M25P20

2 Mbit, Low Voltage, Serial Flash Memory With 25 MHz SPI Bus Interface

PRELIMINARY DATA

FEATURES SUMMARY

- 2 Mbit of Flash Memory
- Page Program (up to 256 Bytes) in 2 ms (typical)
- Sector Erase (512 Kbit) in 2 s (typical)
- Bulk Erase (2 Mbit) in 3 s (typical)
- 2.7 V to 3.6 V Single Supply Voltage
- SPI Bus Compatible Serial Interface
- 25 MHz Clock Rate (maximum)
- Deep Power-down Mode 1 µA (typical)
- Electronic Signature (11h)
- More than 100,000 Erase/Program Cycles per Sector
- More than 20 Year Data Retention

Figure 1. Packages

SO8 (MN)
150 mil width

SUMMARY DESCRIPTION

The M25P20 is a 2 Mbit (256K x 8) Serial Flash Memory, with advanced write protection mechanisms, accessed by a high speed SPI-compatible bus.

The memory can be programmed 1 to 256 bytes at a time, using the Page Program instruction.

The memory is organized as 4 sectors, each containing 256 pages. Each page is 256 bytes wide. Thus, the whole memory can be viewed as consisting of 1024 pages, or 262,144 bytes.

The whole memory can be erased using the Bulk Erase instruction, or a sector at a time, using the Sector Erase instruction.

Figure 2. Logic Diagram

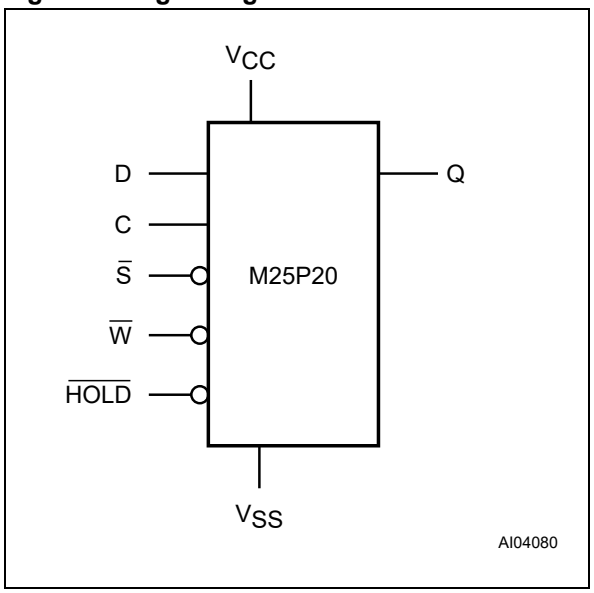
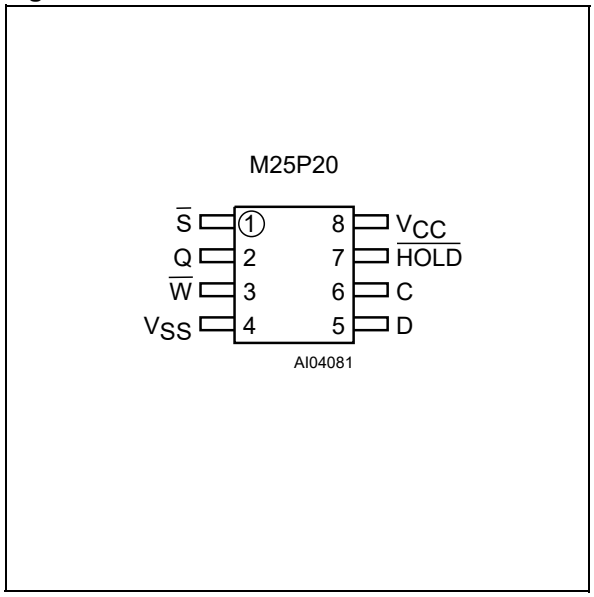


Table 1. Signal Names

C	Serial Clock
D	Serial Data Input
Q	Serial Data Output
$\overline{S}$	Chip Select
$\overline{W}$	Write Protect
$\overline{HOLD}$	Hold
V_{CC}	Supply Voltage
V_{SS}	Ground

Figure 3. SO Connections



SIGNAL DESCRIPTION

Serial Data Output (Q). This output signal is used to transfer data serially out of the device. Data is shifted out on the falling edge of Serial Clock (C).

Serial Data Input (D). This input signal is used to transfer data serially into the device. It receives instructions, addresses, and the data to be programmed. Values are latched on the rising edge of Serial Clock (C).

Serial Clock (C). This input signal provides the timing of the serial interface. Instructions, addresses, or data present at Serial Data Input (D) are latched on the rising edge of Serial Clock (C). Data on Serial Data Output (Q) changes after the falling edge of Serial Clock (C).

Chip Select ($\overline{S}$). When this input signal is High, the device is deselected and Serial Data Output (Q) is at high impedance. Unless an internal Program, Erase or Write Status Register cycle is in progress, the device will be in the Standby mode (this is not the Deep Power-down mode). Driving Chip Select ($\overline{S}$) Low enables the device, placing it in the active power mode.

After Power-up, a falling edge on Chip Select ($\overline{S}$) is required prior to the start of any instruction.

Hold ($\overline{HOLD}$). The Hold ($\overline{HOLD}$) signal is used to pause any serial communications with the device without deselecting the device.

During the Hold condition, the Serial Data Output (Q) is high impedance, and Serial Data Input (D) and Serial Clock (C) are Don't Care.

To start the Hold condition, the device must be selected, with Chip Select ($\overline{S}$) driven Low.

Write Protect ($\overline{W}$). The main purpose of this input signal is to freeze the size of the area of memory that is protected against program or erase instructions (as specified by the values in the BP1 and BP0 bits of the Status Register).

SPI MODES

These devices can be driven by a microcontroller with its SPI peripheral running in either of the two following modes:

– CPOL=0, CPHA=0

– CPOL=1, CPHA=1

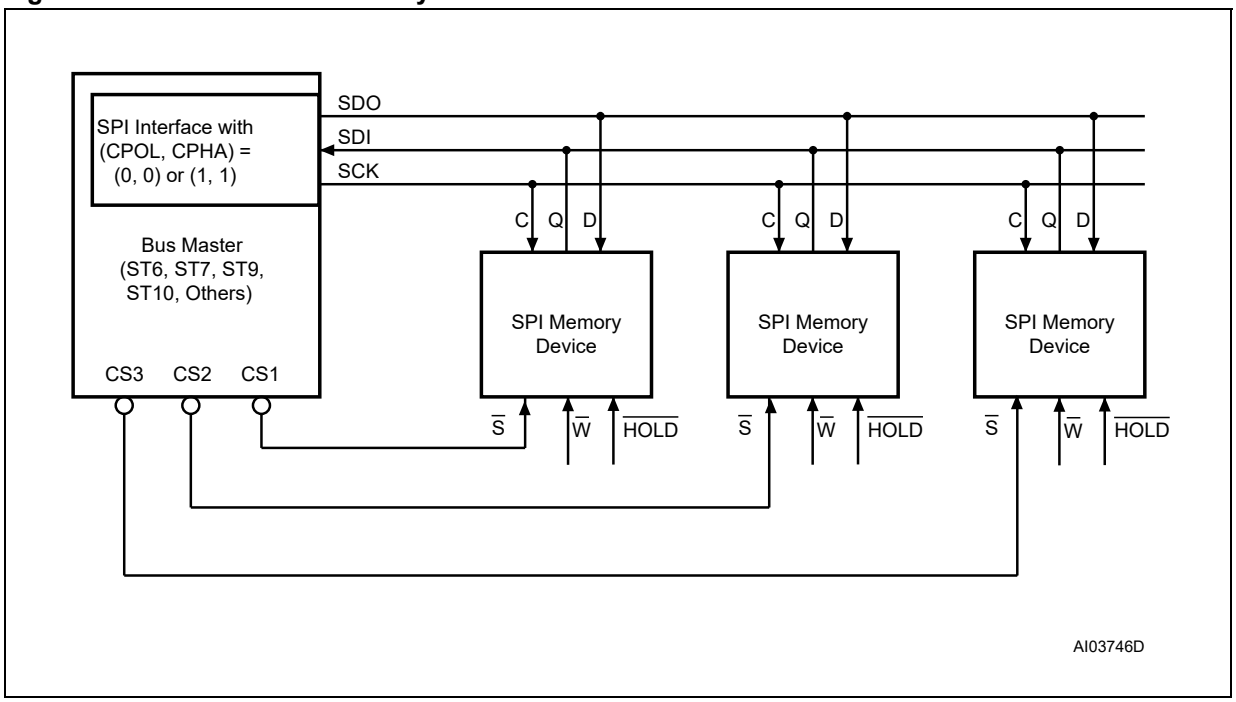
For these two modes, input data is latched in on the rising edge of Serial Clock (C), and output data is available from the falling edge of Serial Clock (C).

The difference between the two modes, as shown in Figure 5, is the clock polarity when the bus master is in Stand-by mode and not transferring data:

– C remains at 0 for (CPOL=0, CPHA=0)

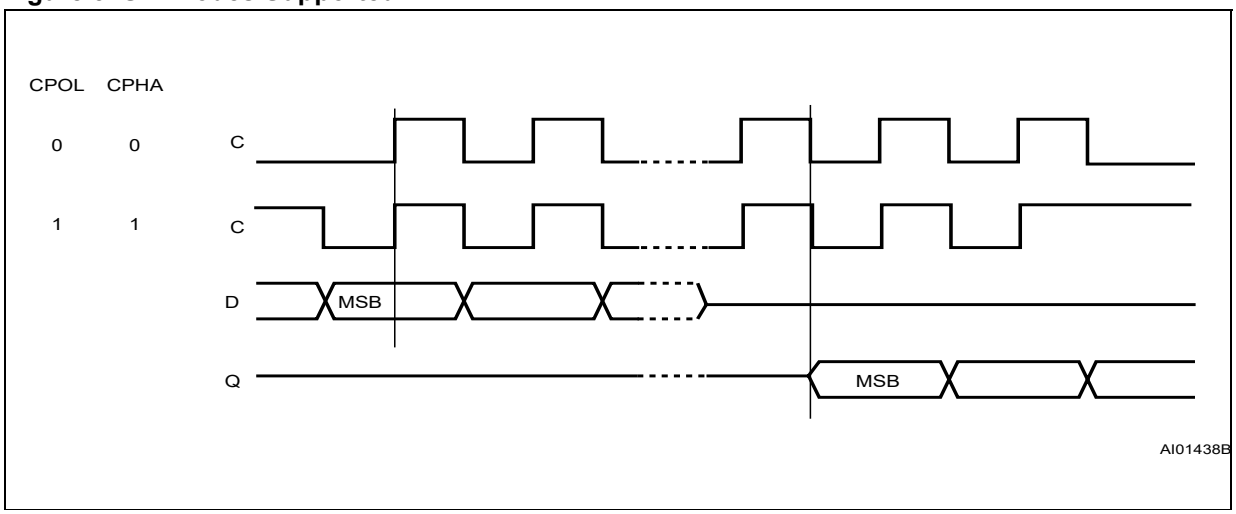
– C remains at 1 for (CPOL=1, CPHA=1)

Figure 4. Bus Master and Memory Devices on the SPI Bus



Note: 1. The Write Protect ($\overline{W}$) and Hold ($\overline{HOLD}$) signals should be driven, High or Low as appropriate.

Figure 5. SPI Modes Supported



OPERATING FEATURES

Page Programming

To program one data byte, two instructions are required: Write Enable (WREN), which is one byte, and a Page Program (PP) sequence, which consists of four bytes plus data. This is followed by the internal Program cycle (of duration t_{PP}).

To spread this overhead, the Page Program (PP) instruction allows up to 256 bytes to be programmed at a time (changing bits from 1 to 0), provided that they lie in consecutive addresses on the same page of memory.

Sector Erase and Bulk Erase

The Page Program (PP) instruction allows bits to be reset from 1 to 0. Before this can be applied, the bytes of memory need to have been erased to all 1s (FFh). This can be achieved either a sector at a time, using the Sector Erase (SE) instruction, or throughout the entire memory, using the Bulk Erase (BE) instruction.

Polling During a Write, Program or Erase Cycle

A further improvement in the time to Write Status Register (WRSR), Program (PP) or Erase (SE or BE) can be achieved by not waiting for the worst case delay (t_W, t_{PP}, t_{SE}, or t_{BE}). The Write In Progress (WIP) bit is provided in the Status Register so that the application program can monitor its value, polling it to establish when the previous Write cycle, Program cycle or Erase cycle is complete.

Active Power, Stand-by Power and Deep Power-Down Modes

When Chip Select ($\overline{S}$) is Low, the device is enabled, and in the Active Power mode.

When Chip Select ($\overline{S}$) is High, the device is disabled, but could remain in the Active Power mode until all internal cycles have completed (Program, Erase, Write Status Register). The device then goes in to the Stand-by Power mode. The device consumption drops to I_{CC1}.

The Deep Power-down mode is entered when the specific instruction (the Enter Deep Power-down Mode (DP) instruction) is executed. The device consumption drops further to I_{CC2}. The device remains in this mode until another specific instruction (the Release from Deep Power-down Mode and Read Electronic Signature (RES) instruction) is executed.

All other instructions are ignored while the device is in the Deep Power-down mode. This can be used as an extra software protection mechanism, when the device is not in active use, to protect the device from inadvertant Write, Program or Erase instructions.

Status Register

The Status Register contains a number of status and control bits, as shown in Table 5, that can be read or set (as appropriate) by specific instructions.

WIP bit. The Write In Progress (WIP) bit indicates whether the memory is busy with a Write Status Register, Program or Erase cycle.

WEL bit. The Write Enable Latch (WEL) bit indicates the status of the internal Write Enable Latch.

BP1, BP0 bits. The Block Protect (BP1, BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase instructions.

SRWD bit. The Status Register Write Disable (SRWD) bit is operated in conjunction with the Write Protect ($\overline{W}$) signal. The Status Register Write Disable (SRWD) bit and Write Protect ($\overline{W}$) signal allow the device to be put in the Hardware Protected mode. In this mode, the non-volatile bits of the Status Register (SRWD, BP1, BP0) become read-only bits.

Protection Modes

The environments where non-volatile memory devices are used can be very noisy. No SPI device can operate correctly in the presence of excessive noise. To help combat this, the M25P20 boasts the following data protection mechanisms:

- Power-On Reset and an internal timer (t_{PUW}) can provide protection against inadvertant changes while the power supply is outside the operating specification.
- Program, Erase and Write Status Register instructions are checked that they consist of a number of clock pulses that is a multiple of eight, before they are accepted for execution.
- All instructions that modify data must be preceded by a Write Enable (WREN) instruction to set the Write Enable Latch (WEL) bit . This bit is returned to its reset state by the following events:
  - Power-up
  - Write Disable (WRDI) instruction completion
  - Write Status Register (WRSR) instruction completion
  - Page Program (PP) instruction completion
  - Sector Erase (SE) instruction completion
  - Bulk Erase (BE) instruction completion
- The Block Protect (BP1, BP0) bits allow part of the memory to be configured as read-only. This is the Software Protected Mode (SPM).
- The Write Protect ($\overline{W}$) signal, in co-operation with the Status Register Write Disable (SRWD) bit, allows the Block Protect (BP1, BP0) bits and Status Register Write Disable (SRWD) bit to be write-protected. This is the Hardware Protected Mode (HPM).
- In addition to the low power consumption feature, the Deep Power-down mode offers extra software protection from inadvertant Write, Program and Erase instructions, as all instructions are ignored except one particular instruction (the Release from Deep Power-down instruction).

Table 2. Protected Area Sizes

Status Register Content		Memory Content	
BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
0	0	none	All sectors[1] (four sectors: 0, 1, 2 and 3)
0	1	Upper quarter (Sector 3)	Lower three-quarters (three sectors: 0 to 2)
1	0	Upper half (two sectors: 2 and 3)	Lower half (Sectors 0 and 1)
1	1	All sectors (four sectors: 0, 1, 2 and 3)	none

Note: 1. The device is ready to accept a Bulk Erase instruction if, and only if, both Block Protect (BP1, BP0) are 0.

Hold Condition

The Hold ($\overline{HOLD}$) signal is used to pause any serial communications with the device without resetting the clocking sequence. However, taking this signal Low does not terminate any Write Status Register, Program or Erase cycle that is currently in progress.

To enter the Hold condition, the device must be selected, with Chip Select ($\overline{S}$) Low.

The Hold condition starts on the falling edge of the Hold ($\overline{HOLD}$) signal, provided that this coincides with Serial Clock (C) being Low (as shown in Figure 6).

The Hold condition ends on the rising edge of the Hold ($\overline{HOLD}$) signal, provided that this coincides with Serial Clock (C) being Low.

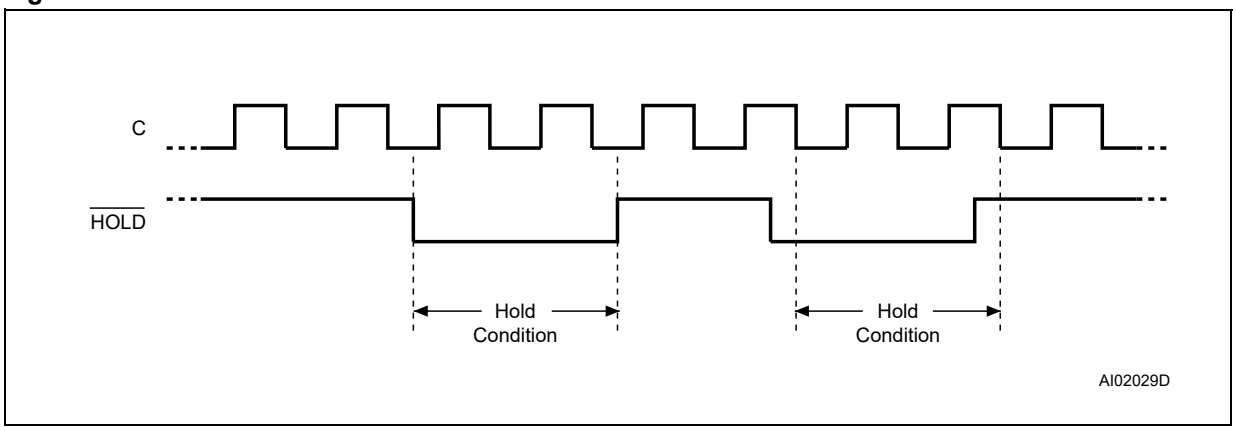
If the falling edge does not coincide with Serial Clock (C) being Low, the Hold condition starts when Serial Clock (C) next goes Low. Similarly, if the rising edge does not coincide with Serial Clock (C) being Low, the Hold condition ends when Serial Clock (C) next goes Low. (This is shown in Figure 6).

During the Hold condition, the Serial Data Output (Q) is high impedance, and Serial Data Input (D) and Serial Clock (C) are Don't Care.

Normally, the device is kept selected, with Chip Select ($\overline{S}$) driven Low, for the whole duration of the Hold condition. This is to ensure that the state of the internal logic remains unchanged from the moment of entering the Hold condition.

If Chip Select ($\overline{S}$) goes High while the device is in the Hold condition, this has the effect of resetting the internal logic of the device. To restart communication with the device, it is necessary to drive Hold ($\overline{HOLD}$) High, and then to drive Chip Select ($\overline{S}$) Low. This prevents the device from going back to the Hold condition.

Figure 6. Hold Condition Activation



MEMORY ORGANIZATION

The memory is organized as:

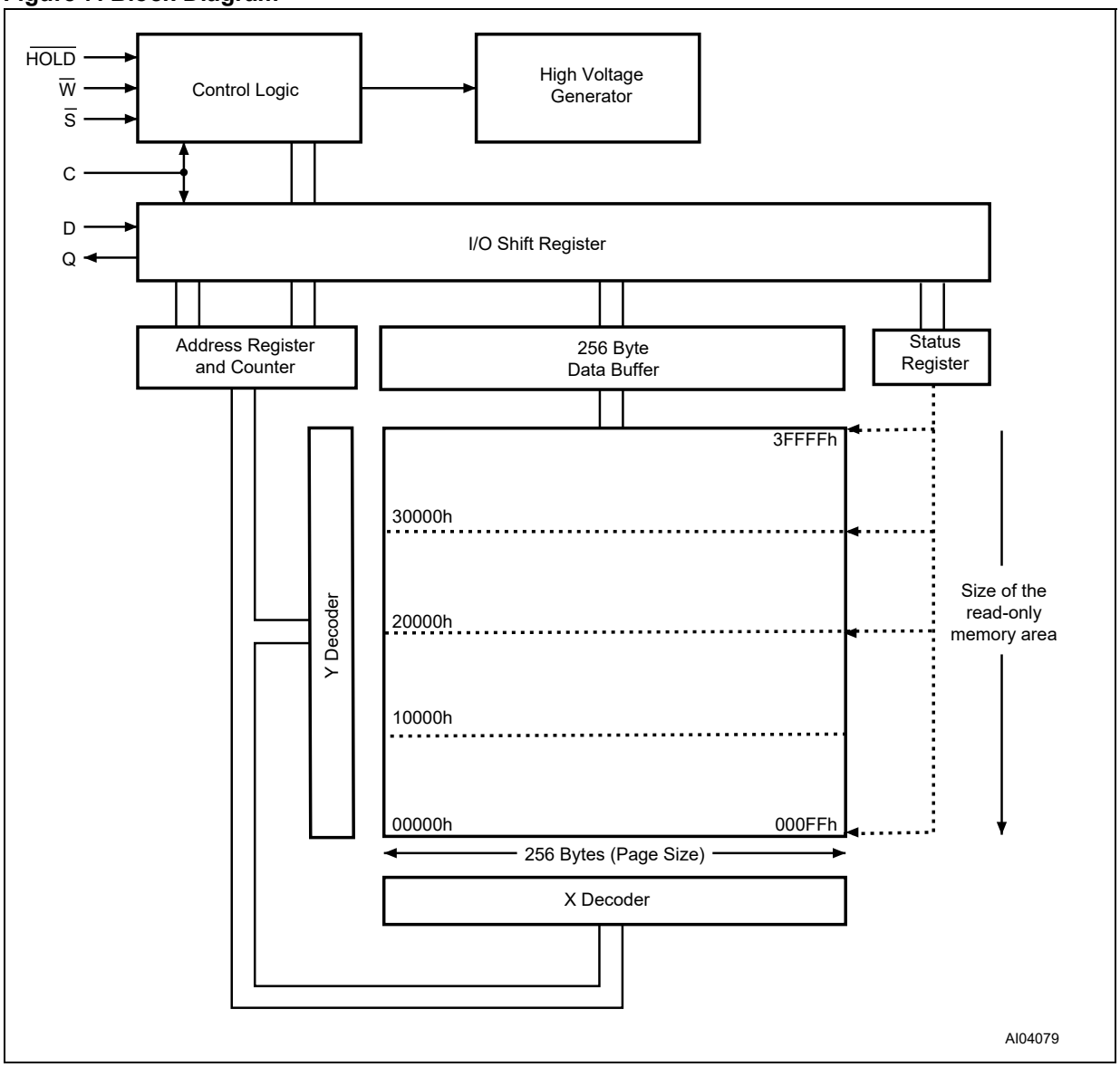
- 262,144 bytes (8 bits each)
- 4 sectors (512 Kbits, 65536 bytes each)
- 1024 pages (256 bytes each).

Each page can be individually programmed (bits are programmed from 1 to 0). The device is Sector or Bulk Erasable (bits are erased from 0 to 1) but not Page Erasable.

Table 3. Memory Organization

Sector	Address Range	
3	30000h	3FFFFh
2	20000h	2FFFFh
1	10000h	1FFFFh
0	00000h	0FFFFh

Figure 7. Block Diagram



INSTRUCTIONS

All instructions, addresses and data are shifted in and out of the device, most significant bit first.

Serial Data Input (D) is sampled on the first rising edge of Serial Clock (C) after Chip Select ($\overline{S}$) is driven Low. Then, the one-byte instruction code must be shifted in to the device, most significant bit first, on Serial Data Input (D), each bit being latched on the rising edges of Serial Clock (C).

The instruction set is listed in Table 4.

Depending on the instruction, the one-byte instruction code is followed by address bytes, or by data bytes, or by both or none. Chip Select ($\overline{S}$) must be driven High after the last bit of the instruction sequence has been shifted in.

At the end of a Page Program (PP), Sector Erase (SE), Bulk Erase (BE) or Write Status Register (WRSR) instruction, Chip Select ($\overline{S}$) must be driven High exactly at a byte boundary, otherwise the instruction is rejected, and is not executed. That is, Chip Select ($\overline{S}$) must driven High when the number of clock pulses after Chip Select ($\overline{S}$) being driven Low is an exact multiple of eight.

All attempts to access the memory array during a Write Status Register cycle, Program cycle or Erase cycle are ignored, and the internal Write Status Register cycle, Program cycle or Erase cycle continues unaffected.

Table 4. Instruction Set

Instruction	Description	One-byte Instruction Code	Address Bytes	Dummy Bytes	Data Bytes
WREN	Write Enable	0000 0110	0	0	0
WRDI	Write Disable	0000 0100	0	0	0
RDSR	Read Status Register	0000 0101	0	0	1 to ∞
WRSR	Write Status Register	0000 0001	0	0	1
READ	Read Data Bytes	0000 0011	3	0	1 to ∞
FAST_READ	Read Data Bytes at Higher Speed	0000 1011	3	1	1 to ∞
PP	Page Program	0000 0010	3	0	1 to 256
SE	Sector Erase	1101 1000	3	0	0
BE	Bulk Erase	1100 0111	0	0	0
DP	Deep Power-down	1011 1001	0	0	0
RES	Release from Deep Power-down, and Read Electronic Signature	1010 1011	0	3	1 to ∞
	Release from Deep Power-down		0	0	0