Embedded Systems Interview Questions

Following are some of the most frequently asked Embedded Systems interview questions in the interview, here are the answers for them.

How does a combination of functions reduce memory requirements in embedded systems?

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 training Hyderabad display of menu list and reading in the choices of user's could be encapsulated as a single unit.

How does, taking the address of local variable result in unoptimized code?

The most powerful optimization for compiler is register allocation. That is it operates the variable from register, than memory.

Generally local variables are allocated in registers. However if we take the address of a local variable, compiler will not allocate the variable to register.

What is a pure function in ARM terminology?

Pure functions are those which return a result which depends only on their arguments.

They can be thought of as mathematical functions: they always return the same result if the arguments are the same. To tell the compiler that a function is pure, use the special declaration keyword __pure.

```
__pure int square(int x)

{ return x * x;

}
```

Compiler does optimization for pure functions. For example, the values which are allocated to memory can be safely cached in registers, instead of being written to memory before a call and reloaded afterwards.

Is 8085 an embedded system?

It’s not an embedded system. Because it will be a part of an embedded system and it does not work on any software.

What is the role of segment register?

In the 8086 processor architecture, memory addresses are specified in two parts called the segment and the offset. Segment values are stored in the segment registers. There are four or more segment registers: Code Segment (CS) contains segment of the current instruction (IP is the offset), Stack segment (SS) contain stack of the segment (SP is the offset), DS is the segment used by default for most data operations; ES is an extra segment register.

What type of registers contains an INTEL CPU?

Special function registers like accumulator, program controller (PC), data pointer (DPTR), TMOD and TCON (timing registers), 3 register banks with r0 to r7, Bit addressable registers like B.

Differentiate microprocessor and micro controller?

As now you are fundamentally familiar of what is a microcontroller And micro processor,it is Easy to recognize the significant contrasts between a micro controller and microprocessor.

1. Key difference in both of them is presence of external peripheral, where microcontrollers have RAM, ROM, EEPROM embedded in it while we have to use external circuits in case of microprocessors.

2. As all the peripheral of microcontroller are on single chip it is compact while microprocessor is bulky.

3. Microcontrollers are made by using complementary metal oxide semiconductor technology so they are far cheaper than microprocessors. In addition the applications made with microcontrollers are cheaper because they need lesser external components, while the overall cost of systems made with microprocessors are high because of the high number of external components required for such systems.

4. Processing speed of microcontrollers is about 8 MHz to 50 MHz, but in contrary processing speed of general microprocessors is above 1 GHz so it works much faster than microcontrollers.

5. Generally microcontrollers have power saving system, like idle mode or power saving mode so overall it uses less power and also since external components are low overall consumption of power is less. While in microprocessors generally there is no power saving system and also many external components are used with it, so its power consumption is high in comparison with microcontrollers.

What is Pre processor ?

Preprocessor is a Program That processes its input data to produce output that is used as input to another program. The output is said to be a pre processed form of the input data, which is often used by some subsequent programs like compilers. The amount and kind of processing done depends on the nature of the pre processor.

Some Pre-processors Are Only Capable Of Performing Relatively Simple Textual Substitutions And Macro Expansions, while others have the power of full-fledged programming languages.

A Common Example From Computer programming is the processing performed on source code before the next step of compilation. In some computer languages (e.g., C and PL/I) there is a phase of translation known as preprocessing. It can also include macro processing, file inclusion and language extensions.

Which is better a char, short or int type for optimization?

Where possible, it is best to avoid using char and short as local variables. For the types char and short the compiler needs to reduce the size of the local variable to 8 or 16 bits after each assignment. This is called sign-extending for signed variables and zeroextending for unsigned variables.

It is implemented by shifting the register left by 24 or 16 bits, followed by a signed or unsigned shift right by the same amount, taking two instructions (zero-extension of an unsigned char takes one instruction).

These shifts can be avoided by using int and unsigned int for local variables. This is particularly important for calculations which first load data into local variables and then process the data inside the local variables.

Even if data is input and output as 8- or 16-bit quantities, it is worth considering processing them as 32-bit quantities

What are inline functions?

The ARM compilers support inline functions with the keyword __inline.

These results in each call to an inline function being substituted by its body, instead of a normal call.

This results in faster code, but it adversely affects code size, particularly if the inline function is large and used often.

Why do we need virtual device drivers when we have physical device drivers?

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 carry out 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 software.

What is Dirac delta function and its Fourier transform and its importance?

Dirac delta function is a continuous time function with unit are and infinite amplitude at t=0.

The Fourier transform of Dirac delta function is 1.

Using Dirac delta as an input to the system, we can get the system response. It is used to study the behaviour of the circuit. We can use this system behaviour to find the output for any input.

Differentiate testing and verification?

Verification is a front end process and testing is a post silicon process. Verification is to verify the functionality of the design during the design cycle. Testing is find manufacturing faults.

```
a &= ~BIT3;

}
```

Some people prefer to define a mask together with manifest constants for the set and clear values. This is also acceptable. The element that I'm looking for is the use of manifest constants, together with the |= and &= ~ constructs

#define cat(x,y) x##y concatenates x to y. But cat(cat(1,2),3) does not expand but gives preprocessor warning. Why?

In this case the cat(x,y) is the macro which is defined by using the preprocessor directive , this will be substituted only at the place where it is called in this example it happens like this

cat(1,2)##3 which will once again become 1##2##3

here if we use ## in between we can join or concatenat only two variables that why it gives a preprocessor warning.

What Does Malloc Do? What Will Happen If We Have A Statement Like Malloc(sizeof(0));

Malloc is the function that is used for dynamically allocating memory to the different variables. The malloc returns a memory pointer of void type (void *). The statement malloc(sizeof(0)) returns a valid integer pointer because sizeof(0) represents the size of memory taken up by the integer value of 0. The memory allocated by memory is not automatically cleaned up by the compiler after execution of the functions and should be cleaned up by the programmer using the free () function.

What Are Hard and Soft Real-Time Systems?

The hard real time Embedded Systems are the one 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.