EN25QH64A
64 Megabit 3.0V Serial Flash Memory with 4Kbyte Uniform Sector

FEATURES

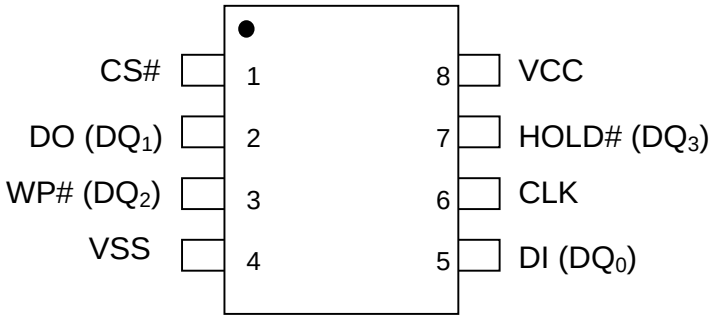
- Single power supply operation
- Full voltage range: 2.7-3.6 volt
- Serial Interface Architecture
- SPI Compatible: Mode 0 and Mode 3
- 64 M-bit Serial Flash
- 64 M-bit / 8,192 KByte / 32,768 pages
- 256 bytes per programmable page
- Standard or Quad SPI
- Standard SPI: CLK, CS#, DI, DO, WP#, HOLD#
- Quad SPI: CLK, CS#, DQ_0, DQ_1, DQ_2, DQ_3
- High performance
- Normal read
  - 83MHz
- Fast read
  - Standard SPI: 104MHz with 1 dummy bytes
  - Quad SPI: 104MHz with 3 dummy bytes
- Burst Modes
- 16/32/64/128 linear burst with wrap-around
- Quad SPI: 104MHz with 3 dummy bytes
- Low power consumption
- 5 mA typical active current
- 1μA typical power down current
- Uniform Sector Architecture:
- 2048 sectors of 4-Kbyte
- 256 blocks of 32-Kbyte
- 128 blocks of 64-Kbyte
- Any sector or block can be erased individually
- Software and Hardware Write Protection:
- Write Protect all or portion of memory via software
- Enable/Disable protection with WP# pin
- High performance program/erase speed
- Page program time: 0.7ms typical
- Sector erase time: 50ms typical
- 32KB Block erase time 200ms typical
- 64KB Block erase time 350ms typical
- Chip erase time: 30 seconds typical
- Write Suspend and Write Resume
- Lockable 512 byte OTP security sector
- Support Serial Flash Discoverable Parameters (SFDP) signature
- Read Unique ID Number
- Minimum 100K endurance cycle
- Package Options
- 8 pins SOP 200mil body width
- 8 contact VDFN (5x6mm)
- 8 pins PDIP
- 16 pins SOP 300mil body width
- 24 balls TFBGA (6x8mm)
- All Pb-free packages are compliant RoHS, Halogen-Free and REACH.
- Industrial temperature Range

GENERAL DESCRIPTION

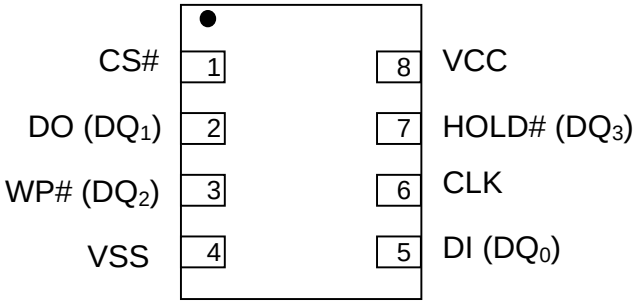
The EN25QH64A is a 64 Megabit (8,192K-byte) Serial Flash memory, with advanced write protection mechanisms. The EN25QH64A supports the standard Serial Peripheral Interface (SPI), and a high performance, Quad I/O using SPI pins: Serial Clock, Chip Select, Serial DQ_0 (DI) and DQ_1(DO), DQ_2(WP#) and DQ_3(HOLD#). SPI clock frequencies of up to 104MHz are supported allowing equivalent clock rates of 416MHz (104MHz x 4) for Quad Output while using the Quad Output Read instructions. The memory can be programmed 1 to 256 bytes at a time, using the Page Program or Quad Page Program instruction.

The EN25QH64A is designed to allow either single Sector/Block at a time or full chip erase operation. The EN25QH64A can be configured to protect part of the memory as the software protected mode. The device can sustain a minimum of 100K program/erase bytes on each sector or block.

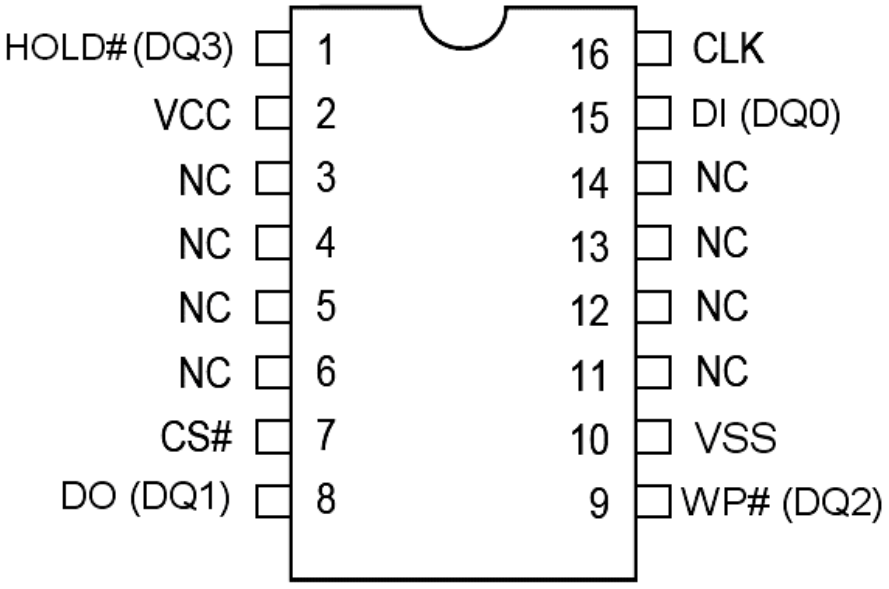
Figure.1 CONNECTION DIAGRAMS



8 - LEAD SOP / PDIP

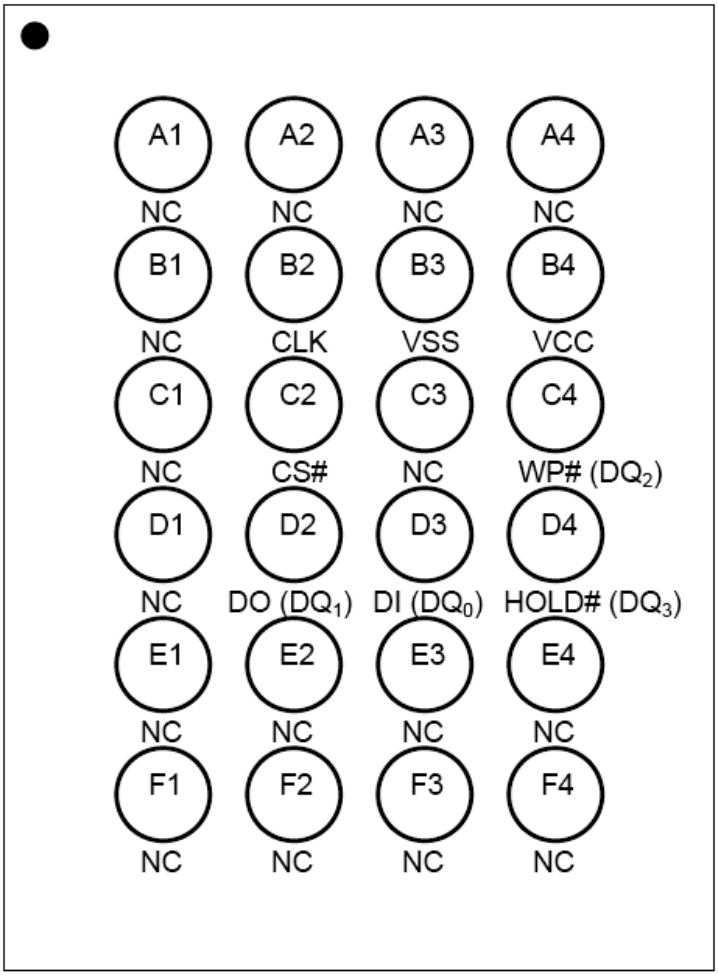


8 - LEAD VDFN



16 - LEAD SOP

Top View, Balls Facing Down



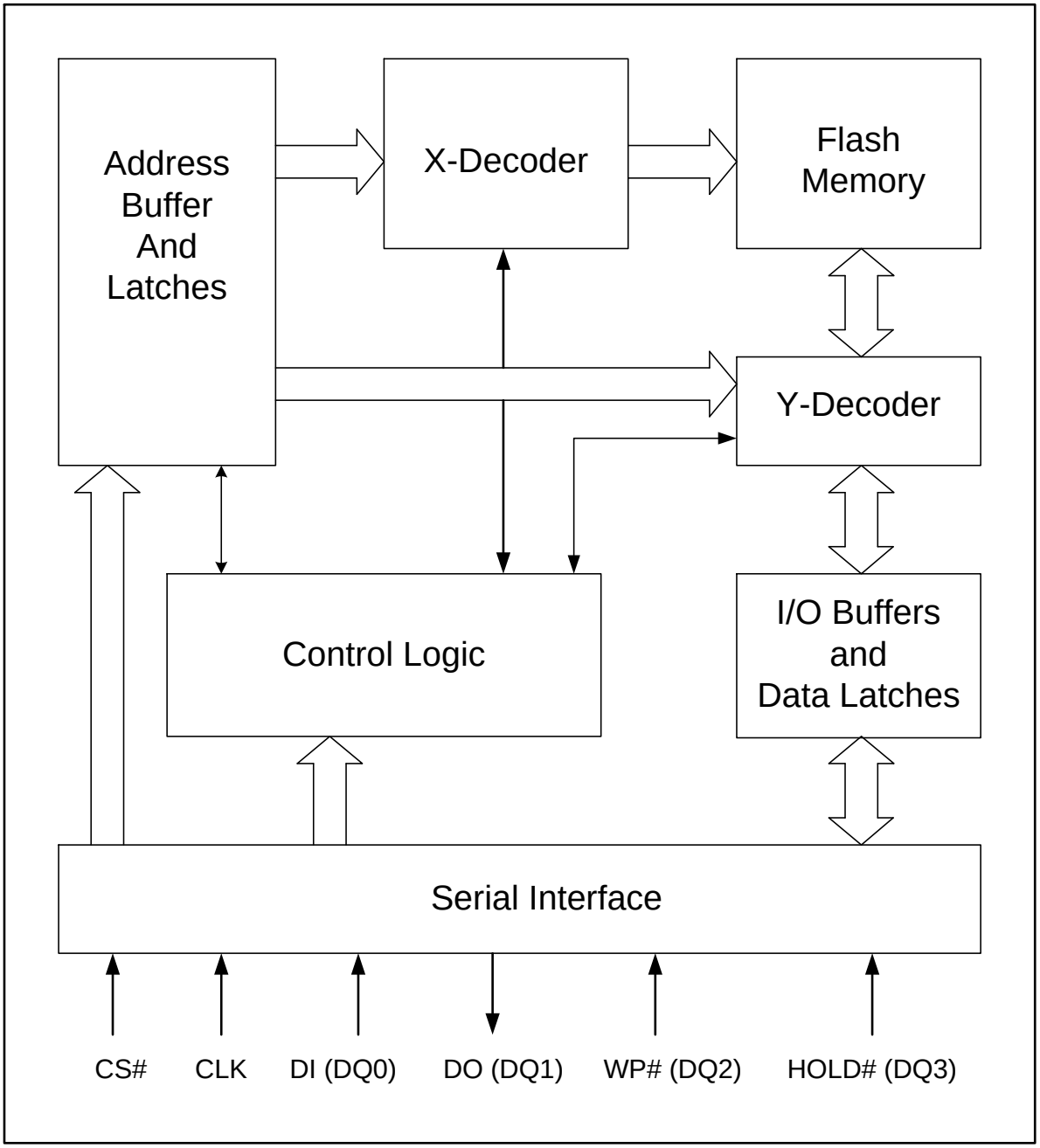
24 - Ball TFBGA

Table 1. Pin Names

Symbol	Pin Name
CLK	Serial Clock Input
DI (DQ_0)	Serial Data Input (Data Input Output 0) *1
DO (DQ_1)	Serial Data Output (Data Input Output 1) *1
CS#	Chip Enable
WP# (DQ_2)	Write Protect (Data Input Output 2) *2
HOLD# (DQ_3)	HOLD# pin (Data Input Output 3) *2
Vcc	Supply Voltage (2.7-3.6V)
Vss	Ground
NC	No Connect

Note:
1. DQ_0 ~ DQ_3 are used for Quad instructions.

Figure 2. BLOCK DIAGRAM



Note:
1. DQ_0 ~ DQ_3 are used for Quad instructions.

SIGNAL DESCRIPTION

Serial Data Input, Output and IOs (DI, DO and DQ_0, DQ_1, DQ_2, DQ_3)

The EN25QH64A support standard SPI, Dual SPI and Quad SPI operation. Standard SPI instructions use the unidirectional DI (input) pin to serially write instructions, addresses or data to the device on the rising edge of the Serial Clock (CLK) input pin. Standard SPI also uses the unidirectional DO (output) to read data or status from the device on the falling edge CLK.

Dual and Quad SPI instruction use the bidirectional IO pins to serially write instruction, addresses or data to the device on the rising edge of CLK and read data or status from the device on the falling edge of CLK.

Serial Clock (CLK)

The SPI Serial Clock Input (CLK) pin provides the timing for serial input and output operations. ("See SPI Mode")

Chip Select (CS#)

The SPI Chip Select (CS#) pin enables and disables device operation. When CS# is high the device is deselected and the Serial Data Output (DO, or DQ_0, DQ_1, DQ_2 and DQ_3) pins are at high impedance. When deselected, the devices power consumption will be at standby levels unless an internal erase, program or status register cycle is in progress. When CS# is brought low the device will be selected, power consumption will increase to active levels and instructions can be written to and data read from the device. After power-up, CS# must transition from high to low before a new instruction will be accepted.

Hold (HOLD#)

The HOLD# pin allows the device to be paused while it is actively selected. When HOLD# is brought low, while CS# is low, the DO pin will be at high impedance and signals on the DI and CLK pins will be ignored (don't care). The hold function can be useful when multiple devices are sharing the same SPI signals. The HOLD# function is only available for standard SPI and Dual SPI operation, when during Quad SPI, this pin is the Serial Data IO (DQ_3) for Quad I/O operation.

Write Protect (WP#)

The Write Protect (WP#) pin can be used to prevent the Status Register from being written. Used in conjunction with the Status Register's Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits and Status Register Protect (SRP) bits, a portion or the entire memory array can be hardware protected. The WP# function is only available for standard SPI and Dual SPI operation, when during Quad SPI, this pin is the Serial Data IO (DQ_2) for Quad I/O operation.

MEMORY ORGANIZATION

The memory is organized as:

- 8,388,608 bytes
- Uniform Sector Architecture
  128 blocks of 64-Kbyte
  256 blocks of 32-Kbyte
  2,048 sectors of 4-Kbyte
  32,768 pages (256 bytes each)

Each page can be individually programmed (bits are programmed from 1 to 0). The device is Sector, Block or Chip Erasable but not Page Erasable.

Table 2. Uniform Block Sector Architecture

64K Block	32K Block	Sector	Address range	
127	255	2047	7FF000h	7FFFFFh
	⋮	⋮	⋮	⋮
	254	2032	7F0000h	7F0FFFh
126	253	2031	7EF000h	7EFFFFh
	⋮	⋮	⋮	⋮
	252	2016	7E0000h	7E0FFFh
125	251	2015	7DF000h	7DFFFFh
	⋮	⋮	⋮	⋮
	250	2000	7D0000h	7D0FFFh
⋮	⋮	⋮	⋮	⋮
114	229	1839	72F000h	72FFFFh
	⋮	⋮	⋮	⋮
	228	1824	720000h	720FFFh
113	227	1823	71F000h	71FFFFh
	⋮	⋮	⋮	⋮
	226	7952	1F10000h	1F10FFFh
112	225	7951	1F0F000h	1F0FFFFh
	⋮	⋮	⋮	⋮
	224	1972	700000h	700FFFh

64K Block	32K Block	Sector	Address range	
111	223	1791	6FF000h	6FFFFFh
	⋮	⋮	⋮	⋮
	222	1776	6F0000h	6F0FFFh
110	221	1775	6EF000h	6EFFFFh
	⋮	⋮	⋮	⋮
	220	1760	6E0000h	6E0FFFh
109	219	1759	6DF000h	6DFFFFh
	⋮	⋮	⋮	⋮
	218	1744	6D0000h	6D0FFFh
⋮	⋮	⋮	⋮	⋮
98	197	1583	62F000h	62FFFFh
	⋮	⋮	⋮	⋮
	196	1568	620000h	620FFFh
97	195	1567	61F000h	61FFFFh
	⋮	⋮	⋮	⋮
	194	1552	610000h	610FFFh
96	193	1551	60F000h	60FFFFh
	⋮	⋮	⋮	⋮
	192	1536	600000h	600FFFh

64K Block	32K Block	Sector	Address range	
31	63	511	01FF000h	01FFFFFh
	⋮	⋮	⋮	⋮
	62	496	01F0000h	01F0FFFh
30	61	495	01EF000h	01EFFFFh
	⋮	⋮	⋮	⋮
	60	480	01E0000h	01E0FFFh
29	59	479	01DF000h	01DFFFFh
	⋮	⋮	⋮	⋮
	58	464	01D0000h	01D0FFFh
⋮	⋮	⋮	⋮	⋮
18	37	303	012F000h	012FFFFh
	⋮	⋮	⋮	⋮
	36	288	0120000h	0120FFFh
17	35	287	011F000h	011FFFFh
	⋮	⋮	⋮	⋮
	34	272	0110000h	0110FFFh
16	33	271	010F000h	010FFFFh
	⋮	⋮	⋮	⋮
	32	256	0100000h	0100FFFh

64K Block	32K Block	Sector	Address range	
15	31	255	00FF000h	00FFFFFh
	⋮	⋮	⋮	⋮
	30	240	00F0000h	00F0FFFh
14	29	239	00EF000h	00EFFFFh
	⋮	⋮	⋮	⋮
	28	224	00E0000h	00E0FFFh
13	27	223	00DF000h	00DFFFFh
	⋮	⋮	⋮	⋮
	26	208	00D0000h	00D0FFFh
⋮	⋮	⋮	⋮	⋮
2	5	47	002F000h	002FFFFh
	⋮	⋮	⋮	⋮
	4	32	0020000h	0020FFFh
1	3	31	001F000h	001FFFFh
	⋮	⋮	⋮	⋮
	2	16	0010000h	0010FFFh
0	1	15	000F000h	000FFFFh
	⋮	⋮	⋮	⋮
	0	0	0000000h	0000FFFh

SPI Clock Mode

Single Data Rate (SDR)

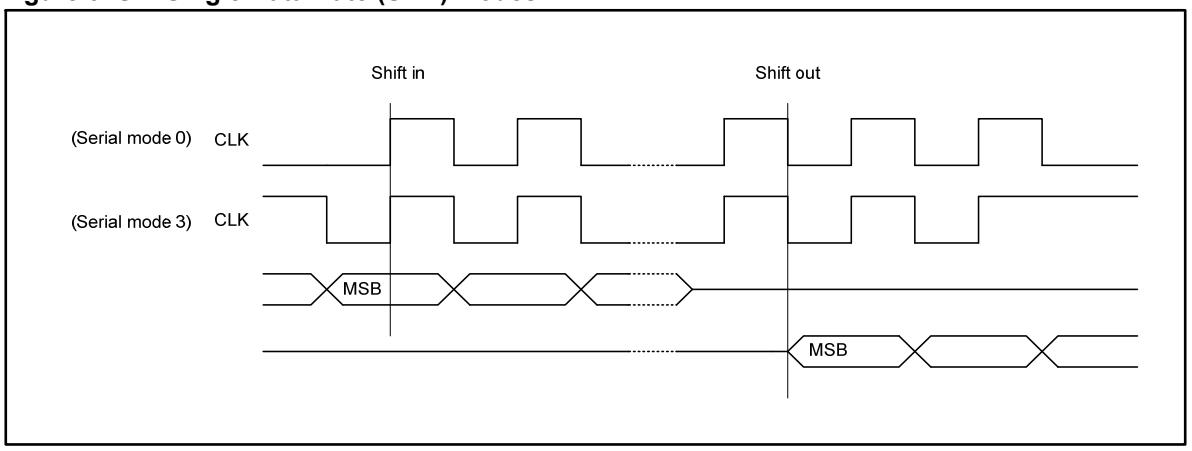
The EN25QH64A device can be driver by an embedded microcontroller (bus master) in either of the two following clocking modes, Mode 0 and Mode 3.

For these two modes, input data into the device is always latched in on the rising edge of the CLK signal and the output data is always available from the falling edge of the CLK clock signal.

Timing diagrams throughout the remainder of the document are generally shown as both mode 0 and 3 by showing CLK as both high and low at the fall of CS# so no CLK rising edge set up or hold time to the falling edge of CS# is needed for mode 3.

CLK cycles are measured (counted) form one falling edge of CLK to next falling edge of CLK. In mode 0 the beginning of the first CLK cycle in a command is measured from the falling edge of CS# to the first falling edge of CLK because CLK is already low at the beginning of a command.

Figure 3. SPI Single Data Rate (SDR) Modes

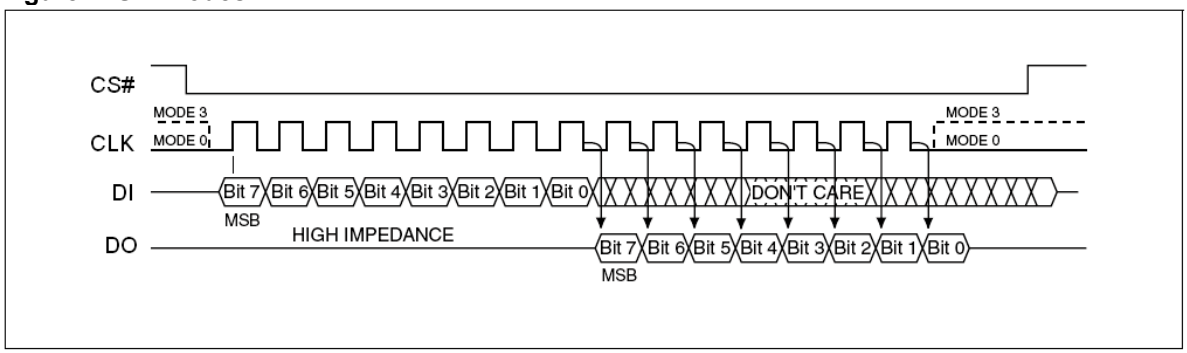


OPERATING FEATURES

Standard SPI Modes

The EN25QH64A is accessed through an SPI compatible bus consisting of four signals: Serial Clock (CLK), Chip Select (CS#), Serial Data Input (DI) and Serial Data Output (DO). Both SPI bus operation Modes 0 (0,0) and 3 (1,1) are supported. The primary difference between Mode 0 and Mode 3, as shown in Figure 4, concerns the normal state of the CLK signal when the SPI bus master is in standby and data is not being transferred to the Serial Flash. For Mode 0 the CLK signal is normally low. For Mode 3 the CLK signal is normally high. In either case data input on the DI pin is sampled on the rising edge of the CLK. Data output on the DO pin is clocked out on the falling edge of CLK.

Figure 4. SPI Modes



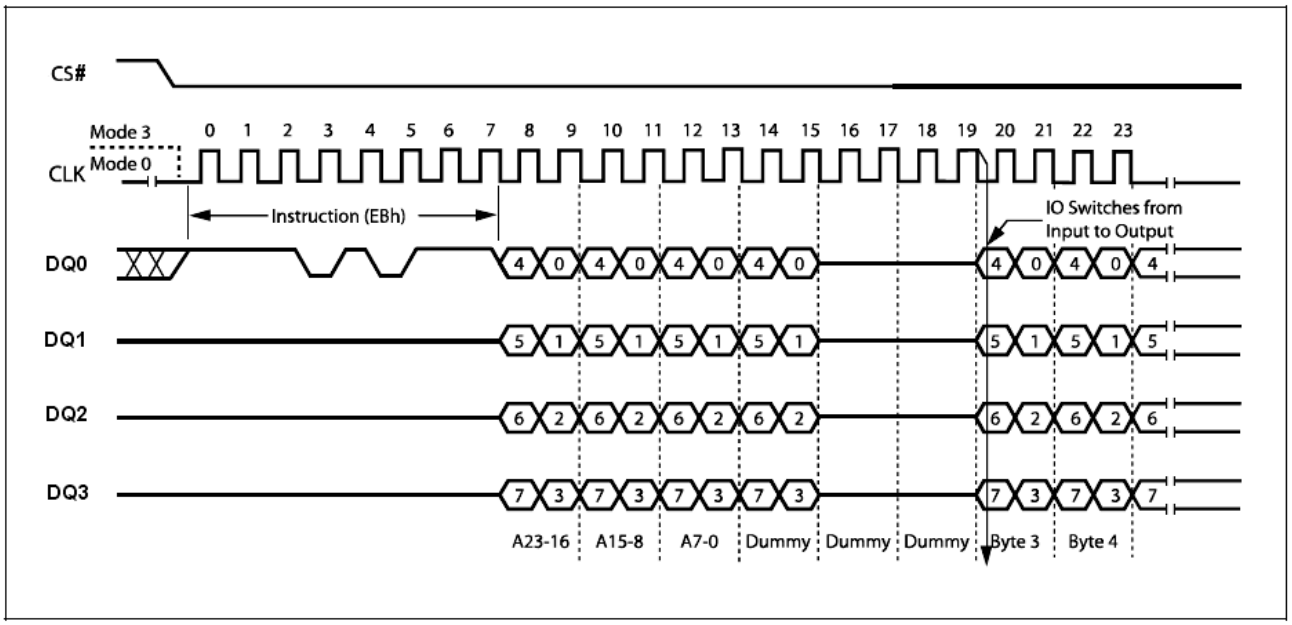
Dual SPI Instruction

The EN25QH64A supports Dual SPI operation when using the “Dual Output Fast Read and Dual I/O Fast Read “ (3Bh and BBh) instructions. These instructions allow data to be transferred to or from the Serial Flash memory at two to three times the rate possible with the standard SPI. The Dual Read instructions are ideal for quickly downloading code from Flash to RAM upon power-up (code-shadowing) or for application that cache code-segments to RAM for execution. The Dual output feature simply allows the SPI input pin to also serve as an output during this instruction. When using Dual SPI instructions the DI and DO pins become bidirectional I/O pins; DQ_0 and DQ_1. All other operations use the standard SPI interface with single output signal.

Quad I/O SPI Modes

The EN25QH64A supports Quad input / output operation when using the Quad I/O Fast Read (EBh).This instruction allows data to be transferred to or from the Serial Flash memory at four to six times the rate possible with the standard SPI. The Quad Read instruction offer a significant improvement in continuous and random access transfer rates allowing fast code-shadowing to RAM or for application that cache code-segments to RAM for execution. When using Quad I/O SPI instructions, the DI and DO pins become bidirectional I/O pins; DQ_0 and DQ_1, and the WP# and HOLD# pins become DQ_2 and DQ_3 respectively.

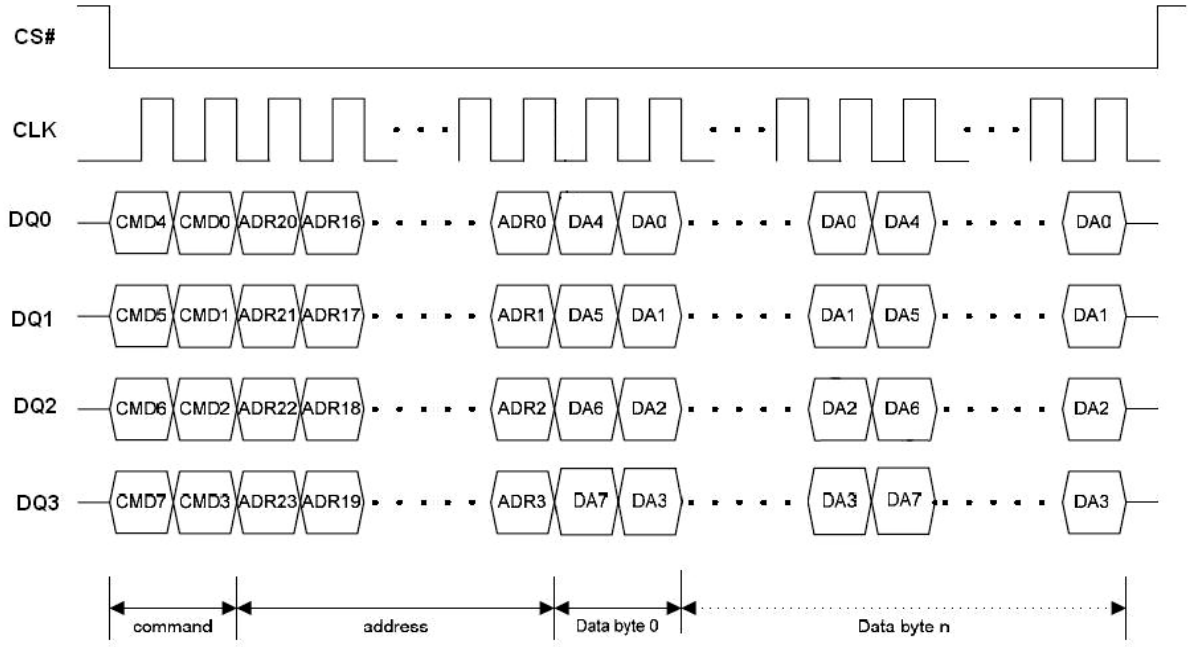
Figure 5. Quad I/O SPI Modes



Full Quad SPI Modes

The EN25QH64A also supports full Quad Mode function while using the Enable Quad Peripheral Interface mode (EQPI) (38h). When using Quad SPI instruction the DI and DO pins become bidirectional I/O pins; DQ_0 and DQ_1, and the WP# and HOLD# pins become DQ_2 and DQ_3 respectively.

Figure 6. Full Quad SPI Modes



Page Programming

To program one data byte, two instructions are required: Write Enable (WREN), which is one byte, and a Page Program (PP) or Quad Input Page Program (QPP) sequence, which consists of four bytes plus data. This is followed by the internal Program cycle (of duration t_{PP}).
To spread this overhead, the Page Program (PP) or Quad Input Page Program (QPP) instruction allows up to 256 bytes to be programmed at a time (changing bits from 1 to 0) provided that they lie in consecutive addresses on the same page of memory.

Sector Erase, Half Block Erase, Block Erase and Chip Erase

The Page Program (PP) or Quad Input Page Program (QPP) instruction allows bits to be reset from 1 to 0. Before this can be applied, the bytes of memory need to have been erased to all 1s (FFh). This can be achieved a sector at a time, using the Sector Erase (SE) instruction, half a block at a time using the Half Block Erase (HBE) instruction, a block at a time using the Block Erase (BE) instruction or throughout the entire memory, using the Chip Erase (CE) instruction. This starts an internal Erase cycle (of duration $t_{SE,}$ $t_{HBE,,}$ t_{BE} or t_{CE}). The Erase instruction must be preceded by a Write Enable (WREN) instruction.

Polling During a Write, Program or Erase Cycle

A further improvement in the time to Write Status Register (WRSR), Program (PP, QPP) or Erase (SE, HBE, BE or CE) can be achieved by not waiting for the worst case delay (t_W, t_{PP}, t_{SE}, t_{HBE}, t_{BE} or t_{CE}). The Write In Progress (WIP) bit is provided in the Status Register so that the application program can monitor its value, polling it to determine when the previous Write cycle, Program cycle or Erase cycle is complete.

Active Power, Stand-by Power and Deep Power-Down Modes

When Chip Select (CS#) is Low, the device is enabled, and in the Active Power mode. When Chip Select (CS#) is High, the device is disabled, but could remain in the Active Power mode until all internal bytes have completed (Program, Erase, Write Status Register). The device then goes into the Stand-by Power mode. The device consumption drops to I_{CC1}.
The Deep Power-down mode is entered when the specific instruction (the Enter Deep Power-down Mode (DP) instruction) is executed. The device consumption drops further to I_{CC2}. The device remains in this mode until another specific instruction (the Release from Deep Power-down Mode and Read Device ID (RDI) instruction) is executed.
All other instructions are ignored while the device is in the Deep Power-down mode. This can be used as an extra software protection mechanism, when the device is not in active use, to protect the device from inadvertent Write, Program or Erase instructions.

Write Protection

Applications that use non-volatile memory must take into consideration the possibility of noise and other adverse system conditions that may compromise data integrity. To address this concern the EN25QH64A provides the following data protection mechanisms:

- Power-On Reset and an internal timer (t_{PUW}) can provide protection against inadvertent changes while the power supply is outside the operating specification.
- Program, Erase and Write Status Register instructions are checked that they consist of a number of clock pulses that is a multiple of eight, before they are accepted for execution.
- All instructions that modify data must be preceded by a Write Enable (WREN) instruction to set the Write Enable Latch (WEL) bit. This bit is returned to its reset state by the following events:
  - Power-up
  - Write Disable (WRDI) instruction completion or Write Status Register (WRSR) instruction completion or Page Program (PP), Quad Input Page Program (QPP) instruction completion or Sector Erase (SE) instruction completion or Half Block Erase (HBE) / Block Erase (BE) instruction completion or Chip Erase (CE) instruction completion
- The Block Protect (BP3, BP2, BP1, BP0) bits or Dynamic Block Protect (DBP3, DBP2, DBP1, DBP0) bits allow part of the memory to be configured as read-only. This is the Software Protected Mode (SPM).
- The Write Protect (WP#) signal allows the Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits and Status Register Protect (SRP) bit to be protected. This is the Hardware Protected Mode (HPM).
- In addition to the low power consumption feature, the Deep Power-down mode offers extra software protection from inadvertent Write, Program and Erase instructions, as all instructions are ignored except one particular instruction (the Release from Deep Power-down instruction).

Table 3. Protected Area Sizes Sector Organization

Status Register Content					Memory Content			
T/B Bit	SR.5 Bit	SR.4 Bit	SR.3 Bit	SR.2 Bit	Protect Areas	Addresses	Density(KB)	Portion
0	0	0	0	0	None	None	None	None
0	0	0	0	1	Block 127	7F0000h-7FFFFFh	64KB	Upper 1/128
0	0	0	1	0	Block 126 to 127	7E0000h-7FFFFFh	128KB	Upper 2/128
0	0	0	1	1	Block 124 to 127	7C0000h-7FFFFFh	256KB	Upper 4/128
0	0	1	0	0	Block 120 to 127	780000h-7FFFFFh	512KB	Upper 8/128
0	0	1	0	1	Block 112 to 127	700000h-7FFFFFh	1024KB	Upper 16/128
0	0	1	1	0	Block 96 to 127	600000h-7FFFFFh	2048KB	Upper 32/128
0	0	1	1	1	Block 64 to 127	400000h-7FFFFFh	4096KB	Upper 64/128
0	1	0	0	0	Block 32 to 127	200000h-7FFFFFh	6144KB	Upper 96/128
0	1	0	0	1	Block 16 to 127	100000h-7FFFFFh	7168KB	Upper 112/128
0	1	0	1	0	Block 8 to 127	080000h-7FFFFFh	7680KB	Upper 120/128
0	1	0	1	1	Block 4 to 127	040000h-7FFFFFh	7936KB	Upper 124/128
0	1	1	0	0	Block 2 to 127	020000h-7FFFFFh	8064KB	Upper 126/128
0	1	1	0	1	Block 1 to 127	010000h-7FFFFFh	8128KB	Upper 127/128
0	1	1	1	0	All	000000h-7FFFFFh	8192KB	All
0	1	1	1	1	All	000000h-7FFFFFh	8192KB	All
1	0	0	0	0	None	None	None	None
1	0	0	0	1	Block 0	000000h-00FFFFh	64KB	Lower 1/128
1	0	0	1	0	Block 0 to 1	000000h-01FFFFh	128KB	Lower 2/128
1	0	0	1	1	Block 0 to 3	000000h-03FFFFh	256KB	Lower 4/128
1	0	1	0	0	Block 0 to 7	000000h-07FFFFh	512KB	Lower 8/128
1	0	1	0	1	Block 0 to 15	000000h-0FFFFFh	1024KB	Lower 16/128
1	0	1	1	0	Block 0 to 31	000000h-1FFFFFh	2048KB	Lower 32/128
1	0	1	1	1	Block 0 to 63	000000h-3FFFFFh	4096KB	Lower 64/128
1	1	0	0	0	Block 0 to 95	000000h-5FFFFFh	6144KB	Lower 96/128
1	1	0	0	1	Block 0 to 111	000000h-6FFFFFh	7168KB	Lower 112/128
1	1	0	1	0	Block 0 to 119	000000h-77FFFFh	7680KB	Lower 120/128
1	1	0	1	1	Block 0 to 123	000000h-7BFFFFh	7936KB	Lower 124/128
1	1	1	0	0	Block 0 to 125	000000h-7DFFFFh	8064KB	Lower 126/128
1	1	1	0	1	Block 0 to 126	000000h-7EFFFFh	8128KB	Lower 127/128
1	1	1	1	0	All	000000h-7FFFFFh	8192KB	All
1	1	1	1	1	All	000000h-7FFFFFh	8192KB	All

Enable Boot Lock

The Enable Boot Lock feature enables user to lock the 64KB block/sector on the top/bottom of the device for protection. This feature is activated by issue Writing Status Register (05h) after entering OTP mode.

The bits' definitions are described in the following table.

Table 4. The Enable Boot Lock feature

Register	Type	Description	Function
SR.3	OTP	Enable 64KB-block/Sector Boot lock	0 (default)
			1 : Permanent lock selected 64KB-Block/Sector
SR.4	OTP	64KB-Block/Sector switch	0 : 64KB-Block (default)
			1 : Sector
SR.6	OTP	Top/Bottom switch	0 : Top (default)
			1 : Bottom

INSTRUCTIONS

All instructions, addresses and data are shifted in and out of the device, most significant bit first. Serial Data Input (DI) is sampled on the first rising edge of Serial Clock (CLK) after Chip Select (CS#) is driven Low. Then, the one-byte instruction code must be shifted in to the device, most significant bit first, on Serial Data Input (DI), each bit being latched on the rising edges of Serial Clock (CLK).

The instruction set is listed in Table 5. Every instruction sequence starts with a one-byte instruction code. Depending on the instruction, this might be followed by address bytes, or by data bytes, or by both or none. Chip Select (CS#) must be driven High after the last bit of the instruction sequence has been shifted in. In the case of a read instruction, the shifted-in instruction sequence is followed by a data-out sequence. Chip Select (CS#) can be driven High after any bit of the data-out sequence is being shifted out.

In the case of a write instruction, Chip Select (CS#) must be driven High exactly at a byte boundary, otherwise the instruction is rejected, and is not executed. That is, Chip Select (CS#) must driven High when the number of clock pulses after Chip Select (CS#) being driven Low is an exact multiple of eight. For Page Program, if at any time the input byte is not a full byte, nothing will happen and WEL will not be reset.

In the case of multi-byte commands of Page Program (PP), Quad Input Page Program (QPP), and Release from Deep Power Down (RES) minimum number of bytes specified has to be given, without which, the command will be ignored.

In the case of Page Program, if the number of byte after the command is less than 4 (at least 1 data byte), it will be ignored too. In the case of SE and HBE / BE, exact 24-bit address is a must, any less or more will cause the command to be ignored.

All attempts to access the memory array during a Write Status Register cycle, Program cycle or Erase cycle are ignored, and the internal Write Status Register cycle, Program cycle or Erase cycle continues unaffected.

Table 5A. Instruction Set

Instruction Name	Byte 1 Code	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	n-Bytes
RSTEN	66h						
RST(1)	99h						
EQPI	38h						
RSTQPI(2)	FFh						
Write Enable	06h						
Write Disable / Exit OTP mode	04h						
Read Status Register	05h	(S7-S0)(3)					continuous(4)
Write Status Register	01h	S7-S0					
Read Status Register 2	09h	(S7-S0)(3)					continuous(4)
Write Status Register 2	07h	S7-S0					
Read Status Register 3	95h	(S7-S0)(3)					
Write Status Register 3	C0h	S7-S0					
Page Program	02h	A23-A16	A15-A8	A7-A0	D7-D0	Next byte	continuous
Quad Input Page Program	32h	A23-A16	A15-A8	A7-A0	(D7-D0, …) (5)		(one byte per 2 clocks, continuous)
Write Suspend	B0h						
Write Resume	30h						
Sector Erase	20h	A23-A16	A15-A8	A7-A0			
32K Half Block Erase (HBE)	52h	A23-A16	A15-A8	A7-A0			
64K Block Erase	D8h	A23-A16	A15-A8	A7-A0			
Chip Erase	C7h/ 60h						
Deep Power-down	B9h						
Release from Deep Power-down, and read Device ID	ABh	dummy	dummy	dummy	(ID7-ID0)		(6)
Release from Deep Power-down							
Manufacturer/ Device ID	90h	dummy	dummy	00h	(M7-M0)	(ID7-ID0)	(7)
				01h	(ID7-ID0)	(M7-M0)	
Read Identification	9Fh	(M7-M0)	(ID15-ID8)	(ID7-ID0)	(8)		
Enter OTP mode	3Ah						
Read SFDP mode and Unique ID Number	5Ah	A23-A16	A15-A8	A7-A0	dummy	(D7-D0)	(Next Byte) continuous

Notes:
1. RST command only executed if RSTEN command is executed first. Any intervening command will disable Reset.
2. Release Full Quad SPI or Fast Read Enhanced mode. Device accepts eight-clocks command in Standard SPI mode, or two-clocks command in Full Quad SPI mode.
3. Data bytes are shifted with Most Significant Bit first. Byte fields with data in parenthesis "( )" indicate data being read from the device on the DO pin.
4. The Status Register contents will repeat continuously until CS# terminate the instruction.
5. Quad Data
   DQ_0 = (D4, D0, ……)
   DQ_1 = (D5, D1, ……)
   DQ_2 = (D6, D2, ……)
   DQ_3 = (D7, D3, ……)
6. The Device ID will repeat continuously until CS# terminates the instruction.
7. The Manufacturer ID and Device ID bytes will repeat continuously until CS# terminates the instruction.
   00h on Byte 4 starts with MID and alternate with DID, 01h on Byte 4 starts with DID and alternate with MID.
8. (M7-M0) : Manufacturer, (ID15-ID8) : Memory Type, (ID7-ID0) : Memory Capacity.

Table 5B. Instruction Set (Read Instruction)

Instruction Name	OP Code	Address bits	Dummy bits / Clocks (Default)	Data Out	Remark
Read Data	03h	24 bits	0	(D7-D0, …)	(Next Byte) continuous
Fast Read	0Bh	24 bits	8 bits / 8 clocks	(D7-D0, …)	(Next Byte) continuous
Dual Output Fast Read	3Bh	24 bits	8 bits / 8 clocks	(D7-D0, …)	(one byte Per 4 clocks, continuous)
Dual I/O Fast Read	BBh	24 bits	8 bits / 4 clocks	(D7-D0, …)	(one byte Per 4 clocks, continuous)
Quad I/O Fast Read	EBh	24 bits	24 bits / 6 clocks	(D7-D0, …)	(one byte Per 2 clocks, continuous)
Read Burst with wrap	0Ch	24 bits	8 bits / 8 clocks	(D7-D0, …)	(Next Byte) continuous

Table 5C. Instruction Set (Read Instruction support mode and apply dummy byte setting)

Instruction Name	OP Code	Start From SPI/QPI (1)		Dummy Byte (2)	
		SPI	QPI	SPI	QPI
Read Data	03h	Yes	No	N/A	N/A
Fast Read	0Bh	Yes	Yes	8 clocks	6 clocks
Dual Output Fast Read	3Bh	Yes	No	8 clocks	N/A
Dual I/O Fast Read	BBh	Yes	No	4 clocks	N/A
Quad I/O Fast Read	EBh	Yes	Yes	By SR3.4~5	By SR3.4~5
Read Burst with wrap	0Ch	Yes	Yes	8 clocks	By SR3.4~5

Note:
1. This command is initiated from SPI or QPI mode.
2. Please refer to table 9 for the dummy byte settings

Table 6. Manufacturer and Device Identification

OP Code	(M7-M0)	(ID15-ID0)	(ID7-ID0)
ABh			16h
90h	1Ch		16h
9Fh	1Ch	7017h	

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 Ready mode. This operation consists of two commands: Reset-Enable (RSTEN) and Reset (RST).

To reset the EN25QH64A the host drives CS# low, sends the Reset-Enable command (66h), and drives CS# high. Next, the host drives CS# low again, sends the Reset command (99h), and drives CS# high.

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.

A successful command execution will reset the Status register and the Suspend Status register to data = 00h, see Figure 7 for SPI Mode and Figure 7.1 for Quad Mode. 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 software latency time (t_{SR}) than recovery from other operations. Please Figure 7.2.

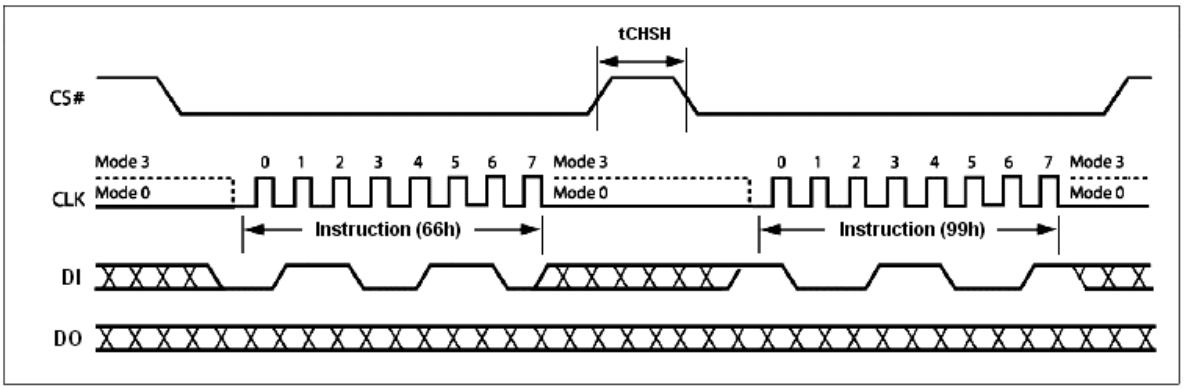


Figure 7. Reset-Enable and Reset Sequence Diagram

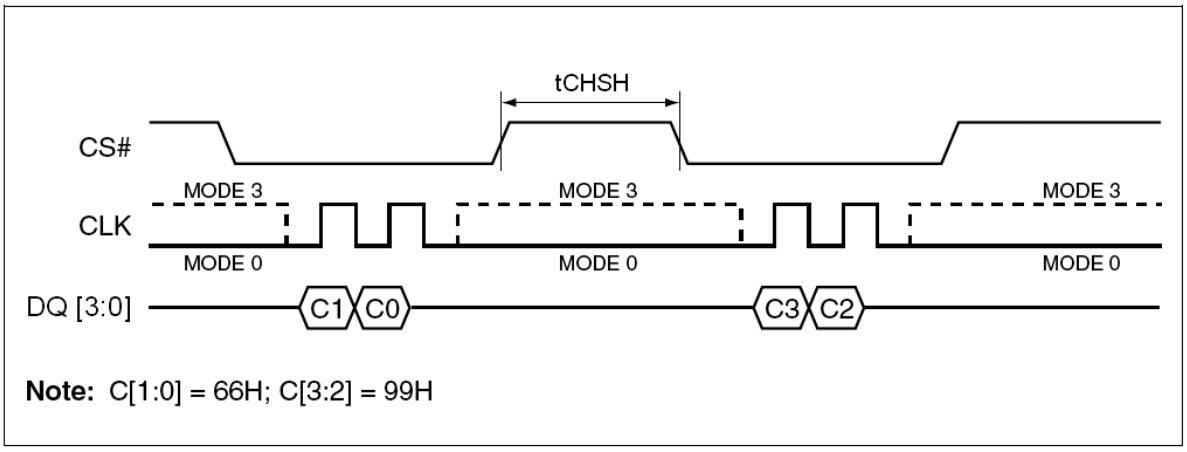


Figure 7.1 . Reset-Enable and Reset Sequence Diagram in QPI Mode

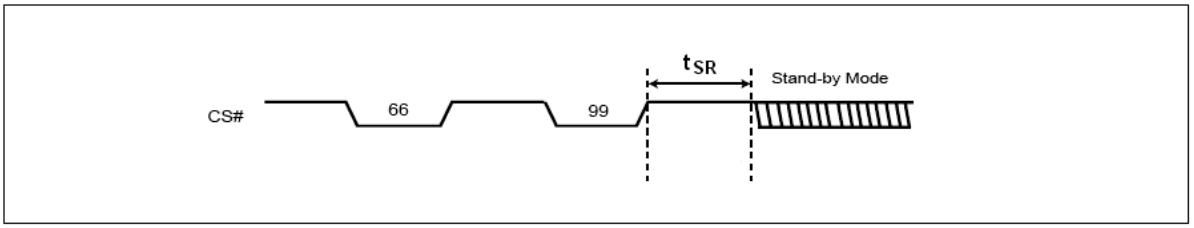
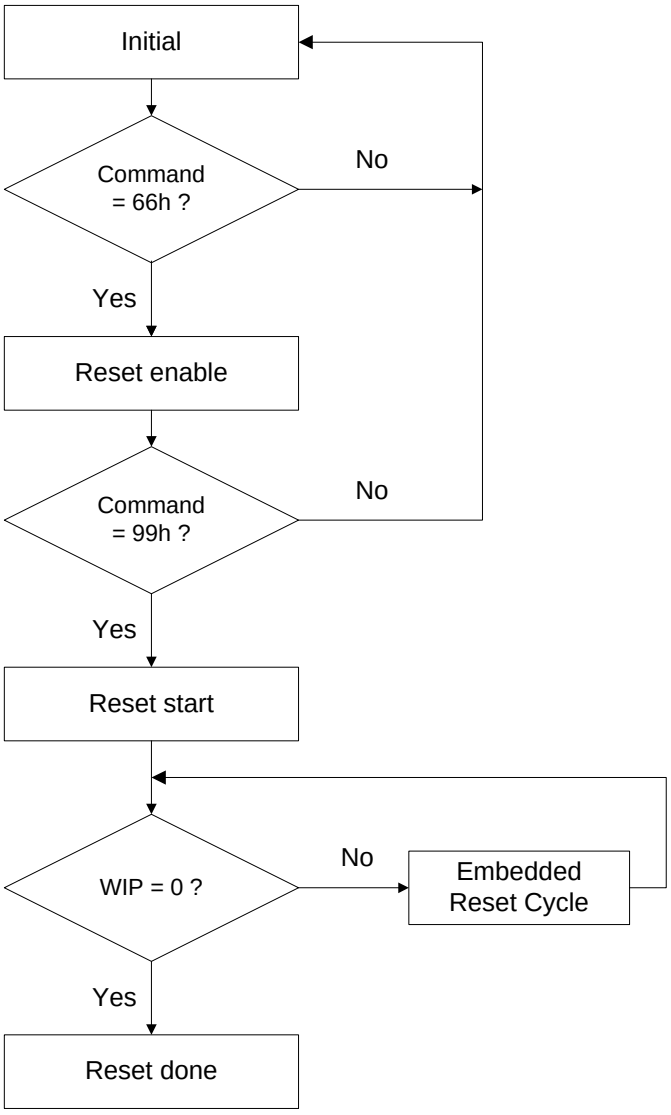


Figure 7.2 Software Reset Recovery

Software Reset Flow



Note:

1. Reset-Enable (RSTEN) (66h) and Reset (RST) (99h) commands need to match standard SPI or EQPI (quad) mode.
2. Continue (Enhance) EB mode need to use quad Reset-Enable (RSTEN) (66h) and quad Reset (RST) (99h) commands.
3. If user is not sure it is in SPI or Quad mode, we suggest to execute sequence as follows:
   Quad Reset-Enable (RSTEN) (66h) -> Quad Reset (RST) (99h) -> SPI Reset-Enable (RSTEN) (66h) -> SPI Reset (RST) (99h) to reset.
4. The reset command could be executed during embedded program and erase process, QPI mode, Continue EB mode and suspend mode to back to SPI mode.
5. This flow cannot release the device from Deep power down mode.
6. The Status Register Bit and Suspend Status Register Bit will reset to default value after reset done.
7. If user reset device during erase, the embedded reset cycle software reset latency will take about 28us in worst case.

Enable Quad Peripheral Interface mode (EQPI) (38h)

The Enable Quad Peripheral Interface mode (EQPI) instruction will enable the flash device for Quad SPI bus operation. Upon completion of the instruction, all instructions thereafter will be 4-bit multiplexed input/output until a power cycle or " Reset Quad I/O instruction " instruction, as shown in Figure 8. The device did not support the 3 byte Read Data Bytes (READ) (03h), 3 Byte FAST_READ) (0Bh) and Quad Input Page Program (32h) modes while the Enable Quad Peripheral Interface mode (EQPI) (38h) turns on.

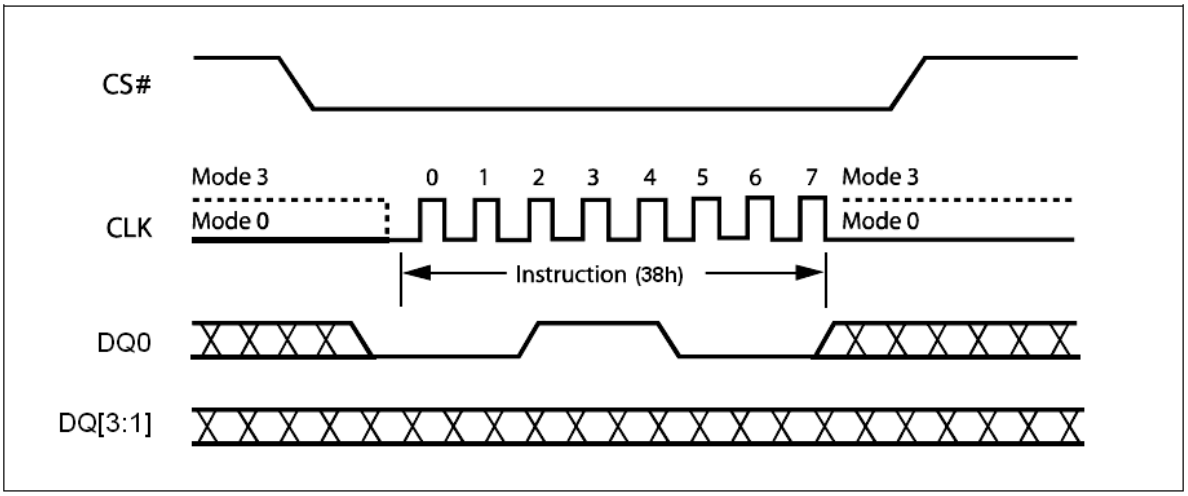


Figure 8. Enable Quad Peripheral Interface mode Sequence Diagram

Reset Full Quad SPI Mode (RSTQPI) (FFh)

The Reset Full Quad SPI Mode instruction resets the device to 1-bit Standard SPI operation. To execute a Reset Quad I/O operation, the host drives CS# low, sends the Reset Quad I/O command cycle (FFh) then, drives CS# high.

User also can use the FFh command to release the Quad I/O Fast Read Enhancement Mode. The detail description, please see the Quad I/O Fast Read Enhancement Mode section.

Write Enable (WREN) (06h)

The Write Enable (WREN) instruction (Figure 9) sets the Write Enable Latch (WEL) bit. The Write Enable Latch (WEL) bit must be set prior to every Page Program (PP), Quad Input Page Program (QPP), Sector Erase (SE), Half Block Erase (HBE), Block Erase (BE), Chip Erase (CE) and Write Status Register (WRSR) instruction.
The Write Enable (WREN) instruction is entered by driving Chip Select (CS#) Low, sending the instruction code, and then driving Chip Select (CS#) High.

The instruction sequence is shown in Figure 10.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

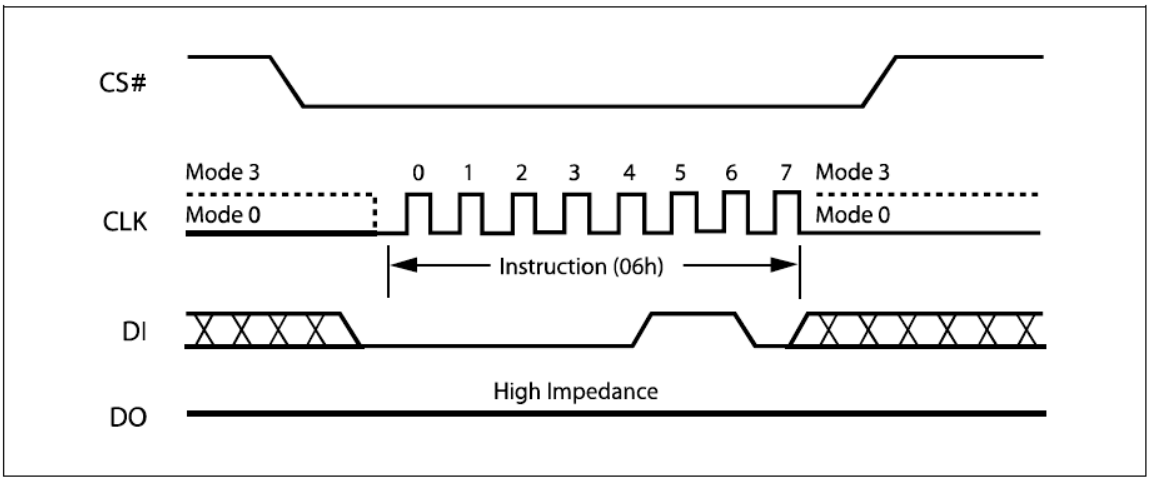


Figure 9. Write Enable Instruction Sequence Diagram

Write Disable (WRDI) (04h)

The Write Disable instruction (Figure 10) resets the Write Enable Latch (WEL) bit in the Status Register to a 0 or exit from OTP mode to normal mode. The Write Disable instruction is entered by driving Chip Select (CS#) low, shifting the instruction code “04h” into the DI pin and then driving Chip Select (CS#) high. Note that the WEL bit is automatically reset after Power-up and upon completion of the Write Status Register, Page Program, Sector Erase, Half Block Erase (HBE), Block Erase (BE) and Chip Erase instructions.

The instruction sequence is shown in Figure 10.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

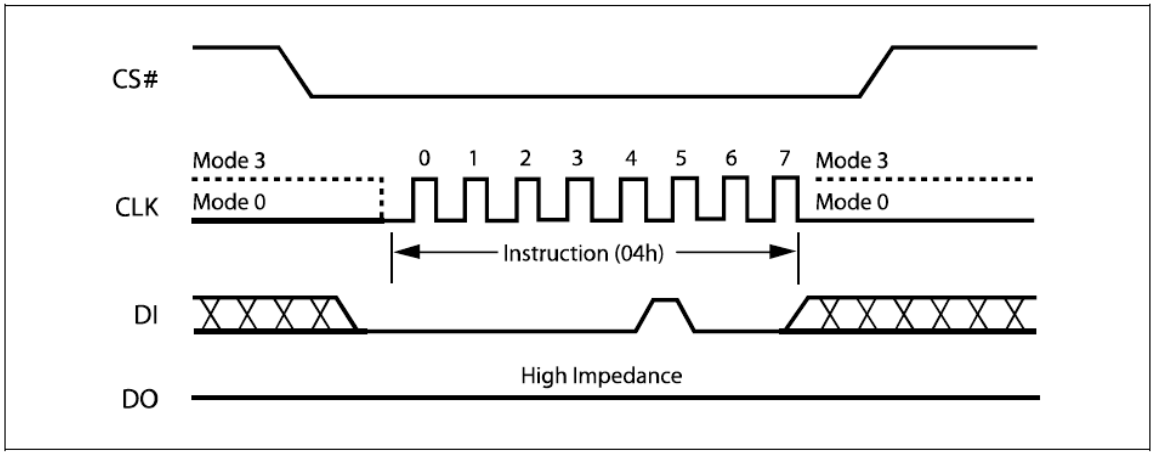


Figure 10. Write Disable Instruction Sequence Diagram

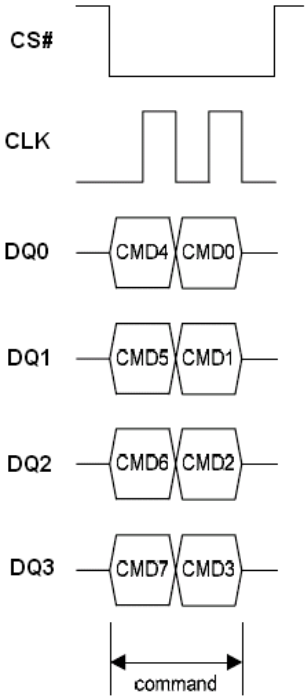


Figure 10.1 Write Enable/Disable Instruction Sequence in QPI Mode

Read Status Register (RDSR) (05h)

The Read Status Register (RDSR) instruction allows the Status Register to be read. The Status Register may be read at any time, even while a Program, Erase or Write Status Register cycle is in progress. When one of these bytes is in progress, it is recommended to check the Write In Progress (WIP) bit before sending a new instruction to the device. It is also possible to read the Status Register continuously, as shown in Figure 11.

	1st byte	2nd byte
104MHz and below	Valid	Valid

The instruction sequence is shown in Figure 11.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

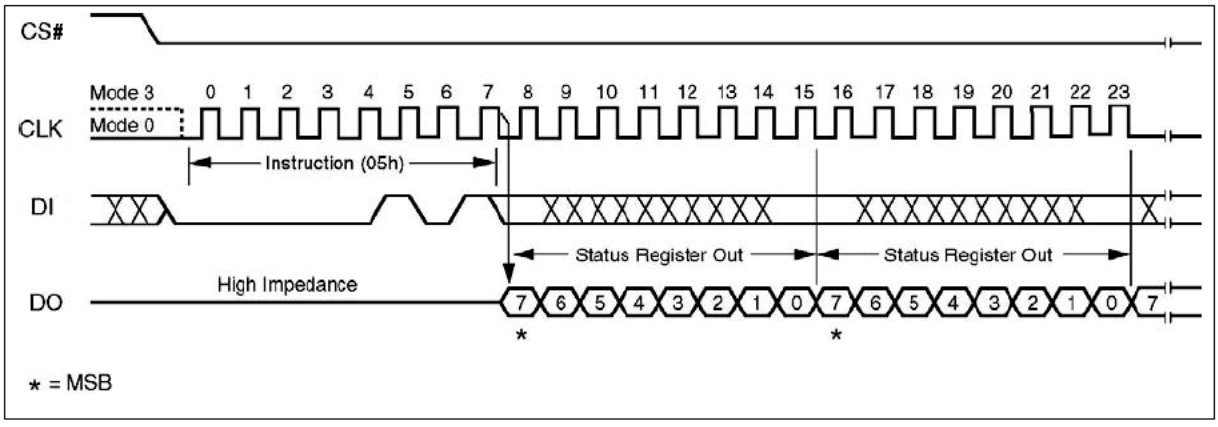


Figure 11. Read Status Register Instruction Sequence Diagram

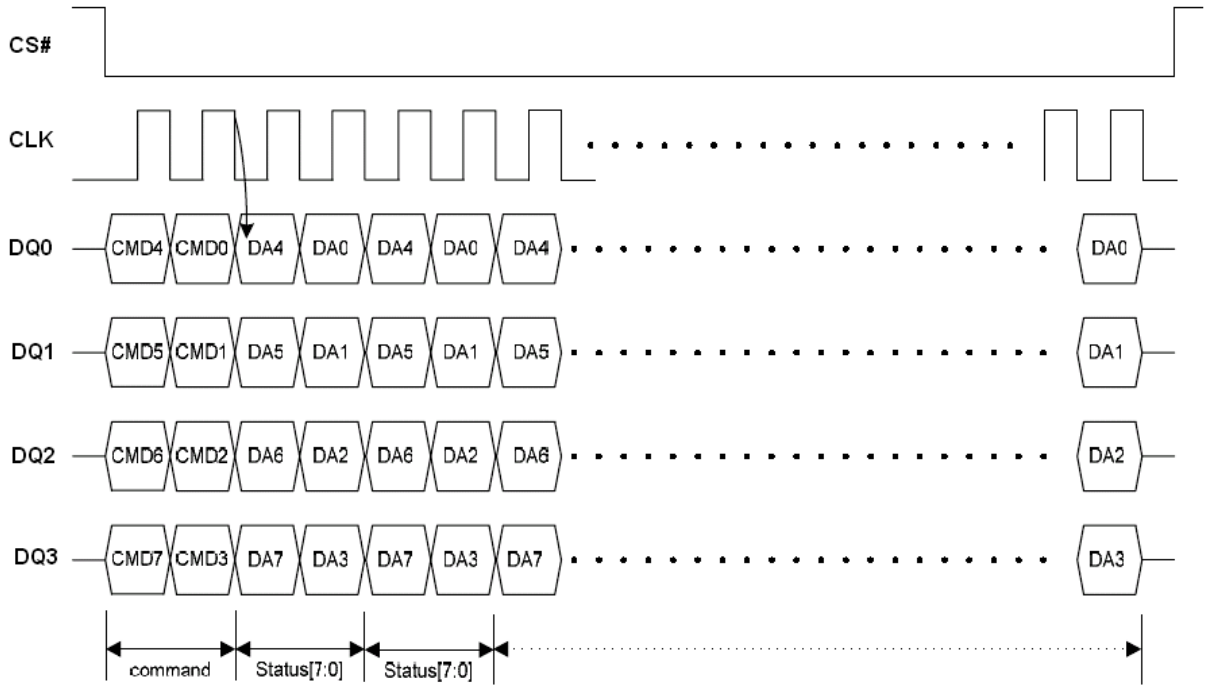


Figure 11.1 Read Status Register Instruction Sequence in QPI Mode

Table 7. Status Register Bit Locations

SR.7	SR.6	SR.5	SR.4	SR.3	SR.2	SR.1	SR.0
SRP bit	WHDIS bit	DBP3 / (BP3) bit	DBP2 / (BP2) bit	DBP1 / (BP1) bit	DBP0 / (BP0) bit	WEL bit	WIP bit
OTP_LOCK bit	TB bit	DBP_LOCK bit	4KB BL bit (4KB boot lock)	EBL bit (Enable boot lock)			

Table 7.1 Status Register Bit Locations

SR.7	SR.6	SR.5	SR.4	SR.3	SR.2	SR.1	SR.0
SRP Status Register Protect	WHDIS bit (WP# and Hold# disabled)	DBP3 / (BP3) bit (Block Protected)	DBP2 / (BP2) bit (Block Protected)	DBP1 / (BP1) bit (Block Protected)	DBP0 / (BP0) bit (Block Protected)	WEL bit (Write Enable Latch)	WIP bit (Write In Progress bit)
1 = status register write disable	1 = WP# and HOLD# disable 0 = WP# and HOLD# enable	(note 2)	(note 2)	(note 2)	(note 2)	1 = write enable 0 = not write enable	1 = write operation 0 = not in write operation
Non-volatile bit	Non-volatile bit					indicator bit	indicator bit

Table 7.2 Status Register Bit Locations (In OTP mode)

SR.7	SR.6	SR.5	SR.4	SR.3	SR.2	SR.1	SR.0
OTP_LOCK bit	TB bit (Top / Bottom Protect)	DBP_LOCK bit	4KB BL bit (4KB boot lock)	EBL bit (Enable boot lock)	none	none	none
1 = OTP sector is protected	1 = Bottom 0 = Top (default 0)	1 = Set all DBP after boot-up 0 = Clear all DBP after boot-up (default 0)	1 = Sector 0 = 64KB Block (default 0)	1 = Permanent lock selected 64KB-Block/Sector			
OTP bit	OTP bit	OTP bit	OTP bit	OTP bit			

Note

1. In OTP mode, S7 bit is served as OTP_LOCK bit; S6 bit is served as TB bit; S5 bit is served as DBP_LOCK bit; S4 bit is served as 4KB BL bit; S3 bit is served as EBL bit.
2. See the table 3 "Protected Area Sizes Sector Organization".

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

WIP bit. The Write In Progress (WIP) bit indicates whether the memory is busy with a Write Status Register, Program or Erase cycle. When set to 1, such a cycle is in progress, when reset to 0 no such cycle is in progress.

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

BP3, BP2, BP1, BP0 bits. The Block Protect (BP3, BP2, BP1, BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase instructions. These bits are written with the Write Status Register (WRSR) instruction. When one or both of the Block Protect (BP3, BP2, BP1, BP0) bits is set to 1, the relevant memory area (as defined in Table 3.) becomes protected against Page Program (PP), Quad Input Page Program (QPP), Sector Erase (SE), Half Block Erase (HBE) and Block Erase (BE), instructions. The Block Protect (BP3, BP2, BP1, BP0) bits can be written and provided that the Hardware Protected mode has not been set. The Chip Erase (CE) instruction is executed if, and only if, all Block Protect (BP3, BP2, BP1, BP0) bits are 0.

DBP3, DBP2, DBP1, DBP0 bits. The Dynamic Block Protect (DBP3, DBP2, DBP1, DBP0) bits are volatile. They define the size of the area to be software protected against Program and Erase instructions. These bits are written with the Write Status Register (WRSR) instruction. When one or both of the Dynamic Block Protect (DBP3, DBP2, DBP1, DBP0) bits is set to 1, the relevant memory area (as defined in Table 3.) becomes protected against Page Program (PP) , Quad Input Page Program (QPP), Sector Erase (SE), Half Block Erase (HBE) and Block Erase (BE) instructions. The Dynamic Block Protect (DBP3, DBP2, DBP1, DBP0) bits can be written and provided that the Hardware Protected mode has not been set. The Chip Erase (CE) instruction is executed if, and only if, all Dynamic Block Protect (DBP3, DBP2, DBP1, DBP0) bits are 0.

WHDIS bit. The WP# and Hold# Disable bit (WHDIS bit), non-volatile bit, it indicates the WP# and HOLD# are enabled or not. When it is "0" (factory default), the WP# and HOLD# are enabled. On the other hand, while WHDIS bit is "1", the WP# and HOLD# are disabled. If the system executes Quad Input/Output FAST_READ (EBh), Quad Input Page Program (32h) or EQPI (38h) command, this WHDIS bit becomes no affection since WP# and HOLD# function will be disabled by Quad Input/Output FAST_READ (EBh) or EQPI mode. This bit is read only if SR2.1 is set to "1".

SRP bit / The Status Register Protect (SRP) bit is operated in conjunction with the Write Protect (WP#) signal. The Status Register Write Protect (SRP) bit and Write Protect (WP#) signal allow the device to be put in the Hardware Protected mode (when the Status Register Protect (SRP) bit is set to 1, and Write Protect (WP#) is driven Low). In this mode, the non-volatile bits of the Status Register (SRP, SR.5, SR.4, SR.3, SR.2) become read-only bits and the Write Status Register (WRSR) instruction is no longer accepted for execution. This bit is read only if SR2.1 is set to "1".

In OTP mode, SR.7, SR.6, SR.5, SR.4 and SR.3 are served as OTP_LOCK bit, TB bit, DBP_LOCK bit,

4KB BL bit and EBL bit.

Enable Boot Lock bit. When this bit is programmed to '1' by WRSR command in OTP mode, the Top/Bottom switch bit and 64KB-Block/Sector switch bit and the selected sector/block will be permanent locked. The Enable Boot Lock bit can only be programmed once.

64KB-Block/Sector switch bit. This bit is set by WRSR command in OTP mode. It is used to set the protection area size as block (64KB) or sector (4KB).

DBP_LOCK bit. DBP_LOCK bit is the bit which indicates whether Dynamic Block Protection Bits (DBP) are set or clear after boot-up. The default state of DBP_LOCK bit is "0" which means all DBP bits are set to all '0' after power-up. All DBP bits are set to all '1' after power-up when DBP_LOCK bit is programmed to '1'. The DBP_LOCK bit can only be programmed once.

Top/Bottom Block Protect Bit (TB). The Top/Bottom bit (TB) controls if the Block Protect Bits (BP3, BP2, BP1, BP0) or Dynamic Block Protect Bits (DBP3, DBP2, DBP1, DBP0) protect from the Top (TB = 0) or the Bottom (TB = 1) of the array as shown in the Status Register Memory Protection table. The factory default setting is TB = 0. The TB bit can be set with the Write Status Register instruction in OTP mode. It is also used to set the protected 64KB-Block/Sector location to the top/bottom in the device when Enable Boot Lock bit is programmed to '1'.

OTP_LOCK bit. This bit is served as OTP_LOCK bit, user can read/program/erase OTP sector as normal sector while OTP_LOCK value is equal 0, after OTP_LOCK is programmed with 1 by WRSR command, the OTP sector is protected from program and erase operation. The OTP_LOCK bit can only be programmed once.

Read Status Register 2 (RDSR 2) (09h)

The Read Status Register 2 (RDSR2) instruction allows the Status Register 2 to be read. The Status Register 2 may be read at any time, even while a Write Suspend or Write Resume cycle is in progress. When one of these bytes is in progress, it is recommended to check the Write In Progress (WIP) bit before sending a new instruction to the device. It is also possible to read the Read Status Register 2 continuously, as shown in Figure 12.

	1st byte	2nd byte
104MHz and below	Valid	Valid

The instruction sequence is shown in Figure 12.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

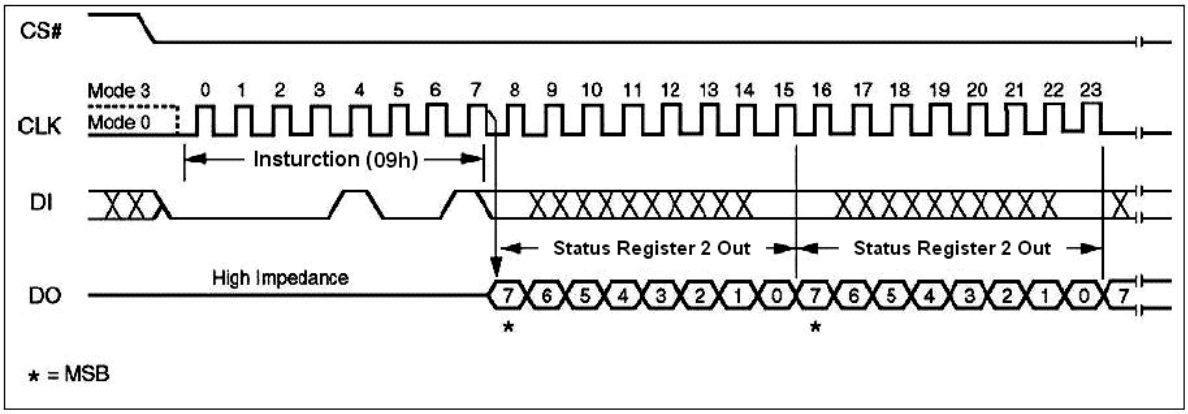


Figure 12. Read Status Register 2 Instruction Sequence Diagram

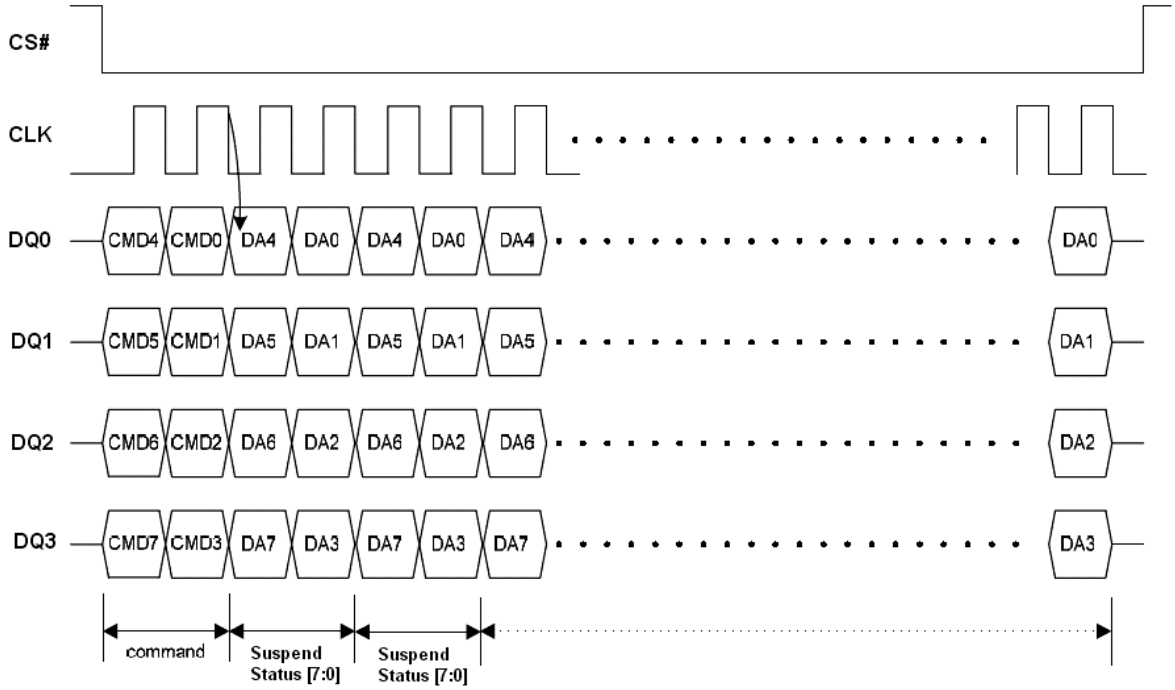


Figure 12.1 Read Status Register 2 Instruction Sequence in QPI Mode

Table 8. Status Register 2 Bit Locations

SR2.7	SR2.6	SR2.5	SR2.4	SR2.3	SR2.2	SR2.1	SR2.0
Reserved bit	Erase Fail Flag	Program Fail Flag	Reserved bit	WSP (Write Suspend Program bits)	WSE (Write Suspend Erase status bit)	DBP select bit	WIP (Write In Progress bit) (Note 1)
	1 = indicate Erase failed 0 = normal Erase succeed (default = 0)	1 = indicate Program failed 0 = normal Program succeed (default = 0)		1 = Program suspended 0 = Program is not suspended	1 = Erase suspended 0 = Erase is not suspended	1 = DBP bits are active 0 = BP bits are active (default = 0)	1 = write operation 0 = not in write operation
	volatile bit	volatile bit		volatile bit	volatile bit	volatile bit	volatile bit
	Read Only	Read Only		Read Only	Read Only		Read Only

Note:

1. The default of each volatile bit is “0” at Power-up or after reset.

The status and control bits of the Status Register 2 are as follows:

WIP bit. The Write In Progress (WIP) bit indicates whether the memory is busy with a Write Status Register, Program or Erase cycle.

DBP select bit. The Dynamic Block Protection (DBP) select bit is the bit which indicates whether Dynamic Block Protection Bits are enable or not. When the device is programmed to set Dynamic Block Protection Bits work, DBP select bit outputs a ‘1’ when the Status Register 2 bits are read. Similarly, when the device is programmed to set Block Protection Bits work, DBP select bit outputs a ‘0’. The default is “0”.

WSE bit. The Write Suspend Erase Status (WSE) bit indicates when an Erase operation has been suspended. The WSE bit is “1” after the host issues a suspend command during an Erase operation. Once the suspended Erase resumes, the WSE bit is reset to “0”.

WSP bit. The Write Suspend Program Status (WSP) bit indicates when a Program operation has been suspended. The WSP is “1” after the host issues a suspend command during the Program operation. Once the suspended Program resumes, the WSP bit is reset to “0”.

Reserved bit. Status Register 2 bit locations SR2.4 and SR2.7 are reserved for future use. Current devices will read 0 for these bit locations. It is recommended to mask out the reserved bit when testing the Suspend Status Register. Doing this will ensure compatibility with future devices.

Program Fail Flag bit. While a program failure happened, the Program Fail Flag bit would be set. This bit will also be set when the user attempts to program a protected main memory region or a locked OTP region. This bit can indicate whether one or more of program operations fail, and can be reset by Program (PP), Quad Input Page Program (QPP) or Erase (SE, HBE/BE or CE) instructions.

Erase Fail Flag bit. While an erase failure happened, the Erase Fail Flag bit would be set. This bit will also be set when the user attempts to erase a protected main memory region or a locked OTP region. This bit can indicate whether one or more of erase operations fail, and can be reset by Program (PP), Quad Input Page Program (QPP) or Erase (SE, HBE/BE or CE) instructions.

Read Status Register 3 (RDSR 3) (95h)

The Read Status Register 3 (RDSR3) instruction allows the Status Register 3 to be read. The Status Register 3 may be read at any time, even while a Write Suspend or Write Resume cycle is in progress. When one of these bytes is in progress, it is recommended to check the Write In Progress (WIP) bit before sending a new instruction to the device. It is also possible to read the Read Status Register 3 continuously, as shown in Figure 13.

	1st byte	2nd byte
104MHz and below	Valid	Valid

The instruction sequence is shown in Figure 13.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

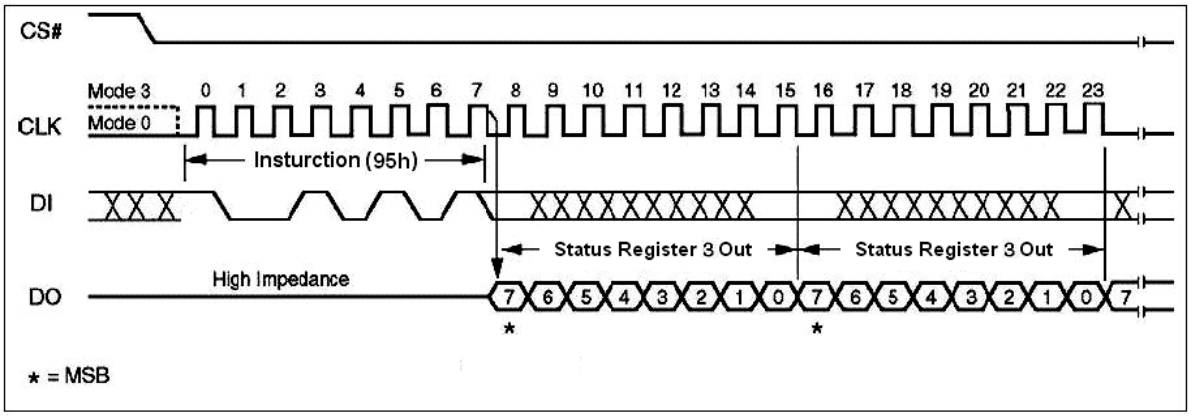


Figure 13. Read Status Register 3 Instruction Sequence Diagram

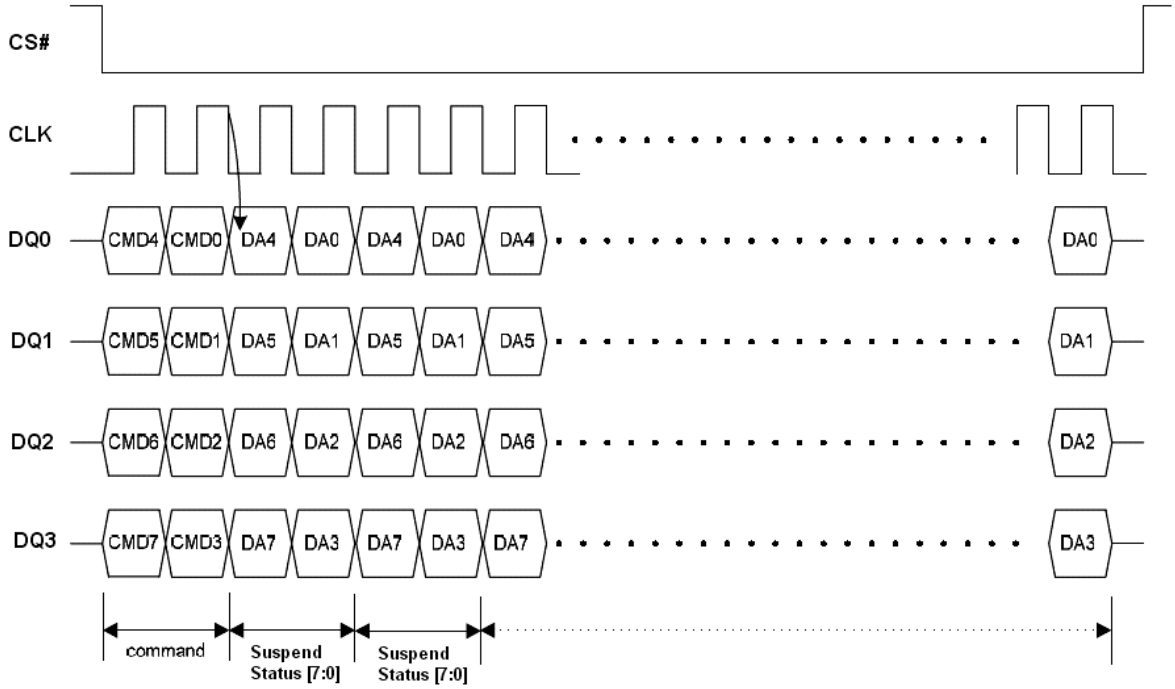


Figure 13.1 Read Status Register 3 Instruction Sequence in QPI Mode

The status and control bits of the Status Register 3 are as follows:

Burst Length. The Burst Length (SR3.1 and SR3.0) bits indicate the status of wrap burst read length.

Output Drive Strength. The Output Drive Strength (SR3.3 and SR3.2) bits indicate the status of output Drive Strength in I/O pins.

Dummy Byte. The Dummy Byte (SR3.5 and SR3.4) bits indicate the status of the number of dummy byte in high performance read.

Reserved bit. SR3.7 and SR3.6 are reserved for future use.

Table 9. Status Register 3 Bit Locations

SR3.7	SR3.6	SR3.5	SR3.4	SR3.3	SR3.2	SR3.1	SR3.0
Reserved	Reserved	Dummy Byte (1) Default = 00		Output Drive Strength		Burst Length	
		For EBh : 00 = 3 Bytes 01 = 3 Bytes 10 = 4 Bytes 11 = 5 Bytes	For 0Ch : 00 = 2 Bytes 01 = 3 Bytes 10 = 4 Bytes 11 = 5 Bytes	00 = Full Drive (default) 01 = 67% (2/3) Drive 10 = 50% (1/2) Drive 11 = 33% (1/3) Drive		00 = 16 Bytes (default) 01 = 32 Bytes 10 = 64 Bytes 11 = 128 Bytes	
volatile bit	volatile bit	volatile bit	volatile bit	volatile bit	volatile bit	volatile bit	volatile bit

Note:

1. 2 Bytes (4 clocks in Quad mode), 3 Bytes (6 clocks in Quad mode), 4 Bytes (8 clocks in Quad mode), 5 Bytes (10 clocks in Quad mode)

Table 10. SR3.4 and SR3.5 Status (for Dummy Bytes)

Instruction Name	Speed	Standard SPI mode			Full Quad SPI mode		
	OP Code	83MHz	104MHz	133MHz	83MHz	104MHz	133MHz
Quad I/O Fast Read	EBh	00 / 01	00 / 01	10	00 / 01	00 / 01	10
Read Burst with wrap	0Ch	N/A	N/A	N/A	00	00	01

Note:
The " N/A " indicates that the dummy byte of 0Ch command in SPI mode is fixed to 8 clocks and this setting takes no effect.

Write Status Register (WRSR) (01h)

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

The Write Status Register (WRSR) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code and the data byte on Serial Data Input (DI).

The instruction sequence is shown in Figure 14. The Write Status Register (WRSR) instruction has no effect on SR.1 and SR.0 of the Status Register. Chip Select (CS#) must be driven High after the eighth bit of the data byte has been latched in. If not, the Write Status Register (WRSR) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Write Status Register cycle (whose duration is t_W) is initiated. While the Write Status Register cycle is in progress, the Status Register may still be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Write Status Register cycle, and is 0 when it is completed. When the cycle is completed, the Write Enable Latch (WEL) is reset.

The Write Status Register (WRSR) instruction allows the user to change the values of the Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits, to define the size of the area that is to be treated as read-only, as defined in Table 3. The Write Status Register (WRSR) instruction also allows the user to set or reset the Status Register Protect (SRP) bit in accordance with the Write Protect (WP#) signal. The Status Register Protect (SRP) bit and Write Protect (WP#) signal allow the device to be put in the Hardware Protected Mode (HPM). The Write Status Register (WRSR) instruction is not executed once the Hardware Protected Mode (HPM) is entered.

NOTE :

In the OTP mode, WRSR command is used to program OTP_LOCK bit, TB bit, DBP_LOCK bit, 4KB BL bit and EBL bit to '1', but these bits can only be programmed once.

The instruction sequence is shown in Figure 14.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

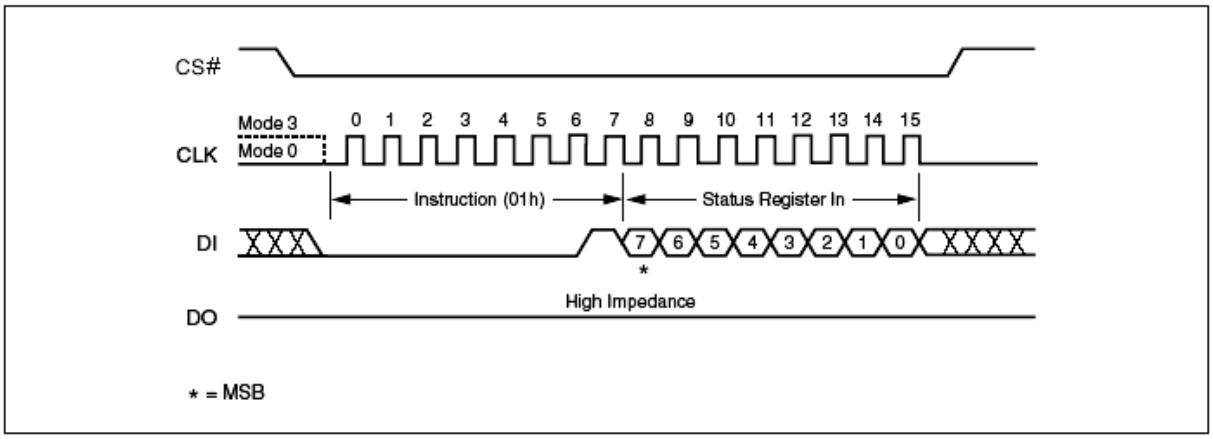


Figure 14. Write Status Register Instruction Sequence Diagram

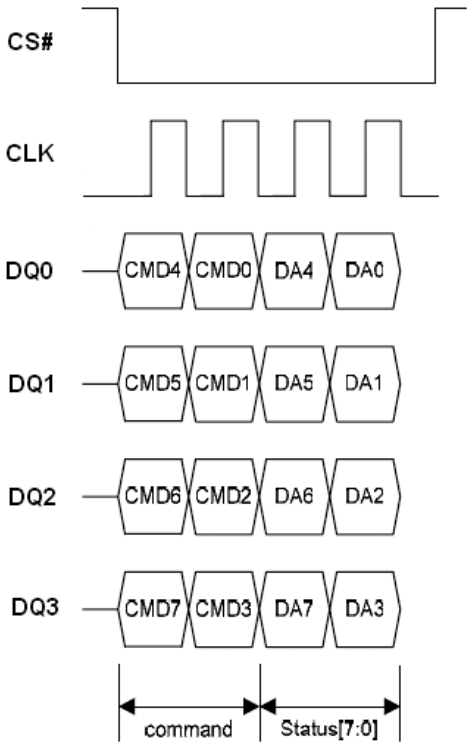


Figure 14.1 Write Status Register Instruction Sequence in QPI Mode

Write Status Register 2 (WRSR2) (07h)

The Write Status Register 2 (WRSR2) instruction allows new values to be written to the Status Register 2. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded and executed, the device sets the Write Enable Latch (WEL).

The Write Status Register 2 (WRSR2) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code and the data byte on Serial Data Input (DI).
The instruction sequence is shown in Figure 15. The Write Status Register 2 (WRSR2) instruction has no effect on SR2.6, SR2.5, SR2.3, SR2.2 and SR2.0 of the Status Register 2. Chip Select (CS#) must be driven High after the eighth bit of the data byte has been latched in. If not, the Write Status Register 2 (WRSR2) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Write Status Register 2 cycle (whose duration is t_W) is initiated. While the Write Status Register 2 cycle is in progress, the Status Register 2 may still be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Write Status Register 2 cycle, and is 0 when it is completed. When the cycle is completed, the Write Enable Latch (WEL) is reset.
The Write Status Register 2 (WRSR2) instruction allows the user to change the values of the Dynamic Block Protection (DBP) select bit, to indicate whether Block Protection Bits are enable or not.

The instruction sequence is shown in Figure 15.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

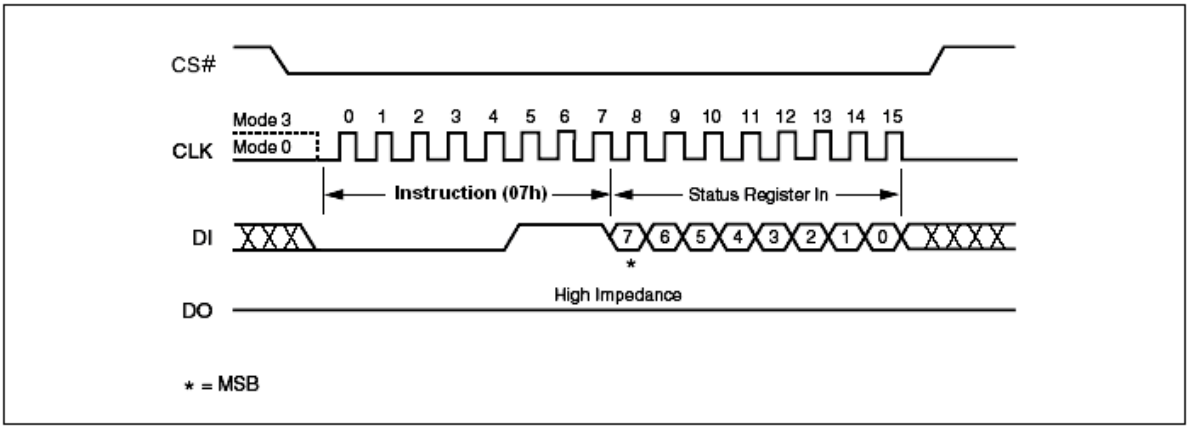


Figure 15. Write Status Register 2 Instruction Sequence Diagram

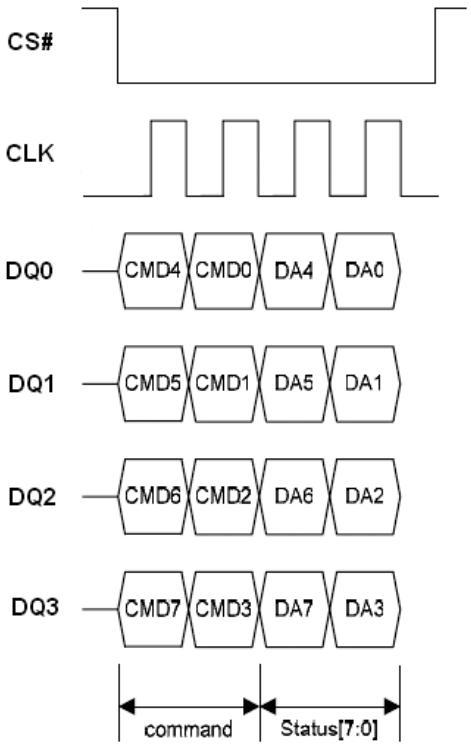


Figure 15.1 Write Status Register 2 Instruction Sequence in QPI Mode

Read Data Bytes (READ) (03h)

The device is first selected by driving Chip Select (CS#) Low. The instruction code for the Read Data Bytes (READ) instruction is followed by a 3-byte address (A23-A0), each bit being latched-in during the rising edge of Serial Clock (CLK). Then the memory contents, at that address, is shifted out on Serial Data Output (DO), each bit being shifted out, at a maximum frequency f_R, during the falling edge of Serial Clock (CLK).

The instruction sequence is shown in Figure 16. The first byte addressed can be at any location from 000000h to 7FFFFFh. The address is automatically incremented to the next higher address after each byte of data is shifted out. The whole memory can, therefore, be read with a single Read Data Bytes (READ) instruction. When the highest address is reached, the address counter rolls over to 000000h, allowing the read sequence to be continued indefinitely.

The Read Data Bytes (READ) instruction is terminated by driving Chip Select (CS#) High. Chip Select (CS#) can be driven High at any time during data output. Any Read Data Bytes (READ) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

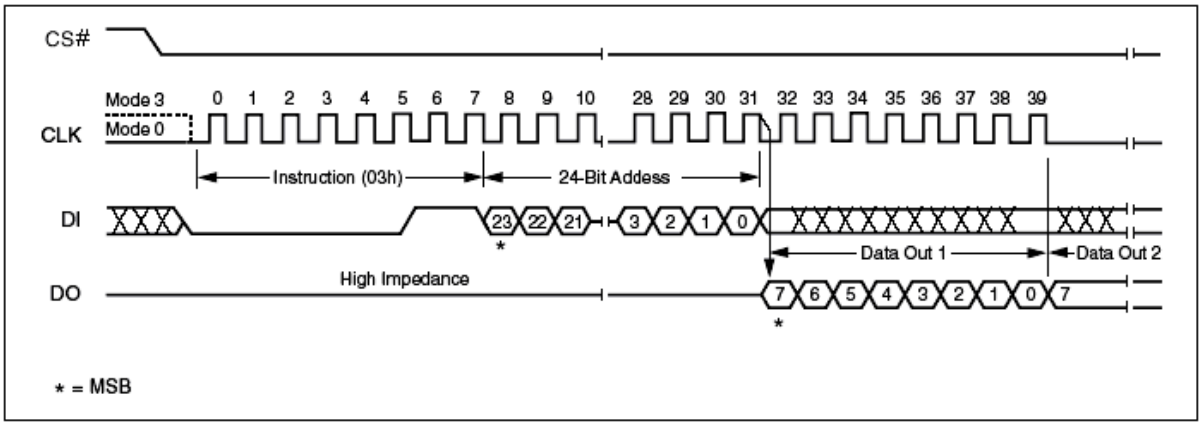


Figure 16. Read Data Instruction Sequence Diagram

Read Data Bytes at Higher Speed (FAST_READ) (0Bh)

The device is first selected by driving Chip Select (CS#) Low. The instruction code for the Read Data Bytes at Higher Speed (FAST_READ) instruction is followed by a 3-byte address (A23-A0) and a dummy byte, each bit being latched-in during the rising edge of Serial Clock (CLK). Then the memory contents, at that address, is shifted out on Serial Data Output (DO), each bit being shifted out, at a maximum frequency F_R, during the falling edge of Serial Clock (CLK).

The instruction sequence is shown in Figure 17. The first byte addressed can be at any location from 000000h to 7FFFFFh. The address is automatically incremented to the next higher address after each byte of data is shifted out. The whole memory can, therefore, be read with a single Read Data Bytes at Higher Speed (FAST_READ) instruction. When the highest address is reached, the address counter rolls over to 000000h, allowing the read sequence to be continued indefinitely.

The Read Data Bytes at Higher Speed (FAST_READ) instruction is terminated by driving Chip Select (CS#) High. Chip Select (CS#) can be driven High at any time during data output. Any Read Data Bytes at Higher Speed (FAST_READ) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

The instruction sequence is shown in Figure 17.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

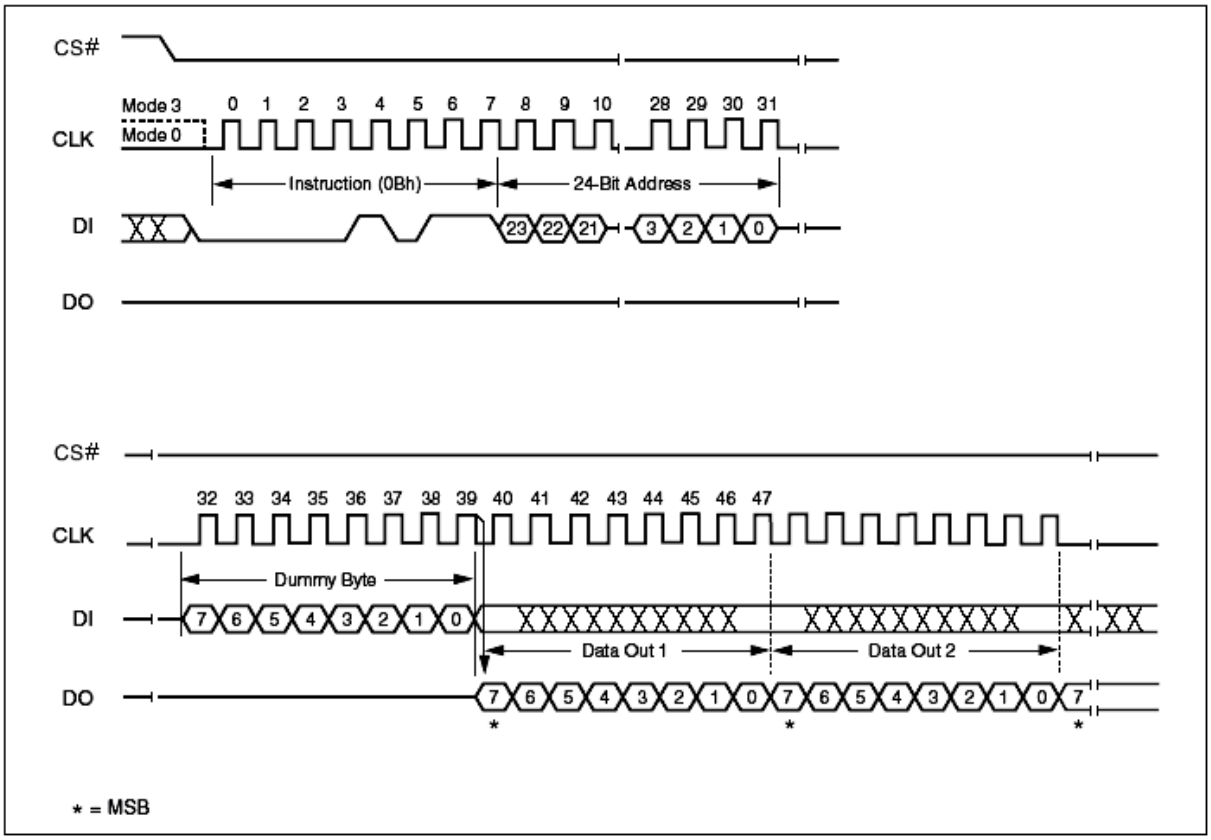


Figure 17. Fast Read Instruction Sequence Diagram

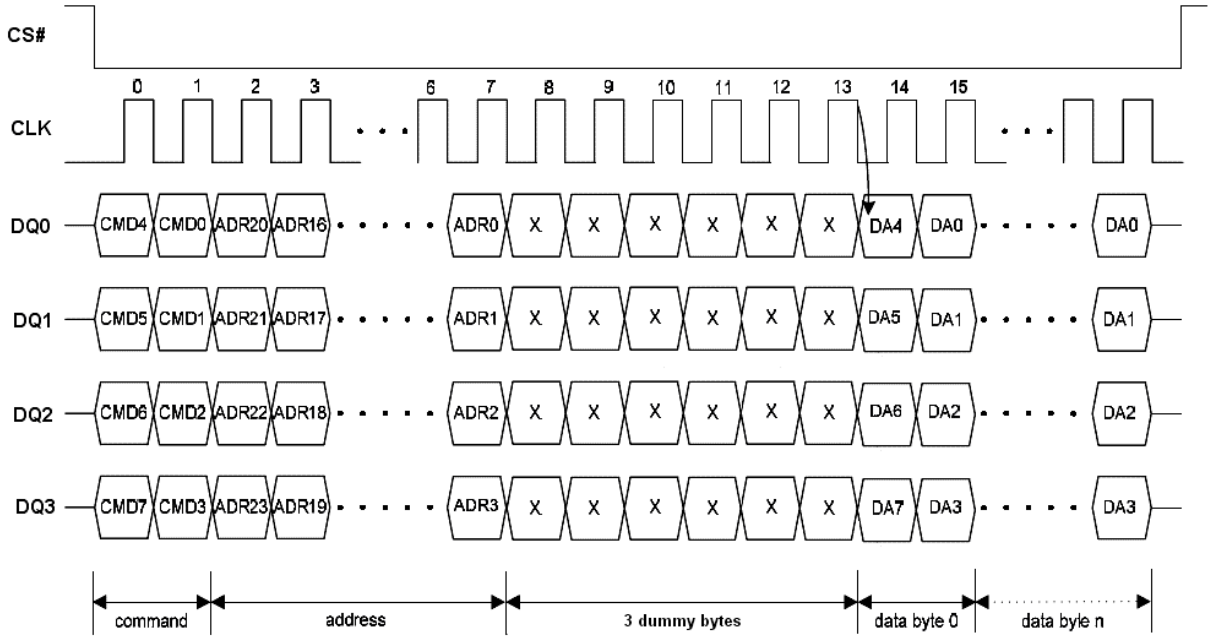


Figure 17.1 Fast Read Instruction Sequence in QPI Mode

Dual Output Fast Read (3Bh)

The Dual Output Fast Read (3Bh) is similar to the standard Fast Read (0Bh) instruction except that data is output on two pins, DQ_0 and DQ_1, instead of just DQ_0. This allows data to be transferred from the EN25QH64A at twice the rate of standard SPI devices. The Dual Output Fast Read instruction is ideal for quickly downloading code from to RAM upon power-up or for applications that cache code-segments to RAM for execution.

Similar to the Fast Read instruction, the Dual Output Fast Read instruction can operation at the highest possible frequency of FR (see AC Electrical Characteristics). This is accomplished by adding eight dummy clocks after the 24-bit address as shown in Figure 18. The dummy clocks allow the device's internal circuits additional time for setting up the initial address. The input data during the dummy clock is "don't care". However, the DI pin should be high-impedance prior to the falling edge of the first data out clock.

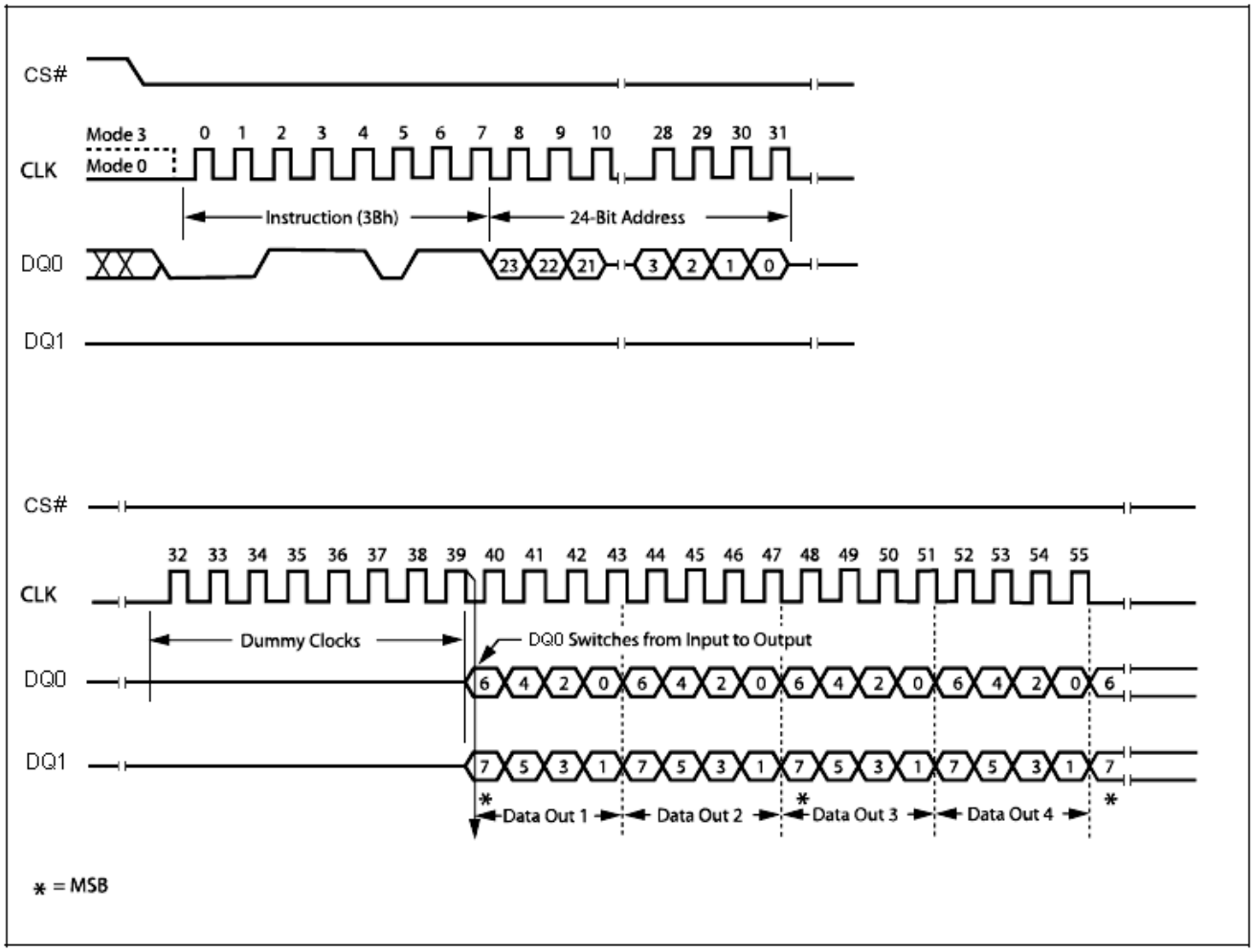


Figure 18. Dual Output Fast Read Instruction Sequence Diagram

Dual Input / Output FAST_READ (BBh)

The Dual I/O Fast Read (BBh) instruction allows for improved random access while maintaining two IO pins, DQ_0 and DQ_1. It is similar to the Dual Output Fast Read (3Bh) instruction but with the capability to input the Address bits (A23-A0) two bits per clock. This reduced instruction overhead may allow for code execution (XIP) directly from the Dual SPI in some applications.

The Dual I/O Fast Read instruction enable double throughput of Serial Flash in read mode. The address is latched on rising edge of CLK, and data of every two bits (interleave 2 I/O pins) shift out on the falling edge of CLK at a maximum frequency. The first address 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 Dual I/O Fast Read instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing Dual I/O Fast Read instruction, the following address/dummy/data out will perform as 2-bit instead of previous 1-bit, as shown in Figure 19.

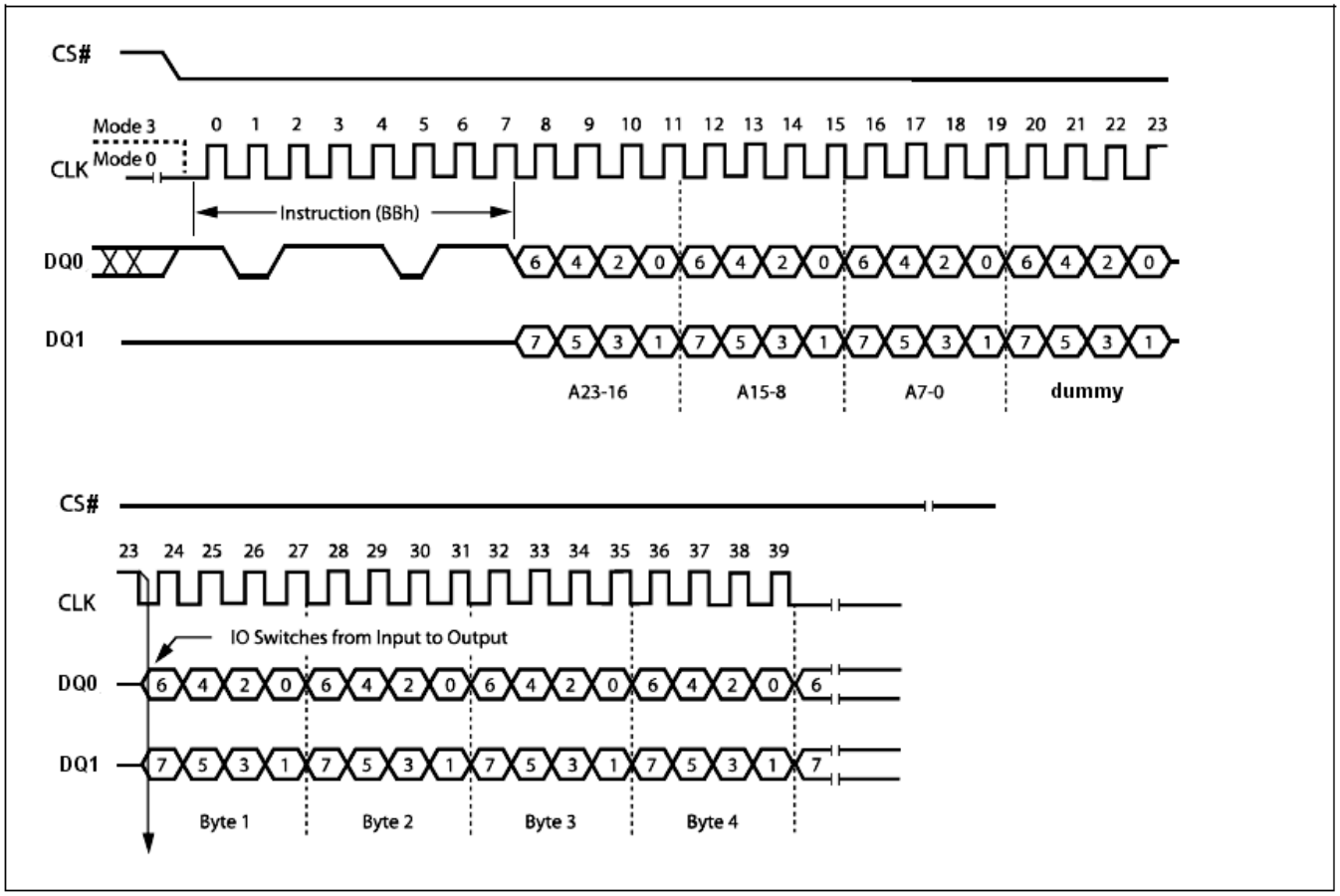


Figure 19. Dual Input / Output Fast Read Instruction Sequence Diagram

Quad Input / Output FAST_READ (EBh)

The Quad Input/Output FAST_READ (EBh) instruction allows for improved random access while maintaining four IO pins, and data bits are input and output through four pins, DQ_0, DQ_1, DQ_2 and DQ_3 and 3 dummy bytes (6 clocks) are required prior to the data output. The Quad I/O dramatically reduces instruction overhead allowing faster random access for code execution (XIP) directly from the Quad SPI.

The Quad Input/Output FAST_READ (EBh) instruction enable quad throughput of Serial Flash in read mode. The address is latching on rising edge of CLK, and data of every four bits (interleave on 4 I/O pins) shift out on the falling edge of CLK at a maximum frequency F_R. The first address can be 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 Quad Input/Output FAST_READ instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing Quad Input/Output FAST_READ instruction, the following address/dummy/data out will perform as 4-bit instead of previous 1-bit.

The sequence of issuing Quad Input/Output FAST_READ (EBh) instruction is: CS# goes low -> sending Quad Input/Output FAST_READ (EBh) instruction -> 24-bit address interleave on DQ_3, DQ_2, DQ_1 and DQ_0 -> 3 dummy bytes (6 clocks) -> data out interleave on DQ_3, DQ_2, DQ_1 and DQ_0 -> to end Quad Input/Output FAST_READ (EBh) operation can use CS# to high at any time during data out, as shown in Figure 20.

The instruction sequence is shown in Figure 20.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

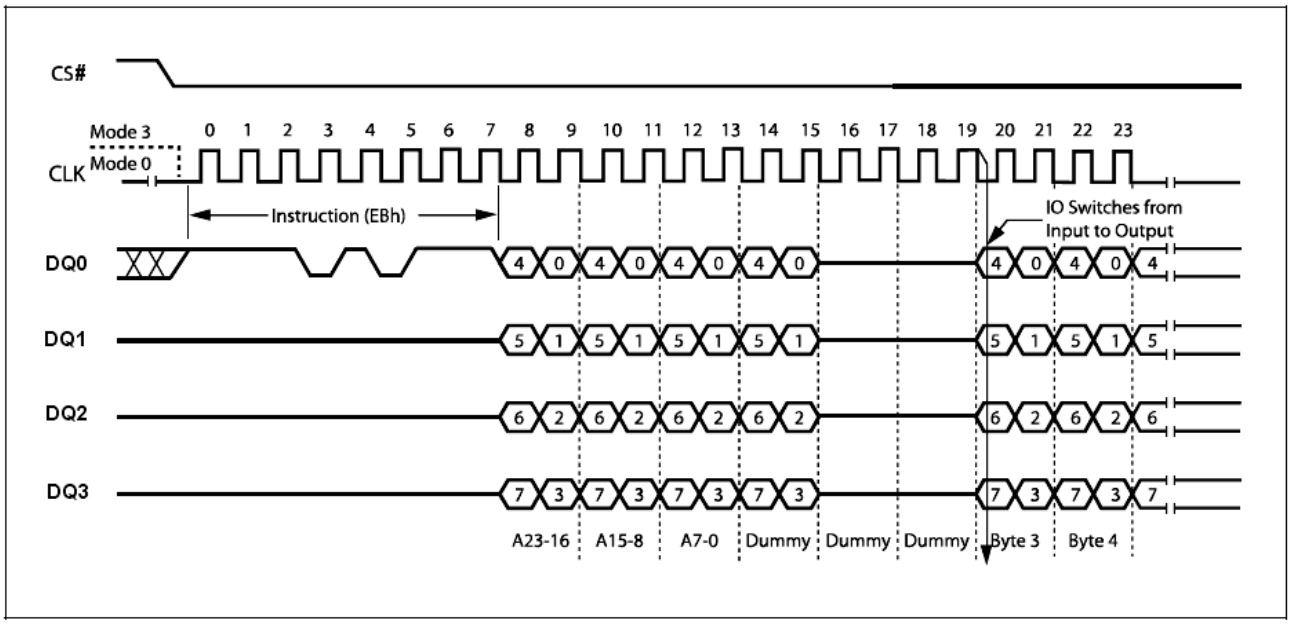


Figure 20. Quad Input / Output Fast Read Instruction Sequence Diagram

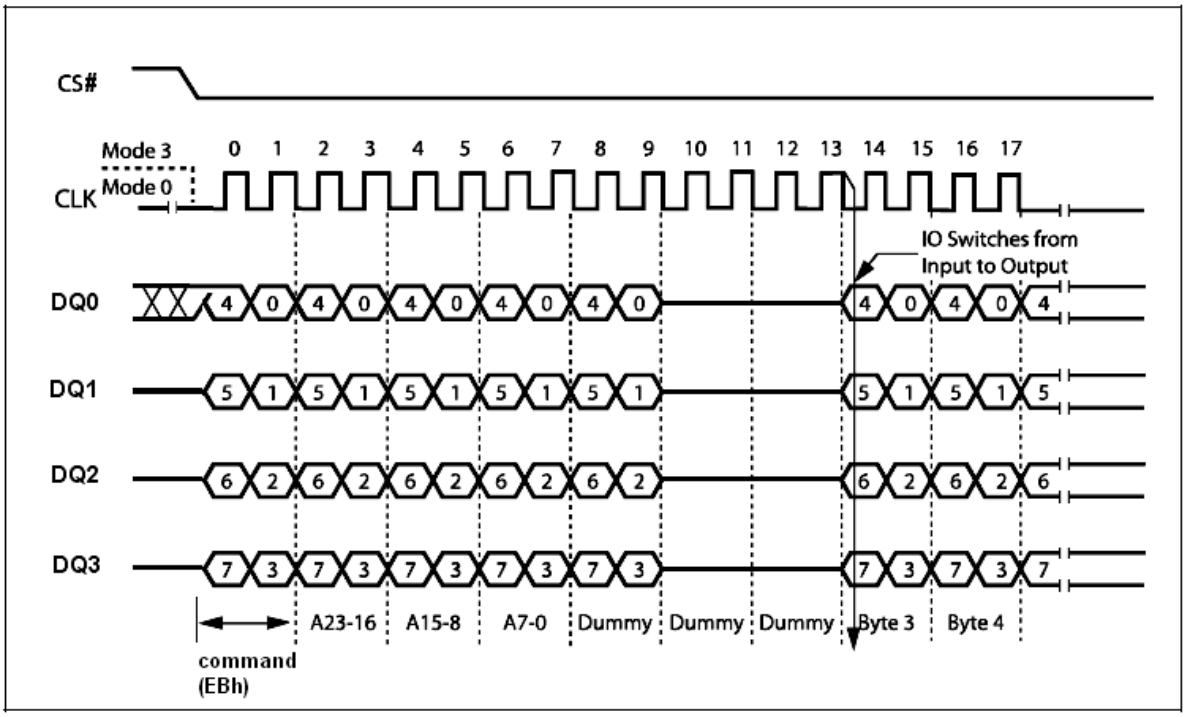


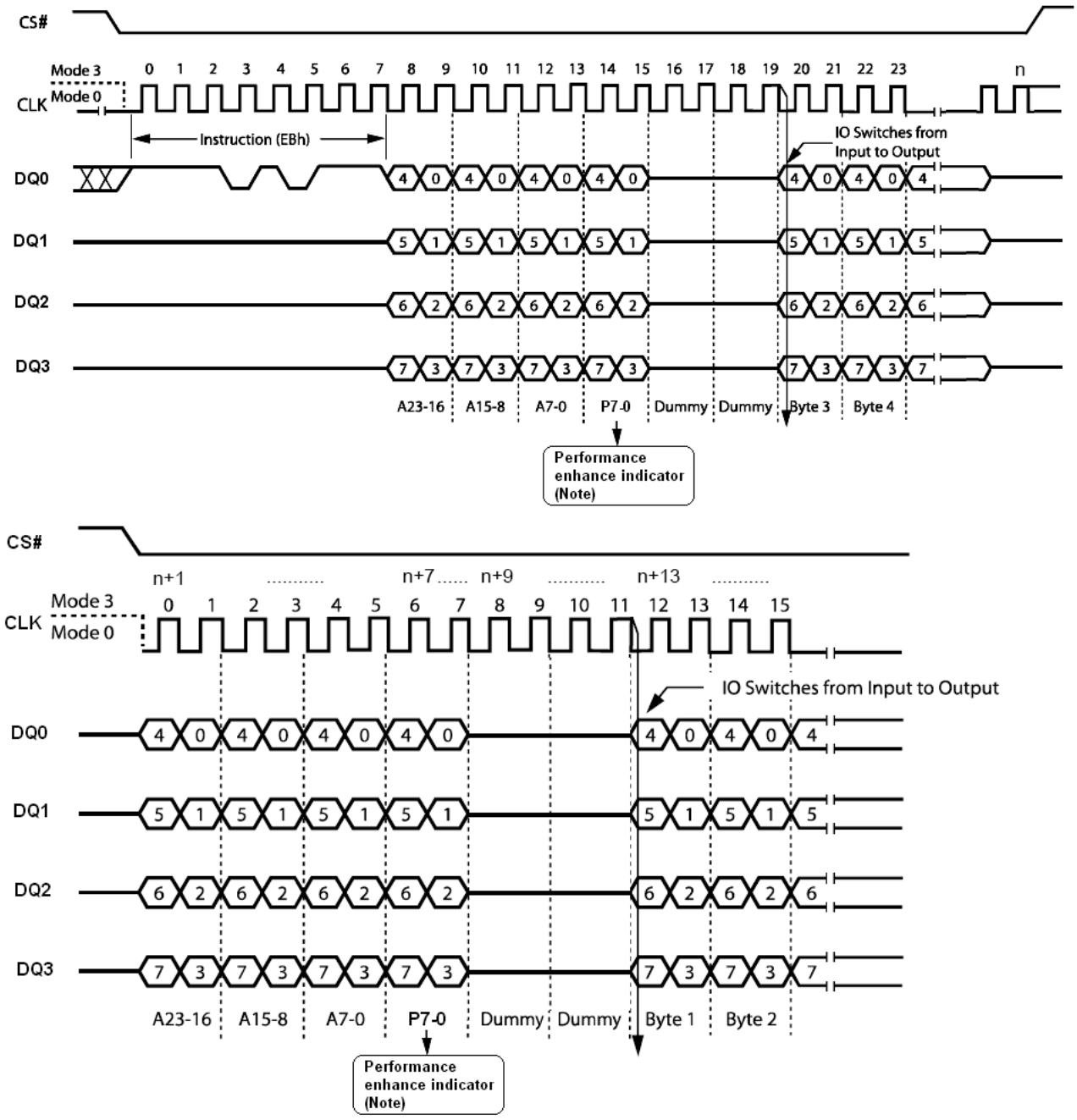
Figure 20.1. Quad Input / Output Fast Read Instruction Sequence in QPI Mode

Another sequence of issuing Quad Input/Output FAST_READ (EBh) instruction especially useful in random access is : CS# goes low -> sending Quad Input/Output FAST_READ (EBh) instruction -> 24-bit address interleave on DQ_3, DQ_2, DQ_1 and DQ_0 -> performance enhance toggling bit P[7:0] -> 4 dummy clocks -> data out interleave on DQ_3, DQ_2, DQ_1 and DQ_0 till CS# goes high -> CS# goes low (reduce Quad Input/Output FAST_READ (EBh) instruction) -> 24-bit random access address, as shown in Figure 21.

In the performance – enhancing mode, P[7:4] must be toggling with P[3:0] ; likewise P[7:0] = A5h, 5Ah, F0h or 0Fh can make this mode continue and reduce the next Quad Input/Output FAST_READ (EBh) instruction. Once P[7:4] is no longer toggling with P[3:0] ; likewise P[7:0] = FFh, 00h, AAh or 55h. These commands will reset the performance enhance mode. And afterwards CS# is raised or issuing FFh command (CS# goes high -> CS# goes low -> sending FFh -> CS# goes high) instead of no toggling, the system then will escape from performance enhance mode and return to normal operation.

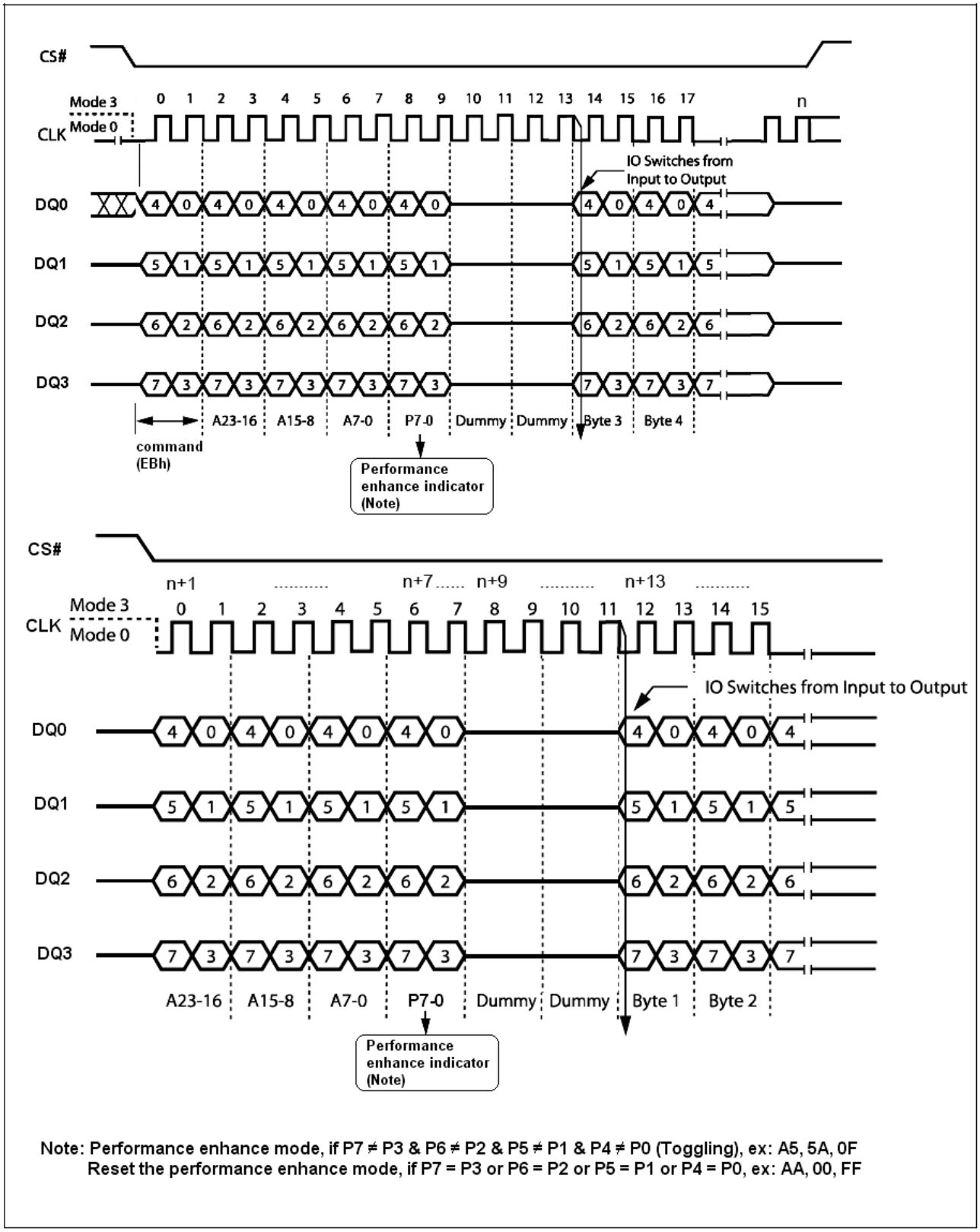
While Program/ Erase/ Write Status Register is in progress, Quad Input/Output FAST_READ (EBh) instruction is rejected without impact on the Program/ Erase/ Write Status Register current cycle.

The instruction sequence is shown in Figure 21.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.



Note: Performance enhance mode, if P7 ≠ P3 & P6 ≠ P2 & P5 ≠ P1 & P4 ≠ P0 (Toggling), ex: A5, 5A, 0F
Reset the performance enhance mode, if P7 = P3 or P6 = P2 or P5 = P1 or P4 = P0, ex: AA, 00, FF

Figure 21. Quad Input/Output Fast Read Enhance Performance Mode Sequence Diagram



Note: Performance enhance mode, if P7 ≠ P3 & P6 ≠ P2 & P5 ≠ P1 & P4 ≠ P0 (Toggling), ex: A5, 5A, 0F
Reset the performance enhance mode, if P7 = P3 or P6 = P2 or P5 = P1 or P4 = P0, ex: AA, 00, FF

Figure 21.1 Quad Input/Output Fast Read Enhance Performance Mode Sequence in QPI Mode

Write Status Register 3 (C0h)

The Write Status Register 3 (C0h) command specifies the number of bytes to be output during a Read Bust command before the device wraps around. This instruction can be used to set wrap burst read length, output drive strength in I/O pins and the number of dummy byte in high performance read. To set the burst length data the host driver CS# low, sends the Write Status Register 3 (C0h) and one data byte, then drivers CS# high, After power-up or reset, the burst length is set to 16 bytes (00b), the output drive strength is set to full drive (00b) and the dummy byte is set to 3 bytes (00b), please refer to Table 9 for Status Register 3 data and Figure 22 for the sequence. In QPI mode, a cycle is two nibbles, or two clocks, long, most significant nibble first.

The instruction sequence is shown in Figure 22.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

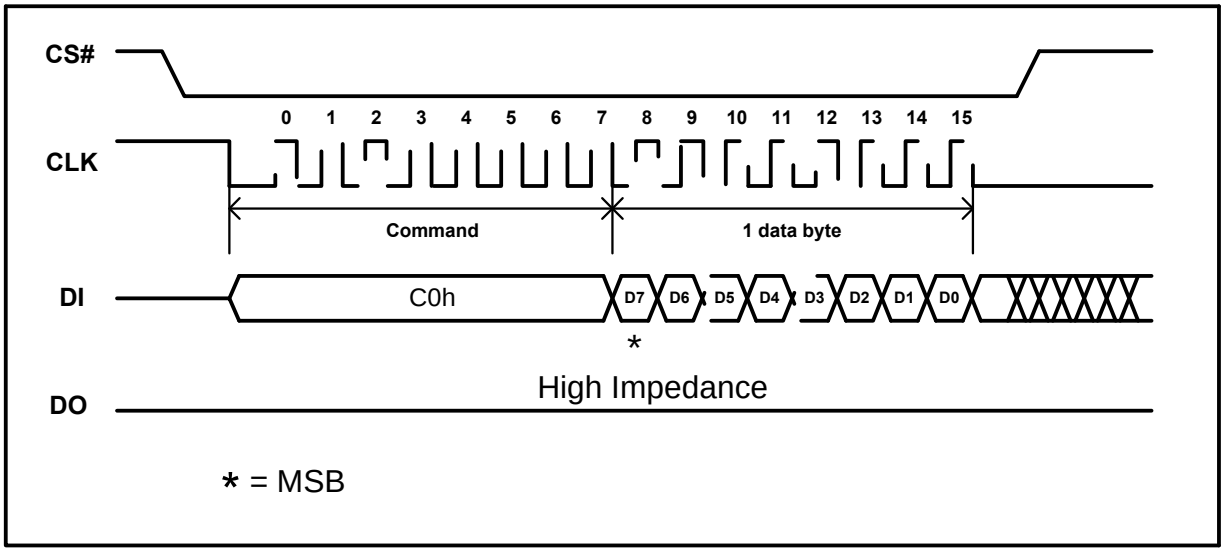
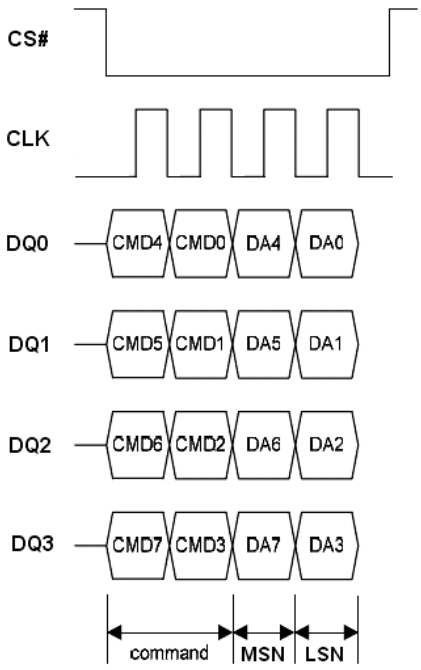


Figure 22. Write Status Register 3 Instruction Sequence Diagram



Note: MSN = Most Significant Nibble, LSN = Least Significant Nibble

Figure 22.1 Write Status Register 3 Instruction Sequence Diagram in QPI mode

Read Burst with wrap (0Ch)

This device supports Read Burst with wrap in both SPI and QPI mode. To execute a Read Burst with wrap operation the host drivers CS# low, and sends the Read Burst with wrap (0Ch) command cycle, followed by three address bytes and one dummy byte (8 clocks) in SPI mode (Figure 23) or default two dummy bytes (4 clocks) in QPI mode (Figure 23.1).

After the dummy byte, the device outputs data on the falling edge of the CLK signal starting from the specific address location. The data output stream is continuous through all addresses until terminated by a low-to high transition of CS# signal.

During Read Burst, the internal address point automatically increments until the last byte of the burst reached, then jumps to first byte of the burst. All bursts are aligned to addresses within the burst length, see Table 11. For example, if the burst length is 16 bytes, and the start address is 06h, the burst sequence should be: 06h, 07h, 08h, 09h, 0A, 0B, 0C, 0Dh, 0Eh, 0Fh, 00h, 01h, 02h, 03h, 04h, 05, 06, etc. The pattern would repeat until the command was terminated by pulling CS# as high status.

The instruction sequence is shown in Figure 23.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

Table 11. Burst Address Range

Burst length	Burst wrap (A[7:A0]) address range
16 Bytes	00-0FH, 10-1FH, 20-2FH, 30-3FH...
32 Bytes	00-1FH, 20-3FH, 40-5FH, 60-7FH...
64 Bytes	00-3FH, 40-7FH, 80-BFH, C0-FFH
128 Bytes	00-7FH, 80-FFH

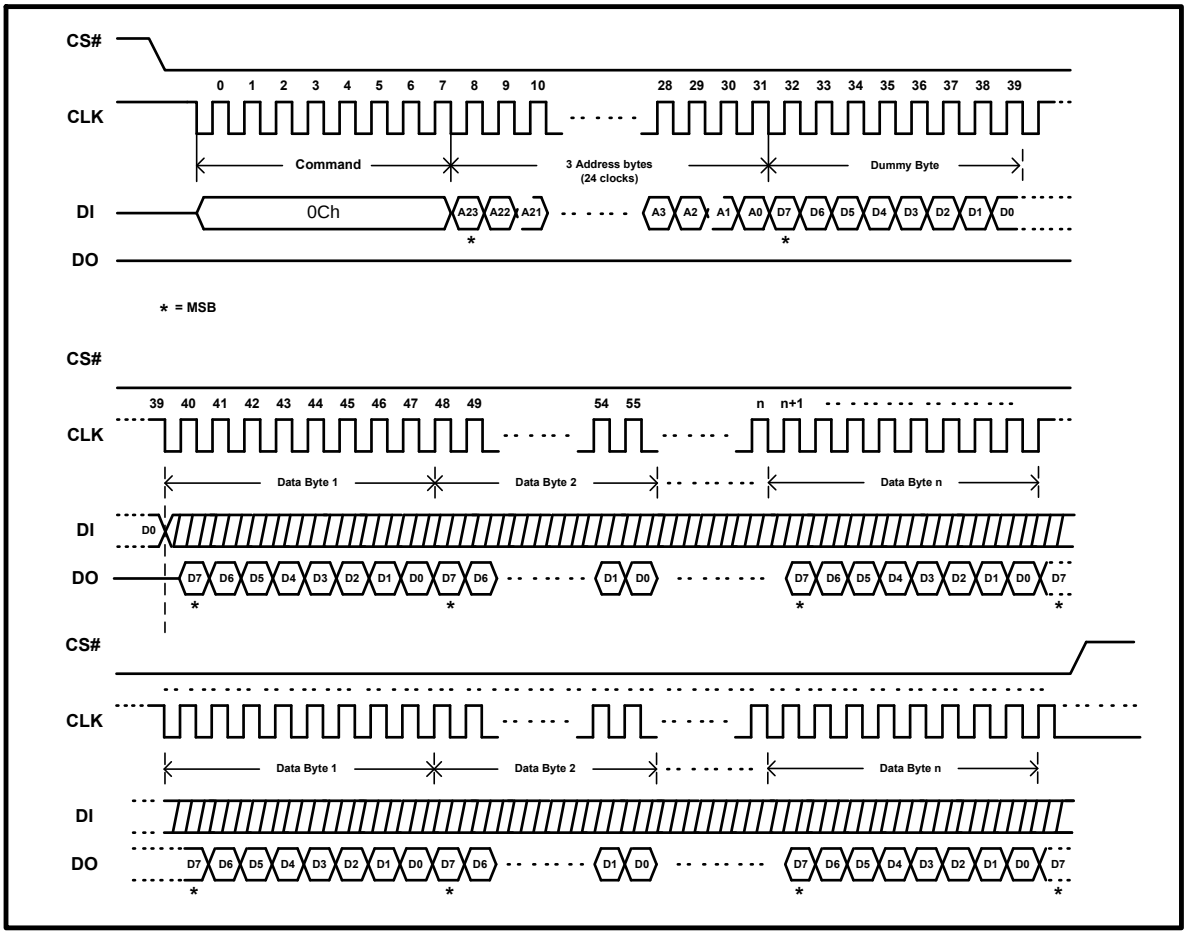


Figure 23. Read Burst Instruction Sequence Diagram

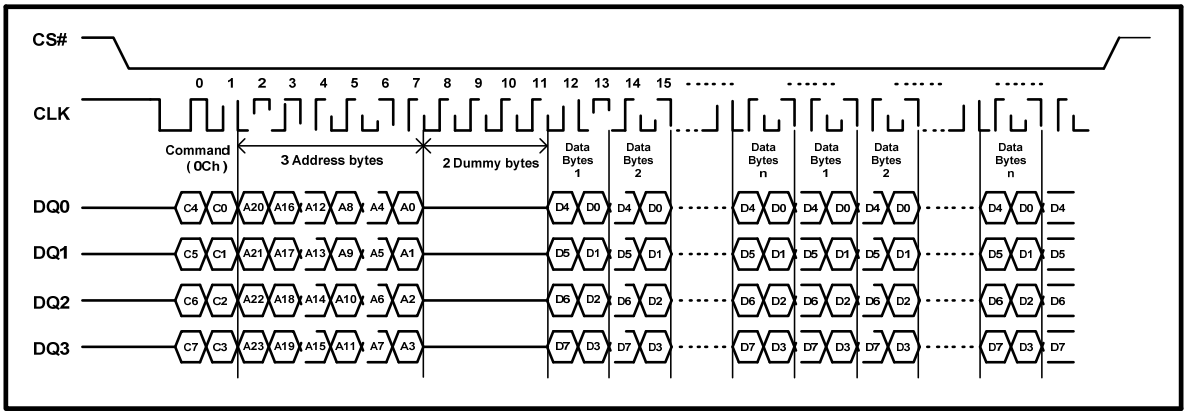


Figure 23.1 Read Burst Instruction Sequence Diagram in QPI mode

Page Program (PP) (02h)

The Page Program (PP) instruction allows bytes to be programmed in the memory. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Page Program (PP) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, three address bytes and at least one data byte on Serial Data Input (DI). If the 8 least significant address bits (A7-A0) are not all zero, all transmitted data that goes beyond the end of the current page are programmed from the start address of the same page (from the address whose 8 least significant bits (A7-A0) are all zero). Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 24. If more than 256 bytes are sent to the device, previously latched data are discarded and the last 256 data bytes are guaranteed to be programmed correctly within the same page. If less than 256 Data bytes are sent to device, they are correctly programmed at the requested addresses without having any effects on the other bytes of the same page.

Chip Select (CS#) must be driven High after the eighth bit of the last data byte has been latched in, otherwise the Page Program (PP) instruction is not executed.

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

A Page Program (PP) instruction applied to a page which is protected by the Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits (see Table 3) is not executed.

The instruction sequence is shown in Figure 24.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

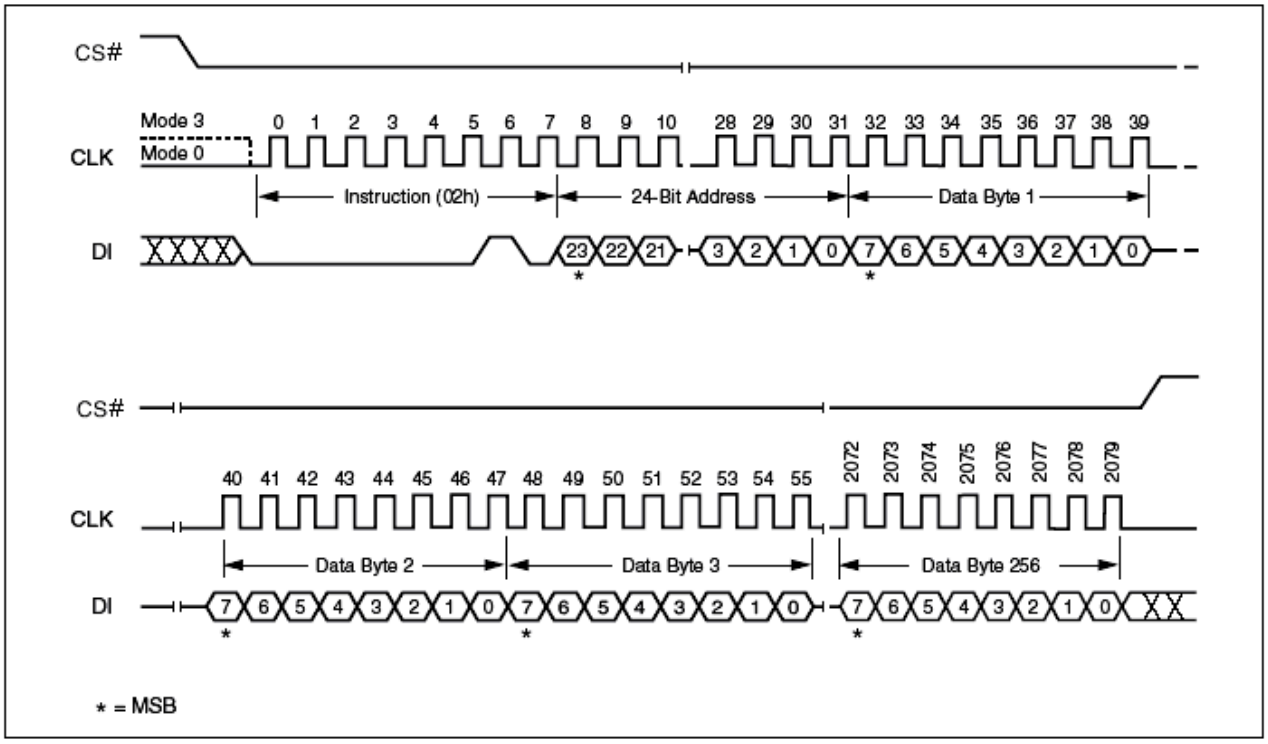


Figure 24. Page Program Instruction Sequence Diagram

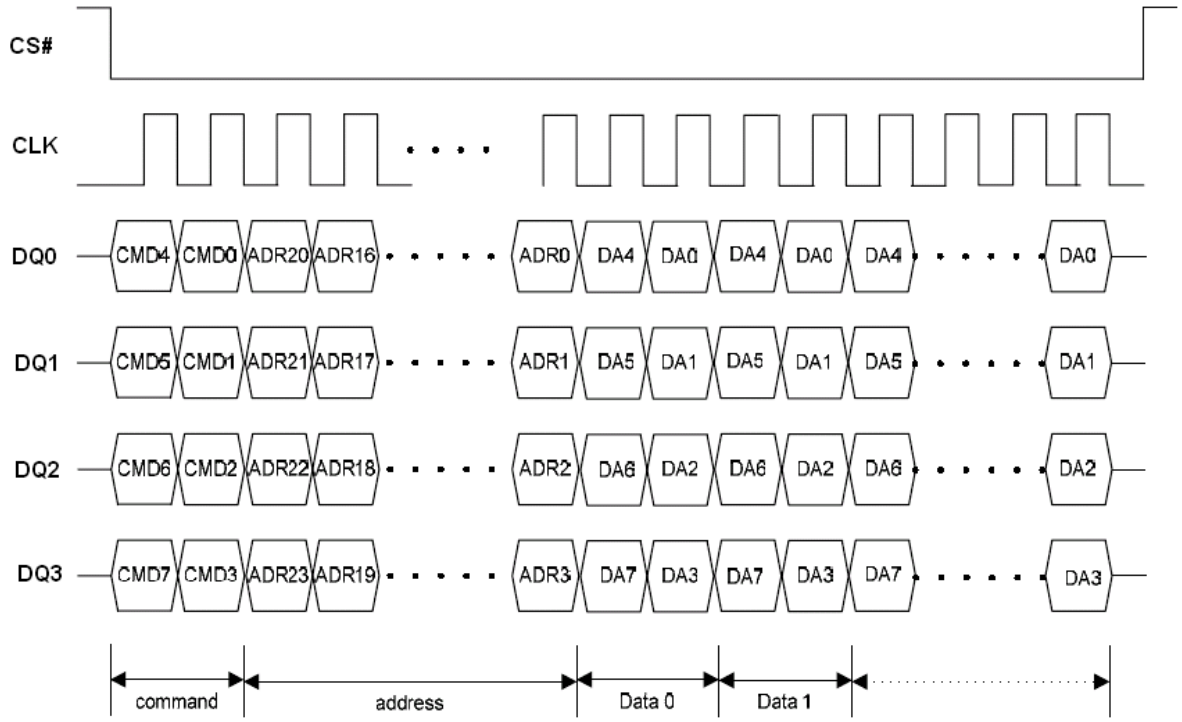


Figure 24.1 Program Instruction Sequence in QPI Mode

Quad Input Page Program (QPP) (32h)

The Quad Page Program (QPP) instruction allows up to 256 bytes of data to be programmed at previously erased (FFh) memory locations using four pins: DQ_0, DQ_1, DQ_2 and DQ_3. The Quad Page Program can improve performance for PROM Programmer and applications that have slow clock speeds < 5MHz. Systems with faster clock speed will not realize much benefit for the Quad Page Program instruction since the inherent page program time is much greater than the time it take to clock-in the data.

To use Quad Page Program (QPP) the WP# & Hold# Disable (WHDIS) bit in Status Register must be set to 1. A Write Enable instruction must be executed before the device will accept the Quad Page Program (QPP) instruction (SR.1, WEL=1). The instruction is initiated by driving the CS# pin low then shifting the instruction code "32h" followed by a 24-bit address (A23-A0) and at least one data byte, into the IO pins. The CS# pin must be held low for the entire length of the instruction while data is being sent to the device. All other functions of Quad Page Program (QPP) are identical to standard Page Program. The Quad Page Program (QPP) instruction sequence is shown in Figure 25.

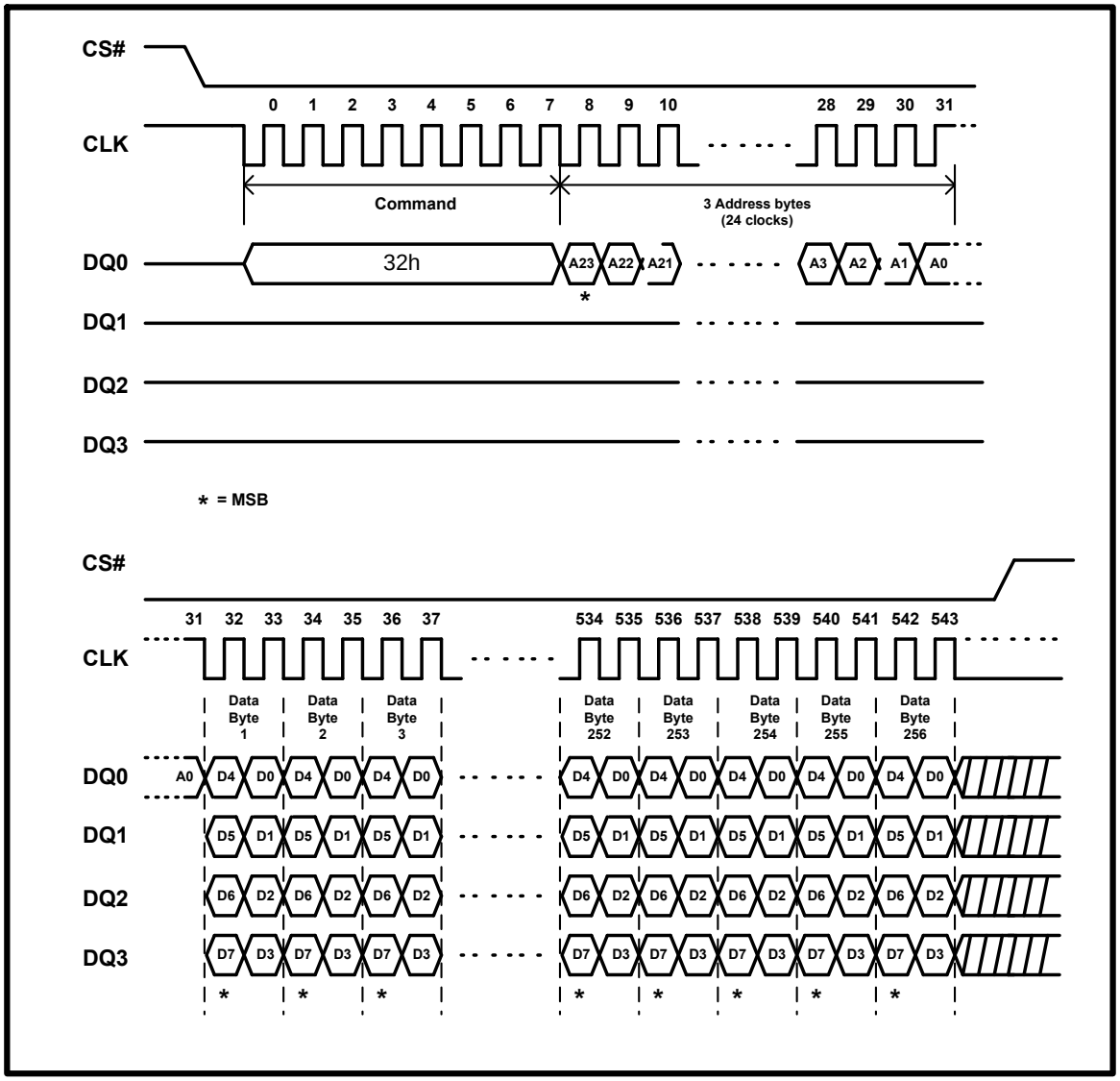


Figure 25. Quad Input Page Program Instruction Sequence Diagram (SPI Mode only)

Write Suspend (B0h)

Write Suspend allows the interruption of Sector Erase, Block Erase or Page Program operations in order to erase, program, or read data in another portion of memory. The original operation can be continued with Write Resume command. The instruction sequence is shown in Figure 26.

Only one write operation can be suspended at a time; if an operation is already suspended, the device will ignore the Write Suspend command. Write Suspend during Chip Erase is ignored; Chip Erase is not a valid command while a write is suspended. During write suspend, if the block protect area is modified by changing the Status Register 1 and Status Register 2, the suspended write operation will be aborted.

Suspend to suspend ready timing: 20us.
Resume to another suspend timing: 1ms.

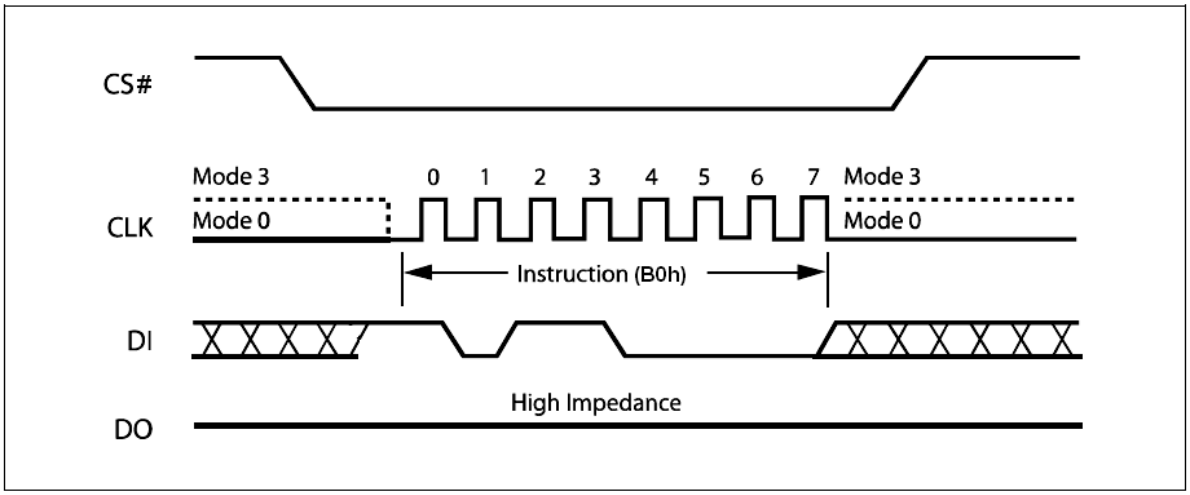


Figure 26. Write Suspend Instruction Sequence Diagram

Write Suspend During Sector Erase or Block Erase

Issuing a Write Suspend instruction during Sector Erase or Block Erase allows the host to program or read any sector that was not being erased. The device will ignore any programming commands pointing to the suspended sector(s). Any attempt to read from the suspended sector(s) will out put unknown data because the Sector or Block Erase will be incomplete.

To execute a Write Suspend operation, the host drives CS# low, sends the Write Suspend command cycle (B0h), then drives CS# high. A cycle is two nibbles long, most significant nibble first. The Suspend Status register indicates that the erase has been suspended by changing the WSE 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 Suspend Status register or after issue program suspend command, latency time 20us is needed before issue another command. For “Suspend to Read”, “Resume to Read”, “Resume to Suspend” timing specification please note Figure 27.1, 27.2 and 27.3.

Write Suspend During Page Programming

Issuing a Write Suspend instruction during Page Programming allows the host to erase or read any sector that is not being programmed. Erase commands pointing to the suspended sector(s) will be ignored. Any attempt to read from the suspended page will output unknown data because the program will be incomplete.

To execute a Write Suspend operation, the host drives CS# low, sends the Write Suspend command cycle (B0h), then drives CS# high. A cycle is two nibbles long, most significant nibble first. The Suspend Status register indicates that the programming has been suspended by changing the WSP 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 Suspend Status register or wait after issue program suspend command, latency time 20us is needed before issue another command. For "Suspend to Read", "Resume to Read", "Resume to Suspend" timing specification please note Figure 27.1, 27.2 and 27.3.

The instruction sequence is shown in Figure 28.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

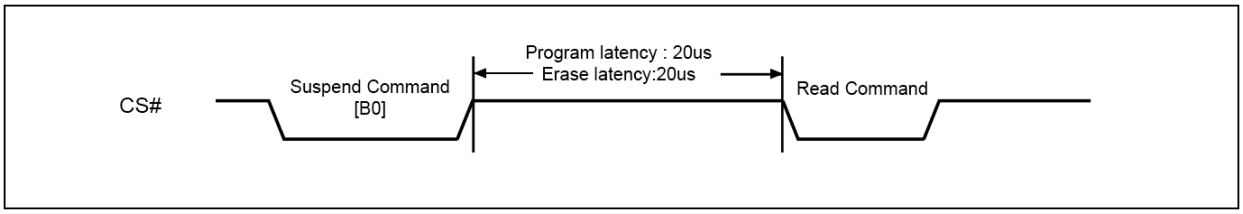


Figure 27.1 Suspend to Read Latency

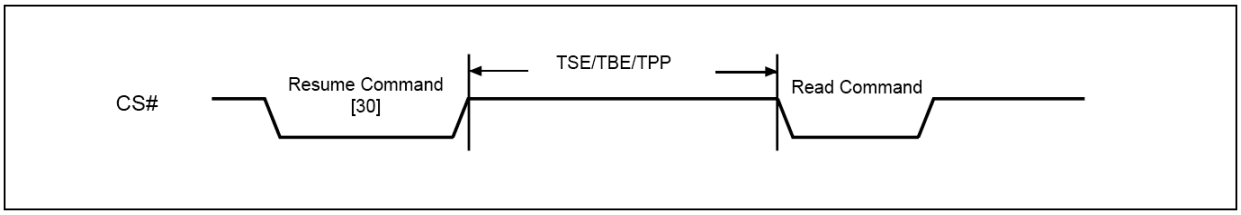


Figure 27.2 Resume to Read Latency

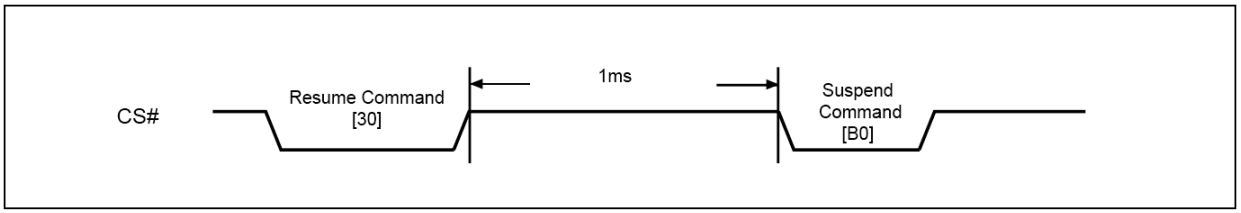


Figure 27.3 Resume to Suspend Latency

Write Resume (30h)

Write Resume restarts a Write command that was suspended, and changes the suspend status bit in the Suspend Status register (WSE or WSP) back to "0".

The instruction sequence is shown in Figure 28. To execute a Write Resume operation, the host drives CS# low, sends the Write Resume command cycle (30h), then drives CS# 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 Suspend 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}. Resume to another suspend operation requires latency time of 1ms.

The instruction sequence is shown in Figure 28.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

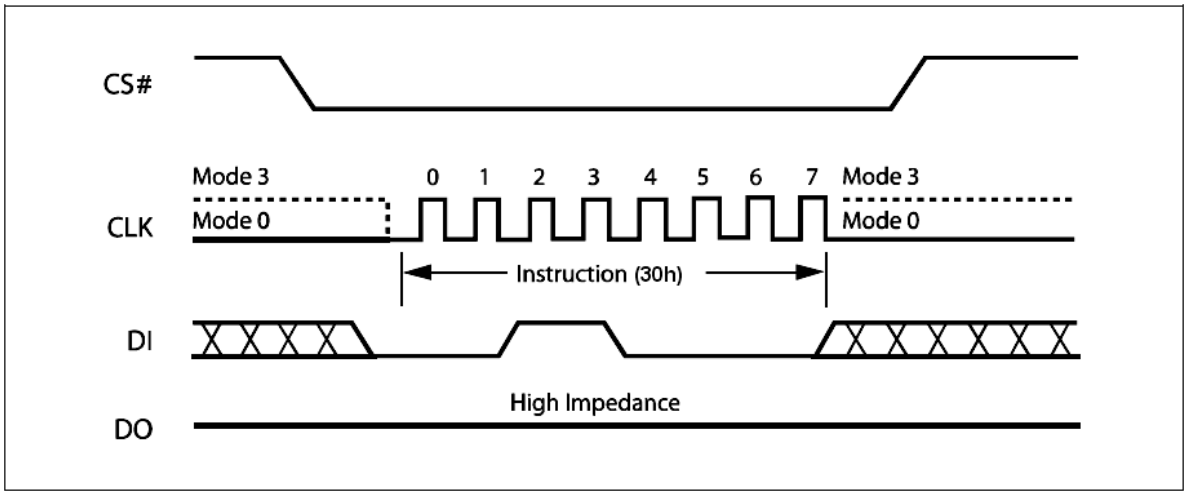


Figure 28. Write Resume Instruction Sequence Diagram

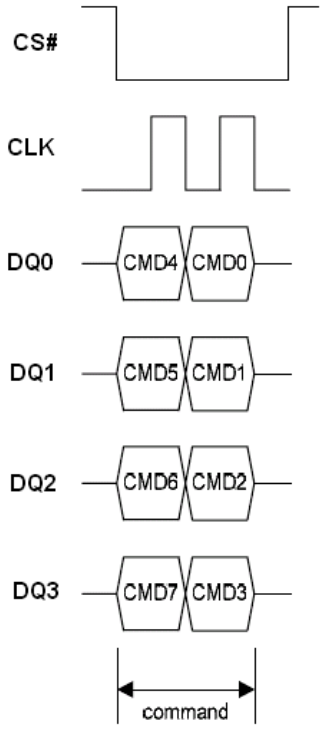


Figure 28.1 Write Suspend/Resume Instruction Sequence in QPI Mode

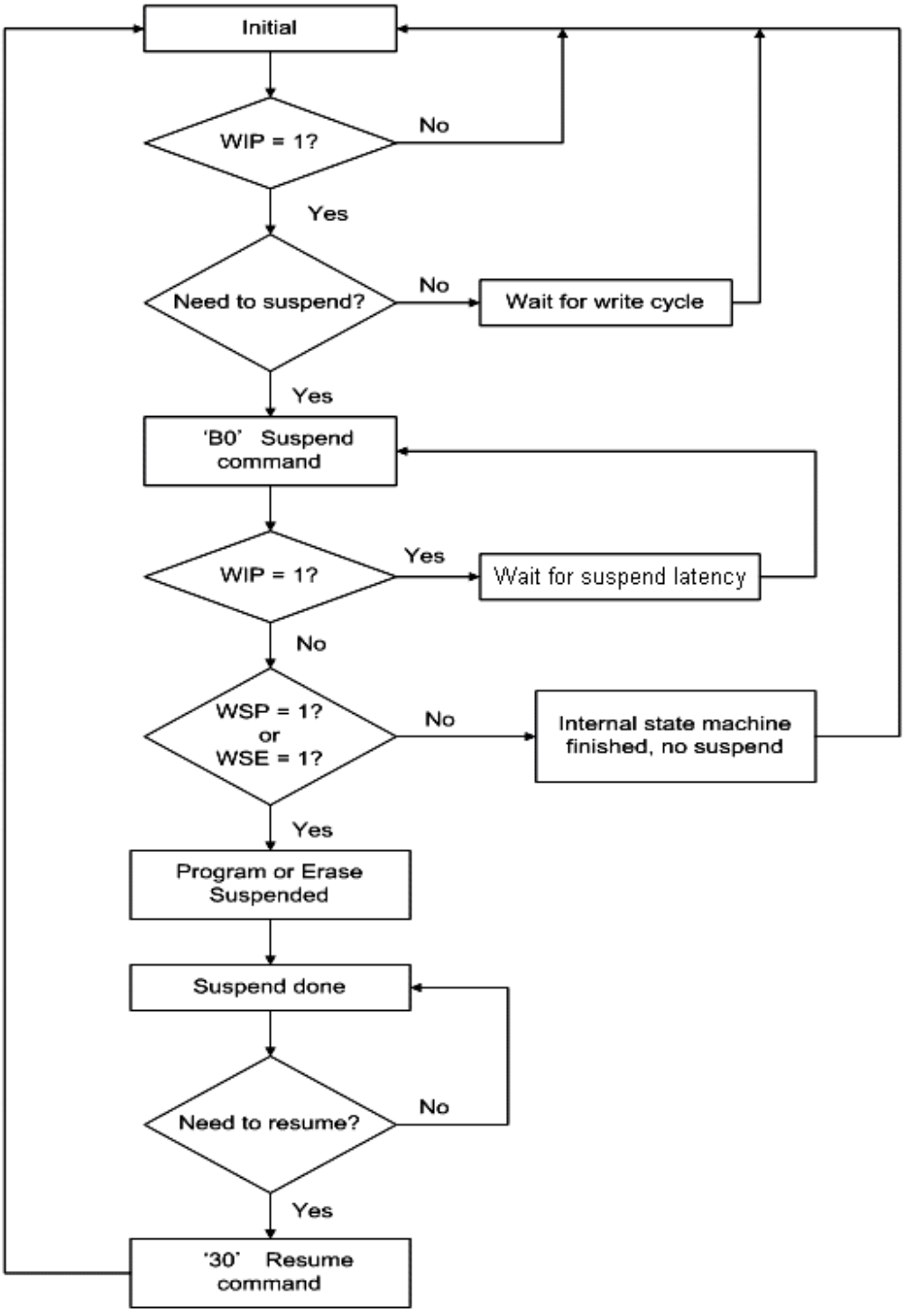


Figure 29. Write Suspend/Resume Flow

Note:

1. The 'WIP' can be either checked by command '09'or '05' polling.
2. 'Wait for write cycle' can be referring to maximum write cycle time or polling the WIP.
3. 'Wait for suspend latency', after issue program suspend command, latency time 20us is needed before issue another command or polling the WIP.
4. The 'WES' and 'WSE' can be checked by command '09' polling.

5. 'Suspend done' means the chip can do further operations allowed by suspend spec.

Sector Erase (SE) (20h)

The Sector Erase (SE) instruction sets to 1 (FFh) all bits inside the chosen sector. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Sector Erase (SE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, and three address bytes on Serial Data Input (DI). Any address inside the Sector (see Table 2) is a valid address for the Sector Erase (SE) instruction. Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 30. Chip Select (CS#) must be driven High after the eighth bit of the last address byte has been latched in, otherwise the Sector Erase (SE) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Sector Erase cycle (whose duration is t_{SE}) is initiated. While the Sector Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Sector Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Sector Erase (SE) instruction applied to a sector which is protected by the Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits (see Table 3) is not executed.

The instruction sequence is shown in Figure 32.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

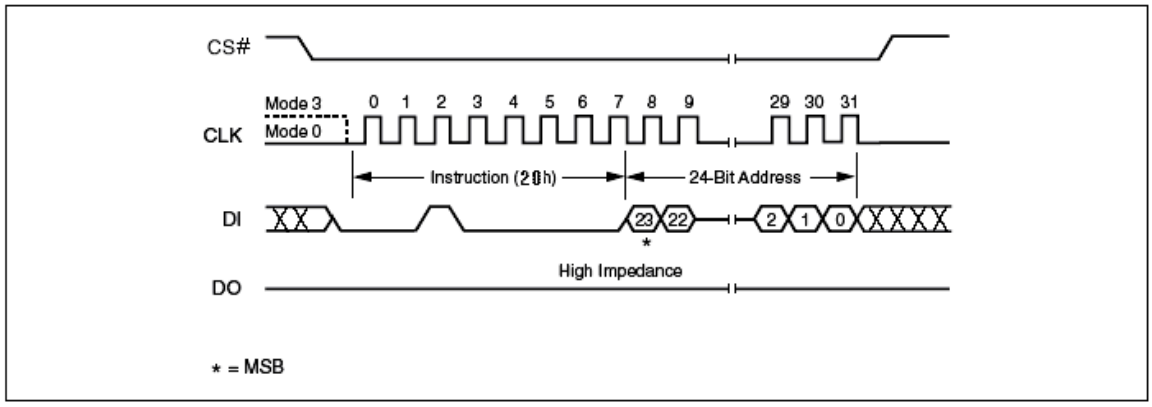


Figure 30. Sector Erase Instruction Sequence Diagram

32KB Half Block Erase (HBE) (52h)

The Half Block Erase (HBE) instruction sets to 1 (FFh) all bits inside the chosen block. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Half Block Erase (HBE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, and three address bytes on Serial Data Input (DI). Any address inside the Block (see Table 2) is a valid address for the Half Block Erase (HBE) instruction. Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 31. Chip Select (CS#) must be driven High after the eighth bit of the last address byte has been latched in, otherwise the Half Block Erase (HBE) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Block Erase cycle (whose duration is t_{HBE}) is initiated. While the Half Block Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Half Block Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Half Block Erase (HBE) instruction applied to a block which is protected by the Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits (see Table 3) is not executed.

The instruction sequence is shown in Figure 32.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

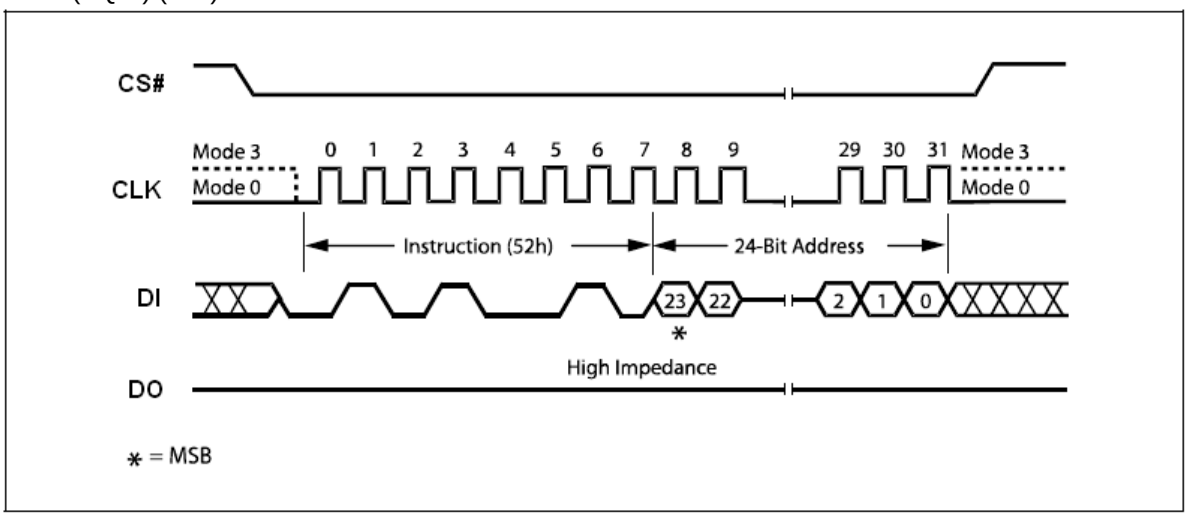


Figure 31. 32KB Half Block Erase Instruction Sequence Diagram

64KB Block Erase (BE) (D8h)

The Block Erase (BE) instruction sets to 1 (FFh) all bits inside the chosen block. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Block Erase (BE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, and three address bytes on Serial Data Input (DI). Any address inside the Block (see Table 2) is a valid address for the Block Erase (BE) instruction. Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 32. Chip Select (CS#) must be driven High after the eighth bit of the last address byte has been latched in, otherwise the Block Erase (BE) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Block Erase cycle (whose duration is t_{BE}) is initiated. While the Block Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Block Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Block Erase (BE) instruction applied to a block which is protected by the Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits (see Table 3) is not executed.

The instruction sequence is shown in Figure 32.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

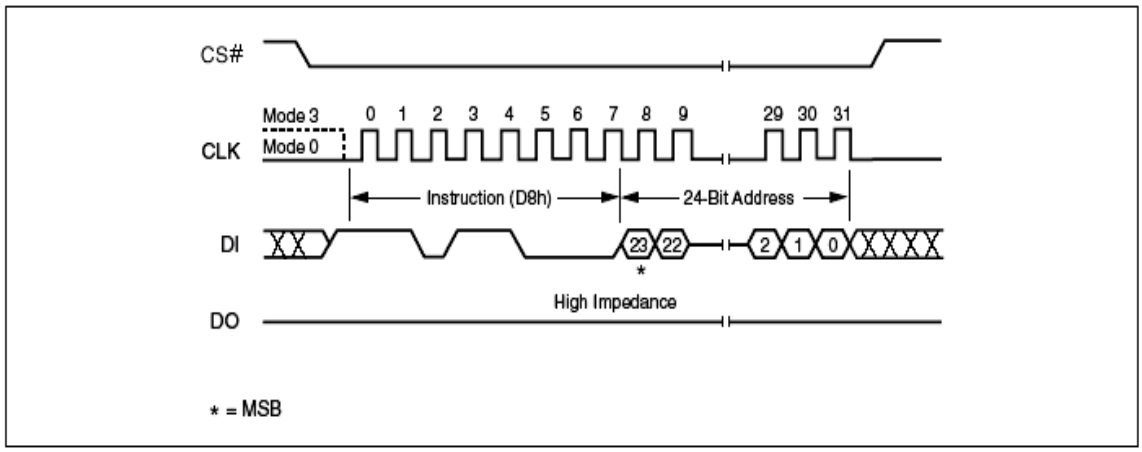


Figure 32. Block Erase Instruction Sequence Diagram

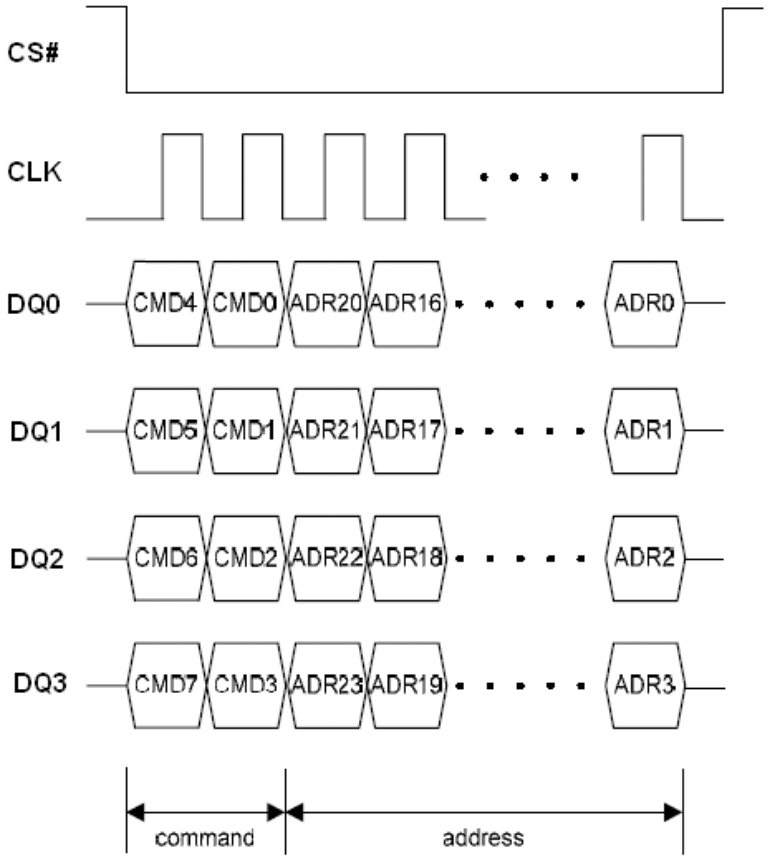


Figure 32.1 Block/Sector Erase Instruction Sequence in QPI Mode

Chip Erase (CE) (C7h/60h)

The Chip Erase (CE) instruction sets all bits to 1 (FFh). Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Chip Erase (CE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code on Serial Data Input (DI). Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 33. Chip Select (CS#) must be driven High after the eighth bit of the instruction code has been latched in, otherwise the Chip Erase instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Chip Erase cycle (whose duration is t_{CE}) is initiated. While the Chip Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Chip Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

The Chip Erase (CE) instruction is executed only if all Block Protect or Dynamic Block Protect (SR.5, SR.4, SR.3, SR.2) bits are 0. The Chip Erase (CE) instruction is ignored if one, or more blocks are protected.

The instruction sequence is shown in Figure 33.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

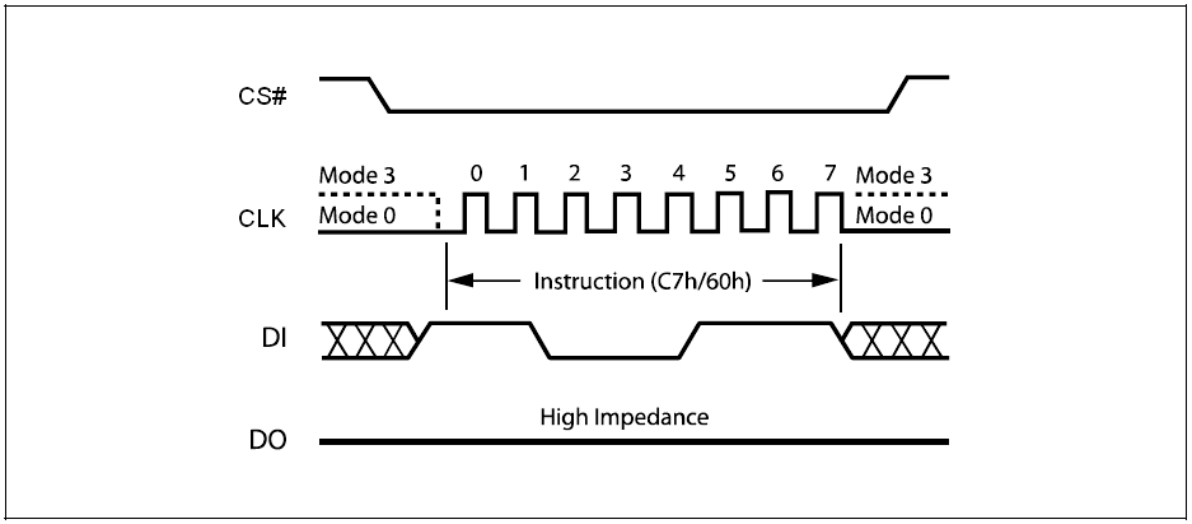


Figure 33. Chip Erase Instruction Sequence Diagram

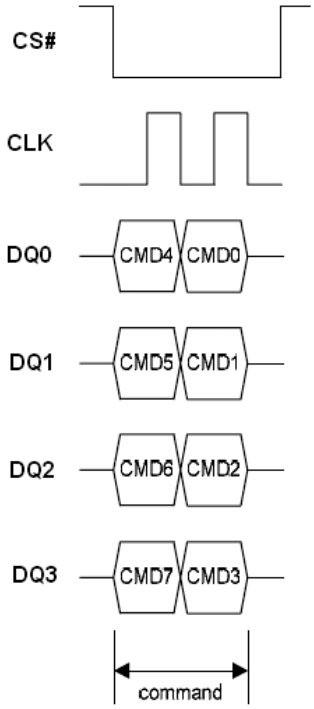


Figure 33.1 Chip Erase Sequence in QPI Mode

Deep Power-down (DP) (B9h)

Executing the Deep Power-down (DP) instruction is the only way to put the device in the lowest consumption mode (the Deep Power-down mode). It can also be used as an extra software protection mechanism, while the device is not in active use, since in this mode, the device ignores all Write, Program and Erase instructions.

Driving Chip Select (CS#) High deselects the device, and puts the device in the Standby mode (if there is no internal cycle currently in progress). But this mode is not the Deep Power-down mode. The Deep Power-down mode can only be entered by executing the Deep Power-down (DP) instruction, to reduce the standby current (from I_{CC1} to I_{CC2}, as specified in Table 17.)

Once the device has entered the Deep Power-down mode, all instructions are ignored except the Release from Deep Power-down and Read Device ID (RDI) instruction. This releases the device from this mode. The Release from Deep Power-down and Read Device ID (RDI) instruction also allows the Device ID of the device to be output on Serial Data Output (DO).

The Deep Power-down mode automatically stops at Power-down, and the device always Powers-up in the Standby mode. The Deep Power-down (DP) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code on Serial Data Input (DI). Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 34. Chip Select (CS#) must be driven High after the eighth bit of the instruction code has been latched in, otherwise the Deep Power-down (DP) instruction is not executed. As soon as Chip Select (CS#) is driven High, it requires a delay of t_{DP} before the supply current is reduced to I_{CC2} and the Deep Power-down mode is entered.

Any Deep Power-down (DP) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

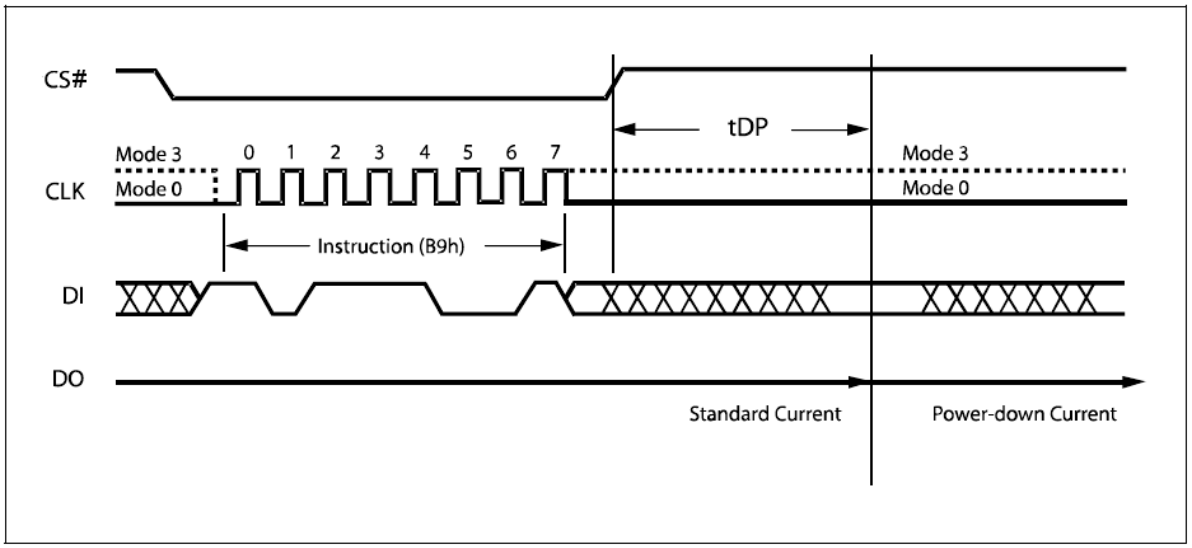


Figure 34. Deep Power-down Instruction Sequence Diagram

Release from Deep Power-down and Read Device ID (RDI)

Once the device has entered the Deep Power-down mode, all instructions are ignored except the Release from Deep Power-down and Read Device ID (RDI) instruction. Executing this instruction takes the device out of the Deep Power-down mode.

Please note that this is not the same as, or even a subset of, the JEDEC 16-bit Electronic Signature that is read by the Read Identifier (RDID) instruction. The old-style Electronic Signature is supported for reasons of backward compatibility, only, and should not be used for new designs. New designs should, instead, make use of the JEDEC 16-bit Electronic Signature, and the Read Identifier (RDID) instruction.

When used only to release the device from the power-down state, the instruction is issued by driving the CS# pin low, shifting the instruction code “ABh” and driving CS# high as shown in Figure 35. After the time duration of t_{RES1} (See AC Characteristics) the device will resume normal operation and other instructions will be accepted. The CS# pin must remain high during the t_{RES1} time duration.

When used only to obtain the Device ID while not in the power-down state, the instruction is initiated by driving the CS# pin low and shifting the instruction code “ABh” followed by 3-dummy bytes. The Device ID bits are then shifted out on the falling edge of CLK with most significant bit (MSB) first as shown in Figure 36. The Device ID value for the EN25QH64A are listed in Table 6. The Device ID can be read continuously. The instruction is completed by driving CS# high.

When Chip Select (CS#) is driven High, the device is put in the Stand-by Power mode. If the device was not previously in the Deep Power-down mode, the transition to the Stand-by Power mode is immediate. If the device was previously in the Deep Power-down mode, though, the transition to the Standby Power mode is delayed by t_{RES2}, and Chip Select (CS#) must remain High for at least t_{RES2} (max), as specified in Table 19. Once in the Stand-by Power mode, the device waits to be selected, so that it can receive, decode and execute instructions.

Except while an Erase, Program or Write Status Register cycle is in progress, the Release from Deep Power-down and Read Device ID (RDI) instruction always provides access to the 8bit Device ID of the device, and can be applied even if the Deep Power-down mode has not been entered.

Any Release from Deep Power-down and Read Device ID (RDI) instruction while an Erase, Program or Write Status Register cycle is in progress, is not decoded, and has no effect on the cycle that is in progress.

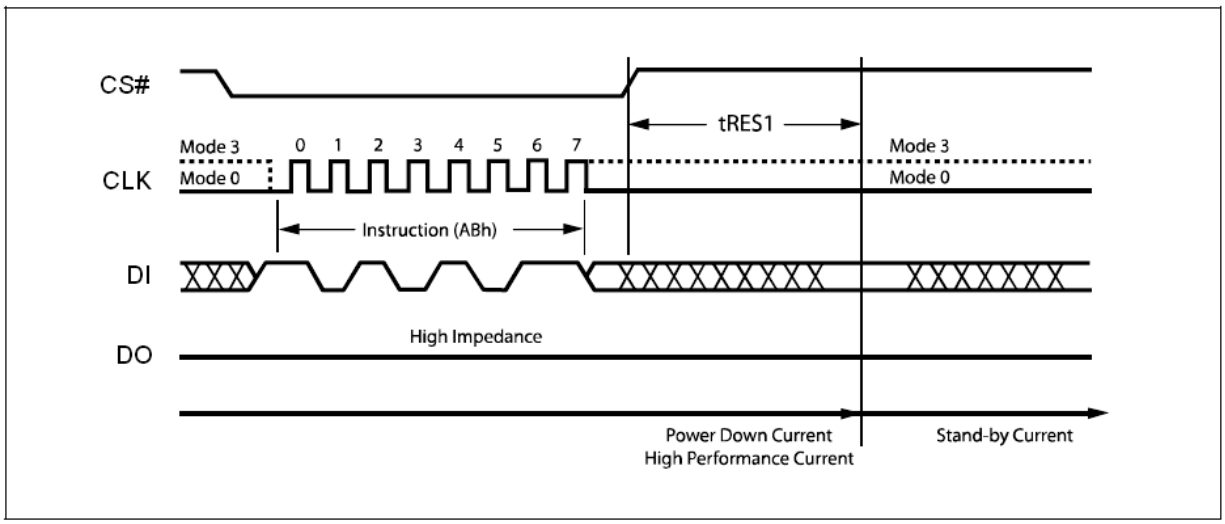


Figure 35. Release Power-down Instruction Sequence Diagram

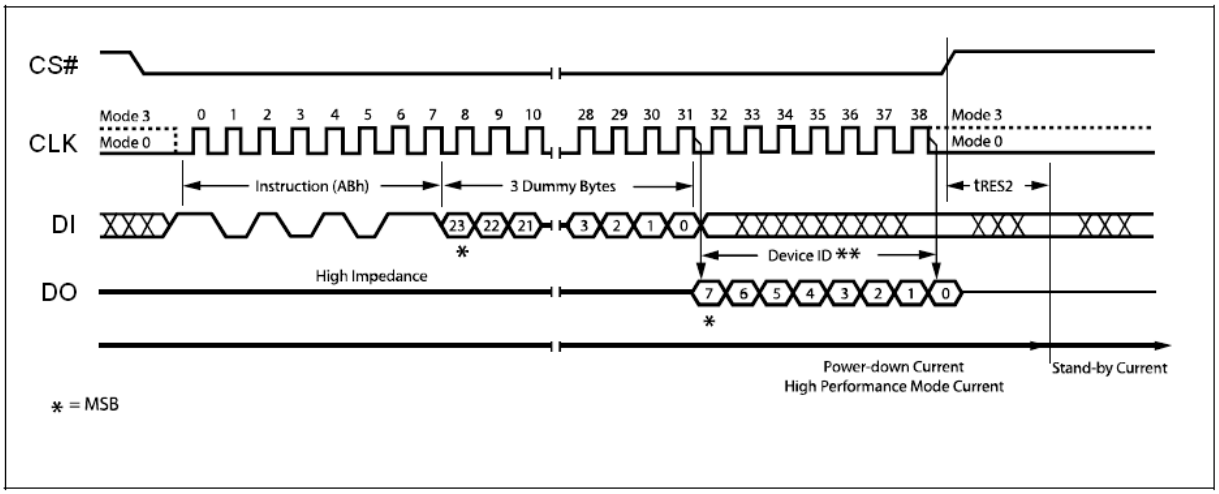


Figure 36. Release Power-down / Device ID Instruction Sequence Diagram

Read Manufacturer / Device ID (90h)

The Read Manufacturer/Device ID instruction is an alternative to the Release from Power-down / Device ID instruction that provides both the JEDEC assigned manufacturer ID and the specific device ID.

The Read Manufacturer/Device ID instruction is very similar to the Release from Power-down / Device ID instruction. The instruction is initiated by driving the CS# pin low and shifting the instruction code "90h" followed by a 24-bit address (A23-A0) of 000000h. After which, the Manufacturer ID for Eon (1Ch) and the Device ID are shifted out on the falling edge of CLK with most significant bit (MSB) first as shown in Figure 37. The Device ID values for the EN25QH64A are listed in Table 6. If the 24-bit address is initially set to 000001h the Device ID will be read first

The instruction sequence is shown in Figure 37.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

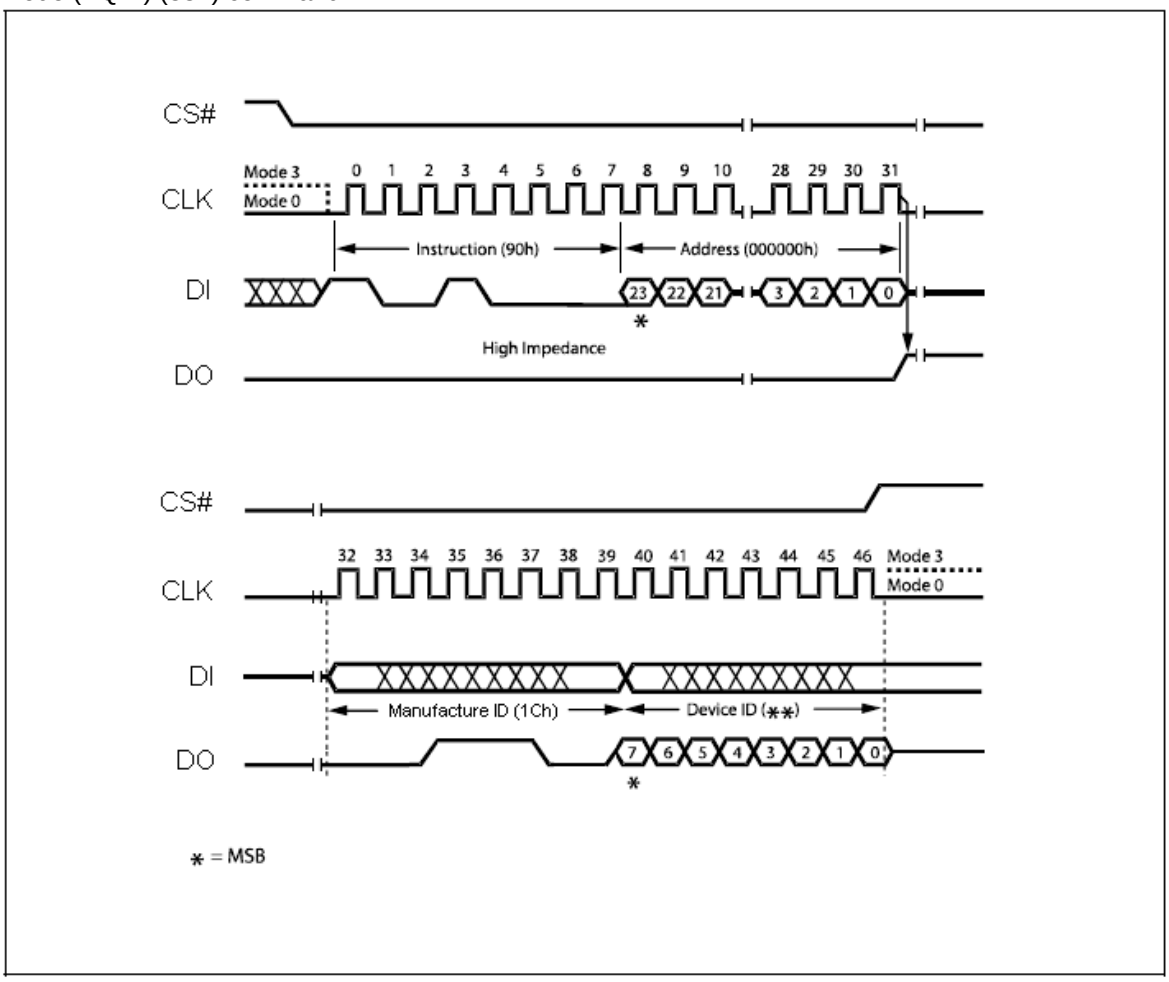


Figure 37. Read Manufacturer / Device ID Diagram

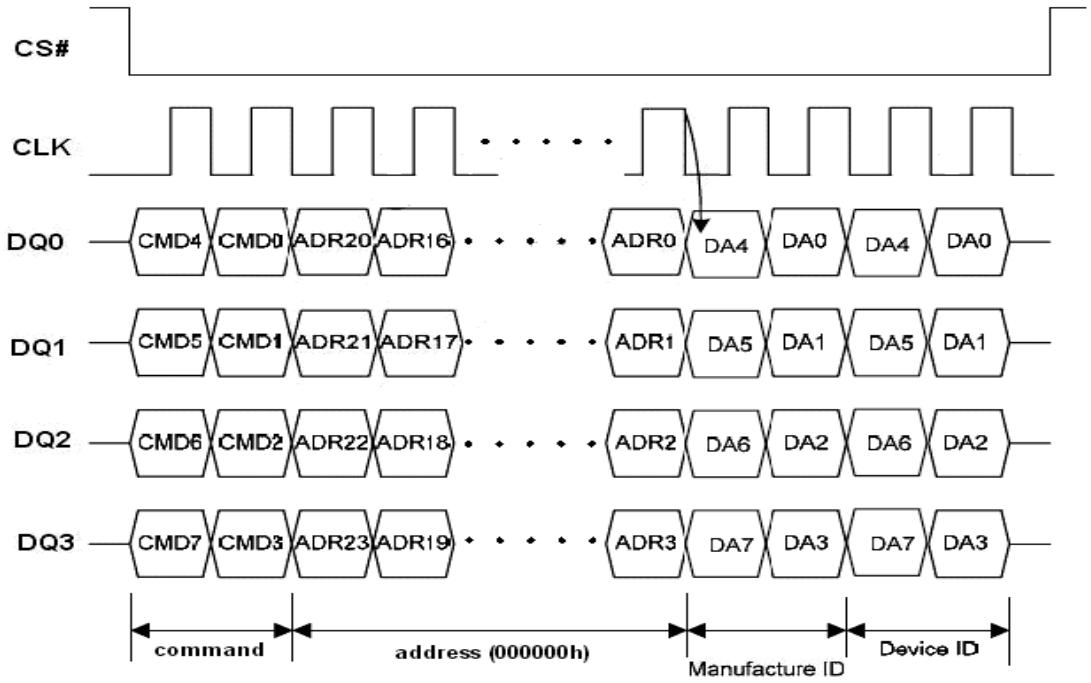


Figure 37.1. Read Manufacturer / Device ID Diagram in QPI Mode

Read Identification (RDID) (9Fh)

The Read Identification (RDID) instruction allows the 8-bit manufacturer identification to be read, followed by two bytes of device identification. The device identification indicates the memory type in the first byte , and the memory capacity of the device in the second byte .

Any Read Identification (RDID) instruction while an Erase or Program cycle is in progress, is not decoded, and has no effect on the cycle that is in progress. The Read Identification (RDID) instruction should not be issued while the device is in Deep Power down mode.

The device is first selected by driving Chip Select Low. Then, the 8-bit instruction code for the instruction is shifted in. This is followed by the 24-bit device identification, stored in the memory, being shifted out on Serial Data Output , each bit being shifted out during the falling edge of Serial Clock . The instruction sequence is shown in Figure 38. The Read Identification (RDID) instruction is terminated by driving Chip Select High at any time during data output.

When Chip Select is driven High, the device is put in the Standby Power mode. Once in the Standby Power mode, the device waits to be selected, so that it can receive, decode and execute instructions.

The instruction sequence is shown in Figure 38.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

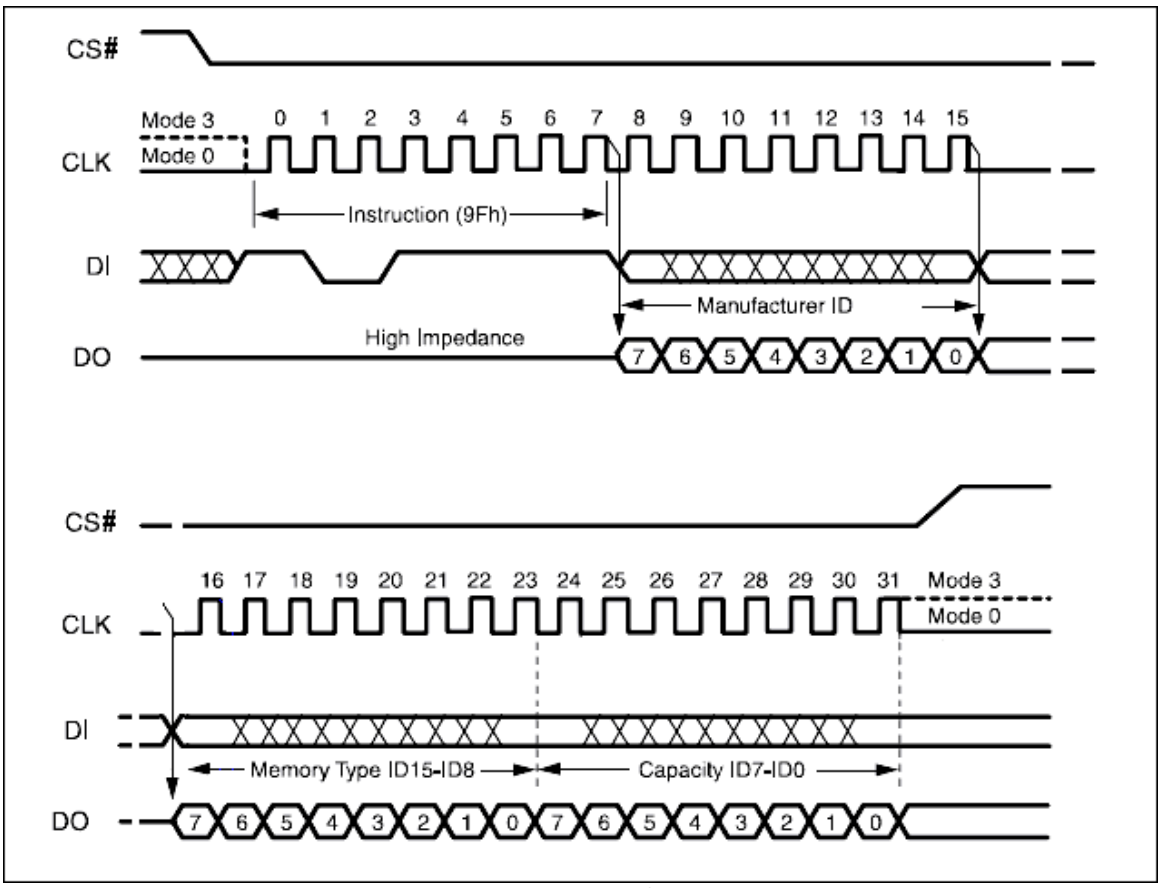


Figure 38. Read Identification (RDID)

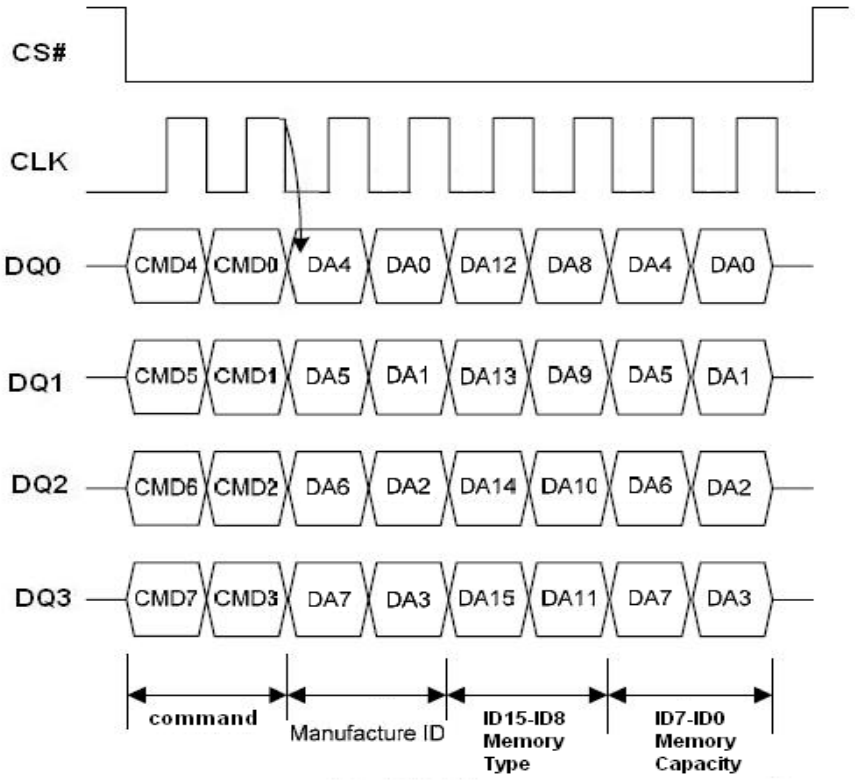


Figure 38.1. Read Identification (RDID) in QPI Mode

Enter OTP Mode (3Ah)

This Flash has an extra 512 bytes OTP sector, user must issue ENTER OTP MODE command to read, program or erase OTP sector. After entering OTP mode, the OTP sector is mapping to sector 2047, **SRP bit** becomes OTP_LOCK bit. The Chip Erase, Block Erase and Half Block Erase commands are also disabled.

In OTP mode, user can read other sectors, but program/erase OTP sector only allowed when OTP_LOCK bit equal to '0'. The OTP sector can ***only*** be erased by Sector Erase (20h) command.

Table 12. OTP Sector Address

Sector	Sector Size	Address Range
2047	512 byte	7FF000h – 7FF1FFh

Note: The OTP sector is mapping to sector 2047

This Flash supports OTP mode to enhance the data protection. In OTP mode, the Status Register S7 bit is served as OTP_LOCK bit; S6 bit is served as TB bit; S5 bit is served as DBP Boot Lock bit; S4 bit is served as 4KB BL bit, S3 bit is served as EBL bit. WRSR command is used to program OTP_LOCK bit, TB bit, DBP Boot Lock bit, 4KB BL bit and EBL bit to '1', but these bits only can be programmed once. They can be read by RDSR command. User can use the Enter OTP mode (3Ah) command to enter OTP mode and WRDI (04h) command to exit OTP mode.

The instruction sequence is shown in Figure 39.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

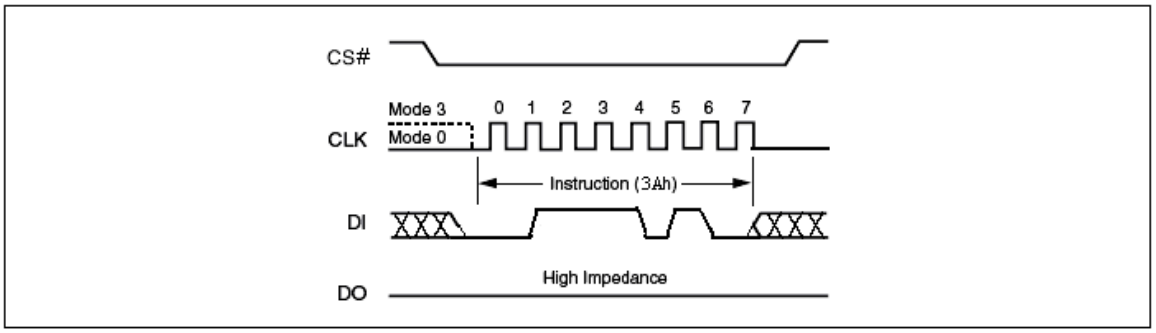


Figure 39. Enter OTP Mode

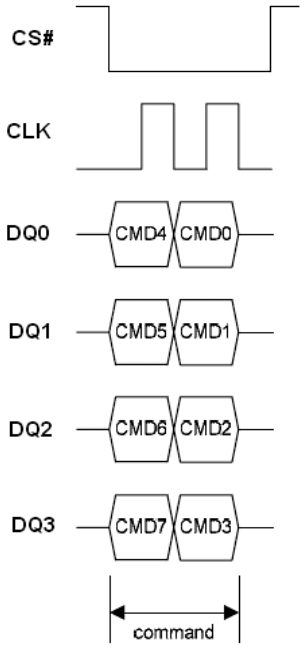


Figure 39.1 Enter OTP Mode Sequence in QPI Mode

Read SFDP Mode and Unique ID Number (5Ah)

Read SFDP Mode

EN25QH64A features Serial Flash Discoverable Parameters (SFDP) mode. Host system can retrieve the operating characteristics, structure and vendor specified information such as identifying information, memory size, operating voltage and timing information of this device by SFDP mode.
The device is first selected by driving Chip Select (CS#) Low. The instruction code for the Read SFDP Mode is followed by a 3-byte address (A23-A0) and a dummy byte, each bit being latched-in during the rising edge of Serial Clock (CLK). Then the memory contents, at that address, is shifted out on Serial Data Output (DO), each bit being shifted out, at a maximum frequency F_R, during the falling edge of Serial Clock (CLK).
The instruction sequence is shown in Figure 40. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. The whole memory can, therefore, be read with a single Serial Flash Discoverable Parameters (SFDP) instruction. When the highest address is reached, the address counter rolls over to 0x00h, allowing the read sequence to be continued indefinitely. The Serial Flash Discoverable Parameters (SFDP) instruction is terminated by driving Chip Select (CS#) High. Chip Select (CS#) can be driven High at any time during data output. Any Read Data Bytes at Serial Flash Discoverable Parameters (SFDP) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

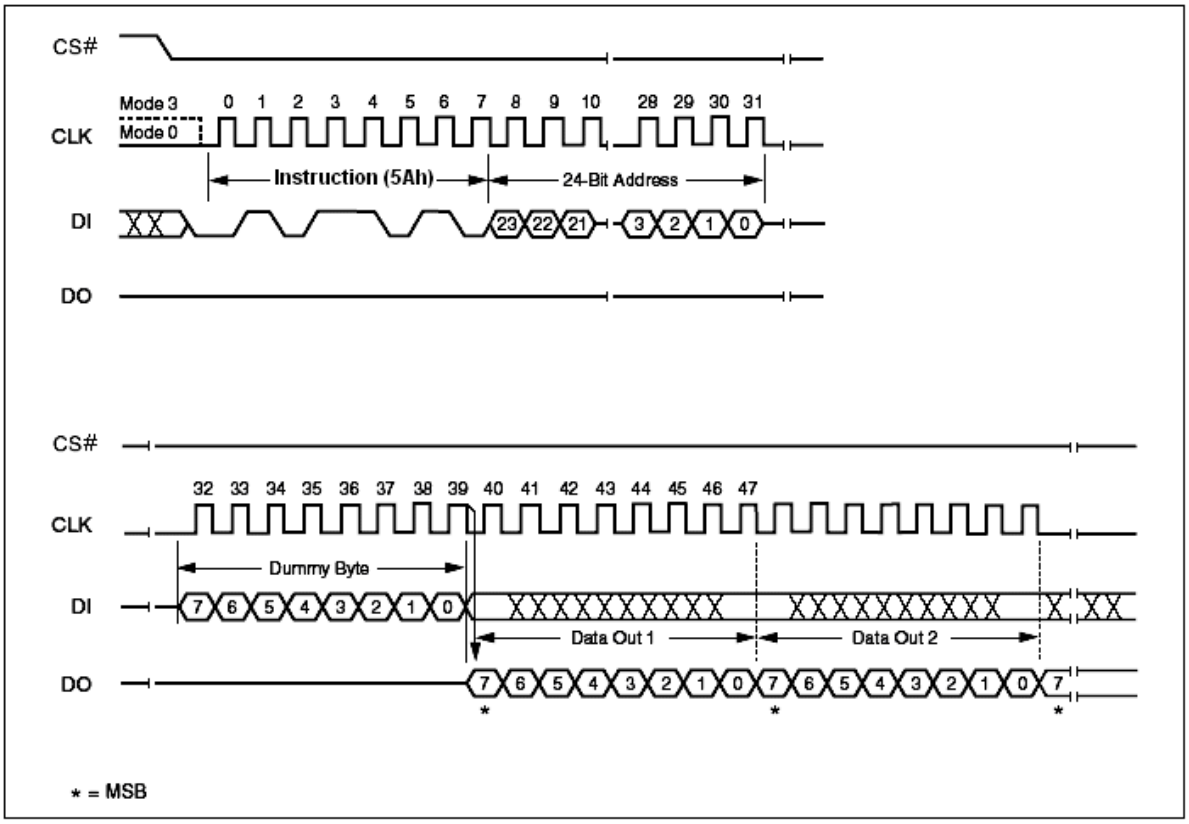


Figure 40. Read SFDP Mode Instruction Sequence Diagram

Table 13. Serial Flash Discoverable Parameters (SFDP) Signature and Parameter Identification Data Value (Advanced Information)

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
SFDP Signature	00h	07 : 00	53h	Signature [31:0]: Hex: 50444653
	01h	15 : 08	46h	
	02h	23 : 16	44h	
	03h	31 : 24	50h	
SFDP Minor Revision Number	04h	07 : 00	00h	Star from 0x00
SFDP Major Revision Number	05h	15 : 08	01h	Star from 0x01
Number of Parameter Headers (NPH)	06h	23 : 16	00h	1 parameter header
Unused	07h	31 : 24	FFh	Reserved
ID Number	08h	07 : 00	00h	JEDEC ID
Parameter Table Minor Revision Number	09h	15 : 08	00h	Star from 0x00
Parameter Table Major Revision Number	0Ah	23 : 16	01h	Star from 0x01
Parameter Table Length (in DW)	0Bh	31 : 24	09h	9 DWORDs
Parameter Table Pointer (PTP)	0Ch	07 : 00	30h	000030h
	0Dh	15 : 08	00h	
	0Eh	23 : 16	00h	
Unused	0Fh	31 : 24	FFh	Reserved

Table 14. Parameter ID (0) (Advanced Information) 1/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Block / Sector Erase sizes Identifies the erase granularity for all Flash Components	30h	00	01h	00 = reserved 01 = 4KB erase 10 = reserved 11 = 64KB erase
		01		
Write Granularity		02	1h	0 = No, 1 = Yes
Write Enable Instruction Required for Writing to Volatile Status Register		03	00h	00 = N/A 01 = use 50h opcode 11 = use 06h opcode
Write Enable Opcode Select for Writing to Volatile Status Register		04		
Unused		05	111h	Reserved
		06		
		07		
4 Kilo-Byte Erase Opcode	31h	08	20h	4 KB Erase Support (FFh = not supported)
		09		
		10		
		11		
		12		
		13		
		14		
		15		
Supports (1-1-2) Fast Read Device supports single input opcode & address and dual output data Fast Read	32h	16	1h	0 = not supported 1 = supported
Address Byte Number of bytes used in addressing for flash arra write and erase.		17	00h	00 = 3-Byte 01 = 3- or 4-Byte (e.g. defaults to 3-Byte mode; enters 4-Byte mode on command) 10 = 4-Byte 11 = reserved
		18		
Supports Double Data Rate (DDR) Clocking Indicates the device supports some type of double transfer rate clocking.		19	0h	0 = not supported 1 = supported
Supports (1-2-2) Fast Read Device supports single input opcode, dual input address, and dual output data Fast Read		20	1h	0 = not supported 1 = supported
Supports (1-4-4) Fast Read Device supports single input opcode, quad input address, and quad output data Fast Read		21	1h	0 = not supported 1 = supported
Supports (1-1-4) Fast Read Device supports single input opcode & address and quad output data Fast Read		22	0h	0 = not supported 1 = supported
Unused		23	1h	Reserved
Unused	33h	24	FFh	Reserved
		25		
		26		
		27		
		28		
		29		
		30		
		31		

Table 14. Parameter ID (0) (Advanced Information) 2/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Flash Memory Density	37h : 34h	31 : 00	3FFFFFFh	64 Mbits

Table 14. Parameter ID (0) (Advanced Information) 3/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
(1-4-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	38h	00 01 02 03 04	1Fh	Configurable
Quad Input Address Quad Output (1-4-4) Fast Read Number of Mode Bits		05 06 07	010h	8 mode bits
(1-4-4) Fast Read Opcode Opcode for single input opcode, quad input address, and quad output data Fast Read.	39h	08 09 10 11 12 13 14 15	EBh	
(1-1-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Ah	16 17 18 19 20	00000h	Not supported
(1-1-4) Fast Read Number of Mode Bits		21 22 23	000h	Not supported
(1-1-4) Fast Read Opcode Opcode for single input opcode & address and quad output data Fast Read.	3Bh	31 : 24	FFh	Not supported

Table 14. Parameter ID (0) (Advanced Information) 4/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
(1-1-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Ch	00	01000h	8 dummy clocks
		01		
		02		
		03		
		04		
(1-1-2) Fast Read Number of Mode Bits		05	000h	Not supported
		06		
		07		
(1-1-2) Fast Read Opcode Opcode for single input opcode & address and dual output data Fast Read.	3Dh	15 : 08	3Bh	Not supported
(1-2-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Eh	16	00100h	4 dummy clocks
		17		
		18		
		19		
		20		
(1-2-2) Fast Read Number of Mode Bits		21	000h	Not supported
		22		
		23		
(1-2-2) Fast Read Opcode Opcode for single input opcode, dual input address, and dual output data Fast Read.	3Fh	31 : 24	BBh	Not supported

Table 14. Parameter ID (0) (Advanced Information) 5/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Supports (2-2-2) Fast Read Device supports dual input opcode & address and dual output data Fast Read.	40h	00	0h	0 = not supported 1 = supported
Reserved. These bits default to all 1's		01	111h	Reserved
		02		
		03		
Supports (4-4-4) Fast Read Device supports Quad input opcode & address and quad output data Fast Read.		04	1h	0 = not supported 1 = supported (EQPI Mode)
Reserved. These bits default to all 1's		05	111h	Reserved
		06		
		07		
Reserved. These bits default to all 1's	43h : 41h	31 : 08	FFh	Reserved

Table 14. Parameter ID (0) (Advanced Information) 6/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Reserved. These bits default to all 1's	45h : 44h	15 : 00	FFh	Reserved
(2-2-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	46h	16	00000h	Not supported
		17		
		18		
		19		
		20		
(2-2-2) Fast Read Number of Mode Bits		21	000h	Not supported
		22		
		23		
(2-2-2) Fast Read Opcode Opcode for dual input opcode & address and dual output data Fast Read.	47h	31 : 24	FFh	Not supported

Table 14. Parameter ID (0) (Advanced Information) 7/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Reserved. These bits default to all 1's	49h : 48h	15 : 00	FFh	Reserved
(4-4-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	4Ah	16	1Fh	Configurable
		17		
		18		
		19		
		20		
(4-4-4) Fast Read Number of Mode Bits		21	010h	8 mode bits
		22		
		23		
(4-4-4) Fast Read Opcode Opcode for quad input opcode/address, quad output data Fast Read.	4Bh	31 : 24	EBh	Must Enter EQPI Mode firstly

Table 14. Parameter ID (0) (Advanced Information) 8/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Sector Type 1 Size	4Ch	07 : 00	0Ch	4 KB
Sector Type 1 Opcode	4Dh	15 : 08	20h	
Sector Type 2 Size	4Eh	23 : 16	0Fh	32 KB
Sector Type 2 Opcode	4Fh	31 : 24	52h	

Table 14. Parameter ID (0) (Advanced Information) 9/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Sector Type 3 Size	50h	07 : 00	10h	64 KB
Sector Type 3 Opcode	51h	15 : 08	D8h	
Sector Type 4 Size	52h	23 : 16	00h	Not Supported
Sector Type 4 Opcode	53h	31 : 24	FFh	Not Supported

Read Unique ID Number

The Read Unique ID Number instruction accesses a factory-set read-only 96-bit number that is unique to each EN25QH64A device. The ID number can be used in conjunction with user software methods to help prevent copying or cloning of a system. The Read Unique ID instruction is initiated by driving the CS# pin low and shifting the instruction code “5Ah” followed by a three bytes of addresses, 0x80h, and one byte of dummy clocks. After which, the 96-bit ID is shifted out on the falling edge of CLK as shown in Figure 41.

Table 15. Unique ID Number

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Unique ID Number	80h : 8Bh	95 : 00	By die	

Power-up Timing

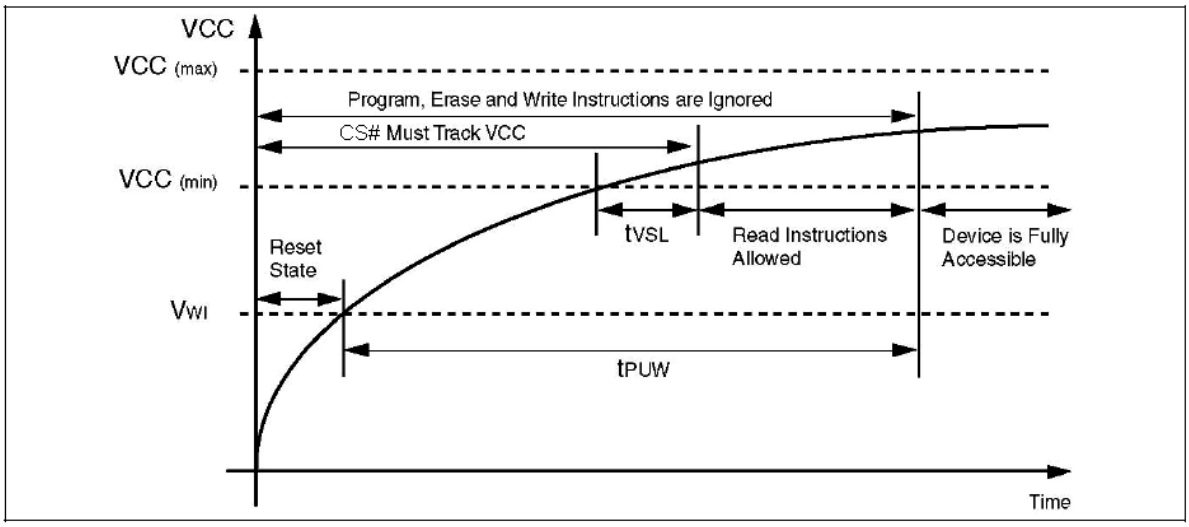


Figure 42. Power-up Timing

Table 16. Power-Up Timing and Write Inhibit Threshold

Symbol	Parameter	Min.	Max.	Unit
t_{VSL}(1)	VCC(min) to CS# low	10		µs
t_{PUW}(1)	Time delay to Write instruction	1	10	ms
V_{WI}(1)	Write Inhibit Voltage	1	2.5	V

Note:
1.The parameters are characterized only.
2. VCC (max.) is 3.6V and VCC (min.) is 2.7V

INITIAL DELIVERY STATE

The device is delivered with the memory array erased: all bits are set to 1 (each byte contains FFh). The Status Register contains 00h (all Status Register bits are 0).

Table 17. DC Characteristics

(T_a = - 40°C to 85°C; V_{CC} = 2.7-3.6V)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I_{LI}	Input Leakage Current				2	µA
I_{LO}	Output Leakage Current				2	µA
I_{CC1}	Standby Current	CS# = V_{CC}, V_{IN} = V_{SS} or V_{CC}			20	µA
I_{CC2}	Deep Power-down Current	CS# = V_{CC}, V_{IN} = V_{SS} or V_{CC}			20	µA
I_{CC3}	Operating Current (READ)	CLK = 0.1 V_{CC} / 0.9 V_{CC} at 104MHz, DQ = open		10	25	mA
		CLK = 0.1 V_{CC} / 0.9 V_{CC} at 33MHz, DQ = open		5	12	mA
		CLK = 0.1 V_{CC} / 0.9 V_{CC} at 104MHz, Quad Output Read, DQ = open		14	35	mA
		CLK = 0.1 V_{CC} / 0.9 V_{CC} at 33MHz, Quad Output Read, DQ = open		7	17	mA
I_{CC4}	Operating Current (PP)	CS# = V_{CC}		9	30	mA
I_{CC5}	Operating Current (WRSR)	CS# = V_{CC}			25	mA
I_{CC6}	Operating Current (SE)	CS# = V_{CC}		13	25	mA
I_{CC7}	Operating Current (BE)	CS# = V_{CC}		15	25	mA
V_{IL}	Input Low Voltage		– 0.5		0.2 V_{CC}	V
V_{IH}	Input High Voltage		0.7V_{CC}		V_{CC}+0.4	V
V_{OL}	Output Low Voltage	I_{OL} = 100 µA, Vcc=Vcc Min.			0.3	V
V_{OH}	Output High Voltage	I_{OH} = –100 µA , Vcc=Vcc Min.	V_{CC}-0.2			V

Note : Typical condition at VCC 3.3V, T = 25℃.

Table 18. AC Measurement Conditions

Symbol	Parameter	Min.	Max.	Unit
C_L	Load Capacitance	30		pF
	Input Rise and Fall Times		5	ns
	Input Pulse Voltages	0.2V_{CC} to 0.8V_{CC}		V
	Input Timing Reference Voltages	0.3V_{CC} to 0.7V_{CC}		V
	Output Timing Reference Voltages	V_{CC} / 2		V

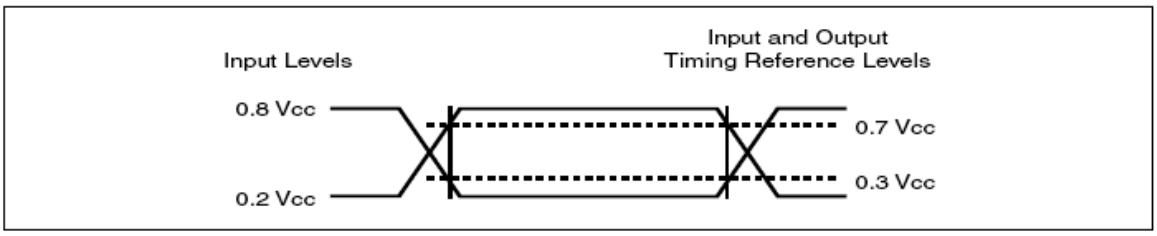


Figure 43. AC Measurement I/O Waveform

Table 19. AC Characteristics

(T_a = - 40°C to 85°C; V_{CC} = 2.7-3.6V)

Symbol	Alt	Parameter		Min	Typ	Max	Unit
F_R	f_C	Serial SDR Clock Frequency for: FAST_READ, PP, QPP, SE, BE, DP, RES, WREN, WRDI, Dual Output Fast Read and Quad I/O Fast Read, Read Burst,		D.C.		104	MHz
		Serial SDR Clock Frequency for: RDID, WRSR, WRSR2, WRSR3, RDSR, RDSR2, RDSR3		D.C.		104	MHz
f_R		Serial SDR Clock Frequency for READ,		D.C.		83	MHz
t_{CH} [1]		Serial Clock High Time for SDR		3.5			ns
t_{CL} [1]		Serial Clock Low Time for SDR		3.5			ns
t_{CLCH} [2]		Serial Clock Rise Time (Slew Rate)		0.1			V / ns
t_{CHCL} [2]		Serial Clock Fall Time (Slew Rate)		0.1			V / ns
t_{SLCH}	t_{CSS}	CS# Active Setup Time		5			ns
t_{CHSH}		CS# Active Hold Time		5			ns
t_{SHCH}		CS# Not Active Setup Time		5			ns
t_{CHSL}		CS# Not Active Hold Time		5			ns
t_{SHSL}	t_{CSH}	CS# High Time for read CS# High Time for program/erase		10 30			ns
t_{SHQZ} [2]	t_{DIS}	Output Disable Time				6	ns
t_{CLQX}	t_{HO}	Output Hold Time		0			ns
t_{DVCH}	t_{DSU}	Data In Setup Time		2			ns
t_{CHDX}	t_{DH}	Data In Hold Time		3			ns
t_{HLCH}		HOLD# Low Setup Time (relative to CLK)		5			ns
t_{HHCH}		HOLD# High Setup Time (relative to CLK)		5			ns
t_{CHHH}		HOLD# Low Hold Time (relative to CLK)		5			ns
t_{CHHL}		HOLD# High Hold Time (relative to CLK)		5			ns
t_{CLQV}	t_V	Output Valid from CLK				7	ns
t_{WHSL} [3]		Write Protect Setup Time before CS# Low		20			ns
t_{SHWL} [3]		Write Protect Hold Time after CS# High		100			ns
t_{DP} [2]		CS# High to Deep Power-down Mode				3	µs
t_{RES1} [2]		CS# High to Standby Mode without Electronic Signature read				3	µs
t_{RES2} [2]		CS# High to Standby Mode with Electronic Signature read				1.8	µs
t_W		Write Status Register Cycle Time			10	50	ms
t_{PP}		Page Programming Time			0.7	3	ms
t_{SE}		Sector Erase Time			0.05	0.5	s
t_{HBE}		32KB Block Erase Time			0.2	1	s
t_{BE}		64KB Block Erase Time			0.35	2	s
t_{CE}		Chip Erase Time			30	100	s
	t_{SR}	Software Reset Latency	WIP = write operation			28	µs
			WIP = not in write operation			0	µs

Note: 1. $t_{CH} + t_{CL}$ must be greater than or equal to 1/ f_C

2. Value guaranteed by characterization, not 100% tested in production.
3. Only applicable as a constraint for a Write status Register instruction when Status Register Protect Bit is set at 1.
4. Write Status Register Cycle Time is for Status Register 1 and Status Register 3

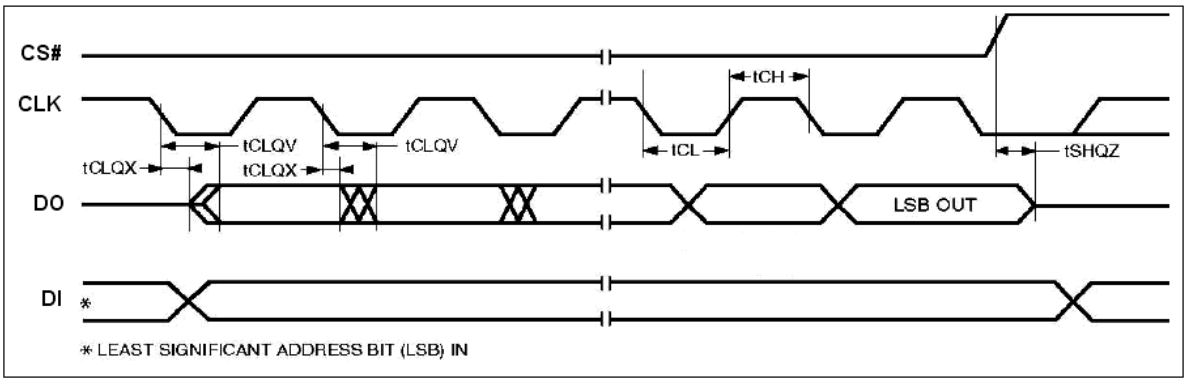


Figure 44. Serial Output Timing

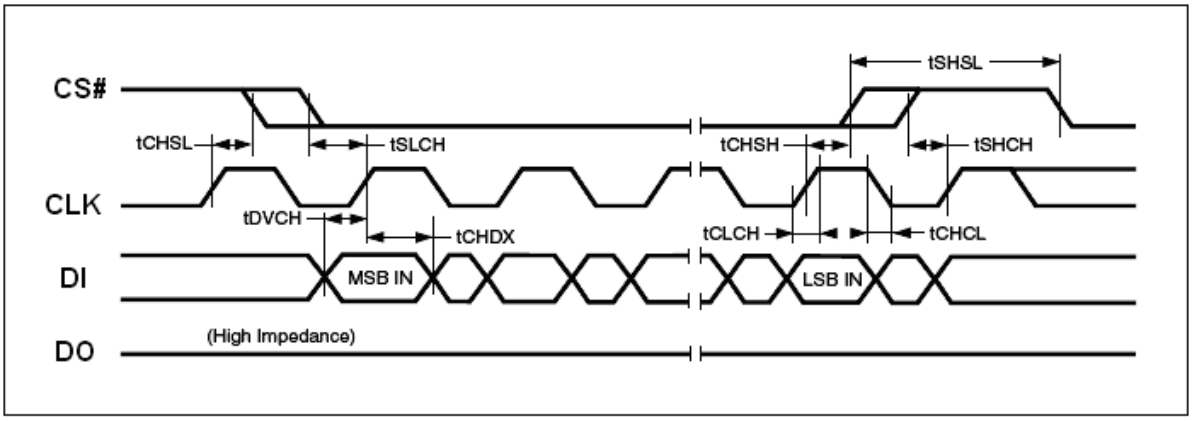


Figure 45. Input Timing

Absolute Maximum Ratings

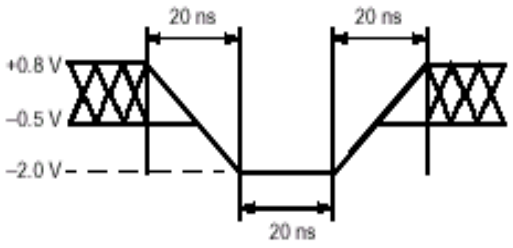
Parameter	Value	Unit
Ambient Operating Temperature Industrial Devices	-40 to 85	C
Operating Supply Voltage Vcc	Full: 2.7 to 3.6	V

Notes:
1. Recommended Operating Ranges define those limits between which the functionality of the device is guaranteed.

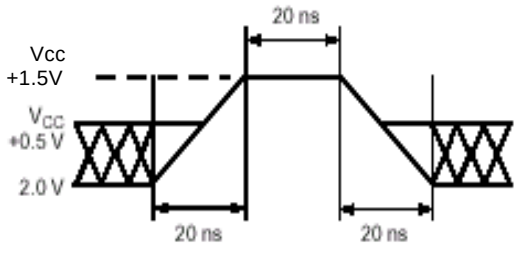
Stresses above the values so mentioned above may cause permanent damage to the device. These values are for a stress rating only and do not imply that the device should be operated at conditions up to or above these values. Exposure of the device to the maximum rating values for extended periods of time may adversely affect the device reliability.

Parameter	Value	Unit
Storage Temperature	-65 to +150	C
Plastic Packages	-65 to +125	C
Output Short Circuit Current[1]	200	mA
Input and Output Voltage (with respect to ground) [2]	-0.5 to Vcc+0.5	V
Vcc	-0.5 to Vcc+0.5	V

Notes:
1. No more than one output shorted at a time. Duration of the short circuit should not be greater than one second.
2. Minimum DC voltage on input or I/O pins is –0.5 V. During voltage transitions, inputs may undershoot V_{ss} to –1.0V for periods of up to 50ns and to –2.0 V for periods of up to 20ns. See figure below. Maximum DC voltage on output and I/O pins is V_{cc} + 0.5 V. During voltage transitions, outputs may overshoot to V_{cc} + 2.0 V for periods up to 20ns. See figure below.



Maximum Negative Overshoot Waveform



Maximum Positive Overshoot Waveform

Table 20. DATA RETENTION and ENDURANCE

Parameter Description	Test Conditions	Min	Unit
Data Retention Time	85°C	20	Years
Erase/Program Endurance	-40 to 85 °C	100k	cycles

Table 21. CAPACITANCE

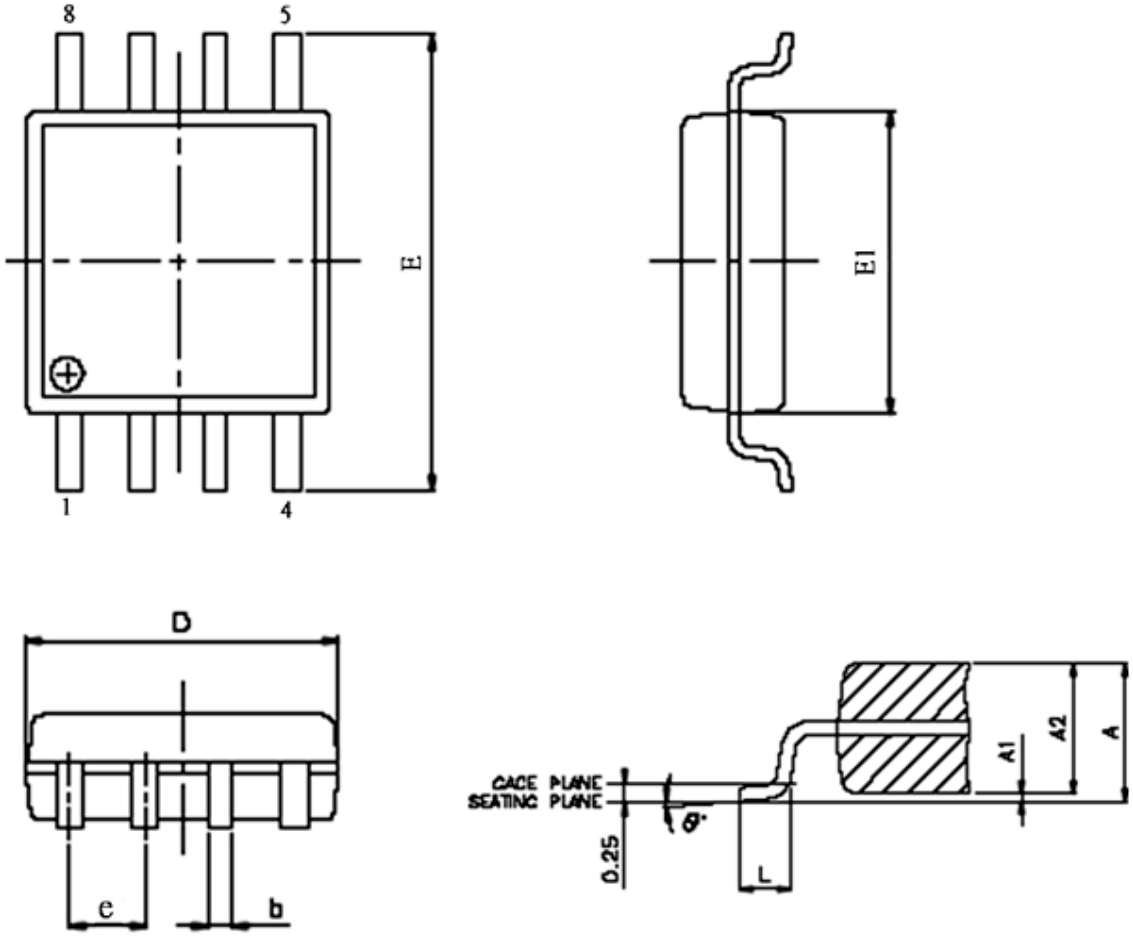
(V_{CC} = 2.7-3.6V)

Parameter Symbol	Parameter Description	Test Setup	Typ	Max	Unit
C_{IN}	Input Capacitance	V_{IN} = 0		6	pF
C_{OUT}	Output Capacitance	V_{OUT} = 0		8	pF

Note : Sampled only, not 100% tested, at T_A = 25°C and a frequency of 20MHz.

PACKAGE MECHANICAL

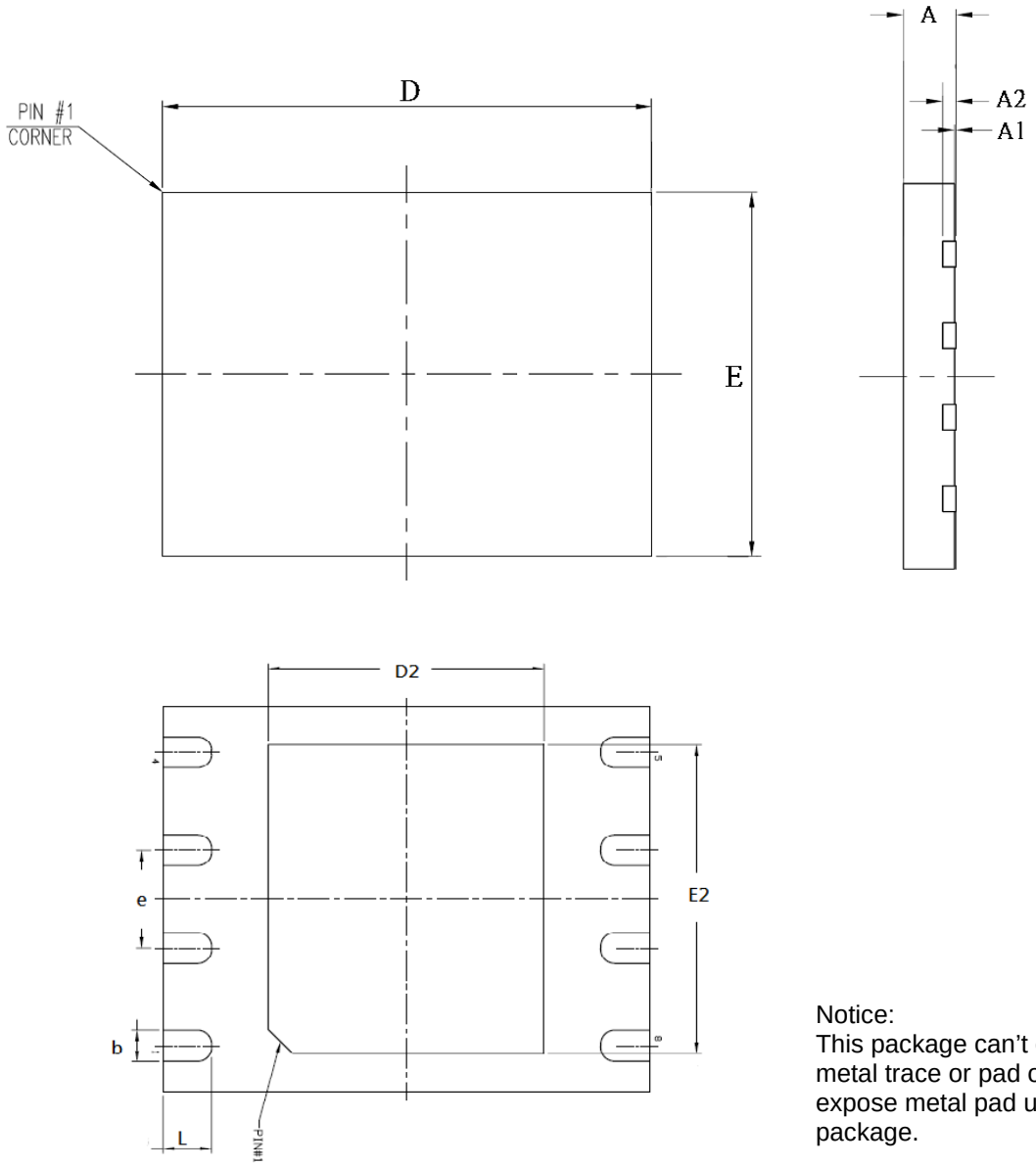
Figure 46. SOP 200 mil (official name = 208 mil)



SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	1.75	1.975	2.20
A1	0.05	0.15	0.25
A2	1.70	1.825	1.95
D	5.15	5.275	5.40
E	7.70	7.90	8.10
E1	5.15	5.275	5.40
e	- - -	1.27	- - -
b	0.35	0.425	0.50
L	0.5	0.65	0.80
θ	0^0	4^0	8^0

Note : 1. Coplanarity: 0.1 mm
2. Max. allowable mold flash is 0.15 mm at the pkg ends, 0.25 mm between leads.

Figure 47. VDFN 8 (5x6 mm)

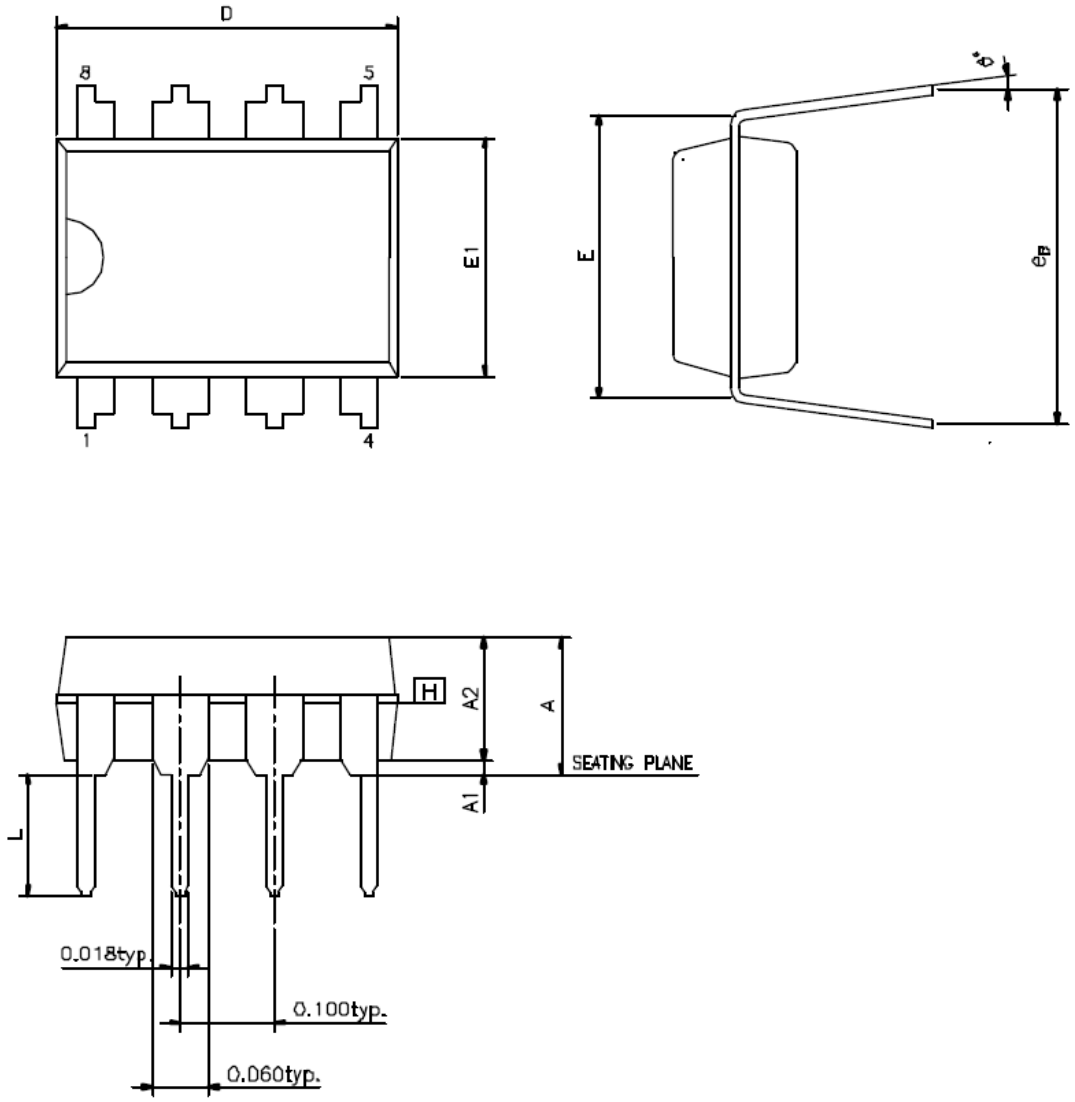


Notice:
This package can't contact to metal trace or pad on board due to expose metal pad underneath the package.

SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	0.70	0.75	0.80
A1	0.00	0.02	0.04
A2	- - -	0.20	- - -
D	5.90	6.00	6.10
E	4.90	5.00	5.10
D2	3.30	3.40	3.50
E2	3.90	4.00	4.10
e	- - -	1.27	- - -
b	0.35	0.40	0.45
L	0.55	0.60	0.65

Note : 1. Coplanarity: 0.1 mm

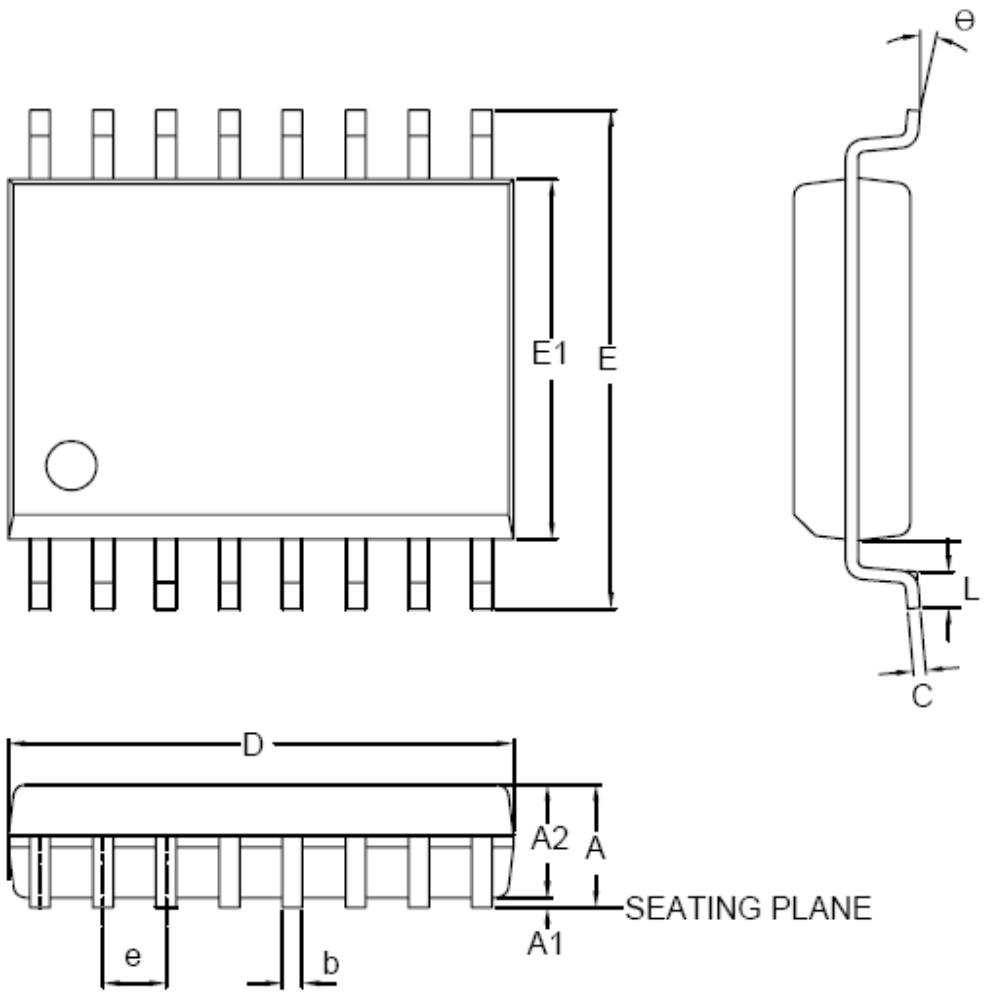
Figure 48. PDIP8



SYMBOL	DIMENSION IN INCH		
	MIN.	NOR	MAX
A	- - -	- - -	0.210
A1	0.015	- - -	- - -
A2	0.125	0.130	0.135
D	0.355	0.365	0.400
E	0.300	0.310	0.320
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
e_B	0.310	0.350	0.375
Θ^0	0	7	15

Note : 1. Coplanarity: 0.1 mm

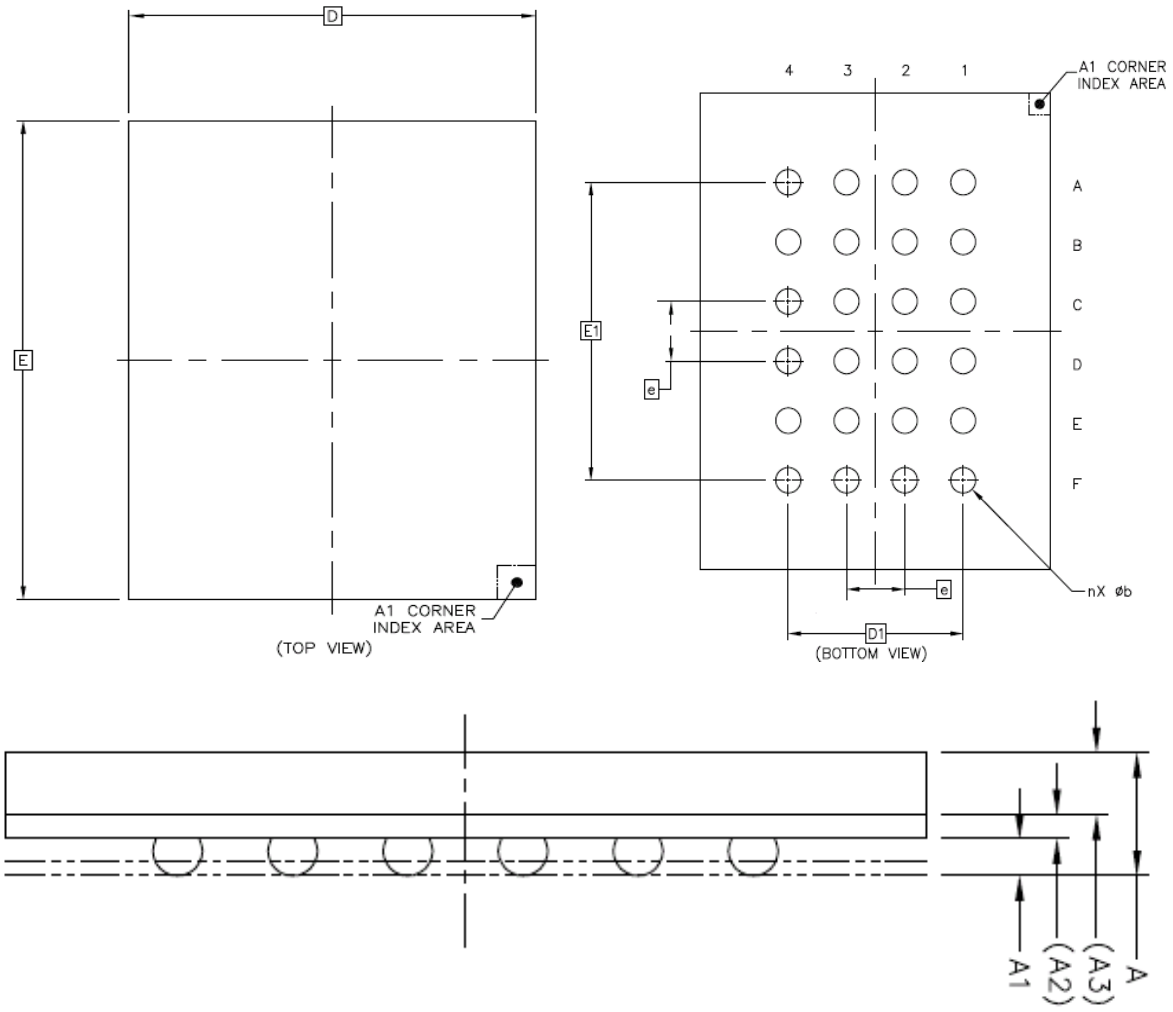
Figure 49. 16 LEAD SOP 300 mil



SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	- - -	- - -	2.65
A1	0.10	0.20	0.30
A2	2.25	- - -	2.40
C	0.20	0.25	0.30
D	10.10	10.30	10.50
E	10.00	- - -	10.65
E1	7.40	7.50	7.60
e	- - -	1.27	- - -
b	0.31	- - -	0.51
L	0.4	- - -	1.27
θ	0^0	5^0	8^0

Note : 1. Coplanarity: 0.1 mm

Figure 50. 24-ball Thin Profile Fine-Pitch Ball Grid Array (6 x 8 mm) Package



SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	- - -	- - -	1.20
A1	0.27	- - -	0.37
A2	0.21 REF		
A3	0.54 REF		
D	6 BSC		
E	8 BSC		
D1	- - -	3.00	- - -
E1	- - -	5.00	- - -
e	- - -	1.00	- - -
b	- - -	0.40	- - -

Note : 1. Coplanarity: 0.1 mm

ORDERING INFORMATION

EN25QH64A - 104 H I P

PACKAGING CONTENT
P = RoHS, Halogen-Free and REACH compliant

TEMPERATURE RANGE
I = Industrial (-40°C to +85°C)

PACKAGE
H = 8-pin 200mil SOP
W = 8-pin VDFN (5x6mm)
Q = 8-pin PDIP
F = 16-pin 300mil SOP
BB = 24-ball TFBGA (6 x 8 x 1.2mm)

SPEED
133 = 133 MHz
104 = 104 MHz

BASE PART NUMBER
EN = Eon Silicon Solution Inc.
25QH = 3V Serial Flash with 4KB Uniform-Sector
64 = 64 Megabit (8192K x 8)
A = version identifier

Revisions List

Revision No	Description	Date
A	Initial Release	2014/10/01