# GD25LR128D

# DATASHEET

# Contents

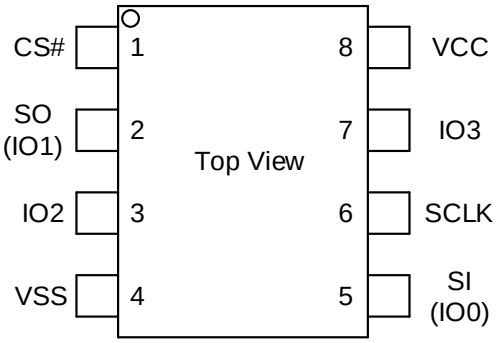
# 1. FEATURES

- 128M-bit Serial Flash
  - 16M-Byte
  - 256 Bytes per programmable page
- Standard, Dual, Quad SPI
  - Standard SPI: SCLK, CS#, SI, SO,
  - Dual SPI: SCLK, CS#, IO0, IO1,
  - Quad SPI: SCLK, CS#, IO0, IO1, IO2, IO3
  - QPI: SCLK, CS#, IO0, IO1, IO2, IO3
- High Speed Clock Frequency
  - 120MHz for fast read with 30PF load
  - Dual I/O Data transfer up to 240Mbits/s
  - Quad I/O Data transfer up to 480Mbits/s
  - QPI Mode Data transfer up to 480Mbits/s
- Software Write Protection
  - Write protect all/portion of memory via software
  - Top/Bottom Block protection
- Endurance and Data Retention
  - Minimum 100,000 Program/Erase Cycles
  - 20-year data retention typical
- RPMC Function
  - Four 32-bit Monotonic Counters
  - Volatile HMAC Key Register
  - Non-volatile Root Key Register
- Fast Program/Erase Speed
  - Page Program time: 0.5ms typical
  - Sector Erase time: 70ms typical
  - Block Erase time: 0.16s/0.3s typical
  - Chip Erase time: 50s typical
- Flexible Architecture
  - Uniform Sector of 4K-Byte
  - Uniform Block of 32/64K-Byte
- Low Power Consumption
  - 45μA typical standby current
  - 2μA typical deep power down current
- Advanced Security Features
  - 128-bit Unique ID for each device
  - Serial Flash Discoverable parameters (SFDP) register
  - 3x1024-Byte Security Registers With OTP Locks
- Allows XIP (eXecute In Place) Operation
  - High speed Read commands reduce overall XiP instruction fetch time
- Single Power Supply Voltage
  - Full voltage range: 1.65-2.0V
- Package Information
  - SOP8 208mil
  - WSON8 (6x5mm)
  - WSON8 (8x6mm)

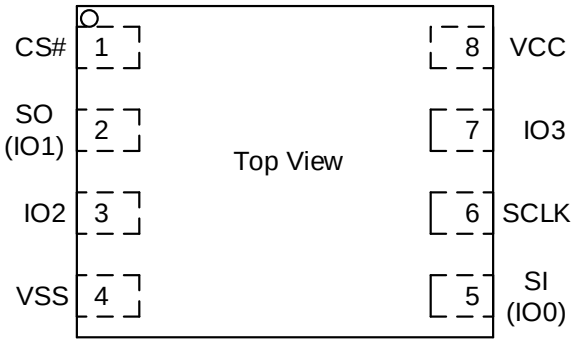
# 2. GENERAL DESCRIPTION

The GD25LR128D (128M-bit) Serial flash supports the standard Serial Peripheral Interface (SPI), and supports the Dual/Quad SPI and QPI mode: Serial Clock, Chip Select, Serial Data I/O0 (SI), I/O1 (SO), I/O2, and I/O3. The Dual I/O data is transferred with speed of 240Mbits/s and the Quad I/O & Quad output data is transferred with speed of 480Mbits/s.

## CONNECTION DIAGRAM




8 - LEAD SOP




8 - LEAD WSON

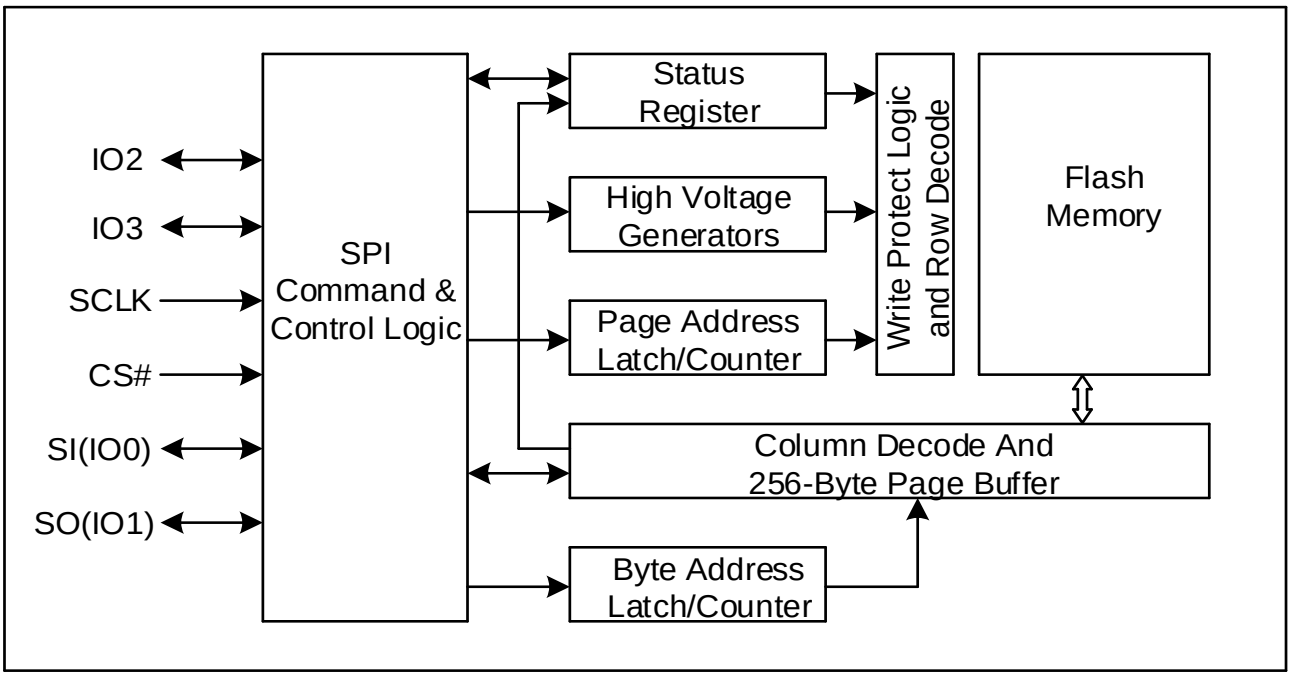
## PIN DESCRIPTION

| Pin No. | Pin Name | I/O | Description |
|---|---|---|---|
| 1 | CS# | I | Chip Select Input |
| 2 | SO (IO1) | I/O | Data Output (Data Input Output 1) |
| 3 | IO2 | I/O | Data Input Output 2 |
| 4 | VSS | | Ground |
| 5 | SI (IO0) | I/O | Data Input (Data Input Output 0) |
| 6 | SCLK | I | Serial Clock Input |
| 7 | IO3 | I/O | Data Input Output 3 |
| 8 | VCC | | Power Supply |

Note:

1. CS# must be driven high if chip is not selected. Please don't leave CS# floating any time after power is on.

## BLOCK DIAGRAM



# 3. MEMORY ORGANIZATION

**GD25LR128D**

| Each device has | Each block has | Each sector has | Each page has | |
|---|---|---|---|---|
| 16M | 64/32K | 4K | 256 | bytes |
| 64K | 256/128 | 16 | - | pages |
| 4096 | 16/8 | - | - | sectors |
| 256/512 | - | - | - | blocks |

**UNIFORM BLOCK SECTOR ARCHITECTURE**

**GD25LR128D 64K Bytes Block Sector Architecture**

| Block | Sector | Address range | |
|---|---|---|---|
| 255 | 4095 | FFF000H | FFFFFFH |
| | ..... | ..... | ..... |
| | 4080 | FF0000H | FF0FFFH |
| 254 | 4079 | FEF000H | FEFFFFH |
| | ..... | ..... | ..... |
| | 4064 | FE0000H | FE0FFFH |
| ..... | ..... | ..... | ..... |
| | ..... | ..... | ..... |
| | ..... | ..... | ..... |
| ..... | ..... | ..... | ..... |
| | ..... | ..... | ..... |
| | ..... | ..... | ..... |
| 2 | 47 | 02F000H | 02FFFFH |
| | ..... | ..... | ..... |
| | 32 | 020000H | 020FFFH |
| 1 | 31 | 01F000H | 01FFFFH |
| | ..... | ..... | ..... |
| | 16 | 010000H | 010FFFH |
| 0 | 15 | 00F000H | 00FFFFH |
| | ..... | ..... | ..... |
| | 0 | 000000H | 000FFFH |

# 4. DEVICE OPERATION

## SPI Mode

### Standard SPI

The GD25LR128D features a serial peripheral interface on 4 signals bus: Serial Clock (SCLK), Chip Select (CS#), Serial Data Input (SI) and Serial Data Output (SO). Both SPI bus mode 0 and 3 are supported. Input data is latched on the rising edge of SCLK and data shifts out on the falling edge of SCLK.

### Dual SPI

The GD25LR128D supports Dual SPI operation when using the “Dual Output Fast Read” and “Dual I/O Fast Read” (3BH and BBH) commands. These commands allow data to be transferred to or from the device at twice times the rate of the standard SPI. When using the Dual SPI command the SI and SO pins become bidirectional I/O pins: IO0 and IO1.

### Quad SPI

The GD25LR128D supports Quad SPI operation when using the “Quad Output Fast Read”,” Quad I/O Fast Read”, “Quad Page Program” (6BH, EBH, 32H) commands. These commands allow data to be transferred to or from the device at four times the rate of the standard SPI. When using the Quad SPI command the SI and SO pins become bidirectional I/O pins: IO0 and IO1. For GD25LR128D, the QE bit is set to 1 as default and cannot be changed.

### QPI

The GD25LR128D supports Quad Peripheral Interface (QPI) operations only when the device is switched from Standard/Dual/Quad SPI mode to QPI mode using the “Enable the QPI (38H)” command. The QPI mode utilizes all four IO pins to input the command code. Standard/Dual/Quad SPI mode and QPI mode are exclusive. Only one mode can be active at any given times. “Enable the QPI (38H)” and “Disable the QPI (FFH)” commands are used to switch between these two modes. Upon power-up and after software reset using “”Reset (99H)” command, the default state of the device is Standard/Dual/Quad SPI mode. For GD25LR128D, the QE bit is set to 1 as default and cannot be changed.

# 5. DATA PROTECTION

The GD25LR128D provide the following data protection methods:

- Write Enable (WREN) command: The WREN command is set the Write Enable Latch bit (WEL). The WEL bit will return to reset by the following situation:
  -Power-Up
  -Write Disable (WRDI)
  -Write Status Register (WRSR)
  -Page Program (PP)
  -Sector Erase (SE) / Block Erase (BE) / Chip Erase (CE)
  -Software reset (66H+99H)
  -Erase Security Registers / Program Security Registers
- Software Protection Mode: The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits define the section of the memory array that can be read but not change.
- Deep Power-Down Mode: In Deep Power-Down Mode, all commands are ignored except the Release from Deep Power-Down Mode command and reset command (66H+99H).

**Table1. GD25LR128D Protected area size (CMP=0)**

| Status Register Content | | | | | Memory Content | | | |
|---|---|---|---|---|---|---|---|---|
| BP4 | BP3 | BP2 | BP1 | BP0 | Blocks | Addresses | Density | Portion |
| X | X | 0 | 0 | 0 | NONE | NONE | NONE | NONE |
| 0 | 0 | 0 | 0 | 1 | 252 to 255 | FC0000H-FFFFFFH | 256KB | Upper 1/64 |
| 0 | 0 | 0 | 1 | 0 | 248 to 255 | F80000H-FFFFFFH | 512KB | Upper 1/32 |
| 0 | 0 | 0 | 1 | 1 | 240 to 255 | F00000H-FFFFFFH | 1MB | Upper 1/16 |
| 0 | 0 | 1 | 0 | 0 | 224 to 255 | E00000H-FFFFFFH | 2MB | Upper 1/8 |
| 0 | 0 | 1 | 0 | 1 | 192 to 255 | C00000H-FFFFFFH | 4MB | Upper 1/4 |
| 0 | 0 | 1 | 1 | 0 | 128 to 255 | 800000H-FFFFFFH | 8MB | Upper 1/2 |
| 0 | 1 | 0 | 0 | 1 | 0 to 3 | 000000H-03FFFFH | 256KB | Lower 1/64 |
| 0 | 1 | 0 | 1 | 0 | 0 to 7 | 000000H-07FFFFH | 512KB | Lower 1/32 |
| 0 | 1 | 0 | 1 | 1 | 0 to 15 | 000000H-0FFFFFH | 1MB | Lower 1/16 |
| 0 | 1 | 1 | 0 | 0 | 0 to 31 | 000000H-1FFFFFH | 2MB | Lower 1/8 |
| 0 | 1 | 1 | 0 | 1 | 0 to 63 | 000000H-3FFFFFH | 4MB | Lower 1/4 |
| 0 | 1 | 1 | 1 | 0 | 0 to 127 | 000000H-7FFFFFH | 8MB | Lower 1/2 |
| X | X | 1 | 1 | 1 | 0 to 255 | 000000H-FFFFFFH | 16MB | ALL |
| 1 | 0 | 0 | 0 | 1 | 255 | FFF000H-FFFFFFH | 4KB | Top Block |
| 1 | 0 | 0 | 1 | 0 | 255 | FFE000H-FFFFFFH | 8KB | Top Block |
| 1 | 0 | 0 | 1 | 1 | 255 | FFC000H-FFFFFFH | 16KB | Top Block |
| 1 | 0 | 1 | 0 | X | 255 | FF8000H-FFFFFFH | 32KB | Top Block |
| 1 | 0 | 1 | 1 | 0 | 255 | FF8000H-FFFFFFH | 32KB | Top Block |
| 1 | 1 | 0 | 0 | 1 | 0 | 000000H-000FFFH | 4KB | Bottom Block |
| 1 | 1 | 0 | 1 | 0 | 0 | 000000H-001FFFH | 8KB | Bottom Block |
| 1 | 1 | 0 | 1 | 1 | 0 | 000000H-003FFFH | 16KB | Bottom Block |
| 1 | 1 | 1 | 0 | X | 0 | 000000H-007FFFH | 32KB | Bottom Block |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 0 | 0 | 000000H-007FFFH | 32KB | Bottom Block |

**Table1a. GD25LR128D Protected area size (CMP=1)**

| Status Register Content | | | | | Memory Content | | | |
|---|---|---|---|---|---|---|---|---|
| **BP4** | **BP3** | **BP2** | **BP1** | **BP0** | **Blocks** | **Addresses** | **Density** | **Portion** |
| X | X | 0 | 0 | 0 | 0 to 255 | 000000H-FFFFFFH | ALL | ALL |
| 0 | 0 | 0 | 0 | 1 | 0 to 251 | 000000H-FBFFFFH | 16128KB | Lower 63/64 |
| 0 | 0 | 0 | 1 | 0 | 0 to 247 | 000000H-F7FFFFH | 15872KB | Lower 31/32 |
| 0 | 0 | 0 | 1 | 1 | 0 to 239 | 000000H-EFFFFFH | 15MB | Lower 15/16 |
| 0 | 0 | 1 | 0 | 0 | 0 to 223 | 000000H-DFFFFFH | 14MB | Lower 7/8 |
| 0 | 0 | 1 | 0 | 1 | 0 to 191 | 000000H-BFFFFFH | 12MB | Lower 3/4 |
| 0 | 0 | 1 | 1 | 0 | 0 to 127 | 000000H-7FFFFFH | 8MB | Lower 1/2 |
| 0 | 1 | 0 | 0 | 1 | 4 to 255 | 040000H-FFFFFFH | 16128KB | Upper 63/64 |
| 0 | 1 | 0 | 1 | 0 | 8 to 255 | 080000H-FFFFFFH | 15872KB | Upper 31/32 |
| 0 | 1 | 0 | 1 | 1 | 16 to 255 | 100000H-FFFFFFH | 15MB | Upper 15/16 |
| 0 | 1 | 1 | 0 | 0 | 32 to 255 | 200000H-FFFFFFH | 14MB | Upper 7/8 |
| 0 | 1 | 1 | 0 | 1 | 64 to 255 | 400000H-FFFFFFH | 12MB | Upper 3/4 |
| 0 | 1 | 1 | 1 | 0 | 128 to 255 | 800000H-FFFFFFH | 8MB | Upper 1/2 |
| X | X | 1 | 1 | 1 | NONE | NONE | NONE | NONE |
| 1 | 0 | 0 | 0 | 1 | 0 to 255 | 000000H-FFEFFFH | 16380KB | L-4095/4096 |
| 1 | 0 | 0 | 1 | 0 | 0 to 255 | 000000H-FFDFFFH | 16376KB | L-2047/2048 |
| 1 | 0 | 0 | 1 | 1 | 0 to 255 | 000000H-FFBFFFH | 16368KB | L-1023/1024 |
| 1 | 0 | 1 | 0 | X | 0 to 255 | 000000H-FF7FFFH | 16352KB | L-511/512 |
| 1 | 0 | 1 | 1 | 0 | 0 to 255 | 000000H-FF7FFFH | 16352KB | L-511/512 |
| 1 | 1 | 0 | 0 | 1 | 0 to 255 | 001000H-FFFFFFH | 16380KB | U-4095/4096 |
| 1 | 1 | 0 | 1 | 0 | 0 to 255 | 002000H-FFFFFFH | 16376KB | U-2047/2048 |
| 1 | 1 | 0 | 1 | 1 | 0 to 255 | 004000H-FFFFFFH | 16368KB | U-1023/1024 |
| 1 | 1 | 1 | 0 | X | 0 to 255 | 008000H-FFFFFFH | 16352KB | U-511/512 |
| 1 | 1 | 1 | 1 | 0 | 0 to 255 | 008000H-FFFFFFH | 16352KB | U-511/512 |

# 6. STATUS REGISTER

| S15 | S14 | S13 | S12 | S11 | S10 | S9 | S8 |
|---|---|---|---|---|---|---|---|
| SUS1 | CMP | LB3 | LB2 | LB1 | SUS2 | QE | SRP1 |

| S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 |
|---|---|---|---|---|---|---|---|
| SRP0 | BP4 | BP3 | BP2 | BP1 | BP0 | WEL | WIP |

The status and control bits of the Status Register are as follows:

**WIP bit.**

The Write in Progress (WIP) bit indicates whether the memory is busy in program/erase/write status register progress. When WIP bit sets to 1, means the device is busy in program/erase/write status register progress, when WIP bit sets 0, means the device is not in program/erase/write status register progress.

**WEL bit.**

The Write Enable Latch (WEL) bit indicates the status of the internal Write Enable Latch. When set to 1 the internal Write Enable Latch is set, when set to 0 the internal Write Enable Latch is reset and no Write Status Register, Program or Erase command is accepted.

**BP4, BP3, BP2, BP1, BP0 bits.**

The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase commands. These bits are written with the Write Status Register (WRSR) command. When the Block Protect (BP4, BP3, BP2, BP1, BP0) bits are set to 1, the relevant memory area (as defined in Table1).becomes protected against Page Program (PP), Sector Erase (SE) and Block Erase (BE) commands. The Chip Erase (CE) command is executed, if the Block Protect (BP2, BP1, and BP0) bits are 0 and CMP=0 or the Block Protect (BP2, BP1, and BP0) bits are 1 and CMP=1.

**SRP1, SRP0 bits.**

The Status Register Protect (SRP1 and SRP0) bits are non-volatile Read/Write bits in the status register. The SRP bits control the method of write protection: software protection, power supply lock-down or one time programmable protection.

| SRP1 | SRP0 | Status Register | Description |
|---|---|---|---|
| 0 | X | Software Protected | The Status Register can be written to after a Write Enable command, WEL=1.(Default) |
| 1 | 0 | Power Supply Lock-Down(1)(2) | Status Register is protected and cannot be written to again until the next Power-Down, Power-Up cycle. |
| 1 | 1 | One Time Program(2) | Status Register is permanently protected and cannot be written to. |

NOTE:

1. When SRP1, SRP0= (1, 0), a Power-Down, Power-Up cycle will change SRP1, SRP0 to (0, 0) state.
2. This feature is available on special order. Please contact GigaDevice for details.

**QE bit.**

The Quad Enable (QE) bit is a non-volatile read only bit which allows Quad operation. The default value of QE bit is 1, and it cannot be changed so that the Quad IO2 and IO3 pins are enabled all the time.

**LB3, LB2, LB1 bits.**

The LB3, LB2, LB1 bits are non-volatile One Time Program (OTP) bits in Status Register (S13-S11) that provide the write protect control and status to the Security Registers. The default state of LB3-LB1 are 0, the security registers are unlocked. The LB3-LB1 bits can be set to 1 individually using the Write Register instruction. The LB3-LB1 bits are One Time

Programmable, once they are set to 1, the Security Registers will become read-only permanently.

**CMP bit**

The CMP bit is a non-volatile Read/Write bit in the Status Register (S14). It is used in conjunction with the BP4-BP0 bits to provide more flexibility for the array protection. Please see the Status registers Memory Protection table for details. The default setting is CMP=0.

**SUS1, SUS2 bits**

The SUS1 and SUS2 bits are read only bit in the status register (S15 and S10) that are set to 1 after executing an Program/Erase Suspend (75H) command (The Erase Suspend will set the SUS1 to 1,and the Program Suspend will set the SUS2 to 1). The SUS1 and SUS2 bits are cleared to 0 by Program/Erase Resume (7AH) command as well as a power-down, power-up cycle.

# 7. COMMANDS DESCRIPTION

All commands, addresses and data are shifted in and out of the device, beginning with the most significant bit on the first rising edge of SCLK after CS# is driven low. Then, the one-byte command code must be shifted in to the device, with most significant bit first on SI, and each bit is latched on the rising edges of SCLK.

See Table2, every command sequence starts with a one-byte command code. Depending on the command, this might be followed by address bytes, or by data bytes, or by both or none. CS# must be driven high after the last bit of the command sequence has been completed shifted in. For the command of Read, Fast Read, Read Status Register or Release from Deep Power-Down, and Read Device ID, the shifted-in command sequence is followed by a data-out sequence. All read instruction can be completed after any bit of the data-out sequence is being shifted out, and then CS# must be driven high to return to deselected status.

For the command of Page Program, Sector Erase, Block Erase, Chip Erase, Write Status Register, Write Enable, Write Disable or Deep Power-Down command, CS# must be driven high exactly at a byte boundary, otherwise the command is rejected, and is not executed. That is CS# must be driven high when the number of clock pulses after CS# being driven low is an exact multiple of eight. For Page Program, if at any time the input byte is not a full byte, nothing will happen and WEL will not be reset.

**Table2. Commands (Standard/Dual/Quad SPI)**

| Command Name | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | n-Bytes |
|---|---|---|---|---|---|---|---|
| Write Enable | 06H | | | | | | |
| Write Disable | 04H | | | | | | |
| Volatile SR Write Enable | 50H | | | | | | |
| Read Status Register-1 | 05H | (S7-S0) | | | | | (continuous) |
| Read Status Register-2 | 35H | (S15-S8) | | | | | (continuous) |
| Write Status Register | 01H | S7-S0 | S15-S8 | | | | |
| Read Data | 03H | A23-A16 | A15-A8 | A7-A0 | (D7-D0) | (Next byte) | (continuous) |
| Fast Read | 0BH | A23-A16 | A15-A8 | A7-A0 | dummy | (D7-D0) | (continuous) |
| Dual Output Fast Read | 3BH | A23-A16 | A15-A8 | A7-A0 | dummy | (D7-D0)(1) | (continuous) |
| Dual I/O Fast Read | BBH | A23-A8(2) | A7-A0 M7-M0(2) | (D7-D0)(1) | | | (continuous) |
| Quad Output Fast Read | 6BH | A23-A16 | A15-A8 | A7-A0 | dummy | (D7-D0)(3) | (continuous) |
| Quad I/O Fast Read | EBH | A23-A0 M7-M0(4) | dummy(5) | (D7-D0)(3) | | | (continuous) |
| Page Program | 02H | A23-A16 | A15-A8 | A7-A0 | D7-D0 | Next byte | |
| Quad Page Program | 32H | A23-A16 | A15-A8 | A7-A0 | D7-D0 | | |
| Sector Erase | 20H | A23-A16 | A15-A8 | A7-A0 | | | |
| Block Erase(32K) | 52H | A23-A16 | A15-A8 | A7-A0 | | | |
| Block Erase(64K) | D8H | A23-A16 | A15-A8 | A7-A0 | | | |
| Chip Erase | C7/60H | | | | | | |
| Enable QPI | 38H | | | | | | |
| Enable Reset | 66H | | | | | | |
| Reset | 99H | | | | | | |
| Set Burst with Wrap | 77H | W6-W4 | | | | | |
| Program/Erase Suspend | 75H | | | | | | |
| Program/Erase Resume | 7AH | | | | | | |
| Release From Deep | ABH | dummy | dummy | dummy | (ID7-ID0) | | (continuous) |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Power-Down, And Read Device ID | | | | | | | |
| Release From Deep Power-Down | ABH | | | | | | |
| Deep Power-Down | B9H | | | | | | |
| Manufacturer/ Device ID | 90H | dummy | dummy | 00H | (M7-M0) | (ID7-ID0) | (continuous) |
| Read Identification | 9FH | (M7-M0) | (ID15-ID8) | (ID7-ID0) | | | (continuous) |
| Read Serial Flash Discoverable Parameter(10) | 5AH | A23-A16 | A15-A8 | A7-A0 | dummy | (D7-D0) | (continuous) |
| Read Unique ID | 4BH | 00H | 00H | 00H | dummy | (UID7-UID0) | (continuous) |
| Erase Security Registers(6) | 44H | A23-A16 | A15-A8 | A7-A0 | | | |
| Program Security Registers(6) | 42H | A23-A16 | A15-A8 | A7-A0 | D7-D0 | D7-D0 | |
| Read Security Registers(6) | 48H | A23-A16 | A15-A8 | A7-A0 | dummy | (D7-D0) | |

**Table2a. Commands (QPI)**

| Command Name | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 |
|---|---|---|---|---|---|---|---|
| Clock Number | (0,1) | (2,3) | (4,5) | (6,7) | (8,9) | (10,11) | (12,13) |
| Write Enable | 06H | | | | | | |
| Volatile SR Write Enable | 50H | | | | | | |
| Write Disable | 04H | | | | | | |
| Read Status Register | 05H | (S7-S0) | | | | | |
| Read Status Register-1 | 35H | (S15-S8) | | | | | |
| Read Status Register-2 | 15H | (S1-S0) | | | | | |
| Write Status Register | 01H | S7-S0 | S15-S8 | | | | |
| Page Program | 02H | A23-A16 | A15-A8 | A7-A0 | D7-D0 | Next byte | |
| Sector Erase | 20H | A23-A16 | A15-A8 | A7-A0 | | | |
| Block Erase(32K) | 52H | A23-A16 | A15-A8 | A7-A0 | | | |
| Block Erase(64K) | D8H | A23-A16 | A15-A8 | A7-A0 | | | |
| Chip Erase | C7/60H | | | | | | |
| Program/Erase Suspend | 75H | | | | | | |
| Program/Erase Resume | 7AH | | | | | | |
| Deep Power-Down | B9H | | | | | | |
| Set Read Parameters | C0H | P7-P0 | | | | | |
| Fast Read | 0BH | A23-A16 | A15-A8 | A7-A0 | dummy | dummy | (D7-D0) |
| Burst Read with Wrap | 0CH | A23-A16 | A15-A8 | A7-A0 | dummy | dummy | (D7-D0) |
| Quad I/O Fast Read | EBH | A23-A16 | A15-A8 | A7-A0 | M7-M0 | dummy | (D7-D0) |
| Release From Deep Power-Down, And Read Device ID | ABH | dummy | dummy | dummy | (ID7-ID0) | (cont.) | |
| Manufacturer/ Device ID | 90H | dummy | dummy | 00H | (M7-M0) | (ID7-ID0) | |
| Read Identification | 9FH | (M7-M0) | (ID15-ID8) | (ID7-ID0) | | | |
| Read Serial Flash Discoverable Parameter | 5AH | A23-A16 | A15-A8 | A7-A0 | dummy | (D7-D0) | |
| Disable QPI | FFH | | | | | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Enable Reset | 66H | | | | | | |
| Reset | 99H | | | | | | |

NOTE:

1. Dual Output data

   IO0 = (D6, D4, D2, D0)

   IO1 = (D7, D5, D3, D1)

2. Dual Input Address

   IO0 = A22, A20, A18, A16, A14, A12, A10, A8  A6, A4, A2, A0, M6, M4, M2, M0

   IO1 = A23, A21, A19, A17, A15, A13, A11, A9  A7, A5, A3, A1, M7, M5, M3, M1

3. Quad Output Data

   IO0 = (D4, D0, …..)

   IO1 = (D5, D1, …..)

   IO2 = (D6, D2, …..)

   IO3 = (D7, D3,…..)

4. Quad Input Address

   IO0 = A20, A16, A12, A8, A4, A0, M4, M0

   IO1 = A21, A17, A13, A9, A5, A1, M5, M1

   IO2 = A22, A18, A14, A10, A6, A2, M6, M2

   IO3 = A23, A19, A15, A11, A7, A3, M7, M3

5. Fast Read Quad I/O Data

   IO0 = (x, x, x, x, D4, D0,…)

   IO1 = (x, x, x, x, D5, D1,…)

   IO2 = (x, x, x, x, D6, D2,…)

   IO3 = (x, x, x, x, D7, D3,…)

6. Security Registers Address:

   Security Register1: A23-A16=00H, A15-A9=000100b, A9-A0=Byte Address;

   Security Register2: A23-A16=00H, A15-A9=001000b, A9-A0=Byte Address;

   Security Register3: A23-A16=00H, A15-A9=001100b, A9-A0=Byte Address.

7. QPI Command, Address, Data input/output format:

   CLK #0 1 2 3 4 5 6 7 8 9 10 11

   IO0= C4, C0, A20, A16, A12, A8, A4, A0, D4, D0, D4, D0,

   IO1= C5, C1, A21, A17, A13, A9, A5, A1, D5, D1, D5, D1

   IO2= C6, C2, A22, A18, A14, A10, A6, A2, D6, D2, D6, D2

   IO3= C7, C3, A23, A19, A15, A11, A7, A3, D7, D3, D7, D3

**Table of ID Definitions:**

**GD25LR128D**

| Operation Code | M7-M0 | ID15-ID8 | ID7-ID0 |
|---|---|---|---|
| 9FH | C8 | 60 | 18 |
| 90H | C8 | | 17 |
| ABH | | | 17 |

## 7.1. Write Enable (WREN) (06H)

The Write Enable (WREN) command is for setting the Write Enable Latch (WEL) bit. The Write Enable Latch (WEL) bit must be set prior to every Page Program (PP), Sector Erase (SE), Block Erase (BE), Chip Erase (CE), Write Status Register (WRSR) and Erase/Program Security Registers command. The Write Enable (WREN) command sequence: CS# goes low → sending the Write Enable command → CS# goes high.

**Figure 1. Write Enable Sequence Diagram**

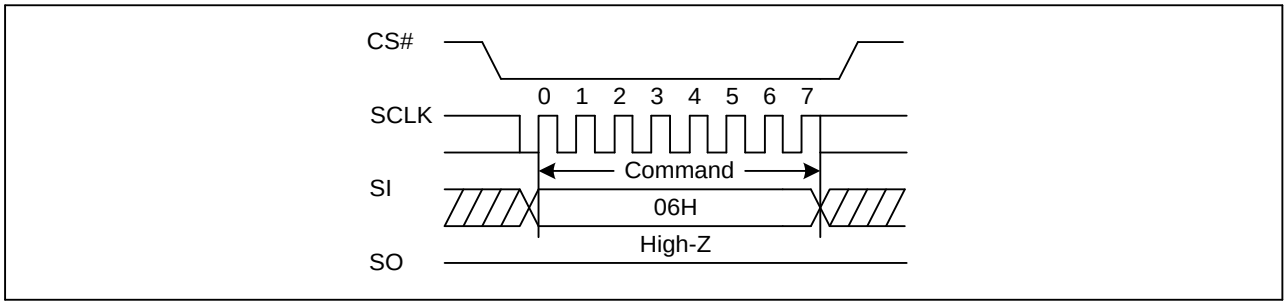
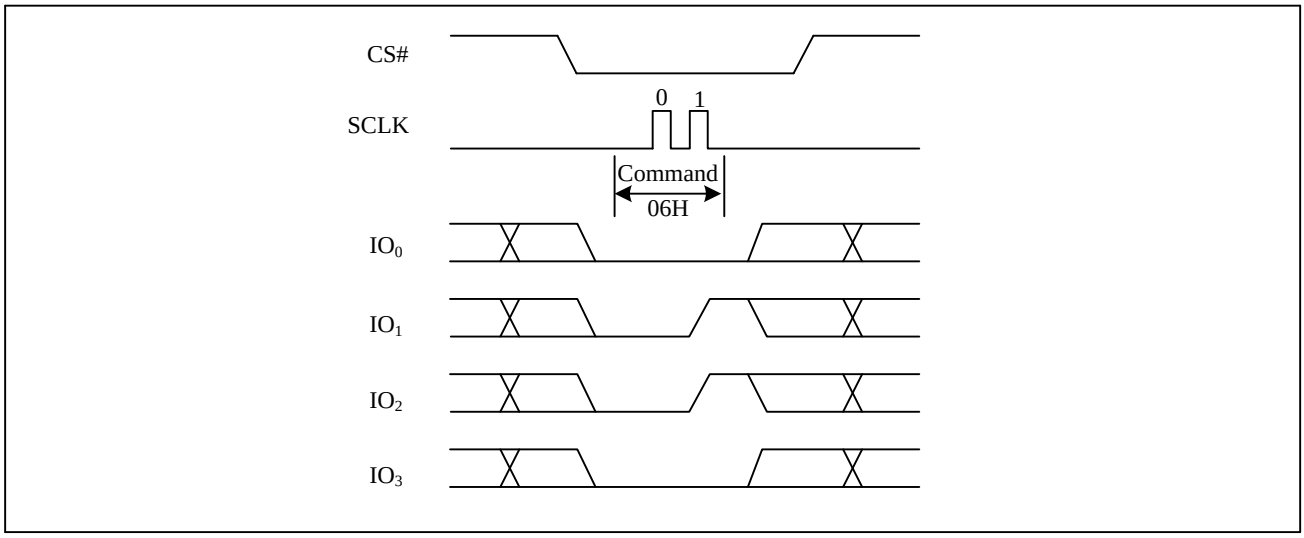



**Figure 2. Write Enable Sequence Diagram (QPI)**



## 7.2. Write Disable (WRDI) (04H)

The Write Disable command is for resetting the Write Enable Latch (WEL) bit. The Write Disable command sequence: CS# goes low →Sending the Write Disable command →CS# goes high. The WEL bit is reset by following condition: Power-up and upon completion of the Write Status Register, Page Program, Sector Erase, Block Erase, Chip Erase, Erase/Program Security Registers and Reset commands.

**Figure3. Write Disable Sequence Diagram**

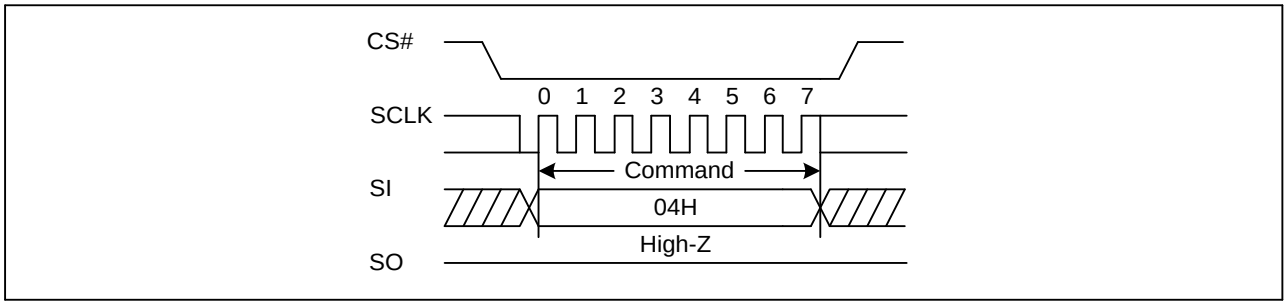
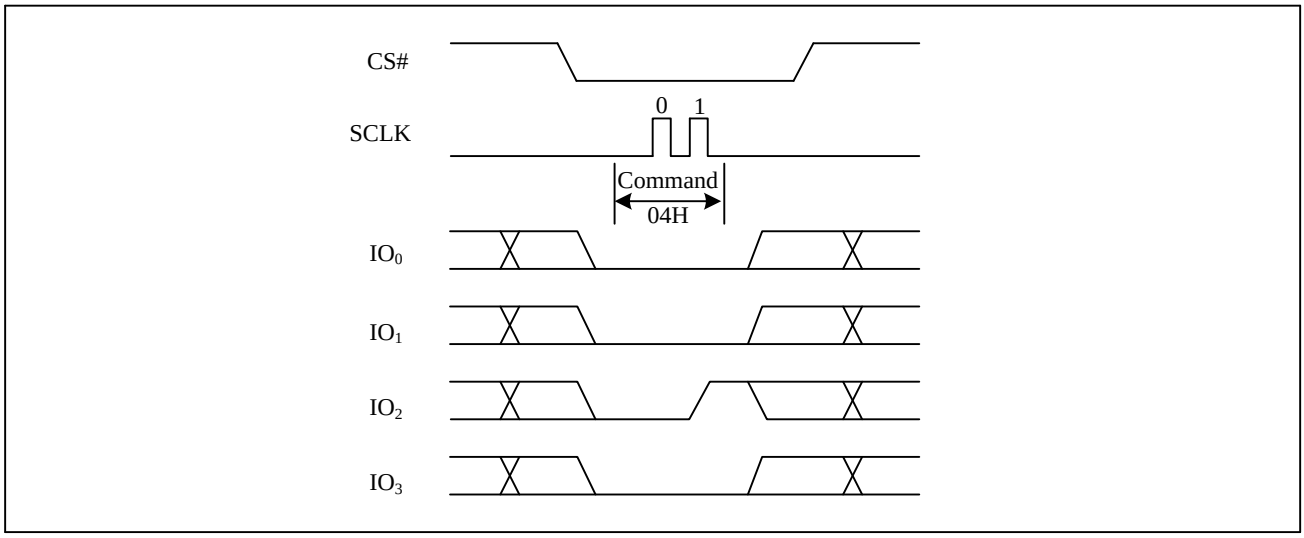



**Figure3a. Write Disable Sequence Diagram (QPI)**



## 7.3. Write Enable for Volatile Status Register (50H)

The non-volatile Status Register bits can also be written to as volatile bits. This gives more flexibility to change the system configuration and memory protection schemes quickly without waiting for the typical non-volatile bit write cycles or affecting the endurance of the Status Register non-volatile bits. The Write Enable for Volatile Status Register command must be issued prior to a Write Status Register command and any other commands can't be inserted between them. Otherwise, Write Enable for Volatile Status Register will be cleared. The Write Enable for Volatile Status Register command will not set the Write Enable Latch bit, it is only valid for the Write Status Register command to change the volatile Status Register bit values.

**Figure4. Write Enable for Volatile Status Register Sequence Diagram**

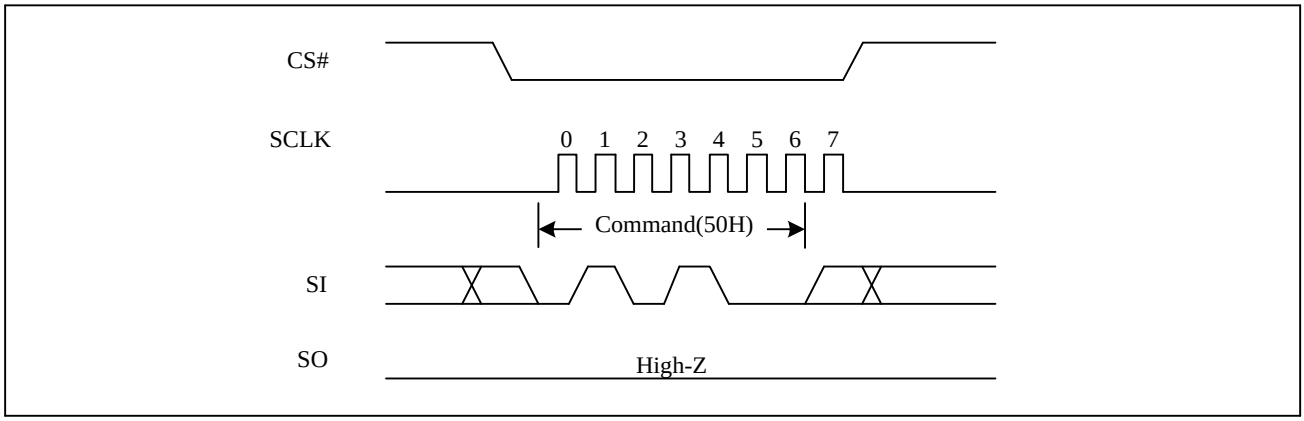
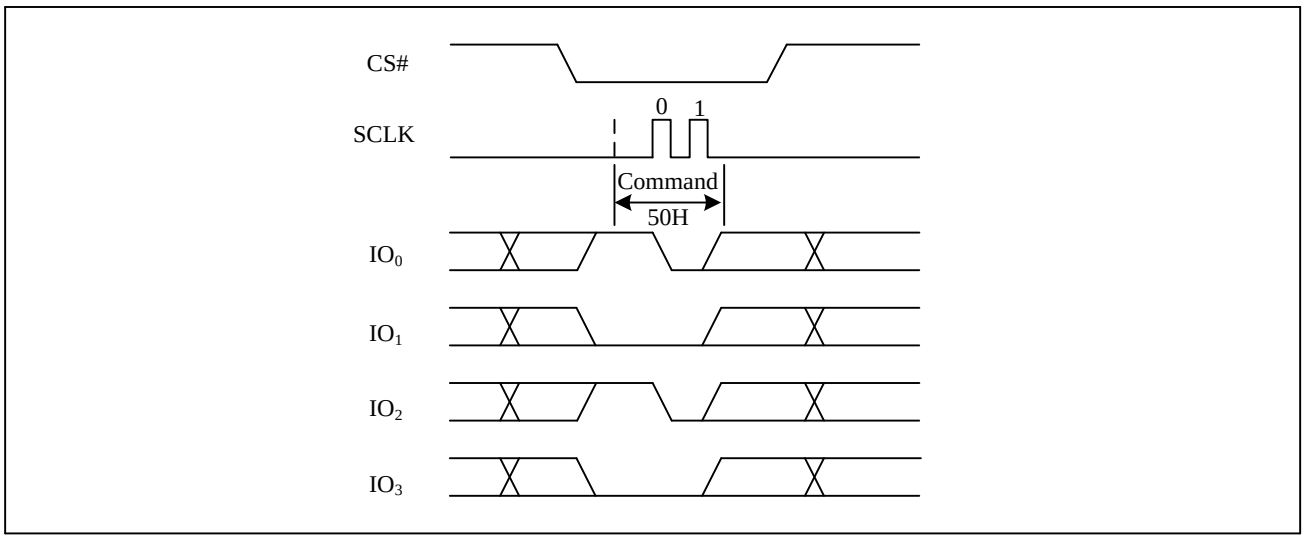



**Figure4a. Write Enable for Volatile Status Register Sequence Diagram (QPI)**



## 7.4. Read Status Register (RDSR) (05H or 35H or 15H)

The Read Status Register (RDSR) command is for reading the Status Register. The Status Register may be read at any time, even while a Program, Erase or Write Status Register cycle is in progress. When one of these cycles is in progress, it is recommended to check the Write in Progress (WIP) bit before sending a new command to the device. It is also possible to read the Status Register continuously. For command code “05H” / “35H”, the SO will output Status Register bits S7~S0 / S15-S8. The command code “15H” only supports the QPI mode, the I/O0 will output Status Register S1-S0.

**Figure5. Read Status Register Sequence Diagram**

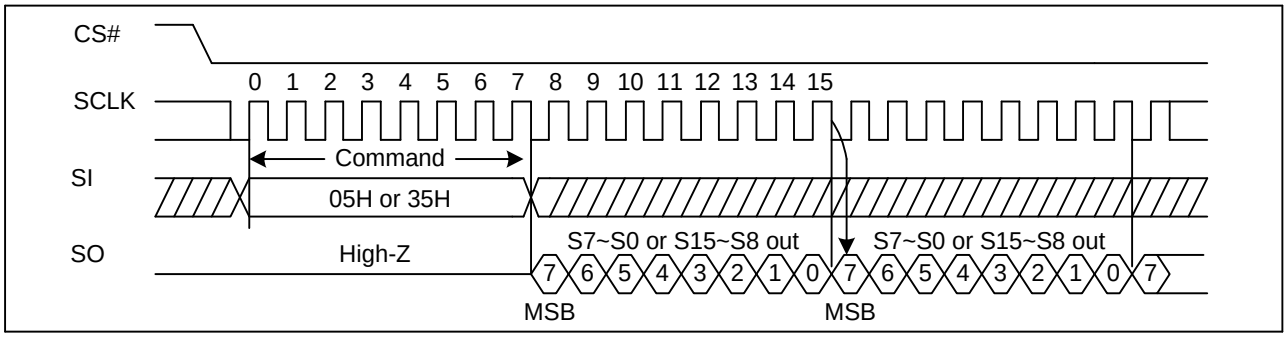



**Figure5a. Read Status Register Sequence Diagram (QPI)**

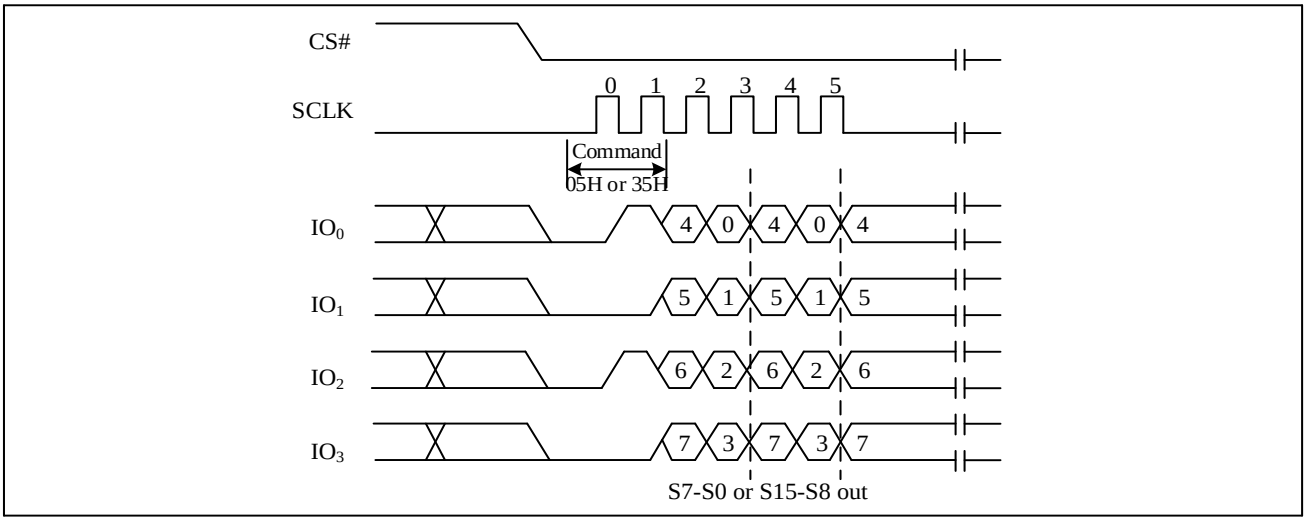
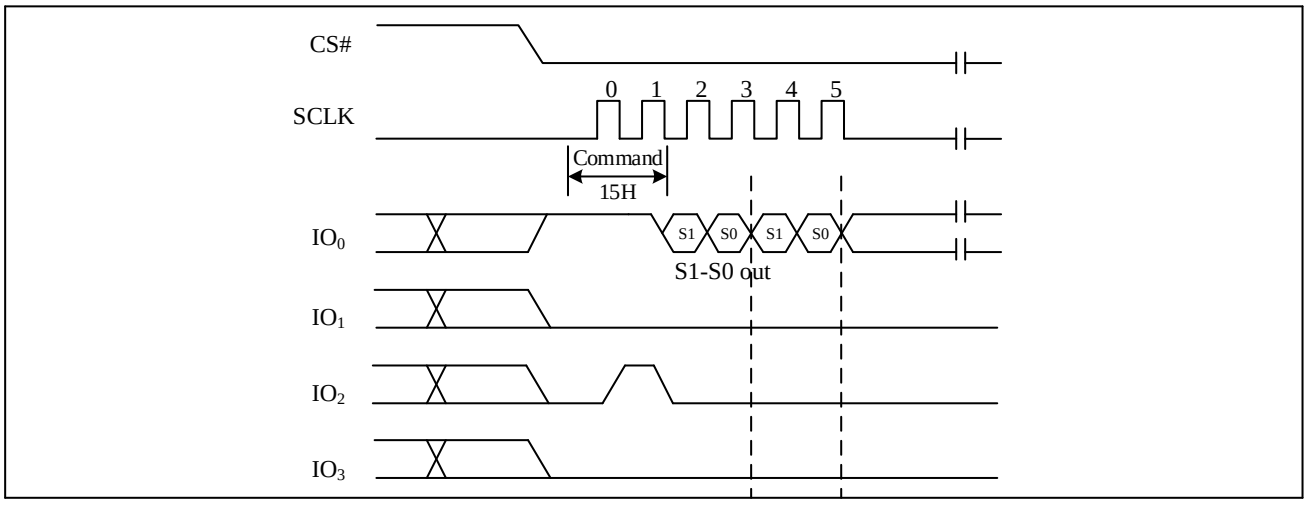



**Figure5b. Read Status Register Sequence Diagram (QPI) (15H)**



## 7.5. Write Status Register (WRSR) (01H)

The Write Status Register (WRSR) command allows new values to be written to the Status Register. Before it can be accepted, a Write Enable (WREN) command must previously have been executed. After the Write Enable (WREN) command has been decoded and executed, the device sets the Write Enable Latch (WEL).

The Write Status Register (WRSR) command has no effect on S15, S10, S1 and S0 of the Status Register. CS# must be driven high after the eighth or sixteenth bit of the data byte has been latched in. If not, the Write Status Register (WRSR) command is not executed. If CS# is driven high after eighth bit of the data byte, the CMP bit will be cleared to 0. As soon as CS# is driven high, the self-timed Write Status Register cycle (whose duration is $t_W$) is initiated. While the Write Status Register cycle is in progress, the Status Register may still be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Write Status Register cycle, and is 0 when it is completed. When the cycle is completed, the Write Enable Latch (WEL) is reset.

The Write Status Register (WRSR) command allows the user to change the values of the Block Protect (BP4, BP3, BP2, BP1, and BP0) bits, to define the size of the area that is to be treated as read-only, as defined in Table1.

**Figure6. Write Status Register Sequence Diagram**

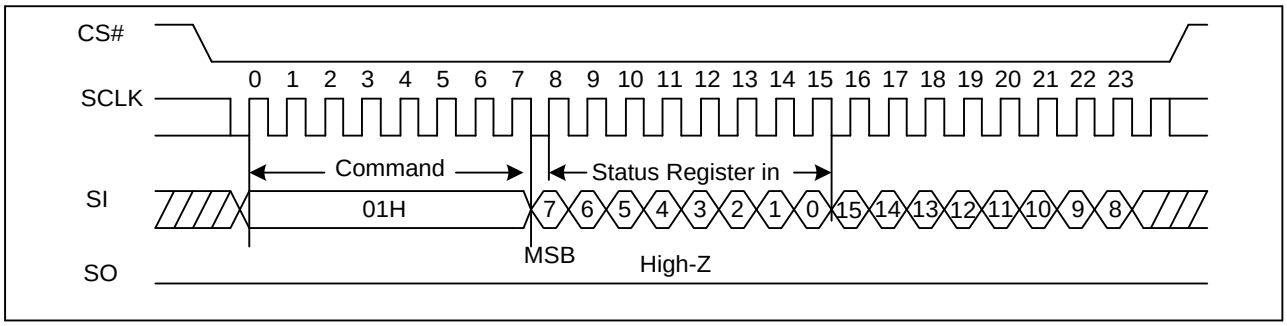
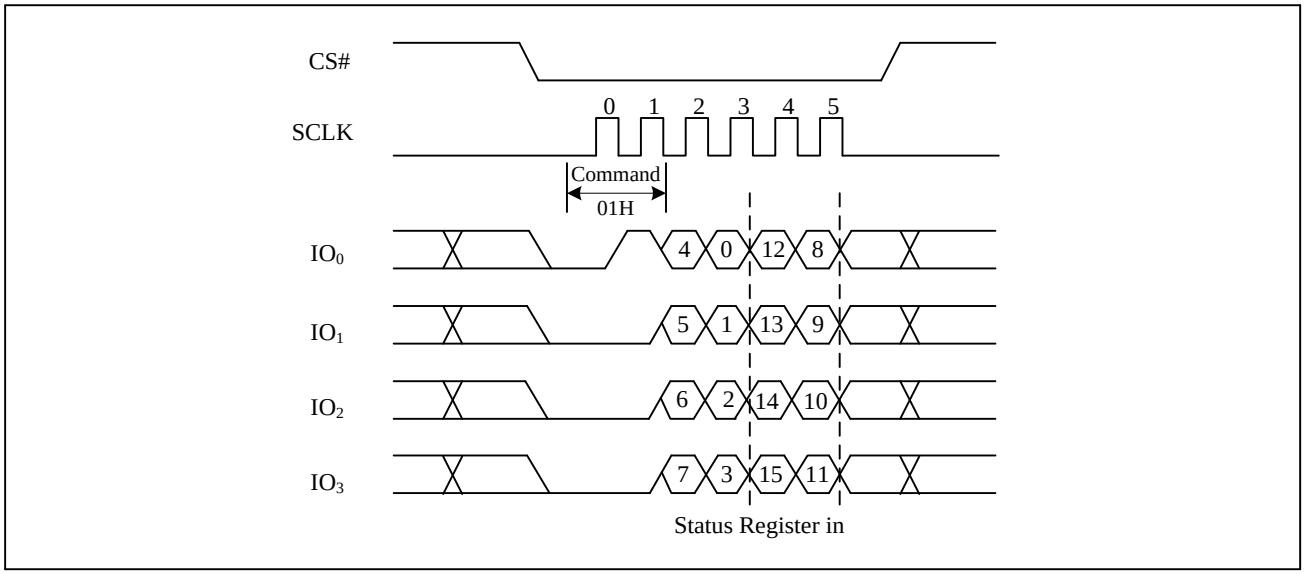



**Figure6a. Write Status Register Sequence Diagram (QPI)**

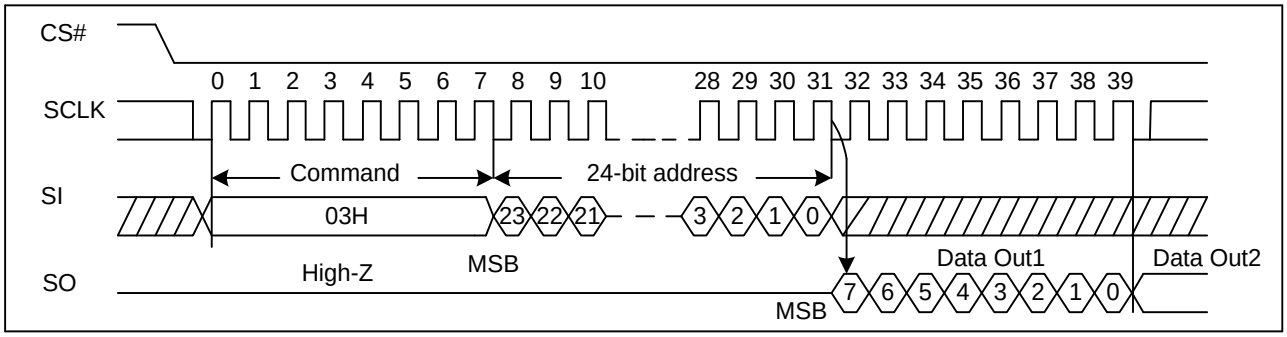



## 7.6. Read Data Bytes (READ) (03H)

The Read Data Bytes (READ) command is followed by a 3-byte address (A23-A0), and each bit is latched-in on the rising edge of SCLK. Then the memory content, at that address, is shifted out on SO, and each bit is shifted out, at a Max frequency $f_R$, on the falling edge of SCLK. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. The whole memory can, therefore, be read

with a single Read Data Bytes (READ) command. Any Read Data Bytes (READ) command, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

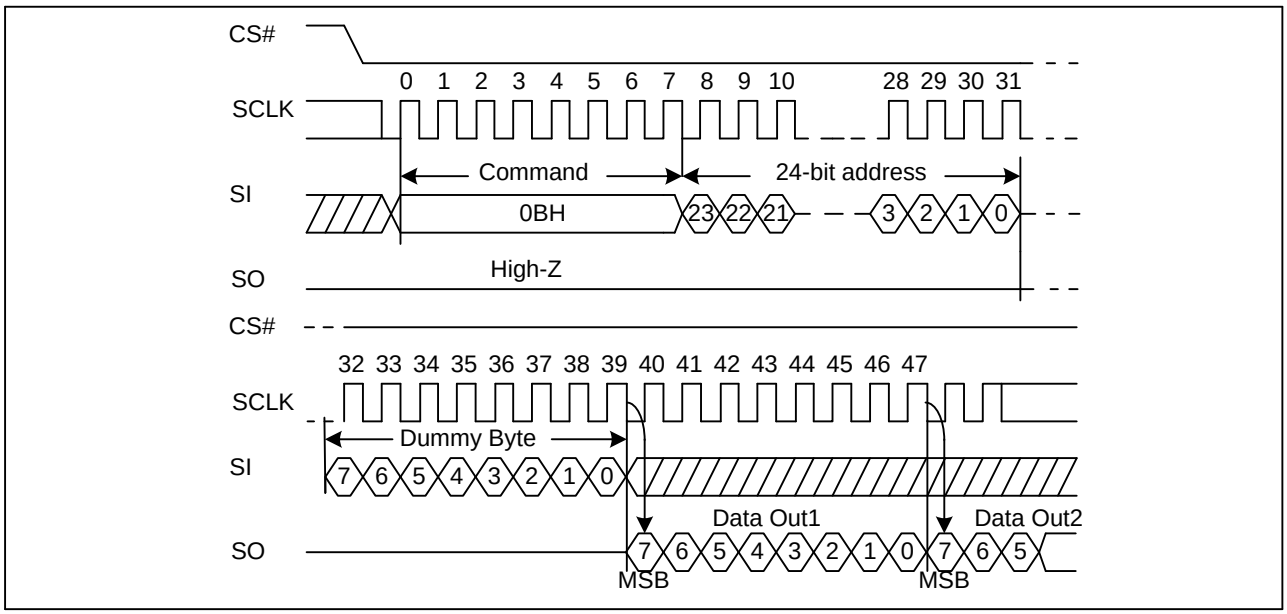
**Figure7. Read Data Bytes Sequence Diagram**




## 7.7. Read Data Bytes at Higher Speed (Fast Read) (0BH)

The Read Data Bytes at Higher Speed (Fast Read) command is for quickly reading data out. It is followed by a 3-byte address (A23-A0) and a dummy byte, and each bit is latched-in on the rising edge of SCLK. Then the memory content, at that address, is shifted out on SO, and each bit is shifted out, at a Max frequency $f_C$, on the falling edge of SCLK. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out.

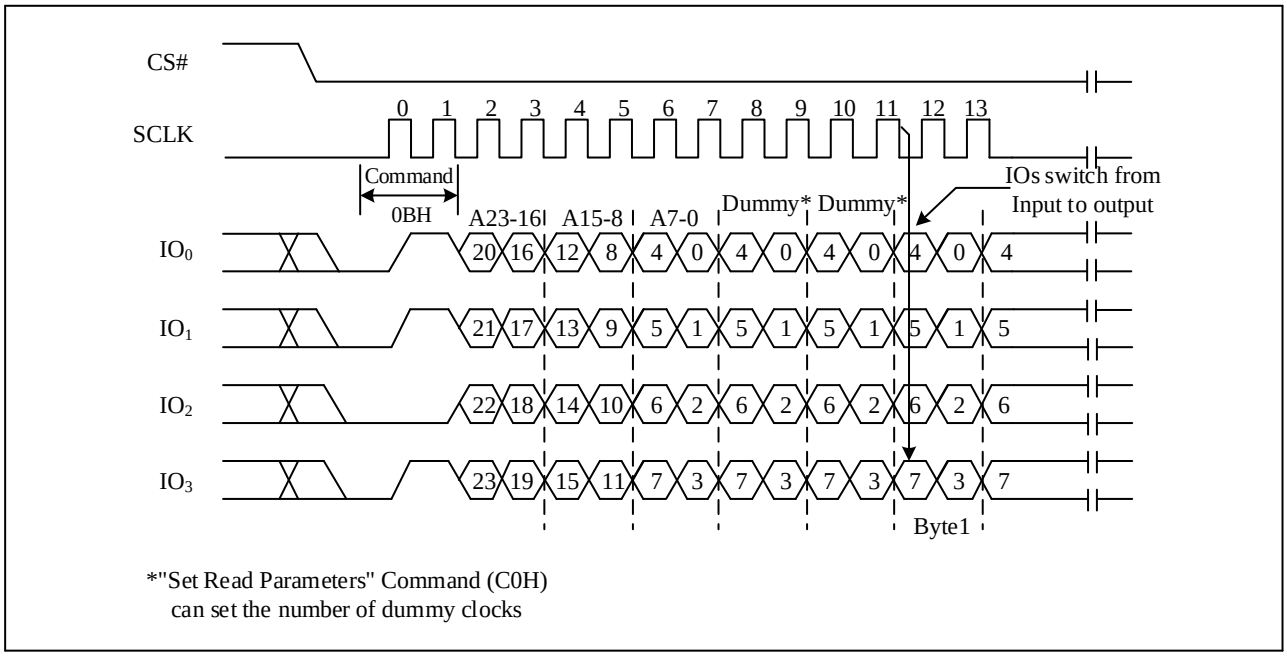
**Figure8. Read Data Bytes at Higher Speed Sequence Diagram**




### Fast Read (0BH) in QPI mode

The Fast Read command is also supported in QPI mode. In QPI mode, the number of dummy clocks is configured by the “Set Read Parameters (C0H)” command to accommodate a wide range application with different needs for either maximum Fast Read frequency or minimum data access latency. Depending on the Read Parameter Bits P[5:4] setting, the number of dummy clocks can be configured as either 4/6/8/8.

**Figure8a. Read Data Bytes at Higher Speed Sequence Diagram (QPI)**

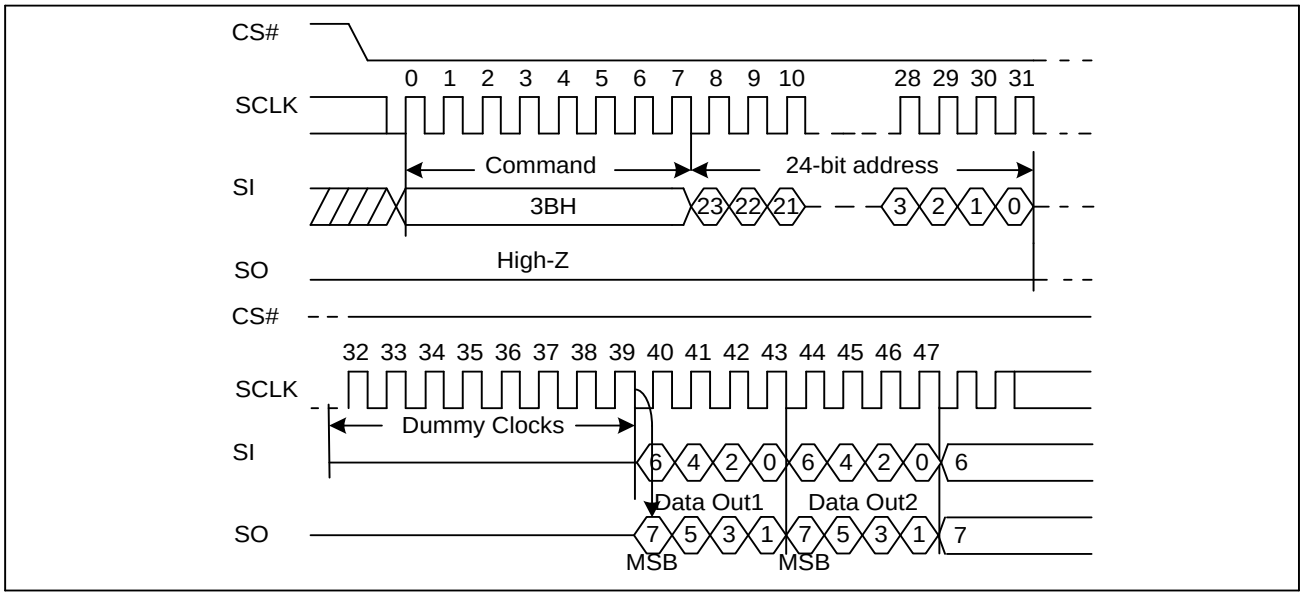



## 7.8. Dual Output Fast Read (3BH)

The Dual Output Fast Read command is followed by 3-byte address (A23-A0) and a dummy byte, and each bit is latched in on the rising edge of SCLK, then the memory contents are shifted out 2-bit per clock cycle from SI and SO.

The command sequence is shown in followed Figure9. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out.

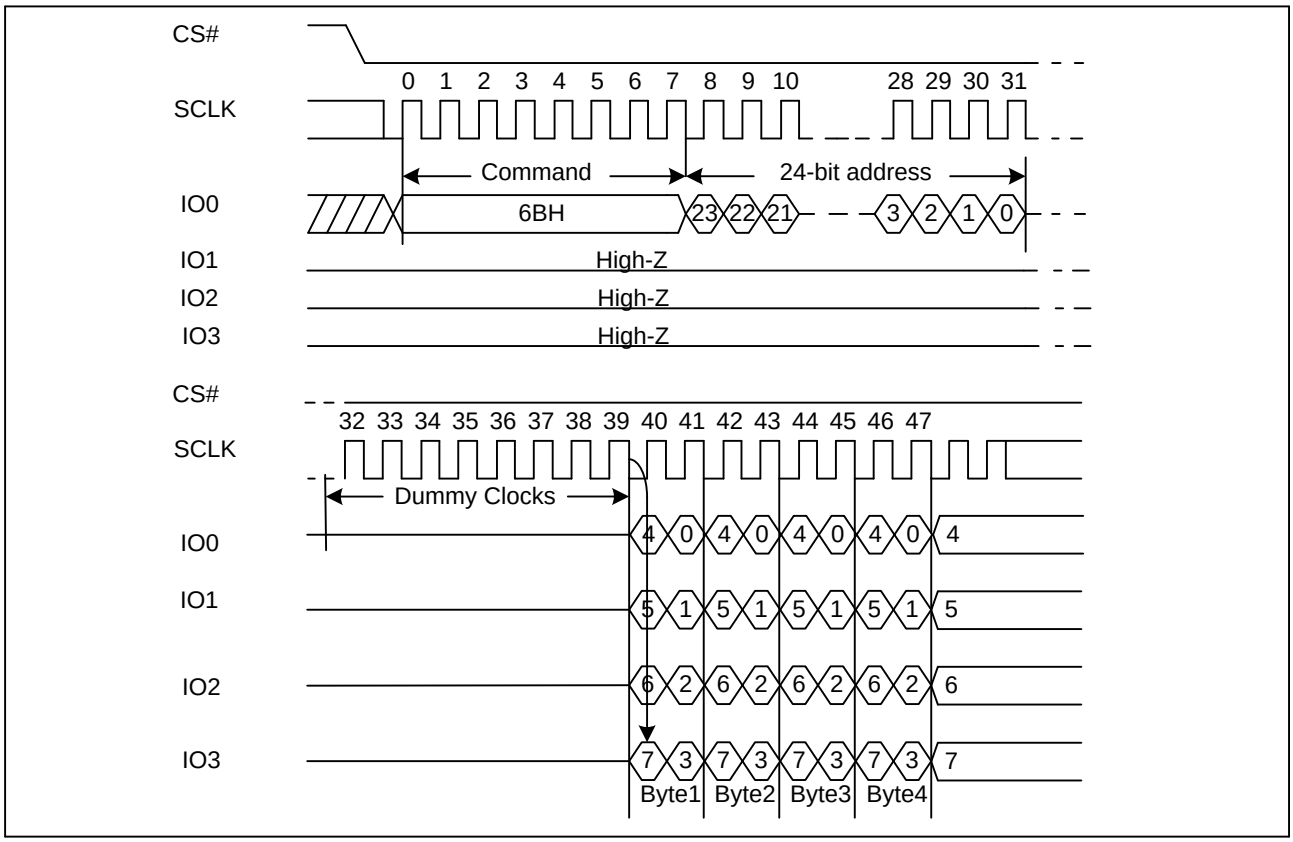
**Figure9. Dual Output Fast Read Sequence Diagram**



## 7.9. Quad Output Fast Read (6BH)

The Quad Output Fast Read command is followed by 3-byte address (A23-A0) and a dummy byte, and each bit is latched in on the rising edge of SCLK, then the memory contents are shifted out 4-bit per clock cycle from IO3, IO2, IO1 and IO0. The command sequence is shown in followed Figure10. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out.

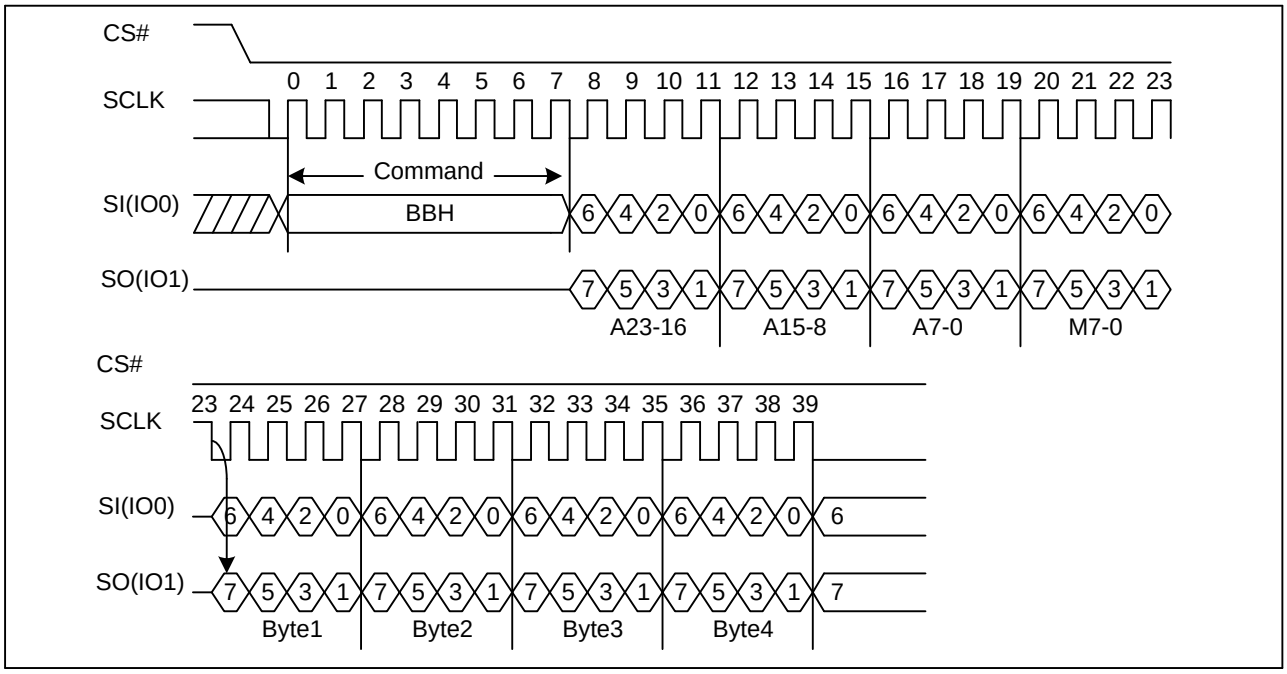
**Figure10. Quad Output Fast Read Sequence Diagram**




## 7.10. Dual I/O Fast Read (BBH)

The Dual I/O Fast Read command is similar to the Dual Output Fast Read command but with the capability to input the 3-byte address (A23-0) and M[7:0] 2-bit per clock by SI and SO, and each bit is latched in on the rising edge of SCLK, then the memory contents are shifted out 2-bit per clock cycle from SI and SO. The command sequence is shown in followed Figure11. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out.

**Figure11. Dual I/O Fast Read Sequence Diagram (M5-4≠ (1, 0))**

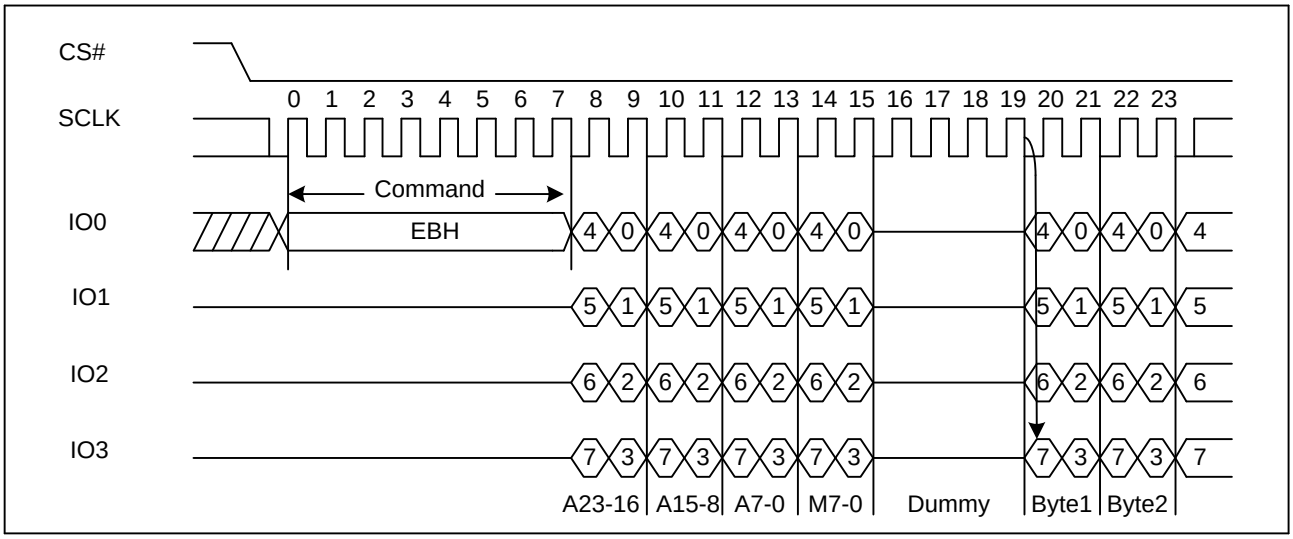



Note: M5-4 must **not** be set as (1, 0).

## 7.11. Quad I/O Fast Read (EBH)

The Quad I/O Fast Read command is similar to the Dual I/O Fast Read command but with the capability to input the 3-byte address (A23-0) ), M[7:0] and 4-dummy clock 4-bit per clock by IO0, IO1, IO2, IO3, and each bit is latched in on the rising edge of SCLK, then the memory contents are shifted out 4-bit per clock cycle from IO0, IO1, IO2, IO3. The command sequence is shown in followed Figure12. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out.

**Figure12. Quad I/O Fast Read Sequence Diagram (M5-4≠ (1, 0))**




Note: M5-4 must **not** be set as (1, 0).

### Quad I/O Fast Read with "8/16/32/64-Byte Wrap Around" in Standard SPI mode

The Quad I/O Fast Read command can be used to access a specific portion within a page by issuing "Set Burst with

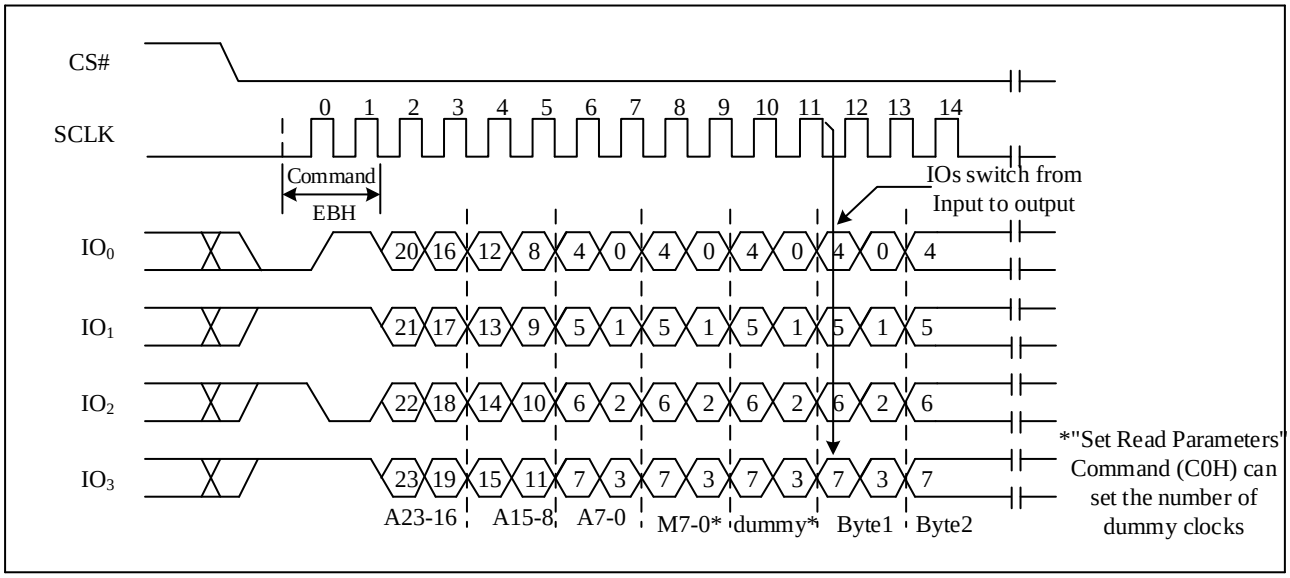
Wrap" (77H) commands prior to EBH. The "Set Burst with Wrap" (77H) command can either enable or disable the "Wrap Around" feature for the following EBH commands. When "Wrap Around" is enabled, the data being accessed can be limited to either an 8/16/32/64-byte section of a 256-byte page. The output data starts at the initial address specified in the command, once it reaches the ending boundary of the 8/16/32/64-byte section, the output will wrap around the beginning boundary automatically until CS# is pulled high to terminate the command.

The Burst with Wrap feature allows applications that use cache to quickly fetch a critical address and then fill the cache afterwards within a fixed length (8/16/32/64-byte) of data without issuing multiple read commands. The "Set Burst with Wrap" command allows three "Wrap Bits" W6-W4 to be set. The W4 bit is used to enable or disable the "Wrap Around" operation while W6-W5 is used to specify the length of the wrap around section within a page.

## Quad I/O Fast Read (EBH) in QPI mode

The Quad I/O Fast Read command is also supported in QPI mode. See Figure12b. In QPI mode, the number of dummy clocks is configured by the "Set Read Parameters (C0H)" command to accommodate a wide range application with different needs for either maximum Fast Read frequency or minimum data access latency. Depending on the Read Parameter Bits P[5:4] setting, the number of dummy clocks can be configured as either 4/6/8/8. "Wrap Around" feature is not available in QPI mode for Quad I/O Fast Read command. To perform a read operation with fixed data length wrap around in QPI mode, a dedicated "Burst Read with Wrap" (0CH) command must be used.

**Figure12b. Quad I/O Fast Read Sequence Diagram (M5-4≠ (1, 0) QPI)**




Note: M5-4 must **not** be set as (1, 0).

## 7.12. Set Burst with Wrap (77H)

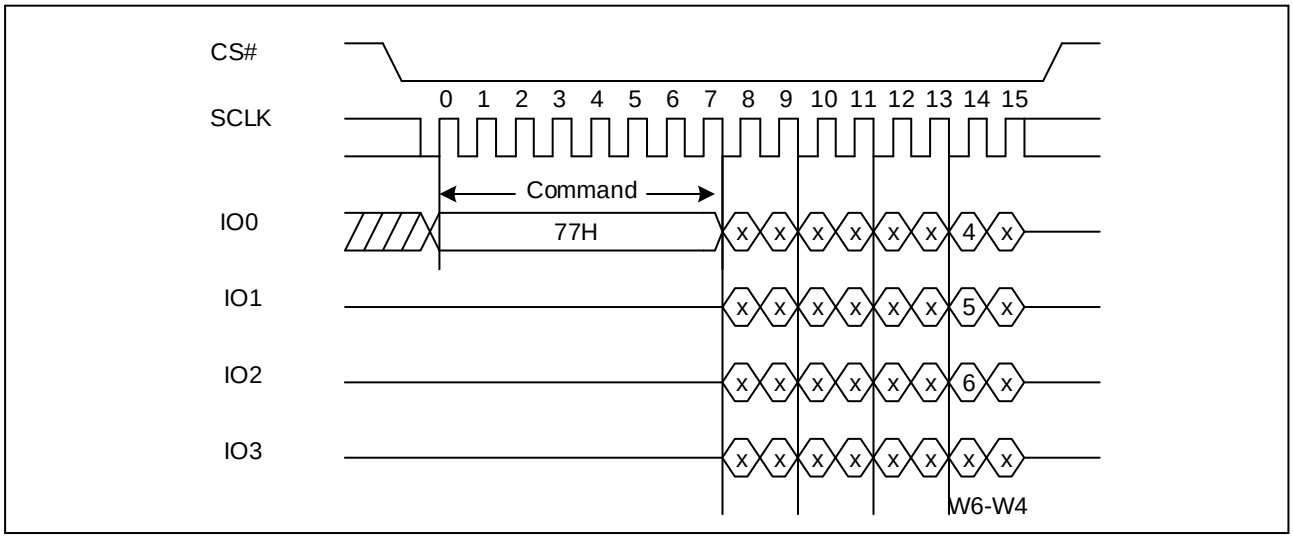
The Set Burst with Wrap command is used in conjunction with "Quad I/O Fast Read" command to access a fixed length of 8/16/32/64-byte section within a 256-byte page.

The Set Burst with Wrap command sequence: CS# goes low → Send Set Burst with Wrap command → Send 24 dummy bits → Send 8 bits "Wrap bits" → CS# goes high.

| W6,W5 | W4=0 | | W4=1 (default) | |
|---|---|---|---|---|
| | Wrap Around | Wrap Length | Wrap Around | Wrap Length |
| 0, 0 | Yes | 8-byte | No | N/A |
| 0, 1 | Yes | 16-byte | No | N/A |
| 1, 0 | Yes | 32-byte | No | N/A |
| 1, 1 | Yes | 64-byte | No | N/A |

If the W6-W4 bits are set by the Set Burst with Wrap command, all the following "Quad I/O Fast Read" command will use the W6-W4 setting to access the 8/16/32/64-byte section within any page. To exit the "Wrap Around" function and return to normal read operation, another Set Burst with Wrap command should be issued to set W4=1. In QPI mode, the "Burst Read with Wrap (0CH)" command should be used to perform the Read Operation with "Wrap Around" feature. The Wrap Length set by W5-W6 in Standard SPI mode is still valid in QPI mode and can also be re-configured by "Set Read Parameters (C0H) command.

**Figure13. Set Burst with Wrap Sequence Diagram**




## 7.13. Page Program (PP) (02H)

The Page Program (PP) command is for programming the memory. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit before sending the Page Program command.

The Page Program (PP) command is entered by driving CS# Low, followed by the command code, three address bytes and at least one data byte on SI. If the 8 least significant address bits (A7-A0) are not all zero, all transmitted data that goes beyond the end of the current page are programmed from the start address of the same page (from the address whose 8 least significant bits (A7-A0) are all zero). CS# must be driven low for the entire duration of the sequence. The Page Program command sequence: CS# goes low → sending Page Program command → 3-byte address on SI → at least 1 byte data on SI → CS# goes high. The command sequence is shown in Figure14. If more than 256 bytes are sent to the

device, previously latched data are discarded and the last 256 data bytes are guaranteed to be programmed correctly within the same page. If less than 256 data bytes are sent to device, they are correctly programmed at the requested addresses without having any effects on the other bytes of the same page. CS# must be driven high after the eighth bit of the last data byte has been latched in; otherwise the Page Program (PP) command is not executed.

As soon as CS# is driven high, the self-timed Page Program cycle (whose duration is $t_{PP}$) is initiated. While the Page Program cycle is in progress, the Status Register may be read to check the value of the Write in Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Page Program cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Page Program (PP) command applied to a page which is protected by the Block Protect (BP4, BP3, BP2, BP1, and BP0) is not executed.

**Figure14. Page Program Sequence Diagram**

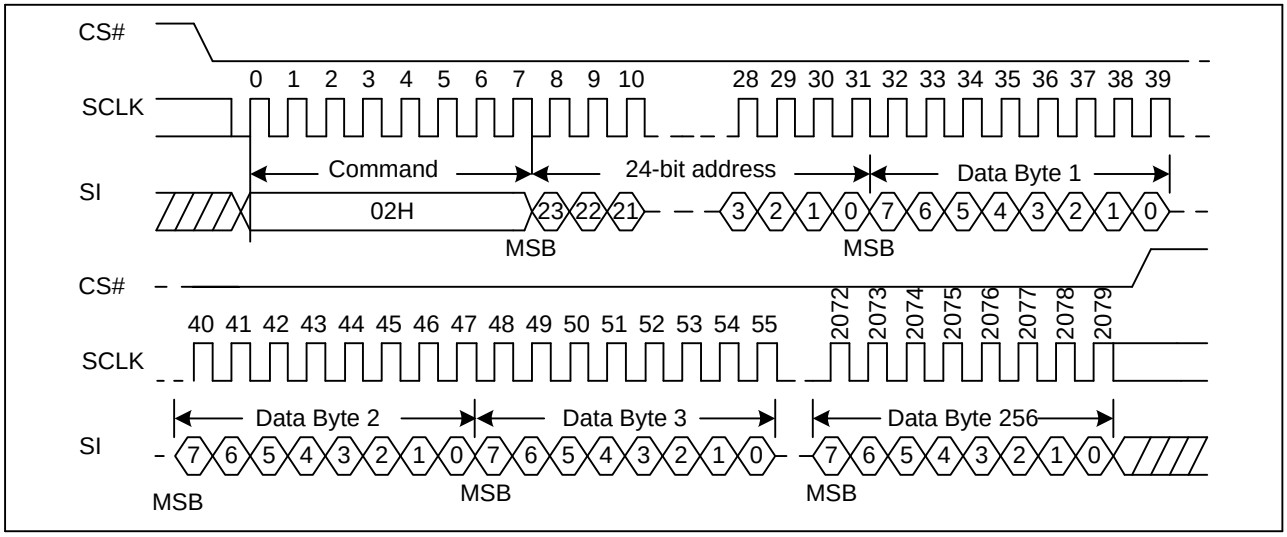
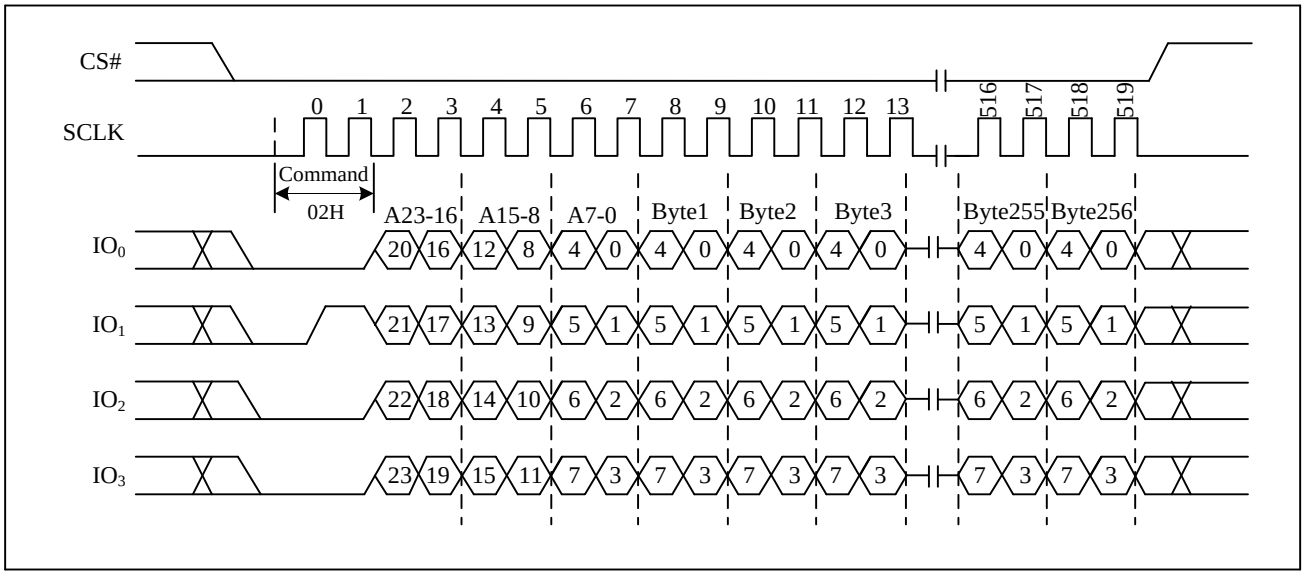



**Figure14a. Page Program Sequence Diagram (QPI)**



## 7.14. Quad Page Program (32H)

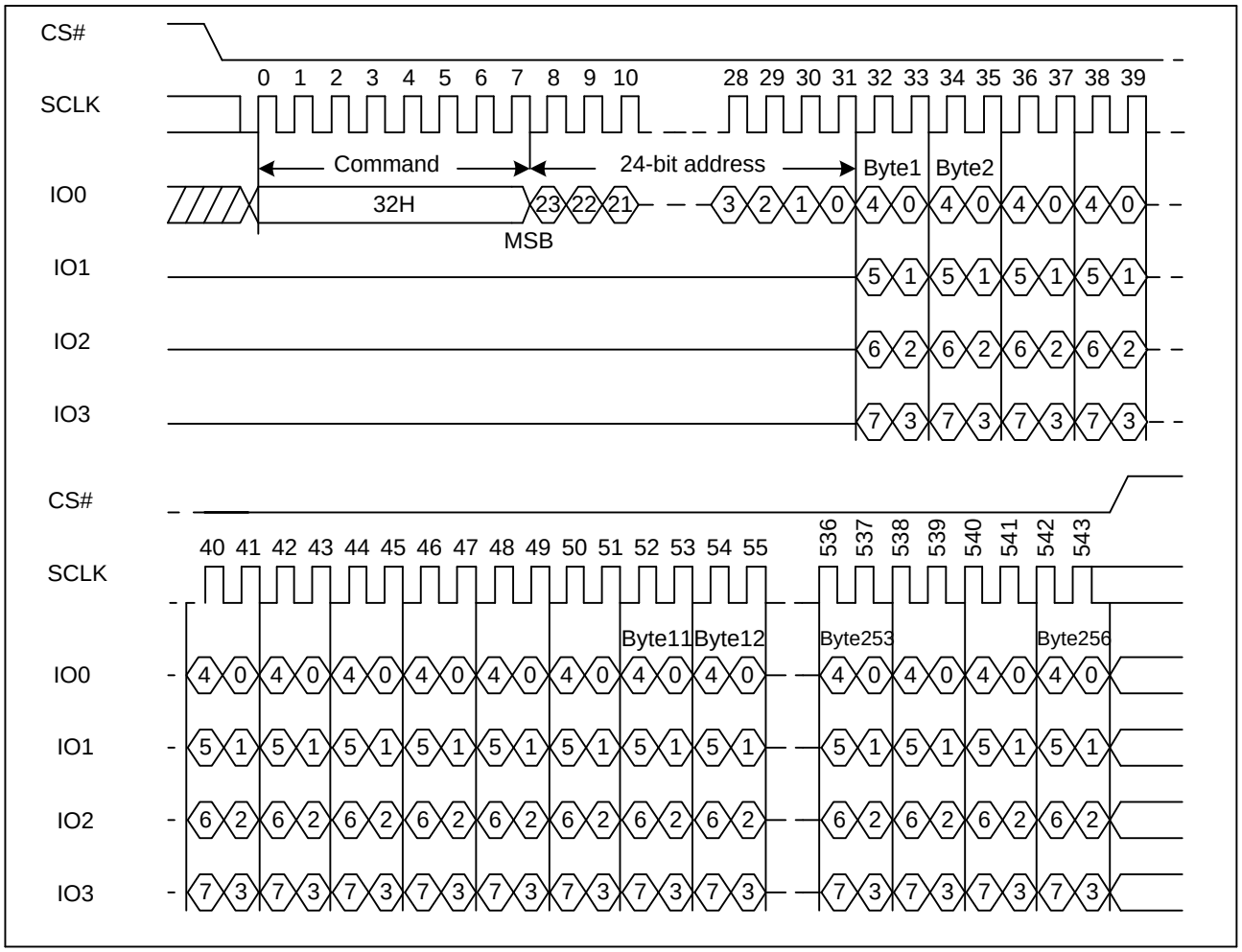
The Quad Page Program command is for programming the memory using four pins: IO0, IO1, IO2, and IO3. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit before sending the Page Program command. The quad Page Program command is entered by driving CS# Low, followed by the command code (32H), three address bytes and at least one data byte on IO pins.

The command sequence is shown in Figure15. If more than 256 bytes are sent to the device, previously latched data are discarded and the last 256 data bytes are guaranteed to be programmed correctly within the same page. If less than 256 data bytes are sent to device, they are correctly programmed at the requested addresses without having any effects on the other bytes of the same page. CS# must be driven high after the eighth bit of the last data byte has been latched in; otherwise the Quad Page Program (PP) command is not executed.

As soon as CS# is driven high, the self-timed Quad Page Program cycle (whose duration is $t_{PP}$) is initiated. While the Quad Page Program cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Quad Page Program cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Quad Page Program command applied to a page which is protected by the Block Protect (BP4, BP3, BP2, BP1, and BP0) is not executed.

**Figure15.Quad Page Program Sequence Diagram**



## 7.15. Sector Erase (SE) (20H)

The Sector Erase (SE) command is erased the all data of the chosen sector. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit. The Sector Erase (SE) command is entered by driving CS# low, followed by the command code, and 3-address byte on SI. Any address inside the sector is a valid address for the Sector Erase (SE) command. CS# must be driven low for the entire duration of the sequence.

The Sector Erase command sequence: CS# goes low → sending Sector Erase command → 3-byte address on SI → CS# goes high. The command sequence is shown in Figure16. CS# must be driven high after the eighth bit of the last address byte has been latched in; otherwise the Sector Erase (SE) command is not executed. As soon as CS# is driven high, the self-timed Sector Erase cycle (whose duration is $t_{SE}$) is initiated. While the Sector Erase cycle is in progress, the Status Register may be read to check the value of the Write in Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Sector Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. A Sector Erase (SE) command applied to a sector which is protected by the Block Protect (BP4, BP3, BP2, BP1, and BP0) bit (see Table1&1a) is not executed.

**Figure16. Sector Erase Sequence Diagram**

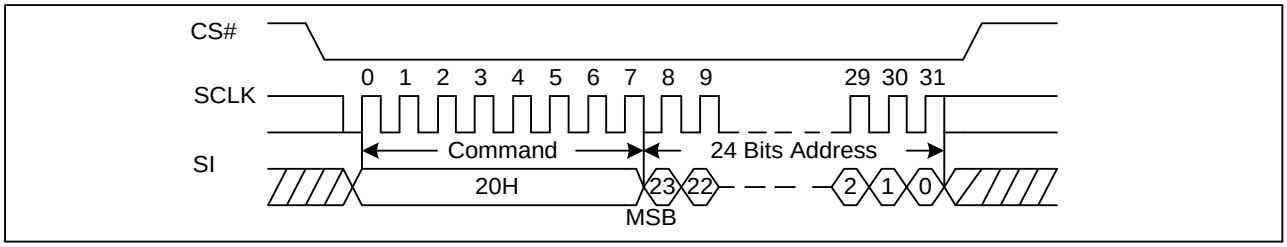
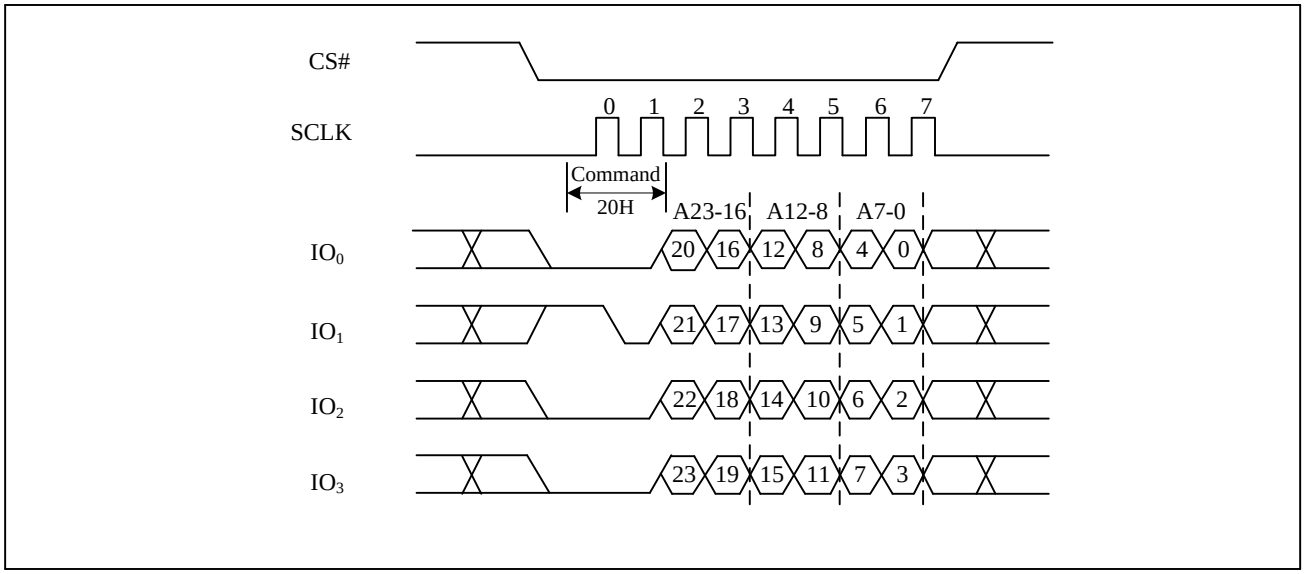



**Figure16a. Sector Erase Sequence Diagram (QPI)**



## 7.16. 32KB Block Erase (BE) (52H)

The 32KB Block Erase (BE) command is erased the all data of the chosen block. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit. The 32KB Block Erase (BE) command is entered by driving CS# low, followed by the command code, and three address bytes on SI. Any address inside the block is a valid address for the 32KB Block Erase (BE) command. CS# must be driven low for the entire duration of the sequence.

The 32KB Block Erase command sequence: CS# goes low → sending 32KB Block Erase command → 3-byte address on SI → CS# goes high. The command sequence is shown in Figure17. CS# must be driven high after the eighth bit of the last address byte has been latched in; otherwise the 32KB Block Erase (BE) command is not executed. As soon as CS# is driven high, the self-timed Block Erase cycle (whose duration is $t_{SE}$) is initiated. While the Block Erase cycle is in progress, the Status Register may be read to check the value of the Write in Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Block Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. A 32KB Block Erase (BE) command applied to a block which is protected by the Block Protect (BP4, BP3, BP2, BP1, and BP0) bits (see Table1&1a) is not executed.

**Figure17. 32KB Block Erase Sequence Diagram**

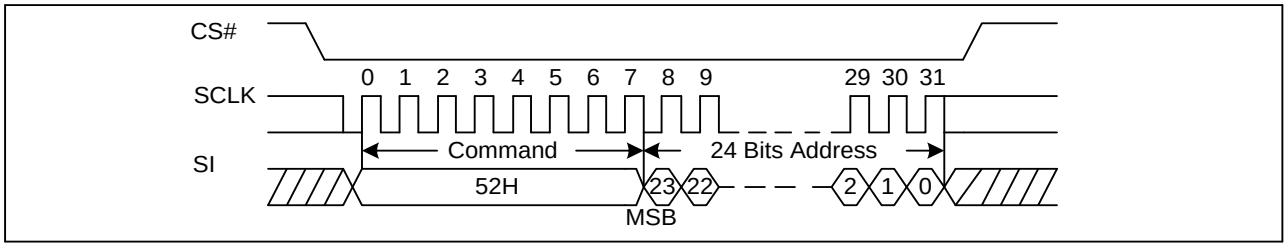
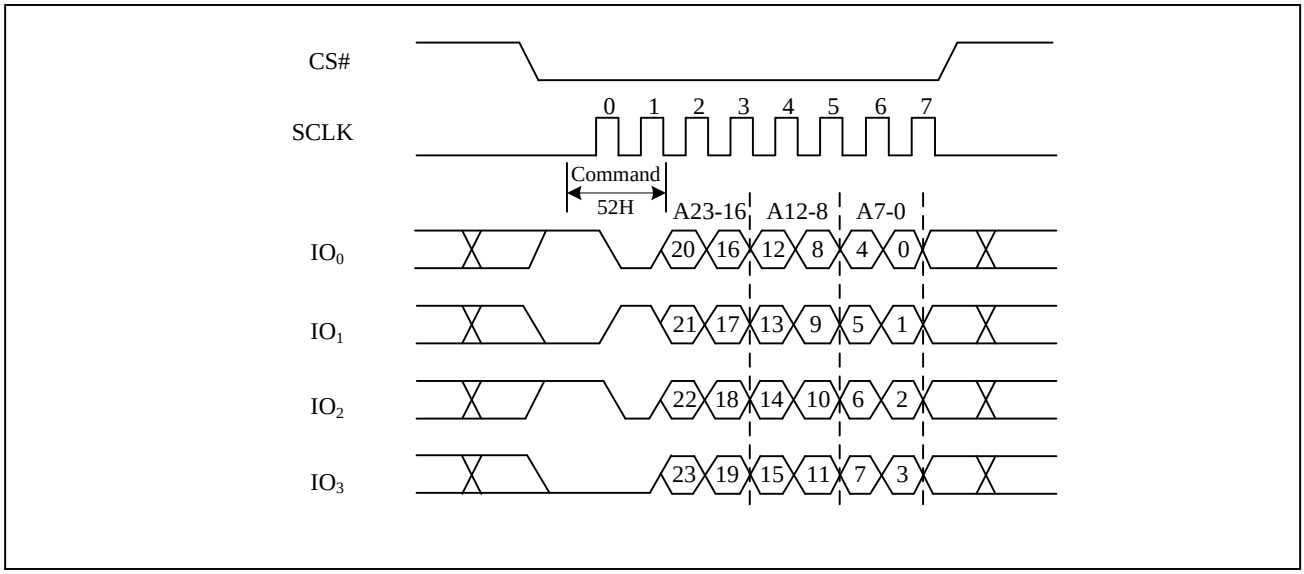



**Figure17a. 32KB Block Erase Sequence Diagram (QPI)**



## 7.17. 64KB Block Erase (BE) (D8H)

The 64KB Block Erase (BE) command is erased the all data of the chosen block. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit. The 64KB Block Erase (BE) command is entered by driving CS# low, followed by the command code, and three address bytes on SI. Any address inside the block is a valid address for the 64KB Block Erase (BE) command. CS# must be driven low for the entire duration of the sequence.

The 64KB Block Erase command sequence: CS# goes low → sending 64KB Block Erase command → 3-byte address on SI → CS# goes high. The command sequence is shown in Figure18. CS# must be driven high after the eighth bit of the last address byte has been latched in; otherwise the 64KB Block Erase (BE) command is not executed. As soon as CS# is driven high, the self-timed Block Erase cycle (whose duration is $t_{SE}$) is initiated. While the Block Erase cycle is in progress, the Status Register may be read to check the value of the Write in Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Block Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. A 64KB Block Erase (BE) command applied to a block which is protected by the Block Protect (BP4, BP3, BP2, BP1, and BP0) bits (see Table1&1a) is not executed.

**Figure18. 64KB Block Erase Sequence Diagram**

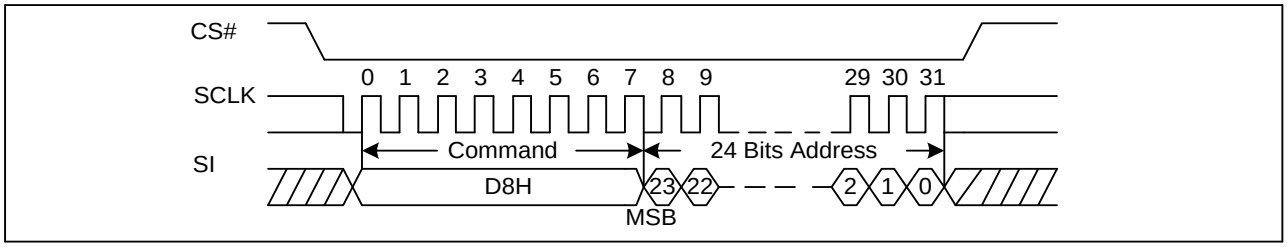
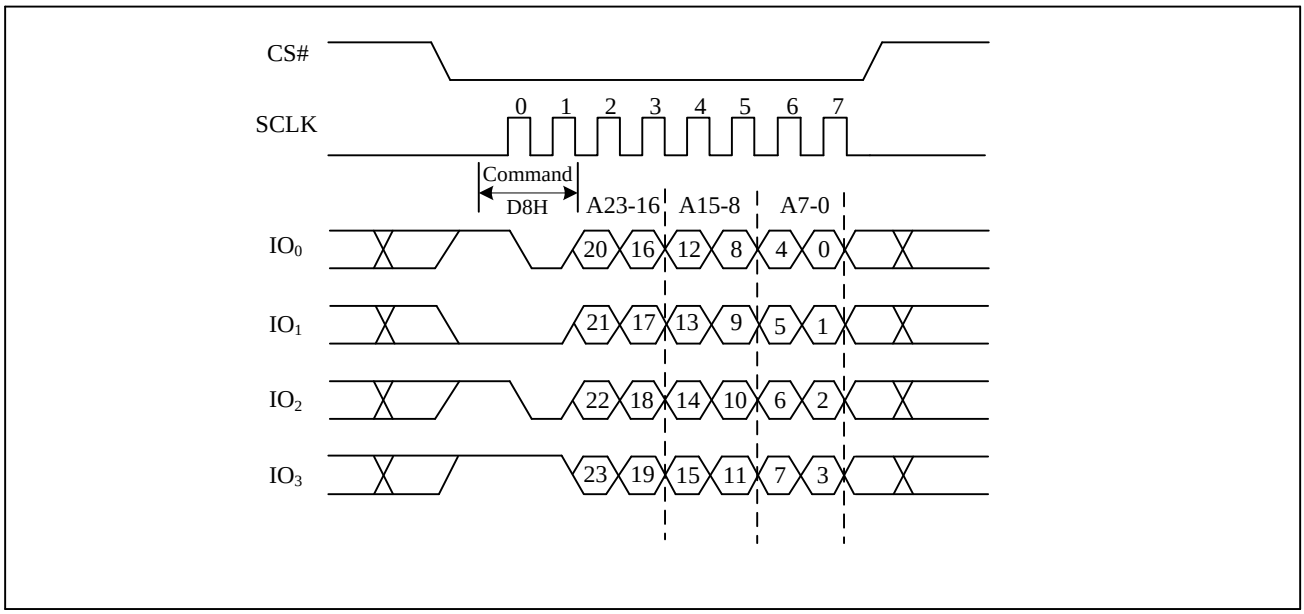



**Figure18a. 64KB Block Erase Sequence Diagram (QPI)**



## 7.18. Chip Erase (CE) (60/C7H)

The Chip Erase (CE) command is erased the all data of the chip. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit .The Chip Erase (CE) command is entered by driving CS# Low, followed by the command code on Serial Data Input (SI). CS# must be driven Low for the entire duration of the sequence.

The Chip Erase command sequence: CS# goes low → sending Chip Erase command → CS# goes high. The command sequence is shown in Figure19. CS# must be driven high after the eighth bit of the command code has been latched in; otherwise the Chip Erase command is not executed. As soon as CS# is driven high, the self-timed Chip Erase cycle (whose duration is $t_{CE}$) is initiated. While the Chip Erase cycle is in progress, the Status Register may be read to check the value of the Write in Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Chip Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. The Chip Erase (CE) command is executed if the Block Protect (BP2, BP1, and BP0) bits are 0 and CMP=0 or the Block Protect (BP2, BP1, and BP0) bits are 1 and CMP=1. The Chip Erase (CE) command is ignored if one or more sectors are protected.

**Figure19. Chip Erase Sequence Diagram**

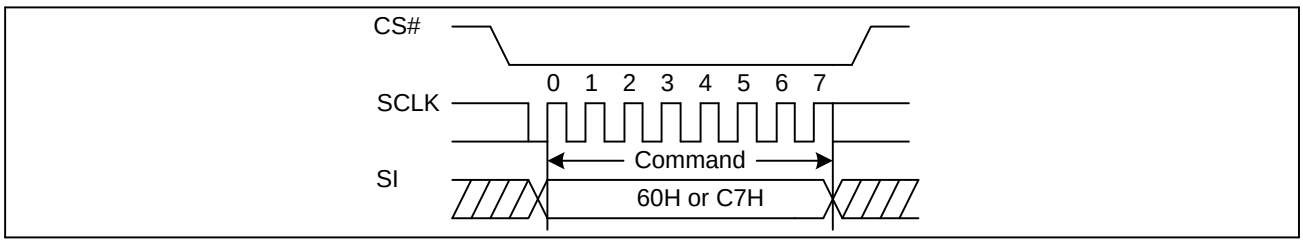
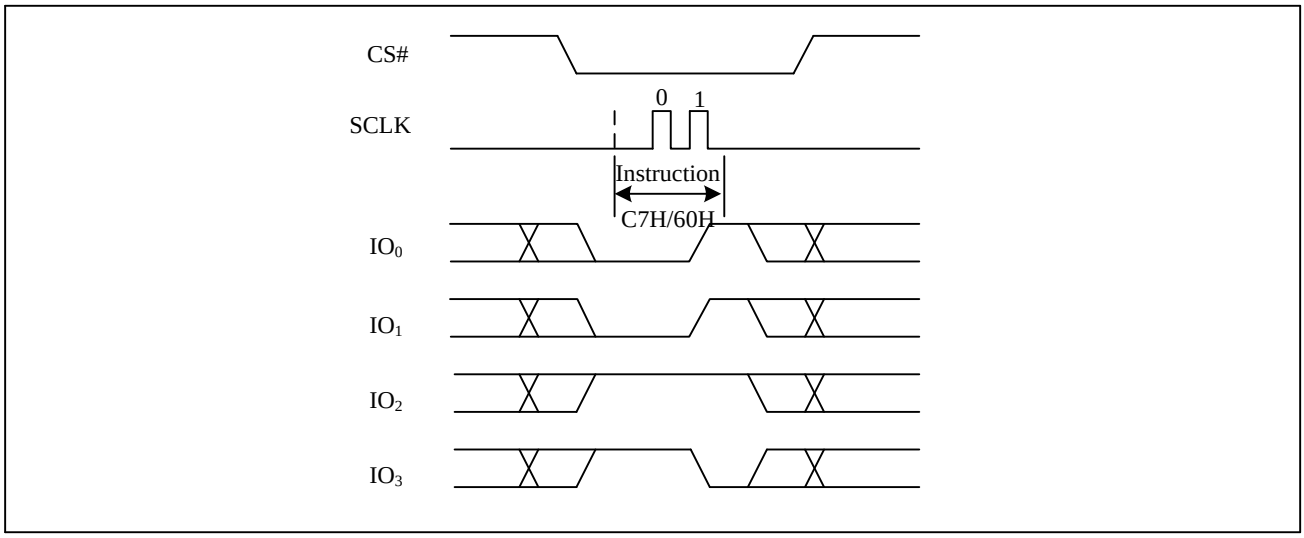



**Figure19a. Chip Erase Sequence Diagram (QPI)**



## 7.19. Deep Power-Down (DP) (B9H)

Executing the Deep Power-Down (DP) command is the only way to put the device in the lowest consumption mode (the Deep Power-Down Mode). It can also be used as an extra software protection mechanism, while the device is not in active use, since in this mode, the device ignores all Write, Program and Erase commands. Driving CS# high deselects the device, and puts the device in the Standby Mode (if there is no internal cycle currently in progress). But this mode is not the Deep Power-Down Mode. The Deep Power-Down Mode can only be entered by executing the Deep Power-Down (DP) command. Once the device has entered the Deep Power-Down Mode, all commands are ignored except the Release from Deep Power-Down and Read Device ID (RDI) command or Enable Reset (66H) and Reset (99H) commands.. These commands can release the device from this mode. The Release from Deep Power-Down and Read Device ID (RDI) command releases the device from deep power down mode , also allows the Device ID of the device to be output on SO.

The Deep Power-Down Mode automatically stops at Power-Down, and the device is in the Standby Mode after Power-Up.

The Deep Power-Down command sequence: CS# goes low → sending Deep Power-Down command → CS# goes high. The command sequence is shown in Figure20. CS# must be driven high after the eighth bit of the command code has been latched in; otherwise the Deep Power-Down (DP) command is not executed. As soon as CS# is driven high, it requires a delay of $t_{DP}$ before the supply current is reduced to $I_{CC2}$ and the Deep Power-Down Mode is entered. Any Deep Power-Down (DP) command, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

**Figure20. Deep Power-Down Sequence Diagram**

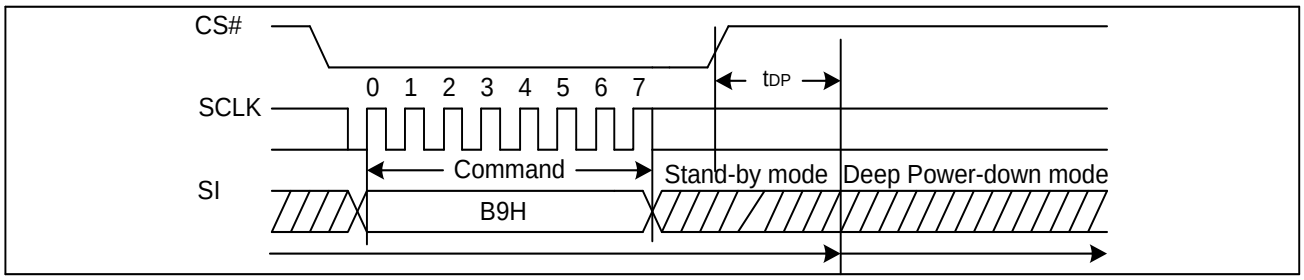
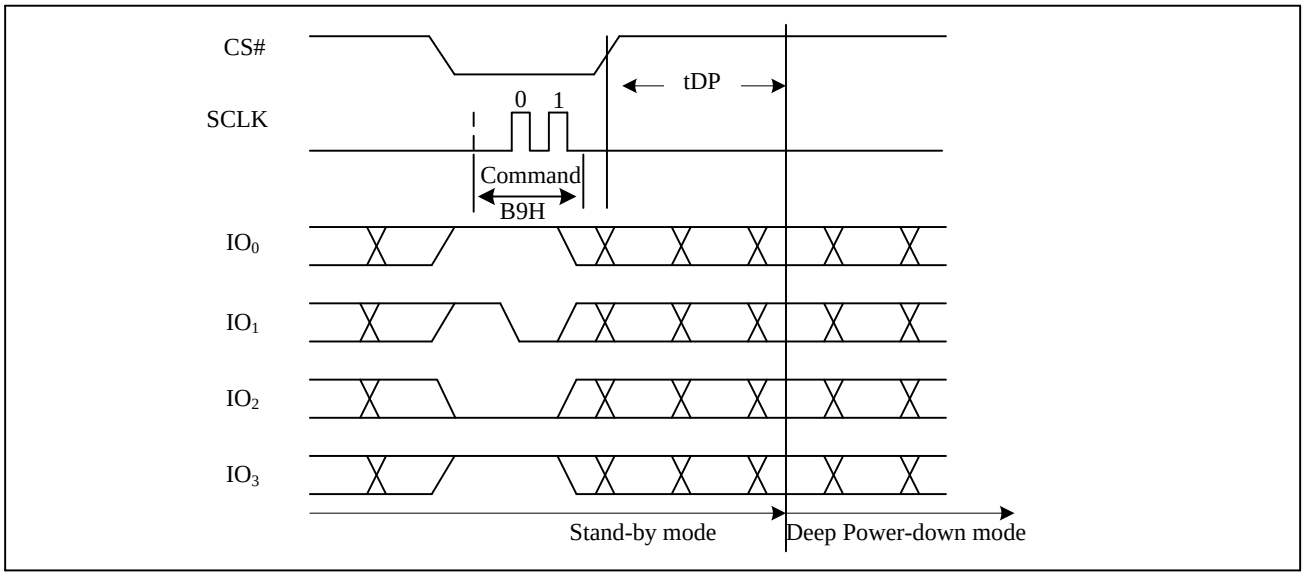



**Figure20a. Deep Power-Down Sequence Diagram (QPI)**



## 7.20. Release from Deep Power-Down and Read Device ID (RDI) (ABH)

The Release from Power-Down and Read Device ID command is a multi-purpose command. It can be used to release the device from the Power-Down state or obtain the devices electronic identification (ID) number.

To release the device from the Power-Down state, the command is issued by driving the CS# pin low, shifting the instruction code "ABH" and driving CS# high as shown in Figure21. Release from Power-Down will take the time duration of $t_{RES1}$ (See AC Characteristics) before the device will resume normal operation and other command are accepted. The CS# pin must remain high during the $t_{RES1}$ time duration.

When used only to obtain the Device ID while not in the Power-Down state, the command is initiated by driving the CS# pin low and shifting the instruction code "ABH" followed by 3-dummy byte. The Device ID bits are then shifted out on the falling edge of SCLK with most significant bit (MSB) first as shown in Figure22. The Device ID value for the GD25LR128D is listed in Manufacturer and Device Identification table. The Device ID can be read continuously. The command is completed by driving CS# high.

When used to release the device from the Power-Down state and obtain the Device ID, the command is the same as previously described, and shown in Figure22, except that after CS# is driven high it must remain high for a time duration of $t_{RES2}$ (See AC Characteristics). After this time duration the device will resume normal operation and other command will be accepted. If the Release from Power-Down / Device ID command is issued while an Erase, Program or Write cycle is in process (when WIP equal 1) the command is ignored and will not have any effects on the current cycle.

**Figure21. Release Power-Down Sequence Diagram**

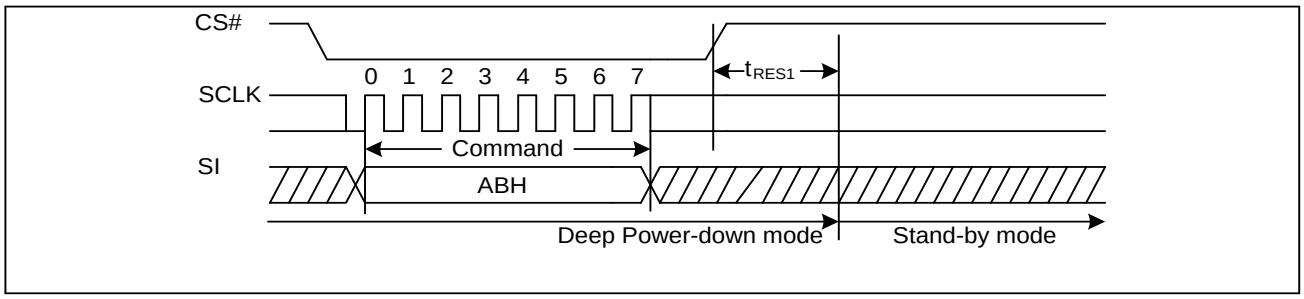



**Figure21a. Release Power-Down Sequence Diagram (QPI)**

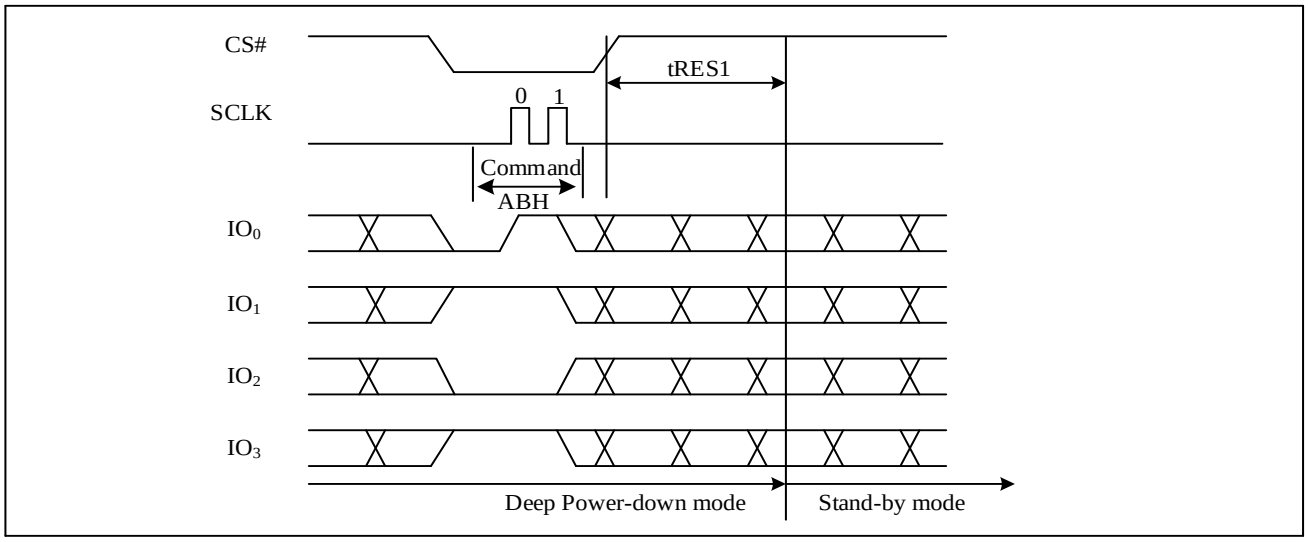


**Figure22. Release Power-Down/Read Device ID Sequence Diagram**

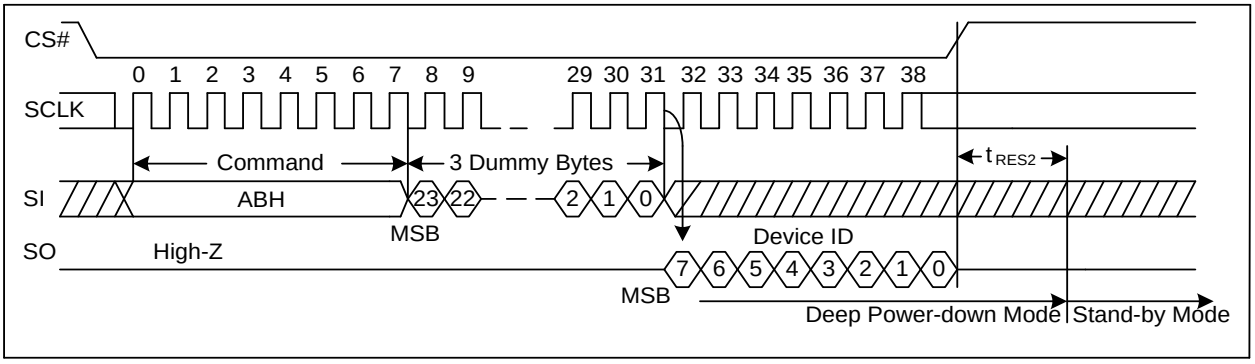
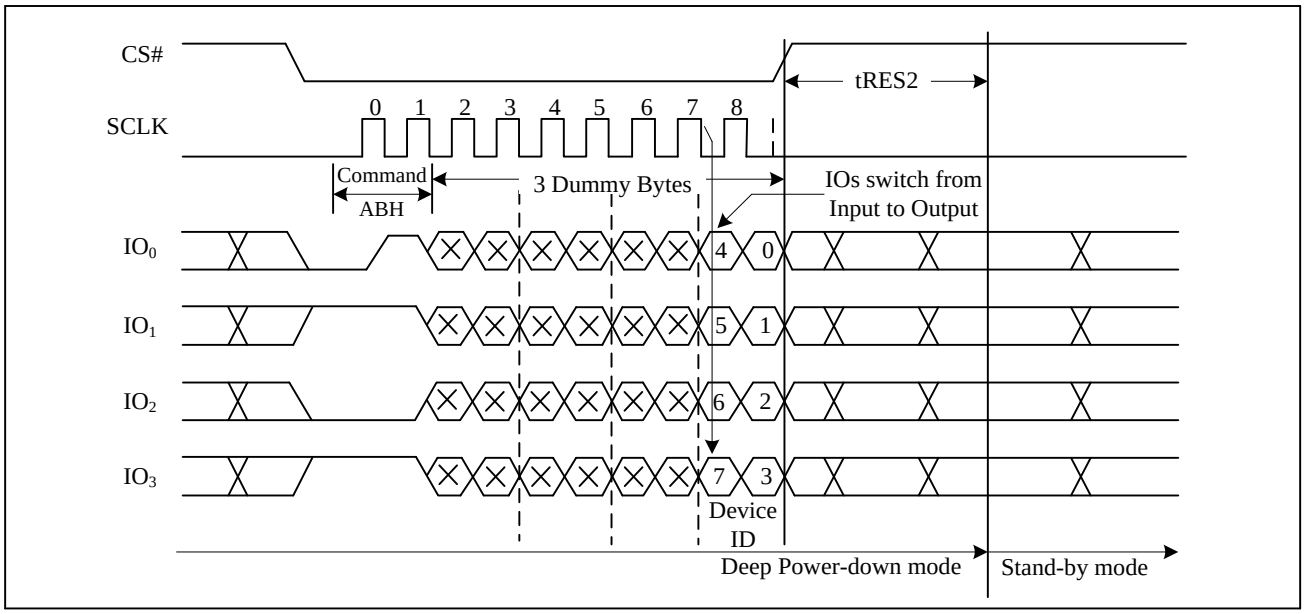



**Figure22a. Release Power-Down/Read Device ID Sequence Diagram (QPI)**



## 7.21. Read Manufacture ID/ Device ID (REMS) (90H)

The Read Manufacturer/Device ID command is an alternative to the Release from Power-Down / Device ID command that provides both the JEDEC assigned Manufacturer ID and the specific Device ID.

The command is initiated by driving the CS# pin low and shifting the command code "90H" followed by a 24-bit address (A23-A0) of 000000H. After which, the Manufacturer ID and the Device ID are shifted out on the falling edge of SCLK with most significant bit (MSB) first as shown in Figure23. If the 24-bit address is initially set to 000001H, the Device ID will be read first.

**Figure23. Read Manufacture ID/ Device ID Sequence Diagram**

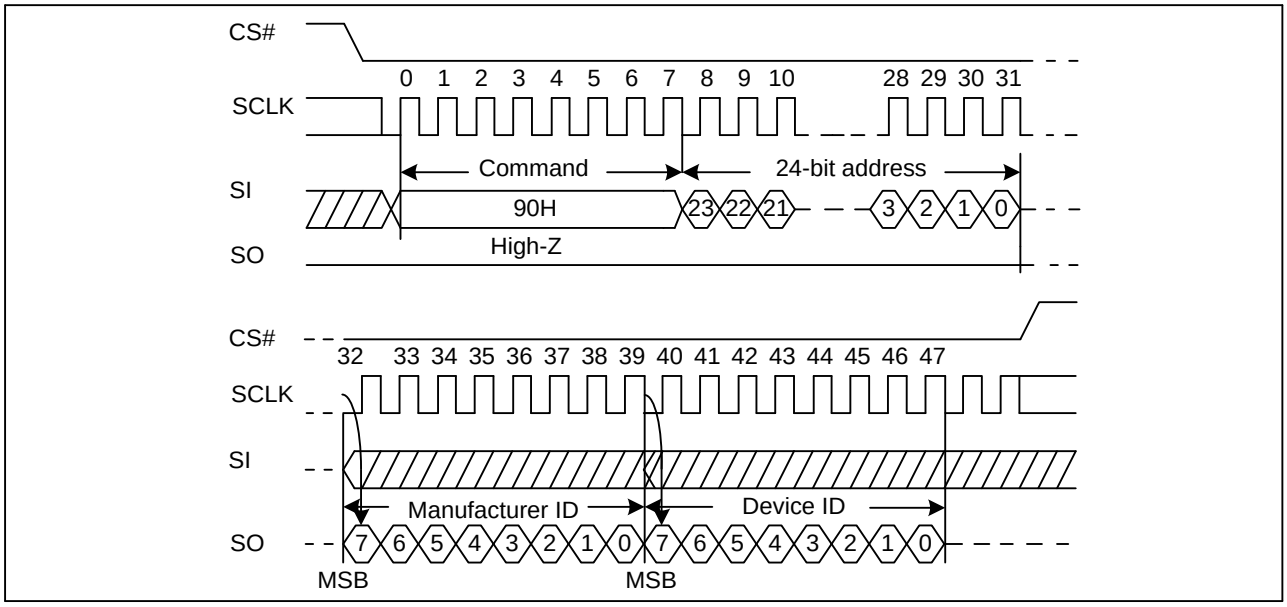
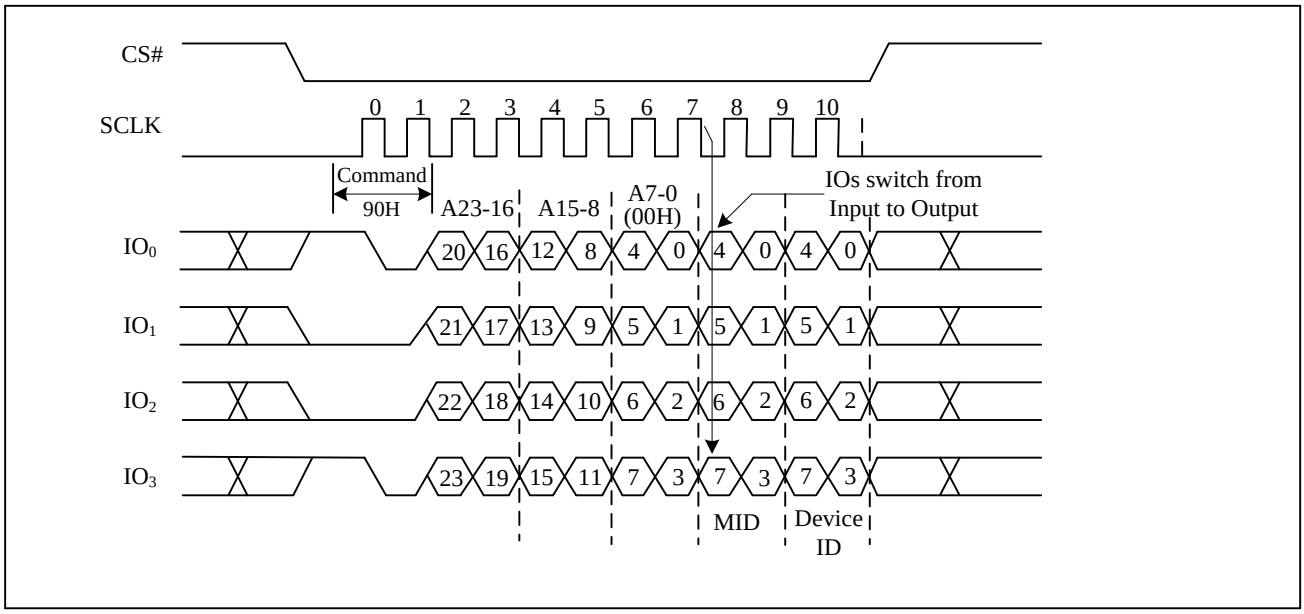



**Figure23a. Read Manufacture ID/ Device ID Sequence Diagram (QPI)**



## 7.22. Read Identification (RDID) (9FH)

The Read Identification (RDID) command allows the 8-bit manufacturer identification to be read, followed by two bytes of device identification. The device identification indicates the memory type in the first byte, and the memory capacity of the device in the second byte. The Read Identification (RDID) command while an Erase or Program cycle is in progress, is not decoded, and has no effect on the cycle that is in progress. The Read Identification (RDID) command should not be issued while the device is in Deep Power-Down Mode.

The device is first selected by driving CS# low. Then, the 8-bit command code for the command is shifted in. This is followed by the 24-bit device identification, stored in the memory. Each bit is shifted out on the falling edge of Serial Clock. The command sequence is shown in Figure24. The Read Identification (RDID) command is terminated by driving CS# high at any time during data output. When CS# is driven high, the device is in the Standby Mode. Once in the Standby Mode, the device waits to be selected, so that it can receive, decode and execute commands.

**Figure24. Read Identification ID Sequence Diagram**

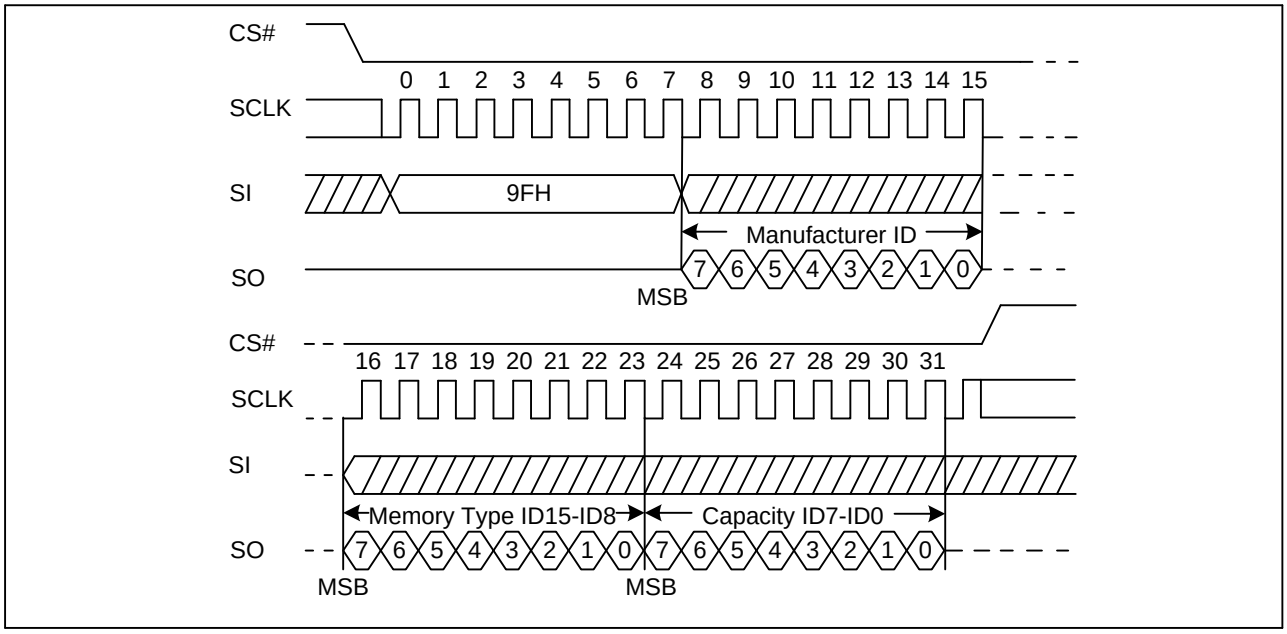
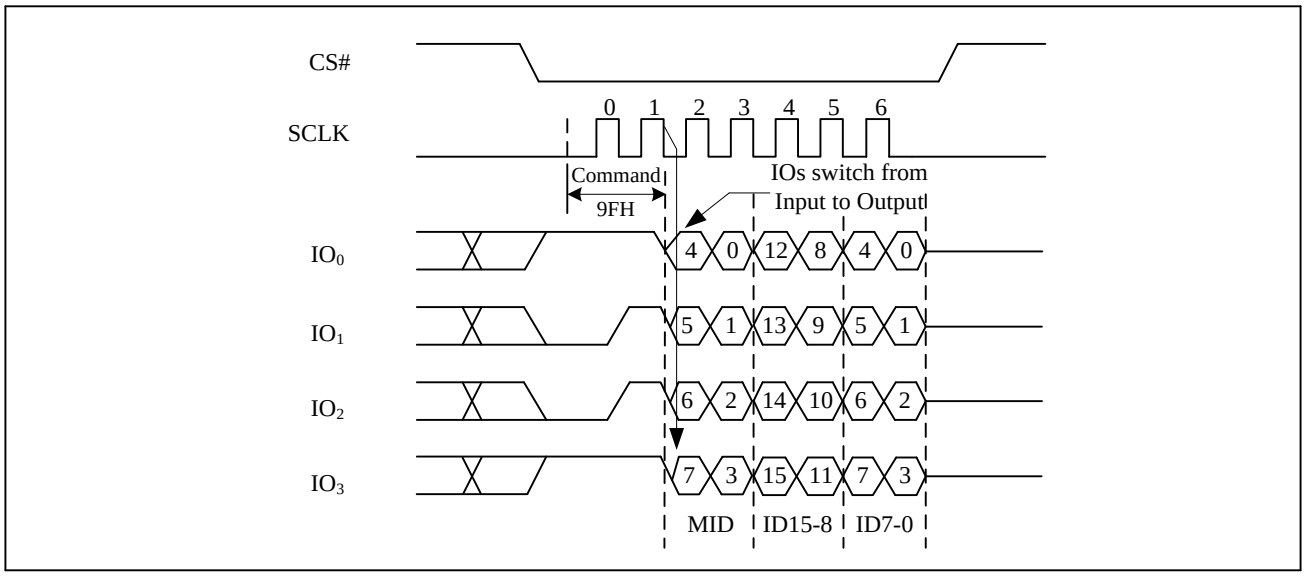



**Figure24a. Read Identification ID Sequence Diagram (QPI)**



## 7.23. Program/Erase Suspend (PES) (75H)

The Program/Erase Suspend command "75H", allows the system to interrupt a page program or sector/block erase operation and then read data from any other sector or block. The Write Status Register command (01H) and Erase Security Registers (44H, 42H) and Erase commands (20H, 52H, D8H, C7H, 60H) and Page Program command are not allowed during Program/Erase suspend. Program/Erase Suspend is valid only during the page program or sector/block erase operation. A maximum of time of "tsus" (See AC Characteristics) is required to suspend the program/erase operation.

The Program/Erase Suspend command will be accepted by the device only if the SUS2/SUS1 bit in the Status Register equal to 0 and WIP bit equal to 1 while a Page Program or a Sector or Block Erase operation is on-going. If the SUS2/SUS1 bit equal to 1 or WIP bit equal to 0, the Suspend command will be ignored by the device. The WIP bit will be cleared from 1 to 0 within "tsus" and the SUS2/SUS1 bit will be set from 0 to 1 immediately after Program/Erase Suspend. A power-off during the suspend period will reset the device and release the suspend state. The command sequence is show in Figure25.

**Figure25. Program/Erase Suspend Sequence Diagram**

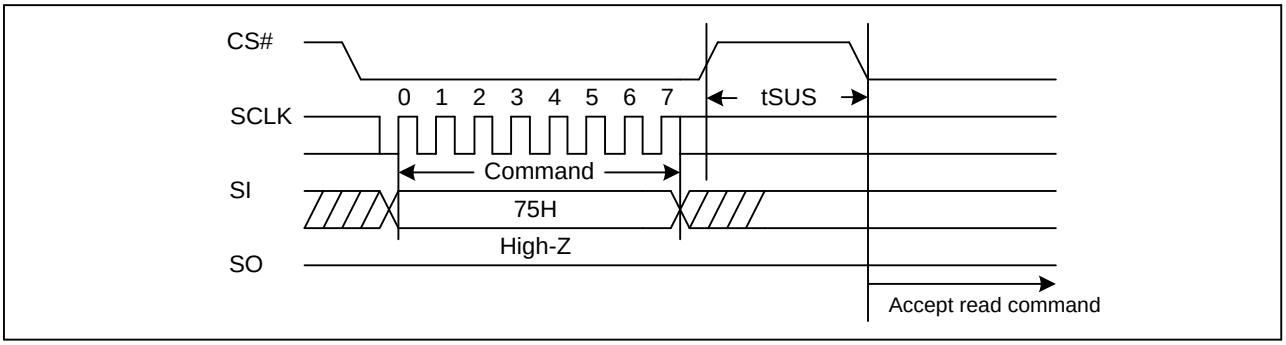
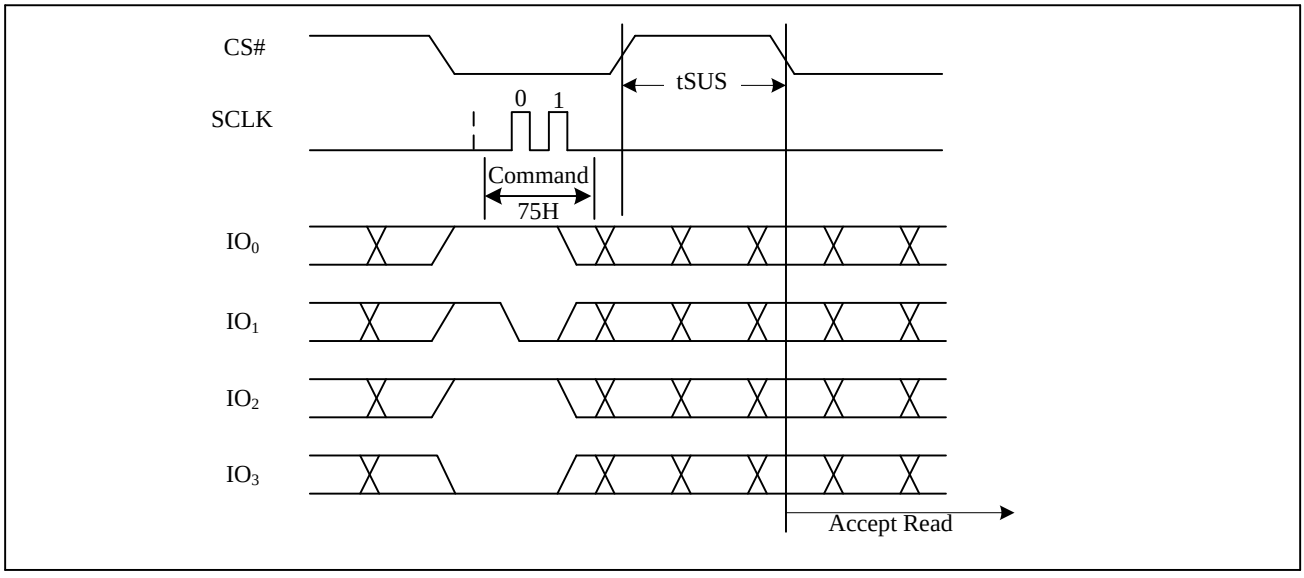



**Figure25a. Program/Erase Suspend Sequence Diagram (QPI)**



## 7.24. Program/Erase Resume (PER) (7AH)

The Program/Erase Resume command must be written to resume the program or sector/block erase operation after a Program/Erase Suspend command. The Program/Erase Resume command will be accepted by the device only if the SUS2/SUS1 bit equal to 1 and the WIP bit equal to 0. After issued the SUS2/SUS1 bit in the status register will be cleared from 1 to 0 immediately, the WIP bit will be set from 0 to 1 within 200ns and the Sector or Block will complete the erase operation or the page will complete the program operation. The Program/Erase Resume command will be ignored unless a Program/Erase Suspend is active. The command sequence is show in Figure26.

**Figure26. Program/Erase Resume Sequence Diagram**

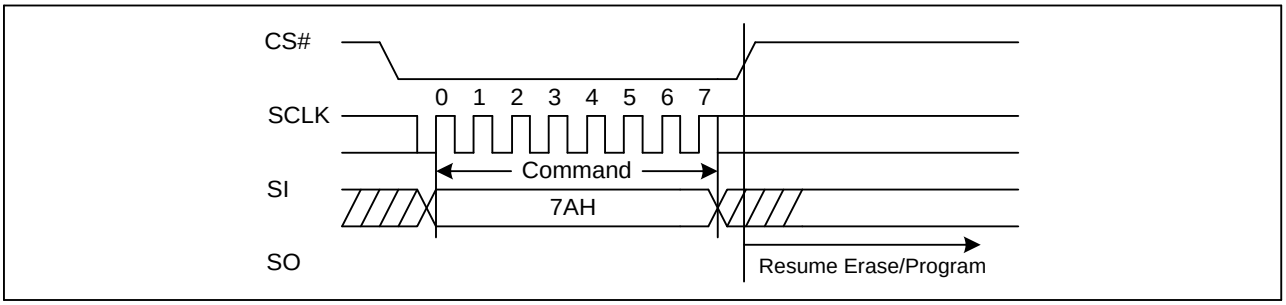
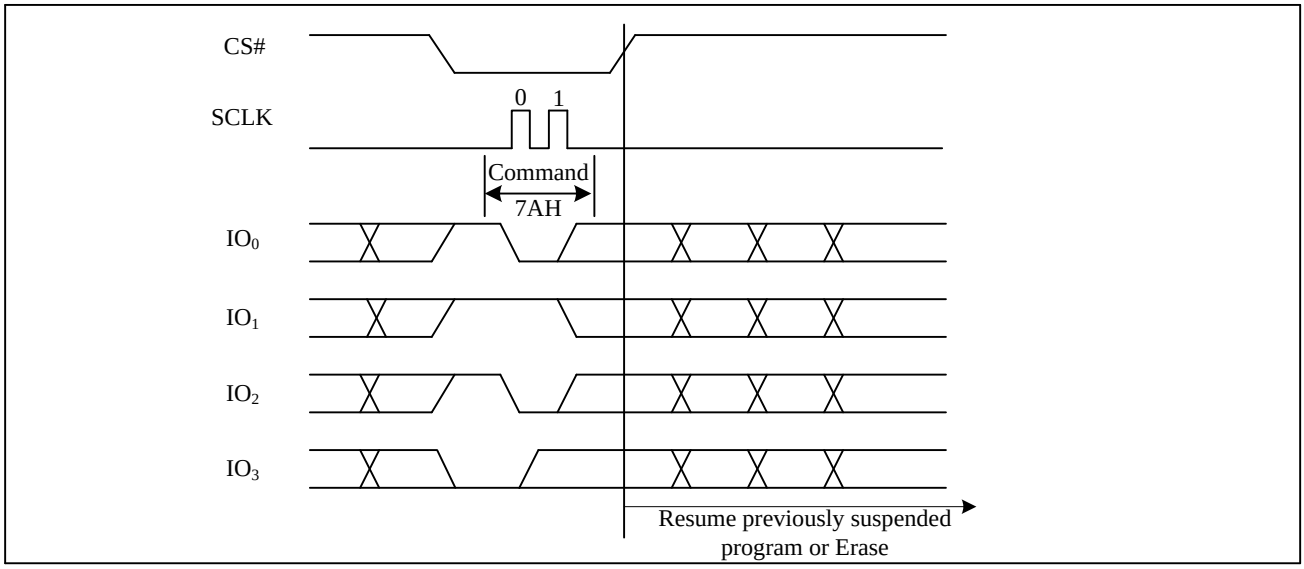



**Figure26a. Program/Erase Resume Sequence Diagram (QPI)**

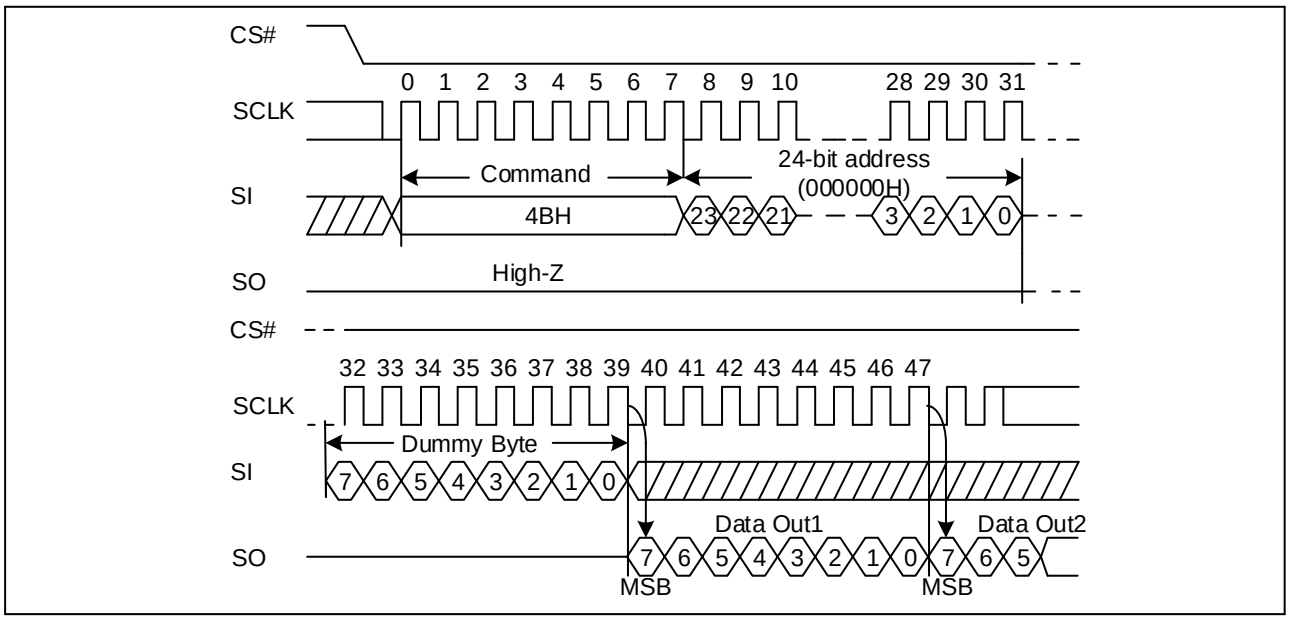


## 7.25. Read Unique ID (4BH)

The Read Unique ID command accesses a factory-set read-only 128bit number that is unique to each device. The Unique ID can be used in conjunction with user software methods to help prevent copying or cloning of a system.

The Read Unique ID command sequence: CS# goes low → sending Read Unique ID command → 3-Byte Address (000000H) →Dummy Byte→128bit Unique ID Out →CS# goes high.

**Figure27. Read Unique ID Sequence Diagram**




## 7.26. Erase Security Registers (44H)

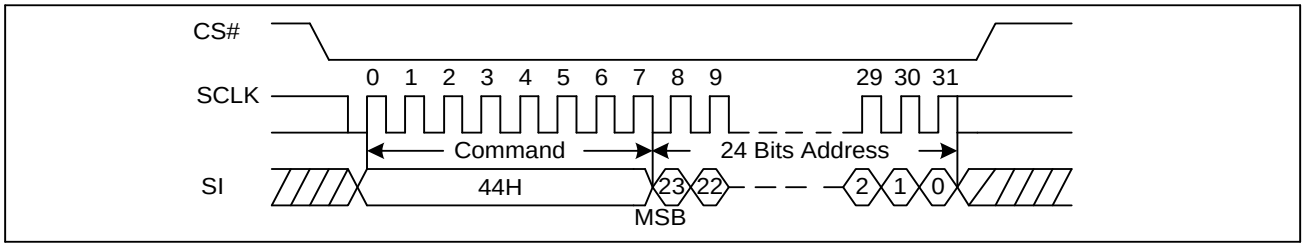
The GD25LR128D provides three 1024-byte Security Registers which can be erased and programmed individually. These registers may be used by the system manufacturers to store security and other important information separately from the main memory array.

The Erase Security Registers command is similar to Sector/Block Erase command. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit.

The Erase Security Registers command sequence: CS# goes low → sending Erase Security Registers command → 3-byte address on SI →CS# goes high. The command sequence is shown in Figure28. CS# must be driven high after the eighth bit of the last address byte has been latched in; otherwise the Erase Security Registers command is not executed. As soon as CS# is driven high, the self-timed Erase Security Registers cycle (whose duration is $t_{SE}$) is initiated. While the Erase Security Registers cycle is in progress, the Status Register may be read to check the value of the Write in Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Erase Security Registers cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. The Security Registers Lock Bit (LB3-1) in the Status Register can be used to OTP protect the security registers. Once the LB bit is set to 1, the Security Registers will be permanently locked; the Erase Security Registers command will be ignored.

| Address | A23-16 | A15-12 | A11-10 | A9-0 |
|---|---|---|---|---|
| Security Register #1 | 00H | 0 0 0 1 | 0 0 | Don't care |
| Security Register #2 | 00H | 0 0 1 0 | 0 0 | Don't care |
| Security Register #3 | 00H | 0 0 1 1 | 0 0 | Don't care |

**Figure28. Erase Security Registers command Sequence Diagram**



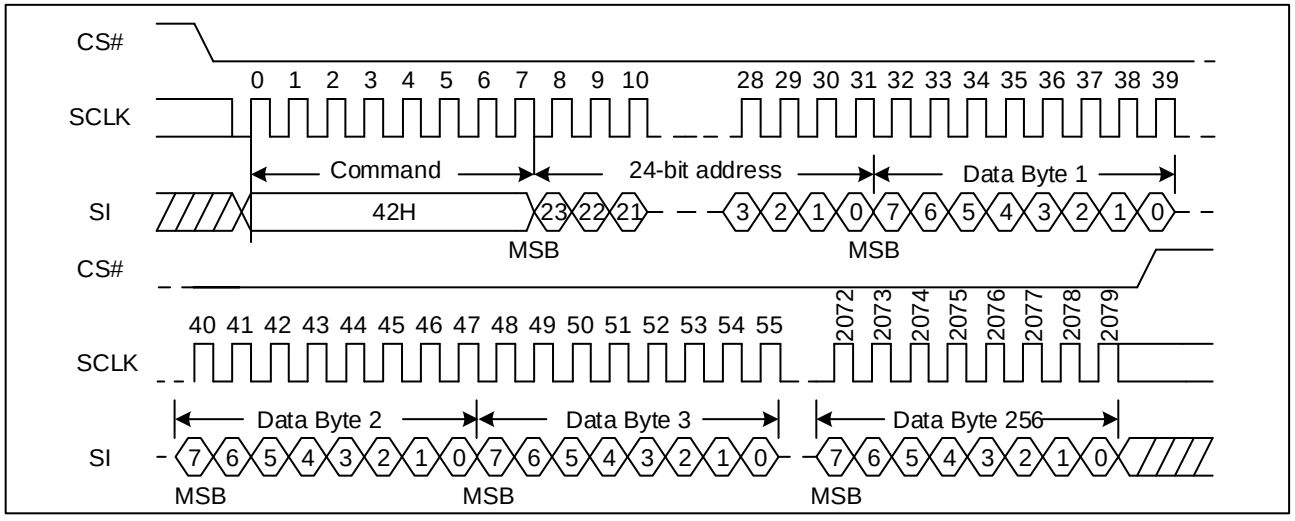

## 7.27. Program Security Registers (42H)

The Program Security Registers command is similar to the Page Program command. Each security register contains four pages content. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit before sending the Program Security Registers command. The Program Security Registers command is entered by driving CS# Low, followed by the command code (42H), three address bytes and at least one data byte on SI. As soon as CS# is driven high, the self-timed Program Security Registers cycle (whose duration is $t_{PP}$) is initiated. While the Program Security Registers cycle is in progress, the Status Register may be read to check the value of the Write in Progress (WIP) bit. The Write in Progress (WIP) bit is 1 during the self-timed Program Security Registers cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

If the Security Registers Lock Bit (LB3-1) is set to 1, the Security Registers will be permanently locked. Program Security Registers command will be ignored.

| Address | A23-16 | A15-12 | A11-10 | A9-0 |
|---|---|---|---|---|
| Security Register #1 | 00H | 0 0 0 1 | 0 0 | Byte Address |
| Security Register #2 | 00H | 0 0 1 0 | 0 0 | Byte Address |
| Security Register #3 | 00H | 0 0 1 1 | 0 0 | Byte Address |

**Figure29. Program Security Registers command Sequence Diagram**

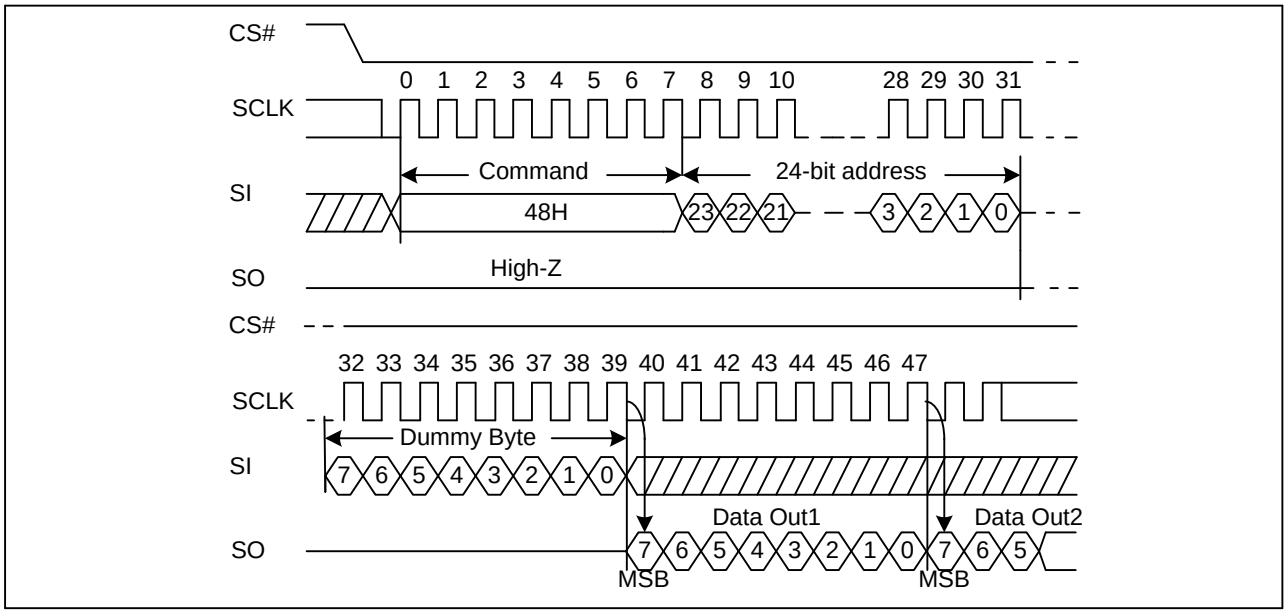


## 7.28. Read Security Registers (48H)

The Read Security Registers command is similar to Fast Read command. The command is followed by a 3-byte address (A23-A0) and a dummy byte, and each bit is latched-in on the rising edge of SCLK. Then the memory content, at that address, is shifted out on SO, and each bit is shifted out, at a Max frequency $f_C$, on the falling edge of SCLK. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. Once the A9-A0 address reaches the last byte of the register (Byte 3FFH), it will reset to 000H, the command is completed by driving CS# high.

| Address | A23-16 | A15-12 | A11-10 | A9-0 |
|---|---|---|---|---|
| Security Register #1 | 00H | 0 0 0 1 | 0 0 | Byte Address |
| Security Register #2 | 00H | 0 0 1 0 | 0 0 | Byte Address |
| Security Register #3 | 00H | 0 0 1 1 | 0 0 | Byte Address |

**Figure30. Read Security Registers command Sequence Diagram**

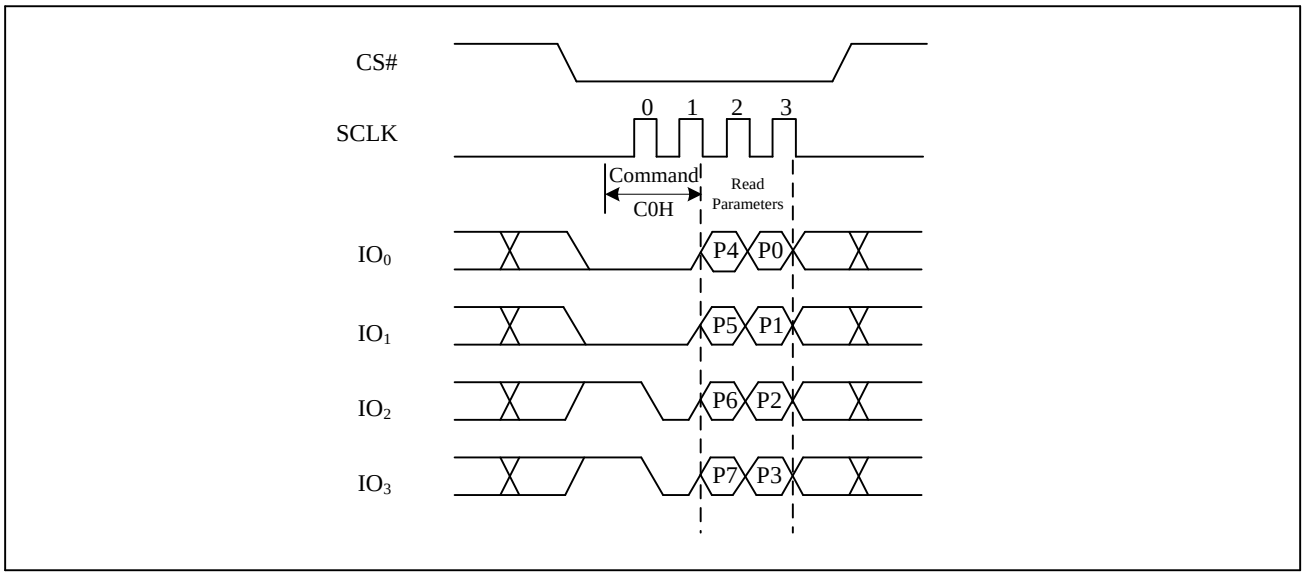



## 7.29. Set Read Parameters (C0H)

In QPI mode the “Set Read Parameters (C0H)” command can be used to configure the number of dummy clocks for “Fast Read (0BH)”, “Quad I/O Fast Read (EBH)” and “Burst Read with Wrap (0CH)” command, and to configure the number of bytes of “Wrap Length” for the “Burst Read with Wrap (0CH)” command. The “Wrap Length” is set by W5-6 bit in the “Set Burst with Wrap (77H)” command. This setting will remain unchanged when the device is switched from Standard SPI mode to QPI mode.

| P5-P4 | Dummy Clocks | Maximum Read Freq. | P1-P0 | Wrap Length |
|---|---|---|---|---|
| 0 0 | 4 | 80MHz | 0 0 | 8-byte |
| 0 1 | 6 | 108MHz | 0 1 | 16-byte |
| 1 0 | 8 | 120MHz | 1 0 | 32-byte |
| 1 1 | 8 | 120MHz | 1 1 | 64-byte |

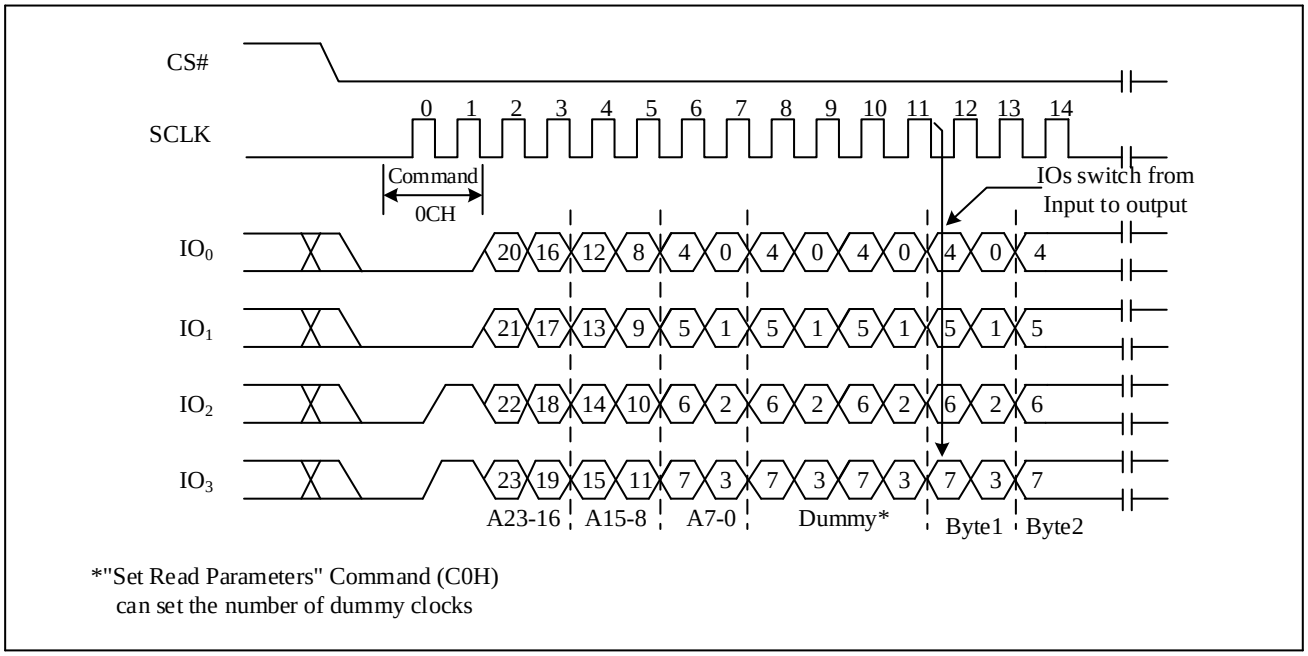
**Figure31. Set Read Parameters command Sequence Diagram**




## 7.30. Burst Read with Wrap (0CH)

The “Burst Read with Wrap (0CH)” command provides an alternative way to perform the read operation with “Wrap Around” in QPI mode. This command is similar to the “Fast Read (0BH)” command in QPI mode, except the addressing of the read operation will “Wrap Around” to the beginning boundary of the “Wrap Around” once the ending boundary is reached. The “Wrap Length” and the number of dummy clocks can be configured by the “Set Read Parameters (C0H)” command.

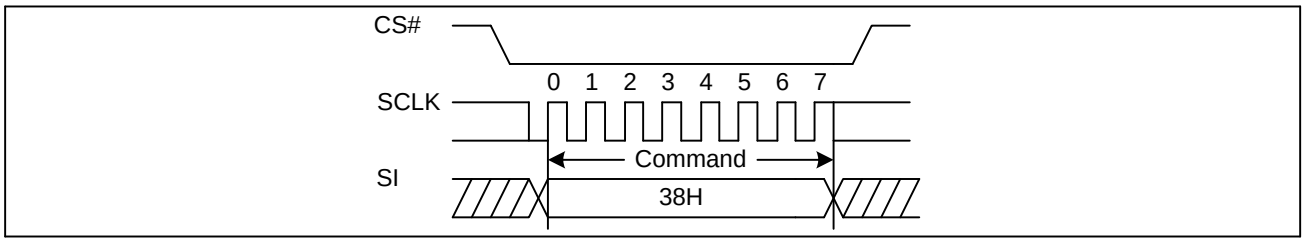
**Figure32. Burst Read with Wrap command Sequence Diagram**



## 7.31. Enable QPI (38H)

The device support both Standard/Dual/Quad SPI and QPI mode. The “Enable QPI (38H)” command can switch the device from SPI mode to QPI mode. See the command Table 2a for all support QPI commands. In order to switch the device to QPI mode, the “Enable QPI (38H)” command must be issued. When the device is switched from SPI mode to QPI mode, the existing Write Enable Latch and Program/Erase Suspend status, and the Wrap Length setting will remain unchanged.

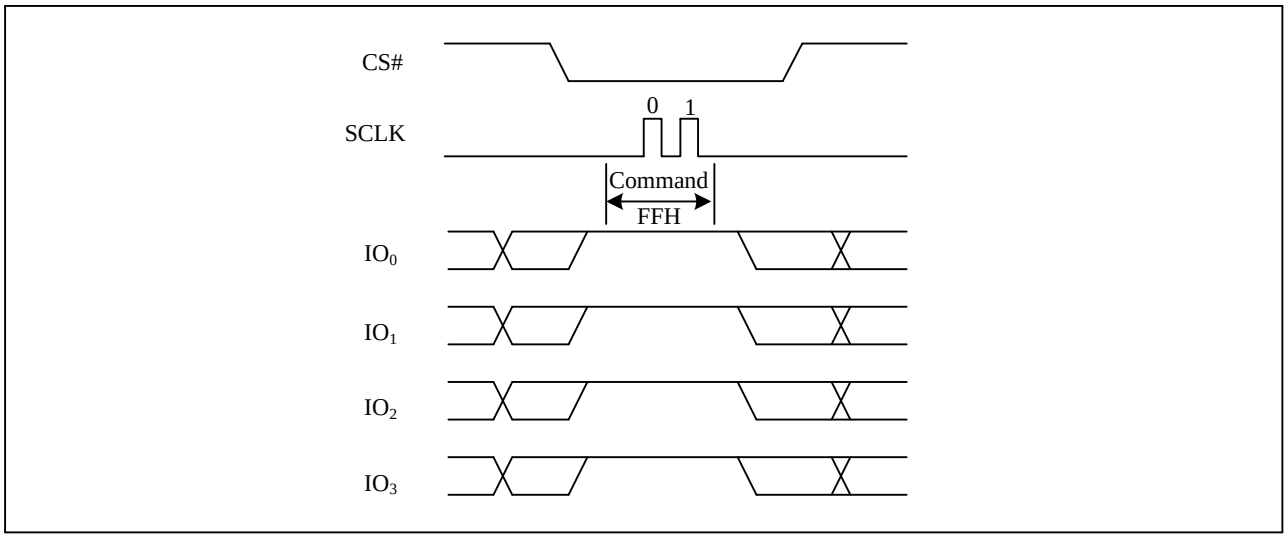
**Figure33. Enable QPI mode command Sequence Diagram**




## 7.32. Disable QPI (FFH)

To exit the QPI mode and return to Standard/Dual/Quad SPI mode, the “Disable QPI (FFH)” command must be issued. When the device is switched from QPI mode to SPI mode, the existing Write Enable Latch and Program/Erase Suspend status, and the Wrap Length setting will remain unchanged.

**Figure34. Disable QPI mode command Sequence Diagram**




## 7.33. Enable Reset (66H) and Reset (99H)

If the Reset command is accepted, any on-going internal operation will be terminated and the device will return to its default power-on state and lose all the current volatile settings, such as Volatile Status Register bits, Write Enable Latch status (WEL), Program/Erase Suspend status, Read Parameter setting (P7-P0) and Wrap Bit Setting (W6-W4).

The “Enable Reset (66H)” and the “Reset (99H)” commands can be issued in either SPI or QPI mode. The “Reset (99H)” command sequence as follow: CS# goes low → Sending Enable Reset command → CS# goes high → CS# goes low → Sending Reset command → CS# goes high. Once the Reset command is accepted by the device, the device will take approximately $t_{RST}/t_{RST\_E}$ to reset. During this period, no command will be accepted. Data corruption may happen if there is an on-going or suspended internal Erase or Program operation when Reset command sequence is accepted by the

device. It is recommended to check the WIP bit and the SUS bit in Status Register before issuing the Reset command sequence.

**Figure35. Enable Reset and Reset command Sequence Diagram**

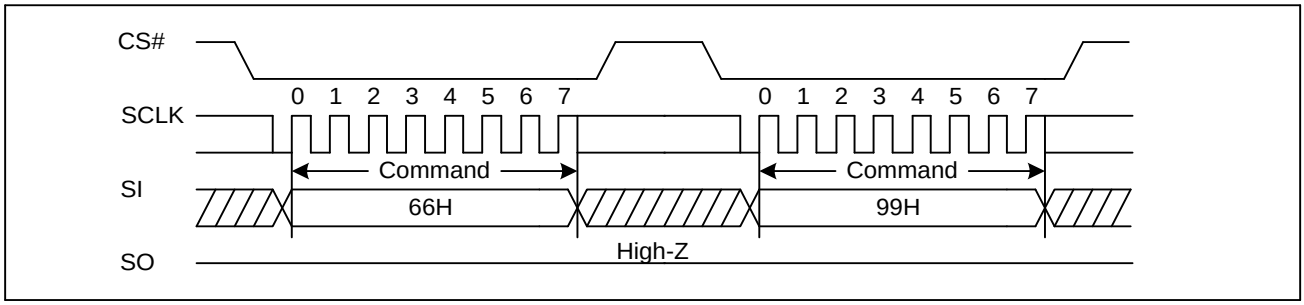
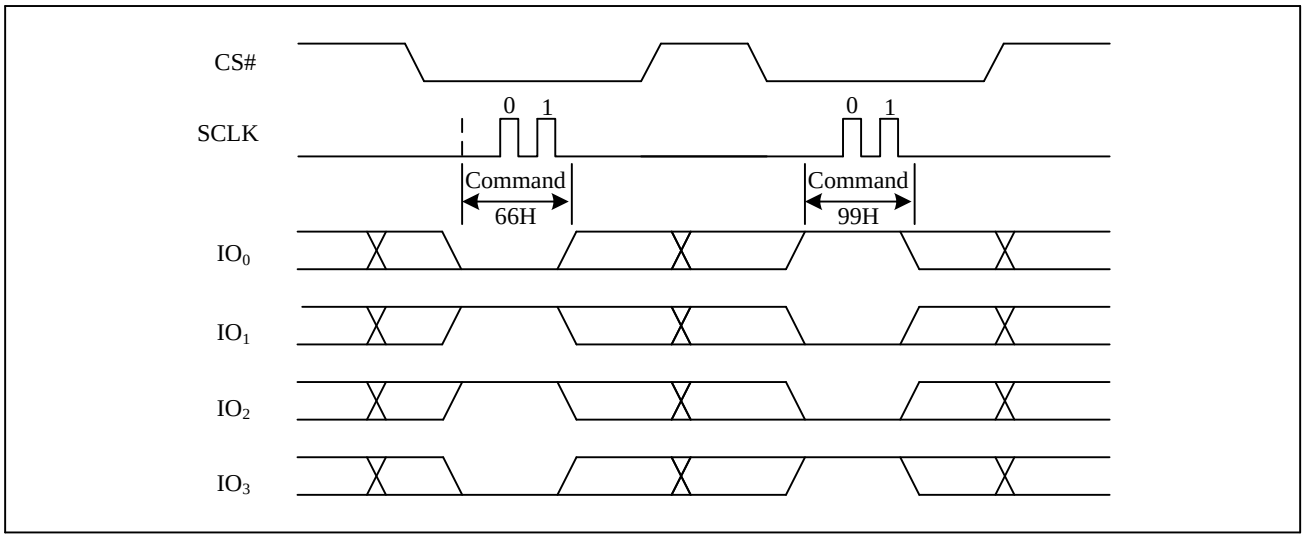



**Figure35a. Enable Reset and Reset command Sequence Diagram (QPI)**




## 7.34. Read Serial Flash Discoverable Parameter (5AH)

The Serial Flash Discoverable Parameter (SFDP) standard provides a consistent method of describing the functional and feature capabilities of serial flash devices in a standard set of internal parameter tables. These parameter tables can be interrogated by host system software to enable adjustments needed to accommodate divergent features from multiple vendors. The concept is similar to the one found in the Introduction of JEDEC Standard, JESD68 on CFI. SFDP is a standard of JEDEC Standard No.216B.

**Figure36. Read Serial Flash Discoverable Parameter command Sequence Diagram**

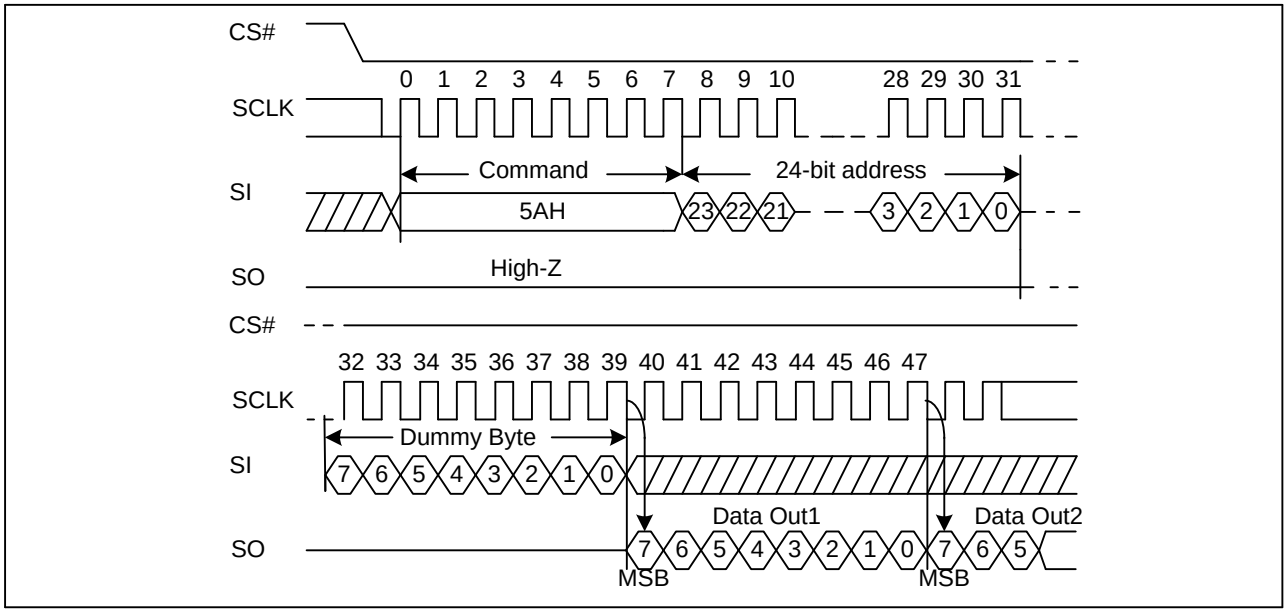
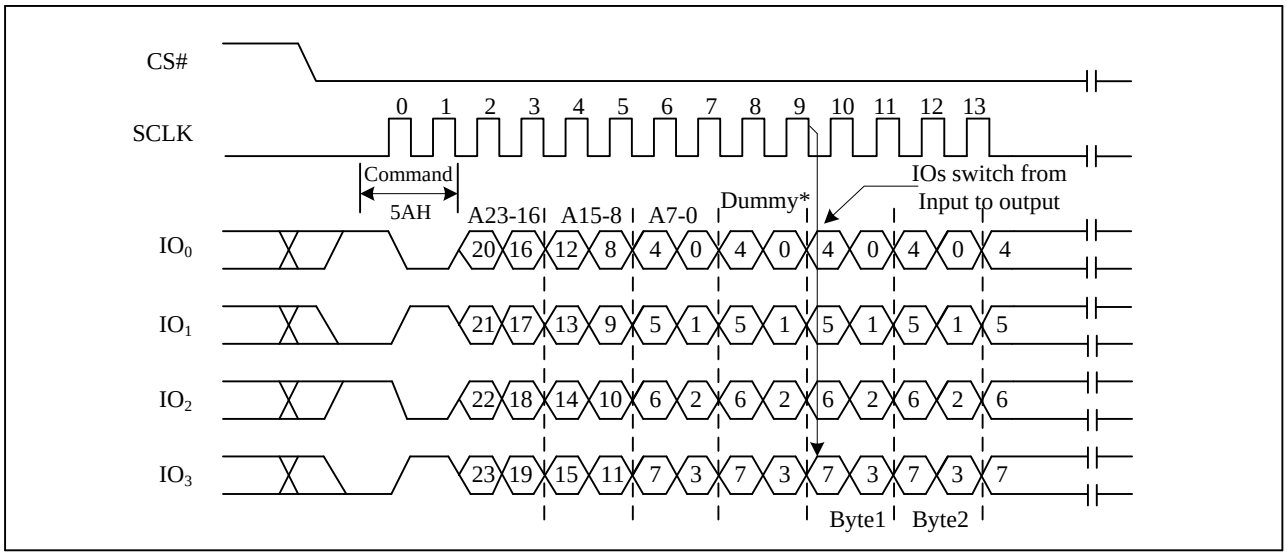



**Figure36a. Read Serial Flash Discoverable Parameter command Sequence Diagram (QPI)**




**For Signature and Parameter Identification Data Values, Please contact GigaDevice for Details.**

# 8. RPMC COMMANDS DESCRIPTION

**Table 7. Replay Protected Monotonic Counter (RPMC) Commands**

| Function | Opcode Phase 8 bits | Payload Phase Max 512 Bits | | Comment |
|---|---|---|---|---|
| | | Byte# | Field Description | |
| Command: Write Root Key Register | OP1 | 1<br>2<br>3<br>4-35<br>36-63 | CmdType[7:0] = 00H<br>CounterAddr[7:0] = 8 Bits<br>Reserved[7:0] = 8 Bits<br>RootKey[255:0] = 256 Bits<br>TruncatedSign[223:0] = 224 Bits | OP1 + Payload phase driven by host controller. Root Key Register is written only once. |
| Command: Update HMAC Key Register | OP1 | 1<br>2<br>3<br>4-7<br>8-39 | CmdType[7:0]= 01H<br>CounterAddr[7:0]= 8 Bits<br>Reserved[7:0] = 8 Bits<br>KeyData[31:0] = 32 Bits,<br>Signature[255:0] = 256 Bits | OP1 + Payload phase is Issued by host controller on every power up to initialize HMAC Key Register. |
| Command: Increment Monotonic Counter | OP1 | 1<br>2<br>3<br>4-7<br>8-39 | CmdType[7:0] = 02H<br>CounterAddr[7:0] = 8 Bits<br>Reserved[7:0] = 8 Bits<br>CounterData[31:0] = 32 Bits,<br>Signature[255:0] = 256 Bits | OP1 + Payload Phase is Issued by host controller during runtime to increment the counter. |
| Command: Request Monotonic Counter | OP1 | 1<br>2<br>3<br>4-15<br>16-47 | CmdType[7:0] = 03H<br>CounterAddr[7:0] = 8 Bits<br>Reserved[7:0] = 8 Bits Tag<br>[95:0] = 96 Bits<br>Signature[255:0] = 256 Bits | OP1 + Payload Phase is Issued by host controller during runtime to request counter data |
| Command: Read Data | OP2 | 2<br>3-14<br>15-18<br>19-50 | ExtendedStatus[7:0] = 8 Bits<br>Tag[95:0] = 96 Bits<br>CounterData[31:0] = 32 Bits<br>Signature[255:0] = 256 Bits | OP2 is issued by Host Controller generally after an OP1. SPI Flash device responds with the Payload phase to return Extended Status and counter data. |

All individual fields are Byte wide fields. For a multi-byte field, Most Significant Byte is issued first; Least Significant Byte is issued last. Within a Byte, Most Significant Bit is issued first; Least Significant Bit is issued last. Cmd Type is always the first byte issued after OP1 commands. OP2 delay is the same as Fast Read Command delay which is 8 dummy bits.

**Table 8. OP1 and OP2 are defined for 1-1-1 mode.**

| Byte # | 0 | 1 | 2 | 3 | 4 | 5 | 6 | … | .. | .. |
|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | OP1 | CmdType | Counter Address | As defined in the table above | | | | | | |
| **Name** | OP2 | 8 Dummy clocks | Extended Status[7:0] | As defined in the table above | | | | | | |

After an OP1 command is received, the SPI Flash will indicate status busy indication using either the status register or extended status register as defined below.

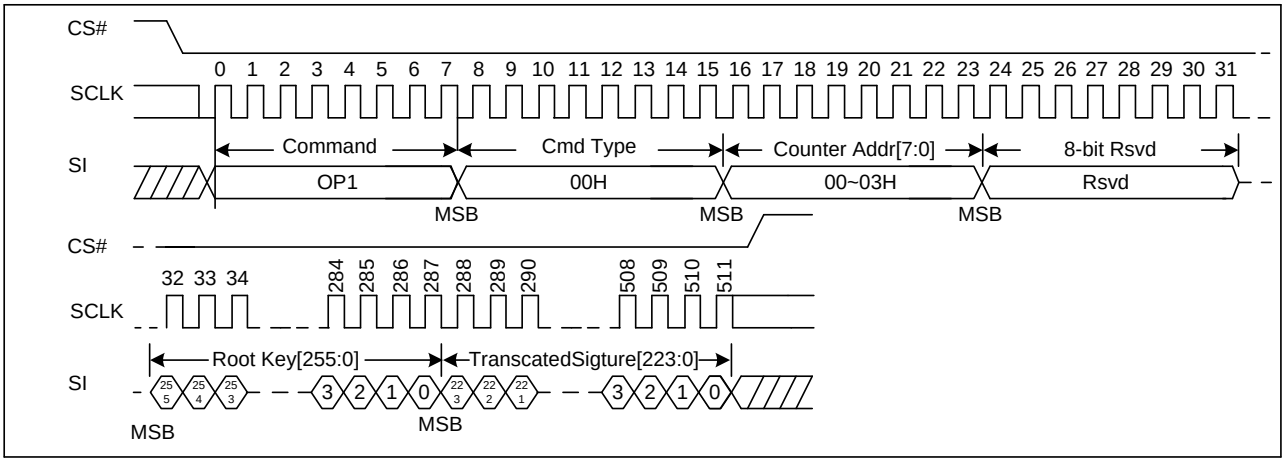
**Table 9. Extended Status Register Definition**

| Extended Status[7:0] | Applicable CmdType(s) | Description |
|---|---|---|
| 00000000 | - | Power On State (OP2 issued directly after power-up). |
| 10000000 | 00,01,02,03, | This status is set on successful completion (no errors) of OP1 command. |
| 0xxxxxx1 | 00,01,02,03, 04-0FF | This bit is set to 1, when device is busy executing OP1 command. It is reset to 0 when OP1 command execution is done. |
| 0xxxxx1x | 00,01 | This bit is set only when the correct payload size is received. When cmdtype = 0, this error bit must be set on Root Key Register Overwrite or Counter Address out of range or Truncated Signature mis-match error.<br>For cmdtype = 01 this bit is set when the corresponding montonic counter is uninitialized |
| 0xxxx1xx | 00,01,02,03 | This bit is set on Signature Mismatch, Counter Address out of range when correct payload size is received; or Cmdtype is out of range; or incorrect payload size is received. |
| 0xxx1xxx | 02,03 | This bit is set on HMAC Key Register or monotonic counter uninitialized on previous OP1 command when correct payload size is received |
| 0xx1xxxx | 02 | This bit is set on Counter Data Mismatch on previous increment when correct payload size is received |
| 0x1xxxxx | - | Fatal Error. It is set when no valid counter is found after initialization. |
| Current value | | Extended status register will not be updated until first 8 bits of OP1 is received. The correct error type is reflected for any OP1 operation that exceeds a minimum of 16 clocks with active chip-select. |

## 8.1. Command: Write Root Key Register

This command is used to initialize the Root Key Register corresponding to the received Counter Address with the received Root Key. It is suggested to be used in an OEM manufacturing environment when the SPI Flash Controller and SPI Flash are powered together for the first time.

**Figure 37. Write Root Key Register Sequence Diagram**




Truncated signature field is the same as least significant 224 bits of HMAC-SHA-256 based signature computed based on received input parameters:

- HMAC message[31:0] = (OpCode[7:0], CmdType[7:0], CounterAddr[7:0], Reserved[7:0])
- HMAC Key[255:0] = Root_Key[255:0]

If Root Key != 256'HFF..FF then this command can be executed one time. If the received transaction is error free SPI Flash device successfully executes the command and posts "successful completion" extended status.

Root Key Register Write with root key is = 256'HFF…FF can be used as a temporary key.

If the received transaction has errors the SPI Flash does not execute the transaction and posts the corresponding error in extended status.

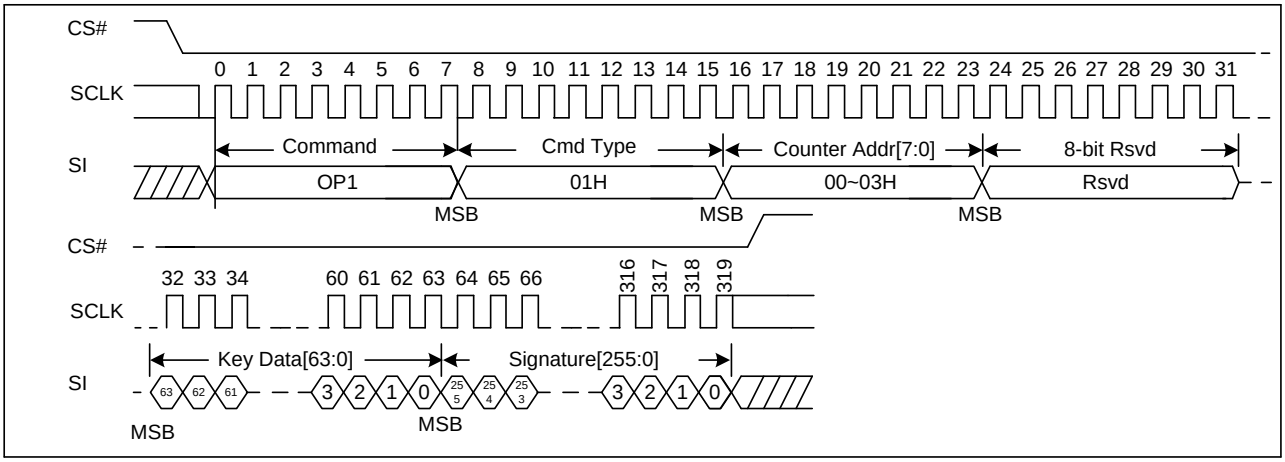
**Table 10. Expected Extended Status [7:0] results**

| Extended Status[7:0] | Applicable CmdType(s) | Description |
|---|---|---|
| 10000000 | 00 | Successful completion |
| 0xxxxxx1 | 00 | If Busy_Polling_Method bit in SFDP table is zero, then this bit must be set to 1, when device is busy executing command. It is reset to 0 when OP1 command execution is done. If Busy_Polling_Method bit in SFDP table is one, then this bit is ignored by the controller. |
| 0xxxxx1x | 00 | This bit is only set when correct payload size is received. It is set on Root Key Register Overwrite or Counter Address is out of range or when there is a truncated signature mismatch error |
| 0xxxx1xx | 00 | This bit is set when incorrect payload size is received. |

## 8.2. Command: Update HMAC Key Register

This command is used by the SPI Flash Controller to update the HMAC-Key register corresponding to the received Counter Address with a new HMAC key calculated based on received input. This command must be issued on every power cycle event on the interface. The HMAC key storage is volatile.

**Figure 38. Update HMAC Key Register Sequence Diagram**




Signature matches the HMAC-SHA-256 based signature computed based on received input parameters. This command performs two HMAC-SHA-256 operations.

- HMAC-SHA-256 Operation 1 Output = HMAC_Storage[255:0]
  - HMAC Message[31:0] = KeyData[31:0]
  - HMAC Key[255:0] = Root_Key_Register[CounterAddr][255:0]
- HMAC-SHA-256 Operation 2 Output = HMAC-SHA-256 based signature[255:0]
  - HMAC message[63:0] = (OpCode[7:0], CmdType[7:0].CounterAddr[7:0].Reserved[7:0], KeyData[31:0])
  - HMAC Key[255:0] = HMAC_Storage[255:0]

If the received transaction is error free SPI Flash device successfully executes the command and posts “successful completion” extended status.

If the received transaction has errors the SPI Flash does not execute the transaction and posts the corresponding error in extended status.

**Table 11. Expected Extended Status [7:0] results**

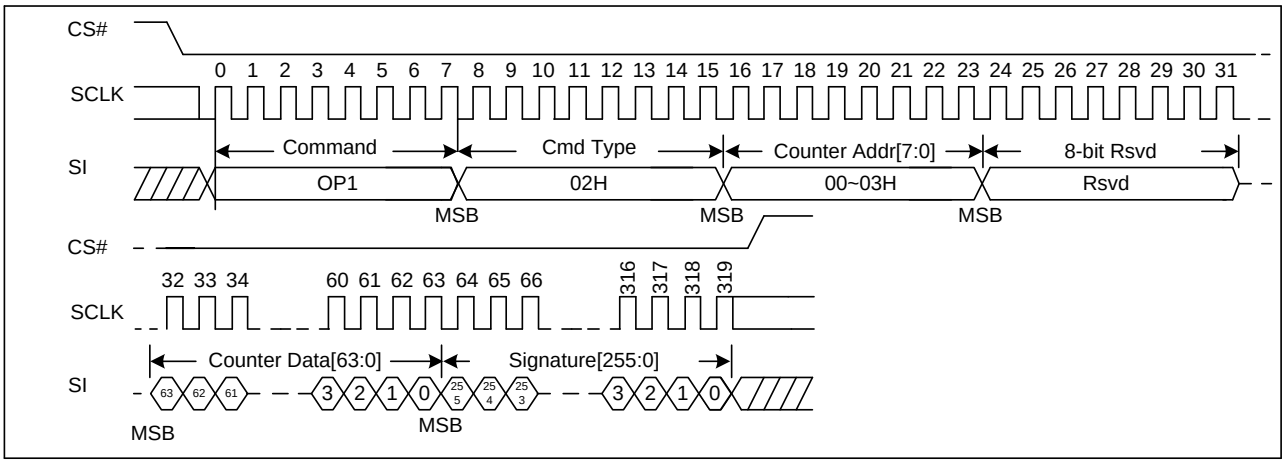
| Extended Status[7:0] | Applicable CmdType(s) | Description |
|---|---|---|
| 10000000 | 01 | This status is set on successful completion (no errors) of OP1 command. |
| 0xxxxxx1 | 01 | This bit is set to 1, when device is busy executing OP1 command. It is reset to 0 when OP1 command execution is done. |
| 0xxxxx1x | 01 | This bit is set only when the correct payload size is received. This bit is set when the corresponding monotonic counter is uninitialized |

| | | |
|---|---|---|
| 0xxxx1xx | 01 | This bit is set on Signature Mismatch, Counter Address out of range when correct payload size is received; or incorrect payload size is received. |

## 8.3. Command: Increment Monotonic Counter

This command is used by the SPI Flash Controller to increment the Monotonic counter by 1 inside the SPI Flash Device.

**Figure 39. Increment Monotonic Counter Sequence Diagram**




The requested Signature matches the HMAC-SHA-256 based signature computed based on received input parameters. The received Counter Data matches the current value of the counter read from the SPI Flash.

- HMAC Message[63:0] = (OpCode[7:0], CmdType[7:0]. CounterAddr[7:0]. Reserved[7:0], CounterData[31:0])
- HMAC Key[255:0] = HMAC_Key_Register [Counter_Addreess][255:0]

If the received transaction is error free SPI Flash device successfully executes the command and posts “successful completion” extended status. If the received transaction has errors the SPI Flash does not execute the transaction and posts the corresponding error in extended status.

**Table 12. Expected Extended Status [7:0] results**

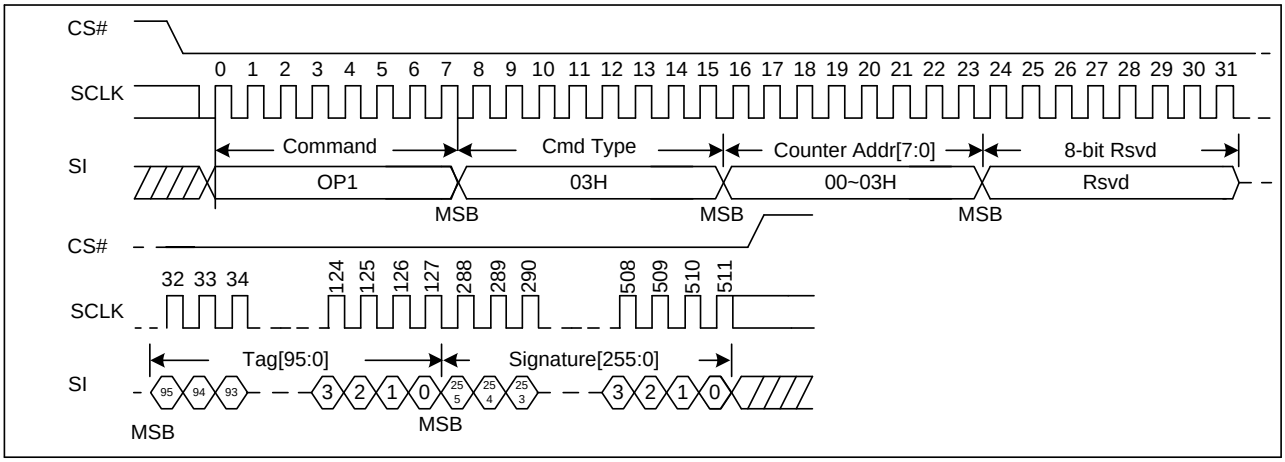
| Extended Status[7:0] | Applicable CmdType(s) | Description |
|---|---|---|
| 10000000 | 02 | This status is set on successful completion (no errors) of OP1 command. |
| 0xxxxxx1 | 02 | This bit is set to 1, when device is busy executing OP1 command. It is reset to 0 when OP1 command execution is done. |
| 0xxxx1xx | 02 | This bit is set on Signature Mismatch, Counter Address out of range when correct payload size is received; or incorrect payload size is received. |
| 0xxx1xxx | 02 | This bit is set only when the correct payload size is received. This bit must be set on HMAC Key Register or Monotonic Counter is uninitialized on previous OP1 command. |

| | | |
|---|---|---|
| 0xx1xxxx | 02 | This bit is set only when the correct payload size is received. The bit must be set when the received counter data filed does not match the actual counter value read from the SPI Flash device. |

## 8.4. Command: Request Monotonic Counter

This command is used by the SPI Flash Controller to request the Monotonic counter value inside the SPI Flash Device.

**Figure 40. Request Monotonic Counter Sequence Diagram**




The requested Signature matches the HMAC-SHA-256 based signature computed based on received input parameters.

- HMAC Message[127:0] = (OpCode[7:0], CmdType[7:0]. CounterAddr[7:0]. Reserved[7:0], Tag[95:0])
- HMAC Key[255:0] = HMAC_Key_Register[Counter_Addreess][255:0]

If the received transaction is error free SPI Flash device successfully executes the command and posts “successful completion” extended status. If the received transaction has errors the SPI Flash does not execute the transaction and posts the corresponding error in extended status.

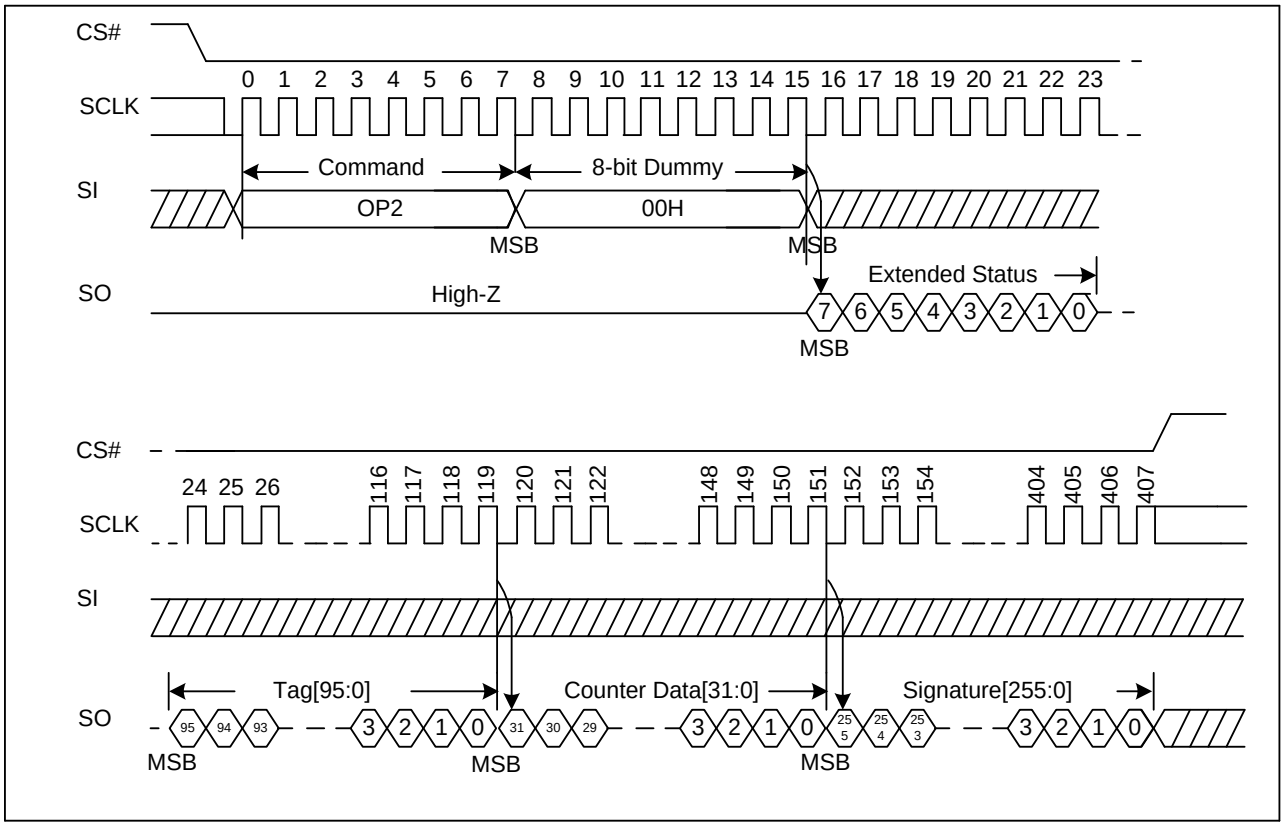
**Table 13. Expected Extended Status [7:0] results**

| Extended Status[7:0] | Applicable CmdType(s) | Description |
|---|---|---|
| 10000000 | 03 | This status is set on successful completion (no errors) of OP1 command. |
| 0xxxxxx1 | 03 | This bit is set to 1, when device is busy executing OP1 command. It is reset to 0 when OP1 command execution is done. |
| 0xxxx1xx | 03 | This bit is set on Signature Mismatch, Counter Address out of range when correct payload size is received; or Cmdtype is out of range; or incorrect payload size is received. |
| 0xxx1xxx | 03 | This bit is set only when the correct payload size is received. This bit must be set on HMAC Key Register or Monotonic Counter is uninitialized on previous OP1 command. |

## 8.5. Command: Read Data

This command is used by the SPI Flash Controller to read extended status from any previously issued OP1 command. In addition if previous OP1 command is Request Monotonic Counter and if SPI Flash returns successful completion extended status then it returns valid values in the Tag, Counter Data and Signature field. Otherwise the values returned in Tag, Counter and Signature field are invalid. The controller may abort the read prematurely prior to completely reading the entire payload. This may occur when the controller wants to simply read the extended status or when is observes an error being returned in the extended status field. The controller may also continue reading past the defined payload size of 49 bytes. Since this is an error condition, the SPI Flash may return any data past the defined payload size. The controller must ignore the data.

**Figure 41. Read Data Sequence Diagram**




If previous OP1 command is Request Monotonic Counter and if SPI Flash returns successful completion extended status then it returns valid values in the Tag, Counter Data and Signature field.It calculates HMAC-SHA-256 signatures based on following parameters.

- HMAC Message[127:0] = Tag [95:0], Counter_Data_Read[31:0]
- HMAC Key[255:0] = HMAC_Key_Register[Counter_Address][255:0]

**Table 14. Extended Status Register Definition**

| Extended Status[7:0] | Applicable CmdType(s) | Description |
|---|---|---|
| 00000000 | - | Power On State (OP2 issued directly after power-up). |
| 10000000 | 00, 01, 02, 03 | This status is set on successful completion (no errors) of OP1 command. |
| 0xxxxxx1 | 00, 01, 02, 03, | This bit is set to 1, when device is busy executing OP1 command. It is reset to 0 when OP1 command execution is done. |
| 0xxxxx1x | 00, 01 | This bit is set only when the correct payload size is received. When cmdtype = 0, this error bit must be set on Root Key Register Overwrite or Counter Address out of range or Truncated Signature mis-match error.<br>For remaining cmdtype = 1 this bit is set when the corresponding monotonic counter is uninitialized |
| 0xxxx1xx | 01, 02, 03 | This bit is set on Signature Mismatch, Counter Address out of range when correct payload size is received; or Cmdtype is out of range; or incorrect payload size is received. |
| 0xxx1xxx | 02, 03 | This bit is set on HMAC Key Register or Monotonic Counter uninitialized on previous OP1 command when correct payload size is received |
| 0xx1xxxx | 02 | This bit is set on Counter Data Mismatch on previous increment when correct payload size is received |
| 0x1xxxxx | - | Fatal Error. It is set when no valid counter is found after initialization. |
| Current value | - | Extended status register will not be updated until first 8 bits of OP1 is received. The correct error type is reflected for any OP1 operation that exceeds a minimum of 16 clocks with active chip-select. |

## 8.6. Operations Allowed/Disallowed During RPMC Operation

In the deep power down state OP1, OP2 commands are ignored until the part comes out of deep power down state.

WREN state does not affect the OP1 command execution inside the SPI Flash.

Suspend operation can be used to execute high-priority reads from the flash device while a long-latency operation is underway. However, OP1 is not recommended when the flash device is in WIP or suspended state.

In the table below, OP1 state is defined as the time starting with a transaction with OP1 op-code sent to the device and ending when the device clears the extended status busy bit. During OP1 state if a suspend transaction is received, the SPI Flash will ignore the suspend command and continue with the execution of the current OP1 command as described in the table below. P/E state is defined as the time starting with a transaction with write or erase op-code sent to the device and ending when the device clears the status busy bit. P/E Suspended State starts when the device sets the program suspend status done bit after receiving a program suspend op-code. During P/E State and P/E Suspended State, OP1 is also allowed but not recommended

The table below shows all operation support in each state.

**Table 15. RPMC Operation**

| Operation | OP1 state | P/E state | P/E Suspended State |
|---|---|---|---|
| Suspend | Ignored | Yes-> P/E Suspended State(not chip erase or write status operation) No ->remain P/E state ( chip erase or write status operation) | No |
| Resume | Ignored | No | Yes -> P/E state |
| All reads except Read status | Yes | No | Yes |
| All writes/erases | Yes | No | No |
| OP1 | No | Yes but not recommended | Yes but not recommended |
| Write status | Yes | No | No |
| OP2 | Yes->OP1 busy state (when extended status busy is 1) ->OP1 done state (when extended status busy is 0) | Yes. Will indicate the status associated with the OP1 operation. | Yes. Will indicate the status associated with the OP1 operation |
| Read status | Yes. Will indicate the busy state associated with the subsequent transaction issued to the SPI Flash. | Yes. Will indicate the busy state associated with the subsequent transaction issued to the SPI Flash. | Yes. Will indicate the busy state associated with the subsequent transaction issued to the SPI Flash. |

# 9. ELECTRICAL CHARACTERISTICS

## 9.1. POWER-ON TIMING

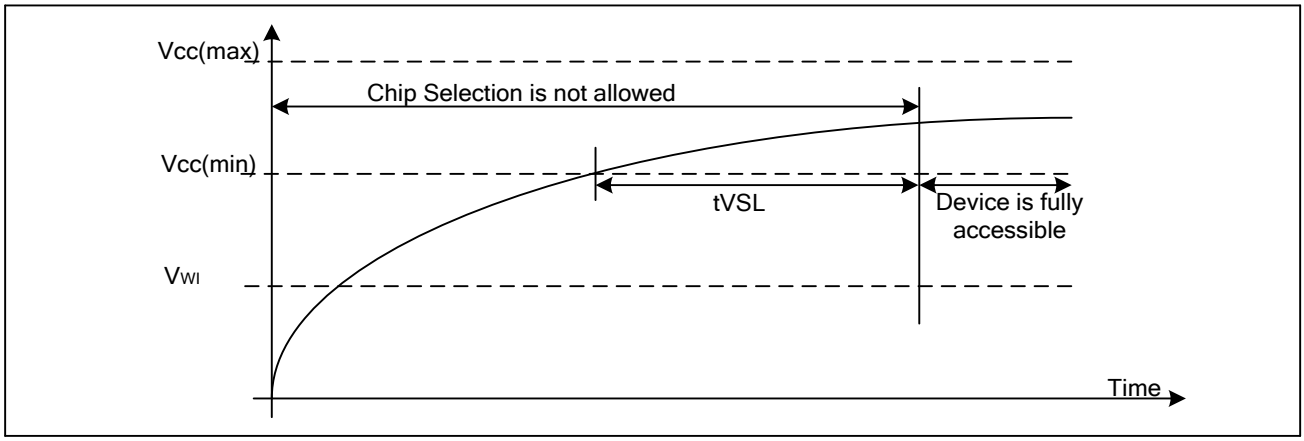



**Table16. Power-Up Timing And Write Inhibit Threshold**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| tVSL | VCC (min) To CS# Low | 2.5 | | ms |
| VWI | Write Inhibit Voltage VCC (min) | 1 | 1.5 | V |

## 9.2. INITIAL DELIVERY STATE

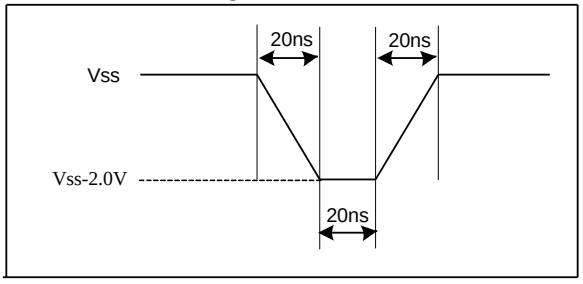
The device is delivered with the memory array erased: all bits are set to 1(each byte contains FFH).The Status Register bits are set to 0, except QE bit (S9) is set to 1.

## 9.3. ABSOLUTE MAXIMUM RATINGS

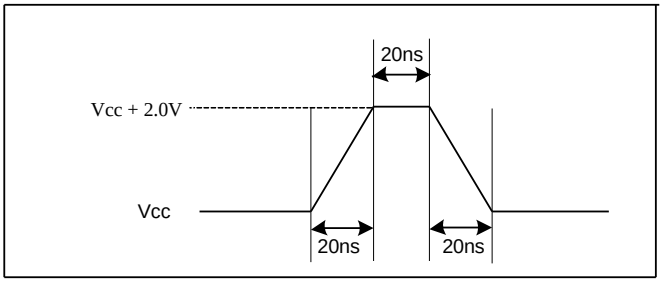
| Parameter | Value | Unit |
|---|---|---|
| Ambient Operating Temperature | -40 to 85 | ℃ |
| Storage Temperature | -65 to 150 | ℃ |
| Applied Input/Output Voltage | -0.6 to VCC+0.4 | V |
| Transient Input/Output Voltage (note: overshoot) | -2.0 to VCC+2.0 | V |
| VCC | -0.6 to 2.5 | V |

**Figure 42. Input Test Waveform and Measurement Level**

**Maximum Negative Overshoot Waveform**



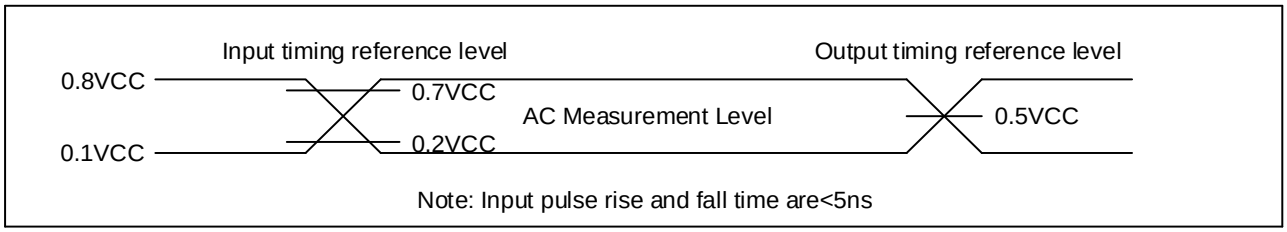

**Maximum Positive Overshoot Waveform**



## 9.4. CAPACITANCE MEASUREMENT CONDITIONS

| Symbol | Parameter | Min | Typ. | Max | Unit | Conditions |
|---|---|---|---|---|---|---|
| CIN | Input Capacitance | | | 12 | pF | VIN=0V |
| COUT | Output Capacitance | | | 16 | pF | VOUT=0V |
| $C_L$ | Load Capacitance | | 30 | | pF | |
| | Input Rise And Fall time | | | 5 | ns | |
| | Input Pause Voltage | 0.1VCC to 0.8VCC | | | V | |
| | Input Timing Reference Voltage | 0.2VCC to 0.7VCC | | | V | |
| | Output Timing Reference Voltage | 0.5VCC | | | V | |

**Figure 43. Input/Output Timing Reference Level**



## 9.5. DC CHARACTERISTICS

(T= -40℃~85℃, VCC=1.65~2.0V)

| Symbol | Parameter | Test Condition | Min. | Typ. | Max. | Unit. |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Leakage Current | | | | ±4 | μA |
| $I_{LO}$ | Output Leakage Current | | | | ±4 | μA |
| $I_{CC1}$ | Standby Current | CS#=VCC, $V_{IN}$=VCC or VSS | | 45 | 95 | μA |
| $I_{CC2}$ | Deep Power-Down Current | CS#=VCC, $V_{IN}$=VCC or VSS | | 2 | 16 | μA |
| $I_{CC3}$ | Operating Current (Read) | CLK=0.1VCC / 0.9VCC at 120MHz, Q=Open(x4 I/O) | | 15 | 20 | mA |
| | | CLK=0.1VCC / 0.9VCC at 80MHz, Q=Open(x4 I/O) | | 13 | 18 | mA |
| $I_{CC4}$ | Operating Current (PP) | CS#=VCC | | | 20 | mA |
| $I_{CC5}$ | Operating Current (WRSR) | CS#=VCC | | | 20 | mA |
| $I_{CC6}$ | Operating Current (SE) | CS#=VCC | | | 20 | mA |
| $I_{CC7}$ | Operating Current (BE) | CS#=VCC | | | 20 | mA |
| $I_{CC8}$ | Operating Current (CE) | CS#=VCC | | | 20 | mA |
| $V_{IL}$ | Input Low Voltage | | -0.5 | | 0.2VCC | V |
| $V_{IH}$ | Input High Voltage | | 0.7VCC | | VCC+0.4 | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ =100μA | | | 0.2 | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ =-100μA | VCC-0.2 | | | V |

Note:

1. Typical value tested at T = 25℃, VCC = 1.8V.
2. Value guaranteed by design and/or characterization, not 100% tested in production.

## 9.6. AC CHARACTERISTICS

(T= -40℃~85℃, VCC=1.65~2.0V, $C_L$=30pf)

| Symbol | Parameter | Min. | Typ. | Max. | Unit. |
|---|---|---|---|---|---|
| $f_{C1}$ | Serial Clock Frequency For: all command except for 03H and RPMC commands | | | 120 | MHz |
| $f_{C2}$ | Serial Clock Frequency For: RPMC commands () | | | 104 | MHz |
| $f_R$ | Serial Clock Frequency For: Read (03H) | | | 80 | MHz |
| $t_{CLH}$ | Serial Clock High Time | 45% (1/Fc) | | | ns |
| $t_{CLL}$ | Serial Clock Low Time | 45% (1/Fc) | | | ns |
| $t_{CLCH}$ | Serial Clock Rise Time (Slew Rate) | 0.2 | | | V/ns |
| $t_{CHCL}$ | Serial Clock Fall Time (Slew Rate) | 0.2 | | | V/ns |
| $t_{SLCH}$ | CS# Active Setup Time | 10 | | | ns |
| $t_{CHSH}$ | CS# Active Hold Time | 10 | | | ns |
| $t_{SHCH}$ | CS# Not Active Setup Time | 10 | | | ns |
| $t_{CHSL}$ | CS# Not Active Hold Time | 10 | | | ns |
| $t_{SHSL}$ | CS# High Time (Read/Write) | 40 | | | ns |
| $t_{SHQZ}$ | Output Disable Time | | | 12 | ns |
| $t_{CLQX}$ | Output Hold Time | 1.2 | | | ns |
| $t_{DVCH}$ | Data In Setup Time | 3 | | | ns |
| $t_{CHDX}$ | Data In Hold Time | 3 | | | ns |
| $t_{CLQV}$ | Clock Low To Output Valid | | | 9 | ns |
| $t_{DP}$ | CS# High To Deep Power-Down Mode | | | 20 | μs |
| $t_{RES1}$ | CS# High To Standby Mode Without Electronic Signature Read | | | 20 | μs |
| $t_{RES2}$ | CS# High To Standby Mode With Electronic Signature Read | | | 20 | μs |
| $t_{SUS}$ | CS# High To Next Command After Suspend | | | 20 | μs |
| $t_{RS}$ | Latency Between Resume And Next Suspend | 100 | | | μs |
| $t_{RST}$ | CS# High To Next Command After Reset (Except From Erase) | | | 30 | μs |
| $t_{RST\_E}$ | CS# High To Next Command After Reset (From Erase) | | | 12 | ms |
| $t_W$ | Write Status Register Cycle Time | | 5 | 30 | ms |
| $t_{PP}$ | Page Programming Time | | 0.5 | 2.4 | ms |
| $t_{BP1}$ | Byte Program Time (First Byte) | | 25 | 50 | μs |
| $t_{BP2}$ | Additional Byte Program Time (After First Byte) | | 2.5 | 5 | μs |
| $t_{SE}$ | Sector Erase Time | | 70 | 400 | ms |
| $t_{BE1}$ | Block Erase Time (32K Bytes) | | 0.16 | 0.8 | s |
| $t_{BE2}$ | Block Erase Time (64K Bytes) | | 0.3 | 1.2 | s |
| $t_{CE}$ | Chip Erase Time (GD25LR128D) | | 50 | 120 | s |

| | | | | | |
|---|---|---|---|---|---|
| $t_{WRKR}$ | Write Root Key Register | | 3 | 5.5 | ms |
| $t_{UHKR}$ | Update HMAC Key Register | | 200 | | μs |
| $t_{IMC}$ | Increment Monotonic Counter | | 0.1 | 300 | ms |
| $t_{RMC}$ | Request Monotonic Counter | | 0.2 | 2 | ms |

Note:

1. Typical value tested at T = 25℃.
2. Value guaranteed by design and/or characterization, not 100% tested in production.

**Figure 44. Serial Input Timing**

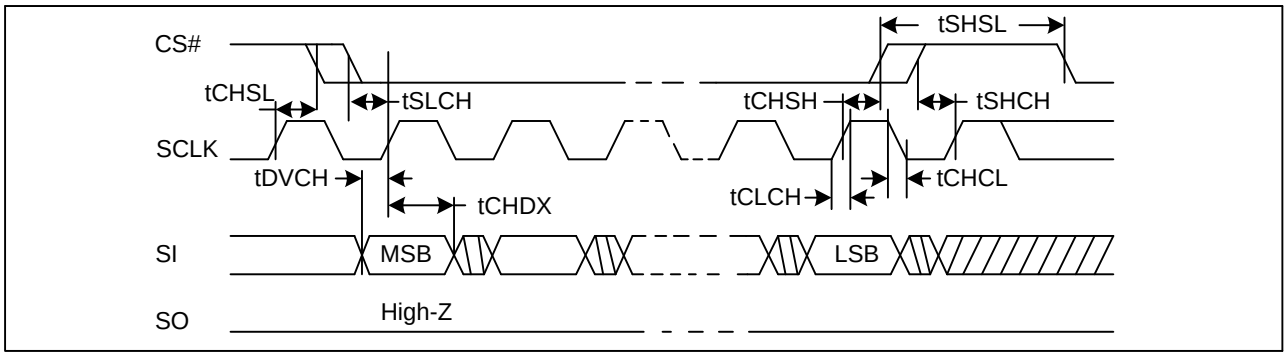



**Figure 45. Output Timing**

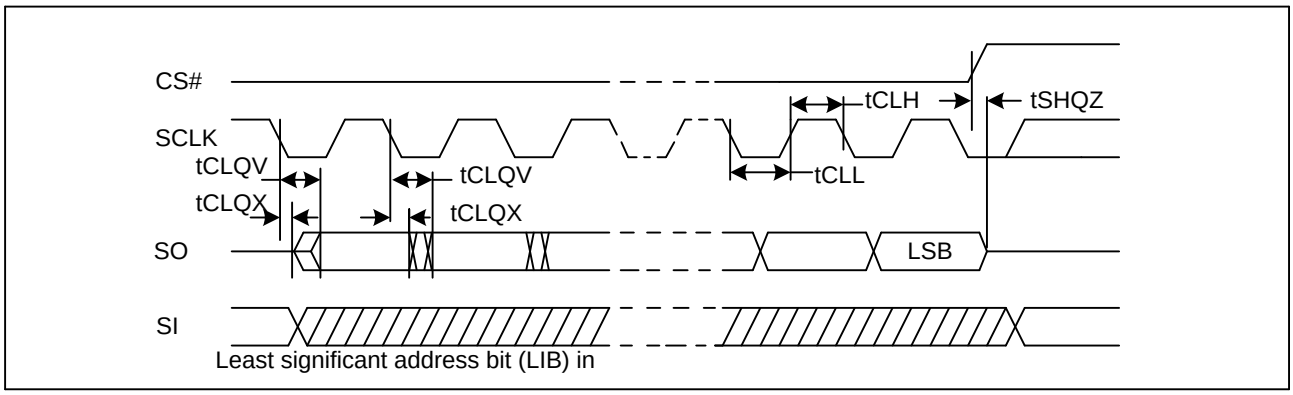
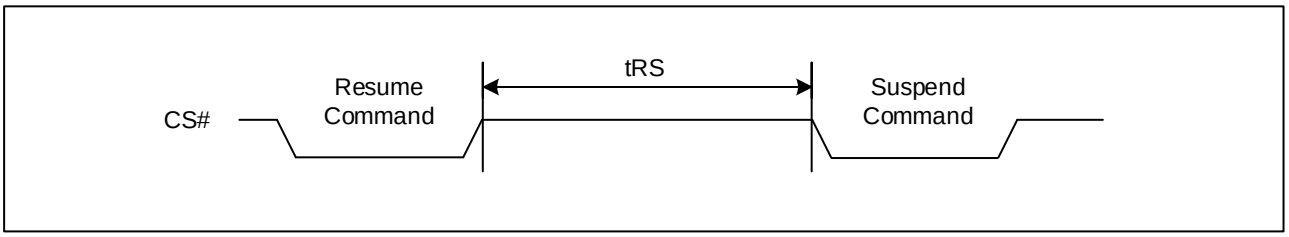
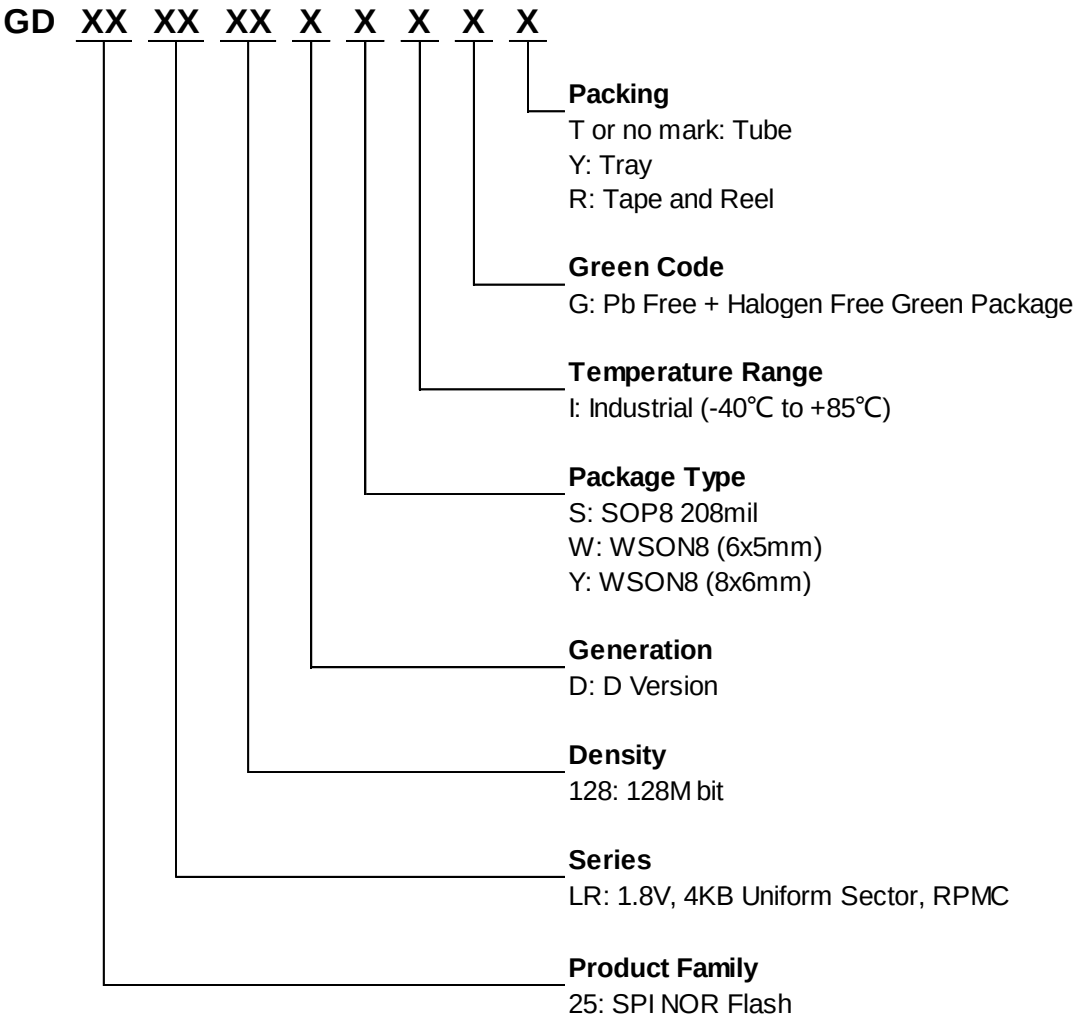



**Figure 46. Resume to Suspend Timing Diagram**



# 10. ORDERING INFORMATION



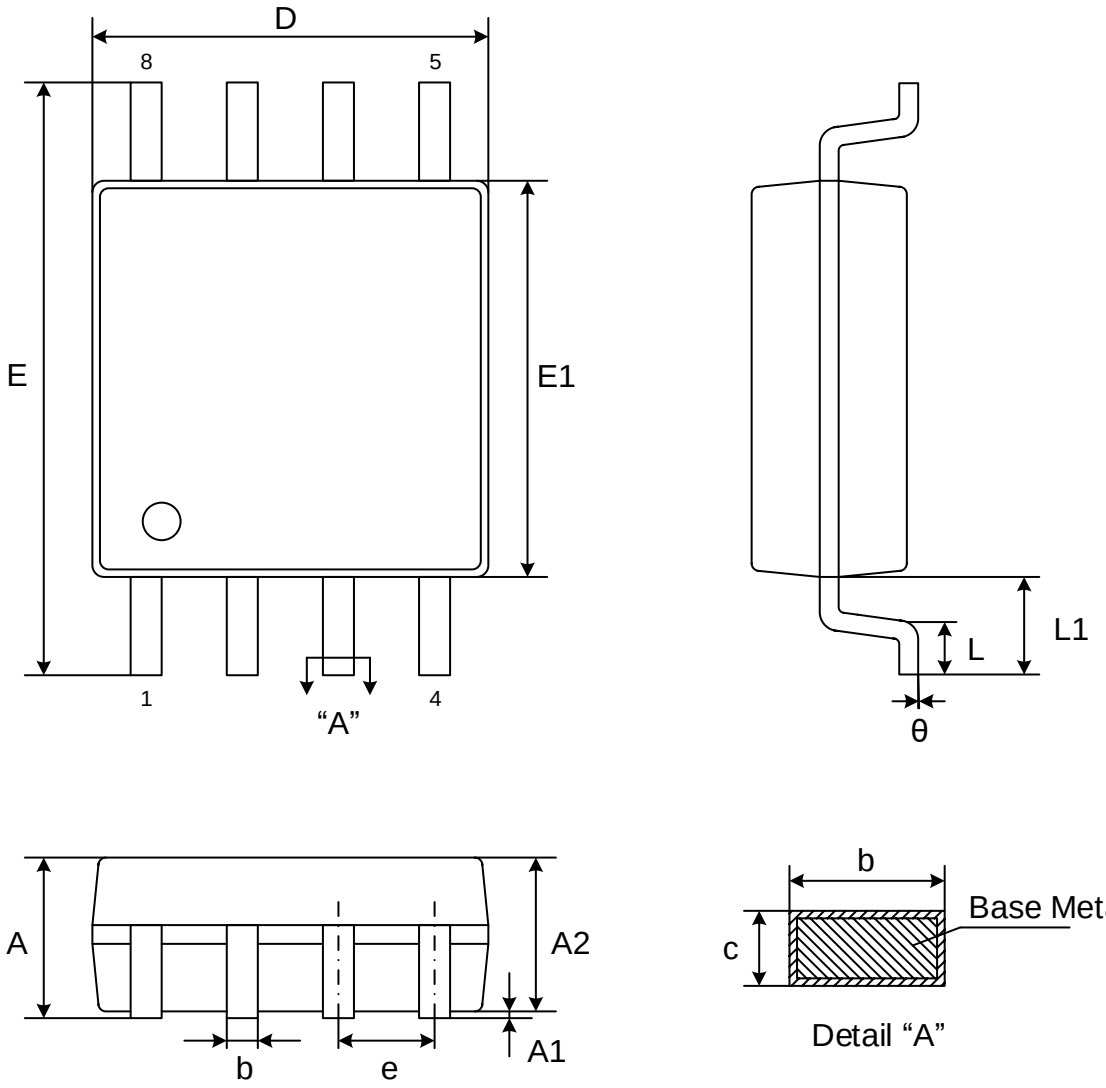
## 10.1. Valid Part Numbers

Please contact GigaDevice regional sales for the latest product selection and available form factors.

| Product Number | Density | Package Type |
|---|---|---|
| GD25LR128DSIG | 128Mbit | SOP8 208mil |
| GD25LR128DWIG | 128Mbit | WSON8 (6x5mm) |
| GD25LR128DYIG | 128Mbit | WSON8 (8x6mm) |

# 11. PACKAGE INFORMATION

## 11.1. Package SOP8 208MIL



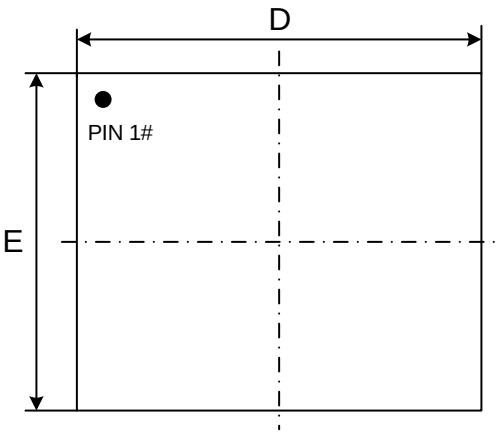

**Dimensions**

| Symbol / Unit | | A | A1 | A2 | b | c | D | E | E1 | e | L | L1 | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | Min | - | 0.05 | 1.70 | 0.31 | 0.15 | 5.13 | 7.70 | 5.18 | 1.27 | 0.50 | 1.31 | 0° |
| | Nom | - | 0.15 | 1.80 | 0.41 | 0.20 | 5.23 | 7.90 | 5.28 | | - | | - |
| | Max | 2.16 | 0.25 | 1.90 | 0.51 | 0.25 | 5.33 | 8.10 | 5.38 | | 0.85 | | 8° |

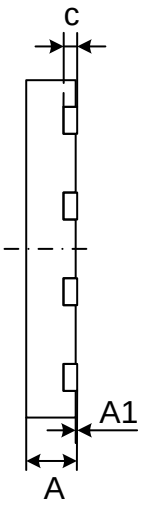
Note:

1. Both the package length and width do not include the mold flash.
2. Seating plane: Max. 0.1mm.

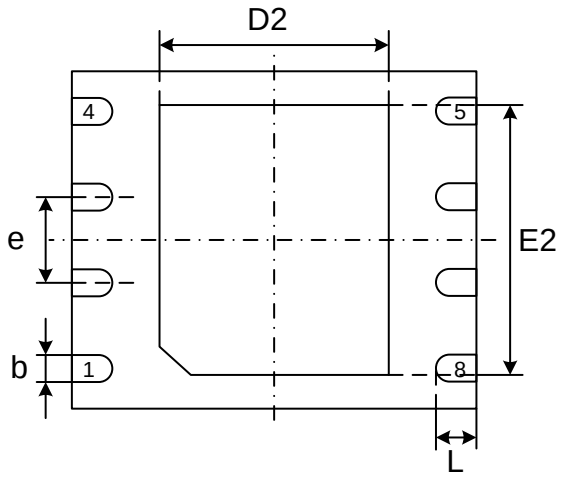
## 11.2. Package WSON8 (6x5mm)




Top View




Side View




Bottom View

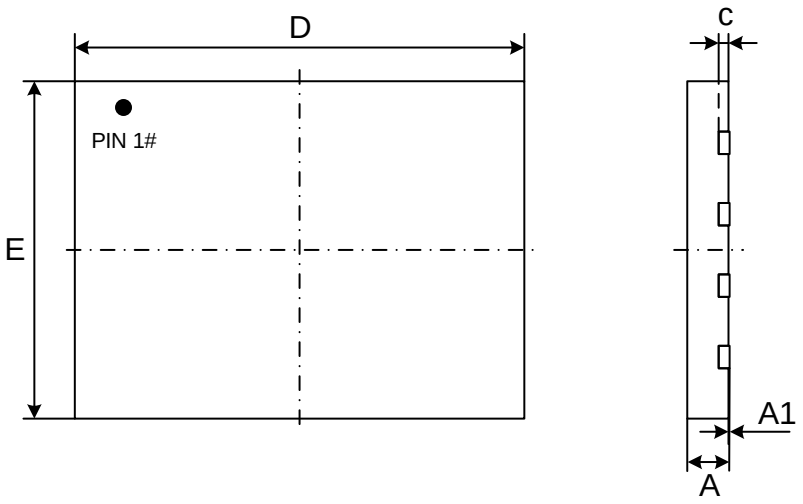
**Dimensions**

| Symbol / Unit | | A | A1 | c | b | D | D2 | E | E2 | e | L |
|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | Min | 0.70 | 0.00 | 0.180 | 0.35 | 5.90 | 3.30 | 4.90 | 3.90 | 1.27 | 0.50 |
| | Nom | 0.75 | 0.02 | 0.203 | 0.40 | 6.00 | 3.40 | 5.00 | 4.00 | | 0.60 |
| | Max | 0.80 | 0.05 | 0.250 | 0.50 | 6.10 | 3.50 | 5.10 | 4.10 | | 0.75 |

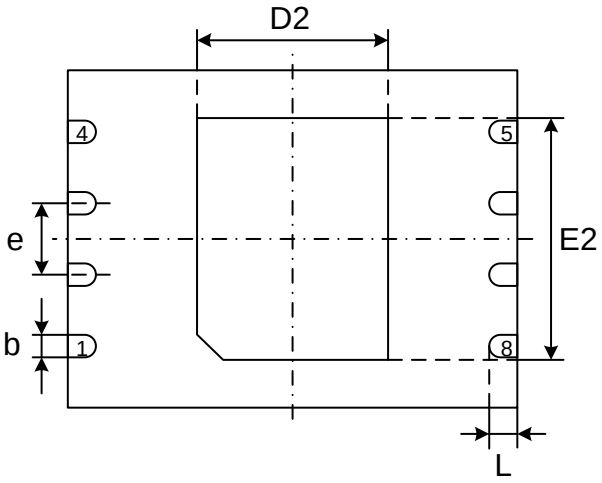
Note:

1. Both the package length and width do not include the mold flash.
2. The exposed metal pad area on the bottom of the package is floating.
3. Coplanarity ≤0.08mm. Package edge tolerance≤0.10mm.
4. The lead shape may be of little difference according to different package lead frames. These lead shapes are compatible with each other.

## 11.3. Package WSON8 (8x6mm)




Top View Side View




Bottom View

**Dimensions**

| Symbol / Unit | | A | A1 | c | b | D | D2 | E | E2 | e | L |
|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | Min | 0.70 | 0.00 | 0.180 | 0.35 | 7.90 | 3.30 | 5.90 | 4.20 | 1.27 | 0.45 |
| | Nom | 0.75 | 0.02 | 0.203 | 0.40 | 8.00 | 3.40 | 6.00 | 4.30 | | 0.50 |
| | Max | 0.80 | 0.05 | 0.250 | 0.45 | 8.10 | 3.50 | 6.10 | 4.40 | | 0.55 |

Note:

1. Both the package length and width do not include the mold flash.
2. The exposed metal pad area on the bottom of the package is floating.
3. Coplanarity ≤0.08mm. Package edge tolerance≤0.10mm.
4. The lead shape may be of little difference according to different package lead frames. These lead shapes are compatible with each other.

# 12. REVISION HISTORY

| Version No | Description | Page | Date |
|---|---|---|---|
| 1.0 | Initial Release | All | 2019-2-25 |
| 1.1 | Add RMPC Commands Description<br>Add $t_{WRKR}$,$t_{UHKR}$,$t_{IMC}$, $t_{RMC}$ | P58-64<br>P71 | 2019-7-18 |
| 1.2 | Add Package WSON8 8x6mm | P76 | 2019-8-7 |
| 1.3 | Remove SFDP table (Contact GigaDevice for AN_SFDP) | --- | 2020-4-7 |
| 1.4 | Modify tWRKR from 500us to 3~5.5ms<br>Modify tUHKR typical value from 100us to 200us<br>Modify tIMC from 0.06~200ms to 0.1~300ms<br>Modify tRMC to 90~400us to 0.2~2ms | P62<br>P62<br>P62<br>P63 | 2020-4-20 |
| 1.5 | Add fC2<br>Modify tSLCH tCHSH tSHCH tCHSL tSHSL tSHQZ tDVCH tCHDX tCLQV | P62<br>P62 | 2020-5-18 |
| 1.6 | Modify the description of BBh/EBh commands | P4<br>P24-26 | 2021-4-23 |

# Important Notice

This document is the property of GigaDevice Semiconductor (Beijing) Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the “Product”), is owned by the Company under the intellectual property laws and treaties of the People’s Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do release the Company as well as it’s suppliers and/or distributors from any claim, damage, or other liability arising from or related to all Unintended Uses of the Products. Customers shall indemnify and hold the Company as well as it’s suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Products. Customers shall discard the device according to the local environmental law.

**Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and the Products and services described herein at any time, without notice.**

单击下面可查看定价，库存，交付和生命周期等信息

>>GigaDevice(兆易创新)