System Software

Module I

Dethy M Ramesh
Ap in ESE

ip 1. System Software Vs Application Software
cp 2. Different System Software & their differences. (Including functions)
ip 3. Functions of OS
up 4. 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 — Data movement.
— Arithmetic.
— Branching.
— Subroutine
g) Input & O/P.

up 5) SIC/XE Architecture (Explain). (10 marks).

a) Memory - 2^{20} 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.

up 6) SIC Vs SIC/XE Architecture.
ip 7) Assembler Directives (Listout & Explain.)

Eg:-

— START
— END
— BYTE
— RESB
— RESW
} Array based

Module II

- SIC Programming
- SIC/XE Programming.
- imp Basic Functions of Assembler (5 marks)
  - ~~Instruction, Data, ...etc~~.
- Assembler Output Format / Records
  - imp
    - Header Record
    - Text Record
    - End Record.
  - } Compulsory.
  - Eg:- H ^ COPY ^ 001000 107A
    T ^ 001000 - - - .
    ⋮
    E ^ 00100
- Assembler Data Structures.
  - imp
    - → Operation Code Table (OPTAB)
    - → Symbol Table (SYMTAB)
    - → Location Counter (LOCCTR)
- Two Pass Assembler (Compulsory Qn) 10 marks
  - imp — Pass 1 (8 mark)
  - imp — 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. Refer

Machine Independent Assembler Features
- ↳ Literals — LITTAB
- ↳ Symbol Defining statements — Labels, forward Reference
- ↳ Expressions — +, -, *, / — absolute, Relativ
- ↳ Program Blocks — USE: Segments of code that are arranged within a single object pgm unit
- ↳ Control Sections — CSECT: Segments of code that are translated into independent object progm units
- ↳ Program linking → External reference

Individual Qn (4 marks)

EXTDEF
EXTREF

Assembler Design Options

- ↳ One Pass Assembler
  - Load & go assembler,
- ↳ Multi Pass Assembler.

Eg:-
I/P HALFZ EQU MAXLEN/2
BUFEND EQU *

no forward Refer
Symbol definition must be completed in pass I

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.

~~difficult~~ ~~will be unable to assemble this stmt.~~

Module 4

Loader & Linker

Basic Loader functions →

- ↳ Design of Absolute Loader (4)
- ↳ 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.

INCLUDE
DELETE
CHANGE

Loader Design Options.

- ⊢ Linkage Editor (2 figures)
- ⊢ Dynamic Linking (i/p)
- ↳ Bootstrap loader

Module 5

I/P Macroprocessor

- Macro Instruction Definition & Expansion
- One pass Macro processor Algm & Data Structure

Compulsory
- Algorithm
- Data structures
  - ↳ NAMTAB
  - ↳ DEFTAB.

I/P → 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 I/P
- General purpose Macro processors
- Macroprocessing within Language Translation

Device Drivers

- Anatomy of a device driver. — [Appln Interface / RTOS Services] RTOS; [Data Handle / Hardware Interface] Device platform
- Very I/P character Vs block device drivers with Eg: —
- General design of device driver

Figure:
1. General design.
2. Synchronous Device Driver
3. Asynchronous Device Driver.
4. Latch Input Only
5. Serial Input & O/p Data Spooler.

Text Editors

- Overview of Editing
- I/P — User Interface (4 marks).
- I/P — Editor Structure. : fig with Explanation (8 mark

Debuggers

- I/P Debugging Functions & Capabilities (4 marks)
- Relationship with other parts of System.
- Compulsory I/P Debugging Methods →
  - Induction — fig
  - Deduction — fig
  - Backtracking

} Essay