

跟涛哥一起学嵌入式 05：一道程序改错题，测出你的嵌入式功底

文档说明	作者	日期
来自微信公众号：宅学部落(armLinuxfun)	wit	2018.7.6
嵌入式视频教程淘宝店： https://wanglitao.taobao.com/		
联系微信：brotau(宅学部落)		

大家好，欢迎阅读《跟涛哥一起学嵌入式》第05集，我们今天讨论一下中断的基本概念。

中断，是嵌入式开发中经常使用的一个功能，也是嵌入式工程师必须要掌握的一个概念：CPU和外设通信时，一般都采用中断的形式异步通信，可以大大提高CPU资源的利用率。而你对中断的理解，到底有多少呢？不要急，一道程序改错题，就可以测出你的嵌入式系统功底。

比如，我们在嵌入式ARM裸机平台上，要实现一个MP3播放器，要求实现如下功能：当按键按下时，可以播放、暂停、播放下一首、上一首。为此，我们设计一个按键中断服务程序，当有按键发生时，我们去读取按键的值，然后再根据按键值去执行不同的操作，设计的按键中断函数如下：

```
int keyboard_isr(int irq_num)
{
    char *buf =(char *)malloc(512);
    int key_value = 0,

    key_value = keyboard_scan();
    if(key_value == 1)
    {
        mp3_decode(buf,"xx.mp3");
        sleep(10);
        mp3_play(buf);//play
    }
    else if(key_value == 2)
        mp3_pause(buf);//pause
    else if(key_value == 3)
        mp3_next(buf);//next song
    else if(key_value == 4)
        mp3_prev(buf);//prev song
    else
    {
        printf("UND key !");
        return -1;
    }
    return 0;
}
```

这个中断函数差不多有8处错误或设计不合理的地方，先不用急着往下翻，你可以自己先思考下，看自己能找出来几处。

(此处应有掌声，暂停2分钟)...

好，我们接下来继续分析，看看这个中断函数到底有哪些错误。

错误1：中断关键字_irq

中断函数没有使用关键字标记。C标准中没有定义中断函数，对于普通的函数调用，编译器会帮我们自动实现函数的调用栈及出栈入栈管理，而对于中断函数，我们知道，中断是可以随时发生的，编译器无法确定你中断函数在哪里发生，所以就无法帮你实现调用栈。那怎么办？难道我们程序员自己实现？没关系，各大编译器厂商一般会通过增加一些关键字，来帮助程序员实现中断的现场保护代码。因此，我们可以使用**interrupt**或**irq**关键字声明即可

```
__irq int keyboard_isr(int irq_num);
```

错误2：malloc/free

对于MP3解码的数据，我们使用malloc函数申请了一个动态堆内存来存放，但是函数退出后没有及时释放，造成了内存泄露。关于函数的调用栈、调用传参过程、程序的内存分配，如果不是很明白，可以参考《C语言嵌入式Linux高级编程》第4期：堆栈管理。

错误3：C标准库

嵌入式裸机开发中，一般很少使用C标准库的。很多嵌入式编译厂商、IDE开发商并没有完全实现C标准库，或者只是实现了一部分。所以在一般在嵌入式系统中，我们可以使用一个全局数组来代替malloc申请的动态内存。

错误4：中断返回值

中断函数由于没有独立的函数栈，是不能有返回值的。想一想：中断函数在哪里被调用、何时被调用，我们全都不知道，中断函数的返回值应该传给谁？它会破坏当前被中断函数的函数栈。

错误5：中断传参

同上，中断函数也是不能传参的，传给谁？谁调用了它？都是未知的。如果你给中断函数传参，也会破坏当前被中断函数的函数栈。所以中断函数正确的声明形式应该为：

```
__irq void keyboard_isr(void);
```

错误6：延时

避免耗时、延时、可重入函数，比如sleep、printf等。

错误7：短小精悍、快速反应

中断函数要求快速响应，短小精悍。要速战速决，快速撤离中断现场，然后等待下一次中断。如果你在中断程序中，要做大量工作：播放、解码、延时，十分不合理。这些耗时的工作应该放到中断外去做。如Linux中的中断处理，就放到了中断下半部去做。

好了，以上都是我们在编写中断函数时必须掌握的一些基础知识和技能，接下来就卖个关子：想一想，上面这个函数的设计，还有什么不合理之处？思考的越多，你就得到的越多。老司机要下车了，先带你到这里，剩下的就靠你了~

本文根据《C语言嵌入式Linux高级编程》视频教程改编。《跟涛哥一起学嵌入式》会持续跟大家分享嵌入式相关技术、学习方法、学习路线、求职面试等。

专注嵌入式、Linux精品教程：<https://wanglitao.taobao.com/>

嵌入式技术教程博客：<http://zhaixue.cc/>

联系 QQ：3284757626

嵌入式技术交流QQ群：475504428

微信公众号：宅学部落(armlinuxfun)

