**Embedded System Design Sample Questions**

1. **What is resistor?**

**Ans:** A resistor is a two-terminal electronic component that opposes an electric current by producing a voltage drop between its terminals in proportion to the current, that is, in accordance with Ohm's law:

V = IR.

### Explain what is interrupt latency? How can you reduce it?

**Ans:**  Interrupt latency is a time taken to return from the interrupt service routine post handling a specific interrupt.  By writing minor ISR routines, interrupt latency can be reduced.

### Explain what is the need for an infinite loop in embedded systems?

### Ans:  Embedded systems require infinite loops for repeatedly processing or monitoring the state of the program.  For instance, the case of a program state continuously being verified for any exceptional errors that might just happen during run-time such as memory outage or divide by zero, etc.

### What Are Hard and Soft Real-Time Systems?

### Ans:The hard real-time Embedded Systems are the ones that depend on the output very strictly on time. Any late response or delay cannot be tolerated and will always be considered a failure. The soft real-time systems on the other are not very rigid as the hard real-time systems. The performance of the system degrades with the lateness of response, but it is bearable and can be optimized to a certain level for reuse of the result.

1. **Mention what are buses used for communication in embedded system (Communication Protocol )?**

**Ans:** For embedded system, the buses used for communication includes

• I2C: It is used for communication between multiple ICs

• CAN: It is used in automobiles with centrally controlled network

• USB: It is used for communication between CPU and devices like mouse, etc.

1. **What are the uses of the keyword static?**

Static keyword can be used with variables as well as functions. A variable declared static will be of static storage class and within a function, it maintains its value between calls to that function. A variable declared as static within a file, scope of that variable will be within that file, but it can't be accessed by other files. Functions declared static within a module can be accessed by other functions within that module. That is, the scope of the function is localized to the module within which it is declared.

1. **Why cannot arrays be passed by values to functions?**

In C, the array name itself represents the address of the first element. So, even if we pass the array name as argument, it will be passed as reference and not its address

1. **What is the use of volatile keyword?**

The C's volatile keyword is a qualifier that tells the compiler not to optimize when applied to a variable. By declaring a variable volatile, we can tell the compiler that the value of the variable may change any moment from outside of the scope of the program. A variable should be declared volatile whenever its value could change unexpectedly and beyond the comprehension of the compiler. In those cases it is required not to optimize the code, doing so may lead to erroneous result and load the variable every time it is used in the program. Volatile keyword is useful for memorymapped peripheral registers, global variables modified by interrupt service routine, global variables accessed by multiple tasks within a multi-threaded application.

1. **What are the uses of the keyword static?**

Static keyword can be used with variables as well as functions. A variable declared static will be of static storage class and within a function, it maintains its value between calls to that function. A variable declared as static within a file, scope of that variable will be within that file, but it can't be accessed by other files. Functions declared static within a module can be accessed by other functions within that module. That is, the scope of the function is localized to the module within which it is declared.

1. **What you mean by pass by reference and pass by value in functions ?**

In C, when you pass arguments to a function, they can be passed using two different methods: pass by value and pass by reference. These methods determine how the function interacts with the original variables that were passed as arguments.

Pass by Value:

When you pass an argument by value to a function, a copy of the argument's value is made and given to the function. Any changes made to the parameter within the function do not affect the original variable outside the function.

When you pass an argument by reference, you're actually passing a reference or memory address of the original variable to the function. This allows the function to modify the original variable directly.