



**深圳市雅创芯瀚电子科技有限公司**  
**SHENZHEN ASTRONG-TECH CO., LTD**

**AST25QW512S 型 512Mbit 宽电压 SPI  
FLASH 存储器数据手册**

**服务电话：13538015750 13691641629**

# 目录

<b>1</b>	<b>产品简介.....</b>	<b>1</b>
1.1	概述.....	1
1.2	产品特性.....	1
1.3	引脚排布和说明.....	2
<b>2</b>	<b>功能描述.....</b>	<b>3</b>
2.1	寄存器.....	3
2.2	指令说明.....	6
<b>3</b>	<b>电气特性.....</b>	<b>28</b>
3.1	绝对最大额定值.....	28
3.2	推荐工作条件.....	28
3.3	电特性表.....	28
<b>4</b>	<b>说明事项.....</b>	<b>31</b>
4.1	运输与储存.....	31
4.2	开箱与检查.....	31
4.3	使用操作规程及注意事项.....	31
4.4	质量保证.....	31
<b>5</b>	<b>封装尺寸.....</b>	<b>32</b>
<b>6</b>	<b>订货信息.....</b>	<b>33</b>
6.1	选型列表.....	33

# 1 产品简介

## 1.1 概述

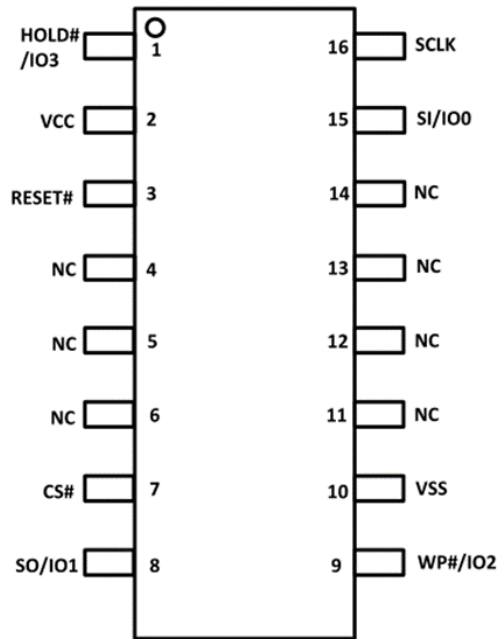
AST25QW512S 是一款容量为 512Mb 的 SPI NOR Flash 存储器，其工作电压为 1.65~3.6V。芯片支持标准 SPI、双 SPI 和四 SPI 指令，最大工作频率可达 90MHz，可通过双 SPI、四 SPI 指令进一步提高数据输出带宽。芯片支持 4KB 扇区、32KB 扇区、64KB 扇区和全芯片擦除指令，通过页编程指令可进行 1~256 字节数据编程，在芯片工作过程中，可通过读取状态寄存器的值来查询芯片工作状态。

AST25QW512S 兼容华邦公司的 W25Q512JV 同封装形式的芯片。

## 1.2 产品特性

- 工作电压：1.65 V~3.6V
- 工作温度：-55℃~+105℃
- 支持标准 SPI/双 SPI/四 SPI 指令
- 支持 4 字节地址指令和 4 字节地址模式
- 灵活的扇区结构(4KB/32KB/64KB)
- 支持 1~256 字节大小的页编程
- 用户可配置读等待周期
- 256 字节页编程时间：0.3ms（典型值）
- 4KB 扇区擦除时间：65ms（典型值）
- 32KB 扇区擦除时间：380ms（典型值）
- 64KB 扇区擦除时间：520ms（典型值）
- 全芯片擦除时间：150s（典型值）
- 擦/写循环次数：10 万次
- 数据保持时间：20 年
- 支持 SOP16 封装形式
- 质量等级：参考 GJB7400《合格制造厂认证用半导体集成电路通用规范》中的 N1 级要求

### 1.3 引脚排布和说明



SOP16 管脚图（顶视图）

引脚名称	引脚类型	引脚功能描述
CS#	输入	片选信号，低电平有效
SCLK	输入	串行时钟
SI / IO0	双向	SPI: 输入；双 SPI: 双向；四 SPI: 双向
SO / IO1	双向	SPI: 输出；双 SPI: 双向；四 SPI: 双向
WP# / IO2	双向	SPI: 输入；双 SPI: 输入；四 SPI: 双向
HOLD# / IO3	双向	SPI: 输入；双 SPI: 输入；四 SPI: 双向
RESET#	输入	复位引脚
VCC	—	1.65~3.6 电源
VSS	—	地
NC	—	内部无连接

图 1-1 引脚排布和说明

## 2 功能描述

### 2.1 寄存器

芯片提供了 4 个寄存器供用户访问，用户可以从这些寄存器中读出芯片的状态信息，也可以向这些寄存器中写入数据以控制芯片的功能，下面分别介绍每个寄存器的含义及用法。

#### 2.1.1 状态寄存器 1

7	6	5	4	3	2	1	0
SRP	TB	BP3	BP2	BP1	BP0	WEL	WIP

**WIP(只读):** 状态寄存器 1 的 WIP 位是只读位，表示芯片是否正在进行内部写操作。如果用户读出的 WIP 数据为 1，表示芯片正在进行编程、擦除或写寄存器等操作。在此期间，用户写入的所有指令（“读状态寄存器”指令除外）都将被芯片忽略。当内部写操作完成后，WIP 位将变为 0，用户可通过查询该位的状态来判断写操作是否完成。

**WEL(只读):** 状态寄存器 1 的 WEL 位是只读位，当上电复位或软件复位完成后，WEL 位为 0。当芯片成功执行“写使能”(06h)指令后，WEL 位将会被设置为 1。为了防止意外改写芯片中存储的数据，在 WEL 为 0 时用户输入的编程、擦除和写寄存器指令都将被芯片忽略。因此用户在输入相应指令前，应该先使用“写使能”指令将 WEL 设置为 1。当编程/擦除/写寄存器指令正确完成后，WEL 位将被芯片自动设置为 0。

**BP3~BP0, TB:** 状态寄存器 1 的 BP 和 TB 位为非易失位，共同实现对主存储区的写保护，具体功能见下表。

TB	BP[3:0]	写保护扇区 (64KB)	写保护地址范围
X	0000	无	无
0	0001	1023	3FF0000h—3FFFFFFh
0	0010	1022—1023	3FE0000h—3FFFFFFh
0	0011	1020—1023	3FC0000h—3FFFFFFh
0	0100	1016—1023	3F80000h—3FFFFFFh
0	0101	1008—1023	3F00000h—3FFFFFFh
0	0110	992—1023	3E00000h—3FFFFFFh
0	0111	960—1023	3C00000h—3FFFFFFh
0	1000	896—1023	3800000h—3FFFFFFh
0	1001	768—1023	3000000h—3FFFFFFh
0	1010	512—1023	2000000h—3FFFFFFh
1	0001	0	0000000h—00FFFFFFh
1	0010	0—1	0000000h—001FFFFh
1	0011	0—3	0000000h—003FFFFh
1	0100	0—7	0000000h—007FFFFh
1	0101	0—15	0000000h—00FFFFFFh

TB	BP[3:0]	写保护扇区 (64KB)	写保护地址范围
1	0110	0—31	0000000h—01FFFFFFh
1	0111	0—63	0000000h—03FFFFFFh
1	1000	0—127	0000000h—07FFFFFFh
1	1001	0—255	0000000h—0FFFFFFFh
1	1010	0—511	0000000h—1FFFFFFFh
X	1011~1111	0—1023	0000000h—3FFFFFFFh

**SRP:** 状态寄存器 1 的 SRP 位为非易失位, 结合/WP 管脚可实现对状态寄存器 1、2、3 的写保护, 具体功能如下表所示。

SRP	/WP	保护状态	说明
0	X	未保护	可以在 Write Enable (写入启用) 命令 WEL=1 之后写入状态寄存器。(默认)
1	0	写保护	WP#=0, 状态寄存器已锁定, 无法写入。
1	1	未保护	WP#=1, 状态寄存器被解锁, 并且可以在写入启用命令之后写入, WEL=1。

### 2.1.2 状态寄存器 2

7	6	5	4	3	2	1	0
SUS1	WPS	RSV	LB2	LB1	SUS2	QE	ADS

**ADS(只读):** 状态寄存器 2 的 ADS 位为只读位, 用于表示芯片当前的地址模式, 如果 ADS 位为 1, 表示芯片处于 4 字节地址模式; 如果 ADS 位为 0, 表示芯片处于 3 字节地址模式。

**QE:** 状态寄存器 2 的 QE 位为非易失位, 用于使能/WP 和/HOLD 功能。当 QE 为 0 时, /WP 和/HOLD 管脚功能有效; 当 QE 为 1 时 (出厂值) /WP 和/HOLD 功能无效。需要注意的是: 在使用四输出或四 I/O 指令之前, 必须将 QE 设置为 1。

**SUS1~SUS0:** SUS1 和 SUS2 位是状态寄存器 2 中的只读位, 在执行程序/擦除挂起 (75H) 命令之后被设置为 1 (擦除挂起将 SUS1 设置为 1, 而程序挂起将把 SUS2 设置为 1)。SUS1 和 SUS2 位通过编程/擦除恢复 (7AH) 命令、软件复位 (66H+99H) 命令以及断电、通电周期清除为 0。

**LB1~LB0:** LB1、LB2 位是状态寄存器 2 中的非易失性一次性编程 (OTP) 位, 其向安全寄存器提供写保护控制和状态。LB1-LB2 的默认状态为 0, 安全寄存器处于解锁状态。可以使用写寄存器指令将 LB1-LB2 位分别设置为 1。LB1-LB2 位是一次性可编程的, 一旦设置为 1, 相应的安全寄存器将永久变为只读。

**WPS:** WPS 位用于选择应使用的写入保护方案。当 WPS=0 时, 设备将使用 T/B、BP (3:0) 位的组合来保护存储器阵列的特定区域。当 WPS=1 时, 设备将使用单个块锁来保护

任何单个扇区或块。设备通电或复位后，所有单独块锁定位的默认值为 1。注意：当 WPS=1 时，在执行芯片擦除操作之前需要全局块解锁（98H）命令。

**RSV**：保留位，在读状态寄存器 2 时，忽略这些位的值；在写状态寄存器 2 时，请向这些位写入 0。

### 2.1.3 状态寄存器 3

7	6	5	4	3	2	1	0
RSV	DRV1	DRV0	ADP	EE	PE	LC	RSV

**LC**：延迟码（LC）选择命令 EDH 的地址结束和读取数据输出开始之间的模式和等待周期数。一些读取命令在地址之后发送模式位，以指示下一个命令将是具有隐含指令而非显式指令的相同类型。因此，下一个命令不提供指令 Byte，只提供新的地址和模式位。这减少了在一系列命令中重复相同命令类型时发送每个命令所需的时间。等待周期提供了额外的延迟，这是在数据返回到主机系统之前完成闪存阵列的初始读取访问所需的延迟。随着 SCLK 频率的增加，一些读取命令需要额外的延迟周期。

LC	等待周期数
0	8（默认）
1	6

**PE**：编程错误（PE）位是表示编程失败的只读位。当用户试图对受保护的阵列扇区进行编程或访问锁定的 OTP 空间时，也会出现编程错误现象。编程错误位可以通过软件复位命令（66H+99H）进行重置。

**EE**：擦除错误（EE）位是指示擦除失败的只读位。当用户试图擦除受保护的阵列扇区或访问锁定的 OTP 空间时，也会出现擦除错误现象。擦除错误位可以通过软件复位（66H+99H）进行重置。

**ADP**：状态寄存器 3 的 ADP 位为非易失位，用于控制芯片上电复位/软件复位后使用的地址模式。如果 ADP 为 0（出厂模式），那么上电复位/软件复位后芯片将处于 3 字节地址模式；如果 ADP 为 1，那么上电复位/软件复位后芯片将处于 4 字节地址模式。

**DRV1~DRV0**：状态寄存器 3 的 DRV1 和 DRV0 为非易失位，用于控制芯片输出 IO 的驱动能力，具体设置及对应的驱动能力见下表。

DRV1	DRV0	驱动能力
0	0	25%
0	1	50%
1	0	70% (出厂模式)
1	1	100%

**RSV**：保留位，在读状态寄存器 3 时，忽略这些位的值；在写状态寄存器 3 时，请向这些位写入 0。

### 2.1.4 地址扩展寄存器

芯片向用户提供一个 8 位的地址扩展寄存器用于后向兼容部分使用 3 字节地址的指令，寄存器的格式如下所示。

7	6	5	4	3	2	1	0
RSV	RSV	RSV	RSV	RSV	RSV	A25	A24

扩展地址位 A25、A24 仅在设备在 3 字节地址模式（ADS=0）下操作时使用，其可通过写入扩展地址寄存器命令（C5H）进行易失性写入。扩展地址寄存器是一个易失性寄存器，当设备在 3 字节地址模式（ADS=0）下操作时，它存储第 4 字节地址（A31-A24）。位 A31-A26 保留用于更高密度。如果 ADP 位设置为 1 时设备通电，或发出“进入 4 字节地址模式（B7H）”指令，则设备将需要为所有地址相关指令输入 4 字节地址，而扩展地址位设置将被忽略。需要注意的是，所有使用 4 字节地址的指令执行后，地址扩展寄存器都将被指令所使用的地址更新，因此从 4 字节地址模式返回到 3 字节地址模式后，用户需仔细检查地址扩展寄存器是否是自己所期望的值。

### 2.2 指令说明

功能	命令（写）	地址（写）	数据（写）	等待周期	数据（读）
写使能	06h	—	—	—	—
写禁用	04h	—	—	—	—
读状态寄存器 1	05h	—	—	—	1 字节
写状态寄存器 1	01h	—	1 字节	—	—
读状态寄存器 2	35h	—	—	—	1 字节
写状态寄存器 2	31h	—	1 字节	—	—
读状态寄存器 3	15h	—	—	—	1 字节
写状态寄存器 3	11h	—	1 字节	—	—
读地址扩展寄存器	C8h	—	—	—	1 字节
写地址扩展寄存器	C5h	—	1 字节	—	—
进入 4 字节地址模式	B7h	—	—	—	—
退出 4 字节地址模式	E9h	—	—	—	—
软件复位使能	66h	—	—	—	—
软件复位	99h	—	—	—	—
读数据	03h	3/4 字节	—	—	1-∞字节
	13h	4 字节	—	—	1-∞字节
快速读数据	0Bh	3/4 字节	—	8 周期	1-∞字节
	0Ch	4 字节	—	8 周期	1-∞字节
双输出快速读数据	3Bh	3/4 字节	—	8 周期	1-∞字节



功能	命令 (写)	地址 (写)	数据 (写)	等待周期	数据 (读)
四输出快速读数据	3Ch	4 字节	—	8 周期	1-∞字节
	6Bh	3/4 字节	—	8 周期	1-∞字节
双 I/O 快速读数据	6Ch	4 字节	—	8 周期	1-∞字节
	BBh	3/4 字节	—	4 周期	1-∞字节
四 I/O 快速读数据	BCh	4 字节	—	4 周期	1-∞字节
	EBh	3/4 字节	—	6 周期	1-∞字节
四 I/O 快速读数据	ECh	4 字节	—	6 周期	1-∞字节
	02h	3/4 字节	1-256 字节	—	—
单输入页编程	32h	3/4 字节	1-256 字节	—	—
四输入页编程	20h	3/4 字节	—	—	—
4KB 扇区擦除	52H	3/4 字节	—	—	—
32KB 扇区擦除	D8h	3/4 字节	—	—	—
64KB 扇区擦除	60h/C7h	—	—	—	—
全芯片擦除	—	—	—	—	—

### 2.2.1 写使能指令(06h)

写使能命令用于设置写使能锁存 (WEL) 位。写入启用锁存器 (WEL) 位必须在每个页面编程、扇区擦除、块擦除、芯片擦除、擦除安全寄存器、编程安全寄存器和写入状态寄存器命令之前设置。写入启用命令序列: CS#变低→发送写入启用命令→CS#变高。

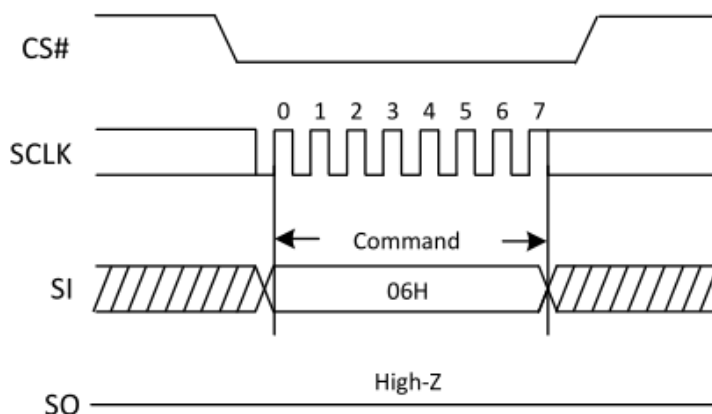


图 2-1 “06h” 指令时序图

### 2.2.2 写禁用指令(04h)

写入禁用命令用于重置写入启用锁存 (WEL) 位。写入禁用命令序列: CS#变低→发送写禁用命令→CS#变高。WEL 位通过以下条件复位: 写禁用命令; 通电; 完成写入状态寄存器、写入扩展地址寄存器、页面程序、四页程序、扇区擦除、块擦除、芯片擦除、擦除/编程安全寄存器和复位命令。

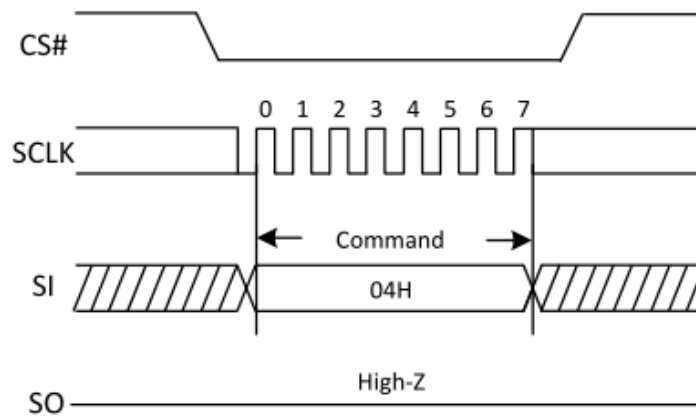


图 2-2 “04h” 指令时序图

### 2.2.3 读状态寄存器(05h/35h/15h)

读取状态寄存器命令用于读取状态寄存器。状态寄存器可以在任何时候读取，即使编程、擦除或写入状态寄存器周期正在进行。当其中一个循环正在进行中，建议在向设备发送新命令之前检查正在写入（WIP）位。也可以连续读取状态寄存器。对于命令代码“05H”，SO 将输出状态寄存器 1 的值。对于命令代码“35H”，SO 将输出状态寄存器 2 的值。对于命令代码“15H”，SO 将输出状态寄存器 3 的值。

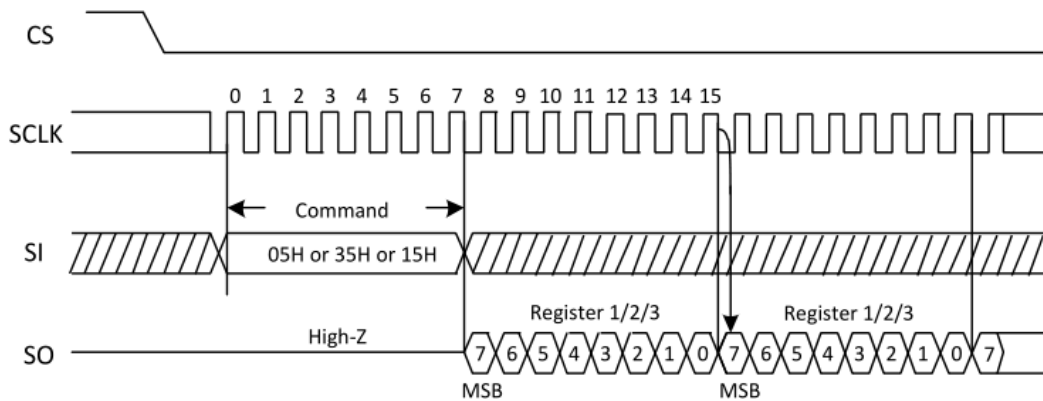


图 2-3 “05h/35h/15h” 指令时序图

### 2.2.4 写状态寄存器(01h/31h/11h)

写入状态寄存器命令允许将新值写入状态寄存器。在被接受之前，必须先执行写入启用命令。在写入使能命令解码并执行后，设备设置写入使能锁存器（WEL）。

写入状态寄存器命令对状态寄存器的易失性位没有影响。在数据字节的第八位被锁存后，CS#必须被驱动为高电平。如果不是，则不执行写入状态寄存器（WRSR）命令。一旦 CS#被驱动为高电平，就启动自定时写入状态寄存器周期（其持续时间为  $t_W$ ）。当写入状态寄存器周期正在进行时，状态寄存器仍可被读取以检查正在写入（WIP）位的值。在自定时写入状

态寄存器周期中，正在写入（WIP）位为 1，完成时为 0。循环完成后，写入启用锁存器（WEL）复位。

写入状态寄存器命令允许用户更改块保护（T/B、BP3、BP2、BP1、BP0）位的值，以定义将被视为只读的区域的大小，如保护区域大小表中所定义的。写入状态寄存器命令还允许用户根据写入保护（WP#）信号设置或重置状态寄存器保护（SRP）位。状态寄存器保护（SRP）位和写入保护（WP#）信号允许设备进入硬件保护模式。一旦进入硬件保护模式，写入状态寄存器命令就不会执行。对于命令代码“01H”，SI 将输入状态寄存器 1。对于命令代码“31H”，SI 将输入状态寄存器 2。对于命令代码“11H”，SI 将输入状态寄存器 3。

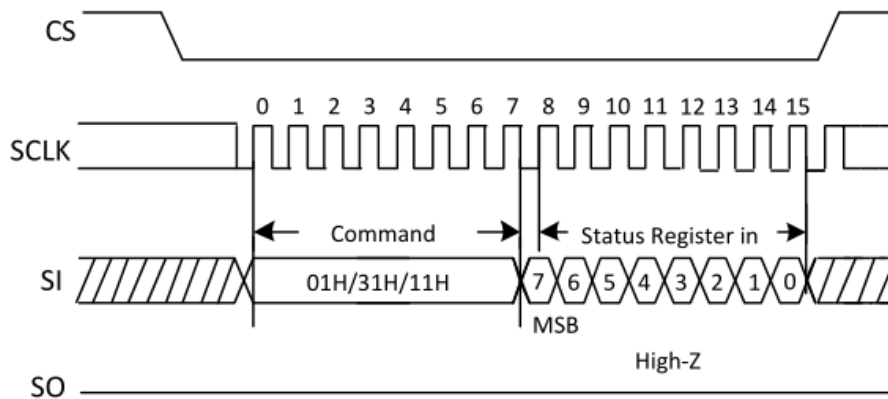


图 2-4 “01h/31h/11h” 指令时序图

### 2.2.5 读地址扩展寄存器(C8h)

扩展地址寄存器包含地址位 A31-A24。读取扩展地址寄存器指令是通过驱动 CS#低电平并将指令代码“C8H”移位到 SCLK 上升沿的 SI 引脚来输入的。然后，扩展地址寄存器位在 SCLK 下降沿的 SO 引脚上移位，最高有效位（MSB）优先。

当设备处于 4 字节地址模式时，A31-A24 位的值被忽略。

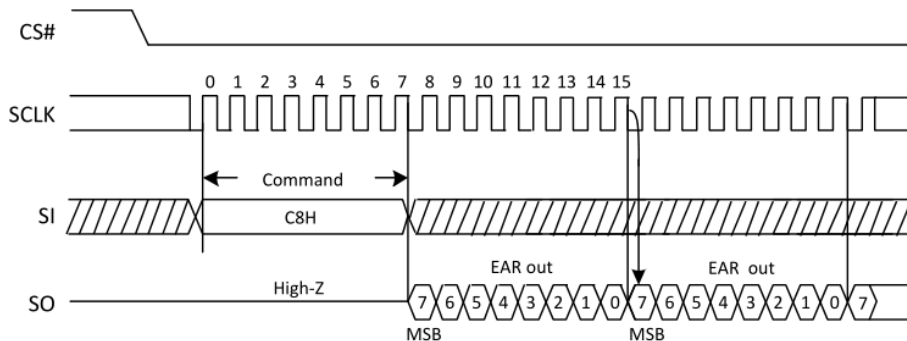


图 2-5 “C8h” 指令时序图

### 2.2.6 写地址扩展寄存器(C5h)

扩展地址寄存器是一个易失性寄存器，当设备在 3 字节地址模式（ADS=0）下操作时，它存储第 4 字节地址（A31-A24）。要写入扩展地址寄存器位，必须事先执行写入启用（06H）指令，设备才能接受写入扩展地址注册指令。写入启用后，通过驱动 CS#低电平，发送指令代码“C5H”，然后写入扩展地址寄存器数据字节来输入指令。通电或执行软件/硬件重置后，扩展地址寄存器位值将被清除为 0。扩展地址寄存器仅在设备处于 3 字节地址模式时有效。当设备在 4 字节地址模式（ADS=1）下操作时，任何地址输入为 A31-A24 的命令都将替换扩展地址寄存器值。当设备从 4 字节地址模式切换到 3 字节地址模式时，如有必要，建议检查并更新扩展地址寄存器。

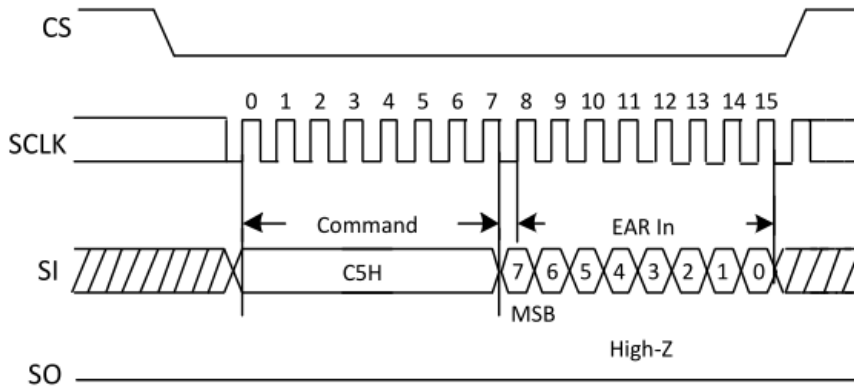


图 2-6 “C5h” 指令时序图

### 2.2.7 进入 4 字节地址模式(B7h)

进入 4 字节模式命令允许访问更高密度（大于 128Mb）的存储区域的 32 位地址长度。设备默认为 24 位地址模式。发送进入 4 字节模式命令后，ADS 位将被设置为 1，表示已进入 4 字节地址模式。一旦进入 4 字节的地址模式，地址长度将变为 32 位，而不是默认的 24 位。退出 4 字节模式命令、复位或掉电将禁用 4 字节模式。

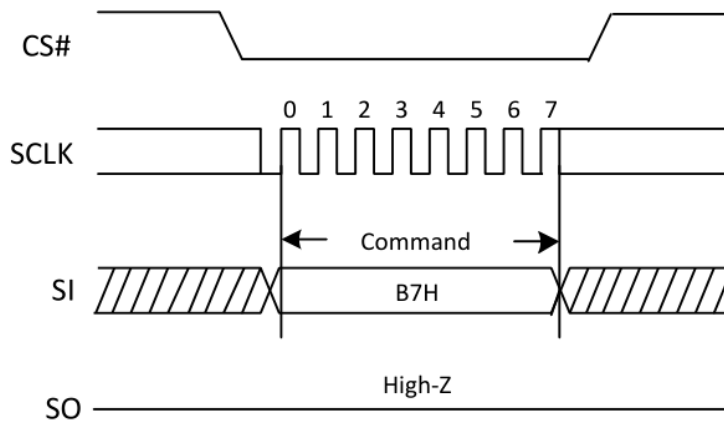


图 2-7 “B7h” 指令时序图

### 2.2.8 退出 4 字节地址模式(E9h)

执行退出 4 字节模式命令以退出 4 字节地址模式并返回默认的 3 字节地址模式。发送退出 4 字节模式命令后，ADS 位将被清除为 0，表示 4 字节地址模式已退出，然后地址长度将返回到 24 位。

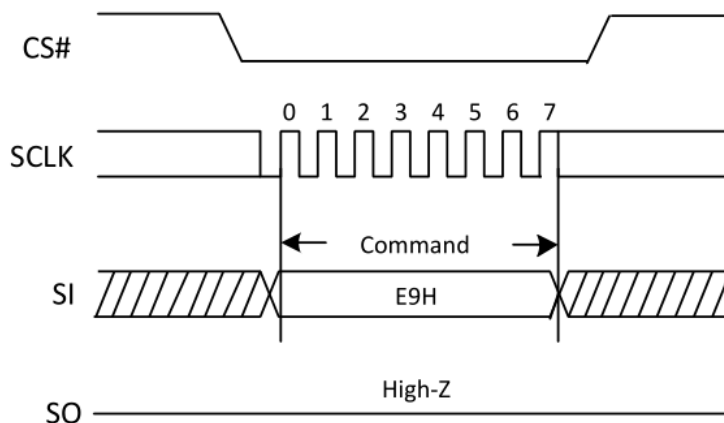


图 2-8 “E9h” 指令时序图

### 2.2.9 软件复位(66h+99h)

如果复位命令被接受，任何正在进行的内部操作都将被终止，设备将返回到其默认通电状态，并丢失所有当前的易失性设置，如易失性状态寄存器位、写使能锁存状态（WEL）、编程/擦除错误位。“复位（99H）”命令序列如下：CS#变低→发送启用复位命令→CS#变高→CS#变低→发送复位命令→CS#变高。一旦设备接受重置命令，设备将花费大约  $t_{RST\_R}$  进行重置。在此期间，将不接受任何命令。当设备接受重置命令序列时，如果正在进行内部擦除或编程操作，则可能发生数据损坏。建议在发出重置命令序列之前检查状态寄存器中的 WIP 位和 SUS 位。必须在发出重置（99H）命令之前发出启用重置（66H）命令，并且不能在它们之间插入任何其他命令。否则，启用复位（66H）命令将被清除。

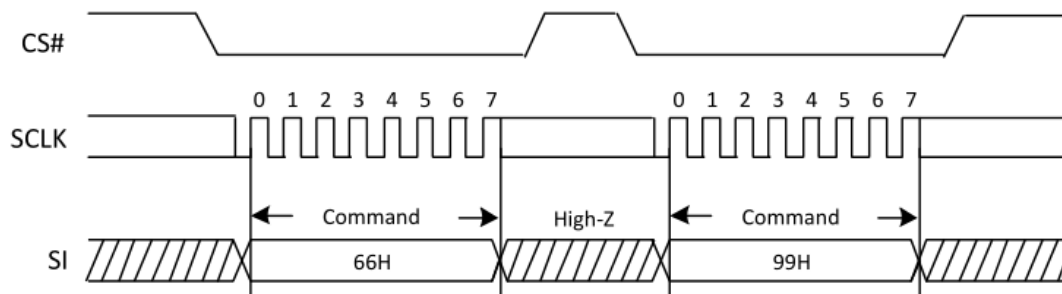


图 2-9 软件复位时序图

### 2.2.10 三字节地址读数据(03h)

读取数据字节命令后面跟着 3 字节地址 (A23-A0) 或 4 字节地址 (A31-A0)，并且每个位在 SCLK 的上升沿被锁存。然后，在该地址处的存储器内容在 SO 上被移出，并且每个比特在 SCLK 的下降沿以最大频率 fR 被移出。被寻址的第一个字节可以在任何位置。在每个字节的数据移出后，地址自动递增到下一个更高的地址。因此，整个内存可以通过单个读取数据字节命令读取。擦除、编程或写入周期正在进行时，任何读取数据字节命令都会被拒绝，而不会对正在进行的周期产生任何影响。

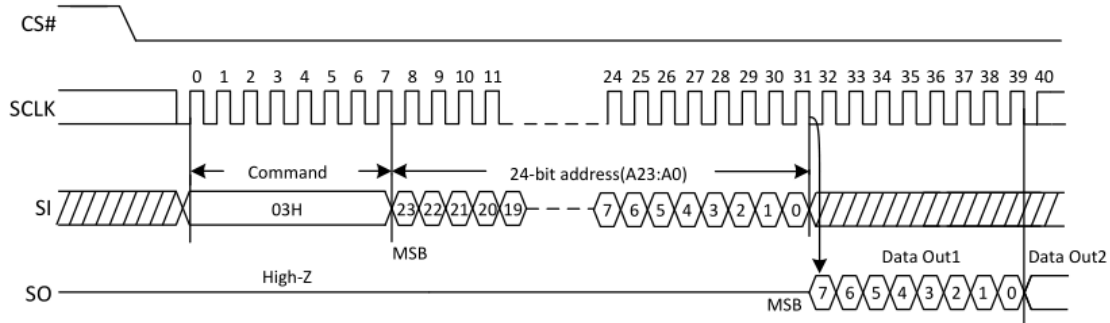


图 2-10 “03h” 指令时序图(3 字节地址模式)

### 2.2.11 四字节地址读数据(13h)

为了方便访问 512Mb 存储空间，芯片向用户提供的“13h”指令可以在不需要地址扩展寄存器的情况下从芯片的主存储区连续读取一个或多个字节的数据。本指令和“03h”指令类似，区别在于无论芯片当前处于哪种地址模式，指令码后都必须使用 4 字节地址，且地址扩展寄存器中的数据将被忽略。指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号的上升沿依次将指令码 “13h”和“4 字节地址”从 SI 输入芯片，当地址的最后一个比特在 SCLK 信号上升沿被芯片锁存后，输入地址对应处存储的字节数据将在 SCLK 信号下降沿从 SO 输出芯片（高位先输出），当用户读出所需的数据后，可通过将 CS#信号置为高电平完成此次读操作。

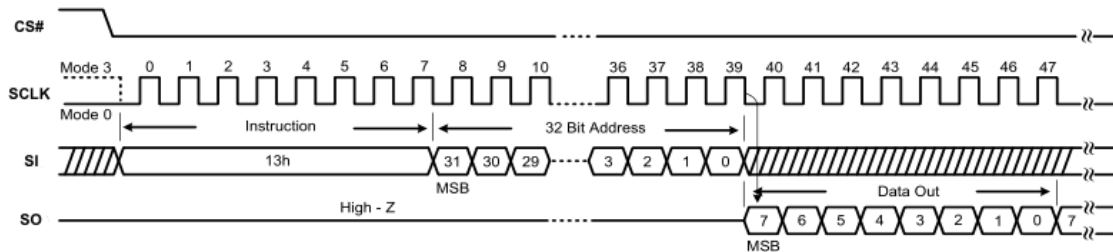


图 2-11 “13h” 指令时序图

### 2.2.12 三字节地址快速读数据(0Bh)

高速读取数据字节（快速读取）命令用于快速读取数据。它后面跟着 3 字节地址（A23-A0）或 4 字节地址（A31-A0）和等待周期，并且每个比特在 SCLK 的上升沿被锁存。然后，在该地址处的存储器数据在 SO 上被移出，并且每个比特在 SCLK 的下降沿上以最大频率 fC 被移出。被寻址的第一个字节可以在任何位置。在每个字节的数据移出后，地址自动递增到下一个更高的地址。

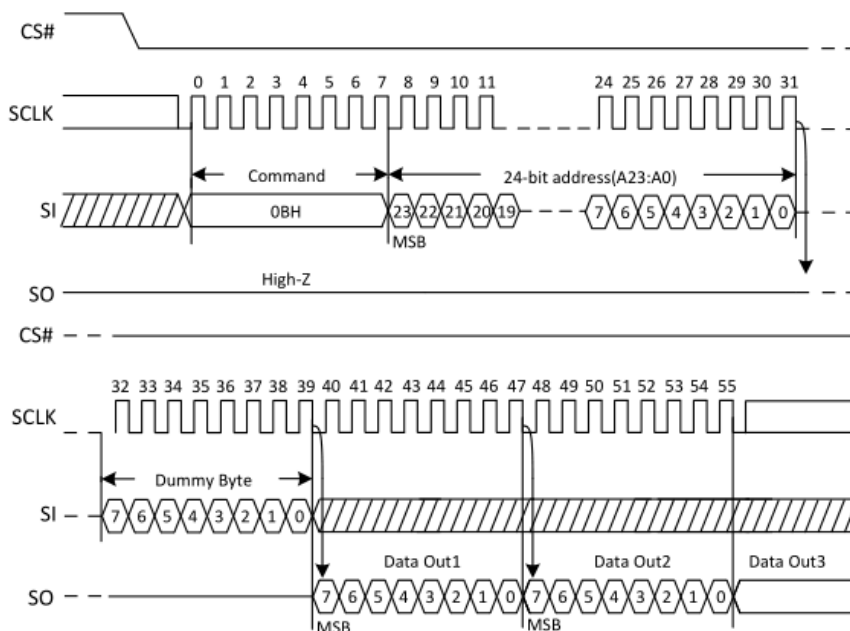


图 2-12 “0Bh” 指令时序图(3 字节地址模式)

### 2.2.13 四字节地址快速读数据(0Ch)

“0Ch”指令是“0Bh”指令的 4 字节地址版本，用户可以在忽略地址扩展寄存器的情况下使用本指令从芯片 512Mb 的主存储区连续读取一个或多个字节的数据。本指令和“0Bh”指令类似，区别在于无论芯片当前处于哪种地址模式，指令码后都必须使用 4 字节地址，且地址扩展寄存器中的数据将被忽略。指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号的上升沿依次将指令码“0Ch”和“4 字节地址”从 SI 输入芯片，然后等待 8 个 SCLK 周期后，输入地址对应处存储的字节数据将在 SCLK 信号的下降沿从 SO 输出芯片（高位先输出），当用户读出所需的数据后，可通过将 CS#信号置为高电平完成此次读操作。

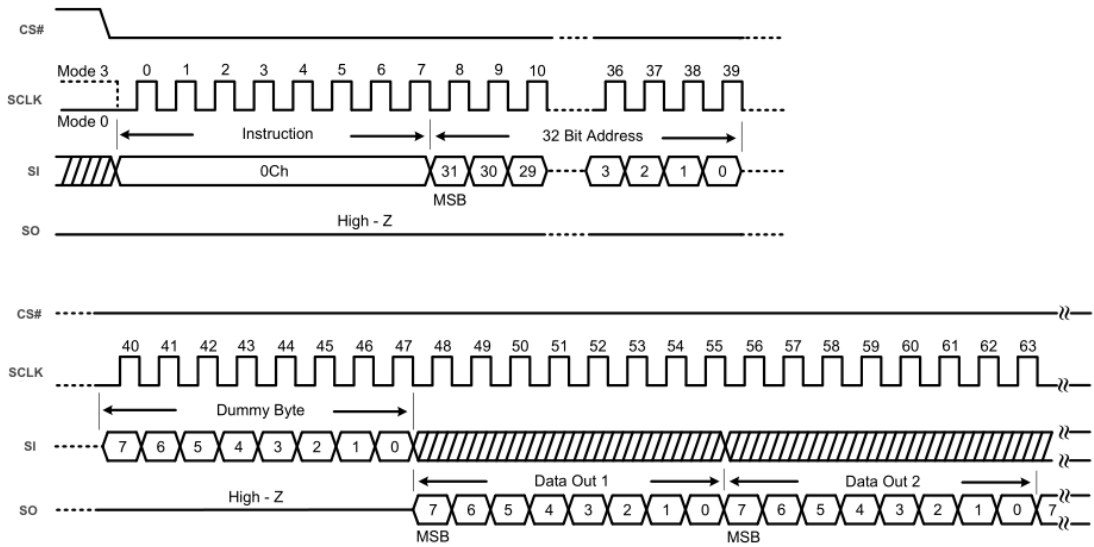


图 2-13 “0Ch” 指令时序图

### 2.2.14 三字节地址双输出快速读数据(3Bh)

双输出快速读取命令后面跟着 3 字节地址 (A23-A0) 或 4 字节地址 (A31-A0) 和等待周期，每个比特在 SCLK 的上升沿期间被锁存，然后存储器内容在每个时钟周期从 SI 和 SO 移出 2 比特。命令序列如下图所示。被寻址的第一个字节可以在任何位置。在每个字节的数据移出后，地址会自动递增到下一个地址。

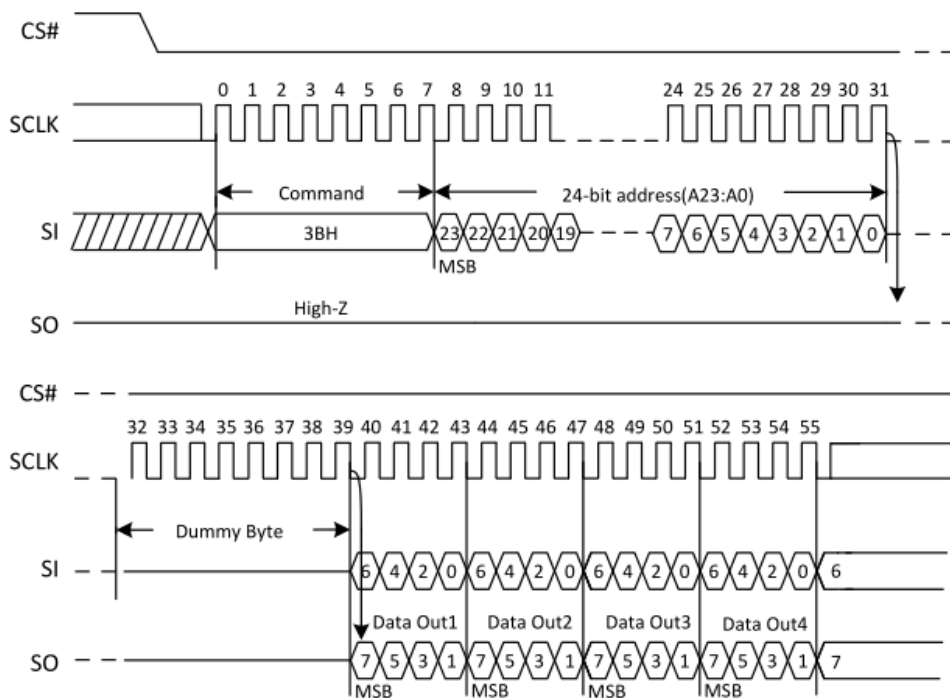


图 2-14 “3Bh” 指令时序图(3 字节地址模式)



### 2.2.15 四字节地址双输出快速读数据(3Ch)

“3Ch”指令是“3Bh”指令的 4 字节地址版本，用户可以在忽略地址扩展寄存器的情况下使用本指令从芯片 512Mb 的主存储区连续读取一个或多个字节的数据。本指令和“3Bh”指令类似，区别在于无论芯片当前处于哪种地址模式，指令码后都必须使用 4 字节地址，地址扩展寄存器中的数据将被忽略。指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号的上升沿依次将指令码“3Ch”和“4 字节地址”从 SI 输入芯片，等待 8 个 SCLK 周期后，输入地址对应处存储的字节数据将在 SCLK 信号的下降沿从 SI 和 SO 输出芯片（高位先输出），当用户读出所需的数据后，可通过将 /CS 信号置为高电平完成此次读操作。

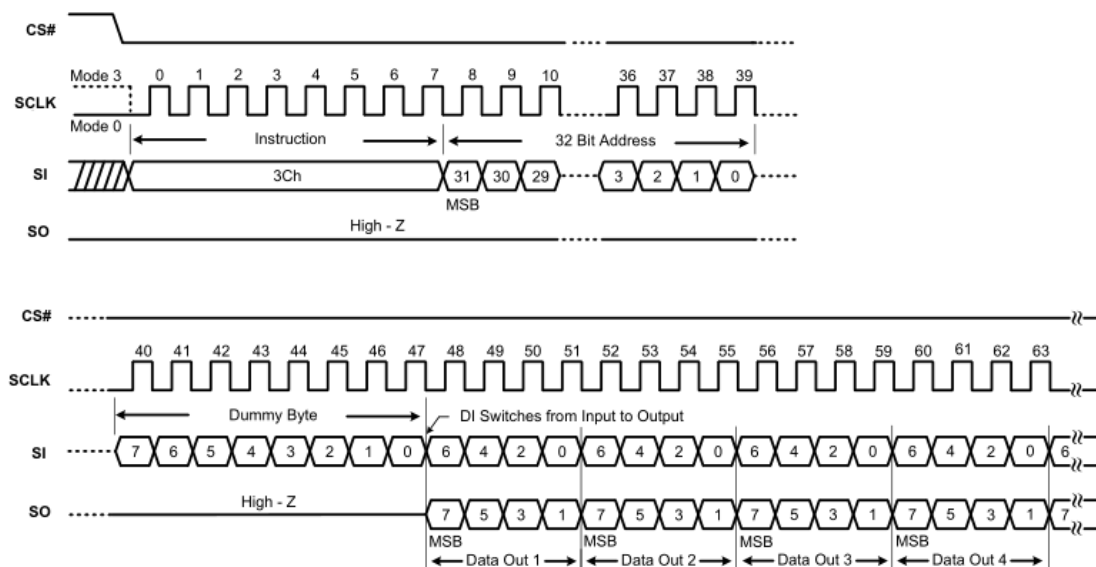


图 2-15 “3Ch” 指令时序图

### 2.2.16 三字节地址四输出快速读数据(6Bh)

四输出快速读取命令后面跟着 3 字节地址（A23-A0）或 4 字节地址（A31-A0）和等待周期，每个比特在 SCLK 的上升沿期间被锁存，然后存储器内容在每个时钟周期从 IO3、IO2、IO1 和 IO0 移出 4 比特。命令序列如下图所示。被寻址的第一个字节可以在任何位置。在每个字节的数据移出后，地址会自动递增到下一个地址。

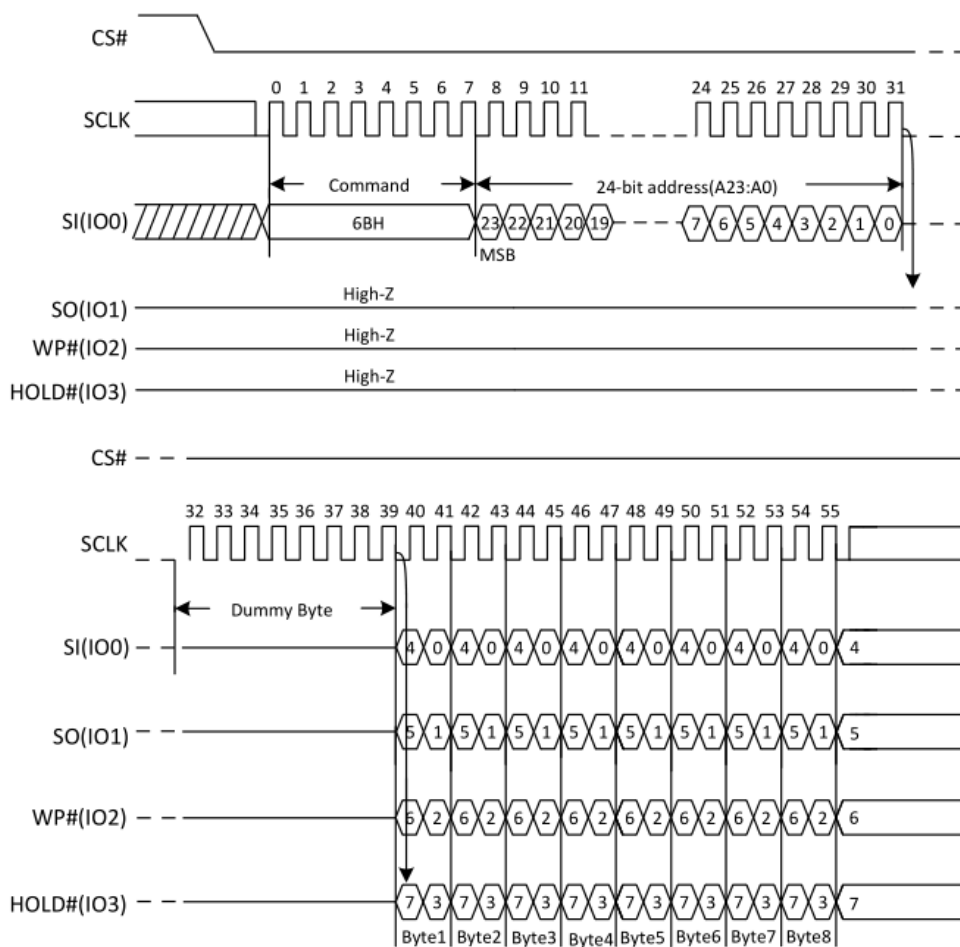


图 2-16 “6Bh” 指令时序图(3 字节地址模式)

### 2.2.17 四字节地址四输出快速读数据(6Ch)

“6Ch”指令是“6Bh”指令的 4 字节地址版本，用户可以在忽略地址扩展寄存器的情况下使用本指令从芯片 512Mb 的主存储区连续读取一个或多个字节的数据。本指令和“6Bh”指令类似，区别在于无论芯片当前处于哪种地址模式，指令码后都必须使用 4 字节地址，且地址扩展寄存器中的数据将被忽略。指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号的上升沿依次将指令码“6Ch”和“4 字节地址”从 SI 输入芯片，然后等待 8 个 SCLK 周期后，输入地址对应处存储的字节数据将在 SCLK 信号的下降沿从 IO3~IO0 输出芯片（高位先输出），当用户读出所需的数据后，可通过将 CS#信号置为高电平完成此次读操作。

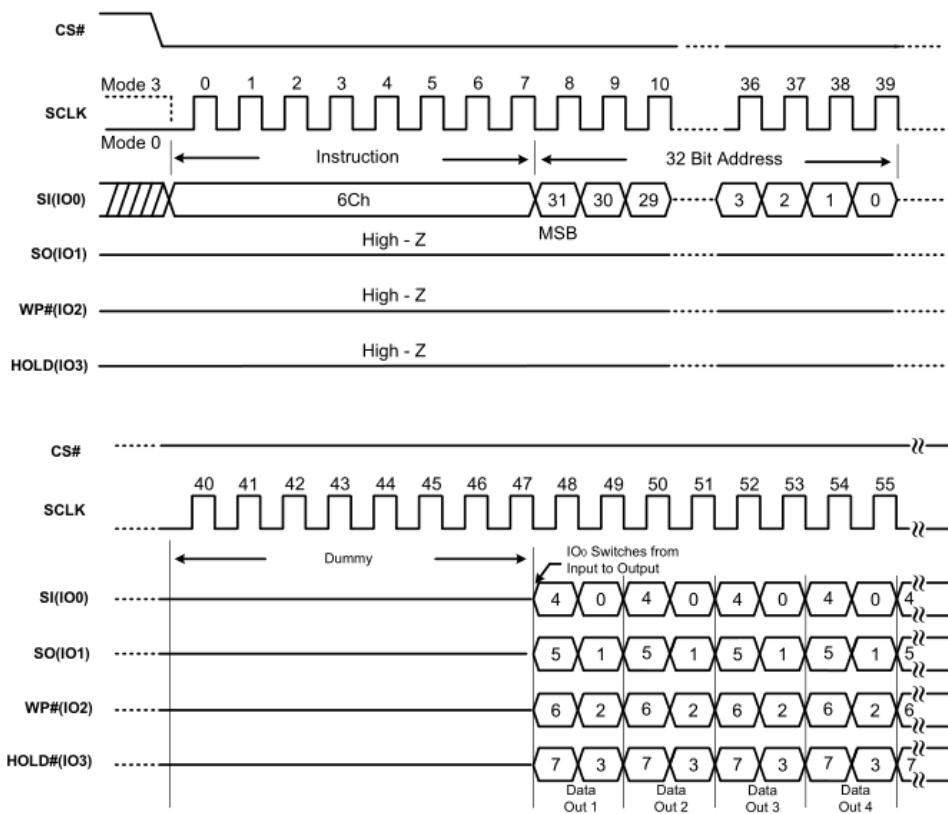


图 2-17 “6Ch” 指令时序图

### 2.2.18 三字节地址双 I/O 快速读数据(BBh)

双 I/O 快速读取命令类似于双输出快速读取命令，但具有通过 SI 和 SO 每个时钟输入 3 字节地址 (A23-0) 或 4 字节地址 (A31-A0) 和“连续读取模式”位 (M7-M0) 的功能，每个位在 SCLK 的上升沿期间被锁存，然后存储器内容从 SI 和 SO 每时钟周期移出 2 位。命令序列如下图所示。被寻址的第一个字节可以在任何位置。在每个字节的数据移出后，地址会自动递增到下一个地址。

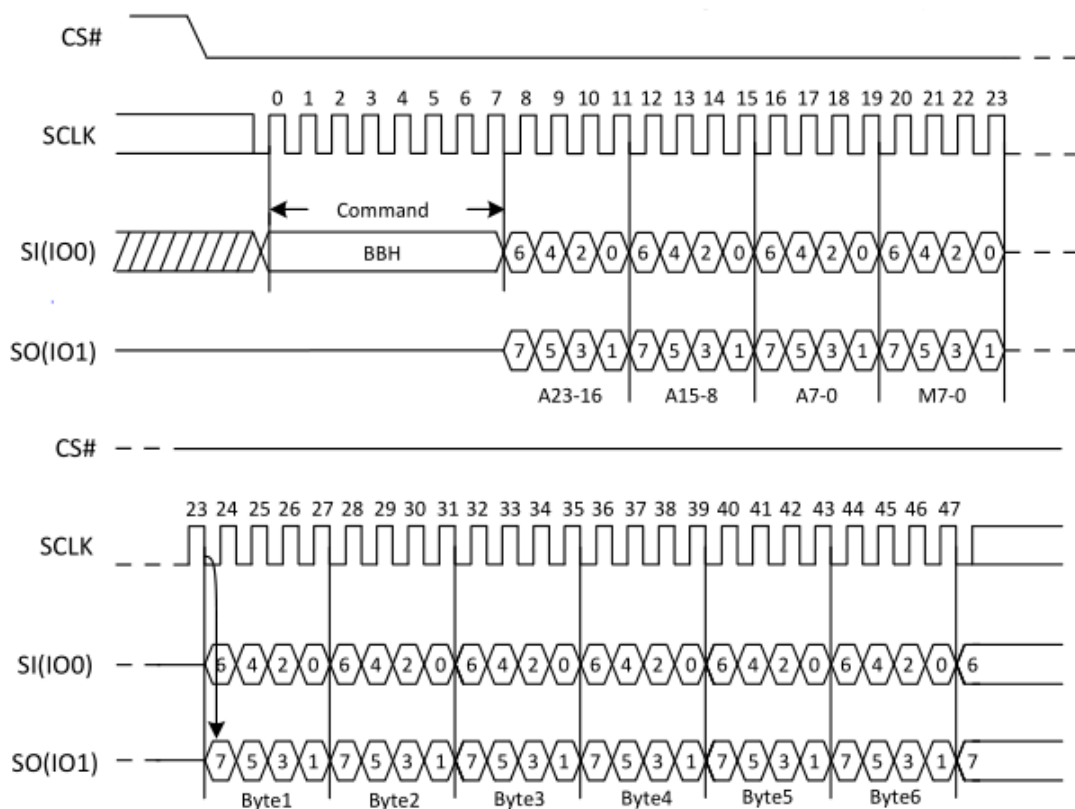


图 2-18 “BBh” 指令时序图（首指令）

双 I/O 快速读取命令可以通过在输入的 3 字节地址（A23-A0）或 4 字节地址（A31-A0）之后设置“连续读取模式”位（M7-0）来进一步减少命令开销。如果“连续读取模式”位（M5-4）=（1，0），则下一个双 I/O 快速读取命令（在 CS# 上升然后下降之后）不需要 BBH 命令代码。命令序列如下图所示。如果“连续读取模式”位（M5-4）不等于（1，0），则下一个命令需要第一个 BBH 命令代码，从而返回正常操作。“连续读取模式”重置命令可用于在发出正常命令之前重置（M5-4）。

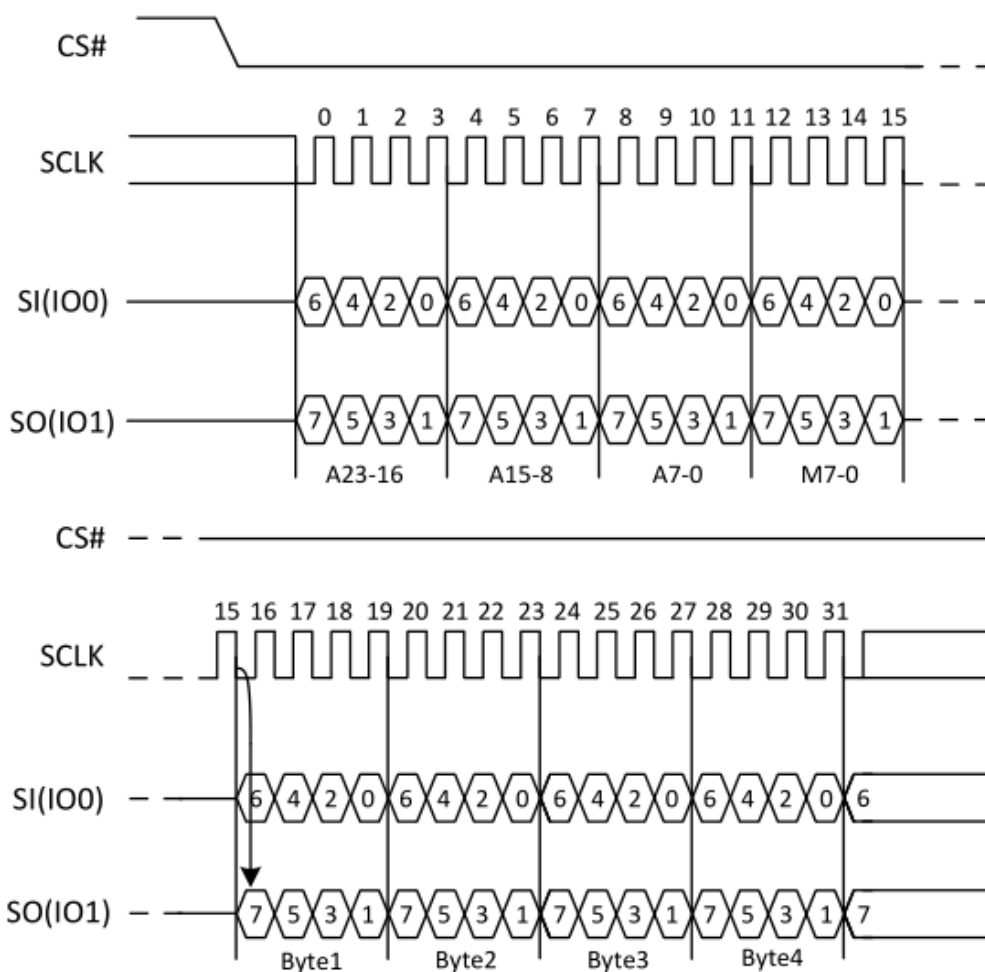


图 2-79 “BBh” 指令时序图（连续读）

### 2.2.19 四字节地址双 I/O 快速读数据(BCh)

“BCh”指令是“BBh”指令的 4 字节地址版本，用户可以在忽略地址扩展寄存器的情况下使用本指令从芯片 512Mb 的主存储区连续读取一个或多个字节的数据。本指令和“BBh”指令类似，区别在于无论芯片当前处于哪种地址模式，指令码后都必须使用 4 字节地址，且地址扩展寄存器中的数据将被忽略。指令时序图如下所示，首先将 CS# 信号置为低电平，然后在 SCLK 信号的上升沿将指令码“BCh”从 IO0 输入芯片，然后将 4 字节地址和 M 码在 SCLK 信号上升沿从 IO1~IO0 输入芯片，然后输入地址对应处存储的字节数据在 SCLK 信号的下降沿从 IO1~IO0 输出芯片（高位先输出），当用户读出所需的数据后，可通过将 CS# 信号置为高电平完成此次读操作。

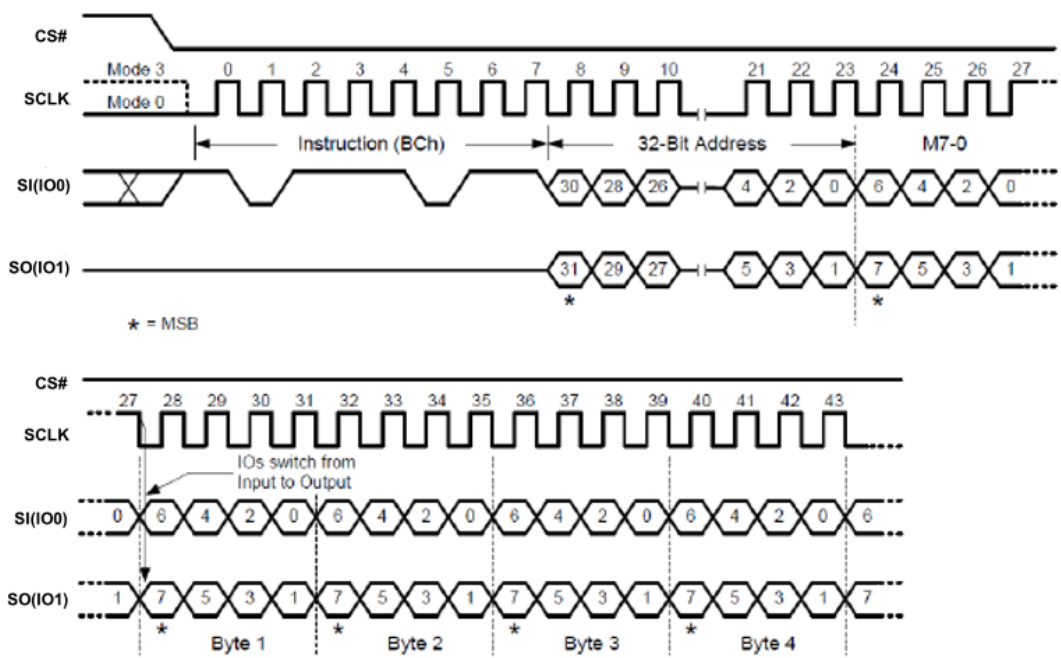


图 2-20 “BCh” 指令时序图（首指令）

“BCh”指令也支持连续读模式，在该模式下后续指令在 CS#变为低电平后无需输入指令码，时序图如下所示。

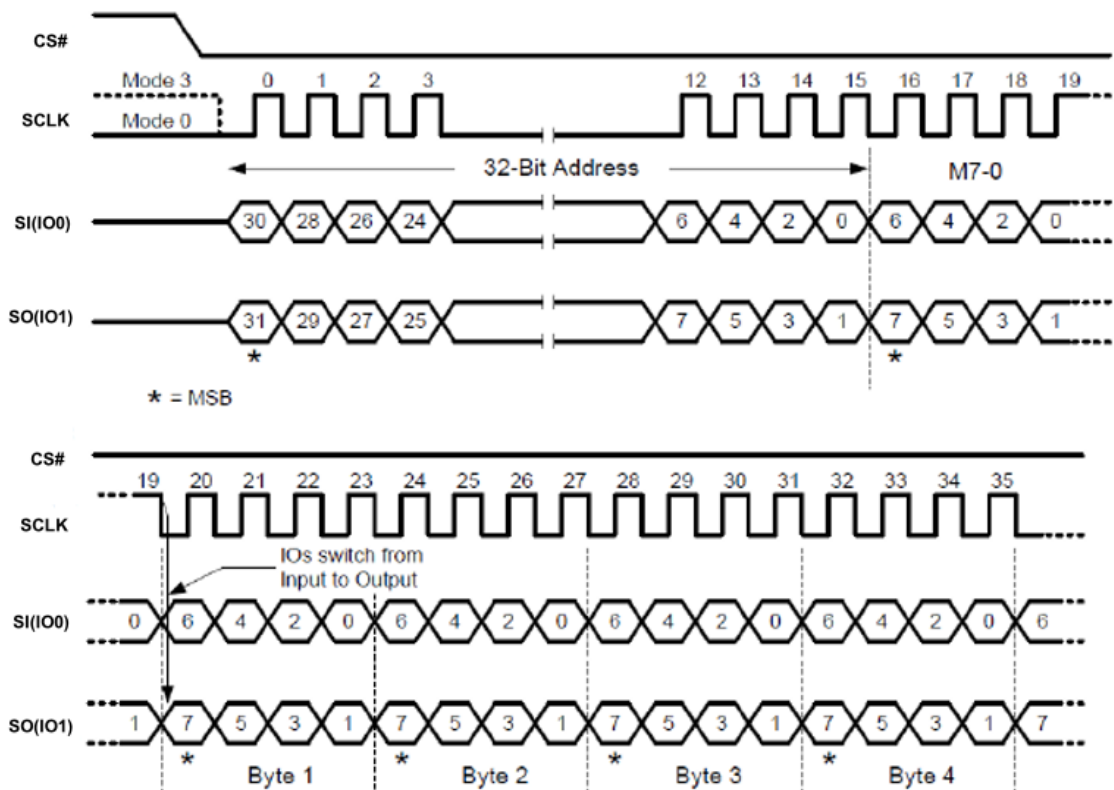


图 2-21 “BCh” 指令时序图（连续读）

### 2.2.20 三字节地址四 I/O 快速读数据(EBh)

为了进一步减小数据输出延迟，用户可以使用“EBh”指令从芯片的主存储区连续读取一个或多个字节的数据，在本指令的地址输入阶段，芯片可在 SCLK 的上升沿可锁存 4 比特数据，在本指令的数据输出阶段，芯片可在 SCLK 的下降沿可输出 4 比特数据，从而可以在保证高输出带宽的情况下进一步减小数据输出延迟；

指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号的上升沿将指令码“EBh”从 IO0 输入芯片，然后在 SCLK 信号的上升沿将“3/4 字节地址”从 IO3~IO0 输入芯片，具体使用 3 字节地址还是 4 字节地址由芯片当前的地址模式决定，然后再在 SCLK 信号的上升沿将 1 字节的“连续读模式码”（简称 M 码）从 IO3~IO0 输入芯片，在 M 码的最后 4 个比特被芯片锁存并且等待了 4 个 SCLK 周期后，输入地址对应处存储的字节数据将在 SCLK 信号的下降沿从 IO3~IO0 输出芯片（高位先输出），当用户读出所需的数据后，可通过将 CS#信号置为高电平完成此次读操作。

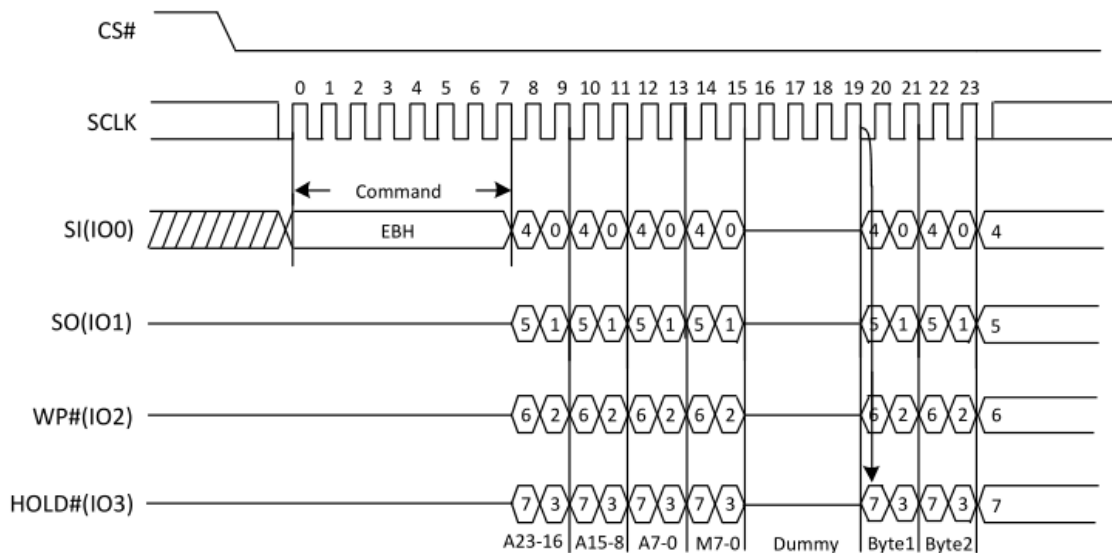


图 2-22 “EBh” 指令时序图（首指令）

使用 M 码可以进一步减小数据输出延迟，M 码共有 8 比特，其中 M[5:4] 被用于控制紧随其后的“EBh”指令（其它位未使用），如果当前“EBh”指令的 M[5:4]为(1,0)，那么紧随当前指令之后的“EBh”指令将不需要输入指令码，也就是当前“EBh”指令完成后 CS#变为高电平，然后 CS#变为低电平，然后直接输入新的地址、新的 M 码和等待周期后芯片即可输出新的数据。相比于正常读，连续读可以减少 8 个指令码输入周期，从而减小数据输出延迟。连续读时序图如下图所示。

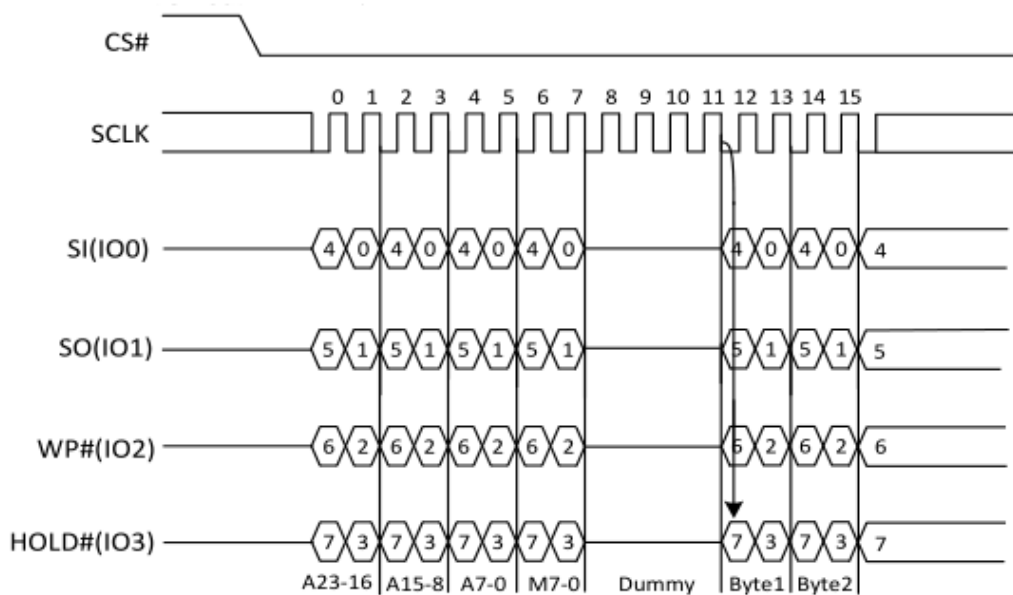


图 2-23 “EBh” 指令时序图（连续读）

### 2.2.21 四字节地址四 I/O 快速读数据(ECh)

“ECh”指令是“EBh”指令的 4 字节地址版本，用户可以在忽略地址扩展寄存器的情况下使用本指令从芯片 512Mb 的主存储区连续读取一个或多个字节的数据。本指令和“EBh”指令类似，区别在于无论芯片当前处于哪种地址模式，指令码后都必须使用 4 字节地址，且地址扩展寄存器中的数据将被忽略。指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号的上升沿将指令码“ECh”从 IO0 输入芯片，然后将“4 字节地址”和 M 码在 SCLK 信号的上升沿从 IO3~IO0 输入芯片，等待 4 个周期后，输入地址对应处存储的字节数据将在 SCLK 信号的下降沿从 IO3~IO0 输出芯片（高位先输出），当用户读出所需的数据后，可通过将 CS#信号置为高电平完成此次读操作。

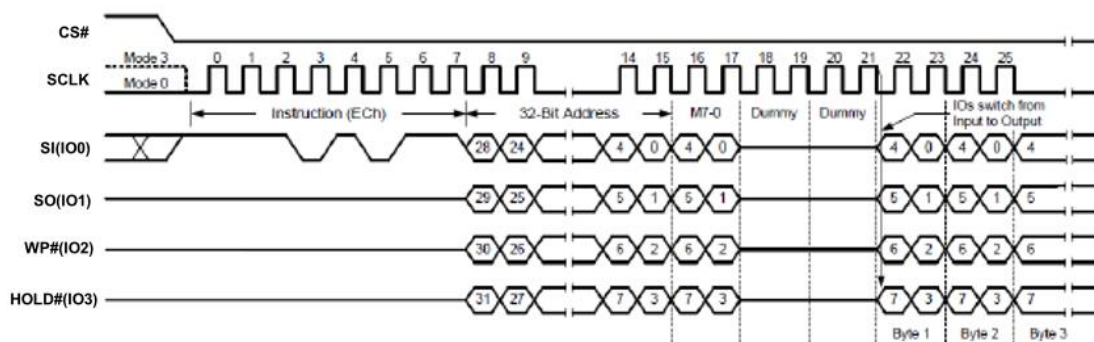


图 2-24 “ECh” 指令时序图（首指令）

“ECh”指令也支持连续读模式，在该模式下后续指令在 CS#变为低电平后无需输入指令码，时序图如下所示。



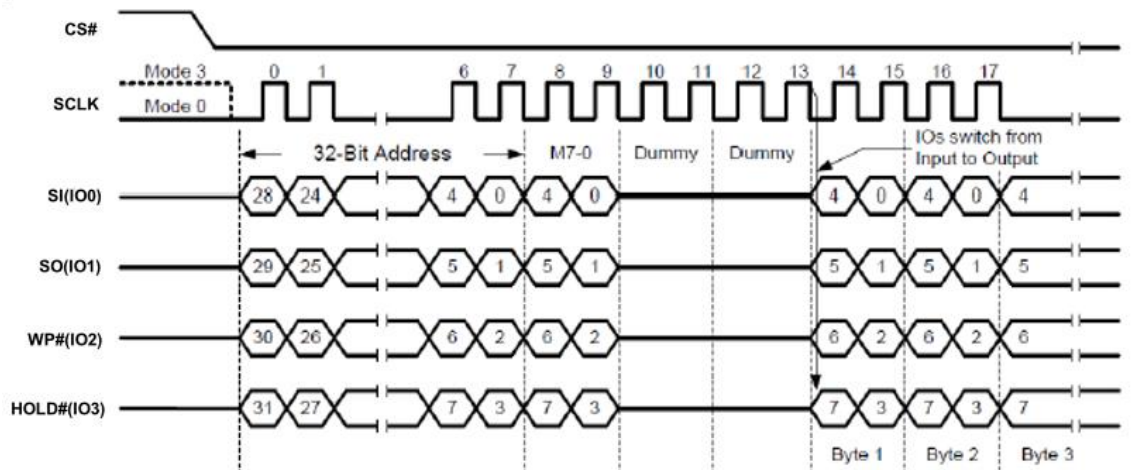


图 2-25 “ECh” 指令时序图（连续读）

### 2.2.22 三字节地址单输入页编程(02h)

页面编程命令用于对内存进行编程。在发送页面程序命令之前，必须预先执行写入启用命令以设置写入启用锁存器（WEL）位。通过驱动 CS#低电平进入页面程序命令，然后是命令代码、三个地址字节和 SI 上的至少一个数据字节。如果 8 个最低有效地址位（A7-A0）不全为零，从同一页的起始地址（从其 8 个最低有效位（A7-A0）全部为零的地址）对超出当前页的末尾的所有发送数据进行编程。CS#必须在整个序列持续时间内被驱动为低电平。页面编程命令序列：CS#变低→发送页面程序命令→ SI 上的 3 字节地址或 4 字节地址→ SI 上至少有 1 个字节的数据→CS#变高。命令序列如下图所示。如果超过 256 个字节被发送到设备，则先前锁存的数据被丢弃，并且最后 256 个数据字节被保证在同一页内被正确编程。如果少于 256 个数据字节被发送到设备，则它们在请求的地址被正确编程，而不会对同一页的其他字节产生任何影响。在最后一个数据字节的第八位已经被锁存之后，CS#必须被驱动为高；否则不执行页面编程命令。一旦 CS#被驱动为高电平，就启动自定时页面编程周期（其持续时间为 tPP）。当页面程序循环正在进行时，可以读取状态寄存器以检查正在写入（WIP）位的值。在自定时的页面程序周期中，正在写入（WIP）位为 1，完成时为 0。在循环完成前的某个未指定时间，写使能锁存（WEL）位被重置。页面编程（PP）命令不会编程受块保护（T/B、BP3、BP2、BP1、BP0）保护的页面。

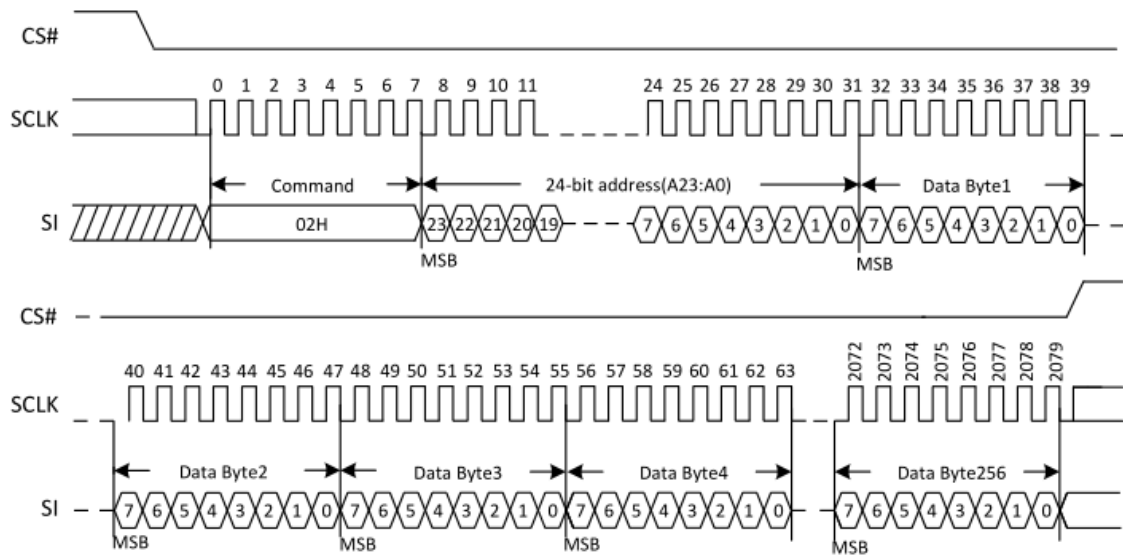


图 2-26 “02h” 指令时序图(3 字节地址模式)

### 2.2.23 三字节地址四输入页编程(32h)

四页编程命令用于使用四个引脚对存储器进行编程：IO0、IO1、IO2 和 IO3。要使用四页编程，必须设置状态寄存器中的四启用位（QE=1）。在发送页面编程命令之前，必须预先执行写入启用命令以设置写入启用锁存（WEL）位。四页编程命令是通过驱动 CS#低电平输入的，后面是命令代码（32H）、三个地址字节和 IO 引脚上的至少一个数据字节。命令序列如下图所示。如果 8 个最低有效地址位（A7-A0）不是全部为零，则从同一页的起始地址（从 8 个最低无效位（A7-A0）全部为零的地址）对超出当前页的末尾的所有传输数据进行编程。如果超过 256 个字节被发送到设备，则先前锁存的数据被丢弃，并且最后 256 个数据字节被保证在同一页内被正确编程。如果少于 256 个数据字节被发送到设备，则它们在请求的地址被正确编程，而不会对同一页的其他字节产生任何影响。在最后一个数据字节的第八位被锁存后，CS#必须被驱动为高电平；否则，四页编程命令将不会执行。一旦 CS#被驱动为高电平，就会启动自定时四页编程周期（其持续时间为 tPP）。当四页编程循环正在进行时，可以读取状态寄存器以检查正在写入（WIP）位。在自定时的四页程序周期中，正在写入（WIP）位为 1，完成时为 0。在循环完成之前的某个未指定时间，写入启用锁存器（WEL）位复位。

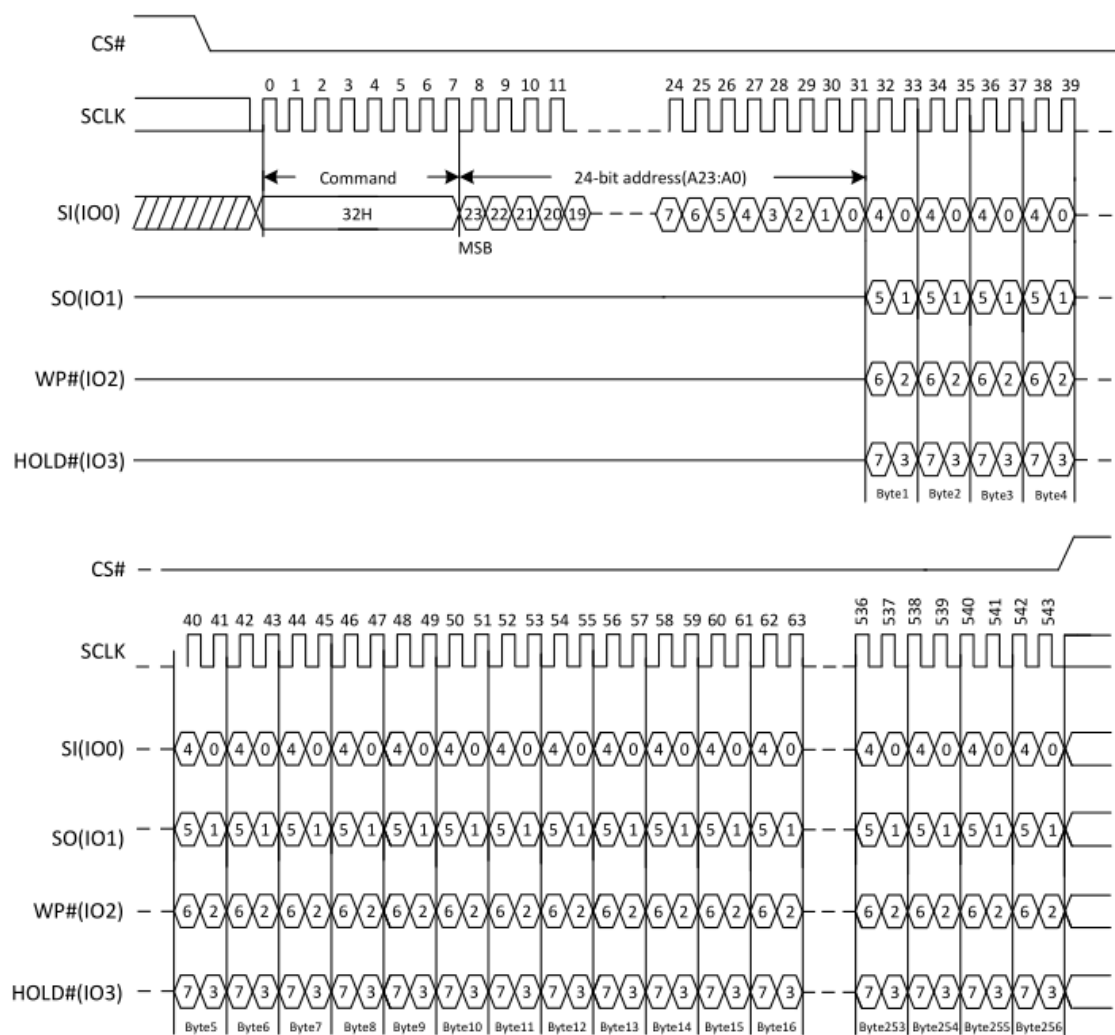


图 2-27 “32h” 指令时序图(3 字节地址模式)

### 2.2.24 三字节地址 4KB 扇区擦除(20h)

扇区擦除命令用于擦除所选扇区的所有数据。在发送扇区擦除命令之前，必须预先执行写入启用命令以设置写入启用锁存（WEL）位。扇区擦除命令是通过将 CS#驱动为低电平，然后是命令代码和 SI 上的 3 地址字节或 4 字节地址来输入的。扇区内的任何地址都是扇区擦除（CE）命令的有效地址。扇区擦除命令序列：CS#变为低电平→发送扇区擦除命令→SI 上的 3 字节地址或 4 字节地址→CS#变高。命令序列如下图所示。在最后一个地址字节的第八位被锁存之后，CS#必须被驱动为高电平；否则将不执行扇区擦除命令。一旦 CS#被驱动为高电平，就启动自定时扇区擦除周期（其持续时间为 tSE）。当扇区擦除周期正在进行时，可以读取状态寄存器以检查正在写入（WIP）位的值。在自定时扇区擦除周期中，正在写入（WIP）位为 1，完成时为 0。在周期完成之前的某个未指定时间，写使能锁存器（WEL）位被重置。应用于由块保护（T/B, BP3, BP2, BP1, BP0）位保护的扇区的扇区擦除（SE）命令将不会被执行。

注意：擦除操作期间电源中断会导致擦除不完整，因此建议在电源恢复后执行重新擦除。

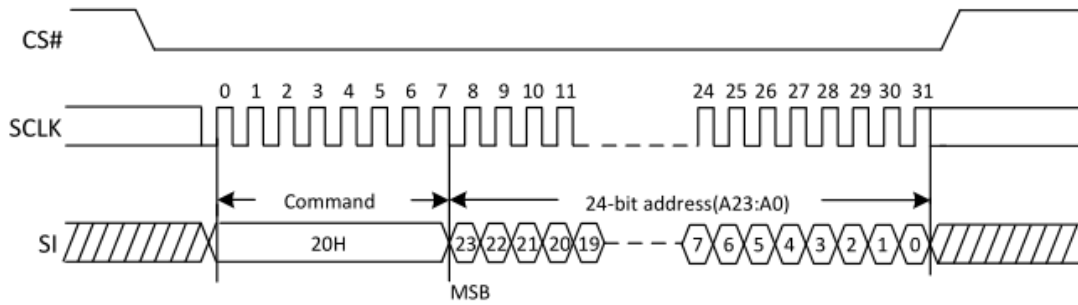


图 2-28 “20h” 指令时序图(3 字节地址模式)

### 2.2.25 三字节地址 32KB 扇区擦除(52h)

用户可以使用“52h”指令将某个 32KB 的扇区擦除。在进行擦除操作之前，用户必须使用“写使能”指令将状态寄存器的 WEL 设置为 1，否则擦除指令将会被芯片忽略。指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号上升沿依次将指令码“52h”和“3/4 字节地址”从 SI 输入芯片，具体使用 3 字节地址还是 4 字节地址由芯片当前的地址模式决定，最后将 CS#信号置为高电平。需要注意的是，CS#信号必须在地址最后一个比特被芯片锁存后变为高电平，否则芯片不会执行擦除操作。

当芯片接受了擦除指令后，内部电路便开始相应操作。整个擦除操作会在  $t_{BE1}$  时间内完成。在此期间，用户可以通过检查 WIP 位的状态来确定擦除操作是否完成，如果 WIP 为 1，表示操作还未完成。当内部逻辑完成擦除操作后，会将 WIP 设置为 0，同时 WEL 也会被设置为 0。需要注意的是，如果被擦除的扇区处于写保护状态，那么擦除指令将不会被执行。

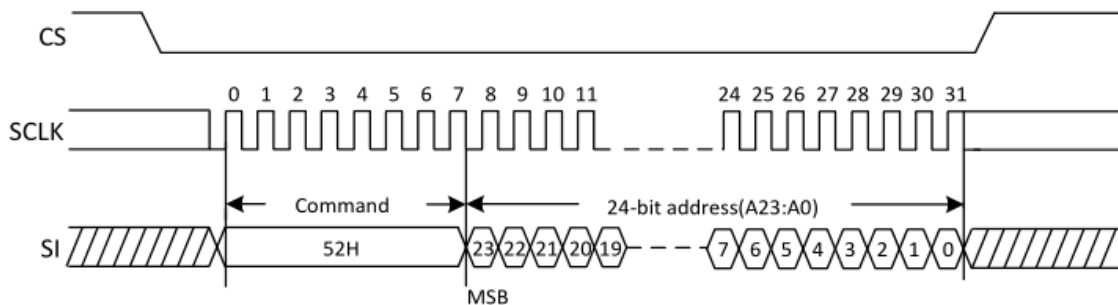


图 2-29 “52h” 指令时序图(3 字节地址模式)

### 2.2.26 三字节地址 64KB 扇区擦除(D8h)

用户可以使用“D8h”指令将某个 64KB 的扇区擦除。在进行擦除操作之前，用户必须使用“写使能”指令将状态寄存器的 WEL 设置为 1，否则擦除指令将会被芯片忽略。指令时序图如下所示，首先将 CS#信号置为低电平，然后在 SCLK 信号上升沿依次将指令码“D8h”和“3/4 字节地址”从 SI 输入芯片，具体使用 3 字节地址还是 4 字节地址由芯片当前的地址模式决定，

最后将 CS# 信号置为高电平。需要注意的是，CS# 信号必须在地址最后一个比特被芯片锁存后变为高电平，否则芯片不会执行擦除操作。

当芯片接受了擦除指令后，内部电路便开始相应操作。整个擦除操作会在  $t_{BE2}$  时间内完成。在此期间，用户可以通过检查 WIP 位的状态来确定擦除操作是否完成，如果 WIP 为 1，表示操作还未完成。当内部逻辑完成擦除操作后，会将 WIP 设置为 0，同时 WEL 也会被设置为 0。需要注意的是，如果被擦除的扇区被写保护，那么擦除指令将不会被执行。

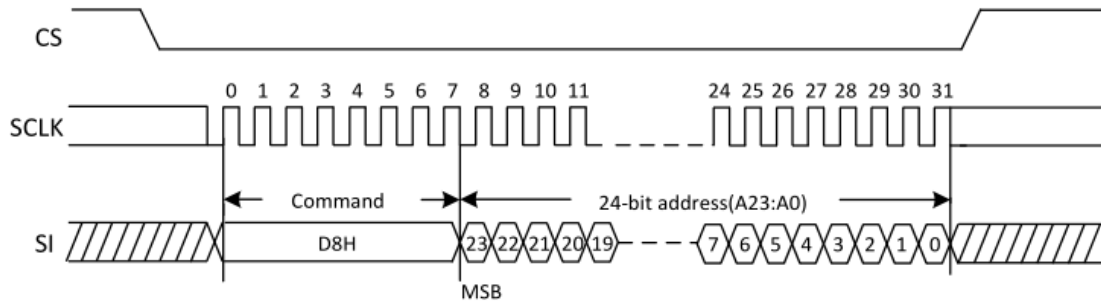


图 2-30 “D8h” 指令时序图 (3 字节地址模式)

### 2.2.27 全芯片擦除(C7h/60h)

用户可以使用“C7h/60h”指令将整个主存储区擦除。在进行擦除操作之前，用户必须使用“写使能”指令将状态寄存器的 WEL 设置为 1，否则“全芯片擦除”指令将会被芯片忽略。指令时序图如下所示，首先将 CS# 信号置为低电平，然后在 SCLK 信号上升沿将指令码“C7h/60h”从 SI 输入芯片，最后将 CS# 信号置为高电平。需要注意的是，CS# 信号必须在指令码最后一个比特被芯片锁存后变为高电平，否则芯片不会执行全芯片擦除操作。

当芯片接受了擦除指令后，内部电路便开始相应操作。整个擦除操作会在  $t_{CE}$  时间内完成。在此期间，用户可以通过检查 WIP 位的状态来确定擦除操作是否完成，如果 WIP 为 1，表示操作还未完成。当内部逻辑完成擦除操作后，会将 WIP 设置为 0，同时 WEL 也会被设置为 0。需要注意的是，如果有扇区处于写保护状态，那么“全芯片擦除”指令将不会被执行。

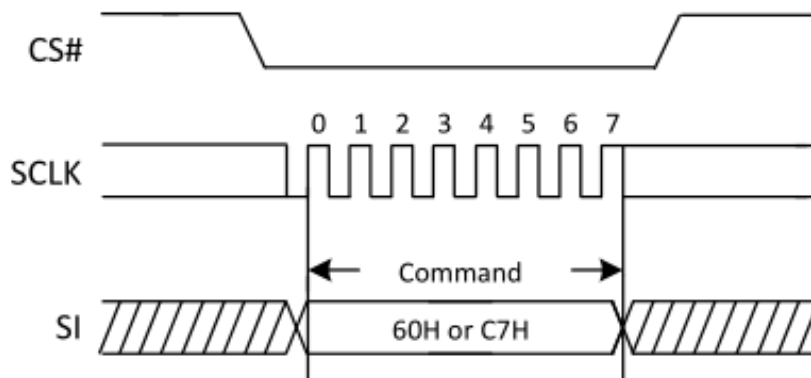


图 2-31 “C7h/60h” 指令时序图

### 3 电气特性

#### 3.1 绝对最大额定值

绝对最大额定值如下：

电源电压 ( $V_{CC}$ ) .....	-0.6V~4.6V
IO 端口电压 ( $V_{IO}$ ) .....	-0.6V~ $V_{CC}+0.4V$
存储温度( $T_{stg}$ ) .....	-65°C~150°C
结温( $T_J$ ).....	-65°C~150°C

#### 3.2 推荐工作条件

推荐工作条件如下：

电源电压 ( $V_{CC}$ ) .....	1.65V~3.6V
工作温度 ( $T_A$ ) .....	-55°C~105°C
结温( $T_J$ ).....	-55°C~105°C

#### 3.3 电特性表

##### 3.3.1 直流参数

表 3-1 直流参数表

参数	符号	测试条件 除另有规定外, $V_{CC}=1.65V\sim 3.6V$ , $T_A=-55^{\circ}C\sim 105^{\circ}C$ 。	最小值	最大值	单位
输入 IO 漏电流	$I_{LI}$	—	-4	4	$\mu A$
双向 IO 漏电流	$I_{LO}$	—	-4	4	$\mu A$
待机电流	$I_{CC1}$	/CS = $V_{CC}$	—	200	$\mu A$
低功耗模式电流	$I_{CC2}$	/CS = $V_{CC}$	—	50	$\mu A$
读数据电流	$I_{CC3}$	f = 35MHz	—	30	mA
		f = 25MHz	—	25	mA
页编程电流	$I_{CC4}$	/CS = $V_{CC}$	—	60	mA
写寄存器电流	$I_{CC5}$	—	—	30	mA
扇区/全芯片擦除电流	$I_{CC6}$	—	—	60	mA
输入低电平电压	$V_{IL}$	—	-0.5	$V_{CC}\times 0.2$	V
输入高电平电压	$V_{IH}$	—	$V_{CC}\times 0.7$	$V_{CC} + 0.4$	V
输出低电平电压	$V_{OL}$	$I_{OL} = 100\mu A$	—	0.4	V
输出高电平电压	$V_{OH}$	$I_{OH} = -100\mu A$	$V_{CC}-0.2$	—	V

### 3.3.2 交流参数

表 3-2 交流参数表 1

参数	符号	测试条件 除另有规定外, $V_{CC}=1.65V \sim 2.7V$ , $T_A=-55^{\circ}C \sim 105^{\circ}C$ 。	最小值	最大值	单位
负载电容	$C_L$	—	—	30	pF
时钟频率	$f_c, f_R$	所有指令	—	55	MHz
输出保持时间	$t_{CLQX}$	—	1	—	ns
时钟到输出有效时间	$t_{CLQV}$	负载 30pf	—	15	ns
/CS 高到输出高阻	$t_{SHQZ}$	—	—	12	ns
/CS 有效建立时间	$t_{SLCH}$	—	10	—	ns
/CS 无效建立时间	$t_{SHCH}$	—	10	—	ns
/CS 有效保持时间	$t_{CHSH}$	—	10	—	ns
/CS 无效保持时间	$t_{CHSL}$	—	10	—	ns
连续读指令间隔时间	$t_{SHSL1}$	—	40	—	ns
连续写指令间隔时间	$t_{SHSL2}$	—	40	—	ns
进入低功耗模式时间	$t_{DP}$	—	—	15	$\mu s$
退出低功耗模式时间	$t_{RES}$	—	—	55	$\mu s$
写寄存器时间	$t_W$	—	1	40	ms
页编程时间	$t_{PP}$	—	0.3	1.5	ms
4KB 扇区擦除时间	$t_{SE}$	—	65	3000	ms
32KB 扇区擦除时间	$t_{BE1}$	—	0.38	8	ms
64KB 扇区擦除时间	$t_{BE2}$	—	0.52	10	s
全芯片擦除时间	$t_{CE}$	—	150	300	s

表 3-3 交流参数表 2

参数	符号	测试条件 除另有规定外, $V_{CC}=2.7V \sim 3.6V$ , $T_A=-55^{\circ}C \sim 105^{\circ}C$ 。	最小值	最大值	单位
负载电容	$C_L$	—	—	30	pF
时钟频率	$f_c, f_R$	不包括 0BH/3BH/6BH 指令	—	70	MHz
快速读时钟频率	$f_{c1}$	0BH/3BH/6BH 指令	—	90	MHz
输出保持时间	$t_{CLQX}$	—	1	—	ns
时钟到输出有效时间	$t_{CLQV}$	负载 30pf	—	11	ns
/CS 高到输出高阻	$t_{SHQZ}$	—	—	12	ns
/CS 有效建立时间	$t_{SLCH}$	—	10	—	ns
/CS 无效建立时间	$t_{SHCH}$	—	10	—	ns
/CS 有效保持时间	$t_{CHSH}$	—	10	—	ns

参数	符号	测试条件		最小值	最大值	单位
		除另有规定外, $V_{CC}=2.7V\sim 3.6V$ , $T_A=-55^{\circ}C\sim 105^{\circ}C$ 。				
/CS 无效保持时间	$t_{CHSL}$	—	—	10	—	ns
连续读指令间隔时间	$t_{SHSL1}$	—	—	40	—	ns
连续写指令间隔时间	$t_{SHSL2}$	—	—	40	—	ns
进入低功耗模式时间	$t_{DP}$	—	—	—	15	$\mu s$
退出低功耗模式时间	$t_{RES}$	—	—	—	30	$\mu s$
写寄存器时间	$t_W$	—	—	1	40	ms
页编程时间	$t_{PP}$	—	—	0.3	1.5	ms
4KB 扇区擦除时间	$t_{SE}$	—	—	65	1500	ms
32KB 扇区擦除时间	$t_{BE1}$	—	—	0.38	4	ms
64KB 扇区擦除时间	$t_{BE2}$	—	—	0.52	5	s
全芯片擦除时间	$t_{CE}$	—	—	150	300	s

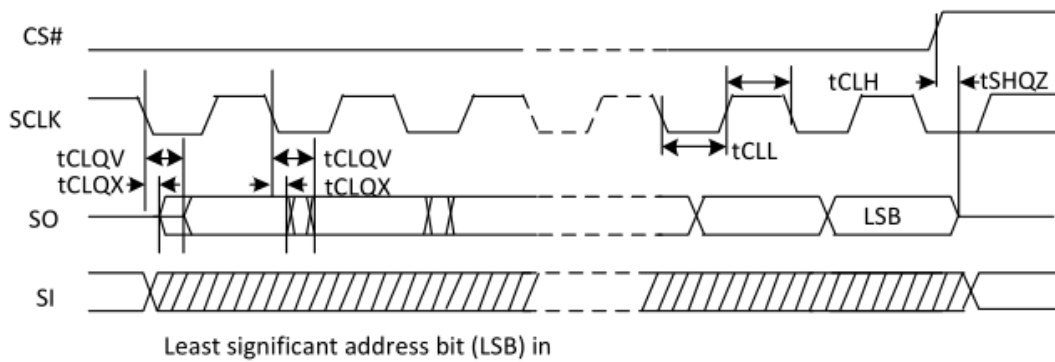


图 3-1 串行输出时序图

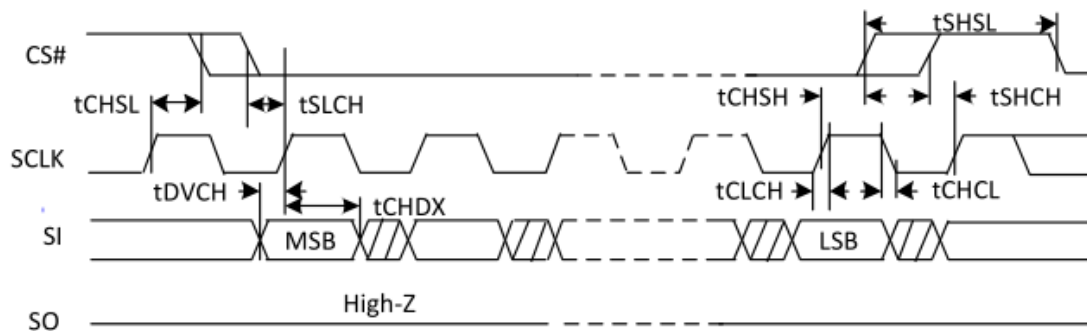


图 3-2 串行输入时序图



## 4 说明事项

### 4.1 运输与储存

芯片在适宜环境下储运。

使用指定的防静电包装盒进行产品的包装和运输。在运输过程中，确保芯片不要与外物发生碰撞。

### 4.2 开箱与检查

开箱使用芯片时，请注意观察产品标识。确定产品标识清晰，无污迹，无擦痕。同时，注意检查无损坏，无伤痕，引脚整齐，无缺失，无变形。

### 4.3 使用操作规程及注意事项

器件必须采取防静电措施进行操作。取用芯片时应佩戴防静电手套，防止人体电荷对芯片的静电冲击，损坏芯片。将芯片插入电路板上的底座时以及将芯片从电路板上的底座取出时，应注意施力方向以确保芯片引脚均匀受力。不要因为用力过猛，损坏芯片引脚，导致无法使用。

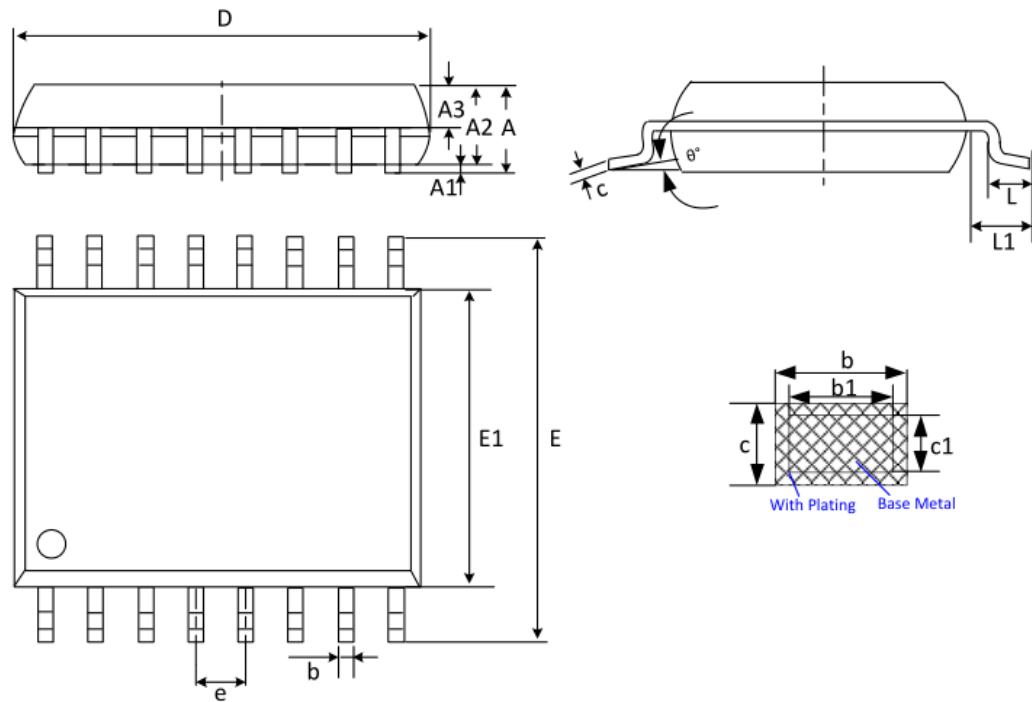
推荐下列操作措施：

- a) 器件应在防静电的工作台上操作，或带指套操作；
- b) 试验设备和器具应接地；
- c) 此不能触摸器件引线；
- d) 器件应存放在导电材料制成的容器中（如：集成电路专用盒）；
- e) 生产、测试、使用以及转运过程中应避免使用引起静电的塑料、橡胶或丝织物；

### 4.4 质量保证

公司质量管理体系根据国军标 GJB9001 要求制定了完善的质量管理工作流程，对产品的设计、生产和销售进行日常质量管理。产品制定依据 GJB7400《合格制造厂认证用半导体集成电路通用规范》裁剪后的标准进行设计和生产，并按照 GJB548B-2005《微电子器件试验方法和程序》的要求进行试验和检验。产品兼容性好、可靠高。

## 5 封装尺寸



单位为毫米

尺寸符号	最小值	公称值	最大值
A	--	--	2.65
A1	0.10	--	0.30
A2	2.25	2.30	2.35
A3	0.97	1.02	1.07
b	0.35	--	0.43
b1	0.34	0.37	0.40
c	0.25	--	0.29
c1	0.24	0.25	0.26
D	10.20	10.30	10.40
E	10.10	10.30	10.50
E1	7.40	7.50	7.60
e	--	1.27	--
L	0.55	--	0.85
L1	--	1.40	--
θ	0°	----	8°

图 5-1 SOP16 封装尺寸图

## 6 订货信息

### 6.1 选型列表

表 6-1 选型列表

型号	封装	引脚数
AST25QW512S	SOP (300mil)	16