Embedded Software Engineer Interview Questions And Answers Guide.

Global Guideline.

https://globalguideline.com/

Embedded Software Engineer Job Interview Preparation Guide.

Question # 1

Tell me what is the need for DMAC in ES?

Answer:-

Direct memory access is mainly used to overcome the disadvantages of interrupt and program controlled I/O.
DMA modules usually take the control over from the processor and perform the memory operations and this is mainly because to counteract the mismatch in the processing speeds of I/O units and the processor. This is comparatively faster.
It is an important part of any embedded systems,and the reason for their use is that they can be used for bursty data transfers instead of single byte approaches.
It has to wait for the systems resources such as the system bus in case it is already in control of it.

Read More Answers.

Question # 2

Tell me how to create a child process in linux?

Answer:-

* Prototype of the function used to create a child process is pid_t fork(void);
* Fork is the system call that is used to create a child process. It takes no arguments and returns a value of type pid_t.
* If the function succeeds it returns the pid of the child process created to its parent and child receives a zero value indicating its successful creation.
* On failure, a -1 will be returned in the parent's context, no child process will be created, and errno will be set.
* The child process normally performs all its operations in its parents context but each process independently of one nother and also inherits some of the important attributes from it such as UID, current directory, root directory and so on.

Read More Answers.

Question # 3

Tell me what are the commonly found errors in Embedded Systems?

Answer:-

* Damage of memory devices due to transient current and static discharges.
* Malfunctioning of address lines due to a short in the circuit.
* Malfunctioning of Data lines.
* Some memory locations being inaccessible in storage due to garbage or errors.
* Improper insertion of Memory devices into the memory slots.
* Faulty control signals.

Read More Answers.

Question # 4

Tell me what are the functional requirements that are used in the embedded systems?

Answer:-

Functional requirements specifies the discrete and the logic to provide the services, functionality, features and the implementation that is independent from the components that are getting used in it. These are used to represent the constraints that are in the form of physical and define the probability to specify the components discretely from each other. The functional requirements are given for the hardware as well that gives more performance and measures the physical resources that are present like clock frequency, latency, etc. Functional requirements allow the system and hardware machines to transfer the functions with the non-deterministic probability.

Read More Answers.

Question # 5

Explain me what is the main function of Multiplexed Address/Data Bus?

Answer:-

The memory bus is used to carry the address and the data from the processor to the memory so that it can be easily accessed by the devices. These buses carry the value of the data that has to be passed for the proper functioning. The use of the technique "Time division multiplexing" is used that allow the reading and writing of the data to be done from the same bus line. This requires lots of time to be given to the bus so that it can complete the read and write operation of the data in the memory. This is very expensive process due to the data transfer technique that is used in between the processor and the memory. This also gives the concept of cache and provides algorithms to solve the problems occurring in read and writes operations.

Read More Answers.

Question # 6

Tell me how does the interrupts handle by using the threads?

Answer:-

The interrupts that comes in between the input/output operations gets detected when the input/output devices are ready. The interrupt never gets handled directly rather, it sends the interrupt signal to the thread to the input/output device that is ready to allow the thread to take necessary actions. The thread uses the signaling concept that allows the initialization to be done using the semaphore that keeps the states updated and handle the interrupt in an easy way. The input/output device getting the request and it also passes the semaphore to handle. The input/output device takes the semaphore that is ready. The thread is having the minimum latency that uses the first level interrupt handler to handle the interrupts completely. It allows the priority of the thread to be set and it doesn't allow the context to be change as well.

Read More Answers.

Question # 7

Please write a program to show the functionality of Power-save super loop?

Answer:-

To check the loop time of the program the power-save super loop is used. If the average loop time of the program is 1ms, and it requires only few instructions to be checked every second the program will save the state and build a delay that will be caused to read the input on every loop and it saves lot of energy or the power that needs to be used. The function that is required to be performed to show the functionality is:

```
Main_Function()
Function
{
  Initialization();
  Do_Forever
  {
    Check_Status();
    Do_Calculations();
    Output_Response();
    Delay_For_Next_Loop();
  }
}
```

Read More Answers.

Question # 8

Tell me what is interrupt latency? How can you reduce it?

Answer:-

It is the time taken to return from the interrupt service routine post handling a specific interrupt. Interrupt latency can be reduced by writing minor ISR routines.

Read More Answers.

Question # 9

Do you know what is the use of volatile keyword?

Answer:-

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 memory-mapped peripheral registers, global variables modified by an interrupt service routine, global variables accessed by multiple tasks within a multi-threaded application.

Read More Answers.

Question # 10

Explain how to reduce interrupt latency?

Answer:-

Interrupt latency can be minimized by writing short ISR routine and by not delaying interrupts for more time.

Read More Answers.

Question # 11

Tell me what are inline functions?

Answer:-

The ARM compilers support inline functions with the keyword __inline. These functions have a small definition and the function body is substituted in each call to the inline function. The argument passing and stack maintenance is skipped and it results in faster code execution, but it increases code size, particularly if the inline function is large or one inline function is used often.

Read More Answers.

Question # 12

Tell me what is embedded system in a computer system?

Answer:-

An embedded system is a computer system that is part of a larger system or machine. It is a system with a dedicated function within a larger electrical or mechanical system.

Read More Answers.

Question # 13

Tell me what is microcontroller?

Answer:-

The microcontroller is a self-contained system with peripherals, memory and a processor that can be used as embedded system.

Read More Answers.

Question # 14

Explain various uses of timers in embedded system?

Answer:-

Timers in embedded system are used in multiple ways
* Real Time Clock (RTC) for the system
* Initiating an event after a preset time delay
* Initiating an even after a comparison of preset times
* Capturing the count value in timer on an event
* Between two events finding the time interval
* Time slicing for various tasks
* Time division multiplexing
* Scheduling of various tasks in RTOS

Read More Answers.

Question # 15

Tell me whether we can use semaphore or mutex or spinlock in interrupt context in Linux Kernel?

Answer:-

Semaphore or Mutex cannot be used for interrupt context in Linux Kernel. While spinlocks can be used for locking in interrupt context.

Read More Answers.

Question # 16

Do you know what is priority inversion?

Answer:-

If two tasks share a resource, the one with higher priority will run first. However, if the lower-priority task is using the shared resource when the higher-priority task becomes ready, then the higher-priority task must wait for the lower-priority task to finish. In this scenario, even though the task has higher priority it needs to wait for the completion of the lower-priority task with the shared resource. This is called priority inversion.

Read More Answers.

Question # 17

Tell me which parameters decide the size of data type for a processor?

Answer:-

Actually, compiler is the one responsible for size of the data type. But it is true as long as OS allows that. If it is not allowable by OS, OS can force the size.

Read More Answers.

Question # 18

Explain me what is spin lock?

Answer:-

If a resource is locked, a thread that wants to access that resource may repetitively check whether the resource is available. During that time, the thread may loop and check the resource without doing any useful work. Suck a lock is termed as spin lock.

Read More Answers.

Question # 19

Tell me what type of scheduling is there in RTOS?

Answer:-

RTOS uses pre-emptive scheduling. In pre-emptive scheduling, the higher priority task can interrupt a running process and the interrupted process will be resumed later.

Read More Answers.

Question # 20

Explain me what is semaphore?

Answer:-

A semaphore is an abstract datatype or variable that is used for controlling access, by multiple processes to a common resource in a concurrent system such as multiprogramming operating system. Semaphores are commonly used for two purposes
* To share a common memory space
* To share access to files

Read More Answers.

Question # 21

Please explain what is interrupt latency? How can you reduce it?

Answer:-

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.

Read More Answers.

Question # 22

Tell me can static variables be declared in a header file?

Answer:-

A static variable cannot be declared without defining it. A static variable can be defined in the header file. But doing so, the result will be having a private copy of that variable in each source file which includes the header file. So it will be wise not to declare a static variable in header file, unless you are dealing with a different scenario.

Read More Answers.

Question # 23

Tell me what is NULL pointer and what is its use?

Answer:-

The NULL is a macro defined in C. Null pointer actually means a pointer that does not point to any valid location. We define a pointer to be null when we want to make sure that the pointer does not point to any valid location and not to use that pointer to change anything. If we don't use null pointer, then we can't verify whether this pointer points to any valid location or not.

Read More Answers.

Question # 24

Tell me what are the different storage classes in C?

Answer:-

* Auto
* Register
* Static
* Extern or Global

Read More Answers.

Question # 25

Tell me can structures be passed to the functions by value?

Answer:-

Yes, it can be passed. However, it has several disadvantages.

Read More Answers.

Question # 26

Tell me what is the function of DMA controlled in embedded system?

Answer:-

DMA stands for Direct Memory Access controller that handles the allocation of the memory dynamically to the components and allows the data to be transferred between the devices. It is used for the communication between different input/output devices. It automatically detects the devices that are present for the transfer of data between the input/output devices. The interrupt can be used to complete the data transfer between the devices. It is used to give the high quality performance as, the input/output device can perform the operations that are in parallel with the code that are in execution phase. It can't be used in all the systems. It consists of 8-bit microcontrollers that are used to control the overall system in execution.

Read More Answers.

Question # 27

Tell me how does Input/Output bus functions?

Answer:-

Input and output devices or functions allow the user to interact with the external files. Input and output functions are used to transfer the load on the bus. It uses the multiplexes having the input and output signals that remain same. Input and output buses move at the slower rate or speed than the processor speed. This increases the problem of bottleneck or the deadlock due to poor performance. There is a possibility to send more transistors for a layout to be given. These different devices may have very different speeds of communication. When programming IO bus control, make sure to take this into account.
In some systems, memory mapped IO is used. In this scheme, the hardware reads its IO from predefined memory addresses instead of over a special bus. This means you'll have simpler software, but it also means main memory will get more access requests.

Read More Answers.

Question # 28

Explain me why is Model transformations used in the embedded system?

Answer:-

Model transformations involve multiple models that are used to define different views of a system. It provides different level of granularity that it doesn't use either the top-down approach or the bottom-up approach to implement the basic functionality of the system. It is used to integrate the library components used that involves the iteration of the model that needs to be constructed. It also involves the analysis of the model so that the process can be made automated by using the construction tools. The compilation made the progress by improving the code that is written in high level language and the code generator produce the code that is required for the machine language.

Read More Answers.

Question # 29

Tell me why is java mostly used in embedded systems?

Answer:-

* Java was mainly designed and conceptualised for code that can work on different platforms without any hassles and also for being secure enough so as to not harm or corrupt other modules of code.
* Features like exception handling, simple syntax and Automatic Garbage collection all work in its favour as the language for use in ES's.
* Also that it is widely used in the form of Java applets makes it very popular confining it to the limits of JVM. It is Dynamic in nature.
* Its use is also being exploited in enterprise systems in the form of J2EE, J2SE, J2ME in case of mobile applications.

Read More Answers.

Question # 30

Tell me what could be the reasons for a System to have gone blank and how would you Debug it?

Answer:-

Possible reasons could be:
* PC being overheated.
* Dust having being accumulated all around.
* CPU fans not working properly .
* Faulty power connections.
* Faulty circuit board from where the power is being drawn.
* Support Drivers not having being installed.
Debugging steps which can be taken are:
* Cleaning the system thoroughly and maintaining it in a dust-free environment. Environment that is cool enough and facilitates for easy passage of air should be ideal enough.
* By locating the appropriate support drivers for the system in consideration and having them installed.

Read More Answers.

Question # 31

Tell me what is the need for an infinite loop in Embedded systems?

Answer:-

Infinite Loops are those program constructs where in there is no break statement so as to get out of the loop, it just keeps looping over the statements within the block defined.
Example:

```
While(Boolean True) OR for(;;);
{
//Code
}
```

Embedded systems need infinite loops for repeatedly processing/monitoring the state of the program. One example could be the case of a program state continuously being checked for any exceptional errors that might just occur during run time such as memory outage or divide by zero etc.,
For e.g. Customer care Telephone systems where in a pre-recorded audio file is played in case the dialer is put on hold..
- Also circuits being responsible for indicating that a particular component is active/alive during its operation by means of LED's.

Read More Answers.

Question # 32

Tell me how are macros different from inline functions?

Answer:-

Macros are normally used whenever a set of instructions/tasks have to be repeatedly performed. They are small programs to carryout some predefined actions.
We normally use the #define directive in case we need to define the values of some constants so in case a change is needed only the value can be changed and is reflected throughout.

```
#define mul(a,b) (a*b)
```

The major disadvantage of macros is that they are not really functions and the usual error checking and stepping through of the code does not occur.
Inline functions are expanded whenever it is invoked rather than the control going to the place where the function is defined and avoids all the activities such as saving the return address when a jump is performed. Saves time in case of short codes.

```
inline float add(float a,float b)
{
   return a+b;
}
```

Inline is just a request to the compiler and it is upto to the compiler whether to substitute the code at the place of invocation or perform a jump based on its performance algorithms.

Read More Answers.

Question # 33

Tell me what is the need for having multibyte data input and output buffers in case of device ports?

Answer:-

It's normally the case that some devices transfer the output either in a bursty or a sequential manner and also during input entry. If we take the example of keyboards, all the data entered is stored in a buffer and given at a time or one character at a time.
In case of networking there may be several requests to access the same resource and all these are queued in a buffer and serviced in the order they are received. Hence to avoid the input/output units from getting overloaded with requests, we use multibyte buffers.

Read More Answers.

Question # 34

Explain me what are the different types of system involved in embedded system?

Answer:-

Embedded systems are used to give the response in real time. So, it consists of the real time systems that allow the correct information to be passed to get the correct responses. For example, it includes of the flight control system that produce the responses in real time and it always the take the values also in real time. If any delay been caused by the system then it deals in the fatal error. The real time system includes the system that provides the response on time with the small delay. Real time systems include of many other system such as:
Hard Real-Time Systems - These are the systems that server the purpose of having constraints that are hard and it totally depends on the time to provide the response on time.
Soft Real-Time Systems - These systems serves the purpose of having few delays in giving up the responses that can tolerate small variations.
Hybrid Real-Time Systems - These systems includes the properties from both the systems and increases the performance.

Read More Answers.

Question # 35

Tell me what are the different types of customizations that is used with the "volatile" keyword?

Answer:-

Volatile keyword is used to show that the value can be changed anytime in the program. It is used for the compiler purpose and for the customization that works with the normal variables that are stored in the memory. There are three types of optimizations associated with the "volatile" keyword:
"Read" optimizations: allow the variable to be read once and put it in the register. If it is done then there is no re-reading of the variable during each and every time the program is compiled. The value can be used from the cache that is present in the register.
"Write" optimizations: allow the variable to be written such that the last write of the variable will be considered and it will be processed on. This takes the normal values that are stored in the memory.
Instruction reordering: allow to reorder the instructions that are used by the compiler and if any modification are required after being written once. The registers are used to perform the task and keep everything together.

Read More Answers.

Question # 36

Tell me what are the 4 types of inheritance relationship?

Answer:-

4 different types of inheritance relationship are Single, Multiple, Multilevel and Hybrid.

Read More Answers.

Question # 37

Tell me can a variable be both const and volatile?

Answer:-

The const keyword make sure that the value of the variable declared as const can't be changed. This statement holds true in the scope of the program. The value can still be changed by outside intervention. So, the use of const with volatile keyword makes perfect sense.

Read More Answers.

Question # 38

Tell me can we put breakpoint inside ISR?

Answer:-

Putting a break point inside ISR is not a good idea because debugging will take some time and a difference of half or more second will lead to different behavior of hardware. To debug ISR, definitive logs are better.

Read More Answers.

Question # 39

Tell me what are the uses of the keyword volatile?

Answer:-

Volatile keyword is used to prevent compiler to optimize a variable which can change unexpectedly beyond compiler's comprehension. Suppose, we have a variable which may be changed from scope out of the program, say by a signal, we do not want the compiler to optimize it. Rather than optimizing that variable, we want the compiler to load the variable every time it is encountered. If we declare a variable volatile, compiler will not cache it in its register.

Read More Answers.

Question # 40

Explain me what is the difference between microprocessor and microcontroller?

Answer:-

Microprocessor is managers of the resources (I/O, memory) which lie outside of its architecture
Microcontroller have I/O, memory, etc. built into it and specifically designed for control

Read More Answers.

Question # 41

Explain difference between RISC and CISC processor?

Answer:-

RISC (Reduced Instruction Set Computer) could carry out a few sets of simple instructions simultaneously. Fewer transistors are used to manufacture RISC, which makes RISC cheaper. RISC has uniform instruction set and those instructions are also fewer in number. Due to the less number of instructions as well as instructions being simple, the RISC computers are faster. RISC emphasise on software rather than hardware. RISC can execute instructions in one machine cycle.
CISC (Complex Instruction Set Computer) is capable of executing multiple operations through a single instruction. CISC have rich and complex instruction set and more number of addressing modes. CISC emphasise on hardware rather that software, making it costlier than RISC. It has a small code size, high cycles per second and it is slower compared to RISC.

Read More Answers.

Question # 42

Please explain what is semaphore?

Answer:-

Semaphore is actually a variable or abstract data type which controls access to a common resource by multiple processes. Semaphores are of two types -
* Binary semaphore - It can have only two values (0 and 1). The semaphore value is set to 1 by the process in charge, when the resource is available.
* Counting semaphore - It can have value greater than one. It is used to control access to a pool of resources.

Read More Answers.

Question # 43

Tell me what happens when recursive functions are declared inline?

Answer:-

Inlining an recursive function reduces the overhead of saving context on stack. But, inline is merely a suggestion to the compiler and it does not guarantee that a function will be inlined. Obviously, the compiler won't be able to inline a recursive function infinitely. It may not inline it at all or it may inline it, just a few levels deep.

Read More Answers.

Question # 44

Tell me what is difference between binary semaphore and mutex?

Answer:-

The differences between binary semaphore and mutex are as follows -
* Mutual exclusion and synchronization can be used by binary semaphore while mutex is used only for mutual exclusion.
* A mutex can be released by the same thread which acquired it. Semaphore values can be changed by other thread also.
* From an ISR, a mutex can not be used.
* The advantage of semaphores is that, they can be used to synchronize two unrelated processes trying to access the same resource.
* Semaphores can act as mutex, but the opposite is not possible.

Read More Answers.

Question # 45

Tell me what is RTOS?

Answer:-

In an operating system, there is a module called the scheduler, which schedules different tasks and determines when a process will execute on the processor. This way, the multi-tasking is achieved. The scheduler in a Real Time Operating System (RTOS) is designed to provide a predictable execution pattern. In an embedded system, a certain event must be entertained in strictly defined time.
To meet real time requirements, the behaviour of the scheduler must be predictable. This type of OS which have a scheduler with predictable execution pattern is called Real Time OS(RTOS). The features of an RTOS are
* Context switching latency should be short.
* Interrupt latency should be short.
* Interrupt dispatch latency should be short.
* Reliable and time bound inter process mechanisms.
* Should support kernel preemption.

Read More Answers.

Question # 46

Tell me what is a Watchdog Timer?

Answer:-

A watchdog timer is an electronic device or electronic card that execute specific operation after certain time period if something goes wrong with an electronic system.

Read More Answers.

Question # 47

Explain what are the essential components of embedded system?

Answer:-

Essential components of embedded system includes:
* Hardware
* Processor
* Memory
* Timers
* I/O circuits
* System application specific circuits
* Software
* It ensures the availability of System Memory
* It checks the Processor Speed availability
* The need to limit power lost when running the system continuously
* Real Time Operating System
* It runs a process as per scheduling and do the switching from one process to another

Read More Answers.

Question # 48

Tell me can include files be nested?

Answer:-

Yes. Include files can be nested any number of times. But you have to make sure that you are not including the same file twice. There is no limit to how many header files that can be included. But the number can be compiler dependent, since including multiple header files may cause your computer to run out of stack memory.

Read More Answers.

Question # 49

Tell me what is ISR?

Answer:-

An ISR(Interrupt Service Routine) is an interrupt handler, a callback subroutine which is called when a interrupt is encountered.

Read More Answers.

Question # 50

Tell me why does pre-emptive multi-threading used to solve the central controller problem?

Answer:-

Multi-threading provide lot of functionality to the system to allow more than one task can be run at a time. It allows a process to execute faster with less difficulty. But, if there any problem comes in any program or the process than the entire system comes to a halt and slows down the whole system. To control the behavior of this the preemptive multi-threading is used. The control in this case is being shifted from one process to another at any time according to the requirement provided. It allows the program to give the control to another program that is having the higher priority. It includes of many problems like giving of a control by a process half way through in execution and the preemption of the process takes place then the data will be entered as corrupted in the memory location, multi-threading keeps the synchronization that is to be performed between different components of the system and the program and try to avoid the problem mentioned above.

Read More Answers.

Question # 51

Tell me what are the different types of Buses used by the embedded systems?

Answer:-

The buses are used to pass the messages between different components of the system. There are buses existing as:
Memory Bus: it is related to the processor that is connected to the memory (RAM) using the data bus. This bus includes the collection of wires that are in series and runs parallel to each other to send the data from memory to the processor and vice versa.
Multiplexed Address/Data Bus: Multiplex data bus consists of the bus that can read and write in the memory but it decreases the performance due to the time consumed in reading and writing of the data in the memory.
De-multiplexed Bus: these consists of two wires in the same bus, where one wire consists of the address that need to be passed and the other one consists of the data that need to be passed from one to another. This is a faster method compared to other.
Input/Output bus: it uses the multiplexing techniques to multiplex the same bus input and output signals. This creates the problem of having the deadlock due to slow processing of it.

Read More Answers.

Question # 52

Can you please explain the difference between mutexes vs semaphores?

Answer:-

Semaphores is a synchronization tool to overcome the critical section problem.
A semaphore S is basically an integer variable that apart from initialization is accesses only through atomic operations such as wait() and signal().
Semaphore object basically acts as a counter to monitor the number of threads accessing a resource.
Mutex is also a tool that is used to provide deadlock free mutual exclusion. It protects access to every critical data item. If the data is locked and is in use, it either waits for the thread to finish or awakened to release the lock from its inactive state.

Read More Answers.

Question # 53

What is interrupt latency and how can we decrease it?

Answer:-

1. Interrupt latency basically refers to the time span an interrupt is generated and it being serviced by an appropriate routine defined, usually the interrupt handler.
2. External signals, some condition in the program or by the occurrence of some event, these could be the reasons for generation of an interrupt.
3. Interrupts can also be masked so as to ignore them even if an event occurs for which a routine has to be executed.

4. Following steps could be followed to reduce the latency
* ISRs being simple and short.
* Interrupts being serviced immediately
* Avoiding those instructions that increase the latency period.
* Also by prioritizing interrupts over threads.
* Avoiding use of inappropriate APIs.

Read More Answers.

Question # 54

Explain how does combination of functions reduce memory requirements in embedded systems?

Answer:-

The amount of code that has to be dealt with is reduced thus easing the overhead and redundancy is eliminated in case if there is anything common among the functions.
Memory allocation is another aspect that is optimized and it also makes sense to group a set of functions related in some way as one single unit rather than having them to be dispersed in the whole program.
In case of interactive systems display of menu list and reading in the choices of user's could be encapsulated as a single unit.

Read More Answers.

Question # 55

As you know a vast majority of High Performance Embedded systems today use RISC architecture why?

Answer:-

According to the instruction sets used, computers are normally classified into RISC and CISC. RISC stands for 'Reduced Instruction Set Computing' . The design philosophy of RISC architecture is such that only one instruction is performed on each machine cycle thus taking very less time and speeding up when compared to their CISC counterparts.
Here the use of registers is optimised as most of the memory access operations are limited to store and load operations.
Fewer and simple addressing modes, and simple instruction formats leads to greater efficiency, optimisation of compilers, re-organisation of code for better throughput in terms of space and time complexities. All these features make it the choice of architecture in majority of the Embedded systems.
CISC again have their own advantages and they are preferred whenever the performance and compiler simplification are the issues to be taken care of.

Read More Answers.

Question # 56

Basic Embedded Software Engineer Job Interview Questions:

Answer:-

* What is something substantive that you've done to improve as a developer in your career?
* Would you call yourself a craftsman (craftsperson) and what does that word mean to you?
* Implement a <basic data structure> using <some language> on <paper|whiteboard|notepad>.
* What is SOLID?
* Why is the Single Responsibility Principle important?
* What is Inversion of Control? How does that relate to dependency injection?
* How does a 3 tier application differ from a 2 tier one?
* Why are interfaces important?
* What is the Repository pattern? The Factory Pattern? Why are patterns important?
* What are some examples of anti-patterns?
* Who are the Gang of Four? Why should you care?
* How do the MVP, MVC, and MVVM patterns relate? When are they appropriate?
* Explain the concept of Separation of Concerns and it's pros and cons.
* Name three primary attributes of object-oriented design. Describe what they mean and why they're important.
* Describe a pattern that is NOT the Factory Pattern? How is it used and when?
* You have just been put in charge of a legacy code project with maintainability problems. What kind of things would you look to improve to get the project on a stable footing?
* Show me a portfolio of all the applications you worked on, and tell me how you contributed to design them.
* What are some alternate ways to store data other than a relational database? Why would you do that, and what are the trade-offs?
* Explain the concept of convention over configuration, and talk about an example of convention over configuration you have seen in the wild.
* Explain the differences between stateless and stateful systems, and impacts of state on parallelism.
* Discuss the differences between Mocks and Stubs/Fakes and where you might use them (answers aren't that important here, just the discussion that would ensue).
* Discuss the concept of YAGNI and explain something you did recently that adhered to this practice.
* Explain what is meant by a sandbox, why you would use one, and identify examples of sandboxes in the wild.

Read More Answers.

Question # 57

Role-specific Embedded Software Engineer Interview Questions:

Answer:-

* How can you reduce memory requirements in embedded systems?
* If a system goes blank, how do you debug it?
* Why would you choose Java in embedded systems?
* What software configuration management tools are you familiar working with?
* Can you name any code testing tools?
* Give an example of troubleshooting on embedded targets. Explain the process step-by-step.
* What's your experience with technical documentation?
* How can you ensure that debugging a program while it's being used will not affect its functionality?
* Are you familiar with design patterns? What design software have you used and in what situations?
* How do you identify hardware errors?
* How can you optimize the I/O performance?

* What's your experience with QA engineering?

Read More Answers.

Question # 58

Computer Science Job Interview Questions:

Answer:-

* What's RISC architecture?
* What are the differences between analytical and computational modeling?
* Why is it better to use multi-threading polling instead of a single threading model?
* Can you name the differences between object-oriented and component-based design?

Read More Answers.

Question # 59

Profesional Embedded Software Engineer Interview Questions:

Answer:-

* What Source Control systems have you worked with?
* What is Continuous Integration? Have you used it and why is it important?
* Describe a software development life cycle that you've managed.
* How do you react to people criticizing your code/documents?
* Whose blogs or podcasts do you follow? Do you blog or podcast?
* Tell me about some of your hobby projects that you've written in your off time.
* What is the last programming book you read?
* Describe, in as much detail as you think is relevant, as deeply as you can, what happens when I type "cnn.com" into a browser and press "Go".
* Describe the structure and contents of a design document, or a set of design documents, for a multi-tiered web application.
* What's so great about <cool web technology of the day>?
* How can you stop your DBA from making off with a list of your users' passwords?
* What do you do when you get stuck with a problem you can't solve?
* If your database was under a lot of strain, what are the first few things you might consider to speed it up?
* What is SQL injection?
* What's the difference between unit test and integration test?
* Tell me about 3 times you failed.
* What is Refactoring ? Have you used it and it is important? Name three common refactorings.
* You have two computers, and you want to get data from one to the other. How could you do it?
* Left to your own devices, what would you create?
* Given Time, Cost, Client satisfaction and Best Practices, how will you prioritize them for a project you are working on? Explain why.
* What's the difference between a web server, web farm and web garden? How would your web application need to change for each?
* What value do daily builds, automated testing, and peer reviews add to a project? What disadvantages are there?
* What elements of OO design are most prone to abuse? How would you mitigate that?
* When do you know your code is ready for production?
* What's YAGNI? Is this list of questions an example?
* Describe to me some bad code you've read or inherited lately.

Read More Answers.

Question # 60

Embedded Software Engineer Job Interview Questions:

Answer:-

* Describe the life-cycle of a software development (application design) process.
* What software languages have you used to design, develop, and debug software application interfaces?
* Explain the software analysis process, such as code analysis, code metrics, and system risk analysis or software reliability.
* What type of software simulator and developer have you used?
* If you've integrated new software in an existing system, tell us how you evaluated and identified the requirements for this new technology.
* Have you designed UI, user interface?
* Have you developed software testing procedures for new systems and performed QA, quality assurance, or audits?

Read More Answers.

Question # 61

Behavioral Embedded Software Engineer Interview Questions:

Answer:-

* What are the most common errors you've found in embedded systems and how have you resolved them?
* How do you learn about the latest technology trends?
* Describe a time when you collaborated with designers and engineers to deploy a new product. What was your role?

Read More Answers.

Question # 62

General Embedded Software Engineer Job Interview Questions:

Answer:-

* What's the difference between Locking and Lockless (Optimistic and Pessimistic) concurrency models?
* What kinds of problems can you hit with locking model? And a lockless model?
* What trade offs do you have for resource contention?
* How might a task-based model differ from a threaded model?
* What's the difference between asynchrony and concurrency?

* Are you still writing code? Do you love it?
* You've just been assigned to a project in a new technology how would you get started?
* How does the addition of Service Orientation change systems? When is it appropriate to use?
* What do you do to stay abreast of the latest technologies and tools?
* What is the difference between "set" logic, and "procedural" logic. When would you use each one and why?

Read More Answers.

Question # 63

Tell me when one must use recursion function? Mention what happens when recursion functions are declared inline?

Answer:-

Recursion function can be used when you are aware of the number of recursive calls is not excessive. Inline functions property says whenever it will called, it will copy the complete definition of that function. Recursive function declared as inline creates the burden on the compilers execution.

Read More Answers.

Question # 64

Tell me what is the difference between hard real-time and soft real-time OS?

Answer:-

A Hard real-time system strictly adheres to the deadline associated with the task. If the system fails to meet the deadline, even once, the system is considered to have failed. In case of a soft real-time system, missing a deadline is acceptable. In this type of system, a critical real-time task gets priority over other tasks and retains that priority until it completes.

Read More Answers.

Question # 65

Explain what is size of character, integer, integer pointer, character pointer?

Answer:-

* The sizeof character is 1 byte.
* Size of integer is 4 bytes.
* Size of integer pointer and character is 8 bytes on 64 bit machine and 4 bytes on 32 bit machine.

Read More Answers.

Question # 66

Do you know what is the function of Watchdog timer in embedded system?

Answer:-

The embedded system should have a function that can allow the fixing the system if anything goes wrong with it. Watchdog timer is a tool that is used in embedded system and having a long-fuse that runs several seconds. Watchdog timer includes the automated timing control that count down the number from max to 0 and when the counter reaches the zero, this WDT reset the micro-controller that gets turned off when the timer was in initial phases. Watchdog require the user to put some value before it runs out of time and reset the whole process as this can harm the data and the system. Resetting by WDT can be done when the process is half complete. Watchdog timer have the counter that is used to watch the processes that are running and if there is any issue occurs then WDT itself times out. The resetting of the system will be done to always give it the best possible way to execute the process.

Read More Answers.

Question # 67

Tell me why is it better to use multi-threading polling then single threading model?

Answer:-

Multi-threading allows a simple thread to be stored and polled. There is no Input/output function that is applied when it is having the poll. When there is no poll available to spawn it makes the system to sleep for an amount of time till the request for another poll reaches. If there is one process that is running then it divides that process into multiple threads and processes it accordingly. It allows the main thread to process all the request and produce the output by combining all other. Multi-threading allows the main thread not to put off the result or the output that will be generated. It also allow the priority of the thread to be changed by allowing to set the priority of the input/output process. It also has some problems with the polling interval that can make a thread starve for some time if the request isn't handled properly.

Read More Answers.

Question # 68

Explain what are the differences between analytical and computational modeling?

Answer:-

Analytical model allows the components to deal with the concurrency that are given during the process and the quantitative constraints that might come in between the components. Whereas, computational model deal with the non-deterministic abstraction hierarchy that has computational complexity to deal with the concurrency and allow it put also the physical constraints.
Analytical models can't deal with the partial and incremental specifications that are non-deterministic. It is also not good in controlling the computation complexity that is used in the hardware design. Whereas, Computational model can, deal with constraints easily and it provides an upgradeable solution.
Analytical model is the equation based model that doesn't have the time-sharing and parallelism concepts. Whereas, time-sharing and parallelism is used, in the abstract method that provides the theories of complexity and the real time evaluation.

Read More Answers.

Question # 69

Explain me what is Endianness of a system and how do different systems communicate with each other?

Answer:-

Endianness basically refers to the ordering of the bytes within words or larger bytes of data treated as a single entity.
When we consider a several bytes of data say for instance 4 bytes of data,XYZQ the lower byte if stored in a Higher address and others in successively decreasing addresses, then it refers to the Big Endian and the vice versa of this refers to Little Endian architecture.
Intel 80x86 usually follows Little Endian and others like IBM systems follow Big Endian formats.
If the data is being transmitted care has to be taken so as to know as to which byte,whether the higher or the lower byte is being transmitted.
Hence a common format prior to communication has to be agreed upon to avoid wrong interpretation/calculations.
Usually layer modules are written so as to automate these conversion in Operating systems.

Read More Answers.

Question # 70

Explain me how does the interrupt architecture works?

Answer:-

Interrupt architecture allows the use of interrupt by the processor whenever an Input/output is ready for the processing. The processor in this case calls a special function to handle the request that comes and leave all the work that is getting performed at that time. The special function that is known as interrupt handler or the interrupt service routine consists of all the input, and output queries, or the interrupts handled by it. It is an efficient and simple way to handle the interrupts. It uses only one function to deal with the interrupts. There are properties of starvation that can creep in when handling the input/output requests. The data can be lost if the interrupt doesn't get handled before the time runs out. This is a technique that is use to deal with the short processes that involve input and output.

Read More Answers.

Question # 71

Tell me what is the difference between Hardware design and Software Design?

Answer:-

Hardware design is designed with the collaboration of interconnected parallel components that inherits the properties of each other. Whereas, Software design is designed with sequential components, that are based on objects and threads.
Hardware design structure doesn't change dynamically and it can't be created, modified or removed easily. Whereas, Software design structure can be changed dynamically and re-usability features, used to define the components. It also includes easy creation, modification and removal of the components from the software.
Hardware design focuses on individual components that are represented using analytical model that uses the transfer functions. Whereas, Software design represent the components using computation model that can have abstract execution engine or it can use the virtual machine that are non-deterministic.

Read More Answers.

Question # 72

Tell me is Count Down_to_Zero Loop better than Count_Up_Loops?

Answer:-

Count down to zero loops are better. Reason behind this is that at loop termination, comparison to zero can be optimized by the compiler. Most processors have instruction for comparing to zero. So they don't need to load the loop variable and the maximum value, subtract them and then compare to zero. That is why count down to zero loop is better.

Read More Answers.

Question # 73

Tell me what is the purpose of using critical sections?

Answer:-

Critical section allows the process to run in an area that is defined by that process only. It is a sequence of instructions that can be corrupted if any other process tries to interrupt it. This process allow the operating system to give the synchronization objects that are used to monitor the processes that are up and running so that no other process will get executed till the critical region consists of a process that is already running. The example includes removal of the data from a queue running in a critical section and if not protected then it can be interrupted and the data have chances of getting corrupted. The processes exit from the critical section as soon as they finish the execution so that the chances can be given to other processes that are waiting for their chance to come.

Read More Answers.

Question # 74

Tell me what is interaction semantics used in embedded systems?

Answer:-

Interaction semantics allow the actions to be performed by the system components to allow it to get the global behavior. The interaction can be atomic or non-atomic dependent on the interaction between the components. These components can't be modified using the interference having the other interactions. Languages that are used, having buffered communication, and other languages, that include multi-threaded languages that use non-atomic interactions. There are two types of interactions that are used:
* Strong synchronization: allow the components to participate together and have strong bonding in between.
* Weakly synchronizing: are asymmetric that required the communication from both the objects.

Read More Answers.

Question # 75

Tell me what is Top half & bottom half of a kernel?

Answer:-

Sometimes to handle an interrupt, a substantial amount of work has to be done. But it conflicts with the speed need for an interrupt handler. To handle this situation, Linux splits the handler into two parts - Top half and Bottom half. The top half is the routine that actually responds to the interrupt. The bottom half on the other hand is a routine that is scheduled by the upper half to be executed later at a safer time.
All interrupts are enabled during execution of the bottom half. The top half saves the device data into the specific buffer, schedules bottom half and exits. The bottom half does the rest. This way the top half can service a new interrupt while the bottom half is working on the previous.

Read More Answers.

Question # 76

Explain me what are the rules followed by Mutexes?

Answer:-

Mutex is also called as Mutual Exclusion is a mechanism that is used to show the preemptive environment and allow providing security methods like preventing an unauthorized access to the resources that are getting used in the system. There are several rules that has to be followed to ensure the security policies:
Mutex are directly managed by the system kernel that provides a secure environment to allow only the applications that passes the security rules and regulations. The mutex consists of objects that are allowed to be called by the kernel.
Mutex can have only one process at a time in its area that is owned by the process using it. This allows less conflict between the different applications or processes that wait for their turn to execute it in the kernel area.
Mutex can be allocated to another mutex that is running some task at a particular time and allow the kernel to have synchronization in between them.
If Mutex is allocated to some other process then the area will consist of the process till the area is having the process in it.

Read More Answers.

Question # 77

Tell me significance of watchdog timer in Embedded Systems?

Answer:-

Watchdog timer is basically a timing device that is set for predefined time interval and some event should occur during that time interval else the device generates a time out signal.
One application where it is most widely used is when the mobile phone hangs and no activity takes place, in those cases watchdog timer performs a restart of the system and comes to the rescue of the users.
It is used to reset to the original state whenever some inappropriate events take place such as too many commands being given at the same time or other activities that result in malfunctioning of the GUI. It is usually operated by counter devices.

Read More Answers.

Question # 78

Tell me what are buses used for communication in embedded system?

Answer:-

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.
While ISA, EISA, PCI are standard buses for parallel communication used in PCs, computer network devices, etc.

Read More Answers.

Question # 79

Do you know what is virtual memory?

Answer:-

Virtual memory is a technique that allows processes to allocate memory in case of physical memory shortage using automatic storage allocation upon a request. The advantage of the virtual memory is that the program can have a larger memory than the physical memory. It allows large virtual memory to be provided when only a smaller physical memory is available. Virtual memory can be implemented using paging.
A paging system is quite similar to a paging system with swapping. When we want to execute a process, we swap it into memory. Here we use a lazy swapper called pager rather than swapping the entire process into memory. When a process is to be swapped in, the pager guesses which pages will be used based on some algorithm, before the process is swapped out again. Instead of swapping whole process, the pager brings only the necessary pages into memory. By that way, it avoids reading in unnecessary memory pages, decreasing the swap time and the amount of physical memory.

Read More Answers.

Question # 80

Tell me why do we need virtual device drivers when we have physical device drivers?

Answer:-

Device drivers are basically a set of modules/routines so as to handle a device for which a direct way of communication is not possible through the user's application program and these can be thought of as an interface thus keeping the system small providing for minimalistic of additions of code, if any.
Physical device drivers can't perform all the logical operations needed in a system in cases like IPC, Signals and so on...
The main reason for having virtual device drivers is to mimic the behaviour of certain hardware devices without it actually being present and these could be attributed to the high cost of the devices or the unavailability of such devices.
These basically create an illusion for the users as if they are using the actual hardware and enable them to carryout their simulation results.
Examples could be the use of virtual drivers in case of Network simulators,also the support of virtual device drivers in case a user runs an additional OS in a virtual box kind of a software.

Read More Answers.

Question # 81

Explain what are the uses of the keyword static?

Answer:-

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.

Read More Answers.

Question # 82

Please explain can structures be passed to the functions by value?

Answer:-

Passing structure by its value to a function is possible, but not a good programming practice. First of all, if we pass the structure by value and the function changes some of those values, then the value change is not reflected in caller function. Also, if the structure is big, then passing the structure by value means copying the whole structure to the function argument stack which can slow the program by a significant amount.

Read More Answers.

Question # 83

Tell me what is the function of simple thread poll in embedded system?

Answer:-

Simple thread poll allow the ready output to be passed for checking by giving it to the bus that is free and then the output is sent along the thread. The bus can send the output depending on the time that has been given and during the transfer the user won't be able to perform any other operation. The input is given after finding out the bus is free or not and if it free then a check is made to see that the input exists or not. This thread poll is easy to understand but it is not efficient method to allow the data to be put over the bus manually. The problem of not doing multi-tasking can occur due to doing one task at a time. The method is only be used when input/output occurs at interval that are infrequent.

Read More Answers.

Question # 84

Do you know what is priority inheritance?

Answer:-

Priority inheritance is a solution to the priority inversion problem. The process waiting for any resource which has a resource lock will have the maximum priority. This is priority inheritance. When one or more high priority jobs are blocked by a job, the original priority assignment is ignored and execution of critical section will be assigned to the job with the highest priority in this elevated scenario. The job returns to the original priority level soon after executing the critical section.

Read More Answers.

Question # 85

Tell me why cannot arrays be passed by values to functions?

Answer:-

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.

Read More Answers.

Question # 86

Tell us what is the need for an infinite loop in embedded systems?

Answer:-

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.

Read More Answers.

Question # 87

Tell me can a pointer be volatile?

Answer:-

If we see the declaration volatile int *p, it means that the pointer itself is not volatile and points to an integer that is volatile. This is to inform the compiler that pointer p is pointing to an integer and the value of that integer may change unexpectedly even if there is no code indicating so in the program.

Read More Answers.

Question # 88

Suppose you buy some RTOS, what are the features you look for in?

Answer:-

* Deterministic operating system having guaranteed worst-case interrupt latency and context-switch times.
* Documentation providing for the minimum, average, and maximum number of clock cycles required by each system call.
* Interrupt response times should be very minute.
* Context switch time should be very low.
* Compatibility with several plugin devices.
* Overall it should be very reliable.

Read More Answers.

Question # 89

Explain some of the commonly found errors in Embedded Systems?

Answer:-

Some of the commonly found errors in embedded systems are
* Damage of memory devices static discharges and transient current
* Address line malfunctioning due to a short in circuit
* Data lines malfunctioning
* Due to garbage or errors some memory locations being inaccessible in storage

* Inappropriate insertion of memory devices into the memory slots
* Wrong control signals

Read More Answers.

Question # 90

Explain me what is kernel paging?

Answer:-

Paging is a memory management scheme by which computers can store and retrieve data from the secondary memory storage when needed in to primary memory. In this scheme, the operating system retrieves data from secondary storage in same-size blocks called pages. The paging scheme allows the physical address space of a process to be non continuous. Paging allows OS to use secondary storage for data that does not fit entirely into physical memory.

Read More Answers.

Question # 91

Tell me can you have constant volatile variables?

Answer:-

Yes, we can have constant volatile variables

Read More Answers.

Question # 92

Tell me how many types of IPC mechanism you know?

Answer:-

Different types of IPC mechanism are -
* Pipes
* Named pipes or FIFO
* Semaphores
* Shared memory
* Message queue
* Socket

Read More Answers.

Question # 93

Tell me can we use any function inside ISR?

Answer:-

We can use function inside ISR as long as that function is not invoked from other portion of the code.

Read More Answers.

Question # 94

Tell me what are real-time embedded systems?

Answer:-

Real-time embedded systems are computer systems that monitor, respond or control an external environment. This environment is connected to the computer system through actuators, sensors, and other input-output interfaces.

Read More Answers.

Question # 95

Tell me what are the qualifiers in C?

Answer:-

Volatile- it is about instructing the compiler that a variable will change
Constant- it is about instructing the compiler that a variable will not change.

Read More Answers.

Question # 96

Tell me what does DMA address will deal with?

Answer:-

DMA address deals with physical addresses. It is a device which directly drives the data and address bus during data transfer. So, it is purely physical address.

Read More Answers.

Question # 97

Tell me what is interrupt latency?

Answer:-

Interrupt latency is the time required for an ISR responds to an interrupt.

Read More Answers.

Question # 98

Explain what is return type of ISR?

Answer:-

ISR does not return anything. An ISR returns nothing because there is no caller in the code to read the returned values.

Read More Answers.

Question # 99

Tell me why embedded system is useful?

Answer:-

With embedded system, it is possible to replace dozens or even more of hardware logic gates, input buffers, timing circuits, output drivers, etc. with a relatively cheap microprocessor.

Read More Answers.

Question # 100

Explain me can we use printf inside ISR?

Answer:-

Printf function in ISR is not supported because printf function is not reentrant, thread safe and uses dynamic memory allocation which takes a lot of time and can affect the speed of an ISR up to a great extent.

Read More Answers.

Engineering Most Popular Interview Topics.

1 : Civil Engineering Frequently Asked Interview Questions and Answers Guide.

2 : Mechanical Engineering Frequently Asked Interview Questions and Answers Guide.

3 : Electrical Engineering Frequently Asked Interview Questions and Answers Guide.

4 : Chemical Engineering Frequently Asked Interview Questions and Answers Guide.

5 : Automobile Engineering Frequently Asked Interview Questions and Answers Guide.

6 : Electronics Communications Frequently Asked Interview Questions and Answers Guide.

7 : Instrumentation Engineering Frequently Asked Interview Questions and Answers Guide.

8 : Marine Engineering Frequently Asked Interview Questions and Answers Guide.

9 : Industrial Engineering Frequently Asked Interview Questions and Answers Guide.

10 : RF Engineer Frequently Asked Interview Questions and Answers Guide.

About Global Guideline.

Global Guideline is a platform to develop your own skills with thousands of job interview questions and web tutorials for fresher's and experienced candidates. These interview questions and web tutorials will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts. Global Guideline invite you to unlock your potentials with thousands of **Interview Questions with Answers** and much more. Learn the most common technologies at Global Guideline. We will help you to explore the resources of the World Wide Web and develop your own skills from the basics to the advanced. Here you will learn anything quite easily and you will really enjoy while learning. Global Guideline will help you to become a professional and Expert, well prepared for the future.

* This PDF was generated from https://GlobalGuideline.com at **November 29th, 2023**

* If any answer or question is incorrect or inappropriate or you have correct answer or you found any problem in this document then don't hesitate feel free and e-mail us we will fix it.

You can follow us on FaceBook for latest Jobs, Updates and other interviews material.
www.facebook.com/InterviewQuestionsAnswers

Follow us on Twitter for latest Jobs and interview preparation guides
https://twitter.com/InterviewGuide

Best Of Luck.

Global Guideline Team
https://GlobalGuideline.com
Info@globalguideline.com