PRELIMINARY

AMD

AmMC0XXA

2, 4, or 8 Megabyte 5.0 Volt-only Flash Miniature Card

DISTINCTIVE CHARACTERISTICS

- **2, 4, or 8 MBytes of addressable Flash memory**
- **5.0 Volt-only, single power supply operation**
  - Write and read voltage: 5.0 V ± 10%
  - No additional supply current required for V_{PP}
- **Fast access time**
  - 150 ns maximum access time
- **CMOS low power consumption**
  - Typical active read current: 50 mA (word mode)
  - Typical active erase/write current: 60 mA (word mode)
  - Typical standby current: 50 µA (8 MByte card)
- **High write endurance**
  - Guaranteed minimum 100,000 write/erase cycles per card
  - More than 1,000,000 cycles per card typical
- **Uniform sector architecture**
  - 64K byte individually useable sectors
  - Erase Suspend/Resume increases system level performance
  - $\overline{\text{BUSY}}$ (RY/$\overline{\text{BY}}$, Ready/$\overline{\text{Busy}}$) and $\overline{\text{RESET}}$ signals
- **Zero data retention power**
  - Batteries not required for data storage
- **Miniature Card standard form factor**
  - True interchangeability
  - 60-pad elastomeric connector
  - Supports multiple technologies
  - Sonic welded stainless steel case
  - PCMCIA Type II adapter available
  - Selectable byte- or word-wide configuration
- **60 connection bus**
  - 16-bit data bus
  - 25-bit address bus
  - Easy system integration
  - Low cost implementation
  - Low cost cards
- **Small Form Factor (38 mm x 33 mm x 3.5 mm)**
  - Minimized system area consumed by card
  - 16-bit non-multiplexed data bus addressing up to 8 MBytes
- **Consumer-friendly mechanicals**
  - User can easily insert and remove card, upgrade memory, and add applications
- **Voltage level keying**
  - Does not allow a 5 V card to plug into a 3 V system and vice versa
  - Single power supply design
  - System does not need a separate program voltage supply; only one is necessary to read and write

GENERAL DESCRIPTION

The Miniature Card is an expansion card that provides a high-performance, small form factor solution for data and file storage to the portable, handheld market, which includes audio, digital film, wireless, and PDA (Portable Digital Assistant) applications. The Miniature Card provides a low cost, low power, high performance interface for memory cards.

Miniature cards can be easily "snapped" into the back of an electronic system and can be readily removed and replaced by end users. AMD's 5 V Flash Miniature Cards are manufactured using AMD's industry leading 5.0 volt-only, single-power-supply Am29F080 and Am29F016 Flash Memory devices, ensuring high reliability and excellent performance. The Miniature Card is less than 30% of the size of a PCMCIA memory card. Applications include digital voice recorders, pocket PCs and intelligent organizers, Smart cellular telephones, voice and data messaging pagers, digital still cameras and portable instrumentation.

The Miniature Card specification is defined by the Miniature Card Implementers Forum. The forum members include major Flash memory vendors and leading consumer electronics OEMs. The goal of the

Publication# **20975** Rev: **C** Amendment/**0**
Issue Date: **January 1997**

forum is to promote an open, interoperable small-form-factor memory card standard. For more information on the Miniature Card specification, visit their web site at http://www.mcif.org.

AMD Flash Miniature Cards can be read in either a byte-wide or word-wide mode, which allows for flexible integration into various system platforms. Compatibility is assured at the hardware interface and software interchange specification.

Miniature Card is also designed with low-cost and rugged handling in mind. The card contains virtually no control logic, which keeps cost and power consumption to a minimum. The Miniature Card is packaged in a sonic welded, stainless steel case that guarantees durability, provides good ESD protection and ease of handling. The Miniature Card connects to the system via a state-of-the-art elastomeric connector. The elastomeric connector is based on a silicon rubber substrate with silver conductors, and offers a low-profile, low cost and reliable connection between the memory card and the system. Unlike alternative pin and socket solutions, the elastomer can be easily replaced if damaged and requires minimal space on the system motherboard.

Miniature Card has extensive third-party support, including socket and connector solution from Augat, software support from the major FTL software vendors, and PCMCIA adapter solutions and programmer support.

AMD's Miniature Flash cards can be used for both code and data storage. Since fast random access is possible, code can be directly executed from the card, reducing the amount of system RAM required. In addition. AMD's Flash technology offers unsurpassed endurance, data retention and reliability, eliminating the need for complex error correction and defect management hardware and software. Each Flash sector provides a minimum of 100,000 cycles, which translates into a typical card life of one million or more cycles.

For more information, please contact your local AMD sales office or visit our Web site at http://www.amd.com/html/products/nvd/nvd.html.

DEFINITIONS

Table 1 lists the terms and definitions that may be used in conjunction with Miniature Card specifications.

Table 1. Miniature Card Definitions

Term	Meaning
AIS	Acronym for Attribute Information Structure. AIS is a Miniature Card specification for storing Miniature Card attribute information.
Elastomeric connector	A connector that is made of alternating layers of conductive and non-conductive silicone rubber.
ESD	Acronym for Electrostatic Discharge. ESD is part of the Miniature Card physical test suite.
FAT	Acronym for File Allocation Table. A FAT is a common method for managing files. FAT is used in DOS based system.
Flash	A type of non-volatile memory that is both readable and writeable, but requires the media to be erased before it is rewritten.
Host	Any system that incorporates a Miniature Card socket
Insertion, Cold	*User Perception:* Insertion of the Miniature Card when the host is off. *Host State:* The host would be either off or in sleep mode, no bus activity is occurring, the host is non-operational by the user. The user inserts the Miniature Card and then presses a button to turn the host on before the system is operational.
Insertion, Hot	*User Perception:* Insertion of a Miniature Card when the host is running. *Host State:* The host would be in running mode, bus activity is occurring, the host is operational by the user. The user inserts the card, the host recognizes it, and the host continues to be operational. Note: Hot insertion may require buffering on the host system for proper operation.
Insertion, Pseudo Hot	*User Perception:* Insertion of a Miniature Card when the host is running. *Host State:* The host would be in running mode, bus activity is occurring, the host is operational by the user. The user inserts the card, the host immediately powers off before the Miniature Card makes contact with the host's internal bus. The user would then need to press a button to turn the host on for it to become operational.
Interface Signals	Miniature Card signals that make connection through the elastomeric connector.

Table 1. Miniature Card Definitions (Continued)

Term	Meaning
JEDEC	Acronym for Joint Electronic Device Engineering Council
Miniature Card Backside	The side of the Miniature Card that contains the latching mechanism. The backside is opposite the frontside.
Miniature Card Bottomside	The side of the Miniature Card that contains the interface signals. The bottomside is opposite the topside.
Miniature Card Frontside	The side of the Miniature Card that contains power, insertion, ground, voltage keys, and alignment notch. The frontside is opposite the backside.
Miniature Card Topside	The side of the Miniature Card that contains the Miniature Card label. The topside is opposite the bottomside.
PC Card	A memory or I/O card compatible with the PC Card Standard
PC Card Adapter	The hardware that connects the Miniature Card 60 contact bus to the PC Card 68 pin bus. This hardware can be mechanically implemented by following the PC Card Type II specification.
Power/Insertion Signals	The three signals on the frontside of the Miniature Card that provide ground, power and early detection of insertion.
Pull-Ups	Resistors used to ensure that signals do not float when no device is driving them.
Removal, Cold	*User Perception:* Removal of a Miniature Card when the host is off. *Host State:* The host would either be off or in sleep mode, no bus activity is occurring, the host is non-operational by the user. User would turn off the host, then remove the Miniature Card and then press a button to turn the host on for it to become operational again.
Removal, Hot	*User Perception:* Removal of the Miniature Card when the host is running. *Host State:* The host would be in running mode, bus activity is occurring, the host is operational by the user. User removes the card, the host recognizes the event, and the host continues to be operational.
Removal, Pseudo Hot	*User Perception:* Removal of the Miniature Card when the host is running. *Host State:* The host would be in running mode, bus activity is occurring, the host is operational by the user. User removes the card, the host recognizes the event, the host immediately powers off before the Miniature Card removes contact with the host's internal bus. The user would then need to press a button to turn the host on for it to be operational again.
Sector	Usually 64 KBytes, but depends on device used in the card. In word mode, a sector is 64 KWords.
Tuple	An element of the PC Card Standard CIS which provide card attribute information, and a link to the next tuple in a string of tuples.
User Insertable	All Miniature Cards should be inserted into the host by the user without the need for any special tools.
User Removable	This type of Miniature Card can be removed by the user without the need for any special tools. It contains programs and data that users may want to switch often. The use of this type of card is similar to a floppy disk.
User Non-Removable	This type of Miniature Card must be removed by the user with a special tool. It contains memory upgrades or boot program that users switches only when they require an upgrade. The use of this type of card is similar to a SIMM memory expansion or boot hard disk.
XIP	Acronym for eXecute-In-Place, which refers to code that executes directly from a Miniature Card.

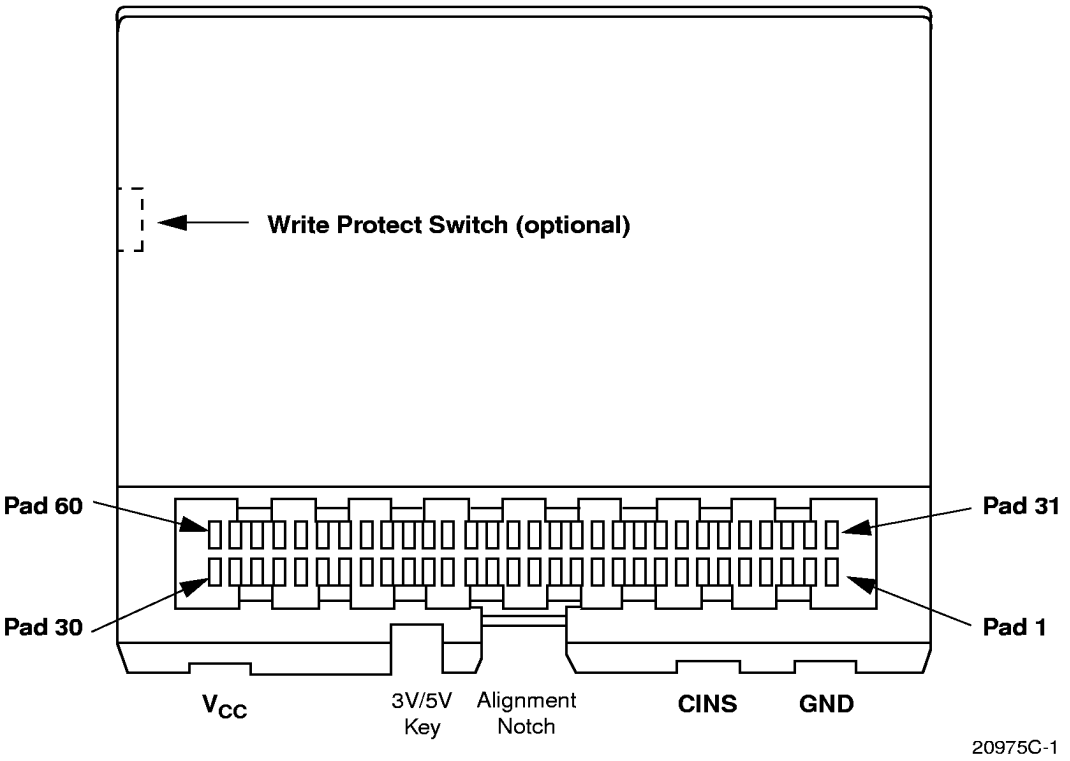


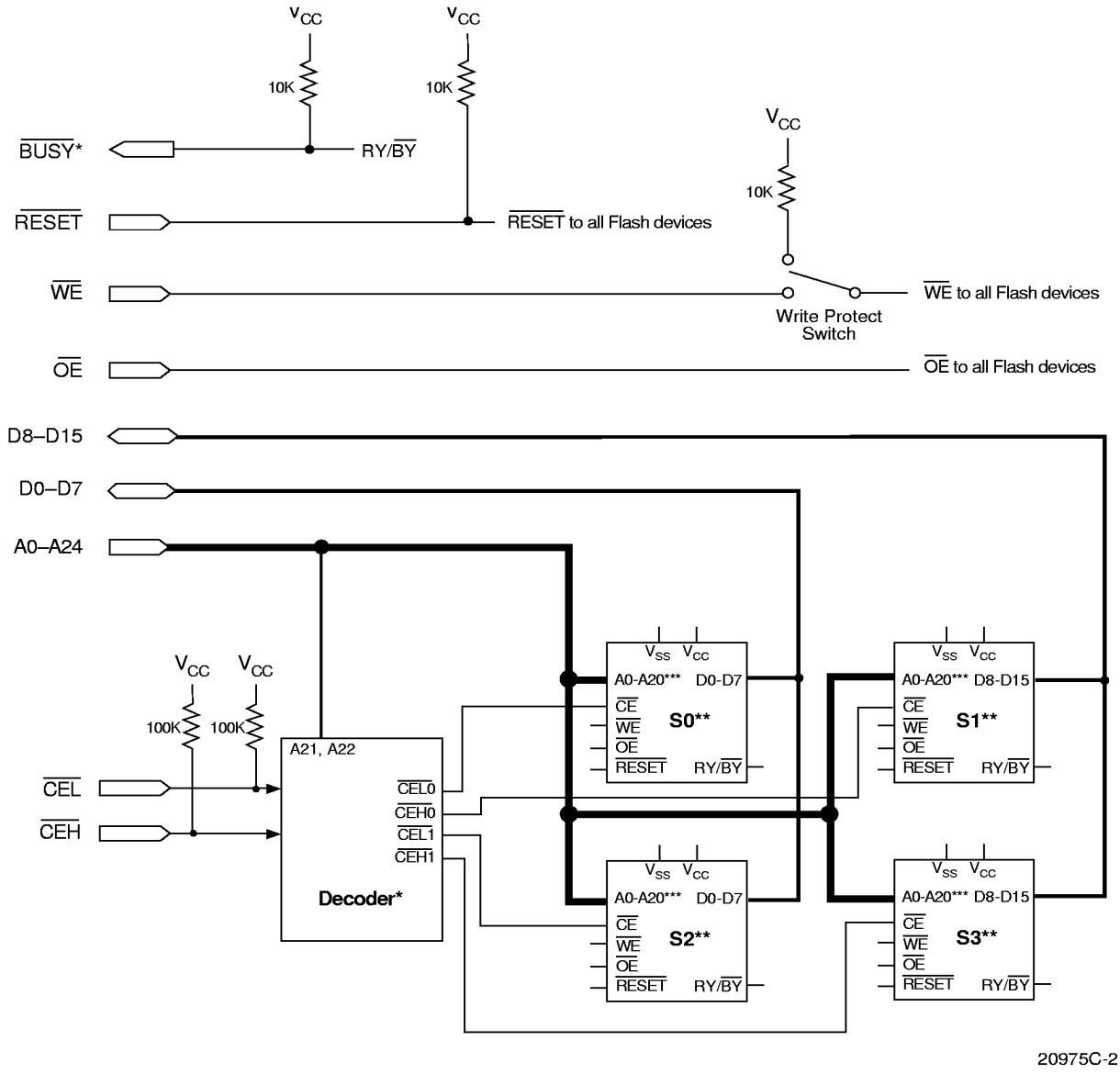
Figure 1. Miniature Card Connector (Card Bottom View)

Note: *Refer to the Physical Dimensions section for more information. Also refer to the MCIF specification for detailed mechanical information, available on the Web at http://www.mcif.org.*

Table 2. AMD Flash Miniature Cards and Flash Devices

Family Part Number	Density	No. of Flash Devices	AMD Flash Memory
AmMC002A	2 MByte	2	Am29F080
AmMC004A	4 MByte	2	Am29F016
AmMC008A	8 MByte	4	Am29F016

BLOCK DIAGRAM



* *8 MByte card only. Not used on 2 and 4 MByte cards.*

** *2 MByte card: Two Am29F080 devices, S0 and S1*
4 MByte card: Two Am29F016 devices, S0 and S1
8 MByte card: Four Am29F016 devices, S0...S3

*** *A0–A19 on 2 MByte card; A0–A20 on 4 and 8 MByte cards.*

Note: *On the 2 MByte card, A20–A24 are not connected. On the 4 MByte cards, A21–A24 are not connected. On the 8 MByte cards, A22-A24 are not connected. Connections not shown in this diagram are* ***not connected*** *internally.*

MINIATURE CARD PAD ASSIGNMENTS

A0–A24

Address A0 to A24 are the address bus lines that can address up to 32 Mwords (64 MBytes). The address lines are word addressed. The Miniature Card specification does not require the Miniature Card to decode the upper address lines. A 2 Mbyte Miniature Card that does not decode the upper address lines would repeat its address space every 2 MBytes. Address 0h would access the same physical location as 200000h, 400000h, 600000h, etc. On the 2 Mbyte cards, A20–A24 are not connected. On the 4 MByte cards, A21–A24 are not connected. On the 8 MByte cards, A22–A24 are not connected.

D0–D15

Data lines D0 through D15 constitute the data bus. The data bus is composed of two bytes; the low byte is D0–D7 and the high byte is D8–D15. These lines are tristated when $\overline{OE}$ is high.

$\overline{OE}$

$\overline{OE}$ indicates to the card that the current bus cycle is a read cycle. The output enable access time (t_{OE}) is the delay from the falling edge of $\overline{OE}$ to valid data at the output pins (assuming the addresses have been stable for at least $t_{ACC} - t_{OE}$ time).

$\overline{WE}$

$\overline{WE}$ indicates to the card that the current bus cycle is a write cycle. The falling edge of $\overline{WE}$ latches address information and the rising edge latches data/command information.

$\overline{VS1}$

Voltage Sense 1 signal. This signal is left open or not connected.

$\overline{VS2}$

Voltage Sense 2 signal. This signal is left open or not connected.

$\overline{CEL}$

$\overline{CEL}$ enables the low byte of the data bus (D0–D7) on the card.

$\overline{CEH}$

$\overline{CEH}$ enables the high byte of the data bus (D8–D15) on the card.

$\overline{RESET}$

$\overline{RESET}$ controls card initialization. When $\overline{RESET}$ transitions from a low state to a high state, the Miniature Card resets to the Read state.

$\overline{BUSY}$

$\overline{BUSY}$ is a signal generated by the card to indicate the status of operations within the Miniature Card. When $\overline{BUSY}$ is high, the Miniature Card is ready to accept the next command from the host. When $\overline{BUSY}$ is low, the Miniature Card is busy and unable to accept some data operations from the host. For example, in Flash Miniature Cards the $\overline{BUSY}$ signal is tied to the components' RY/$\overline{BY}$ signal.

$\overline{CD}$

$\overline{CD}$ is a grounded interface signal. After a Miniature Card has been inserted, $\overline{CD}$ will be forced low. The card detect signal is located in the center of the second row of interface signals, and should be one of the last interface signals to connect to the host. Do not confuse $\overline{CD}$ with $\overline{CINS}$. $\overline{CINS}$ is an early card detect that is one of the first signals to connect to the host.

$\overline{CINS}$

$\overline{CINS}$ is a grounded signal on the front of the Miniature Card that is used for early detection of a card insertion. $\overline{CINS}$ makes contact on the host when the front of the card is inserted into the socket, before the interface signals connect.

$\overline{BS8}$

The $\overline{BS8}$ (Bus size 8) signal indicates to the Miniature Card that the host has an 8-bit bus. AMD Flash Miniature Cards ignore this signal. An 8-bit host must connect its D0–D7 data lines to D8–D15 on the Miniature Card to retrieve the upper (odd) byte.

GND

Ground

V_{CC}

Vcc is used to supply power to the card.

NC

No connect

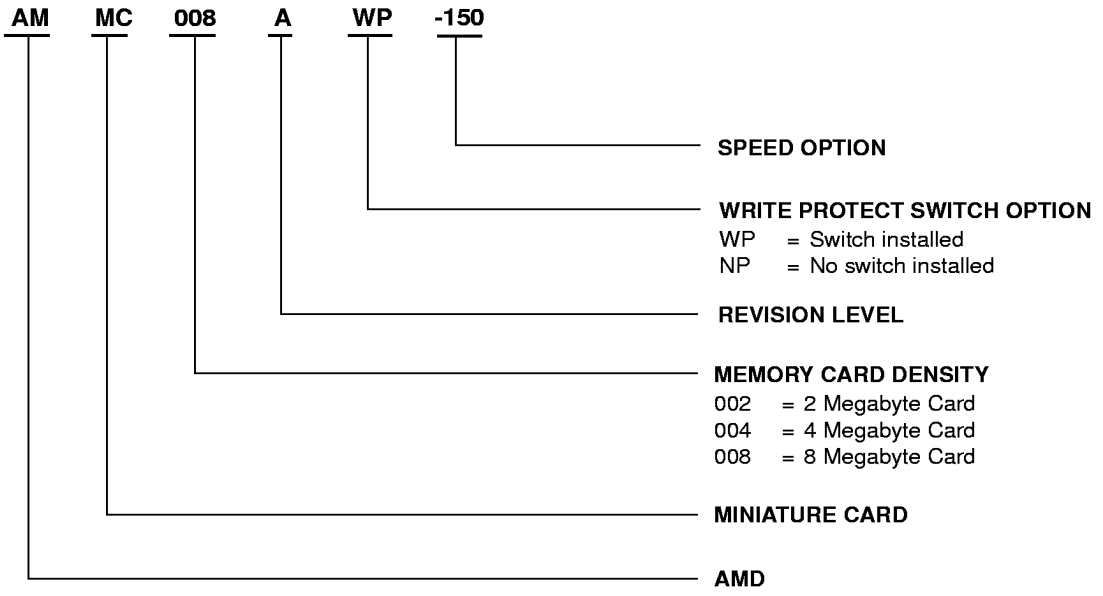
RFU

Reserved for future use

ORDERING INFORMATION

Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of the following:



INTERFACE SIGNAL ASSIGNMENTS

Pad Number	Signal Name	Pad Number	Signal Name	Pad Number	Signal Name
1	A18	21	D12	41	A4
2	A16	22	D10	42	$\overline{CEL}$
3	A14	23	D9	43	A1
4	NC	24	D0	44	NC
5	$\overline{CEH}$	25	D2	45	NC
6	A11	26	D4	46	$\overline{CD}$
7	A9	27	RFU	47	A21
8	A8	28	D7	48	$\overline{BUSY}$
9	A6	29	NC	49	$\overline{WE}$
10	A5	30	NC	50	D14
11	A3	31	A19	51	RFU
12	A2	32	A17	52	D11
13	A0	33	A15	53	$\overline{VS2}$
14	NC	34	A13	54	D8
15	A24	35	A12	55	D1
16	A23	36	$\overline{RESET}$	56	D3
17	A22	37	A10	57	D5
18	$\overline{OE}$	38	$\overline{VS1}$	58	D6
19	D15	39	A7	59	RFU
20	D13	40	$\overline{BS8}$	60	A20

Note: *NC = No Connect; RFU = Reserved for Future Use.*

FLASH MINIATURE CARD OPERATIONS

Voltage Sensing

AMD Miniature Cards provide two voltage sense signals for hosts that support multiple voltages. The multivoltage host can sense the voltage level of the Miniature Card and power up the card at that voltage. See Table 3 for a description of the voltage sense signals.

In addition to the voltage sense pins, there are also mechanical voltage keys on the Miniature Card that ensure the card can only be inserted into host systems that can supply the proper voltage levels to the card. Refer to Section 4.1.2 in the Miniature Card specification for more information on mechanical keying.

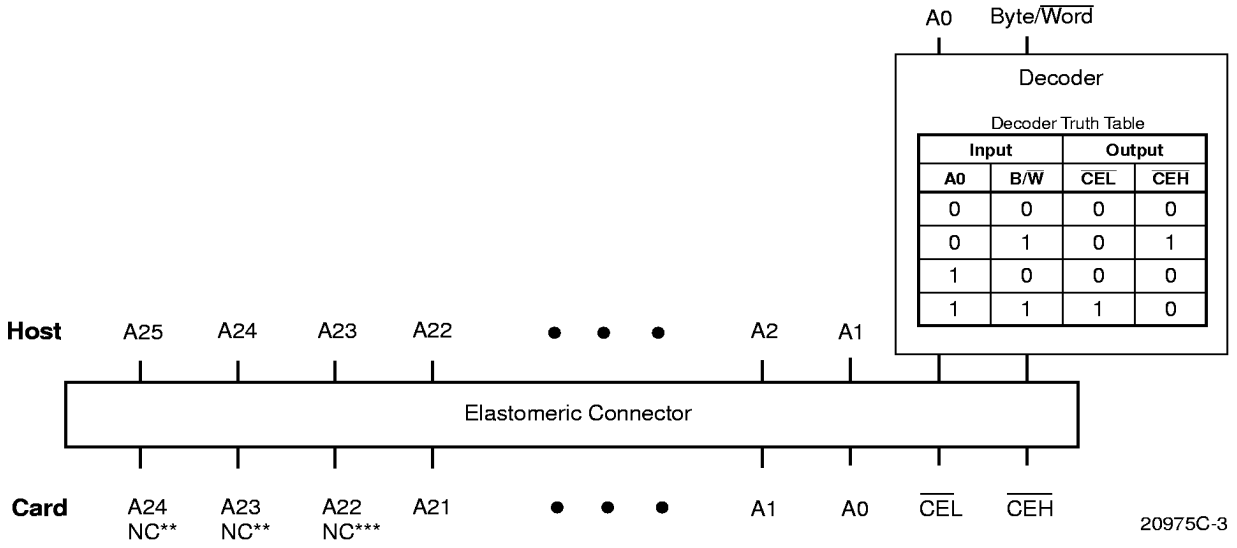
Table 3. Voltage Sense Signals

Miniature Card Power-Up Voltage	$\overline{VS1}$	$\overline{VS2}$
5 Volt-only	Open	Open

Data Accesses

The Miniature Card has a 16-bit data bus that can accommodate word or byte accesses. By individually asserting $\overline{CEL}$ and $\overline{CEH}$, a host can access either byte. However, byte swapping (moving the high byte data to the low byte) is not supported.

Figure 2 shows the connections between the host and Miniature Card. The host system address lines range from A0-A25, whereas the Miniature Card address lines range from A0–A24. On the host, A0 and the byte/word line are sent to a decoder and output to $\overline{CEL}$ and $\overline{CEH}$ on the Miniature Card. These two bits enable a single device for byte accesses and two devices for word accesses, as shown by the decoder truth table in Figure 2. Again, the Miniature Card address lines do not receive input from host address bit A0. In this document, all address references are ***card addresses***, unless otherwise noted. Table 4 shows the read/write modes for Miniature Cards.



* *Not connected*

* *Not connected on 2 and 4 MByte card only*

*** *Not connected on 2 MByte card only*

Figure 2. Host/Card Address Connections

Table 4. Miniature Card Read/Write Modes

Function	$\overline{CEH}$	$\overline{CEL}$	$\overline{WE}$	$\overline{OE}$	D8–D15	D0–D7
Read Mode						
Standby	H	H	X	X	High-Z	High-Z
Word Access	L	L	H	L	High Byte Data	Low Byte Data
Low Byte Access	H	L	H	L	High-Z	Low Byte Data
High Byte Access	L	H	H	L	High Byte Data	High-Z
Write Mode						
Standby	H	H	X	X	High-Z	High-Z
Word Access	L	L	L	H	High Byte Data	Low Byte Data
Low Byte Access	H	L	L	H	High-Z	Low Byte Data
High Byte Access	L	H	L	H	High Byte Data	High-Z

Notes:

1. *Unlisted access combinations are invalid and may return unexpected results.*
2. *X indicates a Don't Care value.*

Erase Operations

The AMD Flash Miniature Card is organized as an array of individual devices. On the 2 MByte Miniature Card, each Am29F080 device contains sixteen 64 KByte sectors, for a total of 1 MByte of memory space per device. On 4 and 8 MByte Miniature Cards, each Am29F016 device contains thirty-two 64 KByte sectors, for a total of 2 MBytes of memory space per device.

Flash technology allows any logical "1" data bit to be programmed to a logical "0". The only way to reset bits to a logical "1" is to erase that entire memory sector or memory device. Once a memory sector or memory device is erased, any address location may be programmed. Two or more devices may be erased concurrently when additional I_{CC} current is supplied to the card. However, erasing more than two devices concurrently is not typical in battery-powered applications, but may take place during procedures such as card testing.

Since erase commands operate on entire sectors or devices, the host should track the affected memory addresses; for example, by determining the sector size and device size and calculating the corresponding addresses.

Erase operations can be performed in several ways:

- Erase a single sector or multiple sectors in a device
- Erase a sector pair
- Erase multiple device pairs *
- Erase the entire card *

* This operation is only feasible in solutions capable of supplying more than the specified miniature card supply current requirement (150 mA) per system. Each AMD Flash memory **device pair** can accept a maximum of 120 mA supply current.

The common memory space data contents are altered in a similar manner as writing to individual Flash memory devices. An on-card address decoder activates the appropriate Flash device in the memory array. Each device internally latches address and data during write cycles. Refer to Table 4.

Word-Wide Operations

The AMD Miniature Card provide the flexibility to operate on data in a byte-wide or word-wide format. In word-wide operations, the low bytes are controlled with $\overline{CEL}$. The high bytes are controlled with $\overline{CEH}$. Refer to the block diagram for more information.

Byte-Wide Operations

Byte-wide data is available for read and write operations ($\overline{CEL}$ = 0, $\overline{CEH}$ = 1). Even and odd bytes are stored in separate memory devices (for example, S0 and S1) and are accessed by controlling $\overline{CEL}$ and $\overline{CEH}$. The even byte is the low order byte and the odd byte is the high order byte of a 16-bit word.

Each memory sector or device pair must be addressed separately for erase operations. Refer to the block diagram for more information.

Card Detection

Each $\overline{CD}$ (output) pin should be detected by the host system to determine if the memory card is adequately seated in the socket. $\overline{CD}$ and $\overline{CINS}$ are internally tied to ground. If both bits are not detected, the system should indicate that the card must be re-inserted.

Data Protection

An optional mechanical write protect switch provides user-initiated write protection. When this switch is activated, $\overline{WE}$ is internally forced high. The Flash memory command register is disabled from accepting any write commands. This prevents the card from responding to any commands (for example, an Autoselect command). See Figure 3.

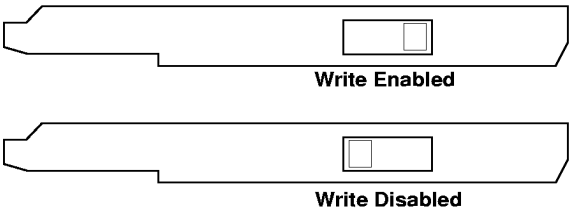


Figure 3. Write Protect Switch (Card Right Side View)

In addition to card-level data protection, AMD Flash Miniature Cards offer several device-level data protection features.

Device-Level Data Protection

AMD Flash memory devices offer protection against accidental erasure or programming caused by spurious system level signals that may exist during power transitions. During power up, each device automatically resets the internal state machine to the read mode. The control register architecture allows alteration of the memory contents only occurs after successful completion of specific multi-bus cycle command sequences.

AMD Flash memory devices also incorporate the following 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, the AMD memory devices in the Miniature Card lock out write cycles for $V_{CC} < V_{LKO}$ (see "DC Characteristics" on page 25 for voltages). When $V_{CC} < V_{LKO}$, the command register is disabled, all internal program/erase circuits are disabled, and the device resets to the read mode. These memory devices ignore all writes until $V_{CC} > V_{LKO}$. The user must ensure that the control pins are in the correct logical state when $V_{CC} > V_{LKO}$ to prevent unintentional writes.

Write Pulse "Glitch" Protection

Noise pulses of less than 5 ns (typical) on $\overline{OE}$, $\overline{CE}$, or $\overline{WE}$ will neither initiate a write cycle nor change the command registers.

Logical Inhibit

Writing is inhibited by holding any one of $\overline{OE} = V_{IL}$, $\overline{CE} = V_{IH}$, or $\overline{WE} = V_{IH}$. To initiate a write cycle $\overline{CE}$ and $\overline{WE}$ must be a logical zero while $\overline{OE}$ is a logical one.

Power-Up Write Inhibit

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

Read Mode

Two Card Enable ($\overline{CE}$) pins are available on the memory card. Both $\overline{CE}$ pins must be active low for word-wide read accesses. Only one $\overline{CE}$ is required for byte-wide accesses. The $\overline{CE}$ pins select and determine when to apply power to the high-byte and low-byte memory devices. The Output Enable ($\overline{OE}$) controls gating accessed data from the memory device outputs.

The Miniature card automatically powers 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.

Output Disable

Data outputs from the card are disabled when $\overline{OE}$ is at a logic-high level. Under this condition, outputs are in the high-impedance state.

Standby Operations

Byte-wide read accesses only require half of the read/write output buffer (x16) to be active. In addition, only one memory device is active within either the high order or low order bank. Activation of the appropriate half of the output buffer is controlled by the combination of both $\overline{CE}$ pins. The $\overline{CE}$ pins also control power to the high and low-order banks of memory. Outputs of the memory bank not selected are placed in the high impedance state. The individual memory device is activated by the address decoders. The other memory devices operate in standby. An active memory device continues to draw power until completion of a write or erase operation if the card is de-selected in the process of one of these operations.

Autoselect Operation

A host system or external card reader/writer can determine the on-card manufacturer and device I.D. codes. Codes are available after writing the 90H command to

the command register of a memory device, as shown in Tables 5 through 9. When the autoselect command is issued to card address 00000H, the Miniature Card returns the manufacturer I.D. If the autoselect command is issued to card address 00001H, the Miniature Card provides the device I.D.

To terminate the Auto Select operation, the Read/Reset command sequence must be written to the same device. The Autoselect command operates only if the card is not write protected.

Write Operations

Write and erase operations are valid only when V_{CC} is above 4.5 V. This activates the state machine of an addressed memory device. The command register is a latch which saves address, commands, and data information used by the state machine and memory array.

When Write Enable ($\overline{WE}$) and appropriate $\overline{CE}$ signals are at a logic-level low, and Output Enable ($\overline{OE}$) is at a logic-high, the command register is enabled for write operations. The falling edge of $\overline{WE}$ latches address information and the rising edge latches data/command information.

Write or erase operations are performed by writing appropriate data patterns to the command register of accessed Flash memory devices.

The byte-wide commands are defined in Tables 6 and 9; word-wide commands are defined in Tables 5 and 8. Note that the Erase Suspend (B0H) and Erase Resume (30H) commands are valid only while the Sector Erase operation is in progress.

Table 5. Word Command Definitions for 2 MByte Cards

Embedded Command Sequence (Note 1)	Bus Write Cycles Req'd	First Bus Write Cycle		Second Bus Write Cycle		Third Bus Write Cycle		Fourth Bus Read/Write Cycle		Fifth Bus Write Cycle		Sixth Bus Write Cycle	
		Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data
Reset/Read	1	XXXH	F0F0H										
Reset/Read	4	555H	AAAAH	2AAH	5555H	555H	F0F0H	RA	RW				
Autoselect Manufacturer ID	4	555H	AAAAH	2AAH	5555H	555H	9090H	X00H	0101H				
Autoselect Device ID	4	555H	AAAAH	2AAH	5555H	555H	9090H	X01H	D5D5H				
Word Write	4	555H	AAAAH	2AAH	5555H	555H	A0A0H	PA	PW				
Device Erase	6	555H	AAAAH	2AAH	5555H	555H	8080H	555H	AAAAH	2AAH	5555H	555H	1010H
Sector Erase	6	555H	AAAAH	2AAH	5555H	555H	8080H	555H	AAAAH	2AAH	5555H	SA	3030H
Sector Erase Suspend		XXXH	B0B0H	Erase can be suspended during sector erase with Addr (don't care), Data (B0B0H)									
Sector Erase Resume		XXXH	3030H	Erase can be resumed after suspend with Addr (don't care), Data (3030H)									

* *During word addressing, $\overline{CEL}$ = 0, $\overline{CEH}$ = 0, and address is applied to Memory Device Pair 0 (S0 and S1). For host-to-card address bit connections, see Figure 2.*

Notes:

1. *Write protect must not be enabled for proper operation of all commands except Reset/Read.*
2. *Address bits A19–A11 = X = Don't Care for all commands except for Read Address (RA), Program Address (PA), and Sector Address (SA).*
3. *See Table 4 for read/write modes.*
4. *RA = Address of the memory location to be read.*
   PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of the $\overline{WE}$ pulse.
   SA = Address of the sector to be erased. Refer to Table 11 for sector addresses.
5. *RW = Data read from location RA during read operation.*
   PW = Data to be programmed at location PA. Data is latched on the rising edge of $\overline{WE}$.
6. *Word = high byte + low byte.*

Table 6. Even Byte Command Definitions for 2 MByte Cards

Embedded Command Sequence (Note 1)	Bus Write Cycles Req'd	First Bus Write Cycle		Second Bus Write Cycle		Third Bus Write Cycle		Fourth Bus Read/Write Cycle		Fifth Bus Write Cycle		Sixth Bus Write Cycle	
		Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data
Reset/Read	1	XXXH	XXF0H										
Reset/Read	4	555H	XXAAH	2AAH	XX55H	555H	XXF0H	RA	RD				
Autoselect Manufacturer ID	4	555H	XXAAH	2AAH	XX55H	555H	XX90H	X00H	XX01H				
Autoselect Device ID	4	555H	XXAAH	2AAH	XX55H	555H	XX90H	X01H	XXD5H				
Byte Write	4	555H	XXAAH	2AAH	XX55H	555H	XXA0H	PA	PD				
Device Erase	6	555H	XXAAH	2AAH	XX55H	555H	XX80H	555H	XXAAH	2AAH	XX55H	555H	XX10H
Sector Erase	6	555H	XXAAH	2AAH	XX55H	555H	XX80H	555H	XXAAH	2AAH	XX55H	SA	XX30H
Sector Erase Suspend		XXXH	XXB0H	Erase can be suspended during sector erase with Addr (don't care), Data (XXB0H)									
Sector Erase Resume		XXXH	XX30H	Erase can be resumed after suspend with Addr (don't care), Data (XX30H)									

* *During even (low) byte accesses, $\overline{CEL}$ = 0, $\overline{CEH}$ = 1, and address is applied to Memory Device 0 (S0) only.*

Table 7. Odd Byte Command Definitions for 2 MByte Cards

Embedded Command Sequence (Note 1)	Bus Write Cycles Req'd	First Bus Write Cycle		Second Bus Write Cycle		Third Bus Write Cycle		Fourth Bus Read/Write Cycle		Fifth Bus Write Cycle		Sixth Bus Write Cycle	
		Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data
Reset/Read	1	XXXH	F0XXH										
Reset/Read	4	555H	AAXXH	2AAH	55XXH	555H	F0XXH	RA	RD				
Autoselect Manufacturer ID	4	555H	AAXXH	2AAH	55XXH	555H	90XXH	X00H	01XXH				
Autoselect Device ID	4	555H	AAXXH	2AAH	55XXH	555H	90XXH	X01H	D5XXH				
Byte Write	4	555H	AAXXH	2AAH	55XXH	555H	A0XXH	PA	PDXXH				
Device Erase	6	555H	AAXXH	2AAH	55XXH	555H	80XXH	555H	AAXXH	2AAH	55XXH	555H	10XXH
Sector Erase	6	555H	AAXXH	2AAH	55XXH	555H	80XXH	555H	AAXXH	2AAH	55XXH	SA	30XXH
Sector Erase Suspend		XXXH	B0XXH	Erase can be suspended during sector erase with Addr (don't care), Data (B0XXH)									
Sector Erase Resume		XXXH	30XXH	Erase can be resumed after suspend with Addr (don't care), Data (30XXH)									

* *During odd (high) byte accesses, $\overline{CEL}$= 1, $\overline{CEH}$ = 0, and address is applied to Memory Device 1 (S1) only.*

Notes for Table 6 and Table 7:

1. *Write protect must not be enabled for proper operation of all commands except Reset/Read.*
2. *For host-to-card address bit connections, see Figure 2.*
3. *Address bits A19–A11 = X = Don't Care for all address commands except for Program Address (PA), Read Address (RA), and Sector Address (SA).*
4. *Bus operations are defined in Table 4.*
5. *RA = Address of the memory location to be read.*
   PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of the $\overline{WE}$ pulse.
   SA = Address of the sector to be erased. Refer to Table 11 for sector addresses.
6. *RD = Data read from location RA during read operation.*
   PD = Data to be programmed at location PA. Data is latched on the rising edge of $\overline{WE}$ pulse.

Table 8. Word Command Definitions for 4 and 8 MByte Cards

Embedded Command Sequence (Note 1)	Bus Write Cycles Req'd	First Bus Write Cycle		Second Bus Write Cycle		Third Bus Write Cycle		Fourth Bus Read/Write Cycle		Fifth Bus Write Cycle		Sixth Bus Write Cycle	
		Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data
Reset/Read	1	XXXXH	F0F0H										
Reset/Read	4	XXXXH	AAAAH	XXXXH	5555H	XXXXH	F0F0H	RA	RW				
Autoselect Manufacturer ID	4	XXXXH	AAAAH	XXXXH	5555H	XXXXH	9090H	XX00H	0101H				
Autoselect Device ID	4	XXXXH	AAAAH	XXXXH	5555H	XXXXH	9090H	XX01H	3D3DH				
Byte Write	4	XXXXH	AAAAH	XXXXH	5555H	XXXXH	A0A0H	PA	PW				
Device Erase	6	XXXXH	AAAAH	XXXXH	5555H	XXXXH	8080H	XXXXH	AAAAH	2AAAH	5555H	XXXXH	1010H
Sector Erase	6	XXXXH	AAAAH	XXXXH	5555H	XXXXH	8080H	XXXXH	AAAAH	2AAAH	5555H	SA	3030H
Sector Erase Suspend		XXXXH	B0B0H	Erase can be suspended during sector erase with Addr (don't care), Data (B0B0H)									
Sector Erase Resume		XXXXH	3030H	Erase can be resumed after suspend with Addr (don't care), Data (3030H)									

* *During word addressing, $\overline{CEL}$ = 0, $\overline{CEH}$ = 0, and address is applied to Memory Device Pair 0 (S0 and S1). On 8 MByte cards, address for Memory Device Pair 1 = (Addr) + 400000H, and address is applied to S2 and S3. For host-to-card address bit connections, see Figure 2.*

Notes:

1. *Write protect must not be enabled for proper operation of all commands except Reset/Read.*
2. *Address bits = X = Don't Care for all address commands except for Read Address (RA), Program Address (PA), and Sector Address (SA).*
3. *Bus operations are defined in Table 4.*
4. *RA = Address of the memory location to be read.*
   PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of the $\overline{WE}$ pulse.
   SA = Address of the sector to be erased. Refer to Table 12 for sector addresses.
5. *RW = Data read from location RA during read operation.*
   PW = Data to be programmed at location PA. Data is latched on the rising edge of $\overline{WE}$.
6. *Word = high byte + low byte.*

Table 9. Even Byte Command Definitions for 4 and 8 MByte Cards

Embedded Command Sequence (Note 1)	Bus Write Cycles Req'd	First Bus Write Cycle		Second Bus Write Cycle		Third Bus Write Cycle		Fourth Bus Read/Write Cycle		Fifth Bus Write Cycle		Sixth Bus Write Cycle	
		Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data
Reset/Read	1	XXXXH	XXF0H										
Reset/Read	4	XXXXH	XXAAH	XXXXH	XX55H	XXXXH	XXF0H	RA	RD				
Autoselect Manufacturer ID	4	XXXXH	XXAAH	XXXXH	XX55H	XXXXH	XX90H	XX00H	XX01H				
Autoselect Device ID	4	XXXXH	XXAAH	XXXXH	XX55H	XXXXH	XX90H	XX01H	XX3DH				
Byte Write	4	XXXXH	XXAAH	XXXXH	XX55H	XXXXH	XXA0H	PA	PD				
Device Erase	6	XXXXH	XXAAH	XXXXH	XX55H	XXXXH	XX80H	XXXXH	XXAAH	XXXXH	XX55H	XXXXH	XX10H
Sector Erase	6	XXXXH	XXAAH	XXXXH	XX55H	XXXXH	XX80H	XXXXH	XXAAH	XXXXH	XX55H	SA	XX30H
Sector Erase Suspend		XXXXH	XXB0H	Erase can be suspended during sector erase with Addr (don't care), Data (XXB0H)									
Sector Erase Resume		XXXXH	XX30H	Erase can be resumed after suspend with Addr (don't care), Data (XX30H)									

* *During low byte addressing, $\overline{CEL}$ = 0, $\overline{CEH}$ = 1, and address applied to Memory Device 0 (S0) = (Addr). On 8 MByte cards, address for S2 = (Addr) + 400000H.*

Table 10. Odd Byte Command Definitions for 4 and 8 MByte Cards

Embedded Command Sequence (Note 1)	Bus Write Cycles Req'd	First Bus Write Cycle		Second Bus Write Cycle		Third Bus Write Cycle		Fourth Bus Read/Write Cycle		Fifth Bus Write Cycle		Sixth Bus Write Cycle	
		Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data	Addr*	Data
Reset/Read	1	XXXXH	F0XXH										
Reset/Read	4	XXXXH	AAXXH	XXXXH	55XXH	XXXXH	F0XXH	RA	RD				
Autoselect Manufacturer ID	4	XXXXH	AAXXH	XXXXH	55XXH	XXXXH	90XXH	XX00H	01XXH				
Autoselect Device ID	4	XXXXH	AAXXH	XXXXH	55XXH	XXXXH	90XXH	XX01H	3DXXH				
Byte Write	4	XXXXH	AAXXH	XXXXH	55XXH	XXXXH	A0XXH	PA	PD				
Device Erase	6	XXXXH	AAXXH	XXXXH	55XXH	XXXXH	80XXH	XXXXH	AAXXH	XXXXH	55XXH	XXXXH	10XXH
Sector Erase	6	XXXXH	AAXXH	XXXXH	55XXH	XXXXH	80XXH	XXXXH	AAXXH	XXXXH	55XXH	SA	30XXH
Sector Erase Suspend		XXXXH	B0XXH	Erase can be suspended during sector erase with Addr (don't care), Data (B0XXH)									
Sector Erase Resume		XXXXH	30XXH	Erase can be resumed after suspend with Addr (don't care), Data (30XXH)									

* *During high byte addressing, $\overline{CEL}$ = 1, $\overline{CEH}$ = 0, and address applied to Memory Device 1 (S1) = (Addr) + 200000H. On 8 MByte cards, address for S3 = (Addr) + 400000H + 200000H. For host-to-card address bit connections, see Figure 2.*

Notes:

1. *Write protect must not be enabled for proper operation of all commands except Reset/Read.*
2. *Address bits = X = Don't Care for all address commands except for Program Address (PA), Read Address (RA) and Sector Address (SA).*
3. *Bus operations are defined in Table 4.*
4. *RA = Address of the memory location to be read.*
   PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of the $\overline{WE}$ pulse.
   SA = Address of the sector to be erased. Refer to Table 12 for sector addresses.
5. *RD = Data read from location RA during read operation.*
   PD = Data to be programmed at location PA. Data is latched on the rising edge of $\overline{WE}$ pulse.

Table 11. Memory Sector Addresses for 2 MByte Card

Sector	Card Address Bits				Device 0 and/or 1 (Note 1)
	A19	A18	A17	A16	Card Address Range
0	0	0	0	0	00000h–0FFFFh
1	0	0	0	1	10000h–1FFFFh
2	0	0	1	0	20000h–2FFFFh
3	0	0	1	1	30000h–3FFFFh
4	0	1	0	0	40000h–4FFFFh
5	0	1	0	1	50000h–5FFFFh
6	0	1	1	0	60000h–6FFFFh
7	0	1	1	1	70000h–7FFFFh
8	1	0	0	0	80000h–8FFFFh
9	1	0	0	1	90000h–9FFFFh
10	1	0	1	0	A0000h–AFFFFh
11	1	0	1	1	B0000h–BFFFFh
12	1	1	0	0	C0000h–CFFFFh
13	1	1	0	1	D0000h–DFFFFh
14	1	1	1	0	E0000h–EFFFFh
15	1	1	1	1	F0000h–FFFFFh

Notes:

1. *For word addressing, devices 0 and 1 (S0 and S1) together form Memory Device Pair 0. Refer to the block diagram for device connections.*
2. *Card address bits range from A0 to A19. Host address bits range from A0 to A20. Host address bit A0 is used for controlling the $\overline{CEL}$ and $\overline{CEH}$ inputs to the card. Refer to Figure 2 for host-to-card address bit connections.*

Table 12. Memory Sector Addresses for 4 and 8 MByte Cards

Sector	Card Address Bits					Device 0 and/or 1	Device 2 and/or 3
	A20	A19	A18	A17	A16	Card Address Range (Note 2)	Card Address Range (Note 2)
0	0	0	0	0	0	00000h–0FFFFh	200000h–20FFFFh
1	0	0	0	0	1	10000h–1FFFFh	210000h–21FFFFh
2	0	0	0	1	0	20000h–2FFFFh	220000h–22FFFFh
3	0	0	0	1	1	30000h–3FFFFh	230000h–23FFFFh
4	0	0	1	0	0	40000h–4FFFFh	240000h–24FFFFh
5	0	0	1	0	1	50000h–5FFFFh	250000h–25FFFFh
6	0	0	1	1	0	60000h–6FFFFh	260000h–26FFFFh
7	0	0	1	1	1	70000h–7FFFFh	270000h–27FFFFh
8	0	1	0	0	0	80000h–8FFFFh	280000h–28FFFFh
9	0	1	0	0	1	90000h–9FFFFh	290000h–29FFFFh
10	0	1	0	1	0	A0000h–AFFFFh	2A0000h–2AFFFFh
11	0	1	0	1	1	B0000h–BFFFFh	2B0000h–2BFFFFh
12	0	1	1	0	0	C0000h–CFFFFh	2C0000h–2CFFFFh
13	0	1	1	0	1	D0000h–DFFFFh	2D0000h–2DFFFFh
14	0	1	1	1	0	E0000h–EFFFFh	2E0000h–2EFFFFh
15	0	1	1	1	1	F0000h–FFFFFh	2F0000h–2FFFFFh
16	1	0	0	0	0	100000h–10FFFFh	300000h–30FFFFh
17	1	0	0	0	1	110000h–11FFFFh	310000h–31FFFFh
18	1	0	0	1	0	120000h–12FFFFh	320000h–32FFFFh
19	1	0	0	1	1	130000h–13FFFFh	330000h–33FFFFh
20	1	0	1	0	0	140000h–14FFFFh	340000h–34FFFFh
21	1	0	1	0	1	150000h–15FFFFh	350000h–35FFFFh
22	1	0	1	1	0	160000h–16FFFFh	360000h–36FFFFh
23	1	0	1	1	1	170000h–17FFFFh	370000h–37FFFFh
24	1	1	0	0	0	180000h–18FFFFh	380000h–38FFFFh
25	1	1	0	0	1	190000h–19FFFFh	390000h–39FFFFh
26	1	1	0	1	0	1A0000h–1AFFFFh	3A0000h–3AFFFFh
27	1	1	0	1	1	1B0000h–1BFFFFh	3B0000h–3BFFFFh
28	1	1	1	0	0	1C0000h–1CFFFFh	3C0000h–3CFFFFh
29	1	1	1	0	1	1D0000h–1DFFFFh	3D0000h–3DFFFFh
30	1	1	1	1	0	1E0000h–1EFFFFh	3E0000h–3EFFFFh
31	1	1	1	1	1	1F0000h–1FFFFFh	3F0000h–3FFFFFh

Notes:

1. *For word addressing, devices 0 and 1 (S0 and S1) together form Memory Device Pair 0; devices 2 and 3 (S2 and S3) form Memory Device Pair 1. Refer to the block diagram for device connections.*
2. *Card address bits range from A0 to A20. Host address bits range from A0 to A21. Host address bit A0 is used for controlling the $\overline{CEL}$ and $\overline{CEH}$ inputs to the card. Refer to Figure 2 for host-to-card address bit connections.*

PROGRAM AND ERASE OPERATIONS

AMD Flash Memory devices include Embedded Algorithms (Embedded Erase and Embedded Program) that allow the host to simply issue a command, after which it is free to perform other tasks. The host then only needs to monitor appropriate status bits to determine when the operation is complete.

Embedded Erase Algorithm

When erasing a sector or device, the Embedded Erase algorithm does not require the host to first entirely preprogram the device. Upon executing the Embedded Erase command sequence, the addressed memory sector or memory device automatically writes and verifies the entire memory device or memory sector for an all "0" data pattern. The system is not required to provide any controls or timing during these operations.

When the memory sector or memory device is automatically verified to contain an all "0" pattern, a self-timed chip erase-and-verify begins. The erase and verify operations are complete when the data on D7 (D15 on the odd byte) of the memory sector or memory device is "1" (see Write Operation Status section), at which time the device returns to the read mode. The system is not required to provide any control or timing during these operations. If a Reset command is issued while the erase operation is in progress, the erase operation will stop, and the data in that device will be undefined. In that case, restart the erase on that sector and allow it to complete.

When using the Embedded Erase algorithm, the erase automatically terminates when adequate erase margin has been achieved for the memory array (no erase verify command is required). The margin voltages are internally generated in the same manner as when the standard erase verify command is used.

The Embedded Erase command sequence is a command only operation that stages the memory sector or memory device for automatic electrical erasure of all bytes in the array. The automatic erase begins on the rising edge of the $\overline{WE}$ and terminates when the data on D7 of the memory sector or memory device is "1" (see Write Operation Status section) at which time the device returns to the Read mode. Please note that for the memory device or memory sector erase operation, $\overline{Data}$ Polling may be performed at any address in that device or sector.

Figure 4 and Table 13 illustrate the Embedded Erase Algorithm, a typical command string and bus operations.

As described earlier, once the memory sector in a device or memory device completes the Embedded Erase operation, it returns to the Read mode and addresses are no longer latched. Therefore, the device requires that a valid address input to the device is supplied by the system at this particular instant of time. Otherwise, the system will never read a "1" on D7. A system designer has the following choices to implement the Embedded Erase algorithm:

1. The host may keep the sector address (within any of the sectors being erased) valid during the entire Embedded Erase operation.
2. Once the system executes the Embedded Erase command sequence, the host may remove the address from the device and perform other tasks. The host is required to keep track of the valid sector address by loading it into a temporary register. When the host comes back to $\overline{Data}$ Poll the device, it must reassert the same address.
3. The host may monitor $\overline{BUSY}$ (RY/$\overline{BY}$) to determine the status of the Embedded Algorithm in progress. A "0" indicates that the device is busy; a "1" indicates that the algorithm is complete.

Since the Embedded Erase operation takes a significant amount of time (typical 1 s), option 2 makes more sense. However, the choice of these three options has been left to the system designer.

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 $\overline{WE}$ (or $\overline{CE}$), whichever occurs later, while the command (data) is latched on the rising edge of the $\overline{WE}$ (or $\overline{CE}$) pulse, whichever occurs first. A time-out of 100 µs from the rising edge of the last sector erase command will initiate the sector erase command(s)

Multiple sectors may be queued for concurrent erase by writing the six bus cycle operations as described above. This sequence is followed with writes of the sector erase command 30H to addresses in other sectors desired to be concurrently erased. A time-out of 100 µs from the rising edge of the $\overline{WE}$ (or $\overline{CE}$) pulse for the last sector erase command will initiate the sector erase. If another sector erase command is written within the 100 µs time-out window the timer is reset. Any command other than sector erase within the time-out window will reset the device to the read mode, ignoring the previous command string (refer to Write Operation Status section for Sector Erase Timer operation). Loading the sector erase buffer may be done in any sequence and with any sector number.

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. A Reset

command issued after the device has begun execution stops the erase operation, but the data in the sector will be undefined. In that case, restart the erase on that sector and allow it to complete.

The automatic sector erase begins after the 100 µs time out from the rising edge of the $\overline{WE}$ (or $\overline{CE}$) pulse for the last sector erase command pulse and terminates when the data on D7 is "1" (see Write Operation Status section) at which time the device returns to read mode. $\overline{Data}$ Polling must be performed at an address within any of the sectors being erased.

Figure 4 illustrates the Embedded Erase Algorithm using typical command strings and bus operations.

Table 13. Embedded Erase Algorithm

Bus Operation	Command	Comments
Standby		Wait for V_{CC} ramp
Write	Embedded Erase command sequence	6 bus cycle operation
Read		$\overline{Data}$ Poll or check $\overline{BUSY}$ ($RY/\overline{BY}$) to verify erasure

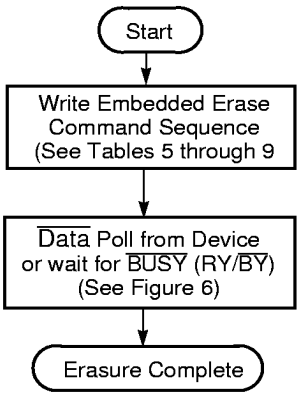


Figure 4. Embedded Erase Algorithm

Note: *The latest release of the software drivers for AMD Miniature Cards and devices may be downloaded from the AMD web site at http://www.amd.com.*

Embedded Program Algorithm

The Embedded Program setup is a four bus cycle operation that stages the addressed memory sector or memory device for automatic programming.

Once the Embedded Program setup operation is performed, the next $\overline{WE}$ (or $\overline{CE}$) pulse causes a transition to an active programming operation. Addresses are internally latched on the falling edge of the $\overline{WE}$ (or $\overline{CE}$) pulse. Data is internally latched on the rising edge of the $\overline{WE}$ pulse. The rising edge of $\overline{WE}$ also begins the programming operation. The system is not required to provide further control or timing. The device will automatically provide an adequate internally generated write pulse and verify margin. The automatic programming operation is completed when the data on D7 of the addressed memory sector or memory device is equivalent to data written to this bit (see Write Operation Status section) at which time the device returns to the Read mode (no write verify command is required).

Addresses are latched on the falling edge of $\overline{WE}$ during the Embedded Program command execution and hence the system is not required to keep the addresses stable during the entire Programming operation. However, once the device completes the Embedded Program operation, it returns to the Read mode and addresses are no longer latched. Therefore, the device requires that a valid address input to the device is supplied by the system at this particular instant of time. Otherwise, the system will never read a valid data on D7. A system designer has two choices to implement the Embedded Programming algorithm:

1. The system (CPU) keeps the address valid during the entire Embedded Programming operation, or
2. Once the system executes the Embedded Programming command sequence, the CPU takes away the address from the device and becomes free to do other tasks. In this case, the CPU is required to keep track of the valid address by loading

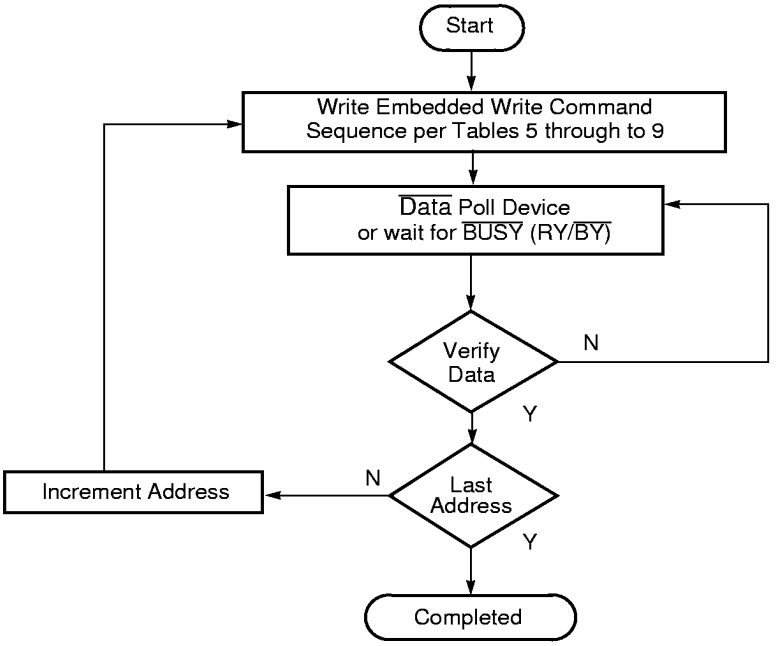
it into a temporary register. When the CPU comes back for performing Data Polling, it should reassert the same address.

3. The host may monitor BUSY (RY/BY) to determine the status of the Embedded Algorithm in progress. A "0" indicates that the device is busy; a "1" indicates that the algorithm is complete.

However, since the Embedded Programming operation takes only 8 µs typically, it may be easier for the CPU to keep the address stable during the entire Embedded Programming operation instead of reasserting the valid address during Data Polling. Anyway, this has been left to the system designer's choice to go for either operation. Any commands written to the device during this period will be ignored. Figure 5 and Table 14 illustrate the Embedded Program Algorithm, a typical command string, and bus operation.

Table 14. Embedded Program Algorithm

Bus Operation	Command	Comments
Standby		Wait for V_{CC} ramp
Write	Embedded Program command sequence	3 bus cycle operation
Write	Program Address/Data	1 bus cycle operation
Read		Data Poll or check BUSY (RY/BY) to verify program



20975C-6

Figure 5. Embedded Program Algorithm

Reset Command

The device automatically powers up in the read/reset state. A command sequence is not required to read data in this case. Standard microprocessor cycles retrieve array data. This default state ensures that no spurious alteration of the memory content occurs during the power transition. Refer to the AC Characteristics section for the specific timing parameters.

The 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.

Sector Erase Suspend

Sector Erase Suspend command allows the user to interrupt the chip and then do data reads (not program) from a non-busy sector while it is in the middle of a Sector Erase operation (which may take up to several seconds). This command is applicable ONLY during the Sector Erase operation and will be ignored if written during the chip Erase or Programming operation. The Erase Suspend command (B0H) will be allowed only during the Sector Erase Operation that will include the sector erase time-out period after the Sector Erase commands (30H). Writing this command during the time-out will result in immediate termination of the time-out period. Any subsequent writes of the Sector Erase command will be ignored as such, but instead will be taken as the Erase Resume command. Note that any other commands during the time out will reset the device to read mode. The addresses are don't-cares in writing the Erase Suspend or Erase Resume commands.

When the Sector Erase Suspend command is written during a Sector Erase operation, the chip will take between 0.1 µs to 10 µs to suspend the erase operation and go into erase suspended read mode (pseudo-read mode), during which the user can read from a sector that is NOT being erased. A read from a sector being erased may result in invalid data. The user must monitor D6 to determine if the chip has entered the pseudo-read mode, at which time D6 stops toggling. An address of a sector NOT being erased must be used to read D6, otherwise the user may encounter intermittent problems. Note that the user must keep track of what state the chip is in since there is no external indication of whether the chip is in pseudo-read mode or actual read mode. After the user writes the Sector Erase Suspend command and waits until D6 stops toggling, data reads from the device may then be performed. Any further writes of the Sector Erase Suspend command at this time will be ignored.

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 ignore. Another Sector Erase Suspend command can be written after the chip has resumed.

Write Operation Status

Table 15 shows the status bit states for device program and erase operations.

$\overline{\text{Data}}$ Polling—D7 (D15 on Odd Byte)

The AMD Flash Miniature Card features $\overline{\text{Data}}$ Polling as a method to indicate to the host system that the Embedded algorithms are either in progress or completed (The host may alternatively monitor $\overline{\text{BUSY}}$ (RY/$\overline{\text{BY}}$).

While the Embedded Programming algorithm is in operation, an attempt to read the device will produce the complement of expected valid data on D7 of the addressed memory sector or memory device. Upon completion of the Embedded Program algorithm an attempt to read the device will produce valid data on D7. The $\overline{\text{Data}}$ Polling feature is valid after the rising edge of the fourth $\overline{\text{WE}}$ pulse of the four write pulse sequence.

While the Embedded Erase algorithm is in operation, D7 will read "0" until the erase operation is completed. Upon completion of the erase operation, the data on D7 will read "1".

The $\overline{\text{Data}}$ Polling feature is only active during the Embedded Programming or Erase algorithms. Please note that D7 may change asynchronously while Output Enable ($\overline{\text{OE}}$) is asserted low. This means that the device is driving status information on D7 at one instant of time and then the byte's valid data at the next instant of time. Depending on when the system samples the D7 output, it may read either the status or valid data. Even if the device has completed the Embedded operation and D7 has a valid data, the data outputs on D0-D6 may be still invalid since the switching time for data bits (D0-D7) will not be the same. This happens since the internal delay paths for data bits (D0-D7) within the device are different. The valid data will be provided only after a certain time delay (>t_{OE}). Please refer to Figure 9 for a detailed timing diagram. See Figure 6 for the $\overline{\text{Data}}$ Polling algorithm.

Toggle Bit—D6 (D14 on Odd Byte)

The toggle bit is used for entering the Erase Suspend mode. Refer to the previous section entitled "Sector Erase Suspend" and Table 15 for information on this bit.

Table 15. Hardware Sequence Flags

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 (Note 1)
		Erase Suspend Read (Non-Erase Suspended Sector)	Data	Data	Data	Data	Data
		Erase Suspend Program (Non-Erase Suspended Sector)	D7	Toggle (Note 2)	0	1	1 (Note 3)
Exceeded Time Limits	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:

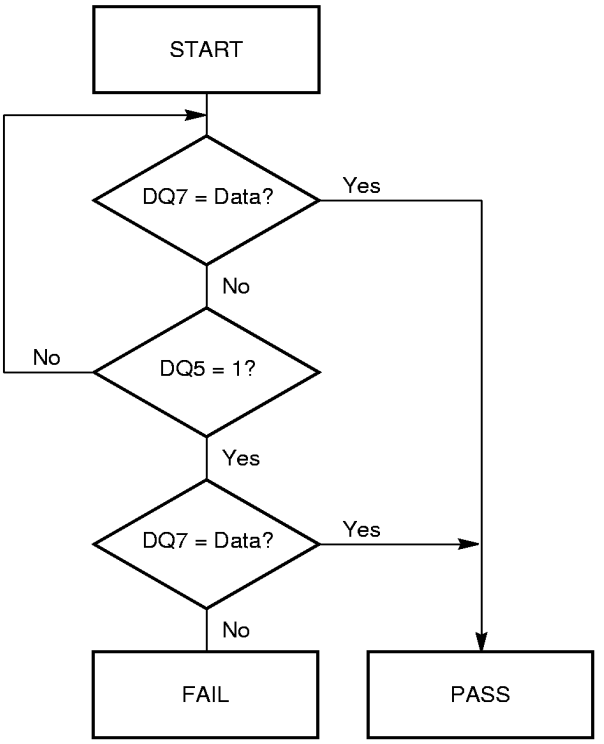
1. *Performing successive read operations from the erase-suspended sector will cause D2 to toggle.*
2. *Performing successive read operations from any address will cause D6 to toggle.*
3. *Reading the byte address being programmed while in the erase-suspend program mode will indicate logic "1" at the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

BUSY (RY/BY—Ready/Busy)

The BUSY signal indicates to the host the status of operations within the Miniature Card. The BUSY signal is tied to the components' RY/BY pins.

The RY/BY signal from AMD Flash devices in the Miniature Card indicate 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 the 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 a Flash device is placed in an Erase Suspend mode, the RY/BY output will be high. Refer to the section "Sector Erase Suspend" for more information.

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 should be ignored while RESET is at V_{IL}.



20975C-7

Note: *D7 is rechecked even if D5 = 1 because D7 may change simultaneously with D5.*

Figure 6. $\overline{\text{Data}}$ Polling Algorithm

WORD-WIDE PROGRAMMING AND ERASING

Word-Wide Programming

The Word-Wide Programming sequence will be as usual per Table 5 or 8. The Program word command is A0A0H. Each byte is independently programmed. For example, if the high byte of the word indicates the successful completion of programming via one of its write status bits such as D15, software polling should continue to monitor the low byte for write completion and data verification, or vice versa. During the Embedded Programming operations the device executes programming pulses in 8 µs increments. Status reads provide information on the progress of the byte programming relative to the last complete write pulse. Status information is automatically updated upon completion of each internal write pulse. Status information does not change within the 8 µs write pulse width.

Word-Wide Sector Erasing

The Word-Wide Sector Erasing of a memory device pair is similar to word-wide programming. The erase word command is a six-bus-cycle command sequence (see Tables 5 and 8). Each byte is independently erased and verified. Word-wide erasure reduces total erase time when compared to byte erasure. Each Flash memory device in the card may erase at different rates. Therefore, each device (byte) must be verified separately.

ABSOLUTE MAXIMUM RATINGS

Storage Temperature . . . . . –30°C to +70°C

Ambient Temperature
with Power Applied. . . . . –10°C to +70°C

Voltage at All Pins (Note 1) . . . . . –0.5 V to +7.0 V

V_{CC} (Note 1) . . . . . –2.0 V to 7.0 V

Output Short Circuit Current (Note 2) . . . . . 200 mA

Notes:

1. *Minimum DC voltage on input or I/O pins is –0.5 V. During voltage transitions, inputs may overshoot V_{SS} to –2.0 V for periods of up to 20 ns. Maximum DC voltage on output and I/O pins is V_{CC} + 0.5 V. During voltage transitions, outputs may overshoot to V_{CC} + 2.0 V for periods up to 20ns.*
2. *No more than one output shorted at a time. Duration of the short circuit should not be greater than one second. Conditions equal V_{OUT} = 0.5 V or 5.0 V, V_{CC} = V_{CCmax}. These values are chosen to avoid test problems caused by tester ground degradation. This parameter is sampled and not 100% tested, but guaranteed by characterization.*
3. *Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.*

OPERATING RANGES

Commercial (C) Devices

Case Temperature (T_C). . . . . .0°C to +70°C

V_{CC} Supply Voltages. . . . . +4.5 V to +5.5 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS

Parameter Symbol	Parameter Description	Test Conditions		Min	Max	Unit
I_{LI}	Input Leakage Current	$V_{IN} = V_{SS}$ to V_{CC}, $V_{CC} = V_{CC\ max}$			±5	µA
I_{LO}	Output Leakage Current	$V_{IN} = V_{SS}$ to V_{CC}, $V_{CC} = V_{CC\ max}$			±5	µA
I_{CCS}	V_{CC} Standby Current	$\overline{CEL}$, $\overline{CEH}$, $\overline{RESET} = V_{IH}$			4	mA
I_{CC}	V_{CC} Supply Current (Note 2)	$\overline{RESET} = V_{IH}$; $\overline{CEL}$ and $\overline{CEH} = V_{IL}$	Read		80	mA
			Program		120	mA
I_{CC}	V_{CC} Standby Current	$CE = V_{CC} \pm 0.3V$			60	µA
V_{IL}	Input Low Voltage	$V_{CC} = 5.0$ V		–0.5	0.8	V
V_{IH}	Input High Voltage			0.7 V_{CC}	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	$I_{OUT} = 12$ mA			0.1 V_{CC}	V
V_{OH}	Output High Voltage	$I_{OUT} = –2.5$ mA		0.9 V_{CC}		V
V_{LKO}	Low V_{CC} Lock-Out Voltage			3.2	4.2	V

Notes:

1. *V_{CC} = 5.0 volts ± 10%*
2. *Supply current is a max RMS value.*

CONNECTOR DC SPECIFICATIONS

Parameter	Min	Max	Units
Interface Signal Resistance (Note 2)		2.0	Ω
Interface Signal Current (Notes 1, 2)	125		mA
Power/Insertion Signal Resistance		0.060	Ω
Power/Insertion Signal Current (Note 1)	500		mA

Notes:

1. *This current is a minimum that the connector should withstand, and a maximum that the host should provide.*
2. *On the host, these specifications must be met for one conducting channel on elastomeric connectors.*

CARD AND PAD CAPACITANCE

Parameter Symbol	Parameter Description	Test Conditions	Max	Unit
C_{CARD}	Card Input Capacitance		40	pF
C_{HOST}	System Load Capacitance		120	pF
$C_{I/O}$	I/O Capacitance D0–D15		40	pF

Notes:

1. *Sampled, not 100% tested.*
2. *Test conditions $T_A = 25°C$, f = 1.0 MHz.*

AC CHARACTERISTICS

Read-only Operations

Parameter Symbol		Parameter Description	Card Speed 150 ns		Unit
JEDEC	Standard		Min	Max	
t_{AVAV}	t_{RC}	Read Cycle Time	150		ns
t_{ELQV}	t_{CE}	Chip Enable Access Time		150	ns
t_{AVQV}	t_{ACC}	Address Access Time		150	ns
t_{GLQV}	t_{OE}	Output Enable Access Time		50	ns
t_{ELQX}	t_{LZ}	Chip Enable to Output in Low-Z	0		ns
t_{EHQZ}	t_{DF}	Chip Disable to Output in High-Z		30	ns
t_{GLQX}	t_{OLZ}	Output Enable to Output in Low-Z	0		ns
t_{GHQZ}	t_{DF}	Output Disable to Output in High-Z		30	ns
t_{AXQX}	t_{OH}	Output Hold from First of Address, $\overline{CE}$, or $\overline{OE}$ Change	0		ns
	t_{Ready}	$\overline{RESET}$ Pin Low to Read Mode*		20	µs

* *Not 100% tested.*

AC CHARACTERISTICS

Write Operations (Erase/Program)

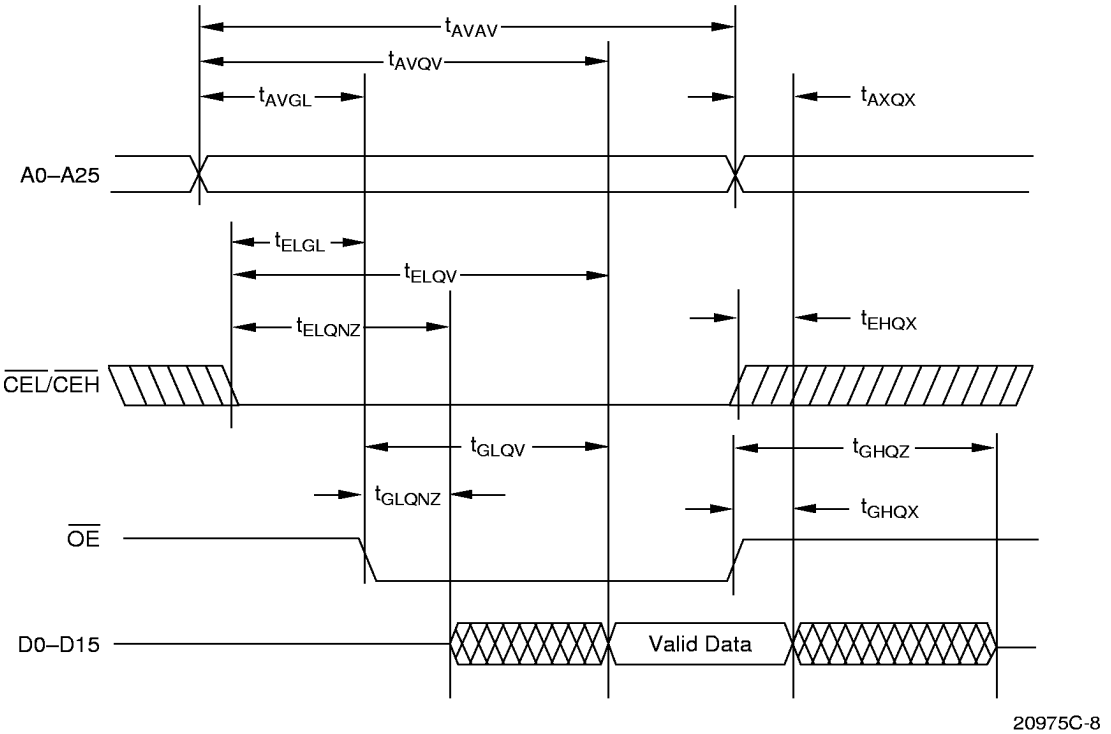
Parameter Symbols			Card Speed			
			150 ns			
JEDEC	Standard	Parameter Description	Min	Typ	Max	Unit
t_{AVAV}	t_{RC}	Write Cycle Time	150			ns
t_{ELQV}	t_{CE}	Chip Enable Access Time			150	ns
t_{AVQV}	t_{ACC}	Address Access Time			150	ns
t_{GLQV}	t_{OE}	Output Enable Access Time			50	ns
t_{ELQX}	t_{LZ}	Chip Enable to Output in Low-Z	0			ns
t_{EHQZ}	t_{DF}	Chip Disable to Output in High-Z			30	ns
t_{GLQX}	t_{OLZ}	Output Enable to Output in Low-Z	0			ns
t_{GHQZ}	t_{DF}	Output Disable to Output in High-Z			30	ns
t_{AXQX}	t_{OH}	Output Hold From First of Address, $\overline{CE}$, or $\overline{OE}$ Change	0			ns
t_{WHGL}		Write Recovery Time Before Read	0			µs
t_{GLQNZ}		Data Output enable time from $\overline{OE}$ active	0			ns
t_{ELQNZ}		Data Output enable time from $\overline{CE}$ active	0			ns
t_{WLWH}	t_{WP}	$\overline{WE}$ pulse width	50			ns
t_{ELGL} t_{ELWL}		$\overline{CE}$ setup time to $\overline{WE}$ or $\overline{OE}$ active	0			ns
t_{AVGL} t_{AVWL}		Address setup time to $\overline{WE}$ or $\overline{OE}$ active	0			ns
t_{DVWH}	t_{DS}	Data setup time to $\overline{WE}$ inactive	50			ns
t_{WHDX}		Data hold time from $\overline{WE}$ inactive	0			ns
t_{WHAX}		Address hold time from $\overline{WE}$ inactive	0			ns
t_{WHEH}		$\overline{CE}$ hold time from $\overline{WE}$ inactive	0			ns
	t_{RP}	$\overline{RESET}$ Pulse Width	500			ns
	t_{BUSY}	Program/Erase Valid to RY/$\overline{BY}$ Delay	50			ns
t_{WHWH1}		Programming Operation		8	300	µs
t_{WHWH2}		Sector Erase Operation		1	1.5	s

KEY TO SWITCHING WAVEFORMS

WAVEFORM	INPUTS	OUTPUTS
	Must be Steady	Will be Steady
	May Change from H to L	Will be Changing from H to L
	May Change from L to H	Will be Changing from L to H
	Don't Care, Any Change Permitted	Changing, State Unknown
	Does Not Apply	Center Line is High-Impedance "Off" State

KS000010

SWITCHING WAVEFORMS



20975C-8

Figure 7. AC Waveforms for Read Operations

SWITCHING WAVEFORMS

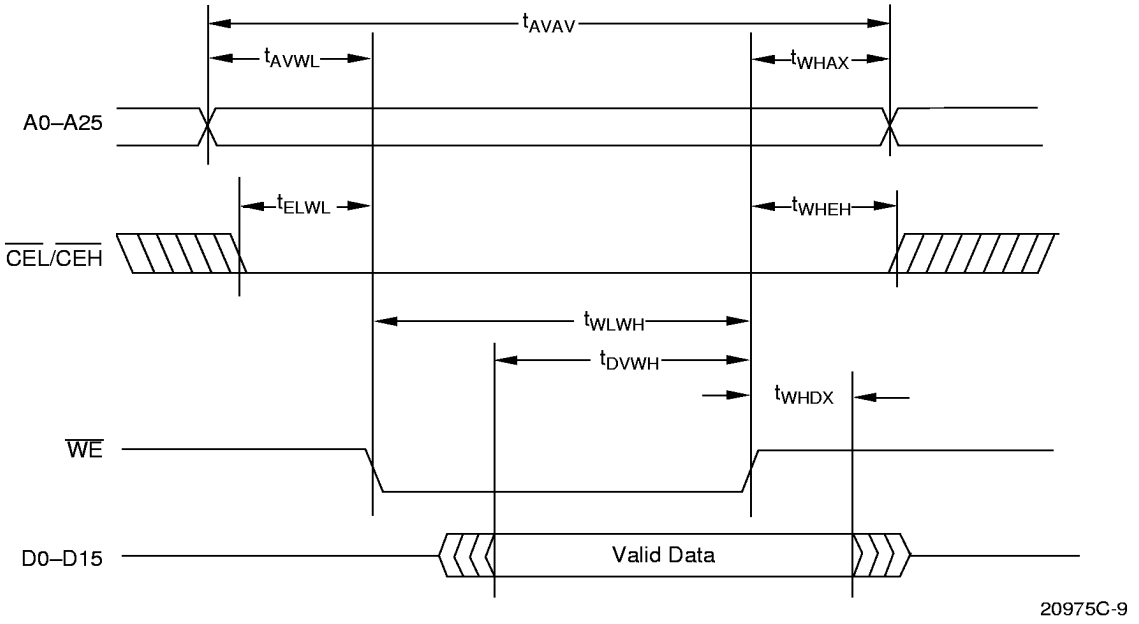
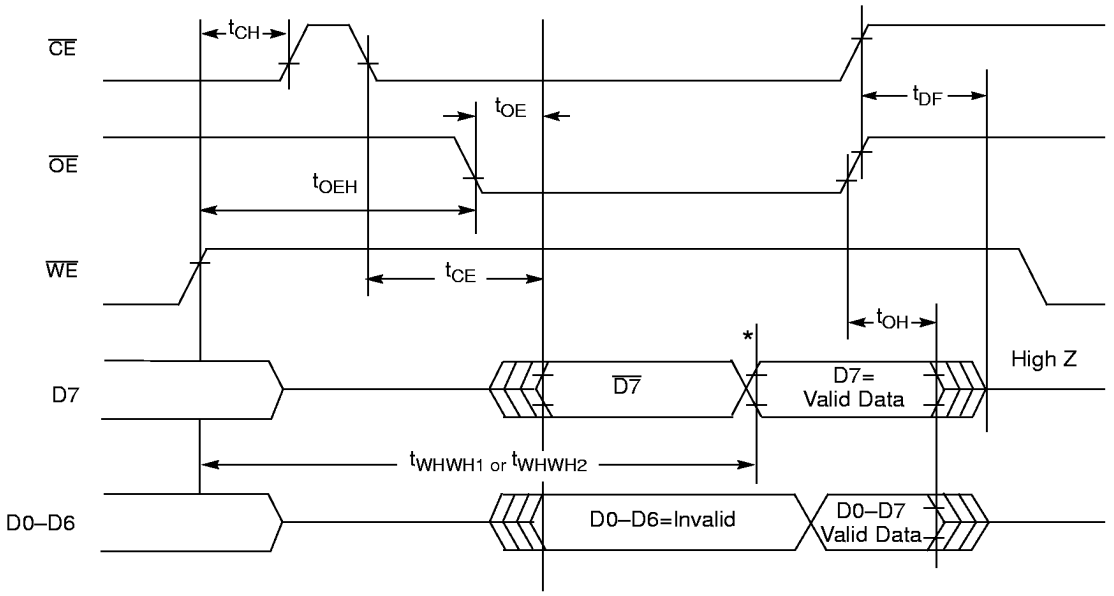


Figure 8. AC Waveforms for Write Operations



**D7=Valid Data (The device has completed the Embedded operation).*

Figure 9. AC Waveforms for $\overline{\text{Data}}$ Polling During Embedded Algorithm Operations

SWITCHING WAVEFORMS

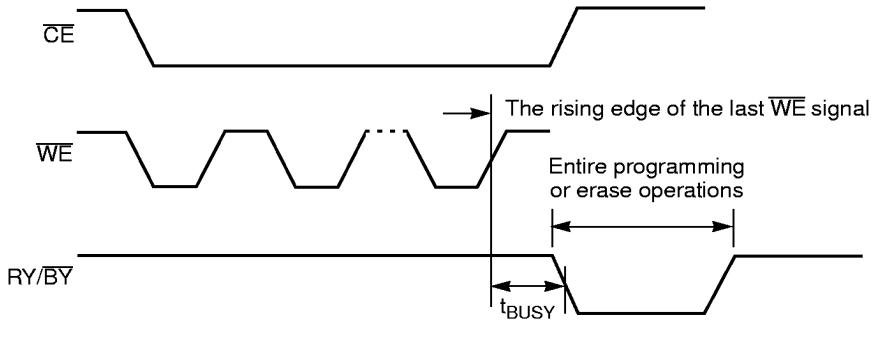


Figure 10. RY/BY Timing Diagram During Program/Erase Operations

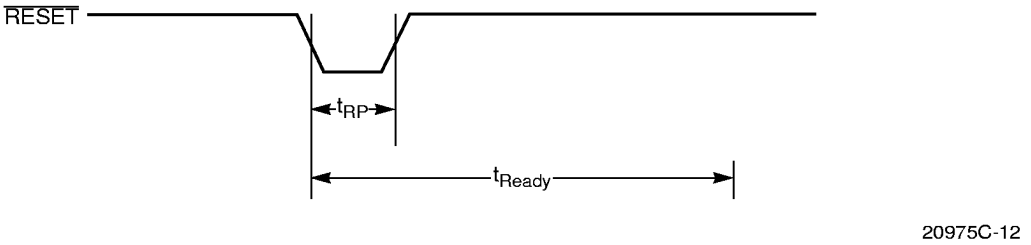


Figure 11. RESET Timing Diagram

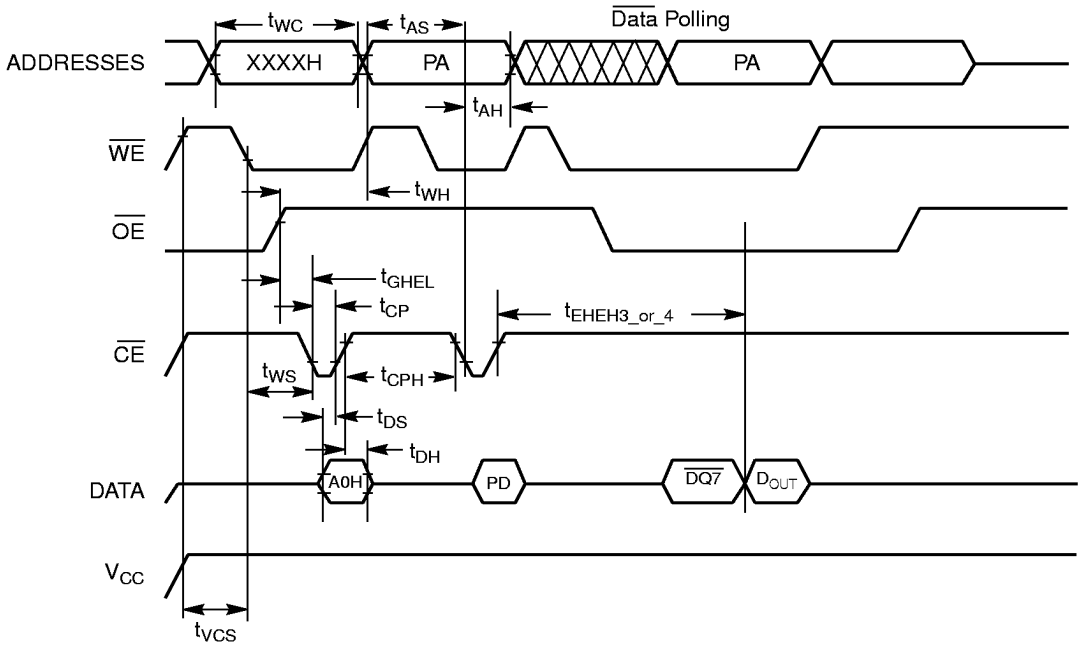
AC CHARACTERISTICS—ALTERNATE $\overline{CE}$ CONTROLLED WRITES

Write/Erase/Program Operations

Parameter Symbols			Card Speed 150 ns			
JEDEC	Standard	Parameter Description	Min	Typ	Max	Unit
t_{AVAV}	t_{WC}	Write Cycle Time	150			ns
t_{AVEL}	t_{AS}	Address Setup Time	20			ns
t_{ELAX}	t_{AH}	Address Hold Time	50			ns
t_{DVEH}	t_{DS}	Data Setup Time	50			ns
t_{EHDX}	t_{DH}	Data Hold Time	20			ns
t_{GLDV}	t_{OEH}	Output Enable Hold Time for Embedded Algorithm	10			ns
t_{GHEL}		Read Recovery Time before Write	0			µs
t_{WLEL}	t_{WS}	$\overline{WE}$ Setup Time before $\overline{CE}$	0			ns
t_{EHWH}	t_{WH}	$\overline{WE}$ Hold Time	0			ns
t_{ELEH}	t_{CP}	$\overline{CE}$ Pulse Width	50			ns
t_{EHEL}	t_{CPH}	$\overline{CE}$ Pulse Width HIGH (Note 2)	20			ns
t_{EHEH3}		Embedded Programming Operation (Notes 2)		8	300	µs
t_{EHEH4}		Embedded Erase Operation for each 64K byte Memory Sector (Notes 1)		1	1.5	s
t_{VCS}		V_{CC} Setup Time to Write Enable LOW	50			µs

Notes:

1. *Rise/fall time ≤10 ns.*
2. *Card Enable Controlled Programming:*
   Flash Programming is controlled by the valid combination of the Card Enable ($\overline{CE1}$, $\overline{CE2}$) and Write Enable (WE) signals. For systems that use the Card Enable signal(s) to define the write pulse width, all setup, hold, and inactive write enable timing should be measured relative to the Card Enable signal(s)/



Notes:

1. *PA is address of the memory location to be programmed.*
2. *PD is data to be programmed at byte address.*
3. $\overline{DQ7}$ *is the output of the complement of the data written to the device.*
4. D_{OUT} *is the output of the data written to the device.*
5. *Figure indicates last two bus cycles of four bus cycle sequence.*
6. *These waveforms are for the x16 mode.*

20975C-13

Figure 12. Alternate $\overline{CE}$ Controlled Write Operation Timings

AIS MEMORY MAP

The AIS (Attribute Information Structure) is an area of memory used for storing information about the configuration of the Miniature Card. The AIS is recommended to be stored in the first sector of the first device of the Flash array. As this area is not explicitly protected, the AIS information must be reloaded onto the card in the event that the information is erased.

The AIS has five unique information areas:

1. Identification Data: This data includes Manufacturer information (Manufacturer and card name).
2. Compatibility Data: This data specifies basic information about the card (memory size, access time, memory type, power, etc.)
3. Burst Data (not applicable)
4. DRAM Data (not applicable)
5. Reserved Data: This data area is reserved for future use.

The AIS supports up to four different memory technologies on a card. Some of the information areas are repeated in the memory map in order to specify different technologies (see Table 16). The Technology Count field in the Identification Data section defines the number of different technologies on a card. The first memory technology is defined in the AIS memory map from address 40H through 7FH. The second memory technology is defined from 80H through BFH. The third memory technology is defined from C0H to DFH. The fourth memory technology is defined from E0H to FFH.

The AIS is stored as bytes within the 16-bit Miniature Card data word. The even byte D0–D7 stores the AIS data, and the odd byte D8–D15 is reserved by the card manufacturer for manufacturing information.

Table 16. Miniature Card AIS Memory Assignments

Card Address	Section	Description
00H–0FH	PC Card Compatibility Area*	Reserved for PC Card Tuples
10H–1FH	Identification Data Identifies Card Type	
20H–2FH	Identification Data Identifies Card Type	
30H–3FH	Identification Data Identifies Card Type	
40H–4FH	Compatibility Data (Area 1)	Memory Technology #1
50H–5FH	Burst Data (not applicable)	
60H–6FH	DRAM Data (not applicable)	
70H–7FH	Reserved for future use	
80H–8FH	Compatibility Data (not applicable)	(Memory Technology #2)
90H–9FH	Burst Data (not applicable)	
A0H–AFH	DRAM Data (not applicable)	
B0H–BFH	Reserved for future use	
C0H–CFH	Compatibility Data (not applicable)	(Memory Technology #3)
D0H–DFH	Reserved for future use	
E0H–EFH	Compatibility Data (not applicable)	(Memory Technology #4)
F0H–FFH	Reserved for future use	

** For more information on PC Card Compatibility refer to Table 17 or the Miniature Card PC Compatibility Guide.*

Note: *"Not applicable" indicates the address space does not apply to AMD Flash Miniature Cards, but is defined by MCIF.*

Table 17. PC Card Compatibility Memory Assignments

Address	Values	Description
00h	01h	TPL_CODE CISTPL_DEVICE
07h	03h	TPL_LINK
02h	53	Device ID
03h	2MB (5V)=7C, 4MB=FC (5V); 8MB (5V) = 1E	Device Size
04h	FF	End of CISTPL_DEVICE
05h	00h	CISTPL_NULL
06h	00h	CISTPL_NULL
07h	00h	CISTPL_NULL
08h	00h	CISTPL_NULL
09h	00h	CISTPL_NULL
0Ah	00h	CISTPL_NULL
0Bh	00h	CISTPL_NULL
0Ch	00h	CISTPL_NULL
0Dh	00h	CISTPL_NULL
0Eh	80h	TPL_CODE CISTPL_MINI
0Fh	F0h	TPL_LINK

Identification Data

The Identification Data provides basic identification information about the card. This data section is required on all cards. Table 18 shows the Identification Data for AMD's 5 volt-only Miniature cards.

Compatibility Data

The compatibility data provides basic compatibility across all cards. This data section is required on all cards. The addresses in parentheses are specified for cards with more than one memory technology on the card. Table 19 shows the compatibility data for AMD 5-volt only Miniature Cards.

Table 18. AMD Identification Data

Card Address	Value	Description
10H	99H	**Miniature Card Identifier:** Fixed value for a host to identify an inserted Miniature Card
11H	11H	**Level of Compliance:** Defines the level of AIS supported. The Miniature Cards described in this document are rev 1.1 compliant.
12H	01H or FDH or F9H	**AIS Checksum:** The modulo-256 sum of all even bytes from 10H–FFH. A valid checksum sums to 00H (2's complement). 2 MByte card: 99H + 01H = 00H 4 MByte card: 03H + FDH = 00H 8 MByte card: 07H + F9H = 00H
13H	41H	**Manufacturer Name**: 13H–26H. String of ASCII characters at addresses 13H to 26H to identify the manufacturer of the Miniature Card. ASCII character "A"
14H	4DH	ASCII character "M"
15H	44H	ASCII character "D"
16H	20H	ASCII character - SPACE
17H	49H	ASCII character - "I"
18H	4EH	ASCII character - "N"
19H	43H	ASCII character - "C"
1AH	00H	ASCII character - NULL
1BH	00H	ASCII character - NULL
1CH–26H	00H	Unused space in manufacturer name field
27H	35H	**Card Name:** (addresses 27H–3AH). String of ASCII characters to identify the card name. ASCII character "5"
28H	56H	ASCII character "V"
29H	4DH	ASCII character "M"
2AH	43H	ASCII character "C"
2BH	20H	ASCII character - SPACE
2CH	53H	ASCII character "S"
2DH	65H	ASCII character "e"
2EH	72H	ASCII character "r"
2FH	69H	ASCII character "i"
30H	65H	ASCII character "e"

Table 18. AMD Identification Data (Continued)

Card Address	Value	Description
31H	73H	ASCII character “s”
32H	00H	ASCII character - NULL
33H–3AH	00H	Unused space in card name field
3BH	01H	**Technology Count:** Defines the number of different memory technologies on the Miniature Card. Technology count set to 1
3CH–3FH	00H	Reserved space set to 00H; for future use

Table 19. AMD Compatibility Data

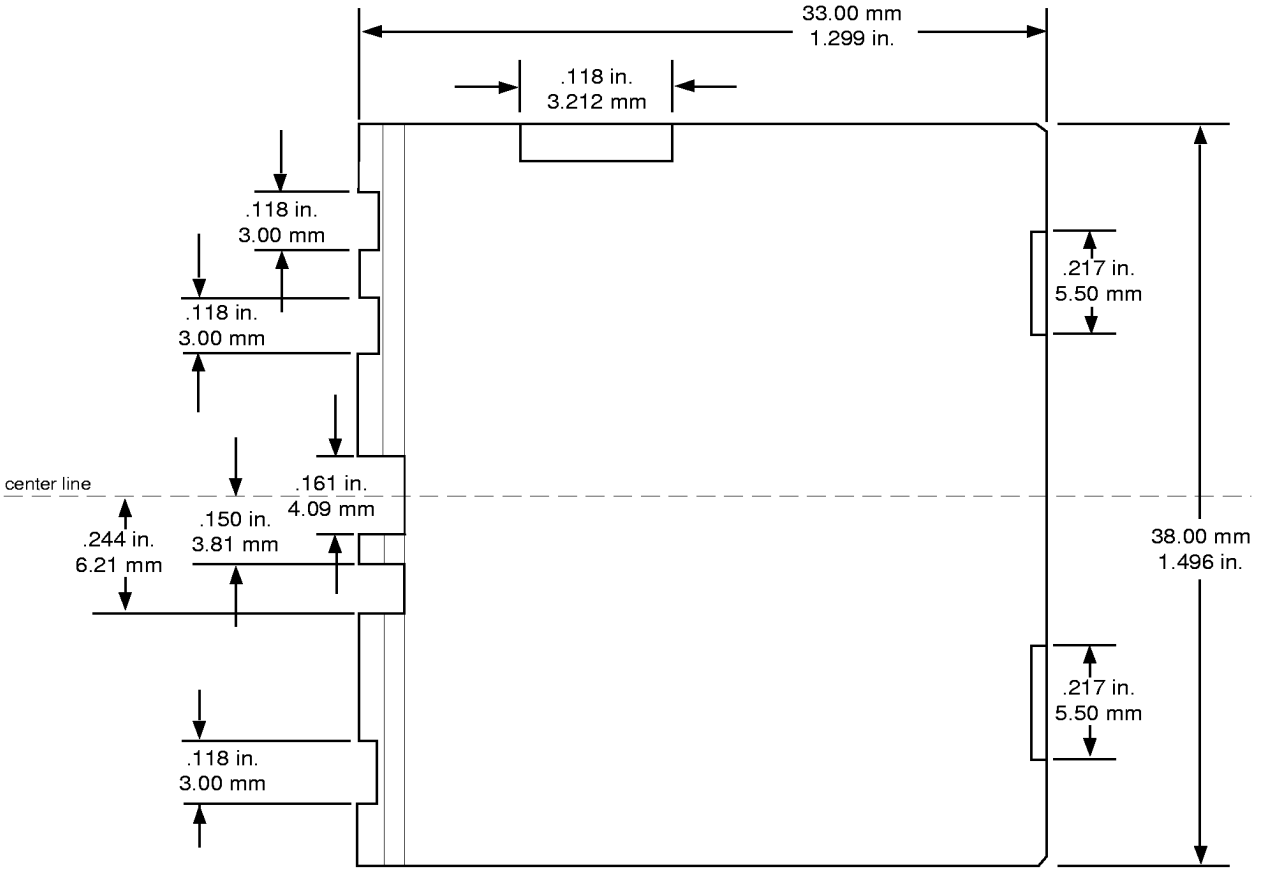
Card Address	Value	Description
40H	00H	Defines the type of memory technology; Flash = 000 binary
41H	01H	Device JEDEC Manufacturer ID
42H	D5H or 3DH	Device JEDEC Component ID: Am29F080 = D5H, Am29F016 = 3DH
43H	01H or 03H or 07H	Memory array size: 01 = 2 MByte, 03 = 4 MByte, 07 = 8 MByte
44H	00H	N/A
45H	00H	N/A
46H	0FH	5.0 Volt Access Time: 150 ns
47H	00H	N/A
48H	00H	N/A
49H	8CH	Typical read/write current at 5.0 Volts (word mode): 80mA read, 120 mA write
4AH	0AH	Typical standby current: 1 mA
4BH–4FH, 8CH–8FH, CCH–CFH, ECH–EFH	00H	Reserved for future use
80H–8BH, C0–CBH, E0H–EBH	00H	These addresses are designated for other memory technologies, which are not used in AMD Flash Miniature Cards.

Notes:

1. *All reserved bytes must be set to 00H. All reserved fields (bits) within bytes must be set to 0 (binary). All unused fields must be set to 00H.*

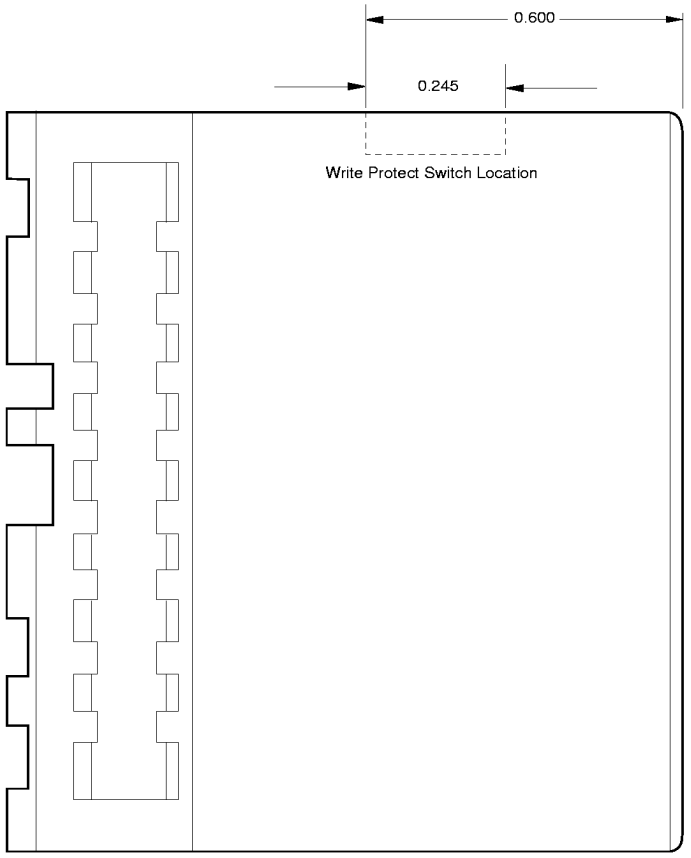
PHYSICAL DIMENSIONS

Top View

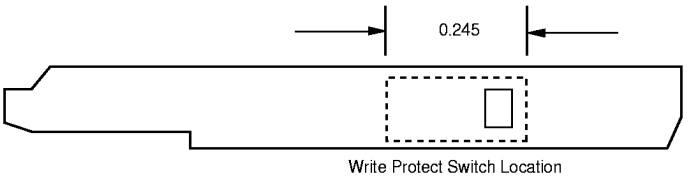


PHYSICAL DIMENSIONS

Bottom View



Right Side View



REVISION SUMMARY

Distinctive Characteristics

Under “Miniature Card standard form factor”, removed small form factor subheading. Updated typical CMOS parameters.

Block Diagram

Pullups added on $\overline{BUSY}$ and $\overline{RESET}$ signals. Revised notes.

Pad Assignments

A20-A24: Changed last sentence changed

$\overline{RESET}$: Miniature Card resets to the Read state.

$\overline{BUSY}$ - Removed last sentence requiring pullup by host.

Table 3, Voltage Sense Signals

Removed 3V-only row. Removed note.

Figure 2, Host/Card Address Connections

Changed decoder truth table entry for 1-9 combination.

Table 4, Read/Write Modes

Added in (*) note.

Erase Operations

Revised paragraph 4.

Data Protection

Removed 1st paragraph in section

Read Mode

Changed heading name from Read Enable.

Write Operations

Changed paragraph 4: added note about sector erase suspend and resume.

Table 12, Memory Sector Addresses

Corrected table.

Absolute Maximum Ratings

Changed output short circuit current and V_{CC} parameters.

DC Characteristics

Updated parameters in table. Added another I_{CC} row.

AC Characteristics

Removed output rise and fall time table.

Read-only operations table: Updated parameters. Changed read cycle time to write cycle time.

Write Operations table: Changed t_{LZ} and t_{OLZ} to 0 ns, added standard variables for t_{WP} and t_{DS}.

Switching Waveforms

Read/Reset Command changed to Reset Command.

AC Characteristics

Alternate $\overline{CE}$ Controlled Operations: updated parameters in table. Added typical specifications. Removed notes 2 and 4.

Table 18, AMD Identification Data

Changed values in 12h to 01h or FDh or F9h.

Table 19, AMD Compatibility Data

Changed value in 49h to 8Ch.