【机试题】2019大疆嵌入式笔试题A卷(附超详细解答)

前不久的大疆嵌入式线上笔试，可能是因为最近只是在做毕设项目，还没有来得及认真系统复习，直接崩了。就凭借着记忆，把一些记得住的笔试题分享一下，作下记录。

整个大疆嵌入式线上笔试，分为选择题（单选题、多选题）、填空题、简答题、编程题。也没有将所有的题目都记得，就分成填空选择题、简答题和编程题三块来介绍吧。

填空选择题

1、ARM指令和Thumb指令。（选择题）

解答：在ARM的体系结构中，可以工作在三种不同的状态，一是ARM状态，二是Thumb状态及Thumb-2状态，三是调试状态。而ARM状态和Thumb状态可以直接通过某些指令直接切换，都是在运行程序，只不过指令长度不一样而已。

ARM状态：arm处理器工作于32位指令的状态，所有指令均为32位；

Thumb状态：arm执行16位指令的状态，即16位状态；

thumb-2状态：这个状态是ARM7版本的ARM处理器所具有的新的状态，新的thumb-2内核技术兼有16位及32位指令，实现了更高的性能，更有效的功耗及更少地占用内存。总的来说，感觉这个状态除了兼有arm和thumb的优点外，还在这两种状态上有所提升，优化；

调试状态：处理器停机时进入调试状态。

也就是说：ARM状态，此时处理器执行32位的字对齐的ARM指令；Thumb状态，此时处理器执行16位的，半字对齐的THUMB指令。

ARM状态和Thumb状态切换程序：

从ARM到Thumb:LDR R0,=lable+1BX R0（状态将寄存器的最低位设置为1，BX指令、R0指令将进入thumb状态）；

从ARM到Thumb:LDR R0,=lable BX R0（寄存器最低位设置为0，BX指令、R0指令将进入arm状态）。

当处理器进行异常处理时，则从异常向量地址开始执行，将自动进入ARM状态。

关于这个知识点还有几个注意点：

ARM处理器复位后开始执行代码时总是只处于ARM状态；

Cortex-M3只有Thumb-2状态和调试状态；

由于Thumb-2具有16位/32位指令功能，因此有了thumb-2就无需Thumb了。

另外，具有Thumb-2技术的ARM处理器也无需再ARM状态和Thumb-2状态间进行切换了，因为thumb-2具有32位指令功能。

参考文章：ARM处理器的工作状态。

2、哪种总线方式是全双工类型、哪种总线方式传输的距离最短？（选择题）

解答：几种总线接口的通信方式的总结如下图所示：

UART、I2C、SPI、USB异同



3、TCP与UDP的区别。（选择题）

解答：TCP和UDP的区别总结如下图所示：

TCP和UDP的区别



4、Linux的用户态与内核态的转换方法。（选择题）

解答：Linux下内核空间与用户空间进行通信的方式主要有syscall（system call）、procfs、ioctl 和netlink等。

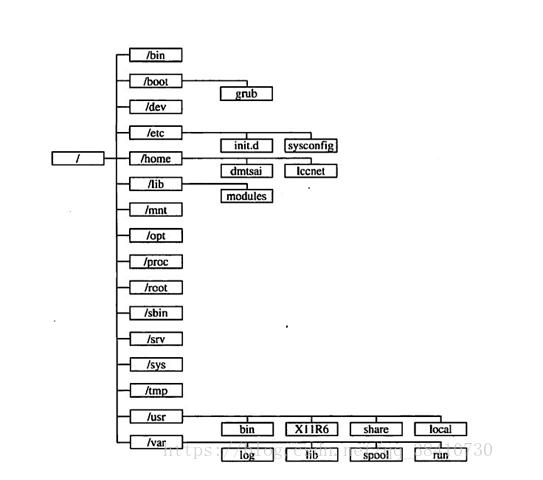
syscall：一般情况下，用户进程是不能访问内核的。它既不能访问内核所在的内存空间，也不能调用内核中的函数。Linux内核中设置了一组用于实现各种系统功能的子程序，用户可以通过调用他们访问linux内核的数据和函数，这些系统调用接口（SCI）称为系统调用；procfs：是一种特殊的伪文件系统，是Linux内核信息的抽象文件接口，大量内核中的信息以及可调参数都被作为常规文件映射到一个目录树中，这样我们就可以简单直接的通过echo或cat这样的文件操作命令对系统信息进行查取；

netlink：用户态应用使用标准的socket API就可以使用netlink提供的强大功能；

ioctl：函数是文件结构中的一个属性分量，就是说如果你的驱动程序提供了对ioctl的支持，用户就可以在用户程序中使用ioctl函数控制设备的I/O通道。

5、linux目录结构，选项是/usr、/tmp、/etc目录的作用。（选择题）

解答：linux目录图：



/usr：不是user的缩写，其实usr是Unix Software Resource的缩写，也就是Unix操作系统软件资源所放置的目录，而不是用户的数据啦。这点要注意。FHS建议所有软件开发者，应该将他们的数据合理的分别放置到这个目录下的次目录，而不要自行建立该软件自己独立的目录；

/tmp：这是让一般使用者或者是正在执行的程序暂时放置档案的地方。这个目录是任何人都能够存取的，所以你需要定期的清理一下。当然，重要资料不可放置在此目录啊。因为FHS 甚至建议在开机时，应该要将/tmp下的资料都删除；

/etc：系统主要的设定档几乎都放置在这个目录内，例如人员的帐号密码档、各种服务的启始档等等。一般来说，这个目录下的各档案属性是可以让一般使用者查阅的，但是只有root 有权力修改。FHS建议不要放置可执行档(binary)在这个目录中。比较重要的档案有：/etc/inittab,/etc/init.d/,/etc/modprobe.conf,/etc/X11/,/etc/fstab,/etc/sysconfig/等等。

参考文章：Linux文件目录结构详解。

6、下面这段程序的运行结果？（选择题）



可能是一般的内存对齐做习惯了，如果本题采用内存对齐的话，结果就是24（int4double char 7）。但是#pragma pack(1)让编译器将结构体数据强制按1来对齐。

每个特定平台上的编译器都有自己的默认“对齐系数”（32位机一般为4，64位机一般为8）。我们可以通过预编译命令#pragma pack(k)，k=1,2,4,8,16来改变这个系数，其中k就是需要指定的“对齐系数”。

只需牢记：

第一个数据成员放在offset为0的地方，对齐按照对齐系数和自身占用字节数中，二者比较小的那个进行对齐；

在数据成员完成各自对齐以后，struct或者union本身也要进行对齐，对齐将按照对齐系数和struct或者union中最大数据成员长度中比较小的那个进行；

参考文章：#pragma pack()的解读。

8、Linux中的文件/目录权限设置命令是什么？（选择题）

解答：chmod

9、下面四个选项是四个整数在内存中的存储情况，请选择其中最大的一个。（选择题）



