41 Embedded C Interview Questions to Ask Your Candidates

Questions

1. Can you explain the difference between a microcontroller and a microprocessor in the context of embedded systems?

2. How do you handle interrupt service routines in Embedded C, and why are they important?

3. What is the purpose of volatile keyword in Embedded C programming?

4. Describe the concept of memory-mapped I/O and its advantages in embedded systems.

5. How do you implement a circular buffer in Embedded C, and what are its applications?

6. Explain the difference between static and dynamic memory allocation in the context of embedded systems.

7. What are watchdog timers, and how are they used in embedded systems?

8. How do you optimize code for memory-constrained embedded systems?

9. Describe the concept of task scheduling in a real-time operating system (RTOS).

10. What are the key considerations when designing a device driver for an embedded system?

11. Can you explain the concept of bit manipulation and its importance in embedded systems?

12. How do you handle endianness issues in embedded systems?

13. Describe the concept of debouncing in embedded systems and how you would implement it.

14. How would you implement a simple state machine in Embedded C?

15. Explain the concept of memory alignment and its importance in embedded systems.

16. How would you implement a simple scheduler for a bare-metal embedded system?

17. Describe the concept of watchdog timers and how you would use them in an embedded system.

18. How would you approach debugging a hard fault in an ARM Cortex-M based system?

19. Can you explain how you would interface an ADC (Analog-to-Digital Converter) in an embedded system?

20. How do you manage power consumption in an embedded system when interfacing with hardware components?

21. Describe how you would implement communication between a microcontroller and a peripheral device using SPI (Serial Peripheral Interface).

22. What are the steps to configure and use a UART (Universal Asynchronous Receiver-Transmitter) for serial communication in Embedded C?

23. How do you handle hardware interrupts and prioritize them in an embedded system?

24. Explain how you would use DMA (Direct Memory Access) to optimize data transfer in an embedded system.

25. What considerations are important when designing an interface between a microcontroller and an external memory device?

26. How would you implement I2C (Inter-Integrated Circuit) communication in Embedded C?

27. Describe the process of setting up PWM (Pulse Width Modulation) for controlling motor speed in an embedded application.

28. How do you handle debouncing for a mechanical switch input in Embedded C?

29. What strategies do you use to ensure reliable data acquisition from sensors in an embedded system?

30. Can you explain how you would implement software to handle multiple simultaneous hardware events?

31. Describe how you would test and validate the hardware interfaces of an embedded system.

32. How do you manage and troubleshoot communication errors in embedded systems?

33. How would you implement a stack data structure in a memory-constrained embedded system?

34. Can you explain the concept of memory fragmentation and how it affects embedded systems?

35. How do you handle memory leaks in an embedded system with limited resources?

36. Explain the concept of memory overlays and when you might use them in embedded systems.

37. How would you optimize string handling in a memory-constrained embedded system?

38. What is the difference between const and #define in terms of memory usage in embedded systems?