



Flash Memory Operations 16M5 Application Note

FLASH MEMORY OPERATION 16M5

FOR THE FOLLOWING PARTS:

WF2M 8-XXX5 • WF2M 16-XXX5 • WF2M 32-XXX5 • WF4M 16-XDTX5 • WF4M 32-XXX5

PRINCIPLES OF OPERATION

The following information is relative to one 2Mx8 Flash memory device. If you are using a module with multiple devices organized to increase bus width (x16, x32), all references to the signal data bus must be applied to each of the multiple data buses. If you are using a module with devices organized to increase memory depth, the CS# pins and upper address pin(s) need to be used to select which devices are being accessed. modules without a RESET# pin have the RESET# internally connected to Vcc. Refer to the product data sheet block diagram and pin configuration to clarify the organization of the module.

BUS OPERATIONS

Read Mode

The 16M5 has two control functions which must be satisfied in order to obtain data at the outputs. CS# is the power control and should be used for device selection. OE# is the output control and should be used to gate data to the output pins if the device is selected.

Output Disable

With the OE# input at a logic-high level (VIH), output from the device is disabled. This will cause output pins to be in a high impedance state.

Standby Mode

There are two ways to implement the standby mode on the 16M5 device, one using both the CS# and RESET# pins; the other via the RESET# pin only.

When using both pins, a CMOS standby mode is achieved with CS# and RESET# inputs both held at Vcc ± 0.3V. Under this condition the current is typically reduced to less than 1µA. A TTL standby mode is achieved with CS# and RESET# pins held at VIH. Under this condition the current is typically reduced to 200µA.

When using the RESET# pin only, a CMOS standby mode is achieved with RESET# input held at Vcc ± 0.3V (CS# = don't care). Under this condition the current is typically reduced to less than 1µA. A TTL standby mode is achieved with RESET# pin held at VIL (CS# = don't care). Under this condition the current is typically reduced to less than 200µA. Once the RESET# pin is taken high, the device

requires 500ns of wake up time before outputs are valid for read access.

In standby mode the outputs are in high impedance state, independent of the OE# input.

Autoselect

The autoselect mode allows the reading of a binary code from the device and will identify its type. This mode is intended for use by programming equipment for the purpose of automatically matching the device to be programmed with its corresponding programming algorithm. This mode is functional over the entire temperature range of the device.

Table 1 – Bus Operations

Operation	CS#	OE#	WE#	A0	A1	A6	A9	DQ0-7	RESET#
Autoselect, AMD Manuf. Code (1)	L	L	H	L	L	L	V _{ID}	Code	H
Autoselect Device Code (1)	L	L	H	H	L	L	V _{ID}	Code	H
Read	L	L	H	A ₀	A ₁	A ₆	A ₉	DOUT	H
Standby	H	X	X	X	X	X	X	HIGH Z	H
Output Disable	L	H	H	X	X	X	X	HIGH Z	H
Write	L	H	L	A ₀	A ₁	A ₆	A ₉	DIN	H
Verify Sector Group Protect(2)	L	L	H	L	H	L	V _{ID}	Code	H
Temporary Sector Group Unprotect	X	X	X	X	X	X	X	X	V _{ID}
Hardware Reset/ Standby	X	X	X	X	X	X	X	HIGH Z	L

LEGEND:

L = logic, H = logic 1, X = don't care. See DC Characteristics for voltage levels.

NOTES:

1. Manufacturer and device codes may also be accessed via a command register with sequence. Refer to Table 5.
2. Refer to the section on Sector Group Protection.

To activate this mode, the programming equipment must force V_{ID} (11.5 to 12.5V) on address pin A₉. Two identifier bytes may then be sequenced from the device outputs by toggling address A₀ from V_{IL} to V_{IH}. All addresses are don't cares except A₀, A₁, and A₆ (see table 2).

Table 2 – Sector Protection Verify Autoselect Codes

Type	A18 to A20			A6	A1	A0	Code (HEX)	D7	D6	D5	D4	D3	D2	D1	D0
16M5 Device Code	X	X	X	V _{IL}	V _{IH}	V _{IL}	ADH	1	0	1	0	1	1	0	1
Sector Group Protection	Sector Group Addr.			V _{IL}	V _{IH}	V _{IL}	01H*	0	0	0	0	0	0	1	1

* Outputs 01H at protected sector addresses.

The device codes may also be read via the command register, for instances when the device is erased or programmed in a system without access to high voltage on the A₉ pin. The command sequence is illustrated in Table 5 (see Autoselect Command Sequence).

The autoselect mode also facilitates the determination of sector group protection in the system. By performing a read operation at the address location XX02H with the higher order address bits A₁₈, A₁₉, and A₂₀ set to the desired sector group address, the device will return 01H for a protected sector group and 00H for a non-protected sector group.

Write

Device erasure and programming are accomplished via the command register. The contents of the register serve as inputs to the internal state machine. The state machine outputs dictate the function of the device.

The command register itself does not occupy any addressable memory location. The register is a latch used to store the commands, along with address and data information needed to execute the command. The command register is written by bringing WE# to a logic-low level (V_{IL}), while CS# is low and OE# is at V_{IH}. Addresses are latched on the falling edge of WE# or CS#, whichever happens later. Data is latched on the rising edge of the WE# or CS# whichever occurs first. Standard microprocessor write timings are used.

Refer to AC Write Characteristics and the Erase/Programming Waveforms for specific timing parameters.

Sector Group Protection

The 16M5 features hardware sector group protection. This feature will disable both program and erase operations in any combination of eight sector groups of memory. Each sector group consists of four adjacent sectors grouped in the following pattern: sectors 0-3, 4-7, 8-11, 12-15, 16-19, 20-23, 24-27, and 28-31 (see Table 4). The sector group protect feature is enabled using programming equipment at the user's site. The device is shipped with all sector groups unprotected.

It is possible to determine if a sector group is protected in the system by writing an Autoselect command. Performing a read operation at the address location XX02H, where the higher order address bits A₁₈, A₁₉, and A₂₀ is the desired sector group address, will produce a logical

"1" at D0 for a protected sector group. See Table 2 for Autoselect codes.

Temporary Sector Group Unprotect

This feature allows temporary unprotection of previously protected sector groups of the 16M5 in order to change data-in system. The Sector Group Unprotect

mode is activated by setting the RESET# pin to high voltage (12V). During this mode, formerly protected sector groups can be programmed or erased by selecting the sector group addresses. Once the 12V is taken away from the RESET# pin, all the previously protected sector groups will be protected again.

COMMAND DEFINITIONS

Device operations are selected by writing specific address and data sequences into the command register. **Writing incorrect address and data values or writing them in improper sequence will reset the device to the read mode.** Table 5 defines the valid register command sequences. Note that the Erase Suspend (B0H) and Erase Resume (30H) commands are valid only while the Sector Erase operation is in progress. Moreover, both Reset/Read commands are functionally equivalent, resetting the device to the read mode.

Read/Reset Command

The read or reset operation is initiated by writing the read/reset command sequence into the command register. Microprocessor read cycles retrieve array data from the memory. The device remains enabled for reads until the command register contents are altered.

The device will automatically power-up in the read/reset state. In this case, a command sequence is not required to read data. Standard microprocessor read cycles will retrieve array data. This default value ensures that no spurious alteration of the memory content occurs during the power transition. Refer to the AC Read Characteristics and Waveforms for the specific timing parameters.

Autoselect Command Sequence

Flash memories are intended for use in applications where the local CPU can alter memory contents. As such, manufacture and device codes must be accessible while the device resides in the target system. PROM programmers typically access the signature codes by

Table 3 – Sector Address Table

	A ₂₀	A ₁₉	A ₁₈	A ₁₇	A ₁₆	Address Range
SA ₀	0	0	0	0	0	00000h-00FFFFh
SA ₁	0	0	0	0	1	01000h-01FFFFh
SA ₂	0	0	0	1	0	02000h-02FFFFh
SA ₃	0	0	0	1	1	03000h-03FFFFh
SA ₄	0	0	1	0	0	04000h-04FFFFh
SA ₅	0	0	1	0	1	05000h-05FFFFh
SA ₆	0	0	1	1	0	06000h-06FFFFh
SA ₇	0	0	1	1	1	07000h-07FFFFh
SA ₈	0	1	0	0	0	08000h-08FFFFh
SA ₉	0	1	0	0	1	09000h-09FFFFh
SA ₁₀	0	1	0	1	0	0A000h-0AFFFFh
SA ₁₁	0	1	0	1	1	0B000h-0BFFFFh
SA ₁₂	0	1	1	0	0	0C000h-0CFFFFh
SA ₁₃	0	1	1	0	1	0D000h-0DFFFFh
SA ₁₄	0	1	1	1	0	0E000h-0EFFFFh
SA ₁₅	0	1	1	1	1	0F000h-0FFFFh
SA ₁₆	1	0	0	0	0	10000h-10FFFFh
SA ₁₇	1	0	0	0	1	11000h-11FFFFh
SA ₁₈	1	0	0	1	0	12000h-12FFFFh
SA ₁₉	1	0	0	1	1	13000h-13FFFFh
SA ₂₀	1	0	1	0	0	14000h-14FFFFh
SA ₂₁	1	0	1	0	1	15000h-15FFFFh
SA ₂₂	1	0	1	1	0	16000h-16FFFFh
SA ₂₃	1	0	1	1	1	17000h-17FFFFh
SA ₂₄	1	1	0	0	0	18000h-18FFFFh
SA ₂₅	1	1	0	0	1	19000h-19FFFFh
SA ₂₆	1	1	0	1	0	1A000h-1AFFFFh
SA ₂₇	1	1	0	1	1	1B000h-1BFFFFh
SA ₂₈	1	1	1	0	0	1C000h-1CFFFFh
SA ₂₉	1	1	1	0	1	1D000h-1DFFFFh
SA ₃₀	1	1	1	1	0	1E000h-1EFFFFh
SA ₃₁	1	1	1	1	1	1F000h-1FFFFh

raising A₉ to a high voltage. However, multiplexing high voltage onto the address lines is not generally a desirable system design practice.

The device contains an autoselect command operation to supplement traditional PROM programming methodology. The operation is initiated by writing the autoselect command sequence into the command register. Following the command write, a read cycle from address XX00H retrieves the manufacturer code of 01H. A read cycle from address XX01H returns the device code ADH (see table 2).

All manufacturer and device codes will exhibit odd parity

Table 4 – Sector Group Addresses

	A ₂₀	A ₁₉	A ₁₈	Sectors
SGA ₀	0	0	0	SA ₀ -SA ₃
SGA ₁	0	0	1	SA ₄ -SA ₇
SGA ₂	0	1	0	SA ₈ -SA ₁₁
SGA ₃	0	1	1	SA ₁₂ -SA ₁₅
SGA ₄	1	0	0	SA ₁₆ -SA ₁₉
SGA ₅	1	0	1	SA ₂₀ -SA ₂₃
SGA ₆	1	1	0	SA ₂₄ -SA ₂₇
SGA ₇	1	1	1	SA ₂₈ -SA ₃₁

with D₇ defined as the parity bit.

Furthermore, the write protect status of sectors can be read in this mode. Scanning the sector group addresses (A₁₈, A₁₉, and A₂₀) while (A₆, A₁, A₀) = (0, 1, 0) will produce a logical “1” at device output D₀ for a protected sector group.

To terminate the operation, it is necessary to write the read/reset command sequence into the register.

Byte Programming

The device is programmed on a byte-by-byte basis. Programming is a four bus cycle operation. There are two “unlock” write cycles. These are followed by the program set-up command and data write cycles. Addresses are latched on the falling edge of CS# or WE#, whichever occurs later, while the data is latched on the rising edge of CS# or WE#, whichever occurs first. The rising edge of CS# or WE# (whichever occurs first) begins programming using the Embedded Program Algorithm. Upon executing the algorithm, the system is not required to provide further controls or timings. The device will automatically provide adequate internally generated program pulses and verify the programmed cell margin.

The automatic programming operation is completed when the data on D₇ (also used as Data# Polling) is equivalent to the data written to this bit at which time the device returns to the read mode and addresses are no longer latched (see Table 6, Write Operation Status). Therefore, the device requires that a valid address to the device be supplied by the system at this time. Data# Polling must be performed at the memory location which is being programmed.

Any commands written to the chip during the Embedded Program Algorithm will be ignored. If a hardware reset occurs during the programming operation, the data at that particular location will be corrupted.

Programming is allowed in any address sequence and across sector boundaries. Beware that a data “0” cannot be programmed back to a “1”. Attempting to do so may cause the device to exceed programming time limits

Table 5 – Command Definitions

Command Sequence (Note 1)		Cycles	Bus Cycles (Notes 2-4)											
			First		Second		Third		Fourth		Fifth		Sixth	
			Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read (Note 5)		1	RA	RD										
Read (Note 6)		1	XXX	F0										
Autoselect (Note 7)	Device ID	4	5555	AA	2AAA	55	5555	90	X00	01				
	Sector Group Protect Verify (Note 8)	4	5555	AA	2AAA	55	5555	90	SGA	XX00				
X03									XX01					
Program		4	5555	AA	2AAA	55	5555	A0	PA	PD				
Chip Erase		6	5555	AA	2AAA	55	5555	80	5555	AA	2AAA	55	5555	10
Sector Erase		6	5555	AA	2AAA	55	5555	80	5555	AA	2AAA	55	SA	30
Erase Suspended (Note 9)		1	XXX	B0										
Erase Resume (Note 10)		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 latch on the falling edge of the *WE#* or *CS#* pulse, whichever happens later.
PD = Data to be programmed at location *PA*. Data latches on the rising edge of the *WE#* or *CS#* pulse, whichever happens first.
SA = Address of the sector to be verified (in autoselect mode) or erased. Address bits A20-16 select a unique sector.
SGA = Address of the sector group to be verified. Address bits A20-18 select a unique sector group.

NOTES:

- See Table 1 for description of bus operations.
- All values are in hexadecimal.
- Except when reading array or autoselect data, all bus cycles are write operation.
- Address bit A20-11 = Don't Care for unlock and command cycles, unless Sector Address (*SA*) or Program Address (*PA*) required.
- No unlock or command cycles required when reading array data.
- The Reset Command is required to return to reading array data when device is in the autoselect mode, or if *D5* goes high (while the device is providing status data).
- The fourth cycle of the autoselect command sequence is a read cycle.
- The data is 00h for an unprotected sector group and 01h for a protected sector group. See "Autoselect Command Sequence" for more information.
- The system may read and program in non-erasing sectors, or enter the autoselect mode, when in Erase Suspend mode. The Erase Suspend command is valid only during a sector erase operation.
- The Erase Resume command is valid only during the Erase Suspend mode.

(*D5* = 1) or result in an apparent success, according to the data polling algorithm, but a read from reset/read mode will show that the data is still "0". Only erase operations can convert "0"s to "1"s.

Figure 1 illustrates the Embedded Programming Algorithm using typical command strings and bus operations.

Chip Erase

Chip erase is a six bus cycle operation. There are two "unlock" write cycles. These are followed by writing the "set-up" command. Two more "unlock" write cycles are then followed by the chip erase command.

Chip erase does not require the user to program the device prior to erase. Upon executing the Embedded Erase Algorithm command sequence the device automatically will program and verify the entire memory for an all zero data pattern prior to electrical erase. The erase is performed concurrently one sector at a time (see

Table "Erase and Programming Performance" for erase times). The system is not required to provide any controls or timings during these operations.

The automatic erase begins on the rising edge of the last *WE#* pulse in the command sequence and terminates when the data in *D7* is "1" (see Write Operation Status section) at which time the device returns to read the mode.

Figure 2 illustrates the Embedded Erase Algorithm using typical command strings and bus operations.

Sector Erase

Sector erase is a six bus cycle operation. There are two "unlock" write cycles. These are followed by writing the "set-up" command. Two more "unlock" write cycles are then followed by the sector erase command. The sector address (any address location within the desired sector) is latched on the falling edge of *WE#*, while the command (30H) is latched on the rising edge of *WE#*. After a

Table 6 – Write Operation Status

Status		D7	D6	D5	D3	D2	
In Progress	Byte Program in Embedded Program Algorithm	D7#	Toggle	0	0	1	
	Embedded Erase Algorithm	0	Toggle	0	1	Toggle	
	Erase Suspended Mode	Erase Suspend Read (Erase Suspended Sector)	1	1	0	0	Toggle(1)
		Erase Suspend Read (Non-Erase Suspended Sector)	Data	Data	Data	Data	Data
	Erase Suspend Program (Non-Erase Suspended Sector)	D7#	Toggle(2)	0	0	1(3)	
Exceeded Time Limit	Byte Program in Embedded Program Algorithm	D7#	Toggle	1	0	1	
	Program/Erase in Embedded Erase Algorithm	0	Toggle	1	1	N/A	
	Erase Suspended Mode	Erase Suspend Program (Non-Erase Suspended Sector)	D7#	Toggle	1	1	N/A

NOTES:

1. Performing successive read operations from erase-suspended sector will cause D₂ to toggle.
2. Performing successive read operations from any address will cause D₆ to toggle.
3. Reading the byte address being programmed while in the erase-suspend program mode will indicate logic '1' at the D₂ bit. However, successive reads from the erase-suspended sector will cause D₂ to toggle.

time-out of 50µs from the rising edge of the last sector erase command, the sector erase operation will begin.

Multiple sectors may be erased concurrently by writing the six bus cycle operations as described above. This sequence is followed with writes of the Sector Erase command to addresses in other sectors desired to be concurrently erased. The time between writes must be less than 50µs otherwise that command will not be accepted and erasure will start. It is recommended that processor interrupts be disabled during this time to guarantee this condition. The interrupts can be re-enabled after the last Sector Erase command is written. A time-out of 50µs from the rising edge of the last WE# will initiate the Sector Erase command(s). If another falling edge of the WE# occurs within the 50µs time-out window the timer is reset. (Monitor D₃ to determine if the sector erase timer window is still open, see section D₃, Sector Erase Timer.) Any command other than Sector Erase or Erase Suspend during this period will reset the device to the read mode, ignoring the previous command string. In that case, restart the erase on those sectors and allow them to complete.

(Refer to the Write Operation Status section for D₃, Sector Erase Timer, operation.) Loading the sector erase buffer may be done in any sequence and with any number of sectors (0 to 31).

Sector erase does not require the user to program the device prior to erase. The device automatically programs all memory locations in the sector(s) to be erased prior to electrical erase. When erasing a sector or sectors the remaining unselected sectors are not affected. The system is not required to provide any controls or timings during these operations.

The automatic sector erase begins after the 50µs time-out from the rising edge of the WE# pulse for the last sector

erase command pulse and terminates when the data on D₇, Data# Polling, is "1" (see Write Operation Status section) at which time the device returns to the read mode. Data# Polling must be performed at an address within any of the sectors being erased.

Figure 2 illustrates the Embedded Erase Algorithm using typical command strings and bus operations.

Erase Suspend

The Erase Suspend command allows the user to interrupt a Sector Erase operation and then perform data reads or programs to a sector not being erased. This command is applicable ONLY during the Sector Erase operation which includes the time-out period for sector erase. The Erase Suspend command will be ignored if written during the Chip Erase operation or Embedded Program Algorithm. Writing the Erase Suspend command during the Sector Erase time-out results in immediate termination of the time-out period and suspension of the erase operation.

Any other command written during the Erase Suspend mode will be ignored except the Erase Resume command. Writing the Erase Resume command resumes the erase operation. The addresses are "don't-care" when writing the Erase Suspend or Erase Resume command.

When the erase Suspend command is written during the Sector Erase operation, the device will take a maximum of 15µs to suspend the erase operation. When the device has entered the erase-suspended mode, the RY/BY# output pin and the D₇ bit will be at logic "1", and D₆ will stop toggling. The user must use the address of the erasing sector for reading D₆ and D₇ to determine the erase operation has been suspended. Further writes of the Erase Suspend command are ignored.

When the erase operation has been suspended, the

device defaults to the erase-suspend-read mode. Reading data in this mode is the same as reading from the standard read mode except that the data must be read from sectors that have not been erase-suspended. Successively reading from the erase-suspend sector while the device is in the erase-suspend-read mode will cause D₂ to toggle. (See the section on D₂).

After entering the erase-suspend-read mode, the user can program the device by writing the appropriate command sequence for Byte Program. This program mode is known as the erase-suspend-program mode. Again, programming in this mode is the same as programming in the regular Byte Program mode except that the data must be programmed to sectors that are not erase-suspend program mode will cause D₂ to toggle. The end of the erase-suspended program operation is detected by the RY/BY# output pin, Data# Polling of D₇, or the Toggle Bit I (D₆) which is the same as the regular Byte Program operation. Note that D₇ must be read from the byte program address while D₆ can be read from any address.

To resume the operation of Sector Erase, the Resume command (30H) should be written. Any further writes of the Resume command at this point will be ignored. Another Erase Suspend command can be written after the chip has resumed erasing.

WRITE OPERATION STATUS

D₇

Data# Polling

The 16M5 device features Data# Polling as a method to indicate to the host that the embedded algorithms are in progress or completed. During the Embedded Program Algorithm, an attempt to read the device will produce the compliment data of the data last written to D₇. Upon completion of the Embedded Program Algorithm an attempt to read the device will produce the true data last written to D₇. During the Embedded Erase Algorithm, an attempt to read the device will produce a “0” at the D₇ output. Upon completion of the Embedded Erase Algorithm an attempt to read the device will produce a “1” at the D₇ output. The flowchart for Data# Polling (D₇) is shown in Figure 4.

Data# Polling will also flag the entry into Erase Suspend. D₇ will switch “0” to “1” at the start of the Erase Suspend mode. Please note that the address of an erasing sector must be applied in order to observe D₇ in the Erase Suspend mode.

During Program in Erase Suspend, Data# Polling will perform the same as in regular program execution outside of the suspend mode.

For chip erase, the Data# Polling is valid after the rising edge of the sixth WE# pulse in the six write pulse sequence.

For sector erase, the Data# Polling is valid after the last rising edge of the sector erase WE# pulse. Data# Polling must be performed at a sector address within any of the sectors being erased and not a protected sector. Otherwise, the status may not be valid.

Just prior to completion of Embedded Algorithm operations, data pins (D₇) may change asynchronously while the output enable (OE#) is asserted low. This means that the device is driving status information on D₇ at one instance of time and then that byte's valid data at the next instant of time. Depending on when the system samples the D₇ output, it may read the status or valid data. Even if the device has completed internal algorithm operation and D₇ has a valid data, the data outputs on D₀₋₆ may be still invalid. The valid data on D₀₋₇ will be read on the successive read attempts. The Data# Polling feature is only active during the Embedded Programming Algorithm, Embedded Erase Algorithm, Erase Suspend, erase-suspend-program mode, or sector erase time-out (Table 6).

D₆

Toggle Bit I

The device also features the “Toggle Bit I” as a method to indicate to the host system that the embedded algorithms are in progress or completed.

During an Embedded Program or Erase Algorithm cycle, successive attempts to read (OE# toggling) data from the device at any address will result in D₆ toggling between one and zero. Once the Embedded Program or Erase Algorithm cycle is completed, D₆ will stop toggling and valid data will be read on the next successive attempt. During programming, the Toggle Bit I is valid after the rising edge of the fourth WE# pulse in the four write pulse sequence. For chip erase the Toggle Bit I is valid after the rising edge of the sixth WE# pulse in the six write pulse sequence. For Sector Erase, the Toggle Bit I is valid after the last rising edge of the sector erase WE# pulse. The Toggle Bit I is active during the sector time out.

Either CS# or OE# toggling will cause D₆ to toggle. In addition, an Erase Suspend/Resume command will cause D₆ to toggle. See Figure 4, Toggle Bit I timing specifications and diagrams.

D₅

Exceeded Timing Limits

D₅ will indicate if the program or erase time has exceeded the specified limits (internal pulse count). Under these conditions D₅ will produce a “1”, which indicates that the program or erase cycle was not successfully completed. Data# Polling is the only operating function of the device under this condition. The CS# circuit will partially power down the device under these conditions to approximately

2mA per chip. The OE# and WE# pins will control the output disable functions as shown in Table 1.

The D₅ failure condition will also appear if a user tries to program a 1 to a location that is previously programmed to 0. In this case the device locks out and never completes the Embedded Program Algorithm. Hence, the system never reads a valid data on D₇ bit and D₆ never stops toggling. Once the device has exceeded timing limits, the D₅ bit will indicate a “1”. Please note that this is not a device failure condition since the device was incorrectly used. If this occurs, reset the device.

D₃

Sector Erase Timer

After the completion of the initial sector erase command sequence the sector erase time-out will begin. D₃ will remain low until the time-out is complete. Data# Polling and Toggle Bit I are valid after the initial sector erase command sequence.

If Data# Polling or the Toggle Bit I indicates the device has been written with a valid erase command, D₃ may be used to determine if the sector erase timer window is still open. If D₃ is high (“1”) the internally controlled erase cycle has begun; attempts to write subsequent commands (other than Erase Suspend) to the device will be ignored until the erase operation is completed as indicated by Data# Polling or Toggle Bit I. If D₃ 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 D₃ prior to and following each subsequent sector erase command. If D₃ were high on the second status check, the command may not have been accepted. See Table 6, Write Operation Status.

D₂

Toggle Bit II

This toggle bit, along with D₆, can be used to determine whether the device is in the Embedded Erase Algorithm or in Erase Suspend.

Successive reads from the erasing sector will cause D₂ to toggle during the Embedded Erase Algorithm. If the device is in the erase-suspend-read mode, successive reads from the erase-suspend sector will cause D₂ to toggle. When the device is in the erase-suspended-program mode, successive reads from the byte address of the non-erase suspended sector will indicate a logic ‘1’ at the D₂ bit.

D₆ is different from D₂ in that D₆ toggles only when the standard Program or Erase, or Erase Suspend Program operation is in progress. The behavior of these two status bits, along with that of D₇, is summarized as follows:

Mode	D ₇	D ₆	D ₂
Program	D ₇ #	toggles	1
Erase	0	toggles	toggles
Erase Suspend Read (1) (Erase-Suspended Sector)	1	1	toggles
Erase Suspend Program	D ₇ # (2)	toggles	toggles

NOTES:

1. These status flags apply when outputs are read from a sector that has been erase-suspended.
2. These status flags apply when outputs are read from the byte address of the non-erase-suspended sector.

For example, D₂ and D₆ can be used together to determine the erase-suspend-read mode (D₂ toggles while D₆ does not). See also Table 6.

Furthermore, D₂ can also be used to determine which sector is being erased. When the device is in the erase mode, D₂ toggles if this bit is read from the erasing sector.

RY/BY#

Ready/Busy

The 16M5 provides a RY/BY# open-drain output pin as a way to indicate to the host system that the Embedded Algorithms are either in progress or have been completed. If the output is low, the device is busy with either a program or erase operation. If the output is high, the device is ready to accept any read/write or erase operation. When RY/BY# pin is low, the device will not accept any additional program or erase commands with the exception of the Erase Suspend command. If the 16M5 is placed in an Erase Suspend mode, the RY/BY# output will be high.

During programming, the RY/BY# pin is driven low after the rising edge of the fourth WE# pulse. During an erase operation, the RY/BY# pin is driven low after the rising edge of the sixth WE# pulse. The RY/BY# pin will indicate a busy condition during the RESET# pulse.

Since this is an open-drain output, several RY/BY# pins can be tied together in parallel with a pull-up resistor to V_{CC}.

RESET#

Hardware Reset

The 16M5 device may be reset by driving the RESET# pin to V_{IL}. The RESET# pin must be kept low (V_{IL}) for at least 500ns. Any operation in progress will be terminated and the internal state machine will be reset to the read mode 200ns after the RESET# pin is driven low. If a hardware reset occurs during a program or erase operation, the data at that particular location will be indeterminate.

When the RESET# pin is low and the internal reset is complete, the device goes to standby mode and cannot be accessed. Also, note that all the data output pins are tri-stated for the duration of the RESET# pulse. Once the RESET# pin is taken high, the device requires 500ns of

wake up time until outputs are valid for read access.

The RESET# pin may be tied to the system reset input. Therefore, if system reset occurs during the Embedded Program or Erase Algorithm, the device will be automatically reset to read mode and this will enable the system's microprocessor to read the boot-up firmware from the Flash memory.

DATA PROTECTION

The 16M5 is designed to offer protection against accidental erasure or programming caused by spurious system level singles that may exist during power transitions. During power up the device automatically resets the internal state machine in the read mode. Also, with its control register architecture, alteration of the memory contents only occurs after successful completion of specific multi-bus cycle command sequences.

The device also incorporates several features to prevent inadvertent write cycles resulting from V_{CC} power-up and power-down transitions or system noise.

Low V_{CC} Write Inhibit

To avoid initiation of a write cycle during V_{CC} power-up and power-down, a write cycle is locked out for $V_{CC} < V_{LKO}$ (See DC Characteristics section for voltages). When $V_{CC} < V_{LKO}$, the command register is disabled and all internal

program/erase circuits are disabled. Under this condition the device will reset to read mode. Subsequent writes will be ignored until the $V_{CC} > V_{LKO}$. The user must ensure that the control pins are in the correct logic state when $V_{CC} > V_{LKO}$ to prevent unintentional writes.

Write Pulse Glitch Protection

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

Logical Inhibit

Writing is 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 logical zero while OE# is a logical one.

Power-Up Write Inhibit

Power-up of the device with WE# = CS# = V_{IL} and OE# = V_{IH} will not accept commands on the rising edge of WE#. The internal state machine is automatically reset to the read mode on power-up.

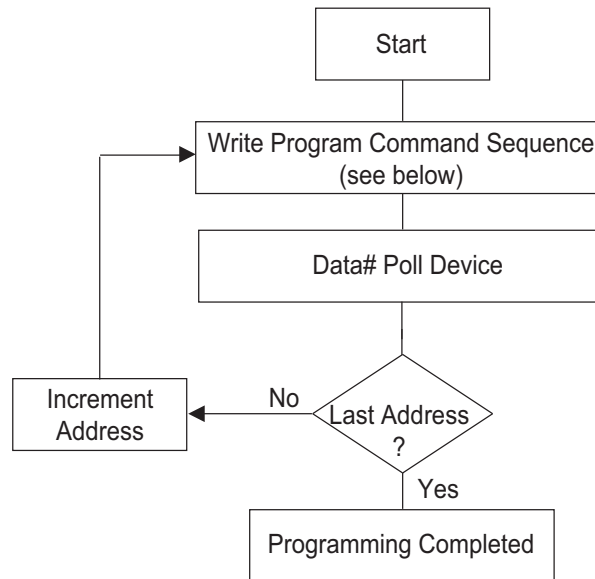
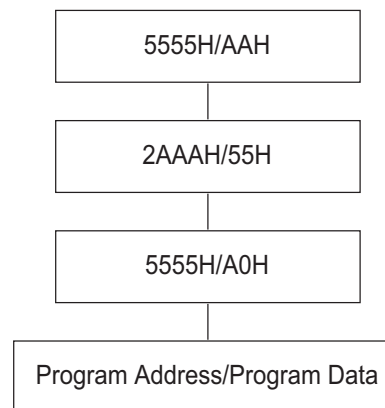
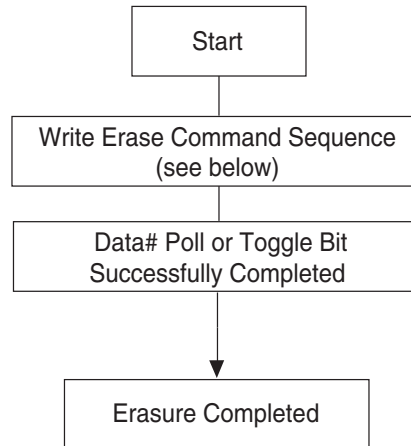
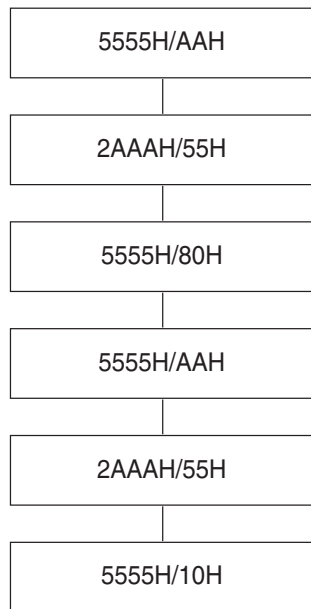
Figure 1 – Programming Algorithm

Program Command Sequence (Address/Command):


Figure 2 – Erase Algorithm

**Chip Erase Command Sequence
(Address/Command):**

**Individual Sector/Multiple Sector
Erase Command Sequence
(Address/Command):**


Additional sector
erase commands
are optional

1. D_6 is rechecked even if $D_5 = 1$ because D_7 may change simultaneously with D_5 .

Figure 4 – Toggle Bit Algorithm

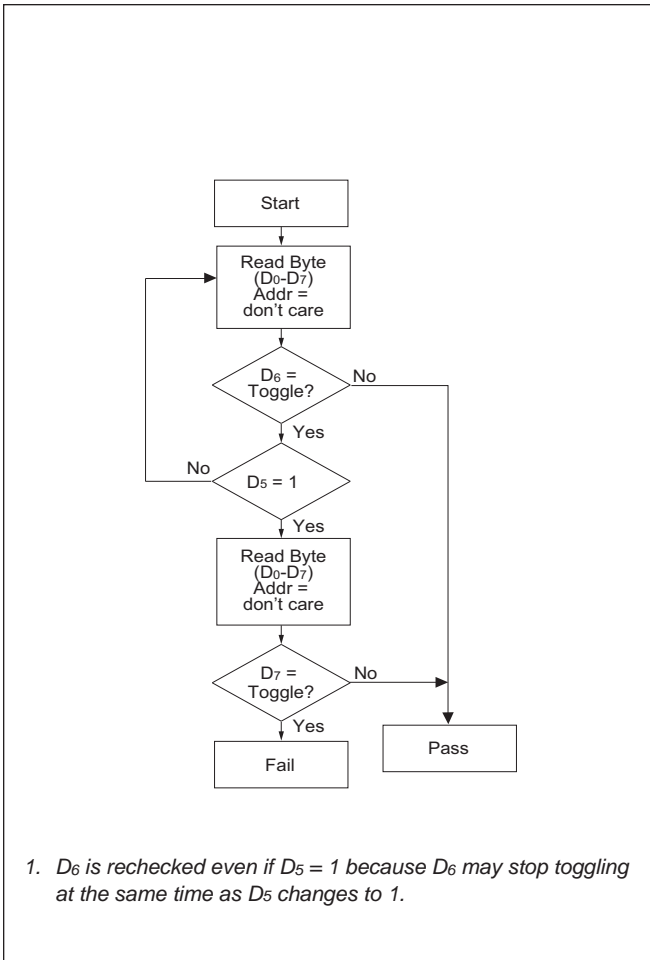


Figure 5 – Data# Polling Algorithm

