System Programming: Assignment 1

1. Explain SIC and SIC/XE
2. Describe addressing modes used in SIC/XE
3. Explain RISC and CISC machines.
4. Write a Sequence of instruction for SIC to clear a 20 byte string to all blanks
5. With an example explain I/O operation of SIC/XE.
6. Translate (by hand) the following assembly program to SIC/XE object code.Starting address program is 1000(H) .Also assume Opcode for instruction. The output format will contains H record, T record, and E record.

```
STRCP2    START    1000
FIRST     LDT      #11
          LDX      #0
MOVECH    LDCH     STR1,X
          STCH     STR2,X
          TIXR     T
          JLT      MOVECH

STR1      BYTE     C'TEST STRING'
STR2      RESB 11
           END     FIRST
```

6. Explain program block with an example, a machine independent assembler feature .
7. Explain the following machine independent features of assembler
    - Literals
    - Symbol defining statements
    - Expressions
8. Explain control section and program linking.
9. Write LOCCTR, Object Code and Object program for the following

```
          START    1000
FIRST     STL      RETADR    Store return address
          LDX      #LENGTH   Load length of input
          JSUB     RDREC     Read first number
          JSUB     CONVRT    Convert from character to numeric
          STA      NUM1      Store first number
          JSUB     RDREC     Read second number
          JSUB     CONVRT    Convert from character to numeric
          STA      NUM2      Store second number
          LDA      NUM1      Load first number
          ADD      NUM2      Add second number
          STA      RESULT    Store result
          JSUB     WRREC     Write result to output
          J        @RETADR   Return to caller
RETADR    RESW     1         Return address storage
NUM1      RESW     1         First number storage
NUM2      RESW     1         Second number storage
```

```
RESULT  RESW    1       Result storage
LENGTH  WORD    6       Length of input buffer
BUFFER  RESB    6       Input buffer
        END     FIRST

RDREC   CLEAR   X       Clear index register
        CLEAR   A       Clear accumulator
        CLEAR   S       Clear status register
RLOOP   TD      INPUT   Test device status
        JEQ     RLOOP   Wait for device to be ready
 RD     INPUT   X       Read character from input
        COMP    X,#6    Check for end of line
        JEQ     EXIT    If end of line, exit loop
        STCH    BUFFER,X Store character in buffer
        TIX     LENGTH  Increment buffer index
        JLT     RLOOP   Continue reading characters
EXIT    STX     LENGTH  Store length of buffer
        RSUB            Return to caller

WRREC   CLEAR   X       Clear index register
        CLEAR A         Clear accumulator
        CLEAR S         Clear status register
WLOOP   TD      OUTPUT  Test device status
        JEQ     WLOOP   Wait for device to be ready
        LDCH    BUFFER,X Load character from buffer
        WD      OUTPUT  Write character to output
        TIX     LENGTH  Increment buffer index
        JLT     WLOOP   Continue writing characters
        RSUB            Return to caller

CONVRT  CLEAR   A       Clear accumulator
        CLEAR   X       Clear index register
        LDX     #0      Initialize index register
CLOOP   TD      BUFFER,X Test for end of input
        JEQ     CXIT    If end of input, exit loop
        ADDR    A,X     Add index to address
        LDA     BUFFER,X Load character from buffer
        SUB     #C_ZERO Convert character to numeric
        STA     BUFFER,X Store numeric value back
        TIX     LENGTH  Increment buffer index
        JLT     CLOOP   Continue processing characters
        CXIT    RSUB    Return to caller

INPUT   BYTE    X'F1'   Input device
```

```
OUTPUT    BYTE    X'05'    Output device
C_ZERO BYTE   X'30'    ASCII value for zero
```

10. Relocate the above program into different memory of your choice and write the resulting object program with the suitable modification record.
11. Divide the program above into different program blocks and write the resulting opcode along with all data structures.
12. Refer the page number 91 of the textbook, and write the resulting object program with D, R and revised M record.
13. Define CSECT, EXTDEF and EXTREF.
14. Explain load and go assembler.
15. Explain multipass assembler.