Practical1 **Date**

Title:- 1.Write 80386 assembly language program to display your name on screen.
2.Write 80386 assembly language program to add elements of array and print answer.

Basic Programming Model of 80386

The basic programming model consists of these aspects:

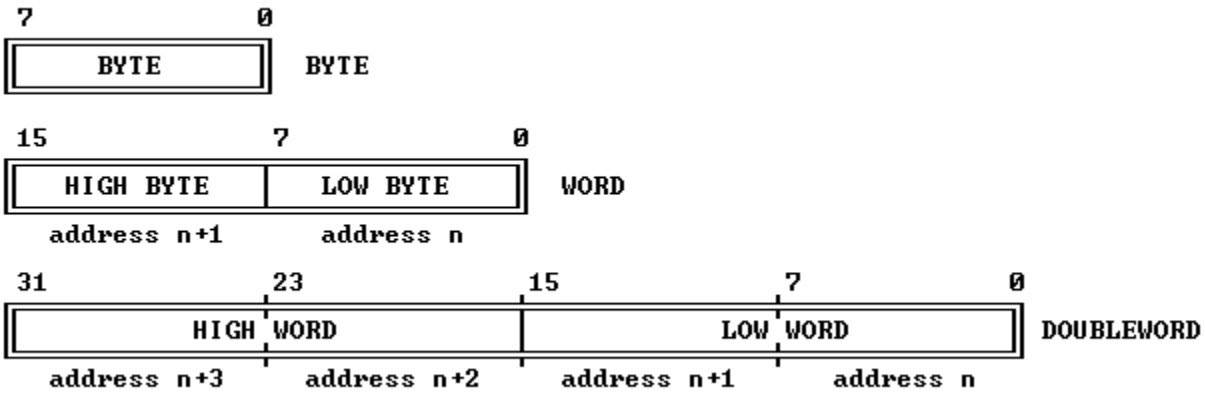
- **Memory organization and segmentation**
- **Data types**
- **Registers**
- **Instruction format**
- **Operand selection**
- **Interrupts and exceptions**

input/output is not included as part of the basic programming model. Systems designers may choose to ke I/O instructions available to applications or may choose to reserve these functions for the operating syste m.

Data types available in 80386

Bytes, words, and doublewords are the fundamental data types.

Figure 2-2. Fundamental Data Types



Byte:-

A byte is eight contiguous bits starting at any logical address. The bits are numbered 0 through 7; bit zero is the least significant bit.

Word:-

A word is two contiguous bytes starting at any byte address. A word thus contains 16 bits. The bits of a word are numbered from 0 through 15; bit 0 is the least significant bit. The byte containing bit 0 of the word is called the low byte; the byte containing bit 15 is called the high byte.

Each byte within a word has its own address, and the smaller of the addresses is the address of the word. The byte at this lower address contains the eight least significant bits of the word, while the byte at the higher address contains the eight most significant bits.

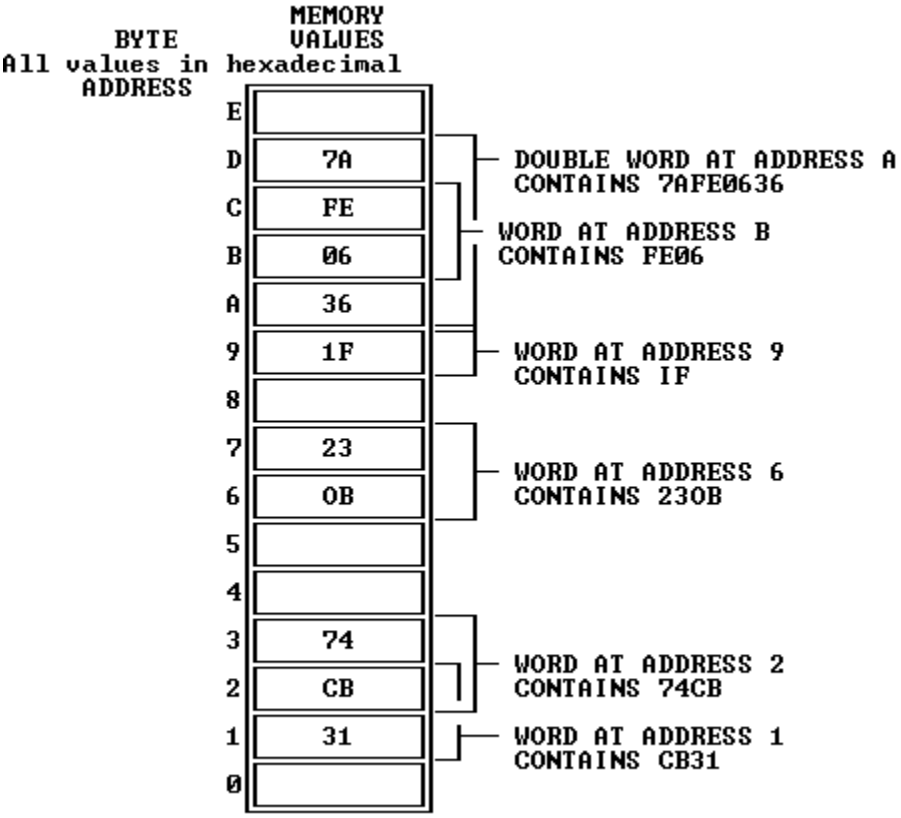
Doublewords

The doubleword is two contiguous words starting at any byte address. A doubleword thus contains 32 bits. The bits of a doubleword are numbered from 0 through 31; bit 0 is the least significant bit. The word containing bit 0 of the doubleword is called the low word; the word containing bit 31 is called the high word.

Each byte within a doubleword has its own address, and the smallest of the addresses is the address of the doubleword. The byte at this lowest address contains the eight least significant bits of the doubleword, while the byte at the highest address contains the eight most significant bits.

The following figure illustrates the arrangement of bytes within words anddoublewords.

Figure 2-3. Bytes, Words, and Doublewords in Memory



Although bytes, words, and doublewords are the fundamental types of operands, the processor also supports additional interpretations of these operands. Depending on the instruction referring to the operand, the following additional data types are recognized:

Integer:

A signed binary numeric value contained in a 32-bit doubleword, 16-bit word, or 8-bit byte. All operations assume a 2's complement representation. The sign bit is located in bit 7 in a byte, bit 15 in a word, and bit 31 in a doubleword. The sign bit has the value zero for positive integers and one for negative. Since the high-order bit is used for a sign, the range of an 8-bit integer is -128 through +127; 16-bit integers may range from -32,768 through +32,767; 32-bit integers may range from -2^(31) through +2^(31) -1. The value zero has a positive sign.

Ordinal:

An unsigned binary numeric value contained in a 32-bit doubleword, 16-bit word, or 8-bit byte. All bits are considered in determining magnitude of the number. The value range of an 8-bit ordinal

number is 0-255; 16 bits can represent values from 0 through 65,535; 32 bits can represent values from 0 through 2^(32) -1.

Near Pointer:

A 32-bit logical address. A near pointer is an offset within a segment. Near pointers are used in either a flat or a segmented model of memory organization.

Far Pointer:

A 48-bit logical address of two components: a 16-bit segment selector component and a 32-bit offset component. Far pointers are used by applications programmers only when systems designers choose a segmented memory organization.

String:

A contiguous sequence of bytes, words, or doublewords. A string may contain from zero bytes to 2^(32) -1 bytes (4 gigabytes).

Bit field:

A contiguous sequence of bits. A bit field may begin at any bit position of any byte and may contain up to 32 bits.

Bit string:

A contiguous sequence of bits. A bit string may begin at any bit position of any byte and may contain up to 2^(32) -1 bits.

BCD:

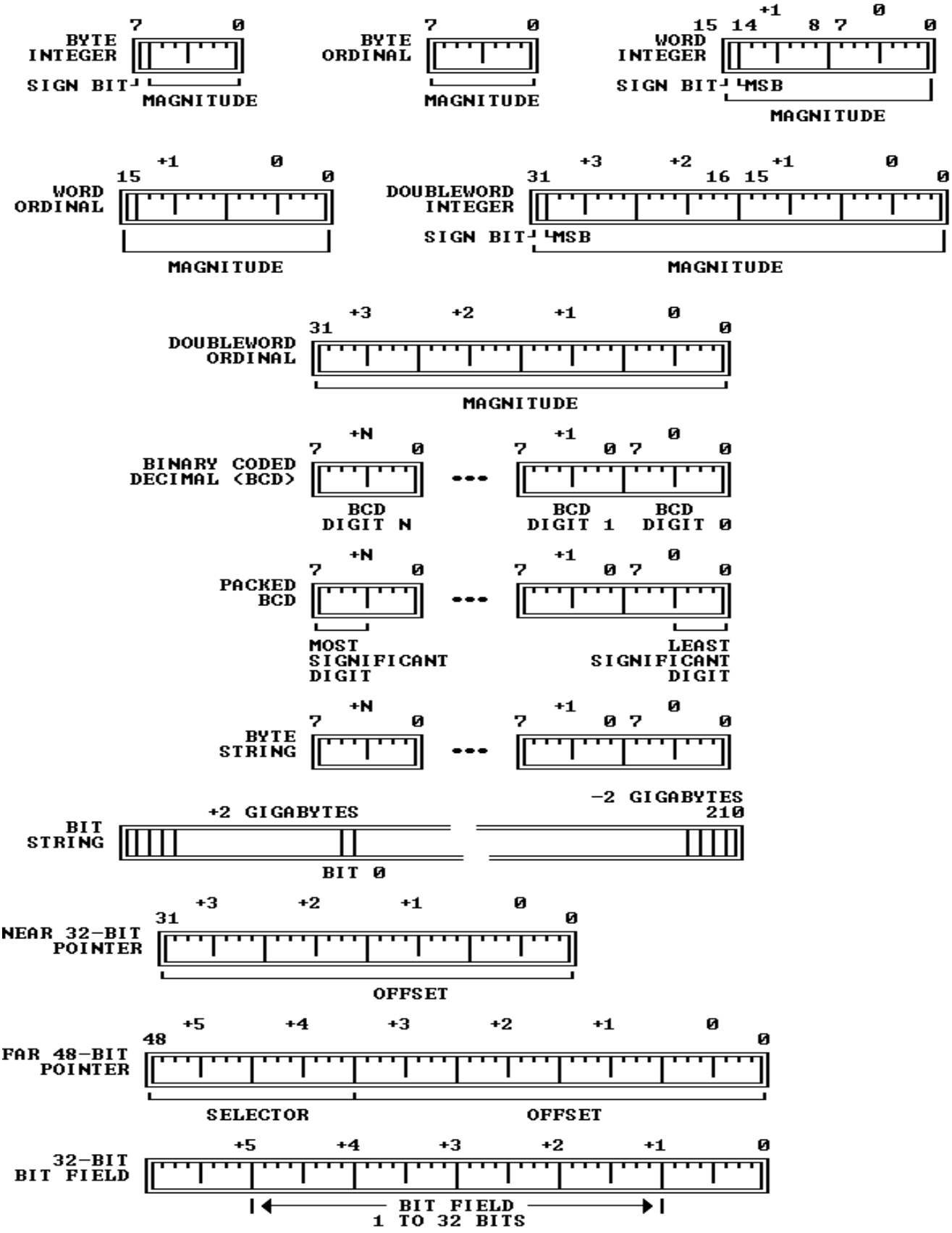
A byte (unpacked) representation of a decimal digit in the range 0 through 9. Unpacked decimal numbers are stored as unsigned byte quantities. One digit is stored in each byte. The magnitude of the number is determined from the low-order half-byte; hexadecimal values 0-9 are valid and are interpreted as decimal numbers. The high-order half-byte must be zero for multiplication and division; it may contain any value for addition and subtraction.

Packed BCD:

A byte (packed) representation of two decimal digits, each in the range 0 through 9. One digit is stored in each half-byte. The digit in the high-order half-byte is the most significant. Values 0-9 are valid in each half-byte. The range of a packed decimal byte is 0-99.

The following figure graphically summarizes the data types supported by the 80386.

Figure 2-4. 80386 Data Types



Data Movement Instructions

MOV (Move) transfers a byte, word, or doubleword from the source operand to the destination operand.

Note:-Explain here what instructions you have used in the program.

Print outs of program code.

Print outs of outputs.