# 一、考试题型

考试题型分为填空题、名称解释、问答题和设计题四种类型，分别占比为30%、20%、30%和20%

1. 填空题

共15空，每空2分

如：

嵌入式应用程序hello.c，编译成ARM平台可执行程序hello，编译指令为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_，依据所做实验平台，假设要将其从PC端搬运到ARM平台，则使用什么工具完成\_\_\_\_\_\_\_\_\_\_\_\_\_(A. ADS B.gcc C.串口超级终端)

填空题基本为教材和实验指导书上内容，部分为课程上课时作业。

2. 名词解释

共4题，每题5分

3.问答题

共2题，30分

如：

简述嵌入式Linux系统构建过程中根文件系统有哪几部分组成，并简述每一部分的基本构建过程（要求写出源码名称，解压命令，裁剪命令、编译命令和安装命令等）

3.设计题

共1题，共20分

主要考察大家对嵌入式系统的基本设计能力，题目结合实验内容进行考察，有较高难度，请大家务必对我们的实验及实验报告认真学习和总结

**二、复习提要**

1.教材

2.实验及实验报告

3.课外补充测试题，见附录1-4

整个课程的脉络大家可以按照下面的知识点进行学习和掌握

1 了解嵌入式系统的一般定义方法及其相关含义。

2 了解基于ARM核的研究和商业运作模式。

3 了解嵌入式应用开发的一般流程及方法。

4 了解ADS集成开发环境及其使用方法。

5 了解arm-linux-\*相关工具的作用及其使用方法。

6 了解嵌入式远程GDB的使用方法。

7 了解CodeBlocks软件的基本使用方法。

8 了解设备驱动和一般应用程序的区别。

9 了解设备驱动的基本作用。

10 了解Linux支持的三类的硬件设备：字符、块和网络设备。

11 了解引入模块的基本原理和模块的本质。

12 了解Qt程序的优点。

13 了解使用Qt开发嵌入式应用的基本方法。

1 掌握非操作系统模式的嵌入式开发模型及开发所涉及的基本过程。

2 掌握基于操作系统模式的嵌入式开发模型及开发所涉及的基本过程。

3 掌握S3C6410看门狗工作的基本原理及控制方法。

4 掌握使用S3C6410芯片的GPIO控制LED灯打开或关闭的基本原理及控制方法。

5 掌握宿主机-目标板开发模式的基本原理和工作方法。

6 掌握bootloader、内核和文件系统等的编译、裁剪和烧写过程。

7 掌握Make及Makefile文件使用方法。

8 掌握模块的加载和卸载方法。

9 掌握Linux设备驱动模型的构建方法。

10 掌握Qt程序的信号和槽的工作机理。

11 掌握看门狗程序的编写原理和编写方法（非操作系统模式和操作系统模式）。

12 掌握跑马灯程序的编写原理和编写方法（非操作系统模式和操作系统模式）。

13 掌握嵌入式应用程序移植基本步骤和方法

14 掌握嵌入式sqlite基本使用命令和编程函数

**附录1：**

（1）ARM汇编程序由\_\_\_\_\_\_\_\_\_\_指令、\_\_\_\_\_\_\_\_\_\_指令和\_\_\_\_\_\_\_\_\_\_指令构成。

（2）ARM伪指令可以分为以下几类：\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_等。

（3）假设存储数据0x12345678于ARM微处理器内存0X30000000开始的位置，则0X30000001内存位置的数据为\_\_\_\_\_（A.0X12 B.0X34 C.0X56 D.0X78）（采用小端模式进行存储）

（4）一般情况下，ARM微处理器异常处理模式共有\_\_\_\_\_\_\_种，机器启动后第一条指令执行的是\_\_\_\_\_（A.复位异常处理函数指令 B.中断异常处理指令 C.IRQ异常处理指令 D.指令预取终止异常）。

（5）调用函数FUN(X,Y,Z)，则实参值分别通过\_\_\_\_、\_\_\_\_、\_\_\_寄存器来进行传递，如果参数超过4个，则参数传递规则为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

（6）举例列出一款ARM7TDMI微内核的嵌入式微处理器\_\_\_\_\_\_\_\_\_\_，ARM920T微内核的嵌入式微处理器\_\_\_\_\_\_\_\_\_\_，ARM11内核的嵌入式微处理器\_\_\_\_\_\_\_\_\_\_，并列举2款64位ARM微内核\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_。

（7）利用汇编和C混合编程，设计代码完成求a,b,c中最大值功能，要求写出汇编启动代码和C代码。

**附录2：**

1.嵌入式Linux操作系统包括 、 、 三部分组成。

2.在PC机上Linux系统编译使用的编译器名为 ，ARM处理器嵌入式编译器名为 。

3.bootloader的功能： 。

4.列出最常用的bootloader： 、 、 、 、 、 。

5.在uboot中，打印开发板上环境变量值的命令为 ，设置IP地址为192.168.1.1的命令为

，假如嵌入式内核名为vmlinux，通过tftp加载内核的命令为

，启动嵌入式Linux内核的命令为 。

6.从C语言角度来理解，嵌入式Linux内核仅是众多 的集合体，其有一个类似于main的函数，名称为 。

7.嵌入式Linux内核裁剪命令为 ；裁剪完成后生成配置选择文件 ；编译内核命令为 。

8.启动linux后常用的命令如ifconfig、cp、ls等通常位于 （A.bootloaderB.内核C.文件系统）中；嵌入式Linux环境下生成文件系统的常用工具为 ；以yaffs方法为例，将嵌入式文件夹压缩成一个yaffs2格式的文件系统命令为 。

9.构建嵌入式Linux系统时，烧写bootloader的工具为 ；烧写Linux内核一般使用

（A.bootloader B.文件系统 C.交叉编译器）来烧写，烧写文件系统一般使用

（A.bootloader B.文件系统 C.交叉编译器）来烧写。

10.简述构建嵌入式Linux系统基本过程

**附录3：**

1. Linux中自动生成makefile的工具集名称为 ；工具集包括aclocal、autoscan、autoconf、autoheader和 等几部分组成。

2.一个源码文件hello.c利用自动生成makefile的工具集后得到文件一批文件后，利用命令

生成Makefile，利用命令 生成发行压缩包文件hello.tar.gz。

3.嵌入式Linux应用程序移植常用configure命令生成Makefile,一般来说，指定安装目录为/opt/soft的命令为 ；如果待生成的可执行目标板为ARM处理器，则命令为 。

4.以web服务器boa移植为例，简述其基本过程。

5.嵌入式数据库经常使用sqlite，为可以在开发板上运行的轻型数库。通常情况下sqlite提供

和 两种使用方式。

6.在sqlite下创建数据库stu.db命令为 ；创建包含学号，姓名和电话号码的数据表StuPhone命令为 ；插入10010201,wangming,13900008888记录的命令为 。

7.利用函数调用创建数据库stu.db的形式为 ；创建包含学号，姓名和电话号码的数据表StuPhone函数调用形式为 ；插入10010201,wangming,13900008888记录的函数调用形式为 。

**附录4：**

1. 嵌入式Linux内核是可裁剪系统，通常情况下使用 机制进行设计。模块程序和应用程序分别位于Linux系统的 空间和 空间。

2. 嵌入式Linux系统模块程序hello.c被编译成模块hello.ko后，动态插入内核的命令为

，查看Linux内核中有哪些模块的命令为

，删除内核中模块hello.ko的命令为 。

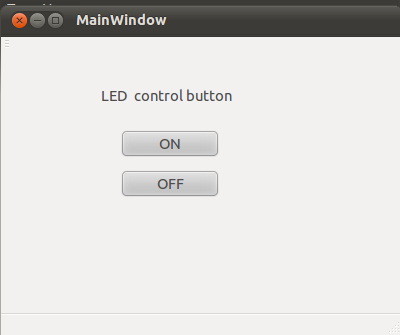
3.编写一模块，向内核中添加两整数相加功能函数和两整数相减功能函数，并要求在加载模块时打印出“hello，I am in kernel now!”，卸载模块时打印“hello, I will leave from kernel now!”,编写Makefile，并简述加载到内核和卸载出内核的基本命令。

**附录5：**

1. 在嵌入式Linux开发过程中，EmbeddedQT通常情况下用于 （A.内核代码B.驱动代码C.图形应用程序）设计。EmbeddedQT采用工具开发包的形式提供给用户，一般情况下包括 、 和Makefile 制作工具，字体国际化工具等。

2. QT开发中采用 机制来连接两个对象之间的通讯，假设对象a的clicked信号和对象b的handleFunction（）相关联，对应语句为：

3．假设有嵌入式LED报警灯驱动设备文件/dev/led，点亮led灯调用函数ioctl(fd,LEDON),熄灭led灯调用函数ioctl(fd,LEDOFF),请设计QT应用程序完成如下任务，点击ON按钮点亮LED灯，点击OFF按钮熄灭LED灯。写出这两个按钮的相关联的函数。假设ON按钮名称为m\_on,OFF按钮名称为m\_off。



**附录6：**

1. 嵌入式Linux驱动设备分为\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_和网络设备三种类型，其中通常情况下键盘、鼠标、LCD屏等设备驱动属于\_\_\_\_\_\_\_\_\_\_\_\_\_设备，硬盘设备属于\_\_\_\_\_\_\_\_\_\_\_\_\_\_设备。

2. 以模块的方式向内核添加功能函数int addFun(int a,int b)和int subFun(int a,int b)，要求编写模块文件，Makefile文件，假设模块名为hello.ko，写出加载模块到内核的命令。

设备要能被使用，通常需要设备驱动软件，在Linux系统中，设备驱动驱动软件位于\_\_\_\_\_\_\_\_\_\_\_(A.内核空间 B.用户空间)，假设有LED报警灯设备驱动软件leddrv.ko，则加载该驱动软件到内核命令为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_，测试完成后，将该软件从内核中卸载命令为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

3. 除网络设备外，设备驱动通常采用\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_方式进行访问，一般包括\_\_\_\_\_\_\_\_\_\_\_\_\_、读写控制设备，最后\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。在Linux中设备文件名本质上是设备号，将两者之间建立关联的命令为mknod，假设LED设备驱动主设备号为253，次设备号为0，设备名为leddev，则建立设备文件的具体命令为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

4．简述编写看门狗驱动基本步骤