

# FT25H16

# DATASHEET

FMD Confidential

# CONTENTS

<b>1. FEATURES</b> .....	<b>4</b>
<b>2. GENERAL DESCRIPTION</b> .....	<b>6</b>
<b>3. MEMORY ORGANIZATION</b> .....	<b>8</b>
<b>4. DEVICE OPERATION</b> .....	<b>9</b>
<b>5. DATA PROTECTION</b> .....	<b>10</b>
<b>6. STATUS REGISTER</b> .....	<b>12</b>
<b>7. COMMANDS DESCRIPTION</b> .....	<b>14</b>
7.1. WRITE ENABLE (WREN) (06H).....	18
7.2. WRITE DISABLE (WRDI) (04H) .....	18
7.3. READ STATUS REGISTER (RDSR) (05H OR 35H).....	18
7.4. WRITE STATUS REGISTER (WRSR) (01H).....	19
7.5. READ DATA BYTES (READ) (03H).....	19
7.6. READ DATA BYTES AT HIGHER SPEED (FAST READ) (0BH) .....	20
7.7. DUAL OUTPUT FAST READ (3BH) .....	20
7.8. QUAD OUTPUT FAST READ (6BH) .....	21
7.9. DUAL I/O FAST READ (BBH) .....	22
7.10. QUAD I/O FAST READ (EBH).....	23
7.11. QUAD I/O WORD FAST READ (E7H).....	24
7.12. PAGE PROGRAM (PP) (02H) .....	25
7.13. QUAD PAGE PROGRAM (QPP) (32H).....	26
7.14. SECTOR ERASE (SE) (20H).....	27
7.15. 32KB BLOCK ERASE (BE) (52H).....	28
7.16. 64KB BLOCK ERASE (BE) (D8H).....	28
7.17. CHIP ERASE (CE) (60/C7H) .....	29
7.18. DEEP POWER-DOWN (DP) (B9H) .....	29
7.19. RELEASE FROM DEEP POWER-DOWN AND READ DEVICE ID (RDI) (ABH).....	30
7.20. READ MANUFACTURE ID/ DEVICE ID (REMS) (90H).....	31
7.21. READ IDENTIFICATION (RDID) (9FH).....	32
7.22. HIGH SPEED MODE (HSM) (A3H) .....	32
7.23. CONTINUOUS READ MODE RESET (CRMR) (FFH).....	33
7.24. PROGRAM/ERASE SUSPEND (PES) (75H) .....	33
7.25. PROGRAM/ERASE RESUME (PER) (7AH) .....	34
7.26. ERASE SECURITY REGISTERS (44H) .....	34
7.27. PROGRAM SECURITY REGISTERS (42H) .....	35
7.28. READ SECURITY REGISTERS (48H) .....	36
<b>8. ELECTRICAL CHARACTERISTICS</b> .....	<b>37</b>
8.1. POWER-ON TIMING .....	37
8.2. INITIAL DELIVERY STATE .....	37
8.3. DATA RETENTION AND ENDURANCE .....	37

---

8.4. LATCH UP CHARACTERISTICS .....	37
8.5. ABSOLUTE MAXIMUM RATINGS .....	38
8.6. CAPACITANCE MEASUREMENT CONDITION .....	38
8.7. DC CHARACTERISTICS.....	39
8.8. AC CHARACTERISTICS .....	40
<b>9. ORDERING INFORMATION .....</b>	<b>42</b>
<b>10. PACKAGE INFORMATION .....</b>	<b>43</b>
10.1. PACKAGE SOP8 150MIL.....	43
10.2. PACKAGE SOP8 208MIL.....	44
10.3. PACKAGE DIP8 300MIL .....	45
10.4. PACKAGE VSOP8 208MIL.....	46
10.5. PACKAGE TSSOP8 173MIL .....	47
<b>11. REVISION HISTORY .....</b>	<b>48</b>

## 1. FEATURES

- 16M -bit Serial Flash
  - 2048K-byte
  - 256 bytes per programmable page
- Standard, Dual, Quad SPI
  - Standard SPI: SCLK, CS#, SI, SO, WP#, HOLD#
  - Dual SPI: SCLK, CS#, IO0, IO1, WP#, HOLD#
  - Quad SPI: SCLK, CS#, IO0, IO1, IO2, IO3
- High Speed Clock Frequency
  - 120MHz for fast read with 30PF load
  - Dual I/O Data transfer up to 240Mbps/s
  - Quad I/O Data transfer up to 480Mbps/s
- Program/Erase Speed
  - Page Program time: 0.4ms typical
  - Sector Erase time: 120ms typical
  - Block Erase time: 0.2/0.4s typical
  - Chip Erase time: 10s typical
- Flexible Architecture
  - Sector of 4K-byte
  - Block of 32/64k-byte
- Low Power Consumption
  - 20mA maximum active current
  - 5uA maximum power down current
- Software/Hardware Write Protection
  - Write protect all/portion of memory via software
  - Enable/Disable protection with WP# Pin
  - Top or Bottom, Sector or Block selection
- Advanced security Features
  - 4\*256-Byte Security Registers With OTP Lock
- Single Power Supply Voltage: Full voltage range:2.7~3.6V
- Minimum 100,000 Program/Erase Cycle

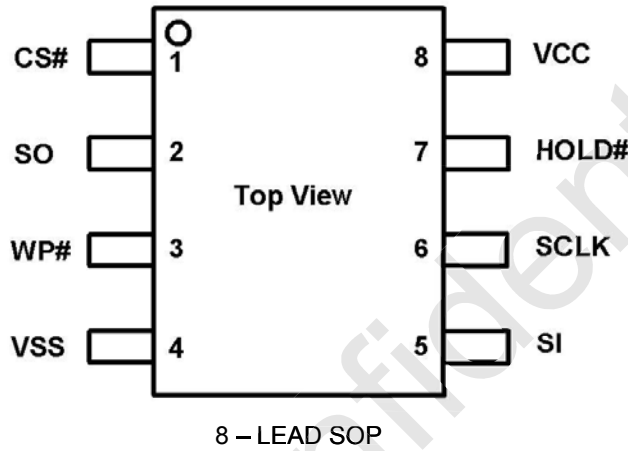
- Hardware Features
  - 8-pin SOP8 (150mil)
  - 8-pin SOP8 (208mil)
  - 8-pin DIP8 (300mil)
  - 8-pin VSOP8 (200mil)
  - 8-pin TSSOP8 (173mil)

FMD Confidential

**2. GENERAL DESCRIPTION**

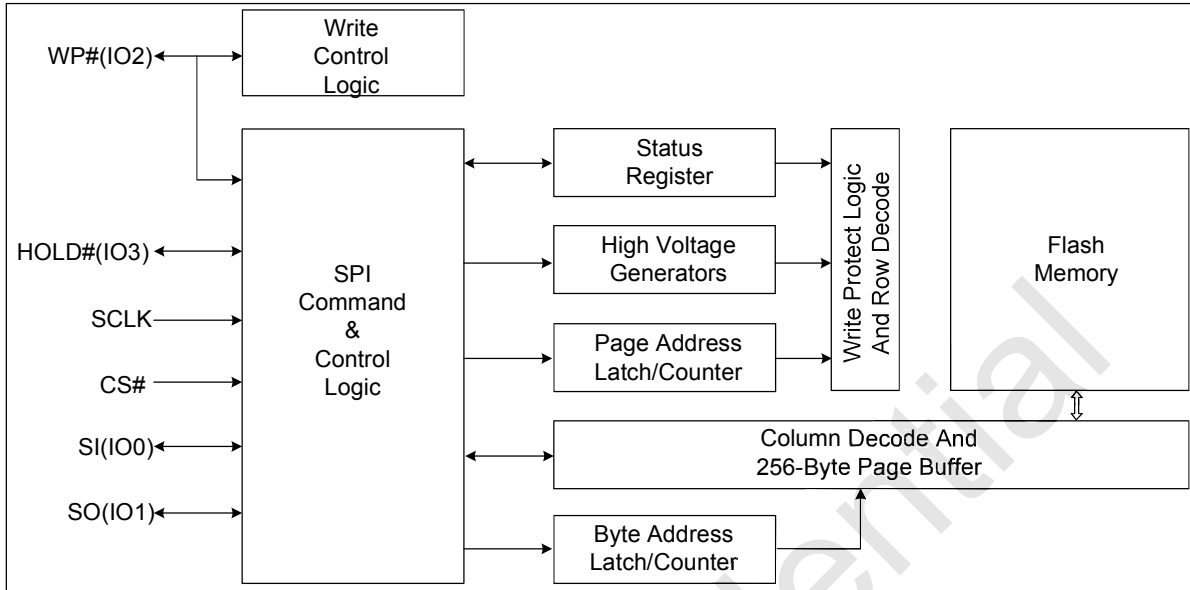
The FT25H16 (16M-bit) Serial flash supports the standard Serial Peripheral Interface (SPI), and supports the Dual/Quad SPI: Serial Clock, Chip Select, Serial Data I/O0 (SI), I/O1 (SO), I/O2 (WP#), and I/O3 (HOLD#). The Dual I/O data is transferred with speed of 240Mbits/s and the Quad I/O & Quad output data is transferred with speed of 480Mbits/s.

**CONNECTION DIAGRAM**



**PIN DESCRIPTION**

Pin Name	I/O	Description
CS#	I	Chip Select Input
SO (IO1)	I/O	Data Output (Data Input Output 1)
WP# (IO2)	I/O	Write Protect Input (Data Input Output 2)
VSS		Ground
SI (IO0)	I/O	Data Input (Data Input Output 0)
SCLK	I	Serial Clock Input
HOLD# (IO3)	I/O	Hold Input (Data Input Output 3)
VCC		Power Supply

**BLOCK DIAGRAM**

### 3. MEMORY ORGANIZATION

#### FT25H16

Each Device has	Each block has	Each sector has	Each page has	
2M	64/32K	4K	256	bytes
8K	256/128	16	-	pages
512	16/8	-	-	sectors
32/64	-	-	-	blocks

#### UNIFORM BLOCK SECTOR ARCHITECTURE FT25H16 64K Bytes Block Sector Architecture

Block	Sector	Address range	
31	511	1FF000H	1FFFFFFH
	.....	.....	.....
	496	1F0000H	1F0FFFFH
30	495	1EF000H	1EFFFFFFH
	.....	.....	.....
	480	1E0000H	1E0FFFFH
.....	.....	.....	.....
	.....	.....	.....
	.....	.....	.....
.....	.....	.....	.....
	.....	.....	.....
	.....	.....	.....
2	47	02F000H	02FFFFFFH
	.....	.....	.....
	32	020000H	020FFFFH
1	31	01F000H	01FFFFFFH
	.....	.....	.....
	16	010000H	010FFFFH
0	15	00F000H	00FFFFFFH
	.....	.....	.....
	0	000000H	000FFFFH



## 4. DEVICE OPERATION

### SPI Mode

#### Standard SPI

The FT25H16 features a serial peripheral interface on 4 signals bus: Serial Clock (SCLK), Chip Select (CS#), Serial Data Input (SI) and Serial Data Output (SO). Both SPI bus mode 0 and 3 are supported. Input data is latched on the rising edge of SCLK and data shifts out on the falling edge of SCLK.

#### Dual SPI

The FT25H16 supports Dual SPI operation when using the “Dual Output Fast Read” and “Dual I/O Fast Read” (3BH and BBH) commands. These commands allow data to be transferred to or from the device at two times the rate of the standard SPI. When using the Dual SPI command the SI and SO pins become bidirectional I/O pins: IO0 and IO1.

#### Quad SPI

The FT25H16 supports Quad SPI operation when using the “Quad Output Fast Read,” “Quad I/O Fast Read”, “Quad I/O Word Fast Read” (6BH, EBH, E7H) commands. These commands allow data to be transferred to or from the device at four times the rate of the standard SPI. When using the Quad SPI command the SI and SO pins become bidirectional I/O pins: IO0 and IO1, and WP# and HOLD# pins become IO2 and IO3. Quad SPI commands require the non-volatile Quad Enable bit (QE) in Status Register to be set.

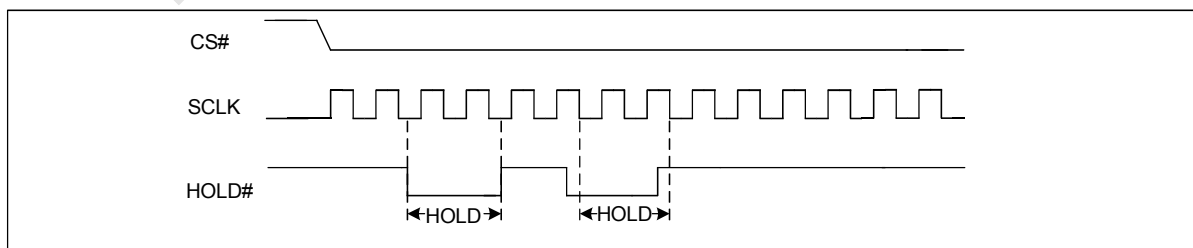
#### Hold

The HOLD# signal goes low to stop any serial communications with the device, but doesn't stop the operation of write status register, programming, or erasing in progress.

The operation of HOLD, need CS# keep low, and starts on falling edge of the HOLD# signal, with SCLK signal being low (if SCLK is not being low, HOLD operation will not start until SCLK being low). The HOLD condition ends on rising edge of HOLD# signal with SCLK being low (If SCLK is not being low, HOLD operation will not end until SCLK being low).

The SO is high impedance, both SI and SCLK don't care during the HOLD operation, if CS# drives high during HOLD operation, it will reset the internal logic of the device. To re-start communication with chip, the HOLD# must be at high and then CS# must be at low.

**Figure1. Hold Condition**



## 5. DATA PROTECTION

The FT25H16 provide the following data protection methods:

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

**Table1.0 FT25H16 Protected area size (CMP=1)**

Status Register Content					Memory Content			
BP4	BP3	BP2	BP1	BP0	Blocks	Addresses	Density	Portion
X	X	0	0	0	0 to 31	000000H-1FFFFFFH	2M	ALL
0	0	0	0	1	0 to 30	000000H-1EFFFFFFH	1984KB	Lower 31/32
0	0	0	1	0	0 to 29	000000H-1DFFFFFFH	1920KB	Lower 15/16
0	0	0	1	1	0 to 27	000000H-1BFFFFFFH	1792KB	Lower 7/8
0	0	1	0	0	0 to 23	000000H-17FFFFFFH	1536KB	Lower 3/4
0	0	1	0	1	0 to 15	000000H-0FFFFFFH	1M	Lower 1/2
0	1	0	0	1	1 to 31	010000H-1FFFFFFH	1984KB	Upper 31/32
0	1	0	1	0	2 to 31	000000H-01FFFFFFH	1920KB	Upper 15/16
0	1	0	1	1	4 to 31	000000H-03FFFFFFH	1792KB	Upper 7/8
0	1	1	0	0	8 to 31	000000H-07FFFFFFH	1536KB	Upper 3/4
0	1	1	0	1	16 to 31	000000H-0FFFFFFH	1M	Upper 1/2
X	X	1	1	X	NONE	000000H-1FFFFFFH	NONE	NONE
1	0	0	0	1	31	1FF000H-1FFFFFFH	4KB	Top Block
1	0	0	1	0	31	1FE000H-1FFFFFFH	8KB	Top Block
1	0	0	1	1	31	1FC000H-1FFFFFFH	16KB	Top Block
1	0	1	0	X	31	1F8000H-1FFFFFFH	32KB	Top Block
1	1	0	0	1	0	000000H-000FFFFH	4KB	Bottom Block
1	1	0	1	0	0	000000H-001FFFFH	8KB	Bottom Block
1	1	0	1	1	0	000000H-003FFFFH	16KB	Bottom Block
1	1	1	0	X	0	000000H-007FFFFH	32KB	Bottom Block

Table1.1 FT25H16 Protected area size (CMP=1)

Status Register Content					Memory Content			
BP4	BP3	BP2	BP1	BP0	Blocks	Addresses	Density	Portion
X	X	0	0	0	0 to 31	000000H-1FFFFFFH	2M	ALL
0	0	0	0	1	0 to 30	000000H-1EFFFFFFH	1984KB	Lower 31/32
0	0	0	1	0	0 to 29	000000H-1DFFFFFFH	1920KB	Lower 15/16
0	0	0	1	1	0 to 27	000000H-1BFFFFFFH	1792KB	Lower 7/8
0	0	1	0	0	0 to 23	000000H-17FFFFFFH	1536KB	Lower 3/4
0	0	1	0	1	0 to 15	000000H-0FFFFFFH	1M	Lower 1/2
0	1	0	0	1	1 to 31	010000H-1FFFFFFH	1984KB	Upper 31/32
0	1	0	1	0	2 to 31	020000H-1FFFFFFH	1920KB	Upper 15/16
0	1	0	1	1	4 to 31	040000H-1FFFFFFH	1792KB	Upper 7/8
0	1	1	0	0	8 to 31	080000H-1FFFFFFH	1536KB	Upper 3/4
0	1	1	0	1	16 to 31	100000H-1FFFFFFH	1M	Upper 1/2
X	X	1	1	X	NONE	NONE	NONE	NONE
1	0	0	0	1	0 to 31	000000H-1FEFFFFH	2044KB	Lower 511/512
1	0	0	1	0	0 to 31	000000H-1FDFFFFH	2040KB	Lower 255/256
1	0	0	1	1	0 to 31	000000H-1FBFFFFH	2032KB	Lower 127/128
1	0	1	0	X	0 to 31	000000H-1F7FFFFH	2016KB	Lower 63/64
1	1	0	0	1	0 to 31	001000H-1FFFFFFH	2044KB	Upper 511/512
1	1	0	1	0	0 to 31	002000H-1FFFFFFH	2040KB	Upper 255/256
1	1	0	1	1	0 to 31	004000H-1FFFFFFH	2032KB	Upper 127/128
1	1	1	0	X	0 to 31	008000H-1FFFFFFH	2016KB	Upper 63/64

## 6. STATUS REGISTER

<b>S15</b>	<b>S14</b>	<b>S13</b>	<b>S12</b>	<b>S11</b>	<b>S10</b>	<b>S9</b>	<b>S8</b>
SUS	CMP	Reserved	Reserved	Reserved	LB	QE	Reserved

<b>S7</b>	<b>S6</b>	<b>S5</b>	<b>S4</b>	<b>S3</b>	<b>S2</b>	<b>S1</b>	<b>S0</b>
SRP	BP4	BP3	BP2	BP1	BP0	WEL	WIP

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

### WIP bit.

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

### WEL bit.

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

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

The Block Protect (BP4, BP3, BP2, BP1, BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase commands. These bits are written with the Write Status Register (WRSR) command. When the Block Protect (BP4, BP3, BP2, BP1, BP0) bits are set to 1, the relevant memory area (as defined in Table1).becomes protected against Page Program (PP), Sector Erase (SE) and Block Erase (BE) commands. The Block Protect (BP4, BP3, BP2, BP1, BP0) bits can be written provided that the Hardware Protected mode has not been set. The Chip Erase (CE) command is executed, if the Block Protect (BP2, BP1, BP0) bits and CMP are all 0 or all 1.

### SRP bit.

The Status Register Protect (SRP) bit is non-volatile Read/Write bits in the status register. The SRP bit controls the method of write protection: software protection, hardware protection, power supply lock-down or one time programmable protection.

SRP	WP#	Status Register	Description
0	X	Software Protected	The Status Register can be written to after a Write Enable command, WEL=1.(Default)
1	0	Hardware Protected	WP#=0,the Status Register locked and can not be written to.
1	1	Hardware Unprotected	WP#=1,the Status Register is unlocked and can be written to after a Write Enable command, WEL=1.

**QE bit.**

The Quad Enable (QE) bit is a non-volatile Read/Write bit in the Status Register that allows Quad operation. When the QE bit is set to 0 (Default) the WP# pin and HOLD# pin are enable. When the QE pin is set to 1, the Quad IO2 and IO3 pins are enabled. (The QE bit should never be set to 1 during standard SPI or Dual SPI operation if the WP# or HOLD# pins are tied directly to the power supply or ground)

**LB bit.**

The LB bit is a non-volatile One Time Program (OTP) bit in Status Register (S10) that provide the write protect control

and status to the Security Registers. The default state of LB is 0, the security registers are unlocked. LB can be set to 1 individually using the Write Register instruction. LB is One Time Programmable, once it's set to 1, the Security Registers will become read-only permanently.

**CMP bit.**

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

**SUS bit.**

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

## 7. COMMANDS DESCRIPTION

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

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

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

**Table2. Commands**

Command Name	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	n-Bytes
Write Enable	06H						
Write Disable	04H						
Read Status Register	05H	(S7-S0)					(continuous)
Read Status Register-1	35H	(S15-S8)					(continuous)
Write Status Register	01H	(S7-S0)	(S15-S8)				(continuous)
Read Data	03H	A23-A16	A15-A8	A7-A0	(D7-D0)	(Next byte)	(continuous)
Fast Read	0BH	A23-A16	A15-A8	A7-A0	dummy	(D7-D0)	(continuous)
Dual Output Fast Read	3BH	A23-A16	A15-A8	A7-A0	dummy	(D7-D0) <sup>(1)</sup>	(continuous)
Dual I/O Fast Read	BBH	A23-A8 <sup>(2)</sup>	A7-A0 M7-M0 <sup>(2)</sup>	(D7-D0) <sup>(1)</sup>			(continuous)
Quad Output Fast Read	6BH	A23-A16	A15-A8	A7-A0	dummy	(D7-D0) <sup>(3)</sup>	(continuous)
Quad I/O Fast Read	EBH	A23-A0 M7-M0 <sup>(4)</sup>	Dummy <sup>(5)</sup>	(D7-D0) <sup>(3)</sup>			(continuous)
Quad I/O Word Fast Read	E7H	A23-A0 M7-M0 <sup>(4)</sup>	Dummy <sup>(6)</sup>	(D7-D0) <sup>(3)</sup>			(continuous)
Continuous Read Reset	FFH						
Page Program	02H	A23-A16	A15-A8	A7-A0	(D7-D0)	(Next byte)	
Quad Page Program	32H	A23-A16	A15-A8	A7-A0	(D7-D0) <sup>(3)</sup>		
Sector Erase	20H	A23-A16	A15-A8	A7-A0			
Block Erase(32KB)	52H	A23-A16	A15-A8	A7-A0			
Block Erase(64KB)	D8H	A23-A16	A15-A8	A7-A0			
Chip Erase	C7/60H						

Program/Erase Suspend	75H						
Program/Erase Resume	7AH						
Deep Power-Down	B9H						
Release From Deep Power-Down, And Read Device ID	ABH	dummy	dummy	dummy	(DID7-DID0)		(continuous)
Release From Deep Power-Down	ABH						
Manufacturer/Device ID	90H	dummy	dummy	00H	(MID7-MID0)	(DID7-DID0)	(continuous)
High Speed Mode	A3H	dummy	dummy	dummy			
Read Identification	9FH	(MID7-MID0)	(JDID15-JDID8)	(JDID7-JDID0)			(continuous)
Erase Security Register <sup>(8)</sup>	44H	A23-A16	A15-A8	A7-A0			
Program Security Register <sup>(8)</sup>	42H	A23-A16	A15-A8	A7-A0	(D7-D0)	(Next byte)	
Read Security Register <sup>(8)</sup>	48H	A23-A16	A15-A8	A7-A0	dummy	(D7-D0)	

**NOTE:**
**1. Dual Output data**
 $IO0 = (D6, D4, D2, D0)$ 
 $IO1 = (D7, D5, D3, D1)$ 
**2. Dual Input Address**
 $IO0 = A22, A20, A18, A16, A14, A12, A10, A8, A6, A4, A2, A0, M6, M4, M2, M0$ 
 $IO1 = A23, A21, A19, A17, A15, A13, A11, A9, A7, A5, A3, A1, M7, M5, M3, M1$ 
**3. Quad Output Data**
 $IO0 = (D4, D0, \dots)$ 
 $IO1 = (D5, D1, \dots)$ 
 $IO2 = (D6, D2, \dots)$ 
 $IO3 = (D7, D3, \dots)$ 
**4. Quad Input Address**
 $IO0 = A20, A16, A12, A8, A4, A0, M4, M0$ 
 $IO1 = A21, A17, A13, A9, A5, A1, M5, M1$ 
 $IO2 = A22, A18, A14, A10, A6, A2, M6, M2$ 
 $IO3 = A23, A19, A15, A11, A7, A3, M7, M3$ 
**5. Quad I/O Fast Read Data**
 $IO0 = (x, x, x, x, D4, D0, \dots)$ 
 $IO1 = (x, x, x, x, D5, D1, \dots)$ 
 $IO2 = (x, x, x, x, D6, D2, \dots)$ 
 $IO3 = (x, x, x, x, D7, D3, \dots)$

## 6. Quad I/O Word Fast Read Data

IO0 = (x, x, D4, D0,...)

IO1 = (x, x, D5, D1,...)

IO2 = (x, x, D6, D2,...)

IO3 = (x, x, D7, D3,...)

## 7. Quad I/O Word Fast Read Data: the lowest address bit must be 0.

## 8. Security Registers Address:

Security Register0: A23-A16=00H, A15-A8=00H, A7-A0= Byte Address;

Security Register1: A23-A16=00H, A15-A8=01H, A7-A0= Byte Address;

Security Register2: A23-A16=00H, A15-A8=02H, A7-A0= Byte Address;

Security Register3: A23-A16=00H, A15-A8=03H, A7-A0= Byte Address.



**Table of ID Definitions:****FT25H16**

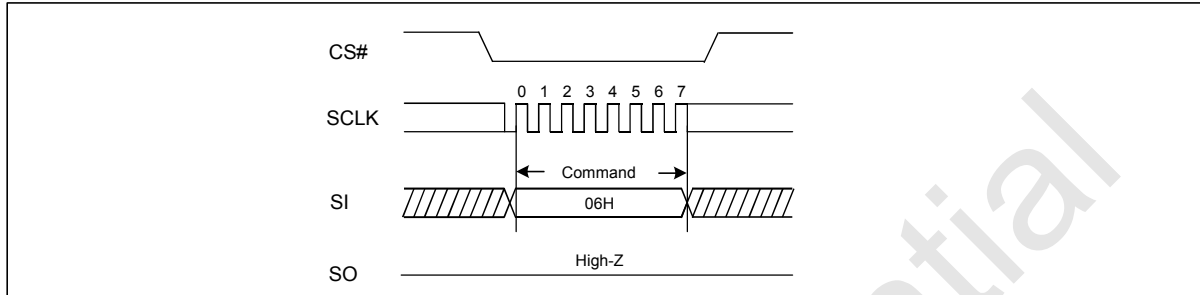
Operation Code	M7-M0	ID15-ID8	ID7-ID0
9FH	0E	40	15
90H	0E		14
ABH			14

FMD Confidential

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

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

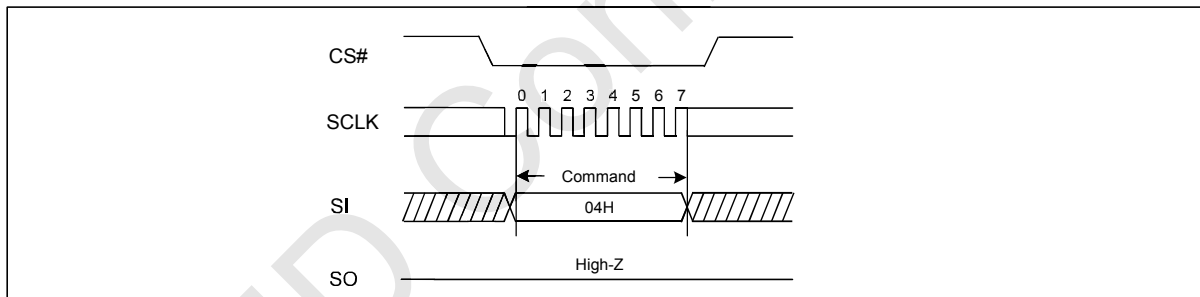
**Figure2. Write Enable Sequence Diagram**



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

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

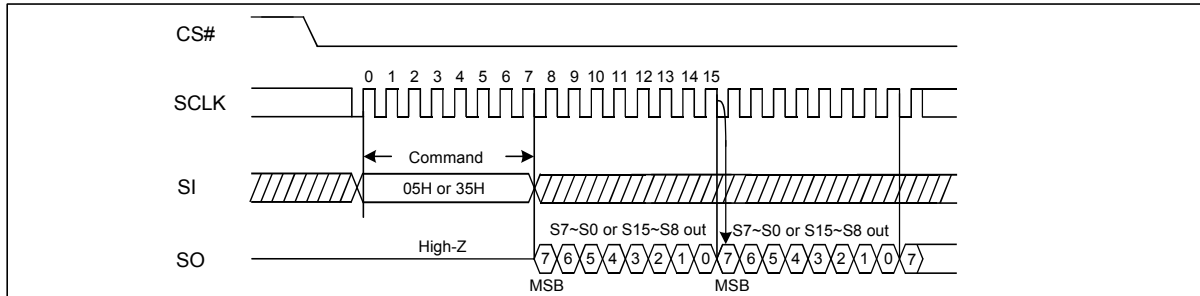
**Figure3. Write Disable Sequence Diagram**



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

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

**Figure4. Read Status Register Sequence Diagram**



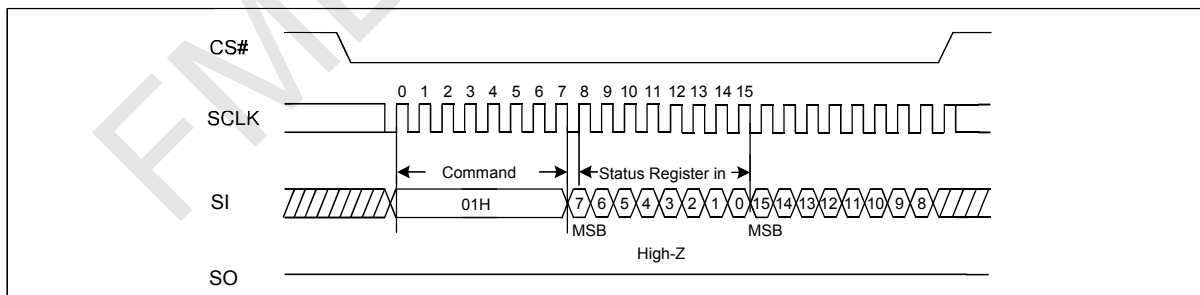
**7.4. Write Status Register (WRSR) (01H)**

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

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

The Write Status Register (WRSR) command allows the user to change the values of the Block Protect (BP4, BP3, BP2, BP1, BP0) bits, to define the size of the area that is to be treated as read-only, as defined in Table1. The Write Status Register (WRSR) command 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. The Write Status Register (WRSR) command is not executed once the Hardware Protected Mode is entered.

**Figure5. Write Status Register Sequence Diagram**

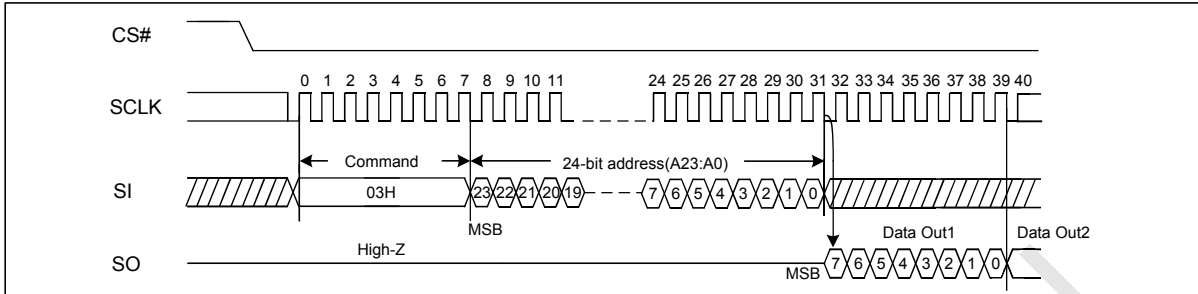


**7.5. Read Data Bytes (READ) (03H)**

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

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

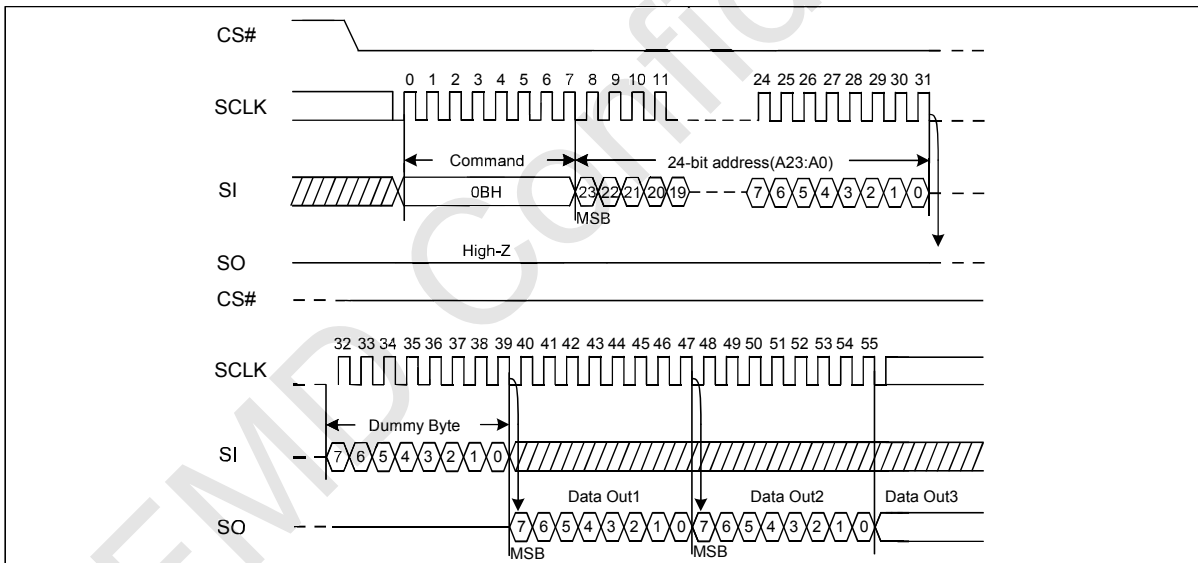
**Figure6. Read Data Bytes Sequence Diagram**



**7.6. Read Data Bytes At Higher Speed (Fast Read) (0BH)**

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

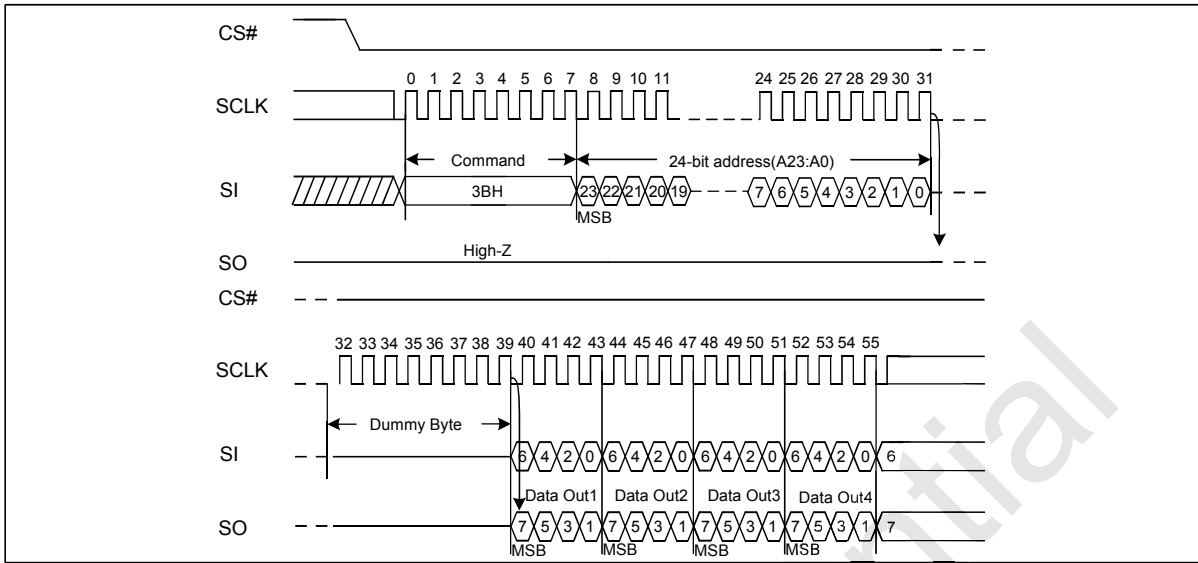
**Figure7. Read Data Bytes at Higher Speed Sequence Diagram**



**7.7. Dual Output Fast Read (3BH)**

The Dual Output Fast Read command is followed by 3-byte address (A23-A0) and a dummy byte, each bit being latched in during the rising edge of SCLK, then the memory contents are shifted out 2-bit per clock cycle from SI and SO. The command sequence is shown in followed Figure8. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out.

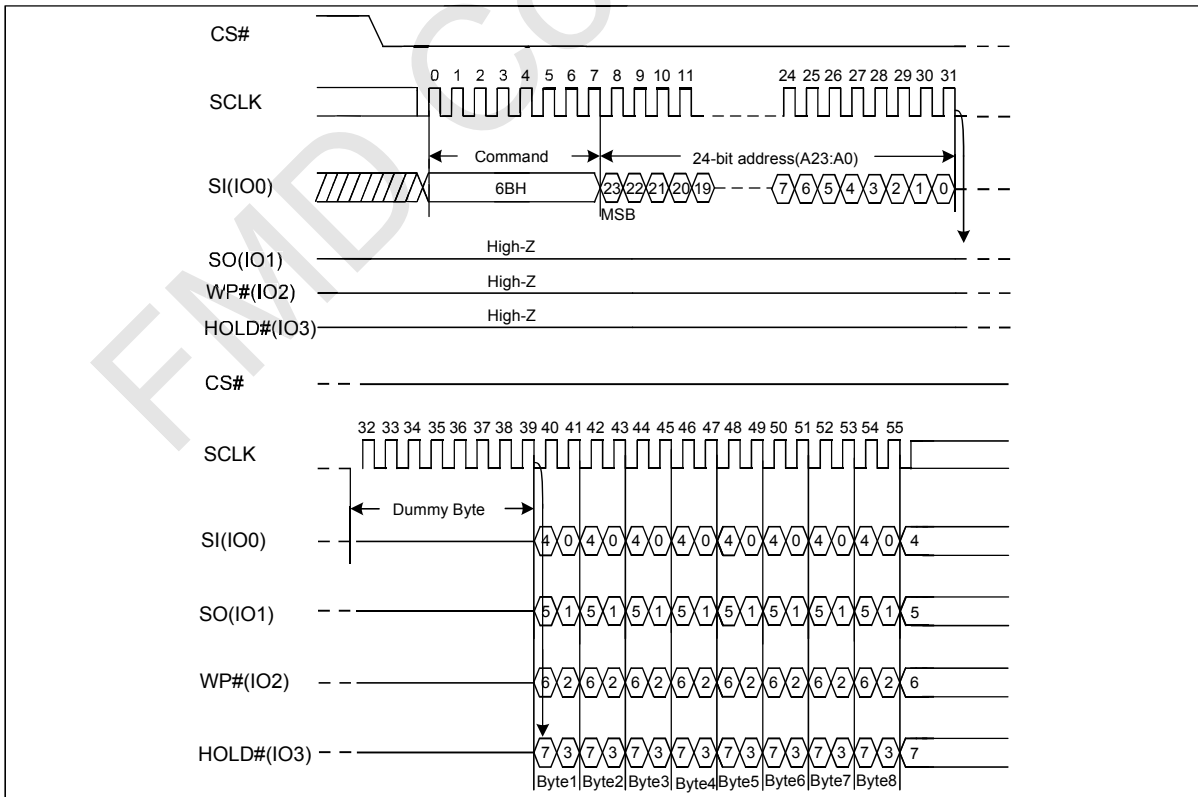
**Figure8. Dual Output Fast Read Sequence Diagram**



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

The Quad Output Fast Read command is followed by 3-byte address (A23-A0) and a dummy byte, each bit being latched in during the rising edge of SCLK, then the memory contents are shifted out 4-bit per clock cycle from IO3, IO2, IO1 and IO0. The command sequence is shown in followed Figure9. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out.

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



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

The Dual I/O Fast Read command is similar to the Dual Output Fast Read command but with the capability to input the 3-byte address (A23-0) and a “Continuous Read Mode” byte 2-bit per clock by SI and SO, each bit being latched in during the rising edge of SCLK, then the memory contents are shifted out 2-bit per clock cycle from SI and SO. The command sequence is shown in followed Figure10. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. To ensure optimum performance the High Speed mode (HSM) command (A3H) must be executed once, prior to the Dual I/O Fast Read command.

#### Dual I/O Fast Read with “Continuous Read Mode”

The Dual I/O Fast Read command can further reduce command overhead through setting the “Continuous Read Mode” bits (M7- 0) after the input 3-byte address (A23-A0). If the “Continuous Read Mode” bits (M5- 4) =(1, 0), then the next Dual I/O Fast Read command (after CS# is raised and then lowered) does not require the BBH command code. The command sequence is shown in followed Figure11. If the “Continuous Read Mode” bits (M5- 4) do not equal (1, 0), the next command requires the first BBH command code, thus returning to normal operation. A “Continuous Read Mode” Reset command can be used to reset (M5- 4) before issuing normal command.

**Figure10. Dual I/O Fast Read Sequence Diagram (M5-4≠(1, 0))**

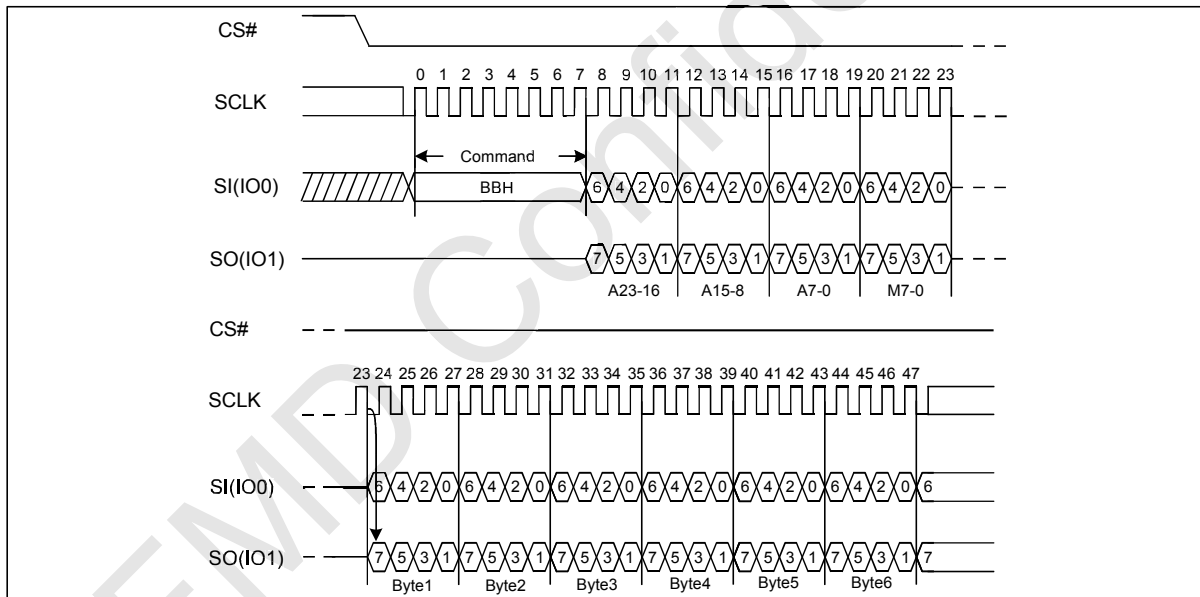
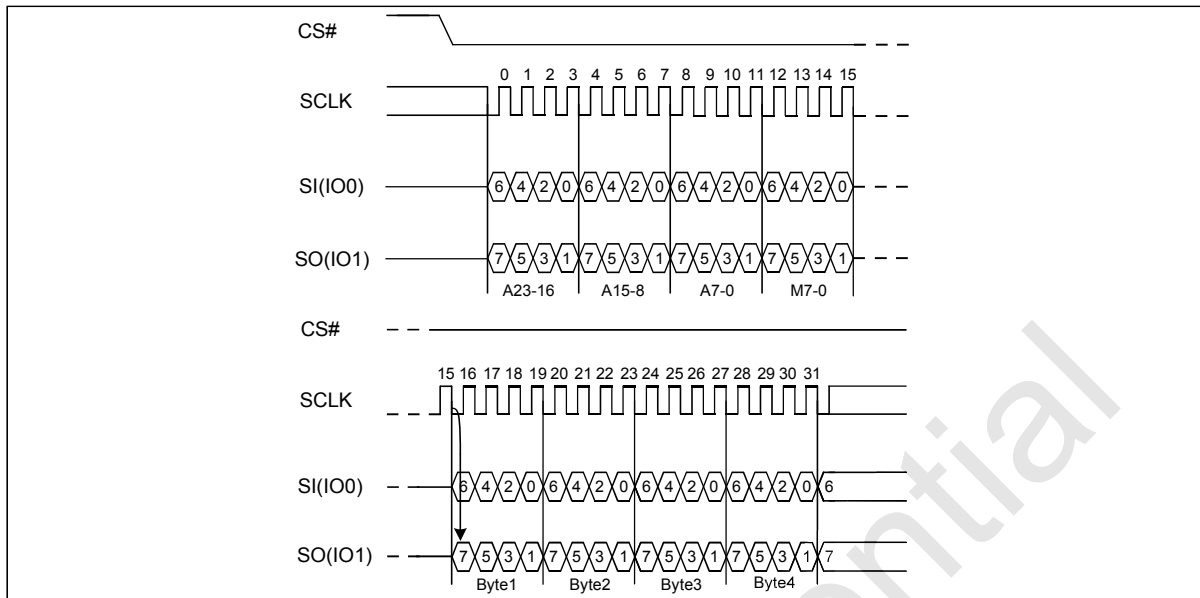


Figure11. Dual I/O Fast Read Sequence Diagram (M5-4= (1, 0))



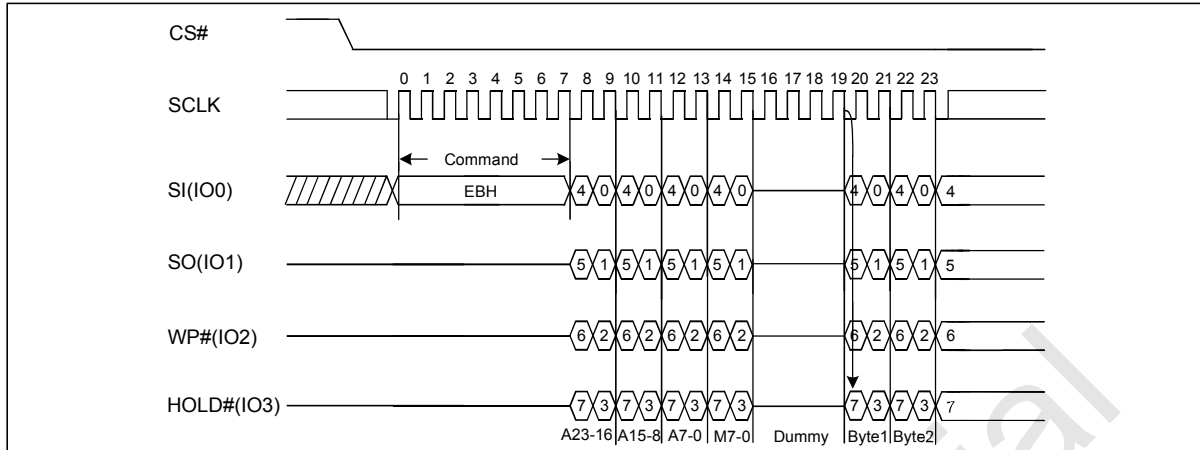
### 7.10. Quad I/O Fast Read (EBH)

The Quad I/O Fast Read command is similar to the Dual I/O Fast Read command but with the capability to input the 3-byte address (A23-0) and a “Continuous Read Mode” byte and 4-dummy clock 4-bit per clock by IO0, IO1, IO3, IO4, each bit being latched in during the rising edge of SCLK, then the memory contents are shifted out 4-bit per clock cycle from IO0, IO1, IO2, IO3. The command sequence is shown in followed Figure12. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. The Quad Enable bit (QE) of Status Register (S9) must be set to enable for the Quad I/O Fast read command. To ensure optimum performance the High Speed mode (HSM) command (A3H) must be executed once, prior to the Quad I/O Fast Read command.

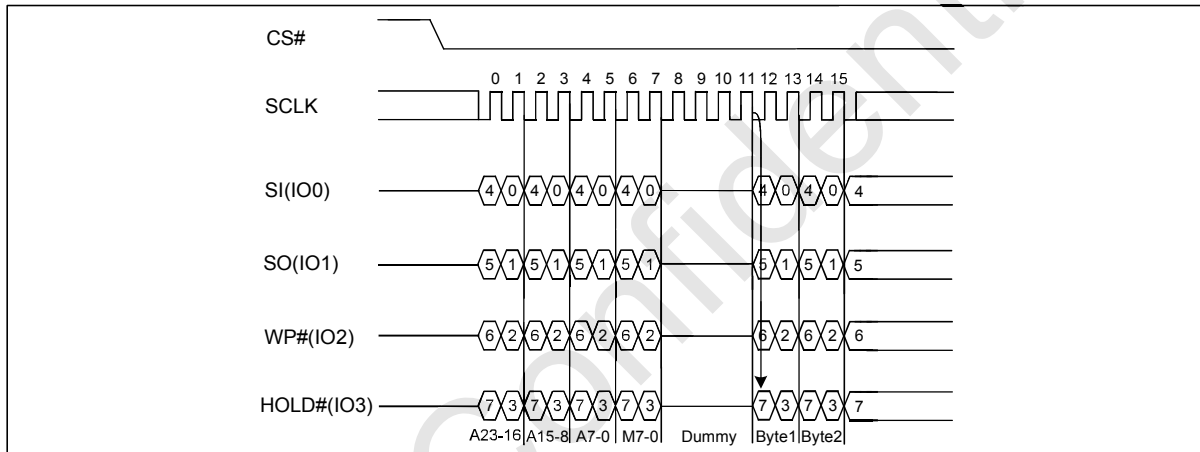
#### Quad I/O Fast Read with “Continuous Read Mode”

The Quad I/O Fast Read command can further reduce command overhead through setting the “Continuous Read Mode” bits (M7-0) after the input 3-byte address (A23-A0). If the “Continuous Read Mode” bits (M5- 4)=(1, 0), then the next Quad I/O Fast Read command (after CS# is raised and then lowered) does not require the EBH command code. The command sequence is shown in followed Figure13. If the “Continuous Read Mode” (M5- 4) do not equal (1, 0), the next command requires the first EBH command code, thus returning to normal operation. A “Continuous Read Mode” Reset command can be used to reset (M5- 4) before issuing normal command.

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



**Figure13. Quad I/O Fast Read Sequence Diagram (M5-4= (1, 0))**



### 7.11. Quad I/O Word Fast Read (E7H)

The Quad I/O Word Fast Read command is similar to the Quad I/O Fast Read command except that the lowest address bit (A0) must equal 0 and only 2-dummy clock. The command sequence is shown in followed Figure14. 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 Quad Enable bit (QE) of Status Register (S9) must be set to enable for the Quad I/O Word Fast read command. To ensure optimum performance the High Speed mode (HSM) command (A3H) must be executed once, prior to the Quad I/O Word Fast Read command.

#### Quad I/O Word Fast Read with “Continuous Read Mode”

The Quad I/O Word Fast Read command can further reduce command overhead through setting the “Continuous Read Mode” bits (M7-0) after the input 3-byte address (A23-A0). If the “Continuous Read Mode” bits (M5- 4) =(1, 0), then the next Quad I/O Word Fast Read command (after CS# is raised and then lowered) does not require the E7H command code. The command sequence is shown in followed Figure15. If the “Continuous Read Mode” bits (M5- 4) do not equal (1, 0), the next command requires the first E7H command code, thus returning to normal operation. A “Continuous Read Mode” Reset command can be used to reset (M7-0) before issuing normal command.



Figure14. Quad I/O Word Fast Read Sequence Diagram (M5-4≠(1, 0))

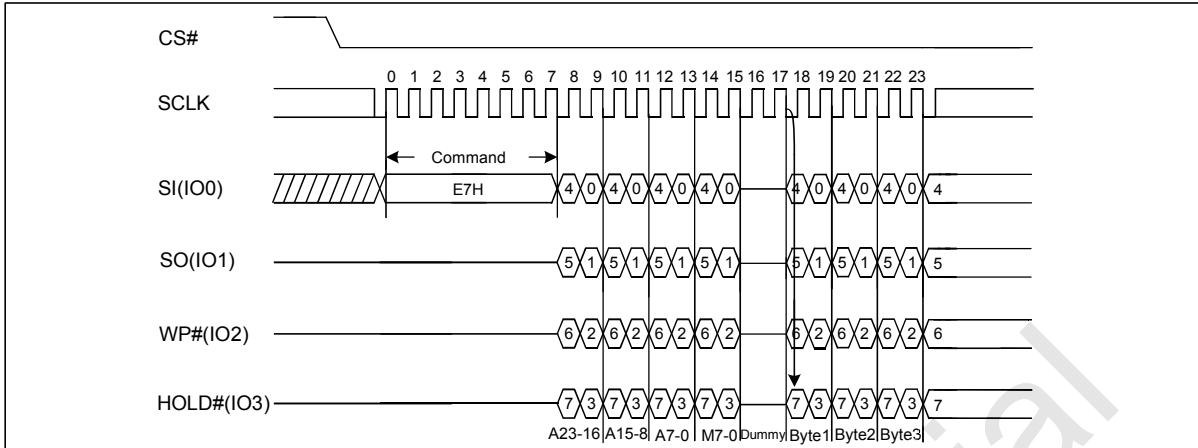
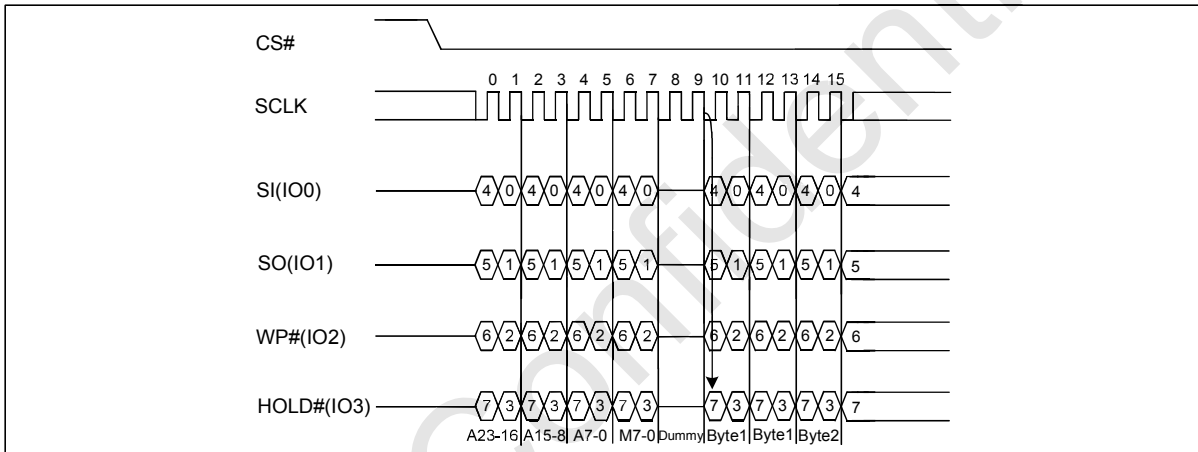


Figure15. Quad I/O Word Fast Read Sequence Diagram (M5-4= (1, 0))



### 7. 12. Page Program (PP) (02H)

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

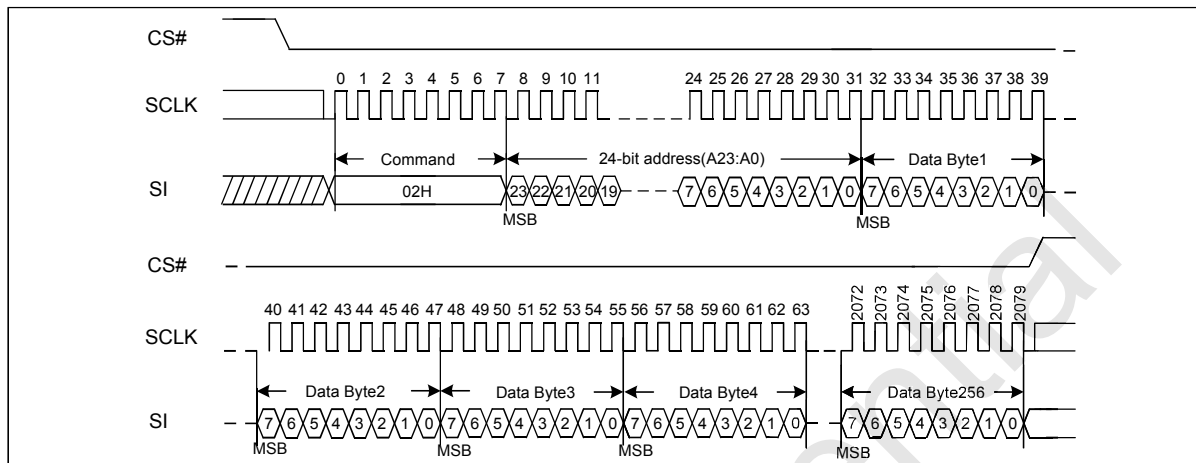
The Page Program (PP) command is entered by driving CS# Low, followed by the command code, three address bytes and at least one data byte on SI. If the 8 least significant address bits (A7-A0) are not all zero, all transmitted data that goes beyond the end of the current page are programmed from the start address of the same page (from the address whose 8 least significant bits (A7-A0) are all zero). CS# must be driven low for the entire duration of the sequence. The Page Program command sequence: CS# goes low→sending Page Program command→3-byte address on SI→at least 1 byte data on SI→ CS# goes high. The command sequence is shown in Figure16. If more than 256 bytes are sent to the device, previously latched data are discarded and the last 256 data bytes are guaranteed to be programmed correctly within the same page. If less than 256 data bytes are sent to device, they are correctly programmed at the requested addresses without having any effects on the other bytes of the same page. CS# must be driven high after the eighth bit of the last data byte has been latched in; otherwise the Page Program (PP) command is not executed.

As soon as CS# is driven high, the self-timed Page Program cycle (whose duration is  $t_{pp}$ ) is initiated. While the Page Program cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Page Program cycle, and is 0

when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

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

**Figure16. Page Program Sequence Diagram**



### 7. 13. Quad Page Program (QPP) (32H)

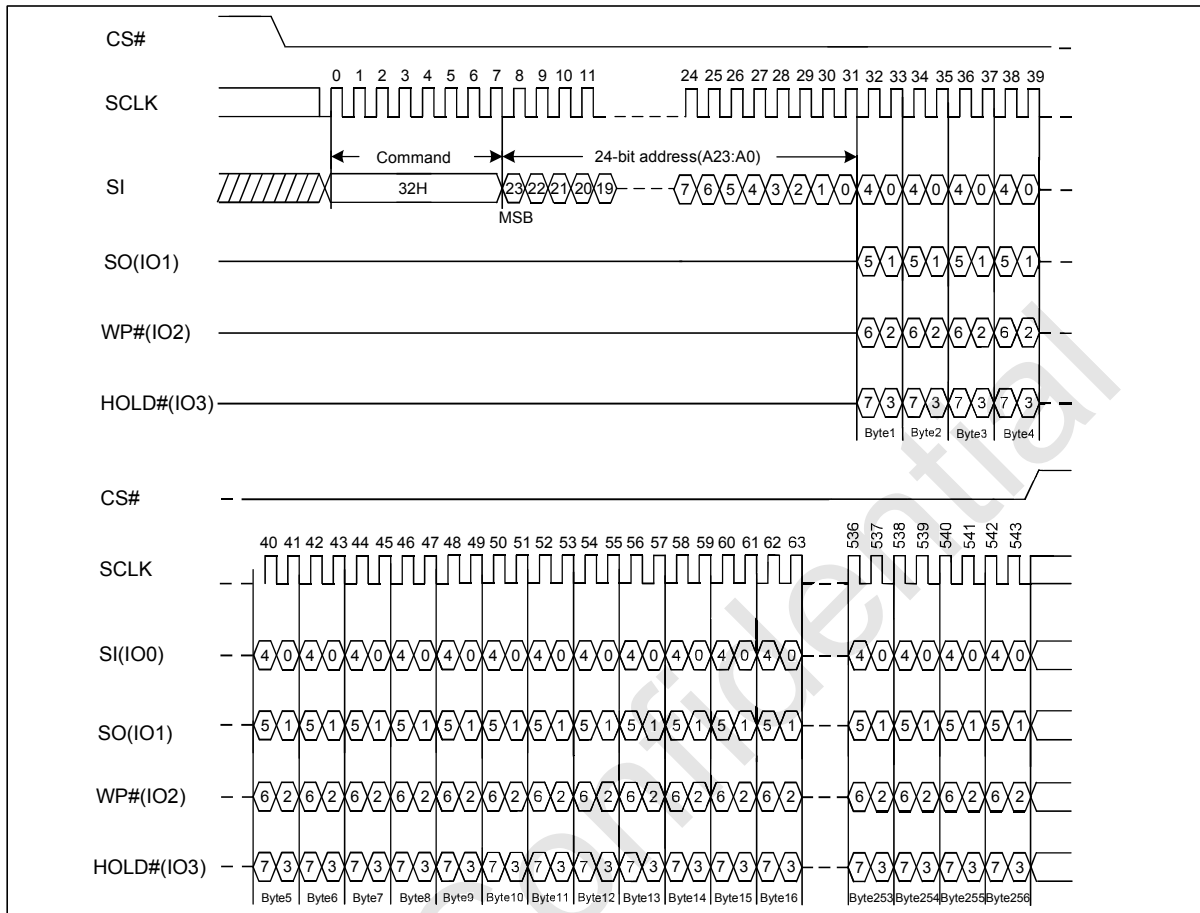
The Quad Page Program command is for programming the memory using four pins: IO0, IO1, IO2, and IO3. To use Quad Page Program the Quad enable in status register Bit9 must be set (QE=1). A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit before sending the Page Program command. The Quad Page Program command is entered by driving CS# Low, followed by the command code (32H), three address bytes and at least one data byte on IO pins.

The command sequence is shown in Figure17. If more than 256 bytes are sent to the device, previously latched data are discarded and the last 256 data bytes are guaranteed to be programmed correctly within the same page. If less than 256 data bytes are sent to device, they are correctly programmed at the requested addresses without having any effects on the other bytes of the same page. CS# must be driven high after the eighth bit of the last data byte has been latched in; otherwise the Quad Page Program command is not executed.

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

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

Figure17. Quad Page Program Sequence Diagram

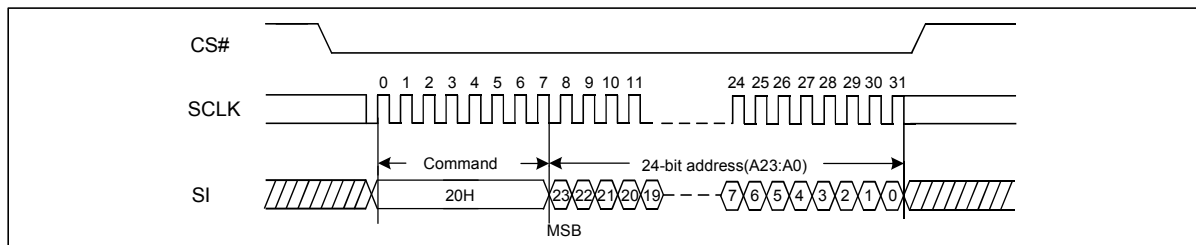


### 7.14. Sector Erase (SE) (20H)

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

The Sector Erase command sequence: CS# goes low → sending Sector Erase command → 3-byte address on SI → CS# goes high. The command sequence is shown in Figure18. CS# must be driven high after the eighth bit of the last address byte has been latched in; otherwise the Sector Erase (SE) command is not executed. As soon as CS# is driven high, the self-timed Sector Erase cycle (whose duration is  $t_{SE}$ ) is initiated. While the Sector Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Sector Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. A Sector Erase (SE) command applied to a sector which is protected by the Block Protect (BP4, BP3, BP2, BP1, BP0) bit (see Table1.0&1.1) is not executed.

**Figure18. Sector Erase Sequence Diagram**

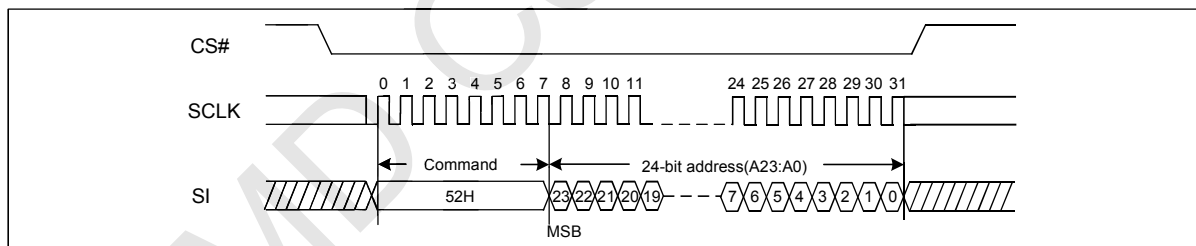


**7.15. 32KB Block Erase (BE) (52H)**

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

The 32KB Block Erase command sequence: CS# goes low → sending 32KB Block Erase command → 3-byte address on SI → CS# goes high. The command sequence is shown in Figure19. CS# must be driven high after the eighth bit of the last address byte has been latched in; otherwise the 32KB Block Erase (BE) command is not executed. As soon as CS# is driven high, the self-timed Block Erase cycle (whose duration is  $t_{BE}$ ) 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 32KB Block Erase (BE) command applied to a block which is protected by the Block Protect (BP4, BP3, BP2, BP1, BP0) bits (see Table1.0&1.1) is not executed.

**Figure19. 32KB Block Erase Sequence Diagram**



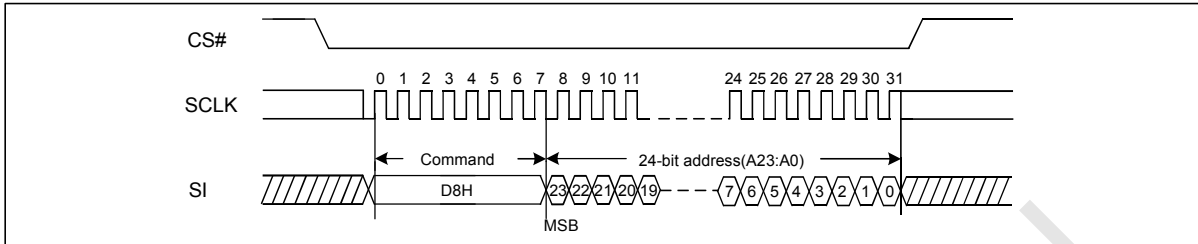
**7.16. 64KB Block Erase (BE) (D8H)**

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

The 64KB Block Erase command sequence: CS# goes low → sending 64KB Block Erase command → 3-byte address on SI → CS# goes high. The command sequence is shown in Figure20. CS# must be driven high after the eighth bit of the last address byte has been latched in; otherwise the 64KB Block Erase (BE) command is not executed. As soon as CS# is driven high, the self-timed Block Erase cycle (whose duration is  $t_{BE}$ ) 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 64KB Block Erase (BE) command applied to a block which is protected by the Block Protect (BP4, BP3, BP2, BP1, BP0) bits (see Table 1.0&1.1) is not executed.

**Figure 20. 64KB Block Erase Sequence Diagram**

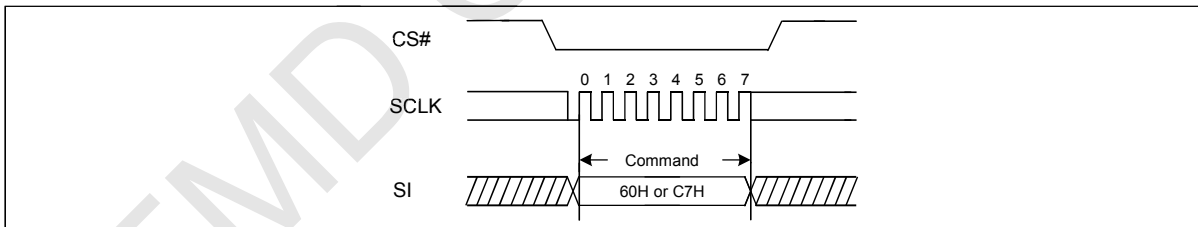


**7.17. Chip Erase (CE) (60/C7H)**

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

The Chip Erase command sequence: CS# goes low → sending Chip Erase command → CS# goes high. The command sequence is shown in Figure 21. CS# must be driven high after the eighth bit of the command code has been latched in, otherwise the Chip Erase command is not executed. As soon as CS# is driven high, the self-timed Chip Erase cycle (whose duration is  $t_{CE}$ ) is initiated. While the Chip Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Chip Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. The Chip Erase (CE) command is executed if the Block Protect (BP2, BP1, BP0) bits and CMP are all 0 or all 1. The Chip Erase (CE) command is ignored if one or more sectors are protected.

**Figure 21. Chip Erase Sequence Diagram**



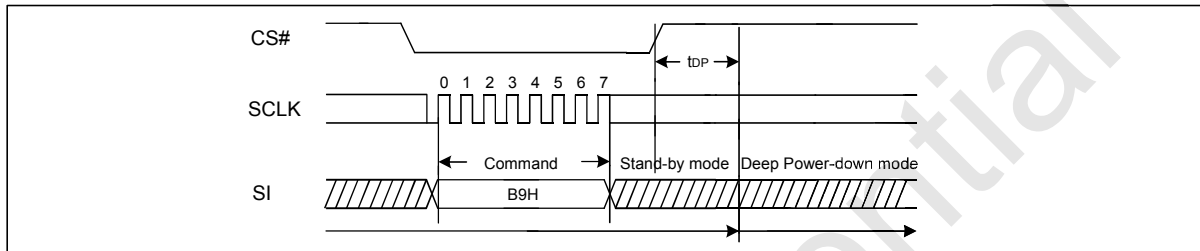
**7.18. Deep Power-Down (DP) (B9H)**

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

The Deep Power-Down Mode automatically stops at Power-Down, and the device always Power-Up in the Standby Mode. The Deep Power-Down (DP) command is entered by driving CS# low, followed by the command code on SI. CS# must be driven low for the entire duration of the sequence.

The Deep Power-Down command sequence: CS# goes low → sending Deep Power-Down command → CS# goes high. The command sequence is shown in Figure22. CS# must be driven high after the eighth bit of the command code has been latched in; otherwise the Deep Power-Down (DP) command is not executed. As soon as CS# is driven high, it requires a delay of  $t_{DP}$  before the supply current is reduced to  $I_{CC2}$  and the Deep Power-Down Mode is entered. Any Deep Power-Down (DP) command, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

**Figure22. Deep Power-Down Sequence Diagram**



### 7.19. Release from Deep Power-Down And Read Device ID (RDI) (ABH)

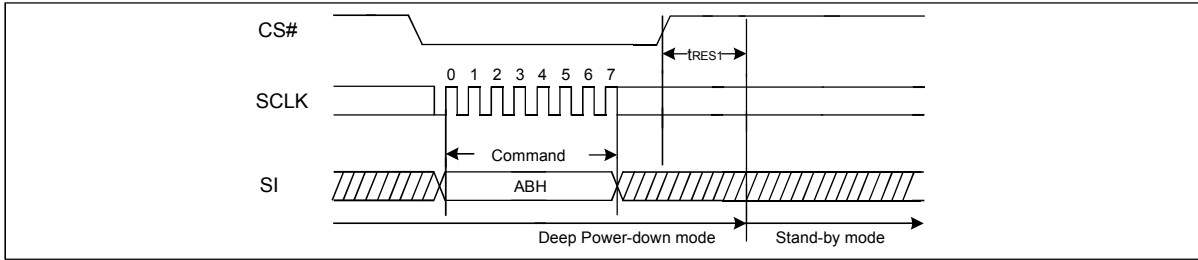
The Release from Power-Down and Read/Device ID command is a multi-purpose command. It can be used to release the device from the Power-Down state or obtain the devices electronic identification (ID) number.

To release the device from the Power-Down state, the command is issued by driving the CS# pin low, shifting the instruction code “ABH” and driving CS# high as shown in Figure23. Release from Power-Down will take the time duration of  $t_{RES1}$  (See AC Characteristics) before the device will resume normal operation and other command are accepted. The CS# pin must remain high during the  $t_{RES1}$  time duration.

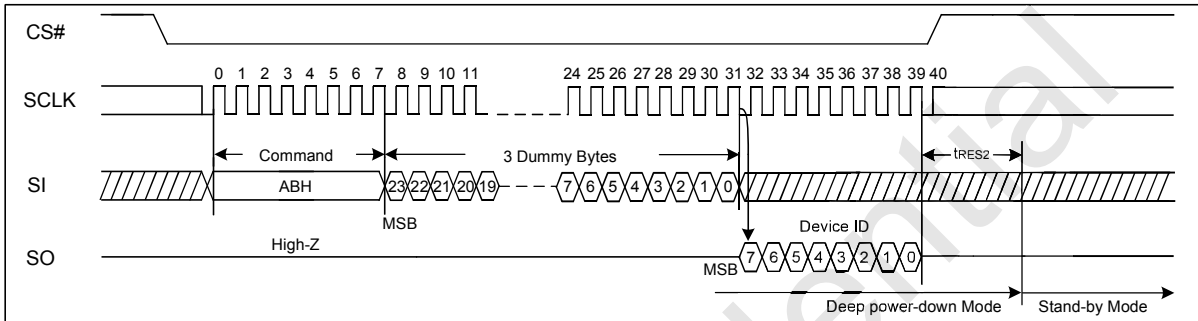
When used only to obtain the Device ID while not in the Power-Down state, the command is initiated by driving the CS# pin low and shifting the instruction code “ABH” followed by 3-dummy byte. The Device ID bits are then shifted out on the falling edge of SCLK with most significant bit (MSB) first as shown in Figure23. The Device ID value for the FT25H16 is listed in Manufacturer and Device Identification table. The Device ID can be read continuously. The command is completed by driving CS# high.

When used to release the device from the Power-Down state and obtain the Device ID, the command is the same as previously described, and shown in Figure23, except that after CS# is driven high it must remain high for a time duration of  $t_{RES2}$  (See AC Characteristics). After this time duration the device will resume normal operation and other command will be accepted. If the Release from Power-Down/Device ID command is issued while an Erase, Program or Write cycle is in process (when WIP equal 1) the command is ignored and will not have any effects on the current cycle.

**Figure23. Release Power-Down Sequence Diagram**



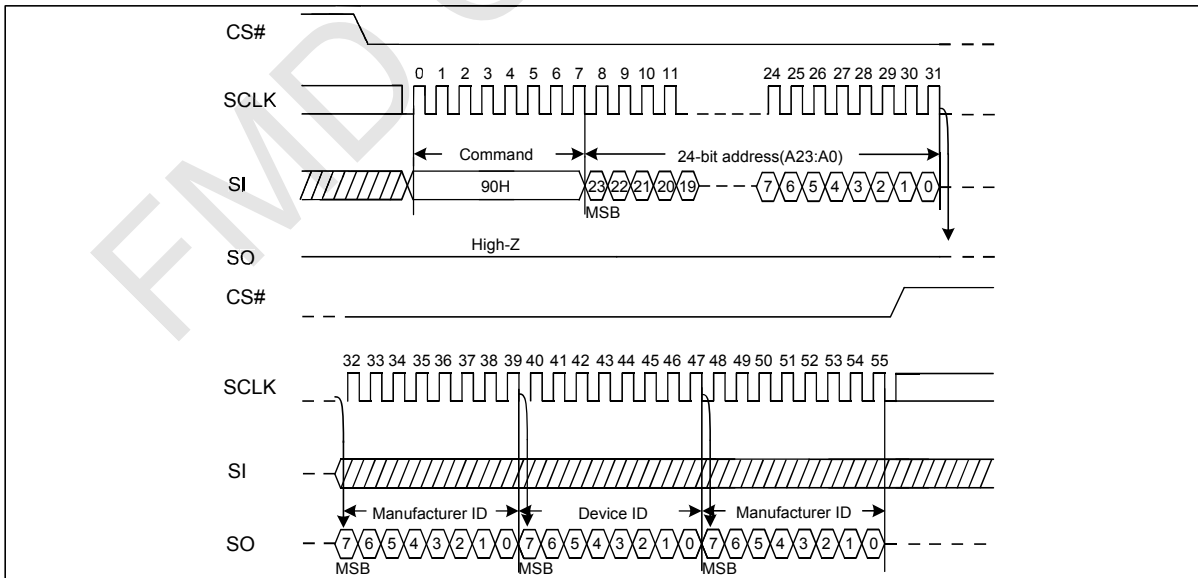
**Figure24. Release Power-Down/Read Device ID Sequence Diagram**



## 7.20. Read Manufacturer ID/ Device ID (REMS) (90H)

The Read Manufacturer/Device ID command is an alternative to the Release from Power-Down / Device ID command that provides both the JEDEC assigned Manufacturer ID and the specific Device ID. The command is initiated by driving the CS# pin low and shifting the command code “90H” followed by a 24-bit address (A23-A0) of 000000H. After which, the Manufacturer ID and the Device ID are shifted out on the falling edge of SCLK with most significant bit (MSB) first as shown in Figure25. If the 24-bit address is initially set to 000001H, the Device ID will be read first.

**Figure25. Read Manufacturer ID/ Device ID Sequence Diagram**



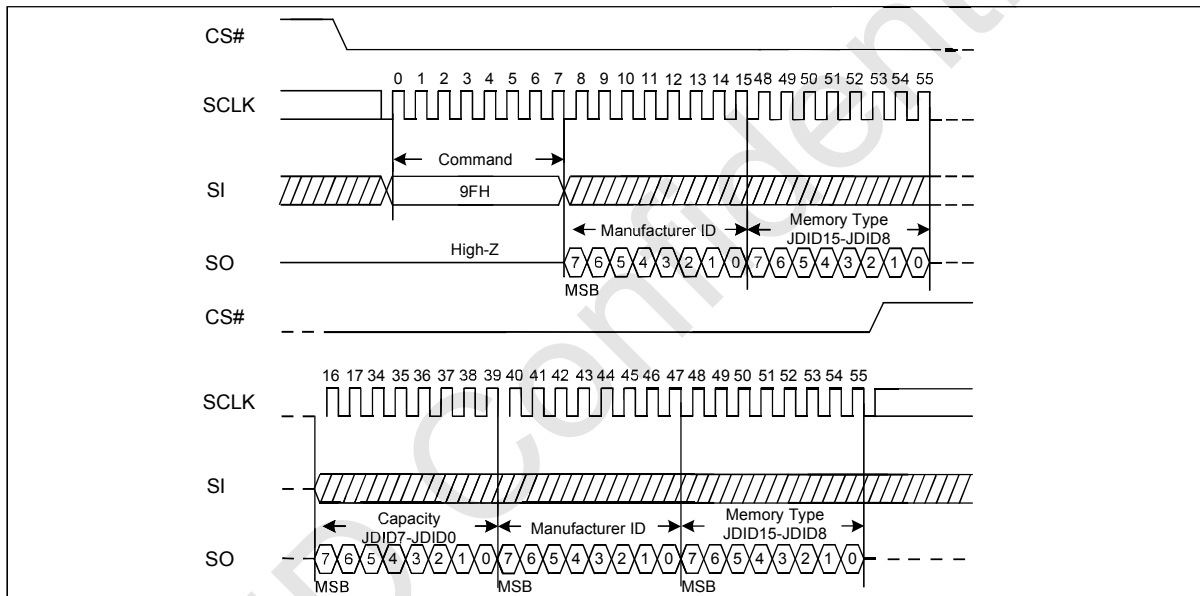


### 7.21. Read Identification (RDID) (9FH)

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

The device is first selected by driving CS# to low. Then, the 8-bit command code for the command is shifted in. This is followed by the 24-bit device identification, stored in the memory, being shifted out on Serial Data Output, each bit being shifted out during the falling edge of Serial Clock. The command sequence is shown in Figure26. The Read Identification (RDID) command is terminated by driving CS# to high at any time during data output. When CS# is driven high, the device is put in the Standby Mode. Once in the Standby Mode, the device waits to be selected, so that it can receive, decode and execute commands.

**Figure26. Read Identification ID Sequence Diagram**

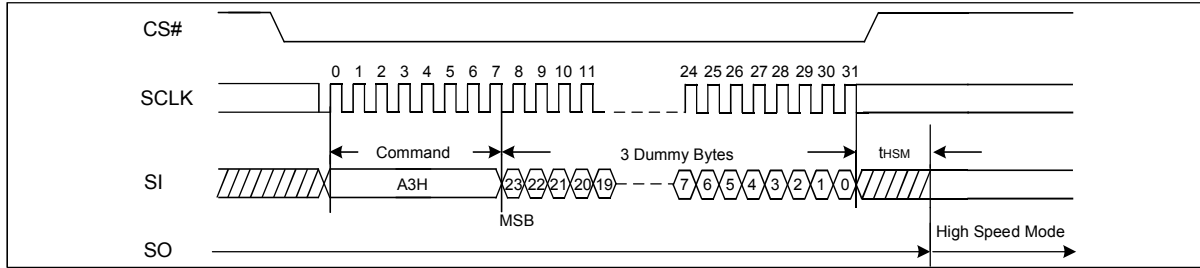


### 7.22. High Speed Mode (HSM) (A3H)

The High Speed Mode (HSM) command must be executed prior to Dual or Quad I/O commands when operating at high frequencies (see  $f_R$  and  $f_{C1}$  in AC Electrical Characteristics). This command allows pre-charging of internal charge pumps so the voltages required for accessing the flash memory array are readily available. The command sequence: CS# goes low → Sending A3H command → Sending 3-dummy byte → CS# goes high. See Figure27. After the HSM command is executed, the device will maintain a slightly higher standby current (ICC8) than standard SPI operation. The Release from Power-Down or HSM command (ABH) can be used to return to standard SPI standby current (ICC1). In addition, Write Enable command (06H) and Power-Down command (B9H) will also release the device from HSM mode back to standard SPI standby state.



**Figure27. High Speed mode Sequence Diagram**

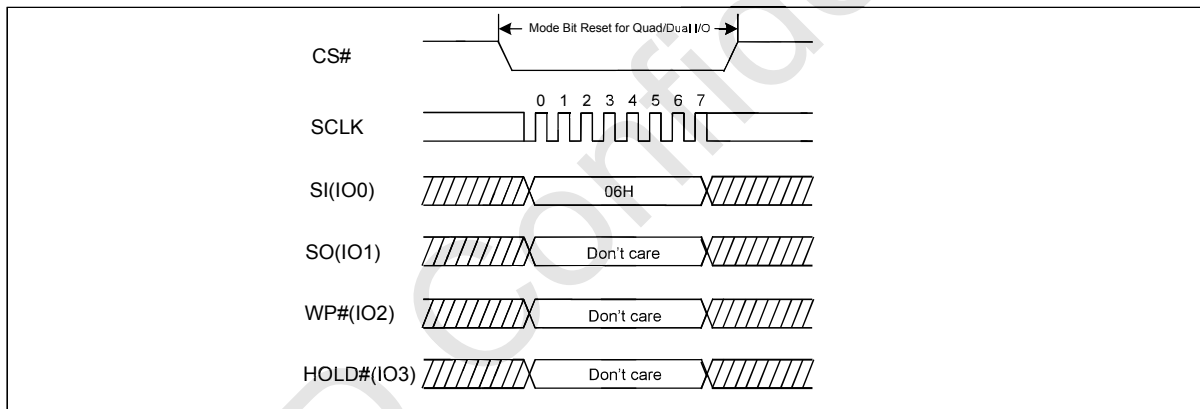


**7.23. Continuous Read Mode Reset (CRMR) (FFH)**

The Dual/Quad I/O Fast Read operations, “Continuous Read Mode” bits (M7-0) are implemented to further reduce command overhead. By setting the (M7-0) to AXH, the next Dual/Quad I/O Fast Read operations do not require the BBH/EBH/E7H command code.

Because the FT25H16 has no hardware reset pin, so if Continuous Read Mode bits are set to “AXH”, the FT25H16 will not recognize any standard SPI commands. So Continuous Read Mode Reset command will release the Continuous Read Mode from the “AXH” state and allow standard SPI command to be recognized. The command sequence is show in Figure28.

**Figure28. Continuous Read Mode Reset Sequence Diagram**

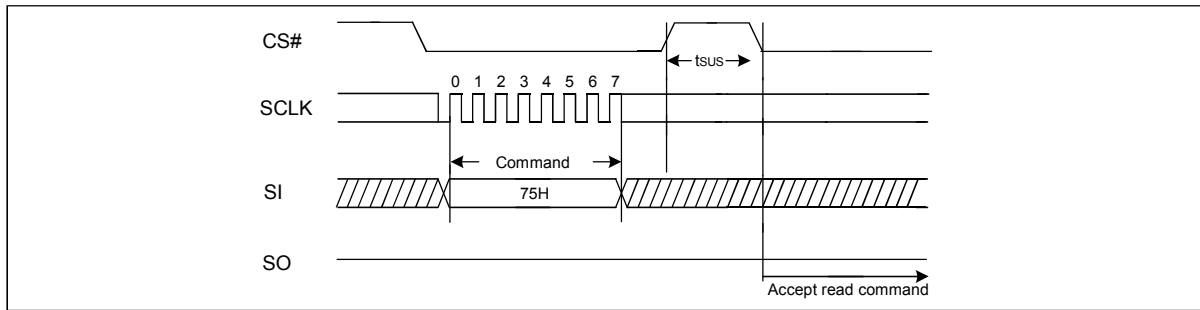


**7.24. Program/Erase Suspend (PES) (75H)**

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

The Program/Erase Suspend command will be accepted by the device only if the SUS bit in the Status Register equal to 0 and WIP bit equal to 1 while a Page Program or a Sector or Block Erase operation is on-going. If the SUS bit equal to 1 or WIP bit equal to 0, the Suspend command will be ignored by the device. The WIP bit will be cleared form 1 to 0 within “t<sub>SUS</sub>” and the SUS bit will be set from 0 to 1 immediately after Program/Erase Suspend. A power-off during the suspend period will reset the device and release the suspend state. The command sequence is show in Figure29.

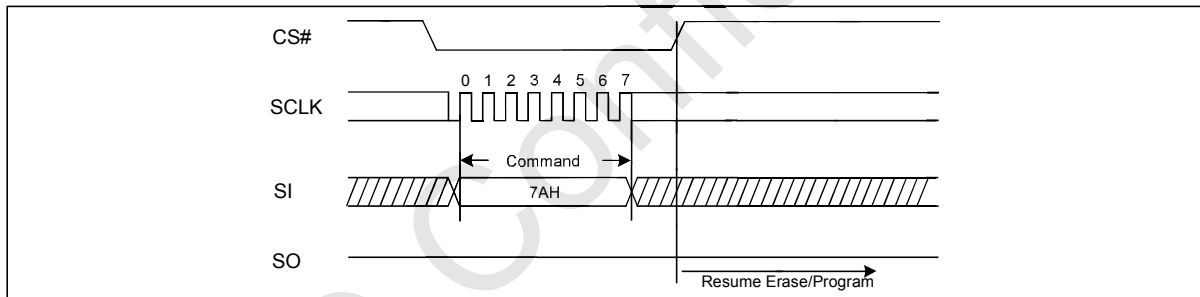
**Figure29. Program/Erase Suspend Sequence Diagram**



**7.25. Program/Erase Resume (PER) (7AH)**

The Program/Erase Resume command must be written to resume the program or sector/block erase operation after a Program/Erase Suspend command. The Program/Erase command will be accepted by the device only if the SUS bit equal to 1 and the WIP bit equal to 0. After issued the SUS bit in the status register will be cleared from 1 to 0 immediately, the WIP bit will be set from 0 to 1 within 200ns and the Sector or Block will complete the erase operation or the page will complete the program operation. The Program/Erase Resume command will be ignored unless a Program/Erase Suspend is active. The command sequence is show in Figure30.

**Figure30. Program/Erase Resume Sequence Diagram**



**7.26. Erase Security Registers (44H)**

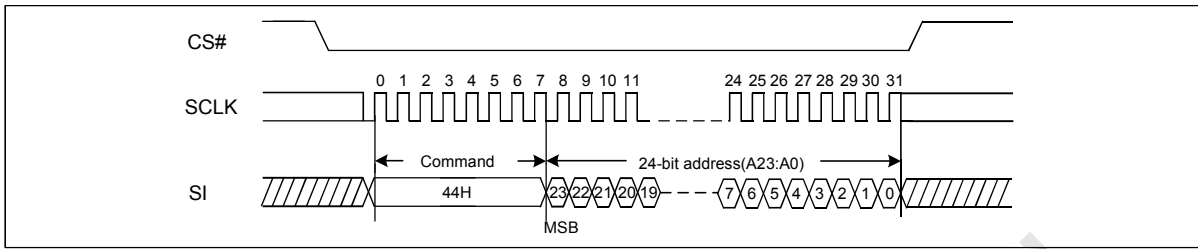
The FT25H16 provides four 256-byte Security Registers which can be erased and programmed individually. These registers may be used by the system manufacturers to store security and other important information separately from the main memory array.

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

The Erase Security Registers command sequence: CS# goes low → sending Erase Security Registers command → CS# goes high. The command sequence is shown in Figure31. CS# must be driven high after the eighth bit of the command code has been latched in, otherwise the Erase Security Registers command is not executed. As soon as CS# is driven high, the self-timed Erase Security Registers cycle (whose duration is  $t_{SE}$ ) is initiated. While the Erase Security Registers cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Erase Security Registers cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset. The Security Registers Lock Bit (LB) in the Status Register can be used to OTP protect the security registers. Once the LB bit is set to 1, the Security Registers will be permanently locked; the Erase Security Registers command will be ignored.

Address	A23-A16	A15-A10	A9-A0
Security Registers	00000000	000000	Don't Care

**Figure31. Erase Security Registers command Sequence Diagram**



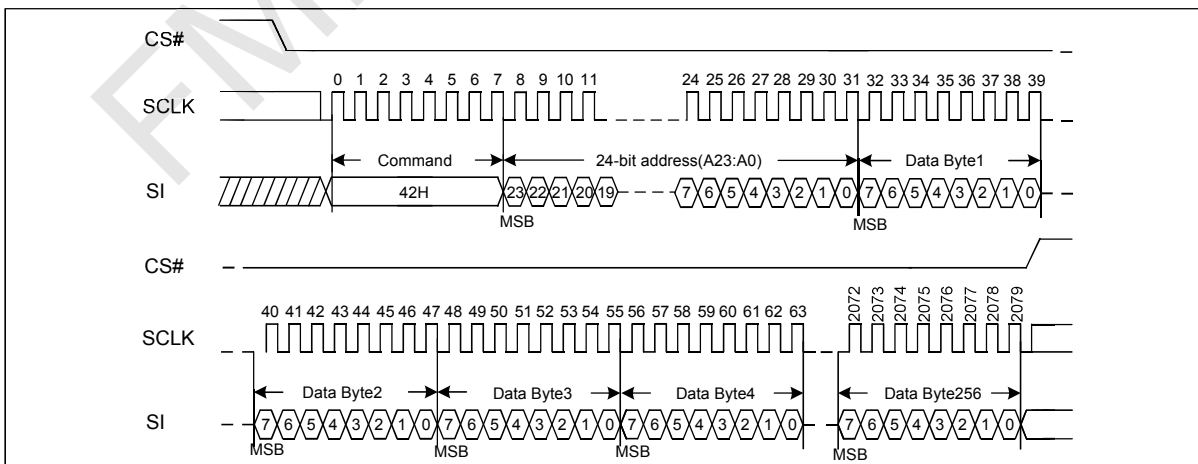
### 7.27. Program Security Registers (42H)

The Program Security Registers command is similar to the Page Program command. It allows from 1 to 256 bytes Security Registers data to be programmed. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit before sending the Program Security Registers command. The Program Security Registers command is entered by driving CS# Low, followed by the command code (42H), three address bytes and at least one data byte on SI. As soon as CS# is driven high, the self-timed Program Security Registers cycle (whose duration is  $t_{PP}$ ) is initiated. While the Program Security Registers cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Program Security Registers cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

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

Address	A23-A16	A15-A8	A7-A0
Security Registers 0	00H	00H	Byte Address
Security Registers 1	00H	01H	Byte Address
Security Registers 2	00H	02H	Byte Address
Security Registers 3	00H	03H	Byte Address

**Figure32. Program Security Registers command Sequence Diagram**

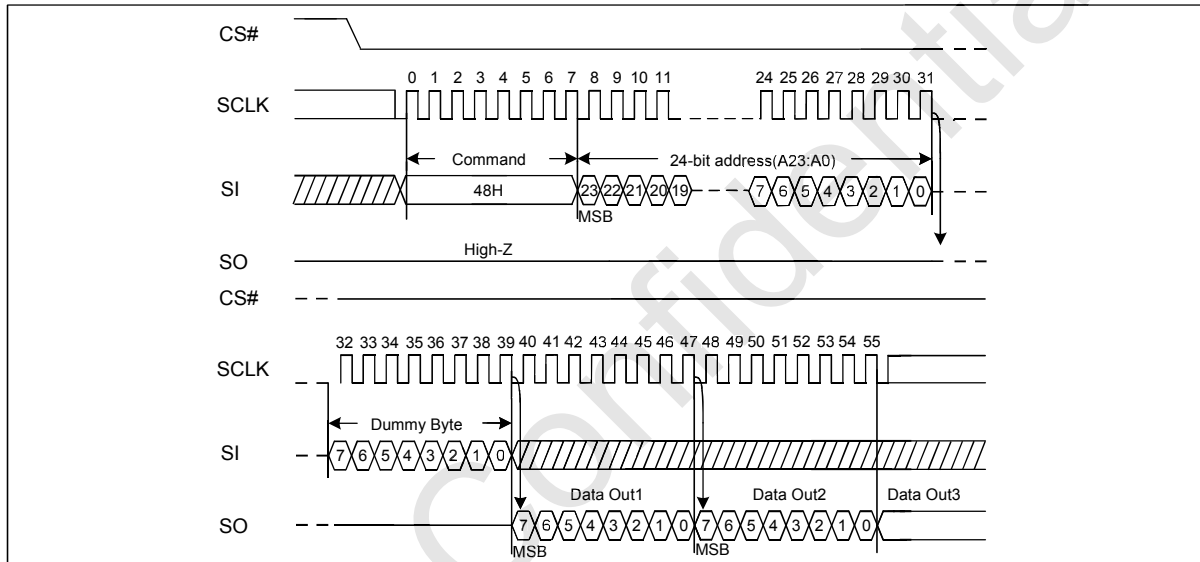


### 7.28. Read Security Registers (48H)

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

Address	A23-A16	A15-A10	A9-A0
Security Registers	00000000	000000	Address

Figure33. Read Security Registers command Sequence Diagram



## 8. ELECTRICAL CHARACTERISTICS

### 8.1. Power-on Timing

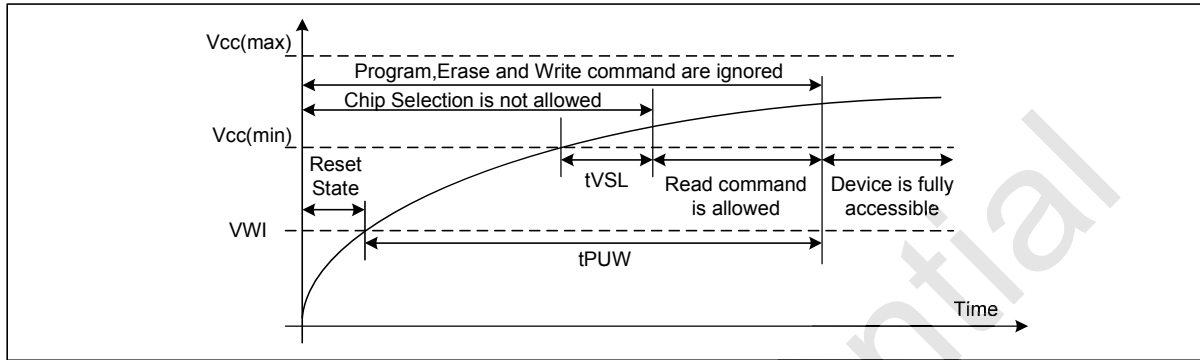


Table3. Power-Up Timing and Write Inhibit Threshold

Symbol	Parameter	Min	Max	Unit
t <sub>VSL</sub>	VCC(min) To CS# Low	10		us
t <sub>PUW</sub>	Time Delay Before Write Instruction	1	10	ms
V <sub>WI</sub>	Write Inhibit Voltage	1	2.5	V

### 8.2. Initial Delivery State

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

### 8.3. Data Retention and Endurance

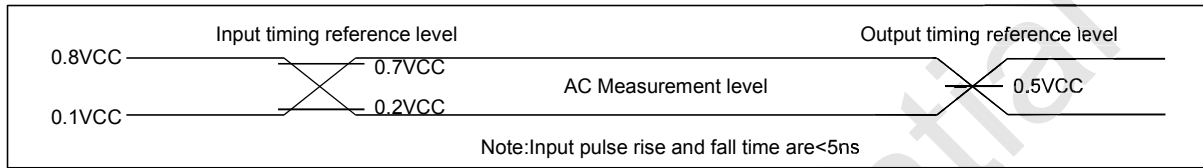
Parameter	Test Condition	Min	Unit
Minimum Pattern Data Retention Time	150°C	10	Years
	125°C	20	Years
Erase/Program Endurance	-40 to 85°C	100K	Cycles

### 8.4. Latch up Characteristics

Parameter	Min	Max
Input Voltage Respect To VSS On I/O Pins	-1.0V	VCC+1.0V
VCC Current	-100mA	100mA

### 8.5. Absolute Maximum Ratings

Parameter	Value	Unit
Ambient Operating Temperature	-40 to 85	°C
Storage Temperature	-65 to 150	°C
Output Short Circuit Current	200	mA
Applied Input/Output Voltage	-0.5 to 4.0	V
VCC	-0.5 to 4.0	V

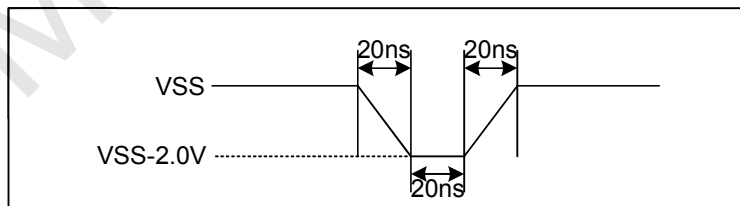


### 8.6. Capacitance Measurement Condition

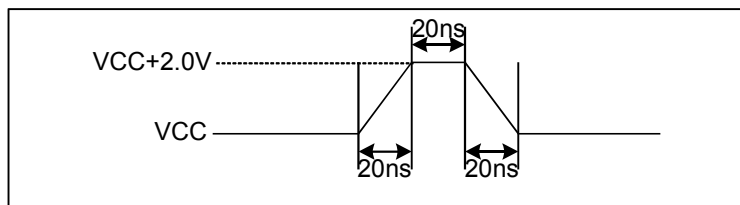
Symbol	Parameter	Min	Typ	Max	Unit	Conditions
C <sub>IN</sub>	Input Capacitance			6	pF	V <sub>IN</sub> =0V
C <sub>OUT</sub>	Output Capacitance			8	pF	V <sub>OUT</sub> =0V
C <sub>L</sub>	Load Capacitance	30			pF	
	Input Rise And Fall time			5	ns	
	Input Pulse Voltage	0.1VCC to 0.8VCC			V	
	Input Timing Reference Voltage	0.2VCC to 0.7VCC			V	
	Output Timing Reference Voltage	0.5VCC			V	

Figure34. Input Test Waveform and Measurement Level

#### Maximum Negative Overshoot Waveform



#### Maximum Positive Overshoot Waveform



### 8.7. DC Characteristics

(T=-40°C~85°C,VCC=2.7~3.6V)

Symbol	Parameter	Test Condition	Min.	Typ	Max.	Unit
I <sub>LI</sub>	Input Leakage Current				±2	μA
I <sub>LO</sub>	Output Leakage Current				±2	μA
I <sub>CC1</sub>	Standby Current	CS#=VCC VIN=VCC or VSS		1	5	μA
I <sub>CC2</sub>	Deep Power-Down Current	CS#=VCC VIN=VCC or VSS		1	5	μA
I <sub>CC3</sub>	Operating Current(Read)	CLK=0.1VCC/0.9VCC at 120MHz, Q=Open(*1 I/O)		15	20	mA
		CLK=0.1VCC/0.9VCC at 80MHz, Q=Open(*1,*2,*4 I/O)		13	18	mA
I <sub>CC4</sub>	Operating Current(PP)	CS#=VCC			10	mA
I <sub>CC5</sub>	Operating Current(WRSR)	CS#=VCC			10	mA
I <sub>CC6</sub>	Operating Current(SE)	CS#=VCC			10	mA
I <sub>CC7</sub>	Operating Current(BE)	CS#=VCC			10	mA
I <sub>CC8</sub>	High Speed Current			600	800	μA
V <sub>IL</sub>	Input Low Voltage		-0.5		0.2VCC	V
V <sub>IH</sub>	Input High Voltage		0.7VCC		VCC+0.4	V
V <sub>OL</sub>	Output Low Voltage	IOL=1.6mA			0.4	V
V <sub>OH</sub>	Output High Voltage	IOH=-100uA	VCC-0.2			V

### 8.8. AC Characteristics

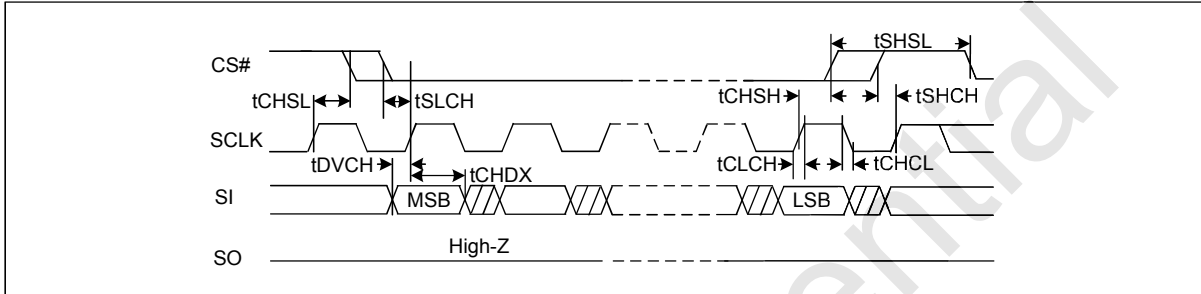
(T=-40°C~85°C, VCC=2.7~3.6V, C<sub>L</sub>=30pf)

Symbol	Parameter	Min.	Typ	Max.	Unit
f <sub>C</sub>	Serial Clock Frequency For:Fast Read(0BH), Dual Output(3BH)	DC		120	MHz
f <sub>C1</sub>	Serial Clock Frequency For: Dual I/O(BBH), Quad I/O(EBH), Quad Output(6BH) (Dual I/O & Quad I/O With High Speed mode)	DC	80	120	MHz
f <sub>C2</sub>	Serial Clock Frequency For: Dual I/O(BBH), Quad I/O(EBH) (Dual I/O & Quad I/O Without High Speed mode)	DC		40	MHz
f <sub>R</sub>	Serial Clock Frequency For:Read(03H)	DC		80	MHz
t <sub>CLH</sub>	Serial Clock High Time	4			ns
t <sub>CLL</sub>	Serial Clock Low Time	4			ns
t <sub>CLCH</sub>	Serial Clock Rise Time(Slew Rate)	0.2			V/ns
t <sub>CHCL</sub>	Serial Clock Fall Time(Slew Rate)	0.2			V/ns
t <sub>SLCH</sub>	CS# Active Setup Time	5			ns
t <sub>CHSH</sub>	CS# Active Hold Time	5			ns
t <sub>SHCH</sub>	CS# Not Active Setup Time	5			ns
t <sub>CHSL</sub>	CS# Not Active Hold Time	5			ns
t <sub>SHSL</sub>	CS# High Time (read/write)	20			ns
t <sub>SHQZ</sub>	Output Disable Time			6	ns
t <sub>CLQX</sub>	Output Hold Time	1			ns
t <sub>DVCH</sub>	Data In Setup Time	2			ns
t <sub>CHDX</sub>	Data In Hold Time	2			ns
t <sub>HLCH</sub>	Hold# Low Setup Time(relative to Clock)	5			ns
t <sub>HHCH</sub>	Hold# High Setup Time(relative to Clock)	5			ns
t <sub>CHHL</sub>	Hold# High Hold Time(relative to Clock)	5			ns
t <sub>CHHH</sub>	Hold# Low Hold Time(relative to Clock)	5			ns
t <sub>HLQZ</sub>	Hold# Low To High-Z Output			6	ns
t <sub>HHQX</sub>	Hold# Low To Low-Z Output			6	ns
t <sub>CLQV</sub>	Clock Low To Output Valid			6.5	ns
t <sub>WHSL</sub>	Write Protect Setup Time Before CS# Low	20			ns
t <sub>SHWL</sub>	Write Protect Hold Time After CS# High	100			ns
t <sub>DP</sub>	CS# High To Deep Power-Down Mode			0.1	us
t <sub>RES1</sub>	CS# High To Standby Mode Without Electronic Signature Read			0.1	us
t <sub>RES2</sub>	CS# High To Standby Mode With Electronic Signature Read			0.1	us
t <sub>HSM</sub>	CS# High To High Speed mode			0.1	us
t <sub>SUS</sub>	CS# High To Next Command After Suspend			2	us

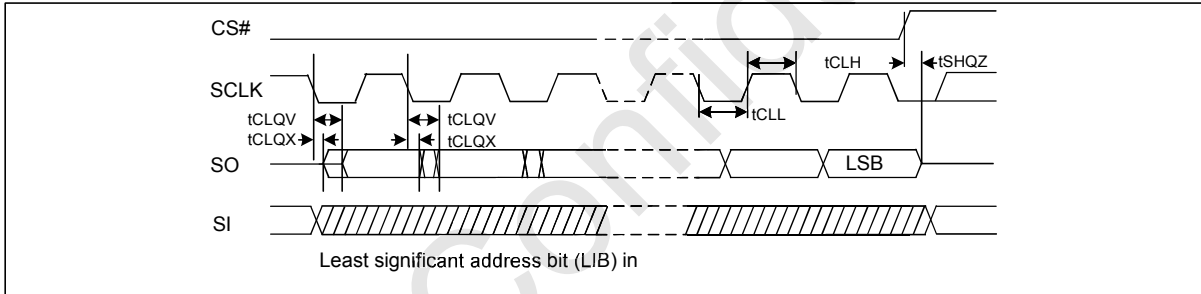


$t_w$	Write Status Register Cycle Time	100	200	ms
$t_{pp}$	Page Programming Time	0.4	2.4	ms
$t_{se}$	Sector Erase Time	120	300	ms
$t_{be}$	Block Erase Time(32K Bytes/64K Bytes)	0.2/0.4	0.5/0.6	s
$t_{ce}$	Chip Erase Time	10	25	s

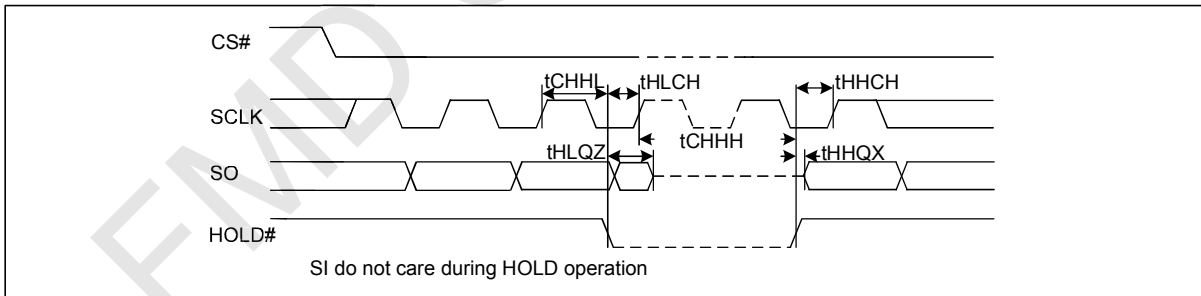
**Figure35. Serial Input Timing**



**Figure36. Output Timing**



**Figure37. Hold Timing**



**9. ORDERING INFORMATION****FT 25 H 16 X X X - X X****PACKING TYPE:**B: Tube  
T: Tape & Reel**OPTION:**R: Rohs  
G: Green**TEMPERATURE RANGE:**

Default: Industrail (-40°C to +85°C)

**PACKAGE:**S: SOP8 150mil  
W: SOP8 208mil  
D: DIP8 300mil  
V: VSOP8 208mil  
T: TSSOP8 173mil**VERSION:**Default: A Version  
B: B Version**DENSITY:**

16: 16M

**TYPE:**H: 3V  
L: 1.8V**DEVICE:**

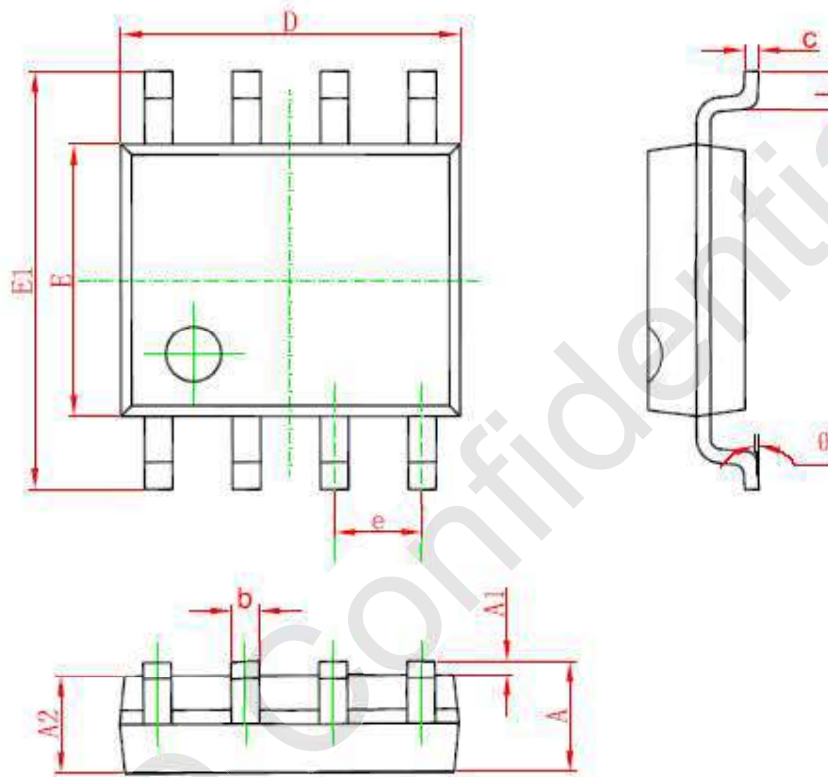
25: Serial Falsh

**NOTE:**

1. Standard bulk shipment is in Tube. Any alternation of packing method (for Tape, Reel and Tray etc.), please advise in advance.

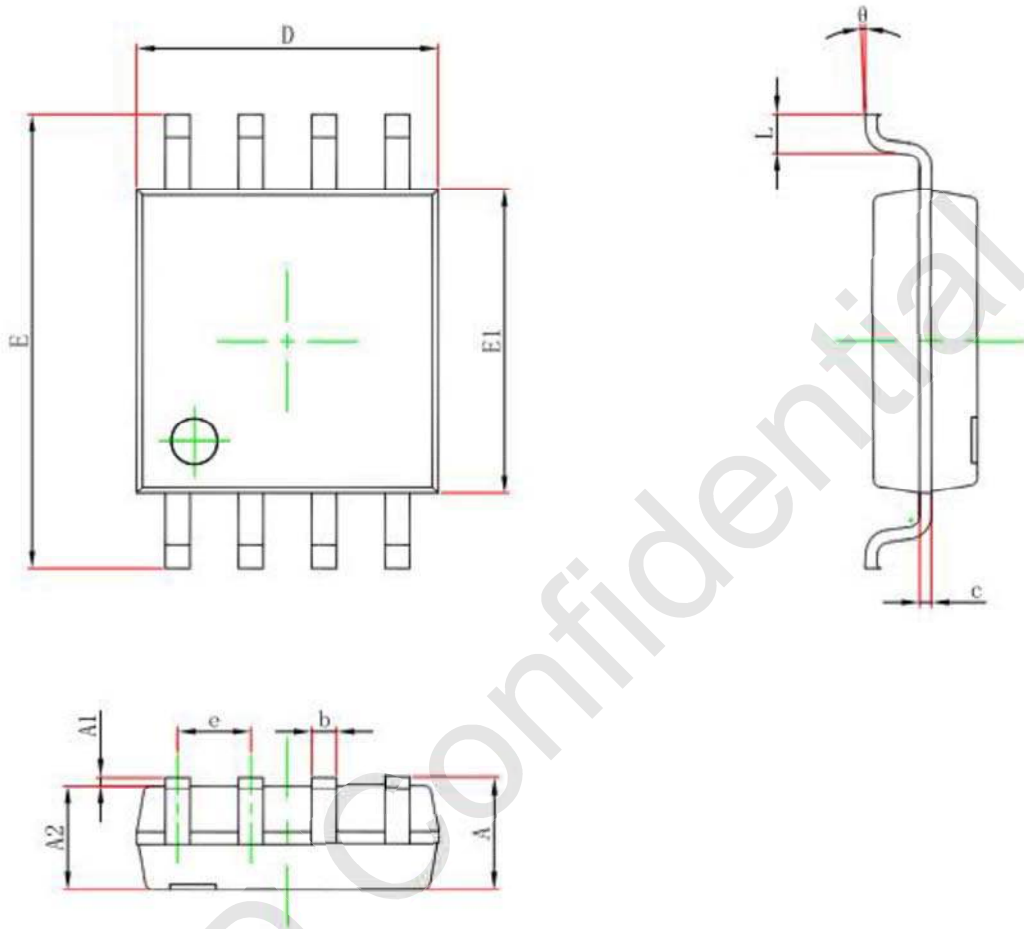
**10.PACKAGE INFORMATION**

**10.1.Package SOP8 150MIL**



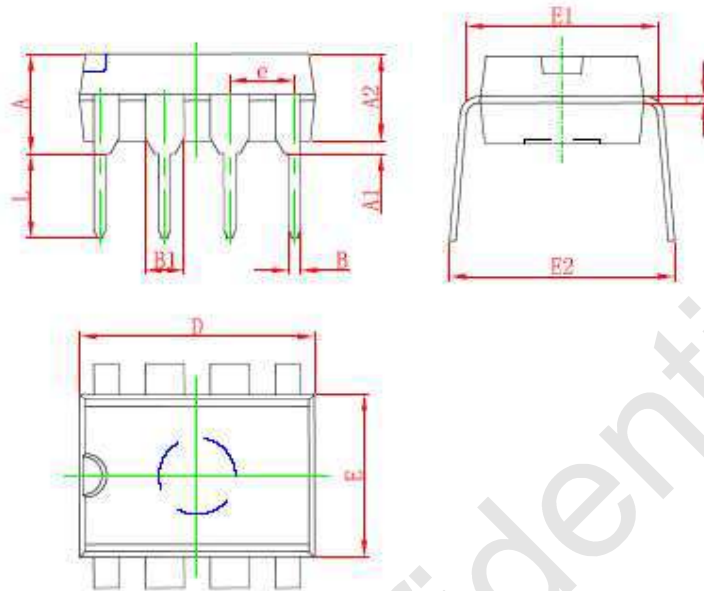
Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	1.350	1.750	0.053	0.069
A1	0.100	0.250	0.004	0.010
A2	1.350	1.550	0.053	0.061
b	0.330	0.510	0.013	0.020
c	0.170	0.250	0.006	0.010
D	4.700	5.100	0.185	0.200
E	3.800	4.000	0.150	0.157
E1	5.800	6.200	0.228	0.244
e	1.270 (BSC)		0.050 (BSC)	
L	0.400	1.270	0.016	0.050
θ	0°	8°	0°	8°

10.2. Package SOP8 208MIL



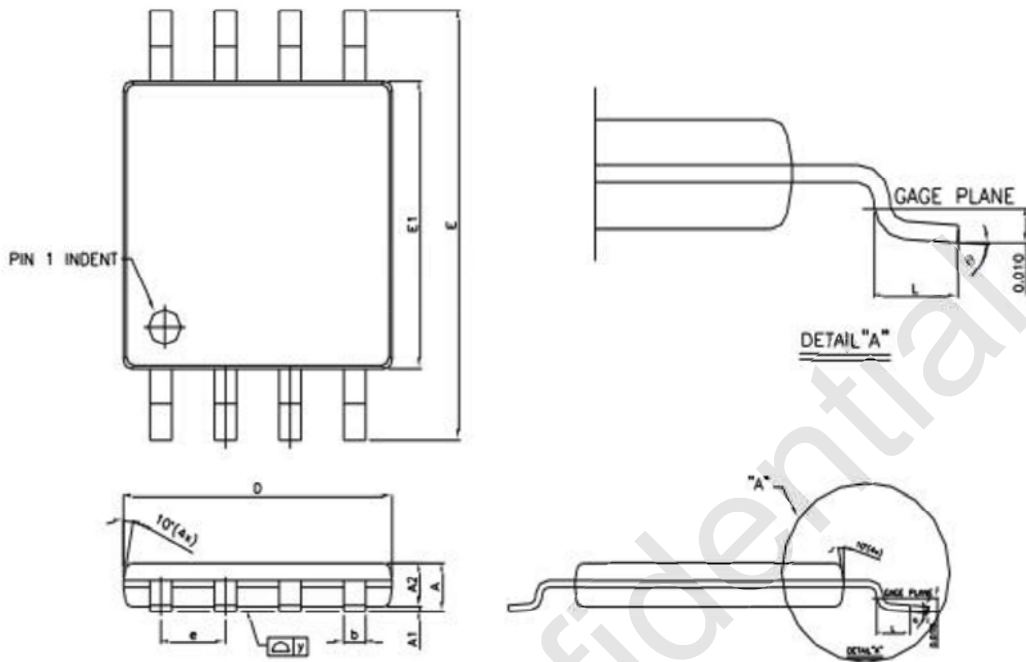
Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	--	2.150	--	0.085
A1	0.050	0.250	0.002	0.010
A2	1.700	1.900	0.067	0.075
b	0.350	0.500	0.014	0.020
c	0.100	0.250	0.004	0.010
D	5.130	5.330	0.202	0.210
E	7.700	8.100	0.303	0.319
E1	5.180	5.380	0.204	0.212
e	1.270 (BSC)		0.050 (BSC)	
L	0.500	0.850	0.020	0.033
θ	0°	8°	0°	8°

**10.3. Package DIP8 300MIL**



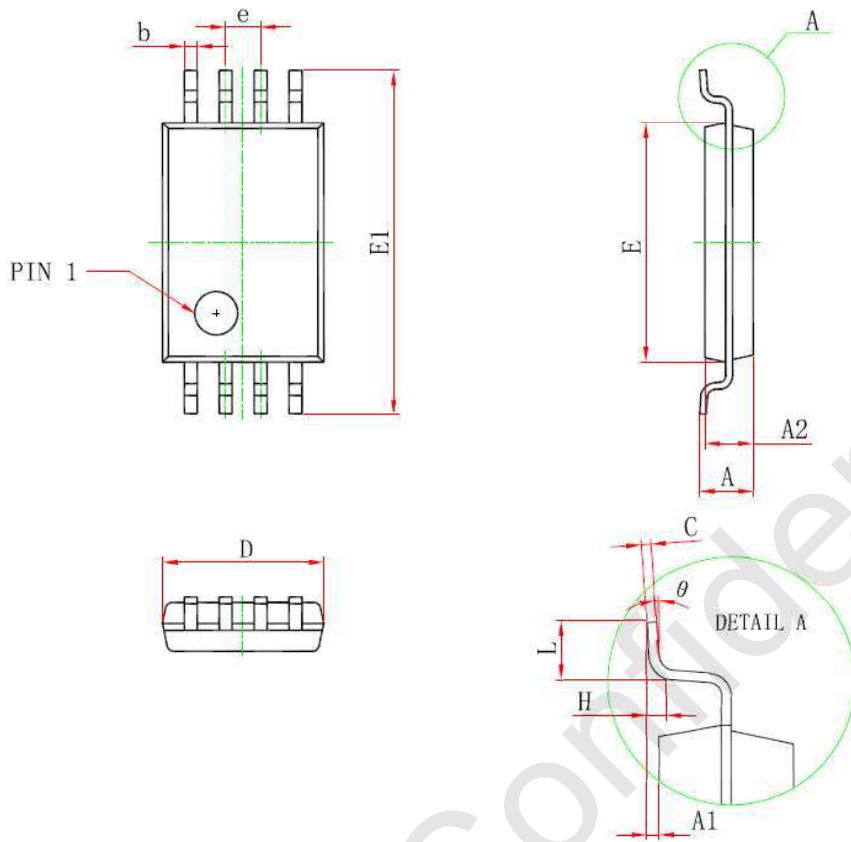
Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	3.710	4.310	0.146	0.170
A1	0.510		0.020	
A2	3.200	3.600	0.126	0.142
B	0.380	0.570	0.015	0.022
B1	1.524 (BSC)		0.060 (BSC)	
C	0.204	0.360	0.008	0.014
D	9.000	9.400	0.354	0.370
E	6.200	6.600	0.244	0.260
E1	7.320	7.920	0.288	0.312
e	2.540 (BSC)		0.100 (BSC)	
L	3.000	3.600	0.118	0.142
E2	8.400	9.000	0.331	0.354

10.4. Package VSOP8 208MIL



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	--	1.000	--	0.039
A1	0.050	0.150	0.002	0.006
A2	0.750	0.850	0.030	0.033
b	0.350	0.480	0.014	0.019
c	0.127 (REF)		0.005 (REF)	
D	5.180	5.380	0.204	0.212
E	7.700	8.100	0.303	0.319
E1	5.180	5.380	0.204	0.212
e	--	--	--	--
L	0.500	0.800	0.020	0.031
y	--	0.100	--	0.004
$\theta$	0°	8°	0°	8°

10.5. Package TSSOP8 173MIL



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
D	2.900	3.100	0.114	0.122
E	4.300	4.500	0.169	0.177
b	0.190	0.300	0.007	0.012
c	0.090	0.200	0.004	0.008
E1	6.250	6.550	0.246	0.258
A		1.200		0.047
A2	0.800	1.000	0.031	0.039
A1	0.050	0.150	0.002	0.006
e	0.65 (BSC)		0.026 (BSC)	
L	0.500	0.700	0.020	0.028
H	0.25 (TYP)		0.01 (TYP)	
θ	1°	7°	1°	7°

**11.REVISION HISTORY**

Version No	Description	Date
Preliminary 1.0		7.14.2014
Preliminary 1.1	Modify AC CHARACTERISTICS: $f_{C2}$ 80MHz max change to 40MHz max Modify AC CHARACTERISTICS: $t_w$ 10ms typ change to 100ms typ, $t_w$ 20ms max change to 200ms max Modify AC CHARACTERISTICS: $t_{SE}$ 80ms typ change to 120ms typ Add Package TSSOP8 173mil	10.29.2014



**Fremont Micro Devices (SZ) Limited**

#5-8, 10/F, Changhong Building, Ke-Ji Nan 12 Road, Nanshan District, Shenzhen, Guangdong 518057

Tel: (86 755) 86117811

Fax: (86 755) 86117810

**Fremont Micro Devices (Hong Kong) Limited**

#16, 16/F, Blk B, Veristrong Industrial Centre, 34-36 Au Pui Wan Street, Fotan, Shatin, Hong Kong

Tel: (852) 27811186

Fax: (852) 27811144

**Fremont Micro Devices (USA), Inc.**

42982 Osgood Road Fremont, CA 94539

Tel: (1-510) 668-1321

Fax: (1-510) 226-9918

Web Site: <http://www.fremontmicro.com/>

\* Information furnished is believed to be accurate and reliable. However, Fremont Micro Devices, Incorporated (BVI) assumes no responsibility for the consequences of use of such information or for any infringement of patents of other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent rights of Fremont Micro Devices, Incorporated (BVI). Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Fremont Micro Devices, Incorporated (BVI) products are not authorized for use as critical components in life support devices or systems without express written approval of Fremont Micro Devices, Incorporated (BVI). The FMD logo is a registered trademark of Fremont Micro Devices, Incorporated (BVI). All other names are the property of their respective own.