

1 Mbit (128K × 8) 四线 SPI nvSRAM

特性

- 容量
 - 1 Mbit (128K × 8)
- 带宽
 - 108 MHz 高速接口
 - 支持以 54 Mbps 的速度执行读写操作
- 串行外设接口
 - 时钟极性和相位模式 0 和 3
 - 多个 I/O 选项 — 单线 SPI (SPI)、双线 SPI (DPI) 以及四线 SPI (QPI)
- 高可靠性
 - 无限次读、写和回读周期
 - 一百万次的存储周期，用于将数据存储到 SONOS FLASH Quantum trap 中非易失性单元内
 - 数据保留时间：温度为 85°C 时，保留时间为 20 年
- 读取
 - 命令：标准、快速、双线 I/O 以及四线 I/O
 - 模式：突发包，持续 (XIP)
- 写入
 - 命令：标准、快速、双线 I/O 以及四线 I/O
 - 模式：突发循环
- 数据保护
 - 硬件：通过写保护引脚 (\overline{WP}) 提供保护
 - 软件：通过写禁用指令提供保护
 - 模块保护：状态寄存器位用于控制保护
- 特殊指令
 - STORE/RECALL：执行访问 SRAM 和 Quantum Trap 之间的数据
 - 序列号：用户可选的 8 字节 (OTP)
 - 标识号：4 字节的制造商 ID 和产品 ID
- 数据从 SRAM 存储到非易失性 SONOS FLASH Quantum Trap 内
 - 自动存储：断电时，通过使用小电容 (V_{CAP}) 自动存储数据
 - 软件：使用 SPI 指令 (STORE)
 - 硬件：HSB 引脚
- 数据从非易失性 SONOS FLASH Quantum Trap 回读到 SRAM 内
 - 自动回读：加电时，自动启动回读功能
 - 软件：使用 SPI 指令 (RECALL)
- 低功耗模式
 - 睡眠模式：温度为 85 °C 时，平均电流为 280 μ A
 - 休眠模式：温度为 85 °C 时，平均电流为 8 μ A
- 供电电压的工作范围
 - 内核 V_{CC} ：2.7 V ~ 3.6 V
 - I/O V_{CCQ} ：1.71 V ~ 2.0 V

■ 温度范围

- 扩展工业级范围：-40 °C ~ 105 °C
- 工业级范围：-40 °C ~ 85 °C

■ 封装

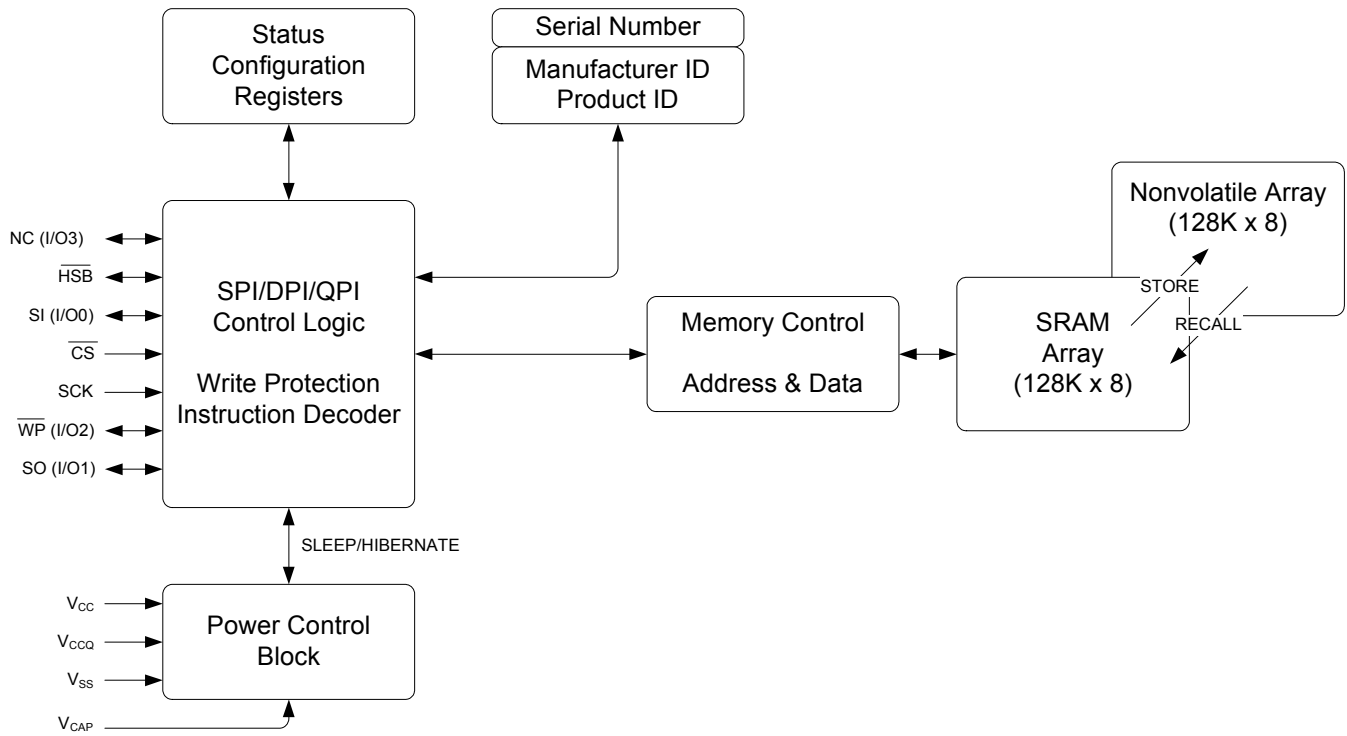
- 16-SOIC
- 24-FBGA

功能概述

赛普拉斯 CY14V101QS 将一个 1 Mbit nvSRAM 和一个 QPI 接口整合在一起。QPI 通过使用所选的操作码能够在单线模式（在每个时钟周期内使用一个 I/O 通道传输一位）、双线模式（在每个时钟周期内使用两个 I/O 通道传输两位）或四线模式（在每个时钟周期内使用四个 I/O 通道传输四位）下对存储器进行读写操作。

存储器被组织为 128 KB，该存储器包括 SRAM 和非易失性 SONOS FLASH Quantum Trap 单元。SRAM 能够实现无限次的读写周期，而非易失性单元则能够提供高可靠性的数据存储空间。断电时，数据会自动从 SRAM 中转移到非易失性存储器中（存储操作）。加电时，数据会从非易失性存储器回读到 SRAM（回读操作）内。也可通过 SPI 指令来触发“存储”和“回读”操作。

逻辑框图



目录

引脚分布	4	写入配置寄存器 (WRCR) 指令	35
引脚定义	5	标识寄存器 (RDID) 指令	36
器件操作	6	标识寄存器 (FAST_RDID) 指令	37
SRAM 写入	6	序列号寄存器写入 (WRSN) 指令	38
SRAM 读取	6	序列号寄存器读取 (RDSN) 指令	38
存储操作	6	快速读取序列号寄存器 (FAST_RDSN) 指令	40
自动存储操作	6	NV 特定指令	41
软件存储操作	7	软件存储 (STORE) 指令	41
硬件存储和 HSB 引脚操作	7	软件回读 (RECALL) 指令	41
回读操作	7	自动存储使能 (ASEN) 指令	42
硬件回读 (加电)	7	自动存储禁用 (ASDI) 指令	42
软件回读	7	最大额定值	43
禁用和使能自动存储	7	工作范围	43
四线串行外设接口	8	直流规范	43
SPI 概述	8	数据保留时间与耐久性	44
双线和四线 I/O 模式	10	电容	44
SPI 模式	10	热阻	44
SPI 操作功能	11	交流测试负载和波形	45
加电	11	交流测试条件	45
断电	11	RTC 特性	45
活动模式和待机模式	11	交流开关特性	46
SPI 功能说明	12	开关波形	46
状态寄存器	14	自动存储或加电回读	47
写入禁用 (WRDI) 指令	18	开关波形	48
写入使能 (WREN) 指令	18	软件控制的存储和回读周期	49
使能 DPI (DPIEN) 指令	19	开关波形	49
使能 QPI (QPIEN) 指令	19	硬件存储周期	50
使能 SPI (SPIEN) 指令	19	开关波形	50
SPI 存储器读取指令	20	订购信息	51
读指令	20	订购代码定义	51
快速读取指令	21	封装图	52
写指令	24	缩略语	53
系统资源指令	28	文档常规	53
软件复位 (RESET) 指令	28	测量单位	53
默认的恢复指令	29	文档修订记录	54
休眠 (HIBEN) 指令	30	销售、解决方案和法律信息	55
睡眠 (SLEEP) 指令	31	全球销售和 design 支持	55
寄存器指令	33	产品	55
读取状态寄存器 (RDSR) 指令	33	PSoC® 解决方案	55
写入状态寄存器 (WRSR) 指令	33	赛普拉斯开发者社区	55
读取配置寄存器 (RDICR) 指令	34	技术支持	55

引脚分布

图 1. 16-SOIC 标准引脚分配

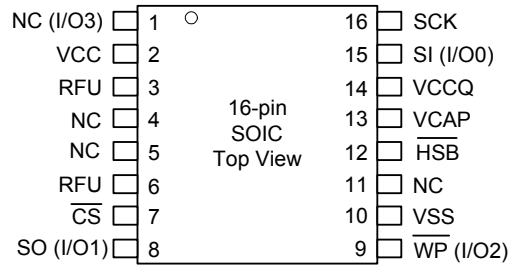


图 2. 16-SOIC 自定义引脚分布

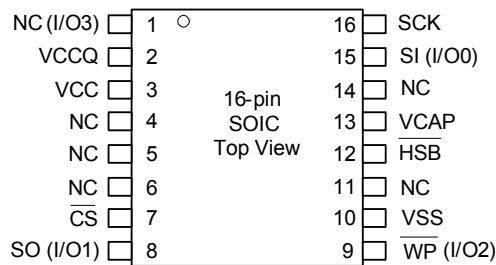
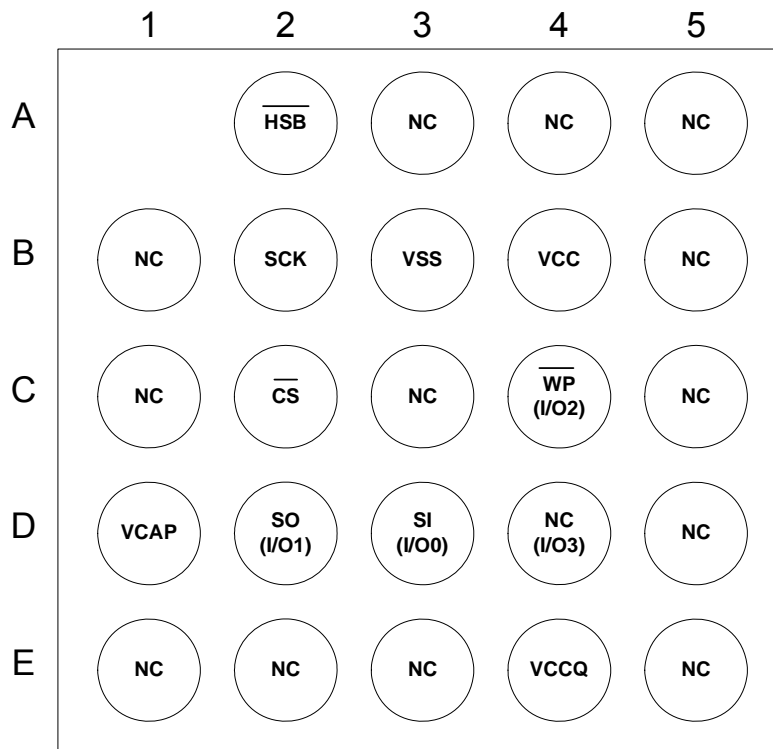


图 3. 24-FPGA 标准引脚分配 — 顶视图（球形焊盘的一面朝下）



引脚定义

引脚名称	I/O 类型	说明
NC (I/O3)	输入	未连接。在单线或双线模式下，不连接该引脚，使其处于悬空状态。该模式不支持 QSPI 指令。
	输入 / 输出	I/O3: 器件处于四线模式时，NC (I/O3) 引脚将作为 I/O3 引脚执行输入 / 输出操作。 在支持 SPI/DPI 指令的四线模式下， \overline{CS} 被使能时，该引脚应处于三态。
V _{CCQ}	电源	为器件上各 I/O 提供的电源电压。
V _{CC}	电源	为器件内核提供的电源电压。
\overline{CS}	输入	芯片选择。将该引脚置于低电平时，会激活该器件。通过将该引脚置于高电平，可以使器件进入待机模式。
SO (I/O1)	输出	串行输出。用于通过 SPI 输出数据的引脚。
	输入 / 输出	I/O1: 当该器件处于双线模式或四线模式时，SO (I/O1) 引脚作为 I/O1 引脚执行输入 / 输出操作。
\overline{WP} (I/O2)	输入	写保护。在 SPI/DPI 模式下实现硬件写保护。
	输入 / 输出	I/O2: 该器件处于四线模式时， \overline{WP} (I/O2) 引脚将作为 I/O2 引脚，执行输入 / 输出操作。
V _{SS}	接地	器件中内核和 I/O 的接地电压。
\overline{HSB}	输入 / 输出	硬件存储繁忙： 输出：该引脚为低电平时表示 nvSRAM 处于繁忙状态。每次执行完硬件存储后，都会使用标准输出高电流将 \overline{HSB} 驱动为高电平一小段时间 (t_{HHHD})，然后使用一个较弱的内部上拉电阻使该引脚保持为高电平状态（可以选用外部上拉电阻连接）。 输入：通过外部将该引脚下拉为低电平，可执行硬件存储。
V _{CAP}	电源	自动存储电容。断电期间为 nvSRAM 提供电源，以在该过程中将数据从 SRAM 存储到非易失性单元中。如果不需要自动存储，那么必须将该引脚置于“无连接”状态。该引脚绝对不能接地。
SI (I/O0)	输入	串行输入。用于输入所有 SPI 指令和数据的引脚。
	输入 / 输出	I/O0: 当该器件处于双线模式或四线模式时，SI (I/O0) 引脚将作为 I/O0 引脚执行输入 / 输出操作。
SCK	输入	串行时钟。运行速度可达到最大的 f_{SCK} 值。在该时钟的上升沿上锁存串行输入。将在该时钟的下降沿上驱动串行输出。
NC	—	未连接。
RFU	—	留作日后使用

器件操作

CY14V101QS 是一个带有 SONOS FLASH 非易失性元件，并且大小为 1 Mb 的四线串行接口 nvSRAM 存储器；在每个存储器单元中，SONOS FLASH 非易失性元件会与 SRAM 元件交叉存在。对 nvSRAM 进行的所有读和写操作都发生在 SRAM 上，它为 nvSRAM 提供了处理无限次存储器写入操作的独特能力。SRAM 中的数据由 STORE 序列保护，该序列将数据传输到非易失性单元中。断电时，使用小电容 (V_{CAP}) 自动将 SRAM 数据存储到非易失性单元内，从而确保数据完整性。采用可靠的 SONOS 技术制造的非易失性单元，使 nvSRAM 成为保护数据安全存储的最理想选择。

这个 1 Mb 的存储器阵列被组织为 128 KB。可以通过标准的 SPI 接口（单线模式、双线模式和四线模式）访问该存储器，该接口能够实现高达 40 MHz 的时钟速率，并且能够实现零周期延迟的读取和写入操作。该 SPI 接口还支持 108 MHz 的操作（单线模式、双线模式和四线模式），只有读取操作存在周期延迟。CY14V101QS 可作为 SPI 从设备，并支持 SPI 模式 0 和 3 (CPOL、CPHA = [0, 0] 和 [1, 1])。在单线模式和双线模式下使用芯片选择 (\overline{CS})、串行输入 (SI) (IO0)、串行输出 (SO) (IO1) 和串行时钟 (SCK) 引脚执行所有指令。在四线模式下还使用了 \overline{WP} IO2 和 IO3 引脚来执行指令、寻址和数据输入等操作。

该器件使用 SPI 操作码执行存储器访问。操作码支持在 SPI、双线数据、双线地址 / 数据、双线 I/O、四线数据、四线地址 / 数据和四线 I/O 模式下执行读 / 写操作。此外，它还包含四条特殊指令，通过这四条特殊指令可以使用四项 nvSRAM 特定功能：STORE（存储）、RECALL（回读）、自动存储禁用 (ASDI) 和自动存储使能 (ASEN)。

CY14V101QS 具有内置的数据安全功能。分别通过 \overline{WP} 引脚和 WRDI 指令提供了硬件和软件写保护。此外，存储器阵列模块通过状态寄存器模块保护位受写保护。

SRAM 写入

对 nvSRAM 的所有写操作都在 SRAM 上执行的，不会消耗 SONOS FLASH 非易失性存储器的擦写次数。这允许您执行无限次的写入操作。通过下面一个写指令可以启动写周期：WRITE、DIW、QIW、DIOW 和 QIOW。写指令包括一个写操作码、三字节地址和一字节数据。以 SPI 总线速度完成对 nvSRAM 进行的写操作，其周期延迟为零。

该器件允许在突发模式下进行写操作。这样，可以在连续的地址上使能写操作，无需发出新的 WRITE 指令。在突发模式下到达存储器的最后一个地址时，地址将翻转为 0x00000，然后该器件继续执行写入操作。

第 12 页上的 SPI 功能说明中的 nvSRAM 读写指令部分对 SPI 写周期序列进行了明确的定义。

SRAM 读取

对 nvSRAM 的所有读取操作都是以 SPI 总线速度在 SRAM 单元上执行的。读指令 (READ) 以 40 MHz 的频率执行，其周期延迟为零。它包括一个读操作码字节，后面是三字节的地址。在数据输出引脚 / 各引脚上读取得到该数据。

速率高于 40 MHz（最多可达 108 MHz）时，需要快速读取指令：FAST_READ、DOR、QOR、DIOR 和 QIOR。快速读取指令包括：一个快速读取操作码字节、三个地址字节和一个虚拟 / 模式字节。在数据输出引脚 / 各引脚上读取得到该数据。

该器件允许在突发模式下进行读操作。这样便可以在连续的地址上执行读操作，而无需发出新的 READ 指令。在突发模式下到达存储器的最后一个地址时，地址将翻转为 0x00000，然后该器件继续执行读取操作。

第 12 页上的 SPI 功能说明中的 nvSRAM 读写指令部分对 SPI 读周期序列进行了明确的定义。

存储操作

存储操作将 SRAM 中的数据传送到非易失性单元内。通过以下三种存储操作中的一种可存储数据：自动存储（器件断电时激活（需要 V_{CAP} ））、软件存储（通过 STORE 指令激活）以及硬件存储（通过 HSB 引脚激活）。在存储周期内，先擦除非易失性单元的内容，然后进行编程。启动存储周期后，将禁止对器件执行的读 / 写操作，直到该周期结束为止。

系统将监控状态寄存器中的 \overline{HSB} 信号或 WIP 位，以检测是否在执行存储周期。通过将 HSB 置于低电平或将 WIP 位置为 '1'，可以指示 nvSRAM 处于忙碌状态。为了避免不必要的非易失性存储，应忽略自动存储和硬件存储操作，除非在最新的存储周期后至少发生了一次 SRAM 写操作。但是，无论是否发生了 SRAM 写操作，都会执行软件触发的存储周期。

自动存储操作

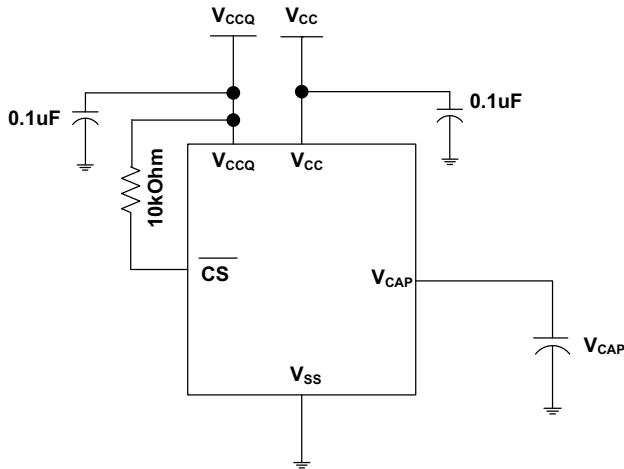
自动存储操作是 nvSRAM 独有的特性，该特性会在断电时自动将 SRAM 中的数据存储到 SONOS FLASH 非易失性单元内。该存储操作利用外部电容 (V_{CAP})，并在断电时，允许器件安全地将数据存储到非易失性存储器中。

正常工作时，器件从 V_{CC} 得到电流，以给连接至 V_{CAP} 引脚的电容充电。断电时，如果 V_{CC} 引脚上的电压降低低于 V_{SWITCH} ，那么器件将禁止对 nvSRAM 进行的所有存储器访问，并通过使用来自 V_{CAP} 电容的电荷自动执行存储操作。如果在上一次回读之后未执行写入周期，则不启动自动存储操作。

注意：如果电容未被连接到 V_{CAP} 引脚，那么必须通过发出自动存储禁用指令（第 42 页上的自动存储禁用（ASDI）指令）来禁用自动存储。如果 V_{CAP} 引脚上不存在电容时启用了自动存储，那么器件将在没有足够电荷的情况下尝试进行自动存储操作，以完成存储操作。这样会损坏存储在存储器阵列中的数据、序列号寄存器和状态寄存器。为了恢复正常的功能，需要更新它们。

图 4 显示了用于自动存储操作的存储电容（ V_{CAP} ）连接情况。请参考第 43 页上的直流规范以便了解 V_{CAP} 的大小。

图 4. 自动存储模式



软件存储操作

通过软件存储可以根据指令执行一次存储操作。无论是否已经执行了写操作，只要执行 STORE 指令便能够启动存储操作。

存储周期需要 t_{STORE} 时间才能完成，在此期间，对 nvSRAM 的所有存储器访问都被禁止。可以轮询状态寄存器中的 WIP 位或 HSB 引脚，以查找就绪状态或忙碌状态。完成 t_{STORE} 周期时间后，nvSRAM 将处于就绪状态以进行正常操作。

硬件存储和 HSB 引脚操作

器件中的 HSB 引脚是一个双目的引脚，用于启动存储操作或轮询存储 / 回读的完成状态。如果并未进行存储或回读操作，那么可以将 HSB 引脚驱动为低电平，以便开始硬件存储周期。

如果在 HSB 引脚上检测到低电平，那么 nvSRAM 将经过 t_{DELAY} 时间后启动存储操作。只有在上一个存储或回读周期后已经执行了 SRAM 写操作时，才会开始硬件存储周期。这样能够优化 SONOS FLASH 擦写次数。在 t_{STORE} 时间内禁止对存储器执行任何读写操作。HSB 引脚还能作为开漏驱动器（内部 100 kΩ 弱上拉电阻），它在进行存储 / 回读时被内部下拉到低电平以指示繁忙状态。

注意：每次硬件和软件存储操作完成后，HSB 通过标准输出高电流在短时间（ t_{HHHD} ）内变为高电平，然后通过内部 100 kΩ 上拉电阻一直保持高电平。

注意：若想成功进行最后一个数据字节的存储操作，在收到最后一个数据位 D0 至少一个时钟周期后启动硬件存储。

注意：建议仅在器件进入待机状态时执行硬件存储操作。同时，应该退出“芯片内执行”（XIP）模式。

存储操作完成后，如果 HSB 引脚返回为高电平，nvSRAM 存储器访问将在 t_{LZHSB} 时间内被禁止。如果不使用 HSB 引脚，则必须将它置于未连接状态。

回读操作

执行回读操作时，可将非易失性元件中所存储的数据传输到 SRAM 单元中。可通过下面某种方法启动回读操作：在加电时启动的硬件回读和通过 SPI 回读指令启动的软件回读。

在内部，回读需要执行两步流程。首先，清除 SRAM 中的数据（设置为‘0’）。然后，将非易失性信息传输到 SRAM 单元中。在进行回读周期时，所有存储器访问都将被禁止。回读操作不会更改非易失性单元中的数据。

硬件回读（加电）

加电时，如果 V_{CC} 的电压超过 V_{SWITCH} ，则会启动自动回读序列操作，该操作可以将非易失性存储器中的内容传输到 SRAM 单元中。

需要 t_{FA} 的时间才能完成加电回读周期，在此期间，存储器访问将被禁用。HSB 引脚用于检测器件的就绪状态。

软件回读

用户可通过软件回读来启动回读操作，从而将非易失性存储器中的内容重新存储到 SRAM 中。使用 RECALL（回读）指令触发软件回读操作。

需要 t_{RECALL} 时间才能完成软件回读操作。在此期间，对 nvSRAM 的所有存储器访问都被禁止。

禁用和使能自动存储

如果应用不需要自动存储功能，可通过 ASDI 指令禁用该功能。禁用后，nvSRAM 不会在断电时执行存储操作。

使用 ASEN 指令可以重新使能自动存储功能。然而，ASEN 和 ASDI 操作需要一个存储操作来使它们成为非易失性数据。

注意：出厂时已经使能了器件的自动存储功能，并且已向所有单元中写入了 0x00。

注意：如果禁用了自动存储功能，并且不需要 V_{CAP} ，那么 V_{CAP} 引脚必须保持为开路状态。 V_{CAP} 引脚绝不能接地。不可禁用加电回读操作。

四线串行外设接口

SPI 概述

SPI 是带有芯片选择 (\overline{CS})、串行输入 (SI)、串行输出 (SO) 和串行时钟 (SCK) 引脚的四引脚接口。该器件通过 SPI 接口可以对 nvSRAM 进行串行访问。器件上 SPI 总线的最大运行频率为 108 MHz。

SPI 是同步的串行接口，它使用时钟和数据引脚进行存储器访问并支持数据总线上的多个器件。使用 \overline{CS} 引脚可激活 SPI 总线上的器件。

芯片选择、时钟和数据之间的关系是由 SPI 模式决定的。该器件支持 SPI 的模式 0 和模式 3。在这两种模式下，数据都将在 SCK 上升沿（从 \overline{CS} 变为有效之后的第一个上升沿）记录到 nvSRAM 内。

SPI 协议由操作码控制。这些操作码规定了从总线主设备到从设备的所有指令。激活 \overline{CS} 后，总线主设备传输的第一个字节便是操作码。随后，可以传输任何地址和数据。在完成某个操作并发出新的操作码前， \overline{CS} 必须进入无效状态。下面列出了 SPI 协议中常用的术语：

SPI 主设备

SPI 主设备控制着 SPI 总线上进行的操作。SPI 总线上仅有一个主设备，但可有一个或多个从设备。所有从设备都共享了同一个 SPI 总线。主设备可通过 \overline{CS} 引脚选择任意一个从设备。要启动操作，必须由主设备将从设备的 \overline{CS} 引脚置于低电平状态来激活从设备。主设备生成 SCK（串行时钟），以便 SI 和 SO 线上的所有数据传输均与该时钟同步。

SPI 从设备

SPI 从设备由主设备通过芯片选择线激活。来自 SPI 主设备的串行时钟 SCK 作为从设备的输入，所有通信均与该时钟同步。SPI 从设备从来不会在 SPI 总线上执行通信，并且它会执行主设备发出的指令。

该器件可作为 SPI 从设备工作，并与其他 SPI 从设备共享 SPI 总线。

芯片选择 (\overline{CS})

要选择任意一个从设备，主设备必须下拉相应 \overline{CS} 引脚。只有 \overline{CS} 引脚为低电平时，才能将指令发送给从设备。如果器件未被选中，将忽略通过 SI 引脚的数据，同时，串行输出引脚 (SO) 保持高阻抗状态。

注意：必须从 \overline{CS} 的下降沿上开始执行新指令。因此，每个有效芯片选择周期内只能发送一个操作码。

注意：建议将与 V_{CCQ} 相连的 10 k Ω 的外部上拉电阻安装在 \overline{CS} 引脚上。

串行时钟 (SCK)

串行时钟由 SPI 主设备生成，在 \overline{CS} 变为低电平后，通信将与该时钟同步。

该器件能够在 SPI 模式 0 和模式 3 下进行数据通信。在这两种模式下，从设备在 SCK 的上升沿上锁存输入，输出在下降沿上发出。因此，SCK 的第一个上升沿表示 SI 引脚上 SPI 指令已接收到第一位 (MSB)。此外，所有数据输入和输出均与 SCK 同步。

数据传输 — SI/SO

SPI 数据总线包括 SI 和 SO 两条线，它们都可用于执行串行数据通信。SI 又称为主出从入 (MOSI)，SO 又称为主入从出 (MISO)。主设备通过 SI 引脚将指令发送到从设备，同时从设备通过 SO 引脚进行响应。如上所述，多个从设备可共享 SI 和 SO 线。

器件为 SI 和 SO 提供了两个可连接至主设备的独立引脚，如第 9 页上的图 5 所示。

该 SI 输入信号用于将数据串行传输给器件。它接收操作码、地址和需要编程的数据。各个数值在串行 SCK 时钟信号的上升沿上被锁存。在执行扩展 SPI 和 DPI/QPI 指令过程中，SI 变成 I/O0（一个输入 / 输出），用于接收操作码、地址和将被写入的数据（锁存在串行 SCK 时钟信号的上升沿上的值），并在 SCK 的下降沿上移出数据。

该 SO 输出信号用于串行传输器件中的数据。在串行 SCK 时钟信号的下降沿上移出数据。在执行扩展 SPI 和 DPI/QPI 指令过程中，SO 变成 I/O1（一个输入 / 输出），用于接收操作码、地址和需要编程的数据（锁存在串行 SCK 时钟信号的上升沿上的值），并在 SCK 的下降沿上移出数据。SO 集成了一个用于实现 Repeater/Bus-Hold 功能的电路。

写保护 (\overline{WP})

在 SPI 模式下，如果 \overline{WP} 引脚被驱动为低电平，可防止对状态寄存器进行写入操作，并保护存储器区域中的所有数据字节，该存储器区域受状态寄存器中块保护位的保护。

如果 \overline{WP} 被驱动为低电平，那么在执行 WRSR 指令期间以及状态寄存器的 SRWD 位被设置为 ‘1’ 时，便无法对状态寄存器和配置寄存器进行写操作。这样可以防止对块保护 (BP2、BP1、BP0) 以及 TBPROT 位进行任何更改。因此，如果 \overline{WP} 在执行 WRSR 指令过程中处于低电平，可防止对存储在存储器区域（受块保护和 TBPROT 位的保护）中的所有数据字节进行修改。

在四线传输模式下， \overline{WP} 功能不可用。在这些模式下，I/O2（作为输入 / 输出）会替代 \overline{WP} 功能，以便接收操作码、地址和需要写入 / 编程的数据以及移出数据。 \overline{WP} 带有一个内部上拉电阻；如果在四线传输模式下不使用该引脚，则在主机系统中可保持它的未连接状态。在 SPI 模式下， \overline{WP} 有一个 100 k Ω 的内部弱上拉电阻。

NC (I/O3)

在四线传输模式下，NC (I/O3) 引脚作为 I/O3 (输入 / 输出) 使用，可以接收操作码、地址和需要写入 / 编程的数据以及移出数据。NC (I/O3) 具有一个内部上拉电阻；如果未使用该引脚，则在主机系统中可以保持它的未连接状态。在 SPI 模式下，NC (I/O3) 带有一个 100 kΩ 的内部弱上拉电阻。

最高有效位 (MSB)

SPI 协议要求发送的第一位是最高有效位 (MSB)。该条件也适用于地址和数据传输。

1 Mb 串行 nvSRAM 需要一个 3 字节地址，以进行读 / 写操作。由于地址只有 17 位，所以器件会忽略所载入的前七位。虽然前 7 位无需关注，但赛普拉斯建议将这些位全部设置为 ‘0’，以使能无缝切换实现更大的存储容量。

串行操作码

通过将 \overline{CS} 置于低电平来选择从设备后，可将收到的第一个字节作为操作码进行期望的操作。该器件使用了标准操作码进行存储器访问。除了存储器访问外，它还提供了额外的操作码，用于实现 nvSRAM 特定的功能：STORE (存储)、RECALL (回读)、AutoStore Enable (自动存储使能) 以及 AutoStore Disable (自动存储禁用)。更多信息，请查阅第 12 上的表 2。

无效的操作码

如果收到无效的操作码，该操作码将被忽略。器件将忽略在 SI 引脚上的所有额外串行数据，直到 \overline{CS} 的下一个下降沿到来为止，与此同时，SO 引脚保持三态。

指令

操作码、地址以及模式 / 虚拟周期的组合，用于发出命令。

模式位

地址位后面的控制位。器件使用该控制位来使能 “芯片内执行” (XIP) 模式。这些位被指定时，系统控制器将驱动它们。

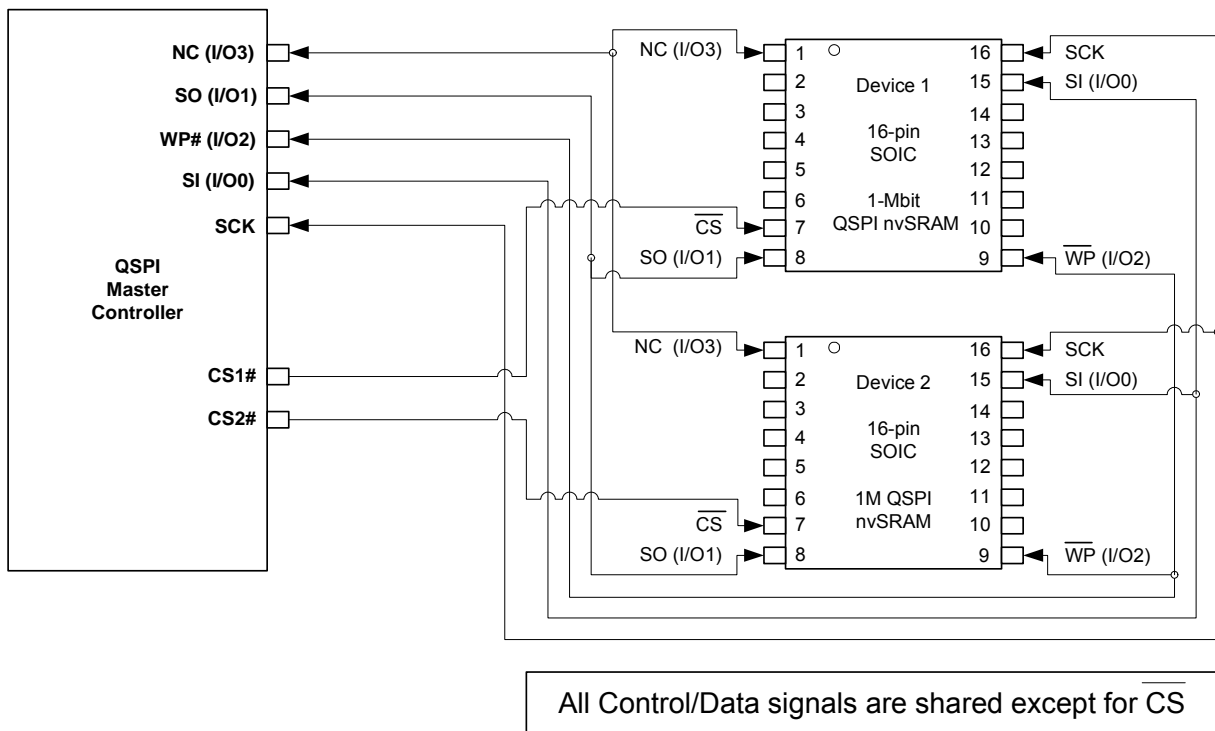
等待状态

在传输地址位或可选模式位后需要虚拟时钟周期。

状态寄存器

该器件具有一个 8 位的状态寄存器。状态寄存器中的位用于配置 SPI 总线。第 14 上的表 3 和表 4 对这些位进行了说明。

图 5. 采用多个 1 Mb 四线 SPI nvSRAM 进行的系统配置



双线和四线 I/O 模式

CY14V101QS 还可以重新配置标准的 SPI 引脚，以便能够在双线和四线 I/O 模式下工作。

当器件处于双线 I/O 模式时，SI 引脚和 SO 引脚将变成 I/O0 引脚和 I/O1 引脚，以便传输操作码、地址和数据（双线 I/O 模式）或传输地址和数据（双线地址 / 数据模式）或只传输数据（双线数据模式）。

当器件处于四线 I/O 模式时，SI、SO、 \overline{WP} 和 NC (I/O3) 引脚分别作为 I/O0、I/O1、I/O2 和 I/O3 引脚，以便传输操作码、地址和数据（四线 I/O 模式）或传输地址和数据（四线地址 / 数据模式），或只传输数据（四线数据模式）。

表 1. I/O 模式

协议	命令输入	地址输入	数据输入 / 输出
SPI	SI	SI	SI/SO
DPI	I/O[1:0]	I/O[1:0]	I/O[1:0]
QPI	I/O[3:0]	I/O[3:0]	I/O[3:0]
双线数据模式 (双线输出)	I/O[0]	I/O[0]	I/O[1:0]
双线地址 / 数据 模式 (双线 I/O)	I/O[0]	I/O[1:0]	I/O[1:0]
四线数据模式 (四线输出)	I/O[0]	I/O[0]	I/O[3:0]
四线地址 / 数据模式 (四线 I/O)	I/O[0]	I/O[3:0]	I/O[3:0]

更多详细信息，请参考数据手册中读和写时序框图。

SPI 模式

该器件还有重新配置功能。可通过微控制器来驱动它，该微控制器的 SPI 外设可运行于下述任意一种模式：

■ SPI 模式 0 (CPOL = 0, CPHA = 0)

■ SPI 模式 3 (CPOL = 1, CPHA = 1)

在这两种模式下，均在 SCK 的上升沿上锁存输入数据（SCK 上升沿是从 CS 有效后的第一个上升沿开始的）。如果时钟由 HIGH 状态开始（处于模式 3 时），则时钟切换后的第一个上升沿被捕获。输出数据在 SCK 的下降沿上有效。

图 6 和图 7 分别显示了两种 SPI 模式。当总线主设备处于待机模式并且不传输数据时，时钟的状态如下：

■ 在模式 0 下，SCK 保持为 ‘0’

■ 在模式 3 下，SCK 保持为 ‘1’

当器件通过将 CS 引脚设置为低电平状态而被选中时，它将通过 SCK 引脚的状态检测出 SPI 模式。器件被选中时，如果 SCK 引脚处于低电平状态，它将工作于 SPI 模式 0；如果 SCK 引脚处于高电平状态，它将工作于 SPI 模式 3。

图 6. SPI 模式 0

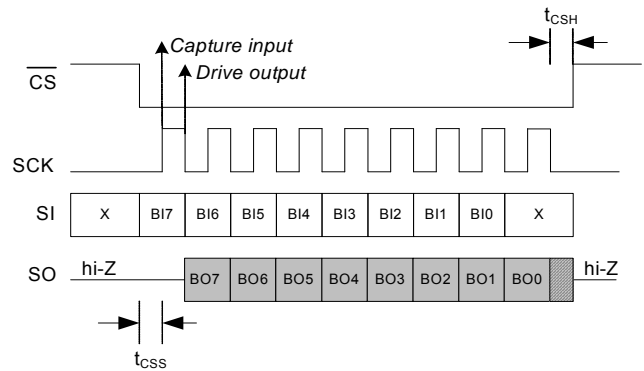
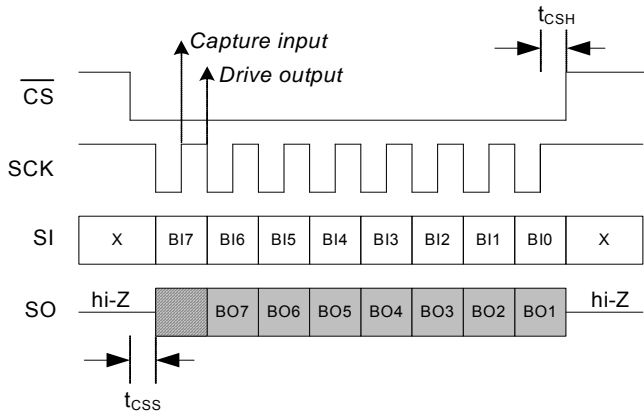


图 7. SPI 模式 3



SPI 操作功能

加电

加电是指启动电源和 V_{CC} 超过 V_{SWITCH} 电压时的状态。

如上面所述，加电时，nvSRAM 在 t_{FA} 的时间内进行加电回读操作。在此期间，所有存储器访问都被禁用。加电后，可检测 HSB 引脚以确定 nvSRAM 处于就绪 / 繁忙状态。

加电后，器件的状态如下：

- SPI I/O 模式
- 为 \overline{HSB} 激活上拉电阻
- \overline{SO} 为三态
- 待机功耗模式（如果 \overline{CS} 引脚处于高电平状态）。活动模式（如果 \overline{CS} 引脚处于低电平状态）。
- 状态寄存器的状态：
 - 写使能（WEN）位被复位为 ‘0’
 - 自前一个存储操作后，SRWD 保持不变
 - 自前一个存储操作后，SNL 保持不变
 - 自前一个存储操作后，块保护位保持不变

- \overline{WP} 和 NC（I/O3）功能由四线数据宽度（QUAD）CR[1] 位定义。如果四线数据宽度 CR[1] 为逻辑 ‘0’，那么 \overline{WP} 和 NC（I/O3）的上拉电阻被激活。

断电

断电时（ V_{CC} 连续减小），如果 V_{CC} 从正常工作电压下降到低于 V_{SWITCH} 电压阈值，则器件对发送到它的任何指令停止响应。

断电时，如果当前正在进行写周期，且已经接收到最后一个数据位 D0，则可以在 t_{DELAY} 的时间内完成写操作。然后，将禁止对存储器进行的所有访问，并执行自动存储操作（如果在上一次回读周期后未执行写操作，那么不会进行自动存储操作）。该功能用于防止断电时意外写入到 nvSRAM 内。

但是，为彻底避免在断电期间发生的无意写入，则需要确保器件未被选中，并且处于待机模式，并确保 \overline{CS} 采用 V_{CC} 上的电压。

活动模式和待机模式

当 \overline{CS} 处于低电平状态时，器件被选中且处于有效功耗模式。器件会消耗第 43 页上的直流规范中所指定的 I_{CC} （ $I_{CC1} + I_{CCQ1}$ ）电流。 \overline{CS} 处于高电平状态时，器件未被选中；如果未进行存储或回读周期，那么器件将进入待机状态。如果正在进行存储 / 回读周期，则该周期完成后器件将进入待机状态。

SPI 功能说明

该器件采用了一个 8 位指令寄存器。表 2 列出了指令及其操作码。所有指令、地址和数据都在 \overline{CS} 从高电平转为低电平时开始传输。通过 SPI 指令和 WP、NC (I/O3) 以及 HSB 引脚，可以对 nvSRAM 的所有功能进行访问。

表 2. 指令集

指令类别	指令名	操作码	SPI	双线输出	四线输出	双线 I/O	四线 I/O	DPI	QPI	最大频率 (MHz)
控制										
写禁用	WRDI	04h	支持	—	—	—	—	支持	支持	108
写入使能	WREN	06h	支持	—	—	—	—	支持	支持	108
使能 DPI	DPIEN	37h	支持	—	—	—	—	—	支持	108
使能 QPI	QPIEN	38h	支持	—	—	—	—	支持	—	108
使能 SPI	SPIEN	FFh	—	—	—	—	—	支持	支持	108
存储器读取操作										
读取	READ	03h	支持	—	—	—	—	支持	支持	40
快速读取	FAST_READ	0Bh	支持	—	—	—	—	支持	支持	108
双线输出 (快速) 读取	DOR	3Bh	—	支持	—	—	—	—	—	108
四线输出 (快速) 读取	QOR	6Bh	—	—	支持	—	—	—	—	108
双线 I/O (快速) 读取	DIOR	BBh	—	—	—	支持	—	—	—	108
四线 I/O (快速) 读取	QIOR	EBh	—	—	—	—	支持	—	—	108
存储器写入操作										
写入	WRITE	02h	支持	—	—	—	—	支持	支持	108
双线输入写入	DIW	A2h	—	支持	—	—	—	—	—	108
四输入写入操作	QIW	32h	—	—	支持	—	—	—	—	108
双线 I/O 写入	DIOW	A1h	—	—	—	支持	—	—	—	108
四线 I/O 写入	QIOW	D2h	—	—	—	—	支持	—	—	108
SR 命令										
软件复位使能	RSTEN	66h	支持	—	—	—	—	支持	支持	108
软件复位	RESET	99h	支持	—	—	—	—	支持	支持	108
进入休眠模式	HIBEN	BAh	支持	—	—	—	—	支持	支持	108
进入睡眠模式	SLEEP	B9h	支持	—	—	—	—	支持	支持	108
退出睡眠模式	EXSLP	ABh	支持	—	—	—	—	支持	支持	108
寄存器命令										
读取状态寄存器	RDSR	05h	支持	—	—	—	—	支持	支持	108
写入状态寄存器	WRSR	01h	支持	—	—	—	—	支持	支持	108
读取配置寄存器	RDCR	35h	支持	—	—	—	—	支持	支持	108
写入配置寄存器	WRCR	87h	支持	—	—	—	—	支持	—	108
读取 ID 寄存器	RDID	9Fh	支持	—	—	—	—	支持	支持	40
快速读取 ID 寄存器	FAST_RDID	9Eh	支持	—	—	—	—	支持	支持	108
写入序列号寄存器	WRSN	C2h	支持	—	—	—	—	支持	支持	108
读取序列号寄存器	RDSN	C3h	支持	—	—	—	—	支持	支持	40

表 2. 指令集 (续)

指令类别	指令名	操作码	SPI	双线输出	四线输出	双线 I/O	四线 I/O	DPI	QPI	最大频率 (MHz)
快速读取序列号寄存器	FAST_RDSN	C9h	支持	—	—	—	—	支持	支持	108
非易失性存储器的特定命令										
存储	STORE	8Ch	支持	—	—	—	—	支持	支持	108
回读	回读	8Dh	支持	—	—	—	—	支持	支持	108
自动存储使能	ASEN	8Eh	支持	—	—	—	—	支持	支持	108
自动存储禁用	ASDI	8Fh	支持	—	—	—	—	支持	支持	108
模式位										
模式位 (设置、复位)	—	Axh, !Axh	支持	—	—	—	—	支持	支持	—

根据它们的功能，SPI 指令被分为以下几种类型：

- 控制指令：
 - 写保护：WREN、WRDI 指令
 - I/O 模式：DPIEN、QPIEN、SPIEN
- 存储器读取指令：
 - 存储器访问：READ、FAST_READ、DOR、QOR、DIOR、QIOR
- 存储器写指令：
 - 存储器访问：WRITE、DIW、QIW、DIOW、QIOW
- 系统资源指令：
 - 软件复位：RSTEN、RESET
 - 功耗模式：HIBEN、SLEEP、EXSLP
- 寄存器指令：
 - 配置寄存器：RDCR、WRCR
 - 状态寄存器：RDSR、WRSR
 - 标识寄存器：RDID、FAST_RDID
 - 序列号寄存器：RDSN、WRSN、FAST_RDSN
- nvSRAM 特殊指令：
 - 存储：STORE
 - 回读：RECALL
 - 使能 / 禁用：ASEN、ASDI

注意：下面部分介绍的指令波形并不涉及上拉电阻对 \overline{WP} (I/O2)、NC (I/O3) 的影响以及中继器 / 总线保持电路对 SO 的影响。

注意：C5h、1Eh、C8h、CEh、CBh、CCh、CDh 均为赛普拉斯的保留指令操作码，用于修改器件配置。如果错误地输入了其中某个操作码，那么需要进行软件复位 (66h、99h) 才能恢复器件的正确配置。否则，器件不能正常工作。

状态寄存器

该器件具有一个状态寄存器；表 3 中列出了该状态寄存器及其位说明。状态寄存器中的位格式显示了对该位只能进行读操作（R）还是也能进行写操作（既可以写 / 也可以读操作（W/R））。唯一的例外是序列号锁定（SNL）。可使用 WRSN 指令多次写

入序列号，而 SNL 仍为 ‘0’。将该位设置为 ‘1’ 时，可防止对序列号进行任何修改。该位在出厂时被设置为 ‘0’，并且只能写入一次。该位被设置为 ‘1’ 后，便不能将其恢复到 ‘0’。

表 3. 状态寄存器格式和位定义

位	字段名称	功能	类型	R/W	默认状态	说明
7	SRWD	状态寄存器写入被禁用	NV	R/W	0	1 = 通过忽略 WRSR 命令， \overline{WP} 处于低电平时 SR 的锁定状态 0 = 不受保护，即使 \overline{WP} 处于低电平
6	SNL	序列号锁定	OTP	R/W	0	锁定序列号
5	TBPROT	配置模块的起始地址	NV	R/W	0	1 = BP 从底部（低地址）开始 0 = BP 从顶部（高地址）开始
4	BP2	模块保护	NV	R/W	0	防止对模块的所选范围进行写入、编程或擦除操作
3	BP1		NV	R/W	0	
2	BP0		NV	R/W	0	
1	WEL	写使能锁存	V	R	0	1 = 器件接收写入寄存器（WRSR）、写入、编程或擦除命令 0 = 器件忽略写入寄存器（WRSR）、写入、编程或擦除命令 该位不受 WRSR 的影响，但受 WREN 和 WRDI 命令的影响
0	WIP	正在进行	V	R	0	1 = 器件处于繁忙状态，正在执行写入寄存器（WRSR）、编程、擦除或其他操作 0 = 就绪的器件处于待机状态，并且可以接收命令

状态寄存器写操作禁用（SRWD）SR[7]

当该位被设置为 ‘1’，并且 \overline{WP} 输入被驱动为低电平时，该器件将处于硬件保护模式。在这种模式下，所有 SRWD 位（非 WEL）都成为只读位，并不会再接收写寄存器（WRSR）指令来进行操作。如果 \overline{WP} 为高电平，那么可以通过 WRSR 指令修改各个 SRWD 位。如果 SRWD 为 ‘0’， \overline{WP} 没有任何作用，并且 SRWD 位可以通过 WRSR 命令得到修改。

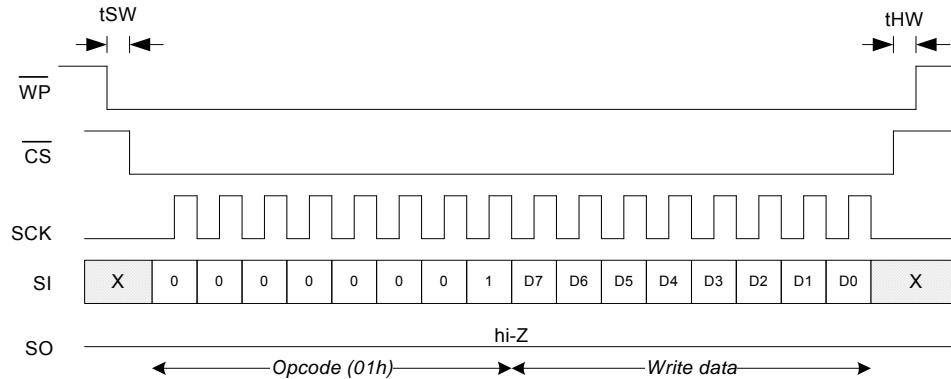
注意：设置配置寄存器中的四线位 CR[1] 后， \overline{WP} 将被内部设置为逻辑 ‘0’。如果 SRWD 被置为逻辑 ‘1’，便不能改变保护状态，直到四线位 CR[1] 复位为逻辑 ‘0’ 为止。

表 4. SRWD、 \overline{WP} 、WEL 和硬件保护

SRWD	\overline{WP}	WEL	受保护的模块	无保护的模块	状态寄存器（WEL 除外）
X	X	0	受保护	受保护	受保护
0	X	1	受保护	可写	可写
1	低电平	1	受保护	可写	受保护
1	高电平	1	受保护	可写	可写

注意：在执行写入状态寄存器指令期间，如果 \overline{CS} 被置为低电平， \overline{WP} 信号将被采样，以确定硬件保护是否被使能。图 8 显示了时序波形。

图 8. 与 \overline{CS} 相关的 \overline{WP} 时序



序列号锁定位 (SNL) SR[6]

将该位设置为 ‘1’ 时，可防止对序列号进行任何修改。该位在出厂时被设置为 ‘0’，并且只能对其进行一次写操作。该位被设置为 ‘1’ 后，就不能将其恢复到 ‘0’。

顶部或底部保护 (TBPROT) CR[5]

该位定义了块保护位 (BP2、BP1 和 BP0) 的操作。在设备的系统制造过程中进行初始化配置时必须选择 TBPROT 所需的状态。

块保护 (BP2、BP1、BP0) 位 SR[4:2]

这些位定义了软件保护的存储器阵列域，以防止执行写入指令。BP 位是非易失性的。当一个或多个 BP 位被设置为 ‘1’ 时，便不能对相关存储区进行写入、编程和擦除操作。

可通过各模块保护位 (状态寄存器位 BP2、BP1 和 BP0) 与 TBPROT 位的组合来保护存储器阵列的地址范围。该范围大小由 BP 位的值确定，并且该范围的上限或下限起始点是由状态寄存器的 TBPROT 位决定的。

表 5. 保护部分的上层阵列起点 (TBPROT = 0)

状态寄存器内容			存储器阵列的受保护部分	地址范围
BP2	BP1	BP0		
0	0	0	无	无
0	0	1	上层的第 64 个扇区	0x1F800 - 0x1FFFF
0	1	0	上层的第 32 个扇区	0x1F000 - 0x1FFFF
0	1	1	上层的第 16 个扇区	0x1E000 - 0x1FFFF
1	0	0	上层的第 8 个扇区	0x1C000 - 0x1FFFF
1	0	1	上层的第 4 个扇区	0x18000 - 0x1FFFF
1	1	0	上半部分	0x10000 - 0x1FFFF
1	1	1	所有扇区	0x00000 - 0x1FFFF

表 6. 保护部分的下层阵列起点 (TBPROT = 1)

状态寄存器内容			存储器阵列的受保护部分	地址范围
BP2	BP1	BP0		
0	0	0	无	无
0	0	1	下层的第 64 个扇区	0x00000 - 0x007FFF
0	1	0	下层的第 32 个扇区	0x00000 - 0x00FFFF
0	1	1	下层的第 16 个扇区	0x00000 - 0x01FFFF
1	0	0	下层的第 8 个扇区	0x00000 - 0x03FFFF
1	0	1	下层的第 4 个扇区	0x00000 - 0x07FFFF
1	1	0	下半部分	0x00000 - 0x0FFFFF
1	1	1	所有扇区	0x00000 - 0x1FFFFF

写使能 (WEL) SR[1]

必须将 WEL 位设置为 ‘1’，这样才能使能编程、写入、擦除操作，并能防止对存储器或寄存器值进行意外更改。执行写使能 (WREN) 命令会将写使能锁存位设置为 ‘1’，这样便能执行后面所有写命令。通过写禁用 (WRDI) 命令，将写使能锁存位设置为 ‘0’，这样可以禁止所有写命令。对寄存器成功完成写操作、存储、回读、编程或擦除操作时，将清除 WEL 位为 ‘0’——请注意，对存储器宏进行写操作后，它不会被清除。断电 / 加电序列、硬件复位或软件复位发生后，写使能锁存位将被设置为 ‘0’。WRSR 命令不会影响该位的状态。

注意：WEL 位不会对自动存储、上电回读和硬件存储（由 $\overline{\text{HSB}}$ 启动）等操作产生任何影响。

表 7. 需要设置 WEL 位的指令

指令说明	指令名	操作码
存储器写入操作		
写入	WRITE	02h
双线输入写入	DIW	A2h
四线输入写入	QIW	32h
双线 I/O 写入	DIOW	A1h
四线 I/O 写入	QIOW	D2h
寄存器指令		
写状态寄存器	WRSR	01h
写入配置寄存器	WRCR	87h
写入序列号寄存器	WRSN	C2h
非易失性存储器的特定命令		
存储	STORE	8Ch
回读	RECALL	8Dh
自动存储使能	ASEN	8Eh
自动存储禁用	ASDI	8Fh

表 8. 配置寄存器

位	字段名称	功能	类型	R/W	默认状态	说明
7	RFU	预留	—	R/W	0	保留供将来使用
6	RFU	预留	—	R/W	1	保留供将来使用
5	RFU	预留	—	—	0	保留供将来使用
4	RFU	预留	—	—	0	保留供将来使用
3	RFU	预留	—	—	0	保留供将来使用
2	RFU	预留	—	—	0	保留供将来使用
1	QUAD	使器件进入四线模式	NV	R/W	0	1 = 四线；0 = 双线或串行
0	RFU	预留	—	—	0	保留供将来使用

正在进行的操作 (WIP) SR[0]

表示器件是否正在执行编程、写入、擦除操作或任何其他操作，在此期间进行的新操作指令将被忽略。当该位被设置为 ‘1’ 时，器件会执行一个后台操作。WIP 位为 ‘1’ 时，只能接收读状态 (RDSR) 命令。将 WIP 位清除为 ‘0’ 时，则没有任何操作正在进行。这是一个只读位。

只有执行一个存储操作后，才会将写到 SR 中的所有值保存到非易失性存储器中。如果自动存储被禁用，则必须通过软件存储操作来保存对状态寄存器所做的修改。

如果对 SRAM 进行写入，那么硬件存储操作会只将状态寄存器的值保存到非易失性存储器内。

配置寄存器

QPI nvSRAM 具有一个配置寄存器；表 8 中列出了该状态寄存器及位说明。配置寄存器中的位格式显示了只能对该位进行读操作 (R) 还是又能进行写操作 (W/R)。配置寄存器控制某些接口功能。

四线数据宽度 (QUAD) CR[1]

该位被设置为 ‘1’ 时，会将数据宽度改为 4 位，即： \overline{WP} 成为 I/O2，而 NC (I/O3) 成为 I/O3。 \overline{WP} 输入不被监控，因此它会正常运行，并且被内部设置为激活状态。可以正常执行串行、双线输出和双线 I/O 读取命令，但在使用不同数据路径宽度切换各个命令时便不用为这些命令驱动 \overline{WP} 输入。当使用四线输出读取、四线 I/O 读取、四线输入写入、四线 I/O 写入，以及所有四线 SPI 命令时，必须将四线位设置为 1。四线位是非易失性的。

注意：需要向配置寄存器内写入 0x42，才能设置四线位。同样，向配置寄存器内写入 0x40，可实现重置四线位。数据的所有其他组合会改变器件的配置情况，并使其无法工作。

注意：设置配置寄存器中的四线位 CR[1] 后， \overline{WP} 内部默认被设置为逻辑 ‘0’。

注意：写入配置寄存器的有效值将在存储操作完成后被保存到非易失性存储器内。如果自动存储被禁用，则必须通过软件存储操作来保存对配置寄存器所做的修改。如果对 SRAM 进行写操作，那么硬件存储操作会只将配置寄存器的值保存到非易失性存储器内。

SPI 控制指令

写入禁用（WRDI）指令

该写禁用指令通过将 WEL 位清除为 ‘0’ 来禁用所有写操作，从而防止发生意外写入。CS 的下降沿和 WRDI 指令的操作码完成后，将发出该指令。WEL 位会在 CS 的上升沿上被清除。

图 9. SPI 模式下的 WRDI 指令

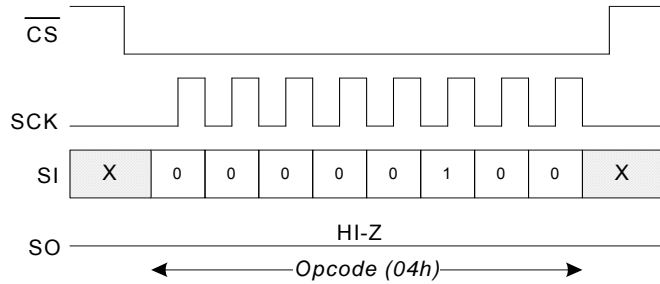


图 10. DPI 模式下的 WRDI 指令

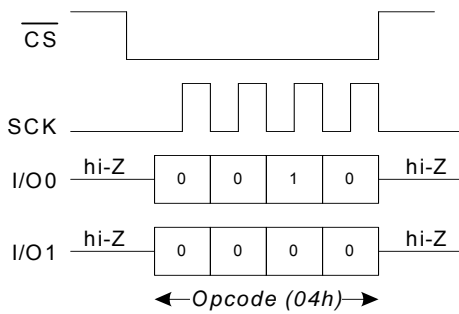
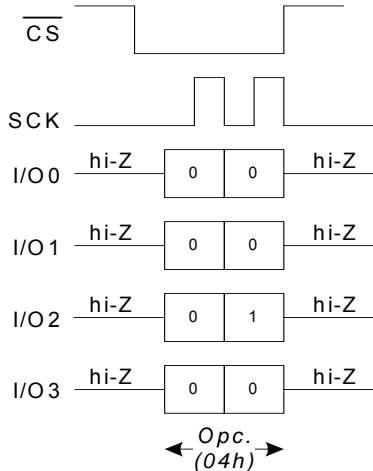


图 11. QPI 模式下的 WRDI 指令



写入使能（WREN）指令

加电时，器件保持写入禁用状态。因此，写指令或 nvSRAM 特殊指令的前面必须是一条写入使能指令。如果器件并非处于写入使能状态（WEL = ‘0’），那么它会忽略写指令，并在 CS 处于高电平状态时返回到待机状态。CS 下降沿过后会发出该指令，然后该指令会将状态寄存器的 WEL 位置为 ‘1’。加电时，WEL 位默认为 ‘0’。

注意：成功写入寄存器内或结束 STORE、RECALL、ASEN 或 ASDI 操作时，WEL 位被清除为 ‘0’。对存储器宏进行写操作后，不会清除它。

图 12. SPI 模式中的 WREN 指令

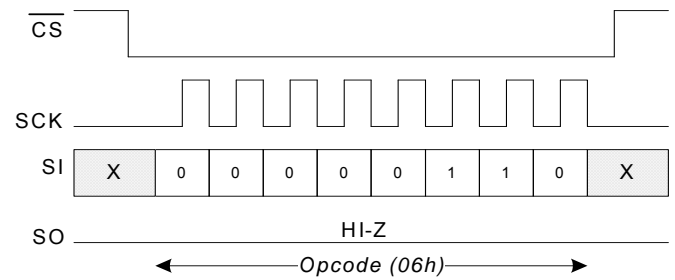


图 13. DPI 模式中的 WREN 指令

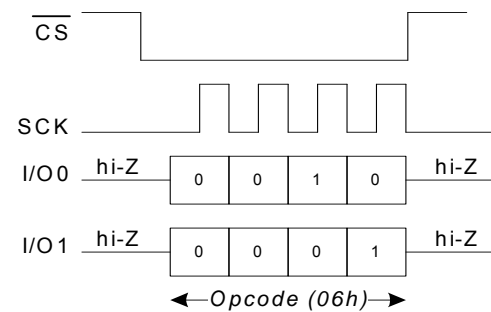
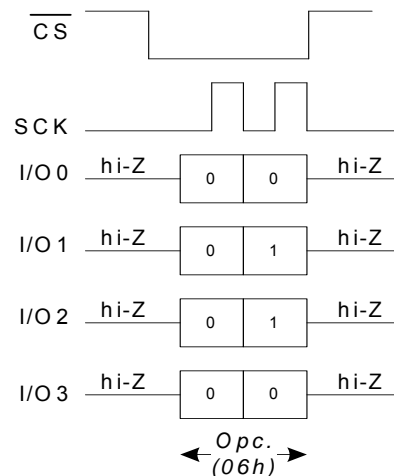


图 14. QPI 模式下的 WREN 指令



使能 DPI (DPIEN) 指令

DPIEN 使能双线 I/O 模式, 在该模式下, 通过 I/O0 和 I/O1 传送操作码、地址、各模式位和数据。

图 15. SPI 模式下的使能双线 I/O 指令

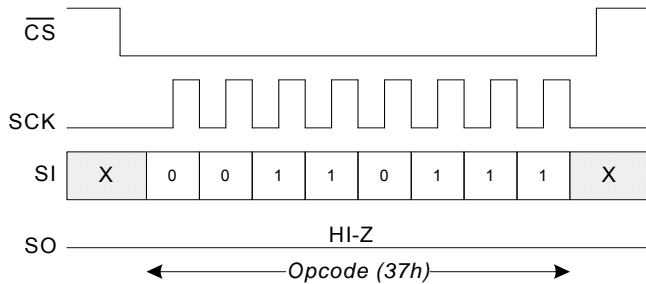
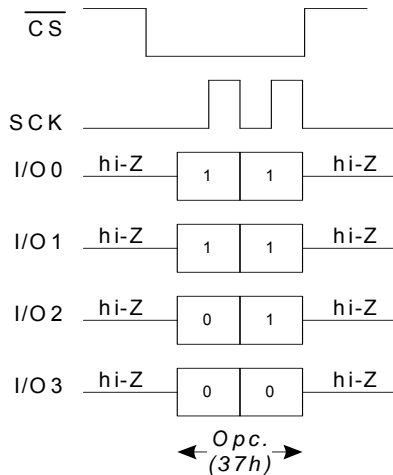


图 16. QPI 模式下的使能双线 I/O 指令



使能 QPI (QPIEN) 指令

QPIEN 使能 QPI 模式, 在该模式下, 操作码、地址、虚拟 I 模式位和数据会通过 IO0、IO1、IO2 和 IO3 被传送。QPIEN 指令不会设置配置寄存器中的四线位 CR[1]。因此, 用来设置四线位 CR[1] 的 WRCR 指令必须执行 QPIEN 指令。

注意: 禁用 QPI 模式, 不会重置四线位 CR[1]。

图 17. SPI 模式下的使能四线 I/O 指令

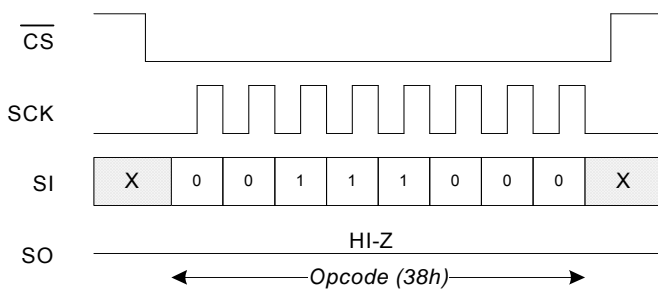
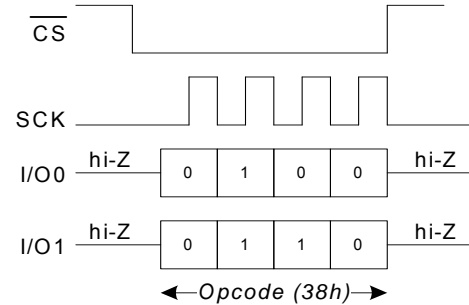


图 18. DPI 模式下的使能四线 I/O



使能 SPI (SPIEN) 指令

SPIEN 会禁用双线 I/O 或四线 I/O 模式, 并使器件返回到 SPI 模式。SPIEN 指令不会对配置寄存器中的四线位 CR[1] 进行复位。

图 19. DPI 模式下的使能 SPI 指令

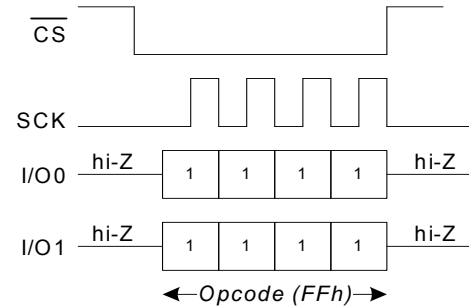
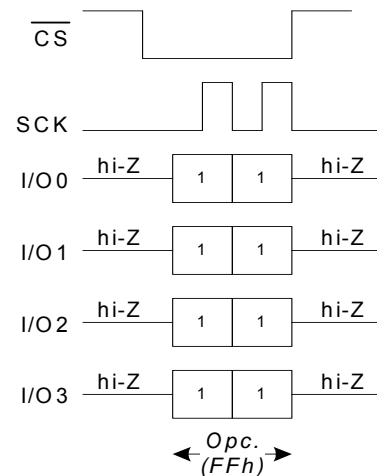


图 20. QPI 模式下的使能 SPI 指令



SPI 存储器读取指令

通过读指令可以访问存储器阵列。在进行存储或回调周期过程中无法使用这些指令。可通过状态寄存器的 WIP 位和 HSB 引脚来表示正在进行的存储周期。

读指令

如果在 SI 引脚上提供了读指令操作码，那么器件将执行读取操作。它将在 SPI 模式下读取 SO 引脚上的数据，在双线 I/O 模式下读取 I/O1 和 I/O0 引脚上的数据，并在四线 I/O 模式下读取 I/O3、I/O2、I/O1 和 I/O0 引脚上的数据。将 CS 线设置为低电平并选取器件后，将发送读取操作码，后面是三个字节的数据。该器件还包含一个用于 1 Mb 配置的 17 位地址空间。

最高有效地址字节在位 0 中包含 A16，无需关注其他位。地址位 A15 至 A0 在以下两个地址字节中发送。发送最后的地址位后，会在 SCK 下降沿到来时移出指定地址上的数据 (D7–D0) (以 D7 开始)。如果将 CS 置于低电平，则可以进行读操作。

输出每一个数据字节后，器件自动递增至下一个更高地址。当到达最后一个数据存储器地址 (0x1FFFF) 时，地址将翻转为 0x00000，然后器件继续进行读操作。输出数据时可以随时将 CS 置于高电平，以停止读指令。

注意：读指令的最高工作频率为 40 MHz。在双线和四线 I/O 模式下，地址字节传输后需要经过一个虚拟周期。这样便允许器件预提取第一个字节并开始执行流水线操作。

READ 指令

读指令可以在 SPI、双线 I/O (DPI) 或四线 I/O (QPI) 模式下使用。在 SPI 模式下，操作码、地址和字节都通过 SI 引脚被传送，并且每个时钟周期发生一位。在最后地址周期的 SCK 的下降沿上，特定地址上的数据 (D7–D0) 将被移出，每个时钟周期一位 (以 D7 开始)。

在 DPI 模式下，操作码和地址字节通过 I/O1 和 I/O0 引脚传送，每个时钟周期两位。在最后地址周期的 SCK 的下降沿上，特定地址上的数据 (D7–D0) 将被移出，每个时钟周期两位 (在 I/O1 上以 D7 开始，在 I/O0 上以 D6 开始)。在 QPI 模式下，操作码和地址字节通过 I/O3、I/O2、I/O1 和 I/O0 引脚传送，每个时钟周期四位。在 SCK 最后一个地址周期的下降沿上，特定地址上的数据 (D7–D0) 将被移出，并且每个时钟周期移出四位 (在 I/O3 上以 D7 开始，在 I/O2 上以 D6 开始，在 I/O1 上以 D5 开始，在 I/O0 上以 D4 开始)。

图 21. SPI 模式下的读指令

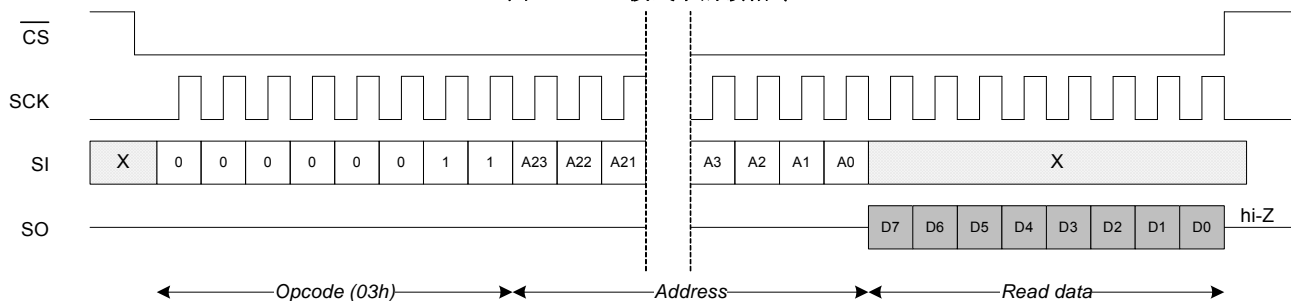


图 22. SPI 模式下的突发模式读指令

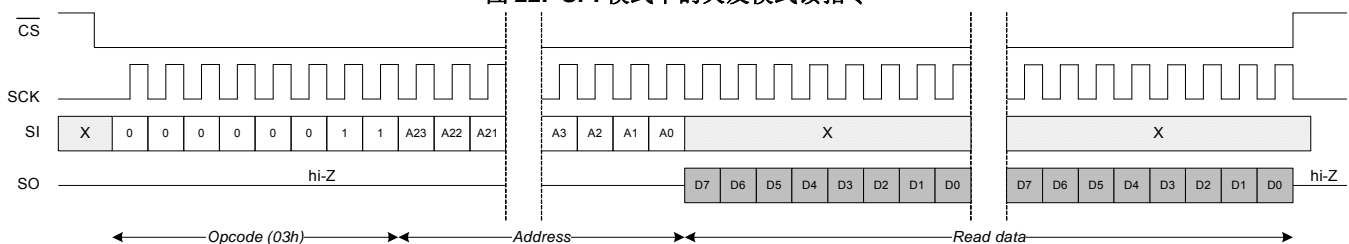


图 23. DPI 模式下的读指令

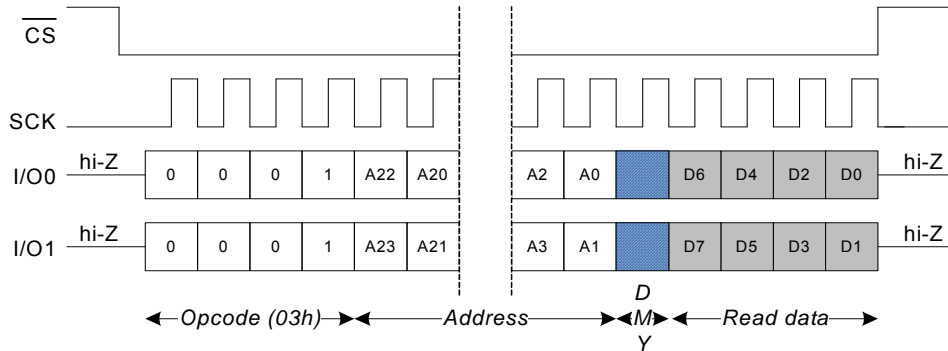
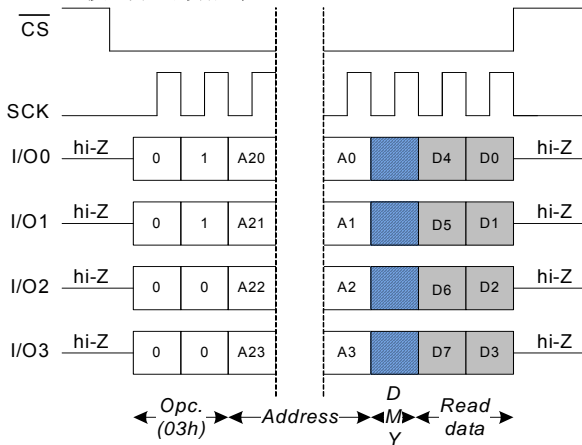


图 24. QPI 模式下的读指令



注意: 在 QPI 模式下执行读指令前，必须将四线位 CR[1] 设置为逻辑 ‘1’。

快速读取指令

利用快速读取指令，能够以高达 108 MHz（最大值）的 SPI 频率读取存储器的数据。与通用的读指令相似，该指令的所有 I/O 配置中都有一个等待状态：发送地址后到发送第一个数据前，必须发送模式字节。这样便允许器件预提取第一个字节并开始执行流水线操作。主机系统首先要通过将 CS 设置为低电平来选中器件，然后输入三个地址字节和一个模式字节。在下一个 SCK 下降沿上，特定地址上的数据分别在 SPI 模式的 SO 引脚、双线 I/O

模式的 I/O1 和 I/O0 引脚以及四线 I/O 模式的 I/O3、I/O2、I/O1 和 I/O0 引脚上被移出。指定的第一个字节可处于任何位置。输出每一个数据字节后，器件自动递增至下一个更高地址。因此，只要通过单一快速读取指令就可读取整个存储器阵列。当到达存储器阵列中的最高地址时，地址计数器将翻转为起始地址 0x00000，并允许读取序列无限期继续。在数据输出时可以随时将 CS 置于高电平以终止快速读取指令。

注意: 这些指令可以在最高达 108 MHz 的 SPI 工作频率下运行。

FAST_READ 指令

FAST_READ 指令可以在 SPI、双线 I/O (DPI) 或四线 I/O (QPI) 模式下使用。在 SPI 模式下，操作码、地址和模式字节都能通过 SI 引脚被传输，每个时钟周期传送一位。在最后模式字节周期的 SCK 的下降沿上，特定地址上的数据 (D7-D0) 在 SO 引脚上被移出，每个时钟周期传送一位 (以 D7 开始)。在 DPI 模式下，操作码、地址和模式字节通过 I/O1 和 I/O 引脚被传送，每个时钟周期传送两位。在最后模式周期的下降沿上，特定地址上的数据 (D7-D0) 将被移出，每个时钟周期两位 (D7 通过 I/O1 移出，D6 通过 I/O0 移出)。在 QPI 模式下，操作码和地址字节通过 I/O3、I/O2、I/O1 和 I/O0 引脚被传送，每个时钟周期传送四位。在最终模式周期的 SCK 的下降沿上，特定地址上的数据 (D7-D0) 将被移出，每个时钟周期移出四位 (D7 通过 I/O3 被移出，D6 通过 I/O2 被移出，D5 通过 I/O1 被移出，另外 D4 通过 I/O0 被移出)。

图 25. SPI 模式下的 FAST_READ 指令

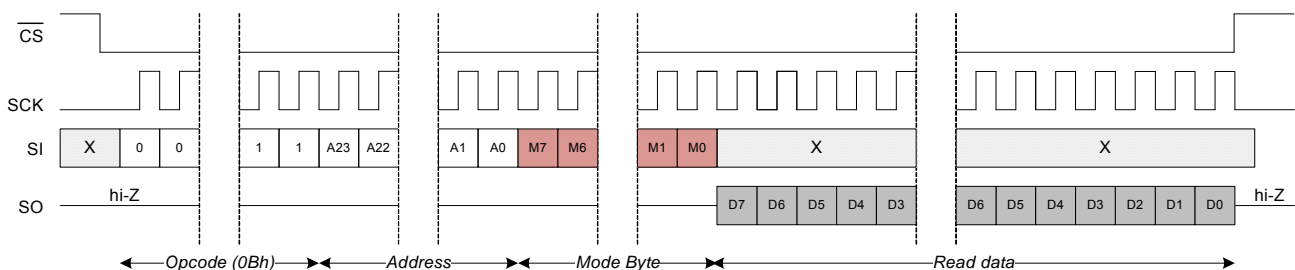


图 26. DPI 模式下的 FAST_READ 指令

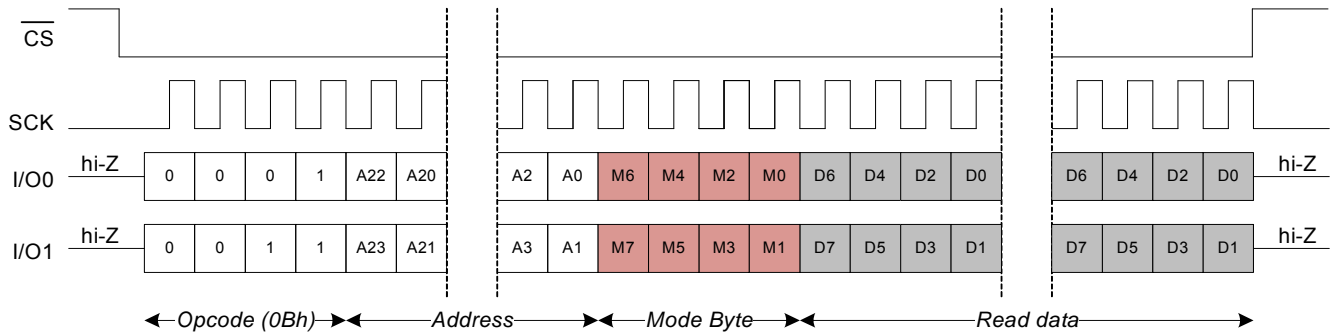
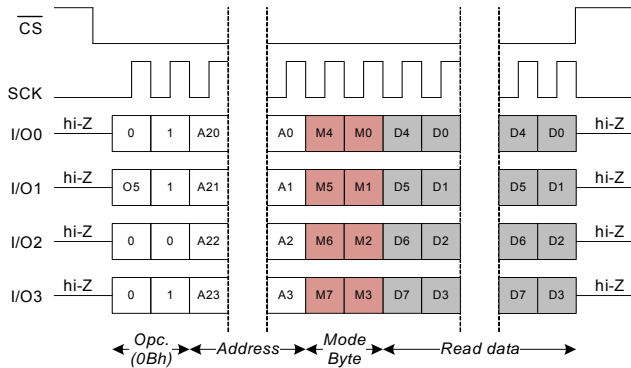


图 27. QPI 模式下的 FAST_READ 指令



DOR 指令

DOR 指令被用于双线数据模式，它是 SPI 扩展读命令的一部分。在双线数据模式下，操作码、地址和模式字节均通过 SI 引脚得到传送，每个时钟周期传送一位。在最后一个模式周期的 SCK 的下降沿上，这些引脚被重新配置：SO 作为 I/O1，SI 作为 I/O0。特定地址上的数据（D7-D0）在 I/O1 和 I/O0 引脚上被移出，每个时钟周期两位（D7 通过 I/O1 移出，D6 通过 I/O0 移出）。

QOR 指令

QOR 指令在四线数据模式下被使用，它是 SPI 扩展读命令的一部分。在四线数据模式下，操作码、地址和模式字节通过 SI 引脚被传送，每个时钟周期传送一位。在最后一个模式周期的 SCK 下降沿上，这些引脚被重新配置：NC 变成 I/O3，WP 变成 I/O2，SO 变成 I/O1 和 SI 变成 I/O0。特定地址上的数据（D7-D0）在 I/O3、I/O2、I/O1 和 I/O0 引脚上被移出，每个时钟周期四位（D7 通过 I/O3 移出，D6 通过 I/O2 移出，D5 通过 I/O1 移出，并 D4 通过 I/O0 移出）。

注意：执行 QOR 指令前，必须将四线位 CR[1] 设置为逻辑 ‘1’。

图 28. DOR 指令

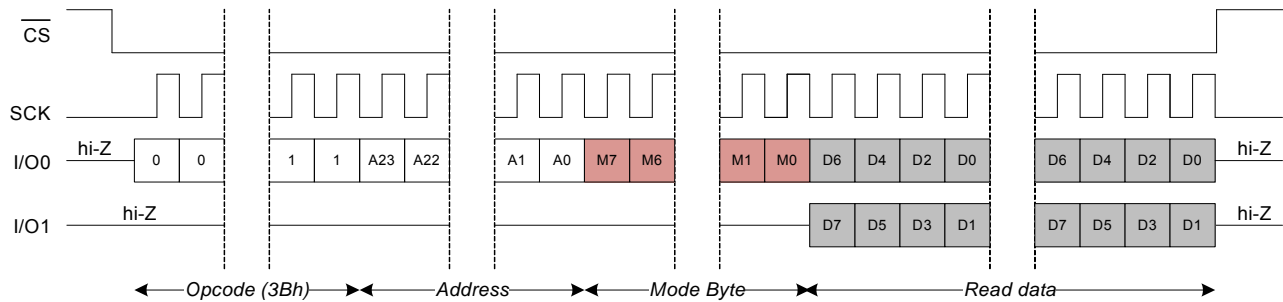
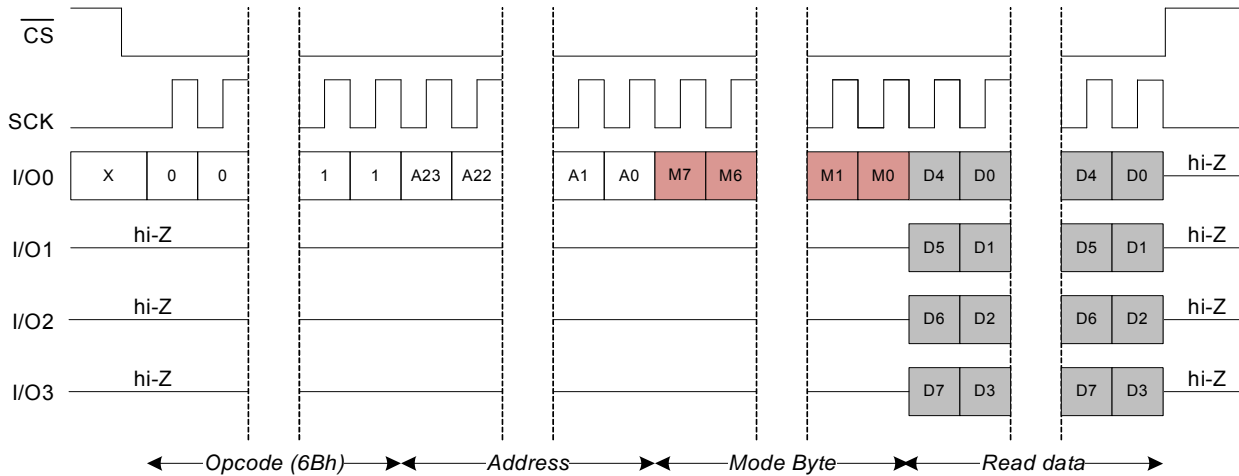
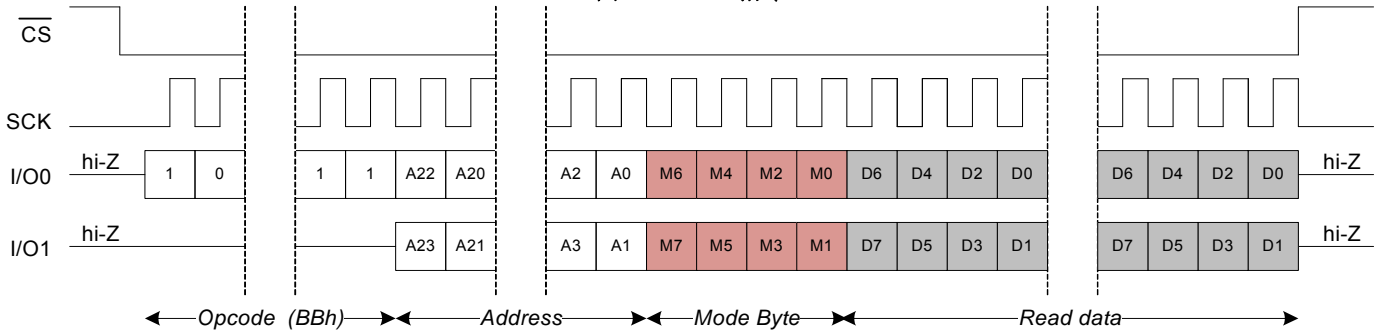


图 29. QOR 指令

DIOR 指令

DIOR 指令在双线地址 / 数据模式下被使用，它是 SPI 扩展读命令的一部分。在双线地址 / 数据模式下，操作码通过 SI 引脚被传送，每个时钟周期传送一位。在传输操作码的最后位后，这些引脚被重新配置：SO 变成 I/O1 和 SI 变成 I/O0。然后，该地址通过 I/O1 和 I/O0 引脚传送到器件，每个时钟周期传送两位（在

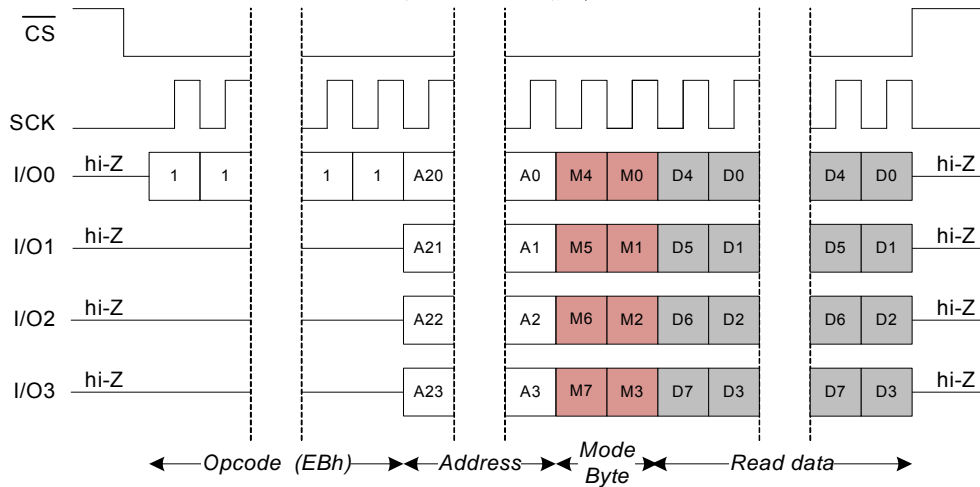
I/O1 上以 A23 开始，在 I/O0 上以 A22 开始），直到输入三字节地址为止。特定地址上的数据（D7-D0）在 I/O1 和 I/O0 引脚上被移出，每个时钟周期两位（D7 通过 I/O1 移出，并 D6 通过 I/O0 移出）。

图 30. DIOR 指令

QIOR 指令

QIOR 指令在四地址 / 数据模式下被使用，它是 SPI 扩展读命令的一部分。在四地址 / 数据模式下，操作码通过 SI 引脚被传送，每个时钟周期传送一位。在操作码的最后位传输后，这些引脚被重新配置：NC 作为 I/O3，WP 作为 I/O2，SO 作为 I/O1 和 SI 作为 I/O0。然后，该地址通过 I/O3、I/O2、I/O1 和 I/O0 引脚传送到器件，每个时钟周期发送四位（A23 通过 I/O3 传输，A22 通过 I/O2 传输，并 A20 通过 I/O0 传输），直至到达地址的输入 3 字节为止。特定地址上的数据（D7-D0）在 I/O3、I/O2、I/O1 和 I/O0 引脚上被移出，每个时钟周期传送四位（D7、D6、D5 和 D4 分别通过 I/O3、I/O2、I/O1 和 I/O0 移出）。

注意：在执行 QIOR 指令前，四线位 CR[1] 必须为逻辑 ‘1’。

图 31. QIOR 指令



写指令

将写指令操作码和写数据提供给 SPI 模式的 SI 引脚、双线 I/O 模式的 I/O1 和 I/O0 引脚或四线 I/O 模式的 I/O3、I/O2、I/O1 和 I/O0 引脚时，器件将进行写操作。如果器件处于写入禁用状态，首先要通过 WREN 指令使能写入功能，这样才能进行写操作。使能写入功能（WEL = ‘1’）后，将在 CS 的下降沿后发出 WRITE 指令。nvSRAM 允许进行突发写入操作，这样无需发出新的写入指令，仍能写入连续的地址。如果只写入一个字节，发送 D0（数据的 LSB）后必须将 CS 引脚置于高电平。如果要写入多个字节，则必须使 CS 引脚保持低电平状态，地址会自动被递增。输入引脚上的数据字节被写入到连续地址内。当达到最后数据存储地址（0x1FFFF）时，地址将翻转为 0x00000，并且该器件将继续进行写操作。

注意：对存储器阵列进行写序列后，状态寄存器中的 WEL 位不会被复位到 ‘0’。

注意：当突发写入达到受保护的模块地址时，它仍在受保护空间内保持地址递增，但是不将任何数据写入到受保护的存储器内。如果地址翻转并采用突发写入到不受保护的的空间，则可恢复写入。如果在写入受保护的模块中使能突发写入，则以同一操作执行。

注意：这些指令的最高工作频率可达 108 MHz。

将 CS 引脚置于低电平并选择器件后，进行传输写操作码，然后再传输三字节地址。该器件还包含一个用于 1 Mb 配置的 17 位地址空间。最高有效地址字节在位 0 中包含 A16，无需关注其他位。地址位 A15 至 A0 在随后的两个地址字节中发送。传送最后地址位后，将立即通过输入线传送数据（D7-D0）。可以在 SPI、DPI 或 QPI 模式下使用该命令。

WRITE（写入）指令

可以在 SPI、DPI 或 QPI 模式下使用写指令。在 SPI 模式下，操作码、地址字节和数据字节通过 SI 引脚传送，每个时钟周期传输一位（以 D7 开始）。在 DPI 模式下，操作码、地址字节和数据字节通过 I/O1 和 I/O 引脚传送，每个时钟周期两位（D7 和 D6 分别通过 I/O1、I/O0 传送）。在 QPI 模式下，操作码、地址字节和数据字节通过 I/O3、I/O2、I/O1 和 I/O0 引脚传送，每个时钟周期四位（D7、D6、D5 和 D4 分别通过 I/O3、I/O2、I/O1 和 I/O0 传送）。

图 32. SPI 模式下的写指令

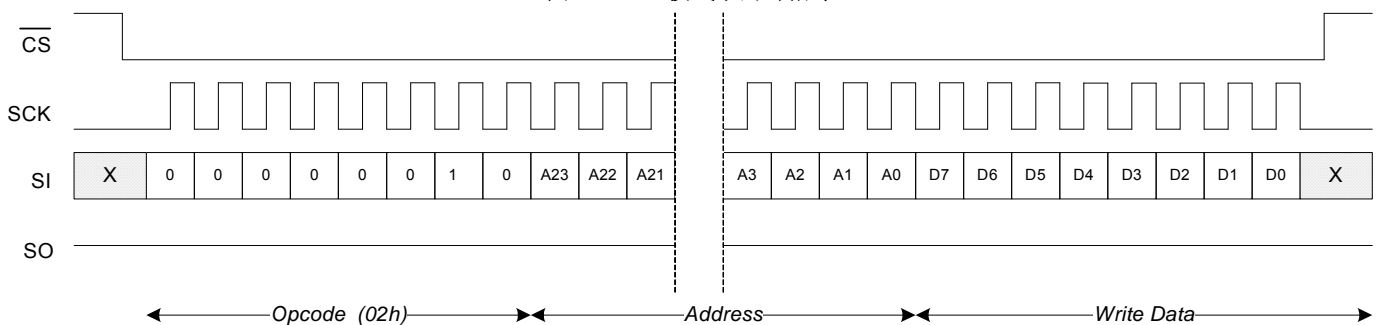


图 33. SPI 模式下的突发写指令

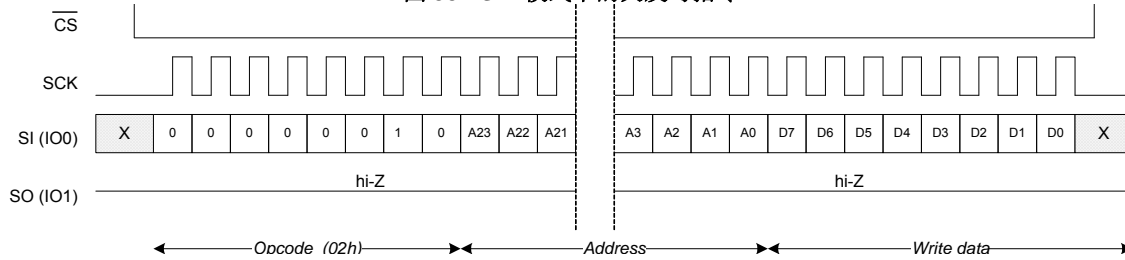


图 34. DPI 模式下的写指令

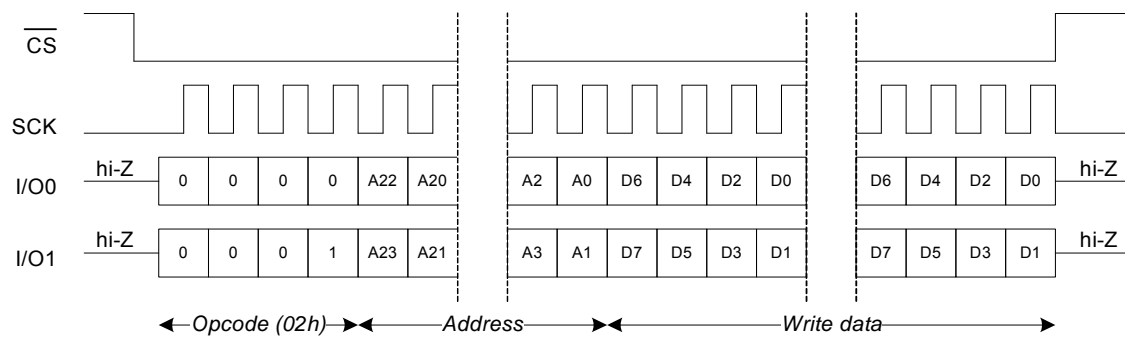
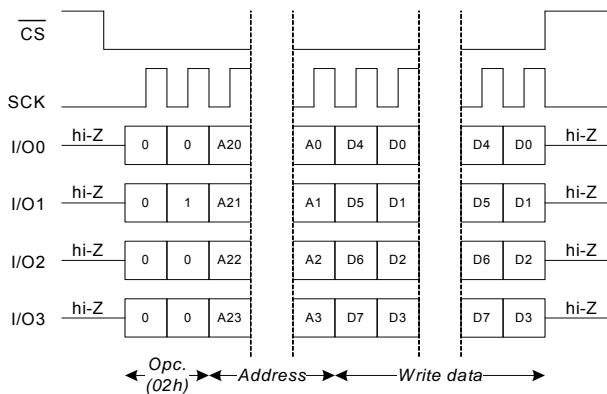


图 35. QPI 模式下的写指令

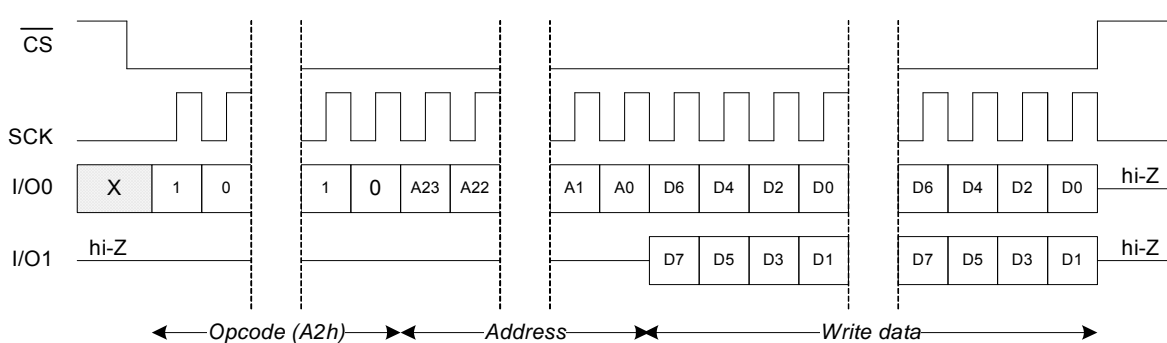


注意: 在 QPI 模式下执行写指令前, 必须将四线位 CR[1] 设置为逻辑 '1'。

DIW 指令

DIW 指令在双线数据模式下被使用, 它是 SPI 扩展写命令的一部分。在双数据模式下, 操作码和地址字节通过 SI 引脚传送, 每个时钟周期一位。传送最后地址位后, 将立即重新配置各引脚: SO 变成 I/O1 和 SI 变成 I/O0。另外, 数据 (D7-D0) 被传送到 I/O1 和 I/O0 引脚, 每个时钟周期两位 (D7 和 D6 分别通过 I/O1 和 I/O0 传送)。

图 36. DIW 指令



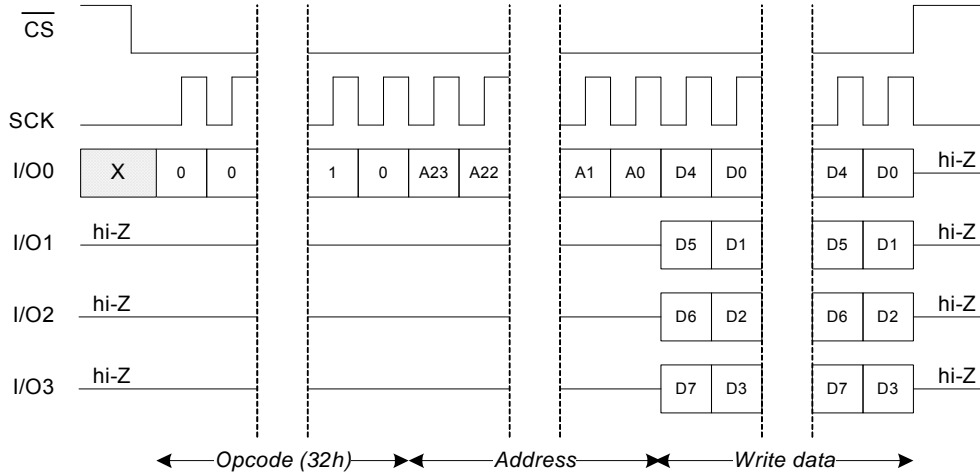
QIW 指令

QIW 指令在四线数据模式下被使用，它是 SPI 扩展写命令的一部分。在四线数据模式下，操作码和地址字节通过 SI 引脚被传送，每个时钟周期传送一位。传送最后地址位后，将立即重新配置各引脚：NC 变成 I/O3，WP 变成 I/O2，SO 变成 I/O1 和 SI 变成

I/O0。另外，数据（D7-D0）分别通过 I/O3、I/O2、I/O1 和 I/O0 引脚进行发送，每个时钟周期四位（D7、D6、D5 和 D4 分别通过 I/O3、I/O2、I/O1 和 I/O0 传送）。

注意：在执行 QIW 指令前，四线位 CR[1] 必须为逻辑 ‘1’。

图 37. QIW 指令

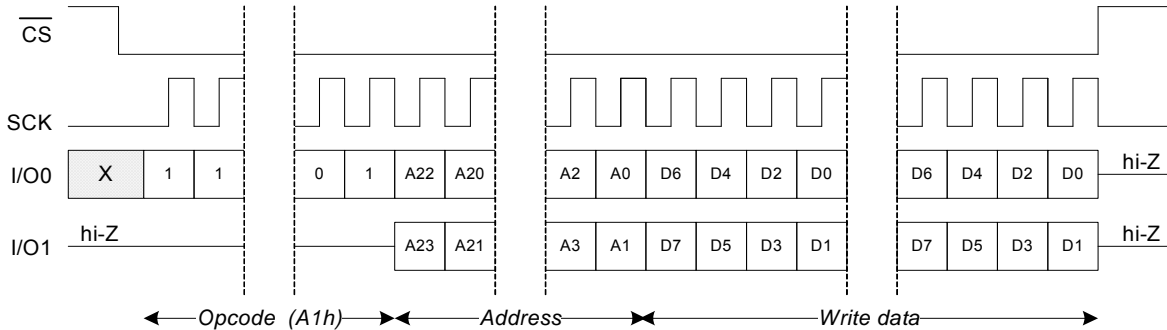


DIOW 指令

DIOW 指令在双线地址 / 数据模式下被使用，它是 SPI 扩展写命令的一部分。在双线地址 / 数据模式下，操作码通过 SI 引脚被传送，每个时钟周期传送一位。传送最后操作码位后，将立即重新配置各引脚：SO 变成 I/O1 和 SI 变成 I/O0。另外，地址通过 I/O1

和 I/O0 引脚传送到器件，每个时钟周期两位（首先 A23 通过 I/O1 传送，A22 通过 I/O0 传送），直到输入三字节地址为止。传送最后地址位后，会通过 I/O1 和 I/O0 将数据（D7-D0）传送到器件，每个时钟周期两位（首先 D7 通过 I/O1 传输，D6 通过 I/O0 传输）。

图 38. DIOW 指令

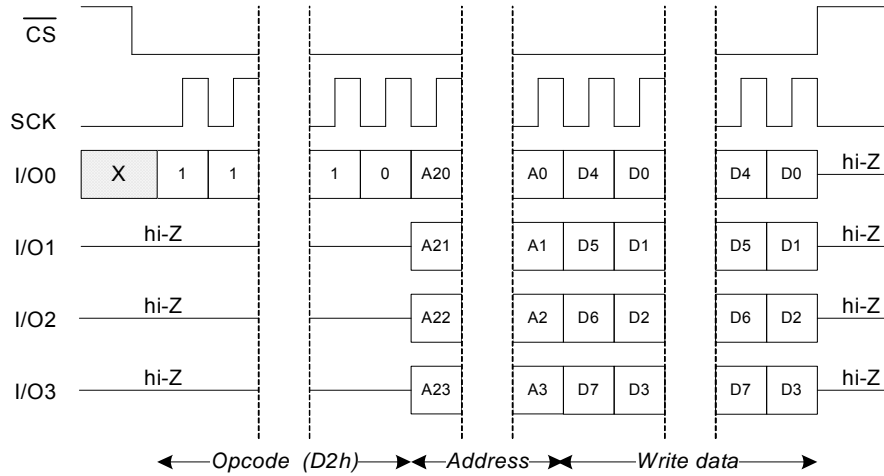


QIOW 指令

QIOW 指令在四线地址 / 数据模式下被使用，它是 SPI 扩展写命令的一部分。在四线地址 / 数据模式下，操作码通过 SI 引脚被传送，并且每个时钟周期传送一位。传送最后操作码位后，将立即重新配置各引脚：NC 变成 I/O3，WP 变成 I/O2，SO 变成 I/O1 和 SI 变成 I/O0。另外，地址通过 I/O3、I/O2、I/O1 和 I/O0 引脚传送到器件，每个时钟周期四位（开始于 A23 通过 I/O3 传送，A22 通过 I/O2 传送，A21 通过 I/O1 传输，并 A20 通过 I/O0 传送），直到输入三字节地址为止。传送最后地址位后，会通过 I/O3、I/O2、I/O1 和 I/O0 将数据（D7-D0）传送给器件，每个时钟周期传送四位（D7、D6、D5 和 D4 分别通过 I/O3、I/O2、I/O1 和 I/O0 得到传送）。

注意：在执行 QIOW 指令前，四线位 CR[1] 必须是逻辑 ‘1’。

图 39. QIOW 指令



Execute In Place (芯片内执行 — XIP)

Execute-in-place (XIP) 模式允许存储器进行一系列开始于不同地址的读操作，而不需要加载每个读操作的命令代码。这样会节省随机访问时间并且不需要将代码映像到 RAM 以快速运行。XIP 模式支持的读命令是 FAST_READ (在 SPI、DPI 和 QPI 模式下)、DOR、DIOR、QOR 和 QIOR。

通过输入 Mode (模式) 位可以设置或复位这些指令的 XIP 模式。Mode 位的高位半字节 (位 7-4) 通过添加或清除第一字节指令代码控制上述读指令的长度。Mode 位的低位半字节 (位 3-0)

无需关注，它们可以为高阻抗。微控制器通常使用这些字节反转总线以读取数据。如果 Mode 位等于 Exh，器件将被设置为保持读取模式，并且可以输入下个地址而不需要操作码，如下图所示；这样将清除操作码序列的某些周期。如果 Mode 位等于 Fxh，XIP 模式将被复位，而且在当前数据传输结束后，器件将等待操作码。

在执行这些命令期间，可以在任何序列中随时进入或退出 XIP。如果需要执行不受 XIP 支持的另一个操作 (如写操作)，那么在输入新的命令代码以实现所需要的操作之前需要退出 XIP。

图 40. 在 SPI 模式下使用 FAST_READ 指令 (0Bh) 执行 XIP

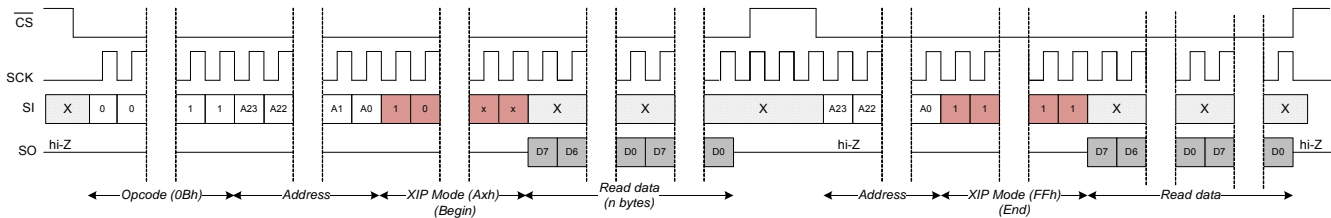
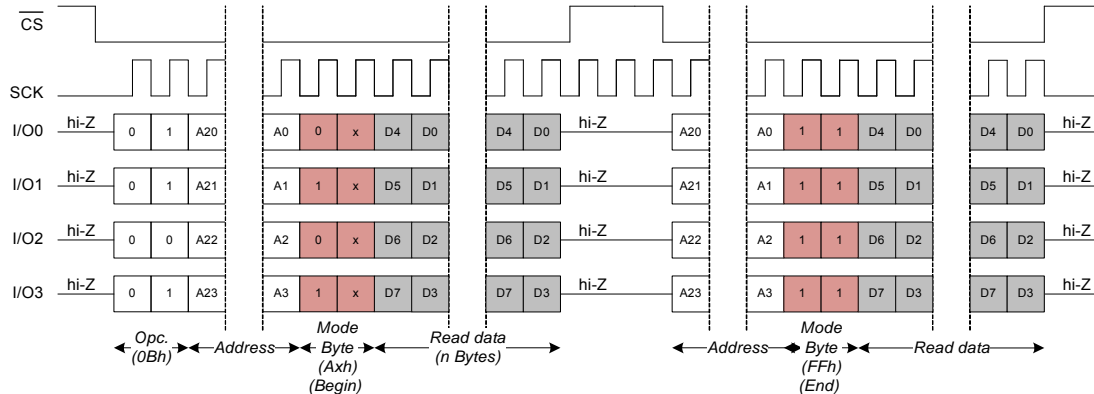


图 41. 在 QPI 模式下使用 FAST_READ 指令 (0Bh) 执行 XIP



系统资源指令

软件复位（RESET）指令

RESET 指令将复位整个器件，并使其可接收各条命令。I/O 模式将被配置为 SPI。所有非易失性寄存器或非易失性寄存器位均保持它们的值。默认情况下，所有易失性寄存器或易失性寄存器位均为逻辑‘0’。该操作需要占用 t_{RESET} 的时长。没有任何存储/回读操作被执行。启动软件复位过程需要使用复位使能（RSTEN）指令。这样可以防止发生所有意外复位。因此，软件复位是一个执行两条命令的序列。

注意：RSTEN 命令之后的任何命令（RESET 除外）都将清除复位使能条件，并防止识别随后的 RESET 命令。

注意：如果将 WIP (SR[0]) 位于高电平并输入 RSTEN/RESET 指令，则器件将忽略 RSTEN/RESET 指令。

注意： $\overline{\text{WP}}$ 和 NC (I/O3) 引脚的功能由配置寄存器中的四线位 CR[1] 控制。如果将四线位配置为逻辑‘1’，那么 $\overline{\text{WP}}$ 和 NC (I/O3) 将分别被配置为 I/O2 和 I/O3。在其他情况下，将配置 $\overline{\text{WP}}$ 和 NC (I/O3) 功能。

表 9 汇总了软件复位后的器件状态。

表 9. 软件复位状态

状态 1	状态 2	状态 3	I/O 模式和寄存器位
待机	软件复位	待机	I/O 模式: SPI SRWD SR[7]: 与状态 1 相同 SNL SR[6]: 与状态 1 相同 TBPROT SR[5]: 与状态 1 相同 BP2 SR[4]: 与状态 1 相同 BP1 SR[3]: 与状态 1 相同 BP0 SR[2]: 与状态 1 相同 WEL SR[1]: 0 WIP SR[0]: 0 QUAD CR[1]: 与状态 1 相同

图 42. SPI 模式下的复位指令

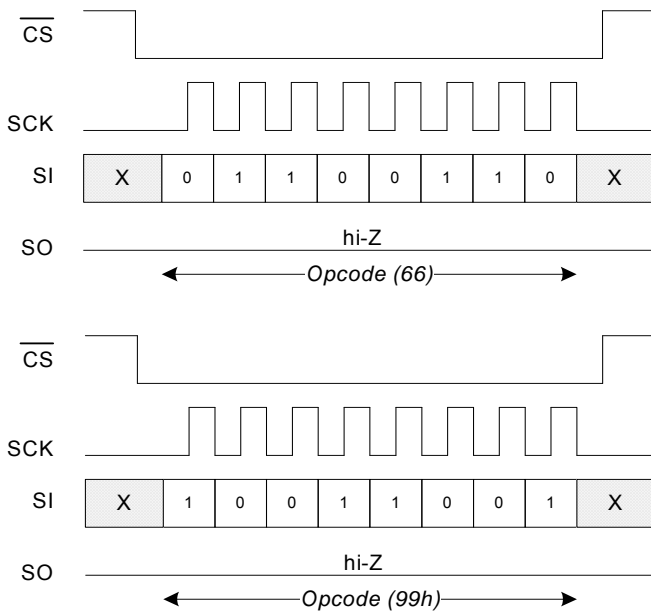


图 43. DPI 模式下的复位指令

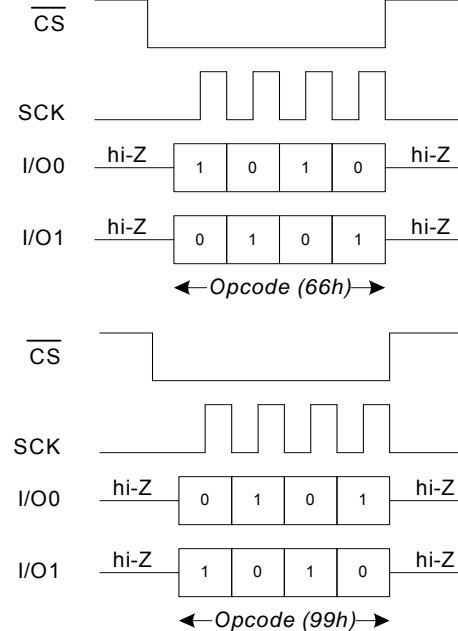
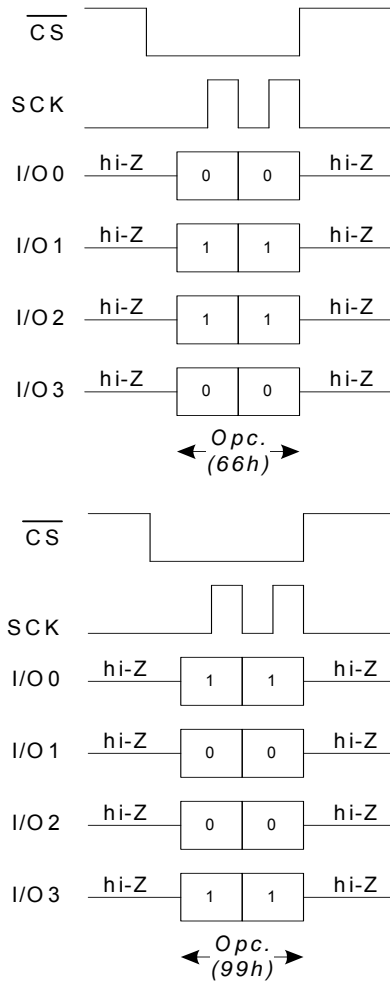


图 44. QPI 模式下的复位指令



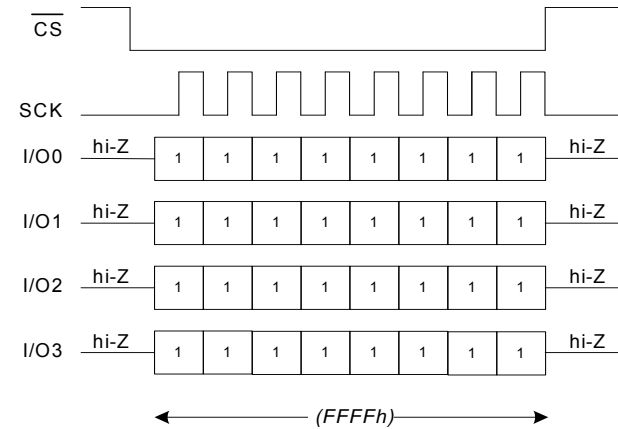
注意：在 QPI 模式下执行 RSTEN/RESET 指令前，必须将四线位 CR[1] 设置为逻辑 ‘1’。

默认的恢复指令

CY14V101QS 提供了默认的恢复模式，从而允许器件返回到 SPI 模式。在八个 SCLK 周期内，将所有 I/O（I/O3、I/O2、I/O1、I/O0）都设置为逻辑高电平会使器件进入已知模式（SPI），以便在未知的起始模式下，主机仍能与器件进行通信。

注意： \overline{WP} 和 \overline{NC} （I/O3）引脚的功能由配置寄存器中的四线位 CR[1] 控制。如果将四线位配置为逻辑 ‘1’，则 \overline{WP} 和 \overline{NC} （I/O3）分别被配置为 I/O2 和 I/O3。在其他情况下，将配置 \overline{WP} 和 \overline{NC} （I/O3）功能。

图 45. 默认的恢复指令



休眠（HIBEN）指令

通过 HIBEN 指令可使 nvSRAM 进入休眠模式。发出 HIBEN 指令后，nvSRAM 需要经过 t_{SS} 时间来处理 HIBEN 请求。成功寄存和处理 HIBEN 命令后，nvSRAM 将 HSB 置于低电平，并执行存储操作，从而能够将数据存入非易失性单元内，然后进入休眠模式。从寄存休眠指令起，经过 t_{HIBEN} 时间后器件会开始消耗 I_{ZZ} 电流。发出 HIBEN 指令后，器件无法进行正常操作。在休眠模式下，将忽略 SCK 和 SI 引脚，并且将 SO 引脚置于高阻态（HI-Z），但是器件仍继续监控 CS 引脚。

要从休眠模式唤醒 nvSRAM，必须通过将 \overline{CS} 引脚从高电平跳转到低电平状态来选择器件。检测到 \overline{CS} 引脚的下降沿，并且持续

表 10. 唤醒（退出休眠）状态

状态 1	状态 2	状态 3	I/O 模式和寄存器位
待机	休眠	待机	I/O 模式：与状态 1 相同的模式（SPI/DPI/QPI） SRWD SR[7]：与状态 1 相同 SNL SR[6]：与状态 1 相同 TBPROT SR[5]：与状态 1 相同 BP2 SR[4]：与状态 1 相同 BP1 SR[3]：与状态 1 相同 BP0 SR[2]：与状态 1 相同 WEL SR[1]：0 WIP SR[0]：0 QUAD CR[1]：与状态 1 相同

图 46. SPI 模式下的 HIBEN 指令

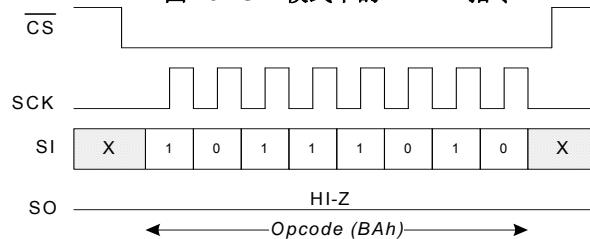
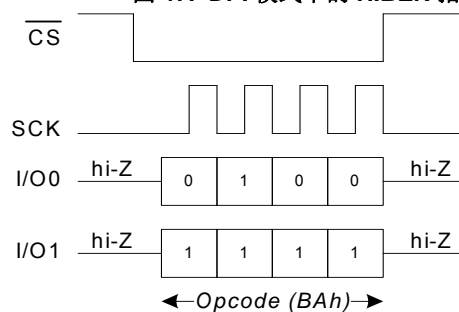


图 47. DPI 模式下的 HIBEN 指令

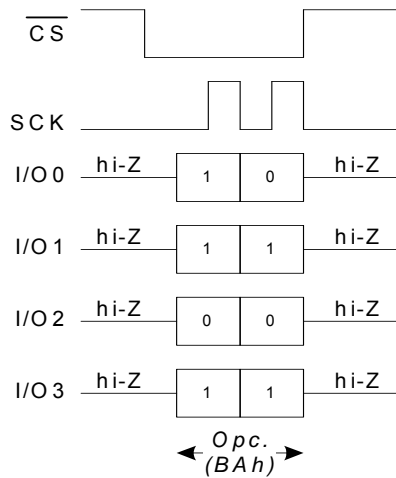


t_{WAKE} 后，器件将被唤醒并且可以正常进行访问操作。器件唤醒后，将返回到进入休眠模式前的工作模式。

注意：只要进入休眠模式，nvSRAM 就会启动非易失性存储周期，每次执行休眠命令后都会产生耐久性周期。仅在上一个存储或回读周期之后已经执行对 SRAM 写操作时，存储周期才会开始。

表 10 汇总了从休眠器件状态唤醒的状态。

图 48. QPI 模式下的 HIBEN 指令



注意：在 QPI 模式下执行 HIBEN 指令前，必须将四线位 CR[1] 设置为逻辑 ‘1’。

表 11. 退出睡眠 (EXSLP) 状态

状态 1	状态 2	状态 3	I/O 模式和寄存器位
待机	睡眠	待机	I/O 模式：与状态 1 相同的模式 (SPI/DPI/QPI) SRWD SR[7]: 与状态 1 相同 SNL SR[6]: 与状态 1 相同 TBPROT SR[5]: 与状态 1 相同 BP2 SR[4]: 与状态 1 相同 BP1 SR[3]: 与状态 1 相同 BP0 SR[2]: 与状态 1 相同 WEL SR[1]: 与状态 1 相同 WIP SR[0]: 0 QUAD CR[1]: 与状态 1 相同

图 49. SPI 模式中的睡眠指令

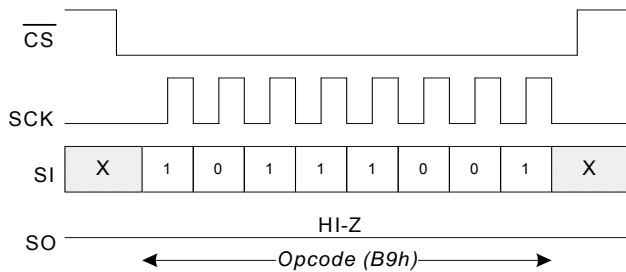
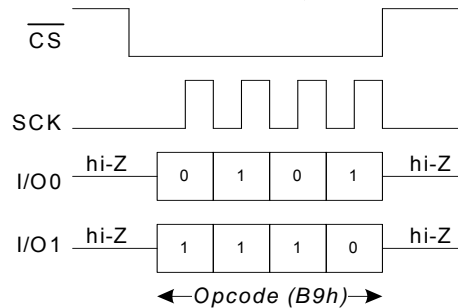


图 50. DPI 模式中的睡眠指令



睡眠 (SLEEP) 指令

可通过睡眠指令使 nvSRAM 进入睡眠模式。发出睡眠指令后，nvSRAM 需要经过 t_{SLEEP} 的时间来处理 SLEEP 请求，并消耗 I_{SLEEP} 大小的电流。发出睡眠指令后，器件无法进行正常操作。在睡眠模式下，所有引脚均可用。

为了将 nvSRAM 从睡眠模式唤醒，必须输入 EXSLP 指令。经过 t_{EXSLP} 的时间后，可以对 nvSRAM 进行正常的操作。器件唤醒后，将返回到进入睡眠模式前的工作模式。当器件在睡眠模式下，除了 EXSLP 和 RDSR 指令以外，其他任何指令都被忽略。

表 11 汇总了器件从睡眠模式唤醒后的状态。

图 51. QPI 模式中的睡眠指令

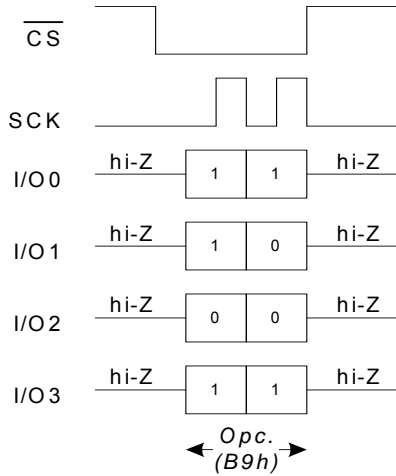


图 52. SPI 模式下的 EXSLP 指令

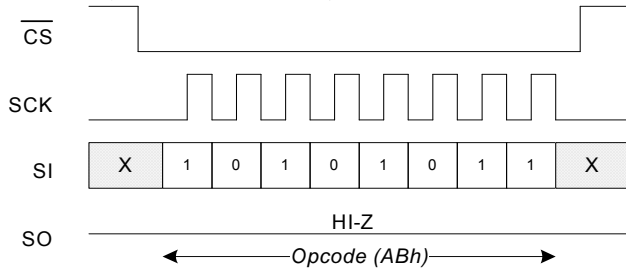


图 53. DPI 模式下的 EXSLP 指令

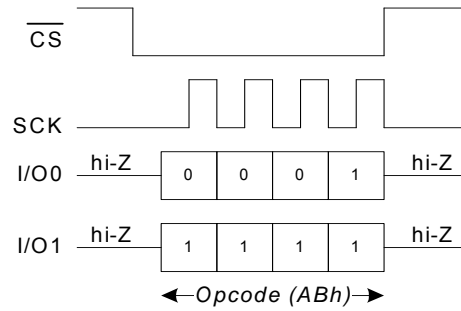
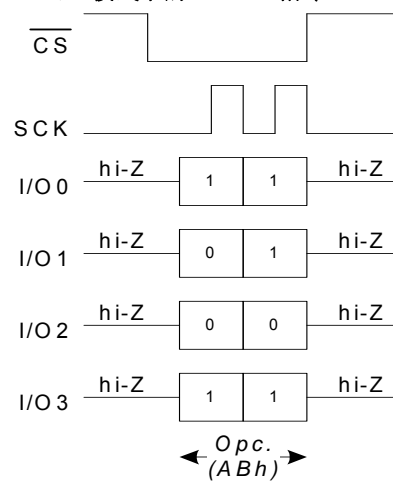


图 54. QPI 模式下的 EXSLP 指令



寄存器指令

读取状态寄存器（RDSR）指令

通过使用 RDSR 指令能够以高达 108 MHz 的 SPI 频率访问状态寄存器。该指令用于检测器件的状态。

注意： 读取状态寄存器的最后一位后，该器件将返回到状态寄存器的第一位。

图 55. SPI 模式下的 RDSR 指令

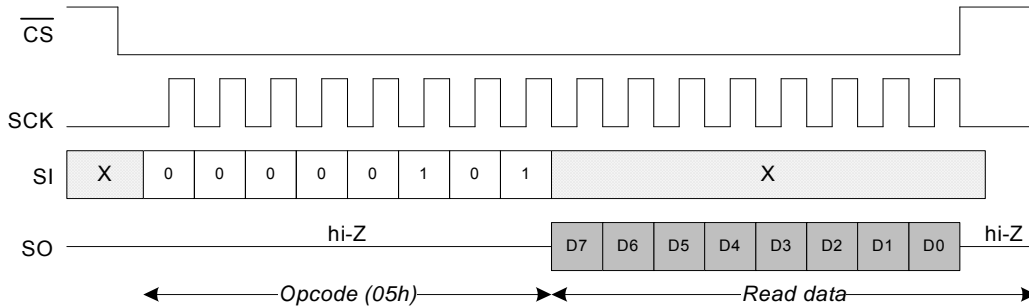


图 56. DPI 模式下的 RDSR 指令

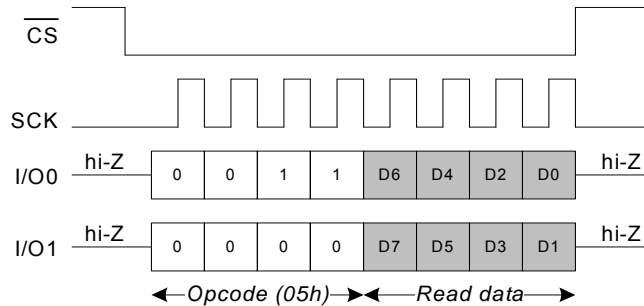
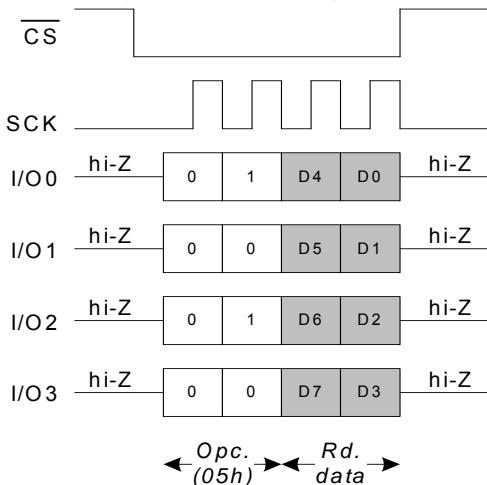


图 57. QPI 模式下的 RDSR 指令



写入状态寄存器（WRSR）指令

用户可通过 WRSR 指令对状态寄存器进行写操作。但该指令只能修改可写位，即：位 2（BP0）、位 3（BP1）、位 4（BP2）、位 5（TBPROT）、位 6（SNL）和位 7（SRWD）。WRSR 指令是一条写入指令，需要在发送该指令前将 WEL 位设置为 ‘1’（通过使用 WREN 指令实现）。WRSR 指令操作码在 CS 下降沿后发送，后面紧接的是要存储到状态寄存器中的 8 位数据。如上面所述，WRSR 指令只能修改状态寄存器的位 2、位 3、位 4、位 5、位 6 和位 7。

注意： 写入状态寄存器的有效值将在存储操作完成后被保存到非易失性存储器内。如果自动存储已禁用，必须进行软件存储操作以保存对状态寄存器所做的修改。

注意： 对状态寄存器进行写序列后，状态寄存器中的 WEL 位将被复位到 ‘0’。

图 58. SPI 模式中的 WRSR 指令

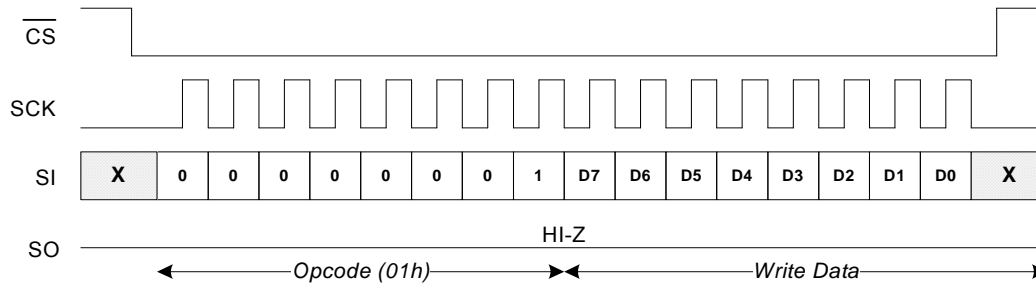
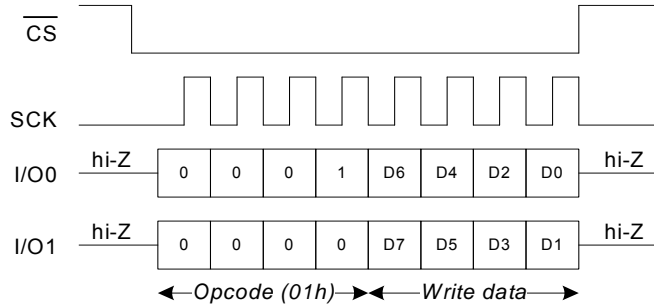


图 59. DPI 模式下的 WRSR 指令



读取配置寄存器（RDCR）指令

通过使用 RDCR 指令能够以高达 108 MHz 的 SPI 频率访问配置寄存器。下图显示的是 SPI、DPI 和 QPI 模式下的配置寄存器指令发送波形。

注意：在读取配置寄存器的最后一位后，器件将返回到配置寄存器的第一位。

图 60. QPI 模式下的 WRSR 指令

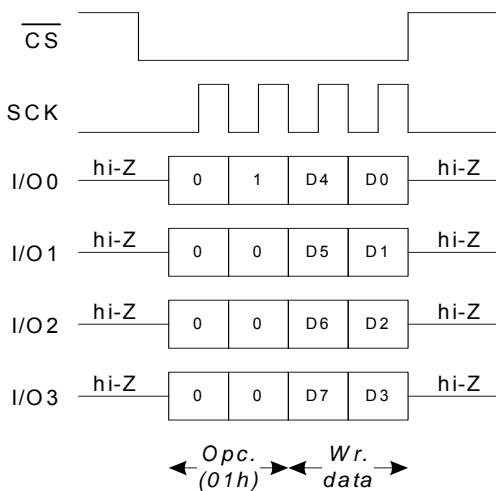


图 61. SPI 模式下的 RDCR 指令

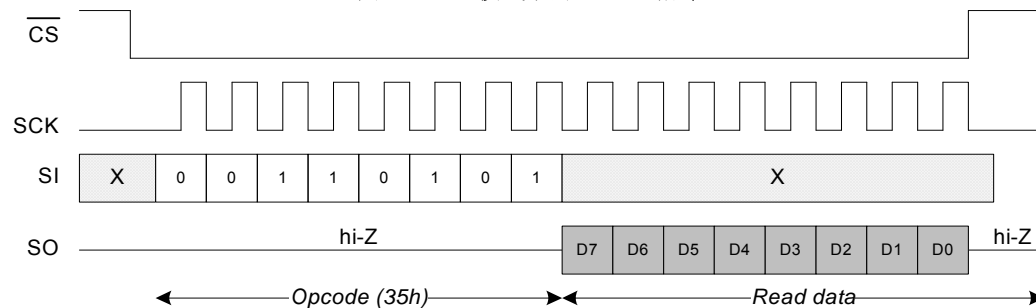


图 62. DPI 模式下的 RDCR 指令

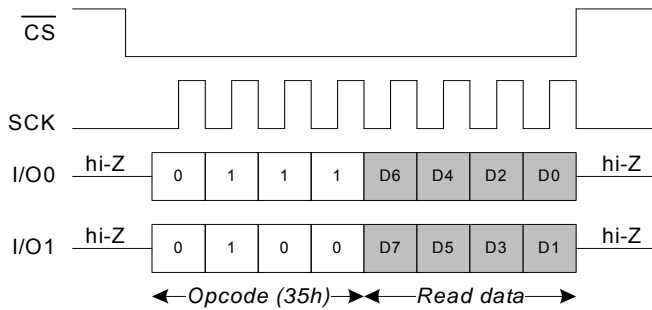
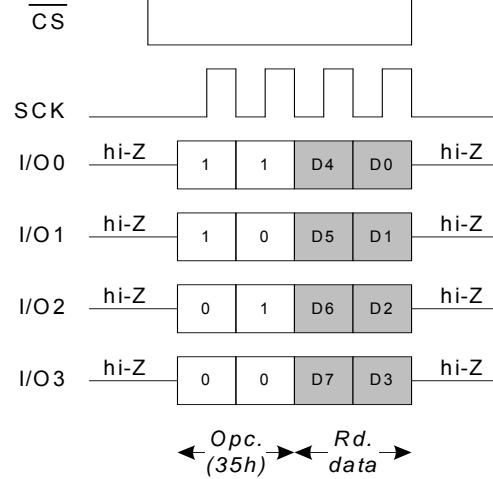


图 63. QPI 模式下的 RDCR 指令



注意：在 QPI 模式下执行 RDCR 指令前，必须将四线位 CR[1] 设置为逻辑 ‘1’。

写入配置寄存器（WRCR）指令

通过设置四线位，用户可以使用写入配置寄存器（WRCR）指令更改器件的数据宽度。当使用读取四线输出、四线 I/O 读取和四线输入写入命令时，必须将四线位设置为 1。四线位是非易失性的。

注意：使能 QPI 模式（QPIEN）指令不会设置配置寄存器中的四线位。

注意：建议始终使用如表 8 中所提供的 RFU 位。

图 64. SPI 模式下的 WRCR 指令

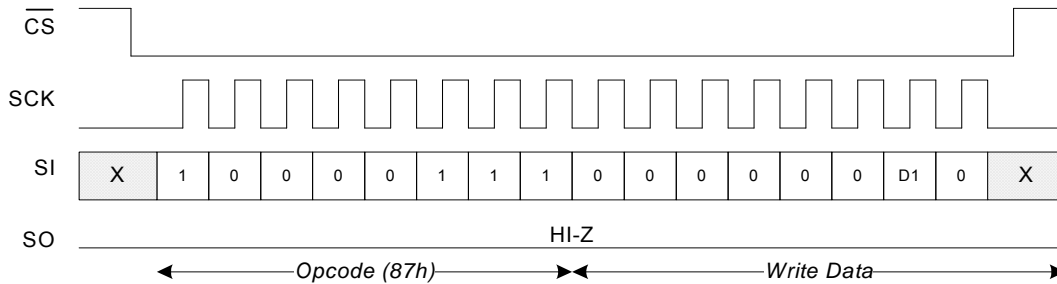
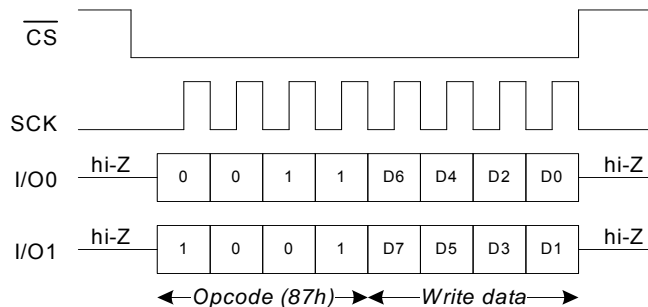


图 65. DPI 模式下的 WRCR 指令



标识寄存器（RDID）指令

通过使用 RDID 指令，能够以高达 40 MHz 的 SPI 频率读取 JEDEC 分配的制造商 ID 和产品 ID。该指令可用于识别总线上的器件。CS# 进入低电平状态后，通过移入 RDID 的操作码可以发出 RDID 指令。

器件 ID 是 4 字节的只读代码，仅用于标识 1 Mb QPI nvSRAM 产品，包括产品系列号、产品配置和密度。

RDID 指令读取 4 字节的器件 ID 结构（不能写入到该结构）。每次只能读取该结构中一个字节。第一个被访问的字节是该结构的

最高有效字节 ID[31:24]，第二个被访问的字节是 ID[23:16]，...，最后被访问的字节是 ID[7:0]。

注意：由于始终按相同的顺序访问该结构，因此不需要进行地址传输。但是会使用一个内部 2 位地址指针。当解码该操作码时，该指针被初始化为 ‘0’。访问每个字节后，内部地址指针将被递增。地址指针从 ‘3’ 环绕到 ‘0’；在访问第四个字节 ID[7:0] 后，将访问第一个字节 ID[31:24]。可以在 SPI、DPI 或 QPI 模式下发出该指令。

表 12. 器件标识

器件	制造商 ID	产品 ID	容量	芯片版本
	31-21	20-7	6-3	2-0
	11 位	14 位	4 位	3 位
	00000110100	00001100010001	0100	001

图 66. SPI 模式下的 RDID 指令

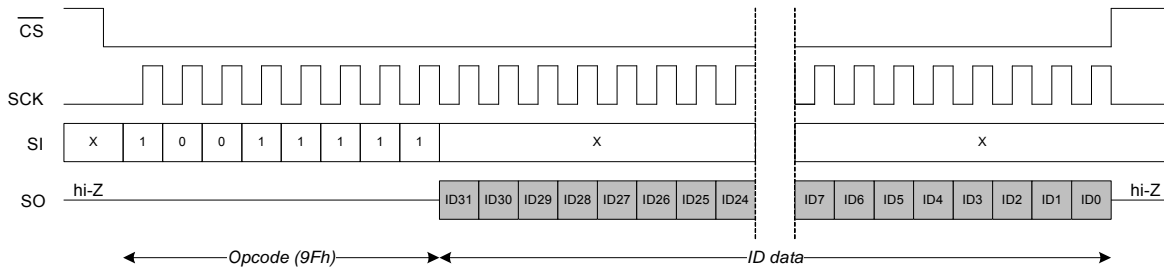


图 67. DPI 模式下的 RDID 指令

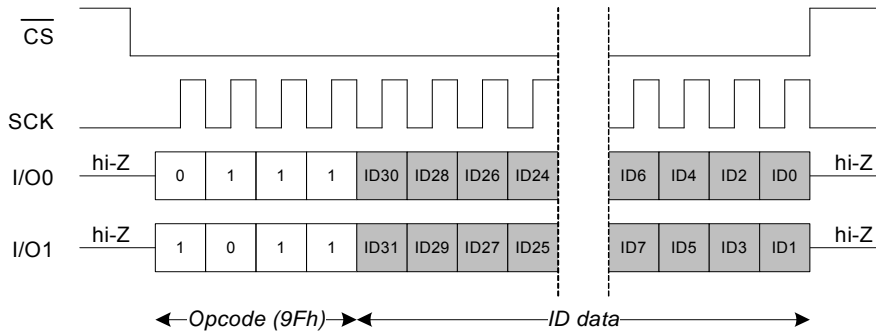
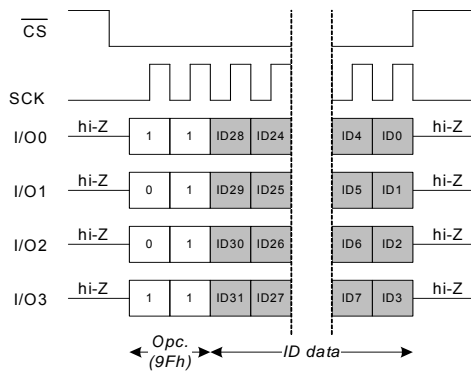


图 68. QPI 模式下的 RDID 指令



注意：在 QPI 模式下执行 RDID 指令前，必须将四线位 CR[1] 设置为逻辑 ‘1’。

标识寄存器（FAST_RDID）指令

FAST_RDID 指令与 RDID 指令相同，但它允许在操作码后放置一个虚拟字节。通过使用 FAST_RDID 指令能以高达 108 MHz 的 SPI 频率读取 JEDEC 分配的制造商 ID 和产品 ID。

图 69. SPI 模式下的 FAST_RDID 指令

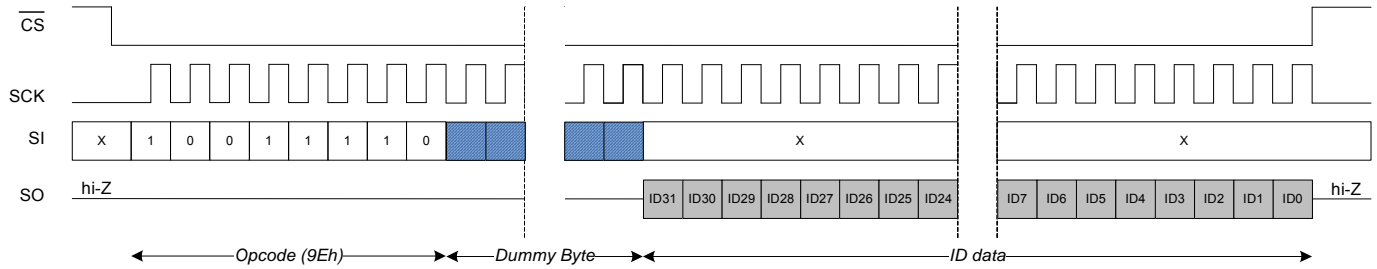


图 70. DPI 模式下的 FAST_RDID 指令

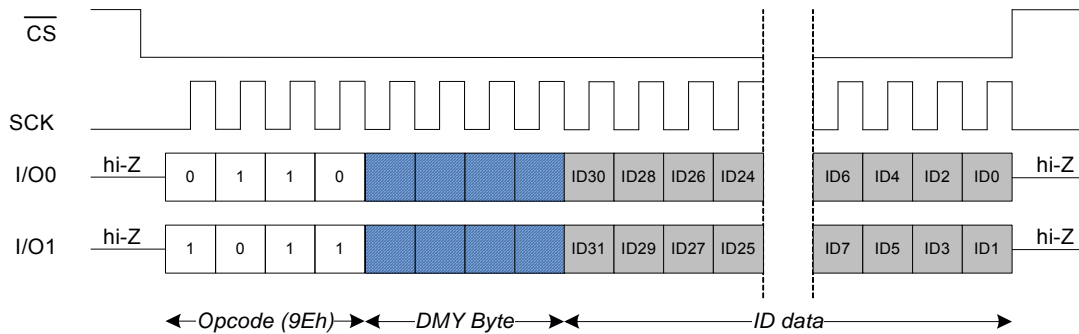
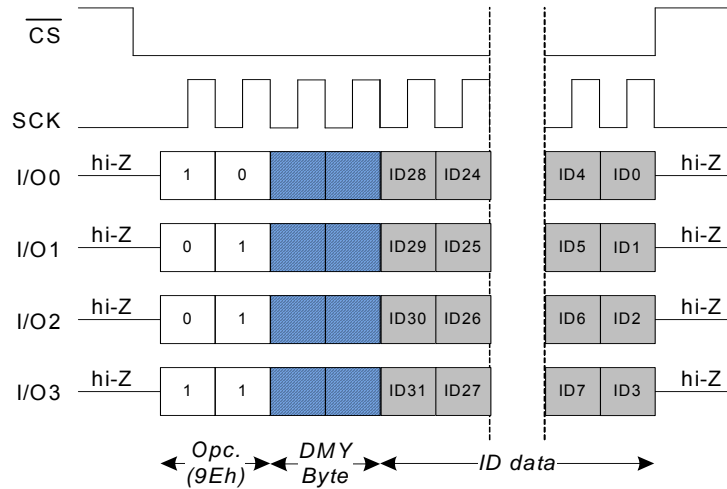


图 71. QPI 模式下的 FAST_RDID 指令



序列号寄存器写入（WRSN）指令

序列号是一个 8 字节的可编程存储空间，为用户提供了用于对器件进行唯一标识。它通常由一个两字节的客户 ID、五字节的唯一序列号和一字节的循环冗余校验组成。然而，器件不计算循环冗余校验，系统设计者可通过任意方式使用八个字节的存储空间。八个字节位置的默认值为 0x00。

通过使用 WRSN 指令，可以编写该序列号。要想编写序列号，必须使用 WREN 指令使能写功能。通过在突发模式下使用 WRSN 指令，可以编写 8 字节的序列号。编写序列号最后的字节后，器件将返回到序列号的第一个（MSB）字节。可使用状态寄存器的 SNL 位锁定序列号。该位被置为 ‘1’ 后，无法对序列号进行任何修改。将 SNL 位设置为 ‘1’ 后，使用 WRSN 指令不会对序列号产生任何影响。执行该指令前需要设置 WEL 位。如果状态寄存器中的 SRWD 位未被设置为 ‘1’，那么在执行该指令后，

WEL 位将复位为 ‘0’。可以在 SPI、DPI 或 QPI 模式下发出该指令。

通过使用 WRSN 指令能够以高达 108 MHz 的 SPI 频率写入序列号。

注意：需要通过存储操作（自动存储或软件存储）将序列号存储到非易失性存储器中。如果自动存储已禁用，必须执行软件存储操作来保存和锁定序列号。如果 SNL 位被设置为 ‘1’，并且未被保存（自动存储被禁用），那么在下次电源循环中，SNL 位和序列号将默认被设置为 ‘0’。如果 SNL 位被设置为 ‘1’ 并保存其设置，无法将它清除为 ‘0’。执行该指令前需要设置 WEL 位。可以在 SPI、DPI 或 QPI 模式下发出该指令。

注意：完成该指令后，WEL 位被复位为 ‘0’。

图 72. SPI 模式下的 WRSN 指令

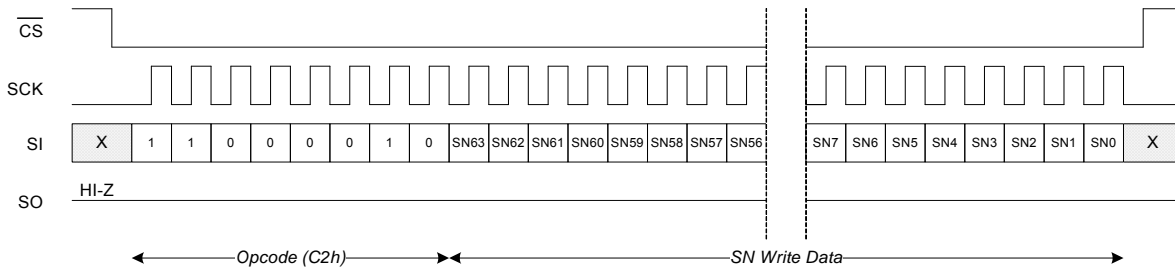


图 73. DPI 模式中的 WRSN 指令

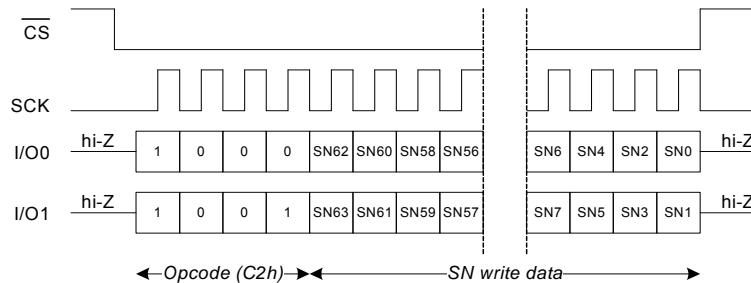
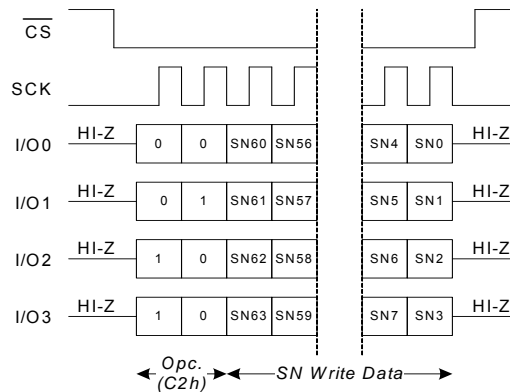


图 74. QPI 模式中的 WRSN 指令



序列号寄存器读取（RDSN）指令

通过使用 RDSN 指令，能够以高达 40 MHz 的 SPI 频率读取序列号。可以在突发模式下读取序列号，实现一次性读取所有八个字节。读取序列号的最后字节后，器件将返回到序列号的第一个

（MSB）字节。 \overline{CS} 进入低电平状态后，通过移入 RDSN 的操作码可以发送 RDSN 指令。此后，nvSRAM 将移出八字节序列号。可以在 SPI、DPI 或 QPI 模式下发出该指令。

图 75. SPI 模式中的 RDSN 指令

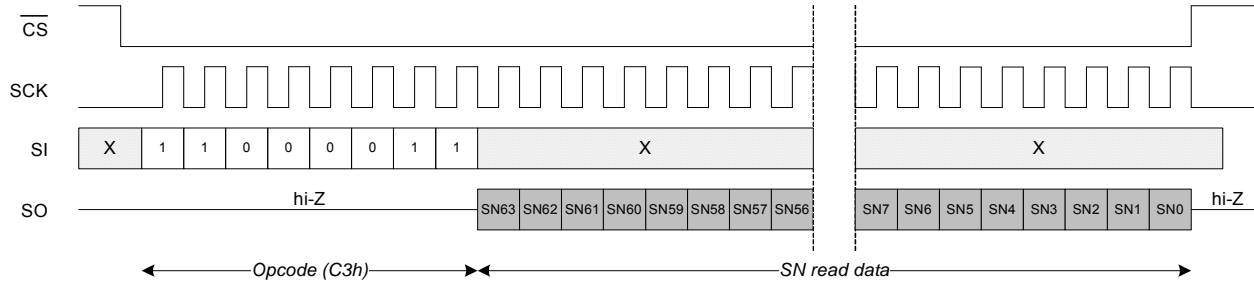


图 76. DPI 模式中的 RDSN 指令

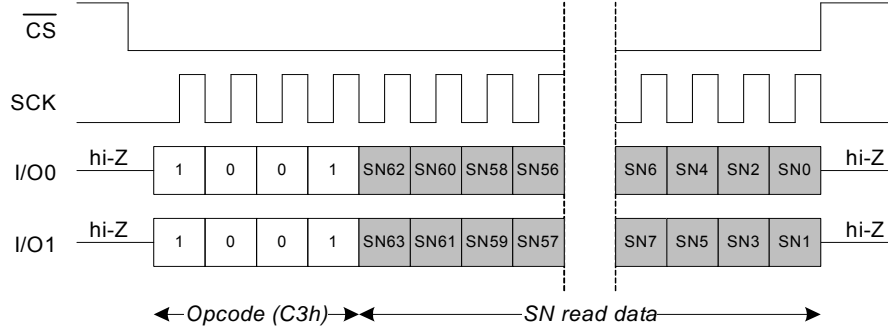
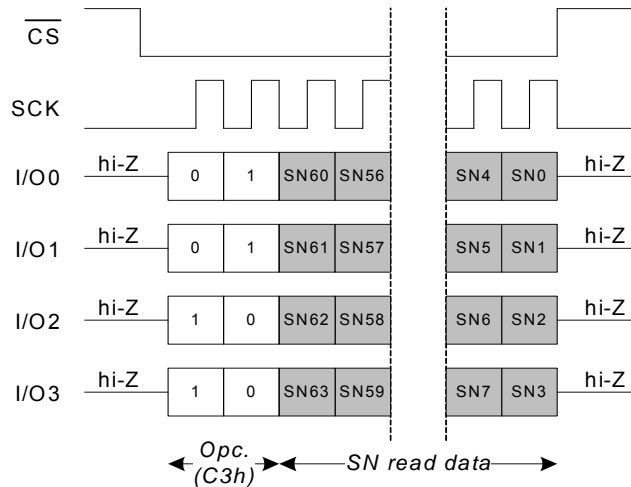


图 77. QPI 模式下的 RDSN 指令



注意：在 QPI 模式下，如果要执行 RDSN 指令，应先将四线位 CR[1] 设置为逻辑 ‘1’。

快速读取序列号寄存器（FAST_RDSN）指令

FAST_RDSN 指令与 RDSN 指令相同，但它允许在操作码后放置一个虚拟字节。能以高达 108 MHz 的频率使用 FAST_RDSN 指令。

图 78. SPI 模式下的 FAST_RDSN 指令

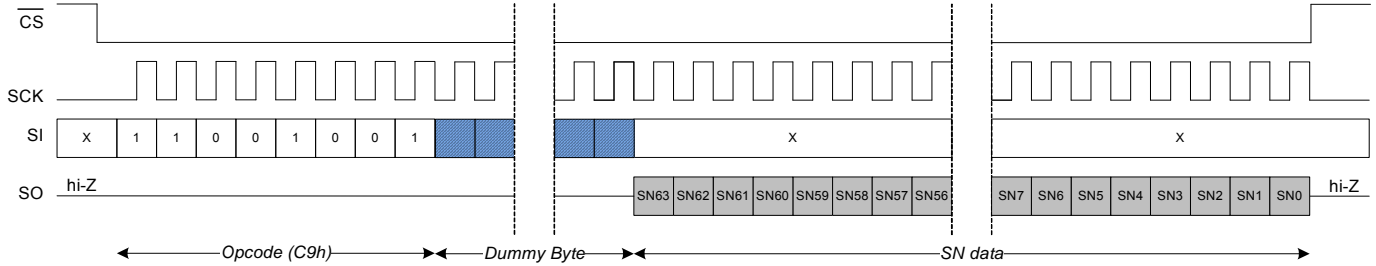


图 79. DPI 模式下的 FAST_RDSN 指令

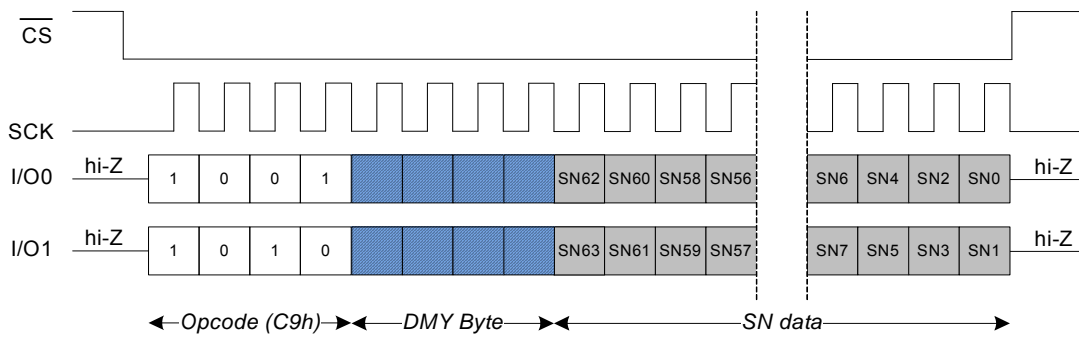
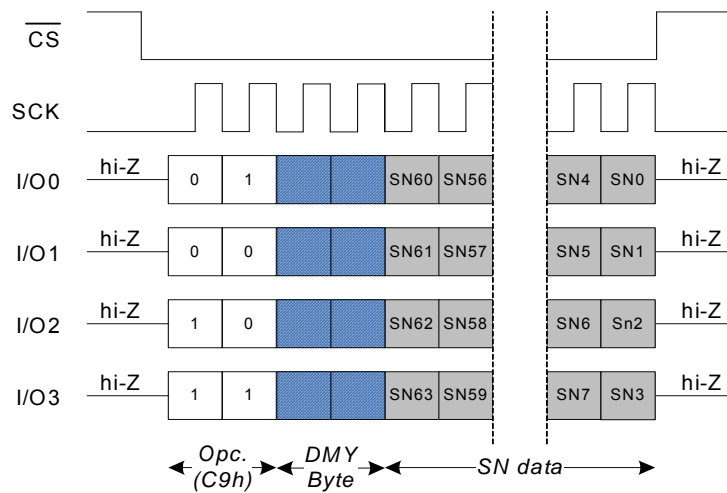


图 80. QPI 模式下的 FAST_RDSN 指令



NV 特定指令

nvSRAM 器件提供了四条特殊指令用于访问 nvSRAM 特定功能，这四条特殊指令分别为：STORE、RECALL、ASEN 和 ASDI。

软件存储（STORE）指令

执行 STORE 指令后，nvSRAM 进行软件存储操作。无论从上次存储或回读操作后是否进行写入，均进行存储操作。要发送该指令，器件必须使能写入功能（WEL 位 = ‘1’）。可以在 SPI、DPI 或 QPI 模式下发送该指令。

注意：发送 STORE 指令后，将在 \overline{CS} 的上升沿上清除 WEL 位。

图 81. SPI 模式下的存储指令

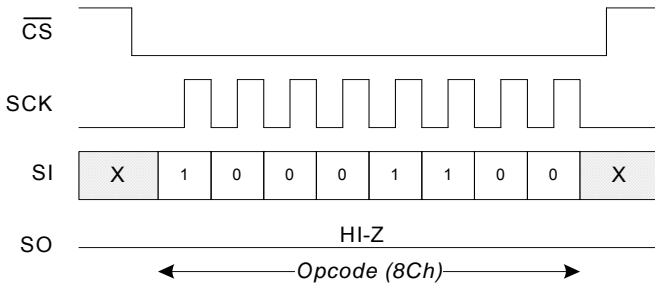
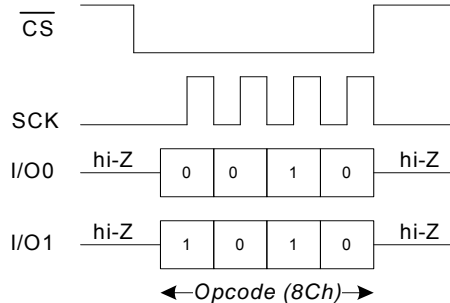


图 82. DPI 模式下的存储指令



软件回读（RECALL）指令

执行 RECALL 指令后，nvSRAM 将执行软件回读操作。要想发送该指令，器件必须使能写入功能（WEL = ‘1’）。可以在 SPI、DPI 或 QPI 模式下发送该指令。

注意：发送 RECALL 指令后，将在 \overline{CS} 的上升沿上清除 WEL 位。

图 83. QPI 模式中的存储指令

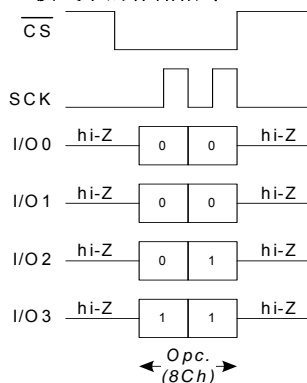


图 84. SPI 模式下的回读指令

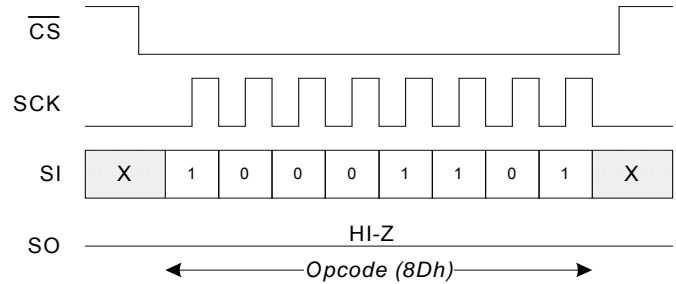


图 85. DPI 模式下的回读指令

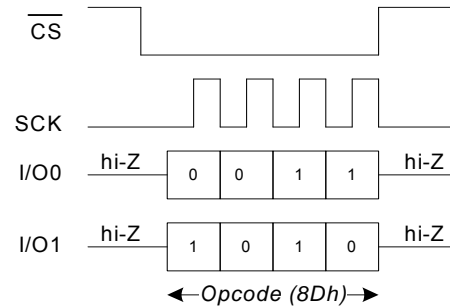
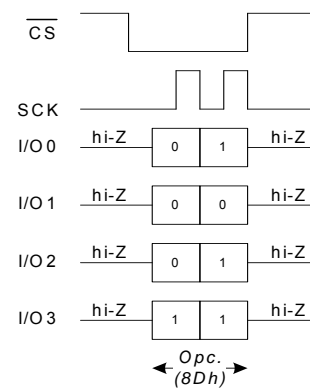


图 86. QPI 模式中的回读指令



自动存储使能 (ASEN) 指令

通过自动存储使能指令可以在 nvSRAM 器件上使能自动存储功能。该设置不是非易失性操作，随后需要一个存储序列来保持加电周期。要想发送该指令，器件必须使能写入功能 (WEL = '1')。可以在 SPI、DPI 或 QPI 模式下发出该指令。

注意：如果执行 ASDI 和 ASEN 指令，在软件序列处理期间 (t_{SS})，器件将处于繁忙状态。

注意：发出 ASEN 指令后，将在 \overline{CS} 的上升沿上清除 WEL 位。

图 87. SPI 模式下的 ASEN 指令

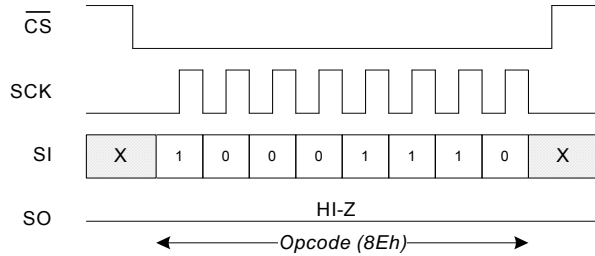


图 88. DPI 模式下的 ASEN 指令

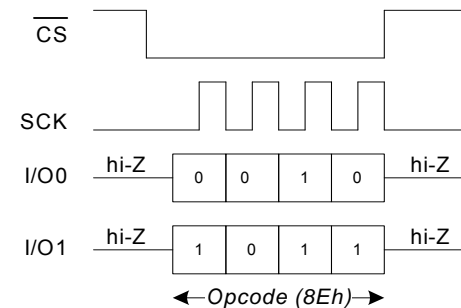
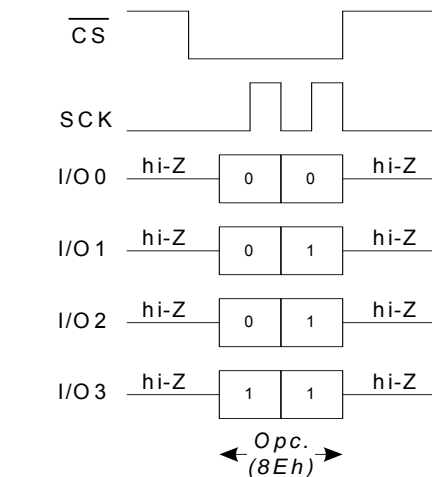


图 89. QPI 模式下的 ASEN 指令



自动存储禁用 (ASDI) 指令

在该器件中默认使能自动存储功能。ASDI 指令禁用自动存储功能。该设置不是非易失性操作，随后需要一个存储序列来保持加电周期。要想发送该指令，器件必须使能写入功能 (WEL = '1')。可以在 SPI、DPI 或 QPI 模式下发出该指令。

注意：发送 ASDI 指令后，将在 \overline{CS} 的上升沿上清除 WEL 位。

图 90. SPI 模式下的 ASDI 指令

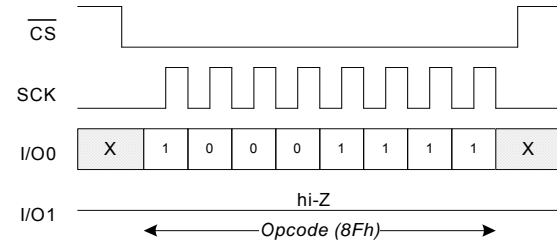


图 91. DPI 模式下的 ASDI 指令

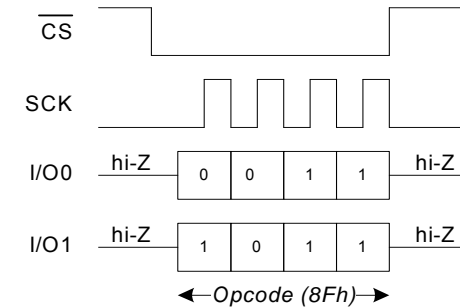
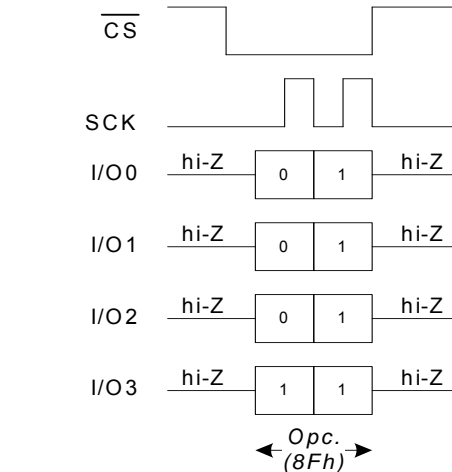


图 92. QPI 模式下的 ASDI 指令



注意：在 QPI 模式下执行 ASDI 指令前，必须将四线位 CR[1] 设置为逻辑 '1'。

最大额定值

超过最大额定值可能会缩短器件的使用寿命。这些用户指导未经过测试。

存储温度 -65 °C 到 +150 °C

最长存储时间

在 150°C 环境温度下 1000 小时

在 85°C 环境温度下 20 年

最高结温 150 °C

V_{CC} (相对于 V_{SS}) 的供电电压 -0.5 V 到 +4.1 V

V_{CCQ} (相对于 V_{SS}) 的供电电压 -0.5 V 到 +2.45 V

在高阻 (High Z) 状态下

输出的直流电压 -0.5 V 到 $V_{CCQ} + 0.5$ V

输入电压 -0.5 V 到 $V_{CCQ} + 0.5$ V

在接地电位的所有引脚上的

瞬变电压 (< 20 ns) -2.0 V 到 $V_{CCQ} + 2.0$ V

封装功率耗散能力 ($T_A = 25$ °C)

16-SOIC 1.0 W

24-FPGA 1.0 W

封装功率耗散能力

($T_A = 25$ °C) 1.0 W

表面组装铅焊

温度 (3 秒) +260 °C

直流输出电流

(一次一个输出, 持续时间为 1 秒) 15 mA

静电放电电压

(根据 MIL-STD-883, 方法 3015) > 2001 V

栓锁电流 > 140 mA

工作范围

范围	环境温度	V_{CC}	V_{CCQ}
工业级的直流特性	-40°C ~ +85°C	2.7 V ~ 3.6 V	1.71 V ~ 2.0 V
扩展的工业级	-40°C ~ +105°C	2.7 V ~ 3.6 V	1.71 V ~ 2.0 V

直流规范

参数	说明	测试条件	最小值	典型值 ^[1]	最大值	单位
V_{CC}	电源 — 内核电压	—	2.70	3.00	3.60	V
V_{CCQ}	电源 — I/O 电压	—	1.71	1.80	2.00	V
I_{CC1}	V_{CC} 的平均读 / 写电流 (所有输入均进行切换, 没有任何输出负载)	SPI = 1 MHz	—	—	1.00	mA
		SPI = 40 MHz	—	—	3.00	mA
		QPI = 108 MHz	—	—	33.00	mA
I_{CCQ1}	V_{CCQ} 的平均电流 (所有输入均进行切换, 没有任何输出负载)	SPI = 1 MHz	—	—	150.00	μA
		SPI = 40 MHz	—	—	1.00	mA
		QPI = 108 MHz	—	—	5.00	mA
I_{SB1}	温度为 +85 °C 时的待机电流 ($V_{CC} + V_{CCQ}$)	$\overline{CS} > (V_{CCQ} - 0.2$ V)。非易失性存储器写周期结束后的待机电流 (\overline{CS} 为高电平, 其他 I/O 不受限制, $f_{SCK} \leq 108$ MHz)	—	—	1.70	mA
	温度为 105 °C 时的待机电流 ($V_{CC} + V_{CCQ}$)	—	—	—	2.00	mA
I_{SB2}	温度为 +85 °C 时的待机电流 ($V_{CC} + V_{CCQ}$)	$\overline{CS} > (V_{CCQ} - 0.2$ V)。完成非易失性循环后待机电流的强度。	—	—	280.00	μA
	温度为 105 °C 时的待机电流 ($V_{CC} + V_{CCQ}$)	所有 I/O 均为静态, $f_{SCK} = 0$ MHz	—	—	540.00	μA
I_{CC2}	存储期间 V_{CC} 平均电流	—	—	—	6.00	mA
I_{CC4}	AUTOSTORE (自动存储) 期间 V_{CAP} 的平均电流	—	—	—	6.00	mA
I_{SLEEP}	温度为 85 °C 时的睡眠模式电流 ($V_{CC} + V_{CCQ}$)	$\overline{CS} > (V_{CCQ} - 0.2$ V)。非易失性存储器写周期完成后的睡眠电流。所有 I/O 均为静态, $f_{SCK} = 0$ MHz	—	—	280.00	μA

注释:

1. 典型值的测量条件为: 温度 = 25 °C, $V_{CC} = V_{CC(Typ)}$ 并且 $V_{CCQ} = V_{CCQ(Typ)}$ 。并非 100% 经过了测试。

参数	说明	测试条件	最小值	典型值 ^[1]	最大值	单位
I_{ZZ}	温度为 85 °C 时的休眠模式电流 ($V_{CC} + V_{CCQ}$)	$\overline{CS} > (V_{CCQ} - 0.2 V)$ 。寄存 HIBEN 指令得到寄存后 t_{HIBEN} 的时长。所有输入均为静态且在 CMOS 逻辑电平下配置。	–	–	8.00	μA
I_{IX}	输入漏电流 (\overline{HSB} 除外)	$V_{CCQ} = \text{最大值}, V_{SS} < V_{IN} < V_{CCQ}$	–1.00	–	1.00	μA
	输入漏电流 (针对 \overline{HSB})		–100.00	–	1.00	μA
	输入漏电流 (针对 SPI/DPI 模式下的 \overline{WP})		–2	–	1	μA
I_{OZ}	断开状态输出漏电流	$V_{CCQ} = \text{最大值}, V_{SS} < V_{IN} < V_{CCQ}$	–1.00	–	1.00	μA
V_{IH}	输入高电平	–	$0.70 * V_{CCQ}$	–	$V_{CCQ} + 0.30$	V
V_{IL}	输入低电平电压	–	–0.30	–	$0.30 * V_{CCQ}$	V
V_{OH}	在 $I_{OH} = -2 \text{ mA}$ 时的输出高电平电压	$I_{OH} = -2 \text{ mA}$	$V_{CCQ} - 0.45$	–	–	V
V_{OL}	在 $I_{OL} = 2 \text{ mA}$ 时的输出低电平电压	$I_{OL} = 2 \text{ mA}$	–	–	0.45	V
$V_{CAP}^{[2]}$	存储电容	在 V_{CAP} 引脚和 V_{SS} 之间	61.00	68.00	120.00	μF
$V_{VCAP}^{[3]}$	在 V_{CAP} 引脚上驱动的最大电压	–	–	–	V_{CC}	V

数据保留时间与耐久性

参数	说明	最小值	单位
$DATA_R$	在温度为 85 °C 条件下数据的保留时间	20	年
NV_C	执行非易失性存储操作的次数	1,000	K

电容

参数 ^[3]	说明	测试条件	最大值	单位
C_{IN}	输入电容	$T_A = 25 \text{ }^\circ\text{C}, f = 1 \text{ MHz}, V_{CC} = V_{CC(typ)}, V_{CCQ} = V_{CCQ(typ)}$	6.00	μF
C_{SCK}	时钟输入电容			
C_{OUT}	输出引脚电容			

热阻

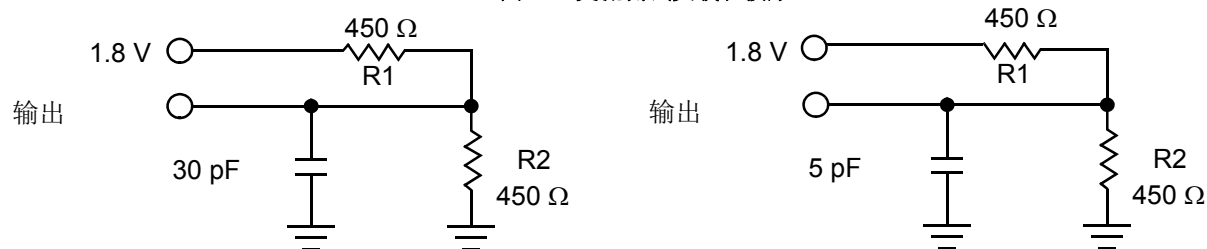
参数 ^[3]	说明	测试条件	16-SOIC	24-FBGA	单位
Θ_{JA}	热阻 (结至环境)	根据 EIA/JESD51 的要求, 测试条件遵循测试热阻的标准测试方法和过程。	61.21	32.08	$^\circ\text{C/W}$
Θ_{JC}	热阻 (结至外壳)		26.20	14.29	

注释:

- V_{CAP} 的最小值可提供足够的电荷来成功完成自动存储操作。在加电回读周期内, V_{CAP} 的最大值可确保通过最小的电压给 V_{CAP} 上的电容充电。这样, 在紧急断电时, 仍能够顺利完成自动存储操作。因此, 建议使用的电容属于规定的最小极限值和最大极限值内。请参考应用笔记 [AN43593](#), 了解有关 V_{CAP} 选项的详细信息。
- 这些参数由设计保证, 并未经过测试。

交流测试负载和波形

图 93. 交流测试负载和波形



交流测试条件

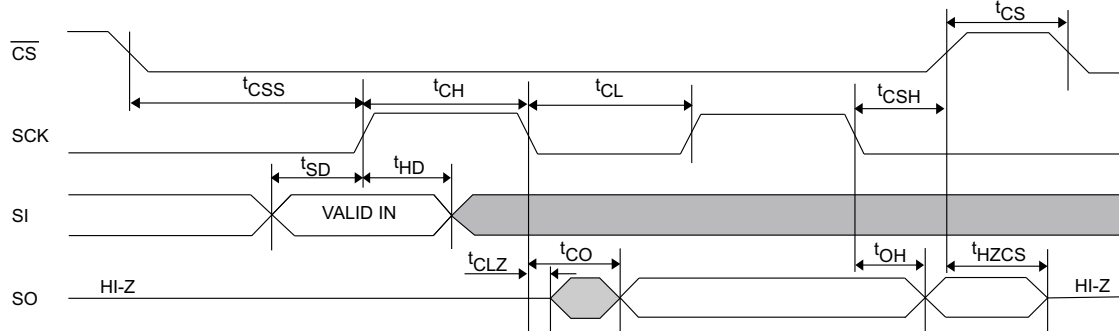
说明	CY14V101QS
输入脉冲电平	0 V 到 1.8 V
输出上升和下降时间（10%–90%）	≤ 1.8 ns
输入和输出时序参考电平	0.9 V

交流开关特性

参数 ^[4]	说明	最小值	最大值	单位
f_{SCK}	时钟频率 (QPI)	—	108.00	MHz
t_{CL}	时钟脉冲宽度低电平	$0.45 * 1/f_{SCK}$	—	ns
t_{CH}	时钟脉冲宽度高电平	$0.45 * 1/f_{SCK}$	—	ns
t_{CS}	\overline{CS} 为高电平的时间			
	结束读取操作所需的时间	10.00	—	ns
	结束写入操作所需的时间	10.00	—	ns
t_{CSS}	\overline{CS} 建立时间	5.00	—	ns
t_{CSH}	\overline{CS} 保持时间	5.00	—	ns
t_{SD}	数据的建立时间	2.00	—	ns
t_{HD}	数据输入的保持时间	3.00	—	ns
t_{SW}	\overline{WP} 建立时间	2.00	—	ns
t_{HW}	\overline{WP} 保持时间	2.00	—	ns
t_{CO}	输入有效时间	—	7.00	ns
t_{CLZ}	从时钟为低电平到输出为低阻态的时间	0.00	—	ns
t_{OH}	输出保持时间	1.00	—	ns
t_{HZCS} ^[5]	输出被禁用的时间	—	7.00	ns

开关波形

图 94. 同步数据时序 (模式 0)



注释:

- 测试条件为: 信号跳变时间不大于 1.8 ns, 时序参考电压电平为 $V_{CCQ}/2$, 输入脉冲的电压电平为 0 至 $V_{CCQ(typ)}$, 以及第 45 页上的图 93 中所示的指定 I_{OL}/I_{OH} 的输出负载和负载电容。
- 这些参数由设计保证, 并未经过测试。

自动存储或加电回读

在工作范围内

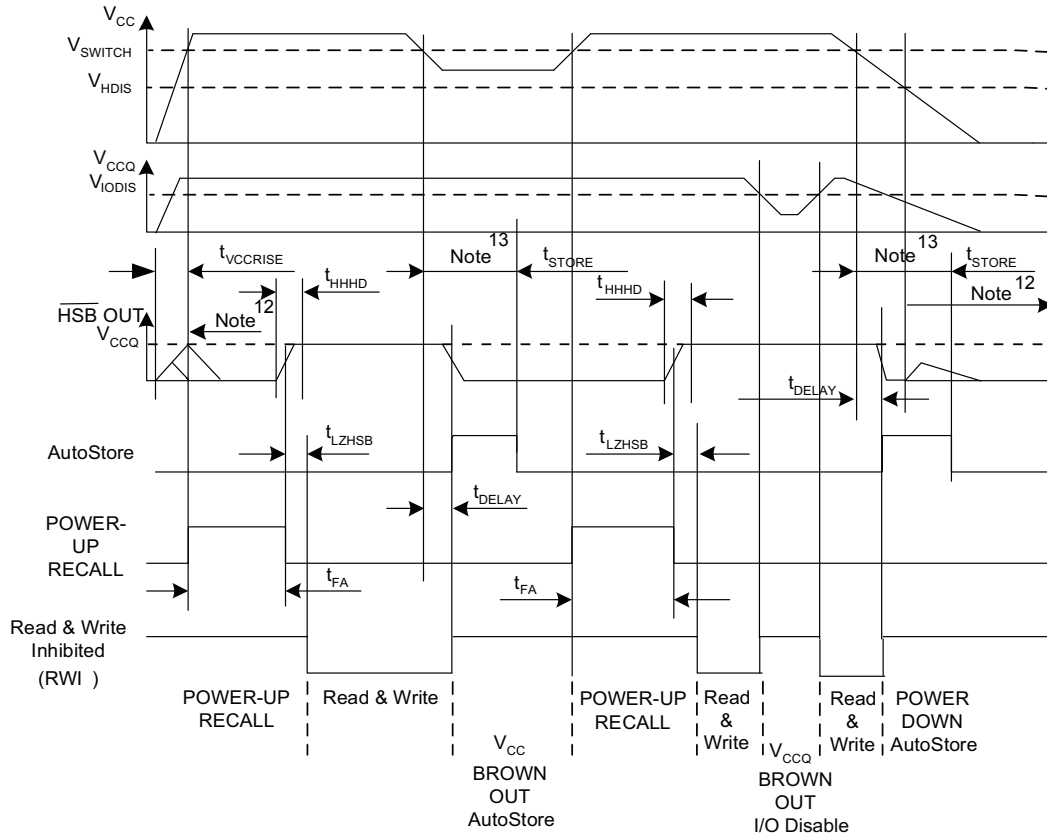
参数	说明	最小值	最大值	单位
$t_{FA}^{[6]}$	加电回读时间	—	20.00	ms
$t_{STORE}^{[7]}$	存储周期时间	—	8.00	ms
$t_{DELAY}^{[8]}$	启动存储周期的时间	—	25.00	ns
V_{SWITCH}	V_{CC} 的低电压触发电平	—	2.60	V
$t_{VCCRRISE}^{[9]}$	V_{CC} 上升时间	150.00	—	μs
$V_{HDIS}^{[9]}$	\overline{HSB} 输出禁用电压	—	1.90	V
$V_{IODIS}^{[10]}$	禁用 I/O 时的 V_{CCQ} 电压	—	1.50	V
$t_{LZHSB}^{[9]}$	从 \overline{HSB} 为高电平到 nvSRAM 有效的的时间	—	5.00	μs
$t_{HHHD}^{[9]}$	\overline{HSB} 高电平有效时间	—	500.00	ns
t_{WAKE}	nvSRAM 从休眠模式唤醒的时间	—	20.00	ms
t_{HIBEN}	nvSRAM 在发出 HIBEN 指令后进入休眠模式所需时间	—	8.00	ms
t_{SLEEP}	nvSRAM 在 \overline{CS} 变为高电平后进入睡眠模式所需要的时间	—	0.00	μs
t_{EXSLP}	nvSRAM 在 \overline{CS} 变为高电平后退出睡眠模式所需要的时间	—	0.00	μs
t_{RESET}	执行软复位所需要的时间	—	500.00	μs

注释:

6. t_{FA} 是从 V_{CC} 超过 V_{SWITCH} 开始计时的。
7. 如果最后一个非易失性存储器写周期完成后尚未对 SRAM 进行写操作，将不会启动自动存储或硬件存储操作。
8. 执行硬件存储、自动存储启动时，在 t_{DELAY} 时间内将持续进行 SRAM 操作。
9. 这些参数由设计保证，但未进行过测试。
10. 当电压低于 V_{IODIS} 时， \overline{HSB} 没有被定义。

开关波形

图 95. 自动存储或加电回读^[11]



注释:

11. 在 V_{CC} 小于 V_{SWITCH} 的情况下, 将在存储、回读过程中忽略读写周期。
12. 在加电和断电期间, 如果通过外部电阻上拉 \overline{HSB} 引脚电平, 那么 \overline{HSB} 会产生瞬时脉冲。
13. 如果最后一次非易失性循环完成后尚未对 SRAM 进行写操作, 则不会发生自动存储或硬件存储操作。

软件控制的存储和回读周期

在工作范围内

参数	说明	最小值	最大值	单位
t_{RECALL}	回读持续时间	—	500	μs
$t_{\text{SS}}^{[14, 15]}$	软序列处理时间	—	500	μs

开关波形

图 96. 软件存储周期^[15]

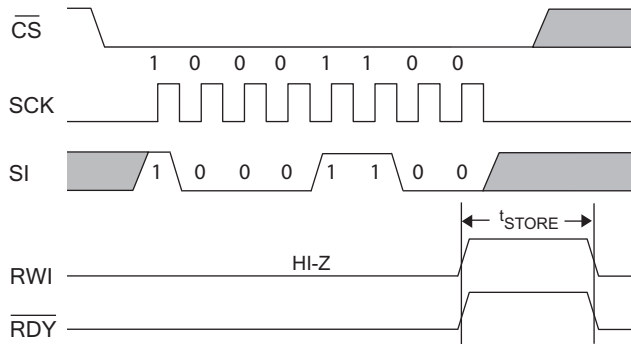


图 97. 软件回读周期^[15]

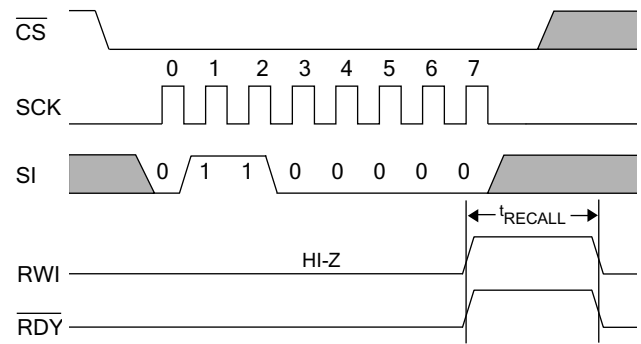


图 98. 自动存储使能周期

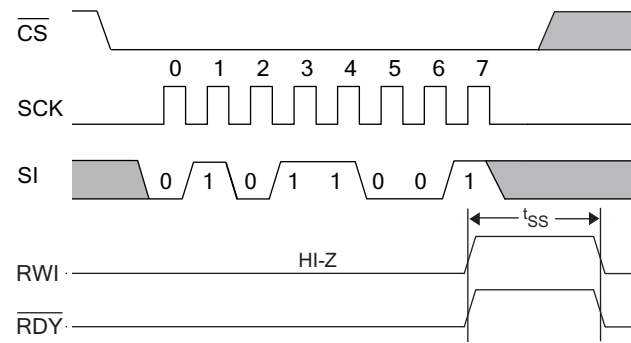
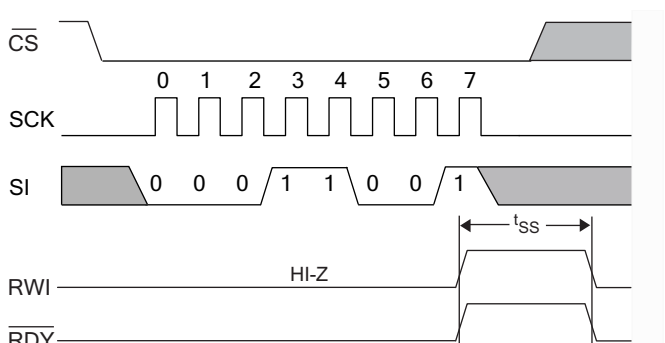


图 99. 自动存储禁用周期



注释:

14. 这是执行软序列指令所耗费的时间。Vcc 电压引脚必须保持为高电平，以保证有效寄存指令。
15. 存储和回读等指令会锁定 I/O，直到操作完成为止，这样可以延长该时间。请参见特定指令。

硬件存储周期

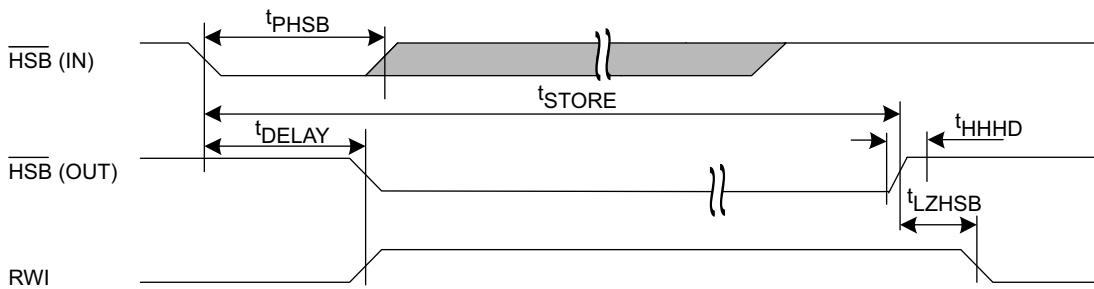
在工作范围内

参数	说明	最小值	最大值	单位
t_{PHSB}	硬件存储脉冲宽度	15	600	ns

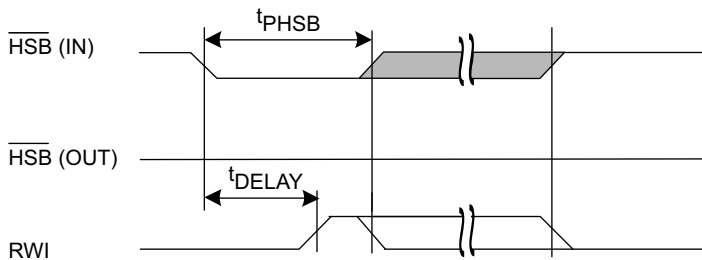
开关波形

图 100. 硬件存储周期^[16]

Write Latch set

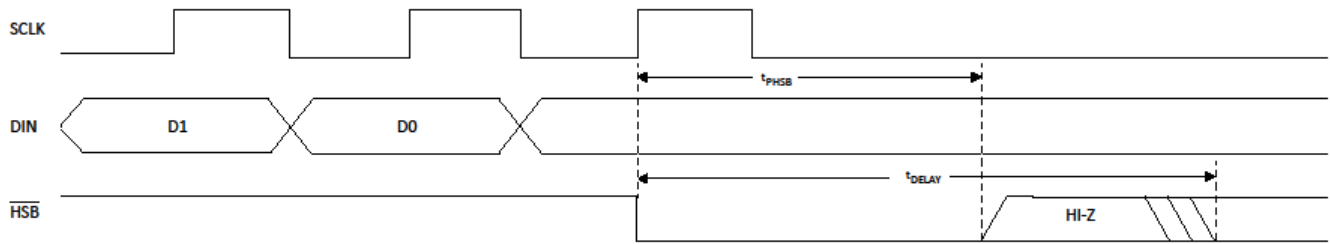


Write Latch not set



HSB pin is driven HIGH to V_{CC} only by Internal 100 K Ω resistor, HSB driver is disabled
SRAM is disabled as long as HSB (IN) is driven LOW.

图 101. 从数据有效到 HSB 的时间



注释:

16. 如果最后一个非易失性存储器写周期完成后尚未对 SRAM 进行写操作，将不会启动自动存储或硬件存储操作。

订购信息

订购代码	封装图	封装类型, 引脚分布	工作范围
CY14V101QS-BK108XI	001-97209	24-FBGA, 标准	工业级直流特性
CY14V101QS-BK108XIT			扩展的工业级
CY14V101QS-BK108XQ			
CY14V101QS-BK108XQT			
CY14V101QS-SE108XI	51-85022	16-SOIC, 自定义	工业级直流特性
CY14V101QS-SE108XIT			扩展的工业级
CY14V101QS-SE108XQ			
CY14V101QS-SE108XQT			
CY14V101QS-SF108XI		16-SOIC, 标准	工业级的直流特性
CY14V101QS-SF108XIT			扩展的工业级
CY14V101QS-SF108XQ			
CY14V101QS-SF108XQT			

这些器件都不含铅。想要了解这些芯片的供应情况, 请联系赛普拉斯本地销售代表。

订购代码定义

CY 14 V 101 QS - SF 108 X I T

Option:	T – Tape and Reel, Blank - Std.
Temperature:	I - Industrial, Q - Extended Industrial
Pb-free	
Frequency:	108 - 108 MHz
Package:	SF - 16 SOIC Standard, SE - 16 SOIC Custom, BK - 24 FBGA
QS - Quad SPI, PS - Quad SPI with RTC	
Density:	101 - 1-Mbit
Voltage:	V - 3.0 V, 1.8 V I/O
14 - nvSRAM	
CY - Cypress	

封装图

图 102. 16-SOIC (0.413 × 0.299 × 0.0932 英寸) 封装外形, 51-85022

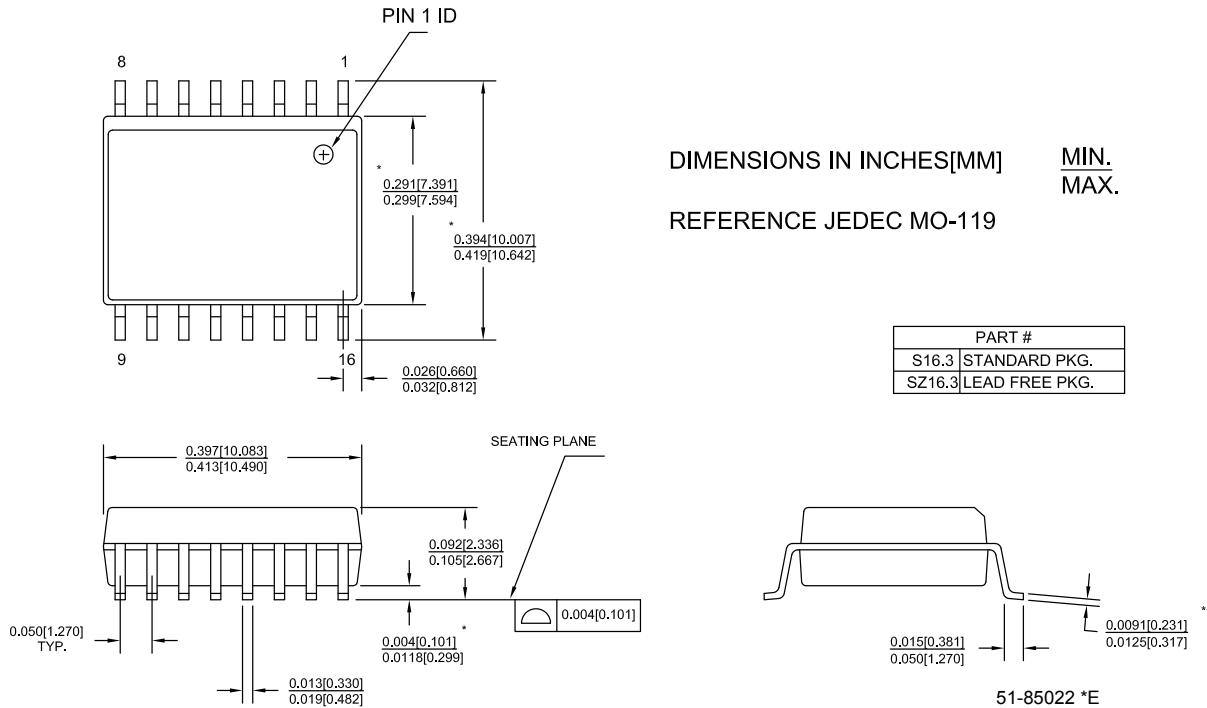
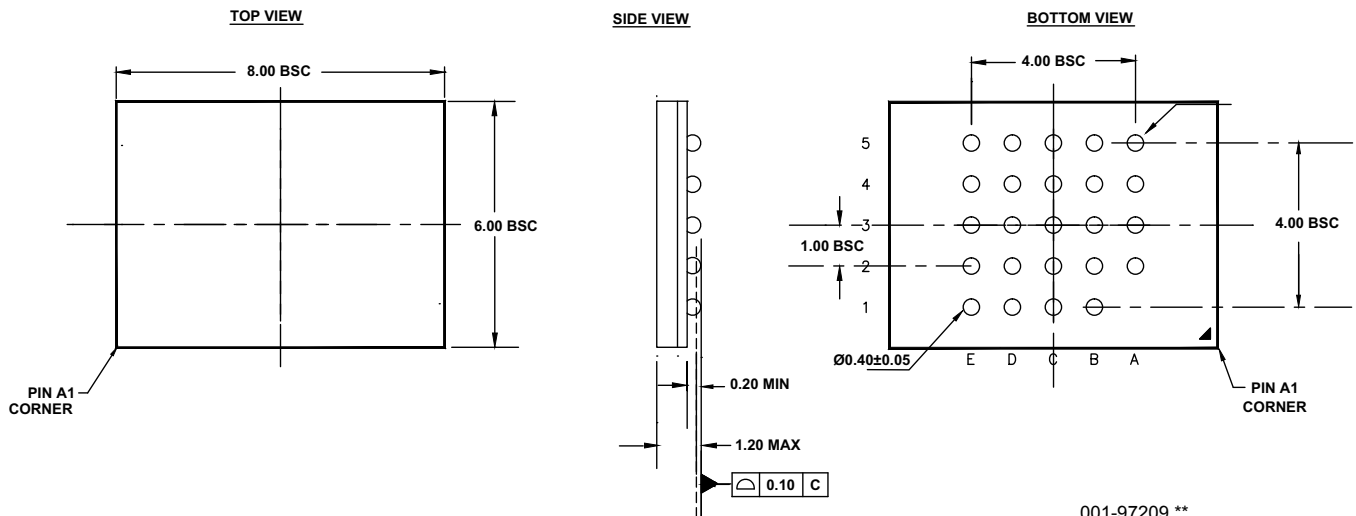


图 103. 24-FBGA 封装



缩略语

缩略语	说明
CPHA	时钟相位
CPOL	时钟极性
CMOS	互补金属氧化物半导体
CRC	循环冗余校验
EEPROM	电可擦除可编程只读存储器
EIA	电子工业联盟
I/O	输入 / 输出
JEDEC	联合电子设备工程委员会
LSB	最低有效位
MSB	最高有效位
nvSRAM	非易失性静态随机存取存储器
RWI	禁止读和写
RoHS	有害物质限制
SNL	序列号锁存
SPI	串行外设接口
SONOS	硅 - 氧化 - 氮化 - 氧化半导体
SOIC	小型塑封集成电路
SRAM	静态随机存取存储器

文档常规
测量单位

符号	测量单位
°C	摄氏度
Hz	赫兹
kHz	千赫兹
kΩ	千欧
Mbit	兆位
MHz	兆赫兹
μA	微安
μF	微法
μs	微秒
mA	毫安
ms	毫秒
ns	纳秒
Ω	欧姆
%	百分比
pF	皮法
V	伏特
W	瓦特

文档修订记录

文档标题: CY14V101QS — 1 Mbit (128K × 8) 四线 SPI nvSRAM 文档编号: 001-96330				
版本	ECN 编号	变更者	提交日期	变更说明
**	4669737	YLIU	03/25/2015	本文档版本号为 Rev**, 译自英文版 001-85257 Rev *C。
*A	4802528	YLIU	07/09/2015	本文档版本号为 Rev*A, 译自英文版 001-85257 Rev *F。
*B	5094030	YLIU	01/25/2016	本文档版本号为 Rev*B, 译自英文版 001-85257 Rev *J。

销售、解决方案和法律信息

全球销售和设计支持

赛普拉斯公司具有一个由办事处、解决方案中心、厂商代表和经销商组成的全球性网络。要想找到离您最近的办事处，请访问[赛普拉斯所在地](#)。

产品

汽车级产品.....[cypress.com/go/automotive](#)
时钟与缓冲区.....[cypress.com/go/clocks](#)
接口.....[cypress.com/go/interface](#)
照明与电源控制.....[cypress.com/go/powerpsoc](#)
存储器.....[cypress.com/go/memory](#)
PSoC.....[cypress.com/go/psoc](#)
触摸感应产品.....[cypress.com/go/touch](#)
USB 控制器.....[cypress.com/go/USB](#)
无线 / 射频.....[cypress.com/go/wireless](#)

PSoC® 解决方案

[psoc.cypress.com/solutions](#)
PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

赛普拉斯开发者社区

[社区](#) | [论坛](#) | [博客](#) | [视频](#) | [培训](#)

技术支持

[cypress.com/go/support](#)

© 赛普拉斯半导体公司，2014-2016。此处所包含的信息可随时更改，恕不另行通知。除赛普拉斯产品内嵌的电路外，赛普拉斯半导体公司不对任何其他电路的使用承担任何责任。也不会根据专利权或其他权利以明示或暗示方式授予任何许可。除非与赛普拉斯签订明确的书面协议，否则赛普拉斯不保证产品能够用于或适用于医疗、生命支持、救生、关键控制或安全应用领域。此外，对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

所有源代码（软件和/或固件）均归赛普拉斯半导体公司（赛普拉斯）所有，并受全球专利法规（美国和美国以外的专利法规）、美国版权法以及国际条约规定的保护和约束。赛普拉斯据此向获许可者授予适用于个人的、非独占性、不可转让的许可，用以复制、使用、修改、创建赛普拉斯源代码的派生作品、编译赛普拉斯源代码和派生作品，并且其目的只能是创建自定义软件和/或固件，以支持获许可者仅将其获得的产品依照适用协议规定的方式与赛普拉斯集成电路配合使用。除上述指定的用途外，未经赛普拉斯明确的书面许可，不得对此类源代码进行任何复制、修改、转换、编译或演示。

免责声明：赛普拉斯不针对此材料提供任何类型的明示或暗示保证，包括（但不限于）针对特定用途的适销性和适用性的暗示保证。赛普拉斯保留在不做出通知的情况下对此处所述材料进行更改的权利。赛普拉斯不对此处所述之任何产品或电路的应用或使用承担任何责任。对于可能发生运转异常和故障，并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

产品使用可能受赛普拉斯相应软件许可协议的限制。