**Experiment No**: 4

**Name of The Experiment**: Write a 8086 Assembly program to take input binary to count the number of ‘1’ bit and take decimal input to determine odd or even.

**Theory:**

**Counting number of ‘1’:**

It takes binary input from the user until a space character (20H) is entered. It converts the binary input to its equivalent decimal value and displays it. It uses interrupts (INT 21H) for input/output operations in DOS. The program reads individual characters and processes them to compute the decimal equivalent of the binary input. After the binary input ends, it displays two messages (MSG2 and MSG3) and then proceeds to count the number of set bits in the converted decimal value (BX register). Finally, it displays the count in ASCII representation by adding '0' to the count value and outputting it using interrupt 21H function 2.

**Odd or Even:**

This assembly code snippet reads individual characters from the user until the Enter key (ASCII 13D) is pressed. It converts the entered characters to their numeric values and stores them in the BX register. If the total count of entered characters is odd (determined by testing the least significant bit of BX), it displays a message (MSG2). If the count is even, it shows a different message (MSG3). This program utilizes BIOS interrupt 21H to interact with the console for input/output operations.

**Program:**

**Counting number of ‘1’:**

.MODEL SMALL

.STACK 100H

.DATA

    MSG1  DB "ENTER THE NUMBER = $"

    MSG2  DB 10D,13D,"THE NUMBER IS = $"

    MSG3  DB 10D,13D,"THE NUMBER OF 1's IS = $"

    COUNT DB 0

.CODE

MAIN PROC

               MOV  AX,@DATA

               MOV  DS,AX

               LEA  DX,MSG1

               MOV  AH,9

               INT  21H

               MOV  BX,0

               MOV  AH,1

               INT  21H

    ENTER\_BIN:

               CMP  AL, 20H

               JE   END\_BIN

               AND  AL, 0FH

               SHL  BX,1

               OR   BL,AL

               INT  21H

               INC  COUNT

               JMP  ENTER\_BIN

    END\_BIN:

               MOV  AH,9

               LEA  DX,MSG2

               INT  21H

               MOV  CX,17

               MOV  AH,2

    ROLOOP:

               RCL  BX,1

               JC   PRINT\_ONE

               JNC  PRINT\_ZERO

    PRINT\_ONE:

               MOV  DL,'1'

               INT  21H

               JMP  CONTINUE

    PRINT\_ZERO:

               MOV  DL,'0'

               INT  21H

               JMP  CONTINUE

    CONTINUE:

               LOOP ROLOOP

               MOV  AH,9

               LEA  DX,MSG3

               INT  21H

               XOR  AX,AX

               MOV  CX,16

    TOP:

               ROL  BX,1

               JNC  NEXT

               INC  AX

    NEXT:

               LOOP TOP

               MOV  AH, 2

               MOV  DL, AL

               ADD  DL,'0'

               INT  21H

               MOV  AH, 4CH

               INT  21H

MAIN ENDP

END MAIN

**Odd or Even:**

.MODEL SMALL

.STACK 100H

.DATA

    MSG1 DB "ENTER THE NUMBER = $"

    MSG2 DB 10D,13D,"THE NUMBER IS ODD.$"

    MSG3 DB 10D,13D,"THE NUMBER IS EVEN.$"

.CODE

MAIN PROC

              MOV  AX,@DATA

              MOV  DS,AX

              MOV  AH,9

              LEA  DX,MSG1

              INT  21h

              MOV  BX,0

              MOV  AH,1

    WHILE\_:

              INT  21H

              CMP  AL,13D

              JE   END\_WHILE

              AND  AL,15D

              SHL  BX,1

              OR   BL,AL

              JMP  WHILE\_

    END\_WHILE:

              TEST BX,1b

              JZ   EVE

              JMP  ODD

    ODD:

              MOV  AH,9

              LEA  DX,MSG2

              INT  21h

              JMP  QUIT

    EVE:

              MOV  AH,9

              LEA  DX,MSG3

              INT  21h

              JMP  QUIT

    QUIT:

              MOV  AH, 4CH

              INT  21H

MAIN ENDP

END MAIN

**Input and Output:**

A computer screen with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

**Comments:**

Please note, This code reads binary input from the user displaying the count of set bits (number of 1s) in the entered binary value.