

Code-Conversion-Using-8085

Aim:

To write 8085 microprocessor programs for converting:

1. Hexadecimal to ASCII
2. ASCII to Hexadecimal

Apparatus Required:

- Laptop with an internet connection

Program 1: Hexadecimal to ASCII Conversion

Algorithm:

1. Load the hexadecimal number from memory location 4200H.
2. Mask the upper nibble and check if it is less than 10H.
3. If it is less than 10H, add 30H to convert it to ASCII.
4. If it is greater than 10H, add 37H to convert it to ASCII.
5. Repeat the process for the lower nibble.
6. Store the ASCII equivalent in memory location 4300H and 4301H.

Program:

ORG 00H

; Input : Port 00H (Hexadecimal number)

; Output: Port 01H (ASCII of upper nibble)

; Port 02H (ASCII of lower nibble)

MVI A, 00H

IN 00H ; Read hexadecimal number

MOV B, A ; Save a copy of original number

ANI 0F0H ; Mask lower nibble

RRC

RRC

RRC

RRC ; Move upper nibble to lower position

CPI 0AH ; Compare with 0AH

JC ADD30U ; If less than 0AH, jump

ADI 37H ; Add 37H for A-F

JMP STORE1

ADD30U: ADI 30H ; Add 30H for 0-9

STORE1: OUT 01H ; Send upper nibble ASCII to port 01H

; Process lower nibble

MOV A, B ; Get original number

ANI 0FH ; Mask upper nibble

CPI 0AH ; Compare with 0AH

JC ADD30L ; If less, jump

ADI 37H ; Add 37H for A-F

JMP STORE2

ADD30L: ADI 30H ; Add 30H for 0-9

STORE2: OUT 02H ; Send lower nibble ASCII to port 02H

HLT ; Stop program

END

Output:

Input Ports (numbers are read from these ports):

00H → Input hexadecimal number

Output Ports (results are displayed on these ports):

01H → ASCII of upper nibble

02H → ASCII of lower nibble

Program 2: ASCII to Hexadecimal Conversion

Algorithm:

1. Load the first ASCII digit from memory location 4200H.
2. Convert it to hexadecimal by subtracting 30H (if it's a number) or 37H (if it's a letter A-F).
3. Load the second ASCII digit from memory location 4201H and repeat the process.
4. Combine the upper and lower nibbles to form a hexadecimal number.
5. Store the result in memory location 4300H.

Program:

ORG 00H

; Input : Port 00H → ASCII of upper nibble

; Port 01H → ASCII of lower nibble

; Output: Port 02H → Combined hexadecimal number

; --- Convert upper ASCII nibble to HEX ---

IN 00H ; Read upper ASCII digit

CPI 3AH ; Compare with '9'+1 (3AH)

JC SUB30U ; If less than 3AH, it's 0–9

SUI 37H ; Else subtract 37H for A–F

JMP STOREU

SUB30U: SUI 30H ; Subtract 30H for 0–9 conversion

STOREU: MOV C, A ; Store upper nibble in C

; --- Convert lower ASCII nibble to HEX ---

IN 01H ; Read lower ASCII digit

CPI 3AH ; Compare with '9'+1 (3AH)

JC SUB30L ; If less than 3AH, it's 0–9

SUI 37H ; Else subtract 37H for A–F

JMP STOREL

SUB30L: SUI 30H ; Subtract 30H for 0–9 conversion

STOREL: MOV B, A ; Store lower nibble in B

; --- Combine upper and lower nibbles ---

```

MOV A, C      ; Get upper nibble

RLC           ; Shift left 4 times

RLC

RLC

RLC

ADD B         ; Add lower nibble

OUT 02H       ; Output final hexadecimal number


HLT           ; Stop program

END

```

Output:

The screenshot displays the Deity IDE interface with the following components:

- I/O Ports:** A table on the left showing input/output values.

Port	Value	Direction
0x00	35	Input
0x01	41	Input
0x02	5A	Output
0x03	00	Input
0x04	00	Input
0x05	00	Input
0x06	00	Input
0x07	00	Input
0x08	00	Input
0x09	00	Input
0x0A	00	Input
0x0B	00	Input
0x0C	00	Input
0x0D	00	Input
0x0E	00	Input
0x0F	00	Input
0x10	00	Input
0x11	00	Input
0x12	00	Input
0x13	00	Input
0x14	00	Input
0x15	00	Input
- Assembly Code:** The main editor shows assembly instructions with comments explaining the logic for converting two input nibbles into a hexadecimal output.


```

1 ORG 00H
2 ; Input : Port 00H = ASCII of upper nibble
3 ; Port 01H = ASCII of lower nibble
4 ; Output: Port 02H = Combined hexadecimal number
5
6 ; --- Convert upper ASCII nibble to HEX ---
7 IN 00H ; Read upper ASCII digit
8 CPI 30H ; Compare with '0' (30H)
9 JC SUB30H ; If less than 30H, it's 0-9
10 SUI 37H ; Else subtract 37H for A-F
11 JMP STOREU
12
13 SUB30H: SUI 30H ; Subtract 30H for 0-9 conversion
14 STOREU: MOV C, A ; Store upper nibble in C
15
16 ; --- Convert lower ASCII nibble to HEX ---
17 IN 01H ; Read lower ASCII digit
18 CPI 30H ; Compare with '0' (30H)
19 JC SUB38H ; If less than 30H, it's 0-9
20 SUI 37H ; Else subtract 37H for A-F
21 JMP STOREL
22
23 SUB38H: SUI 38H ; Subtract 38H for 0-9 conversion
24 STOREL: MOV B, A ; Store lower nibble in B
25
26 ; --- Combine upper and lower nibbles ---
27 MOV A, C ; Get upper nibble
28 RLC ; Shift left 4 times
29 RLC
30 RLC
31 RLC
32 ADD B ; Add lower nibble
33 OUT 02H ; Output final hexadecimal number
34
35 HLT ; Stop program
36
37 END

```
- Machine Code:** A table on the right showing the compiled machine code.

Line	Address	Machine Code	Source Code
1		ORG 00H	
2			; Input : Port 00H = ASCII of
3			; Port 01H = ASCII of
4			; Output: Port 02H = Combin
5			
6			; --- Convert upper ASCII ni
7	0x0	00 00	IN 00H ; Read upper
8	0x2	FE 1A	CPI 30H ; Compare u
9	0x4	0A 9C 00	JC SUB30H ; If less th
10	0x7	06 37	SUI 37H ; Else subtr
11	0x9	C3 6E 00	JMP STOREU
12			
13	0xC	06 30	SUB30H: SUI 30H ; Sub
14	0xE	4F	STOREU: MOV C, A ; S
15			
16			; --- Convert lower ASCII ni
17	0xF	00 01	IN 01H ; Read lower
18	0x11	FE 3A	CPI 30H ; Compare u
19	0x13	0A 18 00	JC SUB38H ; If less th
20	0x16	06 37	SUI 37H ; Else subtr
21	0x18	C3 1D 00	JMP STOREL

Input Ports:

00H → ASCII of upper nibble
 01H → ASCII of lower nibble

Output Port:

02H → Hexadecimal number (combined 8-bit value)

Result:

The 8085 microprocessor successfully converts hexadecimal numbers to ASCII and vice versa, storing the results in memory.