

1、交叉编译的概念

在一种计算机环境中运行的编译程序，能编译出在另外一种环境下运行的代码，我们就称这种编译器支持交叉编译。这个编译过程就叫交叉编译。简单地说，就是在一个平台上生成另一个平台上的可执行代码。

2、简述嵌入式系统的软硬件结构？

嵌入式系统软件包括：设备驱动接口(DDI)、实时操作系统(RTOS)、可编程应用接口(API)和应用软件。嵌入式系统硬件包括：微处理器、外围电路和外设。

3、简述通用计算机和嵌入式系统的主要区别？

通用计算机上要应用千数值计算、信息处理；而嵌入式系统上要应用千控制领域，体积小，应用灵活。

4、嵌入式系统的含义、特点，嵌入式系统软件特征

含义：嵌入式系统被定义为：以应用为中心、以计算机技术为基础、软件硬件可裁剪、适应应用系统对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。

特点：嵌入式系统是面向用户、面向产品、面向应用的。嵌入式系统的硬件和软件都必须高效率地设计，量体裁衣、去除冗余，力争在同样的硅片面积上实现更高的性能。嵌入式系统产品一般具有较长的生命周期，其发展也体现出稳定性。

软件特征：

- (1) 软件要求固态化存储
- (2) 软件代码高质量、高可靠性
- (3) 系统软件(OS)的高实时性是基本要求
- (4) 多任务操作系统是知识集成的平台和走向工业化道路的基础

5、简述嵌入式系统的组成

嵌入式系统通常由包含有嵌入式处理器、嵌入式操作系统、应用软件和外围设备接口的嵌入式计算机系统和执行装置(被控对象)组成。嵌入式计算机系统是整个嵌入式系统的核心，可以分为硬件层、中间层、系统软件层和应用软件层。执行装置接受嵌入式计算机系统发出的控制命令，执行所规定的操作或任务。

6、简单介绍嵌入式系统的开发方法

- 1) 崩溃与烧制 (Crash and Burn)

(1) 在宿主机 PC 上编写代码；(2) 在宿主机 PC 上编译应用程序，生成可执行程序；(3) 固化 (Burn) 到目标机的非易失性存储器 (EPROM、FLASH 等) 中；(4) 启动运行，若正确则转到；(5) 不正确则改写代码，纠正错误；(6) 返回 (7) 重复操作；(8) 固化成功。

2) ROM 监控程序。

(1) 编写代码；(2) 编译应用代码；(3) 将代码下载到目标机上的 RAM 空间；(4) 用户使用调试器进行调试；(5) 如果程序正常转到第 (8) 步；(6) 发现程序有问题，在调试器的帮助下定位错误；(7) 修改错误，重复 (2) 到 (7) 步；(8) 将程序固化到目标机上；

3) ROM 仿真器

ROM 仿真器在一定程度上可以认为是一种用于替代目标机上 ROM 芯片的设备，仅用于 ROM 芯片无法获得的情况。ROM 仿真器通常和 ROM Monitor 配合使用。

4) 在线仿真器

在线仿真器 (ICE) 是一种用于替代目标机上 CPU 的设备。同时支持软件断点和硬件断点的设置；设置各种复杂的断点和触发器；实时跟踪目标程序的运行；选择性的跟踪程序的运行；支持 “时间邮票 (Time Stamp)”；允许用户设置 “定时器”；提供 “Shadow RAM”，能在不中断被调试程序运行的情况下查看内存和变量，即非干扰调试查询。

5) 片上调试

片上调试 (OCD) 是 CPU 芯片提供的一种调试功能，可以认为是一种廉价的 ICE 功能。OCD 方式的主要优点是：不占用目标机的资源，而调试环境和最终的程序运行环境基本一致；支持软、硬件断点，提供跟踪功能，可以精确计量程序的执行时间，支持时序分析等功能。OCD 方式的主要缺点是：调试的实时性不如 ICE 强，不支持非干扰调试查询，CPU 必须具有 OCD 功能，OCD 存在各种实现且标准不统一。

6) 模拟器

模拟器 (Simulator) 通常是指令级的，相当于在宿主机上虚拟了一台目标机，该目标机可以与宿主机的 CPU 不同类型。利用模拟器的最大好处就是可以不用真正的目标机，可以在目标机环境并不存在的条件下开发目标机上的应用系统，并且在调试时可以利用宿主机资源提供更详细的错误诊断信息。

7、简述 MCU 和 DSP 的差异？

MCU (微控制单元) 主要偏重于传输控制，DSP (数字信号处理) 主要偏重于信号处理与运算。

8、简述 ARM 和 S3C2410 之间的关系？

S3C2410 是由三星公司生产的，采用的是 ARM 公司的 ARM920T 内核 + 外围电路 + AMBA 总线 + SDRAM(4K) - i-J/O 组成，是一个 MCU。ARM 则是由 ARM 公司生产的一类 MPU，是对一类 MPV 的通称。

9、ARM 微处理器的运行模式有哪几种，请列举并说明

7 种，分别为：

- (1) 用户模式 (User , usr)：正常程序执行时， ARM 处理器所处的状态。
- (2) 快速中断模式 (FIQ, fiq)：用于快速数据传输和通道处理。
- (3) 外部中断模式 (IRQ, irq)：用于通常的中断处理。
- (4) 特权模式 (Supervisor , sve)：供操作系统使用的一种保护模式。
- (5) 数据访问中止模式 (Abort , abt)：当数据或指令预取终止时进入该模式，用于虚拟存储及存储保护。
- (6) 未定义指令终止模式 (Undefined , und)：用于支持硬件协处理器软件仿真。
- (7) 系统模式 (System , sys)：用于运行特权级的操作系统任务。

10、什么是嵌入式操作系统(RTOS), 为什么需要 RTOS?

实时多任务操作系统 (RTOS) 是嵌入式应用软件的基础和开发平台，是一段嵌入在目标代码中的软件，用户的其他应用程序都建立在 RTOS 之上。 RTOS 通常包括与硬件相关的底层驱动软件、系统内核、设备驱动接口、通信协议、图形界面、标准化浏览器 Browser 等。 RTOS 还是一个标准的内核，将 CPU 时间、中断、 I/O、定时器等资源都包装起来，留给用户一个标准的 API，并根据各个任务的优先级，合理地在不同任务之间分配 CPU 时间。随着应用的复杂化，一个嵌入式控制器系统可能要同时控制 / 监视很多外设，要求有实时响应，有很多处理任务，各个任务之间有多种信息传递，如果仍采用原来的程序设计方法存在两个问题。一是中断可能得不到及时响应，处理时间过长，这对于一些控制场合是不允许的，对于网络通信方面则会降低系统整体的信息流量。二是系统任务多，要考虑的各种可能也多，各种资源如调度不当就会发生死锁，降低软件可靠性，程序编写任务量成指数增加。

11、RTOS 的功能和特点

功能：

(1) 任务管理：

分时操作系统中的基本调度单位一般是进程（或者线程），而对于实时操作系统，其内核调度的基本单位是任务。任务一般由任务控制块、程序区、数据区、堆栈区组成，堆栈又分为系统堆栈和用户堆栈。任务的驱动一般是基于消息或者事件的，即任务的设计是按照依次处理可能接收到的消息和事件，周而复始轮询循环的。实时操作系统中的任务有四种状态：运行 (Executing)，就绪 (Ready)，挂起 (Suspended)，冬眠 (Dormant)。

(2) 任务间同步和通信：

主要实时操作系统的任务间同步和通信的机制有：消息、事件、信号量；少部分实时操作系统仍使用邮箱机制；还有一些实时操作系统提供了共享内存的任务间通信机制。

(3) 内存管理：

实时操作系统内存管理模式可以分为实模式与保护模式。目前主流的实时操作系统一般都可以提供两种模式，让用户根据应用自主选择。另外，实时操作系统的内存管理还有对于内存的优化分配，以尽量减少整个系统的内存占有量的要求。

(4) 实时时钟服务：

实时时钟是系统调度的基础，也是系统定时服务器的基础。实时时钟服务一般包括定时唤醒 (tm_wkafter 或者 tm_wkwhen)、定时事件 (tm_evafter 或者 tm_evwhen) 机制。

(5) 中断管理服务：

是操作系统的一个核心和基本的功能。实时操作系统要求中断处理程序更加短小、精悍，以

减少中断禁止时间和中断延迟时间。

RTOS 的优点：

(1) 保证任务执行的实时性；(2) 简化多任务切换和资源分配及网络消息管理；(3) 模块化便于软件扩展、移植和再使用；(4) 提高编写出的软件可靠性；(5) 使软件开发从 “小生产阶段 ” 进入到 “大生产阶段 ”。

12、请简述 ARM 微处理器的用户模式和特权模式，以及用户模式向特权模式的转换。

(1) 除了用户模式之外的其他 6 种处理器模式称为特权模式。特权模式下，程序可以访问所有的系统资源，也可以任意地进行处理器模式的切换。(2) 特权模式中，除系统模式外，其他 5 种模式又称为异常模式。大多数的用户程序运行在用户模式下，此时，应用程序不能够访问一些受操作系统保护的系统资源，应用程序也不能直接进行处理器模式的切换。(3) 用户模式下，当需要进行处理器模式切换时，应用程序可以产生异常处理，在异常处理中进行处理器模式的切换。

13、NFS 服务器的功能？安装 NFS 服务器的步骤？

功能：实现信息的共享。

步骤：1、关闭防火墙；2、设置目标主机的 IP 地址；3、配置 NFS 服务器；4、设置 NFS 服务器的 IP 地址；5、启动 NFS 服务；6、挂载 NFS 。

14、Linux 优点以及作为嵌入式操作系统的优势

优点：提供了先进的网络支持、多任务、多用户、符合 IEEE POSIX 标准、支持数十种文件系统格式、完全运行于保护模式、开放源代码、采用先进的内存管理机制，更加有效地利用物理内存。

优势：Linux 低成本开发系统、可应用于多种硬件平台、可定制的内核、性能优异、良好的网络支持。

15、嵌入式计算机按照嵌入式处理器的不同，可以分成哪几类？并简要介绍各类处理器的特点

(1) 嵌入式微处理器 (Embedded Microprocessor Unit, EMPU)

嵌入式微处理器的基础是通用计算机中的 CPU。为了满足嵌入式应用的特殊要求，嵌入式微处理器虽然在功能上和标准微处理器基本是一样的，但在工作温度、抗电磁干扰、可靠性等方面一般都做了各种增强。嵌入式处理器目前主要有 Am186/88、386EX、SC-400、Power PC、68000、MIPS、ARM 系列等。

(2) 嵌入式微控制器 (Microcontroller Unit, MCU) 又称单片机

嵌入式微控制器一般以某一种微处理器内核为核心，芯片内部集成 ROM/EPROM、RAM、

总线、总线逻辑、定时 /计数器、 WatchDog、I/O 、串行口、脉宽调制输出、 A/D 、D/A 、Flash RAM 、EEPROM 等各种必要功能和外设。

(3) 嵌入式 DSP 处理器 (Embedded Digital Signal Processor, EDSP) DSP 处理器对系统结构和指令进行了特殊设计,使其适合于执行 DSP 算法,编译效率较高,指令执行速度也较高。嵌入式 DSP 处理器有两个发展来源,一是 DSP 处理器经过单片化、EMC 改造、增加片上外设成为嵌入式 DSP 处理器;二是在通用单片机或 SOC 中增加 DSP 协处理器。

(4) 嵌入式片上系统 SOC(System On Chip)

随着 EDI 的推广和 VLSI 设计的普及化,及半导体工艺的迅速发展,在一个硅片上实现一个更为复杂的系统的时代已来临,这就是 System On Chip(SOC)。用户只需定义出其整个应用系统,仿真通过后就可以将设计图交给半导体工厂制作样品。这样除个别无法集成的器件以外,整个嵌入式系统大部分均可集成到一块或几块芯片中去,应用系统电路板将变得很简洁,对于减小体积和功耗、提高可靠性非常有利。

16、RTOS 的体系结构

1)硬件抽象层(HAL)包含了所有和硬件平台相关的代码,如上下文切换和 I/O 寄存器访问等等。它存在于 RTOS 的最底层,直接访问和控制硬件,对其上层的 RTOS 的机器无关代码提供访问和控制服务。这样可以简化 RTOS 内核的移植工作,除了设备驱动程序之外,在移植的时候只需要修改 HAL 的代码就可以了。

2)RTOS 内核(Kernel)是用来为大多数程序乃至 OS(网络、文件系统、驱动程序)构建一系列在抽象的文件上工作的抽象机,使用户程序及上层 OS 组件对系统设备透明。

3)在提供的 RTOS 接口上需要有对用户程序提供的函数接口,专门为用户定制网络、图形、视频等接口。并且提供驱动程序开发界面,方便开发者对不同需求的设备定制驱动程序。一般来说,RTOS 内核的实现都为微内核的体系结构。所谓微内核技术是指将必需的功能(如进程管理、任务通信、中断处理、进程调度)放在内核中,而将那些不是非常重要的核心功能和服务(文件系统、存储管理、网络通信、设备管理)等等作为内核之上可配置的部分。这样,整个操作系统就是由提供一些基本服务机制的微内核加上一些服务进程构成,系统的各个系统调用和服务都是由内核发消息到不同的服务进程,服务进程执行相应的操作,然后以消息的方式返回内核。

17、BootLoader 的功能,启动代码的功能,以及两者之间的关系

Bootloader 不是一段代码,它是一个具有引导装载功能的完整的程序,如可以引导装载 linux 的 vivi , uboot , 以及通常 PC 机上的 BOIS 程序等。启动代码是系统上电或者复位后运行的第一段代码,是进入 C 语言的主函数之前需要执行的那段汇编代码。它的作用是在用户程序运行之前对系统硬件及软件环境进行必要的初始化并在最后使程序跳转到用户程序。它直接面对 ARM 处理器内核及硬件控制器进行编程,所执行的操作与具体的目标系统紧密相关。Bootloader 也包含了相应的启动代码,也就是说,启动代码是 Bootloader 的一部分,是所有相应应用程序的一部分。

18、ARM 处理器的特点

体积小、低功耗、低成本、高性能；支持 Thumb（16 位）/ARM（32 位）双指令集，能很好的兼容 8 位/16 位器件；大量使用寄存器，指令执行速度更快；大多数数据操作都在寄存器中完成；寻址方式灵活简单，执行效率高；指令长度固定。

19、gcc 编译器的常用参数有那些？它们的功能分别是什么？

gcc 编译器的常用参数： -o, -I, -L, -E, -S, -e

功能：-o 表示编译成一个可执行程序；-I 表示指定头文件目录；-L 表示指定库文件目录选项；-E 表示对源代码进行预编译；-s 表示编译成汇编代码；-e 表示把.e 文件转换为以 o 为扩展名的目标文件。

20、BootLoader 的结构分 2 部分，简述各部分的功能？

两部分：Stage1: 用汇编语言编写，主要进行设备的初始化；Stage2: 用 C 语言编写，增强程序的移植性和可读性。

21、设备驱动程序和应用程序的区别？

设备驱动程序在用户态下，而应用程序在用户态下：设备驱动程序从 module_init 开始，将初始化函数加入内核初始化函数列表中，在内核初始化时执行驱动的初始化函数，从而完成驱动的初始化和注册，之后驱动便开始工作，等待应用程序的调用；而应用程序从 main 函数开始执行。

22、简述根文件系统的创建过程？

1、建立基本的目录结构；2、交叉编译 BusyBox；3、创建配置文件；4、利用 cpio 工具创建根文件系统映像文件。