# 4 Mbit SPI Serial Flash
## SST25VF040B

***Data Sheet***

## FEATURES:

- **Single Voltage Read and Write Operations**
  - 2.7-3.6V
- **Serial Interface Architecture**
  - SPI Compatible: Mode 0 and Mode 3
- **High Speed Clock Frequency**
  - 50 MHz
- **Superior Reliability**
  - Endurance: 100,000 Cycles (typical)
  - Greater than 100 years Data Retention
- **Low Power Consumption:**
  - Active Read Current: 10 mA (typical)
  - Standby Current: 5 µA (typical)
- **Flexible Erase Capability**
  - Uniform 4 KByte sectors
  - Uniform 32 KByte overlay blocks
  - Uniform 64 KByte overlay blocks
- **Fast Erase and Byte-Program:**
  - Chip-Erase Time: 35 ms (typical)
  - Sector-/Block-Erase Time: 18 ms (typical)
  - Byte-Program Time: 7 µs (typical)
- **Auto Address Increment (AAI) Programming**
  - Decrease total chip programming time over Byte-Program operations
- **End-of-Write Detection**
  - Software polling the BUSY bit in Status Register
  - Busy Status readout on SO pin in AAI Mode
- **Hold Pin (HOLD#)**
  - Suspends a serial sequence to the memory without deselecting the device
- **Write Protection (WP#)**
  - Enables/Disables the Lock-Down function of the status register
- **Software Write Protection**
  - Write protection through Block-Protection bits in status register
- **Temperature Range**
  - Commercial: 0°C to +70°C
  - Industrial: -40°C to +85°C
- **Packages Available**
  - 8-lead SOIC (200 mils)
  - 8-lead SOIC (150 mils)
  - 8-contact WSON (6mm x 5mm)

## PRODUCT DESCRIPTION

The 25 series Serial Flash family features a four-wire, SPI-compatible interface that allows for a low pin-count package which occupies less board space and ultimately lowers total system costs. The SST25VF040B devices are enhanced with improved operating frequency and even lower power consumption than the original SST25VFxxxA devices. SST25VF040B SPI serial flash memories are manufactured with SST proprietary, high-performance CMOS SuperFlash technology. The split-gate cell design and thick-oxide tunneling injector attain better reliability and manufacturability compared with alternate approaches.

The SST25VF040B devices significantly improve performance and reliability, while lowering power consumption. The devices write (Program or Erase) with a single power supply of 2.7-3.6V for SST25VF040B. The total energy consumed is a function of the applied voltage, current, and time of application. Since for any given voltage range, the SuperFlash technology uses less current to program and has a shorter erase time, the total energy consumed during any Erase or Program operation is less than alternative flash memory technologies.

The SST25VF040B device is offered in an 8-lead SOIC (200 mils), 8-lead SOIC (150 mils), and 8-contact WSON (6mm x 5mm) packages. See Figure 2 for pin assignments.

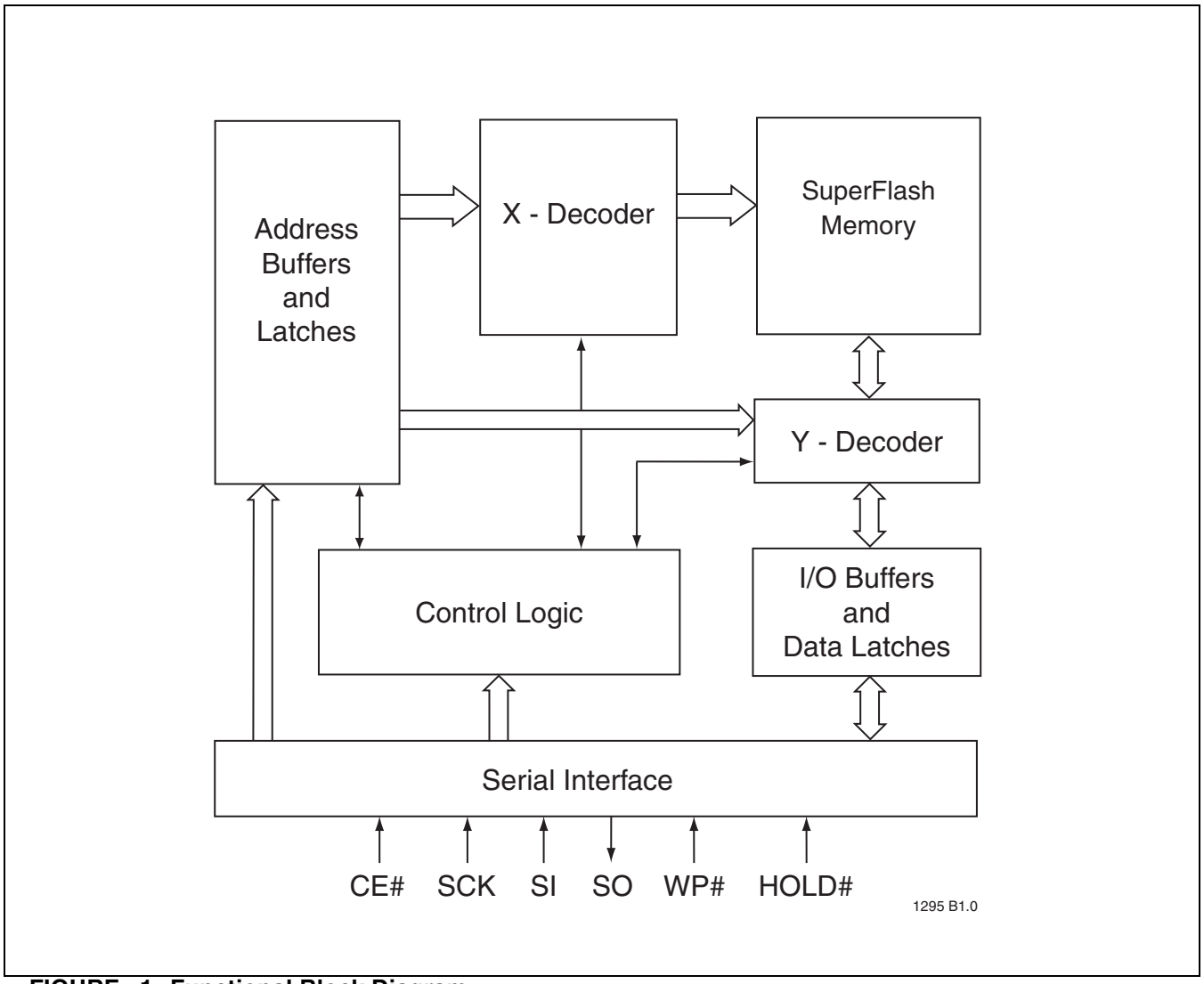



**FIGURE 1: Functional Block Diagram**

## PIN DESCRIPTION

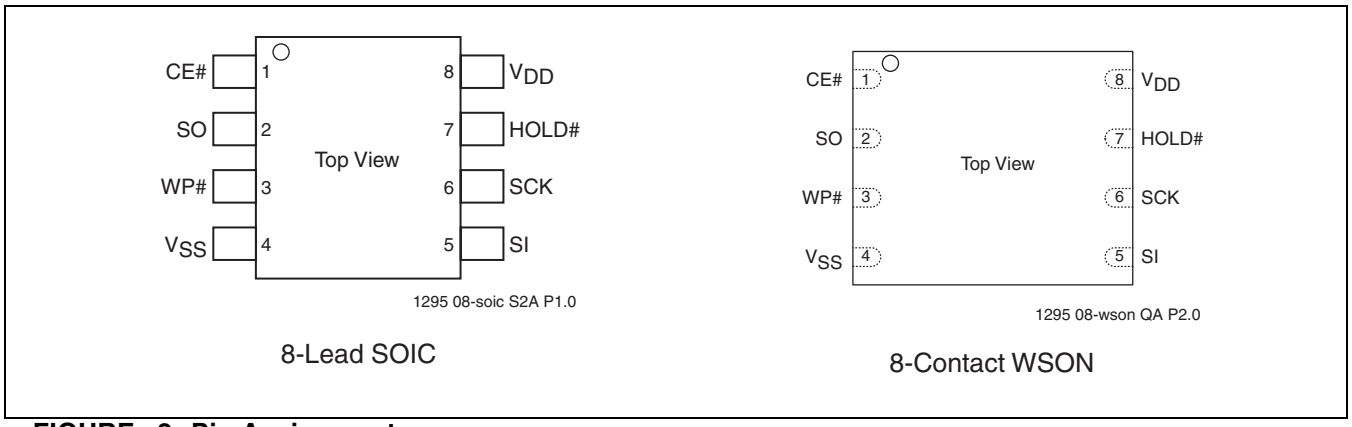



**FIGURE 2: Pin Assignments**

**TABLE 1: Pin Description**

| Symbol | Pin Name | Functions |
|---|---|---|
| SCK | Serial Clock | To provide the timing of the serial interface.<br>Commands, addresses, or input data are latched on the rising edge of the clock input, while output data is shifted out on the falling edge of the clock input. |
| SI | Serial Data Input | To transfer commands, addresses, or data serially into the device.<br>Inputs are latched on the rising edge of the serial clock. |
| SO | Serial Data Output | To transfer data serially out of the device.<br>Data is shifted out on the falling edge of the serial clock.<br>Outputs Flash busy status during AAI Programming when reconfigured as RY/BY# pin.<br>See “Hardware End-of-Write Detection” on page 12 for details. |
| CE# | Chip Enable | The device is enabled by a high to low transition on CE#. CE# must remain low for the duration of any command sequence. |
| WP# | Write Protect | The Write Protect (WP#) pin is used to enable/disable BPL bit in the status register. |
| HOLD# | Hold | To temporarily stop serial communication with SPI flash memory without resetting the device. |
| $V_{DD}$ | Power Supply | To provide power supply voltage: 2.7-3.6V for SST25VF040B |
| $V_{SS}$ | Ground | |

T1.0 1295

## MEMORY ORGANIZATION

The SST25VF040B SuperFlash memory array is organized in uniform 4 KByte erasable sectors with 32 KByte overlay blocks and 64 KByte overlay erasable blocks.

## DEVICE OPERATION

The SST25VF040B is accessed through the SPI (Serial Peripheral Interface) bus compatible protocol. The SPI bus consist of four control lines; Chip Enable (CE#) is used to select the device, and data is accessed through the Serial Data Input (SI), Serial Data Output (SO), and Serial Clock (SCK).

The SST25VF040B supports both Mode 0 (0,0) and Mode 3 (1,1) of SPI bus operations. The difference between the two modes, as shown in Figure 3, is the state of the SCK signal when the bus master is in Stand-by mode and no data is being transferred. The SCK signal is low for Mode 0 and SCK signal is high for Mode 3. For both modes, the Serial Data In (SI) is sampled at the rising edge of the SCK clock signal and the Serial Data Output (SO) is driven after the falling edge of the SCK clock signal.

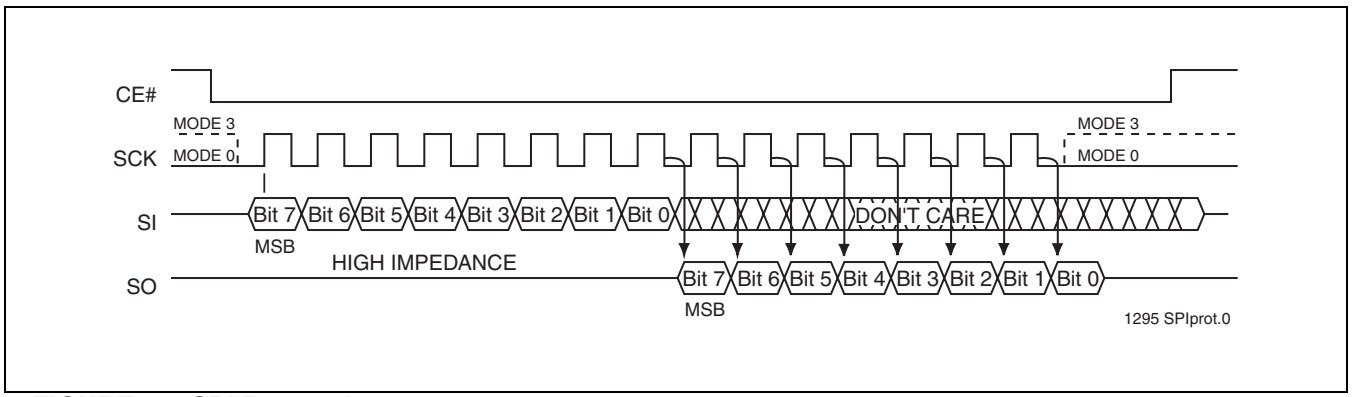



**FIGURE 3: SPI Protocol**

## Hold Operation

The HOLD# pin is used to pause a serial sequence underway with the SPI flash memory without resetting the clocking sequence. To activate the HOLD# mode, CE# must be in active low state. The HOLD# mode begins when the SCK active low state coincides with the falling edge of the HOLD# signal. The HOLD mode ends when the HOLD# signal's rising edge coincides with the SCK active low state.

If the falling edge of the HOLD# signal does not coincide with the SCK active low state, then the device enters Hold mode when the SCK next reaches the active low state. Similarly, if the rising edge of the HOLD# signal does not coincide with the SCK active low state, then the device exits in Hold mode when the SCK next reaches the active low state. See Figure 4 for Hold Condition waveform.

Once the device enters Hold mode, SO will be in high-impedance state while SI and SCK can be $V_{IL}$ or $V_{IH}$.

If CE# is driven active high during a Hold condition, it resets the internal logic of the device. As long as HOLD# signal is low, the memory remains in the Hold condition. To resume communication with the device, HOLD# must be driven active high, and CE# must be driven active low. See Figure 24 for Hold timing.

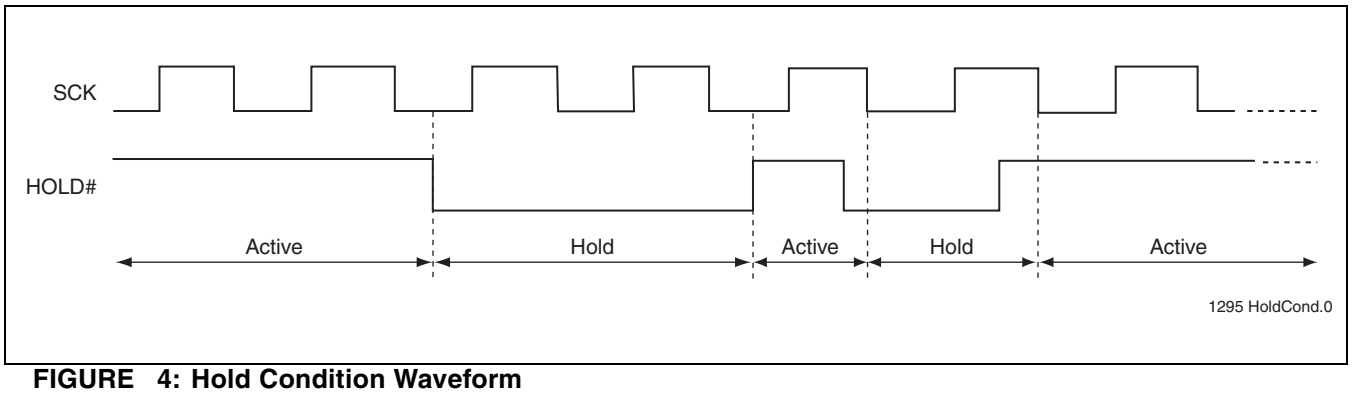



**FIGURE 4: Hold Condition Waveform**

## Write Protection

SST25VF040B provides software Write protection. The Write Protect pin (WP#) enables or disables the lock-down function of the status register. The Block-Protection bits (BP3, BP2, BP1, BP0, and BPL) in the status register provide Write protection to the memory array and the status register. See Table 4 for the Block-Protection description.

### Write Protect Pin (WP#)

The Write Protect (WP#) pin enables the lock-down function of the BPL bit (bit 7) in the status register. When WP# is driven low, the execution of the Write-Status-Register (WRSR) instruction is determined by the value of the BPL bit (see Table 2). When WP# is high, the lock-down function of the BPL bit is disabled.

**TABLE 2: Conditions to execute Write-Status-Register (WRSR) Instruction**

| WP# | BPL | Execute WRSR Instruction |
|---|---|---|
| L | 1 | Not Allowed |
| L | 0 | Allowed |
| H | X | Allowed |

T2.0 1295

## Status Register

The software status register provides status on whether the flash memory array is available for any Read or Write operation, whether the device is Write enabled, and the state of the Memory Write protection. During an internal Erase or Program operation, the status register may be read only to determine the completion of an operation in progress. Table 3 describes the function of each bit in the software status register.

**TABLE 3: Software Status Register**

| Bit | Name | Function | Default at Power-up | Read/Write |
|---|---|---|---|---|
| 0 | BUSY | 1 = Internal Write operation is in progress<br>0 = No internal Write operation is in progress | 0 | R |
| 1 | WEL | 1 = Device is memory Write enabled<br>0 = Device is not memory Write enabled | 0 | R |
| 2 | BP0 | Indicate current level of block write protection (See Table 4) | 1 | R/W |
| 3 | BP1 | Indicate current level of block write protection (See Table 4) | 1 | R/W |
| 4 | BP2 | Indicate current level of block write protection (See Table 4) | 1 | R/W |
| 5 | BP3 | Indicate current level of block write protection (See Table 4) | 0 | R/W |
| 6 | AAI | Auto Address Increment Programming status<br>1 = AAI programming mode<br>0 = Byte-Program mode | 0 | R |
| 7 | BPL | 1 = BP3, BP2, BP1, BP0 are read-only bits<br>0 = BP3, BP2, BP1, BP0 are read/writable | 0 | R/W |

T3.0 1295

### Busy

The Busy bit determines whether there is an internal Erase or Program operation in progress. A “1” for the Busy bit indicates the device is busy with an operation in progress. A “0” indicates the device is ready for the next valid operation.

### Write Enable Latch (WEL)

The Write-Enable-Latch bit indicates the status of the internal memory Write Enable Latch. If the Write-Enable-Latch bit is set to “1”, it indicates the device is Write enabled. If the bit is set to “0” (reset), it indicates the device is not Write enabled and does not accept any memory Write (Program/Erase) commands. The Write-Enable-Latch bit is automatically reset under the following conditions:

- Power-up
- Write-Disable (WRDI) instruction completion
- Byte-Program instruction completion
- Auto Address Increment (AAI) programming is completed or reached its highest unprotected memory address
- Sector-Erase instruction completion
- Block-Erase instruction completion
- Chip-Erase instruction completion
- Write-Status-Register instructions

### Auto Address Increment (AAI)

The Auto Address Increment Programming-Status bit provides status on whether the device is in AAI programming mode or Byte-Program mode. The default at power up is Byte-Program mode.

### Block Protection (BP3,BP2, BP1, BP0)

The Block-Protection (BP3, BP2, BP1, BP0) bits define the size of the memory area, as defined in Table 4, to be software protected against any memory Write (Program or Erase) operation. The Write-Status-Register (WRSR) instruction is used to program the BP3, BP2, BP1 and BP0 bits as long as WP# is high or the Block-Protect-Lock (BPL) bit is 0. Chip-Erase can only be executed if Block-Protection bits are all 0. After power-up, BP3, BP2, BP1 and BP0 are set to 1.

### Block Protection Lock-Down (BPL)

WP# pin driven low ($V_{IL}$), enables the Block-Protection-Lock-Down (BPL) bit. When BPL is set to 1, it prevents any further alteration of the BPL, BP3, BP2, BP1, and BP0 bits. When the WP# pin is driven high ($V_{IH}$), the BPL bit has no effect and its value is "Don't Care". After power-up, the BPL bit is reset to 0.

**TABLE 4: Software Status Register Block Protection FOR SST25VF040B[1]**

| | Status Register Bit[2] | | | | Protected Memory Address |
|---|---|---|---|---|---|
| **Protection Level** | **BP3** | **BP2** | **BP1** | **BP0** | **4 Mbit** |
| None | X | 0 | 0 | 0 | None |
| Upper 1/8 | X | 0 | 0 | 1 | 70000H-7FFFFH |
| Upper 1/4 | X | 0 | 1 | 0 | 60000H-7FFFFH |
| Upper 1/2 | X | 0 | 1 | 1 | 40000H-7FFFFH |
| All Blocks | X | 1 | 0 | 0 | 00000H-7FFFFH |
| All Blocks | X | 1 | 0 | 1 | 00000H-7FFFFH |
| All Blocks | X | 1 | 1 | 0 | 00000H-7FFFFH |
| All Blocks | X | 1 | 1 | 1 | 00000H-7FFFFH |

T4.0 1295

1. X = Don't Care (RESERVED) default is "0
2. Default at power-up for BP2, BP1, and BP0 is '111'. (All Blocks Protected)

## Instructions

Instructions are used to read, write (Erase and Program), and configure the SST25VF040B. The instruction bus cycles are 8 bits each for commands (Op Code), data, and addresses. Prior to executing any Byte-Program, Auto Address Increment (AAI) programming, Sector-Erase, Block-Erase, Write-Status-Register, or Chip-Erase instructions, the Write-Enable (WREN) instruction must be executed first. The complete list of instructions is provided in Table 5. All instructions are synchronized off a high to low transition of CE#. Inputs will be accepted on the rising edge of SCK starting with the most significant bit. CE# must be driven low before an instruction is entered and must be driven high after the last bit of the instruction has been shifted in (except for Read, Read-ID, and Read-Status-Register instructions). Any low to high transition on CE#, before receiving the last bit of an instruction bus cycle, will terminate the instruction in progress and return the device to standby mode. Instruction commands (Op Code), addresses, and data are all input from the most significant bit (MSB) first.

**TABLE 5: Device Operation Instructions**

| Instruction | Description | Op Code Cycle[1] | Address Cycle(s)[2] | Dummy Cycle(s) | Data Cycle(s) | Maximum Frequency |
|---|---|---|---|---|---|---|
| Read | Read Memory at 25 MHz | 0000 0011b (03H) | 3 | 0 | 1 to ∞ | 25 MHz |
| High-Speed Read | Read Memory at 50 MHz | 0000 1011b (0BH) | 3 | 1 | 1 to ∞ | 50 MHz |
| 4 KByte Sector-Erase[3] | Erase 4 KByte of memory array | 0010 0000b (20H) | 3 | 0 | 0 | 50 MHz |
| 32 KByte Block-Erase[4] | Erase 32 KByte block of memory array | 0101 0010b (52H) | 3 | 0 | 0 | 50 MHz |
| 64 KByte Block-Erase[5] | Erase 64 KByte block of memory array | 1101 1000b (D8H) | 3 | 0 | 0 | 50 MHz |
| Chip-Erase | Erase Full Memory Array | 0110 0000b (60H) or 1100 0111b (C7H) | 0 | 0 | 0 | 50 MHz |
| Byte-Program | To Program One Data Byte | 0000 0010b (02H) | 3 | 0 | 1 | 50 MHz |
| AAI-Word-Program[6] | Auto Address Increment Programming | 1010 1101b (ADH) | 3 | 0 | 2 to ∞ | 50 MHz |
| RDSR[7] | Read-Status-Register | 0000 0101b (05H) | 0 | 0 | 1 to ∞ | 50 MHz |
| EWSR | Enable-Write-Status-Register | 0101b 0000b (50H) | 0 | 0 | 0 | 50 MHz |
| WRSR | Write-Status-Register | 0000 0001b (01H) | 0 | 0 | 1 | 50 MHz |
| WREN | Write-Enable | 0000 0110b (06H) | 0 | 0 | 0 | 50 MHz |
| WRDI | Write-Disable | 0000 0100b (04H) | 0 | 0 | 0 | 50 MHz |
| RDID[8] | Read-ID | 1001 0000b (90H) or 1010 1011b (ABH) | 3 | 0 | 1 to ∞ | 50 MHz |
| JEDEC-ID | JEDEC ID read | 1001 1111b (9FH) | 0 | 0 | 3 to ∞ | 50 MHz |
| EBSY | Enable SO to output RY/BY# status during AAI programming | 0111 0000b (70H) | 0 | 0 | 0 | 50 MHz |
| DBSY | Disable SO to output RY/BY# status during AAI programming | 1000 0000b (80H) | 0 | 0 | 0 | 50 MHz |

T5.0 1295

1. One bus cycle is eight clock periods.
2. Address bits above the most significant bit of each density can be $V_{IL}$ or $V_{IH}$.
3. 4KByte Sector Erase addresses: use $A_{MS}$-$A_{12}$, remaining addresses are don't care but must be set either at $V_{IL}$ or $V_{IH}$.
4. 32KByte Block Erase addresses: use $A_{MS}$-$A_{15}$, remaining addresses are don't care but must be set either at $V_{IL}$ or $V_{IH}$.
5. 64KByte Block Erase addresses: use $A_{MS}$-$A_{16}$, remaining addresses are don't care but must be set either at $V_{IL}$ or $V_{IH}$.
6. To continue programming to the next sequential address location, enter the 8-bit command, ADH, followed by 2 bytes of data to be programmed. Data Byte 0 will be programmed into the initial address [$A_{23}$-$A_1$] with $A_0$=0, Data Byte 1 will be programmed into the initial address [$A_{23}$-$A_1$] with $A_0$=1.
7. The Read-Status-Register is continuous with ongoing clock cycles until terminated by a low to high transition on CE#.
8. Manufacturer's ID is read with $A_0$=0, and Device ID is read with $A_0$=1. All other address bits are 00H. The Manufacturer's ID and device ID output stream is continuous until terminated by a low-to-high transition on CE#.

### Read (25 MHz)

The Read instruction, 03H, supports up to 25 MHz Read. The device outputs the data starting from the specified address location. The data output stream is continuous through all addresses until terminated by a low to high transition on CE#. The internal address pointer will automatically increment until the highest memory address is reached. Once the highest memory address is reached, the address pointer will automatically increment to the beginning (wrap-around) of the address space. Once the data from address location 1FFFFFH has been read, the next output will be from address location 000000H.

The Read instruction is initiated by executing an 8-bit command, 03H, followed by address bits [$A_{23}$-$A_0$]. CE# must remain active low for the duration of the Read cycle. See Figure 5 for the Read sequence.

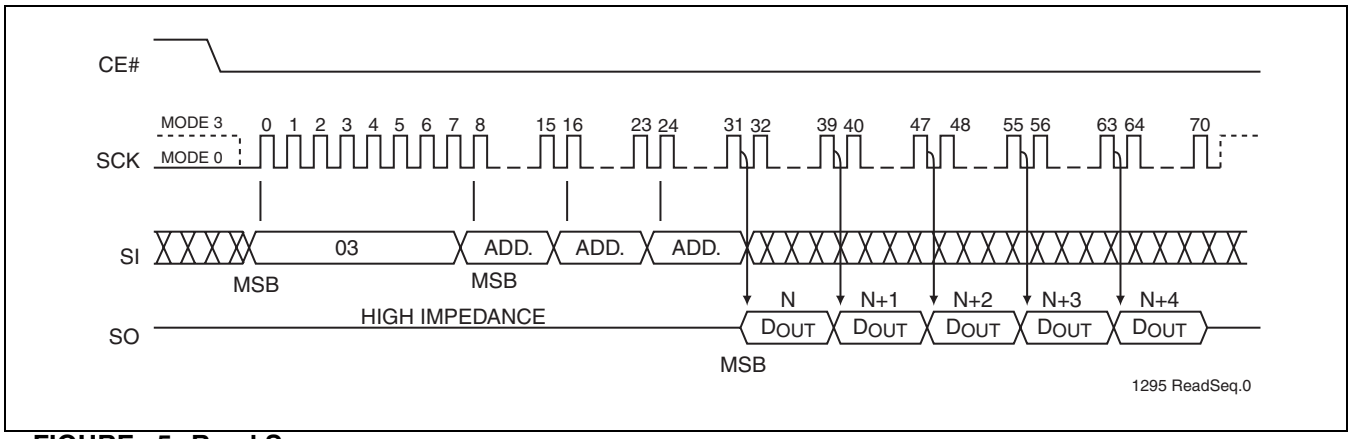



**FIGURE 5: Read Sequence**

### High-Speed-Read (50 MHz)

The High-Speed-Read instruction supporting up to 50 MHz Read is initiated by executing an 8-bit command, 0BH, followed by address bits [$A_{23}$-$A_0$] and a dummy byte. CE# must remain active low for the duration of the High-Speed-Read cycle. See Figure 6 for the High-Speed-Read sequence.

Following a dummy cycle, the High-Speed-Read instruction outputs the data starting from the specified address location. The data output stream is continuous through all addresses until terminated by a low to high transition on CE#. The internal address pointer will automatically increment until the highest memory address is reached. Once the highest memory address is reached, the address pointer will automatically increment to the beginning (wrap-around) of the address space. Once the data from address location 7FFFFH has been read, the next output will be from address location 00000H.

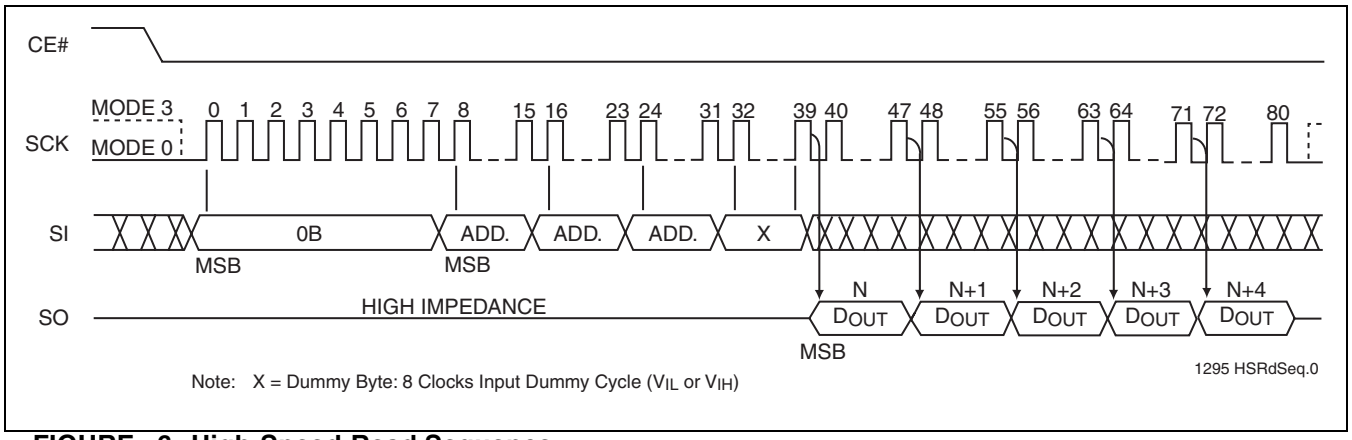



**FIGURE 6: High-Speed-Read Sequence**

### Byte-Program

The Byte-Program instruction programs the bits in the selected byte to the desired data. The selected byte must be in the erased state (FFH) when initiating a Program operation. A Byte-Program instruction applied to a protected memory area will be ignored.

Prior to any Write operation, the Write-Enable (WREN) instruction must be executed. CE# must remain active low for the duration of the Byte-Program instruction. The Byte-Program instruction is initiated by executing an 8-bit command, 02H, followed by address bits [$A_{23}$-$A_0$]. Following the address, the data is input in order from MSB (bit 7) to LSB (bit 0). CE# must be driven high before the instruction is executed. The user may poll the Busy bit in the software status register or wait $T_{BP}$ for the completion of the internal self-timed Byte-Program operation. See Figure 7 for the Byte-Program sequence.

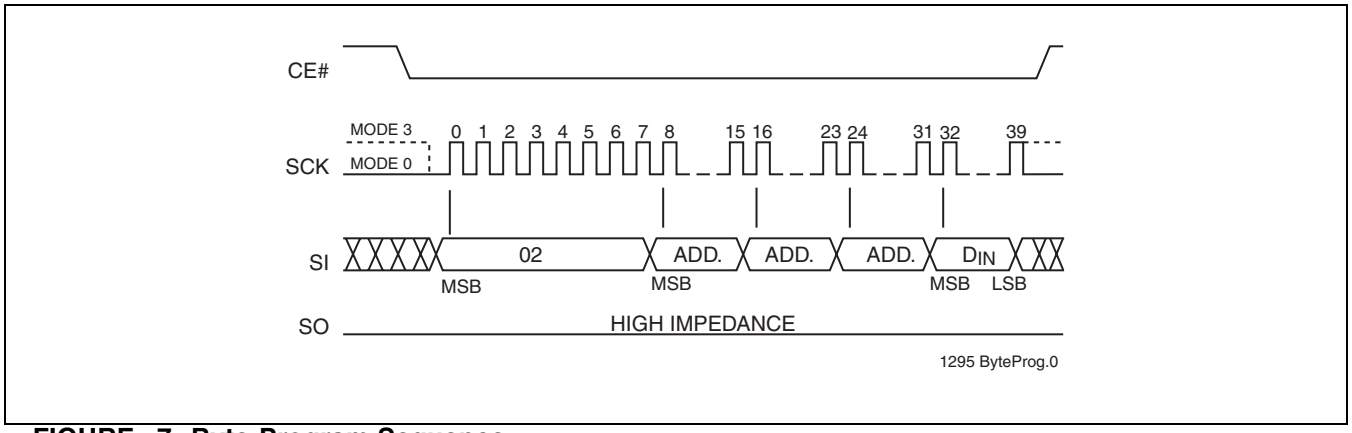



**FIGURE 7: Byte-Program Sequence**

### Auto Address Increment (AAI) Word-Program

The AAI program instruction allows multiple bytes of data to be programmed without re-issuing the next sequential address location. This feature decreases total programming time when multiple bytes or entire memory array is to be programmed. An AAI Word program instruction pointing to a protected memory area will be ignored. The selected address range must be in the erased state (FFH) when initiating an AAI Word Program operation. While within AAI Word Programming sequence, the only valid instructions are AAI Word (ADH), RDSR (05H), or WRDI (04H). Users have three options to determine the completion of each AAI Word program cycle: hardware detection by reading the Serial Output, software detection by polling the BUSY bit in the software status register or wait $T_{BP}$. Refer to End-Of-Write Detection section for details.

Prior to any write operation, the Write-Enable (WREN) instruction must be executed. The AAI Word Program instruction is initiated by executing an 8-bit command, ADH, followed by address bits [$A_{23}$-$A_0$]. Following the addresses, two bytes of data is input sequentially, each one from MSB (Bit 7) to LSB (Bit 0). The first byte of data (D0) will be programmed into the initial address [$A_{23}$-$A_1$] with $A_0$=0, the second byte of Data (D1) will be programmed into the initial address [$A_{23}$-$A_1$] with $A_0$=1. CE# must be driven high before the AAI Word Program instruction is executed. The user must check the BUSY status before entering the next valid command. Once the device indicates it is no longer busy, data for the next two sequential addresses may be programmed and so on. When the last desired byte had been entered, check the busy status using the hardware method or the RDSR instruction and execute the Write-Disable (WRDI) instruction, 04H, to terminate AAI. User must check busy status after WRDI to determine if the device is ready for any command. See Figures 10 and 11 for AAI Word programming sequence.

There is no wrap mode during AAI programming; once the highest unprotected memory address is reached, the device will exit AAI operation and reset the Write-Enable-Latch bit (WEL = 0) and the AAI bit (AAI=0).

**FIGURE 8: Enable SO as Hardware RY/BY# during AAI Programming**

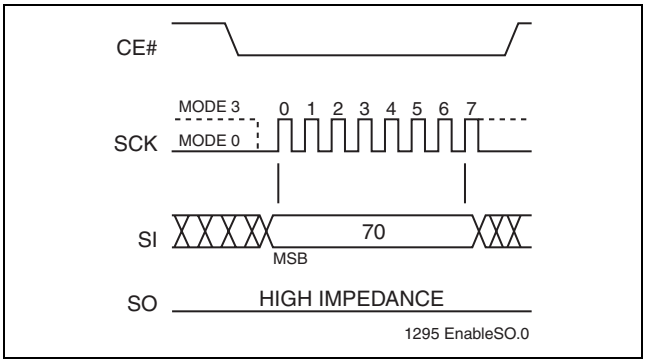
### End-of-Write Detection

There are three methods to determine completion of a program cycle during AAI Word programming: hardware detection by reading the Serial Output, software detection by polling the BUSY bit in the Software Status Register or wait $T_{BP}$. The hardware end-of-write detection method is described in the section below.

### Hardware End-of-Write Detection

The hardware end-of-write detection method eliminates the overhead of polling the Busy bit in the Software Status Register during an AAI Word program operation. The 8-bit command, 70H, configures the Serial Output (SO) pin to indicate Flash Busy status during AAI Word programming. (see Figure 8) The 8-bit command, 70H, must be executed prior to executing an AAI Word-Program instruction. Once an internal programming operation begins, asserting CE# will immediately drive the status of the internal flash status on the SO pin. A “0” indicates the device is busy and a “1” indicates the device is ready for the next instruction. De-asserting CE# will return the SO pin to tri-state.

The 8-bit command, 80H, disables the Serial Output (SO) pin to output busy status during AAI-Word-program operation and return SO pin to output Software Status Register data during AAI Word programming. (see Figure 9)



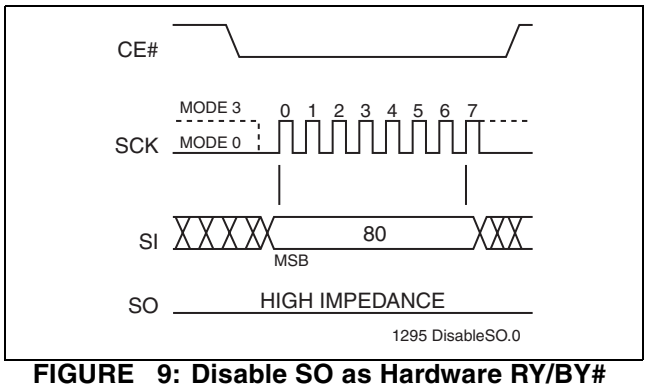



**FIGURE 9: Disable SO as Hardware RY/BY# during AAI Programming**

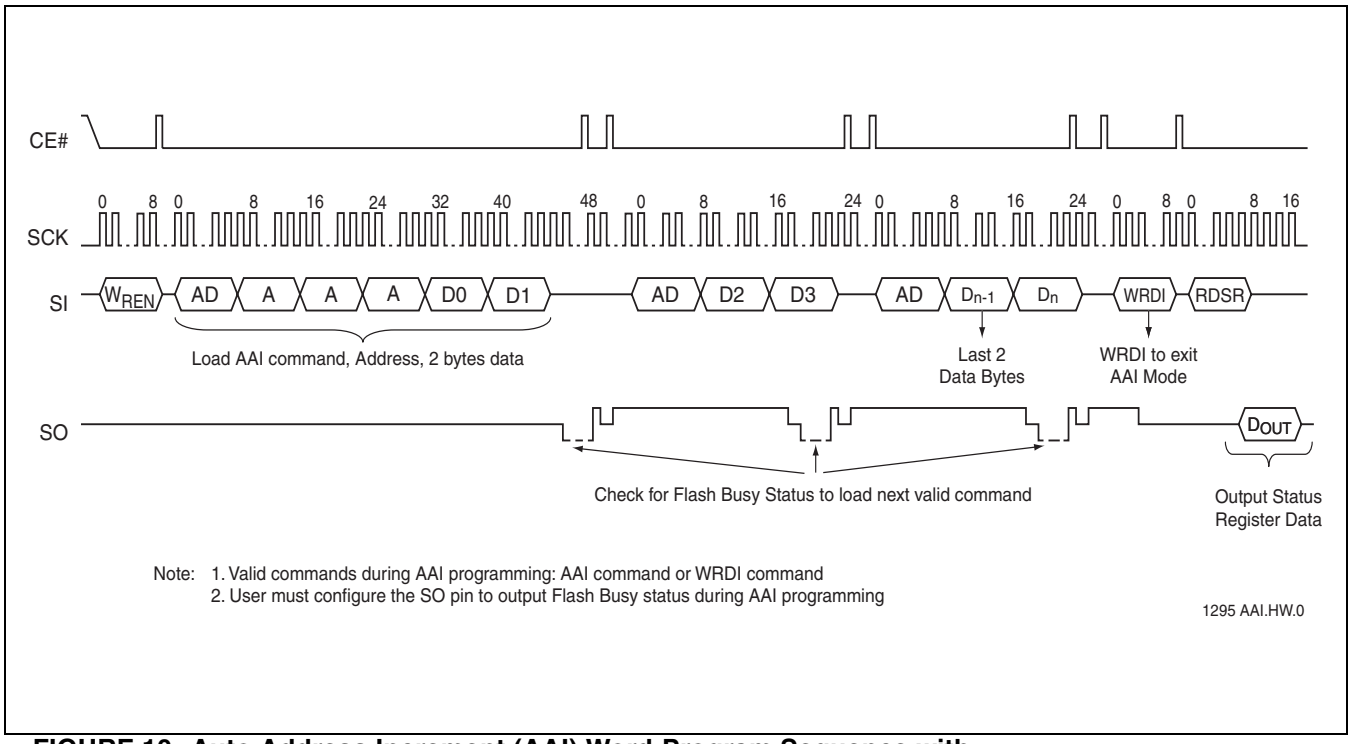



**FIGURE 10: Auto Address Increment (AAI) Word-Program Sequence with Hardware End-of-Write Detection**

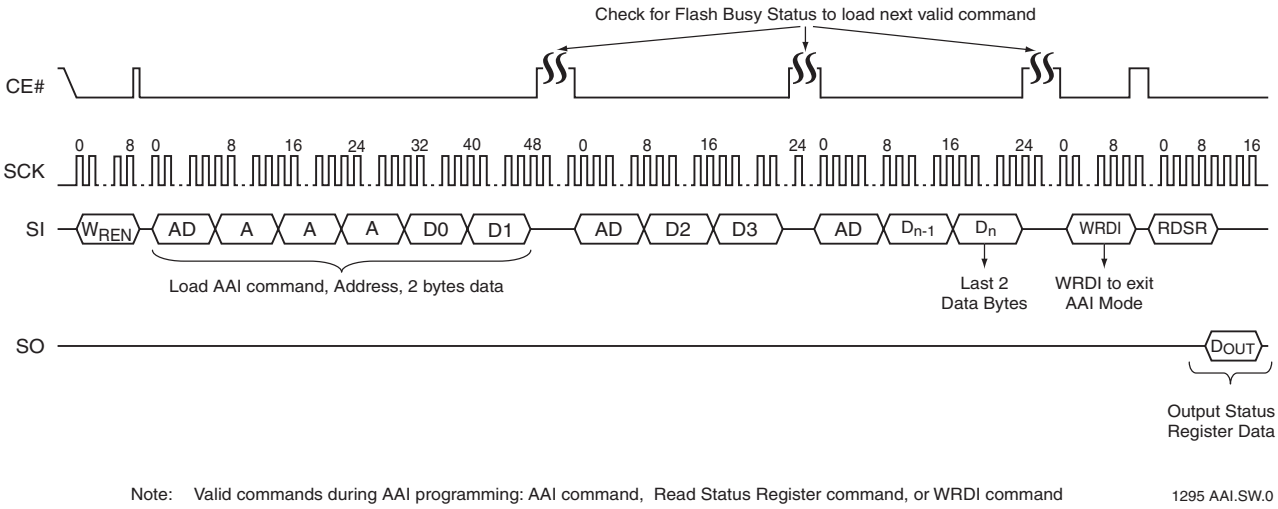



**FIGURE 11: Auto Address Increment (AAI) Word-Program Sequence with Software End-of-Write Detection**

### 4-KByte Sector-Erase

The Sector-Erase instruction clears all bits in the selected 4 KByte sector to FFH. A Sector-Erase instruction applied to a protected memory area will be ignored. Prior to any Write operation, the Write-Enable (WREN) instruction must be executed. CE# must remain active low for the duration of any command sequence. The Sector-Erase instruction is initiated by executing an 8-bit command, 20H, followed by address bits [$A_{23}$-$A_0$]. Address bits [$A_{MS}$-$A_{12}$] ($A_{MS}$ = Most Significant address) are used to determine the sector address ($SA_X$), remaining address bits can be $V_{IL}$ or $V_{IH}$. CE# must be driven high before the instruction is executed. The user may poll the Busy bit in the software status register or wait $T_{SE}$ for the completion of the internal self-timed Sector-Erase cycle. See Figure 12 for the Sector-Erase sequence.

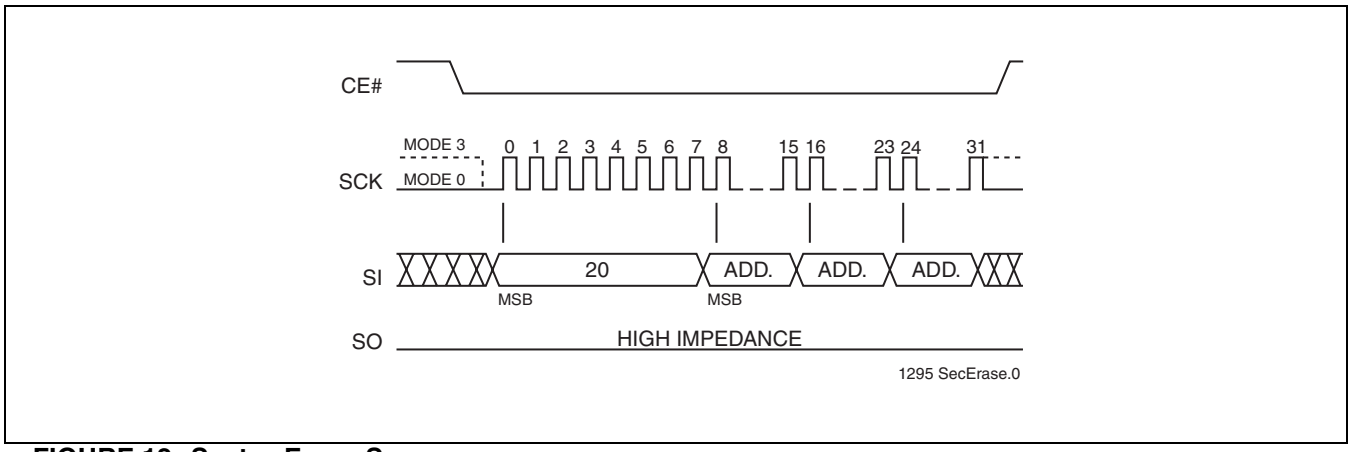



**FIGURE 12: Sector-Erase Sequence**

### 32-KByte and 64-KByte Block-Erase

The 32-KByte Block-Erase instruction clears all bits in the selected 32 KByte block to FFH. A Block-Erase instruction applied to a protected memory area will be ignored. The 64-KByte Block-Erase instruction clears all bits in the selected 64 KByte block to FFH. A Block-Erase instruction applied to a protected memory area will be ignored. Prior to any Write operation, the Write-Enable (WREN) instruction must be executed. CE# must remain active low for the duration of any command sequence. The 32-Kbyte Block-Erase instruction is initiated by executing an 8-bit command, 52H, followed by address bits [$A_{23}$-$A_0$]. Address bits [$A_{MS}$-$A_{15}$] ($A_{MS}$ = Most Significant Address) are used to determine block address ($BA_X$), remaining address bits can be $V_{IL}$ or $V_{IH}$. CE# must be driven high before the instruction is executed. The 64-Kbyte Block-Erase instruction is initiated by executing an 8-bit command D8H, followed by address bits [$A_{23}$-$A_0$]. Address bits [$A_{MS}$-$A_{15}$] are used to determine block address ($BA_X$), remaining address bits can be $V_{IL}$ or $V_{IH}$. CE# must be driven high before the instruction is executed. The user may poll the Busy bit in the software status register or wait $T_{BE}$ for the completion of the internal self-timed 32-KByte Block-Erase or 64-KByte Block-Erase cycles. See Figures 13 and 14 for the 32-KByte Block-Erase and 64-KByte Block-Erase sequences.

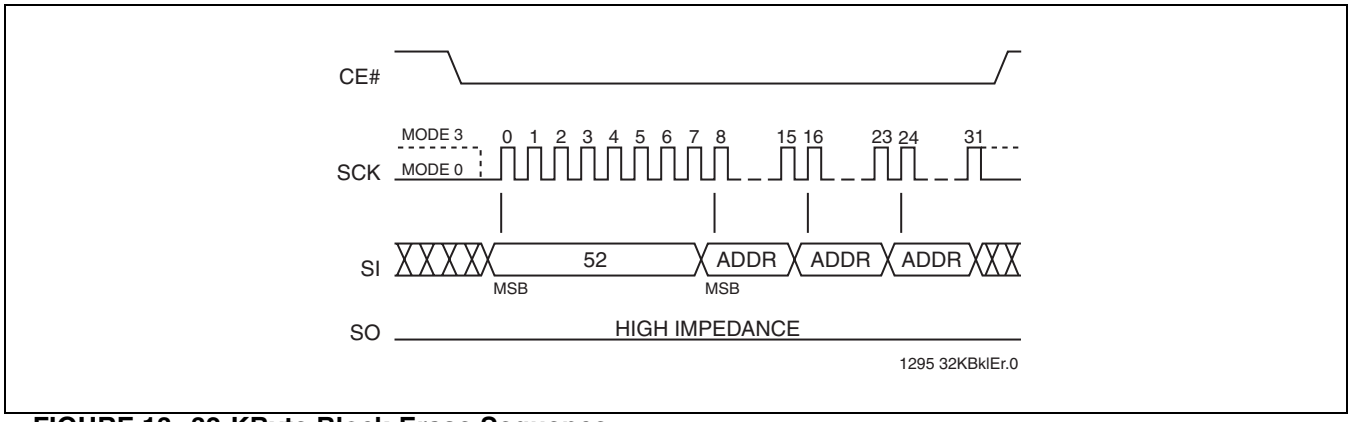



**FIGURE 13: 32-KByte Block-Erase Sequence**

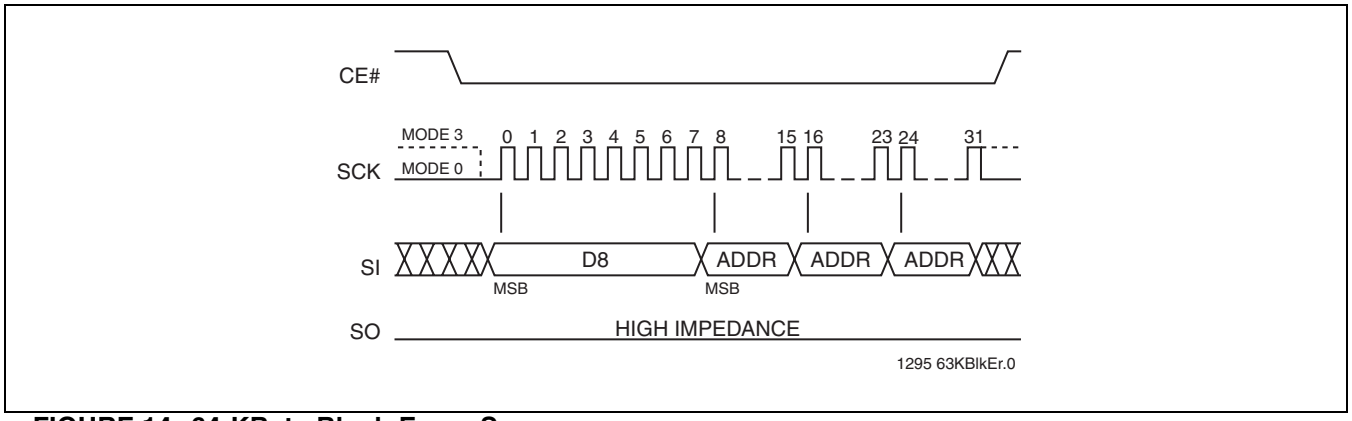



**FIGURE 14: 64-KByte Block-Erase Sequence**

### Chip-Erase

The Chip-Erase instruction clears all bits in the device to FFH. A Chip-Erase instruction will be ignored if any of the memory area is protected. Prior to any Write operation, the Write-Enable (WREN) instruction must be executed. CE# must remain active low for the duration of the Chip-Erase instruction sequence. The Chip-Erase instruction is initiated by executing an 8-bit command, 60H or C7H. CE# must be driven high before the instruction is executed. The user may poll the Busy bit in the software status register or wait $T_{CE}$ for the completion of the internal self-timed Chip-Erase cycle. See Figure 15 for the Chip-Erase sequence.

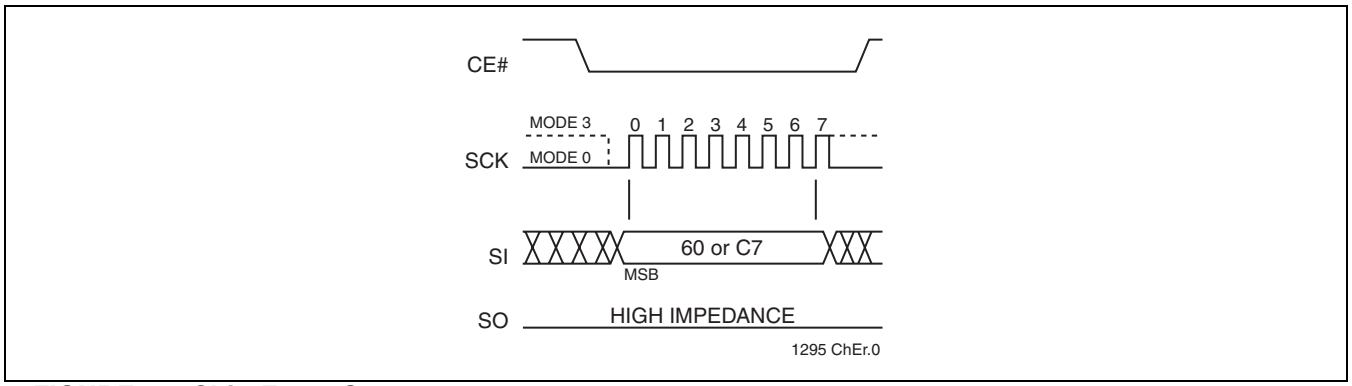



**FIGURE 15: Chip-Erase Sequence**

### Read-Status-Register (RDSR)

The Read-Status-Register (RDSR) instruction allows reading of the status register. The status register may be read at any time even during a Write (Program/Erase) operation. When a Write operation is in progress, the Busy bit may be checked before sending any new commands to assure that the new commands are properly received by the device. CE# must be driven low before the RDSR instruction is entered and remain low until the status data is read. Read-Status-Register is continuous with ongoing clock cycles until it is terminated by a low to high transition of the CE#. See Figure 16 for the RDSR instruction sequence.

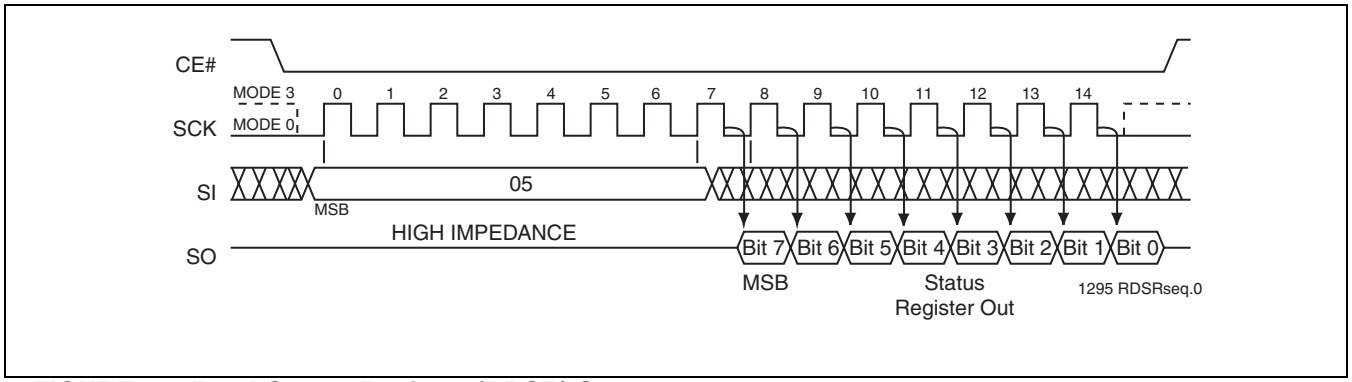



**FIGURE 16: Read-Status-Register (RDSR) Sequence**

### Write-Enable (WREN)

The Write-Enable (WREN) instruction sets the Write-Enable-Latch bit in the Status Register to 1 allowing Write operations to occur. The WREN instruction must be executed prior to any Write (Program/Erase) operation. The WREN instruction may also be used to allow execution of the Write-Status-Register (WRSR) instruction; however, the Write-Enable-Latch bit in the Status Register will be cleared upon the rising edge CE# of the WRSR instruction. CE# must be driven high before the WREN instruction is executed.

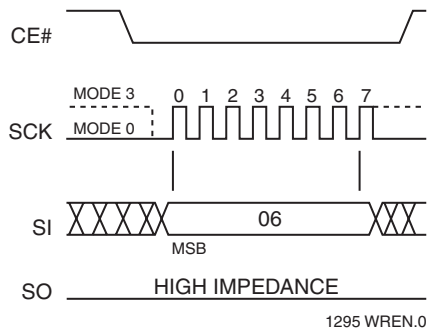



**FIGURE 17: Write Enable (WREN) Sequence**

### Write-Disable (WRDI)

The Write-Disable (WRDI) instruction resets the Write-Enable-Latch bit and AAI bit to 0 disabling any new Write operations from occurring. The WRDI instruction will not terminate any programming operation in progress. Any program operation in progress may continue up to $T_{BP}$ after executing the WRDI instruction. CE# must be driven high before the WRDI instruction is executed.

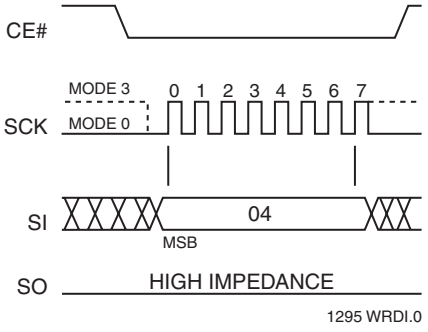



**FIGURE 18: Write Disable (WRDI) Sequence**

### Enable-Write-Status-Register (EWSR)

The Enable-Write-Status-Register (EWSR) instruction arms the Write-Status-Register (WRSR) instruction and opens the status register for alteration. The Write-Status-Register instruction must be executed immediately after the execution of the Enable-Write-Status-Register instruction. This two-step instruction sequence of the EWSR instruction followed by the WRSR instruction works like SDP (software data protection) command structure which prevents any accidental alteration of the status register values. CE# must be driven low before the EWSR instruction is entered and must be driven high before the EWSR instruction is executed.

### Write-Status-Register (WRSR)

The Write-Status-Register instruction writes new values to the BP3, BP2, BP1, BP0, and BPL bits of the status register. CE# must be driven low before the command sequence of the WRSR instruction is entered and driven high before the WRSR instruction is executed. See Figure 19 for EWSR or WREN and WRSR instruction sequences.

Executing the Write-Status-Register instruction will be ignored when WP# is low and BPL bit is set to "1". When the WP# is low, the BPL bit can only be set from "0" to "1" to lock-down the status register, but cannot be reset from "1" to "0". When WP# is high, the lock-down function of the BPL bit is disabled and the BPL, BP0, and BP1 and BP2 bits in the status register can all be changed. As long as BPL bit is set to 0 or WP# pin is driven high ($V_{IH}$) prior to the low-to-high transition of the CE# pin at the end of the WRSR instruction, the bits in the status register can all be altered by the WRSR instruction. In this case, a single WRSR instruction can set the BPL bit to "1" to lock down the status register as well as altering the BP0, BP1, and BP2 bits at the same time. See Table 2 for a summary description of WP# and BPL functions.

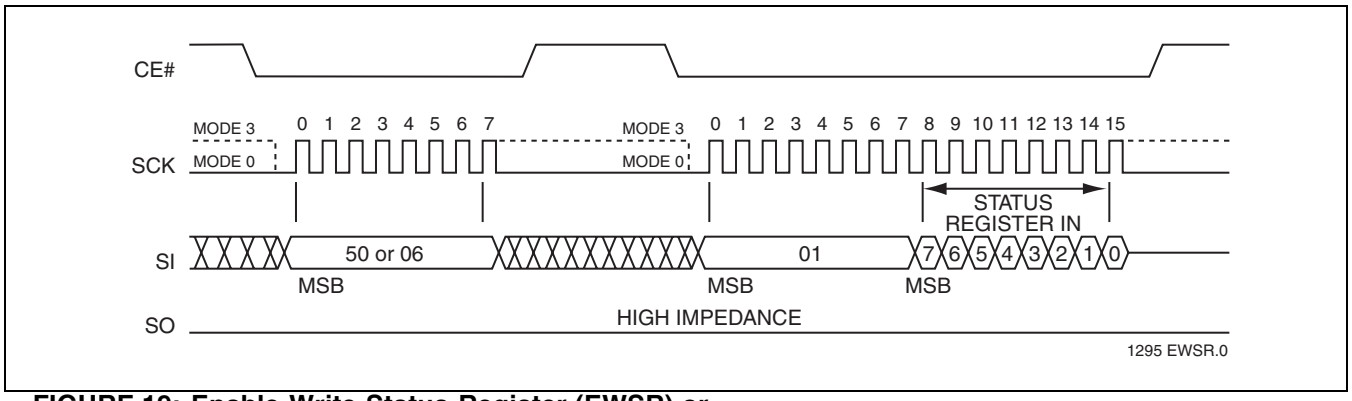



**FIGURE 19: Enable-Write-Status-Register (EWSR) or Write-Enable (WREN) and Write-Status-Register (WRSR) Sequence**

### JEDEC Read-ID

The JEDEC Read-ID instruction identifies the device as SST25VF040B and the manufacturer as SST. The device information can be read from executing the 8-bit command, 9FH. Following the JEDEC Read-ID instruction, the 8-bit manufacturer's ID, BFH, is output from the device. After that, a 16-bit device ID is shifted out on the SO pin. Byte 1, BFH, identifies the manufacturer as SST. Byte 2, 25H, identifies the memory type as SPI Serial Flash. Byte 3, 8DH, identifies the device as SST25VF040B. The instruction sequence is shown in Figure 20. The JEDEC Read ID instruction is terminated by a low to high transition on CE# at any time during data output.

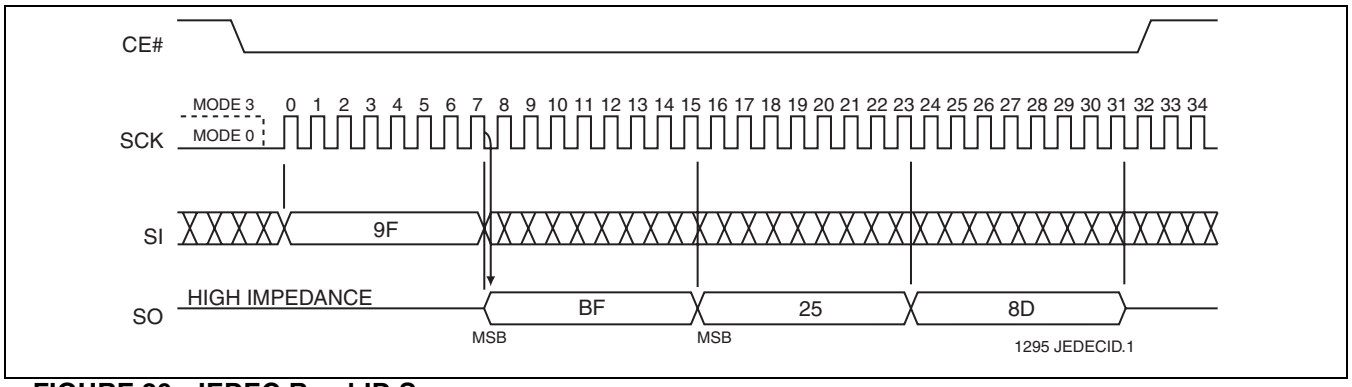



**FIGURE 20: JEDEC Read-ID Sequence**

**TABLE 6: JEDEC Read-ID Data**

| Manufacturer's ID | Device ID | |
|---|---|---|
| | Memory Type | Memory Capacity |
| Byte1 | Byte 2 | Byte 3 |
| BFH | 25H | 8DH |

T6.0 1295

### Read-ID (RDID)

The Read-ID instruction (RDID) identifies the devices as SST25VF040B and manufacturer as SST. This command is backward compatible to all SST25xFxxxA devices and should be used as default device identification when multiple versions of SPI Serial Flash devices are used in a design. The device information can be read from executing an 8-bit command, 90H or ABH, followed by address bits [$A_{23}$-$A_0$]. Following the Read-ID instruction, the manufacturer's ID is located in address 00000H and the device ID is located in address 00001H. Once the device is in Read-ID mode, the manufacturer's and device ID output data toggles between address 00000H and 00001H until terminated by a low to high transition on CE#.

Refer to Tables 6 and 7 for device identification data.

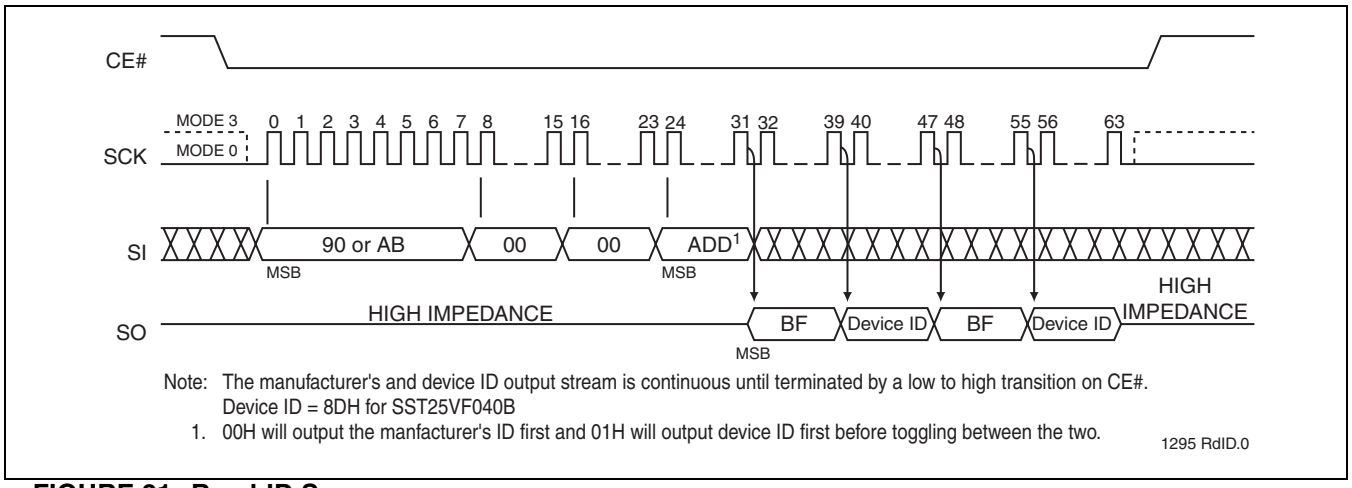



**FIGURE 21: Read-ID Sequence**

**TABLE 7: Product Identification**

| | Address | Data |
|---|---|---|
| Manufacturer's ID | 00000H | BFH |
| Device ID | | |
| SST25VF040B | 00001H | 8DH |

T7.0 1295

## ELECTRICAL SPECIFICATIONS

**Absolute Maximum Stress Ratings** (Applied conditions greater than those listed under "Absolute Maximum Stress Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these conditions or conditions greater than those defined in the operational sections of this data sheet is not implied. Exposure to absolute maximum stress rating conditions may affect device reliability.)

Temperature Under Bias . . . . . -55°C to +125°C
Storage Temperature . . . . . -65°C to +150°C
D. C. Voltage on Any Pin to Ground Potential . . . . . -0.5V to $V_{DD}$+0.5V
Transient Voltage (<20 ns) on Any Pin to Ground Potential . . . . . -2.0V to $V_{DD}$+2.0V
Package Power Dissipation Capability ($T_A$ = 25°C) . . . . . 1.0W
Surface Mount Solder Reflow Temperature . . . . . 260°C for 10 seconds
Output Short Circuit Current[1] . . . . . 50 mA

1. Output shorted for no more than one second. No more than one output shorted at a time.

**OPERATING RANGE**

| Range | Ambient Temp | $V_{DD}$ |
|---|---|---|
| Commercial | 0°C to +70°C | 2.7-3.6V |
| Industrial | -40°C to +85°C | 2.7-3.6V |

**AC CONDITIONS OF TEST**

| | |
|---|---|
| Input Rise/Fall Time | 5 ns |
| Output Load | $C_L$ = 30 pF |
| See Figures 26 and 27 | |

**TABLE 8: DC Operating Characteristics**

| Symbol | Parameter | Limits | | | Test Conditions |
|---|---|---|---|---|---|
| | | Min | Max | Units | |
| $I_{DDR}$ | Read Current | | 10 | mA | CE#=0.1 $V_{DD}$/0.9 $V_{DD}$@25 MHz, SO=open |
| $I_{DDR2}$ | Read Current | | 15 | mA | CE#=0.1 $V_{DD}$/0.9 $V_{DD}$@50 MHz, SO=open |
| $I_{DDW}$ | Program and Erase Current | | 30 | mA | CE#=$V_{DD}$ |
| $I_{SB}$ | Standby Current | | 20 | µA | CE#=$V_{DD}$, $V_{IN}$=$V_{DD}$ or $V_{SS}$ |
| $I_{LI}$ | Input Leakage Current | | 1 | µA | $V_{IN}$=GND to $V_{DD}$, $V_{DD}$=$V_{DD}$ Max |
| $I_{LO}$ | Output Leakage Current | | 1 | µA | $V_{OUT}$=GND to $V_{DD}$, $V_{DD}$=$V_{DD}$ Max |
| $V_{IL}$ | Input Low Voltage | | 0.8 | V | $V_{DD}$=$V_{DD}$ Min |
| $V_{IH}$ | Input High Voltage | 0.7 $V_{DD}$ | | V | $V_{DD}$=$V_{DD}$ Max |
| $V_{OL}$ | Output Low Voltage | | 0.2 | V | $I_{OL}$=100 µA, $V_{DD}$=$V_{DD}$ Min |
| $V_{OL2}$ | Output Low Voltage | | 0.4 | V | $I_{OL}$=1.6 mA, $V_{DD}$=$V_{DD}$ Min |
| $V_{OH}$ | Output High Voltage | $V_{DD}$-0.2 | | V | $I_{OH}$=-100 µA, $V_{DD}$=$V_{DD}$ Min |

T8.0 1295

**TABLE 9: Recommended System Power-up Timings**

| Symbol | Parameter | Minimum | Units |
|---|---|---|---|
| $T_{PU-READ}$[1] | $V_{DD}$ Min to Read Operation | 10 | µs |
| $T_{PU-WRITE}$[1] | $V_{DD}$ Min to Write Operation | 10 | µs |

T9.0 1295

1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.

**TABLE 10: Capacitance ($T_A$ = 25°C, f=1 Mhz, other pins open)**

| Parameter | Description | Test Condition | Maximum |
|---|---|---|---|
| $C_{OUT}$[1] | Output Pin Capacitance | $V_{OUT}$ = 0V | 12 pF |
| $C_{IN}$[1] | Input Capacitance | $V_{IN}$ = 0V | 6 pF |

T10.0 1295

1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.

**TABLE 11: Reliability Characteristics**

| Symbol | Parameter | Minimum Specification | Units | Test Method |
|---|---|---|---|---|
| $N_{END}$[1] | Endurance | 10,000 | Cycles | JEDEC Standard A117 |
| $T_{DR}$[1] | Data Retention | 100 | Years | JEDEC Standard A103 |
| $I_{LTH}$[1] | Latch Up | 100 + $I_{DD}$ | mA | JEDEC Standard 78 |

T11.0 1295

1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.

**TABLE 12: AC Operating Characteristics**

| | | 25 MHz | | 50 MHz | | |
|---|---|---|---|---|---|---|
| **Symbol** | **Parameter** | **Min** | **Max** | **Min** | **Max** | **Units** |
| $F_{CLK}$[1] | Serial Clock Frequency | | 25 | | 50 | MHz |
| $T_{SCKH}$ | Serial Clock High Time | 18 | | 9 | | ns |
| $T_{SCKL}$ | Serial Clock Low Time | 18 | | 9 | | ns |
| $T_{SCKR}$[2] | Serial Clock Rise Time (Slew Rate) | 0.1 | | 0.1 | | V/ns |
| $T_{SCKF}$ | Serial Clock Fall Time (Slew Rate) | 0.1 | | 0.1 | | V/ns |
| $T_{CES}$[3] | CE# Active Setup Time | 10 | | 5 | | ns |
| $T_{CEH}$[3] | CE# Active Hold Time | 10 | | 5 | | ns |
| $T_{CHS}$[3] | CE# Not Active Setup Time | 10 | | 5 | | ns |
| $T_{CHH}$[3] | CE# Not Active Hold Time | 10 | | 5 | | ns |
| $T_{CPH}$ | CE# High Time | 100 | | 50 | | ns |
| $T_{CHZ}$ | CE# High to High-Z Output | | 15 | | 8 | ns |
| $T_{CLZ}$ | SCK Low to Low-Z Output | 0 | | 0 | | ns |
| $T_{DS}$ | Data In Setup Time | 5 | | 2 | | ns |
| $T_{DH}$ | Data In Hold Time | 5 | | 5 | | ns |
| $T_{HLS}$ | HOLD# Low Setup Time | 10 | | 5 | | ns |
| $T_{HHS}$ | HOLD# High Setup Time | 10 | | 5 | | ns |
| $T_{HLH}$ | HOLD# Low Hold Time | 10 | | 5 | | ns |
| $T_{HHH}$ | HOLD# High Hold Time | 10 | | 5 | | ns |
| $T_{HZ}$ | HOLD# Low to High-Z Output | | 20 | | 8 | ns |
| $T_{LZ}$ | HOLD# High to Low-Z Output | | 15 | | 8 | ns |
| $T_{OH}$ | Output Hold from SCK Change | 0 | | 0 | | ns |
| $T_V$ | Output Valid from SCK | | 15 | | 8 | ns |
| $T_{SE}$ | Sector-Erase | | 25 | | 25 | ms |
| $T_{BE}$ | Block-Erase | | 25 | | 25 | ms |
| $T_{SCE}$ | Chip-Erase | | 50 | | 50 | ms |
| $T_{BP}$ | Byte-Program | | 10 | | 10 | µs |

T12.0 1295

1. Maximum clock frequency for Read Instruction, 03H, is 25 MHz
2. Maximum Rise and Fall time may be limited by $T_{SCKH}$ and $T_{SCKL}$ requirements
3. Relative to SCK.

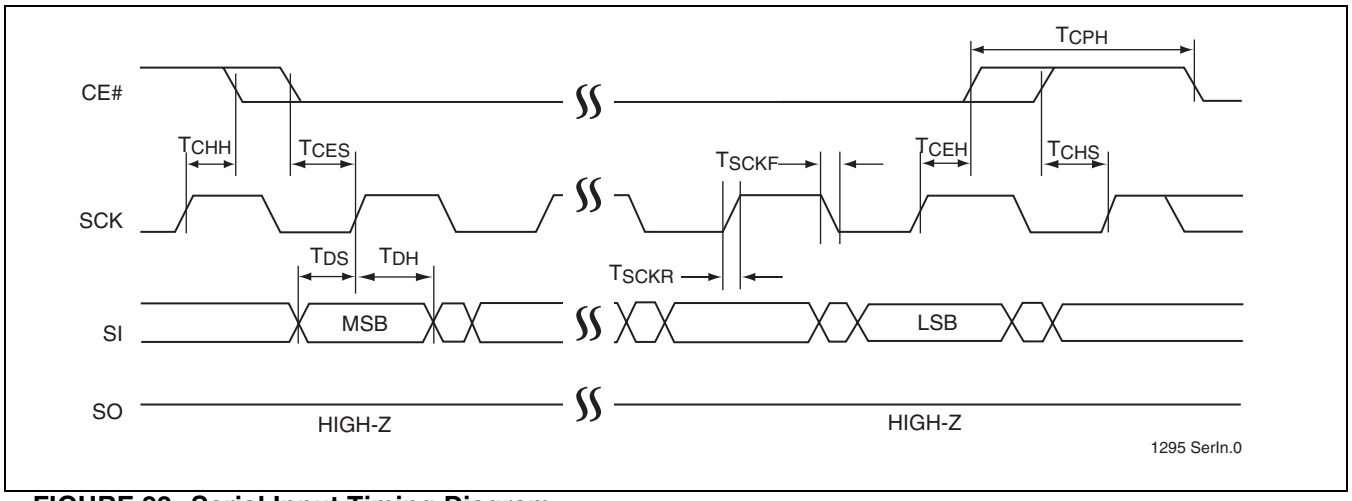



**FIGURE 22: Serial Input Timing Diagram**

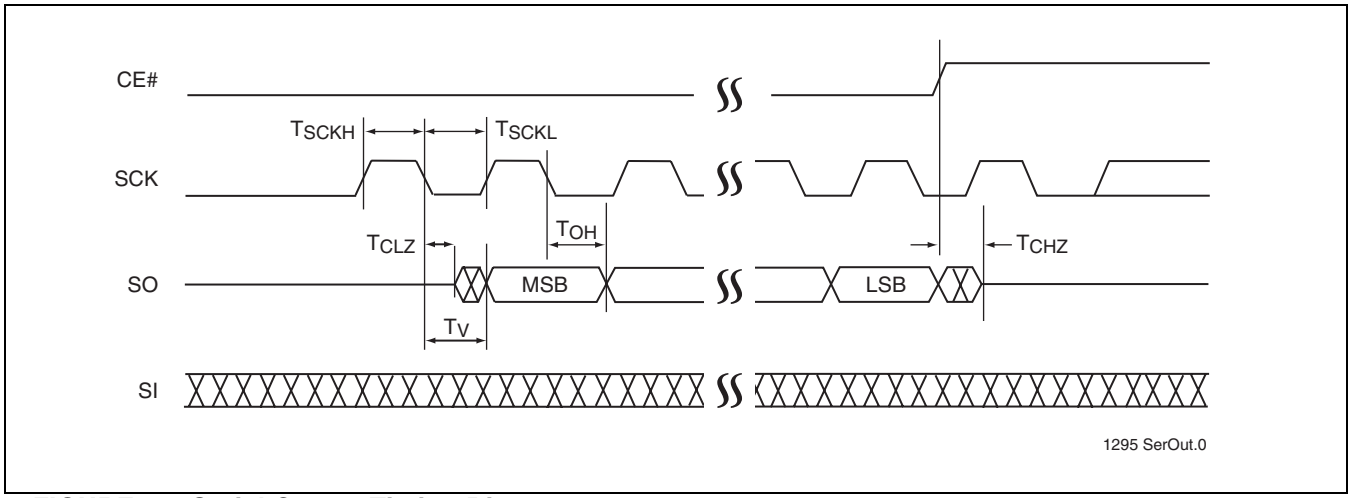



**FIGURE 23: Serial Output Timing Diagram**

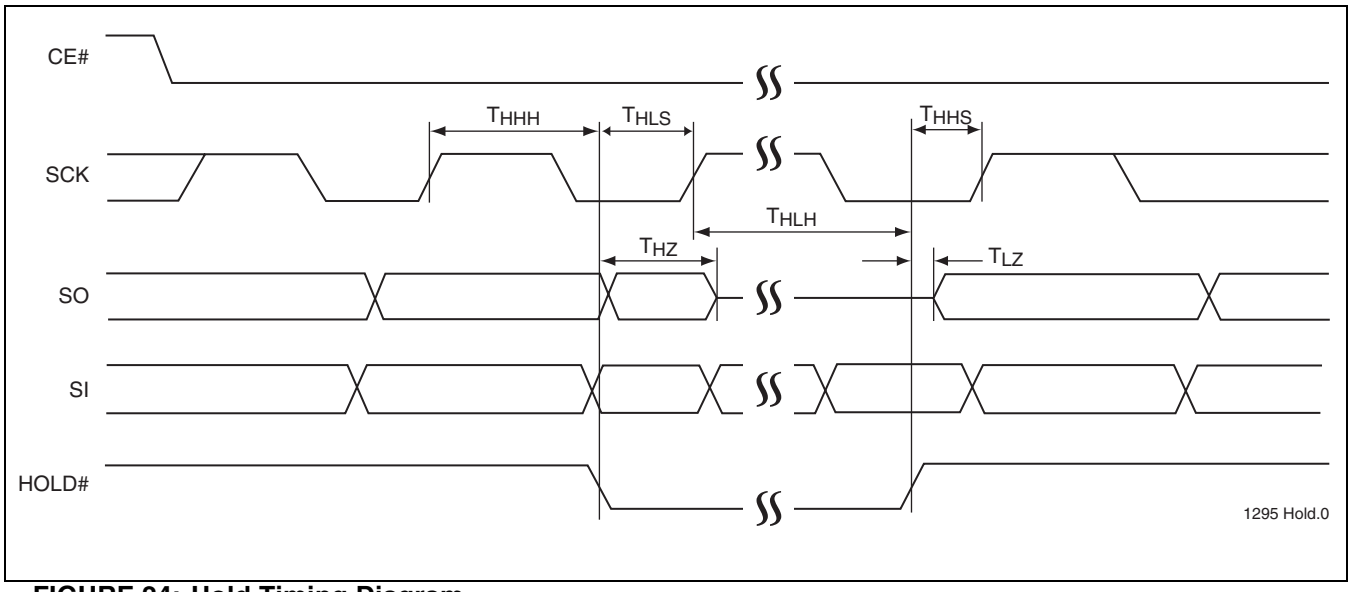



**FIGURE 24: Hold Timing Diagram**

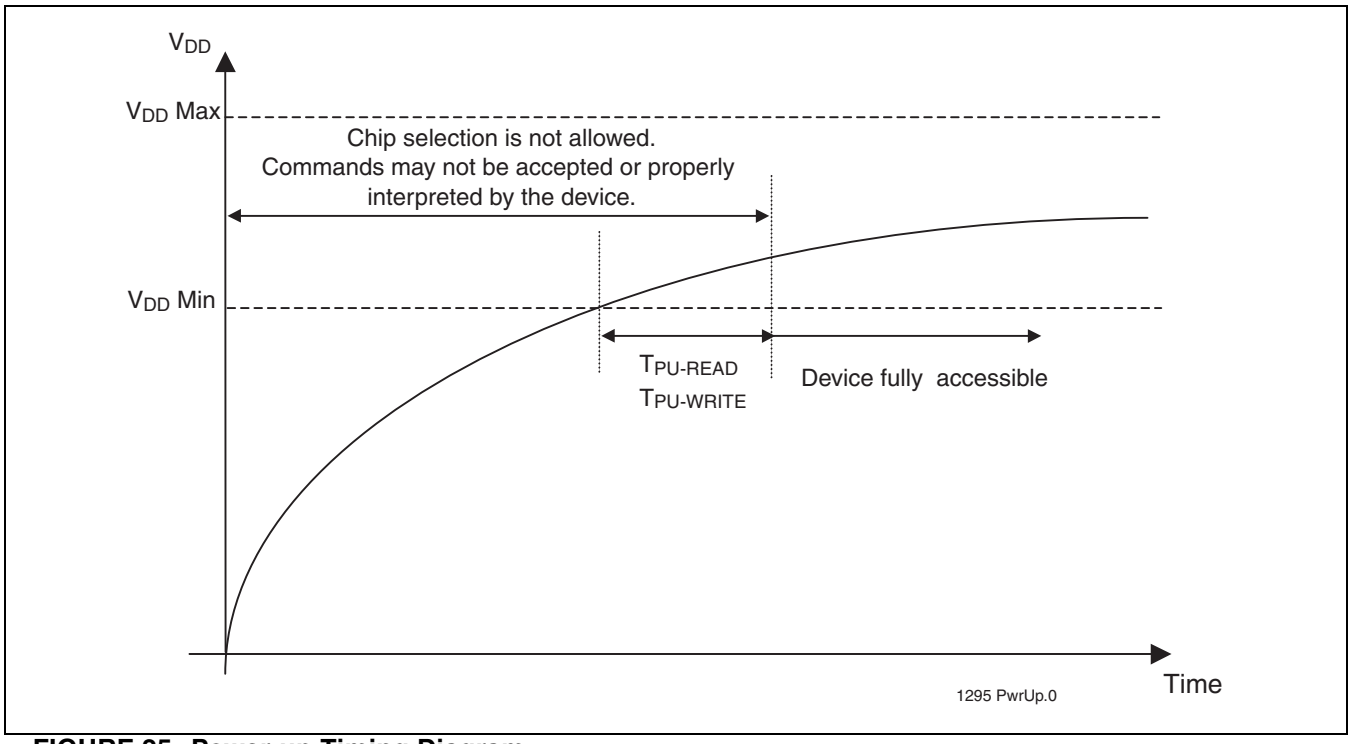
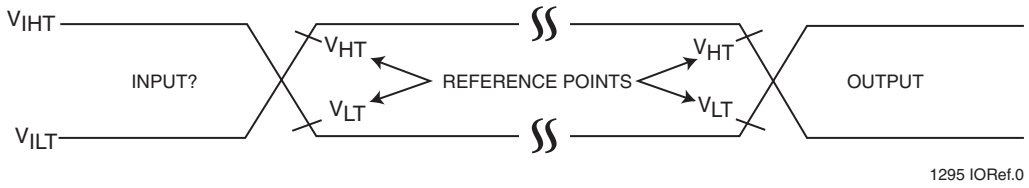



**FIGURE 25: Power-up Timing Diagram**




AC test inputs are driven at $V_{IHT}$ ($0.9V_{DD}$) for a logic "1" and $V_{ILT}$ ($0.1V_{DD}$) for a logic "0". Measurement reference points for inputs and outputs are $V_{HT}$ ($0.6V_{DD}$) and $V_{LT}$ ($0.4V_{DD}$). Input rise and fall times (10% ↔ 90%) are <5 ns.

**Note:** $V_{HT}$ - $V_{HIGH}$ Test
$V_{LT}$ - $V_{LOW}$ Test
$V_{IHT}$ - $V_{INPUT}$ HIGH Test
$V_{ILT}$ - $V_{INPUT}$ LOW Test

**FIGURE 26: AC Input/Output Reference Waveforms**

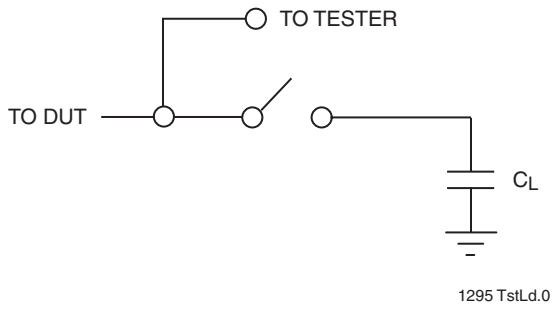
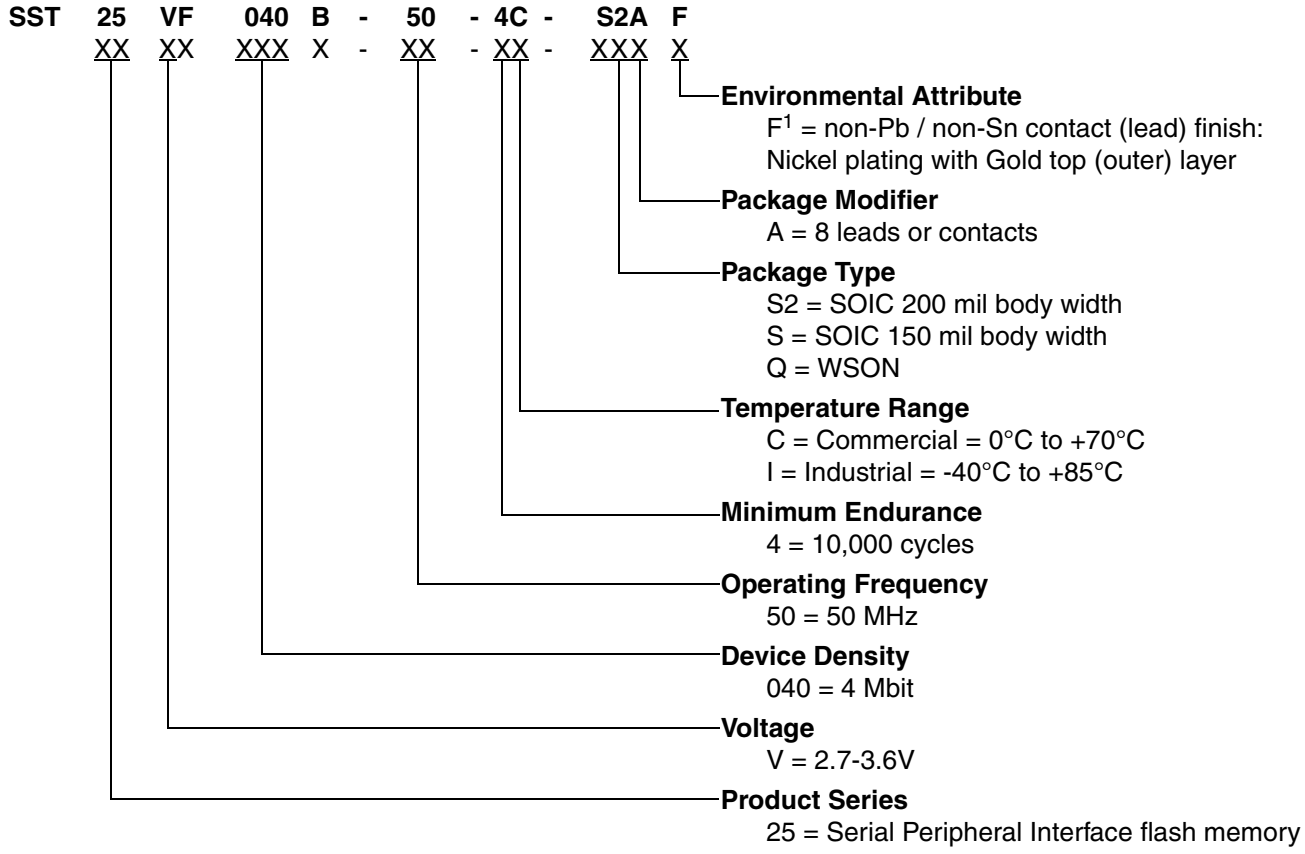



**FIGURE 27: A Test Load Example**

## PRODUCT ORDERING INFORMATION




1. Environmental suffix “F” denotes non-Pb/non-SN solder. SST non-Pb/non-Sn solder devices are “RoHS Compliant”.

**Valid combinations for SST25VF040B**

| | | |
|---|---|---|
| SST25VF040B-50-4C-S2AF | SST25VF040B-50-4C-QAF | SST25VF040B-50-4C-SAF |
| SST25VF040B-50-4I-S2AF | SST25VF040B-50-4I-QAF | SST25VF040B-50-4I-SAF |

**Note:** Valid combinations are those products in mass production or will be in mass production. Consult your SST sales representative to confirm availability of valid combinations and to determine availability of new combinations.

## PACKAGING DIAGRAMS

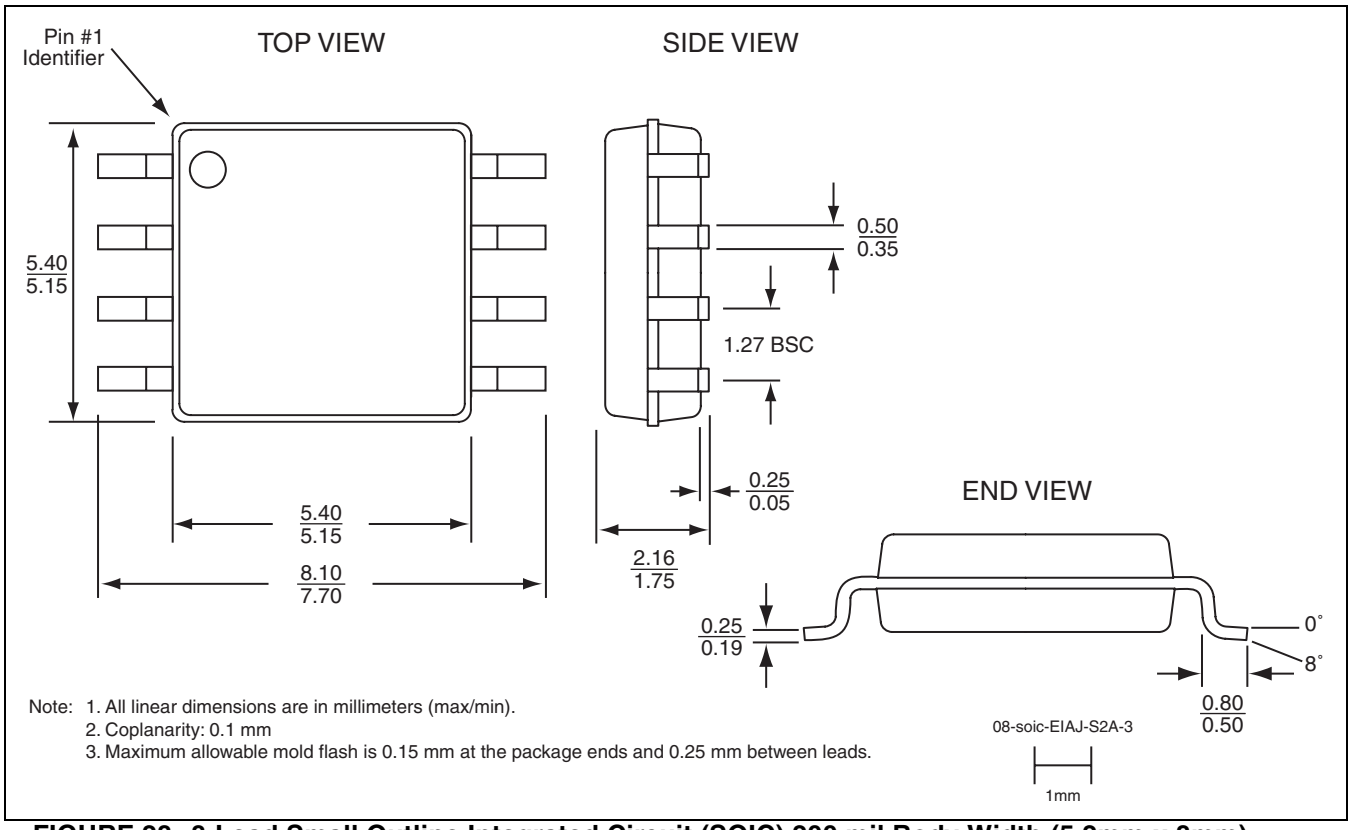
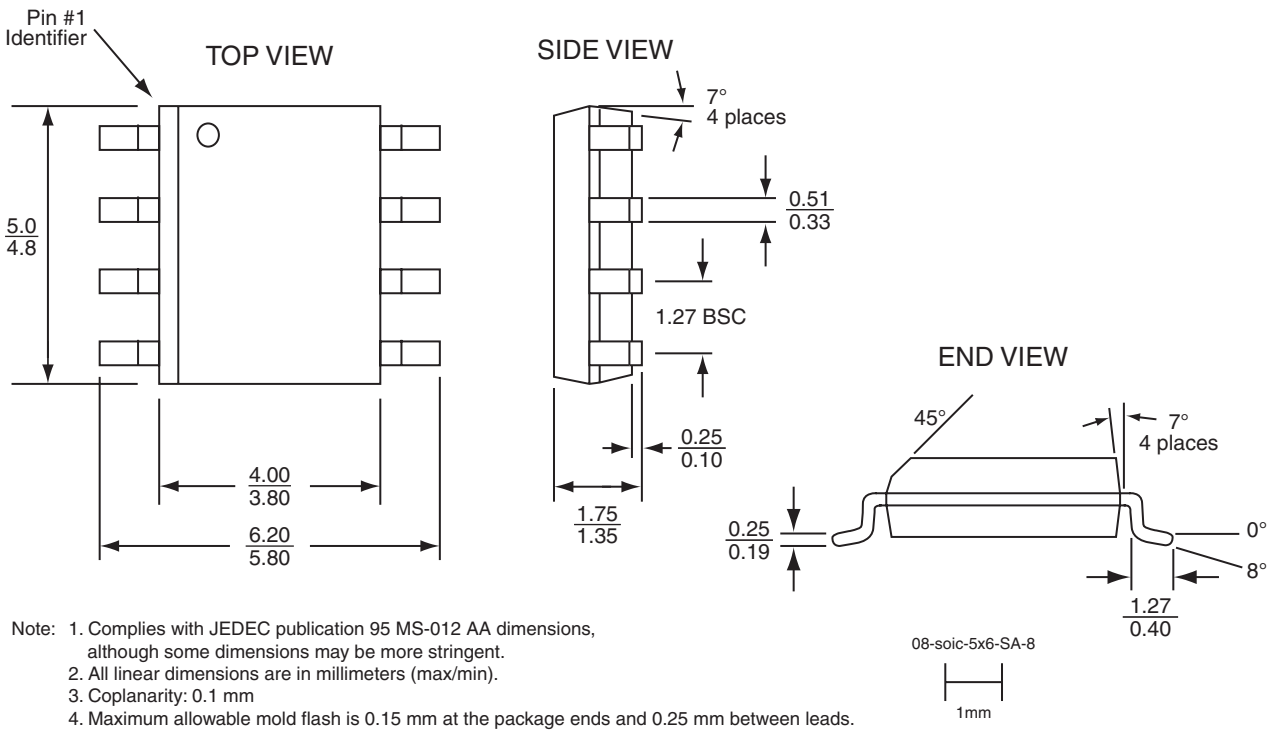
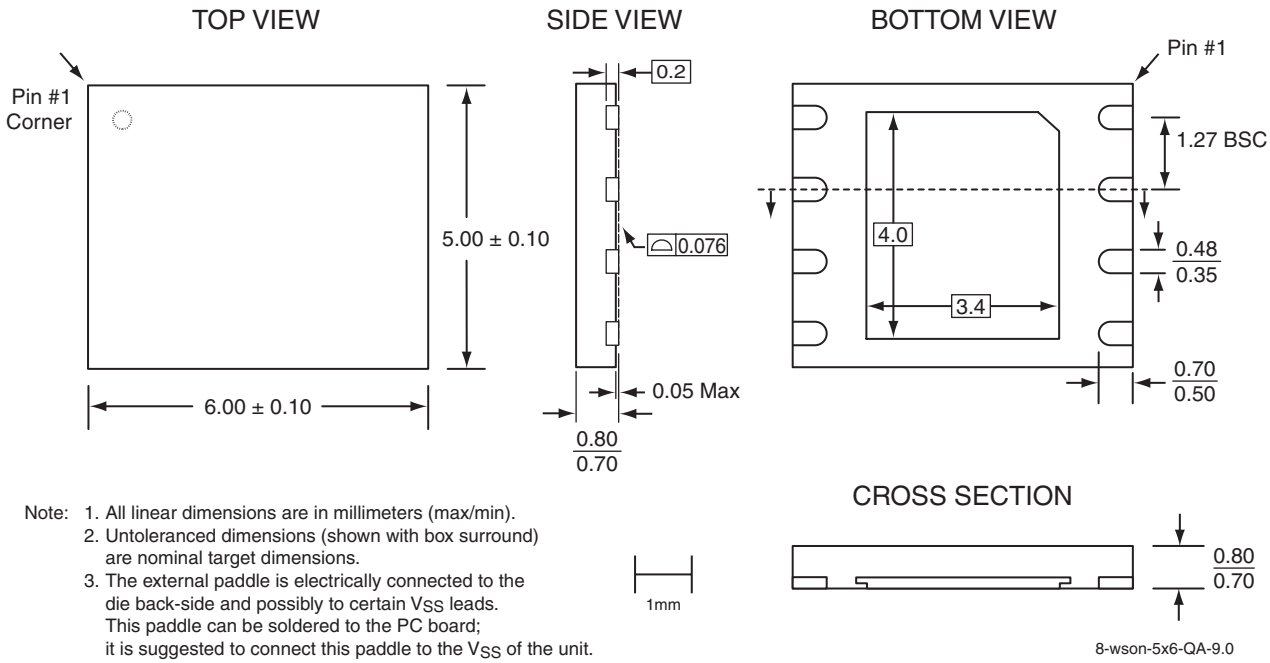



**FIGURE 28: 8-Lead Small Outline Integrated Circuit (SOIC) 200 mil Body Width (5.2mm x 8mm)**
**SST Package Code: S2A**




Note: 1. Complies with JEDEC publication 95 MS-012 AA dimensions, although some dimensions may be more stringent.
2. All linear dimensions are in millimeters (max/min).
3. Coplanarity: 0.1 mm
4. Maximum allowable mold flash is 0.15 mm at the package ends and 0.25 mm between leads.

**FIGURE 29: 8-Lead Small Outline Integrated Circuit (SOIC) 150mil Body Width (5mm x 6mm)**
**SST Package Code: SA**




Note: 1. All linear dimensions are in millimeters (max/min).
2. Untoleranced dimensions (shown with box surround) are nominal target dimensions.
3. The external paddle is electrically connected to the die back-side and possibly to certain $V_{SS}$ leads. This paddle can be soldered to the PC board; it is suggested to connect this paddle to the $V_{SS}$ of the unit. Connection of this paddle to any other voltage potential can result in shorts and/or electrical malfunction of the device.

**FIGURE 30: 8-Contact Very-very-thin Small Outline No-lead (WSON)**
**SST Package Code: QA**

**TABLE 13: Revision History**

| Number | Description | Date |
|---|---|---|
| 00 | • Initial release of data sheet | Sep 2005 |
| 01 | • Migrated document to a Data Sheet<br>• Updated Surface Mount Solder Reflow Temperature information | Jan 2006 |
| 02 | • Added 8-Lead SOIC (150 mils) package drawing.<br>• Updated Features and Product Description to include new package information.<br>• Updated Pin-Assignment, Figure 2<br>• Revised Figure 10 and Figure 11 | Jul 2007 |