# HARDWARE PAPER 3

## Section A

1) Explain the role of firmware in controlling hardware devices in embedded systems.
2) Discuss the trade-offs between using DRAM and SRAM in system memory design.
3) How does the architecture of a modern microprocessor optimize memory access?
4) Describe the process of garbage collection in solid-state drives (SSDs).
5) Explain the significance of wear leveling algorithms in flash memory devices.
6) How does error correction work in NAND flash memory?
7) Describe the architecture of a RAID storage system and its benefits in data redundancy.
8) Explain how a microprocessor-controlled environment handles real-time data inputs from sensors.
9) How does a 3D printer utilize G-code to produce physical objects?
10) Describe the principles of heat dissipation in high-performance computing systems.
11) Explain the process of data encryption in solid-state drives.
12) Discuss the advantages and challenges of using embedded systems in automotive applications.
13) How does dynamic voltage scaling improve the efficiency of modern processors?
14) Explain the role of digital signal processors (DSPs) in audio and video processing.
15) How does an operating system manage memory using virtual memory techniques?
16) Describe the process of accessing data in a non-volatile memory device.
17) Explain the significance of data integrity checks in high-density storage devices.
18) How does a microcontroller differ from a microprocessor in embedded systems?
19) Discuss the limitations of using flash memory in industrial applications.
20) Explain the concept of triple-level cell (TLC) NAND memory and its trade-offs.
21) How does wear-out in NAND flash memory impact long-term storage reliability?
22) What is the role of a memory management unit (MMU) in modern processors?
23) Describe the impact of data fragmentation on solid-state drives (SSDs).
24) How does a real-time operating system (RTOS) manage task scheduling in embedded systems?
25) Discuss the trade-offs between cost and performance in various types of RAM.
26) How does an operating system handle memory paging and segmentation?
27) What is the importance of address translation in memory management?

28)Describe the impact of memory interleaving on system performance.

29)How does an ADC convert analog signals to digital data in a control system?

30)Explain the architecture of a NAND gate and how it is used to create logic circuits.

31)How does static RAM (SRAM) differ from dynamic RAM (DRAM) in terms of power consumption?

32)Describe the process of refreshing data in a DRAM chip.

33)How does a microprocessor handle data inputs from multiple sensors simultaneously?

34)Explain the role of cache memory in improving processor performance.

35)How does a multi-core processor manage memory access across multiple cores?

36)Describe the differences between synchronous and asynchronous memory access.

37)Explain the process of address decoding in a memory system.

38)Discuss the benefits of using NOR flash memory in embedded systems.

39)How does a memory controller optimize data access in a multi-channel memory system?

40)Explain the role of error-correcting code (ECC) in maintaining data integrity in memory.

41)How does a virtual memory system manage page faults?

42)Describe the process of writing data to a flash memory chip.

43)Explain the architecture of a microprocessor-controlled robotic system.

44)How does an ADC improve the accuracy of sensor data in a control system?

45)Discuss the limitations of multi-level cell (MLC) flash memory in terms of reliability.

46)How does a thermal sensor control the cooling system in a high-performance computer?

47)Explain the significance of power gating in reducing energy consumption in modern processors.

48)Describe the impact of quantum tunneling on the miniaturization of memory chips.

49)How does a microprocessor handle interrupts in real-time applications?

50)Discuss the trade-offs between power consumption and performance in mobile DRAM chips.?

# Section B

1) How does a massively parallel computer differ from a multicore system?
2) Explain how pipelining can lead to instruction-level parallelism in a CPU.
3) What is a hazard detection unit in a CPU, and how does it resolve data hazards?
4) How does a RISC processor handle multiple pipeline stages?
5) What are the trade-offs between CISC and RISC architectures in terms of power consumption?
6) Explain the concept of speculative execution and its role in pipelining.
7) How do SIMD and MIMD architectures handle parallelism differently?
8) Describe the role of vector processors in SIMD architecture.
9) How does instruction pipelining reduce the effective instruction cycle time?
10) Explain how parallelism in GPU architectures differs from CPU parallelism.
11) What is the importance of instruction-level parallelism in superscalar processors?
12) How does the design of a CPU’s instruction set affect its performance?
13) Explain how cache coherence is maintained in multicore processors.
14) What is the significance of a load/store unit in a RISC processor?
15) How do modern CPUs manage branch prediction in pipelines?
16) Explain the architecture of a multicore system with shared memory.
17) How does memory latency affect the performance of parallel processors?
18) What are the challenges of parallelizing tasks in SIMD processors?
19) How does instruction decoding in a CISC processor differ from that in a RISC processor?
20) What is the role of a pipeline register in CPU architecture?
21) How does out-of-order execution improve the performance of modern processors?
22) Describe the role of a branch target buffer in branch prediction.
23) What are the key architectural features of a superscalar CPU?
24) How does a CPU handle nested interrupts?
25) What are the advantages of using an EPIC architecture in modern processors?
26) Explain how a multicore processor handles task-level parallelism.
27) What are the challenges of implementing parallelism in modern CPUs?

28) How does a memory controller manage multiple memory access requests?
29) Explain the role of hyper-threading in modern processors.
30) How does a processor handle interrupts during instruction pipelining?
31) What is the significance of vector registers in SIMD architectures?
32) How does a CPU manage speculative execution to improve performance?
33) What is the role of a branch predictor in modern CPUs?
34) Describe how GPUs handle data-parallel tasks using SIMD architecture.
35) Explain the concept of memory consistency in parallel processing systems.
36) How does the memory hierarchy in a CPU affect its performance?
37) What are the trade-offs between pipelining and superscalar execution?
38) How does a CPU resolve control hazards in instruction pipelines?
39) What is the significance of multithreading in modern processors?
40) How does a CPU manage multiple cores in a shared memory system?
41) What are the challenges of designing an efficient pipelined processor?
42) How does a processor manage cache coherence in a multicore system?
43) What is the impact of branch misprediction on pipelined instruction execution?
44) Explain how superscalar architecture achieves instruction-level parallelism.
45) What are the limitations of SIMD architecture in general-purpose computing?
46) How does a memory barrier affect the execution of instructions in parallel processors?
47) What is the role of a prefetch buffer in pipelined instruction execution?
48) How does a CPU handle exceptions in a pipelined architecture?
49) What is the significance of pipeline stalls in modern CPU designs?
50) How does a CPU implement dynamic scheduling to improve performance?