

Embedded Systems Interview Questions

www.embeddedshiksha.com

Embedded C Programming Concepts:

- What is the difference between `#include <file.h>` and `#include "file.h"` in C?
- How is dynamic memory allocated in C, and what are the associated functions?
- Explain the use of bit manipulation in embedded C programming.
- Describe the importance of register-level programming in embedded systems.
- What is the role of the `volatile` keyword in embedded C?
- How do you access hardware peripherals in embedded C?
- Explain the purpose of the `const` keyword in C.
- Describe the differences between `float` and `double` data types in C.
- What is a macro in C, and how is it defined?
- How do you optimize C code for size or speed in embedded systems?

Embedded Systems and Microcontrollers:

- What is an embedded system, and how does it differ from a general-purpose computer?
- Explain the concept of interrupt handling in microcontrollers.
- What is a watchdog timer, and how is it used in embedded systems?
- Describe the purpose and functioning of the bootloader in embedded systems.
- What are the key factors to consider when choosing a microcontroller for an embedded project?
- How is power management achieved in battery-operated embedded devices?
- What is firmware, and how is it different from software?
- What is real-time operating system (RTOS), and why is it used in embedded systems?
- How does the PIC microcontroller handle interrupts?
- Describe the architecture and features of the ARM Cortex-M series.

C Language and Microcontroller Specific:

- Explain the use of volatile pointers in embedded C.
- How is power management achieved in battery-operated embedded devices?
- What is the difference between AVR and PIC microcontrollers?
- Describe the purpose of the SYSTICK timer in ARM Cortex-M microcontrollers.
- How does the PIC microcontroller handle interrupts?
- What is a memory-mapped register, and how is it accessed in C?
- Explain the use of inline assembly in embedded C code.
- What is the purpose of using inline assembly in embedded C code?
- How do you handle variable types that may have different sizes on different microcontrollers?
- What is the significance of the `volatile` keyword in C?
-

Memory and Optimization:

- Explain the concept of stack and heap in embedded C programming.
- What is the purpose of a linker script in embedded software development?
- Describe the use of an Integrated Development Environment (IDE) in embedded software development.
- How is software testing performed in embedded systems development?
- What is a memory-mapped register, and how is it accessed in C?
- How do you optimize C code for size or speed in embedded systems?
- Describe the concept of cross-compilation in embedded software development.
- How is power management achieved in battery-operated embedded devices?
- What is the role of version control systems like Git in embedded software development?
- Explain the concept of bit manipulation in embedded C programming.

Microcontroller Peripherals and Communication:

- Explain the differences between UART, SPI, and I2C communication protocols.
- How does the SPI (Serial Peripheral Interface) protocol work, and what are its advantages in embedded systems?
- Describe the role of the I2C (Inter-Integrated Circuit) protocol in embedded communication.
- What is the function of the Analog-to-Digital Converter (ADC) in microcontrollers?
- Explain the concept of PWM (Pulse-Width Modulation) in microcontroller applications.
- What are GPIO pins, and how are they used in microcontroller applications?
- Describe the purpose of timers and counters in microcontroller applications.
- Explain the use of interrupts for peripheral communication in embedded systems.
- How do you implement serial communication using the UART (Universal Asynchronous Receiver-Transmitter) module?
- Describe the concept of direct memory access (DMA) in microcontroller peripherals.