

Experiment 7

Class: SE Comp

Year: 2020-21

Performed by: Danyl Fernandes, 72

Hex to BCD:

Code:

```
Name hex_to_bcd
title code for hex to bcd
assume cs:code, ds: data
data segment
    array db 10 DUP(0)
    no dw 09f7h
    data ends
code segment
start:
    mov dx, data
    mov ds, dx
    lea si, array
    mov ax, no
    mov bl, 0ah
BACK:cmp ax, 0ah
    jb next
    div bl
    inc si
    mov [si],ah
    mov ah,00
    jmp BACK
next:inc si
    mov [si],al
Print:mov dl,[si]
    add dl, 30h
    mov ah, 02
    int 21h
    dec si
    jnz Print
    mov ah,4ch
    int 21h
code ends
end start
end
```

Output:

emu8086 - assembler and microprocessor emulator 4.08

file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help about

```
01 Name hex_to_bcd
02 title code for hex to bcd
03 assume cs:code, ds:data
04 data segment
05 array db 10 DUP(0)
06 no dw 09f7h
07 data ends
08 code segment
09 start:
10 mov dx, data
11 mov ds, dx
12 lea si, array
13 mov ax, no
14 mov bl, 0ah
15 BACK:cmp ax, 0ah
16 jb next
17 diu bl
18 inc si
19 mov [si],ah
20 mov ah,00
21 jmp BACK
22 next:inc si
23 mov [si],al
24 Print:nov dl,[si]
25 add dl,30h
26 mov ah,02
27 int 21h
28 dec si
29 jnz Print
30 mov ah,4ch
31 int 21h
32 code ends
33 end start
34 end
35 ret
```

emulator: hex_to_b.exe_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers	H	L
AX	4C	31
BX	00	0A
CX	00	3E
DX	07	31
CS	F400	
IP	0204	
SS	0710	
SP	FFFA	
BP	0000	
SI	0000	
DI	0000	
DS	0710	
ES	0700	

F400:0204				F400:0204			
F4200:	FF	255	RES	BIOS	DI		
F4201:	FF	255	RES	INT	021h		
F4202:	CD	205	=	IRET			
F4203:	21	033	!	ADD	[BX + SI], AL		
F4204:	0F	207	!	ADD	[BX + SI], AL		
F4205:	00	000	NULL	ADD	[BX + SI], AL		
F4206:	00	000	NULL	ADD	[BX + SI], AL		
F4207:	00	000	NULL	ADD	[BX + SI], AL		
F4208:	00	000	NULL	ADD	[BX + SI], AL		
F4209:	00	000	NULL	ADD	[BX + SI], AL		
F420A:	00	000	NULL	ADD	[BX + SI], AL		
F420B:	00	000	NULL	ADD	[BX + SI], AL		
F420C:	00	000	NULL	ADD	[BX + SI], AL		
F420D:	00	000	NULL	ADD	[BX + SI], AL		
F420E:	00	000	NULL	ADD	[BX + SI], AL		
F420F:	00	000	NULL	ADD	[BX + SI], AL		
F4210:	00	000	NULL	ADD	[BX + SI], AL		
F4211:	00	000	NULL	ADD	[BX + SI], AL		
F4212:	00	000	NULL	ADD	[BX + SI], AL		
F4213:	00	000	NULL	ADD	[BX + SI], AL		
F4214:	00	000	NULL	ADD	[BX + SI], AL		

screen source reset aux vars debug stack flags

variables

size: word elements: 1

edit show as: hex

ARRAY	00h
NO	09F7h

emulator screen (56x10 chars)

2551

clear screen change font 0/16

BCD to Hex:

Code:

```
Name bcd_to_hex
title code for bcd to hex
assume cs:code, ds: data
data segment
    array db 10 DUP(0)
    no dw 2551
    data ends
code segment
    start:
    mov dx, data
    mov ds, dx
    mov ax, no
    mov cl, 10h
    lea bx, array
    BACK:cmp ax, 10h
    jb next
    div cl
    inc si
    mov [bx+si],ah
    mov ah,00
    jmp BACK
    next:inc si
    mov [bx+si],al
    Print:mov dl,[bx+si]
    cmp dl, 9
    jbe next1
    add dl, 07h
    next1:add dl, 30h
    mov ah, 02
    int 21h
    dec si
    jnz Print
    mov ah,4ch
    int 21h
code ends
end start
end
```

Output:

emu8086 - assembler and microprocessor emulator 4.08

file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help about

```
01 Name bcd_to_hex
02 title code for bcd to hex
03 assume cs:code, ds:data
04 data segment
05 array db 10 DUP(0)
06 no dw 2551
07 data ends
08 code segment
09 start:
10 mov dx, data
11 mov ds, dx
12 mov ax, 0
13 mov cl, 10h
14 lea bx, array
15 BACK:cmp ax, 10h
16 jb next
17 div cl
18 inc si
19 mov [bx+si],ah
20 mov ah,00
21 jmp BACK
22 next:inc si
23 mov [bx+si],al
24 Print:mov dl,[bx+si]
25 cmp dl,9
26 jbe next1
27 add dl,07h
28 next1:add dl,30h
29 mov ah,02
30 int 21h
31 dec si
32 jnz Print
33 mov ah,4ch
34 int 21h
35 code ends
36 end start
37
38
```

emulator: bcd_to_h.exe

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers	H	L
AX	4C	37
BX	00	00
CX	00	10
DX	07	37
CS	F400	
IP	0204	
SS	0710	
SP	FFFA	
BP	0000	
SI	0000	
DI	0000	
DS	0710	
ES	0700	

F400:0204		F400:0204	
F4200:	FF 255 RES	BIOS DI	
F4201:	FF 255 RES	INT 021h	
F4202:	CD 205 =		
F4203:	21 033 ?		
F4204:	CF 207 ±		
F4205:	00 000 NULL		
F4206:	00 000 NULL		
F4207:	00 000 NULL		
F4208:	00 000 NULL		
F4209:	00 000 NULL		
F420A:	00 000 NULL		
F420B:	00 000 NULL		
F420C:	00 000 NULL		
F420D:	00 000 NULL		
F420E:	00 000 NULL		
F420F:	00 000 NULL		
F4210:	00 000 NULL		
F4211:	00 000 NULL		
F4212:	00 000 NULL		
F4213:	00 000 NULL		
F4214:	00 000 NULL		

screen source reset aux vars debug stack flags

variables

size: word elements: 1

edit show as: hex

variable	value
ARRAY	00h
NO	09F7h

emulator screen (70x11 chars)

9F7

clear screen change font 0/16

line: 34 col: 11

Conclusion:

We successfully implemented Hex to BCD and BCD to Hex conversion using assembly language programs



Experiment 07

Aim : To write an assembly language program to convert from Hex to BCD & BCD to Hex.

Software : Emulator 8086

Theory :

Assembler directives : DUP used to create an array of given length & duplicate values

Instructions :

LEA : used to load effective address of given segment

JB : jump if there's a borrow to the given label

JMP : jump to the given label

INC : increment by 1

DEC : decrement by 1

JNZ : Jump to zero to given label

AND : used to calculate logical & of two operands

SHR : Shift the data

INT 21H : Used to read input characters

D2H : Used to print value