



SST 的 25 系列串行闪存采用四线、兼容 SPI 的接口，从而实现占用较少电路板空间的低引脚数封装，并最终降低总系统成本。SST25VF016B 器件是增强型器件，提高了工作频率并降低了功耗。SST25VF016B SPI 串行闪存采用 SST 专有的高性能 CMOS SuperFlash® 技术制造。与其他方法相比，分离栅极单元设计 (split-gate cell design) 和厚氧化层隧穿注入器 (thick-oxide tunneling injector) 可实现更高的可靠性和可制造性。

特性

- 单电压读写操作
 - 2.7-3.6V
- 串行接口架构
 - 兼容 SPI: 模式 0 和模式 3
- 高速时钟频率
 - 高达 80 MHz
- 超高可靠性
 - 可擦写次数: 100,000 次 (典型值)
 - 数据保存时间大于 100 年
- 低功耗:
 - 读操作工作电流: 10 mA (典型值)
 - 待机电流: 5 μ A (典型值)
- 灵活的擦除功能
 - 均一 4 KB 扇区
 - 均一 32 KB 覆盖块
 - 均一 64 KB 覆盖块
- 快速擦除和字节编程:
 - 全片擦除时间: 35 ms (典型值)
 - 扇区 / 块擦除时间: 18 ms (典型值)
 - 字节编程时间: 7 μ s (典型值)
- 自动地址递增 (Auto Address Increment, AAI) 编程
 - 减少字节编程操作期间的总芯片编程时间
- 写操作结束检测
 - 软件轮询状态寄存器中的 BUSY 位
 - AAI 模式下 SO 引脚上的忙状态读出
- 保持引脚 (HOLD#)
 - 在不取消选择器件的情况下暂停存储器的串行序列
- 写保护 (WP#)
 - 使能 / 禁止状态寄存器的锁定功能
- 软件写保护
 - 通过状态寄存器中的块保护位实现写保护
- 温度范围
 - 商业级: 0°C 至 +70°C
 - 工业级: -40°C 至 +85°C
- 可用封装
 - 8 引脚 SOIC (200 mil)
 - 8 触点 WSON (6 mm x 5 mm)
- 所有器件均符合 RoHS 标准



产品描述

SST 的 25 系列串行闪存采用四线、兼容 SPI 的接口，从而实现占用较少电路板空间的低引脚数封装，并最终降低总系统成本。SST25VF016B 器件是增强型器件，与原有的 SST25VFxxxA 器件相比，提高了工作频率并降低了功耗。SST25VF016B SPI 串行闪存采用 SST 专有的高性能 CMOS SuperFlash® 技术制造。与其他方法相比，分离栅极单元设计（split-gate cell design）和厚氧化层隧穿注入器（thick-oxide tunneling injector）可实现更高的可靠性和可制造性。

SST25VF016B 器件可以显著提高性能和可靠性，同时降低功耗。对于 SST25VF016B，器件使用 2.7-3.6V 单电源进行写操作（编程或擦除）。消耗的总能量是应用中施加电压、电流和时间的函数。对于任何给定的电压范围，SuperFlash® 技术的编程电流更低、擦除时间更短；因此，在任何擦除或编程操作期间消耗的总能量低于其他闪存技术。

SST25VF016B 器件提供 8 引脚 SOIC（200 mil）和 8 触点 WSON（6 mm x 5 mm）封装。有关引脚分配，请参见图 2。



框图

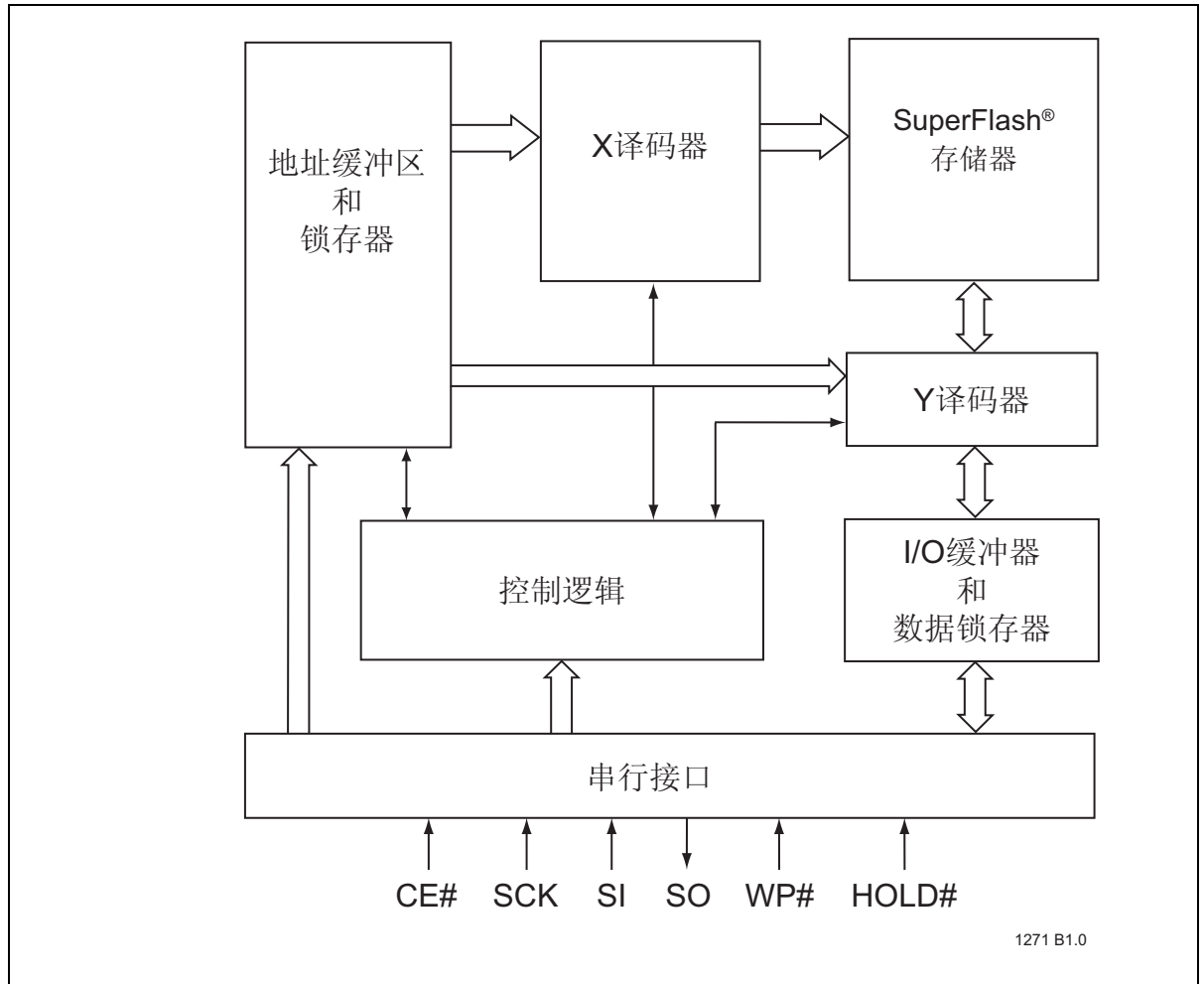


图 1: 功能框图



引脚说明

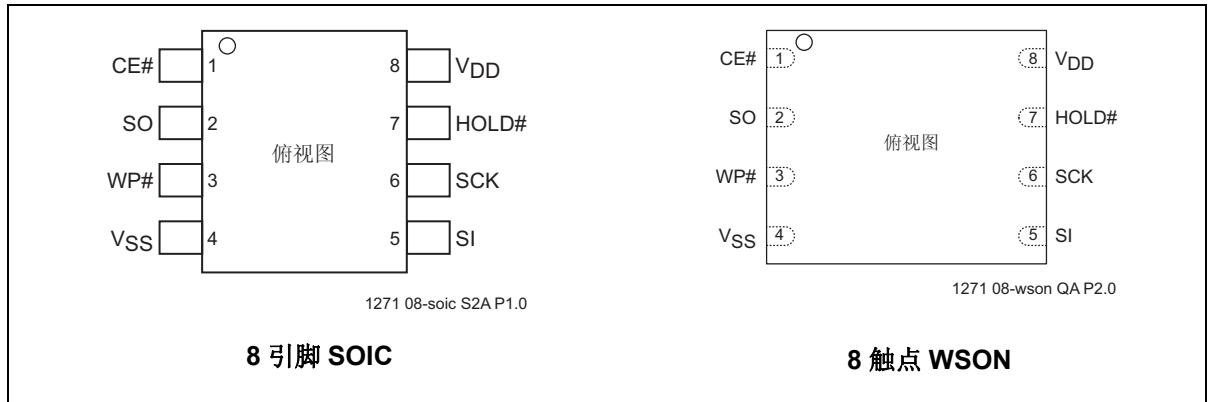


图 2: 引脚分配

表 1: 引脚说明

符号	引脚名称	功能
SCK	串行时钟	用于提供串行接口的时序。命令、地址或输入数据在时钟输入的上升沿进行锁存，而输出数据在时钟输入的下降沿移出。
SI	串行数据输入	用于以串行方式将命令、地址或数据传送到器件中。输入在串行时钟的上升沿进行锁存。
SO	串行数据输出	用于以串行方式将数据传送到器件。数据在串行时钟的下降沿移出。当重新配置为 RY/BY# 引脚后，在 AAI 编程期间输出闪存忙状态。有关详细信息，请参见第 12 页上的“硬件写操作结束检测”。
CE#	芯片使能	CE# 上高电平到低电平的转换可使能器件。在任何命令序列的持续时间内， CE# 都必须保持低电平。
WP#	写保护	写保护 (WP#) 引脚用于使能 / 禁止状态寄存器中的 BPL 位。
HOLD#	保持	用于在不复位器件的情况下暂时停止与 SPI 闪存存储器的串行通信。
VDD	电源	用于提供电源电压：对于 SST25VF016B ，为 2.7-3.6V。
VSS	地	

T1.0 25044



存储器构成

SST25VF016B SuperFlash 存储器阵列由均一 4 KB 可擦除扇区以及 32 KB 覆盖块和 64 KB 覆盖可擦除块组成。

器件操作

SST25VF016B 通过兼容 SPI（串行外设接口）总线的协议进行访问。SPI 总线包含四条控制线：芯片使能（CE#）用于选择器件，数据通过串行数据输入（SI）、串行数据输出（SO）和串行时钟（SCK）进行访问。

SST25VF016B 支持 SPI 总线操作的模式 0（0,0）和模式 3（1,1）。如图 3 所示，两种模式之间的差异在于当总线主器件处于待机模式并且没有数据传送时的 SCK 信号状态。SCK 信号在模式 0 时为低电平，在模式 3 时为高电平。对于这两种模式，串行数据输入（SI）在 SCK 时钟信号的上升沿被采样，串行数据输出（SO）在 SCK 时钟信号的下降沿之后被驱动。

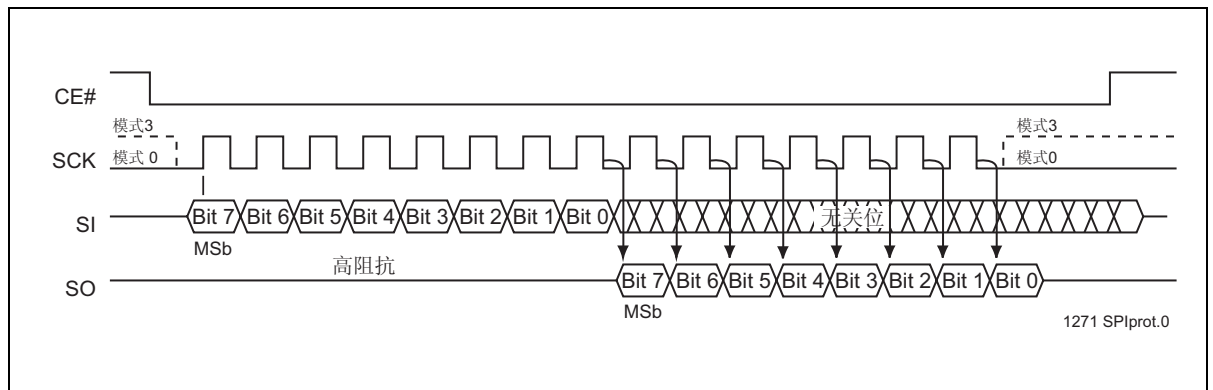


图 3: SPI 协议



保持操作

HOLD# 引脚用于暂停正在操作 SPI 闪存存储器的串行序列，但不会复位时钟序列。要激活 HOLD# 模式，CE# 必须处于有效低电平状态。当 SCK 有效低电平状态与 HOLD# 信号的下降沿同时发生时，HOLD# 模式开始。当 HOLD# 信号的上升沿与 SCK 有效低电平状态同时发生时，保持模式结束。

如果 HOLD# 信号的下降沿未与 SCK 有效低电平状态同时发生，则当 SCK 再次达到有效低电平状态时器件进入保持模式。同样，如果 HOLD# 信号的上升沿未与 SCK 有效低电平状态同时发生，则当 SCK 再次达到有效低电平状态时器件退出保持模式。有关保持条件波形，请参见图 4。

器件进入保持模式之后，SO 将处于高阻抗状态，而 SI 和 SCK 可以为 V_{IL} 或 V_{IH} 。

如果在保持条件期间将 CE# 驱动为有效高电平，则将复位器件的内部逻辑。只要 HOLD# 信号为低电平，存储器就会保持在保持条件下。要继续与器件进行通信，必须将 HOLD# 驱动为有效高电平，并将 CE# 驱动为有效低电平。有关保持时序，请参见图 24。

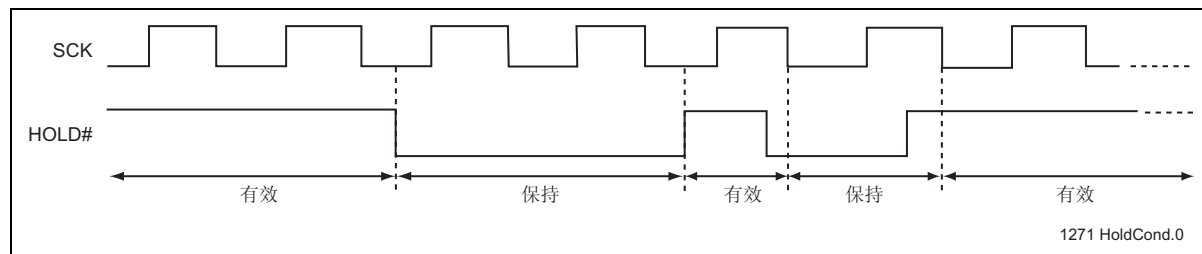


图 4： 保持条件波形

写保护

SST25VF016B 提供了软件写保护。写保护引脚 (WP#) 用于使能或禁止状态寄存器的锁定功能。状态寄存器中的块保护位 (BP3、BP2、BP1、BP0 和 BPL) 提供对存储器阵列和状态寄存器的写保护。有关块保护的说明，请参见表 4。

写保护引脚 (WP#)

写保护 (WP#) 引脚用于使能状态寄存器的 BPL 位 (bit 7) 的锁定功能。当 WP# 驱动为低电平时，是否执行写状态寄存器 (WRSR) 指令由 BPL 位的值 (见表 2) 决定。当 WP# 为高电平时，BPL 位的锁定功能被禁止。

表 2： 执行写状态寄存器 (WRSR) 指令的条件

WP#	BPL	执行 WRSR 指令
L	1	不允许
L	0	允许
H	X	允许

T2.0 25044



状态寄存器

软件状态寄存器提供有关闪存存储器阵列是否可进行任何读或写操作的状态、器件是否写使能以及存储器写保护的状态。在内部擦除或编程操作期间，可对状态寄存器只进行读操作，以确定正在进行的操作的完成状态。表 3 列出了软件状态寄存器中每个位的功能。

表 3: 软件状态寄存器

Bit	名称	功能	上电时的默认值	读 / 写
0	BUSY	1 = 内部写操作正在进行 0 = 没有内部写操作正在进行	0	R
1	WEL	1 = 器件处于存储器写使能状态 0 = 器件未处于存储器写使能状态	0	R
2	BP0	指示当前的块写保护级别（见表 4）	1	R/W
3	BP1	指示当前的块写保护级别（见表 4）	1	R/W
4	BP2	指示当前的块写保护级别（见表 4）	1	R/W
5	BP3	指示当前的块写保护级别（见表 4）	0	R/W
6	AAI	自动地址递增编程状态 1 = AAI 编程模式 0 = 字节编程模式	0	R
7	BPL	1 = BP3、BP2、BP1 和 BP0 为只读位 0 = BP3、BP2、BP1 和 BP0 可读 / 写	0	R/W

T3.0 25044

BUSY

BUSY 位确定是否有内部擦除或编程操作正在进行中。BUSY 位为“1”表示器件正忙于操作。BUSY 位为“0”表示器件准备好进行下一个有效操作。

写使能锁存器（WEL）

写使能锁存器（Write-Enable Latch，WEL）位表示内部寄存器的写使能锁存器状态。如果写使能锁存器位置“1”，则表示器件写使能。如果该位置“0”（复位），则表示器件未写使能，并且不接受任何存储器写（编程 / 擦除）命令。在下列条件下，写使能锁存器位自动复位：

- 上电
- 写禁止（WRDI）指令完成
- 字节编程指令完成
- 自动地址递增（AAI）编程完成或达到其最高的不受保护的存储器地址
- 扇区擦除指令完成
- 块擦除指令完成
- 全片擦除指令完成
- 写状态寄存器指令



自动地址递增 (AAI)

自动地址递增编程状态位提供有关器件处于自动地址递增 (Auto Address Increment, AAI) 编程模式还是字节编程模式的状态。上电时默认为字节编程模式。

块保护 (BP3、BP2、BP1 和 BP0)

块保护 (BP3、BP2、BP1 和 BP0) 位定义了要通过软件保护方式防止任何存储器写 (编程或擦除) 操作的存储器区域的大小, 如表 4 中定义。只要 WP# 为高电平或块保护锁定 (Block-Protect-Lock, BPL) 位为 0, 就可以使用写状态寄存器 (WRSR) 指令编程 BP3、BP2、BP1 和 BP0 位。仅当块保护位全部为 0 时, 才能执行全片擦除。上电后, BP3、BP2、BP1 和 BP0 置为 1。

块保护锁定 (BPL)

WP# 引脚驱动为低电平 (V_{IL}) 时, 将使能块保护锁定 (BPL) 位。当 BPL 置 1 时, 将阻止对 BPL、BP3、BP2、BP1 和 BP0 位的任何进一步修改。当 WP# 引脚驱动为高电平 (V_{IH}) 时, BPL 位没有任何作用, 其值为“无关”。上电后, BPL 位复位为 0。

表 4: SST25VF016B¹ 的软件状态寄存器块保护

保护级别	状态寄存器位 ²				受保护的存储器地址
	BP3	BP2	BP1	BP0	16 Mb
无	X	0	0	0	无
前 1/32	X	0	0	1	1F0000H-1FFFFFFH
前 1/16	X	0	1	0	1E0000H-1FFFFFFH
前 1/8	X	0	1	1	1C0000H-1FFFFFFH
前 1/4	X	1	0	0	180000H-1FFFFFFH
前 1/2	X	1	0	1	100000H-1FFFFFFH
所有块	X	1	1	0	000000H-1FFFFFFH
所有块	X	1	1	1	000000H-1FFFFFFH

T4.0 25044

1. X = 无关 (保留), 默认值为“0”。
2. 上电时, BP2、BP1 和 BP0 的默认值为“111” (所有块均受到保护)。



指令

指令用于读、写（擦除和编程）和配置 SST25VF016B。指令总线周期是 8 个表示命令（操作码）、数据和地址的位。在执行任何字节编程、自动地址递增（AAI）编程、扇区擦除、块擦除、写状态寄存器或全片擦除指令之前，必须先执行写使能（WREN）指令。表 5 提供了完整的指令列表。所有指令在 CE# 从高电平转换到低电平时同步。在 SCK 的上升沿从最高有效位开始接受输入。在输入指令之前，CE# 必须驱动为低电平，而在输入指令的最后一位后，CE# 必须驱动为高电平（读、读 ID 和读状态寄存器指令除外）。在接收到指令总线周期的最后一位之前，CE# 上任何低电平到高电平的转换都将终止正在进行的指令，并将器件恢复为待机模式。指令命令（操作码）、地址和数据都先从最高有效位（Most Significant Bit, MSb）输入。

表 5: 器件操作指令

指令	说明	操作码周期 ¹	地址周期 ²	空周期	数据周期	最大频率
读	以 25 MHz 的频率 读存储器	0000 0011b (03H)	3	0	1 至 ∞	25 MHz
高速读	以 80 MHz 的频率 读存储器	0000 1011b (0BH)	3	1	1 至 ∞	80 MHz
4 KB 扇区擦除 ³	擦除 4 KB 的存储器 阵列	0010 0000b (20H)	3	0	0	80 MHz
32 KB 块擦除 ⁴	擦除 32 KB 块的存储器 阵列	0101 0010b (52H)	3	0	0	80 MHz
64 KB 块擦除 ⁵	擦除 64 KB 块的存储器 阵列	1101 1000b (D8H)	3	0	0	80 MHz
全片擦除	擦除全部存储器阵列	0110 0000b (60H) 或 1100 0111b (C7H)	0	0	0	80 MHz
字节编程	编程一个数据字节	0000 0010b (02H)	3	0	1	80 MHz
AAI 字编程 ⁶	自动地址递增编程	1010 1101b (ADH)	3	0	2 至 ∞	80 MHz
RDSR ⁷	读取状态寄存器	0000 0101b (05H)	0	0	1 至 ∞	80 MHz
EWSR	使能写状态寄存器	0101b 0000b (50H)	0	0	0	80 MHz
WRSR	写状态寄存器	0000 0001b (01H)	0	0	1	80 MHz
WREN	写使能	0000 0110b (06H)	0	0	0	80 MHz
WRDI	写禁止	0000 0100b (04H)	0	0	0	80 MHz
RDID ⁸	读 ID	1001 0000b (90H) 或 1010 1011b (ABH)	3	0	1 至 ∞	80 MHz
JEDEC-ID	JEDEC ID 读	1001 1111b (9FH)	0	0	3 至 ∞	80 MHz
EBSY	在 AAI 编程期间使能 SO 以输出 RY/BY# 状态	0111 0000b (70H)	0	0	0	80 MHz
DBSY	在 AAI 编程期间禁止 SO 为 RY/BY# 状态	1000 0000b (80H)	0	0	0	80 MHz

T5.0 25044

1. 一个总线周期等于八个时钟周期。



2. 每个容量的最高有效位以上的地址位可以是 V_{IL} 或 V_{IH} 。
3. 4 KB 扇区擦除地址：使用 $A_{MS}-A_{12}$ ，其余地址没有影响，但必须设置为 V_{IL} 或 V_{IH} 。
4. 32 KB 块擦除地址：使用 $A_{MS}-A_{15}$ ，其余地址没有影响，但必须设置为 V_{IL} 或 V_{IH} 。
5. 64 KB 块擦除地址：使用 $A_{MS}-A_{16}$ ，其余地址没有影响，但必须设置为 V_{IL} 或 V_{IH} 。
6. 要继续编程到下一个连续地址单元，应输入 8 位命令 ADH ，后面紧跟要编程的 2 字节数据。 $A_0=0$ 时，数据字节 0 将被编程到初始地址 $[A_{23}-A_1]$ 中； $A_0=1$ 时，数据字节 1 将被编程到初始地址 $[A_{23}-A_1]$ 中。
7. 读状态寄存器继续使用当前时钟周期，直到被 $CE\#$ 上低电平到高电平的转换终止。
8. 当 $A_0=0$ 时读取制造商 ID，当 $A_0=1$ 时读取器件 ID。所有其他地址位均为 $00H$ 。制造商 ID 和器件 ID 输出流一直持续到被 $CE\#$ 上低电平到高电平的转换终止为止。

读 (25 MHz)

读指令 (03H) 支持最高为 25 MHz 的读操作。器件从指定的地址单元开始输出数据。数据输出流连续遍历所有地址，直到被 $CE\#$ 上低电平到高电平的转换终止。内部地址指针将自动递增，直到达到最高的存储器地址为止。达到最高的存储器地址后，地址指针将自动递增到地址空间的开始位置 (回绕)。从地址单元 $1FFFFFFH$ 读取数据之后，下一次输出将来自地址单元 $000000H$ 。

通过执行 8 位命令 $03H$ (后面紧跟地址位 $[A_{23}-A_0]$) 来启动读指令。在读周期的持续时间内， $CE\#$ 必须保持有效低电平。有关读序列，请参见图 5。

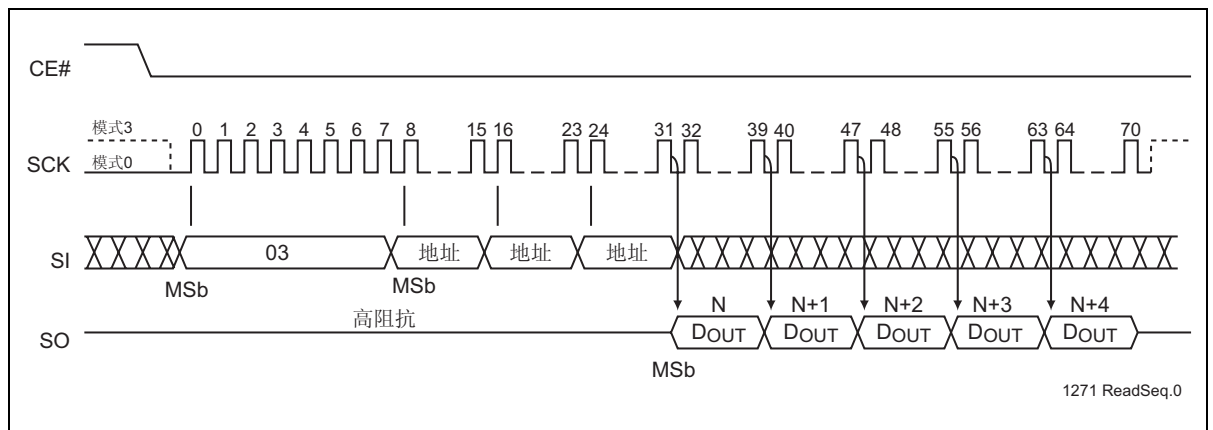


图 5: 读序列



高速读 (80 MHz)

高速读指令支持最高为 80 MHz 的读操作，通过执行 8 位命令 0BH（后面紧跟地址位 [A₂₃-A₀] 和一个空字节）来启动。在高速读周期的持续时间内，CE# 必须保持有效低电平。有关高速读序列，请参见图 6。

在一个空周期之后，高速读指令从指定的地址单元开始输出数据。数据输出流连续遍历所有地址，直到被 CE# 上低电平到高电平的转换终止。内部地址指针将自动递增，直到达到最高的存储器地址为止。达到最高的存储器地址后，地址指针将自动递增到地址空间的开始位置（回绕）。从地址单元 1FFFFFH 读取数据之后，下一次输出将来自地址单元 000000H。

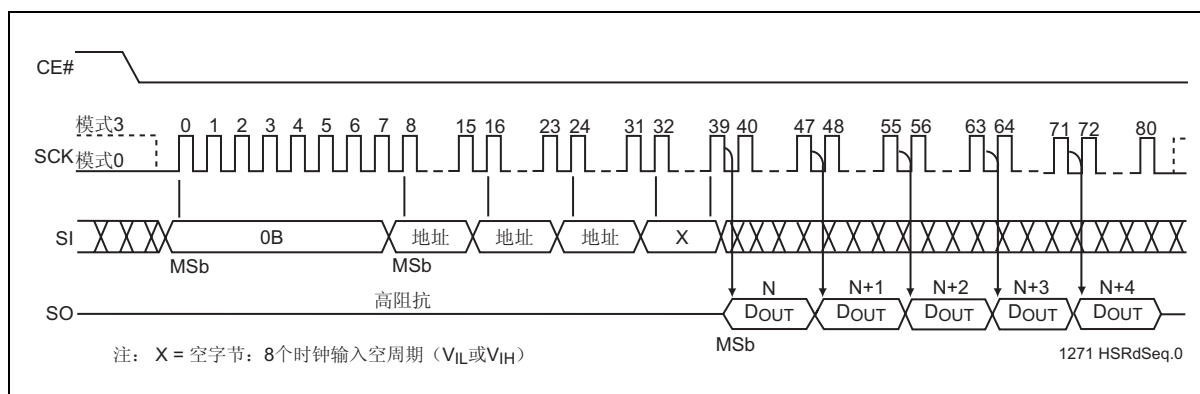


图 6: 高速读序列

字节编程

字节编程指令用于将所选字节中的位编程为所需数据。当启动编程操作时，所选字节必须处于擦除状态 (FFH)。应用到受保护存储器区域的字节编程指令将被忽略。

在执行任何写操作之前，必须先执行写使能 (WREN) 指令。在字节编程指令的持续时间内，CE# 必须保持有效低电平。通过执行 8 位命令 02H（后面紧跟地址位 [A₂₃-A₀]）来启动字节编程指令。在地址后面，按照从 MSb (bit 7) 到 LSb (bit 0) 的顺序输入数据。在执行指令之前，CE# 必须驱动为高电平。用户可以轮询软件状态寄存器中的 BUSY 位，或等待 T_{BP} 时间让内部自定时字节编程操作完成。有关字节编程序列，请参见图 7。

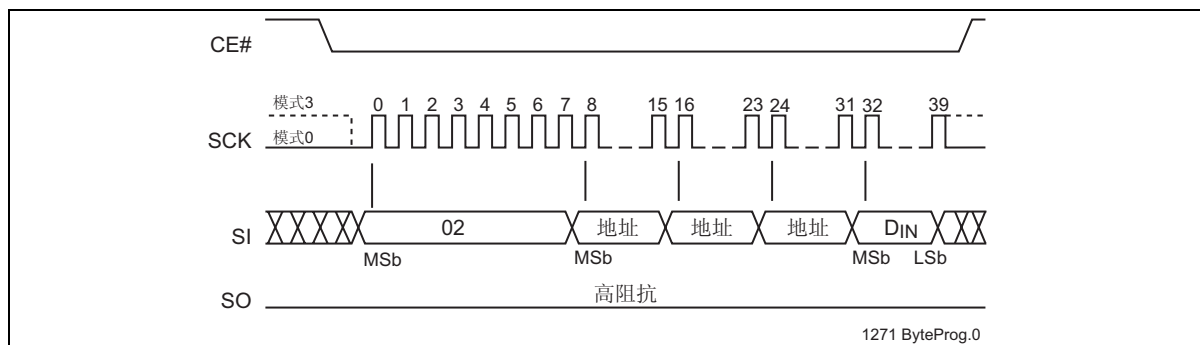


图 7: 字节编程序列



自动地址递增 (AAI) 字编程

AAI 编程指令允许对多个字节的数据进行编程，无需再次发出下一连续地址单元。当要对多个字节或整个存储器阵列编程时，该特性可减少总编程时间。指向受保护存储器区域的 AAI 字编程指令将被忽略。当启动 AAI 字编程操作时，所选地址范围必须处于已擦除状态 (FFH)。当处于 AAI 字编程序列中时，仅下列指令有效：软件写操作结束检测指令——AAI 字 (ADH)、WRDI (04H) 和 RDSR (05H)；硬件写操作结束检测指令——AAI 字 (ADH) 和 WRDI (04H)。可选择三种方式来确定每个 AAI 字编程周期是否完成：通过读取串行输出进行硬件检测、通过轮询软件状态寄存器中的 BUSY 位进行软件检测或等待 T_{BP} 的时间。有关详细信息，请参见“写操作结束检测”。

在执行任何写操作之前，必须先执行写使能 (WREN) 指令。通过执行 8 位命令 ADH (后面紧跟地址位 [A₂₃-A₀]) 来启动 AAI 字编程指令。在地址后面，连续输入两个字节的的数据，每个字节都从 MSb (Bit 7) 到 LSb (Bit 0)。A₀ = 0 时，第一个字节数据 (D0) 将编程到初始地址 [A₂₃-A₁] 中；A₀ = 1 时，第二个字节数据 (D1) 将编程到初始地址 [A₂₃-A₁] 中。在执行 AAI 字编程指令之前，CE# 必须驱动为高电平。在输入下一个有效命令之前，请检查 BUSY 状态。一旦器件指示不再忙，就可以对下两个连续地址的数据进行编程，然后对再下两个连续地址的数据进行编程，依此类推。

当编程最后一个所需字或最高的不受保护存储器地址时，通过硬件或软件 (RDSR 指令) 方法检查忙状态，以确定编程是否完成。编程完成之后，使用适用的方法终止 AAI。如果器件处于软件写操作结束检测模式，则执行写禁止 (WRDI) 指令 04H。如果器件处于 AAI 硬件写操作结束检测模式，则执行写禁止 (WRDI) 指令 04H，再执行 8 位 DBSY 命令 80H。在达到最高的不受保护的存储器地址之后，AAI 编程期间不存在回绕模式。有关 AAI 字编程序列，请参见图 10 和图 11。

写操作结束检测

在 AAI 字编程期间，可以通过三种方式确定编程周期是否完成：通过读取串行输出进行硬件检测、通过轮询软件状态寄存器中的 BUSY 位进行软件检测或等待 T_{BP} 的时间。下一节将介绍硬件写操作结束检测方法。

硬件写操作结束检测

硬件写操作结束检测方法可以消除在 AAI 字编程操作期间轮询软件状态寄存器中的 BUSY 位所产生的开销。8 位命令 70H 用于配置串行输出 (SO) 引脚，以在 AAI 字编程期间指示闪存忙状态 (见图 8)。必须在启动 AAI 字编程指令之前执行 8 位命令 70H。内部编程操作开始后，使能 CE# 会立即将内部闪存状态驱动 SO 引脚上。“0”表示器件忙，“1”表示器件准备好执行下一条指令。取消使能 CE# 将使 SO 引脚恢复到三态。当处于 AAI 和硬件写操作结束检测模式时，仅 AAI 字 (ADH) 和 WRDI (04H) 指令有效。

要退出 AAI 硬件写操作结束检测模式，首先执行 WRDI 指令 04H 以复位写使能锁存器位 (WEL=0) 和 AAI 位。然后执行 8 位 DBSY 命令 80H 以在 AAI 命令期间禁止 RY/BY# 状态。请参见图 9 和图 10。

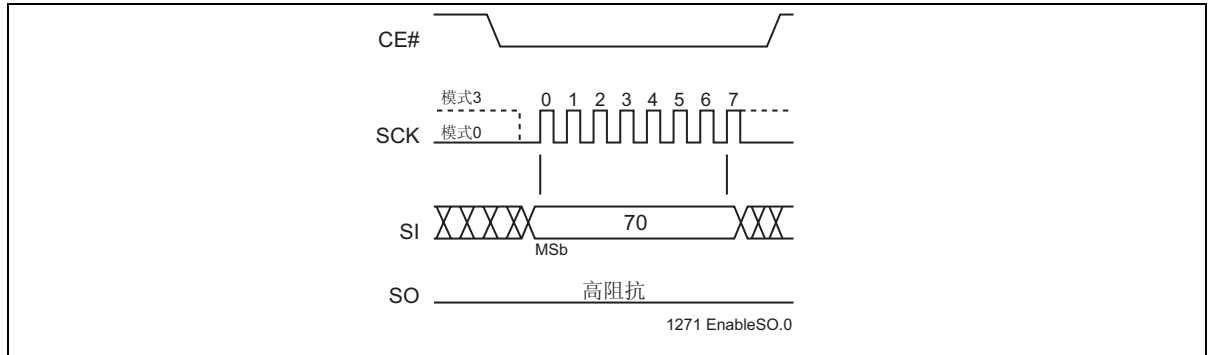


图 8: 在 AAI 编程期间使能 SO 作为硬件 RY/BY#

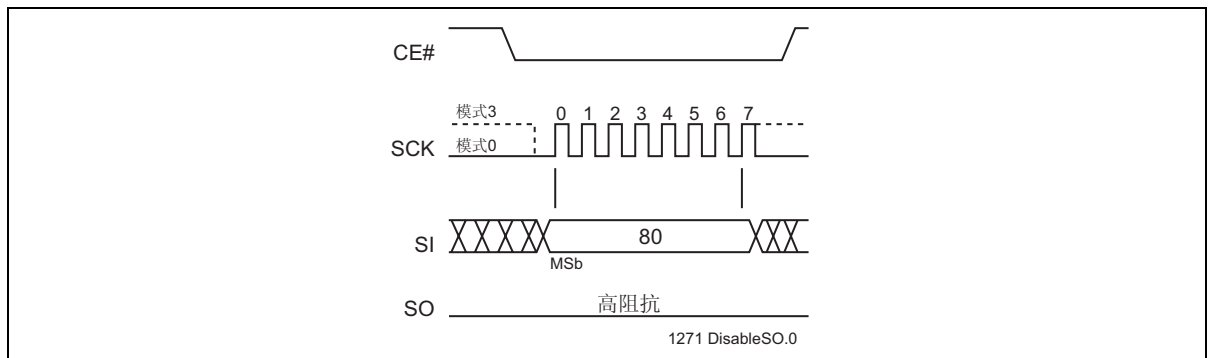


图 9: 在 AAI 编程期间禁止 SO 作为硬件 RY/BY#

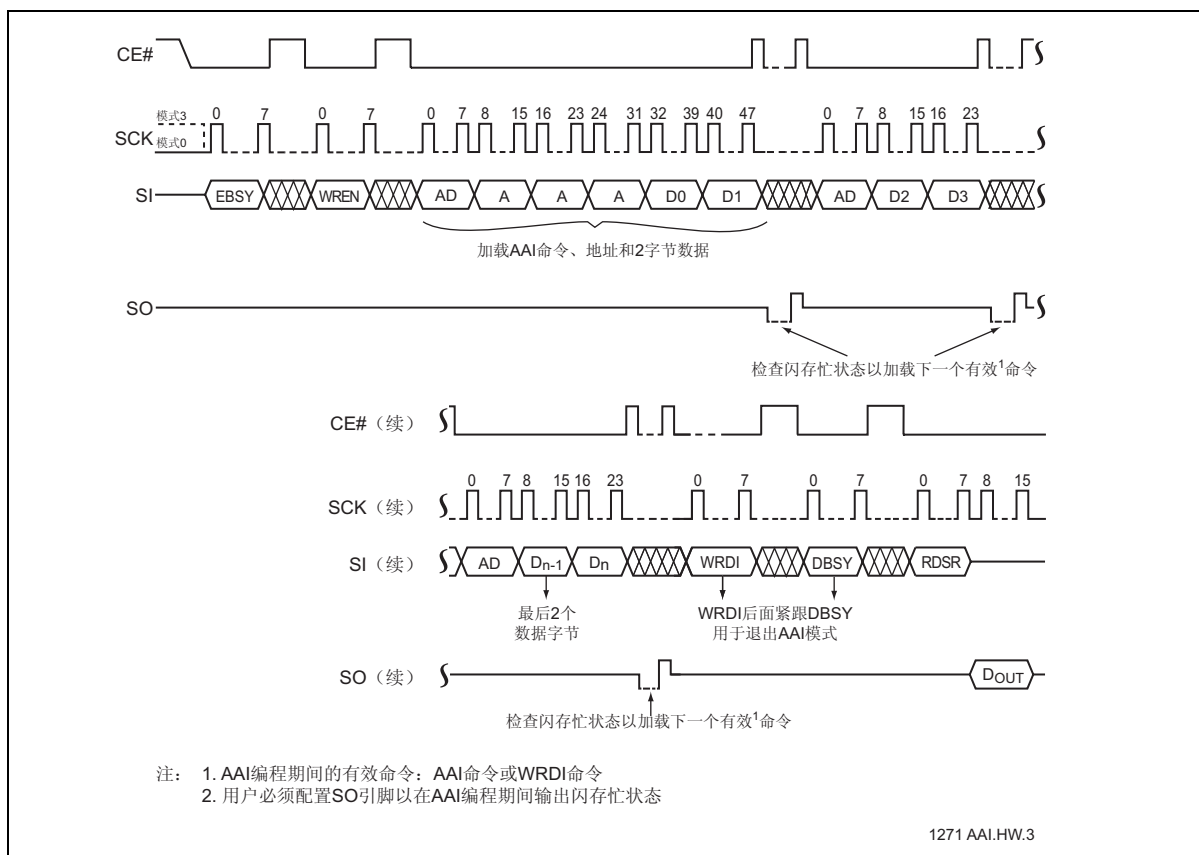


图 10: 使用硬件写操作结束检测时的自动地址递增 (AAI) 字编程序列

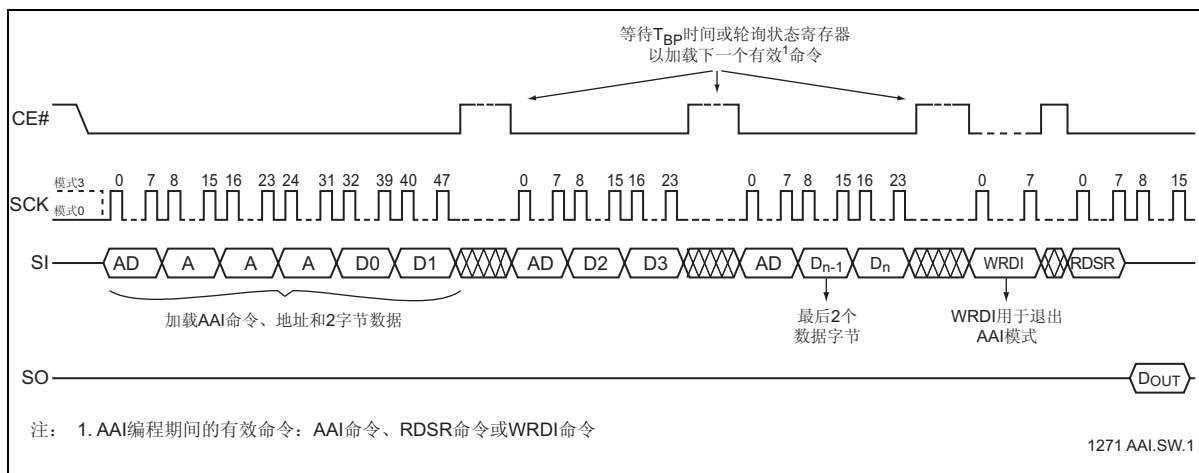


图 11: 使用软件写操作结束检测时的自动地址递增 (AAI) 字编程序列



4 KB 扇区擦除

扇区擦除指令会将所选 4 KB 扇区中的所有位清除为 FFH。应用到受保护存储器区域的扇区擦除指令将被忽略。在执行任何写操作之前，必须先执行写使能（WREN）指令。在任何命令序列的持续时间内，CE# 都必须保持有效低电平。通过执行 8 位命令 20H（后面紧跟地址位 [A₂₃-A₀]）来启动扇区擦除指令。地址位 [A_{MS}-A₁₂]（A_{MS} = 地址最高位）用于确定扇区地址（SA_X），其余地址位可以是 V_{IL} 或 V_{IH}。在执行指令之前，CE# 必须驱动为高电平。用户可以轮询软件状态寄存器中的 BUSY 位，或等待 T_{SE} 时间让内部自定时扇区擦除周期完成。有关扇区擦除序列，请参见图 12。

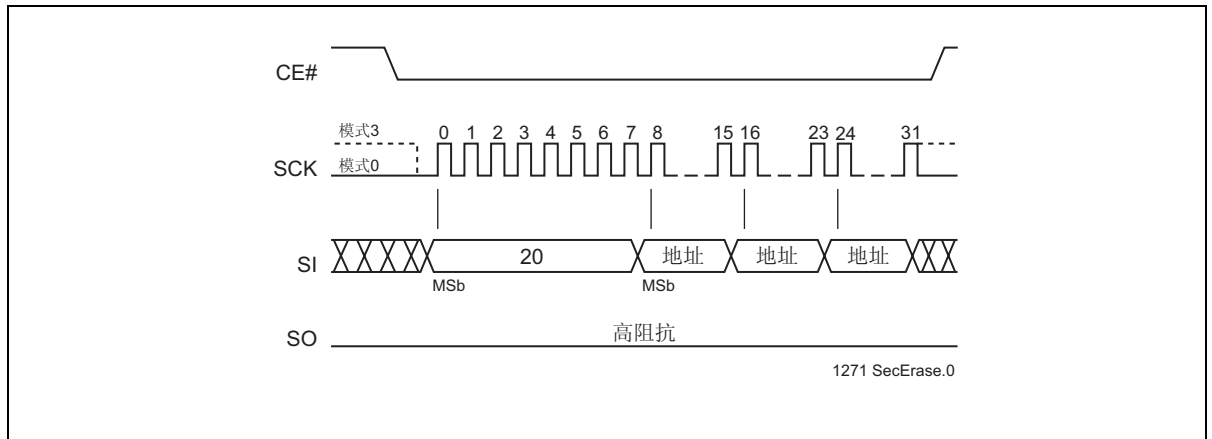


图 12: 扇区擦除序列



32 KB 和 64 KB 块擦除

32 KB 块擦除指令会将所选 32 KB 块中的所有位清除为 FFH。应用到受保护存储器区域的块擦除指令将被忽略。64 KB 块擦除指令会将所选 64 KB 块中的所有位清除为 FFH。应用到受保护存储器区域的块擦除指令将被忽略。在执行任何写操作之前，必须先执行写使能 (WREN) 指令。在任何命令序列的持续时间内，CE# 都必须保持有效低电平。通过执行 8 位命令 52H (后面紧跟地址位 [A₂₃-A₀]) 来启动 32 KB 块擦除指令。地址位 [A_{MS}-A₁₅] (A_{MS} = 地址最高位) 用于确定块地址 (BA_X)，其余地址位可以是 V_{IL} 或 V_{IH}。在执行指令之前，CE# 必须驱动为高电平。通过执行 8 位命令 D8H (后面紧跟地址位 [A₂₃-A₀]) 来启动 64 KB 块擦除指令。地址位 [A_{MS}-A₁₅] 用于确定块地址 (BA_X)，其余地址位可以是 V_{IL} 或 V_{IH}。在执行指令之前，CE# 必须驱动为高电平。用户可以轮询软件状态寄存器中的 BUSY 位，或等待 T_{BE} 时间让内部自定时 32 KB 块擦除或 64 KB 块擦除周期完成。有关 32 KB 块擦除和 64 KB 块擦除序列，请参见图 13 和图 14。

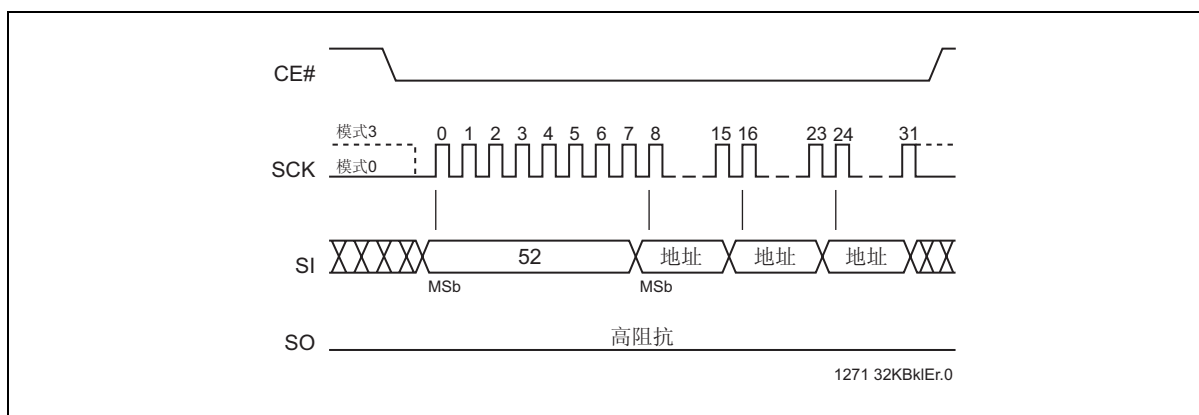


图 13: 32 KB 块擦除序列

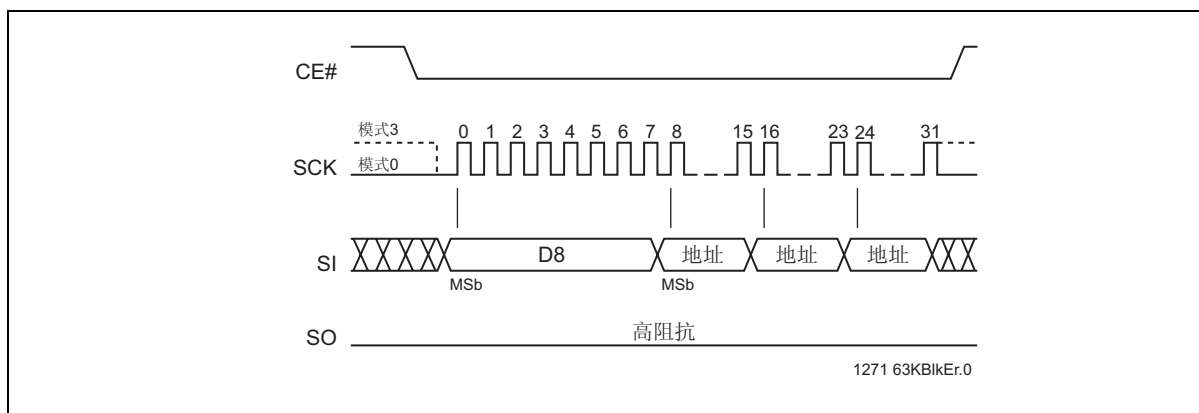


图 14: 64 KB 块擦除序列



全片擦除

全片擦除指令会将器件中的所有位清除为 FFH。如果有任何存储器区域受到保护，全片擦除指令将被忽略。在执行任何写操作之前，必须先执行写使能（WREN）指令。在全片擦除指令序列的持续时间内，CE# 必须保持有效低电平。通过执行 8 位命令 60H 或 C7H 来启动全片擦除指令。在执行指令之前，CE# 必须驱动为高电平。用户可以轮询软件状态寄存器中的 BUSY 位，或等待 T_{CE} 时间让内部自定时全片擦除周期完成。有关全片擦除序列，请参见图 15。

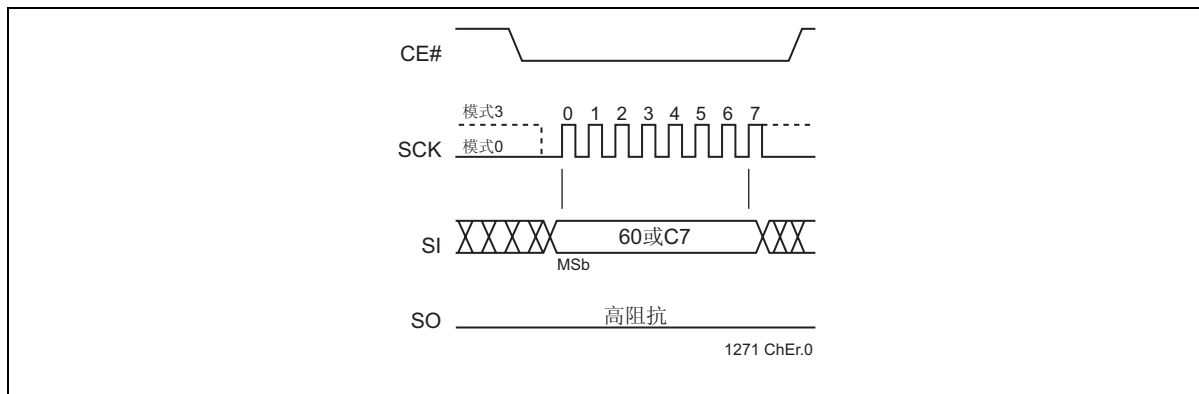


图 15: 全片擦除序列

读取状态寄存器（RDSR）

读取状态寄存器（RDSR）指令允许读取状态寄存器。可在任何时间读取状态寄存器，甚至是在写（编程 / 擦除）操作期间。当进行写操作时，应在发送任何新命令前检查 BUSY 位，以确保新命令会被器件正确接收。在输入 RDSR 指令之前，CE# 必须驱动为低电平，并保持低电平直到状态数据被读取为止。读状态寄存器继续使用当前时钟周期，直到被 CE# 上低电平到高电平的转换终止。有关 RDSR 指令序列，请参见图 16。

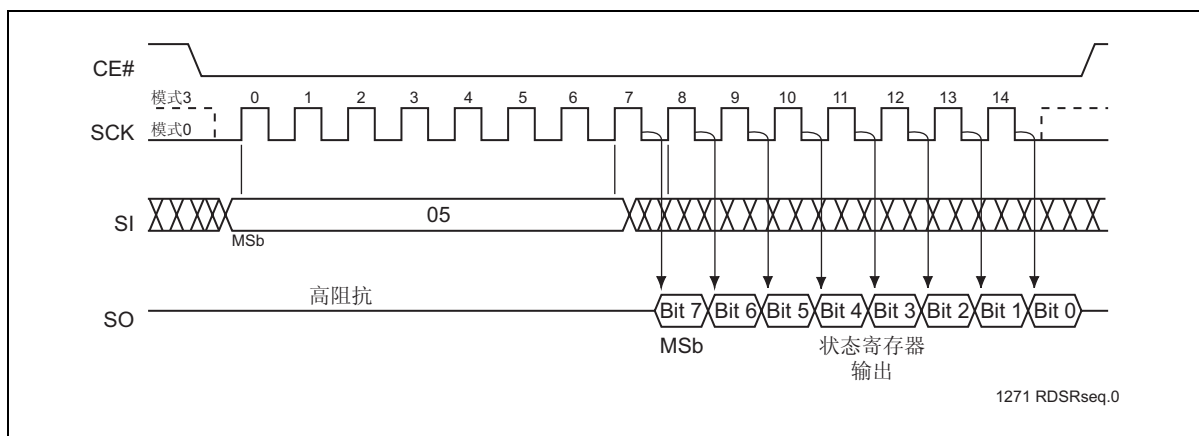


图 16: 读取状态寄存器（RDSR）序列



写使能 (WREN)

写使能 (WREN) 指令可将状态寄存器中的写使能锁存器位设置为 “1”，以允许进行写操作。在执行任何写 (编程 / 擦除) 操作之前，都必须先执行 WREN 指令。WREN 指令也可用于允许执行写状态寄存器 (WRSR) 指令；但是，状态寄存器中的写使能锁存器位将在 WRSR 指令的 CE# 上升沿被清零。在执行 WREN 指令之前，CE# 必须驱动为高电平。

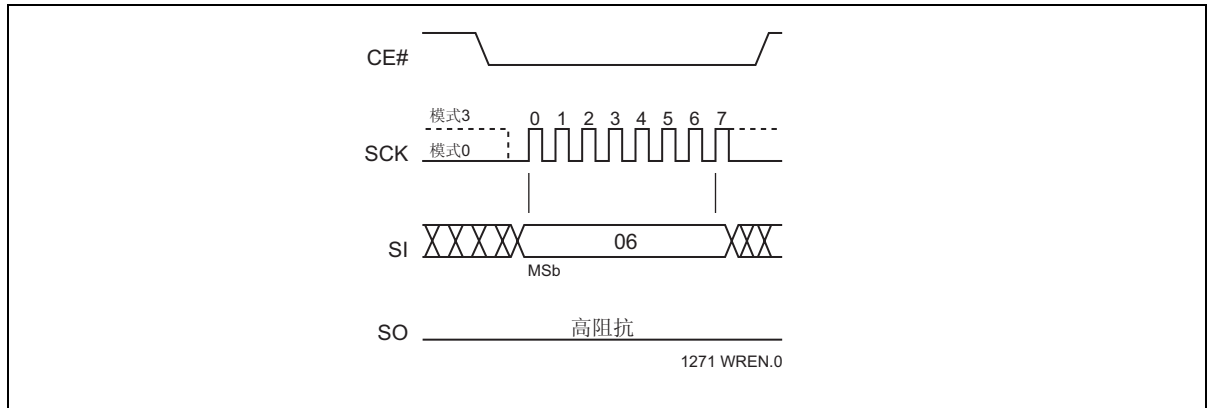


图 17: 写使能 (WREN) 序列

写禁止 (WRDI)

写禁止 (WRDI) 指令将写使能锁存器位和 AAI 位复位为 0，以禁止任何新的写操作发生。WRDI 指令不会终止任何正在进行的编程操作。执行 WRDI 指令之后，所有正在进行的编程操作都可以持续到 T_{BP}。在执行 WRDI 指令之前，CE# 必须驱动为高电平。

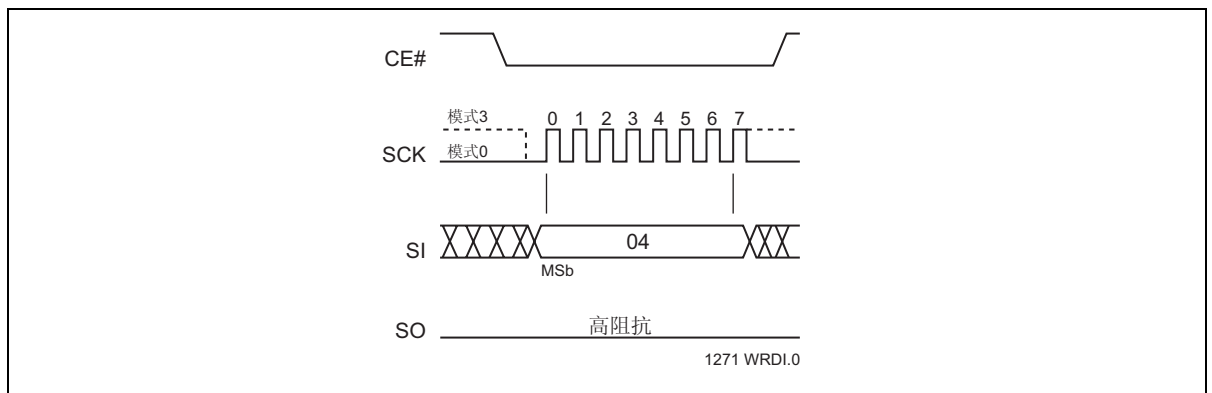


图 18: 写禁止 (WRDI) 序列



使能写状态寄存器 (EWSR)

使能写状态寄存器 (EWSR) 指令使能了写状态寄存器 (WRSR) 指令，可打开状态寄存器进行修改。在执行使能写状态寄存器指令之后，必须立即执行写状态寄存器指令。EWSR 指令后面紧跟 WRSR 指令，这个两步指令序列与 SDP (软件数据保护) 命令结构的工作方式类似，SDP 命令结构可防止状态寄存器值的意外修改。在输入 EWSR 指令之前，CE# 必须驱动为低电平；在执行 WRSR 指令之前，CE# 必须驱动为高电平。

写状态寄存器 (WRSR)

写状态寄存器指令将新值写入状态寄存器的 BP3、BP2、BP1、BP0 和 BPL 位。在输入 WRSR 指令的命令序列之前，CE# 必须驱动为低电平；在执行 WRSR 指令之前，CE# 必须驱动为高电平。有关 EWSR 或 WREN 和 WRSR 指令序列，请参见图 19。

当 WP# 为低电平且 BPL 位置 “1” 时，执行写状态寄存器指令将被忽略。当 WP# 为低电平时，只能将 BPL 位从 “0” 设置为 “1” 以锁定状态寄存器，但是不能将其从 “1” 复位到 “0”。当 WP# 为高电平时，BPL 位的锁定功能被禁止，并且状态寄存器中的 BPL、BP0、BP1 和 BP2 位都可以更改。只要在 WRSR 指令结束时的 CE# 引脚低电平到高电平的转换之前将 BPL 位置 0 或将 WP# 引脚驱动为高电平 (V_{IH})，就可以通过 WRSR 指令修改状态寄存器中的所有位。在这种情况下，单个 WRSR 指令即可将 BPL 位置 “1” 来锁定状态寄存器，同时修改 BP0、BP1 和 BP2 位。有关 WP# 和 BPL 功能的概述，请参见表 2。

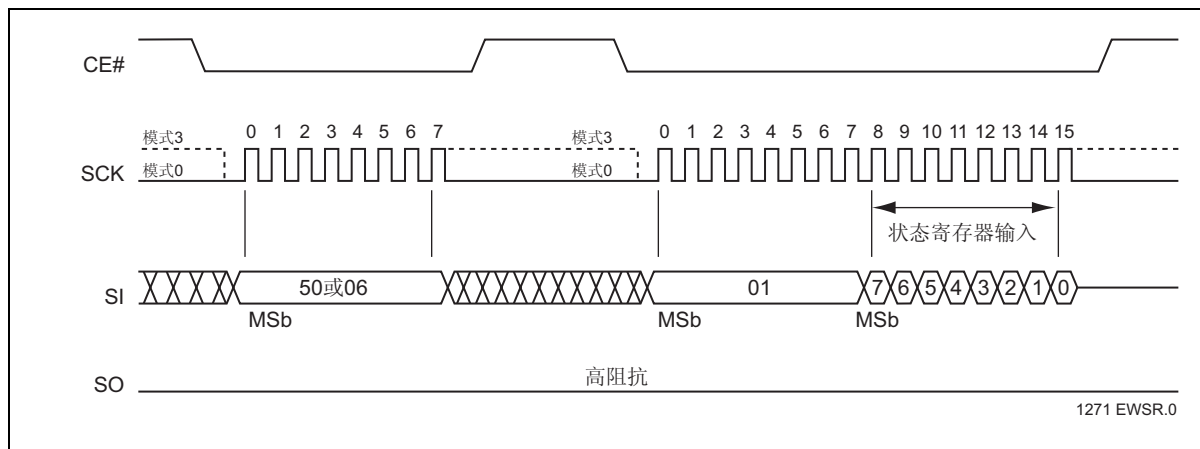


图 19: 使能写状态寄存器 (EWSR) 或写使能 (WREN) 和写状态寄存器 (WRSR) 序列



JEDEC 读 ID

JEDEC 读 ID 指令将器件标识为 SST25VF016B，将制造商标识为 SST。可通过执行 8 位命令 9FH 来读取器件信息。执行 JEDEC 读 ID 指令之后，将从器件输出 8 位制造商 ID (BFH)。然后，16 位器件 ID 在 SO 引脚上移出。字节 1 (BFH) 将制造商标识为 SST。字节 2 (25H) 将存储器类型标识为 SPI 串行闪存。字节 3 (41H) 将器件标识为 SST25VF016B。指令序列如图 20 中所示。在数据输出期间的任何时候，CE# 上低电平到高电平的转换都将终止 JEDEC 读 ID 指令。如果在执行 JEDEC 读 ID 指令之后未发出任何其他命令，则在进入待机模式 (CE#=V_{IH}) 之前发出 00H (NOP) 命令。

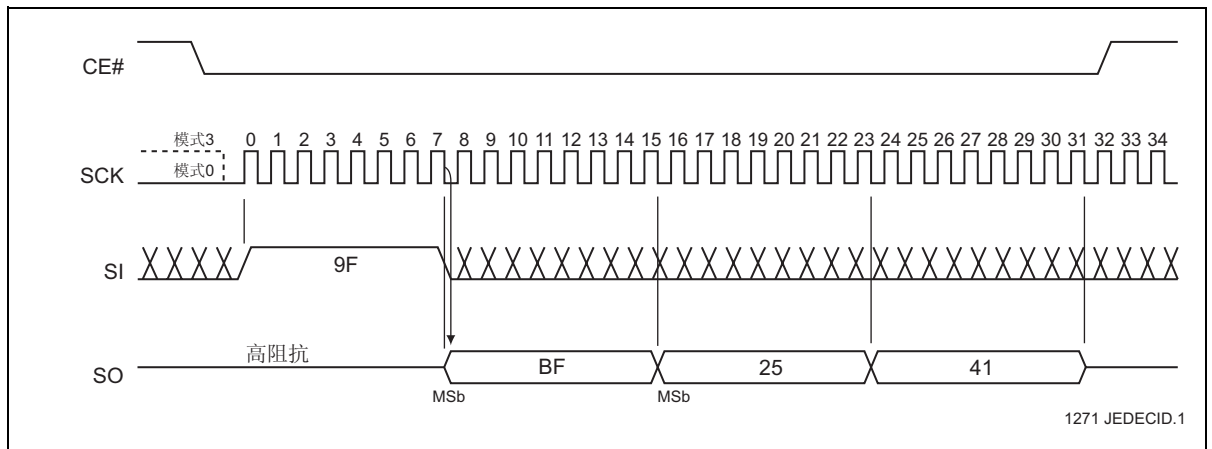


图 20: JEDEC 读 ID 序列

表 6: JEDEC 读 ID 数据

制造商 ID	器件 ID	
	存储器类型	存储器容量
字节 1	字节 2	字节 3
BFH	25H	41H

T6.0 25044



读 ID (RDID)

读ID指令 (RDID) 将器件标识为SST25VF016B, 将制造商标识为SST。此命令向后兼容所有SST25xFxxxA器件, 当在一个设计中使用多个版本的 SPI 串行闪存器件时, 应将此命令用作默认器件标识命令。通过执行 8 位命令 90H 或 ABH (后面紧跟地址位 [A₂₃-A₀]) 来读取器件信息。执行读 ID 指令之后, 制造商 ID 位于地址 00000H, 器件 ID 位于地址 00001H。器件处于读 ID 模式之后, 制造商 ID 和器件 ID 输出数据在地址 00000H 和 00001H 之间翻转, 直到被 CE# 上低电平到高电平的转换终止。

有关器件标识数据, 请参见表 6 和表 7。

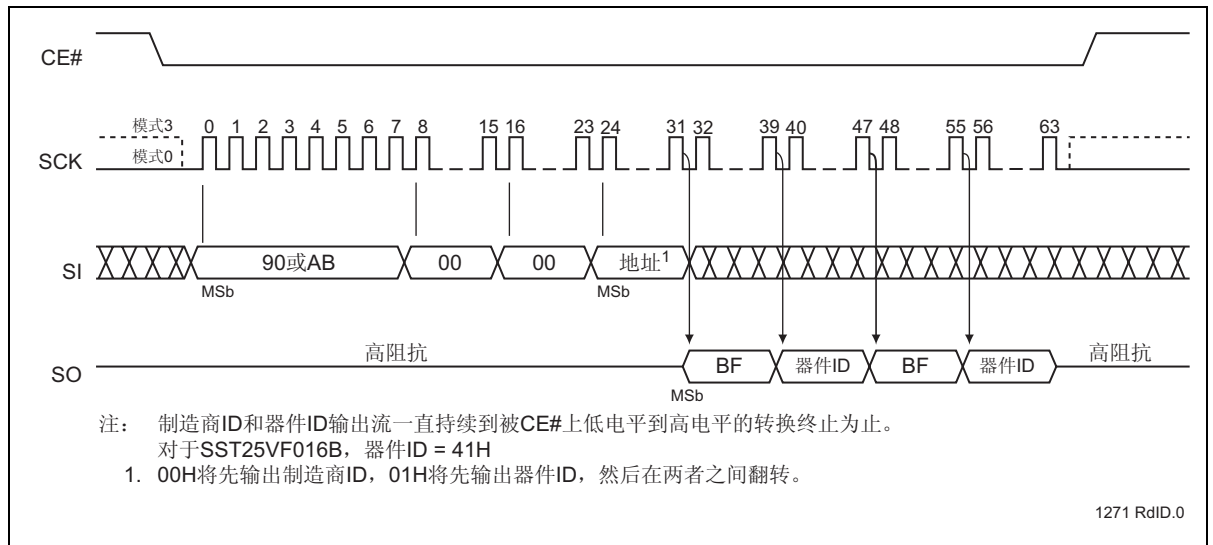


图 21: 读 ID 序列

表 7: 产品标识

	地址	数据
制造商 ID	00000H	BFH
器件 ID SST25VF016B	00001H	41H

T7.0 25044



电气规范

绝对最大极限参数（如果器件工作条件超过下述“绝对最大极限参数”，可能会对器件造成永久性损坏。这仅是极限参数，我们不建议器件工作在极限值甚至超过下述极限值。工作于绝对最大极限参数条件下时，器件可靠性可能会受到影响。）

偏置时的温度范围	-55°C 至 +125°C
存储温度范围	-65°C 至 +150°C
任意引脚上相对于地的直流电压	-0.5V 至 $V_{DD}+0.5V$
任意引脚上相对于地的瞬态电压 (<20 ns)	-2.0V 至 $V_{DD}+2.0V$
封装功率耗散能力 ($T_A = 25^\circ\text{C}$)	1.0W
表面贴装回流焊温度	260°C (10 秒)
输出短路电流 ¹	50 mA

1. 输出短路时间不超过 1 秒。每次短路的输出不超过 1 个。

表 8: 工作范围

范围	环境温度	V_{DD}
商业级	0°C 至 +70°C	2.7-3.6V
工业级	-40°C 至 +85°C	2.7-3.6V

表 9: 交流测试条件¹

输入上升 / 下降时间	输出负载
5 ns	$C_L = 30 \text{ pF}$

T9.1 25044

1. 请参见图 26 和图 27。



表 10: 直流工作特性

符号	参数	限制值			测试条件
		最小值	最大值	单位	
I _{DDR}	读电流		10	mA	CE#=0.1 V _{DD} /0.9 V _{DD} @25 MHz, SO= 开路
I _{DDR2}	读电流		15	mA	CE#=0.1 V _{DD} /0.9 V _{DD} @50 MHz, SO= 开路
I _{DDR3}	读电流		20	mA	CE#=0.1 V _{DD} /0.9 V _{DD} @80 MHz, SO= 开路
I _{DDW}	编程和擦除电流		30	mA	CE#=V _{DD}
I _{SB}	待机电流		20	μA	CE#=V _{DD} , V _{IN} =V _{DD} 或 V _{SS}
I _{LI}	输入泄漏电流		1	μA	V _{IN} =GND 至 V _{DD} , V _{DD} =V _{DD} 最大值
I _{LO}	输出泄漏电流		1	μA	V _{OUT} =GND 至 V _{DD} , V _{DD} =V _{DD} 最大值
V _{IL}	输入低电压		0.8	V	V _{DD} =V _{DD} 最小值
V _{IH}	输入高电压	0.7 V _{DD}		V	V _{DD} =V _{DD} 最大值
V _{OL}	输出低电压		0.2	V	I _{OL} =100 μA, V _{DD} =V _{DD} 最小值
V _{OL2}	输出低电压		0.4	V	I _{OL} =1.6 mA, V _{DD} =V _{DD} 最小值
V _{OH}	输出高电压	V _{DD} -0.2		V	I _{OH} =-100 μA, V _{DD} =V _{DD} 最小值

T10.0 25044

表 11: 电容 (T_A = 25°C, f=1 MHz, 其他引脚开路)

参数	说明	测试条件	最大值
C _{OUT} ¹	输出引脚电容	V _{OUT} = 0V	12 pF
C _{IN} ¹	输入电容	V _{IN} = 0V	6 pF

T11.0 25044

1. 该参数仅在初步认证时进行过测量, 其后经过可能影响该参数的设计或工艺变更。

表 12: 可靠性特性

符号	参数	最低规范	单位	测试方法
N _{END} ¹	可擦写次数	10,000	次	JEDEC 标准 A117
T _{DR} ¹	数据保存时间	100	年	JEDEC 标准 A103
I _{LTH} ¹	闩锁	100 + I _{DD}	mA	JEDEC 标准 78

T12.0 25044

1. 该参数仅在初步认证时进行过测量, 其后经过可能影响该参数的设计或工艺变更。



表 13: 交流工作特性

符号	参数	25 MHz		50 MHz		80 MHz		单位
		最小值	最大值	最小值	最大值	最小值	最大值	
F _{CLK} ¹	串行时钟频率		25		50		80	MHz
T _{SCKH}	串行时钟高电平时间	18		9		6		ns
T _{SCKL}	串行时钟低电平时间	18		9		6		ns
T _{SCKR} ²	串行时钟上升时间 (压摆率)	0.1		0.1		0.1		V/ns
T _{SCKF}	串行时钟下降时间 (压摆率)	0.1		0.1		0.1		V/ns
T _{CES} ³	CE# 有效建立时间	10		5		5		ns
T _{CEH} ³	CE# 有效保持时间	10		5		5		ns
T _{CHS} ³	CE# 非有效建立时间	10		5		5		ns
T _{CHH} ³	CE# 非有效保持时间	10		5		5		ns
T _{CPH}	CE# 高电平时间	100		50		50		ns
T _{CHZ}	CE# 高电平至输出高阻态		15		8		7	ns
T _{CLZ}	SCK 低电平至输出低阻态	0		0		0		ns
T _{DS}	数据输入建立时间	5		2		2		ns
T _{DH}	数据输入保持时间	5		5		4		ns
T _{HLS}	HOLD# 低电平建立时间	10		5		5		ns
T _{HHS}	HOLD# 高电平建立时间	10		5		5		ns
T _{H LH}	HOLD# 低电平保持时间	10		5		5		ns
T _{H HH}	HOLD# 高电平保持时间	10		5		5		ns
T _{HZ}	HOLD# 低电平至输出高阻态		20		8		7	ns
T _{LZ}	HOLD# 高电平至输出低阻态		15		8		7	ns
T _{OH}	从 SCK 变化开始输出保持的时间	0		0		0		ns
T _V	从 SCK 开始到输出有效的时间		15		8		6	ns
T _{SE}	扇区擦除		25		25		25	ms
T _{BE}	块擦除		25		25		25	ms
T _{SCE}	全片擦除		50		50		50	ms
T _{BP}	字节编程		10		10		10	μs

T13.0 25044

1. 读指令 03H 的最大时钟频率为 25 MHz。
2. 最大上升和下降时间可能受到 T_{SCKH} 和 T_{SCKL} 要求的限制。
3. 相对于 SCK。

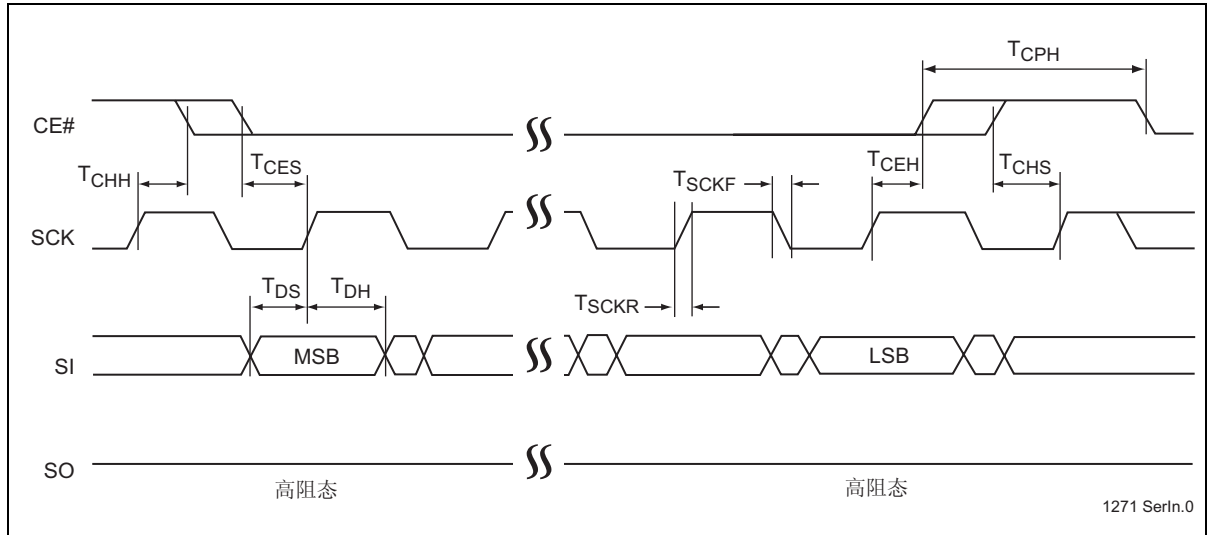


图 22: 串行输入时序图

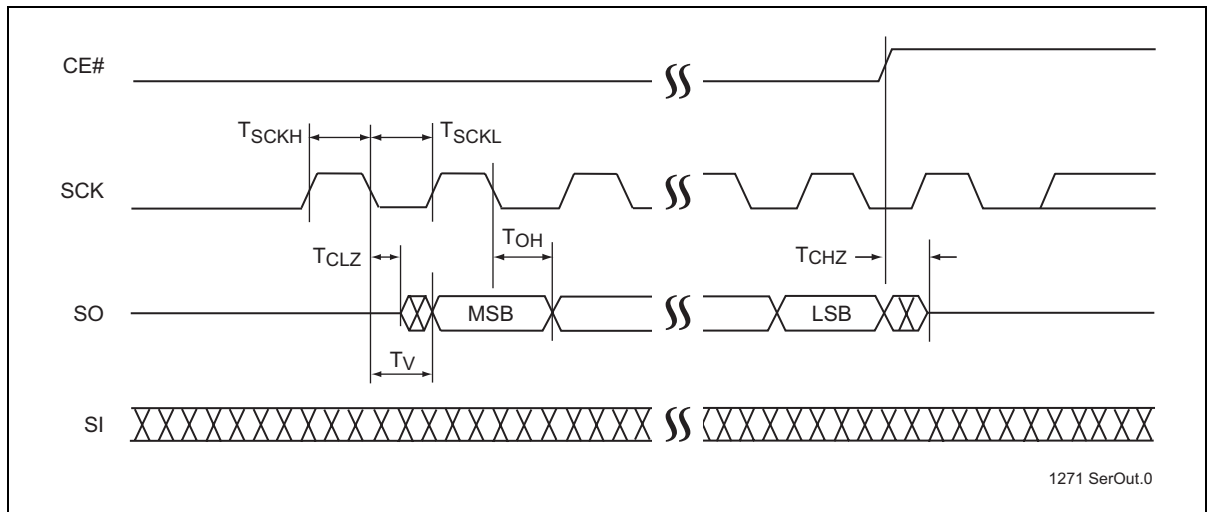


图 23: 串行输出时序图

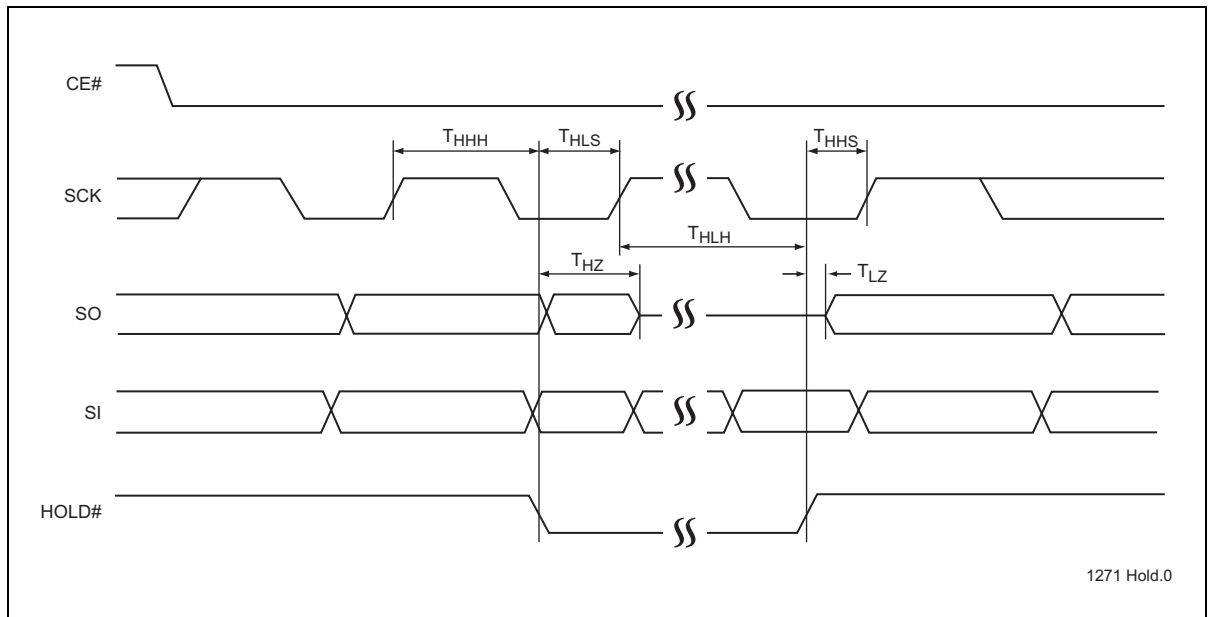


图 24: 保持时序图



上电规范

所有功能和直流规范值均针对 V_{DD} 斜升速率高于 $1V/100\text{ ms}$ ($0V$ 升至 $3.0V$ 的时间小于 300 ms) 的情形而规定。更多信息，请参见表 14 和图 25。

表 14: 建议的系统上电时序

符号	参数	最小值	单位
$T_{PU-READ}^1$	读操作的 V_{DD} 最小值	100	μs
$T_{PU-WRITE}^1$	写操作的 V_{DD} 最小值	100	μs

T14.0 25044

1. 该参数仅在初步认证时进行过测量，其后经过可能影响该参数的设计或工艺变更。

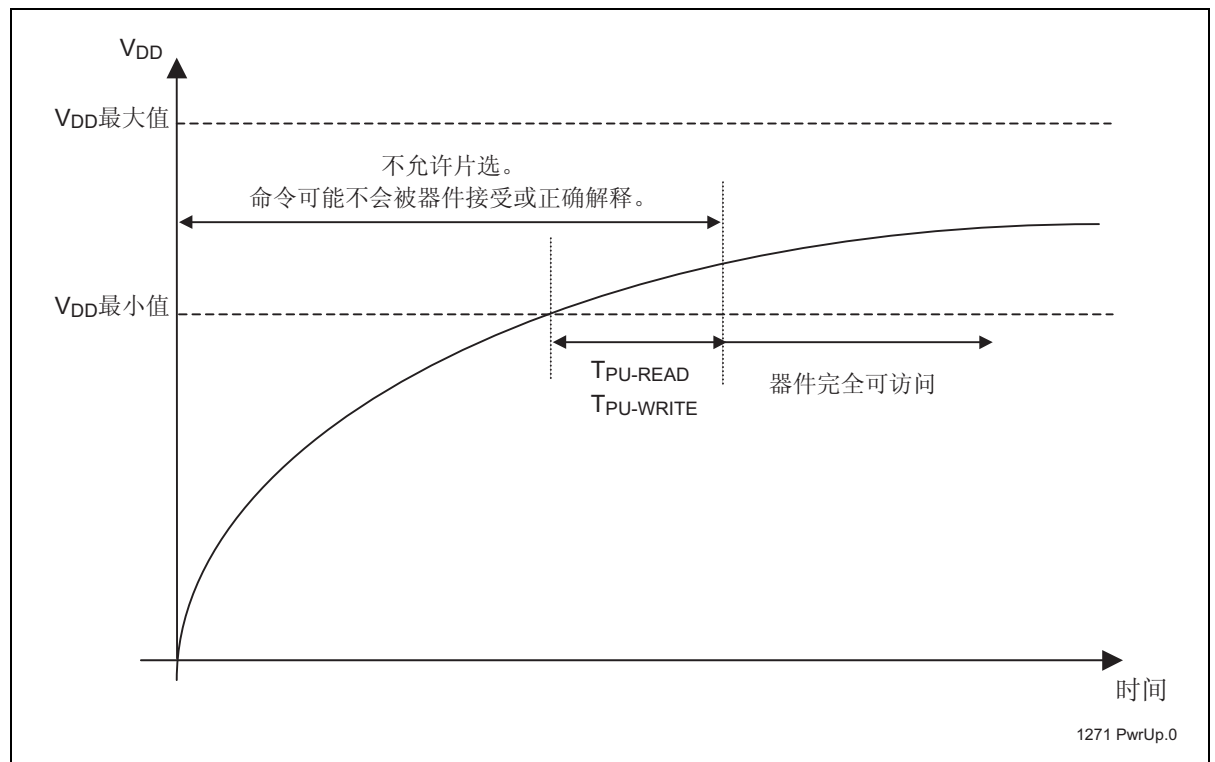


图 25: 上电时序图

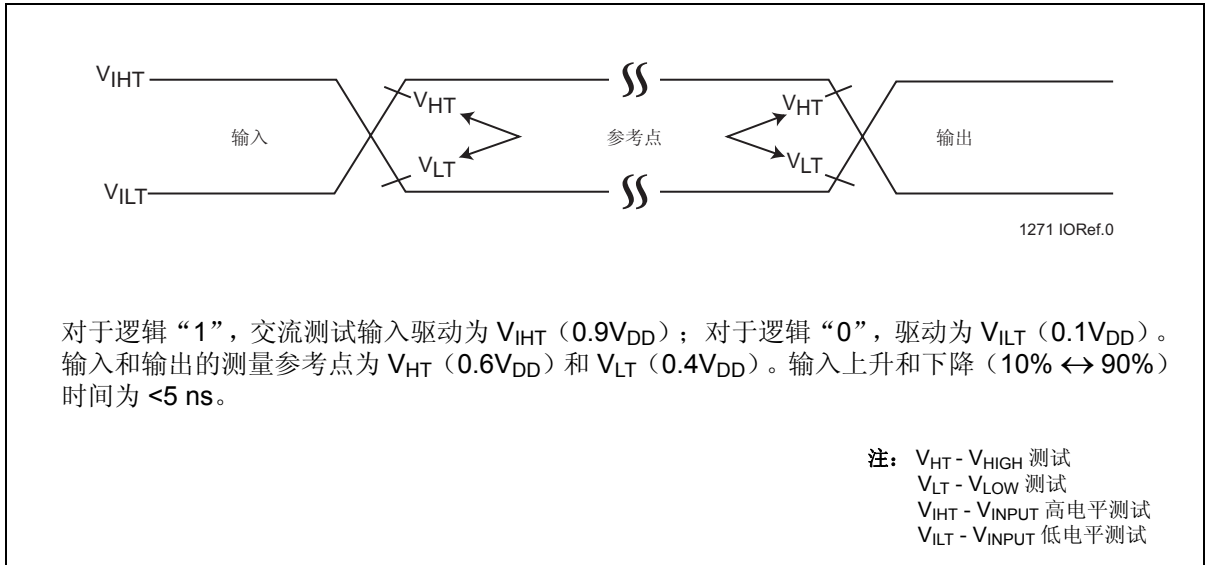


图 26: 交流输入 / 输出参考波形

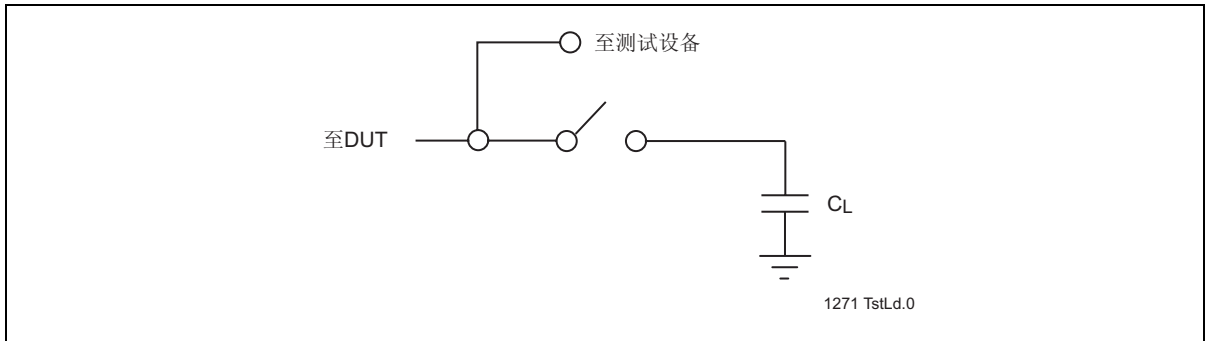
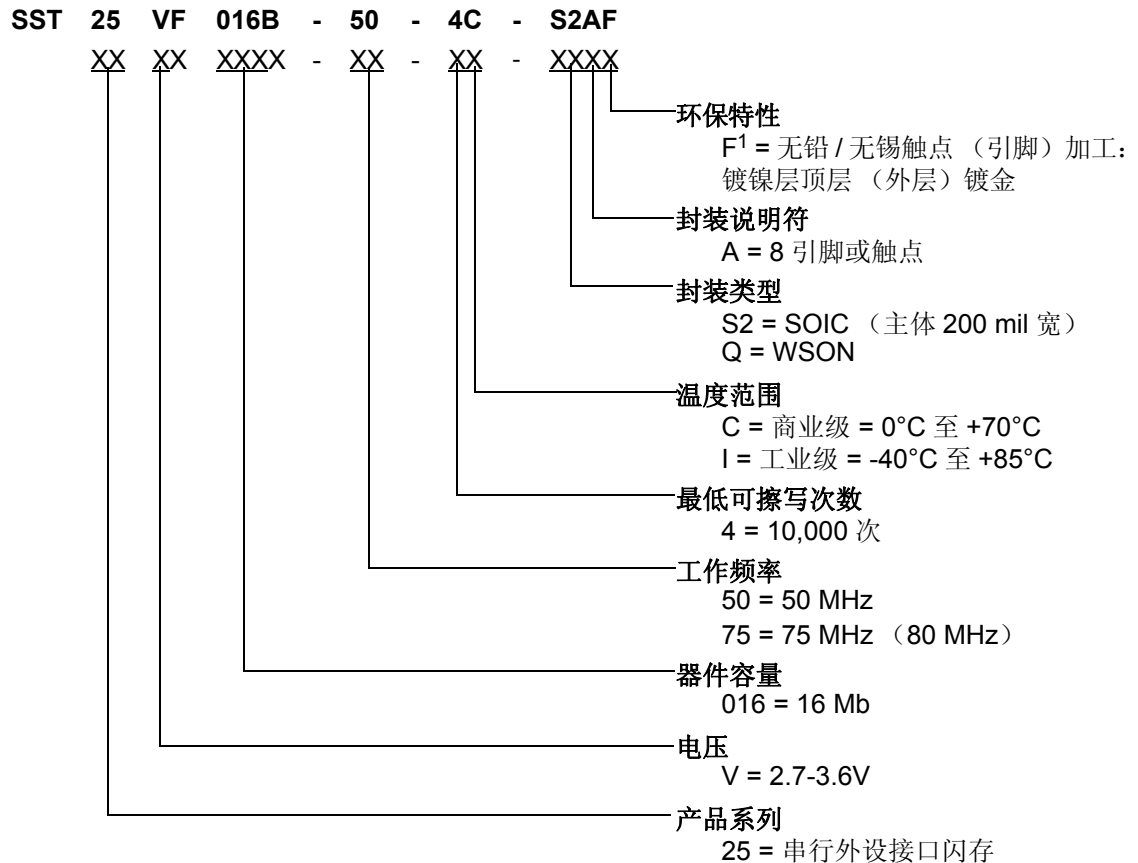


图 27: 测试负载示例



产品订购信息



1. 环保后缀“F”代表无铅 / 无锡锡焊。
SST 无铅 / 无锡锡焊器件均符合 RoHS 标准。

SST25VF016B 的有效组合

SST25VF016B-50-4C-S2AF	SST25VF016B-50-4C-QAF
SST25VF016B-50-4I-S2AF	SST25VF016B-50-4I-QAF
SST25VF016B-75-4I-S2AF	SST25VF016B-75-4I-QAF

注：有效组合是已量产或即将投入量产的产品。要确认有效组合的供应情况，以及确认新组合的供应情况，请咨询您的 SST 销售代表。



封装图

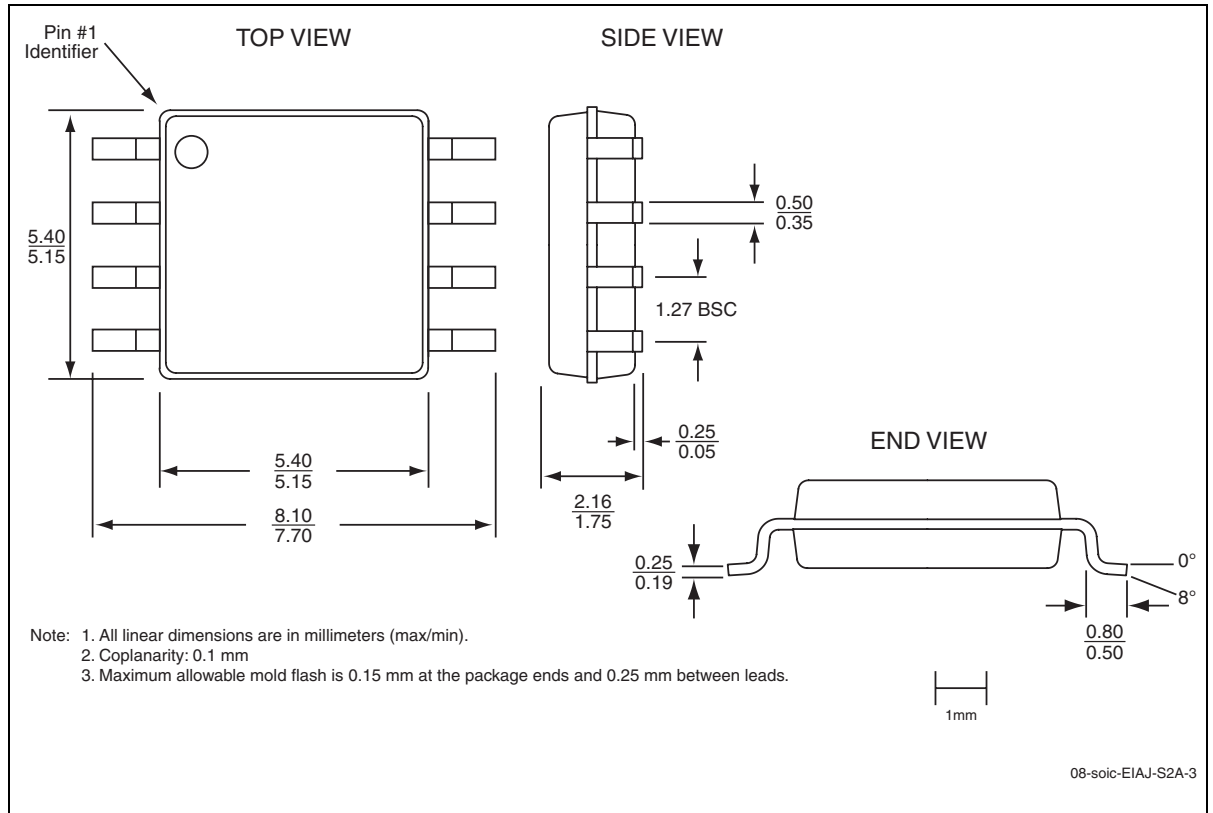


图 28: 8 引脚小外形集成电路 (SOIC) 主体 200 mil 宽 (5.2 mm x 8 mm)
SST 封装代码: S2A

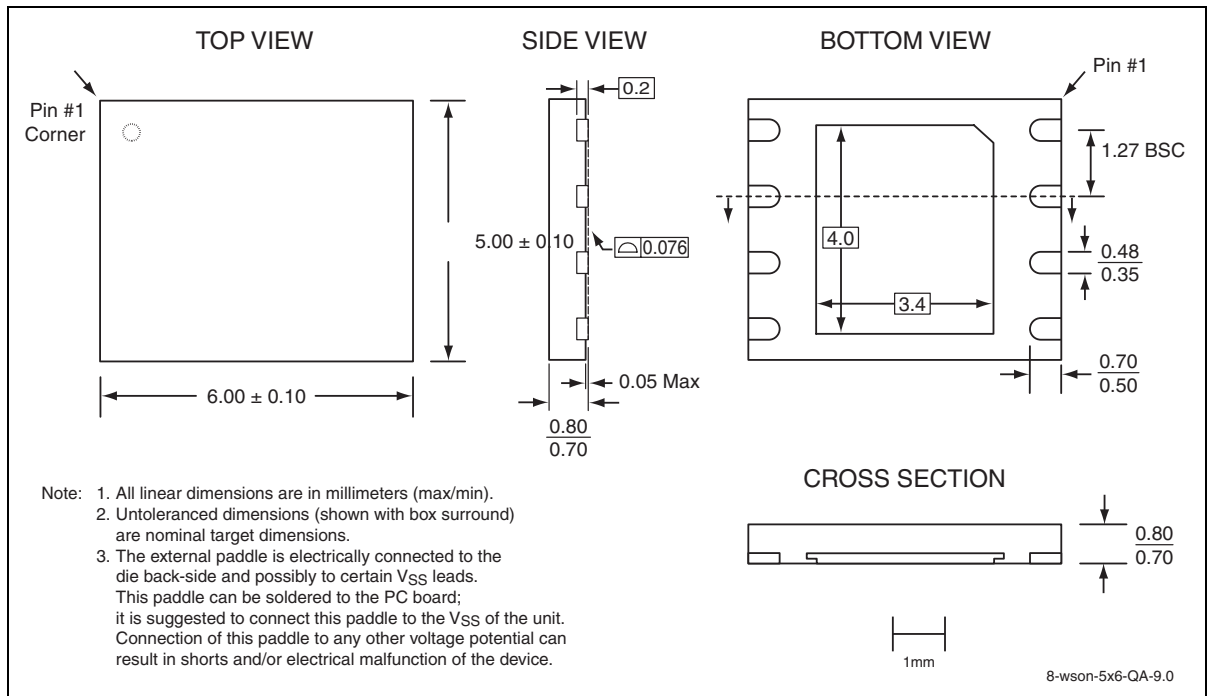


图 29: 8 触点超薄型小外形无引脚 (WSON)
SST 封装代码: QA



表 15: 版本历史

版本	说明	日期
00	<ul style="list-style-type: none"> 数据手册的初始版本。 	2005 年 4 月
01	<ul style="list-style-type: none"> 更正了第 20 页上的“JEDEC 读 ID”（包括时序图）。 更正了第 28 页上图 26 中的 V_{HT} 和 V_{LT} 的值。 	2005 年 9 月
02	<ul style="list-style-type: none"> 文档迁移至数据手册。 更新了表面贴装回流焊温度信息。 	2006 年 1 月
03	<ul style="list-style-type: none"> 将第 1 页“特性”中的时钟频率速度从 50 MHz 编辑为 80 MHz。 针对 80 MHz 修改了表 5。 针对 80 MHz 编辑了第 10 页的高速读。 编辑了第 21 页的表 8。 在第 22 页的表 12 中增加了 80 MHz 列。 更新了第 26 页的产品订购信息和有效组合。 	2008 年 9 月
04	<ul style="list-style-type: none"> 更新了第 12 页上的“自动地址递增 (AAI) 字编程”、“写操作结束检测”和“硬件写操作结束检测”。 修改了第 14 页上的图 10 和图 11。 将文档更新为新格式。 	2011 年 1 月
A	<ul style="list-style-type: none"> 增加了第 27 页上的“上电规范”。 更新了第 27 页上的表 14。 根据字母版本体系发布了文档。 将规范编号从 S71271 更新为 DS25044。 	2011 年 8 月

ISBN: 978-1-61341-7300

© 2011 Silicon Storage Technology, Inc. (Microchip Technology Inc. 的全资子公司)。版权所有。

SST、Silicon Storage Technology、SST 徽标、SuperFlash、MTP 和 FlashFlex 均为 Silicon Storage Technology, Inc. 的注册商标。MPF、SQI、Serial Quad I/O 和 Z-Scale 均为 Silicon Storage Technology, Inc. 的商标。在此提及的所有其他商标和注册商标均为各持有公司所有。

规范值可能发生变更，恕不另行通知。关于最新文档，请访问 www.microchip.com。最新封装图请至 <http://www.microchip.com/packaging> 查看封装规范。

存储器容量表示原始存储容量，实际可用容量可能会少一些。

除非在标准销售条款与条件中明确包含，否则 SST 不对其产品的使用作出任何担保。

关于销售办事处的地点和信息，请访问 www.microchip.com。

Silicon Storage Technology, Inc.
A Microchip Technology Company
www.microchip.com
