

请注意赛普拉斯已正式并入英飞凌科技公司。

此封面页之后的文件标注有“赛普拉斯”的文件即该产品为此公司最初开发的。请注意作为英飞凌产品组合的部分,英飞凌将继续为新的及现有客户提供该产品。

文件内容的连续性

事实是英飞凌提供如下产品作为英飞凌产品组合的部分不会带来对于此文件的任何变更。未来的变更将在恰当的时候发生,且任何变更将在历史页面记录。

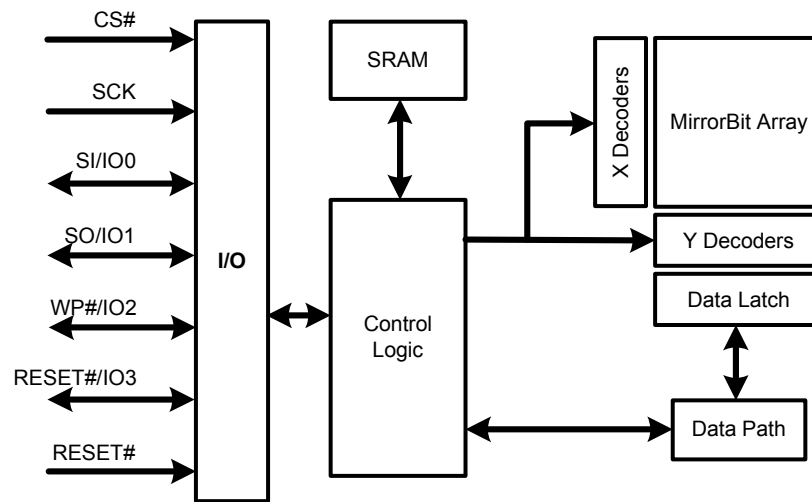
订购零件编号的连续性

英飞凌继续支持现有零件编号的使用。下单时请继续使用数据表中的订购零件编号。

特性

- **具有多个 I/O 的串行外设接口 (SPI)**
 - 支持 SPI 时钟极性以及相位模式 0 和 3
 - 具有双倍数据速率 (DDR) 选项
 - 具有扩展地址: 24 或 32 位地址选项
 - 串行命令子集和封装与 S25FL1-K、S25FL-P 以及 S25FL-S SPI 系列的兼容
 - 多个 I/O 命令子集和封装与 S25FL1-KS25FL-P 以及 S25FL-S SPI 系列的兼容
- **读取**
 - 命令: 普通、快速、双线输出、双线 I/O、四线输出、四线 I/O、DDR 四线 I/O
 - 模式: 回卷突发、持续突发 (XIP)、QPI
 - 串行闪存可发现参数 (SFDP) 和通用闪存接口 (CFI) 用于配置信息
- **编程**
 - 256 或 512 字节页编程缓冲区
 - 编程挂起和恢复
 - 自动 ECC — 支持单比特错误纠正的内部硬件错误纠正码生成
- **擦除**
 - 混合扇区选项
 - 包括 8 个大小为 4 KB 扇区的物理组, 位于顶部或底部地址空间的扇区大小为 32 KB, 其余扇区的大小均为 64 KB。
 - 统一扇区选项
 - 大小统一为 64 KB 或 256 KB 的数据块, 与容量更大的器件以及将来使用器件的软件相兼容
 - 擦除挂起和恢复
 - 擦除状态的评估
- **耐久性周期**
 - 最少具有 10 万次编程 / 擦除周期
- **数据保留时间**
 - 最少 20 年数据保持时间
- **安全功能**
 - 1024 字节的一次性编程 (OTP) 阵列
 - 块保护:
 - 通过使用状态寄存器位防止对扇区内的连续范围进行编程或擦除操作。
 - 硬件和软件控制选项
 - 高级扇区保护 (ASP)
 - 由启动码或密码控制的单个扇区保护
 - 读访问的密码控制选项
- **技术**
 - 带 Eclipse™ 架构的赛普拉斯 65 nm MirrorBit® 技术
- **带 CMOS I/O 的单个供电电压**
 - 1.7 V 到 2.0 V
- **温度范围**
 - 工业级 (-40 °C 到 +85 °C)
 - 扩展的工业级 (-40 °C 到 +105 °C)
 - 扩展范围 (-40 °C 到 +125 °C)
 - 汽车级, AEC-Q100 等级 3 (-40 °C 到 +85 °C)
 - 汽车级, AEC-Q100 等级 2 (-40 °C 到 +105 °C)
 - 汽车级, AEC-Q100 等级 1 (-40 °C 到 +125 °C)
- **封装类型 (所有均无铅)**
 - 8-SOIC 208 mil (SOC008)
 - LGA 5x6 mm (W9A008)
 - BGA-24 6 × 8 mm
 - 5 × 5 ball (FAB024) 封装

逻辑框图



性能总结

最大读取速度

命令	时钟速率 (MHz)	MB/s
读取	50	6.25
快速读取	133	16.5
双线读取	133	33
四线读取	133	66
DDR四线I/O读取	80	80

典型编程和擦除速率

操作	KB/s
页编程 (256 字节的页编程缓冲区)	712
页编程 (512 字节的页编程缓冲区)	1080
4 KB典型扇区擦除 (混合扇区选项)	16
64 KB扇区擦除	275
256 KB扇区擦除	275

典型的电流消耗, -40°C到+85°C

操作	电流 (mA)
串行读取 (时钟频率为50 MHz)	10
串行读取 (时钟频率为133 MHz)	22
四线读取 (时钟频率为133 MHz)	60
四线DDR读取 (时钟频率为80 MHz)	70
编程	60
擦除	60
待机	0.025
深度掉电	0.006

目录

特性	1	9. 地址空间映射	44
逻辑框图	2	9.1 概述	44
性能总结	3	9.2 闪存存储器阵列	44
1. 概述	5	9.3 ID-CFI地址空间	46
1.1 通用说明	5	9.4 JEDEC JESD216串行闪存可发现参数 (SFDP) 空间 ..	46
1.2 移植时注意事项	5	9.5 OTP地址空间	47
1.3 其他资源	7	9.6 寄存器	48
硬件接口		10. 数据保护	64
2. 支持多输入/输出的串行外设接口 (SPI-MIO)	8	10.1 安全硅区域 (OTP)	64
3. 信号说明	8	10.2 写入使能命令	64
3.1 输入/输出简介	8	10.3 块保护	65
3.2 多路输入/输出 (MIO)	9	10.4 高级扇区保护	66
3.3 串行时钟 (SCK)	9	10.5 建议的保护流程	71
3.4 C芯片选择 (CS#)	9	11. 命令	72
3.5 串行输入 (SI) / IO0	9	11.1 命令集汇总	73
3.6 串行输出 (SO) / IO1	9	11.2 标识命令	79
3.7 写保护 (WP#) / IO2	9	11.3 寄存器访问命令	82
3.8 IO3_RESET#	10	11.4 读取闪存阵列命令	93
3.9 RESET#	10	11.5 编程闪存阵列命令	101
3.10 供电电压 (VCC)	10	11.6 擦除闪存阵列命令	104
3.11 供电与信号接地 (V _{SS})	10	11.7 一次性编程阵列命令	111
3.12 无连接 (NC)	10	11.8 高级扇区保护命令	111
3.13 预留供将来使用 (RFU)	10	11.9 复位命令	118
3.14 请勿使用 (DNU)	11	11.10DPD命令	119
3.15 系统框图	11	12. 数据完整性	121
4. 信号协议	13	12.1 擦除耐久性	121
4.1 SPI时钟模式	13	12.2 数据保留时间	121
4.2 命令协议	14	13. 软件接口参考	122
4.3 接口状态	18	13.1 OTP存储空间地址映射	122
4.4 接口上的配置寄存器影响	22	13.2 器件ID和通用闪存接口 (ID-CFI) 地址映射—标准 ..	122
4.5 数据保护	22	13.3 串行闪存可发现参数 (SFDP) 地址映射	128
5. 电气规范	23	13.4 出厂状态	140
5.1 最大绝对额定值	23	14. 订购器件型号	141
5.2 闩锁特性	23	15. 联系赛普拉斯	142
5.3 热阻	23	16. 术语表	143
5.4 工作范围	23	16. 文档修订记录页	144
5.5 上电和掉电	24	销售、解决方案和法律信息	145
5.6 直流特性	26	全球销售和 Design Support	145
6. 时序规范	30	产品	145
6.1 切换波形的关键	30	PSoC®解决方案	145
6.2 交流测试条件	30	赛普拉斯开发者社区	145
6.3 复位	31	技术支持	145
6.4 SDR交流特性	34		
6.5 DDR交流特性	37		
7. 嵌入式算法性能表	39		
8. 物理接口	40		
8.1 连接图	40		
8.2 物理框图	41		
软件接口			

1. 概述

1.1 通用说明

赛普拉斯 FS-S 系列器件系列是非易失性闪存存储器产品，它们采用了以下技术：

- MirrorBit 技术 — 将两个数据位存储在每个存储器阵列晶体管內
- Eclipse 架构 — 显著提高了编程和擦除性能
- 65 nm 光刻技术

FS-S 系列通过串行外设接口（SPI）连接到主机系统。该产品支持传统 SPI 的一比特串行输入和输出（即单线 I/O 或 SIO）、可选的两比特（即双线 I/O 或 DIO）、四比特宽（四线 I/O 或 QIO）以及四线外设接口（QPI）命令。另外，它还为 QIO 和 QPI 提供了双倍数据速率（DDR）命令支持，在时钟的双边沿上传送地址和读取数据。

FS-S Eclipse 架构具有一个页编程缓冲区，从而在一个操作中最多能够编程 512 个字节，与上一代的 SPI 编程或擦除算法相比，它可以有效加快编程和擦除速度。

从闪存存储器直接执行代码的操作通常被称为 Execute-In-Place（芯片内执行）或 XIP。FS-S 系列器件支持更高的时钟频率和四线或 DDR 四线命令，因此指令读取的传输率能够等于或超过传统的并行接口、异步、NOR 闪存存储器，同时可明显降低信号的计数。

FS-S 系列产品为各种移动或嵌入式应用提供了大容量、灵活性和速度快等性能。该产品系列为空间、信号连接，功耗有限的系统提供了一个理想的解决方案。最适合将它们用于输送代码到 RAM，直接执行代码（XIP）和存储可重新编程的数据。

1.2 移植时注意事项

1.2.1 特性比较

FS-S 系列的命令集以及封装与前代 FL-S 和 FL-P 系列相兼容。但电源和接口电压的额定值为 1.8 V。

表 1. 赛普拉斯 SPI 系列比较

参数	FS-S	FS-S	FL-S	FL-P
技术节点	65 nm	65 nm	65 nm	90 nm
架构	MirrorBit® Eclipse™	MirrorBit® Eclipse™	MirrorBit® Eclipse™	MirrorBit®
出厂日期	量产中	2H2015	生产中	生产中
容量	128 Mb、256 Mb、512 MB	64 Mb	128 Mb、256 Mb、512 Mb	32 Mb - 256 Mb
总线宽度	x1、x2、x4	x1、x2、x4	x1、x2、x4	x1、x2、x4
供电电压	1.7 V - 2.0 V	1.7 V - 2.0 V	2.7 V - 3.6 V/1.65 V - 3.6 V V _{IO}	2.7 V - 3.6 V
正常读取速度（SDR）	6 MB/s（50 MHz）	6 MB/s（50 MHz）	6 MB/s（50 MHz）	5 MB/s（40 MHz）
快速读取速度（SDR）	16.5 MB/s（133 MHz）	16.5 MB/s（133 MHz）	16.5 MB/s（133 MHz）	13 MB/s（104 MHz）
双线读取速度（SDR）	33 MB/s（133 MHz）	33 MB/s（133 MHz）	26 MB/s（104 MHz）	20 MB/s（80 MHz）
四线读取速度（SDR）	66 MB/s（133 MHz）	66 MB/s（133 MHz）	52 MB/s（104 MHz）	40 MB/s（80 MHz）
四线读取速度（DDR）	80 MB/s（80 MHz）	80 Mb/s（80 Mhz）	66 MB/s（66 MHz）	-
编程缓冲区大小	256 B/512 B	256 B/512 B	256 B/512 B	256 B
擦除扇区大小	64 KB/256 KB	64 KB/256 KB	64 KB/256 KB	64 KB/256 KB
参数扇区大小	4 KB（可选）	4 KB（可选）	4 KB（可选）	4 KB
扇区擦除速率（典型值）	500 KB/s	500 KB/s	500 KB/s	130 KB/s
页编程速率（典型值）	1.0 MB/s（256 B） 1.2 MB/s（512 B）	1.0 MB/s（256 B） 1.2 MB/s（512 B）	1.2 MB/s（256 B） 1.5 MB/s（512 B）	170 KB/s
OTP	1024 B	1024 B	1024 B	506 B
高级扇区保护	支持	支持	支持	不支持
自动启动模式	不支持	不支持	支持	不支持
擦除挂起 / 恢复	支持	支持	支持	不支持
编程挂起 / 恢复	支持	支持	支持	不支持
工作温度	-40 °C 到 +85 °C/+105 °C	-40 °C 到 +85 °C/+105 °C/ +125 °C	-40 °C 到 +85 °C/+105 °C	-40 °C 到 +85 °C/+105 °C

注意

1. FL-P 列表示 FL129P MIO SPI 器件（128 Mb 容量），FL128P 不支持 MIO、OTP 或 4KB 扇区
2. 64 KB 扇区擦除选项仅适用于容量为 128 Mb/256 Mb 的 FL-P、FL-S 和 FS-S 器件
3. 欲了解详细信息，请参考单独的数据手册。

1.2.2 与前代的区别**1.2.2.1 错误报告**

FL-K 和 FL-P 存储器没有错误状态位，因此尝试对受保护的扇区进行编程或擦除时不能设置这些位。FS-S 和 FL-S 系列没有编程和擦除操作错误报告状态位。当进行编程或擦除操作时发生内部故障，或者当尝试编程或擦除受保护的扇区时，可以设置这些位。在这些情况下，编程或擦除操作不会按该命令的请求完成。在 SR1V 中，P_ERR 或 E_ERR 位以及 WIP 位将被设置为 1，并保持为数值 1。必须发送清除状态寄存器命令，以清除各错误，并使器件返回待机状态。

1.2.2.2 安全硅区域（OTP）

FL-K 和 FL-P 的一次性编程区域的 FS-S 大小和格式（地址映射）不一样。保护 OTP 区域中每一部分的方法也不同。有关的详细信息，请参见第 64 页上的安全硅区域（OTP）。

1.2.2.3 配置寄存器的 Freeze 位

同上一代产品一样，配置寄存器 1 的冻结位 CR1V[0] 锁定块保护位（SR1NV[4:2] 和 SR1V[4:2]）的状态、TBPARM_O 位（CR1NV[2]）和 TBPROT_O 位（CR1NV[5]）。在 FS-S 和 FL-S 系列中，冻结位也会锁定配置寄存器 1 中 BPNV_O 位（CR1NV[3]）的状态以及安全芯片区域（OTP）。

1.2.2.4 扇区擦除命令

擦除 4 KB 扇区的指令只适用于 FS-S 器件地址空间顶部或底部上的 4 KB 参数扇区。

不支持擦除 8 KB 区域（两个 4 KB 扇区）的命令。

不支持擦除 32 KB 区域（8 个 4 KB 扇区）的命令。

当选择容量统一为 64 KB 的扇区的配置选项，或者使用 4 KB 参数扇区和 64 KB 统一扇区的混合配置选项时，可以对 FS-S 的 64 KB 扇区进行扇区擦除命令（SE）。当使用混合选项时，可以使用 64 KB 擦除命令来擦除与 8 个 4 KB 扇区组相邻的 32 KB 地址空间。在这种情况下，64 KB 擦除命令将擦除 64 KB 扇区（该扇区由 8 个 4 KB 扇区组部分覆盖）而不会影响 4 KB 扇区。这样可以擦除 32 KB 地址空间，而不需要擦除 4 KB 扇区。这种情况与 FL-S 系列中所实现的情况不一样。在 FL-S 系列中，可以将 64 KB 扇区擦除命令应用于包含 4 KB 扇区的 64 KB 模块，从而通过单个操作擦除整个参数扇区的模块。在 FS-S 系列中，参数扇区不会填充整个 64 KB 模块，因此只能使用 4 KB 参数扇区擦除命令（20h）来擦除各个参数扇区。

当使用 256 KB 统一逻辑扇区的配置选项时，256 KB 扇区的擦除命令将替换掉 64 KB 擦除命令。

1.2.2.5 深度掉电

在 FS-S 系列器件中，支持深度掉电（DPD）功能。

1.2.2.6 WRR 单个寄存器写操作

在某些传统的 SPI 器件中，只带有一个数据字节的写寄存器（WRR）命令会更新状态寄存器 1，并会清除配置寄存器 1 中的某些位，其中包括四线模式位。这样会引起意外退出四线模式。当提供单个数据字节时，FS-S 系列只会更新状态寄存器 1。在这种情况下，配置寄存器 1 不被修改。

1.2.2.7 不支持 Hold 输入

在某些传统的 SPI 器件中，IO3 输入可以作为 HOLD# 输入使用，从而可以挂起信息传输而不停止串行时钟。FS-S 系列不支持该功能。

1.2.2.8 不支持其他传统命令

- DDR 快速读取
- DDR 双线 I/O 读取

1.2.2.9 新功能

FS-S 系列介绍了赛普拉斯 SPI 类存储器的新特性：

- 为内核和 I/O 电压支持单 1.8 V 电源。
- 可配置初始读取延迟（虚拟周期数量），以获取更快的初始访问时间或更高的时钟速率读取命令
- QPI（QPI、4-4-4）读取模式，在该模式下，所有传输均为 4 位带宽，包含了各条指令
- 符合 JEDEC JESD216 版本 B 标准的串行闪存可发现参数（SFDP），这些参数提供了器件的特性和配置信息。

- 通过评估擦除状态命令来确定对扇区进行的最后擦除操作是否成功完成。可以使用该命令来检测由断电或其他原因导致的未完成的擦除操作。断电后，该命令会对文件系统恢复中的闪存文件系统软件具有很大帮助。
- 高级扇区保护（ASP）的永久保护。另外，选择 ASP 保护模式时，将不能对所有寄存器中的所有 OTP 配置位进行编程操作，这样所有 OTP 配置设置将为永久性的。选用 ASP 保护模式时，OTP 地址空间不受保护。冻结位（CR1V[0]）可用于保护 OTP 地址空间。

1.3 其他资源

1.3.1 赛普拉斯产品路线图链接

www.cypress.com/Flash-Roadmap

1.3.2 软件链接

www.cypress.com/software-and-drivers-cypress-flash-memory

1.3.3 应用笔记链接

www.cypress.com/cypressappnotes

硬件接口

2. 支持多输入 / 输出的串行外设接口 (SPI-MIO)

具有独立的并行控制、地址和数据信号的多个存储器件与其主机系统连接时，需要大量的信号连接和更大的封装。由于大量信号交换，因此信号连接的数量越多，功耗也越高；较大的封装尺寸会增加成本。

FS-S 系列通过 6 个信号将所有控制、地址和数据信息串行传输给主机系统，因此可以减少信号连接的数量。这样可以降低存储器封装的成本和信号交换的功耗，减少主机信号连接的数量，主机可以节省连接以供其他功能使用。

FS-S 系列使用了业界标准的单比特串行外设接口 (SPI)，同时支持可选的扩展命令，用于进行两比特 (双线) 或 4 比特 (四线) 宽的串行传输。这种多宽度接口被称为 SPI Multi-I/O 或 SPI-MIO。

3. 信号说明

3.1 输入 / 输出简介

表 2. 信号表

信号名称	类型	说明
RESET#	输入	硬件复位： 该信号为低电平时，器件复位并返回待机状态，准备接收命令。该信号具有一个内部上拉电阻；在主机系统中如果不使用该信号，可以保持它为未连接状态。
SCK	输入	串行时钟
CS#	输入	芯片选择
SI/IO0	I/O	单比特数据命令的 串行输入 或双线 / 四线命令的 IO0。
SO/IO1	I/O	单比特数据命令的 串行输出 。双线或四线命令的 IO1。
WP#/IO2	I/O	不在四线模式 (CR1V[1] = 0 和 SR1NV[7] = 1) 下时，为 写保护 。 在四线模式 (CR1V[1] = 1) 下，为 IO2 。 该信号具有一个内部上拉电阻。在主机系统中，如果不使用四线模式或不用于写保护，可以保持它为未连接状态。如果通过设置 SR1NV[7] = 1 和 CR1V[1] = 0 使能写保护，那么在 WRR 或 WRAR 命令期间，主机系统需要将 WP# 驱动为高或低电平。
IO3_RESET#	I/O	IO3： 在四线 I/O 模式下 (当配置寄存器 1 的 QUAD 位 CR1V[1] = 1 和 CS# 为低电平时)。 RESET#： (CR2V[5]=1, 并且未处于四线 I/O 模式 (CR1V[1] = 0) 时，或在四线模式下 (CR1V[1] = 1) 且 CS# 高时)。 该信号具有一个内部上拉电阻；在主机系统如果不使用四线模式或不用于 RESET#，可以保持未连接状态。
V _{CC}	电源	供电电源。
V _{SS}	电源	接地电源。
NC	未使用	未连接。 没有任何器件内部信号被连接到封装引脚，将来也不计划用于信号连接。该连接可能安全地用作印刷电路板 (PCB) 上的信号路由空间。然而，连接到 NC 的任何信号电压电平不能大于 V _{CC} 。
RFU	预留	预留供将来使用。 当前没有任何器件内部信号连接到封装引脚，但是将来可能使用。不建议将 RFU 连接到 PCB 电路板上的任何信号，以便 PCB 可以在尺寸兼容的器件中实现未来的增强型特性。
DNU	预留	请勿使用。 器件的内部信号可能被连接到封装引脚。赛普拉斯可以使用该引脚进行测试或其他用途，不应使该引脚与任何主机系统信号连接。DNU 引脚处于低电平 (V _{IL}) 时，与该引脚相关的所有功能均无效。该引脚使用了一个内部下拉电阻，在主机系统中可以保持它为未连接状态或将其连接到 V _{SS} 。请勿将该 DNU 引脚连接到 PCB 板上的任意信号。请勿将主机系统的任何信号连接到该信号。

注意：

- 使用内部上拉或下拉电阻的输入驱动电流低于 2 uA。只有在上电过程中，该电流才最大 (在 4 uS 内为 150 uA)。

3.2 多路输入 / 输出 (MIO)

传统 SPI 单比特命令（即单 IO 或 SIO）只能通过串行输入（SI）信号将主机的数据发送到存储器内。数据会通过串行输出（SO）信号从存储器内串行回送给主机。

双线或四线输入 / 输出（I/O）命令只能通过 SI/IO0 信号将指令发送给存储器。地址或数据可按双比特一组的方式通过 IO0 和 IO1 从主机发送到存储器内，也可以按照四比特（半字节）一组的方式通过 IO0、IO1、IO2 和 IO3 发送。数据同样可以按照两比特一组通过 IO0 和 IO1 从存储器回送给主机，也可以按照四位（半字节）一组通过 IO0、IO1、IO2 和 IO3 被回送。

QPI 模式允许以四位（半字节）一组的方式将所有指令、地址和数据通过 IO0、IO1、IO2 和 IO3 从主机传送到存储器内。数据同样可以按四位（半字节）一组的方式通过 IO0、IO1、IO2 和 IO3 回送到主机内。

3.3 串行时钟 (SCK)

该输入信号为 SPI 接口提供了同步参考。可以在 SCK 信号的上升沿上锁存指令、地址或数据。在 SDR 命令中和在 DDR 命令中的每一沿后，数据输出会在 SCK 的下降沿后发生变化。

3.4 C 芯片选择 (CS#)

芯片选择信号指出某条命令正在向器件或从器件传输信息，并指示存储器件的相关信号。

当 CS# 信号为高逻辑状态时，不会选择芯片，所有输入信号（Reset# 和 IO3_Reset# 除外）均被忽略，并且所有输出信号均为高阻抗。除非正在进行某个内部嵌入式操作，否则器件会处于待机模式。嵌入式操作将状态寄存器 1 正在写入位（SR1V[1]）设置为 1，直到该操作完成为止。嵌入式操作的示例包括：编程、擦除或写入寄存器（WRR）等操作。

将 CS# 输入驱动为低逻辑状态可以启用器件，使器件进入工作模式。上电后，必须等到 CS# 的下降沿到来后才能执行其他命令。

3.5 串行输入 (SI) / IO0

该输入信号用于将数据串行传输给器件。它接收指令、地址和要写入闪存的数据。各个数值在串行 SCK 时钟信号的上升沿上被锁存。

在执行双线和四线命令过程中，SI 变成 IO0（一个输入和输出），用于接收指令、地址和数据来进行编程（数值在串行 SCK 时钟信号的上升沿上被锁存），并在 SCK 下降沿上（若执行 SDR 命令）或在任意一个边沿上（若执行 DDR 命令）移出数据。

3.6 串行输出 (SO) / IO1

该输出信号用于串行传输器件中的数据。在串行 SCK 时钟信号的下降沿上移出数据。

在执行双线和四线命令过程中，SO 变成 IO1（一个输入和输出），用于接收地址和数据（数值在串行 SCK 时钟信号的上升沿上被锁存），并在 SCK 下降沿上（若执行 SDR 命令）或在任意边沿上（若执行 DDR 命令）移出数据。

3.7 写保护 (WP#) / IO2

如果 WP# 被驱动为低电平（ V_{IL} ），那么在执行 WRR 或 WRAR 命令期间以及状态寄存器 1（SR1NV[7]）的状态寄存器写禁用（SRWD_NV）位被设置为 1 时，便无法对状态寄存器 1 或与配置寄存器 1 相关的寄存器进行写入操作。在这种情况下，会忽略 WRR 命令，并忽略选择 SR1NV、SR1V、CR1NV 或 CR1V 的命令，并且不会报告任何错误。

这样可以防止块保护设置发生更改。因此，如果在执行 WRR 或 WRAR 命令过程中 WP# 为低电平且 SRWD_NV 为 1，那么存储在存储器区域（受块保护功能的保护）中的所有数据字节将被防止进行数据修改。

使能四线模式（CR1V[1] = 1）时，会禁用 WP# 的功能。当使能四线模式时，WP# 将被 IO2 输入 / 输出替换，用于接收地址和数据（数值在串行 SCK 时钟信号的上升沿上被锁存），并在 SCK 下降沿上（若执行 SDR 命令）或在任意边沿上（若执行 DDR 命令）移出数据。

WP# 具有一个内部上拉电阻；未连接时，WP# 为高电平（ V_{IH} ），如果四线模式或保护功能未使用该引脚，则在主机系统中它可以被保持为不连接状态。

3.8 IO3_RESET#

当使能四线模式（CR1V[1] = 1）时，会将 IO3 作为输入 / 输出使用，用于接收地址和数据（数值在串行 SCK 时钟信号的上升沿上被锁存），并在 SCK 下降沿上（若执行 SDR 命令）或在任意边沿上（若执行 DDR 命令）移出数据。

通过写入配置寄存器 2 的非易失性位 5（CR2V[5] = 1）来启用复位功能时，IO3_RESET# 也可用于初始化硬件复位功能。如果器件并非处于四线 I/O 模式，即 CR1V[1] = 0，或 CS# 为高电平，则只能将该输入作为 RESET# 使用。如果使能了四线 I/O 模式（CR1V[1] = 1），或通过将 CS# 设置为低电平来选择器件，则只能将 IO3/RESET# 作为 IO3 传送信息。CS# 为高电平时，IO3/RESET# 不能传送信息，只能将它作为 RESET# 输入使用。在四线模式下，如果在 CS# 为高电平时，则复位功能在该模式下仍可用。

当系统进入复位状态时，必须将 CS# 信号驱动为高电平（作为复位过程的一部分），并将 IO3_RESET# 信号驱动为低电平。CS# 变为高电平时，IO3_RESET# 输入便从 IO3 切换为 RESET# 输入。当 CS# 保持为高电平和 IO3_RESET# 信号保持为低电平（ t_{RP} ）时，会检测到复位条件。如果不需要进行复位操作，则系统必须在向存储器发送数据的操作结束时将 IO3_RESET# 和 CS# 驱动为高电平。将数据发送给主机系统后，存储器会驱动 IO3 为高电平，使之保持高电平 t_{CS} 的时长。这样可确保 IO3 / Reset 不进入悬空状态，也不会被内部或外部被动上拉电阻慢慢地上拉到高电平。因此， t_{RP} 时间结束前 IO3_RESET# 不会被视作高电平，从而也不会触发意外的复位事件。

禁用复位功能（CR2V[5] = 0）时，将不使用 IO3_RESET# 信号。

IO3_RESET# 信号具有一个内部上拉电阻；如果四线模式或复位功能未使用该输入，则在主机系统中它可以保持为不连接状态。如果主机系统将 IO3_RESET# 信号驱动为高电平，然后停止驱动该信号，则通过使用内部上拉电阻将它保持为高电平。

请注意，如果多个 SPI-MIO 存储器中任意一个工作于四线 I/O 模式，则它们不能共享 IO3_RESET# 输入，因为可能将发送给或从一个选定存储器发送的 IO3 作为第二个非选定存储器的复位信号，如果该存储器共享了同一个 IO3_RESET# 信号。

3.9 RESET#

RESET# 输入提供了一种硬件复位方法，该方法可使器件返回待机状态，准备好接收命令。如果将 RESET# 驱动为逻辑低（ V_{IL} ）的时间不小于 t_{RP} 时长，则器件将启动硬件复位过程。

在比 t_{RP} 更长的时间内，当从 V_{IH} 转换为 V_{IL} 时，RESET# 输入启动了复位操作，器件会使用与上电复位相同的方式复位寄存器状态，但没有经过 POR 期间所执行的完整复位程序。完成硬件复位程序需要 t_{RPH} 时长。当 RESET# 被拉为电平时，

RESET# 具有一个内部上拉电阻；如果不使用该引脚，则在主机系统中它可以保持为不连接状态。如果主机系统将复位信号驱动为高电平，然后停止驱动该信号，则通过使用内部上拉电阻将该信号保持为高电平。

RESET# 输入在所有封装中均不可用。器件的 RESET# 被锁定为非活动状态。

当使用 RESET#，并且不处于 QIO 或 QPI 模式时，请勿使用 IO3/RESET# 引脚。

3.10 供电电压（ V_{CC} ）

V_{DD} 是所有器件内部逻辑的供电电压。这种单个电压用于器件的所有内部功能（包括读取、编程和擦除操作）。

3.11 供电与信号接地（ V_{SS} ）

V_{SS} 器件内核、输入信号接收器和输出驱动器的公共电压开漏和接地参考。

3.12 无连接（NC）

没有任何器件内部信号被连接到封装引脚，将来也不计划用于信号连接。该连接可能安全地用作印刷电路板（PCB）上的信号路由空间。

3.13 预留供将来使用（RFU）

当前没有任何器件内部信号被连接到封装引脚，预留供将来使用。不建议将 RFU 连接到 PCB 电路板上的任何信号，以便 PCB 可以在尺寸兼容的器件中实现未来的增强型特性。

3.14 请勿使用（DNU）

器件的内部信号可能被连接到封装引脚。赛普拉斯可以使用该引脚进行测试或其他用途，不应使该引脚与任何主机系统信号连接。DNU 引脚处于低电平（ V_{IL} ）时，与该引脚相关的所有功能均无效。该引脚使用了一个内部下拉电阻，在主机系统中可以保持它为未连接状态或将其连接到 V_{SS} 。请勿将该 DNU 引脚连接到 PCB 板上的任意信号。请勿将主机系统的任何信号连接到该信号。

3.15 系统框图

图 1. SPI 总线上的总线主设备和存储器件之间的关系 — 单比特数据路径

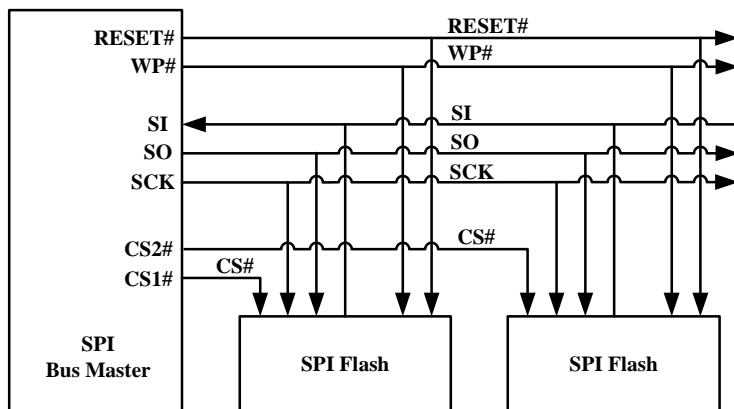


图 2. SPI 总线上的总线主设备和存储器件之间的关系 — 双比特数据路径

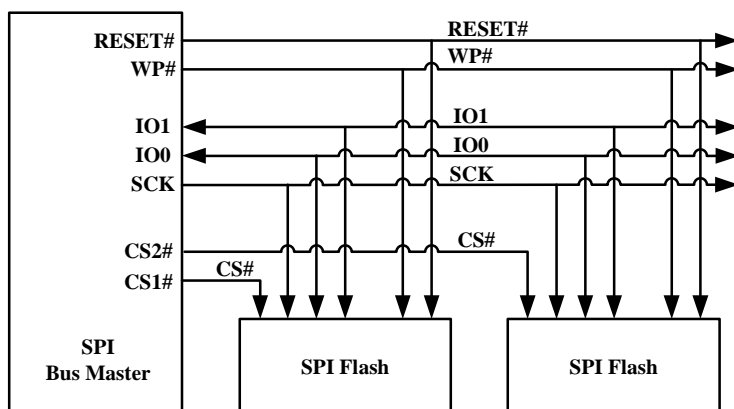


图 3. SPI 总线上的总线主设备和存储器件之间的关系 — 四比特数据路径 - 单独 RESET#

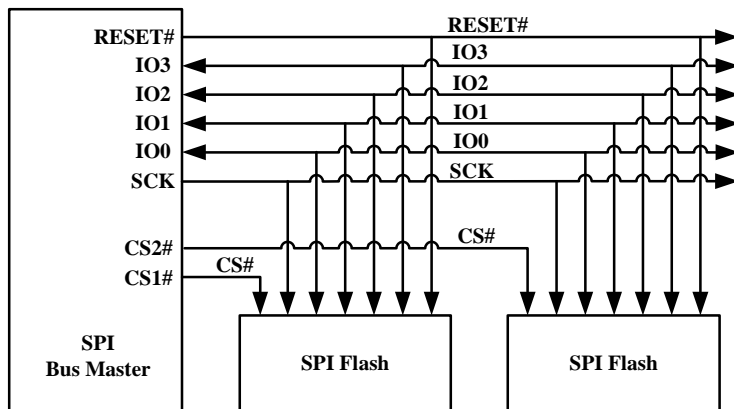
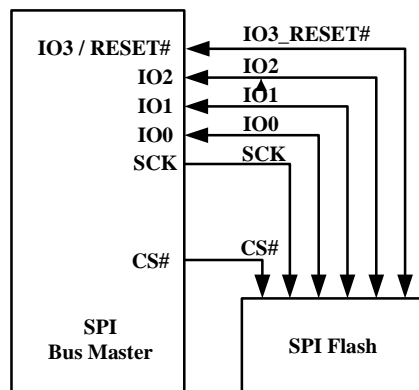


图 4. SPI 总线上的总线主设备和存储器件之间的关系 — 四比特数据路径 - I/O3_RESET#



4. 信号协议

4.1 SPI 时钟模式

4.1.1 单倍数据速率 (SDR)

通过嵌入式微控制器（总线主设备）可以将 FS-S 系列驱动到以下任何一种时钟模式。

■ **模式 0:** 时钟极性 (CPOL) = 0、时钟相位 (CPHA) = 0

■ **模式 3:** CPOL = 1、CPHA = 1

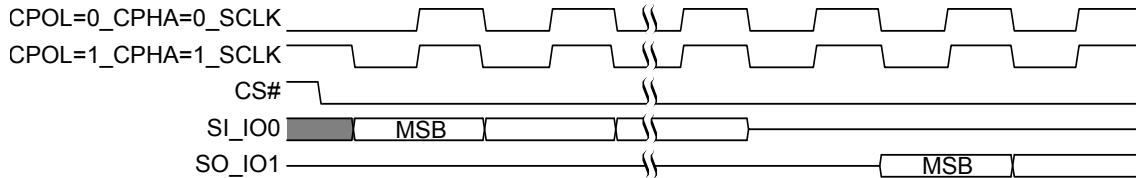
在这两种模式中，器件的输入数据总是在 SCK 信号的上升沿上被锁存，输出数据总是在 SCK 时钟信号的下降沿上可用。

两种模式间的区别主要是：当总线主设备处于待机模式并且不传输数据时的时钟极性。

■ CPOL = 0、CPHA = 0 时，SCK 将处于逻辑低状态

■ CPOL = 1、CPHA = 1 时，SCK 将处于逻辑高状态

图 5. 支持 SPI SDR 模式



整个文档其余部分的时序图通常都是通过显示 SCK 在 CS# 下降沿时的高和低电平来显示模式 0 和模式 3 的。在某些情况下，时序图可能通过显示 SCK 在 CS# 下降沿时的低电平来显示唯一的模式 0。而唯一的模式 3 的时序图只是表示在 CS# 的下降沿上时钟信号为高电平，因此模式 3 不需要从 CS# 下降沿后到 SCK 上升沿前的建立和保持时间。

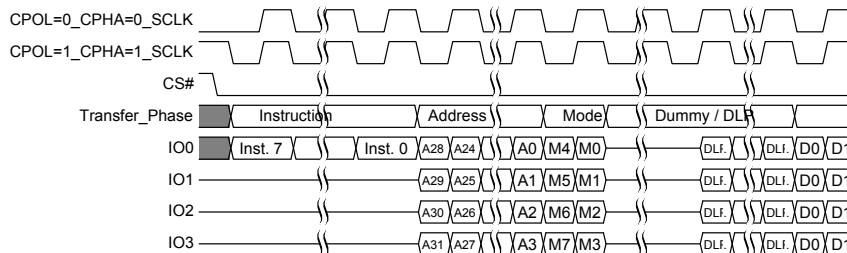
SCK 周期是从一个 SCK 下降沿到下一个 SCK 下降沿进行测量（计数）得到的。在模式 0 下，由于开始执行某个命令时 SCK 已经处于低电平状态，因此执行命令时的第一个 SCK 周期指的是从 CS# 的下降沿到第一个 SCK 下降沿。

4.1.2 双倍数据速率 (DDR)

DDR 命令也支持模式 0 和模式 3。与 SDR 命令相同，在 DDR 命令中，指令位也是在时钟的上升沿上被锁存的。但是指令后的地址和输入数据则在 SCK 的双边沿上被锁存。最后指令位在 SCK 的下降沿上被锁存之后，第一个地址位会在随后的第一个 SCK 上升沿上被锁存。第一个输出数据位则在最后访问延迟（虚拟）周期结束后的下降沿上被发送。

与 SDR 命令相同，SCK 周期也是从一个 SCK 下降沿到下一个 SCK 下降沿进行测量（计数）得到的。在模式 0 下，由于开始执行某个命令时 SCK 已经处于低电平状态，因此执行命令时的第一个 SCK 周期指的是从 CS# 的下降沿到第一个 SCK 下降沿。

图 6. 支持 SPI DDR 模式



4.2 命令协议

主机系统和 FS-S 系列存储器件系列之间都是以命令为单位形式进行通信。

所有命令都以一个 8 位指令开始，该指令用于选择信息传输类型或器件需要执行的操作。命令可能也具有一个地址、指令修饰符、延迟周期、传送到存储器，或者是从存储器传送的数据。主机系统和存储器件之间的所有指令、地址和数据信息都是连续传输的。

通过一个使用 3 个数字的数字命令法来划分命令协议，以便能参考三个命令阶段的发送宽度。

■ 指令

■ 地址和指令修饰符（连续读取模式位）

■ 数据

单比特命令只通过 SI 信号依次传输指令和地址或数据。数据会通过 SO 信号从存储器串行回送到主机内。对于单比特宽指令、单比特宽地址和修饰符、单比特数据的结构，被称为 1-1-1 命令协议。

双线输出或四线输出命令从主机端通过 SI（IO0）将地址和虚拟周期依次进行发送。数据可以按照两比特一组通过 IO0 和 IO1 从存储器回送到主机，或按照四比特（半字节）一组通过 IO0、IO1、IO2 和 IO3 回送。对于双线输出和四线输出命令，这种结构分别被称为 1-1-2 和 1-1-4 命令协议。

双线或四线输入 / 输出（I/O）命令会按照两比特一组通过 IO0 和 IO1 将主机的数据传输给存储器，或按照四比特（半字节）一组通过 IO0、IO1、IO2 和 IO3 从主机发送。数据也可以按照同样的方式回送给主机。对于双线 I/O 和四线 I/O 命令，这种结构分别被称为 1-2-2 和 1-4-4 命令协议。

FS-S 系列也支持 QPI 模式，在该模式下所有信息（包括指令、地址、修饰符和数据）都以 4 比特的宽度传输。这种形式被称为 4-4-4 命令协议。

命令的结构如下：

- 每个命令都在 CS# 的下降沿开始执行，并在 CS# 的上升沿结束。主机通过一个命令将芯片选择（CS#）信号驱动为低电平，从而能够选择器件。
- 串行时钟（SCK）标记了主机和存储器之间所传输的单个位或一组位。
- 每个命令都以一个 8 位（一个字节）的指令开始。该指令用于选择信息传输类型或需要执行的器件操作。该指令在 SCK 的上升沿上进行传送。但某些读命令被前一个读命令修改，因此该指令要从更早的命令中显示。这种形式被称为连续读取模式。器件处于连续读取模式时，由于该指令和用于初始化连续读取模式的读取命令相同，所以各个指令位不会在开始执行命令时被传输。在该模式下，命令会从读取地址开始。因此，连续读取模式会将处于相同类型的读取命令中的每个读命令内删除 8 个指令位。
- 该指令可以单独在设备的某些地址空间选择一个位置，或者跟着地址位一起选择。该指令用于确定所使用的地址空间。地址可能是 24 位或 32 位（字节边界）的地址。地址会在 SCK 的上升沿（若执行 SDR 命令）或在 SCK 双边沿上（若执行 DDR 命令）得到传送。
- 在传统的 SPI 模式下，传输指令后所有传输宽度都由已发送的指令确定。以下表示的可能是其中一种传输类型：仅通过串行输入（SI）或串行输出（SO）信号传输单比特；通过 IO0 和 IO1 信号每次（双线）传输将以双比特的形式进行发送；或通过 IO0-IO3 信号每次（四线）传输以 4 比特的形式执行。在双比特或四比特组中，最低有效位是通过 IO0 信号发送的。其他更高有效位则按照权值大小通过编号更高的 IO 信号发送。单比特或并行比特组则按照从最高有效位到最低有效位的顺序进行传输。
- 在 QPI 模式下，所有信息都按照 4 比特宽（四线传输）通过 IO0-IO3 信号进行发送。
- 双线和四线 I/O 读命令在发送地址后将发送一个指令修饰符（称为连续读取模式位），以指明下一个命令的类型与之前的命令相同，而它并不是明确的命令。这些模式位用于初始化或结束连续读取模式。因此，下一条命令只会提供新地址和模式位，并不提供指令字节。当某个命令序列中重复了相同的命令类型时，会减少发送每一个命令所需要的时间。模式位会在 SCK 的上升沿（若执行 SDR 命令）或在 SCK 双边沿上（若执行 DDR 命令）进行传送。
- 地址或模式位发送之后可以接着发送将被写入存储器的数据，可以是一个读延迟周期（数据返回给主机前）。
- 写入数据位传输会在 SCK 的上升沿（若执行 SDR 命令）或在 SCK 双边沿上（若执行 DDR 命令）锁存。
- SCK 会在读访问的延迟周期内继续进行切换。该延迟（又称为虚拟周期）可能为 0 到几个 SCK 周期。最后的读取延迟周期结束后，第一次读取的数据位会在 SCK 下降沿上被输出。第一次读取的数据位被认为是在下一个 SCK 下降沿上传输给主机的。将在 SCK 的下一个上升沿（若执行 SDR 命令）或在 SCK 的任一沿上（若执行 DDR 命令）进行下一次传输操作。
- 如果命令将读数据返回给主机，则存储器件会继续发送数据，直到主机将 CS# 信号驱动为高电平为止。结束读取数据序列中任一个传输操作后，都可以将 CS# 信号驱动为高电平。这样会终止命令。
- 命令被终止时，不会再返回数据，此时主机可将 CS# 输入信号驱动为高电平。传输单独指令的第八位或最后写数据字节的第八位后，CS# 信号必须变为高电平。这样，当位数量在 CS# 被驱动为低电平后是 8 位的整数倍时，必须将 CS# 信号驱动为高电平。如果 CS# 信号在指令或写入数据的 8 位边界后没有变为高电平，该命令将被拒绝，并且不能执行该命令。

- 所有指令、地址和模式位均被移入到器件内，并且先发送最高有效位（MSB）。数据位都被移入到器件，或从器件移出，并且要先传输 MSB。所有数据都以字节为单位进行传输，并且先发送最低地址字节。随后的数据字节按照从最低到最高字节地址的顺序（即字节地址递增）进行发送。
- 在编程、擦除或写周期（嵌入式操作）中，尝试读取闪存阵列的操作均被忽略。嵌入式操作仍继续进行而不受任何影响。执行嵌入式操作时只接受少数命令集。这些信息会在单独命令的说明部分进行介绍。
- 执行时间会因命令不同而发生变化。提供用于读取某个执行命令的状态信息的命令，可确定命令结束的时间，并且确定该命令是否被成功执行。

4.2.1 命令顺序示例

图 7. 单独指令的命令

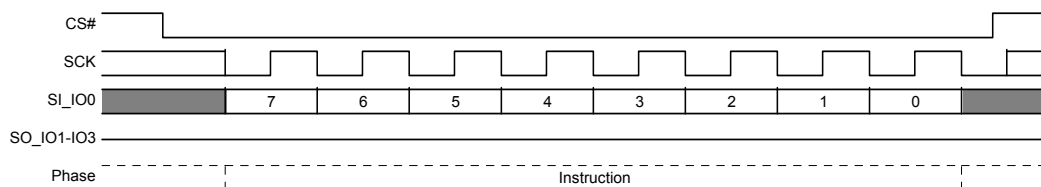


图 8. 单线输入命令

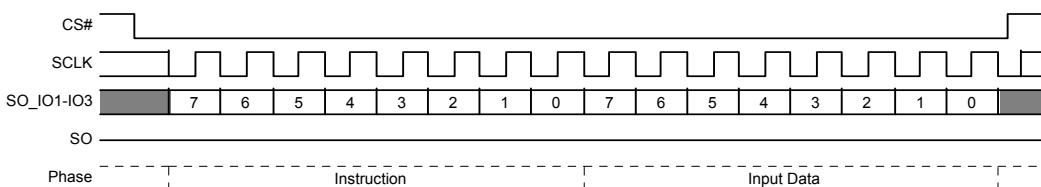


图 9. 单线输出命令（没有延迟）

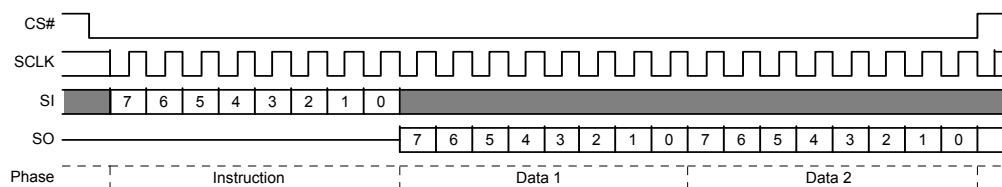


图 10. 单线 I/O 命令（有延迟）

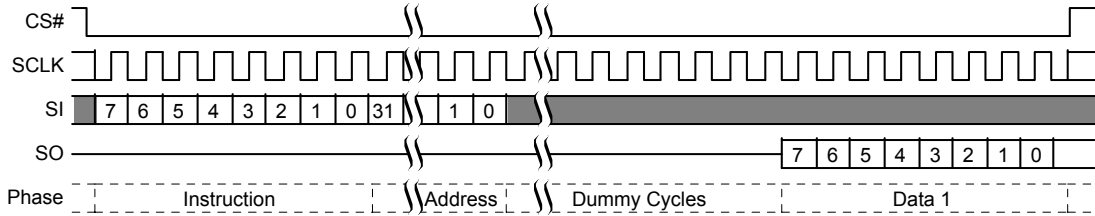


图 11. 双线输出读命令

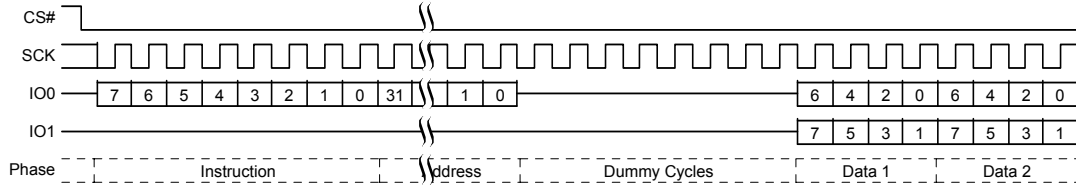


图 12. 四线输出读命令

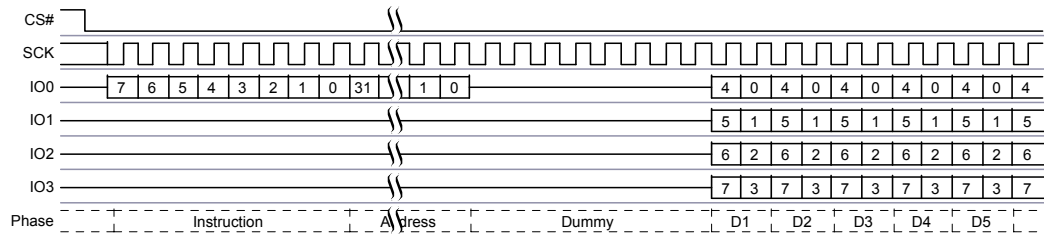


图 13. 双线 I/O 命令

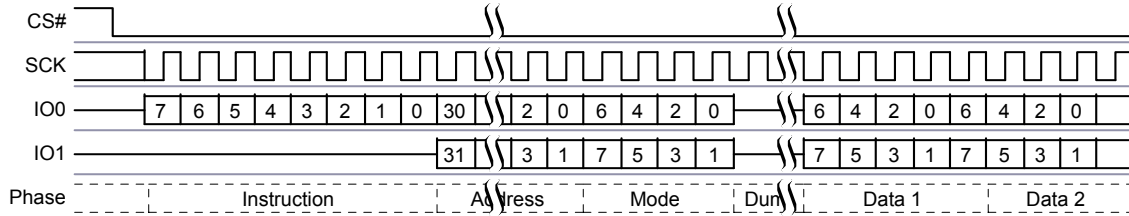
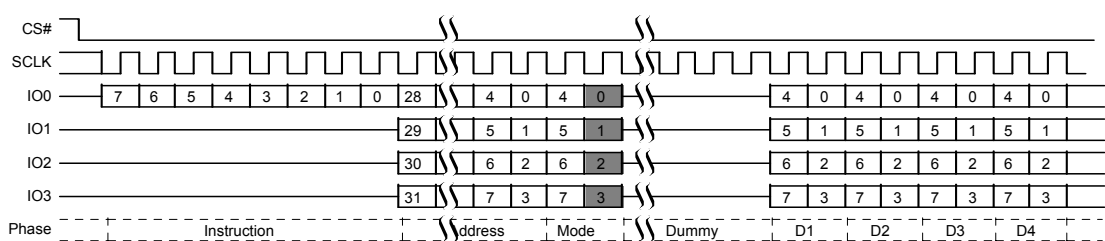


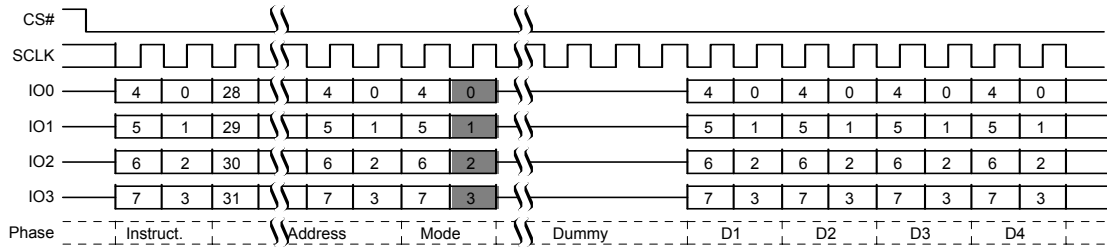
图 14. 四线 I/O 命令



注释

5. 灰色显示的是可选位，在该周期中主机无需发送这些位。

图 15. 在 QPI 模式下的四线 I/O 读命令



注释

6. 灰色显示的是可选位，在该周期中主机无需发送这些位。

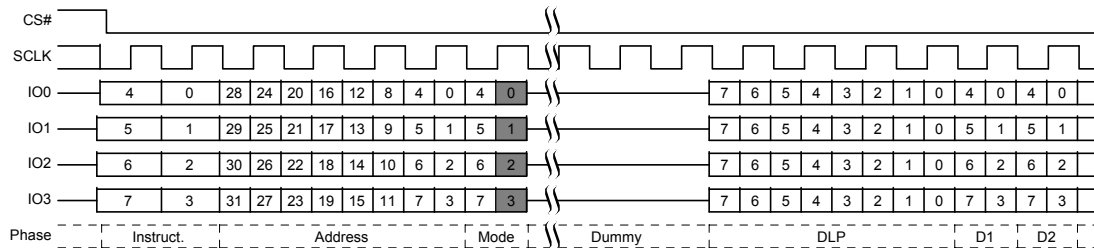
图 16. DDR 双线 I/O 读命令



注意

7. 灰色显示的是可选位，在该周期中主机无需发送这些位。

图 17. 在 QPI 模式下的 DDR 四线 I/O 读命令



注意

8. 灰色显示的是可选位，在该周期中主机无需发送这些位。

有关每个命令的其他序列框图，请参考第 72 页上的命令一节中的内容。

4.3 接口状态

本节将说明与 SPI 接口状态相应的输入和输出信号电平。

表 3. 接口状态汇总

接口状态	V _{CC}	SCK	CS#	RESET#	IO3_RE-SET#	WP#/IO2	SO/IO1	SI/IO0
断电	< V _{CC} (低电平)	X	X	X	X	X	X	X
低电压 硬件数据保护	< V _{CC} (关闭)	X	X	X	X	X	Z	X
上电 (冷) 复位	≥ V _{CC} (最小值)	X	HH	X	X	X	Z	X
硬件 (热) 复位非四线模式	≥ V _{CC} (最小值)	X	X	HL	HL	X	Z	X
硬件 (热) 复位四线模式	≥ V _{CC} (最小值)	X	HH	HL	HL	X	Z	X
接口待机状态	≥ V _{CC} (最小值)	X	HH	HH	X	X	Z	X
指令周期 (传统 SPI)	≥ V _{CC} (最小值)	HT	HL	HH	HH	HV	Z	HV
单线输入周期 从主机到存储器的传输	≥ V _{CC} (最小值)	HT	HL	HH	HH	X	Z	HV
单线延迟 (虚拟) 周期	≥ V _{CC} (最小值)	HT	HL	HH	HH	X	Z	X
单线输出周期 从存储器到主机的传输	≥ V _{CC} (最小值)	HT	HL	HH	HH	X	MV	X
双线输入周期 从主机到存储器的传输	≥ V _{CC} (最小值)	HT	HL	HH	HH	X	HV	HV
双线延迟 (虚拟) 周期	≥ V _{CC} (最小值)	HT	HL	HH	HH	X	X	X
双线输出周期 从存储器到主机的传输	≥ V _{CC} (最小值)	HT	HL	HH	HH	X	MV	MV
四线输入周期 从主机到存储器的传输	≥ V _{CC} (最小值)	HT	HL	HH	HV	HV	HV	HV
四线延迟 (虚拟) 周期	≥ V _{CC} (最小值)	HT	HL	HH	X	X	X	X
四线输出周期 从存储器到主机的传输	≥ V _{CC} (最小值)	HT	HL	HH	MV	MV	MV	MV
DDR 四线输入周期 从主机到存储器的传输	≥ V _{CC} (最小值)	HT	HL	HH	HV	HV	HV	HV
DDR 延迟 (虚拟) 周期	≥ V _{CC} (最小值)	HT	HL	HH	MV 或 Z	MV 或 Z	MV 或 Z	MV 或 Z
DDR 四线输出周期 从存储器到主机的传输	≥ V _{CC} (最小值)	HT	HL	HH	MV	MV	MV	MV

LEGEND Z = 无驱动信号 — 悬空信号, HL = y 由主机驱动的 V_{IL}, HH = y 由主机驱动的 V_{IH}, HV = HL 或 HH, X = HL 或 HH 或 Z, HT = 在 HL 和 HH 间切换, ML = 由存储器驱动的 V_{IL}, MH = 由存储器驱动的 V_{IH}, MV = ML 或 MH。

4.3.1 V_{CC} (断电)

当内核供电电压不大于 $V_{CC}(\text{Low})$ 电压时，器件被视为断电。这时，器件不对外部信号做出响应，防止对该器件执行的所有编写或擦除操作。

4.3.2 低功耗时的硬件数据保护

如果 V_{CC} 小于 $V_{CC}(\text{Cut-off})$ ，那么存储器器件会忽略各条命令，以保证内核供电电压在工作范围外时不会启动编写和擦除操作。

4.3.3 上电 (冷) 复位

如果内核电压在 $\geq t_{PD}$ 时间内保持不超过 $V_{CC}(\text{Low})$ ，然后上升到 $\geq V_{CC}(\text{Minimum})$ 值时，器件将开始执行上电复位 (POR) 程序。继续执行 POR 程序，直到 t_{PU} 时间结束为止。在 t_{PU} 期间，器件不会对外部输入信号做出响应，也不会驱动任何输出。经过 t_{PU} 时间后，器件将切换到接口待机状态，并能够接受各条命令。欲了解有关 POR 的更多信息，请参见第 31 页上的上电 (冷) 复位。

4.3.4 硬件 (暖) 复位状态

这是一种配置选项，当器件未处于任何四线或 QPI 模式时，或者当器件处于任何四线模式或 QPI 模式且 $CS\#$ 为高电平时，通过使用该选项，可以将 $IO3_RESET\#$ 作为硬件复位输入。在四线模式或 QPI 模式下，某些封装中会提供单独的复位输入 ($RESET\#$)。在 t_{RP} 期间，如果 $IO3_RESET\#$ 或 $RESET\#$ 被置为低电平，则器件将启动硬件复位程序。继续执行该程序 t_{RPH} 时间。经过 t_{RPH} 时间以及在 $RESET\#$ 上升后的复位保持时间 (t_{RH}) 后，器件将切换到接口待机状态，并能够接收各条命令。欲了解有关硬件复位的更多信息，请参见第 32 页上的启动硬件 (暖) 复位的 $RESET\#$ 和 $IO3_RESET\#$ 输入。

4.3.5 接口待机

如果 $CS\#$ 为高电平，则 SPI 接口处于待机状态。其他输入 ($RESET\#$ 除外) 均被忽略。接口会保持等待状态，直到开始执行新命令为止。当 $CS\#$ 转为低电平以启动一条新命令时，下一个接口状态是指令周期。

在接口待机状态下，如果不再执行任何嵌入式算法，存储器器件将消耗待机电流 (I_{SB})。如果正在执行某个嵌入式算法，那么将消耗相应的电流，直到完成该算法为止，此时整个器件将消耗待机电流。

在 FS-S 系列器件中，支持深度掉电 (DPD) 模式。如果使用 DPD (B9h) 命令使器件进入 DPD 模式，接口待机电流将为 (I_{DPD})。仅在状态寄存器 1 易失性正在写入 (WIP) 位被清零 ($SR1V[0] = 0$)，表示器件不执行嵌入式算法时，才可接受 DPD 命令。在 DPD 模式下，器件将忽略所有命令 (从 DPD 释放 (RES ABh) 命令除外 — 经过 t_{RES} 延迟后，该命令使器件返回接口待机状态)。

4.3.6 指令周期状态 (传统的 SPI 模式)

当主机驱动了一个指令的最高有效位 (MSB) 且 $CS\#$ 转为低电平时，在 SCK 的下一个上升沿上，器件将捕获指令 (开始新命令) 的最高有效位。在后面每一个 SCK 的上升沿上，器件都将捕获 8 位指令的下一个较低有效位。主机保持 $CS\#$ 为低电平并驱动写保护 ($WP\#$) 和 $IO3_RESET\#/RESET\#$ 信号以供指令使用。不过，仅在处理 WRR 或 $WRAR$ 命令或任何影响状态寄存器、配置寄存器和 DLR 寄存器命令的指令周期内， $WP\#$ 才被验证。除非执行这些指令周期期间，否则 $WP\#$ 将被忽略。如果器件未处于四线模式 ($CR1V[1] = 0$) 或 QPI 模式 ($CR2V[3] = 0$)，并且不需要硬件复位，则 $IO3_RESET\#$ 被置为高电平。

每个指令选择了用于操作的地址空间和在执行命令的其余部分期间使用的传输格式。传输格式可能为单线、双线 O、四线 O、双线 I/O、四线 I/O 或 DDR 四线 I/O。下一个接口的状态取决于所接收到的指令。

一些命令可以独立使用，无需将地址或数据传输到存储器或传输存储器的地址或数据。经过 SCK (在这种命令中指令的第八位) 上升沿后，主机将 $CS\#$ 返回高电平状态。在这种情况下，下一个接口状态是接口待机。

4.3.7 指令周期状态（QPI 模式）

在 QPI 模式下，如果 CR2V[6] = 0，则在每个周期内各条指令能够传输 4 比特。在该模式下，命令周期与四线输出周期相同。请参见第 21 页上的四线输入周期 — 从主机到存储器的传输。

4.3.8 单线输入周期 — 从主机到存储器的传输

将单线串行输入（SI）信号上的指令从主机发送到存储器器件后，一些命令会传输信息。主机保持 RESET# 为高电平、CS# 为低电平，并驱动 SI 以供命令使用。存储器不会驱动串行输出（SO）信号。

下一个期望接口的状态取决于该指令。通过使用额外的单线输入周期，某些指令会连续将地址或数据发送给存储器。其他指令可能会切换到单线延迟状态，或直接转到单线、双线或四线输出周期状态。

4.3.9 单线延迟（虚拟）周期状态

读取命令可能没有延迟周期，也可能会有若干个延迟周期，在这些周期内，直到发送到主机前，主闪存存储器阵列内的数据都被读取。配置寄存器（CR3V[3:0]）中的延迟代码决定了延迟周期的数量。在延迟周期内，主机会保持 RESET# 和 IO3_RESET# 为高电平、CS# 为低电平，SCK 会进行切换。写保护（WP#）信号被忽略。在这些周期内，主机可能会驱动 SI 信号，或者将 SI 保持为悬空状态。在延迟周期内，存储器不会使用在 SI 或其他 I/O 信号上所驱动的任何数据。在延迟周期内，存储器不会驱动串行输出（SI）或其他 I/O 信号。

下一个接口状态取决于命令结构，例如，延迟周期数量以及读取方式是单线、双线或四线宽度。

4.3.10 单线输出周期 — 从存储器到主机的传输

通过单线串行输出（SO）信号，若干个命令会将信息返回到主机。主机将保持 RESET#，并且 IO3_RESET# 为高电平、CS# 为低电平。写保护（WP#）信号被忽略。存储器忽略了串行输入（SI）信号。存储器使用数据驱动 SO 信号。

下一个接口状态仍是单线输出周期，直到主机通过将 CS# 返回到高电平状态来结束命令为止。

4.3.11 双线输入周期 — 从主机到存储器的传输

每个周期内，双线读取 I/O 命令都会将两个地址或模式位传输给存储器。主机将保持 RESET#，并且 IO3_RESET# 为高电平、CS# 为低电平。写保护（WP#）信号被忽略。主机将驱动 SI / IO0 和 SO / IO1 上的地址。

传输地址和模式位后，下一个接口状态是双线延迟周期（若需要延迟周期）或双线输出周期（若无需延迟周期）。

4.3.12 双线延迟（虚拟）周期

读取命令可能没有延迟周期，也可能会有若干个延迟周期，在这些周期内，直到发送到主机前，主闪存存储器阵列内的数据都被读取。配置寄存器（CR3V[3:0]）中的延迟代码决定了延迟周期的数量。在延迟周期内，主机会保持 RESET# 和 IO3_RESET# 为高电平、CS# 为低电平，SCK 会持续切换。写保护（WP#）信号被忽略。在这些周期内，主机可能驱动 SI / IO0 和 SO / IO1 信号，或者将这些信号保持为悬空状态。在延迟周期内，存储器不会使用在 SI / IO0 和 SO / IO1 或其他 I/O 信号上所驱动的任何数据。最后的延迟周期结束时，主机必须停止驱动 SCK 的下降沿上 SI / IO0 和 SO / IO1 信号。在所有延迟周期内，建议停止主机对这些信号的驱动操作，以便在延迟周期结束时存储器开始驱动前，有足够时间使主机驱动器停止传输操作。这样做，如果信号方向发生改变，则能够防止主机与主机和存储器间发生冲突。在延迟周期内，存储器不会驱动 SI / IO0 和 SO / IO1 信号。

最后延迟周期结束后，下一个接口状态是双线输出周期。

4.3.13 双线输出周期 — 从存储器到主机的传输

在每个周期内，双线读取输出和双线读取 I/O 命令会将数据（2 位）返回到主机内。主机将保持 RESET#，并且 IO3_RESET# 为高电平、CS# 为低电平。写保护（WP#）信号被忽略。在双线输出周期内，在 SCK 的下降沿上，存储器将驱动 SI / IO0 和 SO / IO1 信号上的数据。

下一个接口状态仍是双线输出周期，直到主机通过使 CS# 返回高电平状态来结束命令为止。

4.3.14 QPP 或 QOR 地址输入周期状态

四线页编程和四线输出读取命令仅在 IO0 信号上将地址发送到存储器。其他 IO 信号均被忽略。主机将保持 RESET# 和 IO3_RESET# 为高电平、CS# 为低电平，并驱动 IO0。

QPP 命令发送地址后，下一个接口状态将是四线输入周期。QOR 传输地址后，下一个接口状态是四线延迟周期（若需要延迟周期）或四线输出周期（若无需延迟周期）。

4.3.15 四线输入周期 — 从主机到存储器的传输

每个周期内，四线 I/O 读取命令都将四个地址位或模式位传输到存储器中。在 QPI 模式下，每个周期（包括指令周期）内，四线 I/O 读取和页编程命令都会将四个数据位传输到存储器内。主机会保持 CS# 为低电平，并驱动 IO 信号。

四线 I/O 读取命令传输地址和模式位后，下一个接口状态是四线延迟周期（若需要延迟周期）或四线输出周期（若无需延迟周期）。在 QPI 模式下，页编程命令传输需要编程的数据后，主机会使 CS# 返回高电平状态，而且接口会返回待机状态。

4.3.16 四线延迟（虚拟）周期

读取命令可能不存在延迟周期，也可能存在若干个延迟周期。在这些周期内，数据先从主闪存存储器阵列内被读取，然后才被发送给主机。配置寄存器（CR3V[3:0]）中的延迟代码决定了延迟周期的数量。在延迟周期内，主机会保持 CS# 为低电平，并持续切换 SCK。在这些周期内，主机可能驱动 IO 信号，或者将 IO 保持为悬空状态。在延迟周期内，存储器不会使用 IO 信号上所驱动的任意数据。最后的延迟周期结束时，主机必须停止驱动下降沿上的 IO 信号。在所有延迟周期内，建议停止主机对这些信号的驱动操作，以便在延迟周期结束时存储器开始驱动前，有足够时间使主机驱动器停止传输操作。这样做，如果信号方向发生改变，则能够防止主机与主机和存储器间发生冲突。在延迟周期内，存储器不会驱动 IO 信号。

最后延迟周期结束后，下一个接口状态是四线输出周期。

4.3.17 四线输出周期 — 从存储器到主机的传输

每个周期内，四线 O 和四线 I/O 读取命令会将 4 位的数据返回给主机。主机保持 CS# 为低电平。在四线输出周期期间，存储器会驱动 IO0-IO3 信号上的数据。

下一个接口状态仍是四线输出周期，直到主机使 CS# 返回高电平状态而使命令结束为止。

4.3.18 DDR 四线输入周期 — 从主机到存储器的传输

DDR 四线 I/O 读取命令会通过所有 IO 信号将地址和模式位发送给存储器。每个周期内，在 SCK 的上升沿上传输 4 比特，在下降沿上传输 4 比特。主机保持 CS# 为低电平。

传输地址和模式位后，下一个接口状态是 DDR 延迟周期。

4.3.19 DDR 延迟周期

DDR 读取命令可能具有一个或若干个延迟周期，在这些周期内，数据被发送给主机前，主闪存存储器阵列内的数据将被读取。配置寄存器（CR2V[3:0]）中的延迟代码决定了延迟周期的数量。在延迟周期内，主机会保持 CS# 为低电平。此外，主机不会驱动 IO 信号。因此，在存储器开始驱动前，有足够的时间使主机驱动器停止传输操作。这样做，如果信号方向发生改变，便能够防止主机与主机和存储器间发生冲突。在最后 4 个延迟周期内，存储器可以选择使用一个数据学习模式（DLP）驱动所有 IO 信号。如果延迟周期数小于 5，则不应该使能 DLP 选项。因此，在存储器开始驱动 DLP 前，至少有一个高阻抗的周期以等待 IO 信号经过处理并返回。如果有多于 4 个延迟周期，则存储器不会驱动 IO 信号，直到最后四个延迟周期为止。

最后延迟周期结束后，下一个接口状态是 DDR 单线或四线输出周期，具体情况取决于指令。

4.3.20 DDR 四线输出周期 — 从存储器到主机的传输

DDR 四线 I/O 读取命令会通过所有 IO 信号将数据位发送给主机。每个周期内，在 SCK 的上升沿上传输 4 比特，在下降沿上传输 4 比特。主机保持 CS# 为低电平。

下一个接口状态仍是 DDR 四线输出周期，直到主机使 CS# 返回高电平状态而使命令结束为止。

4.4 接口上的配置寄存器影响

配置寄存器 2 的易失性位 3 到 0 (CR2V[3:0]) 为所有阵列读取命令 (读取、RUID 和读 SDFP (RSFDP) 命令除外) 选择变量延迟。读取命令的延迟周期始终为 0。RSFDP 命令的延迟周期始终为 8。OTPR、ECCRD 和 RDAR 命令也使用了变量延迟。

通过使用配置寄存器位 1 (CR1V[1]) 可以选择是否使能四线模式, 从而将 WP# 切换到 IO2 功能, 将 RESET# 切换到 IO3 功能, 这样便允许执行四线 I/O 读取和 QPI 模式命令。必须选择四线模式, 以允许执行 DDR 四线 I/O 读取命令。

4.5 数据保护

硬件设计提供并管理着某些基本的保护功能, 用于防止对存储数据进行意外的更改。具体内容如下。其它软件管理的保护方式将在本文档的软件章节中进行讨论。

4.5.1 上电

当内核供电电压不大于 $V_{CC(Low)}$ 电压时, 器件被视为断电。这时, 器件不对外部信号做出响应, 防止对该器件执行的所有编写或擦除操作。在上电复位 (POR) 期间继续防止执行编程和擦除操作。这是因为所有命令不被接收, 直到从 POR 转换为接口待机状态为止。

4.5.2 低功耗

如果 V_{CC} 小于 $V_{CC(Cut-off)}$, 那么存储器器件会忽略各条命令, 以保证内核供电电压在工作范围外时不会启动编写和擦除操作。

4.5.3 时钟脉冲数量

器件会验证所有修改非易失性存储器和寄存器数据的命令是否包含了一个时钟脉冲数量。执行这些命令前, 该时钟脉冲数量是 8 的倍数 (字节边界)。不在 8 位 (字节) 边界上结束的命令将被忽略, 该命令不存在任何错误状态。

4.5.4 深度掉电 (DPD)

在 DPD 模式下, 器件仅对从 DPD 模式恢复的命令 (RES ABh) 发出响应。在该模式下, 所有其他命令均被忽略, 从而防止对存储器执行编程和擦除操作。如果 IO3_RESET# 功能被使能 (CR2V[5]=1) 或 RESET# 有效, 那么 IO3_RESET# 或 RESET# 被置为低电平状态将启动硬件复位并使器件退出 DPD 模式。

5. 电气规范

5.1 最大绝对额定值

存储温度，塑料包装	-65 °C 到 +150 °C
通电时的环境温度	-65 °C 到 +125 °C
V_{CC}	-0.5 V 到 +2.5 V
对地 (V_{SS}) 输入电压 (注意 1)	-0.5 V 到 $V_{CC} + 0.5 V$
输出短路电流 (注意 2)	100 mA

注意

9. 请参考第 24 页上的输入信号过冲以了解信号切换期间受允许的最大值。
10. 不能有多于一个输出同时短路接地。并且短路时间不能超过一秒。
11. 器件在高于“最大绝对额定值”部分中所列出的值工作可能会造成永久性的损害。它只是一个额定值，并不表示器件一定要采用这些值，或者采用本数据手册操作部分所显示的数值都能正常运行。如果让器件长时间在绝对最大额定值情况下运行，会影响器件的可靠性。

5.2 闩锁特性

表 4. 闩锁规范

说明	最小值	最大值	单位
所有输入连接上相对于 V_{SS} 的输入电压	-1.0	$V_{CC} + 1.0$	V
所有 I/O 连接上相对于 V_{SS} 的输入电压	-1.0	$V_{CC} + 1.0$	V
V_{CC} 电流	-100	+100	mA

注意

12. 不包括 V_{CC} 电源。测试条件: $V_{CC} = 1.8 V$ ，每次检查一个连接， V_{SS} 下的连接不被检查。

5.3 热阻

表 5. 热电阻

参数	说明	W9A008	SOC008	FAB024	单位
Theta JA	热阻 (结至环境)	85.8	53.27	38.93	°C/W

5.4 工作范围

工作范围定义了一些限值，在这些限值之间能够保证设备正常运行。

5.4.1 供电电压

V_{CC}	1.7 V 到 2.0 V
----------	---------------

5.4.2 温度范围

参数	符号	器件	规范		单位
			最小值	最大值	
环境温度	T_A	工业级 (I)	-40	+85	°C
		扩展的工业级 (V)	-40	+105	
		扩展型 (N)	-40	+125	
		汽车级, AEC-Q100 等级 3 (A)	-40	+85	
		汽车级, AEC-Q100 等级 2 (B)	-40	+105	
		汽车级, AEC-Q100 等级 1 (M)	-40	+125	

扩展的工业级操作和性能参数由器件的特性决定，它们与标准工业级温度范围器件有所不同，如现行规范所示。

5.4.3 输入信号过冲

在直流条件下，输入或 I/O 信号电压应处于从 V_{SS} 到 V_{CC} 的范围内。在电压转换期间，输入或 I/O 信号可能低于 V_{SS} ，降到 -1.0 V 或升到 $V_{CC} + 1.0\text{ V}$ ，时间最长为 20 ns 。

图 18. 最大负过冲波形

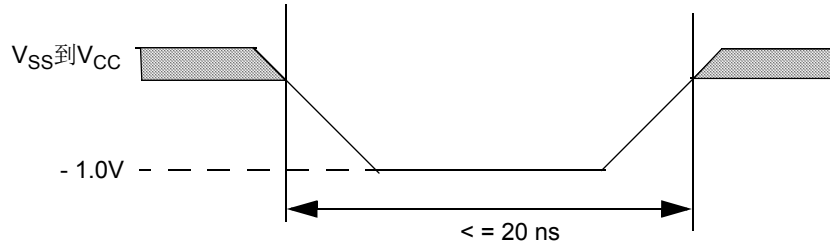
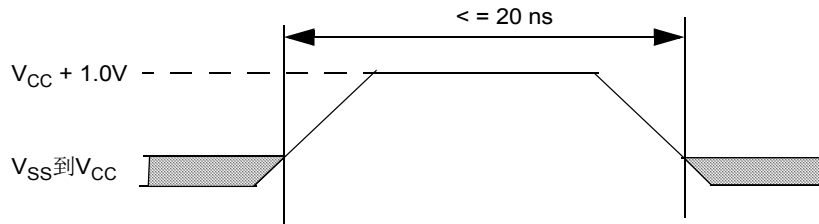


图 19. 最大正过冲波形



5.5 上电和掉电

发生上电或掉电（ $CS\#$ 的电压必须采用 V_{CC} 上的电压）时，不能选择器件，直到 V_{CC} 准确满足以下值：

- 上电时为 V_{CC} （最小值）并在 t_{PU} 的更长延迟期间持续保持该值
- 掉电时为 V_{SS}

为保证安全、准确进行上电和掉电操作，可以在片选（ $CS\#$ ）上安装一个简单的上拉电阻。

器件会忽略所有指令，直到 V_{CC} 上升到超过 V_{CC} 最小阈值后经过 t_{PU} 延迟时间为止。请参见图 20。但是在 t_{PU} 期间，如果 V_{CC} 返回到低于 V_{CC} （最小值）的电平状态，将不能保证器件正常操作。在 t_{PU} 结束前，不应该发出任何命令。

在 t_{PU} 期间，器件消耗的电流为 I_{POR} 。上电（ t_{PU} ）后，器件将进入待机模式，并消耗 CMOS 待机电流（ I_{SB} ）， WEL 位被复位。

在掉电期间或电压降至 $V_{CC}(\text{cut-off})$ 以下时，电压必须在 t_{PD} 的时间内保持小于 $V_{CC}(\text{LOW})$ ，这样器件上电后能够正常初始化。请参见图 21。如果在电压下降过程中， V_{CC} 保持为高于 V_{CC} （关闭）的状态，那么当 V_{CC} 再次超过 V_{CC} 最小值时，器件将被初始化，并正常运行。如果上电后不能正常完成上电复位（POR）事件，那么当激活 $RESET\#$ 信号或接收到一个软件复位命令（RESET）时，器件将重启 POR 程序。

为了稳定器件的 V_{CC} 电源，必须采取正常的预防措施进行电源轨退耦。系统中的每个器件都应该由封装连接附近大小合适的电容对 V_{CC} 轨进行退耦（此电容通常为 $0.1\text{ }\mu\text{F}$ ）。

表 6. 上电 / 掉电电压和时序

符号	参数	最小值	最大值	单位
V_{CC} （最小值）	V_{CC} （最小工作电压）	1.7		V
V_{CC} （关闭）	V_{CC} （需要重新初始化时要求的截断电压）	1.55		V
V_{CC} （低电平）	V_{CC} （确保发生初始化的低电压）	0.7		V
t_{PU}	V_{CC} 为最小值到进行读取操作的时间		300	μs
t_{PD}	V_{CC} （低电平）时间	10.0		μs

图 20. 上电

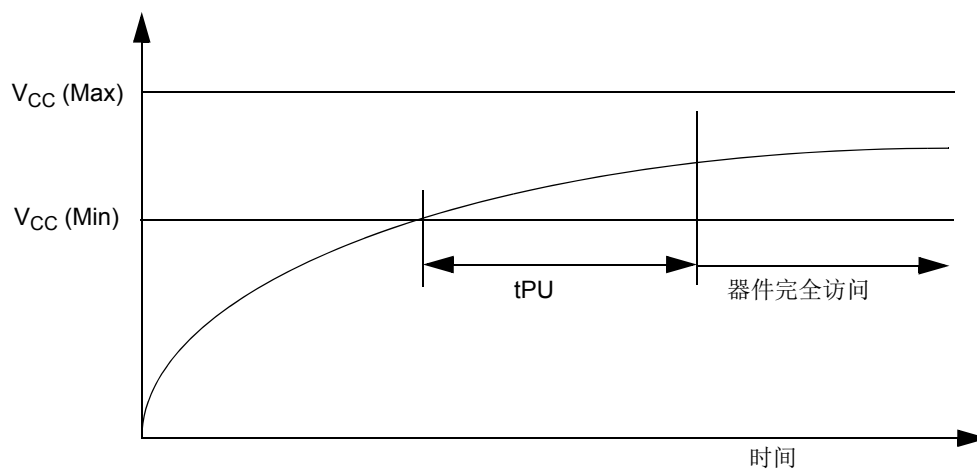
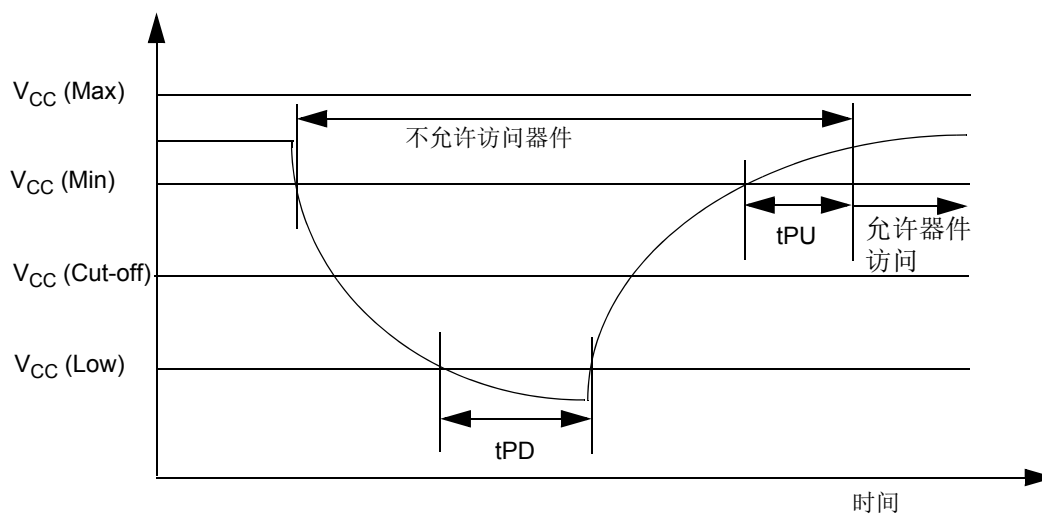


图 21. 掉电或电压下降



5.6 直流特性

5.6.1 工业

适用的工作范围为 -40°C 到 +85°C。

表 7. 直流特性 — 工业级

符号	参数	测试条件	最小值	典型值 (13)	最大值	单位
V_{IL}	输入低电压		-0.5		$0.3 \times V_{CC}$	V
V_{IH}	输入高电压		$0.7 \times V_{CC}$		$V_{CC} + 0.4$	V
V_{OL}	输出低电压	$I_{OL} = 0.1 \text{ mA}$			0.2	V
V_{OH}	输出高电压	$I_{OH} = -0.1 \text{ mA}$	$V_{CC} - 0.2$			V
I_{LI}	输入漏电流	$V_{CC} = V_{CC} \text{ Max}$, $V_{IN} = V_{IH}$ 或 V_{SS} , $CS\# = V_{IH}$			± 2	μA
I_{LO}	输出漏电流	$V_{CC} = V_{CC} \text{ Max}$, $V_{IN} = V_{IH}$ 或 V_{SS} , $CS\# = V_{IH}$			± 2	μA
I_{CC1}	活动电流 (读取) (14)	串行 SDR@50 MHz 串行 SDR@133 MHz QIO/QPI SDR@133 MHz QIO/QPI DDR@80 MHz		10 25 60 70	18 30 65 90	mA
I_{CC2}	活动电流 (页编程)	$CS\# = V_{CC}$		60	100	mA
I_{CC3}	活动电流 (WRR 或 WRAR)	$CS\# = V_{CC}$		60	100	mA
I_{CC4}	活动电流 (SE)	$CS\# = V_{CC}$		60	100	mA
I_{CC5}	活动电流 (BE)	$CS\# = V_{CC}$		60	100	mA
I_{SB}	待机电流	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS} , 工业级温度		25	100	μA
I_{DPD}	深度掉电	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS} , 工业级温度		6	50	μA
I_{POR}	上电复位电流	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS}			80	mA

注意

13. 典型值的条件为 $T_{AI} = 25^\circ\text{C}$ 和 $V_{CC} = 1.8 \text{ V}$ 。

14. 读取返回数据期间, 输出处于悬空状态。不包含输出开关电流。

5.6.2 工业级扩展的直流特性

适用的工作范围为 -40°C 到 +105°C。

表 8. 直流特性 — 扩展的工业级

符号	参数	测试条件	最小值	典型值 (13)	最大值	单位
V_{IL}	输入低电压		-0.5		$0.3 \times V_{CC}$	V
V_{IH}	输入高电压		$0.7 \times V_{CC}$		$V_{CC} + 0.4$	V
V_{OL}	输出低电压	$I_{OL} = 0.1 \text{ mA}$			0.2	V
V_{OH}	输出高电压	$I_{OH} = -0.1 \text{ mA}$	$V_{CC} - 0.2$			V
I_{LI}	输入漏电流	$V_{CC} = V_{CC} \text{ Max}$, $V_{IN} = V_{IH}$ 或 V_{SS} , $CS\# = V_{IH}$			± 4	μA
I_{LO}	输出漏电流	$V_{CC} = V_{CC} \text{ Max}$, $V_{IN} = V_{IH}$ 或 V_{SS} , $CS\# = V_{IH}$			± 4	μA
I_{CC1}	活动电流 (读取)	串行 SDR@50 MHz 串行 SDR@133 MHz QIO/QPI SDR@133 MHz QIO/QPI DDR@80 MHz		10 25 60 70	18 30 65 90	mA
I_{CC2}	活动电流 (页编程)	$CS\# = V_{CC}$		60	100	mA
I_{CC3}	活动电流 (WRR 或 WRAR)	$CS\# = V_{CC}$		60	100	mA
I_{CC4}	活动电流 (SE)	$CS\# = V_{CC}$		60	100	mA
I_{CC5}	活动电流 (BE)	$CS\# = V_{CC}$		60	100	mA
I_{SB}	待机电流	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS}		25	300	μA
I_{DPD}	深度掉电	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS}		6	100	μA
I_{POR}	上电复位电流	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS}			80	mA

注意

15. 典型值的条件为 $T_{AI} = 25^\circ\text{C}$ 和 $V_{CC} = 1.8 \text{ V}$ 。

16. 读取返回数据期间, 输出处于未连接状态。不包含输出开关电流。

5.6.3 MWEX

适用的工作范围为 -40°C 到 +125°C。

表 9. 直流特性 - 扩展范围

符号	参数	测试条件	最小值	典型值 (13)	最大值	单位
V_{IL}	输入低电压		-0.5		$0.3 \times V_{CC}$	V
V_{IH}	输入高电压		$0.7 \times V_{CC}$		$V_{CC} + 0.4$	V
V_{OL}	输出低电压	$I_{OL} = 0.1 \text{ mA}$			0.2	V
V_{OH}	输出高电压	$I_{OH} = -0.1 \text{ mA}$	$V_{CC} - 0.2$			V
I_{LI}	输入漏电流	$V_{CC} = V_{CC \text{ Max}}$, $V_{IN} = V_{IH}$ 或 V_{SS} , $CS\# = V_{IH}$			± 4	μA
I_{LO}	输出漏电流	$V_{CC} = V_{CC \text{ Max}}$, $V_{IN} = V_{IH}$ 或 V_{SS} , $CS\# = V_{IH}$			± 4	μA
I_{CC1}	有源电源电流 (读取)	串行 SDR@50 MHz 串行 SDR@133 MHz QIO/QPI SDR@133 MHz QIO/QPI DDR@80 MHz		10 25 60 70	18 30 65 90	mA
I_{CC2}	活动电流 (页编程)	$CS\# = V_{CC}$		60	100	mA
I_{CC3}	活动电流 (WRR 或 WRAR)	$CS\# = V_{CC}$		60	100	mA
I_{CC4}	活动电流 (SE)	$CS\# = V_{CC}$		60	100	mA
I_{CC5}	活动电流 (BE)	$CS\# = V_{CC}$		60	100	mA
I_{SB}	待机电流	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS}			300	μA
I_{DPD}	深度掉电电流	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS}		6	170	μA
I_{POR}	上电复位电流	IO3/RESET#, $CS\# = V_{CC}$; SI, SCK = V_{CC} 或 V_{SS}			80	mA

注意

17. 典型值的条件为 $T_{AI} = 25^\circ\text{C}$ 和 $V_{CC} = 1.8 \text{ V}$ 。

18. 读取返回数据期间, 输出处于悬空状态。不包含输出开关电流。

5.6.4 有效功耗模式和待机功耗模式

如果片选（CS#）为低电平，器件被使能且处于有效功耗模式。如果 CS# 为高电平，器件将被禁用，但仍处于活动模式，直到完成所有编程、擦除以及写入操作为止。然后，器件会进入待机功耗模式，功耗降低到 I_{SB} 。

5.6.5 深度掉电模式（DPD）

通过输入命令指令代码“B9h”，并将功耗降低到 I_{DPD} ，可使能深度掉电模式。仅在状态寄存器 1 易失性正在写入（WIP）位被清零（SR1V[0] = 0），表示器件不执行嵌入式算法时，才可接受 DPD 命令。在 DPD 模式下，器件仅对从 DPD 模式恢复的命令（RES ABh）或硬件复位（RESET# 和 IO3_RESET#）发出响应。在 DPD 模式下，所有其他命令均被忽略。

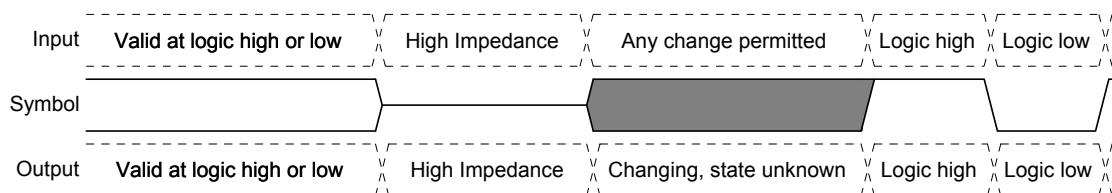
表 10. 有效进入 DPD 模式并退出 DPD 模式序列

当前模式	CS#	SCK	命令	下一个模式	说明
活动	低电平到高电平	N/A	N/A	待机	
待机	高电平到低电平	切换	B9h 进入 DPD 模式	DPD	CS# 为高电平且在 t_{DPD} 时间后进入 DPD 模式（请参考第 34 页上的表 14）
DPD	高电平到低电平	未切换	N/A	DPD	如果 SCK 为切换，且命令不是 ABh，器件保持处于 DPD 模式
		切换	命令不是 ABh		
DPD	高电平到低电平	切换	ABh 退出 DPD 模式	待机	CS# 为高电平，并在经过 t_{RES} 时间后退出 DPD 模式（请参考第 34 页上的表 14）。CS# 为高电平开始退出 DPD 模式后，SCK 为未切换时，将没有一个有效的 CS# 切换序列。

6. 时序规范

6.1 切换波形的关键

图 22. 波形因素的含义



6.2 交流测试条件

图 23. 测试设置

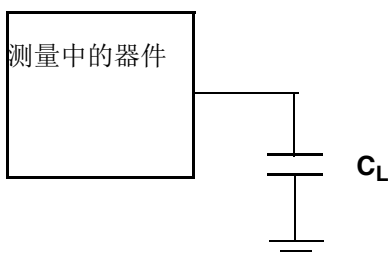


表 11. 交流测量条件

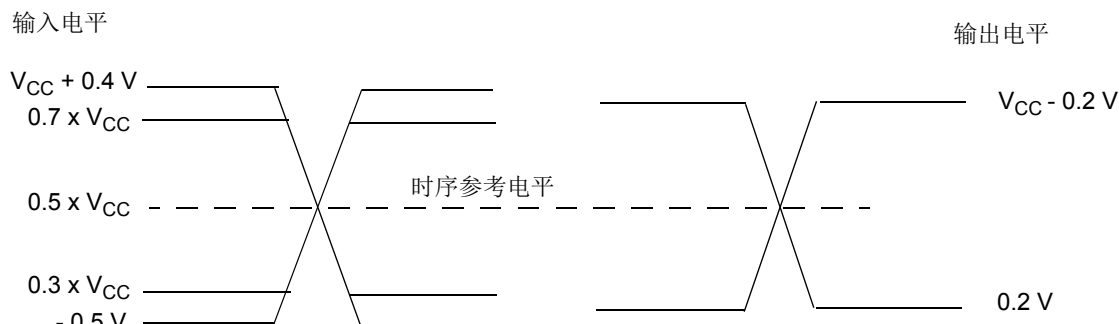
符号	参数	最小值	最大值	单位
C_L	负载电容		30	pF
	输入脉冲电压	$0.2 \times V_{CC}$	$0.8 V_{CC}$	V
	输入转换速率	0.23	1.25	V/ns
	输入上升和下降时间	0.9	5	ns
	输入时序参考电压	$0.5 V_{CC}$		V
	输出时序参考电压	$0.5 V_{CC}$		V

注意

19. 输入转换速率是 V_{CC} 为最大值时的最小输入脉冲与最大输入脉冲的偏差。例如： $(1.9 V \times 0.8) - (1.9 V \times 0.2) = 1.14 V$ ； $1.14 V / 1.25 V/ns = 0.9 ns$ （上升或下降时间）。

20. 交流特性表假定时钟和数据信号具有相同的转换速率（斜率）。

图 24. 输入、输出和时序的参考电平



6.2.1 电容特性

表 12. 电容

	参数	测试条件	封装	最小值	最大值	单位
C_{IN}	输入电容（用于 SCK、CS#、IO3/RESET#）	1 MHz	SOIC		12.5	pF
			LGA、BGA		8	
C_{OUT}	输出电容（使用于所有 I/O）	1 MHz	SOIC		12	pF
			LGA、BGA		8	

6.3 复位

6.3.1 上电（冷）复位

器件会执行上电复位（POR）程序，直到 V_{CC} 上升到超过 V_{CC} 最小阈值后经过 t_{PU} 延迟时间为止。请参考第 25 页上的图 20、第 24 页上的表 6。上电期间（ t_{PU} ），不能选择器件（不允许 CS# 的电压上升到 V_{CC} ）。例如，不能将任何命令发送给器件，直到经过 t_{PU} 为止。

电源复位期间，RESET# 和 IO3_RESET# 被忽略，但必须保持高电平或低电平。如果在 POR 期间，RESET# 或 IO3_RESET# 为低电平，并且在 t_{PU} 期间以及之后仍然保持为低电平状态，那么 CS# 必须保持为高电平，直到 RESET# 和 IO3_RESET# 返回高电平状态并经过 t_{RH} 为止。在比 t_{RS} 更长的时间内，并且返回低电平状态前，RESET# 和 IO3_RESET# 必须返回高电平状态，以便启动一次硬件复位。

仅当 CS# 在比 t_{CS} 更长的时间内保持高电平状态，或者未使能四线模式或 QPI 模式（CR1V[1] = 0 或 CR2V[6] = 0）时，IO3_RESET# 输入才可作为 RESET# 信号使用。

图 25. POR 结束时，复位信号为低电平

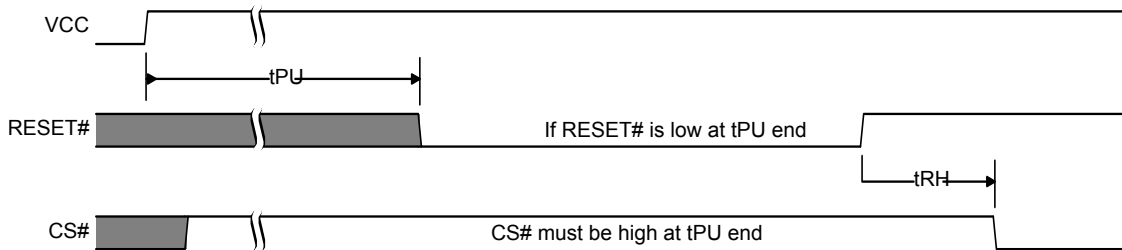


图 26. POR 结束时，复位信号为高电平

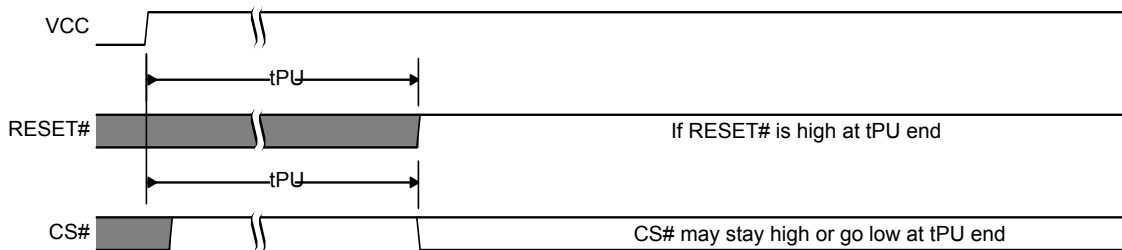
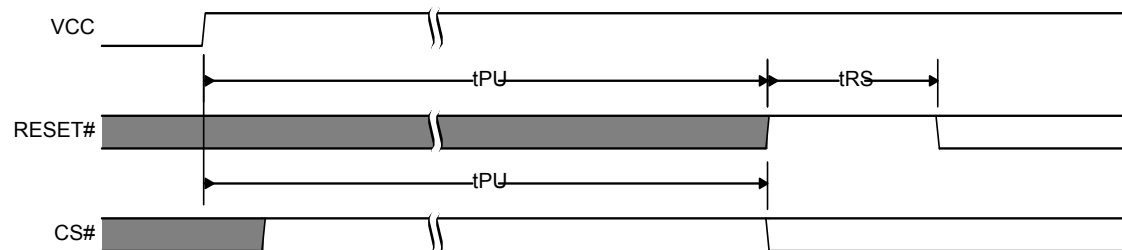


图 27. 由硬件复位随后的 POR



6.3.2 启动硬件（暖）复位的 RESET# 和 IO3_RESET# 输入

RESET# 和 IO3_RESET# 输入可作为 RESET# 信号使用。根据条件，两个输入可以启动复位操作。

在比 t_{RP} 更长的时间内，当从 V_{IH} 转换为 V_{IL} 时，RESET# 输入启动了复位操作，器件会使用与上电复位相同的方式复位寄存器状态，但没有经过 POR 期间所执行的完整复位程序。完成硬件复位程序需要 t_{RPH} 时长。RESET# 输入仅在 BGA ball 封装中有效。

当在比 t_{CS} 更长的时间内 CS# 为高电平时，或者当未使能四线模式或 QPI 模式（CR1V[1] = 0 或 CR2V[6] = 0）时，IO3_RESET# 输入将启动复位操作。IO3_RESET# 输入具有一个连接至 V_{CC} 的内部上拉电阻；如果不处于四线模式或 QPI 模式，该输入会保持为无连接状态。CS# 为低电平时该输入用作四线或 QPI 模式 I/O 信号后，需要 CS# 变为高电平后经过 t_{CS} 延迟时间，以便存储器或主机系统将 IO3 驱动为高电平。然后，将内部上拉电阻连接到 V_{CC} ，可保持 IO3_RESET# 为高电平，直到主机系统开始驱动 IO3_RESET# 为止。在 t_{CS} 期间，CS# 保持高电平状态时，IO3_RESET# 输入将被忽略，从而可以避免发生意外的复位操作。如果 CS# 被驱动为低电平，便会启动一个新命令，IO3_RESET# 被作为 IO3。

当器件未处于四线模式或 QPI 模式时，或者当 CS# 为高电平，并且 IO3_RESET# 在大于 t_{RP} （随后 t_{CS} ）的时间内从 V_{IH} 转换为 V_{IL} 时，器件会使用与上电复位相同的方式复位寄存器状态，但没有经过在 POR 期间所执行的完整复位程序。

完成硬件复位程序需要 t_{RPH} 时长。如果由于任意原因不能在上电（ t_{PU} ）期间完成 POR 程序，RESET# 变为低电平会启动完整的 POR 程序（而不是硬件复位程序），并需要占用 t_{PU} 长的时间才能完成 POR 程序。

软件复位命令（由 RST 99h 随后的 RSTEN 66h）独立于 RESET# 和 IO3_RESET# 的状态。如果 RESET# 和 IO3_RESET# 为高电平，且已经发送了软件复位指令，则器件将执行软件复位。

其他 IO3 RESET# 注意事项：

- 如果 RESET# 和 IO3_RESET# 输入选项均可用，那么在您的系统中仅使用了一个复位选项。通过将 CR2NV[7] 设置为 0（请参考第 55 页上的表 32），并设置 IO3_RESET，使其仅作为 IO3 使用，可以禁用 IO3/RESET# 输入复位操作。不连接 RESET# 输入，或将其绑定到 V_{IH} 上，可以禁用该输入。
- 重新变为低电平以启动硬件复位前，在经过加上 t_{PU} 或 t_{RPH} 的 t_{RS} 时间内，RESET# 或 IO3_RESET# 必须为高电平。
- 当 IO3_RESET# 被驱动为低电平的时间不短于一个最小周期（在 t_{CS} 后面的 t_{RP} ）时，器件会立即终止正在执行的所有操作，使所有输出为高阻抗，并在 t_{RPH} 期间忽略所有读 / 写命令。器件将接口复位为待机状态。
- 如果使能了四线模式或 QPI 模式和 IO3_RESET# 性能，则在 t_{CS} 期间，主机系统不会将 IO3 驱动为低电平，从而可以避免在 IO3 上发生冲突。发送在四线或 QPI 模式下将数据传送给主机的命令后（例如四线 I/O 读取命令），存储器立即在 t_{CS} 时间内将 IO3_RESET# 驱动为高电平，从而防止发生意外的复位操作。发送在四线模式下将数据传送给存储器的命令后（例如页编程命令），主机系统立即在 t_{CS} 时间内将 IO3_RESET# 驱动为高电平，以防止发生意外复位操作。
- 如果未使能四线模式，并且 IO3_RESET# 被置为低电平时 CS# 为低电平，那么 CS# 经过 t_{RH} 后可能被重新置为低电平前，该信号必须在 t_{RPH} 时间内返回高电平。

表 13. 硬件复位参数

参数	说明	限制	时间	单位
t_{RS}	复位设置 — 从前面复位结束 RESET# 为高电平到 RESET# 为低电平的时间	最小值	50	ns
t_{RPH}	复位脉冲保持 — RESET# 为低电平到 CS# 为低电平的时间	最小值	35	μs
t_{RP}	RESET# 脉冲宽度	最小值	200	ns
t_{RH}	复位保持 — CS# 为低电平前 RESET# 为高电平的时间	最小值	50	ns

注意

21. 上电（ t_{PU} ）期间，RESET# 和 IO3_RESET# 低电平将被忽略。如果 t_{PU} 结束时 Reset# 被激活，那么器件将保持为复位状态， t_{RH} 会决定 CS# 变为低电平的时间。

22. 如果使能了四线模式，则在 t_{CS} 期间，IO3_RESET# 低电平将被忽略。

23. $t_{RP} + t_{RH}$ 的和不能小于 t_{RPH} 。

图 28. 使用 RESET# 输入的硬件复位

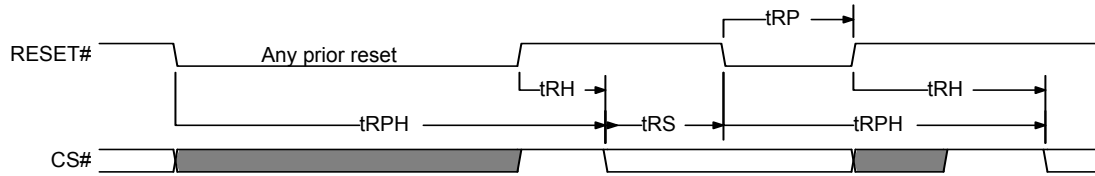


图 29. 未使能四线模式或 QPI 模式，并且使能了 IO3_RESET# 时的硬件复位

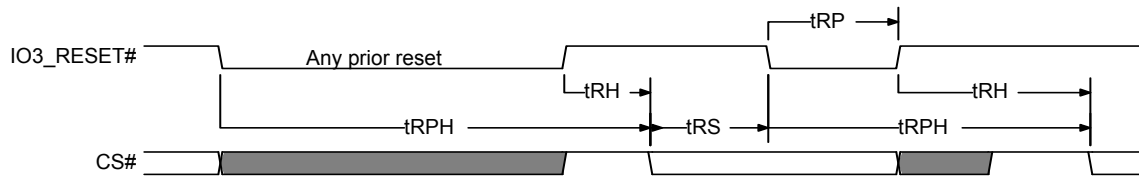
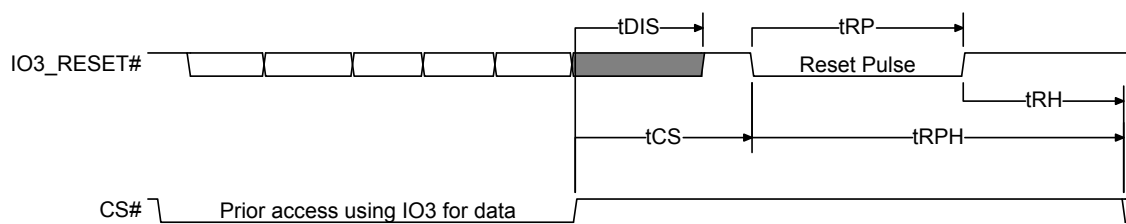


图 30. 使能了四线模式或 QPI 模式和 IO3_RESET# 时的硬件复位



6.4 SDR 交流特性

表 14. SDR 交流特性

符号	参数	最小值	最大值	单位
$F_{SCK, R}$	使用于 READ 和 4READ 指令的 SCK 时钟频率	DC	50	MHz
$F_{SCK, C}$	使用于 DOR、4DOR、DIOR、4DIOR、QOR、4QOR、QIOR、4QIOR 等双线命令和四线命令的 SCK 时钟频率	DC	133	MHz
P_{SCK}	SCK 时钟周期	$1/F_{SCK}$		
t_{WH} 、 t_{CH}	时钟高电平时间	$50\% P_{SCK} - 5\%$	$50\% P_{SCK} + 5\%$	ns
t_{WL} 、 t_{CL}	时钟低电平时间	$50\% P_{SCK} - 5\%$	$50\% P_{SCK} + 5\%$	ns
t_{CRT} 、 t_{CLCH}	时钟上升时间 (转换速率)	0.1		V/ns
t_{CFT} 、 t_{CHCL}	时钟下降时间 (转换速率)	0.1		V/ns
t_{CS}	CS# 为高电平的时间 (执行读取指令时) CS# 为高电平的时间 (在使能复位功能和四线模式时执行读指令) CS# 为高电平的时间 (执行编程 / 擦除指令时)	10 20 (28) 50		ns
t_{CSS}	CS# 有效的建立时间 (相对于 SCK)	2		ns
t_{CSH}	CS# 有效的保留时间 (相对于 SCK)	3		ns
t_{SU}	数据输入的建立时间	2		ns
t_{HD}	数据输入的保留时间	3		ns
t_V	从时钟为低到输出有效的时间		8 (25) 6 (26) 6.5 (26) (29)	ns
t_{HO}	输出保留时间	1		ns
t_{DIS}	输出禁用时间 (27) 输出禁用时间 (当复位功能和四线模式均被使能时)		8 20 (28)	ns
t_{WPS}	WP# 建立时间 (24)	20		ns
t_{WPH}	WP# 保留时间 (24)	100		ns
t_{DPD}	CS# 为高电平到进入掉电模式的时间		3	μs
t_{RES}	CS# 为高电平到进入待机模式 (无电子签名读取) 的时间 读取		30	μs

注意

24. 仅在 SRWD 被设置为 1 时, 才能作为 WRR 或 WRAR 指令的限制

25. 整个 V_{CC} 范围和 $CL = 30$ pF。

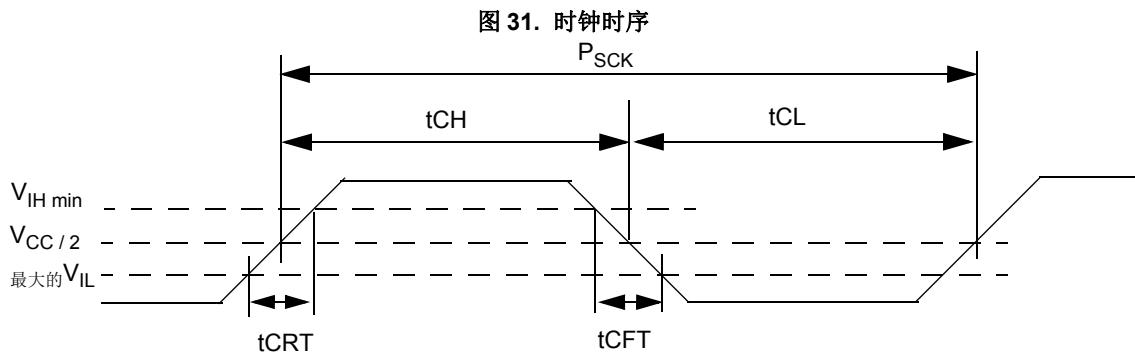
26. 整个 V_{CC} 范围和 $CL = 15$ pF。

27. 输出高阻态是数据不再被驱动的点。

28. 当使能复位功能和四线模式时 ($CR2V[5] = 1$ 和 $CR1V[1] = 1$), t_{CS} 和 t_{DIS} 需要其他时间。

29. SOIC 封装。

6.4.1 时钟时序



6.4.2 输入 / 输出时序

图 32. SPI 单比特输入时序

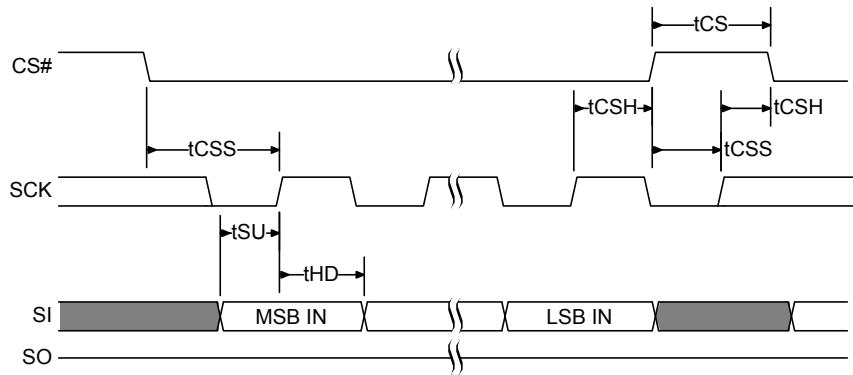


图 33. SPI 单比特输出时序

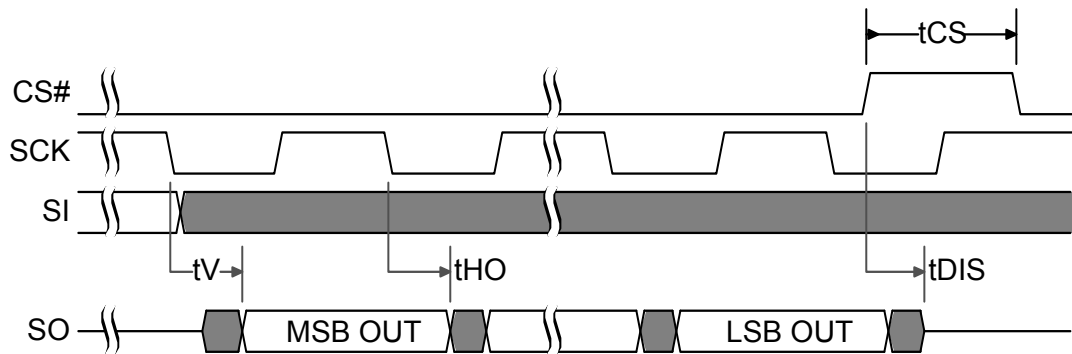


图 34. SDR MIO 时序

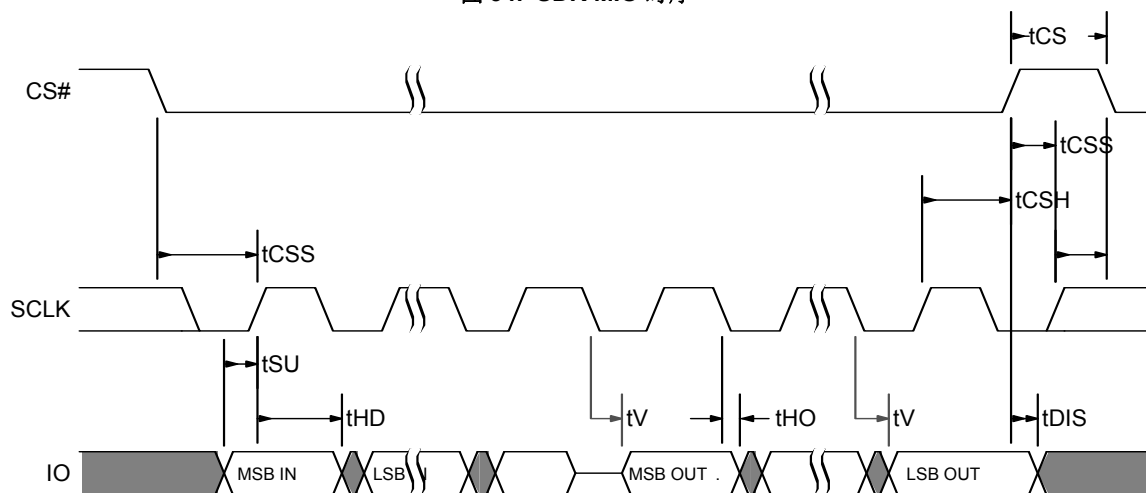
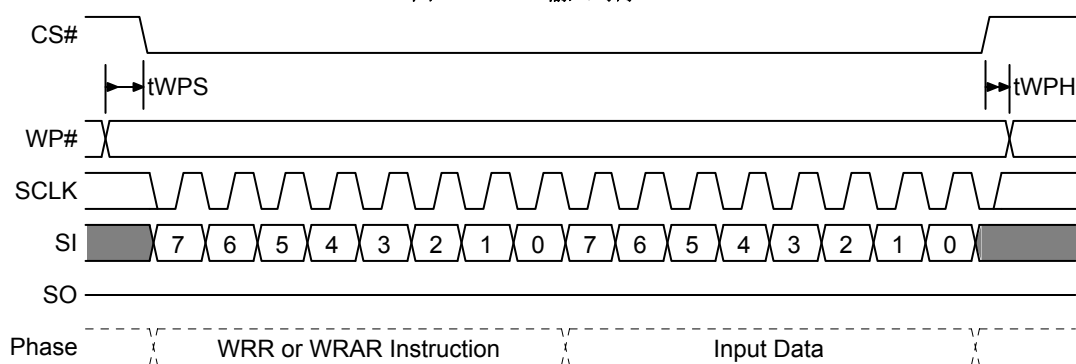


图 35. WP# 输入时序



6.5 DDR 交流特性。

表 15. DDR 交流特性

符号	参数	最小值	最大值	单位
F_{SCK_R}	使用于 DDR READ 指令的 SCK 时钟频率	DC	80	MHz
P_{SCK_R}	使用于 DDR READ 指令的 SCK 时钟周期	$1/F_{SCK}$		ns
t_{WH} 、 t_{CH}	时钟高电平时间	$45\% P_{SCK}$		ns
t_{WL} 、 t_{CL}	时钟低电平时间	$45\% P_{SCK}$		ns
t_{CS}	CS# 为高电平的时间 (执行读取指令时) CS# 为高电平的时间 (复位功能使能并执行读取指令时)	10 20		ns
t_{CSS}	CS# 有效的建立时间 (相对于 SCK)	2		ns
t_{CSH}	CS# 有效的保留时间 (相对于 SCK)	3		ns
t_{SU}	IO 输入的建立时间	1.5		ns
t_{HD}	数据输入的保留时间	1.5		ns
t_v	从时钟为低到输出有效的时间	1.5	6.0 (30) 6.5 (30) (32)	ns
t_{HO}	输出保留时间	1.5		ns
t_{DIS}	输出禁用时间 输出禁用时间 (使能复位功能时)		8 20	ns
t_{IO_skew}	从第一个 IO 数据有效到最后 IO 数据有效的时间 (31)		600 700 (32)	ps
t_{DPD}	CS# 为高电平到进入掉电模式的时间		3	μs
t_{RES}	CS# 为高电平到进入待机模式 (无电子签名读取) 的时间		30	μs

注意

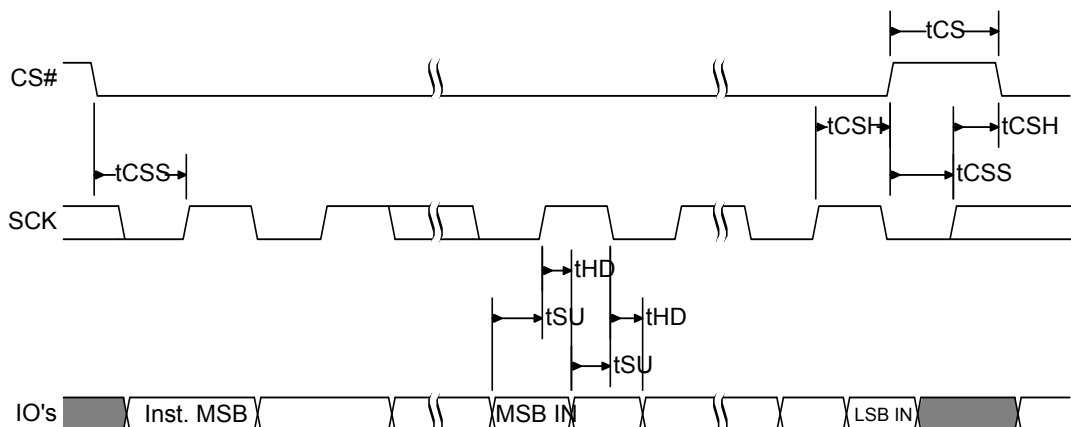
30.CL = 15 pF。

31.未测试。

32.SOIC 封装。

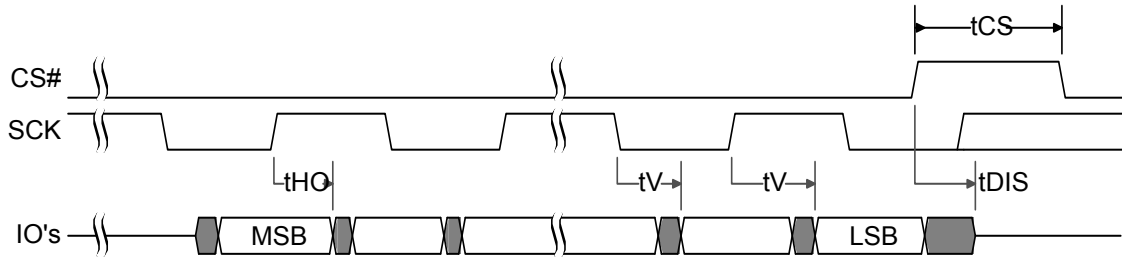
6.5.1 DDR 输入时序。

图 36. SPI DDR 输入时序



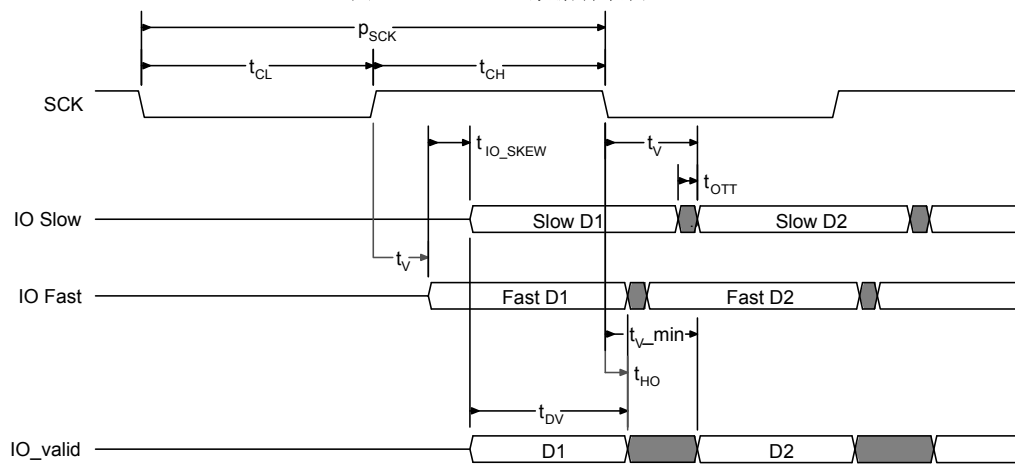
6.5.2 DDR 输出时序

图 37. SPI DDR 输出时序



6.5.3 使用 DLP 的 DDR 数据有效时序

图 38. SPI DDR 数据有效窗口



最小数据有效窗口 (t_{DV}) 和 $t_{V_{\text{minimum}}}$ 如下计算:

$$t_{DV} = \text{最小半时钟周期 } (t_{CLH})^{(33)} - t_{OTT}^{(35)} - t_{IO_SKEW}^{(34)}$$

$$t_{V_min} = t_{HO} + t_{IO_SKEW} + t_{OTT}$$

示例:

80 MHz 时钟频率 = 12.5 ns 时钟周期, 占空比为 45% 以上。

$$t_{CLH} = 0.45 \times PSCK = 0.45 \times 12.5 \text{ ns} = 5.625 \text{ ns}$$

总线阻抗为 45 Ω 和电容为 22 pf, 参考时序为 0.75 V_{CC} , 上升时间从 0 到 1 或下降时间从 1 至 0, 即 1.4⁽³⁸⁾ x RC 时间常数 (τ)⁽³⁷⁾ = 1.4 x 0.99 ns = 1.39 ns

$$t_{OTT} = \text{上升时间} + \text{下降时间} = 1.39 \text{ ns} + 1.39 \text{ ns} = 2.78 \text{ ns}。$$

有效数据窗口

$$t_{DV} = t_{CLH} - t_{IO_SKEW} - t_{OTT} = 5.625 \text{ ns} - 400 \text{ ps} - 2.78 \text{ ns} = 2.45 \text{ ns}$$

t_V 最小值

$$t_{V_min} = t_{HO} + t_{IO_SKEW} + t_{OTT} = 1.0 \text{ ns} + 400 \text{ ps} + 2.78 \text{ ns} = 4.38 \text{ ns}$$

注意

33. t_{CLH} 是 t_{CL} 或 t_{CH} 中较小的值。

34. t_{IO_SKEW} 是所有 IO 信号中最小值同最大 t_V (输出有效) 间的最大差值 (偏差)。

35. t_{OTT} 是每个 IO 上从一个有效数据值到下一个有效数据值的最大输出转换时间。 t_{OTT} 由系统级注意事项决定, 这些注意事项包括:

- 存储器器件输出阻抗 (强驱动)。
- IO 上的系统级寄生电容 (通常为总线电容)。
- 主存储器控制器输入 V_{IH} 和 V_{IL} 电平, 在该电平下可以识别 0 到 1 和 1 到 0 的转换。
- t_{OTT} 不是由赛普拉斯测试的规范。它的大小由系统决定, 并且系统设计师必须根据上述注意事项推导出该值。

36. t_{DV} 是有效数据窗口。

37. $\tau = R$ (输出阻抗) x C (负载电容)。

38. 电压上升到 V_{CC} 的 75% 的 τ 时间乘数。

7. 嵌入式算法性能表

表 16. 编程和擦除性能

符号	参数	最小值	典型值 (39)	最大值	单位
t_W	非易失性寄存器写入时间		240	750	ms
t_{PP}	页编程 (512 个字节) 页编程 (256 个字节)		475 360	2000 2000	μs
t_{SE}	扇区擦除时间 (64 KB 或 4 KB 物理扇区)		240	725	ms
	扇区擦除时间 (256 KB 逻辑扇区 = 4x64K 物理扇区)		930	2900	ms
t_{BE}	批量擦除时间 (S25FS064S)		30	94	秒
t_{EES}	评估擦除状态时间 (64 KB 或 4 KB 物理扇区)		20	25	μs
	评估擦除状态时间 (256 KB 物理或逻辑扇区)		80	100	

注意

39.在以下条件下得到典型的编程和擦除时间：温度 = 25°C、 $V_{CC} = 1.8 V$ ；使用交错式数据组合。

40.任何 OTP 编程命令的编程时间都与 t_{PP} 相同。它包含 OTP 42h、PNVDLR 43h、ASPP 2Fh 和 PASSP E8h。

41.PPBP E3h 命令的编程时间都与 t_{PP} 相同。PPBE E4h 命令的擦除时间与 t_{SE} 相同。

表 17. 编程或擦除挂起交流参数

参数	典型值	最大值	单位	注释
挂起延迟时间 (t_{SL})		50	μs	从发出挂起命令到 WIP 位为 0 的时间
恢复到下一次编程挂起的时间 (t_{RS})	100		μs	发送下一条挂起命令需要的时间，该时间要 \geq 编程或擦除操作所需的典型时间。

8. 物理接口

8.1 连接图

8.1.1 8 引脚封装

图 39. 8 引脚塑料小外型封装 (SOIC8)

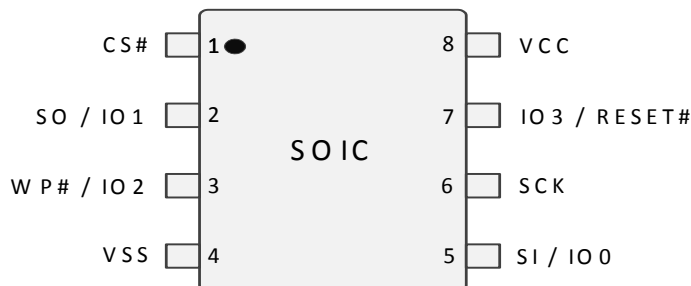
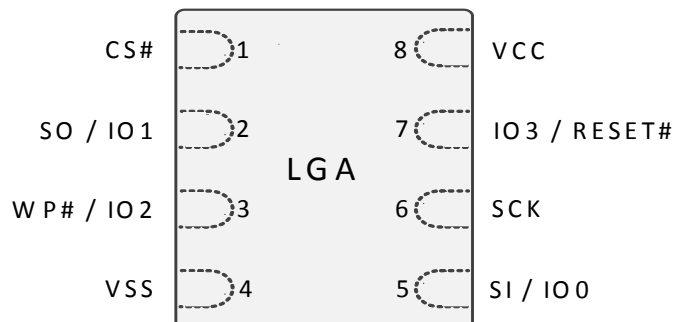


图 40. 8-Pad LGA 5x6 (W9A008), 顶视图

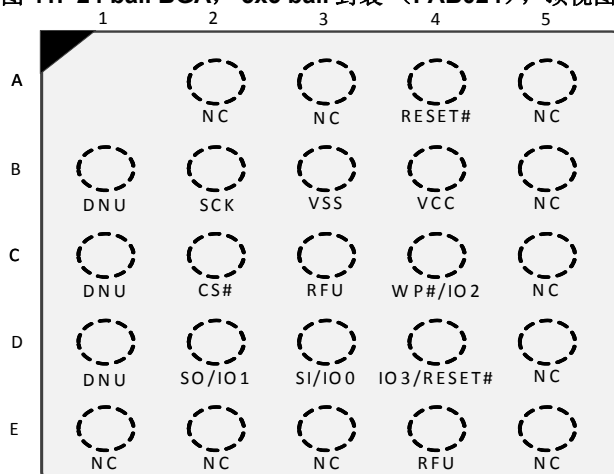


注意:

42.RESET# 输入端连接着一个内部上拉电阻。如果未使用四线模式或进行硬件复位, 那么系统中该输入会保持为未连接状态。

8.1.2 BGA ball 封装

图 41. 24 ball BGA, 5x5 ball 封装 (FAB024), 顶视图



注意:

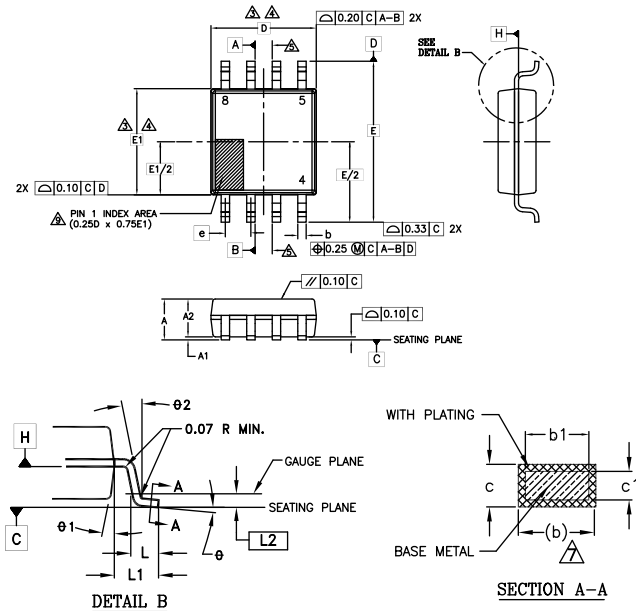
43.RESET# 输入端连接着一个内部上拉电阻。如果未使用四线模式或进行硬件复位, 那么系统中该输入会保持为未连接状态。

8.1.3 FBGA 封装的特殊处理指令说明

如果使用超声波清洁方法, BGA 封装中的闪存存储器器件可能被损坏。如果将封装长期放置在温度高于 150°C 的地方, 封装和 / 或数据的完整性可能受到损坏。

8.2 物理框图

8.2.1 SOIC 8 引脚、208 mil 宽度的封装体 (SOC008)



SYMBOL	DIMENSIONS		
	MIN.	NOM.	MAX.
A	1.75	-	2.16
A1	0.05	-	0.25
A2	1.70	-	1.90
b	0.36	-	0.48
b1	0.33	-	0.46
c	0.19	-	0.24
c1	0.15	-	0.20
D	5.28 BSC		
E	8.00 BSC		
E1	5.28 BSC		
e	1.27 BSC		
L	0.51	-	0.76
L1	1.36 REF		
L2	0.25 BSC		
N	8		
theta	0°	-	8°
theta 1	5°	-	15°
theta 2	0-8° REF		

NOTES:

- ALL DIMENSIONS ARE IN MILLIMETERS.
- DIMENSIONING AND TOLERANCING PER ASME Y14.5M - 1994.
- DIMENSION D DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 mm PER END. DIMENSION E1 DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 mm PER SIDE. D AND E1 DIMENSIONS ARE DETERMINED AT DATUM H.
- THE PACKAGE TOP MAY BE SMALLER THAN THE PACKAGE BOTTOM. DIMENSIONS D AND E1 ARE DETERMINED AT THE OUTMOST EXTREMES OF THE PLASTIC BODY EXCLUSIVE OF MOLD FLASH, TIE BAR BURRS, GATE BURRS AND INTERLEAD FLASH, BUT INCLUSIVE OF ANY MISMATCH BETWEEN THE TOP AND BOTTOM OF THE PLASTIC BODY.
- DATUMS A AND B TO BE DETERMINED AT DATUM H.
- "N" IS THE MAXIMUM NUMBER OF TERMINAL POSITIONS FOR THE SPECIFIED PACKAGE LENGTH.
- THE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 TO 0.25 mm FROM THE LEAD TIP.
- DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.10 mm TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION. THE DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE LEAD FOOT.
- THIS CHAMFER FEATURE IS OPTIONAL. IF IT IS NOT PRESENT, THEN A PIN 1 IDENTIFIER MUST BE LOCATED WITHIN THE INDEX AREA INDICATED.
- LEAD COPLANARITY SHALL BE WITHIN 0.10 mm AS MEASURED FROM THE SEATING PLANE.

THIS DRAWING CONTAINS INFORMATION WHICH IS THE PROPRIETARY PROPERTY OF CYPRESS SEMICONDUCTOR CORPORATION. THIS DRAWING IS RECEIVED IN CONFIDENCE AND ITS CONTENTS MAY NOT BE DISCLOSED WITHOUT WRITTEN CONSENT OF CYPRESS SEMICONDUCTOR CORPORATION.


PACKAGE CODE(S) **SOC008**

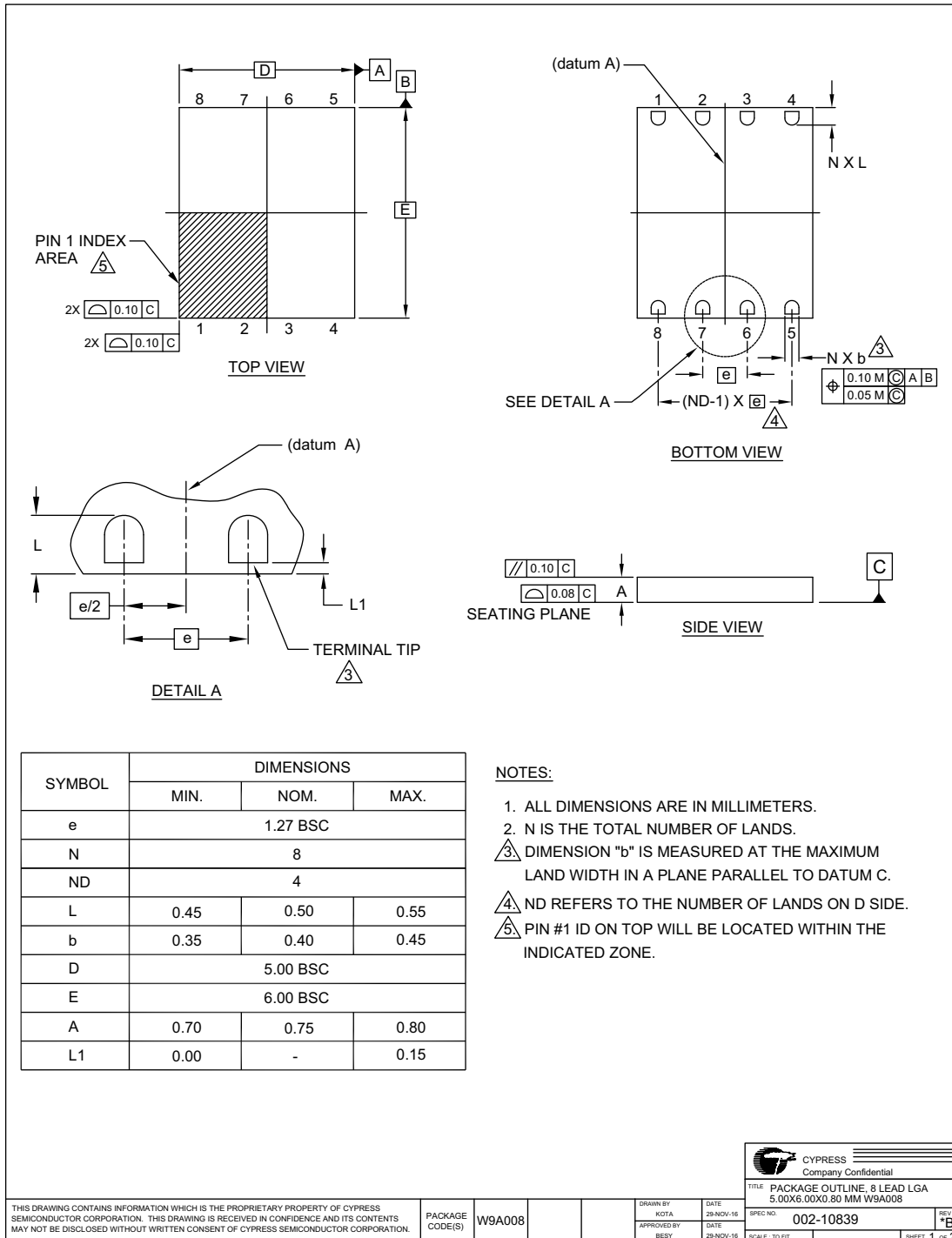
DRAWN BY: **MDTA**

DATE: **18-JUL-16**

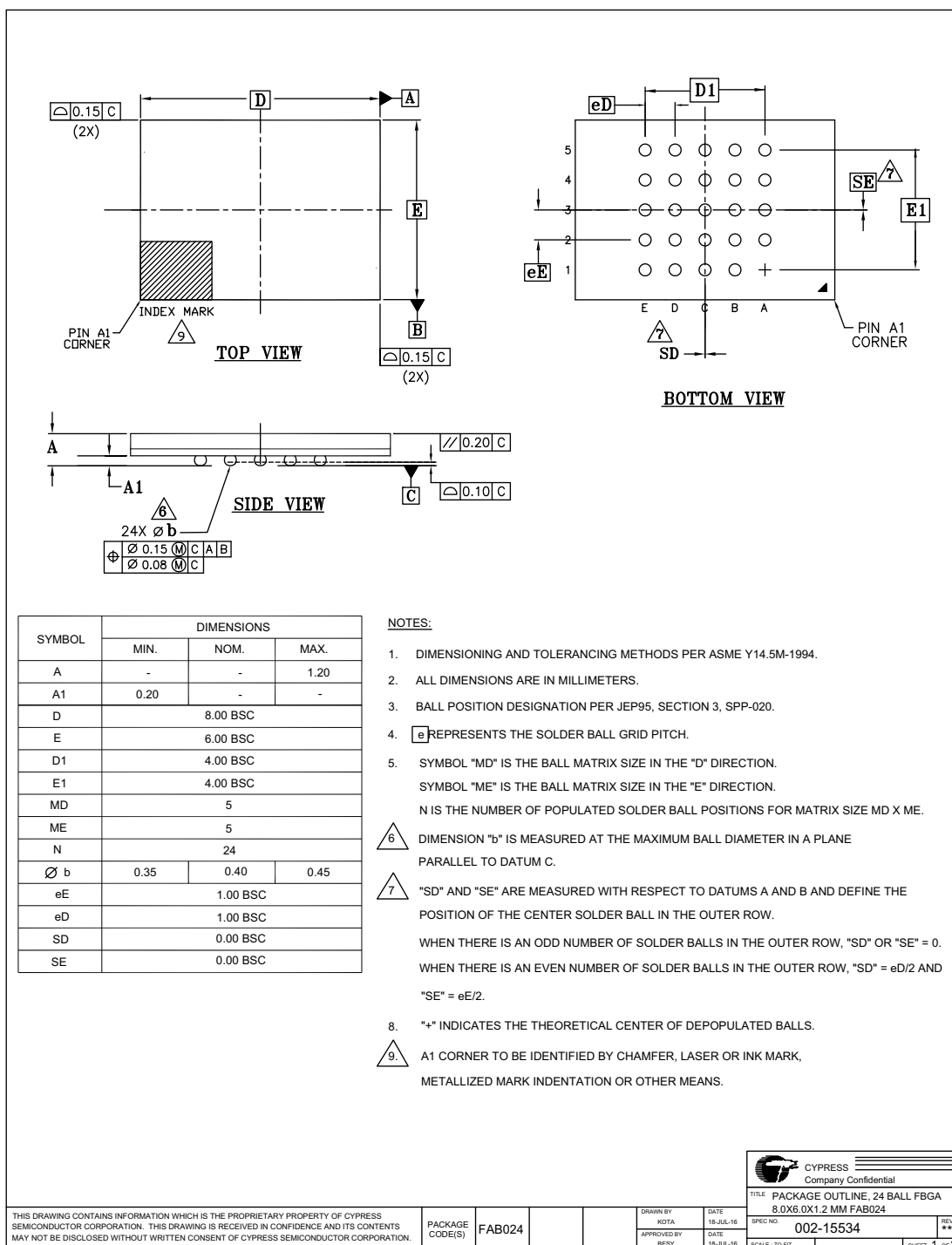
APPROVED BY: **BESY**

DATE: **18-JUL-16**

 CYPRESS Company Confidential	
TITLE: PACKAGE OUTLINE, 8 LEAD SOIC 5.28X5.28X2.16 MM SOC008	
SPEC NO: 002-15548	REV: **
SCALE: TO FIT	SHEET 1 OF 2

8.2.2 LGA 8-contact 5 x 6 mm (W9A008)


8.2.3 球栅阵列 24 ball 6 x 8 mm (FAB024)



软件接口

本节介绍了与主机系统软件最紧密相关的功能和行为，这些软件与存储器器件 FS-S 系列相互配合。

9. 地址空间映射

9.1 概述

9.1.1 扩展地址

FS-S 系列支持 32 位（4 个字节）地址。这样，同前一代（旧版）SPI 器件相比，器件容量更高（前代 SPI 器件仅支持 24 位（3 个字节）地址。对于 24 位、字节分辨率系统，地址寄存器允许对 16 MBytes（128 Mb）大的地址空间进行访问。对于 32 位、字节分辨率，地址寄存器允许直接对 4 GBytes（32 Gbits）大的地址空间进行访问。并且允许容量为 4 MB（32 Mb）到 4 GB（32 Gb）的器件与软件兼容。

传统的命令仍支持 24 位地址，以实现软件向后兼容。通过以下两种方法可以获取扩展的 32 位地址：

- 扩展地址模式 — 一个易失性配置寄存器位，它更改了所有传统命令，以获取由主机系统提供的 32 位地址。
- 4 字节地址命令 — 执行传统的和新的功能，从而能够获取 32 位地址。

上电或复位后，扩展地址模式的默认条件是由非易失性配置位控制的。可将默认的扩展地址模式设置为 24 或 32 位地址。这样可以对器件的前 128 Mb 空间进行传统的软件兼容访问，并允许器件在 32 位地址模式下直接启动。

FS-S 系列的 64 Mb 容量器件以相同的方式支持扩展地址性能，但本质上它忽略了所有地址的位 31 到位 23 或位 22。这是因为主闪存阵列只需要地址的 23 或 22 位。这样便允许将 64 Mb 容量器件移植到容量更高的器件内，而不需要更改软件的地址处理特性。

9.1.2 多地址空间

许多命令在主闪存存储器阵列上运行。某些命令在独立于主闪存阵列的地址空间内运行。每个单独地址空间使用了整个 24 或 32 位地址，但只定义可用地址空间的一小部分。

9.2 闪存存储器阵列

主闪存阵列被分为称为物理扇区的擦除单元。

可以将 FS-S 系列物理扇区配置为地址空间顶部或底部上 8 个 4 KB 参数扇区的混合组，所有扇区均为统一大小（被划分的扇区除外）。由于 8 个 4 KB 参数扇区的总和小于一个统一的扇区，因此 4 KB 物理扇区组将分别覆盖（替代）最高或最低地址统一扇区的顶部或底部 32 KB。

必须使用参数扇区擦除命令（20h 或 21h）来单独擦除 4 KB 扇区。必须使用扇区（统一模块）擦除命令（D8h 或 DCh）擦除剩下的所有扇区，包括最高或最低地址扇区部分（这些扇区不被参数扇区覆盖掉）。统一模块擦除命令不会影响到参数扇区。

非易失性配置寄存器 1 的位 2（CR1NV[2]）= 0 时，将覆盖最低地址统一扇区底部上的参数扇区。CR1NV[2] = 1 时，将覆盖最高地址统一扇区顶部的参数扇区。相关的详细信息，请参见第 48 页上的 9.6 寄存器节。

通过一个配置选项可以清除地址映射中的 4 KB 参数扇区，以便统一所有扇区的大小。配置寄存器 3 的易失性位 3（CR3V[3]）= 0 时，将选择带有 4 KB 参数扇区的混合扇区架构。CR3V[3] = 1 时，将选择无参数扇区的统一扇区架构。统一物理扇区为：

■ 64 KB 或 256 KB

可以将这些器件配置为使用扇区（统一模块）擦除命令擦除 256 KB 逻辑模块，而不是单独的 64 KB 物理扇区。通过该配置选项（CR3V[1] = 1），容量更低的器件可以模拟该产品系列中更大容量器件的相同扇区的擦除性能（该产品系列使用 256 KB 物理扇区）。这样可以简化移植到容量更高器件的过程。

表 18. S25FS064S 的扇区和存储器地址映射（4 KB 扇区位于底部上）

扇区大小 (KB)	扇区数量	扇区范围	地址范围 (字节地址)	注意
4	8	SA00	00000000h-00000FFFh	扇区起始地址 — 扇区结束地址
		:	:	
		SA07	00007000h-00007FFFh	
32	1	SA08	00008000h-0000FFFFh	
64	127	SA09	00010000h-0001FFFFh	
		:	:	
		SA135	007F0000h-007FFFFFh	

表 19. S25FS064S 的扇区和存储器地址映射（4 KB 扇区位于顶部上）

扇区大小 (KB)	扇区数量	扇区范围	地址范围 (字节地址)	注意
64	127	SA00	00000000h-000FFFFFh	扇区起始地址 — 扇区结束地址
		:	:	
		SA126	007E0000h-007EFFFFh	
32	1	SA127	007F0000h - 007F7FFFh	
4	8	SA128	007F8000h - 007F8FFFh	
		:	:	
		SA135	007FF000h-007FFFFFFh	

表 20. S25FS064S 的扇区和存储器地址映射（数据块大小统一为 64 KB）

扇区大小 (KB)	扇区数量	扇区范围	地址范围 (字节地址)	注意
64	128	SA00	00000000h-0000FFFFh,	扇区起始地址 — 扇区结束地址
		:	:	
		SA127	007F0000h-07FFFFFFh	

表 21. S25FS064S 扇区地址映射、4 KB 的底部扇区、256 KB 的逻辑统一扇区

扇区大小 (KB)	扇区数量	扇区范围	地址范围 (字节地址)	注意
4	8	SA00	00000000h-00000FFFh	扇区起始地址 — 扇区结束地址
		:	:	
		SA07	00007000h-00007FFFh	
224	1	SA08	00008000h-0003FFFFh	
256	31	SA09	00040000h-0007FFFFh	
		:	:	
		SA39	007C0000h-007FFFFFFh	

表 22. S25FS064S 扇区地址映射、4 KB 的顶部扇区、256 KB 的逻辑统一扇区

扇区大小 (KB)	扇区数量	扇区范围	地址范围 (字节地址)	注意
256	31	SA00	00000000h-0003FFFFh	扇区起始地址 — 扇区结束地址
		:	:	
		SA30	00780000h-007BFFFFh	
224	1	SA31	007C0000h-007F7FFFh	
4	8	SA32	007F8000h-007F8FFFh	
		:	:	
		SA39	007FF000h-007FFFFFFh	

表 23. S25FS064S 的扇区和存储器地址映射 (数据块大小统一为 256 KB)

扇区大小 (KB)	扇区数量	扇区范围	地址范围 (字节地址)	注意
256	32	SA00	00000000h-0003FFFFh	扇区起始地址 — 扇区结束地址
		:	:	
		SA31	007C0000h-007FFFFFFh	

注意： 这些表格使用了一些扇区作为参考。某些地址范围未被明显列出。所有 4 KB 扇区的组合均为 XXXX000h-XXXXFFFh。所有 64 KB 扇区的组合均为 XXX0000h-XXXFFFh。所有 256 KB 扇区的组合均为 XX00000h-XX3FFFh、XX40000h-XX7FFFh、XX80000h-XXCFFFh 或 XxD0000h-XXFFFFh。

9.3 ID-CFI 地址空间

RDID 命令 (9Fh) 从单独的闪存存储器地址空间内读取信息，以获取器件标识 (ID) 和通用闪存接口 (CFI) 的信息。请参考第 122 页上的器件 ID 和通用闪存接口 (ID-CFI) 地址映射 — 标准，了解定义 ID-CFI 地址空间内容的表格。该 ID-CFI 地址空间由赛普拉斯编程，主机系统只能对其执行读取操作。

9.3.1 赛普拉斯编程的唯一 ID

64 位唯一编号位于唯一器件 ID 地址空间中的 8 个字节内。该唯一 ID 可以作为每个器件的唯一软件可读序列号。

9.4 JEDEC JESD216 串行闪存可发现参数 (SFDP) 空间

根据串行闪存可发现参数的 JEDEC JESD216 版本 B 标准，RSFDP 命令 (5Ah) 从单独闪存存储器地址空间读取信息，以获取器件标识、特性和配置信息。ID-CFI 地址空间被组合为某个 SFDP 参数。请访问第 122 页上的软件接口参考，查看描述 SFDP 地址空间定义的表格。该 SFDP 地址空间由赛普拉斯编程，主机系统只能对其进行读取操作。

9.5 OTP 地址空间

每个 FS-S 系列存储器器件具有一个独立于主闪存阵列的 1024 字节的一次性编程（OTP）地址空间。OTP 区域分为 32 个可独立锁定并且长度为 32 字节的对齐区域。

在起始地址为 0 的 32 字节区域中：

- 低 16 个地址字节由赛普拉斯通过一个 128 位随机数来编程。只有赛普拉斯才能将 0 编程到这些字节内。会忽略将 1 编程到这些字节地址的操作，并且不会影响赛普拉斯所编程的值。尝试将 0 编程到这些字节地址的操作会失败，并设置 P_ERR。
- 后面的 4 个高地址字节（OTP 锁定字节）分别为每个 OTP 区域提供一个位，以便永久性阻止对每个区域进行的编程。从赛普拉斯出厂时，这些字节都被擦除。编程某个 OTP 区域后，可通过锁定该区域来防止发生额外的编程。通过编程 OTP 锁定字节中的相关保护位，可以锁定所需区域。
- 最低地址范围后面的 12 个高字节被保留，以供将来使用（RFU）。主机系统可能对这些 RFU 字节中的各个位进行编程处理，但必须注意，新器件可能使用这些位来保护范围更大的 OTP 空间。从赛普拉斯出厂时，这些字节都被擦除。

从赛普拉斯出厂时，剩余的区域也被擦除。此外，还可以使用这些区域编程额外的永久数据。

欲了解 OTP 存储器空间的图形表示，请参见第 47 页上的图 42。

OTP 存储器空间用于增强系统安全性。OTP 值（比如由赛普拉斯编程的随机数）可用来连接闪存组件和系统 CPU/ASIC，以防止器件被替换。

将配置寄存器的冻结位（CR1V[0]）设置为“1”时，便不能对整个 OTP 存储器空间进行编程操作。因此可靠的启动代码可以控制 OTP 区域的编程程序，然后设置冻结位。这样，在通用上电系统操作剩余过程内可以防止对 OTP 存储器空间执行编程操作。

图 42. OTP 地址空间

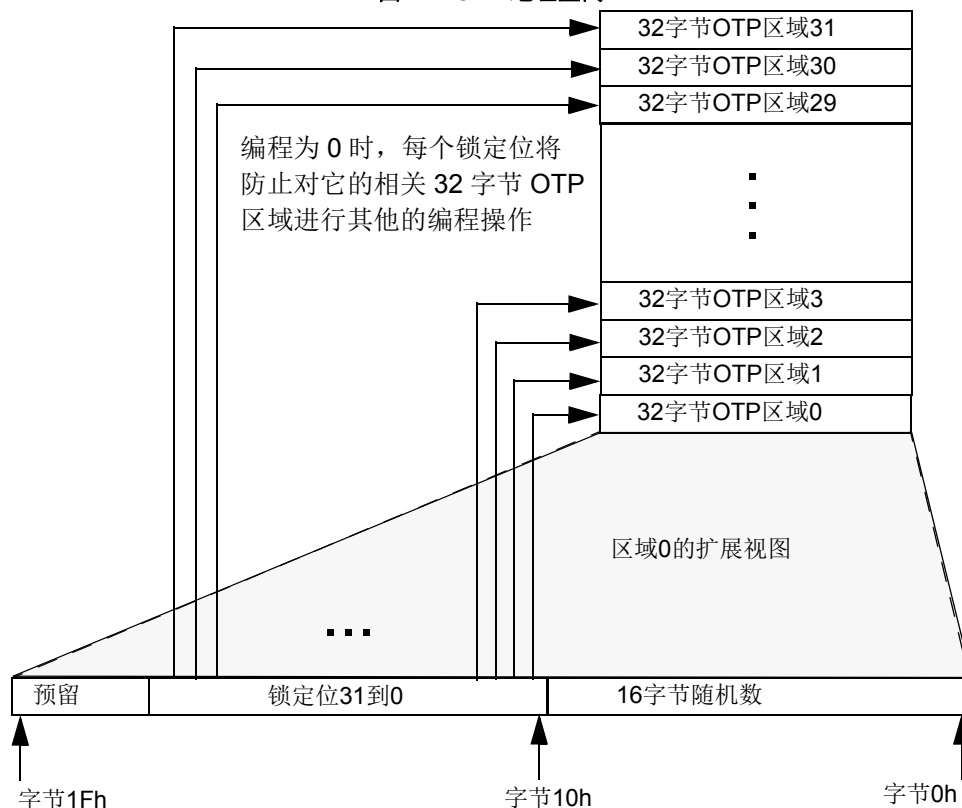


表 25. OTP 地址映射

区域	字节地址范围（十六进制）	内容	出厂初始状态（十六进制）
区域 0	000	赛普拉斯编程随机数值的最低有效字节	赛普拉斯编程随机数值
	
	00F	赛普拉斯编程随机数值的最高有效字节	
	010 到 013	区域锁定位 字节 10[位 0] = 0 时，它将锁定区域 0，阻止对该区域进行编程操作 ... 字节 13[位 7] = 0 时，它将锁定区域 31，禁止向该区域进行编程操作	所有字节 = FF
	014 到 01F	预留供将来使用（RFU）	所有字节 = FF
区域 1	020 到 03F	可供用户编程	所有字节 = FF
区域 2	040 到 05F	可供用户编程	所有字节 = FF
...	...	可供用户编程	所有字节 = FF
区域 31	300 到 3FF	可供用户编程	所有字节 = FF

9.6 寄存器

寄存器是存储器单元的小组，用于配置 FS-S 系列存储器器件的运行方式以及报告器件的运行状态。通过使用特殊命令可以访问这些寄存器。用于每个寄存器的命令（和十六进制指令代码）被注释在每个寄存器的说明内。

在旧版的 SPI 存储器器件中，独立寄存器位可以是同一个寄存器中易失性、非易失性或一次性可编程（OTP）位的混合体。在某些配置选项中，寄存器位的类型会发生变化，例如，从非易失性变为易失性。

FS-S 系列使用单独的非易失性或易失性存储器单元组（区域）来实现不同类型的寄存器位。然而，继续显示了传统寄存器和各种命令，它们始终有效，从而实现与传统软件的兼容。当传统寄存器具有易失性位或读取传统寄存器的命令具有 0 读取延迟时，每个传统寄存器有非易失性和易失性版本。当读取这种寄存器时，将提供该寄存器的易失性版本。在上电复位（POR）、硬件复位或软件复位期间，寄存器的非易失性版本被复制为易失性版本，以提供易失性寄存器的默认状态。当写入非易失性寄存器位时，寄存器的非易失性版本将被擦除并编程新的位值，而且寄存器的易失性版本将更新非易失性版本的新内容。当编程 OTP 位时，寄存器的非易失性版本将被编程，而且寄存器易失性版本中的相应位被更新。当写入易失性寄存器位时，只有寄存器易失性版本中的相应位被更新。

每一位的类型注释均位于每个寄存器的说明内。每一位的默认状态指的是执行上电复位、硬件复位或软件复位后的状态（如果它是易失性位）。如果该位是非易失性或 OTP 位，则默认状态将为器件从赛普拉斯出厂时的位值。非易失性位和主闪存阵列的周期耐久性（擦除和编程）是相同的。

表 26. 寄存器说明

寄存器	类型	位	缩写
状态寄存器 1	非易失性	7:0	SR1NV[7:0]
	易失性	7:0	SR1V[7:0]
状态寄存器 2	易失性	7:0	SR2V[7:0]
配置寄存器 1	非易失性 /OTP	7:0	CR1NV[7:0]
	易失性	7:0	CR1V[7:0]
配置寄存器 2	非易失性 /OTP	7:0	CR2NV[7:0]
	易失性	7:0	CR2V[7:0]
配置寄存器 3	非易失性 /OTP	7:0	CR3NV[7:0]
	易失性	7:0	CR3V[7:0]
配置寄存器 4	非易失性 /OTP	7:0	CR4NV[7:0]
	易失性	7:0	CR4V[7:0]
ECC 状态寄存器	易失性 只读	7:0	ECCSRV[7:0]
ASP 寄存器	OTP	15:0	ASPR[15:0]
密码寄存器	OTP	63:0	PASS[63:0]
PPB 锁定寄存器	易失性 只读	7:0	PPBL[7:0]
PPB 访问寄存器	非易失性	7:0	PPBAR[7:0]
DYB 访问寄存器	易失性	7:0	DYBAR[7:0]
SPI DDR 数据学习寄存器	OTP	7:0	NVDLR[7:0]
	易失性	7:0	VDLR[7:0]

9.6.1 状态寄存器 1

9.6.1.1 非易失性状态寄存器 1 (SR1NV)

相关命令：写寄存器 (WRR 01h)、读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)

表 27. 非易失性状态寄存器 1 (SR1NV)

位	字段名称	功能	类型	默认状态	说明
7	SRWD_NV	默认的状态寄存器写入禁用	非易失性	0	1 = WP# 为低电平时 SRWD、BP 和配置寄存器 1 的锁存状态，通过不执行影响 SR1NV、SR1V、CR1NV 或 CR1V 的 WRR 或 WRAR 命令实现该操作。 0 = 不受保护，即使 WP# 为低电平
6	P_ERR_D	默认的编程错误	非易失性 只读	0	为编程错误状态提供默认状态。用户不能编程该位。
5	E_ERR_D	默认的擦除操作	非易失性 只读	0	为擦除错误状态提供默认状态。用户不能编程该位。
4	BP_NV2	非易失性块保护	非易失性	000b	BP 位被配置为非易失性 (CR1NV[3] = 0) 时，防止对已选定扇区范围 (块) 进行编程或擦除操作。当 BP 位被配置为易失性 (CR1NV[3] = 1) 时，这些位将被编程为 111b。这样用户便不能再编程这些位。
3	BP_NV1				
2	BP_NV0				
1	WEL_D	WEL 的默认设置	非易失性 只读	0	为 WEL 状态位提供默认状态。用户不能编程该位。
0	WIP_D	WIP 的默认设置	非易失性 只读	0	为 WIP 状态位提供默认状态。用户不能编程该位。

非易失性的写状态寄存器 (SRWD_NV) SR1NV[7]: 当该位为 1 并且 WP# 输入为低电平时, 器件将处于硬件保护模式。在该模式下, 由于这些寄存器变为只读寄存器, 因此, 选择状态寄存器 1 或配置寄存器 1 的写入任何寄存器 (WRAR) 命令均被忽略且不被执行。从而有效地锁存状态寄存器 1 和配置寄存器 1 (SR1NV、SR1V、CR1NV 或 CR1V) 位。如果 WP# 为高电平, 则 WRR 或 WRAR 命令可能会更改状态寄存器 1 和配置寄存器 1。如果 SRWD_NV 为 0, WP# 将无效而且 WRR 或 WRAR 命令可能会更改状态寄存器 1 和配置寄存器 1。WP# 不对任何其他寄存器的写操作产生影响。SRWD_NV 位的非易失性耐久性和主闪存阵列的相同。SRWD (SR1V[7]) 位作为 SRWD_NV 位的副本, 以提供 0 读取延迟。

默认的编程错误状态 (P_ERR_D) SR1NV[6]: 为 SR1V[6] 中的编程错误状态提供默认状态。用户不能编程该位。

擦除错误 (E_ERR) SR1V[5]: 为 SR1V[5] 中的擦除错误状态提供默认状态。用户不能编程该位。

块保护 (BP_NV2、BP_NV1、BP_NV0) SR1NV[4:2]: 这些位定义了防止编程和擦除操作的主闪存阵列区域。根据配置寄存器 CR1NV[3] 中的 BP 非易失性位 (BPNV_O) 的状态, 可以将 BP 位设置为易失性或非易失性。CR1NV[3] = 0 时, 可以使用 BP 位的非易失性版本 (SR1NV[4:2]) 来控制块保护和 WRR 写命令 SR1NV[4:2], 并将 SR1V[4:2] 更新到相同值。CR1NV[3] = 1 时, 可以使用 BP 位的易失性版本 (SR1V[4:2]) 来控制块保护和 WRR 写命令 SR1V[4:2] 而不会影响 SR1NV[4:2]。当一个或多个 BP 位被设置为 1 时, 便不能对相关存储区进行编程和擦除操作。仅当 BP 位被清除为 0 时, 才能执行批量擦除 (BE) 命令。第 65 页上的块保护说明了 BP 位值如何选择受保护的存储器阵列区域。BP 位的非易失性版本具有与主闪存阵列相同的非易失性耐久性。

写使能锁存默认 (WEL_D) SR1NV[1]: 为 SR1V[1] 中的 WEL 状态提供默认状态。该位由赛普拉斯编程, 用户不能编程它。

正在写入的默认位 (WIP_D) SR1NV[0]: 为 SR1V[0] 中的 WIP 状态提供默认状态。该位由赛普拉斯编程, 用户不能编程它。

9.6.1.2 易失性状态寄存器 1 (SR1V)

相关命令: 读取状态寄存器 (RDSR1 05h)、写入寄存器 (WRR 01h)、写使能 (WREN 06h)、写禁用 (WRDI 04h)、清除状态寄存器 (CLSR 30h 或 82h)、读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。通过使用 RDSR1 命令可以显示该寄存器。

表 28. 状态寄存器 1 易失性 (SR1V)

位	字段名称	功能	类型	默认状态	说明
7	SRWD	状态寄存器 写入操作禁用	易失性 只读	SR1NV	SR1NV[7] 的易失性副本
6	P_ERR	发生了编程 错误	易失性 只读		1 = 发生了错误 0 = 无错误
5	E_ERR	发生了擦除 错误	易失性 只读		1 = 发生了错误 0 = 无错误
4	BP2	块保护易失性	易失性		当 BP 位被配置为易失性（CR1NV[3] = 1）时，防止对已选定的扇区范围（块）进行编程或擦除操作。将 BP 位编程为非易失性时，这些位是 SR1NV[4:2] 的易失性副本。当 BP 位被配置为易失性时，用户可以对这些位进行写操作。
3	BP1				
2	BP0				
1	WEL	写使能锁存	易失性		1 = 器件接收写入寄存器（WRR 和 WRAR）、编程或擦除命令 0 = 器件忽略写入寄存器（WRR 和 WRAR）、编程或擦除命令 该位不受 WRR 和 WRAR 的影响，但受 WREN 和 WRDI 命令的影响。
0	WIP	正在执行写入 操作	易失性 只读		1 = 器件处于忙碌状态，正在执行编程或擦除等嵌入式操作 0 = 器件已经处于待机模式，可以接收命令 该位不受 WRR 或 WRAR 的影响，它仅提供 WIP 状态。

写入状态寄存器 (SRWD) SR1V[7]: SRWD 是 SR1NV[7] 的易失性副本。它跟踪该位非易失性版本的更改。

编程错误 (P_ERR) SR1V[6]: 编程错误位用于指出编程操作的成功或失败。当编程错误位为 “1” 时, 则表示最后编程操作有错误。当用户尝试在受保护的主存储器扇区或一个被锁定的 OTP 区域中进行编程时, 该位将被设置。当编程错误位为 “1” 时, 通过使用清除状态寄存器 (CLSR) 命令可以将该位清除为 0。这是一个只读位, 它不受 WRR 或 WRAR 命令的影响。

擦除错误 (E_ERR) SR1V[5]: 擦除错误位用于指出擦除操作的成功或失败。当擦除错误位为 “1” 时, 则表示最后擦除操作有错误。当用户尝试擦除受保护的单独主存储器扇区时, 该位将被设置。如果在执行命令期间找到受保护的扇区时, 批量擦除命令将不设置 E_ERR。当擦除错误位为 “1” 时, 通过使用清除状态寄存器 (CLSR) 命令可以将该位清除为 0。这是一个只读位, 它不受 WRR 或 WRAR 命令的影响。

块保护 (BP2、BP1、BP0) SR1V[4:2]: 这些位定义了防止编程和擦除操作的主闪存阵列区域。根据配置寄存器 CR1NV[3] 中的 BP 非易失性位 (BPNV_O) 的状态, 可以将 BP 位设置为易失性或非易失性。CR1NV[3] = 0 时, 可以使用 BP 位的非易失性版本 (SR1NV[4:2]) 来控制块保护和 WRR 写命令 SR1NV[4:2], 并将 SR1V[4:2] 更新到相同值。CR1NV[3] = 1 时, 可以使用 BP 位的易失性版本 (SR1V[4:2]) 来控制块保护和 WRR 写命令 SR1V[4:2] 而不会影响 SR1NV[4:2]。当一个或多个 BP 位被设置为 1 时, 便不能对相关存储区进行编程和擦除操作。仅当 BP 位被清除为 0 时, 才能执行批量擦除 (BE) 命令。第 65 页上的块保护说明了 BP 位值如何选择受保护的存储器阵列区域。

写使能锁存 (WEL) SR1V[1]: 必须将 WEL 位设置为 1, 这样才能使能编程、写入或擦除操作, 并能够防止对存储器或寄存器值进行意外更改。执行写使能 (WREN) 命令会将写使能锁存位置 1, 这样便能执行后面所有的编程、擦除或写命令。通过写禁用 (WRDI) 命令, 将写使能锁存位置 0, 这样可以防止执行编程、擦除和写命令。编程、写入或擦除操作成功后, WEL 位将被清除为 0。如果这些操作失败, WEL 位将保持设置状态, 应该依次使用 WRDI 命令和 CLSR 命令来清除它。掉电 / 上电序列、硬件复位或软件复位发生后, 写使能锁存位将被设置为 “0”。WRR 或 WRAR 命令不会影响该位的状态。

正在写入 (WIP) SR1V[0]: 表示器件是否正在执行编程、写入、擦除或其他操作, 器件在这段时间内会忽略新的操作命令。当该位被设置为 “1” 时, 器件会执行一个操作。当 WIP 为 “1” 时, 仅接收读取状态寄存器 (RDSR1、RDSR2)、读取任何寄存器 (RDAR)、擦除挂起 (ERSP)、编程挂起 (PGSP)、清除状态寄存器 (CLSR) 和软件复位 (RESET) 等命令。只有正在对存储器阵列进行擦除或编程操作时, ERSP 和 PGSP 命令才被接收。当 WIP = 1 时, 状态寄存器 E_ERR 和 P_ERR 位将被更新。当 P_ERR 或 E_ERR 位为 1 时, WIP 位将保持设置为 1, 以表示器件的繁忙状态并且无法接收新的操作命令。必须接收清除状态寄存器 (CLSR) 命令, 这样可以使器件返回待机模式。将 WIP 位清除为 0 时, 表示没有任何操作正在进行。这是一个只读位。

9.6.2 状态寄存器 2 易失性 (SR2V)

相关命令: 读取状态寄存器 2 (RDSR2 07h)、读取任何寄存器 (RDAR 65h)。状态寄存器 2 中没有用户可编程的非易失性位, 它的所有定义位都是易失性只读状态位。这些位的默认状态是由硬件设置的。

表 29. 状态寄存器 2 易失性 (SR2V)

位	字段名称	功能	类型	默认状态	说明
7	RFU	预留		0	预留供将来使用
6	RFU	预留		0	预留供将来使用
5	RFU	预留		0	预留供将来使用
4	RFU	预留		0	预留供将来使用
3	RFU	预留		0	预留供将来使用
2	ESTAT	擦除状态	易失性 只读	0	1 = 扇区擦除状态命令结果 = 擦除完全 0 = 扇区擦除状态命令结果 = 擦除没有完成
1	ES	擦除挂起	易失性 只读	0	1 = 处于擦除挂起模式。 0 = 未处于擦除挂起模式。
0	PS	编程挂起	易失性 只读	0	1 = 处于编程挂起模式。 0 = 未处于编程挂起模式。

擦除状态 (ESTAT) SR2V[2]: 擦除状态位指出由前一个擦除状态命令所选择的扇区是否完成最后的擦除命令。在读取 SR2V 前, 必须先发送擦除状态命令, 以获取有效的擦除状态。在编程或擦除挂起期间读取 SR2V 不会获取有效的擦除状态。系统软件可以使用擦除状态位来检测最后擦除操作发生故障的所有扇区。使用该命令可以检测擦除操作失败的原因, 原因可能是在执行擦除操作期间发生断电。

擦除挂起 (ES) SR2V[1]: 挂起状态位指示器件是否是在擦除或编程挂起状态。用户不能写入该状态位。当擦除挂起位为 1 时, 器件将处于擦除挂起模式。当擦除挂起位为 0 时, 则器件不处于擦除挂起模式。请参考第 107 页上的擦除或编程挂起 (EPS 85h、75h、B0h), 了解擦除挂起 / 恢复命令的详细内容。

编程挂起 (PS) SR2V[0]: 编程挂起位用于指出器件是否处于编程挂起模式。用户不能写入该状态位。当编程挂起位为 1 时, 则器件处于编程挂起模式。当编程挂起位为 0 时, 则器件不处于编程挂起模式。更多信息, 请查阅第 107 页上的擦除或编程挂起 (EPS 85h、75h、B0h)。

9.6.3 配置寄存器 1

配置寄存器 1 控制某些接口和数据保护功能。通过使用带有十六个输入周期的 WRR 命令或通过使用 WRAR 命令可以更改这些寄存器位。

9.6.3.1 非易失性配置寄存器 1 (CR1NV)

相关命令：写寄存器 (WRR 01h)、读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。

表 30. 非易失性配置寄存器 1 (CR1NV)

位	字段名称	功能	类型	默认状态	说明
7	RFU	预留供将来使用	非易失性	0	预留
6	RFU			0	
5	TBPROT_O	配置块保护的起始地址	OTP	0	1 = BP 从底部 (低地址) 开始 0 = 从顶部 (高地址) 开始执行块保护
4	RFU	预留供将来使用	RFU	0	预留
3	BPNV_O	配置状态寄存器中的 BP2-0	OTP	0	1 = 易失性 0 = 非易失性
2	TBPARAM_O	配置参数扇区位置	OTP	0	1 = 4 KB 的物理扇区位于顶部 (高位地址) 0 = 4 KB 的物理扇区位于底部 (低位地址) 统一扇区配置中的 RFU。
1	QUAD_NV	四线非易失性	非易失性	0	为 QUAD 位提供默认状态。
0	FREEZE_D	FREEZE 的默认设置	非易失性 只读	0	为 Freeze 位提供默认状态。用户不能编程该位。

顶部或底部保护 (TBPROT_O) CR1NV[5]: 该位定义状态寄存器中块保护位 BP2、BP1 和 BP0 的操作。如状态寄存器部分中所述，通过使用 BP2-0 位，用户可以选择保护阵列的某一部分 (1/64、1/4、1/2 等等) 或整个阵列。将 TBPROT_O 设置为 “0” 时，块保护被定义为从该阵列的顶部 (最大地址) 开始。将 TBPROT_O 设置为 “1” 时，块保护被定义为从该阵列的底部 (0 地址) 开始。TBPROT_O 位为 OTP，从赛普拉斯出厂时，它便被设置为 “0”。如果将 TBPROT_O 编程为 1，那么向该位写入 0 将不能修改该值或设置编程错误位 (SR1V[6] 中的 P_ERR)。

在系统制造的器件初始配置期间，以及第一次对主闪存阵列进行编程或擦除操作前必须选择 TBPROT_O 的状态。对主闪存阵列进行编程或擦除操作后，不能编程 TBPROT_O。

非易失性块保护 (BPNV_O) CR1NV[3]: BPNV_O 位定义了选择状态寄存器中的 BP_NV 2-0 位还是 BP 2-0 位用于控制块保护功能。BPNV_O 位为 OTP。从赛普拉斯出厂时，BPNV_O 位被清除为“0”，BP_NV 位被清除为“000”。将 BPNV_O 设置为“0”时，可以选择状态寄存器中的 BP_NV 2-0 位来控制块保护，通过使用 WRR 命令可以写入该位。写入 BP_NV 位所需要的时间为 t_W 。将 BPNV 设置为“1”时，可以选择状态寄存器中的 BP2-0 位来控制块保护，而且 BP_NV 2-0 位将被编程为二进制“111”。这样，在 POR、硬件复位或命令复位后，BP 2-0 位将被设置为二进制 111。将 BPNV 设置为 1 时，WRR 命令只会写入 BP 位的易失性版本 (SR1V[4:2])。BP 位的非易失性版本 (SR1NV[4:2]) 不受 WRR 命令的影响。这样写入 BP 位的次数不受限制，因为它们是易失性的。写入易失性 BP 位的时间比 t_{CS} 易失性寄存器写时间更短。如果将 BPNV_O 编程为 1，将 0 写入到该位将不能修改该值或设置编程错误位 (SR1V[6] 中的 P_ERR)。

TBPARM_O CR1NV[2]: TBPARM_O 定义了参数模块的逻辑位置。该参数模块包含 8 个 4 KB 参数扇区，它们可以替代顶部或底部地址扇区中 32 KB 的空间。将 TBPARM_O 设置为“1”时，该参数模块位于存储器阵列地址空间的顶部。将 TBPARM_O 设置为“0”时，该参数模块位于该阵列的底部。TBPARM_O 位为 OTP，从赛普拉斯出厂时，它被设置为“0”。如果将 TBPARM_O 编程为 1，将 0 写入到该位将不能修改该值或设置编程错误位 (SR1V[6] 中的 P_ERR)。

在系统制造的器件初始配置期间，并在对主闪存阵列进行第一个编程或擦除操作前必须选择 TBPARM_O 的状态。对主闪存阵列进行编程或擦除操作后，不能编程 TBPARM_O。

设置或清除 TBPROT_O 位的操作完全独立于 TBPARM_O 位。因此，用户可以选择存储阵列底部的参数信息和保护开始于阵列顶部的启动代码，反之亦然。或者，用户可以选择存储和保护开始于顶部或底部的参数信息。

将存储器阵列配置为统一扇区时，TBPARM_O 位是保留给未来使用的 (RFU) 位，由于统一了所有扇区的大小，因此该位无效。

四线数据宽度非易失性 (QUAD_NV) CR1NV[1]: 为 CR1V[1] 中的 QUAD 位提供默认状态。WRR 或 WRAR 命令会影响该位的状态。通过将 CR2NV[6] 设置为 1 选择非易失性 QPI 模式也会将 QUAD_NV 编程为 1，从而将默认非易失性更改为四线数据宽度模式。通过将 CR2V[6] 设置为 1 来选择 QPI 模式时，不能将 Quad_NV 位清除为 0。

默认冻结保护 (FREEZE) CR1NV[0]: 为 CR1V[0] 中的冻结位提供了默认状态。用户不能编程该位。

9.6.3.2 易失性配置寄存器 1 (CR1V)

相关命令：读取配置寄存器 (RDCR 35h)、写入寄存器 (WRR 01h)、读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。通过使用 RDCR 命令可以显示该寄存器。

表 31. 易失性配置寄存器 1 (CR1V)

位	字段名称	功能	类型	默认状态	说明
7	RFU	预留供将来使用	易失性	CR1NV	预留
6	RFU				用户不可写入 请参见 CR1NV[5] TBPROT_O
5	TBPROT	TBPROT_O 的易失性副本	易失性 只读		预留供将来使用
4	RFU	RFU	RFU		用户不可写入 请参见 CR1NV[3] BPNV_O
3	BPNV	BPNV_O 的易失性副本	易失性 只读		用户不可写入 请参见 CR1NV[2] TBPARM_O
2	TBPARM	TBPARM_O 的易失性副本	易失性 只读		1 = 四线 0 = 双线或串行
1	QUAD	四线 I/O 模式	易失性		锁定块保护控制位的当前状态和 OTP 区域 1 = 块保护和 OTP 被锁定 0 = 块保护和 OTP 不被锁定
0	FREEZE	锁定块保护，直到下一个电源周期到来为止	易失性		

TBPROT、BPNV 和 TBPARM CR1V[5,3,2] 这些位是 CR1NV 的相关非易失性位的易失性副本。它们跟踪相关非易失性版本的更改。

四线数据宽度 (QUAD) CR1V[1]: 设置为 1 时，这些位将器件的数据宽度切换到 4 比特四线模式。当 CS# 为低电平时，WP# 变为 IO2，IO3_RESET# 变为有效的 I/O 信号。当 CS# 为高电平时，IO3_RESET# 变为 RESET# 输入。WP# 输入不被监控，因此它会正常运行，并且它被内部设置为高电平（无效）。可以正常执行串行和双线 I/O 读取命令，但在使用不同的数据路径宽度切换各个命令时不用为这些命令驱动 WP# 输入。同样，在执行这些命令期间，不需要驱动 IO3_RESET# (CS# 为低电平)。当使用四线读取 I/O、DDR 四线读取 I/O、QPI 模式 (CR2V[6] = 1) 和四线读取 ID 命令时，必须将 QUAD 位设置为 1。通过将 CR2V[6] 设置为 1 来选择 QPI 模式时，不能将 Quad 位清除为 0。WRR 命令写入 Quad 位的非易失性版本 (CR1NV[1]) 时，也会更新易失性版本 CR1V[1]。WRR 命令写入易失性版本 CR1V[1] 时会影响非易失性版本 CR1NV[1]。要想写入易失性 Quad 位 CR1V[1] 而不影响到非易失性版本 CR1NV[1]，则必须使用 WRAR 命令。

冻结保护 (FREEZE) CR1V[0]: 将冻结位设置为 1 时，将锁定块保护控制位和 OTP 区域的当前状态：

- 非易失性状态寄存器 1 中的 BPNV_2-0 位 (SR1NV[4:2])
 - 易失性状态寄存器 1 中的 BP 2-0 位 (SR1V[4:2])
 - 非易失性配置寄存器中的 TBPROT_O、TBPARM_O 和 BPNV_O 位 (CR1NV[53, 2])
 - 易失性配置寄存器中的 TBPROT、TBPARM 和 BPNV 位 (CR1V[5, 3, 2]) 被间接保护，这是因为它们是相关 CR1NV OTP 位的映射，并且为只读位
 - 整个 OTP 存储器空间
- FREEZE = 1 时尝试更改上述位的操作均被阻止：
- WRR 命令不会影响所列位，并且无错误状态被设置。
 - WRAR 命令没有影响到所列位并且无错误状态被设置。
 - OTP 区域中带有地址的 OTPP 命令发生故障并设置 P-ERR 状态。

一旦冻结位保持擦除为逻辑 0，将可以写入块保护控制位和冻结位，并且可以编程 OTP 地址空间。

将一个逻辑 1 写入到冻结位后，只能通过掉电到上电周期或硬件复位将其清除为逻辑 0。软件复位将不影响到冻结位的状态。

CR1V[0] FREEZE 位为易失性位，上电后，冻结位的默认状态由 CR1NV[0] 中的 FREEZE_D 决定。通过使用单个 WRR 或 WRAR 命令可以同时设置冻结位并更新 CR1V 中的其他值。

冻结位不会防止 WRR 或 WRAR 命令更改 SRWD_NV (SR1NV[7])、Quad_NV (CR1NV[1]) 或 QUAD (CR1V[1]) 位。

9.6.4 配置寄存器 2

配置寄存器 2 控制某些接口功能。通过使用读取任何寄存器和写入任何寄存器命令可以读取和更改这些寄存器位。该寄存器的非易失性版本允许为这些控制设置 POR、硬件复位或软件复位状态。这些配置位为 OTP，在系统配置期间将它们的默认状态更改为相反值的操作只能进行一次。正常操作期间，该寄存器的易失性版本控制功能的性能。

9.6.4.1 非易失性配置寄存器 2 (CR2NV)

相关命令：读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。

表 32. 配置寄存器 2 非易失性 (CR2NV)

位	字段名称	功能	类型	默认状态	说明
7	AL_NV	地址长度	OTP	0	1 = 4 字节地址 0 = 3 字节地址
6	QA_NV	QPI		0	1 = 使能 — 使用 QPI（4-4-4）协议 0 = 禁用 — 使用传统 SPI 协议，指令始终在 SI 线上串行传输
5	IO3R_NV	IO3 复位		0	1 = 使能 — 当 CS# 为高电平或四线模式被禁用（CR1V[1] = 1）时，IO3 作为 RESET# 输入使用 0 = 禁用 — IO3 没有其他功能，硬件复位功能被禁用。
4	RFU	预留		0	预留供将来使用
3	RL_NV	读延迟		1	0 到 15 个延迟（虚拟）周期（随后读地址或连续模式位）。 请注意，位 3 的默认值为 1。可以将该位编程为 0，但不能编程返回为 1。
2				0	
1				0	
0				0	

非易失性的地址长度 CR2NV[7]: 该位为所有命令的预期地址长度控制 POR、硬件复位或软件复位状态。这些命令需要地址而且这些地址不被固定为 3 字节或 4 字节 (32 位) 地址。多数传统的 SPI 命令需要一个地址，这些命令通常使用 3 字节 (24 位) 地址。如果器件容量超过 128 Mb，将需要一个 4 字节地址来访问整个存储器阵列。通过使用地址长度配置位可以将几乎所有 3 字节地址命令更改为所期待的 4 字节地址。请参考第 75 页上的表 51，了解命令地址长度。4 字节地址长度也适用于 FS-S 系列的 128 Mb 器件，因此可以将相同的 4 字节地址硬件和软件接口用于所有器件系列，以简化各种容量间的移植程序。FS-S 系列的 128 Mb 器件会忽略第四个高位地址字节中的内容。该非易失性地址长度配置位允许器件在 4 字节地址模式立即启动 (引导)，而不是在传统的 3 字节地址模式下进行。

QPI 非易失性 CR2NV[6]: 该位为所有命令的预期指令宽度控制 POR、硬件复位或软件复位状态。传统 SPI 命令始终在 SI (IO0) 信号上以 1 位宽度 (串行 I/O) 发送指令。FS-S 系列也支持 QPI 模式。在该模式下，主机系统和存储器之间的所有传输在 IO0 到 IO3 上实现，传输宽度为 4 比特，包括所有指令。该非易失性 QPI 配置位允许器件在 QPI 模式下立即启动 (引导)，而不是在传统串行指令模式下进行。将该位编程为 QPI 模式时，QUAD_NV 位也被编程为四线模式 (CR1NV[1]=1)。转到 QPI 模式的建议流程是先使用 WRAR 命令将 CR2V[6] 设置为 1，QPI 模式。QPI 模式的易失性寄存器写操作拥有 (t_{CS}) 时间，用于将器件接口切换到 QPI 模式。因此，能以 QPI 协议方式立即发送以下命令。通过使用 WRAR 命令将 CR2NV[6] 编程为 1，然后轮询 SR1V[0]，以确定编程操作是否完成。同样，要想退出 QPI 模式，需要使用 WRAR 命令将 CR2V[6] 清除为 0。不能将 CR2NV[6] 清除为 0，因为它是 OTP。

IO3 复位非易失性 CR2NV[5]: 该位控制 IO3 信号性能的 POR、硬件复位或软件复位状态。由于传统 SPI 器件封装中的信号数量和连接有限，因此几乎所有传统 SPI 器件都没有硬件复位输入信号。当没有将 IO3 信号用于传输主机系统和存储器之间的信息时，FS-S 系列允许将 IO3 信号作为硬件复位输入使用。通过使用该非易失性 IO3 复位配置位，可以立即启动 (引导) 器件，而且 IO3 作为 RESET# 信号使用。

读取延迟非易失性 CR2NV[3:0]: 该位控制着所有可变延迟读命令中读取延迟（虚拟周期）的 POR、硬件复位或软件复位状态。以下读命令在地址或模式的结束和读取数据开始返回到主机之间有一个可变的延迟周期：

- 快速读取
- 双线输出读取
- 四线输出读取
- 双线 I/O 读取
- 四线 I/O 读取
- DDR 四线 I/O 读取
- OTPR
- ECCRD
- RDAR

非易失性读延迟配置位设置需要使用的读延迟（虚拟周期）数量，因此可以使用主机系统的合适读延迟立即启动（引导）器件。

表 33. 延迟代码（周期）与频率

延迟代码	读取命令的最大频率（MHz）				
	快速读取（1-1-1）双线输出 （1-1-2） 四线输出（1-1-4） OTPR（1-1-1） ECCRD（1-1-1） RDAR（1-1-1） RDAR（4-4-4）	双线 I/O（1-2-2）	四线 I/O（1-4-4） QPI（4-4-4）	ECCRD（4-4-4）	DDR 四线 I/O（1-4-4） DDR QPI（4-4-4）
	模式周期 = 0	模式周期 = 4	模式周期 = 2	模式周期 = 0	模式周期 = 1
0	50	80	40	16	N/A
1	66	92	53	26	22
2	80	104	66	40	34
3	92	116	80	53	45
4	104	129	92	66	57
5	116	133	104	80	68
6	129	133	116	92	80
7	133	133	129	104	80
8	133	133	133	116	80
9	133	133	133	129	80
10	133	133	133	133	80
11	133	133	133	133	80
12	133	133	133	133	80
13	133	133	133	133	80
14	133	133	133	133	80
15	133	133	133	133	80

注意

44.SCK 频率 > 133 MHz SDR，或 80MHz DDR 不受该器件系列的支持。

45.双线 I/O、四线 I/O、QPI、DDR 四线 I/O 和 DDR QPI 命令协议包括随后地址的连续读取模式位。这些位的时钟周期不计入下表中的延迟周期。例如：传统的四线 I/O 命令中，地址后面是两个连续读取模式周期。因此，对于无其他读取延迟的传统四线 I/O 命令，受支持的频率如下表所示（读取延迟为 0）。通过递增可变读取延迟，可增大四线 I/O 命令的频率，从而允许执行受支持的最大 133 MHz 频率的操作。

46.其他读取命令拥有固定的延迟。例如，读取命令始终拥有 0 读取延迟，RSFDP 始终拥有 8 个延迟周期，RUID 在 QPI 中始终拥有四个虚拟字节或 16 个虚拟字节（32 个时钟周期）。

9.6.4.2 易失性配置寄存器 2 (CR2V)

相关命令：读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)、4BAM。

表 34. 配置寄存器 2 易失性 (CR2V)

位	字段名称	功能	类型	默认状态	说明
7	AL	地址长度	易失性	CR2NV	1 = 4 字节地址 0 = 3 字节地址
6	QA	QPI			1 = 使能 — 使用 QPI（4-4-4）协议 0 = 禁用 — 使用传统 SPI 协议，指令始终在 SI 线上串行传输
5	IO3R_S	IO3 复位			1 = 使能 — 当 CS# 为高电平或四线模式被禁用（CR1V[1] = 1）时，IO3 作为 RESET# 输入使用 0 = 禁用 — IO3 没有其他功能，硬件复位功能被禁用。
4	RFU	预留			预留供将来使用
3	RL	读延迟			0 到 15 个延迟（虚拟）周期（随后读地址或连续模式位）。
2					
1					
0					

地址长度 CR2V[7]: 该位控制所有命令的预期地址长度。这些命令需要地址, 这些地址不被固定为 3 字节或 4 字节 (32 位) 地址。请参考第 75 页上的表 51, 了解命令地址长度。通过易失性地址长度配置位, 用户可以在正常操作期间更改地址的长度。4 字节地址模式 (4BAM) 命令直接将该位设置为 4 字节地址模式。

QPI CR2V[6]: 该位控制所有命令的预期指令宽度。通过使用易失性 QPI 配置位, 在正常操作期间, 器件可以进入和退出 QPI 模式。将该位编程为 QPI 模式时, QUAD 位也被设置为四线模式 (CR1V[1] = 1)。将该位清除为传统的 SPI 模式时, QUAD 位不受影响。

IO3 复位 CR2V[5]: 该位控制 IO3_RESET# 信号的性能。易失性 IO3 复位配置位允许将 IO3 作为 RESET# 输入使用。

读取延迟 CR2V[3:0]: 该位控制可变延迟读命令中的读延迟 (虚拟周期)。通过使用这些易失性配置位, 用户可以在正常操作期间调整读取延迟, 从而按需要优化不同命令的延迟或不同频率的延迟。

9.6.5 配置寄存器 3

配置寄存器 3 控制某些命令的性能。通过使用读取任何寄存器和写入任何寄存器命令可以读取和更改这些寄存器位。该寄存器的非易失性版本允许为这些控制设置 POR、硬件复位或软件复位状态。这些配置位为 OTP, 在系统配置期间可能会将它们的默认状态一次更改为相反状态 (若需要)。通过使用配置寄存器 3 的易失性版本可以在系统操作或测试期间更改该配置。

9.6.5.1 非易失性配置寄存器 3 (CR3NV)

相关命令：读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。

表 35. 配置寄存器 3 非易失性 (CR3NV)

位	字段名称	功能	类型	默认状态	说明
7	RFU	预留	OTP	0	预留供将来使用
6	RFU	预留		0	预留供将来使用
5	BC_NV	空白检查		0	1 = 在擦除期间使能空白检查 0 = 禁用空白检查
4	02h_NV	页缓冲区回卷		0	1 = 512 字节缓冲区回卷 0 = 256 字节缓冲区回卷
3	20h_NV	4 KB 擦除		0	1 = 禁用 4 KB 擦除 (统一扇区架构) 0 = 使能 4 KB 擦除 (混合扇区架构)
2	30h_NV	清除状态 / 恢复选择		0	1 = 30h 为擦除或编程恢复命令 0 = 30h 为清除状态命令
1	D8h_NV	块擦除大小		0	1 = 256 KB 擦除 0 = 64 KB 擦除
0	F0h_NV	传统的软件复位使能		0	1 = F0h 为软件复位使能 0 = F0h 为软件复位禁用 (忽略)

非易失性空白检查 CR3NV[5]: 在执行擦除期间, 该位控制着空白检查的 POR、硬件复位或软件复位的状态。

02h 非易失性 CR3NV[4]: 该位控制着页编程缓冲区地址回卷点的 POR、硬件复位或软件复位状态。

20h 非易失性 CR3NV[3]: 该位用于控制主闪存阵列地址映射中 4 KB 参数扇区的可用性的 POR、硬件复位或软件复位状态。

30h 非易失性 CR3NV[2]: 该位用于控制所使用 30h 指令代码的 POR、硬件复位或软件复位状态。

D8h 非易失性 CR3NV[1]: 对于通过 FS-S 系列中的 D8h 或 DCh 指令实现擦除操作的区域尺寸配置, 需要使用该位来控制它的 POR、硬件复位或软件复位状态。

F0h 非易失性 CR3NV[0]: 对于普拉斯传统 FL-S 系列软件复位指令的可用性, 需要使用该位来控制它的 POR、硬件复位或软件复位状态。

9.6.5.2 易失性配置寄存器 3 (CR3V)

相关命令：读取任意寄存器 (RDAR 65h)、写入任意寄存器 (WRAR 71h)。

表 36. 易失性配置寄存器 3 (CR3V)

位	字段名称	功能	类型	默认状态	说明
7	RFU	预留	易失性	CR3NV	预留供将来使用
6	RFU	预留			预留供将来使用
5	BC_V	空白检查			1 = 在擦除期间使能空白检查 0 = 禁用空白检查
4	02h_V	页缓冲区回卷			1 = 512 字节缓冲区回卷 0 = 256 字节缓冲区回卷
3	20h_V	4 KB 擦除	易失性 只读		1 = 禁用 4 KB 擦除 （统一扇区架构） 0 = 使能 4 KB 擦除 （混合扇区架构）
2	30h_V	清除状态 / 恢复选择	易失性		1 = 30h 为擦除或编程恢复命令 0 = 30h 为清除状态命令
1	D8h_V	块擦除大小			1 = 256 KB 擦除 0 = 64 KB 擦除
0	F0h_V	传统的软件复位使能			1 = F0h 为软件复位使能 0 = F0h 为软件复位禁用 （忽略）

空白检查易失性 CR3V[5]: 该位用于控制在擦除期间的空白检查。启用该功能时，一个擦除命令将先对扇区的擦除状态求值。如果发现该扇区未成功完成最后一个擦除操作，将无条件擦除它。如果已经成功完成最后一个擦除操作，那么将通过读取扇区来指出它是否仍被擦除（空白）。检测到任意编程零后，将立即开始擦除操作。如果扇区仍然空白（没有检测到编程零位），将跳过擦除操作剩余部分。擦除中的扇区则不需要再进行擦除操作，这样可以明显缩短擦除时间。启用时，空白检查功能可用于参数擦除、扇区擦除和批量擦除命令。禁用该功能时，擦除命令会无条件启动擦除操作。

02h 易失性 CR3V[4]: 该位用于控制页编程缓冲器的地址翻转点。传统 SPI 器件一般使用了页大小为 256 字节的编程缓冲器，它会定义数据是否被加载到超出 255 字节位置的缓冲器内，并且额外加载的字节是否被翻转为缓冲器的地址零。FS-S 系列提供了页大小为 512 字节的编程缓冲器，从而能够改善编程性能。为了与传统软件相兼容，该配置位提供了一个选项，可选择在 256 字节边界上持续进行翻转行为，还是在 256 字节边界上禁止翻转加载地址，因而允许使用整个 512 字节的缓冲器。

20h 易失性 CR3V[3]: 该位用于控制主闪存阵列地址映射中 4 KB 参数扇区的可用性。参数扇区可以覆盖器件的最高或最低 32 KB 地址范围，或者可以从地址映射中删除这些参数扇区，使所有扇区的大小相同。该位的值必须与 CR3NV[3] 的相同。只有写入 CR3NV[3]，才能修改 CR3V[3] 的值。

30h 易失性 CR3V[2]: 该位用于控制 30h 指令代码的使用。该指令可用作清除状态命令或代替编程 / 擦除恢复命令使用。这样可使该器件和赛普拉斯的旧型 SPI 器件或其它供应商的器件在软件上相互兼容。

D8h 易失性 CR3V[1]: 该位用于控制 FS-S 系列中使用 D8h 或 DCh 指令删除的区域大小。通过使用该指令，可以擦除 64 KB 扇区或 256 KB 的对齐模块。低容量系列产品中的擦除 256 KB 模块的选项允许所有容量器件在软件上相互兼容，从而能够轻松在不同容量间移植。

F0h 易失性 CR3V[0]: 该位用于控制赛普拉斯旧型 FL-S 系列软件复位指令的可用性。FS-S 系列支持通过业界内通用的 66h + 99h 指令序列进行软件复位。通过该配置位，可以选择是否继续使用传统的 F0h 单命令进行软件复位。

9.6.6 配置寄存器 4

配置寄存器 4 控制着主闪存阵列读取命令的回卷突发性能。回卷突发配置不会影响读取非主闪存阵列的区域的命令，如：读取寄存器或 OTP 阵列的命令。由于在 POR、硬件复位或软件复位期间寄存器的内容被复制到寄存器的易失性版本内，因此该寄存器的非易失性版本允许设置该控制操作的启动（引导）状态。在正常操作期间，该寄存器的易失性版本控制着功能状态。通过使用读取任何寄存器和写入任何寄存器命令可以读取和更改这些寄存器位。通过使用设置突发长度（C0h）命令也可以写入该寄存器的易失性版本。

9.6.6.1 非易失性配置寄存器 4（CR4NV）

相关命令：读取任何寄存器（RDAR 65h）、写入任何寄存器（WRAR 71h）。

表 37. 非易失性配置寄存器 4（CR4NV）

位	字段名称	功能	类型	默认状态	说明
7	OI_O	输出阻抗	OTP	0	请参见第 60 页上的表 38. 输出阻抗控制
6				0	
5				0	
4	WE_O	回卷使能		1	0 = 回卷使能 1 = 回卷禁用
3	RFU	预留		0	预留供将来使用
2	RFU	预留		0	预留供将来使用
1	WL_O	回卷长度		0	00 = 8 字节回卷 01 = 16 字节回卷 10 = 32 字节回卷 11 = 64 字节回卷
0				0	

输出阻抗非易失性 CR4NV[7:5]: 这些位控制 IO 信号输出阻抗（驱动强度）的 POR、硬件复位或软件复位状态。通过多个驱动强度可以使输出阻抗与系统印刷电路板环境相互匹配，从而能够尽量降低过冲和振铃。这些非易失性输出阻抗配置位允许器件使用合适的驱动强度立即启动（引导）。

表 38. 输出阻抗控制

CR4NV[7:5] 阻抗选项	对 V _{SS} 的典型阻抗 (Ω)	对 V _{CC} 的典型阻抗 (Ω)	注释
000	47	45	工厂默认设置
001	124	105	
010	71	64	
011	47	45	
100	34	35	
101	26	28	
110	22	24	
111	18	21	

回卷使能非易失性 CR4NV[4]: 该位控制着回卷使能的 POR、硬件复位或软件复位状态。受回卷使能影响的命令包括：读取四路 I/O、读取 DDR 四路 I/O、读取四路输出和读取 QPI。该配置位允许器件在回卷突发（wrapped burst）读取模式下立即启动，而不是在传统的连续读取模式下进行。

回卷长度非易失性 CR4NV[1:0]: 这些位控制着回卷读取长度和对齐的 POR、硬件复位或软件复位状态。这些非易失性配置位允许器件在回卷突发读取模式下立即启动（引导），而不是在传统的连续读取模式下进行。

9.6.6.2 易失性配置寄存器 4（CR4V）

相关命令：读取任何寄存器（RDAR 65h）、写入任何寄存器（WRAR 71h）和设置突发长度（SBL C0h）。

表 39. 易失性配置寄存器 4（CR4V）

位	字段名称	功能	类型	默认状态	说明
7	OI	输出阻抗	易失性	CR4NV	请参见第 60 页上的表 38. 输出阻抗控制
6					
5					
4	WE	回卷使能			0 = 回卷使能 1 = 回卷禁用
3	RFU	预留			预留供将来使用
2	RFU	预留			预留供将来使用
1	WL	回卷长度			00 = 8 字节回卷 01 = 16 字节回卷 10 = 32 字节回卷 11 = 64 字节回卷
0					

输出阻抗 CR2V[7:5]: 这些位控制 IO 信号输出阻抗（驱动强度）。通过使用该易失性输出阻抗配置位，用户可以在正常操作期间调整驱动强度。

回卷使能 CR4V[4]: 这些位控制着突发回卷功能。通过使用该易失性配置位，在正常操作期间，器件可以进入和退出突发回卷模式。

回卷长度 CR4V[1:0]: 在正常操作期间，这些位控制着回卷读取长度和对齐。通过使用这些易失性配置位，用户可以在正常操作期间调整连续回卷读取长度。

9.6.7 ECC 状态寄存器 (ECCSR)

相关命令：读取 ECC (ECCRD 18h 或 19h)。ECCSR 没有用户可编程的非易失性。所有定义位都处于易失性只读状态。这些位的默认状态是由硬件设置的。请参见第 101 页上的自动纠错码 (ECC)。

每个 ECC 单元中的 ECC 状态均由 8 位 ECC 状态寄存器 (ECCSR) 提供。写入 ECC 寄存器读指令后，将写入一个 ECC 单元的地址。接下来状态寄存器的内容会表示：在选定的 ECC 单元中，错误出现在 ECC 单元八比特纠错码、ECC 单元的 16 比特数据还是该 ECC 单元的 ECC 已经被禁用。

表 40. ECC 状态寄存器 (ECCSR)

位数	字段名称	功能	类型	默认状态	说明
7 到 3	RFU	预留		0	预留供将来使用
2	EECC	ECC 中的错误	易失性，只读	0	1 = ECC 单元八比特纠错码中发现单比特错误 0 = 无错误
1	EECCD	ECC 单元数据中的错误	易失性，只读	0	1 = 在 ECC 单元数据中纠正了单比特错误。 0 = 无错误
0	ECCDI	ECC 被禁用	易失性，只读	0	1 = 在已选择的 ECC 单元中 ECC 被禁用。0 = 在已选择的 ECC 单元中 ECC 被使能。

ECCSR[2] = 1 表示纠正了 ECC 中的错误。ECCSR[1] = 1 表示纠正了 ECC 单元数据中的错误。ECCSR[0] = 1 表示 ECC 被禁用。所有位的默认状态“0”表示不存在任何错误，并且 ECC 功能是打开的。

ECCSR[7:3] 被预留。这些位的值不确定（高或低），在每次读取 ECC 状态时，该值将发生变化。软件读取状态时，这些位将被忽略并被视为“无需关注”。

9.6.8 ASP 寄存器 (ASPR)

相关指令：ASP 读取 (ASPRD 2Bh) 和 ASP 编程 (ASPP 2Fh)、读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。ASP 寄存器指的是一个 16 位的 OTP 存储位置，用于永久地配置高级扇区保护 (ASP) 特性的状态。ASPR 中没有用户可编程的易失性位，它的所有定义位都是 OTP。

各 ASPR 位的默认状态是由赛普拉斯编程的。

表 41. ASP 寄存器 (ASPR)

位	字段名称	功能	类型	默认状态	说明
15 到 9	RFU	预留	OTP	1	预留供将来使用
8	RFU	预留	OTP	1	预留供将来使用
7	RFU	预留	OTP	1	预留供将来使用
6	RFU	预留	OTP	1	预留供将来使用
5	RFU	预留	OTP	1	预留供将来使用
4		预留	RFU	1	预留供将来使用
3		预留	RFU	1	预留供将来使用
2	PWDMLB	密码保护模式锁定位	OTP	1	0 = 密码保护模式被永久使能。 1 = 密码保护模式没有被永久使能。
1	PSTMLB	持久保护模式锁定位	OTP	1	0 = 持久保护模式被永久使能。 1 = 持久保护模式没有被永久使能。
0		预留	RFU	1	预留供将来使用

密码保护模式锁定位 (PWDMLB) ASPR[2]: 被设置为“0”时, 将永久地选中密码保护模式。

永久性保护模式锁定位 (PSTMLB) ASPR[1]: 被设置为“0”时, 将永久地选中永久性保护模式。

PWDMLB (ASPR[2]) 和 PSTMLB (ASPR[1]) 是相互排除的, 其中只能有一位被设置为“0”。

只有 ASPR[2:1] = 11b 时, 才能编程 ASPR 位。ASPR[2:1] 值不是 11b 时, 尝试编程 ASPR 位将导致编程错误, 使 P_ERR (SR1V[6]) 被设置为 1。通过将 ASPR[2:1] 编程为 10b 或 01b, 选中 ASP 保护模式后, 所有 ASPR 位的状态被锁定, 并且永久性禁止对其进行编程。尝试将 ASPR[2:1] 位编程为 00b 会引起编程错误, 并且 P_ERR (SR1V[6]) 的值被设置为 1。

同样, 只有 ASPR[2:1] = 11b 时, 才能编程 ASP 寄存器描述部分 (请参见第 68 页上的 ASP 寄存器) 所列出的 OTP 配置位。在选择 ASP 保护模式前必须先选择 OTP 配置。选中 ASP 模式时, 将永久防止对 OTP 配置位进行编程操作。当 ASPR[2:1] 为 11b 时, 尝试编程这些 OTP 配置位会引起编程错误, 并且 P_ERR (SR1V[6]) 位被设为 1。

在系统配置期间, 应一直选中 ASP 保护模式, 以确保恶意程序不会选中您不想要的保护模式。在通过 ASP 模式选择来锁定所有保护配置后, 恶意程序在后续时间内将不能对保护模式进行任意更改。

9.6.9 密码寄存器 (PASS)

相关指令: 密码读取 (PASSRD E7h) 和密码编程 (PASSP E8h)、读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。PASS (密码) 寄存器指的是 64 位的 OTP 存储位置, 该寄存器用于永久性定义高级扇区保护 (ASP) 功能的密码。PASS 中没有用户可编程的易失性位, 它的所有定义位都是 OTP。可以使用 PASS 的一个易失性副本来满足易失性寄存器的读取延迟要求, 但该易失性寄存器并不是用户可写的空间, 或没有详细的说明。

表 42. 密码寄存器 (PASS)

位	字段名称	功能	类型	默认状态	说明
63: 0	PWD	隐藏密码	OTP	FFFFFFFF-FFFFFFFFh	64 位密码的非易失性 OTP 存储。通过将 ASP 寄存器位 2 设置为 0 选择密码保护模式后, 密码将不再可读。

9.6.10 PPB 锁定寄存器 (PPBL)

相关命令: PPB 锁定读取 (PLBRD A7h、PLBWR A6h)、读取任何寄存器 (RDAR 65h)。

PPBL 中没有用户可编程的非易失性位, 它的所有定义位都是易失性只读状态位。RFU 位的默认状态是由硬件设置的。PPBLOCK 位的默认状态由 ASPR[2:1] 中的 ASP 保护模式位定义。PPBL 寄存器没有非易失性版本。

PPBLOCK 位用于保护 PPB 位。PPBL[0] = 0 时, 不能编程 PPB 位。

表 43. PPB 锁定寄存器 (PPBL)

位	字段名称	功能	类型	默认状态	说明
7 到 1	RFU	预留	易失性	00h	预留供将来使用
0	PPBLOCK	保护 PPB 阵列	易失性 只读	ASPR[2:1] = 1xb = 持久保护模式 = 1 ASPR[2:1] = 01b = 密码保护模式 = 0	0 = PPB 阵列受保护 1 = PPB 阵列被编程或擦除。

9.6.11 PPB 访问寄存器 (PPBAR)

相关命令：PPB 读取 (PPBRD FCh 或 4PPBRD E2h)、PPB 编程 (PPBP FDh 或 4PPBP E3h)、PPB 擦除 (PPBE E4h)。

PPBAR 并没有用户可写的易失性位，所有 PPB 阵列位都是非易失性的。PPB 阵列的默认状态由赛普拉斯擦除为 FFh。PPBAR 寄存器没有易失性版本。

表 44. PPB 访问寄存器 (PPBAR)

位	字段名称	功能	类型	默认状态	说明
7 到 0	PPB	读取或编程每个扇区 PPB	非易失性	FFh	00h = PPBRD 或 PPBP 命令所寻址的扇区的 PPB 被设置为 0，这样便不能对该扇区进行编程或擦除操作。 FFh = PPBRD 命令所寻址的扇区的 PPB 被设置为 1，这样能够对扇区进行编程或擦除操作。

9.6.12 DYB 访问寄存器 (DYBAR)

相关命令：DYB 读取 (DYBRD FAh 或 4DYBRD E0h)，DYB 写入 (DYBWR FBh 或 4DYBWR E1h)。

DYBAR 中没有用户可编程的非易失性位，它的每一位都代表 DYB 阵列中的易失性位。DYB 阵列的默认状态是由硬件设置的。DYBAR 寄存器没有非易失性版本。

表 45. DYB 访问寄存器 (DYBAR)

位	字段名称	功能	类型	默认状态	说明
7 到 0	DYB	读取或写入每个扇区 DYB	易失性		00h = DYBRD 或 DYBWR 命令所寻址的扇区的 DYB 被清除为 “0”，这样便不能对该扇区进行编程或擦除操作。 FFh = DYBRD 或 DYBWR 命令所寻址的扇区的 DYB 被设置为 “1”，这样能够对扇区进行编程或擦除操作。

9.6.13 SPI DDR 数据学习寄存器

相关命令：编程 NVDLR (PNVDLR 43h)、写入 VDLR (WVDLR 4Ah)、数据学习模式读取 (DLPRD 41h)、读取任何寄存器 (RDAR 65h)、写入任何寄存器 (WRAR 71h)。

数据学习模式 (DLP) 位于一个 8 位非易失性数据学习寄存器 (NVDLR) 内，它也被称为 8 位易失性数据学习寄存 (VDLR)。从赛普拉斯出厂时，NVDLR 的默认值为 00h。一旦对 NVDLR 进行了编程，便不能重新编程或擦除它；NVDLR 中数据模式的副本也会被写入到 VDLR 内。可以随时向 VDLR 进行写操作，但如果正在进行上电周期时，数据模式将返回到它在 NVDLR 中的状态。在执行 SPI DDR 模式中所描述的学习阶段内，DLP 将来自 VDLR。在每个时钟沿上，各个 IO 均输出相同的 DLP 值。例如，如果 DLP 为 34h (或二进制 00110100)，那么在第一个时钟沿上，所有 IO 都输出数值 “0”；随后，在第二个时钟沿上，所有 IO 输出数值 “0”，在第三个时钟沿上输出数值 “1”，等等。

VDLR 的值为 “00h” 时，DDR 命令中的虚拟阶段不会显示前导码数据模式。

表 46. 非易失性数据学习寄存器 (NVDLR)

位	字段名称	功能	类型	默认状态	说明
7:0	NVDLP	非易失性数据学习模式	OTP	00h	在 DDR 读取命令延迟 (虚拟) 周期中传输到主机的 OTP 值，用以为主机提供一个培训模式，来准确地确定所收数据位中的中心点。

表 47. 易失性数据学习寄存器 (VDLR)

位	字段名称	功能	类型	默认状态	说明
7:0	VDLP	易失性数据学习模式	易失性	在 POR 或复位期间提取 NVDLR 的值	NVDLP 的易失性副本，用于使能并将数据学习模式 (DLP) 传送到输出端。系统操作期间，主机可以修改 VDLP。

10. 数据保护

10.1 安全硅区域（OTP）

该器件具有一个独立于主闪存阵列的 1024 字节的一次性编程（OTP）地址空间。OTP 区域分为 32 个可独立锁定并且长度为 32 字节的对齐区域。

OTP 存储器空间用于增强系统安全性。可以使用 OTP 值将闪存组件与系统 CPU/ASIC “配对”起来，用于防止设备替换。请参见第 47 页上的 [OTP 地址空间](#)、第 111 页上的 [OTP 编程（OTPP 42h）](#) 以及第 111 页上的 [OTP 读取（OTPR 4Bh）](#)。

10.1.1 读取 OTP 存储空间

OTP 读取命令与快速读取使用了相同的协议。超过有效范围 1KB 的 OTP 地址的 OTP 读取操作会引起不正确的数据。

10.1.2 编程 OTP 存储空间

OTP 编程命令使用与页面编程命令相同的协议。可多次向任意已给 OTP 地址发送 OTP 编程命令，但无法擦除该地址空间。

自动 ECC 在第一次编程操作时被编程到每个 16 字节区域。在 16 字节区域内多次编程会禁用 ECC。建议对每个 32 字节区域的每个 16 字节部分进行一次编程，以使 ECC 保持使能状态，从而提供最佳的数据完整性。

第 47 页上的图 42 中显示的是 OTP 编程命令的有效地址范围。超过 OTP 地址有效范围的 OTP 编程操作将被忽略，并且 SR1V 中的 P_ERR 位不被设置为“1”。FREEZE = 1 时，如果 OTP 编程操作超出有效的 OTP 地址范围，该操作将失败，并且 SR1V 中的 P_ERR 被设置为“1”。选用 ASP 保护模式时，OTP 地址空间不受保护。冻结位（CR1V[0]）可用于保护 OTP 地址空间。

10.1.3 赛普拉斯编程随机数值

赛普拉斯的标准做法是使用线性同余随机数方法将 OTP 存储空间的低 16 字节（从 0x0 到 0xF 地址）编程为 128 位的随机数值。该算法的种子值是随机数与测试日期和时间连接起来的值。

10.1.4 锁定字节

每个锁定字节的最低有效位（LSB）和最高有效位（MSB）分别保护着有关该字节的最低和最高地址区域。同样，下一个更高的地址字节将保护后面更高的 8 个区域。最低地址锁定字节的 LSB 位将保护最低地址区域的更高 16 个地址字节。换句话说，0x10 位置的 LSB 将保护最低地址区域中所有锁定字节和 RFU 字节，防止对它们进行编程操作。请参见第 47 页上的 [OTP 地址空间](#)。

10.2 写入使能命令

执行用于修改非易失性数据的任意命令前，先要写好写使能（WREN）命令。WREN 命令用于设置写使能锁存（WEL）位。执行上电、硬件复位期间或在完成下面命令后，WEL 位将被清零（禁用写操作）：

复位

页面编程（PP 或 4PP）

参数 4KB 擦除（P4E 或 4P4E）

扇区擦除（SE 或 4SE）

批量擦除（BE）

写入禁用（WRDI）

写入寄存器（WRR）

写入任何寄存器（WRAR）

OTP 字节编程（OTPP）

高级扇区保护寄存器编程（ASPP）

永久保护位编程（PPBP）

永久保护位擦除（PPBE）

密码编程（PASSP）

编程非易失性数据学习寄存器（PNVDLR）

10.3 块保护

可以结合使用块保护位（状态寄存器中的位 BP2、BP1 和 BP0）和配置寄存器 TBPROT_O 位来禁止对主闪存阵列中某个地址范围进行所有编程和擦除操作。该范围大小由 BP 位的值指定，并且该范围的上限或下限起始点是由配置寄存器的 TBPROT_O 位（CR1NV[5]）决定。

表 48. S25FS064S 的保护部分的上层阵列起点（TBPROT_O = 0）

状态寄存器内容			存储器阵列的受保护部分	受保护的存储器（KB）
BP2	BP1	BP0		
0	0	0	否	0
0	0	1	上层第 64 个扇区	128
0	1	0	上层第 32 个扇区	256
0	1	1	上层第 16 个扇区	512
1	0	0	上层第 8 个扇区	1024
1	0	1	上层第 4 个扇区	2048
1	1	0	上半部分	4096
1	1	1	所有扇区	8192

表 49. S25FS064S 的保护部分的下层阵列起点（TBPROT_O = 1）

状态寄存器内容			存储器阵列的受保护部分	受保护的存储器（KB）
BP2	BP1	BP0		
0	0	0	否	0
0	0	1	下层第 64 个扇区	128
0	1	0	下层第 32 个扇区	256
0	1	1	下层第 16 个扇区	512
1	0	0	下层第 8 个扇区	1024
1	0	1	下层第 4 个扇区	2048
1	1	0	下半部分	4096
1	1	1	所有扇区	8192

使能块保护功能时（即任意 BP2-0 被设为“1”），通过使用高级扇区保护（ASP），保护扇区可不受块保护机制的保护。如果在同一个扇区上同时采用了 ASP 和块保护功能，那么会对有关该扇区的 ASP 和块保护进行逻辑 OR 运算。

10.3.1 冻结位

配置寄存器 1 的位 0（CR1V[0]）是冻结位。被设置为 1 时，冻结位会锁定块保护控制位和 OTP 区域的当前状态，直到下一次电源开关周期。更多的详细信息，请参考第 54 页上的易失性配置寄存器 1（CR1V）。

10.3.2 写入保护信号

写入保护（WP#）输入和状态寄存器中的写禁用（SRWD）位（SR1NV[7]）一起提供由硬件输入信号控制的保护。当 WP# 为低电平和 SRWD 的值为“1”时，状态寄存器 1（SR1NV 和 SR1V）和配置寄存器 1（CR1NV 和 CR1V）的状态不会发生改变。这样会防止禁用或更改由块保护位指定的保护。请参见第 49 页上的状态寄存器 1。

10.4 高级扇区保护

高级扇区保护（ASP）是一组独立的硬件和软件方法，分别用于禁止或允许对任何或所有扇区执行编程或擦除操作。

每个主闪存阵列扇区都有与其关联的一个非易失性永久性保护位（PPB）和一个易失性动态保护位（DYB）。无论哪一位是 0 时，扇区都受到保护，不能执行编程和擦除操作。当易失性 PPB 锁定位是 0 时，PPB 位受到保护，不能进行编程和擦除。PPB 锁定位的状态可通过两种方法进行保护：密码保护和永久性保护。第 67 页上的图 44 中显示的是这些方法的概述内容。

需要对块保护和 ASP 保护设置进行逻辑或（OR）运算，以便定义每个扇区的保护情况。这便意味着，无论使用哪个扇区保护机制，都无法对扇区进行编程或擦除操作。请参见第 65 页上的块保护，了解 BP2-0 位的详细信息。

图 43. 扇区保护控制

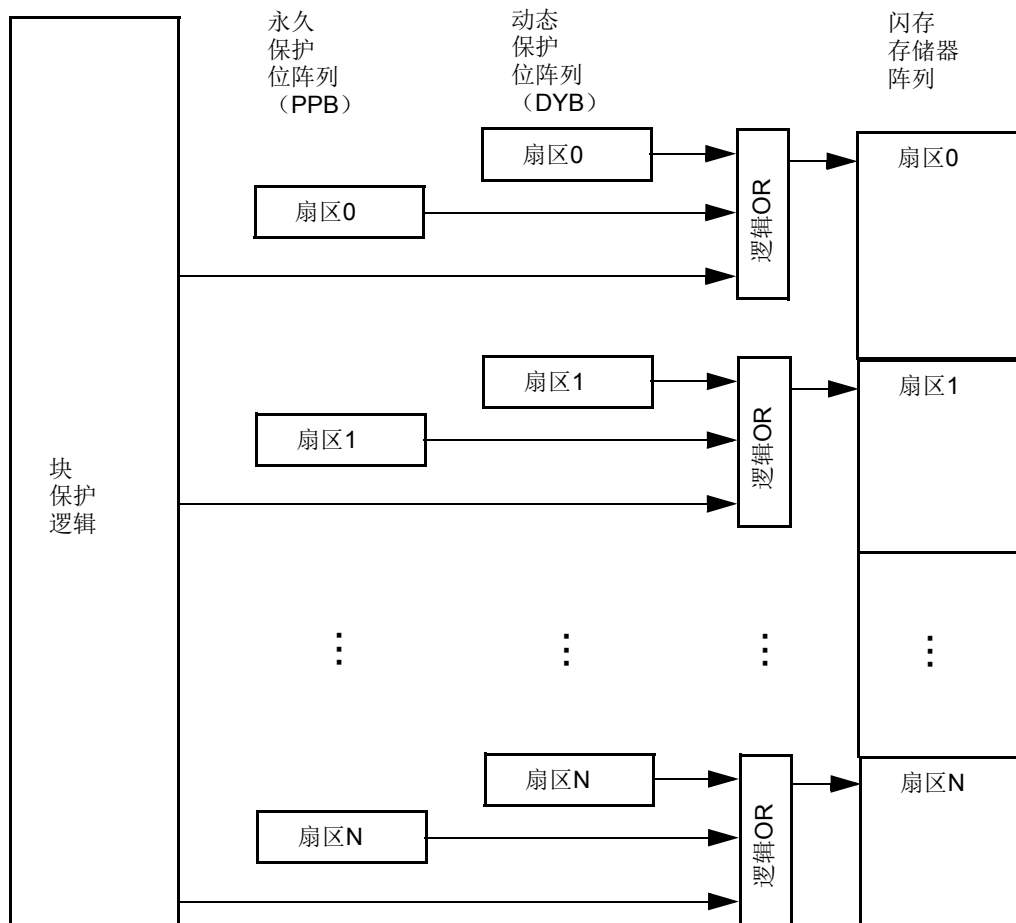
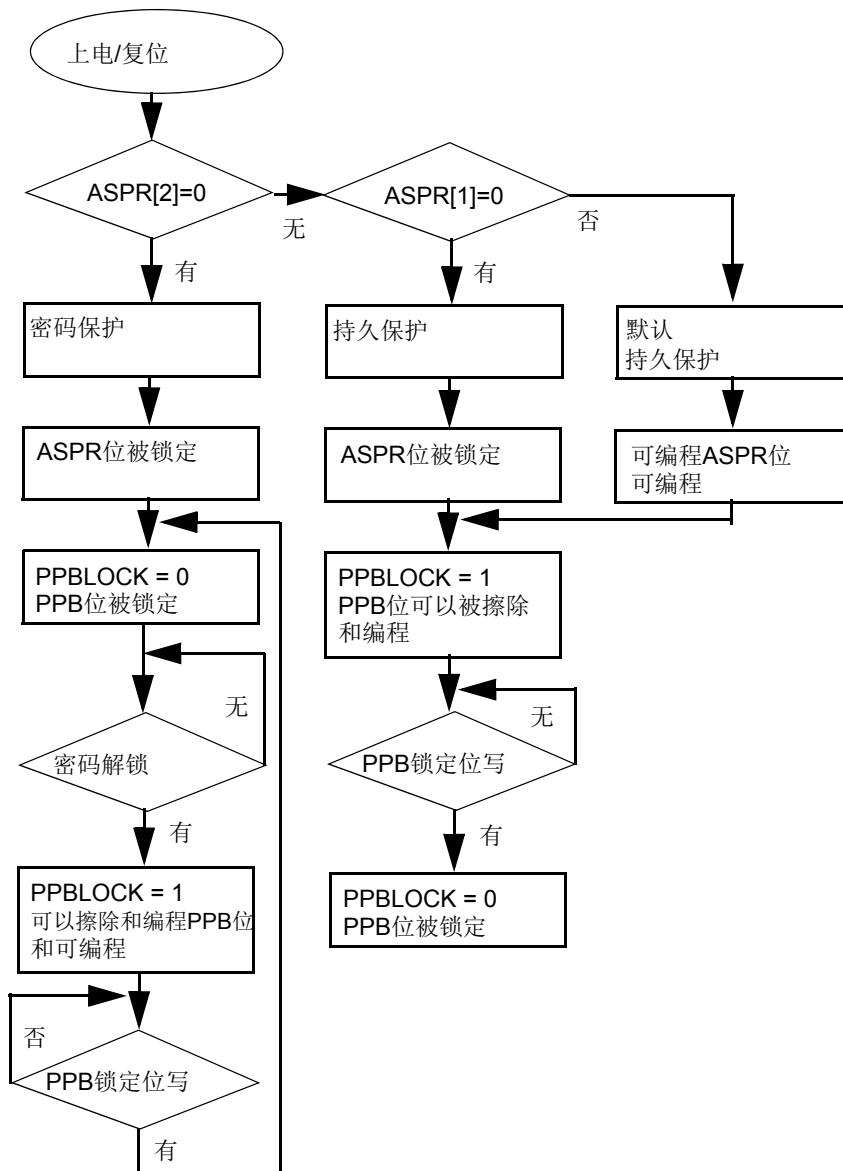


图 44. 高级扇区保护概述



密码保护模式上电后保护 PPB。通过密码解锁命令可对 PPB 进行更改。通过 PPB 锁定位写命令可以重新使能保护功能。

持久保护模式在上电后不保护 PPB。可以更改 PPB 位。通过 PPB 锁定位写命令可以保护 PPB 位，直到下一个断电或复位为止。

默认模式对 ASPR 进行编程，以永久选择保护模式。

否则，默认模式将作为持久保护模式。

选择保护模式后，不能对 ASPR 进行编程，因此已选择的保护模式为永久性的模式。

永久性保护方法在 POR 或硬件复位期间将 PPB 锁定位设为 1，因此 PPB 位不会受到器件复位的保护。可以通过一个指令来将 PPB 锁定位清除为 0，以保护 PPB。永久性保护方法没有任何指令可将 PPB 锁定位设为 1，因此 PPB 锁定位一直为 0，直到下一次断电或硬件复位为止。永久性保护方法允许引导代码通过编程或擦除 PPB 来更改扇区保护，然后通过清零 PPB 锁定位在正常系统操作的剩余时间内保护 PPB，不令其更改。这有时称为引导代码控制的扇区保护。

密码方法在 POR 或硬件复位期间将 PPB 锁定位清除为 0，以保护 PPB。采用密码方法时，64 位的密码是被永久性的编程的，并且编译密码方法是隐藏的。通过一个指令来提供一个密码，将其与隐藏密码进行比较。如果密码匹配，则 PPB 锁定位设为 1，以取消 PPB 保护。PPB 锁定位可以通过一个指令来清除为 0。该方法需要使用一个密码来控制 PPB 保护。

PPB 锁定管理方法通过编程 ASP 寄存器中的 OTP 位来选择，以便永久性选择使用的方法。

10.4.1 ASP 寄存器

ASP 寄存器用于永久地配置高级扇区保护（ASP）特性的状态。请参见第 61 页上的表 41。

从工厂出厂时，所有器件在上电时均默认使用 ASP 永久保护方法，所有扇区均不受保护。器件编程器或主机系统随后必须选择使用哪种扇区保护方法。通过编程一次性可编程或保护模式锁定位，持久将器件锁定为选定模式：

- ASPR[2:1] = “11”：未选择 ASP 模式，默认使用永久性保护模式。
- ASPR[2:1] = “10”：持久选择永久性保护模式。
- ASPR[2:1] = “01”：持久选择密码保护模式。
- ASPR[2:1] = “00”：非法状态，尝试将多个位编程为 0，会引起编程故障。

ASP 寄存器的编程规则：

- 如果选中了密码模式，则必须在设置保护模式的锁定位前编写密码。
- 一旦选中了保护模式，将永久保护下面的 OTP 配置寄存器位，并禁止对 OTP 寄存器位进行任何更改：
 - CR1NV
 - CR2NV
 - CR3NV
 - CR4NV
 - ASPR
 - PASS
 - NVDLR
- 选择 ASP 模式后，如果尝试修改上述任何寄存器位，该操作会失败，并且 P_ERR（SR1V[6]）被设置为 1。

ASP 寄存器的编程时间与典型的页面编程时间相同。系统可以通过读取状态寄存器中的 WIP 位来指出 ASP 寄存器的状态。请参考第 49 页上的状态寄存器 1，了解 WIP 的详细信息。请参见第 69 页上的扇区保护状态汇总。

10.4.2 永久保护位

永久保护位 (PPB) 位于一个单独的非易失性闪存阵列中。每个 PPB 位相应于一个扇区。当一个 PPB 位为 0 时，相应的扇区受到保护，不能对它执行编程和擦除操作。PPB 位可单独编程，但必须按组进行擦除。这与字相似，各个字可以在主阵列中单独编程，但整个扇区必须同时擦除。PPB 的编程和擦除时间与主闪存存储阵列的相同。擦除前的预编程和验证由器件执行。

编程一个 PPB 位需要典型的页编程时间。擦除所有 PPB 需要典型的扇区擦除时间。在编程和擦除 PPB 位期间，可通过读取状态寄存器来指出其状态。读取 PPB 位需要器件的初始访问时间。

注意：

1. 每个 PPB 可单独编程为 0，但必须同时擦除为 1。
2. PPB 锁定位为 0 时，PPB 编程或擦除指令不被执行。
3. 通过使用 PPB 读取指令可以验证已给扇区的 PPB 状态。

10.4.3 动态保护位

动态保护位是易失性位，可以单独更改该位。每个扇区只有一个唯一的动态保护位。DYB 只控制那些已将 PPB 设置为 1 的扇区的保护。通过发出 DYB 写入指令，DYB 可清除为 0 或设置为 1，从而相应地将每个扇区置于受保护或不受保护状态。该功能允许软件方便地保护扇区，以免意外更改；但需要更改时可以方便地去除保护。由于 DYB 位都是易失性位，可以按要求对它们进行设置或清除。

10.4.4 PPB 锁定位 (PPBL[0])

PPB 锁定位是一个易失性位，用于保护所有 PPB 位。清除为 0 时，它锁定所有 PPB；设成 1 时，便可以更改 PPB。相关的详细信息，请参见第 62 页上的 [PPB 锁定寄存器 \(PPBL\)](#)。

PPB 锁定位可以通过 PLBWR 指令来清除为 0。仅在所有 PPB 均配置为所需的设置后，才将 PPB 锁定位清零。

在永久保护模式下，PPB 锁定在 POR 或硬件复位期间被设为 1。清除为 0 后，无法通过软件指令序列设置 PPB 锁定为 1，只有通过硬件复位或上电才能设置 PPB 锁定位。

在密码保护模式下，PPB 锁定位在 POR 或硬件复位期间被清除为 0。PPB 锁定位只能通过密码解锁命令被设为“1”。

10.4.5 扇区保护状态汇总

每个扇区均可以处于下面一种保护状态：

- 解锁：扇区不受保护，保护状态可通过一个简单的指令进行更改。从赛普拉斯出厂时，器件的默认保护状态既为解锁状态。
- 动态锁定：扇区受保护，保护状态可通过一个简单的指令进行更改。在开关电源或进行复位后，保护状态不被保存。
- 永久锁定：扇区受保护，仅当 PPB 锁定位为 1 时，方可更改保护状态。保护状态是非易失性的，在开关电源或复位后不被保存。更改保护状态需要编程或擦除 PPB 位。

表 50. 扇区保护状态

保护位值			扇区状态
PPB 锁定	PPB	DYB	
1	1	1	无保护 — 可更改 PPB 和 DYB
1	1	0	受保护 — 可更改 PPB 和 DYB
1	0	1	受保护 — 可更改 PPB 和 DYB
1	0	0	受保护 — 可更改 PPB 和 DYB
0	1	1	受保护 — 不可更改 PPB, 可更改 DYB
0	1	0	受保护 — 不可更改 PPB, 可更改 DYB
0	0	1	受保护 — 不可更改 PPB, 可更改 DYB
0	0	0	受保护 — 不可更改 PPB, 可更改 DYB

10.4.6 永久保护模式

永久性保护方法在 POR 或硬件复位期间将 PPB 锁定位设为 1, 因此 PPB 位不会受到器件硬件复位的保护。软件复位不会对 PPB 锁定位产生影响。可以通过 PLBWR 指令来将 PPB 锁定位清除为 0, 以保护 PPB。无法通过指令可设置 PPB 锁定位, 因此 PPB 锁定位一直为 0, 直到下一次关闭电源或硬件复位为止。

10.4.7 密码保护模式

密码保护模式需要 64 位密码来对 PPB 锁定位进行解锁, 因此能够提供比永久扇区保护模式级别更高的安全性。除了密码要求外, 在上电和硬件复位后, PPB 锁定位还清除为 0 以确保在上电时提供保护。通过输入整个密码并成功执行密码解锁指令后, PPB 锁定将设为 1, 从而允许修改扇区 PPB。

密码保护注意事项:

- 编程和验证密码后, 必须设置密码模式 (ASPR[2] = 0), 以防止读取密码。
- 密码编程命令只能编程 “0”。若将一个单元编程为 0 后再编程为 1, 该单元将保留 0, 并且不会设置编程错误。
- 从赛普拉斯出厂时, 密码全部是 1。它位于自己的存储空间中, 可通过使用密码编程、密码读取、RDAR 和 WRAR 命令进行访问。选中密码锁定模式后, 即使使用这些命令也不能进行访问。
- 所有 64 位密码组合均为有效密码。
- 一旦编程了密码保护模式, 便能够防止对 64 位密码进行读取和编程操作。对密码区域的所有额外编程和读取命令均被禁止, 这些命令被忽略或返回未定义数据。选择了密码模式锁定位后, 无法验证密码内容。只能在选择密码保护模式前进行密码验证。
- 保护模式锁定位不可擦除。
- 只有输入准确的密码, 才能解锁。如果密码解锁命令提供的密码与隐藏的内部密码不匹配, 解锁操作会失败, 与对受保护的扇区执行编程操作相似。P_ERR 位被设为 0, WIP 位保持设置状态, 并且 PPB 锁定位保持清零状态。
- 每经过 $100\ \mu\text{s} \pm 20\ \mu\text{s}$ 时间, 只能接收一条密码解锁命令。这样, 黑客要想尝试所有 64 位组合以找到正确匹配的密码, 就要经过不可思议的超长时间 (5800 万年)。读取状态寄存器 1 命令可用于读取 WIP 位, 从而可以指出器件完成了密码解锁命令还是准备好接收新的密码命令。提供有效的密码时, 密码解锁命令会立即将 WIP 位清零, 不需要经过 $100\ \mu\text{s}$ 的延迟时间。
- 如果在选择密码模式后丢失了密码, 则无法将 PPB 锁定位设为 1。
- 只能从可读的扇区内读取 ECC 状态。在读保护模式下, 各地址均被强制转移到启动扇区地址。ECC 状态表示读保护模式时的扇区是被激活的。

10.5 建议的保护流程

在系统制造中，闪存器件的配置如下：

1. 根据需要对 CR1NV[5, 3:2]、CR2NV、CR3NV 和 CR4NV 中的 OTP 配置位进行编程。
2. 根据需要对安全硅区域（OTP 区域）进行编程。
3. 根据需要通过 PPBP 命令对 PPB 位进行编程。
4. 如果在 DDR 读取命令中需要使用非易失性数据学习模式（NVDLR），则需要编程它。
5. 若使用密码保护，则需要编程密码寄存器（PASS）。
6. 根据需要编程 ASP 寄存器，包括通过 ASPR[2:1] 选择永久性保护或密码 ASP 保护模式。明确指定一种保护模式非常重要，这样可以防止以后对 ASP 寄存器和 OTP 配置的意外或特意编程。这样可确保只有特定的 OTP 保护和配置功能被使能。

在系统上电和引导代码执行期间：

1. 通过可靠引导代码能够指出是否需要编程附加的 SSR（OTP 区域）信息。如果无需修改 SSR，可以将冻结位（CR1V[0]）的值设置为 1，以便在通电时防止在正常系统操作期间对 SSR 进行修改。
2. 如果使用永久性保护模式，则通过可靠引导代码能够指出是否需要通过 PPBP 或 PPBE 命令修改永久性（PPB）扇区保护。如果无需修改 PPB，则可以使用 PPBL 命令将 PPBLOCK 位清零，以便在通电时防止在正常系统操作期间对 PPB 位进行修改。
3. 可通过 DYBAR 命令根据需要编写动态（DYB）扇区保护位。

11. 命令

主机系统和 FS-S 系列存储器件系列之间都是以命令为单位形式进行通信。

所有命令都以一个 8 位指令开始，该指令用于选择信息传输类型和将被执行的器件操作。命令可能也具有一个地址、指令修饰符、延迟周期、传送到存储器，或者是从存储器传送的数据。主机系统和存储器件之间的所有指令、地址和数据信息都是连续传输的。

通过一个使用 3 个数字的数字命令法来划分命令协议，以便能参考三个命令阶段的发送宽度。

- 指令；
- 地址和指令修饰符（模式）；
- 数据。

单比特宽的命令只通过 SI/IO0 信号依次传输指令和地址或数据。数据会通过 SO/IO1 信号从存储器串行回送到主机内。单比特宽指令、单比特宽地址和修饰符、单比特数据的结构，被称为 1-1-1 命令协议。

双线输出或四线输出命令提供了 IO0 上来自主机端的地址。数据可以按照两位一组通过 IO0 和 IO1 从存储器回送到主机，也可以按照四位（半字节）一组通过 IO0、IO1、IO2 和 IO3 回送。对于双线输出和四线输出命令，这种结构分别被称为 1-1-2 和 1-1-4 命令协议。

双线或四线输入 / 输出（I/O）命令会按照两比特一组通过 IO0 和 IO1 将主机的数据传输给存储器，或按照四比特（半字节）一组通过 IO0、IO1、IO2 和 IO3 从主机发送。数据也可以按照同样的方式回送给主机。对于双线 I/O 和四线 I/O 命令，这种结构分别被称为 1-2-2 和 1-4-4 命令协议。

FS-S 系列也支持 QPI 模式，在该模式下所有信息（包括指令、地址、修饰符和数据）都以 4 比特的宽度传输。这种形式被称为 4-4-4 命令协议。

命令的结构如下：

- 每个命令都以一个 8 位（一字节）的指令开始。但某些读指令被前一个读指令修改，因此该指令要从更早的命令中显示。这种形式被称为连续读取模式。器件处于连续读取模式时，由于该指令和用于初始化连续读取模式的读取命令相同，所以各个指令位不会在开始执行命令时被传输。在该模式下，命令会从读取地址开始。因此，连续读取模式会将处于相同类型的读取命令中的每个读命令内删除 8 个指令位。
- 该指令可以单独在设备的某些地址空间选择一个位置，或者跟着地址位一起选择。地址可能是 24 位或 32 位（字节边界）的地址。
- 多 IO 的串行外设接口允许并行传输 1、2 或 4 个地址和数据位。这样需要在信号引脚数量（IO 总线宽度）和信息传输速度间进行权衡。如果主机系统支持 2 比特或 4 比特带宽的 IO 总线，则通过使用能够提供并行 2 比特（双线）或并行 4 比特（四线）传输，可以提高存储器的性能。
- 在传统的多 IO SPI 模式下，指令后的所有传输宽度都由已发送的指令指定。以下表示的可能是其中一种传输类型：仅通过串行输入（SI）或串行输出（SO）信号传输单位；通过 IO0 和 IO1 信号每一（双线）传输以两位形式进行发送；或通过 IO0-IO3 信号每一（四线）传输以 4 位形式进行。在双比特或四比特组中，最低有效位是通过 IO0 信号发送的。其他更高有效位则按照权值大小通过编号更高的 IO 信号发送。单比特或并行比特组则按照从最高有效位到最低有效位的顺序进行传输。
- 在 QPI 模式下，所有信息，包括指令都按照 4 比特宽（四线传输）通过 IO0-IO3 信号进行发送。
- 双线和四线 I/O 读指令在发送地址后将发送一个指令修饰符（称为模式位），以指明下一个命令的类型与之前的命令相同，而它并不是明确的命令。因此，下一条命令只会提供新地址和模式位，并不提供指令字节。当某个命令序列中重复了相同的命令类型时，会减少发送每一个命令所需要的时间。
- 地址或模式位发送之后可以接着发送将被存储在存储器件内的写数据，也可能存在一个读延迟周期（数据返回给主机前）。
- 读取延迟（又称为虚拟周期）可能为 0 到几个 SCK 周期。
- 所有指令、地址、模式和数据信息均按字节宽度进行传输。地址按最高有效位优先方式被移入到器件内。所有数据则按最低地址字节优先方式进行传输。随后的数据字节按照从最低到最高字节地址的顺序（即字节地址递增）进行发送。
- 在编程、擦除或写周期（嵌入式操作）中，尝试读取闪存阵列的操作均被忽略。嵌入式操作仍继续进行而不受任何影响。执行嵌入式操作时只接受少数命令集。这些信息会在单独命令的说明部分进行介绍。正在进行编程、擦除或写入操作时，建议检查发送大多数命令到器件前正在写入（WIP）位是否为 0，从而确保是否可以接受新的命令。
- 执行时间会因命令不同而发生变化。提供用于读取某个执行命令的状态信息的命令，可确定命令结束的时间，并且确定该命令是否被成功执行。
- 虽然在某些情况下可以使用主机软件直接控制 SPI 接口信号，但是信号关系和时序详情则通常由主机系统和存储器设备的硬件接口处理。本节内容着重于软件接口，所以没有详细描述信号关系和时序。它的重点内容是每个命令中位传输的逻辑序列。下面是一些需要记住的通用信号关系说明。更多有关命令的比特级别格式和信号时序关系的信息，请参见第 14 页上的命令协议。

- 主机总是控制片选（CS#）、串行时钟（SCK）和单比特传输中的串行输入（SI/IO0）。存储器则驱动单比特读取传输的串行输出（SO/IO1）。主机和存储器在双比特和四比特传输期间交替驱动 IO0-IO3 信号。
- 所有命令的启动方式均相同，都是主机在 SCK 的第一个上升沿前会将 CS# 驱动为低电平，从而选择存储器。执行命令过程中，CS# 一直保持低电平状态。CS# 返回高电平状态时，命令结束。一般情况下，CS# 在八位传输的倍数期间保持低电平状态，用以传输字节粒度的信息。如果 CS# 在非 8 位对齐位置上返回高电平状态，则有些命令不被接受。

11.1 命令集汇总

11.1.1 扩展寻址

1. 始终要求一个 4 字节地址的指令，用于访问高达 32 Gb 的存储器。

命令名称	功能	指令（十六进制）
4READ	读取	13
4FAST_READ	快速读取	0C
4DOR	双线输出读取	3C
4QOR	四线输出读取	6C
4DIOR	双线 I/O 读取	BC
4QIOR	四线 I/O 读取	EC
4DDRQIOR	DDR 四线 I/O 读取	EE
4PP	页编程	12
4QPP	四线页编程	34
4P4E	参数 4 KB 擦除	21
4SE	擦除 64 KB	DC
4ECCRD	ECC 状态读取	18
4DYBRD	DYB 读取	E0
4DYBWR	DYBWR	E1
4PPBRD	PPB 读取	E2
4PPBP	PPB 编程	E3

2. 4 字节地址模式与 3 字节地址指令向后兼容。标准的 3 字节指令可以同地址长度配置位（CR2V[7]）控制的 4 字节地址模式结合使用。CR2V[7] 的默认值可从 CR2NV[7] 中加载（在执行上电、硬件复位或软件复位后），实现默认的 3 字节（24 位）或 4 字节（32 位）寻址。地址长度（CR2V[7]）被设为 1 时，将修改传统命令，要求地址字段中的 4 字节（32 位）地址。以下指令可以同 4 字节地址模式配置结合起来使用，以便从 3 字节地址字段转为 4 字节地址字节。

命令名称	功能	指令（十六进制）
READ	读取	03
FAST_READ	快速读取	0B
DOR	双线输出读取	3B
QOR	四线输出读取	6B
DIOR	双线 I/O 读取	BB
QIOR	四线 I/O 读取	EB
DDRQIOR	DDR 四线 I/O 读取	ED
PP	页编程	02
QPP	四线页编程	32
P4E	参数 4 KB 擦除	20
SE	擦除 64/256 KB	D8
RDAR	读取任何寄存器	65
WRAR	写入任何寄存器	71
EES	评估擦除状态	D0
OTPP	OTP 编程	42
OTPR	OTP 读取	4B
ECCRD	ECC 状态读取	19
DYBRD	DYB 读取	FA
DYBWR	DYBWR	FB
PPBRD	PPB 读取	FC
PPBP	PPB 编程	FD

11.1.2 按功能分类的命令汇总

表 51. FS-S 系列 命令集（按功能排列）

功能	命令名称	命令说明	指令值 (十六进制)	最大频率 (MHz)	地址长度 (字节)	QPI
读取器件 ID	RDID	读取 ID（JEDEC 制造商 ID 和 JEDEC CFI）	9F	133	0	支持
	RSFDP	读取 JEDEC 串行闪存可发现参数	5A	50	3	支持
	RDQID	四线读取 ID	AF	133	0	支持
	RUID	读取唯一 ID	4C	133	0	支持
寄存器访问	RDSR1	读取状态寄存器 1	05	133	0	支持
	RDSR2	读取状态寄存器 2	07	133	0	不支持
	RDCR	读取配置寄存器 1	35	133	0	不支持
	RDAR	读取任何寄存器	65	133	3 或 4	支持
	WRR	写寄存器（状态 1、配置 1）	01	133	0	支持
	WRDI	写禁用	04	133	0	支持
	WREN	写使能	06	133	0	支持
	WRAR	写入任何寄存器	71	133	3 或 4	支持
	CLSR	清除状态寄存器 1 - 擦除 / 编程失败复位 该命令可能被禁用，反而，将该指令值用于编程 / 擦除恢复命令 — 请参考第 57 页上的配置寄存器 3。	30	133	0	有
	CLSR	清除状态寄存器 1（备用指令）— 擦除 / 编程失败复位	82	133	0	支持
	4BEN	进入 4 字节地址模式	B7	133	0	不支持
	SBL	设置突发长度	C0	133	0	不支持
	EES	评估擦除状态	D0	133	3 或 4	支持
	ECCRD	ECC 读取	19	133	3 或 4	支持
	4ECCRD	ECC 读取	18	133	4	支持
	DLPRD	数据学习模式读取	41	133	0	不支持
	PNVDLR	编程 NV 数据学习寄存器	43	133	0	不支持
	WVDLR	写入易失性数据学习寄存器	4A	133	0	不支持
读取闪存阵列	READ	读取	03	50	3 或 4	不支持
	4READ	读取	13	50	4	不支持
	FAST_READ	快速读取	0B	133	3 或 4	不支持
	4FAST_READ	快速读取	0C	133	4	不支持
	DOR	双线输出读取	3B	133	3 或 4	不支持
	4DOR	双线输出读取	3C	133	4	不支持
	QOR	四线输出读取	6B	133	3 或 4	不支持
	4QOR	四线输出读取	6C	133	4	不支持
	DIOR	双线 I/O 读取	BB	66	3 或 4	不支持
	4DIOR	双线 I/O 读取	BC	66	4	不支持
	QIOR	四线 I/O 读取	EB	133	3 或 4	支持
	4QIOR	四线 I/O 读取	EC	133	4	支持
	DDRQIOR	DDR 四线 I/O 读取	ED	80	3 或 4	支持
	4DDRQIOR	DDR 四线 I/O 读取	EE	80	4	支持
编程闪存阵列	PP	页编程	02	133	3 或 4	支持
	4PP	页编程	12	133	4	支持
	QPP	四线页编程	32	133	3 或 4	不支持
	4QPP	四线页编程	34	133	4	不支持

表 51. FS-S 系列 命令集（按功能排列）（续）

功能	命令名称	命令说明	指令值 (十六进制)	最大频率 (MHz)	地址长度 (字节)	QPI
擦除闪存 阵列	P4E	参数 4 KB 扇区擦除	20	133	3 或 4	支持
	4P4E	参数 4 KB 扇区擦除	21	133	4	支持
	SE	擦除 64 KB	D8	133	3 或 4	支持
	4SE	擦除 64 KB	DC	133	4	支持
	BE	批量擦除	60	133	0	支持
	BE	批量擦除（备用指令）	C7	133	0	支持
擦除 / 编程 挂起 / 恢复	EPS	擦除 / 编程挂起	75	133	0	支持
	EPS	擦除 / 编程挂起（备用指令）	85	133	0	支持
	EPS	擦除 / 编程挂起（备用指令）	B0	133	0	支持
	EPR	擦除 / 编程恢复	7A	133	0	支持
	EPR	擦除 / 编程恢复（备用指令）	8A	133	0	支持
	EPR	擦除 / 编程恢复（备用指令） 该命令可能被禁用，反而，将该指令值用于清除状态命令 — 请参考 第 57 页上的配置寄存器 3。	30	133	0	支持
一次性编程 阵列	OTPP	OTP 编程	42	133	3 或 4	不支持
	OTPR	OTP 读取	4B	133	3 或 4	不支持
高级扇区 保护	DYBRD	DYB 读取	FA	133	3 或 4	支持
	4DYBRD	DYB 读取	E0	133	4	支持
	DYBWR	DYB 写入	FB	133	3 或 4	支持
	4DYBWR	DYB 写	E1	133	4	支持
	PPBRD	PPB 读取	FC	133	3 或 4	不支持
	4PPBRD	PPB 读取	E2	133	4	不支持
	PPBP	PPB 编程	FD	133	3 或 4	不支持
	4PPBP	PPB 编程	E3	133	4	不支持
	PPBE	PPB 擦除	E4	133	0	不支持
	ASPRD	ASP 读取	2B	133	0	不支持
	ASPP	ASP 编程	2F	133	0	不支持
	PLBRD	PPB 锁定位读取	A7	133	0	不支持
	PLBWR	PPB 锁定位写	A6	133	0	不支持
	PASSRD	密码读取	E7	133	0	不支持
	PASSP	密码编程	E8	133	0	不支持
	PASSU	密码解锁	E9	133	0	不支持
复位	RSTEN	软件复位使能	66	133	0	支持
	RST	软件复位	99	133	0	支持
	RESET	传统的软件复位	F0	133	0	不支持
	MBR	模式位复位	FF	133	0	支持
DPD	DPD	进入深度掉电模式	B9	133	0	支持
	RES	退出深度掉电模式	AB	133	0	支持

注意

47.若在 QPI 模式下发送不受支持的命令，这些命令将发生未定义行为。

11.1.3 读取器件标识

有多条用于读取有关器件制造商、器件类型和器件特性等信息的命令。来自不同供应商的 SPI 存储器使用了不同的命令和格式来读取存储器信息。FS-S 系列支持三种器件信息命令。

11.1.4 寄存器读取或写入

有多种寄存器用于报告嵌入式操作状态或控制器件配置选项。可以通过命令对这些寄存器进行读取或写入操作。寄存器包括易失性和非易失性位。寄存器中的非易失性位自动被擦除和编程，如单个（写）操作一样。

11.1.4.1 监控操作状态

主机系统通过监控状态寄存器中的“正在写入”（WIP）位，可以指出是否完成了写入、编程、擦除、挂起或其它嵌入式操作。读取状态寄存器 1 命令或读取任意寄存器命令可提供 WIP 位的状态。状态寄存器中的编程错误（P_ERR）和擦除错误（E_ERR）位指明最新的编程或擦除命令是否尚未完成。当 P_ERR 或 E_ERR 位为 1 时，WIP 位将保持为 1，以表示器件的繁忙状态并且无法接收新的操作命令。P_ERR 或 E_ERR 被设为 1 时，只有状态读取（RDSR1 05h）、读取任何寄存器（RDAR 65h）、状态清除（CLSR 30h 或 82h）和软件复位（RSTEN 66h、RST 99h 或 RESET F0h）为有效命令。并且必须依次发送清除状态寄存器（CLSR）和写入禁用（WRDI）命令，才能使器件返回待机状态。清除状态寄存器会清除 WIP、P_ERR 和 E_ERR 位。WRDI 会清除 WEL 位。另外，也可以通过硬件复位或软件复位（RST 或 RESET），使器件返回待机状态。

11.1.4.2 配置

可以使用一些命令来读取、写入和保护寄存器，从而控制接口路径宽度、接口时序、接口地址长度和数据保护相关方面。

11.1.5 读取闪存阵列

可以从存储器的任意字节边界上读取数据。数据字节从低位字节地址到高位字节地址被连续读取，直到主机将 CS# 置高（表示结束数据传输）为止。如果字节地址达到存储器阵列的最大地址，读取操作将从阵列的地址 0 重新开始。

突发循环读取可以通过设置突发长度（SBL 77h）命令按要求的循环读取长度和对齐实现，请参见第 91 页上的设置突发长度（SBL C0h）。突发循环读取仅适用于四线 I/O、四线输出和 QPI 模式。

有些不同的读取命令用于指定不同的访问延迟和数据路径宽度。双倍数据速率（DDR）命令还定义了地址和数据位与两个 SCK 边沿的关系：

- 读取命令在每个 SCK 上升沿上通过 SI/IO0 信号提供一位，并在每个 SCK 下降沿上通过 SO/IO1 信号返回一个数据位。使用该命令，地址和返回数据间并没有延迟，但是最大 SCK 速率被限于 50MHz。
- 使用其它读取命令，地址和返回数据间有一定的延迟，但能够以更高的 SCK 频率进行操作。延迟时间取决于配置寄存器中的读取延迟值。
- 快速读取命令在每个 SCK 上升沿上通过 SI/IO0 信号提供一个地址位，并在每个 SCK 下降沿上通过 SO/IO1 信号返回一个数据位。
- 双线或四线输出读取命令在每个 SCK 上升沿上通过 SI/IO0 引脚提供地址位，并在每个 SCK 下降沿上通过 IO0 - IO3 信号返回两个或四个数据位。
- 双线或四线 I/O 读取命令在每个 SCK 上升沿上提供两个或四个地址位，并在每个 SCK 下降沿上通过 IO0 - IO3 信号返回两个或四个数据位。
- 四线双倍数据速率读取命令在每个 SCK 边沿上提供了四个地址位，并在每个 SCK 边沿上通过 IO0 - IO3 信号返回四个数据位。

11.1.6 编程闪存阵列

编程数据需要两条命令：写使能（WREN）、页编程（PP）和四线页编程（QPP）。通过页编程和四线页编程命令，可以在一次操作中编程从 1 字节到连续 256 个或 512 个字节的数据。编程操作可以使位值保持为 1，或从 1 改为 0。要想将位值从 0 变为 1，需要使用擦除操作。

11.1.7 擦除闪存阵列

参数扇区擦除、扇区擦除或批量擦除命令将一个扇区中或整个存储器阵列中的所有位设置为 1。先要将某位擦除为 1，然后才能将它编程为 0。可以独立将每个位从 1 编程为 0，但必须按一个扇区或整个阵列（批量）的级别将位从 0 擦除为 1。执行擦除命令前，必须执行写使能（WREN）命令。

11.1.8 OTP、块保护和高级扇区保护

器件有一些命令用于读取和编程保存永久性保护数据（例如，序列号）的可一次性编程（OTP）阵列。有一些命令用于控制防止进行编程和擦除操作的闪存存储器阵列扇区连续组（块）。还有一些命令用于控制防止对哪个个别闪存存储器阵列扇区进行编程和擦除操作。

11.1.9 复位

有一些命令用于在器件上电后将器件复位为默认状态。然而，软件复位命令对 FREEZE 或 PPB 锁定位没有产生影响。在所有其它方面，软件复位与硬件复位相同。

有一个命令用于复位（退出）连续读取模式。

11.1.10 DPD

在 FS-S 系列器件中，支持深度掉电（DPD）模式。如果使用 DPD（B9h）命令使器件进入 DPD 模式，接口待机电流将为（ I_{DPD} ）。仅在状态寄存器 1 易失性正在写入（WIP）位被清零（SR1V[0] = 0），表示器件不执行嵌入式算法时，才可接受 DPD 命令。在 DPD 模式下，器件将忽略所有指令（从 DPD 释放（RES ABh）命令除外）— 经过 t_{RES} 延迟后，该命令使器件返回接口待机状态。

11.1.11 预留

有些指令留作日后使用。在这一代 FS-S 系列系列产品中，有些命令指令可能未被使用，并且不会影响器件的操作；有些命令指令则会返回未定义结果。

这类命令中的一部分被保留，以确保可以执行传统或交替源器件命令，而不会产生影响。这样允许传统软件发送一些与当前一代 FS-S 系列系列无关的命令，并要保证这些命令不会导致意外行为。

有些命令则被保留，用于本文档中未提到的特别版本 FS-S，或用于下一代产品。使新的主机存储控制器设计可以灵活使用这些命令指令。如果在发布本文档的修订版本时已确定好，那么命令格式也会得到定义。

11.2 标识命令

11.2.1 读取标识 (RDID 9Fh)

通过读取标识 (RDID) 命令, 可以对制造商标识、器件标识和通用闪存接口 (CFI) 信息进行读取访问。制造商标识由 JEDEC 分配, CFI 结构由 JEDEC 标准定义。器件标识和 CFI 值则由赛普拉斯分配。

JEDEC 通用闪存结构 (CFI) 规范定义了器件信息结构, 允许供应商在整个闪存器件系列中使用它指定的软件闪存管理程序 (驱动器)。软件支持变得与器件和 JEDEC 制造商 ID 无关, 并且对指定的闪存器件系列前后兼容。系统供应商可以利用当前使用器件的 CFI 信息来配置产品系列驱动器, 从而对闪存驱动器进行标准化, 与软件长期兼容。

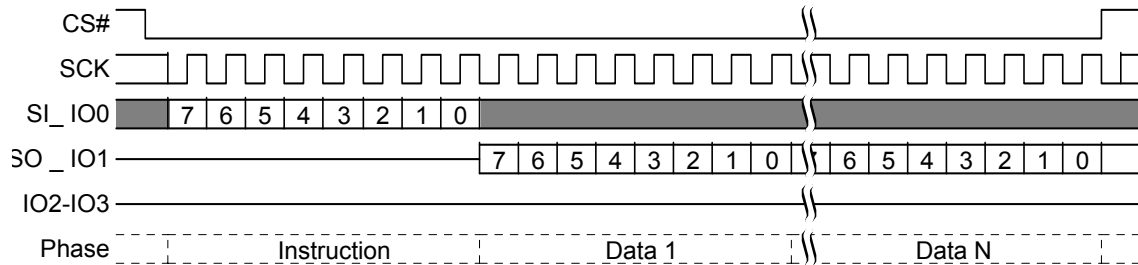
在执行编程、擦除或写入周期期间发送的任何 RDID 命令被忽略, 并且不会影响到这些操作的执行。

RDID 指令通过 SI 传入到器件内。RDID 指令的最后一位传入到器件后, 通过 SO 信号连续输出一个制造商标识字节、两个器件标识、扩展型器件标识以及 CFI 信息字节。这全部信息被称为 ID-CFI。有关 ID-CFI 内容的详细说明, 请参见第 122 页上的器件 ID 和通用闪存接口 (ID-CFI) 地址映射 — 标准。

所定义的 ID-CFI 地址结束后, 继续移出的输出保存未定义数据。输出数据期间可随时将 CS# 置为逻辑高电平状态, RDID 命令序列立即被终止。

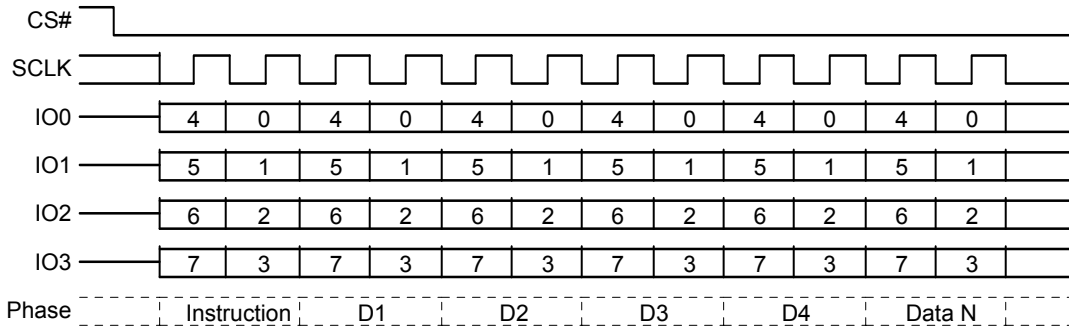
RDID 命令的最大时钟频率为 133 MHz。

图 45. 读取标识 (RDID) 命令序列



在 QPI 模式下, 器件也支持该命令。在 QPI 模式下, 将通过 IO0-IO3 信号传入指令并传出返回数据。

图 46. QPI 模式下的读取标识 (RDID) 命令



11.2.2 四线读取标识 (RDQID AFh)

通过四线读取标识 (RDQID) 命令，可以对制造商标识、器件标识和通用闪存接口 (CFI) 读取。该命令是 RDID 命令的交替选项，用于在 QPI 模式下读取同样的信息。在所有其它方面，该命令与 RDID 命令相同。

只有器件处于 QPI 模式 (CR2V[6] = 1) 时，才能识别该命令。该指令通过 IO0-IO3 传入到器件内。该指令的最后一位传入到器件后，通过 IO0-IO3 信号连续输出一个制造商标识字节、两个器件标识、扩展型器件标识以及 CFI 信息字节。这全部信息被称为 ID-CFI。有关 ID-CFI 内容的详细说明，请参见第 122 页上的器件 ID 和通用闪存接口 (ID-CFI) 地址映射 — 标准。

所定义的 ID-CFI 地址结束后，继续移出的输出保存未定义数据。输出数据期间可以随时将 CS# 置为逻辑高电平状态，该命令序列立即被终止。

该命令的最大时钟频率为 133 MHz。输出数据期间可以随时将 CS# 置为逻辑高电平状态，该命令序列立即被终止。

图 47. 四线模式下的四线读取标识 (RDID) 命令序列

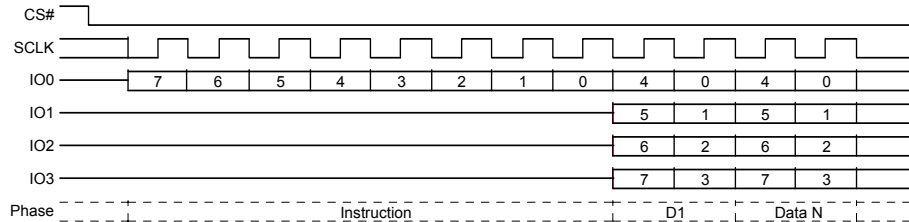
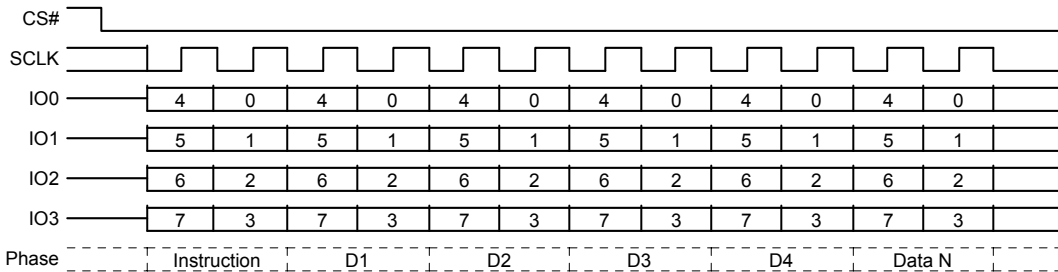


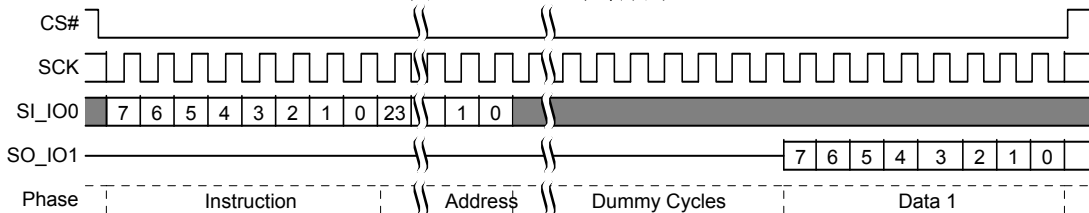
图 48. QPI 模式下的四线读取标识 (RDID) 命令序列



11.2.3 读取串行闪存可发现参数 (RSFDP 5Ah)

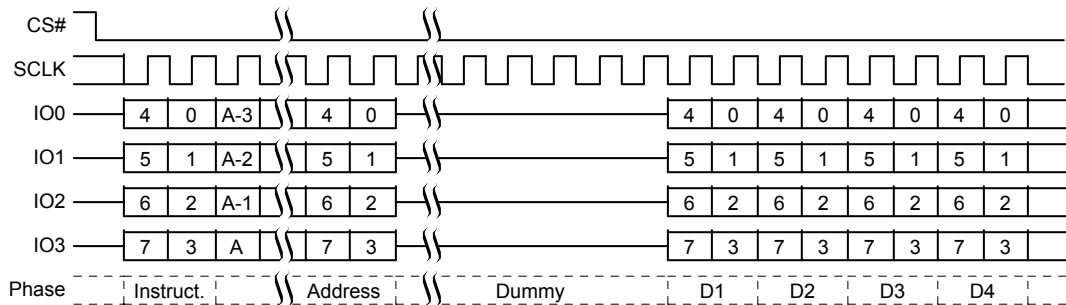
该命令通过 SI 信号首先传输指令代码 “5Ah”，然后传输 24 位地址 000000h，最后是 8 个虚拟周期。虚拟周期结束后，通过 SO 从 SCK 下降沿开始传出 SFDP 字节。SFDP 字节始终按最高有效位优先的方式传出。如果 24 位地址被设为其它值，则 SFDP 空间中的选定位置作为读取数据的起始点。这样可以随机访问 SFDP 空间中的任意参数。传送 RSFDP 命令的频率最高可达 50 MHz 频率。

图 49. RSFDP 命令序列



该命令也支持 QPI 模式。在 QPI 模式下，将通过 IO0-IO3 信号传入指令并输出返回数据。

图 50. QPI 模式下的 RSFDP 命令序列



11.2.4 读取唯一 ID (RUID 4Ch)

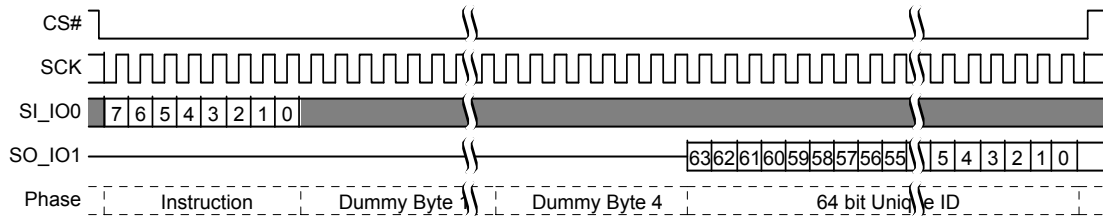
读取标识 (RUID) 命令允许读取访问工厂设置的只读 64 位编号，该编号是每个器件的唯一值。

RUID 指令在四个虚拟字节或 16 个虚拟字节 QPI (32 个时钟周期) 后通过 SI 被传入器件内。延迟周期 (虚拟字节) 为器件的内部电路提供需要的时间来访问初始地址的数据。在这些延迟周期内，IO0-IO3 上的数据值为“无需关注”，该值可能为高阻状态。

然后，唯一 ID 的 8 个字节通过 SO/IO1 被连续输出。

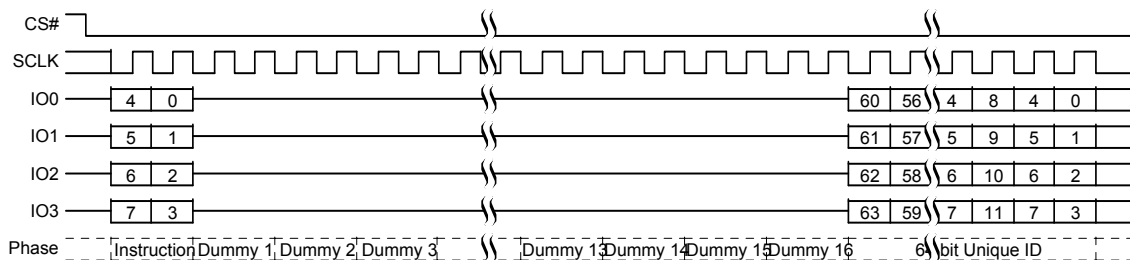
所定义的唯一 ID 地址结束后，继续移出的输出会提供未定义数据。输出数据期间可以随时将 CS# 置为逻辑高电平状态，RUID 命令序列立即被终止。

图 51. 读取唯一 ID (RUID) 命令序列



该命令也支持 QPI 模式。在 QPI 模式下，将通过 IO0-IO3 信号传入指令及传出返回数据。

图 52. QPI 模式下的读取唯一 ID (RUID) 命令



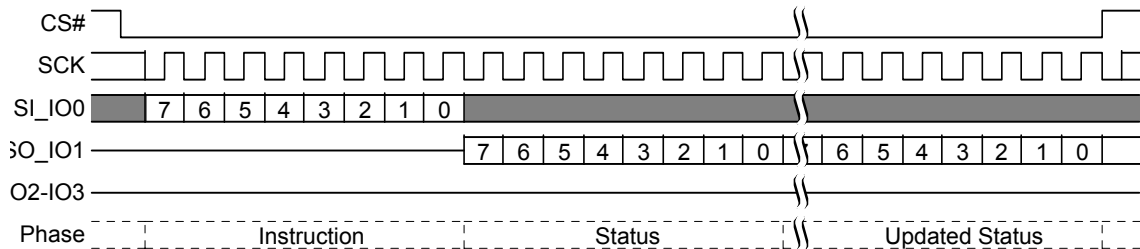
11.3 寄存器访问命令

11.3.1 读取状态寄存器 1 (RDSR1 05h)

通过读取状态寄存器 1 (RDSR1) 命令，可以从 SO/IO1 读取状态寄存器 1 的内容。

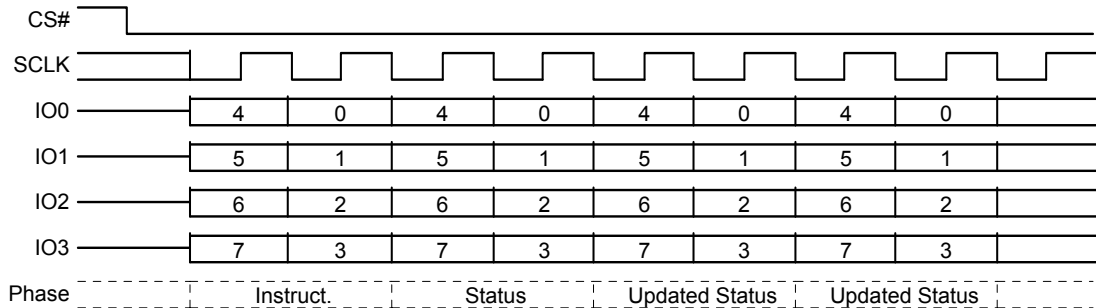
可以随时读取易失性状态寄存器 1 (SR1V) 的内容，甚至在执行编程、擦除或写入操作期间也能进行。通过提供 8 个时钟周期的倍数，可以连续读取状态寄存器 1。每 8 个读取周期更新一次状态。RDSR1 (05h) 命令的最大时钟频率为 133 MHz。

图 53. 读取状态寄存器 1 (RDSR1) 命令序列



该命令也支持 QPI 模式。在 QPI 模式下，将通过 IO0–IO3 信号传入指令及传出返回数据。

图 54. QPI 模式下的读取状态寄存器 1 (RDSR1) 命令

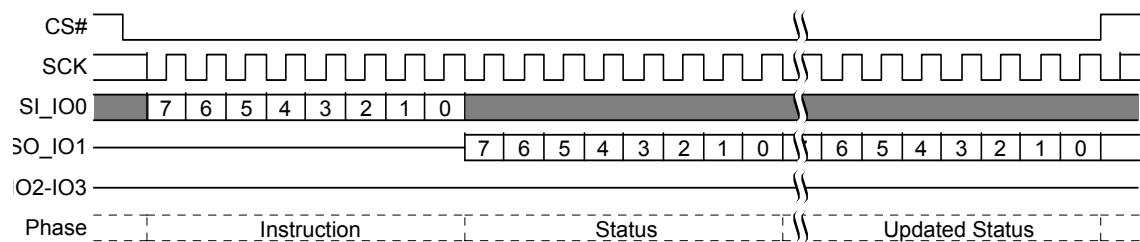


11.3.2 读取状态寄存器 2 (RDSR2 07h)

通过读取状态寄存器 2 (RDSR2) 命令，可以从 SO/IO1 读取状态寄存器 2 的内容。

可以随时读取状态寄存器 2 的内容，就算在执行编程、擦除或写入操作期间也能进行。通过提供 8 个时钟周期的倍数，可以连续读取状态寄存器 2。每 8 个读取周期更新一次状态。RDSR2 命令的最大时钟频率为 133 MHz。

图 55. 读取状态寄存器 2 (RDSR2) 命令



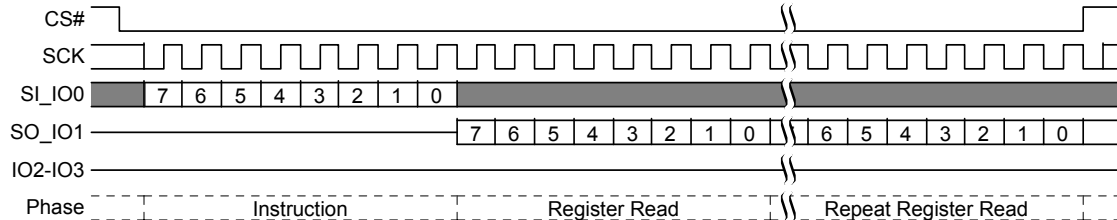
在 QPI 模式下，可以使用读取任意寄存器命令来读取状态寄存器 2，请参考第 89 页上的读取任何寄存器 (RDAR 65h)。

11.3.3 读取配置寄存器 (RDCR 35h)

读取配置寄存器 (RDCR) 命令允许从 SO/IO1 读取易失性配置寄存器 (CR1V) 的内容。

通过提供 8 个时钟周期的倍数，可以连续读取 CR1V。可以随时读取配置寄存器的内容，就算在执行编程、擦除或写入操作期间也能进行。

图 56. 读取配置寄存器 (RDCR) 命令序列



在 QPI 模式下，可以使用读取任意寄存器命令来读取配置寄存器 1，请参考第 89 页上的读取任何寄存器 (RDAR 65h)。

11.3.4 写入寄存器 (WRR 01h)

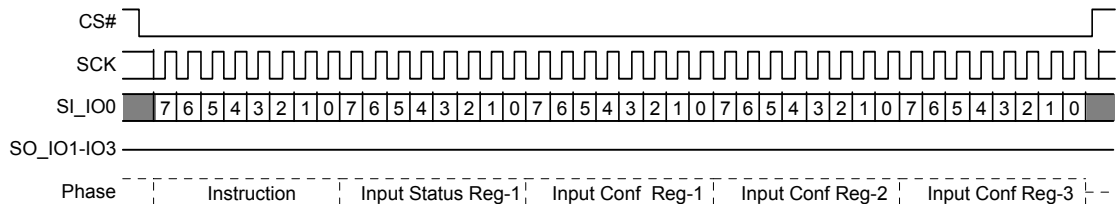
写入寄存器 (WRR) 命令允许将新值写入到状态寄存器 1 和配置寄存器 1。器件接受写入寄存器 (WRR) 命令前，必须接收到写使能 (WREN) 命令。成功解码写入使能 (WREN) 命令后，器件将设置状态寄存器中的写入使能锁存 (WEL) 位，以使能任意写入操作。

通过将指令和数据字节移至 SI/IO0，可以输入写入寄存器 (WRR) 命令。状态寄存器的长度为一个数据字节。

WRR 操作先会擦除寄存器，然后编程新值；该过程被定义为一个操作。如果 WRR 操作失败，则写入寄存器 (WRR) 命令将设置 P_ERR 或 E_ERR 位。有关错误位的说明，请参见第 50 页上的易失性状态寄存器 1 (SR1V)。所有留作日后使用的状态或配置寄存器位都必须被写为“0”。

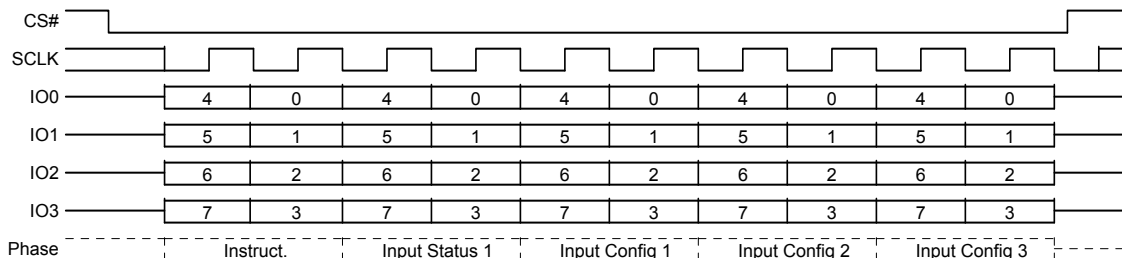
数据字节的第 8 位或第 16 位被锁存后，必须将 CS# 驱动为逻辑高电平状态。否则，无法执行写入寄存器 (WRR) 命令。经过第 8 个周期，如果 CS# 被驱动为高电平，则只能对状态寄存器 1 进行写操作；在第 16 个周期后，可对状态和配置寄存器进行写操作。CS# 被驱动为逻辑高电平状态后，会立即启动自定时写入寄存器 (WRR) 操作。在执行写入寄存器 (WRR) 操作的过程中，还可以读取状态寄存器，以检查‘正在写入’ (WIP) 位的值。执行自定时写入寄存器 (WRR) 操作期间，正在写入 (WIP) 位为 1；完成该操作时，WIP 位为 0。完成写入寄存器 (WRR) 操作时，写使能锁存 (WEL) 位将被设置为“0”。WRR 命令的最大时钟频率为 133 MHz。

图 57. 写入寄存器 (WRR) 命令序列



在 QPI 模式下，器件也支持该命令。在 QPI 模式下，通过 IO0-IO3 信号将指令和数据传入到芯片内。

图 58. QPI 模式下的写入寄存器 (WRR) 命令序列



通过写入寄存器（WRR）命令，用户可以修改非易失性状态寄存器 1 或易失性状态寄存器 1 中的块保护（BP2、BP1 和 BP0）位值，用于定义只读区域的大小。BPNV_O 位（CR1NV[3]）用于控制 WRR 对状态寄存器 1 进行写入的是非易失性还是易失性版本。CR1NV[3] = 0 时，WRR 对 SR1NV[4:2] 进行写入。CR1NV[3] = 1 时，WRR 将写入 SR1V[4:2]。

用户也可以通过写入寄存器（WRR）命令将状态寄存器的写入禁用位（SRWD）设为“1”或“0”。状态寄存器的写入禁用（SRWD）位和写入保护（WP#）信号允许对 BP 位进行硬件保护。

状态寄存器的写入禁用（SRWD）位为“0”时（它的默认状态），无论写入保护（WP#）信号被驱动为逻辑高或逻辑低状态，都可以对状态寄存器进行写操作，前提是已经通过写入使能（WREN）命令设置写入使能锁存（WEL）位。

状态寄存器的写入禁用（SRWD）位被设置为“1”时，需要考虑写入保护（WP#）信号的状态：

- 如果写入保护（WP#）信号被驱动为逻辑高状态，可以对状态和配置寄存器进行写操作，假定通过启动写入使能（WREN）命令已经将写入使能锁存（WEL）位设置为 1。
- 如果写入保护（WP#）信号被驱动为逻辑低状态，即使通过写入使能（WREN）命令将写入使能锁存（WEL）位设置为 1，也不能对状态和配置寄存器进行写操作。尝试对状态和配置寄存器进行的写操作被拒绝，不会执行，并且不提供错误指示。因此，受保护的存储区域中（由状态寄存器中的块保护（BP2、BP1 和 BP0）位保护）的所有数据字节也受 WP# 的硬件保护。

通过以下方式，可以提供 WP# 硬件保护：

- 将写入保护（WP#）信号驱动为逻辑低状态后设置状态寄存器的写入禁用（SRWD）位；
- 或者，设置状态寄存器的写入禁用（SRWD）位为“1”后将写入保护（WP#）信号驱动为逻辑低状态。

释放硬件保护的唯一方法是将写入保护（WP#）信号拉到逻辑高状态。如果永久性将 WP# 置高，则 BP 位的硬件保护永远不会被激活。

表 52. 块保护模式

WP#	SRWD 位	模式	寄存器的写保护	存储器内容	
				受保护的区域	无保护的区域
1	1	软件保护	可以写入状态和配置寄存器（如果 WREN 命令已经设置了 WEL 位）。可以更改 SRWD、BP2、BP1 和 BP0 以及配置寄存器中的值。	防止页编程、扇区擦除和批量擦除操作	就绪接收页编程和扇区擦除命令
1	0				
0	0				
0	1	硬件保护	状态和配置寄存器受硬件写保护。不可以更改 SRWD、BP2、BP1 和 BP0 以及配置寄存器中的值。	防止页编程、扇区擦除和批量擦除操作	就绪接收页编程或擦除命令

注意

48. 器件从赛普拉斯出厂时，状态寄存器的原始状态为 00h。

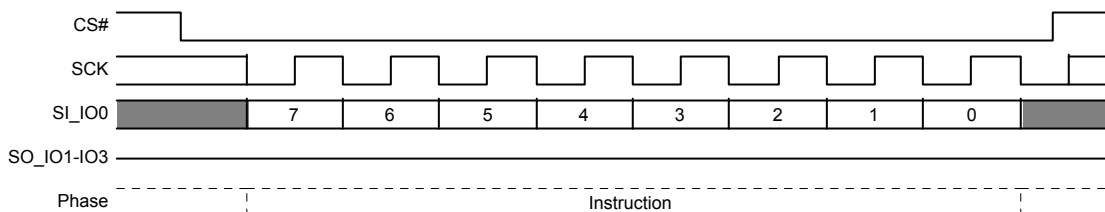
49. 四线模式被使能时（CR1V[1] = 1），硬件保护被禁用。WP# 变为 IO2，因此，不能使用它。

11.3.5 写入使能 (WREN 06h)

写入使能 (WREN) 命令会将状态寄存器 1 (SR1V[1]) 的写入使能锁存 (WEL) 位设置为 1。需要发送写入使能 (WREN) 命令以使能写入、编程和擦除操作，这样才能将写入使能锁存 (WEL) 位设置为 1。

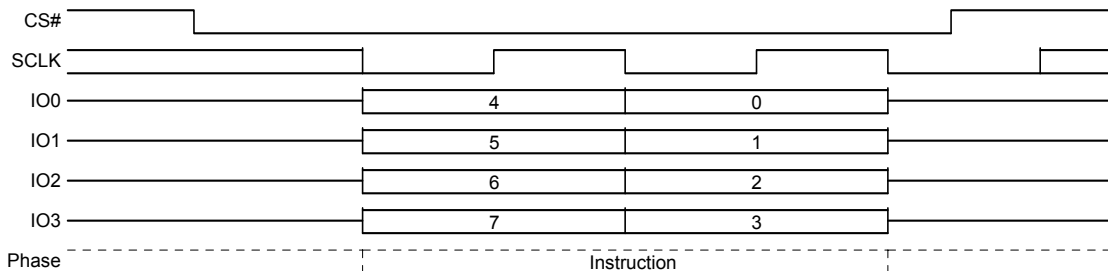
指令字节的第 8 位被锁存在 SI/IO0 上之后，必须将 CS# 信号驱动为逻辑高电平状态。否则，将不会执行写入使能操作。

图 59. 写入使能 (WREN) 命令序列



该命令也支持 QPI 模式。在 QPI 模式下，通过 IO0-IO3 信号将指令传入到器件内。

图 60. QPI 模式下的写入使能 (WREN) 命令序列



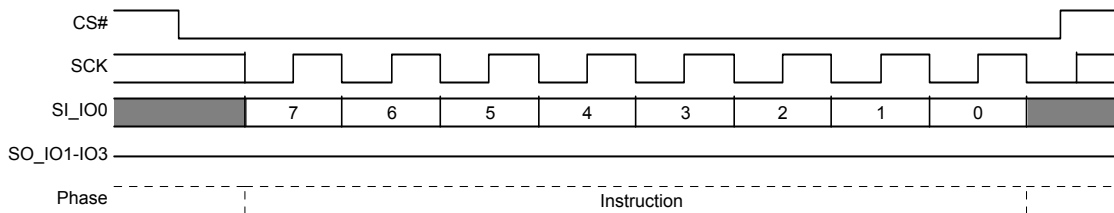
11.3.6 写入禁用 (WRDI 04h)

写入禁用 (WRDI) 命令会将状态寄存器 1 的写入使能锁存 (WEL) 位 (SR1V[1]) 清除为 “0”。

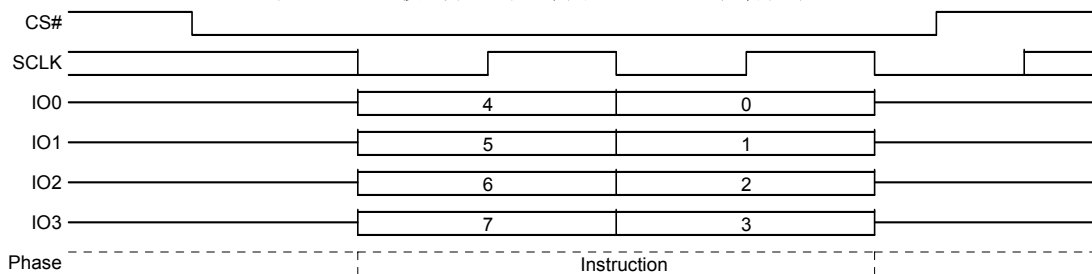
要想将写入使能锁存 (WEL) 位清除为 “0”，需要发送写入禁用 (WRDI) 命令，以便禁止页编程 (PP)、扇区擦除 (SE)、批量擦除 (BE)、写入寄存器 (WRR 或 WRAR)、OTP 编程 (OTPP) 和其它命令。这些命令需要 WEL 设为 “1” 才可执行。用户可以使用 WRDI 命令来保护存储区域，防止可能使存储器内容受损害的意外写操作。WIP 位 = 1 时，WRDI 命令在执行嵌入式操作期间被忽略。

指令字节的第 8 位被锁存在 SI/IO0 上之后，必须将 CS# 信号驱动为逻辑高电平状态。否则，将不会执行写入禁用操作。

图 61. 写入禁用 (WRDI) 命令序列



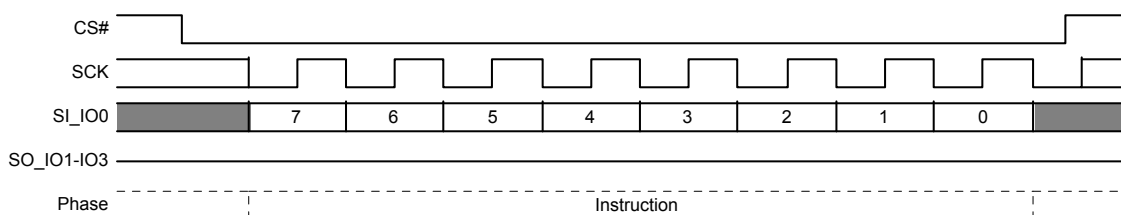
该命令也支持 QPI 模式。在该模式下，通过 IO0-IO3 信号将指令传入到器件内。

图 62. QPI 模式下的写入禁用 (WRDI) 命令序列


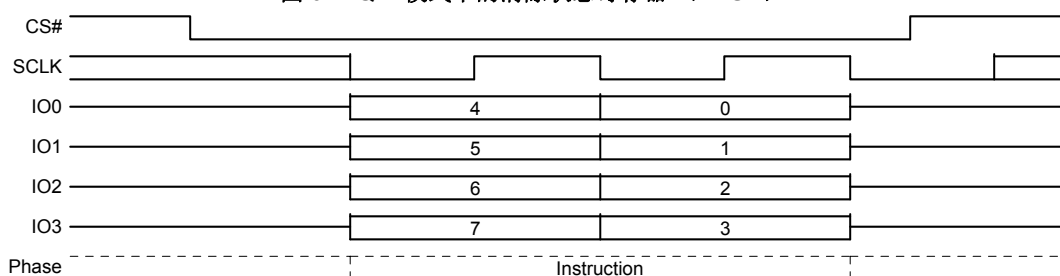
11.3.7 清除状态寄存器 (CLSR 30h 或 82h)

清除状态寄存器命令用于复位 SR1V[5] 位 (擦除失败标志) 和 SR1V[6] 位 (编程失败标志)。执行清除状态寄存器命令前, 不需要设置 WEL 位。即使器件由于 WIP 被设为 1 而保持繁忙状态, 它也会接受清除状态寄存器命令, 这是因为当两个错误位中的一个被设置时, 器件也会保持繁忙状态。执行该命令后, WEL 位没有发生变化。

可以禁用传统的清除状态寄存器 (CLSR 30h) 指令, 并且使用 30h 指令值用于代替编程 / 擦除恢复命令, 请参考第 57 页上的 9.6.5 配置寄存器 3 节。可以随时使用清除状态寄存器代替指令 (CLSR 82h), 用于清除状态寄存器。

图 63. 清除状态寄存器 (CLSR) 命令序列


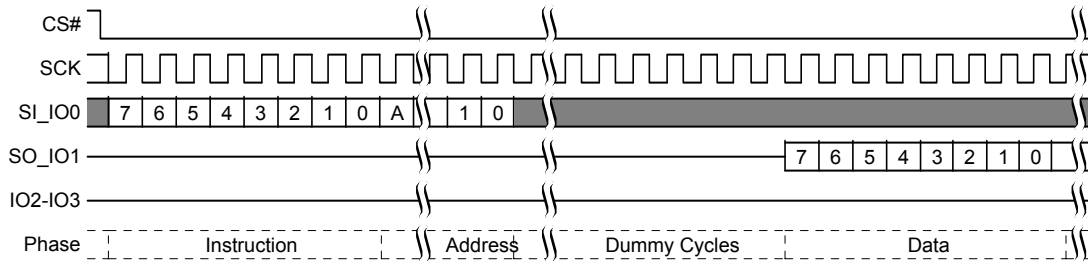
该命令也支持 QPI 模式。在 QPI 模式下, 通过 IO0-IO3 信号将指令传入到器件内。

图 64. QPI 模式下的清除状态寄存器 (CLSR)


11.3.8 ECC 状态寄存器读取（ECCRD 19h 或 4EECRD 18h）

要读取 ECC 状态寄存器，命令后面是 ECC 单元（16 字节）地址，地址的四个最低有效位（LSB）必须置为零。这些地址字节后面是虚拟周期数（虚拟周期数则由 CR2V[3:0] 中的读取延迟值选定的）。接下来是所选 ECC 单元的 ECC 寄存器的 8 位数据，通过 SO/IO1 进行 16 次传出数据，ECC 单元中每一字节一次。如果 CS# 为低电平，则下一个 ECC 单元状态通过 SO/IO1 发送 16 次，ECC 单元中的每个字节一次，一直到 CS # 为高电平为止。ECC READ 命令的最大工作时钟频率为 133 MHz。请参见第 101 页上的[自动纠错码（ECC）](#)了解更多有关 ECC 单元的信息。

图 65. ECC 状态寄存器读取命令序列

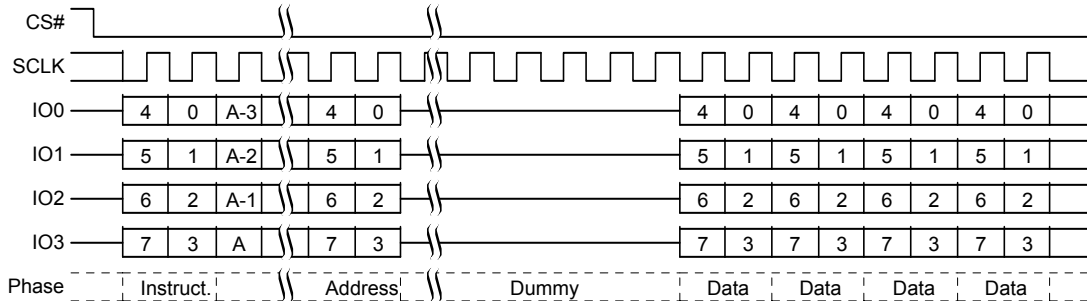


注意：

- 50.A = 地址的 MSB = 23（CR2V[7] = 0），或 31（CR2V[7] = 1 和命令 19h）。
- 51.A = 地址的 MSB = 31（使用命令 18h）。

该命令也支持 QPI 模式。在 QPI 模式下，通过 IO0-IO3 信号传入指令和地址及返回数据。

图 66. ECC 状态寄存器读取 QPI 模式，命令序列



注意：

- 52.A = 地址的 MSB = 23（CR2V[7] = 0），或 31（CR2V[7] = 1 和命令 19h）。
- 53.A = 地址的 MSB = 31（使用命令 18h）。

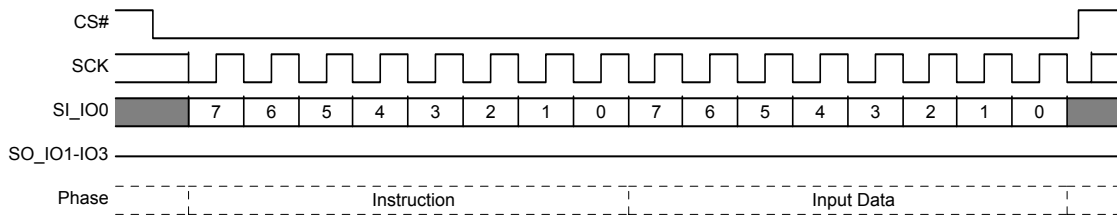
11.3.9 编程 NVDLR (PNVDLR 43h)

接受编程 NVDLR (PNVDLR) 命令前，器件必须发送写入使能 (WREN) 命令并对其进行解码。成功解码写入使能 (WREN) 命令后，器件将设置写入使能锁存 (WEL) 位以使能 PNVDLR 操作。

通过将指令和数据字节移至 SI/IO0，可以输入 PNVDLR 命令。

数据字节的第八 (8) 位被锁存后，必须将 CS# 驱动为逻辑高电平状态。否则，不能执行 PNVDLR 命令。CS# 被驱动为逻辑高电平状态时，会立即启动自定时 PNVDLR 操作。在执行 PNVDLR 操作的过程中，可以读取状态寄存器以检查正在写入 (WIP) 位的值。执行自定时 PNVDLR 操作期间，正在写入 (WIP) 位为 “1”；完成该操作时，WIP 位为 “0”。PNVDLR 命令可以报告状态寄存器中 P_ERR 位的编程错误。完成 PNVDLR 操作时，写入使能锁存 (WEL) 被设置为 “0”。PNVDLR 命令的最大时钟频率为 133 MHz。

图 67. 编程 NVDLR (PNVDLR) 命令序列

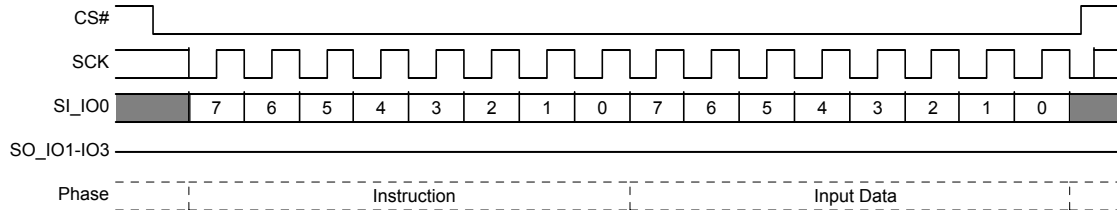


11.3.10 写入 VDLR (WVDLR 4Ah)

接受写入 VDLR (WVDLR) 命令前，器件必须发送写入使能 (WREN) 命令并对其进行解码。成功解码写入使能 (WREN) 命令后，器件将设置写入使能锁存 (WEL) 位以使能 WVDLR 操作。

通过将指令和数据字节移至 SI/IO0，可以输入 WVDLR 命令。数据字节的第八 (8) 位被锁存后，必须将 CS# 驱动为逻辑高电平状态。否则，不能执行 WVDLR 命令。CS# 被驱动为逻辑高电平状态后，会立即启动 WVDLR 操作。该 WVDLR 命令的最大时钟频率为 133 MHz。

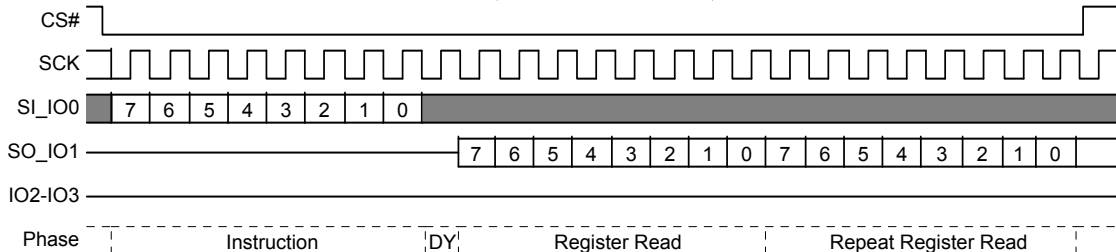
图 68. 写入 VDLR (WVDLR) 命令序列



11.3.11 数据学习模式读取 (DLPRD 41h)

该指令被传入到 SI/IO0，然后通过 SO/IO1 传出 8 位 DLP。通过提供 8 个时钟周期的倍数，可以连续读取 DLP。DLPRD 命令的最大工作时钟频率为 133 MHz。

图 69. DLP 读取 (DLPRD) 命令序列

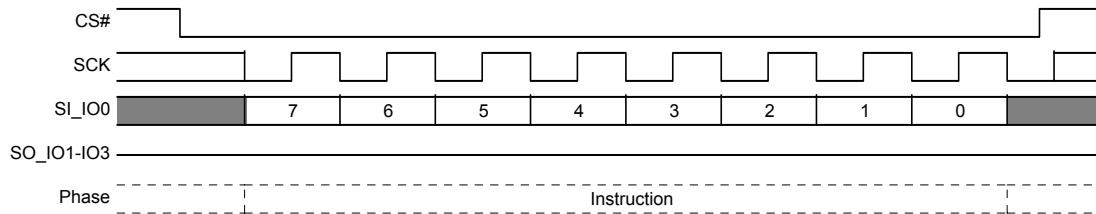


11.3.12 进入 4 字节地址模式（4BAM B7h）：

通过进入 4 字节地址模式（4BAM）命令，可以将易失性地址长度（ADS）位（CR2V[7]）设为 1，使几乎所有 3 字节地址命令要求 4 字节地址。读取 SFDP（RSFDP）命令是唯一不受地址长度位影响的 3 字节命令。JEDEC JESD216 B 标准要求 RSFDP 命令一直使用 3 字节地址。

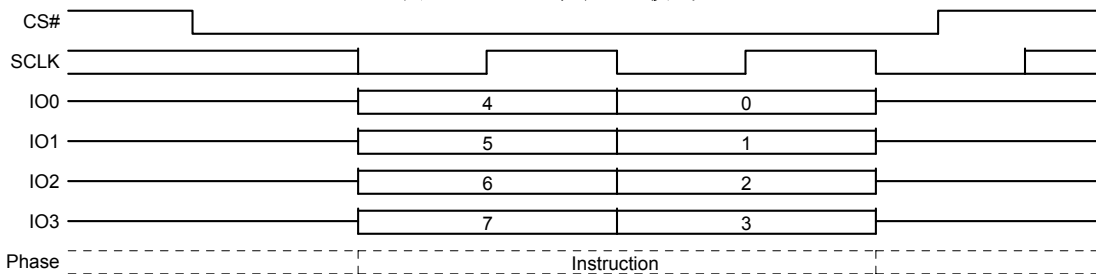
通过硬件或软件复位，可以退出 4 字节地址模式。

图 70. 进入 4 字节地址模式（4BEN B7h）命令序列



该命令也支持 QPI 模式。在 QPI 模式下，通过 IO0-IO3 信号将指令传入到器件内。

图 71. 进入 4 字节地址模式



11.3.13 读取任何寄存器（RDAR 65h）

通过读取任何寄存器（RDAR）命令，可以读取所有器件寄存器，包括非易失性和易失性寄存器。该指令后面是 3 或 4 字节地址（取决于地址长度配置 CR2V[7]），最后是由 CR2V[3:0] 指定的延迟（虚拟）周期数量。然后将返回选定的寄存器的内容。如果继续执行读取访问，将返回同样的寄存器内容，直到命令终止为止。每个 RDAR 命令只能读取一个寄存器。

读取未定义地址会返回未定义数据。

在执行嵌入式操作期间，可以使用 RDAR 命令来读取状态寄存器 1（SR1V）。

RDAR 命令不用于读取反映更大阵列的寄存器，例如：ECCSR、PPBAR 和 DYBAR。需要使用一些单独命令来选择并读取被访问阵列中的位置。

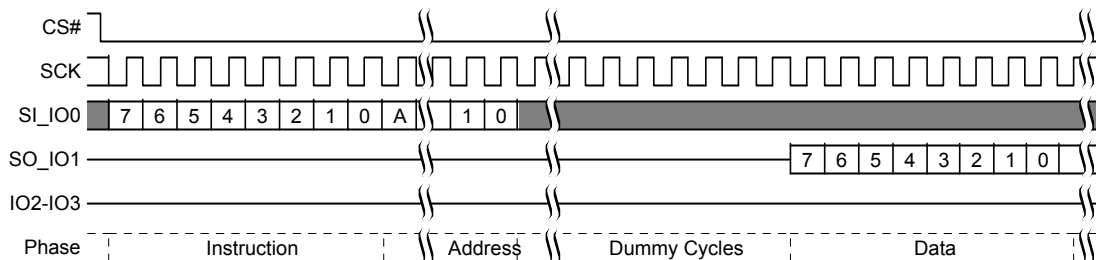
如果将 ASPR[2] 编程为 0，则表示选择 ASP 密码保护模式，RDAR 命令会从 PASS 寄存器中读取无效数据。

表 53. 寄存器地址映射

字节地址（十六进制）	寄存器名称	说明
00000000	SR1NV	非易失性状态和配置寄存器
00000001	N/A	
00000002	CR1NV	
00000003	CR2NV	
00000004	CR3NV	
00000005	CR4NV	
...	N/A	非易失性数据学习寄存器
00000010	NVDLR	
...	N/A	

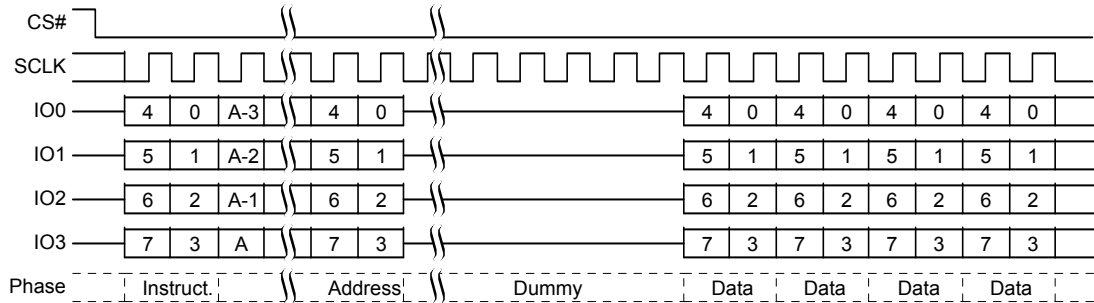
表 53. 寄存器地址映射（续）

字节地址（十六进制）	寄存器名称	说明
00000020	PASS[7:0]	非易失性密码寄存器
00000021	PASS[15:8]	
00000022	PASS[23:16]	
00000023	PASS[31:24]	
00000024	PASS[39:32]	
00000025	PASS[47:40]	
00000026	PASS[55:48]	
00000027	PASS[63:56]	
...	N/A	
00000030	ASPR[7:0]	非易失性 ASP 寄存器
00000031	ASPR[15:8]	
...	N/A	
00800000	SR1V	易失性状态和配置寄存器
00800001	SR2V	
00800002	CR1V	
00800003	CR2V	
00800004	CR3V	
00800005	CR4V	
...	N/A	
00800010	VDLR	易失性数据学习寄存器
...	N/A	
00800040	PPBL	易失性 PPB 锁定寄存器
...	N/A	

图 72. 读取任意寄存器命令序列

注释:

54.A = 地址的 MSB = 23 (CR2V[7] = 0), 或 31 (CR2V[7] = 1)。

该命令也支持 QPI 模式。在 QPI 模式下, 通过 IO0-IO3 信号传入指令和地址及返回数据。

图 73. QPI 模式下的读取任意寄存器命令序列


注释:

55.A = 地址的 MSB = 23 (CR2V[7] = 0), 或 31 (CR2V[7] = 1)。

11.3.14 写入任何寄存器 (WRAR 71h)

通过写入任意寄存器 (WRAR) 命令, 可以对任意的非易失性或易失性器件寄存器进行写操作。该指令后面是 3 或 4 字节地址 (取决于地址长度配置 CR2V[7]), 接下来是将会写入到选定寄存器内的 1 字节数据。

接受 WRAR 命令前, 器件必须发送写入使能 (WREN) 命令并对其进行解码。这样将设置状态寄存器中的写入使能锁存 (WEL) 位, 用于使能所有写入操作。通过检查 SR1V 中的 WIP 位, 可以确定该操作是否完成。同样, 通过检查 SR1V 中的 P_ERR 和 E_ERR 位可以确定在操作过程中是否发生了错误。

一些寄存器结合使用了各个位类型和规则, 以便控制可以修改的位。有的是只读位, 也有的是 OTP 位。

只读位永远不能被修改, 另外 WRAR 命令中数据字节的相应位被忽略, 并没有设置编程或擦除错误指示 (SR1V 中的 P_ERR 或 E_ERR)。因此, WRAR 数据字节中这些位的值并不重要。

OTP 位只能被编程为其默认状态的相反状态。将 OTP 位返回其默认状态的操作被忽略, 并且不会设置错误。

由 WRAR 数据修改的非易失性位需要非易失性寄存器的写入时间 (t_{w}) 来更新。更新过程包括对非易失性寄存器位进行的擦除和编程操作。如果更新过程的擦除或编程操作失败, 则 SR1V 中的相关错误位和 WIP 位将被设置为 1。

由 WRAR 数据修改的易失性位需要易失性寄存器的写入时间 (t_{cs}) 来更新。

态寄存器 1 被重复读取 (轮询), 用以监控正在写入 (WIP) 位 (SR1V[0]) 和错误位 (SR1V[6,5]), 从而确定寄存器写入操作已经成功完成还是失败。如果发生写入故障, 需要使用清除状态命令来清除错误状态, 使器件返回到待机状态。

但不能通过 WRAR 命令对 PPBL 寄存器进行写操作。只有使用 PPB 锁定写入 (PLBWR) 命令, 才可对 PPBL 寄存器进行写操作。

该命令的序列和行为与 PP 或 4PP 命令相同, 只是该命令能提供一个数据字节。请参见第 102 页上的页编程 (PP 02h 或 4PP 12h)。

寄存器的地址映射如第 89 页上的读取任何寄存器 (RDAR 65h) 所示的内容一样。

11.3.15 设置突发长度 (SBL C0h)

设置突发长度 (SBL) 命令用于配置回卷突发特性。在 SPI 或 QPI 模式下, 可以将回卷突发功能同四线 I/O 读取、DDR 四线 I/O 读取和四线输出读取命令结合起来使用, 用于访问长度和对齐固定的数据。某些应用可以利用该特性来提高整个系统代码执行的性能。通过回卷突发功能, 使用缓存器的应用能够先将关键地址的指令或数据填充到第一个缓存行内, 然后将固定长度 (8/16/32/64 字节) 的数据填充到剩余的缓存行, 而不需要发出多个读命令。

设置突发长度 (SBL) 命令仅通过写入 CR4V 寄存器的位 4、1 和 0 来使能或禁用回卷读取功能, 并设置回卷边界。CR4V 寄存器的其它位不受 SBL 命令的影响。使能回卷读取功能时, 会将相关读取命令从连续读取改变为对一组字节进行连续回卷读取, 直到该命令停止为止。

CR4V[4] = 1 时, 将不使能回卷模式而且可以连续读取无限长度的数据。

CR4V[4] = 0 时, 将使能回卷模式而且只能从读命令提供的字节地址开始读取固定长度和对齐的 8、16、32 或 64 字节组, 并且在该字节组边界上回卷。

字节组的长度和对齐情况都按 8、16、32 或 64 字节的边界实现。CR4V[1:0] 用于选择边界。请参见第 60 页上的易失性配置寄存器 4 (CR4V)。

读取命令的起始地址选择了字节组，并且所返回的第一个数据是地址字节。然后，会连续读取各个字节，直到字节组边界结束为止。如果继续执行读取，则会返回到该组的起始地址，然后再次连续读取数据。该回卷读取序列会继续进行，直到将 **CS#** 返回高电平来停止该命令为止。

表 54. 突发回卷序列示例

SBL 数值 (十六进制)	回卷边界 (字节)	起始地址 (十六进制)	地址序列 (十六进制)
1X	连续	XXXXXX03	03、04、05、06、07、08、09、0A、0B、0C、0D、0E、0F、10、11、12、13、14、15、16、17、18、...
00	8	XXXXXX00	00、01、02、03、04、05、06、07、00、01、02、...
00	8	XXXXXX07	07、00、01、02、03、04、05、06、07、00、01、...
01	16	XXXXXX02	02、03、04、05、06、07、08、09、0A、0B、0C、0D、0E、0F、00、01、02、03、...
01	16	XXXXXX0C	0C、0D、0E、0F、00、01、02、03、04、05、06、07、08、09、0A、0B、0C、0D、0E、...
02	32	XXXXXX0A	0A、0B、0C、0D、0E、0F、10、11、12、13、14、15、16、17、18、19、1A、1B、1C、1D、1E、1F、00、01、02、03、04、05、06、07、08、09、0A、0B、0C、0D、0E、0F、...
02	32	XXXXXX1E	1E、1F、00、01、02、03、04、05、06、07、08、09、0A、0B、0C、0D、0E、0F、10、11、12、13、14、15、16、17、18、19、1A、1B、1C、1D、1E、1F、00、...
03	64	XXXXXX03	03、04、05、06、07、08、09、0A、0B、0C、0D、0E、0F、10、11、12、13、14、15、16、17、18、19、1A、1B、1C、1D、1E、1F、20、21、22、23、24、25、26、27、28、29、2A、2B、2C、2D、2E、2F、30、31、32、33、34、35、36、37、38、39、3A、3B、3C、3D、3E、3F、00、01、02、...
03	64	XXXXXX2E	2E、2F、30、31、32、33、34、35、36、37、38、39、3A、3B、3C、3D、3E、3F、00、01、02、03、04、05、06、07、08、09、0A、0B、0C、0D、0E、0F、10、11、12、13、14、15、16、17、18、19、1A、1B、1C、1D、1E、1F、20、21、22、23、24、25、26、27、28、29、2A、2B、2C、2D、...

通过使用 **WRAR** 命令将 **CR4NV** 编程为所需的值，可以改变上电复位、硬件复位或软件复位的默认突发长度。

图 74. 四线 I/O 模式下的设置突发长度命令序列

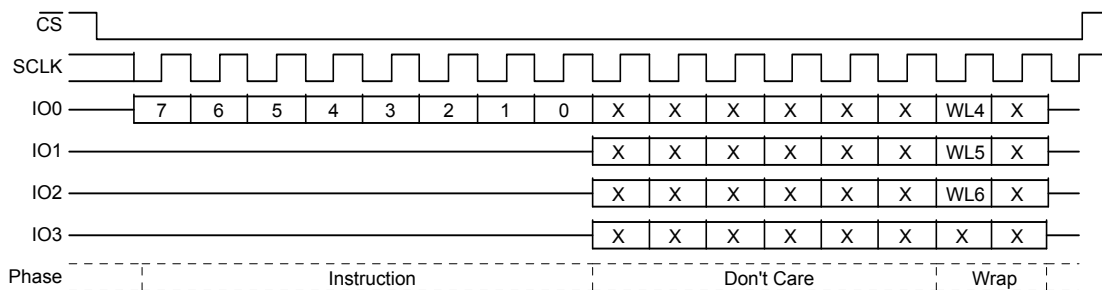
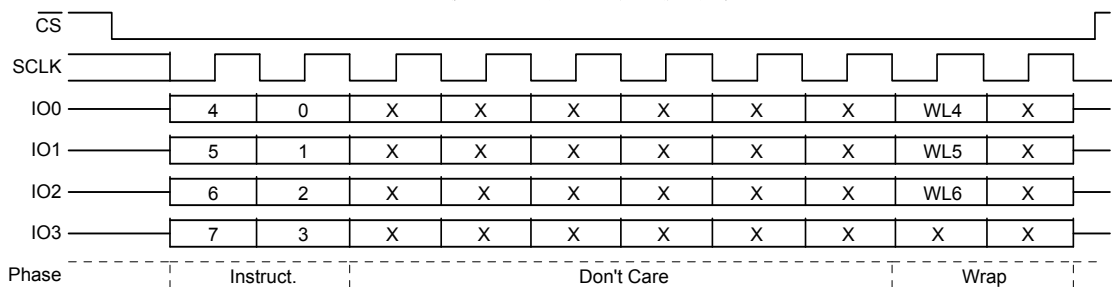


图 75. QPI 模式下的设置突发长度命令序列



11.4 读取闪存阵列命令

闪存主阵列的读取命令允许灵活地与前一代 SPI 相兼容或增大 SPI 性能：

- 某些命令可在每个 SCK 的上升沿上传输地址或数据。它们被称为单倍数据速率（SDR）命令。
- 某些 SDR 命令可在每个 SCK 的上升沿上发送一位地址，并在 SCK 的上升沿上返回一位数据。它们被称为单位宽命令。
- 某些 SDR 命令可在每个 SCK 的上升沿上发送一位地址，并在 SCK 的上升沿上返回 2 位或 4 位数据。它们被称为双线输出（2 比特）和四线输出（4 比特）。
- 某些 SDR 命令则在每个 SCK 的上升沿上发送两位或 4 位的地址和数据，它们被分别称为双线 I/O（2 比特）和四线 I/O 和 QPI（4 比特）。
- 在 QPI 模式下，某些 SDR 命令在每个 SCK 的上升沿上发送 4 个指令、地址和数据位。
- 某些命令可在 SCK 的双边沿上传输地址和数据。它们被称为双倍数据速率（DDR）命令。
- 一些 DDR 命令可在 SCK 的任一边沿上发送 4 比特的地址或数据。它们被称为四线 I/O DDR 和 QPI DDR。

所有这些命令（QPI 读取命令除外）都先发送指令代码，并且每一个 SCK 上升沿传输一位。QPI 读取命令在每个 SCK 的上升沿上发送 4 比特的指令，然后 SDR 或 DDR 会发送 3 或 4 字节的地址。在每个时钟沿上传输双比特或 4 比特地址或数据的命令被称为多路 I/O（MIO）命令。将这些器件配置为能够从主机系统接收 4 字节地址，不像传统的命令那样只能接收 3 字节地址。通过将配置寄存器 2 中的地址长度位设置为 0，可以激活传统命令的 4 字节地址模式。4 字节地址命令中超过 A23 的高端地址位和使用 4 字节地址模式的命令不相关，并被忽略。

四线 I/O 和 QPI 命令通过模式位（该位在发送地址位后被发送）可以提高器件的性能。模式位指出下一个命令的类型是否与当前读命令相同，并且在命令的开始无需发出指令。进行多个四线读取访问时，通过这些模式位可以消除指令周期。

某些命令要求移入地址或模式位后需要添加延迟周期（读取延迟），以便能够访问存储器阵列。读取延迟周期通常被称为虚拟周期。由于存储器忽略虚拟周期，因此主机在这些周期内发送的所有数据也被忽略（“无需关注”），此时主机可以将 SI/I/O1 信号保持为高阻抗状态。使用 MIO 命令时，主机必须在最后虚拟周期结束前停止驱动 I/O 信号（各输出均为高阻抗状态）。使用 DDR 命令时，主机在任意虚拟周期内都不能驱动 I/O 信号。因为由配置寄存器 2（CR2V[3:0]）延迟代码选定的 SCK 频率或性能不同，虚拟周期数量也不一样。虚拟周期指的是从 SCK 的下降沿到下一个 SCK 的下降沿。通常在每个 SCK 的下降沿上将 SPI 输出驱动到新的数值。需要零虚拟周期是因为在同一个 SCK 下降沿上，在存储器将数据返回给主机，同时主机停止发送地址或模式位。

DDR 命令可以在数据的开始前添加一个 8 边沿数据学习模式（DLP），该 DLP 由存储器通过所有数据输出在虚拟周期内驱动的。通过 DLP，主机存储器控制器能够确定 SCK 与数据边沿之间的相移，从而存储器控制器可以在数据边沿的中心捕获数据。

在较高的 SCK 时钟频率（> 50 MHz）下使用 SDR I/O 命令时，应选中提供了一个或多个虚拟周期的 LC，这样可以为主机提供额外的时间，使它在存储器开始发送数据前停止执行发送操作，以避免在 I/O 上发生冲突。如果使用使能了 DLP 的 SDR I/O 命令，应选中能够提供 5 个或多个虚拟周期的 LC，这样可以为主机提供一个周期时长的额外时间，使它在存储器开始发送 4 周期 DLP 前停止执行发送操作。

在数据返回的过程中，一旦 CS# 返回高电平会结束读取命令。在连续读取模式下或在虚拟周期中，CS# 不能在返回数据前变为高电平，否则会导致模式位被错误捕获，从而不能确定器件是否仍处于连续读取模式。

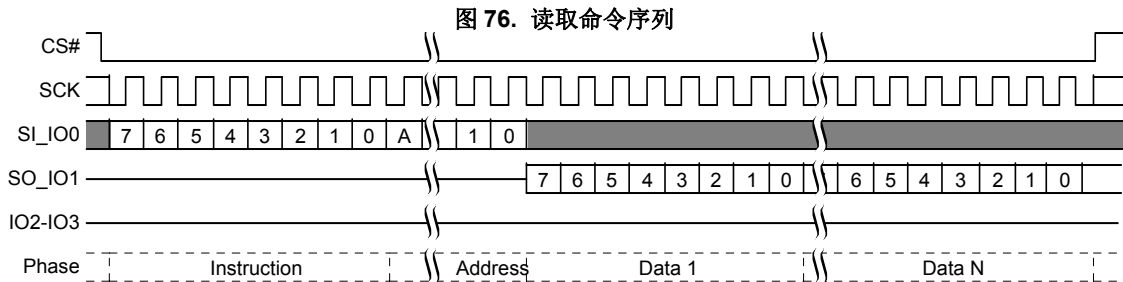
11.4.1 读取（读取 03h 或 4READ 13h）

指令

- 03h (CR2V[7] = 0) 后面是 3 字节地址 (A23-A0)，或
- 03h (CR2V[7] = 1) 后面是 4 字节地址 (A31-A0)，或
- 13h 后面是 4 字节地址 (A31-A0)

指定地址下的存储器内容通过 SO/IO1 被传出。READ 命令的最大工作时钟频率为 50 MHz。

该地址可以从存储器阵列中的任意字节位置开始。输出每一个数据字节后，会按顺序自动递增到下一个更高的地址。因此，只要使用一个读取命令和已给的地址 000000h，便可以读取整个存储器空间。当到达最高地址时，地址计数器翻转到起始地址 000000h，这样允许读取序列继续下去。



注意

56.A = 地址的 MSB = 23 (CR2V[7] = 0)，或 31 (CR2V[7] = 1 或命令 13h)。

11.4.2 快速读取（FAST_READ 0Bh 或 4FAST_READ 0Ch）

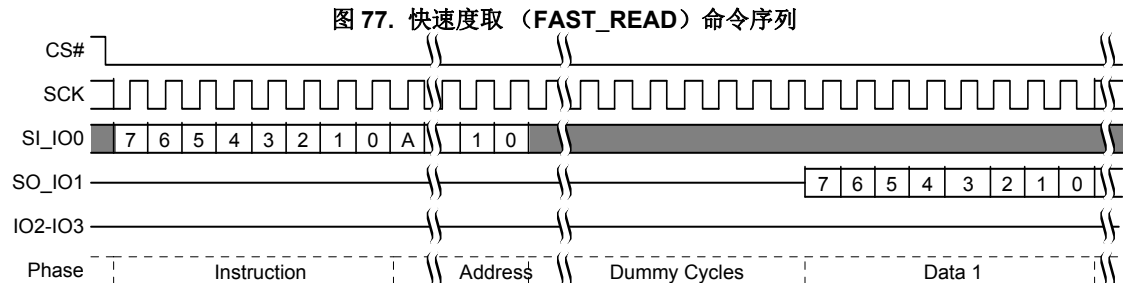
指令

- 0Bh (CR2V[7] = 0) 后面是 3 字节地址 (A23-A0)，或
- 0Bh (CR2V[7] = 1) 后面是 4 字节地址 (A31-A0)，或
- 0Ch 后面是 4 字节地址 (A31-A0)

发送地址后会继续发送虚拟周期，该周期数量取决于配置寄存器 CR2V[3:0] 中设置的延迟值。虚拟周期为器件的内部电路提供额外的时间，使之能够访问重要的地址位置。在这些延迟周期内，SO/IO1 上的所有数据均被忽略，并且该信号会保持为高阻抗状态。此时，存储器通过 SO/IO1 信号发送指定地址的内容。

FAST READ 命令的最大工作时钟频率为 133 MHz。

该地址可以从存储器阵列中的任意字节位置开始。输出每一个数据字节后，会按顺序自动递增到下一个更高的地址。因此，只要使用一个读取命令和已给的地址 000000h，便可以读取整个存储器空间。当到达最高地址时，地址计数器翻转到起始地址 000000h，这样允许读取序列继续下去。



注意

57.A = 地址的 MSB = 23 (CR2V[7] = 0)，或 31 (CR2V[7] = 1 或命令 0Ch)。

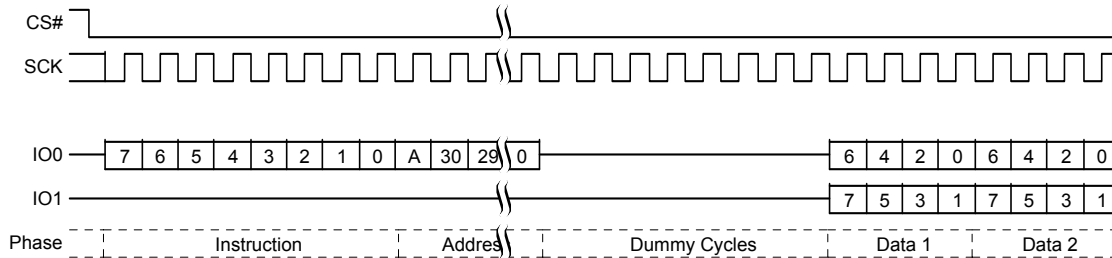
11.4.3 双线输出读取（DOR 3Bh 或 4DOR 3Ch）

指令

- 3Bh（CR2V[0] = 0）后面是 3 字节地址（A23-A0），或
- 3Bh（CR2V[0] = 1）后面是 4 字节地址（A31-A0），或
- 3Ch 后面是 4 字节地址（A31-A0）

发送地址后会提供虚拟周期，该周期数量取决于配置寄存器 CR3V[3:0] 中所设置的延迟值。虚拟周期为器件的内部电路提供额外的时间，使之能够访问重要的地址位置。在这些虚拟周期内，IO0（SI）和 IO1（SO）上的所有数据均被忽略，并会保持为高阻抗状态。此时，存储器通过 IO0（SI）和 IO1（SO）信号每次发送指定地址中的双比特数据。在 SCK 频率下，在 SCK 的下降沿上输出双比特数据。该地址可以从存储器阵列中的任意字节位置开始。输出每一个数据字节后，会按顺序自动递增到下一个更高的地址。因此，只要使用一个读取命令和已给的地址 000000h，便可以读取整个存储器空间。当到达最高地址时，地址计数器翻转到起始地址 000000h，这样允许读取序列继续下去。

图 78. 双线输出读取命令序列



注意

58.A = 地址的 MSB = 23（CR2V[7] = 0），或 31（CR2V[7] = 1 或命令 3Ch）。

11.4.4 双线输出读取（QOR 6Bh 或 4QOR 6Ch）

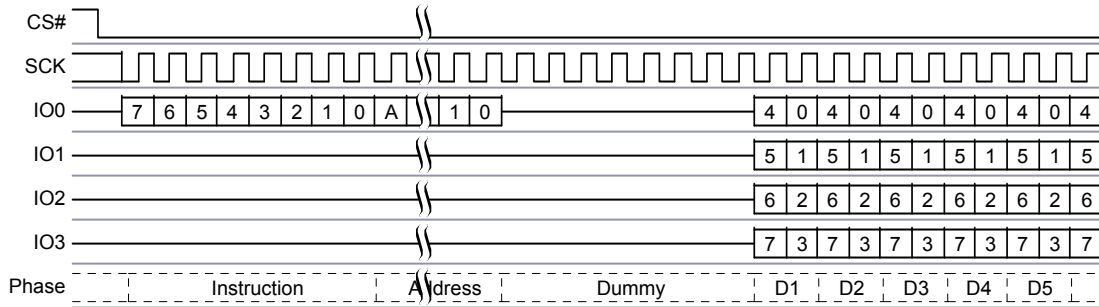
指令

- 6Bh（CR2V[0] = 0）后面是 3 字节地址（A23-A0），或
- 6Bh（CR2V[0] = 1）后面是 4 字节地址（A31-A0），或
- 6Ch 后面是 4 字节地址（A31-A0）

发送地址后会提供虚拟周期，该周期数量取决于配置寄存器 CR3V[3:0] 中所设置的延迟值。虚拟周期为器件的内部电路提供额外的时间，使之能够访问重要的地址位置。在这些虚拟周期内，IO0 - IO3 上的数值被视为“无需关注”，并且保持高阻抗状态。

此时，存储器通过 IO0 - IO3 信号每次发送指定地址中的四位数据。在 SCK 的下降沿上以 SCK 频率输出半字节（4 位）。

该地址可以从存储器阵列中的任意字节位置开始。输出每一个数据字节后，会按顺序自动递增到下一个更高的地址。因此，只要使用一个读取命令和已给的地址 000000h，便可以读取整个存储器空间。当到达最高地址时，地址计数器翻转到起始地址 000000h，这样允许读取序列继续下去。

图 79. 四线输出读命令序列

注释

59.A = 地址的 MSB = 23 (CR2V[7] = 0), 或 31 (CR2V[7] = 1 或命令 6Ch)。

11.4.5 双线 I/O 读取 (DIOR BBh 或 4DIOR BCh)
指令

- BBh (CR2V[7] = 0) 后面是 3 字节地址 (A23-A0), 或
- BBh (CR2V[7] = 1) 后面是 4 字节地址 (A31-A0), 或
- BCh 后面是 4 字节地址 (A31-A0)

双线 I/O 读命令通过两个 I/O 信号, 即 IO0 和 IO1 提高了吞吐量。该命令在每个 SCK 的上升沿上接收地址的输入, 并返回两位读取数据。在某些应用中, 减少地址输入和数据输出的时间会允许立即执行代码 (XIP), 即直接从存储器件执行代码。

双线 I/O 读取命令的最大工作时钟频率为 133 MHz。

双线 I/O 读命令中的连续读取模式位也位于地址后面: 第一个双线读命令发出模式位 Axh 来表示下一个命令也是双线 I/O 读命令之后, 一系列双线 I/O 读命令可以去除 8 位指令。一系列双线 I/O 读取命令中的第一条命令的顺序为: 8 位指令、地址、4 个模式位周期, 最后是可选的延迟周期。如果模式位格式为 Axh, 假设下个命令是不提供指令位的其他双线 I/O 读取命令。该命令的组成部分分别为地址、模式位和可选的延迟。

数据从 IO0-IO1 输出前, 在模式位被传入 IO0 和 IO1 后需要添加可变的延迟周期。这样, 器件的内部电路会有足够的时间来访问初始地址中的数据。在这些虚拟周期内, IO0 和 IO1 上的数值被视为 “无需关注”, 并且保持高阻抗状态。虚拟周期的数量由 SCK 的频率确定。该延迟在 CR2V[3:0] 中指定。

凭借连续读取特性, 在读访问的序列中无需读取指令位, 并能够明显提高代码执行 (XIP) 的性能。模式位的高位半字节 (位 7-4) 通过添加或清除第一个字节指令代码来控制下一个双线 I/O 读取命令的长度。模式位的低位半字节 (位 3-0) 则被忽略 (使用 “x” 表示), 并保持为高阻抗状态。如果模式位为 Axh, 则器件仍处于双线 I/O 连续读取模式, 并且在 CS# 被驱动为高电平, 再被确认为低电平后能够移入下一个地址 (而不需要读取 BBh 或 BCh 指令 (如第 97 页上的图 80 所示)); 这样可以消除命令序列中的 8 个周期。发生以下情况时会使器件退出

双线 I/O 连续读取模式: 然后器件可以接收标准的 SPI 命令:

1. 在执行双线 I/O 连续读取命令序列过程中, 如果模式位为非 Axh 值, 则下次将 CS# 驱动为高电平时会使器件退出双线 I/O 连续读取模式。
2. 发送模式复位命令。

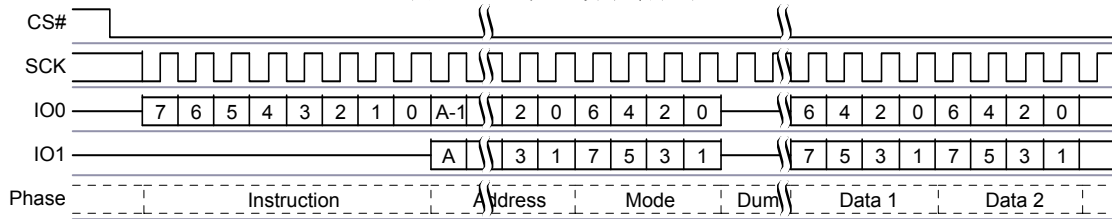
请注意, 最后的读取地址周期被定时在 IO0-IO1 后, 器件的内部电路需要 4 个模式位周期的延迟时间来访问初始地址。

在第一个数据移出的时钟的下降沿上 (或之前), 必须将 I/O 信号设为高阻抗状态。在更高的时钟速度下, 存储器件开始发送数据前需要停止主机传输数据的时间 (即总线周转时间) 被减少。这样, 主机系统可以在最后两个 “无需关注” 的模式周期中或在任意虚拟周期内停止通过 I/O 信号发送数据 (将它们保持为高阻抗状态), 从而避免在 I/O 信号上发生冲突。

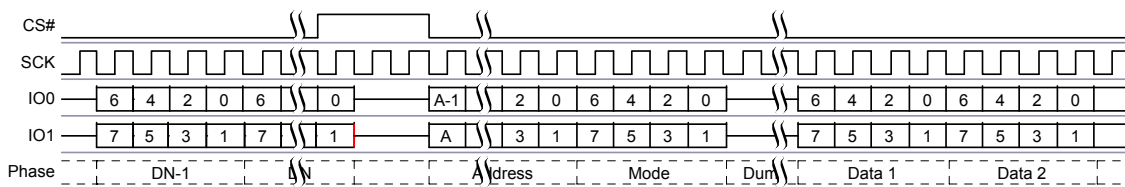
该延迟周期结束后, 存储器便通过 IO0-IO1 信号每次发送指定地址中的双比特数据。以 SCK 频率, 在 SCK 的下降沿上将输出双比特数据。

该地址可以从存储器阵列中的任意字节位置开始。输出每一个数据字节后, 会按顺序自动递增到下一个更高的地址。因此, 只要使用一个读取命令和已给的地址 000000h, 便可以读取整个存储器空间。当到达最高地址时, 地址计数器翻转到起始地址 000000h, 这样允许读取序列继续下去。

在模式 / 虚拟位周期内不应该将 CS# 驱动为高电平, 否则会使这些模式位的状态不确定。

图 80. 双线 I/O 读命令序列

注意：

60.4 个最低有效模式位均被忽略，主机可以选择发送这些位。主机在这些周期内会停止发送数据，以便延长从主机发送模式位到存储器返回数据之间总线周转的时间。
 61.A = 地址的 MSB = 23 (CR2V[7] = 0)，或 31 (CR2V[7] = 1 或命令 BBh)。

图 81. 双线 I/O 连续读取命令序列

注意

62.4 个最低有效模式位均被忽略，主机可以选择发送这些位。主机在这些周期内会停止发送数据，以便延长从主机发送模式位到存储器返回数据之间总线周转的时间。
 63.A = 地址的 MSB = 23 (CR2V[7] = 0)，或 31 (CR2V[7] = 1 或命令 BBh)。

11.4.6 四线 I/O 读取 (QIOR EBh 或 4QIOR ECh)

指令

- EBh (CR2V[7] = 0) 后面是 3 字节地址 (A23-A0)，或
- EBh (CR2V[7] = 1) 后面是 4 字节地址 (A31-A0)，或
- ECh 后面是 4 字节地址 (A31-A0)

由于四线 I/O 读取命令使用了四个 I/O 信号 (IO0-IO3)，因而提高了吞吐量。这样，可以在每个串行时钟 SCK 的边沿上移入 4 位地址。在某些应用中，降低指令的开销，可以直接从 FS-S 系列器件执行代码 (XIP)。必须设置配置寄存器的 QUAD 位 (CR1V[1] = 1)，以使能 FS-S 系列器件的四线能力。

四线 I/O 读取命令的最大工作时钟频率为 133 MHz。

执行四线 I/O 读取命令时，移入模式位后 (如下所述) 到通过 IO0-IO3 开始移出数据前需要添加一个延迟周期。该延迟周期 (虚拟周期) 为器件的内部电路提供了所需时间，用于访问初始地址的数据。在这些延迟周期内，IO0-IO3 上的数据值为“无需关注”项，该值可能为高阻状态。虚拟周期的数量由 SCK 的频率确定。该延迟在 CR2V[3:0] 中指定。

该延迟周期结束后，存储器便通过 IO0-IO3 信号每次发送指定地址中的四位数据。在 SCK 信号的下降沿上以 SCK 频率移出半字节 (4 位)。

该地址可以从存储器阵列中的任意字节位置开始。输出每一个数据字节后，会按顺序自动递增到下一个更高的地址。因此，只要使用一个读取命令和已给的地址 000000h，便可以读取整个存储器空间。当到达最高地址时，地址计数器翻转到起始地址 000000h，这样可使读取序列继续下去。

不需要添加四线 I/O 读指令仍能跳到其他地址。通过设置地址序列后的模式位 (如图 82 所示)，可以控制该操作。凭借这个附加的特性，无需读取指令序列，这样可明显提高代码执行 (XIP) 的能力。模式位的高位半字节 (位 7-4) 通过添加或清除第一个字节指令代码来控制下一个四线 I/O 指令的长度。模式位的低位半字节 (位 3-0) 被忽略 (使用“x”表示)。如果模式位为 Axh，则器件仍处于四线 I/O 高性能读取模式，并且在 CS# 被驱动为高电平，然后再被确认为低电平后，能够移入下一个地址 (不需要读取 EBh 或 ECh 指令 (如第 99 页上的图 84 所示))。这样可以消除命令序列的 8 个周期。发生以下情况时会使器件退出四线 I/O 高性能读取模式；然后器件可以接收标准的 SPI 命令：

1. 在执行四线 I/O 读取命令序列过程中，如果模式位为非 Axh 的值，则下次将 CS# 驱动为高电平时会使器件退出四线 I/O 高性能读取模式。
2. 发送模式复位命令。

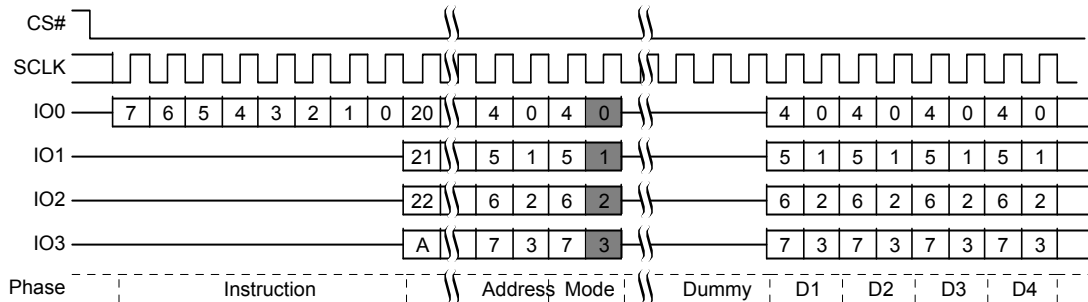
请注意，最后的读取地址周期被定时在 $IIO0-I03$ 后，器件的内部电路需要两个模式位时钟周期和额外的等待时间（即虚拟周期）来访问初始地址。

应注意，在第一个数据移出的时钟的下降沿上（或之前），必须将 $IO0-I03$ 信号设为高阻抗。在更高的时钟速度下，存储器件开始发送数据前需要停止主机传输数据的时间（即总线周转时间）被减少。这样，主机系统可以在最后“无需关注”的模式周期内或在任意虚拟周期内停止通过 $IO0-I03$ 信号发送数据（将它们保持为高阻抗），从而避免在 $IO0-I03$ 信号上发生冲突。

在模式 / 虚拟位周期内不应该将 $CS\#$ 驱动为高电平，否则会使这些模式位的状态不确定。

在 QPI 模式下（ $CR2V[6] = 1$ ），四线 I/O 指令都会在每个 $SCLK$ 的上升沿上发送 4 比特信息。该命令协议的其余部分和四线 I/O 命令完全相同。

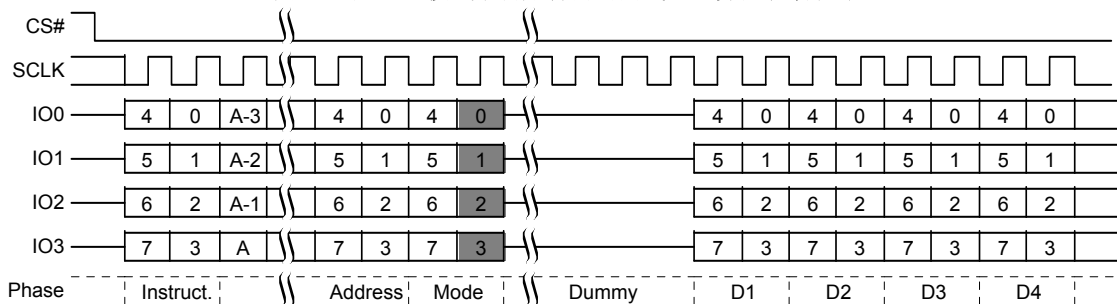
图 82. 初始访问的四线 I/O 读取命令序列



注意

64.A = 地址的 MSB = 23（ $CR2V[7] = 0$ ），或 31（ $CR2V[7] = 1$ 或命令 ECh）。

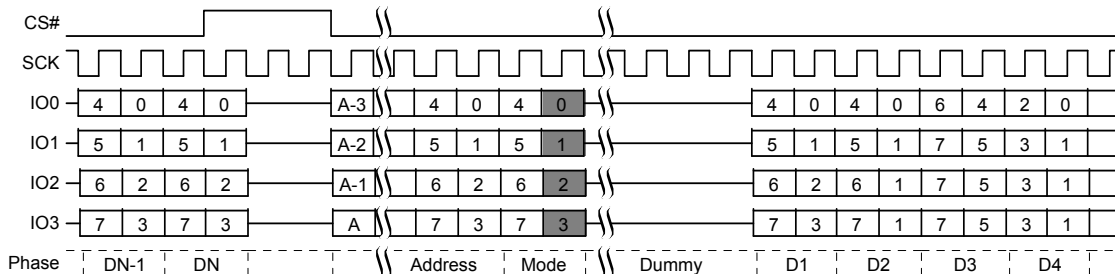
图 83. 在 QPI 模式下初始访问的四线 I/O 读取命令序列



注意

65.A = 地址的 MSB = 23（ $CR2V[7] = 0$ ），或 31（ $CR2V[7] = 1$ 或命令 ECh）。

图 84. 四线 I/O 连续读取命令序列


注意

66.A = 地址的 MSB = 23 (CR2V[7] = 0), 或 31 (CR2V[7] = 1 或命令 ECh)。

67.在 QPI 模式下, 命令序列也相同。

11.4.7 DDR 四线 I/O 读取 (EDh、EEh)

DDR 四线 I/O 读取命令因为使用了四个 I/O 信号 (IO0-IO3), 从而提高了吞吐量。它和四线 I/O 读取命令相同, 但仍允许在每个时钟沿上移入 4 比特地址。在某些应用中, 降低指令的开销, 可以直接从 FS-S 系列器件执行代码 (XIP)。必须设置配置寄存器的 QUAD 位 (CR1V[1] = 1), 以使能器件的四线能力。

指令

- EDh (CR2V[7] = 0) 后面是 3 字节地址 (A23-A0), 或
- EDh (CR2V[7] = 1) 后面是 4 字节地址 (A31-A0), 或
- EEh 后面是 4 字节地址 (A31-A0)

先发送地址, 然后发送模式位。在 DDR 模式下, 存储器通过 IO0 - IO3 信号每次在一个时钟沿上发送特定地址中的四比特数据。

DDR 四线 I/O 读取命令的最大工作时钟频率为 100 MHz。

执行 DDR 四线 I/O 读取命令时, 通过 IO0-IO3 移入最后地址和模式位后到数据从 IO0-IO3 开始移出前, 需要添加一个延迟周期。这样, 器件的内部电路会有足够的时间来访问初始地址的数据。在这些延迟周期内, IO0-IO3 上的数据值均被忽略, 并且这些信号会保持为高阻抗。使能数据学习模式 (DLP) 时, 主机不能在虚拟周期内驱动 IO 信号, 而必须将它保持为高阻抗状态, 以便存储器件能够驱动 DLP。

虚拟周期的数量由 SCK 的频率确定。该延迟在 CR2V[3:0] 中指定。

通过模式位, 一系列 I/O DDR 命令可以在第一条命令发送一个互补模式位后消除 8 位指令, 如图 85 和图 87 中所示。由于具备该特性, 因此不需要 8 位 SDR 指令, 并且能够大量减少最初的访问时间 (提高 XIP 性能)。模式位通过添加或清除第一个字节指令代码来控制下一个 DDR 四线 I/O 读取操作的长度。如果模式位的高位半字节 (IO[7:4]) 和低位半字节 (IO[3:0]) 是互补的 (即 5h 和 Ah), 那么器件会切换为 DDR 四线 I/O 连续读取模式, 并且能够在 CS# 被驱动为高电平再被确认为低电平后读取下一个地址 (不需要读取 EDh 或 EEh 指令), 从而可以清除命令序列的 8 个周期。发生以下情况时会使器件退出 DDR 四线 I/O 连续读取模式; 然后器件可以接收标准的 SPI 命令:

1. 在执行 DDR 四线 I/O 读取命令序列过程中, 如果模式位不是互补的, 则下次 CS# 被驱动为高电平然后再被确认为低电平时会使器件退出 DDR 四线 I/O 读取模式。
2. 发送模式复位命令。

该地址可以从存储器阵列中的任意字节位置开始。输出每一个数据字节后, 会按顺序自动递增到下一个更高的地址。因此, 只要使用一个读取命令和已给的地址 000000h, 便可以读取整个存储器空间。当到达最高地址时, 地址计数器翻转到起始地址 000000h, 这样允许读取序列继续下去。

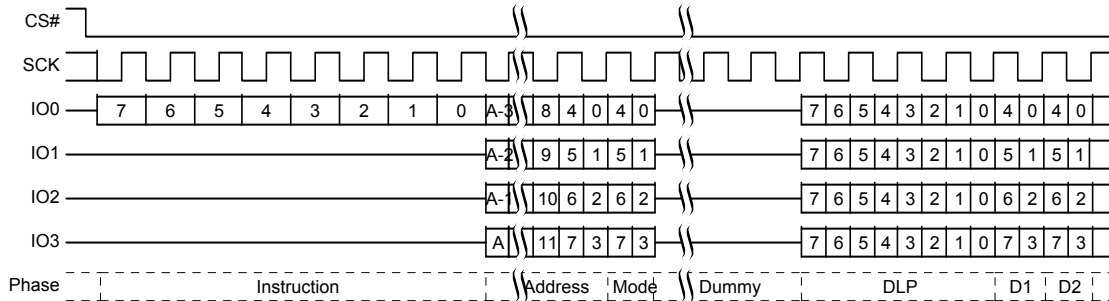
在模式 / 虚拟位周期内不应该将 **CS#** 驱动为高电平，否则会使这些模式位的状态不确定。请注意，存储器件在发送第一个数据值前先在 **IO** 信号上发送一个前导码。前导码是一个数据学习模式（**DLP**），主机控制器通过它使数据捕获在更高的频率下得到优化。数据被输出前，前导码会立即驱动 **IO** 总线上四个时钟周期。请确保主机在存储器开始发送前导码前已经停止驱动 **IO** 总线。

前导码用于向主机控制器表示从主机驱动一个时钟沿到存储器件返回相应数据的完整时间。主机控制器会在前导码期间使数据捕获点偏移，以便优化时序容限，然后使用同样的偏移时间捕获读取操作其余部分的数据。在所有读取操作的前导码期间，都可确定得到优化的数据捕获点。该优化特性不但能补偿存储器件和主机控制器的 **PVT**（处理过程、电压、温度），还可以补偿 **PCB** 板上数据传送时间所导致的系统级延迟。

虽然数据学习模式（**DLP**）是可编程的，但以下显示的是 **DLP** 为 34h 的示例。**DLP 34h**（或 00110100）是在每个活跃的输出（即所有四个 **IO**）上被发送的。选中该模式以包含“**DC**”和“**AC**”的数据切换情况。两个 **DC** 切换情况包括：数据在较长时间（2.5 个时钟周期）内为低电平，然后切换为高电平（001）；互补低电平切换（110）。两个 **AC** 切换情况包括：数据在较短时间（1.5 个时钟周期）内为低电平，然后切换为高电平（101）；互补低电平切换（010）。与 **AC** 切换相比，直流切换通常在更接近于电源轨的起始点上发生。在该起始点上，**AC** 切换的电源未达到稳定状态。在多种情况下，直流切换决定了数据有效期间的开始，交流切换则决定了数据有效期间的结束。通过这些切换，主机控制器可以确定数据有效眼图的开始和结束。一旦确认了数据眼图，便可以选中最佳的数据捕获点。更多信息，请参考第 63 页上的 **SPI DDR 数据学习寄存器** 中介绍的内容。

在 **QPI** 模式下（**CR2V[6] = 1**），**DDR** 四线 **I/O** 指令都会在每个 **SCK** 的上升沿上发送 4 比特信息。该命令协议的其余部分和 **DDR** 四路 **I/O** 命令完全相同。

图 85. **DDR** 初始访问的四路 **I/O** 读取命令序列

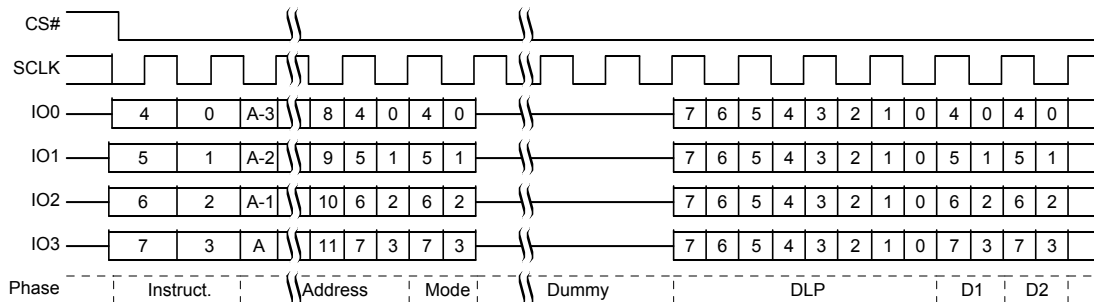


注意

68.A = 地址的 MSB = 23（**CR2V[7] = 0**），或 31（**CR2V[7] = 1** 或命令 **EEh**）。

69.DLP 为 34h（或 00110100）的示例。

图 86. 在 **QPI** 模式下初始访问的 **DDR** 四线 **I/O** 读取命令序列

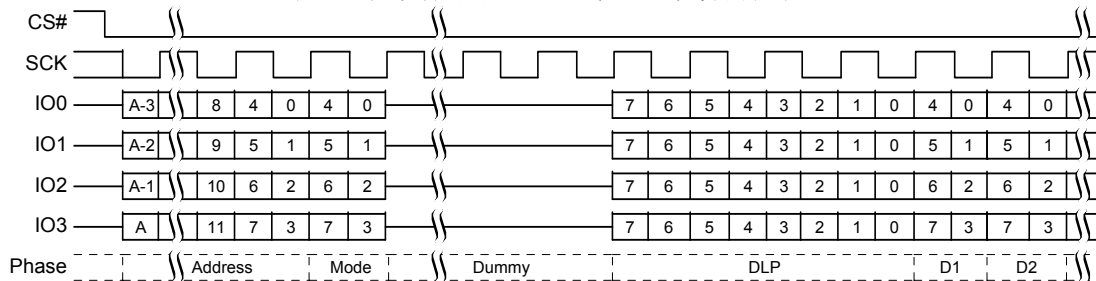


注意

70.A = 地址的 MSB = 23（**CR2V[7] = 0**），或 31（**CR2V[7] = 1** 或命令 **EEh**）。

71.DLP 为 34h（或 00110100）的示例。

图 87. 后续访问的 DDR 四线 I/O 连续读取序列


注意

72.A = 地址的 MSB = 23 (CR2V[7] = 0), 或 31 (CR2V[7] = 1 或命令 EEh)。

73.在 QPI 模式下, 命令序列也相同。

74.DLP 为 34h (或 00110100) 的示例。

11.5 编程闪存阵列命令

11.5.1 编程粒度

11.5.1.1 自动纠错码 (ECC)

每 16 个字节被对齐, 16 个字节长度的编程块具有一个自动纠错码 (ECC) 值。带 ECC 的数据块形成了一个 ECC 单元。与错误检测和纠正 (EDC) 逻辑结合, ECC 用于检测和纠正读访问期间发生的任何单比特错误。任意数据量先被编程到一个 ECC 单元内时, 便设置整个 ECC 单元的 ECC 值。如果对同一 ECC 单元进行多次编程, 则更改 ECC 值以禁用 EDC 功能。需要完成一个扇区擦除操作才能重新启用编程块上的自动 ECC。16 字节编程块是可启用自动 ECC 的最小编程粒度。

这些自动操作对于用户是透明的。自动 ECC 功能的透明性增强了典型编程操作 (将数据一次性写入到每个 ECC 单元内) 中数据的准确性, 同时仍支持单字节编程和位编程, 可以在同一个 ECC 单元中进行多次编程, 从而实现软件与 FL 系列的前代产品向后兼容。当 ECC 单元中禁用了自动 ECC 功能时, 则不能对 ECC 单元区域读取的数据进行 EDC 操作。

ECC 状态寄存器用于确定 ECC 单元上是否使能了 ECC, 同时是否在 ECC 单元数据或 ECC 中检测并纠正了任何错误 (请参见第 61 页上的 [ECC 状态寄存器 \(ECCSR\)](#))。ECC 状态寄存器读取 (ECCRD) 命令用于读取任何 ECC 单元上的 ECC 状态 (请参见第 87 页上的 [ECC 状态寄存器读取 \(ECCRD 19h 或 4EECRD 18h\)](#))。

错误检测和纠正 (EDC) 应用于寄存器以外的所有闪存地址空间部分。针对每个受保护的字节组计算纠错码 (ECC), 并将 ECC 存储在与该字节组相关的隐藏区域中。受保护的字节组和相关 ECC 一起被称为 ECC 单元。

- 针对每 16 字节对齐和长度 ECC 单元进行计算 ECC
- 单比特 EDC 支持 8 ECC 位 /ECC 单元, 以及一个 ECC 禁用标志位
- 扇区擦除将扇区中的所有 ECC 位和 ECC 禁用标志复位为默认状态 (即使能状态)
- ECC 被编程为标准程序命令操作的一部分
- 如果在同一 ECC 单元上执行多个编程操作, 则会自动禁用 ECC。
- 允许单字节编程或位编程, 但在对同一个 16 字节 ECC 单元的第二个程序上禁用 ECC。
- 禁用 ECC 时, 则会编程 ECC 禁用标志
- 要为已禁用的 ECC 单元重新启用 ECC, 必须擦除包含 ECC 单元的扇区
- 为了确保 EDC 提供的最佳数据完整性, 每个 ECC 单元只应编程一次, 以便将 ECC 存储在该单元中, 并且不禁用。
- ECC 的计算、编程和禁用是自动完成的, 并作为编程操作的一部分。如果需要, 检测和纠正也是自动完成的, 并作为读取操作的一部分。主机系统只查看读取操作中已纠正的数据。
- ECC 将保护 OTP 区域。但是, 在同一 ECC 单元上执行的第二个程序操作将永久禁用该单元上的 ECC (OTP 是一次性可编程的, 因此将禁止重新启用 ECC 使能 / 指示位的擦除操作)

11.5.1.2 页编程

通过将需要编程的数据加载到页缓冲区，并发送编程命令将缓冲区中的数据传送到存储器阵列内，这样可以实现页编程操作。该操作设置了通过单个编程命令可以编程的数据量的上限。页编程在一个操作中最多可编程全页（256 或 512 字节）。页大小由配置寄存器位 CR3V[4] 指定。该页在页大小地址边界上对齐。通过每个页编程操作可以编程一位到一页的容量。建议按长度为 16 字节的倍数对齐模块进行写入操作。这样可以防止禁用 ECC 功能。为达到最佳性能，应在 512 字节边界上一次完成整页 512 字节对齐的编程操作，并且每一页仅被编程一次。

11.5.1.3 单字节编程

单字节编程允许在存储器阵列中的任何位置内编程单个字节，因此该操作与传统标准 SPI 页编程（PP）命令完全向后兼容。虽然支持单字节编程，但如果在同一个 ECC 单元上编程另一个字节，则将禁用 16 字节 ECC 单元上的自动 ECC。

11.5.2 页编程（PP 02h 或 4PP 12h）

通过页编程（PP）命令，可在存储器中进行字节编程（将位从 1 改成 0）。接收页编程（PP）命令前，器件必须发送写入使能（WREN）命令，并对其进行解码。成功解码写入使能（WREN）命令后，器件将设置状态寄存器中的写入使能锁存（WEL）位，以使能任意写入操作。

指令

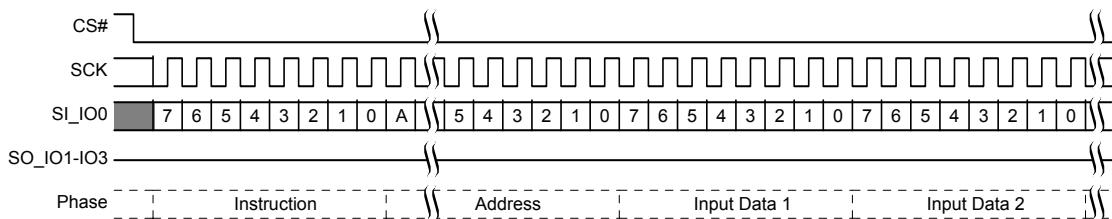
- 02h（CR2V[7] = 0）后面是 3 字节地址（A23-A0），或
- 02h（CR2V[7] = 1）后面是 4 字节地址（A31-A0），或
- 12h 后面是 4 字节地址（A31-A0）

并且 SI/IO0 信号上至少有一个数据字节。根据 CR3V[4]，页大小可以为 256 或 512 字节。通过传送指令 02h，然后传送 3 字节地址或通过传送指令 12h，然后传送 4 字节地址后，可在 SI/IO0 提供多达一页的数据。如果传送到器件的数据超过起始地址和页对齐结束边界所组成的空间，则数据加载序列将从页中的最后字节回卷到同一页的 0 字节位置，并开始覆盖已经写入到页中的数据。最后一页的数据在该页中被编程。这是由于器件的页编程缓冲器只有页大小的长度。如果传送到器件的数据小于一页，则这些数据字节会按顺序被编程，从页中所提供的地址开始，并不会影响同一页中的其它字节。

通过页编程（PP）命令加载页边界内的整页，将节省整个编程时间，并且能缩短将一页加载到编程缓冲器内的时间。

编程过程由闪存存储器器件的内部控制逻辑管理。发送编程命令后，可通过读取状态寄存器 1 命令检查编程操作状态。WIP 位（SR1V[0]）指出了编程操作是否完成。P_ERR 位（SR1V[6]）将指出该编程操作是否发生了错误，因此无法成功完成。它包括尝试对保护区进行编程的错误。

图 88. 页编程（PP 02h 或 4PP 12H）命令序列

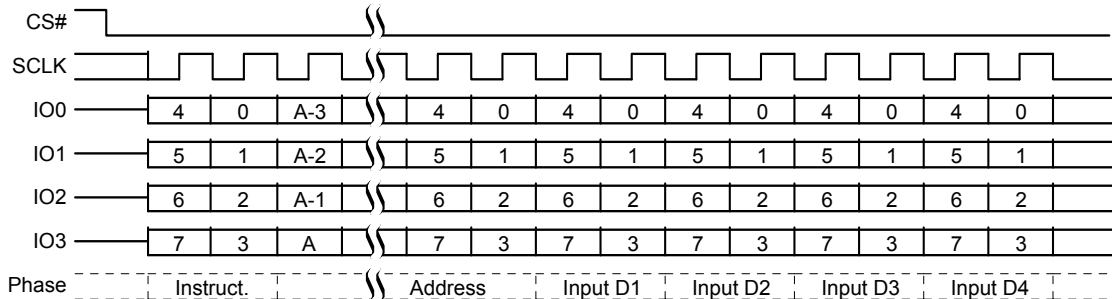


注意

75A = 地址的 MSB = A23（PP 02h，CR2V[7] = 0），或 = A31（PP 02h，CR2V[7] = 1，或 4PP 12h 命令）。

该命令也支持 QPI 模式。在 QPI 模式下，指令、地址和数据将通过 IO0-IO3 信号传入到芯片内。

图 89. QPI 模式下的页编程（PP 02h 或 4PP 12H）命令序列



注意：

76.A = 地址的 MSB = A23（PP 02h，CR2V[7] = 0），或 = A31（PP 02h，CR2V[7] = 1，或 4PP 12h 命令）。

11.5.3 四线页编程（QPP 32h 或 4QPP 34h）

通过四线输入页编程（QPP）命令，可在存储器中进行字节编程（将各个位值从 1 改成 0）。四线输入页编程（QPP）命令允许通过使用四个信号（IO0-IO3）将一个数据页加载到页缓冲区内。通过在每个时钟周期内加载 4 位数据，QPP 可以提高具有更慢时钟速度（< 12 MHz）的 PROM 编程器和各个应用的性能。因为内在的页编程时间大于时钟数据所需的时间，所以具有更高时钟频率的系统体现不了 QPP 命令所带来的好处。QPP 命令的最大频率为 133 MHz。

要想使用四线页编程命令，必须设置配置寄存器中的四线使能位（QUAD = 1）。接收 QPP 命令之前，器件要执行写入使能命令（状态寄存器 1 的 WEL = 1）。

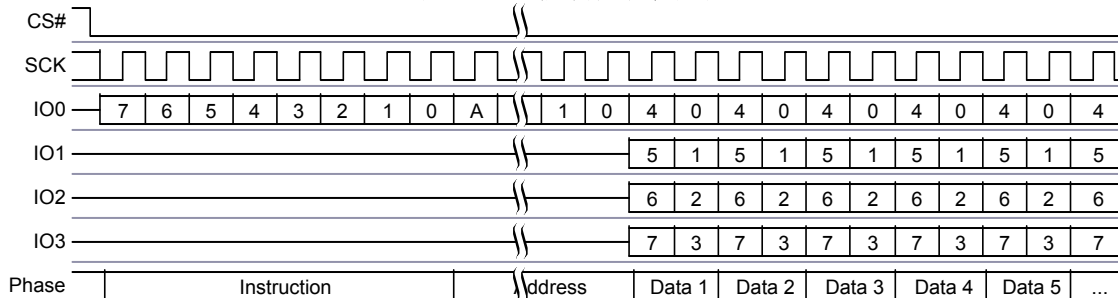
指令

- 32h（CR2V[0] = 0）后面是 3 字节地址（A23-A0），或
- 32h（CR2V[0] = 1）后面是 4 字节地址（A31-A0），或
- 34h 后面是 4 字节地址（A31-A0）

并且在 IO 信号上，至少有一个数据字节。

QPP 的所有其它功能与页编程的相同。QPP 命令序列如下面图所示。

图 90. 四线页编程命令序列



注意

77.A = 地址的 MSB = A23（QPP 32h，CR2V[7] = 0）或 = A31（QPP 32h，CR2V[7] = 1，或 4QPP 34h 命令）

11.6 擦除闪存阵列命令

11.6.1 参数扇区擦除 (P4E 20h 或 4P4E 21h)

可以对主闪存阵列地址映射进行配置，从而使参数扇区覆盖掉最低地址相同扇区中地址最低的部分（底部参数扇区），或覆盖掉最高地址相同扇区中地址最高的部分（顶部参数扇区）。也可以配置主闪存阵列地址的映射，使它只有尺寸相同的扇区。参数扇区的配置由配置位 CR3V[3] 控制。器件被配置为具有尺寸相同的扇区时（CR3V[3]=1），P4E 和 4P4E 命令被忽略。

参数扇区擦除命令将参数扇区中的所有位设置为 1（所有字节均为 FFh）。接受 P4E 或 4P4E 命令前，器件必须发送写入使能（WREN）命令并对其进行解码。这样将设置状态寄存器中的写入使能锁存（WEL）位，用于使能所有写入操作。

指令

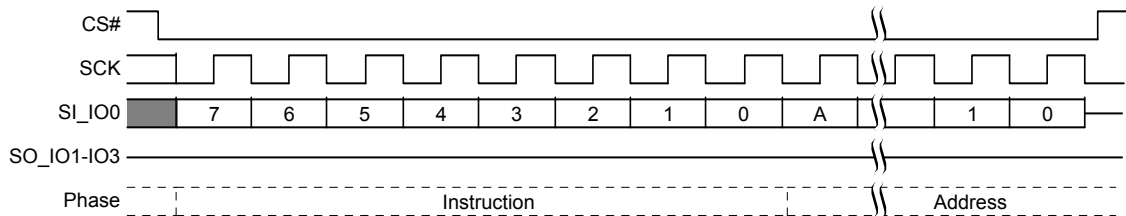
- 20h[CR2V[7] = 0] 后面是 3 字节地址（A23-A0），或
- 20h[CR2V[7] = 1] 后面是 4 字节地址（A31-A0），或
- 21h 后面是 4 字节地址（A31-A0）

地址的第 24 或第 32 位通过 SI/IO0 被锁存在芯片内后，必须将 CS# 信号驱动为逻辑高电平状态。该操作将启动内部擦除周期，包括预编程和擦除闪存存储器阵列的已选扇区。如果锁存地址的最后一位之后未将 CS# 驱动为高电平，则不会执行扇区擦除操作。

将 CS# 信号驱动为高电平后，将启动内部擦除周期。在内部擦除周期中，用户可以读取正在进行写入操作（WIP）位的值，从而确定该操作是否完成。在擦除周期期间，WIP 位将为“1”；在擦除周期完成后，该位显示为“0”。

对于通过模块保护位或 ASP 受保护的扇区，P4E 或 4P4E 命令不被执行，并且 E_ERR 状态被设置。不会执行使用于大小超过 4 KB 的扇区的 P4E 命令，并且不会报告错误（E_ERR）状态。

图 91. 参数扇区擦除 (P4E 20h 或 4P4E 21h) 命令序列

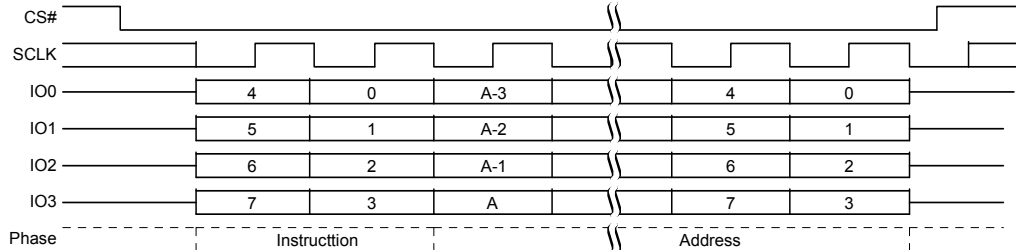


注意

78.A = 地址的 MSB = A23（使用命令 SE 20h，CR2V[7] = 0），或 = A31（使用命令 SE 20h 和 CR2V[7] = 1，或命令 4SE 21h）。

该命令也支持 QPI 模式。在该模式下，指令、地址和数据通过 IO0-IO3 信号被传入到芯片内。

图 92. QPI 模式下的参数扇区擦除 (P4E 20h 或 4P4E 21h) 命令序列



注意

79.A = 地址的 MSB = A23（使用命令 SE 20h，CR2V[7] = 0），或 = A31（使用命令 SE 20h 和 CR2V[7] = 1，或命令 4SE 21h）。

11.6.2 扇区擦除（SE D8h 或 4SE DCh）

扇区擦除（SE）命令将地址扇区中的所有位设置为 1（这样所有字节均为 FFh）。接收扇区擦除（SE）命令前，器件必须发送写入使能（WREN）命令并对其进行解码。这样将设置状态寄存器中的写入使能锁存（WEL）位，以使能所有写入操作。

指令

- D8h [CR2V[7] = 0] 后面是 3 字节地址（A23-A0），或
- D8h [CR2V[7] = 1] 后面是 4 字节地址（A31-A0），或
- DCh 后面是 4 字节地址（A31-A0）

地址的第 24 或第 32 位通过 SI 锁存在芯片内后，必须将 CS# 信号驱动为逻辑高电平状态。该操作将启动擦除周期，包括所选扇区的预编程和擦除操作。如果锁存地址的最后一位后未将 CS# 驱动为高电平，则不会执行扇区擦除操作。

将 CS# 信号驱动为逻辑高电平状态后，将启动内部擦除周期。在内部擦除周期中，用户可以读取正在写入（WIP）位的值，从而检查该操作是否完成。当器件处于擦除周期过程中时，WIP 位将显示为 1；当擦除周期完成后，WIP 则显示为 0。

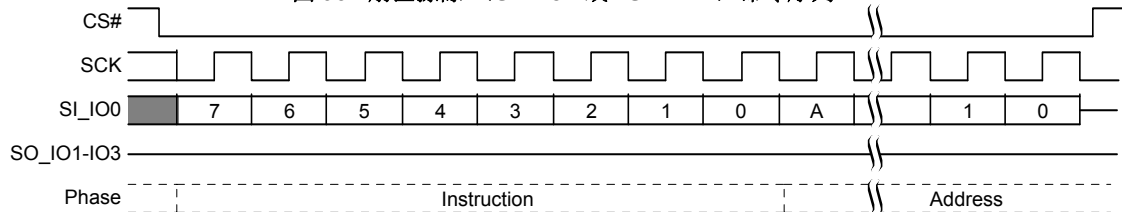
对于通过模块保护位或 ASP 受保护的扇区，扇区擦除（SE）命令不被执行，并且 E_ERR 状态被设置。

器件配置项 CR3V[1] 用于指出 SE 命令擦除的区域是 64 KB 还是 256 KB。

器件配置项 CR3V[3] 用于确定当前是否使用 4 KB 的参数扇区。CR3V[3] = 0 时，参数扇区将覆盖器件地址空间中的最高或最低 32 KB 部分。如果将扇区擦除命令使用于被参数扇区覆盖的 64 KB 扇区，则覆盖的参数扇区不受擦除命令的影响。只有 64 KB 扇区的可见（无覆盖）部分被擦除。类似的，如果将扇区擦除命令使用于被参数扇区覆盖的 256 KB 扇区，覆盖的参数扇区则不受擦除命令的影响。CR3V[3] = 1 时，在器件的地址空间中没有参数扇区，并且扇区擦除命令一直对全部可见的 64 KB 或 256 KB 扇区进行操作。

ASP 为每个扇区（包括参数扇区）提供了一个 PPB 和一个 DYB 保护位。如果将扇区擦除命令使用于包括 64 KB 受保护扇区的 256 KB 范围，该擦除命令不被执行，并且将设置 E_ERR 状态。

图 93. 扇区擦除（SE D8h 或 4SE DCh）命令序列

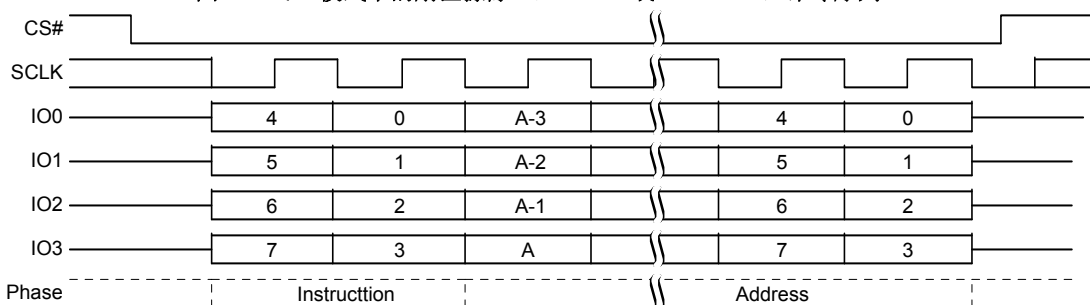


注意

80.A = 地址的 MSB = A23（SE D8h，CR2V[7] = 0），或 = A31（SE D8h，CR2V[7] = 1 或 4SE DCh）。

该命令也支持 QPI 模式。在该模式下，指令、地址和数据通过 IO0-IO3 信号被传入到芯片内。

图 94. QPI 模式下的扇区擦除（SE D8h 或 4SE DCh）命令序列



注意

81.A = 地址的 MSB = A23（SE D8h，CR2V[7] = 0），或 = A31（SE D8h，CR2V[7] = 1 或 4SE DCh）。

11.6.3 批量擦除（BE 60h 或 C7h）

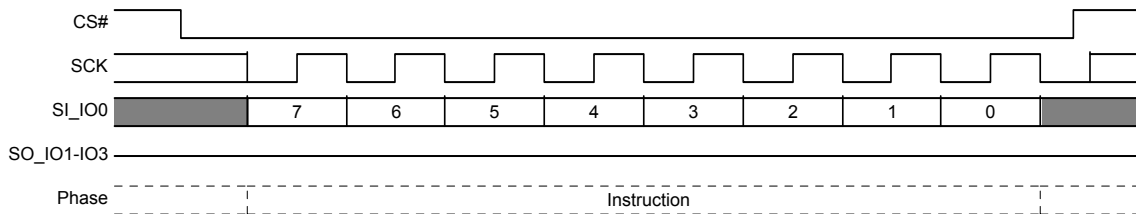
批量擦除（BE）命令将整个闪存存储器阵列中的所有位设置为 1（所有字节均为 FFh）。接受 BE 命令之前，器件必须发送写入使能（WREN）命令并对其进行解码。该器件也设置状态寄存器中的写入使能锁存（WEL）命令，以使能任意写入操作。

指令字节的第 8 位通过 SI/IO0 锁存在芯片内之后，必须将 CS# 信号驱动为逻辑高电平状态。该操作将导致擦除周期发生，该周期包含对整个闪存存储器阵列的预编程和擦除。如果锁存指令的最后位后不将 CS# 驱动为高电平，则不会执行 BE 操作。

一将 CS# 信号驱动为逻辑高电平状态，就发生擦除周期。在擦除周期过程中，用户可以读取正在写入（WIP）位的值，从而确定该操作是否完成。当器件处于擦除周期过程中时，WIP 位将显示为 1；当擦除周期完成后，WIP 则显示为 0。

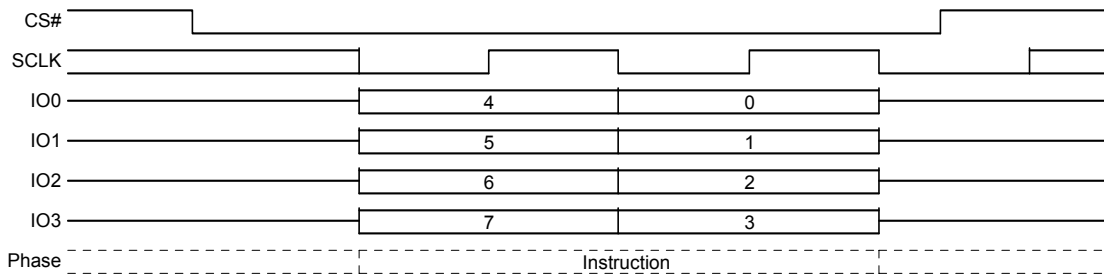
只有块保护（BP2、BP1 和 BP0）位被设为 0 时，才可执行 BE 命令。如果 BP 位为非零数值，将不执行 BE 命令，并且不会报告错误（E_ERR）。BE 命令将跳过所有被 DYB 或 PPB 保护的扇区，并且不会报告错误（E_ERR）。

图 95. 批量擦除命令序列



该命令也支持 QPI 模式。在该模式下，通过 IO0–IO3 信号将指令传入到器件内。

图 96. QPI 模式下的批量擦除命令序列



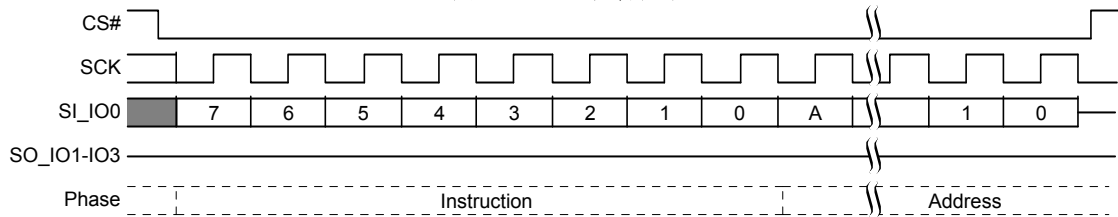
11.6.4 评估擦除状态（EES D0h）

评估擦除状态（EES）指令用于验证对已指定地址的扇区进行的擦除操作是否完全成功。如果成功擦除了选定扇区，则擦除状态位（SR2V[2]）被置 1。如果擦除操作未成功，则该位（SR2V[2]）被清零。

使用 EES 指令可以检测擦除操作失败的原因，原因可能是：断电、复位或擦除操作过程失败。

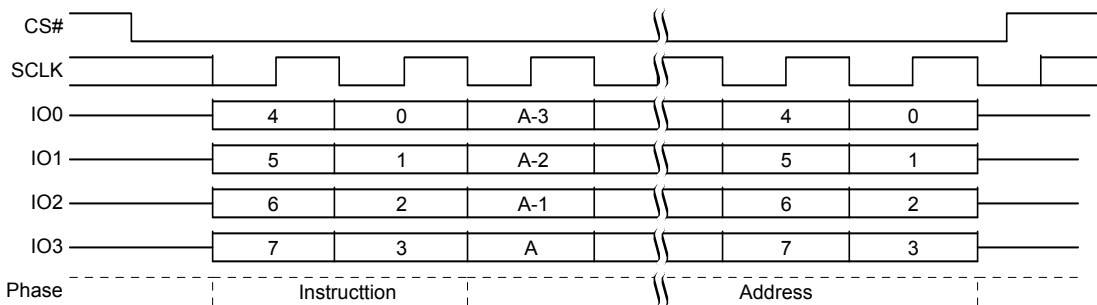
EES 指令后面是 3 字节还是 4 字节地址，具体取决于地址长度配置（CR2V[7]）。EES 指令需要经过 tEES 时间来完成并更新 SR2V 中的擦除状态。通过使用 RDSR1（05h）命令，可以读取 WIP 位（SR1V[0]），用于确定 EES 命令是否结束。然后，可以使用 RDSR2（07h）或 RDAR（65h）命令来读取 SR2V[2]。如果 SR2V[2] = 0，则表示扇区未被擦除，那么必须重新执行擦除操作，以确保该扇区中数据存储的可靠性。

在 EES 命令前不会要求使用写入使能命令（以便设置 WEL 位）。但是，器件自设置了 WEL 位，并且在操作结束时自动将其清除，如读取状态时在 SR1V[1] 中所示情况。

图 97. EES 命令序列

注意

82.A = 地址的 MSB = A23 (ESS D0h, CR2V[7] = 0), 或 = A31 (ESS D0h, CR2V[7] = 1)

该命令也支持 QPI 模式。在该模式下, 指令、地址和数据通过 IO0-IO3 信号被传入到芯片内。

图 98. QPI 模式下的 EES 命令序列

注意

83.A = 地址的 MSB = A23 (ESS D0h, CR2V[7] = 0), 或 = A31 (ESS D0h, CR2V[7] = 1)

11.6.5 擦除或编程挂起 (EPS 85h、75h、B0h)

编程或擦除挂起 (EPS) 命令共有三个指令代码, 这样是为了与传统和替换源软件相兼容。

通过 EPS 命令, 系统可中断编程或擦除操作, 然后读取其它非擦除挂起扇区或非编程挂起页取。仅在编程或扇区擦除操作过程中, 编程或擦除挂起才有效。批量擦除操作不能被挂起。

必须检查状态寄存器 1 (SR1V[0]) 中的正在写入 (WIP) 位, 以了解编程或擦除操作停止的时间。可以使用状态寄存器 2 (SR2[0]) 中的编程挂起状态位来指出编程操作在 WIP 位改成 '0' 时被挂起还是完成。同样, 可以使用状态寄存器 2 (SR2[1]) 中的擦除挂起状态位来确定擦除操作在 WIP 位改成 '0' 时已被挂起还是完成。挂起操作完成所需的时间为 t_{SL} , 请参见第 39 页上的表 17。

可挂起擦除操作以执行编程或读取操作。在擦除挂起过程中, 可通过读取 DYB 阵列来检查扇区的保护, 并可通过对该阵列的写操作来去除或恢复对被编程扇区的保护。

可挂起编程操作来执行读取操作。

对于已挂起擦除或编程操作的扇区, 不能执行新的擦除操作。在此情况下, 擦除命令被忽略。

表 55. 在编程或擦除挂起时可执行的命令

指令名称	指令代码 (十六进制)	在擦除挂起 期间可用	在编程挂起 期间可用	注释
READ	03	X	X	在挂起期间可执行所有阵列读取操作
RDSR1	05	X	X	读取 WIP 以确定挂起期间是否结束的方法
RDAR	65	X	X	读取 WIP 以确定挂起流程结束的其他方法
WREN	06	X		用于执行擦除挂起期间所发出的编程命令。
RDSR2	07	X	X	用于读取挂起状态来确定操作被挂起还是已经完成。
RUID	4C	X	X	在挂起期间允许读取唯一的 ID
PP	02	X		用于在擦除挂起期间执行阵列编程操作。仅在无编程挂起操作时有效 (SR2V[0] = 0)。存在某一挂起编程操作时, 编程命令均被忽略。如果将编程命令发送到擦除挂起扇区中某个位置, 则编程操作失败, 且 P_ERR 位被置为 1。
4PP	12	X		用于在擦除挂起期间执行阵列编程操作。仅在无编程挂起操作时有效 (SR2V[0] = 0)。存在某一挂起编程操作时, 编程命令均被忽略。如果将编程命令发送到擦除挂起扇区中某个位置, 则编程操作失败, 且 P_ERR 位被置为 1。
QPP	32	X		用于在擦除挂起期间执行阵列编程操作。仅在无编程挂起操作时有效 (SR2V[0] = 0)。存在某一挂起编程操作时, 编程命令均被忽略。如果将编程命令发送到擦除挂起扇区中某个位置, 则编程操作失败, 且 P_ERR 位被置为 1。
4QPP	34	X		用于在擦除挂起期间执行阵列编程操作。仅在无编程挂起操作时有效 (SR2V[0] = 0)。存在某一挂起编程操作时, 编程命令均被忽略。如果将编程命令发送到擦除挂起扇区中某个位置, 则编程操作失败, 且 P_ERR 位被置为 1。
4READ	13	X	X	在挂起期间可执行所有阵列读取操作
CLSR	30	X		擦除挂起期间, 如果编程操作失败, 则允许使用擦除状态寄存器命令。请注意, 只在通过设置 CR4NV[2] = 1 清除状态的情况下, 该指令才有效
CLSR	82	X		擦除挂起期间, 如果编程操作失败, 则允许使用擦除状态寄存器命令。
EPR	30	X	X	用于从擦除或编程挂起恢复。请注意, 必须通过设置 CR3NV[2] = 1 将该命令作为恢复命令使用
EPR	7A	X	X	用于从擦除或编程挂起恢复。
EPR	8A	X	X	用于从擦除或编程挂起恢复。
RSTEN	66	X	X	复位可随时发生
RST	99	X	X	复位可随时发生
FAST_READ	0B	X	X	在挂起期间可执行所有阵列读取操作
4FAST_READ	0C	X	X	在挂起期间可执行所有阵列读取操作
DOR	3B	X	X	在挂起期间可执行所有阵列读取操作
4DOR	3C	X	X	在挂起期间可执行所有阵列读取操作
QOR	6B	X	X	四线读取输出 (3 或 4 字节地址)
4QOR	6C	X	X	四线读取输出 (4 字节地址)
EPR	7A	X		用于从擦除挂起恢复。
EPR	8A	X		用于从擦除挂起恢复。
DIOR	BB	X	X	在挂起期间可执行所有阵列读取操作
4DIOR	BC	X	X	在挂起期间可执行所有阵列读取操作
DYBRD	FA	X		擦除挂起期间可能需要移除和恢复动态保护, 以便执行编程操作。
DYBWR	FB	X		擦除挂起期间可能需要移除和恢复动态保护, 以便执行编程操作。
PPBRD	FC	X		在擦除挂起期间, 允许在尝试执行编程命令前检查持久保护。
4DYBRD	E0	X		擦除挂起期间可能需要移除和恢复动态保护, 以便执行编程操作。
4DYBWR	E1	X		擦除挂起期间可能需要移除和恢复动态保护, 以便执行编程操作。
4PPBRD	E2	X		在擦除挂起期间, 允许在尝试执行编程命令前检查持久保护。
QIOR	EB	X	X	在挂起期间可执行所有阵列读取操作
4QIOR	EC	X	X	在挂起期间可执行所有阵列读取操作

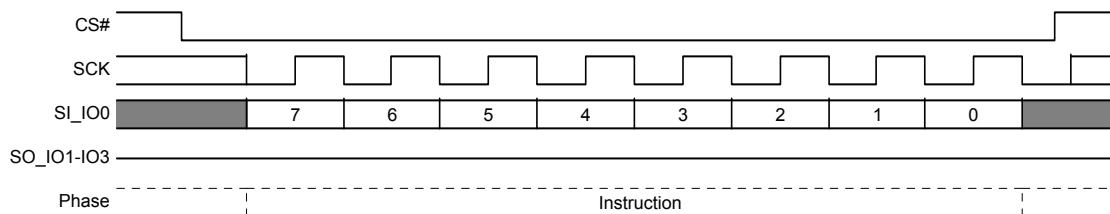
表 55. 在编程或擦除挂起时可执行的命令 (续)

指令名称	指令代码 (十六进制)	在擦除挂起 期间可用	在编程挂起 期间可用	注释
DDRQIOR	ED	X	X	在挂起期间可执行所有阵列读取操作
4DDRQIOR	EE	X	X	在挂起期间可执行所有阵列读取操作
RESET	F0	X	X	复位可随时发生
MBR	FF	X	X	在挂起期间可用于复位读取操作

读取擦除挂起扇区或编程挂起页中的任何地址都会生成不确定的数据。

在擦除或编程挂起过程中，不允许执行 WRR、WRAR 或 PPB 擦除命令，因此，在擦除挂起时不能改变块保护或 PPB 位。如果在擦除挂起中需要对某些扇区进行编程，只有通过 DYB 位进行保护的扇区才能在擦除挂起中禁止保护。

擦除挂起模式下的编程操作完成后，器件返回擦除挂起模式。系统可以通过读取状态寄存器的 WIP 位来确定编程操作的状态，同在执行标准编程操作一样。

图 99. 编程或擦除挂起命令序列


该命令也支持 QPI 模式。在 QPI 模式下，通过 IO0-IO3 信号将指令传入到器件内。

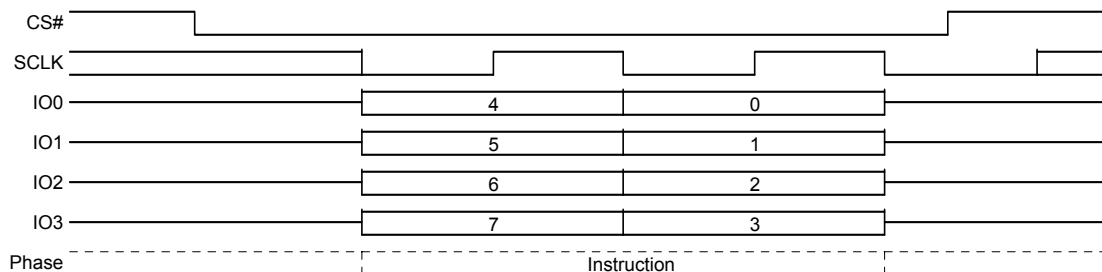
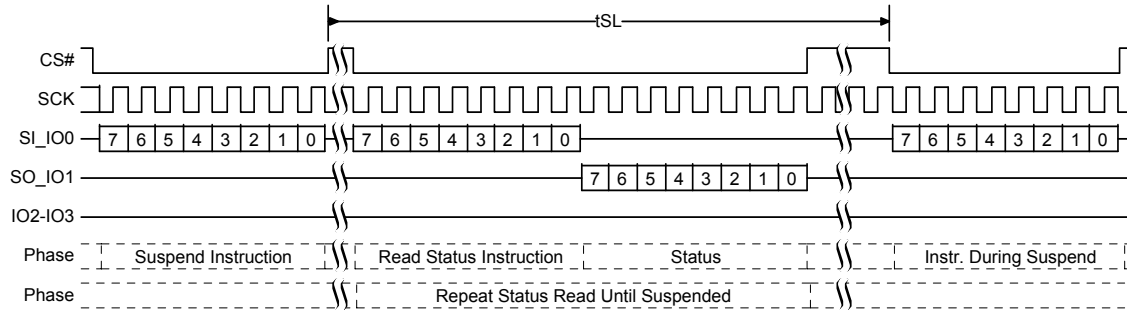
图 100. QPI 模式下的编程或擦除挂起命令序列


图 101. 带有连续命令序列的编程或擦除挂起命令


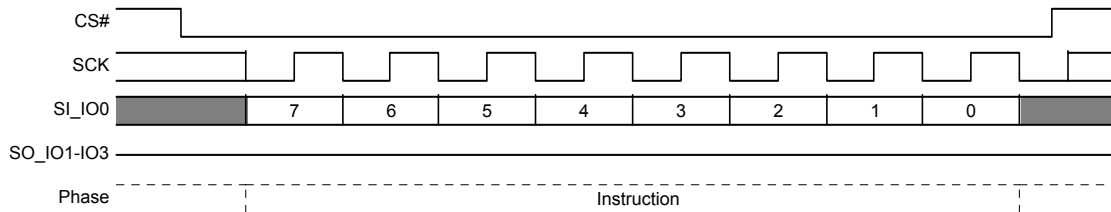
11.6.6 擦除或编程恢复 (EPR 7Ah、8Ah、30h)

器件必须写入擦除或编程恢复命令来恢复被挂起的操作。擦除或编程恢复 (EPR) 命令共有三个指令代码，为的是与传统和替换源软件相兼容。

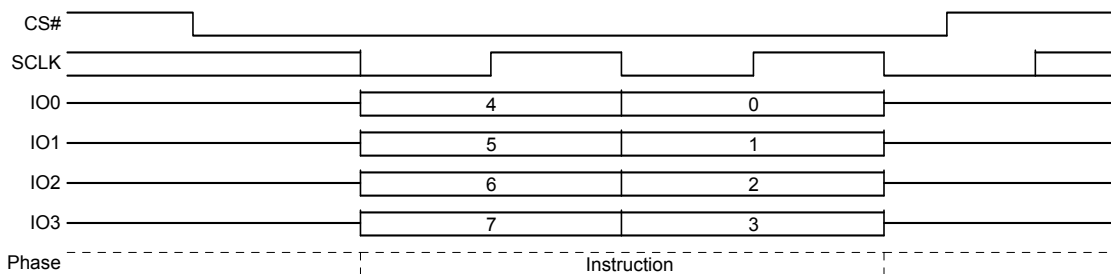
编程或读取操作完成后，在编程或擦除挂起过程中，器件将发送擦除或编程恢复命令，以便继续执行被挂起的操作。

发送擦除或编程恢复命令后，状态寄存器中的 WIP 位将被设置为 1，并且恢复被挂起的编程操作（若它被挂起）。如果没有被挂起的编程操作，将恢复挂起的擦除操作。如果没有任何被挂起的编程或擦除操作，将忽略恢复命令。

编程或擦除操作可以根据需要被中断。例如，执行编程挂起命令后可立即执行编程恢复命令。但为了正常完成编程或擦除操作，恢复操作和下一次挂起命令之间的时间不能小于 t_{RS} 。请参见第 39 页上的表 17。

图 102. 擦除或编程恢复的命令序列


该命令也支持 QPI 模式。在该模式下，通过 IO0–IO3 信号将指令传入到器件内。

图 103. QPI 模式下擦除或编程恢复的命令序列


11.7 一次性编程阵列命令

11.7.1 OTP 编程 (OTPP 42h)

OTP 编程命令将编程一次性编程区域中的数据，该数据与主阵列数据位于不同的地址空间。OTP 区域大小为 1024 字节，因此该命令的地址位从 A31 到 A10 必须为 0。欲了解 OTP 区域的详细信息，请参考第 47 页上的 [OTP 地址空间](#)。

接受 OTP 编程命令之前，器件必须发送写入使能 (WREN) 命令并对其进行解码。该器件也设置状态寄存器中的写入使能锁存 (WEL) 命令，以使能任意写入操作。通过检查 SR1V 中的 WIP 位，可以确定该操作是否完成。同样，通过检查 SR1V 中的 P_ERR 位可以指出在操作过程中是否发生了错误。

为了以位粒度编程 OTP 阵列，可将数据字节中的剩下位设置为 “1”。

假定该区域不被锁定，则可对 OTP 存储器空间的每一区域进行一次或多次编程。尝试编程锁定区域中的零值时，操作将失败，而且 SR1V 中的 P_ERR 位被设置为 “1”。就算在保护区中编程 1 值，也不会报告错误，并且不会设置 P_ERR 位。后续的 OTP 编程操作只能使用于未被编程的位（即 “1” 值数据）。对一个 ECC 单元执行多次编程使得该单元上的 ECC 功能被禁用。

OTP 编程命令和页编程命令的协议相同。欲了解指令序列，请参见第 102 页上的 [页编程 \(PP 02h 或 4PP 12h\)](#)。

11.7.2 OTP 读取 (OTPR 4Bh)

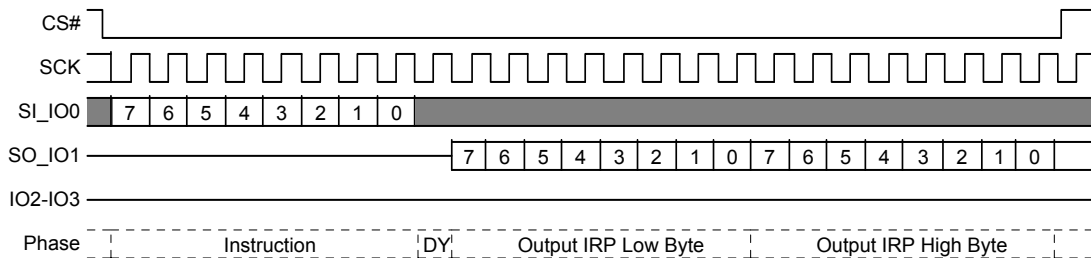
OTP 读取命令用于读取 OTP 区域中的数据。OTP 区域大小为 1024 字节，因此该命令的地址位从 A31 到 A10 必须为 0。欲了解 OTP 区域的详细信息，请参考第 47 页上的 [OTP 地址空间](#)。OTP 读取命令的协议与快速读取命令的协议相同，只是当 OTP 地址达到最大值后，该命令不会翻转到起始地址；但是超出最大地址的数据将是不确定的。OTP 读取命令的读取延迟由 CR2V[3:0] 中的值指定。欲了解指令序列，请参见第 94 页上的 [快速读取 \(FAST_READ 0Bh 或 4FAST_READ 0Ch\)](#)。

11.8 高级扇区保护命令

11.8.1 ASP 读取 (ASPRD 2Bh)

ASP 读取指令 2Bh 在 SCK 信号的上升沿上被传入到 SI。然后，在串行输出 SO 上按最低有效字节优先方式输出 16 位 ASP 寄存器的内容。在 SCK 的下降沿上以 SCK 频率输出一位。通过提供 16 时钟周期的倍数，可以连续读取 ASP 寄存器。ASP 读取 (ASPRD) 命令的最大工作时钟频率为 133 MHz。

图 104. ASPRD 命令序列



11.8.2 ASP 编程 (ASPP 2Fh)

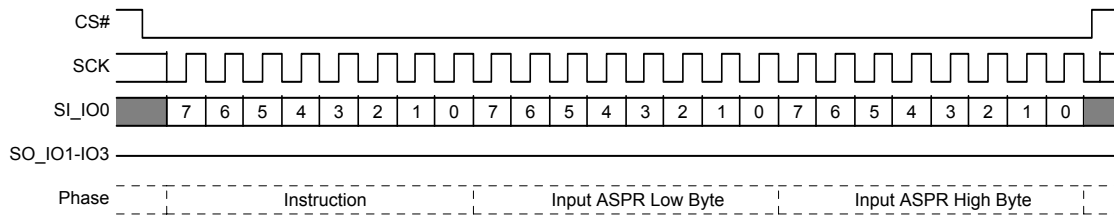
接受 ASP 编程 (ASPP) 命令前, 器件必须发送写入使能 (WREN) 命令。成功解码写入使能 (WREN) 命令后, 器件将设置状态寄存器中的写入使能锁存 (WEL) 位, 以使能任意写入操作。

通过将 CS# 驱动为逻辑低电平状态, 可以输入 ASPP 命令, 然后按最低有效字节优先方式在 SI 上输入指令和两个数据字节。ASP 寄存器的长度为两个数据字节。

与所有其它编程操作相同, ASPP 命令也会影响状态和配置寄存器的 P_ERR 和 WIP 位。

数据字节的第十六位被锁存后, 必须将 CS# 驱动为逻辑高电平状态。否则, 不能执行 ASPP 命令。CS# 被驱动为逻辑高电平状态时, 会立即启动自定时 ASPP 操作。在执行 ASPP 操作的过程中, 可以读取状态寄存器, 以检查正在写入 (WIP) 位的值。执行自定时 ASPP 操作期间, 正在写入 (WIP) 位为 '1'; 完成该操作时, WIP 位为 '0'。完成 ASPP 操作时, 写入使能锁存 (WEL) 位将被设为 '0'。

图 105. ASPP 命令



11.8.3 DYB 读取 (DYBRD FAh 或 4DYBRD E0h)

该指令在 SCK 信号的上升沿上被锁存到 SI/IO0 中。指令的后面是 24 位或 32 位地址 (具体取地址长度配置 CR2V[7] 值), 用于选择在所需扇区中的位置 0。请注意, 必须将器件未使用的高端地址位设置为 0。然后, 8 位 DYB 访问寄存器内容将通过串行输出 SO/IO1 被移出。在 SCK 的下降沿上以 SCK 频率输出一位。通过提供 8 个时钟周期的倍数, 可以连续读取相同的 DYB 访问寄存器。DYB 寄存器的地址不递增, 所以这不是读取整个 DYB 阵列的方法。必须通过一个单独 DYB 读取命令读取每个位置。READ 命令的最大工作时钟频率为 133 MHz。

图 106. DYBRD 命令序列



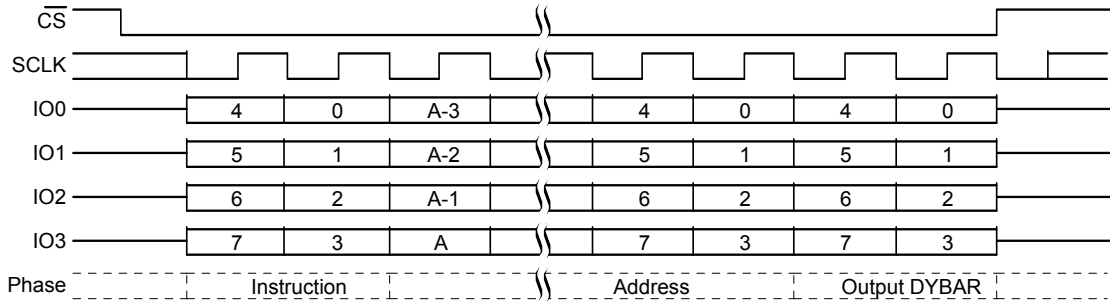
注意

84.A = 地址的 MSB = 23 (地址长度 CR2V[7] = 0), 或 31 (CR2V[7] = 1 和命令 FAh)。

85.A = 地址的 MSB = 31 (使用命令 E0h)。

在该模式下, 通过 IO0-I/O3 信号将指令传入到器件内。

图 107. QPI 模式下的 DYBRD 命令序列



注意

86.A = 地址的 MSB = 23 (地址长度 CR2V[7] = 0), 或 31 (CR2V[7] = 1 和命令 FAh)。

87.A = 地址的 MSB = 31 (QPI 模式下的 E0hDYBRD 命令序列)

11.8.4 DYB 写入 (DYBWR FBh 或 4DYBWR E1h)

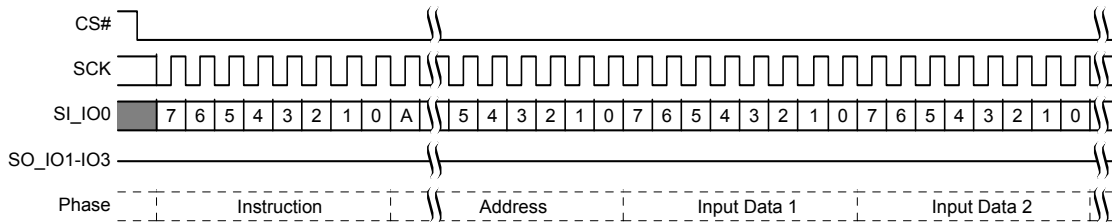
接收 DYB (DYBWR) 命令之前, 器件必须发送写入使能 (WREN) 命令。成功解码写入使能 (WREN) 命令后, 器件将设置状态寄存器中的写入使能锁存 (WEL) 位, 以使能任意写入操作。

通过将 CS# 驱动为逻辑低电平, 可以启动 DYBWR 命令。然后在 SI/IO0 上分别传入指令、用于选择所需扇区中的位置 0 的 24 位或 32 位地址 (具体取决于地址长度配置 CR2V[7])。请注意, 必须将器件未使用的高端地址位设置为 0, 最后是数据字节。DYB 访问寄存器的长度为一个数据字节。数据值必须为 00h 或 FFh, 从而对所选扇区提供保护或不保护。

与所有其它编程操作相同, DYBWR 命令也会影响状态和配置寄存器的 P_ERR 和 WIP 位。

数据字节的第八位被锁存后, 必须将 CS# 驱动为逻辑高电平状态。CS# 被驱动为逻辑高电平状态时, 会立即启动自定时 DYBWR 操作。在执行 DYBWR 操作的过程中, 可以读取状态寄存器以检查正在写入 (WIP) 位的值。执行自定时 DYBWR 操作期间, 正在写入 (WIP) 位为 '1'; 完成该操作时, WIP 位为 '0'。完成 DYBWR 操作时, 写使能锁存 (WEL) 位将被设置为 '0'。

图 108. DYB 写入命令序列

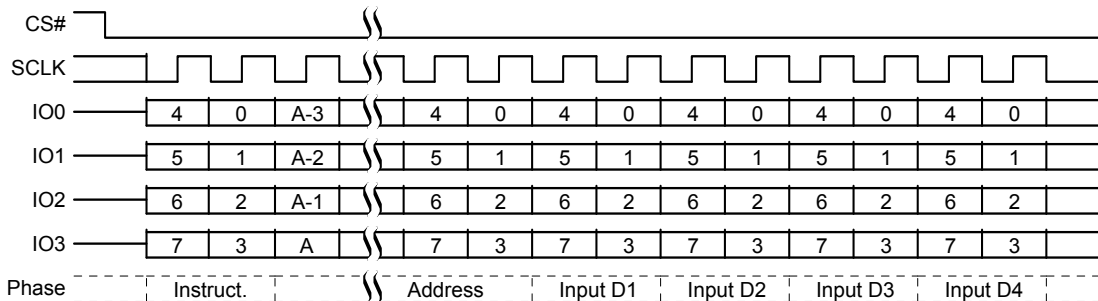


注意

88.A = 地址的 MSB = 23 (地址长度 CR2V[7] = 0), 或 31 (CR2V[7] = 1 和命令 FBh)。

89.A = 地址的 MSB = 31 (使用命令 E1h)。

该命令也支持 QPI 模式。在 QPI 模式下, 指令、地址和数据将通过 IO0-IO3 信号传入到芯片内。

图 109. QPI 模式下的 DYB 写入命令序列

注意

90.A = 地址的 MSB = 23 (地址长度 CR2V[7] = 0), 或 31 (CR2V[7] = 1 和命令 FBh)。

91.A = 地址的 MSB = 31 (使用命令 E1h)。

11.8.5 PPB 读取 (PPBRD FCh 或 4PPBRD E2h)

指令 E2h 在 SCK 信号的上升沿上被传送到 SI/IO0 中, 后面是 24 或 32 位地址 (取决于地址长度配置位 CR2V[7]), 用于选择所需扇区中的零地址 (请注意, 器件不使用的高端地址位必须设置为 0)。然后, 8 位 PPB 访问寄存器内容将通过 SO/IO1 被移出。

通过提供 8 个时钟周期的倍数, 可以连续读取相同的 PPB 访问寄存器。PPB 寄存器的地址不递增, 所以这不是读取整个 PPB 阵列的方法。必须通过一个单独 PPB 读取命令读取每个位置。PPB 读取命令的最大工作时钟频率为 133 MHz。

图 110. PPB 读取命令序列

注意:

92.A = 地址的 MSB = 23 (地址长度 CR2V[0] = 0), 或 31 (CR2V[0] = 1 和命令 FCh)。

93.A = 地址的 MSB = 31 (使用命令 E2h)。

11.8.6 PPB 编程 (PPBP FDh 或 4PPBP E3h)

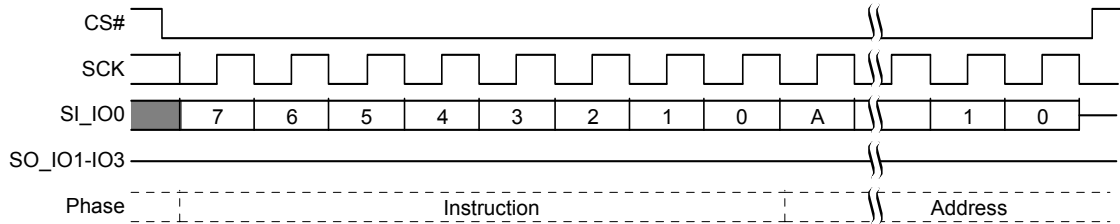
接受 PPB 编程 (PPBP) 命令前, 器件必须发送写入使能 (WREN) 命令。成功解码写入使能 (WREN) 命令后, 器件将设置状态寄存器中的写入使能锁存 (WEL) 位, 以使能任意写入操作。

通过将 CS# 驱动为逻辑低电平, 可以启动 PPBP 命令。然后分别传入指令、用于选择所需扇区中的位置 0 的 24 位或 32 位地址 (具体取决于地址长度配置 CR2V[7])。请注意, 器件不使用的高端地址位必须设置为 0)。

与所有其它编程操作相同, PPBP 命令也会影响状态和配置寄存器的 P_ERR 和 WIP 位。

锁存地址最后位结束后, 必须将 CS# 驱动为逻辑高电平状态。否则, 不能执行 PPBP 命令。CS# 被驱动为逻辑高电平状态时, 会立即启动自定时 PPBP 操作。在执行 PPBP 操作的过程中, 可以读取状态寄存器, 以检查正在写入 (WIP) 位的值。执行自定时 PPBP 操作期间, 正在写入 (WIP) 位为 '1'; 完成该操作时, WIP 位为 '0'。完成 PPBP 操作时, 写入使能锁存 (WEL) 位将被设为 '0'。

图 111. PPB 命令序列



注意

94.A = 地址的 MSB = 23（地址长度 CR2V[0] = 0），或 31（CR2V[0] = 1，命令 FDh）。

95.A = 地址的 MSB = 31（使用命令 E3h）。

11.8.7 PPB 擦除（PPBE E4h）

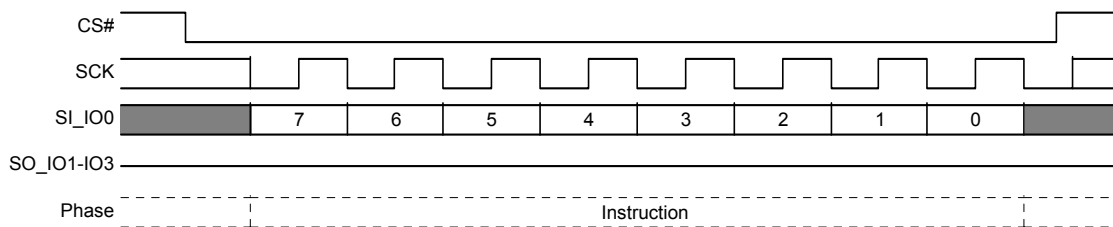
PPB 擦除（PPBE）指令将所有 PPB 位设置为 1。接受 PPB 擦除命令前，器件必须发送写入使能（WREN）命令并对其进行解码。该器件也设置状态寄存器中的写入使能锁存（WEL）命令，以使能任意写入操作。

E4h 指令在 SCK 信号的上升沿上被传送到 SI/IO0 中。

指令字节的第 8 位通过 SI/IO0 锁存在芯片内之后，必须将 CS# 信号驱动为逻辑高电平状态。该操作将启动内部擦除周期，包括对整个 PPB 存储器阵列的预编程和擦除操作。传入指令的第 8 位后，如果未将 CS# 信号驱动为逻辑高电平状态，则不会执行 PPB 擦除操作。

在内部擦除周期中，用户可以读取正在进行写入操作（WIP）位的值，从而检查该操作是否完成。当器件处于擦除周期过程中时，WIP 位将显示为 ‘1’；当擦除周期完成后，WIP 则显示为 ‘0’。执行 PPB 擦除期间，不能挂起擦除操作。

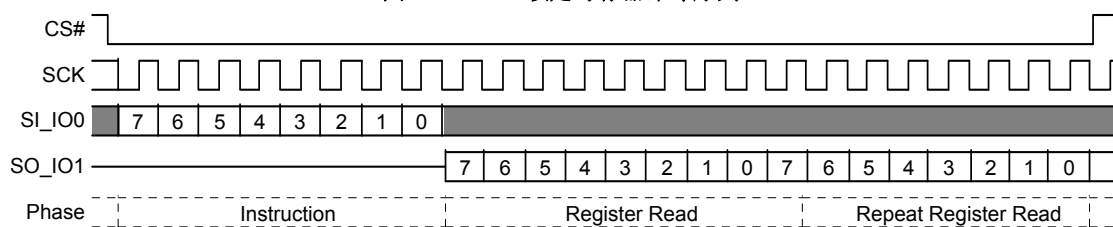
图 112. PPB 擦除命令序列



11.8.8 PPB 锁定位读取（PLBRD A7h）

通过 PPB 锁定位读取（PLBRD）命令，可以从 SO/IO1 上读出 PPB 锁定寄存器的内容。通过提供 8 时钟周期的倍数，可以连续读取 PPB 锁定寄存器。如果器件处于待机状态并不在执行任何其他操作，则仅能读取 PPB 锁定寄存器中的内容。建议在发出新命令前，应该检查状态寄存器中的正在写入（WIP）位。

图 113. PPB 锁定寄存器命令序列



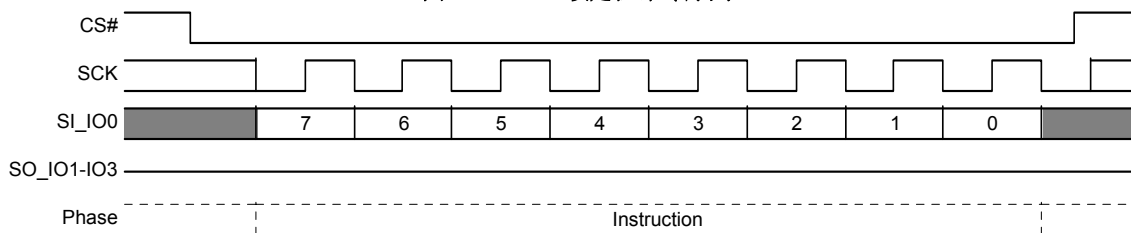
11.8.9 PPB 锁定位写入 (PLBWR A6h)

PPB 锁定位写入 (PLBWR) 命令将 PPB 锁定寄存器清除为 0。接受 PLBWR 命令之前，器件必须发送写入使能 (WREN) 命令并对其进行解码。该器件也设置状态寄存器中的写入使能锁存 (WEL) 命令，以使能任意写入操作。

通过将 CS# 驱动为逻辑低电平状态，可以输入 PLBWR 命令，然后输入指令。

指令的第 8 位被锁存后，必须将 CS# 信号驱动为逻辑高电平状态。否则，不能执行 PLBWR 命令。CS# 被驱动为逻辑高电平状态时，会立即启动自定时 PLBWR 操作。在执行 PLBWR 操作的过程中，还可以读取状态寄存器，以检查正在写入 (WIP) 位上的值。执行自定时 PLBWR 操作期间，正在写入 (WIP) 位为 '1'；完成该操作时，WIP 位为 '0'。完成 PLBWR 操作时，写使能锁存 (WEL) 位将被设置为 '0'。PLBWR 命令的最大时钟频率为 133 MHz。

图 114. PPB 锁定位命令序列

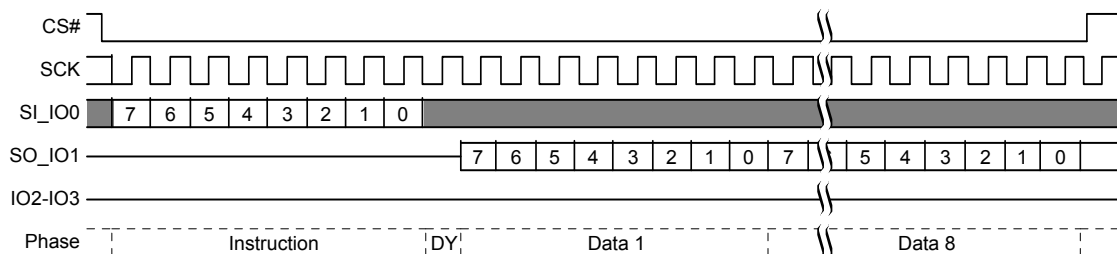


11.8.10 密码读取 (PASSRD E7h)

仅在编程正确密码值后且通过在 ASP 寄存器 (ASP[2]) 中将密码保护模式位编程为 '0' 来选定密码模式前，才能读取正确密码值。选定密码保护模式后，密码不再可读，此时 PASSRD 命令将输出未定义的数据。

PASSRD 命令被传入到 SI/IO0 中。然后，在串行输出 SO/IO1 上移出 64 位密码，并且为最低有效字节优先，每个字节的最高有效位优先。在 SCK 的下降沿上以 SCK 频率输出一位。通过提供 64 时钟周期的倍数，可以连续读取密码。PASSRD 命令的最大工作时钟频率为 133 MHz。

图 115. 密码读取 (PASSRD) 命令序列



11.8.11 密码编程 (PASSP E8h)

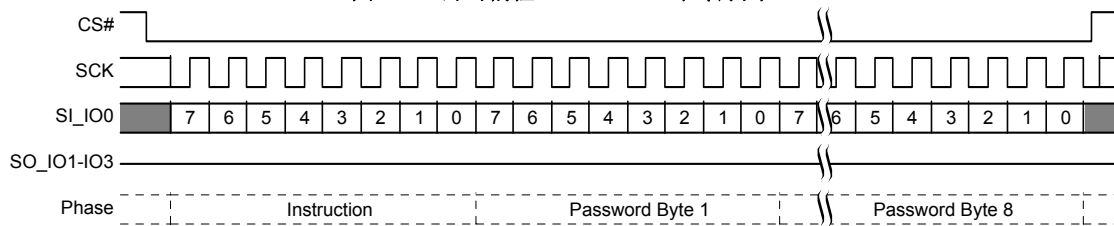
接受密码编程 (PASSP) 命令前, 器件必须发送写入使能 (WREN) 命令并对其进行解码。对写使能 (WREN) 命令进行解码后, 器件将设置状态寄存器中的写使能锁存 (WEL) 位以使能 PASSP 操作。

仅当通过在 ASP 寄存器 (ASP[2]) 中将密码保护模式位编程为 '0' 来选定密码模式前, 才能编程密码。选定密码保护模式后, PASSP 命令将被忽略。

通过将 CS# 驱动为逻辑低电平状态, 可以输入 PASSP 命令, 然后输入 SI/IO0 上的指令和密码数据字节, 最低有效字节优先, 每个字节的最高有效位优先。密码的长度为六十四 (64) 位。

数据的第六十四 (64) 位被锁存后, 必须将 CS# 驱动为逻辑高电平状态。否则, 不能执行 PASSP 命令。CS# 被驱动为逻辑高电平状态时, 会立即启动自定时 PASSP 操作。在执行 PASSP 操作的过程中, 可以读取状态寄存器以检查正在写入 (WIP) 位的值。执行自定时 PASSP 操作期间, WIP 位为 1; 完成该操作时, WIP 位为 0。PASSP 命令可以报告状态寄存器中 P_ERR 位的编程错误。完成 PASSP 操作时, 写使能锁存 (WEL) 位将被设置为 '0'。PASSP 命令的最大时钟频率为 133 MHz。

图 116. 密码编程 (PASSP) 命令序列



11.8.12 密码解锁 (PASSU E9h)

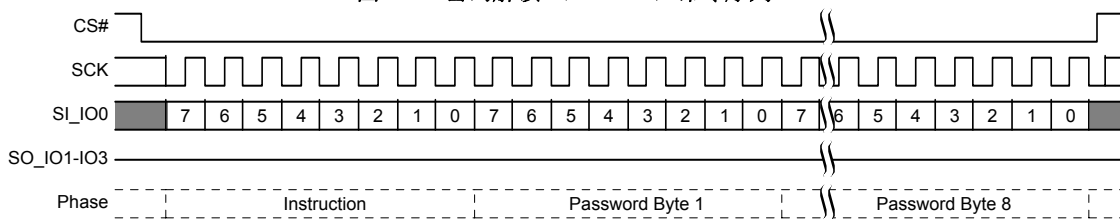
通过将 CS# 驱动为逻辑低电平状态, 可以启动 PASSU 命令。然后按最低有效字节优先、每个字节的最高有效位优先方式通过 SI/IO0 上锁存指令和密码数据字节。密码的长度为六十四 (64) 位。

数据的第六十四 (64) 位被锁存后, 必须将 CS# 驱动为逻辑高电平状态。否则, 不能执行 PASSU 命令。CS# 被驱动为逻辑高电平状态时, 会立即启动自定时 PASSU 操作。在执行 PASSU 操作的过程中, 可以读取状态寄存器以检查正在写入 (WIP) 位的值。执行自定时 PASSU 操作期间, 正在写入 (WIP) 位为 '1'; 完成该操作时, WIP 位为 '0'。

如果 PASSU 命令提供的密码与密码寄存器中隐藏密码不匹配, 将 P_ERR 位设为 '1' 便可以报告错误。状态寄存器的 WIP 位仍保持为 '1'。另外需要使用 CLSR 命令来清除状态寄存器中的内容, 使用复位命令来对器件进行软件复位, 或驱动 RESET# 输入为低电平以启动硬件复位, 从而使 P_ERR 和 WIP 位返回 '0'。这样会使器件进入待机状态, 以准备好接受新命令 (如 PASSU 命令)。

如果密码不匹配, PPB 锁定位将被设置为 '1'。PASSU 命令的最大时钟频率为 133 MHz。

图 117. 密码解锁 (PASSU) 命令序列



11.9 复位命令

通过将非易失性默认值重新载入易失性寄存器内，软件控制的复位命令将器件恢复为上电时的默认状态。然而，软件复位不能修改配置寄存器 CR1V[0] 中的 FREEZE 易失性位和 PPB 锁定寄存器中的 PPB 锁定易失性位。不能使用软件复位来将其它安全配置位代替 FREEZE 或 PPB 锁定位的保护机制。

FREEZE 位和 PPB 锁定位会保持在软件复位前的最后值。要想清除 FREEZE 位并将 PPB 锁定位设置为其保护模式选定的上电状态，需要完成完整的上电复位序列或硬件复位。

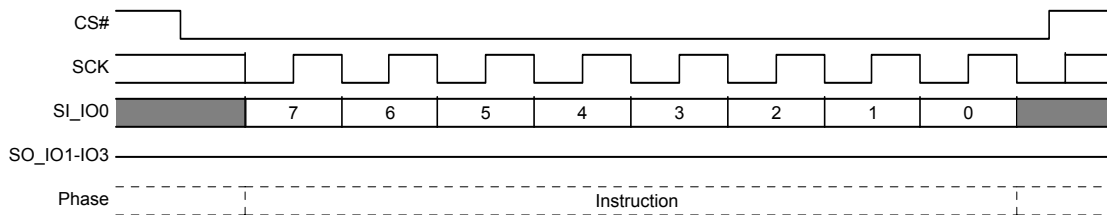
配置寄存器 (CR1NV)、TBPROT_O、TBPARM 和 BPNV_O 中的非易失性位在软件复位后仍保持它们的状态。

状态寄存器 (SR1V) 中的块保护位 BP2、BP1 和 BP0 仅在 FREEZE = 0 时才复位为默认状态。

如果在指令结束时 CS# 被置为高电平，将会执行复位命令 (RST 或 RESET)，该过程需要 t_{RPH} 时间。

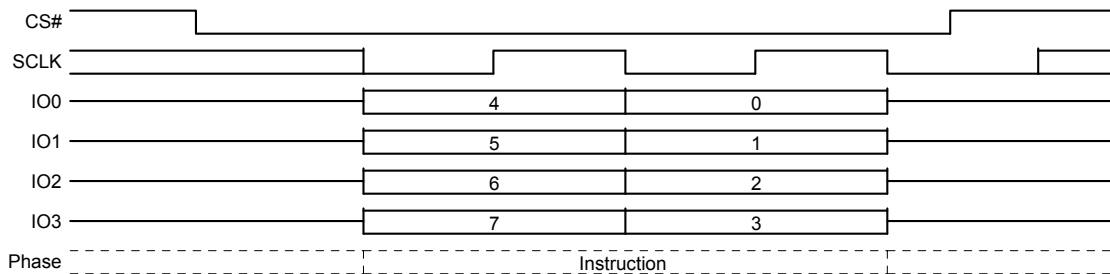
如果无法完成先前的上电复位 (POR)，那么复位命令会触发一个完整的上电序列，需要 t_{PU} 来完成该序列。

图 118. 软件 / 模式位复位命令序列



该命令也支持 QPI 模式。在该模式下，通过 IO0–IO3 信号将指令传入到器件内。

图 119. QPI 模式下的软件复位 / 模式位命令序列



11.9.1 软件复位使能 (RSTEN 66h)

在复位命令 (RST) 前需要复位使能 (RSTEN) 命令，这样的软件复位是一个双命令的序列。如果 RSTEN 命令后不是 RST 命令，将清除复位使能条件并防止识别随后的 RST 命令。

11.9.2 软件复位 (RST 99h) 命令

紧随 RSTEN 命令的复位 (RST) 命令会启动软件复位过程。

11.9.3 传统的软件复位 (RESET F0h)

传统软件复位 (RESET) 是用于启动软件复位过程的单一命令。默认情况下该命令被禁用，但是通过将 CR3V[0] 编程为 1 可使能该命令，这样是为了与赛普拉斯的传统 FL-S 器件软件兼容。

11.9.4 模式位复位 (MBR FFh) 命令

模式位复位 (MBR) 命令用于使器件从高性能读取模式返回到正常待机模式，并且等待新命令。因为有些器件封装没有硬件 RESET# 输入，并且处于高性能连续读取模式的器件可能不会识别出任何正常的 SPI 命令，因此器件可能无法区分是系统硬件复位还是软件复位命令。在执行系统复位（当 RESET# 信号不可用时）后或在发送软件复位前，建议使用 MBR 命令以确保器件已经退出高性能连续读取模式。

在八个 SCK 周期内，MBR 命令将通过 SI/IO0 发送 ‘1’。在这些周期期间，IO1-IO3 被视为“无需关注”。该命令也支持 QPI 模式。在 QPI 模式下，指令将通过 IO0-IO3 信号传入到芯片内，每字节要两个时钟周期。

11.10 DPD 命令

11.10.1 进入深度掉电 (DPD B9h)

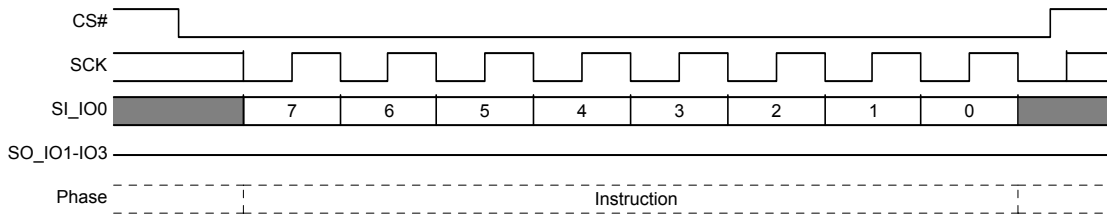
虽然在执行正常操作时待机电流相对低，但使用深度掉电命令还可以更加降低待机电流。由于能够降低功耗，深度掉电 (DPD) 命令对电池供电的应用非常有用（请参考第 26 页上的直流特性中的 I_{DPD} ）。

仅在状态寄存器 1 易失性正在写入 (WIP) 位被清零 ($SR1V[0] = 0$)，表示器件不执行嵌入式算法时，才可接受 DPD 命令。

通过将 CS# 引脚驱动为低电平并发送指令代码“B9h”，启动该命令，如第 119 页上的图 120 所示。锁存第八位后，必须将 CS# 引脚驱动为高电平。否则，不能执行深度掉电命令。CS# 被驱动为高电平后，在 t_{DP} （请参见第 30 页上的时序规范）期间，器件将进入掉电状态。

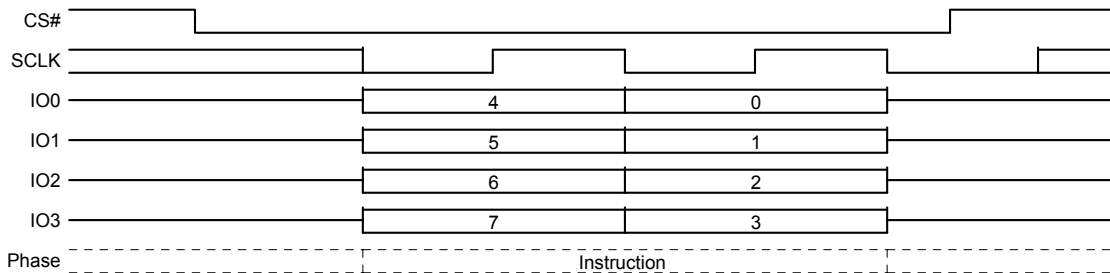
在掉电状态下，器件只能识别出退出深度掉电模式命令（使器件恢复正常操作）。所有其它命令均被忽略，包括读取状态寄存器命令，在执行正常操作过程中，该命令始终可用。该状态只能识别一个命令而忽略所有其它命令，所以掉电状态对写保护很有用。在接口待机模式下，器件始终被加电，消耗的是 I_{CC1} 大小的待机电流。

图 120. 深度掉电 (DPD) 命令序列



该命令也支持 QPI 模式。在该模式下，通过 IO0-IO3 信号将指令传入到器件内。

图 121. QPI 模式下的深度掉电 (DPD) 命令序列



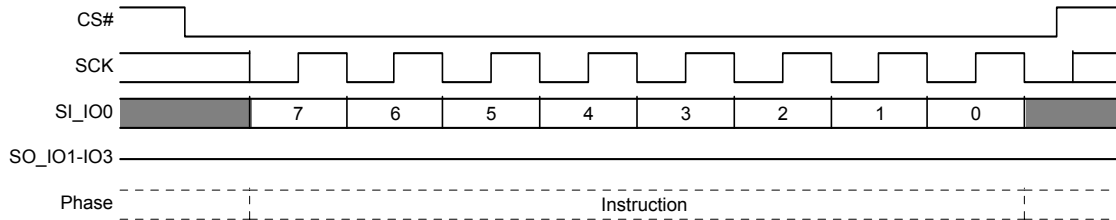
11.10.2 退出深度掉电模式 (RES ABh)

退出深度掉电命令用于将器件退出深度掉电状态。在某些传统 SPI 器件中，RES 命令也用于获取器件的电子标识 (ID) 号。但 RES 命令不支持器件 ID 函数。

要想使器件退出深度掉电状态，首先要将 CS# 引脚驱动为低电平、发送指令代码“ABh”，然后将 CS# 信号驱动为高电平，即实现了该命令，如第 120 页上的图 122 所示。器件恢复正常操作并接受其他命令前，执行退出深度掉电命令需要 t_{RES} （请参见第 30 页上的时序规范）时长。在 t_{RES} 期间，CS# 引脚必须保持为高电平状态。

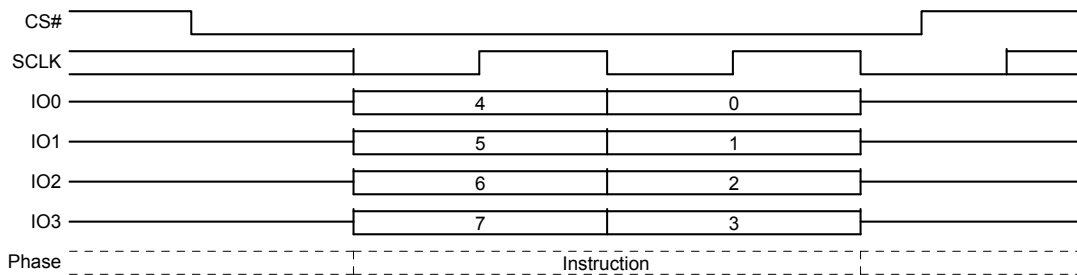
硬件复位也能使器件退出 DPD 状态。

图 122. 退出深度掉电模式 (RES) 命令序列



该命令也支持 QPI 模式。在该模式下，通过 IO0–IO3 信号将指令传入到器件内。

图 123. 退出深度掉电模式 (RES) 命令序列 — QPI 模式



12. 数据完整性

12.1 擦除耐久性

表 56. 擦除次数

参数	最小值	单位
每个主闪存阵列扇区的编程 / 擦除次数	10 万	编程 / 擦除周期
每个 PPB 阵列或非易失性寄存器阵列的编程 / 擦除次数 (96)	10 万	编程 / 擦除周期

注意:

96. 每次将写命令发送到非易失性寄存器都会使整个非易失性寄存器阵列上的数据增加一个 P/E 周期。OTP 位和寄存器的位内部位于一个不受 P/E 周期影响的单独阵列中。

12.2 数据保留时间

表 57. 数据保留时间

参数	测试条件	最小时间	单位
数据保留时间	1 万个编程 / 擦除周期	20	年
	10 万个编程 / 擦除周期	2	年

更多有关数据完整性的信息，请联系赛普拉斯销售和 FAE 部门。请参考 [AN98549 — 保留与耐久性的管理与验证](#) 以了解更多信息。

13. 软件接口参考

13.1 OTP 存储空间地址映射

SFDP 地址空间具有起始地址为 0 的标示，用于识别 SFDP 数据结构并向每个参数提供一个指针。参数由 JEDEC JESD216 B 标准规定。赛普拉斯通过指向 ID-CFI 地址空间来提供另一个参数；ID-CFI 地址空间是 SFDP 地址空间的子集。JEDEC 参数位于 ID-CFI 地址空间中，因此它包括 CFI 参数和 SFDP 参数。这样可通过 RSFDP 或 RDID 命令来访问 SFDP 和 ID-CFI 信息。

表 58. SFDP 映射概述

字节地址	说明
0000h	JEDEC JESD216B SFDP 空间中的地址零 — SFDP 头文件的起点
...	SFDP 头文件的剩下部分，后面是未定义空间
1000h	ID-CFI 空间中的地址 0 — ID-CFI 参数表的起点
...	ID-CFI 参数
1090h	SFDP 参数表的起点，这些参数表被分组为 CFI 参数表中的某一个（CFI 参数开始于 108Eh，SFDP 参数表数据是开始于 1090h 的双字对齐）
...	SFDP 参数表剩余部分的后面为 CFI 参数或未定义空间

13.2 器件 ID 和通用闪存接口（ID-CFI）地址映射 — 标准

13.2.1 字段定义

表 59. 制造商和器件 ID

字节地址	数据	说明
00h	01h	赛普拉斯制造商 ID
01h	02h	器件 ID 最高有效字节 — 存储器接口类型
02h	17h (64 Mb)	器件 ID 最低有效字节 — 容量
03h	4Dh	ID-CFI 长度 — 后面的字节数量。将该值添加到当前地址 03h 上会获取 ID-CFI 传统地址映射中的最后有效地址。传统 CFI 地址映射以主要供应商特定的扩展查询数值结束。原始的传统长度得到保持，使之能与软件向后兼容。然而，CFI 查询识别字符串包括一个指向备用供应商特定扩展查询的指针，该查询包含 FS-S 系列的其他相关信息。
04h	00h (统一的 256 KB 物理扇区) 01h (统一的 64 KB 物理扇区)	物理扇区架构 除了统一扇区外，可以将 FS-S 系列配置为带有或不带有 4 KB 参数扇区。
05h	81h (FS-S 系列)	系列 ID
06h	xxh	模型的 ASCII 字符 请参考第 141 页上的订购器件型号了解模型编号的定义。
07h	xxh	
08h	xxh	预留
09h	xxh	预留
0Ah	xxh	预留
0Bh	xxh	预留
0Ch	xxh	预留
0Dh	xxh	预留
0Eh	xxh	预留
0Fh	xxh	预留

表 60. CFI 查询识别字符串

字节地址	数据	说明
10h 11h 12h	51h 52h 59h	查询唯一 ASCII 字符串“QRY”
13h 14h	02h 00h	主要 OEM 命令集 FL-P 向后兼容命令集 ID
15h 16h	40h 00h	主扩展表地址
17h 18h	53h 46h	备用 OEM 命令集 SPI (F) 接口的 Ascii 字符“FS”，S 技术
19h 1Ah	51h 00h	备用 OEM 扩展表地址

表 61. CFI 系统接口字符串

字节地址	数据	说明
1Bh	17h	V _{CC} 最小值 (擦除 / 编程): 100 毫伏 BCD
1Ch	19h	V _{CC} 最大值 (擦除 / 编程): 100 毫伏 BCD
1Dh	00h	V _{PP} 最小电压 (00h = 不存在 V _{PP} 引脚)
1Eh	00h	V _{PP} 最大电压 (00h = 不存在 V _{PP} 引脚)
1Fh	09h	单字节编程的典型超时时间为 2 ^N μs
20h	09h	容量最小的页编程的典型超时时间为 2 ^N μs (00h = 不支持)
21h	08h (4 KB 或 64 KB)	单个扇区擦除的典型超时时间为 2 ^N ms
22h	05h (64 Mb)	整个芯片擦除的典型超时时间为 2 ^N ms (00h = 不支持)
23h	02h	字节编程的最大超时时间为典型值的 2 ^N 倍
24h	02h	页编程的最大超时时间为典型值的 2 ^N 倍
25h	03h	单个扇区擦除的最大超时时间为典型值的 2 ^N 倍
26h	02h	整个芯片擦除的最大超时时间为典型值的 2 ^N 倍 (00h = 不支持)

表 62. 底部引导初始供应状态的器件几何定义

字节地址	数据	说明
27h	17h (64 Mb)	器件容量 = 2 ^N 个字节;
28h	02h	闪存器件接口说明;
29h	01h	0000h = 总线宽度为 x8 0001h = 总线宽度为 x16 0002h = 总线宽度为 x8/x16 0003h = 总线宽度为 x32 0004h = 单个 I/O SPI、3 字节地址 0005h = 多个 I/O SPI、3 字节地址 0102h = 多个 I/O SPI、3 或 4 字节地址
2Ah	08h	多字节写操作的最大字节数量 = 2 ^N
2Bh	00h	0000h = 不支持 0008h = 在一个页上编程 256 个字节 0009h = 在一个页上编程 512 个字节
2Ch	03h	器件内擦除块区域的数量 1 = 统一器件, > 1 = 引导器件

表 62. 底部引导初始供应状态的器件几何定义 (续)

字节地址	数据	说明
2Dh	07h	擦除块区域 1 信息 (参见 JEDEC JEP137 规范) 8 扇区 = 8-1 = 0007h 4 KB 扇区 = 256 字节 x 0010h
2Eh	00h	
2Fh	10h	
30h	00h	
31h	00h	擦除块区域 2 的信息 (请参考 JEDEC JEP137 规范) 1 扇区 = 1-1 = 0000h 32 KB 扇区 = 256 字节 x 0080h
32h	00h	
33h	80h	
34h	00h	
35h	7Eh (64 Mb)	擦除块区域 3 信息 127 扇区 = 127-1 = 007Eh (64 Mb)
36h	00h	
37h	00h	
38h	00h	
39h 到 3Fh	FFh	RFU

注意

97. 用户可以配置 FS-S MD 器件, 使它具有混合扇区结构 (具有 8 个 4 KB 扇区, 剩余所有扇区均为 64 KB 或 256 KB), 或者具有相同的扇区结构 (所有扇区均为 64 KB 或 256 KB)。用户也可以按其它方法配置 FS-S 器件, 使它具有位于存储器地址空间顶部的 4 KB 参数扇区。上表中的 CFI 几何信息只与出厂状态有关。所有器件从赛普拉斯出厂时均有混合的扇区结构, 其中 4 KB 扇区位于阵列地址映射中的底部。但可以通过编程器件配置 TBPARM 位 CR1NV[2] 来翻转扇区映射, 使 4 KB 扇区位于阵列地址映射的顶部。可以编程 20h_NV 位 (CR3NV[3]), 从地址映射中移除 4 KB 扇区。闪存器件驱动器软件必须检查 TBPARM 和 20h_NV 位, 以便指出扇区映射是否被翻转, 以及是否移除了混合扇区。

表 63. CFI 主要厂商特定的扩展查询

字节地址	数据	说明
40h	50h	查询唯一 ASCII 字符串 "PRI"
41h	52h	
42h	49h	
43h	31h	主版本号 = 1, ASCII
44h	33h	次版本号 = 3, ASCII
45h	21h	地址敏感解锁 (位 1-0) 00b = 需要, 01b = 不需要 生产技术 (位 5-2) 0000b = 0.23 μm 浮栅 0001b = 0.17 μm 浮栅 0010b = 0.23 μm MirrorBit 0011b = 0.11 μm 浮栅 0100b = 0.11 μm MirrorBit 0101b = 0.09 μm MirrorBit 1000b = 0.065 μm MirrorBit
46h	02h	擦除挂起 0 = 不支持, 1 = 只读, 2 = 读取和编程
47h	01h	扇区保护 00 = 不支持, X = 组中的扇区数量
48h	00h	临时取消扇区保护 00 = 不支持, 01 = 支持
49h	08h	扇区保护 / 不保护方案 04 = 高电压方法 05 = 软件命令锁定方法 08 = 高级扇区保护
4Ah	00h	并发操作 00 = 不支持, X = 扇区数量
4Bh	01h	突发模式 (同步连续读取) 支持 00 = 不支持, 01 = 支持

表 63. CFI 主要厂商特定的扩展查询（续）

字节地址	数据	说明
4Ch	03h	用户可以配置的出厂状态页模式类型（512 B） 00 = 不支持，01 = 4 字读取页，02 = 8 字读取页，03 = 256 字节编程页，04 = 512 字节编程页
4Dh	00h	ACC（加速）最小供电电压 00 = 不支持，100 mV
4Eh	00h	ACC（加速）最大供电电压 00 = 不支持，100 mV
4Fh	07h	WP# 保护 01 = 整个芯片 04 = 带有底部 WP 保护功能的统一容量器件 05 = 带有顶部 WP 保护功能的统一容量器件 07 = 带有顶部或底部写保护功能（用户可配置）的统一容量器件。
50h	01h	编程挂起 00 = 不支持，01 = 支持

供应商特定的备用扩展型查询提供了与 FS-S 系列提供的扩展命令集有关的信息。每个备用查询参数都是从一个标识字节和一个参数长度字节开始的。如果软件不需要某个参数或不能识别它，驱动器软件可以检查每个参数的 ID，并根据长度值来跳转到下一个参数。

表 64. CFI 备用供应商特定的扩展查询

字节地址	数据	说明
51h	41H	查询唯一 ASCII 字符串“ALT”
52h	4Ch	
53H	54H	
54H	32h	主版本号 = 2，ASCII
55H	30h	次版本号 = 0，ASCII

表 65. CFI 备用供应商特定的扩展查询参数 0

参数的相对字节地址偏移	数据	说明
56h	00h	参数 ID（订购器件型号）
57H	10h	参数长度（该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节）
58h	53H	制造商（赛普拉斯）的 Ascii“S”
59h	32h	产品字符（单芯片 SPI）的 Ascii“25”
5Ah	35h	
5Bh	46h	接口字符（SPI 1.8V）的 Ascii“FS”
5Ch	53H	
5Dh	30h（64 Mb）	容量的 Ascii 字符
5Eh	36h（64 Mb）	
5Fh	34h（64 Mb）	
60h	53H	技术（65 nm MirrorBit）的 Ascii“S”
61h	FFh	预留供将来使用
62h	FFh	
63h	FFh	
64h	FFh	预留供将来使用
65h	FFh	
66h	xxh	
67h	xxh	模型的 Ascii 字符请参考第 141 页上的订购器件型号，以了解模型编号的定义。

表 66. CFI 备用供应商特定的扩展查询参数 80h 地址选项

参数的相对字节地址偏移	数据	说明
68h	80h	参数 ID（订购器件型号）
69h	01h	参数长度（该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节）
6Ah	EBh	位 7:5 — 预留 = 111b 位 4 — CR2V[7] 中的地址长度位 — 支持 = 0b 位 3 — 自动启动支持 — 不支持 = 1b 位 2 — 支持 4 字节地址指令 — 支持 = 0b 位 1 — 支持组地址 + 3 字节地址指令 — 不支持 = 1b 位 0 — 支持 3 字节地址指令 — 不支持 = 1b

表 67. CFI 备用供应商特定的扩展查询参数 84h 挂起命令

参数的相对字节地址偏移	数据	说明
6Bh	84h	参数 ID（挂起命令）
6Ch	08h	参数长度（该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节）
6Dh	85h	编程挂起指令代码
6Eh	2Dh	最大编程挂起延迟（uS）
6Fh	8Ah	编程恢复指令代码
70h	64h	从编程恢复到下个挂起指令的典型时间（uS）
71h	75h	擦除挂起指令代码
72h	2Dh	最大擦除挂起延迟（uS）
73h	7Ah	编程恢复指令代码
74h	64h	从擦除恢复到下个挂起指令的典型时间（uS）

表 68. CFI 备用供应商特定的扩展查询参数 88h 数据保护

参数的相对字节地址偏移	数据	说明
75h	88h	参数 ID（数据保护）
76h	04h	参数长度（该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节）
77h	0Ah	OTP 大小为 2^N 字节，FFh = 不支持
78h	01h	OTP 地址映射格式，01h = FL-S 和 FS-S 格式，FFh = 不支持
79h	xxh	块保护类型，由模型决定 00h = FL-P、FL-S、FS-S FFh = 不支持
7Ah	xxh	高级扇区保护类型，由模型决定 01h = FL-S 和 FS-S ASP。

表 69. CFI 备用供应商特定的扩展查询参数 94h ECC

参数的相对字节地址偏移	数据	说明
83h	94h	参数 ID（ECC）
84h	01h	参数长度（该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节）
85h	10h	ECC 单元大小字节，FFh = 使能 ECC

表 70. CFI 备用供应商特定的扩展查询参数 8Ch 复位时序

参数的相对字节地址偏移	数据	说明
7Bh	8Ch	参数 ID（复位时序）
7Ch	06h	参数长度（该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节）
7Dh	96h	POR 最大值
7Eh	01h	指数 2^N （ μ S）的最大 POR 值
7Fh	23h	硬件复位的最大值，FFh = 不支持（初始定义状态禁用了硬件复位功能，但稍后用户可以使能它）
80h	00h	指数 2^N （ μ S）的最大硬件复位值
81h	23h	软件复位的最大值，FFh = 不支持
82h	00h	指数 2^N （ μ S）的最大软件复位值

表 71. CFI 备用供应商特定的扩展查询参数 F0h RFU

参数的相对字节地址偏移	数据	说明
83h	F0h	参数 ID（RFU）
84h	09h	参数长度（该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节）
85h	FFh	RFU
...	FFh	RFU
8Dh	FFh	RFU

该参数类型（参数 ID F0h）可以多次出现，并且每次都有不同的长度。该参数用于保留 ID-CFI 映射中的空间，或强制使空间延伸到所需的边界，从而与下一个参数对齐。

13.3 串行闪存可发现参数（SFDP）地址映射

13.3.1 JEDEC SFDP B 标题表

表 72. SFDP 头文件

SFDP 字节地址	SFDP Dword 名称	数据	说明
00h	SFDP 头文件的第一个 DWORD	53H	读取 SFDP（5Ah）命令的入口点，即为 SFDP 空间中的地址零 ASCII “S”
01h		46H	ASCII “F”
02h		44h	ASCII “D”
03h		50h	ASCII “P”
04h	SFDP 头文件的第二个 DWORD	06h	SFDP 次要版本（06h = JEDEC JESD216 版本 B） 该版本与以前所有次要版本向后兼容。次要版本更改的内容包括：定义之前保留字段、将字段添加到终端或说明现有字段的定义。如果次要版本值递增，则表示已经给之前保留的参数字段分配了新的定义，或者整个 Dwords 已被添加到参数表内。但之前字段的定义未被更改，因此它们仍与旧版本的 SFDP 参数表向后兼容。如果使用软件支持的参数，软件可以安全地忽略次要版本编号的递增，即屏蔽或忽略之前的保留字段和其他 Dword。请勿对次要版本编号进行简单的比较。只需要检查该版本是否匹配软件需要处理的版本编号。完全可以使用更高的次要版本。
05h		01h	SFDP 主要版本 是主要的原始版本。该版本与所有 SFDP 读取和分析软件均兼容。
06h		05h	参数头文件的数量（以零为基准，05h = 6 个参数）
07h		FFh	未使用
08h	参数头文件 0 第一个 DWORD	00h	参数 ID LSB（00h = JEDEC SFDP 基本 SPI 闪存参数）
09h		00h	参数次要版本（00h = JESD216） — 该旧版本参数头文件适用于所有的传统 SFDP 读取和解析软件（该软件需要检查次要版本 0 的参数头文件）。处理更高次要版本的 SFDP 软件将继续读取参数头文件以寻找包含软件版本的其他参数的更高次要版本。
0Ah		01h	参数主要版本（01h = 原始主要版本 — 所有 SFDP 软件均与该版本兼容）。
0Bh		09h	参数表长度（单位为双字 = Dwords = 4 字节）09h = 9 Dwords
0Ch	参数头 0 第二个 DWORD	90h	参数表指针字节 0（Dword = 4 字节对齐） JEDEC 基本 SPI 闪存参数字节偏移 = 1090h
0Dh		10h	参数表指针字节 1
0Eh		00h	参数表指针字节 2
0Fh		FFh	参数 ID MSB（FFh = JEDEC 定义的传统参数 ID）
10h	参数头文件 1 第一个 DWORD	00h	参数 ID LSB（00h = JEDEC SFDP 基本 SPI 闪存参数）
11h		05h	参数次要版本（05h = JESD216 版本 A） — 该旧版本参数头文件适用于所有的传统 SFDP 读取和解析软件（该软件需要检查次要版本 5 的参数头文件）。处理更高次要版本的 SFDP 软件将继续读取参数头文件以寻找包含其他参数的更高次要版本。
12h		01h	参数主要版本（01h = 原始主要版本 — 所有 SFDP 软件均与该版本兼容）
13h		10h	参数表长度（单位为双字 = Dwords = 4 字节）10h = 16 Dwords
14h	参数头文件 1 第二个 DWORD	90h	参数表指针字节 0（Dword = 4 字节对齐） JEDEC 基本 SPI 闪存参数字节偏移 = 1090h 地址
15h		10h	参数表指针字节 1
16h		00h	参数表指针字节 2
17h		FFh	参数 ID MSB（FFh = JEDEC 定义的参数）
18h	参数头文件 2 第一个 DWORD	00h	参数 ID LSB（00h = JEDEC SFDP 基本 SPI 闪存参数）
19h		06h	参数次要版本（06h = JESD216 版本 B）
1Ah		01h	参数主要版本（01h = 原始主要版本 — 所有 SFDP 软件均与该版本兼容）
1Bh		10h	参数表长度（单位为双字 = Dwords = 4 字节）10h = 16 Dwords

表 72. SFDP 头文件（续）

SFDP 字节地址	SFDP Dword 名称	数据	说明
1Ch	参数头文件 2 第二个 DWORD	90h	参数表指针字节 0（Dword = 4 字节对齐） JEDEC 基本 SPI 闪存参数字节偏移 = 1090h 地址
1Dh		10h	参数表指针字节 1
1Eh		00h	参数表指针字节 2
1Fh		FFh	参数 ID MSB（FFh = JEDEC 定义的参数）
20h	参数头文件 3 第一个 DWORD	81h	参数 ID LSB（81h = SFDP 扇区映射参数）
21h		00h	参数次要版本（00h = 初始版本，如 JESD216 版本 B 的定义）
22h		01h	参数主要版本（01h = 原始主要版本 — 识别该参数 ID 的所有 SFDP 软件均与该主要版本兼容）。
23h		1Ah	参数表长度（单位为双字 = Dwords = 4 字节）取决于 OPN 26 = 1Ah
24h	参数头文件 3 第二个 DWORD	D8h	参数表指针字节 0（Dword = 4 字节对齐） JEDEC 参数字节偏移 = 10D8h
25h		10h	参数表指针字节 1
26h		00h	参数表指针字节 2
27h		FFh	参数 ID MSB（FFh = JEDEC 定义的参数）
28h	参数头文件 4 第一个 DWORD	84h	参数 ID LSB（00h = SFDP 4 字节地址指令参数）
29h		00h	参数次要版本（00h = 初始版本，如 JESD216 版本 B 的定义）
2Ah		01h	参数主要版本（01h = 原始主要版本 — 识别该参数 ID 的所有 SFDP 软件均与该主要版本兼容）。
2Bh		02h	参数表长度（单位为双字 = Dwords = 4 字节）（2h = 2 Dwords）
2Ch	参数头文件 4 第二个 DWORD	D0h	参数表指针字节 0（Dword = 4 字节对齐） JEDEC 参数字节偏移 = 10D0h
2Dh		10h	参数表指针字节 1
2Eh		00h	参数表指针字节 2
2Fh		FFh	参数 ID MSB（FFh = JEDEC 定义的参数）
30h	参数头文件 5 第一个 DWORD	01h	参数 ID LSB（赛普拉斯供货商特定 ID-CFI 参数） 传统制造商 ID 01h = AMD / Spansion
31h		01h	参数次要版本（在 SFDP 版本 B 表中，01h = ID-CFI）
32h		01h	参数主要版本（01h = 原始主要版本 — 识别该参数 ID 的所有 SFDP 软件均与该主要版本兼容）。
33h		50h	参数表长度（单位为双字 = Dwords = 4 字节）参数表长度（单位为双字 = Dwords = 4 字节）
34h	参数头文件 5 第二个 DWORD	00h	参数表指针字节 0（Dword = 4 字节对齐） 从 SFDP 的位置 0 算起，ID-CFI 参数入口点的地址偏移 = 1000h。
35h		10h	参数表指针字节 1
36h		00h	参数表指针字节 2
37h		01h	参数 ID MSB（01h = JEDEC JEP106 组编号 1）

13.3.2 JEDEC SFDP B 参数表

从 CFI 数据结构的角度来看，所有 SFDP 参数表与单一 CFI 参数合并成一个连续的字节序列。

从 SFDP 数据结构的角度来看，存在三个独立的参数表。两个表具有固定的长度，另一个表具有可变的结构和长度（具体取决于器件容量的订购型号（OPN）。基本的闪存参数表和 4 字节地址指令参数表具有固定的长度，在下面显示为一个参数表。该表是整个 CFI 参数的第一部分。

JEDEC 扇区映射参数表的结构和长度取决于容量 OPN，并显示为一组表格，每个器件容量一个表。OPN 的相应表格为整个 CFI 参数的第二部分，并且位于第一部分的后面。

表 73. CFI 和 SFDP 的第一 1 部分，基本闪存和 4 字节地址指令参数

CFI 参数的相对 字节地址偏移 (从 108Eh 开始)	SFDP 参数的相对 字节地址偏移 (从 1090h 开始)	SFDP Dword 名称	数据	说明
00h	--	N/A	A5h	CFI 参数 ID (JEDEC SFDP)
01h	--	N/A	B0h	CFI 参数长度 (该参数中随后字节的数量。将该值添加到当前地址值 +1 = 下个参数的第一个字节)。由 OPN 决定: 18Dw + 26Dw = 44Dw * 4B = 176B = B0h B
02h	00h	JEDEC 基本 闪存参数 Dword-1	E7h	SFDP JEDEC 参数的起点，位于整个 SFDP 地址空间中的 1090h 位。 位 7:5 = 未使用 = 111b 位 4:3 = 06h 是状态寄存器写入指令和状态寄存器的非易失性默认值 = 00b 位 2 = 编程缓冲区 > 64 B = 1 位 1:0 = 4 KB 擦除不可用 = 11b
03h	01h		FFh	位 15:8 = 统一 4 KB 擦除操作码 = 不支持 = FFh
04h	02h		FBh	位 23 = 未使用 = 1b 位 22 = 支持四线输出读取 = 有 = 1b 位 21 = 支持四线 I/O 读取 = 有 = 1b 位 20 = 支持双线 I/O 读取 = 有 = 1b 位 19 = 支持 DDR = 有 = 1b ; 位 18:17 = 地址字节的数量，3 或 4 = 01b 位 16 = 支持双线输出读取 = 有 = 1b
05h	03h		FFh	位 31:24 = 未使用 = FFh
06h	04h	JEDEC 基本 闪存参数 Dword-2	FFh	容量 (单位为位，基于零)，16Mb = 00FFFFFFh
07h	05h		FFh	
08h	06h		FFh	
09h	07h		03h (64 Mb)	
0Ah	08h	JEDEC 基本 闪存参数 Dword-3	48h	位 7:5 = 四线 I/O (1-4-4) 模式周期数量 = 010b 位 4:0 = 四线 I/O 虚拟周期数量 = 01000b (初始供应状态)
0Bh	09h		EBh	四线 I/O 指令代码
0Ch	0Ah		08h	位 23:21 = 四线输出 (1-1-4) 模式周期的数量 = 000b 位 20:16 = 四线输出虚拟周期的数量 = 01000b
0Dh	0Bh		6Bh	四线输出指令代码
0Eh	0Ch	JEDEC 基本 闪存参数 Dword-4	08h	位 7:5 = 双线输出 (1-1-2) 模式周期的数量 = 000b 位 4:0 = 双线输出虚拟周期的数量 = 01000b
0Fh	0Dh		3Bh	双线输出指令代码
10h	0Eh		88h	位 23:21 = 双线 I/O (1-2-2) 模式周期的数量 = 100b 位 20:16 = 双线 I/O 虚拟周期的数量 = 01000b (初始供应状态)
11h	0Fh		BBh	双线 I/O 指令代码
12h	10h	JEDEC 基本 闪存参数 Dword-5	FEh	位 7:5 RFU = 111b 位 4 = 支持 QPI = 有 = 1b 位 3:1 RFU = 111b 位 0 = 不支持所有双线命令 = 0b
13h	11h		FFh	位 15:8 = RFU = FFh
14h	12h		FFh	位 23:16 = RFU = FFh
15h	13h		FFh	位 31:24 = RFU = FFh
16h	14h	JEDEC 基本 闪存参数 Dword-6	FFh	位 7:0 = RFU = FFh
17h	15h		FFh	位 15:8 = RFU = FFh
18h	16h		FFh	位 23:21 = 所有双线模式周期的数量 = 111b 位 20:16 = 所有双线虚拟周期的数量 = 11111b
19h	17h		FFh	所有双线指令代码

表 73. CFI 和 SFDP 的第一 1 部分，基本闪存和 4 字节地址指令参数（续）

CFI 参数的相对 字节地址偏移 (从 108Eh 开始)	SFDP 参数的相对 字节地址偏移 (从 1090h 开始)	SFDP Dword 名称	数据	说明	
1Ah	18h	JEDEC 基本 闪存参数 Dword-7	FFh	位 7:0 = RFU = FFh	
1Bh	19h		FFh	位 15:8 = RFU = FFh	
1Ch	1Ah		48h	位 23:21 = QPI 模式周期的数量 = 010b 位 20:16 = QPI 虚拟周期的数量 = 01000b	
1Dh	1Bh		EBh	QPI 模式的四线 I/O（4-4-4）指令代码	
1Eh	1Ch	JEDEC 基本 闪存参数 Dword-8	0Ch	对于混合设计，擦除 1 类大小为 2^N 字节 = 4 KB = 0Ch（出厂状态）	
1Fh	1Dh		20h	擦除 1 类指令	
20h	1Eh		10h	擦除 2 类大小为 2^N 字节 = 64 KB = 10h	
21h	1Fh		D8h	擦除 2 类指令	
22h	20h	JEDEC 基本 闪存参数 Dword-9	12h	擦除 3 类大小为 2^N 字节 = 256 KB = 12h	
23h	21h		D8h	擦除 3 类指令	
24h	22h		00h	擦除 4 类大小为 2^N 字节 = 不支持 = 00h	
25h	23h		FFh	擦除 4 类指令 = 不支持 = FFh	
26h	24h	JEDEC 基本 闪存参数 Dword-10	B1h	位 31:30 = 擦除 4 类擦除，典型时间单位（00b: 1 ms、01b: 16 ms、10b: 128 ms、11b: 1 s）= 1S = 11b（RFU）	
27h	25h		72h	位 29:25 = 擦除 4 类擦除，典型时间计数 = 11111b（RFU）	
28h	26h		1Dh	位 24:23 = 擦除 3 类擦除，典型时间单位（00b: 1 ms、01b: 16 ms、10b: 128 ms、11b: 1 s）= 128mS = 10b	
29h	27h		FFh	位 22:18 = 擦除 3 类擦除，典型时间计数 = 00111b（典型擦除时间 = 计数值 +1 * 单位 = 8*128 mS = 1024 mS）	
				位 17:16 = 擦除 2 类擦除，典型时间单位（00b: 1 ms、01b: 16 ms、10b: 128 ms、11b: 1 s）= 16mS = 01b	
				位 15:11 = 擦除 2 类擦除，典型时间计数 = 01110b（典型擦除时间 = 计数值 +1 * 单位 = 15*16 mS = 240 mS）	
				位 10:9 = 擦除 1 类擦除，典型时间单位（00b: 1 ms、01b: 16 ms、10b: 128 ms、11b: 1 s）= 16mS = 01b	
				位 8:4 = 擦除 1 类擦除，典型时间计数 = 01011b（典型擦除时间 = 计数值 +1 * 单位 = 12*16mS = 192mS）	
				位 3:0 = 典型的擦除时间与最长擦除时间之间的倍数 = 2*(N+1)，N=2h = 4x	
				二进制字段：11-11111-10-00111-01-01110-01-01011-0001 半字节格式：1111_1111_0001_1101_0111_0010_1011_0001 十六进制格式：FF_1D_72_B1	
2Ah	28h			82h	位 31 预留 = 1b
2Bh	29h			26h	位 30:29 = 芯片擦除，典型时间单位（00b: 16 ms、01b: 256 ms、10b: 4 s、11b: 64 s）= 64 Mb = 4s = 10b；
2Ch	2Ah			07h	位 28:24 = 芯片擦除，典型时间计数，（计数值 +1）* 单位， 64 Mb = 00111b = (7+1)*4 = 32s；
2Dh	2Bh	JEDEC 基本 闪存参数 Dword-11	C7h	位 23 = 字节编程的典型时间，其他字节单位（0b:1uS、1b:8uS）= 1 uS = 0b	
				位 22:19 = 字节编程的典型时间，附加字节计数，（计数值 +1）* 单位，计数值 = 0000b，（典型编程时间 = 计数值 +1 * 单位 = 1*1uS = 1uS）	
				位 18 = 字节编程的典型时间，第一个字节单位（0b:1uS、1b:8uS）= 8 uS = 1b	
				位 17:14 = 字节编程的典型时间，第一个字节计数，（计数值 +1）* 单位，计数值 = 1100b，（典型编程时间 = 计数值 +1 * 单位 = 13*8uS = 104 uS	
				位 13 = 页编程的典型时间单位（0b: 8uS、1b:64uS）= 64 uS = 1b	
				位 12:8 = 页编程的典型时间计数，（计数值 +1）* 单位，计数值 = 00110b，（典型编程时间 = 计数值 +1 * 单位 = 6*64uS = 384 uS）	
				位 7:4 = 页大小为 2^N，N=8h，= 256B 页	
				位 3:0 = 典型页编程时间与最大页编程时间之间的倍数 = 2*（N+1），N=2h = 6x	
				64 Mb 二进制字段：1-10-00111-0-0000-1-1100-1-00110-1000-0010 半字节格式：1100_0111_0000_0111_0010_0110_1001_0010 十六进制格式：C7_07_26_82	

表 73. CFI 和 SFDP 的第一 1 部分，基本闪存和 4 字节地址指令参数（续）

CFI 参数的相对 字节地址偏移 (从 108Eh 开始)	SFDP 参数的相对 字节地址偏移 (从 1090h 开始)	SFDP Dword 名称	数据	说明
2Eh	2Ch	JEDEC 基本 闪存参数 Dword-12	ECh	位 31 = 支持挂起和恢复 = 0b
2Fh	2Dh		93h	位 30:29 = 正在执行挂起操作擦除最长延迟单位 (00b: 128 ns、01b: 1 us、10b: 8 us、11b: 64 us) = 8 us = 10b
30h	2Eh		18h	位 28:24 = 正在执行挂起操作擦除最长延迟计数 = 00101b, 最长擦除挂起延迟 = 计数值 +1 * 单位 = 6*8uS = 48uS
31h	2Fh		45h	位 23:20 = 擦除恢复到挂起间隔计数 = 0001b, 间隔 = 计数值 +1 * 64 us = 2 * 64 us = 128 us 位 19:18 = 正在执行挂起操作编程最长延迟单位 (00b: 128 ns、01b: 1 us、10b: 8 us、11b: 64 us) = 8 us = 10b 位 17:13 = 正在执行挂起操作编程最长延迟计数 = 00101b, 最长擦除挂起延迟 = 计数值 +1 * 单位 = 6*8uS = 48uS 位 12:9 = 编程恢复到挂起间隔计数 = 0001b, 间隔 = 计数值 +1 * 64 us = 2 * 64 us = 128 us 位 8 = RFU = 1b 位 7:4 = 擦除挂起过程中禁用各种操作 = xxx0b: 可能不会启动任何新的擦除操作 (不允许擦除嵌套) + xx1xb: 可能不会在擦除挂起扇区的空间内启动任何页编程操作 + x1xxb: 可能不会在擦除挂起扇区空间内启动读操作 + 1xxxb: 位 5:4 中的擦除和编程限制充足 = 1100b 位 3:0 = 编程挂起过程中禁用各种操作 = xxx0b: 可能不会启动任何新的擦除操作 (不允许擦除嵌套) + xx0xb: 可能不会启动任何新的页编程操作 (不允许编程嵌套) + x1xxb: 可能不会在编程挂起页空间内启动读操作 + 1xxxb: 位 1:0 中的擦除和编程限制充足 = 1100b 二进制字段: 0-10-00101-0001-10-00100-1001-1-1110-1100 半字节格式: 0100_0101_0001_1000_1001_0011_1110_1100 十六进制格式: 45_18_93_EC
32h	30h	JEDEC 基本 闪存参数 Dword-13	8Ah	位 31:24 = 擦除挂起指令 = 75h 位 23:16 = 擦除恢复指令 = 7Ah 位 15:8 = 编程挂起指令 = 85h 位 7:0 = 编程恢复指令 = 8Ah
33h	31h		85h	
34h	32h		7Ah	
35h	33h		75h	
36h	34h	JEDEC 基本 闪存参数 Dword-14	F7h	位 31 = 支持深度掉电模式 = 支持 = 0 位 30:23 = 进入深度掉电指令 = B9h 位 22:15 = 退出深度掉电指令 = ABh 位 14:13 = 退出深度掉电到下一个操作的延迟单位 = (00b: 128 ns、01b: 1 us、10b: 8 us、11b: 64 us) = 1 us = 01b 位 12:8 = 退出深度掉电直到下一个操作的延迟计数 = 11101b, 退出深度掉电直到下一个操作的延迟 = (计数值 +1) * 单位 = 29+1 * 1us = 30 us 位 7:4 = RFU = Fh 位 3:2 = 状态寄存器轮询器件忙碌 = 01b: 支持传统状态轮询 = 发出 05h 指令来读取状态寄存器并检查 WIP 位 [0] (0 = 就绪; 1 = 忙碌), 以进行传统轮询。 = 01b 位 1:0 = RFU = 11b 二进制字段: 0-10111001-10101011-01-11101-1111-01-11 半字节格式: 0101_1100_1101_0101_1011_1101_1111_0111 十六进制格式: 5C_D5_BD_F7
37h	35h		BDh	
38h	36h		D5h	
39h	37h		5Ch	

表 73. CFI 和 SFDP 的第一 1 部分，基本闪存和 4 字节地址指令参数（续）

CFI 参数的相对 字节地址偏移 (从 108Eh 开始)	SFDP 参数的相对 字节地址偏移 (从 1090h 开始)	SFDP Dword 名称	数据	说明
3Ah	38h	JEDEC 基本 闪存参数 Dword-15	8Ch	位 31:24 = RFU = FFh
3Bh	39h		F6h	位 23 = 保持和 WP 禁用 = 不支持 = 0b
3Ch	3Ah		5Dh	位 22:20 = 四线使能要求 = 101b QE 是状态寄存器 2 的位 1。通过读取状态指令 05h 读取状态寄存器 1。 通过指令 35h 读取状态寄存器 2。通过带有两个数据字节（其中的第一个字节为 “1”）的写入状态指令 01h 设置 QE。通过两个数据字节（其中的第一个字节 为“0”）的擦除状态将其清零。
3Dh	3Bh		FFh	位 19:16 0-4-4 模式进入方法 = xxx1b: 模式位 [7:0] = A5h 注意：使用该模式前，需要设置 QE + x1xxb: 模式位 [7:0] = Axh + 1xxxb: RFU = 1101b 位 15:10 0-4-4 模式退出方法 = xx_xxx1b: 模式位 [7:0] = 00h 在当前读取操作结束时，将终止该模式 + xx_1xxxb: 8 个周期内在 DQ0-DQ3 上输入 Fh（模式位复位）。这样会使模式 在执行下一个读操作前被终止。 + x1_xxxxb: 模式位 [7:0] != Axh + 1x_x1xx: RFU = 11_1101 位 9 = 支持 0-4-4 模式 = 1 位 8:4 = 4-4-4 模式使能序列 = x_1xxxb: 器件使用读取 - 修改 - 写入操作序列：使用地址 800003h 前面的指 令 65h 读取配置，设置位 6，使用地址 800003h 前面的命令 71h 写入配置。该 配置是易失性的。 = 01000b 位 3:0 = 4-4-4 模式禁用序列 = x1xxb: 器件使用读取 - 修改 - 写入操作序列：使用地址 800003h 前面的指令 65h 读取配置，设置位 6，使用地址 800003h 前面的指令 71h 写入配置 ... 该配 置是易失性的。 + 1xxxb: 发送软复位 66/99 序列 = 1100b 二进制字段：11111111-0-101-1101-111101-1-01000-1100 半字节格式：1111_1111_0101_1101_1111_0110_0010_0010 十六进制格式：FF_5D_F6_8C
3Eh	3Ch	JEDEC 基本 闪存参数 Dword-16	F0h	位 31:24 = 进入 4 字节地址模式
3Fh	3Dh		30h	= xxxx_xxx1b: 发送指令 B7h（无需事先使能写入）
40h	3Eh		F8h	+ xx1x_xxxxb: 支持专用的 4 字节地址指令集。请查看供应商数据手册以了解指 令集的定义。 + 1xxx_xxxxb: 预留 = 10100001b
41H	3Fh		A1h	位 23:14 = 退出 4 字节地址模式 = xx_xx1x_xxxxb: 硬件复位 + xx_x1xx_xxxxb: 软件复位（请参见该 DWORD 中的位 13:8） + xx_1xxx_xxxxb: 电源循环 + x1_xxxx_xxxxb: 预留 + 1x_xxxx_xxxxb: 预留 = 11_1110_0000b 位 13:8 = 支持软复位和挽救序列 = x1_xxxxb: 发出复位使能指令 66h，然后发出复位指令 99h。复位将 根据器件的操作模式被使能，复位序列可能在 1、2 或 4 条线上被发送。 + 1x_xxxxb: 如果器件运行于 0-4-4 模式，则需要预先退出该模式，然后才能进 行上述其他复位序列。 = 1100000b 位 7 = RFU = 1 位 6:0 = 易失性或非易失性寄存器和状态寄存器 1 的写使能指令 = + xx1_xxxxb: 状态寄存器 1 包含易失性和非易失性位。通过 06h 指令可以对该寄存器进行写操作。 + x1x_xxxxb: 预留 + 1xx_xxxxb: 预留 = 1110000b 二进制字段：10100001-1111100000-110000-1-1110000 半字节格式：1010_0001_1111_1000_0011_0000_1111_0000 十六进制格式：A1_F8_30_F0

表 73. CFI 和 SFDP 的第一 1 部分，基本闪存和 4 字节地址指令参数（续）

CFI 参数的相对 字节地址偏移 (从 108Eh 开始)	SFDP 参数的相对 字节地址偏移 (从 1090h 开始)	SFDP Dword 名称	数据	说明
42h	40h	JEDEC 4 字节地址 指令参数 Dword-1	FFh	支持 = 1，不支持 = 0
43h	41h		CEh	位 31:20 = RFU = FFFh
44h	42h		FFh	位 19 = 支持非易失性单独扇区锁定写入指令，指令 = E3h = 1 位 18 = 支持非易失性单独扇区锁定读取指令，指令 = E2h = 1 位 17 = 支持易失性单独扇区锁定写入指令，指令 = E1h = 1 位 16 = 支持易失性单独扇区锁定读取指令，指令 = E0h = 1
45h	43h		FFh	位 15 = 支持 (1-4-4) DTR_Read 命令，指令 = EEh = 1 位 14 = 支持 (1-2-2) DTR_Read 命令，指令 = BEh = 0 位 13 = 支持 (1-1-1) DTR_Read 命令，指令 = 0Eh = 0 位 12 = 支持擦除命令 - 4 类 = 0 位 11 = 支持擦除命令 - 3 类 = 1 位 10 = 支持擦除命令 - 2 类 = 1 位 9 = 支持擦除命令 - 1 类 = 1 位 8 = 支持 (1-4-4) 页编程命令，指令 = 3Eh = 0 位 7 = 支持 (1-1-4) 页编程命令，指令 = 34h = 1 位 6 = 支持 (1-1-1) 页编程命令，指令 = 12h = 1 位 5 = 支持 (1-4-4) FAST_READ 命令，指令 = ECh = 1 位 4 = 支持 (1-1-4) FAST_READ 命令，指令 = 6Ch = 1 位 3 = 支持 (1-2-2) FAST_READ 命令，指令 = BCh = 1 位 2 = 支持 (1-1-2) FAST_READ 命令，指令 = 3Ch = 1 位 1 = 支持 (1-1-1) FAST_READ 命令，指令 = 0Ch = 1 位 0 = 支持 (1-1-1) READ 命令，指令 = 13h = 1
46h	44h	JEDEC 4 字节地址 指令参数 Dword-2	21h	位 31:24 = FFh = 用于擦除 4 类的指令：RFU 位 23:16 = DCh = 用于擦除 3 类的指令 位 15:8 = DCh = 用于擦除 2 类的指令 位 7:0 = 21h = 用于擦除 1 类的指令
47h	45h		DCh	
48h	46h		DCh	
49h	47h		FFh	

扇区映射参数表的注意事项:

下面的扇区映射参数表提供了一个工具，用于识别器件地址映射的配置方式，并且为每个受支持的配置提供一个扇区映射。为了实现该目的，需要一个命令序列，用于读取相关配置寄存器中影响地址映射选择的位。必须读取多个配置位时，所有位均被连接到一个索引值，用于选择当前地址映射。

要想识别 S25FS064S 中的扇区映射配置，需要按照从 MSB 到 LSB 的顺序读取以下配置位，从而组成配置映射索引值：

- CR3NV[3] — 0 = 混合结构，1 = 相同结构
- CR1NV[2] — 0 = 底部的 4 KB 参数扇区，1 = 顶部的 4 KB 扇区
- CR3NV[1] — 0 = 扇区的相同尺寸为 64 KB，1 = 扇区的相同尺寸为 256 KB

有些配置位值会使其它配置位的值不再相关（无需关注），因此并不是所有可能的索引值的组合定义的都是有效的地址映射。SFDP 扇区映射参数表只支持选定的配置位组合。使用该 SFDP 参数表来确定扇区映射时，在配置扇区地址映射过程中，不能使用其它组合。受支持的索引值组合如下。

器件	CR3NV[3]	CR1NV[2]	CR3NV[1]	索引值	说明
FS64S	0	0	0	00h	底部的 4 KB 扇区和剩余的 64 KB 扇区
	0	1	0	02h	顶部的 4 KB 扇区和剩余的 64 KB 扇区
	0	0	1	01h	底部的 4 KB 扇区和剩余的 256 KB 扇区
	0	1	1	03h	顶部的 4 KB 扇区和剩余的 256 KB 扇区
	1	0	0	04h	各扇区的空间统一为 64 KB
	1	0	0	05h	各扇区的空间统一为 256 KB

表 74. CFI 和 SFDP 的第二部分，扇区映射参数表

CFI 参数的相对字节地址偏移	SFDP 参数的相对字节地址偏移	SFDP Dword 名称	数据	说明
4Ah	48h	JEDEC 扇区映射参数 Dword-1 配置检测 1	FCh	位 31:24 = 读取数据屏蔽 = 0000_1000b: 为 20h_NV 值选择数据字节的位 3 0= 带有 4 KB 参数扇区的混合映射 1= 统一映射 位 23:22 = 配置检测命令地址长度 = 11b: 可变长度 位 21:20 = RFU = 11b 位 19:16 = 配置检测命令延迟 = 1111b: 可变延迟 位 15:8 = 配置检测指令 = 65h: 读取任何寄存器 位 7:2 = RFU = 111111b 位 1 = 命令描述符 = 0 位 0 = 非最终描述符 = 0
4Bh	49h		65h	
4Ch	4Ah		FFh	
4Dh	4Bh		08h	
4Eh	4Ch	JEDEC 扇区映射参数 Dword-2 配置检测 1	04h	位 31:0 = 扇区映射配置检测命令地址 = 00_00_00_04h: CR3NV 的地址
4Fh	4Dh		00h	
50h	4Eh		00h	
51h	4Fh		00h	
52h	50h	JEDEC 扇区映射参数 Dword-3 配置检测 2	FCh	位 31:24 = 读取数据屏蔽 = 0000_0100b: 为 TBPARM_O 值选择数据字节的位 2 0= 底部的 4 KB 参数扇区 1= 顶部的 4 KB 参数 位 23:22 = 配置检测命令地址长度 = 11b: 可变长度 位 21:20 = RFU = 11b 位 19:16 = 配置检测命令延迟 = 1111b: 可变延迟 位 15:8 = 配置检测指令 = 65h: 读取任何寄存器 位 7:2 = RFU = 111111b 位 1 = 命令描述符 = 0 位 0 = 非最终描述符 = 0
53H	51h		65h	
54H	52h		FFh	
55H	53H		04h	
56h	54H	JEDEC 扇区映射参数 Dword-4 配置检测 2	02h	位 31:0 = 扇区映射配置检测命令地址 = 00_00_00_02h: CR1NV 的地址
57H	55H		00h	
58h	56h		00h	
59h	57H		00h	

表 74. CFI 和 SFDP 的第二部分，扇区映射参数表（续）

CFI 参数的相对字节地址偏移	SFDP 参数的相对字节地址偏移	SFDP Dword 名称	数据	说明
5Ah	58h	JEDEC 扇区映射参数 Dword-5 配置检测 3	FDh	位 31:24 = 读取数据屏蔽 = 0000_0010b: 为 D8h_NV 值选择数据字节的位 1 0 = 64 KB 统一扇区 1 = 256 KB 统一扇区 位 23:22 = 配置检测命令地址长度 = 11b: 可变长度 位 21:20 = RFU = 11b 位 19:16 = 配置检测命令延迟 = 1111b: 可变延迟 位 15:8 = 配置检测指令 = 65h: 读取任何寄存器 位 7:2 = RFU = 111111b 位 1 = 命令描述符 = 0 位 0 = 最终描述符 = 1
5Bh	59h		65h	
5Ch	5Ah		FFh	
5Dh	5Bh		02h	
5Eh	5Ch	JEDEC 扇区映射参数 Dword-6 配置检测 3	04h	位 31:0 = 扇区映射配置检测命令地址 = 00_00_00_04h: CR3NV 的地址
5Fh	5Dh		00h	
60h	5Eh		00h	
61h	5Fh		00h	
62h	60h	JEDEC 扇区映射参数 Dword-7 配置 0 头部分	FEh	位 31:24 = RFU = FFh 位 23:16 = 区域计数 (Dwords -1) = 02h: 三个区域 位 15:8 = 配置 ID = 00h: 底部上的 4 KB 扇区和剩余的 64 KB 扇区 位 7:2 = RFU = 111111b 位 1 = 映射描述符 = 1 位 0 = 非最终描述符 = 0
63h	61h		00h	
64h	62h		02h	
65h	63h		FFh	
66h	64h	JEDEC 扇区映射参数 Dword-8 配置 0 区域 0	F1h	位 31:8 = 区域大小 = 00007Fh: 区域大小等于计数 - 1 (单位为 256 字节) = 8 x 4 KB 扇区 = 32 KB 计数值 = 32KB/256 = 128, 值 = 计数值 -1 = 128 -1 = 127 = 7Fh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 4 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 4 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 1b --- 擦除 1 类为 4 KB 擦除且在 4 KB 扇区区域中得到支持
67h	65h		7Fh	
68h	66h		00h	
69h	67h		00h	
6Ah	68h	JEDEC 扇区映射参数 Dword-9 配置 0 区域 1	F2h	位 31:8 = 区域大小 = 00007Fh: 区域大小等于计数 - 1 (单位为 256 字节) = 8 x 4 KB 扇区 = 32 KB 计数值 = 32KB/256 = 128, 值 = 计数值 -1 = 128 -1 = 127 = 7Fh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 32 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 1b --- 擦除 2 类为 64 KB 擦除且在 32 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 32 KB 扇区区域中得到支持
6Bh	69h		7Fh	
6Ch	6Ah		00h	
6Dh	6Bh		00h	

表 74. CFI 和 SFDP 的第二部分，扇区映射参数表（续）

CFI 参数的相对字节地址偏移	SFDP 参数的相对字节地址偏移	SFDP Dword 名称	数据	说明
6Eh	6Ch	JEDEC 扇区映射参数 Dword-10 配置 0 区域 2	F2h	位 31:8 = 64 Mb 器件的区域大小 = 007EFFh: 区域大小等于计数 - 1（单位为 256 字节）= 127x 65536B 扇区 = 8323072B 计数值 = 8323072B/256 = 32512, 值 = 计数值 -1 = 32512-1 = 32511= 7EFFh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256KB 擦除并且在 64 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 1b --- 擦除 2 类为 64 KB 擦除且在 64 KB 扇区区域中受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 64 KB 扇区区域中不受支持
6Fh	6Dh		FFh	
70h	6Eh		7Eh (64 Mb)	
71h	6Fh		00h	
72h	70h	JEDEC 扇区映射参数 Dword-11 配置 2 头文件	FEh	位 31:24 = RFU = FFh 位 23:16 = 区域计数 (Dwords -1) = 02h: 三个区域 位 15:8 = 配置 ID = 00h: 顶部上的 4 KB 扇区和剩余的 64KB 扇区 位 7:2 = RFU = 111111b 位 1 = 映射描述符 = 1 位 0 = 非最终描述符 = 0
73h	71h		02h	
74h	72h		02h	
75h	73h		FFh	
76h	74h	JEDEC 扇区映射参数 Dword-12 配置 2 区域 0	F2h	位 31:8 = 64Mb 器件的区域大小 = 007EFFh: 区域大小等于计数 - 1（单位为 256 字节）= 127x 65536B 扇区 = 8323072B 计数值 = 8323072B/256 = 32512, 值 = 计数值 -1 = 32512-1 = 32511= 7EFFh 位 7:4 = RFU = Fh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 64 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 1b --- 擦除 2 类为 64 KB 擦除且在 64 KB 扇区区域中受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 64 KB 扇区区域中不受支持
77h	75h		FFh	
78h	76h		7Eh (64 Mb)	
79h	77h		00h	
7Ah	78h	JEDEC 扇区映射参数 Dword-13 配置 2 区域 1	F2h	位 31:8 = 区域大小 = 00007Fh: 区域大小等于计数 - 1（单位为 256 字节）= 8 x 4 KB 扇区 = 32 KB 计数值 = 32KB/256 = 128, 值 = 计数值 -1 = 128 -1 = 127 = 7Fh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 32 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 1b --- 擦除 2 类为 64 KB 擦除且在 32KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 32KB 扇区区域中得到支持
7Bh	79h		7Fh	
7Ch	7Ah		00h	
7Dh	7Bh		00h	
7Eh	7C	JEDEC 扇区映射参数 Dword-14 配置 2 区域 2	F1h	位 31:8 = 区域大小 = 00007Fh: 区域大小等于计数 - 1（单位为 256 字节）= 8 x 4 KB 扇区 = 32 KB 计数值 = 32KB/256 = 128, 值 = 计数值 -1 = 128 -1 = 127 = 7Fh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 4 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 4 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 1b --- 擦除 1 类为 4 KB 擦除且在 4 KB 扇区区域中得到支持
7Fh	7D		7Fh	
80h	7E		00h	
81h	7F		00h	

表 74. CFI 和 SFDP 的第二部分，扇区映射参数表（续）

CFI 参数的相对字节地址偏移	SFDP 参数的相对字节地址偏移	SFDP Dword 名称	数据	说明
82h	80h	JEDEC 扇区映射参数 Dword-15 配置 1 头文件	FEh	位 31:24 = RFU = FFh
83h	81h		01h	位 23:16 = 区域计数 (Dwords -1) = 02h: 三个区域
84h	82h		02h	位 15:8 = 配置 ID = 00h: 底部上的 4KB 扇区和剩余的 256 KB 扇区
85h	83h		FFh	位 7:2 = RFU = 111111b 位 1 = 映射描述符 = 1 位 0 = 非最终描述符 = 0
86h	84h	JEDEC 扇区映射参数 Dword-16 配置 1 区域 0	F1h	位 31:8 = 区域大小 = 00007Fh:
87h	85h		7Fh	区域大小等于计数 - 1 (单位为 256 字节) = 8 x 4 KB 扇区 = 32 KB
88h	86h		00h	计数值 = 32KB/256 = 128, 值 = 计数值 -1 = 128 -1 = 127 = 7Fh 位 7:4 = RFU = Fh
89h	87h		00h	不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 4 KB 扇区区域中受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 4 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 1b --- 擦除 1 类为 4 KB 擦除且在 4 KB 扇区区域中得到支持
8Ah	88h	JEDEC 扇区映射参数 Dword-17 配置 1 区域 1	F4h	位 31:8 = 区域大小 = 00037Fh:
8Bh	89h		7Fh	区域大小等于计数 - 1 (单位为 256 字节) = 8 x 4 KB 扇区 = 32 KB
8Ch	8Ah		03h	计数值 = 224KB/256 = 896, 值 = 计数值 -1 = 896 -1 = 895 = 37Fh 位 7:4 = RFU = Fh
8Dh	8Bh		00h	不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 1b --- 擦除 3 类为 256 KB 擦除并且在 224 KB 扇区区域中受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 224 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4KB 擦除且在 224 KB 扇区区域中不受支持
8Eh	8Ch	JEDEC 扇区映射参数 Dword-18 配置 1 区域 2	F4h	位 31:8 = 64 Mb 器件的区域大小 = 007BFFh:
8Fh	8Dh		FFh	区域大小等于计数 - 1 (单位为 256 字节) = 8 x 4 KB 扇区 = 32 KB
90h	8Eh		7Bh (64 Mb)	计数值 = 8126464B/256 = 31744, 值 = 计数值 -1 = 31744 -1 = 31743 = 7BFFh 位 7:4 = RFU = Fh
91h	8Fh		00h	不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 1b --- 擦除 3 类为 256 KB 擦除并且在 256 KB 扇区区域中受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 256 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 256 KB 扇区区域中不受支持
92h	90h	JEDEC 扇区映射参数 Dword-19 配置 3 头文件	FEh	位 31:24 = RFU = FFh
93h	91h		03h	位 23:16 = 区域计数 (Dwords -1) = 02h: 三个区域
94h	92h		02h	位 15:8 = 配置 ID = 03h: 顶部上的 4KB 扇区和剩余的 256 KB 扇区
95h	93h		FFh	位 7:2 = RFU = 111111b 位 1 = 映射描述符 = 1 位 0 = 非最终描述符 = 0

表 74. CFI 和 SFDP 的第二部分，扇区映射参数表（续）

CFI 参数的相对字节地址偏移	SFDP 参数的相对字节地址偏移	SFDP Dword 名称	数据	说明
96h	94h	JEDEC 扇区映射参数 Dword-20 配置 3 区域 0	F4h	位 31:8 = 64 Mb 器件的区域大小 = 007BFFh: 区域大小等于计数 - 1（单位为 256 字节）= 8 x 4KB 扇区 = 32 KB 计数值 = 8126464B/256 = 31744, 值 = 计数值 -1 = 31744-1 = 31743= 7BFFh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 1b --- 擦除 3 类为 256 KB 擦除并且在 256 KB 扇区区域中受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 256 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 256 KB 扇区区域中不受支持
97h	95h		FFh	
98h	96h		7Bh (64 Mb)	
99H	97h		00h	
9Ah	98h	JEDEC 扇区映射参数 Dword-21 配置 3 区域 1	F4h	位 31:8 = 区域大小 = 00037Fh: 区域大小等于计数 - 1（单位为 256 字节）= 8 x 4KB 扇区 = 32 KB 计数值 = 224KB/256 = 896, 值 = 计数值 -1 = 896 -1 = 895 = 37Fh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 1b --- 擦除 3 类为 256 KB 擦除并且在 224KB 扇区区域中受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 224KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 224KB 扇区区域中不受支持
9Bh	99H		7Fh	
9Ch	9Ah		03h	
9Dh	9Bh		00h	
9Eh	9Ch	JEDEC 扇区映射参数 Dword-22 配置 3 区域 2	F1h	位 31:8 = 区域大小 = 00007Fh: 区域大小等于计数 - 1（单位为 256 字节）= 8 x 4KB 扇区 = 32 KB 计数值 = 32KB/256 = 128, 值 = 计数值 -1 = 128 -1 = 127 = 7Fh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 4 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 4 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 1b --- 擦除 1 类为 4 KB 擦除且在 4 KB 扇区区域中得到支持
9Fh	9Dh		7Fh	
A0h	9Eh		00h	
A1h	9Fh		00h	
A2h	A0h	JEDEC 扇区映射参数 Dword-23 配置 4 头文件	FEh	位 31:24 = RFU = FFh 位 23:16 = 区域计数（Dwords -1）= 00h: 一个区域 位 15:8 = 配置 ID = 04h: 扇区容量统一为 64 KB 位 7:2 = RFU = 111111b 位 1 = 映射描述符 = 1 位 0 = 非最终描述符 = 0
A3h	A1h		04h	
A4h	A2h		00h	
A5h	A3h		FFh	
A6h	A4h	JEDEC 扇区映射参数 Dword-24 配置 4 区域 0	F2h	位 31:8 = 64 Mb 器件的区域大小 = 007FFBh: 区域大小等于计数 - 1（单位为 256 字节）= 127x 65536B 扇区 = 8323072B 计数值 = 8388608B/256 = 32768, 值 = 计数值 -1 = 32768-1 = 32767= 7FFFh 位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 0b --- 擦除 3 类为 256 KB 擦除并且在 64 KB 扇区区域中不受支持 位 1 = 支持擦除 2 类 = 1b --- 擦除 2 类为 64 KB 擦除且在 64 KB 扇区区域中受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 64 KB 扇区区域中不受支持
A7h	A5h		FFh	
A8h	A6h		7Fh (64 Mb)	
A9h	A7h		00h	

表 74. CFI 和 SFDP 的第二部分，扇区映射参数表（续）

CFI 参数的相对字节地址偏移	SFDP 参数的相对字节地址偏移	SFDP Dword 名称	数据	说明
AAh	A8h	JEDEC 扇区映射参数 Dword-25 配置 5 头文件	FFh	位 31:24 = RFU = FFh
ABh	A9h		05h	位 23:16 = 区域计数 (Dwords -1) = 00h: 一个区域
ACh	AAh		00h	位 15:8 = 配置 ID = 05h: 扇区容量统一为 256 KB
ADh	ABh		FFh	位 7:2 = RFU = 111111b 位 1 = 映射描述符 = 1 位 0 = 最终描述符 = 1
A Eh	A Ch	JEDEC 扇区映射参数 Dword-26 配置 5 区域 0	F4h	位 31:8 = 64Mb 器件的区域大小 = 01FFFFh:
A Fh	A Dh		FFh	区域大小等于计数 - 1 (单位为 256 字节) = 127x 65536B 扇区 = 8323072B
B0h	A Eh		7Fh (64 Mb)	计数值 = 8388608B/256 = 32768, 值 = 计数值 -1 = 32768-1 = 32767= 7FFFFh
B1h	A Fh		00h	位 7:4 = RFU = Fh 不支持擦除类型 = 0/ 支持 = 1 位 3 = 支持擦除 4 类 = 0b --- 未定义擦除 4 类 位 2 = 支持擦除 3 类 = 1b --- 擦除 3 类为 256 KB 擦除并且在 256 KB 扇区区域中受支持 位 1 = 支持擦除 2 类 = 0b --- 擦除 2 类为 64 KB 擦除且在 256 KB 扇区区域中不受支持 位 0 = 支持擦除 1 类 = 0b --- 擦除 1 类为 4 KB 擦除且在 256 KB 扇区区域中不受支持

13.4 出厂状态

该器件由赛普拉斯生产，其非易失性位如下设置：

- 整个存储器阵列被擦除：例如，所有位均被设为 1（因此每个字节都包含 FFh）。
- OTP 地址空间将前 16 个字节编程为随机数。所有其它字节均被擦除为 FFh。
- SFDP 地址空间包含如 SFDP 地址空间的说明部分中所定义的值。
- ID-CF 地址空间包含如 ID-CF 地址空间的说明部分中所定义的值。
- RUID 地址空间包含 64 位唯一的 ID 号。
- 状态寄存器 1 非易失性包含 00h（全部 SR1NV 位均被清除为 ‘0’）。
- 配置寄存器 1 非易失性包含 00h。
- 配置寄存器 2 非易失性包含 00h。
- 配置寄存器 3 非易失性包含 00h。
- 配置寄存器 4 非易失性包含 10h。
- 密码寄存器包含 FFFFFFFF-FFFFFFFh。
- 所有 PPB 位均为 “1”。
- ASP 寄存器位为 FFFFh。

14. 订购器件型号

下面介绍了订购器件编码的有效组合：

S25FS	064	S	AG	M	F	I	00	1	
									包装类型
									0 = 托盘
									1 = 包装管
									3 = 13" 盘带封装
									型号（其他订购选项）
									01 = SOIC8 封装
									02 = 5x5 ball BGA 封装 FAB
									03 = LGA 封装
									温度范围 / 级别
									I = 工业级（-40 °C 到 +85 °C）
									V = 扩展的工业级（-40°C 到 +105°C）
									N = 扩展范围（-40°C ~ +125°C）
									A = 汽车级，AEC-Q100 等级 3（-40°C 到 +85°C）
									B = 汽车级，AEC-Q100 等级 2（-40°C 到 +105°C）
									M = 汽车级，AEC-Q100 等级 1（-40°C 到 +125°C）
									封装材料
									F = 无卤素，无铅（Pb）
									F = 无卤素，无铅（Pb）
									封装类型
									M = 8 引脚 SOIC
									N = 8 接触点 LGA
									B = 24 ball BGA 6 x 8 mm 封装，间距为 1.00 mm
									速度
									AG = 133 MHz
									DS = 80 MHz DDR
									生产工艺
									S = 65 nm MirrorBit 工艺技术
									容量
									064 = 64 Mbit
									器件系列
									S25FS
									赛普拉斯 1.8 V 串行外设接口（SPI）闪存存储器

注意

98. 无卤素定义，符合 IE 61249-2-21 规范

有效组合 — 标准

有效组合列出了计划针对该器件批量支持的配置。如要确认特定有效组合的可用性和了解最新推出的组合，请咨询您当地的销售办事处。

表 75. S25FS064S 有效组合 — 标准

有效组合					
基本订购器件型号	速度选项	封装和温度	型号	包装类型	封装标识
S25FS064S	AG	MFI、MFV、MFN	01	0、1、3	FS064S + A + (温度) + F + 1
		NFI、NFV、NFN	03	0、1、3	FS064S + A + (温度) + F + 3
		BHI、BHV、BHN	02	0、3	FS064S + A + (温度) + H + 2
	DS	MFI、MFV、MFN	01	0、1、3	FS064S + D + (温度) + F + 1
		NFI、NFV、NFN	03	0、1、3	FS064S + D + (温度) + F + 3
		BHI、BHV、BHN	02	0、3	FS064S + D + (温度) + H + 2

有效组合 — 汽车级 /AEC-Q100

下表列出了符合汽车级 /AEC-Q100 标准并计划批量提供的配置。该表会随着新组合的推出而不断更新。如要确认特定组合的供货情况并了解最新推出的组合，请咨询您当地的销售代表。

生产件批准程序 (PPAP) 仅支持 AEC-Q100 等级的产品。

要用于需要符合 ISO/TS-16949 标准的最终用途应用的产品必须是符合 AEC-Q100 级和 PPAP 的产品。对于非 AEC-Q100 级的产品，将不许投产或不提供符合 ISO/TS-16949 标准的全部材料。

此外，对于不要求符合 ISO/TS-16949 标准的终端应用，也会提供无 PPAP 支持的 EC-Q100 级产品。

表 76. S25FS064S 有效组合 — 汽车级 /AEC-Q100

有效组合 — 汽车级 /AEC-Q100					
基本订购器件型号	速度选项	封装和温度	型号	包装类型	封装标识
S25FS064S	AG	MFA、MFB、MFM	01	0、1、3	FS064S + A + (温度) + F + 1
		NFA、NFB、NFM	03	0、1、3	FS064S + A + (温度) + F + 3
		BHA、BHB、BHM	02	0、3	FS064S + A + (温度) + H + 2
	DS	MFA、MFB、MFM	01	0、1、3	FS064S + D + (温度) + F + 1
		NFA、NFB、NFM	03	0、1、3	FS064S + D + (温度) + F + 3
		BHA、BHB、BHM	02	0、3	FS064S + D + (温度) + H + 2

15. 联系赛普拉斯

如需公司各地的最新清单及联系信息，请访问下列网站：www.cypress.com/cypresslocations。

16. 术语表

- **BCD** = 二进制码十进数。每四位代表一个十进制数的值。
- **命令** = **CS#** 为低电平时，在一个周期内主机系统与存储器间传输的所有信息。信息包括指令（有时称为操作代码或操作码）和全部所需地址、模式位、延迟周期或数据。
- **DPD** = 双晶片封装 = 两个晶片在同一个封装中堆叠，以增加单个封装的存储容量。有时被称为多芯片封装（MCP）。
- **DDR** = 双倍数据速率 = 输入和输出在 **SCK** 的每一个边沿上被锁存。
- **ECC 单位** = 闪存阵列和 OTP 阵列中 16 字节对齐和数据长度组。每个单位具有自己的隐藏 ECC，用以在每组中使能纠错功能。
- **闪存** = 它是一种电可擦可编程只读存储器（EEPROM）的名称，能够同时擦除大量的存储器位，这样可使擦除速度比早期的 EEPROM 快得多。
- **高电平** = $\geq V_{IH}$ 的信号电压电平，或逻辑电平为二进制一（“1”）。
- **指令** = 一个 8 位代码，它指明了命令需要执行的功能（有时称为操作代码或操作码）。指令始终是主机系统传送给存储器的命令中最靠前的 8 位。
- **低电平** = $\leq V_{IL}$ 的信号电压电平，或逻辑电平为二进制零（“0”）。
- **LSB** = 最低有效位 = 寄存器或数据值位组中单位数值最小的位，通常是最右边的位。
- **MSB** = 最高有效位 = 寄存器或数据值位组中单位数值最大的位，通常是最左边的位。
- **N/A** = 不适用。某个直不适用于所述条件。
- **非易失性** = 无需使用电源，但仍能保持存储器中的数据。
- **OPN** = 订购器件型号 = 指出存储器件类型、容量、封装、工厂非易失性配置等信息的字母数字，用于选择所需器件。
- **页** = 512 字节或 256 字节大小的对齐数据组。
- **PCB** — 印刷电路板
- 寄存器位参考的格式为：Register_name[bit_number] 或 Register_name[bit_range_MSB: bit_range_LSB]
- **SDR** = 单倍数据速率 = 输入和输出分别在 **SCK** 的上升和下降沿上被锁存。
- **扇区** = 擦除单位大小；根据器件模型和扇区位置，扇区大小可为 4 KB、64 KB 或 256 KB
- **写入** = 该操作用于修改易失性或非易失性寄存器位中或非易失性闪存中的数据。改变非易失性数据时，会对无改变的非易失性数据进行擦除和重新编程，这样可以使用单一的写入操作改变非易失性数据，该方式与改变易失性数据同样的。非易失性数据显示到主机，并会经过单一的写入命令得到更新，这便无需分开使用各种命令来擦除和编程各个相邻但无受影响的数据。

16. 文档修订记录页

文档标题: S25FS064S, 64 Mbit (8 Mbyte), 1.8 V FS-S闪存 文档编号: 002-04137				
版本	ECN编号	变更者	提交日期	变更说明
**	5008504	YLIU	11/13/2015	本文档版本号为Rev**, 译自英文版002-03631 Rev**。
*A	5132377	YLIU	02/10/2016	<p>更新表1.1: 更新FS-S 两列中, “自动启动模式” 和 “工作温度” 两个参数</p> <p>更新章节 芯片选择 (CS#): 更新描述</p> <p>更新表4.1: 更新 “SO / SI” 列 “下电状态” 的接口状态 更新 “IO3_RESET#” 列 “接口待机” 的接口状态</p> <p>更新章节指令周期状态 (QPI 模式): 更新描述</p> <p>更新章节温度范围: 更新描述</p> <p>更新表5.3: 在ICC1 参数的描述中, 指出参考注意项(2)</p> <p>更新章节工业级扩展的直流特性: 更新描述</p> <p>更新表5.4: 将IDPD 参数的最大值从60 μA 改为80 μA</p> <p>更新表5.5: 将IDPD 参数的最大值从100 μA 改为150 μA</p> <p>更新表7.1: 将tPP 参数的最大值从1080 μs 改为2000 μs 更新表7.1 的注意项(1)</p> <p>更新表7.3: 更新tRS 参数的注释</p> <p>更新表9.13: 更新位4 对应的各列</p> <p>更新表9.14: 更新位4 对应的 “类型” 列</p> <p>更新表9.16: 更新表9.16 的注意项(3)</p> <p>更新表9.19: 更新位1 的 “说明” 列</p> <p>更新表9.23: 更新位4、3和0的 “类型” 列</p> <p>更新章节ASP 寄存器: 更新描述</p>

文档标题: S25FS064S, 64 Mbit (8 Mbyte), 1.8 V FS-S闪存 文档编号: 002-04137				
版本	ECN编号	变更者	提交日期	变更说明
*A (续)	5132377	YLIU	02/10/2016	更新章节写入VDLR (WVDLR 4Ah) : 更新描述 更新表12.1: 将“典型值”改为“最小值” 更新表13.9: 更新“参数的相对字节地址偏移”“69h”对应的“数据”列 更新表13.10: 更新“参数的相对字节地址偏移”“6Ch”对应的“数据”列 更新表13.11: 更新“参数的相对字节地址偏移”“76h”对应的“数据”列 更新表13.12: 更新“参数的相对字节地址偏移”“7Ch”对应的“数据”列 更新表13.13: 更新“参数的相对字节地址偏移”列和“数据”列
*B	6536750	YLIU	04/24/2019	本文档版本号为Rev*B, 译自英文版002-03631 Rev*F。

销售、解决方案和法律信息

全球销售和設計支持

赛普拉斯公司拥有一个由办事处、解决方案中心、厂商代表和经销商组成的全球性网络。要想找到最靠近您的办事处，请访问[赛普拉斯所在地](#)。

产品

ARM® Cortex® 微控制器	cypress.com/arm
汽车级产品	cypress.com/automotive
时钟与缓冲器	cypress.com/clocks
接口	cypress.com/interface
物联网	cypress.com/iot
存储器	cypress.com/memory
微控制器	cypress.com/mcu
PSoC	cypress.com/psoc
电源管理 IC	cypress.com/pmic
触摸感应	cypress.com/touch
USB 控制器	cypress.com/usb
无线连接	cypress.com/wireless

PSoC® 解决方案

[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#) | [PSoC 6](#)

赛普拉斯开发者社区

[论坛](#) | [WICED IOT 论坛](#) | [项目](#) | [视频](#) | [博客](#) | [培训](#) | [组件](#)

技术支持

cypress.com/support

© 赛普拉斯半导体公司，2015-2019 年。本文件是赛普拉斯半导体公司及其子公司，包括 Spansion LLC（“赛普拉斯”）的财产。本文件，包括其包含或引用的任何软件或固件（“软件”），根据全球范围内的知识产权法律以及美国与其他国家签署条约由赛普拉斯所有。除非在本款中另有明确规定，赛普拉斯保留在该等法律和条约下的所有权利，且未就其专利、版权、商标或其他知识产权授予任何许可。如果软件并不附随有一份许可协议且贵方未以其他方式与赛普拉斯签署关于使用软件的书面协议，赛普拉斯特此授予贵方属人性质的、非独家且不可转让的如下许可（无再许可权）（1）在赛普拉斯特软件著作权项下的下列许可权（一）对以源代码形式提供的软件，仅出于在赛普拉斯硬件产品上使用之目的且仅在贵方集团内部修改和复制软件，和（二）仅限于在有关赛普拉斯硬件产品上使用之目的将软件以二进制代码形式的向外部最终用户提供（无论直接提供或通过经销商和分销商间接提供），和（2）在被软件（由赛普拉斯公司提供，且未经修改）侵犯的赛普拉斯专利的权利主张项下，仅出于在赛普拉斯硬件产品上使用之目的制造、使用、提供和进口软件的许可。禁止对软件的任何其他使用、复制、修改、翻译或汇编。

在适用法律允许的限度内，赛普拉斯未对本文件或任何软件作出任何明示或暗示的担保，包括但不限于关于适销性和特定用途的默示保证。没有任何电子设备是绝对安全的。因此，尽管赛普拉斯在其硬件和软件产品中采取了必要的安全措施，但是赛普拉斯并不承担任何由于使用赛普拉斯产品而引起的安全问题及安全漏洞的责任，例如未经授权的访问或使用赛普拉斯产品。此外，本材料中所介绍的赛普拉斯产品有可能存在设计缺陷或设计错误，从而导致产品的性能与公布的规格不一致。（如果发现此类问题，赛普拉斯会提供勘误表）赛普拉斯保留更改本文件的权利，届时将不另行通知。在适用法律允许的限度内，赛普拉斯不对因应用或使用本文件所述任何产品或电路引起的任何后果负责。本文件，包括任何样本设计信息或程序代码信息，仅为供参考之目的提供。文件使用人应负责正确设计、计划和测试信息应用和由此生产的任何产品的功能和安全性。赛普拉斯产品不应被设计为、设定为或授权用作武器操作、武器系统、核设施、生命支持设备或系统、其他医疗设备或系统（包括急救设备和手术植入物）、污染控制或有害物质管理系统中的关键部件，或产品植入之设备或系统故障可能导致人身伤害、死亡或财产损失其他用途（“**非预期用途**”）。关键部件指，若该部件发生故障，经合理预期会导致设备或系统故障或会影响设备或系统安全性和有效性的部件。针对由赛普拉斯产品非预期用途产生或相关的任何主张、费用、损失和其他责任，赛普拉斯不承担全部或部分责任且贵方不应追究赛普拉斯之责任。贵方应赔偿赛普拉斯因赛普拉斯产品任何非预期用途产生或相关的所有索赔、费用、损失和其他责任，包括因人身伤害或死亡引起的主张，并使之免受损失。

赛普拉斯、赛普拉斯徽标、Spansion、Spansion 徽标，及上述项目的组合，WICED，及 PSoC、CapSense、EZ-USB、F-RAM 和 Traveo 应视为赛普拉斯在美国和其他国家的商标或注册商标。请访问 cypress.com 获取赛普拉斯商标的完整列表。其他名称和品牌可能由其各自所有者主张为该方财产。