Prepared By Abhiyan Dev Rimal
https://github.com/abhirimal/be-software-notes

classmate
Date
Page

Chapter-1 (2Q)
Introduction to System Software

1) Explain system software with its importance.
or
What is system software? Explain its role in a system.

Ans: Software is a set of instructions or programs written to carry out certain task on digital computers.
It is classified into System Software and Application Software.

→ System software is the type of software which is the interface between application software and system.

→ Application software is a type of software that is mainly developed to perform a specific task as per the user's request

System Software (Importance)

→ System software consist of a variety of programs that support the operation of computer.

→ It performs variety of system functions such as resource management, I/O management and storage management.

→ Low level languages are used to write the system software.

→ It maintains the system resources and gives the path for application software to run.

Scanned with CamScanner

→ It is solely responsible for running the operating system (OS) and managing the computer device entirely, and without it the system cannot run.
(↳ without system software, system cannot run)

→ The operating speed of system software is very fast and and they are smaller in size.

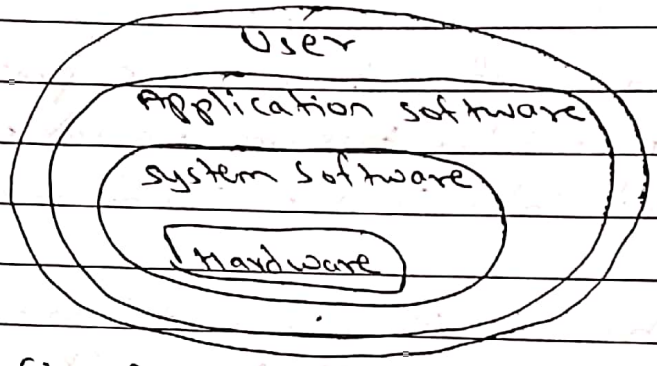


Fig:- A computer system.

Types/Example of System Software

→ Operating System
→ Assembler
→ Loader or Linker
→ Macro Processor

Scanned with CamScanner

2) Difference between application software and system software.

System Software	Application Software
1) System software is a set of computer programs that is designed to manage system resources.	1) Application software is a software that is mainly developed to perform a specific task as per the user's request.
2) It acts as an interface between the application software and computer hardware.	2) It acts as an interface between the end user and system software.
3) They are written in low level languages such as assembly language.	3) They are written in high-level language such as C, C++, Java
4) System cannot run without system software	4) System can run without application software.
5) It starts running when the system is powered on and runs until the system is powered off.	5) It starts when the user launch it and stop when user closes it.
6) Programming of system software is complex	6) Programming of application software is comparatively easy.
7) Capable of running independently.	7) Can't run independently.
8) Example of system software is OS, compiler, assembler, loader and linker.	8) Examples: web browsers, MS office, VLC player, etc.

Scanned with CamScanner

13) Briefly explain the machine dependent and machine independent parts of the system software.

→ Machine dependent software is software that runs only on a specific computer.

→ Applications that run on multiple computer architectures are called machine independent.

→ System software are intended to support the operation and use of the computer itself rather than any particular application. For this reason, they are usually related to the architecture of the machine on which they are to run.

→ Most system software is machine dependent.

→ Operating system are machine dependent with other softwares.

→ On the other hand, there are some aspects of system software that do not directly depend upon the machine being run.

→ For example, the general design and logic of an assembler is basically the same on most computers.

Scanned with CamScanner

~~4) Describe the architecture of SIC machine.~~

4) What is SIC machine?

→ It refers to simplified Instruction computer which is a hypothetical computer that has been designed to include the hardware features most often found on real machines while avoiding unusual and irrelevant complexities.

5) Describe the architecture of SIC machine.

→ Simplified Instruction computer (SIC) is a typical microcomputer which comes in two versions i.e The Standard Model & XE version.
SIC machine architecture consists of memory, registers, data formats, instruction formats, addressing mode & I/O.

* Memory

→ There are 2^{15} bytes in computer memory i.e 32,768 bytes.

→ 1 word = 3 bytes = 24 bits

→ Each location in memory contains 8 bit-bytes.

P.T.O →

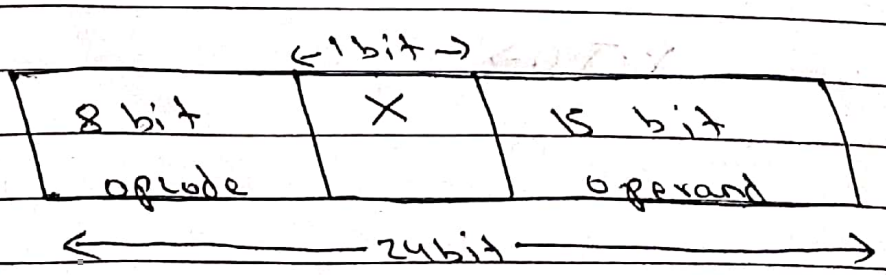
Scanned with CamScanner

* Registers
→ There are 5 registers
→ Each are 24 bits in length.
→ Their name, mnemonic, number and use are given in the following table.

Name	Mnemonic	Number	Use
Accumulator	A	0	Used for arithmetic operation
Index Register	X	1	used for addressing
Linkage Register	L	2	JSUB
Program Counter	PC	8	it contains the address of the current instruction being executed.
Status Word	SW	9	it contains variety of information including the condition code

* Data formats
→ Size of integer = 24 bit
→ Negative numbers are represented in 2's complement format.
→ Character are represented by using 8-bit ASCII code.
→ Float point data cannot be represented.

* Instruction format
→ They are of 24 bits



TA = Target Address

* Addressing modes

X = 0 -> directing addressing mode

Here, TA = address

X = 1 -> indexed addressing mode

here, TA = address + (X)
↳ represents content of index register.

* Instruction Set

i) ~~Data~~ Arithmetic operating instruction.
ADD, SUB, MUL, DIV

ii) Data Management Instruction
LDA, LDX, STA, STX

iii) Subroutine linkage
RSUB, JSUB

* Input and Output (I/O)

-> I/O is performed by transferring one byte at a time.

-> each device is assigned unique 8 bit code.

-> 3 I/O instructions.

Scanned with CamScanner

6) Explain about SIC/XE architecture

* Memory
→ Memory size = 2^{20} bytes = 1 mb
→ 1 word = 24 bits = 3 bytes
→ 20 bits are used to represent memory location.

* Registers
→ 10 × 24 bit registers
→ Same as that of SIC simple. Additional B, S, T, F registers are provided by SIC/XE

UseName	Mnemonics	Number
Accumulator	A	0
Indexed Register	X	1
Linkage Register	L	2
Base register	B	3
General working register	S	4
General working register	T	5
Floating point accumulator	F	6
Program Counter	PC	8
Status word	SW	9

* Data formats
→ Integers are stored in 24 bits, 2's complement format.

Scanned with CamScanner

→ Characters are stored in 8-bit ASCII format.
→ Size of floating point number = 48 bit.

* Instruction format

1) 1 byte instruction

8 bit
Opcode

2) 2 byte instruction

8 bit Opcode	r_1	r_2

3) 3 byte instruction

← 6 bit →						← 12 bit →
Opcode	n	i	x	b	p	e	Operand

4) 4 byte instruction

← 6 bit →						← 20 bit →
Opcode	n	i	x	b	p	e	address

Note:- if e = 0, 3 byte instruction
if e = 1, 4 byte instruction

Scanned with CamScanner

* Addressing mode

1) $n=0, i=0$ } Direct addressing mode
$n=1, i=1$ }

2) $n=0, i=1$ → immediate addressing mode
$n=1, i=0$ → indirect addressing mode

3) $x=0$ → no indexing
$x=1$ → indexed addressing

4) $b=0$ } absolute addressing mode
$p=0$ }

5) $p=1$ → PC relative
TA = PC + displacement

$b=1$ → base relative
TA = base + displacement

{ $0 \le base \le FFF$ } Hex
{ $-800 \le PC \le 7FF$ }

* Instruction Set
1) Load Store (LDA, STA, LDX, STX)
2) Arithmetic Operations (ADD, SUB, MUL, DIV)
3) Compare (Comp)
4) Input Output Control (RD, WD)

Scanned with CamScanner

* Input/Output

-> 2^8 (256) I/O devices may be attached, each has its own unique 8 bit address.

-> 1 byte of data will be transferred.

-> It has 6 I/O operations

RD - Read Data

WD -> Write Data

TD -> Test Data

Q) RISC Machines

- UltraSPARC
- PowerPC Architecture
- Cray T3E Architecture

Q) CISC Machines

- VAX Architecture
- Pentium Pro Architecture

Scanned with CamScanner

7) Differentiate RISC and CISC architectures.

CISC	RISC
1) It stands for Complex Instruction Set computer.	1) It stands for Reduced Instruction set computer
2) It has large number of instructions	2) It has less no of instructions.
3) large number of addressing mode	3) few number of addressing modes.
4) It has ~~fixed~~ variable length instruction format.	4) It has fixed length instruction format.
5) It requires more than 1 cycle for execution.	5) It requires single cycle for execution.
6) ~~Execution time is very high.~~ Cost is high	6) ~~Execution time is very low.~~ Cost is less
7) It is more powerful	7) It is less powerful
8) Execution time is very less.	8) Execution time is very high.
9) Microprogrammed control unit	9) Hardwired control unit.
10) Eg. VAX, Pentium Pro, etc.	10) Eg. UltraSPARC, PowerPC etc.

Scanned with CamScanner