

基于 eCos 系统的 SPCE3200 中 SD 卡驱动程序的开发

王玉叶, 宋国富

(安徽职业技术学院 电气工程系, 安徽 合肥 230051)

摘要: 采用 eCos 系统对 SPCE3200 的 SD 卡驱动程序进行开发。围绕编写设备表入口、实现设备接口函数、设计 CDL 文件、向组件库添加驱动程序包等几个方面, 对 SPCE3200 的 SD 卡驱动程序在 eCos 系统中的实现进行了详细叙述。

关键词: 嵌入式系统; 设备驱动程序; 组件定义语言

中图分类号: TP316.89 **文献标识码:** A

Design of SD memory card driver based on eCos system in SPCE3200

WANG Yu Ye, SONG Guo Fu

(Department of Electrical Engineering, Anhui Vocational and Technical College, Hefei 230051, China)

Abstract: This design is about the development of the SD card driver of SPCE3200 by eCos system. In this article, SPCE3200 SD card driver realization in eCos system is given with very detail, the content being recounted includes the start address of the driver, realization of the equipment interface function, designing the CDL document, loading the program package to the components library.

Key words: embedded systems; equipment driver; component definition language

设备驱动程序是介于硬件和嵌入式系统 eCos 内核之间的软件接口, 是一种底层的、专用于某一硬件的软件组件。在 eCos 系统^[1]中, 设备驱动程序以包 (Package) 的形式存在, 当需要使用该设备时, 通过 eCos 环境配置工具把该设备驱动程序包加载到内核中, 编译后生成一个库文件, 用户使用时, 只需要打开该设备, 直接调用接口 API 函数即可。设备驱动程序的作用如图 1 所示^[2]。

在 eCos 系统中一般把设备分为 4 类: 字符设备、块设备、网络接口、其他设备。

SPCE3200 是凌阳公司推出的一款 32 位支持嵌入式系统处理器, 其内部具有 SD 卡控制器, 属于块设备类^[3]。对 SPCE3200 内部的 SD 卡控制器必须编写驱动程序, 以便完成 SD 卡的读写操作。下面将介绍具体实现过程, 这里规定 SD 卡使用 “/dev/sd” 作为其名称。

1 建立 SD 卡驱动程序目录结构

在编写驱动程序前, 首先需要为 SD 卡驱动建立其目录结构。eCos 系统要求所有驱动程序均保存在 /packages/devs 目录下的对应子目录内, 按照图 2 所示在 eCos/packages/devs/sd/Spce3200 目录下分别建立名为 “cdl”、“include”、“current” 3 个目录, 分别用于存放 SD 卡驱动所需的 cdl 文件、驱动程序头文件和源程序文件。

2 建立 SD 卡驱动程序文件

在 cdl 目录内, 首先创建一个叫做 “sd_spce3200.cdl” 的文件。然后, 在 src 目录内建立一个名为 “sd_spce3200.c” 驱动程序的源程序, 并将名为 “sd_card_driver_spce3200.c” 的文件复制到 src 目录内, 该文件是凌阳科技提供的针对 SPCE3200 内部 SD 卡控制器的基本操作程序。

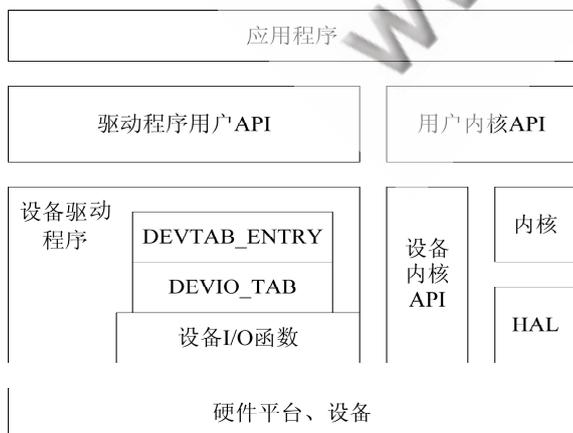


图 1 设备驱动程序的作用

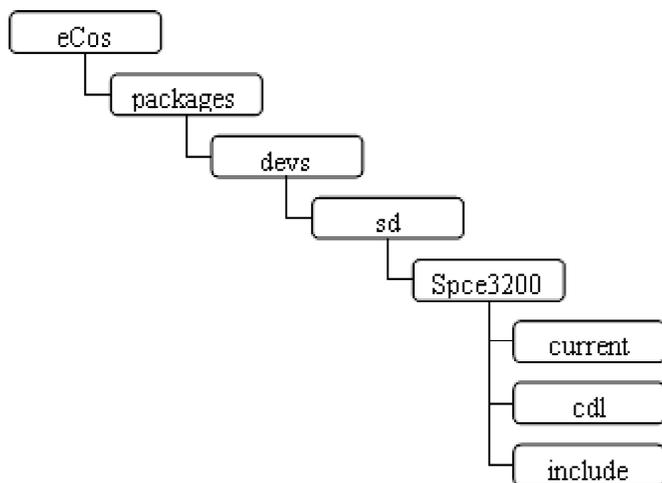


图2 SD卡驱动的目录结构

3 设计 SD 卡驱动的 CDL 文件

组件定义语言 CDL (Component Definition Language) 是 eCos 组件框架的一个关键部分。eCos 中所有的包都必须具有至少一个 CDL 脚本对其进行描述^[4]。CDL 脚本包含了该包中所有配置选项的详细信息，并提供了如何对该包进行编译的信息^[5]。这里，在 CDL 文件的模板框架中添加关于 SD 卡配置项的代码如下：

```

cdl_package CYGPKG_DEVS_SD_SPCE3200 {
display "SD driver for SPCE3200"
include_dir cyg/io
compile sd_card_driver_spce3200.c
compile -library=libextras.a sd_spce3200.c
description "SD driver for SCORE SPCE3200"
cdl_component CYGPKG_DEVS_SD_SPCE3200_OPTIONS {
display "CompileOptions"
flavor rnone
no_define
cdl_option CYGDAT_DEVS_SD_SPCE3200_NAME {
display "Device name for theSDDriver"
flavor data
default_value{"\\dev/sd\\"}
description " This option specifies the name of the sd-card
device"
}
}
}

```

4 编写 SD 卡设备表入口

配置好 CDL 文件后，便可编写设备 I/O 函数表宏以及设备表入口宏，分别如下：

```

BLOCK_DEVIO_TABLE(sd_spce3200_handlers, // I/O 函
数表的标识
sd_WriteSector, //SD 卡写操作函数
sd_ReadSector, //SD 卡读操作函数

```

```

NULL, // 选择函数，SD 卡驱动不需要支持
sd_get_config, // 读 SD 卡配置状态函数
sd_set_config //SD 卡配置函数
);
BLOCK_DEVTAB_ENTRY(sd_spce3200_device,
//SD 卡驱动入口标识
CYGDAT_DEVS_SD_SPCE3200_NAME, //SD 卡驱动的名称，在 cdl 文件里定义
NULL, //SD 卡驱动不依赖于其他设备
&sd_spce3200_handlers, //SD 卡设备 I/O 函数表
标识
&SD_Initial, //SD 卡初始化函数
0, //SD 卡查找函数
0 //SD 卡驱动的私有指针
);

```

5 实现 SD 卡设备接口函数

接下来需要实现上面提到的几个接口函数，这些函数是上层访问硬件设备的唯一通道。在凌阳科技提供的 sd_card_driver_spce3200.c 文件中包含了 SD 卡初始化、读写操作等函数^[6]，它们分别为：SD 卡初始化函数 int __SDDrv_SDDrv_Initial_(void)、SD 卡扇区写操作函数 int __SDDrv_SDDrv_WriteSector_(unsigned int block, unsigned int blocknum, unsigned char *outaddr)、SD 卡扇区读操作函数 int __SDDrv_SDDrv_ReadSector_(unsigned int block, unsigned int blocknum, unsigned char *buf)。

但这些函数都不是基于操作系统的，而是 SD 卡的基本操作函数。需要将它们封装成 eCos 操作系统下的驱动程序。这就需要用到凌阳科技提供的 SD 卡设备驱动接口函数，包括 SD 卡设备批量写操作函数 sd_WriteSector()、SD 卡设备批量读操作函数 sd_ReadSector()、SD 卡读设备配置状态函数 sd_get_config()、SD 卡设备设置函数 sd_set_config()^[7]。

完整的 SD 设备驱动源程序文件的代码如下：

```

//----- 文件名：sd_spce3200.c-----//
//----- 包含相关头文件 -----//
#include <pkgconf/hal.h>
#include <pkgconf/devs_sd_spce3200.h>
#include <cyg/infra/cyg_type.h>
#include <cyg/infra/cyg_ass.h>
#include <cyg/hal/hal_io.h>
#include <cyg/hal/hal_if.h>
#include <cyg/io/disk.h>
#include <cyg/hal/SPCE3200_Register.h>
#include <cyg/hal/SPCE3200_Constant.h>
#include <cyg/error/codes.h>
#include <cyg/io/devtab.h>
#include <cyg/io/disk.h>
//----- 驱动函数实现 -----//

```

```

static bool sd_init(struct cyg_devtab_entry* tab)
{
    bool InitResult = true;
    if(__SDDrv_SDDrv_Initial()==0)
        InitResult=true;
    else
        InitResult=false;
    return(InitResult);
}

static int sd_WriteSector(void* disk, const void* buf_arg,
cyg_uint32 *blocks, cyg_uint32 first_block)
{
    cyg_intret=ENOERR;
    ret=__SDDrv_SDDrv_WriteSector__(first_block,*blocks,
(cyg_uint8 *)buf_arg); (1)
    return ret;
}

static int sd_ReadSector(void* disk, void* buf_arg,
cyg_uint32 *blocks, cyg_uint32 first_block)
{
    cyg_int32 ret = ENOERR;
    ret = __SDDrv_SDDrv_ReadSector__(first_block,*blocks,
(cyg_uint8*)buf_arg); (1)
    return ret;
}

static Cyg_ErrNo sd_get_config(disk_channel* chan,
cyg_uint32 key, const void* buf, cyg_uint32* len)
{
    return -EINVAL;
}

static Cyg_ErrNo sd_set_config(disk_channel* chan,
cyg_uint32 key, const void* buf, cyg_uint32* len)
{
    return -EINVAL;
}
//----- 设备 I/O 函数表 -----//
(略, 见前面)
//----- 设备表入口 -----//
(略, 见前面)

```

6 向 eCos 数据库中添加 SD 卡驱动程序组件包

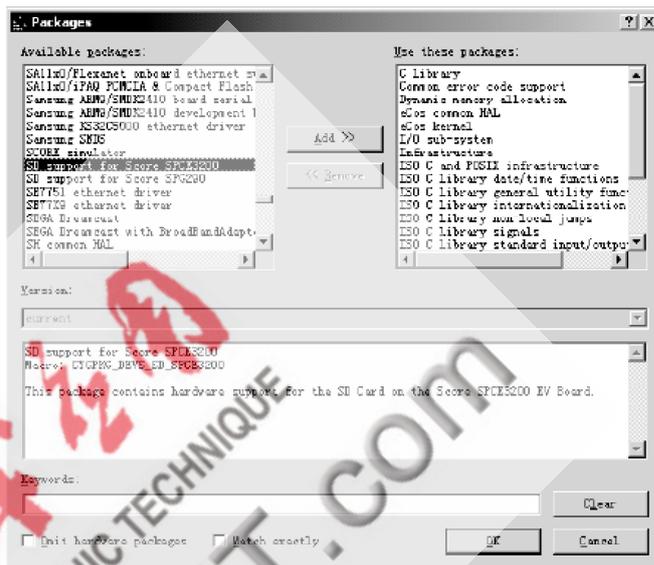
在 ecos.db 中的任意位置加如下程序段：

```

package CYGPKG_DEVS_SD_SPCE3200 {
    alias["SD support for Score SPCE3200"]
    directory devs/sd/score/spce3200
    script sd_spce3200.cdl
    description "
This package contains hardware support for theSDCard
on the Score SPCE3200 EV Board."
}

```

“devs/sd/score/spce3200” 指示了 SD 卡设备驱动的路径，该路径是相对于 /packages 目录的^[8]。此时打开 eCos 配置工具添加包时，就可以看见 SD 卡驱动程序组件包已经出现在 eCos 数据库中，如图 3 所示。点击“Add”加载到平台上并编译生成一个名为“CIGPRG_DEVS_SD_SPCE3200.ecc”的库文件，即 SD 卡驱动程序库文件。



至此，在嵌入式操作系统 eCos 中成功地创建了一个基于凌阳 SPCE3200 的 SD 卡驱动程序。有了该驱动程序，就能正常访问 SD 卡了。

除了通过 I/O 组件访问硬件设备之外，eCos 还提供了 FAT 文件系统用以管理块设备。FAT 文件系统的使用，可以大大方便嵌入式系统与 PC 系统之间的文件交换，并为应用程序提供更为方便的操作接口。

参考文献

- [1] 郭淳学. 嵌入式系统是嵌入式软件与 IC 发展基础[N]. 中国电子报, 2008.6.30 (第 3 版).
- [2] 周源. 片上系统(SOC)设计流程及其集成开发环境[J]. 上海应用技术学院学报, 2008, 32 (4): 23-26.
- [3] 管晓春. 基于凌阳 SPCE061A 的嵌入式应用系统设计[J]. 电子产品世界, 2008, 187 (2): 10-13.
- [4] TOM R. H. ARM dons armor-trustzone security extensions strengthen ARMv6 architecture"[J]. MPR August 25, 2007,(5):14-17.
- [5] JACH B. Common criteria portal. <http://www.commoncriteriaportal.org/>.2006.
- [6] 凌阳科技. SPCE3200 嵌入式 eCos 系统驱动开发. <http://www.unsp.com/>. 2008.
- [7] 罗亚非. 凌阳 16 位单片机应用基础[M]. 北京: 北京航空航天大学出版社, 2007: 132-134.
- [8] 凌阳科技. SPCE3200 实验仪——eCos 平台篇. <http://www.unsp.com/>.2008.

(收稿日期: 2008-12-11)