

基于开放系统的测试软件架构设计研究

张秀芳^{1,2}, 刘洲洲³

(1. 西安欧亚学院, 陕西 西安 710065;

2. 西安工程大学 电信学院, 陕西 西安 710000;

3. 西北工业大学 计算机学院, 陕西 西安 710072)

摘要:针对嵌入式设备的文件开放系统实现的数据库受到文件系统的牵制从而导致影响到数据库的慢速存取的问题,文中以嵌入式开放系统 Nucleus 为例,将基于数据库设备,直接面向 Flash 驱动的方法,将若干个数据库由每个库名和可变长记录表和自定义若干索引表组成。其中的索引表进行排序,同时为数据库提供向上的寻址空间,将存储空间的擦除块映射用地址映射表来实现,使逻辑地址空间连续且可逆。记录主要由一个类别加上一个目录信息再加上记录数据构成,对于数据库的保护采用备份两份数据库定义文件和两份索引文件的方式,同时对于擦除块的掉电保护在底层设备独立完成以及记录等操作的掉电保护由数据库系统操作完成。测试软件结果表明,提出的方法便于移植,其他模块崩溃不会直接影响数据库系统,且提高了存取速度,同时占用内存也大大减少。

关键词:开放系统;Nucleus;数据库;测试软件;架构

中图分类号:TP311.5

文献标识码:A

文章编号:1673-629X(2020)07-0087-04

doi:10.3969/j.issn.1673-629X.2020.07.019

Design of Test Software Architecture Based on Open System

ZHANG Xiu-fang^{1,2}, LIU Zhou-zhou³

(1. Xi'an Eurasia University, Xi'an 710065, China;

2. School of Telecommunications, Xi'an University of Engineering, Xi'an 710000, China;

3. School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: Aiming at the problem that the database implemented by the open file system of embedded devices is constrained by the file system, which results in slow access to the database, we take Nucleus as an example, which is based on the database device and directly oriented to the Flash-driven method. Several databases consist of each library name and variable-length record tables and custom index tables. The index table is sorted, and the upward addressing space is provided for the database. The erasable block mapping of the storage space is realized by the address mapping table, so that the logical address space is continuous and reversible. Records are mainly composed of one category plus one catalog information plus records. The database is protected by backing up two database definition files and two index files. At the same time, the power loss protection of erasure block is completed independently in the underlying equipment and the power loss protection of recording operation is completed by the database system operation. The test results show that the improved method is easy to be transplanted, and other module crashes will not directly affect the database system. It also increases the access speed and reduces the memory consumption.

Key words: open system; Nucleus; database; testing software; architecture

0 引言

当前嵌入式开放系统已经广泛应用在各行各业,比如通讯、电子、航空、过程控制、消费电子产品等领域^[1-4]。当前嵌入式操作系统有数百种,比较流行的有苹果 iOS,谷歌的安卓系统,ATI 的 Nucleus Plus、VXWorks 等。用户可以根据自身的软硬件环境的要

求,选择合适的操作系统。其中数据库作为系统的关键模块,有至关重要的作用,当前常见的数据库存储介质由连续的 Flash 组成,同时配合驱动和文件管理系统,且逻辑地址空间可逆且连续。文中以嵌入式 Nucleus 数据库为例^[5-9],设计一种测试软件架构,采用直接面向 Flash 驱动的方法,相比传统存取方法速

度有较大提高,内存占用也随之降低。

1 Nucleus 开放系统

Nucleus 是为实时嵌入式应用而设计的一个抢先式多任务操作系统^[10-12]。Nucleus 系统的各层通讯协议都提供了由 ANSIC 写成的源码,有利于系统的可移植性并支持大部分处理器。而且 Nucleus 系统针对不同的处理器的源码大部分也是相同的且少部分跟 CPU 相关由汇编完成。该操作系统完全开源,可以通过修改开源代码进行所需的配置,调试可以在中断和寄存器级进行。核心函数 API 和程序代码链接一起生成目标代码,可以直接烧制到目标板卡中,整个核心代码区内核规模非常小,方便调试。

2 数据库测试架构

文中提出了面向 Flash 驱动的数据库方法,该方法向上提供寻址空间,且空间为线性,将连续且可逆的逻辑地址空间用地址映射表的擦除块来映射。数据库相关设备和本身数据库相互关联,其中在连续的逻辑地址的基础上,记录需要写在同一个擦除块,数据库设备则需要提供动态记录空间。数据库设备不能独立完成整理擦除块,需要依赖记录索引,同时设备需要记录擦除块的工作情况,使得动态分配空间功能得以完善。擦除块的掉电保护在底层设备独立完成,记录等操作的掉电保护由数据库系统操作完成。记录索引表和交换块以及记录区构成本数据库系统,索引表占用首地址空间,而交换块不占用地址空间且存储索引表最后节点部分,记录区占用其他地址空间,详细记录每个数据库地址类别大小等属性,且记录唯一地址信息,ID 号不冲突同时可区分数据库类别^[13-15]。

2.1 地址映射

文中假定数据库相关设备存储大小为 2 M,且擦除块大小占据 64 K,擦除块为 32 块,其中一块作为交换块,其余为擦除块,每个擦除块前四字节写入 ID 号和 bank 号,结构如图 1 所示。

0 ~ 27 Bits, Bank 号	28 ~ 31 Bits, Flag
---------------------	--------------------

图 1 擦除块结构

当嵌入式开发系统初始化时,数据库相关设备则开始扫描 32 块 ID,并动态生成一个映射表且存储在系统的内存中,由于保留一个交换块,只可用 31 节点,定义最后节点地址,其余节点包括物理地址和擦除块信息。文中映射表用 C 语言表述成:

```
struct tagDbDevTab{
    long Addr;//物理地址
    long Free;//剩余大小
```

```
long Dirty;//脏块大小
```

```
} DbDevTab[ DB_DEV_SIZE/0x10000];
```

2.2 分配记录空间和脏块回收

文中数据库相关设备可为系统动态分配记录空间,数据库记录表存储在同一块擦除块,且不能夸块存储。将地址映射表的空闲字段进行优先分配,减少碎片和物联块充分使用,由于操作如删除和修改等会产生脏块,为有效利用,脏块需要反复利用,回收记录触发后当设备空间无空余,脏块回收则选中要回收的目标块,根据地址映射表记录字段的大小优先判断,选定后记录转存其他空间同时删除原记录。空间所有记录移除后,调用擦除函数写入,同时修改地址映射表,如果脏块太多,需要整块回收,将有用记录逐条转存其他空间后擦除该空间,将保留空间作为逻辑空间写入记录,这样将整个物理空间有效利用。

2.3 数据库设备初始化

数据库系统初始化先由设备初始化再进行数据本身初始化。初始化不但要对调电保护做出检测和数据修复,还要初始化许多全局变量,它们包括:地址映射表、保留 Bank 的物理基地址、记录索引表的最大节点、索引表中已经删除的节点个数、系统索引的记录个数等,另外还要打开系统索引。

初始化过程为扫描各个 Bank ID,填写地址映射表的 addr 字段,如果发现 Bank ID 的 Flag 是 0xF,则擦掉该 Bank,如果发现 Bank0 ID 是 0xE0000000,则改写为 0xC0000000。如果注册表没有填写满,也就是有的逻辑 Bank 找不到,则找到一个 ID 为 0xFFFFFFFF 的块,擦除后写入 ID 为 0xC * * * * *,并作为这个逻辑 Bank。找到保留 Bank 并把基地址写入地址映射表的最后一个节点。扫描 Bank0 的记录索引表的 addr 字段,如果某个节点的 Flag 是 0xF,改写为 0x8,如果 Flag 是 0xE,读出该节点 id,找出该记录旧的节点并删除,同时改写 Flag 为 0xC。最后统计出记录索引表的最大节点以及索引表中已经删除的节点个数。还要统计出各个 Bank 的已经用掉的空间、剩余空间、脏记录的空间,其中剩余空间是根据该 Bank 的最后一条记录再加上这个记录的大小算出来一个初始值,然后再从该 Bank 的最后向前搜索到初始值,遇到不是 0xFFFFFFFF 的时候退出,该地址就是该 Bank 的剩余起始地址。

3 掉电保护

嵌入式开放数据库设备必须使用掉电保护,目的是嵌入式系统在突发掉电情况,资料完整性得以保存,要求在写操作步骤进行。首先系统启动后初始化过程中需要记录之前掉电现象是正常模式下掉电还是异常

模式下掉电,当突发掉电可以还原掉电前资料信息,数据库的空间地址分别由四个字节信息组成,相关记录含义目录名的相关性不得冲突、同时依靠数据库的鲁棒性。

数据库的掉电保护是指在用户打开数据库后,在数据库的操作期间(关闭数据库之前)发生低电压或者突然掉电时应保证数据库资料的完整性。在这个过程中允许且只允许丢失用户最后一次的操作。在这个数据库的实现中,利用了两套数据文件来保证这一点,每个数据库打开后同时存在两套文件:一是(*.dat, *.def, *.ida, *.idb, *.idc,...),另一套是(*.dat, *.bef, *.bda, *.bdb, *.bdc,...),其中*.dat是同一个文件。其余的文件两者一一对应。在数据库打开时,根据两套文件的完整性(文件个数)及其相互关系(记录数的个数)来确定当前使用哪一套文件,并将另外一套文件全部置空(此时这套文件无效)。在对数据库进行操作的过程中,先将当前这套文件的内容修改后写入另一套文件中,全部完成后将当前使用的文件标志修改并将原来那套文件全部置

空。这样一来就可以保证在任何时候都有一套完整的数据,在关闭数据库的时候将空的那一套文件删除。对于数据库的保护,在本数据库系统里数据处理过程中采用备份的方式:两份数据库定义文件(*.def, *.bef),两份索引文件(*.id*, *.bd*),但只有一份数据文件,因为数据块的使用可以由文件系统保障。这两份文件组在使用中互为备份:在第一组为当前使用时,执行操作前恢复当前文件组的数据,然后在备份组上进行修改,修改完成,在将当前组指向第二组。

4 系统实验测试

通过上位机和下位机测试两种方式:同时按下电源开关键(挂断键)和退出键(右功能键)用传输线将小机同PC连接起来:在PC端启动“数据库测试程序”,小机端:按开关键开机连接成功后,在PC端选择“系统自检”。首先进行串口测试如图2所示,同时分别测试串口0和串口1,并显示测试结果。在测试夹具上,会将串口0和串口1短接,故各串口自发自收,自动判断测试结果。

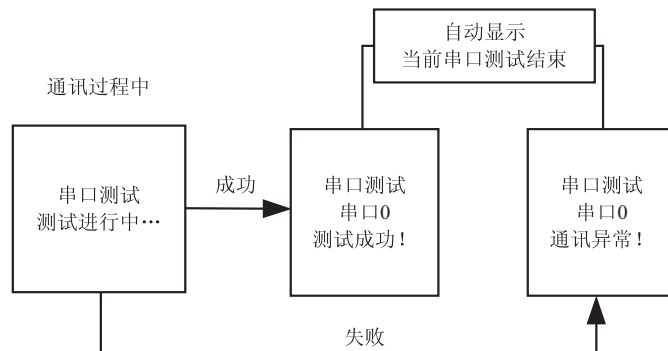


图2 串口通讯

然后再通过SRAM测试和DATA/PROGRAM校验,如图3和图4所示。测试过程中对于数据库的保护,在本数据库系统里数据处理过程中采用备份的方式:两份数据库定义文件(*.def, *.bef),两份索引文件(*.id*, *.bd*),但只有一份数据文件,因为数据块的使用可以由文件系统保障。这两份文件组在使用中互为备份:在第一组为当前使用时,执行操作前恢复当前文件组的数据,然后在备份组上进行修改,修

改完成,再将当前组指向第二组。存储一条记录时,首先存贮该记录的数据,然后存入记录的节点,最后对该记录进行排序。针对上述流程,发生掉电时,如果发生在存数据过程,该记录将完全不存在;如果发生在存节点过程中,该节点将不存在,但会浪费一块数据文件空间;如果发生在排序过程中,由于记录数目不一致,将会删除失效的一套文件。数据库保持原来的结构。

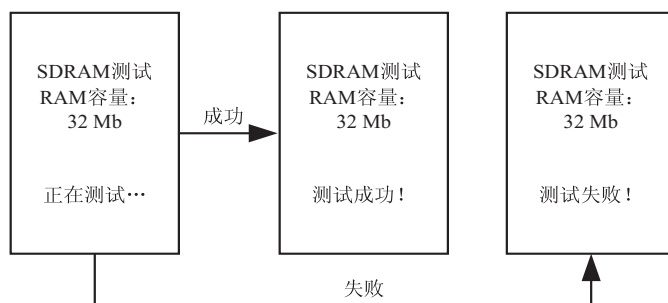


图3 SDRAM测试

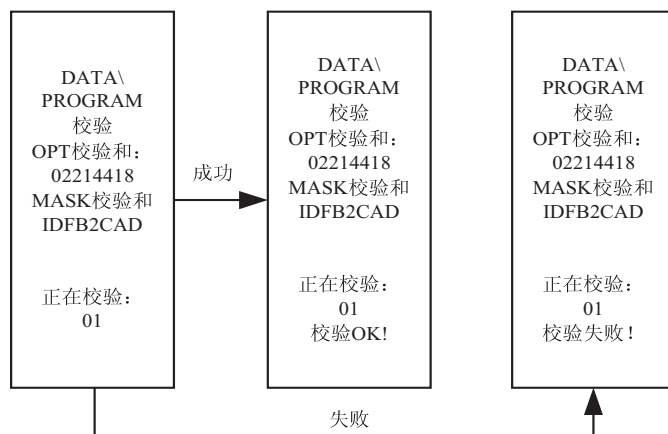


图 4 DATA/PROGRAM 校验

通过软件测试,表明该方法便于移植且当其他模块崩溃时不会直接影响数据库系统同时存取速度有所提高,同时占用内存也会大大降低,由于有掉电保护,系统可靠性也得到了增强。

5 结束语

Nucleus 作为开放嵌入式系统,具有实时性和多任务性,系统资源的调度都在其有效管理控制下可以使嵌入式多任务的开发变得省时省力,同时高效,文中分析了基于 Nucleus 的嵌入式数据库测试软件架构的机理,对于嵌入式实时操作系统的开发具有重要的意义。

参考文献:

- [1] AGBINYA J I, MASIHPUR M. Power equations and capacity performance of magnetic induction communication systems[J]. Wireless Personal Communications, 2012, 64(4):831-845.
- [2] MASIHPUR M, FRANKLIN D, ABOLHASAN M. Multi-hop relay techniques for communication range extension in near-field magnetic induction communication systems[J]. Journal of Networks, 2013, 8(5):999-1011.
- [3] 汪 洋,禹 珉.基于 Nucleus 操作系统实现 TCP 和 UDP 协议通信[J]. 软件工程, 2018, 21(9):29-33.
- [4] 段亚林,谢永斌.基于 Nucleus 的通信终端中 AT 指令模块的设计与实现[J]. 计算机测量与控制, 2015, 23(11):3744-3746.
- [5] 陈连坤. 嵌入式系统的设计与开发[M]. 北京:清华大学出版社, 2005.
- [6] 台湾广达微电子设计有限公司. 基带处理器手册[M]. 台湾:台湾广达微电子设计有限公司, 2007.
- [7] LUO J, FENG D, CHEN S, et al. Experiments for on-line bearing-only target localization in acoustic array sensor networks[J]. World Congress on Intelligent Control & Automation, 2010, 20(1):1425-1428.
- [8] 蔡志明,卢传富,李立夏. 精通 Qt4 编程[M]. 北京:电子工业出版社, 2008.
- [9] AKYILDIZ I F, KASIMOGLU I H. Wireless sensor and actor networks: research challenges[J]. Ad hoc Networks, 2004, 2(4):351-367.
- [10] 王继刚,顾国昌,徐立峰. 面向智能手机的嵌入式实时操作系统[J]. 中兴通讯技术, 2005, 11(5):41-44.
- [11] 彭 涛. 嵌入式操作系统移植技术研究[D]. 武汉:华中科技大学, 2006.
- [12] 李元实,王 智,鲍 明,等. 基于无线声阵列传感器网络的实时多目标跟踪平台设计及实验[J]. 仪器仪表学报, 2012, 33(1):146-154.
- [13] MARWEDEL P. Embedded system design[M]. 北京:科学出版社, 2007.
- [14] ALESSANDRO G D. Microprocessor design for embedded system[J]. Journal of Systems Architecture, 2009, 45(6):1139-1149.
- [15] 魏振华,洪炳熔,乔永强,等. 嵌入式实时操作系统 Nucleus 中线程控制部件的实现方法[J]. 计算机应用研究, 2003, 20(4):97-99.