# IS25LP032/064

## 32M-BIT/ 64M-BIT

### 3V- MULTI I/O SERIAL FLASH MEMORY WITH 133MHZ SPI BUS & QUAD I/O (QPI) DTR INTERFACE

**DATA SHEET**

# 32M-BIT/ 64M-BIT

### 3V- MULTI I/O SERIAL FLASH MEMORY WITH 133MHZ SPI BUS & QUAD I/O (QPI) DTR INTERFACE

## FEATURES

- **Industry Standard Serial Interface**
  - IS25LP064:64M-bit/ 8M-byte
  - IS25LP032:32M-bit/ 4M-byte
  - 256-bytes per Programmable Page
  - Supports standard SPI, Fast, Dual, Dual I/O, QPI, DTR, Dual DTR I/O, and QPI DTR SPI
  - Supports Serial Flash Discoverable Parameters (SFDP)

- **High Performance Serial Flash (SPI)**
  - 50MHz Normal and 133Mhz Fast Read
  - 532 MHz equivalent QPI SPI
  - DTR (Dual Transfer Rate) up to 66MHz
  - Selectable dummy cycles
  - Configurable drive strength
  - Supports SPI Modes 0 and 3
  - More than 100,000 erase/program cycles
  - More than 20-year data retention

- **Flexible & Efficient Memory Architecture**
  - Chip Erase:128Mbit with Uniform: Sector/Block Erase (4K/32K/64K-Byte)
  - Program 1 to 256 bytes per page
  - Program/Erase Suspend & Resume

- **Efficient Read and Program modes**
  - Low Instruction Overhead Operations
  - Continuous Read 8/16/32/64-Byte Wrap
  - Selectable burst length
  - QPI for reduced instruction overhead
  - Allows XIP operations (execute in place)

- **Low Power with Wide Temp. Ranges**
  - Single 2.3V to 3.6V Voltage Supply
  - 10 mA Active Read Current
  - 5 µA Standby Current
  - Deep Power Down
  - Temp Grades:
    Extended: -40°C to +105°C
    Auto Grade: up to +125°C (call factory)

- **Advanced Security Protection**
  - Software and Hardware Write Protection
  - Power Supply lock protect
  - 4x256-Byte dedicated security area with user-lockable bits, (OTP) One Time Programmable Memory
  - 128 bit Unique ID for each device

- **Industry Standard Pin-out & Packages**
  - JM =16-pin SOIC 300mil
  - JB = 8-pin SOIC 208mil
  - JF = 8-pin VSOP 208mil
  - JK = 8-pin WSON 6x5mm
  - JL = 8-pin WSON 8x6mm
  - KGD (call factory)

## GENERAL DESCRIPTION

The IS25LP032/064 Serial Flash memory offers a storage solution with the flexibility and performance in a simplified pin count package. ISSI's "Industry Standard Serial Interface" flash are for systems that have limited space, pins, and power. The IS25LP128 are accessed through a 4-wire SPI Interface consisting of a Serial Data Input (SI), Serial Data Output (SO), Serial Clock (SCK), and Chip Enable (CE#) pins, which also serve as multi-function I/O (see pin descriptions).

The device supports the standard Serial Peripheral Interface (SPI), Dual/Quad output (SPI), and Dual/Quad I/O (SPI). Clock frequencies of up to 133MHz allow for equivalent clock rates of up to 532MHz (133MHz x 4) allowing more than 66MBytes/S of throughput. The IS25xP series of flash adds support for DTR (Double Transfer Rate) commands that transfer address and read data on both edges of the clock. These transfer rates can outperform 16-bit Parallel Flash memories allowing for efficient memory access for a XIP (execute in place) operation.

The memory array is organized into programmable pages of 256-bytes each. The IS25LP128 supports page program mode where 1 to 256 bytes of data can be programmed into the memory with one command. QPI (Quad Peripheral Interface) supports 2-cycles instruction cycles further reducing instruction times. Pages can be erased in groups of 4K-byte sectors, 32K-byte blocks, 64K-byte blocks, and/or the entire chip. The uniform sectors and blocks allow greater flexibility for a variety of applications requiring solid data retention.

## GLOSSARY

### *Standard SPI*

The IS25LP032/064 is accessed through a 4-wire SPI Interface consisting of Serial Data Input (SI), Serial Data Output (SO), Serial Clock (SCK), and Chip Enable (CE#) pins. Instructions are inputted through the SI pin to encode instructions, addresses, or input data to the device on the rising edge of SCK. The DO pin is used to read data or check the status of the device on the falling edge of SCK. This device supports SPI bus operation mode (0,0) and (1,1).

### *Mutil I/O SPI*

The IS25LP032/064 allows accessing enhanced SPI protocol to use Dual output, Dual input and output, and Quad input and output operation. Executing these instructions through SPI mode will achieve double or quadruple the transfer bandwidth for READ and PROGRAM.

### *Quad I/O (QPI)*

The IS25LP032/064 can enable QPI protocol by issuing an "Enter QPI mode (35h)" command. The QPI mode uses four IO pins for input and output to reduce SPI instruction overhead and increase output bandwidth. QPI mode can exit by issuing an "Exit QPI (F5h) command. A power reset or software reset can also return the device into the standard SPI mode. SI and SO pins become bidirectional IO0 and IO1, and WP# and HOLD# pins become IO2 and IO3 respectively during QPI mode.

### *DTR*

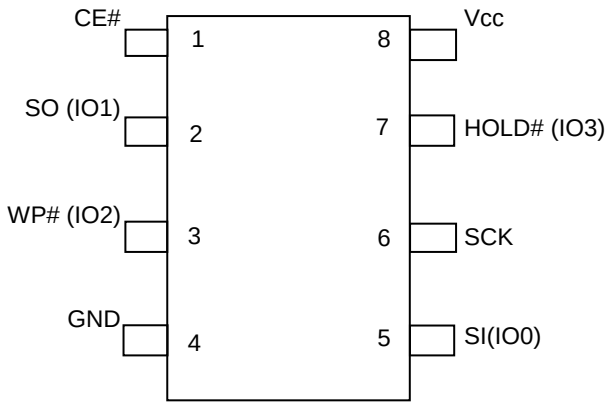
In addition to SPI and QPI features, IS25LP032/064 also supports DTR READ. DTR allows high data throughput while running at lower clock frequencies. As DTR READ option uses both rising and falling clock to drive output, the dummy cycles are reduced by half as well.

### *Programmable drive strength and Selectable burst setting.*

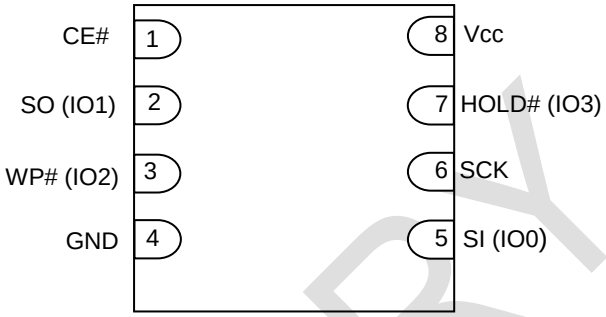
The IS25LP032/064 offers programmable output drive strength and selectable burst (wrap) length features to increase the efficiency and performance of READ operations. The driver strength and burst setting features are controlled by setting READ registers. A total of six different drive strengths and four different burst sizes (8/16/32/64Bytes) are selectable.

## TABLE OF CONTENTS

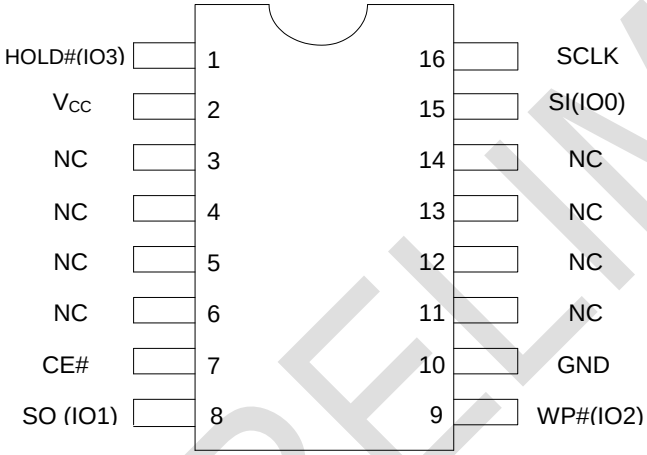
## 1. PIN CONFIGURATION




**8-pin SOIC 208mil**
**8-pin VSOP 208mil**




**8-pin WSON 6x5mm**
**8-pin WSON 8x6mm**

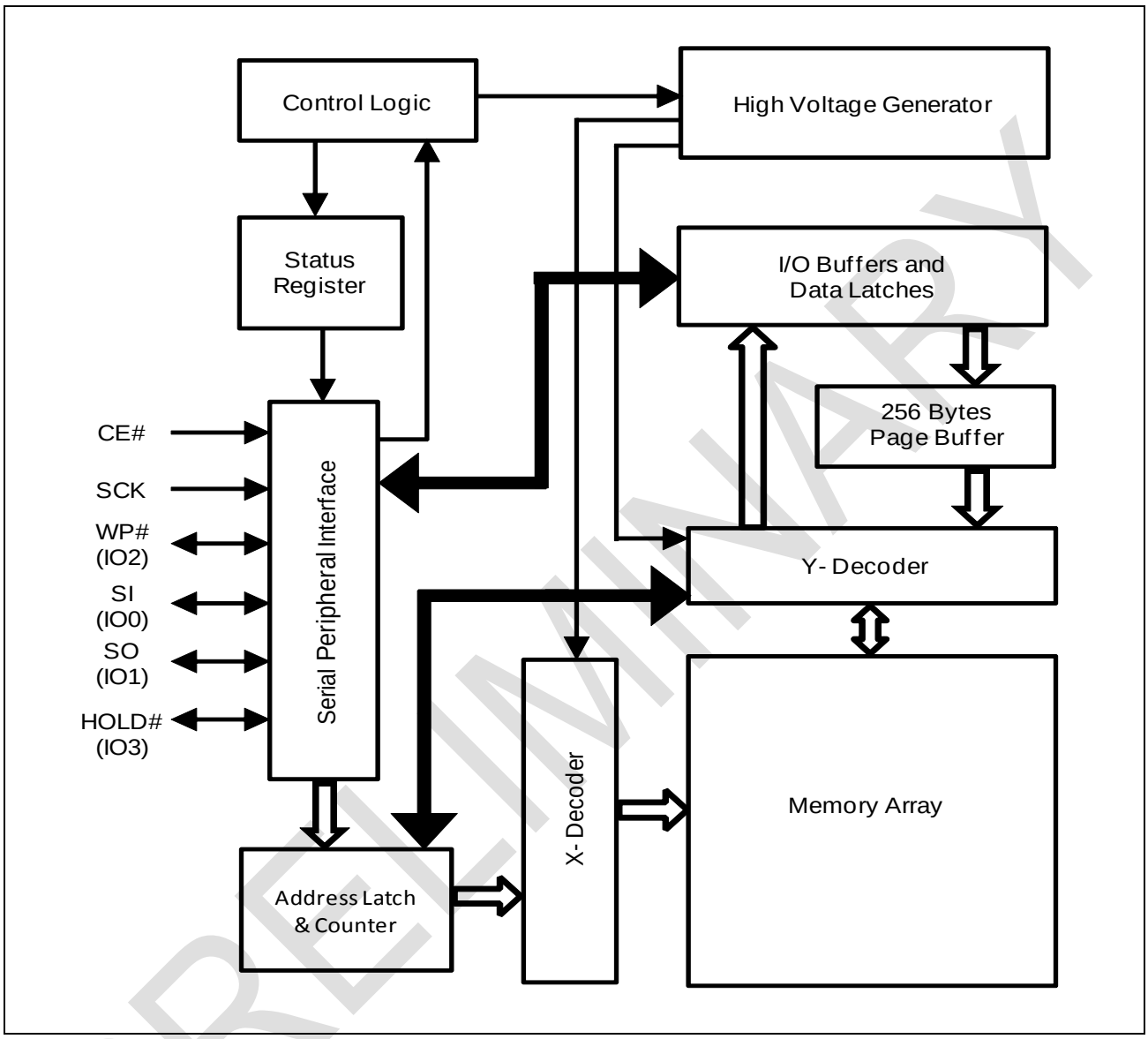



**16-Pin SOIC 300mil**

## 2. PIN DESCRIPTIONS

| SYMBOL | TYPE | DESCRIPTION |
|---|---|---|
| CE# | INPUT | Chip Enable: CE# low activates the internal circuitries for device operation. CE# high deselects the device and switches into standby mode to reduce the power consumption. When the device is not selected, data will not be accepted via the serial input pin (SI), and the serial output pin (SO) will remain in a high impedance state. |
| SCK | INPUT | Serial Data Clock |
| SI (IO0) | INPUT/OUTPUT | Serial Data Input/Output |
| SO (IO1) | INPUT/OUTPUT | Serial Data Input/Output |
| GND | | Ground |
| Vcc | | Device Power Supply |
| WP# (IO2) | INPUT/OUTPUT | Write Protect/Serial Data Output: When the WP# pin is low, memory array write-protection depends on the setting of BP3, BP2, BP1, and BP0 bits in the Status Register. |
| HOLD# (IO3) | INPUT/OUTPUT | HOLD: Pause serial communication by the master device without resetting the serial sequence.<br>When the QE bit of Status Register is set for “1”, the pin function becomes an I/O pin. |

## 3. BLOCK DIAGRAM



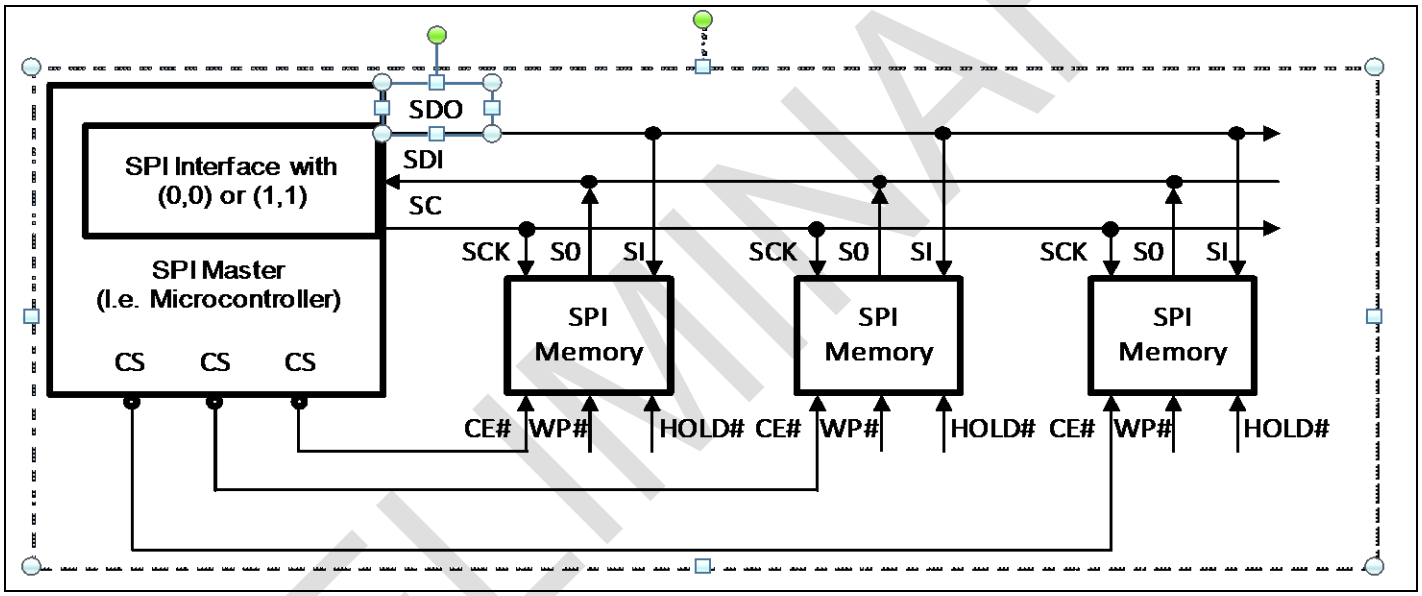
## 4. SPI MODES DESCRIPTION

Multiple IS25LP032/064 devices can be connected on the SPI serial bus and controlled by a SPI Master, i.e. microcontroller, as shown in Figure 4.1 The devices support either of two SPI modes:

Mode 0 (0, 0)
Mode 3 (1, 1)

The difference between these two modes is the clock polarity. When the SPI master is in stand-by mode, the serial clock remains at "0" (SCK = 0) for Mode 0 and the clock remains at "1" (SCK = 1) for Mode 3. Please refer to Figure 4.2 and Figure 4.3. For both modes, the input data is latched on the rising edge of Serial Clock (SCK), and the output data is available from the falling edge of SCK.

**Figure 4.1 Connection Diagram among SPI Master and SPI Slaves (Memory Devices)**




**Note : SI and SO pins become bidirectional IO0 and IO1, and WP# and HOLD# pins become IO2 and IO3 respectively during QPI mode.**

**Figure 4.2 SPI Modes Support**

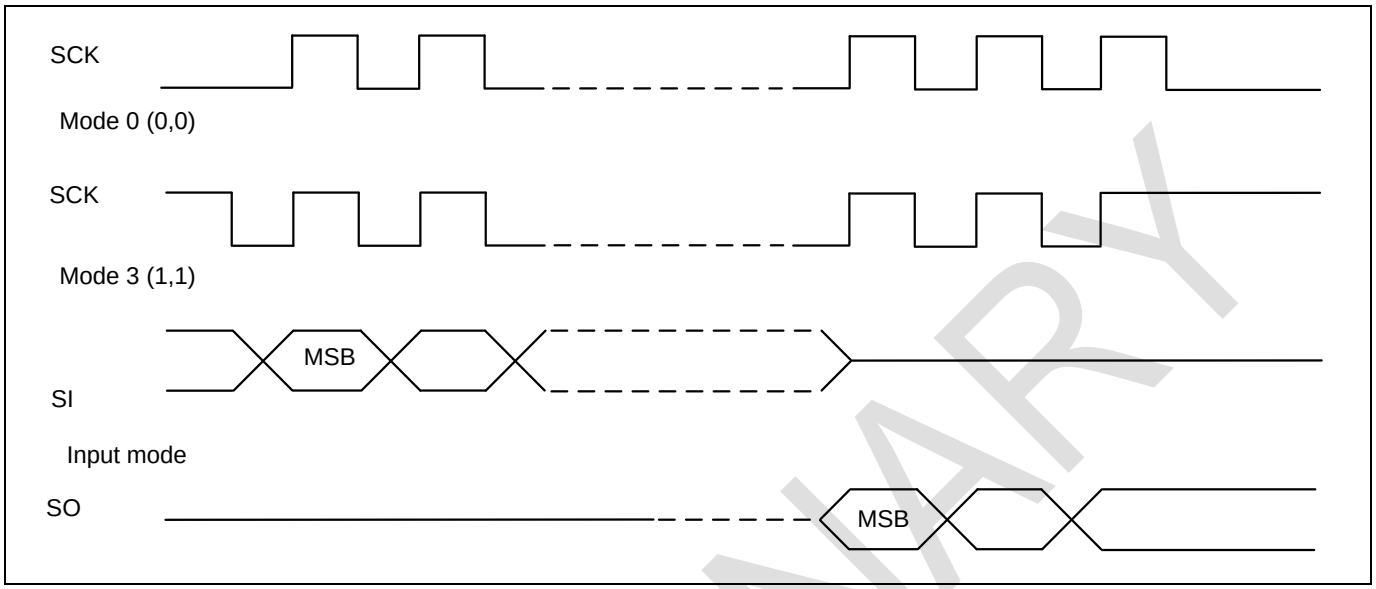
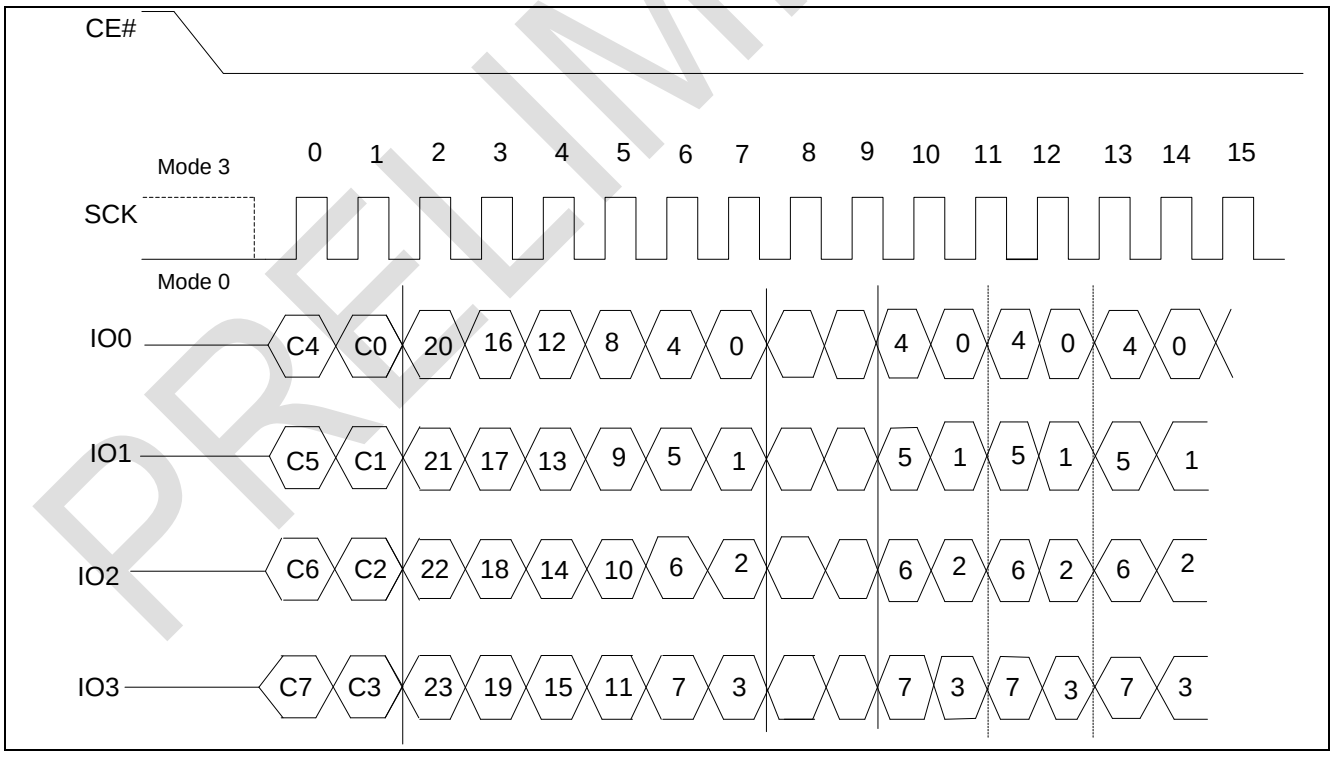



**Figure 4.3 QPI Modes Support**



# 5. SYSTEM CONFIGURATION

The IS25LP032/064 is designed to interface directly with the synchronous Serial Peripheral Interface (SPI) of the Motorola MC68HCxx series of microcontrollers or any SPI interface-equipped system controllers.

The memory array of IS25LP032/064 is divided into uniform 4 KByte sectors or uniform 32K/64 KByte blocks (a block consists of sixteen adjacent sectors).

Table 5.1 illustrates the memory map of the device. The Status Register controls how the memory is mapped.

## 5.1 BLOCK/SECTOR ADDRESSES

**Table 5.1 Block/Sector Addresses of IS25LP032/064**

| Memory Density | | Block No. (64Kbyte) | Block No. (32Kbyte) | Sector No. | Sector Size (KBytes) | Address Range |
|---|---|---|---|---|---|---|
| 32Mb | 64Mb | Block 0 | Block 0 | Sector 0 | 4 | 000000h - 0000FFh |
| | | | | : | : | : |
| | | | Block 1 | : | : | : |
| | | | | Sector 15 | 4 | 00FF00h - 00FFFFh |
| | | Block 1 | Block 2 | Sector 16 | 4 | 010000h – 0100FFh |
| | | | | : | : | : |
| | | | Block 3 | : | : | : |
| | | | | Sector 31 | 4 | 01FF00h - 01FFFFh |
| | | Block 2 | Block 4 | Sector 32 | 4 | 020000h - 0200FFh |
| | | | | : | : | : |
| | | | Block 5 | : | : | : |
| | | | | Sector 47 | 4 | 02FF00h – 02FFFFh |
| | | : | : | : | : | : |
| | | Block 63 | Block 126 | : | : | : |
| | | | Block 127 | Sector 1023 | 4 | 3FFF00h – 3FFFFFh |
| | | : | : | : | : | : |
| | | : | : | : | : | : |
| | | | | : | : | : |
| | | | : | : | : | : |
| | | | | : | : | : |
| | | : | : | : | : | : |
| | | | | : | : | : |
| | | | : | : | : | : |
| | | | | : | : | : |
| | | Block 127 | : | : | : | : |
| | | | | : | : | : |
| | | | Block 255 | : | : | : |
| | | | | Sector 2047 | 4 | 7FFF00h – 7FFFFFh |

# 6. REGISTERS

The IS25LP032/064 has three sets of Registers: Status, Function and Read.

## 6.1. STATUS REGISTER

Status Register Format and Status Register Bit Definitions are described in Tables 6.1 & 6.2.

**Table 6.1 Status Register Format**

| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| | SRWD | QE | BP3 | BP2 | BP1 | BP0 | WEL | WIP |
| Default (flash bit) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Note : The default value of the BP3, BP2, BP1, BP0, and SRWD bits were set to "0" at factory.**

**Table 6.2 Status Register Bit Definition**

| Bit | Name | Definition | Read-/Write | Non-Volatile bit |
|---|---|---|---|---|
| Bit 0 | WIP | Write In Progress Bit:<br>"0" indicates the device is ready(default)<br>"1" indicates a write cycle is in progress and the device is busy | R | No |
| Bit 1 | WEL | Write Enable Latch:<br>"0" indicates the device is not write enabled (default)<br>"1" indicates the device is write enabled | R/W | No |
| Bit 2 | BP0 | Block Protection Bit: (See Tables 6.5 for details)<br>"0" indicates the specific blocks are not write-protected (default)<br>"1" indicates the specific blocks are write-protected | R/W | Yes |
| Bit 3 | BP1 | | | |
| Bit 4 | BP2 | | | |
| Bit 5 | BP3 | | | |
| Bit 6 | QE | Quad Enable bit:<br>"0" indicates the Quad output function disable (default)<br>"1" indicates the Quad output function enable | R/W | Yes |
| Bit 7 | SRWD | Status Register Write Disable: (See Table 7.1 for details)<br>"0" indicates the Status Register is not write-protected (default)<br>"1" indicates the Status Register is write-protected | R/W | Yes |

The BP0, BP1, BP2, BP3, SRWD, and QE are non-volatile memory cells that can be written by a Write Status Register (WRSR) instruction. The default value of the BP2, BP1, BP0, and SRWD bits were set to "0" at factory. The Status Register can be read by the Read Status Register (RDSR).

The function of Status Register bits are described as follows:

**WIP bit**: The Write In Progress (WIP) bit is read-only, and can be used to detect the progress or completion of a program or erase operation. When the WIP bit is "0", the device is ready for write status register, program or erase operation. When the WIP bit is "1", the device is busy.

**WEL bit**: The Write Enable Latch (WEL) bit indicates the status of the internal write enable latch. When the WEL is "0", the write enable latch is disabled and all write operations, including write status register, write configuration register, page program, sector erase, block and chip erase operations are inhibited. When the WEL bit is "1", write operations are allowed. The WEL bit is set by a Write Enable (WREN) instruction. Each write register, program and erase instruction must be preceded by a WREN instruction. The WEL bit can be reset by a Write Disable (WRDI) instruction. It will automatically be the reset after the completion of any write operation.

**BP3, BP2, BP1, BP0 bits**: The Block Protection (BP3, BP2, BP1 and BP0) bits are used to define the portion of the memory area to be protected. Refer to Tables 6.3 for the Block Write Protection (BP) bit settings. When a defined combination of BP3, BP2, BP1 and BP0 bits are set, the corresponding memory area is protected. Any program or erase operation to that area will be inhibited.

**Note : A Chip Erase (CER) instruction will be ignored unless all the Block Protection Bits are "0"s.**

**SRWD bit**: The Status Register Write Disable (SRWD) bit operates in conjunction with the Write Protection (WP#) signal to provide a Hardware Protection Mode. When the SRWD is set to "0", the Status Register is not write-protected. When the SRWD is set to "1" and the WP**#** is pulled low ($V_{IL}$), the bits of Status Register (SRWD, BP3, BP2, BP1, BP0) become read-only, and a WRSR instruction will be ignored. If the SRWD is set to "1" and WP# is pulled high ($V_{IH}$), the Status Register can be changed by a WRSR instruction.

**QE bit**: The Quad Enable (QE) is a non-volatile bit in the status register that allows quad operation. When the QE bit is set to "0", the pin WP# and HOLD# are enabled. When the QE bit is set to "1", the IO2 and IO3 pins are enabled.

WARNING: The QE bit must be set to 0 if WP# or HOLD# pin is tied directly to the power supply or ground.

**Table 6.3 Block assignment by Block Write Protect (BP) Bits.**

| Status Register Bits | | | | **IS25LP032** Protected Memory Area (32Mb) | |
|---|---|---|---|---|---|
| BP3 | BP2 | BP1 | BP0 | **TBS (T/B selection) = 0, TOP area** | **TBS (T/B selection) = 1, Bottom area** |
| 0 | 0 | 0 | 0 | 0 ( None) | 0 ( None) |
| 0 | 0 | 0 | 1 | 1 (1 block : 63rd) | 1 (1 block : 0th) |
| 0 | 0 | 1 | 0 | 2 (2 block : 62nd and 63rd) | 2 (2 block : 0th and 1st) |
| 0 | 0 | 1 | 1 | 3 (4 blocks : 60th to 63nd) | 3 (4 blocks : 0th to 3rd) |
| 0 | 1 | 0 | 0 | 4 (8 blocks : 56th to 63rd) | 4 (8 blocks : 0th to 7th) |
| 0 | 1 | 0 | 1 | 5 (16 blocks :48th to 63rd) | 5 (16 blocks : 0th to 15th) |
| 0 | 1 | 1 | 0 | 6 (32 blocks : 32nd to 63rd) | 6 (32 blocks : 0th to 31st) |
| 0 | 1 | 1 | 1 | 7 (64 blocks : 0th to 63rd) | 7 (64 blocks : 0th to 63rd) |
| 1 | x | x | X | 8 (128 blocks : 127th to 255th) All blocks | 8 (128 blocks : 0th to 127th) All blocks |

*note x is don't care

| Status Register Bits | | | | **IS25LP064** Protected Memory Area (64Mb) | |
|---|---|---|---|---|---|
| BP3 | BP2 | BP1 | BP0 | **TBS (T/B selection) = 0, TOP area** | **TBS (T/B selection) = 1, Bottom area** |
| 0 | 0 | 0 | 0 | 0 ( None) | 0 ( None) |
| 0 | 0 | 0 | 1 | 1 (1 block : 127th) | 1 (1 block : 0th) |
| 0 | 0 | 1 | 0 | 2 (2 block : 126th and 127th) | 2 (2 block : 0th and 1st) |
| 0 | 0 | 1 | 1 | 3 (4 blocks : 124th to 127th) | 3 (4 blocks : 0th to 3rd) |
| 0 | 1 | 0 | 0 | 4 (8 blocks : 120th to 127th) | 4 (8 blocks : 0th to 7th) |
| 0 | 1 | 0 | 1 | 5 (16 blocks :112nd to 127th) | 5 (16 blocks : 0th to 15th) |
| 0 | 1 | 1 | 0 | 6 (32 blocks : 96th to 127th) | 6 (32 blocks : 0th to 31st) |
| 0 | 1 | 1 | 1 | 7 (64 blocks : 64th to 127th) | 7 (64 blocks : 0th to 63rd) |
| 1 | x | x | x | 8 (128 blocks : 0th to 127th) | 8 (128 blocks : 0th to 127th) |

*note x is don't care

### 6.2. FUNCTION REGISTER

Function Register Format and Bit definition are described in Table 6.4 and 6.5

**Table 6.4 Function Register Format**

| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| | IRL3 | IRL2 | IRL1 | IRL0 | ESUS | PSUS | TBS | Reserved |
| Default | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Note: Bit 0 reserved and must be set to '1'**

**Table 6.5 Function Register Bit Definition**

| Bit | Name | Definition | Read/Write | Non-Volatile bit |
|---|---|---|---|---|
| Bit 0 | Reserved | Reserved set to 1 | Reserved | Reserved |
| Bit 1 | Top/Bottom Selection | Top/Bottom Selection. (See Tables 6.5 for details)<br>"0" indicates Top area.<br>"1" indicates Bottom area. | R/W | Yes |
| Bit 2 | PSUS | Program suspend bit:<br>"0" indicates program is not suspend<br>"1" indicates program is suspend | R | No |
| Bit 3 | ESUS | Erase suspend bit:<br>"0" indicates Erase is not suspend<br>"1" indicates Erase is suspend | R | No |
| Bit 4 | IR Lock 0 | Lock the information row 0:<br>"0" indicates the information row can be programmed<br>"1" indicates the information row can not be programmed | R/W | Yes |
| Bit 5 | IR Lock 1 | Lock the information row 1:<br>"0" indicates the information row can be programmed<br>"1" indicates the information row can not be programmed | R/W | Yes |
| Bit 6 | IR Lock 2 | Lock the information row 2:<br>"0" indicates the information row can be programmed<br>"1" indicates the information row can not be programmed | R/W | Yes |
| Bit 7 | IR Lock 3 | Lock the information row 3:<br>"0" indicates the information row can be programmed<br>"1" indicates the information row can not be programmed | R/W | Yes |

**Note: Table 6.5 Function Register bits 4-7 are one time programmable and cannot be modified**

**Top/Bottom Selection**: BP0~3 area assignment changed from Top or Bottom. See Tables 6.5 for details

**PSUS bit**: The Program Suspend Status bit indicates when a Program operation has been suspended. The PSUS changes to '1' after a suspend command is issued during the program operation. Once the suspended Program resumes, the PSUS bit is reset to '0.'

**ESUS bit**: The Erase Suspend Status indicates when an Erase operation has been suspended. The ESUS bit is '1' after a suspend command is issued during an Erase operation. Once the suspended Erase resumes, the ESUS bit is reset to '0.'

**IR lock bit 0 ~ 3**: The information row lock bits are programmable. If the bit set to "1", it can't be programmed.

### 6.3 READ REGISTERS

Read Register format and Bit definitions pertaining to QPI mode are described below.'

### READ PARAMETER BITS

Table 6.6 defines all bits that control features in SPI/QPI modes. The ODS2, ODS1, ODS0 (P7,P6,P5) bits provide a method to set and control driver strength. The Dummy Cycle bits (P4, P3) define how many dummy cycles are used during various READ modes. The Wrap Length bits (P2, P1, P0) define whether or not Wrap Around is enabled and the depth of data to Wraparound on.

The SET READ PARAMETERS Operation (SRP, C0h) is used to set all the Read Register bits, and can thereby define the Output Driver Strength, Number of Dummy Cycles used during READ modes, Burst length and the Data Wrapping features.

**Table 6.6 Read Parameter Table**

| | P7 | P6 | P5 | P4 | P3 | P2 | P1 | P0 |
|---|---|---|---|---|---|---|---|---|
| | ODS2 | ODS1 | ODS0 | Dummy Cycles | Dummy Cycles | Wrap Enable | Wrap Length | Wrap Length |
| Default (flash bit) | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |

**Table 6.7 Burst Length data**

| | P1 | P0 |
|---|---|---|
| 8 bytes | 0 | 0 |
| 16 bytes | 0 | 1 |
| 32 bytes | 1 | 0 |
| 64 bytes | 1 | 1 |

**Table 6.8 Wrap Function**

| | P2 |
|---|---|
| Disable | 0 |
| Enable | 1 |

**Table 6.9 Read Dummy Cycles.**

| Read Modes | P4,P3 = 00 | P4,P3 = 01 | P4,P3 = 10 | Max Freq | Mode |
|---|---|---|---|---|---|
| Normal Read<br>03h | 0 | 0 | 0 | 50MHz | SPI |
| Fast read<br>0Bh | 8 | 8 | 8 | 133MHz | SPI |
| Fast read<br>0Bh | 6 | 4 | 8 | 4cc : 84MHz<br>6cc : 104MHz<br>8cc : 133MHz | QPI |
| Dual IO Read [1]<br>BBh | 4 | 4 | 8 | 4cc : 104MHz<br>8cc : 133MHz | SPI |
| Fast Read Dual Output<br>3Bh | 8 | 8 | 8 | 133MHz | SPI |
| Quad IO Read [2]<br>EBh | 6 | 4 | 8 | 4cc : 84MHz<br>6cc : 104MHz<br>8cc : 133MHz | SPI , QPI |

**Notes :**
**1. When 4 Dummy cycles are used the max clock frequency is 104MHz; when 8 dummy cycles are used the max clock frequency is 133MHz.**
**2. When 4 Dummy cycles are used the max clock frequency is 84MHz; when 6 dummy cycles are used the max clock frequency is 104MHz; when 8 dummy cycles are used the max clock frequency is 133MHz.**
**3. In DTR mode the dummy cycles are reduced by half.**

**Table 6.10 Driver Strength Table**

| ODS2 | ODS1 | ODS0 | Description | Remark |
|---|---|---|---|---|
| 0 | 0 | 0 | Reserved | |
| 0 | 0 | 1 | 12.50% | |
| 0 | 1 | 0 | 25% | |
| 0 | 1 | 1 | 37.50% | |
| 1 | 0 | 0 | Reserved | |
| 1 | 0 | 1 | 75% | |
| 1 | 1 | 0 | 100% | |
| 1 | 1 | 1 | 50% | Default |

# 7. PROTECTION MODE

The IS25LP032/064 supports hardware and software write-protection mechanisms.

## 7.1 HARDWARE WRITE PROTECTION

The Write Protection (WP#) pin provides a hardware write protection method for BP3, BP2, BP1, BP0, SRWD, and QE in the Status Register. Refer to the section 6.1 STATUS REGISTER.

Write inhibit voltage is 2.1V. All write sequence will be ignored when Vcc drops to 2.1V or lower.

## 7.2 SOFTWARE WRITE PROTECTION

The IS25LP032/064 also provide a software write protection feature. The Block Protection (BP3, BP2, BP1, BP0) bits allow part or the whole memory area to be write-protected.

**Table 7.1 Hardware Write Protection on Status Register**

| SRWD | WP# | Status Register |
|---|---|---|
| 0 | Low | Writable |
| 1 | Low | Protected |
| 0 | High | Writable |
| 1 | High | Writable |

**Note : Before the execution of any program, erase or write status register instruction, the Write Enable Latch (WEL) bit must be enabled by executing a Write Enable (WREN) instruction. If the WEL bit is not enabled, the program, erase or write register instruction will be ignored.**

## 8. DEVICE OPERATION

The IS25LP032/064 utilizes an 8-bit instruction register. Refer to Table 8.1. Instruction Set for details on Instructions and Instruction Codes. All instructions, addresses, and data are shifted in with the most significant bit (MSB) first on Serial Data Input (SI). The input data on SI is latched on the rising edge of Serial Clock (SCK) after Chip Enable (CE#) is driven low ($V_{IL}$). Every instruction sequence starts with a one-byte instruction code and is followed by address bytes, data bytes, or both address bytes and data bytes, depending on the type of instruction. CE# must be driven high ($V_{IH}$) after the last bit of the instruction sequence has been shifted in to end the operation.

**Table 8.1 Instruction Set**

| Instruction Name | Operation | Total Bytes | Mode | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Comments |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NORD | Normal Read Mode | 4 | SPI | 03h | A <23:16> | A <15:8> | A <7:0> | Data out | | | |
| FRD | Fast Read Mode | 5 | SPI QPI | 0Bh | A <23:16> | A <15:8> | A <7:0> | Dummy Byte | Data out | | |
| FRDIO | Fast Read Dual I/O | 3 | SPI | BBh | A <23:16> Dual | A <15:8> Dual | A <7:0> Dual | Axh Dual | Dual Data out | | |
| FRDO | Fast Read Dual Output | 5 | SPI | 3Bh | A <23:16> | A <15:8> | A <7:0> | Dummy Byte | Dual Data out | | |
| FRQIO | Fast Read Quad I/O | 2 | SPI QPI | EBh | A <23:16> Quad | A <15:8> Quad | A <7:0> Quad | Axh Quad | Quad Data out | | |
| FRDTR | Fast Read DTR Mode | 5 | SPI QPI | 0Dh | A <23:16> | A <15:8> | A <7:0> | Dummy Byte | Dual Data out | | |
| FRDDTR | Fast Read Dual I/O DTR | 3 | SPI | BDh | A <23:16> Dual | A <15:8> Dual | A <7:0> Dual | Axh Dual | Dual Data out | | |
| FRQDTR | Fast Read Quad I/O DTR | 5 | SPI QPI | EDh | A <23:16> | A <15:8> | A <7:0> | Dummy Byte | Quad Data out | | |
| PP | Input Page Program | 4 + 256 | SPI QPI | 02h | 02h | A <23:16> | A <15:8> | A <7:0> | PD +256byte | | |
| PPQ | Quad Input Page Program | 4 + 256 | SPI | 32h 38h | A <23:16> | A <15:8> | A <7:0> | PD +256byte Quad | | | |
| SER | Sector Erase | 4 | SPI QPI | D7h 20h | A <23:16> | A <15:8> | A <7:0> | | | | |
| BER32 (32Kb) | Block Erase 32K | 4 | SPI QPI | 52h | A <23:16> | A <15:8> | A <7:0> | | | | |
| BER64 (64Kb) | Block Erase 64K | 4 | SPI QPI | D8h | A <23:16> | A <15:8> | A <7:0> | | | | |
| CER | Chip Erase | 1 | SPI QPI | C7h 60h | | | | | | | |
| WREN | Write Enable | 1 | SPI QPI | 06h | | | | | | | |
| WRDI | Write Disable | 1 | SPI QPI | 04h | | | | | | | |
| RDSR | Read Status Register | 2 | SPI QPI | 05h | SR | | | | | | |
| WRSR | Write Status Register | 2 | SPI QPI | 01h | WSR Data | | | | | | |
| RDFR | Read Function Register | 2 | SPI QPI | 48h | Data out | | | | | | |

| Instruction Name | Operation | Total Bytes | Mode | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Comments |
|---|---|---|---|---|---|---|---|---|---|---|---|
| WRFR | Write Function Register | 2 | SPI QPI | 42h | WFR Data | | | | | | |
| QIOEN | Enter Quad IO mode (QPI) | 1 | SPI | 35h | | | | | | | |
| QIODI | Exit Quad IO mode(QPI) | 1 | QPI | F5h | | | | | | | |
| PERSUS | Suspend during program/erase | 1 | SPI QPI | 75h B0h | | | | | | | |
| PERRSM | Resume program/erase | 1 | SPI QPI | 7Ah 30h | | | | | | | |
| Power Down | Power Down | 1 | SPI QPI | B9h | | | | | | | |
| RDID, RDPD | Read ID / Release Power Down | 4 | SPI QPI | ABh | Dummy Byte | Dummy Byte | Dummy Byte | ID7-ID0 | | | |
| SRP | Set Read Parameters | 4 | SPI QPI | C0h | Data in | | | | | | |
| RDJDID | Read JEDEC ID Command | 1 | | 9Fh | MID | ID15-8 | ID7-ID0 | | | | |
| RDMDID | Read Manufacturer & Device ID | 4 | SPI QPI | 90h | XXh | XXh | 00h | MID | ID7-ID0 | | |
| | | | | | | | 01h | DID | MID1 | | |
| RDIDQ | Read ID QPI mode | 4 | QPI | AFh | MID | ID15-8 | ID7-ID0 | | | | |
| RDUID | Read Unique ID | 4 | SPI QPI | 4Bh | XXh | XXh | 0Xh | Data Out | | | |
| RDSFDP | SFDP Read | 5 | SPI QPI | 5Ah | A <23:16> | A <15:8> | A <7:0> | Dummy Byte | Data out | | |
| RSTEN | Software Reset Enable | 1 | SPI QPI | 66h | | | | | | | |
| RST | Software Reset | 1 | SPI QPI | 99h | | | | | | | |
| RSTM | Mode Reset | 1 | SPI QPI | FFh | | | | | | | |
| IRER | Erase Information Row | 4 | SPI QPI | 64h | A <23:16> | A <15:8> | A <7:0> | | | | |
| IRP | Program Information Row | 4 + 256 | SPI QPI | 62h | A <23:16> | A <15:8> | A <7:0> | PD +256byte | | | |
| IRRD | Read Information Row | 4 | SPI QPI | 68h | A <23:16> | A <15:8> | A <7:0> | Data out | | | |

### 8.1 NORMAL READ OPERATION (NORD, 03h)

The Normal Read Data (NORD) instruction is used to read memory contents of the IS25LP032/064 at a maximum frequency of 50MHz.

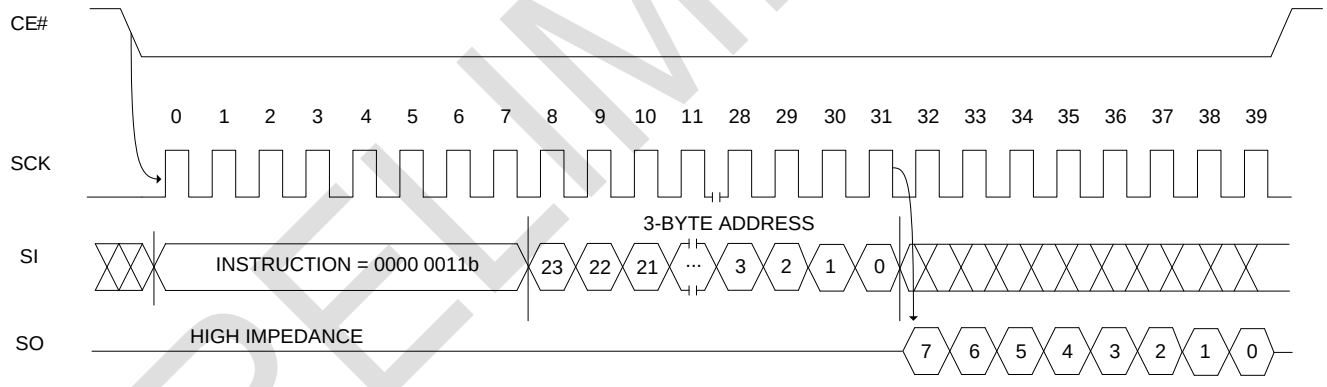
The NORD instruction code is transmitted via the SI line, followed by three address bytes (A23 - A0) of the first memory location to be read. A total of 24 address bits are shifted in, but only $A_{MSB}$ (most significant bit) - $A_0$ are decoded. The remaining bits (A23 – $A_{MSB}$) are ignored. The first byte address can be at any memory location. Upon completion, any data on the SI will be ignored. Refer to Table 8.2 for the related Address Key.

The first byte data (D7 - D0) address is shifted out on the SO line, MSB first. A single byte of data, or up to the whole memory array, can be read out in one READ instruction. The address is automatically incremented after each byte of data is shifted out. The read operation can be terminated at any time by driving CE# high (VIH) after the data comes out. When the highest address of the device is reached, the address counter will roll over to the 000000h address, allowing the entire memory to be read in one continuous READ instruction.
If a Read Data instruction is issued while an Erase, Program or Write cycle is in process (WIP=1) the instruction is ignored and will not have any effects on the current cycle.

**Table 8.2 Address Key**

| Address | IS25LP032/064 |
|---|---|
| $A_N$ ($A_{MSB}$ – $A_{0)}$ | A23 - A0 |

**Figure 8.1 Read Data Sequence**

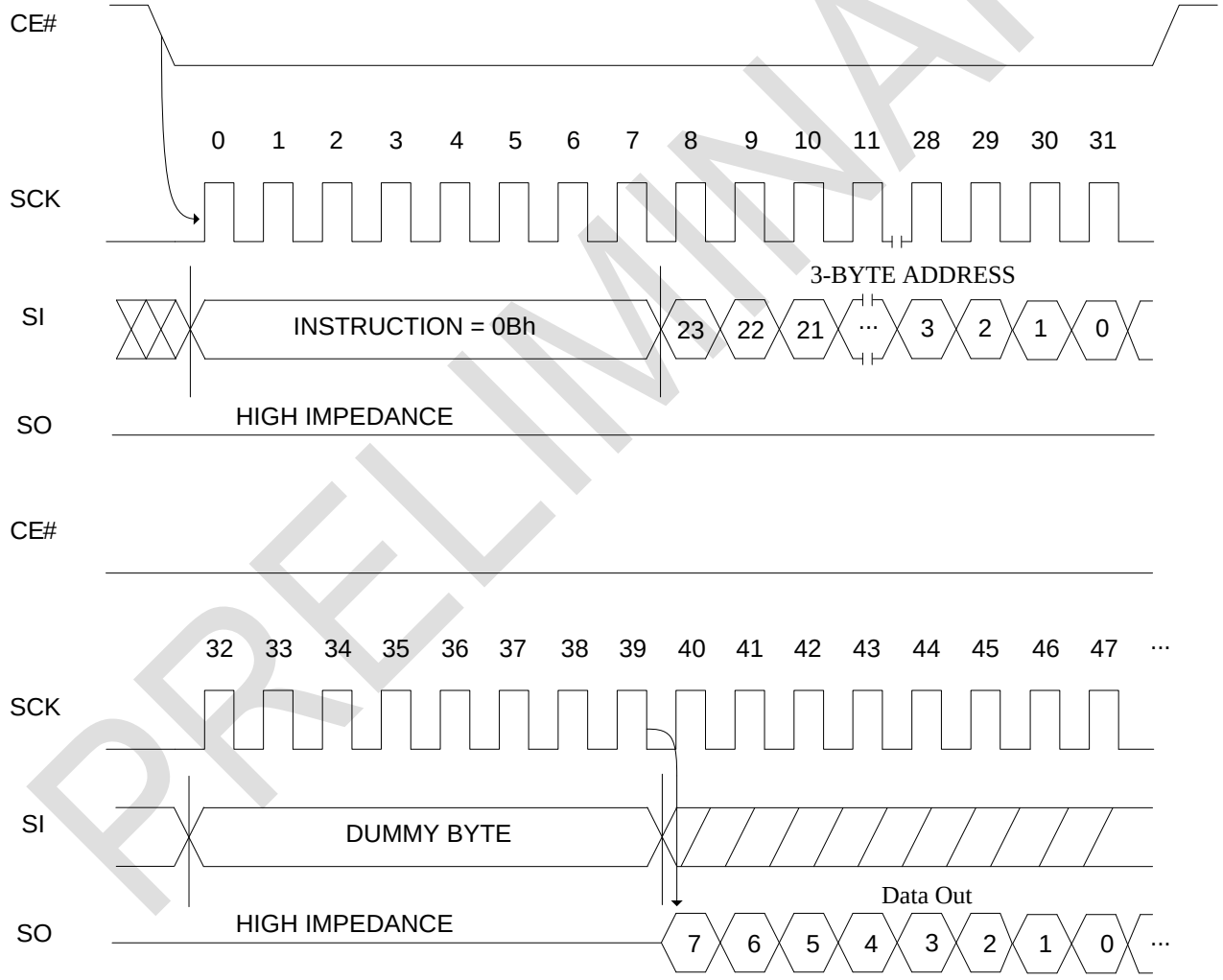


### 8.2 FAST READ DATA OPERATION (FRD, 0Bh)

The FAST READ (FRD) instruction is used to read memory data at up to a 133MHz clock.

The FAST READ instruction code is followed by three address bytes (A23 - A0) and a dummy byte (8 clocks), transmitted via the SI line, with each bit latched-in during the rising edge of SCK. Then the first data byte from the address is shifted out on the SO line, with each bit shifted out at a maximum frequency $f_{CT}$, during the falling edge of SCK.
The first byte addressed can be at any memory location. The address is automatically incremented after each byte of data is shifted out. When the highest address is reached, the address counter will roll over to the 000000h address, allowing the entire memory to be read with a single FAST READ instruction. The FAST READ instruction is terminated by driving CE# high (VIH). If a Fast Read Data instruction is issued while an Erase, Program or Write cycle is in process (WIP=1) the instruction is ignored and will not have any effects on the current cycle.

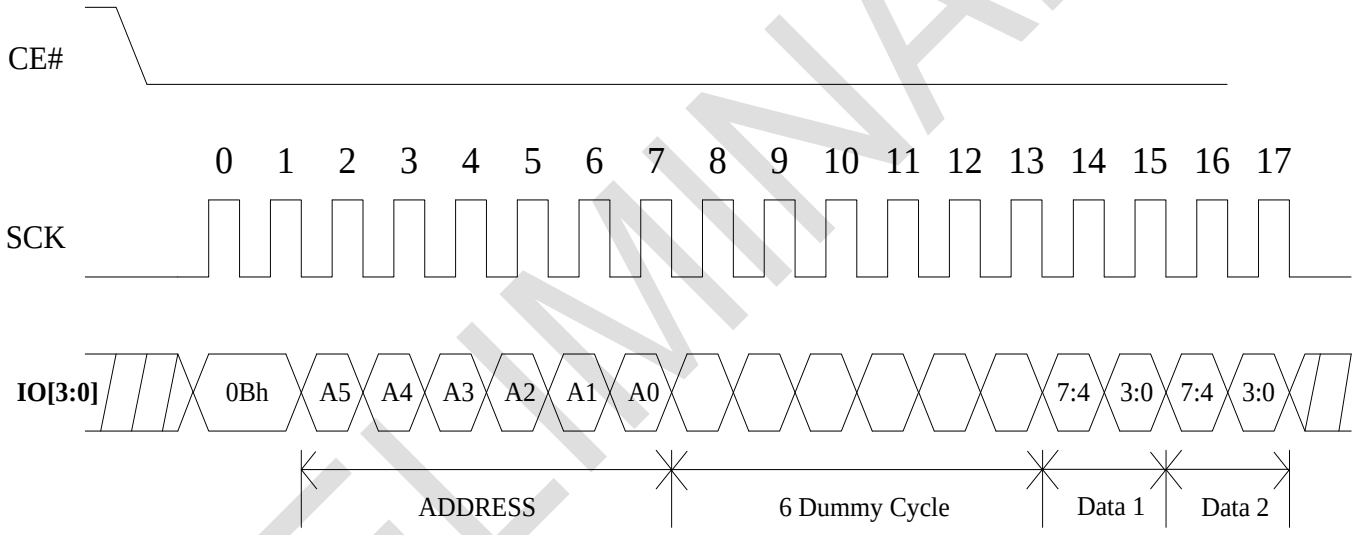
**Figure 8.2 Fast Read Data Sequence**



**FAST READ DATA QPI OPERATION (FRD QPI, 0Bh)**
The QPI FAST READ (FRD QPI) instruction is used to read memory data at up to a 133MHz clock.

The FAST READ instruction code (2 clocks) is followed by three address bytes (A23-A0—6clocks) and mode bits, dummy byte (4clocks), transmitted via the QPI line, with each bit latched-in during the rising edge of SCK. Then the first data byte addressed is shifted out on the SO line, with each bit shifted out at a maximum frequency $f_{CT}$, during the falling edge of SCK.
.
The first byte addressed can be at any memory location. The address is automatically incremented after each byte of data is shifted out. When the highest address is reached, the address counter will roll over to the 000000h address, allowing the entire memory to be read with a single FAST_READ instruction. The FAST_READ instruction is terminated by driving CE# high (VIH). If a Fast Read Data instruction is issued while an Erase, Program or Write cycle is in process (WIP=1) the instruction is ignored and will not have any effects on the current cycle

**Figure 8.3 Fast Read Data Sequence, QPI Mode**




**Note : Number of dummy cycles depends on Read Parameter setting. Detailed information in Table 6.9 Read Dummy Cycles.**

### 8.3 HOLD OPERATION

HOLD# is used in conjunction with CE# to select the IS25LP032/064. When the device is selected and a serial sequence is underway, HOLD# can be used to pause the serial communication with the master device without resetting the serial sequence. To pause, HOLD# is brought low while the SCK signal is low. To resume serial communication, HOLD# is brought high while the SCK signal is low (SCK may still toggle during HOLD). Inputs to SIO will be ignored while SO is in the high impedance state.

**Note : HOLD is not supported in DTR mode or with QE=1.**

Timing graph can be referenced in AC Parameters Figure 9.3

### 8.4 FAST READ DUAL I/O OPERATION (FRDIO, BBh)

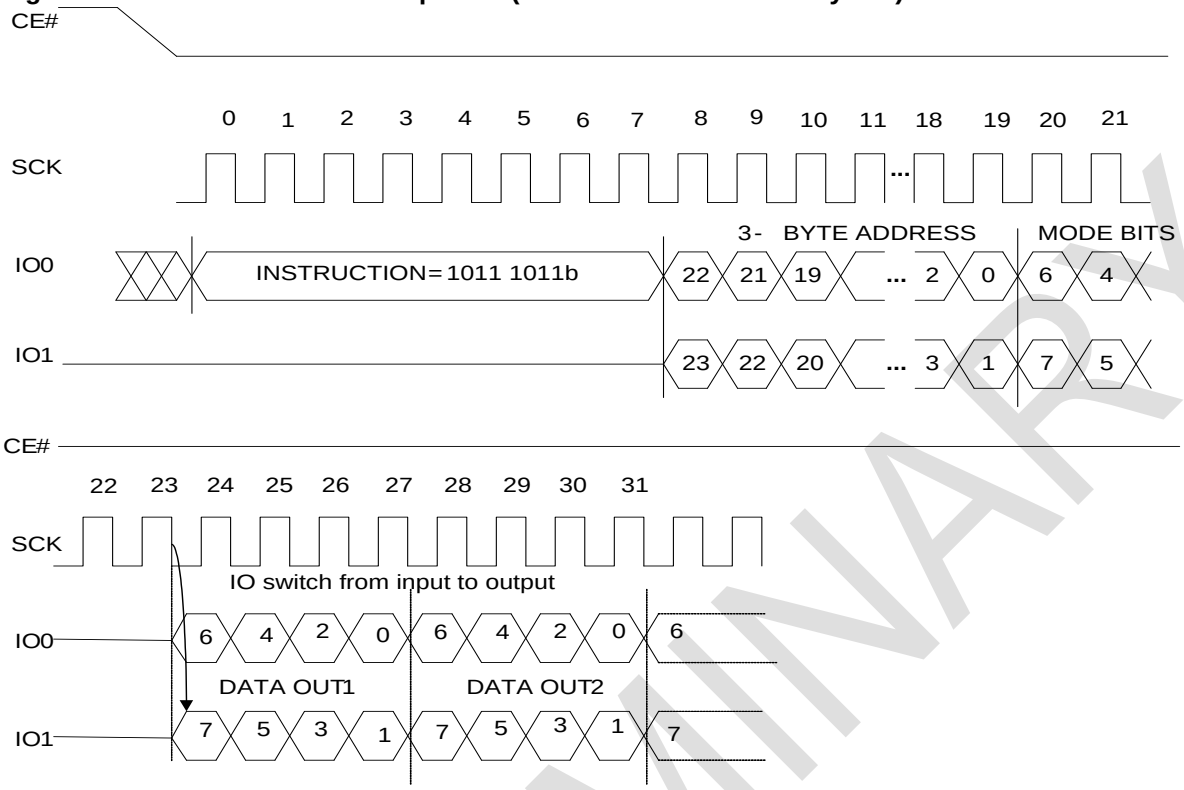
The FRDIO allows the address bits to be input two bits at a time. This may allow for code to be executed directly from the SPI in some applications.

The FRDIO instruction code is followed by three address bytes (A23 – A0) and a mode byte, transmitted via the IO0 and IO1 lines, with each pair of bits latched-in during the rising edge of SCK. The address MSB is input on IO1, the next bit on IO0, and continues to shift in alternating on the two lines. The mode bit contains the value Ax, where x is a “don’t care” value. Then the first data byte addressed is shifted out on the IO1 and IO0 lines, with each pair of bits shifted out at a maximum frequency $f_{CT}$, during the falling edge of SCK. The MSB is output on IO1, while simultaneously the second bit is output on IO0. Figure 8.4 illustrates the timing sequence.

The first byte addressed can be at any memory location. The address is automatically incremented after each byte of data is shifted out. When the highest address is reached, the address counter will roll over to the 000000h address, allowing the entire memory to be read with a single FRDIO instruction. FRDIO instruction is terminated by driving CE# high ($V_{IH}$).

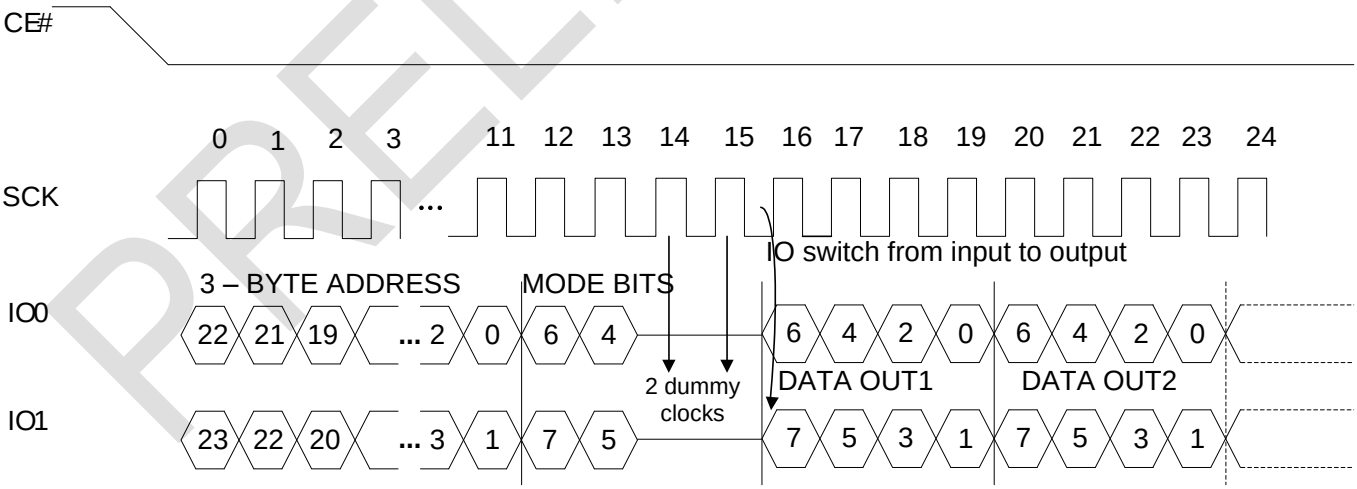
The device expects the next operation will be another FRDIO. It remains in this mode until it receives a Mode Reset (FFh) command. In subsequent FRDIO execution, the command code is skipped. It saves timing cycles as described in Figure 8.5. If a FRDIO instruction is issued while an Erase, Program or Write cycle is in process (WIP=1) the instruction is ignored and will not have any effects on the current cycle.

**Figure 8.4 Fast Read Dual I/O Sequence (with command decode cycles)**




**Note : Number of dummy cycles depends on Read Parameter setting. Detailed information in Table 6.9 Read Dummy Cycles.**

**Figure 8.5 Fast Read Dual I/O Sequence (without command decode cycles)**




**Notes : If the mode bits=Ax (x don't care), it can execute the continuous read mode (without command).**
**Number of dummy cycles depends on Read Parameter setting. Detailed information in Table 6.9 Read Dummy Cycles.**

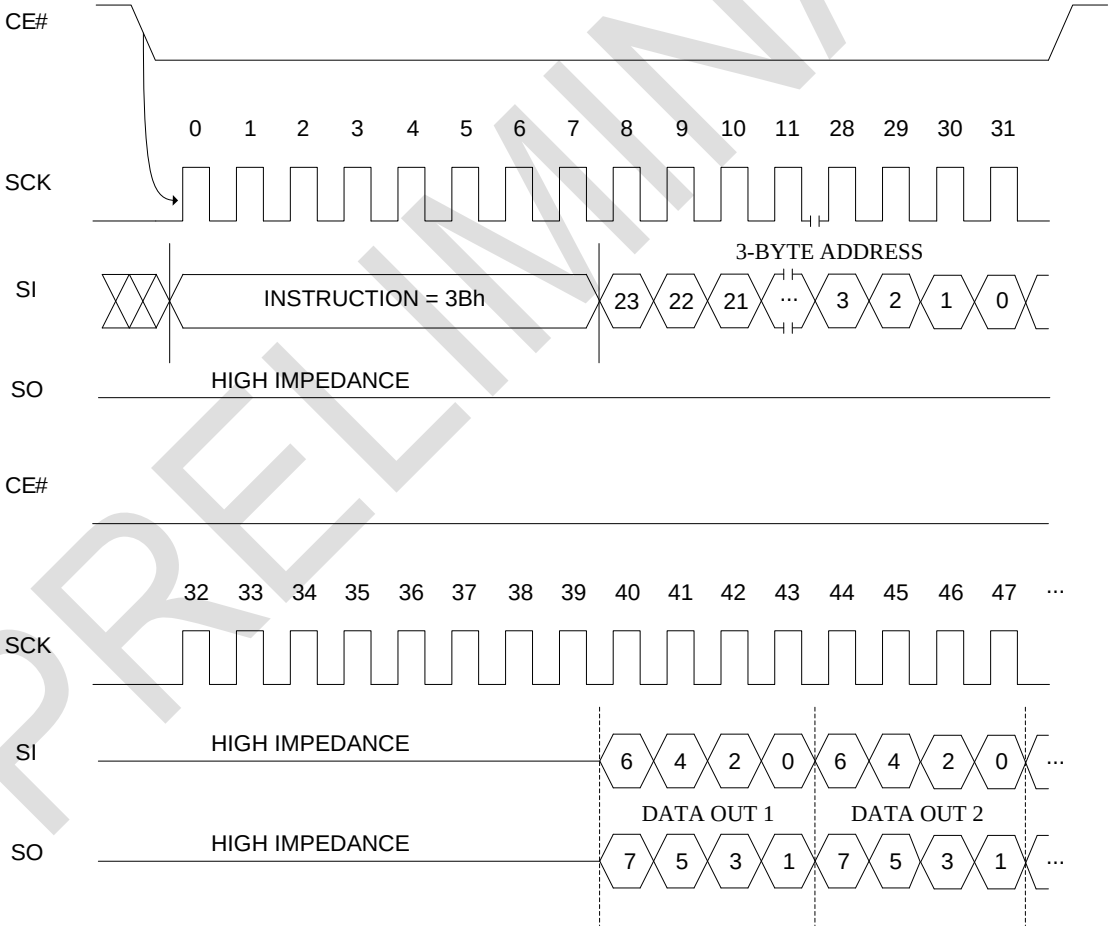
## 8.5 FAST READ DUAL OUTPUT OPERATION (FRDO, 3Bh)

The FRDO instruction is used to read memory data on two output pins each at up to a 133MHz clock.

The FRDO instruction code is followed by three address bytes (A23 – A0) and a dummy byte (8 clocks), transmitted via the SI line, with each bit
latched-in during the rising edge of SCK. Then the first data byte addressed is shifted out on the SO and SIO lines, with each pair of bits shifted out at a maximum frequency fCT, during the falling edge of SCK. The first bit (MSB) is output on SO, while simultaneously the second bit is output on SIO.

The first byte addressed can be at any memory location. The address is automatically incremented after each byte of data is shifted out. When the highest address is reached, the address counter will roll over to the 000000h address, allowing the entire memory to be read with a single FRDO instruction. FRDO instruction is terminated by driving CE# high (VIH). If a FRDO instruction is issued while an Erase, Program or Write cycle is in process (BUSY=1) the instruction is ignored and will not have any effects on the current cycle.

**Figure 8.6 Fast Read Dual-Output Sequence**




## 8.6 FAST READ QUAD I/O OPERATION (FRQIO, EBh)

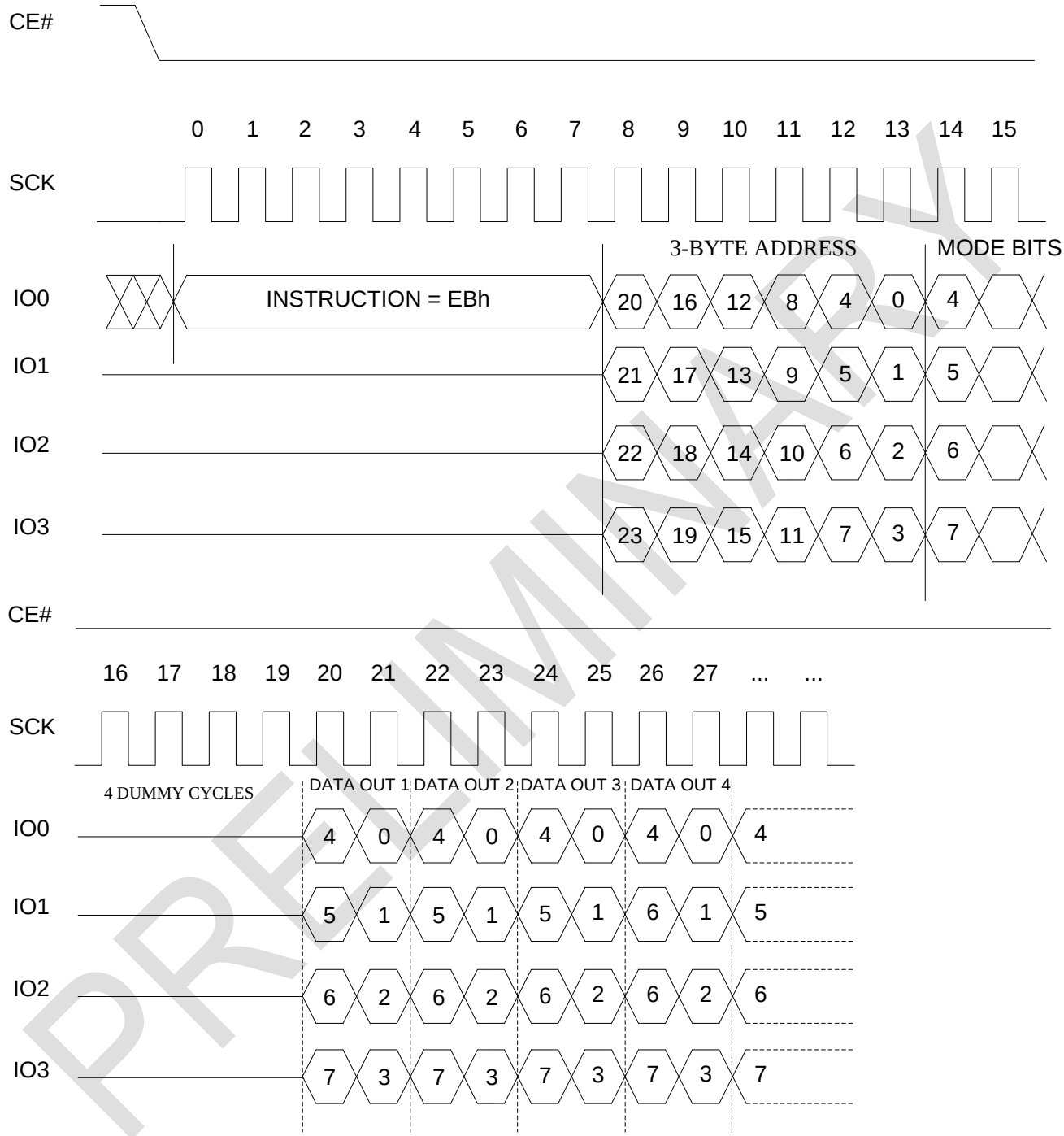
The FRQIO instruction allows the address bits to be input four bits at a time. This may allow for code to be executed directly from the SPI in some applications.

The FRQIO instruction code is followed by three address bytes (A23 – A0) and a mode byte, transmitted via the IO3, IO2, IO0 and IO1 lines, with each group of four bits latched-in during the rising edge of SCK. The address of MSB inputs on IO3, the next bit on IO2, the next bit on IO1, the next bit on IO0, and continue to shift in alternating on the four. The mode byte contains the value Ax, where x is a “don’t care” value. After four dummy clocks, the first data byte addressed is shifted out on the IO3, IO2, IO1 and IO0 lines, with each group of four bits shifted out at a maximum frequency $f_{CT}$, during the falling edge of SCK. The first bit (MSB) is output on IO3, while simultaneously the second bit is output on IO2, the third bit is output on IO1, etc. Figure 8.7 illustrates the timing sequence.

The first byte addressed can be at any memory location. The address is automatically incremented after each byte of data is shifted out. When the highest address is reached, the address counter will roll over to the 000000h address, allowing the entire memory to be read with a single FRQIO instruction. FRQIO instruction is terminated by driving CE# high ($V_{IH}$).

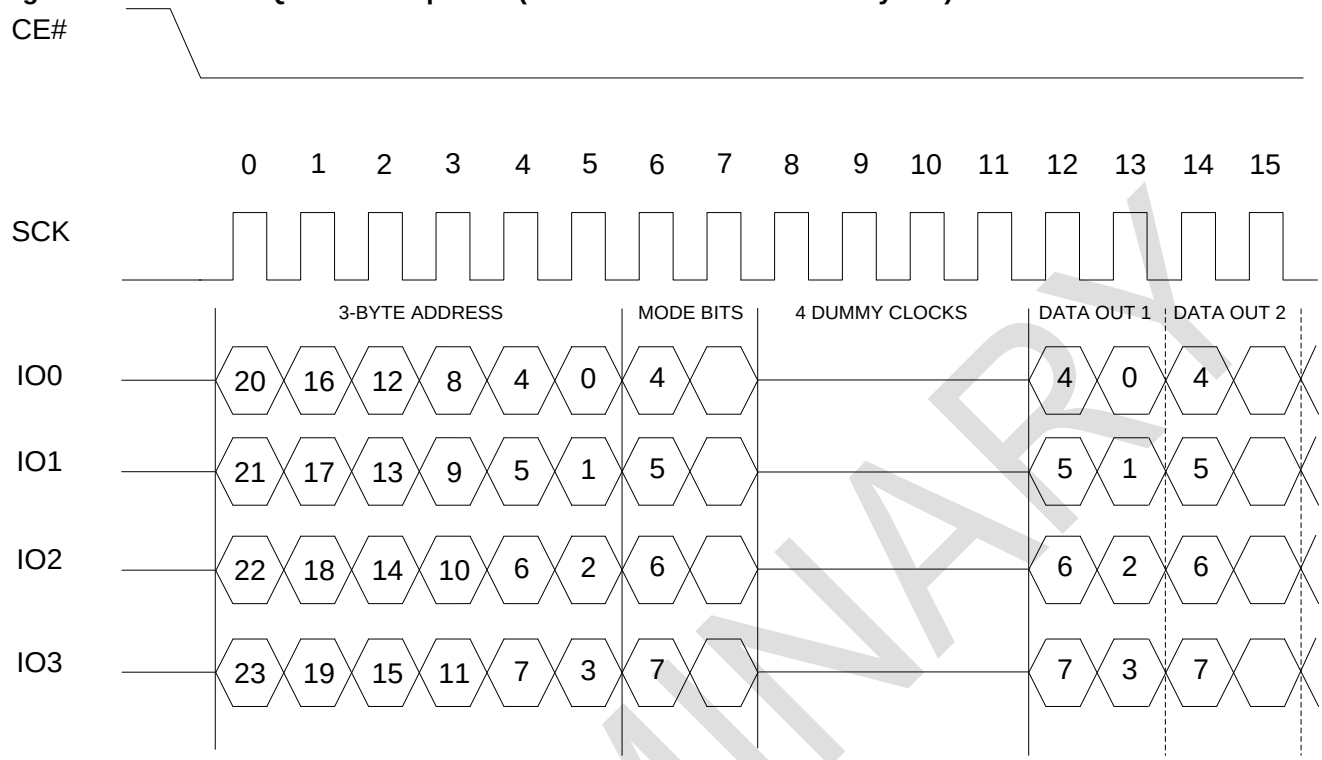
The device expects the next operation will be another FRQIO. It remains in this mode until it receives a Mode Reset (FFh) command. In subsequent FRDIO execution, the command code is not input, saving cycles as described in Figure 8.8. If a FRQIO instruction is issued while an Erase, Program or Write cycle is in process (WIP=1) the instruction is ignored and will not have any effects on the current cycle.

**Figure 8.7 Fast Read Quad I/O Sequence (with command decode cycles)**




**Note : Number of dummy cycles depends on clock speed. Detailed information in Table 6.9 Read Dummy Cycles**

**Figure 8.8 Fast Read Quad I/O Sequence (without command decode cycles)**




**Note : Number of dummy cycles depends on clock speed. Detailed information in Table 6.9 Read Dummy Cycles**

### 8.7 PAGE PROGRAM OPERATION (PP, 02h)

The Page Program (PP) instruction allows up to 256 bytes data to be programmed into memory in a single operation. The destination of the memory to be programmed must be outside the protected memory area set by the Block Protection (BP2, BP1, BP0) bits. A PP instruction which attempts to program into a page that is write-protected will be ignored. Before the execution of PP instruction, the Write Enable Latch (WEL) must be enabled through a Write Enable (WREN) instruction.

The PP instruction code, three address bytes and program data (1 to 256 bytes) are input via the SI line. Program operation will start immediately after the CE# is brought high, otherwise the PP instruction will not be executed. The internal control logic automatically handles the programming voltages and timing. During a program operation, all instructions will be ignored except the RDSR instruction. The progress or completion of the program operation can be determined by reading the WIP bit in Status Register via a RDSR instruction. If the WIP bit is “1”, the program operation is still in progress. If WIP bit is “0”, the program operation has completed.

If more than 256 bytes data are sent to a device, the address counter rolls over within the same page, the previously latched data are discarded, and the last 256 bytes are kept to be programmed into the page. The starting byte can be anywhere within the page. When the end of the page is reached, the address will wrap around to the beginning of the same page. If the data to be programmed are less than a full page, the data of all other bytes on the same page will remain unchanged.

**Note : A program operation can alter “1”s into “0”s, but an erase operation is required to change “0”s back to “1”s. A byte cannot be reprogrammed without first erasing the whole sector or block.**

**Figure 8.09 Page Program Sequence**

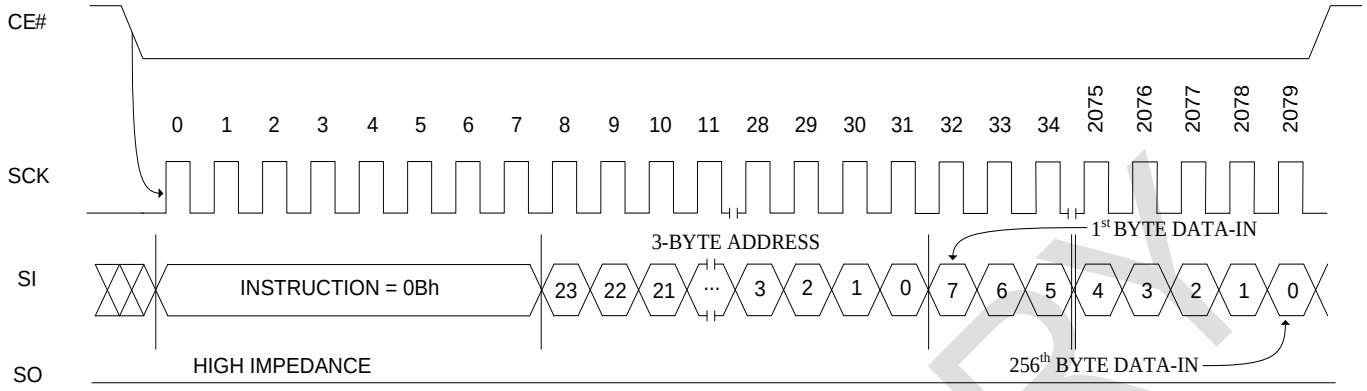
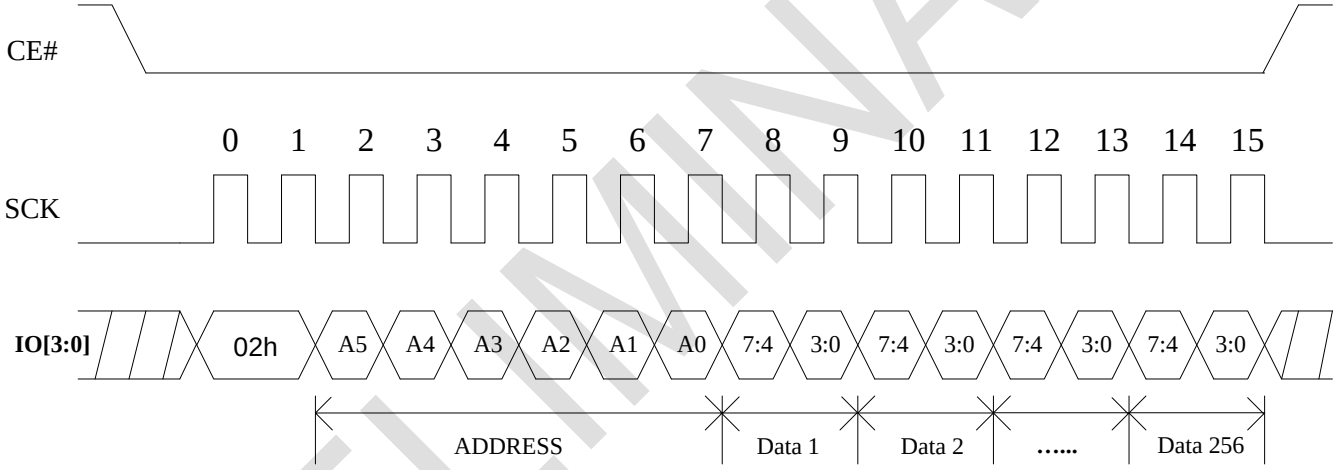



**Figure 8.10 Page Program Sequence (QPI)**



### 8.8 QUAD INPUT PAGE PROGRAM OPERATION (PPQ, 32h/38h)

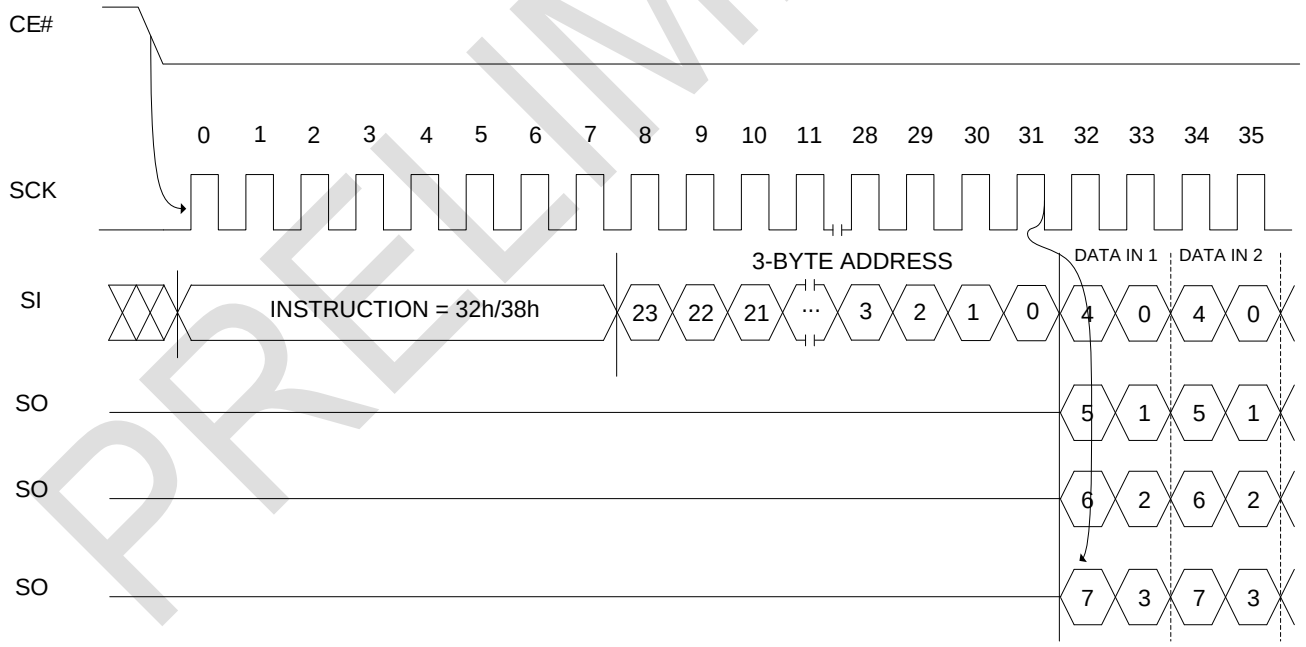
The Quad Input Page Program instruction allows up to 256 bytes data to be programmed into memory in a single operation with four pins (IO0, IO1, IO2 and IO3). The destination of the memory to be programmed must be outside the protected memory area set by the Block Protection (BP3, BP2, BP1, BP0) bits. A Quad Input Page Program instruction which attempts to program into a page that is write-protected will be ignored. Before the execution of Quad Input Page Program instruction, the QE bit in the status register must be set to "1" and the Write Enable Latch (WEL) must be enabled through a Write Enable (WREN) instruction.

The Quad Input Page Program instruction code, three address bytes and program data (1 to 256 bytes) are input via the four pins (IO0, IO1, IO2 and IO3). Program operation will start immediately after the CE# is brought high, otherwise the Quad Input Page Program instruction will not be executed. The internal control logic automatically handles the programming voltages and timing. During a program operation, all instructions will be ignored except the RDSR instruction. The progress or completion of the program operation can be determined by reading the WIP bit in Status Register via a RDSR instruction. If the WIP bit is "1", the program operation is still in progress. If WIP bit is "0", the program operation has completed.

If more than 256 bytes data are sent to a device, the address counter rolls over within the same page, the previously latched data are discarded, and the last 256 bytes data are kept to be programmed into the page. The starting byte can be anywhere within the page. When the end of the page is reached, the address will wrap around to the beginning of the same page. If the data to be programmed are less than a full page, the data of all other bytes on the same page will remain unchanged.

**Note : A program operation can alter "1"s into "0"s, but an erase operation is required to change "0"s back to "1"s. A byte cannot be reprogrammed without first erasing the whole sector or block.**

**Figure 8.11 Quad Input Page Program operation**



### 8.9 ERASE OPERATION

The memory array of the IS25LP032/064 is organized into uniform 4 KByte sectors or 32K/64 KByte uniform blocks (a block consists of sixteen adjacent sectors).

Before a byte is reprogrammed, the sector or block that contains the byte must be erased (erasing sets bits to "1"). In order to erase the device, there are three erase instructions available: Sector Erase (SER), Block Erase (BER) and Chip Erase (CER). A sector erase operation allows any individual sector to be erased without affecting the data in other sectors. A block erase operation erases any individual block. A chip erase operation erases the whole memory array of a device. A sector erase, block erase or chip erase operation can be executed prior to any programming operation.

### 8.10 SECTOR ERASE OPERATION (SER, D7h/20h)

A Sector Erase (SER) instruction erases a 4 KByte sector Before the execution of a SER instruction, the Write Enable Latch (WEL) must be set via a Write Enable (WREN) instruction. The WEL bit is reset automatically after the completion of sector an erase operation.

A SER instruction is entered, after CE# is pulled low to select the device and stays low during the entire instruction sequence The SER instruction code, and three address bytes are input via SI. Erase operation will start immediately after CE# is pulled high. The internal control logic automatically handles the erase voltage and timing. Refer to Figure 8.12-8.13 for the Sector Erase Sequence.

During an erase operation, all instruction will be ignored except the Read Status Register (RDSR) instruction. The progress or completion of the erase operation can be determined by reading the WIP bit in the Status Register using a RDSR instruction. If the WIP bit is "1", the erase operation is still in progress. If the WIP bit is "0", the erase operation has been completed.

**Figure 8.12 Sector Erase Sequence**

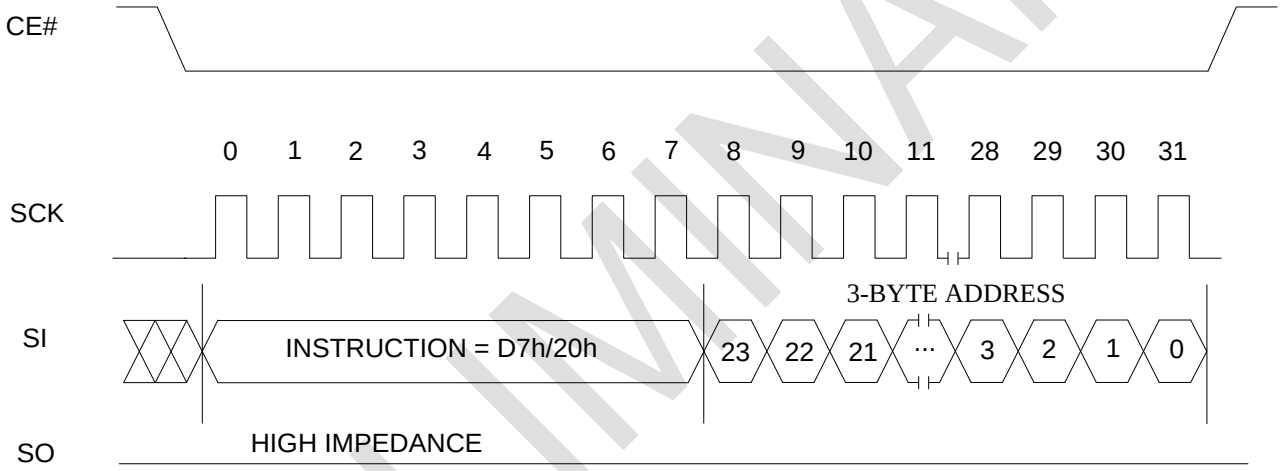
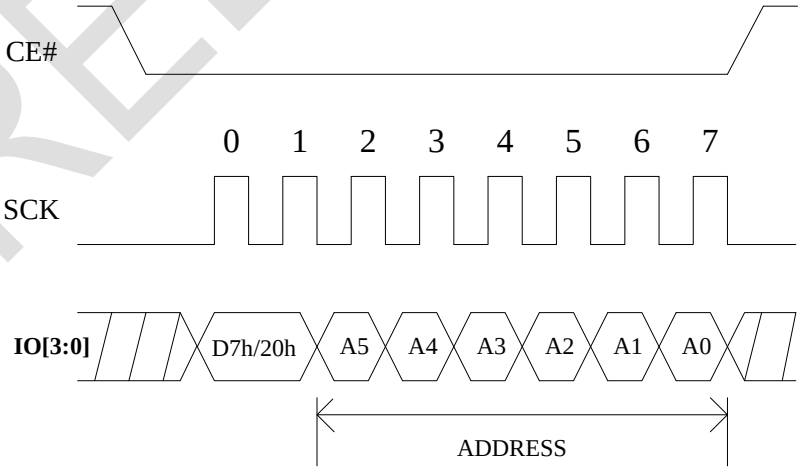



**Figure 8.13 Sector Erase Sequence (QPI)**



### 8.11 BLOCK ERASE OPERATION (BER32K:52h, BER64K:D8h)

A Block Erase (BER) instruction erases a 32/64 KByte block of the IS25LP032/064. Before the execution of a BER instruction, the Write Enable Latch (WEL) must be set via a Write Enable (WREN) instruction. The WEL is reset automatically after the completion of a block erase operation.

The BER instruction code and three address bytes are input via SI. Erase operation will start immediately after the CE# is pulled high, otherwise the BER instruction will not be executed. The internal control logic automatically handles the erase voltage and timing. Refer to Figure 8.14-8.17 for the Block Erase Sequence.

**Figure 8.14 Block Erase(64k) Sequence**

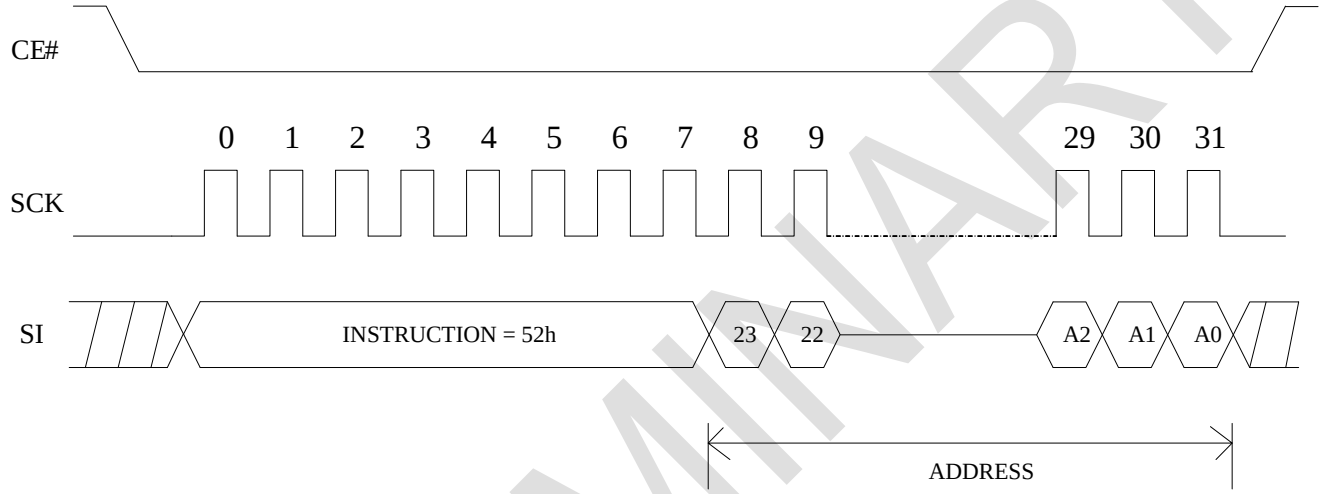



**Figure 8.15 Block Erase(64k) Sequence (QPI)**

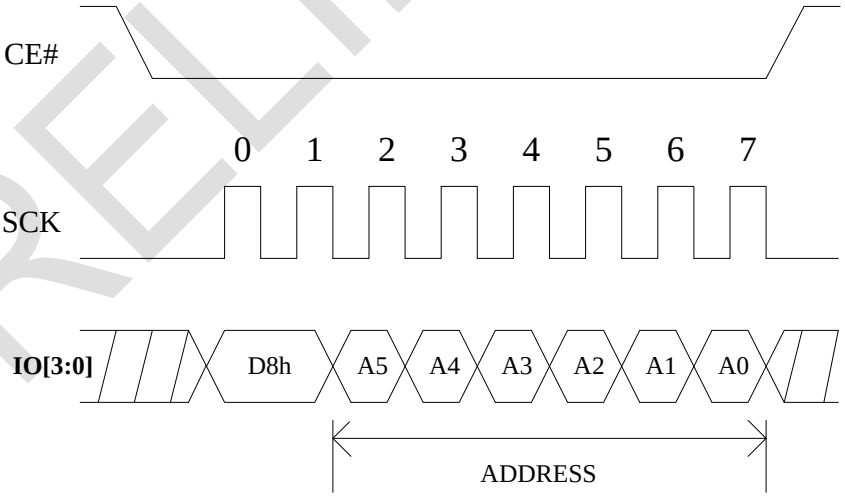


**Figure 8.16 Block Erase Sequence (32K)**

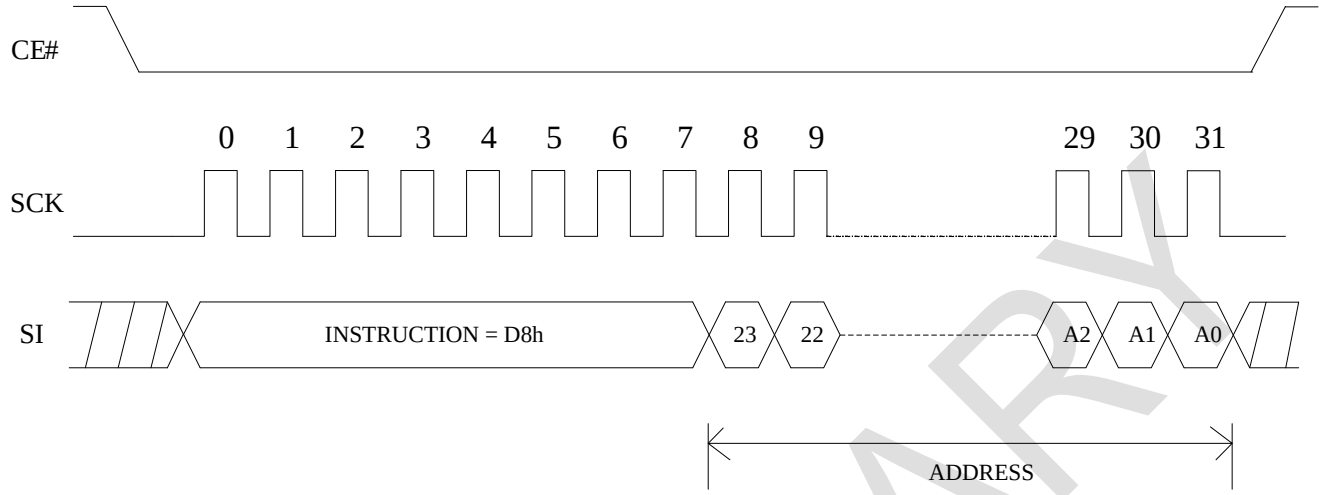
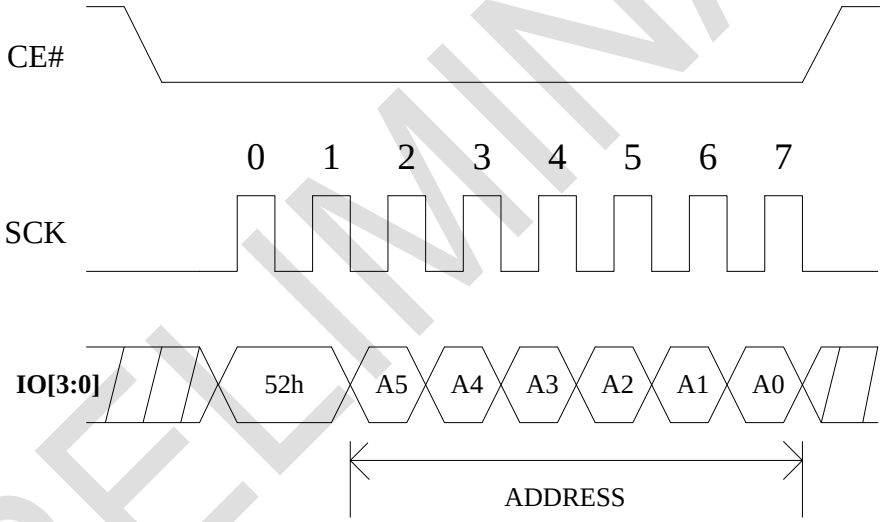



**Figure 8.17 Block Erase (32K) Sequence (QPI)**



### 8.12 CHIP ERASE OPERATION (CER, C7h/60h)

A Chip Erase (CER) instruction erases the entire memory array of a IS25LP032/064. Before the execution of CER instruction, the Write Enable Latch (WEL) must be set via a Write Enable (WREN) instruction. The WEL is reset automatically after completion of a chip erase operation.

The CER instruction code is input via the SI. Erase operation will start immediately after CE# is pulled high, otherwise the CER instruction will not be executed. The internal control logic automatically handles the erase voltage and timing. Refer to Figure 8.18-8.19 for the Chip Erase Sequence.

**Figure 8.18 Chip Erase Sequence**

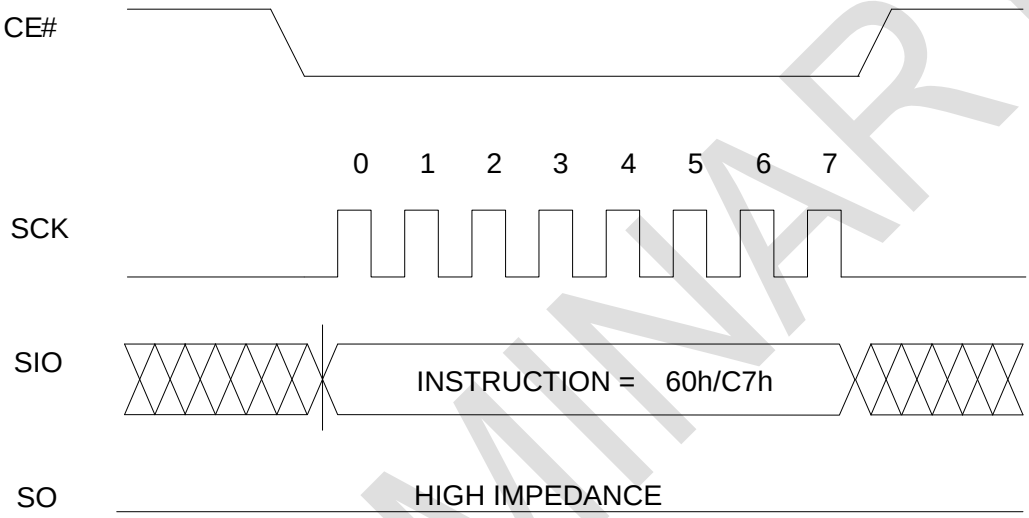
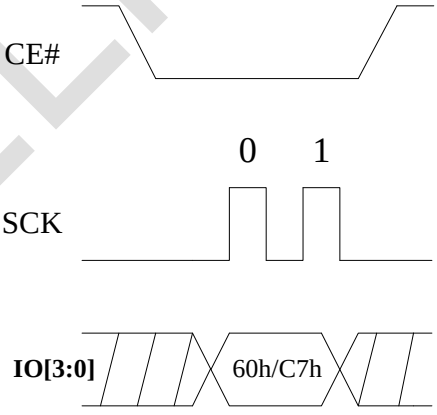



**Figure 8.19 Chip Erase Sequence (QPI)**



### 8.13 WRITE ENABLE OPERATION (WREN, 06h)

The Write Enable (WREN) instruction is used to set the Write Enable Latch (WEL) bit. The WEL bit of the IS25LP032/064 is reset to the write –protected state after power-up. The WEL bit must be write enabled before any write operation, including sector, block erase, chip erase, page program, write status register, and write configuration register operations. The WEL bit will be reset to the write-protect state automatically upon completion of a write operation. The WREN instruction is required before any above operation is executed.

**Figure 8.20 Write Enable Sequence**

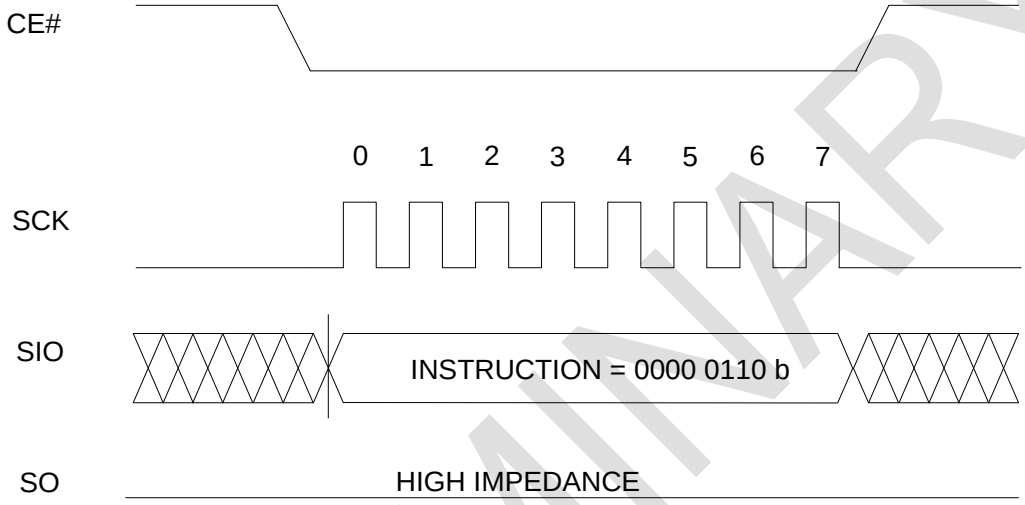
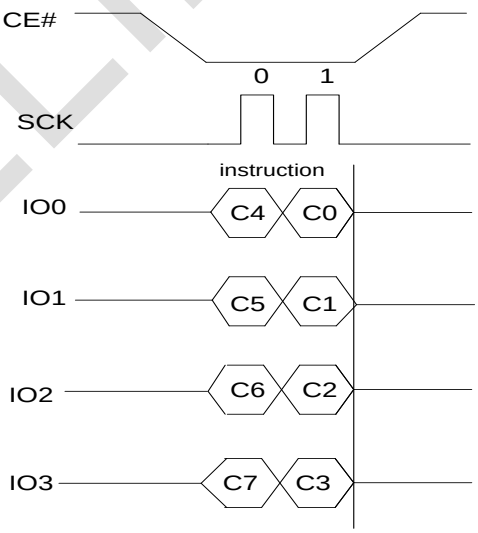



**Figure 8.21 WRITE ENABLE OPERATION (QPI)**



### 8.14 WRITE DISABLE OPERATION (WRDI, 04h)

The Write Disable (WRDI) instruction resets the WEL bit and disables all write instructions. The WRDI instruction is not required after the execution of a write instruction, since the WEL bit is automatically reset.

**Figure 8.22 Write Disable Sequence**

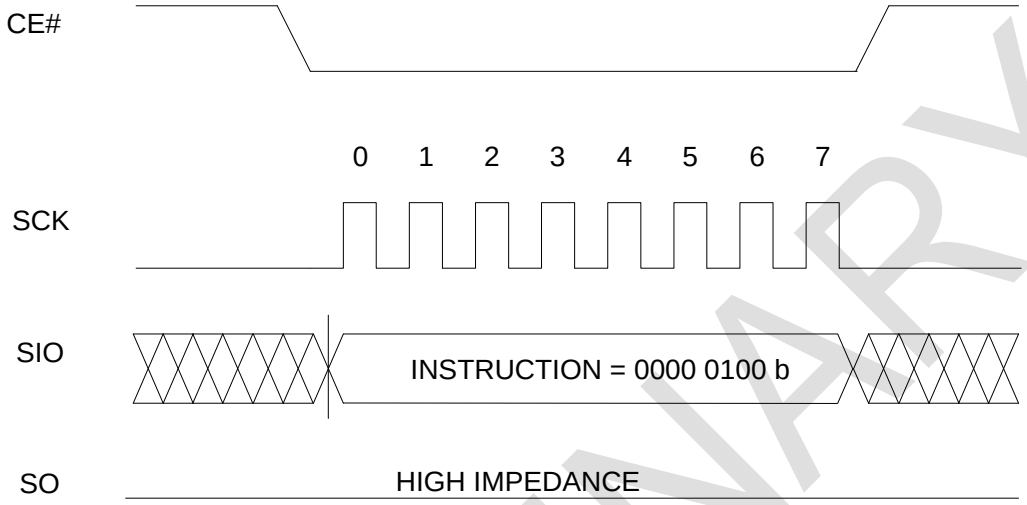
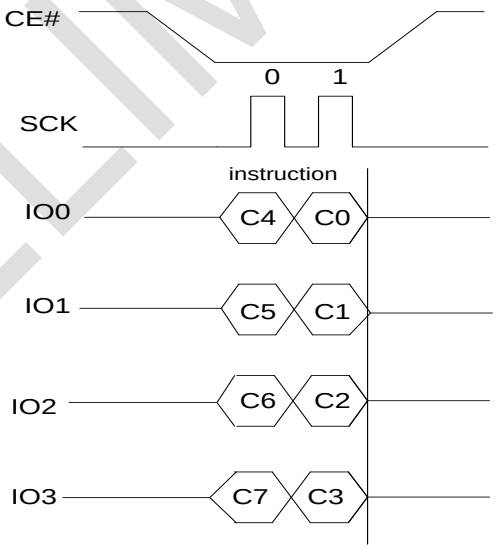



**Figure 8.23 WRITE DISABLE OPERATION (QPI)**



### 8.15 READ STATUS REGISTER OPERATION (RDSR, 05h)

The Read Status Register (RDSR) instruction provides access to the Status Register. During the execution of a program, erase or write status register operation, all other instructions will be ignored except the RDSR instruction, which can be used to check the progress or completion of an operation by reading the WIP bit of Status Register.

**Figure 8.24 Read Status Register Sequence**

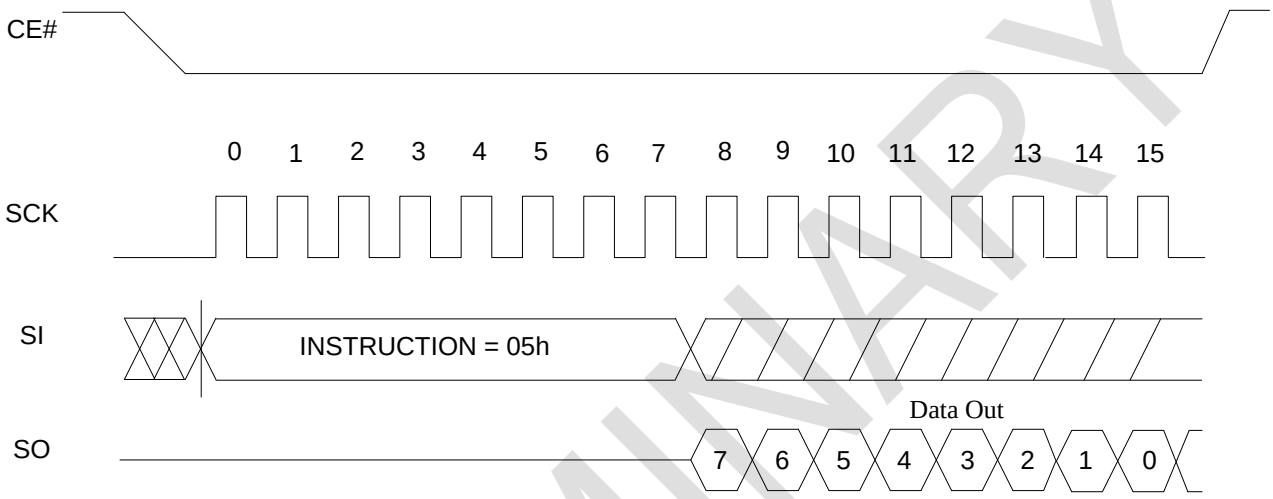
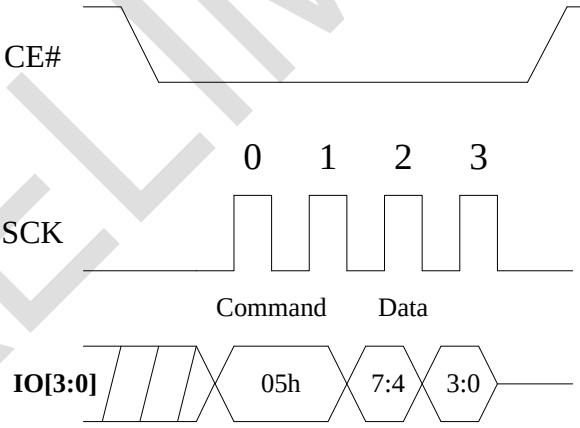



**Figure 8.256 RDSR COMMAND (READ STATUS REGISTER) OPERATION (QPI)**



### 8.16 WRITE STATUS REGISTER OPERATION (WRSR, 01h)

The Write Status Register (WRSR) instruction allows the user to enable or disable the block protection and status register write protection features by writing “0”s or “1”s into the non-volatile BP3, BP2, BP1, BP0, QE and SRWD bits.

**Figure 8.26 Write Status Register Sequence**

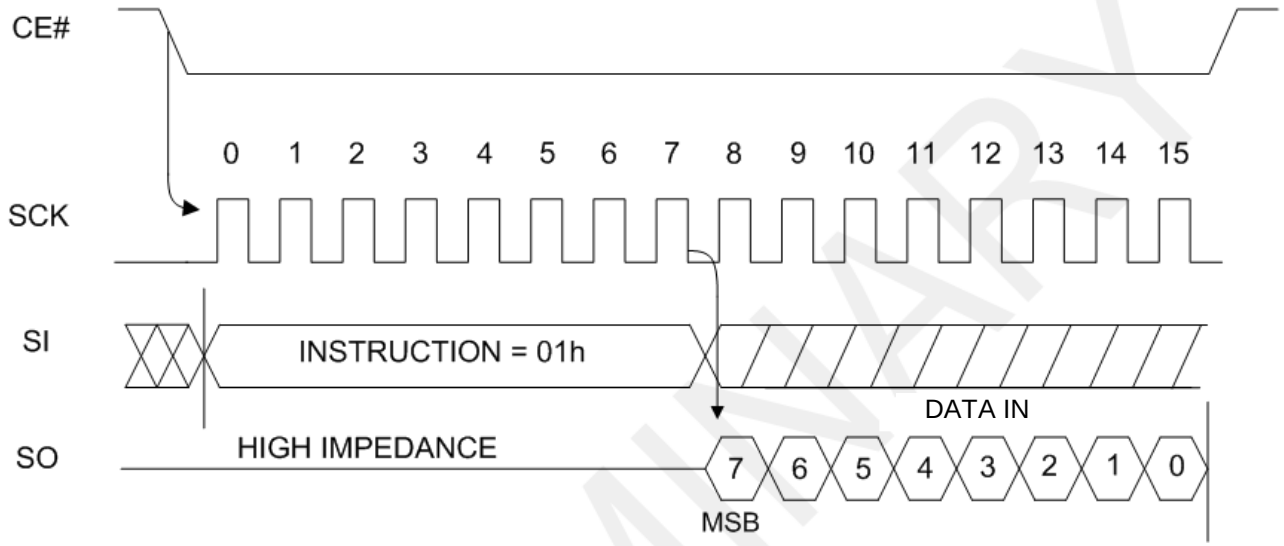
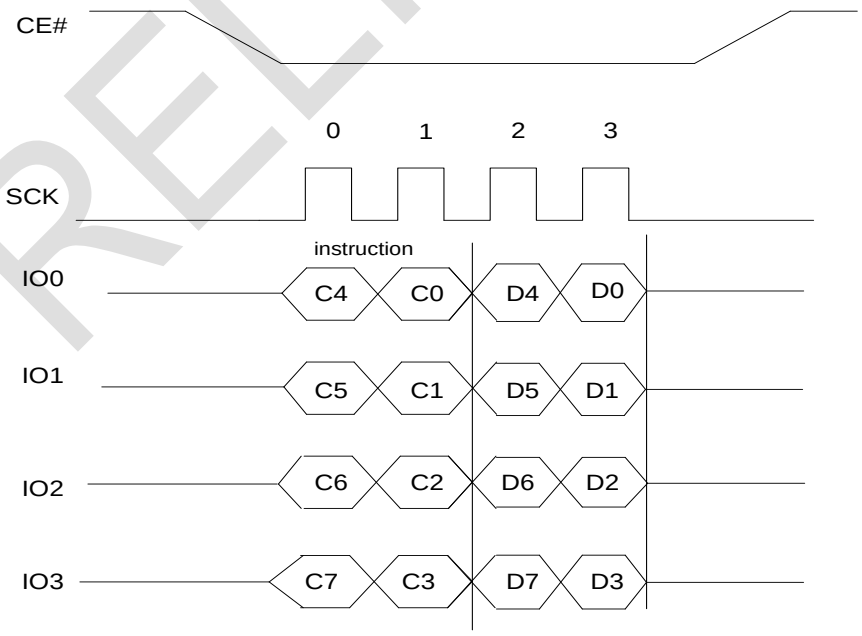



**Figure 8.27 WRSR COMMAND (WRITE STATUS REGISTER) OPERATION (QPI)**



### 8.17 READ FUNCTION REGISTER OPERATION (RDFR, 48h)

The Read Function Register (RDFR) instruction provides access to the Erase/Program suspend register. During the execution of a program, erase or write status register suspend, which can be used to check the suspend status.

**Figure 8.28 Read Function Register Sequence**

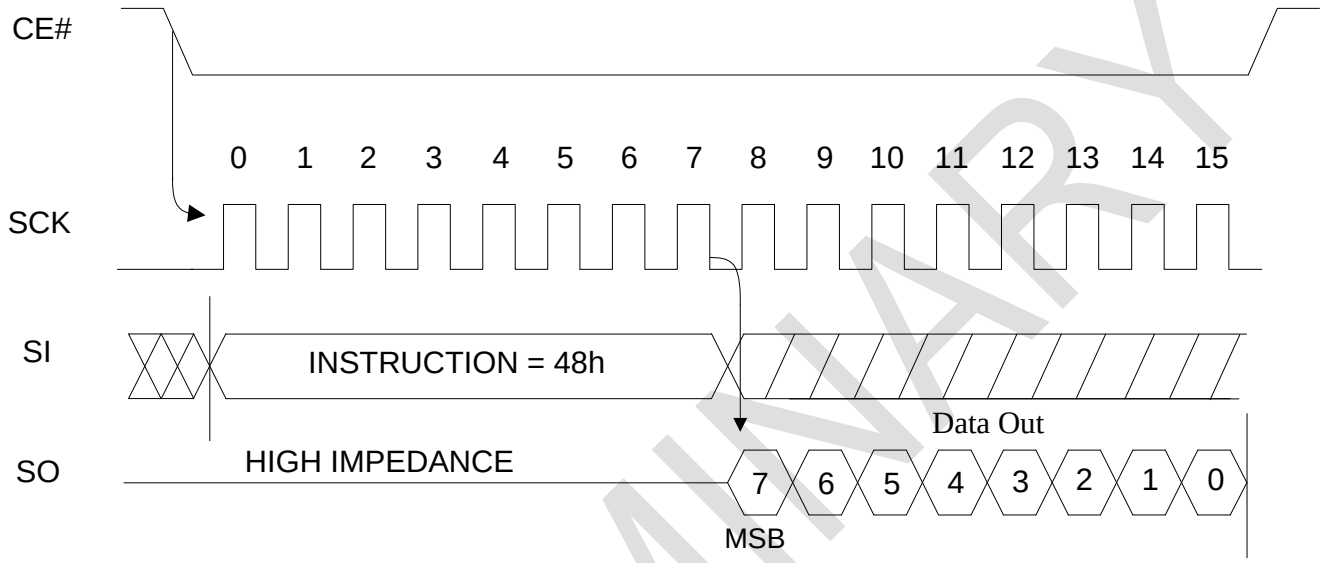
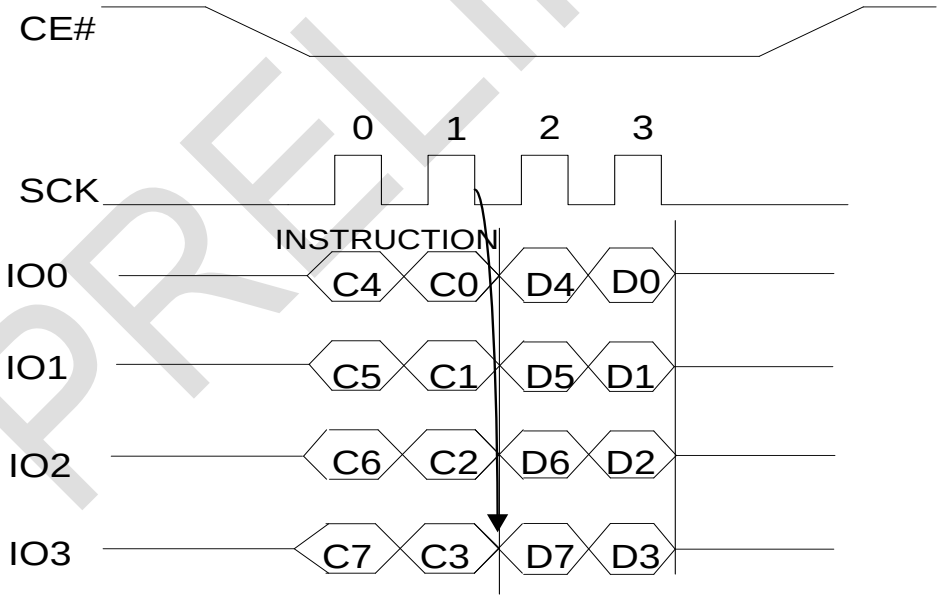



**Figure 8.29 READ FUNCTION REGISTER OPERATION (QPI) RDFR**



### 8.18 WRITE FUNCTION REGISTER OPERATION (WRFR, 42h)

The Write Function Register (WRFR) instruction allows the user to lock the information row by bit 0. (IR lock)

**Figure 8.30 Write Function Register Sequence**

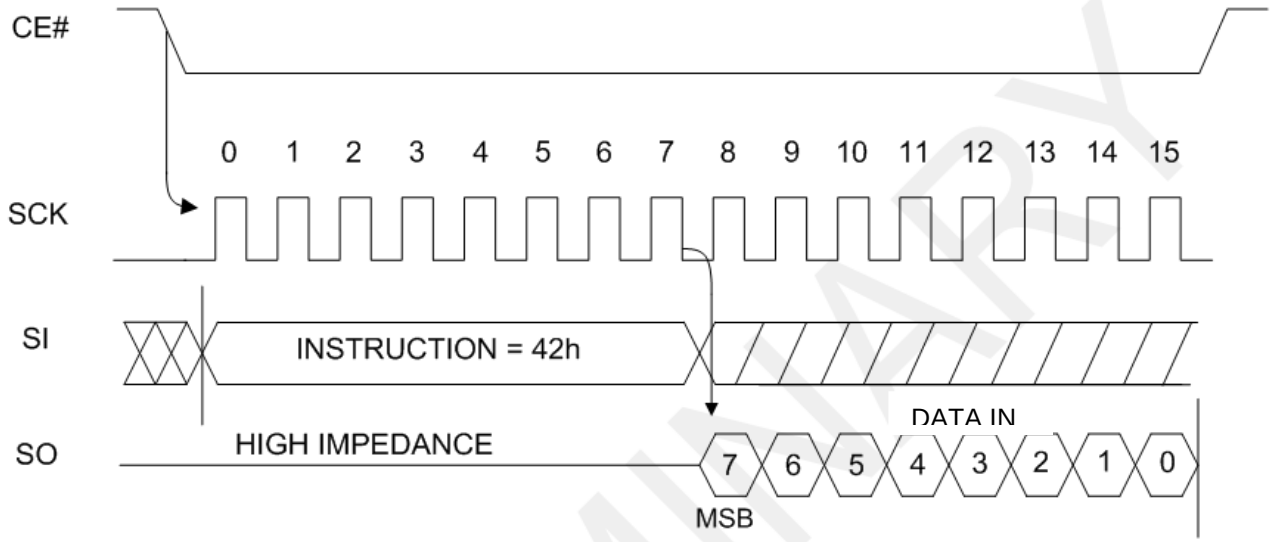
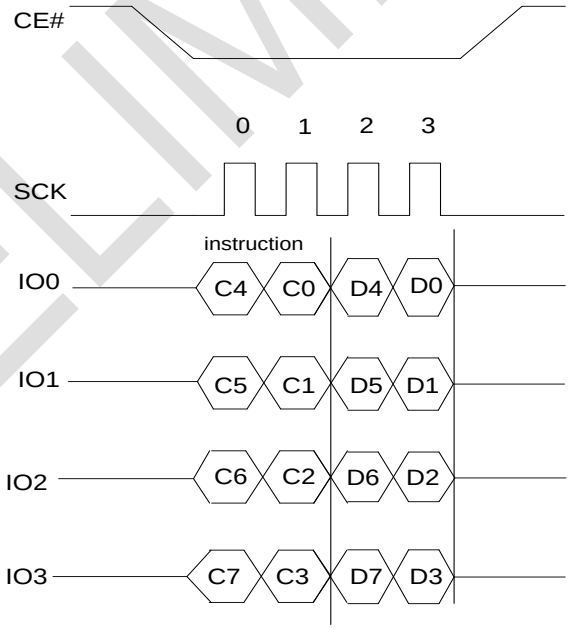



**Figure 8.31 WRFR COMMAND (WRITE Function REGISTER) OPERATION (QPI)**



### 8.19 ENTER QUAD PERIPHERAL INTERFACE (QPI) MODE OPERATION (QIOEN,35h; QIODI,F5h)

The Enter Quad I/O (ENQIO) instruction, 35h, enables the flash device for QPI bus operation. Upon completion of the instruction, all instructions thereafter will be 4-bit multiplexed input/output until a power cycle or a “Exit Quad I/O instruction” instruction.

**Figure 8.32 Enter Quad Peripheral Interface OPERATION (QPI)**

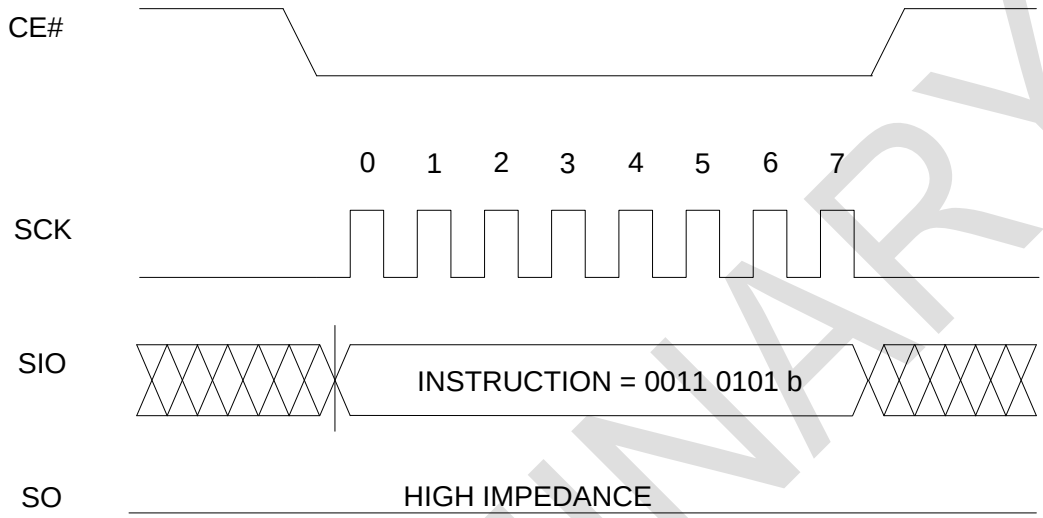
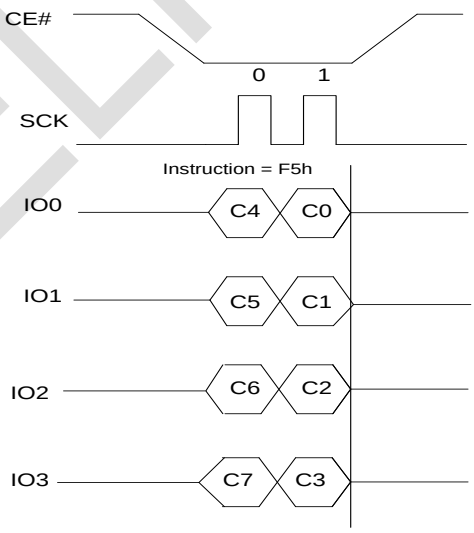



**Figure 8.33 Exit Quad Peripheral Interface (QPI) mode OPERATION**

The Exit Quad I/O instruction, F5h, resets the device to 1-bit SPI protocol operation. To execute a Exit Quad I/O operation, the host drives CE# low, sends the Exit Quad I/O command cycle then, drives CE# high. The device just accepts SQI (2 clocks) command cycles.



### 8.20 PROGRAM/ERASE SUSPEND & RESUME

The device allows the interruption of Sector-Erase, Block-Erase or Page-Program operations to conduct other operations. B0h command for suspend and 30h for resume will be used. (SPI/QPI all acceptable) Function register bit2 (PSUS) and bit3 (ESUS) are used to check whether or not the device is in suspend mode.

Suspend to read ready timing: 100µs.
Resume to another suspend timing: 400µs (recommendation).

**PROGRAM/ERASE SUSPEND DURING SECTOR-ERASE OR BLOCK-Erase (PERSUS 75h/B0h)**
After erase suspend, WEL bit will be disabled, therefore only read related, resume and reset commands will be accepted (03h, 0Bh, BBh, EBh, 05h, ABh, 30h, 9Fh, ABh, 90h, 4Bh, 00h, 66h, 99h, AFh, C0h).
To execute a Program/Erase Suspend operation, the host drives CE# low, sends the Program/Erase Suspend command cycle (B0H), then drives CE# high. The Function register indicates that the erase has been suspended by changing the ESUS bit from '0' to '1,' but the device will not accept another command until it is ready. To determine when the device will accept a new command, poll the WIP bit in the Status register or wait $t_{SUS}$. When ESUS bit is issued, the Write Enable Latch (WEL) bit will be reset.

**PROGRAM/ERASE SUSPEND DURING PAGE PROGRAMMING (PERSUS 75h/B0h)**
Program suspend allows the interruption of all program operations.
After a program suspend command, WEL bit will be disabled, only read related, resume and reset command can be accepted (03h, 0Bh, BBh, EBh, 05h, ABh, 30h, 9Fh, ABh, 90h, 4Bh, 00h, 66h, 99h, AFh, C0h).
To execute a Program/Erase Suspend operation, the host drives CE# low, sends the Program/Erase Suspend command cycle (B0H), then drives CE# high. The Function register indicates that the programming has been suspended by changing the PSUS bit from '0' to '1,' but the device will not accept another command until it is ready. To determine when the device will accept a new command, poll the WIP bit in the Status register or wait $t_{SUS}$.

**PROGRAM/ERASE RESUME (PERRSM 7A/30h)**
Program/Erase Resume restarts a Program/Erase command that was suspended, and changes the suspend status bit in the (ESUS or PSUS bits) back to '0'. To execute a Program/Erase Resume operation, the host drives CE# low, sends the Program/Erase Resume command cycle (30H), then drives CE# high. A cycle is two nibbles long, most significant nibble first. To determine if the internal, self-timed Write operation completed, poll the WIP bit in the Status register, or wait the specified time $t_{SE}$, $t_{BE}$ or $t_{PP}$ for Sector- Erase, Block-Erase, or Page-Programming, respectively. The total write time before suspend and after resume will not exceed the uninterrupted write times $t_{SE}$, $t_{BE}$ or $t_{PP}$.

### 8.21 DEEP POWER DOWN (DP, B9h)

The Deep Power-down (DP) instruction is for setting the device on the minimizing the power consumption (enter into Power-Down mode), the standby current is reduced from $I_{sb1}$ to $I_{sb2}$). During the Power-down mode, the device is not active and all Write/Program/Erase instructions are ignored. The instruction is initiated by driving the CE# pin low and shifting the instruction code "B9h" as show in the figure 8.34. The CE# pin must be driven high after the instruction has been latched. If this is not done the Power-Down will not be executed. After CE# pin driven high, the power-down state will entered within the time duration of $t_{DP}$. While in the power-down state only the Release from Power-down / RDID instruction, which restores the device to normal operation, will be recognized. All other instructions are ignored. This includes the Read Status Register instruction, which is always available during normal operation. Ignoring all but one instruction makes the Power Down state a useful condition for securing maximum write protection. It can support in SPI and QPI mode.

**Figure 8.34 Enter Deep Power Down Mode Operation. (SPI)**

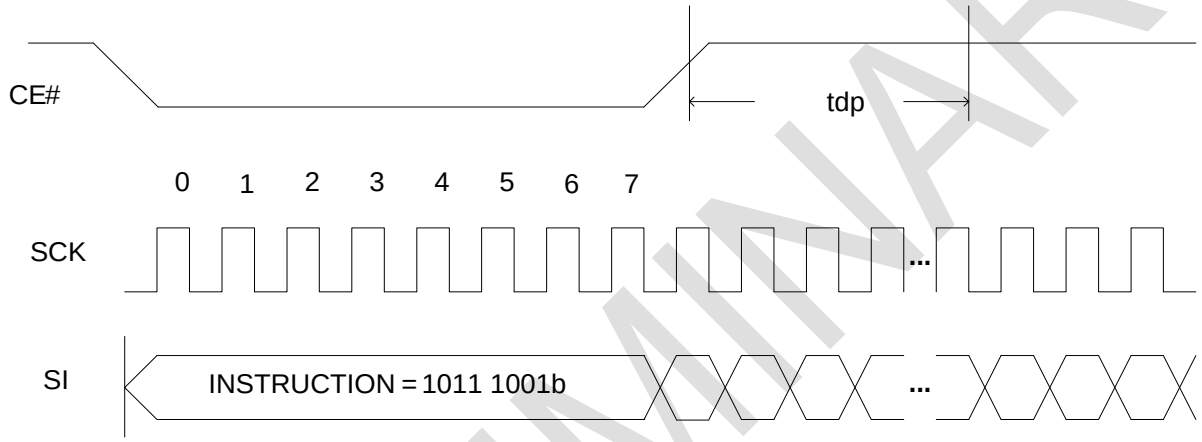
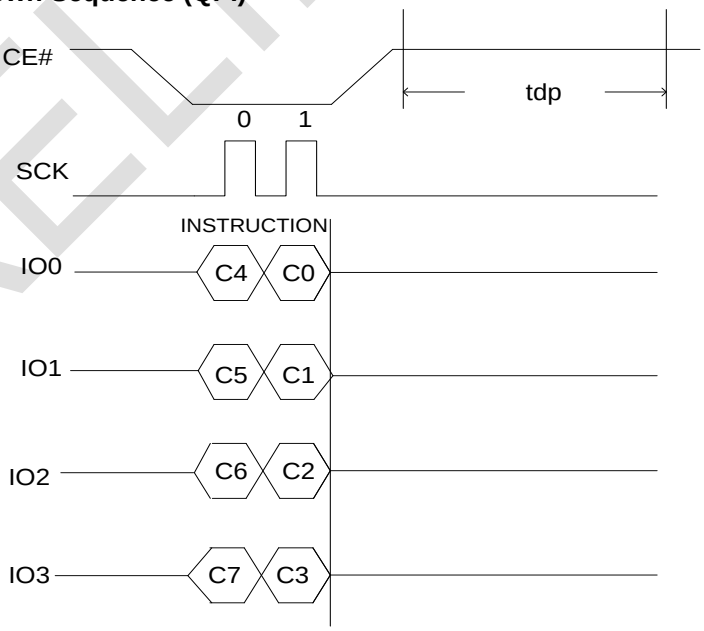



**Figure 8.35 Deep Power Down Sequence (QPI)**



### 8.22 RELEASE DEEP POWER DOWN (RDPD, ABh)

The Release from Power-down /read Device ID instruction is a multi-purpose instruction. To release the device from the power-down state Mode, the instruction is issued by driving the CE# pin low, shifting the instruction code “ABh” and driving CE# high as shown in Figure 8.36, 8.37
Release from power-down will take the time duration of $t_{RES1}$ before the device will resume normal operation and other instructions are accepted. The CE# pin must remain high during the $t_{RES1}$ time duration. If the Release from Power-down / RDID instruction is issued while an Erase, Program or Write cycle is in process (when WIP equals 1) the instruction is ignored and will not have any effects on the current cycle.

**Figure 8.36 Release Power Down Sequence (SPI)**

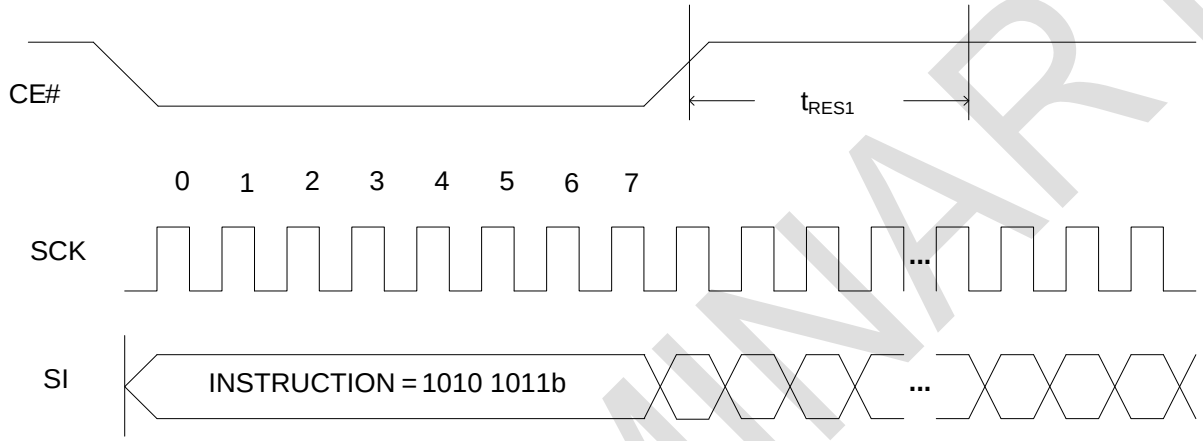
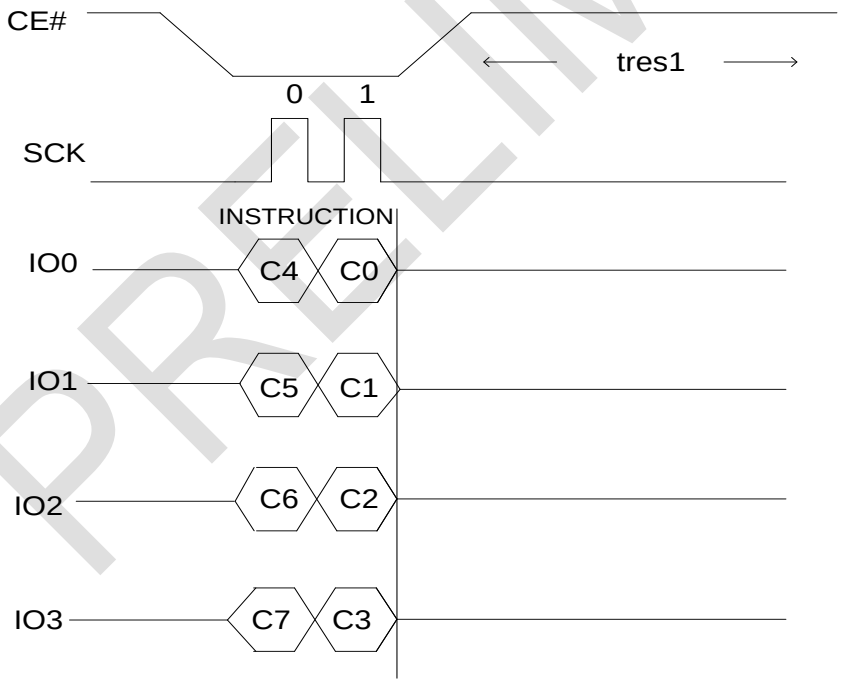



**Figure 8.37 Release Power Down Sequence (QPI)**




### 8.23 SET READ PARAMETERS OPERATION (SRP, C0h)

**Set Read Operational Driver Strength**

This device supports configurable Operational Driver Strengths in both SPI and QPI mode by setting three bits within the READ Register (ODS0, ODS1, ODS3). To set the ODS bits the SRP operation (C0h) instruction is required. The device's driver strength can be reduced as low as 12.50% of full drive strength. Details regarding the driver strength can be found in table 6.10.

**Note: The default driver strength is set to 50%**

**Figure 8.38 Set Read Parameters Operation.**

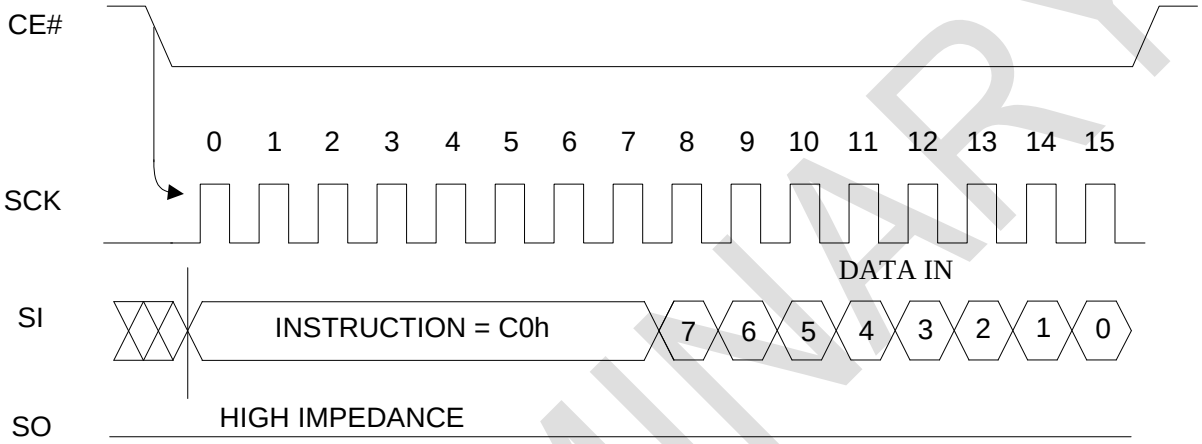
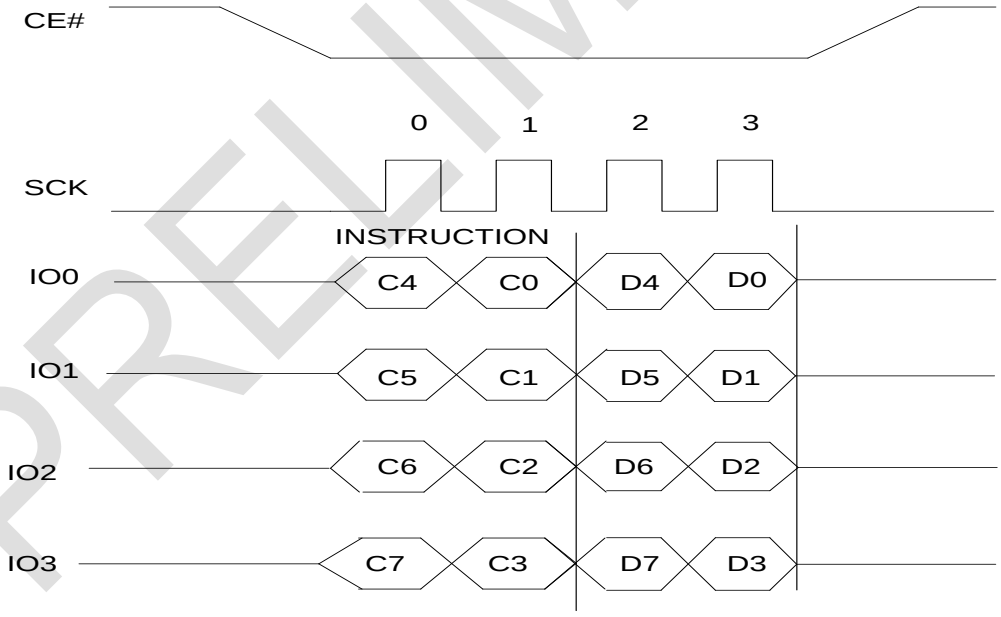



**Figure 8.39 Set Read Parameters Operation. (QPI)**



**Read with "8/16/32/64-Byte Wrap Around"**

This device supports Burst Read in both SPI and QPI mode and defines the wrap around feature enable. To set the Burst length, following command operations are required: "C0h" in the first Byte (8-clocks), "0h" to enable wrap around and to define the data to wrap around on and "1h" to disable wrap around.
The wrap around unit is defined within the one page, 256Bytes, with random initial address. Wrap around mode disabled for the default state of the device. To exit wrap around, it is required to issue another "C0" command to set bit3 = 0. Otherwise, wrap around status will be retained until power down or reset operations. To change wrap around depth, it is required to issue another "C0" command to set bit0 and bit1 (Detailed information in Table 6.7 Burst Length Data). QPI "0Bh" "EBh" and SPI (EBh, 03h, 0Bh, BBh) support wrap around feature after wrap around enable.

Refer to Figures 8.38 and 8.39 for instruction sequence.

### 8.24 READ PRODUCT IDENTIFICATION (RDID, ABh)

The Release from Power-down /read Device ID instruction is a multi-purpose instruction. It can support both SPI and QPI mode. The Read Product Identification (RDID) instruction is for reading out the old style of 8-bit Electronic Signature, whose values are shown as table of ID Definitions.

The RDID instruction code is followed by three dummy bytes, each bit being latched-in on SI during the rising SCK edge. Then the Device ID is shifted out on SO with the MSB first, each bit been shifted out during the falling edge of SCK. The RDID instruction is ended by CE# going high. The Device ID7-ID0 outputs repeatedly if additional clock cycles are continuously sent on SCK while CE# is at low.

**Table 8.3 Product Identification**

| **Manufacturer ID** | **(MF7-MF0)** | |
|---|---|---|
| ISSI Serial Flash | 9Dh | |
| **Instruction** | **ABh, 90h** | **9Fh** |
| Device ID | (ID7-ID0) | (ID15-ID0) |
| 64Mb | 16h | 6017h |
| 32Mb | 15h | 6016h |

**Figure 8.40 Read Product Identification Sequence**

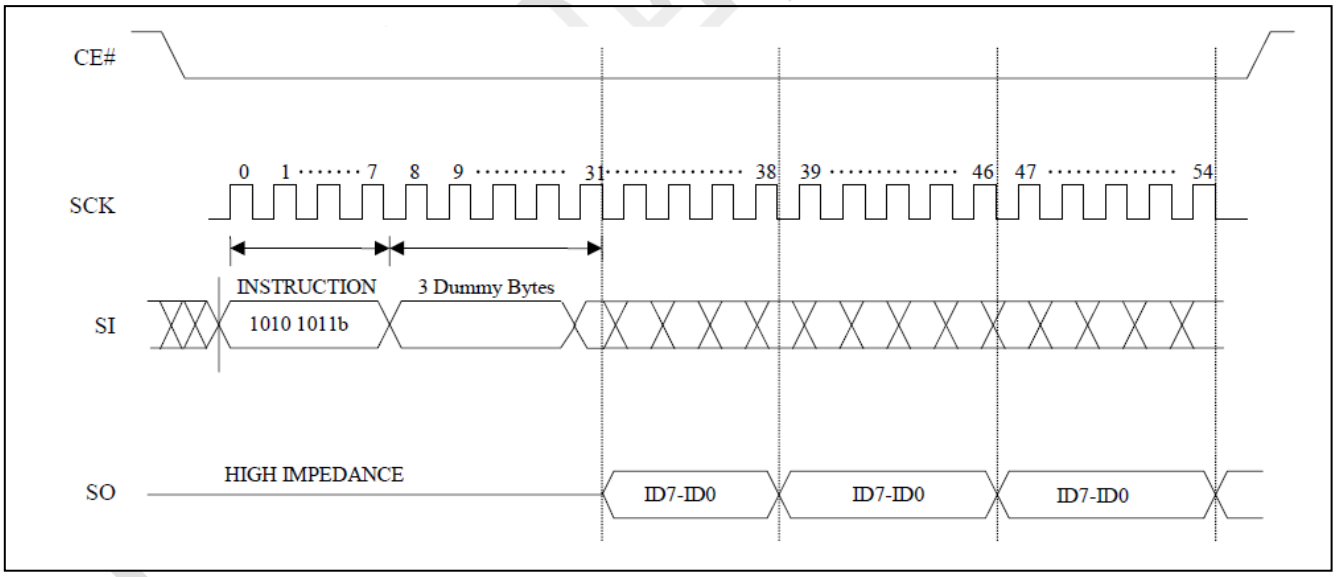
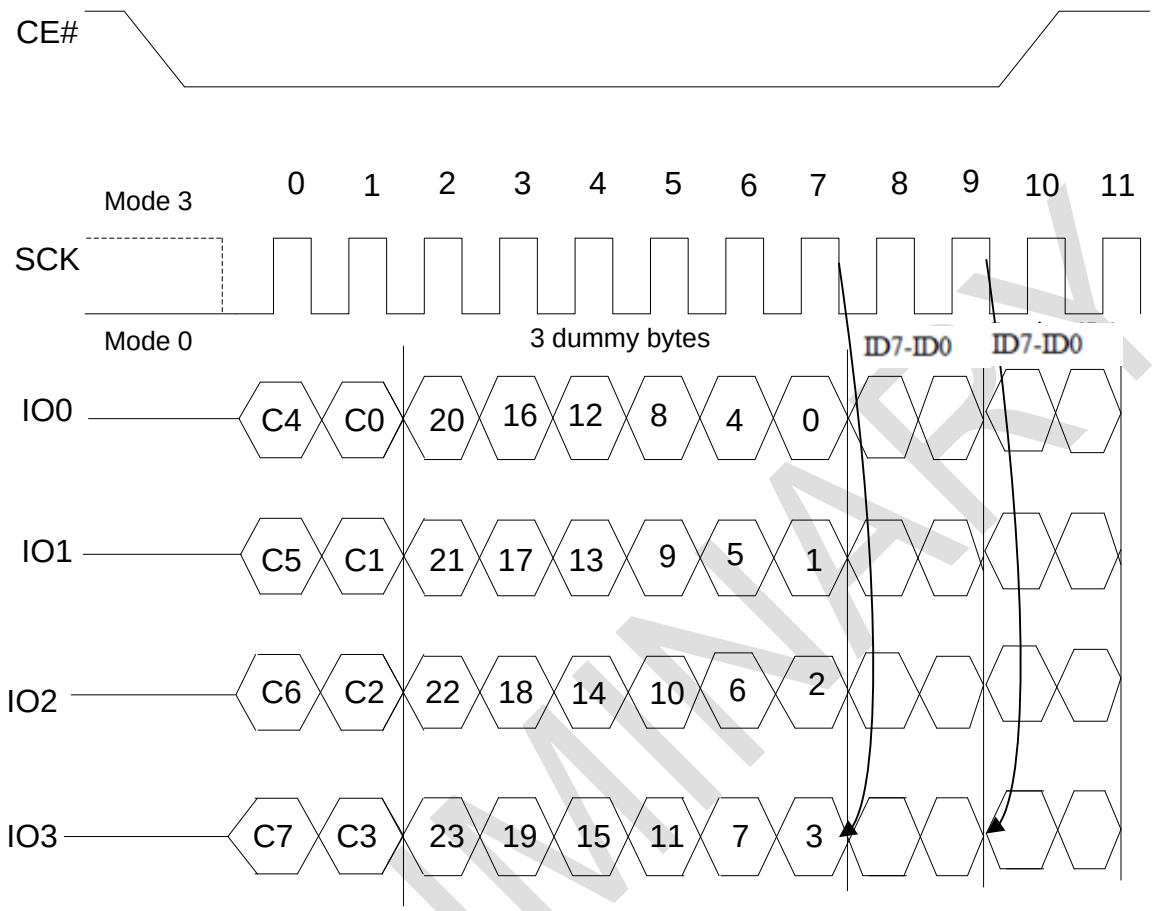



**Figure 8.41 Read Product Identification Sequence (QPI)**



### 8.25 READ PRODUCT IDENTIFICATION BY JEDEC ID OPERATION (RDJDID, 9Fh)

The JEDEC ID READ instruction allows the user to read the manufacturer and product ID of devices. Refer to Table 8.3 Product Identification for Manufacturer ID and Device ID. After the JEDEC ID READ command (9Fh in SPI mode, AFh in QPI mode) is input, the Manufacturer ID is shifted out MSB first followed by the 2-byte electronic ID (ID15-ID0), one bit at a time. Each bit is shifted out during the falling edge of SCK. If CE# stays low after the last bit of the 2-Byte electronic ID, the Manufacturer ID and 2-byte electronic ID will loop until CE# is pulled high..

**Figure 8.42 Read Product Identification by JEDEC ID READ Sequence**

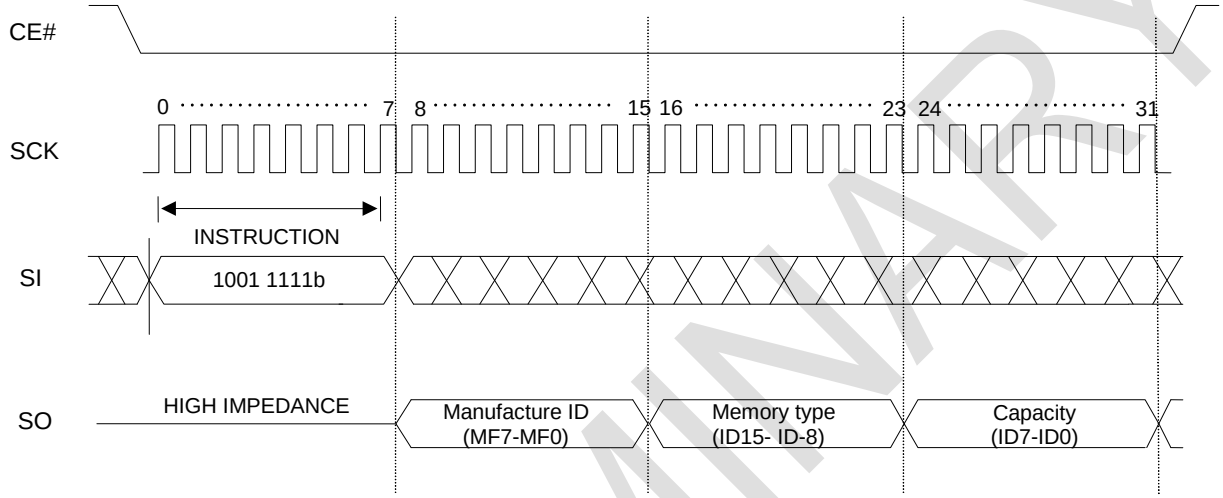
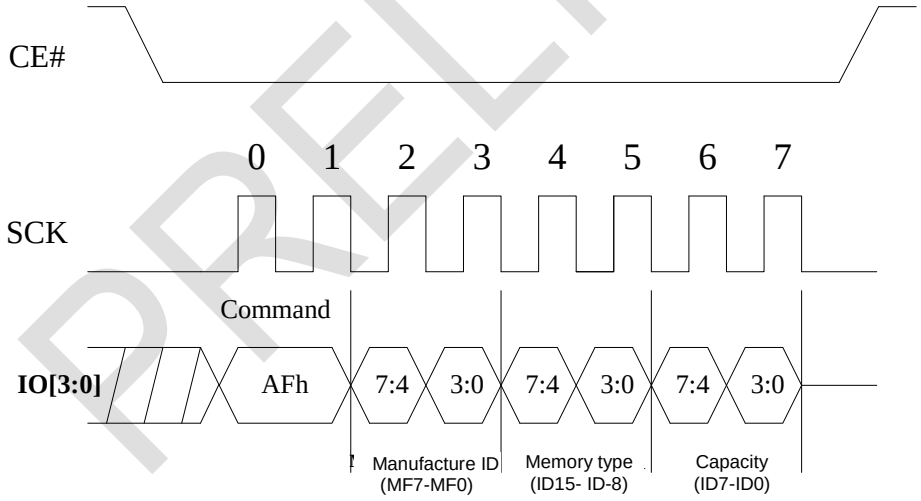


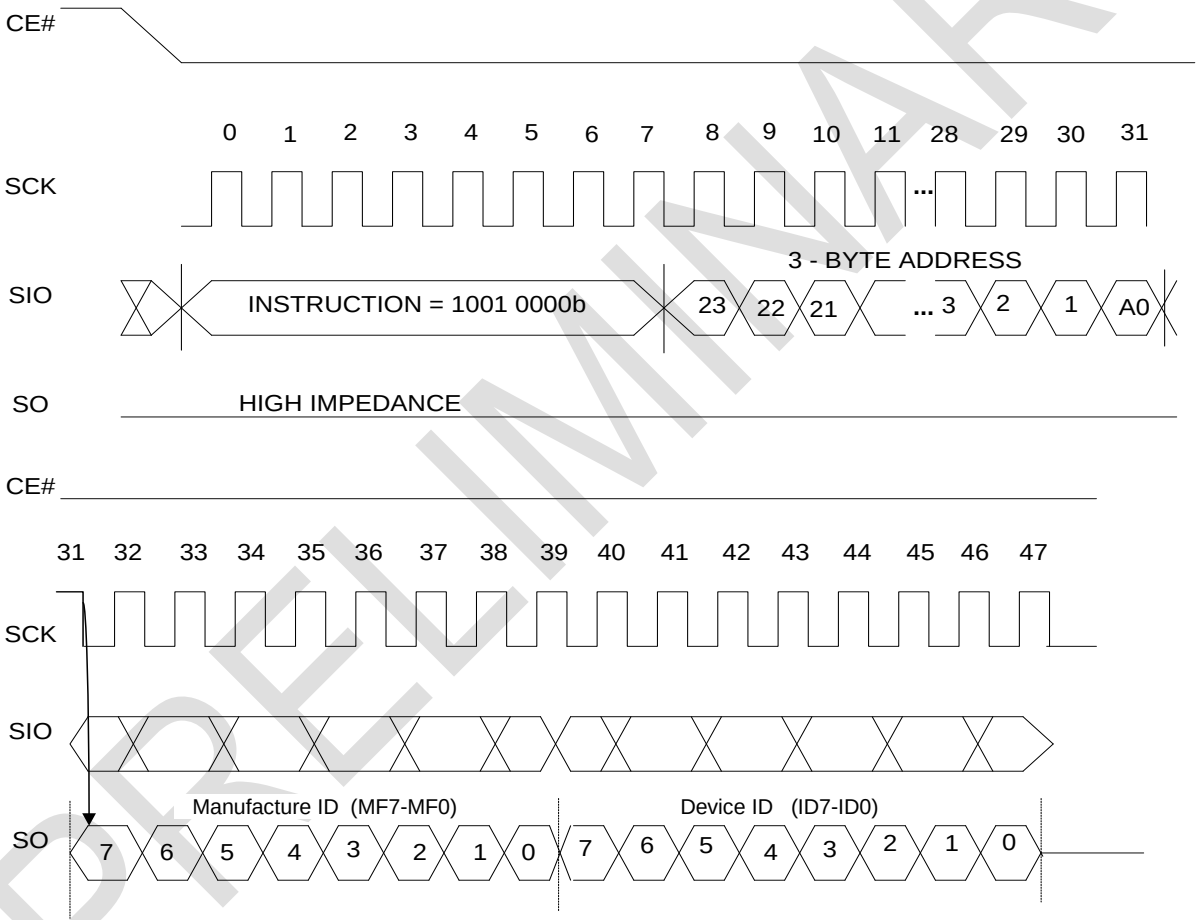

**Figure 8.43 RDIDQ COMMAND (Read ID in QPI Mode) OPERATION**

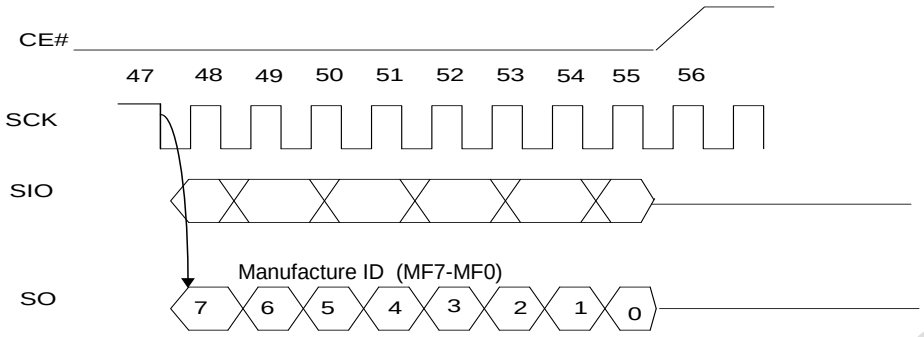


### 8.26 READ DEVICE MANUFACTURER AND DEVICE ID OPERATION (RDMDID, 90h)

The Read Product Identification (RDID) instruction allows the user to read the manufacturer and product ID of the devices. Refer to Table 8.3 Product Identification for manufacturer ID and device ID. The RDID instruction code is followed by two dummy bytes and one byte address (A7~A0), each bit being latched-in on SI during the rising edge of SCK. If one byte address is initially set to A0 = 0, then the manufacturer ID (9Dh) is shifted out on SO with the MSB first followed by the device ID7- ID0. Each bit shifted out during the falling edge of SCK. If one byte address is initially set to A0 = 1, then Device ID7-ID0 will be read first followed by the Manufacture ID (9Dh). The manufacture and device ID can be read continuously alternating between the two until CE# is driven high.

**Figure 8.44 Read Product Identification by RDMDID READ Sequence**






**Note1.**
(1) ADDRESS A0 = 0, will output the 1-byte Manufacture ID (MF7-MF0) -> 1-byte device (ID7-ID0)
ADDRESS A0 = 1, will output the 1-byte device (ID7-ID0) -> 1-byte Manufacture ID (MF7-MF0)
(2) The Manufacture and Device ID can be read continuously and will alternate from one to the other until CE# pin is pulled high.

### 8.27 READ UNIQUE ID NUMBER (RDUID, 4Bh)

The Read Unique ID Number (RDUID) instruction accesses a factory-set read-only 16-byte number that is unique to the device. The Id number can be used in conjunction with user software methods to help prevent copying or cloning of a system. The RDUID instruction is instated by driving the CE# pin low and shifting the instruction code (4Bh) followed by 3 address bytes and a dummy byte. After which, the 16-byte ID is shifted out on the falling edge of CLK as shown below.

**Note :**
**The last byte of data will repeat when as the data will be read continuously.**

**Figure 8.45 RDUID COMMAND OPERATION**

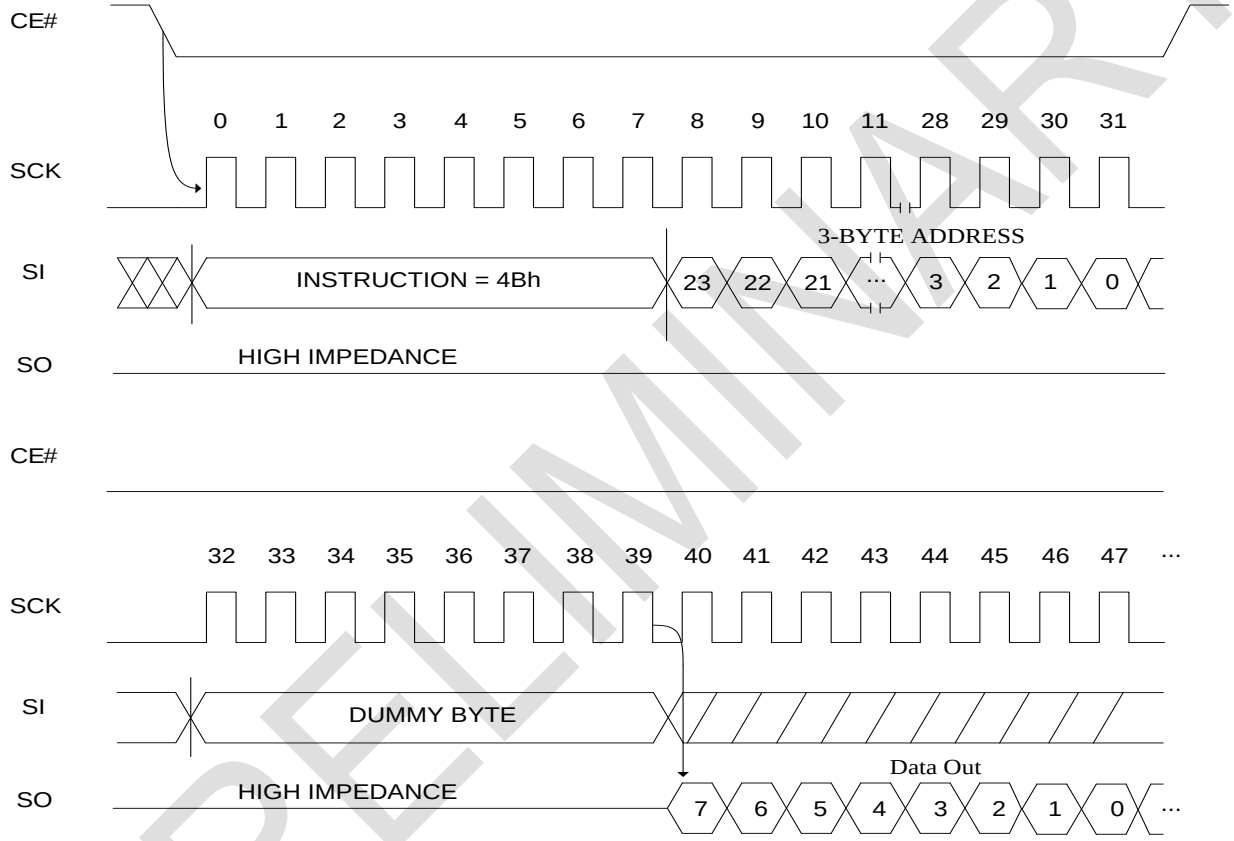



Table 8.4 Unique ID Addressing

| A[23:16] | A[15:9] | A[8:4] | A[3:0] |
|---|---|---|---|
| XXh | XXh | 00h | 0h Byte address |
| XXh | XXh | 00h | 1h Byte address |
| XXh | XXh | 00h | 2h Byte address |
| XXh | XXh | 00h | ⋮ |
| XXh | XXh | 00h | Fh Byte address |

### 8.28 READ SFDP OPERATION (RDSFDP, 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.
The sequence of issuing RDSFDP instruction is same as FAST_READ: CE# goes low→send RDSFDP instruction (5Ah)→send 3 address bytes on SI pin→send 1 dummy byte on SI pin→read SFDP code on SO→to end RDSFDP operation can use CE# to high at any time during data out.

**Figure 8.46 RDSFDP COMMAND (Read SFDP) OPERATION**

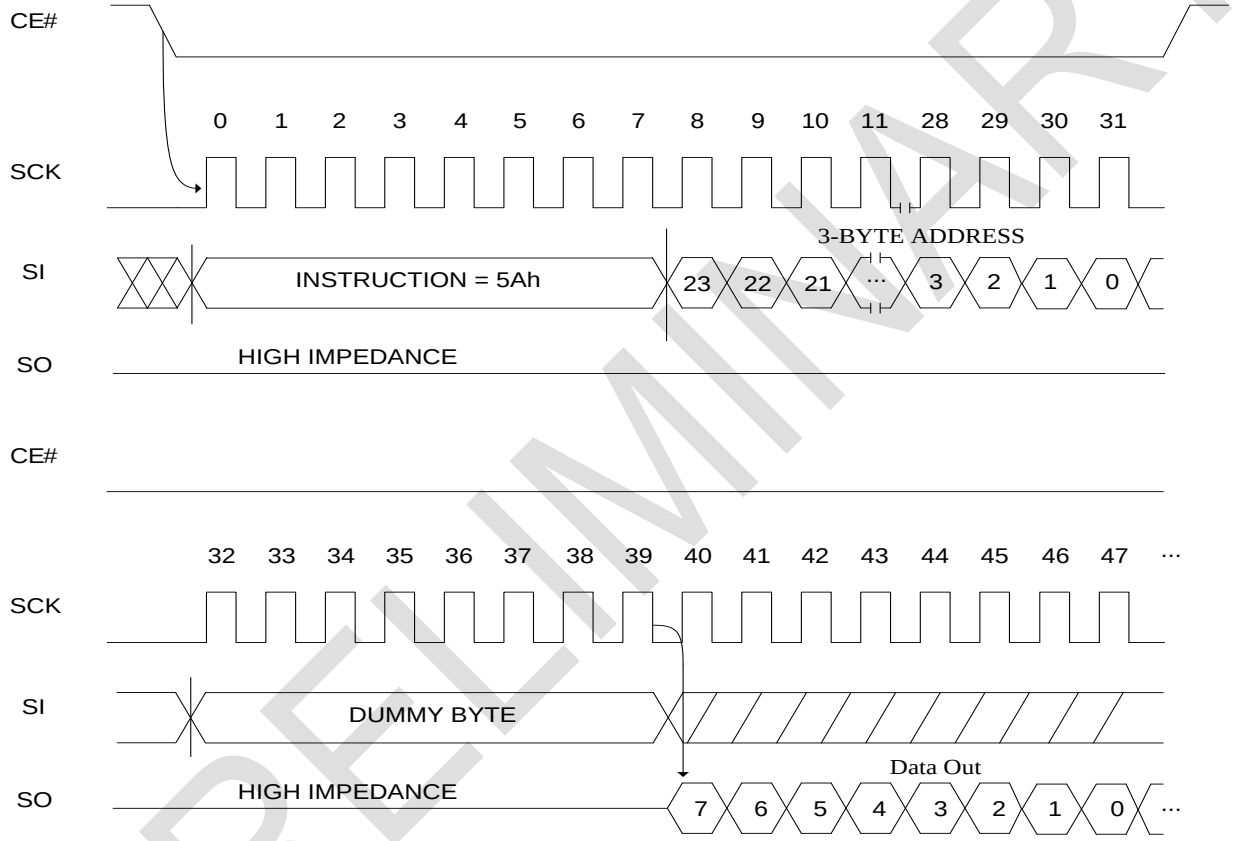


**Table 8.4 SFDP (Serial Flash Discoverable Parameters) Definition Table**

| Address (Byte) | Data (Hex) | Data (binary) | Description | Comment |
|---|---|---|---|---|
| 00h | 53h | 0101 0011 | SFDP Signature | Fixed : 50444653h |
| 01h | 46h | 0100 0110 | | |
| 02h | 44h | 0100 0100 | | |
| 03h | 50h | 0101 0000 | | |
| 04h | 00h | 0000 0000 | SFDP Minor Revision Number | Rev 1.0 |
| 05h | 01h | 0000 0001 | SFDP Major Revision Number | |
| 06h | 00h | 0000 0000 | Number of Parameter Numbers | 1 Parameter Header |
| 07h | FFh | 1111 1111 | Reserved | |
| 08h | 00h | 0000 0000 | JEDEC Specified | JEDEC specified=00h |
| 09h | 00h | 0000 0000 | Parameter Table Minor Revision Number | Rev 1.0 |
| 0Ah | 01h | 0000 0001 | Parameter Table Major Revision Number | |
| 0Bh | 09h | 0000 1001 | Parameter Table Length | 9 DWORDs |
| 0Ch | 30h | 0011 0000 | JEDEC Parameter Table Pointer (PTP) | Point = 000030h |
| 0Dh | 00h | 0000 0000 | | |
| 0Eh | 00h | 0000 0000 | | |
| 0Fh | FFh | 1111 1111 | Reserved | |
| 10h | 9Dh | 1001 1101 | Manufacturer ID | |
| 11h | 00h | 0000 0000 | Reserved | |
| 12h | 01h | 0000 0001 | Reserved | |
| 13h | 09h | 0000 1001 | Reserved | |
| 14h | 60h | 0110 0000 | ISSI Parameter Table Pointer (PTP) | Point = 000060h |
| 15h | 00h | 0000 0000 | | |
| 16h | 00h | 0000 0000 | | |
| 17h | FFh | 1111 1111 | Reserved | |

| Address (Byte) | Data (Hex) | Data (binary) | Description | Comment |
|---|---|---|---|---|
| 30h | 7Dh | 0001 | Bit[01:00 ]: Block/Sector Erase<br>00=Reserved<br>01=4KB Erase<br>10=Reserved<br>11=not support 4K Erase | Supports 4K sector erase |
| | | 0001 | Bit[2] : Write Granularity<br>0:1Byte ,1:64Byte or larger | Granularity is 64Byte or larger |
| | | 0001 | Bit[3] : Write Enable Instruction Required for Writing to Non-volatile Status Registers<br>0 : not required<br>1 : required | Required |
| | | 0001 | Bit[4] : Write Enable Opcode Select for writing to Non-Volatile Status Registers<br>0 : use 50h opcode<br>1 : use 06h opcode | use 01h opcode |
| | | 111 | Bit[7:5] : Reserved | |
| 31h | D7h/20h | 1101 0111<br>or<br>0010 0000 | Sector Erase OP Code (4KB) | |
| 32h | 3Fh | 0001 | Bit[0] : Reserved | |
| | | 0001 | Bit[1] : (1-1-2) Fast Read<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[1] : (1-1-4) Fast Read<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[2] : (1-4-4) Fast Read<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[3] : (1-2-2) Fast Read<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[4] : Double Transfer Rate (DTR)<br>0=not supported<br>1=supported | supported |
| | | 0000 0000 | Bit[6:5] : Address Bytes Number Type<br>00 : 3Bytes only, 01 : 3 or 4 Byte,<br>10 : 4Byte only, 11 : Reserved | 3 Bytes only |
| 33h | FFh | 1111 1111 | Reserved | |
| **Address (Byte)** | **Data (Hex)** | **Data (binary)** | **Description** | **Comment** |

| Address (Byte) | Data (Hex) | Data (binary) | Description | Comment |
|---|---|---|---|---|
| 34h | FFh | 1111 1111 | 64Mb = 03FF FFFF<br>32Mb = 01FF FFFF | |
| 35h | FFh | 1111 1111 | | |
| 36h | FFh | 1111 1111 | | |
| 37h | 03h (64Mb)<br>or<br>01h (32Mb) | 0000 0111 | | |
| 38h | 44h | 0100 | Bit[4:0] : (1-4-4)Fast Read Number of Dummy Cycles<br>0 0000b : Dummy cycles not supported | 4 dummy cycles |
| | | 0100 | Bit[7:5] : (1-4-4)Fast Read Number of Mode Bits<br>000b : Mode Bits not support | 4 mode bits |
| 39h | EBh | 1110 1011 | (1-4-4) Fast Read Opcode | |
| 3Ah | 00h | 1000 | Bit[4:0] : (1-1-4)Fast Read Number of Dummy Cycles<br>0 0000b : Dummy cycles not support | 8 Dummy cycles supported |
| | | 0000 | Bit[7:5] : (1-1-4)Fast Read Number of Mode Bits<br>000b : Mode Bits not support | Mode Bits not supported |
| 3Bh | 00h | 0000 0000 | (1-1-4) Fast Read Opcode | |
| 3Ch | 00h | 0000 | Bit[4:0] : (1-1-2)Fast Read Number of Dummy Cycles<br>0 0000b : Dummy cycles not support | Dummy cycles not supported |
| | | 0000 | Bit[7:5] : (1-1-2)Fast Read Number of Mode Bits<br>000b : Mode Bits not support | Mode Bits not supported |
| 3Dh | 3Bh | 0011 1011 | (1-1-2) Fast Read Opcode | |
| 3Eh | 04h | 0100 | Bit[4:0] : (1-2-2)Fast Read Number of Dummy Cycles<br>0 0000b : Dummy cycles not support | 4 dummy cycles |
| | | 0000 | Bit[7:5] : (1-2-2)Fast Read Number of Mode Bits<br>000b : Mode Bits not support | Mode Bits not supported |
| 3Fh | BBh | 1011 1011 | (1-2-2) Fast Read Opcode | |
| 40h | EEh | 0000 | Bit[0] : (2-2-2) Fast Read<br>0=not supported<br>1=supported | Not Supported |
| | | 111 | Bit[3:1] : Reserved | |
| | | 0000 | Bit[4] : (4-4-4) Fast Read<br>0=not supported<br>1=supported | Not Supported |
| | | 111 | Bit[7:5] : Reserved | |
| 41h | FFh | 1111 1111 | Reserved | |
| 42h | FFh | 1111 1111 | Reserved | |

| Address (Byte) | Data (Hex) | Data (binary) | Description | Comment |
|---|---|---|---|---|
| 43h | FFh | 1111 1111 | Reserved | |
| 44h | FFh | 1111 1111 | Reserved | |
| 45h | FFh | 1111 1111 | Reserved | |
| 46h | 00h | 0000 | Bit[4:0] : (2-2-2)Fast Read Number of Dummy Cycles<br>0 0000b : Dummy cycles not support | Dummy cycles not supported |
| | | 0000 | Bit[7:5] : (2-2-2)Fast Read Number of Mode Bits<br>000b : Mode Bits not support | Mode Bits not supported |
| 47h | FFh | 1111 1111 | (2-2-2) Fast Read Opcode | |
| 48h | FFh | 1111 1111 | Reserved | |
| 49h | FFh | 1111 1111 | Reserved | |
| 4Ah | 44h | 0100 | Bit[4:0] : (4-4-4)Fast Read Number of Dummy Cycles<br>0 0000b : Dummy cycles not support | 4 Dummy cycles supported |
| | | 0100 | Bit[7:5] : (4-4-4)Fast Read Number of Mode Bits<br>000b : Mode Bits not support | 4 Mode Bits supported |
| 4Bh | EBh | 1110 1011 | (4-4-4) Fast Read Opcode | |
| 4Ch | 0Ch | 0000 1100 | Sector Type 1 Size (4KB) | |
| 4Dh | 20h | 0010 0000 | Sector Type 1 Opcode | |
| 4Eh | 0Fh | 0000 1111 | Sector Type 1 Size (32KB) | |
| 4Fh | 52h | 0101 0010 | Sector Type 1 Opcode | |
| 50h | 10h | 0001 0000 | Sector Type 1 Size (64KB) | |
| 51h | D8h | 1101 1000 | Sector Type 1 Opcode | |
| 52h | 00h | 0000 0000 | Sector Type 1 Size (256KB) – Not support | |
| 53h | FFh | 1111 1111 | Sector Type 1 Opcode | |
| 61h:60h | 3600h | 0011 0110<br>0000 0000 | Vcc Supply Maximum Voltage<br>2000h = 2.00V<br>2700h = 2.70V<br>3600h = 3.60V | 3.60V |

| Address (Byte) | Data (Hex) | Data (binary) | Description | Comment |
|---|---|---|---|---|
| 63h:62h | 2300h | 0010 0011<br>0000 0000 | Vcc Supply Minimum Voltage<br>1650h = 1.65V<br>2300h = 2.30V<br>2700h = 2.70V | 2.30V |
| 65h:64h | B99Eh | 0000 | Bit[0] : HW Reset# Pin<br>0=not supported<br>1=supported | not supported |
| | | 0001 | Bit[1] : HW Hold# Pin<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[2] : Deep Power Down Mode<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[3] : SW Reset<br>0=not supported<br>1=supported | supported |
| | | 1001 1001 | Bit[11:04] : SW Reset Opcode :99h | |
| | | 0001 | Bit[12] : Power Suspend / Resume<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[13] : Erase Suspend / Resume<br>0=not supported<br>1=supported | supported |
| | | 0000 | Bit[14] : Reserved | |
| | | 0001 | Bit[15] : Wrap-Around Read Mode<br>0=not supported<br>1=supported | supported |
| 66h | C0h | 1100 0000 | Wrap-Around Read Mode Opcode | |
| 67h | 64h | 0110 0100 | Wrap-Around Read data length<br>08h=support 8B Wrap-Around Read<br>16h=support 8B & 16B<br>32h=support 8B &16B & 32B<br>64h=support 8B & 16B & 32B & 64B | Supports 8, 16, 32 or 64 Byte Wrap-Around Read |

| Address (Byte) | Data (Hex) | Data (binary) | Description | Comment |
|---|---|---|---|---|
| 69h:68h | C8DBh | 0001 | Bit[0] : Individual Block Lock<br>0=not supported<br>1=supported | supported |
| | | 0001 | Bit[1] : Individual Block Lock bit<br>0=not supported<br>1=supported | supported |
| | | 0011 0110 | Bit[09:02] : Individual Block Lock Opcode : 36h | |
| | | 0000 | Bit[10] : Individual Block Lock bit default status<br>0=not supported<br>1=supported | not Supported |
| | | 1000 | Bit[11] : Secured Programmable Area<br>0=not supported<br>1=supported | supported |
| | | 0000 | Bit[12] : Read Lock<br>0=not supported<br>1=supported | not supported |
| | | 0000 | Bit[13] : Permanent Lock<br>0=not supported<br>1=supported | not supported |
| | | 0001 0001 | Bit[15:14] : Reserved | |
| 6Ah | FFh | 1111 1111 | Reserved | |
| 6Bh | FFh | 1111 1111 | Reserved | |

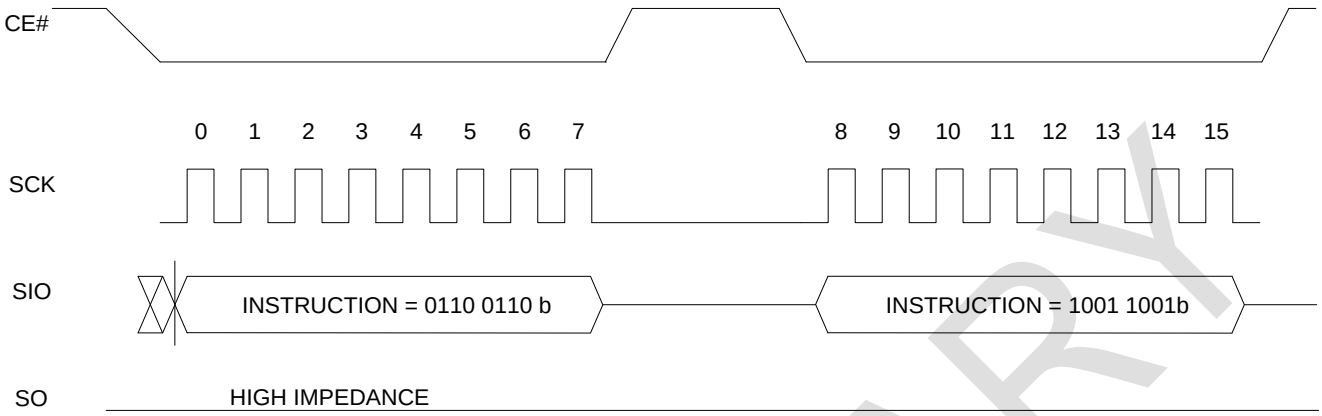
### 8.29 NO OPERATION (NOP, 00h)

The No Operation command only cancels a Reset Enable command. NOP has no impact on any other command. It can use in the SPI and QPI mode.
To execute a NOP, the host drives CE# low, sends the NOP command cycle (00H), then drives CE# high.

### 8.30 SOFTWARE RESET (RESET-ENABLE (RSTEN, 66h) AND RESET (RST, 99h)

The Reset operation is used as a system (software) reset that puts the device in normal operating mode. This operation consists of two commands: Reset-Enable (RSTEN) and Reset (RST).
The Reset operation requires the Reset-Enable command followed by the Reset command. Any command other than the Reset command after the Reset-Enable command will disable the Reset-Enable.
Execute the CE# pin low → sends the Reset-Enable command (66h), and drives CE# high. Next, the host drives CE# low again, sends the Reset command (99h), and drives CE# high.
A device reset during an active Program or Erase operation aborts the operation, which can cause the data of the targeted address range to be corrupted or lost. Depending on the prior operation, the reset timing may vary. Recovery from a Write operation requires more latency time than recovery from other operations.

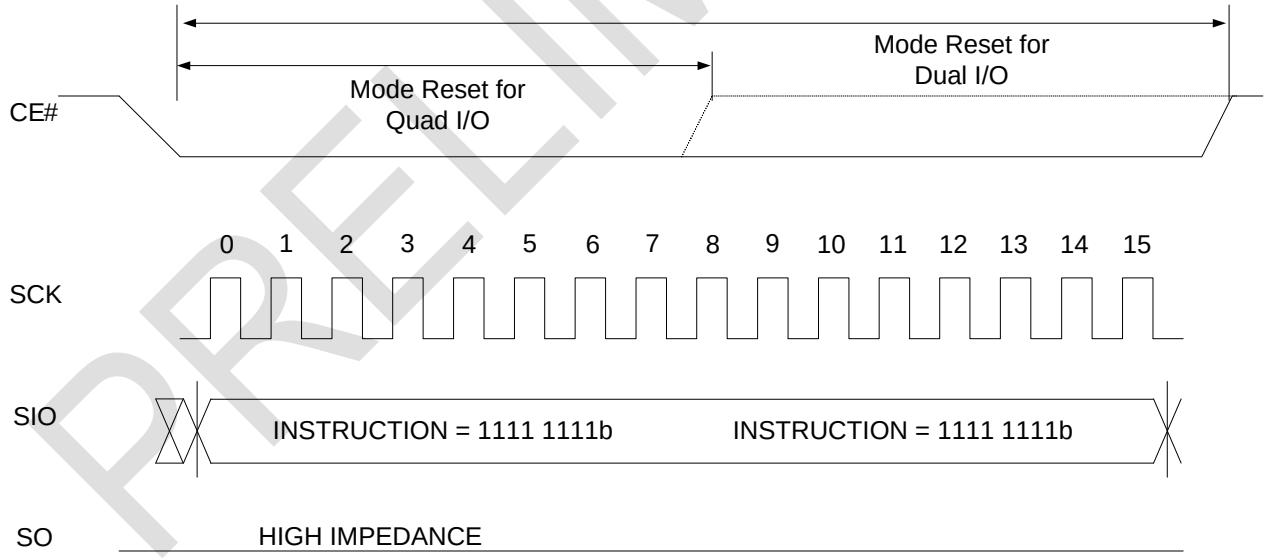
**Figure 8.47 SOFTWARE RESET ENABLE, SOFTWARE RESET OPERATIONS (RSTEN, 66h + RST, 99h)**




## 8.31 MODE RESET OPERATION (RSTM, FFh)

The Mode Reset command is used to conclude subsequent FRDIO and FRQIO operations. It resets the Mode bits to a value that is not Ax. It should be executed after an FRDIO or FRQIO operation, and is recommended also as the first command after a system reset. The timing sequence is different depending whether the MR command is used after an FRDIO or FRQIO, as shown in Figure 8.48

**Figure 8.48 Mode Reset Command**



### 8.32 SECURITY INFORMATION ROW

The security information row is comprised of an additional 4 x 256 bytes of programmable information. The security bits can be reprogrammed by the user. Any program security instruction issued while an erase, program or write cycle is in progress is rejected without having any effect on the cycle that is in progress.

**Table 8.5 Information Row Address**

| Address Assignment | A[23:16] | A[15:8] | A[7:0] |
|---|---|---|---|
| IRL0(Information row lock0) | 00h | 00h | Byte address |
| IRL1 | 00h | 10h | Byte address |
| IRL2 | 00h | 20h | Byte address |
| IRL3 | 00h | 30h | Byte address |

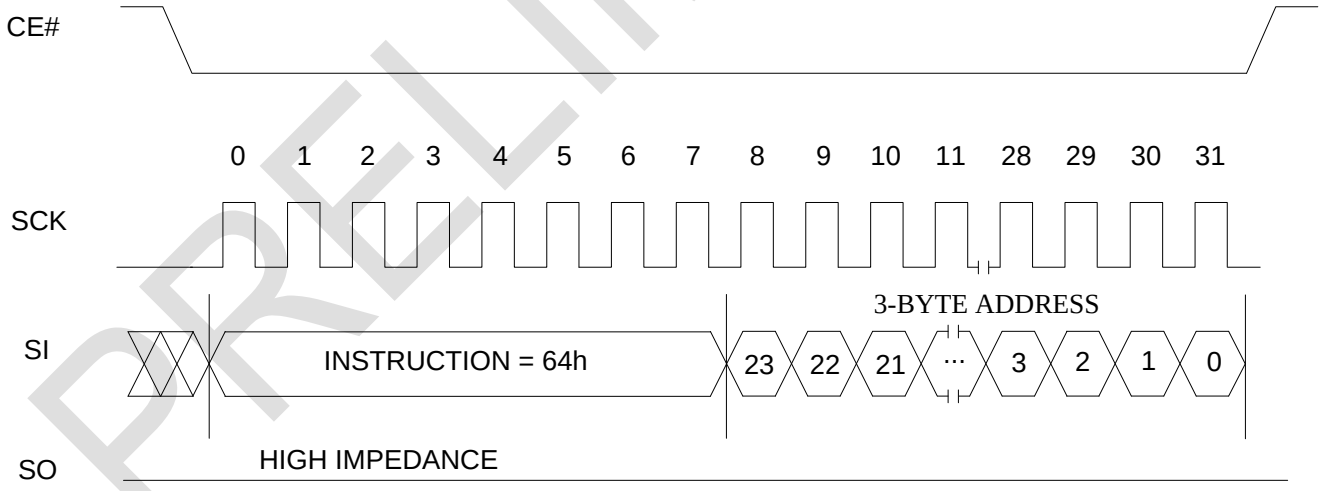
Bit 7~4 of the Function Register is used to permanently lock the programmable memory array.
- When Function Register bit IRLx = '0', the 256 bytes of the programmable memory array can be programmed.
- When Function Register bit IRLx = '1', the 256 bytes of the programmable memory array function as read only.

### 8.33 INFORMATION ROW ERASE OPERATION (IRER, 64h)

Information Row Erase instruction erases the Information Row x (x : 0~3) array , the Write Enable Latch (WEL) must be set via a Write Enable (WREN) instruction. The WEL bit is reset automatically after the completion of sector an erase operation.

A IRER instruction is entered, after CE# is pulled low to select the device and stays low during the entire instruction sequence The IRER instruction code, and three address bytes are input via SI. Erase operation will start immediately after CE# is pulled high. The internal control logic automatically handles the erase voltage and timing. Refer to Figure 8.49 for IRER Sequence.

**Figure 8.49 IRER COMMAND (Information Row Erase) OPERATION**



### 8.34 INFORMATION ROW PROGRAM OPERATION (IRP, 62h)

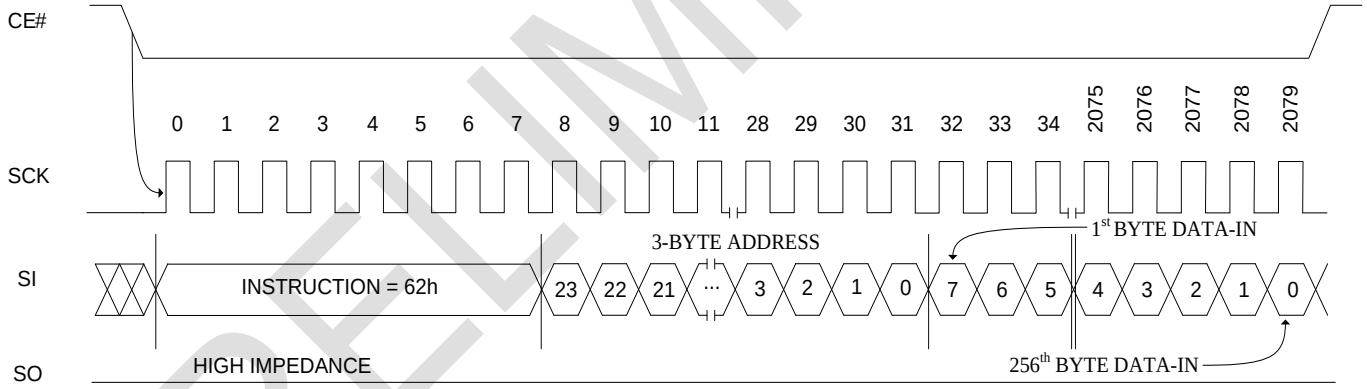
The Information Row Program (IRP) instruction allows up to 1024 bytes, 4x256 bytes, data to be programmed into memory in a single operation. Before the execution of PP instruction, the Write Enable Latch (WEL) must be enabled through a Write Enable (WREN) instruction.

The IRP instruction code, three address bytes and program data (1 to 256 bytes) are input via the SI line. Program operation will start immediately after the CE# goes high, otherwise the IRP instruction will not be executed. The internal control logic automatically handles the programming voltages and timing. During a program operation, all instructions will be ignored except the RDSR instruction. The progress or completion of the program operation can be determined by reading the WIP bit in Status Register via a RDSR instruction. If the WIP bit is "1", the program operation is still in progress. If WIP bit is "0", the program operation has completed.

If more than 1024 bytes data are sent to a device, the address counter rolls over within the same page. The previously latched data are discarded and the last 1024 bytes data are kept to be programmed into the page. The starting byte can be anywhere within the page. When the end of the page is reached, the address will wrap around to the beginning of the same page. If the data to be programmed are less than a full page, the data of all other bytes on the same page will remain unchanged.

**Note : A program operation can alter "1"s into "0"s, but an erase operation is required to change "0"s back to "1"s. A byte cannot be reprogrammed without first erasing the whole sector or block.**

**Figure 8.50 IRP COMMAND (Information Row Program) OPERATION**



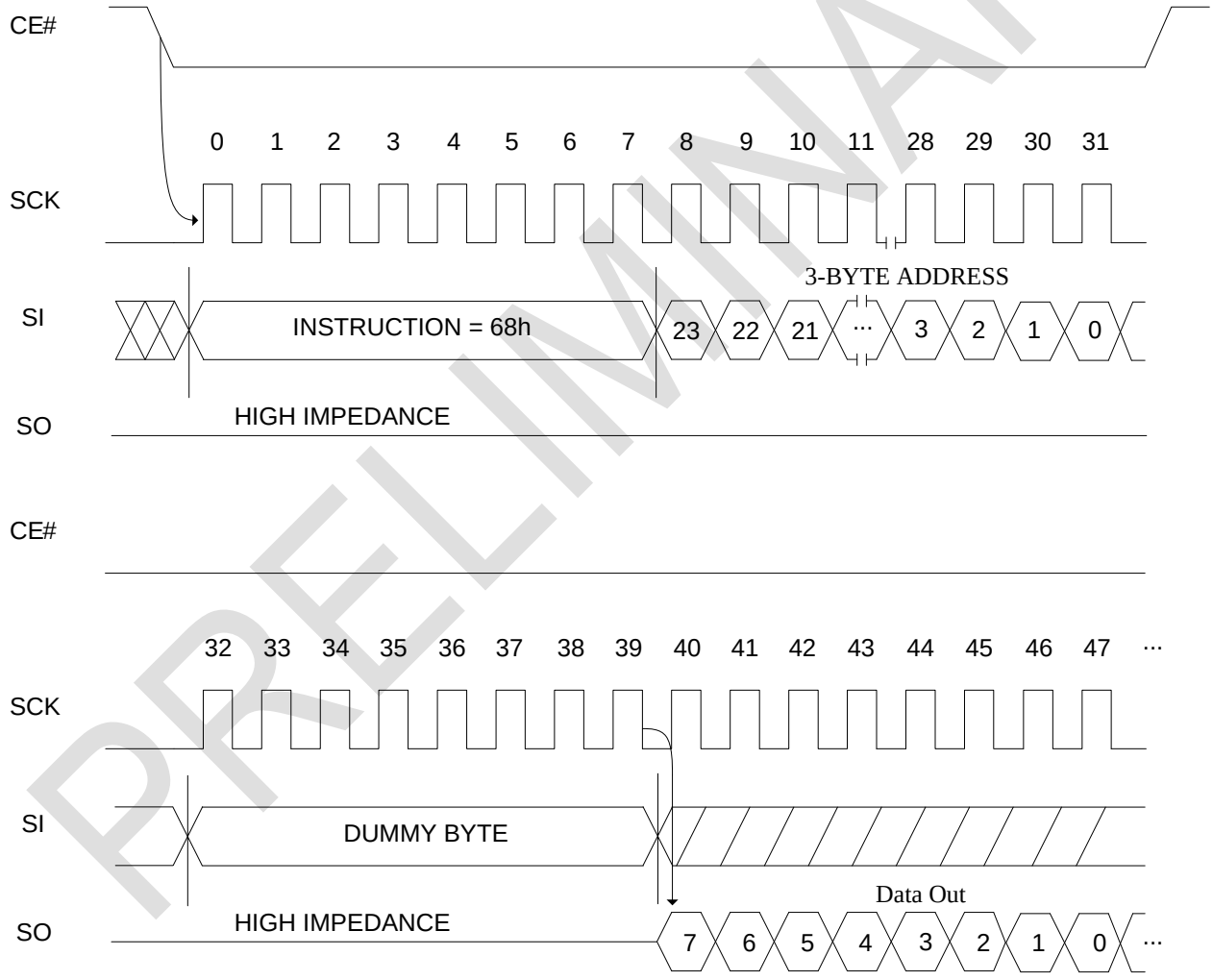
### 8.35 INFORMATION ROW READ OPERATION (IRRD, 68h)

The IRRD instruction is used to read memory data at up to a 133MHz clock.

The IRRD instruction code is followed by three address bytes (A23 - A0) and a dummy byte (8 clocks), transmitted via the SI line, with each bit latched-in during the rising edge of SCK. Then the first data byte addressed is shifted out on the SO line, with each bit shifted out at a maximum frequency $f_{CT}$, during the falling edge of SCK.

The address is automatically incremented after each byte of data is shifted out. When the highest address is reached, the address counter will roll over to the 000000h address, allowing the entire memory to be read with a single IRRD instruction. The IRRD instruction is terminated by driving CE# high (VIH). If a IRRD instruction is issued while an Erase, Program or Write cycle is in process (WIP=1) the instruction is ignored and will not have any effects on the current cycle

**Figure 8.51 IRRD COMMAND (Information Row Read) OPERATION**

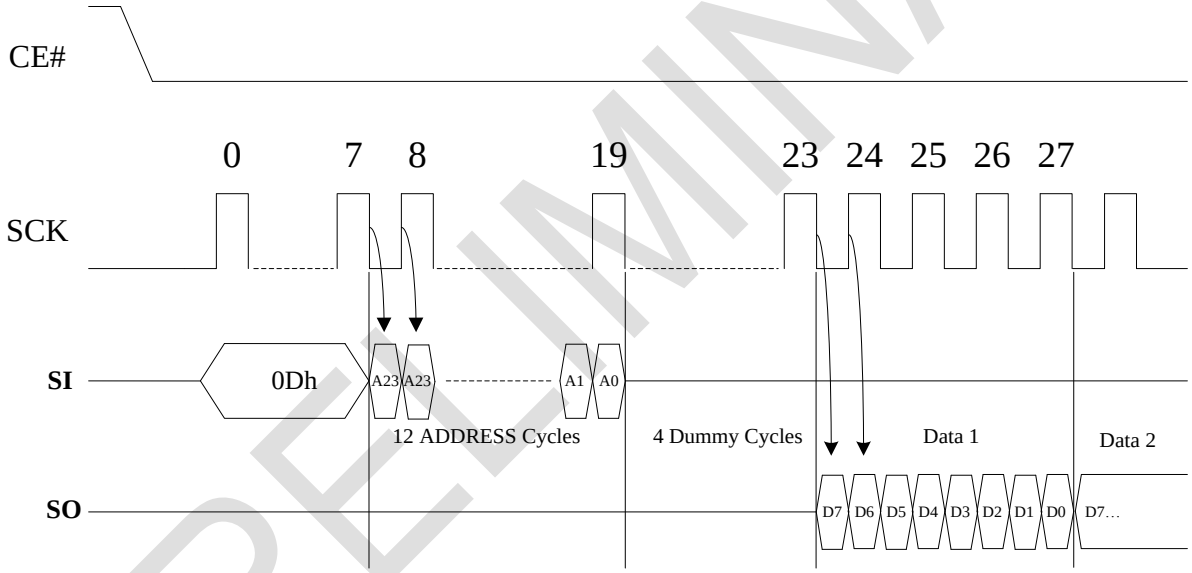


### 8.36 FAST READ DTR MODE OPERATION (FRDTR, 0DH)

The FRDTR instruction is for doubling reading data out, signals are triggered on both rising and falling edge of clock. The address is latched on both rising and falling edge of SCK, and data of each bit shifts out on both rising and falling edge of SCK at a maximum frequency $f_{C2}$. The 2-bit address can be latched-in at one clock, and 2-bit data can be read out at one clock, which means one bit at rising edge of clock, the other bit at falling edge of clock. The first address byte can be at any location.
The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single FRDTR instruction. The address counter rolls over to 0 when the highest address has been reached.
The sequence of issuing FRDTR instruction is: CE# goes low → sending FRDTR instruction code (1bit per clock) → 3-byte address on SI (2-bit per clock) → 6-dummy clocks (default) on SI → data out on SO (2-bit per clock) → to end FRDTR operation can use CE# to high at any time during data out. (Please refer to Figure 8.52)
While Program/Erase/Write Status Register cycle is in progress, FRDTR instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

**Note: Mode 0 is not supported in DTR operation**

**Figure 8.52 FRDTR COMMAND (Fast Read DTR Mode) OPERATION**



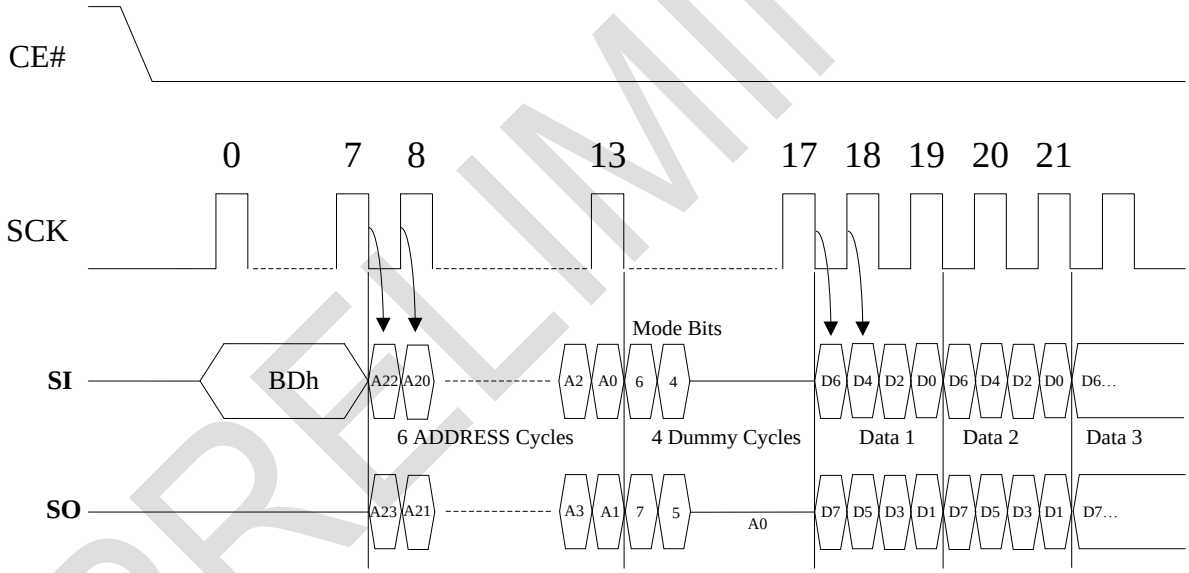

**Note : Number of dummy cycles depends on clock speed. Detailed information in Table 6.9. Read Dummy Cycles**

### 8.37 FAST READ DUAL IO DTR MODE OPERATION (FRDDTR, BDH)

The FRDDTR instruction enables Double Transfer Rate throughput on dual I/O of Serial Flash in read mode. The address (interleave on dual I/O pins) is latched on both rising and falling edge of SCK, and data (interleave on dual I/O pins) shift out on both rising and falling edge of SCK at a maximum frequency $f_{T2}$. The 4-bit address can be latched-in at one clock, and 4-bit data can be read out at one clock, which means two bits at rising edge of clock, the other two bits at falling edge of clock. The first address byte can be at any location.
The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single FRDDTR instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing FRDDTR instruction, the following address/dummy/ data out will perform as 4-bit instead of previous 1-bit.
The sequence of issuing FRDDTR instruction is: CE# goes low → sending FRDDTR instruction (1-bit per clock) → 24-bit address interleave on SIO1 & SIO0 (4-bit per clock) → 6-bit dummy clocks on SIO1 & SIO0 → data out interleave on SIO1 & SIO0 (4-bit per clock) → to end FRDDTR operation can use CE# to high at any time during data out (Please refer to Figure 8.53 for 2 x I/O Double Transfer Rate Read Mode Timing Waveform).
While Program/Erase/Write Status Register cycle is in progress, FRDDTR instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

**Note: Mode 0 is not supported in DTR operation**

**Figure 8.53 FRDDTR COMMAND (Fast Read Dual IO DTR Mode) OPERATION**




**Note : Number of dummy cycles depends on clock speed. Detailed information in Table 6.9. Read Dummy Cycles**

### 8.38 FAST READ QUAD IO DTR MODE OPERATION (FRQDTR, EDH)

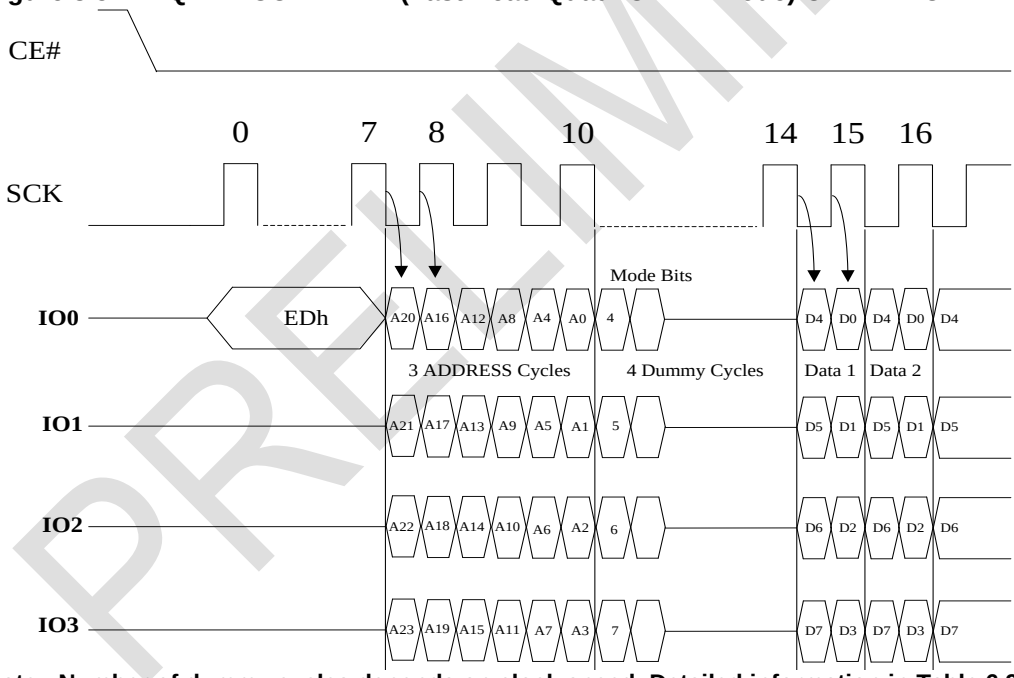
The FRQDTR instruction enables Double Transfer Rate throughput on quad I/O of Serial Flash in read mode. A Quad Enable (QE) bit of status Register must be set to "1" before sending the FRQDTR instruction. The address (interleave on 4 I/O pins) is latched on both rising and falling edge of SCK, and data (interleave on 4 I/O pins) shift out on both rising and falling edge of SCK at a maximum frequency $f_{Q2}$. The 8-bit address can be latched-in at one clock, and 8-bit data can be read out at one clock, which means four bits at rising edge of clock, the other four bits at falling edge of clock. The first address byte can be at any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single FRQDTR instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing FRQDTR instruction, the following address/dummy/data out will perform as 8-bit instead of previous 1-bit.

The sequence of issuing FRQDTR instruction is: CE# goes low → sending FRQDTR instruction (1-bit per clock) → 24-bit address interleave on SIO3, SIO2, SIO1 & SIO0 (8-bit per clock) → 8 dummy clocks → data out interleave on SIO3, SIO2, SIO1 & SIO0 (8-bit per clock) → to end FRQDTR operation can use CE# to high at any time during data out.

Another sequence of issuing enhanced mode of FRQDTR instruction especially useful in random access is: CE# goes low → sending FRQDTR instruction (1-bit per clock) → 3-bytes address interleave on SIO3, SIO2, SIO1 & SIO0 (8-bit per clock) → performance enhance toggling bit P[7:0] → 7 dummy clocks → data out(8-bit per clock) still CE# goes high → CE# goes low (eliminate 4 Read instruction) → 24-bit random access address (Please refer to Figure 8.54 for 4x I/O Double Transfer Rate read enhance performance mode timing waveform).

While Program/Erase/Write Status Register cycle is in progress, FRQDTR instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

**Figure 8.54 FRQDTR COMMAND (Fast Read Quad IO DTR Mode) OPERATION**



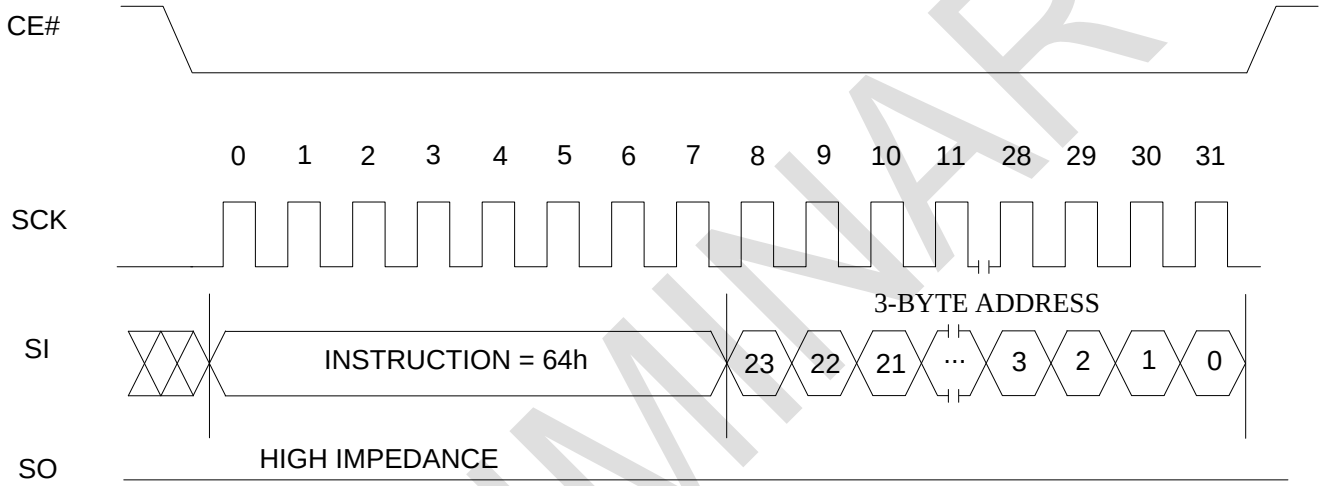

**Note : Number of dummy cycles depends on clock speed. Detailed information in Table 6.9. Read Dummy Cycles**

## 8.39 SECTOR LOCK/UNLOCK FUNCTIONS

### SECTOR UNLOCK OPERATION (SECUNLOCK, 26h)

The Sector unlock command allows the user to select a specific sector to allow program and erase operations. This instruction is effective when the blocks are designated as write-protected through the BP0, BP1, BP2 and BP3 bits in the status register. Only one sector can be enabled at any time. To enable a different sector, a previously enabled sector must be disabled by executing a Sector Lock command. The instruction code is followed by a 24-bit address specifying the target sector, but A0 through A11 are not decoded. The remaining sectors within the same block remain in read-only mode.

**Sector Unlock Sequence**



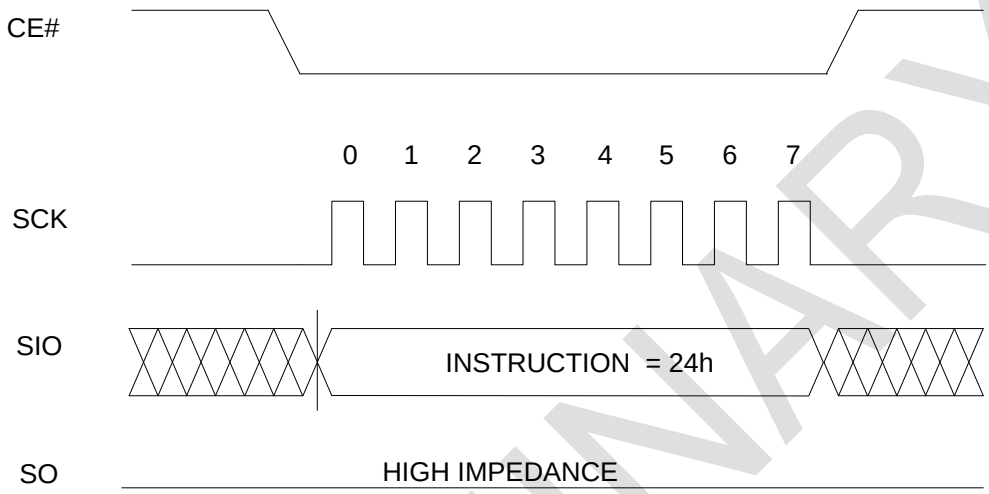

**In the Sector Unlock procedure, [A11:A0] need to be "0," in order for the unlock procedure to be completed, the chip will regard anything else as an illegal command.**

**Note:**
**1. If the number of clock cycles do not match 8 cycles (command) + 24 clocks (address), the command will be ignored.**
**2. WREN (06h) must be executed before sector unlock instructions.**

**SECTOR LOCK OPERATION (SECLOCK, 24h)**

The Sector Lock command reverses the function of the Sector Unlock command. The instruction code does not require an address to be specified, as only one sector can be enabled at a time. The remaining sectors within the same block remain in read-only mode.

**Sector Lock Sequence**



# 9. ELECTRICAL CHARACTERISTICS

## 9.1 ABSOLUTE MAXIMUM RATINGS (1)

| | | |
|---|---|---|
| Storage Temperature | | -55°C to +130°C |
| Surface Mount Lead Soldering Temperature | Standard Package | 240°C 3 Seconds |
| | Lead-free Package | 260°C 3 Seconds |
| Input Voltage with Respect to Ground on All Pins | | -0.5V to Vcc + 0.5V |
| All Output Voltage with Respect to Ground | | -0.5V to Vcc + 0.5V |
| Vcc | | -0.5V to +6.0V |

**Notes :**
**1. Applied conditions greater than those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.**

## 9.2 OPERATING RANGE

| | |
|---|---|
| Part Number | IS25LP032/064 |
| Operating Temperature (Extended Grade) | -40°C to 105°C |
| Operating Temperature (Automotive Grade A1) | -40°C to 85°C |
| Operating Temperature (Automotive Grade A2) | -40°C to 105°C |
| Operating Temperature (Automotive Grade A3) | -40°C to 125°C |
| Vcc Power Supply | 2.3 V – 3.6 V |

## 9.3 DC CHARACTERISTICS

(Under operating range)

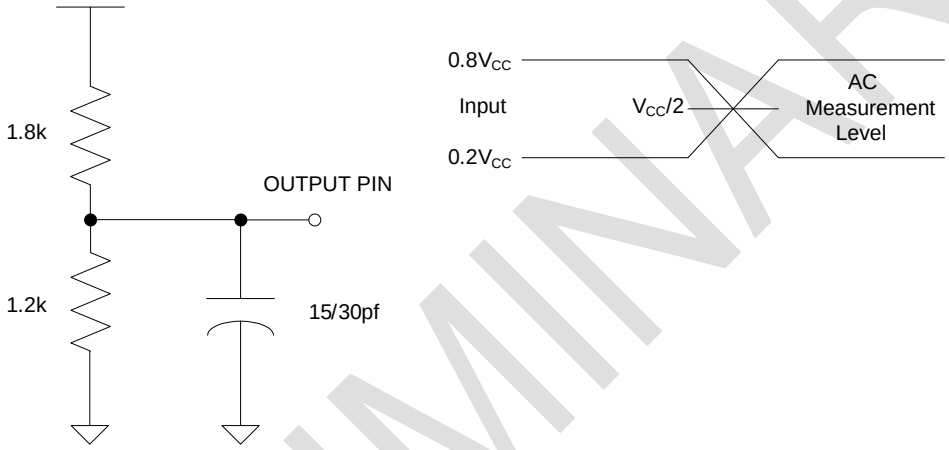
| Symbol | Parameter | Condition | | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|---|
| $I_{CC1}$ | $V_{cc}$ Active Read Current | $V_{CC}$ = 3.6V at 50MHz, SO = Open | | | 10 | 15 | mA |
| $I_{CC2}$ | $V_{cc}$ Program/Erase Current | $V_{CC}$ = 3.6V at 50MHz, SO = Open | | | 25 | 40 | mA |
| $I_{SB1}$ | $V_{cc}$ Standby Current CMOS | $V_{CC}$ = 3.6V, CE# = $V_{CC}$ | | | | 50 | □A |
| $I_{SB2}$ | $V_{cc}$ Standby Current TTL | $V_{CC}$ =3.6V, CE# = $V_{IH}$ to $V_{CC}$ | | | | 3 | mA |
| $I_{LI}$ | Input Leakage Current | $V_{IN}$ = 0V to $V_{CC}$ | | | | 1 | □A |
| $I_{LO}$ | Output Leakage Current | $V_{IN}$ = 0V to $V_{CC}$, $T_{AC}$ = 0°C to 135°C | | | | 1 | □A |
| $V_{IL}$ (1) | Input Low Voltage | | | -0.5 | | $0.3V_{CC}$ | V |
| $V_{IH}$ (1) | Input HIgh Voltage | | | $0.7V_{CC}$ | | $V_{CC}$ + 0.3 | V |
| $V_{OL}$ | Output Low Voltage | 2.3V < $V_{CC}$ < 3.6V | $I_{OL}$ = 100 µA | | | 0.2 | V |
| $V_{OH}$ | Output High Voltage | | $I_{OH}$ = -100 □A | $V_{CC}$ - 0.2 | | | V |

**Note 1. Maximum DC voltage on input or I/O pins is Vcc + 0.5V. During voltage transitions, input or I/O pins may overshoot Vcc by + 2.0 V for a period of time not to exceed 20ns. Minimum DC voltage on input or I/O pins is -0.5 V. During voltage transitions, input or I/O pins may undershoot GND by -2.0 V for a period of time not to exceed 20ns.**

### 9.4 AC MEASUREMENT CONDITIONS

| Symbol | Parameter | Min | Max | Units |
|---|---|---|---|---|
| CL | Load Capacitance up to 104MHz | | 30 | pF |
| CL | Load Capacitance up to 133MHz | | 15 | pF |
| TR,TF | Input Rise and Fall Times | | 5 | ns |
| VIN | Input Pulse Voltages | $0.2V_{CC}$ to $0.8V_{CC}$ | | V |
| VREFI | Input Timing Reference Voltages | $0.3V_{CC}$ to $0.7V_{CC}$ | | V |
| VREFO | Output Timing Reference Voltages | $0.5V_{CC}$ | | V |

**Figure9.1 Output test load & AC measurement I/O Waveform**



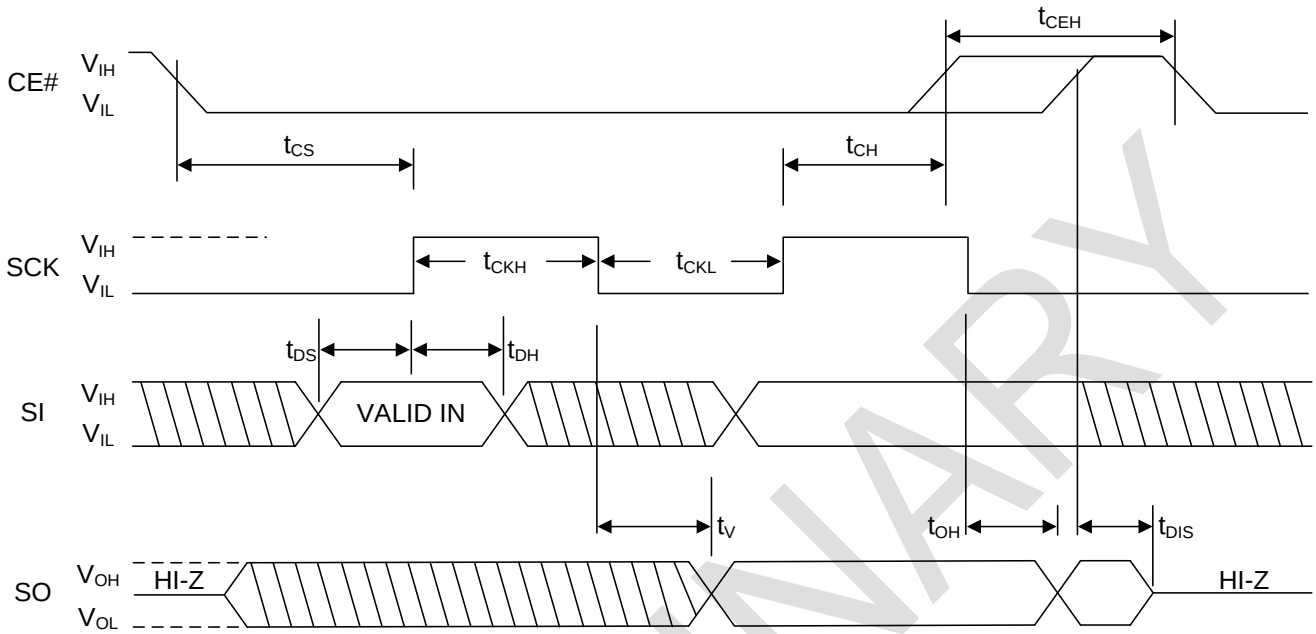
## 9.5 AC CHARACTERISTICS

(Under operating range, refer to section 9.4 for AC measurement conditions)

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $f_{CT}$ | Clock Frequency for fast read mode | 0 | | 133 | MHz |
| $f_C$ | Clock Frequency for read mode | 0 | | 50 | MHz |
| $f_{C2}$ | Clock Frequency for fast read DTR mode | 0 | | 66 | MHz |
| $f_{T2}$ | Clock Frequency for fast read Dual I/O DTR mode | 0 | | 66 | MHz |
| $f_{Q2}$ | Clock Frequency for fast read Quad I/O DTR mode | 0 | | 66 | MHz |
| $t_{RI}$ | Input Rise Time | | | 8 | ns |
| $t_{FI}$ | Input Fall Time | | | 8 | ns |
| $t_{CKH}$ | SCK High Time | 4 | | | ns |
| $t_{CKL}$ | SCK Low Time | 4 | | | ns |
| $t_{CEH}$ | CE# High Time | 7 | | | ns |
| $t_{CS}$ | CE# Setup Time | 5 | | | ns |
| $t_{CH}$ | CE# Hold Time | 5 | | | ns |
| $t_{DS}$ | Data In Setup Time | 2 | | | ns |
| $t_{DH}$ | Data in Hold Time | 2 | | | ns |
| $t_V$ | Output Valid @ 133MHz | | | 7 | ns |
| | Output Valid @ 104MHz | | | 8 | ns |
| $t_{OH}$ | Output Hold Time Normal Mode | 2 | | | ns |
| $t_{DIS}$ | Output Disable Time | | | 8 | ns |
| $t_{HD}$ | Output Hold Time | 2 | | | ns |
| $t_{HLCH}$ | HOLD Active Setup Time relative to SCK | 5 | | | ns |
| $t_{CHHH}$ | HOLD Active Hold Time relative to SCK | 5 | | | ns |
| $t_{HHCH}$ | HOLD Not Active Setup Time relative to SCK | 5 | | | ns |
| $t_{CHHL}$ | HOLD Not Active Hold Time relative to SCK | 5 | | | ns |
| $t_{LZ}$ | HOLD to Output Low Z | | | 12 | ns |
| $t_{HZ}$ | HOLD to Output High Z | | | 12 | ns |
| $t_{EC}$ | Sector Erase Time | | 50 | 150 | ms |
| | Block Erase Time (32Kbyte) | | 0.25 | 0.75 | s |
| | Block Erase time (64Kbyte) | | 0.5 | 1.5 | s |
| | Chip Erase Time (032Mb) | | 45 | 90 | s |
| $t_{PP}$ | Page Program Time | | 0.3 | 1.0 | ms |
| $t_{VCE}$ | Vcc(min) to CE# Low | 1 | | | ms |
| $t_{res1}$ | Release deep power down | | | 3 | µs |
| $t_{DP}$ | Deep power down | | | 3 | µs |
| $t_W$ | Write Status Register time | | 10 | 15 | ms |
| $t_{SUS}$ | Suspend to read ready | | | 100 | µs |
| $t_{SRST}$ | Software Reset cover time | 20 | | 100 | µs |

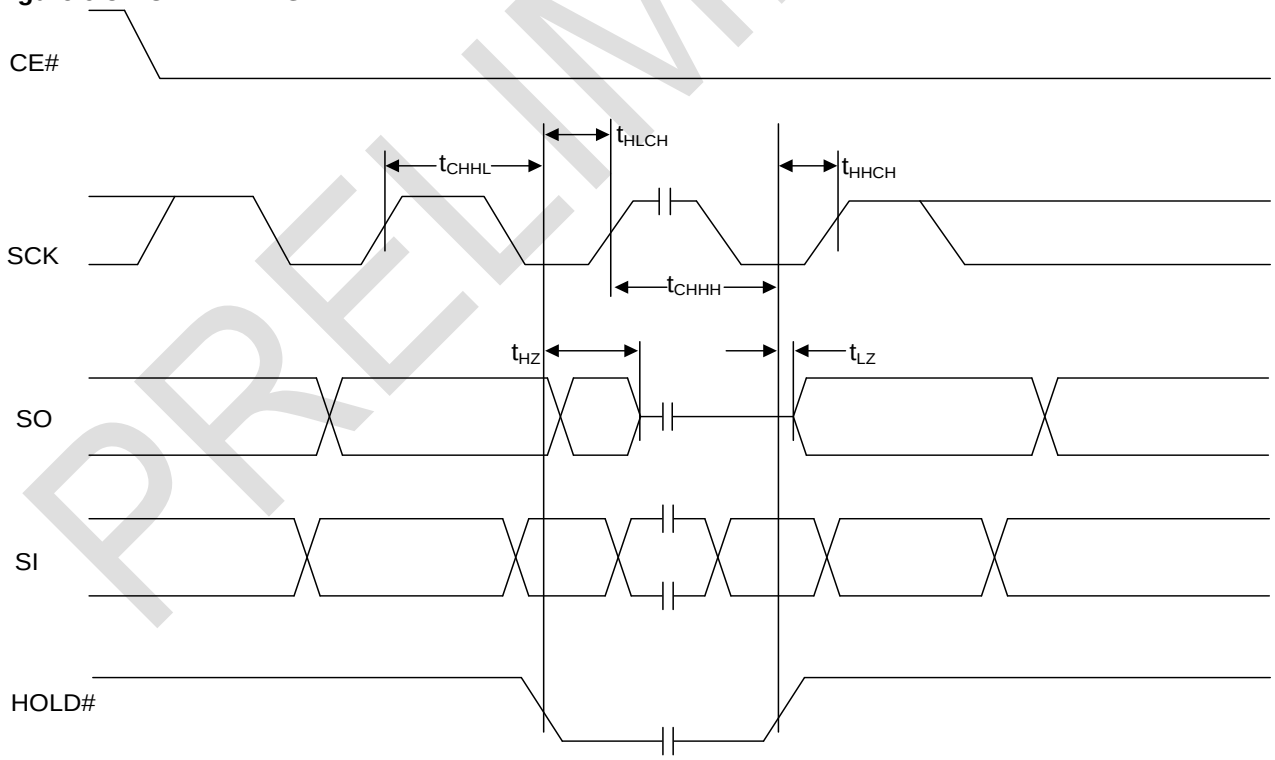
### 9.6 SERIAL INPUT/OUTPUT TIMING (1)

**Figure 9.2 SERIAL INPUT/OUTPUT TIMING**




**Note1. For SPI Mode 0 (0,0)**

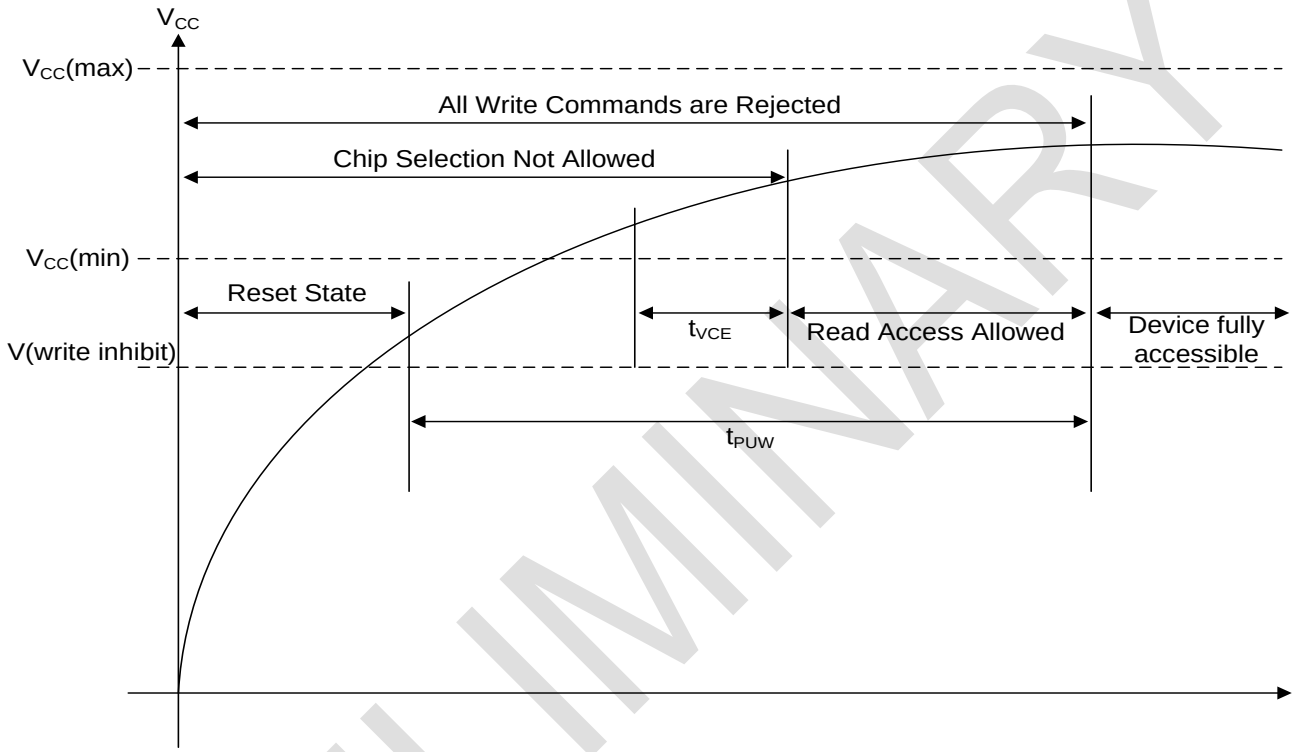
**Figure 9.3 HOLD TIMING**



### 9.7 POWER-UP AND POWER-DOWN

At Power-up and Power-down, the device must be NOT SELECTED until Vcc reaches at the right level. (Adding a simple pull-up resistor on CE# is recommended.)

**Power up timing**




| Symbol | Parameter | Min. | Max | Unit |
|---|---|---|---|---|
| tVCE(1) | Vcc(min) to CE# Low | 1 | | ms |
| tPUW(1) | Power-up time delay to write instruction | 1 | 10 | ms |
| $V_{WI}$(1) | Write Inhibit Voltage | | 2.1 | V |

**Note1. These parameters are characterized and are not 100% tested.**

### 9.8 PROGRAM/ERASE PERFORMANCE

| Parameter | Unit | Typ | Max |
|---|---|---|---|
| Sector Erase Time | ms | 50 | 200 |
| Block Erase Time 32Kbyte | s | 0.25 | 0.75 |
| Block Erase Time 64Kbyte | s | 0.5 | 1.5 |
| Chip Erase Time (032Mb) | s | 45 | 90 |
| Page Programming Time | ms | 0.3 | 1.0 |
| Byte Program | µs | TBD | TBD |

**Note : These parameters are characterized and are not 100% tested.**

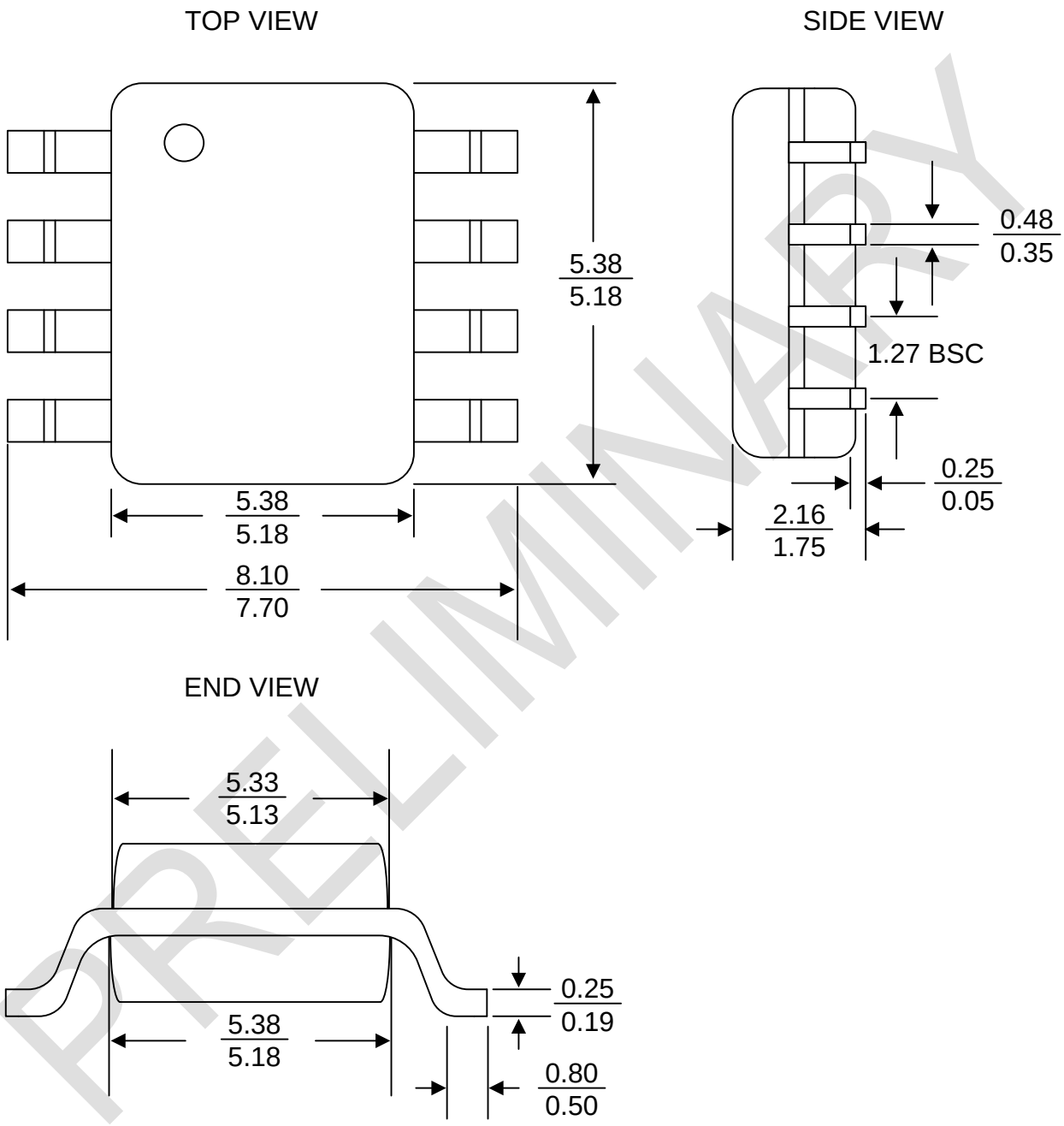
9.9 RELIABILITY CHARACTERISTICS

| Parameter | Min | Unit | Test Method |
|---|---|---|---|
| Endurance | 100,000 | Cycles | JEDEC Standard A117 |
| Data Retention | 20 | Years | JEDEC Standard A103 |
| ESD – Human Body Model | 2,000 | Volts | JEDEC Standard A114 |
| ESD – Machine Model | 200 | Volts | JEDEC Standard A115 |
| Latch-Up | 100 + ICC1 | mA | JEDEC Standard 78 |

**Note : These parameters are characterized and are not 100% tested.**

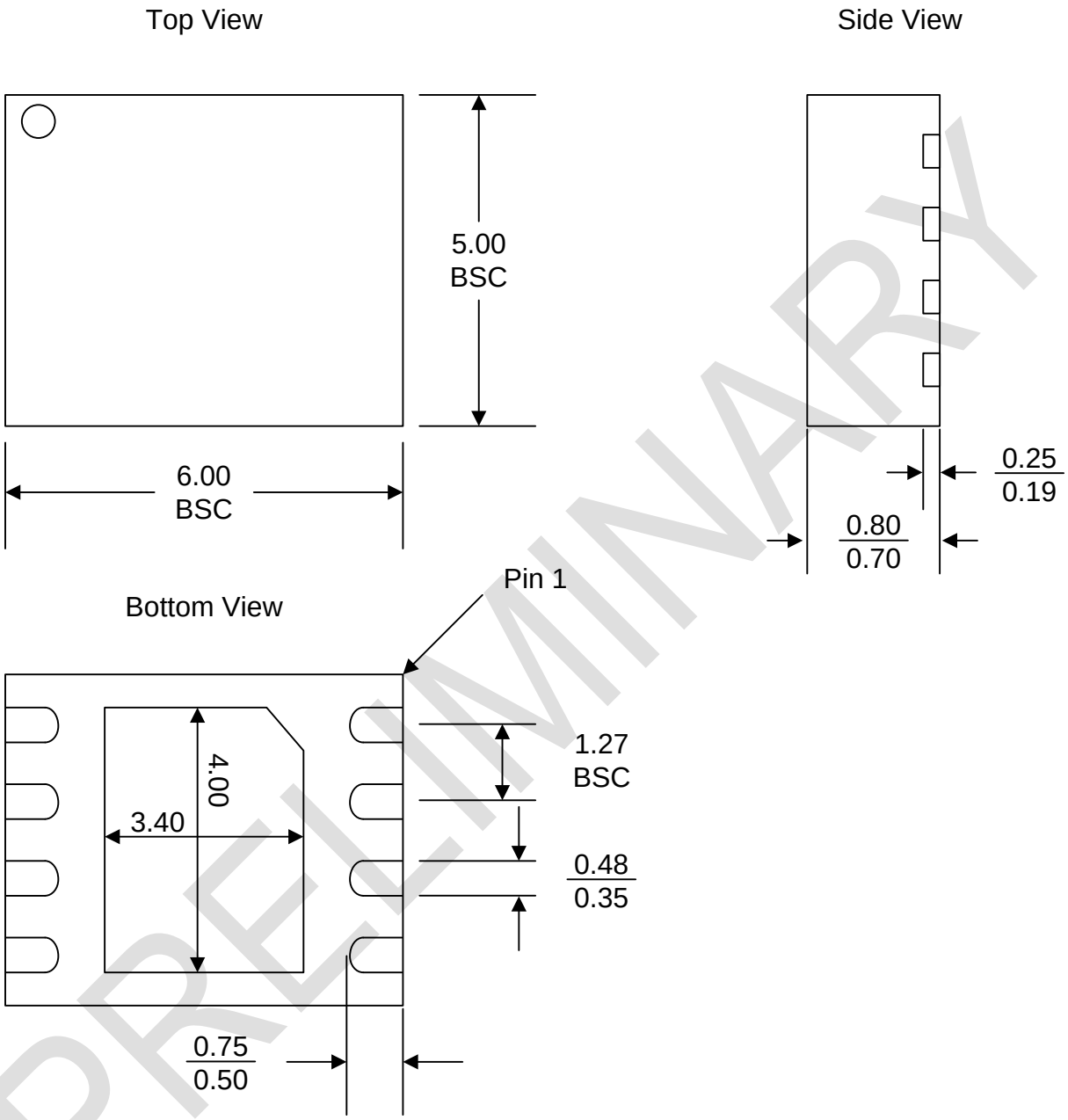
# 10. PACKAGE TYPE INFORMATION

## 10.1 8-PIN JEDEC 208MIL BROAD SMALL OUTLINE INTEGRATED CIRCUIT (SOIC) PACKAGE (JB) (1)




**Note1. All dimensions are in millimeters.**

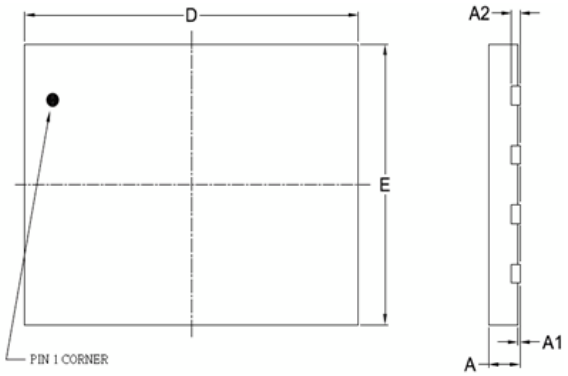
## 10.2 8-CONTACT ULTRA-THIN SMALL OUTLINE NO-LEAD (WSON) PACKAGE 6X5MM (JK) (1)

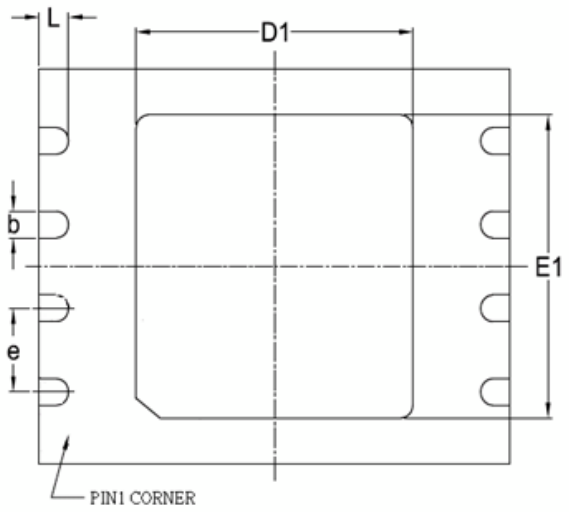





**Note1. All dimensions are in millimeters.**

## 10.3 8-CONTACT ULTRA-THIN SMALL OUTLINE NO-LEAD (WSON) PACKAGE 8X6MM (JL) (1)

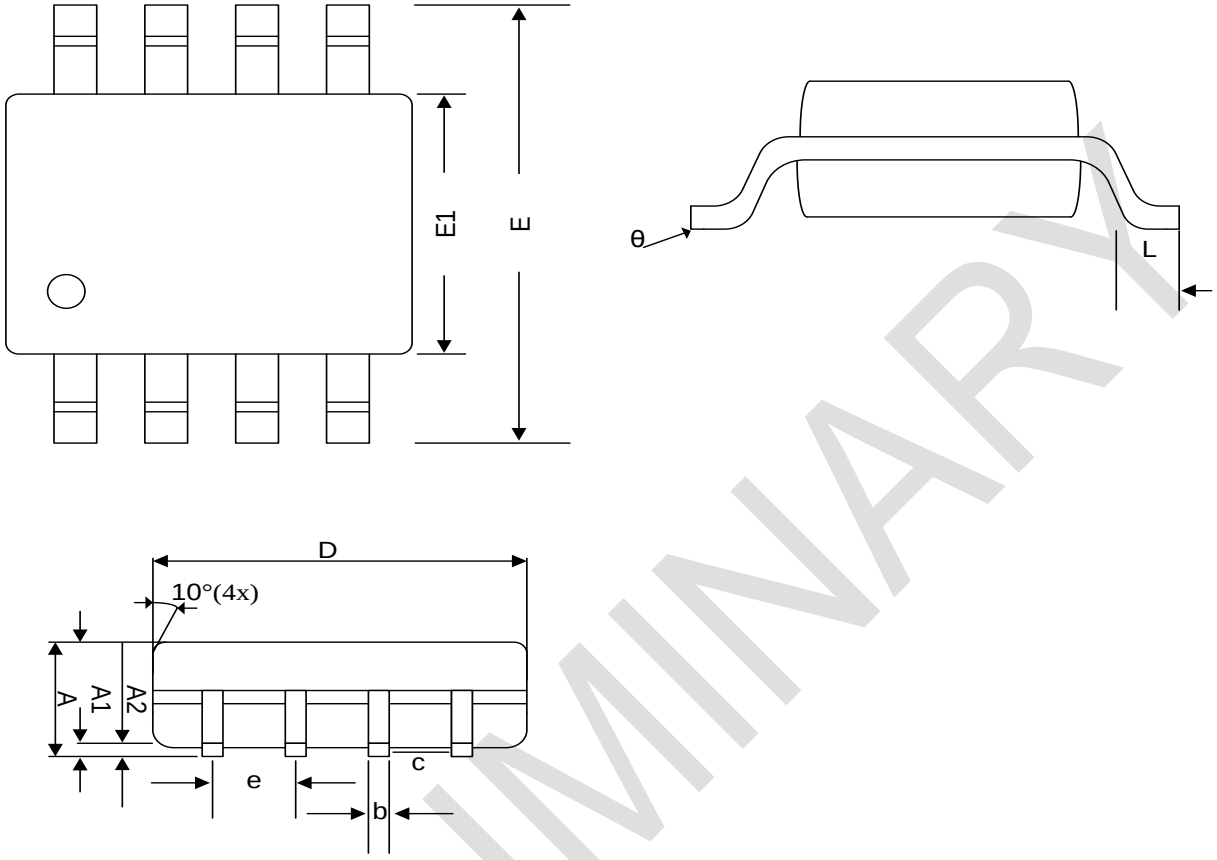



| SYMBOL | DIMENSION IN MM | | |
|---|---|---|---|
| | MIN. | NOM | MAX |
| A | 0.70 | 0.75 | 0.80 |
| A1 | 0.00 | 0.02 | 0.05 |
| A2 | - - - | 0.20 | - - - |
| D | 7.90 | 8.00 | 8.10 |
| E | 5.90 | 6.00 | 6.10 |
| D1 | 4.65 | 4.70 | 4.75 |
| E1 | 4.55 | 4.60 | 4.65 |
| e | - - - | 1.27 | - - - |
| b | 0.35 | 0.40 | 0.48 |
| L | 0.4 | 0.50 | 0.60 |

**Note1. All dimensions are in millimeters.**

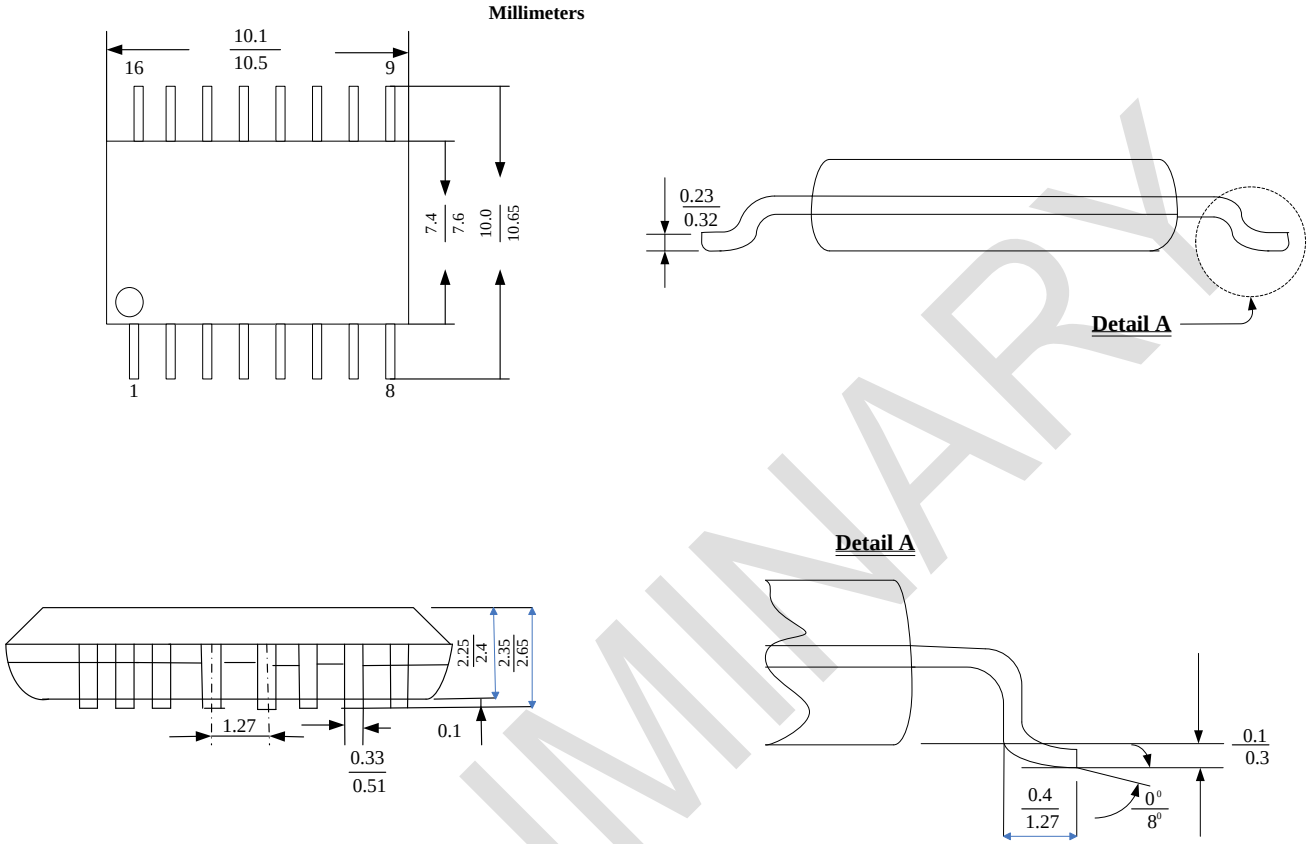
## 10.4 8-PIN 208MIL VSOP PACKAGE (JF) (1)




| Symbols | Min | Typ | Max |
|---|---|---|---|
| A | - | - | 1 |
| A1 | 0.05 | 0.1 | 0.15 |
| A2 | 0.75 | 0.8 | 0.85 |
| b | 0.35 | 0.42 | 0.48 |
| c | - | .127 REF | - |
| D | 5.18 | 5.28 | 5.38 |
| E | 7.7 | 7.9 | 8.1 |
| E1 | 5.18 | 5.28 | 5.38 |
| e | - | 1.27 | - |
| L | 0.5 | 0.65 | 0.8 |
| y | - | - | 0.1 |
| θ | 0° | - | 8° |

**Note1. All dimensions are in millimeters.**

## 10.5 16-LEAD PLASTIC SMALL OUTLINE PACKAGE (300 MILS BODY WIDTH) (JM) (1)




**Note1. All dimensions are in millimeters.**

## 11. ORDERING INFORMATION

| Density | Frequency (MHz) | Order Part Number | Package |
|---|---|---|---|
| 64Mb | 133 | IS25LP064-JBLE | 8-pin SOIC 208mil |
| | | IS25LP064-JKLE | 8-pin WSON (6x5 mm) |
| | | IS25LP064-JLLE | 8-pin WSON (6x8 mm) |
| | | IS25LP064-JFLE | 8-pin VSOP 208mil |
| | | IS25LP064-JMLE | 16-pin 300mil |
| | | IS25LP064-JBLA* | 8-pin SOIC 208mil (Call Factory) |
| | | IS25LP064-JKLA* | 8-pin WSON (6x5 mm) (Call Factory) |
| | | IS25LP064-JLLA* | 8-pin WSON (6x8 mm) (Call Factory) |
| | | IS25LP064-JFLA* | 8-pin VSOP 208mil (Call Factory) |
| | | IS25LP064-JMLA* | 16-pin 300mil (Call Factory) |
| | | IS25LP064-JWLE | KGD (Call Factory) |

| Density | Frequency (MHz) | Order Part Number | Package |
|---|---|---|---|
| 32Mb | 133 | IS25LP032-JBLE | 8-pin SOIC 208mil |
| | | IS25LP032-JKLE | 8-pin WSON (6x5 mm) |
| | | IS25LP032-JLLE | 8-pin WSON (6x8 mm) |
| | | IS25LP032-JFLE | 8-pin VSOP 208mil |
| | | IS25LP032-JMLE | 16-pin 300mil |
| | | IS25LP032-JBLA* | 8-pin SOIC 208mil (Call Factory) |
| | | IS25LP032-JKLA* | 8-pin WSON (6x5 mm) (Call Factory) |
| | | IS25LP032-JLLA* | 8-pin WSON (6x8 mm) (Call Factory) |
| | | IS25LP032-JFLA* | 8-pin VSOP 208mil (Call Factory) |
| | | IS25LP032-JMLA* | 16-pin 300mil (Call Factory) |
| | | IS25LP032-JWLE | KGD (Call Factory) |

**A* = A1, A2, A3 Automotive Temperature Range**

**E = -40 to 105 C**
**A1= -40 to 85 C**
**A2= -40 to 105 C**
**A3= -40 to 125 C**