

Assignment-3

Title : Write 64 bit ALP to convert 4-digit Hex number into its equivalent BCD number and 5 digit BCD number into its equivalent HEX number. make your program user friendly to accept the choice from user for

a) HEX to BCD b) BCD to HEX c) EXIT.

Objective : To learn 1) Data representation and conversion
2) Understand the stack operations.

Outcome : Students will study use of stack operations and number conversion in ALP.

Theory :

i) Hexadecimal : In this number system the value of base is 16. This mean that there are only 16 symbols or possible digit values, there are 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F where A,B,C,D,E, and F are single bit representations of decimal values 10,11,12,13,14,15 respectively. It requires only 4 bits to represent value of any digit. Hexadecimal numbers are indicated by the either a 0x prefix or H suffix.

Position of every digit has a weight which is a power of 16. Each position in the Hexadecimal system is 16 times more significant than the previous position, that means numeric value of a

hexadecimal number is determined by multiplying each digit of the number by the value of the position in which the digit appears and then adding the products.

ii) BCD representation

There are two types of BCD representation.

- 1) Unpacked BCD representation.
- 2) Packed BCD representation.

In unpacked BCD representation each byte stores the binary equivalent of a decimal digit. For example the number 1234 is stored as 01, 02, 03 04 H

In packed BCD representation, each digit is stored using four bits. Two decimal digits are packed into a byte. For example the number 1234 is stored as 12 34H

There is no support for multiplication and division in packed BCD representation.

* ASCII Representation.

In ASCII representation decimal numbers are stored as string of ASCII characters. For example the decimal value of 365H is stored as 33 36 35 H

where 33 is ascii for 3, 36 is for 6 and 35 is for 5.

Different number systems can be used with the computer such as hexadecimal, octal, binary and BCD. The number can be converted to any number system from any source to other.

Algorithm :

- 1) Hex to BCD
 - i) Start
 - ii) Move the hexadecimal to any 8 bit register.
 - iii) ~~Div~~ Divide the number by 0AH
 - iv) Push the remainder to stack.
 - v) Check whether the contents of 8 bit register is zero
 - vi) If not then goto step iii)
 - vii) If yes then pop all contents of stack and display the answer.

- 2) BCD to Hex
 - i) Initialize sum=0
 - ii) Accept the first digit multiplicand
 - iii) Multiply the number by 10,000 multiplier.
 - iv) ~~Add result~~ to sum.
 - v) ~~Divide the multiplier~~ by 10.
 - vi) Repeat the step 2 to 4 till multiplier becomes zero
 - vii) Display the sum which is result of converting BCD number to Hex.

Test Cases :

	Description	Expected	Actual	Result
1)	Hex to BCD of 1234 H	BCD no 4660	Same as expected	Pass
2)	BCD to Hex of 65535	Hex no. FFFF	Same as expected	Pass

Conclusion:

We successfully converted no.'s from Hex to BCD and BCD to Hex and implemented learnt about stack operations and MUL and DIV instructions.

B