

第六章 Nand Flash 控制器

6.1 概述

目前,Nor Flash 价格较高, 而 SDRAM 和 Nand Flash 存储器相对经济, 这样促使一些用户在 NAND Flash 上执行启动代码, 在 SDRAM 上执行主程序。

s3c2440A 的驱动代码可以在外部的 NAND Flash 存储器上被执行。为了支持 NAND Flash 的 boot loader, s3c2440A 配备了一个内部的 SRAM 缓冲器名为“Steppingstone”。启动时, NAND Flash 上的前 4KByte 字节将被装载到 Steppingstone 上并且装载到 Steppingstone 上的启动代码会被执行。

一般情况下, 启动代码会拷贝 NAND Flash 上的内容到 SDRAM。使用硬件的 ECC, NAND Flash 的数据被检查。在完成拷贝的基础上, 主程序将在 SDRAM 上被执行。

6.2 特性

- (1) **自动启动:** 启动代码在重启时被传输到 4kbytes 的 Steppingstone 上。传输后代码会在 Steppingstone 上被执行
- (2) **NAND Flash 存储器接口:** 支持 256 字、512 字节、1000 字和 2000Byte 页
- (3) **软件模式:** 用户可以直接访问 NAND Flash, 例如这个特性可以被用于对 NADN Flash 存储器的读/擦除/编程。
- (4) **接口:** 8/16 微的 NADN Flash 存储器接口总线
- (5) **硬件 ECC 生成,** 检测和指示 (软件纠错)
- (6) **SFR 接口:**支持小端模式, 对于数据和 ECC 数据寄存器的字节/半字/字访问, 对于其他寄存器的字访问。
- (7) **Steppingstone 接口:** 支持大小端, 字节/半字/字访问
- (8) **Steppingstone 4kB 内部 SRAM 缓冲器**可以在 NAND Flash 启动后被用于其他目的。

6.2.1 模块图

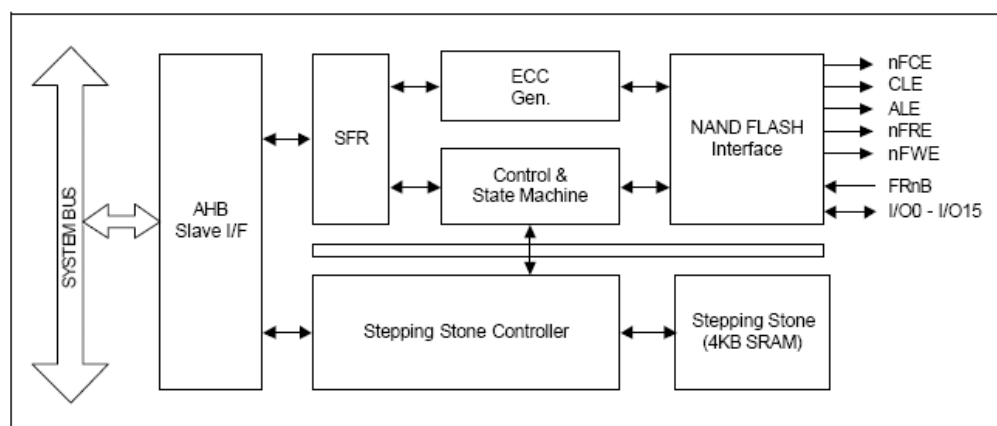


Figure 6-1. NAND Flash Controller Block Diagram

6.2.2 Boot loader 功能

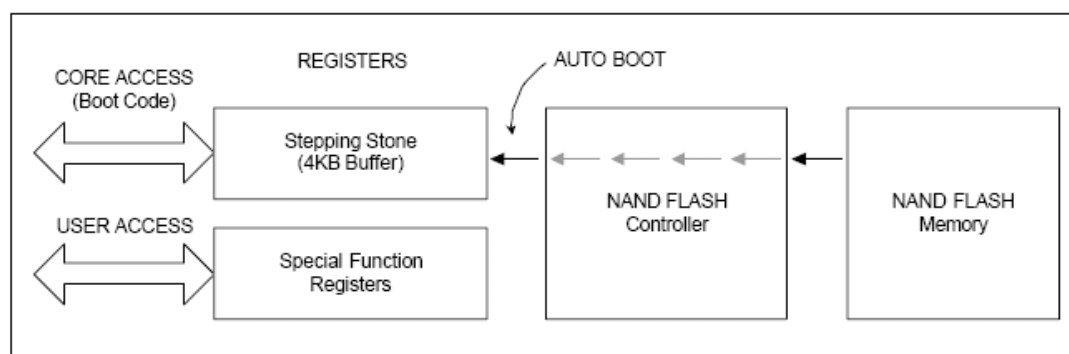


Figure 6-2. NAND Flash Controller Boot Loader Block Diagram

在重启期间，NAND Flash控制器通过引脚状态得到连接NAND Flash的信息 (NCON(Adv flash)，GPG13(页大小)，GPG14(地址周期)，GPG15(总线宽度) - 参考引脚配置)。在上电或重启以后，NAND Flash控制器自动的装载 4KB的boot loader代码。在装载boot loader代码后，其在steppingstone中被执行。

注：在自动重启期间，ECC没有检查，因为NAND Flash的前 4KB数据一般认为没有位错误。

6.2.3 引脚配置

OM[1:0]=00: 使能NAND Flash存储器启动

NCON: NAND Flash存储器选择

0: 普通NAND Flash (256 字/512 字节页大小, 3/4 地址周期)

1: 先进NAND Flash (1K字/2K字节页大小, 4/5 地址周期)

GPG13: NAND Flash存储器页容量选择

0: 页=256 字 (NCON=0) 或页=1K字 (NCON=1)

1: 页=512 字节 (NCON=0) 或页=2K字节(NCON=1)

GPG14: NAND Flash存储器地址周期选择

0: 3 个地址周期 (NCON=0) 或 4 个地址周期 (NCON=1)

1: 4 个地址周期 (NCON=0) 或 5 个地址周期 (NCON=1)

GPG15 NAND Flash存储器总线宽度选择

0: 8 位总线宽度

1: 16 位总线宽度

注: NCON,GPG[15:13]引脚配置在重启期间被?

在通常状态下, 这些引脚必须被设置为输入以至于当通过软件方式进入睡眠模式或异常

状态时, 引脚状态不会被改变。

NAND Flash 存储器配置表

NCON0	GPG13	GPG14	GPG15
0: Normal NAND	0: 256Words	0: 3-Addr	0:8-bit bus width
	1: 512Bytes	1: 4-Addr	
1: Advance NAND	0: 1Kwords	0: 4-Addr	1: 16-bit bus width
	1: 2Kbytes	1: 5-Addr	

注:

例: NAND Flash配置设置

部件	页大小/总大小	NCON0	GPG13	GPG14	GPG15
K9S1208V0M-xxxx	512Byte/512Mbit	0	1	1	0
K9K2G16U0M-xxxx	1KW / 2Gbit	1	0	1	1

6.2.4 NAND Flash 存储器时序

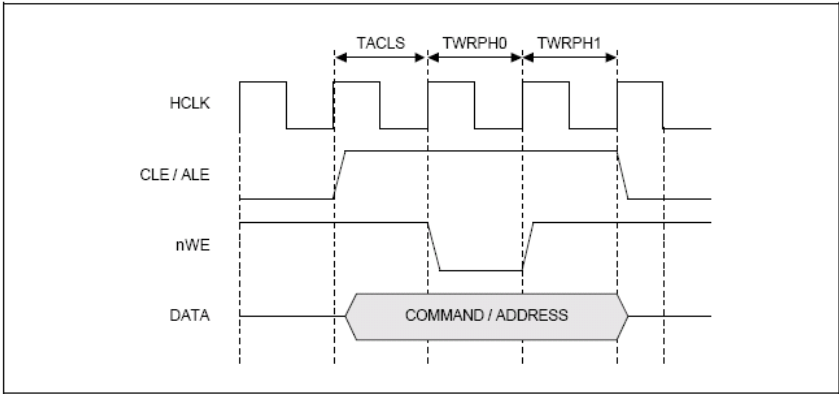


Figure 6-3. CLE & ALE Timing (TACLS=1, TWRPH0=0, TWRPH1=0)

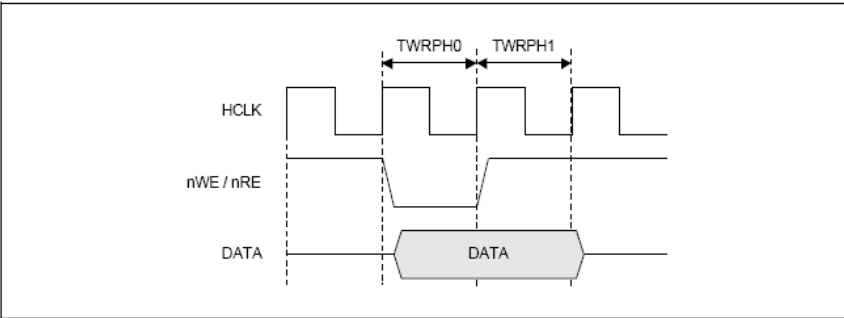


Figure 6-4. nWE & nRE Timing (TWRPH0=0, TWRPH1=0)

6.2.5 软件模式

s3c2440A仅支持软件模式访问。使用此模式，你可以完整的访问NAND Flash。NAND Flash控制器支持NAND Flash存储器的直接访问接口。

- (1) 写命令寄存器=NAND Flash存储器命令周期
- (2) 写地址寄存器=NAND Flash存储器地址周期
- (3) 写数据寄存器=写数据到NAND Flash（写周期）
- (4) 读数据寄存器=从NAND Flash读数据（读周期）
- (5) 读主ECC寄存器和空闲ECC寄存器=从NAND Flash存储器读数据

注：在软件模式中，你必须用查询或中断来检测RnB状态输入引脚。

6.2.6 数据寄存器配置

- (1) 16 位NAND Flash存储器接口

A.字访问

寄存器	大小端	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFDATA	小端	2 nd I/O[15:8]	2 nd I/O[7:0]	1 st I/O[15:8]	1 st I/O[7:0]
NFDATA	大端	1 st I/O[15:8]	1 st I/O[7:0]	2 nd I/O[15:8]	2 nd I/O[7:0]

B.半字访问

寄存器	大小端	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFDATA	大/小端	无效值	无效值	1 st I/O[15:8]	1 st I/O[7:0]

- (2) 8 位 NAND Flash 存储器接口

A.字访问

寄存器	大小端	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFDATA	小端	4 th I/O[7:0]	3 rd I/O[7:0]	2 nd I/O[7:0]	1 st I/O[7:0]
NFDATA	大端	1 st I/O[7:0]	2 nd I/O[7:0]	3 rd I/O[7:0]	4 th I/O[7:0]

B.半字访问

寄存器	大小端	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFDATA	小端	无效值	无效值	2 nd I/O[7:0]	1 st I/O[7:0]
NFDATA	大端	无效值	无效值	1 st I/O[7:0]	2 nd I/O[7:0]

C.字节访问

寄存器	大小端	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFDATA	大/小端	无效值	无效值	无效值	1 st I/O[7:0]

6.2.7 Steppingstone (4k 字节 SRAM)

NAND Flash 控制器使用 NAND Flash 作为启动缓冲器且你可以将这个区域用作其他用途。

6.2.8 ECC（错误纠错码）

NAND Flash 控制器包括 4 个 ECC(错误纠错码)模块。两个模块（一个用于 data[7:0],一个用于 data[15:8]）可以被用于（上限）2048bytes 的 ECC 奇偶码的生成，另外两个模块（一个用于 data[7:0],一个用于 data[15:8]）可以被用于（上限）16bytes 的 ECC 奇偶校验码的生

成。

-28 位 ECC 奇偶码=22 位行奇偶+6 位列奇偶

-14 位 ECC 奇偶码=8 位行奇偶+6 位列奇偶

2048 字节 ECC 奇偶码分配表

	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
MECCn_0	P64	P64'	P32	P32'	P16	P16'	P8	P8'
MECCn_1	P1024	P1024'	P512	P512'	P256	P256'	P128	P128'
MECCn_2	P4	P4'	P2	P2'	P1	P1'	P2048	P2048'
MECCn_3	P8192	P8192'	P4096	P4096'	—	—	—	—

16 字节ECC奇偶码分配表

	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
SECCn_0	P16	P16'	P8	P8'	P4	P4'	P2	P2'
SECCn_1	P1	P1'	P64	P64'	P32	P32'	-	-

6.2.8.1 ECC 模块特点

ECC 生成由控制寄存器的 ECC 锁位（主 ECC 锁、空闲 ECC 锁）来控制。

6.2.8.2 ECC 寄存器配置（大小端）

（1）16 位 NAND Flash 存储器接口

寄存器	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFMECCD0	2 nd ECC for I/O[15:8]	2 nd ECC for I/O[7:0]	1 st ECC for I/O[15:8]	1 st ECC for I/O[7:0]
NFMECCD1	4 th ECC for I/O[15:8]	4 th ECC for I/O[7:0]	3 rd ECC for I/O[15:8]	3 rd ECC for I/O[7:0]

寄存器	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFSECCD	2 nd ECC for I/O[15:8]	2 nd ECC for I/O[7:0]	1 st ECC for I/O[15:8]	1 st ECC for I/O[7:0]

（2）8 位 NAND Flash 存储器接口

寄存器	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFMECCD0	-	2 nd ECC for I/O[7:0]	-	1 st ECC for I/O[7:0]
NFMECCD1	-	4 th ECC for I/O[7:0]	-	3 rd ECC for I/O[7:0]

寄存器	位[31:24]	位[23:16]	位[15:8]	位[7:0]
NFSECCD	-	2 nd ECC for I/O[7:0]	-	1 st ECC for I/O[7:0]

6.2.8.3 ECC 编程指引

（1）在软件模式下，ECC 模块为所有的读写数据生成 ECC 奇偶码。因此你通过写 InitECC 位（NFCONT[4]）置 1 来重置 ECC 的值，且在读写数据之前清 0 theMainECCLock 位（NFCONT[15]）。

（2）不管读还是写数据，ECC 模块在寄存器 NFMECC0/1 上生成 ECC 奇偶码。

（3）在你完整的读写一页后（不包括空闲区域数据），theMainECCLock 位置 1（锁定）。ECC

奇偶码被锁定且 ECC 状态寄存器的值不会被改变。

（4）生成空闲区域 ECC 奇偶码，清 0（非锁定）SpareECCLock 位（NFCONT[6]）

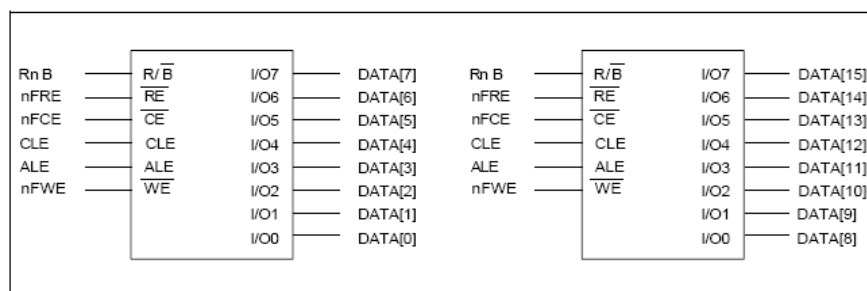


Figure 6-7. Two 8-bit NAND Flash Memory Interface

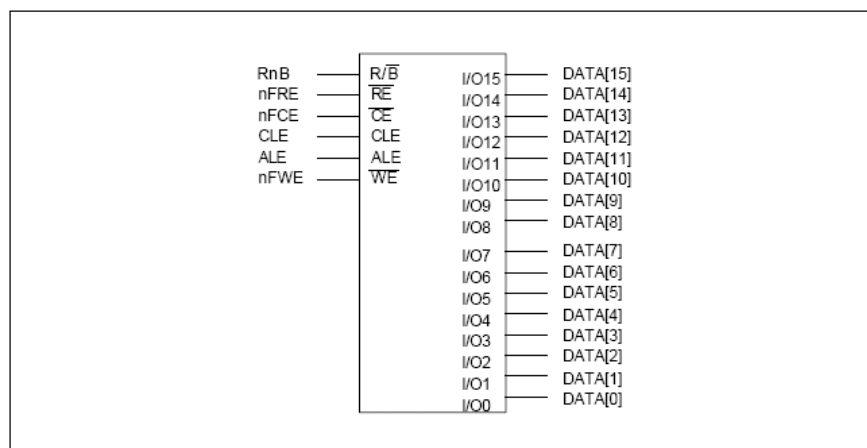


Figure 6-8. A 16-bit NAND Flash Memory Interface

6.3 NADN Flash 寄存器

- (1) NADN Flash 配置寄存器 (NFCONF)
- (2) NADN Flash 控制寄存器 (NFCONT)
- (3) NADN Flash 命令寄存器 (NFCMMD)
- (4) NADN Flash 地址寄存器(NFADDR)
- (5) NADN Flash 数据寄存器 (NFDATA)
- (6) NADN Flash 主数据区域 ECC 寄存器 (NFMECCD0/1)
- (7) NADN Flash 空闲区域 ECC 寄存器 (NFSECCD)
- (8) NADN Flash 操作状态寄存器(NFSTAT)
- (9) NADN Flash ECC0/1 状态寄存器 (NFESTAT0/1)
- (10) NADN Flash 主数据区域ECC状态寄存器 (NFMECC)
- (11) NADN Flash 空闲区域ECC状态寄存器 (NFSECC)
- (12) NADN Flash 块地址寄存器(NFSBLK &NFEBLK)

6.3.1 NADN Flash 配置寄存器

NAND FLASH CONFIGURATION REGISTER (NFCONF)

寄存器	地址	读写	描述	复位值
NFCONF	0x4E000000	R/W	NAND Flash配置寄存器	0x0000100X

NFCONF	位	描述	初始值
保留	[15:14]	保留	-
TACLS	[13:12]	CLE & ALE duration setting value (0~3) Duration = HCLK x TACLS	01
保留	[11]	保留	0
TWRPH0	[10:8]	TWRPH0 duration setting value (0~7) Duration = HCLK x (TWRPH0 + 1)	000
保留	[7]	保留	0
TWRPH1	[6:4]	TWRPH1 duration setting value (0~7) Duration = HCLK x (TWRPH1 + 1)	000
AdvFlash (只读)	[3]	Advance NAND flash memory for auto-booting 0: Support 256 or 512 byte/page NAND flash memory 1: Support 1024 or 2048 byte/page NAND flash memory This bit is determined by NCON0 pin status during reset and wake-up from sleep mode.	H/W Set (NCON0)
PageSize (只读)	[2]	NAND flash memory page size for auto-booting AdvFlash PageSize When AdvFlash is 0, 0: 256 Word/page, 1: 512 Bytes/page When AdvFlash is 1, 0: 1024 Word/page, 1: 2048 Bytes/page This bit is determined by GPG13 pin status during reset and wake-up from sleep mode. After reset, the GPG13 can be used as general I/O port or External interrupt.	H/W Set (GPG13)
AddrCycle (只读)	[1]	NAND flash memory Address cycle for auto-booting AdvFlash AddrCycle When AdvFlash is 0, 0: 3 address cycle 1: 4 address cycle When AdvFlash is 1, 0: 4 address cycle 1: 5 address cycle This bit is determined by GPG14pin status during reset and wake-up from sleep mode. After reset, the GPG14can be used as general I/O port or External interrupt.	H/W Set (GPG14)
BusWidth (读写)	[0]	NAND Flash Memory I/O bus width for auto-booting and general access. 0: 8-bit bus 1: 16-bit bus This bit is determined by GPG15 pin status during reset and wake-up from sleep mode. After reset, the GPG15 can be used as general I/O port or External interrupt. This bit can be changed by software.	H/W Set (GPG15)

6.3.2 控制寄存器

CONTROL REGISTER (NFCONT)

寄存器	地址	读写	描述	复位值
NFCONT	0x4E000004	R/W	NAND Flash控制寄存器	0x0384

NFCONT	位	描述	初始值
保留	[15:14]	保留	0
Lock-tight	[13]	Lock-tight configuration 0: Disable lock-tight 1: Enable lock-tight, Once this bit is set to 1, you cannot clear. Only reset or wake up from sleep mode can make this bit disable (cannot cleared by software). When it is set to 1, the area setting in NFSBLK (0x4E000038) to NFEBLK (0x4E00003C)-1 is unlocked, and except this area, write or erase command will be invalid and only read command is valid. When you try to write or erase locked area, the illegal a	0