

冀博

tiger-john不懈学习嵌入式ing-- 尘事中一个迷途，小小硕士

目录视图

摘要视图

RSS 订阅

个人资料



冀博



访问： 562732次

积分： 7459分

排名： 第649名

原创： 176篇 转载： 40篇

译文： 0篇 评论： 571条

文章分类

嵌入式开发 (60)

Linux内核开发 (23)

linux应用层编程 (31)

C的往事 (39)

网络编程 (6)

网络协议 (11)

set up environment (24)

内核编译bug (1)

Makefile (1)

shell (5)

python (1)

小白学Linux的那些往事 (7)

数据结构和算法 (1)

Javascript (1)

Flex (1)

一句话心情 (5)

杂项 (4)

John看经济 (3)

英语学习 (4)

笔试 (2)

博客专栏



Linux网络编程

文章： 8篇

阅读： 31213

有奖征资源，博文分享有内涵 4月推荐博文汇总 第六届中国云计算大会（大数据读书汇）

container_of分析

分类： Linux内核开发 2012-12-15 19:23 3653人阅读 评论(5) 收藏 举报

目录(?) [+]

1.container_of宏

1> Container_of在Linux内核中是一个常用的宏，用于从包含在某个结构中的指针获得结构本身的指针，通俗地讲就是通过结构体变量中某个成员的首地址进而获得整个结构体变量的首地址。

2>接口：

container_of(ptr, type, member)

ptr:表示结构体中member的地址

type:表示结构体类型

member:表示结构体中的成员

通过ptr的地址可以返回结构体的首地址

3> container_of的实现：

#define container_of(ptr, type, member) ({ \

const typeof(((type *)0)->member) *__mptr = (ptr); \

(type *) ((char *)__mptr - offsetof(type,member));})

其实它的语法很简单，只是一些指针的灵活应用，它分两步：

第一步，首先定义一个临时的数据类型（通过typeof(((type *)0)->member)获得）与ptr相同的指针变量__mptr，然后用它来保存ptr的值。

说明：typeof是GNU C对标准C的扩展，它的作用是根据变量获取变量的类型《typeof关键字在linux 内核中很常见》

第二步，用(char *)__mptr减去member在结构体中的偏移量，得到的值就是整个结构体变量的首地址（整个宏的返回值就是这个首地址）。

关于offsetof的用法可参见offsetof宏的使用。

2. 举例来说明container_of的使用

1>正确示例：

#include <stdio.h>

#define offsetof(TYPE, MEMBER) ((size_t) &((TYPE *)0)->MEMBER)

#define container_of(ptr, type, member) ({ \

const typeof(((type *)0)->member) *__mptr = (ptr); \

(type *) ((char *)__mptr - offsetof(type,member));})

struct test_struct {

int num;

char ch;

float fl;

};

int main(void)

{

struct test_struct *test_struct;

http://blog.csdn.net/tigerjibo/article/details/8299589



Tiger-John的嵌入式开发专栏

文章：20篇

阅读：62573

文章搜索

文章存档

2013年11月 (1)

2013年10月 (6)

2013年09月 (5)

2013年08月 (3)

2013年07月 (9)

展开

评论排行

C语言中volatile关键字的

(72)

一场流水线带来的灾难

(37)

基于uCOS-II的UDP网络:

(21)

Linux中断学习之小试牛:

(20)

小白学Linux之内核模块编

(18)

我的UC/OS, 我做主

(16)

嵌入式C语言编程小知识.

(15)

Linux下的串口编程(二)

(15)

Linux内核链表的研究与分

(14)

Linux内核哈希表分析与分

(13)

阅读排行

C语言中volatile关键字的

(39994)

C函数之memcpy()函数用

(31558)

Linux时间函数之gettime

(22287)

C语言中全局变量、局部

(14903)

kmalloc/kfree,vmalloc/vf

(11666)

ICMP报文分析

(10721)

Linux中断学习之小试牛:

(10398)

C语言中函数strcpy, strnc

(10277)

ARM之汇编学习---如何编

(9273)

基于Linux下的TCP编程

(9205)

最新评论

小白学Linux之内核模块编程

xcp2009: 实际底层结构设计比较复杂

John 看股市

john_han: 中国股市没有内幕消息赚钱的概率有多大?

Linux内核哈希表分析与应用

giant1st: 还是没有解释为什么要用二级指针的问题啊? 你说的那两点原因, 用一级指针还是可以的啊? 只是解释了p...

基于Linux下的TCP编程

dongwu123: 经典, , , ,

offset宏的讲解

草原牧龙: 感谢博主的分析

offset宏的讲解

zyb228107: 分析的太好了, 十分感谢

```
struct test_struct init_struct = {12, 'a', 12.3};
char *ptr_ch = &init_struct.ch;
test_struct = container_of(ptr_ch, struct test_struct, ch);
printf("test_struct->num =%d\n", test_struct->num);
printf("test_struct->ch =%c\n", test_struct->ch);
printf("test_struct->ch =%f\n", test_struct->f1);
return 0;
}
```

执行结果:

jibo@jibo-VirtualBox: ~/cv_work/work/list/container_of \$./main
test_struct->num =12
test_struct->ch =a
test_struct->ch =12.300000

2>错误示例:

```
#include <stdio.h>

#define offsetof(TYPE, MEMBER) ((size_t) &((TYPE *)0)->MEMBER)
#define container_of(ptr, type, member) ({
    const typeof( ((type *)0)->member ) *__mptr =
(ptr);
    (type *) ( (char *)__mptr - offsetof(type, member)
);})

struct test_struct {
    int num;
    char ch;
    float f1;
};

int main(void)
{
    struct test_struct *test_struct;
    char real_ch = 'A';
    char *ptr_ch = &real_ch;
    test_struct = container_of(ptr_ch, struct test_struct, ch);
    printf("test_struct->num =%d\n", test_struct->num);
    printf("test_struct->ch =%c\n", test_struct->ch);
    printf("test_struct->ch =%f\n", test_struct->f1);
    return 0;
}
```

执行结果为:

jibo@jibo-VirtualBox: ~/cv_work/work/list/container_of1 \$./main
test_struct->num =0
test_struct->ch =A
test_struct->ch =0.000000

注意, 由于这里并没有使用一个具体的结构体变量, 所以成员num和f1的值是不确定的。

更多 2

上一篇 offset宏的讲解
下一篇 Linux内核链表的研究与应用

顶

3

踩

1

小白学Linux之内核模块编程
森森冲冲: 太感谢了，正好需要！
C语言中volatile关键字的作用
qquan2: 学习了
DNS报文格式（RFC1035）
冀博: @u011774512:可以转载，但是不要叫哥老前辈。哥正当青春时
DNS报文格式（RFC1035）
爱神CODE: 您好老前辈，请问这篇您的译文可以转载吗？

csdn 的牛人

王一刀(哲学，思想)
masefee c/c
July (对算法有着无穷的乐趣)
陈皓专栏 空谷幽兰，心如皓月
gnuhpcc(很给力的IT达人)
夏天的树人（致力于flash和flex技术）
无为和尚（致力于Linux内核技术，有很高的造诣）
雨松（致力于Iphone开发，objective-c）

推荐文章

Linux达人

牛涛
吴刚
陈继峰
李则良
许振文

学习网站

jquery学习网址
http://kb.cnblogs.com/zt/jquery/
XML学习网址
http://www.w3school.com.cn/xasp

主题推荐	linux内核	virtualbox	指针	扩展	数据
猜你在找					
platform_driver与file_operations两种方法开发					put_user、get_user、copy_to_user、
System.map解析					uboot新版本源码下载
getevent/sendevent 使用说明					Service与Android系统实现（1）— 应用程序里的
ARM中的各种寄存器名称 LR_PC_SP_CPSR_SPSR					wxpython编程之 事件处理原理
start-stop-daemon example					MTD NANDFLASH驱动相关知识介绍

免费学习IT4个月,月薪12000

中国[官方授权]IT培训与就业示范基地,学成后名企直接招聘,月薪12000起!

查看评论

4楼 枫-落 2014-03-28 18:58发表



分析的非常的好，很细致，之前一直看这段代码，总是看不太明白，前辈几句话就解释的很清楚了，十分感谢！

3楼 marshalDyoung 2014-03-16 13:53发表



讲的很透彻，看的很明白，谢谢！！
printf("test_struct->ch =%fn",test_struct->f1);应该是
printf("test_struct->f1 =%fn",test_struct->f1);吧？

2楼 fantangshan 2014-02-13 01:04发表



聪明人真正理解的人才知道要举个简单明了的例子来说明概念，百度到其它的,废话一堆,找个正确用法都没有
真心好懂,谢谢楼主

1楼 sunliezhi 2013-11-12 22:38发表



兄台真是高明，举例都如此合我的胃口，轻松看懂。

Re: 冀博 2013-11-13 21:37发表
 回复sunliezhi: 亲，喜欢的帮我投出宝贵的一票：<http://blog.51cto.com/contest2013/7718010>。在此先谢过

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

核心技术类目

全部主题

Java

VPN

Android

iOS

ERP

IE10

Eclipse

CRM

JavaScript

Ubuntu

NFC

WAP

jQuery

数据库

BI

HTML5

Spring

Apache

Hadoop

.NET

API

HTML

SDK

IIS

Fedora

XML

LBS

Unity

Splashtop

UML

components

Windows Mobile

Rails

QEMU

KDE

Cassandra

CloudStack

FTC

coremail

OPhone

CouchBase

云计算

iOS6

Rackspace

Web App

SpringSide

Maemo

Compuware

大数据

aptech

Perl

Tornado

Ruby

Hibernate

ThinkPHP

Spark

HBase

Pure

Solr

Angular

Cloud Foundry

Redis

Scala

Django

Bootstrap