**Unit I**

**PART-A**

| **Q.No** | **Question** | **Text Book** | **Blooms Taxonomy level** | **CLO** |
| --- | --- | --- | --- | --- |
| 1. | 1. The \_\_\_\_\_\_ format is usually used to store data. **a) BCD** b) Decimal c) Hexadecimal d) Octal |  |  |  |
| 2. | A source program is usually in \_\_\_\_\_\_\_ a) Assembly language b) Machine level language **c) High-level language** d) Natural language |  |  |  |
| 3. | The small extremely fast, RAM’s are called as \_\_\_\_\_\_\_ **a) Cache** b) Heaps c) Accumulators d) Stacks |  |  |  |
| 4. | \_\_\_\_\_\_ bus structure is usually used to connect I/O devices. **a) Single bus** b) Multiple bus c) Star bus d) Rambus |  |  |  |
| 5. | The Input devices can send information to the processor. **a) When the SIN status flag is set** b) When the data arrives regardless of the SIN flag c) Neither of the cases d) Either of the cases |  |  |  |
| 6. | The decoded instruction is stored in \_\_\_\_\_\_ **a) IR** b) PC c) Registers d) MDR |  |  |  |
| 7. | Which of the register/s of the processor is/are connected to Memory Bus? a) PC **b) MAR** c) IR d) Both PC and MAR |  |  |  |
| 8. | ISP stands for \_\_\_\_\_\_\_\_\_ **a) Instruction Set Processor** b) Information Standard Processing c) Interchange Standard Protocol d) Interrupt Service Procedure |  |  |  |
| 9. | The internal components of the processor are connected by \_\_\_\_\_\_\_ a) Processor intra-connectivity circuitry **b) Processor bus** c) Memory bus d) Rambus |  |  |  |
| 10. | The registers, ALU and the interconnection between them are collectively called as \_\_\_\_\_ a) process route b) information trail c) information path **d) data path** |  |  |  |
| 11. | An optimizing Compiler does \_\_\_\_\_\_\_\_\_ a) Better compilation of the given piece of code **b) Takes advantage of the type of processor and reduces its process time** c) Does better memory management d) None of the mentioned |  |  |  |
| 12. | . When Performing a looping operation, the instruction gets stored in the \_\_\_\_\_\_ a) Registers **b) Cache** c) System Heap d) System stack |  |  |  |
| 13. | To reduce the memory access time we generally make use of \_\_\_\_\_\_ a) Heaps b) Higher capacity RAM’s c) SDRAM’s **d) Cache’s** |  |  |  |
| 14. | The time delay between two successive initiations of memory operation \_\_\_\_\_\_\_ a) Memory access time b) Memory search time **c) Memory cycle time** d) Instruction delay |  |  |  |
| 15. | During the execution of a program which gets initialized first? a) MDR b) IR **c) PC** d) MAR |  |  |  |
| 16. | The internal components of the processor are connected by \_\_\_\_\_\_\_ a)Processor intra-connectivity circuitry b)**Processor bus** c)Memory bus d) Rambus |  |  |  |
| 17. | In multiple Bus organisation, the registers are collectively placed and referred as \_\_\_\_\_\_ a) Set registers **b) Register file** c) Register Block d) Map registers |  |  |  |
| 18. | he ISA standard Buses are used to connect \_\_\_\_\_\_\_\_\_\_\_ a) RAM and processor b) GPU and processor c) **Harddisk and Processor** d) CD/DVD drives and Processor |  |  |  |
| 19. | An optimizing Compiler does \_\_\_\_\_\_\_\_\_ a) Better compilation of the given piece of code **b) Takes advantage of the type of processor and reduces its process time** c) Does better memory management d) None of the mentioned |  |  |  |
| 20. | When Performing a looping operation, the instruction gets stored in the \_\_\_\_\_\_ a) Registers **b) Cache** c) System Heap d) System stack |  |  |  |
| 21. | The circuit used to store one bit of data is called   1. Registers 2. Encoder 3. Decoder 4. **Flip flop** |  |  |  |
| 22. | The average time required to reach a storage location in memory and obtain its content is   1. Turnaround time 2. **Access time** 3. Seek time 4. Transfer time |  |  |  |
| 23. | The addressing mode used in the instruction ADD X,Y is  a) Direct  b) Indirect  c) Absolute  **d) index** |  |  |  |
| 24. | A stack organised computer uses instruction of   1. **Zero addressing** 2. One addressing 3. Two addressing 4. Three addressing |  |  |  |
| 25. | An n-bit microprocessor has   1. n-bit program counter 2. **n-bit instruction register** 3. n-bit stack pointer 4. n-bit address register |  |  |  |
| 26. | The register that keeps track of the instructions of a program in the memory   1. Instruction Register 2. **Program Counter** 3. Index Register 4. Accumulator |  |  |  |
| 27. | The BSA instruction is   1. Branch and store accumulator 2. **Branch and save return address** 3. Branch and shift address 4. Branch and show accumulator |  |  |  |
| 28. | The load instruction is used to designate a transfer from memory to a processor register known as   1. **Accumulator** 2. Instruction register 3. Program counter 4. Index register |  |  |  |
| 29. | Status bit is also called as   1. Binary bit 2. **Flag bit** 3. Signed bit 4. Unsigned bit |  |  |  |
| 30. | What is the content of Stack Pointer(SP)?   1. Address of current instruction 2. Address of next instruction 3. **Address of top element of stack** 4. Size of stack |  |  |  |
| 31. | In assembly language programming, the minimum number of operands required for an instruction is/are   1. **Zero** 2. One 3. Two 4. Three |  |  |  |
| 32. | In which addressing mode the operand is given explicitly in the instruction   1. **Immediate** 2. Register 3. Direct 4. Indirect |  |  |  |
| 33. | The ALU makes use of \_\_\_\_\_\_\_ to store the intermediate results. **a) Accumulators** b) Registers c) Heap d) Stack |  |  |  |
| 34. | The control unit controls other units by generating \_\_\_\_\_\_\_\_\_\_\_ a) Control signals **b) Timing signals** c) Transfer signals d) Command Signals |  |  |  |
| 35. | The instruction -> Add LOCA, R0 does \_\_\_\_\_\_\_ a) Adds the value of LOCA to R0 and stores in the temp register b) Adds the value of R0 to the address of LOCA **c) Adds the values of both LOCA and R0 and stores it in R0** d) Adds the value of LOCA with a value in accumulator and stores it in R0 |  |  |  |
| 36. | The bus used to connect the monitor to the CPU is \_\_\_\_\_\_ a) PCI bus **b) SCSI bus** c) Memory bus d) Rambus |  |  |  |
| 37. | Add #45, when this instruction is executed the following happen/s \_\_\_\_\_\_\_ a) The processor raises an error and requests for one more operand **b) The value stored in memory location 45 is retrieved and one more operand is requested** c) The value 45 gets added to the value on the stack and is pushed onto the stack d) None of the mentioned |  |  |  |
| 38. | In the following indexed addressing mode instruction, MOV 5(R1), LOC the effective address is \_\_\_\_\_\_ a) EA = 5+R1 b) EA = R1 c) EA = [R1] **d) EA = 5+[R1]** |  |  |  |
| 39. | The effective address of the following instruction is MUL 5(R1,R2). a) 5+R1+R2 b) 5+(R1\*R2) **c) 5+[R1]+[R2]** d) 5\*([R1]+[R2]) |  |  |  |
| 40. | An 24 bit address generates an address space of \_\_\_\_\_\_ locations. a) 1024 b) 4096 c) 248 **d) 16,777,216** |  |  |  |
| 41. | The type of memory assignment used in Intel processors is \_\_\_\_\_ **a) Little Endian** b) Big Endian c) Medium Endian d) None of the mentioned |  |  |  |
| 42. | RTN stands for \_\_\_\_\_\_\_\_\_\_\_ **a) Register Transfer Notation** b) Register Transmission Notation c) Regular Transmission Notation d) Regular Transfer Notation |  |  |  |
| 43. | The instruction, Add R1,R2,R3 in RTN is \_\_\_\_\_\_\_ a) R3=R1+R2+R3 b) R3<-[R1]+[R2]+[R3] c) R3=[R1]+[R2] **d) R3<-[R1]+[R2]** |  |  |  |
| 44. | The two phases of executing an instruction are \_\_\_\_\_\_\_\_\_\_ a) Instruction decoding and storage **b) Instruction fetch and instruction execution** c) Instruction execution and storage d) Instruction fetch and Instruction processing |  |  |  |
| 45. | When using Branching, the usual sequencing of the PC is altered. A new instruction is loaded which is called as \_\_\_\_\_\_ **a) Branch target** b) Loop target c) Forward target d) Jump instruction |  |  |  |
| 46. | \_\_\_\_\_\_\_\_\_\_ converts the programs written in assembly language into machine instructions. a) Machine compiler b) Interpreter **c) Assembler** d) Converter |  |  |  |
| 47. | The alternate way of writing the instruction, ADD #5,R1 is \_\_\_\_\_\_ a) ADD [5],[R1]; **b) ADDI 5,R1;** c) ADDIME 5,[R1]; d) There is no other way |  |  |  |
| 48. | \_\_\_\_\_ the most suitable data structure used to store the return addresses in the case of nested subroutines. a) Heap **b) Stack** c) Queue d) List |  |  |  |
| 49. | The system is notified of a read or write operation by \_\_\_\_\_\_\_\_\_\_\_ a) Appending an extra bit of the address b) Enabling the read or write bits of the devices c) Raising an appropriate interrupt signal **d) Sending a special signal along the BUS** |  |  |  |
| 50. | The method of synchronising the processor with the I/O device in which the device sends a signal when it is ready is? a) Exceptions b) Signal handling **c) Interrupts** d) DMA |  |  |  |

**PART-B**

| **Q.No** | **Question** | **Text book** | **Blooms Taxonomy Level** | **CLO** |
| --- | --- | --- | --- | --- |
| 1. | Explain the various functional units of a computer. |  |  |  |
| 2. | Explain the various types of Memory System. |  |  |  |
| 3. | Explain the role of registers – PC, IR, MAR and MDR in processor. |  |  |  |
| 4. | What is a Bus? Explain Single and Multiple Bus Structure. |  |  |  |
| 5. | Explain Register Transfer Notation with examples. |  |  |  |
| 6. | Explain Assembly Language Notation with examples. |  |  |  |
| 7. | What are the different types of Instruction Format? Explain each with an example. |  |  |  |
| 8. | Differentiate RISC and CISC. |  |  |  |
| 9. | What are Assembler Directives? Give an example. |  |  |  |
| 10 | Explain various Processor and CPU cores in ARM processor. |  |  |  |

**PART-C**

| **Q.No** | **Question** | **Text book** | **Blooms Taxonomy Level** | **CLO** |
| --- | --- | --- | --- | --- |
| 1. | Explain Instruction Execution in Straight Line Sequencing and Branching. |  |  |  |
| 2. | Describe various Addressing Modes with suitable examples. |  |  |  |
| 3. | Explain basic I/O operations in detail. |  |  |  |
| 4. | Explain Memory location, memory addresses and memory operation in detail. |  |  |  |
| 5. | Define Microprocessor. Explain the evolution of Microprocessors in detail. |  |  |  |
| 6. | Explain Assembly Language Program with an example. |  |  |  |
| 7. | Explain ARM processor and Thumb instruction set in detail. |  |  |  |
| 8. | Explain Instruction encoding format for Load and Store instructions in ARM processor. |  |  |  |