# UCS1512 – Microprocessors Lab

# Code Conversion

Exp no : 4 Name: Sreedhar V

Date : 18-09-2020 Reg no: 185001161

# AIM:

To program and execute the code conversion of BCD to Hexadecimal and vice verse in 8086 using an emulator.

# BCD to Hexadecimal:

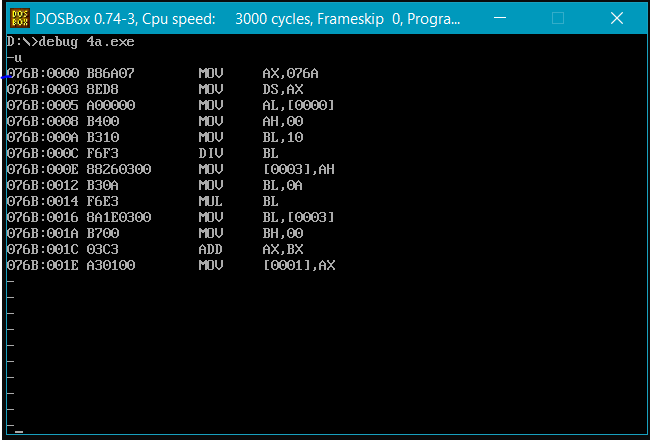
## Algorithm:

* + Program is set to run from any specified memory position.
  + Move the address of data segment to register DS .
  + Move the BCD value to AX register.
  + Move 10h to BL register and divide BL to separate the each value and store the results.
  + Move 0Ah to BL register and multiply BL for conversion(as quotient is already in AL).
  + Move the remainder to BX register .
  + Add AX register and BX register to get final Hexadecimal value.
  + Terminate the program.

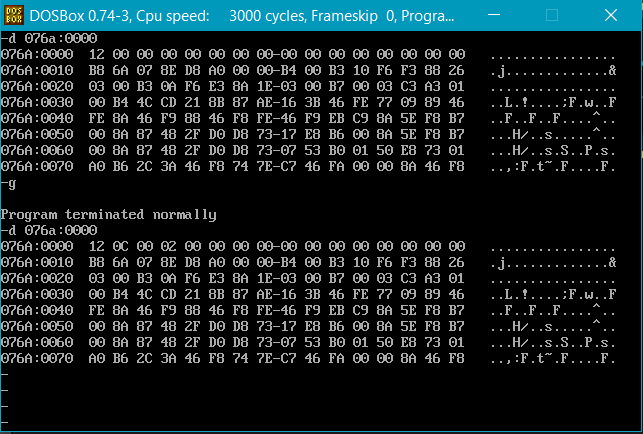
## Program:

|  |  |
| --- | --- |
| CODE | COMMENT |
| Program for BCD to Hexadecimal:  assume code: cs,ds:data  data segment  decimal db 12h  hexa dw 0000h  rem db 00h  data ends  code segment  org 0100h  start : mov ax,data  mov ds,ax  mov al,decimal  mov ah,00h  mov bl,10h  div bl  mov rem,ah  mov bl,0ah  mul bl  mov bl,rem  mov bh,00h  add ax,bx  mov hexa,ax  mov ah,4ch  int 21h  code ends  ends start | Data segment is initialized  decimal is initialized and set to 12h.  Code segment begins  Originating address is set to 0100h  Address of the data is transferred to AX , from AX transferred to DS.  Move the decimal to AL  Extend AX by moving 00h to AH  Move 10h to BL register  Divide by BL  Move the AH to rem to store remainder  Move 0ah to BL register  Multiply BL  Move rem to BL  Extend BX by moving 00h to BH  Add AX and BX registers.  Store the result  Program terminates |

Unassembled code:



Execution:



Result:

Conversion of BCD to Hexadecimal is executed and verified using an emulator.

# Hexadecimal to BCD:

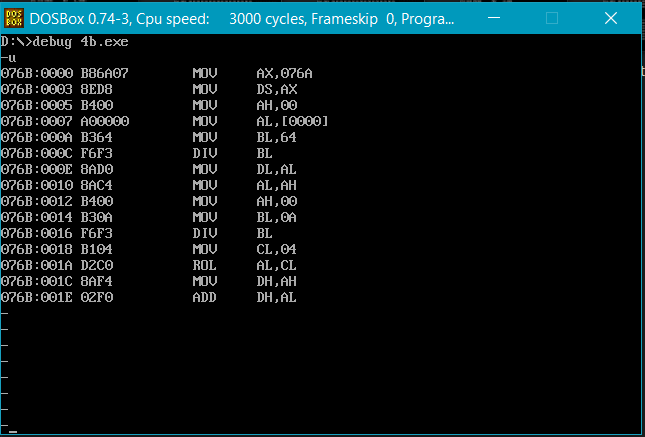
## Algorithm:

* + Program is set to run from any specified memory position.
  + Move the address of data segment to register DS .
  + Move the hexadecimal value to AX register.
  + Move 64h to BL register and divide BL and store the results.
  + Move 0Ah to BL register and divide BL with remainder of the previous operation.
  + Shift left the quotient of the last operation by 4 bits and add remainder to pack into one byte.
  + Store the result.
  + Terminate the program.

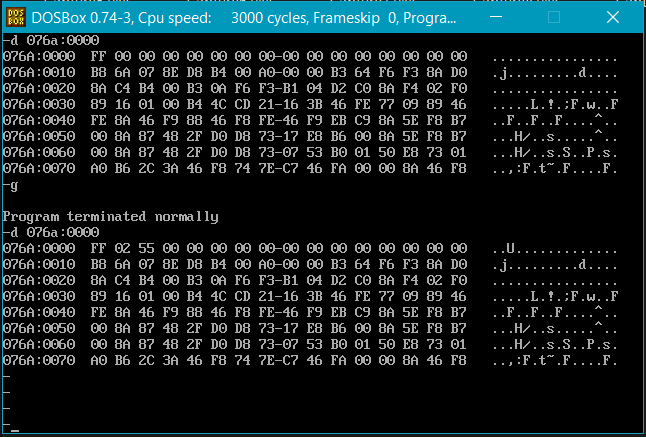
Program:

|  |  |
| --- | --- |
| CODE | COMMENT |
| Program for Hexadecimal to BCD:  assume code: cs,ds:data  data segment  hexa db 0FFh  decimal dw ?  data ends  code segment  org 0100h  start : mov ax,data  mov ds,ax  mov ah,00h  mov al,hexa  mov bl,64h  div bl  mov dl,al  mov al,ah  mov ah,00h  mov bl,0ah  div bl  mov cl,004h  rol al,cl  mov dh,ah  add dh,al  mov decimal,dx  mov ah,4ch  int 21h  code ends  ends start | Data segment is initialized  hexa is initialized and set to FFh  decimal is initialized .  Code segment begins  Originating address is set to 0100h  Address of the data is transferred to AX , from AX transferred to DS.  Move hexa to AX with extension of bits.  Move 64h to BL register.  Divide by BL.  Store the quotient in DL.  Move remainder from AH to AL.  Move 00h to AH for extension.  Move 0Ah to BL  Divide by BL  Shift left by 4 bits in AL register(quotient)  Move the Remainder from AH to DH  Add DH and AL registers  Store the results.  Program terminates |

Unassembled code:



Execution:



Result:

Conversion of hexadecimal to BCD is executed and verified using an emulator.