



个人资料



sunheshan



访问：96127次

积分：1551

等级：BLOG > 4

排名：千里之外

原创：48篇

转载：38篇

译文：1篇

评论：7条

文章搜索

文章分类

编程基础 (11)

电子器件 (3)

基础原理 (11)

工具 (3)

嵌入式系统 (7)

linux (18)

iar (10)

上位机 (0)

C语言 (6)

uc/OS (8)

LWIP (2)

文章存档

2015年12月 (1)

2015年09月 (24)

2015年05月 (2)

2015年04月 (1)

2015年03月 (1)

展开

阅读排行

ucos_ii定时器详解 (4130)

PCB上电源走线注意 (4062)

菜鸟详解iar的icf文件 (链 (3471)

STM32 FLASH模拟 EEP (3057)

IAR中常用的 #pragma 命 (3018)

异步赠书：9月重磅新书升级，本本经典

SDCC 2017之区块链技术实战线上峰会

程序员9月书讯

每周荐书：ES6、虚拟现实、物联网（评论送书）

uC/OS-ii在M3中的任务切换与任务堆栈

标签：[任务堆栈](#) [uc-os任务切换](#) [uc-os](#)

2015-09-18 21:49 1181人阅读 评论(0)

分类：

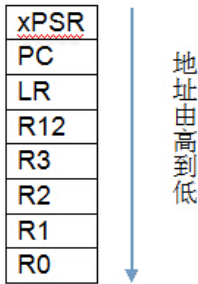
[uc/OS \(7 \)](#)

版权声明：本文为博主原创文章，未经博主允许不得转载。

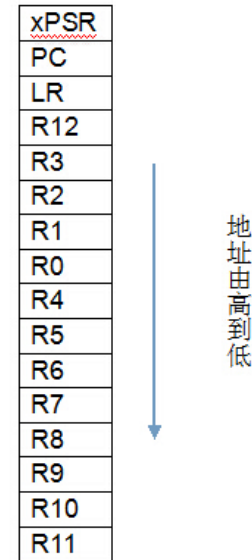
[目录\(?\)](#) [\[+\]](#)

1 uc/os ii在M3中的堆栈结构

1.1 M3入账序列



1.2 加上手工入栈序列



2 PendSV在Cortex-M3中的应用

Systick为嵌入到内核中，优先级比一般中断优先级高。若在一般中断的ISR执行过程中，发生了Systick异常，则Systick会抢占该ISR。若此时Systick做上下文切换，在M3中将触发用法fault（在中断活跃时尝试切入线程模式）。即使在别的内核体系下不发生硬fault，ISR也会被延迟，这对于任一讲究实时性的系统是不能接受的。

- orcad中的警告Warning [(2611)
- 宏定义的作用范围 (2551)
- LWIP的调试方式 (2542)
- cortex-m3启动代码详解 (2491)
- IAR中查看程序运行时间 (2235)

评论排行

考考你的C语言	(3)
cortex-m3启动代码详解	(2)
菜鸟详解iar的icf文件（链	(2)
内联函数与宏	(0)
24AA512/24LC512/24FC	(0)
UART与SPI的区别	(0)
深入辨析 FLASH与EEP	(0)
STM32 FLASH模拟 EEP	(0)
IAR中设置和编译信息解	(0)
JTAG与JLINK、H-JTAG	(0)

推荐文章

- * CSDN日报20170828——《4个方法快速打造你的阅读清单》
- * Android检查更新下载安装
- * 动手打造史上最简单的Recycleview 侧滑菜单
- * TCP网络通讯如何解决分包包问题
- * SDCC 2017之区块链技术实战线上峰会
- * 快速集成一个视频直播功能

最新评论

好手机排行榜



群控系统



初学者efm32上移植uC/OSII
sdy19930904: 您是用的keil还是
Simplicity Studio 编译器，
EFM32的固件库从哪里下载的，
万分...

cortex-m3启动代码详解
 忧郁和弦: 分析的非常好，期待后续的分析！

所以Systick只是在其服务程序中触发一个PendSV中断（事先将PendSV中断优先级设置的很低）。Systick返回后将继续执行被抢占的中断ISR。执行完后，程序跳转到PendSV服务程序中执行任务切换。

3 M3中堆栈的切换

3.1 在中断返回时，通过修改LR中的EXC_RETURN

3.1.1 EXC RETURN

在出入ISR的时候，LR的值将得到重新的诠释，这种特殊的值称为“EXC_RETURN”，在异常进入时由系统计算并赋给LR，并在异常返回时使用它。EXC_RETURN的二进制值除了最低4位外全为1，而其最低4位则有另外的含义。

表1 EXC RETURN位段详解

位段	含义
[31:4]	EXC_RETURN的标识：必须全为1
3	0=返回后进入Handler模式 1=返回后进入线程模式
2	0=从主堆栈中做出栈操作，返回后使用MSP， 1=从进程堆栈中做出栈操作，返回后使用PSP
1	保留，必须为0
0	0=返回ARM状态。 1=返回Thumb状态。 在CM3中必须为1

表2 合法的EXC_RETURN值及功能

EXC_RETURN	功能
0xFFFF_FFF1	返回handler模式
0xFFFF_FFF9	返回线程模式，并使用主堆栈(SP=MSP)
0xFFFF_FFFD	返回线程模式，并使用线程堆栈(SP=PSP)

3.2 异常返回和异常返回序列

M3 提供的异常返回指令

BX <reg>	当LR存储了EXC_RETURN时，使用BX LR即可返回
POP {PC}和 POP {...,PC}	在服务例程中，LR的值常常会被压入栈。此时即可使用POP指令把LR存储的EXC_RETURN往PC里弹，从而启动处理器的中断返回序列
LDR与LDM	把PC作为目的寄存器，亦可启动中断返回序列

出栈：先前压入栈中的寄存器在这里恢复。内部的出栈顺序与入栈时的相对应，堆栈指针的值也改回先前的值。

只要ISR没有更改过CONTROL[1]，就依然使用发生本次异常的瞬间正在使用的SP指针来执行出栈操作。

3.3 修改EXC_RETURN，进行模式和堆栈的选择。

在异常ISR中，处理器处于特权模式，可以访问所有存储器（除MPU规定）。通过修改LR的值来达到不同模式和堆栈的切换。

4、再论M3双堆栈机制

已经知道M3堆栈分为MSP和PSP，CONTROL[1]决定如何选择。当CONTROL[1]为0时，只使用MSP，此时用户程序和异常handler共享一个堆栈。

当CONTROL[1]为1时，线程模式将不再使用MSP，而改用PSP。这样做的好处在OS内核中防止用户程序的堆栈破坏OS的堆栈。在这种情况下进入异常的自动压栈使用的是进程堆栈，进入异常handler后才自动改为MSP，退出异常时切换回PSP，并且从进程堆栈上弹出数据。

1 在中断
在出入ISR的时候，LR的值将得到重新的诠释，这种特殊的值称为“EXC_RETURN”。在异常进入时由系统计算并赋给LR，并在异常返回时使用它。

2 在函数跳转
当呼叫一个子程序时，由R14存储返回地址。

顶 0 踩 0

上一篇 临界区管理
下一篇 一个类型转换的讨论

相关文章推荐

- UC/OS-II的任务切换
- 携程机票大数据基础平台架构演进-- 许鹏
- uC/OS-II的任务切换总结
- Python可以这样学--董付国
- uC/OS II 多任务，堆栈检查，DSP定时器卷积滤...
- 一步一步学Spring Boot
- uC/OS-II任务调度中判断最高优先级
- 深入浅出C++程序设计

- uc/os-ii任务调度(二)
- Android Material Design 新控件
- uC/OS-II学习笔记--任务
- 机器学习需要用到数学知识
- UC/OS-ii任务优先级管理与查找算法
- 操作系统之进程和线程--uC/os-II实例--建立任务
- uc/os-ii任务调度(一)
- uc/os-ii任务延时

 个人社保查询

 OA办公系统

 主播直播间

 订单管理系统

 绩效考核系统



查看评论

暂无评论

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

好手机排行榜



群控系统



公 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

江 服 webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |