Flash Memory Operations 8M3
Application Note

FLASH MEMORY OPERATION 8M3

PRINCIPLES OF OPERATION

The following Principles of Operation of the WF1M32B-XXX3 MCM is applicable to each of the four memory chips inside the MCM. Chip 1 is distinguished by CS1# and I/O0-7, Chip 2 by CS2# and I/O8-15, Chip 3 by CS3# and I/O16-23, and Chip 4 by CS4# and I/O24-31.

The device requires only single 3.0V power supply for both read and write functions. Internally generated and regulated voltages are provided for the program and erase operations.

Commands are written to the command register using standard microprocessor timings. Register contents serve as input to an internal state-machine that controls the erase and programming circuitry. Write cycles also internally latch addresses and data needed for the programming and erase operations. Reading data out of the device is similar to reading from other Flash or EPROM devices.

Device programming occurs by excecuting the program command sequence. This initiates the Embedded Program algorithm – an internal algorithm that automatically times the program pulse widths and verifies proper cell margin. The Unblock Bypass mode facilitates faster programming times by requiring only two write cycles to program data instead of four.

Device erasure occurs by executing the erase command sequence. This initiates the Embedded Erase algorithm – an internal algorithm that automatically preprograms the array (if it is not already programmed) before executing the erase operation. During erase, the device automatically times the erase pulse widths and verifies proper cell margin.

The host system can detect whether a program or erase operation is complete by reading the DQ7 (Data# Polling) and DQ6 (toggle) Status Bits. After a program or erase cycle has been completed, the device is ready to read array data or accept another command.

The Sector Erase Architecture allows memory sectors to be erased and reprogrammed without affecting the data contents of other sectors.

Hardware Data Protection measures include a low V_{CC} detector and automatically inhibits write operations during power transitions. The Hardware Sector Protection feature disables both program and erase operations in any combination of the sectors of memory. This can be achieved in-system or via programming equipment.

The Erase Suspend feature enables the user to put erase on hold for any period of time to read data from, or program data to, any sector that is not selected for erasure. True background erase can thus be achieved.

The Hardware Reset pin terminates any operation in progress and resets the internal state machine to reading array data. The RESET# pin may be tied to the system reset circuitry. A system reset would thus also reset the device, enabling the system microprocessor to read the boot-up firmware from the Flash memory.

The device offers two power-saving features. When addresses have been stable for a specified amount of time, the device enters the Automatic Sleep Mode. The system can also place the device into the Standby Mode. Power consumption is greatly reduced in both these modes

Bus Operation

This section describes the requirements and use of the device bus operations, which are initiated through the internal command register. The command register itself does not occupy any addressable memory location. The register is composed of latches that store the commands, along with the address and data information needed to execute the command. The contents of the register serve as inputs to the internal state machine. The state machine outputs dictate the function of the device. Table 1 lists the device bus operations, the inputs and control levels required, and the resulting output. The following subsections describe each of these operations in further detail.

Requirements For Reading Array Data

To read array data from the outputs, the system must drive the CS# and OE# pins to V_{IL}. CS# is the power control and selects the device. OE# is the output control and gates array data to the output pins. WE# should remain at V_{IH}.

The internal state machine is set for reading array data upon device power-up, or after a hardware reset. This ensures that no spurious alteration of the memory content occurs during the power transition. No command is necessary in this mode to obtain array data. Standard microprocessor read cycles that assert valid addresses

Table 1 – Bus Operations

Operation	CS#	OE#	W#E	RESET#	Addresses (1)	DQ0-7
Read	L	L	H	H	A_{IN}	D_{OUT}
Write	L	H	L	H	A_{IN}	D_{OUT}
Standby	V_{CC} ±0.3V	X	X	V_{CC} ± 0.3V	X	High Z
Output Disable	L	H	H	H	X	High Z
Reset	X	X	X	L	X	High Z
Sector Protect (1)	L	H	L	V_{ID}	Sector Address, A7 = L, A2 = H, A1 = L	D_{IN}
Sector Unprotect (1)	L	H	L	V_{ID}	Sector Address, A7 = H, A2 = H, A1 = L	D_{IN}
Temporary Sector Unprotect	X	X	X	V_{ID}	A_{IN}	D_{IN}

LEGEND:
L = Logic Low = V_{IL} X = Don't Care D_{OUT} = Data Out
H = Logic High = V_{IH} A_{IN} = Address In
V_{ID} = 12.0 ± 0.5V D_{IN} = Data In

NOTES:
1. *The sector protect and sector unprotect functions may also be implemented via programming equipment. See the "Sector Protection/ Unprotection" section.*

on the device data outputs. The device remains enabled for read access until the command register contents are altered.

See "Reading Array Data" for more information. Refer to the Data Sheet AC Read Operations table for timing specifications and to AC Waveforms for Read Operations for the timing diagram. I_{CC1} in the DC Characteristics table represents the active current specification for reading array data.

Write Commands/Command Sequences

To writes a command or command sequence (which includes programming data to the device and erasing sectors of memory), the system must drive WE# and CS# to V_{IL}, and OE# to V_{IH}.

The device features an Unlock Bypass mode to facilitate faster programming. Once the device enters the Unlock Bypass mode, only two write cycles are required to program a byte, instead of four. The "Byte Program Command Sequence" section has details on programming data to the device using both standard and Unlock Bypass command sequences.

An erase operation can erase one sector, multiple sectors, or the entire device. Table 2 indicates the address space that each sector occupies. A "sector address" consists of the address bits required to uniquely select a sector. The "Command Definitions" section has details on erasing a sector or the entire chip, or suspending/resuming the erase operation.

After the system writes the autoselect command sequence, the device enters the autoselect mode. The system can then read autoselect codes from the internal register (which is separate from the memory array) on DQ7-0. Standard read cycle timings apply in this mode. Refer to the "Autoselect Mode" and "Autoselect Command Sequence" sections for mor information.

I_{CC2} in the DC Characteristics table represents the active current specificatips for the write mode. The "AC Characteristics" section contains timing specification tables and timing diagrams for write operations.

Program and Erase Operation Status

During an erase or program operation, the system may check the status of the operation by reading the status bits on DQ7-0. Standard read cycle timings and I_{CC} read specifications apply. Refer to "Write Operation Status" for more information, and to the Data Sheet "AC Characteristics" for timing diagrams

Standby Mode

When the system is not reading or writing to the device, it can place the device in standby mode. In this mode, current consumption is greatly reduced, and the outputs are placed in the high impedance state, independent of the OE# input.

The device enters the CMOS standby mode when the CS# and RESET# pins are both held at V_{CC} ±0.3V. (Note that this is a more restricted voltage range than V_{IH}.) If

CS# and RESET# are held at V_{IH}, but not within the device will be in the standby mode, but the standby current will be greater. The device requires standard access time (t_{CE}) for read access when the device is in either of these standby modes, before it is ready to read data.

If the device is deselected during erasure or programming, the device draws active current until the operation is completed.

In the Data Sheet DC Characteristics table, I_{CC3} represents the standby current specification.

RESET#: Hardware Reset Pin

The RESET# pin provides a hardware method of resetting the device to reading array data. When the RESET# pin is driven low for at least a period of 500ns or greater the device immediately terminates any operation in progress, tristates all output pins, and ignores all read/write commands for the duration of the RESET# pulse. The device also resets the internal state machine to reading array data. The operation that was interrupted should be re-initiated once the device is ready to accept another command sequence, to ensure data integrity.

Current is reduced for the duration of the RESET# pulse. When RESET# is held at V_{CC} ± 0.3V, the device draws CMOS standby current. If RESET# is held at V_{IL} but not within V_{CC} ± 0.3V, the standby current will be greater.

The RESET# pin may be tied to the system reset circuitry. A system reset would thus also reset the Flash memory, enabling the system to read the boot-up firmware from the Flash memory.

Output Disable Mode

When the OE# input is at V_{IH}, output from the device is disabled. The output pins are placed in the high impedance state.

Autoselect Mode

The autoselect mode provides manufacturer and device identification, and sector protection verification, through identifier codes input codes output on DQ7-0. This mode is primarily intended for programming equipment to automatically match a device to be programmed with its corresponding programming algorithm. However, the autoselect codes can also be accessed in-system through the command register.

When using programming equipment, the autoselect mode requires V_{ID} (11.5V to 12.5V) on address pin A10. Address pins A7, A2, and A1 must be as shown in Table 3. In addition, when verifying sector protection, the sector address must appear on the appropriate highest order address bits (see Table 2). Table 3 shows the remaining address bits that are "don't care." When all necessary bits have been set as required, the programming equipment may then read the corresponding identifier code on DQ7-0.

To access the autoselect codes in-system, the host system can issue the autoselect command via the command register, as shown in Table 4. This method does not require V_{ID}. See "Command Definitions" for details on using the autoselect mode.

Sector Protection/Unprotection

The hardware sector protection feature disables both program and erase operations in any sector. The hardware sector unprotection feature re-enables both program and erase operations in previously protected sectors.

The device is shipped with all sectors unprotected.

It is possible to determine whether a sector is protected or unprotected. See "Autoselect Mode" for details.

Sector Protection/unprotection can be implemented via two methods.

The primary method requires V_{ID} on the RESET# pin only, and can be implemented either in-system or via programming equipment. Figure 3 shows the algorithms and the timing diagram is shown on the Sector Protect/Unprotect Timing Diagram in the data sheet. This method uses standard microprocessor bus cycle timing. For sector unprotect, all unprotected sectors must first be protected prior to the first sector unprotect write cycle.

The alternate method intended only for programming equipment requires V_{ID} on address pin A10 and OE#. This method is compatible with programmer routines written for earlier 3.3V only flash devices.

Temporary Sector Unprotect

This feature allows temporary unprotection of previously protected sector groups to change data-in system. The Sector Unprotect mode is activated by setting the RESET# pin to V_{ID}. During this mode, formerly protected sector can be programmed or erased by selecting the sector addresses. Once V_{ID} is taken away from the RESET# pin, all the previously protected sector groups will be protected again. Figure 2 shows the algorithm and the timing diagram is shown in Figure 1, Temporary Sector Unprotect Timing Diagram.

Temporary Sector Unprotect

Symbol	Description	Min	Unit
t_{VIDR}	V_{ID} Rise and Fall Time*	500	ns
t_{RSP}	RESET# Setup Time for Temporary Sector Unprotect	4	µs

** Not 100% tested.Hardware Data Protection*

Table 2 – Bottom Boot Block Sector Address Table

Sector	A19	A18	A17	A16	A15	A14	A13	Sector Size (Kbytes)	(x8) Address Range (In hexidecimal)
SA0	0	0	0	0	0	0	X	16	00000h-03FFFh
SA1	0	0	0	0	0	1	0	8	04000h-05FFFh
SA2	0	0	0	0	0	1	1	8	06000h-07FFFh
SA3	0	0	0	0	1	X	X	32	08000h-0FFFFh
SA4	0	0	0	1	X	X	X	64	10000h-1FFFFh
SA5	0	0	1	0	X	X	X	64	20000h-2FFFFh
SA6	0	0	1	1	X	X	X	64	30000h-3FFFFh
SA7	0	1	0	0	X	X	X	64	40000h-4FFFFh
SA8	0	1	0	1	X	X	X	64	50000h-5FFFFh
SA9	0	1	1	0	X	X	X	64	60000h-6FFFFh
SA10	0	1	1	1	X	X	X	64	70000h-7FFFFh
SA11	1	0	0	0	X	X	X	64	80000h-8FFFFh
SA12	1	0	0	1	X	X	X	64	90000h-9FFFFh
SA13	1	0	1	0	X	X	X	64	A0000h-AFFFFh
SA14	1	0	1	1	X	X	X	64	B0000h-BFFFFh
SA15	1	1	0	0	X	X	X	64	C0000h-CFFFFh
SA16	1	1	0	1	X	X	X	64	D0000h-DFFFFh
SA17	1	1	1	0	X	X	X	64	E0000h-EFFFFh
SA18	1	1	1	1	X	X	X	64	F0000h-FFFFFh

Table 3 – Autoselect Codes (High Voltage Method)

Description	CS#	OE#	WE#	A19-13	A12-11	A10	A9-8	A7	A6-3	A2	A1	DQ7-0
Manufacturer ID	L	L	H	X	X	V_{ID}	X	L	X	L	L	01h
Device ID (Bottom Boot Block)	L	L	L	X	X	V_{ID}	X	L	X	L	H	5Bh
Sector Protection	L	L	H	SA	X	V_{ID}	X	L	X	H	L	01h (protected)
												01h (protected)

L = Logic Low = V_{IL}, H = Logic High = V_{IH}, SA = Sector Address, X = Don't Care

The command sequence requirement of unlock cycles for programming or erasing provides data protection against inadvertent writes (refer to Table 4 for command definitions). In addition, the following hardware data protection measures prevent accidental erasure or programming, which might otherwise be caused by spurious system level signals during V_{CC} power-up and power-down transitions, or from system noise.

Low V_{CC} Write Inhibit

When V_{CC} is less than 2.3V, the device does not accept any write cycles. This protects data during V_{CC} power-up and power-down. The command register and all internal program/erase circuits are disabled, and the device resets. Subsequent writes are ignored until V_{CC} is greater than 2.5V. The system must provide the proper signals to the control pins to prevent unintentional writes when V_{CC} is greater than 2.5V.

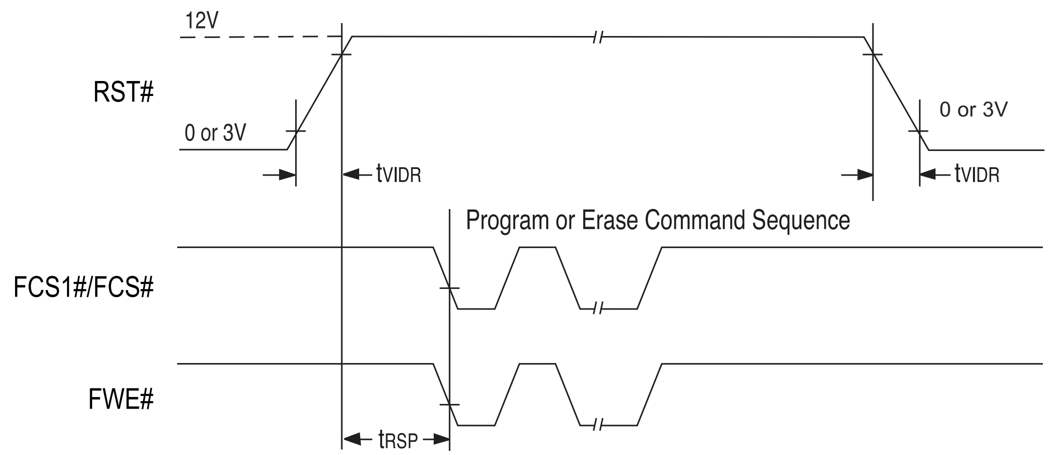
Write Pulse "Glitch" Protection

Noise pulses of less than 5ns (typical) on OE#, CS# or WE# do not initiate a write cycle.

Logical Inhibit

Write cycles are inhibited by holding any one of OE# = V_{IL}, CS# = V_{IH} or WE# = V_{IH}. To initiate a write cycle, CS# and WE must be a logical zero while OE# is a logical one.

Figure 1 – Temporary Sector Unprotect Timing Diagram



Power-Up Write Inhibit

If WE# = CS# = V_{IL} and OE# = V_{IH} during power up, the device does not accept commands on the rising edge of WE#. The internal state machine is automatically reset to reading array data on power-up.

Command definitions

Writing specific address and data commands or sequences into the command register initiates device operations. Table 4 defines the valid register command sequences. Writing incorrect address and data values or writing them in improper sequence will reset the device to the read array data.

All addresses are latched on falling edge of WE# or CS#, whichever occurs later. All data is latched on the rising edge of WE# or CS#, whichever occurs first. Refer to the appropriate timing diagrams in the Data Sheet "AC Characteristics" section.

Read Array Data

Upon initial device power-up the device defaults to read array data. No commands are required to retrieve data. The device is also ready to read array data after it has completed an Embedded Program or Embedded Erase algorithm.

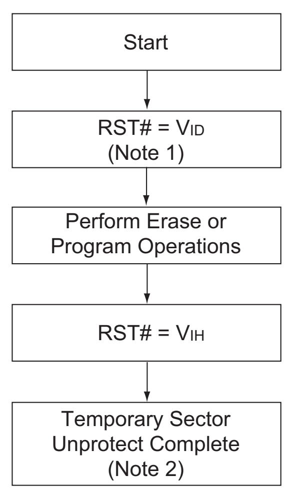
After the device accepts an Erase Suspend command, the device enters the Erase Suspend mode. The system can read array data using the standard read timings, except that if it reads at an address within erase-suspend sectors, the device outputs status data. After completing a programming operation in the Erase Suspend mode, the system may once again read array data with the same exception. See "Erase Suspend/Erase Resume Commands" for more information on this mode.

The system *must* issue the reset command to re-enable the device for reading array data if DQ5 goes high, or while in the autoselect mode. See the "Reset Command" section, next.

See also "Requirements for Reading Array Data" on the "Bus Operations" section for more information. The Data Sheet Read Operations table provides the read parameters, and the Read Operations Timing Diagram shows the timing diagram.

Continued on page 7

**Figure 2 –
Temporary Sector Unprotect Operation**



1. *All protected sectors unprotected.*
2. *All previously protected sectors are protected once again.*

Table 4 – Command Definitions

Command Sequence (Note 1)		Bus Write Cycles Req'd	Bus Cycles (Notes 2-5) First Bus Cycle Addr	First Bus Cycle Data	Second Bus Cycle Addr	Second Bus Cycle Data	Third Bus Cycle Addr	Third Bus Cycle Data	Fourth Bus Cycle Addr	Fourth Bus Cycle Data	Fifth Bus Cycle Addr	Fifth Bus Cycle Data	Sixth Bus Cycle Addr	Sixth Bus Cycle Data
Read (Note 5)		1	RA	RD										
Reset (Note 6)		1	XXX	F0										
Autoselect (Note 7): Manufacturer ID	Byte	4	AAA	AA	555	55	AAA	90	X00	01				
Autoselect (Note 7): Device ID, Bottom Boot Block	Byte	4	AAA	AA	555	55	AAA	90	X02	5B				
Autoselect (Note 7): Sector Protect Verify (Note 8)	Byte	4	AAA	AA	555	55	AAA	90	(SA) X04	00 01				
Program	Byte	4	AAA	AA	555	55	AAA	A0	PA	PD				
Unlock Bypass	Byte	3	AAA	AA	555	55	AAA	20						
Unlock Bypass Program (Note9)		2	XXX	A0	PA	PD								
Unlock Bypass Reset (Note10)		2	XXX	90	PA	00								
Chip Erase	Byte	6	AAA	AA	555	55	AAA	80	AAA	AA	555	55	AAA	10
Sector Erase	Byte	6	AAA	AA	555	55	AAA	80	AAA	AA	555	55	SA	30
Erase Suspended (Note 11)		1	XXX	B0										
Erase Resume (Note 12)		1	XXX	30										

LEGEND:

X = Don't Care

RA = Address of the memory location to be read.

RD = Data read from location RA during read operation.

PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of the WE# or CS# pulse, whichever occurs first.

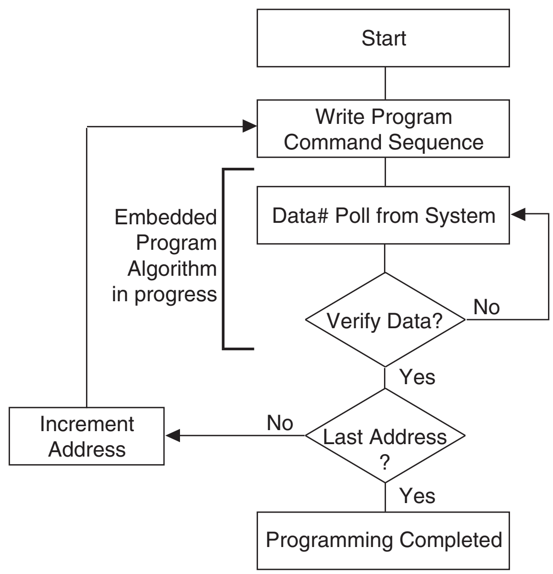
PD = Data to be programmed at location PA. Data is latched on the rising edge of WE# or CS# pulse, whichever occurs first.

SA = Address of the sector to be erased. The combination of A19-13 will uniquely select any sector.

NOTES:

1. *Bus operations are defined in Table 1.*
2. *All values are in hexadecimal.*
3. *Except when reading array or autoselect data, all bus cycles are write operations.*
4. *Address bits A19-12 = don't care for unlock and command cycles, unless PA or SA is required.*
5. *No unlock or command cycles required when reading array data.*
6. *The Reset command is required to return to reading array data when device is in the autoselect mode, or if DQ5 goes high (while the device is providing status data).*
7. *The fourth cycle of the autoselect command sequence is a read cycle.*
8. *The data is 00h for an unprotected sector and 01h for a protected sector. See "Autoselect Command Sequence" for more information.*
9. *The Unlock Bypass command is required prior to the Unlock Bypass Program command.*
10. *The Unlock Bypass Reset command is required to return to reading array data when the device is in the Unlock Bypass mode.*
11. *The system may read and program in non-erasing sectors, or enter the autoselect mode, when in the Erase Suspend mode. The Erase Suspend command is valid only during a sector erase operation.*
12. *The Erase Resume command is valid only during the Erase Suspend mode.*

Figure 3 – Program Operation



NOTE: See Table 4 for program command sequence.

Reset Command

Writing the reset command to the device resets the device to reading array data. Address bits are "don't care" for this command.

The reset command may be written between the sequence before erasing begins. This resets the device to reading array data. Once erasure begins, however, the device ignores reset commands until the operation is complete.

The reset command may be written between the sequence cycles in a program command sequence before programming begins. This resets the device to reading array data (also applies to programming in Erase Suspend mode). Once programming begins, however, the device ignores reset commands until the operation is complete.

The reset command may be written between the sequence cycles in an autoselect command sequence. Once in autoselect mode, the reset command *must* be written to return to reading array data (also applies to autoselect during Erase Suspend mode).

If DQ5 goes high during a program or erase operation, writing the reset command returns the device to reading array data (also applies during Erase Suspend).

Autoselect Command Sequence

The autoselect command sequence allows the host system to access the manufacturer and device codes, and determine whether or not a sector is protected. Table 4 shows the address and data requirements. This method is an alternative to that shown in Table 3, which is intended for PROM programmers and requires V_{ID} on address bit A10.

The autoselect command sequence is initiated by writing two unlock cycles, followed by the autoselect command. The device then enters the autoselect mode, and the system may read at any address any number of times, without initiating another command sequence.

A read cycle at address XX00h retrieves the manufacturer code. A read cycle at address 02h returns the device code. A read cycle containing a sector address (SA) and the address 04h returns 01h if that sector is protected, or 00h if it is unprotected. Refer to Table 2 for valid sector addresses.

The system must write the reset command to exit the autoselect mode and return to reading array data.

Byte Program Command Sequence

Programming is a four-bus-cycle operation. The program command sequence is initiated by writing two unlock write cycles, followed by the program setup command. The program address and data are written next, which in turn initiate the Embedded Program algorithm. The system is not required to provide further controls or timings. The device automatically provides internally generated program pulses and verifies the programmed cell margin. Table 4 shows the address and data requirements for the byte program command sequence.

When the Embedded Program algorithm is complete, the device then returns to reading array data and addresses are no longer latched. The system can determine the status of the program operation by using DQ7, or DQ6. See "Write Operation Status" for information on these status bits.

Any commands written to the device during the Embedded Program algorithm are ignored. Note that a hardware reset immediately terminates the programming operation. The program command sequence should be re-initiated once the device has reset to reading array data, to ensure data integrity.

Programming is allowed in any sequence and across sector boundaries. A bit cannot be programmed from a "0" back to a "1". Attempting to do so may halt the operation and set DQ5 to "1", or cause the Data# Polling algorithm to indicate the operation was successful. However, a succeeding read will show that the data is still "0". Only erase operations can convert a "0" to a "1".

Unlock Bypass Command Sequence

The unlock bypass feature allows the system to program bytes or words to the device faster than using the standard program command sequence. The unlock bypass command sequence is initiated by first writing two unlock cycles. This is followed by a third write cycle containing the unlock bypass command, 20h. The device then enters the unlock bypass mode. A two-cycle unlock bypass program command sequence is all that is required to program in this mode. The first cycle in this sequence contains the unlock bypass program command, A0h; the second cycle contains the program address and data. Additional data is programmed in the same manner. This mode dispenses with the initial two unlock cycles required in the standard program command sequence, resulting in fast total programming time. Table 4 shows the requirements for the command sequence.

During the unlock bypass mode, only the Unlock Bypass Program and Unlock Bypass Reset commands are valid. To exit the unlock bypass mode, the system must issue the two-cycle unlock bypass reset command sequence. The first cycle must contain the data 90h; the second cycle the data 00h. Addresses are "don't care" for both cycles. The device then returns to reading array data.

Figure 3 illustrates the algorithm for the program operation. See the Erase/Program Operations table in the Data Sheet "AC Characteristics" for parameters, and to the Program Operation Timings Diagram.

Chip Erase Command Sequence

Chip erase is six bus cycle operation. The chip erase command sequence is initiated by writing two unlock cycles, followed by a setup command. Two additional unlock write cycles are then followed by the chip erase command, which in turn invokes the Embedded Erase algorithm. The device does not require the system to preprogram prior to erase. The Embedded Erase algorithm automatically programs and verifies the entire memory for an all zero data pattern prior to electrical erase. The system is not required to provide any controls or timings during these operations. Table 4 shows the address and data requirements for the chip erase command sequence.

Any commands written to the chip during the Embedded Erase algorithm are ignored. Note that a hardware reset during the chip erase operation immediately terminates the operation. The Chip Erase command sequence should be re-initiated once the device has returned to reading array data, to ensure data integrity.

The system can determine the status of the erase operation by using DQ7, DQ6, or DQ2. See "Write Operation Status" for information on these status bits. When the Embedded Erase algorithm is complete, the device returns to reading array data and addresses are no longer latched.

Figure 4 illustrates the algorithm for the erase operation. See the Erase/Program Operations tables in "AC Characteristics" for parameters, and to the Chip/Sector Erase Operation Timings Diagram in the Data Sheet.

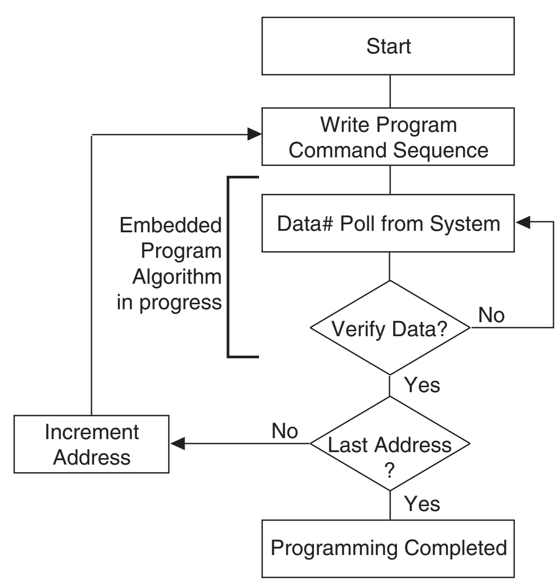
Sector Erase Command Sequence

Sector erase is six bus cycle operation. The sector erase command sequence is initiated by writing two unlock cycles, followed by a setup command. Two additional unlock write cycles are then followed by the address of the sector to be erased, and the sector erase command, which in turn invokes the Embedded Erase algorithm. Table 4 shows the address and data requirements for the sector erase command sequence.

The device does not require the system to preprogram the memory prior to erase. The Embedded Erase algorithm automatically programs and verifies the entire memory for an all zero data pattern prior to electrical erase. The system is not required to provide any controls or timings during these operations.

After the command sequence is written, a sector erase time-out of 50µs begins. During the time-out period, additional sector addresses and sector erase commands may be written. Loading the sector erase buffer may be done in any sequence, and the number of sectors may be from one sector to all sectors. The time between these additional cycles must be less than 50µs, otherwise the last address and command might not be accepted, and erasure may begin. It is recommended that processor interrupts can be re-enabled after the last Sector Erase command is written. If the time between additional sector erase commands can be assumed to be less than 50µs, the system need not monitor DQ3.

Figure 4 – Erase Operation



1. *See Table 4 for erase command sequence.*
2. *See "DQ3 : Sector Erase Timer" for more information.*

Any command other than the Sector Erase or Erase Suspend during the time-out period resets the device to reading array data. The system must rewrite the command sequence and any additional sector addresses and commands.

The system can monitor DQ3 to determine if the sector erase timer has timed out. See the "DQ3: Sector Erase Timer" section. The time-out begins from the rising edge of the final WE pulse in command sequence.

When the Embedded Erase algorithm is complete, the device returns to reading array data and addresses are no longer latched. The system can determine the status of the erase operation by using DQ7, DQ6, or DQ2. See "Write Operation Status" for information on these status bits.

Figure 4 illustrates the algorithm for the erase operation. See the Erase/Program Operations tables in the Data Sheet "AC Characteristics" for parameters, and to the Chip/Sector Erase Operation Timings Diagram.

Erase Suspend/Erase Resume Command Sequence

The Erase Suspend command allows the system to interrupt a sector erase operation and then read data from, or program data to, any sector not selected for erasure. This command is valid only during the sector erase operation, including the 50µs time-out period during the sector erase command sequence. The Erase Suspend command is ignored if written during the chip erase operation or Embedded Program algorithm. Writing the Erase Suspend command during the Sector Erase time-out immediately terminates the time-out period and suspends the erase operation. Addresses are "don't cares" when writing the Erase Suspend command.

When the Erase Suspend command is written during a sector erase operation, the device requires a maximum of 20µs to suspend the erase operation. However, when the Erase Suspend command is written during the sector erase time-out, the device immediately terminates the time-out period and suspends the erase operation.

After the erase operation has been suspended, the system can read array data from or program data to any sector not selected for erasure. (The device "erase suspends" all sectors selected for erasure.) Normal read and write timings and command definitions apply. Reading at any address within erase-suspended sectors produces status data on DQ7-0. The system can use DQ7, or DQ6, and DQ2 together, to determine if a sector is actively erasing or is erase suspended. See "Write Operation Status" for information on these status bits.

After an erase-suspended program operation is complete, the system can once again read array data within non-suspended sectors. The system can determine the status of the program operation using the DQ7 or DQ6 status bits, just as in the standard program operation. See the "Write Operation Status" for more information.

The system may also write the autoselect command sequence when the device is in the Erase Suspend mode. The device allows reading autoselect codes even at addresses within erasing sectors, since the codes are not stored in the memory array. When the device exits the autoselect mode, the device reverts to the Erase Suspend mode, and is ready for another valid operation. See "Autoselect Command Sequence" for more information.

The system must write the Erase Resume command (address bits are "don't care") to exit the erase suspend mode and continue the sector erase operation. Further writes of the Resume command are ignored. Another Erase Suspend command can be written after the device has resumed erasing.

Write Operation Status

The device provides several bits to determine the status of a write operation: DQ2, DQ3, DQ5, DQ6, and DQ7. Table 5 and the following subsections describe the functions of these bits. DQ7 and DQ6 each offer a method for determining whether a program or erase operation is complete or in progress. These two bits are discussed first.

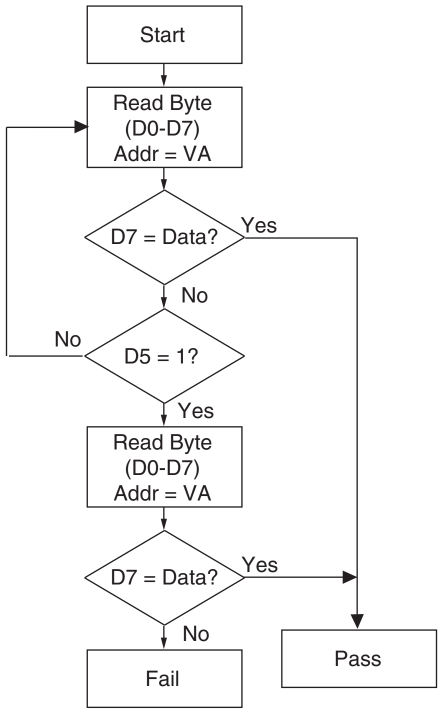
Table 5 – Write Operation Status

	Status	DQ_7(2)	DQ_6	DQ_5(1)	DQ_3	DQ_2(2)
Standard Mode	Embedded Program Algorithm	DQ_7#	Toggle	0	N/A	No Toggle
	Embedded Erase Algorithm	0	Toggle	0	1	Toggle
Erase Suspend Mode	Reading within Erase Suspended Sector	1	No Toggle	0	N/A	Toggle
	Reading within Non-Erase Suspended Sector	Data	Data	Data	Data	Data
	Erase Suspended Program	DQ_7#	Toggle	0	N/A	N/A

NOTES:

1. *DQ_5 switches to "1" when an Embedded Program or Embedded Erase operation has exceeded the maximum timing limits. See "DQ_5: Exceed Timing Limits" for more information.*
2. *DQ_7 and DQ_2 require valid address when reading status information. Refer to the appropriate subsection for further details.*

Figure 5 – Data# Polling Algorithm



VA = Byte address for programming
= Any of the sector addresses within the sector being erased during sector erase operation
= Valid address equals any non-protected sector group address during chip erase

1. *DQ_7 should be rechecked even if $DQ_5 = 1$ because DQ_7 may change simultaneously with DQ_5.*

DQ_7

Data# Polling

The Data# Polling bit, DQ7, indicates to the host system whether an Embedded Algorithm is in progress or completed, or whether the device is in Erase Suspend Data# Polling valid after the rising edge of the final WE# pulse in the program or erase command sequence.

During the Embedded Program algorithm, the device outputs on DQ7 the complement of the datum programmed to DQ7. This DQ7 status also applies to programming during Erase Suspend. When the Embedded Program algorithm is complete, the device outputs the datum programmed to DQ7. The system must provide the program address to read valid status information on DQ7. If a program address falls within a protected sector, Data# Polling on DQ7 is active for approximately 1μs, then the device returns to reading array data.

During the Embedded Erase algorithm, Data# Polling produces a “0” on DQ7. When the Embedded Erase algorithm is complete, or if the device enters the Erase Suspend mode, Data# Polling produces a “1” on DQ7. This analogous to the complement/true datum output described for the Embedded Program algorithm: the erase function changes all the bits in a sector to “1”; prior to this, the device outputs the “complement,” or “0.” The system must provide an address within any of the sectors selected for erasure to read valid status information on DQ7.

After an erase command sequence is written, if all sectors selected for erasing are protected, Data# Polling on DQ7 is active for approximately 100μs, then the device returns to reading array data#. If not all selected sectors are protected, the Embedded Erase algorithm erases the unprotected sectors, and ignores the selected sectors that are protected.

When the system detects DQ7 has changed from the complement to true data, it can read valid data at DQ7-0 on the following read cycles. This because DQ7 may change asynchronously with DQ0-6 while Output Enable (OE#) is asserted low. Data# Polling (During Embedded algorithms), in the selected sectors that are protected.

DQ_6

Toggle Bit I

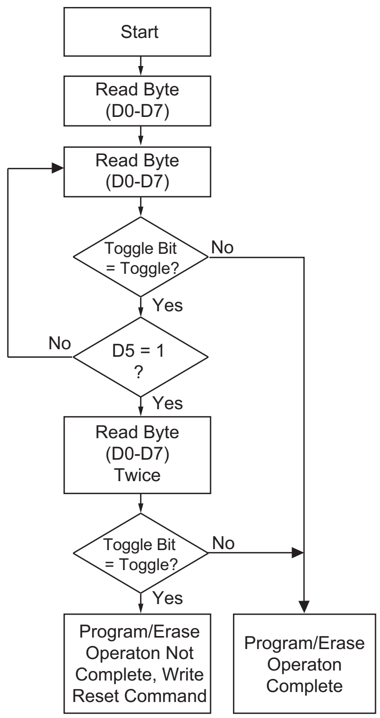
“Toggle Bit I” on DQ6 indicates whether an Embedded Program or Erase Algorithm is in progress or has been completed, or whether the device has entered the Erase Suspend mode. Toggle Bit I may read at any address, and is valid after the rising edge of the final WE pulse in the command sequence (prior to the program or erase operation), and during the sector erase time-out.

During an Embedded Program or Erase Algorithm operation, successive read cycles to any address will result in DQ6 toggling. (The system may use either OE# or CS# to control the read cycles.) When operation is complete, DQ6 stops toggling.

After the erase command sequence is written, if all selectors selected for erasing are protected. DQ6 toggles for approximately 100μs, then returns to reading array data. If not all selected sectors are protected, the Embedded Erase Algorithm erases the unprotected sectors, and ignores the selected sectors that are protected.

The system can use DQ6 and DQ2 together to determine whether a sector is actively erasing or is erase-suspended. When the device is actively erasing (that is, the Embedded Erase Algorithm is in progress) DQ6 toggles. When the device enters the Erase Suspend mode, DQ6 stops toggling. However, the system must also use DQ2 to determine which sectors are erasing or erase-suspended. Alternatively, the system can use DQ7 (see the subsection on “DQ7: Data# Polling”).

Figure 6 – Toggle Bit Algorithm



1. *Read toggle bit twice to detemine whether or not it is toggling. See text.*
2. *Recheck toggle bit because it may stop toggling as DQ_5 changes to 1. See text.*

If a program address falls within a protected sector, DQ6 also toggles for approximately 1µs after the program command sequence is written, then returns to reading array data.

DQ6 also toggles during erase-suspend-program mode, and stops toggling once the Embedded Program algorithm is complete.

Table 5 shows the outputs for “Toggle Bit I” on DQ6. Figure 6 shows the Toggle Bit Algorithm. See also the subsection on “DQ2: Toggle Bit II”.

DQ_2

Toggle Bit II

The “Toggle Bit II” on DQ2, when used with DQ6, indicates whether a particular sector is actively erasing (that is, the Embedded Erase Algorithm is in progress) or whether that sector is erase-suspended. “Toggle Bit II” is valid after the rising edge of the final WE# pulse in the command sequence.

DQ2 toggles when the system reads at addresses within those sectors that have been selected for erasure. (The system may use either OE# or CS# to control the read cycles.) DQ2 cannot distinguish whether the sector is actively erasing or is erase-suspended. DQ6, by comparison, indicates whether the device is actively erasing, or is in Erase Suspend, but cannot distinguish which sectors are selected for erasure. Thus, both status bits are required for sector and mode information. Refer to Table 6 to compare outputs for DQ2 and DQ6.

Figure 6 shows the Toggle Bit Algorithm in flowchart form, and the section “DQ2: Toggle Bit II” explains the algorithm. See also the subsection on “DQ6: Toggle Bit I”.

Reading Toggle Bits DQ_6/DQ_2

Refer to Figure 6 for the following discussion. Whenever the system initially begins reading toggle bit status, it must read DQ7-0 at least twice in a row to determine whether a toggle bit is toggling. Typically, the system would note and store the value of the toggle bit after the first read. After the second read, the system would compare the new value of the toggle bit with the first. If the toggle bit is not toggling, the device has completed the program or erase operation. The system can read array data on DQ7-0 on the following read cycle.

However, if after the initial two read cycles, the system determines that the toggle bit is still toggling, the system also should note whether the value of DQ5 is high (see the section on DQ5). If it is, the system should then determine again whether the toggle bit is toggling, since the toggle bit may have stopped toggling just as the device has successfully completed the program or erase operation. If it is still toggling, the device did not complete the operation successfully, and the system must write the reset command to return to reading array data.

The remaining scenario is that the system initially determines that the toggle bit is toggling and DQ5 has not gone high. The system may continue to monitor the toggle bit and DQ5 through successive read cycles, determining the status as described in the previous paragraph. Alternatively, it may choose to perform other system tasks. In this case, the system must start at the beginning of the algorithm when it returns to determine the status of the operation (top of Figure 6).

DQ_5

Exceeded Timing Limits

DQ5 will indicate whether the program or erase time has exceeded the specified limits (internal pulse count). Under these conditions DQ5 will produce a “1”. This is a failure condition that indicates the program or erase cycle was not successfully completed.

The DQ5 failure condition may appear if the system tries to program a “1” to a location that is previously programmed to “0.” Only an erase operation can change a “0” back to a “1.” Under this condition, the device halts the operation, and when the operation has exceeded timing limits, the DQ5 bit will produce a “1”.

Under both these conditions, the system must issue the reset command to return the device to reading array data.

DQ3

Sector Erase Timer

After writing a sector erase command sequence, the system may read DQ3 to determine whether or not an erase operation has begun. (The sector erase timer does not apply to the chip erase command.) If additional sectors are selected for erasure, the entire time-out also applies after each additional sector erase command. When the time-out is completed, DQ3 switches from “0” to “1.” The system may ignore DQ3 if the system can guarantee that the time between additional sector erase commands will always be less than 50µs. See also the “Sector Command Sequence” section.

After the sector erase command sequence is written, the system should read the status on DQ7 (Data# Polling) or DQ6 (Toggle Bit I) to ensure the device has accepted the command sequence, and then read DQ3. If DQ3 is high (“1”) the internally controlled erase cycle has begun; all further commands (other than Erase Suspend) will be ignored until the erase operation is completed. If DQ3 is low (“0”), the device will accept additional sector erase commands. To ensure the command has been accepted, the system software should check the status of DQ3 prior to and following each subsequent sector erase command. If DQ3 is high on the second status check, the last command may not have been accepted. Table 5 shows the outputs for DQ3.