

NorFlash 使用手册

南京博芯电子技术有限公司

2009-04

This document contains information on a product under development. Prochip Corp reserves the right to change or discontinue this product without notice.
Prochip Crop, 2009. All rights reserved.

版权说明

版权所有，未经南京博芯信息技术有限公司的授权，本说明文档不可以被复制或以任何形式或方式（电子的或是机械的）传播，包括影印，记录或是用其他任何信息存储及检索系统。文档所描述的任何一种电路对于第三方没有专利权及专利特许权。

否认书：

南京博芯信息技术有限公司保留对文档随时进行修改的权利，无须任何申明。南京博芯信息技术有限公司所提供的信息是精确可靠的。对于它的应用以及由于应用而导致违反专利权或是第三方的其他权利，本公司不负任何责任。

版本历史

日期	版本	描述	备注
2009-4-22	1.0	初稿	

目 录

版本历史	2
一、 NorFlash 模块简介	4
1.1 概述	4
二、 NorFlash 模块功能介绍.....	4
2.1 NorFlash 功能介绍	4
2.2 NorFlash 基本特性	4
2.3 寄存器介绍	5
三、 NorFlash 模块实现原理.....	5
3.1 NorFlash 硬件实现原理	5
3.2 NorFlash 软件实现原理	6
四、 NorFlash 模块测试流程及结果说明.....	7
4.1 NorFlash 测试流程图	7
4.2 NorFlash 测试结果说明	8
五、 其他注意事项	9

一、NorFlash 模块简介

1.1 概述

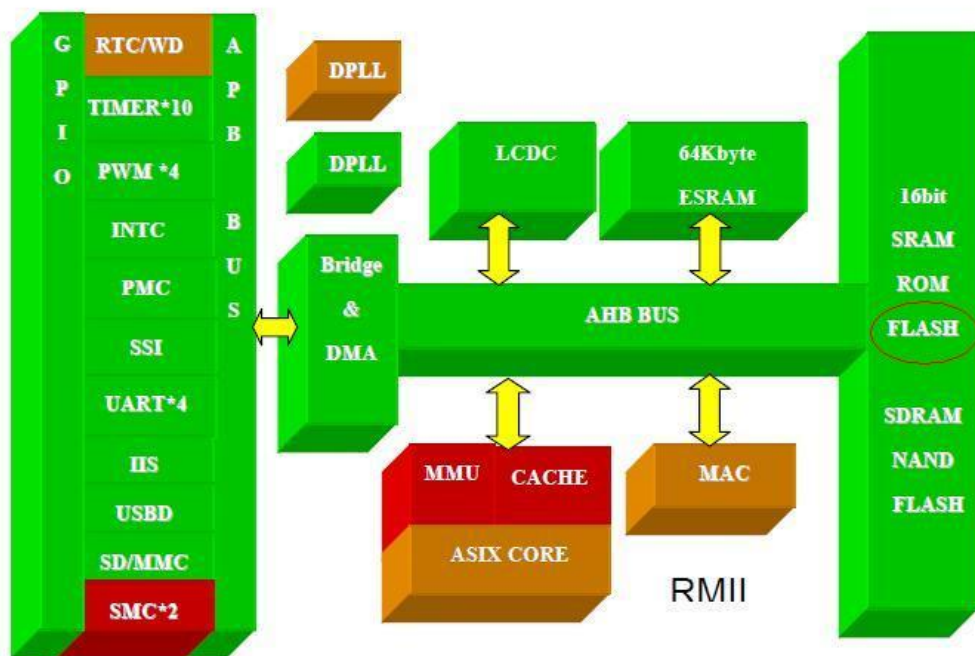


图1 NorFlash 在SEP4020中的位置

二、NorFlash 模块功能介绍

2.1 NorFlash 功能介绍

NorFlash 属于 Code Flash 类型，用于储存系统程序代码及少量资料。NorFlash 带有通用的 SRAM 接口，有足够的地址引脚来寻址，可以轻松地挂接在 CPU 地址和数据总线上，对 CPU 接口要求低，可以实现连续空间寻址。NorFlash 的特点是芯片内执行 (XIP, eXecute In Place)，这样应用程序可以直接在 Flash 中运行，不必再把代码读到系统 RAM 中。NorFlash 的传输效率很高，在 1~4MB 的小容量时具有很高的成本效益，但是很低的写入和擦除速度影响了它的性能。

SEP4020 采用 SST 的 39VF1601 NorFlash，支持从外部 NorFlash 启动。

2.2 NorFlash 基本特性

1、读写特性

读：可以实现字节寻址

写：不能字节寻址，须按页或块写入

2、UB4020MBT 上 NorFlash 地址

起始地址：CSA, 0x20000000

结束地址：2MB, 0x20200000

3、NorFlash 启动系统零地址映射

CSA, NorFlash, 0x20000000

2.3 寄存器介绍

EMI 模块的基址：0x11000000

寄存器名称	偏移地址	复位值	功能描述
CSACONF	0x000	0x08656551	CSA参数配置寄存器

配置NorFlash的参数都在该寄存器中，详细请参考《SEP4020用户手册》。

三、 NorFlash 模块实现原理

3.1 NorFlash 硬件实现原理

3.1.1 接口定义

NorFlash 接口定义：

信号名称	定义	备注
A[0..21]	地址总线	连接A[0..19]对应：16Mbit 连接A[0..20]对应：32Mbit 连接A[0..21]对应：64Mbit
D[0..15]	数据总线	
nCSA	片选信号	
nOE	读使能信号	
nWE	写使能信号	
nRESET	系统复位信号	
nWP	写保护信号	低电平时禁止写操作

3.1.2 NorFlash 原理图

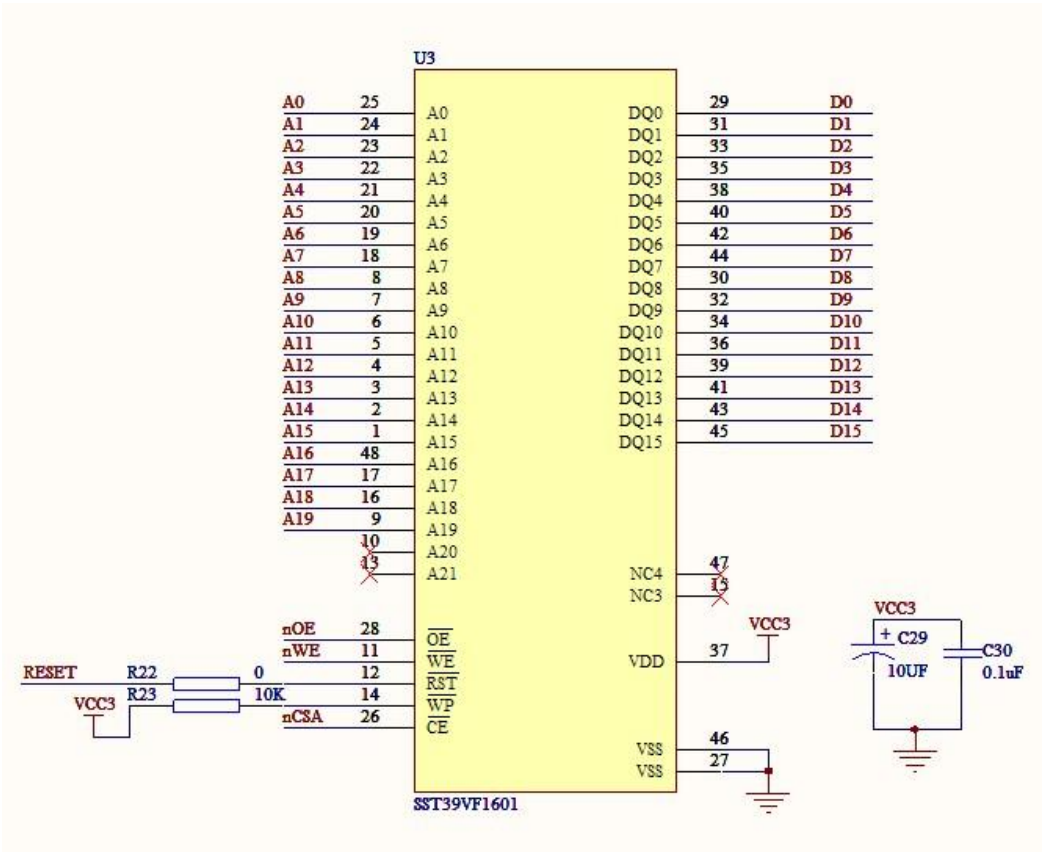


图 2 NorFlash 接口电路

3.1.3 简单设计注意事项

对于一般的处理器而言，由于NorFlash省略了A0地址线，A0管脚实际上是地址的A1，因此需要将CPU的A1接NorFlash的A0。但是对于SEP4020而言，在将存储器接口配置为16位的时候，SEP4020的A0已经是地址A1，因此只需要将SEP4020的A0连接NorFlash的A0。软件处理上，与其它处理器并无二致，只需要在写命令的时候注意NorFlash的“伪地址”。

3.2 NorFlash 软件实现原理

3.2.1 包含的头文件

```
#include <stdio.h> //标准输入输出库文件
#include "norflash.h" //NorFlash 驱动头文件
#include "intc.h" //中断头文件
#include "sep4020.h" //寄存器定义以及类型定义的头文件
```

3.2.2 主要函数介绍

void BlocPrect(void)

作用：初始化PORTG以解除对第0块的保护

参数：无

返回值：无

int EraseEntireChip(void)

作用：擦除整块Flash

参数：无

返回值：成功则返回True，失败则返回False

int CheckSST39VF160X(void)

作用：读取Flash型号，

参数：无

返回值：成功则返回型号，否则False

void CfiQuery(WORD *Src)

作用：从Flash中读取CFI信息

参数：字符数组指针，用于保存CFI信息

返回值：无

int ProgramOneBlock (WORD *Src, Uint32 Dst)

作用：编程一个块

参数：源地址Src, 目的地址Dst

返回值：成功则返回True，失败返回False

int ProgramOneWord (WORD *SrcWord, Uint32 Dst)

作用：编程一个字，仅供ProgramOneBlock调用？

参数：源地址SrcWorld, 目的地址 Dst。

返回值：成功则返回True，失败返回False。

int CheckToggleReady (Uint32 Dst)

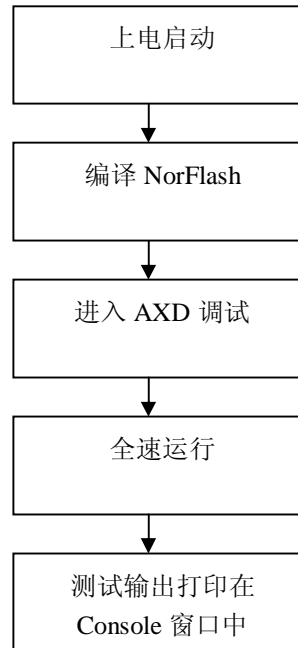
作用：检查Flash中偏移为DST的地址处是否正在进行写或者擦除操作

参数：Dst目的地址

返回值：成功则返回True，失败返回False。

四、 NorFlash 模块测试流程及结果说明

4.1 NorFlash 测试流程图



4.2 NorFlash 测试结果说明

Erase Entire Chip OK!!

The type is SST_39VF1601!!!

CFI read ok

81 82 89 1 7 0 0 0 0 0 0

1 Block Program successfully !

2 Block Program successfully !

3 Block Program successfully !

4 Block Program successfully !

5 Block Program successfully !

6 Block Program successfully !

7 Block Program successfully !

8 Block Program successfully !

9 Block Program successfully !

10 Block Program successfully !

11 Block Program successfully !

12 Block Program successfully !

13 Block Program successfully !

14 Block Program successfully !

15 Block Program successfully !

16 Block Program successfully !

17 Block Program successfully !

18 Block Program successfully !

19 Block Program successfully !

20 Block Program successfully !
21 Block Program successfully !
22 Block Program successfully !
23 Block Program successfully !
24 Block Program successfully !
25 Block Program successfully !
26 Block Program successfully !
27 Block Program successfully !
28 Block Program successfully !
29 Block Program successfully !
30 Block Program successfully !
31 Block Program successfully !
32 Block Program successfully !
0xAAAA is writed to 2Mflash!
NorFlash test ok!

五、 其他注意事项

对于 NorFlash 而言，向上提供某个固定大小的读写操作函数即可。但是在写入 Flash 之前，必须先擦除该块为全“1”，所以驱动必须提供读、写和擦除的这几个最主要的函数集。此外，在对 NorFlash 进行操作后必须保证 Flash 处于“空闲状态”下，才能保证对其的操作成功。所以必须再添加上其他与 NorFlash 状态操作相关的其他函数