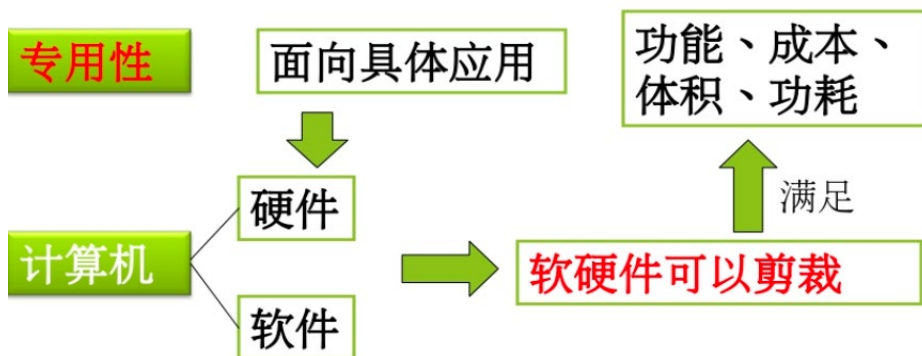


嵌入式概念

第一章 嵌入式系统概论

1. 嵌入式系统的定义

- ①IEEE 定义：嵌入式系统是控制、监视或者辅助设备、机器和车间运行的装置。
- ②嵌入式系统是以应用为中心，以计算机技术为基础，采用可剪裁软硬件，适用于对功能、可靠性、成本、体积、功耗等有严格要求的专用计算机系统。



2. 嵌入式系统发展的阶段

- ①嵌入式系统的出现和兴起 (1960-1970)
- ②嵌入式系统走向繁荣，软件、硬件日臻完善(1971-1989)
- ③嵌入式系统应用走向纵深化发展 (1990-至今)

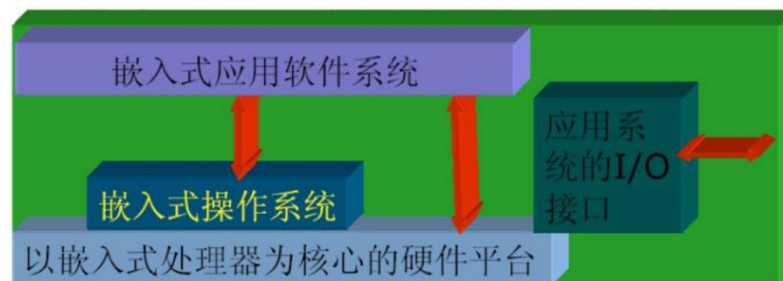
版本 2: 无操作系统阶段、简单操作系统阶段、实时操作系统阶段、面向 Internet 阶段。

3. 嵌入式系统构成

嵌入式系统一般由嵌入式微处理器、外围硬件设备、嵌入式操作系统(可选)，以及用户的应用软件系统等四个部分组成。

硬件包括：处理器、存储器、总线、I/O 设备和接口。

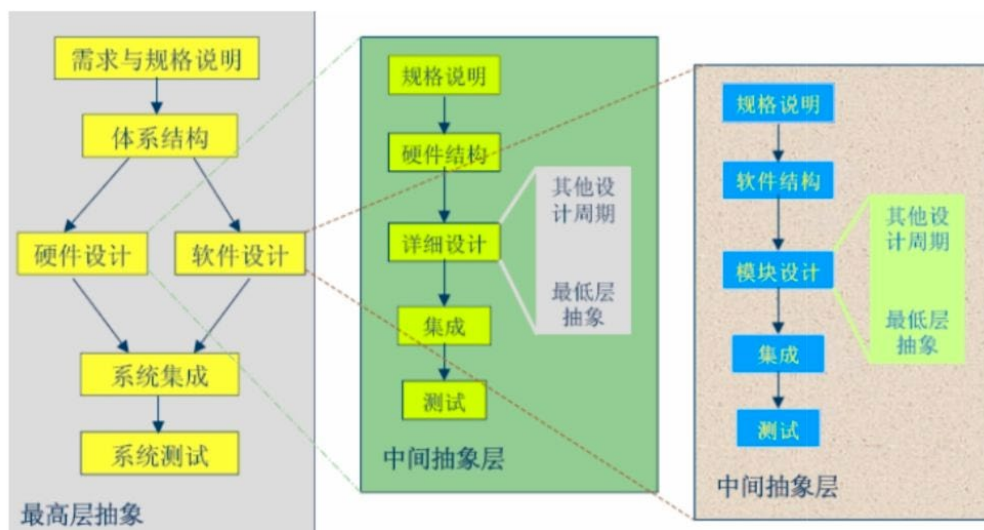
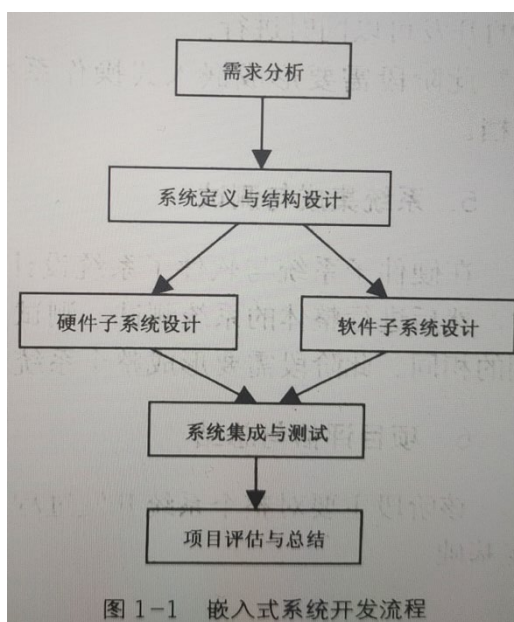
软件包括：嵌入式操作系统、嵌入式中间件、嵌入式应用系统等。



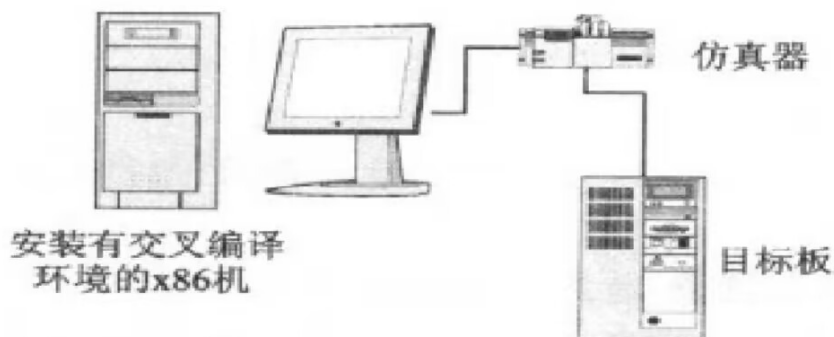
4. 嵌入式系统的特点

- ①系统精简②速度快③高实时性④专用性强⑤专业性强

5. 嵌入式系统的开发流程



6. 简述嵌入式系统软件的开发模式, 及开发环境中主要组成部分。



(1) 开发模式: 嵌入式系统的软件使用交叉开发平台进行开发,系统软件和应用软件在主机开发平台上开发,在嵌入式硬件平台上运行。

(2) 开发环境主要包含三个部分:

①宿主机: 开发嵌入式软件的系统。

②目标机：被开发的目的嵌入式系统。

③交叉编译器：进行交叉平台开发的主要软件工具。它是运行在一种处理器体系结构上,但是可以生成在另一种不同的处理器体系结构上运行的目标代码的编译器。

7. 什么是交叉编译？为什么要交叉编译？

(1) 编译源代码的平台和执行源代码编译后程序的平台是两个不同的平台。

(2) 原因：交叉编译的目标系统一般都是内存较小、显示设备简陋甚至没有，没有能力在其上进行本地编译；有能力进行源代码编译的平台 CPU 架构或操作系统与目标平台不同。

8. 嵌入式操作系统的定义和特性。列举常用的嵌入式操作系统。

(1) 定义：支持嵌入式系统应用的操作系统软件。

(2) 特点：①体积小②启动速度快③实时性强④可靠性高⑤专用性⑥可移植性

(3) 常用的嵌入式操作系统

- ❖ 嵌入式 Linux:①开源②内核小③效率高④运行稳定⑤可剪裁性好⑥更新速度快⑦跨平台⑧网络功能强大⑨丰富的软件开发工具
- ❖ Windows CE: ①模块化②结构化③基于 Win32 应用程序接口④与处理器无关
- ❖ VxWorks: ①高性能内核②友好的用户开发环境③价格贵
- ❖ Palm OS: ①操作系统的节能功能②合理的内存管理③数据是以数据库的格式来存储的
- ❖ uC/OS-II: ①开源②空间小③结构小巧④效率高⑤可剪裁⑥抢先式⑦可移植性强

9. 简述什么是实时操作系统，特性是什么？

(1) 定义：当外界事件发生或数据产生时，能够觉察事件的发生或接收数据，以足够快的速度予以处理，其处理的结果能在规定的时间之内控制生产过程或对处理系统做出快速响应，并控制所有实时任务协调一致运行的操作系统。提供及时响应和高可靠性是其主要特点。

(2) 特性：①多任务操作系统②多级中断机制③优先级调度机制

第二章 Linux 系统介绍

无。

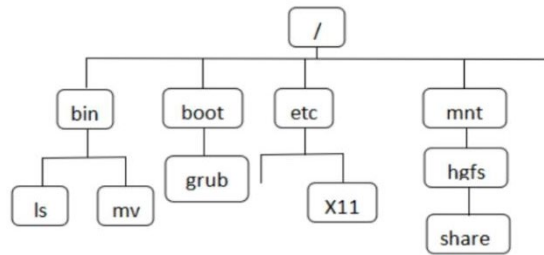
第三章 Linux 常用命令

1. Linux 中的文件类型

4 种基本文件类型:普通文件(-)、目录文件(d)、符号链接文件(l)和设备文件(块设备 b、字符设备 c)。

此外，还有一些其他类型的文件：命名管道文件(p)、socket 文件(s)。

2. Linux 文件系统是倒树状结构。



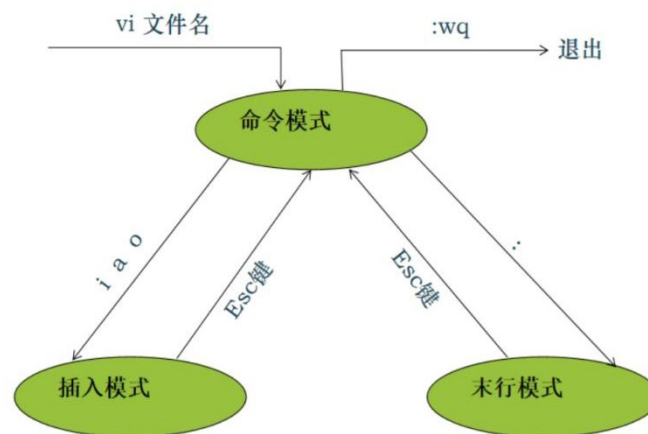
倒树状结构

Linux 中表示文件夹： /bin/ls

Windows 中表示文件夹： D:\vm\redflag4

3. Vi 编辑器的三种工作模式

三种工作模式：vi+文件名可以进入 vim 模式，:wq 可以退出 vim



第四章 Linux 服务

1. TFTP (简单文件传输协议)

(1) 概念

是 TCP/IP 协议簇中的一个用来在客户机与服务器之间进行简单文件传输的协议，它基于 UDP 协议实现，使用超时重传方式来保证数据的到达。

(2) 特点

功能简单，系统开销小

(3) 如何配置

- ①在终端运行“ntsysv”启动“服务”
- ②用键盘上的方向键移动光标到“TFTP”上
- ③然后按空格键在“TFTP”前加上“*”
- ④按 Tab 键将光标移到“确定”按钮
- ⑤按回车键退出。

(4) 如何启动

TFTP 是 xinetd 服务组合中的一员，不能单独启动，必须随 xinetd 一起启动。启动 xinetd 服务组合有命令行和图形界面两种方式。

(5) TFTP 在三种客户端下如何使用及其区别

- 1) 普通 Linux 客户端。具有一个命令行状态，在命令行状态下可以通过自身的命令实现文件下载、上传、状态设置、状态查看等功能。
- 2) 嵌入式 Linux 客户端。该客户端没有命令行状态，只能通过命令选项来设置数据的传输，并且只能以二进制的模式进行传输。
- 3) Windows 客户端。该客户端也没有命令行状态。

2. Telnet（远程登录协议）

是一个基于字符界面的远程登录协议。通过 Telnet，用户可以远程登录到指定的主机中，以本地主机为仿真终端对远程主机进行操作。

3. NFS（网络文件系统）

（1）概念

是一种将远程主机上的指定目录经网络挂载到本地系统的一种机制，通过网络文件系统，用户可以在本地系统上像操作本地目录一样来对远程主机的目录进行操作。

（2）如何配置服务器端

使用文本编辑工具编辑服务配置文件/etc/exports, 指定需要共享的目录。

（3）如何启动 NFS 服务？

- ①执行命令“service portmap start”启动 portmap 服务，
- ②执行命令“service nfs start”启动 NFS 服务。

（4）如何定制防火墙？

- 1)在/etc/services 文件中指定 mountd 和 rquotad 服务的端口为 1011 和 1012。
- 2)定制防火墙:放行端口和协议的数据包,使用“service iptables restart”重新启动防火墙。

（5）如何使用 NFS 服务？

- 1) 在服务器端执行命令“mount-t nfs 127.0.0.1:/home/share/mnt”，对 NFS 服务进行回环测试。
- 2) 在客户端输入命令“mount-t nfs 192.168.0.30:/home/share/mnt”，将服务端的共享目录挂载到客户端主机上。192.168.0.30 是服务器的 IP 地址。

（6）如果客户端不能正确挂载共享目录该怎么办？

- 1) 检查服务器端的 NFS 配置文件、portmap 服务和 NFS 服务。
- 2) 检查服务器的防火墙设置、文件/etc/hosts.allow 和/etc/hosts.deny 的内容。
- 3) 可使用命令“rpcinfo-p 127.0.0.1”查看端口使用情况。

4. NFS 实验步骤（可以不背，因为上面背了）

配置服务器-启动服务-定制防火墙-使用服务

要实现 NFS 功能，首先需要在服务器端进行配置，然后启动相应的服务后，客户端就可以挂载服务器端所指定的目录到本机的挂载点目录。

RPC（远程过程调用协议），NFS 可以看成是 RPC 服务的一个应用。NFS 是个文件系统，而 RPC 负责信息的传输。

rpc.nfsd，即 NFS 服务本身。作用是管理客户端能否登录服务器。

rpc.mountd，即 mountd 服务。作用是管理 NFS 的文件系统。

rpc.rquotad，即 rquotad 服务。作用是对远程用户提供用户配额信息。

步骤如下：

- ①查询 NFS 服务的软件包是否安装：
- ②开启 NFS 服务
- ③配置 NFS 服务
- ④启动 NFS 服务

⑤定制防火墙

步骤 1: 在/etc/services 文件中指定 mountd 和 rquotad 服务的端口为 1011 和 1012。

步骤 2: 存盘退出后, 要重启服务。

步骤 3: 定制防火墙

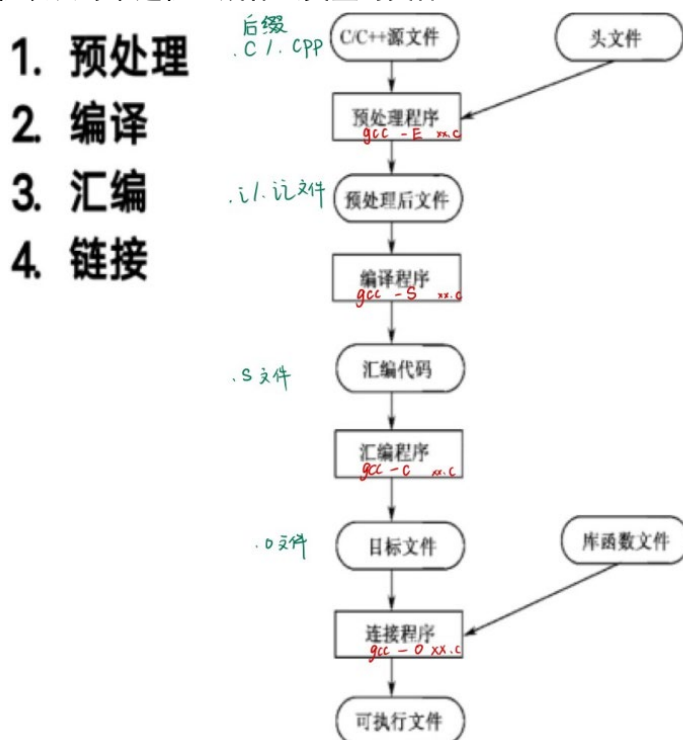
步骤 4: 重启防火墙

⑥验证 NFS 服务

⑦使用 NFS 服务

第五章 Linux 中的 C 语言编程

1. GCC 编译过程以及每个过程生成什么类型的文件。



2. 利用 GDB 调试的前提

在用 GCC 编译程序过程中, 使用 -g 选项, 生成 gdb 工作所必须的调试信息。

Eg: #gcc -g test.c -o test

3. 启动 gdb 的方法 :gdb 可执行文件名。

第六章 ARM 体系结构和指令系统

1. 哈佛和冯诺依曼的区别

①冯诺依曼结构: 是一种将系统指令存储器和数据存储器合并在一起的存储器结构。

②哈佛结构: 是一种将程序指令存储和数据存储分开的存储器结构。

③区别:

a. 哈佛结构处理器使用两个独立的存储器模块分别存储指令和数据, 每个存储模

块都不允许指令和数据并存。

b. 哈佛结构处理器使用独立的两条总线分别作为 CPU 与每个存储器之间的专用通信路径，两条总线毫无关联。

2. CORTEX 系列

①CORTEx-A 复杂的操作系统和大型的应用场合，

如：高端手机，金融事务处理机等

②CORTEx-R 实时应用场合，

如：大型发电机控制器，机械手臂等

③CORTEx-M 传统的低成本、低功耗、极速中断反应及高处理效率的自动控制场合，

如：医用器械，电子玩具，无线网络等

3. 存储器的存储格式（大端是顺序，小端是逆序）

Arm、windows 都是小端存储。

A. 小端存储法(0x01234567)

地址	0x8000	0x8001	0x8002	0x8003
数据（16进制表示）	0x67	0x45	0x23	0x01
数据（二进制表示）	01100111 1	01000101	00100011	00000001

B. 大端存储法

地址	0x8000	0x8001	0x8002	0x8003
数据（16进制表示）	0x01	0x23	0x45	0x67
数据（二进制表示）	00000001	00100011	01000101	01100111

4. 什么叫做寄存器，ARM 寄存器有哪些种类，寄存器又有什么作用？

（1）寄存器：CPU 内用来暂存指令、数据和地址的电脑存储器。

（2）种类：ARM 寄存器共有 37 个（31 个通用寄存器+6 个状态寄存器）。

（3）作用：

①寄存器 R0-R7（未备份寄存器，各对应 1 个物理寄存器）

②寄存器 R8-R12（备份寄存器，各对应 2 个物理寄存器）

③寄存器 R13-R14（备份寄存器，各对应 6 个物理寄存器）

R13 (SP)：堆栈指针。使异常程序不会破坏被其中断的应用程序的现场。

R14(LR)：连接寄存器。子程序的返回地址；异常模式返回的返回地址。

④寄存器 R15(PC，对应 1 个物理寄存器)：程序计数器。指出当前执行指令的地址。

⑤CPSR(对应 1 个物理寄存器)：设置处理器运行模式和工作状态；控制允许或者禁止 FIQ 和 IRQ；保存 ALU 当前操作信息，设置条件标志位，控制条件执行。

⑥SPSR(对应 5 个物理寄存器)：备份状态寄存器，当异常发生时，用于保存 CPSR 的状态，usr 和 sys 模式下无 SPSR。

寄存器R0~R7（各对应1个物理寄存器）；

寄存器R8~R12（各对应2个物理寄存器）；

寄存器R13(SP)~R14(LR)（各对应6个物理寄存器）；

寄存器R15-程序计数器PC（1个物理寄存器）；

寄存器CPSR（1个物理寄存器）和SPSR（5个寄存器）；

通用寄存器 (31)

状态寄存器 (6)

5. 嵌入式处理器的 7 种工作模式及两种工作状态

处理器的模式与CPSR

31	30	29	28	27	26	8	7	6	5	4	3	2	1	0	
N	Z	C	V	Q	DNM (RAZ)			I	F	T	M4	M3	M2	M1	M0
条件标识位					中断禁止位					控制位					
M4~M0					处理器模式					可访问的寄存器					
10000					用户模式					PC, R14~R0, CPSR (只读)					
10001					快速中断					PC, R14_fiq~R8_fiq, R7~R0,CPSR,SPSR_fiq					
10010					外部中断					PC, R14_irq~R13_irq, R12~R0,CPSR,SPSR_irq					
10011					特权模式					PC, R14_svc~R13_svc, R12~R0,CPSR,SPSR_svc					
10111					中止模式					PC, R14_abt~R13_abt, R12~R0,CPSR,SPSR_abt					
11011					未定义模式					PC, R14_und~R13_und, R12~R0,CPSR,SPSR_und					
11111					系统模式					PC, R14~R0, CPSR (读写)					

当异常发生时，在特权模式下，程序可以修改这些位 MSR CPSR_c, #0xd3

(1) 中断禁止位I、F、T：当I=1时禁止IRQ中断、当F=1时禁止FIQ中断

(2) T控制位：用于控制指令执行的状态，T=0表示执行ARM指令，T=1时标识执行Thumb指令；

(3) M控制位：控制处理器模式，具体含义如表所示

(1) 7 种工作模式

处理器模式	描述
用户模式 usr	正常用户程序执行的模式
快速中断模式 fiq	支持高速数据传输和通道处理
外部中断模式 irq	通常的中断处理
管理模式 svc	操作系统使用的一种保护模式
中止模式 abt	实现虚拟存储器或存储器保护
未定义模式 und	用于支持通过软件仿真的硬件协处理器
系统模式 sys	用于运行特权级的操作系统任务

除用户模式的其他 6 种模式为特权模式。在特权模式下，程序可以访问所有系统资源，也可以进行处理器模式切换。特权模式中除系统模式外的其他 5 种模式又称为异常模式。在每一种异常模式中都有一组寄存器，供相应的异常处理程序使用，这样既保证了在进入异常模式时用户模式下的寄存器内容不被破坏，又加快了各模式之间切换的速度。

(2) 2 种工作状态

- ARM 状态：执行 32 位指令。
- Thumb 状态：执行 16 位指令。
 - ❖ 切换不影响处理器的工作模式和寄存器的内容。
 - ❖ ARM 处理器在处理异常时，不管处理器处于什么状态，则都将切换到 ARM

状态。

- ❖ 用 Bx Rn 指令来进行两种状态的切换。其中 Bx 是跳转指令，而 Rn 是寄存器（1 个字，32 位），如果 Rn 的位为 1，则进入 Thumb 状态；如果 Rn 的位为 0，则进入 ARM 状态。

6. CPSR（程序状态寄存器）条件标志位含义

- 1) N（负数？）：当两个补码表示的带符号数运算时，N=1 表示运算结果为负数，N=0 表示运算结果为正数或 0
- 2) Z（零？）：Z=1 表示运算结果为零，Z=0 表示运算结果不为 0，对于 CMP 指令来说，Z=1 表示相互比较的两个数大小相等
- 3) C（进位/无借位？）：在加法指令中，若结果产生了进位，则 C=1，表示无符号数运算发生了上溢，其他情况下 C=0；
- 4) V（符号位溢出？）：对于加/减法运算指令，当操作数和运算结果为二进制补码表示的带符号数时，V=1 表示符号位溢出，通常其他的指令不影响 V 位。

7. 异常处理机制，异常响应步骤。

在异常被处理前，当前的处理器状态必须被保存，以便处理程序完成后，最后的程序可以被恢复。

异常响应	拷贝 CPSR 到 SPSR_<mode>
	保存返回地址到 LR_<mode>
	设置控制位CPSR_c位 (ARM状态，处理器模式，设置中断)
	设置 PC 为相应的异常向量（即跳转地址）
异常退出	从 SPSR_<mode>恢复CPSR
	从LR_<mode>恢复PC

8. ARM 处理器异常模式

异常类型	优先级	处理器模式	异常向量	说明
复位 RESET	1	管理模式	0x00000000	复位引脚有效时进入
未定义指令 UDEF	7	未定义模式	0x00000004	协处理器认为当前指令未定义时产生
软件中断 SWI	6	管理模式	0x00000008	用户定义的中断指令
指令预取中止 PABT	5	中止模式	0x0000000C	预取指令目标不存在或该地址不允许当前指令访问
数据中止访问 DABT	2	中止模式	0x00000010	指令目标地址不存在或该地址不允许当前指令访问
IRQ 中断	4	外部中断	0x00000018	有外部中断时发生的异常
FIQ 中断	3	快速中断	0x0000001C	有快速中断时发生的异常

10. ARM 处理器的特点

- ①体积小、低功耗、低成本、高性能。
- ②支持 Thumb（16 位）/ARM（32 位）双指令集。
- ③大量使用寄存器，指令执行速度更快。
- ④多数数据操作都在寄存器中完成。
- ⑤寻址方式灵活简单，执行效率高。
- ⑥指令长度固定。

【大题】随着技术的不断进步和人民生活水平的持续提高，智能家居正走进我们的生活。试运用嵌入式软件/硬件设计技术、传感器环境感知技术、无线通信等技术，设计一个简单智能家居系统用于控制家庭的各种电器设备，使人们的生活更方便和快捷。根据上述设计需求，给出该系统的设计过程，(1)系统功能定义；(2)工作原理表述；(3)硬件结构设计；(4)软件编程框架设计等。

答：根据嵌入式的开发流程来设计：

(可以在这里加入有抽象层次的开发流程图)

(1) 需求分析 (以功能需求分析为主)

- 1) 实现安全防范功能，对情况实现报警；
- 2) 对家居环境条件进行检测；
- 3) 各检测、传输设备采用无线方式组成网络进行信息传输；

(2) 系统定义和结构设计

系统由检测设备、传输设备、网关设备组成，组成簇树结构的无线传感器网络。

(3) 硬件设计 (子系统以传输设计设备为例)

1) 需求和规格说明：

- ① 工作温度范围：-20 ~ 85° C。
- ② 供电方式：交流 220V。
- ③ 无线频段：433MHz 频段。

2) 硬件结构

选用嵌入式操作系统，中继节点和路由节点采用相同硬件设计。

3) 详细设计

硬件平台：MSP430F149。
射频芯片：CC1100。
功放：SKY65004。

.....

4) 集成

5) 测试

(4) 软件设计 (子系统以传输设计设备为例)

1) 需求和规格说明：

- ① 簇树网络的建立与维护
- ② 稳定可靠的对数据进行传输。
- ③ 协调节点自身系统的运行。

2) 软件设计

中继节点的软件工作、路由节点的软件工作。

3) 模块设计 (以路由节点为例)

(见右图)

4) 集成

5) 测试

(5) 项目的集成与测试

进行整体系统测试，生成集成与测试文档。

(6) 项目的评估与总结

此次的传输设计设备非常成功，而网关设备的硬件平台控制中心选择失败，下次更正。

