NX25F011A
NX25F041A

NEXFLASH® Technologies, Inc.

1M-BIT AND 4M-BIT SERIAL FLASH MEMORIES WITH 4-PIN SPI INTERFACE

PRELIMINARY
JUNE 1999

FEATURES

- **Flash Storage for Resource-Limited Systems**
  - Ideal for portable/mobile and microcontroller-based applications that store voice, text, and data
- ***NexFlash* Serial Flash Memory**
  - Patented single transistor EEPROM technology
  - High-density, low-voltage & power, cost-effective
  - Small 264-byte sectors
  - 10K/100K write cycles, ten years data retention
- **Ultra-low Power for Battery-Operation**
  - Single 5V or 3V supply for read and erase/write
  - 1 µA standby current, 5 mA active @ 3V (typical)
  - Low frequency read command for very low power
  - No pre-erase. Erase/Write time of 5 ms/sector @ 5V ensures efficient battery use
- **4-pin SPI Serial Interface**
  - Easily interfaces with popular microcontrollers
  - Clock operation as fast as 16 MHz
- **On-chip Serial SRAM**
  - Dual 264-byte Read/Write SRAM buffers
  - Use in conjunction with or independent of Flash
  - Off-loads RAM-limited microcontrollers
- **Special Features for Media-Storage Applications**
  - Byte-level addressing
  - Transfer and compare sector to SRAM commands
  - Versatile hardware and software write-protection
  - Alternate oscillator frequency for EMI sensitive applications.
  - In-system electronic part number identification
  - Removable Serial Flash Module package option

DESCRIPTION

The NX25F011A and NX25F041A Serial Flash memories provide a storage solution for systems limited in power, pins, space, hardware, and firmware resources. They are ideal for applications that store voice, text, and data in a portable or mobile environment. Using *NexFlash's* patented single transistor EEPROM cell, the devices offer a high-density, low-voltage, low-power, and cost-effective non-volatile memory solution. The devices operate on a single 5V or 3V (2.7V-3.6V) supply for Read and Erase/Write with typical current consumption as low as 5 mA active and less than 1 µA standby. Sector erase/write speeds as fast as 5 ms increase system performance, minimize power-on time, and maximize battery life.

The NX25F011A and NX25F041A provide 1M-bit and 4M-bit of flash memory organized as 512 and 2048 sectors of 264 bytes each. Each sector is individually addressable through basic serial-clocked commands. The 4-pin SPI serial interface works directly with popular microcontrollers. Special features include: on-chip serial SRAM, byte-level addressing, double-buffered sector writes, transfer/compare sector to SRAM, hardware and software write protection, alternate oscillator frequency, electronic part number, and removable Serial Flash Module package option. Development is supported with the PC-based Serial Flash Development Kit.

FUNCTIONAL OVERVIEW

An architectural block diagram of the NX25F011A and NX25F041A is shown in Figure 2. Key elements of the architecture include:

- SPI Interface and Command Set Logic
- Serial Flash Memory Array
- Serial SRAM and Program Buffer
- Write Protection Logic
- Configuration and Status Registers
- Device Information Sector

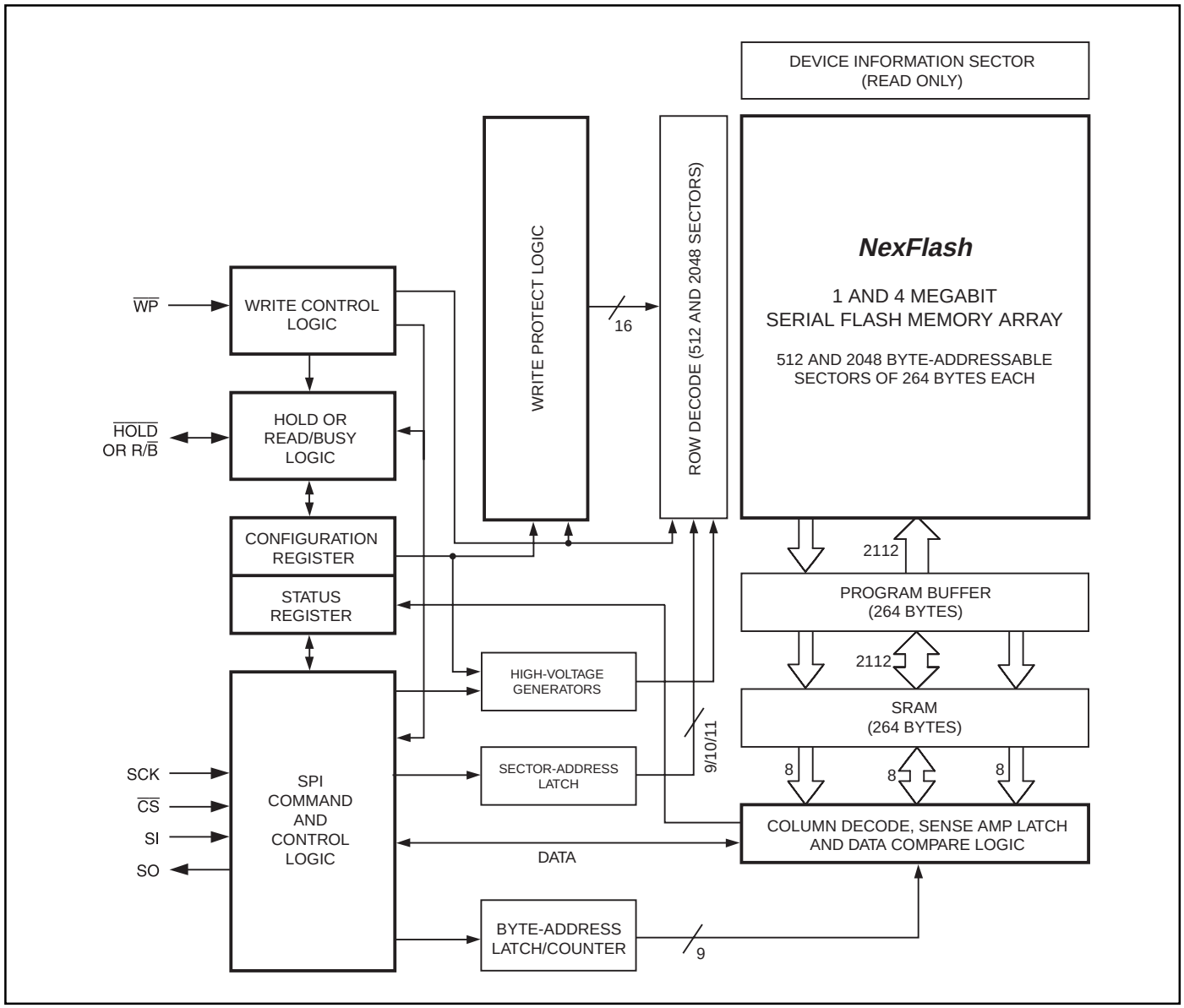


Figure 2. NX25F011A and NX25F041A Architectural Block Diagram

Pin Descriptions

Package
The NX25F011A and NX25F041A are available in a 28-pin TSOP (Type I) surface mount package. See Figure 3 and Table 1 for pin assignments. All interface and supply pins are on one side of the package. The “No Connect” (NC) pins are not connected to the device, allowing the pads and the area around them to be used for routing PCB system traces. The devices are also available in a cost-effective and space-efficient removable Serial Flash Module package (see NX25Mxxx data sheet).

Serial Data Input (SI)
The SPI bus Serial Data Input (SI) provides a means for data to be written to (shifted into) the device.

Serial Data Output (SO)
The SPI bus Serial Data Output (SO) provides a means for data to be read from (shifted out of) the device during a read operation. When the device is deselected ($\overline{CS}$=1 or $\overline{HOLD}$=0) the SO pin is in a high-impedance state.

Serial Clock (SCK)
All commands and data written to the Serial Input (SI) are clocked relative to the rising edge of the Serial Clock (SCK). By default all data read from the Serial Data Output (SO) is clocked relative to the falling edge of SCK, allowing compatibility with standard SPI systems. The user may specify reading relative to the rising edge of SCK by changing the setting of the RCE bit in the Configuration Register (see Figure 6). Clock rates of up to 16 MHz for 5V devices and up to 8 MHz for 3V devices are supported.

Chip Select ($\overline{CS}$)
The NX25F011A and NX25F041A are selected for operation when the Chip Select input ($\overline{CS}$) is asserted low. Upon power-up, an initial low-to-high transition of $\overline{CS}$ is required before any command sequence will be acknowledged. The device can be deselected to a non-active state when $\overline{CS}$ is brought high. Once deselected, the SO pin will enter a high-impedance state and power consumption will decrease to standby levels unless programming is in process, in which case standby will resume when programming is complete.

Write Protect ($\overline{WP}$)
The Write Protect input ($\overline{WP}$) works in conjunction with the write protect range set in the configuration register bits. When $\overline{WP}$ is asserted (active low) the entire Flash memory array is write protected. When high, any Flash memory sector can be written to unless its address is within the write protect range that is set in the configuration register.

Hold or Ready/Busy ($\overline{HOLD}$ or R/$\overline{B}$)
This multi-function pin can serve either as a Hold input ($\overline{HOLD}$) or as a Ready-Busy output (R/$\overline{B}$). Factory-programmed as a no connect, the pin can be reconfigured as a Ready-Busy output or as a Hold input by setting the configuration register. Warning: this pin is tied low in the Serial Flash Module and must be left as a no connect (NC).

Power Supply Pins (Vcc and GND)
The NX25F011A and NX25F041A support single power supply Read and Erase/Write operations in 5V and 3V versions. Typical active power is as low as 5 mA for the 3V version with standby current less than 1 µA.

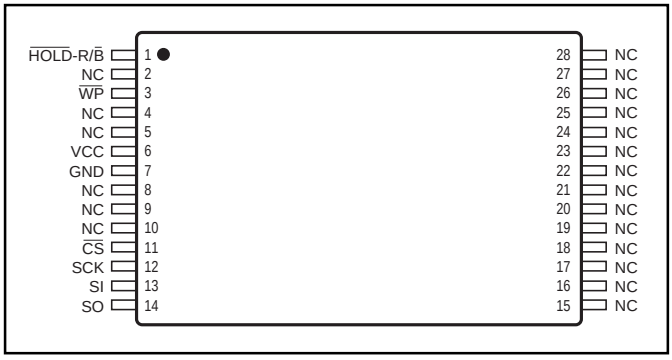


Figure 3. NX25F011A and NX25F041A Pin Assignments, 28-Pin TSOP (Type I)

Table 1. Pin Descriptions

SI	Serial Data Input
SO	Serial Data Output
SCK	Serial Clock Input
$\overline{CS}$	Chip Select Input
$\overline{WP}$	Write Protect Input
$\overline{Hold}$, R/$\overline{B}$	Hold Input or Read Busy Output
Vcc	Power Supply
GND	Ground

Serial Flash Memory Array

The Flash memory array of the NX25F011A and NX25F041A are organized as 512 and 2048 sectors of 264-bytes (2,112 bits) each, as shown in Figure 4. Grouping sectors as pairs offers a convenient format for applications that store and transfer data in a DOS compatible sector size of 512-bytes. The additional 16-bytes per sector pair can be used for sector management such as header, checksum, CRC, or other related application requirements.

The Serial Flash memory of the NX25F011A and NX25F041A is byte-addressable. That is, each sector is individually addressable and each byte within a sector is individually addressable. This allows a single byte, or specified sequence of bytes, to be read without having to clock an entire 264-byte sector out of the device. Data can be read directly from a sector in the Flash memory array by using a *Read from Sector* command from the SPI bus. Data can be written to a sector in the Flash memory array using a *Write to Sector* command or a *Transfer SRAM to Sector* command.

After a sector has been written, the memory array will become busy while it is programming the specified non-volatile memory cells of that sector. This busy time will not exceed t_{WP} (~5 ms for 5V devices), during which time the Flash array is unavailable for read or write access. The device can be tested to determine the array's availability by reading the Ready/Busy status, by reading the status register, or by testing the Ready/Busy pin. Note that the SRAM is always available, even when the memory array is busy. See the Serial SRAM section for more details.

The NX25F011A and NX25F041A do not require pre-erase. The device incorporates an auto-erase-before-write feature that automatically erases the addressed sector at the beginning of the write operation. This allows for fast and consistent programming times and simplifies firmware support by eliminating the need for a separate pre-erase algorithm and the complex management of disproportional erase and write block sizes commonly found in other devices.

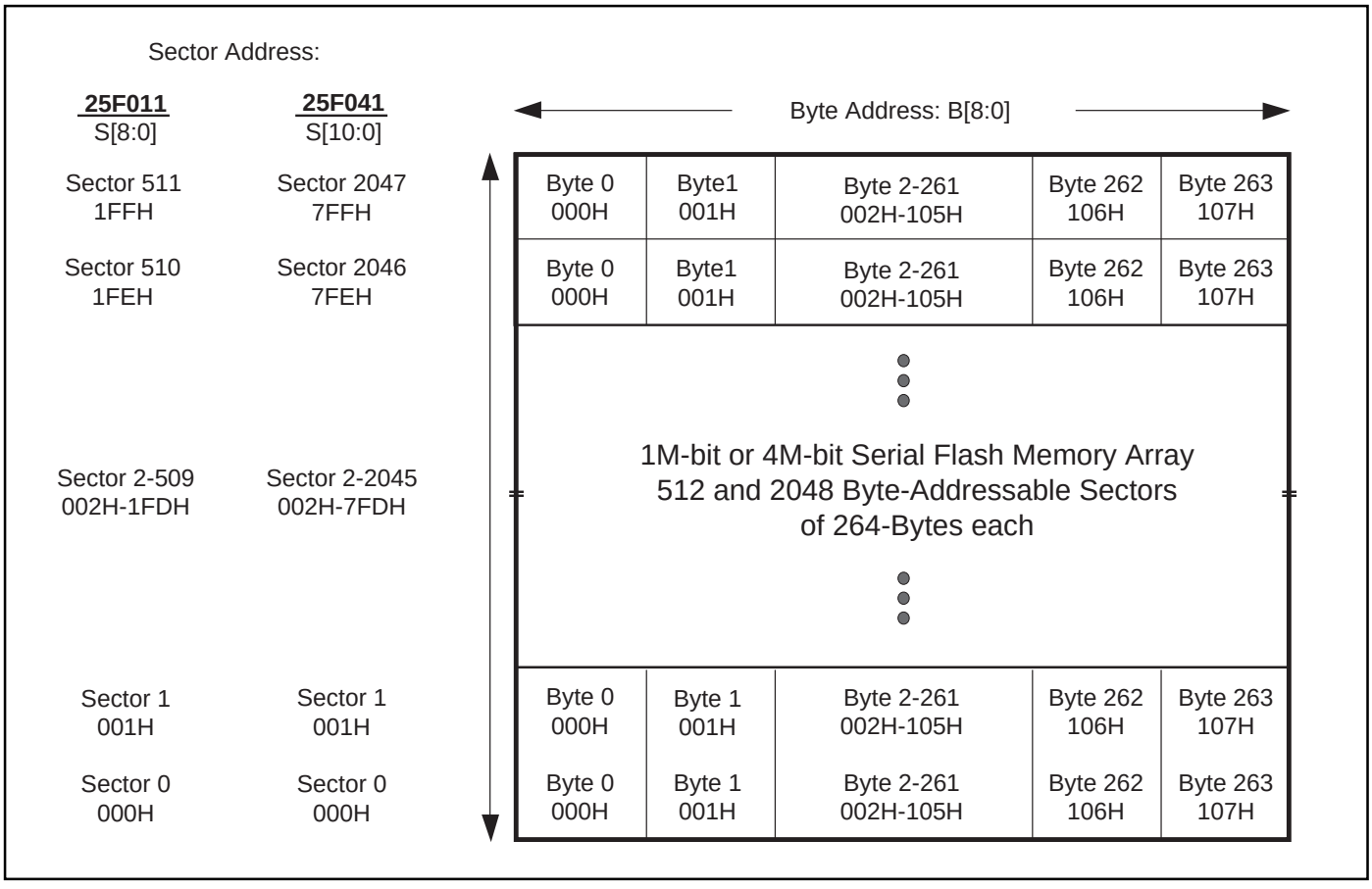


Figure 4. NX25F011A and NX25F041A Serial Flash Memory Array

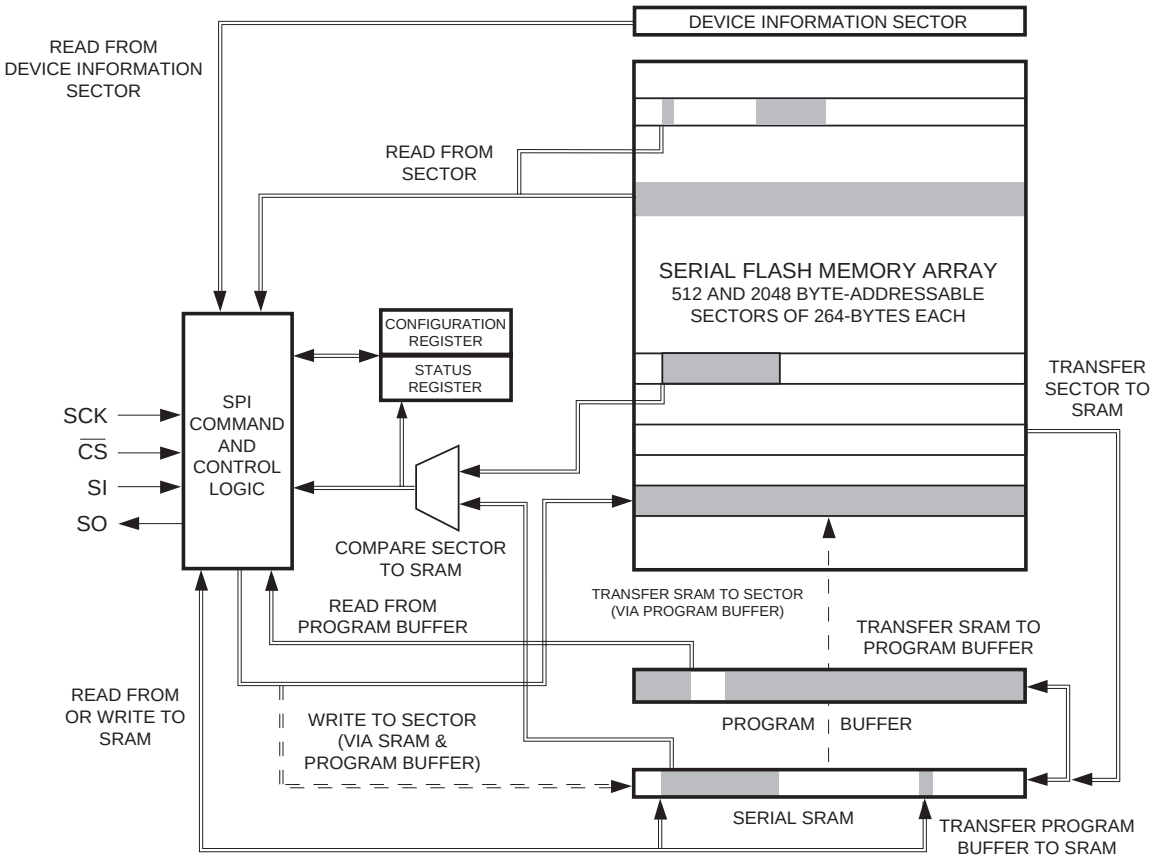
Serial SRAM and Program Buffer

One of the most powerful features of the NX25F011A and NX25F041A is the integrated Serial SRAM and its associated Program Buffer. Together, the 264-byte Serial SRAM and 264-byte Program Buffer provide up to 528-bytes of usable SRAM storage. The SRAM can be used in conjunction with the Flash memory or independently.

The main purpose of the Serial SRAM is to serve as the primary buffer for data to be written into the Serial Flash memory array. Using the *Write to Sector* command, data is first shifted into the SRAM from the SPI bus. When the command sequence has been completed, the entire 264-bytes is transferred to the Program Buffer. The Program Buffer supports the array during the Erase/Write cycle (t_{WP}), freeing the SRAM to accept new data. This double-buffering scheme increases erase/write transfer rates and can eliminate the need for external RAM buffers (Figure 5).

The SRAM is fully byte-addressable. Thus, the entire 264-bytes, a single byte, or a sequence of bytes can be read from or written to the SRAM. This allows the SRAM to be used as a temporary work area for read-modify-write operations prior to a sector write.

The *Transfer Sector to SRAM* command allows the contents of a specified sector of Flash memory to be moved to the SRAM. This can be useful when only a portion of a sector needs to be altered. In this case the sector is first transferred to the SRAM, where modifications are made using the *Write to SRAM* command. Once complete, a *Transfer SRAM to Sector* command is used to update the sector.



Note:
1. A single byte, several bytes, or all bytes of a Flash sector, the SRAM, or Program Buffer may be addressed.
2. All double lines represent implied connections or actions.

Figure 5. Command Relationships of the SPI Interface, Serial Flash Memory Array, SRAM and Program Buffer

The *Compare Sector* command allows the contents of the SRAM to be compared with the specified sector in memory. The result of the compare is set in the status register. This command can be useful for verifying a sector after write (see High Data Integrity Applications towards the end of this data sheet). It is also useful when rewriting multi-sector files that have only minor changes from the previous write. If the new data in the SRAM is the same as the previously written data, the sector write can be skipped. Used in this way, the command saves time that would have been used for re-programming. It also extends the endurance of the Flash memory cells.

Using the SRAM Independent of Flash Memory

The SRAM can be used independently of Flash memory operations for lookup tables, variable storage, or scratch pad purposes. If the Flash memory needs to be written to while SRAM is being used for a different purpose, the contents can be temporarily stored to a sector and then transferred back again when needed. The SRAM can be especially useful for RAM-limited microcontroller-based systems, eliminating the need for external SRAM and freeing pins for other purposes. It can also make it possible to use small pin-count microcontrollers, since only a few pins are needed for the interface instead of the 20-40 pins required for parallel bus-oriented Flash devices.

If more than 264 bytes of SRAM are needed, the *Transfer SRAM to Program Buffer*, *Transfer Program Buffer to SRAM*, and *Read Program Buffer* commands can be used to expand the storage to 528 bytes. In this mode of operation, all writes must be handled through the 264-byte SRAM and the Program buffer is essentially used as a stack.

Write Protection

The NX25F011A and NX25F041A provide advanced software and hardware write protection features. Software-controlled write protection of the entire array is handled using the *Write Enable and Write Disable* commands. Hardware write protection is possible using the Write Protect pin ($\overline{WP}$). Write-protecting a portion of Flash memory is accommodated by programming a write protect range in the configuration register. For applications needing a portion of the memory to be permanently write-protected, a onetime programmable write protection feature is supported. Contact *NexFlash* for further information.

Configuration Register

The Configuration Register stores the current configuration of the $\overline{HOLD}$-R/$\overline{B}$ pin, read clock edge, write protect range, and alternate oscillator frequency (Figure 6). The configuration register is accessed using the *Write and Read Configuration Register* commands. The non-volatile configuration register will maintain its setting even when power is removed.

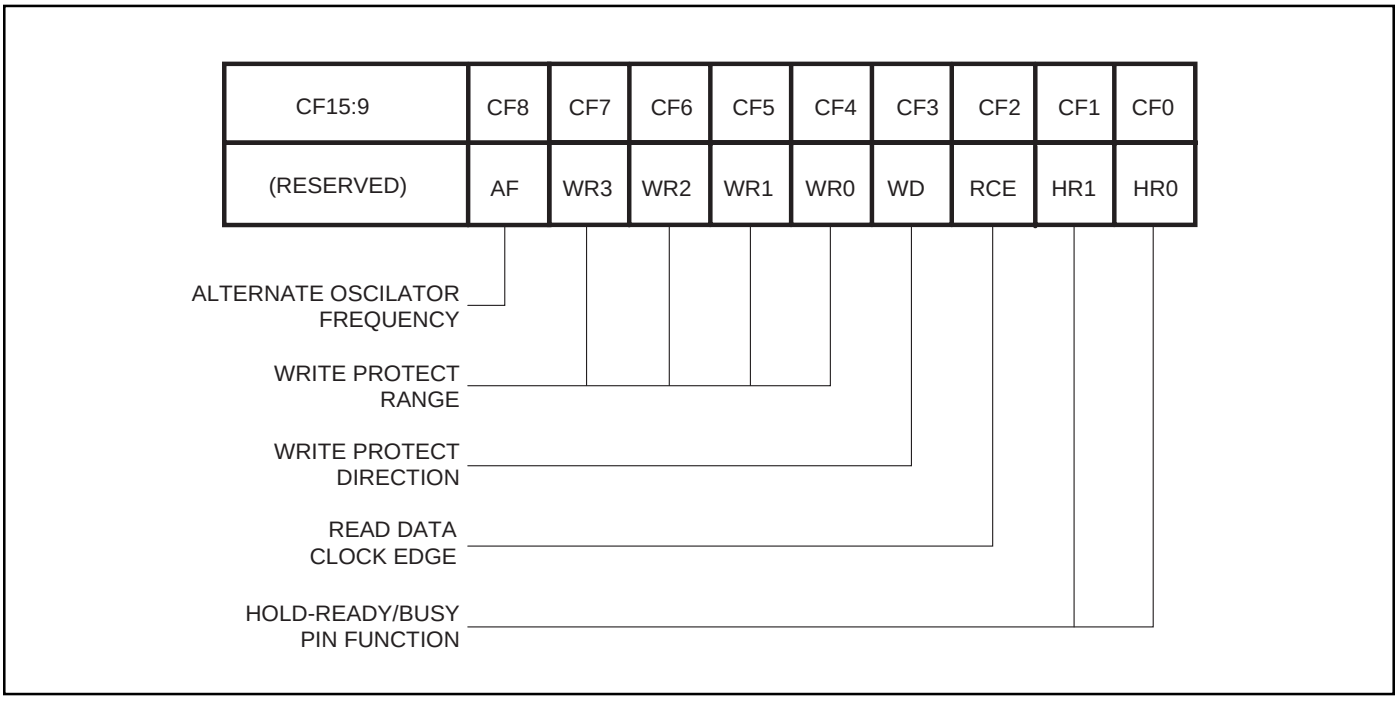


Figure 6. Configuration Register Bit Locations

The factory default setting for bits CF8-CF0 is: 0 0000 1001 B(write protect range = none, read uses falling edge of the clock, and pin 1 = no connect). Bits CF15-CF9 are reserved. When writing to the configuration register CF15-CF9 should be 0. When reading, the settings of CF15-CF9 should be ignored.

Standard write endurance rating of the memory array allows for 10,000 erase/write cycles. Extended endurance to 100,000 cycles is possible using ECC techniques like those provided in the Serial Flash Development Kit. The rating of the configuration register EEPROM cells is 1,000 write cycles. This is more than adequate considering the configuration seldom needs to be changed. To minimize writes to the configuration register, the configuration register should be read upon power-up to determine if a change is required. If no change is needed, the write configuration command can be skipped. This process will extend the life of the configuration register and save processing time (Figure 7).

Alternate Oscillator Frequency, AF
Flash memory devices have charge pump oscillators to generate internal high-voltages used for programming non-volatile memory cells. In some applications, the oscillator frequency of the charge pump may cause noise interference. To solve this problem, an alternate oscillator frequency (AF) can be selected by setting bit CF[8] of the configuration register. The alternate frequency is a non-harmonic frequency of the standard oscillator. The factory default setting is for the standard oscillator frequency, AF equal to 0.

AF=0 Standard Oscillator Frequency is used.
AF=1 Alternate Oscillator Frequency is used.

Write Protect Range and Direction, WR[3:0], WD
The write protect range and direction bits WR[3:0] and WD are located at configuration bits CF[7:4] and CF[3] respectively. The write protect range and direction bits select how the array is protected. They work in conjunction with the $\overline{WP}$ input pin, valid only if $\overline{WP}$ is inactive (high). WR[3:0] can select write protection of all sectors, none of the sectors, or specific sectors grouped in blocks of 32 (~8 KB). The WD bit specifies whether the protected block range starts from the first sector, address 0 (000H), or from the last sector (1FFH for the NX25F011A and 7FF for the NX25F041A). Table 2 lists the write protect sector range for both devices. Once protected, all further writes to sectors within the range will be ignored. The factory default setting is with no write protected sectors, WR=[0,0,0,0] and WD=1.

Read Clock Edge, RCE
The Read Clock Edge bit (RCE) is located at configuration bit location CF[2]. It selects which edge of the clock (SCK) is used while reading data out of the device. Although the SPI protocol specifies that data is written during the rising edge and read on the falling edge of the clock, the output can be driven on the rising edge of the clock by setting the configuration registers RCE bit to a 1. Using the rising edge of clock for data reads may be beneficial to the timing of some high-speed systems. The factory default setting is for reading on the falling edge of SCK.

RCE=0 Read data is output on the falling edge of SCK.
RCE=1 Read data is output on the rising edge of SCK.

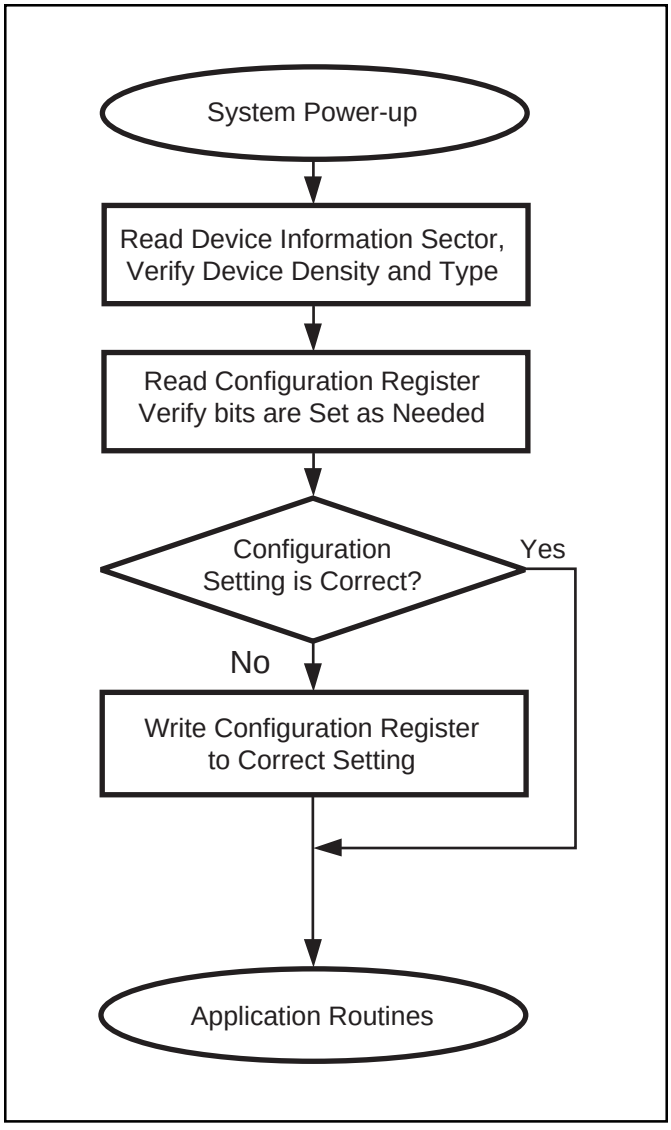


Figure 7. Flow Chart for Checking the Configuration Register upon Power-up

Table 2. Write Protect Range Sector Selection (Hex)

Write Protect Range Config. Bits				Write Protected Sectors	
WR3	WR2	WR1	WR0	WD=0	WD=1[1]
0	0	0	0	None	None
0	0	0	1	000 - 01FH	x E0 - 1FF/ 7FFH
0	0	1	0	000 - 03FH	x C0 - 1FF/ 7FFH
0	0	1	1	000 - 05FH	x A0 - 1FF/ 7FFH
0	1	0	0	000 - 07FH	x 80 - 1FF/ 7FFH
0	1	0	1	000 - 09FH	x 60 - 1FF/ 7FFH
0	1	1	0	000 - 0BFH	x 40 - 1FF/ 7FFH
0	1	1	1	000 - 0DFH	x 20 - 1FF/ 7FFH
1	0	0	0	000 - 0FFH	x 00 - 1FF/ 7FFH
1	0	0	1	000 - 11FH	y E0 - 1FF/ 7FFH
1	0	1	0	000 - 13FH	y C0 - 1FF/ 7FFH
1	0	1	1	000 - 15FH	y A0 - 1FF/ 7FFH
1	1	0	0	000 - 17FH	y 80 - 1FF/ 7FFH
1	1	0	1	000 - 19FH	y 60 - 1FF/ 7FFH
1	1	1	0	000 - 1BFH	y 40 - 1FF/ 7FFH
1	1	1	1	ALL	ALL

Note:

1. NX25F011A x=1 y=0 and NX25F041A x=7 Y=6,

$\overline{\text{HOLD}}$-R/$\overline{\text{B}}$, HR[1:0]

The Hold-Ready/Busy ($\overline{\text{HOLD}}$-R/$\overline{\text{B}}$) bits HR1 and HR0 are located at bits CF[1:0] of the configuration register. These two bits select one of four possible functions: no connect, $\overline{\text{HOLD}}$ input, R/$\overline{\text{B}}$ Output, or R/$\overline{\text{B}}$ Output with open drain. The factory setting for the pin is "No Connect". Warning: this pin is tied low in the Serial Flash Module and must be left as a no connect (NC) for Serial Flash Module Applications.

HR1	HR0	Pin Configuration
0	0	$\overline{\text{HOLD}}$ input
0	1	No Connect
1	0	R/$\overline{\text{B}}$ Output (Open Drain)
1	1	R/$\overline{\text{B}}$ Output

Configured as a R/$\overline{\text{B}}$ output, the pin can serve as a system interrupt. When R/$\overline{\text{B}}$ is high, the array is ready to be programmed. When R/$\overline{\text{B}}$ is low, it is busy programming. If configured with an open-drain, an external pull-up resistor should be used.

As a $\overline{\text{HOLD}}$ input, the pin can be used in conjunction with the $\overline{\text{CS}}$ and SCK pin to suspend a serial command sequence without resetting the command. This can be useful if a command is in process and a higher priority task on the same SPI bus needs to be attended to. To suspend a command, $\overline{\text{HOLD}}$ must be brought low while $\overline{\text{CS}}$ and SCK are low. With $\overline{\text{HOLD}}$ low, further data on the SI pin is ignored (even while SCK is clocked) and the SO pin goes to a high-impedance state. To resume the command sequence, $\overline{\text{HOLD}}$ must be brought high when $\overline{\text{CS}}$ and SCK are low. See timing diagrams.

Status Register Bit Descriptions

The status register provides status of the Flash array's Ready/Busy condition (R/$\overline{\text{B}}$), transfers between the SRAM and program buffer (TX), Write-Enable/Disable (WE), and Compare Not Equal (CNE). The register can be read using the Read Status Register command (Figure 8).

Ready/Busy Status, BUSY

The BUSY status bit is located at bit ST[7] of the status register. Testing the BUSY bit is one of several ways to check Ready/Busy status of the array. At power-up the BUSY bit is reset to 0.

BUSY=1 The memory array is busy programming.
BUSY=0 The memory array is ready for further use.

SRAM and Program Buffer Transfer, TR

The TR status bit is located at bit ST[6] of the status register. The bit provides status primarily for use during the *Transfer SRAM to Program Buffer* command and *Transfer Program Buffer to SRAM* command. An active state 1 indicates a transfer is in process and the SRAM or Program Buffer is not available for use. The device will indicate a BUSY state while the TR bit is active. Upon power up the TR bit resets to 0.

TR=1 SRAM and Program Buffer Transferring.
TR=0 SRAM and Program Buffer Not Transferring.

Write Enable/Disable, WE

The WE status bit is located at bit ST[4] of the status register. The bit provides write protect status of global *Write Enable and Write Disable* commands. Upon power- up the WE bit resets to 0.

WE=1 Write Enabled, array can be written to.
WE=0 Write Disabled, array can not be written to.

The WE status bit can also be used to determine the state of the WP (write protect) pin. This can be done by first issuing the Write Enable Command and then reading the WE status bit. If the status bit indicates a "0" (write disabled) then the WP pin is likely held low.

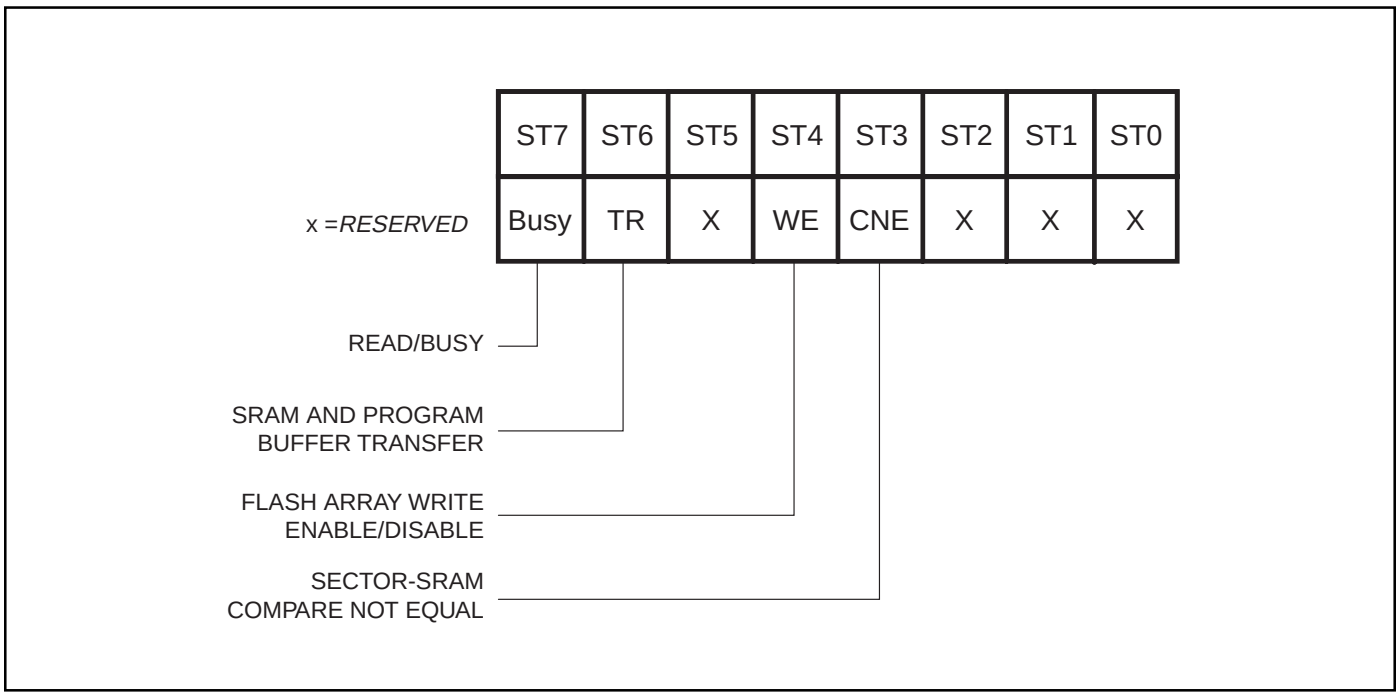


Figure 8. Status Register Bit Locations

Compare Not Equal, CNE

The CNE status bit is located at bit ST[3] of the status register. The bit provides a cumulative comparison result during a *Compare Sector with SRAM* command. The CNE bit is reset to a 0 upon power-up or after a Clear Compare Bit command is executed.

CNE=1 Sector and SRAM contents are not equal.
CNE=0 Sector and SRAM are equal or CNE bit reset.

Command Set

The NX25F011A and NX25F041A have a powerful command set that is fully controlled through the SPI bus. Command relations are shown in Figure 5 and a list of commands and their associated address, status, clock, and data bytes are shown in Table 3. Detailed clock timing of the *Read Sector* and *Write Sector* command sequences are shown in Figures 10 and 11.

After power up, a device enters an idle state that will maintain until $\overline{CS}$ pin is asserted low. All commands are entered from the SPI serial data input (SI) pin on the rising edge of SCK while $\overline{CS}$ is asserted low. All command, address, and configuration bits are shifted into the device with most-significant-bit-first. Data bits read from the device are shifted out with least significant byte first (i.e., byte-00H, byte-01H,...). The bit order within each byte is most-significant-bit first (i.e.,D7,...D0). All commands are completed by asserting the $\overline{CS}$ pin high.

Note that the entire 264-byte contents of a Flash sector, the SRAM, or Program Buffer does not have to be accessed all at once. Read, Write, Transfer, and Compare commands allow for byte addressing. Thus a single byte, or clocked sequence of bytes, can be accessed at any starting location within the 264-byte boundary as specified by the byte-address field.

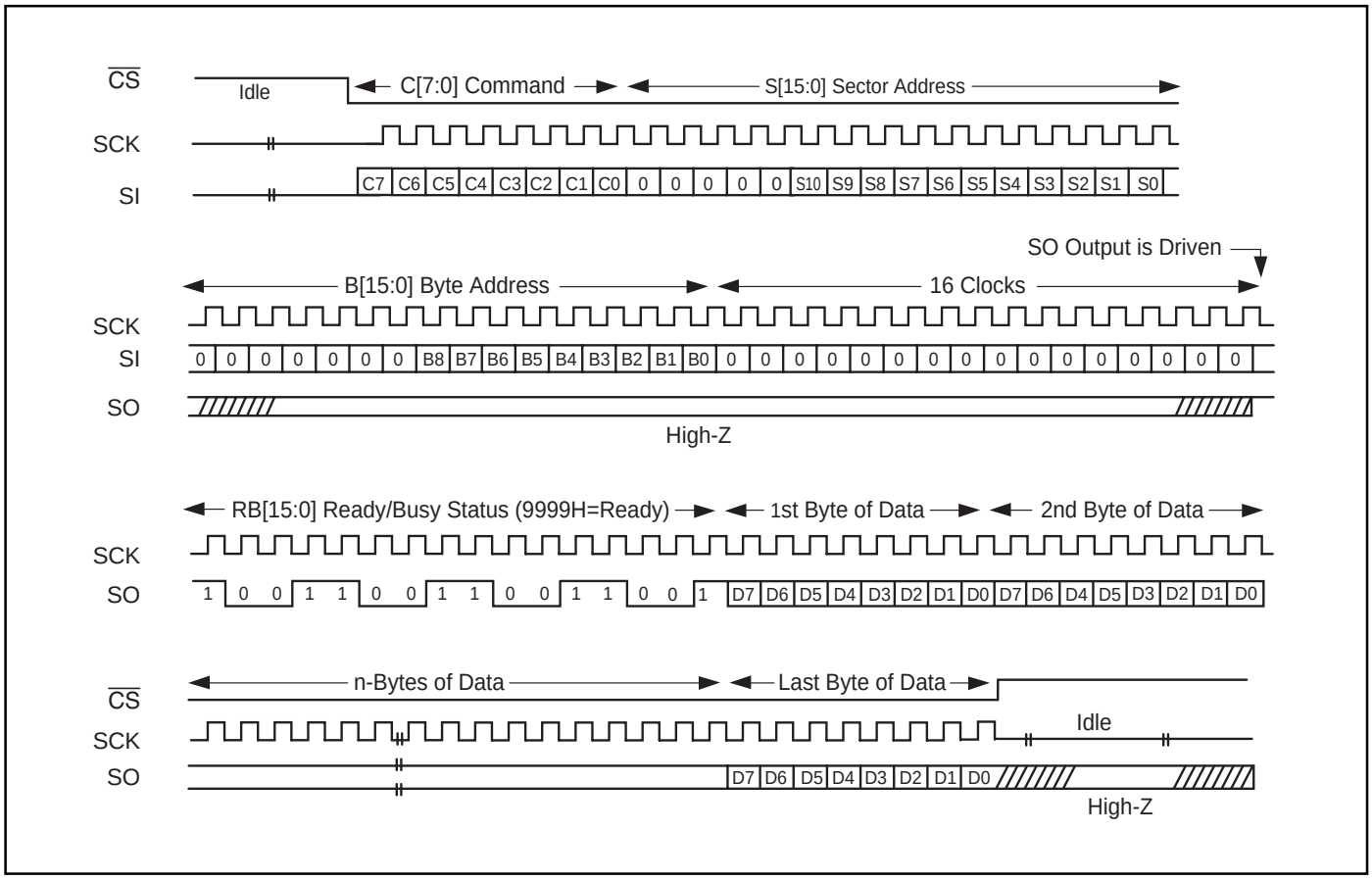


Figure 10. Read from Sector Command Sequence

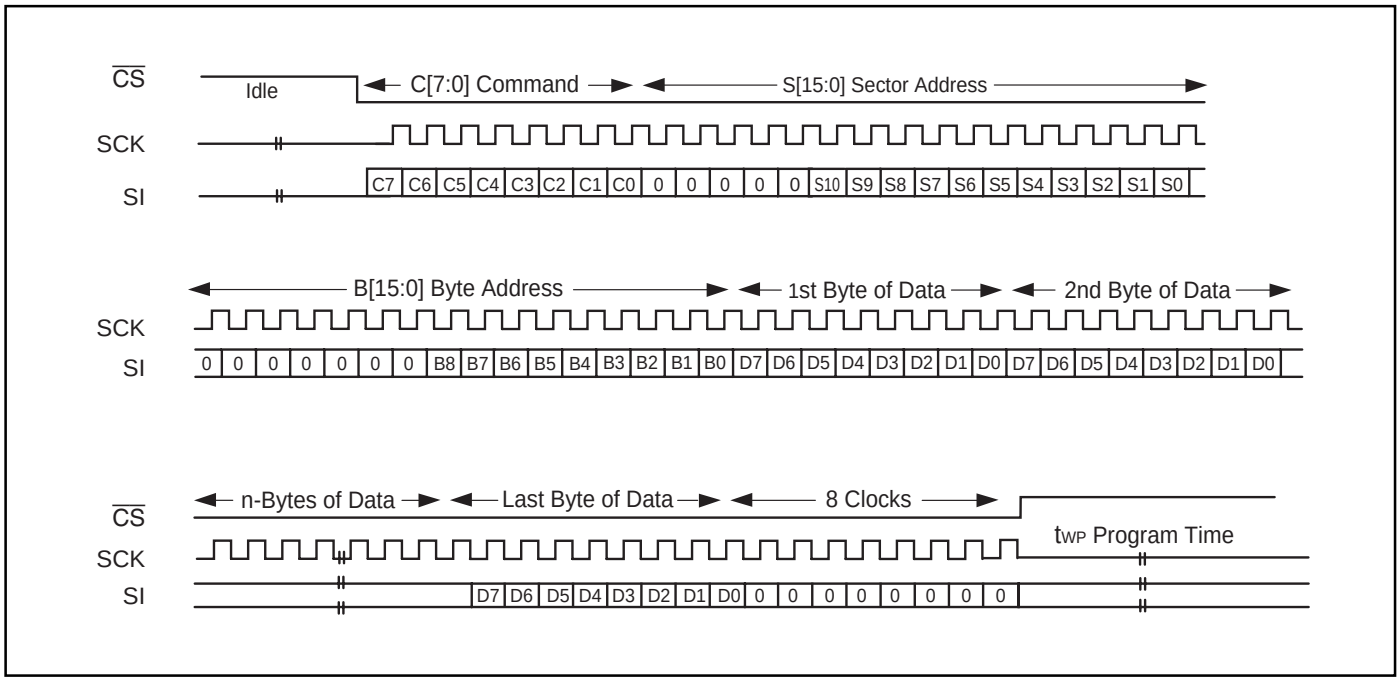


Figure 11. Write to Sector Command Sequence

Table 3. Command Set for the NX25F011A, and NX25F041A Serial Flash Memory

Command Name	Byte 0	Byte 1-2	Byte 3-4	n - bytes (*italics indicate device output*)			
Serial Flash Sector Commands							
Read from Sector	52H	sector addr.	byte add.	0000H	*ready/busy*	*read data*	
Read from Sector Low Frequency	51H	sector addr.	byte add.	0000H	*ready/busy*	*read data*	
Write Enable*	06H	00H					
Write Disable*	04H	00H					
Write to SectorF3H		sector addr.	byte add.	write data		00H	
Transfer SRAM to Sector	F3H	sector addr.	0000H				
Transfer Sector to SRAM	54H	sector addr.	byte add.	clock 00H per byte		00H	
Compare Sector with SRAM	86H	sector addr.	byte add.	0000H	*ready/busy*	*bit compare of data*	
Serial SRAM Program Buffer Commands							
Write to SRAM**	82H	0000H	byte add.	write data		00H	
Read from SRAM*	81H	0000H	byte add.	0000H	*read/busy*	*read data*	
Transfer SRAM to Prog. Buffer	92H	0000H	0000H	0000H			
Transfer Prog. Buffer to SRAM	55H	0000H	0000H	0000H			
Read from Program Buffer	91H	0000H	byte add.	0000H	*ready/busy*	*read data*	
Configuration and Status Commands							
Read Configuration Register*	8BH	0000H	0000H	0000H	*ready/busy*	*configuration*	
Write Configuration Register	8AH	configuration	0000H				
Read Status Register*	83H	0000H	0000H	0000H	*ready/busy*	*status*	
Clear Compare Status*	89H	0000H					
Read Device Information Sector	15H	0000H	byte add.	0000H	*ready/busy*	*read data*	

Notes:
1. * Command may be used when device is busy
2. ** Command may not be used when device is busy and TR bit=0

SERIAL FLASH SECTOR COMMANDS

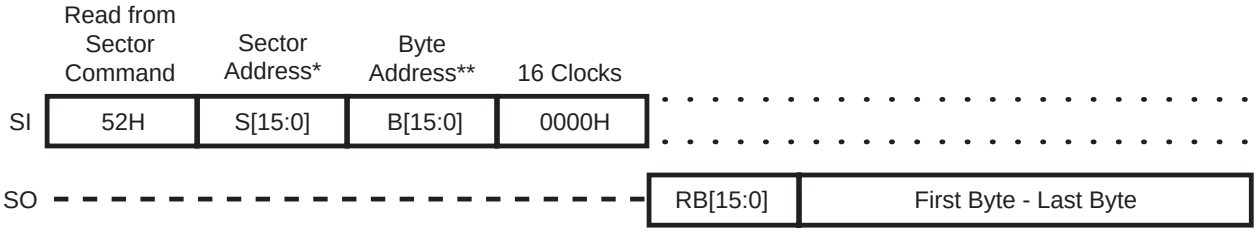
Read From Sector

Reading from a sector is accomplished by first bringing $\overline{CS}$ low then shifting in the *Read from Sector* command (52H) followed by its 16-bit "sector-address" field. Although the sector-address field is 16-bits, only bits S[8:0] for the NX25F011A (0-1FFH) and S[10:0] for the NX25F041A (0-7FFH) are used. The uppermost sector address bits are not used but must be clocked using 0 for data. Next a 16-bit "byte-address" field is clocked into the device to designate the starting location within the 264-byte sector. Only B[8:0] of the byte-address field are used; the uppermost bits are not used but must be clocked in (use 0 for data). Only byte-addresses of 0 to 107H (264 bytes) are valid.

Following the byte-address field, 16 control clocks are required with data=0. The Serial Data Output (SO) will change from a high-impedance state and begin to drive the output with Ready/Busy status RB[15:0]. If SO uses the rising edge of clock (configuration register RCE=1), the output will be driven after the last control clock. If SO uses the falling edge of clock (RCE=0), the output will be driven on the next falling edge of clock. If the array is not busy, the output status will be 9999H, followed by the sector data on the SO pin. If the array is busy, the status will be 6666H, and the command should be terminated and restarted after a ready state occurs. The data field is shifted out with the least significant byte first (i.e., byte-00H, byte-01H, ...). The bit order within each byte is the most significant bit first (i.e.,D7,...D0). The byte-address is internally incremented to the next higher byte address as the clock continues. When the highest byte-address (107H) is reached, the address counter rolls over to byte-0H and continues to increment. Asserting the $\overline{CS}$ pin high completes (or terminates) the command. Detailed timing for the *Read from Sector* command is shown in Figure 10.

Read Sector (Low Frequency)

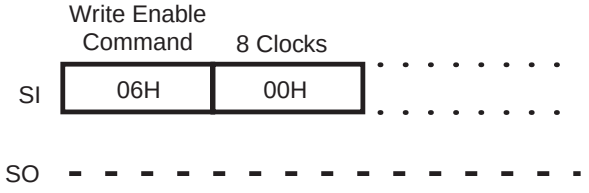
The *Read Sector at Low Frequency* command (51H) can reduce power consumption during read operations by 25%-40% when the system clock frequency is 1 MHz or lower. The command sequence is identical to the standard *Read from Sector* command.



*The sector address only uses bits [8:0], [9:0] or [10:0]
**The byte address only uses bits [8:0]

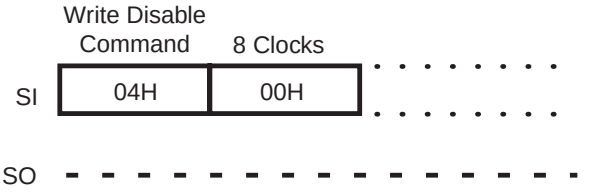
Write Enable

Upon power-up, the Flash memory array is write- protected until the *Write Enable* command (06H) has been issued. The $\overline{WP}$ pin must be inactive while writing the command for the write enable to be accepted. The status of the device's write protect state can be read in the status register. The *Write Enable* command sequence is completed by asserting $\overline{CS}$ high after eight additional clocks.



Write Disable

The *Write Disable* command (04H) protects the Flash memory array from being programmed. Once issued, further *Write to Sector* or *Transfer SRAM to Sector* commands will be ignored. The status of the write protect state can be read in the status register. The *Write Disable* command sequence is completed by asserting $\overline{CS}$ high after eight additional clocks.



Write to Sector

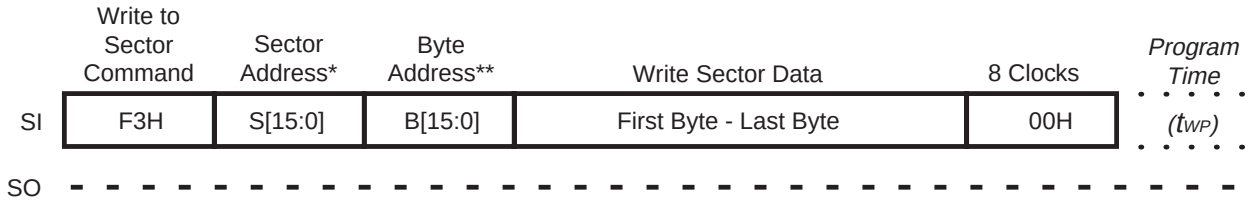
Before writing to a sector in the Flash memory array, all hardware and software write protection must be in an enabled state. This means that the $\overline{WP}$ pin must be in a high state, a *Write Enable* command must have previously been issued, and the sector location that is to be written to must be outside the write protect range set in the configuration register. Additionally, the Ready/Busy status should be checked to confirm that the memory array is available to be written to.

Writing to a sector is accomplished by first bringing $\overline{CS}$ low and shifting in the *Write to Sector* command (F3H) followed by a 16-bit "sector-address" field. Although the sector-address field is 16-bits, only bits S[8:0] for the NX25F011A (0-1FFH) and S[10:0] for the NX25F041A (0-7FFH) are used. The uppermost sector address bits are not used but must be clocked in (use 0 data). Following the sector address, a 16-bit "byte-address" field is clocked into the device to designate the starting location within the 264-byte sector. Only bits B[8:0] of the byte-address field are used and only values of 0-107H (264 bytes)are valid.

After the byte-address has been loaded, data is shifted into the 264-byte SRAM, which serves as a temporary storage buffer. Existing data in the SRAM will be written over. The byte order of the data shifted into the SRAM is least significant byte first (i.e., byte-00H, byte-01H,...).

The bit order within each byte is most significant bit first (i.e., D7,...D0). The byte-address is automatically incremented to the next higher byte address as the clock continues. When the last byte address to be written is reached, the command can be completed with an additional eight control clocks (with data=0) followed by asserting $\overline{CS}$ high. If the clock continues to increment past the highest byte-address (107H), the address counter will roll over to byte 0H.

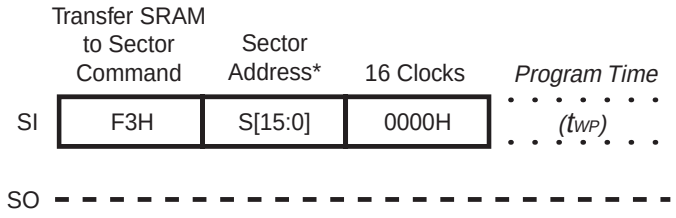
After the $\overline{CS}$ pin is brought high, the data in the SRAM is automatically transferred to the Program Buffer, which handles the self-timed programming of the specified sector in memory array. See t_{WP} timing specifications. During this time the array will be "busy" and will ignore further array-related commands until complete. All Ready/Busy status indicators will indicate a busy status. Since the Program Buffer handles all array programming, the SRAM is still available to be read from or written to during the busy state. This allows the SRAM to be written to in preparation of the next sector write. Once the SRAM is loaded, its contents can be transferred to a selected sector using the *Transfer SRAM to Sector* command. Note that if the write operation is to be verified the contents of the SRAM should be maintained. Detailed clock timing for the *Write to Sector* command is shown in Figure 11.



*The sector address only uses bits [8:0] or [10:0]
**The byte address only uses bits [8:0]

Transfer SRAM to Sector

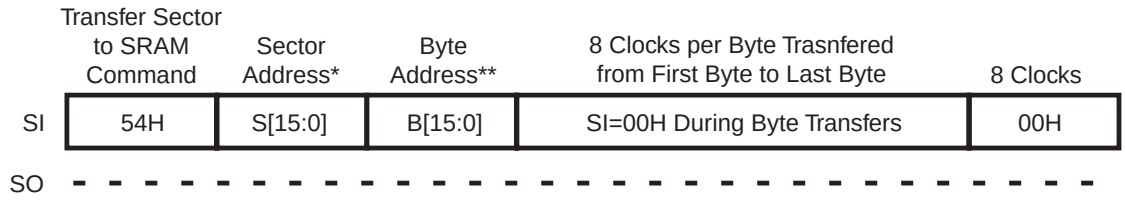
The *Transfer SRAM to Sector* command (F3H) will write the existing contents of the SRAM to the specified sector in memory. The command sequence is identical to that of the *Write to Sector* command except that immediately after the sector address field S[15:0] and 16 control clocks, the $\overline{CS}$ pin is asserted high. This automatically transfers the 264-bytes of SRAM data to the Program Buffer, which handles the programming of the specified sector in the memory array. During this time, the array will be busy. Since the entire 264-bytes are transferred, the byte-address field B[15:0] is not used.



*The sector address only uses bits [8:0] or [10:0]

Transfer Sector to SRAM

The *Transfer Sector to SRAM* command (54H) allows the contents of a sector to be transferred directly to the SRAM without having to read the sector out of the device and rewrite it into the SRAM. The command is similar to the *Write to Sector* command except that instead of inputting data from the SI pin, the data is taken from the specified sector and is transferred to the SRAM. Every eight clocks on SCK, a byte from the specified sector to the SRAM will be transferred. Although data on SI is ignored, it is recommended to write data bytes of 00H in order to support the clocking requirements. During the transfer, the SO output is in a high-impedance state. When the last byte address is transferred, the command can be completed by issuing eight more control clocks and asserting $\overline{CS}$ high. If the clock continues to increment past the highest byte-address (107H), the address counter will roll over to byte-0H.

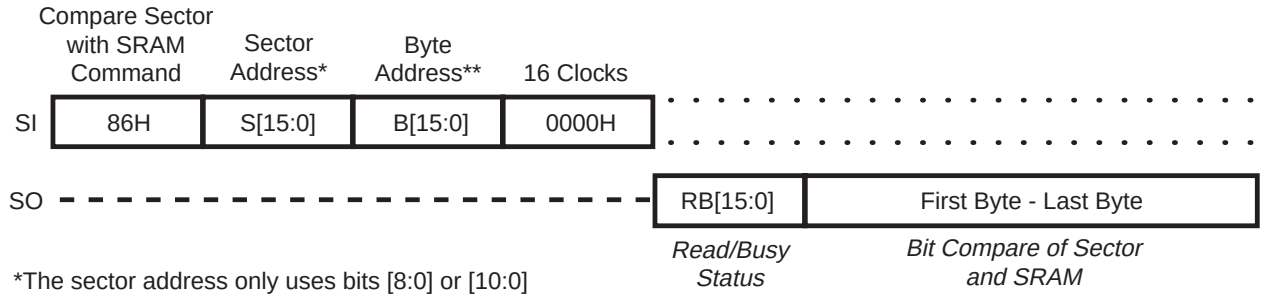


*The sector address only uses bits [8:0] or [10:0]
**The byte address only uses bits [8:0]

Compare Sector to SRAM

The *Compare Sector to SRAM* command does a bit-by-bit comparison of the data stored in the addressed sector against data in the SRAM. The command is similar to the *Read from Sector* command except that data is not read out of the Serial Output pin (SO). Instead, the SO pin provides a bit-by-bit compare of each sector and SRAM bit. A high (1) per bit will be output if the bit compare is equal. A low (0) per bit will be output if the bit compare is not equal.

The compare can start from any location in the 264-byte range as specified by the byte-address field B[15:0]. The byte-address counter is automatically incremented and will wrap around to the first address (0H) if it passes the last address (107H). If any of the compared bits are not equal, then the *Compare Not Equal* (CNE) bit in the Status Register is set to a 1. This bit will stay set until a *Clear Compare Status* command has been issued.

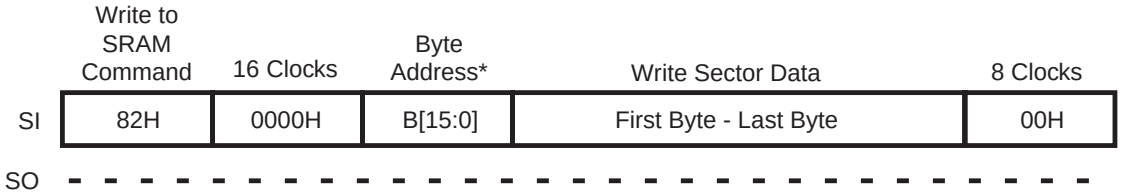


*The sector address only uses bits [8:0] or [10:0]
**The byte address only uses bits [8:0]

SERIAL SRAM AND PROGRAM BUFFER COMMANDS

Write to SRAM Command

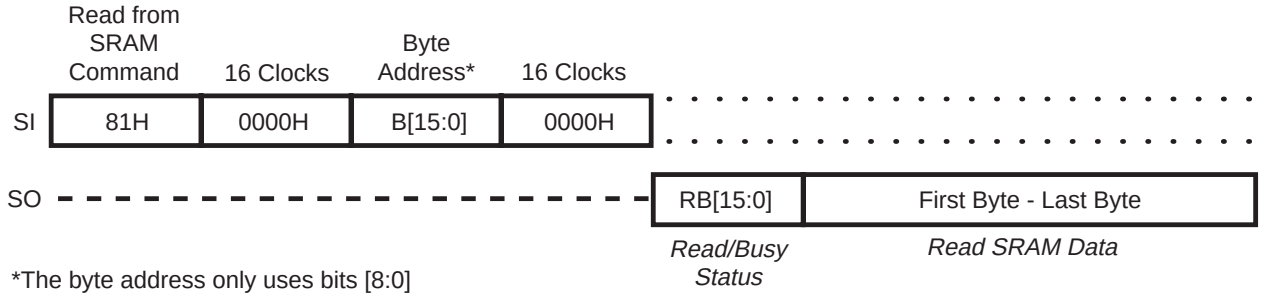
The *Write to SRAM* command (82H) provides access to the 264-Byte SRAM independently of any Flash memory array operation. The command is similar to the *Write to Sector* command sequence except that the sector address field S[15:0] is replaced by all 0 bits. When $\overline{CS}$ is asserted high to complete the command, the contents of the SRAM will be maintained until overwritten via another command or the power is removed. Using the *Write to SRAM* command, data can be loaded in preparation of writing to a sector in memory and then transferred to a selected sector using the *Transfer SRAM to Sector* command.



*The byte address only uses bits [8:0]

Read from SRAM

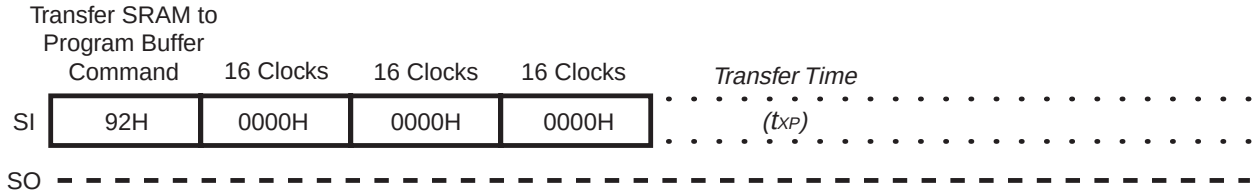
The *Read from SRAM* command (81H) provides access to the 264-Byte SRAM independent of any Flash memory array operations. The command is similar to the *Read from Sector* command except for the sector address field S[15:0] which is replaced with all 0 bits.



*The byte address only uses bits [8:0]

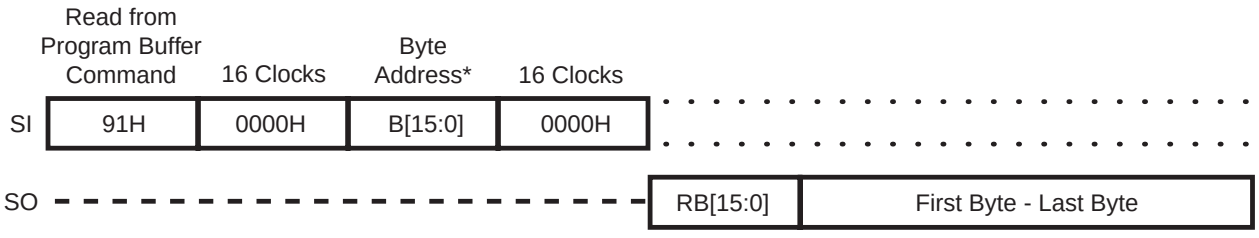
Transfer SRAM to Program Buffer

The *Transfer SRAM to Program Buffer* command transfers all 264 bytes from the SRAM to the Program Buffer at one time without the clock sequencing required in the *Transfer Sector to SRAM* command. This command can be useful in applications where the SRAM and Program Buffer are to be used independently of the Flash memory. Effective use of the *Transfer to SRAM or Program Buffer* commands allow the two 264-byte buffers to act as 528-bytes of user SRAM. The command sequence is similar to the *Write or Read SRAM* commands except that the sector address field S[15:0] and byte address B[15:0] field are replaced with all 0 bits. After the last byte address is transferred, the command is completed by issuing 16 control clocks and then asserting $\overline{CS}$ high. There is a required delay time after $\overline{CS}$ is asserted high (see t_{XP} timing specification). During this time the data from the SRAM is being transferred to the Program Buffer and neither are available for use. Status of this operation can be checked by testing the *Transfer in Process* bit (TR) in the status register.



Read from Program Buffer Command

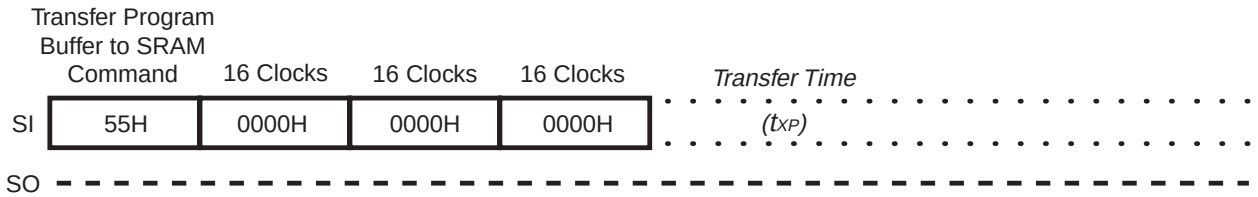
The *Read from Program Buffer* command (91H) provides access to the 264-Byte Program Buffer. The command is similar to the *Read from Sector* command except that the sector address field S[15:0] is replaced with all 0 bits. This command can be useful in applications where the SRAM and Program Buffer are used independently of the Flash memory. This command cannot be used while the device is busy.



*The byte address only uses bits [8:0]

Transfer Program Buffer to SRAM

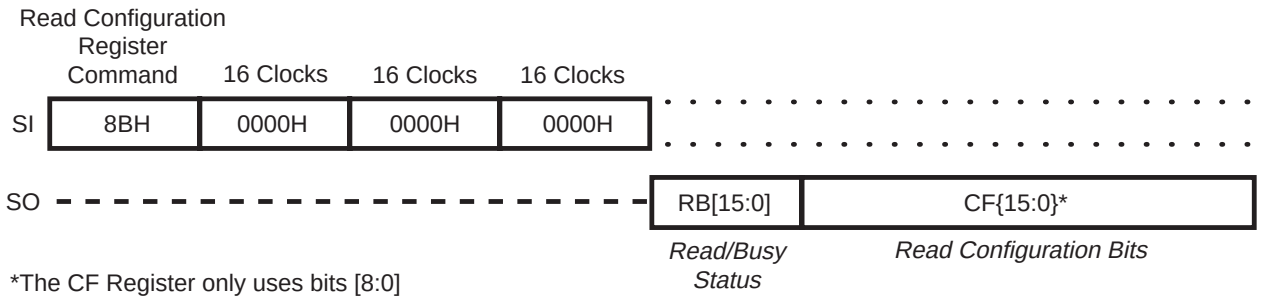
The *Transfer Program Buffer to SRAM* command (55H) provides access to the 264-Byte Program Buffer. The command sequence is similar to the *Write or Read SRAM* commands except that the sector address field S[15:0] and byte address B[15:0] field are replaced with all 0 bits. After the last byte address is transferred, the command is completed by issuing 16 control clocks and then asserting $\overline{CS}$ high. There is a delay time after $\overline{CS}$ is asserted high (see t_{XP} timing specification). During this time the data from the Program Buffer is being transferred to the SRAM and neither are available for use. Status of this operation can be checked by testing the *Transfer in Process* bit (TR) in the status register. This command cannot be used while the device is busy.



CONFIGURATION AND STATUS COMMANDS

Read Configuration Register

The *Read Configuration Register* command provides access to the configuration register, which stores the current configuration of the $\overline{HOLD}$-R/$\overline{B}$ pin, read clock edge, write protect range, and alternate oscillator frequency (Figure 6). The command sequence is similar to the *Read from Sector* command except that the sector address field S[15:0] and the byte-address field B[15:0] are replaced with all 0 bits. After 16 control clocks and after the Ready/Busy status field has been clocked through, a 16-bit Configuration Data field CF[15:0] provides the contents of the Configuration Register. Although the field is 16-bits long, only bits CF[8:0] are used. All other upper bits are reserved for future features.

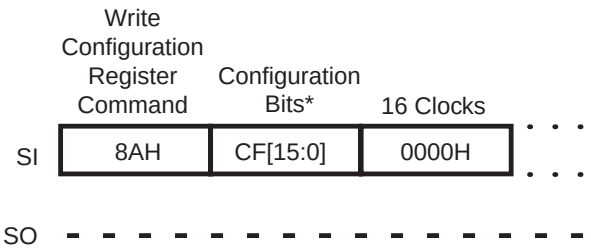


*The CF Register only uses bits [8:0]

Write Configuration Register

The *Write Configuration Register* command provides access to the configuration register which stores the current configuration of the $\overline{HOLD}$-R/$\overline{B}$ pin, read-data clock edge, write protect range, and alternate oscillator frequency. The configuration register is non-volatile. Once set using the *Write Configuration Register* command, the contents will maintain even when power is removed. Because the register's state is stored in non-volatile memory, there is a finite endurance limit to the number of times it can be written to. To limit the number of writes, it is recommended that before writing to the configuration register it should first be read from using the *Read Configuration Register* command. If no change is required, the *Write Configuration Register* command can be skipped. This process will help extend the endurance of the configuration register bits and eliminate additional programming "busy" time.

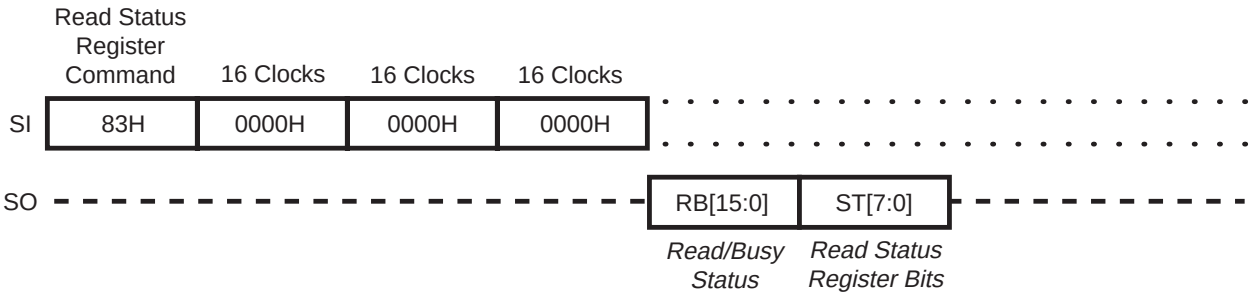
The *Write Configuration Register* command sequence starts with the command byte (8AH) followed by a 16-bit field that specifies configuration register bit settings. Although the field is 16-bits long, only bits CF[8:0] are used. All other upper bits are reserved and must be clocked using 0 for data. After an additional 16 control clocks using 0 for data, the command can be completed by asserting $\overline{CS}$ high. The device will become busy for a short time (t_{WP}) while the non-volatile memory cells of the configuration register are programmed.



*The CF Register only uses bits [8:0]

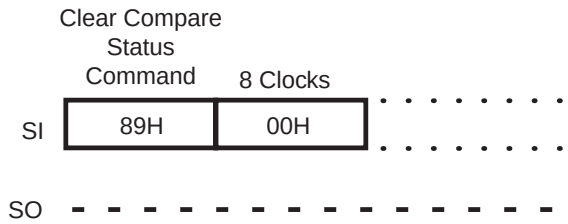
Read Status Register

The *Read Status Register* command provides access to the status register and its status flags for Ready/Busy (R/$\overline{B}$), SRAM and program buffer transfer operations (TX), Write Enable/Disable ($\overline{WE}$), and Compare Not Equal (CNE) (Figure 8). The command sequence is similar to the 0 command except that the sector address field S[15:0] and the byte-address field B[15:0] are replaced by all 0 bits. After 16 clocks and the Ready/Busy status field RB[15:0] has been read, an 8-bit Status field ST[7:0] provides the contents of the Status Register.



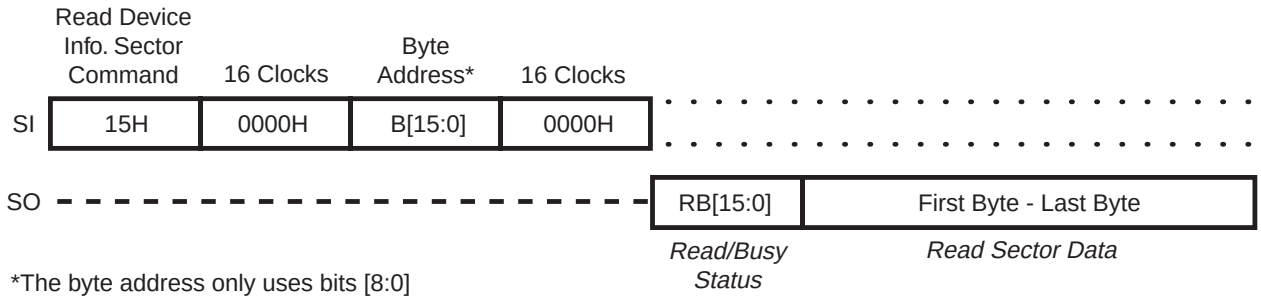
Clear Compare Status

The *Clear Compare Status* command (89H) works in conjunction with the *Compare Sector to SRAM* command and the Status Register. If any of the compared bits are not equal, then the Compare Not Equal (CNE) bit in the Status Register is set to a 1. The *Clear Compare Status* command must be executed to reset the CNE bit to a 0.



Read Device Information Sector

The *Read Device Information* command provides access to a read-only sector that can be used to electronically identify the *NexFlash* Serial Flash device being used. Information available includes: part number, density, voltage, temperature range, package type, and any special options. This can be extremely useful for systems that need to accommodate optional densities (e.g., both 1M-bit and 4M-bit). In this case the firmware can interrogate the Device Information Sector and determine the density. The Device Information Sector also includes a list of any restricted sectors that might exist in the device. Refer to Application Note SFAN-02 for more detailed information on the Device Information Sector format.



*The byte address only uses bits [8:0]

Sector Format and Tag/Sync Bytes

The first byte of each sector is pre-programmed during manufacturing with a tag/sync value of C9H. Although this byte location of the sector can be changed, it is recommended that it be maintained and incorporated into the application's sector formatting.

The tag/sync values serve two purposes. First, they provide a sync-detect that can help verify if the command sequence was clocked into the device properly. Secondly, they serve as a tag to identify a fully functional (valid) sector. This is especially important if "restricted sector" devices are to be used. Restricted sector devices have a limited number of sectors that do not meet manufacturing programming criteria over the specified operating range. When such a sector is detected, the first byte is tagged with a pattern other than C9H. In addition to individual sector tagging, all restricted sectors for a given device are listed in the Device Information Sector. For more information see the Device Information Sector Application Note SFAN-02.

High Data Integrity Applications

Data storage applications that use Flash memory or other non-volatile media must take into consideration the possibility of noise or other adverse system conditions that may affect data integrity. For those applications that require higher levels of data integrity it is a recommended practice to use Error Correcting Code (ECC) techniques. The NexFlash Serial Flash Development Kit provides a software routine for a 32-bit ECC that can detect up to two bit errors and correct one. The ECC not only minimizes problems caused by system noise but can also extend Flash memory endurance. For those systems without the processing power to handle ECC algorithms, a simple "verification after write" is recommended (Figure 12). The Serial Flash Development Kit software includes a simple Write/Verify routine that will compare data written to a given sector and rewrite the sector if the compare is not correct.

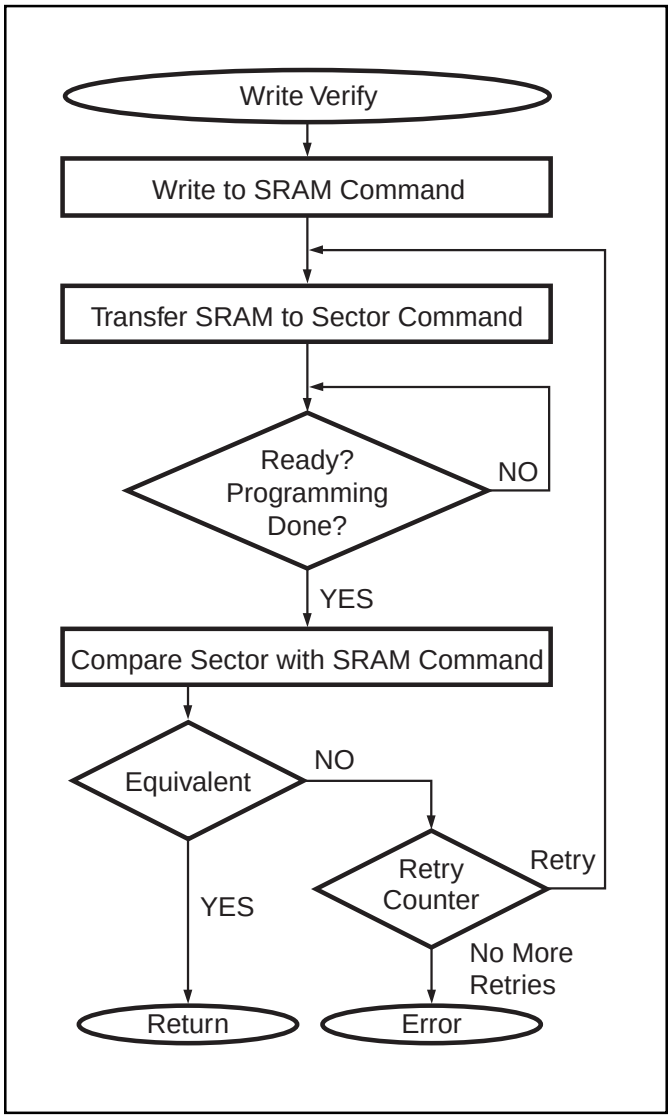


Figure 12. Write/Verify Flow for High Data Integrity Applications

ABSOLUTE MAXIMUM RATINGS (1)

Symbol	Parameters	Conditions	Range	Unit
V_{CC}	Supply Voltage		0 to 7.0	V
V_{IN}, V_{OUT}	Voltage Applied to Any Pin	Relative to Ground	–0.5 to Vcc + 0.5	V
T_{STG}	Storage Temperature		–65 to +150	°C
T_{LEAD}	Lead Temperature	Soldering 10 Seconds	+300	°C

Note:
1. This device has been designed and tested for the specified operation ranges. Proper operation outside of these levels is not guaranteed. Exposure beyond absolute maximum ratings (listed above) may cause permanent damage.

OPERATING RANGES

Symbol	Parameter	Conditions	Min	Max	Unit
Vcc	Supply Voltage	5.0V	4.5	5.5	V
		3.0V	2.7	3.6	V
T_A	Ambient Temperature, Operating	Commercial	0	+70	°C
		Extended	–20	+70	°C
		Industrial	–40	+85	°C

DC ELECTRICAL CHARACTERISTICS

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
V_{IL}	Input Low Voltage			–0.4	—	Vcc x 0.2	V
V_{IH}	Input High Voltage			Vcc x 0.6	—	Vcc + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 2 mA V_{CC} = 4.5V		—	—	0.45	V
V_{OH}	Output High Voltage	I_{OH} = –400 µA V_{CC} = 4.5V		2.4	—	—	V
V_{OLC}	Output Low Voltage CMOS	V_{CC} = Min, I_{OL} = 10 µA		—	—	0.15	V
V_{OHC}	Output High Voltage CMOS	V_{CC} = Min, I_{OH} = –10 µA		V_{CC} – 0.3	—	—	V
I_{IL}	Input Leakage	0 < V_{IN} < Vcc		–10	—	+10	µA
I_{OL}	I/O Leakage	0 < V_{IN} < Vcc		–10	—	+10	µA
I_{CC} (active)	Active Power Supply Current	f_{CLK} @ 8 MHz (1/t_{CP})	V_{CC} = 5V	—	15	30	mA
			V_{CC} = 3V	—	5	10	mA
I_{CCLF}	Active Current Low	f_{CLK} @1 MHz (1/t_{CP})	V_{CC} = 5V	—	10	20	mA
(low frequency)	Frequency. Read		V_{CC} = 3V	—	4	7	mA
I_{CCSB} (standby)	Standby Power Supply Current	$\overline{CS}$ = V_{CC}, V_{IN} = Vcc or 0		—	<1	10	µA
C_{IN}	Input Capacitance (1)	T_A = 25°C, V_{CC} = 5V or 3V		—	—	10	pF
		Frequency = 1 MHz					
C_{OUT}	Output Capacitance (1)	T_A = 25°C, V_{CC} = 5V or 3V		—	—	10	pF
		Frequency = 1 MHz					

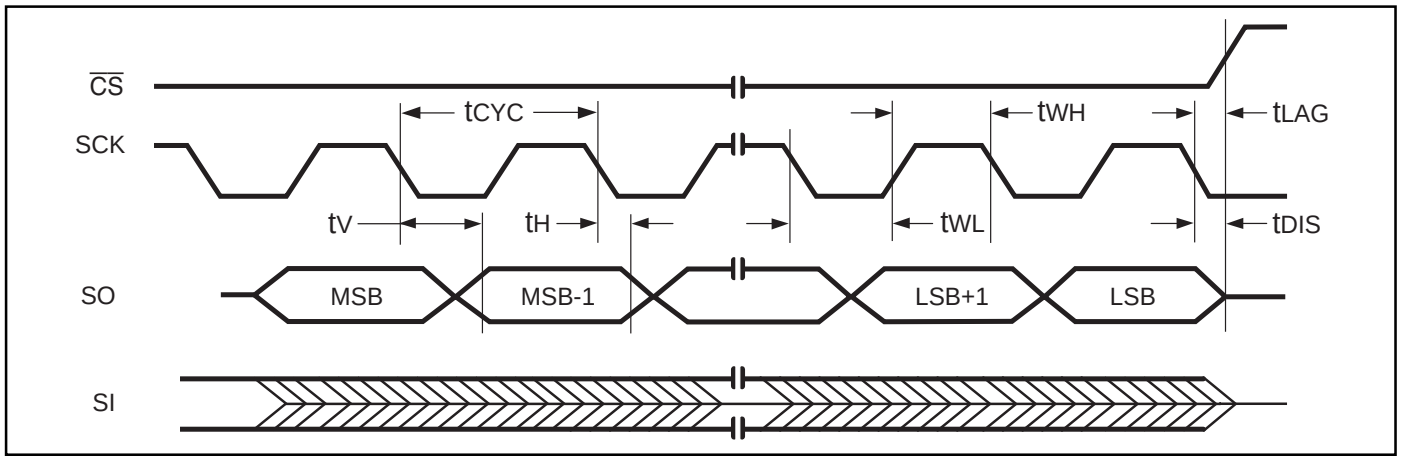
Note:
1. Tested on a sample basis or specified via design or characterization data.

AC ELECTRICAL CHARACTERISTICS

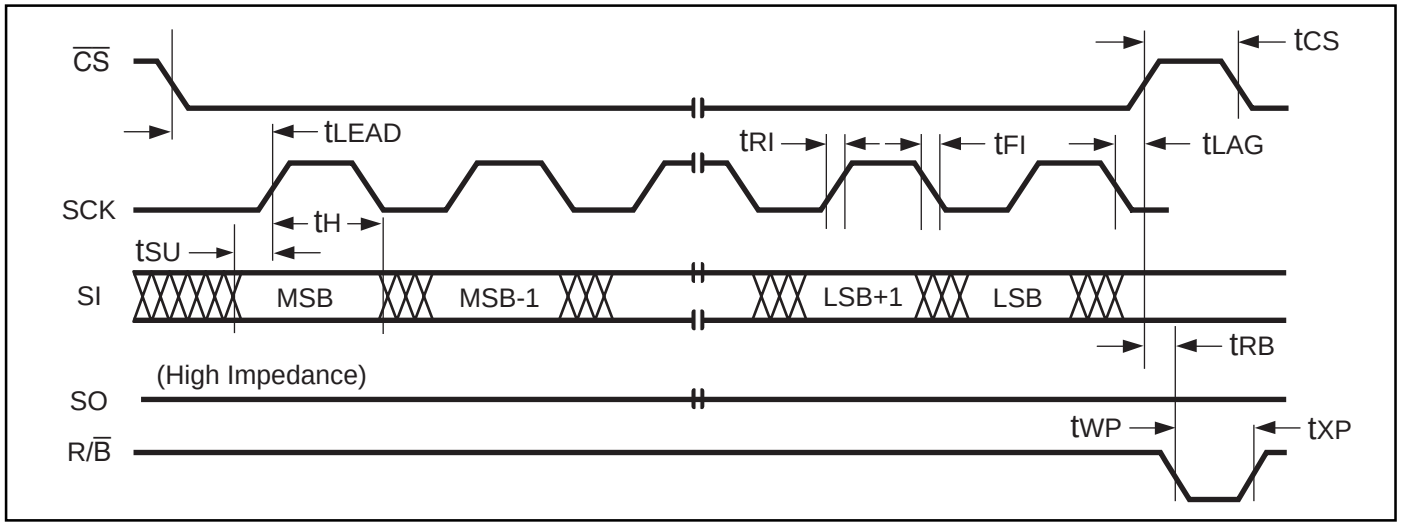
			16 MHz			8 MHz			
Symbol	**Description**		**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Unit**
t_{CYC}	SCK Serial Clock Period [1]		62	—	—	125	—	—	ns
t_{WH}	SCK Serial Clock High or Low Time		26	—	—	57	—	—	ns
t_{WL}									
t_{RI}	SCK Serial Clock Rise or Fall Time [2]		—	—	7	—	—	10	ns
t_{FI}									
t_{SU}	Data Input Setup Time to SCLK		40	—	—	100	—	—	ns
t_H	Data Input Hold Time from SCLK		0	—	—	0	—	—	ns
t_V	Data Output Valid after SCLK [1,3]	Vcc = 5.0V	—	—	60	—	—	70	ns
		Vcc = 3.0V	—	—	—	—	—	115	ns
t_{LEAD}	$\overline{CS}$ Setup Time to Command	Vcc = 5.0V	100	—	—	150	—	—	ns
		Vcc = 3.0V	—	—	—	300	—	—	ns
t_{LAG}	$\overline{CS}$ Delay Time after Command	Vcc = 5.0V	100	—	—	150	—	—	ns
		Vcc = 3.0V	—	—	—	300	—	—	ns
t_{WP}	Erase/Write Program Time	Vcc = 5.0V	—	5	10	—	5	10	ms
	(see Write to Sector Command)	Vcc = 3.0V	—	5	—	—	5	10	ms
t_{XP}	Transfer Program-Buffer/SRAM	Vcc = 5.0V	—	—	100	—	—	100	µs
	(see Transfer PB/SRAM Command)	Vcc = 3.0V	—	—	—	—	—	200	µs
t_{HD}	SCK Setup Time to $\overline{HOLD}$		10	—	—	20	—	—	ns
t_{CD}	SCK Hold Time from $\overline{HOLD}$		30	—	—	50	—	—	ns
t_{CS}	$\overline{CS}$ Deselect Time		160	—	—	200	—	—	ns
t_{RB}	READY / BUSY Valid Time		160	—	—	200	—	—	ns
t_{DIS}	Data Output Disable Time		—	—	160	—	—	200	ns
t_{HO}	Data Output Hold Time		0	—	—	0	—	—	ns

Notes:
1. To achieve maximum clock performance, the read clock edge will need to be set for rising edge operation in the configuration register (RCE=1).
2. Test points are 10% and 90% points for rise/fall times. All others timings are measured at 50% point.
3. With 50 pF (8 MHz) or 30 pF (16 MHz) load SO to GND.

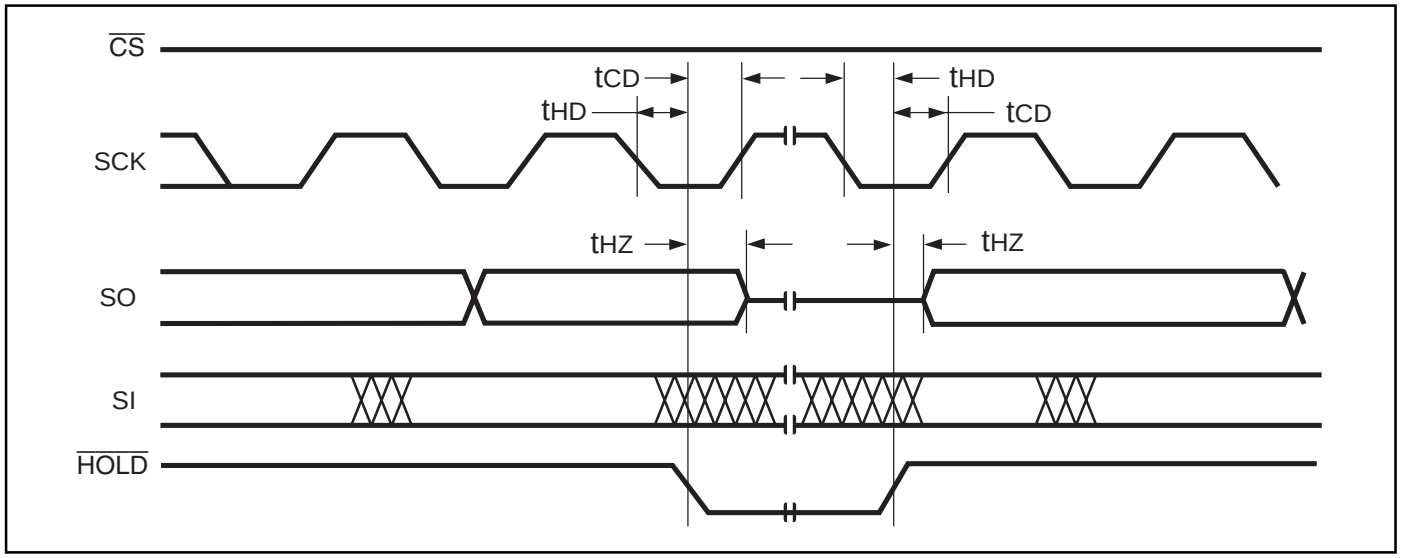
SERIAL OUTPUT TIMING



SERIAL INPUT TIMING

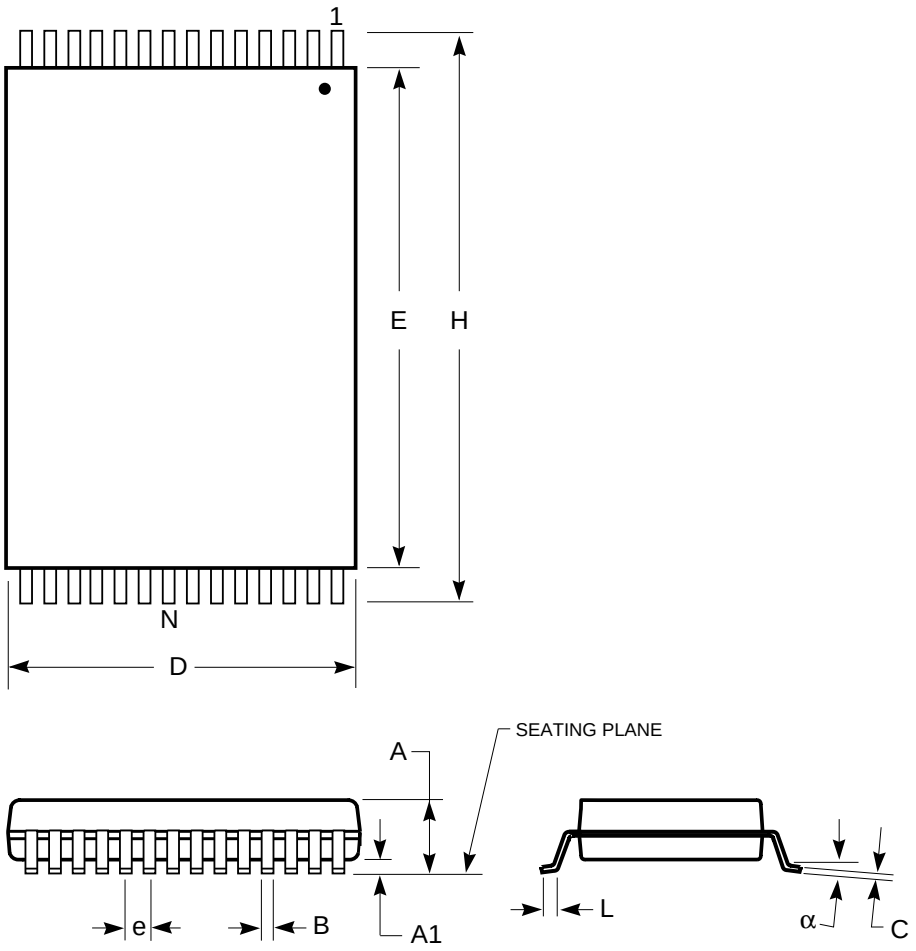


HOLD TIMING



PACKAGE INFORMATION

Plastic TSOP - 28-pins
Package Code: V (Type I)



Plastic TSOP (V—Type I)				
	Millimeters		Inches	
Symbol	Min	Max	Min	Max
Ref. Std.				
No. Leads		28		
A	1.00	1.20	0.039	0.047
A1	0.05	0.20	0.002	0.008
B	0.15	0.25	0.006	0.010
C	0.10	0.20	0.004	0.008
D	7.90	8.10	0.311	0.319
E	11.60	11.80	0.457	0.465
H	13.30	13.50	0.524	0.531
e	0.55 BSC		0.022 BSC	
L	0.50	0.70	0.020	0.028
α	0°	5°	0°	5°

Notes:
1. Controlling dimension: millimeters, unless otherwise specified.
2. BSC = Basic lead spacing between centers.
3. Dimensions D and E do not include mold flash protrusions and should be measured from the bottom of the package.
4. Formed leads shall be planar with respect to one another within 0.004 inches at the seating plane.

ORDERING INFORMATION

Size	Order Part No. (1)	Package (2)
1M-bit	NX25F011A-3V-R	SPI, 28-pin, TSOP (Type I) <32 RS, 3V Low Voltage
1M-bit	NX25F011A-3V	SPI, 28-pin, TSOP (Type I) 3V Low Voltage (no restricted sectors)
1M-bit	NX25F011A-5V-R	SPI, 28-pin, TSOP (Type I) <32 RS, 5V Standard Voltage
1M-bit	NX25F011A-5V	SPI, 28-pin, TSOP (Type I) 5V Standard Voltage (no restricted sectors)
4M-bit	NX25F041A-3V-R	SPI, 28-pin, TSOP (Type I) <32 RS, 3V Low Voltage
4M-bit	NX25F041A-5V-R	SPI, 28-pin, TSOP (Type I) <32 RS, 5V Standard Voltage

Note:
1. Add E (Extended) or I (Industrial) after package designator (V) for alternative temperature grade devices.
2. For Serial Flash Module package see 25Mxxx data sheet.
3. RS = Restricted Sector.

PRELIMINARY DESIGNATION

The “Preliminary” designation on an *NexFlash* data sheet indicates that the product is not fully characterized. The specifications are subject to change and are not guaranteed. *NexFlash* or an authorized sales representative should be consulted for current information before using this product.

IMPORTANT NOTICE

NexFlash reserves the right to make changes to the products contained in this publication in order to improve design, performance or reliability. *NexFlash* assumes no responsibility for the use of any circuits described herein, conveys no license under any patent or other right, and makes no representation that the circuits are free of patent infringement. Charts and schedules contained herein reflect representative operating parameters, and may vary depending upon a user’s specific application. While the information in this publication has been carefully checked, *NexFlash* shall not be liable for any damages arising as a result of any error or omission.

LIFE SUPPORT POLICY

NexFlash does not recommend the use of any of it's products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure in the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless *NexFlash* receives written assurances, to it’s satisfaction, that:

(a) the risk of injury or damage has been minimized;

(b) the user assumes all such risks; and

(c) potential liability of *NexFlash* is adequately protected under the circumstances.

3114 Scott Blvd • Santa Clara, CA 95054
Phone: 408-969-4700 • Fax: 408-588-0819
sales@nexflash.com • www.nexflash.com