**EASWARI ENGINEERING COLLEGE**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**IT2354 – Embedded System**

**UNIT –I**

**PART-A(2 MARKS)**

1. What is an embedded computer system?

2. Why do we use microprocessors to design a digital system?

3. Mention the challenges in embedded computing system design.

4. Mention the reasons that makes embedded computing machines design difficult.

5. State the importance of design methodology.

6. Mention the major steps in embedded system design process.

7. Mention the major goals of embedded system design.

8. Mention the non functional requirements.

9. Mention the components of GPS system specification.

10. Mention the different types of relationships.

11. What is called a von Neumann machine?

12. What is called a Harvard machine?

13. Mention the characteristics of instructions.

14. State the importance of current program status register (CPSR).

15. Mention the addressing modes of C55x DSP.

16. Define procedure linkage.

17. Define CISC.

18. Define RISC.

19. Mention the features of assembly language.

20. Differentiate big and little endian byte ordering modes.

**PART-B(16 MARKS)**

1. Explain in detail about the challenges in embedded computing system design.

2. Explain in detail about the embedded system design process.

3. Explain in detail about ARM processor.

4. Explain in detail about TI C55x DSP.

5. Explain in detail about the characteristics of embedded computing applications.

6. Explain Structural description in detail.

7. Explain Behavioral description in detail.

8. Explain Conceptual specification in detail.

9. Explain in detail about 8051 microcontroller.

10. Explain in detail about data operations of ARM processor.

**UNIT II**

**PART-A(2 MARKS)**

1. State the importance of data register and status register.

2. Mention the two ways used for performing input and output operations.

3. Define polling.

4. Define an interrupt.

5. Mention the signals used for by i/o devices for interrupting.

6. Define foreground program.

7. Mention the ways used for generalizing the interrupts to handle multiple devices.

8. Define non maskable interrupts.

9. Define exception.

10. Define trap.

11. Define cache memory.

12. Define cache hit.

13. Define cache miss.

14. Mention the types of cache misses.

15. Mention the different strategies used for writing in a cache.

16. Define page fault.

17. Define DMA.

18. Mention the registers present in the DMA controller.

19. What is a watch dog timer?

20. Define aspect ratio.

**PART-B(16 MARKS)**

1. Explain the concept of interrupts in detail.

2. Explain the working of cache memory in detail.

3. Explain memory mapping and address translation in detail.

4. Explain the working of CPU bus in detail.

5. Explain direct memory access in detail.

6. Explain the various I/O devices in detail.

7. Explain memory devices in detail.

8. Explain the various display devices and the methods of interfacing in detail.

9. Explain about exceptions and trap in detail.

10. Explain about interrupt priority and vector in detail.

**UNIT –III**

**PART-A(2 MARKS)**

1. Define process.

2. Define thread.

3. Mention the requirements on processes.

4. Define period.

5. Define task graph.

6. Define initiation time and completion time.

7. Mention the various scheduling states of a process.

8. Define scheduling policy.

9. Define utilization.

10. Define time quantum.

11. Define context switching.

12. Mention the two ways of assigning priority to a process.

13. Define rate monolithic scheduling.

14. Define earliest deadline first scheduling.

15. Define priority inversion.

16. Mention the two different styles used for inter process communication.

17. Define signal.

18. Define response time.

19. Define PCB.

20.Define critical instant.

**PART– B(16 MARKS)**

1. Explain multiple tasks and multiple processes in detail.

2. Explain the various scheduling policies in detail.

3. Explain Preemptive real time operating system in detail.

4. Explain Non-Preemptive real time operating systems in detail.

5. Explain priority based scheduling in detail.

6. Explain the various inter process communication mechanism in detail.

7. Explain the various types of Performance issues.

**UNIT-IV**

**PART-A(2 MARKS)**

1. List the advantages in assembly language coding for an application.

2. What do you mean by optimization of memory?

3. How to declare NULL pointers in embedded C?

4. Define exe file.

5. What is an in-circuit emulator?

6. What are reentrant functions in embedded software?

7. Define a testbench.

8. Define a host and a target.

9. What is a cross compiler?

10. What are real-time constraints for an embedded system?

11. Specify the 2 categories of multi-state systems.

12. What is a linker?

13. What is a locator?

14. What is a simulator?

15. What are the debugging tools available?

16. What type of processor, operating system and language are used for embedded systems?

17. List the advantages in high-level language programming.

18. What are configuration files?

19. How does a macro differ from a function?

20. How will you optimize embedded system codes in OOPs language?

**PART-B(16 MARKS)**

1. a) Tabulate program elements: Macros and Functions and their uses.

b) Explain the use of pointers, NULL pointers.

2. a) What are the advantages of using multiple function calls in cyclic order in the main() ?

Also, write the advantages of building ISR queues.

b) What are the programming advantages of C++ ? What are the disadvantages of C++ ?

3. a) Compare programming in assembly language and in high level language ‘C’.

b) What are the typical challenges of programming for embedded systems?

4. a) Compare C program compiler and cross-compiler.

b) Discuss the steps used for optimizing the use of memory in a system.

5. a) List the various software tools of embedded system and its uses.

b) Explain Software tools application with exemplary systems.

6. Explain the following program elements with suitable syntax and examplesa)

Include directories

b) Source files

c) Configuration files

d) Preprocessor directives

7. a) Discuss about the usage of function calls in embedded C language.

b) Discuss about multiple function calls in cyclic order.

8. a) Describe the debugging strategies used in embedded systems in detail.

b) what are the differences between linker and loader?

9. Explain the features of assemblers, compilers and cross-compilers used in embedded systems.

10. a) What are simulators? What are the advantages and disadvantages of simulators?

b) Describe the role of in-circuit emulator in the design of embedded system.

**UNIT-V**

**PART-A(2 MARKS)**

1. Why is UML a powerful modeling language?

2. What is a synchronization object?

3. List the various software layers in software architecture of a camera system.

4. Define Hardware Architecture.

5. What are the 2 different approaches for designing an embedded system?

6. List the issues in hardware and software design for an embedded system.

7. Specify the advantages of hardware implementations.

8. Specify the advantages of software implementations.

9. List the choices available for embedded system processors.

10. Mention the language used for simulating and synthesizing gate level design.

11. How will you implement FSM and state transitions?

12. List any 4 factors which should be taken into account while choosing a microprocessor or microcontroller.

13. Specify few data sets which are allotted memory

14. What are the registers in a device?

15. List the various performance metrics.

16. List the metrics based on which real-time program performance depends upon.

17. What are the ways in which you accelerate the performance?

18. What does a PLC unit consist of?

19. How does a port instruction data type differ from one platform to another platform?

20. What are the criteria to be considered about which the microcontroller is to be used?

**PART-B(16 MARKS)**

1. Draw and explain basic system(ACVS) of an Automatic Chocolate Vending Machine.

2. Identify the tasks for an ACVS. Explain the various interprocess communication

methods required in implementing the application.

3. Perform the case study for an ACVS using MUCOS RTOS.

4. Perform the case study of an embedded system for a smart card.

5. What are the hardware and software design issues to be considered while designing an embedded system?