**C and C++ and Embedded C Questions**

1. Object Oriented concepts
2. Reentrant and non reentrant functions

A function can be non-reentrant if it uses memory that is not on the stack.

* If a function uses a static variable or a global variable, or a dynamically-allocated object that it finds for itself, then it is non-reentrant and any two calls to the function can interfere.

1. Extern function or variables and other types(registers, volatile,…)

Register variables are a special case of automatic variables. Automatic variables are allocated storage in the memory of the computer; however, for most computers, accessing data in memory is considerably slower than processing in the CPU. These computers often have small amounts of storage within the CPU itself where data can be stored and accessed quickly. These storage cells are called *registers*.

1. Union and structure
2. #if and #ifdef used for
3. #define and marcos
4. PLL(phase locked loop) prescaler

A **phase-locked loop** or **phase lock loop** (PLL) is a [control system](http://en.wikipedia.org/wiki/Control_system) that tries to generate an output [signal](http://en.wikipedia.org/wiki/Signal_(electrical_engineering)) whose [phase](http://en.wikipedia.org/wiki/Phase_(waves)) is related to the phase of the input "reference" signal. It is an [electronic circuit](http://en.wikipedia.org/wiki/Electronic_circuit)consisting of a variable frequency [oscillator](http://en.wikipedia.org/wiki/Electronic_oscillator) and a [phase detector](http://en.wikipedia.org/wiki/Phase_detector). This circuit compares the phase of the input signal with the phase of the signal derived from its output oscillator and adjusts the frequency of its oscillator to keep the phases matched. The signal from the phase detector is used to control the oscillator in a [feedback loop](http://en.wikipedia.org/wiki/Feedback_loop).

1. Can (Controller Area Network) and Lin (Local Interconnection Network) default values, both are automotive protocols
2. Different between dynamic and static design
3. Little Indian and Big Indian

Big-endian and little-endian are terms that describe the order in which a sequence of [byte](http://whatis.techtarget.com/definition/0,289893,sid9_gci211721,00.html)s are stored in computer memory. Big-endian is an order in which the "big end" (most significant value in the sequence) is stored first (at the lowest storage address). Little-endian is an order in which the "little end" (least significant value in the sequence) is stored first. For example, in a big-endian computer, the two bytes required for the [hexadecimal](http://whatis.techtarget.com/definition/0,289893,sid9_gci212247,00.html)number 4F52 would be stored as 4F52 in storage (if 4F is stored at storage address 1000, for example, 52 will be at address 1001). In a little-endian system, it would be stored as 524F (52 at address 1000, 4F at 1001).

1. Logic analyzer

A **logic analyzer** is an electronic instrument which displays signals in a [digital circuit](http://en.wikipedia.org/wiki/Digital_circuit). A logic analyzer may convert the captured data into timing diagrams, protocol decodes, [state machine](http://en.wikipedia.org/wiki/State_machine) traces, [assembly language](http://en.wikipedia.org/wiki/Assembly_language), or correlate assembly with source-level software.

1. Priorities of operators (ex:12/8\*8)
2. For (;;)
3. There are also questions about concepts of **Embedded Systems**, **Microcontrollers** (I/O, Modulation, and ADC) **and Real Time Systems** (context switching).