SS Module I Important Questions

1. Differentiate between system software and application software

System software

- System software consists of a variety of programs that support the operation of a computer .
- These softwares make it possible for the user to focus on an application without needing to know how the system works internally.
- System software manages system resources and provides a platform for application software to run.
- A system is unable to run without system software. System software is general purpose. Eg: OS

Application Software

- Application software allows users to accomplish one or more specific tasks.
- Application software focuses on an application. Application software runs when the user requests.
- Application software is specific purpose.\
- Eg: Microsoft office, Photoshop, Educational software

2. Explain any three system softwares.

Assembler

Assembler is a system software which converts an assembly language program into machine language

Linker

- Linking is a process of collecting and combining various pieces of code and data into a single file that can be loaded into memory and executed
- Linkers play an important role in software development. Instead of organizing a large application as one single source file we can decompose it into smaller modules that can be modified and compiled separately.

Loader

- A loader copies programs from a storage device to computer’s main memory where the program can be executed.
- Most loaders function without user involvement.

3. Explain SIC architecture

- Simplified Instructional Computer (SIC) is a hypothetical computer that includes the hardware features most often found on real machines
- There are two versions of SIC, they are, standard model (SIC), and, extension version (SIC/XE)
- The two versions have been designed to be upward compatible
    - Upward compatibility -> Object program for the standard SIC machine will also execute properly on SIC/XE machine.

SIC Architecture

Memory

- There are 2^15 bytes in the computer memory
- 3 consecutive bytes form a word(24) bits
    - 2^3 = 24

Register

There are five registers, each 24 bits in length. Their mnemonic, number and use are given in the following table

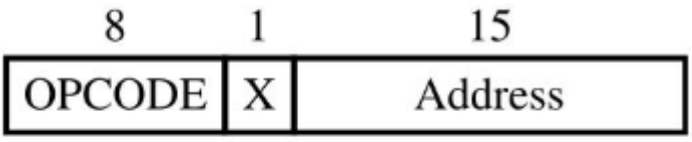
Mnemonic	Number	Use
A	0	Accumulator; used for arithmetic operations
X	1	Index register; used for addressing
L	2	Linkage register; JSUB
PC	8	Program counter
SW	9	Status word, including CC

Data formats

- Supports only integer and character formats

Instruction Formats

All machine instructions in the standard version of SIC have the following 24 bit format:



Flag bit x is used to indicate the indexed addressing mode.

Addressing mode

- These are of 2 types

  Direct Addressing Mode: Here flag bit x=0

  Target Address= Actual Address

  Indexed Addressing Mode: Here flag bit x=1

  Target Address= Actual Address+Index Register (X) contents

  i.e. **Target Address= Address+(X)**

Instruction set (Acronym: DACSI)

- Data transfer instruction
  - For loading and storing register
    - LDA, STA, LDX, STX
- Arithmetic Operation instruction
  - ADD, SUB, MUL, DIV, COMPR
- Conditional Branching Instruction:
  - JLT, JEQ, JGT
- Subroutine call instruction
  - JSUB
    - To jump

    - RSUB
        - To return
- Input and output instruction
    - Target port is specified by last bits of Register A
    - Operations are executed by transferring a single byte each time
    - Each device is assigned a unique 8 bit code to send and receive data and control signal

Input and Output

- Performed by transferring 1 byte each time to or from rightmost 8 bits of register A
- Test Device TD
    - Tests whether the addressed device is ready to send and receive a byte of data
- Read Data RD and Write Data is used for reading and writing Data

Data movement and storage definition

- 3 byte words
    - LDA, STA,LDX, STX
- 1 byte words
    - LDCH, STCH
- Storage Definitions
    a. WORD- ONE WORD CONSTANT
    b. RESW- ONE WORD VARIABLE
    c. BYTE- ONE BYTE CONSTANT
    d. RESB- ONE BYTE VARIABLE

4. Explain SIC/XE architecture

SIC XE Architecture

Memory

- There are 2^20 bytes in the computer memory = 1 MB

Registers

Mnemonic	Number	Special use
B	3	Base register
S	4	General working register
T	5	General working register
F	6	Floating-point accumulator (48 bits)

Floating point datatype

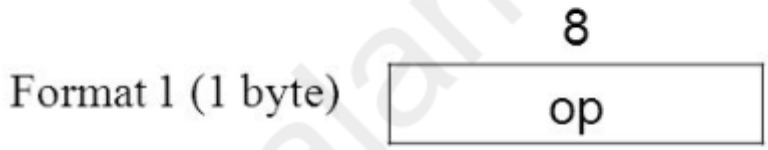
- 48 bits

3. Floating point Data type: There is a 48 bit floating point data type, $F*2^{(e-1024)}$

Instruction formats

- Format (1byte)

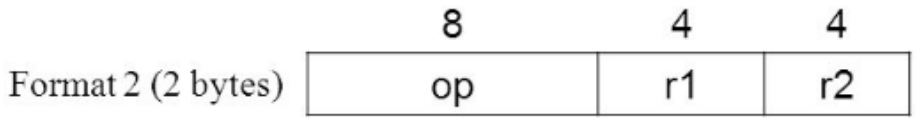
Format 1 (1 Byte): Contains only operation code



Eg: RSUB (Return to Subroutine)

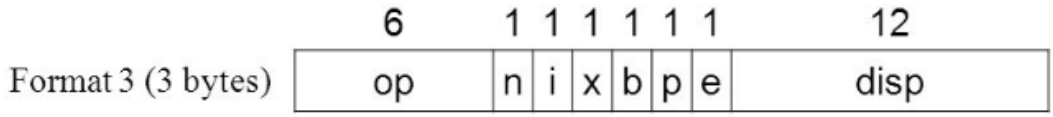
- Format(2Byte)

Format 2 (2 Bytes): First 8 bits for operation code, next four for register 1 and following for register 2.



Eg: COMPR A, S (Compare contents of register A and S)

- Format 3 Byte



First 6 bits contain operation code.

Next 6 bits contain flags.

Last 12 bits contain displacement for the address of the operand.

Flags are in order -n, i, x, b, p, e.

e indicates instruction format.

Bits i and n are used for target address calculation

- Example LDA #3

Different cases in format 3

- i = 0, n =1
  - indirect addressing
- i = 1, n = 0
  - Immediate addressing
- i = 0, n = 0 or i = 1, n = 1
  - Simple addressing

Format 4 (4 bytes): Here e=1

	6	1	1	1	1	1	1	20
Format 4 (4 bytes)	op	n	i	x	b	p	e	address

Addressing mode and flag bits

- Direct
  - x,b,p = 0
  - n=i= 1 or 0
- Relative
  - b or p = 1, other one = 0
    - b=0, p = 1 or b = 1, p = 0
- Immediate
  - i = 1, n=0
- Indirect
  - i = 0, n = 1
- indexed
  - x = 1

Two relative Addressing modes are

- Base relative
- PC Relative

Mode	Indication	Target Address Calculation
Base Relative Addressing Mode	b = 1 P = 0	**TA = Displacement + (B)** B – Base Register Displacement is 12 bit unsigned register. Displacement lies between 0 to 4095
Program Counter Relative Addressing Mode	b = 0 P = 1	**TA = Displacement + (PC)** PC – program counter Displacement is 12 bit signed integer. Displacement lies between – 2048 to 2047.
Direct Addressing Mode	b = 0, P = 0 (Format 4 instruction) b = 0, P = 0 (Format 3 instruction)	**TA = address field of format 4 instruction** TA = Displacement field value of format 3 instruction
Base Relative Indexed Addressing Mode	b = 1, P = 0 X = 1	**TA = Displacement + (B) + (X)** B – Base register X – Index register Displacement is 12 bit unsigned register. Displacement lies between 0 to 4095
Program Counter Relative Indexed Addressing Mode	b = 0, P = 1 X = 1	**TA = Displacement + (PC) + (X)** PC – program counter X – Index Register Displacement is 12 bit signed integer. Displacement lies between – 2048 to 2047

Instruction Set

- Instruction that load and store new register B
  - LDB
  - STB
- Arithmetic operation
  - ADDF
  - SUBF
  - MULF
  - DIVF
  - Here F is floating point register
- Instruction that take operand from register
  - RMO - Register move
    - RMO S,B
      - Register ‘S’ content is moved to ‘B’ register
- Register Artihmetic operation
  - ADDR
  - SUBR
  - MULTR
  - DIVR
  - Eg ADDR S,B
    - Add value of register B with register S and store result in register B

Input and output

- Supports all I/O instructions
- There are special I/O Channels which are utilized for data transfer when CPU is involved in another process at the same time
- Maxmum 16 I/O Channels
- RD and WD is used to read and write data from or to specified I/O devices.
- instructions
  - For Start, Test and Halt

SIO	Instruction is used to Start an I/O Channel number
TIO	Instruction is used to Test an I/O Channel number
HIO	Instruction is used to Halt an I/O Channel number

5. What are assembler directives? Give examples

- Assembler directives are pseudo instructions
- Provides instruction to assembler itself
- Not translated into machine operations
- SIC and SIC XE has the following assemble directives
  - START
    - Name and starting address
  - END
    - End of the source program and first executable statement in the program
  - BYTE
    - Generates Character/Hexadecimal Constant
  - WORD
    - Generates one WORD Constant
  - RESB
    - Indicates Number of bytes for data area
  - RESW
    - Reserves indicated number of words

b) Describe the use of n,i,x,b,p and e bits in the SIC/XE instruction format. Write the binary combination for these bits such that the resultant target address would be as below and also state what would be the addressing modes for each.

i. (PC) + disp ii. (B) + disp iii. (PC) + disp + (X) iv. (B) + disp + (X)

Module -2