

- 4p1. System Software Vs Application Software
 4p2. Different System Software & their differences. (Including functions)
 4p3. Functions of OS
 4p4. SIC Architecture (Explain) (10 marks).

a) Memory : 2^{15} bytes

b) Register : A, X, L, PC, SW (5)

c) Data formats

d) Instruction formats

e) Addressing modes

f) Instruction Set

g) Input & Output

4p5) SIC/XE Architecture (Explain) (10 marks).

a) Memory - 220 bytes

b) Register : A, X, L, B, S, T, F, PC, SW.

c) Data Format (floating pt including)

d) Instruction format (4 types)

e) Addressing modes.

f) Instruction Set

g) Input and Output.

4p6) SIC Vs SIC/XE Architecture.

4p7) Assembler Directives (List out & Explain)

Eg:-

— START

— END

— BYTE

— RESB

= } Array base

Module II

- SIC Programming
- SIC/XE Programming.
- ~~if~~ Basic Functions of Assembler (5 marks)
 - ~~operation, data, etc.~~
- Assembler Output Format / Records

~~if~~

- Header Record
- Text Record
- End Record.



Compulsary.

Eg:- H ^ COPY ^ 001000107A
T ^ 001000 - - -

- Assembler Data Structures

~~if~~



- Operation Code Table (OPTAB)
- Symbol Table (SYMTAB)
- Location Counter (LOCCTR)

E ^ 00100

- Two Pass Assembler (Compulsary / Qn) 10 marks

~~if~~ - Pass 1 (8 mark)

~~if~~ - Pass 2 (8 mark)



With Data Structures.

- Hand assembly (optional).

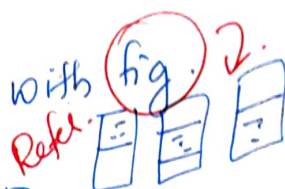
Module 3

Machine Dependent Assembler Features

↳ Instruction Format

↳ Addressing Modes

↳ Program Relocation. with **fig. 2**



Machine Independent Assembler Features

↳ Literals

↳ Symbol Defining statements

↳ Expressions

↳ Program Blocks

↳ Control Sections

↳ Program linking

LITTAB

Labels

forward

Reference

+1, *, absolute, Relativ

USE: Segments of code that are merged within a single object program

CSECT: Segments of code that are translated into independent object program units

External reference

Assembler Design Options

↳ One Pass Assembler

- Load & go assembler,

↳ Multi Pass Assembler.

Eg:- 1/P HALFZ EQU MAXLEN/2 no forward Refer
BUFEND EQU * Symbol definition must be completed in pass

Eg:- ALPHA EQU BETA
BETA EQU DELTA
DELTA RESW 1

Implementation

↳ MASM Assembler

- CODE, DATA, CONST, STACK.
ASSUME, jump, -----

Module 2

Important - Forward Reference

With Eg.

A reference to a label that is defined later in the program.

Module 4

Loader & Linker

Basic Loader Functions →

✓ L Design of Absolute Loader (4)

✓ L Simple bootstrap Loader (4)

✓ Machine Dependent Loader Features

↳ Relocation

↳ Program Linking

✓ Algorithm & Data Structures of Two pass Linking Loader

↳ ESTAB

Two Variables - PROGADDR, CSADDR.

✓ Machine Independent Loader Features

↳ Automatic library search

↳ Loader Options

✓ Loader Design Options.

INCLUDE
DELETE
CHANGE

↳ Linkage Editor (2 figures)

✓ ↳ Dynamic Linking (i/p)

↳ Bootstrap loader

Module 5

4p Macroprocessor

- Macro Instruction Definition & Expansion

- One pass Macro processor Algm & Data Structure

Compulsary Algorithm -
Data structures

↳ NAMTAB

↳ DEFTAB.

4p → Machine Independent Macro processor Features

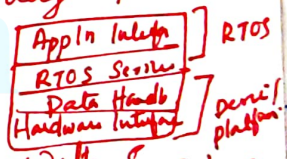
- Concatenation of Macro processor
- Generation of unique labels
- Conditional Macro Expansion
- keyword Macro Parameters.

→ Macro processor Design Options.

- Recursive Macro Expansion - Very //p
- General purpose Macro processors
- Macro processor with in Language Translator.

Device Drivers

- Anatomy of a device driver.



Very //p character & block device drivers with Eeg:-

- General design of device driver

- Figure:
1. General design.
 2. Synchronous Device Driver
 3. Asynchronous Device Driver.
 4. Callback Input only
 5. Serial Input & Output Data Spooler.

Text Editors

- Overview of Editing

4p - user Interface (4 marks).

4p - Editor Structure. : fig with Explanation (8 mark)

Debuggers

4p Debugging Functions & Capabilities. (4 marks)

- Relationship with other parts of system.

Compulsary 4p Debugging Methods → Induction - fig
Deduction - fig } Essay

Back backing