Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

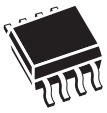
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

Numonyx® Forté™ Serial Flash Memory M25P40

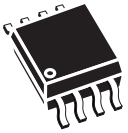
4 Mbit, low voltage, serial Flash memory with 75 MHz SPI bus interface

Features

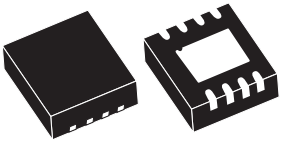
- 4 Mbit of Flash memory
- 2.3 V to 3.6 V single supply voltage
- SPI bus compatible serial interface
- 75 MHz clock rate (maximum)
- Page Program (up to 256 bytes) in 0.8 ms (typical)
- Sector Erase (512 Kbit) in 0.6 s (typical)
- Bulk Erase (4 Mbit) in 4.5 s (typical)
- Deep Power-down mode 1 µA (typical)
- Hardware Write Protection: protected area size defined by three non-volatile bits (BP0, BP1 and BP2)
- Electronic signatures
  - JEDEC standard two-byte signature (2013h)
  - Unique ID code (UID) with 16 bytes read-only, available upon customer request
  - RES instruction, one-byte, signature (12h), for backward compatibility
- Packages
  - RoHS compliant
- Automotive grade parts available



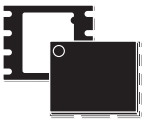
SO8 (MN)
150 mils width



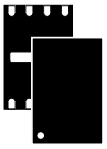
SO8W (MW)
208 mils width



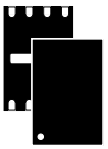
DFN8 (MS)
MLP8 6 x 5 mm



VFDFPN8 (MP)
(MLP8 6 x 5 mm)



UFDFPN8 (MB)
(MLP8 2 x 3 mm)



UFDFPN8 (MC)
(MLP8 4 x 3 mm)

Contents

List of tables

List of figures

1 Description

The M25P40 is a 4 Mbit (512 K × 8) Serial Flash memory, with advanced write protection mechanisms, accessed by a high speed SPI-compatible bus. The M25P40 features high performance instructions allowing clock frequency up to 75 MHz.(1)

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

The memory is organized as 8 sectors, each containing 256 pages. Each page is 256 bytes wide. Thus, the whole memory can be viewed as consisting of 2048 pages, or 524,288 bytes.

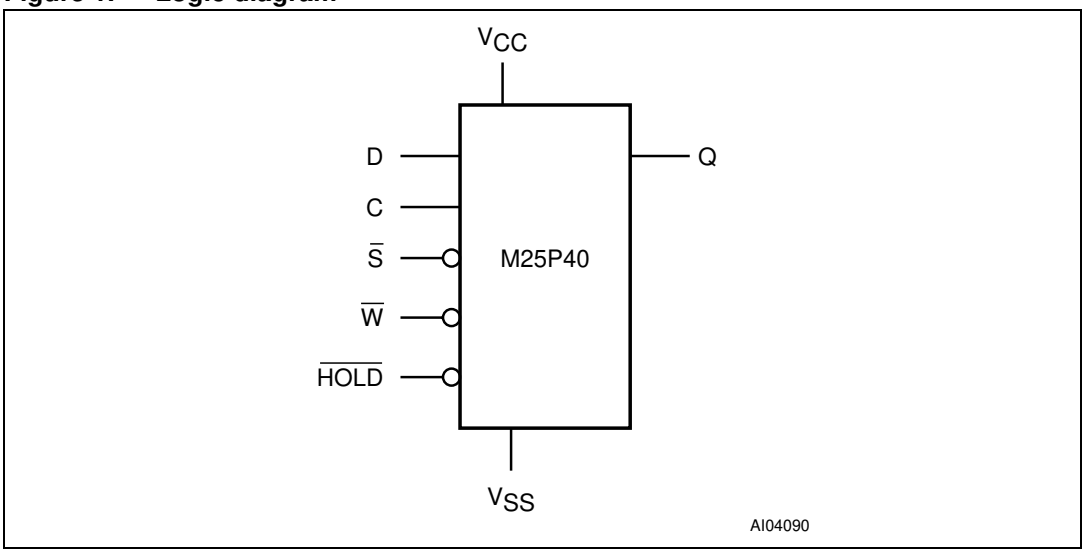
The whole memory can be erased using the Bulk Erase instruction, or a sector at a time, using the Sector Erase instruction.

In order to meet environmental requirements, Numonyx offers the M25P40 in RoHS compliant packages, which are Lead-free. RoHS specifications are available at: *www.Numonyx.com*.

Important: This datasheet details the functionality of the M25P40 devices, based on the previous 150 nm process or based on the current 110 nm process (available since August 2008). The new device in the 110 nm process has the following additional features and is completely backward compatible with the old one in 150 nm:

- *improved max frequency (Fast Read) to 75 MHz in the standard Vcc range 2.7 V to 3.6 V, while the max frequency (Fast Read) in the extended Vcc range 2.3 V to 2.7 V is 40 MHz*
- *UID/CFD protection feature*

Figure 1. Logic diagram

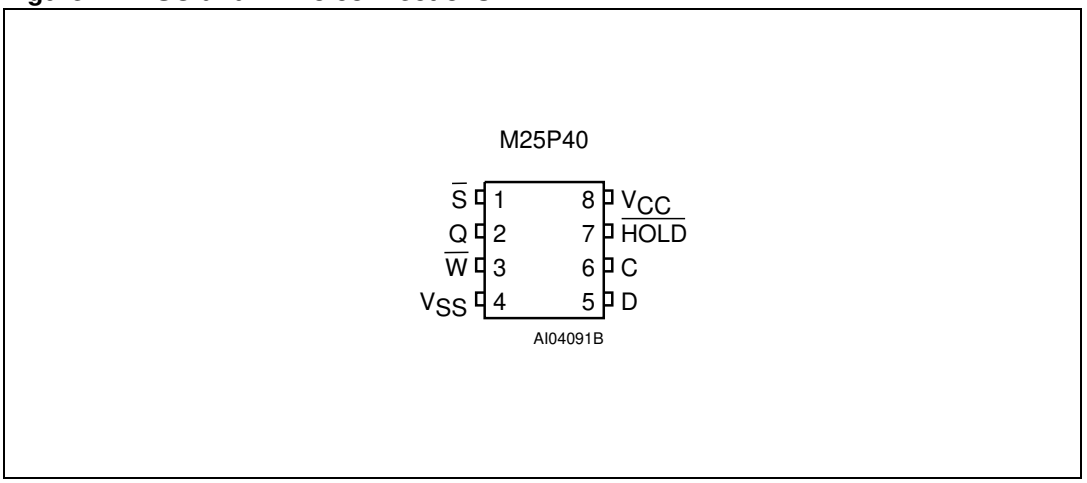


1. 75 MHz operation is available only on the VCC range 2.7 V - 3.6 V and for 110 nm process technology devices, identified by process identification digit "4" in the device marking and process letter "B" in the part number.

Table 1. Signal names

Signal name	Function	Direction
C	Serial Clock	Input
D	Serial Data input	Input
Q	Serial Data output	Output
$\overline{S}$	Chip Select	Input
$\overline{W}$	Write Protect	Input
$\overline{HOLD}$	Hold	Input
V_{CC}	Supply voltage	—
V_{SS}	Ground	—

Figure 2. SO and MLP8 connections



1. There is an exposed central pad on the underside of the MLP8 packages. This is pulled, internally, to V_{SS}, and must not be allowed to be connected to any other voltage or signal line on the PCB.
2. See *Section 11: Package mechanical* for package dimensions, and how to identify pin-1.

2 Signal description

2.1 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).

2.2 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).

2.3 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).

2.4 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 Power mode (this is not the Deep Power-down mode). Driving Chip Select ($\overline{S}$) Low selects 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.

2.5 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.

2.6 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 BP2, BP1 and BP0 bits of the Status Register).

2.7 V_{CC} supply voltage

V_{CC} is the supply voltage.

2.8 V_{SS} ground

V_{SS} is the reference for the V_{CC} supply voltage.

3 SPI modes

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

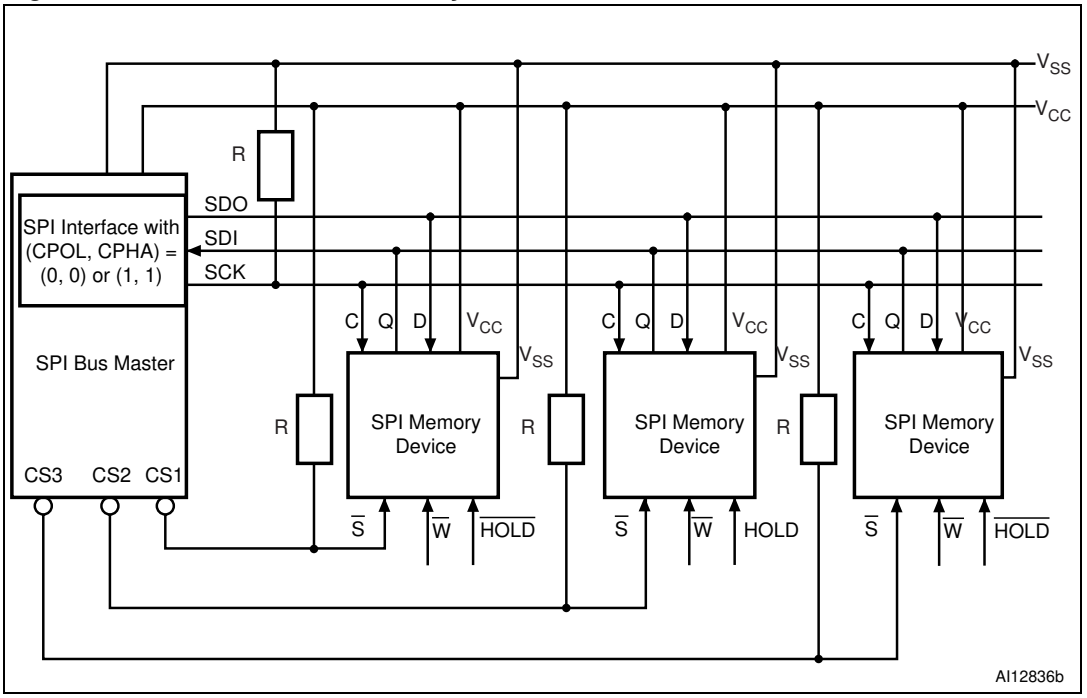
- CPOL=0, CPHA=0
- CPOL=1, CPHA=1

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 4*, 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 3. Bus Master and memory devices on the SPI bus

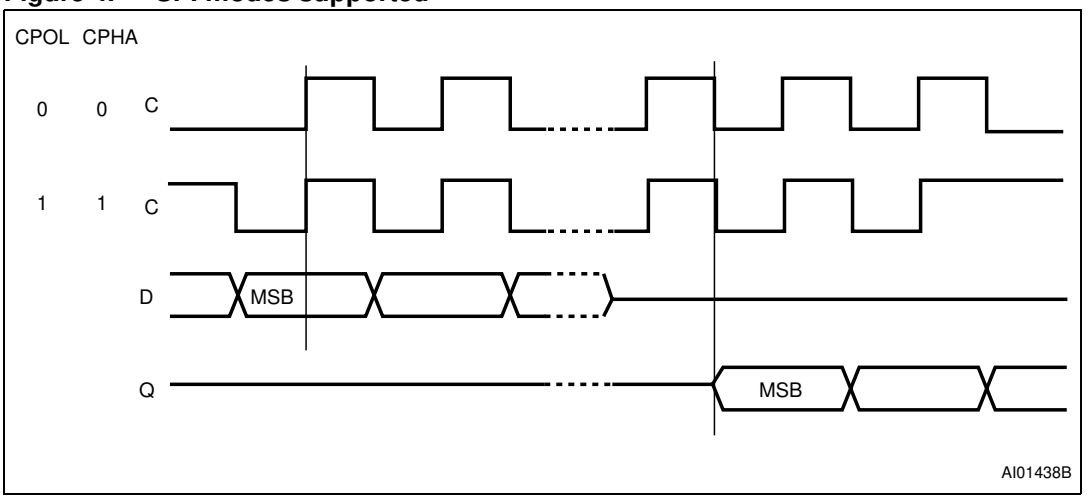


1. The Write Protect ($\overline{W}$) and Hold ($\overline{HOLD}$) signals should be driven, High or Low as appropriate.

Figure 3: Bus Master and memory devices on the SPI bus shows an example of three devices connected to an MCU, on an SPI bus. Only one device is selected at a time, so only one device drives the Serial Data output (Q) line at a time, the other devices are high impedance. Resistors R (represented in *Figure 3*) ensure that the M25P40 is not selected if the Bus Master leaves the $\overline{S}$ line in the high impedance state. As the Bus Master may enter a state where all inputs/outputs are in high impedance at the same time (for example, when the Bus Master is reset), the clock line (C) must be connected to an external pull-down resistor so that, when all inputs/outputs become high impedance, the $\overline{S}$ line is pulled High while the C line is pulled Low (thus ensuring that $\overline{S}$ and C do not become High at the same time, and so, that the t_{SHCH} requirement is met). The typical value of R is 100K Ω, assuming that the time constant $R*C_p$ (C_p = parasitic capacitance of the bus line) is shorter than the time during which the Bus Master leaves the SPI bus in high impedance.

Example: C_p = 50 pF, that is $R*C_p$ = 5 µs: the application must ensure that the Bus Master never leaves the SPI bus in the high impedance state for a time period shorter than 5 µs.

Figure 4. SPI modes supported



4 Operating features

4.1 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.

For optimized timings, it is recommended to use the Page Program (PP) instruction to program all consecutive targeted bytes in a single sequence versus using several Page Program (PP) sequences with each containing only a few bytes (See *Page Program (PP)*, *Instruction times, process technology 110 nm*.)

4.2 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. This starts an internal Erase cycle (of duration t_{SE} or t_{BE}).

The Erase instruction must be preceded by a Write Enable (WREN) instruction.

4.3 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.

4.4 Active Power, Standby Power and Deep Power-down modes

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

When Chip Select ($\overline{S}$) is High, the device is deselected, 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 Standby Power mode. The device consumption drops to I_{CC1}.

The Deep Power-down mode is entered when the specific instruction (the Deep Power-down (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 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 inadvertent Write, Program or Erase instructions.

4.5 Status Register

The Status Register contains a number of status and control bits that can be read or set (as appropriate) by specific instructions. For a detailed description of the Status Register bits, see *Section 6.4: Read Status Register (RDSR)*.

4.6 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 M25P40 features the following data protection mechanisms:

- Power On Reset and an internal timer (t_{PUW}) can provide protection against inadvertent 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 after 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
- Software Protected Mode (SPM): The Block Protect (BP2, BP1, BP0) bits allow part of the memory to be configured as read-only.
- Hardware Protected Mode (HPM): The Write Protect ($\overline{W}$) signal allows the Block Protect (BP2, BP1, BP0) bits and Status Register Write Disable (SRWD) bit to be protected.

In addition to the low power consumption feature, the Deep Power-down mode offers extra software protection from inadvertent Write, Program, and Erase instructions, as all instructions are ignored except the Release from Deep Power-down instruction.

Table 2. Protected area sizes

Status Register content			Memory content	
BP2 bit	BP1 bit	BP0 bit	Protected area	Unprotected area
0	0	0	none	All sectors(1) (eight sectors: 0 to 7)
0	0	1	Upper eighth (Sector 7)	Lower seven-eighths (seven sectors: 0 to 6)
0	1	0	Upper quarter (two sectors: 6 and 7)	Lower three-quarters (six sectors: 0 to 5)
0	1	1	Upper half (four sectors: 4 to 7)	Lower half (four sectors: 0 to 3)
1	0	0	All sectors (eight sectors: 0 to 7)	none
1	0	1	All sectors (eight sectors: 0 to 7)	none
1	1	0	All sectors (eight sectors: 0 to 7)	none
1	1	1	All sectors (eight sectors: 0 to 7)	none

1. The device is ready to accept a Bulk Erase instruction only if all Block Protect bits (BP2, BP1, BP0) are 0.

4.7 Hold condition

The Hold ($\overline{\text{HOLD}}$) 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 ($\overline{\text{S}}$) Low.

The Hold condition starts on the falling edge of the Hold ($\overline{\text{HOLD}}$) signal, provided that this coincides with Serial Clock (C) being Low (as shown in *Figure 5: Hold condition activation*).

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

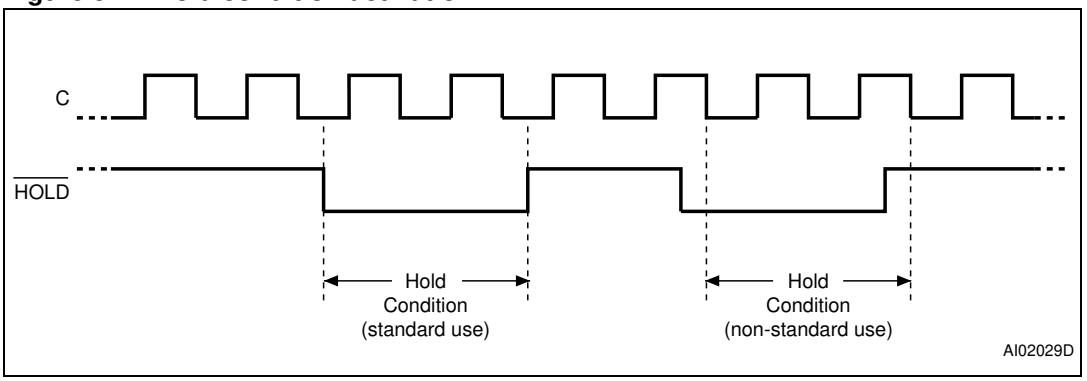
If the falling edge does not coincide with Serial Clock (C) being Low, the Hold condition starts after Serial Clock (C) next goes Low. Similarly, if the rising edge does not coincide with Serial Clock (C) being Low, the Hold condition ends after Serial Clock (C) next goes Low. (This is shown in *Figure 5: Hold condition activation*).

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 ($\overline{\text{S}}$) 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 ($\overline{\text{S}}$) 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 ($\overline{\text{HOLD}}$) High, and then to drive Chip Select ($\overline{\text{S}}$) Low. This prevents the device from going back to the Hold condition.

Figure 5. Hold condition activation



5 Memory organization

The memory is organized as:

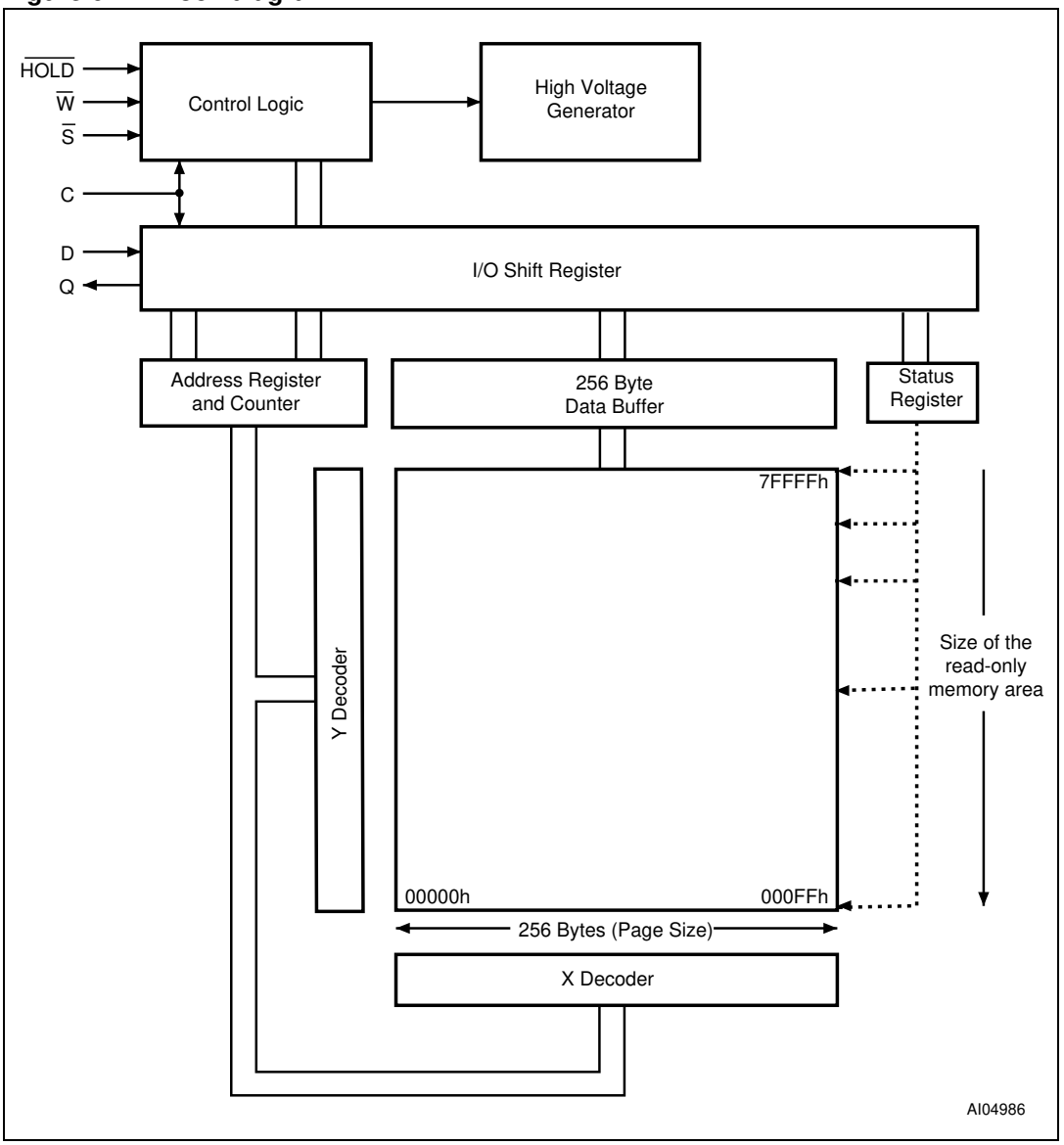
- 524,288 bytes (8 bits each)
- 8 sectors (512 Kbits, 65,536 bytes each)
- 2048 pages (256 bytes each)

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	
7	70000h	7FFFFh
6	60000h	6FFFFh
5	50000h	5FFFFh
4	40000h	4FFFFh
3	30000h	3FFFFh
2	20000h	2FFFFh
1	10000h	1FFFFh
0	00000h	0FFFFh

Figure 6. Block diagram



6 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*.

Every instruction sequence starts with a one-byte instruction code. Depending on the instruction, this might be 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.

In the case of a Read Data Bytes (READ), Read Data Bytes at Higher Speed (Fast_Read), Read Identification (RDID), Read Status Register (RDSR) or Release from Deep Power-down, and Read Electronic Signature (RES) instruction, the shifted-in instruction sequence is followed by a data-out sequence. Chip Select ($\overline{S}$) can be driven High after any bit of the data-out sequence is being shifted out.

In the case of a Page Program (PP), Sector Erase (SE), Bulk Erase (BE), Write Status Register (WRSR), Write Enable (WREN), Write Disable (WRDI) or Deep Power-down (DP) 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	06h	0	0	0
WRDI	Write Disable	0000 0100	04h	0	0	0
RDID(1)	Read Identification	1001 1111	9Fh	0	0	1 to 3
RDSR	Read Status Register	0000 0101	05h	0	0	1 to ∞
WRSR	Write Status Register	0000 0001	01h	0	0	1
READ	Read Data Bytes	0000 0011	03h	3	0	1 to ∞
FAST_READ	Read Data Bytes at Higher Speed	0000 1011	0Bh	3	1	1 to ∞
PP	Page Program	0000 0010	02h	3	0	1 to 256
SE	Sector Erase	1101 1000	D8h	3	0	0
BE	Bulk Erase	1100 0111	C7h	0	0	0
DP	Deep Power-down	1011 1001	B9h	0	0	0
RES	Release from Deep Power-down, and Read Electronic Signature	1010 1011	ABh	0	3	1 to ∞
	Release from Deep Power-down			0	0	0

1. The Read Identification (RDID) instruction is available only for parts made with 110 nm Technology identified with Process letter '4'. (Also, see Application Note AN1995).

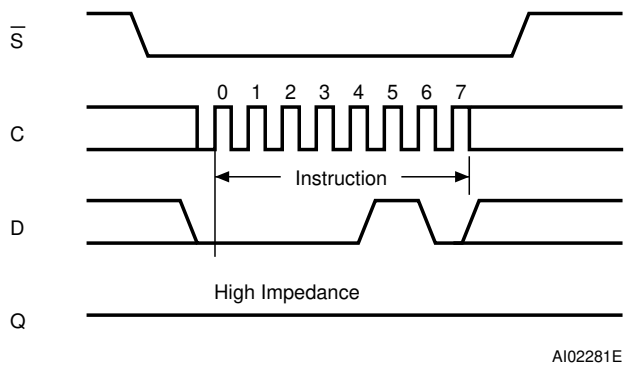
6.1 Write Enable (WREN)

The Write Enable (WREN) instruction (*Figure 7*) sets the Write Enable Latch (WEL) bit.

The Write Enable Latch (WEL) bit must be set prior to every Page Program (PP), Sector Erase (SE), Bulk Erase (BE) and Write Status Register (WRSR) instruction.

The Write Enable (WREN) instruction is entered by driving Chip Select ($\overline{S}$) Low, sending the instruction code, and then driving Chip Select ($\overline{S}$) High.

Figure 7. Write Enable (WREN) instruction sequence



6.2 Write Disable (WRDI)

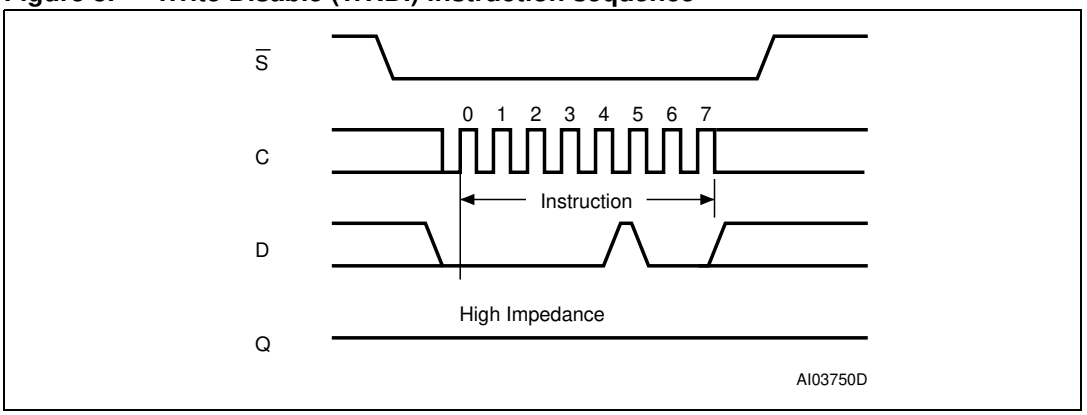
The Write Disable (WRDI) instruction (*Figure 8*) resets the Write Enable Latch (WEL) bit.

The Write Disable (WRDI) instruction is entered by driving Chip Select ($\overline{S}$) Low, sending the instruction code, and then driving Chip Select ($\overline{S}$) High.

The Write Enable Latch (WEL) bit is reset under the following conditions:

- 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

Figure 8. Write Disable (WRDI) instruction sequence



6.3 Read Identification (RDID)

The Read Identification (RDID) instruction reads the device identification data:

- Manufacturer identification (1 byte)
- Device identification (2 bytes)
- Unique ID code (UID) (17 bytes, 16 which are available upon customer request).(2)

The manufacturer identification is assigned by JEDEC, and has the value 20h for Numonyx. The device identification is assigned by the device manufacturer, and indicates the memory type in the first byte (20h), and the memory capacity of the device in the second byte (13h). The UID contains the length of the following data in the first byte (set to 10h), and 16 bytes of the optional Customized Factory Data (CFD) content. The CFD bytes are read-only and can be programmed with customers data upon their demand. If the customers do not make requests, the devices are shipped with all the CFD bytes programmed to zero (00h). Any Read Identification (RDID) instruction while an Erase or Program cycle is in progress, is not decoded, and has no effect on the cycle that is in progress. The device is first selected by driving Chip Select (S) Low. Then, the 8-bit instruction code for the instruction is shifted in. After this, the 24-bit device identification, stored in the memory, the 8-bit CFD length followed by 16 bytes of CFD content will be shifted out on Serial Data output (Q). Each bit is shifted out during the falling edge of Serial Clock (C).

The instruction sequence is shown in *Figure 9: Read Identification (RDID) instruction sequence and data-out sequence*.

The Read Identification (RDID) instruction is terminated by driving Chip Select ($\overline{S}$) High at any time during data output.

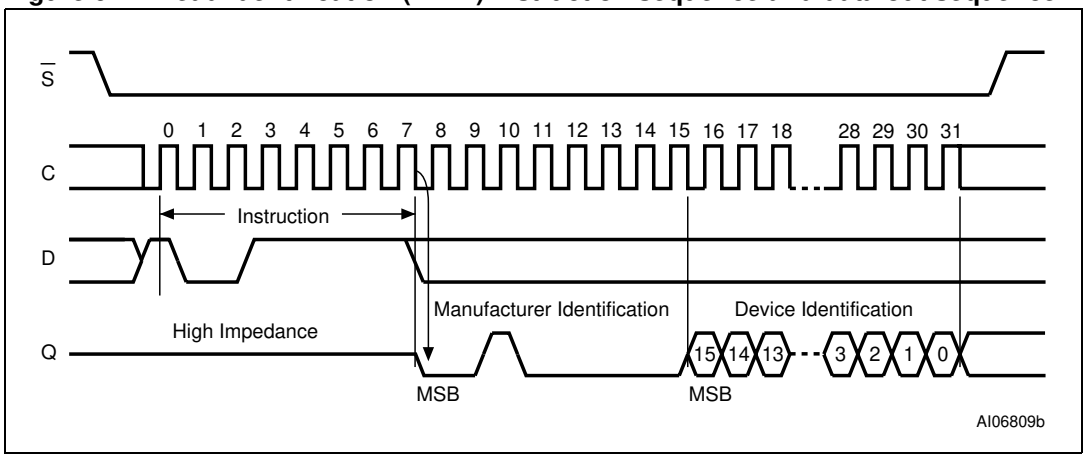
When Chip Select ($\overline{S}$) is driven High, the device is put in the Stand-by Power mode. Once in the Stand-by Power mode, the device waits to be selected, so that it can receive, decode and execute instructions.

Table 5. Read Identification (RDID) data-out sequence

Manufacturer identification	Device identification		UID	
	Memory type	Memory capacity	CFD length	CFD content
20h	20h	13h	10h	16 bytes

2. The UID feature is available only for 110 nm process technology devices, identified by process identification digit "4" in the device marking and process letter "B" in the part number.

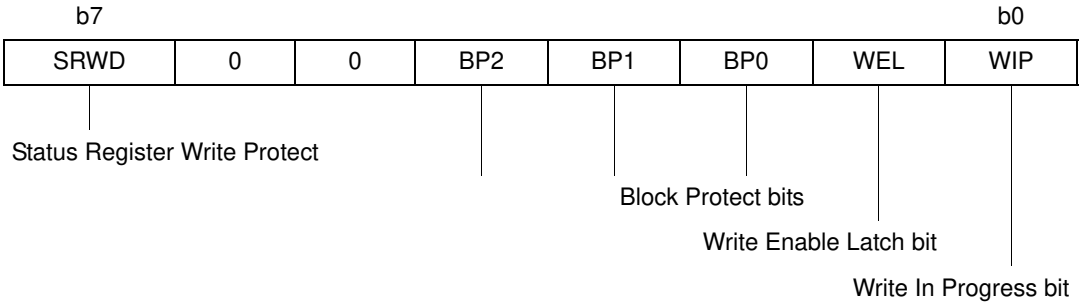
Figure 9. Read Identification (RDID) instruction sequence and data-out sequence



6.4 Read Status Register (RDSR)

The Read Status Register (RDSR) instruction allows the Status Register to be read. The Status Register may be read at any time, even while a Program, Erase or Write Status Register cycle is in progress. When one of these cycles is in progress, it is recommended to check the Write In Progress (WIP) bit before sending a new instruction to the device. It is also possible to read the Status Register continuously, as shown in *Figure 10*.

Table 6. Status Register format



The status and control bits of the Status Register are as follows:

6.4.1 WIP bit

The Write In Progress (WIP) bit indicates whether the memory is busy with a Write Status Register, Program or Erase cycle. When set to 1, such a cycle is in progress, when reset to 0 no such cycle is in progress.

6.4.2 WEL bit

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

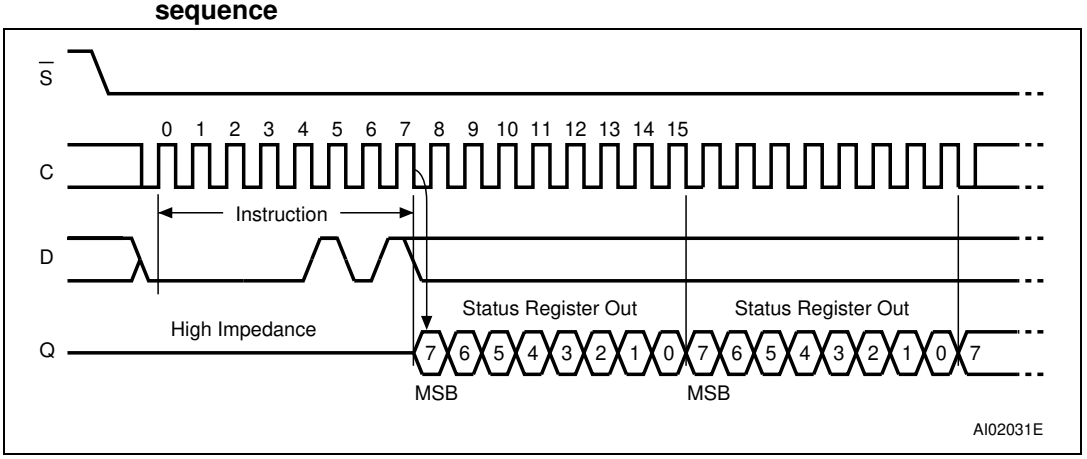
6.4.3 BP2, BP1, BP0 bits

The Block Protect (BP2, BP1, BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase instructions. These bits are written with the Write Status Register (WRSR) instruction. When one or more of the Block Protect (BP2, BP1, BP0) bits is set to 1, the relevant memory area (as defined in *Table 2*) becomes protected against Page Program (PP) and Sector Erase (SE) instructions. The Block Protect (BP2, BP1, BP0) bits can be written provided that the Hardware Protected mode has not been set. The Bulk Erase (BE) instruction is executed if, and only if, all Block Protect (BP2, BP1, BP0) bits are 0.

6.4.4 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 (when the Status Register Write Disable (SRWD) bit is set to 1, and Write Protect ($\overline{W}$) is driven Low). In this mode, the non-volatile bits of the Status Register (SRWD, BP2, BP1, BP0) become read-only bits and the Write Status Register (WRSR) instruction is no longer accepted for execution.

Figure 10. Read Status Register (RDSR) instruction sequence and data-out sequence



6.5 Write Status Register (WRSR)

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

The Write Status Register (WRSR) instruction is entered by driving Chip Select ($\overline{S}$) Low, followed by the instruction code and the data byte on Serial Data input (D).

The instruction sequence is shown in *Figure 11*.

The Write Status Register (WRSR) instruction has no effect on b6, b5, b1 and b0 of the Status Register. b6 and b5 are always read as 0.

Chip Select ($\overline{S}$) must be driven High after the eighth bit of the data byte has been latched in. If not, the Write Status Register (WRSR) instruction is not executed. As soon as Chip Select ($\overline{S}$) is driven High, the self-timed Write Status Register cycle (whose duration is t_W) is initiated. While the Write Status Register cycle is in progress, the Status Register may still be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Write Status Register cycle, and is 0 when it is completed. When the cycle is completed, the Write Enable Latch (WEL) is reset.

The Write Status Register (WRSR) instruction allows the user to change the values of the Block Protect (BP2, BP1, BP0) bits, to define the size of the area that is to be treated as read-only, as defined in *Table 2*. The Write Status Register (WRSR) instruction also allows the user to set or reset the Status Register Write Disable (SRWD) bit in accordance 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 (HPM). The Write Status Register (WRSR) instruction is not executed once the Hardware Protected Mode (HPM) is entered.

Figure 11. Write Status Register (WRSR) instruction sequence

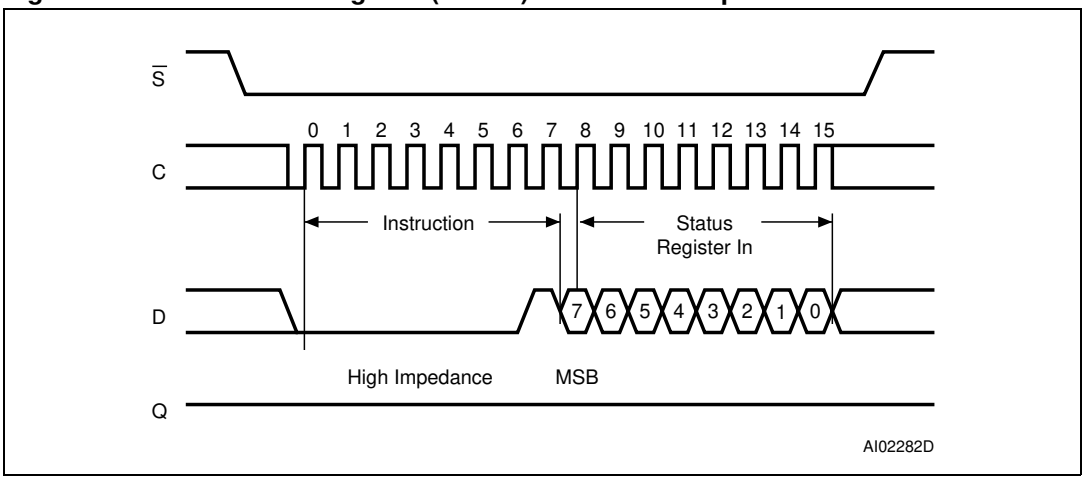


Table 7. Protection modes

$\overline{W}$ signal	SRWD bit	Mode	Write Protection of the Status Register	Memory content	
				Protected area(1)	Unprotected area(1)
1	0	Software Protected (SPM)	Status Register is Writable (if the WREN instruction has set the WEL bit) The values in the SRWD, BP2, BP1 and BP0 bits can be changed	Protected against Page Program, Sector Erase and Bulk Erase	Ready to accept Page Program and Sector Erase instructions
0	0				
1	1				
0	1	Hardware Protected (HPM)	Status Register is Hardware write protected The values in the SRWD, BP2, BP1 and BP0 bits cannot be changed	Protected against Page Program, Sector Erase and Bulk Erase	Ready to accept Page Program and Sector Erase instructions

1. As defined by the values in the Block Protect (BP2, BP1, BP0) bits of the Status Register, as shown in *Table 2.*

The protection features of the device are summarized in *Table 7*.

When the Status Register Write Disable (SRWD) bit of the Status Register is 0 (its initial delivery state), it is possible to write to the Status Register provided that the Write Enable Latch (WEL) bit has previously been set by a Write Enable (WREN) instruction, regardless of the whether Write Protect ($\overline{W}$) is driven High or Low.

When the Status Register Write Disable (SRWD) bit of the Status Register is set to 1, two cases need to be considered, depending on the state of Write Protect ($\overline{W}$):

- If Write Protect ($\overline{W}$) is driven High, it is possible to write to the Status Register provided that the Write Enable Latch (WEL) bit has previously been set by a Write Enable (WREN) instruction.
- If Write Protect ($\overline{W}$) is driven Low, it is *not* possible to write to the Status Register *even* if the Write Enable Latch (WEL) bit has previously been set by a Write Enable (WREN) instruction. (Attempts to write to the Status Register are rejected, and are not accepted for execution). As a consequence, all the data bytes in the memory area that are software protected (SPM) by the Block Protect (BP2, BP1, BP0) bits of the Status Register, are also hardware protected against data modification.

Regardless of the order of the two events, the Hardware Protected Mode (HPM) can be entered:

- by setting the Status Register Write Disable (SRWD) bit after driving Write Protect ($\overline{W}$) Low
- or by driving Write Protect ($\overline{W}$) Low after setting the Status Register Write Disable (SRWD) bit.

The only way to exit the Hardware Protected Mode (HPM) once entered is to pull Write Protect ($\overline{W}$) High.

If Write Protect ($\overline{W}$) is permanently tied High, the Hardware Protected Mode (HPM) can never be activated, and only the Software Protected Mode (SPM), using the Block Protect (BP2, BP1, BP0) bits of the Status Register, can be used.