Experiment No 4: Code Conversion

Date: 14-09-2020 **NAME: Harshini S**
REG.NO: 185001058

A. AIM:

Program for converting BCD to hexadecimal.

ALGORITHM:

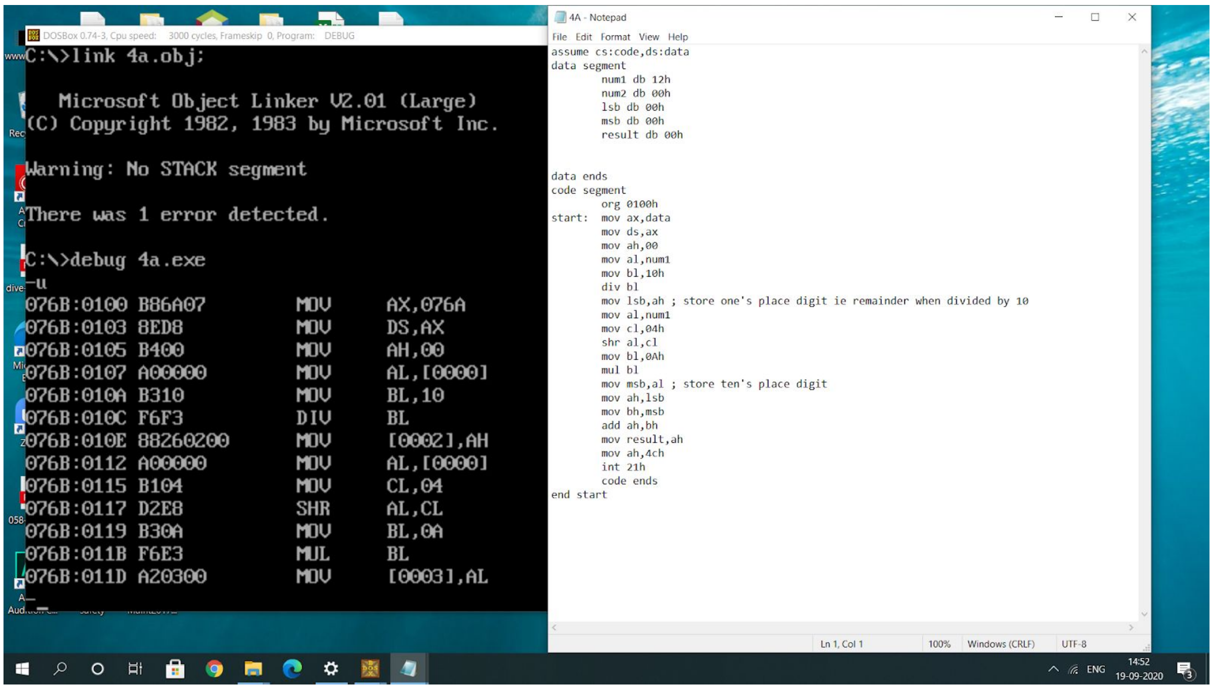
- Initialize the data segment.
- Move data segment address to ds
- Initialize ah with 00h
- Load num1(12) to al and 10h to bl
- Divide ax by bl and move the remainder to variable lsb
- Load al with num1 and shift right 4 bits
- Move 0A to bl and multiply al and bl and store the result in msb variable
- Load lsb to ah and msb to bh and add both
- Store the sum in result
- Terminate the program

PROGRAM:

PROGRAM	**COMMENTS**
mov ax,data mov ds,ax	Load data segment to ds
mov ah,00h	Load ah with 00h
mov al,num1	Load num1 value to al
mov bl,10h	Load 10h to bl
div bl	Divide ax by bl
mov lsb,ah	Move the remainder to lsb
mov al,num1	Load num1 to al
mov cl,04h	Load cl with 04h
shr al,cl	Shift right al by 4 bits

mov bl,0Ah	Load bl with 0A
mul bl	Multiply al by bl
mov msb,al	Store the product in msb
mov ah,lsb	Load ah with lsb value
mov bh,msb	Load bh with msb value
add ah,bh	Add ah and bh
mov result,ah	Store the sum in result variable
mov ah,4ch int 21h	Terminate the program

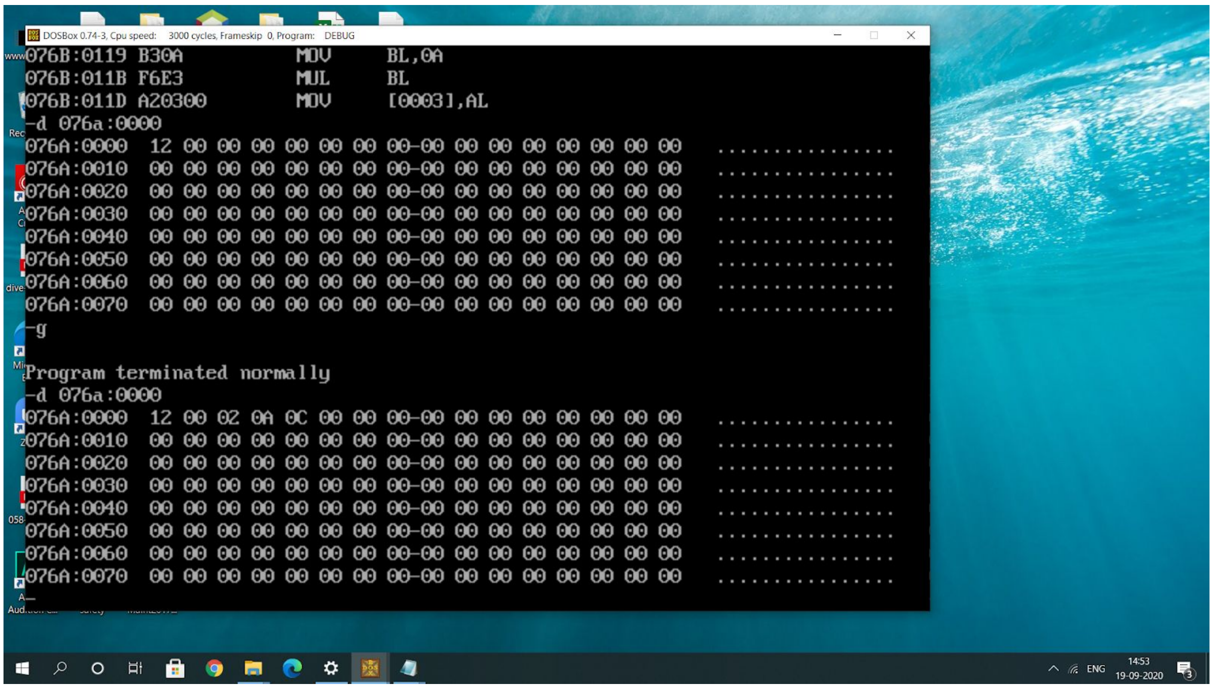
UNASSEMBLED CODE:


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
4A - Notepad
File Edit Format View Help
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8
ENG
14:52
19-09-2020

```
C:\>link 4a.obj;

   Microsoft Object Linker V2.01 (Large)
(C) Copyright 1982, 1983 by Microsoft Inc.

Warning: No STACK segment

There was 1 error detected.

C:\>debug 4a.exe
-u
076B:0100 B86A07        MOV     AX,076A
076B:0103 8ED8          MOV     DS,AX
076B:0105 B400          MOV     AH,00
076B:0107 A00000        MOV     AL,[0000]
076B:010A B310          MOV     BL,10
076B:010C F6F3          DIV     BL
076B:010E 88260200      MOV     [0002],AH
076B:0112 A00000        MOV     AL,[0000]
076B:0115 B104          MOV     CL,04
076B:0117 D2E8          SHR     AL,CL
076B:0119 B30A          MOV     BL,0A
076B:011B F6E3          MUL     BL
076B:011D A20300        MOV     [0003],AL
-
```

```
assume cs:code,ds:data
data segment
        num1 db 12h
        num2 db 00h
        lsb db 00h
        msb db 00h
        result db 00h


data ends
code segment
        org 0100h
start:  mov ax,data
        mov ds,ax
        mov ah,00
        mov al,num1
        mov bl,10h
        div bl
        mov lsb,ah ; store one's place digit ie remainder when divided by 10
        mov al,num1
        mov cl,04h
        shr al,cl
        mov bl,0Ah
        mul bl
        mov msb,al ; store ten's place digit
        mov ah,lsb
        mov bh,msb
        add ah,bh
        mov result,ah
        mov ah,4ch
        int 21h
        code ends
end start
```

SAMPLE INPUT/OUTPUT:


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
076B:0119 B30A          MOV     BL,0A
076B:011B F6E3          MUL     BL
076B:011D A20300        MOV     [0003],AL
-d 076a:0000
076A:0000  12 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-g

Program terminated normally
-d 076a:0000
076A:0000  12 00 02 0A 0C 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-
```

RESULT:

Thus BCD has been converted to hexadecimal.

B. AIM:

Program for converting hexadecimal to BCD.

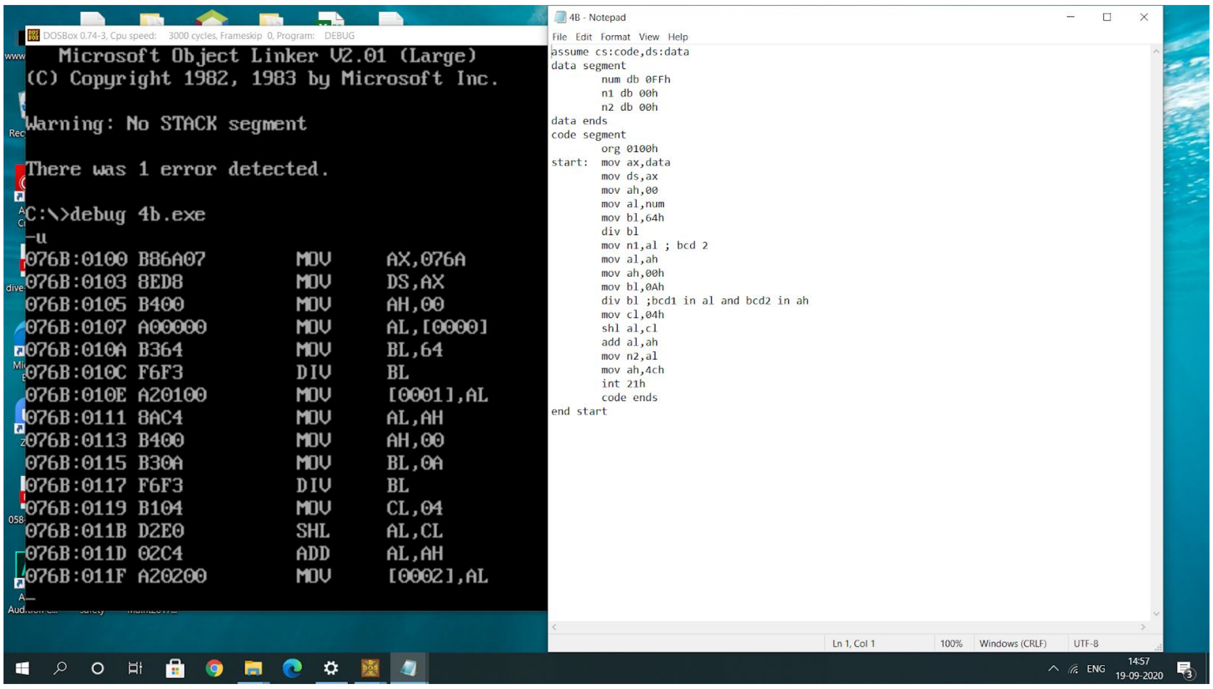
ALGORITHM:

- Initialize the data segment
- Load ah with 00h
- Load value of num(0FFh) to al and 64h to bl
- Divide ax by bl
- Store quotient(al) in n1 and remainder(ah) in al
- Load ah with 00h
- Load bl with 0Ah
- Divide ax by bl
- Load cl with 04h and shift left al by four bits
- Add al and ah
- Move al(sum) to n2
- Terminate the program

PROGRAM:

PROGRAM	COMMENTS
mov ax,data mov ds,ax	Load data segment to ds
mov ah,00h	Load ah with 00h
mov al,num	Load al with num
mov bl,64h	Load bl with 64h
div bl	Divide ax by bl
mov n1,al	Load n1 with al
mov al,ah	Load ah to al
mov ah,00h	Load ah with 00h
mov bl,0Ah	Load bl with 0Ah
div bl	Divide ax by bl
mov cl,04h	Move cl with 04h
shl al,cl	Shift left al by 4 bits
add al,ah	Add al and ah
mov n2,al	Load n2 with al
mov ah,4ch int 21h	Terminate the program

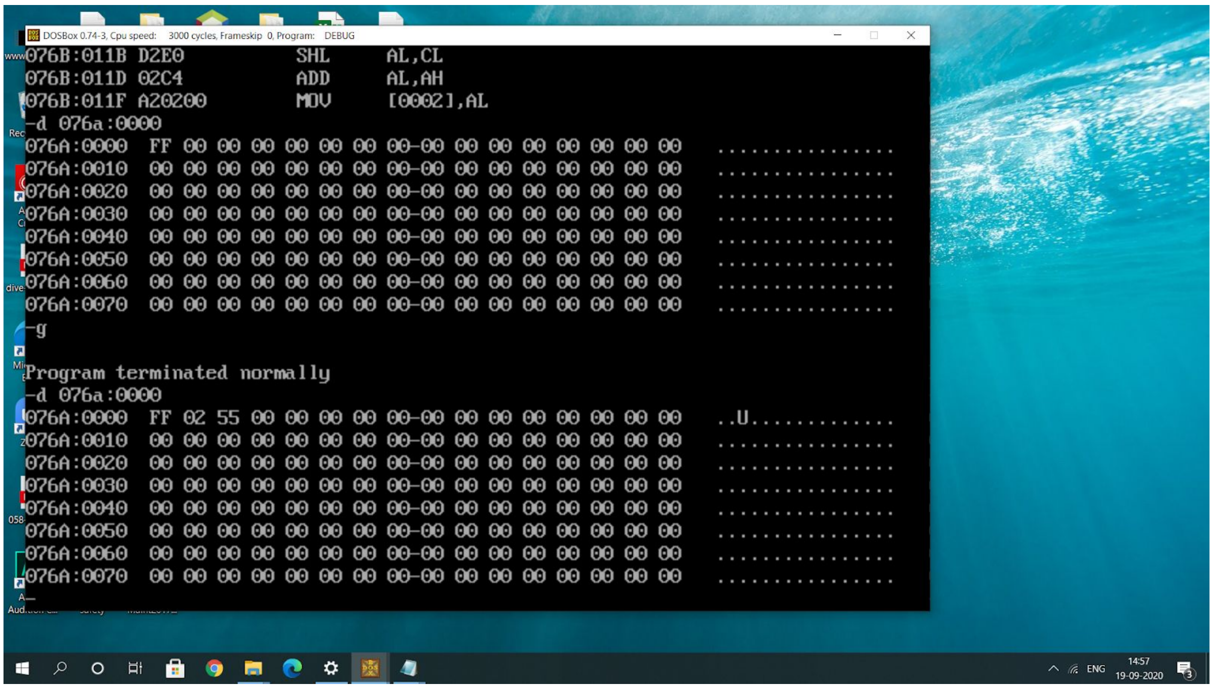
UNASSEMBLED CODE:



```
 Microsoft Object Linker V2.01 (Large)
(C) Copyright 1982, 1983 by Microsoft Inc.

Warning: No STACK segment

There was 1 error detected.

C:\>debug 4b.exe
-u
076B:0100 B86A07        MOV     AX,076A
076B:0103 8ED8          MOV     DS,AX
076B:0105 B400          MOV     AH,00
076B:0107 A00000        MOV     AL,[0000]
076B:010A B364          MOV     BL,64
076B:010C F6F3          DIV     BL
076B:010E A20100        MOV     [0001],AL
076B:0111 8AC4          MOV     AL,AH
076B:0113 B400          MOV     AH,00
076B:0115 B30A          MOV     BL,0A
076B:0117 F6F3          DIV     BL
076B:0119 B104          MOV     CL,04
076B:011B D2E0          SHL     AL,CL
076B:011D 02C4          ADD     AL,AH
076B:011F A20200        MOV     [0002],AL
```

```
assume cs:code,ds:data
data segment
        num db 0FFh
        n1 db 00h
        n2 db 00h
data ends
code segment
        org 0100h
start:  mov ax,data
        mov ds,ax
        mov ah,00
        mov al,num
        mov bl,64h
        div bl
        mov n1,al ; bcd 2
        mov al,ah
        mov ah,00h
        mov bl,0Ah
        div bl ;bcd1 in al and bcd2 in ah
        mov cl,04h
        shl al,cl
        add al,ah
        mov n2,al
        mov ah,4ch
        int 21h
        code ends
end start
```

SAMPLE INPUT/OUTPUT



```
076B:011B D2E0          SHL     AL,CL
076B:011D 02C4          ADD     AL,AH
076B:011F A20200        MOV     [0002],AL
-d 076a:0000
076A:0000  FF 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-g

Program terminated normally
-d 076a:0000
076A:0000  FF 02 55 00 00 00 00 00-00 00 00 00 00 00 00 00   ..U.............
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
```

RESULT:

Thus hexadecimal has been converted to BCD.