MirrorBit™ Flash Memory Write Buffer Programming and Page Buffer Read

Application Note

July 2003

The following document refers to Spansion memory products that are now offered by both Advanced Micro Devices and Fujitsu. Although the document is marked with the name of the company that originally developed the specification, these products will be offered to customers of both AMD and Fujitsu.

Continuity of Specifications

There is no change to this document as a result of offering the device as a Spansion product. Any changes that have been made are the result of normal documentation improvements and are noted in the document revision summary, where supported. Future routine revisions will occur when appropriate, and changes will be noted in a revision summary.

Continuity of Ordering Part Numbers

AMD and Fujitsu continue to support existing part numbers beginning with "Am" and "MBM". To order these products, please use only the Ordering Part Numbers listed in this document.

For More Information

Please contact your local AMD or Fujitsu sales office for additional information about Spansion memory solutions.

Publication Number **25539** Revision **A** Amendment **0** Issue Date **October 4, 2001**

MirrorBit™ Flash Memory Write Buffer Programming and Page Buffer Read

Application Note

INTRODUCTION

A Write Buffer is implemented in MirrorBit™ flash memory devices to speed up programming operations. A write buffer is a set of registers used to hold several words that are to be programmed as a group. The buffer is filled with words to be programmed before issuing the write buffer programming command. The time to program each word is reduced by performing programming overhead operations once for the entire group of words. This results in faster effective word/byte programming time than the standard “word/byte” programming algorithms. Write Buffer Programming allows the system to write to a maximum of 16 words (32 bytes) in one programming operation.

Write Buffer Programming is performed through the use of a few new memory device commands. These are in addition to the commands normally used to control all AMD Flash devices. Please refer to an AMD Flash datasheet for a review of the full command set, the method for issuing commands, and the method for polling the status of memory during a command operation.

WRITE BUFFER OPERATION

Write-Buffer Programming is only available through the “Write to Buffer” and “Program Buffer to Flash” confirm command sequences. The “Write-to-Buffer Abort Reset” command sequence is used to reset out of the Write-Buffer-Abort state. Table 1 lists all software program sequences associated with the Write-Buffer.

Table 1. MirrorBit™ Write Buffer Programming Command Definitions x8/x16 Devices

		Bus Cycles											
		First		Second		Third		Fourth		Fifth		Sixth	
Command Sequence	**Interface**	**Addr**	**Data**	**Addr**	**Data**	**Addr**	**Data**	**Addr**	**Data**	**Addr**	**Data**	**Addr**	**Data**
Write to Buffer	Word	555	AA	2AA	55	SA	25	SA	WC	PA	PD	WBL (Note)	PD
	Byte	AAA		555									
Program Buffer to Flash (Confirm)	Both	SA	29										
Write-to-BufferAbort Reset	Word	555	AA	2AA	55	XXX	F0						
	Byte	AAA		555									

Note: *The sixth cycle must be repeated to complete the number of buffer writes specified by WC in cycle four.*

PA = Program Address of the memory location to be programmed. This can be any address within the target Write-Buffer-Page.

PD = Program Data to be programmed at location PA.

SA = Sector Address containing locations to be programmed. This can be any valid address within the sector.

WC = Write Count is the number of write buffer locations to load minus one.

WBL = Write Buffer Location. The address must be within the same write buffer page (32 byte range located on a 32 byte boundary) as PA.

Publication# **25539** Rev: **A** Amendment/**0**
Issue Date: **October 4, 2001**

In Table 1, the user starts loading data at any location in the target write-buffer-page. Subsequent write buffer locations do not need to be loaded in any particular order as long as they reside in the same write-buffer-page.

Note that the internal write counter decrements for every data load operation, not for each unique write-buffer address location. If the same write-buffer-location is loaded multiple times, the internal write counter will decrement after each load operation. The last data loaded into a given write-buffer location will be programmed into the device after the “Program Buffer to Flash” confirm command. It is the software's responsibility to comprehend the ramifications of loading a write-buffer location more than once.

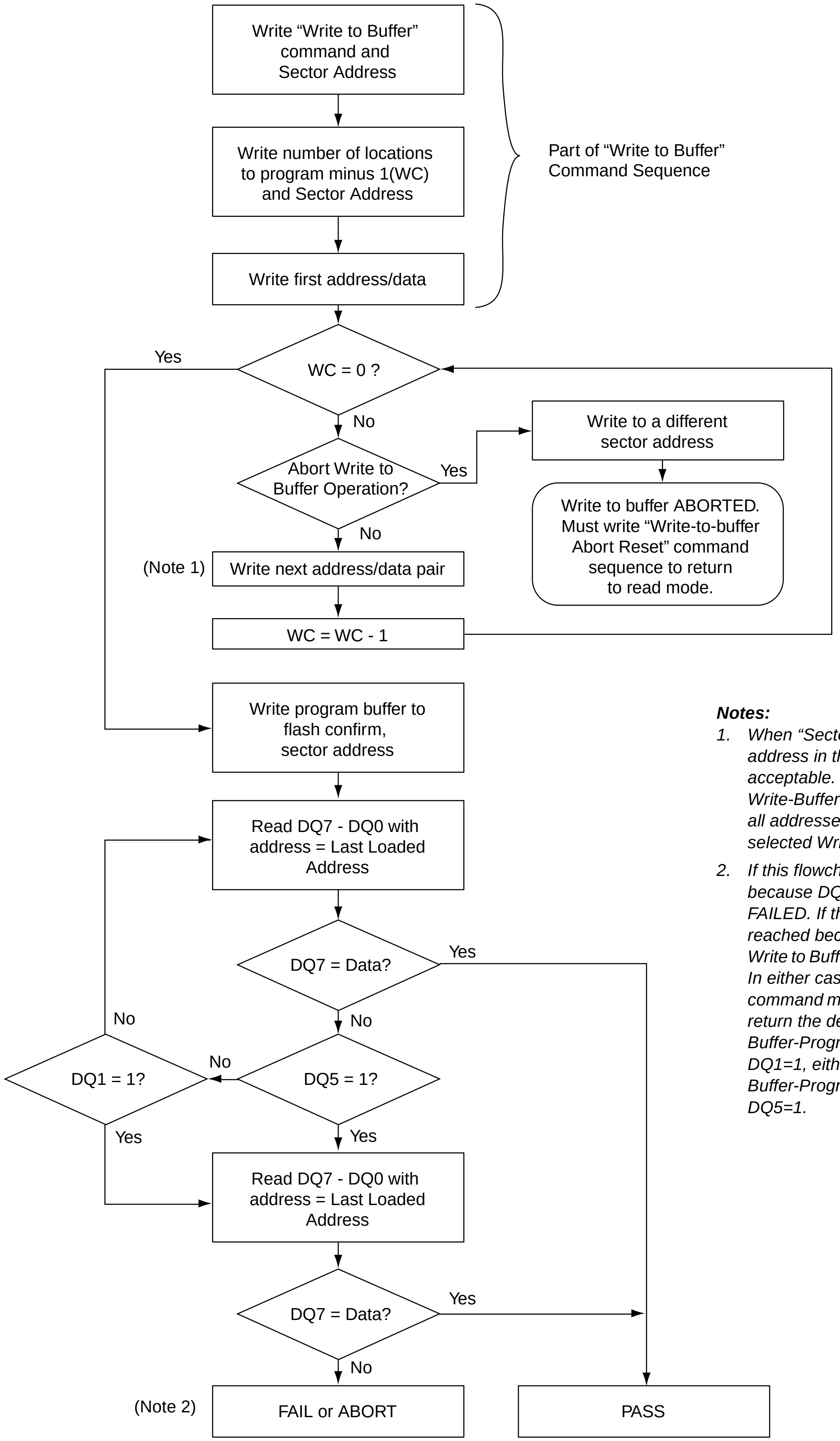
When the “Write to Buffer” command programming sequence has been completed, the “Program Buffer to Flash” confirm command must be issued to move the data from the write-buffer into the flash memory array.

Programming Steps

Table 2. Table 2. MirrorBit™ Write Buffer Programming Procedure

Cycle #	Write Buffer Program Sequence	Address	Data	Comment
1, 2	Issue Two Unlock Cycles:Unlock 1, Unlock 2			Refer to “Write-to-Buffer” Software Command Definition for first and second bus cycles
3	Issue “Write-Buffer-Load” Command @ Sector Address	SA	0025h	Sector Address is issued starting with the third bus cycle
4	Issue Number of Write Buffer Locations to load minus one@ Sector Address	SA	WC	WC = number of locations to program minus 1WC of 0 = 1 location to pgmWC of 1 = 2 locations to pgm, etc.The Word Count is issued during the fourth bus cycle
5	Load First Address/Data pair	PA	PD	Selects Write-Buffer-Page and loads first Address/Data Pair. The first address location can be any location in the target Write-Buffer-Page.The first address is loaded into the write-buffer during the fifth bus cycle
6 to N	Load remaining Address/Data pairs into write buffer	WBL	PD	All addresses MUST be within the selected write-buffer-page boundaries but do not have to be in any order.Number of cycles depends on the cycle count loaded in fourth bus cycle.
N + 1	Issue Write Buffer Program Confirm@ Sector Address	SA	0029h	This command MUST follow the last write buffer location loaded, or the operation will ABORT.
	Perform Data Bar Polling on Last Loaded Address			

A flowchart for the “Write to Buffer” command sequence is demonstrated in Figure 1.


Write "Write to Buffer" command and Sector Address
Write number of locations to program minus 1(WC) and Sector Address
Write first address/data
Part of "Write to Buffer" Command Sequence
Yes
WC = 0 ?
No
Abort Write to Buffer Operation?
Yes
Write to a different sector address
Write to buffer ABORTED. Must write "Write-to-buffer Abort Reset" command sequence to return to read mode.
No
(Note 1)
Write next address/data pair
WC = WC - 1
Write program buffer to flash confirm, sector address
Read DQ7 - DQ0 with address = Last Loaded Address
DQ7 = Data?
Yes
No
No
DQ1 = 1?
No
DQ5 = 1?
Yes
Yes
Read DQ7 - DQ0 with address = Last Loaded Address
DQ7 = Data?
Yes
No
(Note 2)
FAIL or ABORT
PASS

Notes:

1. *When "Sector Address" is specified, any address in the selected sector is acceptable. However, when loading Write-Buffer address locations with data, all addresses MUST fall within the selected Write-Buffer Page.*
2. *If this flowchart location was reached because DQ5= "1", then the device FAILED. If this flowchart location was reached because DQ1= "1", then the Write to Buffer operation was ABORTED. In either case, the proper RESET command must be written to the device to return the device to READ mode. Write-Buffer-Programming-Abort-Reset if DQ1=1, either Software RESET or Write-Buffer-Programming-Abort-Reset if DQ5=1.*

Figure 1. Write Buffer Programming Operation

Write Buffer Programming Abort

A "write-buffer-page" is selected by addresses A4-A(max) for x16 or x8/x16 flash memory devices or by addresses A5-A(max) for x8 flash memory devices. The "write-buffer-page" addresses must be the same for all addresses loaded during a write buffer programming operation. Write Buffer Programming cannot be performed across multiple "write-buffer-pages" or across multiple sectors. If the above conditions are violated, the Write Buffer Programming operation will be automatically aborted.

Listed below are the ways in which the Write Buffer Programming sequence can be automatically aborted:

1. *Loading a value that is greater than the write buffer size (write-buffer-page) during the "Numbers of Locations to Program" step.*
2. *Writing to an address in a sector that is different than the one specified during the "Write-Buffer-Load" command.*
3. *Writing an Address/Data pair to a different write-buffer-page than the one selected by the "Starting Address" during the "write buffer data loading" stage of the operation.*
4. *Writing data other than the "Confirm Command" after loading the specified number of write buffer locations.*

Note that the "Write-to-Buffer Abort" condition is always indicated by the DQ1 "Write-To-Buffer Abort" Operation Status Bit.

DQ1: Write-to-Buffer Abort

DQ1 is "1" when a Write-to-Buffer operation has been aborted.

A Write-to-Buffer-Abort-Reset command sequence must be issued to return the flash memory device to reading array data.

The ABORT condition is indicated by DQ1 = 1, DQ7 = DATA# (for the "Last Loaded Address"), DQ6 = TOGGLE, DQ5=0. This indicates that the Write Buffer Programming Operation was ABORTED. A "Write-to-Buffer-Abort Reset" command sequence must be written to the device to return to READ mode.

PAGE BUFFER READ INTRODUCTION

Whenever the system changes a "page address" (or toggles CE# during a read) the device performs a "random access". During this "random access" the read page buffer is loaded in parallel with data within the selected read-page boundaries. Subsequent "intra-page" accesses are 3 to 4 times faster than random accesses because the data are already available in the buffer (please refer to differences between t_{CE}/t_{ACC} and t_{PACC} in the Am29LVxxxM datasheet). Therefore, read performance is significantly improved.

READ BUFFER OPERATION

For page buffer read operation, the user has to issue a read address, or "RA", for any memory location. During the initial access time (t_{CE}/t_{ACC}) a page of 4 words (8 bytes), located on an 8-byte boundary, is read into the page buffer. If the device is in word mode address bits A1 and A0 can then be used to access any of the four words within the page with a reduced page access time (t_{PACC}). If the device is in byte mode in a x8/x16 device. A1 through A-1 can be used to access any of the eight bytes in the page. If the device is a x8-only device, A2 through A0 can be used to access any of the eight bytes within the page.

The appropriate page is selected by the higher address bits A2-A(max) for x16-only and x8/x16 devices, and A3-A(max) for x8-only devices. Fast page mode accesses are obtained by keeping the high-order "read-page address bits" constant and changing the "intra-read page address bits" addresses: A0 to A1 for x16-only and x8/x16 in word mode; A-1 to A1 for x8/x16 in byte mode; and A0 to A2 for x8-only. This is an asynchronous operation with the microprocessor supplying the specific byte or word location.

A depiction of the command sequence definition for read accesses is shown in Table 3.

A depiction of the device bus operation for read accesses is shown in Table 4.

Table 3. Read Access

Command Sequence	Interface	Bus Cycles									
		First		Second		Third		Fourth		Fifth	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read	Both	RA	RD	RA	RD	RA	RD	RA	RD	Note	Note

Note: *For reading bytes, eight consecutive memory locations can be read, compared to four memory locations for reading words. "Intra-read page" locations can be accessed in any order.*

RA = Read Address

RD = Read Data

Table 4. Device Bus Operation for Read Access

Operation	CE#	OE#	WE#	RESET#	WP#/ACC	Address	Data
Read	L	L	H	H	X	A_{IN}	D_{OUT}

During page buffer read operations, the CE# pin must be kept at voltage level V_{IL} during all fast page mode accesses. If the CE# pin toggles or changes state during a page buffer read operation, the current data transfer will automatically be aborted and another initial page access is started. This will result in unnecessary penalty and overhead in read timings.

CONCLUSION

The Write Buffer Programming feature of AMD MirrorBit™ Flash memories increases the programming speed by roughly 16 times compared to single byte or word write operations in the same memory for a full write buffer of 16 words or 32 bytes. Write Buffer Programming performance is roughly two times faster than previous AMD Low Voltage Flash Memories. Write Buffer Programming is enabled via a simple addition of three commands to the standard AMD embedded algorithm bus command set.

The Read Page Buffer feature of AMD MirrorBit Flash memories can increase read performance significantly. Following each random (inter-page) access all locations of the referenced 8-byte page are available for fast access. When read accesses can be grouped within a page the average read performance can be increased by 3 to 4 times.