

S29GL-P MirrorBit® 系列闪存

S29GL01GP、S29GL512P、S29GL256P、S29GL128P
1 Gigabit、512 Megabit、256 Megabit 和 128 Megabit
3.0 V 页面模式闪存
90 nm MirrorBit 工艺技术



数据表

读者须知： 本档提供此处介绍的 Spansion 产品的当前技术规格。Spansion Inc. 认为产品已生产足够的数量，因此本档的后续版本不会有变更。但是，可能会有印刷或规格更正，也可能对有效组合进行修改。

数据表标记说明

Spansion Inc. 发行的数据表带有“预备信息”或“初级”等标记，旨在告知读者在整个产品生命周期（包括开发、品质鉴定、试产和量产）内的产品信息或预定规格。无论在任何情况下，我们都鼓励读者在其设计定型前确认他们是否拥有最新信息。为了强调 Spansion 数据表标记的存在及定义，下面列出了他们的详细说明。

预备信息

“预备信息”标记是指 Spansion Inc. 正在开发一个或多个特定产品，但尚未将任何设计交付生产。文档中带有此标记的信息可能变更，在某些情况下，产品开发还可能停止。因此，Spansion Inc. 会对“预备信息”内容施加下列条件：

“本文档包含 Spansion Inc. 正在开发的一个或多个产品的相关信息。这些信息旨在帮助您评估该产品。在未与工厂联系的情况下，请勿基于该产品进行设计。Spansion Inc. 保留变更该预定产品或停止其相关工作的权利，恕不另行通知。”

初级

“初级”标记是指产品开发已在进行中，并且已委托生产。此标记涵盖产品生命周期的多个方面，包括产品品质鉴定、试产、以及制造过程中的后续阶段（量产前）。在考虑到生产的这些方面的情况下，“初级”文档中介绍的技术规格可能会变更。Spansion 对“初级”内容施加下列条件：

“本文档提供此处介绍的 Spansion 产品的当前技术规格。本文档“初级”状态表示产品品质鉴定已经完成，并且试产已经开始。由于制造过程的各个阶段需要保持效率和质量，本文档可能会因为技术规格变更而推出后续版本或修订。”

组合

一些数据表包含带有不同标记（预备信息、初级或量产）的一组产品。这种类型的文档会在必要的地方区分这些产品以及它们的标记，通常是在第一页、订购信息页、以及包含直流特性表及交流擦除和程序表的页（在表注中）。第一页的免责声明会引导读者参阅本页上的说明。

量产（文档上无标记）

当产品生产一段时间后，不再有变更或者只有名义上的变更时，会删除数据表中的“初级”标记。名义变更包括那些会影响可用订购部件号数量的变更，如增加或减少速度选项、温度范围、封装类型或 V_{IO} 范围。变更还包括那些用于澄清描述或者更正印刷错误或规格错误的变更。Spansion Inc. 对此类文档施加下列条件：

“本文档提供此处介绍的 Spansion 产品的当前技术规格。Spansion Inc. 认为产品已生产足够的数量，因此本文档的后续版本不会有变更。但是，可能会有印刷或规格更正，也可能对有效组合进行修改。”

如对这些文档标记有任何疑问，请与您当地的销售机构联系。

S29GL-P MirrorBit® 系列闪存

S29GL01GP、S29GL512P、S29GL256P、S29GL128P
1 Gigabit、512 Megabit、256 Megabit 和 128 Megabit
3.0 V 页面模式闪存
90 nm MirrorBit 工艺技术



数据表

概述

Spansion S29GL01G/512/256/128P 是采用 90nm 工艺技术制造的 Mirrorbit® 闪存产品。这些设备提供 25ns 的快速页面读取时间，相应的随机读取时间可达 90ns。它们配备写入缓冲器，在一个操作中最多可编程 32 字 /64 字节，与标准编程算法相比，有效编程时间更短。因此，对于当今需要更高密度、更好性能和更低功耗的嵌入式应用程序来说，这些设备是理想的选择。

独有特性

- 单个 3V 读取 / 编程 / 擦除 (2.7-3.6 V)
- 增强的 Versatile/O™ 控制
 - 所有输入电平（寻址、控制和 DQ 输入电平）和输出均由 V_{IO} 输入的电压决定。 V_{IO} 的范围是 1.65 到 V_{CC}
- 90 nm MirrorBit 工艺技术
- 8 字 /16 字节页面读取缓冲器
- 32 字 /64 字节写入缓冲器可缩短多字更新的总体编程时间
- 安全硅扇区区域
 - 128 字 /256 字节扇区，通过 8 字 /16 字节随机电子序列号提供永久安全的标识
 - 可在工厂或者由用户进行编程和锁定
- 统一的 64 千字 /128 千字节扇区架构
 - S29GL01GP: 1024 个扇区
 - S29GL512P: 512 个扇区
 - S29GL256P: 256 个扇区
 - S29GL128P: 128 个扇区
- 通常情况下每个扇区可擦除 100,000 次
- 通常情况下 20 年数据保持能力
- 提供的封装
 - 56 针 TSOP
 - 64 球加固 BGA
- 编程和擦除操作的挂起和恢复命令
- 写入操作状态位指明编程和擦除操作是否完成
- 解锁省略编程命令可缩短编程时间
- 支持 CFI (Common Flash Interface, 通用闪存接口)
- 高级扇区保护的持久和密码方法
- WP#/ACC 输入
 - 加快编程速度（当应用了 V_{HH} 时），在系统生产期间提高生产率
 - 保护第一个或最后一个扇区，忽略扇区保护设置
- 硬件复位输入 (RESET#) 复位设备
- 就绪 / 忙碌 # 输出 (RY/BY#) 检测编程或擦除循环是否完成

性能特性

最大读取访问时间 (ns)					
容量	电压范围 (1)	随机访问时间 (t_{ACC})	页面访问时间 (t_{PACC})	CE# 访问时间 (t_{CE})	OE# 访问时间 (t_{OE})
128 & 256 Mb	可调 V_{CC}	90	25	90	25
	最大 V_{CC}	100/110		100/110	
	VersatileIO V_{IO}	110		110	
512 Mb	可调 V_{CC}	100	25	100	25
	最大 V_{CC}	110		110	
	VersatileIO V_{IO}	120		120	
1 Gb	可调 V_{CC}	110	25	110	25
	最大 V_{CC}	120		120	
	VersatileIO V_{IO}	130		130	

备注

- 访问时间取决于 V_{CC} 和 V_{IO} 运行范围。
 有关的详细信息, 请参见订购信息页。
 可调 V_{CC} : $V_{CC} = 3.0-3.6 V$ 。
 最大 V_{CC} : $V_{CC} = V_{IO} = 2.7-3.6 V$ 。
 VersatileIO V_{IO} : $V_{IO} = 1.65-V_{CC}$, $V_{CC} = 2.7-3.6 V$ 。
- 请咨询销售代表以了解具体可用性。

电流消耗 (典型值)	
随机访问读取 ($f = 5 MHz$)	30 mA
8 字页面读取 ($f = 10 MHz$)	1 mA
编程 / 擦除	50 mA
待机	1 μA

编程和擦除时间 (典型值)	
单字编程	60 μs
每字有效写入缓冲器编程 (V_{CC})	15 μs
每字有效写入缓冲器编程 (V_{HH})	13.5 μs
扇区擦除时间 (64 千字扇区)	0.5 s

目录

概述	3
独有特性	3
性能特性	4
1. 订购信息	9
2. 输入 / 输出描述和逻辑符号	11
3. 框图	12
4. 物理尺寸 / 连接图	12
4.1 相关文档	12
4.2 关于 BGA 封装的特殊处置说明	12
4.3 LAA064—64 球加固球栅阵列, 11 x 13 mm	14
4.4 TS056—56 针标准薄小外形封装 (TSOP)	16
5. 更多资源	17
5.1 应用说明	17
5.2 规格公告板	17
5.3 硬件和软件支持	17
5.4 联系 Spansion	17
6. 产品概述	18
6.1 存储映射	18
7. 设备操作	19
7.1 设备操作表	19
7.2 字 / 字节配置	20
7.3 VersatileIO™ (VIO) 控制	20
7.4 读取	20
7.5 页面读取模式	20
7.6 自动选择	21
7.7 编程 / 擦除操作	24
7.8 写入操作状态	36
7.9 写入命令 / 命令序列	39
8. 高级扇区保护 / 解保护	41
8.1 锁定寄存器	42
8.2 持久保护位	42
8.3 持久保护位锁定	44
8.4 密码保护方法	44
8.5 高级扇区保护软件示例	46
8.6 硬件数据保护方法	46
9. 节能模式	47
9.1 待机模式	47
9.2 自动睡眠模式	47
9.3 硬件 RESET# 输入操作	47
9.4 输出禁用 (OE#)	47
10. 安全硅扇区闪存区域	48
10.1 工厂锁定的安全硅扇区	48
10.2 用户可锁定的安全硅扇区	49
10.3 安全硅扇区进入 / 退出命令序列	49
11. 电气规格	51
11.1 绝对最大额定值	51
11.2 运行范围	52
11.3 测试条件	52
11.4 波形切换	52
11.5 切换波形	53
11.6 直流特性	54
11.7 交流特性	55

12. 附录	66
12.1 命令定义	66
12.2 通用闪存接口	71
13. 关于 S29GL-R 65 nm MirrorBit 硬件的高级信息 复位 (RESET#) 和 加电 序列	74
14. 修订历史	75

图

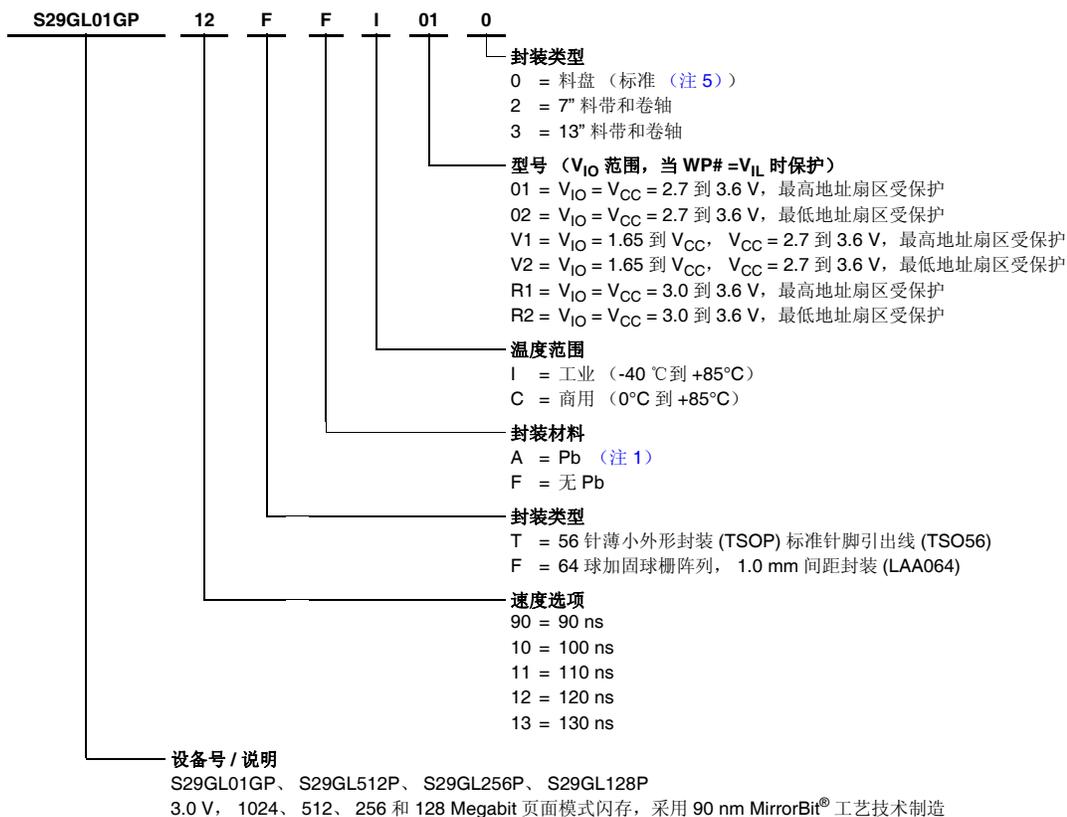
图 3.1	S29GL-P 框图	12
图 4.1	64 球加固球栅阵列	13
图 4.2	LAA064—64 球加固球栅阵列 (FBGA), 11 x 13 mm	14
图 4.3	56 针标准 TSOP (顶视图)	15
图 4.4	56 针薄小外形封装 (TSOP), 14 x 20 mm	16
图 7.1	单字编程	25
图 7.2	写入缓冲器编程操作	29
图 7.3	扇区擦除操作	31
图 7.4	写入操作状态流程图	37
图 8.1	高级扇区保护 / 解保护	41
图 8.2	PPB 编程算法	42
图 8.3	锁定寄存器编程算法	45
图 11.1	最大负过冲波形	51
图 11.2	最大过冲波形	51
图 11.3	测试设置	52
图 11.4	输入波形和测量电平	53
图 11.5	读取操作时序	56
图 11.6	页面读取时序	56
图 11.7	复位时序	57
图 11.8	加电序列时序	58
图 11.9	编程操作时序	60
图 11.10	加速编程时序图	60
图 11.11	芯片 / 扇区擦除操作时序	61
图 11.12	Data# 轮询时序 (在嵌入式算法期间)	61
图 11.13	跳变位时序 (在嵌入式算法期间)	62
图 11.14	DQ2 对 DQ6	62
图 11.15	备用 CE# 控制的写入 (擦除 / 编程) 操作 时序	64
图 13.1	复位时序	74
图 13.2	加电复位时序	74

表

表 2.1	输入 / 输出描述	.11
表 6.1	S29GL01GP 扇区和存储地址映射	.18
表 6.2	S29GL512P 扇区和存储地址映射	.18
表 6.3	S29GL256P 扇区和存储地址映射	.18
表 6.4	S29GL128P 扇区和存储地址映射	.18
表 7.1	设备操作	.19
表 7.2	自动选择代码, (高电压方法)	.22
表 7.3	自动选择系统内地址	.22
表 7.4	自动选择系统内进入	.23
表 7.5	自动选择退出	.23
表 7.6	单字 / 字节编程	.26
表 7.7	写入缓冲器编程	.28
表 7.8	扇区擦除	.30
表 7.9	芯片擦除	.32
表 7.10	擦除挂起	.33
表 7.11	擦除恢复	.33
表 7.12	编程挂起	.34
表 7.13	编程恢复	.34
表 7.14	解锁省略进入	.35
表 7.15	解锁省略编程	.35
表 7.16	解锁省略复位	.36
表 7.17	写入操作状态	.39
表 7.18	复位	.40
表 8.1	锁定寄存器	.42
表 8.2	扇区保护方案: DYB、PPB 和 PPB 锁定位组合	.46
表 10.1	安全硅扇区地址	.48
表 10.2	安全硅扇区进入	.49
表 10.3	安全硅扇区编程	.49
表 10.4	安全硅扇区退出	.50
表 11.1	测试规格	.52
表 11.2	S29GL-P 直流特性 (CMOS 兼容)	.54
表 11.3	S29GL-P 读取操作	.55
表 11.4	硬件复位 (RESET#)	.57
表 11.5	加电序列时序	.58
表 11.6	S29GL-P 擦除和编程操作	.59
表 11.7	S29GL-P 备用 CE# 控制的擦除和编程 操作	.63
表 11.8	擦除和编程性能	.65
表 11.9	封装电容	.65
表 12.1	S29GL-P 存储阵列命令定义, x16	.67
表 12.2	S29GL-P 扇区保护命令定义, x16	.68
表 12.3	S29GL-P 存储阵列命令定义, x8	.69
表 12.4	S29GL-P 扇区保护命令定义, x8	.70
表 12.5	CFI 查询识别字符串	.71
表 12.6	系统接口字符串	.71
表 12.7	设备几何定义	.72
表 12.8	主要厂商特定的扩展查询	.73
表 13.1	硬件复位 (RESET#)	.74
表 13.2	加电序列时序	.74

1. 订购信息

订购部件号是如下形式的有效组合：



推荐组合

推荐组合列出了计划针对此设备批量支持的配置。如要确认特定推荐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。

S29GL-P 有效组合					
基本部件号	速度	封装 (2)(3)	温度 (4)	型号	封装类型 (5)
S29GL01GP	11	TA (1), TF	I, C	R1, R2	0, 3
	12		I	01, 02	
	13			V1, V2	
	11	FA (1), FF	I, C	R1, R2	0, 2, 3
	12		I	01, 02	
	13			V1, V2	
S29GL512P	10	TA (1), TF	I, C	R1, R2	0, 3
	11		I	01, 02	
	12			V1, V2	
	10	FA (1), FF	I, C	R1, R2	0, 2, 3
	11		I	01, 02	
	12			V1, V2	
S29GL128P, S29GL256P	90	TA (1), TF	I, C	R1, R2	0, 3
	10, 11		I	01, 02	
	11			V1, V2	
	90	FA (1), FF	I, C	R1, R2	0, 2, 3
	10, 11		I	01, 02	
	11			V1, V2	

备注

1. 请咨询当地销售代表以了解具体可用性。
2. TSOP 封装标识省略了订购部件号中的封装类型符号。
3. BGA 封装标识省略了前面的“S29”和订购部件号中的封装类型符号。
4. 运行温度范围: I = 工业 (-40°C 到 +85°C)
C = 商用 (0°C 到 +85°C)
5. 类型 0 是标配。根据需要指定其他选项。

2. 输入 / 输出描述和逻辑符号

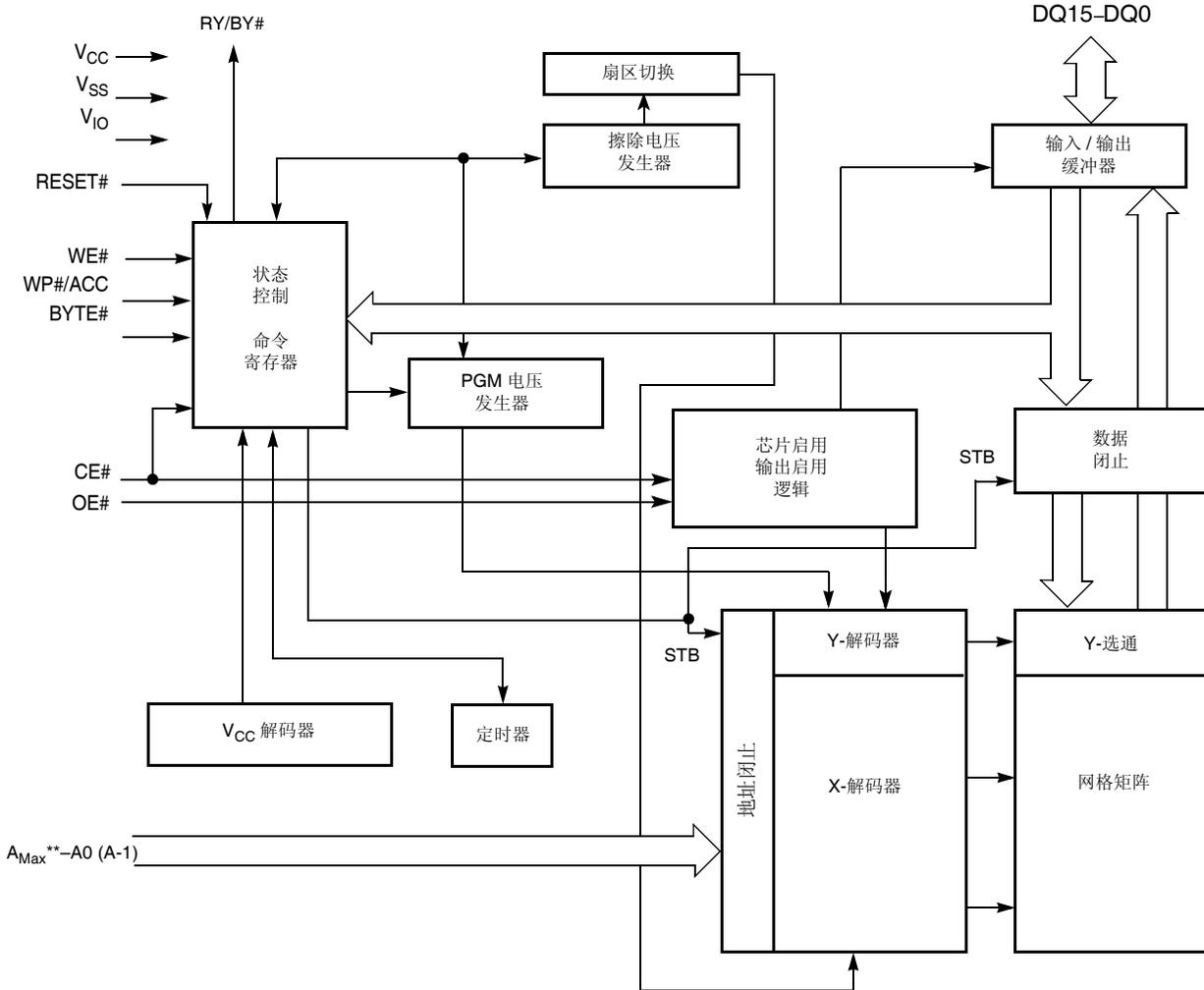
表 2.1 标识设备上提供的输入和输出封装连接。

表 2.1 输入 / 输出描述

符号	类型	描述
A25-A0	输入	GL01GP 地址行 A24-A0 (GL512P) A23-A0 (GL256P), A22-A0 (GL128P)。
DQ14-DQ0	I/O	数据输入 / 输出。
DQ15/A-1	I/O	DQ15: 字模式数据输入 / 输出。 A-1: 字节模式 LSB 地址输入。
CE#	输入	片选。
OE#	输入	输出使能。
WE#	输入	写使能。
V _{CC}	电源	设备电源。
V _{IO}	电源	多用途 IO 输入。
V _{SS}	电源	接地。
NC	不连接	内部未连接。
RY/BY#	输出	准备就绪 / 忙碌。指明嵌入式算法是正在执行还是已完成。在 V _{IL} ，设备正在擦除或编程。 在高 Z，设备准备就绪。
BYTE#	输入	选择数据总线宽度。在 V _{IL} ，设备处于字节配置，数据 I/O 引脚 DQ0-DQ7 有效，DQ15/A-1 成为 LSB 地址输入。在 V _{IH} ，设备处于字配置，数据 I/O 引脚 DQ0-DQ15 有效。
RESET#	输入	硬件复位。低 = 设备复位并返回读取阵列数据。
WP#/ACC	输入	写入保护 / 加速输入。在 V _{IL} ，在最外面的扇区禁用编程和擦除功能。在 V _{HH} ，加速编程； 自动将设备置于解锁省略模式。对于所有其他情形，应处于 V _{IH} 。WP# 有一个内部拉升； 当断开连接时，WP# 处于 V _{IH} 。

3. 框图

图 3.1 S29GL-P 框图



** A_{Max} GL01GP=A25, A_{Max} GL512P = A24, A_{Max} GL256P = A23, A_{Max} GL128P = A22

4. 物理尺寸 / 连接图

此部分介绍 S29GL-P 系列的 I/O 标记和封装规格。

4.1 相关文档

下列文档包含与 S29GL-P 设备相关的信息。单击标题或转到 www.spansion.com 下载 PDF 文件，或者向相关的销售办事处索取副本。

- 关于表面贴装闪存集成电路 X 光检查的注意事项

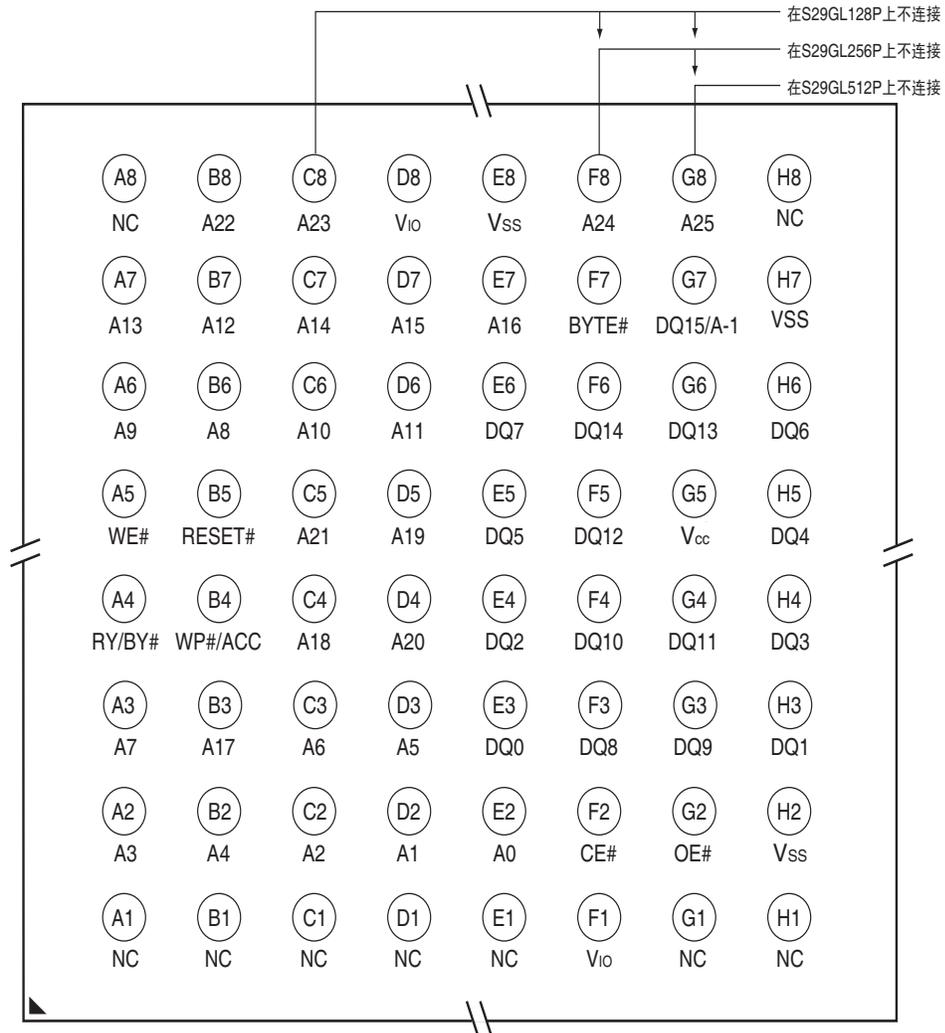
4.2 关于 BGA 封装的特殊处置说明

BGA 封装的闪存产品需要特殊处置。

如果使用超声波清洁方法，BGA 封装的闪存设备可能损坏。如果封装体长时间放置在温度高于 150°C 的地方，封装和 / 或数据完整性可能受损。

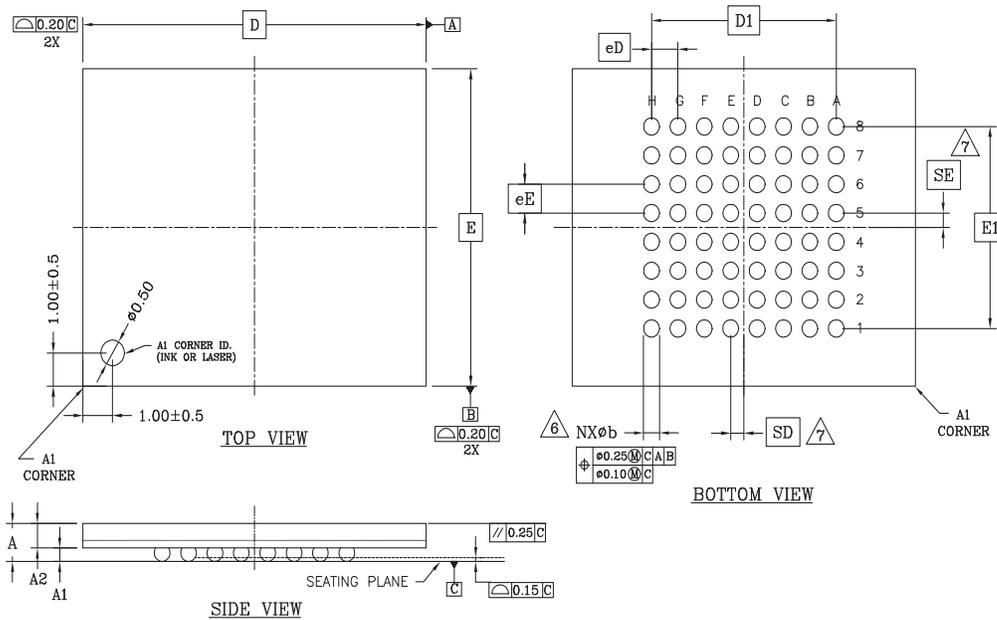
图 4.1 64 球加固球栅阵列

顶视图，球朝下



4.3 LAA064—64 球加固球栅阵列, 11 x 13 mm

图 4.2 LAA064—64 球加固球栅阵列 (FBGA), 11 x 13 mm



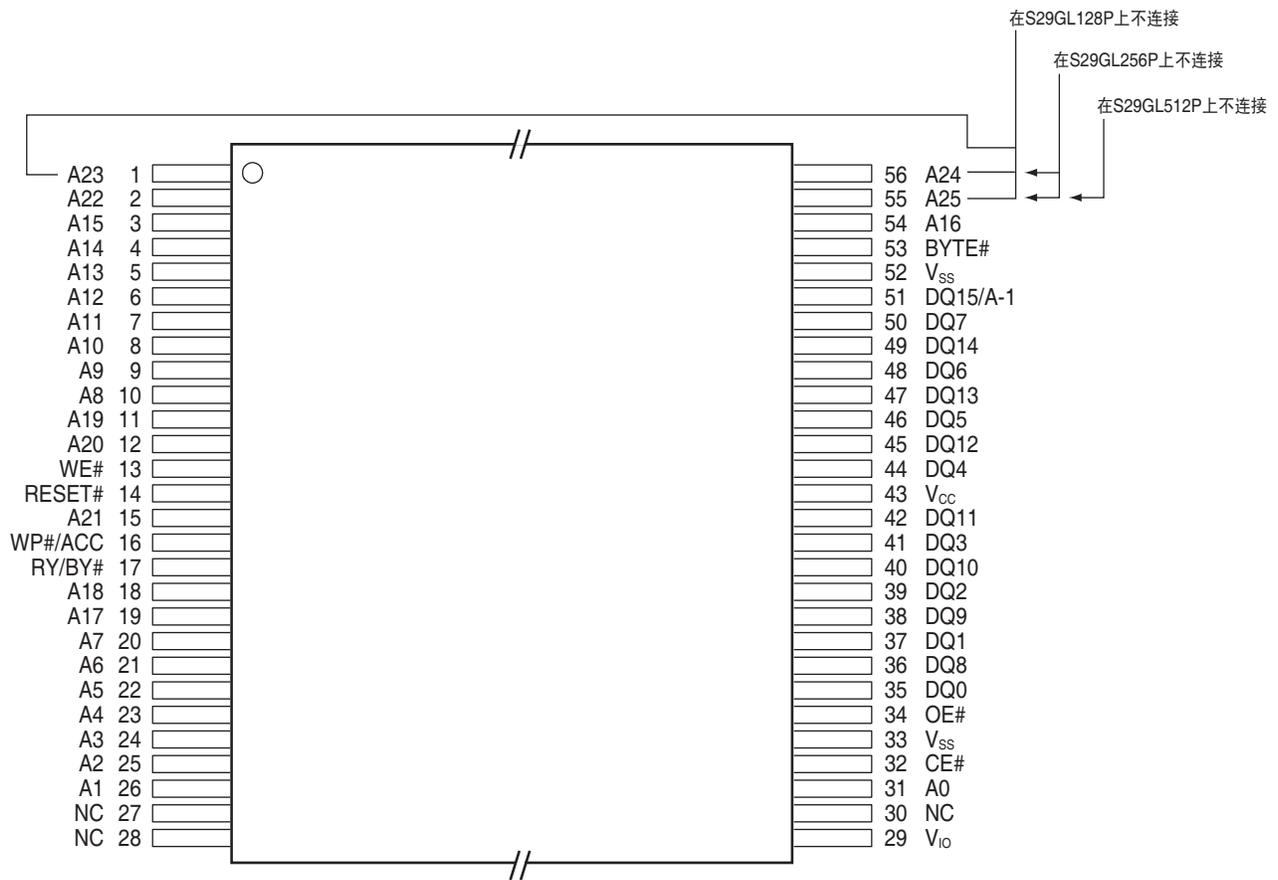
符号	MIN	NOM	MAX	注
封装	LAA 064			
JEDEC	不适用			
	13.00 mm x 11.00 mm			
	封装			
A	---	---	1.40	轮廓高度
A1	0.40	---	---	有支架
A2	0.60	---	---	本体厚度
D	13.00 BSC.			本体尺寸
E	11.00 BSC.			本体尺寸
D1	7.00 BSC.			矩阵轨迹
E1	7.00 BSC.			矩阵轨迹
MD	8			矩阵尺寸D方向
ME	8			矩阵尺寸E方向
N	64			球计数
φb	0.50	0.60	0.70	球直径
eD	1.00 BSC.			球间距 - D方向
eE	1.00 BSC.			球间距 - E方向
SD / SE	0.50 BSC.			焊接球位置
	无			缩减焊接球

注:

- 外形尺寸和公差基于ASME Y14.5M-1994
- 所有外形尺寸单位为毫米。
- 球位置说明基于JESD 95-1, SPP-010 (除非特别说明)。
- e表示焊接球栅间距。
- 符号“MD”是“D”方向的球形矩阵尺寸。
符号“ME”是“E”方向的球形矩阵尺寸。
N是焊接球的总数量。
- 外形尺寸“b”是在与基准C平行的平面上对最大球直径进行测量而得到的。
SD和SE根据基准A和B测量得到, 它们定义中心焊接球在外部行中的位置。
当与D或E方向平行的外部行中的焊接球数量是单数时, 相应地, SD或SE = 0.000。
当外部行中的焊接球数量是偶数时, SD或SE = $\frac{e}{2}$
- 不使用。
- “+”指明缩减球的理论中心。

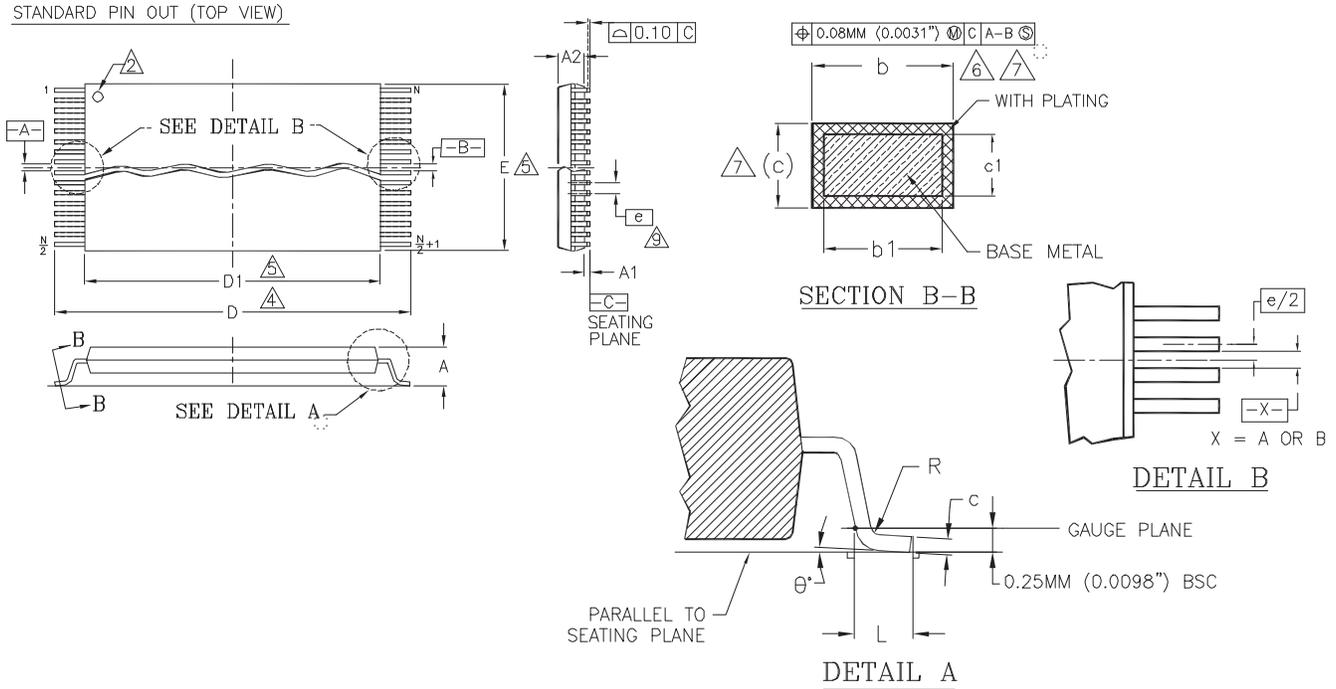
3354 116-038.12d

图 4.3 56 针标准 TSOP (顶视图)



4.4 TS056—56 针标准薄小外形封装 (TSOP)

图 4.4 56 针薄小外形封装 (TSOP), 14 x 20 mm



封装	TS 56		
JEDEC	MO-142 (B) EC		
符号	MIN.	NOM.	MAX.
A	---	---	1.20
A1	0.05	---	0.15
A2	0.95	1.00	1.05
b1	0.17	0.20	0.23
b	0.17	0.22	0.27
c1	0.10	---	0.16
c	0.10	---	0.21
D	19.80	20.00	20.20
D1	18.30	18.40	18.50
E	13.90	14.00	14.10
e	0.50 BASIC		
L	0.50	0.60	0.70
θ	0°	-	8°
R	0.08	---	0.20
N	56		

注:

- 1 控制外形尺寸单位为毫米(mm)。(外形尺寸和公差符合ANSI Y14.5M-1982)
- 2 标准针脚引线的针脚1标识符(晶片朝上)。
- 3 由底座平面-C决定。底座平面定义为:封装引线被允许随意放置于平坦水平表面上时制作的接触平面。
- 4 外形尺寸D1和E不包括铸模凸起。允许的铸模凸起是每个侧面0.15 mm。
- 5 外形尺寸b不包括DAMBAR凸起。允许的DAMBAR凸起应在最大材料条件下超过b外形尺寸最多0.08 mm。凸起和临近引线间的最小间距应是0.07 mm。
- 6 这些外形尺寸适用于距引线端头0.10 mm和0.25 mm之间的引线扁平部分。
- 7 引线共面性应在0.10 mm以内(从底座平面测量)。
- 8 外形尺寸"e"在引线中心线测量。

3160/38.10A

5. 更多资源

www.spansion.com 提供下列相关文档：

5.1 应用说明

下面列出了与此产品有关的应用说明。下面网站提供 Spansion 所有产品的应用说明：
<http://www.spansion.com/Support/TechnicalDocuments/Pages/ApplicationNotes.aspx>

- 使用 AMD 设备中的操作状态位
- 了解页面模式闪存设备
- MirrorBit® 闪存写入缓冲器编程和页面缓冲器读取
- 通用闪存接口 1.4 版厂商专用扩展
- MirrorBit® 闪存写入缓冲器编程和页面缓冲器读取
- 在 MCF5407 Coldfire 上利用页面模式读取
- 移植到基于 110nm MirrorBit® 技术的 S29GL128N 和 S29GL256N
- 优化编程 / 擦除时间
- 耐久性和数据保持性实践指南
- 使用 Spansion S29GL-N 闪存配置 FPGA
- 将 Spansion™ 闪存连接到系统地址总线
- 连接不使用的 MirrorBit® 闪存数据线
- MirrorBit® 闪存复位电压和时序要求
- 多用途 IO: DQ 和增强

5.2 规格公告板

请联系您当地销售机构了解详细信息。

5.3 硬件和软件支持

可从下面的网站下载闪存设备的支持信息
<http://www.spansion.com/Support/Pages/DriversSoftware.aspx>

- Spansion 底层驱动程序
- 增强型闪存驱动程序
- 闪存文件系统

从下面的网站下载仿真模型和 CAD 模型的支持信息
<http://www.spansion.com/Support/Pages/SimulationModels.aspx>

- VHDL 和 Verilog
- IBIS
- ORCAD

5.4 联系 Spansion

如需公司各个地点的最新清单及联系信息，请访问我们的网站：
<http://www.spansion.com/About/Pages/Locations.aspx>

6. 产品概述

S29GL-P 系列包括 1 Gb/512 Mb/256 Mb/128 Mb 3.0V 页面模式闪存设备，它们专为当今要求大容量存储阵列和丰富功能的嵌入式设计而优化。这些设备采用 90 nm MirrorBit 技术制造。这些产品提供统一的 64 千字（128 千字节）统一扇区，支持 VersatileIO 控制，允许控制以及输入 / 输出信号在 1.65 V 到 V_{CC} 范围内运行。更多功能包括：

- 单字编程或 32 字编程缓冲器，提高编程速度
- 编程挂起 / 恢复和擦除挂起 / 恢复
- 高级扇区保护方法，根据需要保护扇区
- 128 字 /256 字节安全硅区域，可存储用户和工厂安全信息。安全硅扇区是一次性可编程。

6.1 存储映射

S29GL-P 设备包括统一的 64 千字（128 千字节）扇区，其组织方式如表 6.1– 表 6.4 所示。

表 6.1 S29GL01GP 扇区和存储地址映射

统一的扇区大小	扇区数量	扇区范围	地址范围（16 位）	备注
64 千字 / 128 千字节	1024	SA00	0000000h-000FFFFh	扇区起始地址
		:	:	
		SA1023	3FF0000H - 3FFFFFFFh	扇区结束地址

注

为了在单个页面上显示整个设备的扇区相关信息，此表进行了简化。对于未明确列出的扇区以及它们的地址范围（如 SA001-SA1022），其扇区起始地址和结束地址与该大小的所有其他扇区具有相同的组合。例如，所有 128 Kb 扇区的组合均为 xxx0000h-xxxFFFFh。

表 6.2 S29GL512P 扇区和存储地址映射

统一的扇区大小	扇区数量	扇区范围	地址范围（16 位）	备注
64 千字 / 128 千字节	512	SA00	0000000h-000FFFFh	扇区起始地址
		:	:	
		SA511	1FF0000H - 1FFFFFFFh	扇区结束地址

注

为了在单个页面上显示整个设备的扇区相关信息，此表进行了简化。对于未明确列出的扇区以及它们的地址范围（如 SA001-SA510），其扇区起始地址和结束地址与该大小的所有其他扇区具有相同的组合。例如，所有 128 Kb 扇区的组合均为 xxx0000h-xxxFFFFh。

表 6.3 S29GL256P 扇区和存储地址映射

统一的扇区大小	扇区数量	扇区范围	地址范围（16 位）	备注
64 千字 / 128 千字节	256	SA00	0000000h-000FFFFh	扇区起始地址
		:	:	
		SA255	0FF0000H - 0FFFFFFFh	扇区结束地址

注

为了在单个页面上显示整个设备的扇区相关信息，此表进行了简化。对于未明确列出的扇区以及它们的地址范围（如 SA001-SA254），其扇区起始地址和结束地址与该大小的所有其他扇区具有相同的组合。例如，所有 128 Kb 扇区的组合均为 xxx0000h-xxxFFFFh。

表 6.4 S29GL128P 扇区和存储地址映射

统一的扇区大小	扇区数量	扇区范围	地址范围（16 位）	备注
64 千字 / 128 千字节	128	SA00	0000000h-000FFFFh	扇区起始地址
		:	:	
		SA127	07F0000 - 7FFFFFF	扇区结束地址

注

为了在单个页面上显示整个设备的扇区相关信息，此表进行了简化。对于未明确列出的扇区以及它们的地址范围（如 SA001-SA510），其扇区起始地址和结束地址与该大小的所有其他扇区具有相同的组合。例如，所有 128 Kb 扇区的组合均为 xxx0000h-xxxFFFFh。

7. 设备操作

此部分介绍闪存设备的读取、编程、擦除、握手、以及复位等功能。

操作通过向命令寄存器写入特定命令或包含特定地址和数据组合的序列来启动（参见表 12.1 到表 12.4）。命令寄存器本身并不占用任何可寻址的存储位置；它包含许多锁存器，这些锁存器存储命令以及执行命令所需的地址和数据信息。寄存器的内容用作内部状态机的输入，状态机的输出指明设备的功能。如果写入错误的地址和数据值或者将它们写入错误的序列，可能使设备进入未知状态；在此情况下，系统必须拉低 RESET# 引脚或下电重启，以使设备返回读取阵列数据模式。

7.1 设备操作表

设备必须根据每个操作进行相应的设置。表 7.1 介绍各个特定操作要求的每个控制引脚的状态。

表 7.1 设备操作

操作	CE#	OE#	WE#	RESET#	WP#/ACC	地址 (注 1)	DQ0-DQ7	DQ8-DQ15	
								BYTE#= V _{IH}	BYTE#= V _{IL}
读取	L	L	H	H	X	A _{IN}	D _{OUT}	D _{OUT}	DQ8-DQ14 = High-Z, DQ15 = A-1
写入（编程 / 擦除）	L	H	L	H	(注 2)	A _{IN}	(注 3)	(注 3)	
加速编程	L	H	L	H	V _{HH}	A _{IN}	(注 3)	(注 3)	
待机	V _{CC} ± 0.3 V	X	X	V _{CC} ± 0.3 V	H	X	High-Z	High-Z	High-Z
输出禁用	L	H	H	H	X	X	High-Z	High-Z	High-Z
复位	X	X	X	L	X	X	High-Z	High-Z	High-Z

缩略语

L = 逻辑低 = V_{IL}, H = 逻辑高 = V_{IH}, V_{HH} = 11.5–12.5V, X = 不用考虑, A_{IN} = 地址输入, D_{IN} = 数据输入, D_{OUT} = 数据输出

备注

- 地址分别是 A_{Max}:A₀（字模式）；A_{Max}:A-1（字节模式）。
- 如果 WP# = V_{IL}，最外面的扇区保持受保护。如果 WP# = V_{IH}，最外面的扇区不受保护。WP# 有一个内部拉升；当断开连接时，WP# 处于 V_{IH}。出厂时，所有扇区均不受保护（根据所订购的版本，安全硅扇区可以在工厂设为保护）。
- D_{IN} 或 D_{OUT}，根据命令序列、数据轮询或扇区保护算法的需要。

7.2 字 / 字节配置

BYTE# 引脚控制设备数据 I/O 引脚是以字节配置还是字配置运行。如果 BYTE# 引脚设在逻辑 '1'，则设备采用字配置，DQ0-DQ15 有效并受 CE# 和 OE# 控制。

如果 BYTE# 引脚设在逻辑 '0'，则设备采用字节配置，仅数据 I/O 针 DQ0-DQ7 有效并受 CE# 和 OE# 控制。数据 I/O 针脚 DQ8-DQ14 处于三态，DQ15 针脚用作 LSB (A-1) 地址功能的输入。

7.3 VersatileIO™ (V_{IO}) 控制

VersatileIO™ (V_{IO}) 控制允许主机系统设置设备在所有输入和输出（地址、控制和 DQ 信号）上产生和容忍的电压。V_{IO} 范围是 1.65 到 V_{CC}。有关此设备的 V_{IO} 选项，请参见第 9 页的订购信息。

例如，1.65-3.6V 的 V_{IO} 允许 I/O 为 1.8 或 3V，将信号驱动到同一数据总线上的其他 1.8 或 3V 设备或从其接收信号。

7.4 读取

所有存储器都需要存取时间以输出阵列数据。在读取操作中，每次从一个存储位置读取数据。地址以随机顺序提供给设备，经由设备的传播延时导致其输出上的数据到达其输入上的地址。

在默认情况下，设备在设备加电或硬件复位后读取阵列数据。为了从存储阵列读取数据，系统必须先在 A_{max}-A₀ 上声明一个有效地址，同时驱动 OE# 和 CE# 至 V_{IL}。WE# 必须保持在 V_{IH}。所有地址在 CE# 下降沿被锁存。在地址存取时间 (t_{ACC}) 之后，即从稳定地址到有效输出数据的延迟之后，数据出现在 DQ15-DQ0 上。OE# 信号必须驱动至 V_{IL}。自 OE# 下降沿起经过读取时间 (t_{OE}) 之后，假若 t_{ACC} 存取时间已满足，则数据在 DQ15-DQ0 针脚上输出。

7.5 页面读取模式

设备支持快速页面模式读取，与页面模式 Mask ROM 读取操作兼容。此模式可对一个页面内的随机位置提供更快的读取访问速度。设备的页面大小是 8 字 /16 字节。相应的页面由地址高位 A(max)-A3 来选择。字模式下的地址位 A2-A0（字节模式下的 A2 到 A-1）决定一个页面内的特定字。微处理器提供特定字位置。

随机或初始页面读取等于 t_{ACC} 或 t_{CE}，后续页面读取（只要微处理器指定的位置位于该页面内）等效于 t_{PACC}。当 CE# 被拉高无效然后又拉低有效时，读取时间为 t_{ACC} 或 t_{CE}（即快速页面读取是在 CE# 持续保持有效期内实现的，而 CE# 拉高后再拉低，则意味新的页面读取开始。）快速页面模式读取通过使“读取页面地址”保持不变而更改“页面内读取”地址来实现。

7.6 自动选择

自动选择模式通过 DQ7-DQ0 上内部寄存器（独立于存储阵列）的标识符代码输出来提供制造商 ID、设备标识、以及扇区保护信息。此模式的主要目的是对装置编程，以自动使要编程的设备与对应的编程算法相匹配（参见表 7.3）。自动选择代码也可以在系统内读取。

可通过两种方法读取自动选择代码。一种使用自动选择命令，另一种在地址引脚 A9 上应用 V_{ID} 。

当使用编程装置时，自动选择模式要求地址引脚 A9 上为 V_{ID} （11.5 V 到 12.5 V）。地址引脚必须如表 7.2 所示。

- 如要在 A9 上不是高电压时进入自动选择模式，主机系统必须发出自动选择命令。
- 自动选择命令序列可以写入处于读取或擦除挂起读取模式的扇区内的一个地址。
- 当设备正在编程或擦除时，不能写入自动选择命令。
- 系统必须写入复位命令，才能返回读取模式（或擦除挂起读取模式，如果扇区之前处于擦除挂起状态）。
- 建议在加电序列完成后在 A9 上应用 V_{ID} 。此外，建议在关闭 V_{CC}/V_{IO} 之前，在 A9 上应用 V_{ID} 到 V_{IH}/V_{IL} 。
- 有关命令序列的详细信息，请参见第 67 页的表 12.1。
- 在检验扇区保护时，扇区地址必须出现在相应的地址高位上（参见表 7.4 到表 7.5）。其余地址位不用考虑。根据需要设置了所有必要的位后，编程装置就可以读取 DQ15-DQ0 上相应的标识符代码。自动选择代码也可以通过命令寄存器在系统内读取。

表 7.2 自动选择代码，（高电压方法）

描述	CE#	OE#	WE#	Amax 到 A16	A14 到 A10	A9	A8 到 A7	A6	A5 到 A4	A3 到 A2	A1	A0	DQ8 到 DQ15		DQ7 到 DQ0
													BYTE# = V _{IH}	BYTE# = V _{IL}	
制造商 ID: Spansion 产品	L	L	H	X	X	V _{ID}	X	L	X	L	L	L	00	X	01h
设备 ID S29GL01GP	L	L	H	X	X	V _{ID}	X	L	X	L	L	H	22	X	7Eh
										H	H	L	22	X	28h
										H	H	H	22	X	01h
设备 ID S29GL512P	L	L	H	X	X	V _{ID}	X	L	X	L	L	H	22	X	7Eh
										H	H	L	22	X	23h
										H	H	H	22	X	01h
设备 ID S29GL256P	L	L	H	X	X	V _{ID}	X	L	X	L	L	H	22	X	7Eh
										H	H	L	22	X	22h
										H	H	H	22	X	01h
设备 ID S29GL128P	L	L	H	X	X	V _{ID}	X	L	X	L	L	H	22	X	7Eh
										H	H	L	22	X	21h
										H	H	H	22	X	01h
扇区群组保护校验	L	L	H	SA	X	V _{ID}	X	L	X	L	H	L	X	X	01h（受保护）， 00h（不受保护）
安全硅扇区指示位 (DQ7)，WP# 保护最 高地址扇区	L	L	H	X	X	V _{ID}	X	L	X	L	H	H	X	X	99h（工厂锁定）， 19h（工厂不锁定）
安全硅扇区指示位 (DQ7)，WP# 保护最 低地址扇区	L	L	H	X	X	V _{ID}	X	L	X	L	H	H	X	X	89h（工厂锁定）， 09h（工厂不锁定）

缩略语

 L = 逻辑低 = V_{IL}, H = 逻辑高 = V_{IH}, SA = 扇区地址, X = 不用考虑。V_{ID} = 11.5V 到 12.5V

表 7.3 自动选择系统内地址

描述	地址	读取数据（字/字节模式）
制造商 ID	基址 + 00h	xx01h/1h
设备 ID, 字 1	基址 + 01h	227Eh/7Eh
设备 ID, 字 2	基址 + 0Eh	2228h/28h (GL01GP) 2223h/23h (GL512P) 2222h/22h (GL256P) 2221h/21h (GL128P)
设备 ID, 字 3	基址 + 0Fh	2201h/01h
安全设备检验	基址 + 03h	对于 S29GLxxxPH: XX19h/19h = 工厂不锁定。XX99h/99h = 工厂锁定。 对于 S29GLxxxPL: XX09h/09h = 工厂不锁定。XX89h/89h = 工厂锁定。
扇区保护检验	(SA) + 02h	xx01h/01h = 锁定, xx00h/00h = 不锁定

软件功能和代码示例

表 7.4 自动选择进入
(LLD 功能 = lld_AutoselectEntryCmd)

循环	操作	字节地址	字地址	数据
解锁循环 1	写入	基址 + AAAh	基址 + 555h	0x00AAh
解锁循环 2	写入	基址 + 555h	基址 + 2AAh	0x0055h
自动选择命令	写入	基址 + AAAh	基址 + 555h	0x0090h

表 7.5 自动选择退出
(LLD 功能 = lld_AutoselectExitCmd)

循环	操作	字节地址	字地址	数据
自动选择退出命令	写入	基址 + XXXh	基址 + XXXh	0x00F0h

注

1. 设备内的任何偏移均能工作。
2. 基址 = 基准地址。

下面是使用自动选择功能读取制造商 ID 的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```

/* Here is an example of Autoselect mode (getting manufacturer ID) */
/* Define UINT16 example:typedef unsigned short UINT16; */

UINT16 manuf_id;

/* Auto Select Entry */

*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x0090; /* write autoselect command */

/* multiple reads can be performed after entry */

manuf_id = * ( (UINT16 *)base_addr + 0x000 ); /* read manuf. id */

/* Autoselect exit */

*( (UINT16 *)base_addr + 0x000 ) = 0x00F0; /* exit autoselect (write reset command) */

```

7.7 编程 / 擦除操作

这些设备支持多种编程和 / 或擦除操作模式，以下部分会详细介绍。

在写入操作期间，当提供地址、命令和数据时，系统必须驱动 $CE\#$ 和 $WE\#$ 至 V_{IL} ， $OE\#$ 至 V_{IH} 。地址在 $WE\#$ 或 $CE\#$ 的最后一个下降沿被锁存，数据在 $WE\#$ 或 $CE\#$ 的第一个上升沿被锁存。

解锁省略功能允许主机系统向闪存设备发送编程命令，而无需预先在命令序列内写入解锁循环。有关解锁省略功能的详细信息，请参见 [章节 7.7.8](#)。

请注意下列事项：

- 嵌入式编程算法完成后，设备返回读取模式。
- 系统可通过读取 DQ 状态位来确定编程操作的状态。有关这些状态位的信息，请参见 [第 36 页](#) 的 [写入操作状态](#)。
- “0”无法编程回“1”。后续读取显示数据仍为“0”。
- 只有擦除操作能使“0”恢复回“1”。
- 在嵌入式编程 / 擦除期间，除了挂起命令外，写入设备的所有命令均被忽略。
- 在编程操作期间，不能使用安全硅扇区、自动选择、CFI 等功能。
- 硬件复位和 / 或断电会立即终止编程 / 擦除操作，为确保数据完整性，一旦设备返回读取模式，应重新启动编程 / 擦除命令序列。
- 对于单字编程操作，可以以任何序列和跨扇区边界进行编程。当使用写入缓冲器时，请参见 [第 26 页](#) 的 [写入缓冲器编程](#)。
- 可以对同一地址进行多次编程，而无需插入擦除。

7.7.1 单字编程

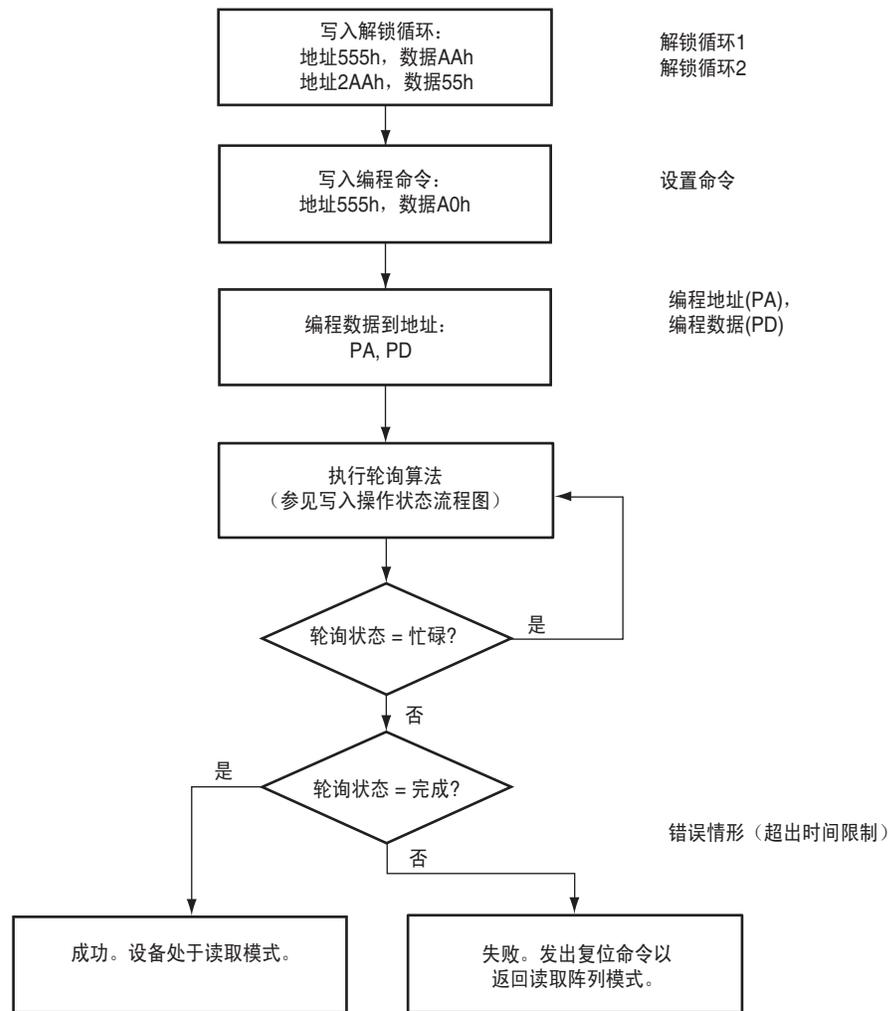
单字编程模式是一种闪存编程方法。在此模式下，使用四个闪存命令写入循环对各个闪存地址进行编程。此编程操作的数据宽度可以是 8 或 16 位。

虽然大多数 Spansion 设备均支持单字编程方法，但一般而言，不建议对支持写入缓冲器编程的设备使用单字编程。请参见 [第 67 页](#) 的 [表 12.1](#) 了解所需的总线循环，参见 [图 7.1](#) 的流程图。

嵌入式编程算法完成后，设备返回读取模式，地址不再锁存。系统可通过读取 DQ 状态位来确定编程操作的状态。有关这些状态位的信息，请参见 [第 36 页](#) 的 [写入操作状态](#)。

- 在编程期间，所有命令（编程挂起命令除外）均被忽略。
- 在编程操作期间，不能使用安全硅扇区、自动选择、CFI 等功能。
- 硬件复位会立即终止编程操作。一旦设备返回读取模式，应重新启动编程命令序列，以确保数据完整性。
- 可以对同一地址连续多次编程（例如，对同一个字的不同比特位进行多次改写）。

图 7.1 单字编程



软件功能和代码示例

表 7.6 单字 / 字节编程
(LLD 功能 = lld_ProgramCmd)

循环	操作	字节地址	字地址	数据
解锁循环 1	写入	基址 + AAh	基址 + 555h	00AAh
解锁循环 2	写入	基址 + 555h	基址 + 2AAh	0055h
编程设置	写入	基址 + AAh	基址 + 555h	00A0h
编程	写入	字节地址	字地址	数据

注
基址 = 基准地址, 即闪存首地址。

下面是使用单字编程功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息, 请参见 *Spansion Low Level Driver User's Guide* (在 www.spansion.com 提供)。

```

/* Example: Program Command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x00A0; /* write program setup command */
*( (UINT16 *)pa ) = data; /* write data to be programmed */
/* Poll for program completion */

```

7.7.2 写入缓冲器编程

写入缓冲器编程允许系统在一个编程操作中写入最多 32 个字。与标准的“字”编程算法相比, 这可以有效地加快字编程速度。写入缓冲器编程命令序列由前两个写入解锁循环来启动。第三个写入循环包含在编程的扇区地址写入的写入缓冲器加载命令。此时, 系统写入数值“写入字数减 1”, 这些字位置加载到编程的扇区地址的页面缓冲器中。这会告知设备有多少写入缓冲器地址加载了数据以及何时会发出“编程缓冲器到闪存”确认命令。要编程的位置数不能超出写入缓冲器的容量, 否则操作中止。(加载的数值 = 要编程的位置数减 1。例如, 如果系统编程 6 个地址位置, 则应向设备写入 05h。)

系统随后写入起始地址 / 数据组合。这个起始地址是要编程的第一个地址 / 数据对, 选择“写入缓冲器页面”地址。所有后续地址 / 数据对必须位于所选的写入缓冲器页面内。

使用地址 A_{MAX}-A5 来选择“写入缓冲器页面”。

加载到写入缓冲器中的所有地址 / 数据对的“写入缓冲器页面”地址必须相同。(这意味着写入缓冲器编程不能跨越多个“写入缓冲器页面”执行。也意味着写入缓冲器编程不能跨越多个扇区执行。如果系统试图加载所选“写入缓冲器页面”以外的编程数据, 操作将中止。)

写入起始地址 / 数据对后, 系统随后将其余地址 / 数据对写入到写入缓冲器中。

请注意, 如果一个写入缓冲器地址位置加载多次, 则每执行一次数据加载操作, “地址 / 数据对”计数器递减。而且, 在“编程缓冲器到闪存”确认命令之前在最后一个位置加载的数据是编程到设备中的数据。软件应负责确保避免同一个写入缓冲器位置多次加载。计数器递减是针对每次数据加载操作, 而不是每个唯一的写入缓冲器地址位置。一旦已加载指定数量的写入缓冲器地址, 系统必须随后在扇区地址写入“编程缓冲器到闪存”命令。任何其他地址 / 数据写入组合会中止写入缓冲器编程操作。应使用最后一个数据加载地址位置来作为检测编程操作状态时的地址。这样的话, 系统可以无需记录前面的数据加载地址, 在最后加载的地址位置发出编程确认命令, 然后在同一地址检查写入操作状态。在写入缓冲器编程期间, 应监控 DQ7、DQ6、DQ5、DQ2 和 DQ1 以确定设备状态。

写入缓冲器“嵌入式”编程操作可以通过标准的挂起 / 恢复命令来挂起。成功完成写入缓冲器编程后, 设备返回读取模式。

在下列任一情况下, 写入缓冲器编程序列将中止:

- 在“要编程的位置数”步骤期间, 加载的值超出页面缓冲器容量。
- 写入地址不在写入缓冲器加载命令所指定的扇区中。
- 在操作的“写入缓冲器数据加载”阶段, 地址 / 数据对未写入到“起始地址”选择的写入缓冲器页面。
- 在指定的“数据加载”循环次数后, 写入编程到缓冲器闪存命令之外的其他内容。

中止情形通过“加载的最后一个地址位置”的数据 $DQ1 = 1$ 、 $DQ7 = DATA\#$ 、 $DQ6 = TOGGLE$ 、 $DQ5 = 0$ 来指明。这表示写入缓冲器编程操作已中止。在解锁省略模式下使用写入缓冲器编程功能时，需要一个“写入到缓冲器中止复位”命令序列来退出该异常终止状态。请注意，在编程操作期间，不能使用安全硅扇区、自动选择、CFI 等功能。

写入缓冲器编程可以采用任何存储（或地址）位置序列。这些闪存设备支持对同一写入缓冲器地址范围执行多次写入缓冲器编程操作，而无需插入擦除。

需要编程多个字时，强烈建议使用写入缓冲器进行编程。

软件功能和代码示例

表 7.7 写入缓冲器编程
(LLD 功能已使用 = lld_WriteToBufferCmd, lld_ProgramBufferToFlashCmd)

循环	描述	操作	字节地址	字地址	数据
1	解锁	写入	基址 + AAAh	基址 + 555h	00AAh
2	解锁	写入	基址 + 555h	基址 + 2AAh	0055h
3	写入缓冲器加载命令	写入	扇区地址		0025h
4	写入字计数	写入	扇区地址		字计数 (N-1)h
加载到写入缓冲器中的字数 (N) 可以是 1 到 32 个字 (1 到 64 字节)。					
5 到 36	加载缓冲器字 N	写入	编程地址, 字 N		字 N
最后一个	写入缓冲器到闪存	写入	扇区地址		0029h

备注

1. 基址 = 基准地址。
2. 最后一个 = 写入缓冲器编程操作的最后一个循环; 根据写入的字数, 循环总数可以是 6 到 37。
3. 为达到最大效率, 建议向写入缓冲器中加载可能的最多字数 (N 个字)。

下面是使用写入缓冲器编程功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息, 请参见 *Spansion Low Level Driver User's Guide* (在 www.spansion.com 提供)。

```

/* Example:Write Buffer Programming Command */
/* NOTES:Write buffer programming limited to 16 words.*/
/* All addresses to be written to the flash in */
/* one operation must be within the same flash */
/* page.A flash page begins at addresses */
/* evenly divisible by 0x20.*/

UINT16 *src = source_of_data; /* address of source data */
UINT16 *dst = destination_of_data; /* flash destination address */
UINT16 wc = words_to_program -1; /* word count (minus 1) */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)sector_address ) = 0x0025; /* write write buffer load command */
*( (UINT16 *)sector_address ) = wc; /* write word count (minus 1) */

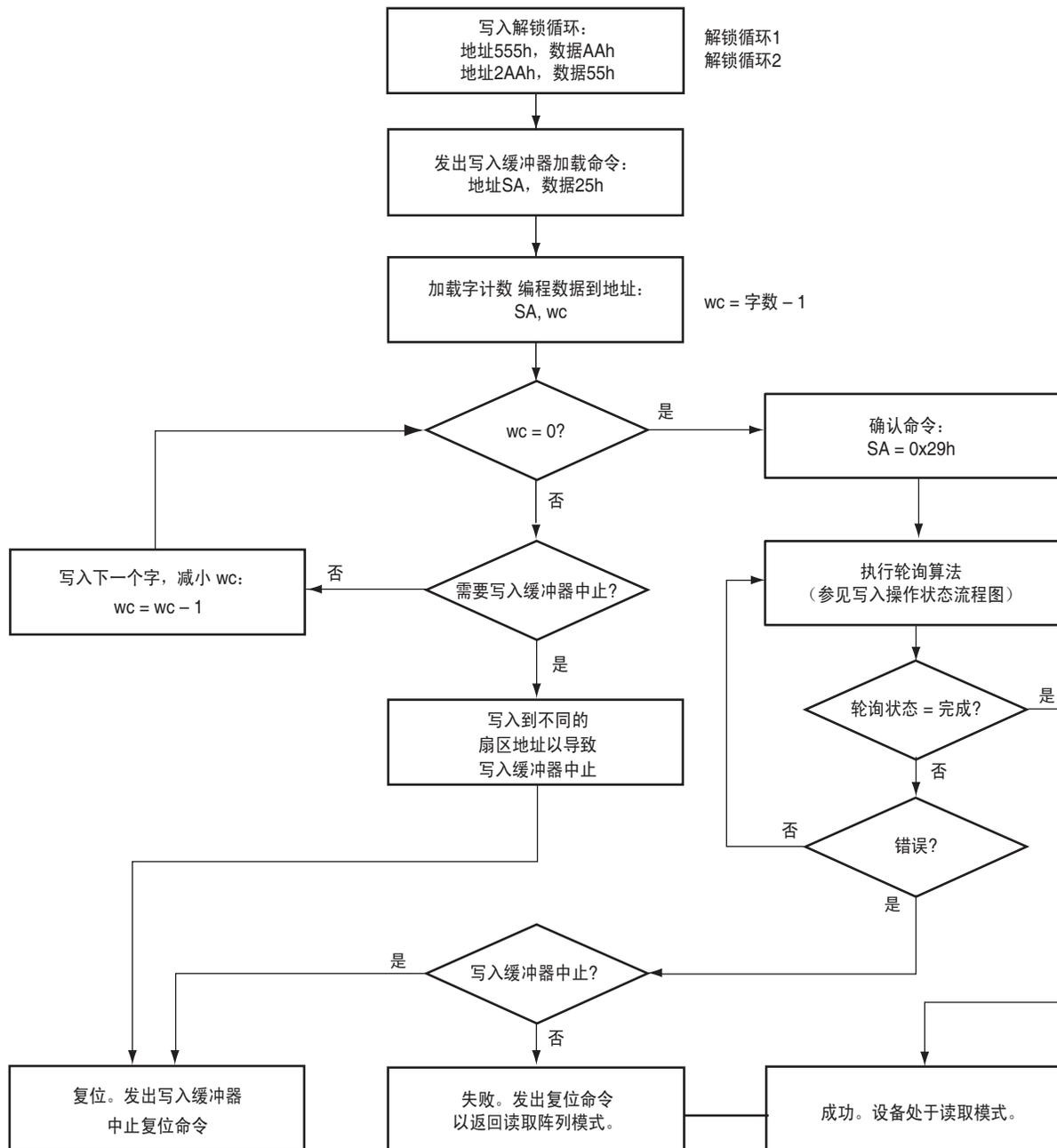
for (i=0;i<=wc;i++)
{
*dst++ = *src++; /* ALL dst MUST BE in same Write Buffer */
}

*( (UINT16 *)sector_address ) = 0x0029; /* write confirm command */
/* poll for completion */

/* Example:Write Buffer Abort Reset */
*( (UINT16 *)addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)addr + 0x555 ) = 0x00F0; /* write buffer abort reset */

```

图 7.2 写入缓冲器编程操作



7.7.3 扇区擦除

扇区擦除功能擦除存储阵列中的一个或多个扇区。（参见第 67 页的表 12.1 和图 7.3。）设备不需要系统在擦除之前对扇区预编程。嵌入式擦除算法在执行电擦除前，自动编程和检验整个存储器是否为全零数据组合。扇区擦除成功后，被擦除扇区内的所有位置均包含 FFFFh。在这些操作期间，系统不需要提供任何控制或时序。

写入命令序列后，会出现不小于 t_{SEA} 的扇区擦除超时。在超时期间，可以写入更多扇区地址。加载扇区擦除缓冲器可以通过任意序列完成，扇区数可以是从一个扇区到所有扇区。这些附加循环之间的时间必须小于 50 μ s。超出超时 (50 μ s) 之后的任何扇区擦除地址和命令可能被接受，也可能不被接受。在超时期间内，扇区擦除或擦除挂起之外的命令会使扇区返回读取模式。系统可以监控 DQ3，以确定扇区擦除定时器是否超时（参见章节 7.8.6。）超时从命令序列中最后 WE# 脉冲的上升沿开始计算。

嵌入式擦除算法完成后，扇区返回读取阵列数据，地址不再锁存。系统可通过读取擦除扇区中的 DQ7 或 DQ6/DQ2 来确定擦除操作的状态。有关这些状态位的信息，请参见章节 7.8。

一旦扇区擦除操作开始，仅擦除挂起命令有效。所有其他命令均被忽略。不过请注意，硬件复位可立即终止擦除操作。若发生这种情况，为确保扇区被正确擦除，一旦扇区返回读取阵列数据，应重新启动扇区擦除命令序列。

解锁省略功能允许主机系统向闪存设备发送编程命令，而无需预先在命令序列内写入解锁循环。有关解锁省略功能的详细信息，请参见章节 7.7.8。

图 7.3 介绍了擦除操作的算法。关于参数和时序图，请参见章节 11.7.5。

软件功能和代码示例

表 7.8 扇区擦除
(LLD 功能 = lld_SectorEraseCmd)

循环	描述	操作	字节地址	字地址	数据
1	解锁	写入	基址 + AAAh	基址 + 555h	00AAh
2	解锁	写入	基址 + 555h	基址 + 2AAh	0055h
3	设置命令	写入	基址 + AAAh	基址 + 555h	0080h
4	解锁	写入	基址 + AAAh	基址 + 555h	00AAh
5	解锁	写入	基址 + 555h	基址 + 2AAh	0055h
6	扇区擦除命令	写入	扇区地址	扇区地址	0030h

可选择的擦除扇区数量没有限制；命令必须在 50 μ s 内写入。

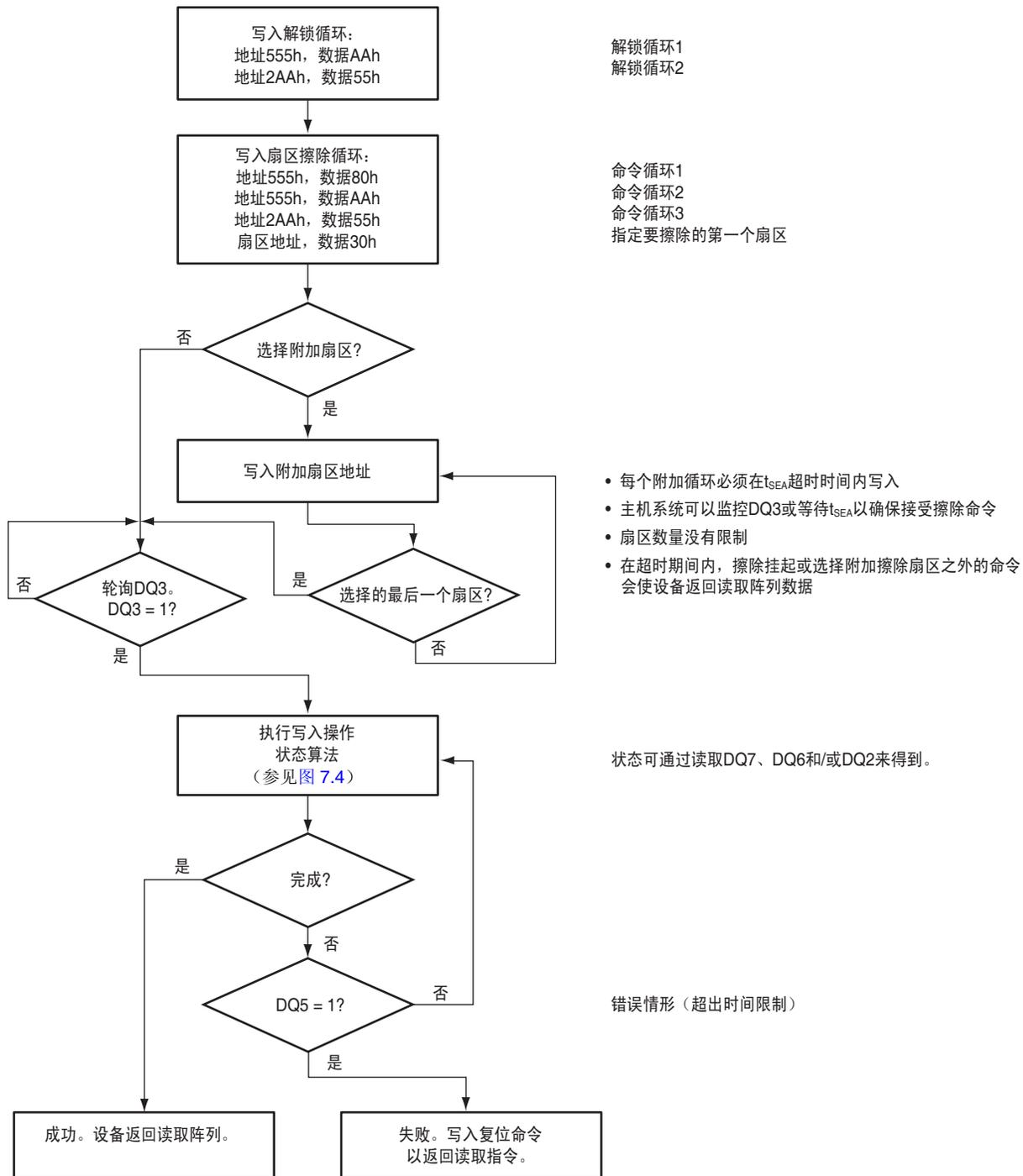
下面是使用扇区擦除功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```

/* Example:Sector Erase Command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x0080; /* write setup command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write additional unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write additional unlock cycle 2 */
*( (UINT16 *)sector_address ) = 0x0030; /* write sector erase command */

```

图 7.3 扇区擦除操作



备注

1. 关于擦除命令序列，请参见第 67 页的表 12.1。
2. 有关扇区擦除超时的信息，请参见第 38 页的 DQ3：扇区擦除超时状态指示。

7.7.4 芯片擦除命令序列

芯片擦除是一个六总线循环操作，如第 67 页的表 12.1 所示。这些命令调用嵌入式擦除算法，不需要系统在擦除之前进行预编程。嵌入式擦除算法在执行电擦除前，自动预编程和检验整个存储器是否为全零数据组合。芯片擦除成功后，芯片的所有位置均包含 FFFFh。在这些操作期间，系统不需要提供任何控制或时序。第 66 页的命令定义介绍芯片擦除命令序列的地址和数据要求。

嵌入式擦除算法完成后，该扇区返回读取模式，地址不再锁存。系统可使用 DQ7 或 DQ6/DQ2 来确定擦除操作的状态。有关这些状态位的信息，请参见“写入操作状态”。

解锁省略功能允许主机系统向闪存设备发送编程命令，而无需预先在命令序列内写入解锁循环。有关解锁省略功能的详细信息，请参见章节 7.7.8。

在芯片擦除操作期间写入的任何命令均被忽略。不过请注意，硬件复位可立即终止擦除操作。若发生这种情况，为确保整个阵列被正确擦除，一旦扇区返回读取阵列数据，应重新启动芯片擦除命令序列。

软件功能和代码示例

表 7.9 芯片擦除
(LLD 功能 = Ild_ChipEraseCmd)

循环	描述	操作	字节地址	字地址	数据
1	解锁	写入	基址 + AAAh	基址 + 555h	00AAh
2	解锁	写入	基址 + 555h	基址 + 2AAh	0055h
3	设置命令	写入	基址 + AAAh	基址 + 555h	0080h
4	解锁	写入	基址 + AAAh	基址 + 555h	00AAh
5	解锁	写入	基址 + 555h	基址 + 2AAh	0055h
6	芯片擦除命令	写入	基址 + AAAh	基址 + 555h	0010h

下面是使用芯片擦除功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```

/* Example:Chip Erase Command */
/* Note:Cannot be suspended */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x0080; /* write setup command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write additional unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write additional unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x0010; /* write chip erase command */
    
```

7.7.5 擦除挂起 / 擦除恢复命令

擦除挂起命令允许系统中断扇区擦除操作，然后从未被选择擦除的任何扇区读取数据或向其编程数据。当写入此命令时，扇区地址“不用考虑”。此命令仅在扇区擦除操作期间有效，包括扇区擦除命令序列期间的最小 t_{SEA} 超时期间。擦除挂起命令如果在芯片擦除操作期间写入，将被忽略。

在扇区擦除操作期间写入擦除挂起命令时，设备需要最长 20 μs （典型值 5 μs ）来挂起擦除操作。不过，在扇区擦除超时期间写入擦除挂起命令时，设备会立即终止超时期间并挂起擦除操作。

擦除操作被挂起后，设备进入擦除挂起读取模式。系统可以从未被选择擦除的任何扇区读取数据或向其编程数据。（设备“擦除挂起”所有被选择擦除的扇区。）读取擦除挂起扇区内的任何地址时，会在 DQ7-DQ0 上生成状态信息。系统可以同时使用 DQ7 或 DQ6 和 DQ2 来确定扇区是正在擦除还是擦除挂起。有关这些状态位的信息，请参见表 7.35。

擦除挂起模式下的编程操作完成后，设备返回擦除挂起读取模式。系统可以使用写入操作状态位来确定编程操作的状态，就像在执行标准编程操作一样。

在擦除挂起读取模式下，系统也可以发出自动选择命令序列。有关的详细信息，请参见第 26 页的写入缓冲器编程和第 21 页的自动选择。

为恢复扇区擦除操作，系统必须写入擦除恢复命令。当写入此命令时，擦除挂起的扇区的地址“不用考虑”。进一步写入的恢复命令被忽略。在芯片恢复擦除后，方可写入另一个擦除挂起命令。

软件功能和代码示例

表 7.10 擦除挂起
(LLD 功能 = lld_EraseSuspendCmd)

循环	操作	字节地址	字地址	数据
1	写入	基址 + XXXh	基址 + XXXh	00B0h

下面是使用擦除挂起功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```
/* Example:Erase suspend command */
*( (UINT16 *)base_addr ) = 0x00B0; /* write suspend command */
```

表 7.11 擦除恢复
(LLD 功能 = lld_EraseResumeCmd)

循环	操作	字节地址	字地址	数据
1	写入	扇区地址	扇区地址	0030h

下面是使用擦除恢复功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```
/* Example:Erase resume command */
*( (UINT16 *)sector_addr ) = 0x0030; /* write resume command */
/* The flash needs adequate time in the resume state */
```

7.7.6 编程挂起 / 编程恢复命令

编程挂起命令允许系统中断嵌入式编程操作或“写入缓冲器”编程操作，以便能够从未挂起的扇区中读取数据。在编程过程中写入编程挂起命令后，设备在最长 15 μs（典型值 5 μs）内中止编程操作，并更新状态位。当写入编程挂起命令时，不用考虑地址。

编程操作挂起后，系统可以读取任何未挂起的扇区中的阵列数据。当擦除被挂起时，编程挂起命令也可以在编程操作期间发出。在此情况下，可以从擦除挂起或编程挂起扇区之外的任意地址读取数据。如果需要从安全硅扇区区域读取，则用户必须使用正确的命令序列才能进入和退出此区域。

当设备处于编程挂起模式时，系统也可以写入自动选择命令序列。设备允许读取挂起的扇区中的自动选择代码，因为这些代码不存储在存储阵列中。当设备退出自动选择模式时，设备返回编程挂起模式，并做好执行另一个有效操作的准备。有关的详细信息，请参见第 21 页的 *自动选择*。

写入编程恢复命令后，设备返回编程。系统可以使用写入操作状态位来确定编程操作的状态，就像在执行标准编程操作一样。有关的详细信息，请参见第 36 页的 *写入操作状态*。

系统必须写入编程恢复命令（地址位“不用考虑”），才能退出编程挂起模式并继续编程操作。进一步写入的编程恢复命令被忽略。在设备恢复编程后，方可写入另一个编程挂起命令。

软件功能和代码示例

表 7.12 编程挂起
(LLD 功能 = Ild_ProgramSuspendCmd)

循环	操作	字节地址	字地址	数据
1	写入	基址 + XXXh	基址 + XXXh	00B0h

下面是使用编程挂起功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```
/* Example: Program suspend command */
*( (UINT16 *)base_addr ) = 0x00B0; /* write suspend command */
```

表 7.13 编程恢复
(LLD 功能 = Ild_ProgramResumeCmd)

循环	操作	字节地址	字地址	数据
1	写入	基址 + XXXh	基址 + XXXh	0030h

下面是使用编程恢复功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```
/* Example: Program resume command */
*( (UINT16 *)base_addr ) = 0x0030; /* write resume command */
```

7.7.7 加速编程

加速单字编程和写入缓冲器编程操作通过 WP#/ACC 针脚来启用。此方法快于标准的编程命令序列。

注

加速编程功能不能对一个扇区使用 10 次以上。

如果系统在此输入上加 V_{HH} ，则设备自动进入前述的解锁省略模式，并使用输入上的较高电压缩短编程操作所需的时间。系统随后可以使用解锁省略模式提供的写入缓冲器加载命令序列。请注意，如果在解锁省略模式下需要“写入到缓冲器中止复位”，则必须使用完整的 3 循环 RESET 命令序列来复位设备。如果在嵌入式编程操作完成时从 ACC 输入去除 V_{HH} ，设备将返回正常操作。

- 扇区必须在 WP#/ACC 提升至 V_{HH} 之前进行解锁。
- 除了加速编程操作外，WP#/ACC 针脚不得处于 V_{HH} ，否则设备可能损坏。
- 建议在加电序列完成后才在 WP#/ACC 上加 V_{HH} 。此外，建议在 V_{CC}/V_{IO} 下电之前，将 WP#/ACC 上的 V_{HH} 拉到 V_{IH}/V_{IL} 。

7.7.8 解锁省略

此设备提供解锁省略模式，可缩短编程命令。一旦设备进入解锁省略模式，只需两个写入循环（而不是正常的四个循环）即可编程数据。

此模式不使用标准编程命令序列所需的前两个解锁循环，从而缩短总计编程时间。第 66 页的命令定义介绍解锁省略命令序列的要求。

在解锁省略模式下，只有读取、编程、写入缓冲器编程、写入到缓冲器中止复位、解锁省略扇区擦除、解锁省略芯片擦除、以及解锁省略复位等命令有效。为了退出解锁省略模式，系统必须发出两循环解锁省略复位命令序列。第一个循环地址“不用考虑”，数据 90h。第二个循环只需包含数据 00h。扇区随后返回读取模式。

软件功能和代码示例

下面是使用解锁省略进入、编程、以及退出功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（即将在 www.spansion.com 提供）。

表 7.14 解锁省略进入
(LLD 功能 = lld_UnlockBypassEntryCmd)

循环	描述	操作	字节地址	字地址	数据
1	解锁	写入	基址 + AAAh	基址 + 555h	00AAh
2	解锁	写入	基址 + 555h	基址 + 2AAh	0055h
3	进入命令	写入	基址 + AAAh	基址 + 555h	0020h

```

/* Example:Unlock Bypass Entry Command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x0020; /* write unlock bypass command */
/* At this point, programming only takes two write cycles.*/
/* Once you enter Unlock Bypass Mode, do a series of like */
/* operations (programming or sector erase) and then exit */
/* Unlock Bypass Mode before beginning a different type of */
/* operations.*/

```

表 7.15 解锁省略编程
(LLD 功能 = lld_UnlockBypassProgramCmd)

循环	描述	操作	字节地址	字地址	数据
1	编程设置	写入	基址 + XXXh	基址 + XXXh	00A0h
2	编程命令	写入	编程地址	编程地址	编程数据

```

/* Example:Unlock Bypass Program Command */
/* Do while in Unlock Bypass Entry Mode!*/
*( (UINT16 *)base_addr ) = 0x00A0; /* write program setup command */
*( (UINT16 *)pa ) = data; /* write data to be programmed */
/* Poll until done or error.*/
/* If done and more to program, */
/* do above two cycles again.*/

```

表 7.16 解锁省略复位
(LLD 功能 = lld_UnlockBypassResetCmd)

循环	描述	操作	字节地址	字地址	数据
1	复位循环 1	写入	基址 + XXXh	基址 + XXXh	0090h
2	复位循环 2	写入	基址 + XXXh	基址 + XXXh	0000h

```
/* Example:Unlock Bypass Exit Command */
*( (UINT16 *)base_addr ) = 0x0090;
*( (UINT16 *)base_addr ) = 0x0000;
```

7.8 写入操作状态

设备提供多个数位来确定编程或擦除操作的状态。以下部分介绍 DQ1、DQ2、DQ3、DQ5、DQ6 和 DQ7 的功能。

7.8.1 DQ7: Data# 轮询

Data# 轮询位 DQ7 向主机系统指明嵌入式编程或擦除算法是正在执行中还是已完成，以及设备是否处于擦除挂起。Data# 轮询在命令序列中最后 WE# 脉冲的上升沿之后生效。请注意，对于写入缓冲器编程，Data# 轮询只对编程期间写入缓冲器页面内正在编程的最后一个字有效。如果对写入缓冲器页面中正在编程的最后一个字之外的任何字读取 Data# 轮询状态，将返回错误的状态信息。

在嵌入式编程算法期间，设备在 DQ7 输出为 DQ7 编程的数据的补码。此 DQ7 状态还应用于擦除挂起期间编程操作的状态检查。嵌入式编程算法完成后，设备输出为 DQ7 编程的数据。系统必须提供编程地址，才能读取 DQ7 上的有效状态信息。如果编程地址位于受保护的扇区内，则 DQ7 上的 Data# 轮询有效，随后该扇区返回读取模式。

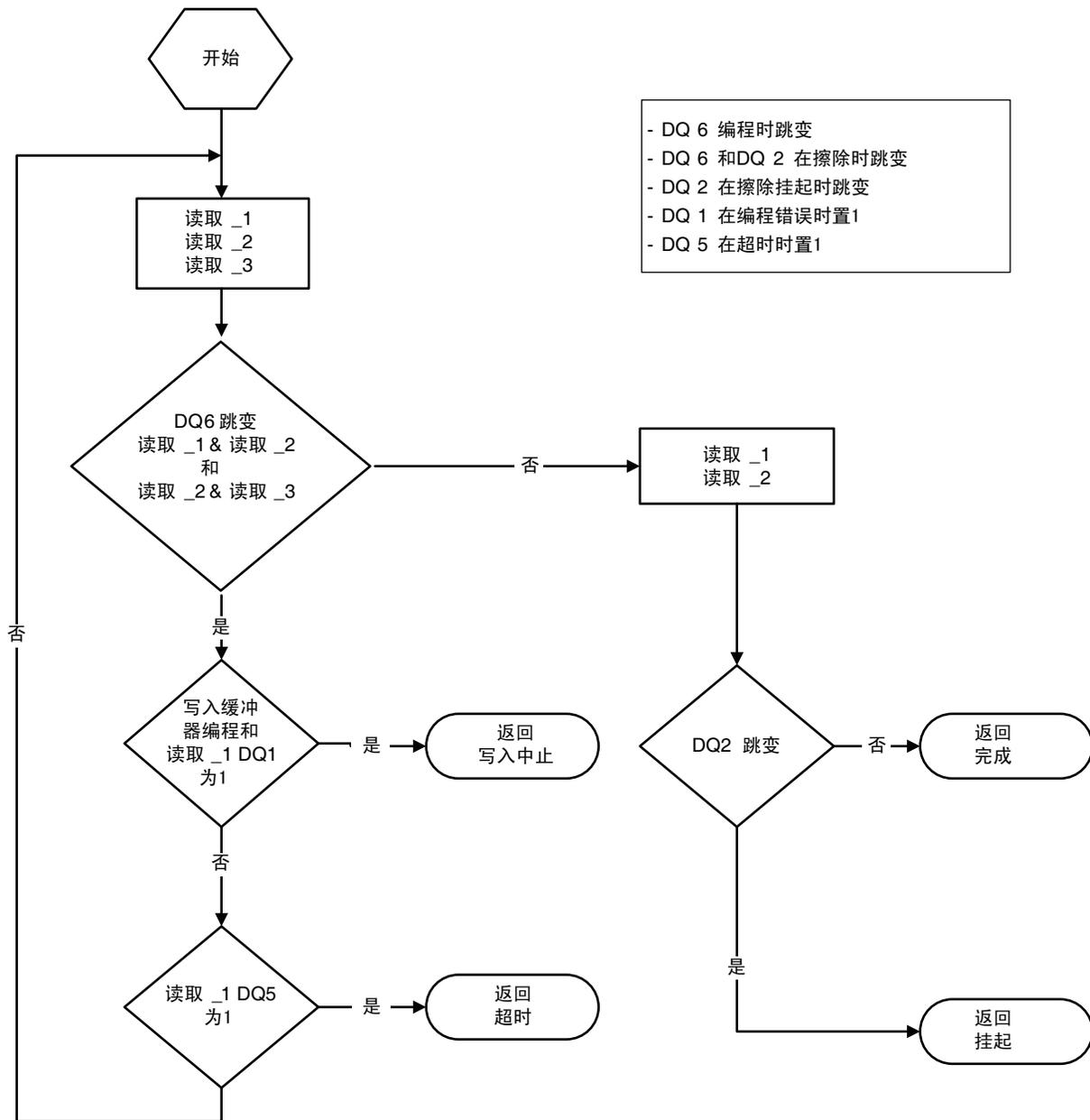
在嵌入式擦除算法期间，Data# 轮询在 DQ7 上生成“0”。当嵌入式擦除算法完成时，或者如果设备进入擦除挂起模式，Data# 轮询在 DQ7 上生成 1。系统必须提供一个位于所选任一擦除扇区内的地址，以读取 DQ7 上的有效状态信息。

擦除命令序列写入后，如果所选擦除扇区全部受到保护，DQ7 上的 Data# 轮询保持有效约 100 μs，然后设备返回读取模式。如果所选扇区不是全部受保护，则嵌入式擦除算法擦除不受保护的扇区，忽略所选的但受保护的扇区。不过，如果系统在受保护扇区内的一个地址读取 DQ7，状态可能无效。

在嵌入式编程或擦除操作即将完成前，当输出启用 (OE#) 拉低时，DQ7 可能与 DQ6-DQ0 异步变化。即，设备可能从提供状态信息变为在 DQ7 上提供有效数据。根据系统对 DQ7 输出进行抽样的时间，可能读取状态或有效数据。即使设备已完成编程或擦除操作并且 DQ7 为有效数据，DQ6-DQ0 上的数据输出仍可能无效。DQ7-DQ0 上的有效数据出现在后续的读取循环。

有关的详细信息，请参见以下内容：表 7.17，显示 DQ7 上 Data# 轮询的输出。图 7.4，显示 Data# 轮询算法；图 11.7，显示 Data# 轮询时序图。

图 7.4 写入操作状态流程图



7.8.2 DQ6: 跳变位 I

DQ6 上的跳变位 I 指明嵌入式编程或擦除算法是正在执行还是已完成，或者设备是否已进入擦除挂起模式。跳变位 I 可在任意地址读取，在命令序列最后 WE# 脉冲的上升沿之后（编程或擦除操作之前）或者在扇区擦除超时期间生效。

在嵌入式编程或擦除算法操作期间，对正在编程或擦除的任何地址的连续读取循环会导致 DQ6 跳变。当操作完成时，DQ6 停止跳变。

写入擦除命令序列后，如果被选择擦除的所有扇区均受保护，DQ6 跳变约 100 μ s，然后返回读取阵列数据。如果所选扇区不是全部受保护，则嵌入式擦除算法擦除不受保护的扇区，忽略所选的但受保护的扇区。

系统可以同时使用 DQ6 和 DQ2 来确定扇区是正在擦除还是擦除挂起。当设备正在擦除时（即嵌入式擦除算法正在执行中），DQ6 跳变。当设备进入擦除挂起模式时，DQ6 停止跳变。不过，系统还必须使用 DQ2 来确定哪个扇区正在擦除或擦除挂起。此外，系统还可以使用 DQ7（参见第 36 页的 DQ7: Data# 轮询）。

如果编程地址位于受保护的扇区内，在写入编程命令序列后，DQ6 跳变约 1 μ s，然后返回读取阵列数据。

DQ6 在擦除挂起编程模式下也会跳变，当嵌入式编程算法完成时停止跳变。

有关的详细信息，请参见以下内容：图 7.4、第 62 页的图 11.13 和表 7.17。

DQ6 上的跳变位 I 要求 OE# 或 CE# 先拉高然后拉低，以体现状态变更。

7.8.3 DQ2: 跳变位 II

DQ2 上的“跳变位 II”与 DQ6 一起使用时，指明特定的扇区是正在擦除（即嵌入式擦除算法正在执行）还是擦除挂起。跳变位 II 在命令序列中最后 WE# 脉冲的上升沿之后生效。当系统读取所选擦除扇区内的地址时，DQ2 跳变。但 DQ2 无法区分扇区是正在擦除还是擦除挂起。DQ6 通过比较，指明设备是正在擦除还是擦除挂起，但无法区分选择了哪些扇区进行擦除。因此，扇区和模式信息需要两个状态位。参见表 7.17 以比较 DQ2 和 DQ6 的输出。有关的详细信息，请参见第 62 页的图 11.14。

7.8.4 读取跳变位 DQ6/DQ2

每当系统开始读取跳变位状态时，它必须连续读取 DQ7-DQ0 至少两次，以确定跳变位是否跳变。通常，系统在第一次读取后会记录和存储跳变位的值。第二次读取后，系统会比较跳变位的新值和第一个值。如果跳变位未跳变，则设备已完成编程或擦除操作。系统可以在随后的读取循环读取 DQ7-DQ0 上的阵列数据。不过，如果最初两个读取循环后，系统确定跳变位仍跳变，系统也应注意 DQ5 的值是否为高（参见第 38 页的 DQ5: 超时）。若是，系统随后应再次确定跳变位是否跳变，因为 DQ5 变为高后跳变位可能停止跳变。如果跳变位不再跳变，则设备已成功完成编程或擦除操作。如果仍跳变，则设备未成功完成操作，系统必须写入复位命令以返回读取阵列数据。其余情况是系统在最初确定跳变位正跳变，并且 DQ5 尚未变为高。系统可能通过后续读取循环继续监控跳变位和 DQ5，以确定状态，如前一段所述。此外，它可以选择执行其他系统任务。在此情况下，当系统返回以确定操作状态时，必须在算法开始处启动。有关的详细信息，请参见图 7.4。

注

当检验存储扇区的写入操作（嵌入式编程 / 擦除）状态时，DQ6 和 DQ2 在一系列连续且临近的状态读取循环期间在高低状态之间跳变。为了正确识别此跳变行为，在连续状态位读取期间不得插入对其他存储扇区的读取访问。如果无法临时防止对其他存储扇区的读取，建议使用 DQ7 状态位作为一种替代方法来确定写入操作的有效或无效状态。

7.8.5 DQ5: 超时

DQ5 指明编程或擦除时间是否超出了指定的内部时钟计数限制。在这些情况下，DQ5 生成“1”，以指明编程或擦除循环未成功完成。如果系统试图对之前编程 0 的一个位置编程 1，则设备不会在 DQ5 上输出 1。只有擦除操作能将 0 改回 1。在此情况下，设备忽略收到错误指示从 0 编程为 1 的数位，而对其他收到正确请求从 1 变为 0 的数位进行编程。在编程期间，从 0 编程为 1 的尝试被屏蔽。在 DQ5 为 1，即超时的情况下，系统必须写入复位命令，才能返回读取模式（或擦除挂起读取模式，如果扇区之前处于擦除挂起编程模式）。

7.8.6 DQ3: 扇区擦除超时状态指示

写入扇区擦除命令序列后，系统可以读取 DQ3 以确定是否已开始擦除。（扇区擦除定时器不适用于芯片擦除命令。）如果选择附加扇区进行擦除，整个超时在每个附加扇区擦除命令之后也适用。当超时期间结束时，DQ3 从“0”切换到“1”。如果系统附加扇区擦除命令之间的时间可认定小于 t_{SEA} ，则系统不需要监控 DQ3。有关的详细信息，请参见第 30 页的扇区擦除。

扇区擦除命令写入后，系统应读取 DQ7（Data# 轮询）或 DQ6（跳变位 1）的状态以确保设备已接受命令序列，然后读取 DQ3。如果 DQ3 是“1”，则嵌入式擦除算法已开始；所有进一步命令（擦除挂起除外）会被忽略，直至擦除操作完成。如果 DQ3 是“0”，设备会接受附加扇区擦除命令。为确保命令已接受，系统软件应在每个后续扇区擦除命令之前和之后检查 DQ3 的状态。如果 DQ3 在第二次状态检查时为高，则最后一个命令可能尚未被接受。表 7.17 显示相对于其他状态位的 DQ3 状态。

7.8.7 DQ1：写入到缓冲器中止

DQ1 指明写入到缓冲器操作是否已中止。在这些情况下，DQ1 生成“1”。系统必须发出“写入到缓冲器中止复位”命令序列，以使设备返回读取阵列数据。有关的详细信息，请参见第 26 页的写入缓冲器编程。

表 7.17 写入操作状态

状态		DQ7 (注 2)	DQ6	DQ5 (注 1)	DQ3	DQ2 (注 2)	DQ1	RY/ BY#	
标准模式	嵌入式编程算法	DQ7#	跳变	0	不适用	不跳变	0	0	
	嵌入式擦除算法	0	跳变	0	1	跳变	不适用	0	
编程挂起模式	编程挂起读取	编程挂起的扇区	无效（不允许）					1	
		非编程挂起的扇区	数据					1	
擦除挂起模式	擦除挂起读取	擦除挂起的扇区	1	不跳变	0	不适用	跳变	不适用	1
		非擦除挂起的扇区	数据					1	
	擦除挂起编程（嵌入式编程）	DQ7#	跳变	0	不适用	不适用	不适用	0	
写入到缓冲器	忙碌（注 3）	DQ7#	跳变	0	不适用	不适用	0	0	
	中止（注 4）	DQ7#	跳变	0	不适用	不适用	1	0	

备注

- 当嵌入式编程、嵌入式擦除或写入到缓冲器操作超时，DQ5 变为 1。有关的详细信息，请参见第 38 页的 DQ5：超时。
- 当读取状态信息时，DQ7 和 DQ2 需要一个有效的地址。有关的详细信息，请参见相应的小节。
- 应使用 Data# 轮询算法来监控最后加载的写入缓冲器地址位置。
- 当设备已中止了写入到缓冲器操作时，DQ1 变为 1

7.9 写入命令 / 命令序列

在写入操作期间，当提供地址、命令和数据时，系统必须驱动 CE# 和 WE# 至 V_{IL} ，OE# 至 V_{IH} 。地址在 WE# 或 CE# 的最后一个下降沿被锁存，数据在 WE# 或 CE# 的第一个上升沿被锁存。擦除操作可以擦除一个扇区、多个扇区或整个设备。表 6.1 – 表 6.4 指明每个扇区占用的地址空间。设备地址空间分成统一的 64KW/128KB 扇区。扇区地址是为了唯一地选择一个扇区而需要的一组地址位。“直流特性”中的 I_{CC2} 表示写入模式的有效电流规格。“交流特性”包含写入操作的时序规格表和时序图。

7.9.1 RY/BY#

RY/BY# 是专用的漏极开路输出引脚，它指明嵌入式算法是正在执行还是已完成。RY/BY# 状态在命令序列中最后 WE# 脉冲的上升沿之后生效。由于 RY/BY# 是漏极开路输出，多个 RY/BY# 引脚可通过上拉电阻并行绑定至 V_{CC} 。此功能使得系统只需监控 RY/BY# 引脚（专用输出，由 CE# 控制而不是 OE# 控制），即可检测数据何时做好读取准备。

7.9.2 硬件复位

RESET# 输入提供一种硬件复位方法，可使设备返回读取阵列数据。当 RESET# 被驱动至至少 t_{RP} （RESET# 脉冲宽度）时，设备立即中止正在执行的任何操作，使所有输出三态化，复位配置寄存器，忽略 RESET# 脉冲期间的所有读取 / 写入命令。设备还使内部状态机返回读取阵列数据。

为确保数据完整性，一旦设备准备好接受另一个命令序列，应重新启动被中断的编程 / 擦除操作。

当 RESET# 保持在 V_{SS} 时，设备消耗 V_{CC} 复位电流 (I_{CC5})。如果 RESET# 保持在 V_{IL} 而不是 V_{SS} ，待机电流会增大。RESET# 可以绑定至系统复位电路，在系统复位时让系统从闪存读取引导固件。关于时序图，请参见第 57 页的图 11.7 和第 58 页的图 11.8。

7.9.3 软件复位

软件复位是命令集的一部分（参见第 67 页的表 12.1），也可使设备返回阵列读取模式，在下列情况下必须使用：

1. 退出自动选择模式
2. 当 DQ5 在写入状态操作期间变为高以指明编程或擦除循环未成功完成时
3. 退出扇区锁定 / 解锁操作。
4. 返回擦除挂起读取模式，如果设备之前处于擦除挂起模式。
5. 任何中止操作之后

软件功能和代码示例

表 7.18 复位
(LLD 功能 = lld_ResetCmd)

循环	操作	字节地址	字地址	数据
复位命令	写入	基址 + xxxh	基址 + xxxh	00F0h

注
基址 = 基准地址。

下面是使用复位功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```
/* Example:Reset (software reset of Flash state machine) */
*( (UINT16 *)base_addr ) = 0x00F0;
```

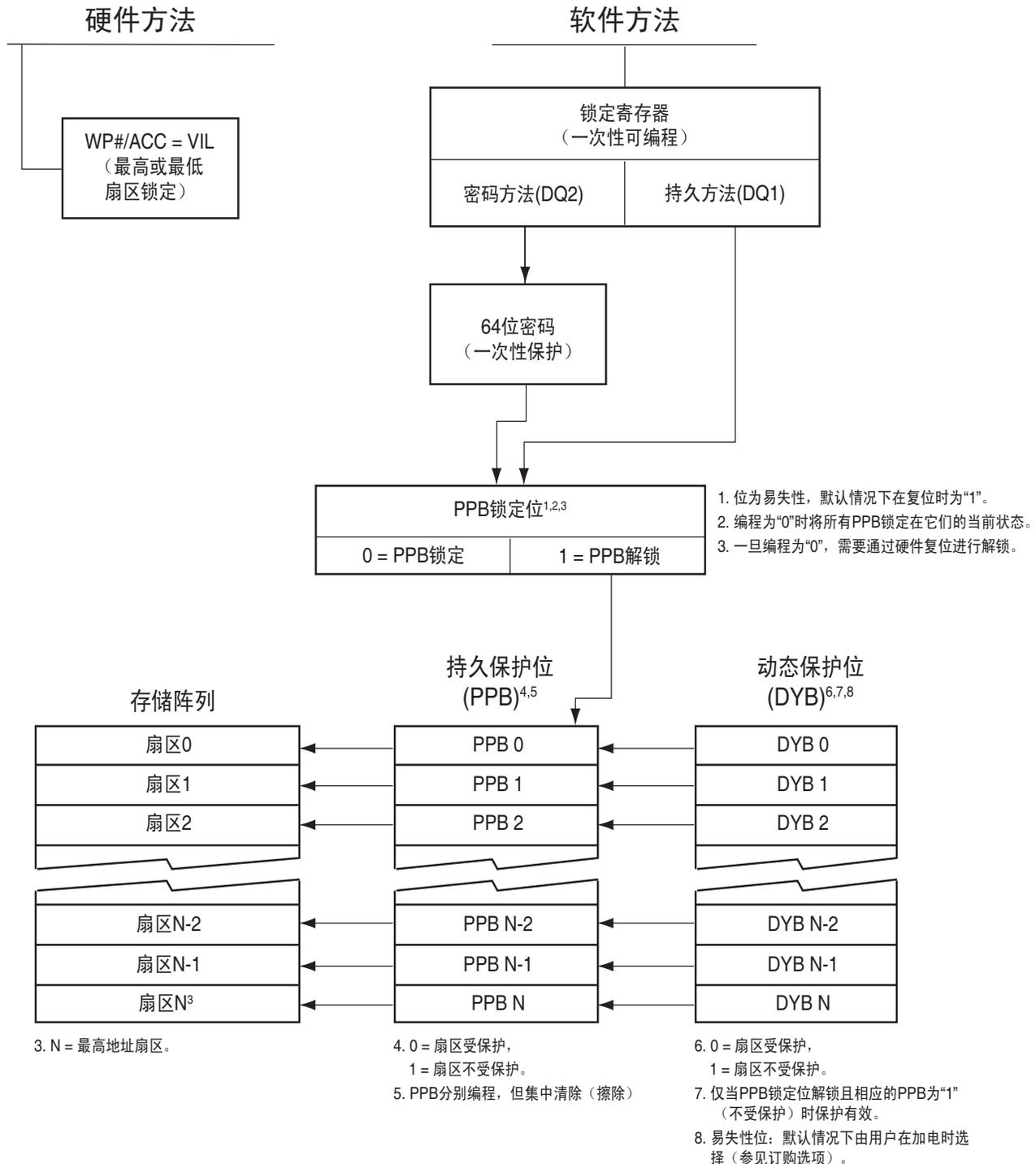
下面是使用复位命令时的其他注意事项：

- 此命令使扇区返回读取，地址位被忽略。
- 复位命令在编程和擦除操作期间被忽略。
- 复位命令可以在编程命令序列编程开始前的循环之间写入（第三个循环之前）。这会使系统正在写入的扇区返回读取模式。
- 如果编程命令序列写入到的扇区处于擦除挂起模式，写入复位命令会使该扇区返回擦除挂起读取模式。
- 复位命令可以在自动选择命令序列期间写入。
- 如果一个扇区从擦除挂起模式进入了自动选择模式，写入复位命令会使该扇区返回擦除挂起读取模式。
- 如果 DQ1 在写入缓冲器编程操作期间变为高，系统必须写入“写入到缓冲器中止复位”命令序列，才能使设备返回读取阵列数据。标准复位命令在此情况下不工作。
- 为了退出解锁省略模式，系统必须发出两循环解锁省略复位命令序列 [参见第 66 页的命令定义了解详细信息]。

8. 高级扇区保护 / 解保护

高级扇区保护 / 不保护功能禁止或允许对任何或所有扇区执行编程或擦除操作，可通过软件和 / 或硬件方法来实 现，两者之间互相独立。此部分介绍存储阵列中所存储数据的各种保护方法。图 8.1 是这些方法的概览。

图 8.1 高级扇区保护 / 解保护



8.1 锁定寄存器

出厂时，在默认情况下，所有设备在加电时处于持久模式，所有扇区不受保护，除非通过 **DYB** 订购选项专门选择（参见第 9 页的订购信息）。设备编程器或主机系统随后必须选择使用哪种扇区保护方法。通过对下面两个、一次性可编程、非易失性位中的一位进行编程（设为“0”），在相应模式下永久性锁定该部分：

- 锁定寄存器持久保护模式锁定位 (DQ1)
- 锁定寄存器密码保护模式锁定位 (DQ2)

表 8.1 锁定寄存器

DQ15-3	DQ2	DQ1	DQ0
不用考虑	密码保护模式 锁定位	持久保护模式 锁定位	安全硅扇区保护位

关于锁定寄存器位的编程，请参见第 68 页的表 12.2 和第 70 页的表 12.4。

备注

1. 如果选择密码模式，必须在设置相应的锁定寄存器位之前编程密码。
2. 写入锁定寄存器位命令设置进入命令后，在退出此模式前，对扇区 0 的读取和写入被禁止，但允许读取其他扇区。
3. 如果选择对两个锁定位同时编程（为零），操作将中止。
4. 对密码模式锁定位编程后，持久模式锁定位被永久性禁用，不允许对保护方案进行更改。类似地，对持久模式锁定位编程后，密码模式被永久性禁用。

选择扇区保护方法后，每个扇区可运行在下列三种状态之一：

1. **稳定锁定**。除非通过密码、硬件复位或关开机清除 **PPB** 锁定位，否则所选的扇区受到保护而不能重新编程。
2. **动态锁定**。所选的扇区受到保护，但可通过软件命令进行变更。
3. **解锁**。扇区不受保护，可以被擦除和 / 或编程。

这些状态由章节 8.2– 章节 8.5 中介绍的位类型来控制。

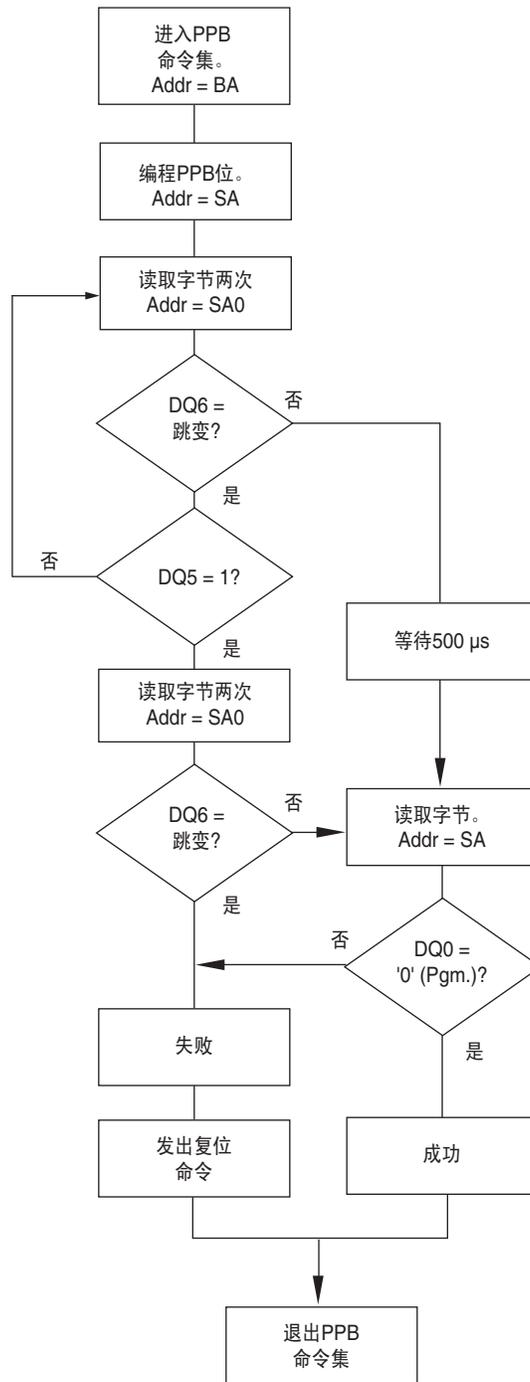
8.2 持久保护位

持久保护位对每个扇区来说唯一且具有非易失性，它们与闪存具有相同的耐久性。擦除前的预编程和检验由设备来处理，因此不需要系统监控。

备注

1. 每个 **PPB** 分别编程，但整体并行擦除。
2. 对一个扇区编程 **PPB** 时，可以从扇区 0（用于 **Data#** 轮询）以及正在编程扇区 **PPB** 的扇区之外的任何其他扇区读取阵列数据。
3. 进入命令禁止对所选的扇区进行读取和写入。
4. 读取该扇区会返回该扇区的 **PPB** 状态。
5. 所有读取必须使用读取模式来执行。
6. 特定扇区地址（**A25-A16 GL01GP**、**A24-A16 GL512P**、**A23-A16 GL256P**、**A22-A16 GL128P**）与编程命令在同一时间写入。
7. 如果设置了 **PPB** 锁定位，**PPB** 编程或擦除命令不执行，直至超时而编程或擦除 **PPB**。
8. 没有方法可以个别地擦除特定 **PPB**，此操作不需要特定扇区地址。
9. 执行后必须发出退出命令，以使设备返回读取模式并对扇区 0 重新启用读取和写入。
10. 特定扇区的 **PPB** 的编程状态可通过向设备写入 **PPB** 状态读取命令来检验，如图 8.2 的流程图所示。

图 8.2 PPB 编程算法



8.2.1 动态保护位

动态保护位是易失性的，对于每个扇区来说是唯一的，可分别进行修改。DYB 只控制那些 PPB 被清除（擦除为“1”）的非保护扇区的保护方案。通过发出 DYB 设置或清除命令序列，DYB 被设置（编程为“0”）或清除（擦除为“1”），从而相应地将每个扇区置于受保护或不受保护状态。此功能允许软件方便地保护扇区，以免意外更改；但并不禁止在需要更改时方便地去除保护。

备注

1. DYB 可以在需要时随时设置（编程为“0”）或清除（擦除为“1”）。当部件出厂时，PPB 被清除（擦除为“1”）；在加电或复位时，可根据所选的订购选项设置或清除 DYB。
2. 如果选择在加电后清除 DYB 选项（擦除为“1”），则根据扇区的 PPB 状态，可以修改扇区（参见表 8.2）。
3. 如果选择在加电后设置 DYB 选项（编程为“0”），扇区将处于受保护状态。
4. 在部分扇区被持久锁定的同时，其余扇区可处于动态状态。
5. 动态扇区的 DYB 设置或清除命令分别是指扇区的受保护或不受保护状态。不过，如需改变持久锁定扇区的状态，则需要另外一些步骤。首先，必须通过设备关开机或硬件复位清除 PPB 锁定位。接下来，可以改变 PPB 以反映所需的设置。再次设置 PPB 锁定位可锁定 PPB，设备恢复正常运行。
6. 为达到最佳保护效果，建议在引导代码的开始阶段执行 PPB 锁定位设置命令，并通过保持 $WP\#/ACC = V_{IL}$ 来保护引导代码。请注意，PPB 和 DYB 位在 $WP\#/ACC = V_{HH}$ 时具有相同的功能，与 $ACC = V_{IH}$ 时一样。

8.3 持久保护位锁定位

持久保护位锁定位对于所有扇区来说是全局易失性位。设置（编程为“0”）时，它锁定所有 PPB；清除（编程为“1”）时，允许更改 PPB。每个设备只有一个 PPB 锁定位。

备注

1. 除非设备处于密码保护模式，否则没有软件命令序列可解锁此位；只有硬件复位或加电可清除此位。
2. 仅在所有 PPB 均配置为所需的设置后，才必须设置 PPB 锁定位（编程为“0”）。

8.4 密码保护方法

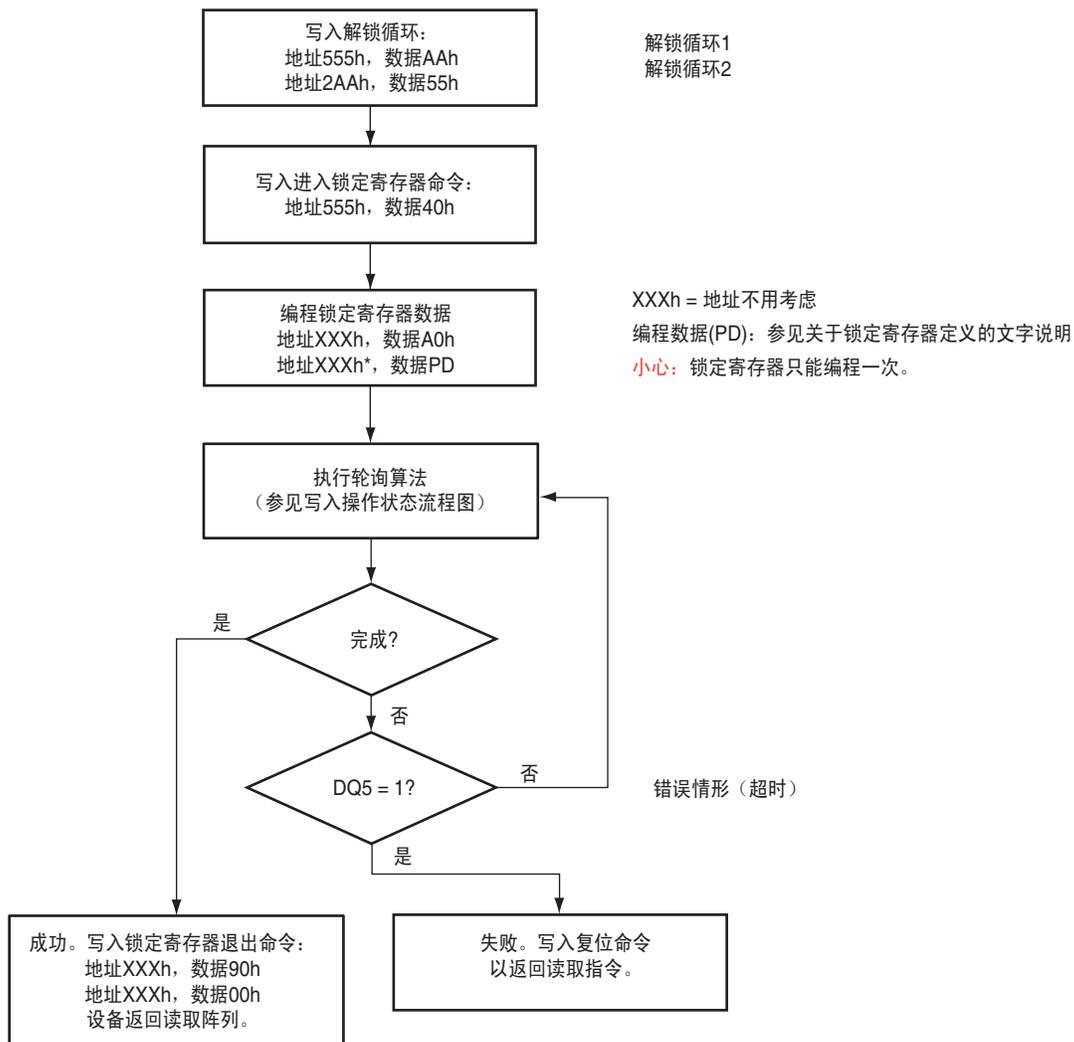
密码保护方法通过在解锁设备 PPB 锁定位时要求 64 位密码，可比持久扇区保护模式提供更高级别的安全性。除了此密码要求外，在加电和复位后，PPB 锁定位设为“0”以保持操作的密码模式。通过输入整个密码并成功执行密码解锁命令后，PPB 锁定位被清除，从而允许修改扇区 PPB。

备注

1. 编程密码不需要特殊的寻址顺序。写入和检验密码后，必须设置密码模式锁定位，以防止存取。
2. 密码编程命令只能编程“0”。若在一个比特位编程为“0”后编程“1”，会导致超时。
3. 出厂时，密码全部是“1”。
4. 所有 64 位密码组合均为有效密码。
5. 密码一经设置，没有方法检验密码是什么。
6. 一旦设置了密码模式锁定位，即可防止在数据总线上读取 64 位密码和进一步编程密码。
7. 密码模式锁定位不可擦除。
8. 两个低地址位 (A1–A0) 在密码读取、密码编程和密码解锁期间有效。
9. 只有输入准确的密码，方可解锁功能。
10. 每次发出密码解锁命令的时间不能快于 1 μ s，以防止黑客运行所有 64 位组合从而找到正确匹配的密码。
11. 向设备提供有效的 64 位密码后，需要约 1 μ s 来解锁设备。
12. 密码检验只可在密码编程操作期间进行。

13. 对密码区域的所有进一步命令均被禁止，所有操作均被忽略。
14. 如果在设置密码模式锁定位后密码丢失，没有办法可清除 PPB 锁定位。
15. 进入命令序列必须在任何操作之前发出，它会禁止对扇区 0 进行读取和写入。对于扇区 0 之外的其他扇区，可以执行读取和写入。
16. 如果用户试图编程或擦除一个受保护的扇区，设备会忽略命令并返回读取模式。
17. 对受保护扇区的编程或擦除命令会启用状态轮询并返回读取模式，而不修改受保护扇区的内容。
18. 对特定扇区的 DYB、PPB 和 PPB 锁定等编程可通过向设备写入相应的状态读取命令 DYB 状态、PPB 状态和 PPB 锁定状态来检验。

图 8.3 锁定寄存器编程算法



8.5 高级扇区保护软件示例

表 8.2 扇区保护方案：DYB、PPB 和 PPB 锁定位组合

唯一的设备 PPB 锁定位 0 = 锁定 1 = 不锁定	扇区 PPB 0 = 保护 1 = 不保护	扇区 DYB 0 = 保护 1 = 不保护	扇区保护状态
任意扇区	0	0	通过 PPB 保护
任意扇区	0	0	通过 PPB 保护
任意扇区	0	1	不保护
任意扇区	0	1	通过 DYB 保护
任意扇区	1	0	通过 PPB 保护
任意扇区	1	0	通过 PPB 保护
任意扇区	1	1	通过 DYB 保护
任意扇区	1	1	不保护

表 8.2 包含与扇区状态有关的 DYB、PPB 和 PPB 锁定位的所有可能组合。简而言之，如果 PPB 锁定位被锁定（设为“0”），则不允许更改 PPB。PPB 锁定位只可通过硬件复位或关开机来解锁（复位为“1”）。关于高级扇区保护功能的概述，另请参见图 8.1。

8.6 硬件数据保护方法

设备通过硬件控制提供两种主要类型的扇区级数据保护：

- 当 WP#/ACC 为 V_{IL} 时，最高或最低扇区被锁定（视设备而定）。

另外还有一些方法可通过硬件方式防止故意或意外擦除任何扇区。以下小节介绍这些方法：

8.6.1 WP#/ACC 方法

写入保护功能提供一种保护最外面一个扇区的硬件方法。此功能由 WP#/ACC 针脚提供，取代之前介绍的扇区保护 / 不保护方法。

如果系统在 WP#/ACC 针脚上提供 V_{IL} ，则设备禁止对最高或最低扇区执行编程和擦除功能，而不考虑是否使用了第 41 页的高级扇区保护 / 不保护中介绍的方法保护或不保护该扇区。

如果系统在 WP#/ACC 针脚上提供 V_{IH} ，则无论引导扇区此前设为受保护还是不受保护，设备均返回。即，这些扇区的扇区保护或不保护取决于它们此前是受保护还是不受保护。

WP#/ACC 针脚在命令序列执行期间必须保持稳定。WP# 有一个内部拉升；当断开连接时，WP# 设在 V_{IH} 。

注

如果设备处于待机模式时 WP#/ACC 为 V_{IL} ，最大输入负载电流会增大。有关的详细信息，请参见第 52 页的表 11.2。

8.6.2 低 V_{CC} 写入禁止

当 V_{CC} 小于 V_{LKO} 时，设备不接受任何写入循环。这可在 V_{CC} 加电和断电期间保护数据。

命令寄存器和所有内部编程 / 擦除电路被禁用，设备返回读取阵列数据。后续写入一直被忽略，直至 V_{CC} 大于 V_{LKO} 。系统必须向控制输入提供正确的信号，以防止当 V_{CC} 大于 V_{LKO} 时意外写入。

8.6.3 写入脉冲“假信号保护”

OE#、CE# 或 WE# 上短于 5 ns（典型值）的噪声脉冲不会启动写入循环。

8.6.4 加电写入禁止

如果在加电期间 WE# = CE# = RESET# = V_{IL} 且 OE# = V_{IH} ，设备不接受 WE# 上升沿上的命令。内部状态机在加电时自动返回读取模式。

9. 节能模式

9.1 待机模式

当系统不读取或写入设备时，可将设备置于待机模式。在此模式下，电流消耗显著降低，输出被置于高阻抗状态，而与 $OE\#$ 输入无关。当 $CE\#$ 和 $RESET\#$ 两个输入均保持在 $V_{CC} \pm 0.3 V$ 时，设备进入 CMOS 待机模式。在做好数据读取准备前，设备的读取访问需要标准的存取时间 (t_{CE})。如果在擦除或编程期间取消选择设备，设备将消耗有效电流，直至操作完成。“直流特性”中的 I_{CC4} 表示待机电流规格

9.2 自动睡眠模式

自动睡眠模式可将闪存设备能耗降至最低。在地址保持稳定 $t_{ACC} + 30 ns$ 后，设备自动启用此模式。自动睡眠模式与 $CE\#$ 、 $WE\#$ 和 $OE\#$ 等控制信号无关。当地址发生变化时，标准地址存取时序提供新数据。在睡眠模式下，输出数据处于闭止状态，系统可随时使用。章节 11.6 中的 I_{CC6} 是自动睡眠模式的电流规格。

9.3 硬件 $RESET\#$ 输入操作

$RESET\#$ 输入提供一种硬件复位方法，可使设备返回读取阵列数据。当 $RESET\#$ 被驱动至低至至少 t_{RP} 时，设备立即中止正在执行的任何操作，使所有输出三态化，忽略 $RESET\#$ 脉冲期间的所有读取 / 写入命令。设备还使内部状态机返回读取阵列数据。为确保数据完整性，一旦设备准备好接受另一个命令序列，应重新启动被中断的操作。

当 $RESET\#$ 保持在 $V_{SS} \pm 0.3 V$ 时，设备消耗 I_{CC} 复位电流 (I_{CC5})。如果 $RESET\#$ 保持在 V_{IL} 但不在 $V_{SS} \pm 0.3 V$ 范围内，待机电流会增大。

$RESET\#$ 可与系统复位电路绑定在一起，在系统复位时也复位闪存，从而允许系统读取闪存中的引导固件。

9.4 输出禁用 ($OE\#$)

当 $OE\#$ 输入为 V_{IH} 时，设备输出被禁用。输出被置于高阻抗状态。(RY/BY# 除外。)

10. 安全硅扇区闪存区域

安全硅扇区额外提供一个闪存区域，通过电子序列号 (ESN) 启用永久性部件标识。安全硅扇区的长度是 128 个字，128 字地址范围之外的所有安全硅读取均返回无效数据。安全硅扇区指示位 DQ7（位于自动选择地址 03h）用于指明安全硅扇区在出厂时是否锁定。

请注意下列一般情形：

- 在加电时或者在硬件复位后，设备返回向正常地址空间发送命令。
- 扇区 SA0 之外的读取返回存储阵列数据。
- 扇区 SA0 从存储阵列重新映射至安全硅扇区阵列。
- 一旦发出了安全硅扇区进入命令，则必须发出安全硅扇区退出命令才能退出安全硅扇区模式。
- 当设备正在执行嵌入式编程或嵌入式擦除算法时，无法存取安全硅扇区。
- 启用了安全硅扇区时，无法使用 ACC 功能和解锁省略模式。

表 10.1 安全硅扇区地址

安全硅扇区地址范围	用户可锁定	ESN 工厂锁定	ExpressFlash 工厂锁定
000000h-000007h	由用户决定	ESN	ESN 或由用户决定
000008h-00007Fh		不可用	由用户决定

10.1 工厂锁定的安全硅扇区

工厂锁定的安全硅扇区在出厂时总是受到保护，安全硅扇区指示位 (DQ7) 永久地设为“1”。一旦产品出厂后，这可防止克隆工厂锁定的部件，确保 ESN 和用户代码的安全。

这些设备可通过下面一种方式进行预编程：

- 随机的 8 字安全 ESN 只位于安全硅扇区内（地址 000000H - 000007H）
- 随机的安全 ESN 和用户代码都通过 Spansion 编程服务设置。

用户可以选择通过 Spansion 编程服务来编程他们的代码。Spansion 编程用户代码时可有或没有随机 ESN。这样，设备从 Spansion 工厂出厂时安全硅扇区被永久性锁定。有关如何使用 Spansion 编程服务的详细信息，请咨询您当地的代表。

10.2 用户可锁定的安全硅扇区

用户可锁定的安全硅扇区在出厂时总是不受保护（DQ7 设为“0”），用户可任意利用该扇区。如果不需要安全功能，安全硅扇区可当作附加闪存空间。

请注意下列事项：

- 一旦安全硅扇区区域受到保护，安全硅扇区指示位永久性设为“0”。
- 安全硅扇区可以读取任意次，但只能编程和锁定一次。安全硅扇区锁定必须谨慎使用，因为一旦锁定，没有任何方法可解锁安全硅扇区区域，也无法修改安全硅扇区存储空间内的任何数位。
- 启用了安全硅扇区时，无法使用加速编程 (ACC) 和解锁省略功能。
- 一旦安全硅扇区被锁定且通过检验，系统必须写入退出安全硅扇区区域命令序列，才能使设备返回至扇区 0 存储阵列。

10.3 安全硅扇区进入 / 退出命令序列

系统可通过发出三循环进入安全硅扇区命令序列来存取安全硅扇区区域。设备将一直处于安全硅扇区模式，直到系统发出四循环退出安全硅扇区命令序列。

有关两个命令序列的地址和数据要求，请参见第 66 页的命令定义 [安全硅扇区命令表]、附录第 67 页的表 12.1 到第 70 页的表 12.4。

安全硅扇区进入命令允许执行下列命令

- 读取用户和工厂安全硅区域
- 编程用户安全硅扇区

系统写入进入安全硅扇区命令序列后，可使用存储阵列中通常被扇区 SA0 占用的地址来读取安全硅扇区。此操作模式一直持续到系统发出退出安全硅扇区命令序列，或者直到设备断电。

软件功能和代码示例

下面是使用安全硅扇区进入、编程、以及退出命令的 C 功能和源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（即将在 www.spansion.com 提供）。

表 10.2 安全硅扇区进入
(LLD 功能 = lld_SecSiSectorEntryCmd)

循环	操作	字节地址	字地址	数据
解锁循环 1	写入	基址 + AAAh	基址 + 555h	00AAh
解锁循环 2	写入	基址 + 555h	基址 + 2AAh	0055h
进入循环	写入	基址 + AAAh	基址 + 555h	0088h

注
基址 = 基准地址。

```
/* Example:SecSi Sector Entry Command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x0088; /* write Secsi Sector Entry Cmd */
```

表 10.3 安全硅扇区编程
(LLD 功能 = lld_ProgramCmd)

循环	操作	字节地址	字地址	数据
解锁循环 1	写入	基址 + AAAh	基址 + 555h	00AAh
解锁循环 2	写入	基址 + 555h	基址 + 2AAh	0055h
编程设置	写入	基址 + AAAh	基址 + 555h	00A0h
编程	写入	字地址	字地址	数据字

注
基址 = 基准地址。

```
/* Example:Program Command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
```

```

*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x00A0; /* write program setup command */
*( (UINT16 *)pa ) = data; /* write data to be programmed */
/* Poll for program completion */

```

表 10.4 安全硅扇区退出
(LLD 功能 = lld_SecSiSectorExitCmd)

循环	操作	字节地址	字地址	数据
解锁循环 1	写入	基址 + AAAh	基址 + 555h	00AAh
解锁循环 2	写入	基址 + 555h	基址 + 2AAh	0055h
退出循环 3	写入	基址 + AAAh	基址 + 555h	0090h
退出循环 4	写入	基址 + XXXh	基址 + XXXh	0000h

注
基址 = 基准地址。

```

/* Example:SecSi Sector Exit Command */
*( (UINT16 *)base_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)base_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)base_addr + 0x555 ) = 0x0090; /* write SecSi Sector Exit cycle 3 */
*( (UINT16 *)base_addr + 0x000 ) = 0x0000; /* write SecSi Sector Exit cycle 4 */

```

11. 电气规格

11.1 绝对最大额定值

描述	额定值	
存储温度, 塑料包装	-65 °C 到 +150 °C	
环境温度 (加电)	-65 °C 到 +125 °C	
对地电压	所有输入和 I/O, 下面特别说明的除外 (注 1)	-0.5 V 到 $V_{CC} + 0.5 V$
	V_{CC} (注 1)	-0.5 V 到 +4.0 V
	V_{IO}	-0.5V 到 +4.0V
	A9 和 ACC (注 2)	-0.5 V 到 +12.5 V
输出短路电流 (注 3)	200 mA	

备注

1. 输入或 I/O 上的最小直流电压是 -0.5 V。在电压转换期间, 输入或 I/O 可能低于 $V_{SS} - 2.0 V$, 时间最长 20 ns。请参见图 11.1。输入或 I/O 上的最大直流电压是 $V_{CC} + 0.5 V$ 。在电压转换期间, 输入或 I/O 可能高于 $V_{CC} + 2.0 V$, 时间最长 20 ns。请参见图 11.2。
2. 引脚 A9 和 ACC 上的最小直流输入电压是 -0.5V。在电压转换期间, A9 和 ACC 可能高于 $V_{SS} - 2.0 V$, 时间最长 20 ns。请参见图 11.1。引脚 A9 和 ACC 上的最大直流电压是 +12.5 V, 可能高于 14.0 V, 时间最长 20 ns。
3. 不能有一个以上的输出同时对地短路。短路时间不应超过一秒。
4. 超过“绝对最大额定值”列出的那些值可能造成设备永久性损坏。这只是压力额定值; 并不暗示设备在这些值或者在此数据表操作部分所示值之上的任何其他情形下能正常运行。如果让设备长时间在绝对最大额定值情况下运行, 会影响设备的可靠性。

图 11.1 最大负过冲波形

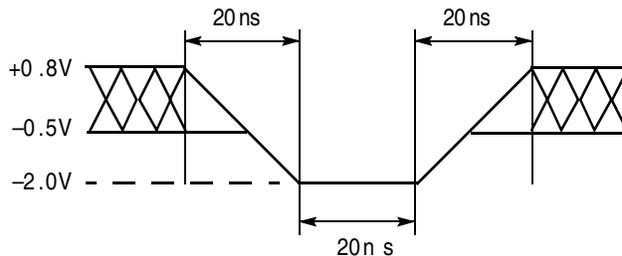
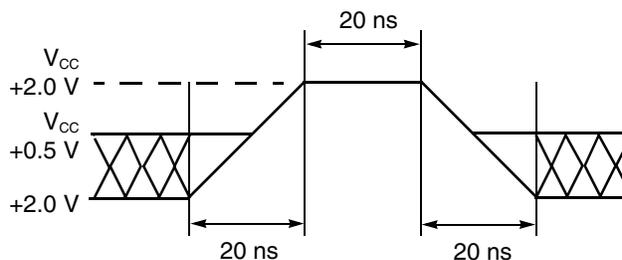


图 11.2 最大过冲波形



11.2 运行范围

规格		范围
环境温度 (TA), 工业 (I) 设备		-40 °C 到 +85 °C
环境温度 (TA), 商用 (C) 设备		0 °C 到 +85 °C
电源电压	V_{CC}	+2.7 V 到 3.6 V 或者 +3.0 V 到 3.6 V
V_{IO} 电源电压	V_{IO}	+1.65 V 到 V_{CC}

备注

1. 运行范围定义一些限值, 在这些限值之间可保证设备正常运行。
2. 另请参见第 9 页的订购信息。
3. 关于有效的 V_{CC}/V_{IO} 范围组合, 请参见第 9 页的订购信息。当 $V_{IO} = 1.8 V$ 时, I/O 不能在 3 V 工作。

11.3 测试条件

图 11.3 测试设置

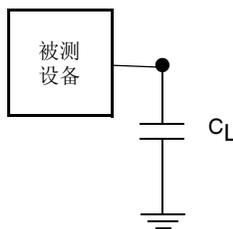


表 11.1 测试规格

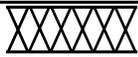
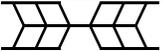
测试条件	所有速度	单位
输出负载电容, C_L (包括 jig 电容)	30	pF
输入上升和下降时间	5	ns
输入脉冲电平	0.0- V_{IO}	V
输入时序测量参考电平 (见注)	0.5 V_{IO}	V
输出时序测量参考电平	0.5 V_{IO}	V

注

如果 $V_{IO} < V_{CC}$, 参考电平为 0.5 V_{IO} 。

11.4 波形切换

波形	输入	输出
		稳定
		从 H 变为 L
		从 L 变为 H

波形	输入	输出
	不用考虑，任何更改都可以	更改，状态未知
	不适用	中间线是高阻抗状态 (High Z)

11.5 切换波形

图 11.4 输入波形和测量电平



注
如果 $V_{IO} < V_{CC}$ ，输入测量参考电平为 $0.5 V_{IO}$ 。

11.6 直流特性

表 11.2 S29GL-P 直流特性 (CMOS 兼容)

参数符号	参数描述 (注)	测试条件		最小	典型	最大	单位
I_{LI}	输入负载电流	$V_{IN} = V_{SS}$ 到 V_{CC} $V_{CC} = V_{CC\ max}$	WP/ACC			± 5.0	μA
			其他			± 2.0	
I_{LIT}	A9 输入负载电流	$V_{CC} = V_{CC\ max}$; A9 = 12.5 V				35	μA
I_{LO}	输出泄漏电流	$V_{OUT} = V_{SS}$ 到 V_{CC} , $V_{CC} = V_{CC\ max}$				± 1.0	μA
I_{CC1}	V_{CC} 有效读取电流 (1)	$CE\# = V_{IL}$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$, $f = 1\ MHz$			6	20	mA
		$CE\# = V_{IL}$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$, $f = 5\ MHz$			30	55	
		$CE\# = V_{IL}$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$, $f = 10\ MHz$			60	110	
I_{IO2}	V_{IO} 非有效输出	$CE\# = V_{IL}$, $OE\# = V_{IH}$			0.2	10	mA
I_{CC2}	V_{CC} 页面内读取电流 (1)	$CE\# = V_{IL}$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$, $f = 10\ MHz$			1	10	mA
		$CE\# = V_{IL}$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$, $f = 33\ MHz$			5	20	
I_{CC3}	V_{CC} 有效擦除 / 编程电流 (2、3)	$CE\# = V_{IL}$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$			50	90	mA
I_{CC4}	V_{CC} 待机电流	$CE\#, RESET\# = V_{CC} \pm 0.3\ V$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$ $V_{IL} = V_{SS} + 0.3\ V/-0.1\ V$,			1	5	μA
I_{CC5}	V_{CC} 复位电流	$V_{CC} = V_{CC\ max}$; $V_{IL} = V_{SS} + 0.3\ V/-0.1\ V$, $RESET\# = V_{SS} \pm 0.3\ V$			250	500	μA
I_{CC6}	自动睡眠模式 (4)	$V_{CC} = V_{CC\ max}$, $V_{IH} = V_{CC} \pm 0.3\ V$, $V_{IL} = V_{SS} + 0.3\ V/-0.1\ V$, $WP\#/ACC = V_{IH}$			1	5	μA
I_{ACC}	ACC 加速编程电流	$CE\# = V_{IL}$, $OE\# = V_{IH}$, $V_{CC} = V_{CC\ max}$, $WP\#/ACC = V_{HH}$	WP#/ACC 引脚		10	20	mA
			V_{CC} 引脚		50	80	
V_{IL}	输入低电压 (5)			-0.1		$0.3 \times V_{IO}$	V
V_{IH}	输入高电压 (5)			$0.7 \times V_{IO}$		$V_{IO} + 0.3$	V
V_{HH}	编程加速的电压	$V_{CC} = 2.7 - 3.6\ V$		11.5		12.5	V
V_{ID}	自动选择和临时扇区不保护的电压	$V_{CC} = 2.7 - 3.6\ V$		11.5		12.5	V
V_{OL}	输出低电压 (5)	$I_{OL} = 100\ \mu A$				$0.15 \times V_{IO}$	V
V_{OH}	输出高电压 (5)	$I_{OH} = -100\ \mu A$		$0.85 \times V_{IO}$			V
V_{LKO}	低 V_{CC} 锁定电压 (3)			2.3		2.5	V

备注

- 列出的 I_{CC} 电流通常小于 2 mA/MHz, $OE\#$ 为 V_{IH} 。
- I_{CC} 在嵌入式擦除或嵌入式编程或写入缓冲器编程正在执行有效。
- 未 100% 测试。
- 自动睡眠模式在地址保持稳定 $t_{ACC} + 30\ ns$ 后启用低功耗模式。
- $V_{IO} = 1.65 - 3.6\ V$
- $V_{CC} = 3\ V$ 和 $V_{IO} = 3\ V$ 或 $1.8\ V$ 。当 V_{IO} 为 $1.8\ V$ 时, I/O 引脚不能在 $3\ V$ 工作。

11.7 交流特性

11.7.1 S29GL-P 读取操作

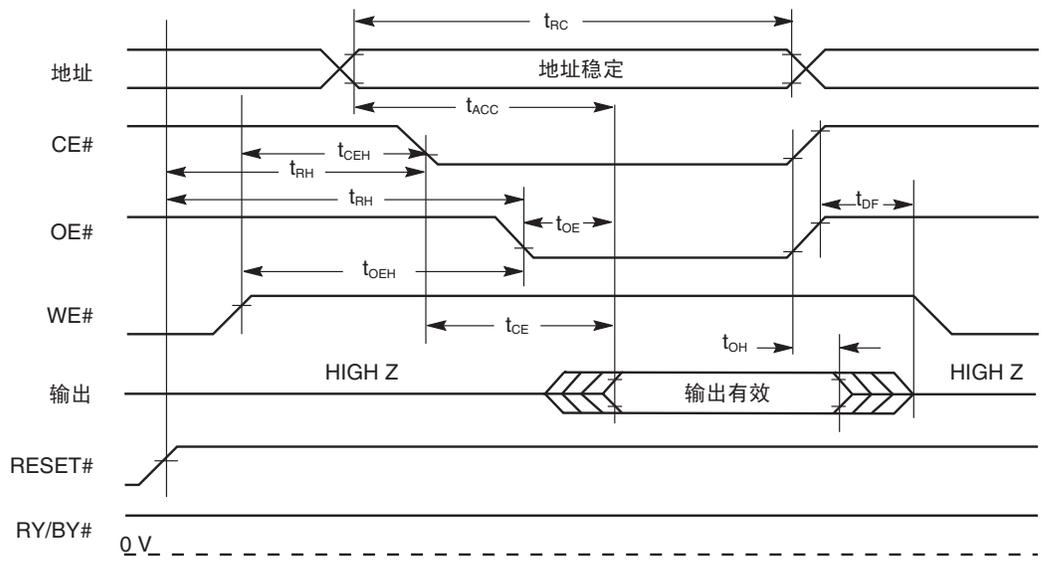
表 11.3 S29GL-P 读取操作

参数		描述 (注)	测试设置	速度选项					单位	
JEDEC	Std.			90	100	110	120	130		
t_{AVAV}	t_{RC}	读取循环时间	$V_{IO} = V_{CC} = 2.7 V$	最小	-	100	110	120	-	ns
			$V_{IO} = 1.65 V$ 到 V_{CC} , $V_{CC} = 3 V$		-	-	110	120	130	
			$V_{IO} = V_{CC} = 3.0 V$		90	100	110	-	-	
t_{AVQV}	t_{ACC}	地址到输出延迟 (1)	$V_{IO} = V_{CC} = 2.7 V$	最大	-	100	110	120	-	ns
			$V_{IO} = 1.65 V$ 到 V_{CC} , $V_{CC} = 3 V$		-	-	110	120	130	
			$V_{IO} = V_{CC} = 3.0 V$		90	100	110	-	-	
t_{ELQV}	t_{CE}	芯片启用到输出延迟 (2)	$V_{IO} = V_{CC} = 2.7 V$	最大	-	100	110	120	-	ns
			$V_{IO} = 1.65 V$ 到 V_{CC} , $V_{CC} = 3 V$		-	-	110	120	130	
			$V_{IO} = V_{CC} = 3.0 V$		90	100	110	-	-	
	t_{PACC}	页面存取时间		最大	25					ns
t_{GLQV}	t_{OE}	输出使能到输出延迟		最大	25					ns
t_{EHQZ}	t_{DF}	芯片启用到输出 High Z (3)		最大	20					ns
t_{GHQZ}	t_{DF}	输出启用到输出 High Z (3)		最大	20					ns
t_{AXQX}	t_{OH}	地址、CE# 或 OE# 中先发生者的输出保持时间		最小	0					ns
	t_{OEh}	输出启用保持时间 (3)	读取	最小	0					ns
			跳变和 Data# 轮询	最小	10					ns
	t_{CEH}	芯片启用保持时间	读取	最小	35					ns

备注

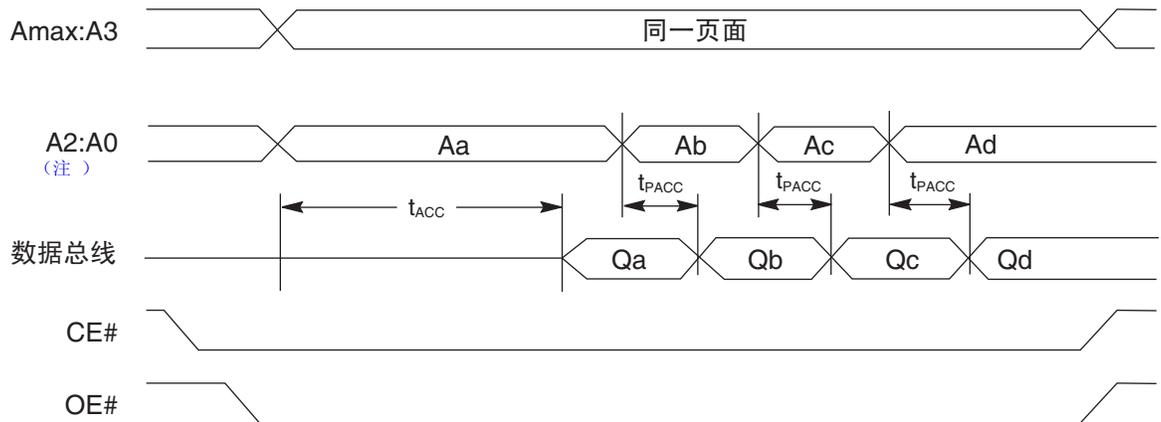
1. $CE\#, OE\# = V_{IL}$
2. $OE\# = V_{IL}$
3. 未 100% 测试。
4. 关于测试规格, 请参见图 11.3 和表 11.1。
5. 除非另有说明, 否则 110 ns 速度选项的交流规格基于 $V_{IO} = V_{CC} = 2.7 V$ 进行测试。测试 110 ns 速度选项的交流规格时, $V_{IO} = 1.8 V$ 和 $V_{CC} = 3.0 V$ 。

图 11.5 读取操作时序



注
对于图 11.5, 参数 t_{CEH} 和 t_{OEH} 取决于闪存写入操作之后的读取循环。

图 11.6 页面读取时序



注
图 11.6 显示的是字模式。字节模式时地址是 A2:A-1。

11.7.2 S29GL-P 硬件复位 (RESET#) 操作

表 11.4 硬件复位 (RESET#)

参数		描述		速度	单位
JEDEC	Std.				
	t_{Ready}	RESET# 引脚低 (在嵌入式算法期间), 至读取模式或写入模式	最小	35	μs
	t_{Ready}	RESET# 引脚低 (非嵌入式算法期间), 至读取模式或写入模式	最小	35	μs
	t_{RP}	RESET# 脉冲宽度	最小	35	μs
	t_{RH}	在读取之前复位高时间	最小	200	ns
	t_{RPD}	RESET# 低, 至待机模式	最小	10	μs
	t_{RB}	RY/BY# 恢复时间	最小	0	ns

图 11.7 复位时序

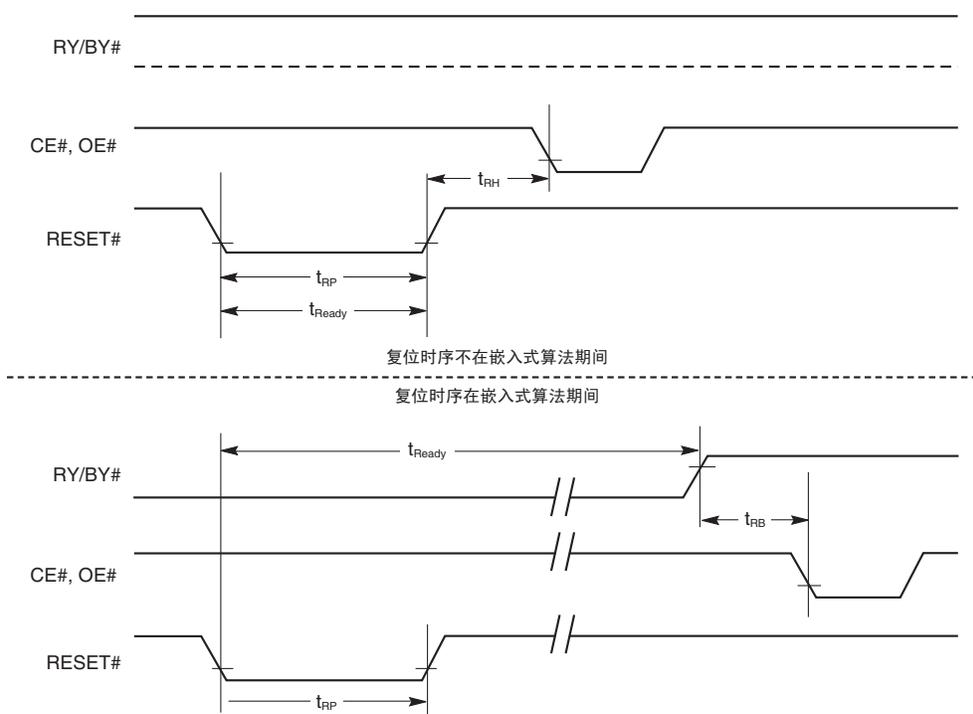


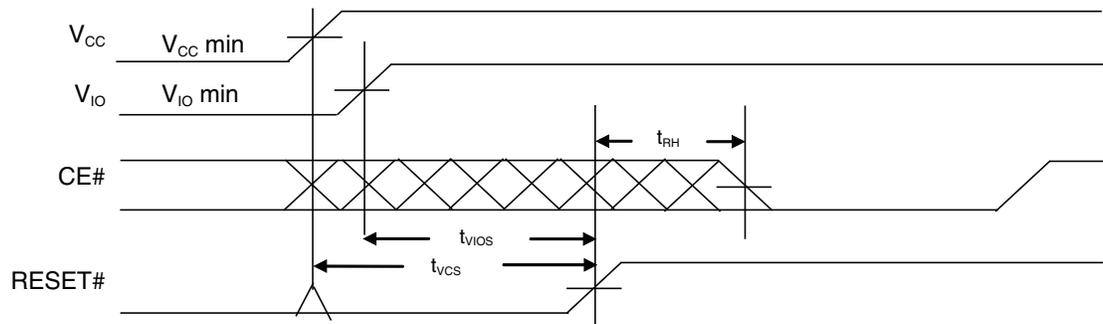
表 11.5 加电序列时序

参数	描述	速度	单位
t_{VCS}	从 V_{CC} (或最后一个复位脉冲) 的上升沿复位至 $RESET\#$ 的上升沿	最小	35 μs
t_{VIOS}	从 V_{IO} (或最后一个复位脉冲) 的上升沿复位至 $RESET\#$ 的上升沿	最小	35 μs
t_{RH}	在读取之前复位高时间	最小	200 ns

备注

1. $V_{IO} < V_{CC} + 200\text{ mV}$ 。
2. V_{IO} 和 V_{CC} 斜率在加电期间必须同步。
3. 如果 $RESET\#$ 不稳定时间达到 t_{VCS} 或 t_{VIOS} :
设备不允许任何读取和写入操作。
有效的读取操作返回 FFh。
需要执行硬件复位。
4. V_{CC} 最大加电电流 ($RST=V_{IL}$) 是 20 mA。

图 11.8 加电序列时序



11.7.3 S29GL-P 擦除和编程操作

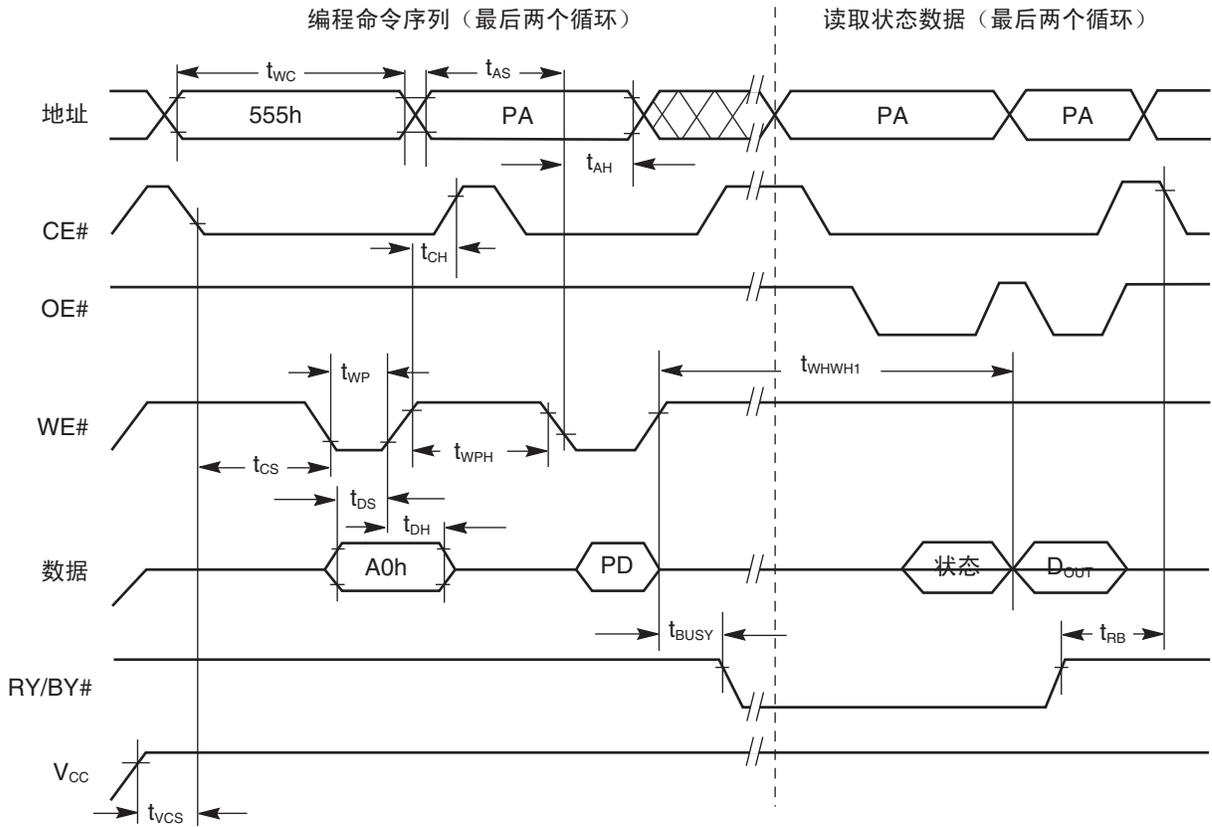
表 11.6 S29GL-P 擦除和编程操作

参数		描述		速度选项					单位
JEDEC	Std.			90	100	110	120	130	
t_{AVAV}	t_{WC}	写入循环时间 (注 1)	最小	90	100	110	120	130	ns
t_{AVWL}	t_{AS}	地址设置时间	最小	0					ns
	t_{ASO}	跳变位轮询期间地址设置时间到 OE# 低	最小	15					ns
t_{WLAX}	t_{AH}	地址保持时间	最小	45					ns
	t_{AHT}	在跳变位轮询期间自 CE# 或 OE# 高的地址保持时间	最小	0					ns
t_{DVWH}	t_{DS}	数据设置时间	最小	30					ns
t_{WHDX}	t_{DH}	数据保持时间	最小	0					ns
	t_{CEPH}	在跳变位轮询期间 CE# 高	最小	20					ns
	t_{OEPH}	在跳变位轮询期间输出启用高	最小	20					ns
t_{ELWL}	t_{CS}	CE# 设置时间	最小	0					ns
t_{WHEH}	t_{CH}	CE# 保持时间	最小	0					ns
t_{WLWH}	t_{WP}	写入脉冲宽度	最小	35					ns
t_{WHDL}	t_{WPH}	写入脉冲宽度高	最小	30					ns
t_{WHWH1}	t_{WHWH1}	写入缓冲器编程操作 (注 2、3)	典型	480					μ s
		有效写入缓冲器编程操作 (注 2、4)	每字	15					μ s
		加速的有效写入缓冲器编程操作 (注 2、4)	每字	13.5					μ s
		编程操作 (注 2)	字	60					μ s
		加速编程操作 (注 2)	字	54					μ s
t_{WHWH2}	t_{WHWH2}	扇区擦除操作 (注 2)	典型	0.5					秒
	t_{VHH}	V_{HH} 上升和下降时间 (注 1)	最小	250					ns
	t_{VCS}	V_{CC} 设置时间 (注 1)	最小	35					μ s
	t_{BUSY}	擦除 / 编程有效到 RY/BY# 延迟	最大	90					ns
	t_{SEA}	扇区擦除超时	最大	50					μ s

备注

- 未 100% 测试。
- 有关的详细信息，请参见章节 11.6。
- 适用于 1-32 字 / 1-64 字节编程。
- 有效写入缓冲器规格基于 32 字 / 64 字节写入缓冲器操作。
- 除非另有说明，否则 110 ns 速度选项的交流规格基于 $V_{IO} = V_{CC} = 2.7 V$ 进行测试。测试 110 ns 速度选项的交流规格时， $V_{IO} = 1.8 V$ 和 $V_{CC} = 3.0 V$ 。

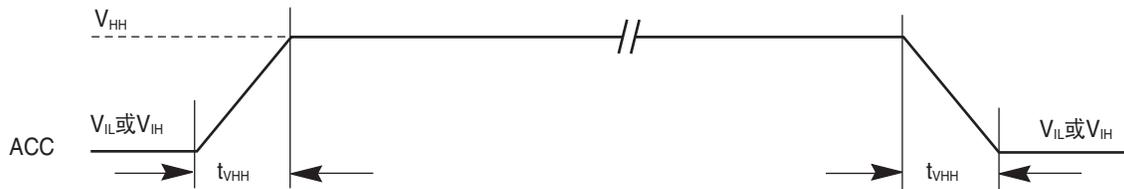
图 11.9 编程操作时序



备注

1. PA = 编程地址, PD = 编程数据, D_{OUT} 是编程地址的真实数据。
2. 插图所示是设备处于字模式。

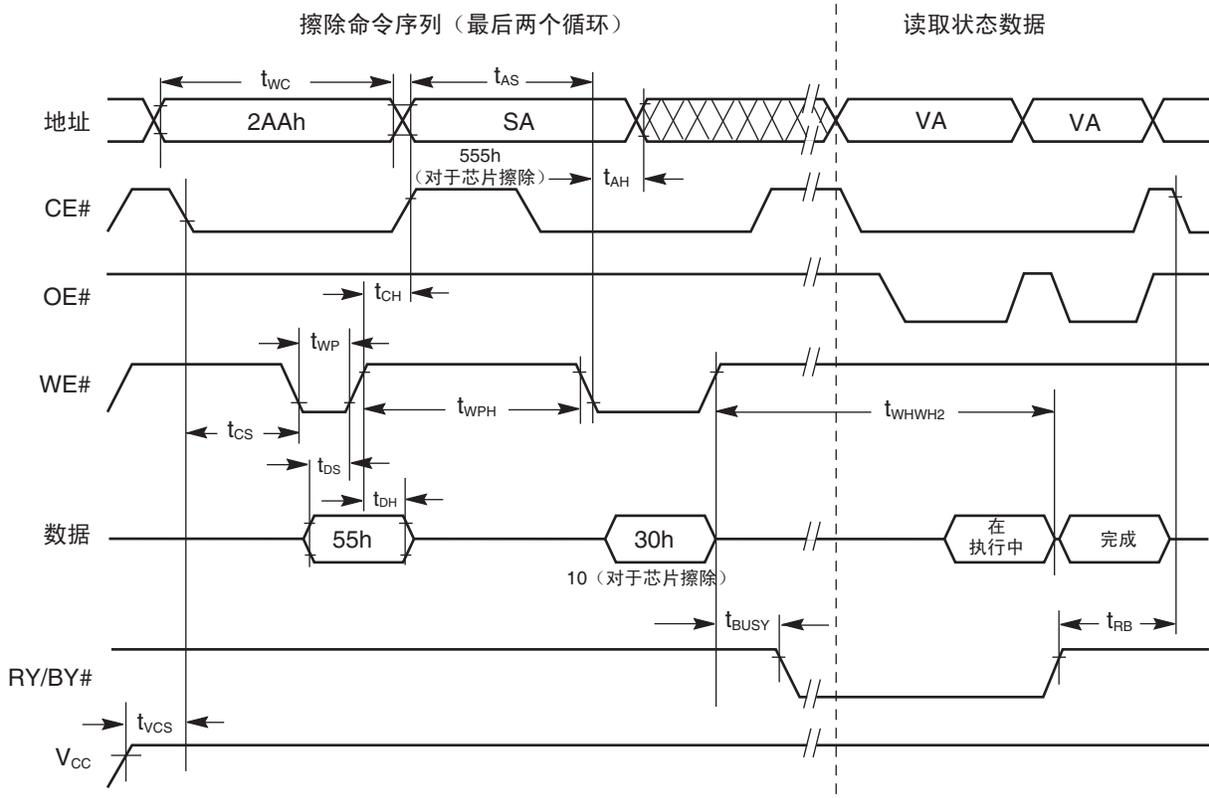
图 11.10 加速编程时序图



备注

1. 未 100% 测试。
2. CE#, OE# = V_{IL}
3. OE# = V_{IL}
4. 关于测试规格, 请参见图 11.3 和表 11.1。

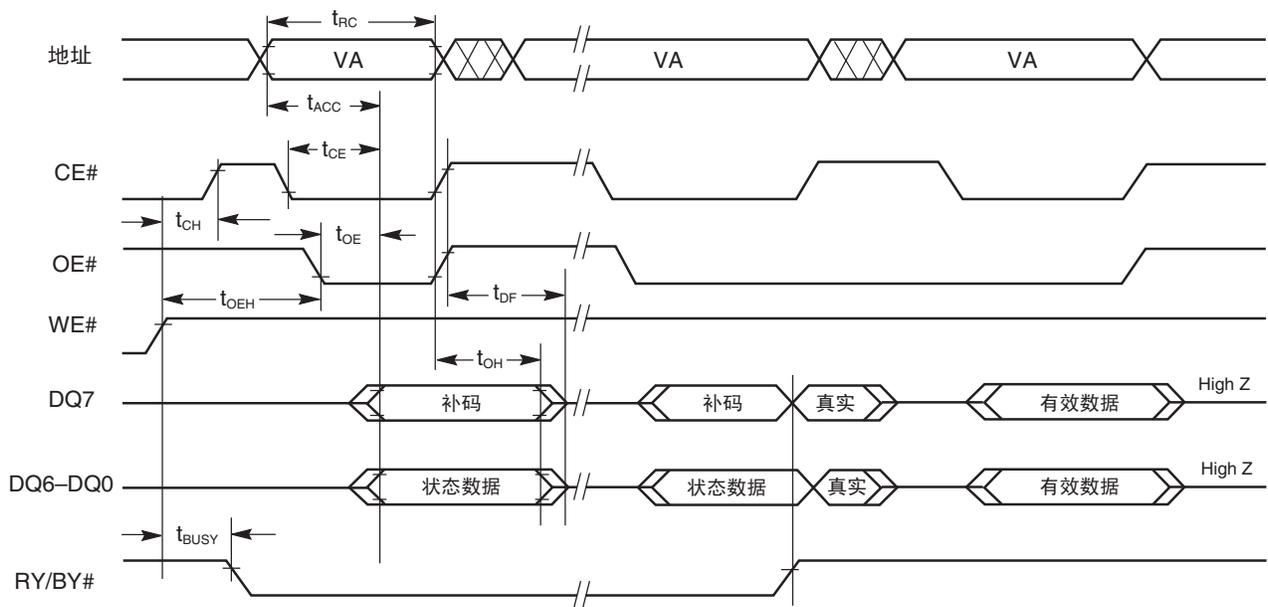
图 11.11 芯片 / 扇区擦除操作时序



备注

1. SA = 扇区地址 (对于扇区擦除), VA = 用于读取状态数据的有效地址 (参见第 36 页的写入操作状态)。
2. 这些波形适用于字模式

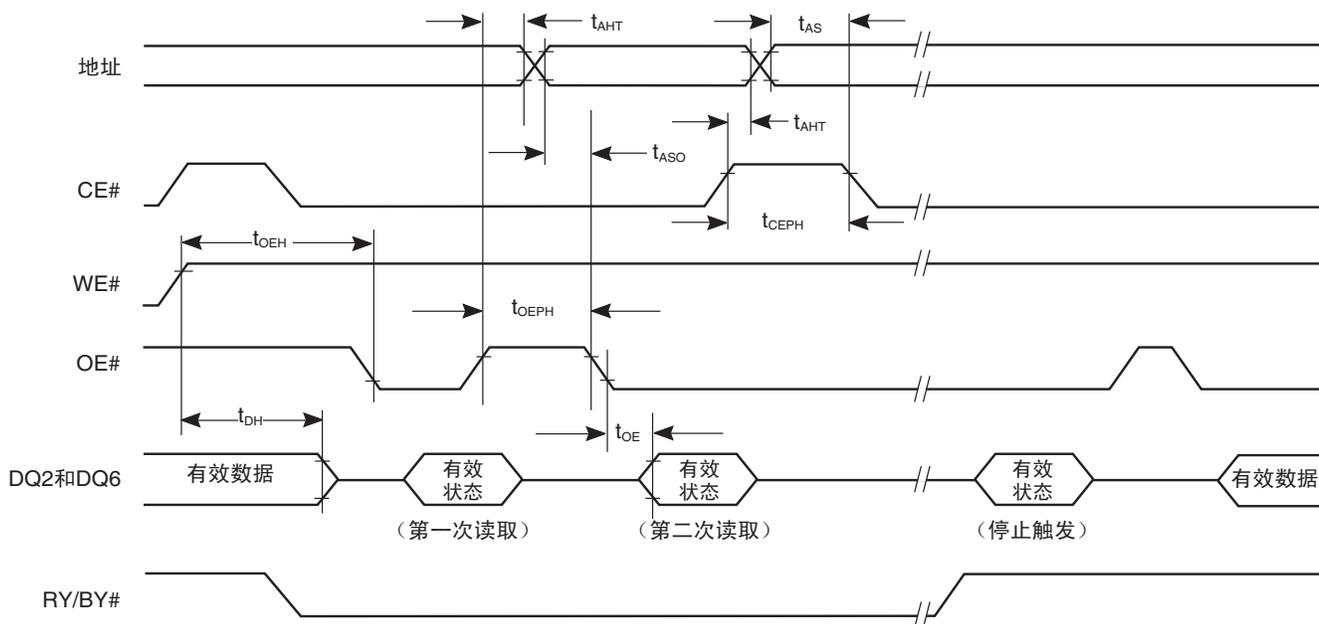
图 11.12 Data# 轮询时序 (在嵌入式算法期间)



备注

1. VA = 有效地址。图例显示命令序列之后的第一个状态循环, 最后一个状态读取循环, 以及阵列数据读取循环。
2. 当 V_{IO} = 1.65 到 2.7 V 时, 数据轮询的 t_{OE} 是 45 ns; 当 V_{IO} = 2.7 到 3.6 V 时, 是 35 ns
3. CE# 在状态位读取之间不需要变为高

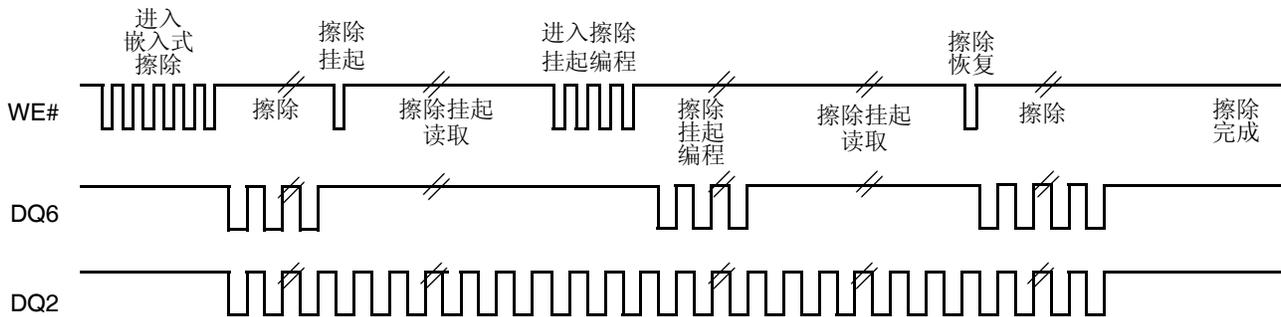
图 11.13 跳变位时序（在嵌入式算法期间）



注

A = 有效地址；DQ6 不需要。插图所示是命令序列之后的前两个状态循环、最后一个状态读取循环、以及阵列数据读取循环。CE# 在状态位读取之间不需要变为高

图 11.14 DQ2 对 DQ6



注

仅当读取的地址在擦除-挂起的扇区内时，DQ2 跳变。系统可以使用 OE# 或 CE# 跳变 DQ2 和 DQ6。

11.7.4 S29GL-P 备用 CE# 控制的擦除和编程 操作

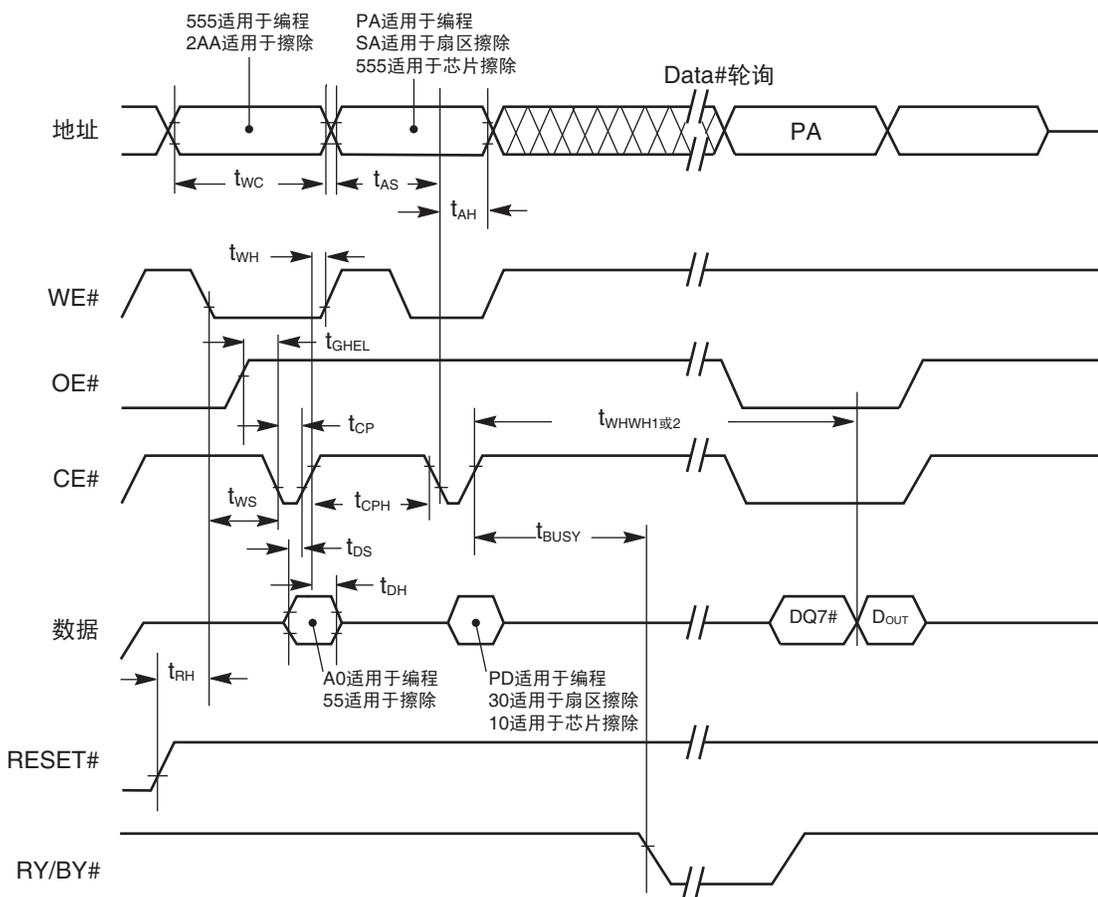
表 11.7 S29GL-P 备用 CE# 控制的擦除和编程 操作

参数		描述 (注)		速度选项					单位
JEDEC	Std.			90	100	110	120	130	
t_{AVAV}	t_{WC}	写入循环时间 (注 1)	最小	90	100	110	120	130	ns
t_{AVWL}	t_{AS}	地址设置时间	最小	0					ns
	t_{ASO}	跳变位轮询期间地址设置时间到 OE# 低	最小	15					ns
t_{ELAX}	t_{AH}	地址保持时间	最小	45					ns
	t_{AHT}	在跳变位轮询期间自 CE# 或 OE# 高的地址保持时间	最小	0					ns
t_{DVEH}	t_{DS}	数据设置时间	最小	30					ns
t_{EHDX}	t_{DH}	数据保持时间	最小	0					ns
	t_{CEPH}	在跳变位轮询期间 CE# 高	最小	20					ns
	t_{OEPH}	在跳变位轮询期间 OE# 高	最小	20					ns
t_{GHLEL}	t_{GHLEH}	写入之前的读取恢复时间 (OE# 高到 CE# 低)	最小	0					ns
t_{WLEL}	t_{WS}	WE# 设置时间	最小	0					ns
t_{EHWH}	t_{WH}	WE# 保持时间	最小	0					ns
t_{ELEH}	t_{CP}	CE# 脉冲宽度	最小	35					ns
t_{EHEL}	t_{CPH}	CE# 脉冲宽度高	最小	30					ns
t_{WHWH1}	t_{WHWH1}	写入缓冲器编程操作 (注 2、3)	典型	480					μ s
		有效写入缓冲器编程操作 (注 2、4)	每字 典型	15					μ s
		有效加速写入缓冲器编程操作 (注 2、4)	每字 典型	13.5					μ s
		编程操作 (注 2)	字 典型	60					μ s
		加速编程操作 (注 2)	字 典型	54					μ s
t_{WHWH2}	t_{WHWH2}	扇区擦除操作 (注 2)	典型	0.5					秒

备注

1. 未 100% 测试。
2. 有关的详细信息, 请参见第 54 页的直流特性。
3. 适用于 1-32 字 / 1-64 字节编程。
4. 有效写入缓冲器规格基于 32 字 / 64 字节写入缓冲器操作。
5. 除非另有说明, 否则测试交流规格时 $V_{IO} = 1.8 V$ 和 $V_{CC} = 3.0 V$ 。

图 11.15 备用 CE# 控制的写入（擦除 / 编程）操作 时序



备注

1. 图 11.15 指明编程或擦除操作的最后两个总线循环。
2. PA = 编程地址，SA = 扇区地址，PD = 编程数据。
3. DQ7# 是写入设备的数据的补码。DOUT 是写入设备的数据。
4. 波形适用于字模式。

11.7.5 擦除和编程性能

表 11.8 擦除和编程性能

参数		典型 (注 1)	最大 (注 2)	单位	备注
扇区擦除时间		0.5	3.5	秒	不包括擦除之前的 00h 编程 (注 4)
芯片擦除时间	S29GL128P	64	256	秒	
	S29GL256P	128	512		
	S29GL512P	256	1024		
	S29GL01GP	512	2048		
总计写入缓冲器时间 (注 3)		480		μs	不包括系统级 开销 (注 5)
总计加速写入缓冲器编程时间 (注 3)		432		μs	
芯片编程时间	S29GL128P	123		秒	
	S29GL256P	246			
	S29GL512P	492			
	S29GL01GP	984			

备注

- 典型的编程和擦除时间基于下列条件：25°C，3.6 V V_{CC} ，10,000 次循环，方格数据样本。
- 在最差的 -40°C 情况下， $V_{CC} = 3.0$ V，100,000 次循环。
- 有效写入缓冲器规格基于 32 字写入缓冲器操作。
- 在嵌入式擦除算法的预编程步骤，在擦除前所有位均编程至 00h。
- 系统级开销是为执行编程命令二或四总线循环序列而需要的时间。参见表 12.1–12.4。

11.7.6 TSOP 插针和 BGA 封装电容

表 11.9 封装电容

参数符号	参数描述	测试设置	典型	最大	单位
C_{IN}	输入电容	$V_{IN} = 0$	6	10	pF
C_{OUT}	输出电容	$V_{OUT} = 0$	10	12	pF
C_{IN2}	控制引脚电容	$V_{IN} = 0$	8	10	pF
WP#/ACC	分离的控制引脚	$V_{IN} = 0$	42	45	pF
RESET#	分离的控制引脚	$V_{IN} = 0$	25	28	pF
CE#	分离的控制引脚	$V_{IN} = 0$	22	25	pF

备注

- 抽样，未 100% 测试。
- 测试条件 $T_A = 25^\circ\text{C}$ ， $f = 100$ MHz。

12. 附录

此部分介绍与软件控制或闪存设备接口相关的信息。如需与软件相关的更多信息和帮助，请参见[章节 5](#)。如需最新信息，请浏览 Spansion 网站 www.spansion.com。

12.1 命令定义

向命令寄存器写入特定地址和数据命令或序列时，会启动设备操作。表 12.1–12.4 定义有效的寄存器命令序列。如果写入的地址和数据值不正确或写入的序列不正确，可能使设备进入未知状态。在此情况下，需要执行复位命令，使设备返回读取阵列数据。

表 12.1 S29GL-P 存储阵列命令定义, x16

命令 (注)		循环	总线循环 (注 1-5)											
			第一个		第二个		第三个		第四个		第五个		第六个	
			地址	数据	地址	数据	地址	数据	地址	数据	地址	数据	地址	数据
读取 (6)		1	RA	RD										
复位 (7)		1	XXX	F0										
自动选择 (8)	制造商 ID	4	555	AA	2AA	55	555	90	X00	01				
	设备 ID(8)	6	555	AA	2AA	55	555	90	X01	227E	X0E	(8)	X0F	(8)
	扇区保护检验 (10)	4	555	AA	2AA	55	555	90	[SA]X02	(10)				
	安全设备检验 (11)	4	555	AA	2AA	55	555	90	X03	(11)				
CFI 查询 (12)		1	55	98										
编程		4	555	AA	2AA	55	555	A0	PA	PD				
写入到缓冲器 (13)		6	555	AA	2AA	55	SA	25	SA	WC	WBL	PD	WBL	PD
编程缓冲器到闪存 (确认)		1	SA	29										
写入到缓冲器中止复位 (14)		3	555	AA	2AA	55	555	F0						
解锁省略	进入	3	555	AA	2AA	55	555	20						
	编程 (15)	2	XXX	A0	PA	PD								
	扇区擦除 (15)	2	XXX	80	SA	30								
	芯片擦除 (15)	2	XXX	80	XXX	10								
	复位 (16)	2	XXX	90	XXX	00								
芯片擦除		6	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10
扇区擦除		6	555	AA	2AA	55	555	80	555	AA	2AA	55	SA	30
擦除挂起 / 编程挂起 (17)		1	XXX	B0										
擦除恢复 / 编程恢复 (18)		1	XXX	30										
安全硅扇区进入		3	555	AA	2AA	55	555	88						
安全硅扇区退出 (19)		4	555	AA	2AA	55	555	90	XX	00				

缩略语

X = 不用考虑

RA = 要读取的存储地址。

RD = 在读取操作期间从位置 RA 读取的数据。

PA = 要编程的存储位置的地址。地址在 WE# 或 CE# 脉冲 (后发生者) 下降沿锁存。

备注

- 有关总线操作的说明, 请参见第 19 页的表 7.1。
- 所有值均为十六进制。
- 除非特别注明, 否则所有总线循环均为写入循环。
- 对于解锁和命令循环, 不用考虑数据位 DQ15-DQ8。
- 对于解锁和命令循环, 除非需要 SA 或 PA, 否则地址位 A_{MAX}:A16 不用考虑。(A_{MAX} 是最高地址引脚。)
- 读取阵列数据时不需要解锁或命令循环。
- 当设备处于自动选择模式时, 或者如果 DQ5 变为高 (当设备正在提供状态数据时), 需要通过复位命令返回读取阵列数据。
- 关于设备 ID 值和定义, 请参见第 22 页的表 7.2。
- 自动选择命令序列的第四、第五和第六个循环是读取循环。
- 对于不受保护的扇区, 数据是 00h; 对于受保护的扇区, 数据是 01h。有关的详细信息, 请参见第 21 页的自动选择。除了保护和不保护状态在此相反外, 这与 PPB 状态读取相同。

PD = 要在位置 PA 编程的数据。数据在 WE# 或 CE# 脉冲 (先发生者) 上升沿锁存。

SA = 要检验 (在自动选择模式下) 或擦除的扇区的地址。地址位 A_{MAX}-A16 唯一地选择任一扇区。

WBL = 写入缓冲器位置。地址必须与 PA 位于同一写入缓冲器页面。

WC = 字节数等于要加载的写入缓冲器位置数量减 1。

- 对于串行化、受保护的安全硅扇区区域, DQ7 的数据值是“1”; 对于非串行化、不受保护的区域, 是“0”。关于数据和定义, 请参见第 22 页的表 7.3。
- 当设备准备好读取阵列数据或者当设备处于自动选择模式时, 命令有效。
- 根据写入的字数, 总循环数可能是 6 到 37。
- 在被置于写入到缓冲器中止状态后, 命令序列使设备返回读取阵列。在解锁省略模式下从中止状态复位时, 需要完整的命令序列。
- 在解锁省略编程命令之前, 需要解锁省略命令。
- 当设备处于解锁省略模式时, 需要执行解锁省略复位命令, 才能返回读取阵列数据。
- 系统可以在非擦除扇区中读取和编程 / 编程挂起, 或者当处于擦除挂起模式时进入自动选择模式。擦除挂起命令仅在扇区擦除操作期间有效。
- 擦除恢复 / 编程恢复命令仅在擦除挂起 / 编程挂起模式期间有效。
- 退出命令使设备返回读取阵列。

表 12.2 S29GL-P 扇区保护命令定义, x16

命令 (注)		循环	总线循环 (注 1-5)											
			第一个 / 第七个		第二个		第三个		第四个		第五个		第六个	
			地址	数据	地址	数据	地址	数据	地址	数据	地址	数据	地址	数据
锁定 寄存器	命令集进入	3	555	AA	2AA	55	555	40						
	编程 (6)	2	XXX	A0	XXX	数据								
	读取 (6)	1	00	RD										
	命令集退出 (7、8)	2	XXX	90	XXX	00								
密码 保护	命令集进入	3	555	AA	2AA	55	555	60						
	密码编程 (9)	2	XXX	A0	PWA x	PWD x								
	密码读取 (10)	4	00	PWD0	01	PWD 1	02	PWD 2	03	PWD 3				
	密码解锁 (10)	7	00	25	00	03	00	PWD 0	01	PWD 1	02	PWD 2	03	PWD 3
			00	29										
命令集退出 (7、8)	2	XXX	90	XXX	00									
全局 非易失性	PPB 命令集进入	3	555	AA	2AA	55	555	C0						
	PPB 编程 (11、12)	2	XXX	A0	SA	00								
	所有 PPB 擦除 (13)	2	XXX	80	00	30								
	PPB 状态读取 (12)	1	SA	RD (0)										
	PPB 命令集退出 (7、8)	2	XXX	90	XXX	00								
全局 易失性冻结	PPB 锁定命令集进入	3	555	AA	2AA	55	555	50						
	PPB 锁定集 (12)	2	XXX	A0	XXX	00								
	PPB 锁定状态读取 (12)	1	XXX	RD (0)										
	PPB 锁定命令集退出 (7、8)	2	XXX	90	XXX	00								
易失性	DYB 命令集进入	3	555	AA	2AA	55	555	E0						
	DYB 设置 (11、12)	2	XXX	A0	SA	00								
	DYB 清除 (12)	2	XXX	A0	SA	01								
	DYB 状态读取 (12)	1	SA	RD (0)										
	DYB 命令集退出 (7、8)	2	XXX	90	XXX	00								

缩略语

X = 不用考虑

RD(0) = 读取数据。

SA = 扇区地址。地址位 A_{max}-A16 唯一地选择任一扇区。

备注

- 有关总线操作的说明, 请参见第 19 页的表 7.1。
- 所有值均为十六进制。
- 除非特别注明, 否则所有总线循环均为写入循环。
- 对于解锁和命令循环, 不用考虑数据位 DQ15-DQ8。
- 对于解锁和命令循环, 除非需要 SA 或 PA, 否则地址位 A_{MAX}:A16 不用考虑。(A_{MAX} 是最高地址引脚。)
- 所有锁定寄存器均为一次性可编程。编程状态 = “0”, 擦除状态 = “1”。持久保护模式锁定和密码保护模式锁定不能同时编程, 否则锁定寄存器位编程操作会中止, 设备返回读取模式。在默认情况下, 保留给未来使用的锁定寄存器位全部为“1”。在执行锁定寄存器位编程前, 锁定寄存器一直保持出厂值“FFFF”。

PWD = 密码

PWD_x = 密码 word0、word1、word2 和 word3。

数据 = 锁定寄存器内容; PD(0) = 安全硅扇区保护位, PD(1) = 持久保护模式锁定位, PD(2) = 密码保护模式锁定位。

- 退出命令使设备返回读取阵列。
- 如果写入了任何命令集进入命令, 则必须发出退出命令才能使设备返回读取模式。
- 对于 PWD_x, 每个“A0”命令只能编程密码的一个部分。
- 请注意, 密码的一部分可以以任何顺序输入或读取, 前提是输入或读取整个 64 位密码。
- 如果 ACC = V_{HH}, 当 ACC = V_{IH} 时扇区保护匹配。
- 受保护状态 = “00h”, 不受保护状态 = “01h”。
- 所有 PPB 擦除命令在擦除前嵌入所有 PPB 位编程。

表 12.3 S29GL-P 存储阵列命令定义, x8

命令 (注)	循环	总线循环 (注 1-5)											
		第一个		第二个		第三个		第四个		第五个		第六个	
		地址	数据	地址	数据	地址	数据	地址	数据	地址	数据	地址	数据
读取 (6)	1	RA	RD										
复位 (7)	1	XXX	F0										
自动选择 (8, 9)	制造商 ID	4	AAA	AA	555	55	AAA	90	X00	01			
	设备 ID(8)	6	AAA	AA	555	55	AAA	90	X02	XX7E	X1C	(8)	X1E (8)
	扇区保护检验 (10)	4	AAA	AA	555	55	AAA	90	[SA]X04	(10)			
	安全设备检验 (11)	4	AAA	AA	555	55	AAA	90	X06	(11)			
CFI 查询 (12)	1	AA	98										
编程	4	AAA	AA	555	55	AAA	A0	PA	PD				
写入到缓冲器 (13)	6	AAA	AA	555	55	SA	25	SA	WC	WBL	PD	WBL	PD
编程缓冲器到闪存 (确认)	1	SA	29										
写入到缓冲器中止复位 (14)	3	AAA	AA	555	55	555	F0						
解锁省略	进入	3	AAA	AA	555	55	AAA	20					
	编程 (15)	2	XXX	A0	PA	PD							
	扇区擦除 (15)	2	XXX	80	SA	30							
	芯片擦除 (15)	2	XXX	80	XXX	10							
	复位 (16)	2	XXX	90	XXX	00							
芯片擦除	6	AAA	AA	555	55	AAA	80	AAA	AA	555	55	AAA	10
扇区擦除	6	AAA	AA	555	55	AAA	80	AAA	AA	555	55	SA	30
擦除挂起 / 编程挂起 (17)	1	XXX	B0										
擦除恢复 / 编程恢复 (18)	1	XXX	30										
安全硅扇区进入	3	AAA	AA	555	55	AAA	88						
安全硅扇区退出 (19)	4	AAA	AA	555	55	AAA	90	XX	00				

缩略语

X = 不用考虑

RA = 要读取的存储地址。

RD = 在读取操作期间从位置 RA 读取的数据。

PA = 要编程的存储位置的地址。地址在 WE# 或 CE# 脉冲 (后发生者) 下降沿锁存。

PD = 要在位置 PA 编程的数据。数据在 WE# 或 CE# 脉冲 (先发生者) 上升沿锁存。

SA = 要检验 (在自动选择模式下) 或擦除的扇区的地址。地址位 A_{max}-A16 唯一地选择任一扇区。

WBL = 写入缓冲器位置。地址必须与 PA 位于同一写入缓冲器页面。

WC = 字计数等于要加载的写入缓冲器位置数量减 1。

备注

- 有关总线操作的说明, 请参见第 19 页的表 7.1。
- 所有值均为十六进制。
- 除非特别注明, 否则所有总线循环均为写入循环。
- 对于解锁和命令循环, 不用考虑数据位 DQ15-DQ8。
- 对于解锁和命令循环, 除非需要 SA 或 PA, 否则地址位 A_{MAX}:A16 不用考虑。(A_{MAX} 是最高地址引脚。)
- 读取阵列数据时不需要解锁或命令循环。
- 当设备处于自动选择模式时, 或者如果 DQ5 变为高 (当设备正在提供状态数据时), 需要通过复位命令返回读取阵列数据。
- 关于设备 ID 值和定义, 请参见第 22 页的表 7.2。
- 自动选择命令序列的第四、第五和第六个循环是读取循环。
- 对于不受保护的扇区, 数据是 00h; 对于受保护的扇区, 数据是 01h。有关的详细信息, 请参见第 21 页的自动选择。除了保护和不受保护状态在此相反外, 这与 PPB 状态读取相同。

- 对于串行化、受保护的安全硅扇区区域, DQ7 的数据值是 "1"; 对于非串行化、不受保护的区域, 是 "0"。关于数据和定义, 请参见第 22 页的表 7.3。

- 当设备准备好读取阵列数据或者当设备处于自动选择模式时, 命令有效。
- 根据写入的字数, 总循环数可能是 6 到 69。
- 在被置于写入到缓冲器中止状态后, 命令序列使设备返回读取阵列。在解锁省略模式下从中止状态复位时, 需要完整的命令序列。
- 在解锁省略编程命令之前, 需要解锁省略命令。
- 当设备处于解锁省略模式时, 需要执行解锁省略复位命令, 才能返回读取阵列数据。
- 系统可以在非擦除扇区中读取和编程 / 编程挂起, 或者当处于擦除挂起模式时进入自动选择模式。擦除挂起命令仅在扇区擦除操作期间有效。
- 擦除恢复 / 编程恢复命令仅在擦除挂起 / 编程挂起模式期间有效。
- 退出命令使设备返回读取阵列。

表 12.4 S29GL-P 扇区保护命令定义, x8

命令 (注)		循环	总线循环 (注 1-5)											
			第一个 / 第七个		第二个 / 第八个		第三个		第四个		第五个		第六个	
			地址	数据	地址	数据	地址	数据	地址	数据	地址	数据	地址	数据
锁定寄存器	命令集进入	3	AAA	AA	555	55	AAA	40						
	位编程 (6)	2	XXX	A0	XXX	数据								
	读取 (6)	1	00	RD										
	命令集退出 (7、8)	2	XXX	90	XXX	00								
密码保护	命令集进入	3	AAA	AA	555	55	AAA	60						
	密码编程 (9)	2	XXX	A0	PWA x	PWD x								
	密码读取 (10)	8	00	PWD0	01	PWD 1	02	PWD 2	03	PWD 3	04	PWD 4	05	PWD 5
			06	PWD 6	07	PWD 7								
	密码解锁 (10)	11	00	25	00	03	00	PWD 0	01	PWD 1	02	PWD 2	03	PWD 3
			04	PWD 4	05	PWD 5	06	PWD 6	07	PWD 7	00	29		
命令集退出 (7、8)	2	XXX	90	XXX	00									
全局 非易失性	PPB 命令集进入	3	AAA	AA	55	55	AAA	C0						
	PPB 编程 (11、12)	2	XXX	A0	SA	00								
	所有 PPB 擦除 (13)	2	XXX	80	00	30								
	PPB 状态读取 (12)	1	SA	RD (0)										
	PPB 命令集退出 (7、8)	2	XXX	90	XXX	00								
全局 易失性擦除	PPB 锁定命令集进入	3	AAA	AA	555	55	AAA	50						
	PPB 锁定位设置 (12)	2	XXX	A0	XXX	00								
	PPB 锁定状态读取 (12)	1	XXX	RD (0)										
	PPB 锁定命令集退出 (7、8)	2	XXX	90	XXX	00								
易失性	DYB 命令集进入	3	AAA	AA	555	55	AAA	E0						
	DYB 设置 (11、12)	2	XXX	A0	SA	00								
	DYB 清除 (12)	2	XXX	A0	SA	01								
	DYB 状态读取 (12)	1	SA	RD (0)										
	DYB 命令集退出 (7、8)	2	XXX	90	XXX	00								

缩略语

X = 不用考虑

RD(0) = 读取数据。

SA = 扇区地址。地址位 A_{max}-A16 唯一地选择任一扇区。

备注

- 有关总线操作的说明, 请参见第 19 页的表 7.1。
- 所有值均为十六进制。
- 除非特别注明, 否则所有总线循环均为写入循环。
- 对于解锁和命令循环, 不用考虑数据位 DQ15-DQ8。
- 对于解锁和命令循环, 除非需要 SA 或 PA, 否则地址位 A_{MAX}:A16 不用考虑。(A_{MAX} 是最高地址引脚。)
- 所有锁定寄存器位均为一次性可编程。编程状态 = “0”, 擦除状态 = “1”。持久保护模式锁定位和密码保护模式锁定位不能同时编程, 否则锁定寄存器位编程操作会中止, 设备返回读取模式。在默认情况下, 保留给未来使用的锁定寄存器位全部为 “1”。在执行锁定寄存器位编程前, 锁定寄存器一直保持出厂值 “FFFF”。

PWD = 密码

PWD_x = 密码 word0、word1、word2 和 word3。

数据 = 锁定寄存器内容; PD(0) = 安全硅扇区保护位, PD(1) = 持久保护模式锁定位, PD(2) = 密码保护模式锁定位。

- 退出命令使设备返回读取阵列。
- 如果写入了任何命令集进入命令, 则必须发出退出命令才能使设备返回读取模式。
- 对于 PWD_x, 每个 “A0” 命令只能编程密码的一个部分。
- 请注意, 密码的一部分可以以任何顺序输入或读取, 前提是输入或读取整个 64 位密码。
- 如果 ACC = V_{HH}, 当 ACC = V_{IH} 时扇区保护匹配。
- 受保护状态 = “00h”, 不受保护状态 = “01h”。
- 所有 PPB 擦除命令在擦除前嵌入所有 PPB 位编程。

12.2 通用闪存接口

通用闪存接口 (CFI) 规格概述设备和主机系统软件询问握手，它允许在整个系列的设备中使用特定厂商的软件算法。这样，软件支持就变得与设备和 JEDEC ID 无关，并且对特定闪存设备系列前后兼容。闪存厂商可以将现有接口标准化，以实现长期兼容性。

在设备准备好读取阵列数据的情况下，当系统向地址 55h 写入 CFI 查询命令 98h 时，设备进入 CFI 查询模式。系统可以在表 12.6–12.8 中给出的地址读取 CFI 信息。超出 CFI 地址范围之外的所有读取会返回非有效数据。对其他扇区可以读取，但不能写入。如要终止读取 CFI 数据，系统必须写入复位命令。

当设备处于自动选择模式时，系统也可以写入 CFI 查询命令。设备进入 CFI 查询模式，系统可以在表 12.6–12.8 中给出的地址读取 CFI 数据。系统必须写入复位命令，才能使设备返回读取阵列数据。

下面是使用 CFI 进入和退出功能的 C 源代码示例。有关 Spansion 闪存软件开发指南的一般信息，请参见 *Spansion Low Level Driver User's Guide*（在 www.spansion.com 提供）。

```
/* Example:CFI Entry command */
*( (UINT16 *)base_addr + 0x55 ) = 0x0098; /* write CFI entry command */

/* Example:CFI Exit command */
*( (UINT16 *)base_addr + 0x000 ) = 0x00F0; /* write cfi exit command */
```

有关的详细信息，请参见 CFI 规范（参见 JEDEC 出版物 JEP137-A 和 JESD68.01 以及 CFI 出版物 100）。如需这些文档的副本，请与您当地的销售机构联系。

表 12.5 CFI 查询识别字符串

地址 (x16)	地址 (x8)	数据	描述
10h 11h 12h	20h 22h 24h	0051h 0052h 0059h	查询唯一 ASCII 字符串“QRY”
13h 14h	26h 28h	0002h 0000h	主要 OEM 命令集
15h 16h	2Ah 2Ch	0040h 0000h	主扩展表地址
17h 18h	2Eh 30h	0000h 0000h	备用 OEM 命令集（00h = 不存在）
19h 1Ah	32h 34h	0000h 0000h	备用 OEM 扩展表地址（00h = 不存在）

表 12.6 系统接口字符串

地址 (x16)	地址 (x8)	数据	描述
1Bh	36h	0027h	V _{CC} 最小（写入 / 擦除）D7-D4: v, D3-D0: 100 mV
1Ch	38h	0036h	V _{CC} 最大（写入 / 擦除）D7-D4: v, D3-D0: 100 mV
1Dh	3Ah	0000h	V _{PP} 最小电压（00h = 不存在 V _{PP} 引脚）
1Eh	3Ch	0000h	V _{PP} 最大电压（00h = 不存在 V _{PP} 引脚）
1Fh	3Eh	0006h	每个单字节 / 字写入的典型超时 2 ^N μs
20h	40h	0009h	缓冲器写入的典型超时 2 ^N μs（00h = 不支持）
21h	42h	0009h	个别块擦除的典型超时 2 ^N ms
22h	44h	0013h = 1 Gb 0012h = 512 Mb 0011h = 256 Mb 0010h = 128 Mb	完全芯片擦除的典型超时 2 ^N ms（00h = 不支持）
23h	46h	0003h	典型的字节 / 字写入 2 ^N 次最大超时
24h	48h	0005h	典型的缓冲器写入 2 ^N 次最大超时
25h	4Ah	0003h	典型的个别块擦除 2 ^N 次最大超时
26h	4Ch	0002h	典型的完全芯片擦除 2 ^N 次最大超时（00h = 不支持）

表 12.7 设备几何定义

地址 (x16)	地址 (x8)	数据	描述
27h	4Eh	001Bh 001Ah 0019h 0018h	设备容量 = 2^N 字节 1B = 1 Gb, 1A = 512 Mb, 19 = 256 Mb, 18 = 128 Mb
28h 29h	50h 52h	0002h 0000h	闪存设备接口描述 (参见 CFI 出版物 100)
2Ah 2Bh	54h 56h	0006h 0000h	多字节写入的最多字节数 = 2^N (00h = 不支持)
2Ch	58h	0001h	设备内擦除块区域的数量 (01h = 统一设备, 02h = 引导设备)
2Dh 2Eh 2Fh 30h	5Ah 5Ch 5Eh 60h	00xxh 000xh 0000h 000xh	擦除块区域 1 信息 (参见 CFI 规范或 CFI 出版物 100) 00FFh, 0003h, 0000h, 0002h = 1 Gb 00FFh, 0001h, 0000h, 0002h = 512 Mb 00FFh, 0000h, 0000h, 0002h = 256 Mb 007Fh, 0000h, 0000h, 0002h = 128 Mb
31h 32h 33h 34h	62h 64h 66h 68h	0000h 0000h 0000h 0000h	擦除块区域 2 信息 (参见 CFI 出版物 100)
35h 36h 37h 38h	6Ah 6Ch 6Eh 70h	0000h 0000h 0000h 0000h	擦除块区域 3 信息 (参见 CFI 出版物 100)
39h 3Ah 3Bh 3Ch	72h 74h 76h 78h	0000h 0000h 0000h 0000h	擦除块区域 4 信息 (参见 CFI 出版物 100)

表 12.8 主要厂商特定的扩展查询

地址 (x16)	地址 (x8)	数据	描述
40h 41h 42h	80h 82h 84h	0050h 0052h 0049h	查询唯一 ASCII 字符串“PRI”
43h	86h	0031h	主版本号, ASCII
44h	88h	0033h	次版本号, ASCII
45h	8Ah	0014h	地址敏感解锁 (位 1-0) 0 = 需要, 1 = 不需要 工艺技术 (位 7-2) 0101b = 90 nm MirrorBit
46h	8Ch	0002h	擦除挂起 0 = 不支持, 1 = 只读, 2 = 读取和写入
47h	8Eh	0001h	扇区保护 0 = 不支持, X = 每组中扇区的数量
48h	90h	0000h	扇区临时取消保护 00 = 不支持, 01 = 支持
49h	92h	0008h	扇区保护 / 不保护方案 0008h = 高级扇区保护
4Ah	94h	0000h	并发操作 00 = 不支持, X = 扇区数量
4Bh	96h	0000h	突发模式类型 00 = 不支持, 01 = 支持
4Ch	98h	0002h	页面模式类型 00 = 不支持, 01 = 4 字页面, 02 = 8 字页面
4Dh	9Ah	00B5h	ACC (加速) 供应最小 00h = 不支持, D7-D4: V, D3-D0: 100 mV
4Eh	9Ch	00C5h	ACC (加速) 供应最大 00h = 不支持, D7-D4: V, D3-D0: 100 mV
4Fh	9Eh	00xxh	WP# 保护 04h = 统一扇区底部 WP# 保护, 05h = 统一扇区顶部 WP# 保护
50h	A0h	0001h	编程挂起 00h = 不支持, 01h = 支持

13. 关于 S29GL-R 65 nm MirrorBit 硬件的高级信息 复位 (RESET#) 和 加电 序列

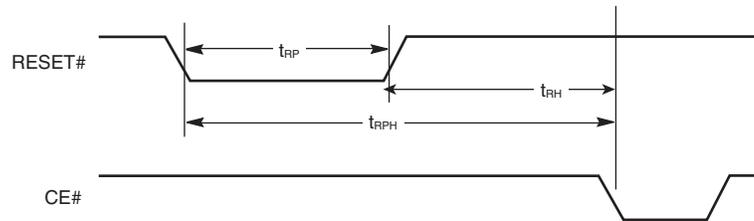
表 13.1 硬件复位 (RESET#)

参数	描述	限制	时间	单位
t_{RPH}	RESET# 低到 CE# 低	最小	35	μs
t_{RP}	RESET# 脉冲宽度	最小	200	ns
t_{RH}	RESET# (高) 和 CE# (低) 之间的时间	最小	200	ns

注

CE#、OE# 和 WE# 在复位时间内必须为逻辑高。

图 13.1 复位时序



注

t_{RP} 和 t_{RH} 之和必须等于或大于 t_{RPH} 。

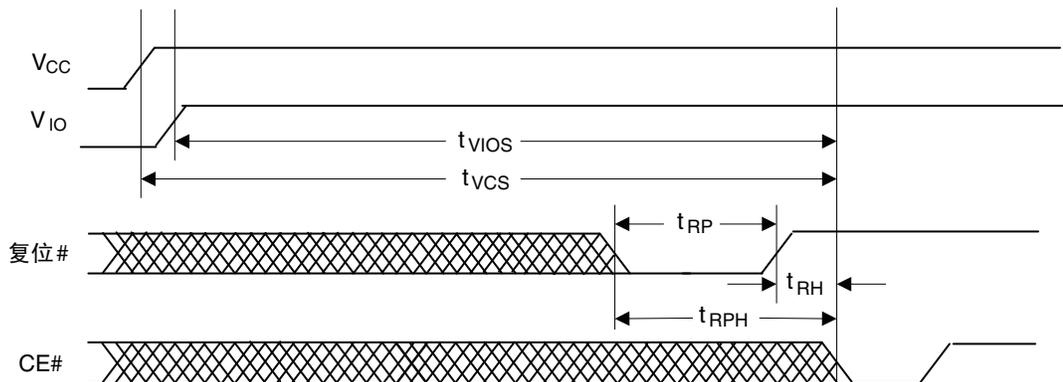
表 13.2 加电序列时序

参数	描述	限制	时间	单位
t_{VCS}	V_{CC} 第一次存取的设置时间	最小	300	μs
t_{VIO}	V_{IO} 第一次存取的设置时间	最小	300	μs
t_{RPH}	RESET# 低到 CE# 低	最小	35	μs
t_{RP}	RESET# 脉冲宽度	最小	200	ns
t_{RH}	RESET# (高) 和 CE# (低) 之间的时间	最小	200	ns

备注

- $V_{IO} < V_{CC} + 200 \text{ mV}$ 。
- V_{IO} 和 V_{CC} 斜率在加电期间必须同步。如果 RESET# 不稳定时间达到 300 μs ，可能发生下列情况：设备不允许任何读取和写入操作，有效的读取操作返回 FFh，需要硬件复位。
- 最大 V_{CC} 加电电流是 20 mA (RESET# = V_{IL})。

图 13.2 加电复位时序



注

t_{RP} 和 t_{RH} 之和必须等于或大于 t_{RPH} 。

14. 修订历史

章节	描述
修订 A0 (2004 年 10 月 29 日)	
	首次发布
修订 A1 (2005 年 10 月 20 日)	
全文	修改文档中的所有章节。
修订 A2 (2006 年 10 月 19 日)	
全文	修改文档中的所有章节。按照新模板重新调整文档格式。更改 S29GL01GP 的速度选项。
修订 A3 (2006 年 11 月 21 日)	
交流特性	<i>擦除和编程操作表</i> : 将 t_{BUSY} 改为最大规格。
修订 A4 (2006 年 12 月 18 日)	
全文	更改 128 Mb、256 Mb 和 512 Mb 设备的 t_{ACC} 、 t_{CE} 规格。增加 90 和 100 ns 速度选项。
写入缓冲器编程, 扇区擦除	<i>写入缓冲器编程操作</i> 、 <i>扇区擦除操作图</i> : 从流程图中删除“等待 4 ms”框。
密码保护方法	<i>锁定寄存器编程算法图</i> : 从流程图中删除“等待 4 ms”框。
只读操作表	修改 t_{RC} 、 t_{ACC} 、 t_{CE} 、 t_{OE} 规格。
编程和擦除操作表	更改 t_{DS} 规格, 删除关于写入循环时间的注释。
TSOP 插针和 BGA 电容表	更改表中所有的规格。
修订 A5 (2007 年 5 月 18 日)	
全文	将数据表状态改为“初级”。 删除对外部 WP# 拉升要求的引用。
性能特性	<i>最大读取访问时间表</i> : 添加注释。
硬件复位	从章节中删除注释。
交流特性	<i>复位时序图</i> : 删除注释。
命令定义表	<i>S29GL-P 扇区保护命令定义表</i> : 将“全局非易失性冻结”改为“全局易失性冻结”。
直流特性	<i>CMOS 兼容表</i> : 更改 5 MHz 和 MHz 测试条件的 I_{CC1} 最大电流。
页面读取时序图	更正上部波形的地址范围。
修订 A6 (2007 年 10 月 23 日)	
性能特性	更改 S29GL512P 的速度选项
订购信息	更正样品 OPN 有效组合; 更改 S29GL512P 的速度选项
64 球加固 BGA	澄清球“D1”连接
56 针 TSOP	澄清插针“30”连接
自动选择	增加推荐声明
加速编程	增加推荐声明
持久保护位	从标题和流程图中删除“擦除”
安全硅扇区	章节“工厂锁定的安全硅扇区”和“用户可锁定的安全硅扇区”: 澄清运输选项
加电序列时序	将 t_{RH} 从“最大”改为“最小”值
关于 S29GL-R 65 nm MirrorBit 硬件复位 (RESET#) 和加电序列的高级信息	新增章节
全文	更正超级链接无效的交叉引用。
修订 A7 (2007 年 11 月 8 日)	
关于 S29GL-R 65 nm MirrorBit 硬件复位 (RESET#) 和加电序列的高级信息	更改时序规格和波形

章节	描述
修订 A8 (2007 年 11 月 28 日)	
订购信息	新商用运行温度选项
运行范围	新运行温度范围
修订 A9 (2008 年 2 月 15 日)	
电气规格	修改测试条件
擦除和编程性能	芯片编程时间：删除注释
扇区保护命令定义，x16 表	更正锁定寄存器“读取”地址
关于 S29GL-R 65 nm MirrorBit 硬件复位 (RESET#) 和加电序列的高级信息	加电序列时序表：修改注 2 - 将时序从 500 μ s 缩短至 300 μ s
修订 A10 (2008 年 3 月 19 日)	
全文	将文档状态改为“量产”。
直流特性	更改输入负载电流的最大值 (I_{LI})
扇区保护命令定义 (x16 和 x8 表)	将锁定寄存器读取命令从“DATA”改为“RD”
修订 A11 (2008 年 6 月 11 日)	
订购信息	修订商用温度范围
图：写入操作状态流程图	更新流程图
修订 A12 (2009 年 11 月 20 日)	
表输入 / 输出描述	删除 RFU 描述
图 64 球加固球栅阵列	将所有 RFU 针改为 NC 针
图 56 针标准 TSOP (顶视图)	将所有 RFU 针改为 NC 针
表自动选择退出	将循环描述改为 <i>自动选择退出命令</i>
表芯片擦除	将最后一个 C 源代码命令的地址从 0x000h 改为 0x555h
擦除挂起 / 擦除恢复	将第一段第二句中关于扇区地址的描述改为对于擦除挂起“不用考虑” 将第六段第二句中关于扇区地址的描述改为对于擦除挂起“不用考虑”
表 编程挂起 编程恢复 解锁省略进入 解锁省略编程 解锁省略复位	向表中加入字节地址
解锁省略	在第三段第一句中，增加解锁省略扇区擦除和解锁省略芯片擦除作为有效命令 将段落第三句中关于退出命令扇区地址的描述改为“不用考虑”。
写入命令 / 命令序列	将第四句中列出的表改为表 6.1-6.4
WP#/ACC 方法	将“注”部分中列出的表改为 11.2。
安全硅扇区进入 / 退出命令序列	在表 10.3 下方添加程序源代码
表安全硅扇区退出	将退出循环的字节和字地址改为“XXXh”
图测试设置	将测试设置改为只显示 CL 负载
表测试规格	删除输出负载测试条件
表 S29GL-P 擦除和编程操作	删除 t_{GHWL}
表 S29GL-P 备用 CE# 控制的擦除和编程操作	将 t_{GHEL} 的描述改为 (OE# 高到 CE# 低) 将注 2 改为“直流特性”
TSOP 插针和 BGA 封装电容	更改 RESET# 值。
表 S29GL-P 存储阵列命令定义，x16	将设备 ID 的循环数改为 6 将写入缓冲器的循环数改为 6 增加关于写入缓冲器命令中循环次数的注释
表 S29GL-P 存储阵列命令定义，x8	将设备 ID 的循环数改为 6 将写入缓冲器的循环数改为 6 增加关于写入缓冲器命令中循环次数的注释

章节	描述
表系统接口字符串	将地址 20h (x16) 的值改为 0009h, 将描述改为 “典型的缓冲器写入超时 2 ⁿ μs” 为地址 22h (x16) 增加 128 Mb-512 Mb 密度值
表设备几何定义	对于地址 31h (x16), 更正 x8 地址
修订 A13 (2010 年 11 月 17 日)	
性能特性	更新 S29GL512P 的存取时间选项
订购信息	更新 S29GL512P 的速度选项
读取操作时序图	增加注释

书末出版说明

本文档介绍的产品，其设计、开发和制造均基于一般用途，包括但不限于普通工业使用、普通办公使用、个人使用及家庭使用，不应用于：(1) 存在严重风险或危险，除非能够保证极高的安全性，否则可能对公众造成严重影响，甚至可能直接造成死亡、人员伤害、物品损坏或其他损失的用途（如核设施的核反应控制、飞机飞行控制、空中交通控制、公共交通控制、医学生命支持系统、武器系统的导弹发射控制），或者(2) 不允许出现故障的用途（如潜艇中继器和人造卫星）。请注意，对于您和/或任何第三方由于将产品用于上述用途而造成的任何索赔和损失，Spansion 不承担任何责任。任何半导体设备都可能发生故障。您必须在自己的设施和装置中加入安全设计措施，如冗余、防火、防止电流过载及其他异常运行情形等，以防由于此类故障而造成伤害、损坏或损失。如果根据日本 Foreign Exchange and Foreign Trade Law、美国 US Export Administration Regulations 或其他国家（地区）的适用法律的规定，本文档中介绍的任何产品是在出口方面受到特别限制的商品或技术，则这些产品的出口必须预先得到相关政府的许可。

商标和声明

本文档的内容如有变更，恕不另行通知。本文档可能包含 Spansion 正在开发的 Spansion 产品的相关信息。Spansion 保留变更任何产品或停止其相关工作的权利，恕不另行通知。本文档中的信息“按原样”提供，对于其精确性、完整性、可操作性、对特定用途的适用性、适销性、不侵犯第三方权利等不提供任何担保或保证，也不提供任何明确的、隐含的或法定的其他担保。对于因使用本文档中的信息而造成的任何形式的任何损失，Spansion 不承担任何责任。

版权所有 © 2004–2010 Spansion Inc. 保留所有权利。Spansion®、Spansion 标识、MirrorBit®、MirrorBit® Eclipse™、ORNAND™、EcoRAM™ 以及它们的组合，是 Spansion LLC 在美国和其他国家（地区）的商标和注册商标。使用的其他名称只是一般性参考信息，可能是其各自所有者的商标。