# THE DATASHEET OF

# CY14V101QS-BK108XQ

CY14V101QS

# 1-Mbit (128K × 8) Quad SPI nvSRAM

## Features

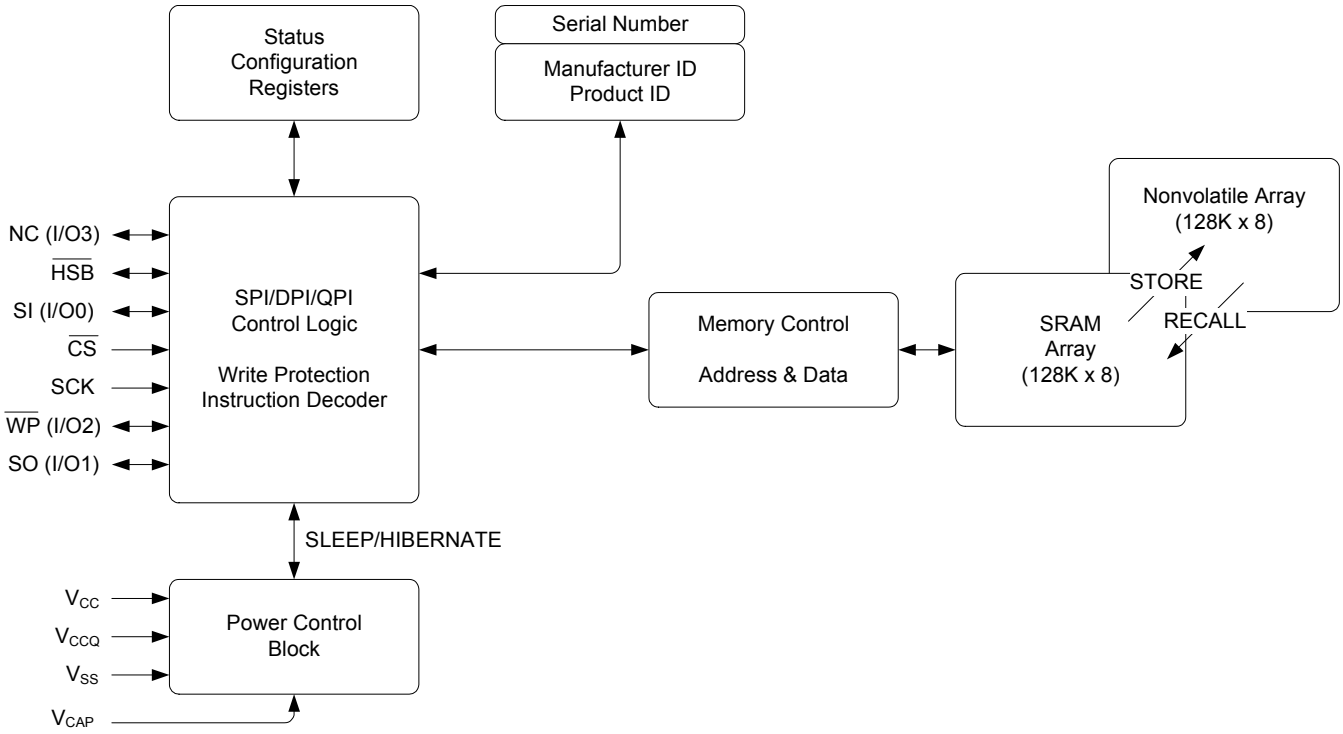
- Density
  - 1-Mbit (128K × 8)
- Bandwidth
  - 108-MHz high-speed interface
  - Read and write at 54 MBps
- Serial Peripheral Interface
  - Clock polarity and phase modes 0 and 3
  - Multi I/O option – Single SPI (SPI), Dual SPI (DPI), and Quad SPI (QPI)
- High reliability
  - Infinite read, write, and RECALL cycles
  - One million STORE cycles to nonvolatile elements (SONOS FLASH Quantum trap)
  - Data retention: 20 years at 85 °C
- Read
  - Commands: Standard, Fast, Dual I/O, and Quad I/O
  - Modes: Burst Wrap, Continuous (XIP)
- Write
  - Commands: Standard, Fast, Dual I/O, and Quad I/O
  - Modes: Burst Wrap
- Data protection
  - Hardware: Through Write Protect Pin ($\overline{WP}$)
  - Software: Through Write Disable instruction
  - Block Protection: Status Register bits to control protection
- Special instructions
  - STORE/RECALL: Access data between SRAM and Quantum Trap
  - Serial Number: 8-byte customer selectable (OTP)
  - Identification Number: 4-byte Manufacturer ID and Product ID
- Store from SRAM to nonvolatile SONOS FLASH Quantum Trap
  - AutoStore: Initiated automatically at power-down with a small capacitor ($V_{CAP}$)
  - Software: Using SPI instruction (STORE)
  - Hardware: $\overline{HSB}$ pin
- Recall from nonvolatile SONOS FLASH Quantum Trap to SRAM
  - Auto RECALL: Initiated automatically at power-up
  - Software: Using SPI instruction (RECALL)
- Low-power modes
  - Sleep: Average current = 280 µA at 85 °C
  - Hibernate: Average current = 8 µA at 85 °C
- Operating supply voltages
  - Core $V_{CC}$: 2.7 V to 3.6 V
  - I/O $V_{CCQ}$: 1.71 V to 2.0 V
- Temperature range
  - Extended Industrial: –40 °C to 105 °C
  - Industrial: –40 °C to 85 °C
- Packages
  - 16-pin SOIC
  - 24-ball FBGA

## Functional Overview

The Cypress CY14V101QS combines a 1-Mbit nvSRAM with a QPI interface. The QPI allows writing and reading the memory in either a single (one I/O channel for one bit per clock cycle), dual (two I/O channels for two bits per clock cycle), or quad (four I/O channels for four bits per clock cycle) through the use of selected opcodes.

The memory is organized as 128Kbytes each consisting of SRAM and nonvolatile SONOS Quantum Trap cells. The SRAM provides infinite read and write cycles, while the nonvolatile cells provide highly reliable storage of data. Data transfers from SRAM to the nonvolatile cells (STORE operation) take place automatically at power-down. On power-up, data is restored to the SRAM from the nonvolatile cells (RECALL operation). You can also initiate the STORE and RECALL operations through SPI instructions.

## Logic Block Diagram



# Contents

## Pinout

**Figure 1. 16-Pin SOIC Standard Pinout**

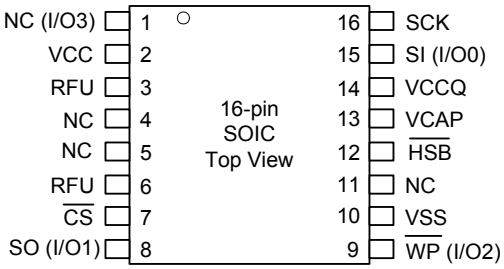



**Figure 2. 16-Pin SOIC Custom Pinout**

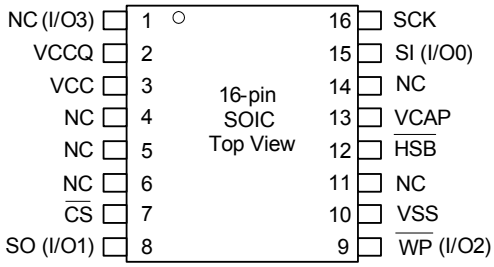
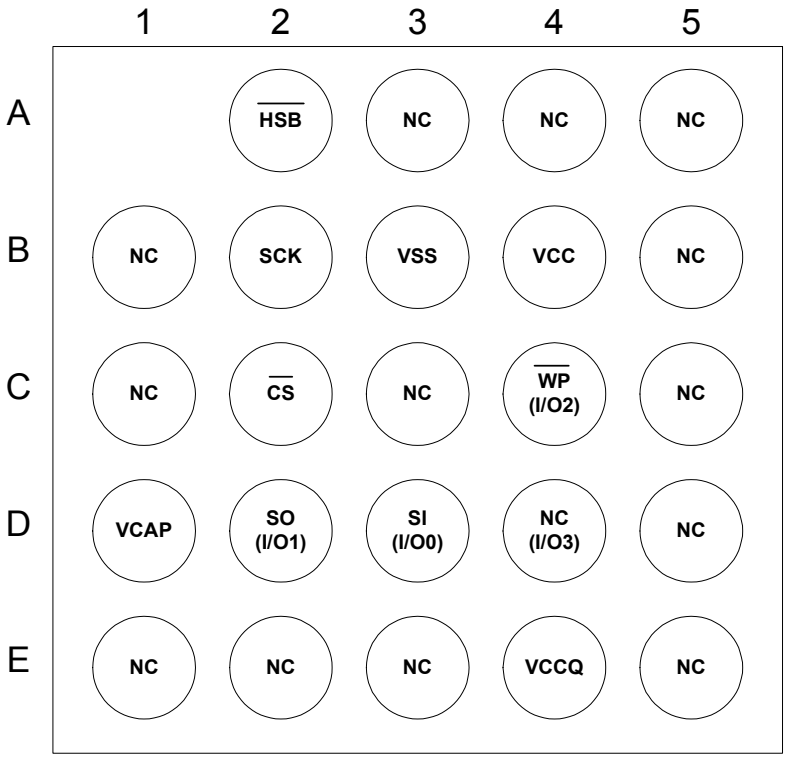



**Figure 3. 24-Ball FBGA Standard Pinout-Top View (Ball Side Down)**



## Pin Definitions

| Pin Name | I/O Type | Description |
|---|---|---|
| NC (I/O3) | Input | Not connected. In Single or Dual mode, this pin is not connected and left floating. This mode does not support QSPI instructions. |
| | Input/Output | I/O3: When the part is in Quad mode, the NC (I/O3) pin becomes I/O3 pin and acts as input/output.<br>In Quad mode supporting SPI/DPI instructions, this pin needs to be tri-stated while $\overline{CS}$ is enabled. |
| $V_{CCQ}$ | Power Supply | Power supply for the I/Os of the device. |
| $V_{CC}$ | Power Supply | Power supply to the core of the device. |
| $\overline{CS}$ | Input | Chip Select. Activates the device when pulled LOW. Driving this pin HIGH puts the device in standby state. |
| SO (I/O1) | Output | Serial Output. Pin for output of data through SPI. |
| | Input/Output | I/O1: When the part is in dual or quad mode, the SO (I/O1) pin becomes I/O1 pin and acts as input/output. |
| $\overline{WP}$ (I/O2) | Input | Write Protect. Implements hardware write-protection in SPI/DPI modes. |
| | Input/Output | I/O2: When the part is in quad mode, the $\overline{WP}$ (I/O2) pin becomes an I/O2 pin and acts as input/output. |
| $V_{SS}$ | Ground | Ground power supply to the core and I/Os of the device. |
| $\overline{HSB}$ | Input/Output | Hardware STORE Busy:<br>Output: Indicates the busy status of nvSRAM when LOW. After each Hardware and Software STORE operation, $\overline{HSB}$ is driven HIGH for a short time ($t_{HHHD}$) with standard output HIGH current and then a weak internal pull-up resistor keeps this pin HIGH (external pull-up resistor connection is optional).<br>Input: Hardware STORE implemented by pulling this pin LOW externally. |
| $V_{CAP}$ | Power Supply | AutoStore Capacitor. Supplies power to the nvSRAM during power loss to STORE data from the SRAM to nonvolatile elements. If AutoStore is not needed, this pin must be left as No Connect. It must never be connected to ground. |
| SI (I/O0) | Input | Serial Input. Pin for input of all SPI instructions and data. |
| | Input/Output | I/O0: When the part is in dual or quad mode, the SI (I/O0) pin becomes I/O0 pin and acts as input/output. |
| SCK | Input | Serial Clock. Runs at speeds up to a maximum of $f_{SCK}$. Serial input is latched at the rising edge of this clock. Serial output is driven at the falling edge of the clock. |
| NC | – | Not connected. |
| RFU | – | Reserved for future use. |

## Device Operation

CY14V101QS is a 1-Mbit quad serial interface nvSRAM memory with a SONOS FLASH nonvolatile element interleaved with an SRAM element in each memory cell. All the reads and writes to nvSRAM happen to the SRAM, which gives nvSRAM the unique capability to handle infinite writes to the memory. The data in SRAM is secured by a STORE sequence, which transfers the data to the nonvolatile cells. A small capacitor ($V_{CAP}$) is used to AutoStore the SRAM data into the nonvolatile cells when power goes down providing data integrity. The nonvolatile cells are built in the reliable SONOS technology make nvSRAM the ideal choice for data storage.

The 1-Mbit memory array is organized as 128Kbytes. The memory can be accessed through a standard SPI interface (Single mode, Dual mode, and Quad mode) up to clock speeds of 40-MHz with zero-cycle latency for read and write operations. This SPI interface also supports 108-MHz operations (Single mode, Dual mode, and Quad mode) with cycle latency for read operations only. The device operates as a SPI slave and supports SPI modes 0 and 3 (CPOL, CPHA = [0, 0] and [1, 1]). All instructions are executed using Chip Select (CS), Serial Input (SI) (I/O0), Serial Output (SO) (I/O1), and Serial Clock (SCK) pins in single and dual modes. Quad mode uses WP I/O2 and I/O3 pins as well for command, address, and data entry.

The device uses SPI opcodes for memory access. The opcodes support SPI, Dual Data, Dual Addr/Data, Dual I/O, Quad Data, Quad Addr/Data, and Quad I/O modes for read and write operations. In addition, four special instructions are included that allow access to nvSRAM specific functions: STORE, RECALL, AutoStore Disable (ASDI), and AutoStore Enable (ASEN).

The device has built-in data security features. It provides hardware and software write-protection through the $\overline{WP}$ pin and WRDI instruction respectively. Furthermore, the memory array block is write-protected through Status register block protect bits.

### SRAM Write

All writes to nvSRAM are carried out on the SRAM cells and do not use any endurance cycles of the SONOS FLASH nonvolatile memory. This allows you to perform infinite write operations. A write cycle is initiated through one of the Write instructions: WRITE, DIW, QIW, DIOW, and QIOW. The Write instructions consist of a write opcode, three bytes of address, and one byte of data. Write to nvSRAM is done at SPI bus speed with zero-cycle latency.

The device allows burst mode writes. This enables write operations on consecutive addresses without issuing a new Write instruction. When the last address in memory is reached in burst mode, the address rolls over to 0x00000 and the device continues to write.

The SPI write cycle sequence is defined explicitly in the nvSRAM Read Write Instructions in “SPI Functional Description” on page 12.

### SRAM Read

All reads to nvSRAM are carried out on the SRAM cells at SPI bus speeds. Read instruction (READ) executes at 40-MHz with zero cycle latency. It consists of a Read opcode byte followed by three bytes of address. The data is read out on the data output pin/pins.

Speeds higher than 40 MHz (up to 108 MHz) require Fast Read instructions: FAST_READ, DOR, QOR, DIOR, and QIOR. The Fast Read instructions consist of a Fast Read opcode byte, three bytes of address, and a dummy/mode byte. The data is read out on the data output pin/pins.

The device allows burst mode reads. This enables read operations on consecutive addresses without issuing a new Read instruction. When the last address in memory is reached in burst mode, the address rolls over to 0x00000 and the device continues to read.

The SPI read cycle sequence is defined explicitly in the nvSRAM Read Write Instructions in “SPI Functional Description” on page 12.

### STORE Operation

STORE operation transfers the data from the SRAM to the nonvolatile cells. The device stores data using one of the three STORE operations: AutoStore, activated on device power-down (requires $V_{CAP}$); Software STORE, activated by a STORE instruction; and Hardware STORE, activated by the $\overline{HSB}$ pin. During the STORE cycle, the nonvolatile cell is first erased and then programmed. After a STORE cycle is initiated, read/write to the device is inhibited until the cycle is completed.

The $\overline{HSB}$ signal or the WIP bit in Status Register can be monitored by the system to detect if a STORE cycle is in progress. The busy status of nvSRAM is indicated by $\overline{HSB}$ being pulled LOW or the WIP bit being set to ‘1’. To avoid unnecessary nonvolatile STOREs, AutoStore and Hardware STORE operations are ignored unless at least one SRAM write operation has taken place since the most recent STORE cycle. However, software initiated STORE cycles are performed regardless of whether a SRAM write operation has taken place.

### AutoStore Operation

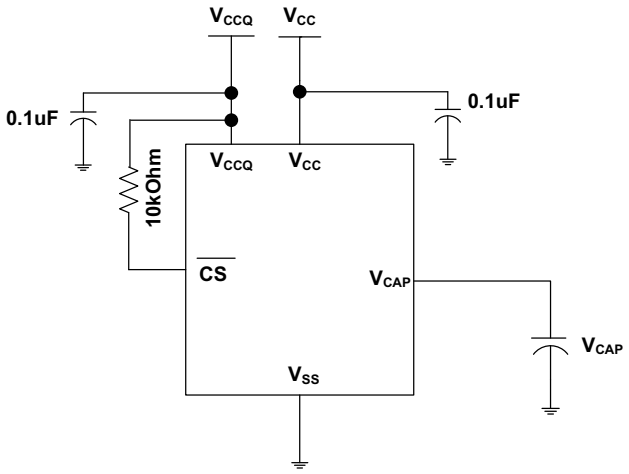
The AutoStore operation is a unique feature of nvSRAM, which automatically stores the SRAM data to the SONOS FLASH nonvolatile cells during power-down. This STORE makes use of an external capacitor ($V_{CAP}$) and enables the device to safely STORE the data in the nonvolatile memory when power goes down.

During normal operation, the device draws current from $V_{CC}$ to charge the capacitor connected to the $V_{CAP}$ pin. When the voltage on the $V_{CC}$ pin drops below $V_{SWITCH}$ during power-down, the device inhibits all memory accesses to nvSRAM and automatically performs a STORE operation using the charge from the $V_{CAP}$ capacitor. The AutoStore operation is not initiated if a write cycle has not been performed since last RECALL.

**Note** If a capacitor is not connected to the $V_{CAP}$ pin, AutoStore must be disabled by issuing the AutoStore Disable instruction (Autostore Disable (ASDI) Instruction on page 42). If AutoStore is enabled without a capacitor on the $V_{CAP}$ pin, the device attempts AutoStore without sufficient charge to complete the operation. This will corrupt the data stored in the memory array along with the serial number and Status Register. Updating them will be required to resume normal functionality.

Figure 4 shows the connection of the storage capacitor ($V_{CAP}$) for AutoStore operation. Refer to on page 43 for the size of the $V_{CAP}$.

**Figure 4. AutoStore Mode**




## Software STORE Operation

Software STORE allows an instruction-based STORE operation. It is initiated by executing a STORE instruction, irrespective of whether a write has been previously performed.

A STORE cycle takes $t_{STORE}$ time to complete, during which all the memory accesses to nvSRAM are inhibited. The WIP bit of the Status Register or the $\overline{HSB}$ pin may be polled to find the Ready or Busy status. After the $t_{STORE}$ cycle time is completed, the nvSRAM is ready for normal operations.

## Hardware STORE and $\overline{HSB}$ Pin Operation

The $\overline{HSB}$ pin in the device is a dual-purpose pin used to either initiate a STORE operation or to poll STORE/RECALL completion status. If a STORE or RECALL is not in progress, the $\overline{HSB}$ pin can be driven low to initiate a Hardware STORE cycle.

Detecting a low on $\overline{HSB}$, nvSRAM will start a STORE operation after $t_{DELAY}$ duration. A hardware STORE cycle is only possible if a SRAM write operation has been performed since the last STORE/RECALL cycle. This allows for optimizing the SONOS FLASH endurance cycles. All reads and writes to the memory are inhibited for $t_{STORE}$ duration. The $\overline{HSB}$ pin also acts as an open drain driver (internal 100-kΩ weak pull-up resistor) that is internally driven LOW to indicate a busy condition when the STORE/RECALL is in progress.

**Note** After each Hardware and Software STORE operation, $\overline{HSB}$ is driven HIGH for a short time ($t_{HHHD}$) with standard output HIGH current and then remains HIGH by an internal 100-kΩ pull-up resistor.

**Note** For successful last data byte STORE, a hardware STORE should be initiated at least one clock cycle after the last data bit D0 is received.

**Note** It is recommended to perform a Hardware STORE only when the device is in Standby state. Execute-in-place (XIP) should be exited as well.

Upon completion of the STORE operation, the nvSRAM memory access is inhibited for $t_{LZHSB}$ time after $\overline{HSB}$ pin returns HIGH. The $\overline{HSB}$ pin must be left unconnected if not used.

## RECALL Operation

A RECALL operation transfers the data stored in the nonvolatile cells to the SRAM cells. A RECALL may be initiated in two ways: Hardware RECALL, initiated on power-up and Software RECALL, initiated by a SPI RECALL instruction.

Internally, RECALL is a two-step procedure. First, the SRAM data is cleared (set to '0'). Next, the nonvolatile information is transferred into the SRAM cells. All memory accesses are inhibited while a RECALL cycle is in progress. The RECALL operation does not alter the data in the nonvolatile elements.

## Hardware RECALL (Power-Up)

During power-up, when $V_{CC}$ crosses $V_{SWITCH}$, an automatic RECALL sequence is initiated, which transfers the content of nonvolatile cells to the SRAM cells.

A Power-Up RECALL cycle takes $t_{FA}$ time to complete and the memory access is disabled during this time. The $\overline{HSB}$ pin is used to detect the ready status of the device.

## Software RECALL

Software RECALL allows you to initiate a RECALL operation to restore the content of the nonvolatile memory to the SRAM. A Software RECALL is issued by using the RECALL instruction.

A Software RECALL takes $t_{RECALL}$ time to complete during which all memory accesses to nvSRAM are inhibited.

## Disabling and Enabling AutoStore

If the application does not require the AutoStore feature, it can be disabled by using the ASDI instruction. If this is done, the nvSRAM does not perform a STORE operation at power-down.

AutoStore can be re-enabled by using the ASEN instruction. However, ASEN and ASDI operations require a STORE operation to make them nonvolatile.

**Note** The device has AutoStore enabled and 0x00 written to all cells from the factory.

**Note** If AutoStore is disabled and $V_{CAP}$ is not required, the $V_{CAP}$ pin must be left open. The $V_{CAP}$ pin must never be connected to ground. The Power-Up RECALL operation cannot be disabled.

# Quad Serial Peripheral Interface

## SPI Overview

The SPI is a four-pin interface with Chip Select ($\overline{CS}$), Serial Input (SI), Serial Output (SO), and Serial Clock (SCK) pins. The device provides serial access to the nvSRAM through the SPI interface. The SPI bus on the device can run at speed up to 108 MHz.

The SPI is a synchronous serial interface, which uses clock and data pins for memory access and supports multiple devices on the data bus. A device on the SPI bus is activated using the $\overline{CS}$ pin.

The relationship between chip select, clock, and data is dictated by the SPI mode. This device supports SPI modes 0 and 3. In both these modes, data is clocked into the nvSRAM on the rising edge of SCK starting from the first rising edge after $\overline{CS}$ goes active.

The SPI protocol is controlled by opcodes. These opcodes specify the commands from the bus master to the slave device. After $\overline{CS}$ is activated, the first byte transferred from the bus master is the opcode. Following the opcode, any addresses and data are then transferred. The $\overline{CS}$ must go inactive after an operation is complete and before a new opcode can be issued. The commonly used terms in the SPI protocol are described in the following sections.

*SPI Master*

The SPI master device controls the operations on an SPI bus. An SPI bus may have only one master with one or more slave devices. All the slaves share the same SPI bus lines and the master may select any of the slave devices with its own $\overline{CS}$ pin. All the operations must be initiated by the master activating a slave device by pulling the $\overline{CS}$ pin of the slave LOW. The master also generates the SCK and all the data transmission on SI and SO lines are synchronized with this clock.

*SPI Slave*

The SPI slave device is activated by the master through the Chip Select line. A slave device gets the SCK as an input from the SPI master and all the communication is synchronized with this clock. The SPI slave never initiates a communication on the SPI bus and acts on the instruction from the master.

The device operates as an SPI slave and may share the SPI bus with other SPI slave devices.

*Chip Select ($\overline{CS}$)*

For selecting any slave device, the master needs to pull down the corresponding $\overline{CS}$ pin. Any instruction can be issued to a slave device only while the $\overline{CS}$ pin is LOW. When the device is not selected, data through the SI pin is ignored and the serial output pin (SO) remains in a high-impedance state.

**Note** A new instruction must begin with the falling edge of $\overline{CS}$. Therefore, only one opcode can be issued for each active Chip Select cycle.

**Note** It is recommended to attach an external 10-kΩ pull-up resistor to $V_{CCQ}$ on $\overline{CS}$ pin.

*Serial Clock (SCK)*

The serial clock is generated by the SPI master and the communication is synchronized with this clock after $\overline{CS}$ goes LOW.

The device enables SPI modes 0 and 3 for data communication. In both these modes, the inputs are latched by the slave device on the rising edge of SCK and outputs are issued on the falling edge. Therefore, the first rising edge of SCK signifies the arrival of the first bit (MSB) of SPI instruction on the SI pin. Further, all data inputs and outputs are synchronized with SCK.

*Data Transmission - SI/SO*

The SPI data bus consists of two lines, SI and SO, for serial data communication. The SI is also referred to as Master Out Slave In (MOSI) and SO is referred to as Master In Slave Out (MISO). The master issues instructions to the slave through the SI pin, while the slave responds through the SO pin. Multiple slave devices may share the SI and SO lines as described earlier.

The device has two separate pins for SI and SO, which can be connected with the master as shown in Figure 5 on page 9.

This SI input signal is used to transfer data serially into the device. It receives opcode, addresses, and data to be programmed. Values are latched on the rising edge of serial SCK clock signal. SI becomes I/O0 - an input and output during Extended-SPI and DPI/QPI commands for receiving opcodes, addresses, and data to be written (values latched on rising edge of serial SCK clock signal) as well as shifting out data (on the falling edge of SCK).

The SO output signal is used to transfer data serially out of the device. Data is shifted out on the falling edge of the serial SCK clock signal. SO becomes I/O1 - an input and output during Extended-SPI and DPI/QPI commands for receiving opcodes, addresses, and data to be programmed (values latched on rising edge of serial SCK clock signal) as well as shifting out data (on the falling edge of SCK). SO has a Repeater/Bus-Hold circuit implemented.

*Write-Protect ($\overline{WP}$)*

In SPI mode, the $\overline{WP}$ pin when driven low protects against writes to the Status registers and all data bytes in the memory area that are protected by the Block Protect bits in the Status registers.

When $\overline{WP}$ is driven Low, during a WRSR command and while the Status Register Write Disable (SRWD) bit of the Status Register is set to a 1, it is not possible to write to the Status and Configuration Registers. This prevents any alteration of the Block Protect (BP2, BP1, BP0) and TBPROT bits. As a consequence, all the data bytes in the memory area that are protected by the Block Protect and TBPROT bits, are protected against data modification if $\overline{WP}$ is Low during a WRSR command.

The $\overline{WP}$ function is not available while in the Quad transfer mode. The $\overline{WP}$ function is replaced by I/O2 for input and output during these modes for receiving opcode, addresses, and data to be written/programmed as well as shifting out data. $\overline{WP}$ has an internal pull-up resistor; and may be left unconnected in the host system if not used for Quad transfer mode. $\overline{WP}$ has an internal 100-kΩ weak pull-up resistor in SPI mode.

*NC (I/O3)*

The NC (I/O3) pin functions as I/O3 for input and output during Quad transfer modes for receiving opcode, addresses, data to be written/programmed and shifting out data. NC (I/O3) has an internal pull-up resistor; and may be left unconnected in the host system if not used for Quad transfer mode. NC (I/O3) has an internal 100-kΩ weak pull-up resistor in SPI mode.

*Most Significant Bit (MSB)*

The SPI protocol requires that the first bit to be transmitted is the MSB. This is valid for both address and data transmission.

The 1-Mbit serial nvSRAM requires a 3-byte address for any read or write operation. However, because the address is only 17 bits, it implies that the first seven bits that are fed in are ignored by the device. Although these seven bits are 'don't care', Cypress recommends that these bits are treated as 0s to enable seamless transition to higher memory densities.

*Serial Opcode*

After the slave device is selected with $\overline{CS}$ going LOW, the first byte received is treated as the opcode for the intended operation. The device uses the standard opcodes for memory accesses. In addition to the memory accesses, it provides additional opcodes for the nvSRAM specific functions: STORE, RECALL, AutoStore Enable, and AutoStore Disable. Refer to Table 2 on page 12 for details.

*Invalid Opcode*

If an invalid opcode is received, the opcode is ignored and the device ignores any additional serial data on the SI pin until the next falling edge of $\overline{CS}$ and the SO pin remains tristated.

*Instruction*

The combination of the opcode, address, and mode/dummy cycles used to issue a command.

*Mode Bits*

Control bits that follow the address bits. The device uses control bits to enable execute-in-place (XIP). These bits are driven by the system controller when they are specified.

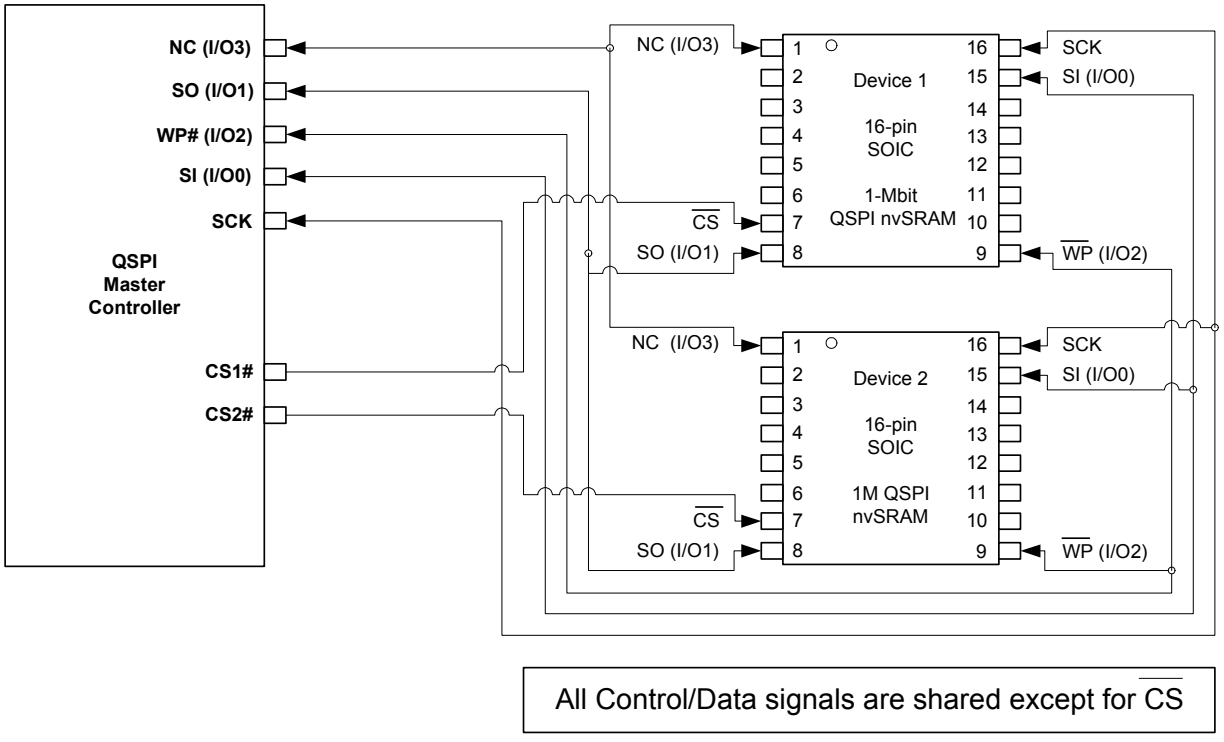
*Wait States*

Required dummy clock cycles after the address bits or optional mode bits.

*Status Register*

The device has one 8-bit Status Register. The bits in the Status Registers are used to configure the SPI bus. These bits are described in Table 3 and Table 4 on page 14.

**Figure 5. System Configuration Using Multiple 1-Mbit Quad SPI nvSRAM Devices**



## Dual and Quad I/O Modes

The device also has the capability to reconfigure the standard SPI pins to work in dual or quad I/O modes.

When the part is in the dual I/O mode, the SI pin and SO pin become I/O0 pin and I/O1 pin for either opcode, address, and data (Dual I/O mode) or both the address and data (Dual Addr/Data Mode) or just the data (Dual Data Mode).

When the part is in the quad I/O mode, the SI pin, SO pin, $\overline{WP}$ pin, and NC (I/O3) pin become I/O0 pin, I/O1 pin, I/O2 pin, and I/O3 pin for either opcode, address and data (Quad I/O Mode), or both the address and data (Quad Addr/Data Mode), or just the data (Quad Data Mode).

**Table 1. I/O Modes**

| Protocol | Command Input | Address Input | Data Input/Output |
|---|---|---|---|
| SPI | SI | SI | SI/SO |
| DPI | I/O[1:0] | I/O[1:0] | I/O[1:0] |
| QPI | I/O[3:0] | I/O[3:0] | I/O[3:0] |
| Dual Data Mode (Dual Out) | I/O[0] | I/O[0] | I/O[1:0] |
| Dual Address/ Data Mode (Dual I/O) | I/O[0] | I/O[1:0] | I/O[1:0] |
| Quad Data Mode (Quad Out) | I/O[0] | I/O[0] | I/O[3:0] |
| Quad Address/ Data Mode (Quad I/O) | I/O[0] | I/O[3:0] | I/O[3:0] |

For more details, refer to read and write timing diagrams later in the datasheet.

## SPI Modes

The device also has the capability to reconfigure. The device may be driven by a microcontroller with its SPI peripheral running in either of the following two modes:

- SPI Mode 0 (CPOL = 0, CPHA = 0)
- SPI Mode 3 (CPOL = 1, CPHA = 1)

For both these modes, the input data is latched in on the rising edge of SCK starting from the first rising edge after $\overline{CS}$ goes active. If the clock starts from a HIGH state (in mode 3), the first rising edge after the clock toggles, is considered. The output data is available on the falling edge of SCK.

The two SPI modes are shown in Figure 6 and Figure 7. The status of clock when the bus master is in standby state and not transferring data is:

- SCK remains at '0' for Mode 0
- SCK remains at '1' for Mode 3

The device detects the SPI mode from the status of SCK pin when the device is selected by bringing the CS pin LOW. If the SCK pin is LOW when the device is selected, SPI Mode 0 is assumed and if the SCK pin is HIGH, it works in SPI Mode 3.

**Figure 6. SPI Mode 0**

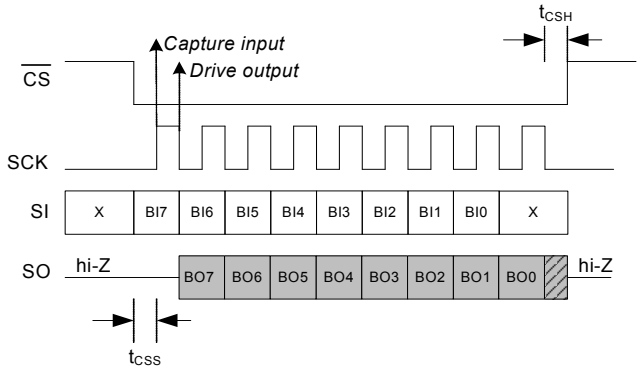
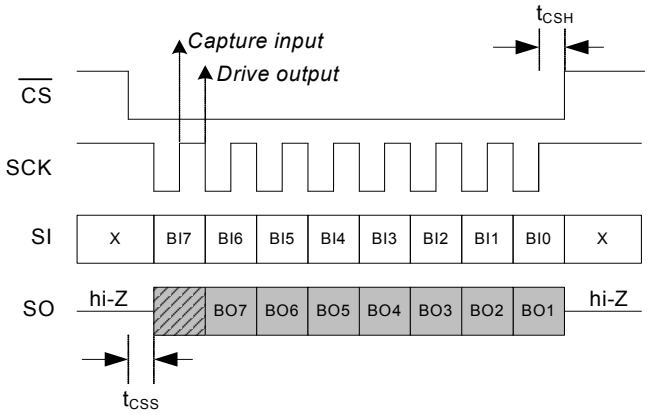



**Figure 7. SPI Mode 3**



## SPI Operating Features

### Power-Up

Power-up is defined as the condition when the power supply is turned on and $V_{CC}$ crosses $V_{SWITCH}$ voltage.

As described earlier, at power-up nvSRAM performs a Power-Up RECALL operation for $t_{FA}$ duration during which all memory accesses are disabled. The HSB pin can be probed to check the Ready/Busy status of nvSRAM after power-up.

The following is the device status after power-up:

- SPI I/O Mode
- Pull-ups activated for HSB
- SO is tristated
- Standby power mode if CS pin is high. Active power mode if CS pin is LOW.
- Status Register state:
  - Write Enable bit is reset to '0'
  - SRWD not changed from previous STORE operation
  - SNL not changed from previous STORE operation
  - Block Protection bits are not changed from previous STORE operation
- WP and NC (I/O3) functionality as defined by Quad Data Width (QUAD) CR[1]. Pull-ups activated on WP and NC (I/O3) if Quad Data width CR[1] is logic '0'.

### Power-Down

At power-down (continuous decay of $V_{CC}$), when $V_{CC}$ drops from the normal operating voltage and below the $V_{SWITCH}$ threshold voltage, the device stops responding to any instruction sent to it.

If a write cycle is in progress and the last data bit D0 has been received when the power goes down, it is allowed $t_{DELAY}$ time to complete the write. After this, all memory accesses are inhibited and a AutoStore operation is performed (AutoStore is not performed, if no write operations have been executed since the last RECALL cycle). This feature prevents inadvertent writes to nvSRAM from happening during power-down.

However, to completely avoid the possibility of inadvertent writes during power-down, ensure that the device is deselected and is in standby state, and the CS follows the voltage applied on $V_{CC}$.

### Active Power Mode and Standby State

When CS is LOW, the device is selected and is in the active power mode. The device consumes $I_{CC}$ ($I_{CC1}$ + $I_{CCQ1}$) current, as specified in on page 43. When CS is HIGH, the device is deselected and the device goes into the standby state time, if a STORE or RECALL cycle is not in progress. If a STORE/RECALL cycle is in progress, the device goes into the standby state after the STORE or RECALL cycle is completed.

## SPI Functional Description

The device has an 8-bit instruction register. Instructions and their opcodes are listed in Table 2. All instructions, addresses, and data are transferred with a HIGH to LOW $\overline{CS}$ transition. The SPI instructions along with $\overline{WP}$, NC (I/O3), and $\overline{HSB}$ pins provide access to all the functions in nvSRAM.

**Table 2. Instruction Set**

| Instruction Category | Instruction Name | Opcode | SPI | Dual Out | Quad Out | Dual I/O | Quad I/O | DPI | QPI | Max. Frequency (MHz) |
|---|---|---|---|---|---|---|---|---|---|---|
| **Control** | | | | | | | | | | |
| Write Disable | WRDI | 04h | Yes | – | – | – | – | Yes | Yes | 108 |
| Write Enable | WREN | 06h | Yes | – | – | – | – | Yes | Yes | 108 |
| Enable DPI | DPIEN | 37h | Yes | – | – | – | – | – | Yes | 108 |
| Enable QPI | QPIEN | 38h | Yes | – | – | – | – | Yes | – | 108 |
| Enable SPI | SPIEN | FFh | – | – | – | – | – | Yes | Yes | 108 |
| **Memory Read** | | | | | | | | | | |
| Read | READ | 03h | Yes | – | – | – | – | Yes | Yes | 40 |
| FastRead | FAST_READ | 0Bh | Yes | – | – | – | – | Yes | Yes | 108 |
| Dual Out (Fast) Read | DOR | 3Bh | – | Yes | – | – | – | – | – | 108 |
| Quad Out (Fast) Read | QOR | 6Bh | – | – | Yes | – | – | – | – | 108 |
| Dual I/O (Fast) Read | DIOR | BBh | – | – | – | Yes | – | – | – | 108 |
| Quad I/O (Fast) Read | QIOR | EBh | – | – | – | – | Yes | – | – | 108 |
| **Memory Write** | | | | | | | | | | |
| Write | WRITE | 02h | Yes | – | – | – | – | Yes | Yes | 108 |
| Dual Input Write | DIW | A2h | – | Yes | – | – | – | – | – | 108 |
| Quad Input Write | QIW | 32h | – | – | Yes | – | – | – | – | 108 |
| Dual I/O Write | DIOW | A1h | – | – | – | Yes | – | – | – | 108 |
| Quad I/O Write | QIOW | D2h | – | – | – | – | Yes | – | – | 108 |
| **SR Commands** | | | | | | | | | | |
| Software Reset Enable | RSTEN | 66h | Yes | – | – | – | – | Yes | Yes | 108 |
| Software Reset | RESET | 99h | Yes | – | – | – | – | Yes | Yes | 108 |
| Enter Hibernate Mode | HIBEN | BAh | Yes | – | – | – | – | Yes | Yes | 108 |
| Enter Sleep Mode | SLEEP | B9h | Yes | – | – | – | – | Yes | Yes | 108 |
| Exit Sleep Mode | EXSLP | ABh | Yes | – | – | – | – | Yes | Yes | 108 |
| **Register Commands** | | | | | | | | | | |
| Read Status Register | RDSR | 05h | Yes | – | – | – | – | Yes | Yes | 108 |
| Write Status Register | WRSR | 01h | Yes | – | – | – | – | Yes | Yes | 108 |
| Read Configuration Register | RDCR | 35h | Yes | – | – | – | – | Yes | Yes | 108 |

**Table 2. Instruction Set** (continued)

| Instruction Category | Instruction Name | Opcode | SPI | Dual Out | Quad Out | Dual I/O | Quad I/O | DPI | QPI | Max. Frequency (MHz) |
|---|---|---|---|---|---|---|---|---|---|---|
| Write Configuration Register | WRCR | 87h | Yes | – | – | – | – | Yes | – | 108 |
| Read ID Register | RDID | 9Fh | Yes | – | – | – | – | Yes | Yes | 40 |
| Fast Read ID Register | FAST_RDID | 9Eh | Yes | – | – | – | – | Yes | Yes | 108 |
| Write Serial Number Register | WRSN | C2h | Yes | – | – | – | – | Yes | Yes | 108 |
| Read Serial Number Register | RDSN | C3h | Yes | – | – | – | – | Yes | Yes | 40 |
| Fast Read Serial Number Register | FAST_RDSN | C9h | Yes | – | – | – | – | Yes | Yes | 108 |
| **NV Specific Commands** | | | | | | | | | | |
| STORE | STORE | 8Ch | Yes | – | – | – | – | Yes | Yes | 108 |
| RECALL | RECALL | 8Dh | Yes | – | – | – | – | Yes | Yes | 108 |
| Autostore Enable | ASEN | 8Eh | Yes | – | – | – | – | Yes | Yes | 108 |
| Autostore Disable | ASDI | 8Fh | Yes | – | – | – | – | Yes | Yes | 108 |
| **Mode Bits** | | | | | | | | | | |
| Mode Bit (Set, Reset) | – | Axh, not Axh | Yes | – | – | – | – | Yes | Yes | – |

Based on their functionality, the SPI instructions are divided into the following types:

- Control instructions:
  - Write-protection: WREN, WRDI instructions
  - I/O modes: DPIEN, QPIEN, SPIEN
- Memory Read instructions:
  - Memory access: READ, FAST_READ, DOR, QOR, DIOR, QIOR
- Memory Write instructions:
  - Memory access: WRITE, DIW, QIW, DIOW, QIOW
- System Resources instructions:
  - Software Reset: RSTEN, RESET
  - Power modes: HIBEN, SLEEP, EXSLP
- Register instructions:
  - Configuration Register: RDCR, WRCR
  - Status Register: RDSR, WRSR
  - Identification: RDID, FAST_RDID
  - Serial Number: RDSN, WRSN, FAST_RDSN
- nvSRAM Special instructions:
  - STORE: STORE
  - RECALL: RECALL
  - Enable/Disable: ASEN, ASDI

**Note** The instruction waveforms shown in the following sections do not incorporate the effects of pull-ups on WP (I/O2), NC (I/O3) and Repeater/Bus-Hold circuitry on SO.

**Note** Instruction Opcode C5h, 1Eh, C8h, CEh, CBh, CCh, CDh are Cypress reserved opcodes and change the configuration of the device. If any one of these opcodes are erroneously entered, a software reset (66h, 99h) is required to return the device back to correct configuration. Otherwise, the device will not behave correctly.

## Status Register

The device has one Status Register, which is listed in Table 3 along with its bit descriptions. The bit format in the Status Register shows whether the bit is read only (R) or can be written to as well (W/R). The only exception to this is the serial number lock bit (SNL). The serial number can be written using the WRSN instruction multiple times while SNL is still '0'. When set to '1', this bit prevents any modification to the serial number. This bit is factory-programmed to '0' and can only be written to once. After this bit is set to '1', it can never be cleared to '0'.

**Table 3. Status Register Format and Bit Definitions**

| Bit | Field Name | Function | Type | R/W | Default State | Description |
|---|---|---|---|---|---|---|
| 7 | SRWD | Status Register Write Disable | NV | R/W | 0 | 1 = Locks state of SR when $\overline{WP}$ is low by ignoring WRSR command<br>0 = No protection, even when $\overline{WP}$ is low |
| 6 | SNL | Serial Number Lock | OTP | R/W | 0 | Locks the Serial Number |
| 5 | TBPROT | Configures Start of Block | NV | R/W | 0 | 1 = BP starts at bottom (Low address)<br>0 = BP starts at top (High address) |
| 4 | BP2 | Block Protection | NV | R/W | 0 | Protects selected range of Block from Write, Program or Erase |
| 3 | BP1 | | NV | R/W | 0 | |
| 2 | BP0 | | NV | R/W | 0 | |
| 1 | WEL | Write Enable Latch | V | R | 0 | 1 = Device accepts Write Registers (WRSR), Write, program or erase commands<br>0 = Device ignores Write Registers (WRSR), write, program or erase commands<br>This bit is not affected by WRSR, only WREN and WRDI commands affect this bit |
| 0 | WIP | Work in Progress | V | R | 0 | 1 = Device Busy, a Write Registers (WRSR), program, erase or other operation is in progress<br>0 = Ready Device is in standby state and can accept commands |

*Status Register Write Disable (SRWD) SR[7]*

Places the device in the Hardware Protected mode when this bit is set to '1' and the $\overline{WP}$ input is driven LOW. In this mode, all the SRWD bits except WEL, become read-only bits and the Write Registers (WRSR) command is no longer accepted for execution. If $\overline{WP}$ is HIGH, the SRWD bits may be changed by the WRSR command. If SRWD is '0', $\overline{WP}$ has no effect and the SRWD bits may be changed by the WRSR command.

**Note** $\overline{WP}$ internally defaults to logic '0', if Quad bit CR[1] in Configuration register is set. If SRWD is set to logic '1', protection cannot be changed till Quad bit CR[1] is reset to logic '0'.

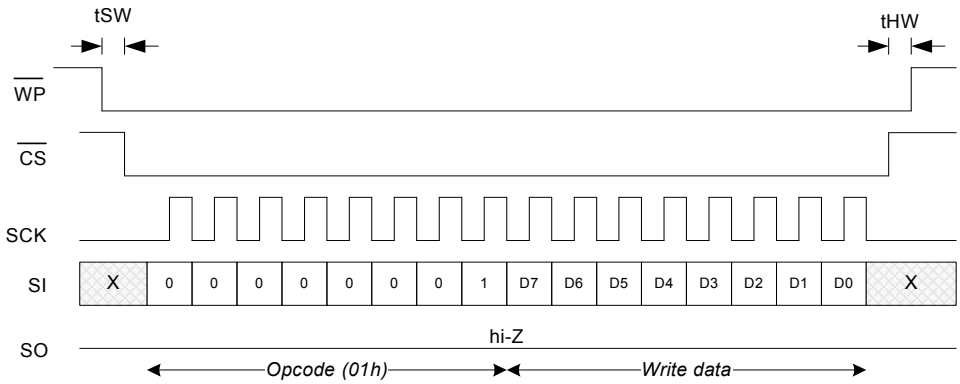
.

**Table 4. SRWD, $\overline{WP}$, WEL and Protection**

| SRWD | $\overline{WP}$ | WEL | Protected Blocks | Unprotected Blocks | Status Register (Except WEL) |
|---|---|---|---|---|---|
| X | X | 0 | Protected | Protected | Protected |
| 0 | X | 1 | Protected | Writable | Writable |
| 1 | Low | 1 | Protected | Writable | Protected |
| 1 | High | 1 | Protected | Writable | Writable |

**Note** $\overline{WP}$ is sampled with respect to $\overline{CS}$ during a write Status register instruction to determine if hardware protection is enabled. The timing waveforms are shown in Figure 8.

**Figure 8. $\overline{WP}$ Timing w.r.t $\overline{CS}$**




*Serial Number Lock (SNL) SR[6]*

When set to '1', this bit prevents any modification to the serial number. This bit is factory programmed to '0' and can only be written to once. After this bit is set to '1', it can never be cleared to '0'.

*Top or Bottom Protection (TBPROT) CR[5]*

This bit defines the operation of the Block Protection bits BP2, BP1, and BP0.The desired state of TBPROT must be selected during the initial configuration of the device during system manufacture.

*Block Protection (BP2, BP1, BP0) SR[4:2]*

These bits define the memory array area to be software-protected against write commands. The BP bits are nonvolatile. When one or more of the BP bits is set to '1', the relevant memory area is protected against write, program, and erase.

The Block Protect bits (Status Register bits BP2, BP1, BP0) in combination with the TBPROT bit can be used to protect an address range of the memory array. The size of the range is determined by the value of the BP bits and the upper or lower starting point of the range is selected by the TBPROT bit of the status register.

**Table 5. Upper Array Start of Protection (TBPROT = 0)**

| Status Register Content | | | Protection Fraction of Memory Array | Address Range |
|---|---|---|---|---|
| BP2 | BP1 | BP0 | | |
| 0 | 0 | 0 | None | None |
| 0 | 0 | 1 | Upper 64th | 0x1F800 - 0x1FFFF |
| 0 | 1 | 0 | Upper 32nd | 0x1F000 - 0x1FFFF |
| 0 | 1 | 1 | Upper 16th | 0x1E000 - 0x1FFFF |
| 1 | 0 | 0 | Upper 8th | 0x1C000 - 0x1FFFF |
| 1 | 0 | 1 | Upper 4th | 0x18000 - 0x1FFFF |
| 1 | 1 | 0 | Upper Half | 0x10000 - 0x1FFFF |
| 1 | 1 | 1 | All Sectors | 0x00000 - 0x1FFFF |

**Table 6. Lower Array Start of Protection (TBPROT = 1)**

| Status Register Content | | | Protection Fraction of Memory Array | Address Range |
|---|---|---|---|---|
| BP2 | BP1 | BP0 | | |
| 0 | 0 | 0 | None | None |
| 0 | 0 | 1 | Lower 64th | 0x00000 - 0x007FF |
| 0 | 1 | 0 | Lower 32nd | 0x00000 - 0x00FFF |
| 0 | 1 | 1 | Lower 16th | 0x00000 - 0x01FFF |
| 1 | 0 | 0 | Lower 8th | 0x00000 - 0x03FFF |
| 1 | 0 | 1 | Lower 4th | 0x00000 - 0x07FFF |
| 1 | 1 | 0 | Lower Half | 0x00000 - 0x0FFFF |
| 1 | 1 | 1 | All Sectors | 0x00000 - 0x1FFFF |

*Write Enable (WEL) SR[1]*

The WEL bit must be set to '1' to enable program, write, or erase operations as a means to provide protection against inadvertent changes to memory or register values. The Write Enable (WREN) command execution sets the Write Enable Latch to a '1' to allow any write commands to execute afterwards. The Write Disable (WRDI) command sets the Write Enable Latch to 0 to prevent all write commands from execution. The WEL bit is cleared to 0 at the end of any successful write to registers, STORE, RECALL, program or erase operation – note it is not cleared after write operations to memory macro. After a power-down/power-up sequence, hardware reset, or software reset, the Write Enable Latch is set to '0'. The WRSR command does not affect this bit.

**Note:** AutoStore, power up RECALL and Hardware STORE (HSB based) are not affected by WEL bit.

**Table 7. Instructions Requiring WEL Bit Set**

| Instruction Description | Instruction Name | Opcode |
|---|---|---|
| **Memory Write** | | |
| Write | WRITE | 02h |
| Dual Input Write | DIW | A2h |
| Quad Input Write | QIW | 32h |
| Dual I/O Write | DIOW | A1h |
| Quad I/O Write | QIOW | D2h |
| **Register Commands** | | |
| Write Status Register | WRSR | 01h |
| Write Configuration Register | WRCR | 87h |
| Write Serial Number Register | WRSN | C2h |
| **NV Specific Commands** | | |
| STORE | STORE | 8Ch |
| RECALL | RECALL | 8Dh |
| AutoStore Enable | ASEN | 8Eh |
| AutoStore Disable | ASDI | 8Fh |

*Work In Progress (WIP) SR[0]*

Indicates whether the device is performing a program, write, erase operation, or any other operation, during which a new operation command will be ignored. When the bit is set to '1', the device is busy performing a background operation. While WIP is '1', only Read Status (RDSR) command may be accepted. When the WIP bit is cleared to '0', no operation is in progress. This is a read-only bit.

All values written to SR are saved to nonvolatile memory only after a STORE operation. If AutoStore is disabled, any modifications to the Status Register must be secured by performing a software STORE operation.

Hardware Store will only commit Status register values to nonvolatile memory if there is a write to the SRAM.

*Configuration Register*

QPI nvSRAM has one Configuration register which is listed in Table 8 along with its bit descriptions. The bit format in the Configuration register shows whether the bit is read only (R) or can be written to as well (W/R). The Configuration register controls interface functions.

**Table 8. Configuration Register**

| Bit | Field Name | Function | Type | R/W | Default State | Description |
|---|---|---|---|---|---|---|
| 7 | RFU | Reserved | – | R/W | 0 | Reserved for future use |
| 6 | RFU | Reserved | – | R/W | 1 | Reserved for future use |
| 5 | RFU | Reserved | – | – | 0 | Reserved for future use |
| 4 | RFU | Reserved | – | – | 0 | Reserved for future use |
| 3 | RFU | Reserved | – | – | 0 | Reserved for future use |
| 2 | RFU | Reserved | – | – | 0 | Reserved for future use |
| 1 | QUAD | Puts device in Quad Mode | NV | R/W | 0 | 1 = Quad; 0 = Dual or Serial |
| 0 | RFU | Reserved | – | – | 0 | Reserved for future use |

*Quad Data Width (QUAD) CR[1]*

When set to ‘1’, this bit switches the data width of the device to four bits i.e. WP becomes I/O2 and NC (I/O3) becomes I/O3. The WP input is not monitored for its normal function and is internally taken to be active. The commands for Serial, Dual Output, and Dual I/O Read still function normally but, there is no need to drive WP input for those commands when switching between commands using different data path widths. The QUAD bit must be set to ‘1’ when using QUAD Out Read, QUAD I/O Read, QUAD Input Write, QUAD I/O Write, and all QUAD SPI commands. The QUAD bit is non-volatile.

**Note** To set the Quad bit, 0x42 must be written to the Configuration register. Similarly, to reset the Quad bit, 0×40 must be written to the Configuration register. Any other data combination will change the configuration of the device and make it unusable.

**Note** When Quad bit CR[1] in Configuration register is set, WP internally defaults to logic ‘0’.

**Note** The values written to Configuration Register are saved to nonvolatile memory only after a STORE operation. If AutoStore is disabled, any modifications to the Configuration Register must be secured by performing a Software STORE operation. Hardware Store will only commit Configuration register values to nonvolatile memory if there is a write to the SRAM.

## SPI Control Instructions

### Write Disable (WRDI) Instruction

The Write Disable instruction disables all writes by clearing the WEL bit to ‘0’ to protect the device against inadvertent writes. This instruction is issued after the falling edge of CS followed by opcode for WRDI instruction. The WEL bit is cleared on the rising edge of CS.

**Figure 9. WRDI Instruction in SPI Mode**

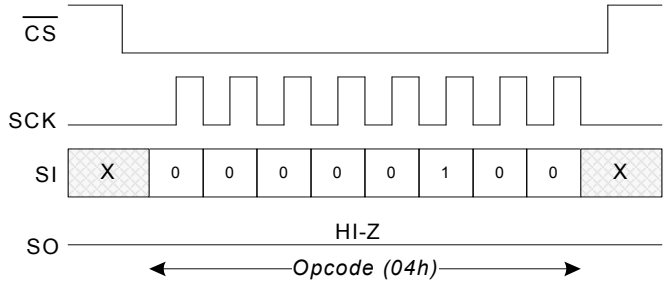



**Figure 10. WRDI Instruction in DPI Mode**

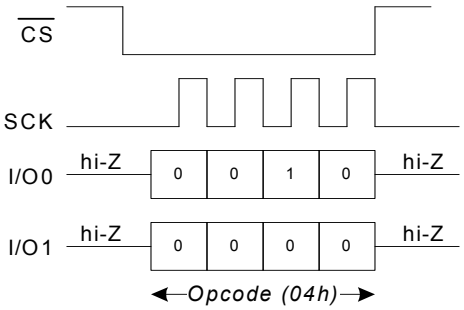
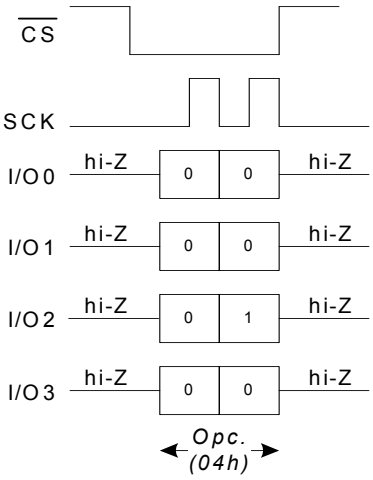



**Figure 11. WRDI Instruction in QPI Mode**




### Write Enable (WREN) Instruction

On power-up, the device is always in the Write Disable state. The write instructions and nvSRAM special instruction must therefore be preceded by a Write Enable instruction. If the device is not write enabled (WEL = ‘0’), it ignores the write instructions and returns to the standby state when CS is brought HIGH. This instruction is issued following the falling edge of CS and sets the WEL bit of the Status Register to ‘1’. The WEL bit defaults to ‘0’ on power-up.

**Note** The WEL bit is cleared to 0 at the end of any successful write to registers, STORE, RECALL, ASEN, and ASDI operation. It is not cleared after write operations to memory macro.

**Figure 12. WREN Instruction in SPI Mode**

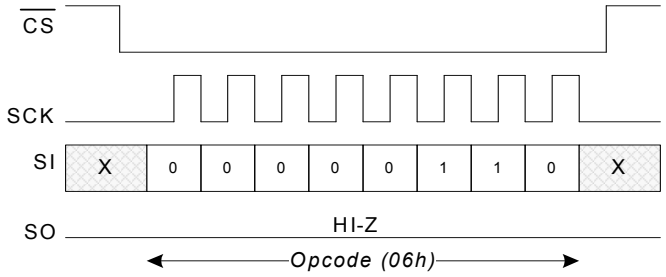



**Figure 13. WREN Instruction in DPI Mode**

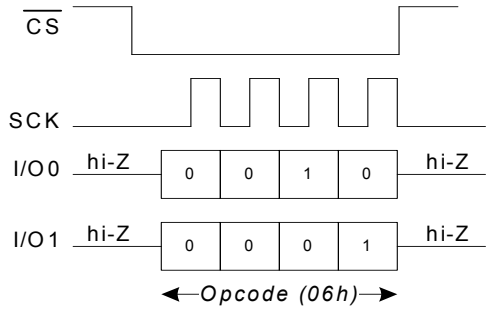
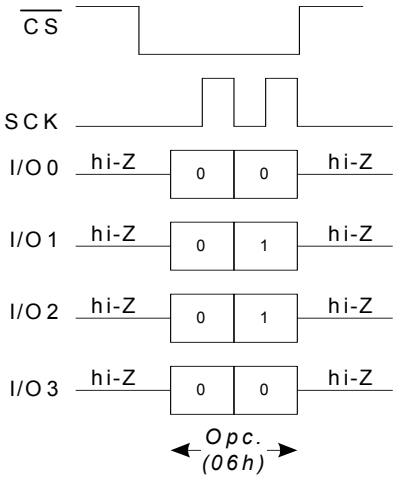



**Figure 14. WREN Instruction in QPI Mode**



## Enable DPI (DPIEN) Instruction

DPIEN enables the Dual I/O mode wherein opcode, address, mode bits, and data is sent over I/O0 and I/O1.

**Figure 15. Enable Dual I/O Instruction in SPI Mode**

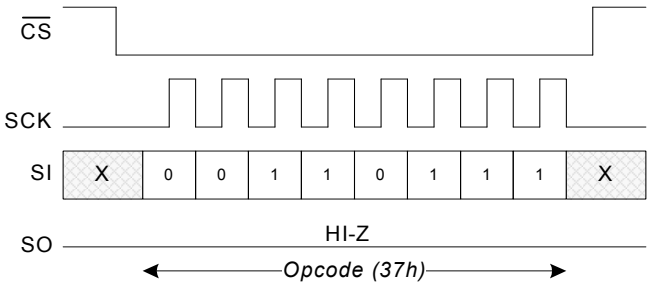
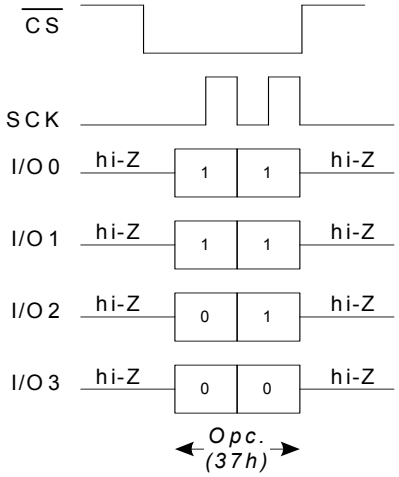



**Figure 16. Enable Dual I/O Instruction in QPI Mode**




## Enable QPI (QPIEN) Instruction

QPIEN enables QPI mode wherein opcode, address, dummy/mode bits and data is sent over I/O0, I/O1, I/O2, and I/O3. QPIEN instruction does not set the Quad bit CR[1] in Configuration register. WRCR instruction to set Quad bit CR[1] must therefore proceed QPIEN instruction.

**Note** Disabling QPI mode does not reset Quad bit CR[1].

**Figure 17. Enable Quad I/O instruction in SPI Mode**

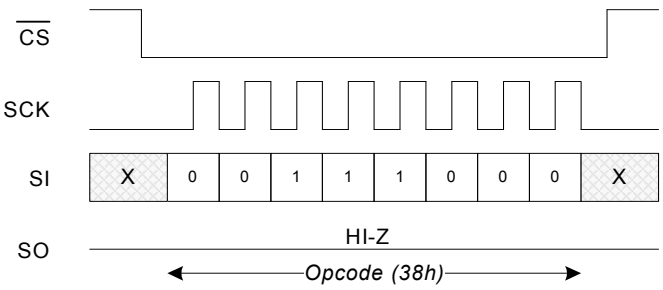
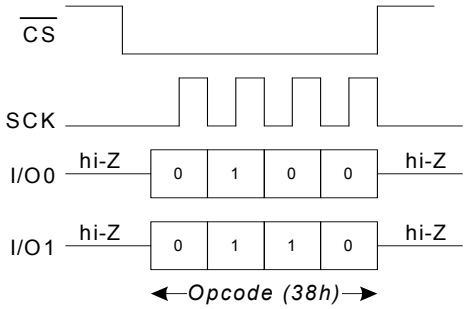



**Figure 18. Enable Quad I/O in DPI Mode**




## Enable SPI (SPIEN) Instruction

SPIEN disables Dual I/O or Quad I/O modes and returns the device in SPI mode. SPIEN instruction does not reset the Quad bit CR[1] in Configuration register.

**Figure 19. Enable SPI Instruction in DPI Mode**

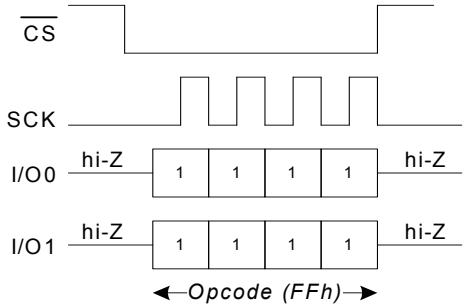
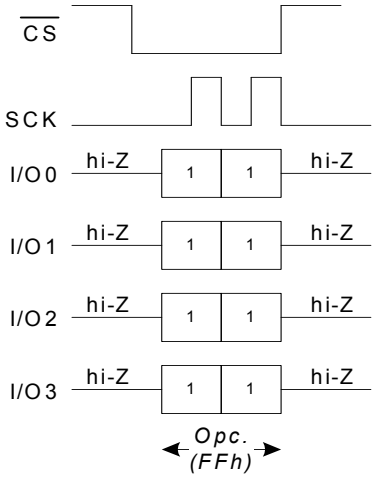



**Figure 20. Enable SPI Instruction in QPI Mode**



## SPI Memory Read Instructions

Read instructions access the memory array. These instructions cannot be used while a STORE or RECALL cycle is in progress. A STORE cycle in progress is indicated by the WIP bit of the Status Register and the HSB pin.

### Read Instructions

The device performs the read operations when read instruction opcodes are given on the SI pin and provides the read output data on the SO pin for SPI mode or the I/O1, I/O0 pins for Dual I/O Mode or the I/O3, I/O2, I/O1, and I/O0 pins for Quad I/O Mode. After the CS pin is pulled LOW to select a device, the read opcode is entered followed by three bytes of address. The device contains a 17-bit address space for 1-Mbit configuration.

The most significant address byte contains A16 in bit 0 and other bits as 'don't care'. Address bits A15 to A0 are sent in the following two address bytes. After the last address bit is transmitted, the data (D7-D0) at the specific address is shifted out on the falling edge of SCK starting with D7. The reads can be performed in burst mode if CS is held LOW.

The device automatically increments to the next higher address after each byte of data is output. When the last data memory address (0x1FFFF) is reached, the address rolls over to 0x00000 and the device continues the read instruction. The read operation is terminated by driving CS HIGH at any time during data output.

**Note** The Read instruction operates up to maximum of 40-MHz frequency. In Dual and Quad I/O modes, dummy cycle is required after the address bytes. This allows the device to pre-fetch the first byte and start the pipeline flowing.

*READ Instruction*

READ instruction can be used in SPI, Dual I/O (DPI) or Qua I/O (QPI) Modes. In SPI Mode, opcode and address bytes are transmitted through SI pin, one bit per clock cycle. At the falling edge of SCK of the last address cycle, the data (D7-D0) at the specific address is shifted out on SO pin one bit per clock cycle starting with D7.

In DPI Mode, opcode and address bytes are transmitted through I/O1 and I/O0 pins, two bits per clock cycle. At the falling edge of SCK after the last address cycle, the data (D7-D0) at the specific address is shifted out two bits per clock cycle starting with D7 on I/O1 and D6 on I/O0. In QPI Mode, opcode and address bytes are transmitted through I/O3, I/O2, I/O1, and I/O0 pins, four bits per clock cycle. At the falling edge of SCK of the last address cycle, data (D7-D0) at the specific address is shifted out four bits per clock cycle starting with D7 on I/O3, D6 on I/O2, D5 on I/O1, and D4 on I/O0.

**Figure 21. READ Instruction in SPI Mode**

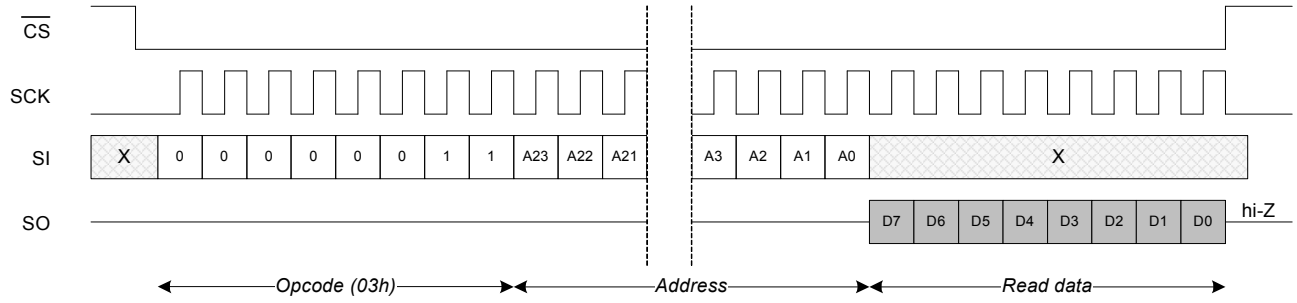



**Figure 22. Burst Mode READ Instruction in SPI Mode**

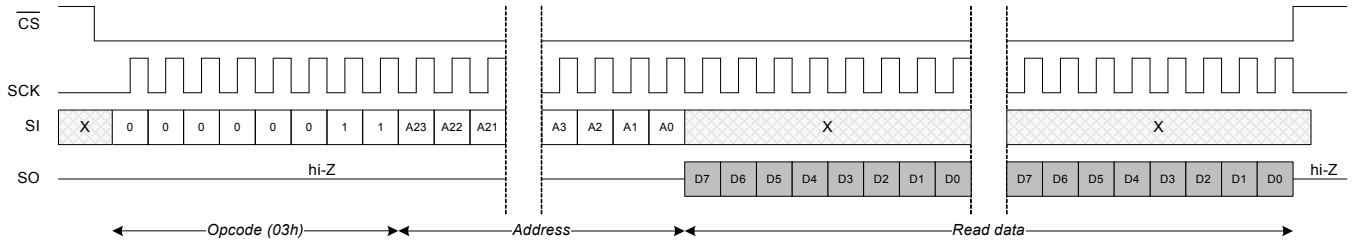


**Figure 23. READ Instruction in DPI Mode**

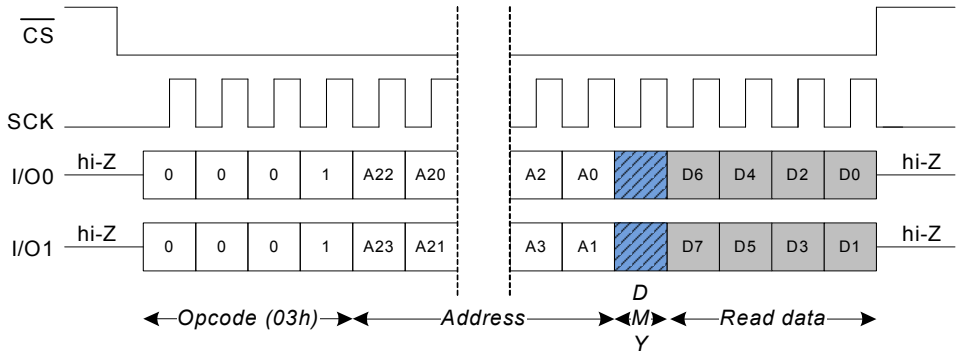
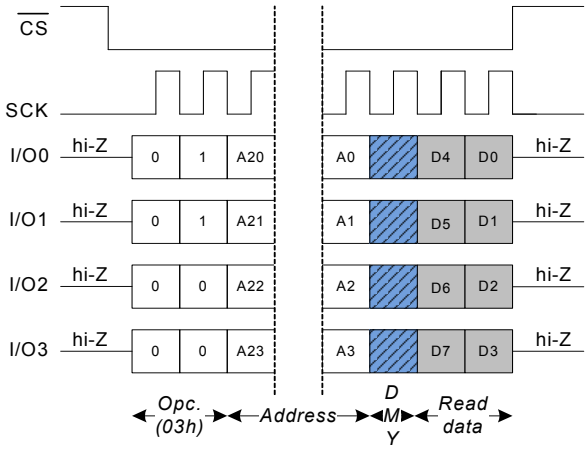



**Figure 24. READ Instruction in QPI Mode**




**Note:** Quad bit CR[1] must be logic '1' before executing the READ instruction in QPI mode.

## Fast Read Instructions

The fast read instructions allow you to read memory at SPI frequency up to 108 MHz (max). The instruction is similar to the normal read instruction with the addition of a wait state in all I/O configurations; a mode byte must be sent after the address and before the first data is sent out. This allows the device to pre-fetch the first byte and start the pipeline flowing. The host system must first select the device by driving CS LOW, followed by the 3 address bytes and then a mode byte. At the next falling edge of the SCK, data from the specific address is shifted out on the SO pin for SPI Mode or the I/O1, I/O0 pins for Dual I/O Mode or the I/O3, I/O2, I/O1, and I/O0 pins for Quad I/O Mode. The first byte specified can be at any location. The device automatically increments to the next higher address after each byte of data is output. The entire memory array can therefore be read with a single fast read instruction. When the highest address in the memory array is reached, the address counter rolls over to starting address 0x00000 and allows the read sequence to continue indefinitely. The fast read instructions are terminated by driving CS HIGH at any time during data output.

**Note** These instructions operate up to maximum of 108-MHz SPI frequency.

### *FAST_READ Instruction*

FAST_READ instruction can be used in SPI, Dual I/O (DPI) or Quad I/O (QPI) Modes. In SPI Mode, opcode, address and mode byte are transmitted through SI pin, one bit per clock cycle. At the falling edge of SCK of the last mode byte cycle, the data (D7-D0) from the specific address is shifted out on SO pin, one bit per clock cycle starting with D7. In DPI Mode, opcode, address and mode byte are transmitted through I/O1 and I/O pins, two bits per clock cycle. At the falling edge of the last mode cycle, the data (D7-D0) from the specific address is shifted out two bits per clock cycle starting with D7 on I/O1 and D6 on I/O0. In QPIO Mode, opcode, and address bytes are transmitted through I/O3, I/O2, I/O1, and I/O0 pins, four bits per clock cycle. At the falling edge of SCK of the last mode cycle, the data (D7-D0) from the specific address is shifted out, four bits per clock cycle starting with D7 on I/O3, D6 on I/O2, D5 on I/O1, and D4 on I/O0.

**Figure 25. FAST_READ Instruction in SPI Mode**

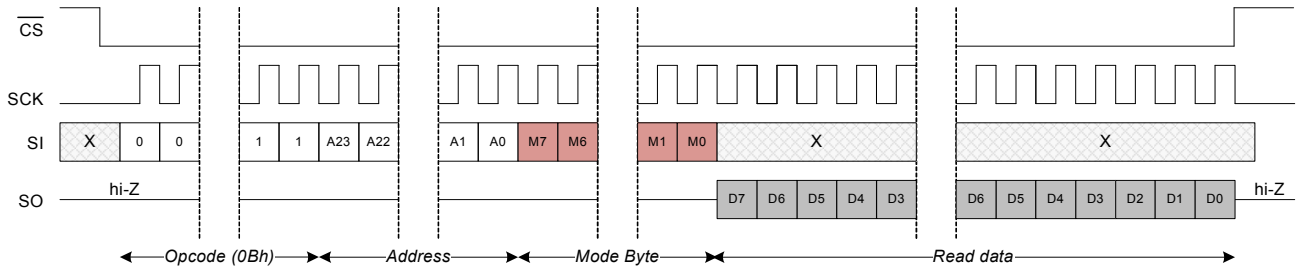


**Figure 26. FAST_READ Instruction in DPI Mode**

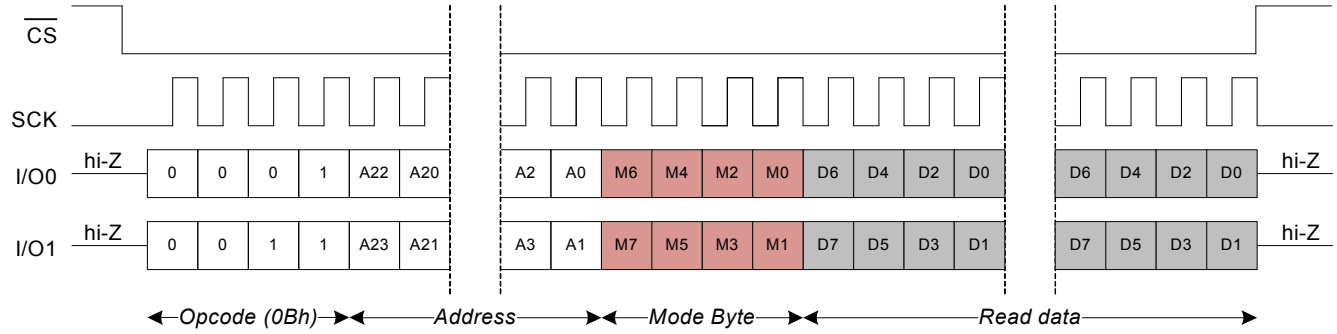
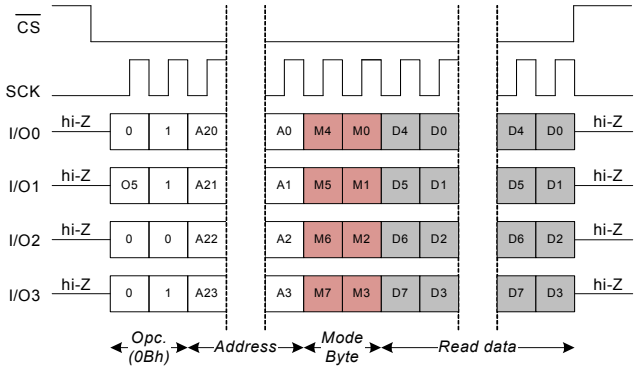



**Figure 27. FAST_READ Instruction in QPI Mode**




*DOR Instruction*

DOR instruction is used in Dual Data Mode, which is part of Extended SPI Read commands. In Dual Data Mode, opcode, address and mode byte are transmitted through SI pin, one bit per clock cycle. At the falling edge of SCK of the last mode cycle, the pins are reconfigured as SO becoming I/O1, and SI becoming I/O0. The data (D7-D0) from the specified address is shifted out on I/O1, and I/O0 pins two bits per clock cycle starting with D7 on I/O1, and D6 on I/O0.

*QOR Instruction*

QOR instruction is used in Quad Data Mode, which is part of Extended SPI Read commands. In Quad Data Mode, opcode, address and mode byte are transmitted through SI pin, one bit per clock cycle. At the falling edge of SCK of the last mode cycle, the pins are reconfigured as NC becoming I/O3, WP becoming I/O2, SO becoming I/O1, and SI becoming I/O0. The data (D7-D0) from the specified address is shifted out on I/O3, I/O2, I/O1, and I/O0 pins four bits per clock cycle starting with D7 on I/O3 and D6 on I/O2, D5 on I/O1, and D4 on I/O0.

**Note** Quad bit CR[1] must be logic ‘1’ before executing the QOR instruction.

**Figure 28. DOR Instruction**

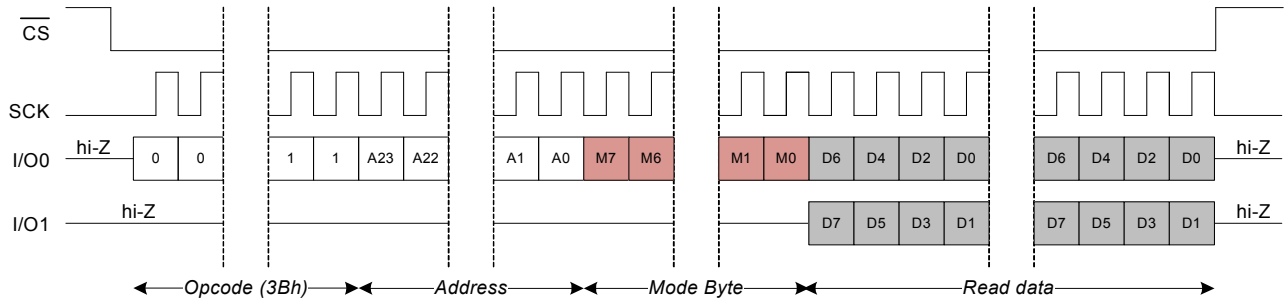
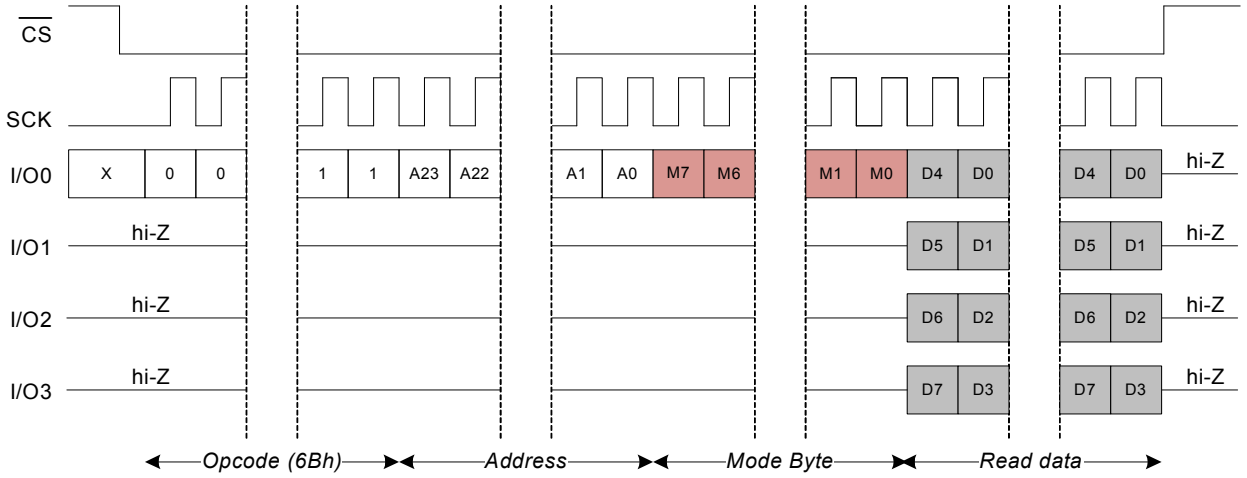


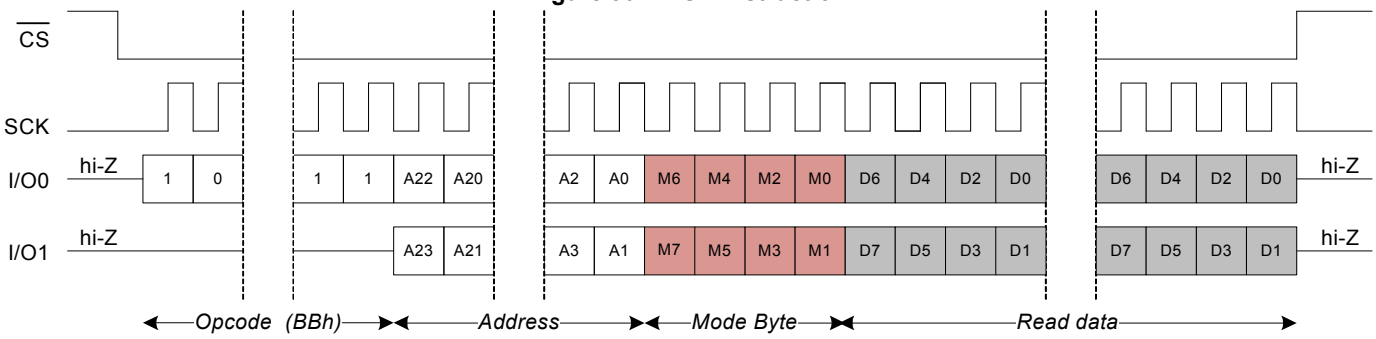
**Figure 29. QOR Instruction**




*DIOR Instruction*

DIOR instruction is used in Dual Addr/Data Mode, which is part of Extended SPI Read commands. In Dual Addr/Data Mode, opcode is transmitted through SI pin, one bit per clock cycle. After the last bit of the opcode, the pins are reconfigured as SO becoming I/O1, and SI becoming I/O0. The address is then transmitted into the part through I/O1 and I/O0 pins, 2 bits per clock cycle, starting with A23 on I/O1 and A22 on I/O0, until three bytes worth of address is input. The data (D7-D0) at the specific address is shifted out on I/O1, and I/O0 pins two bits per clock cycle starting with D7 on I/O1, and D6 on I/O0.

**Figure 30. DIOR Instruction**

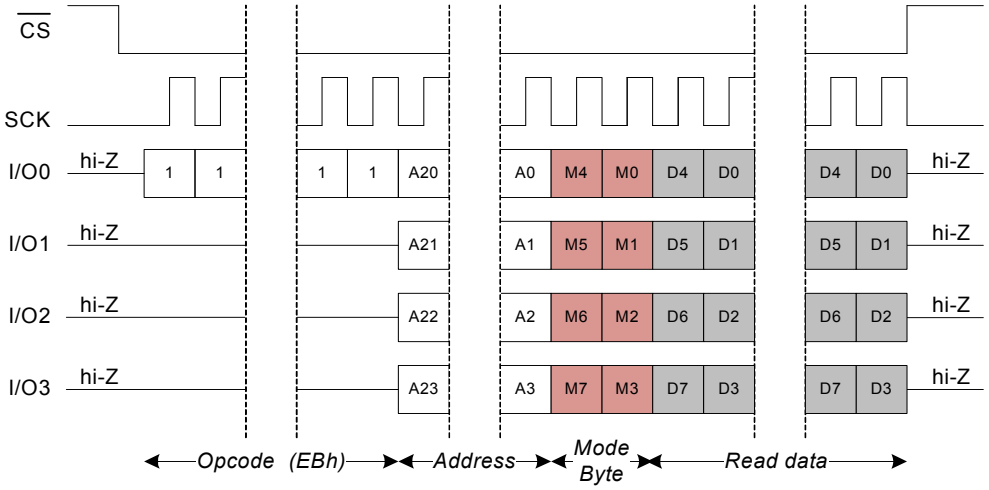



*QIOR Instruction*

QIOR instruction is used in Quad Addr/Data Mode, which is part of Extended SPI Read commands. In Quad Addr/Data Mode, opcode is transmitted through SI pin, one bit per clock cycle. After the last bit of the opcode, the pins are reconfigured as NC becoming I/O3, $\overline{WP}$ becoming I/O2, SO becoming I/O1, and SI becoming I/O0. The address is then transmitted into the part through I/O3, I/O2, I/O1 and I/O0 pins, 4 bits per clock cycle, starting with A23 on I/O3, A22 in I/O2, A21 on I/O1 and A20 on I/O0, until three bytes worth of address is input. The data (D7-D0) at the specific address is shifted out on I/O3, I/O2, I/O1, and I/O0 pins four bits per clock cycle starting with D7 on I/O3 and D6 on I/O2, D5 on I/O1, and D4 on I/O0.

**Note** Quad bit CR[1] must be logic '1' before executing the QIOR instruction.

**Figure 31. QIOR Instruction**




## Write Instructions

The device performs the write operations when write instruction opcodes along with write data are given on the SI pin for SPI Mode or the I/O1, I/O0 pins for Dual I/O Mode or the I/O3, I/O2, I/O1, and I/O0 pins for Quad I/O Mode. To perform a write operation, if the device is write disabled, then the device must be first write enabled through the WREN instruction. When the writes are enabled (WEL = '1'), WRITE instruction is issued after the falling edge of CS. nvSRAM enables writes to be performed in bursts which can be used to write consecutive addresses without issuing a new Write instruction. If only one byte is to be written, the CS pin must be driven HIGH after the D0 (LSB of data) is transmitted. However, if more bytes are to be written, CS pin must be held LOW and the address is incremented automatically. The data bytes on the input pin(s) are written in successive addresses. When the last data memory address (0x1FFFF) is reached, the address rolls over to 0x00000 and the device continues to write.

**Note** The WEL bit in the Status Register does not reset to '0' on completion of a Write sequence to the memory array.

**Note** When a burst write reaches a protected block address, it continues incrementing the address into the protected space but does not write any data to the protected memory. If the address rolls over and takes the burst write to unprotected space, it resumes writes. The same operation is true if a burst write is initiated within a write-protected block.

**Note** These instructions operate up to a maximum of 108-MHz frequency.

After the CS pin is pulled LOW to select a device, the write opcode is followed by three bytes of address. The device has a 17-bit address space for 1-Mbit configuration. The most significant address byte contains A16 in bit 0 and the remaining bits as 'don't care'. Address bits A15 to A0 are sent in the following two address bytes. Immediately after the last address bit is transmitted, the data (D7-D0) is transmitted through the input line(s). This command can be used in SPI, DPI or QPI Modes.

### *WRITE Instruction*

WRITE instruction can be used in SPI, DPI, or QPI Modes. In SPI Mode, opcode, address bytes and data bytes are transmitted through SI pin, one bit per clock cycle starting with D7. In DPI Mode, opcode, address bytes and data bytes are transmitted through I/O1 and I/O pins, two bits per clock cycle starting with D7 on I/O1 and D6 on I/O0. In QPI Mode, opcode, address bytes, and data bytes are transmitted through I/O3, I/O2, I/O1, and I/O0 pins, four bits per clock cycle starting with D7 on I/O3, D6 on I/O2, D5 on I/O1, and D4 on I/O0.

**Figure 32. WRITE Instruction in SPI Mode**

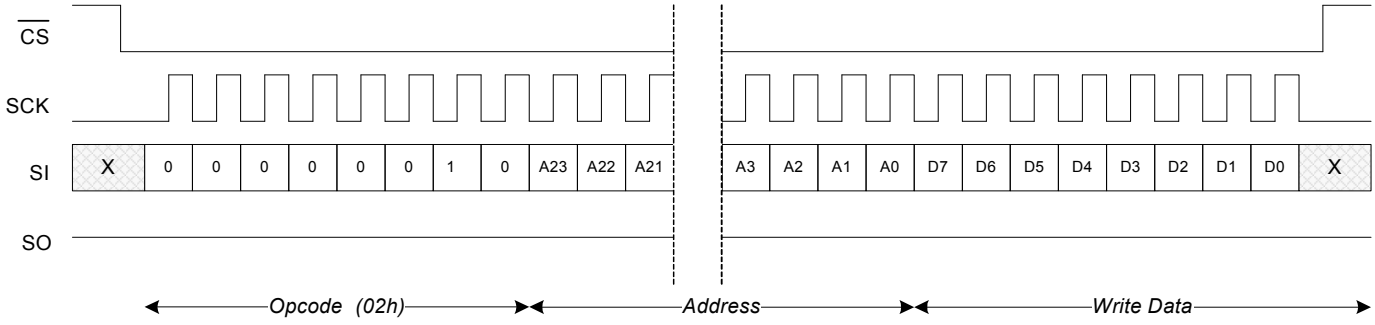


**Figure 33. Burst WRITE Instruction in SPI Mode**

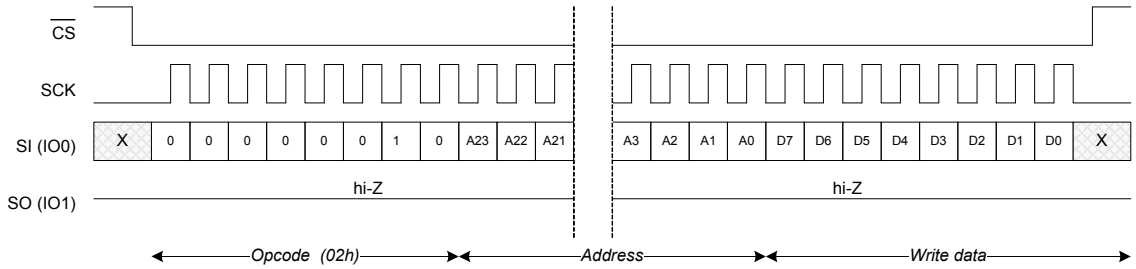



**Figure 34. WRITE Instruction in DPI Mode**

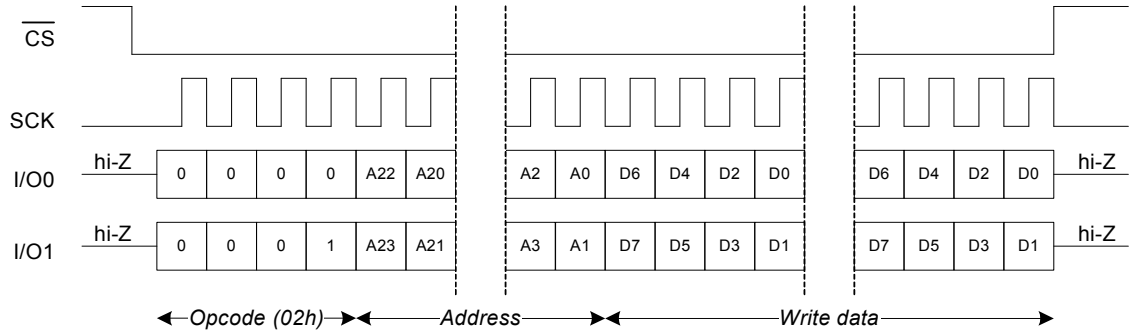
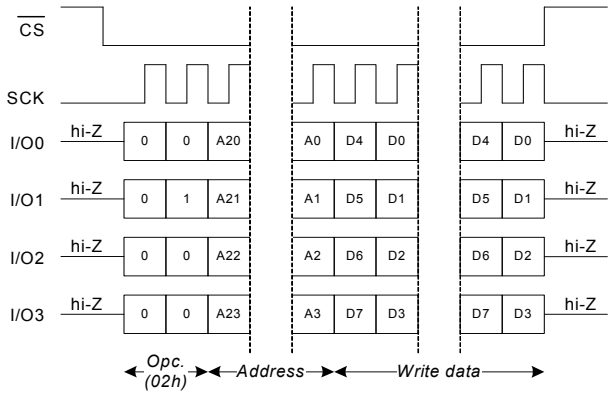



**Figure 35. WRITE Instruction in QPI Mode**

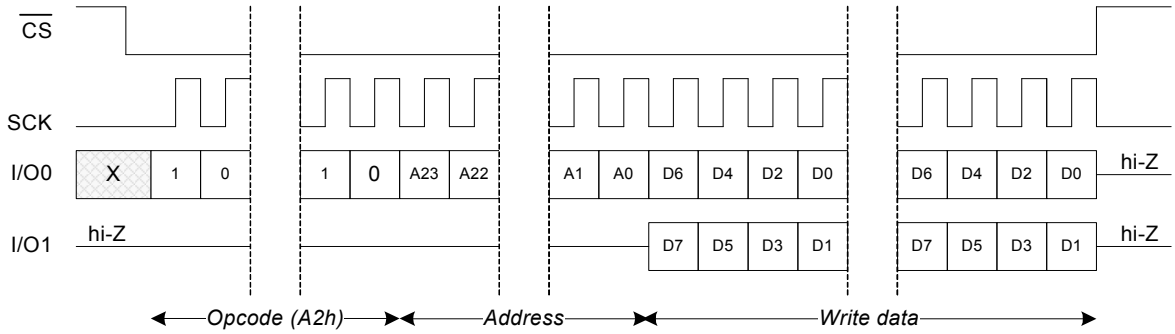



**Note** Quad bit CR[1] must be logic '1' before executing the WRITE instruction in QPI mode.

*DIW Instruction*

DIW Instruction can be used in Dual Data Mode, which is part of Extended SPI Write commands. In Dual Data Mode, opcode, and address bytes are transmitted through SI pin, one bit per clock cycle. Immediately after the last address bit is transmitted, the pins are reconfigured as SO becoming I/O1, and SI becoming I/O0, and the data (D7-D0) is transmitted into the I/O1, and I/O0 pins, 2 bits per clock cycle, starting with D7 on I/O1 and D6 on I/O0.

**Figure 36. DIW Instruction**

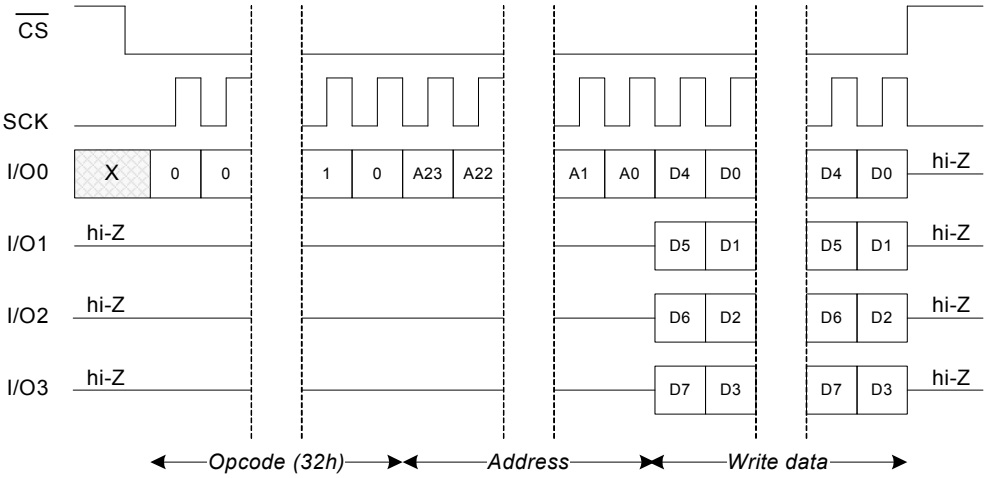


*QIW Instructions*

QIW Instruction can be used in Quad Data Mode, which is part of Extended SPI Write commands. In Quad Data Mode, opcode, and address bytes are transmitted through SI pin, one bit per clock cycle. Immediately after the last address bit is transmitted, the pins are reconfigured as NC becoming I/O3, $\overline{WP}$ becoming I/O2, SO becoming I/O1, and SI becoming I/O0, and the data (D7-D0) is transmitted into the I/O3 I/O2, I/O1, and I/O0 pins, 4 bits per clock cycle, starting with D7 on I/O3 and D6 on I/O2, D5 on I/O1, and D4 on I/O0.

**Note** Quad bit CR[1] must be logic '1' before executing the QIW instruction.

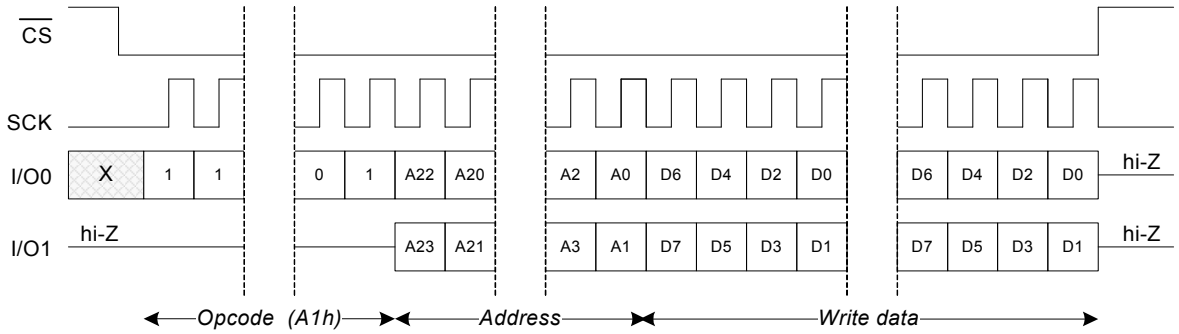
**Figure 37. QIW Instruction**




*DIOW Instruction*

DIOW Instruction can be used in Dual Addr/Data Mode, which is part of Extended SPI Write commands. In Dual Addr/Data Mode, opcode is transmitted through SI pin, one bit per clock cycle. Immediately after the last opcode bit is transmitted, the pins are reconfigured as SO becoming I/O1, and SI becoming I/O0, and the address is transmitted into the part through I/O1 and I/O0 pins, 2 bits per clock cycle, starting with A23 on I/O1, A22 on I/O0, until three bytes worth of address is input. After the last address bits are transmitted, the data (D7-D0) is transmitted into the part through I/O1 and I/O0 two bits per clock cycle starting with D7 on I/O1 and D6 on I/O0.

**Figure 38. DIOW Instruction**

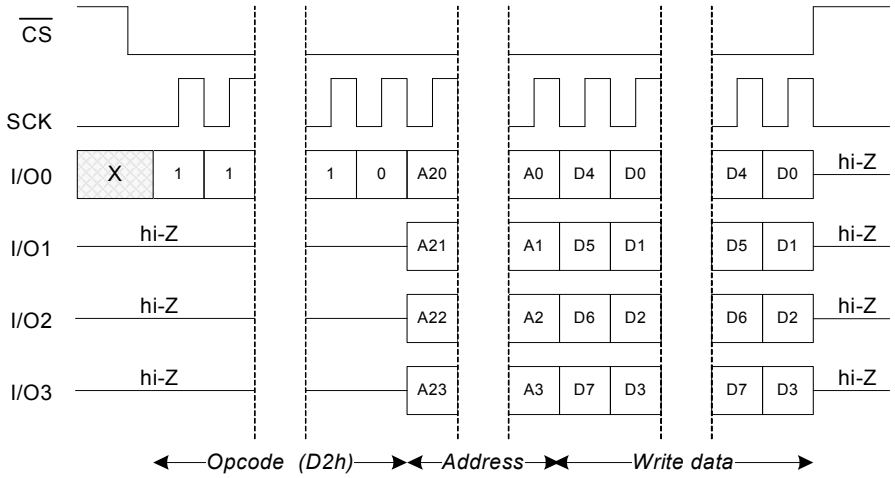



*QIOW Instruction*

QIOW instruction can be used in Quad Addr/Data Mode, which is part of Extended SPI Write commands. In Quad Addr/Data Mode, opcode is transmitted through SI pin, one bit per clock cycle. Immediately after the last opcode bit is transmitted, the pins are reconfigured as NC becoming I/O3, $\overline{WP}$ becoming I/O2, SO becoming I/O1, and SI becoming I/O0, and the address is transmitted into the part through I/O3, I/O2, I/O1 and I/O0 pins, 4 bits per clock cycle, starting with A23 on I/O3, A22 in I/O2, A21 on I/O1, and A20 on I/O0, until three bytes worth of address is input. After the last address bits are transmitted, the data (D7-D0) is transmitted into the part through I/O3, I/O2, I/O1 and I/O0 four bits per clock cycle starting with D7 on I/O3, D6 on I/O2, D5 on I/O1, and D4 on I/O0.

**Note** Quad bit CR[1] must be logic '1' before executing the QIOW instruction.

**Figure 39. QIOW Instruction**




*Execute-In-Place (XIP)*

Execute-in-place (XIP) mode allows the memory to perform a series of reads beginning at different addresses without having to load the command code for every read. This improves random access time and eliminates the need to shadow code onto RAM for fast execution. The read commands supported in XIP mode are FAST_READ (in SPI, DPI, and QPI mode), DOR, DIOR, QOR and QIOR.

XIP mode for these commands is Set or Reset by entering the Mode bits. The upper nibble (bits 7-4) of the Mode bits control the length of the next afore mentioned read command through the inclusion or exclusion of the first byte instruction code. The lower nibble (bits 3-0) of the Mode bits are “don’t care” (“x”) and may be high impedance – it is often used by the microcontrollers to turn the bus around for read data. If the Mode bits equal Axh, then the device is set to be/remain in read Mode and the next address can be entered without the opcode, as shown in figure below; thus, eliminating some cycles for the opcode sequence. If the Mode bits equal Axh, then the XIP mode is reset and the device expects an opcode after the end of the current transaction.

XIP can be entered or exited during these commands at any time and in any sequence. If it is necessary to perform another operation, not supported by XIP, such as a write, then XIP must be exited before the new command code is entered for the desired operation.

**Figure 40. XIP for SPI Mode and FAST_READ Instruction (0Bh)**

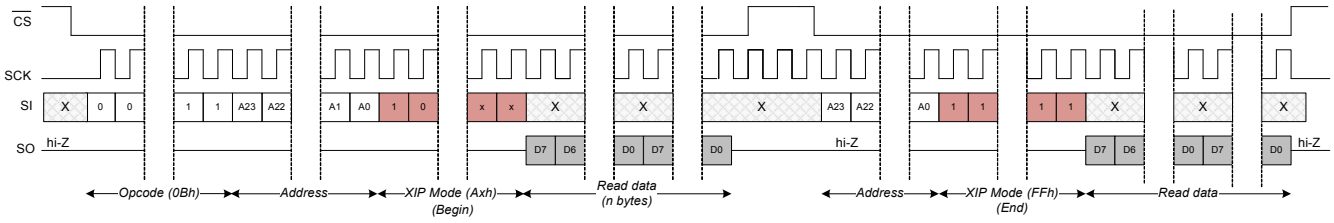
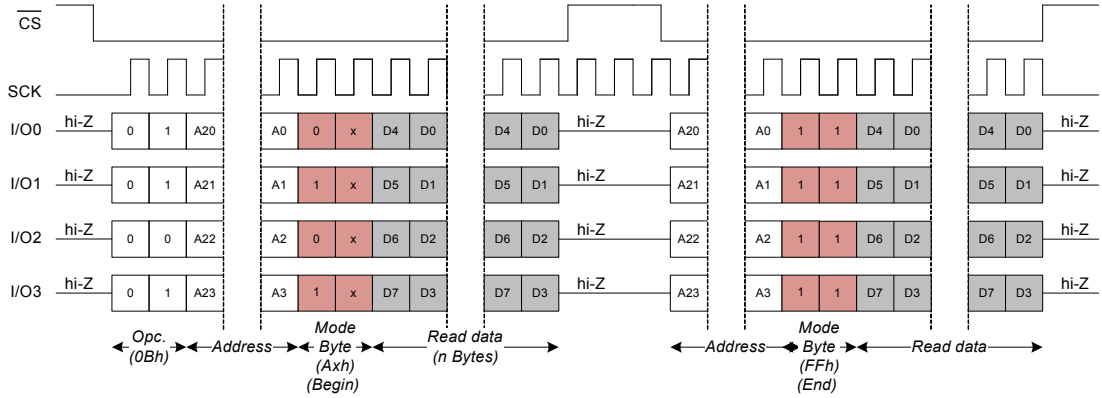



**Figure 41. XIP for QPI Mode and FAST_READ Instruction (0Bh)**



## System Resources Instructions

### Software Reset (RESET) Instruction

RESET instruction resets the whole device and makes it ready to receive commands. The I/O mode is configured to SPI. All nonvolatile registers or nonvolatile register bits maintain their values. All volatile registers or volatile register bits default to logic '0'. It takes $t_{RESET}$ time to complete. No STORE/RECALL operations are performed. To initiate the software reset process, the reset enable (RSTEN) instruction is required. This ensures protection against any inadvertent resets. Thus software reset is a sequence of two commands.

**Note** Any command other than RESET following the RSTEN command, will clear the reset enable condition and prevent a later RESET command from being recognized.

**Note** If WIP (SR[0]) bit is high and the RSTEN/RESET instruction is entered, the device ignores the RSTEN/RESET instruction.

**Note** The functionalities of $\overline{WP}$ and NC (I/O3) are controlled by the Quad bit CR[1] in Configuration register. If Quad bit is set to logic '1', $\overline{WP}$ and NC (I/O3) are configured as I/O2 and I/O3 respectively. Otherwise, $\overline{WP}$ and NC (I/O3) functionality is configured.

Table 9 summarizes the device's state after software reset.

**Table 9. Software Reset State**

| State 1 | State 2 | State 3 | I/O Mode & Register Bits |
|---|---|---|---|
| STANDBY | Software RESET | STANDBY | I/O Mode: SPI<br>SRWD SR[7]: Same as State 1<br>SNL SR[6]: Same as State 1<br>TBPROT SR[5]: Same as State 1<br>BP2 SR[4]: Same as State 1<br>BP1 SR[3]: Same as State 1<br>BP0 SR[2]: Same as State 1<br>WEL SR[1]: 0<br>WIP SR[0]: 0<br>QUAD CR[1]: Same as State 1 |

**Figure 42. RESET Instruction in SPI Mode**

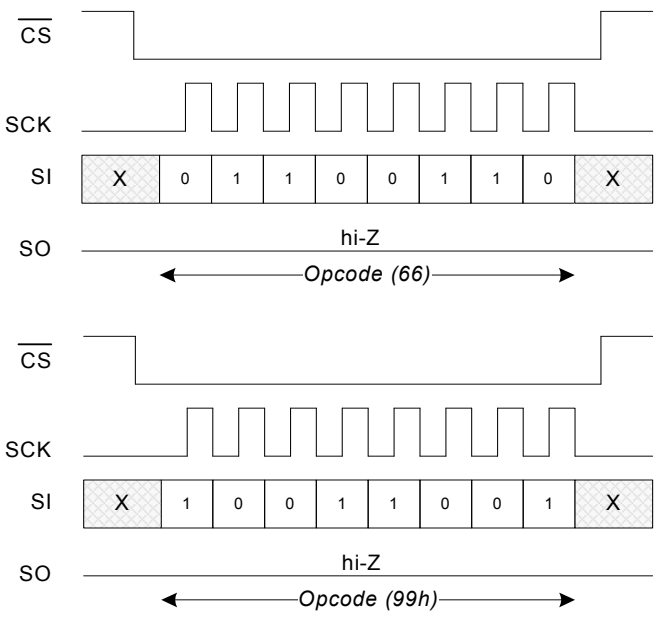



**Figure 43. RESET Instruction in DPI Mode**

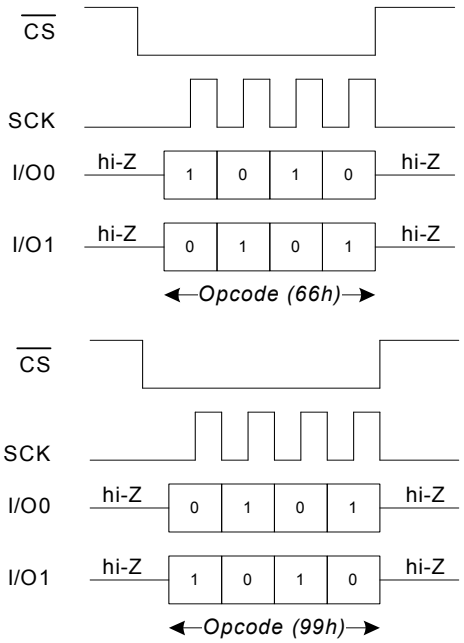
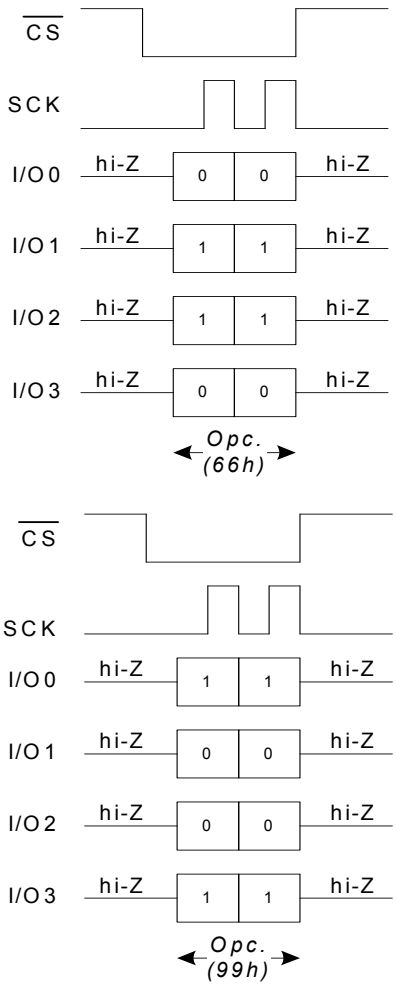


**Figure 44. RESET Instruction in QPI Mode**



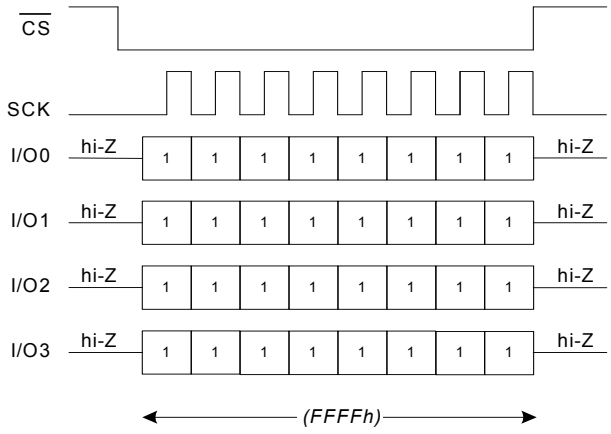

**Note** Quad bit CR[1] must be logic ‘1’ before executing RSTEN/RESET instructions in QPI mode.

## Default Recovery Instruction

The device provides a default recovery mode where the device is brought back to SPI mode. A logic high on all I/Os (I/O3, I/O2, I/O1, I/O0) with eight SCLKs brings the device into a known mode (SPI) so that the host can communicate to the device if the starting mode is unknown.

**Note** The functionalities of $\overline{WP}$ and NC (I/O3) are controlled by the Quad bit CR[1] in configuration register. If Quad bit is set to logic ‘1’, $\overline{WP}$ and NC (I/O3) are configured as I/O2 and I/O3 respectively. Otherwise, $\overline{WP}$ and NC (I/O3) functionality is configured.

**Figure 45. Default Recovery Instruction**



### Hibernate (HIBEN) Instruction

HIBEN instruction puts the nvSRAM in hibernate mode. When the HIBEN instruction is issued, the nvSRAM takes tSS time to process the HIBEN request. After the HIBEN command is successfully registered and processed, the nvSRAM toggles HSB LOW, performs a STORE operation to secure the data to nonvolatile cells and then enters hibernate mode. The device starts consuming $I_{ZZ}$ current after $t_{HIBEN}$ time when the HIBEN instruction is registered. The device is not accessible for normal operations after the HIBEN instruction is issued. In hibernate mode, the SCK and SI pins are ignored and SO will be HI-Z but the device continues to monitor the $\overline{CS}$ pin.

To wake the nvSRAM from the hibernate mode, the device must be selected by toggling the $\overline{CS}$ pin from HIGH to LOW. The device wakes up and is accessible for normal operations after tWAKE duration after a falling edge of $\overline{CS}$ pin is detected. The part will wake up in the same mode as before the HIBEN instruction.

**Note** Whenever nvSRAM enters hibernate mode, it initiates a nonvolatile STORE cycle, which results in an endurance cycle per hibernate command execution. A STORE cycle starts only if a write to the SRAM has been performed since the last STORE or RECALL cycle.

Table 10 summarizes the wake from Hibernate device states.

**Table 10. Wake (Exit Hibernate) States**

| State 1 | State 2 | State 3 | I/O Mode and Register Bits |
|---|---|---|---|
| STANDY | Hibernate | STANDBY | I/O Mode: Same mode as State 1 (SPI/DPI/QPI)<br>SRWD SR[7]: Same as State 1<br>SNL SR[6]: Same as State 1<br>TBPROT SR[5]: Same as State 1<br>BP2 SR[4]: Same as State 1<br>BP1 SR[3]: Same as State 1<br>BP0 SR[2]: Same as State 1<br>WEL SR[1]: 0<br>WIP SR[0]: 0<br>QUAD CR[1]: Same as State 1 |

**Figure 46. HIBEN Instruction in SPI Mode**

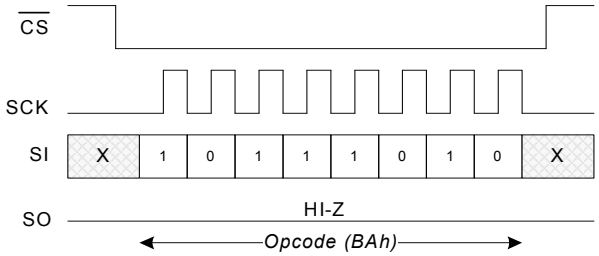



**Figure 47. HIBEN Instruction in DPI Mode**

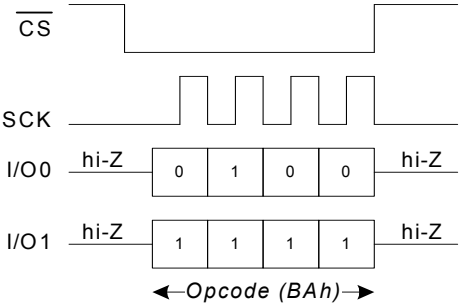
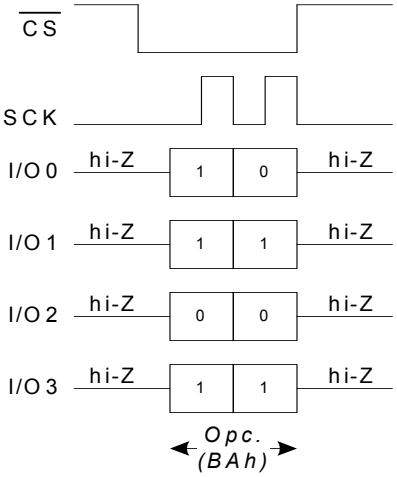


**Figure 48. HIBEN Instruction in QPI Mode**




**Note** Quad bit CR[1] must be logic '1' before executing the HIBEN instruction in QPI mode.

## Sleep (SLEEP) Instruction

SLEEP instruction puts the nvSRAM in sleep mode. When the SLEEP instruction is issued, the nvSRAM takes $t_{SLEEP}$ time to process the SLEEP request and starts consuming $I_{SLEEP}$ current. The device is not accessible for normal operations after the SLEEP instruction is issued. In sleep mode, all pins are active.

To wake the nvSRAM from sleep mode, EXSLP instruction must be entered. The nvSRAM is accessible for normal operations after $t_{EXSLP}$ duration. The part will wake in the same mode as before the SLEEP instruction. Any instructions entered other than EXSLP and RDSR instructions while the device is in sleep mode will be ignored.

Table 11 summarizes the exit from sleep device states.

**Table 11. Exit SLEEP (EXSLP) States**

| State 1 | State 2 | State 3 | I/O Mode & Register Bits |
|---|---|---|---|
| STANDY | SLEEP | STANDBY | I/O Mode: Same mode as State 1 (SPI/DPI/QPI)<br>SRWD SR[7]: Same as State 1<br>SNL SR[6]: Same as State 1<br>TBPROT SR[5]: Same as State 1<br>BP2 SR[4]: Same as State 1<br>BP1 SR[3]: Same as State 1<br>BP0 SR[2]: Same as State 1<br>WEL SR[1]: Same as State 1<br>WIP SR[0]: 0<br>QUAD CR[1]: Same as State 1 |

**Figure 49. SLEEP Instruction in SPI Mode**

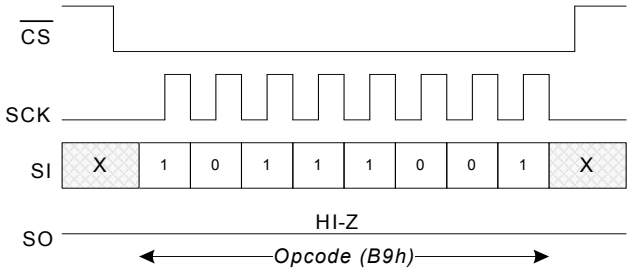



**Figure 50. SLEEP Instruction in DPI Mode**

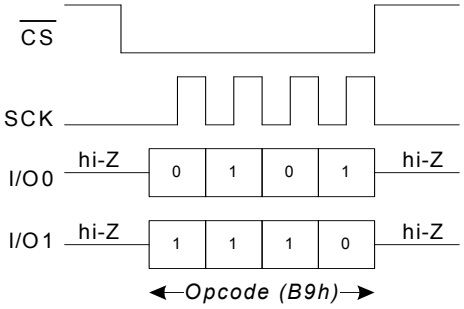


**Figure 51. SLEEP Instruction in QPI Mode**

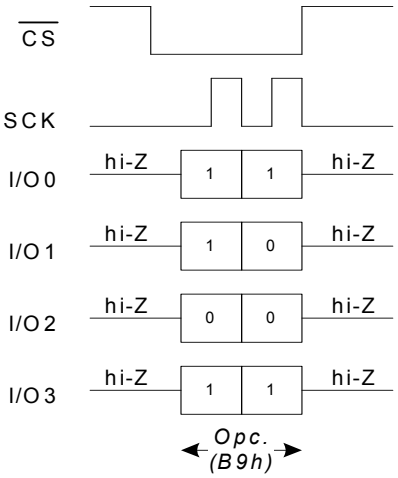



**Figure 52. EXSLP Instruction in SPI Mode**

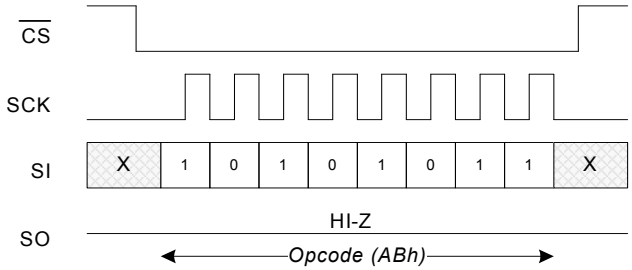



**Figure 53. EXSLP Instruction in DPI Mode**

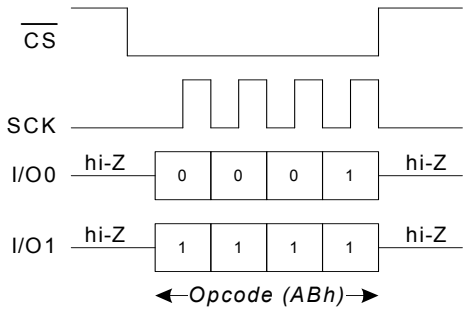
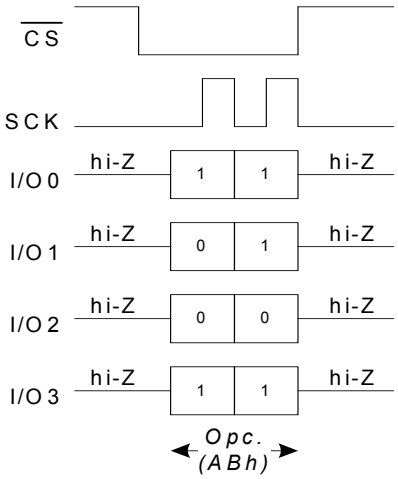



**Figure 54. EXSLP Instruction in QPI Mode**



## Register Instructions

### Read Status Register (RDSR) Instruction

The RDSR instruction provides access to Status Register at SPI frequencies up to 108 MHz. This instruction is used to probe the status of the device.

**Note** After the last bit of Status Register is read, the device loops back to the first bit of the Status Register.

**Figure 55. RDSR Instruction in SPI Mode**

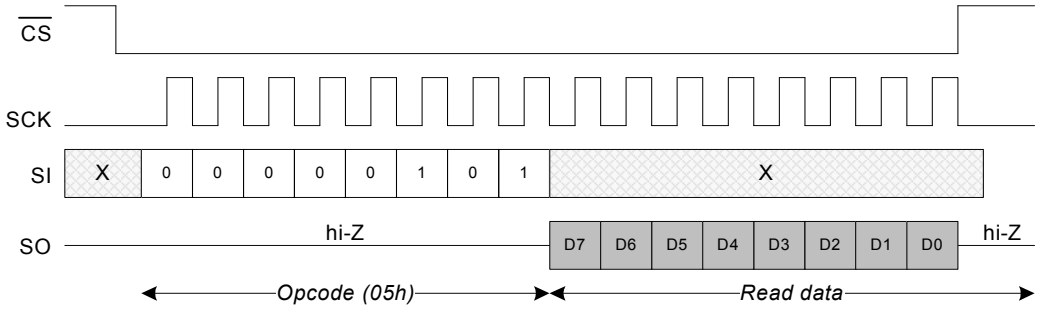



**Figure 56. RDSR Instruction in DPI Mode**

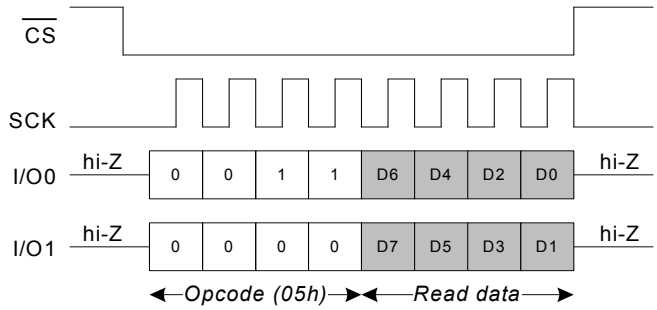
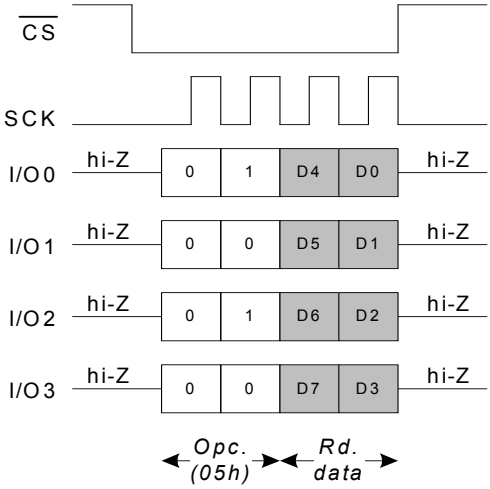



**Figure 57. RDSR Instruction in QPI Mode**




### Write Status Register (WRSR) Instruction

The WRSR instruction enables the user to write to Status Register. However, this instruction can only modify writable bits - bit 2 (BP0), bit 3 (BP1), bit 4 (BP2) bit 5 TBPROT, bit 6 SNL, and bit 7 (SRWD). WRSR instruction is a write instruction and needs the WEL bit set to '1' (by using WREN instruction). WRSR instruction opcode is issued after the falling edge of $\overline{CS}$ followed by eight bits of data to be stored in Status Register. As mentioned before, WRSR instruction can only modify bits 2, 3, 4, 5, 6, and 7 of Status Register.

**Note** The values written to Status Register are saved to nonvolatile memory only after a STORE operation. If AutoStore is disabled, any modifications to the Status Register must be secured by performing a Software STORE operation.

**Note** The WEL bit in the Status Register resets to '0' on completion of a Status Register Write sequence.

**Figure 58. WRSR Instruction in SPI Mode**

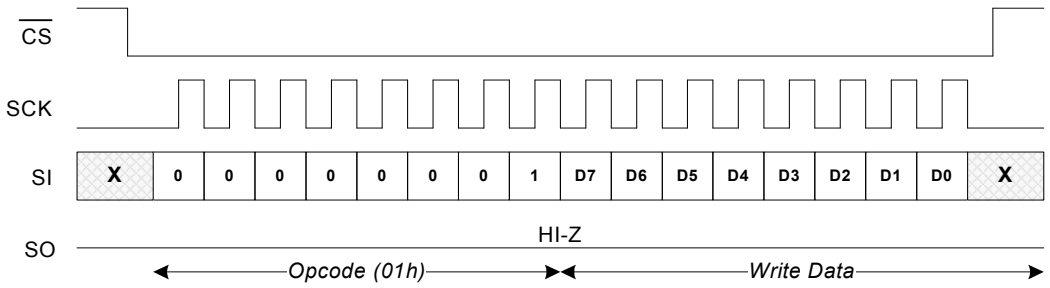
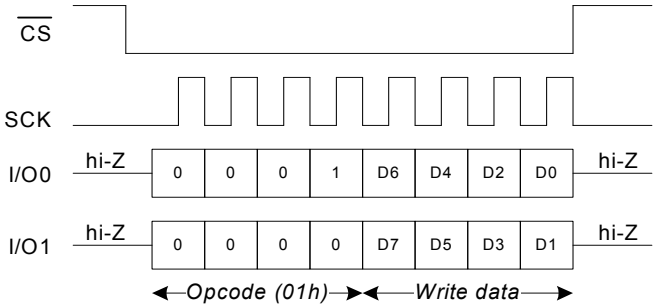



**Figure 59. WRSR Instruction in DPI Mode**




## Read Configuration Register (RDCR) Instruction

The RDCR instruction provides access to Configuration Register at SPI frequencies up to 108 MHz. The following figures provide the configuration register instruction transfer waveforms in SPI, DPI, and QPI modes.

**Note** After the last bit of Configuration Register is read, the device loops back to the first bit of the Configuration register.

**Figure 60. WRSR Instruction in QPI Mode**

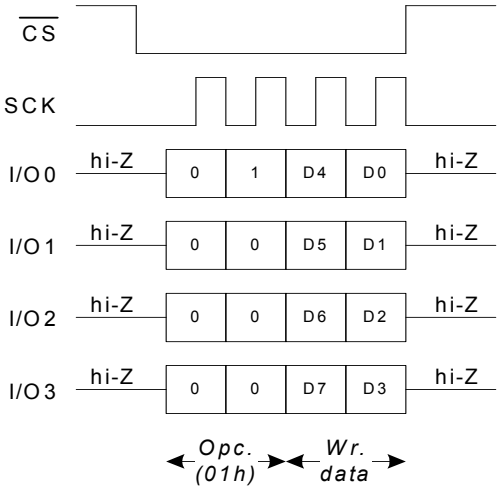



**Figure 61. RDCR Instruction in SPI Mode**

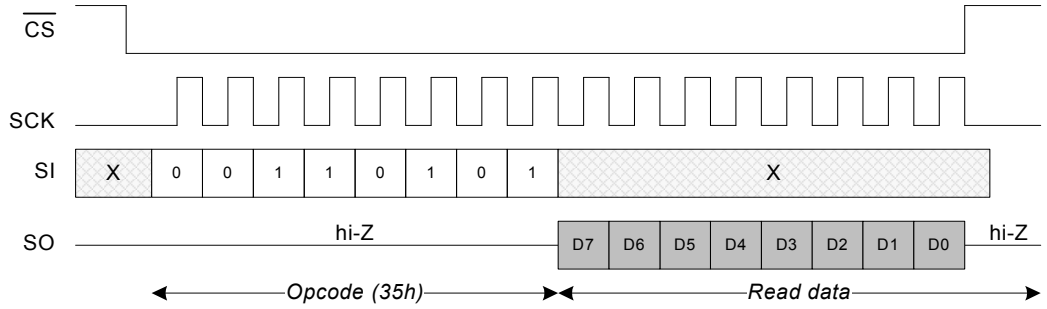


**Figure 62. RDCR Instruction in DPI Mode**

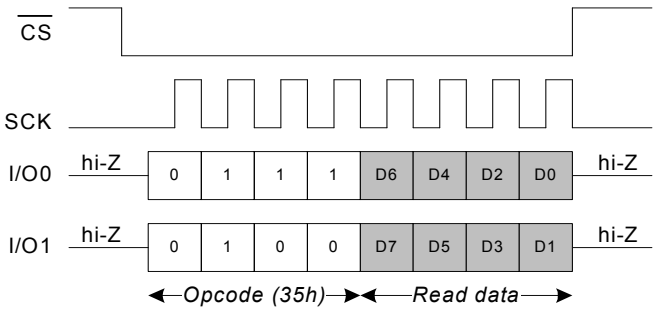
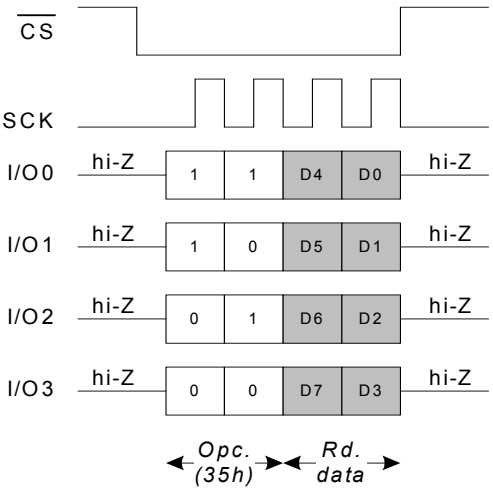



**Figure 63. RDCR Instruction in QPI Mode**




**Note** Quad bit CR[1] must be logic ‘1’ before executing the RDCR instruction in QPI mode.

## Write Configuration Register (WRCR) Instruction

The WRCR instruction writes enables user to change the data width of the device by setting the Quad Bit. The Quad bit must be set to one when using Read Quad Out, Quad I/O Read, and Quad Input Write commands. The QUAD bit is non-volatile.

**Note** Enabling the QPI mode (QPIEN Instruction) does not set the Quad bit in configuration register.

**Note** It is recommended that RFU bits should always be written as provided in Table 8.

**Figure 64. WRCR Instruction in SPI Mode**

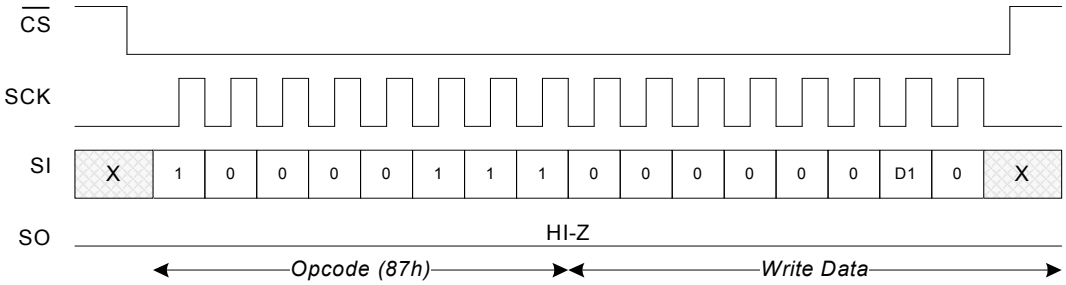
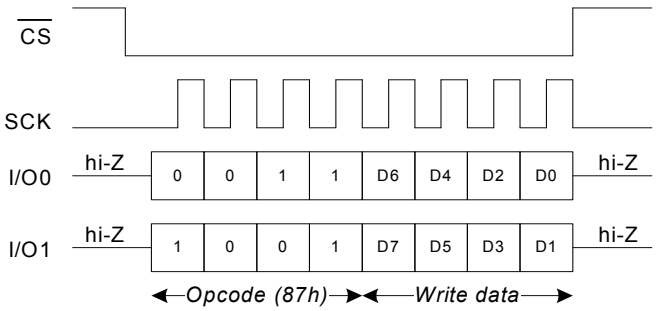



**Figure 65. WRCR Instruction in DPI Mode**



### Identification Register (RDID) Instruction

RDID instruction is used to read the JEDEC-assigned manufacturer ID and product ID of the device at an SPI frequency of up to 40 MHz. This instruction can be used to identify a device on the bus. An RDID instruction can be issued by shifting the opcode for RDID after CS# goes LOW.

Device ID is 4-byte read only code identifying 1-Mbit QPI nvSRAM product uniquely. This includes the product family code, configuration and density of the product.

The RDID command reads the 4 byte Device ID structure (the structure cannot be written to). The structure is accessed one Byte at a time. The first accessed Byte is the most significant byte of the structure ID[31:24], the second accessed byte is ID[23:16], …, the last accessed Byte is ID[7:0].

**Note** As the structure is always accessed in the same order, no address transfer is required. Instead an internal 2-bit address pointer is used that is initialized to “0” when the opcode is decoded. After each Byte access the internal address pointer is incremented. The address pointer wraps around from ‘3’ to ‘0’; after the 4th Byte ID[7:0] is accessed, the 1st Byte ID[31:24] is accessed. This command can be issued in SPI, DPI or QPI Modes.

**Table 12. Device Identification**

| Device | Manufacturer ID | Product ID | Density | Die REV |
|---|---|---|---|---|
| | 31-21 | 20-7 | 6-3 | 2-0 |
| | 11 bits | 14 bits | 4 bits | 3 bits |
| CY14V101QS | 00000110100 | 00001100010001 | 0100 | 001 |

**Figure 66. RDID Instruction in SPI Mode**

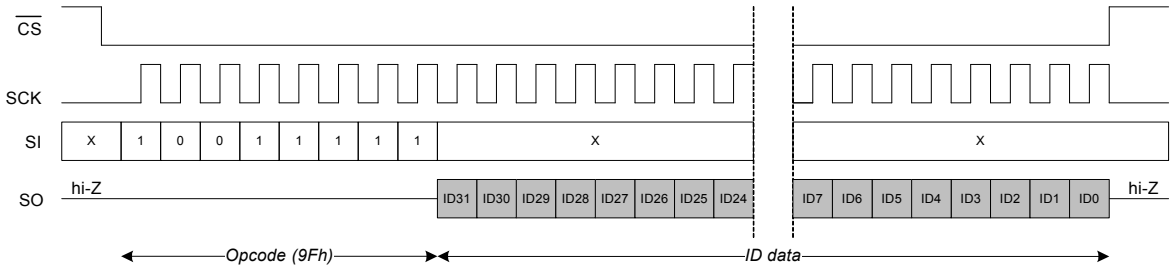



**Figure 67. RDID Instruction in DPI Mode**

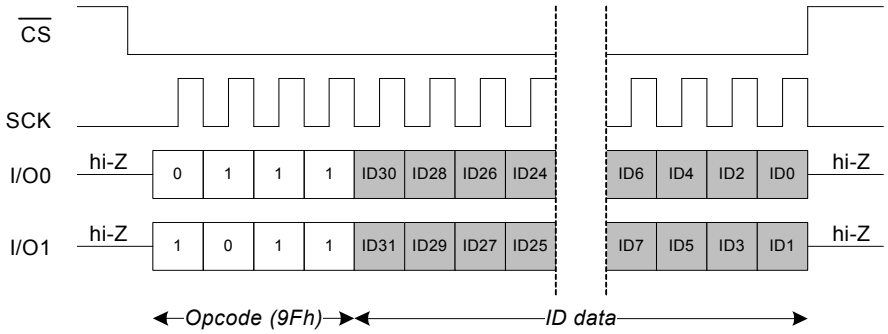
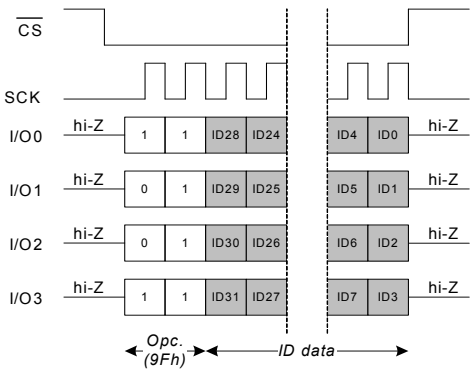



**Figure 68. RDID Instruction in QPI Mode**




**Note:** Quad bit CR[1] must be logic ‘1’ before executing the RDID instruction in QPI mode.

### Identification Register (FAST_RDID) Instruction

The FAST_RDID instruction is similar to RDID except it allows for a dummy byte after the opcode. FAST_RDID instruction is used to read the JEDEC-assigned manufacturer ID and product ID of the device at an SPI frequency of up to 108 MHz.

**Figure 69. FAST_RDID in SPI Mode**

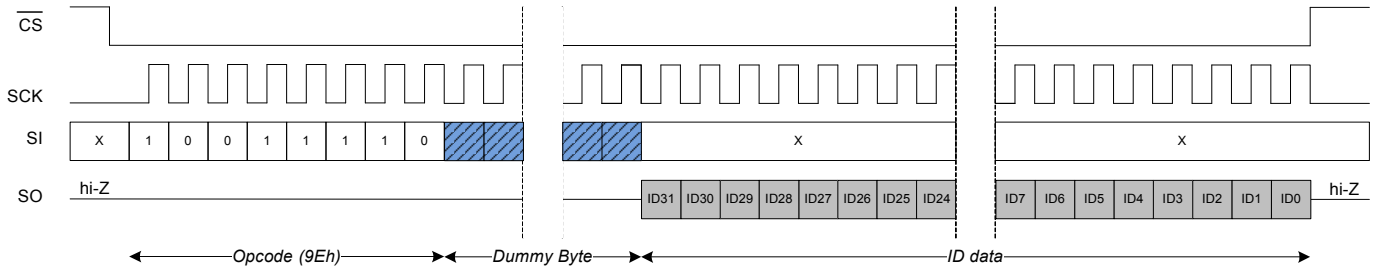



**Figure 70. FAST_RDID in DPI Mode**

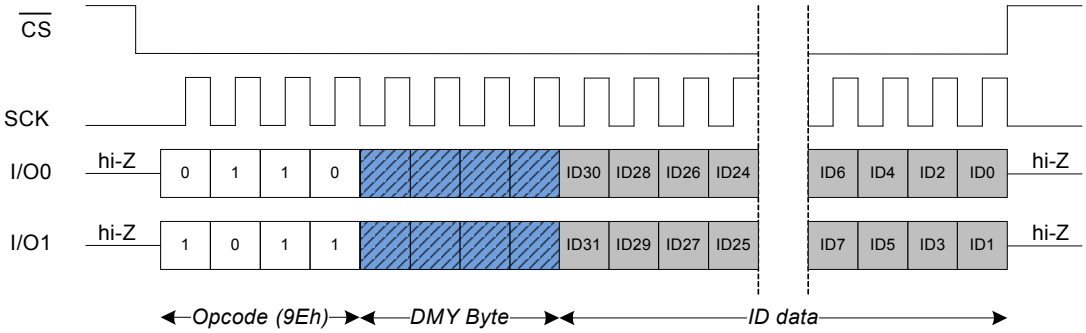
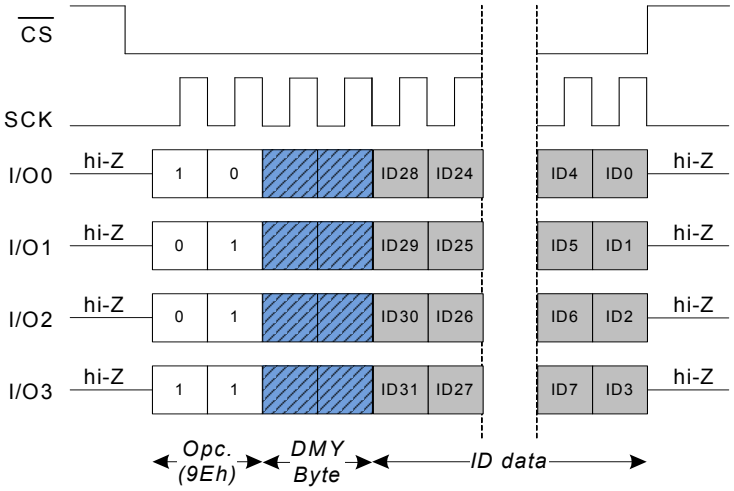



**Figure 71. FAST_RDID in QPI Mode**



### Serial Number Register Write (WRSN) Instruction

The serial number is an 8 byte programmable memory space provided to the user to uniquely identify the device. It typically consists of a two byte Customer ID, followed by five bytes of unique serial number and one byte of CRC check. However, device does not calculate the CRC and it is up to the system designer to utilize the eight byte memory space in whatever manner desired. The default value for eight byte locations are set to ‘0x00’.

The serial number is written using WRSN command. To write serial number, the write must be enabled using the WREN command. The WRSN command can be used in burst mode to write all the 8 bytes of serial number. After the last byte of serial number is written, the device loops back to the first (MSB) byte of the serial number. The serial number is locked using the SNL bit of the Status Register. Once this bit is set to '1', no modification to the serial number is possible. After the SNL bit is set to '1', using the WRSN command has no effect on the serial number. This command requires the WEL bit to be set before it can be executed. The WEL bit is reset to '0' after completion of this command if SRWD bit in the Status register is not set to ‘1’ This command can be issued in SPI, DPI or QPI Modes.

The serial number is written using the WRSN instruction at an SPI frequency of up to 108 MHz.

**Note** A STORE operation (AutoStore or Software STORE) is required to store the serial number in the nonvolatile memory. If AutoStore is disabled, you must perform a Software STORE operation to secure and lock the serial number. If the SNL bit is set to ‘1’ and is not stored (AutoStore disabled), the SNL bit and serial number defaults to ‘0’ at the next power cycle. If the SNL bit is set to ‘1’ and is stored, the SNL bit can never be cleared to ‘0’. This instruction requires the WEL bit to be set before it can be executed. This instruction can be issued in SPI, DPI, or QPI modes.

**Note** The WEL bit is reset to ‘0’ after completion of this instruction.

**Figure 72. WRSN Instruction in SPI Mode**

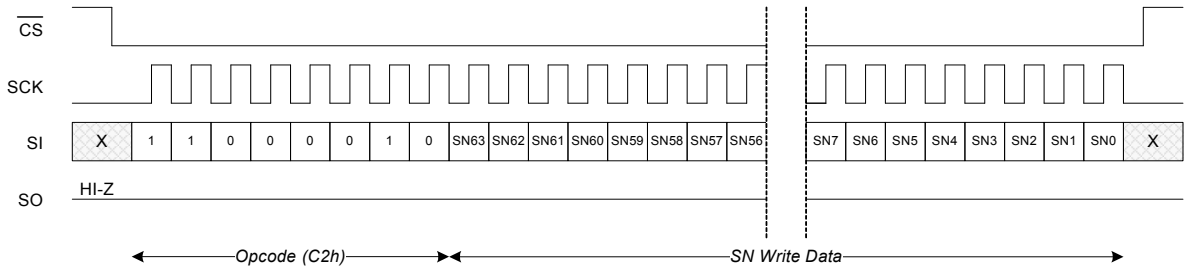



**Figure 73. WRSN Instruction in DPI Mode**

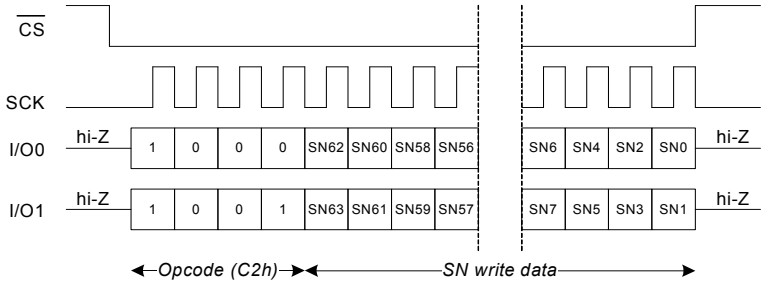
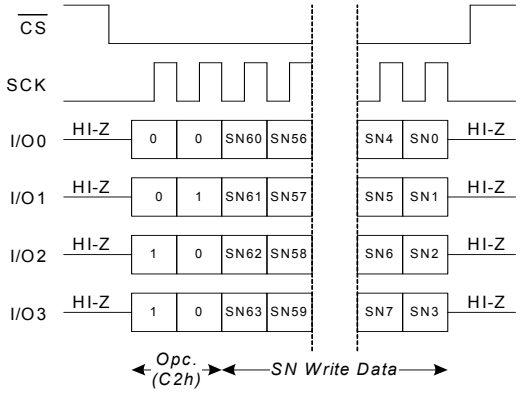



**Figure 74. WRSN Instruction in QPI Mode**



### Serial Number Register Read (RDSN) Instruction

The serial number is read using the RDSN instruction at an SPI frequency of up to 40 MHz. A serial number read may be performed in burst mode to read all the eight bytes at once. After the last byte of serial number is read, the device loops back to the first (MSB) byte of the serial number. An RDSN instruction can be issued by shifting the opcode for RDSN after $\overline{CS}$ goes LOW. This is followed by nvSRAM shifting out the eight bytes of the serial number. This instruction can be issued in SPI, DPI or QPI modes.

**Figure 75. RDSN Instruction in SPI Mode**

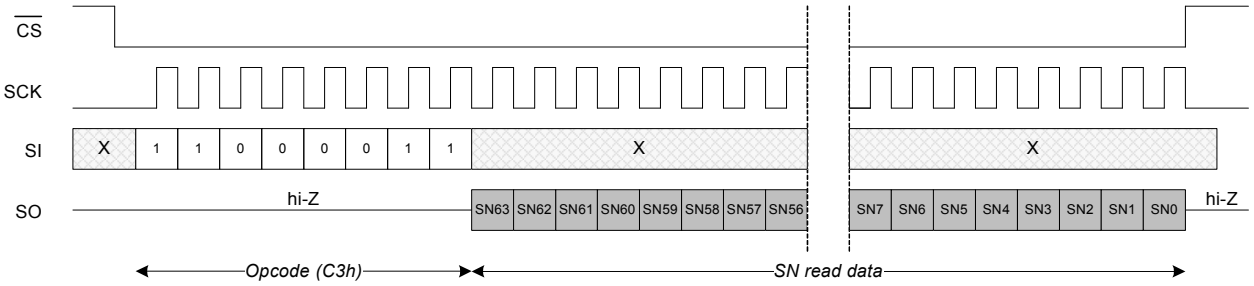



**Figure 76. RDSN Instruction in DPI Mode**

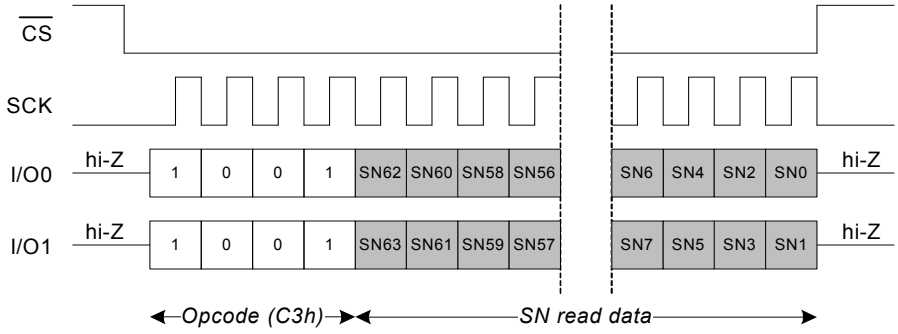
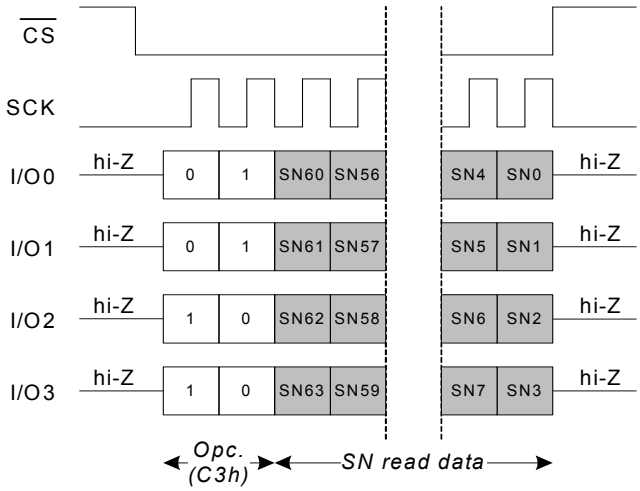



**Figure 77. RDSN Instruction in QPI Mode**




**Note** Quad bit CR[1] must be logic ‘1’ before executing the RDSN instruction in QPI mode.

### Fast Read Serial Number Register (FAST_RDSN) Instruction

The FAST_RDSN instruction is similar to RDSN except it allows for a dummy byte after the opcode. FAST_RDSN instruction is used up to 108 MHz.

**Figure 78. FAST_RDSN Instruction in SPI Mode**

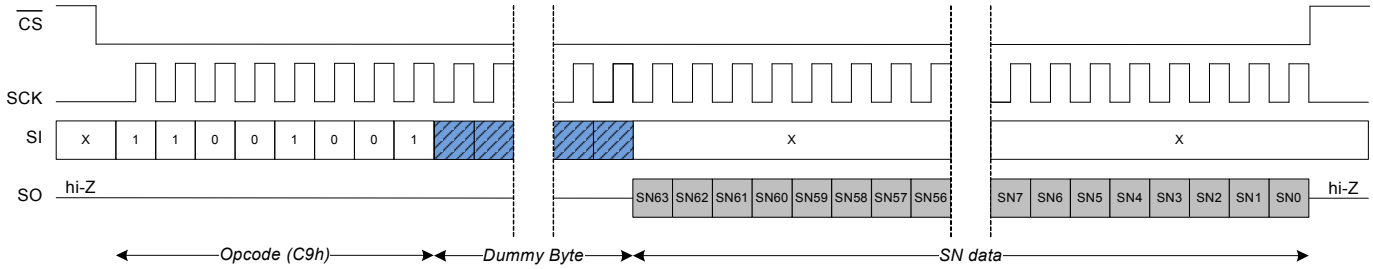



**Figure 79. FAST_RDSN Instruction in DPI Mode**

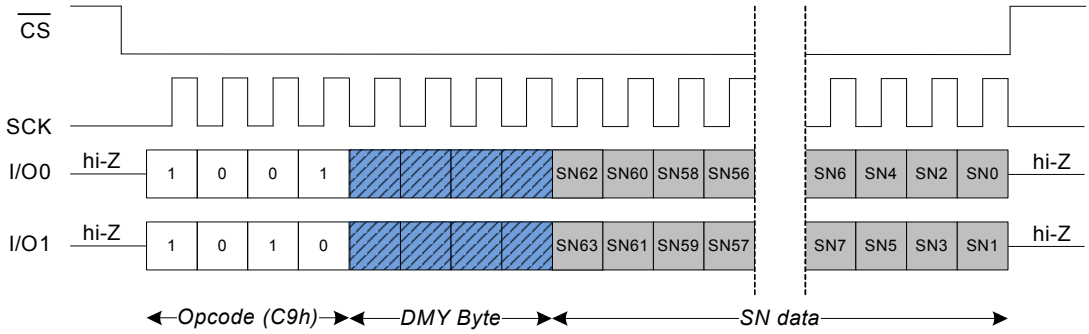
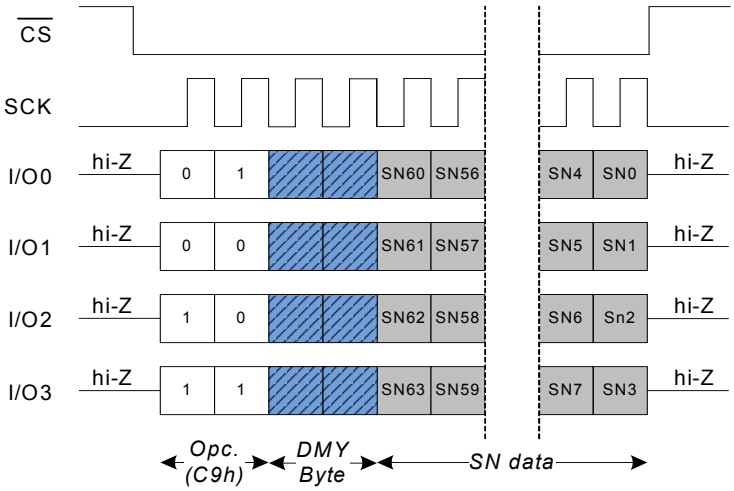



**Figure 80. FAST_RDSN Instruction in QPI Mode**



## NV Specific Instructions

The nvSRAM device provides four special instructions, which enable access to the nvSRAM specific functions: STORE, RECALL, ASEN, and ASDI.

### Software Store (STORE) Instruction

When a STORE instruction is executed, nvSRAM performs a Software STORE operation. The STORE operation is performed irrespective of whether a write has taken place since the last STORE or RECALL operation. To issue this instruction, the device must be write enabled (WEL bit = '1'). The instruction can be issued in SPI, DPI and QPI modes.

**Note** The WEL bit is cleared on the positive edge of $\overline{CS}$ following the STORE instruction.

**Figure 81. STORE Instruction in SPI Mode**

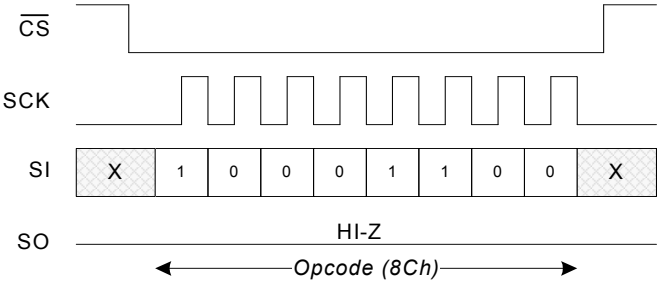
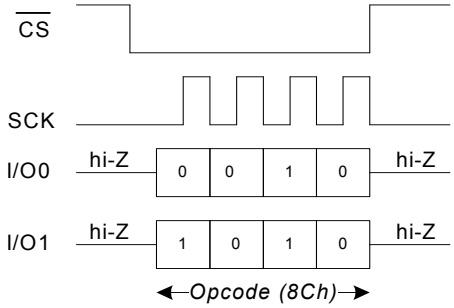



**Figure 82. STORE Instruction in DPI Mode**




### Software Recall (RECALL) Instruction

When a RECALL instruction is executed, nvSRAM performs a Software RECALL operation. To issue this instruction, the device must be write enabled (WEL = '1'). This instruction can be issued in SPI, DPI, or QPI modes.

**Note** The WEL bit is cleared on the positive edge of $\overline{CS}$ following the RECALL instruction.

**Figure 83. STORE Instruction in QPI Mode**

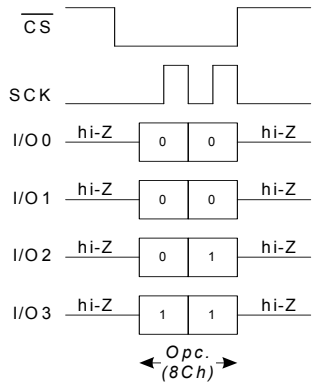



**Figure 84. RECALL Instruction in SPI Mode**

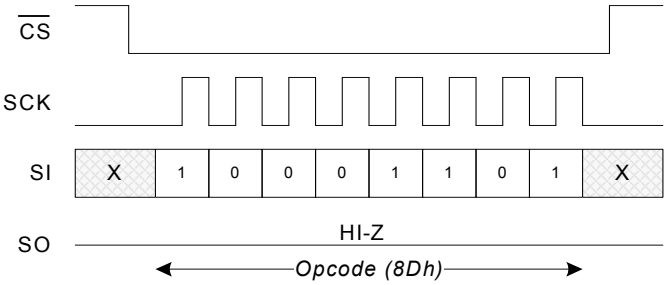



**Figure 85. RECALL Instruction in DPI Mode**

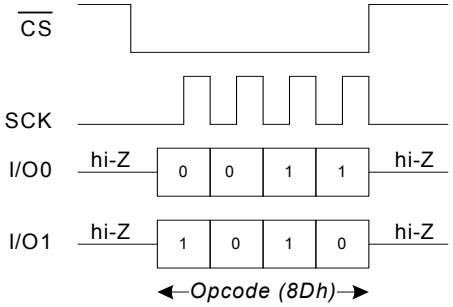
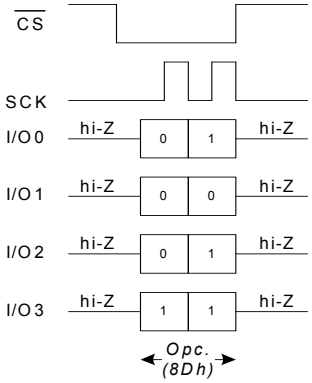



**Figure 86. RECALL Instruction in QPI Mode**



## Autostore Enable (ASEN) Instruction

The AutoStore Enable instruction enables the AutoStore on the nvSRAM device. This setting is not nonvolatile and needs to be followed by a STORE sequence to survive the power cycle. To issue this instruction, the device must be write enabled (WEL = '1'). This instruction can be issued in SPI, DPIO, or QPI modes.

**Note** If the ASDI and ASEN instructions are executed, the device is busy for the duration of software sequence processing time ($t_{SS}$).

**Note** The WEL bit is cleared on the positive edge of $\overline{CS}$ following the ASE instruction.

**Figure 87. ASEN Instruction in SPI Mode**

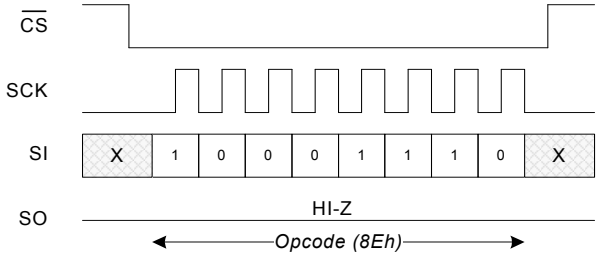



**Figure 88. ASEN Instruction in DPI Mode**

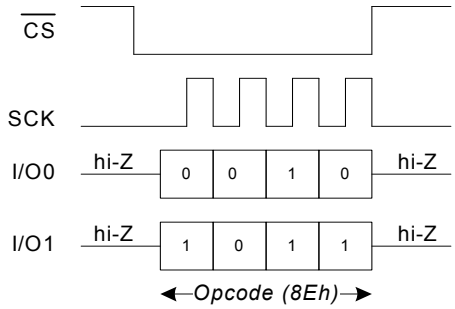
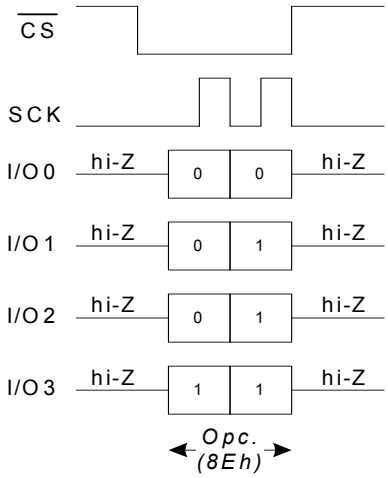



**Figure 89. ASEN Instruction in QPI Mode**




## Autostore Disable (ASDI) Instruction

AutoStore is enabled by default in this device. The ASDI instruction disables the AutoStore. This setting is not nonvolatile and needs to be followed by a STORE sequence to survive the power cycle. To issue this instruction, the device must be write enabled (WEL = '1'). This instruction can be issued in SPI, DPI, or QPI modes.

**Note** The WEL bit is cleared on the positive edge of $\overline{CS}$ following the ASDI instruction.

**Figure 90. ASDI Instruction in SPI Mode**

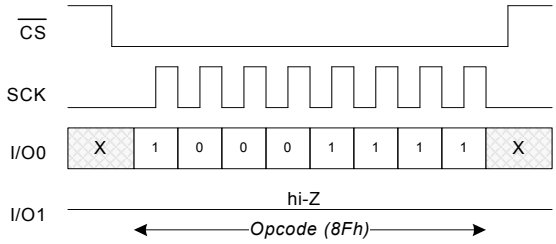



**Figure 91. ASDI Instruction in DPI Mode**

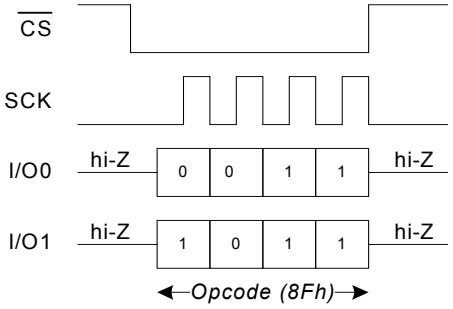
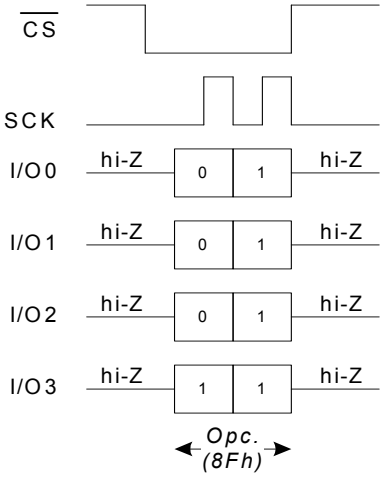



**Figure 92. ASDI Instruction in QPI Mode**




**Note:** Quad bit CR[1] must be logic '1' before executing the ASDI instruction in QPI mode.

## Maximum Ratings

Exceeding maximum ratings may shorten the useful life of the device. These user guidelines are not tested.

Storage temperature ..... –65 °C to +150 °C

Maximum accumulated storage time

At 150 °C ambient temperature ..... 1000 h

At 85 °C ambient temperature ..... 20 Years

Maximum junction temperature ..... 150 °C

Supply voltage on $V_{CC}$ relative to $V_{SS}$ .....–0.5 V to +4.1 V

Supply voltage on $V_{CCQ}$ relative to $V_{SS}$ .....–0.5 V to +2.45 V

DC voltage applied to outputs in HI-Z state .....–0.5 V to $V_{CCQ}$ + 0.5 V

Input voltage .....–0.5 V to $V_{CCQ}$ + 0.5 V

Transient voltage (< 20 ns) on any pin to ground potential .....–2.0 V to $V_{CCQ}$ + 2.0 V

Package power dissipation capability ($T_A$ = 25 °C)

16-pin SOIC ..... 1.0 W

24-ball FBGA ..... 1.0W

Package power dissipation capability ($T_A$ = 25 °C) ..... 1.0 W

Surface mount lead soldering temperature (3 seconds) ..... +260 °C

DC output current (1 output at a time, 1-s duration) ... 15 mA

Static discharge voltage (per MIL-STD-883, Method 3015) ..... > 2001 V

Latch-up current ..... > 140 mA

## Operating Range

| Range | Ambient Temperature | $V_{CC}$ | $V_{CCQ}$ |
|---|---|---|---|
| Industrial | –40 °C to +85 °C | 2.7 V to 3.6 V | 1.71 V to 2.0 V |
| Extended Industrial | –40 °C to +105 °C | 2.7 V to 3.6 V | 1.71 V to 2.0 V |

## DC Specifications

| Parameter | Description | Test Conditions | Min | Typ[1] | Max | Units |
|---|---|---|---|---|---|---|
| $V_{CC}$ | Power Supply - Core voltage | – | 2.70 | 3.00 | 3.60 | V |
| $V_{CCQ}$ | Power Supply - I/O voltage | – | 1.71 | 1.80 | 2.00 | V |
| $I_{CC1}$ | Average Read/Write $V_{CC}$ Current (all inputs toggling, no output load) | SPI = 1 MHz | – | – | 1.00 | mA |
| | | SPI = 40 MHz | – | – | 3.00 | mA |
| | | QPI = 108 MHz | – | – | 33.00 | mA |
| $I_{CCQ1}$ | Average $V_{CCQ}$ Current (all inputs toggling, no output load) | SPI = 1 MHz | – | – | 150.00 | µA |
| | | SPI= 40 MHz | – | – | 1.00 | mA |
| | | QPI = 108 MHz | – | – | 5.00 | mA |
| $I_{SB1}$ | Standby Current at 85 °C ($V_{CC}$ + $V_{CCQ}$) | $\overline{CS}$ > ($V_{CCQ}$ – 0.2 V). Standby current level after nonvolatile cycle is complete. ($\overline{CS}$ High, Other I/Os have no restrictions, $f_{SCK}$ ≤ 108 MHz) | – | – | 1.70 | mA |
| | Standby Current at105 °C ($V_{CC}$ + $V_{CCQ}$) | | – | – | 2.00 | mA |
| $I_{SB2}$ | Standby Current at 85 °C ($V_{CC}$ + $V_{CCQ}$) | $\overline{CS}$ > ($V_{CCQ}$ – 0.2 V). Standby current level after nonvolatile cycle is complete. All I/Os Static, $f_{SCK}$ = 0 MHz | – | – | 280.00 | µA |
| | Standby Current at 105 °C ($V_{CC}$ + $V_{CCQ}$) | | – | – | 540.00 | µA |
| $I_{CC2}$ | Average $V_{CC}$ current during STORE | – | – | – | 6.00 | mA |
| $I_{CC4}$ | Average $V_{CAP}$ current during AUTOSTORE | – | – | – | 6.00 | mA |

**Notes**

1. Typical values are at 25 °C, $V_{CC}$ = $V_{CC(Typ)}$ and $V_{CC\,Q}$= $V_{CCQ(Typ)}$. Not 100% tested.

## DC Specifications (continued)

| Parameter | Description | Test Conditions | Min | Typ[1] | Max | Units |
|---|---|---|---|---|---|---|
| $I_{SLEEP}$ | Sleep Mode current at 85 °C ($V_{CC}$ + $V_{CCQ}$) | $\overline{CS}$ > ($V_{CCQ}$ – 0.2 V). Sleep current level after nonvolatile cycle is complete. All I/Os Static, $f_{SCK}$ = 0 MHz | – | – | 280.00 | µA |
| $I_{ZZ}$ | Hibernate mode current at 85 °C ($V_{CC}$ + $V_{CCQ}$) | $\overline{CS}$ > ($V_{CCQ}$ – 0.2 V). $t_{HIBEN}$ time after HIBEN Instruction is registered. All inputs are static and configured at CMOS logic level. | – | – | 8.00 | µA |
| $I_{IX}$ | Input leakage current (except $\overline{HSB}$) | $V_{CCQ}$ = Max, VSS < VIN < $V_{CCQ}$ – | –1.00 | – | 1.00 | µA |
| | Input leakage current (for $\overline{HSB}$) | | –100.00 | – | 1.00 | µA |
| | Input leakage current (for $\overline{WP}$ in SPI/DPI modes) | | –2 | – | 1 | µA |
| $I_{OZ}$ | Off State Output Leakage Current | $V_{CCQ}$ = Max, VSS < VIN < $V_{CCQ}$ | –1.00 | – | 1.00 | µA |
| $V_{IH}$ | Input high voltage | – | 0.70 * $V_{CCQ}$ | – | $V_{CCQ}$ + 0.30 | V |
| $V_{IL}$ | Input low voltage | – | –0.30 | – | 0.30 * $V_{CCQ}$ | V |
| $V_{OH}$ | Output high voltage at -2 mA | $I_{OH}$ = –2 mA | $V_{CCQ}$–0.45 | – | – | V |
| $V_{OL}$ | Output low voltage at 2 mA | $I_{OL}$= 2 mA | – | – | 0.45 | V |
| $V_{CAP}$[2] | Storage capacitor | Between $V_{CAP}$ pin and $V_{SS}$ | 61.00 | 68.00 | 120.00 | µF |
| $V_{VCAP}$[3] | Maximum Voltage Driven on $V_{CAP}$ Pin | – | – | – | $V_{CC}$ | V |

## Data Retention and Endurance

| Parameter | Description | Min | Unit |
|---|---|---|---|
| $DATA_R$ | Data retention at 85 °C | 20 | Years |
| $NV_C$ | Nonvolatile STORE operations | 1,000 | K |

## Capacitance

| Parameter[3] | Description | Test Conditions | Max | Unit |
|---|---|---|---|---|
| $C_{IN}$ | Input capacitance | $T_A$ = 25 °C, f = 1 MHz, $V_{CC}$ = $V_{CC(typ)}$, $V_{CC\,Q}$= $V_{CCQ(typ)}$ | 6.00 | pF |
| $C_{SCK}$ | Clock input capacitance | | | |
| $C_{OUT}$ | Output pin capacitance | | | |

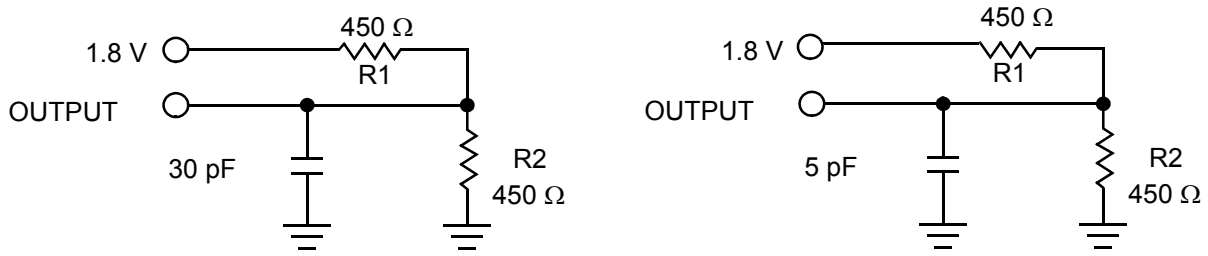
## Thermal Resistance

| Parameter[3] | Description | Test Conditions | 16-Pin SOIC | 24-Ball FBGA | Unit |
|---|---|---|---|---|---|
| $\Theta_{JA}$ | Thermal resistance (junction to ambient) | Test conditions follow standard test methods and procedures for measuring thermal impedance, per EIA/JESD51. | 61.21 | 32.08 | °C/W |
| $\Theta_{JC}$ | Thermal resistance (junction to case) | | 26.20 | 14.29 | |

**Notes**
2. Min $V_{CAP}$ value guarantees that there is a sufficient charge available to complete a successful AutoStore operation. Max $V_{CAP}$ value guarantees that the capacitor on $V_{CAP}$ is charged to a minimum voltage during a power-up RECALL cycle so that an immediate power-down cycle can complete a successful AutoStore. Therefore, it is always recommended to use a capacitor within the specified min and max limits. Refer application note AN43593 for more details on $V_{CAP}$ options.
3. These parameters are guaranteed by design and are not tested.

## AC Test Loads and Waveforms

**Figure 93. AC Test Loads and Waveforms**




## AC Test Conditions

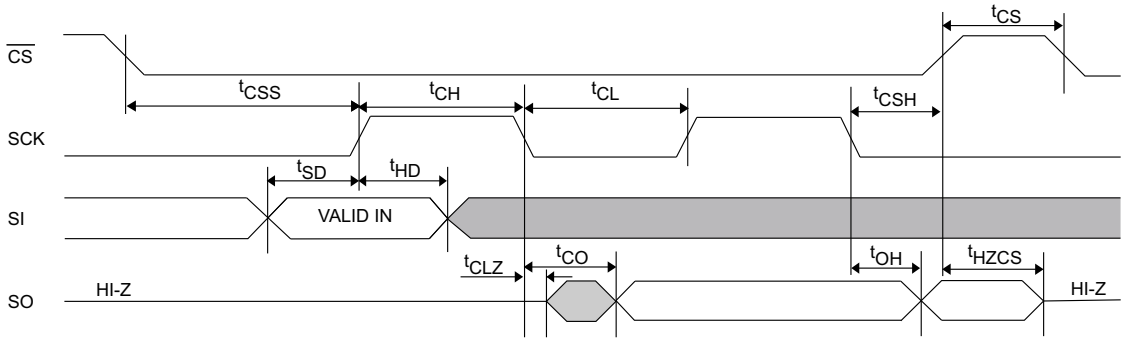
| Description | CY14V101QS |
|---|---|
| Input pulse levels | 0 V to 1.8 V |
| Input rise and fall times (10%–90%) | ≤ 1.8 ns |
| Input and output timing reference levels | 0.9 V |

## AC Switching Characteristics

| Parameter[4] | Description | Min | Max | Units |
|---|---|---|---|---|
| $f_{SCK}$ | Clock frequency (QPI) | – | 108.00 | MHz |
| $t_{CL}$ | Clock Pulse Width Low | 0.45 * $1/f_{SCK}$ | – | ns |
| $t_{CH}$ | Clock Pulse Width High | 0.45 * $1/f_{SCK}$ | – | ns |
| $t_{CS}$ | $\overline{CS}$ HIGH time | | | |
| | End of READ | 10.00 | – | ns |
| | End of WRITE | 10.00 | – | ns |
| $t_{CSS}$ | $\overline{CS}$ setup time | 5.00 | – | ns |
| $t_{CSH}$ | $\overline{CS}$ hold time | 5.00 | – | ns |
| $t_{SD}$ | Data in setup time | 2.00 | – | ns |
| $t_{HD}$ | Data in hold time | 3.00 | – | ns |
| $t_{SW}$ | $\overline{WP}$ setup time | 2.00 | – | ns |
| $t_{HW}$ | $\overline{WP}$ hold time | 2.00 | – | ns |
| $t_{CO}$ | Output Valid | – | 7.00 | ns |
| $t_{CLZ}$ | Clock Low to Output Low Z | 0.00 | – | ns |
| $t_{OH}$ | Output Hold Time | 1.00 | – | ns |
| $t_{HZCS}$[5] | Output Disable Time | – | 7.00 | ns |

## Switching Waveforms

**Figure 94. Synchronous Data Timing (Mode 0)**




**Notes**
4. Test conditions assume signal transition time of 1.8 ns or less, timing reference levels of $V_{CCQ}/2$, input pulse levels of 0 to $V_{CCQ(typ)}$, and output loading of the specified $I_{OL}/I_{OH}$ and load capacitance shown in Figure 93 on page 45.
5. These parameters are guaranteed by design and are not tested.

## AutoStore or Power-Up RECALL

Over the Operating Range

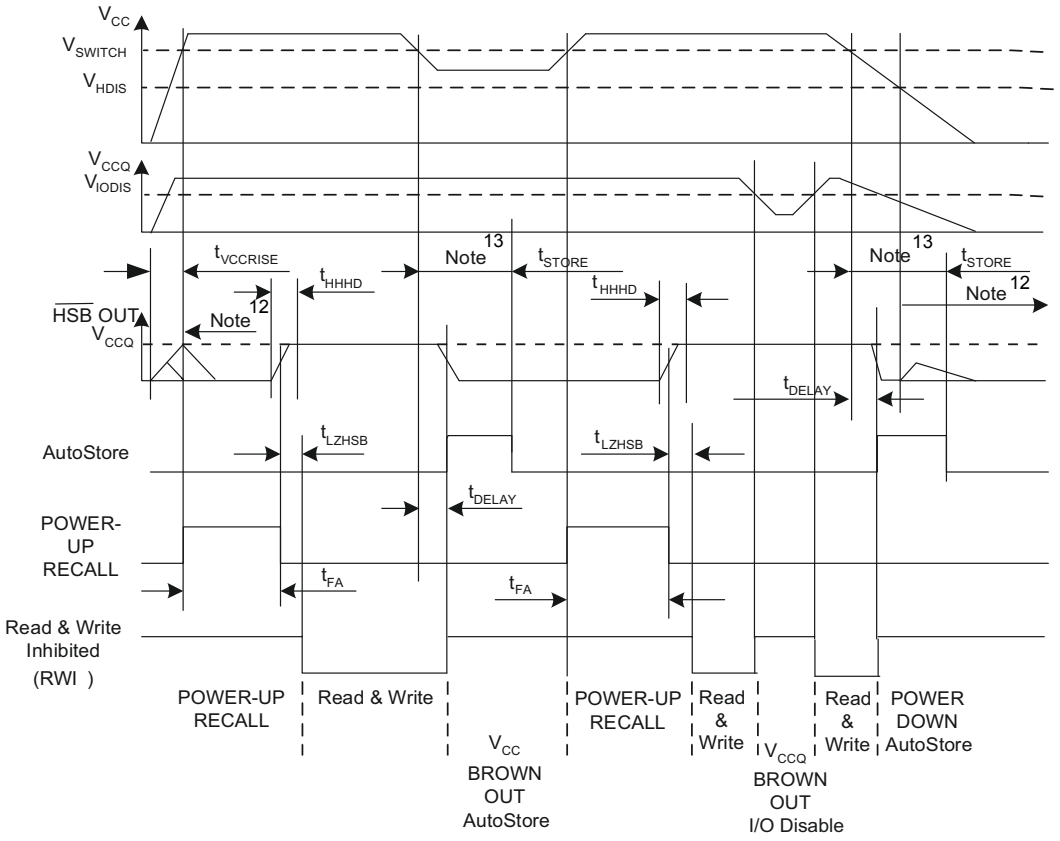
| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| $t_{FA}$[6] | Power-Up RECALL duration | – | 20.00 | ms |
| $t_{STORE}$[7] | STORE cycle duration | – | 8.00 | ms |
| $t_{DELAY}$[8] | Time to initiate store cycle | – | 25.00 | ns |
| $V_{SWITCH}$ | Low voltage trigger level for $V_{CC}$ | – | 2.60 | V |
| $t_{VCCRISE}$[9] | $V_{CC}$ rise time | 150.00 | – | µs |
| $V_{HDIS}$[9] | $\overline{HSB}$ output disable voltage | – | 1.90 | V |
| $V_{IODIS}$[10] | $\overline{I/O}$ disable voltage on $V_{CCQ}$ | – | 1.50 | V |
| $t_{LZHSB}$[9] | $\overline{HSB}$ HIGH to nvSRAM active time | – | 5.00 | µs |
| $t_{HHHD}$[9] | $\overline{HSB}$ HIGH active time | – | 500.00 | ns |
| $t_{WAKE}$ | Time for nvSRAM to wake up from HIBERNATE mode | – | 20.00 | ms |
| $t_{HIBEN}$ | Time to enter HIBERNATE mode after issuing HIBEN instruction | – | 8.00 | ms |
| $t_{SLEEP}$ | Time to enter into sleep mode after $\overline{CS}$ going HIGH | – | 0.00 | µs |
| $t_{EXSLP}$ | Time to exit from sleep mode after $\overline{CS}$ going HIGH | – | 0.00 | µs |
| $t_{RESET}$ | Soft reset duration | – | 500.00 | µs |

**Notes**

6. $t_{FA}$ starts from the time $V_{CC}$ rises above $V_{SWITCH}$.
7. If an SRAM write has not taken place since the last nonvolatile cycle, AutoStore or Hardware STORE is not initiated.
8. On a Hardware STORE, AutoStore initiation, SRAM operation continues to be enabled for time $t_{DELAY}$.
9. These parameters are guaranteed by design and are not tested.
10. $\overline{HSB}$ is not defined below $V_{IODIS}$ voltage.

## Switching Waveforms

**Figure 95. AutoStore or Power-Up RECALL[11]**




**Notes**
11. Read and write cycles are ignored during STORE, RECALL, and while $V_{CC}$ is below $V_{SWITCH}$.
12. During power-up and power-down, HSB glitches when HSB pin is pulled up through an external resistor.
13. If an SRAM write has not taken place since the last nonvolatile cycle, AutoStore or Hardware STORE is not initiated.

## Software Controlled STORE and RECALL Cycles

Over the Operating Range

| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| $t_{RECALL}$ | RECALL duration | – | 500 | µs |
| $t_{SS}$[14, 15] | Soft sequence processing time | – | 500 | µs |

## Switching Waveforms

**Figure 96. Software STORE Cycle[15]**

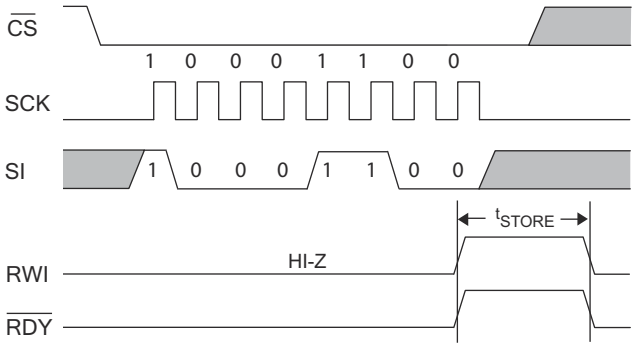



**Figure 97. Software RECALL Cycle[15]**

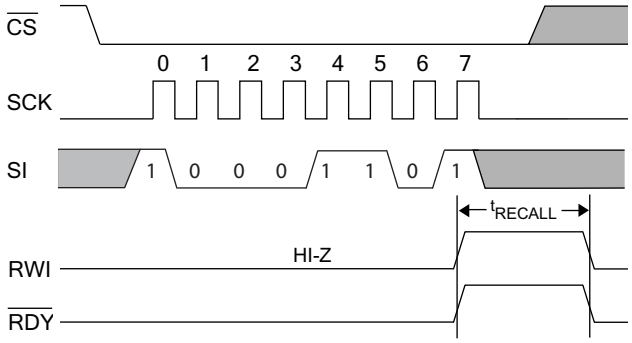



**Figure 98. AutoStore Enable Cycle**

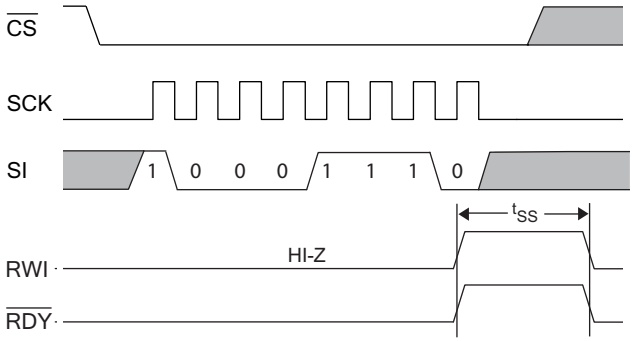
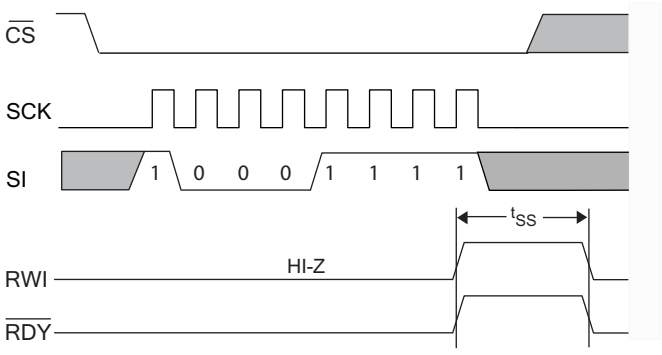



**Figure 99. AutoStore Disable Cycle**




**Notes**
14. This is the amount of time it takes to take action on a soft sequence command. Vcc power must remain HIGH to effectively register command.
15. Commands such as STORE and RECALL lock out I/O until operation is complete which further increases this time. See the specific command.

## Hardware STORE Cycle

Over the Operating Range

| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| $t_{PHSB}$ | Hardware STORE pulse width | 15 | 600 | ns |

## Switching Waveforms

**Figure 100. Hardware STORE Cycle[16]**

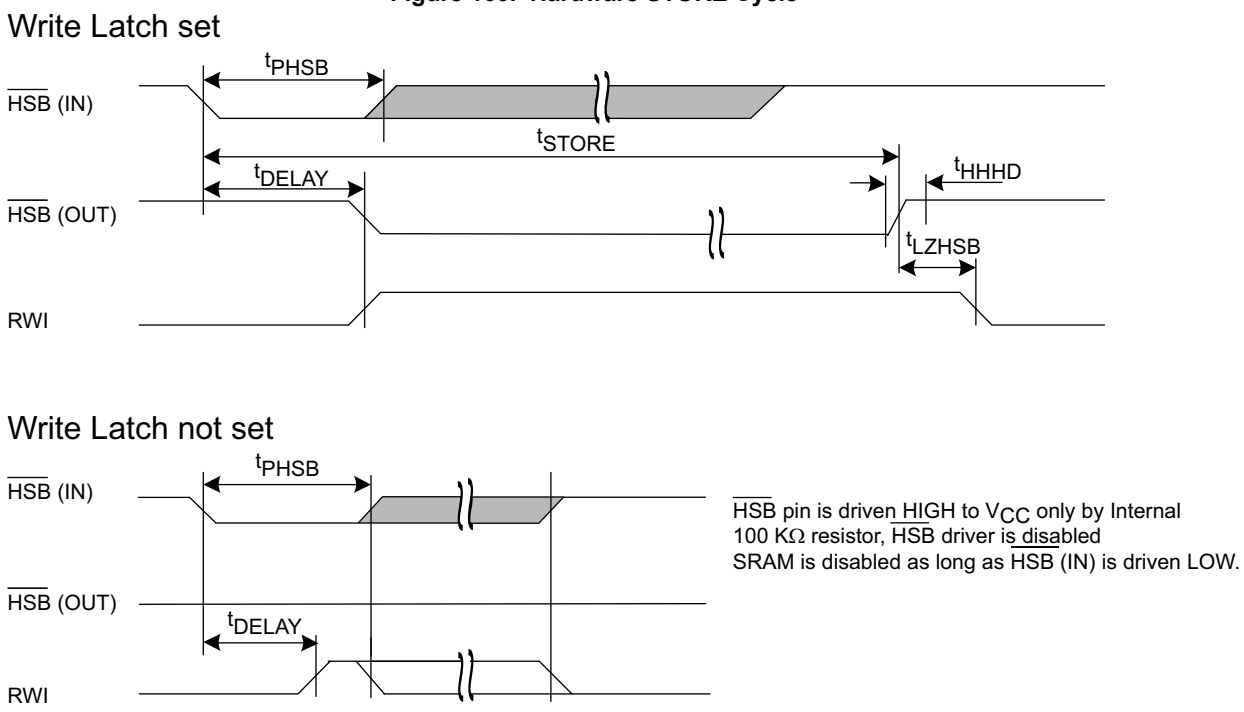
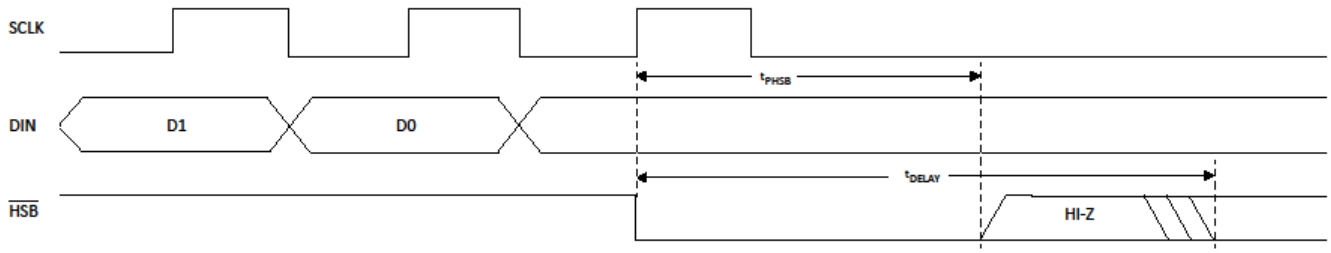



**Figure 101. Data Valid to HSB**



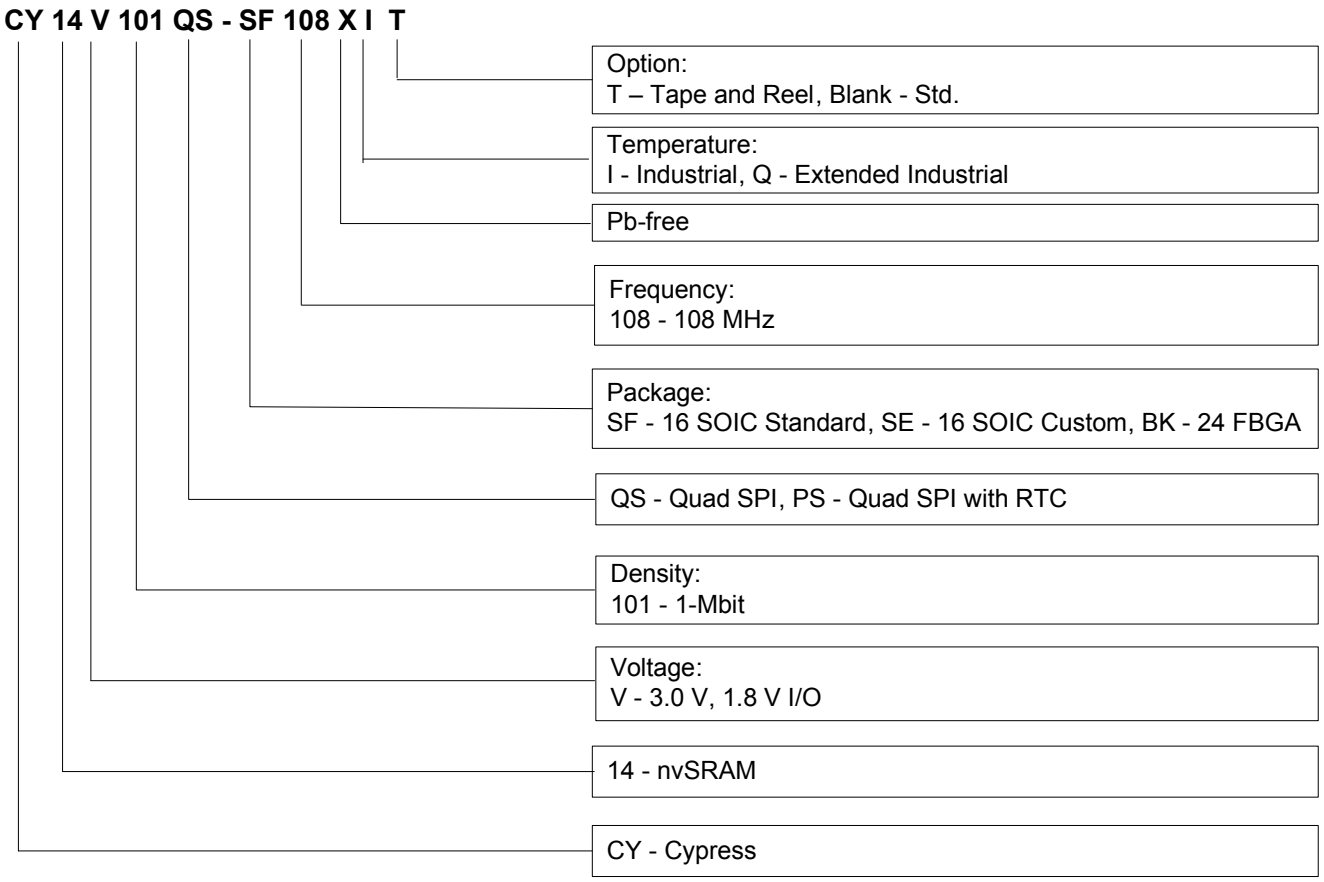

**Note**
16. If an SRAM write has not taken place since the last nonvolatile cycle, AutoStore or Hardware STORE is not initiated.

## Ordering Information

| Ordering Code | Package Diagram | Package Type, Pinout | Operating Range |
|---|---|---|---|
| CY14V101QS-BK108XI | 001-97209 | 24-FBGA, Standard | Industrial |
| CY14V101QS-BK108XIT | | | |
| CY14V101QS-BK108XQ | | | Extended Industrial |
| CY14V101QS-BK108XQT | | | |
| CY14V101QS-SE108XI | 51-85022 | 16-SOIC, Custom | Industrial |
| CY14V101QS-SE108XIT | | | |
| CY14V101QS-SE108XQ | | | Extended Industrial |
| CY14V101QS-SE108XQT | | | |
| CY14V101QS-SF108XI | | 16-SOIC, Standard | Industrial |
| CY14V101QS-SF108XIT | | | |
| CY14V101QS-SF108XQ | | | Extended Industrial |
| CY14V101QS-SF108XQT | | | |

All these parts are Pb-free. Contact your local Cypress sales representative for availability of these parts.

### Ordering Code Definitions



## Package Diagrams

**Figure 102. 16-Pin SOIC (0.413 × 0.299 × 0.0932 Inches) Package Outline, 51-85022**

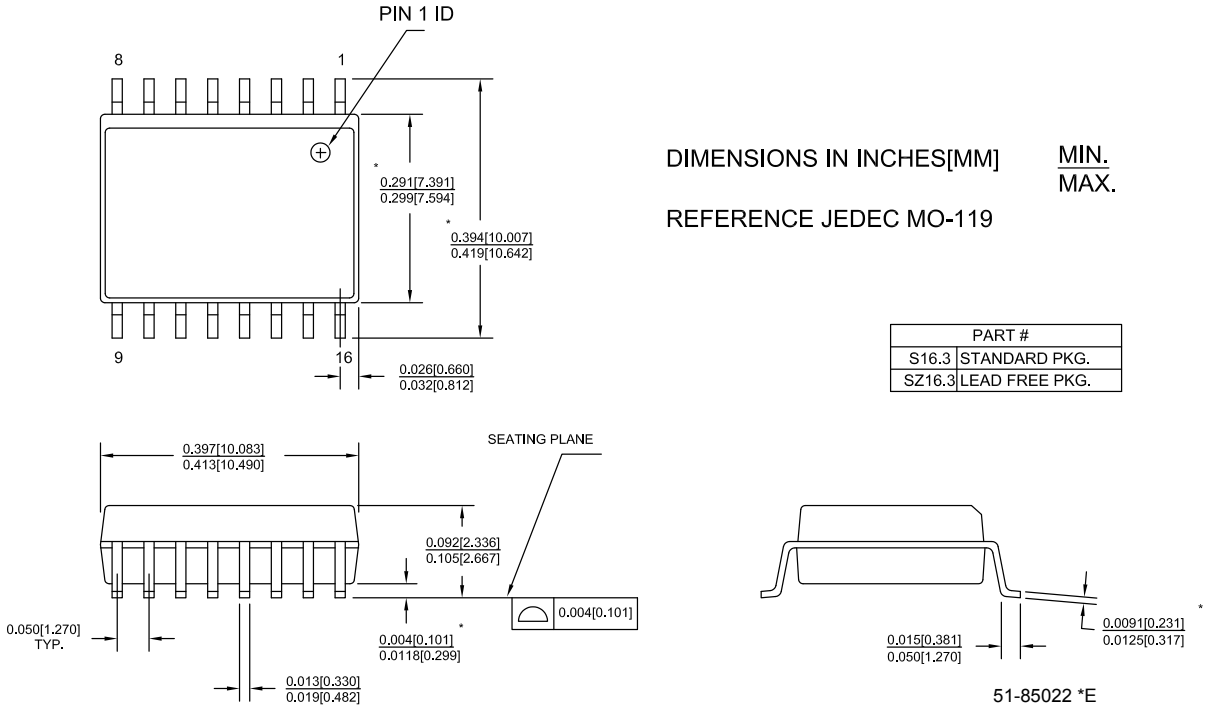
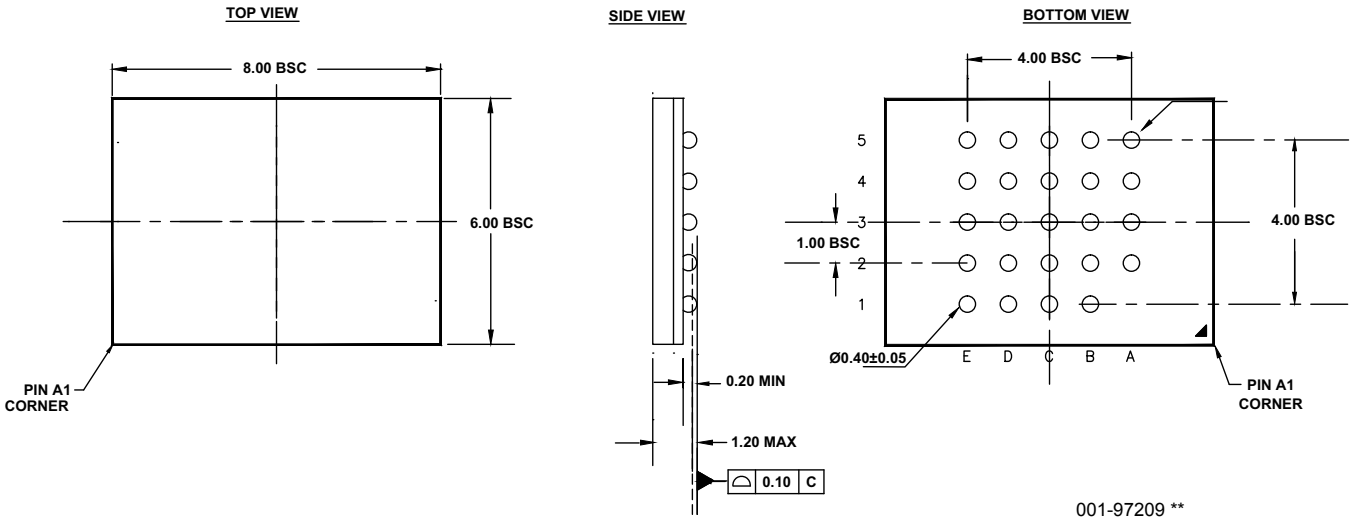



**Figure 103. 24-Ball FBGA Package**



## Acronyms

| Acronym | Description |
|---|---|
| CPHA | clock phase |
| CPOL | clock polarity |
| CMOS | complementary metal oxide semiconductor |
| CRC | cyclic redundancy check |
| EEPROM | electrically erasable programmable read-only memory |
| EIA | Electronic Industries Alliance |
| I/O | input/output |
| JEDEC | Joint Electron Devices Engineering Council |
| LSB | least significant bit |
| MSB | most significant bit |
| nvSRAM | nonvolatile static random access memory |
| RWI | read and write inhibit |
| RoHS | restriction of hazardous substances |
| SNL | serial number lock |
| SPI | serial peripheral interface |
| SONOS | silicon-oxide-nitride-oxide semiconductor |
| SOIC | small outline integrated circuit |
| SRAM | static random access memory |

## Document Conventions

### Units of Measure

| Symbol | Unit of Measure |
|---|---|
| °C | degree Celsius |
| Hz | hertz |
| kHz | kilohertz |
| kΩ | kilohm |
| Mbit | megabit |
| MHz | megahertz |
| µA | microampere |
| µF | microfarad |
| µs | microsecond |
| mA | milliampere |
| ms | millisecond |
| ns | nanosecond |
| Ω | ohm |
| % | percent |
| pF | picofarad |
| V | volt |
| W | watt |

## Document History Page

| Document Title: CY14V101QS, 1-Mbit (128K × 8) Quad SPI nvSRAM<br>Document Number: 001-85257 | | | | |
|---|---|---|---|---|
| **Rev.** | **ECN No.** | **Orig. of Change** | **Submission Date** | **Description of Change** |
| *I | 5003596 | SZZX | 11/05/2015 | Release to web |
| *J | 5081889 | JLTO | 01/18/2016 | Changed status from Preliminary to Final.<br>Updated Functional Overview, Pin Definitions, Device Operation, STORE Operation, Hardware RECALL (Power-Up), Read Instructions, DC Specifications, and AC Switching Characteristics.<br>Updated Figure 6 through Figure 92, and Figure 96 through Figure 99.<br>Updated Table 1 and Table 2.<br>Updated $t_{DELAY}$ description in AutoStore or Power-Up RECALL table.<br>Added Figure 101. |
| *K | 5209171 | ZSK | 04/06/2016 | Replaced FPGA with FBGA in all instances across the document.<br>Updated Functional Overview (Updated description).<br>Updated to new template.<br>Completing Sunset Review. |
| *L | 5461974 | MEDU | 10/04/2016 | Updated SPI Memory Read Instructions:<br>Updated Write Instructions:<br>Updated Execute-In-Place (XIP):<br>Updated description.<br>Updated to new template. |
| *M | 5727920 | HARA | 05/05/2017 | Updated logo and copyright. |

## Sales, Solutions, and Legal Information

### Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

### Products

| | |
|---|---|
| ARM® Cortex® Microcontrollers | cypress.com/arm |
| Automotive | cypress.com/automotive |
| Clocks & Buffers | cypress.com/clocks |
| Interface | cypress.com/interface |
| Internet of Things | cypress.com/iot |
| Lighting & Power Control | cypress.com/powerpsoc |
| Memory | cypress.com/memory |
| PSoC | cypress.com/psoc |
| Touch Sensing | cypress.com/touch |
| USB Controllers | cypress.com/usb |
| Wireless/RF | cypress.com/wireless |

### PSoC®Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

### Cypress Developer Community

Forums | Projects | Video | Blogs | Training | Components

### Technical Support

cypress.com/support

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View CY14V101QS-BK108XQ on WIN SOURCE
- Infineon Technologies Information

## Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management