Lab No.: 02

Topic: Data Transfer Instructions, Basic Functions, and Interrupt 21H

Course: Microprocessor and Assembly Language

Software: EMU8086 Microprocessor Simulator

1. Objectives

By the end of this lab, students will be able to:

- Understand and apply different types of data transfer instructions.
- Use basic arithmetic and increment/decrement operations.
- Use DOS interrupt 21H for character display and program termination.
- Write and execute a short interactive assembly program using EMU8086.

2. Prerequisites

Students should already:

- Understand EMU8086 interface and simulation steps.
- Know the 8086 register structure and general-purpose register usage.
- Know how to assemble and execute code and view register/memory changes.

3. Concept Overview — Data Transfer Instructions (Teaching Part)

3.1 Types of Data Transfer

Type	Description	Example
Register → Register	Move data between registers	`MOV AX, BX`
Immediate → Register	Load a constant directly to register	`MOV AX, 1234H`
Memory → Register	Load data from memory into register	`MOV AX, [2000H]`
Register → Memory	Store data from register into memory	`MOV [2000H], AX`
Immediate → Memory	Store a constant directly into memory	`MOV [2000H], 45H`

> ⚠️ Note: 8086 does not allow memory-to-memory transfer directly; a register must be used as an intermediary.

---

4. Demonstration Programs (Instructor-led)

Example 1: Register to Register Transfer

```
MOV AX, 1234H
MOV BX, AX
HLT
```

🟢 **Observation:** BX = 1234H

Example 2: Immediate to Register Transfer

```
MOV CX, 0A2FH
HLT
```

🟢 **Observation:** CX = 0A2FH

Example 3: Register to Memory Transfer

```
MOV AX, 1234H
MOV [2000H], AX
HLT
```

🟢 **Observation:** Memory at 2000H stores 1234H.

Example 4: Memory to Register Transfer

```
MOV BX, [2000H]
HLT
```

🟢 **Observation:** BX receives data from memory location 2000H.

---

5. Basic Functions (Arithmetic and Display)

5.1 Increment Function

Increases register value by 1.

```
MOV AL, 09H
INC AL
HLT
```

🟢 **Result:** AL = 0AH

5.2 Decrement Function

Decreases register value by 1.

```
MOV BL, 05H
DEC BL
HLT
```

🟢 **Result:** BL = 04H

5.3 Display a Character using INT 21H

Displays a single character on the screen using **DOS Interrupt 21H**.

```
MOV AH, 02H     ; Function 02H - Display Character
MOV DL, 'A'     ; Character to display
INT 21H         ; Call interrupt
MOV AH, 4CH     ; Exit program
INT 21H
```

🟢 **Output:** Displays the character **A** on the output window.

---

6. Common INT 21H Functions

Function (AH value)	Purpose	Description
01H	Input a character	Reads a character from keyboard (Echoed on screen).
02H	Display a character	Prints the character in **DL** register.

Function (AH value)	Purpose	Description
09H	Display a string	Prints a string ending with `$`.
0AH	Input a string	Accepts a string from keyboard.
4CH	Exit program	Terminates the program and returns control to DOS.

> 💡 Tip for students:
>
> - Always load **AH** with the function number before calling `INT 21H`.
> - For displaying, the character must be stored in **DL**.
> - Always end the program with `MOV AH, 4CH` and `INT 21H` for safe termination.

---

7. Practice problem

Problem Statement:

Write an assembly program that:

1. Stores two 8-bit numbers in memory (25H and 15H).
2. Loads them into AL and BL registers.
3. Adds both numbers.
4. Increments the result by 1.
5. Stores the final result in memory (address 2002H).
6. Displays the character `'R'` on the screen when finished.

Program:

```
MOV AL, [2000H]   ; Load first number
MOV BL, [2001H]   ; Load second number
ADD AL, BL        ; Add both numbers
INC AL            ; Increment the result by 1
MOV [2002H], AL   ; Store result in memory

MOV AH, 02H       ; Display 'R' on screen
MOV DL, 'R'
INT 21H
```

```
MOV AH, 4CH       ; Exit program
INT 21H
```

🟢 **Expected Result:**

- Memory(2002H) = 3BH (25H + 15H + 1 = 3BH)
- Screen Output → **R**
