Experiment Name:

Write a program that takes a hex number of 4 digit or less as input and outputs its binary on the next line and count the number of 1's in its reversed binary number.

Theory:

The objective of this program is to convert a hex number into 16-bit binary number and then reversing that binary number, the number of 1's need to be counted and printed in the output. For this program in assembly, loops, JMP instructions and some registers need to be used.

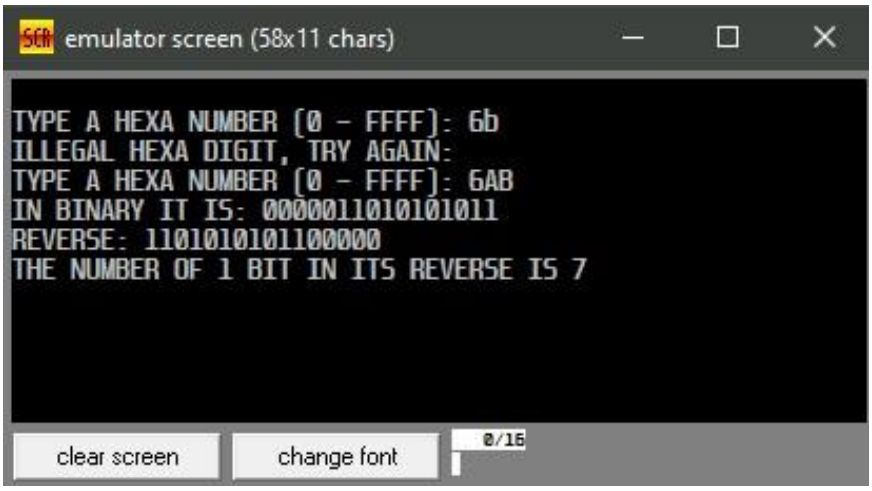
Code:

```
.STACK 100H
.MODEL SMALL
.DATA
   M1 DB 0AH,0DH,'TYPE A HEXA NUMBER (0 - FFFF) : ','$'
   M2 DB 0AH,0DH,'IN BINARY IT IS : ','$'
   M3 DB 0AH,0DH,'ILLEGAL HEXA DIGIT, TRY AGAIN :','$'
   M5 DB 0DH,0AH,'REVERSE : $'
   M4 DB 0AH,0DH,'THE NUMBER OF 1 BIT IN ITS REVERSE IS '
   C2 DB 0,'$'

.CODE
MAIN PROC
   MOV AX,@DATA
   MOV DS,AX

   MOV AH,9
   LEA DX,M1
   INT 21H



   START :
      XOR BX,BX
      MOV CL,4
      MOV AH,1
      INT 21H
   WHILE_:
      CMP AL,0DH
      JE END_WHILE
      CMP AL,'0'
      JL ERROR
      CMP AL, '9'
      JG LETTER
      AND AL,0FH
      JMP SHIFT
```

```
LETTER:
   CMP AL,'F'
   JG ERROR
   CMP AL,'A'
   JL ERROR
   SUB AL,37H

SHIFT:
   SHL BX,CL
   OR BL,AL
   INT 21H
   JMP WHILE_


END_WHILE:
   MOV AH,9
   LEA DX,M2
   INT 21H
   MOV CX,16
   MOV AH,2

SHOW:
   ROL BX,1
   JC ONE
   MOV DL,'0'
   INT 21H
   JMP LOOP1
ONE:
   MOV DL,'1'
   INT 21H
   ADD C2,1

LOOP1:
   LOOP SHOW

MOV AH,9
LEA DX,M5
INT 21H

MOV CX, 16
MOV AH,2

REVERSE:
   SHR BX,1
   JC ONEE
   MOV DL,'0'
   INT 21H
   JMP LOP1
ONEE:
   MOV DL,'1'
   INT 21H

LOP1:
   LOOP REVERSE
   JMP OUT_
```

```
    ERROR:
        MOV AH,9
        LEA DX,M3
        INT 21H
    JMP START


    OUT_:
        ADD C2,30H
        MOV AH,9
        LEA DX,M4
        INT 21H
        MOV C2,0


    MOV AH,4CH
    INT 21H
    MAIN ENDP
END MAIN
```

Output:



Discussion:

In the above program, a hexadecimal number was taken as input using a loop. In the loop CMP instructions were used to check the correction of the hex number and if it was not correct then, JMP was used to jump to another block named ERR where error message was printed. After taking the input, using Left Shift and Right Rotate and Hexadecimal number was converted into 16-bit binary number. After the conversion, it was reversed using Right shift with the help of loop. Then a variable C2 was used to count the occurrence of 1s in that 16-bit binary stream and that number was printed finally.