

P25D40SH

Ultra Low Power, 4M-bit Serial Multi I/O Flash Memory Datasheet

Apr. 02. 2021

Performance Highlight

- ◆ *Wide Supply Range from 2.3 to 3.6V for Read, Erase and Program*
- ◆ *Ultra Low Power consumption for Read, Erase and Program*
- ◆ *X1, X2 Multi I/O Support*
- ◆ *High reliability with 100K cycling and 10 Year-retention*



Puya Semiconductor (Shanghai) Co., Ltd

1 Overview

General

- **Single 2.3V to 3.60V supply**
- **Industrial Temperature Range 0C to 70C**
- **Serial Peripheral Interface (SPI) Compatible: Mode 0 and Mode 3**
- **Single, Dual IO mode**
 - 4M x 1 bit
 - 2M x 2 bits
- **Flexible Architecture for Code and Data Storage**
 - Uniform 256-byte Page Program
 - Uniform 256-byte Page Erase
 - Uniform 4K-byte Sector Erase
 - Uniform 32K/64K-byte Block Erase
 - Full Chip Erase
- **Hardware Controlled Locking of Protected Sectors by WP Pin**
- **One Time Programmable (OTP) Security Register**
 - 3*512-Byte Security Registers With OTP Lock
- **128 bit unique ID for each device**
- **Fast Program and Erase Speed**
 - 2ms Single/Dual Page(s) program time
 - 16ms Page erase time
 - 16ms 4K-byte sector erase time
 - 16ms 32K/64K-byte block erase time
- **JEDEC Standard Manufacturer and Device ID Read Methodology**
- **Ultra-Low Power Consumption**
 - 0.6uA Deep Power Down current
 - 10.0uA Standby current
 - 2.0mA Active Read current at 33MHz
 - 2.5mA Active Program or Erase current
- **High Reliability**
 - 100,000 Program / Erase Cycles
 - 10-year Data Retention
- **Industry Standard Green Package Options**
 - 8-pin SOP (150mil)
 - 8-land USON (3x2x0.55mm)
 - KGD for SiP

2 Description

The P25D40SH is a serial interface Flash memory device designed for use in a wide variety of high-volume consumer based applications in which program code is shadowed from Flash memory into embedded or external RAM for execution. The flexible erase architecture of the device, with its page erase granularity it is ideal for data storage as well, eliminating the need for additional data storage devices.

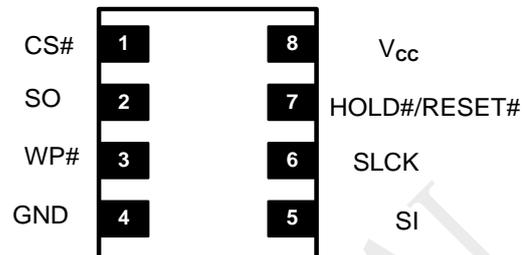
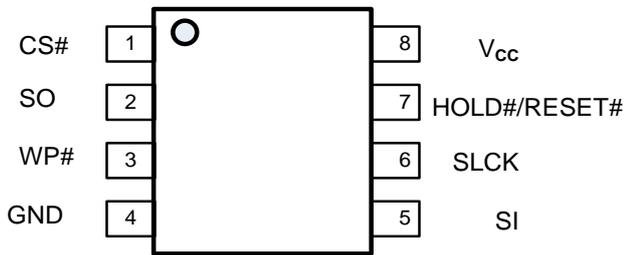
The erase block sizes of the device have been optimized to meet the needs of today's code and data storage applications. By optimizing the size of the erase blocks, the memory space can be used much more efficiently. Because certain code modules and data storage segments must reside by themselves in their own erase regions, the wasted and unused memory space that occurs with large sectored and large block erase Flash memory devices can be greatly reduced. This increased memory space efficiency allows additional code routines and data storage segments to be added while still maintaining the same overall device density.

The device also contains an additional 3*512-byte security registers with OTP lock (One-Time Programmable), can be used for purposes such as unique device serialization, system-level Electronic Serial Number (ESN) storage, locked key storage, etc.

Specifically designed for use in many different systems, the device supports read, program, and erase operations with a wide supply voltage range of 2.3V to 3.6V. No separate voltage is required for programming and erasing.

3 Pin Definition

3.1 Pin Configurations



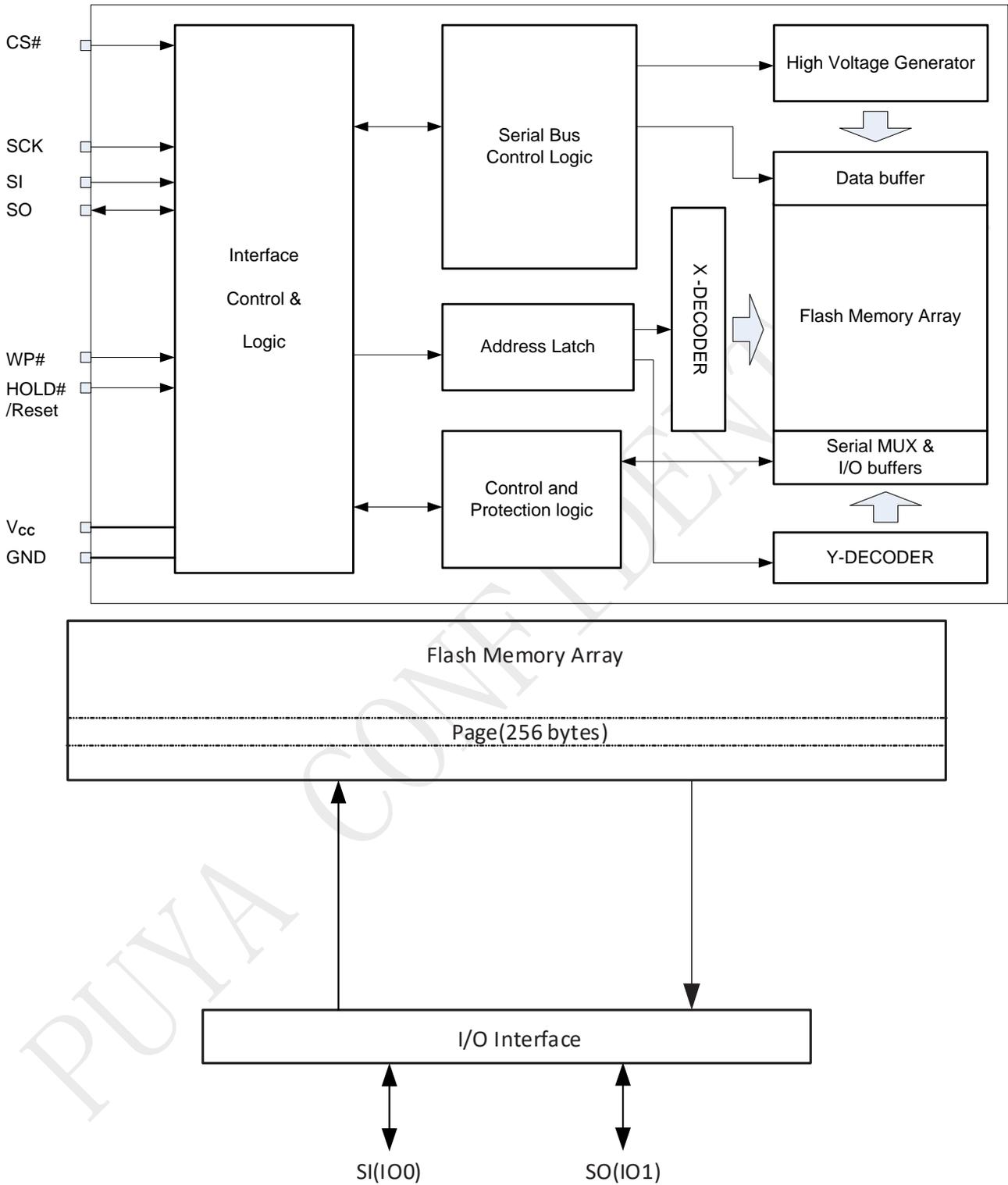
3.2 Pin Descriptions

| No. | Symbol | Extension | Remarks |
|-----|--------------|-----------|--|
| 1 | CS# | | Chip select |
| 2 | SO | SIO1 | Serial data output for 1 x I/O Serial data input and output for 2 x I/O read mode |
| 3 | WP# | - | Write protection active low |
| 4 | GND | - | Ground of the device |
| 5 | SI | SIO0 | Serial data input for 1x I/O Serial data input and output for 2 x I/O read mode |
| 6 | SCLK | - | Serial interface clock input |
| 7 | HOLD#/RESET# | - | Hardware Reset Pin Active low or to pause the device without deselecting the device |
| 8 | Vcc | - | Power supply of the device |

Notes:

1. SIO0 and SIO1 are used for Standard and Dual SPI instructions
2. WP# & HOLD# (or RESET#) functions are only available for Standard/Dual SPI.

4 Block Diagram



5 Electrical Specifications

5.1 Absolute Maximum Ratings

- Storage Temperature-65°C to +150°C
- Operation Temperature0°C to +70°C
- Maximum Operation Voltage..... 4.0V
- Voltage on Any Pin with respect to Ground.-0.6V to + 4.1V
- DC Output Current5.0 mA

NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Table 5-1 Pin Capacitance [1]

| Symbol | Parameter | Max. | Units | Test Condition |
|------------------|--------------------|------|-------|-----------------------|
| C _{OUT} | Output Capacitance | 8 | pF | V _{OUT} =GND |
| C _{IN} | Input Capacitance | 6 | pF | V _{IN} =GND |

Note:

1. Test Conditions: T_A = 25°C, F = 1MHz, V_{CC} = 3.0V.

Figure 5-1 Maximum Overshoot Waveform



Figure 5-2 Input Test Waveforms and Measurement Level

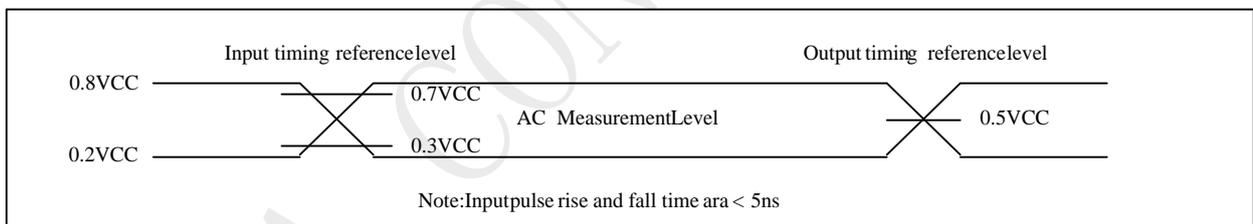
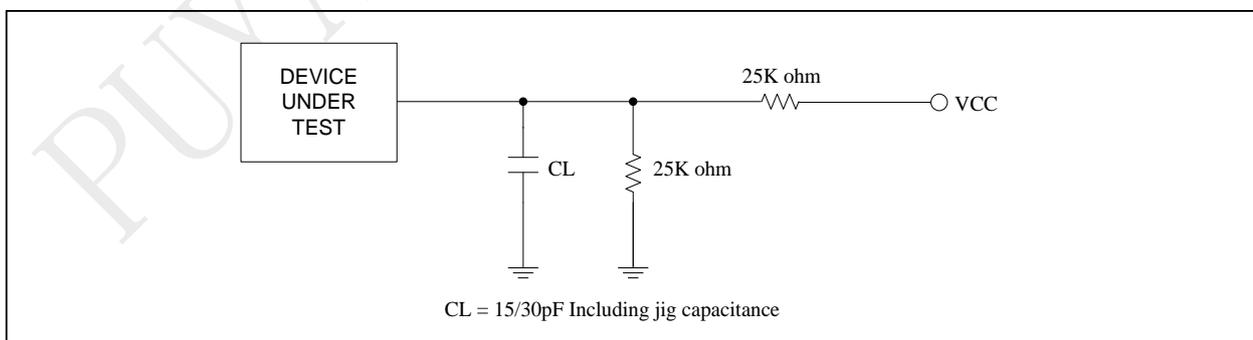


Figure 5-3 Output Loading



5.2 DC Characteristics

Table 5-2 DC parameters(Ta=0°C to +70°C)

| Sym. | Parameter | Conditions | 2.3V to 3.6V | | | Units |
|------------------|--------------------------------|---|--------------|------|--------|-------|
| | | | Min. | Typ. | Max. | |
| I _{DPD} | Deep power down current | CS#=Vcc, all other inputs at 0V or Vcc | | 0.6 | 1.5 | uA |
| I _{SB} | Standby current | CS#, HOLD#, WP#=VIH all inputs at CMOS levels | | 10.0 | | uA |
| I _{CC1} | Read current(STR) (1, 2 IO) | f=33MHz(03h); IOU=0mA | | 2.0 | 2.5 | mA |
| | | f=85MHz; IOU=0mA | | 5 | 8 | mA |
| I _{CC2} | Program current | CS#=Vcc | | 2.5 | 4.0 | mA |
| I _{CC3} | Erase current | CS#=Vcc | | 2.5 | 4.0 | mA |
| I _{LI} | Input load current | All inputs at CMOS level | | | 1.0 | uA |
| I _{LO} | Output leakage | All inputs at CMOS level | | | 1.0 | uA |
| V _{IL} | Input low voltage | | | | 0.3Vcc | V |
| V _{IH} | Input high voltage | | 0.7Vcc | | | V |
| V _{OL} | Output low voltage | IOL=100uA | | | 0.2 | V |
| V _{OH} | Output high voltage | IOH=-100uA | Vcc-0.2 | | | V |

Note

1. Typical values measured at 3.0V @ 25°C for the 2.3V to 3.6V range.

5.3 AC Characteristics

Table 5-3-1 AC parameters(Ta=0°C to +70°C)

| Symbol | Alt. | Parameter | 2.3V~3.6V | | | Unit |
|----------|------|--|-----------|-----|-----|------|
| | | | min | typ | max | |
| fSCLK | fC | Clock Frequency for the following instructions: FREAD, RDSFDP, PP, SE, BE32K, BE, CE, DP, RES, WREN, WRDI, RDID, RDSR, WRSR(7) | | | 104 | MHz |
| fRSCLK | fR | Clock Frequency for READ instructions | | | 55 | MHz |
| fTSCLK | fT | Clock Frequency for 2READ,DREAD instructions | | | 104 | MHz |
| tCH(1) | tCLH | Clock High Time | 4.5 | | | ns |
| tCL(1) | tCLL | Clock Low Time (fSCLK) 45% x (1fSCLK) | 4.5 | | | ns |
| tCLCH(7) | | Clock Rise Time (peak to peak) | 0.1 | | | v/ns |
| tCHCL(7) | | Clock Fall Time (peak to peak) | 0.1 | | | v/ns |
| tSLCH | tCSS | CS# Active Setup Time (relative to SCLK) | 5 | | | ns |
| tCHSL | | CS# Not Active Hold Time (relative to SCLK) | 5 | | | ns |
| tDVCH | tDSU | Data In Setup Time | 2 | | | ns |
| tCHDX | tDH | Data In Hold Time | 3 | | | ns |
| tCHSH | | CS# Active Hold Time (relative to SCLK) | 5 | | | ns |
| tSHCH | | CS# Not Active Setup Time (relative to SCLK) | 5 | | | ns |
| tSHSL | tCSH | CS# Deselect Time From Read to next Read | 20 | | | ns |
| | | CS# Deselect Time From Write,Erase,Program to Read Status Register | 30 | | | ns |
| tSHQZ(7) | tDIS | Output Disable Time | | | 6 | ns |
| tCLQV | tV | Clock Low to Output Valid Loading 30pF | | | 7 | ns |
| | | Clock Low to Output Valid Loading 15pF | | | 6 | ns |
| tCLQX | tHO | Output Hold Time | 0 | | | ns |
| tHLCH | | HOLD# Active Setup Time (relative to SCLK) | 5 | | | ns |
| tCHHH | | HOLD# Active Hold Time (relative to SCLK) | 5 | | | ns |
| tHHCH | | HOLD# Not Active Setup Time (relative to SCLK) | 5 | | | ns |
| tCHHL | | HOLD# Not Active Hold Time (relative to SCLK) | 5 | | | ns |
| tHHQX | tLZ | HOLD# to Output Low-Z | | | 6 | ns |
| tHLQZ | tHZ | HOLD# to Output High-Z | | | 6 | ns |
| tWHS(3) | | Write Protect Setup Time | 20 | | | ns |
| tSHWL(3) | | Write Protect Hold Time | 100 | | | ns |
| tDP | | CS# High to Deep Power-down Mode | | | 3 | us |
| tRES1 | | CS# High To Standby Mode Without Electronic Signature Read | | | 8 | us |
| tRES2 | | CS# High To Standby Mode With Electronic Signature Read | | | 8 | us |
| tW | | Write Status Register Cycle Time | | 8 | 12 | ms |
| tReady | | Reset recovery time(for erase/program operation except WRSR) | 30 | | | us |
| | | Reset recovery time(for WRSR operation) | 12 | 8 | | ms |

Table 5-3-2 SPI Read Command Performance Comparison(MHz)

| Read command | Dummy Cycles (VCC=2.3V~3.6V) | | | |
|--------------|------------------------------|---|-----|----|
| | 4 | 6 | 8 | 10 |
| FREAD | - | - | 120 | - |
| DREAD | - | - | 120 | - |
| 2READ | 104(default) | - | 120 | - |

5.4 AC Characteristics for Program and Erase

Table 5-4 AC parameters for program and erase(Ta=0°C to +70°C)

| Sym. | Parameter | 2.3V to 3.6V | | | Units |
|------------------|-------------------------------------|--------------|------|------|-------|
| | | Min. | Typ. | Max. | |
| t _{PP} | Page program time (up to 256 bytes) | | 2 | 3 | ms |
| t _{PE} | Page erase time | | 16 | 30 | ms |
| t _{SE} | Sector erase time | | 16 | 30 | ms |
| t _{BE1} | Block erase time for 32K bytes | | 16 | 30 | ms |
| t _{BE2} | Block erase time for 64K bytes | | 16 | 30 | ms |
| t _{CE} | Chip erase time | | 16 | 30 | ms |

Note

1. t_{CH} + t_{CL} must be greater than or equal to 1/ Frequency.
2. Typical values given for TA=25°C. Not 100% tested.
3. Only applicable as a constraint for a WRSR instruction.
4. The value guaranteed by characterization, not 100% tested in production.

Figure 5-4 Serial Input Timing

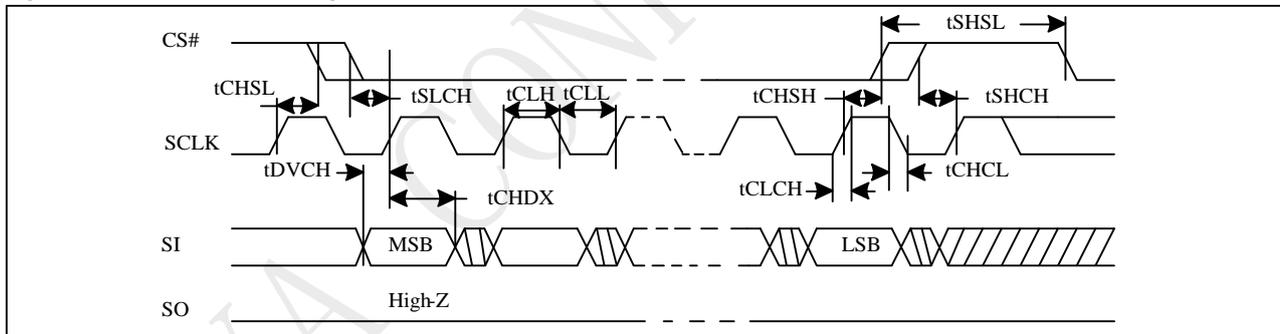


Figure 5-5 Output Timing

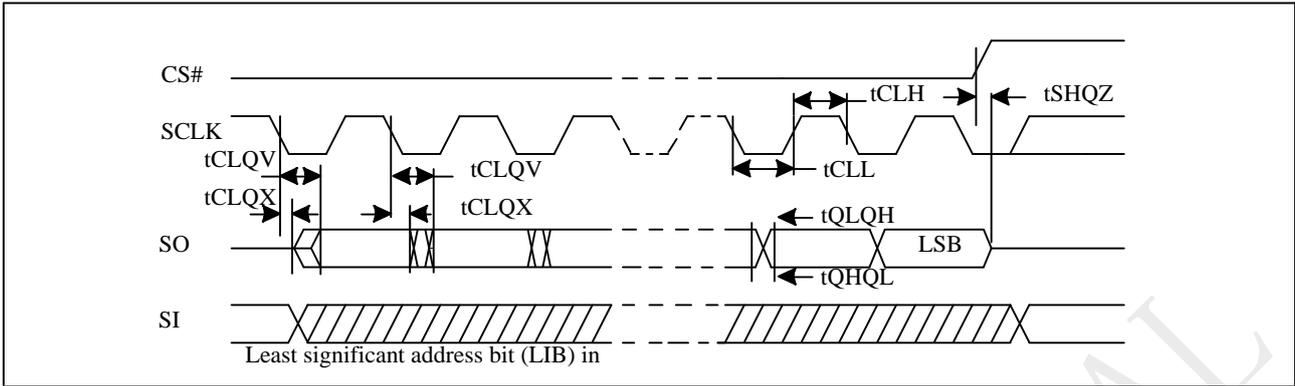


Figure 5-6 Hold Timing

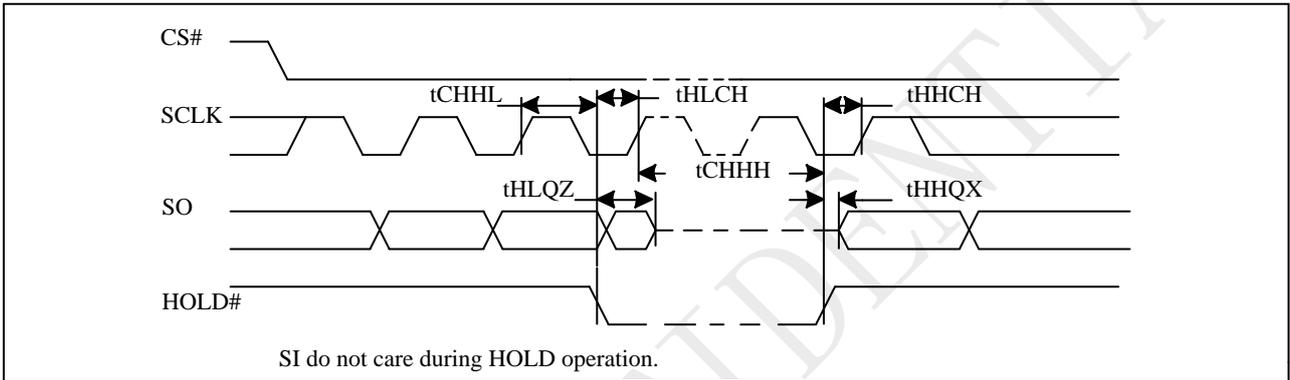
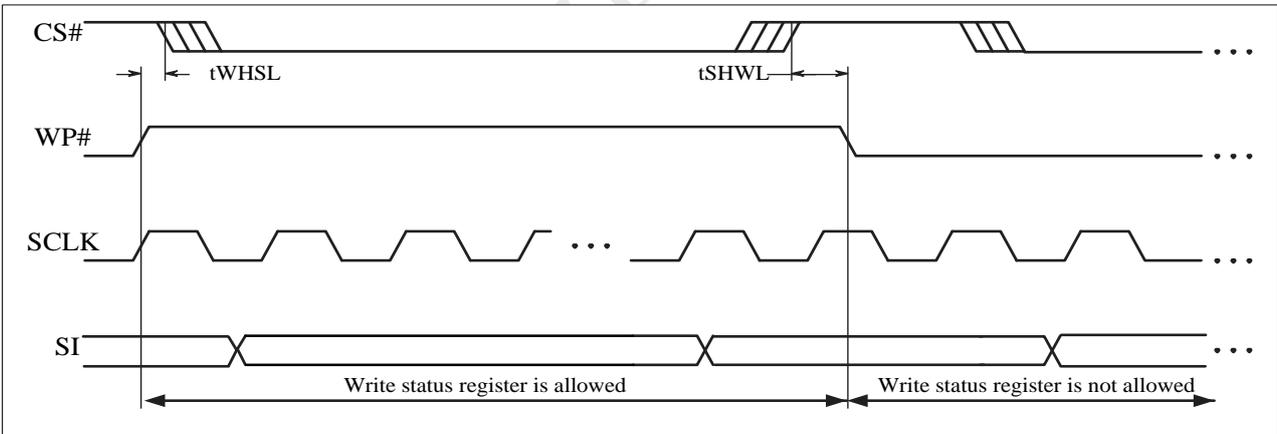


Figure 5-7 WP Timing



5.5 Operation Conditions

At Device Power-Up and Power-Down

AC timing illustrated in "Figure AC Timing at Device Power-Up" and "Figure Power-Down Sequence" are for the supply voltages and the control signals at device power-up and power-down. If the timing in the figures is ignored, the device will not operate correctly.

During power-up and power-down, CS# needs to follow the voltage applied on VCC to keep the device not to be selected. The CS# can be driven low when VCC reach $V_{cc(min)}$ and wait a period of t_{VSL} .

Figure 5-8 AC Timing at Device Power-Up

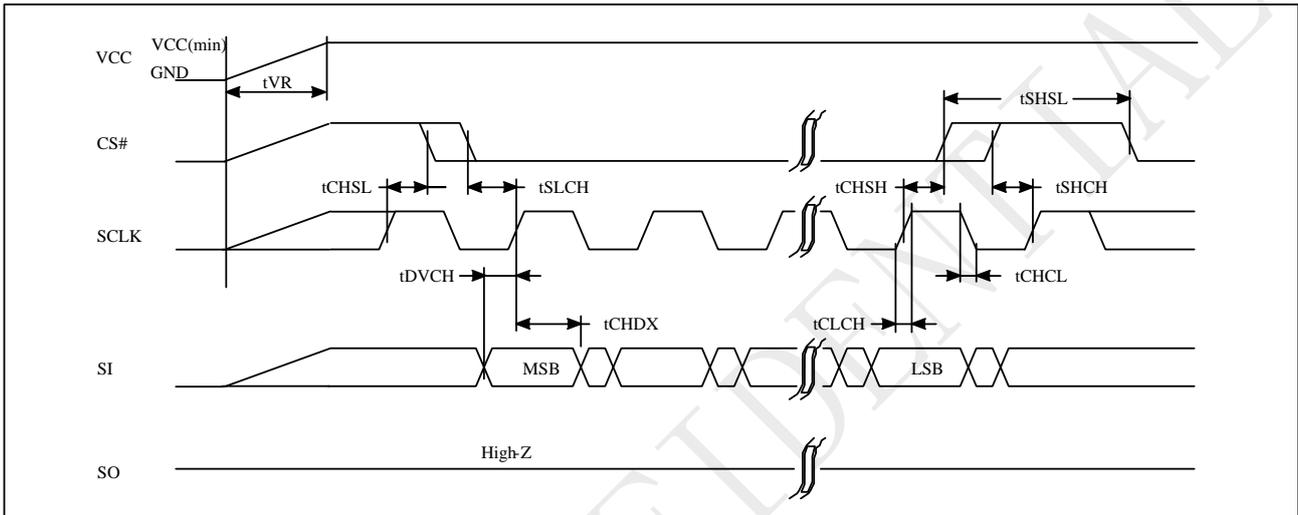
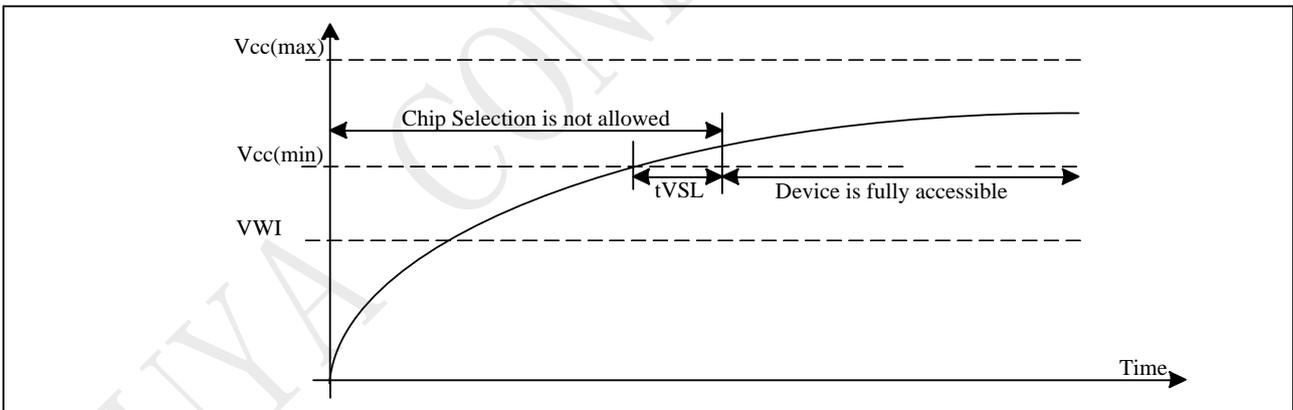


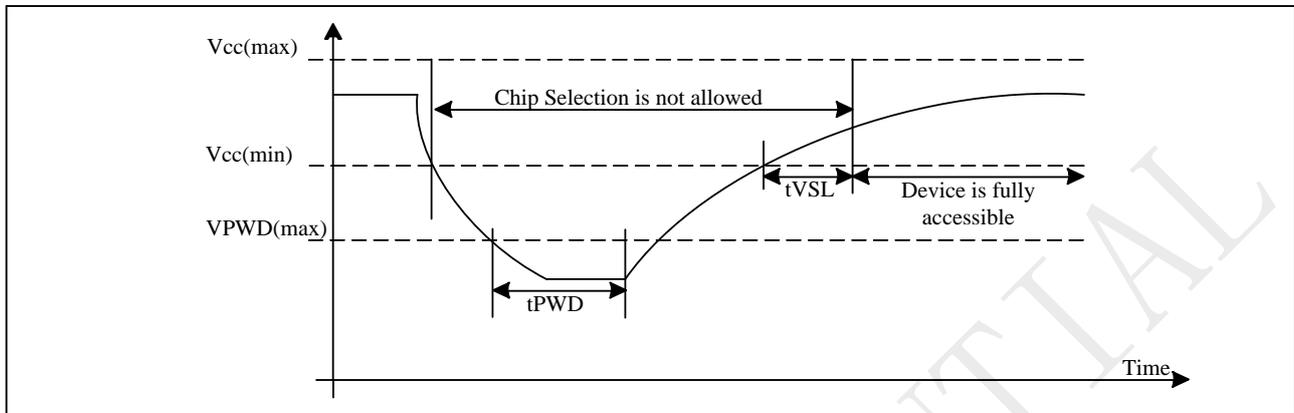
Figure 5-9 Power-up Timing



Power Up/Down and Voltage Drop

For Power-down to Power-up operation, the VCC of flash device must below VPWD for at least tPWD timing. Please check the table below for more detail.

Figure 5-10 Power down-up Timing



| Symbol | Parameter | min | max | unit |
|--------|---|------|--------|------|
| VPWD | VCC voltage needed to below VPWD for ensuring initialization will occur | | 1 | V |
| tPWD | The minimum duration for ensuring initialization will occur | 300 | | us |
| tVSL | VCC(min.) to device operation | 300 | | us |
| tVR | VCC Rise Time | 1 | 500000 | us/V |
| VWI | Write Inhibit Voltage | 1.45 | 1.55 | V |

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).

6 Data Protection

During power transition, there may be some false system level signals which result in inadvertent erasure or programming. The device is designed to protect itself from these accidental write cycles.

The state machine will be reset as standby mode automatically during power up. In addition, the control register architecture of the device constrains that the memory contents can only be changed after specific command sequences have completed successfully.

In the following, there are several features to protect the system from the accidental write cycles during VCC power-up and power-down or from system noise.

- Power-on reset: to avoid sudden power switch by system power supply transition, the power-on reset may protect the Flash.
- Valid command length checking: The command length will be checked whether it is at byte base and completed on byte boundary.
- Write Enable (WREN) command: WREN command is required to set the Write Enable Latch bit (WEL) before issuing other commands to change data.
- Software Protection Mode: The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits define the section of the memory array that can be read but not change.
- Hardware Protection Mode: WP# going low to protected the BP0~BP4bits and SRP0~1bits
- Deep Power-Down Mode: By entering deep power down mode, the flash device is under protected from writing all commands except the Release from Deep Power-Down Mode command.

Protected Area Sizes

Table 6-1. P25D40SH Protected Area Sizes (WPS=0,CMP bit = 0)

| Status bit | | | | | Memory Content | | | |
|------------|-----|-----|-----|-----|----------------|------------------|---------|-------------|
| BP4 | BP3 | BP2 | BP1 | BP0 | Blocks | Addresses | Density | Portion |
| x | x | 0 | 0 | 0 | NONE | NONE | NONE | NONE |
| 0 | 0 | 0 | 0 | 1 | 7 | 070000H-07FFFFH | 64KB | Upper 1/8 |
| 0 | 0 | 0 | 1 | 0 | 6 and 7 | 060000H-07FFFFH | 128KB | Upper 1/4 |
| 0 | 0 | 0 | 1 | 1 | 4 to 7 | 040000H-07FFFFH | 256KB | Upper 1/2 |
| 0 | 1 | 0 | 0 | 1 | 0 | 000000H-00FFFFH | 64KB | Lower 1/8 |
| 0 | 1 | 0 | 1 | 0 | 0 and 1 | 000000H-01FFFFH | 128KB | Lower 1/4 |
| 0 | 1 | 0 | 1 | 1 | 0 to 3 | 000000H-03FFFFH | 256KB | Lower 1/2 |
| 0 | x | 1 | x | x | 0 to 7 | 000000H-07FFFFH | 512KB | ALL |
| 1 | 0 | 0 | 0 | 1 | 7 | 07F000H-07FFFFH | 4KB | Upper 1/128 |
| 1 | 0 | 0 | 1 | 0 | 7 | 07E000H-07FFFFH | 8KB | Upper 1/64 |
| 1 | 0 | 0 | 1 | 1 | 7 | 07C000H-07FFFFH | 16KB | Upper 1/32 |
| 1 | 0 | 1 | 0 | x | 7 | 078000H- 07FFFFH | 32KB | Upper 1/16 |
| 1 | 0 | 1 | 1 | 0 | 7 | 078000H-07FFFFH | 32KB | Upper 1/16 |
| 1 | 1 | 0 | 0 | 1 | 0 | 000000H-000FFFH | 4KB | Lower 1/128 |
| 1 | 1 | 0 | 1 | 0 | 0 | 000000H-001FFFH | 8KB | Lower 1/64 |
| 1 | 1 | 0 | 1 | 1 | 0 | 000000H-003FFFH | 16KB | Lower 1/32 |
| 1 | 1 | 1 | 0 | x | 0 | 000000H-007FFFH | 32KB | Lower 1/16 |
| 1 | 1 | 1 | 1 | 0 | 0 | 000000H-007FFFH | 32KB | Lower 1/16 |
| 1 | x | 1 | 1 | 1 | 0 to 7 | 000000H-07FFFFH | 512KB | ALL |

Table 6-2. P25D40SH Protected Area Sizes (WPS=0,CMP bit = 1)

| Status bit | | | | | Memory Content | | | |
|------------|-----|-----|-----|-----|----------------|------------------|---------|---------------|
| BP4 | BP3 | BP2 | BP1 | BP0 | Blocks | Addresses | Density | Portion |
| x | x | 0 | 0 | 0 | 0 to 7 | 000000H-07FFFFH | 512KB | ALL |
| 0 | 0 | 0 | 0 | 1 | 0 to 6 | 000000H-06FFFFH | 448KB | Lower 7/8 |
| 0 | 0 | 0 | 1 | 0 | 0 to 5 | 000000H-05FFFFH | 384KB | Lower 3/4 |
| 0 | 0 | 0 | 1 | 1 | 0 to 3 | 000000H-03FFFFH | 256KB | Lower 1/2 |
| 0 | 1 | 0 | 0 | 1 | 1 to 7 | 010000H-07FFFFH | 448KB | Upper 7/8 |
| 0 | 1 | 0 | 1 | 0 | 2 to 7 | 020000H-07FFFFH | 384KB | Upper 3/4 |
| 0 | 1 | 0 | 1 | 1 | 4 to 7 | 040000H-07FFFFH | 256KB | Upper 1/2 |
| 0 | x | 1 | x | x | NONE | NONE | NONE | NONE |
| 1 | 0 | 0 | 0 | 1 | 0 to 7 | 000000H-07EFFFFH | 508KB | Lower 127/128 |
| 1 | 0 | 0 | 1 | 0 | 0 to 7 | 000000H-07DFFFFH | 504KB | Lower 63/64 |
| 1 | 0 | 0 | 1 | 1 | 0 to 7 | 000000H-07BFFFFH | 496KB | Lower 31/32 |
| 1 | 0 | 1 | 0 | x | 0 to 7 | 000000H-077FFFFH | 480KB | Lower 15/16 |
| 1 | 0 | 1 | 1 | 0 | 0 to 7 | 000000H-077FFFFH | 480KB | Lower 15/16 |
| 1 | 1 | 0 | 0 | 1 | 0 to 7 | 001000-07FFFFH | 508KB | Upper 127/128 |
| 1 | 1 | 0 | 1 | 0 | 0 to 7 | 002000-07FFFFH | 504KB | Upper 63/64 |
| 1 | 1 | 0 | 1 | 1 | 0 to 7 | 004000-07FFFFH | 496KB | Upper 31/32 |
| 1 | 1 | 1 | 0 | x | 0 to 7 | 008000-07FFFFH | 480KB | Upper 15/16 |
| 1 | 1 | 1 | 1 | 0 | 0 to 7 | 008000-07FFFFH | 480KB | Upper 15/16 |
| 1 | x | 1 | 1 | 1 | NONE | NONE | NONE | NONE |

Note:

1. X=don't care
2. If any erase or program command specifies a memory that contains protected data portion, this command will be ignored.

7 Memory Address Mapping

The memory array can be erased in three levels of granularity including a full chip erase. The size of the erase blocks is optimized for both code and data storage applications, allowing both code and data segments to reside in their own erase regions.

| Each device has | Each block has | Each sector has | Each page has | |
|-----------------|----------------|-----------------|---------------|---------|
| 512 K | 64/32K | 4K | 256 | bytes |
| 2048 | 256/128 | 16 | - | pages |
| 128 | 16/8 | - | - | sectors |
| 8 | - | - | - | blocks |

P25D40SH Memory Organization

| Block64K | Block32K | Sector | Address Range | |
|----------|----------|--------|---------------|---------|
| 7 | 15-14 | 127 | 07F000H | 07FFFFH |
| | | | | |
| 6 | 13-12 | 112 | 070000H | 07FFFFH |
| | | | | |
| | | 96 | 06F000H | 06FFFFH |
| | | | | |
| | | | | |
| | | | | |
| 2 | 5-4 | 47 | 02F000H | 02FFFFH |
| | | | | |
| 1 | 3-2 | 32 | 020000H | 020FFFH |
| | | | | |
| 0 | 1-0 | 31 | 01F000H | 01FFFFH |
| | | | | |
| | | 16 | 010000H | 010FFFH |
| | | | | |
| | | 15 | 00F000H | 00FFFFH |
| | | | | |
| | | 0 | 000000H | 000FFFH |
| | | | | |

8 Device Operation

Before a command is issued, status register should be checked to ensure device is ready for the intended operation.

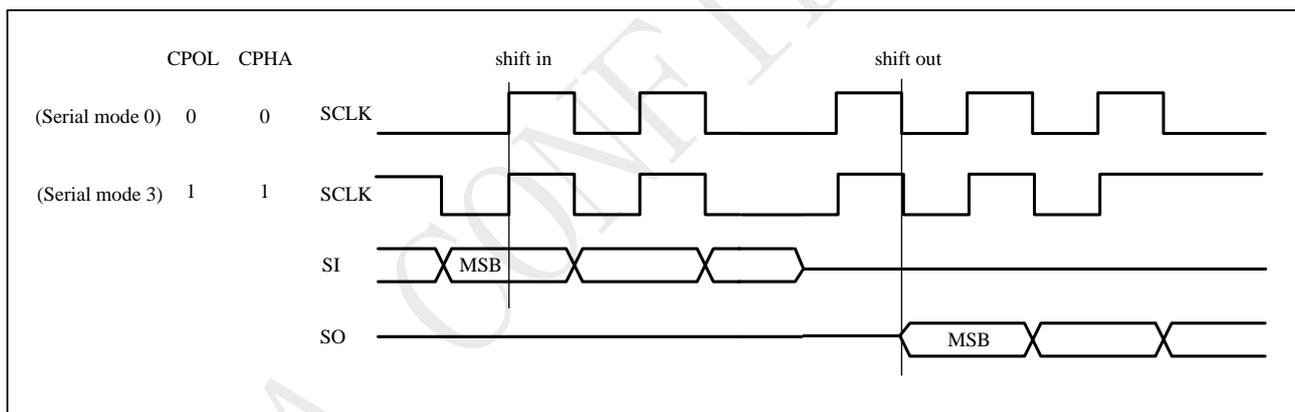
When incorrect command is inputted to this LSI, this LSI becomes standby mode and keeps the standby mode until next CS# falling edge. In standby mode, SO pin of this LSI should be High-Z. When correct command is inputted to this LSI, this LSI becomes active mode and keeps the active mode until next CS# rising edge.

Input data is latched on the rising edge of Serial Clock (SCLK) and data shifts out on the falling edge of SCLK. The difference of serial peripheral interface mode 0 and mode 3 is shown as Figure 8-1.

For the following instructions: RDID, RDSR, RDSR1, RDSCUR, READ, FREAD, DREAD, 2READ, RDSFDP, RES, REMS, DREMS, the shifted-in instruction sequence is followed by a data-out sequence. After any bit of data being shifted out, the CS# can be high. For the following instructions: WREN, WRDI, WRSR, PE, SE, BE32K, BE, CE, PP, DP, ERSCUR, PRSCUR, RSTEN, RST, the CS# must go high exactly at the byte boundary; otherwise, the instruction will be rejected and not executed.

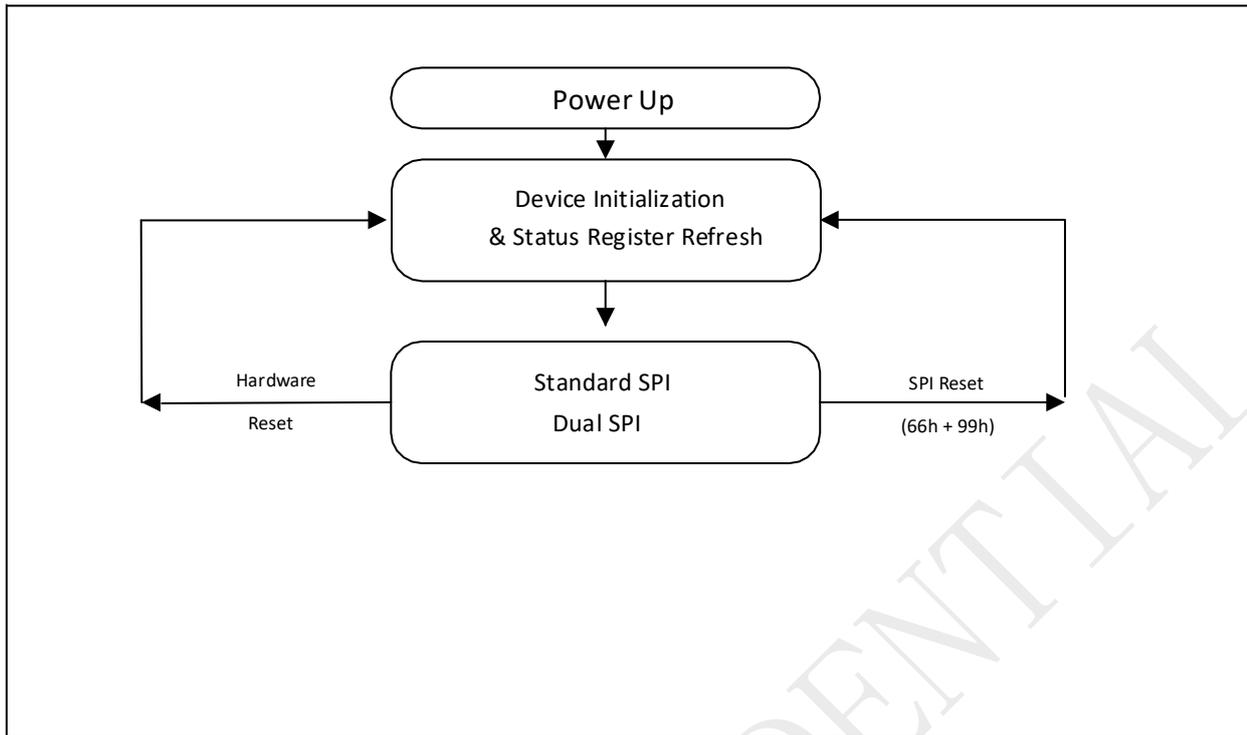
During the progress of Write Status Register, Program, Erase operation, to access the memory array is neglected and not affect the current operation of Write Status Register, Program, Erase.

Figure 8-1 Serial Peripheral Interface Modes Supported



Note:

CPOL indicates clock polarity of serial master, CPOL=1 for SCLK high while idle, CPOL=0 for SCLK low while not transmitting. CPHA indicates clock phase. The combination of CPOL bit and CPHA bit decides which serial mode is supported.



Standard SPI

The P25D40SH 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 P25D40SH supports Dual SPI operation when using the “Dual Output Fast Read” and “Dual I/O Fast Read”(3BHand 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.

Software Reset

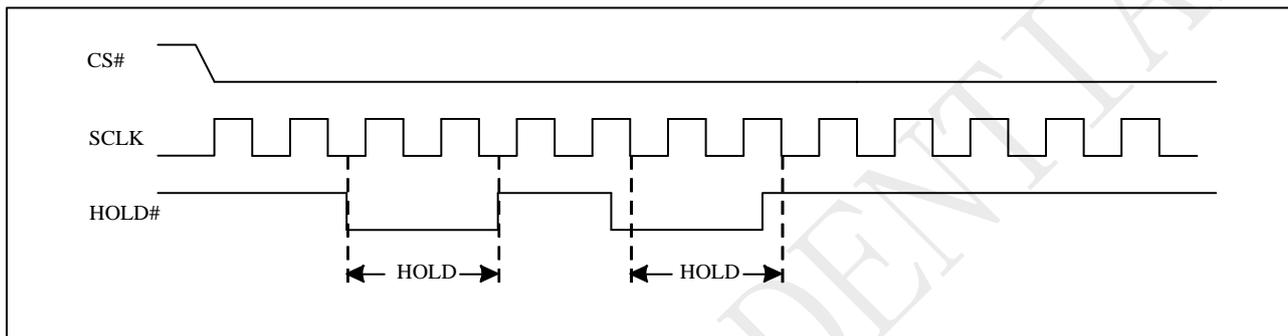
The P25D40SH can be reset to the initial power-on state by a software Reset sequence, either in SPI mode. This sequence must include two consecutive commands: Enable Reset (66h) & Reset (99h). If the command sequence is successfully accepted, the device will take approximately 30uS (tReady) to reset. No command will be accepted during the reset period.

9 Hold Feature

HOLD# pin signal goes low to hold any serial communications with the device. The HOLD feature will not stop the operation of write status register, programming, or erasing in progress.

The operation of HOLD requires Chip Select(CS#) keeping low and starts on falling edge of HOLD# pin signal while Serial Clock (SCLK) signal is being low (if Serial Clock signal is not being low, HOLD operation will not start until Serial Clock signal being low). The HOLD condition ends on the rising edge of HOLD# pin signal while Serial Clock(SCLK) signal is being low(if Serial Clock signal is not being low, HOLD operation will not end until Serial Clock being low).

Figure 9-1 Hold Condition Operation



During the HOLD operation, the Serial Data Output (SO) is high impedance when Hold# pin goes low and will keep high impedance until Hold# pin goes high. The Serial Data Input (SI) is don't care if both Serial Clock (SCLK) and Hold# pin goes low and will keep the state until SCLK goes low and Hold# pin goes high. If Chip Select (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 CS# must be at low.

10 Commands

10.1 Commands listing

Figure 10-1 Command set

| Commands | Abbr. | Code | ADR Bytes | DMY Bytes | Data Bytes | Function description |
|----------------------------|---------|---------|-----------|-----------|------------|--------------------------------------|
| Read | | | | | | |
| Read Array (fast) | FREAD | 0Bh | 3 | 1 | 1+ | n bytes read out until CS# goes high |
| Read Array (low power) | READ | 03h | 3 | 0 | 1+ | n bytes read out until CS# goes high |
| Read Dual Output | DREAD | 3Bh | 3 | 1 | 1+ | n bytes read out by Dual output |
| Read 2IO | 2READ | BBh | 3 | 1(2) | 1+ | n bytes read out by 2IO |
| Program and Erase | | | | | | |
| Page Erase | PE | 81h | 3 | 0 | 0 | erase selected page |
| Sector Erase (4K bytes) | SE | 20h | 3 | 0 | 0 | erase selected sector |
| Block Erase (32K bytes) | BE32 | 52h | 3 | 0 | 0 | erase selected 32K block |
| Block Erase (64K bytes) | BE | D8h | 3 | 0 | 0 | erase selected 64K block |
| Chip Erase | CE | 60h/C7h | 0 | 0 | 0 | erase whole chip |
| Page Program | PP | 02h | 3 | 0 | 1+ | program selected page |
| Protection | | | | | | |
| Write Enable | WREN | 06h | 0 | 0 | 0 | sets the write enable latch bit |
| Write Disable | WRDI | 04h | 0 | 0 | 0 | resets the write enable latch bit |
| Volatile SR Write Enable | VWREN | 50h | 0 | 0 | 0 | Write enable for volatile SR |
| Individual Block Lock | SBLK | 36h | 3 | 0 | 0 | Individual block lock |
| Individual Block Unlock | SBULK | 39h | 3 | 0 | 0 | Individual block unlock |
| Read Block Lock Status | RDBLOCK | 3Dh | 3 | 0 | 0 | Read individual block lock register |
| Global Block Lock | GBLK | 7Eh | 0 | 0 | 0 | Whole chip block protect |
| Global Block Unlock | GBULK | 98h | 0 | 0 | 0 | Whole chip block unprotect |
| Security | | | | | | |
| Erase Security Registers | ERSCUR | 44h | 3 | 0 | 0 | Erase security registers |
| Program Security Registers | PRSCUR | 42h | 3 | 0 | 1+ | Program security registers |
| Read Security Registers | RDSCUR | 48h | 3 | 1 | 1+ | Read value of security register |

Command set Cont'd

| Commands | Abbr. | Code | ADR Bytes | DMY Bytes | Data Bytes | Function |
|---|---------|------|-----------|-----------|------------|--|
| Status Register | | | | | | |
| Read Status Register | RDSR | 05h | 0 | 0 | 1 | read out status register |
| Read Status Register-1 | RDSR1 | 35h | 0 | 0 | 1 | Read out status register-1 |
| Read Configure Register | RDCR | 15h | 0 | 0 | 1 | Read out configure register |
| Write Status Register <i>note1</i> | WRSR | 01h | 0 | 0 | 1 | Write data to status registers |
| Write Status Register | WRSR | 01h | 0 | 0 | 2 | Write data to status registers |
| Write Status Register-1 <i>note2</i> | WRSR1 | 31h | 0 | 0 | 1 | Write data to status registers-1 |
| Write Configure Register | WRRCR | 11h | 0 | 0 | 1 | Write data to configuration register |
| Other Commands | | | | | | |
| Reset Enable | RSTEN | 66h | 0 | 0 | 0 | Enable reset |
| Reset | RST | 99h | 0 | 0 | 0 | Reset |
| Read Manufacturer/device ID | RDID | 9Fh | 0 | 0 | 1 to 3 | output JEDEC ID: 1-byte manufacturer ID & 2-byte device ID |
| Read Manufacture ID | REMS | 90h | 3 | | 1+ | Read manufacturer ID/device ID data |
| Dual Read Manufacture ID | DREMS | 92h | 3 | 1 | 1+ | Dual output read manufacture/device ID |
| Deep Power-down | DP | B9h | 0 | 0 | 0 | enters deep power-down mode |
| Release Deep Power-down / Read Electronic ID | RDP/RES | ABh | 3 | 0 | 1 | Read electronic ID data |
| Set burst length | SBL | 77h | 0 | 0 | 0 | Set burst length |
| Read SFDP | RDSFDP | 5Ah | | | | Read SFDP parameter |
| Release read enhanced | | FFh | | | | Release from read enhanced |
| Read unique ID | RUID | 4Bh | | 4 | 1+ | Read unique ID |

Note1 1byte data for SR0 with ordering option "D"

Note2 Only support with ordering option "D"

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

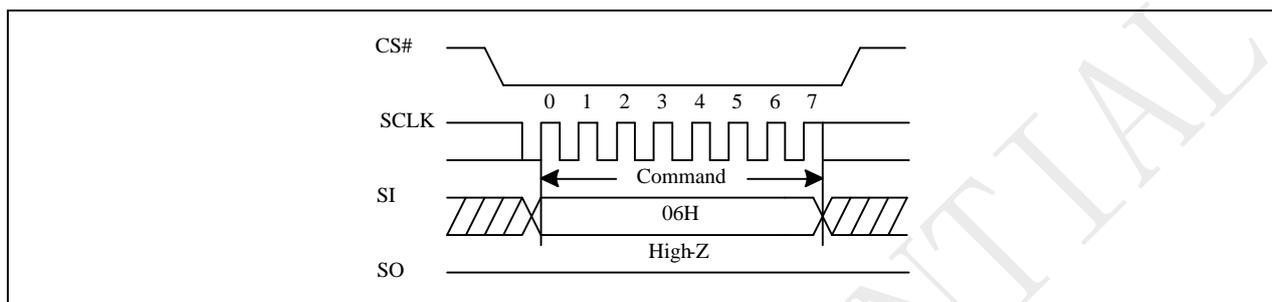
IO1 = A23, A21, A19, A17, A15, A13, A11, A9, A7, A5, A3, A1, M7, M5, M3, M1

10.2 Write Enable (WREN)

The Write Enable (WREN) instruction is for setting Write Enable Latch (WEL) bit. For those instructions like PP, PE, SE, BE32K, BE, CE and WRSR, WRCR, ERSCUR, PRSCUR which are intended to change the device content, should be set every time after the WREN instruction setting the WEL bit.

The sequence of issuing WREN instruction is: CS# goes low → sending WREN instruction code → CS# goes high.

Figure 10-2 Write Enable (WREN) Sequence (Command 06)



10.3 Write Disable (WRDI)

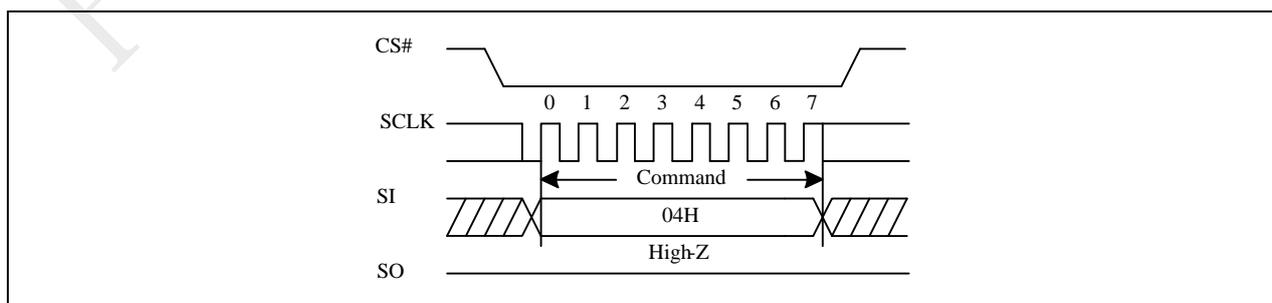
The Write Disable (WRDI) instruction is for resetting Write Enable Latch (WEL) bit.

The sequence of issuing WRDI instruction is: CS# goes low → sending WRDI instruction code → CS# goes high.

The WEL bit is reset by following situations:

- Power-up
- Write Disable (WRDI) instruction completion
- Write Status Register (WRSR/WRCR) instruction completion
- Page Program (PP) instruction completion
- Page Erase (PE) instruction completion
- Sector Erase (SE) instruction completion
- Block Erase (BE32K, BE) instruction completion
- Chip Erase (CE) instruction completion
- Erase Security Register (ERSCUR) instruction completion
- Program Security Register (PRSCUR) instruction completion
- Reset (RST) instruction completion

Figure 10-3 Write Disable (WRDI) Sequence (Command 04)

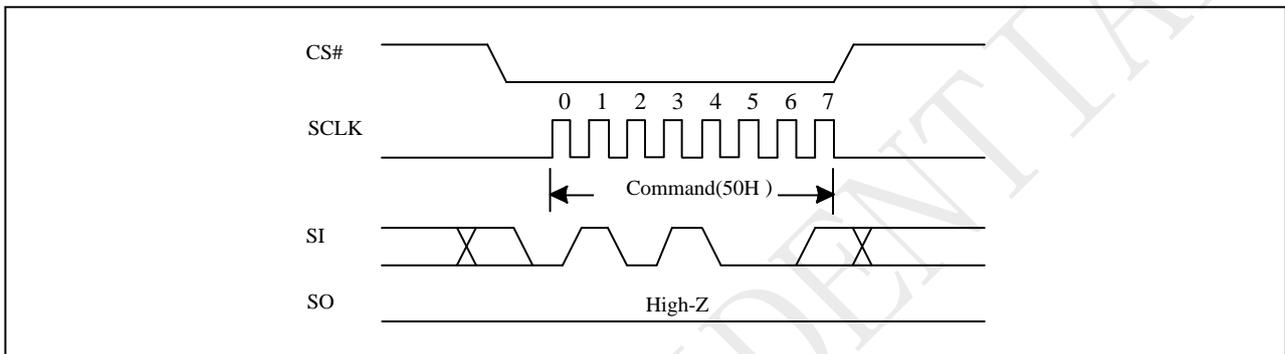


10.4 Write Enable for Volatile Status Register

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

The sequence of issuing Write Enable for Volatile Status Register instruction is: CS# goes low → sending Write Enable for Volatile Status Register instruction code → CS# goes high.

Figure 10-4 Write Enable for Volatile Status Register Sequence (Command 50)

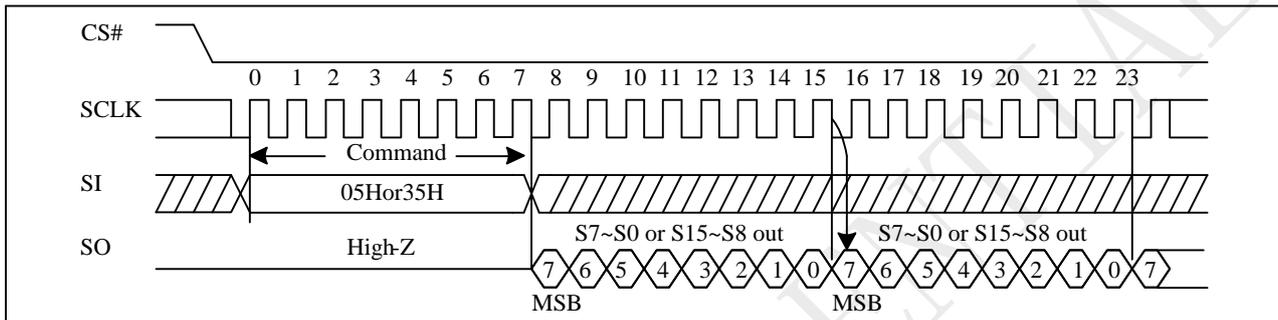


10.5 Read Status Register (RDSR)

The RDSR instruction is for reading Status Register Bits. The Read Status Register can be read at any time (even in program/erase/write status register condition). It is recommended to check the Write in Progress (WIP) bit before sending a new instruction when a program, erase, or write status register operation is in progress. 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.

The sequence of issuing RDSR instruction is: CS# goes low→ sending RDSR instruction code→ Status Register data out on SO. The SIO[3:1] are "don't care".

Figure 10-5 Read Status Register (RDSR) Sequence (Command 05 or 35)



Status Register

| | | | | | | | |
|-------------|------------|------------|------------|------------|----------------|------------|-------------|
| S15 | S14 | S13 | S12 | S11 | S10 | S9 | S8 |
| - | CMP | LB3 | LB2 | LB1 | EP_FAIL | - | SRP1 |
| S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 |
| SRP0 | BP4 | BP3 | BP2 | BP1 | BP0 | WEL | WIP |

The definition of the status register bits is as below:

WIP bit.

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

WEL bit.

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

BP4, BP3, BP2, BP1, BP0 bits.

The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase commands. These bits are written with the Write Status Register (WRSR) command. When the Block Protect (BP4, BP3, BP2, BP1, BP0) bits are set to 1, the relevant memory area (as defined in Table “Protected Area Sizes”).becomes protected against Page Program (PP), Page Erase (PE), Sector Erase (SE) and Block Erase (BE) commands. The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits can be written provided that the Hardware Protected mode has not been set. The Chip Erase (CE) command is executed, only if the Block Protect (BP4, BP3, BP2, BP1and BP0) are set to “None protected”.

SRP1, SRP0 bits.

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

| SRP1 | SRP0 | WP# | Status Register | Description |
|------|------|-----|---------------------------|---|
| 0 | 0 | x | Software Protected | The Status Register and Configure Register can be written to after a Write Enable command, |
| 0 | 1 | 0 | Hardware Protected | WP#=0, the Status Register and Configure Register locked and cannot be written to. |
| 0 | 1 | 1 | Hardware Unprotected | WP#=1, the Status Register and Configure Register is unlocked and can be written to after a Write Enable |
| 1 | 0 | x | Power Supply Lock-Down(1) | Status Register and Configure Register are protected and cannot be written to again until the next Power- |
| 1 | 1 | x | One Time Program(2) | Status Register and Configure Register are permanently protected and cannot be written to. |

NOTE:

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

EP_FAIL bit.

The Erase/Program Fail bit is a read only bit which shows the status of the last Program/Erase operation. The bit will be set to "1" if the program/erase operation failed or reseted or the program/erase region was protected. It will be automatically cleared to "0" if the next program/erase operation succeeds. Please note that it will not interrupt or stop any operation in the flash memory.

LB3, LB2, LB1, bits.

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

CMP bit

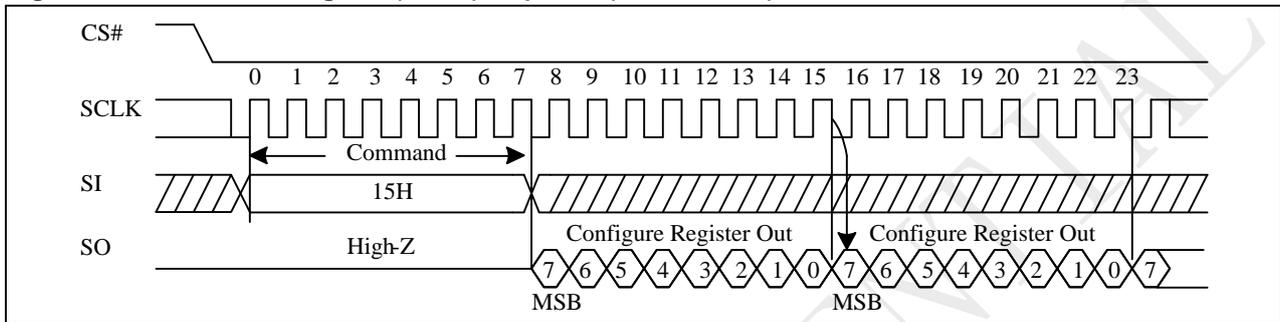
The CMP bit is a non-volatile Read/Write bit in the Status Register(S14). It is used in conjunction the BP4-BP0 bits to provide more flexibility for the array protection. Please see the table "Protected Area Size" for details. The default setting is CMP=0.

10.6 Read Configure Register (RDCR)

The RDCR instruction is for reading Configure Register Bits. The Read Configure Register can be read at any time (even in program/erase/write status register condition). It is recommended to check the Write in Progress (WIP) bit before sending a new instruction when a program, erase, or write status register operation is in progress.

The sequence of issuing RDCR instruction is: CS# goes low→ sending RDCR instruction code→ Configure Register data out on SO. The SIO[3:1] are "don't care".

Figure 10-6 Read Status Register (RDCR) Sequence (Command 15)



Configure Register

| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------|---------|---------|---------|---------|---------|------|---------|
| HOLD/RST | reserve | reserve | reserve | reserve | reserve | DC | reserve |

HOLD/RST bit.

The HOLD/RST bit is a non volatile Read/Write bit in the Configure Register which is used to determine whether /HOLD or /RESET function should be implemented on the hardware pin for 8-pin packages. When HOLD/RST=0 (factory default), the pin acts as /HOLD; when HOLD/RST=1, the pin acts as /RESET.

DC bit

The Dummy Cycle (DC) bit is a volatile bit. The Dummy Cycle (DC) bit can be used to configure the number of dummy clocks for "SPI 2 X IO Read (BBH)" command, "SPI 4X I/O Read (EBH)" command.

Table Dummy Cycle Table

| | | DC1 | Number of dummy | Max Read Freq. |
|----------------|--------|------------|-----------------|----------------|
| SPI command | BB SPI | 0(default) | 4 | 104MHz |
| | | 1 | 8 | 120MHz |
| | EB SPI | 0(default) | 6 | 104MHz |
| | | 1 | 10 | 120MHz |

10.7 Write Status Register (WRSR)

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

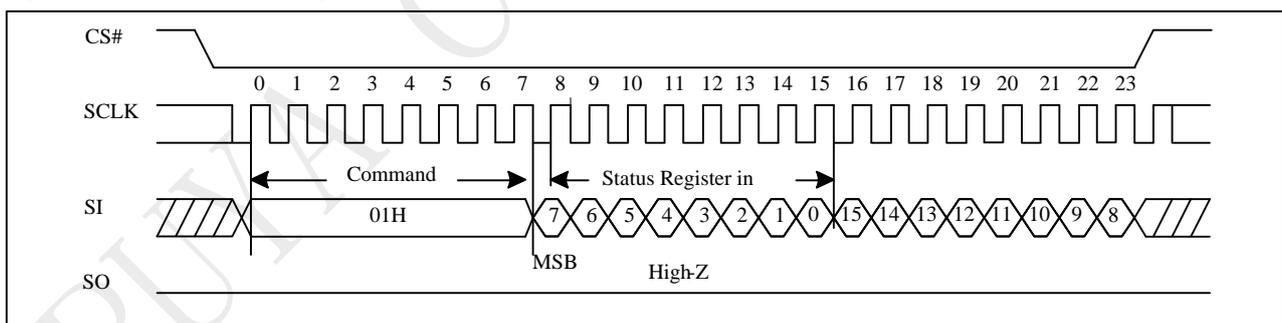
The Write Status Register (WRSR) command has no effect on S15, S10, S1 and S0 of the Status Register. CS# must be driven high after the eighth or sixteen bit of the data byte has been latched in. If not, the Write Status Register (WRSR) command is not executed. If CS# is driven high after eighth bit of the data byte, the CMP and QE and SRP1 bits will be cleared to 0. As soon as CS# is driven high, the self-timed Write Status Register cycle (whose duration is t_W) is initiated. While the Write Status Register cycle is in progress, the Status Register may still be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Write Status Register cycle, and is 0 when it is completed. When the cycle is completed, the Write Enable Latch (WEL) is reset.

The Write Status Register (WRSR) command allows the user to change the values of the Block Protect (BP4, BP3, BP2, BP1, and BP0) bits, to define the size of the area that is to be treated as read-only, as defined in Table 1. The Write Status Register (WRSR) command also allows the user to set or reset the Status Register Protect (SRP1 and SRP0) bits in accordance with the Write Protect (WP#) signal. The Status Register Protect (SRP1 and SRP0) bits 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.

The sequence of issuing WRSR instruction is: CS# goes low → sending WRSR instruction code → Status Register data on SI → CS# goes high.

The CS# must go high exactly at the 8 bits or 16 bits data boundary; otherwise, the instruction will be rejected and not executed. The self-timed Write Status Register cycle time (t_W) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Write Status Register cycle is in progress. The WIP sets 1 during the t_W timing, and sets 0 when Write Status Register Cycle is completed, and the Write Enable Latch (WEL) bit is reset.

Figure 10-7 Write Status Register (WRSR) Sequence (Command 01)



10.8 Write Status Register with ordering option "D"(WRSR)

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

The Write Status Register (WRSR) command has no effect on S15, S10, S1 and S0 of the Status Register. CS# must be driven high after the eighth 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 and SRP1 bits will be cleared to 0. As soon as CS# is driven high, the self-timed Write Status Register cycle (whose duration is t_W) is initiated. While the Write Status Register cycle is in progress, the Status Register may still be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Write Status Register cycle, and is 0 when it is completed. When the cycle is completed, the Write Enable Latch (WEL) is reset.

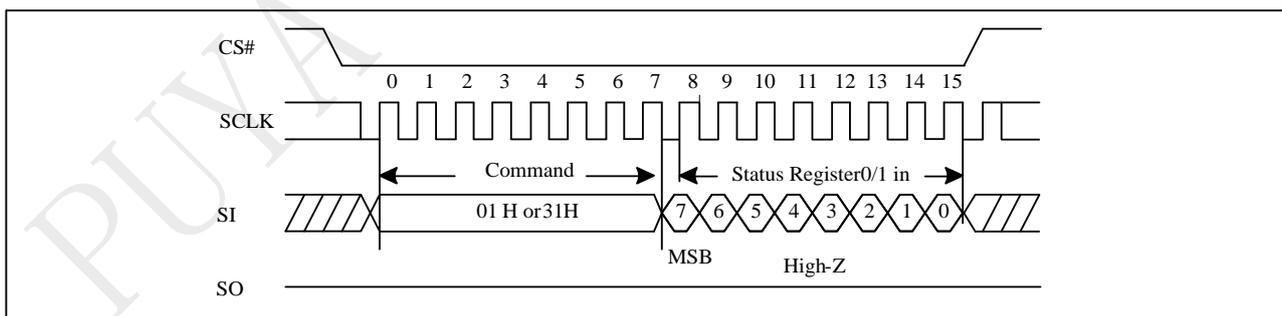
The Write Status Register (WRSR) command allows the user to change the values of the Block Protect (BP4, BP3, BP2, BP1, and BP0) bits, to define the size of the area that is to be treated as read-only, as defined in Table1. The Write Status Register (WRSR) command also allows the user to set or reset the Status Register Protect (SRP1 and SRP0) bits in accordance with the Write Protect (WP#) signal. The Status Register Protect (SRP1 and SRP0) bits 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.

The sequence of issuing WRSR instruction is: CS# goes low → sending WRSR instruction code → Status Register data on SI → CS# goes high.

The CS# must go high exactly at the 8 bits data boundary; otherwise, the instruction will be rejected and not executed. The self-timed Write Status Register cycle time (t_W) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Write Status Register cycle is in progress. The WIP sets 1 during the t_W timing, and sets 0 when Write Status Register Cycle is completed, and the Write Enable Latch (WEL) bit is reset.

Command code "31H" is only valid with ordering option "D"

Figure 10-8 Write Status Register (WRSR) Sequence with ordering option "D" (Command 01/31h)



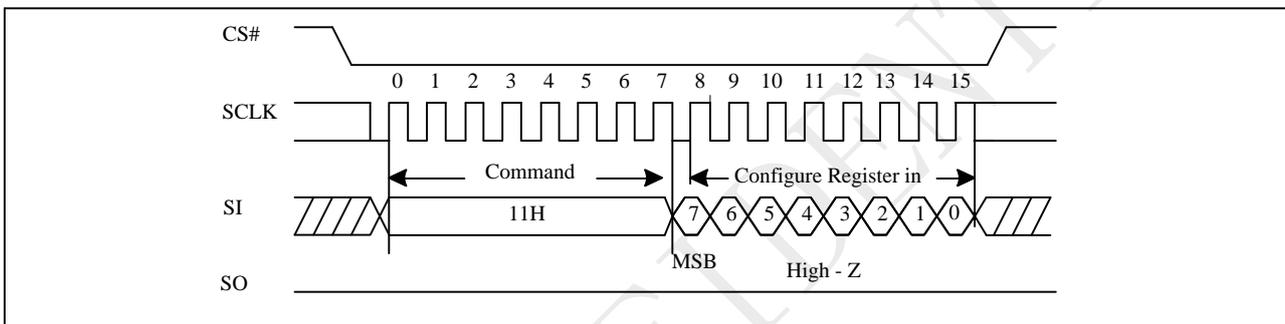
10.9 Write Configure Register (WRCR)

The Write Configure Register (WRCR) command allows new values to be written to the Configure 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 sequence of issuing WRCR instruction is: CS# goes low → sending WRCR instruction code → Configure Register data on SI → CS# goes high.

The CS# must go high exactly at the 8 bits data boundary; otherwise, the instruction will be rejected and not executed. The self-timed Write Status Register cycle time (t_W) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Write Status Register cycle is in progress. The WIP sets 1 during the t_W timing, and sets 0 when Write Configure Register Cycle is completed, and the Write Enable Latch (WEL) bit is reset.

Figure 10-9 Write Configure Register (WRCR) Sequence (Command 11)

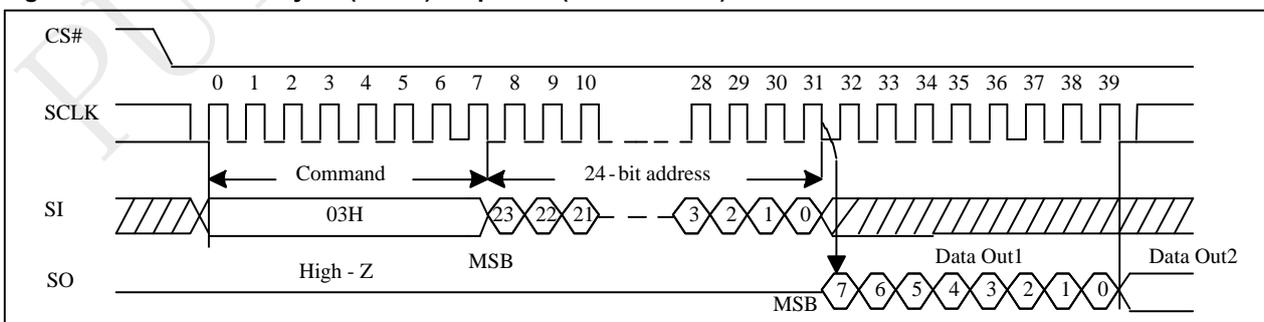


10.10 Read Data Bytes (READ)

The read instruction is for reading data out. The address is latched on rising edge of SCLK, and data shifts out on the falling edge of SCLK at a maximum frequency f_R . The first address byte can be at any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single READ instruction. The address counter rolls over to 0 when the highest address has been reached.

The sequence of issuing READ instruction is: CS# goes low → sending READ instruction code → 3-byte address on SI → data out on SO → to end READ operation can use CS# to high at any time during data out.

Figure 10-10 Read Data Bytes (READ) Sequence (Command 03)



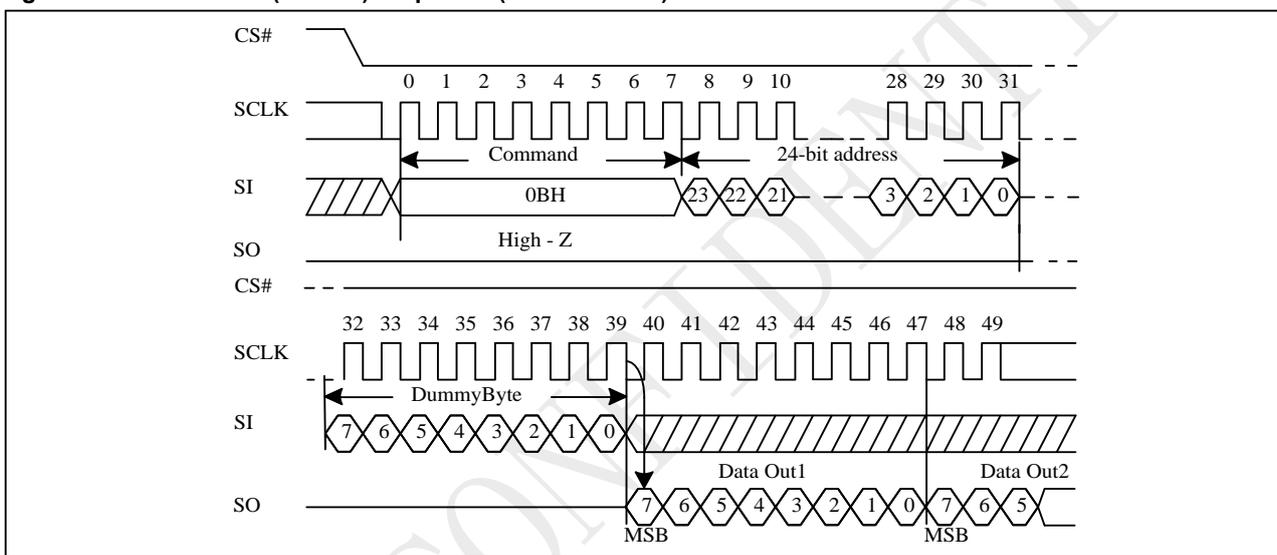
10.11 Fast Read (FREAD)

The FAST READ instruction is for quickly reading data out. The address is latched on rising edge of SCLK, and data of each bit shifts out on the falling edge of SCLK at a maximum frequency f_C . The first address byte can be at any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single FREAD instruction. The address counter rolls over to 0 when the highest address has been reached.

The sequence of issuing FREAD instruction is: CS# goes low → sending FREAD instruction code → 3-byte address on SI → 1-dummy byte address on SI → data out on SO → to end FREAD operation can use CS# to high at any time during data out.

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

Figure 10-10 Fast Read (FREAD) Sequence (Command 0B)



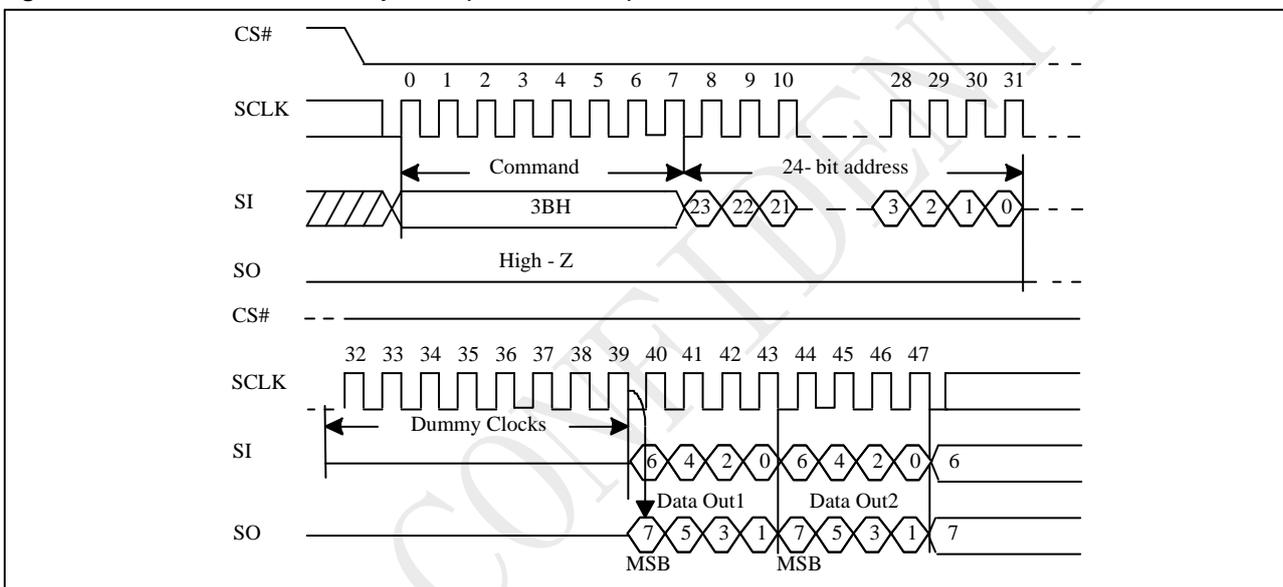
10.12 Dual Read (DREAD)

The DREAD instruction enable double throughput of Serial NOR Flash in read mode. The address is latched on rising edge of SCLK, and data of every two bits (interleave on 2 I/O pins) shift out on the falling edge of SCLK at a maximum frequency f_T . The first address byte can be at any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single DREAD instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing DREAD instruction, the following data out will perform as 2-bit instead of previous 1-bit.

The sequence of issuing DREAD instruction is: CS# goes low → sending DREAD instruction → 3-byte address on SI → 8-bit dummy cycle → data out interleave on SIO1 & SIO0 → to end DREAD operation can use CS# to high at any time during data out.

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

Figure 10-12 Dual Read Mode Sequence (Command 3B)



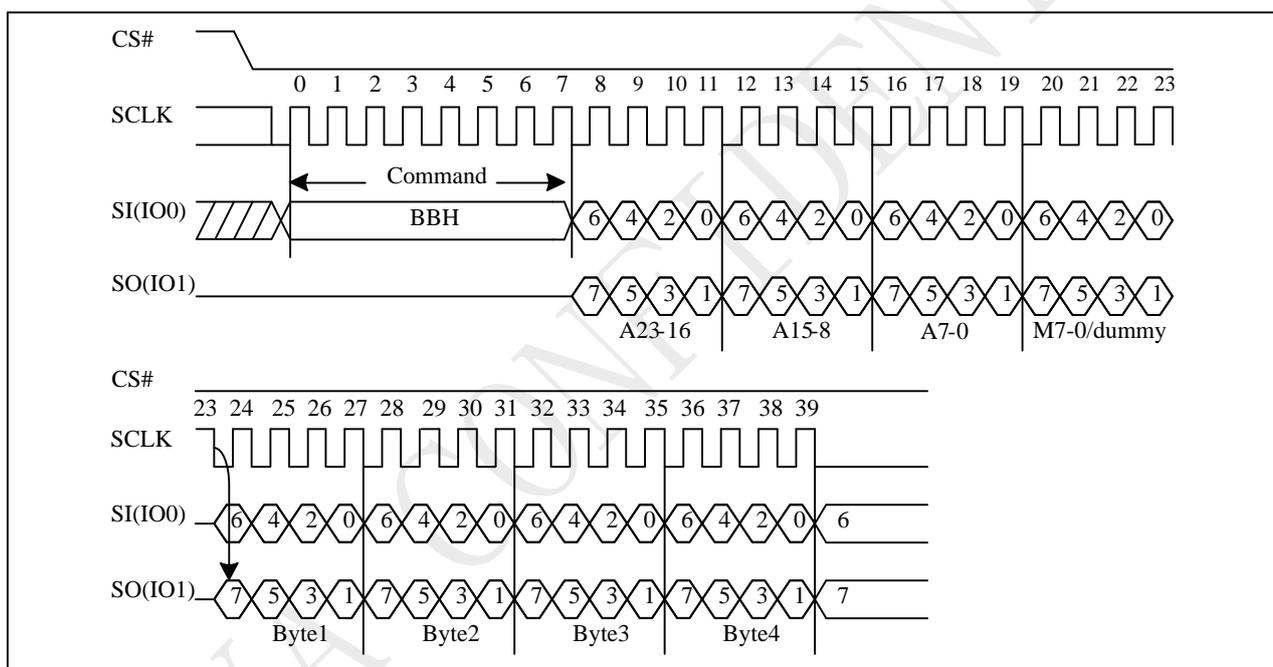
10.13 2IO Read (2READ)

The 2READ instruction enables Double Transfer Rate of Serial NOR Flash in read mode. The address is latched on rising edge of SCLK, and data of every two bits (interleave on 2 I/O pins) shift out on the falling edge of SCLK at a maximum frequency f_T . The first address byte can be at any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single 2READ instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing 2READ instruction, the following address/dummy/data out will perform as 2-bit instead of previous 1-bit.

The sequence of issuing 2READ instruction is: CS# goes low → sending 2READ instruction → 24-bit address interleave on SIO1 & SIO0 → 8-bit dummy cycle on SIO1 & SIO0 → data out interleave on SIO1 & SIO0 → to end 2READ operation can use CS# to high at any time during data out.

While Program/Erase/Write Status Register cycle is in progress, 2READ instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

Figure 10-13 2IO Read Sequence (Command BB M5-4 ≠ (1,0))



Note:

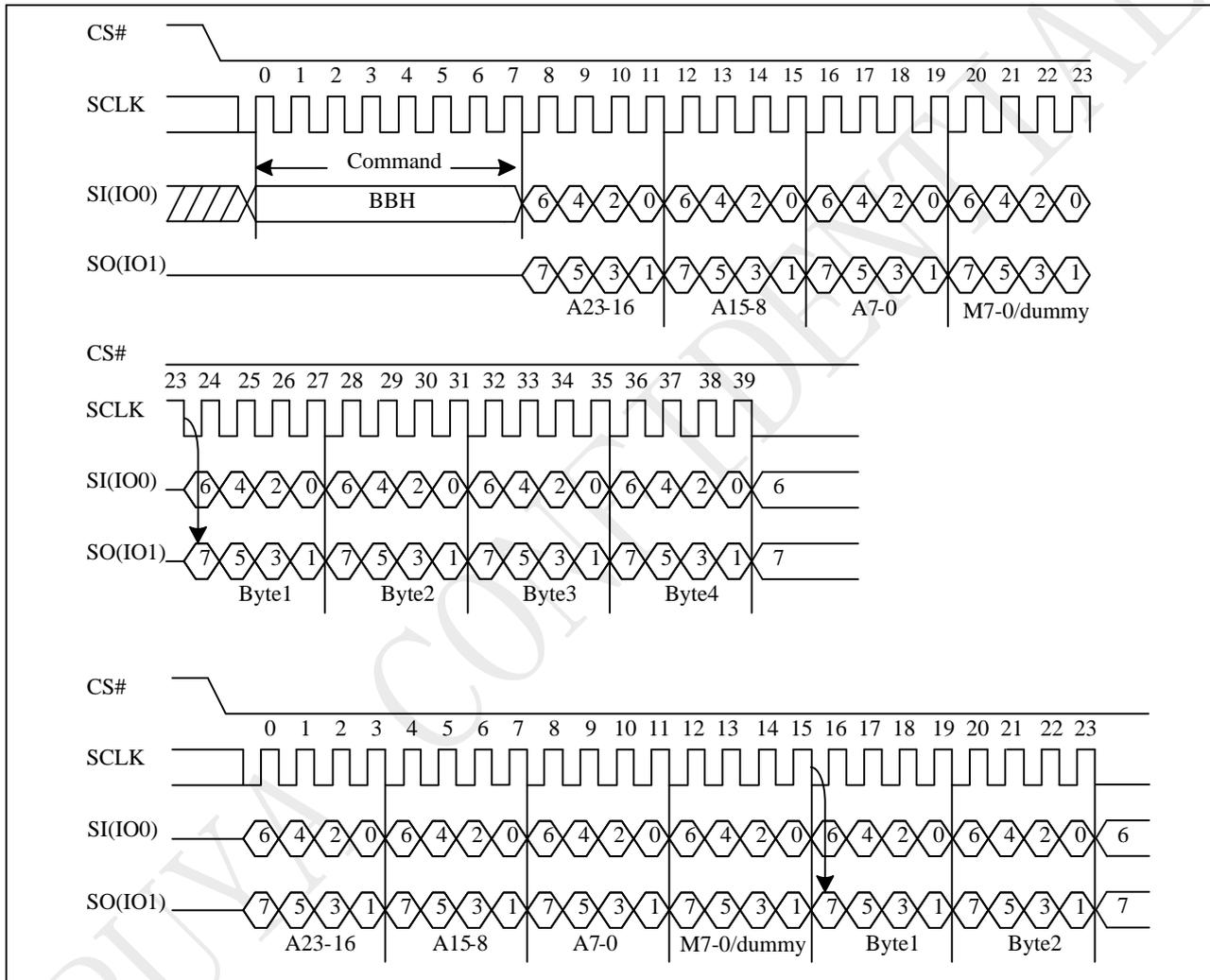
1. M[5-4] = (1,0) is inhibited.
2. DC bit can set the number of dummy clocks.

2IO Continuous Read

“BBh” command supports 2IO Continuous Read which 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 2IO Read command (after CS# is raised and then lowered) does not require the BBH command code.

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.

Figure 10-13a 2IO Continue Read (M5-4 = (1,0))



Note:

1. 2IO Continue Read, if M5-4 = 1, 0. If not using Continue Read recommend to set M5-4 ≠ 1, 0.
2. DC bit can set the number of dummy clocks.

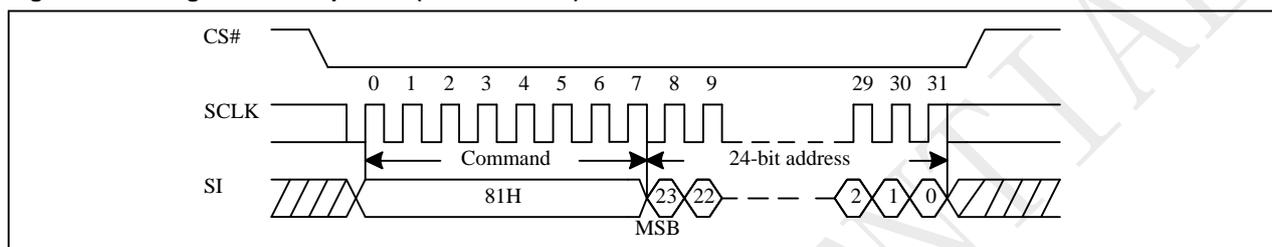
10.14 Page Erase (PE)

The Page Erase (PE) instruction is for erasing the data of the chosen Page to be "1". A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Page Erase (PE).

To perform a Page Erase with the standard page size (256 bytes), an instruction of 81h must be clocked into the device followed by three address bytes comprised of 2 page address bytes that specify the page in the main memory to be erased, and 1 dummy byte.

The sequence of issuing PE instruction is: CS# goes low → sending PE instruction code → 3-byte address on SI → CS# goes high.

Figure 10-26 Page Erase Sequence (Command 81)



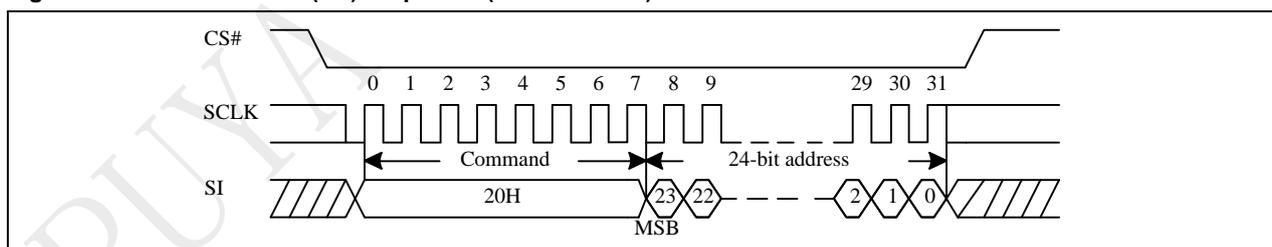
10.15 Sector Erase (SE)

The Sector Erase (SE) instruction is for erasing the data of the chosen sector to be "1". A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Sector Erase (SE). Any address of the sector is a valid address for Sector Erase (SE) instruction. The CS# must go high exactly at the byte boundary (the latest eighth of address byte been latched-in); otherwise, the instruction will be rejected and not executed.

Address bits [Am-A12] (Am is the most significant address) select the sector address.

The sequence of issuing SE instruction is: CS# goes low → sending SE instruction code → 3-byte address on SI → CS# goes high. The SIO[3:1] are don't care.

Figure 10-27 Sector Erase (SE) Sequence (Command 20)



The self-timed Sector Erase Cycle time (tSE) is initiated as soon as Chip Select (CS#) goes high. The Write in progress (WIP) bit still can be checked out during the Sector Erase cycle is in progress. The WIP sets 1 during the tSE timing, and sets 0 when Sector Erase Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the sector is protected by BP4, BP3, BP2, BP1, BP0 bits, the Sector Erase (SE) instruction will not be executed on the sector.

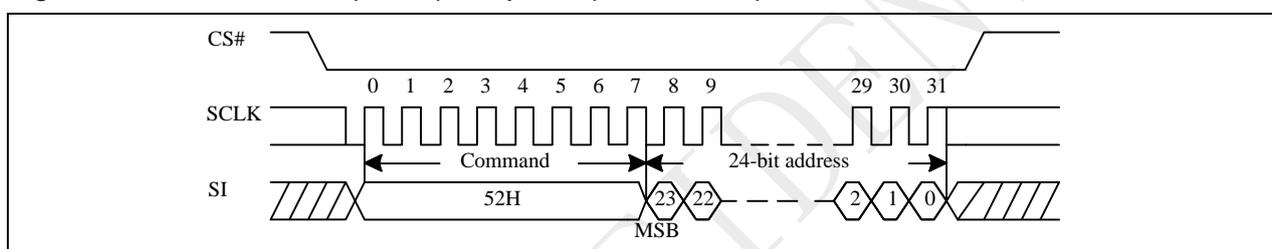
10.16 Block Erase (BE32K)

The Block Erase (BE32K) instruction is for erasing the data of the chosen block to be "1". The instruction is used for 32K-byte block erase operation. A Write Enable (WREN) instruction must be executed to set the Write Enable Latch (WEL) bit before sending the Block Erase (BE32K). Any address of the block is a valid address for Block Erase (BE32K) instruction. The CS# must go high exactly at the byte boundary (the least significant bit of address byte has been latched-in); otherwise, the instruction will be rejected and not executed.

The sequence of issuing BE32K instruction is: CS# goes low → sending BE32K instruction code → 3-byte address on SI → CS# goes high. The SIO[3:1] are don't care.

The self-timed Block Erase Cycle time (tBE32K) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked while the Block Erase cycle is in progress. The WIP sets during the tBE32K timing, and clears when Block Erase Cycle is completed, and the Write Enable Latch (WEL) bit is cleared. If the block is protected by BP4, BP3, BP2, BP1, BP0 bits, the array data will be protected (no change) and the WEL bit still be reset.

Figure 10-28 Block Erase 32K(BE32K) Sequence (Command 52)



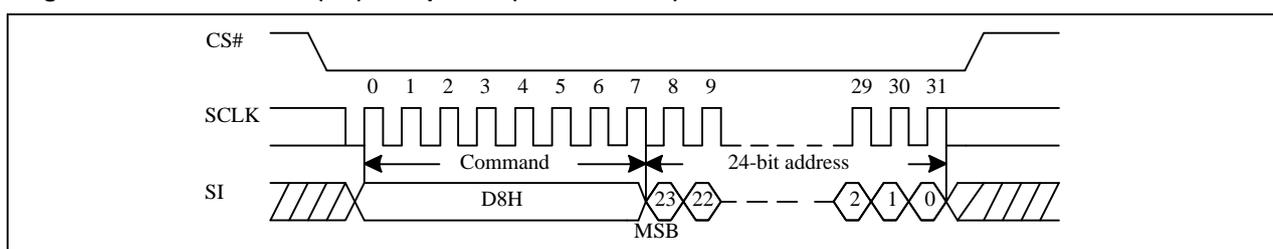
10.17 Block Erase (BE)

The Block Erase (BE) instruction is for erasing the data of the chosen block to be "1". The instruction is used for 64K-byte block erase operation. A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Block Erase (BE). Any address of the block is a valid address for Block Erase (BE) instruction. The CS# must go high exactly at the byte boundary (the latest eighth of address byte been latched-in); otherwise, the instruction will be rejected and not executed.

The sequence of issuing BE instruction is: CS# goes low → sending BE instruction code → 3-byte address on SI → CS# goes high. The SIO[3:1] are "don't care".

The self-timed Block Erase Cycle time (tBE) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Block Erase cycle is in progress. The WIP sets 1 during the tBE timing, and sets 0 when Block Erase Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the block is protected by BP4, BP3, BP2, BP1, BP0 bits, the Block Erase (BE) instruction will not be executed on the block.

Figure 10-29 Block Erase (BE) Sequence (Command D8)

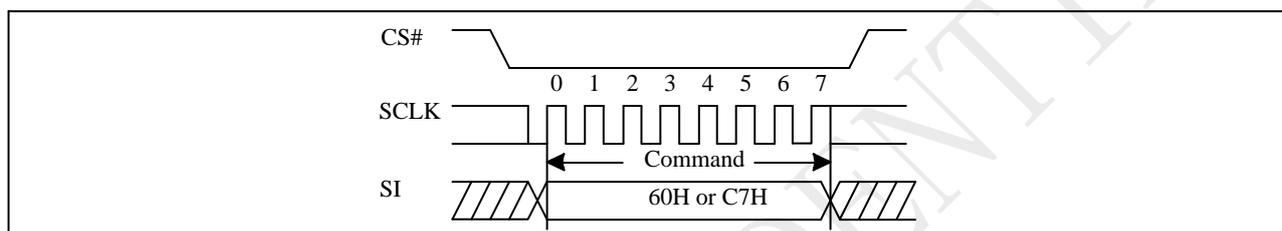


10.18 Chip Erase (CE)

The Chip Erase (CE) instruction is for erasing the data of the whole chip to be "1". A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Chip Erase (CE). The CS# must go high exactly at the byte boundary (the latest eighth of address byte been latched-in); otherwise, the instruction will be rejected and not executed.

The sequence of issuing CE instruction is: CS# goes low→ sending CE instruction code→ CS# goes high. The self-timed Chip Erase Cycle time (tCE) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Chip Erase cycle is in progress. The WIP sets 1 during the tCE timing, and sets 0 when Chip Erase Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the chip is protected by BP4, BP3, BP2, BP1, BP0 bits, the Chip Erase (CE) instruction will not be executed. It will be only executed when all Block Protect (BP4, BP3, BP2, BP1, BP0) are set to "None protected".

Figure 10-30 Chip Erase (CE) Sequence (Command 60 or C7)



10.19 Page Program (PP)

The Page Program (PP) instruction is for programming the memory to be "0". A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Page Program (PP). The device programs only the last 256 data bytes sent to the device. If the entire 256 data bytes are going to be programmed, A7-A0 (The eight least significant address bits) should be set to 0. If the eight least significant address bits (A7-A0) are not all 0, all transmitted data going beyond the end of the current page are programmed from the start address of the same page (from the address A7-A0 are all 0). If more than 256 bytes are sent to the device, the data of the last 256-byte is programmed at the request page and previous data will be disregarded. If less than 256 bytes are sent to the device, the data is programmed at the requested address of the page.

Each page programming operation should be done only once after erase operation. It is possible to program from one byte up to a page size in each Page programming operation.

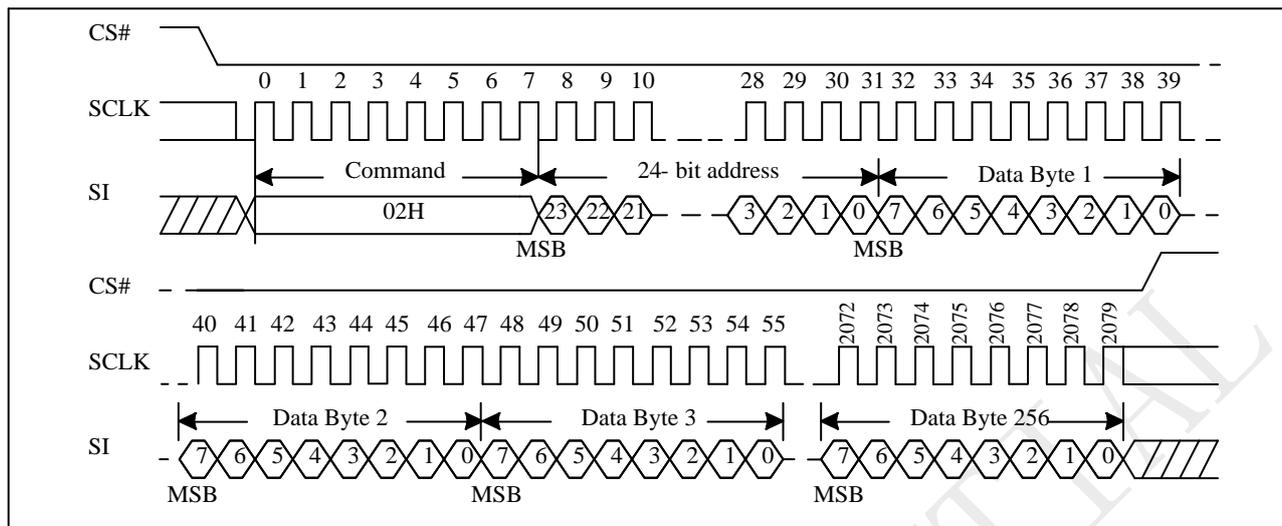
For the very best performance, programming should be done in full pages of 256 bytes aligned on 256 byte boundaries with each Page being programmed only once.

The sequence of issuing PP instruction is: CS# goes low→ sending PP instruction code→ 3-byte address on SI→ at least 1-byte on data on SI→ CS# goes high.

The CS# must be kept low during the whole Page Program cycle; The CS# must go high exactly at the byte boundary (the latest eighth bit of data being latched in), otherwise the instruction will be rejected and will not be executed.

The self-timed Page Program Cycle time (tPP) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Page Program cycle is in progress. The WIP sets 1 during the tPP timing, and sets 0 when Page Program Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the page is protected by BP4, BP3, BP2, BP1, BP0 bits, the Page Program (PP) instruction will not be executed. The SIO[3:1] are "don't care".

Figure 10-31 Page Program (PP) Sequence (Command 02)



10.20 Erase Security Registers (ERSCUR)

The product provides three 512-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.

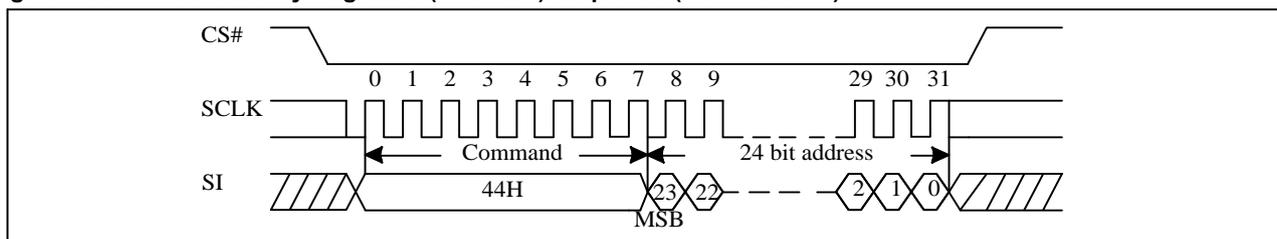
The Erase Security Registers command is similar to Sector/Block Erase command. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit.

The Erase Security Registers command sequence: CS# goes low → sending ERSCUR instruction → sending 24 bit address → CS# goes high.

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 tSE) 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. The Security Registers Lock Bit (LB3-1) in the Status Register can be used to OTP protect the security registers. Once the LB bit is set to 1, the Security Registers will be permanently locked; the Erase Security Registers command will be ignored.

| Address | A23-16 | A15-12 | A11-9 | A8-0 |
|----------------------|--------|--------|-------|--------------|
| Security Register #1 | 00H | 0001 | 000 | Byte Address |
| Security Register #2 | 00H | 0010 | 000 | Byte Address |
| Security Register #3 | 00H | 0011 | 000 | Byte Address |

Figure 10-37 Erase Security Registers (ERSCUR) Sequence (Command 44)



10.21 Program Security Registers (PRSCUR)

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.

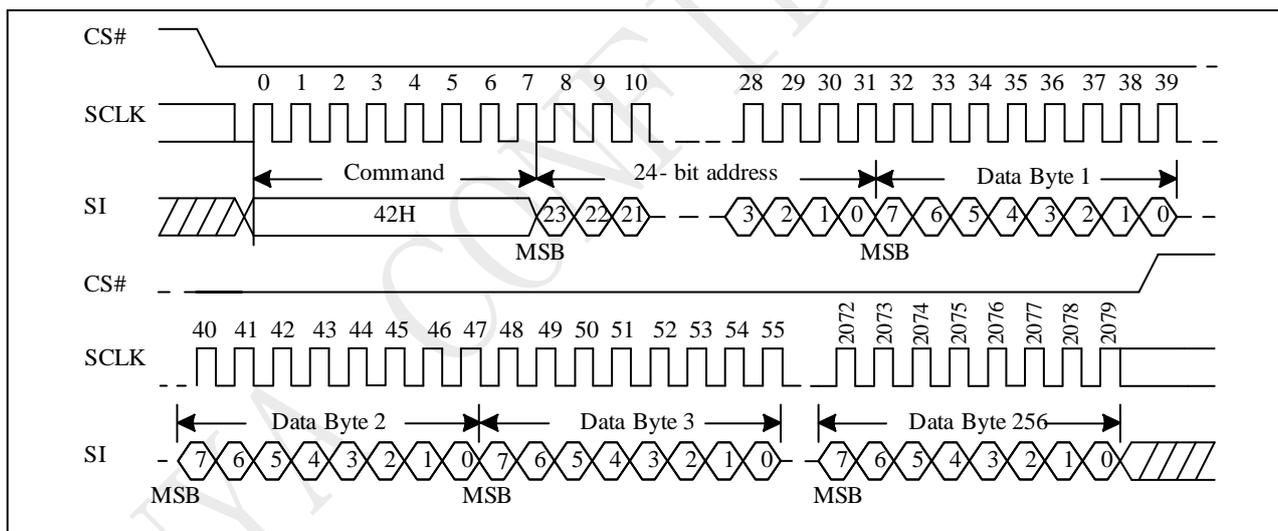
The Program Security Registers command sequence: CS# goes low → sending PRSCUR instruction → sending 24 bit address → sending at least one byte data → CS# goes high.

As soon as CS# is driven high, the self-timed Program Security Registers cycle (whose duration is tPP) 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.

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

| Address | A23-16 | A15-12 | A11-9 | A8-0 |
|----------------------|--------|--------|-------|--------------|
| Security Register #1 | 00H | 0001 | 0000 | Byte Address |
| Security Register #2 | 00H | 0010 | 0000 | Byte Address |
| Security Register #3 | 00H | 0011 | 0000 | Byte Address |

Figure 10-38 Program Security Registers (PRSCUR) Sequence (Command 42)



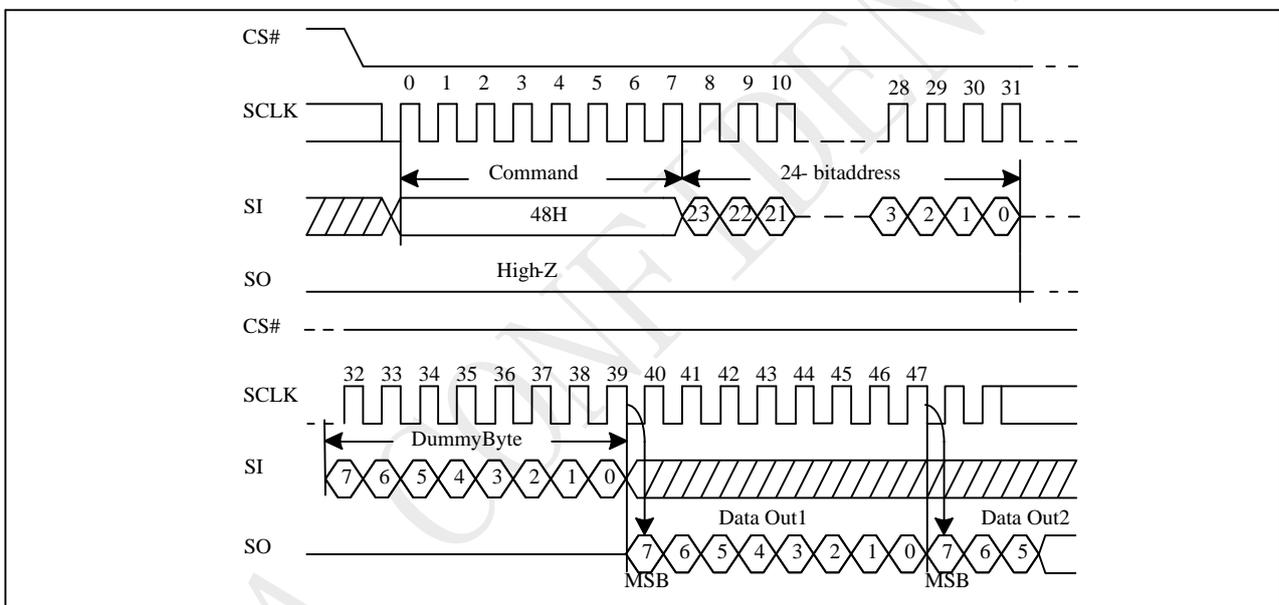
10.22 Read Security Registers (RDSCUR)

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 fC, 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 A8-A0 address reaches the last byte of the register (Byte 1FFH), it will reset to 00H, the command is completed by driving CS# high.

The sequence of issuing RDSCUR instruction is : CS# goes low → sending RDSCUR instruction → sending 24 bit address → 8 bit dummy byte → Security Register data out on SO → CS# goes high.

| Address | A23-16 | A15-12 | A11-9 | A8-0 |
|----------------------|--------|--------|-------|--------------|
| Security Register #1 | 00H | 0001 | 000 | Byte Address |
| Security Register #2 | 00H | 0010 | 000 | Byte Address |
| Security Register #3 | 00H | 0011 | 000 | Byte Address |

Figure 10-39 Read Security Registers (RDSCUR) Sequence (Command 48)



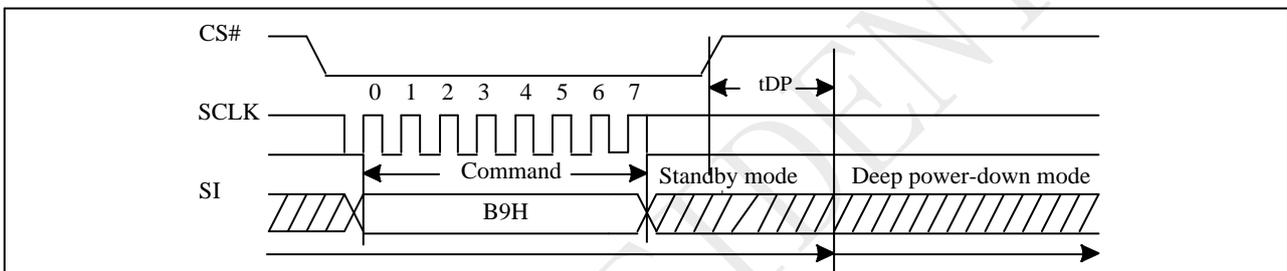
10.23 Deep Power-down (DP)

The Deep Power-down (DP) instruction is for setting the device on the minimizing the power consumption (to entering the Deep Power-down mode), the standby current is reduced from ISB1 to ISB2). The Deep Power-down mode requires the Deep Power-down (DP) instruction to enter, during the Deep Power-down mode, the device is not active and all Write/Program/Erase instruction are ignored.

The sequence of issuing DP instruction is: CS# goes low → sending DP instruction code → CS# goes high.

Once the DP instruction is set, all instruction will be ignored except the Release from Deep Power-down mode (RDP), Read Electronic Signature (RES) instruction, and soft reset instruction (66H, 99H). (RES instruction to allow the ID been read out). When Power-down, the deep power-down mode automatically stops, and when power-up, the device automatically is in standby mode. For RDP instruction the CS# must go high exactly at the byte boundary (the latest eighth bit of instruction code been latched-in); otherwise, the instruction will not be executed. As soon as Chip Select (CS#) goes high, a delay of t_{DP} is required before entering the Deep Power-down mode and reducing the current to ISB2.

Figure 10-40 Deep Power-down (DP) Sequence (Command B9)



10.24 Release from Deep Power-Down (RDP), Read Electronic Signature (RES)

The Release from Deep Power-down (RDP) instruction is terminated by driving Chip Select (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 Stand-by Power mode is delayed by t_{RES2} , and Chip Select (CS#) must remain High for at least $t_{RES2(max)}$. Once in the Stand-by Power mode, the device waits to be selected, so that it can receive, decode and execute instructions.

RES instruction is for reading out the old style of 8-bit Electronic Signature, whose values are shown as table of ID Definitions. This is not the same as RDID instruction. It is not recommended to use for new design. For new design, please use RDID instruction. Even in Deep power-down mode, the RDP and RES are also allowed to be executed, only except the device is in progress of program/erase/write cycle; there's no effect on the current program/erase/ write cycle in progress.

The RES instruction is ended by CS# goes high after the ID been read out at least once. The ID outputs repeatedly if continuously send the additional clock cycles on SCLK while CS# is at low. If the device was not previously in Deep Power-down mode, the device transition to standby mode is immediate. If the device was previously in Deep Power-down mode, there's a delay of t_{RES2} to transit to standby mode, and CS# must remain to high at least $t_{RES2 (max)}$. Once in the standby mode, the device waits to be selected, so it can be receive, decode, and execute instruction.

The RDP instruction is for releasing from Deep Power-Down Mode.

Figure 10-41 Read Electronic Signature (RES) Sequence (Command AB)

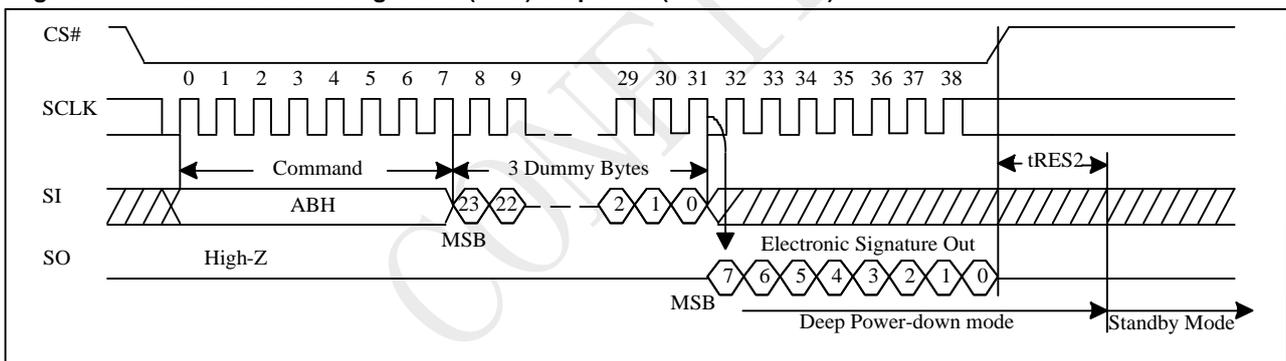
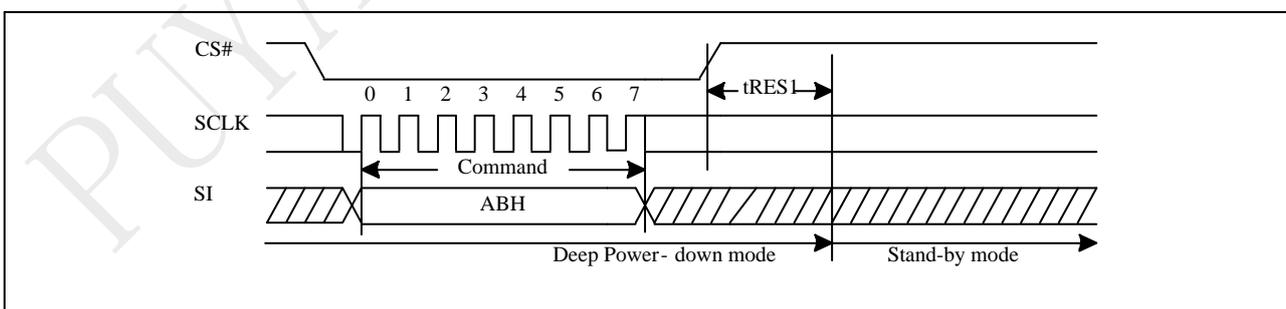


Figure 10-41b Release from Deep Power-down (RDP) Sequence (Command AB)

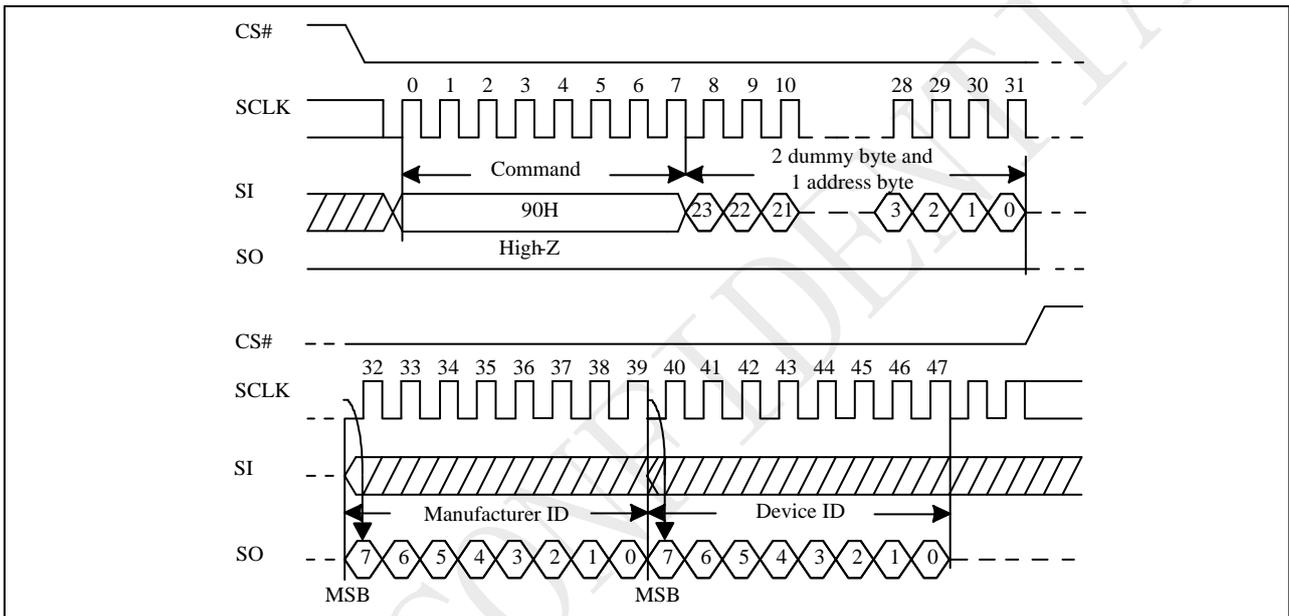


10.25 Read Electronic Manufacturer ID & Device ID (REMS)

The REMS instruction returns both the JEDEC assigned manufacturer ID and the device ID. The Device ID values are listed in "Table ID Definitions".

The REMS instruction is initiated by driving the CS# pin low and sending the instruction code "90h" followed by two dummy bytes and one address byte (A7~A0). After which the manufacturer ID for PUYA (85h) and the device ID are shifted out on the falling edge of SCLK with the most significant bit (MSB) first. If the address byte is 00h, the manufacturer ID will be output first, followed by the device ID. If the address byte is 01h, then the device ID will be output first, followed by the manufacturer ID. While CS# is low, the manufacturer and device IDs can be read continuously, alternating from one to the other. The instruction is completed by driving CS# high.

Figure 10-42 Read Electronic Manufacturer & Device ID (REMS) Sequence (Command 90)

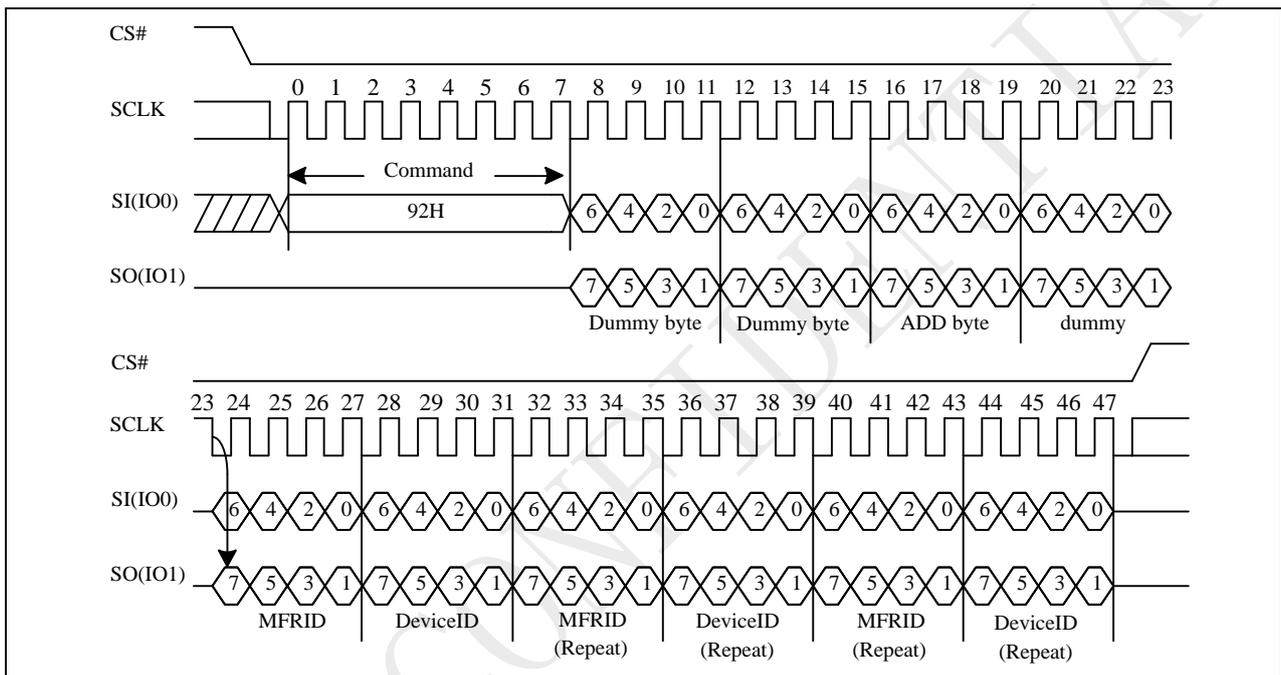


10.26 Dual I/O Read Electronic Manufacturer ID & Device ID (DREMS)

The DREMS instruction is similar to the REMS command and returns the JEDEC assigned manufacturer ID which takes two pins: SIO0, SIO1 as address input and ID output I/O

The instruction is initiated by driving the CS# pin low and shift the instruction code "92h" followed by two dummy bytes and one bytes address (A7~A0). After which, the Manufacturer ID for PUYA (85h) and the Device ID are shifted out on the falling edge of SCLK with most significant bit (MSB) first. If the one-byte address is initially set to 01h, then the device ID will be read first and then followed by the Manufacturer ID. The Manufacturer and Device IDs can be read continuously, alternating from one to the other. The instruction is completed by driving CS# high.

Figure 10-43 DUAL I/O Read Electronic Manufacturer & Device ID (DREMS) Sequence (Command 92)



10.27 Read Identification (RDID)

The RDID instruction is for reading the manufacturer ID of 1-byte and followed by Device ID of 2-byte. The PUYA Manufacturer ID and Device ID are list as "Table . ID Definitions".

The sequence of issuing RDID instruction is: CS# goes low → sending RDID instruction code → 24-bits ID data out on SO → to end RDID operation can use CS# to high at any time during data out. While Program /Erase operation is in progress, it will not decode the RDID instruction, so there's no effect on the cycle of program/erase operation which is currently in progress. When CS# goes high, the device is at standby stage.

Figure 10-45 Read Identification (RDID) Sequence (Command 9F)

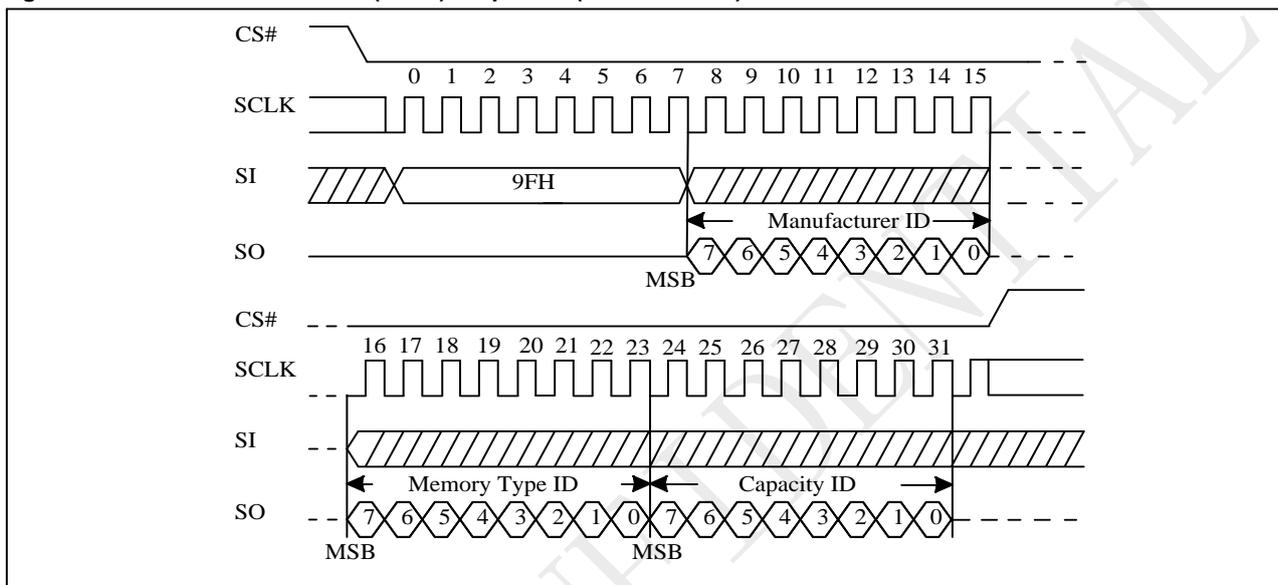


Table ID Definitions

| | RDID command | manufacturer ID | memory type | memory density |
|----------|--------------|-----------------|-------------|----------------|
| P25D40SH | | 85 | 60 | 13 |
| | RES command | electronic ID | | |
| | | 12 | | |
| | REMS command | manufacturer ID | | device ID |
| | | 85 | 12 | |

10.28 No Operation (NOP)

The "No Operation" command is only able to terminate the Reset Enable (RSTEN) command and will not affect any other command.

The SIO[3:1] are don't care.

10.29 Individual Block Lock (SBLK)

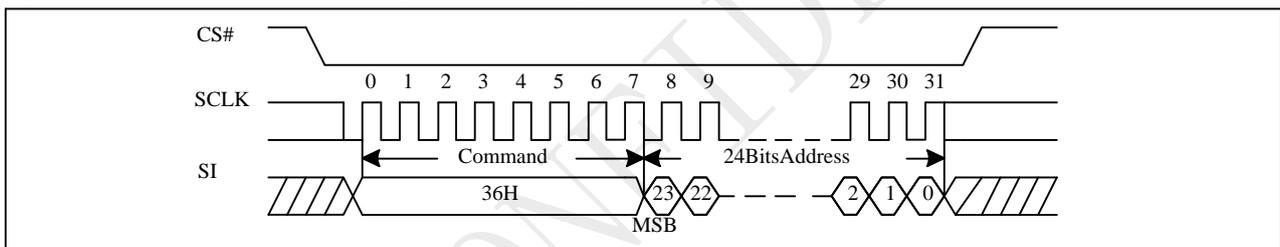
The write protection will be determined by the combination of CMP, BP[4:0] bits in the Status Registers. The Individual Block Lock bits are volatile bits. The default values after device power up or after a Reset are 1, so the entire memory array is being protected.

The SBLK instruction is for write protection a specified block (or sector) of memory, using AMAX-A16 or (AMAX-A12) address bits to assign a 64Kbyte block (or 4K bytes sector) to be protected as read only.

The WREN (Write Enable) instruction is required before issuing SBLK instruction.

The sequence of issuing SBLK instruction is: CS# goes low → send SBLK (36h) instruction → send 3-byte address assign one block (or sector) to be protected on SI pin → CS# goes high. The CS# must go high exactly at the byte boundary, otherwise the instruction will be rejected and not be executed.

Figure 10-50 Individual Block Lock(Command 36H)



10.30 Individual Block Unlock (SBULK)

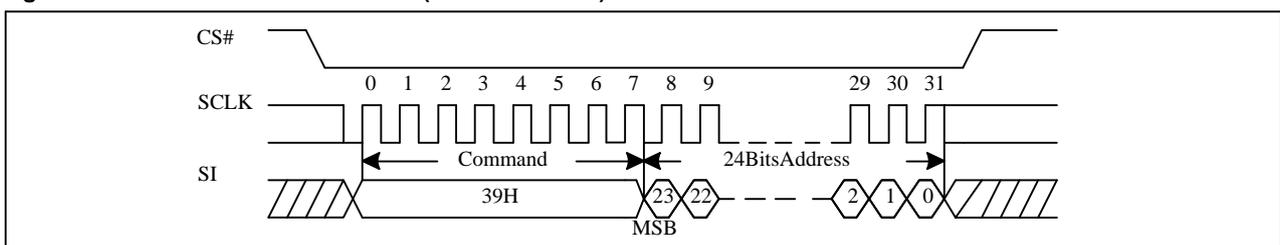
The write protection will be determined by the combination of CMP, BP[4:0] bits in the Status Registers. The Individual Block Lock bits are volatile bits. The default values after device power up or after a Reset are 1, so the entire memory array is being protected.

The SBULK instruction will cancel the block (or sector) write protection state using AMAX-A16 or (AMAX-A12) address bits to assign a 64Kbyte block (or 4K bytes sector) to be unprotected.

The WREN (Write Enable) instruction is required before issuing SBULK instruction.

The sequence of issuing SBULK instruction is: CS# goes low → send SBULK (39h) instruction → send 3-byte address assign one block (or sector) to be protected on SI pin → CS# goes high. The CS# must go high exactly at the byte boundary, otherwise the instruction will be rejected and not be executed.

Figure 10-51 Individual Block Unlock(Command 39H)



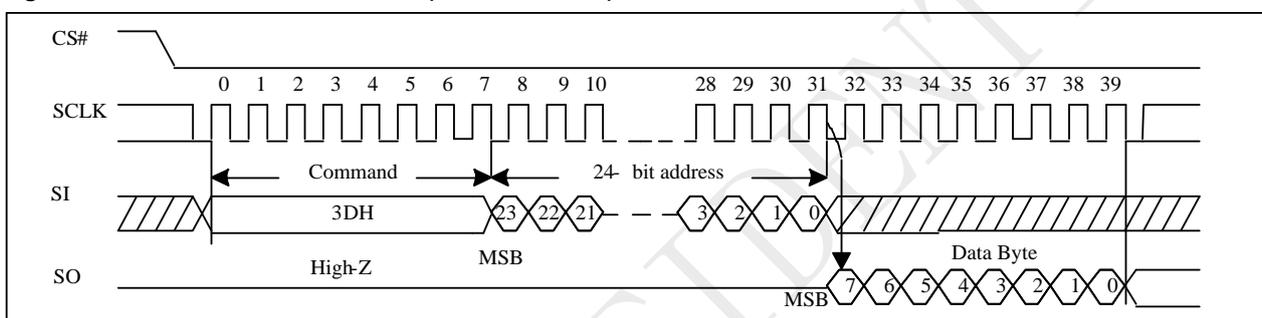
10.31 Read Block Lock Status (RDBLK)

The write protection will be determined by the combination of CMP, BP[4:0] bits in the Status Registers. The Individual Block Lock bits are volatile bits. The default values after device power up or after a Reset are 1, so the entire memory array is being protected.

The RDBLOCK instruction is for reading the status of protection lock of a specified block (or sector), using AMAX-A16 (or AMAX-A12) address bits to assign a 64K bytes block (4K bytes sector) and read protection lock status bit which the first byte of Read-out cycle. The status bit is "1" to indicate that this block has be protected, that user can read only but cannot write/program /erase this block. The status bit is "0" to indicate that this block hasn't be protected, and user can read and write this block.

The sequence of issuing RDBLOCK instruction is: CS# goes low → send RDBLOCK (3Dh) instruction → send 3-byte address to assign one block on SI pin → read block's protection lock status bit on SO pin → CS# goes high. SPI (8 clocks) command cycle can accept by this instruction.

Figure 10-52 Read Block Lock Status(Command 3DH)



10.32 Global Block Lock (GBLK)

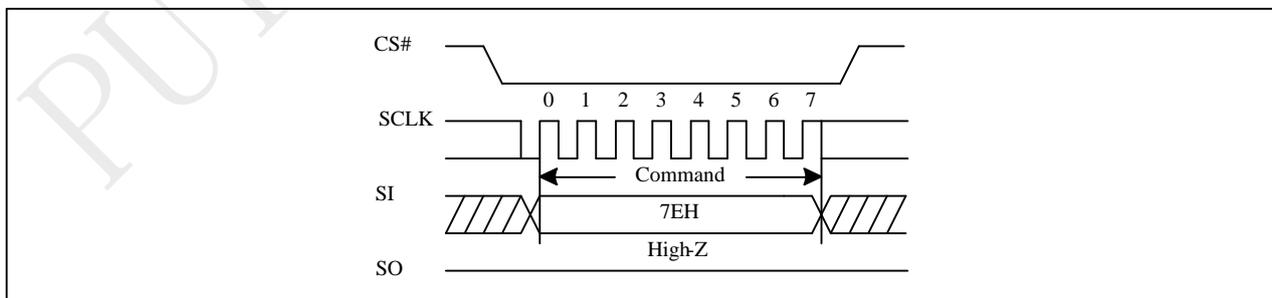
The GBLK instruction is for enable the lock protection block of the whole chip. The WREN (Write Enable) instruction is required before issuing GBLK instruction.

The sequence of issuing GBLK instruction is: CS# goes low → send GBLK (7Eh) instruction → CS# goes high.

SPI (8 clocks) command cycle can accept by this instruction. The SIO[3:1] are "don't care" in SPI mode.

The CS# must go high exactly at the byte boundary, otherwise, the instruction will be rejected and not be executed.

Figure 10-53 Global Block Lock(Command 7EH)

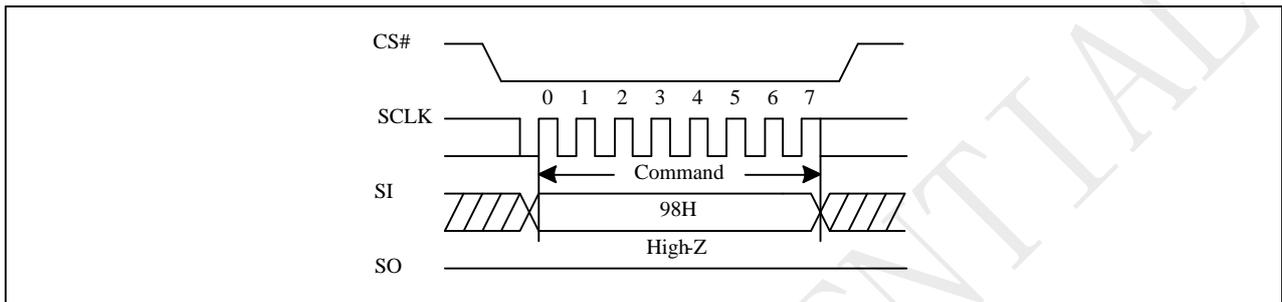


10.33 Global Block Unlock (GBULK)

The GBULK instruction is for disable the lock protection block of the whole chip. The WREN (Write Enable) instruction is required before issuing GBULK instruction. The sequence of issuing GBULK instruction is: CS# goes low → send GBULK (98h) instruction → CS# goes high. SPI (8 clocks) command cycle can accept by this instruction. The SIO[3:1] are "don't care" in SPI mode.

The CS# must go high exactly at the byte boundary, otherwise, the instruction will be rejected and not be executed.

Figure 10-54 Global Block Unlock(Command 98H)



10.34 Software Reset (RSTEN/RST)

The Software Reset operation combines two instructions: Reset-Enable (RSTEN) command and Reset (RST) command. It returns the device to a standby mode. All the volatile bits and settings will be cleared then, which makes the device return to the default status as power on, except the ep_fail bit.

To execute Reset command (RST), the Reset-Enable (RSTEN) command must be executed first to perform the Reset operation. If there is any other command to interrupt after the Reset-Enable command, the Reset-Enable will be invalid. The SIO[3:1] are "don't care".

If the Reset command is executed during program or erase operation, the operation will be disabled, the data under processing could be damaged or lost.

Figure 10-55 Software Reset Recovery

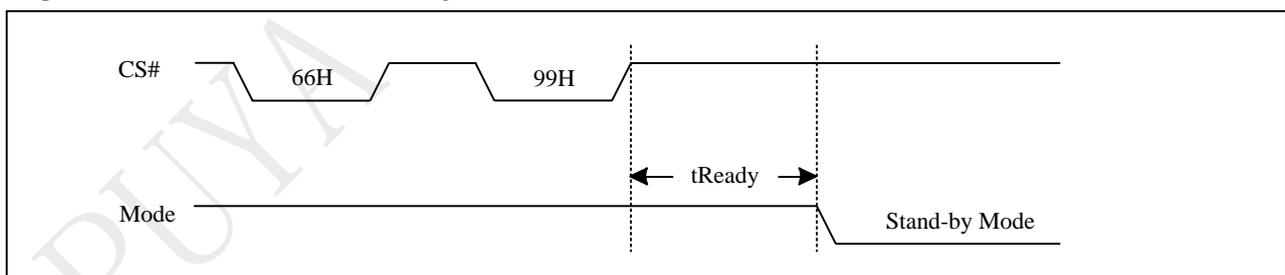
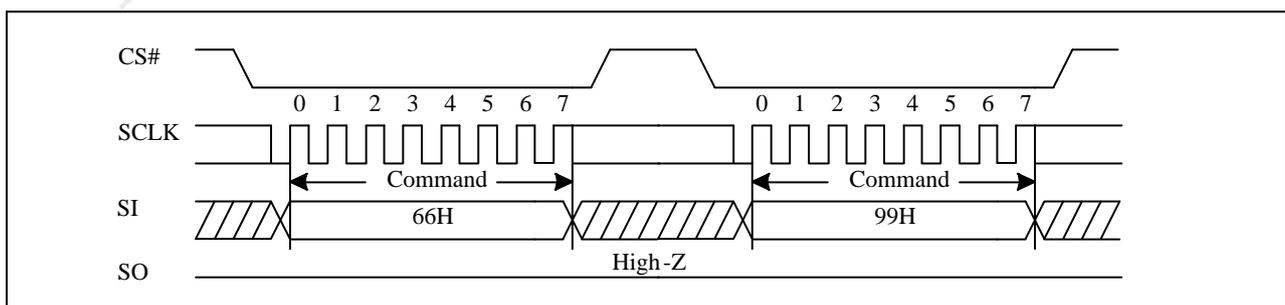


Figure 10-55a Reset Sequence



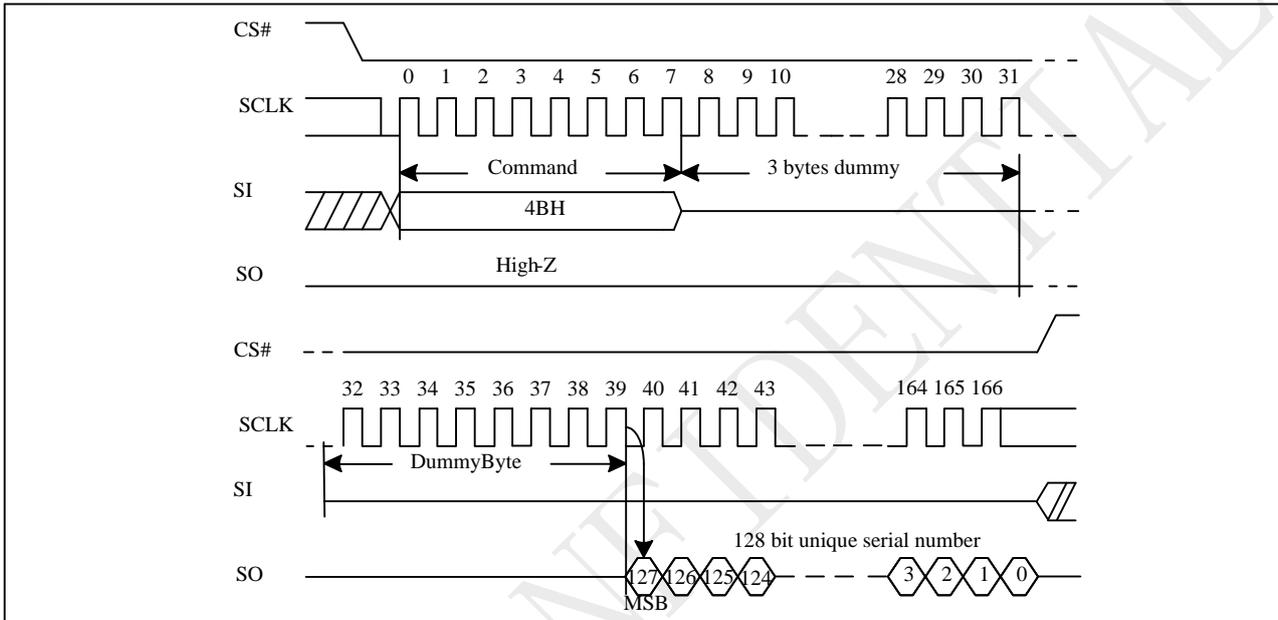
10.35 Read Unique ID (RUID)

The Read Unique ID command accesses a factory-set read-only 128bit number that is unique to each P25DSxx device. The Unique ID can be used in conjunction with user software methods to help prevent copying or cloning of a system.

The Read Unique ID command sequence: CS# goes low → sending Read Unique ID command → Dummy Byte1 → Dummy Byte2 → Dummy Byte3 → Dummy Byte4 → 128bit Unique ID Out → CS# goes high.

The command sequence is show below.

Figure 10-56 Read Unique ID (RUID) Sequence (Command 4B)



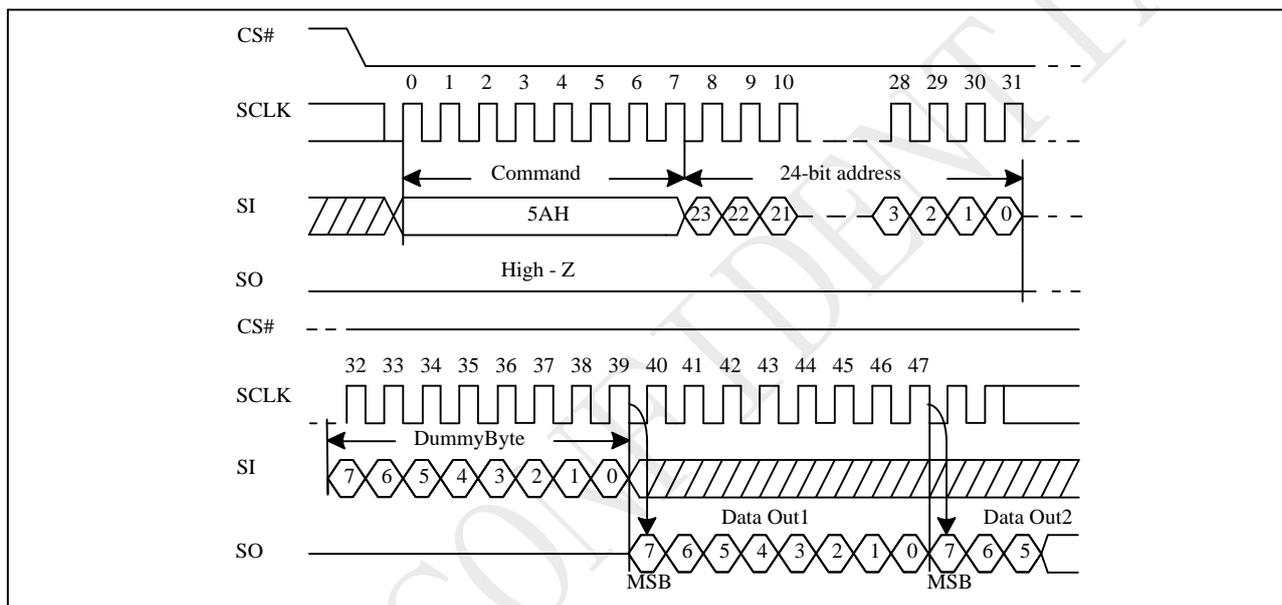
10.36 Read SFDP Mode (RDSFDP)

The Serial Flash Discoverable Parameter (SFDP) standard provides a consistent method of describing the functional and feature capabilities of serial flash devices in a standard set of internal parameter tables. These parameter tables can be interrogated by host system software to enable adjustments needed to accommodate divergent features from multiple vendors. The concept is similar to the one found in the Introduction of JEDEC Standard, JESD68 on CFI.

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

SFDP is a JEDEC Standard, JESD216B.

Figure 10-57 Read Serial Flash Discoverable Parameter (RDSFDP) Sequence



Serial Flash Discoverable Parameter (SFDP) Table

Table Signature and Parameter Identification Data Values

| Description | Comment | Add(H) (Byte) | DW Add (Bit) | Data | Data |
|---|--|------------------|-----------------|------|------|
| SFDP Signature | Fixed:50444653H | 00H | 07:00 | 53H | 53H |
| | | 01H | 15:08 | 46H | 46H |
| | | 02H | 23:16 | 44H | 44H |
| | | 03H | 31:24 | 50H | 50H |
| SFDP Minor Revision Number | Start from 00H | 04H | 07:00 | 00H | 00H |
| SFDP Major Revision Number | Start from 01H | 05H | 15:08 | 01H | 01H |
| Number of Parameters Headers | Start from 00H | 06H | 23:16 | 01H | 01H |
| Unused | Contains 0xFFH and can never be changed | 07H | 31:24 | FFH | FFH |
| ID number (JEDEC) | 00H: It indicates a JEDEC specified header | 08H | 07:00 | 00H | 00H |
| Parameter Table Minor Revision Number | Start from 0x00H | 09H | 15:08 | 00H | 00H |
| Parameter Table Major Revision Number | Start from 0x01H | 0AH | 23:16 | 01H | 01H |
| Parameter Table Length (in double word) | How many DWORDs in the Parameter table | 0BH | 31:24 | 09H | 09H |
| Parameter Table Pointer (PTP) | First address of JEDEC Flash Parameter table | 0CH | 07:00 | 30H | 30H |
| | | 0DH | 15:08 | 00H | 00H |
| | | 0EH | 23:16 | 00H | 00H |
| Unused | Contains 0xFFH and can never be changed | 0FH | 31:24 | FFH | FFH |
| ID Number (PUYA Device Manufacturer ID) | It indicates PUYA manufacturer ID | 10H | 07:00 | 85H | 85H |
| Parameter Table Minor Revision Number | Start from 0x00H | 11H | 15:08 | 00H | 00H |
| Parameter Table Major Revision Number | Start from 0x01H | 12H | 23:16 | 01H | 01H |
| Parameter Table Length (in double word) | How many DWORDs in the Parameter table | 13H | 31:24 | 03H | 03H |
| Parameter Table Pointer (PTP) | First address of PUYA Flash Parameter table | 14H | 07:00 | 60H | 60H |
| | | 15H | 15:08 | 00H | 00H |
| | | 16H | 23:16 | 00H | 00H |
| Unused | Contains 0xFFH and can never be changed | 17H | 31:24 | FFH | FFH |

Table Parameter Table (0): JEDEC Flash Parameter Tables

| Description | Comment | Add(H) (Byte) | DW Add (Bit) | Data | Data |
|---|--|------------------|-----------------|------------|------|
| Block/Sector Erase Size | 00: Reserved; 01: 4KB erase; 10: Reserved; 11: not support 4KB erase | 30H | 01:00 | 01b | E5H |
| Write Granularity | 0: 1Byte, 1: 64Byte or larger | | 02 | 1b | |
| Write Enable Instruction Requested for Writing to Volatile Status Registers | 0: Nonvolatile status bit 1: Volatile status bit (BP status register bit) | | 03 | 0b | |
| Write Enable Opcode Select for Writing to Volatile Status Registers | 0: Use 50H Opcode, 1: Use 06H Opcode, Note: If target flash status register is Nonvolatile, then bits3 and 4 must be set to 00b. | | 04 | 0b | |
| Unused | Contains 111b and can never be changed | | 07:05 | 111b | |
| 4KB Erase Opcode | | 31H | 15:08 | 20H | 20H |
| (1-1- 2) Fast Read | 0=Not support, 1=Support | 32H | 16 | 1b | 91H |
| Address Bytes Number used in addressing flash array | 00: 3Byte only, 01: 3 or 4Byte, 10: 4Byte only, 11: Reserved | | 18:17 | 00b | |
| Double Transfer Rate (DTR) clocking | 0=Not support, 1=Support | | 19 | 0b | |
| (1-2- 2) FastRead | 0=Not support, 1=Support | | 20 | 1b | |
| (1-4- 4) Fast Read | 0=Not support, 1=Support | | 21 | 0b | |
| (1-1- 4) Fast Read | 0=Not support, 1=Support | | 22 | 0b | |
| Unused | | | 23 | 1b | |
| Unused | | | 33H | 31:24 | |
| Flash Memory Density | | 37H:34H | 31:00 | 003FFFFFFH | |
| (1-4- 4) Fast Read Number of Wait states | 0 0000b: Wait states (Dummy Clocks) not support | 38H | 04:00 | 00000b | 00H |
| (1-4- 4) Fast Read Number of Mode Bits | 000b:Mode Bits not support | | 07:05 | 000b | |
| (1-4- 4) Fast Read Opcode | | 39H | 15:08 | FFH | FFH |
| (1-1- 4) Fast Read Number of Wait states | 0 0000b: Wait states (Dummy Clocks) not support | 3AH | 20:16 | 00000b | 00H |
| (1-1- 4) Fast Read Number of Mode Bits | 000b:Mode Bits not support | | 23:21 | 000b | |
| (1-1- 4) Fast Read Opcode | | 3BH | 31:24 | FFH | FFH |

| Description | Comment | Add(H) (Byte) | DW Add (Bit) | Data | Data |
|--|---|------------------|-----------------|--------|-------|
| (1-1-2) Fast Read Number of Wait states | 0 0000b: Wait states (Dummy Clocks) not support | 3CH | 04:00 | 01000b | 08H |
| (1-1- 2) Fast Read Number of Mode Bits | 000b: Mode Bits not support | | 07:05 | 000b | |
| (1-1- 2) Fast Read Opcode | | 3DH | 15:08 | 3BH | 3BH |
| (1-2- 2) Fast Read Number of Wait states | 0 0000b: Wait states (Dummy Clocks) not support | 3EH | 20:16 | 00000b | 80H |
| (1-2- 2) Fast Read Number of Mode Bits | 000b: Mode Bits not support | | 23:21 | 100b | |
| (1-2- 2) Fast Read Opcode | | 3FH | 31:24 | BBH | BBH |
| (2-2- 2) Fast Read | 0=not support 1=support | 40H | 00 | 0b | EEH |
| Unused | | | 03:01 | 111b | |
| (4-4- 4) Fast Read | 0=not support 1=support | | 04 | 0b | |
| Unused | | | 07:05 | 111b | |
| Unused | | 43H:41H | 31:08 | 0xFFH | 0xFFH |
| Unused | | 45H:44H | 15:00 | 0xFFH | 0xFFH |
| (2-2-2) Fast Read Number of Wait states | 0 0000b: Wait states (Dummy Clocks) not support | 46H | 20:16 | 00000b | 00H |
| (2-2- 2) Fast Read Number of Mode Bits | 000b: Mode Bits not support | | 23:21 | 000b | |
| (2-2- 2) Fast Read Opcode | | 47H | 31:24 | FFH | FFH |
| Unused | | 49H:48H | 15:00 | 0xFFH | 0xFFH |
| (4-4-4) Fast Read Number of Wait states | 0 0000b: Wait states (Dummy Clocks) not support | 4AH | 20:16 | 00000b | 00H |
| (4-4- 4) Fast Read Number of Mode Bits | 000b: Mode Bits not support | | 23:21 | 000b | |
| (4-4- 4) Fast Read Opcode | | 4BH | 31:24 | FFH | FFH |
| Sector Type 1 Size | Sector/block size=2 ^N bytes 0x00b: this sector type don't exist | 4CH | 07:00 | 0CH | 0CH |
| Sector Type 1 erase Opcode | | 4DH | 15:08 | 20H | 20H |
| Sector Type 2 Size | Sector/block size=2 ^N bytes 0x00b: this sector type don't exist | 4EH | 23:16 | 0FH | 0FH |
| Sector Type 2 erase Opcode | | 4FH | 31:24 | 52H | 52H |
| Sector Type 3 Size | Sector/block size=2 ^N bytes 0x00b: this sector type don't exist | 50H | 07:00 | 10H | 10H |
| Sector Type 3 erase Opcode | | 51H | 15:08 | D8H | D8H |
| Sector Type 4 Size | Sector/block size=2 ^N bytes 0x00b: this sector type don't exist | 52H | 23:16 | 08H | 08H |
| Sector Type 4 erase Opcode | | 53H | 31:24 | 81H | 81H |

Table Parameter Table (1): PUYA Flash Parameter Tables

| Description | Comment | Add(H) (Byte) | DW Add (Bit) | Data | Data | | |
|---|--|------------------|-----------------|---------------------|-------|-------|-------|
| Vcc Supply Maximum Voltage | 2000H=2.000V 2700H=2.700V 3600H=3.600V | 61H:60H | 15:00 | 3600H | 3600H | | |
| Vcc Supply Minimum Voltage | 1650H=1.650V 2250H=2.250V 2350H=2.350V 2700H=2.700V | 63H:62H | 31:16 | 2300H | 2300H | | |
| HW Reset# pin | 0=not support 1=support | 65H:64H | 00 | 0b | F99EH | | |
| HW Hold# pin | 0=not support 1=support | | 01 | 1b | | | |
| Deep Power Down Mode | 0=not support 1=support | | 02 | 1b | | | |
| SW Reset | 0=not support 1=support | | 03 | 1b | | | |
| SW Reset Opcode | Should be issue Reset Enable(66H) before Reset cmd. | | 11:04 | 1001 1001b (99H) | | | |
| Program Suspend/Resume | 0=not support 1=support | | 12 | 1b | | | |
| Erase Suspend/Resume | 0=not support 1=support | | 13 | 1b | | | |
| Unused | | | 14 | 1b | | | |
| Wrap Around Read mode | 0=not support 1=support | | 15 | 1b | | | |
| Wrap - Around Read mode Opcode | | | 66H | 23:16 | | 77H | 77H |
| Wrap - Around Read data length | 08H:support 8B wraparound read 16H:8B&16B 32H:8B&16B&32B 64H:8B&16B&32B&64B | 67H | 31:24 | 64H | 64H | | |
| Individual block lock | 0=not support 1=support | 6BH:68H | 00 | 1b | E8D9H | | |
| Individual block lock bit (Volatile/Nonvolatile) | 0=Volatile 1=Nonvolatile | | 01 | 0b | | | |
| Individual block lock Opcode | | | 09:02 | 36H | | | |
| Individual blocklock Volatile protect bit default protect status | 0=protect 1=unprotect | | 10 | 0b | | | |
| Secured OTP | 0=not support 1=support | | 11 | 1b | | | |
| Read Lock | 0=not support 1=support | | 12 | 0b | | | |
| Permanent Lock | 0=not support 1=support | | 13 | 1b | | | |
| Unused | | | 15:14 | 11b | | | |
| Unused | | | | 31:16 | | FFFFH | FFFFH |

11 Ordering Information

P 25 D 40S H A - SS H - C T - x

Company Designator

P = Puya Semiconductor

Product Family

25 = SPI interface flash

Product Serial

D = D serial

Memory Density

40S = new version 4MB

Operation Voltage

H = 2.3V~3.6V

Generation

A = A Version Default = blank

Package Type

SS = SOP8 150mil
WF = WAFER

UX= USON8 3x2x0.55mm

Plating Technology

H: RoHS Compliant Halogen free, Antimony free

Device Grade

C=0 ~ 70C

Packing Type

T = TUBE
R = TAPE & REEL
Y = TRAY
W = WAFER

Ordering Option

Default = blank

D = "01H" command with only 1byte data and "31H" command is valid

Note For shipments with ordering option "D", please contact PUYA sales.

12 Valid Part Numbers and Top Marking

The following table provides the valid part numbers for the P25D40SH Flash Memory. Please contact PUYA for specific availability by density and package type. PUYA Flash memories use a 14-digit Product Number for ordering.

4M bit Flash Valid Part Number

| Package Type | Product Number | Density | Top Side Marking | Temp. | Packing Type |
|--------------------|-----------------|---------|---------------------|-------|--------------|
| SS SOP8 150mil | P25D40SH-SSH-CT | 4M-bit | P25D40SH xxxxxxx | 70C | Tube |
| SS SOP8 150mil | P25D40SH-SSH-CR | 4M-bit | P25D40SH xxxxxxx | 70C | Reel |
| UX* USON8 3x2mm | P25D40SH-UXH-CR | 4M-bit | PD40S Hxxx | 70C | Reel |

Note: There is no mass production of the package marked with "*", if necessary, please contact Puya sales

13 Package Information

13.1 8-Lead SOP(150mil)

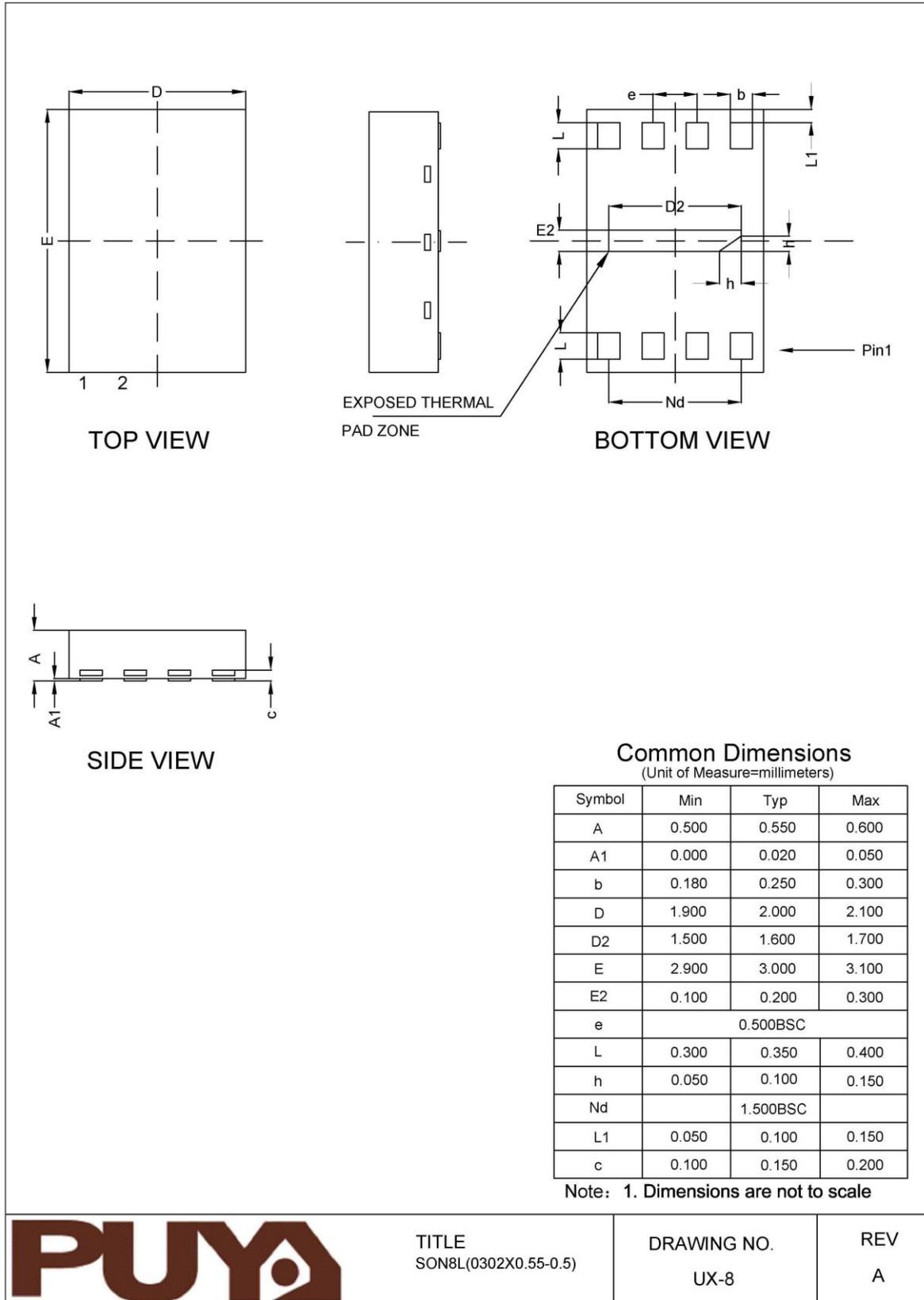
Common Dimensions
(Unit of Measure=millimeters)

| Symbol | Min | Typ | Max |
|--------|-------|-------|-------|
| A | - | - | 1.750 |
| A1 | 0.100 | - | 0.250 |
| A2 | 1.250 | - | - |
| b | 0.280 | - | 0.480 |
| c | 0.170 | - | 0.230 |
| D | 4.800 | 4.900 | 5.000 |
| E | 5.800 | 6.000 | 6.200 |
| E1 | 3.800 | 3.900 | 4.000 |
| e | - | 1.270 | - |
| h | 0.250 | - | 0.500 |
| k | 0° | - | 8° |
| L | 0.400 | - | 1.270 |
| L1 | - | 1.040 | - |

Note: 1. Dimensions are not to scale

| | | | |
|--|------------|-------------|-----|
| | TITLE | DRAWING NO. | REV |
| | 8-lead SOP | SP-8 | A |

13.2 8-Land USON(3x2x0.55mm)



14 Revision History

| Rev. | Date | Description | Note |
|------|------------|---|------|
| V1.0 | 2020-02-22 | 1.0 datasheet | - |
| V1.1 | 2020-12-02 | Update memory type. | - |
| V1.2 | 2021-02-02 | Delete 2M 1M 512K configuration information | - |
| V1.3 | 2021-04-02 | Update Command function | - |



Puya Semiconductor Co., Ltd.

IMPORTANT NOTICE

Puya Semiconductor reserves the right to make changes without further notice to any products or specifications herein. Puya Semiconductor does not assume any responsibility for use of any its products for any particular purpose, nor does Puya Semiconductor assume any liability arising out of the application or use of any its products or circuits. Puya Semiconductor does not convey any license under its patent rights or other rights nor the rights of others.