**填空题+选择题：5\*6=30分**

**问答题\*：3\*10=30分（四选三）**

**阅读理解题\*2：2\*10=20分。**

**系统的设计题20分。**

填空选择：

**一些基本的概念方法**

**对相关的一些国产的一些原型的一些产品或者是公司有个基本的了解**

1. 对于嵌入式系统的话，我们应该知道什么是嵌入式系统？什么是单片机，单片机跟入式系统的一个什么样的关系？
2. 作为一个嵌入式系统，实际上是包括这个硬件和软件两个大的部分啊，硬件和软件两个大的部分。这个硬件部分是最不可缺少的东西。软件部分就是讲一个电子字典的时候，我们讲这个系统的需求功能之外。对于嵌入式系统的选型，包括两个部分，一个是硬件和软件。
3. 关于架构，有这个是arm架构啊，这个是关于硬件的架构。然后这个软件的架构就是这个什么操作系统，比方说μcOSII，Linux，两个大的部分。那么实际上对于这个硬件的话，就是说我们除了这个硬件芯片选型，然后这个组件在选定之后，我们要把这个要有画原理图。要画原理图，要画PCB图。然后这个原理图的话，这个原理图我们是不是正确。我们可以要经过验证，这个验证可以经过仿真的实现。而负责画原理图、PCB图以及基于这个电子仿真的这样一种工具。EDA
4. 电子设计自动化工具（EDA）把它称之为芯片之母。工业软件芯片之母国外有几个典型的公司，三家企业分别为新思科技(Synopsys)、铿腾电子(Cadence)和Siemens EDA（领导国际），前两者均为美国企业，第三家是德国西门子旗下企业。国内：华大九天，他是一个上市公司。他也是通过做EDA在模拟电子这一方面很不错有先进优势的。
5. MCU在处理器的话这种架构有很多。从早期的这个单片机系列51单片机系列过来，有几种比较典型的比方说MIPS系列，X86系列，ARM系列，还有目前发展趋势比较好的RISC-V，每一种系列都有它的市场优势。
6. X86，它就在这个基于PC的这样台式机里面是具有市场主流的这种应用的。51系列的话，低端的应用大量的市场，单片机也是先锋系统的一种，大概占有70%的市场。ARM系列和MIPS系列，在移动终端，小区应用是一个主流。开放性最好的是RISC-V，在版权方面没有什么限制，我们国家对于这一块比较重视。整个这样一个系统里大概有百分之六七十的力量来自国内。很有希望，但是目前他的那个生态还没有完美建立。
7. 嵌入式系统的表现形式也有很多种。比方说我们的计算机MCU，后来我们发展的SOC片上系统，SOPC片上系统可以编程的，还有GPU，还有FPGA。关于计算机，我们不要去理解它，计算机一定是软件和硬件的结合体制，软件和硬件是可以相互转化的。历史上第一台计算机就全是由硬件构成的。现在绝大部分场景之下，这个计算机都有软件和硬件综合起来。但实际上某些特定的应用场景的话，比如说对于实时性要求很强的末端控制系统的话，也可以完全用FPGA来实现，全硬件。用硬件实现软件的功能，因为场景的不一样。
8. 做芯片的话，像ARM芯片的话，哪个厂家在做？国内做ARM芯片的很多，通用的ARM芯片有珠海的全志科技，还有福州的瑞芯微；做GPU的比较好的也是一个上市公司，就是景嘉微；做FPGA国外的公司的话是Xilinx（赛灵思）、Altera（阿尔特拉）、Lattice（莱迪思）、Microsemi(美高森美)，国内是上海的复旦微。单片机51系列国外也很多，国内实际上来说也有一个比较有名的叫做宏晶电子，STC，又好又便宜。国内有对应有能力的厂家和国外相对的产品去配合去对应。所以就说如果我们要选自主普通产品的话，从单片机到ARM到GPU到FPGA，都有国内流行的比较有名的公司可以相对应的。
9. ARM。ARM的发展。ARM公司，ARM是一种芯片的总称，也是一种技术的总称，也是个公司的名字。ARM公司本身并不生产SOC或MCU也不生产芯片，只是跟全球100多家有名的半导体制造商出售IP CORE（IP内核）或者ISA（指令集架构）。我们做实验的STM32，就是意法半导体公司购买了ARM的这个IP CORE生产。ARM芯片的话，最开始的命名规则就是ARM，ARM7，ARM9，ARM11。然后就朝cortex-三个方向发展，cortex-M，cortex-R，cortex-A。-A在手机上用的比较多，高性能。-R是实时性比较强，-M是功耗低，成本比较低的。STM32意法半导体公司为cortex-M系列。对于强实时系统的话，我们应该对于硬件有一个基本的了解啊。
10. ARM与X86对比学习，硬件的架构上是不一样的，ARM有七种工作模式，几种模式是什么？每一种模式的特点是什么？要去清楚。
11. 对于操作系统，在嵌入式系统里操作系统可以分两大类，一种是实时的，一种非实时的。比如桌面，桌面是基于PC Windows的系统，它就是一个非实时的。一般的嵌入式设备有带操作系统的和不带操作系统的。带操作系统又可以分为强实时和弱实时，或者说硬实时和软实时，我们的μcOSII，还有PSource，PSource Plus，还有VxWorks都是典型的强实时操作系统。一般像RT Linux，鸿蒙，安卓，都是属于弱实时操作系统。
12. 一个最简单的操作系统μCOSII，这个操作系统有几种状态，休眠，就绪，运行，中断，挂起。对于操作系统状态之间如何转换，要清楚。
13. 操作系统主程序被中断后，进入中断状态，中断状态里面的话，任务恢复后不一定回到以前那个任务，因为中断需要花时间的，花了时间之后呢，可能外面的环境就发生变化了。之前的这个优先级高的任务，由以前的挂起状态变成了就绪状态。操作系统调用的时候，可能会把优先级最高的任务。
14. μCOSII，它一共是64个任务，其中有两个缺省的任务。一个是叫最低优先级的任务，一般来说idle（空闲）任务的话是不能删除的，统计任务可以删除。这64个任务的优先级的高低，怎么样去判断？调用系统是通过知道优先级任务表（优先级任务组++优先级任务表），每个都是一个字节，八位，八位构成8×8 64的一个矩阵。因此我们正是通过优先级任务组里面知道八组里面的哪一组有任务就绪。该组里面的一个表也是八位，要知道具体是哪一个组里面哪一个任务。根据任务就绪组和就绪表判断哪个优先级是最高。有多种优先级的任务可能存在。
15. 操作系统有一个任务控制块。计算机调度的时候就是通过任务控制块。计算机会管理一个任务控制块，每一个任务控制块还有一个指针，任务控制块的一个地址作为一个指针。然后从0~63的任务都把这个指针放在一个列表里。根据指针列表，根据任务优先级的高低找到任务控制块指针。根据这个任务控制块指针指向的地址找到任务控制块。任务控制块里面放的是一个任务的这个堆栈地址。一个任务，主要是包括程序本身和程序的堆栈。在运行的时候是运行态，在不运行的时候，把CPU的环境拷到任务堆栈里。我们要知道操作系统到底是怎样找到任务本身的？他只要找到任务控制块的指针就会找到任务控制块。找到任务控制块的话就可以找到任务的堆栈地址。把堆栈地址赋给PC里面的SP，可以恢复所有PC，SP寄存器，这个时候PC就能找到任务本身代码段的地址。
16. 操作系统里面的优先级反转。优先级高的任务，想得到某一个资源。而这个资源被优先级低任务霸占的没有释放，他就不能获得，不能从挂起态变成就绪态。这时候其他不需要使用这个资源的优先级低的任务，就比这个优先高的任务先执行完。什么叫优先级反转？优先级反转怎么样去消除啊？什么措施啊？
17. 关于ARM的异常（中断），七个异常。从复位异常开始，每一个异常有一个中断向量的地址，在ARM7里是固定的。比如复位是8个0H开始。软中断从008开始执行，每一个异常有一个异常的向量地址，异常向量地址是一个字，放的是异常的中断服务程序的入口地址。一旦产生异常，就通过向量地址找到中断服务入口地址。ARM发生异常的一条指令，包括指令的取指周期，译码周期和执行周期，还有可能包括中断周期。指令是存放在内存里的，计算机CPU里是空的，要到内存里取指令。取指令是花时间的，然后把指令进行译码，译码就决定指令是干什么的。然后执行，对于ARM来说，执行时的操作主要涉及到CPU里的操作。
18. ARM大量使用寄存器。在执行的时候，操作数要么来自于指令本身，称为立即数寻址，要么来自于寄存器。称为寄存器寻址。在指令执行过程中，ARM是不访问内存的，所以ARM是基于RISC速度是比较快的。一个指令在执行的末尾时，如CPU会通过硬件自动查询外部设备是不是有中断请求，如果外部设备有中断请求，CPU本身又是允许中断的，那么CPU就会响应中断，这时指令就多了一个中断周期。对于同一个机器，同一个程序的同一条指令在不同时刻执行的时候，可能上一个时刻执行指令时没有中断周期，而在下一个时刻执行该指令时就有中断周期。中断周期是由所谓的中断来引起完成的。
19. 中断周期在这个汇编里要保存断点，要关中断，要找到中断服务程序入口地址，在X86的汇编里面，通过push CS、push IP、push PSW来实现断点的保存工作。我们知道push都涉及到内存的操作，所以中断在x86里速度是慢的，这个中断引起完成的时间是长的。
20. 在基于RISC的ARM里，每一种异常模式都有自己私有的寄存器（小金库）。比如link，LR（链接计算器），还有SPSR，可以用LR寄存器保存之前被中断服务程序中断的那条指令的下一条指令的地址。可以用SPSR保存之前中断的程序的CPSR。状态的断点是从寄存器到寄存器，所以ARM在实现中断引起的过程当中不需要访问内存，很快。
21. 异常，也就是说指令中断。在中断周期中，指令在CPU内部的这个微操作过程也要清楚。第一步保存断点。保存断点的方式就是用当前异常模式下的LR寄存器。也就是R14保存之前被中断的程序的那条指令的下一条指令的地址。用当前模式下的SPSR寄存器保存之前被中断的程序工作段的CPSR；然后把CPSR里对应的模式位给改掉，改成我们当前这个模式，比方说普通中断模式，比方说快速中断模式啊，比方说软中断模式；接下来把状态设置为ARM状态；然后要把普通中断关闭，是否关闭快速中断取决于当前中断是快速中断还是普通中断。如果当前中断是快速中断，就必须关闭快速中断。如果当前为普通中断，就不需要关快速中断；接下来PC把程序指针指到中断服务程序对应的中断向量地址，在中断向量地址里取中断服务程序的入口地址。这样一个过程。我们应该能用语言去描述，或者是用那个伪指令把它写出来，表示我们对中断有一个比较深刻的了解。
22. 关于硬件设计，对于我们学的ARM简单的芯片，我们要对其架构比较清楚。它是ARM 7TDMI S3C44B0X，本来有32位地址线，可以接4 GB的内存空间，但是一个嵌入式系统一般用不了4GB。所以我们所学的芯片，ARM 7TDMI三星公司的实际上只能够引出256M空间，256M空间分为八个块。每一块是32M字节，每一块可以分为SRAM，DRAM，FLASH或者是等等。它的总线的宽度，速度都可以通过存储器控制器调试，配置。八个块要清楚。
23. ARM是基于RISC的，它的IO指令和内存是共享空间的。它所有IO的配置，所有的一些存储器控制器，中断控制器，以及IO设备的配置都是通过第一块里面的最高的4M的寄存器的地址配置的。
24. 对存储器控制器有了解，中断控制器有了解。中断控制器有一个主模块和三个从模块。对于普通中断，可以把它扩展到很多个优先级来进行配置，也可以把它设置为这个向量中断或者非向量中断。
25. 对I/O方面，一个基本的概念，有发送接收，RS 232C是全双工的，发送可以引起中断，接收也可以中断。可以显示查询模式、中断模式和DMA显示。一般对于单字节的使用，用中断方式就可以了啊，就是串口。
26. I2C实际上来说是系统和系统之间进行信息交互啊。它只有两根线，一个SDA一个SCA，通过这两根线多个主设备可以实现半双工的通信，因为在同一时刻要么发要么收，不能收发同时进行。I2C总线通信机制一定要清晰，一个主设备怎么知道这个I2c总线是空闲的。怎么知道总线没有存在竞争。
27. I2C一个主设备访问一个从设备时，首先是给出从设备的地址。从设备的地址是七位。然后第8位决定读或写。对七位地址高位在前还是低位在前，怎么发送的要清晰。I2C总线工作原理以及它怎么知道总线是否有竞争，怎么知道自己有没有竞争，怎样获取总线的使用权和放弃使用权。要搞清楚。
28. 看门狗，可以通过中断来恢复正常，也可以通过复位。一般把看门狗放到优先级最低的任务里去。这样一方面衡量看门狗是不是工作，另一方面也衡量我们任务的调度，是不是达到一种最优状态。
29. ARM有七种模式，有两种状态，ARM状态和THUMB状态编程，THUMB编程简单的应该会。里面的替换（基本概念）。
30. 只出了程序阅读题。给一个程序，读懂之后加注释，然后要说这个程序的结果是什么？完成的什么功能？我们不用你们自己去记那些指令，但是有指令给出来，分析这个程序的功能是什么？作用是什么？ARM特别适用于这个程序，C调汇编，汇编调C，ARM我们就一个atp CS，汇编程序资源之间调用的话是有规则的，这个参数带进去的参数是什么参数。返回的参数是什么参数。因此我们参数怎样传入，怎样返回规则应该要清晰，书上都有的。
31. 最后就是系统设计题目，系统设计题目来源于这个我们日常的生活，课堂上我们讲的是要做电子字典。要理解嵌入式系统最小的系统是什么？另外给我们给我们一些功能要求，我们能够把那个硬件的方块图的画出来，硬件方块图，不要求我们画硬件的原理图，硬件的框图应该是会的。并不要求具体细化到哪一个芯片，原理图。系统结构图我们要画出来基本的东西。第二个根据这个功能要求，我们要把软件的流程图啊，用这个自己的语言描述。或者说伪指令把它描述出来。给你提出一些改变的功能的话，你就要想利用所学知识怎么去实现？实际上这个呃设计情况说难也难说，容易很容易啊。