

# 一种基于 CPLD 译码的 DSP 外部 Flash 烧写方法

张芝贤<sup>1</sup>, 邱委华<sup>1</sup>, 朱庆贤<sup>2</sup>, 王明贺<sup>1</sup>

(1. 沈阳航空工业学院 电子信息工程学院, 辽宁 沈阳 110136;

2. 中北大学 信息与通信工程学院, 山西 太原 030051)

**摘要:** 以实际的图像监控系统为背景, 介绍了对 S29AL008D Flash 存储器进行操作的指令格式和控制方法。通过 CPLD 实现 DSP 和 Flash 存储器之间的硬件连接, 给出了部分 VHDL 源程序。利用 TMS320C5509A DSP 通过 CPLD 快速译码实现 Flash 烧写具有接口简单、通用性强的优点, 在实践中证明了方法的有效性。

**关键词:** DSP; S29AL008D; CPLD 译码; Flash 烧写

中图分类号: TP333

文献标识码: A

## Burning method for DSP external Flash based on CPLD decoding

ZHANG Zhi Xian<sup>1</sup>, QIU Wei Hua<sup>1</sup>, ZHU Qing Xian<sup>2</sup>, WANG Ming He<sup>1</sup>

(1. College of Electronic Information Engineering, Shenyang Institute of Aeronautical Engineering, Shenyang 110136, China;

2. College of Information and Communication Engineering, North University of China, Taiyuan 030051, China)

**Abstract:** This paper introduces the instruction format and control methods of the operating to S29AL008D Flash, taking the practical project of image monitor system as background. By the use of CPLD, it realizes the hardware connection between the DSP and Flash memory, and some core VHDL source programs are given. It has some advantages of simple to interface and strong universality to realize Flash burning with the use of CPLD's fast decoding by TMS320VC5509A DSP. It is demonstrated effectiveness of the project.

**Key words:** DSP; S29AL008D; CPLD decoding; Flash burning

数字信号处理器 DSP 因其运算精度高、实时性强等特点, 在通信、工业控制和消费类电子等领域得到了广泛应用, 成为进行数字信号处理的首选器件。在图像监控系统中, 往往要对大量的数据和应用程序进行处理和非易失性存储。采用 Flash 存储器对这些大量数据和应用程序进行存储是一种性价比较高的选择。本文设计的以 TI 公司 DSP 芯片 TMS320VC5509A (以下简称 VC5509A) 为主处理器的图像监控系统中, 使用了 AMD 公司 S29AL008D Flash 存储器作为数据和应用程序存储芯片。使用 VC5509A 对 S29AL008D 进行操作, 在硬件接口设计中采用 CPLD 进行译码和时序控制。

### 1 VC5509A 及其外部存储空间

TMS320VC5509A 是 TI 公司的一款 16 位高速低功耗定点 DSP 芯片, 主要应用于对音频、静态图像的数字信

号处理, 是设计便携设备的较佳解决方案。该 DSP 芯片在核心电压为 1.35 V 时最大工作频率为 144 MHz; 地址线为 24 位, 对地址的寻址范围为 16 M×8 bit 或 8 M×16 bit。VC5509A 采用统一的编址方式, 即存储空间地址没有重叠但是寻址方式却有 2 种不同的方式: 字节寻址和字寻址<sup>[1]</sup>。当 DSP 中的 CPU 访问 VC5509A 程序存储空间时, 通过外部存储器接口 (EMIF) 对外部存储空间进行管理。外部存储空间被分成 CE0~CE3 4 个空间, 分别由 EMIF 的 CE0#~CE3# 引脚管理, 外部存储空间划分如图 1 所示<sup>[2]</sup>。CEn#(n=0~3) 与外设的片选端连接。

### 2 硬件电路设计

#### 2.1 VC5509A 与 Flash 的硬件连接

使用 PGE 封装的 VC5509A 对外只提供 14 条地址线引脚, 因此只能寻址 8 K×16 bit 的空间范围<sup>[3]</sup>, 要对

字节地址	VC5509A 芯片存储资源	字地址
000000	存储器映射寄存器(MMR)	
0C0000	DARAM /HPI访问	
008000	DARAM	004000
010000	SARAM	008000
040000	CE0	020000
400000	CE1	200000
800000	CE2	400000
C00000	CE3	600000
FF0000	ROM	CE3
	MPNMC=0	MPNMC=1
FF8000	时有效	时有效
FFC000	同上	同上
FFFFFF	同上	同上

图1 VC5509A 外部存储空间的划分

S29AL008D 的 512 K×16 bit 存储空间寻址需 19 条地址线。解决办法是用 CPLD 实现锁存译码使 D[5:0]具有数据/地址复用功能,为 S29AL008D Flash 存储器提供高 6 位地址<sup>[4]</sup>。VC5509A 和 CPLD 与 S29AL008D 的硬件接口电路如图 2 所示<sup>[5]</sup>。通过 CPLD 的快速译码功能在 CE2 空间模拟 1 个 Flash 高位地址控制寄存器(换页控制寄存器)FPCR,上电复位后通过写 FPCR 控制 FLASH 的高位地址来实现换页功能。此处用 VHDL 对 XC9536XL 进行编程将 FPCR 映射到 0x400000,由于篇幅限制仅给出如下部分核心源程序:

```

begin
  fce<=ce1;
  foe<=aoe;
  fwe<=awe;
  addr<=a3&a2&a1;
  datain<=d5&d4&d3&d2&d1&d0;
  facs<='1' when ce2='0' and addr="000"
    else '0';
  FAR:process(facs,awe,reset)
  begin
    if reset='0' then
      fa<="000000";
    else if reset='1' then
      if awe'event and awe='1' then
        if facs='1' then
          fa<=datain(5 downto 0);
        end if;
      end if;
    end process;
    dataout<=fa when aoe='0'
  and facs='1'
    else "ZZZZZZ";
  
```

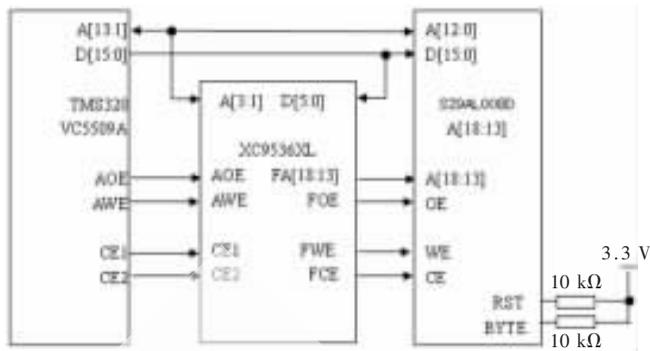


图2 VC5509A 和 CPLD 与 S29AL008D 硬件接口图

### 2.2 VC5509ADSP 相关寄存器的初始化

在 DSP 对 S29AL008D 进行读写访问之前,要对 DSP 内相关寄存器进行正确的设置。此系统的初始化设置如下<sup>[6-7]</sup>:

```

ioport_unsigned int *ebsr=(unsigned int *) 0x6c00;
ioport_unsigned int *eger=(unsigned int *) 0x800;
ioport_unsigned int *ce11=(unsigned int *) 0x806;
ioport_unsigned int *ce12=(unsigned int *) 0x807;
ioport_unsigned int *ce13=(unsigned int *) 0x808;
*ebsr=0x01; //外部总线选择寄存器选定完全
EMIF 模式
*eger=0x0a10; //关闭 ARDY 控制
在 CPU 一半工作频率下,设置 CE1
*ce11=0x162a //设置存储器模式为异步 16 位,
读操作的 setup、strobe、hold 与写操作的相同
*ce12=0x162a; //写操作的 setup、strobe、hold
*ce13=0;
  
```

### 2.3 Flash 存储器的操作

对 S29AL008D 的读写过程必须严格按照 S29AL008D 提供的命令时序来完成。表 1 给出了 S29AL008D 对数据字进行复位、擦除、读、写的命令时序<sup>[7]</sup>。

表 1 S29AL008D 的命令时序<sup>[7]</sup>

	周期	总线周期											
		第一步		第二步		第三步		第四步		第五步		第六步	
		地址	数据	地址	数据	地址	数据	地址	数据	地址	数据	地址	数据
读数据	1	RA	RD										
复位	1	XXX	F0										
编程	字	555	AA	2AA	55	555	A0	PA	PD				
	字节	AAA		555		AAA							
芯片擦除	字节	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10
	字节	AAA		555		AAA		AAA		555		AAA	
扇区擦出	字	555	AA	2AA	55	555	80	555	AA	2AA	55	SA	30
	字节	AAA		555		AAA		AAA		555			

## 技术与方法 Technique and Method

Flash 操作写子函数 C 程序如下:

```
# define flash_ba 0x200000
//ptr 是 FLASH 存储器映射在整个 DSP 系统地址空
间中的基地址
# define FAR
unsigned int PA,PD;
void write_se(unsigned int se_addr,unsigned int se_data)
{
    unsigned int *flash_adr,*FAR=(unsignedint*)
        0x400000;
    flash_adr=(unsigned int * )
        (flash_ba+se_addr);
    //se_addr 由 FAR 和 A[13:1]来决定
    *flash_adr=se_data;
}
```

对 Flash 进行写操作前必须先擦除片内原有的数据。Flash 的整片字擦除操作时序 C 语言实现如下:

```
void Erase-Chip()
{
    void write_se (0x555,0xAA);
    void write_se 0x2AA,0x55);
    void write_se 0x555,0x80);
    void write_se 0x555,0Xaa);
    void write_se 0x2AA,0x55);
    void write_se 0x555,0x10);
}
```

Flash 要对写入的数据进行校验以及判断操作是否正确完成。Flash 写校验 C 语言实现如下:

```
void program_check (void)
{
    unsigned int algorithm_out;
    unsigned int datapolling_bit;
    unsigned int exceed_time;
    algorithm_out = read_array( PA);
    datapolling_bit = intalgorithm_out &0x80;
    exceed_time = algorithm_out &0x20 ;
    while((datapolling_bit !=PD&0x0080)
        &&(exceed_time != 0x20))
    {
        algorithm_out = read_array( PA);
        datapolling_bit = algorithm_out &0x80;
        exceed_time = algorithm_out &0x20;
    }
    if (exceed_time == 0x20)
    {
```

```
datapolling_bit =read_array(PA) &0x80 ;
if (datapolling_bit !=PD & 0x0080)
{
    reset_flash();
    wrong();
}
}
```

Flash 字编程程序如下:

```
void Word-Program(void)
{
    write_se (0x555,0xAA);
    write_se 0x2AA,0x55)
    write_se (0x555,0XA0);
    write_se (PA,PD);
    program_check();
}
```

由于篇幅所限, 本文仅给出部分核心程序代码, 读者可利用上述代码编写一个完整的程序。

本文以解决 DSP 静态图像无线传输系统项目中对大规模程序和大量图像数据的存储问题为背景, 并最终在整个系统调试中实现其存储功能和程序的二次加载。经验证, 本接口设计和编程实现具有操作方便, 易于扩展等优点。

对 DSP 外部 Flash 编程虽不是一项关键技术, 但它在整个 DSP 嵌入式系统开发中却有着至关重要的作用。如果开发者在设计之初就掌握了这项技术, 将会大大方便系统的调试, 缩短开发时间。

参考文献

- [1] Texas Instruments. TMS320VC5509 DSP external memory interface(EMIF) reference guide[S]. Literature Number:SP-2 RU670, October 2003.
- [2] 彭启琮, 武乐琴.TMS320VC55X 系列 DSP 的 CPU 与外设[M]. 北京:清华大学出版社, 2005: 191-212.
- [3] Texas Instruments.TMS320C55x DSP CPU reference guide [S]. Literature Number:SPRU371F, February 2004.
- [4] 申敏, 邓奕冰.DSP 原理及其在移动通信系统中的应用[M]. 北京:人民邮电出版社, 2001: 130-139.
- [5] 刘伟, 闫玉华.基于 CPLD 译码的 DSP 二次 Bootloader 方法[J].电子技术应用, 2008, 34(10): 61-63.
- [6] 李坤, 肖恒.TMS320VC33 的引导加载方法设计与实现[J].电子技术应用, 2003, 23(2):166-168.
- [7] Publication number S29AL008D\_00 revision A amendment 3. Issue Date June 16, 2005.

(收稿日期: 2009-03-12)