Question Bank on RTSA

Chapter-I: Introduction

1. Consider a payroll processing system for an elevator company. Describe three different scenarios in which the system can be justified as hard, firm, or soft real - time.
2. Discuss whether the following are hard, firm, or soft real - time systems:

a) The Library of Congress print - manuscript database system.
b) A police database that provides information on stolen automobiles.
c) An automatic teller machine in a shopping mall.
d) A coin - operated video game in some amusement park.
e) A university grade - processing system.
f) A computer - controlled routing switch used at a telephone company branch exchange.

3. Consider a real - time weapons control system aboard a fighter aircraft. Discuss which of the following events would be considered synchronous and which would be considered asynchronous to the real - time computing system.
   (a) A 5 - ms, externally generated clock interrupt.
   (b) An illegal - instruction - code (trap) interrupt.
   (c) A built - in - test memory failure.
   (d) A discrete signal generated by the pilot pushing a button to fi re a missile.
   (e) A discrete signal indicating " low on fuel. "
4. Describe a system that is completely nonreal - time, that is, there are no bounds whatsoever for any response time. Do such systems exist in reality?
5. For the following systems concepts, fill in the cells of Table as shown below with descriptors for possible events. Estimate event periods for the periodic events.

TABLE Taxonomy of Events and Some Typical Examples

	Periodic	Aperiodic	Sporadic
Synchronous	Cyclic code	Conditional branch	Divide-by-zero (trap) interrupt
Asynchronous	Clock interrupt	Regular, but not fixed-period interrupt	Power-loss alarm

   (a) Elevator group dispatcher: this subsystem makes optimal hall – call allocation for a bank of high - speed elevators that service a 40 – story building in a lively city like Louisville.
   (b) Automotive control: this on - board crash avoidance system uses data from a variety of sensors and makes decisions and affects behavior to avoid collision, or protect the occupants in the event of an imminent collision. The system might need to take control of the automobile from the driver temporarily.
6. For the example systems introduced (inertial measurement, nuclear - power - plant monitoring, airline reservation, pasta bottling, and traffic -light control) enumerate some possible events and note whether they are periodic, aperiodic, or sporadic. Discuss reasonable response times for the events.
7. In the response – time calculation as shown below, the time from observing a passenger between the closing door blades and starting to reopen the elevator door varies between 305 and 515 ms. How could you further justify if these particular times are appropriate for this situation?

Sensor Response Time: $t_{S\_min} = 5$ ms, $t_{S\_max} = 15$ ms, $t_{S\_mean} = 9$ ms.

Hardware Response Time: $t_{HW\_min} = 1$ µs, $t_{HW\_max} = 2$ µs, $t_{HW\_mean} = 1.2$ µs.

System Software Response Time: $t_{SS\_min} = 16$ µs, $t_{SS\_max} = 48$ µs, $t_{SS\_mean} = 37$ µs.

Application Software Response Time: $t_{AS\_min} = 0.5$ µs, $t_{AS\_max} = 0.5$ µs, $t_{AS\_mean} = 0.5$ µs.

Door Drive Response Time: $t_{DD\_min} = 300$ ms, $t_{DD\_max} = 500$ ms, $t_{DD\_mean} = 400$ ms.

the minimum, maximum, and mean values of the composite response time: $t_{min} \approx 305$ ms, $t_{max} \approx 515$ ms, and $t_{mean} \approx 409$ ms.

8. A control system is measuring its feedback quantity at the rate of 100 µ s. Based on the measurement, a control command is computed by a heuristic algorithm that uses complex decision making. The new command becomes available 27 – 54 µ s (rather evenly distributed) after each sampling moment. This considerable jitter introduces harmful distortion to the controller output. How could you avoid (reduce) such a jitter? What (if any) are the drawbacks of your solution?
9. Consider the CPU utilization factor, how short could the execution period of Task 1, $e1$, be made to maintain the CPU utilization zone no worse than " questionable "? Refer to table given below.

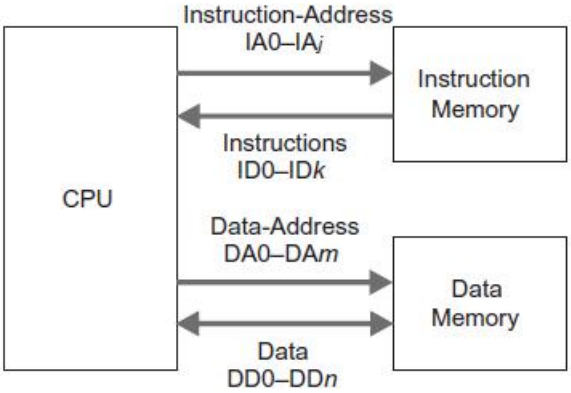
TABLE CPU Utilization (%) Zones

Utilization (%)	Zone Type	Typical Application
<26	Unnecessarily safe	Various
26–50	Very safe	Various
51–68	Safe	Various
69	Theoretical limit	Embedded systems
70–82	Questionable	Embedded systems
83–99	Dangerous	Embedded systems
100	Critical	Marginally stressed systems
>100	Overloaded	Stressed systems

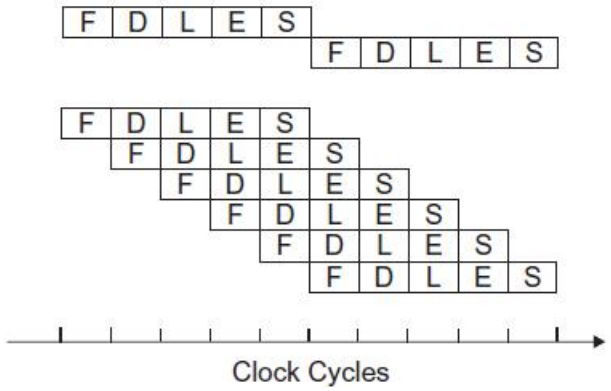
Chapter- II: Hardware

1. Compose a table providing the available memory spaces for the following address - bus widths: 16, 20, 24, and 32 bits.
2. It is common practice for programmers to create continuous test - and -loop code in order to poll I/O devices or wait for interrupts to occur. Some processors provide an instruction (WAIT or HALT) that allows the processor to hibernate until an interrupt occurs. Why is the latter form more efficient and desirable?
3. In general terms, suggest a possible scheme that would allow a machine - language instruction to be interruptible. What would be the overall effect on instruction ' s execution time and CPU ' s throughput and response times?
4. Figure 2.5 illustrates the interface lines of a generic memory component. Assume $m = 15$ and $n = 7$. The address bus of your microprocessor is 24 bits wide. How, in principle, could you locate this particular memory block to begin from the address 040000 (hexadecimal)? What is the corresponding end address?

5. Compare and contrast the different memory technologies discussed in this chapter as they pertain to embedded real - time systems.
6. How would you test the validity and integrity of factory parameters stored in EEPROM? Sketch a suitable procedure for that purpose.
7. Assume a hierarchical memory system having a joint instruction/data cache with a memory - access cost of 10 ns on a hit and 90 ns on a miss. An alternative design without hierarchical memory architecture has a memory - access cost of 70 ns. What is the minimum cache - hit percentage that would make the hierarchical memory system useful?
8. The Harvard architecture as shown in figure below offers separate address and data buses for instruction codes and data. Why is not it feasible to have separate buses for programmed I/O as well?



9. Show with an illustrative example how the five - stage pipeline as shown in figure below, what could the benefit from the Harvard architecture.

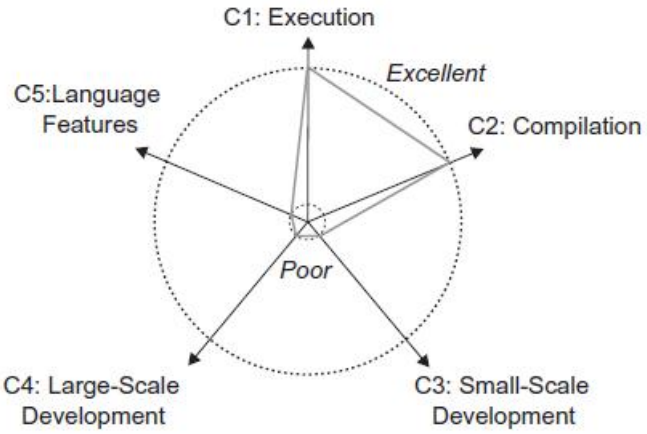


10. What special problems do super-pipelined and superscalar architectures pose for real - time system designers? Are they any different for nonreal - time systems?
11. In CISC - type processors, most instructions are having memory operands, while RISC - type processors access memory by LOAD and STORE instructions only. What are the advantages and disadvantages of both schemes?
12. You are designing the architecture of a high - performance CPU for hard real - time applications. List and justify the principal architectural selections that you would make.
13. Discuss the relative advantages and disadvantages of memory – mapped I/O, programmed I/O, and DMA as they pertain to real - time systems.
14. Why is DMA controller access to main memory in most systems given higher priority than CPU access to main memory?

15. An embedded system has a 12 - bit A/D converter for measuring voltages between − 10 V and + 10 V. What is the digital value corresponding to + 5.6 V?
16. Find a microcontroller with unique, special instructions and, considering the application area for that processor, discuss the need for those special instructions.
17. What are the advantages of systems on chip over computers on chip? Find a few examples of commercial systems on chip from the Web.
18. A watchdog timer is used for supervising the operation of an embedded system in a high - EMI environment. Why is it practical to connect the watchdog - circuit ' s output to the CPU 's non-maskable (instead of maskable) interrupt input?
19. List the different data - transmission media mentioned in this chapter and give typical applications for each.

Chapter- III: Programming Language

1. It can be argued that in some cases there exists an apparent conflict between good software engineering practices and real - time performance. Consider the relative merits of recursive program design versus interactive techniques, and the use of global variables versus parameter lists. Using these topics and an appropriate programming language for examples, compare and contrast real - time performance versus good software engineering practices as you understand them.
2. What programming restrictions should be used in a programming language to permit straightforward analysis of real - time applications?
3. Write a set of coding standards for use with any of the real - time applications for the programming language of your choice. Document the rationale for each provision of the coding standard.
4. Why is it very important to cite the reference for any computational algorithm that is used in a real - time program in the program 's annotation?
5. In a procedural language of your choice, develop an abstract data type called "image" with associated functions. Be sure to follow the principle of information hiding. Make any assumptions that you need to about the properties of the images.
6. In the object - oriented language of your choice, design and code an " image " class that could be useful across a wide range of projects. Be sure to follow the best principles of object - oriented design.
7. How can misuse or misunderstanding of a software technology impede a software project? For instance, writing structured C code instead of classes in C ++ , or reinventing a tool for each project instead of using a standard one.
8. Java has been compared with Ada 95 in terms of "hype" and "unification " — defend or refute the arguments for this comparison.
9. By using the five metrics of Cardelli, compare the fitness of C and C ++ languages for real - time programming; use pentacle diagrams as shown below for visualizing your justified comparison.



10. Are there any language features that are exclusive to C/C ++ ? Do these features provide specific advantage or disadvantage in embedded environments?
11. You are hired to defi ne a set of principal requirements for a new real -time programming language for embedded control applications. What are the most important requirements that your defi nition would contain? Justify your answer.
12. What compiler options are available in your favorite C compiler and what do they specifically do?
13. Develop a set of tests to exercise a compiler to determine the best use of the language in a real - time processing environment. For example, your tests should determine such things as when to use case statements versus nested if-then-else statements; when to use integers versus Boolean variables for conditional branching; whether to use while or for loops, and when; and so on.
14. Use standard compiler - optimization methods and multiple optimization phases to optimize the following C code by hand:

```
#define UNIT 1
#define FULL 1
void main(void)
{
int a,b;
a=FULL;
b=a;
if ((a ==FULL) && (b ==FULL))
{
if (debug)
printf("a=%d b =%d",a,b);
a=(b*UNIT)/2;
a=2.0*a*4;
b=b*sqrt(a);
}
}
```

MORE QUESTIONS

1. Why hard real-time task is used in sophisticated Real-Time System?
2. State the impact incurred in Hard Real-Time System if it misses a deadline?
3. Among Missile Guide System and Satellite system Lunching, which is(are) hard Real-Time System?
4. State 2 cases where firm Real-Time System can be used.
5. What is response time?
6. What is release time?
7. State any relation between that may exists between Response time and Release Time.
8. What criteria for fail-safe system?
9. Arrange the following memories in order of their increasing performance.
10. Optical Disk, HDD, Magnetic Tape, Flash ROM.
11. If it required to have RAM and ROM of sizes 64kB and 1MB to be interfaced with processor. If lower address is given to RAM and address for both are in continuous location. Find starting and end address for both assuming starting address is 0x4000000.
12. The starting and end addresses of RAM and ROM are (0x100000, 0x100FFF) and (0x300000, 0x3F122FF) respectively. Calculate the size of RAM and ROM assuming data width is 8 bit.
13. In what respect Von-Neumann architecture is better than Harvard Architecture?
14. Explain Interrupt driven I/O based data transfer.
15. Among the Interrupt driven I/O and DMA base I/O, which is better?
16. State the performance improvement using 5 stage instruction pipeline over non-pipeline instruction execution.
17. If a processor takes 1 clock cycle for each instruction using 9 stage pipelines. If a program consists of 765 instructions, then state how many cycles it will take to execute the program.
18. To determine the fitness of a programming language, which parameters are taken into consideration?
19. Is there any improvement in programming when some functions are called which are written in assembly code within high-level program?
20. State 2 advantages of high level language over assembly languages.
21. Is object oriented programming is better real-time application development? If yes, explain briefly.
22. How concurrency in Real-Time System modelling improves performance?

23. Differentiate between Petrinet and Statechart.
24. What state chart?
25. What is an history mechanism used in state chart?
26. Why RTOS is necessary for Rela-time System?
27. State 2 parameters that is available in RTOS.
28. What is a counting semaphore?
29. For synchronization of different tasks, which functions of RTOS are used.
30. A hard real-time system application is made only for hard real-time task. If yeas, why?
31. What will happen when a hard real-time system mises is deadline?
32. Taking deadline as a parameter, state different real-time systems exist.
33. Every safety critical system should be hard real-tiem system. If yes, why?
34. What is response time?
35. What is release time?
36. Is Response time and release times being inter related? Give a suitable explanation to it. Among response time and release time which is larger and why?
37. What do mean by a failed system?
38. Why cache memory is faster than RAM though both are SRAM cells?
39. If it required to have RAM and ROM of sizes 64kB and 1MB to be interfaced with processor. The RAM and ROM starting addresses are 0x0F00000 and ROM 0x1F00000. Calculate no of address lines needed to interface and also the end address for both.
40. The starting and end addresses of RAM and ROM are (0x100000, 0x100FFF) and (0x300000, 0x3F122FF) respectively. Calculate the size of RAM and ROM assuming data width is 8 bit.
41. Why Harvard architecture gives faster performance than Von-Neumann architecture?
42. Among the programmed I/O and Interrupt driven I/O, which is better?
43. Among the Interrupt driven I/O and DMA base I/O, which is better?
44. Why instruction pipeline gives better performance than non-pipeline instruction execution?
45. If a processor takes 1 clock cycle for each instruction using 9 stage pipeline. If a program consists of 765 instructions, then state how many cycle it will take to execute the program.
46. To determine the fitness of a programming language, which parameters are taken into consideration?

47. Why high level languages are better than assembly language for development of real-time application?
48. State 2 advantages of assembly language over high level languages.
49. Is object oriented programming is better real-time application development? If yes, explain briefly.
50. Why concurrency need in Real-time system application development?
51. Why state chart modelling is better than FSM modelling?
52. What Petri net?
53. What is an OR-Superstate in state chart modelling?
54. Why RTOS is necessary for Real-time System?
55. State 2 parameters that is available in RTOS.
56. What is semaphore?
57. What is need of message queue in RTOS?
58. Differentiate between deadlock and livelock.
59. Describe a system that is completely nonreal - time, that is, there are no bounds whatsoever for any response time. Do such systems exist in reality?
60. Automotive control: this on - board crash avoidance system uses data from a variety of sensors and makes decisions and affects behaviour to avoid collision, or protect the occupants in the event of an imminent collision. The system might need to take control of the automobile from the driver temporarily.
61. Calculate the minimum and maximum response time for the data given below.
62. With neat diagram explain memory hierarchy? Why the external hard disk has more response time than a semiconductor memory?
63. Every real-time system should have feedback path. Why this is essential for the Real-Time System Design?
64. Describe the following terms in relation to Real-Time System: Response Time, Release time, jazzle time
65. A CPU supports 3-stage and 5-stage pipeline. What is the speed achieved over 3-stage pipeline when a 5-stage pipeline is used? With suitable diagram explain the 5-stage pipeline operation.
66. Differentiate between Superscalar and Very Long Instruction Word architecture.
67. Differentiate between Von Neumann and Harvard CPU Architecture.
68. Differentiate between DMA based I/O and Interrupt driven I/O data transfer.
69. With neat diagram explain Interrupt mode of data transfer.

70. Why DMA mode data transfer offers better system performance than other mode of data transfer that you know?
71. With neat diagram explain the DMA mode of data transfer.
72. State and explain 5 properties of RTOS.
73. With neat diagram explain Round Robbin scheduling.
74. State different objects of a Kernel. Explain Message Queue and how it helps in task synchronization.
75. Why multitasking is an essential requirement for Real-Time System? With neat diagram, explain the multitasking.
76. Explain how an RTOS ensures the dead line of different tasks?
77. With neat diagram explain the Pre-emptive scheduling.
78. With neat diagram explain different parts of a petrinet.
79. Explain a traffic light control using petrinet and draw neat diagram for it.
80. Differentiate between FSM and state chart.
81. With neat petri-net diagram solve the given equation.
82. What is the difference between OR-Superstate and AND-Superstate? Use proper diagram to explain.
83. Discuss whether the following are hard, firm, or soft real - time systems:
    i. The Library of Congress print - manuscript database system.
    ii. A police database that provides information on stolen automobiles.
    iii. An automatic teller machine in a shopping mall.
    iv. A coin - operated video game in some amusement park.
    v. A university grade - processing system.
    vi. A computer - controlled routing switch used at a telephone company branch exchange.
84. For the example systems introduced (inertial measurement, nuclear - power - plant monitoring, airline reservation, pasta bottling, and traffic -light control) enumerate some possible events and note whether they are periodic, aperiodic, or sporadic. Discuss reasonable response times for the events.
85. Calculate the minimum and maximum response time for the data given below.
86. With neat diagram explain memory hierarchy? Why the external hard disk has more response time than a semiconductor memory?
87. State and explain 4 properties of real-time systems.

88. State the different memory technology with their brief explanation.

89. With suitable diagram explain the 5 stage pipeline operation.

90. Differentiate between Superscalar and Very Long Instruction Word architecture.

91. Differentiate between RISC and CISC Architecture.

92. Differentiate between Programmed I/O and Interrupt driven I/O data transfer.

93. With neat diagram explain DMA mode of data transfer.

94. Why DMA mode data transfer offers better system performance than other mode of data transfer that you know?

95. With neat diagram explain the Interrupt driven data transfer.

96. State and explain 5 properties of RTOS.

97. With neat diagram explain cyclic scheduling.

98. In what way RTOS is different from GPOS?

99. Dispatcher and message queue are essential part of a RTOS. Why?

100. Explain how an RTOS ensures the dead line of different tasks?

101. With neat diagram explain the cyclic scheduling with pre-emption.

102. With neat diagram explain different parts of a state chart modelling.

103. Explain a digital stopwatch using state chart modelling.

104. Differentiate between state chart and petri-net.

105. With neat petri-net diagram solve the given equation.

$$Y = A + \frac{(B - C)}{BC}$$

106. What is the difference between OR-Superstate and AND-Superstate? Use proper diagram to explain.

107. With neat petri-net diagram solve the given equation.

$$Y = A + \frac{(2 * B + C)}{BC}$$

108. Problems on EDF, RMS, SJF, FCFS, Petrinet, State chart as discussed in class.