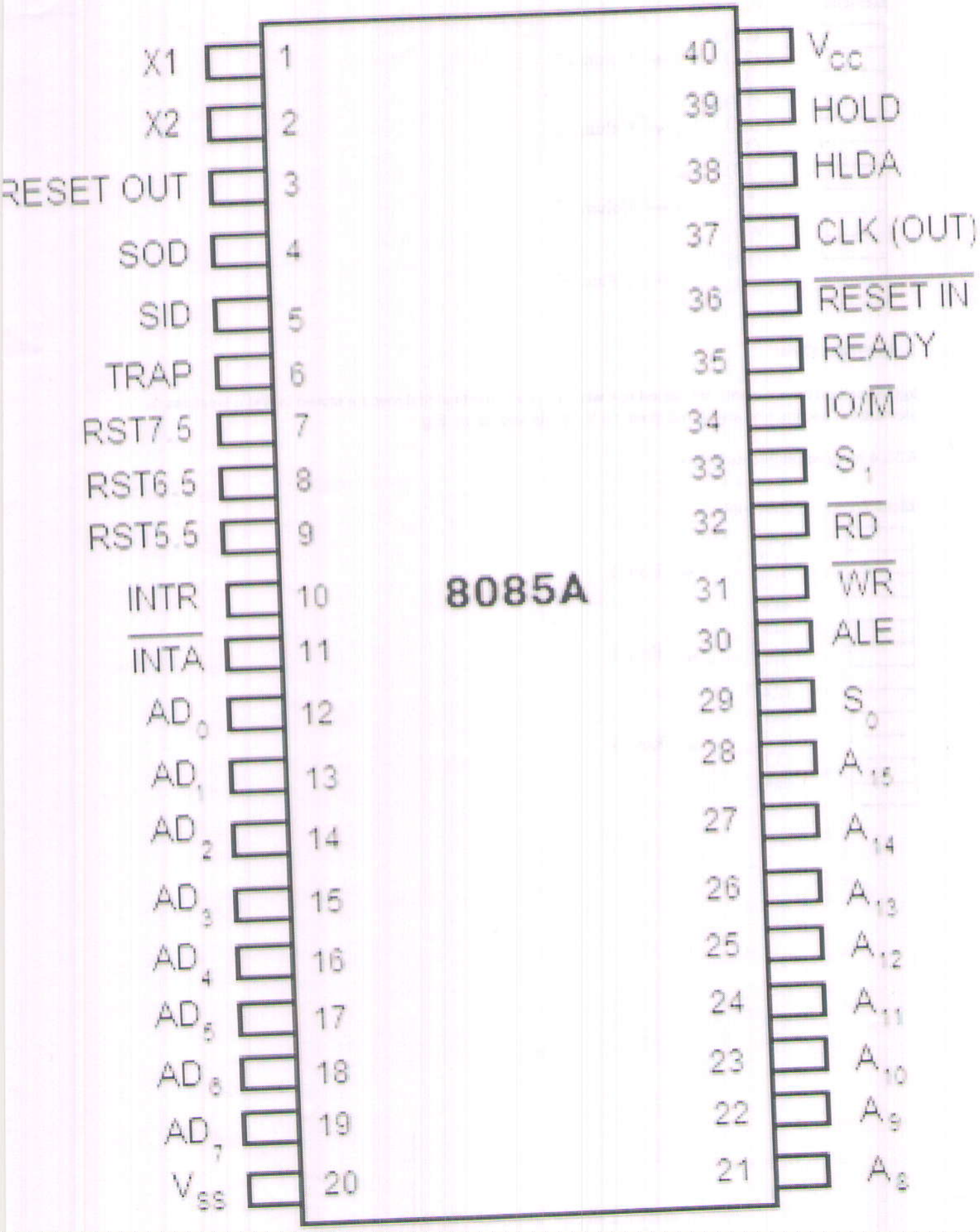
Ans. 1 ~~Pin~~ Functional block diagram of 8085.

EE-208



Ans.1 Pin Out diagram of 8085



Que2 Explain Address bus, Data bus and control bus of 8085 architecture.

System bus is basically a group of communication lines/wires that are responsible for transferring information between different units of the device or peripherals. A typical microprocessor communicates with memory and other devices using three buses: address bus, data bus and control bus.

**Address bus:** Address bus is a unidirectional group of 16 lines i.e. bits flow in one direction from the μP to the peripheral devices. The 8085 μP with it's 16 address lines is capable of addressing 216=65536 (64K) memory locations. Data bus:

**Data bus** carries data in binary form, between microprocessor and peripheral devices as well as memory. It is a group of 8-bits and is bidirectional. Data bus also carries instructions from memory to the microprocessor. Size of the bus therefore limits the number of possible instructions. The 8085 μP has 246 bit patterns amounting to 74 different instructions. These 74 different instructions are therefore called its instruction set.

**Control bus:** The control bus is combination of various single lines that carry control signals. The control lines are not group of lines like address and data bus but are individual lines. Microprocessor generates specific control signals for every operation it performs

Que 3: Write a program to search an element in an array.

```
#include <stdio.h>

#define MAX_SIZE 100  // Maximum array size

int main()
{
    int arr[MAX_SIZE];
    int size, i, toSearch, found;

    /* Input size of array */
    printf("Enter size of array: ");
    scanf("%d", &size);

    /* Input elements of array */
    printf("Enter elements in array: ");
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);
    }

    printf("\nEnter element to search: ");
    scanf("%d", &toSearch);

    /* Assume that element does not exists in array */
    found = 0;

    for(i=0; i<size; i++)
```

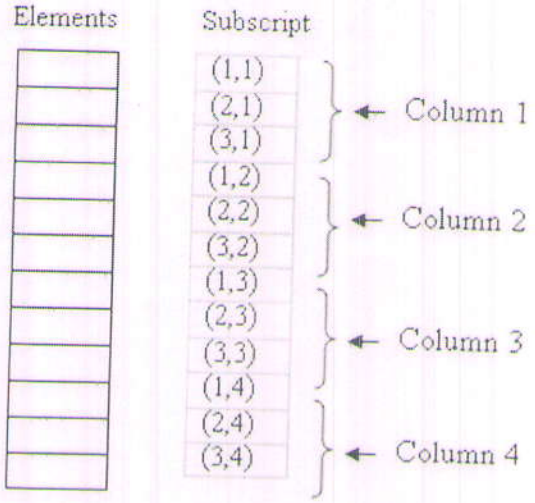
```
    {
        /*
         * If element is found in array then raise found flag
         * and terminate from loop.
         */
        if(arr[i] == toSearch)
        {
            found = 1;
            break;
        }
    }

    /*
     * If element is not found in array
     */
    if(found == 1)
    {
        printf("\n%d is found at position %d", toSearch, i + 1);
    }
    else
    {
        printf("\n%d is not found in the array", toSearch);
    }

    return 0;
}
```

Que 4 explain memory representation of 2-D array.

A 2D array's elements are stored in continuous memory locations. It can be represented in memory using any of the following two ways:

1. Column-Major Order
2. Row-Major Order

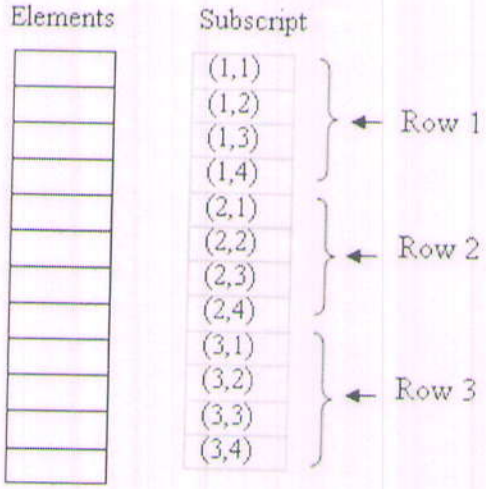
1. Column-Major Order:
In this method the elements are stored column wise, i.e. m elements of first column are stored in first m locations, m elements of second column are stored in next m locations and so on. E.g.
A 3 x 4 array will stored as below:




2. Row-Major Order:

In this method the elements are stored row wise, i.e. n elements of first row are stored in first n locations, n elements of second row are stored in next n locations and so on. E.g.

A 3 x 4 array will stored as below:




Que 5 Define the following:-

**instruction** is (computing) a single operation of a processor defined by an instruction set architecture.

**program** is (computing): a software application, or a collection of software applications, designed to perform a specific task.

**Compiler :** It's a computer program(s) that transforms source code written in a programming language into machine language that is the target language which usually has a binary form known as object code.

**Assembler :** It is a program that takes basic computer instruction(s) and converts then into a pattern of bits that the computer's processor can use to perform it's basic operations. The language used to program the assembler is called assembly language.