IEEE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

devices used to control, monitor, or assist the operation of equipment, machinery or plants

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

实时系统: 指系统能够在限定的响应时间内提供所需水平的服务

嵌入式实时系统可分为

- 强实时型: 响应时间us ~ ms级;
- 一般实时: 响应时间ms ~ s级
- 弱实时型: 响应时间s级以上

嵌入式系统运行需要高可靠性保障, 需要忍受长时间、无人值守条件下的运行， 运行的环境恶劣

嵌入式系统的资源配置遵循够用就行

物联网从信息和通信的角度考虑，集中在anytime, anyplace, anyone去获取信息

Thumb/ARM双指令集, 采用固定长度的指令格式

嵌入式微处理器(EMPU) 以通用处理器CPU为基础

嵌入式微控制器(MCU) 单片机

嵌入式DSP处理器(EDSP)添加DSP协处理器

ARM内核中各个字母的表示

- T: 内含16位压缩指令集Thumb
- D: 支持片内Debug调试
- M: 采用增强型乘法器
- I: 内含嵌入式ICE宏单元
- S: 可综合的软核Softcore
- E:具有DSP的功能
- J: 允许直接执行Java字节码

一个嵌入式系统可以含有多个总线

CPU总线、存储器总线、I/O总线(AGP、PCI、ISA)

ARM-CPU总线，芯片外部提供的总线随芯片生产商不同而不同

AMBA协议包含两条总线: 高性能总线AHB, 外围设备总线APB

典型ARM 处理器架构

- ARM系列处理器共有37个寄存器
  - 31个通用寄存器
  - 6个不同工作模式所设立的专用状态寄存器
- ARM总共有7中不同的处理器模式
  - 用户模式(User)，这是程序正常执行的模式
  - 快中断模式(FIQ)，用于高速数据传输和通道处理
  - 外部中断模式(IRQ)，用于普通的外部中断请求处理
  - 管理模式(Supervisor)，这是供操作系统使用的保护模式
  - 数据访问中止模式(Abort)，用于虚拟存储和存储保护
  - 未定义模式(Undef)，用于支持硬件协处理器软件仿真
  - 系统模式(System)，用于运行特权级的操作系统任务

ARM七种工作魔兽之间, 模式切换的方法有软件控制、外部中断、异常处理

CPSR中包括 条件代码flag, IRQ控制位, FIQ控制位，Thumb控制位，模式位

ARM指令格式

```
<opcode> {<cond>} {S} <Rd>, <Rn> {,<operand2>}
```

- opcode: 指令助记符
- cond: 执行条件
- S是否影响标志位
- Rd: 目标寄存器
- Rn: 第一个操作数的寄存器

所有ARM指令都可以条件执行，而Thumb指令只有B(跳转)指令具有条件执行功能。

- 前变址模式 `LDR R0, [R1, #4] ; R0<- [R1+4]`
- 自动变址模式 `LDR R0, [R1,#4]!; R0<-[R1+4]; R1<-R1+4`
- 后变址模式 `LDR R0,[R1],#4; R0<-[R1]; R1<-R1+4`

根据堆栈生长方向 可以区分是满还是空, 向上增长为 递增, 向下增长为 递减

根据堆栈指针指向的目标 可以区分为 满 指向有效数据项, 空 指向 下一个空位置

BL指令在跳转时将当前PC+4存入R14

ARM 汇编

AAPCS代码寄存器使用

R0-R3 LR(R14), PSR 调用者保护

堆栈寻址用的从sp开始的偏移量

[sp, #offset] 偏移量需要乘以4,

最大的地址范围为256 * 4

软件模型

轮询的优势: 系统执行的时间是可以预计的, 所需的内存是可以预计的, 没有共享数据冲突的风险

前后台模式

前台为主循环

后台为中断处理

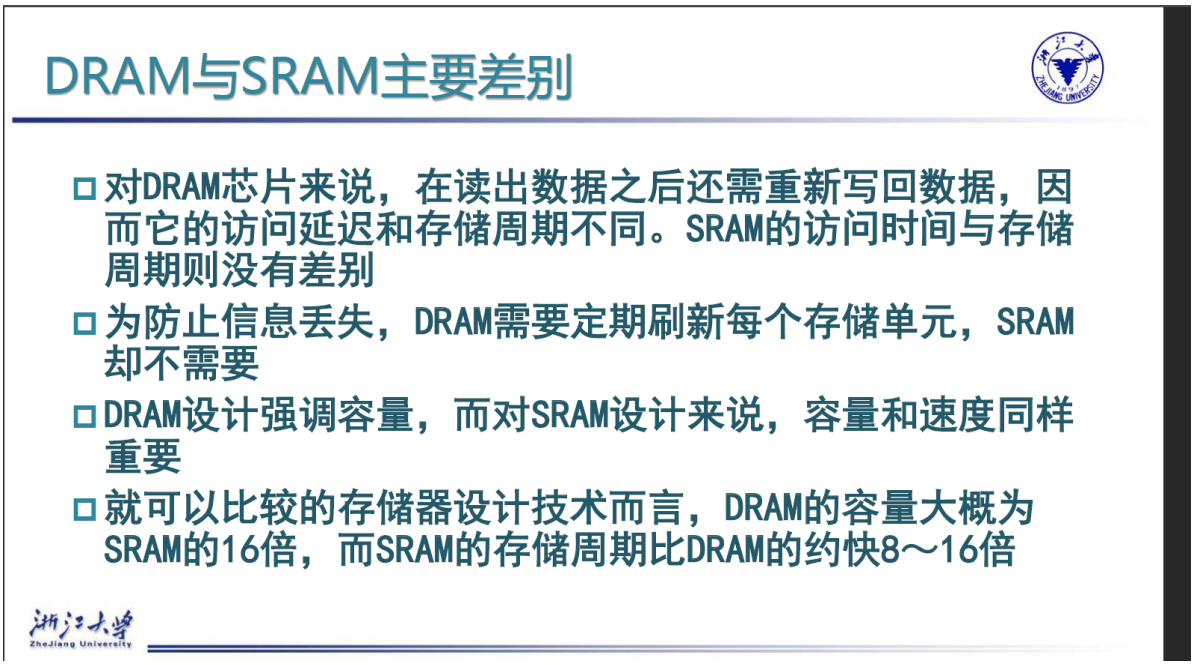
中断驱动: 没有主循环，大量时间睡眠，功耗低

适用于功能任务简单，任务之间冲突机会小，空闲时间多的产品，不适合任务间存在冲突可能的产品

动态执行队列: 将处理的各个功能编写成函数的形式，中断发生时，将相应的处理函数的指针放入队列中，主循环不断的读取队列，取出函数指针并调用该函数

动态队列有更好的响应时间，用更低的处理器速度，节省内存，降低功耗

这些存储器的速度，为触发器最快，寄存器次之， SRAM 再次，DRAM 较慢，然后是 FLASH ，磁盘最慢



看门狗定时器是一个很有用的外围设备。看门狗实际是一个简
单的定时器，在固定时间内若无正常清零，则自动复位处理器。

I 2 C Inter Integrated Circuit ）是由飞利浦半导体公司 20 年前开发并推出的 是一种具有多端控制能力的双线双向串行数据总线系统 用于 I/O 串行扩展。

USB(universal serial bus)

软中断指令 swi 在软件层模拟产生一个中断，这个中断会传送给CPU，常用于实现系统调用

- 常用的预定义寄存器名称
  - R0-R15，通用寄存器
  - A1-A4，入口参数、处理结果、暂存，同R0-R3
  - V1-V8，变量寄存器，同R4-R11
  - IP，保存栈指针SP，同R12；SP，栈指针，同R13
  - LR，链接寄存器，同R14；PC，同R15
  - CPSR，当前程序状态寄存器
  - SPSR，程序状态备份寄存器
  - F0-F7，浮点运算加速寄存器
  - S0-S31，单精度浮点寄存器
  - D0-D15，双精度浮点寄存器

名词解释

SPI: Serial peripheral interface

I2C: Inter integrated circuit

MMC: multimedia card

SD: Secure Digital Memory card

JTAG: Joint test action group

TMS: Test mode select

TDI, TDO, TCLK, nTRST

UEFI: Unified Extensible Firmware Interface 统一的可扩展固件接口

ELF: Executable and Linkable Format

在嵌入式调度中一般基于优先级

抢占式， 中断服务以后控制权不是回到原来被中断了的那个任务，二十高优先级任务得到CPU使用权

BootLoader

Boot Loader的阶段1通常包括以下步骤

1. 硬件设备初始化
    - 屏蔽所有中断
    - 设置CPU的速度和时钟频率
    - RAM初始化
    - 初始化LED
    - 关闭CPu内部指令/数据Cache
2. 为加载阶段2准备RAM空间
    - 除了阶段2可执行映像的大小外，还必须把堆栈空间也考虑进来
    - 必须确保所安排的地址范围的确是可读写的RAM空间
    - 内存区域有效性检测
3. 拷贝阶段2代码到RAM中
4. 设置堆栈指针sp
5. 跳转到阶段2的C语言入口点

Bootloader阶段2包含

1. 初始化本阶段要用到的硬件设备

   初始化至少一个串口，以便和终端用户进行I/O输出信息

   初始化计时器等
2. 检测系统的内存映射
3. 加载内核映像和根文件系统印象
    - 内核印象所占用的内存范围 MEM_START+0x8000
    - 根文件系统所占用的内存范围 MEM_START+0x100000
4. 设置内核的启动参数

5. 调用内核

Bootloader在跳转时, 下列条件要满足

1. CPU寄存器的设置
2. CPU模式

   必须禁止中断(IRQ和FIQ)

   CPU必须SVC模式
3. Cache 和MMU的设置

   MMU必须关闭, 指令Cache可以打开也可以关闭, 数据Cache必须关闭

嵌入式实时操作系统

嵌入式实时操作系统内核的基本功能:

- 实时多任务管理
- 中断与异常管理
- 共享资源互斥管理
- 多任务同步与互斥
- 内存管理
- 时钟定时器管理
- 电源管理

RTOS

中断机制:

外部中断: 一般是指系统外设发出的中断请求，外部中断大多是可以屏蔽的，程序可以根据具体需要，通过中断控制器来屏蔽这些中断请求

内部终端: 指因处理器自身的原因引起的异常事件，内部中断基本是不可屏蔽的中断

软件中断: 是一种特殊的中断，程序通过软件指令触发的，从而主动引起程序流程的变化

内核服务可以可抢占

优先级反转: 在优先级多任务系统中引入互斥方案，会导致任务优先级反转的问题：例如某时 低优先级的任务占有资源，然后又有高优先级的任务申请资源，但因为不能满足而不被挂起了，即低优先级任务阻塞了高优先级任务的运行。例如这时又有一个中优先级任务，那么它会把低优先级任务抢占

嵌入式实时操作系统内核的重要特性

- 实时性
- 可裁剪、可配置性
- 可靠性支持
- 应用编程接口支持
- 可移植性

内核的实时性能定量指标包括:

- 任务上下文切换时间

- 中断延迟时间
- 中断响应时间
- 中断恢复时间
- 任务响应时间

内核实时性能关键指标

- 最大中断禁止时间 反映内核对外界停止中断响应的最长时间
- 任务上下文切换时间

许多嵌入式操作系统不区分系统空间和用户空间, 如VxWorks, RTEMS

内核的裁剪方法: 模块级剪裁、函数级剪裁、代码级剪裁

一个最小的多任务嵌入式软件包括:

- Bootloader
- 具有任务管理及定时功能的基本内核
- 一个初始化任务

uC/OS

经裁剪最小可达2KB，最小数据RAM需求10KB

uC/OS不支持优先级反转

μC/OSII系统功能

序号	模块名称	功能	支持情况
1	任务管理和调度	任务调度方式	优先级抢占（不支持时间片轮转）
		内核抢占	否
		优先级数量	64
		任务数量	64（其中，8个系统任务）
2	同步和通信	同步	信号量、事件标志组
		通信	消息队列、邮箱
		优先级反转	不支持
		有限等待	支持
		递归申请	不支持
3	内存管理	MMU	不支持
		内核数据结构	静态分配
		用户内存使用	固定大小的内存分配
4	中断管理	中断服务程序	不支持ISR自动插桩 10

OS_CPU.H中定义了两个宏来开关中断 OS_ENTER_CRITICAL()，OS_EXIT_CRITICAL()

ucOS的多任务处理

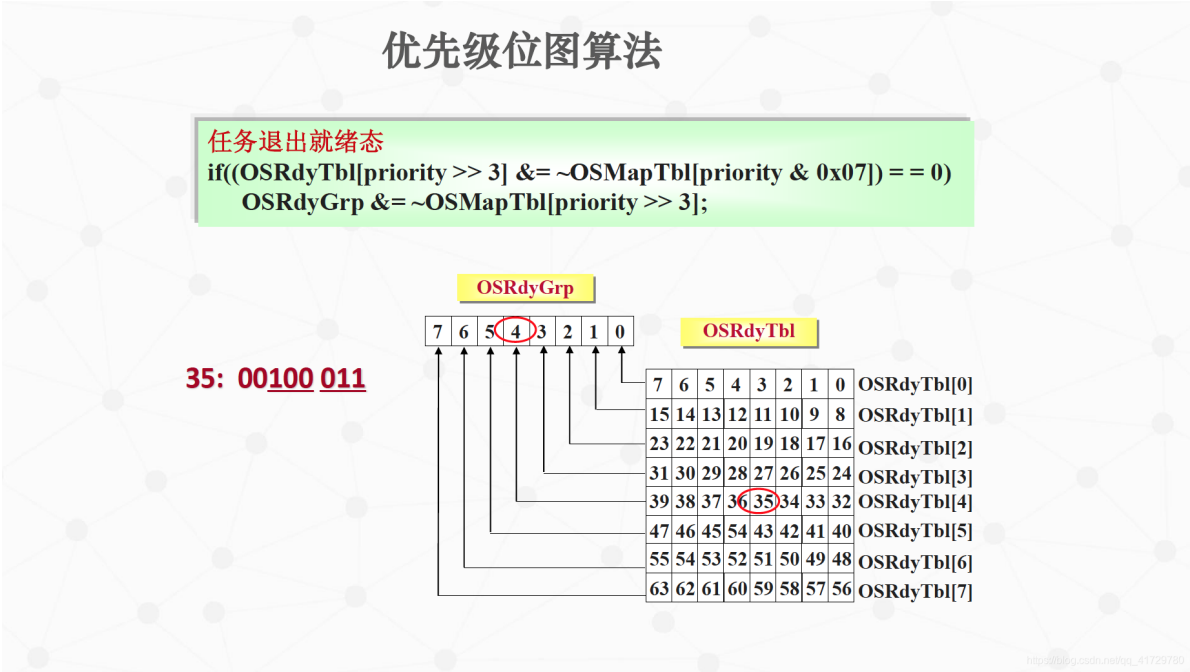
在内存中为每个任务创建一个虚拟的处理器(处理器部分的运行环境) 当需要运行某个任务时就把该任务的虚拟处理器复制到实际处理器中

ucOS的任务优先级

ucOS把任务的优先级分为64个优先级别，每一个级别都有一个数字来表示，数字0表示任务的优先级别最高，数字越大表示任务的优先级别越低

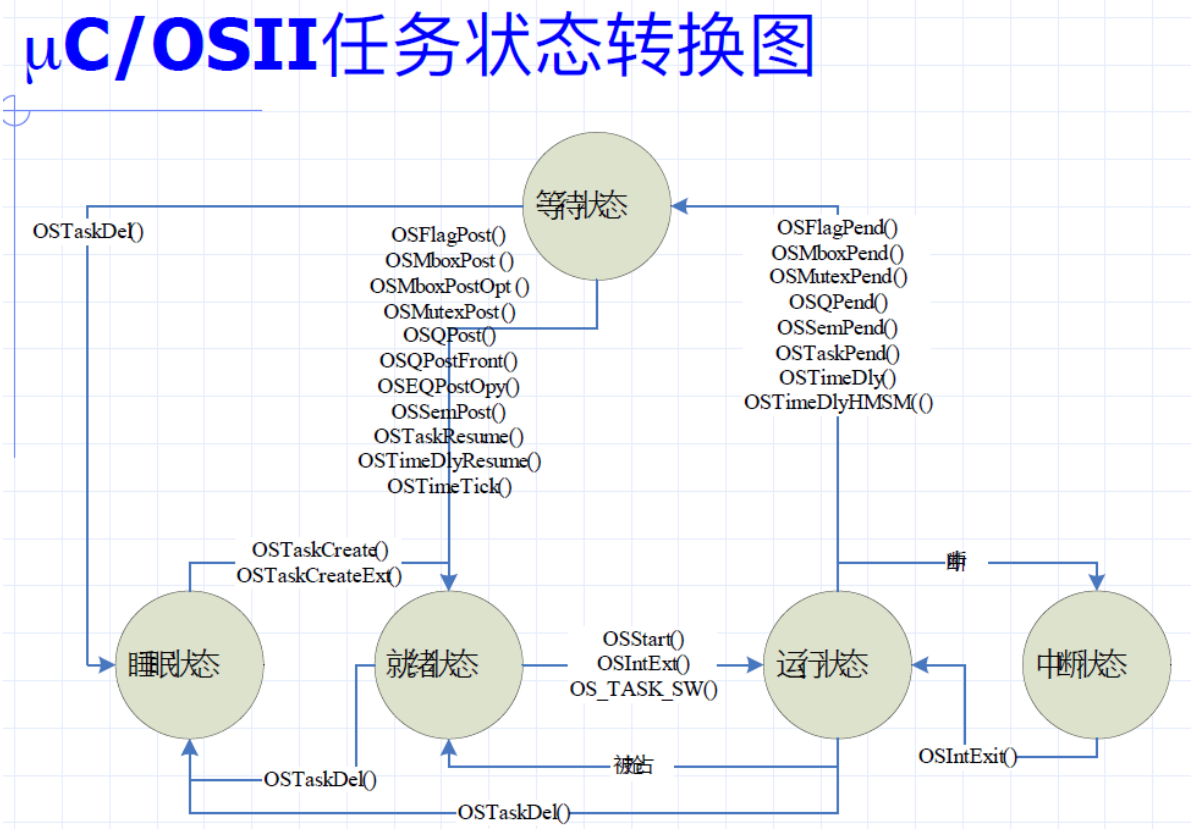
在文件OS_CFG.H中通过OS_LOWEST_PRIO来配置最低优先级。

系统总是把最低优先级OS_LOWEST_PRIO自动赋给空闲任务。如果还是使用了统计任务，系统会把优先级OS_LOWEST_PRIO-1自动赋给统计任务，因此用户任务可以使用的优先级别范围0到OS_LOWEST_PRIO-2



uCOS的从任务就绪表中获取优先级别最高的就绪任务可用以下代码

```
y = OSUnMapTal[OSRdyGrp];
x = OSUnMapTal[OSRdyTbl[y]];
prio=(y<<3) + x
```



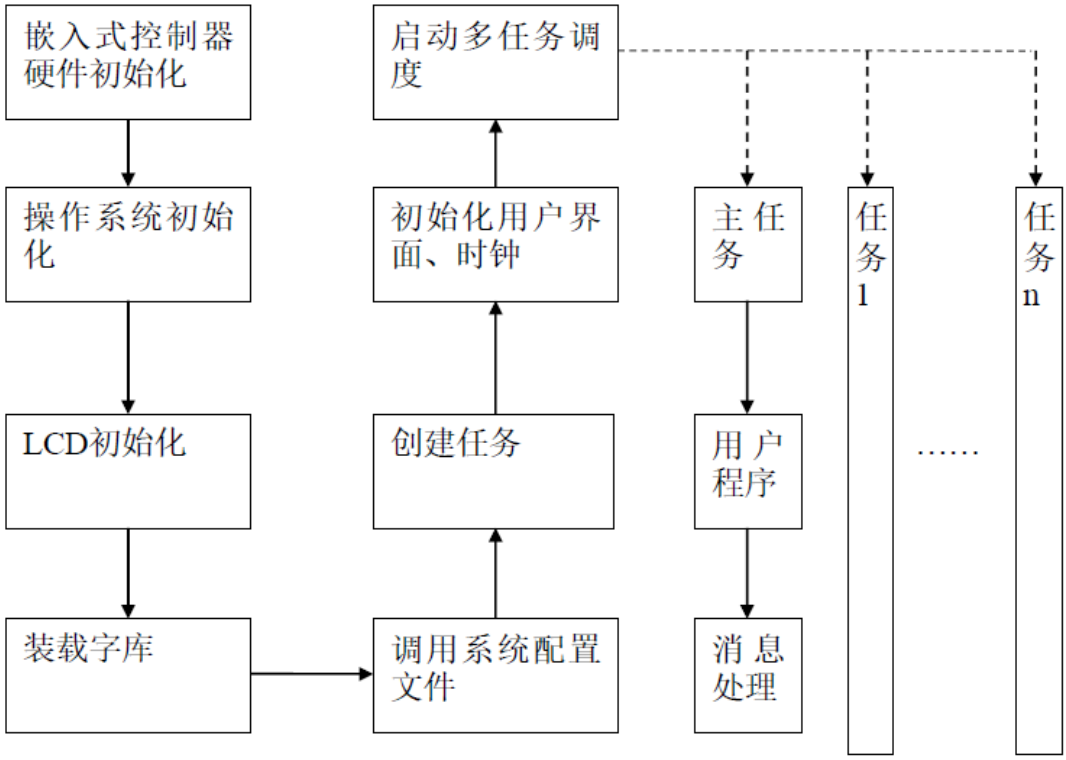
ucOS中断响应中，中断服务子程序运行结束之后，系统将会根据情况进行一次任务调度去运行优先级别最高的就绪任务，而并不是一定要接续运行被中断的任务

ucOS支持四种类型的中断

- 设备中断

- 外部中断
- 陷阱中断
- 现场控制量的中断

ucOS启动流程



ucOS使用事件控制块ECB的数据接口来描述信号量、邮箱、消息队列等事件

ucOS改进了ANSI C用来动态分配和释放内存的malloc()和free()函数使它们可以对大小固定的内存块进行操作

uCOS对内存进行了两级管理，即把一个大片连续的内存空间分成了若干个分区，每个分区又分成了若干个大小相等的内存块来管理，操作系统以分区为单位来管理动态内存，任务以内存块为单位来获得和释放动态内存

shell的主要功能: 提供环境， 解析指令

ARM-Linux内核

ARM-linux 32位地址形成4GB 虚拟地址空间, 3G以下是用户空间, 3G以上是内核空间

Linux进程的五种状态:

- TASK_RUNNING

- TASK_INTERRUPTIBLE
- TASK_UNINTERRUPTIBLE
- TASK_ZOMBIE
- TASK_STOPPED

ARM将中断控制器集成在CPU内部，由外设产生的中断请求都由芯片上的中断控制器汇总成一个IRQ中断请求

SPI

SPI通信双方是不平等的

SPI接口一般使用四条线: 串行时钟线SCK, 主输入/从输出数据线MISO, 主输出/从输入数据线MOSI和从器件选择线SS

设备驱动程序

BSP(Board Support Package)

针对具体目标机平台所编写的与硬件结构相关的代码部分

主要完成系统上加电后初始化、各类控制芯片初始化等工作

设备驱动程序

设备驱动程序是直接控制设备操作的那部分程序，是设备上层的一个软件接口，功能是对I/O进行操作

与硬件结构相关的部分，包含在BSP中

通用部分，可以在其他部分中实现

设备驱动程序的主要功能包括

- 对设备进行初始化
- 打开设备操作
- 关闭设备操作
- 从设备上接受数据-读操作
- 把数据从主机上发送给设备-写操作
- 对设备进行控制操作

Linux文件系统有两条独立控制设备驱动

- 通过设备驱动的接口
- 通过文件管理器结构

每个GPIO端口一般包含8个引脚，例如PA端口为PA0-PA7，可以控制I/O接口作为输入或者输出

GPIO接口一般至少会有两个寄存器，控制寄存器和数据寄存器

文件系统 & 存储设备

flash or 磁盘文件系统 fat, ext, NTFS

flash文件系统 yaffs, jaffs,NTFS

ROM文件系统 romfs

RAM文件系统(内存文件系统) ramfs

JFFS主要针对NOR Flash

YFFS主要针对NAND Flash

NOR flash 支持随机读和随机写 字节

NAND flash 只能读取一个页和写一个页， 写页之前需要先擦除页

NOR flash的读取速率快于NAND, NAND的写入和擦除速率快于NOR, 且NAND的寿命高于NOR