

基于NAND Flash的大容量立体封装芯片在嵌入式系统中的应用

Application of Bulk Stereo Package Chip Based on NAND Flash in Embedded System

叶振荣 王烈洋 珠海欧比特控制工程股份有限公司 (广东 珠海519080)

摘要: NAND Flash应用的困难在于管理和需要特殊的系统接口。本文介绍了一种利用MCU存储器管理接口结合I/O口来实现NAND Flash存储结构的搭建和管理的方法,并介绍底层的驱动程序。本文网络版地址: <http://www.eepw.com.cn/article/192739.htm>

关键词: 嵌入式系统; 立体封装; 存储器; 系统接口

DOI: 10.3969/j.issn.1005-5517.2013.12.014

元件介绍

基于sparc架构的MCU

S698-MIL 是珠海欧比特控制工程股份有限公司为了满足嵌入式应用而开发的32位RISC高性能嵌入式微处理器,它遵循SPARC V8构架。

S698-MIL 内部配置了32位整数处理单元(IU),32/64位浮点处理单元(FPU)。

S698-MIL 的应用软件开发环境具有很强的灵活性,除了可以使用欧比特公司提供的专用多任务嵌入式实时操作系统ORION外,开发者还可以选择如RTEMS、VxWorks等现今流

行的嵌入式操作系统进行开发。

S698-MIL 可应用于包括税控收款机、银行POS机、电力系统等高端工业控制领域和消费电子领域以及高性能高可靠的航空、航天及武器领域。

NAND Flash结构的存储芯片

目前的Flash memory 主要包括以下两大类:针对程序和数据存储的NOR flash;针对大容量存储的NAND flash。其中,NOR的特点为芯片内执行(XIP, eXecute In Place),这样应用程序可以直接在flash闪存内运行,不必再把代码读到系统RAM中,随机读取快、功耗低、稳定性高;而

NAND 的特点为容量大、写速度快、芯片面积小。

VDNF64G08是一个快速、高存储密度的随机访问存储器。它由8个8G位的NAND Flash芯片堆叠而成,结构见图2。整个模块采用堆叠技术,它们之间的互连接线非常短,寄生电容小。这种芯片非常适用于高速、高性能、高容量的嵌入式系统中。

VDNF64G08对一个2048(+64)字节的页进行典型的编程操作只要200μs的时间,对一个128K(+4K)大小的块进行擦除需要1.5ms的时间,对页中一个字节的读周期为25ns。它的I/O管脚既作地址和数据的输入输出口,也作为命令的输入口。VDNF64G08的片上写控制器能自动完成所有的编程、擦除功能包括产生所需的脉冲重复和内部数据校验。VDNF64G08能擦除和编程百万次以上,并通过ECC或实时制定算法保

证擦除和编程的可靠性。

硬件设计

由于S698没有专用的NAND Flash的专用接口,所有

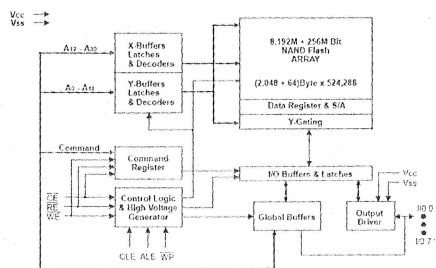


图1 NAND Flash的内部结构

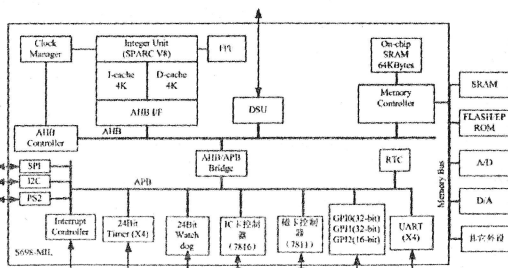


图2 S698-mil内部结构

责任编辑：万钟

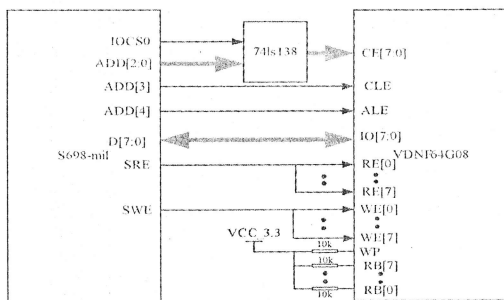


图3 硬件连接简图

NAND Flash与S698-mil的通信使用GPIO实现，NAND Flash的IO口与S698-mil的GPIO0低8位GPIO0[7:0]连接，用于数据，地址，命令的传输。NAND Flash的片选信号使用S698-mil的低3位地址线ADD[2:0]，通过3-8译码实现，其中3-8译码器使用S698-mil的IOCS0控制工作，防止对VDNF64G08的误操作。VDNF64G08的读写信号直接与S698-mil的读写信号相连，如果线路较长可以考虑串联一个49.9R的电阻。命令锁存CLE，地址锁存ALE分别使用S698-mil的地址线ADD[3]，ADD[4]。状态信号RB上拉10kΩ电阻到电源，不再与S698-mil相连，VDNF64G08的状态信息通过IO口读寄存器实现，以节约S698-mil的端口资源。写保护端子上拉10kΩ到电源，否则不可擦除及编程。相关连接见图2。

软件设计

根据前面的介绍，通过图3的硬件连接方式，可以实现对芯片的读写，擦除等控制操作，已可以满足各种场合的应用，S698-mil处理器是通过GPIO来操作控制VDNF64G08的，所以需要到底层操作需要比较了解，下面简单介绍一些常用操作的实现。

S698-mil芯片GPIO0对应的寄存

器地址为0x2000 0000 - 0x27ff ffff，因为VDNF64G08芯片位宽是8bit的，所以需要把MCU的GPIO位宽也定义成8bit，根据前面的硬件连接可以知道地址低3位对应到VDNF64G08芯片的片选信号，add[4]对应到CLE，add[5]对应到ALE，为了方便

编程，这里做以下宏定义：

```
#define NF_ADDREG(CE)
(*(volatile unsigned char *) (0x20000010 + CE))

#define NF_CMDREG(CE)
(*(volatile unsigned char *) (0x20000008 + CE))

#define NF_DATAREG(CE)
(*(volatile unsigned char *) (0x20000000 + CE))
```

读芯片ID

根据芯片指令表，可以知道读ID芯片只需要写入命令0x90，然后在写入地址0x0即可，详细操作可以根据以下时序图进行：

```
int readID(unsigned char CE,unsigned IDlength)
{
    unsigned char ID[ ];
    NF_CMDREG(CE) = 0x90;
    NF_ADDREG(CE) = 0x0;
    delayed(1);
    for(i = 0; i < IDlength; i++)
    {
        ID[i] = NF_DATAREG(CE);
    }
    i = i - 1;
    printf("CHIP CE d %ID:0x%x",ID[0]);
```

```
while(i){
    printf("-%x",ID[IDlength - i]);
    i--;
}
printf("\n\r");
}
```

芯片坏块查询

由于NAND Flash的工艺不能保证NAND的Memory Array在其生命周期中保持性能的可靠，因此，在NAND的生产中及使用过程中会产生坏块。为了检测数据的可靠性，在应用NAND Flash的系统中一般都会采用一定的坏区管理策略，而管理坏区的前提是能比较可靠的进行坏区检测。检测流程如图4所示。

芯片擦除操作

NAND Flash的擦除操作是以块为基础进行的。只有已擦除的块才能编程，因为NAND Flash芯片的工艺特性决定了，芯片的CELL只能由1写成0，而不能由0写成1。芯片的擦除操作有2个命令周期和3个地址周期构成，其中列地址不需要输入，并且行地址的页地址也不会影响块擦除效果，即块擦除地址只有块地址有效，操作时序及流程图如图5所示。

芯片页读操作

页读操作通过将00h指令写入指令寄存器，接着写入5个地址（2个列地址，3个行地址），最后写入30h指令来启动。一旦页读指令被器件锁存，下面的页读操作就不需要再重复写入指令了。

写入指令和地址后，处理器可以通过对信号线R/的分析来判断该操作是否完成。如果信号为低电平，表示器件正“忙”；为高电平，说明器

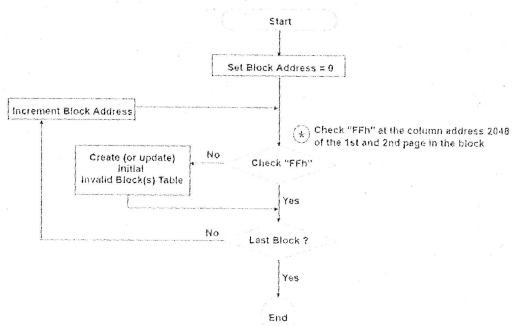


图4 检测流程

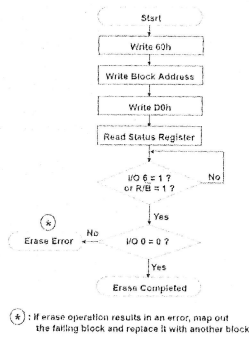


图5 块擦除时序

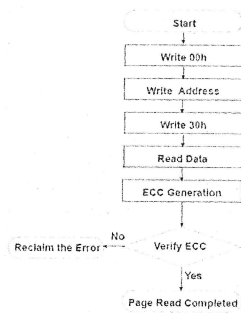
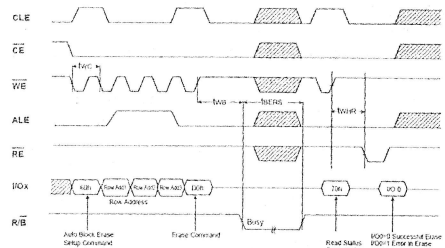


图6 页读时序

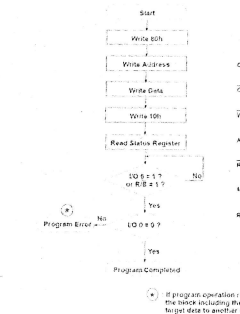
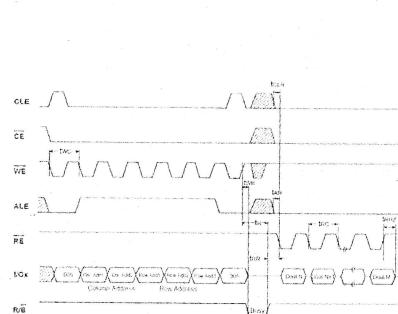
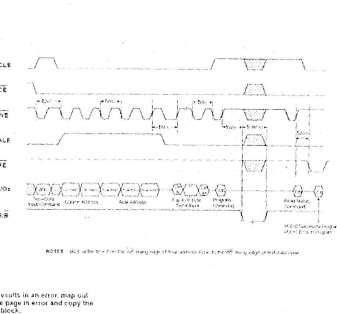


图7 页编程时序



件内部操作完成, 要读取的数据被送入了数据寄存器。外部控制器可以在以50ns为周期的连续脉冲信号的控制下, 从I/O口依次读出数据。

连续页读操作中, 输出的数据是从指定的列地址开始, 直到该页的最后一个列地址的数据为止。操作时序及流程图如图6所示。

芯片页编程操作

VDNF64G08芯片的写入操作也以页为单位。写入之前必须先擦除, 否则写入将出错。

页写入周期总共包括3个步骤: 写入串行数据输入指令(80h), 然后写入5个字节的地址信息, 最后串行写入数据。

串行写入的数据最多为4096字节, 它们首先被写入器件内的页寄存器, 接着器件进入一个内部写入过程, 将数据从页寄存器写入存储宏单元。

串行数据写入完成后, 需要写入“页写入确认”指令10h, 这条指令将初始化器件的内部写入操作。如果单独写入10h而没有前面的步骤, 则10h不起作用。10h写入之后, 内部写控制器将自动执行内部写入和校验中必要的算法和时序, 这是系统控制器就可以去做别的事了。

内部写入操作开始后, 器件自动进入“读状态寄存器”模式, 在这一模式下, 当RE/CE为低电平时, 系统就可以读取状态寄存器。系统可以通过检测R/B的输出, 或读状态寄存器的状态位(I/O7)来判断内部写入是否结束。在器件进行内部写入操作时, 只有读状态寄存器指令和复位指令会被响应。当页写入操作完成, 应该检测写状态位(I/O1)的电平。

内部写校验只对1没有成功地写为0的情况进行检测。指令寄存器始

终保持着读状态寄存器模式, 直到其它有效的指令写入指令寄存器为止。操作时序及流程图如图7所示。

结束语

VDNF64G08是一个快速、高存储密度的随机访问存储器。整个模块采用堆叠技术, 它们之间的互连线非常短, 寄生电容小。在研发初期理论论证和生产之后的实验数据表明, 这种芯片非常适用于高速、高性能、高容量的嵌入式系统中, 并得到用户的好评。

参考文献:

- [1] 珠海欧比特控制工程股份有限公司.VDNF64G08-F使用说明书[Z].2013
- [2] H M Peitel, P J Deitel. C How to program, second Edition. 蒋才鹏等译. C程序设计教程[M]. 北京: 机械工业出版社, 2000
- [3] 珠海欧比特控制工程股份有限公司.VDNF64D08-K使用说明书[Z].2013
- [4] 珠海欧比特控制工程股份有限公司.S698-T芯片用户手册[Z].2011
- [5] 夏宇闻. Verilog数字系统设计教程[M]. 北京: 北京航空航天大学出版社, 2003
- [6] SPARC International Inc. The SPARC Architecture Manual[K]. Version 8