Display System Date and Time

Expt No:	11	**Name:**	Mahesh Bharadwaj K
Date :	15/10/2020	**Reg No:**	185001089

Aim:

To write and execute 8086 programs for displaying system date and time.

Procedure:

- Mount masm folder to a drive on DOSBOX.
- Navigate to mounted drive using 'dir' .
- Save 8086 program with the extension **'.asm'** in the same folder using the command **'edit'**.
- Assemble the **.asm** file using the command **'masm filename.asm'**.
- Link the assmebled **.obj** file using the command **'link filename.obj'**.
- Debug the executable file **.exe** with the **'debug filename.exe'** command.
    i. **U:** To view the un-assembled code.
    ii. **D:** Used as 'D segment:offset' to see the content of memory locations starting from segment:offset address.
    iii. **E:** To change the values in memory.
    iv. **G:** Execute the program using command.
    v. **Q** exits from the debug session.

Algorithm:

1. System Date

* START: Move the starting address of data segment to AX register and move the data from AX register to DS register.
* Move 2ah to AH register.
* Calling int 21H with 2a in AH register will return year in CX register, month in DH register, day in DL register and day of the week in AL register.
* Move the offset of the variable DAY in SI register.
* Move the contents stored in DL register to the location in SI register.
* Move the offset of the variable MONTH in SI register.

* Move the contents stored in DH register to the location in SI register.
* Move the offset of the variable YEAR in SI register.
* Move the contents stored in CX register to the location in SI register.

2. System Time

* START: Move the starting address of data segment to AX register and move the data from AX register to DS register.
* Move 2ch to AH register.
* Calling int 21H with 2c in AH register will return hour in CH register, minute in CL register and second in DH register.
* Move the offset of the variable HOUR in SI register.
* Move the contents stored in CH register to the location in SI register.
* Move the offset of the variable MINUTE in SI register.
* Move the contents stored in CL register to the location in SI register.
* Move the offset of the variable SECOND in SI register.
* Move the contents stored in DH register to the location in SI register.

Program:

1. System Date

Program		Comments
start:	MOV AX,data	Move data segment address contents to AX register
	MOV ds,AX	Move data in AX register to DS register
	MOV AH, 2AH	AH is loaded with 2Ah
	INT 21H INT 21h	when AH = 2Ah - get system date
	MOV SI, OFFSET DAY	Load offset of DAY into SI
	MOV [SI], DL	Load the day in DL into DAY
	MOV SI, OFFSET MONTH	Load offset of MONTH into [SI]
	MOV [SI], DH	Load month in DH into [SI]
	MOV SI, OFFSET YEAR	Load offset of YEAR into SI
	MOV [SI], CX	Load year in CX into [SI]
	MOV ah,4ch	
	INT 21h	Request interrupt routine

Unassembled Code:

```
D:\>debug 11-A.EXE
-U
076B:0100 B86A07        MOV     AX,076A
076B:0103 8ED8          MOV     DS,AX
076B:0105 B42A          MOV     AH,2A
076B:0107 CD21          INT     21
076B:0109 BE0000        MOV     SI,0000
076B:010C 8814          MOV     [SI],DL
076B:010E BE0100        MOV     SI,0001
076B:0111 8834          MOV     [SI],DH
076B:0113 BE0200        MOV     SI,0002
076B:0116 890C          MOV     [SI],CX
076B:0118 B44C          MOV     AH,4C
076B:011A CD21          INT     21
```

Input and Output:

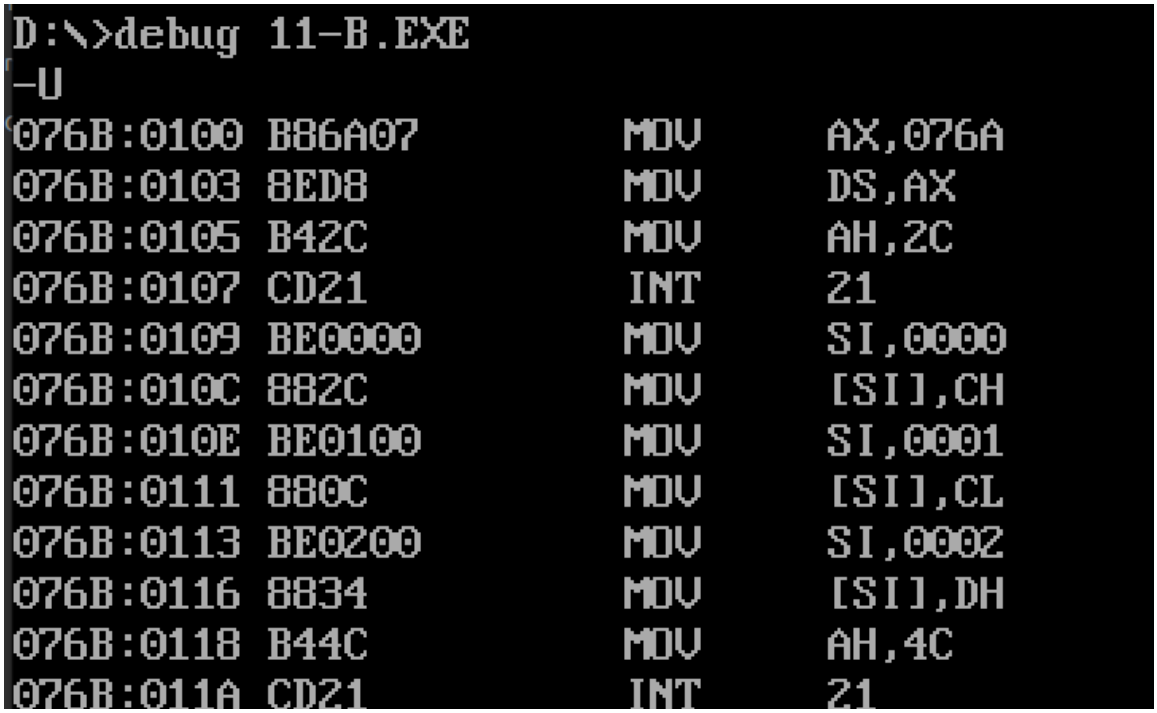
```
-d 076A:0000
076A:0000  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 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  15 0A E4 07 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   ................
```

2. System Time

Program		Comments
start:	MOV AX,data	Move data segment address contents to AX register
	MOV ds,AX	Move data in AX register to DS register
	MOV AH, 2CH	AH is loaded with 2Ch
	INT 21H INT 21h	when AH = 2Ch - get system time
	MOV SI, OFFSET HOUR	Load offset of HOUR into SI
	MOV [SI], DL	Load the hour in DL into HOUR
	MOV SI, OFFSET MINUTE	Load offset of MINUTE into [SI]
	MOV [SI], DH	Load minute in DH into [SI]
	MOV SI, OFFSET SECOND	Load offset of SECOND into SI
	MOV [SI], CX	Load second in CX into [SI]
	MOV ah,4ch	
	INT 21h	Request interrupt routine

Unassembled Code:



Input and Output:

```
-d 076A:0000
076A:0000  00 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  17 04 0A 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   ................
```

Result:

8086 ASL programs for system date and time have been executed successfully using MS - DOSBox.