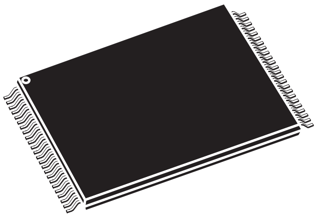
M29DW640F

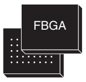
64 Mbit (8Mb x8 or 4Mb x16, Multiple Bank, Page, Boot Block) 3V Supply Flash Memory

Feature summary

- Supply voltage
  - V_{CC} = 2.7V to 3.6V for Program, Erase and Read
  - V_{PP} =12V for Fast Program (optional)
- Asynchronous Page Read mode
  - Page Width 8 Words
  - Page Access 25, 30ns
  - Random Access 60, 70ns
- Programming time
  - 10µs per Byte/Word typical
  - 4 Words / 8 Bytes at-a-time Program
- Memory blocks
  - Quadruple Bank Memory Array: 8Mbit+24Mbit+24Mbit+8Mbit
  - Parameter Blocks (at both Top and Bottom)
- Dual operations
  - While Program or Erase in a group of banks (from 1 to 3), Read in any of the other banks
- Program/Erase Suspend and Resume
  - Read from any Block during Program Suspend
  - Read and Program another Block during Erase Suspend
- Unlock Bypass Program command
  - Faster Production/Batch Programming
- $V_{PP}/\overline{WP}$ pin for Fast Program and Write Protect
- Temporary Block Unprotection mode
- Common Flash Interface
  - 64 bit Security Code
- Extended Memory Block
  - Extra block used as security block or to store additional information



TSOP48 (N)
12 x 20mm



TFBGA48 (ZE)
6 x 8 mm

- Low power consumption
  - Standby and Automatic Standby
- 100,000 Program/Erase cycles per block
- Electronic Signature
  - Manufacturer Code: 0020h
  - Device Code: 227Eh + 2202h + 2201
- ECOPACK® packages available

Contents

List of tables

List of figures

1 Summary description

The M29DW640F is a 64 Mbit (8Mb x8 or 4Mb x16) non-volatile memory that can be read, erased and reprogrammed. These operations can be performed using a single low voltage (2.7 to 3.6V) supply. On power-up the memory defaults to its Read mode.

The device features an asymmetrical block architecture, with 16 parameter and 126 main blocks, divided into four Banks, A, B, C and D, providing multiple Bank operations. While programming or erasing is underway in one group of banks (from 1 to 3), reading can be conducted in any of the other banks. The bank architecture is summarized in *Table 2*. Eight of the Parameter Blocks are at the top of the memory address space, and eight are at the bottom.

The M29DW640F has one extra 256 Byte block (Extended Block) that can be accessed using a dedicated command. The Extended Block can be protected and so is useful for storing security information. However the protection is irreversible, once protected the protection cannot be undone.

Each block can be erased independently, so it is possible to preserve valid data while old data is erased. The blocks can be protected to prevent accidental Program or Erase commands from modifying the memory. Program and Erase commands are written to the Command Interface of the memory. An on-chip Program/Erase Controller simplifies the process of programming or erasing the memory by taking care of all of the special operations that are required to update the memory contents. The end of a program or erase operation can be detected and any error conditions identified. The command set required to control the memory is consistent with JEDEC standards.

Chip Enable, Output Enable and Write Enable signals control the bus operation of the memory. They allow simple connection to most microprocessors, often without additional logic.

The memory is offered in TSOP48 (12x20mm), and TFBGA48 (6x8mm, 0.8mm pitch) packages.

In order to meet environmental requirements, Numonyx also offers the   in ECOPACK® packages. ECOPACK packages are Lead-free. The category of second Level Interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97. The maximum ratings related to soldering conditions are also marked on the inner box label.

The memory is supplied with all the bits erased (set to ’1’).

Figure 1. Logic diagram

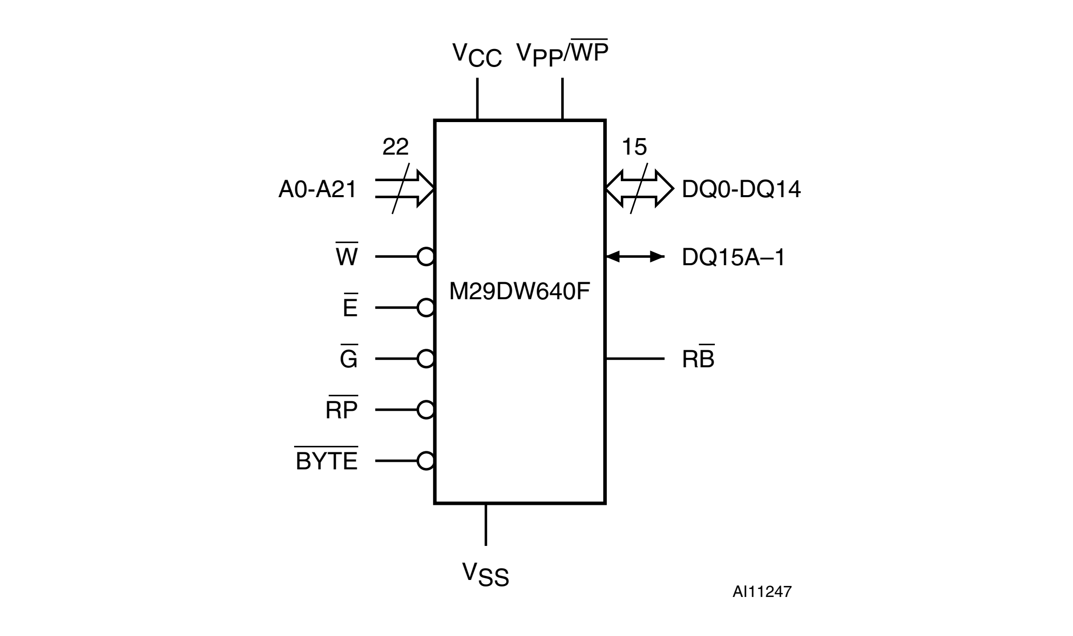


Table 1. Signal names

A0-A21	Address Inputs
DQ0-DQ7	Data Inputs/Outputs
DQ8-DQ14	Data Inputs/Outputs
DQ15A–1	Data Input/Output or Address Input
$\overline{E}$	Chip Enable
$\overline{G}$	Output Enable
$\overline{W}$	Write Enable
$\overline{RP}$	Reset/Block Temporary Unprotect
R$\overline{B}$	Ready/Busy Output
$\overline{BYTE}$	Byte/Word Organization Select
V_{CC}	Supply voltage
$V_{PP}/\overline{WP}$	V_{PP}/Write Protect
V_{SS}	Ground
NC	Not Connected Internally

Figure 2. TSOP connections

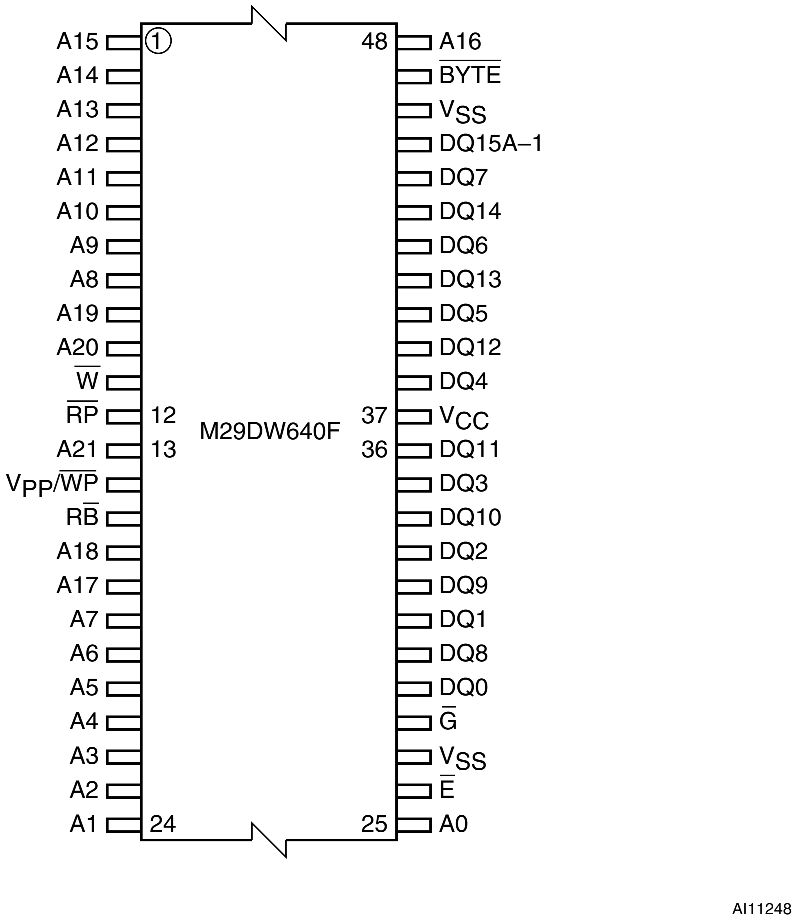
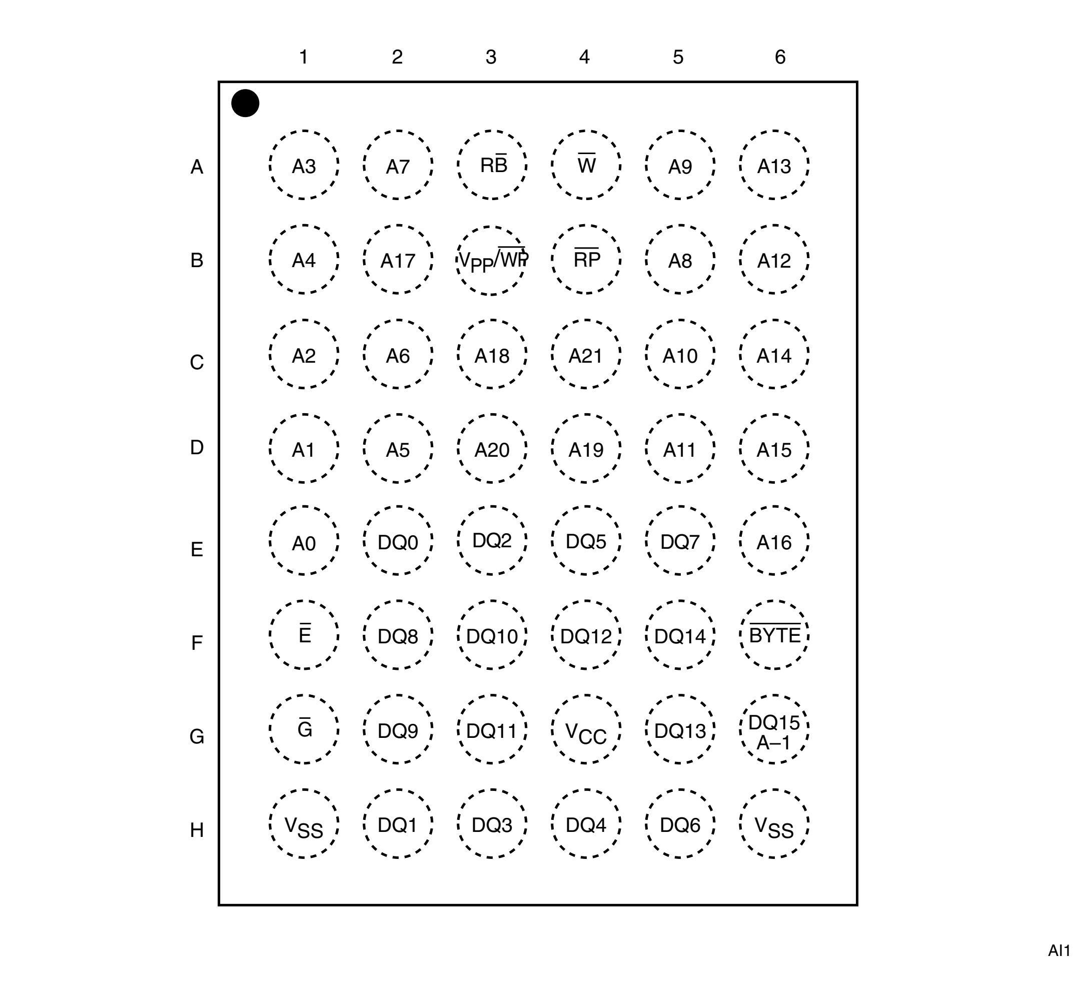


Figure 3. TFBGA48 connections (top view through package)



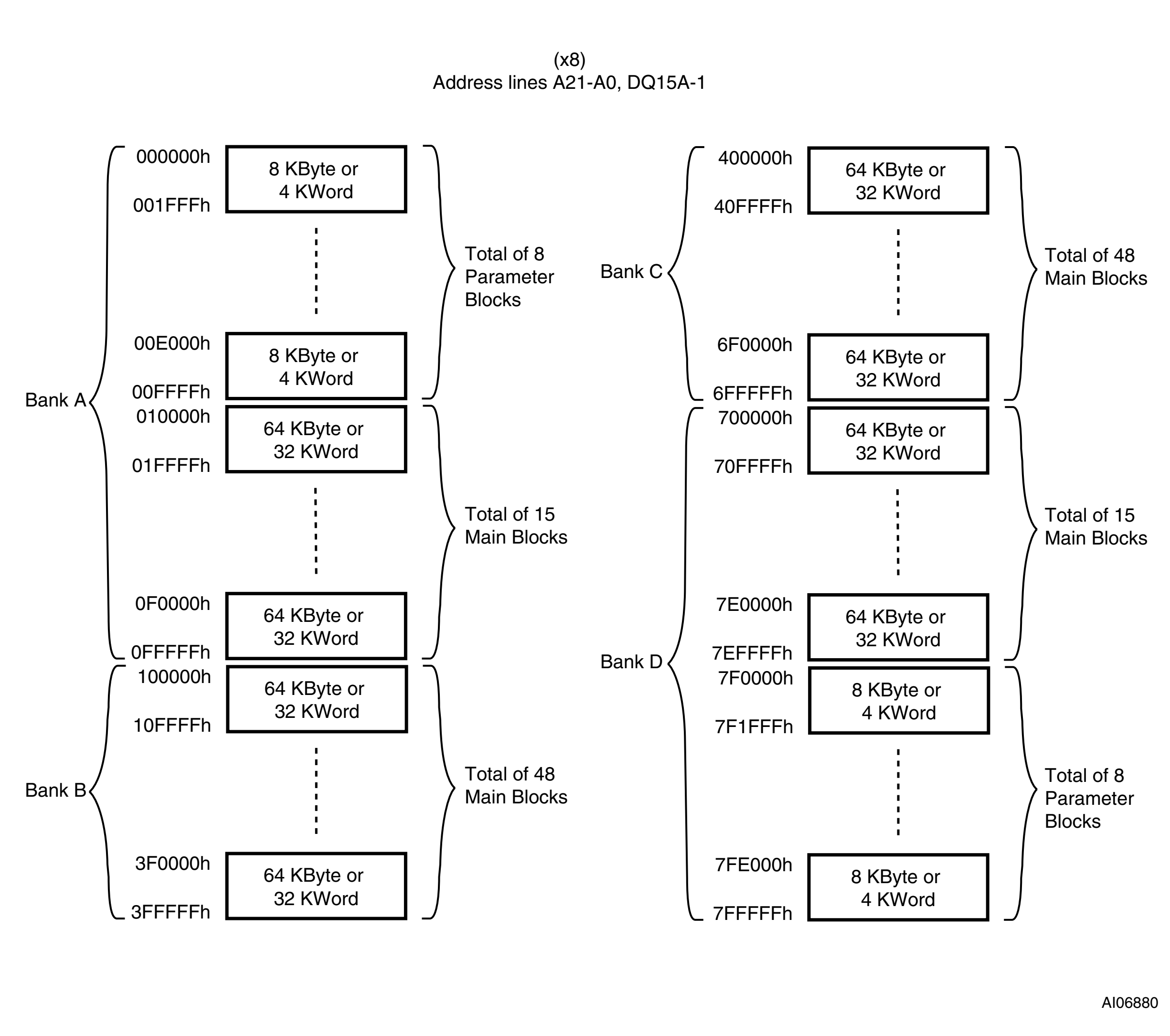
	1	2	3	4	5	6
A	A3	A7	R$\overline{\text{B}}$	$\overline{\text{W}}$	A9	A13
B	A4	A17	$V_{PP}/\overline{WP}$	$\overline{\text{RP}}$	A8	A12
C	A2	A6	A18	A21	A10	A14
D	A1	A5	A20	A19	A11	A15
E	A0	DQ0	DQ2	DQ5	DQ7	A16
F	$\overline{\text{E}}$	DQ8	DQ10	DQ12	DQ14	$\overline{\text{BYTE}}$
G	$\overline{\text{G}}$	DQ9	DQ11	V_{CC}	DQ13	DQ15 A–1
H	V_{SS}	DQ1	DQ3	DQ4	DQ6	V_{SS}

1. Balls are shorted together via the substrate but not connected to the die.

Table 2. Bank architecture

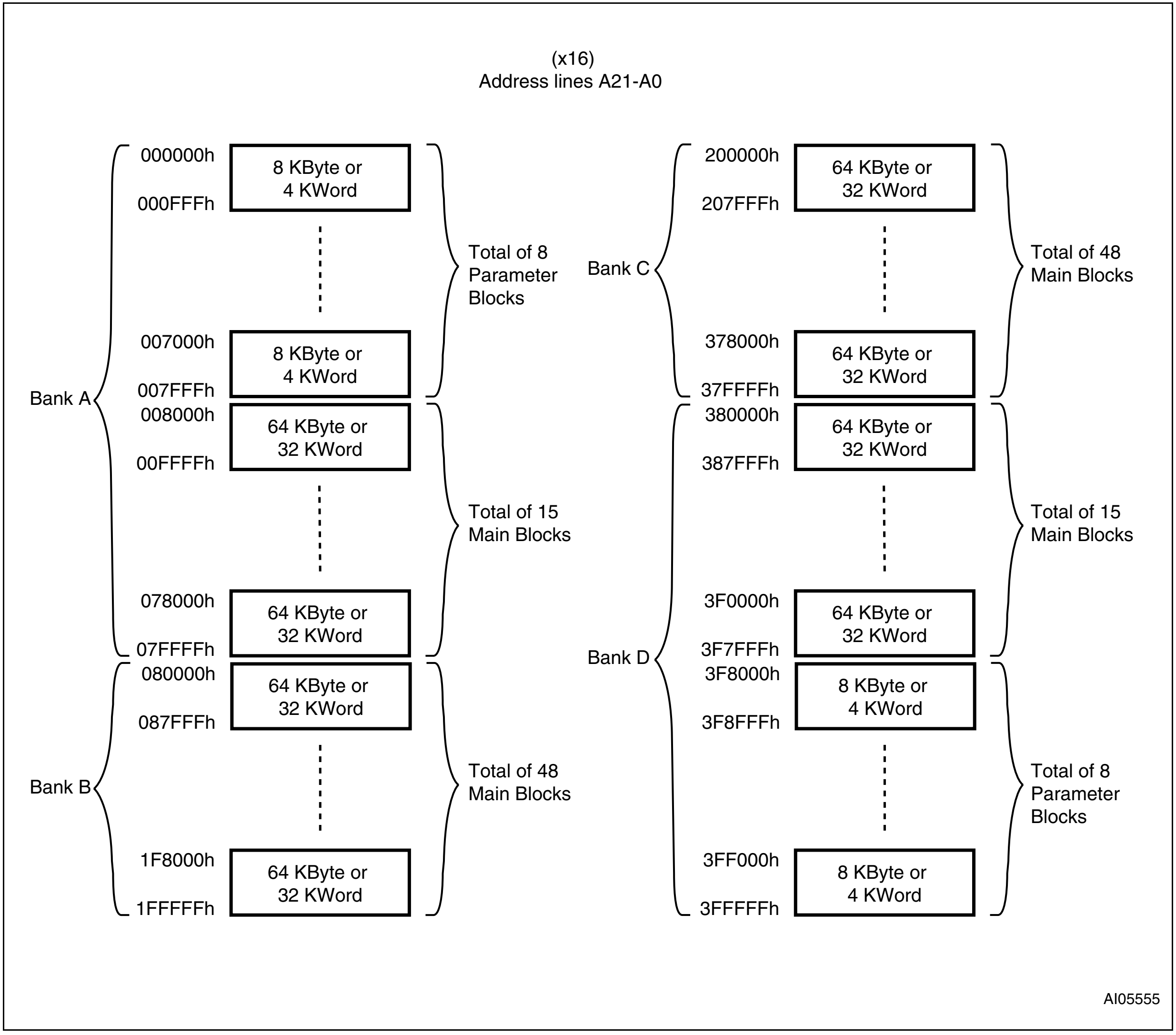
Bank	Bank Size	Parameter Blocks		Main Blocks	
		No. of Blocks	Block Size	No. of Blocks	Block Size
A	8 Mbit	8	8KByte/ 4 KWord	15	64KByte/ 32 KWord
B	24 Mbit	—	—	48	64KByte/ 32 KWord
C	24 Mbit	—	—	48	64KByte/ 32 KWord
D	8 Mbit	8	8KByte/ 4 KWord	15	64KByte/ 32 KWord

Figure 4. Block addresses (x8)



1. Also see *Appendix A*, *Table 24* for a full listing of the Block addresses.

Figure 5. Block addresses (x16)



1. Also see *Appendix A*, *Table 24* for a full listing of the Block addresses.

2 Signal descriptions

See *Figure 1: Logic diagram*, and *Table 1: Signal names*, for a brief overview of the signals connected to this device.

2.1 Address Inputs (A0-A21)

The Address Inputs select the cells in the memory array to access during Bus Read operations. During Bus Write operations they control the commands sent to the Command Interface of the internal state machine.

2.2 Data Inputs/Outputs (DQ0-DQ7)

The Data I/O outputs the data stored at the selected address during a Bus Read operation. During Bus Write operations they represent the commands sent to the Command Interface of the internal state machine.

2.3 Data Inputs/Outputs (DQ8-DQ14)

The Data I/O outputs the data stored at the selected address during a Bus Read operation when $\overline{\text{BYTE}}$ is High, V_{IH}. When $\overline{\text{BYTE}}$ is Low, V_{IL}, these pins are not used and are high impedance. During Bus Write operations the Command Register does not use these bits. When reading the Status Register these bits should be ignored.

2.4 Data Input/Output or Address Input (DQ15A–1)

When $\overline{\text{BYTE}}$ is High, V_{IH}, this pin behaves as a Data Input/Output pin (as DQ8-DQ14). When $\overline{\text{BYTE}}$ is Low, V_{IL}, this pin behaves as an address pin; DQ15A–1 Low will select the LSB of the addressed Word, DQ15A–1 High will select the MSB. Throughout the text consider references to the Data Input/Output to include this pin when $\overline{\text{BYTE}}$ is High and references to the Address Inputs to include this pin when $\overline{\text{BYTE}}$ is Low except when stated explicitly otherwise.

2.5 Chip Enable ($\overline{\text{E}}$)

The Chip Enable, $\overline{\text{E}}$, activates the memory, allowing Bus Read and Bus Write operations to be performed. When Chip Enable is High, V_{IH}, all other pins are ignored.

2.6 Output Enable ($\overline{\text{G}}$)

The Output Enable, $\overline{\text{G}}$, controls the Bus Read operation of the memory.

2.7 Write Enable ($\overline{W}$)

The Write Enable, $\overline{W}$, controls the Bus Write operation of the memory's Command Interface.

2.8 V_{PP}/Write Protect ($V_{PP}/\overline{WP}$)

The V_{PP}/Write Protect pin provides two functions. The V_{PP} function allows the memory to use an external high voltage power supply to reduce the time required for Program operations. This is achieved by bypassing the unlock cycles and/or using the multiple Word (2 or 4 at-a-time) or multiple Byte Program (2, 4 or 8 at-a-time) commands. The Write Protect function provides a hardware method of protecting the four outermost boot blocks (two at the top, and two at the bottom of the address space).

When V_{PP}/Write Protect is Low, V_{IL}, the memory protects the four outermost boot blocks; Program and Erase operations in these blocks are ignored while V_{PP}/Write Protect is Low, even when $\overline{RP}$ is at V_{ID}.

When V_{PP}/Write Protect is High, V_{IH}, the memory reverts to the previous protection status of the four outermost boot blocks (two at the top, and two at the bottom of the address space). Program and Erase operations can now modify the data in these blocks unless the blocks are protected using Block Protection.

Applying V_{PPH} to the $V_{PP}/\overline{WP}$ pin will temporarily unprotect any block previously protected (including the four outermost parameter blocks) using a High Voltage Block Protection technique (In-System or Programmer technique). See *Table 3: Hardware protection* for details.

When V_{PP}/Write Protect is raised to V_{PP} the memory automatically enters the Unlock Bypass mode. When V_{PP}/Write Protect returns to V_{IH} or V_{IL} normal operation resumes. During Unlock Bypass Program operations the memory draws I_{PP} from the pin to supply the programming circuits. See the description of the Unlock Bypass command in the Command Interface section. The transitions from V_{IH} to V_{PP} and from V_{PP} to V_{IH} must be slower than t_{VHVPP}, see *Figure 17*.

Never raise V_{PP}/Write Protect to V_{PP} from any mode except Read mode, otherwise the memory may be left in an indeterminate state.

The V_{PP}/Write Protect pin must not be left floating or unconnected or the device may become unreliable. A 0.1µF capacitor should be connected between the V_{PP}/Write Protect pin and the V_{SS} Ground pin to decouple the current surges from the power supply. The PCB track widths must be sufficient to carry the currents required during Unlock Bypass Program, I_{PP}.

Table 3. Hardware protection

$V_{PP}/\overline{WP}$	RP	Function
V_{IL}	V_{IH}	4 outermost parameter blocks protected from Program/Erase operations
	V_{ID}	All blocks temporarily unprotected except the 4 outermost blocks
V_{IH} or V_{ID}	V_{ID}	All blocks temporarily unprotected
V_{PPH}	V_{IH} or V_{ID}	All blocks temporarily unprotected

2.9 Reset/Block Temporary Unprotect ($\overline{RP}$)

The Reset/Block Temporary Unprotect pin can be used to apply a Hardware Reset to the memory or to temporarily unprotect all Blocks that have been protected.

Note that if $V_{PP}/\overline{WP}$ is at V_{IL}, then the four outermost boot blocks will remain protected even if RP is at V_{ID}.

A Hardware Reset is achieved by holding Reset/Block Temporary Unprotect Low, V_{IL}, for at least t_{PLPX}. After Reset/Block Temporary Unprotect goes High, V_{IH}, the memory will be ready for Bus Read and Bus Write operations after t_{PHEL} or t_{RHEL}, whichever occurs last. See the Ready/Busy Output section, *Table 20* and *Figure 16: Reset/Block Temporary Unprotect AC waveforms*.

Holding $\overline{RP}$ at V_{ID} will temporarily unprotect the protected Blocks in the memory. Program and Erase operations on all blocks will be possible. The transition from V_{IH} to V_{ID} must be slower than t_{PHPHH}.

2.10 Ready/Busy Output (R$\overline{B}$)

The Ready/Busy pin is an open-drain output that can be used to identify when the device is performing a Program or Erase operation. During Program or Erase operations Ready/Busy is Low, V_{OL}. Ready/Busy is high-impedance during Read mode, Auto Select mode and Erase Suspend mode.

After a Hardware Reset, Bus Read and Bus Write operations cannot begin until Ready/Busy becomes high-impedance. See *Table 20* and *Figure 16: Reset/Block Temporary Unprotect AC waveforms*.

The use of an open-drain output allows the Ready/Busy pins from several memories to be connected to a single pull-up resistor. A Low will then indicate that one, or more, of the memories is busy.

2.11 Byte/Word Organization Select ($\overline{BYTE}$)

The Byte/Word Organization Select pin is used to switch between the x8 and x16 Bus modes of the memory. When Byte/Word Organization Select is Low, V_{IL}, the memory is in x8 mode, when it is High, V_{IH}, the memory is in x16 mode.

2.12 V_{CC} Supply Voltage

V_{CC} provides the power supply for all operations (Read, Program and Erase).

The Command Interface is disabled when the V_{CC} Supply Voltage is less than the Lockout voltage, V_{LKO}. This prevents Bus Write operations from accidentally damaging the data during power up, power down and power surges. If the Program/Erase Controller is programming or erasing during this time then the operation aborts and the memory contents being altered will be invalid.

A 0.1µF capacitor should be connected between the V_{CC} Supply Voltage pin and the V_{SS} Ground pin to decouple the current surges from the power supply. The PCB track widths must be sufficient to carry the currents required during Program and Erase operations, I_{CC3}.

2.13 V_{SS} Ground

V_{SS} is the reference for all voltage measurements. The device features two V_{SS} pins both of which must be connected to the system ground.

3 Bus operations

There are five standard bus operations that control the device. These are Bus Read (Random and Page modes), Bus Write, Output Disable, Standby and Automatic Standby.

Using the multiple bank architecture of the M29DW640F, while programming or erasing is underway in one group of banks (from 1 to 3), reading can be conducted in any of the other banks. Write operations are only allowed in one bank at a time.

See *Table 4* and *Table 5*, Bus operations, for a summary. Typically glitches of less than 5ns on Chip Enable, Write Enable, and Reset pins are ignored by the memory and do not affect bus operations.

3.1 Bus Read

Bus Read operations read from the memory cells, or specific registers in the Command Interface. To speed up the read operation the memory array can be read in Page mode where data is internally read and stored in a page buffer. The Page has a size of 8 Words and is addressed by the address inputs A0-A2.

A valid Bus Read operation involves setting the desired address on the Address Inputs, applying a Low signal, V_{IL}, to Chip Enable and Output Enable and keeping Write Enable High, V_{IH}. The Data Inputs/Outputs will output the value, see *Figure 10: Random Read AC waveforms*, *Figure 11: Page Read AC waveforms*, and *Table 16: Read AC characteristics*, for details of when the output becomes valid.

3.2 Bus Write

Bus Write operations write to the Command Interface. A valid Bus Write operation begins by setting the desired address on the Address Inputs. The Address Inputs are latched by the Command Interface on the falling edge of Chip Enable or Write Enable, whichever occurs last. The Data Inputs/Outputs are latched by the Command Interface on the rising edge of Chip Enable or Write Enable, whichever occurs first. Output Enable must remain High, V_{IH}, during the whole Bus Write operation. See *Figure 12* and *Figure 13*, Write AC waveforms, and *Table 17* and *Table 18*, Write AC characteristics, for details of the timing requirements.

3.3 Output Disable

The Data Inputs/Outputs are in the high impedance state when Output Enable is High, V_{IH}.

3.4 Standby

When Chip Enable is High, V_{IH}, the memory enters Standby mode and the Data Inputs/Outputs pins are placed in the high-impedance state. To reduce the Supply Current to the Standby Supply Current, I_{CC2}, Chip Enable should be held within V_{CC} ± 0.2V. For the Standby current level see *Table 15: DC characteristics*.

During program or erase operations the memory will continue to use the Program/Erase Supply Current, I_{CC3}, for Program or Erase operations until the operation completes.

3.5 Automatic Standby

If CMOS levels (V_{CC} ± 0.2V) are used to drive the bus and the bus is inactive for 300ns or more the memory enters Automatic Standby where the internal Supply Current is reduced to the Standby Supply Current, I_{CC2}. The Data Inputs/Outputs will still output data if a Bus Read operation is in progress.

3.6 Special bus operations

Additional bus operations can be performed to read the Electronic Signature and also to apply and remove Block Protection. These bus operations are intended for use by programming equipment and are not usually used in applications. They require V_{ID} to be applied to some pins.

3.6.1 Electronic Signature

The memory has two codes, the manufacturer code and the device code, that can be read to identify the memory. These codes can be read by applying the signals listed in *Table 4* and *Table 5*, Bus operations.

3.6.2 Block Protect and Chip Unprotect

Groups of blocks can be protected against accidental Program or Erase. The Protection Groups are shown in *Appendix A*, *Table 24: Block addresses* The whole chip can be unprotected to allow the data inside the blocks to be changed.

The V_{PP}/Write Protect pin can be used to protect the four outermost boot blocks. When V_{PP}/Write Protect is at V_{IL} the four outermost boot blocks are protected and remain protected regardless of the Block Protection Status or the Reset/Block Temporary Unprotect pin status.

Block Protect and Chip Unprotect operations are described in *Appendix D*.

Table 4. Bus operations, $\overline{\text{BYTE}} = V_{IL}$[1]

Operation	$\overline{E}$	$\overline{G}$	$\overline{W}$	Address Inputs						Data Inputs/Outputs	
				A21-A12	A3	A2	A1	A0	Others, DQ15A-1	DQ14-DQ8	DQ7-DQ0
Bus Read	V_{IL}	V_{IL}	V_{IH}	Cell address						Hi-Z	Data Output
Bus Write	V_{IL}	V_{IH}	V_{IL}	Command address						Hi-Z	Data Input
Output Disable	X	V_{IH}	V_{IH}	X						Hi-Z	Hi-Z
Standby	V_{IH}	X	X	X						Hi-Z	Hi-Z
Read Manufacturer Code	V_{IL}	V_{IL}	V_{IH}	Bank addrs	V_{IL}	V_{IL}	V_{IL}	V_{IL}	A6 = V_{IL} A9 = V_{ID}, others =X	Hi-Z	20h
Read Device Code (Cycle 1)	V_{IL}	V_{IL}	V_{IH}		V_{IL}	V_{IL}	V_{IL}	V_{IH}		Hi-Z	7Eh
Read Device Code (Cycle 2)	V_{IL}	V_{IL}	V_{IH}		V_{IH}	V_{IH}	V_{IH}	V_{IL}		Hi-Z	02h
Read Device Code (Cycle 3)	V_{IL}	V_{IL}	V_{IH}		V_{IH}	V_{IH}	V_{IH}	V_{IH}		Hi-Z	01h
Extended Block Indicator Bit (DQ7)	V_{IL}	V_{IL}	V_{IH}	Bank A	V_{IL}	V_{IL}	V_{IH}	V_{IH}		Hi-Z	80h (factory locked) 00h (not locked)
Block Protection Verification	V_{IL}	V_{IL}	V_{IH}	Block addrs	V_{IL}	V_{IL}	V_{IH}	V_{IL}		Hi-Z	01h (protected) 00h (unprotected)

[1] X = V_{IL} or V_{IH}.

Table 5. Bus operations, $\overline{\text{BYTE}}$ = V_{IH} (1)

Operation	$\overline{E}$	$\overline{G}$	$\overline{W}$	Address Inputs						Data Inputs/Outputs
				A21-A12	A3	A2	A1	A0	Others	DQ15A-1, DQ14-DQ0
Bus Read	V_{IL}	V_{IL}	V_{IH}	Cell address						Data Output
Bus Write	V_{IL}	V_{IH}	V_{IL}	Command address						Data Input
Output Disable	X	V_{IH}	V_{IH}	X						Hi-Z
Standby	V_{IH}	X	X	X						Hi-Z
Read Manufacturer Code	V_{IL}	V_{IL}	V_{IH}	Bank addrs	V_{IL}	V_{IL}	V_{IL}	V_{IL}	A6 = V_{IL} A9 = V_{ID}, others =X	0020h
Read Device Code (Cycle 1)	V_{IL}	V_{IL}	V_{IH}		V_{IL}	V_{IL}	V_{IL}	V_{IH}		227Eh
Read Device Code (Cycle 2)	V_{IL}	V_{IL}	V_{IH}		V_{IH}	V_{IH}	V_{IH}	V_{IL}		2202h
Read Device Code (Cycle 3)	V_{IL}	V_{IL}	V_{IH}		V_{IH}	V_{IH}	V_{IH}	V_{IH}		2201h
Extended Block Indicator Bit (DQ7)	V_{IL}	V_{IL}	V_{IH}	Bank A	V_{IL}	V_{IL}	V_{IH}	V_{IH}		0080h (factory locked) 0000h (not locked)
Block Protection Verification	V_{IL}	V_{IL}	V_{IH}	Block addrs	V_{IL}	V_{IL}	V_{IH}	V_{IL}		0001h (protected) 0000h (unprotected)

1. X = V_{IL} or V_{IH}.

4 Command interface

All Bus Write operations to the memory are interpreted by the Command Interface. Commands consist of one or more sequential Bus Write operations. Failure to observe a valid sequence of Bus Write operations will result in the memory returning to Read mode. The long command sequences are imposed to maximize data security.

The address used for the commands changes depending on whether the memory is in 16-bit or 8-bit mode. See either *Table 6*, or *Table 7*, depending on the configuration that is being used, for a summary of the commands.

4.1 Standard commands

4.1.1 Read/Reset command

The Read/Reset command returns the memory to its Read mode. It also resets the errors in the Status Register. Either one or three Bus Write operations can be used to issue the Read/Reset command.

The Read/Reset command can be issued, between Bus Write cycles before the start of a program or erase operation, to return the device to read mode. If the Read/Reset command is issued during the timeout of a Block erase operation then the memory will take up to 10µs to abort. During the abort period no valid data can be read from the memory. The Read/Reset command will not abort an Erase operation when issued while in Erase Suspend.

4.1.2 Auto Select command

The Auto Select command is used to read the Manufacturer Code and Device Code, the Block Protection Status and the Extended Block Indicator. It can be addressed to either Bank. Three consecutive Bus Write operations are required to issue the Auto Select command. The final Write cycle must be addressed to one of the Banks. Once the Auto Select command is issued Bus Read operations to the Bank where the command was issued output the Auto Select data. Bus Read operations to the other Bank will output the contents of the memory array. The memory remains in Auto Select mode until a Read/Reset or CFI Query command is issued. This command must be issued addressing the same Bank, as was given when entering Auto Select Mode.

In Auto Select mode the Manufacturer Code can be read using a read operation, A6 and A3 to A0 each held at V_{IL}, and A21-A19 set to the Bank Address. The other address bits may be set to either V_{IL} or V_{IH}.

The Device Codes can be read using a read operation, A6 held at V_{IL}, A3 to A0 each held at the levels given in *Table 4* and *Table 5*, and A21-A19 set to the Bank Address. The other address bits may be set to either V_{IL} or V_{IH}.

The Block Protection Status of each block can be read using a read operation, A6 A3 A2 A0 each held at V_{IL}, A1 held at V_{IH}, and A21-A19 set to the Bank Address, and A18-A12 specifying the address of the block inside the Bank. The other address bits may be set to either V_{IL} or V_{IH}. If the addressed block is protected then 01h is output on Data Inputs/Outputs DQ0-DQ7, otherwise 00h is output.

The Extended Block Status of the Extended Block can be read using a read operation, A6, A3 and A2, at V_{IL}, A0 and A1, at V_{IH}, and A21-A19 set to Bank Address A. The other bits may be set to either V_{IL} or V_{IH} (Don't Care). If the Extended Block is "Factory Locked" then 80h is output on Data Input/Outputs DQ0-DQ7, otherwise 00h is output.

4.1.3 Read CFI Query command

The Read CFI Query Command is used to put the addressed bank in Read CFI Query mode. Once in Read CFI Query mode Bus Read operations to the same bank will output data from the Common Flash Interface (CFI) Memory Area. If the read operations are to a different bank from the one specified in the command then the read operations will output the contents of the memory array and not the CFI data.

One Bus Write cycle is required to issue the Read CFI Query Command. Care must be taken to issue the command to one of the banks (A21-A19) along with the address shown in *Table 4* and *Table 5* (A-1, A0-A10). Once the command is issued subsequent Bus Read operations in the same bank (A21-A19) to the addresses shown in *Appendix B* (A7-A0), will read from the Common Flash Interface Memory Area.

This command is valid only when the device is in the Read Array or Autoselected mode. To enter Read CFI query mode from Auto Select mode, the Read CFI Query command must be issued to the same bank address as the Auto Select command, otherwise the device will not enter Read CFI Query mode.

The Read/Reset command must be issued to return the device to the previous mode (the Read Array mode or Autoselected mode). A second Read/Reset command would be needed if the device is to be put in the Read Array mode from Autoselected mode.

See *Appendix B*, *Table 25*, *Table 26*, *Table 27*, *Table 28*, *Table 29* and *Table 30* for details on the information contained in the Common Flash Interface (CFI) memory area.

4.1.4 Chip Erase command

The Chip Erase command can be used to erase the entire chip. Six Bus Write operations are required to issue the Chip Erase Command and start the Program/Erase Controller.

If any blocks are protected then these are ignored and all the other blocks are erased. If all of the blocks are protected the Chip Erase operation appears to start but will terminate within about 100µs, leaving the data unchanged. No error condition is given when protected blocks are ignored.

During the erase operation the memory will ignore all commands, including the Erase Suspend command. It is not possible to issue any command to abort the operation. Typical chip erase times are given in *Table 8*. All Bus Read operations during the Chip Erase operation will output the Status Register on the Data Inputs/Outputs. See the section on the Status Register for more details.

After the Chip Erase operation has completed the memory will return to the Read Mode, unless an error has occurred. When an error occurs the memory will continue to output the Status Register. A Read/Reset command must be issued to reset the error condition and return to Read Mode.

The Chip Erase Command sets all of the bits in unprotected blocks of the memory to ’1’. All previous data is lost.

4.1.5 Block Erase command

The Block Erase command can be used to erase a list of one or more blocks in one or more Banks. It sets all of the bits in the unprotected selected blocks to ’1’. All previous data in the selected blocks is lost.

Six Bus Write operations are required to select the first block in the list. Each additional block in the list can be selected by repeating the sixth Bus Write operation using the address of the additional block. The Block Erase operation starts the Program/Erase Controller after a time-out period of 50µs after the last Bus Write operation. Once the Program/Erase Controller starts it is not possible to select any more blocks. Each additional block must therefore be selected within 50µs of the last block. The 50µs timer restarts when an additional block is selected. After the sixth Bus Write operation a Bus Read operation within the same Bank will output the Status Register. See the Status Register section for details on how to identify if the Program/Erase Controller has started the Block Erase operation.

If any selected blocks are protected then these are ignored and all the other selected blocks are erased. If all of the selected blocks are protected the Block Erase operation appears to start but will terminate within about 100µs, leaving the data unchanged. No error condition is given when protected blocks are ignored.

During the Block Erase operation the memory will ignore all commands except the Erase Suspend command and the Read/Reset command which is only accepted during the 50µs time-out period. Typical block erase times are given in *Table 8*.

After the Erase operation has started all Bus Read operations to the Banks being erased will output the Status Register on the Data Inputs/Outputs. See the section on the Status Register for more details.

After the Block Erase operation has completed the memory will return to the Read Mode, unless an error has occurred. When an error occurs Bus Read operations to the Banks where the command was issued will continue to output the Status Register. A Read/Reset command must be issued to reset the error condition and return to Read mode.

4.1.6 Erase Suspend command

The Erase Suspend command may be used to temporarily suspend a Block or multiple Block Erase operation. One Bus Write operation specifying the Bank Address of one of the Blocks being erased is required to issue the command. Issuing the Erase Suspend command returns the whole device to Read mode.

The Program/Erase Controller will suspend within the Erase Suspend Latency time (see *Table 8* for value) of the Erase Suspend Command being issued. Once the Program/Erase Controller has stopped the memory will be set to Read mode and the Erase will be suspended. If the Erase Suspend command is issued during the period when the memory is waiting for an additional block (before the Program/Erase Controller starts) then the Erase is suspended immediately and will start immediately when the Erase Resume Command is issued. It is not possible to select any further blocks to erase after the Erase Resume.

During Erase Suspend it is possible to Read and Program cells in blocks that are not being erased; both Read and Program operations behave as normal on these blocks. If any attempt is made to program in a protected block or in the suspended block then the Program command is ignored and the data remains unchanged. The Status Register is not read and no error condition is given. Reading from blocks that are being erased will output the Status Register.

It is also possible to issue the Auto Select, Read CFI Query and Unlock Bypass commands during an Erase Suspend. The Read/Reset command must be issued to return the device to Read Array mode before the Resume command will be accepted.

During Erase Suspend a Bus Read operation to the Extended Block will output the Extended Block data. Once in the Extended Block mode, the Exit Extended Block command must be issued before the erase operation can be resumed.

4.1.7 Erase Resume command

The Erase Resume command is used to restart the Program/Erase Controller after an Erase Suspend. The command must include the Bank Address of the Erase-Suspended Bank, otherwise the Program/Erase Controller is not restarted.

The device must be in Read Array mode before the Resume command will be accepted. An Erase can be suspended and resumed more than once.

4.1.8 Program Suspend command

The Program Suspend command allows the system to interrupt a program operation so that data can be read from any block. When the Program Suspend command is issued during a program operation, the device suspends the program operation within the Program Suspend Latency time (see *Table 8* for value) and updates the Status Register bits. The Bank Addresses of the Block being programmed must be specified in the Program Suspend command.

After the program operation has been suspended, the system can read array data from any address. However, data read from Program-Suspended addresses is not valid.

The Program Suspend command may also be issued during a program operation while an erase is suspended. In this case, data may be read from any addresses not in Erase Suspend or Program Suspend. If a read is needed from the Extended Block area (One-time Program area), the user must use the proper command sequences to enter and exit this region.

The system may also issue the Auto Select command sequence when the device is in the Program Suspend mode. The system can read as many Auto Select codes as required. When the device exits the Auto Select mode, the device reverts to the Program Suspend mode, and is ready for another valid operation. See Auto Select command sequence for more information.

4.1.9 Program Resume command

After the Program Resume command is issued, the device reverts to programming. The controller can determine the status of the program operation using the DQ7 or DQ6 status bits, just as in the standard program operation. See Write Operation Status for more information.

The system must write the Program Resume command, specifying the Bank addresses of the Program-Suspended Block, to exit the Program Suspend mode and to continue the programming operation.

Further issuing of the Resume command is ignored. Another Program Suspend command can be written after the device has resumed programming.

4.1.10 Program command

The Program command can be used to program a value to one address in the memory array at a time. The command requires four Bus Write operations, the final Write operation latches the address and data in the internal state machine and starts the Program/Erase Controller.

Programming can be suspended and then resumed by issuing a Program Suspend command and a Program Resume command, respectively (see *Section 4.1.8: Program Suspend command* and *Section 4.1.9: Program Resume command*).

If the address falls in a protected block then the Program command is ignored, the data remains unchanged. The Status Register is never read and no error condition is given.

After programming has started, Bus Read operations in the Bank being programmed output the Status Register content, while Bus Read operations to the other Bank output the contents of the memory array. See the section on the Status Register for more details. Typical program times are given in *Table 8*.

After the program operation has completed the memory will return to the Read mode, unless an error has occurred. When an error occurs Bus Read operations to the Bank where the command was issued will continue to output the Status Register. A Read/Reset command must be issued to reset the error condition and return to Read mode.

Note that the Program command cannot change a bit set at ’0’ back to ’1’. One of the Erase Commands must be used to set all the bits in a block or in the whole memory from ’0’ to ’1’.

4.2 Fast Program commands

There are five Fast Program commands available to improve the programming throughput, by writing several adjacent Words or Bytes in parallel.

When V_{PPH} is applied to the V_{PP}/Write Protect pin the memory automatically enters the Fast Program mode. The user can then choose to issue any of the Fast Program commands. Care must be taken because applying a V_{PPH} to the $V_{PP}/\overline{WP}$ pin will temporarily unprotect any protected block. Fast programming should not be attempted when V_{PP} is not at V_{PPH}.

4.2.1 Double Word Program command

This is used to write two adjacent Words in x16 mode, in parallel. The addresses of the two Words must differ only in A0.

Three bus write cycles are necessary to issue the command.

1. The first bus cycle sets up the command.
2. The second bus cycle latches the Address and the Data of the first Word to be written.
3. The third bus cycle latches the Address and the Data of the second Word to be written and starts the Program/Erase Controller.

4.2.2 Quadruple Word Program command

This is used to write a page of four adjacent Words, in x16 mode, in parallel. The addresses of the four Words must differ only in A1 and A0.

Five bus write cycles are necessary to issue the command.

1. The first bus cycle sets up the command.
2. The second bus cycle latches the Address and the Data of the first Word to be written.
3. The third bus cycle latches the Address and the Data of the second Word to be written.
4. The fourth bus cycle latches the Address and the Data of the third Word to be written.
5. The fifth bus cycle latches the Address and the Data of the fourth Word to be written and starts the Program/Erase Controller.

4.2.3 Double Byte Program command

This is used to write two adjacent Bytes in x8 mode, in parallel. The addresses of the two Bytes must differ only in DQ15A-1.

Three bus write cycles are necessary to issue the command.

1. The first bus cycle sets up the command.
2. The second bus cycle latches the Address and the Data of the first Byte to be written.
3. The third bus cycle latches the Address and the Data of the second Byte to be written and starts the Program/Erase Controller.

4.2.4 Quadruple Byte Program command

This is used to write four adjacent Bytes in x8 mode, in parallel. The addresses of the four Bytes must differ only in A0, DQ15A-1.

Five bus write cycles are necessary to issue the command.

1. The first bus cycle sets up the command.
2. The second bus cycle latches the Address and the Data of the first Byte to be written.
3. The third bus cycle latches the Address and the Data of the second Byte to be written.
4. The fourth bus cycle latches the Address and the Data of the third Byte to be written.
5. The fifth bus cycle latches the Address and the Data of the fourth Byte to be written and starts the Program/Erase Controller.

4.2.5 Octuple Byte Program command

This is used to write eight adjacent Bytes, in x8 mode, in parallel. The addresses of the eight Bytes must differ only in A1, A0 and DQ15A-1.

Nine bus write cycles are necessary to issue the command.

1. The first bus cycle sets up the command.
2. The second bus cycle latches the Address and the Data of the first Byte to be written.
3. The third bus cycle latches the Address and the Data of the second Byte to be written.
4. The fourth bus cycle latches the Address and the Data of the third Byte to be written.
5. The fifth bus cycle latches the Address and the Data of the fourth Byte to be written.
6. The sixth bus cycle latches the Address and the Data of the fifth Byte to be written.
7. The seventh bus cycle latches the Address and the Data of the sixth Byte to be written.
8. The eighth bus cycle latches the Address and the Data of the seventh Byte to be written.
9. The ninth bus cycle latches the Address and the Data of the eighth Byte to be written and starts the Program/Erase Controller.

Only one bank can be programmed at any one time. The other bank must be in Read mode or Erase Suspend.

After programming has started, Bus Read operations in the Bank being programmed output the Status Register content, while Bus Read operations to the other Bank output the contents of the memory array.

Programming can be suspended and then resumed by issuing a Program Suspend command and a Program Resume command, respectively. (See *Section 4.1.8: Program Suspend command* and *Section 4.1.9: Program Resume command*)

After the program operation has completed the memory will return to the Read mode, unless an error has occurred. When an error occurs Bus Read operations to the Bank where the command was issued will continue to output the Status Register. A Read/Reset command must be issued to reset the error condition and return to Read mode.

Note that the Fast Program commands cannot change a bit set at ’0’ back to ’1’. One of the Erase Commands must be used to set all the bits in a block or in the whole memory from ’0’ to ’1’.

Typical Program times are given in *Table 8: Program, Erase times and Program, Erase Endurance cycles*.

4.2.6 Unlock Bypass command

The Unlock Bypass command is used in conjunction with the Unlock Bypass Program command to program the memory faster than with the standard program commands. When the cycle time to the device is long, considerable time saving can be made by using these commands. Three Bus Write operations are required to issue the Unlock Bypass command.

Once the Unlock Bypass command has been issued the bank enters Unlock Bypass mode. The Unlock Bypass Program command can then be issued to program addresses within the bank, or the Unlock Bypass Reset command can be issued to return the bank to Read mode. In Unlock Bypass mode the memory can be read as if in Read mode.

When V_{PP} is applied to the V_{PP}/Write Protect pin the memory automatically enters the Unlock Bypass mode and the Unlock Bypass Program command can be issued immediately.

4.2.7 Unlock Bypass Program command

The Unlock Bypass Program command can be used to program one address in the memory array at a time. The command requires two Bus Write operations, the final write operation latches the address and data in the internal state machine and starts the Program/Erase Controller.

The Program operation using the Unlock Bypass Program command behaves identically to the Program operation using the Program command. The operation cannot be aborted, a Bus Read operation to the Bank where the command was issued outputs the Status Register. See the Program command for details on the behavior.

4.2.8 Unlock Bypass Reset command

The Unlock Bypass Reset command can be used to return to Read/Reset mode from Unlock Bypass Mode. Two Bus Write operations are required to issue the Unlock Bypass Reset command. Read/Reset command does not exit from Unlock Bypass Mode.

4.3 Block Protection commands

4.3.1 Enter Extended Block command

The M29DW640F has one extra 256-Byte block (Extended Block) that can only be accessed using the Enter Extended Block command. Three Bus write cycles are required to issue the Extended Block command. Once the command has been issued the device enters Extended Block mode where all Bus Read or Program operations to the 000000h-00007Fh (Word) or 000000h-0000FFh (Byte) addresses access the Extended Block. The Extended Block cannot be erased, and can be treated as one-time programmable (OTP) memory. In Extended Block mode only array cell locations (Bank A) with the same addresses as the Extended Block (000000h-00007Fh (Word) or 000000h-0000FFh (Byte)) are not accessible. In Extended Block mode dual operations are allowed and the Extended Block physically belongs to Bank A.

When in Extended Block mode, Erase, Chip Erase, Erase Suspend and Erase resume commands are not allowed.

To exit from the Extended Block mode the Exit Extended Block command must be issued.

The Extended Block can be protected, however once protected the protection cannot be undone.

4.3.2 Exit Extended Block command

The Exit Extended Block command is used to exit from the Extended Block mode and return the device to Read mode. Four Bus Write operations are required to issue the command.

4.3.3 Block Protect and Chip Unprotect commands

Groups of blocks can be protected against accidental Program or Erase. The Protection Groups are shown in *Appendix A*, *Table 24: Block addresses*. The whole chip can be unprotected to allow the data inside the blocks to be changed.

Block Protect and Chip Unprotect operations are described in *Appendix D*.

Table 6. Commands, 16-bit mode, $\overline{\text{BYTE}}$ = V_{IH}

Command	Length	Bus Write operations[1]											
		1st		2nd		3rd		4th		5th		6th	
		Add	Data	Add	Data	Add	Data	Add	Data	Add	Data	Add	Data
Read/Reset	1	X	F0										
	3	555	AA	2AA	55	X	F0						
Auto Select	3	555	AA	2AA	55	(BKA) 555	90						
Program	4	555	AA	2AA	55	555	A0	PA	PD				
Double Word Program	3	555	50	PA0	PD0	PA1	PD1						
Quadruple Word Program	3	555	56	PA0	PD0	PA1	PD1	PA2	PD2	PA3	PD3		
Unlock Bypass	3	555	AA	2AA	55	555	20						
Unlock Bypass Program	2	X	A0	PA	PD								
Unlock Bypass Reset	2	X	90	X	00								
Chip Erase	6	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10
Block Erase	6+	555	AA	2AA	55	555	80	555	AA	2AA	55	BA	30
Erase/Program Suspend	1	BKA	B0										
Erase/Program Resume	1	BKA	30										
Read CFI Query[2]	1	(BKA) 55	98										
Enter Extended Block	3	555	AA	2AA	55	555	88						
Exit Extended Block	4	555	AA	2AA	55	555	90	X	00				

[1] X Don't Care, PA Program Address, PD Program Data, BA Any address in the Block, BKA Bank Address. All values in the table are in hexadecimal.

[2] Normally the Command Interface only uses A–1, A0-A10 and DQ0-DQ7 to verify the commands and A11-A21 are Don't Care, however for the Read CFI command A21-A14 must specify a bank address, and the subsequent read operations must be addressed to the same bank.

Table 7. Commands, 8-bit mode, $\overline{\text{BYTE}} = V_{IL}$

Command	Length	Bus Write operations[1]																	
		1st		2nd		3rd		4th		5th		6th		7th		8th		9th	
		Add	Data	Add	Data	Add	Data	Add	Data	Add	Data	Add	Data	Add	Data	Add	Data	Add	Data
Read/Reset	1	X	F0																
	3	AAA	AA	555	55	X	F0												
Auto Select	3	AAA	AA	555	55	(BKA) AAA	90												
Program	4	AAA	AA	555	55	AAA	A0	PA	PD										
Double Byte Program	3	AAA	50	PA0	PD1	PA1	PD1												
Quadruple Byte Program	5	AAA	56	PA0	PD0	PA1	PD1	PA2	PD2	PA3	PD3								
Octuple Byte Program	5	AAA	8B	PA0	PD0	PA1	PD1	PA2	PD2	PA3	PD3	PA4	PD4	PA5	PD5	PA6	PD6	PA7	PD7
Unlock Bypass	3	AAA	AA	555	55	AAA	20												
Unlock Bypass Program	2	X	A0	PA	PD														
Unlock Bypass Reset	2	X	90	X	00														
Chip Erase	6	AAA	AA	555	55	AAA	80	AAA	AA	555	55	AAA	10						
Block Erase	6+	AAA	AA	555	55	AAA	80	AAA	AA	555	55	BA	30						
Erase/ Program Suspend	1	BKA	B0																
Erase/ Program Resume	1	BKA	30																
Read CFI Query[2]	1	(BKA) AA	98																
Enter Extended Block	3	AAA	AA	555	55	AAA	88												
Exit Extended Block	4	AAA	AA	555	55	AAA	90	X	00										

1. X Don't Care, PA Program Address, PD Program Data, BA Any address in the Block. All values in the table are in hexadecimal.
2. Normally the Command Interface only uses A–1, A0-A10 and DQ0-DQ7 to verify the commands and A11-A21 are Don't Care, however for the Read CFI command A21-A14 must specify a bank address, and the subsequent read operations must be addressed to the same bank.

Table 8. Program, Erase times and Program, Erase Endurance cycles

Parameter	Min	Typ(1)(2)	Max(2)	Unit
Chip Erase		80	400(3)	s
Block Erase (64 KBytes)		0.8	6(4)	s
Erase Suspend latency time			50(4)	µs
Byte Program (1, 2, 4 or 8 at-a-time)		10	200(3)	µs
Word Program (1, 2 or 4 at-a-time)		10	200(3)	µs
Chip Program (Byte by Byte)		80	400(3)	s
Chip Program (Word by Word)		40	200(3)	s
Chip Program (quadruple Byte or double Word)		20	100(3)	s
Chip Program (octuple Byte or quadruple Word)		10	50(3)	s
Program Suspend latency time			4	µs
Program/Erase cycles (per Block)	100,000			cycles
Data Retention	20			years

1. Typical values measured at room temperature and nominal voltages.
2. Sampled, but not 100% tested.
3. Maximum value measured at worst case conditions for both temperature and V_{CC} after 100,00 program/erase cycles.
4. Maximum value measured at worst case conditions for both temperature and V_{CC}.

5 Status register

The M29DW640F has one Status Register. The Status Register provides information on the current or previous Program or Erase operations executed in each bank. The various bits convey information and errors on the operation. Bus Read operations from any address within the Bank, always read the Status Register during Program and Erase operations. It is also read during Erase Suspend when an address within a block being erased is accessed.

The bits in the Status Register are summarized in *Table 9: Status Register Bits*.

5.1 Data Polling Bit (DQ7)

The Data Polling Bit can be used to identify whether the Program/Erase Controller has successfully completed its operation or if it has responded to an Erase Suspend. The Data Polling Bit is output on DQ7 when the Status Register is read.

During Program operations the Data Polling Bit outputs the complement of the bit being programmed to DQ7. After successful completion of the Program operation the memory returns to Read mode and Bus Read operations from the address just programmed output DQ7, not its complement.

During Erase operations the Data Polling Bit outputs ’0’, the complement of the erased state of DQ7. After successful completion of the Erase operation the memory returns to Read Mode.

In Erase Suspend mode the Data Polling Bit will output a ’1’ during a Bus Read operation within a block being erased. The Data Polling Bit will change from a ’0’ to a ’1’ when the Program/Erase Controller has suspended the Erase operation.

Figure 6: Data Polling flowchart, gives an example of how to use the Data Polling Bit. A Valid Address is the address being programmed or an address within the block being erased.

5.1.1 Toggle Bit (DQ6)

The Toggle Bit can be used to identify whether the Program/Erase Controller has successfully completed its operation or if it has responded to an Erase Suspend. The Toggle Bit is output on DQ6 when the Status Register is read.

During Program and Erase operations the Toggle Bit changes from ’0’ to ’1’ to ’0’, etc., with successive Bus Read operations at any address. After successful completion of the operation the memory returns to Read mode.

During Erase Suspend mode the Toggle Bit will output when addressing a cell within a block being erased. The Toggle Bit will stop toggling when the Program/Erase Controller has suspended the Erase operation.

Figure 7: Toggle flowchart, gives an example of how to use the Data Toggle Bit. *Figure 14* and *Figure 15* describe Toggle Bit timing waveform.

5.1.2 Error Bit (DQ5)

The Error Bit can be used to identify errors detected by the Program/Erase Controller. The Error Bit is set to ’1’ when a Program, Block Erase or Chip Erase operation fails to write the correct data to the memory. If the Error Bit is set a Read/Reset command must be issued before other commands are issued. The Error bit is output on DQ5 when the Status Register is read.

Note that the Program command cannot change a bit set to ’0’ back to ’1’ and attempting to do so will set DQ5 to ‘1’. A Bus Read operation to that address will show the bit is still ‘0’. One of the Erase commands must be used to set all the bits in a block or in the whole memory from ’0’ to ’1’.

5.1.3 Erase Timer Bit (DQ3)

The Erase Timer Bit can be used to identify the start of Program/Erase Controller operation during a Block Erase command. Once the Program/Erase Controller starts erasing the Erase Timer Bit is set to ’1’. Before the Program/Erase Controller starts the Erase Timer Bit is set to ’0’ and additional blocks to be erased may be written to the Command Interface. The Erase Timer Bit is output on DQ3 when the Status Register is read.

5.1.4 Alternative Toggle Bit (DQ2)

The Alternative Toggle Bit can be used to monitor the Program/Erase controller during Erase operations. The Alternative Toggle Bit is output on DQ2 when the Status Register is read.

During Chip Erase and Block Erase operations the Toggle Bit changes from ’0’ to ’1’ to ’0’, etc., with successive Bus Read operations from addresses within the blocks being erased. A protected block is treated the same as a block not being erased. Once the operation completes the memory returns to Read mode.

During Erase Suspend the Alternative Toggle Bit changes from ’0’ to ’1’ to ’0’, etc. with successive Bus Read operations from addresses within the blocks being erased. Bus Read operations to addresses within blocks not being erased will output the memory cell data as if in Read mode.

After an Erase operation that causes the Error Bit to be set the Alternative Toggle Bit can be used to identify which block or blocks have caused the error. The Alternative Toggle Bit changes from ’0’ to ’1’ to ’0’, etc. with successive Bus Read Operations from addresses within blocks that have not erased correctly. The Alternative Toggle Bit does not change if the addressed block has erased correctly.

Figure 14 and *Figure 15* describe Alternative Toggle Bit timing waveform.

Table 9. Status Register Bits

Operation	Address	DQ7	DQ6	DQ5	DQ3	DQ2	R$\overline{B}$
Program	Bank address	$\overline{DQ7}$	Toggle	0	–	–	0
Program During Erase Suspend	Bank address	$\overline{DQ7}$	Toggle	0	–	–	0
Program Error	Bank address	$\overline{DQ7}$	Toggle	1	–	–	Hi-Z
Chip Erase	Any address	0	Toggle	0	1	Toggle	Hi-Z
Block Erase before timeout	Erasing block	0	Toggle	0	0	Toggle	0
	Non-Erasing block	0	Toggle	0	0	No Toggle	0
Block Erase	Erasing block	0	Toggle	0	1	Toggle	Hi-Z
	Non-Erasing block	0	Toggle	0	1	No Toggle	0
Erase Suspend	Erasing block	1	No Toggle	0	–	Toggle	Hi-Z
	Non-Erasing block	Data read as normal					Hi-Z
Erase Error	Good Block address	0	Toggle	1	1	No Toggle	0
	Faulty Block address	0	Toggle	1	1	Toggle	0

1. Unspecified data bits should be ignored.
1. *Figure 14* and *Figure 15* describe Toggle and Alternative Toggle Bits timing waveforms.

Figure 6. Data Polling flowchart

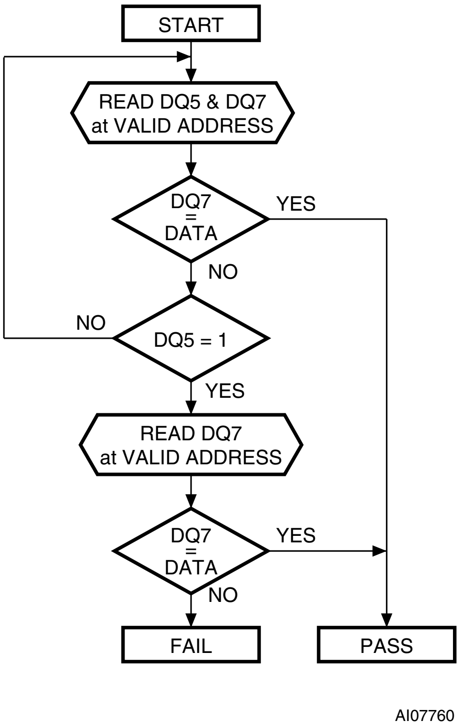
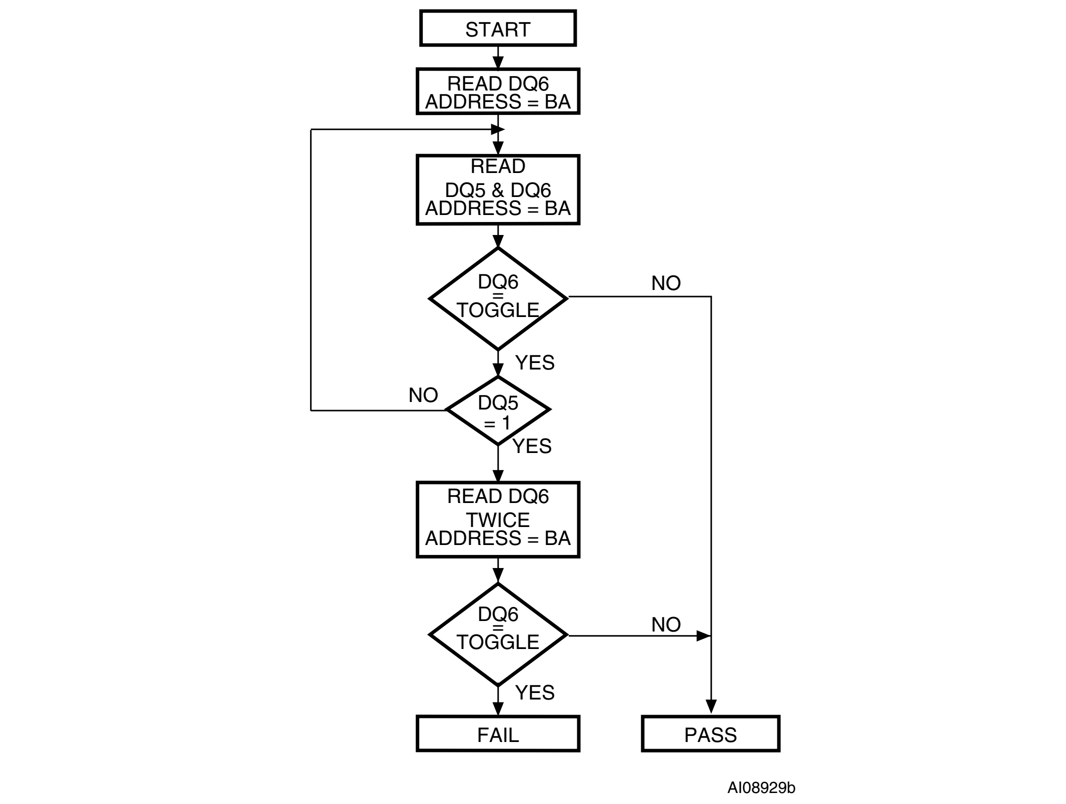


Figure 7. Toggle flowchart



1. BA = Address of Bank being Programmed or Erased.

6 Dual operations and multiple bank architecture

The Multiple Bank Architecture of the M29DW640F gives greater flexibility for software developers to split the code and data spaces within the memory array. The Dual Operations feature simplifies the software management of the device by allowing code to be executed from one bank while another bank is being programmed or erased.

The Dual Operations feature means that while programming or erasing in one bank, read operations are possible in another bank with zero latency.

Only one bank at a time is allowed to be in program or erase mode. However, certain commands can cross bank boundaries, which means that during an operation only the banks that are not concerned with the cross bank operation are available for dual operations. For example, if a Block Erase command is issued to erase blocks in both Bank A and Bank B, then only Banks C or D are available for read operations while the erase is being executed.

If a read operation is required in a bank, which is programming or erasing, the program or erase operation can be suspended.

Also if the suspended operation was erase then a program command can be issued to another block, so the device can have one block in Erase Suspend mode, one programming and other banks in read mode.

By using a combination of these features, read operations are possible at any moment in the device.

Table 10 and *Table 11* show the dual operations possible in other banks and in the same bank. Note that only the commonly used commands are represented in these tables.

Table 10. Dual operations allowed in other banks

Status of bank[1]	Commands allowed in another bank[1]							
	Read Array	Read Status Register[2]	Read CFI Query	Auto Select	Program	Erase	Program/ Erase Suspend	Program/ Erase Resume
Idle	Yes	Yes[3]	Yes	Yes	Yes	Yes	Yes[3]	Yes[4]
Programming	Yes	No	No	No	–	–	No	No
Erasing	Yes	No	No	No	–	–	No	No
Program Suspended	Yes	No	Yes	Yes	No	No	–	Yes[5]
Erase Suspended	Yes	No	Yes	Yes	Yes	No	–	Yes[6]

1. If several banks are involved in a program or erase operation, then only the banks that are not concerned with the operation are available for dual operations.
2. Read Status Register is not a command. The Status Register can be read during a block program or erase operation.
3. Only after a program or erase operation in that bank.
4. Only after a Program or Erase Suspend command in that bank.
5. Only a Program Resume is allowed if the bank was previously in Program Suspend mode.
6. Only an Erase Resume is allowed if the bank was previously in Erase Suspend mode.

Table 11. Dual operations allowed in same bank

Status of bank	Commands allowed in same bank							
	Read Array	Read Status Register[1]	Read CFI Query	Auto Select	Program	Erase	Program/ Erase Suspend	Program/ Erase Resume
Idle	Yes	Yes	Yes	Yes	Yes	Yes	Yes[2]	Yes[3]
Programming	No	Yes	No	No	–	–	Yes[4]	–
Erasing	No	Yes	No	No	–	No	Yes[5]	–
Program Suspended	Yes[6]	No	Yes	Yes	No	–	–	Yes
Erase Suspended	Yes[6]	Yes[7]	Yes	Yes	Yes[6]	No	–	Yes

1. Read Status Register is not a command. The Status Register can be read during a block program or erase operation.
2. Only after a program or erase operation in that bank.
3. Only after a Program or Erase Suspend command in that bank.
4. Only a Program Suspend.
5. Only an Erase suspend.
6. Not allowed in the Block or Word that is being erased or programmed.
7. The Status Register can be read by addressing the block being erase suspended.

7 Maximum ratings

Stressing the device above the rating listed in the Absolute Maximum Ratings table may cause permanent damage to the device. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability. These are stress ratings only and operation of the device at these or any other conditions above those indicated in the Operating sections of this specification is not implied. Refer also to the Numonyx SURE Program and other relevant quality documents.

Table 12. Absolute maximum ratings

Symbol	Parameter	Min	Max	Unit
T_{BIAS}	Temperature Under Bias	–50	125	°C
T_{STG}	Storage Temperature	–65	150	°C
V_{IO}	Input or Output voltage [1][2]	–0.6	V_{CC} +0.6	V
V_{CC}	Supply voltage	–0.6	4	V
V_{ID}	Identification voltage	–0.6	13.5	V
V_{PP}[3]	Program voltage	–0.6	13.5	V

1. Minimum voltage may undershoot to –2V during transition and for less than 20ns during transitions.
2. Maximum voltage may overshoot to V_{CC} +2V during transition and for less than 20ns during transitions.
3. V_{PP} must not remain at 12V for more than a total of 80hrs.

8 DC and AC parameters

This section summarizes the operating measurement conditions, and the DC and AC characteristics of the device. The parameters in the DC and AC characteristics Tables that follow, are derived from tests performed under the Measurement Conditions summarized in *Table 13: Operating and AC measurement conditions*. Designers should check that the operating conditions in their circuit match the operating conditions when relying on the quoted parameters.

Table 13. Operating and AC measurement conditions

Parameter	M29DW640F				Unit
	60		70		
	Min	Max	Min	Max	
V_{CC} Supply voltage	2.7	3.6	2.7	3.6	V
Ambient Operating Temperature	–40	85	–40	85	°C
Load capacitance (C_L)	30		30		pF
Input Rise and Fall times		10		10	ns
Input pulse voltages	0 to V_{CC}		0 to V_{CC}		V
Input and Output Timing Ref. voltages	$V_{CC}/2$		$V_{CC}/2$		V

Figure 8. AC measurement I/O waveform

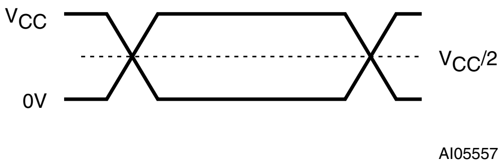
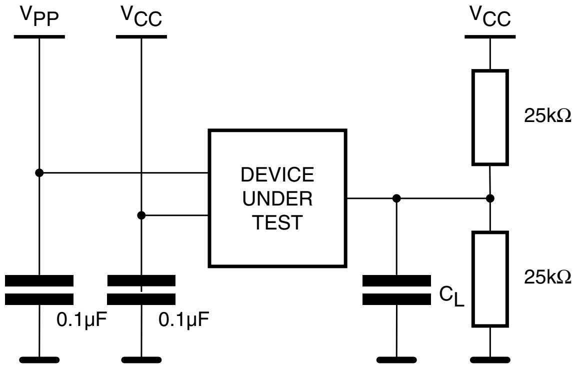


Figure 9. AC measurement Load Circuit



C_L includes JIG capacitance

AI05558

Table 14. Device capacitance(1)

Symbol	Parameter	Test condition	Min	Max	Unit
C_{IN}	Input capacitance	V_{IN} = 0V		6	pF
C_{OUT}	Output capacitance	V_{OUT} = 0V		12	pF

1. Sampled only, not 100% tested.

Table 15. DC characteristics

Symbol	Parameter	Test condition		Min	Max	Unit
I_{LI}	Input Leakage Current	0V $\leq V_{IN} \leq V_{CC}$			±1	µA
I_{LO}	Output Leakage Current	0V $\leq V_{OUT} \leq V_{CC}$			±1	µA
I_{CC1}(1)	Supply Current (Read)	$\overline{E} = V_{IL}$, $\overline{G} = V_{IH}$, f = 6MHz			10	mA
I_{CC2}	Supply Current (Standby)	$\overline{E} = V_{CC}$ ±0.2V, $\overline{RP} = V_{CC}$ ±0.2V			100	µA
I_{CC3} (1)(2)	Supply Current (Program/Erase)	Program/Erase Controller active	$V_{PP}/\overline{WP}$ = V_{IL} or V_{IH}		20	mA
			$V_{PP}/\overline{WP} = V_{PP}$		20	mA
V_{IL}	Input Low voltage			–0.5	0.8	V
V_{IH}	Input High voltage			0.7V_{CC}	V_{CC} +0.3	V
V_{PP}	Voltage for $V_{PP}/\overline{WP}$ Program Acceleration	V_{CC} = 2.7V ±10%		11.5	12.5	V
I_{PP}	Current for $V_{PP}/\overline{WP}$ Program Acceleration	V_{CC} =2.7V ±10%			15	mA
V_{OL}	Output Low voltage	I_{OL} = 1.8mA			0.45	V
V_{OH}	Output High voltage	I_{OH} = –100µA		V_{CC} –0.4		V
V_{ID}	Identification voltage			11.5	12.5	V
V_{LKO}	Program/Erase Lockout supply voltage			1.8	2.3	V

1. In Dual operations the Supply Current will be the sum of I_{CC1}(read) and I_{CC3} (program/erase).
2. Sampled only, not 100% tested.

Figure 10. Random Read AC waveforms

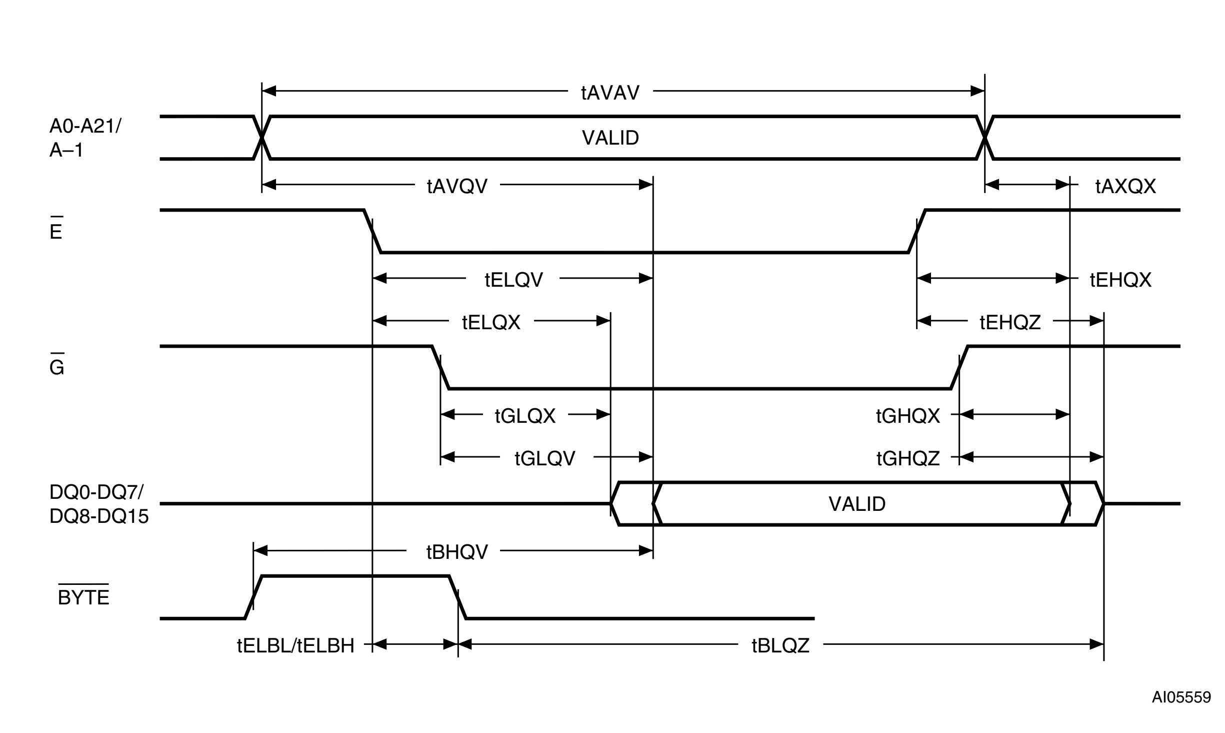


Figure 11. Page Read AC waveforms

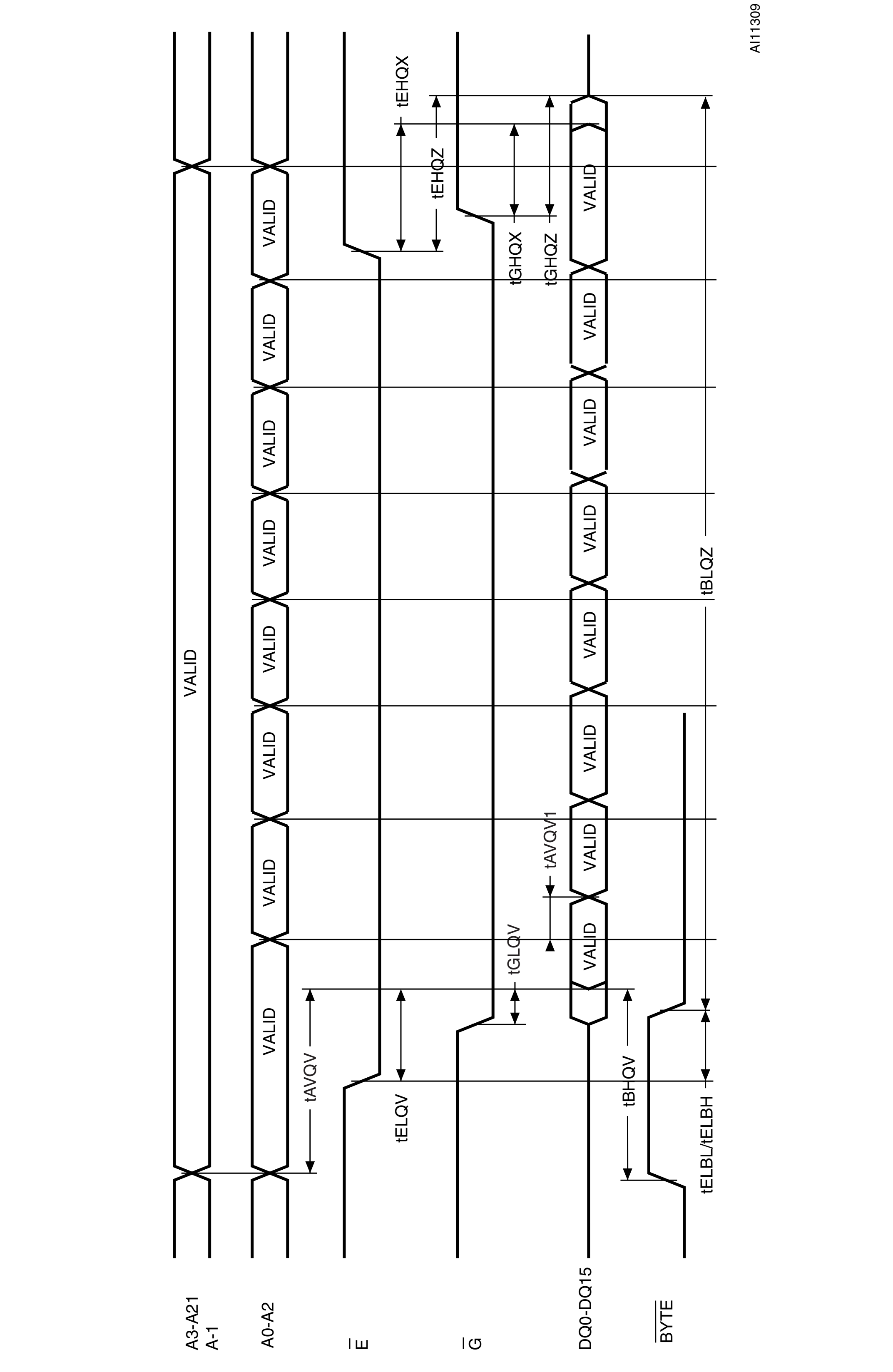


Table 16. Read AC characteristics

Symbol	Alt	Parameter	Test condition		M29DW640F		Unit
					60	70	
t_{AVAV}	t_{RC}	Address Valid to Next Address Valid	$\overline{E} = V_{IL}$, $\overline{G} = V_{IL}$	Min	60	70	ns
t_{AVQV}	t_{ACC}	Address Valid to Output Valid	$\overline{E} = V_{IL}$, $\overline{G} = V_{IL}$	Max	60	70	ns
t_{AVQV1}	t_{PAGE}	Address Valid to Output Valid (Page)	$\overline{E} = V_{IL}$, $\overline{G} = V_{IL}$	Max	25	25	ns
t_{BLQZ}	t_{FLQZ}	$\overline{\text{BYTE}}$ Low to Output Hi-Z		Max	25	25	ns
t_{BHQV}	t_{FHQV}	$\overline{\text{BYTE}}$ High to Output valid		Max	30	30	ns
t_{ELQX}[(1)]	t_{LZ}	Chip Enable Low to Output Transition	$\overline{G} = V_{IL}$	Min	0	0	ns
t_{EHQZ}[(1)]	t_{HZ}	Chip Enable High to Output Hi-Z	$\overline{G} = V_{IL}$	Max	25	25	ns
t_{ELQV}	t_{CE}	Chip Enable Low to Output Valid	$\overline{G} = V_{IL}$	Max	60	70	ns
t_{ELBL} t_{ELBH}	t_{ELFL} t_{ELFH}	Chip Enable to $\overline{\text{BYTE}}$ Low or High		Max	5	5	ns
t_{EHQX} t_{GHQX} t_{AXQX}	t_{OH}	Chip Enable, Output Enable or Address Transition to Output Transition		Min	0	0	ns
t_{GLQX}[(1)]	t_{OLZ}	Output Enable Low to Output Transition	$\overline{E} = V_{IL}$	Min	0	0	ns
t_{GLQV}	t_{OE}	Output Enable Low to Output Valid	$\overline{E} = V_{IL}$	Max	25	25	ns
t_{GHQZ}[(1)]	t_{DF}	Output Enable High to Output Hi-Z	$\overline{E} = V_{IL}$	Max	25	25	ns

1. Sampled only, not 100% tested.

Figure 12. Write AC waveforms, Write Enable controlled

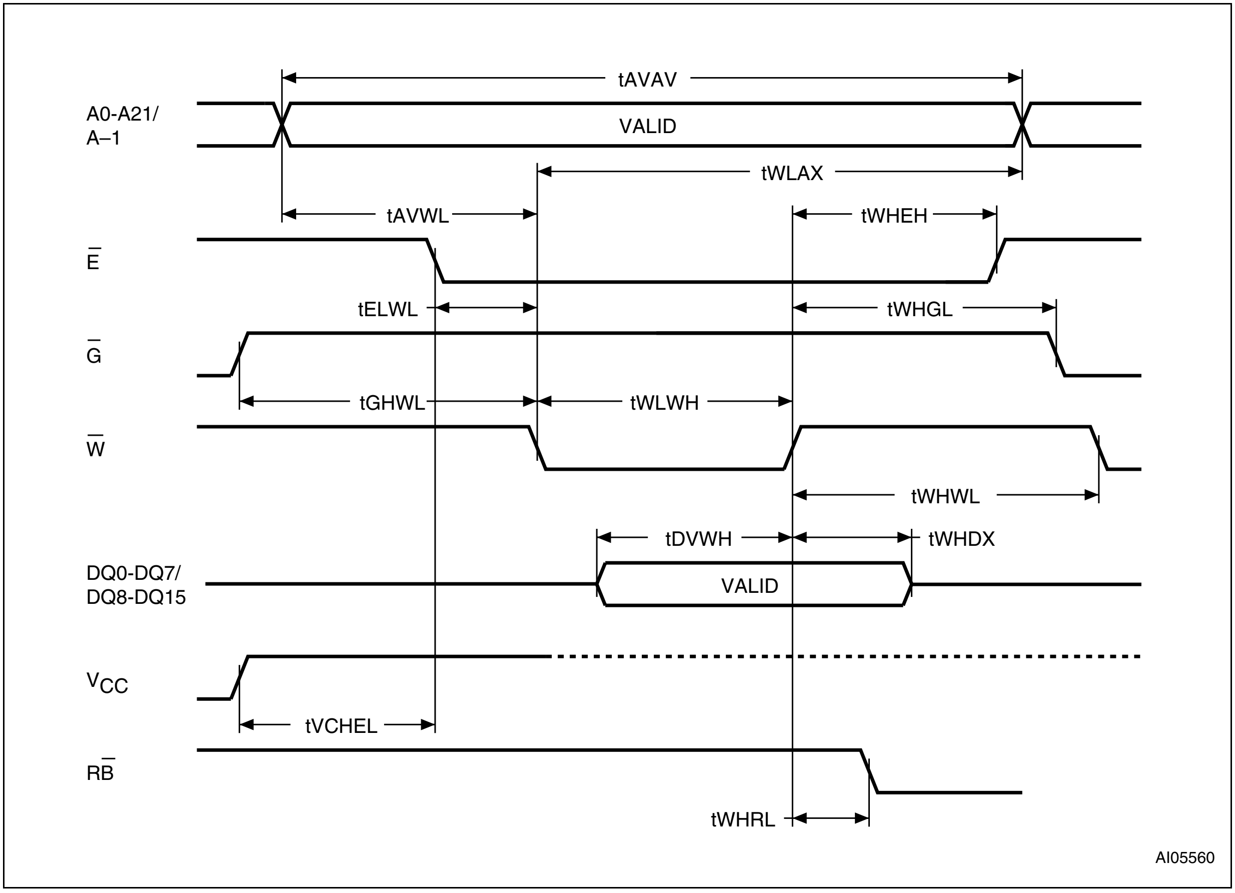


Table 17. Write AC characteristics, Write Enable controlled

Symbol	Alt	Parameter		M29DW640F		Unit
				60	70	
t_{AVAV}	t_{WC}	Address Valid to Next Address Valid	Min	60	70	ns
t_{AVWL}	t_{AS}	Address Valid to Write Enable Low	Min	0	0	ns
t_{DVWH}	t_{DS}	Input Valid to Write Enable High	Min	45	45	ns
t_{ELWL}	t_{CS}	Chip Enable Low to Write Enable Low	Min	0	0	ns
t_{GHWL}		Output Enable High to Write Enable Low	Min	0	0	ns
t_{VCHEL}	t_{VCS}	V_{CC} High to Chip Enable Low	Min	50	50	µs
t_{WLWH}	t_{WP}	Write Enable Low to Write Enable High	Min	45	45	ns
t_{WHDX}	t_{DH}	Write Enable High to Input Transition	Min	0	0	ns
t_{WHEH}	t_{CH}	Write Enable High to Chip Enable High	Min	0	0	ns
t_{WHWL}	t_{WPH}	Write Enable High to Write Enable Low	Min	30	30	ns
t_{WLAX}	t_{AH}	Write Enable Low to Address Transition	Min	45	45	ns
t_{WHGL}	t_{OEH}	Write Enable High to Output Enable Low	Min	0	0	ns
t_{WHRL}[1]	t_{BUSY}	Program/Erase Valid to R$\overline{B}$ Low	Max	30	30	ns

1. Sampled only, not 100% tested.

Figure 13. Write AC waveforms, Chip Enable controlled

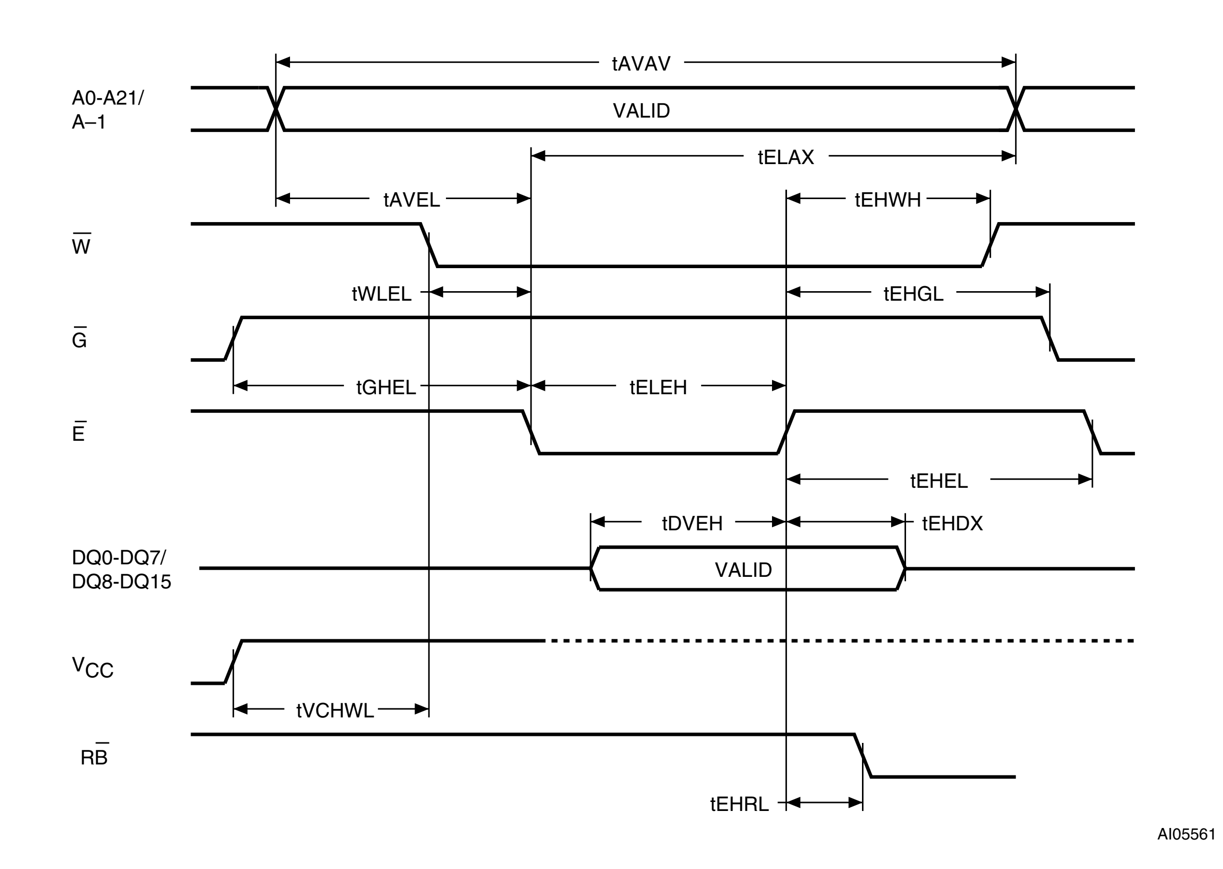
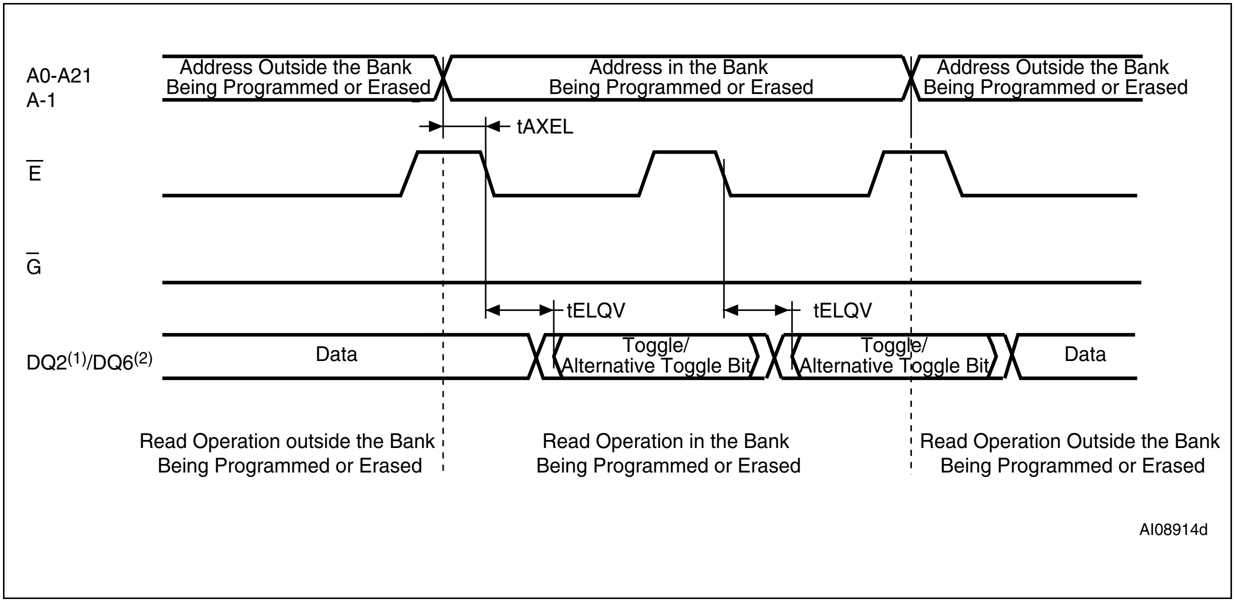


Table 18. Write AC characteristics, Chip Enable controlled

Symbol	Alt	Parameter		M29DW640F		Unit
				60	70	
t_{AVAV}	t_{WC}	Address Valid to Next Address Valid	Min	60	70	ns
t_{AVEL}	t_{AS}	Address Valid to Chip Enable Low	Min	0	0	ns
t_{DVEH}	t_{DS}	Input Valid to Chip Enable High	Min	45	45	ns
t_{ELEH}	t_{CP}	Chip Enable Low to Chip Enable High	Min	45	45	ns
t_{ELAX}	t_{AH}	Chip Enable Low to Address Transition	Min	45	45	ns
t_{EHDX}	t_{DH}	Chip Enable High to Input Transition	Min	0	0	ns
t_{EHWH}	t_{WH}	Chip Enable High to Write Enable High	Min	0	0	ns
t_{EHEL}	t_{CPH}	Chip Enable High to Chip Enable Low	Min	30	30	ns
t_{EHGL}	t_{OEH}	Chip Enable High to Output Enable Low	Min	0	0	ns
t_{EHRL}[1]	t_{BUSY}	Program/Erase Valid to R$\overline{\text{B}}$ Low	Max	30	30	ns
t_{GHEL}		Output Enable High Chip Enable Low	Min	0	0	ns
t_{VCHWL}	t_{VCS}	V_{CC} High to Write Enable Low	Min	50	50	µs
t_{WLEL}	t_{WS}	Write Enable Low to Chip Enable Low	Min	0	0	ns

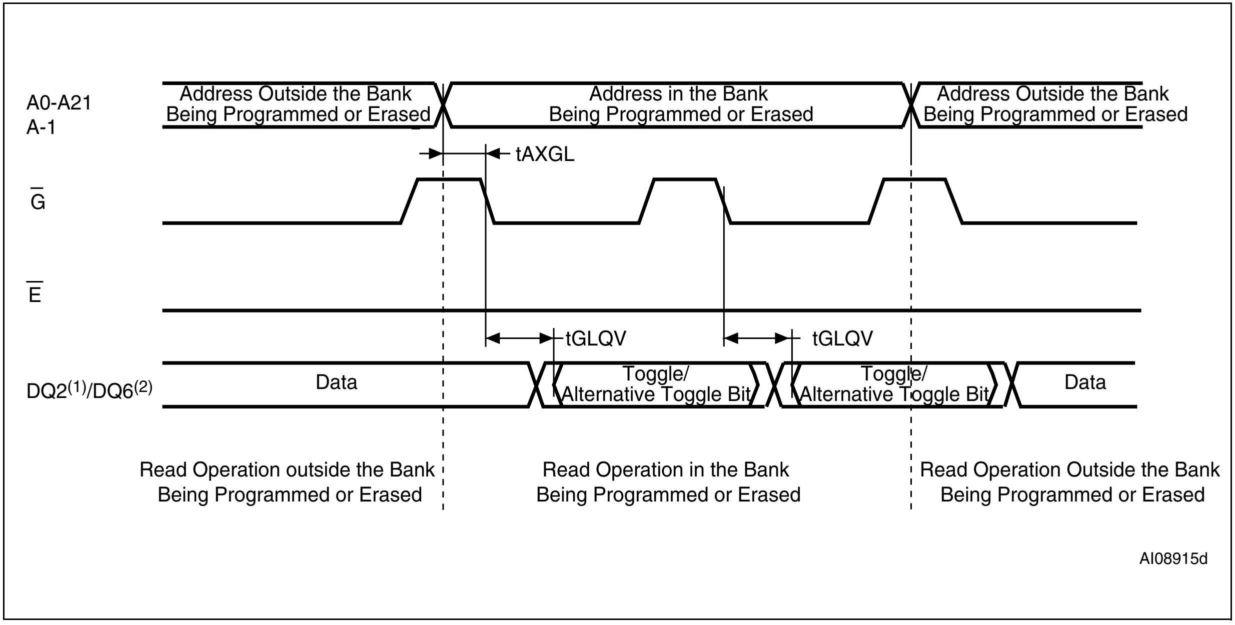
[1] Sampled only, not 100% tested.

Figure 14. Toggle and Alternative Toggle Bits mechanism, Chip Enable controlled


A0-A21
A-1
Address Outside the Bank Being Programmed or Erased
Address in the Bank Being Programmed or Erased
Address Outside the Bank Being Programmed or Erased
tAXEL
E
G
tELQV
tELQV
DQ2(1)/DQ6(2)
Data
Toggle/ Alternative Toggle Bit
Toggle/ Alternative Toggle Bit
Data
Read Operation outside the Bank Being Programmed or Erased
Read Operation in the Bank Being Programmed or Erased
Read Operation Outside the Bank Being Programmed or Erased
AI08914d

1. The Toggle Bit is output on DQ6.
2. The Alternative Toggle Bit is output on DQ2.
3. Refer to *Table 16: Read AC characteristics* for t_{ELQV} value.

Figure 15. Toggle and Alternative Toggle Bits mechanism, Output Enable controlled


A0-A21
A-1
Address Outside the Bank Being Programmed or Erased
Address in the Bank Being Programmed or Erased
Address Outside the Bank Being Programmed or Erased
tAXGL
G
E
tGLQV
tGLQV
DQ2(1)/DQ6(2)
Data
Toggle/ Alternative Toggle Bit
Toggle/ Alternative Toggle Bit
Data
Read Operation outside the Bank Being Programmed or Erased
Read Operation in the Bank Being Programmed or Erased
Read Operation Outside the Bank Being Programmed or Erased
AI08915d

1. The Toggle Bit is output on DQ6.
4. The Alternative Toggle Bit is output on DQ2.
5. Refer to *Table 16: Read AC characteristics* for t_{GLQV} value.

Table 19. Toggle and Alternative Toggle Bits AC characteristics

Symbol	Alt	Parameter				Unit
				60	70	
t_{AXEL}		Address Transition to Chip Enable Low	Min	10	10	ns
t_{AXGL}		Address Transition to Output Enable Low	Min	10	10	ns

Figure 16. Reset/Block Temporary Unprotect AC waveforms

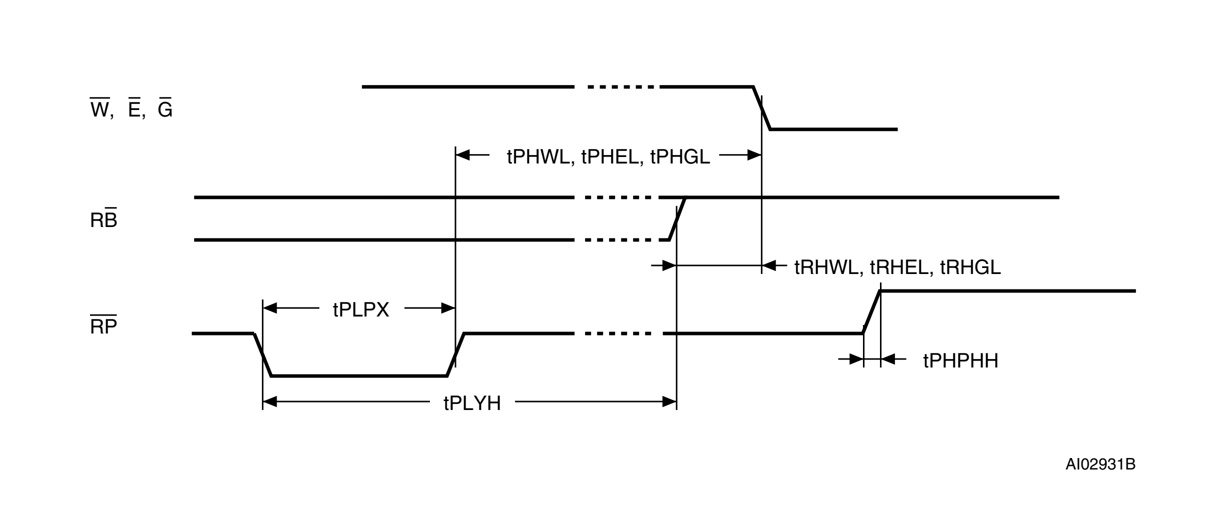


Figure 17. Accelerated Program Timing waveforms

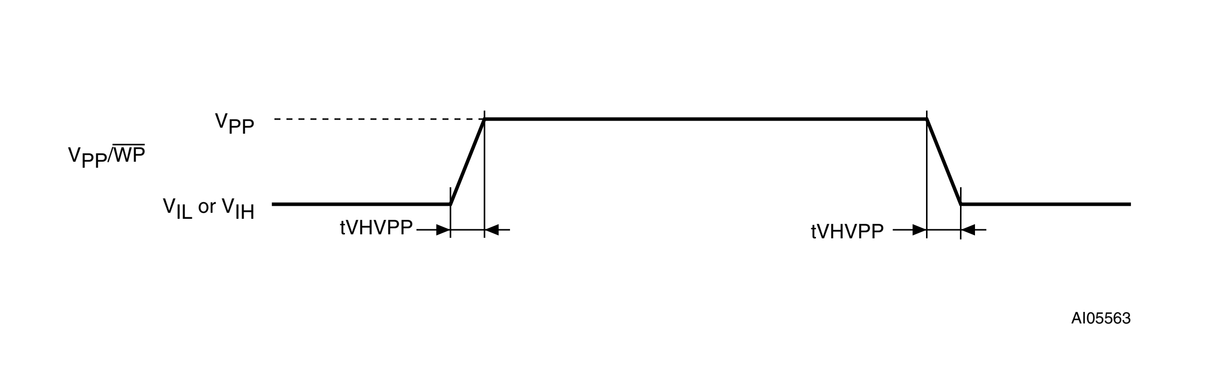


Table 20. Reset/Block Temporary Unprotect AC characteristics

Symbol	Alt	Parameter		M29DW640F		Unit
				60	70	
t_{PHWL}(1) t_{PHEL} t_{PHGL}(1)	t_{RH}	$\overline{RP}$ High to Write Enable Low, Chip Enable Low, Output Enable Low	Min	50	50	ns
t_{RHWL}(1) t_{RHEL}(1) t_{RHGL}(1)	t_{RB}	$\overline{RB}$ High to Write Enable Low, Chip Enable Low, Output Enable Low	Min	0	0	ns
t_{PLPX}	t_{RP}	$\overline{RP}$ Pulse Width	Min	500	500	ns
t_{PLYH}	t_{READY}	$\overline{RP}$ Low to Read Mode	Max	20	20	µs
t_{PHPHH}(1)	t_{VIDR}	$\overline{RP}$ Rise Time to V_{ID}	Min	500	500	ns
t_{VHVPP}(1)		V_{PP} Rise and Fall Time	Min	250	250	ns

1. Sampled only, not 100% tested.

9 Package mechanical

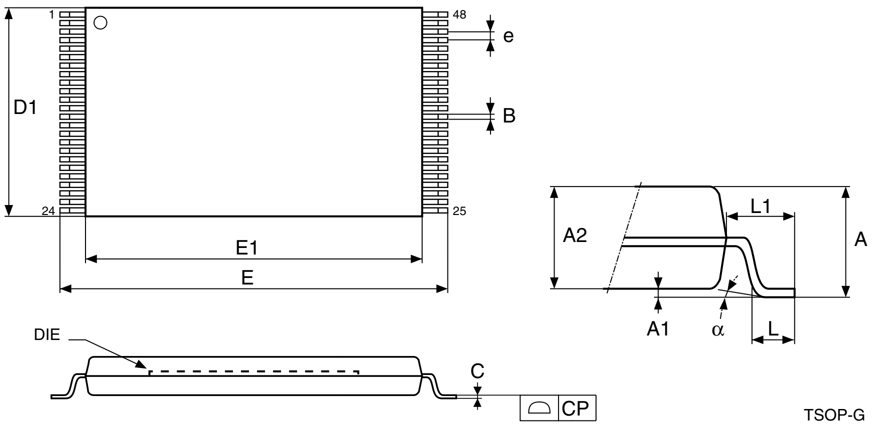


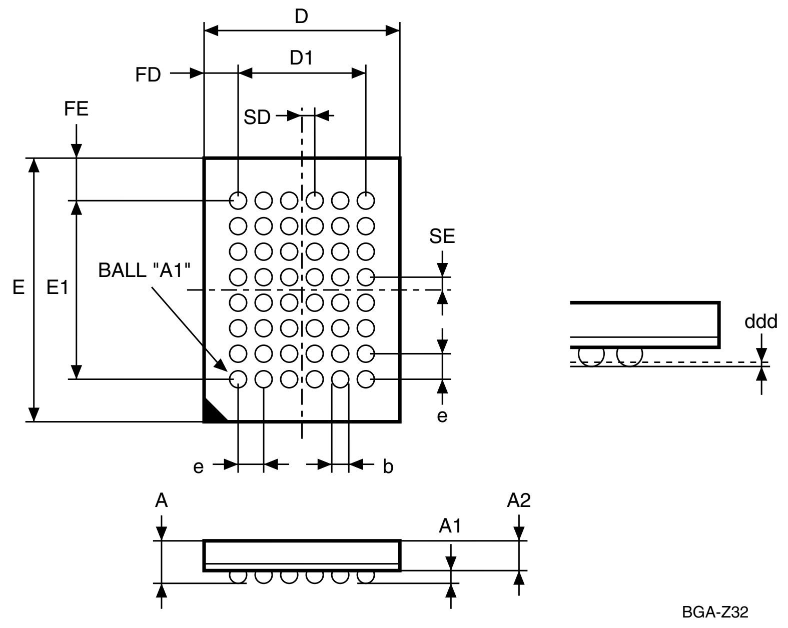
Figure 18. TSOP48 – 48 lead Plastic Thin Small Outline, 12 x 20mm, package outline

1. Drawing is not to scale.

Table 21. TSOP48 – 48 lead Plastic Thin Small Outline, 12 x 20mm, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.200			0.0472
A1	0.100	0.050	0.150	0.0039	0.0020	0.0059
A2	1.000	0.950	1.050	0.0394	0.0374	0.0413
B	0.220	0.170	0.270	0.0087	0.0067	0.0106
C		0.100	0.210		0.0039	0.0083
CP			0.080			0.0031
D1	12.000	11.900	12.100	0.4724	0.4685	0.4764
E	20.000	19.800	20.200	0.7874	0.7795	0.7953
E1	18.400	18.300	18.500	0.7244	0.7205	0.7283
e	0.500	–	–	0.0197	–	–
L	0.600	0.500	0.700	0.0236	0.0197	0.0276
L1	0.800			0.0315		
α	3°	0°	5°	3°	0°	5°

Figure 19. TFBGA48 6x8mm - 6x8 active ball array, 0.8mm pitch, package outline



1. Drawing is not to scale.

Table 22. TFBGA48 6x8mm - 6x8 active ball array, 0.8mm pitch, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.200			0.0472
A1		0.260			0.0102	
A2			0.900			0.0354
b		0.350	0.450		0.0138	0.0177
D	6.000	5.900	6.100	0.2362	0.2323	0.2402
D1	4.000	–	–	0.1575	–	–
ddd			0.100			0.0039
E	8.000	7.900	8.100	0.3150	0.3110	0.3189
E1	5.600	–	–	0.2205	–	–
e	0.800	–	–	0.0315	–	–
FD	1.000	–	–	0.0394	–	–
FE	1.200	–	–	0.0472	–	–
SD	0.400	–	–	0.0157	–	–
SE	0.400	–	–	0.0157	–	–

10 Part numbering

Table 23. Ordering information scheme

Example: M29DW640F 70 N 1 T

Device Type

M29

Architecture

D = Dual or Multiple Bank

Operating Voltage

W = V_{CC} = 2.7 to 3.6V

Device Function

640F = 64 Mbit (x8/x16), Boot Block, 8+24+24+8 partitioning, 0.13µm technology

Speed

60 = 60ns

70 = 70ns

Package

N = TSOP48: 12 x 20 mm

ZE = TFBGA48 6 x 8mm, 0.8 mm pitch

Temperature Range

1 = 0 to 70 °C

6 = –40 to 85 °C

Option

Blank = Standard Packing

T = Tape & Reel Packing

E = ECOPACK Package, Standard Packing

F = ECOPACK Package, Tape & Reel Packing

Note: *This product is also available with the Extended Block factory locked. For further details and ordering information contact your nearest Numonyx sales office.*

Devices are shipped from the factory with the memory content bits erased to ’1’.

For a list of available options (Speed, Package, etc.) or for further information on any aspect of this device, please contact your nearest Numonyx Sales Office.

Appendix A Block addresses

Table 24. Block addresses

Bank	Block	(KBytes/ KWords)	Protection Block Group	(x8)	(x16)
Bank A	0	8/4	Protection Group	000000h-001FFFh[1]	000000h–000FFFh[1]
	1	8/4	Protection Group	002000h-003FFFh[1]	001000h–001FFFh[1]
	2	8/4	Protection Group	004000h-005FFFh[1]	002000h–002FFFh[1]
	3	8/4	Protection Group	006000h-007FFFh[1]	003000h–003FFFh[1]
	4	8/4	Protection Group	008000h-009FFFh[1]	004000h–004FFFh[1]
	5	8/4	Protection Group	00A000h-00BFFFh[1]	005000h–005FFFh[1]
	6	8/4	Protection Group	00C000h-00DFFFh[1]	006000h–006FFFh[1]
	7	8/4	Protection Group	00E000h-00FFFFh[1]	007000h–007FFFh[1]
	8	64/32	Protection Group	010000h-01FFFFh	008000h–00FFFFh
	9	64/32		020000h-02FFFFh	010000h–017FFFh
	10	64/32		030000h-03FFFFh	018000h–01FFFFh
	11	64/32	Protection Group	040000h-04FFFFh	020000h–027FFFh
	12	64/32		050000h-05FFFFh	028000h–02FFFFh
	13	64/32		060000h-06FFFFh	030000h–037FFFh
	14	64/32		070000h-07FFFFh	038000h–03FFFFh
	15	64/32	Protection Group	080000h-08FFFFh	040000h–047FFFh
	16	64/32		090000h-09FFFFh	048000h–04FFFFh
	17	64/32		0A0000h-0AFFFFh	050000h–057FFFh
	18	64/32		0B0000h-0BFFFFh	058000h–05FFFFh
	19	64/32	Protection Group	0C0000h-0CFFFFh	060000h–067FFFh
	20	64/32		0D0000h-0DFFFFh	068000h–06FFFFh
	21	64/32		0E0000h-0EFFFFh	070000h–077FFFh
	22	64/32		0F0000h-0FFFFFh	078000h–07FFFFh

Table 24. Block addresses (continued)

Bank	Block	(KBytes/ KWords)	Protection Block Group	(x8)	(x16)
Bank B	23	64/32	Protection Group	100000h-10FFFFh	080000h–087FFFh
	24	64/32		110000h-11FFFFh	088000h–08FFFFh
	25	64/32		120000h-12FFFFh	090000h–097FFFh
	26	64/32		130000h-13FFFFh	098000h–09FFFFh
	27	64/32	Protection Group	140000h-14FFFFh	0A0000h–0A7FFFh
	28	64/32		150000h-15FFFFh	0A8000h–0AFFFFh
	29	64/32		160000h-16FFFFh	0B0000h–0B7FFFh
	30	64/32		170000h-17FFFFh	0B8000h–0BFFFFh
	31	64/32	Protection Group	180000h-18FFFFh	0C0000h–0C7FFFh
	32	64/32		190000h-19FFFFh	0C8000h–0CFFFFh
	33	64/32		1A0000h-1AFFFFh	0D0000h–0D7FFFh
	34	64/32		1B0000h-1BFFFFh	0D8000h–0DFFFFh
	35	64/32	Protection Group	1C0000h-1CFFFFh	0E0000h–0E7FFFh
	36	64/32		1D0000h-1DFFFFh	0E8000h–0EFFFFh
	37	64/32		1E0000h-1EFFFFh	0F0000h–0F7FFFh
	38	64/32		1F0000h-1FFFFFh	0F8000h–0FFFFFh
	39	64/32	Protection Group	200000h-20FFFFh	100000h–107FFFh
	40	64/32		210000h-21FFFFh	108000h–10FFFFh
	41	64/32		220000h-22FFFFh	110000h–117FFFh
	42	64/32		230000h-23FFFFh	118000h–11FFFFh
	43	64/32	Protection Group	240000h-24FFFFh	120000h–127FFFh
	44	64/32		250000h-25FFFFh	128000h–12FFFFh
	45	64/32		260000h-26FFFFh	130000h–137FFFh
	46	64/32		270000h-27FFFFh	138000h–13FFFFh
	47	64/32	Protection Group	280000h-28FFFFh	140000h–147FFFh
	48	64/32		290000h-29FFFFh	148000h–14FFFFh
	49	64/32		2A0000h-2AFFFFh	150000h–157FFFh
	50	64/32		2B0000h-2BFFFFh	158000h–15FFFFh
	51	64/32	Protection Group	2C0000h-2CFFFFh	160000h–167FFFh
	52	64/32		2D0000h-2DFFFFh	168000h–16FFFFh
	53	64/32		2E0000h-2EFFFFh	170000h–177FFFh
	54	64/32		2F0000h-2FFFFFh	178000h–17FFFFh

Table 24. Block addresses (continued)

Bank	Block	(KBytes/ KWords)	Protection Block Group	(x8)	(x16)
Bank B	55	64/32	Protection Group	300000h-30FFFFh	180000h–187FFFh
	56	64/32		310000h-31FFFFh	188000h–18FFFFh
	57	64/32		320000h-32FFFFh	190000h–197FFFh
	58	64/32		330000h-33FFFFh	198000h–19FFFFh
	59	64/32	Protection Group	340000h-34FFFFh	1A0000h–1A7FFFh
	60	64/32		350000h-35FFFFh	1A8000h–1AFFFFh
	61	64/32		360000h-36FFFFh	1B0000h–1B7FFFh
	62	64/32		370000h-37FFFFh	1B8000h–1BFFFFh
	63	64/32	Protection Group	380000h-38FFFFh	1C0000h–1C7FFFh
	64	64/32		390000h-39FFFFh	1C8000h–1CFFFFh
	65	64/32		3A0000h-3AFFFFh	1D0000h–1D7FFFh
	66	64/32		3B0000h-3BFFFFh	1D8000h–1DFFFFh
	67	64/32	Protection Group	3C0000h-3CFFFFh	1E0000h–1E7FFFh
	68	64/32		3D0000h-3DFFFFh	1E8000h–1EFFFFh
	69	64/32		3E0000h-3EFFFFh	1F0000h–1F7FFFh
	70	64/32		3F0000h-3FFFFFh	1F8000h–1FFFFFh
Bank C	71	64/32	Protection Group	400000h–40FFFFh	200000h–207FFFh
	72	64/32		410000h–41FFFFh	208000h–20FFFFh
	73	64/32		420000h–42FFFFh	210000h–217FFFh
	74	64/32		430000h–43FFFFh	218000h–21FFFFh
	75	64/32	Protection Group	440000h–44FFFFh	220000h–227FFFh
	76	64/32		450000h–45FFFFh	228000h–22FFFFh
	77	64/32		460000h–46FFFFh	230000h–237FFFh
	78	64/32		470000h–47FFFFh	238000h–23FFFFh
	79	64/32	Protection Group	480000h–48FFFFh	240000h–247FFFh
	80	64/32		490000h–49FFFFh	248000h–24FFFFh
	81	64/32		4A0000h–4AFFFFh	250000h–257FFFh
	82	64/32		4B0000h–4BFFFFh	258000h–25FFFFh
	83	64/32	Protection Group	4C0000h–4CFFFFh	260000h–267FFFh
	84	64/32		4D0000h–4DFFFFh	268000h–26FFFFh
	85	64/32		4E0000h–4EFFFFh	270000h–277FFFh
	86	64/32		4F0000h–4FFFFFh	278000h–27FFFFh

Table 24. Block addresses (continued)

Bank	Block	(KBytes/ KWords)	Protection Block Group	(x8)	(x16)
Bank C	87	64/32	Protection Group	500000h–50FFFFh	280000h–287FFFh
	88	64/32		510000h–51FFFFh	288000h–28FFFFh
	89	64/32		520000h–52FFFFh	290000h–297FFFh
	90	64/32		530000h–53FFFFh	298000h–29FFFFh
	91	64/32	Protection Group	540000h–54FFFFh	2A0000h–2A7FFFh
	92	64/32		550000h–55FFFFh	2A8000h–2AFFFFh
	93	64/32		560000h–56FFFFh	2B0000h–2B7FFFh
	94	64/32		570000h–57FFFFh	2B8000h–2BFFFFh
	95	64/32	Protection Group	580000h–58FFFFh	2C0000h–2C7FFFh
	96	64/32		590000h–59FFFFh	2C8000h–2CFFFFh
	97	64/32		5A0000h–5AFFFFh	2D0000h–2D7FFFh
	98	64/32		5B0000h–5BFFFFh	2D8000h–2DFFFFh
	99	64/32	Protection Group	5C0000h–5CFFFFh	2E0000h–2E7FFFh
	100	64/32		5D0000h–5DFFFFh	2E8000h–2EFFFFh
	101	64/32		5E0000h–5EFFFFh	2F0000h–2F7FFFh
	102	64/32		5F0000h–5FFFFFh	2F8000h–2FFFFFh
	103	64/32	Protection Group	600000h–60FFFFh	300000h–307FFFh
	104	64/32		610000h–61FFFFh	308000h–30FFFFh
	105	64/32		620000h–62FFFFh	310000h–317FFFh
	106	64/32		630000h–63FFFFh	318000h–31FFFFh
	107	64/32	Protection Group	640000h–64FFFFh	320000h–327FFFh
	108	64/32		650000h–65FFFFh	328000h–32FFFFh
	109	64/32		660000h–66FFFFh	330000h–337FFFh
	110	64/32		670000h–67FFFFh	338000h–33FFFFh
	111	64/32	Protection Group	680000h–68FFFFh	340000h–347FFFh
	112	64/32		690000h–69FFFFh	348000h–34FFFFh
	113	64/32		6A0000h–6AFFFFh	350000h–357FFFh
	114	64/32		6B0000h–6BFFFFh	358000h–35FFFFh
	115	64/32	Protection Group	6C0000h–6CFFFFh	360000h–367FFFh
	116	64/32		6D0000h–6DFFFFh	368000h–36FFFFh
	117	64/32		6E0000h–6EFFFFh	370000h–377FFFh
	118	64/32		6F0000h–6FFFFFh	378000h–37FFFFh

Table 24. Block addresses (continued)

Bank	Block	(KBytes/ KWords)	Protection Block Group	(x8)	(x16)
Bank D	119	64/32	Protection Group	700000h–70FFFFh	380000h–387FFFh
	120	64/32		710000h–71FFFFh	388000h–38FFFFh
	121	64/32		720000h–72FFFFh	390000h–397FFFh
	122	64/32		730000h–73FFFFh	398000h–39FFFFh
	123	64/32	Protection Group	740000h–74FFFFh	3A0000h–3A7FFFh
	124	64/32		750000h–75FFFFh	3A8000h–3AFFFFh
	125	64/32		760000h–76FFFFh	3B0000h–3B7FFFh
	126	64/32		770000h–77FFFFh	3B8000h–3BFFFFh
	127	64/32	Protection Group	780000h–78FFFFh	3C0000h–3C7FFFh
	128	64/32		790000h–79FFFFh	3C8000h–3CFFFFh
	129	64/32		7A0000h–7AFFFFh	3D0000h–3D7FFFh
	130	64/32		7B0000h–7BFFFFh	3D8000h–3DFFFFh
	131	64/32	Protection Group	7C0000h–7CFFFFh	3E0000h–3E7FFFh
	132	64/32		7D0000h–7DFFFFh	3E8000h–3EFFFFh
	133	64/32		7E0000h–7EFFFFh	3F0000h–3F7FFFh
	134	8/4	Protection Group	7F0000h–7F1FFFh	3F8000h–3F8FFFh
	135	8/4	Protection Group	7F2000h–7F3FFFh	3F9000h–3F9FFFh
	136	8/4	Protection Group	7F4000h–7F5FFFh	3FA000h–3FAFFFh
	137	8/4	Protection Group	7F6000h–7F7FFFh	3FB000h–3FBFFFh
	138	8/4	Protection Group	7F8000h–7F9FFFh	3FC000h–3FCFFFh
	139	8/4	Protection Group	7FA000h–7FBFFFh	3FD000h–3FDFFFh
	140	8/4	Protection Group	7FC000h–7FDFFFh	3FE000h–3FEFFFh
	141	8/4	Protection Group	7FE000h–7FFFFFh	3FF000h–3FFFFFh

1. Used as Extended Block addresses in Extended Block mode.

Appendix B Common Flash Interface (CFI)

The Common Flash Interface is a JEDEC approved, standardized data structure that can be read from the Flash memory device. It allows a system software to query the device to determine various electrical and timing parameters, density information and functions supported by the memory. The system can interface easily with the device, enabling the software to upgrade itself when necessary.

When the Read CFI Query command is issued the addressed bank enters Read CFI Query mode and read operations in the same bank (A21-A19) output the CFI data. *Table 25*, *Table 26*, *Table 27*, *Table 28*, *Table 29* and *Table 30* show the addresses (A-1, A0-A10) used to retrieve the data.

The CFI data structure also contains a security area where a 64 bit unique security number is written (see *Table 30: Security Code Area*). This area can be accessed only in Read mode by the final user. It is impossible to change the security number after it has been written by Numonyx.

Table 25. Query structure overview

Address		Sub-section name	Description
x16	x8		
10h	20h	CFI Query Identification String	Command set ID and algorithm data offset
1Bh	36h	System Interface Information	Device timing & voltage information
27h	4Eh	Device Geometry Definition	Flash device layout
40h	80h	Primary Algorithm-specific Extended Query table	Additional information specific to the Primary Algorithm (optional)
61h	C2h	Security Code Area	64 bit unique device number

1. Query data are always presented on the lowest order data outputs.

Table 26. CFI Query Identification String

Address		Data	Description	Value
x16	x8			
10h	20h	0051h		“Q”
11h	22h	0052h	Query Unique ASCII String "QRY"	"R"
12h	24h	0059h		"Y"
13h	26h	0002h	Primary Algorithm Command Set and Control Interface ID code 16 bit ID code defining a specific algorithm	AMD Compatible
14h	28h	0000h		
15h	2Ah	0040h	Address for Primary Algorithm extended Query table (see *Table 29*)	P = 40h
16h	2Ch	0000h		
17h	2Eh	0000h	Alternate Vendor Command Set and Control Interface ID Code second vendor - specified algorithm supported	NA
18h	30h	0000h		
19h	32h	0000h	Address for Alternate Algorithm extended Query table	NA
1Ah	34h	0000h		

1. Query data are always presented on the lowest order data outputs (DQ7-DQ0) only. DQ8-DQ15 are ‘0’.

Table 27. CFI Query System Interface Information

Address		Data	Description	Value
x16	x8			
1Bh	36h	0027h	V_{CC} Logic Supply Minimum Program/Erase voltage bit 7 to 4 BCD value in volts bit 3 to 0 BCD value in 100 mV	2.7V
1Ch	38h	0036h	V_{CC} Logic Supply Maximum Program/Erase voltage bit 7 to 4 BCD value in volts bit 3 to 0 BCD value in 100 mV	3.6V
1Dh	3Ah	00B5h	V_{PP} [Programming] Supply Minimum Program/Erase voltage bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 mV	11.5V
1Eh	3Ch	00C5h	V_{PP} [Programming] Supply Maximum Program/Erase voltage bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 mV	12.5V
1Fh	3Eh	0004h	Typical timeout per single Byte/Word program = 2^n µs	16µs
20h	40h	0000h	Typical timeout for minimum size write buffer program = 2^n µs	NA
21h	42h	000Ah	Typical timeout per individual block erase = 2^n ms	1s
22h	44h	0000h	Typical timeout for full Chip Erase = 2^n ms	NA
23h	46h	0004h	Maximum timeout for Byte/Word program = 2^n times typical	256 µs
24h	48h	0000h	Maximum timeout for write buffer program = 2^n times typical	NA
25h	4Ah	0003h	Maximum timeout per individual block erase = 2^n times typical	8s
26h	4Ch	0000h	Maximum timeout for Chip Erase = 2^n times typical	NA

Table 28. Device Geometry Definition

Address		Data	Description	Value
x16	x8			
27h	4Eh	0017h	Device Size = 2^n in number of Bytes	8 MBytes
28h 29h	50h 52h	0002h 0000h	Flash Device Interface Code description	x8, x16 Async.
2Ah 2Bh	54h 56h	0003h 0000h	Maximum number of Bytes in multi-Byte program or page = 2^n	8
2Ch	58h	0003h	Number of Erase Block Regions[1]. It specifies the number of regions containing contiguous Erase Blocks of the same size.	3
2Dh 2Eh	5Ah 5Ch	0007h 0000h	Erase Block Region 1 Information Number of Erase Blocks of identical size = 0007h+1	8
2Fh 30h	5Eh 60h	0020h 0000h	Erase Block Region 1 Information Block size in Region 1 = 0020h * 256 Byte	8 KBytes
31h 32h	62h 64h	007Dh 0000h	Erase Block Region 2 Information Number of Erase Blocks of identical size = 007Dh+1	126
33h 34h	66h 68h	0000h 0001h	Erase Block Region 2 Information Block size in Region 2 = 0100h * 256 Byte	64 KBytes
35h 36h	6Ah 6Ch	0007h 0000h	Erase Block Region 3 information Number of Erase Blocks of identical size = 0007h + 1	8
37h 38h	6Eh 70h	0020h 0000h	Erase Block Region 3 information Block size in region 3 = 0020h * 256 Bytes	8 KBytes

1. Erase Block Region 1 corresponds to addresses 000000h to 007FFFh; Erase block Region 2 corresponds to addresses 008000h to 3F7FFFh and Erase Block Region 3 corresponds to addresses 3F8000h to 3FFFFFh.

Table 29. Primary Algorithm-specific Extended Query table

Address		Data	Description	Value
x16	x8			
40h	80h	0050h	Primary Algorithm extended Query table unique ASCII string “PRI”	"P"
41h	82h	0052h		"R"
42h	84h	0049h		"I"
43h	86h	0031h	Major version number, ASCII	"1"
44h	88h	0033h	Minor version number, ASCII	"3"
45h	8Ah	0000h	Address Sensitive Unlock (bits 1 to 0) 00 = required, 01= not required Silicon Revision Number (bits 7 to 2)	Yes
46h	8Ch	0002h	Erase Suspend 00 = not supported, 01 = Read only, 02 = Read and Write	2
47h	8Eh	0001h	Block Protection 00 = not supported, x = number of sectors in per group	1
48h	90h	0001h	Temporary Block Unprotect 00 = not supported, 01 = supported	Yes
49h	92h	0005h	Block Protect /Unprotect 04 = M29W400B 05=	5
4Ah	94h	0077h	Simultaneous Operations, x = number of blocks (excluding Bank A)	119
4Bh	96h	0000h	Burst Mode, 00 = not supported, 01 = supported	No
4Ch	98h	0002h	Page Mode, 00 = not supported, 01 = 4 page Word, 02 = 8 page Word	Yes
4Dh	9Ah	00B5h	V_{PP} Supply Minimum Program/Erase voltage bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 mV	11.5V
4Eh	9Ch	00C5h	V_{PP} Supply Maximum Program/Erase voltage bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 mV	12.5V
4Fh	9Eh	0001h	Top/Bottom Boot Block Flag 00h = uniform device 01h = 8 x8 KByte Blocks, Top and Bottom Boot with Write Protect 02h = Bottom boot device 03h = Top Boot Device 04h = Both Top and Bottom	T/B
50h	A0h	0001h	Program Suspend, 00 = not supported, 01 = supported	Yes
57h	AEh	0004h	Bank Organization, 00 = data at 4Ah is zero X = number of banks	4

Table 29. Primary Algorithm-specific Extended Query table

Address		Data	Description	Value
x16	x8			
58h	B0h	0017h	Bank A information X = number of blocks in Bank A	23
59h	B2h	0030h	Bank B information X = number of blocks in Bank B	48
5Ah	B4h	0030h	Bank C information X = number of blocks in Bank C	48
5Bh	B6h	0017h	Bank D information X = number of blocks in Bank D	23

Table 30. Security Code Area

Address		Data	Description
x16	x8		
61h	C3h, C2h	XXXX	64 bit: unique device number
62h	C5h, C4h	XXXX	
63h	C7h, C6h	XXXX	
64h	C9h, C8h	XXXX	

Appendix C Extended Memory Block

The has an extra block, the Extended Block, that can be accessed using a dedicated command.

This Extended Block is 128 Words in x16 mode and 256 Bytes in x8 mode. It is used as a security block (to provide a permanent security identification number) or to store additional information.

The Extended Block is either Factory Locked or Customer Lockable, its status is indicated by bit DQ7. This bit is permanently set to either ‘1’ or ‘0’ at the factory and cannot be changed. When set to ‘1’, it indicates that the device is factory locked and the Extended Block is protected. When set to ‘0’, it indicates that the device is customer lockable and the Extended Block is unprotected. Bit DQ7 being permanently locked to either ‘1’ or ‘0’ is another security feature which ensures that a customer lockable device cannot be used instead of a factory locked one.

Bit DQ7 is the most significant bit in the Extended Block Verify Code and a specific procedure must be followed to read it. See “Extended Block Indicator Bit” in *Table 4: Bus operations, BYTE =* V_{IL} and *Table 5: Bus operations, BYTE =* V_{IH}, respectively, for details of how to read bit DQ7.

The Extended Block can only be accessed when the device is in Extended Block mode. For details of how the Extended Block mode is entered and exited, refer to *Section 4.3: Block Protection commands* and *Section 4.3.2: Exit Extended Block command*, and to *Table 6: Commands, 16-bit mode, BYTE =* V_{IH} and *Table 7: Commands, 8-bit mode, BYTE =* V_{IL}, respectively.

C.1 Factory Locked Extended Block

In devices where the Extended Block is factory locked, the Security Identification Number is written to the Extended Block address space (see *Table 31: Extended Block address and data*) in the factory. The DQ7 bit is set to ‘1’ and the Extended Block cannot be unprotected.

C.2 Customer Lockable Extended Block

A device where the Extended Block is customer lockable is delivered with the DQ7 bit set to ‘0’ and the Extended Block unprotected. It is up to the customer to program and protect the Extended Block but care must be taken because the protection of the Extended Block is not reversible.

There are two ways of protecting the Extended Block:

- Issue the Enter Extended Block command to place the device in Extended Block mode, then use the In-System Technique with $\overline{RP}$ either at V_{IH} or at V_{ID} (refer to *Appendix D*, *Figure 22: In-System Equipment Group Protect flowchart* and *Figure 23: In-System Equipment Chip Unprotect flowchart*, for a detailed explanation of the technique).
- Issue the Enter Extended Block command to place the device in Extended Block mode, then use the Programmer Technique (refer to *Appendix D*, *Figure 20: Programmer Equipment Group Protect flowchart* and *Figure 21: Programmer Equipment Chip Unprotect flowchart*, for a detailed explanation of the technique).

Once the Extended Block is programmed and protected, the Exit Extended Block command must be issued to exit the Extended Block mode and return the device to Read mode.

Table 31. Extended Block address and data

Device	Address[1]		Data		
	x8	x16	Factory Locked		Customer Lockable
	000000h-00000Fh	000000h-000007h	Random Number	Security Identification Number	Determined by Customer
	000010h-00007Fh	000008h-00003Fh	ESN[2]		
	000080h-0000FFh	000040h-00007Fh	Unavailable		

1. See *Table 24: Block addresses*.
2. ENS = Electronic Serial Number.

Appendix D Block protection

Block protection can be used to prevent any operation from modifying the data stored in the memory. The blocks are protected in groups, refer to *Appendix A*, *Table 24* for details of the Protection Groups. Once protected, Program and Erase operations within the protected group fail to change the data.

There are three techniques that can be used to control Block Protection, these are the Programmer technique, the In-System technique and Temporary Unprotection. Temporary Unprotection is controlled by the Reset/Block Temporary Unprotection pin, $\overline{\text{RP}}$; this is described in the Signal Descriptions section.

To protect the Extended Block issue the Enter Extended Block command and then use either the Programmer or In-System technique. Once protected issue the Exit Extended Block command to return to read mode. The Extended Block protection is irreversible, once protected the protection cannot be undone.

D.1 Programmer technique

The Programmer technique uses high (V_{ID}) voltage levels on some of the bus pins. These cannot be achieved using a standard microprocessor bus, therefore the technique is recommended only for use in Programming Equipment.

To protect a group of blocks follow the flowchart in *Figure 20: Programmer Equipment Group Protect flowchart*. To unprotect the whole chip it is necessary to protect all of the groups first, then all groups can be unprotected at the same time. To unprotect the chip follow *Figure 21: Programmer Equipment Chip Unprotect flowchart*. *Table 32: Programmer technique bus operations, BYTE = V_{IH} or V_{IL}*, gives a summary of each operation.

The timing on these flowcharts is critical. Care should be taken to ensure that, where a pause is specified, it is followed as closely as possible. Do not abort the procedure before reaching the end. Chip Unprotect can take several seconds and a user message should be provided to show that the operation is progressing.

D.2 In-System technique

The In-System technique requires a high voltage level on the Reset/Blocks Temporary Unprotect pin, $\overline{\text{RP}}$ [(1)]. This can be achieved without violating the maximum ratings of the components on the microprocessor bus, therefore this technique is suitable for use after the memory has been fitted to the system.

To protect a group of blocks follow the flowchart in *Figure 22: In-System Equipment Group Protect flowchart*. To unprotect the whole chip it is necessary to protect all of the groups first, then all the groups can be unprotected at the same time. To unprotect the chip follow *Figure 23: In-System Equipment Chip Unprotect flowchart*.

The timing on these flowcharts is critical. Care should be taken to ensure that, where a pause is specified, it is followed as closely as possible. Do not allow the microprocessor to service interrupts that will upset the timing and do not abort the procedure before reaching the end. Chip Unprotect can take several seconds and a user message should be provided to show that the operation is progressing.

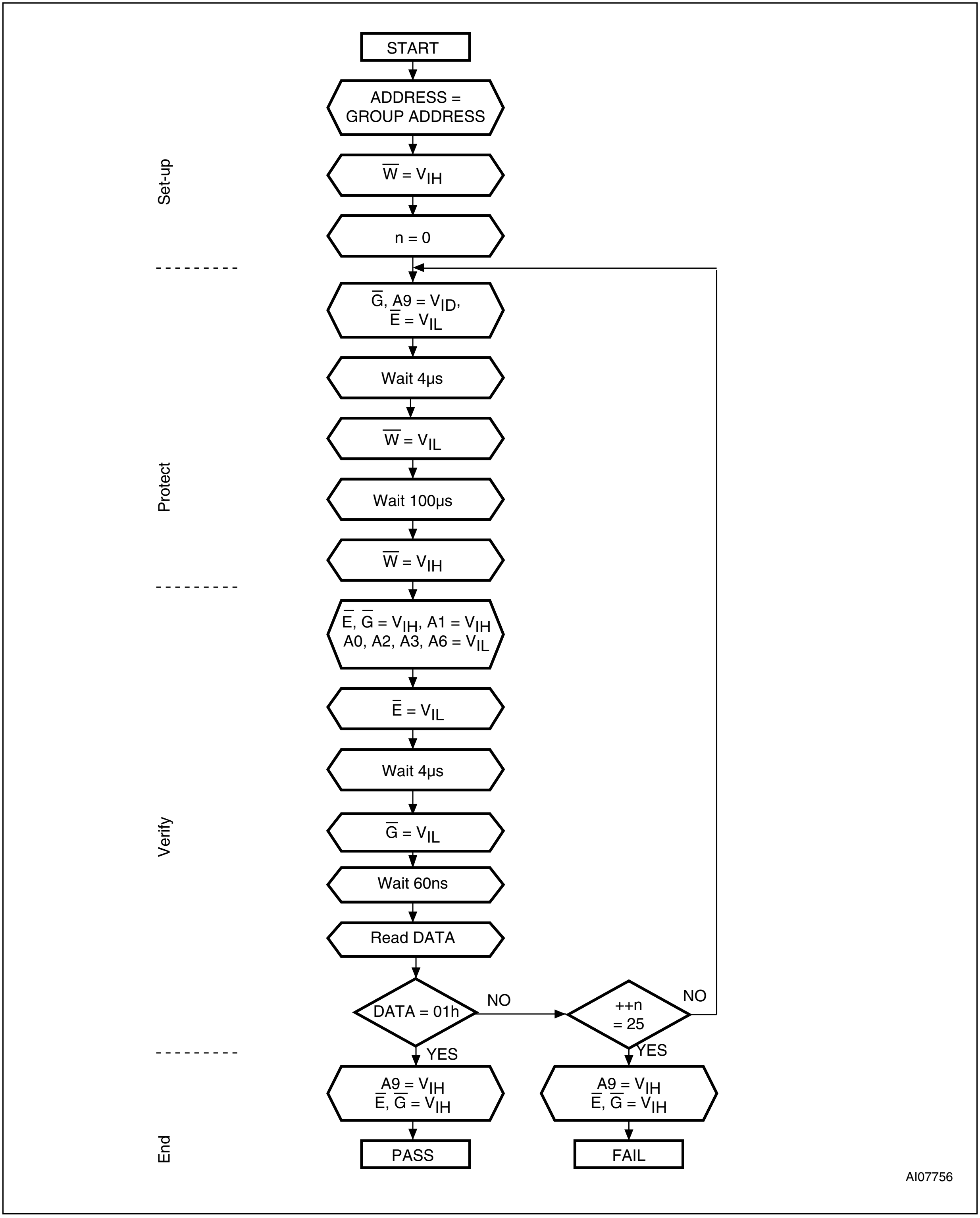
Note: *$\overline{RP}$ can be either at V_{IH} or at V_{ID} when using the In-System Technique to protect the Extended Block.*

Table 32. Programmer technique bus operations, $\overline{\text{BYTE}}$ = V_{IH} or V_{IL}

Operation	E	G	W	Address Inputs A0-A21	Data Inputs/Outputs DQ15A–1, DQ14-DQ0
Block (Group) Protect[1]	V_{IL}	V_{ID}	V_{IL} Pulse	A9 = V_{ID}, A12-A21 Block Address Others = X	X
Chip Unprotect	V_{ID}	V_{ID}	V_{IL} Pulse	A9 = V_{ID}, A12 = V_{IH}, A15 = V_{IH} Others = X	X
Block (Group) Protect Verify	V_{IL}	V_{IL}	V_{IH}	A0 = V_{IL}, A1 = V_{IH}, A2 = V_{IL}, A3 = V_{IL}, A6 = V_{IL}, A9 = V_{ID}, A12-A21 Block address Others = X	Pass = xx01h Retry = xx00h.
Block (Group) Unprotect Verify	V_{IL}	V_{IL}	V_{IH}	A0 = V_{IL}, A1 = V_{IH}, A2 = V_{IL}, A3 = V_{IL}, A6 = V_{IH}, A9 = V_{ID}, A12-A21 Block address Others = X	Pass = xx00h Retry = xx01h.

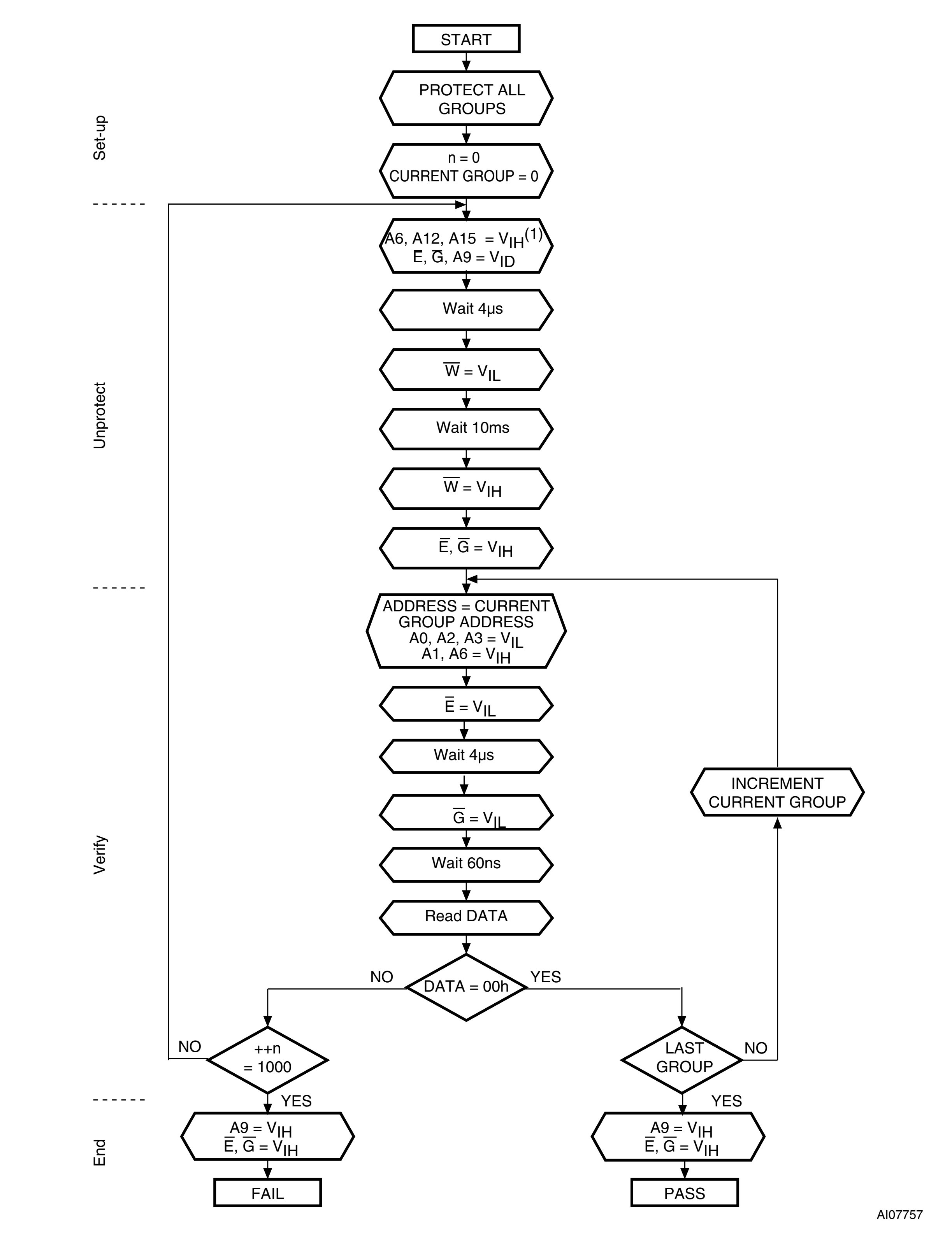
[1] Block Protection Groups are shown in *Appendix D*, *Table 24*.

Figure 20. Programmer Equipment Group Protect flowchart


Set-up
Protect
Verify
End
START
ADDRESS = GROUP ADDRESS
$\overline{W}$ = V_{IH}
n = 0
$\overline{G}$, A9 = V_{ID}, $\overline{E}$ = V_{IL}
Wait 4μs
$\overline{W}$ = V_{IL}
Wait 100μs
$\overline{W}$ = V_{IH}
$\overline{E}$, $\overline{G}$ = V_{IH}, A1 = V_{IH} A0, A2, A3, A6 = V_{IL}
$\overline{E}$ = V_{IL}
Wait 4μs
$\overline{G}$ = V_{IL}
Wait 60ns
Read DATA
DATA = 01h
NO
++n = 25
NO
YES
YES
A9 = V_{IH} $\overline{E}$, $\overline{G}$ = V_{IH}
A9 = V_{IH} $\overline{E}$, $\overline{G}$ = V_{IH}
PASS
FAIL
AI07756

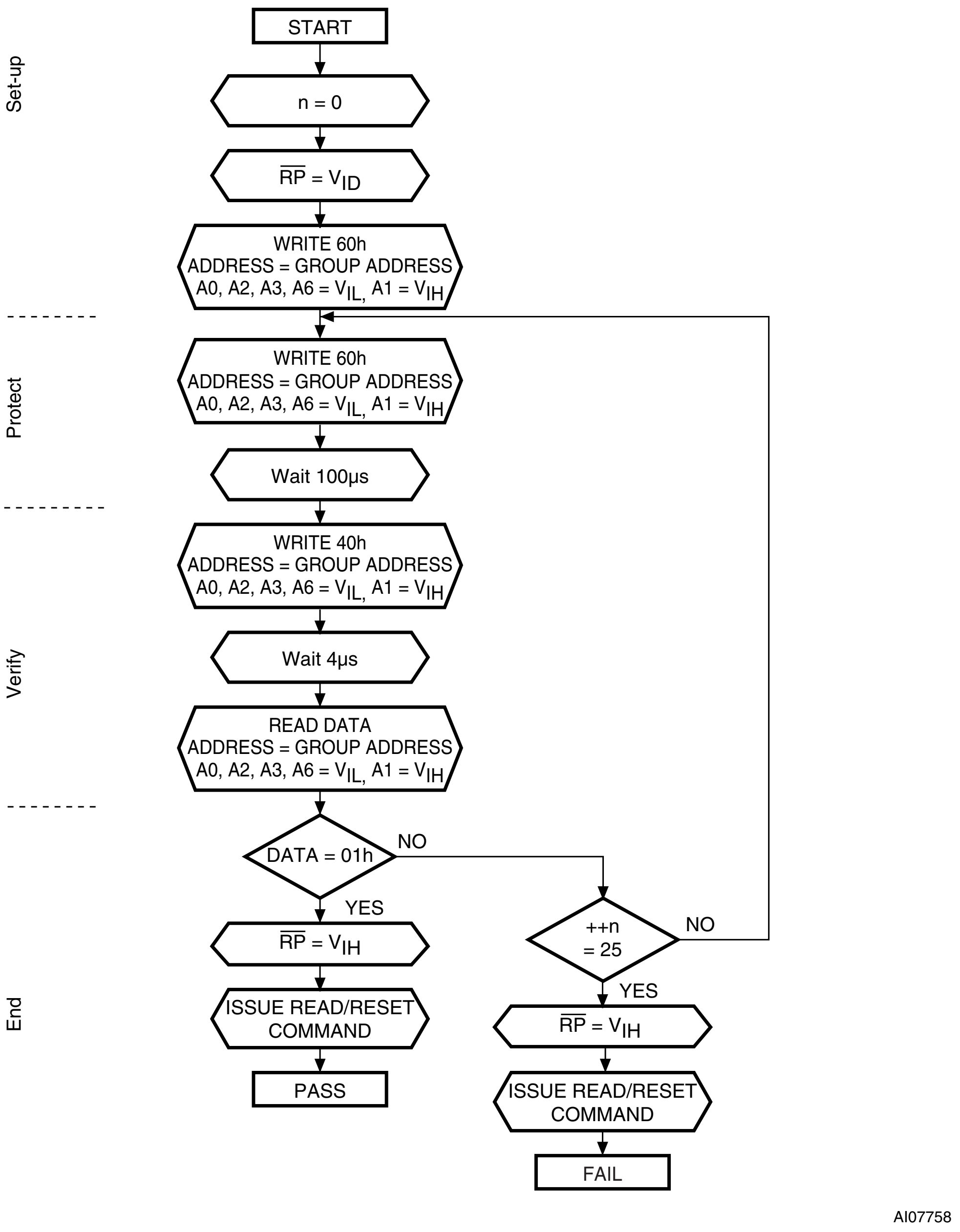
1. Block Protection Groups are shown in *Appendix D*, *Table 24*.

Figure 21. Programmer Equipment Chip Unprotect flowchart



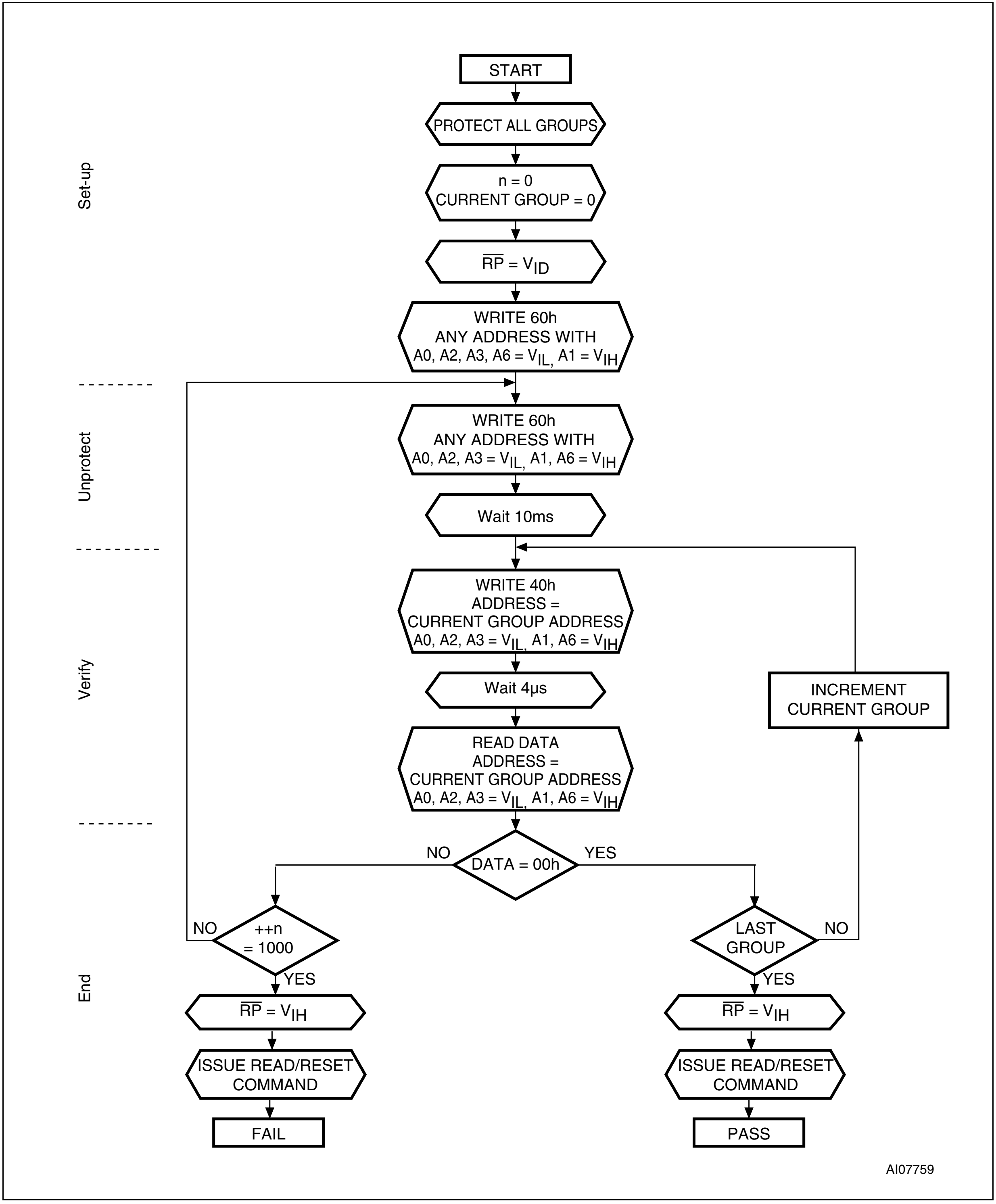
1. Block Protection Groups are shown in *Appendix D*, *Table 24*.

Figure 22. In-System Equipment Group Protect flowchart



1. Block Protection Groups are shown in *Appendix D*, *Table 24*.
2. $\overline{RP}$ can be either at V_{IH} or at V_{ID} when using the In-System Technique to protect the Extended Block.

Figure 23. In-System Equipment Chip Unprotect flowchart



1. Block Protection Groups are shown in *Appendix D*, *Table 24*.

Revision history

Table 33. Document revision history

Date	Revision	Changes
02-Dec-2005	1.0	First issue.
10-Mar-2006	2.0	DQ7 changed to $\overline{DQ7}$ for Program, Program During Erase Suspend and Program Error in *Table 9: Status Register Bits*. Converted to new template. Updated address values in *Table 31: Extended Block address and data*.
23-Aug-2006	3	Amended data in *Table 28: Device Geometry Definition*
10-Dec-2007	4	Applied Numonyx branding.

Please Read Carefully:

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH NUMONYX™ PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN NUMONYX'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, NUMONYX ASSUMES NO LIABILITY WHATSOEVER, AND NUMONYX DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF NUMONYX PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Numonyx products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Numonyx may make changes to specifications and product descriptions at any time, without notice.

Numonyx, B.V. may have patents or pending patent applications, trademarks, copyrights, or other intellectual property rights that relate to the presented subject matter. The furnishing of documents and other materials and information does not provide any license, express or implied, by estoppel or otherwise, to any such patents, trademarks, copyrights, or other intellectual property rights.

Designers must not rely on the absence or characteristics of any features or instructions marked “reserved” or “undefined.” Numonyx reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

Contact your local Numonyx sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Numonyx literature may be obtained by visiting Numonyx's website at http://www.numonyx.com.

Numonyx StrataFlash is a trademark or registered trademark of Numonyx or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others.