嵌入式：

1. 关键字static的作用

对于static变量，

1. .h头文件中的#ifndef/#define/#endif作用：
2. #include <>与#include “”区别：
3. 描述实时系统的基本特性：
4. 全局变量与局部变量是否存在区别？
5. 堆栈溢出一般是由于什么原因？
6. 队列与栈的区别：
7. 局部变量是否能和全局变量重名？
8. 如何引用一个已经定义过的全局变量？
9. 全局变量能不能定义在可被多个.c文件包含的头文件中？
10. for(;1;)有什么问题？什么意思？
11. do…while()与while…do有什么区别？
12. 程序的内存分配
13. 什么是预编译，什么时候需要预编译？
14. 关键字const含义
15. 关键字volatile含义？应用例子：
16. 结构体与联合体有什么区别
17. #define与const区别：
18. 数组与指针区别：
19. 分别写出bool、int、float、指针类型的变量a与零比较的语句：
20. 讨论含参数的宏与函数优缺点：
21. 给定一个整型变量a，写两段代码，第一个设置a的bit 3，第二个清除a的bit 3。在以上操作中，保持其它位不变。

35.写一个“标准”的宏

（1）交换两个参数的宏定义

（2）输入两个参数，输出较小的参数

（3）1年中有多少秒？（忽略闰年）

1. 已知一个数组table，使用宏定义求出数组元素的个数
2. GPIO的输入输出模式有哪些
3. TCP与UDP区别：

硬件笔试题：

02请解释电阻、电容、电感封装的含义：0402、0603、0805

04请问电阻、电容、电感的封装大小分别与什么参数有关？

05如果某CPU有很多IO端口需要接上下拉电阻，电阻范围1~10K 欧姆均可。以下规格的电阻，您会选择哪一种：1K/1%、4.99K/1%、10K/1%、1K/5%、2.2K/5%、4.7K/5%、8.2K/5%、10K/5%、3.9K/10%、5.6K/10%、4.7K/20%？说明你选择该电阻的理由。

06请简述压敏电阻工作原理。

07请简述 PTC 热敏电阻作为电源电路保险丝的工作原理。

09某磁珠的参数为100R@100MHz，请解释参数的含义。

10请问共模电感的作用是什么？

11绿色发光二极管的导通压降大概是多少伏？

12如果一个LED指示灯没有定义颜色，红、绿、黄、橙、蓝、白色你会选择哪一种，为什么？

13请简述TVS瞬态电压抑制二极管的工作原理。

19为什么OD(开漏)门和OC(开集)门输出必须加上拉电阻？

22请简述一下DC-DC和LDO的区别。

23请问电荷泵升压电路一般应用在什么场合？电荷泵可以胜任大电流的应用吗，为什么？

27请解释WatchDog(看门狗)的工作原理。

30晶体管基本放大电路有共射、共集、共基三种接法，请简述这三种基本放大电路的特点。

32名词解释：耦合、去耦、旁路、滤波。

34无源滤波器和有源滤波器有什么区别？

36请问RS-232C标准的逻辑0和逻辑1电压范围是多少？

37名词解释：UART、USRT、USART。

38请问串口异步通信的字符帧格式由哪几部分组成？

39请列举您知道的差分平衡电平接口。

42您所遇到的需要控制单端阻抗为50欧姆、75欧姆的信号有哪些？您所遇到的需要控制差分阻抗为90欧姆、100欧姆、120欧姆的信号有哪些？

43差分线走线有两个原则：等长和等距。但在实际布线中可能无法两者都完全满足，那么请问是等长优先还是等距优先？

44为什么高频信号线的参考地平面要连续（即高频信号线不能跨岛）？

46请问什么是通孔、盲孔和埋孔？孔径多大可以做机械孔，孔径多小必须做激光孔？请问激光微型孔可以直接打在元件焊盘上吗，为什么？

47请问过孔有哪两个寄生参数？这两个寄生参数对电路有什么影响？

48您知道的画原理图和PCB的软件都有哪些？

49您知道的计算PCB阻抗的软件都有哪些？

50示波器铭牌一般都会标识两个参数，比如泰克TDS1002B示波器标识的60MHz和1GS/s，请解释这两个参数的含义。

53如果一个门电路，输入高电平阈值是2.0V，输入低电平阈值是0.8V。那么如果输入一个1.2V的电平，请问门电路工作在什么状态？

物联网部分：

1. 简述，你认识的物联网是一个什么样的存在？
2. 说说你知道的物联网通信技术有哪些？各有什么特点？选择一个你熟悉的通信协议，尽可能详细的进行讲解。
3. 列举两个物联网应用，并选择一个尽可能详细的介绍实现方法。
4. 怎么样可以让客户可以快速的接受一个技术方案？
5. **The**static**Keyword:**
   * In embedded C programming, the static keyword has two main purposes:
     1. **Allocating Single Storage Space:** When used within a function, a static variable is allocated a single storage space that persists across function calls. It retains its value between function invocations.
     2. **Associating with the Class (Not Instance):** When used outside a function (i.e., as a global variable), a static variable is associated with the class (or file) rather than any specific instance. It is accessible only within the same file where it is defined.
6. #ifndef**,**#define**, and**#endif**in Header Files:**
   * These preprocessor directives are used to prevent header file redefinition errors when including the same header in multiple files.
   * #ifndef: Checks if a macro (usually the header guard) is not defined.
   * #define: Defines the macro (usually the header guard) to prevent redefinition.
   * #endif: Ends the conditional compilation block.
7. **Difference Between**#include <>**and**#include ""**:**
   * Both #include <> and #include "" are used to include header files.
   * The difference lies in the search path:
     1. #include <>: Searches for the header file in system directories (standard library headers).
     2. #include "": Searches for the header file in the current directory first and then in system directories.
8. **Basic Characteristics of Real-Time Systems:**
   * Real-time systems are designed to complete specific tasks within specified time constraints.
   * Key characteristics:
     1. **Time Constraint:** Tasks must complete within predetermined time limits.
     2. **Predictability:** Behavior can be predicted and analyzed.
     3. **Reliability:** Must operate correctly even under varying conditions.
     4. **Interaction with External Environment:** Responds to external events or inputs.
9. **Difference Between Global and Local Variables:**
   * **Scope:**
     1. Global variables have global scope (accessible from any function).
     2. Local variables have local scope (limited to the function where they are defined).
   * **Lifetime:**
     1. Global variables exist throughout the program’s execution.
     2. Local variables exist only during the function’s execution.
   * **Visibility:**
     1. Global variables can be accessed from any function.
     2. Local variables are visible only within the function where they are declared.
10. **Common Causes of Stack Overflow:**
    * Stack overflow occurs when the call stack exceeds its allocated memory.
    * Common reasons:
      1. Excessive recursion (too many nested function calls).
      2. Large local variables or arrays.
      3. Infinite loops without proper termination conditions.
11. **Difference Between Queues and Stacks:**
    * **Queues:**
      1. Follows the First-In-First-Out (FIFO) order.
      2. Elements are added at the rear (enqueue) and removed from the front (dequeue).
    * **Stacks:**
      1. Follows the Last-In-First-Out (LIFO) order.
      2. Elements are added and removed from the same end (top).
12. **Local Variables and Global Variables with the Same Name:**
    * Yes, local variables can have the same name as global variables.
    * When accessing the variable within a function, the local variable takes precedence over the global one.
13. **Referencing an Already Defined Global Variable:**
    * Use the extern keyword to reference a global variable defined in another file.
    * Declare the variable using extern before using it in the current file.
14. **Defining Global Variables in Header Files Included by Multiple**.c**Files:**
    * Yes, global variables can be defined in a header file included by multiple .c files.
    * However, it’s essential to use include guards (#ifndef/#define/#endif) to prevent redefinition errors.