**RAJSHAHI UNIVERSITY OF ENGINEERING AND TECHNOLOGY**

**Course No:** CSE 3110

**Course Title:** Sessional Based on CSE 3109

**Problem No:** 03

**Problem Name:** Write a program to count number of 1 bits in a binary number and check the number is odd or even.

**Submitted To:**

Sadia Zaman Mishu

Assistant Professor

Computer Science and Engineering

Rajshahi University of Engineering and Technology

**Submitted By:**

Name : Md. Fakhrul Islam

Roll: 1803071

Session: 2018-19

Department : Computer Science and Engineering

Rajshahi University of Engineering and Technology

**Problem No:** 03

**Problem Name:** Write a program to count number of 1 bits in a binary number and check the number is odd or even.

**Description:**

The shift and rotate instructions shift the bit in destination operand by one or more positions either to the left or right. For a shift instruction, the bits shifted out are lost; for a rotate instruction, bits shifted out from one end of the operand are put back into the other end. The instructions for left shift,

SHL destination, 1

OR, SHL destination, CL

For right shift,

SHR destination, 1

OR, SHR destination, CL

For Rotate Left,

ROL destination, 1

OR, ROL destination, CL

For Rotate Right,

ROR destination, 1

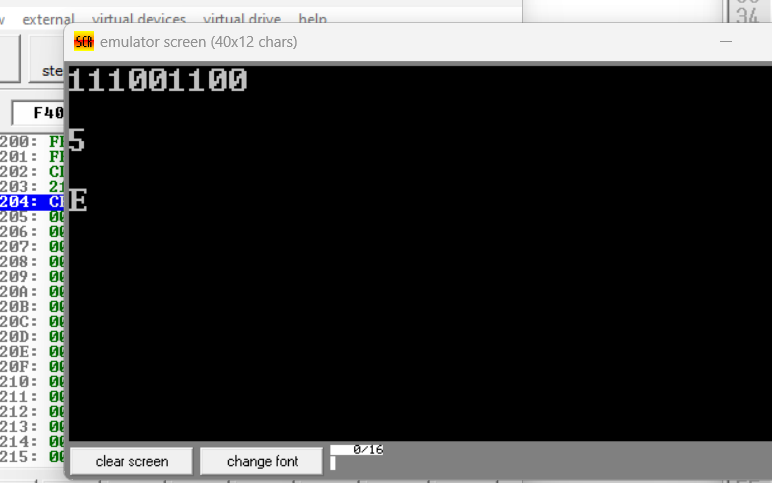
OR, ROR destination, CL

To check the number is odd or even, bit masking is used. The instruction is, AND destination, 1. If the result is 1 , the number is odd, otherwise the number is even.

**CODE:**

|  |  |
| --- | --- |
| .MODEL SMALL  .STACK 100H  .DATA  .CODE  MAIN PROC    XOR BX, BX  MOV CL, 0  MOV AH, 1  INT 21H    WHILE:  CMP AL, 0DH  JE END\_WHILE  AND AL, 0FH  SHL BX, 1  OR BL, AL  INT 21H  JMP WHILE  END\_WHILE:  MOV AH, 2  MOV DL, 0AH  INT 21H  MOV DL, 0DH  INT 21H    XOR AX, AX  MOV CX, 16  TOP:  ROL BX, 1 | JNC NEXT  INC AX  NEXT:  LOOP TOP  ADD AX, 48  MOV AH, 2  MOV DX, AX  INT 21H  MOV AH, 2  MOV DL, 0AH  INT 21H  MOV DL, 0DH  INT 21H  AND BL,1  CMP BL,0  JE NEXT1  MOV AH,2  MOV DL,'O'  INT 21H  JMP END  NEXT1:  MOV AH, 2  MOV DL, 'E'  INT 21H  END:  MOV AH, 4CH  INT 21H  MAIN ENDP  END MAIN |

**OUTPUT:**



**Conclusion:**

The left shift shifts the bits to the left and the msb is shifted into Carry Flag. On the other hand, the right shift shifts the bits to the right and the lsb is shifted into Carry Flag. Bit masking is used to checking a number is odd or even. It is much faster than any other method to checking odd or even. RCL or RCR instructions can be used for counting 1 bits in the number. In this case, the input will not be changed.