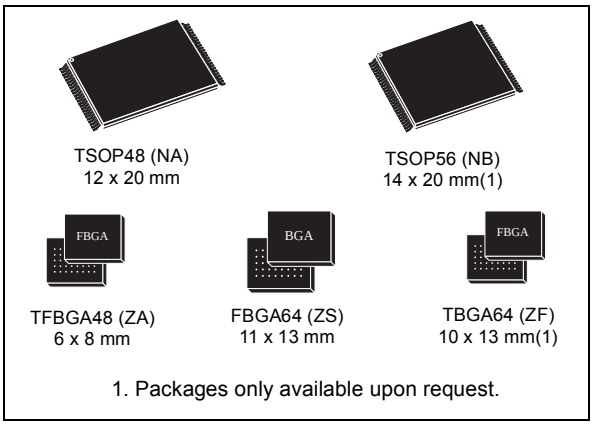
M29W640GH M29W640GL
M29W640GT M29W640GB

64-Mbit (8 Mbit x8 or 4 Mbit x16, uniform block or boot block) 3 V supply flash memory

Feature

- Supply voltage
  - V_{CC} = 2.7 to 3.6 V for program/erase/read
  - V_{PP} =12 V for fast program (optional)
- Asynchronous random/page read
  - Page width: 4 words
  - Page access: 25 ns
  - Random access: 60 ns, 70 ns, 90 ns
- Fast program commands
  - 2-word/4-byte program (without V_{PP}=12 V)
  - 4-word/8-byte program (with V_{PP}=12 V)
  - 16-word/32-byte write buffer
- Programming time
  - 10 µs per byte/word typical
  - Chip program time: 10 s (4-word program)
- Memory organization
  - M29W640GH/L:
    128 main blocks, 64 Kbytes each
  - M29W640GT/B
    Eight 8-Kbyte boot blocks (top or bottom)
    127 main blocks, 64 Kbytes each
- Program/erase controller
  - Embedded byte/word program algorithms
- Program/erase suspend and resume
  - Read from any block during program suspend
  - Read and program another block during erase suspend



1. Packages only available upon request.

- RoHS compliant packages
- 128-word extended memory block
- Low power consumption:standby and automatic standby
- Unlock Bypass Program command
  - Faster production/batch programming
- Common flash interface: 64-bit security code
- $V_{PP}/\overline{WP}$ pin for fast program and write protect
- Temporary block unprotection mode
- 100,000 program/erase cycles per block
- Electronic signature
  - Manufacturer code: 0020h
  - Device code (see *Table 1*)
- Automotive Certified Parts Available

Table 1. Device summary

Root part number	Device code
M29W640GH: uniform, last block protected by $V_{PP}/\overline{WP}$	227Eh + 220Ch + 2201h
M29W640GL: uniform, first block protected by $V_{PP}/\overline{WP}$	227Eh + 220Ch + 2200h
M29W640GT: top boot blocks	227Eh + 2210h + 2201h
M29W640GB: bottom boot blocks	227Eh + 2210h + 2200h

Contents

List of tables

List of figures

1 Description

The M29W640G is a 64-Mbit (8 Mbit x8 or 4 Mbit 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 memory is divided into blocks that can be erased independently so it is possible to preserve valid data while old data is erased. 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.

The M29W640GH and M29W640GL memory array is organized into 128 uniform blocks of 64 Kbytes each (or 32 Kwords each).

The M29W640GT and M29W640GB feature an asymmetric block architecture. The devices have an array of 135 blocks, divided into 8 parameter blocks of 8 Kbytes each (or 4 Kwords each), and 127 main blocks of 64 Kbytes each (or 32 Kwords each). The M29W640GT has the parameter blocks at the top of the memory address space while the M29W640GB locates the parameter blocks starting from the bottom.

Blocks are protected by groups to prevent accidental program or erase commands from modifying the memory.

- *Table 3*, describes the protection granularity on the M29W640GH and M29W640GL.
- *Table 4*, and *Table 5*. describe the protection granularity on the M29W640GT and M29W640GB.

The M29W640G support asynchronous random read and page read from all blocks of the memory array.

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 $V_{PP}/\overline{WP}$ signal is used to enable faster programming of the device. Protection from program/erase operation can be obtained by holding $V_{PP}/\overline{WP}$ to V_{SS}:

- On the M29W640GH and M29W640GL, the last and the first block is protected, respectively.
- On the M29W640GT and M29W640GB, the last two and the first two boot blocks are protected.

The devices feature a full set of fast program commands to improve the programming throughput:

- 2-byte program: it is not necessary to raise $V_{PP}/\overline{WP}$ to 12 V before issuing this command
- 2-words/4-byte program: it is not necessary to raise $V_{PP}/\overline{WP}$ to 12 V before issuing this command.
- 4-word/8-byte program: $V_{PP}/\overline{WP}$ must be raised to 12 V before issuing this command.
- write to buffer and program, allowing to program in one shot a buffer of 16 words/32 bytes.

The M29W640G has an extra block, the extended block, of 128 words in x16 mode or of 256 bytes in x8 mode 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 not reversible, once protected the protection cannot be undone.

The M29W640GT, M29W640GB, M29W640GH and M29W640GL, are offered in TSOP48 (12 x 20 mm), TSOP56 (14 x 20 mm), TFBGA48 (6 x 8 mm, 0.8 mm pitch), and TBGA64 (10 x 13 mm, 1 mm pitch) packages.

The memory is delivered with all the bits erased (set to '1').

Figure 1. Logic diagram

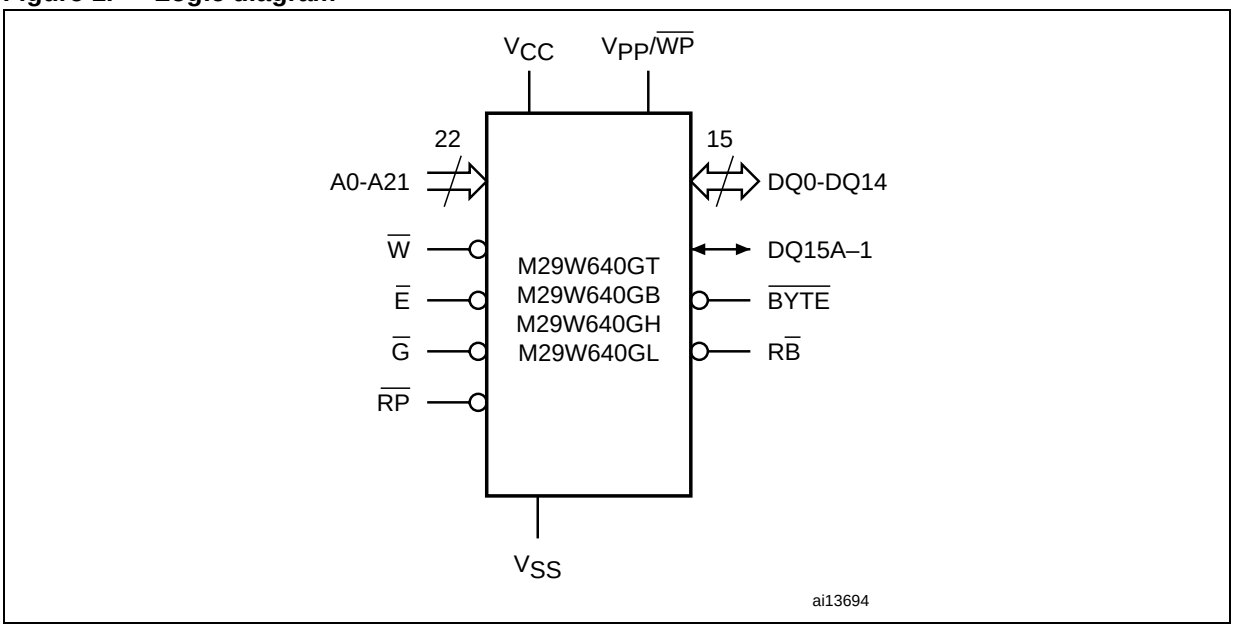


Table 2. Signal names(1)

Name	Description	Direction
A0-A21	Address Inputs	Inputs
DQ0-DQ7	Data Inputs/Outputs	Inputs/Outputs
DQ8-DQ14	Data Inputs/Outputs	Inputs/Outputs
DQ15A–1 (or DQ15)	Data Input/Output or Address Input (or Data Input/Output)	Inputs/Outputs
$\overline{E}$	Chip Enable	Input
$\overline{G}$	Output Enable	Input
$\overline{W}$	Write Enable	Input
$\overline{RP}$	Reset/Block Temporary Unprotect	Input
$R\overline{B}$	Ready/Busy	Output
$\overline{BYTE}$	Byte/Word Organization Select	Input
V_{CC}	Supply voltage	Supply voltage
$V_{PP}/\overline{WP}$	Supply voltage for fast program (optional) or write protect	Supply voltage
V_{SS}	Ground	–
NC	Not connected internally	–

1. $V_{PP}/\overline{WP}$ may be left floating since it is internally connected to an pull-up resistor to enable program/erase operations,

Table 3. Protection granularity on the M29W640GH and M29W640GL

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
0 to 3	4 x 64/32	Block level	000000h–03FFFFh(1)	000000h–01FFFFh(1)
4 to 7	4 x 64/32	Protection group	040000h–07FFFFh	020000h–03FFFFh
--	--	--	--	--
120 to 123	4 x 64/32	Protection group	780000h–7BFFFFh	3C0000h–3DFFFFh
124 to 127	4x 64/32	Block level	7C0000h–7FFFFFh	3E0000h–3FFFFFh

1. Used as the extended block addresses in extended block mode.

Table 4. Protection granularity on the M29W640GT

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
0 to 3	4 x 64/32	Protection group	000000h–03FFFFh(1)	000000h–01FFFFh(1)
4 to 7	4 x 64/32	Protection group	040000h–07FFFFh	020000h–03FFFFh
--	--	--	--	--
120 to 123	4 x 64/32	Protection group	780000h–7BFFFFh	3C0000h–3DFFFFh
124 to 126	3 x 64/32	Protection group	7C0000h–7EFFFFh	3E0000h–3F7FFFh
127 to 134	8x 8/4(2)	Block level	7F0000h–7FFFFFh	3F8000h–3FFFFFh

1. Used as the extended block addresses in extended block mode.
2. Boot blocks.

Table 5. Protection granularity on the M29W640GB

Block	Kbytes/Kwords	Protection Block Group	(x8)	(x16)
0 to 7	8x 8/4(1)	Block level	000000h–00FFFFh(2)	000000h–007FFFh(2)
8 to 10	3 x 64/32	Protection Group	010000h-03FFFFh	008000h-01FFFFh
11 to 14	4 x 64/32	Protection Group	040000h-07FFFFh	020000h-03FFFFh
--	--	--	--	--
127 to 130	4 x 64/32	Protection Group	780000h-7BFFFFh	3C0000h-3DFFFFh
131 to 134	4 x 64/32	Protection Group	7C0000h–7FFFFFh	3E0000h–3FFFFFh

1. Boot blocks.
2. Used as the extended block addresses in extended block mode.

Figure 2. TSOP48 connections

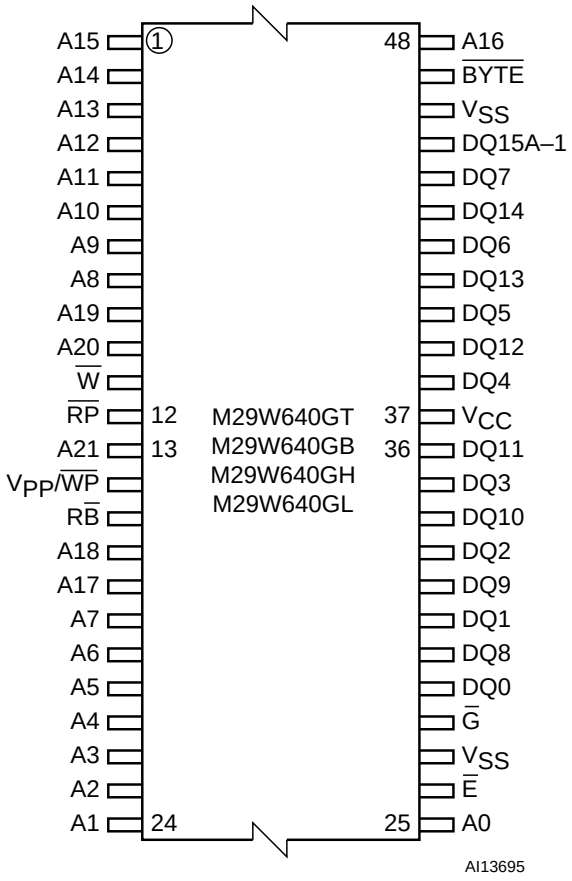


Figure 3. TSOP56 connections

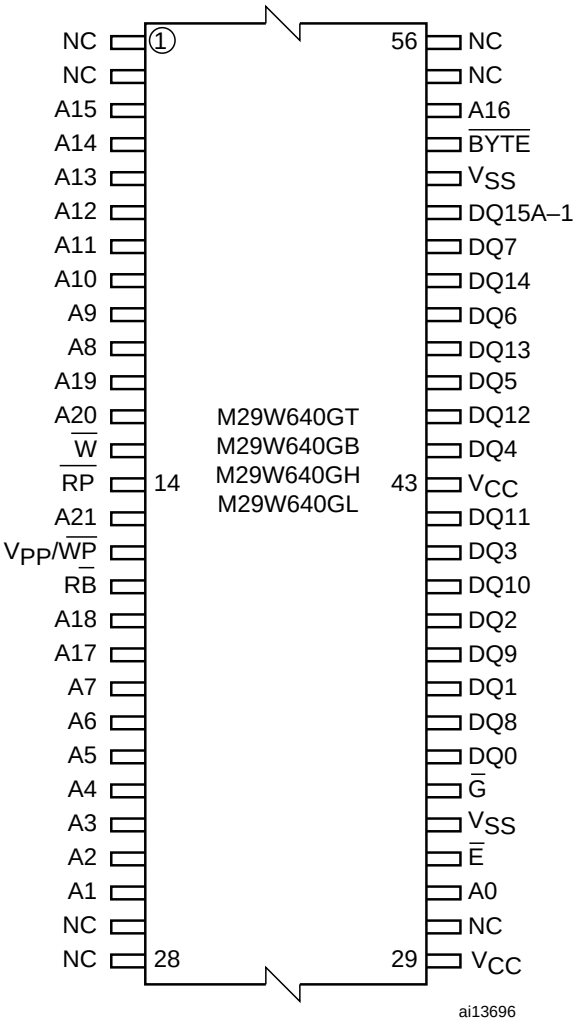


Figure 4. TFBGA48 connections (top view through package)

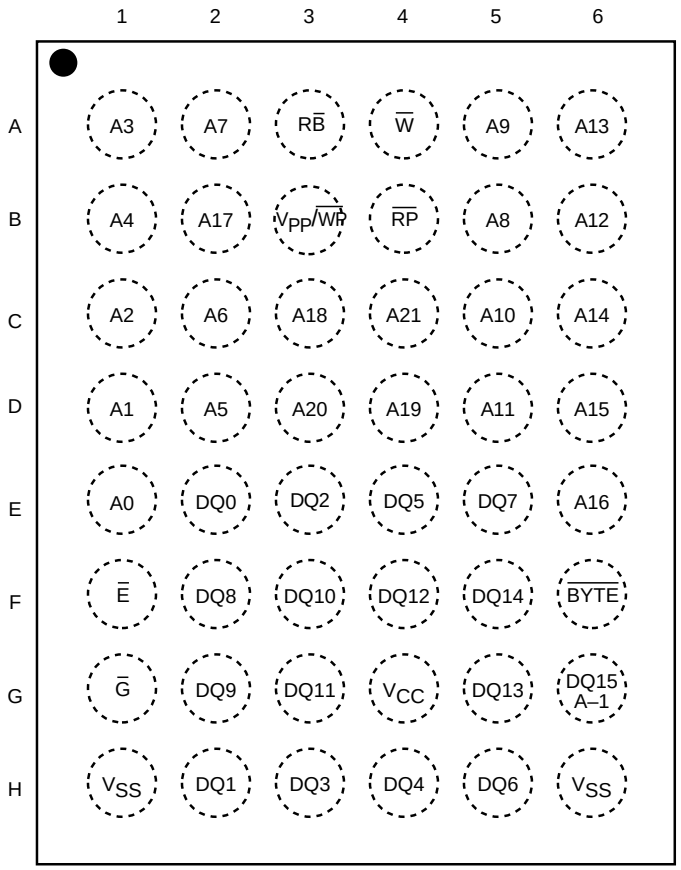
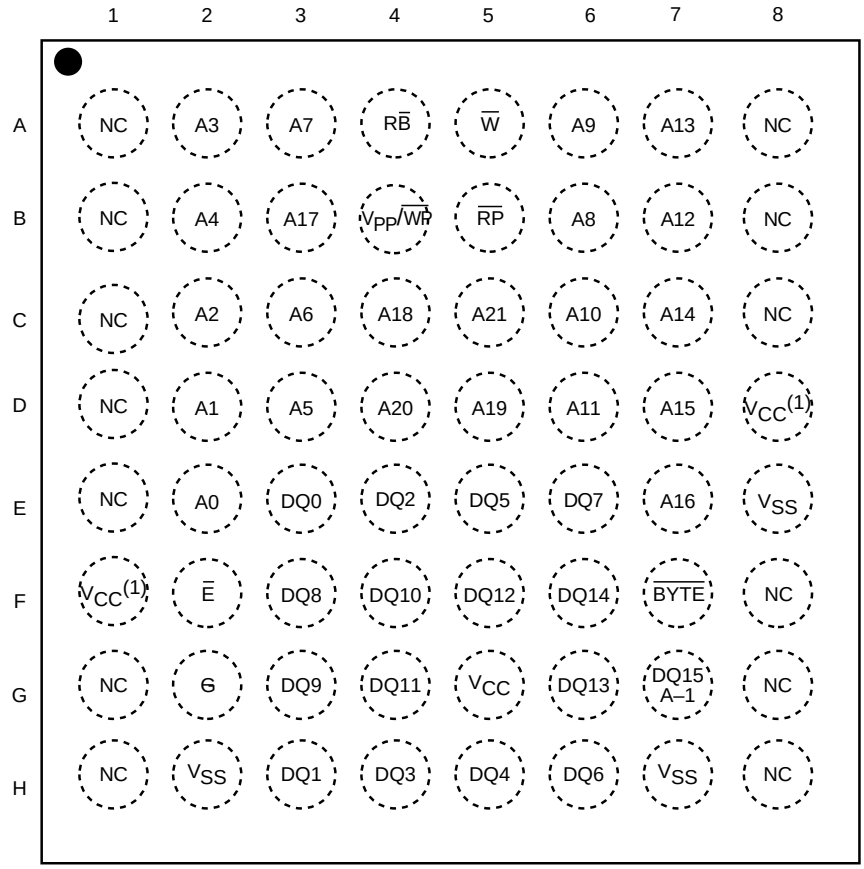


Figure 5. TBGA64 connections (top view through package)



1. Pads D8 and F1 are not connected (NC) on the M29W640GT and M29W640GB devices.

2 Signal descriptions

See *Figure 1: Logic diagram*, and *Table 2: Signal names*, for a brief overview of the signals connected to the 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 program/erase controller.

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 program/erase controller.

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 $\overline{BYTE}$ is High, V_{IH}. When $\overline{BYTE}$ 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 $\overline{BYTE}$ is High, V_{IH}, this pin behaves as a Data Input/Output pin (as DQ8-DQ14). When $\overline{BYTE}$ 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 $\overline{BYTE}$ is High and references to the Address Inputs to include this pin when $\overline{BYTE}$ is Low except when stated explicitly otherwise.

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

The Chip Enable, $\overline{E}$, 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 ($\overline{G}$)

The Output Enable, $\overline{G}$, 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 Unlock Bypass Program operations. The Write Protect function performs hardware protection:

- It protects the last block at the end of the addressable area (M29W640GH) or the first block at the beginning of the addressable area (M29W640GL)
- It protects the last two blocks at the end of the addressable area (M29W640GT) and the first two boot blocks at the beginning of the addressable area (M29W640GB).

The V_{PP}/Write Protect pin may be left floating or unconnected (see *Table 17: DC characteristics*).

When V_{PP}/Write Protect is Low, V_{IL}, the last or first block in the M29W640GH and M29W640GL, respectively, and the last or first two blocks in the M29W640GT and M29W640GB, respectively, are protected. Program and erase operations in this block 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 outermost blocks. Program and erase operations can now modify the data in the outermost blocks unless the block is protected using block protection.

Applying 12 V to the $V_{PP}/\overline{WP}$ pin will temporarily unprotect any block previously protected (including the outermost blocks) using a high voltage block protection technique (in-system or programmer technique). See *Table 6: 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 18: Accelerated program timing waveforms*.

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.

A 0.1 µF 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 6. Hardware protection

$V_{PP}/\overline{WP}$	$\overline{RP}$		Function
V_{IL}	V_{IH}	M29W640GT and M29W640GB	Last 2 blocks at the end of the addressable area (M29W640GT) and first 2 blocks at the beginning of the addressable area (M29W640GB) protected from program/erase operations
		M29W640GH and M29W640GL	Last block at the end of the addressable area (M29W640GH) and first block at the beginning of the addressable area (M29W640GL) protected from program/erase operations
	V_{ID}	M29W640GT and M29W640GB	All blocks temporarily unprotected except the last 2 blocks at the end of the addressable area (M29W640GT) and first 2 blocks at the beginning of the addressable area (M29W640GB)
		M29W640GH and M29W640GL	All blocks temporarily unprotected except the last block at the end of the addressable area (M29W640GH) and first block at the beginning of the addressable area (M29W640GL)
V_{IH} or V_{ID}	V_{ID}		All blocks temporarily unprotected
V_{PP}	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 last and the first block in the M29W640GH and M29W640GL, respectively, and the last two and first two blocks in the M29W640GT and M29W640GB, respectively, will remain protected even if $\overline{RP}$ 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 *Section 2.10: Ready/Busy Output (RB)*, *Table 20: Reset/Block Temporary Unprotect AC characteristics* and *Figure 17: Reset/Block Temporary Unprotect AC waveforms*, for more details.

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 ($\overline{RB}$)

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: Reset/Block Temporary Unprotect AC characteristics* and *Figure 17: Reset/Block Temporary Unprotect AC waveforms*, for more details.

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 0.1 µF 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 which must be both connected to the system ground.

3 Bus operations

There are five standard bus operations that control the device. These are bus read, bus write, output disable, standby and automatic standby. See *Table 7: Bus operations, BYTE = VIL* and *Table 8: Bus operations, BYTE = VIH*, for a summary. Typically glitches of less than 5 ns on Chip Enable or Write Enable 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. 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 13: Read mode AC waveforms (8-bit mode)*, and *Table 18: Read AC characteristics*, for details of when the output becomes valid.

3.2 Bus write

Bus write operations write to 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 4 words and is addressed by the address inputs A0-A1.

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 15: Write AC waveforms, write enable controlled (8-bit mode)*, *Figure 16: Write AC waveforms, chip enable controlled (8-bit mode)*, and *Table 19: 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} ± 0.2 V. For the standby current level see *Table 17: 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} ± 0.2 V) are used to drive the bus and the bus is inactive for 300 ns 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 7: Bus operations, BYTE = VIL* and *Table 8: Bus operations, BYTE = VIH*, with A9 set to V_{ID}.

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: Block addresses*, *Table 29* and *Table 30*. 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 blocks as described in *Table 6: Hardware protection*.

Block protect and chip unprotect operations are described in *Appendix D: Block protection*.

Table 7. Bus operations, $\overline{BYTE}$ = V_{IL} (1)

Operation	$\overline{E}$	$\overline{G}$	$\overline{W}$	Address Inputs DQ15A–1, A0-A21	Data Inputs/Outputs	
					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}	*Table 9*	Hi-Z	20h
Read device code (cycle 1)	V_{IL}	V_{IL}	V_{IH}		Hi-Z	7Eh
Read device code (cycle 2)						M29W640GH, M29W640GL: 0Ch
						M29W640GT, M29W640GB: 10h
Read device code (cycle 3)						M29W640GH, M29W640GT: 01h
						M29W640GL, M29W640GB: 00h
Read extended memory block verify code	V_{IL}	V_{IL}	V_{IH}		Hi-Z	M29W640GL, M29W640GT, M29W640GB: 88h (factory locked) 08h (customer lockable)
						M29W640GH: 98h(factory locked) 18h (customer lockable)
Read block protection status	V_{IL}	V_{IL}	V_{IH}		Hi-Z	01h (protected) 00h (unprotected)

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

Table 8. Bus operations, $\overline{BYTE}$ = V_{IH}(1)

Operation	$\overline{E}$	$\overline{G}$	$\overline{W}$	Address Inputs A0-A21	Data Inputs/Outputs 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}	Table 9	0020h	
Read device code (cycle 1)	V_{IL}	V_{IL}	V_{IH}		227Eh	
Read device code (cycle 2)					M29W640GH, M29W640GL	220Ch
					M29W640GT, M29W640GB	2210h
Read device code (cycle 3)					M29W640GH, M29W640GT	2201h
					M29W640GL, M29W640GB	2200h
Read extended memory block verify code	V_{IL}	V_{IL}	V_{IH}		M29W640GL, M29W640GT, M29W640GB	2288h (factory locked) 2208h (customer lockable)
					M29W640GH	2298h(factory locked) 2218h (customer lockable)
Read block protection status	V_{IL}	V_{IL}	V_{IH}		0001h (protected) 0000h (unprotected)	

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

Table 9. Read electronic signature addresses(1)

Code	A7-A0 $\overline{BYTE}$ = V_{IH}	A6-A0, DQ15A–1 $\overline{BYTE}$ = V_{IL}
Manufacturer code	00h	00h
Device code (cycle 1)	01h	02h
Device code (cycle 2)	0Eh	1Ch
Device code (cycle 3)	0Fh	1Eh
Extended memory block verify code	03h	06h
Block protection status	02h(2)	04h(2)

1. A9 = V_{ID}; other address bits set to V_{IL} or V_{IH}.
2. A12- A21 must be set to the block address.

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 10* or *Table 11*, 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 10 µs 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, the device code, the block protection status and the extended memory block verify code. Three consecutive bus write operations are required to issue the Auto Select command. Once the Auto Select command is issued the memory remains in auto select mode until a Read/Reset command is issued. Read CFI Query and Read/Reset commands are accepted in auto select mode, all other commands are ignored.

In auto select mode, the manufacturer code and the device code can be read by using a bus read operation with addresses and control signals set as shown in *Table 7: Bus operations, BYTE = VIL* and *Table 8: Bus operations, BYTE = VIH*, except for A9 that is ‘don’t care’.

The block protection status of each block can be read using a bus read operation with addresses and control signals set as shown in *Table 7: Bus operations, BYTE = VIL* and *Table 8: Bus operations, BYTE = VIH*, except for A9 that is ‘don’t care’. If the addressed block is protected then 01h is output on Data Inputs/Outputs DQ0-DQ7, otherwise 00h is output (in 8-bit mode).

The protection status of the extended memory block, or extended memory block verify code, can be read using a bus read operation with addresses and control signals set as shown in *Table 7: Bus operations, BYTE = VIL* and *Table 8: Bus operations, BYTE = VIH*, except for A9 that is ‘don’t care’. If the extended block is ‘factory locked’ then 80h is output on Data Input/Outputs DQ0-DQ7, otherwise 00h is output (8-bit mode).

4.1.3 Read CFI Query command

The Read CFI Query command is used to read data from the common flash interface (CFI) memory area. This command is valid when the device is in the read array mode, or when the device is in autoselected mode.

One bus write cycle is required to issue the Read CFI Query command. Once the command is issued subsequent bus read operations read from the common flash interface memory area.

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 auto selected mode.

See *Appendix B: Common flash interface (CFI)*, Tables *31*, *32*, *33*, *34*, *35* and *36* 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 100 µs, 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 12: Program, erase times and endurance cycles*. 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.

Refer to *Figure 8: Chip/block erase waveforms (8-bit mode)* for a description of Chip Erase AC waveforms.

4.1.5 Block Erase command

The Block Erase command can be used to erase a list of one or more blocks. 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 about 50 µs 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 50 µs of the last block. The 50 µs timer restarts when an additional block is selected. The status register can be read after the sixth bus write operation. 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 100 µs, 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. Typical block erase times are given in *Table 12: Program, erase times and endurance cycles*. All bus read operations during the block erase operation will output the status register on the Data Inputs/Outputs. See the *Section 5: 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 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 Block Erase command sets all of the bits in the unprotected selected blocks to ’1’. All previous data in the selected blocks is lost.

Refer to *Figure 8: Chip/block erase waveforms (8-bit mode)* for a description of Block Erase AC waveforms.

4.1.6 Erase Suspend command

The Erase Suspend command may be used to temporarily suspend a block erase operation and return the memory to read mode. The command requires one bus write operation.

The program/erase controller will suspend within the erase suspend latency time 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.

4.1.7 Erase Resume command

The Erase Resume command must be used to restart the program/erase controller after an Erase Suspend. 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 12: Program, erase times and endurance cycles* for value) and updates the status register bits.

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, 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, 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.

During the program operation the memory will ignore all commands. It is not possible to issue any command to abort or pause the operation. Typical program times are given in *Table 12: Program, erase times and endurance cycles*. Bus read operations during the program operation will output the status register on the Data Inputs/Outputs. See the section on the status register for more details.

After the program 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.

Note that the Program command cannot change a bit set to ’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’.

Refer to *Figure 6: Write enable controlled program waveforms (8-bit mode)* and *Figure 7: Chip enable controlled program waveforms (8-bit mode)* for a description of Program AC waveforms.

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:

- Quadruple and Octuple Byte Program, available for x8 operations
- Double and Quadruple Word Program, available for x16 operations
- Write to Buffer and Program

Fast program commands 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*).

4.2.1 Double Byte Program command

The Double Byte Program command is used to write a page of two adjacent bytes in parallel. The two bytes must differ only in DQ15A-1. Three bus write cycles are necessary to issue the Double Byte Program command:

1. The first bus cycle sets up the Double Byte Program 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.

Note: It is not necessary to raise $V_{PP}/\overline{WP}$ to 12 V before issuing this command.

4.2.2 Quadruple Byte Program command

The Quadruple Byte Program command is used to write a page of four adjacent bytes in parallel. The four bytes must differ only for addresses A0, DQ15A-1. Five bus write cycles are necessary to issue the Quadruple Byte Program command:

1. The first bus cycle sets up the Quadruple Byte Program 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.

Note: It is not necessary to raise $V_{PP}/\overline{WP}$ to 12 V before issuing this command.

4.2.3 Octuple Byte Program command

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

12 V must be applied to the V_{PP}/Write Protect pin, $V_{PP}/\overline{WP}$, prior to issuing an Octuple Byte Program command. Care must be taken because applying a 12 V voltage to the $V_{PP}/\overline{WP}$ pin will temporarily unprotect any protected block.

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.

4.2.4 Double Word Program command

The Double Word Program command is used to write a page of two adjacent words in parallel. The two words must differ only for the address A0.

Three bus write cycles are necessary to issue the Double Word Program command:

- The first bus cycle sets up the Double Word Program command
- The second bus cycle latches the address and the data of the first word to be written
- The third bus cycle latches the address and the data of the second word to be written and starts the program/erase controller.

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 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 to ’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 12: Program, erase times and endurance cycles*.

Note: It is not necessary to raise $V_{PP}/\overline{WP}$ to 12 V before issuing this command.

4.2.5 Quadruple Word Program command

This is used to write a page of four adjacent words (or 8 adjacent bytes), in x16 mode, simultaneously. The addresses of the four words must differ only in A1 and A0.

12 V must be applied to the V_{PP}/Write Protect pin, $V_{PP}/\overline{WP}$, prior to issuing a Quadruple Byte Program command. Care must be taken because applying a 12 V voltage to the $V_{PP}/\overline{WP}$ pin will temporarily unprotect any protected block.

Five bus write cycles are necessary to issue the command:

- The first bus cycle sets up the command.
- The second bus cycle latches the address and the data of the first word to be written
- The third bus cycle latches the address and the data of the second word to be written
- The fourth bus cycle latches the address and the data of the third word to be written
- The fifth bus cycle latches the address and the data of the fourth word to be written and starts the program/erase controller.

4.2.6 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 memory will only accept the Unlock Bypass Program command and the Unlock Bypass Reset command. 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 command is used in conjunction with the Unlock Bypass Program command to program the memory. 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 memory will only accept the Unlock Bypass Program command and the Unlock Bypass Reset command. The memory can be read as if in read mode.

The memory offers accelerated program operations through the V_{PP}/Write Protect pin. When the system asserts V_{PP} on the V_{PP}/Write Protect pin, the memory automatically enters the unlock bypass mode. The system may then write the two-cycle unlock bypass program command sequence. The memory uses the higher voltage on the V_{PP}/Write Protect pin, to accelerate the unlock bypass program operation.

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.

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.

4.2.9 Write to Buffer and Program command

The Write to Buffer and Program command makes use of the device's 32-byte write buffer to speed up programming. 16 words/32 bytes can be loaded into the write buffer. Each write buffer has the same A4-A22 addresses.The Write to Buffer and Program command dramatically reduces system programming time compared to the standard non-buffered Program command.

When issuing a Write to Buffer and Program command, the $V_{PP}/\overline{WP}$ pin can be either held High, V_{IH} or raised to V_{PPH}.

See *Table 12* for details on typical Write to Buffer and Program times in both cases.

Five successive steps are required to issue the Write to Buffer and Program command:

1. The Write to Buffer and Program command starts with two unlock cycles
2. The third bus write cycle sets up the Write to Buffer and Program command. The setup code can be addressed to any location within the targeted block.
3. The fourth bus write cycle sets up the number of words to be programmed. Value n is written to the same block address, where n+1 is the number of words to be programmed. n+1 must not exceed the size of the write buffer or the operation will abort
4. The fifth cycle loads the first address and data to be programmed
5. Use n bus write cycles to load the address and data for each word into the write buffer. Addresses must lie within the range from the start address+1 to the start address + n-1. Optimum performance is obtained when the start address corresponds to a 64-byte boundary. If the start address is not aligned to a 64-byte boundary, the total programming time is doubled.

All the addresses used in the Write to Buffer and Program operation must lie within the same page. If an address is written several times during a Write to Buffer and Program operation, the address/data counter will be decremented at each data load operation and the data will be programmed to the last word loaded into the buffer. Invalid address combinations or failing to follow the correct sequence of bus write cycles will abort the Write to Buffer and Program.

The status register bits DQ1, DQ5, DQ6, DQ7 can be used to monitor the device status during a Write to Buffer and Program operation. It is possible to detect program operation fails when changing programmed data from '0' to '1', that is when reprogramming data in a portion of memory already programmed. The resulting data will be the logical OR between the previous value and the current value.

To program the content of the write buffer, this command must be followed by a Write to Buffer and Program Confirm command.

A Write to Buffer and Program Abort and Reset command must be issued to abort the Write to Buffer and Program operation and reset the device in read mode.

The write buffer programming sequence can be aborted in the following ways:

- Load a value that is greater than the page buffer size during the number of locations to program step
- Write to an address in a block different than the one specified during the Write-Buffer-Load command
- Write an address/data pair to a different write-buffer-page than the one selected by the starting address during the write buffer data loading stage of the operation
- Write data other than the Confirm command after the specified number of data load cycles.

The abort condition is indicated by DQ1 = 1, DQ7 = DATA# (for the last address location loaded), DQ6 = toggle, and DQ5=0. A Write-to-Buffer-Abort Reset command sequence must be written to reset the device for the next operation. Note that the full 3-cycle Write-to-Buffer-Abort Reset command sequence is required when using write-buffer-programming features in unlock bypass mode.

See *Appendix E: Flowchart*, *Figure 30: Write to Buffer and Program flowchart and pseudocode*, for a suggested flowchart on using the Write to Buffer and Program command.

4.2.10 Write to Buffer and Program Confirm command

The Write to Buffer and Program Confirm command is used to confirm a Write to Buffer and Program command and to program the n+1 words loaded in the write buffer by this command.

4.2.11 Write to Buffer and Program Abort and Reset command

The Write to Buffer and Program Abort and Reset command is used to reset the device after a Write to Buffer and Program command has been aborted.

4.3 Block protection commands

4.3.1 Enter Extended Block command

The device has an 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 write operations to the boot block addresses access the extended block. The extended block (with the same address as the boot blocks) cannot be erased, and can be treated as one-time programmable (OTP) memory. In extended block mode the boot blocks are not accessible.

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: Block addresses*, *Table 29: Top boot block addresses, M29W640GT* and *Table 30: Bottom boot block addresses, M29W640GB*. 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: Block protection*.

Table 10. Commands, 16-bit mode, $\overline{BYTE}$ = V_{IH}(1)

Command	Length	Bus write operations											
		1st		2nd		3rd		4th		5th		6th	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read/Reset	1	X	F0										
	3	555	AA	2AA	55	X	F0						
Auto Select	3	555	AA	2AA	55	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	5	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								
Write to Buffer and Program	N+5	555	AA	2AA	55	BA	25	BA	N(2)	PA(3)	PD	WBL (4)	PD
Write to Buffer and Program Abort and Reset	3	555	AA	2AA	55	555	F0						
Write to Buffer and Program Confirm	1	BA(5)	29										
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
Program/Erase Suspend	1	X	B0										
Program/Erase Resume	1	X	30										
Read CFI Query	1	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. All values in the table are in hexadecimal.
The command interface only uses A–1, A0-A10 and DQ0-DQ7 to verify the commands; A11-A20, DQ8-DQ14 and DQ15 are don't care. DQ15A–1 is A–1 when BYTE is V_{IL} or DQ15 when BYTE is V_{IH}.
2. The maximum number of cycles in the command sequence is 36. N+1 is the number of words to be programmed during the write to buffer and program operation.
3. Each buffer has the same A4-A22 addresses. A0-A3 are used to select a word within the N+1 word page.
4. The 6th cycle has to be issued N time. WBL scans the word inside the page.
5. BA must be identical to the address loaded during the Write to buffer and Program 3rd and 4th cycles.

Table 11. Commands, 8-bit mode, $\overline{BYTE}$ = 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	AAA	90												
Program	4	AAA	AA	555	55	AAA	A0	PA	PD										
Double Byte Program	3	AAA	50	PA0	PD0	PA1	PD1												
Quadruple Byte Program	5	AAA	56	PA0	PD0	PA1	PD1	PA2	PD2	PA3	PD3								
Octuple Byte Program	9	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														
Write to Buffer and Program	N+5	AAA	AA	555	55	BA	25	BA	N[2]	PA (3)	PD	WBL (4)	PD						
Write to Buffer and Program Abort and Reset	3	AAA	AA	555	55	AAA	F0												
Write to Buffer and Program Confirm	1	BA (5)	29																
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						
Program/Erase Suspend	1	X	B0																
Program/Erase Resume	1	X	30																
Read CFI Query	1	AA	98																
Enter Extended Block	3	AAA	AA	555	55	AAA	88												
Exit Extended Block	4	AAA	AA	555	55	AAA	90	X	00										

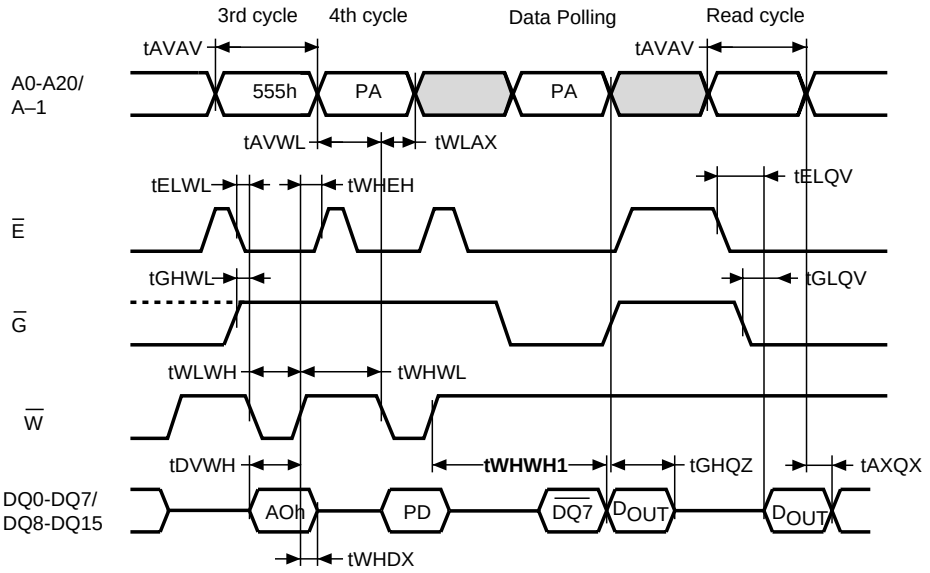
1. X Don't Care, PA Program Address, PD Program Data, BA Any address in the Block. All values in the table are in hexadecimal.
   The command interface only uses A–1, A0-A10 and DQ0-DQ7 to verify the commands; A11-A20, DQ8-DQ14 and DQ15 are don't care. DQ15A–1 is A–1 when $\overline{BYTE}$ is V_{IL} or DQ15 when $\overline{BYTE}$ is V_{IH}.
2. The maximum number of cycles in the command sequence is 68. N+1 is the number of bytes to be programmed during the write to buffer and program operation.
3. Each buffer has the same A4-A22 addresses. A0-A3 and DQ15A-1 are used to select a byte within the N+1 byte page.
4. The 6th cycle has to be issued N time. WBL scans the byte inside the page.
5. BA must be identical to the address loaded during the Write to buffer and Program 3rd and 4th cycles.

Table 12. Program, erase times and endurance cycles

Parameter	Symbol	Min	Typ(1)(2)	Max(2)	Unit
Chip Erase			80	400(3)	s
Block Erase (64 Kbytes)(4)(5)	t_{WHWH2}		0.5		s
Erase Suspend Latency Time				50(6)	µs
Program (byte or word)			10	200(3)	µs
Double Byte			10	200(3)	µs
Double Word /Quadruple Byte Program			10	200(3)	µs
Quadruple Word / Octuple Byte Program			10	200(3)	µs
Single Byte and Word Program(7)	t_{WHWH1}		10		µs
32-byte/16-word Program using Write to Buffer and Program			180		µs
32-byte/16-word Program using Write to Buffer and Program ($V_{PP}/\overline{WP}$ = 12 V)			45		µs
Chip Program (byte by byte)			80	400(3)	s
Chip Program (word by word)			40	200(3)	s
Chip Program (Double Word/Quadruple Byte Program)			20	100(3)	s
Chip Program (Quadruple Word/Octuple Byte Program)			10	50(3)	s
Program Suspend Latency Time				4	µs
Program/Erase Cycles (per block)		100,000			cycles
Data Retention		20			years

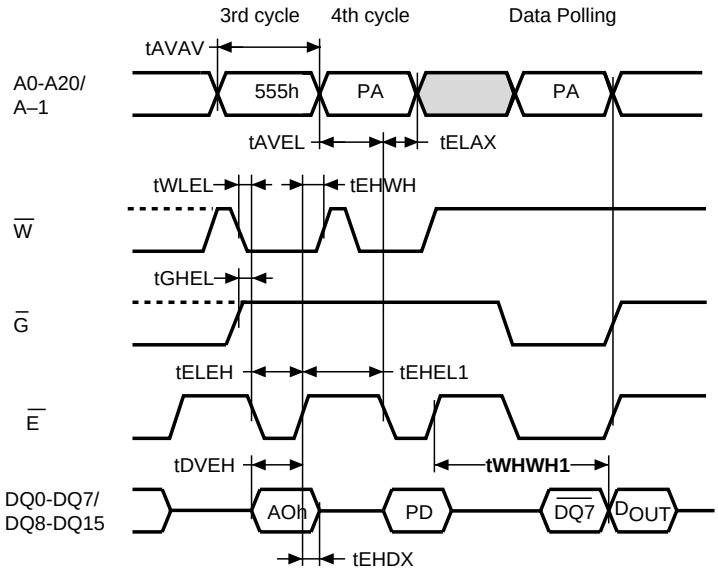
1. Typical values measured at room temperature and nominal voltages.
2. Sampled, but not 100% tested.
3. Maximum value measured at worst case conditions for both temperature and V_{CC} after 100,000 program/erase cycles.
4. This time does not include the pre-programming time.
5. Block erase polling cycle time (see *Figure 19*).
6. Maximum value measured at worst case conditions for both temperature and V_{CC}.
7. Program polling cycle time (see *Figure 6*, *Figure 7* and *Figure 19*).

Figure 6. Write enable controlled program waveforms (8-bit mode)



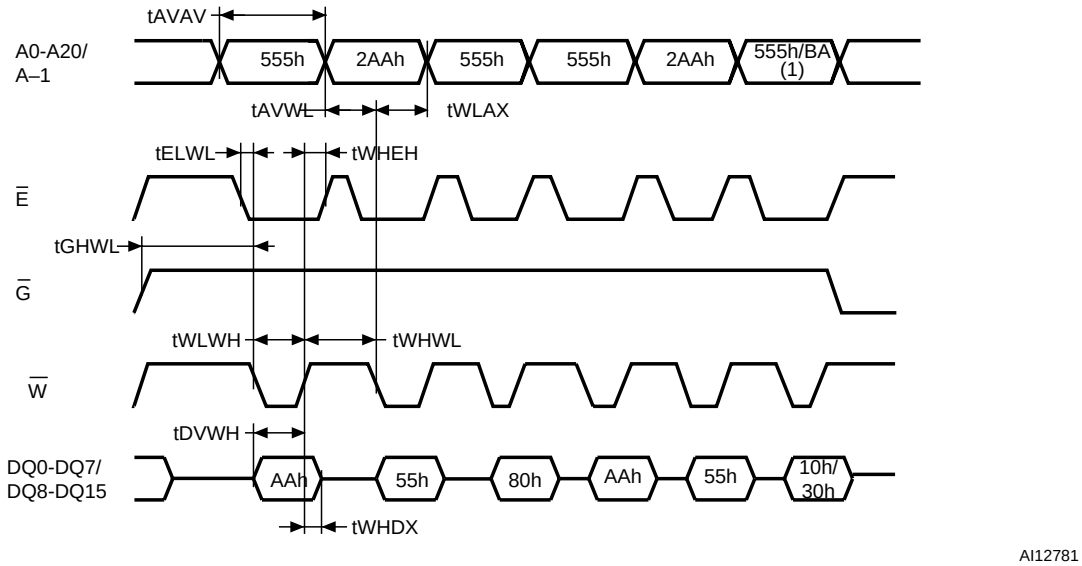
1. Only the third and fourth cycles of the Program command are represented. The Program command is followed by the check of status register data polling bit and by a read operation that outputs the data, D_{OUT}, programmed by the previous Program command.
2. PA is address of the memory location to be programmed. PD is the data to be programmed.
3. $\overline{DQ7}$ is the complement of the data bit being programmed to DQ7 (see *Section 5.1: Data polling bit (DQ7)*).
4. Addresses differ in x8 mode.
5. See *Table 19: Write AC characteristics* and *Table 18: Read AC characteristics* for details on the timings.

Figure 7. Chip enable controlled program waveforms (8-bit mode)



1. Only the third and fourth cycles of the Program command are represented. The Program command is followed by the check of status register data polling bit.
2. PA is address of the memory location to be programmed. PD is the data to be programmed.
3. $\overline{DQ7}$ is the complement of the data bit being programmed to DQ7 (see *Section 5.1: Data polling bit (DQ7)*).
4. Addresses differ in x8 mode.
5. See *Table 19: Write AC characteristics* and *Table 18: Read AC characteristics* for details on the timings.

Figure 8. Chip/block erase waveforms (8-bit mode)



1. For a Chip Erase command, addresses and data are 555h and 10h, respectively, while they are BA and 30h for a Block Erase command.
2. BA is the block address.
3. See *Table 19: Write AC characteristics* and *Table 18: Read AC characteristics* for details on the timings.

5 Status register

Bus read operations from any address 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 13: 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 9: 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. *Table 20: Reset/Block Temporary Unprotect AC characteristics* gives a description of the data polling operation and timings.

5.2 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 10: Data toggle flowchart, gives an example of how to use the data toggle bit.

Figure 20: Toggle/alternative toggle bit polling AC waveforms (8-bit mode) gives a description of the data polling operation and timings.

5.3 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’.

5.4 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.

5.5 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 20: Toggle/alternative toggle bit polling AC waveforms (8-bit mode) gives a description of the data polling operation and timings.

5.6 Write to buffer and program abort bit (DQ1)

The write to buffer and program abort bit, DQ1, is set to ‘1’ when a write to buffer and program operation aborts. Otherwise, DQ1 bit is set to ‘0’. The Write to Buffer and Program Abort and Reset command must be issued to return the device to read mode (see Write to Buffer and Program in the commands section).

Table 13. Status register bits(1)

Operation	Address	DQ7	DQ6	DQ5	DQ3	DQ2	DQ1	$\overline{RB}$
Program	Any address	$\overline{DQ7}$	Toggle	0	–	–	0	0
Program During Erase Suspend	Any address	$\overline{DQ7}$	Toggle	0	–	–	–	0
Write to Buffer and Program Abort	Any address	$\overline{DQ7}$	Toggle	0	–	–	1	0
Write to Buffer and Program	Any address	$\overline{DQ7}$	Toggle	0	–	–	0	0
Program Error	Any address	$\overline{DQ7}$	Toggle	1	–	–	–	Hi-Z
Chip Erase	Any address	0	Toggle	0	1	Toggle	–	0
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	–	0
	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	–	Hi-Z
	Faulty block address	0	Toggle	1	1	Toggle	–	Hi-Z

1. Unspecified data bits should be ignored.

Figure 9. Data polling flowchart

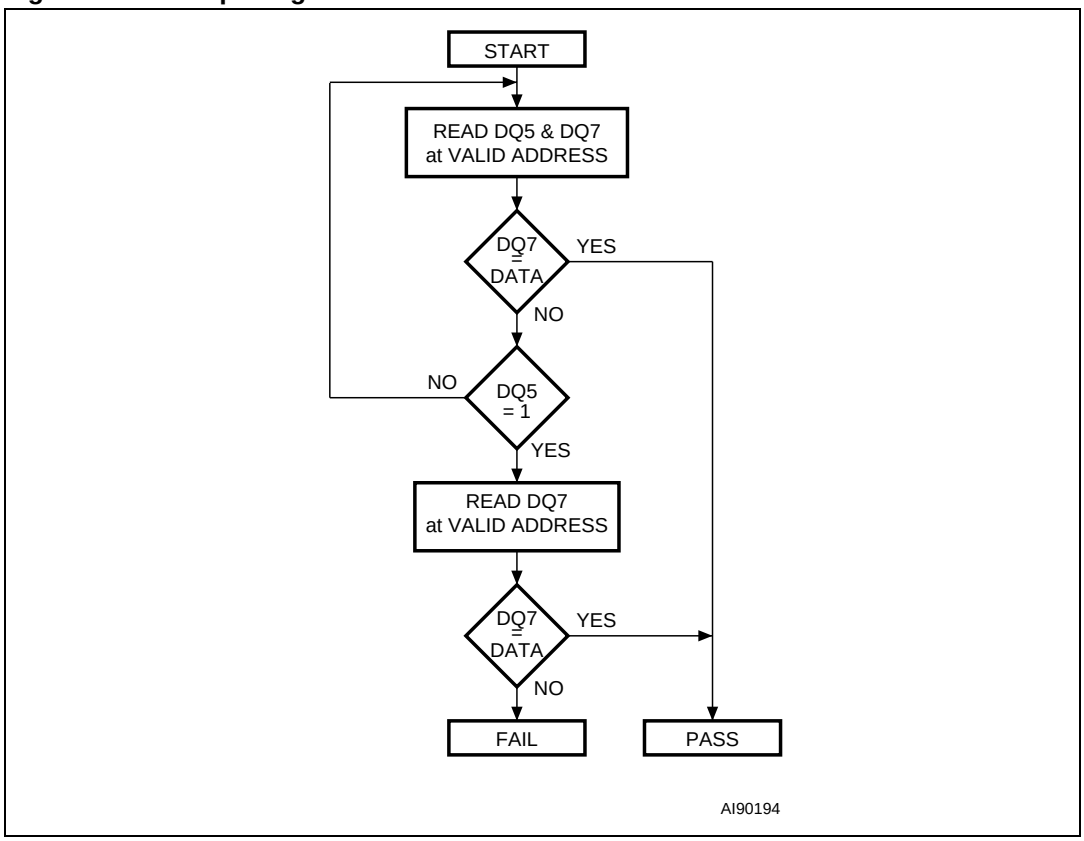
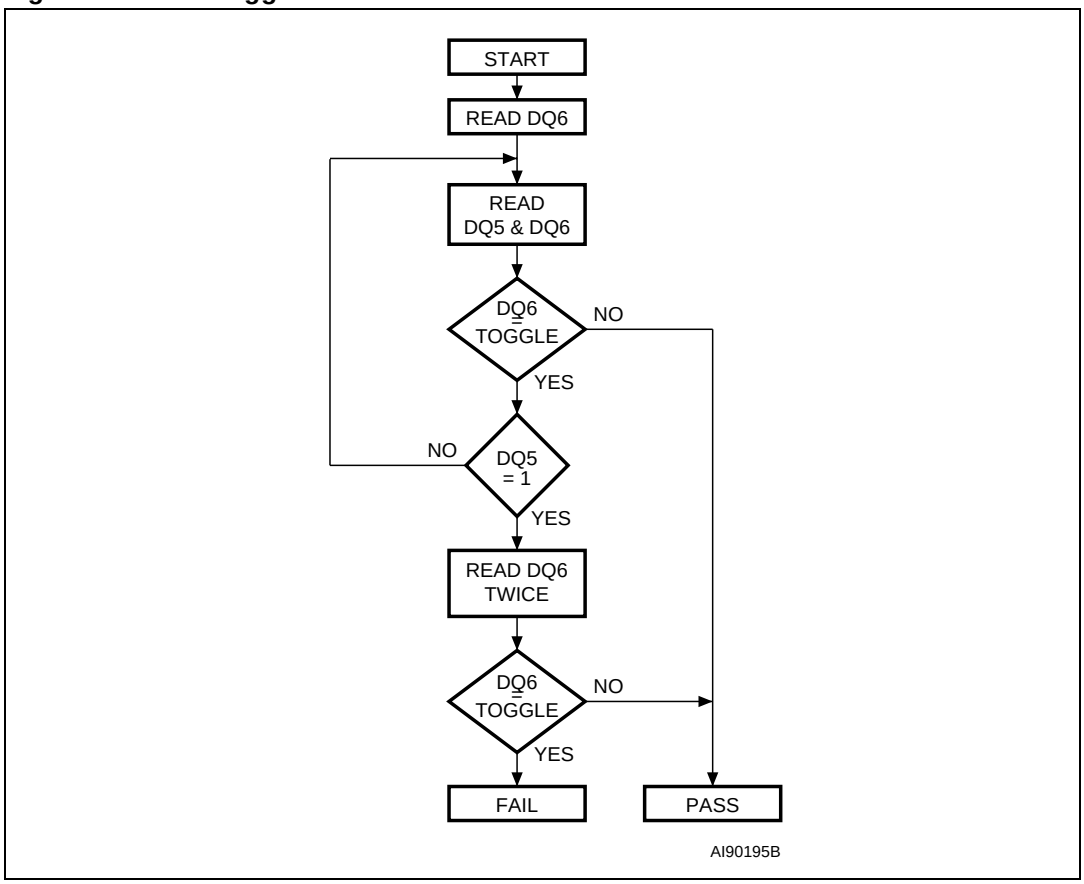


Figure 10. Data toggle flowchart



6 Maximum ratings

Stressing the device above the rating listed in the *Table 14: Absolute maximum ratings* 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.

Table 14. 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 –2 V during transition and for less than 20 ns during transitions.
2. Maximum voltage may overshoot to V_{CC} +2 V during transition and for less than 20 ns during transitions.
3. V_{PP} must not remain at 12 V for more than a total of 80 hrs.

7 DC and AC parameters

This section summarizes the operating and measurement conditions, and the DC and AC characteristics of the device. The parameters in the DC and AC characteristic tables that follow are derived from tests performed under the measurement conditions summarized in the relevant tables. Designers should check that the operating conditions in their circuit match the measurement conditions when relying on the quoted parameters.

Table 15. Operating and AC measurement conditions

Parameter	M29W640GT, M29W640GB, M29W640GH, M29W640GL		Unit
	Min	Max	
V_{CC} supply voltage	2.7	3.6	V
Ambient operating temperature	–40	85	°C
Load capacitance (C_L)	30		pF
Input rise and fall times		10	ns
Input pulse voltages	0 to V_{CC}		V
Input and output timing ref. voltages	$V_{CC}/2$		V

Figure 11. AC measurement I/O waveform

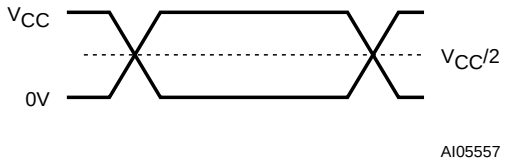


Figure 12. AC measurement load circuit

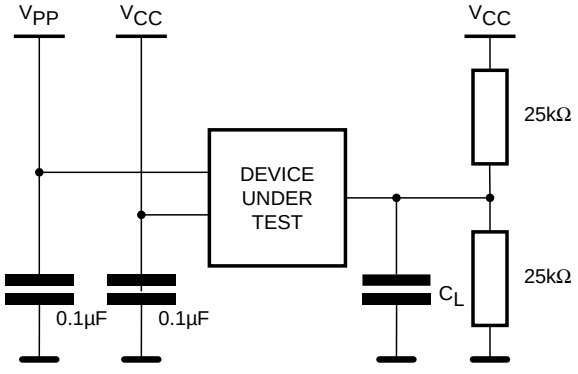


Table 16. Device capacitance

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

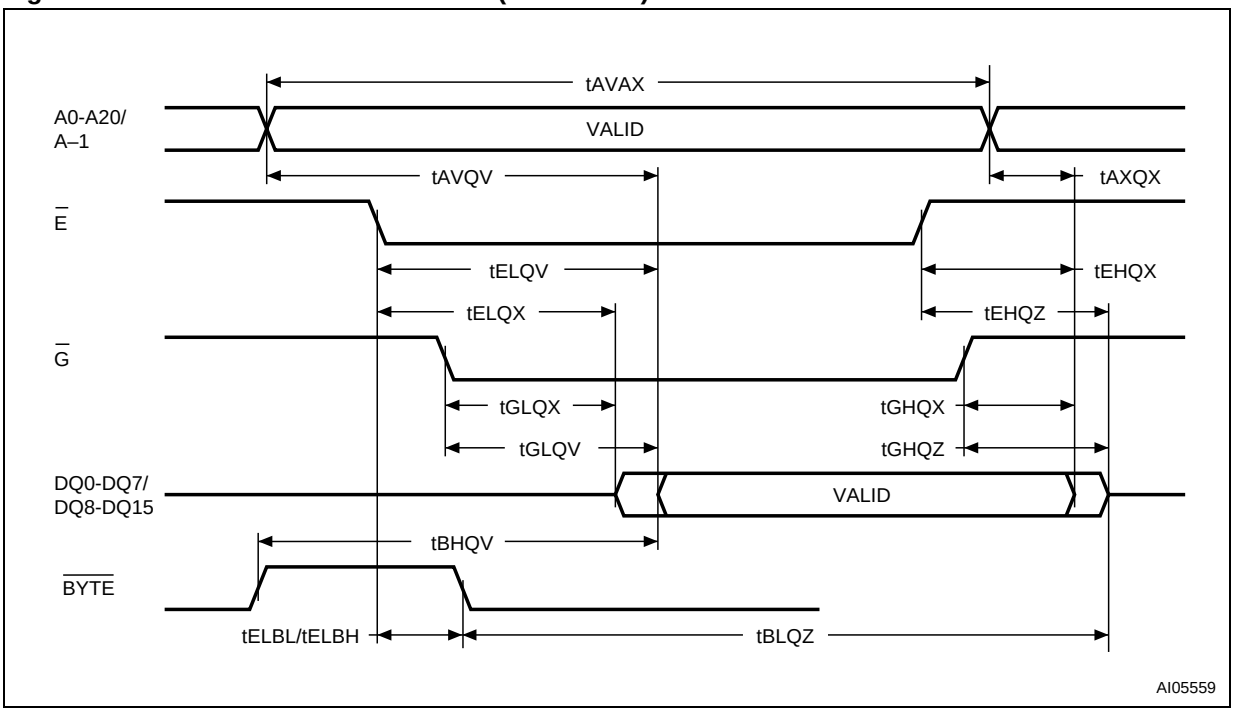
1. Sampled only, not 100% tested.

Table 17. DC characteristics

Symbol	Parameter	Test condition		Min	Max	Unit
I_{LI}(1)	Input leakage current	0 V ≤ V_{IN} ≤ V_{CC}			±1	µA
I_{LO}	Output leakage current	0 V ≤ V_{OUT} ≤ V_{CC}			±1	µA
I_{CC1}	Supply current (read)	$\overline{E}$ = V_{IL}, $\overline{G}$ = V_{IH}, f = 6 MHz			10	mA
I_{CC2}	Supply current (standby)	$\overline{E}$ = V_{CC} ±0.2 V, $\overline{RP}$ = V_{CC} ±0.2 V			100	µA
I_{CC3}	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} = 2.7 V ± 10%		11.5	12.5	V
I_{PP}	Current for $V_{PP}/\overline{WP}$ program acceleration	V_{CC} = 2.7 V ± 10%			15	mA
V_{OL}	Output low voltage	I_{OL} = 1.8 mA			0.45	V
V_{OH}	Output high voltage	I_{OH} = –100 µA		V_{CC} –0.4		V
V_{ID}	Identification voltage			11.5	12.5	V
V_{LKO}(2)	Program/erase lockout supply voltage			1.8	2.3	V

1. The maximum input leakage current is ± 5 µA on the $V_{PP}/\overline{WP}$ pin.
2. Sampled only, not 100% tested.

Figure 13. Read mode AC waveforms (8-bit mode)



1. Data are output on DQ0-DQ7. DQ8-DQ15 are Hi-Z.

Figure 14. Page read AC waveforms (8-bit mode)

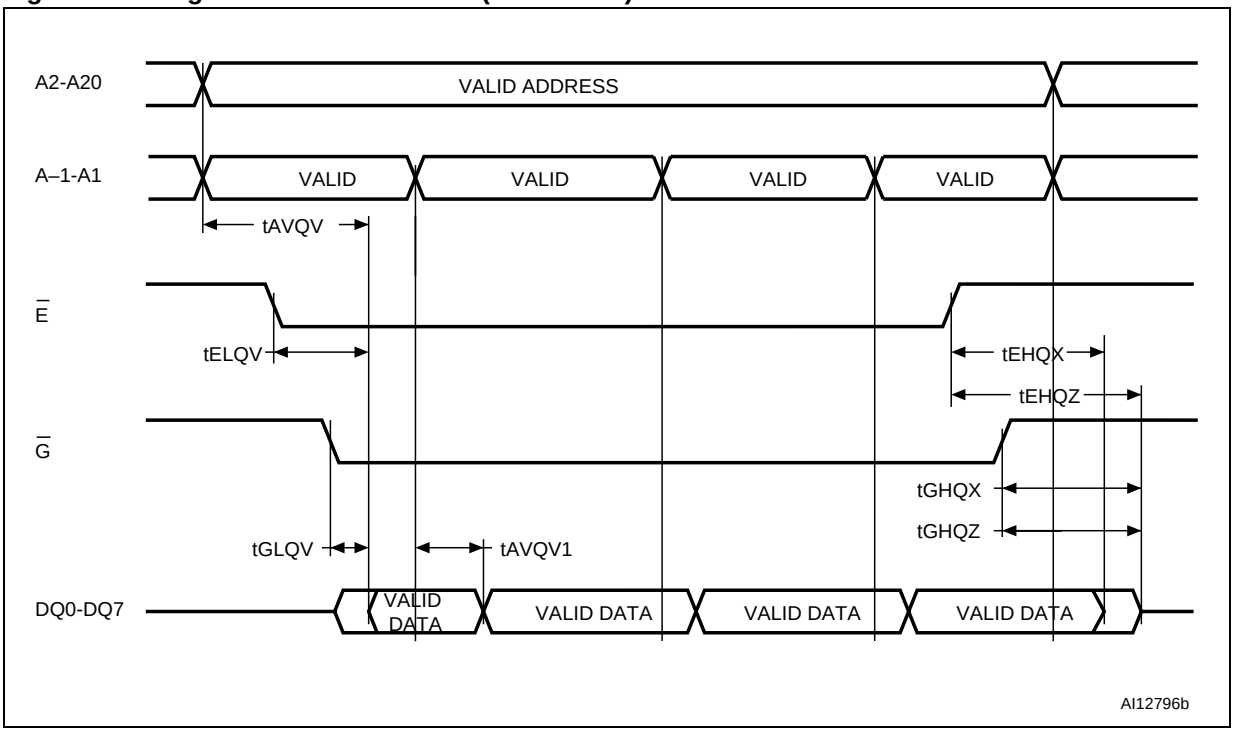


Table 18. Read AC characteristics

Symbol	Alt	Parameter	Test condition		M29W640GT, M29W640GB, M29W640GH, M29W640GL			Unit
					60	70	90	
t_{AVAX}	t_{RC}	Address Valid to Next Address Valid	$\overline{E} = V_{IL}$, $\overline{G} = V_{IL}$	Min	60	70	90	ns
t_{AVQV}	t_{ACC}	Address Valid to Output Valid	$\overline{E} = V_{IL}$, $\overline{G} = V_{IL}$	Max	60	70	90	ns
t_{AVQV1}	t_{PAGE}	Address Valid to Output Valid (Page)	$\overline{E} = V_{IL}$, $\overline{G} = V_{IL}$	Max	25	30	30	ns
t_{ELQX}(1)	t_{LZ}	Chip Enable Low to Output Transition	$\overline{G} = V_{IL}$	Min	0	0	0	ns
t_{ELQV}	t_{CE}	Chip Enable Low to Output Valid	$\overline{G} = V_{IL}$	Max	60	70	90	ns
t_{GLQX}(1)	t_{OLZ}	Output Enable Low to Output Transition	$\overline{E} = V_{IL}$	Min	0	0	0	ns
t_{GLQV}	t_{OE}	Output Enable Low to Output Valid	$\overline{E} = V_{IL}$	Max	25	30	30	ns
t_{EHQZ}(1)	t_{HZ}	Chip Enable High to Output Hi-Z	$\overline{G} = V_{IL}$	Max	25	30	30	ns
t_{GHQZ} t_{EHQZ}(1)	t_{DF}	Output Enable High to Output Hi-Z	$\overline{E} = V_{IL}$	Max	25	30	30	ns
t_{EHQX} t_{GHQX} t_{AXQX}	t_{OH}	Chip Enable, Output Enable or Address Transition to Output Transition		Min	0	0	0	ns
t_{ELBL} t_{ELBH}	t_{ELFL} t_{ELFH}	Chip Enable to $\overline{BYTE}$ Low or High		Max	5	5	5	ns
t_{BLQZ}	t_{FLQZ}	$\overline{BYTE}$ Low to Output Hi-Z		Max	25	25	25	ns
t_{BHQV}	t_{FHQV}	$\overline{BYTE}$ High to Output Valid		Max	25	30	30	ns

1. Sampled only, not 100% tested.

Figure 15. Write AC waveforms, write enable controlled (8-bit mode)

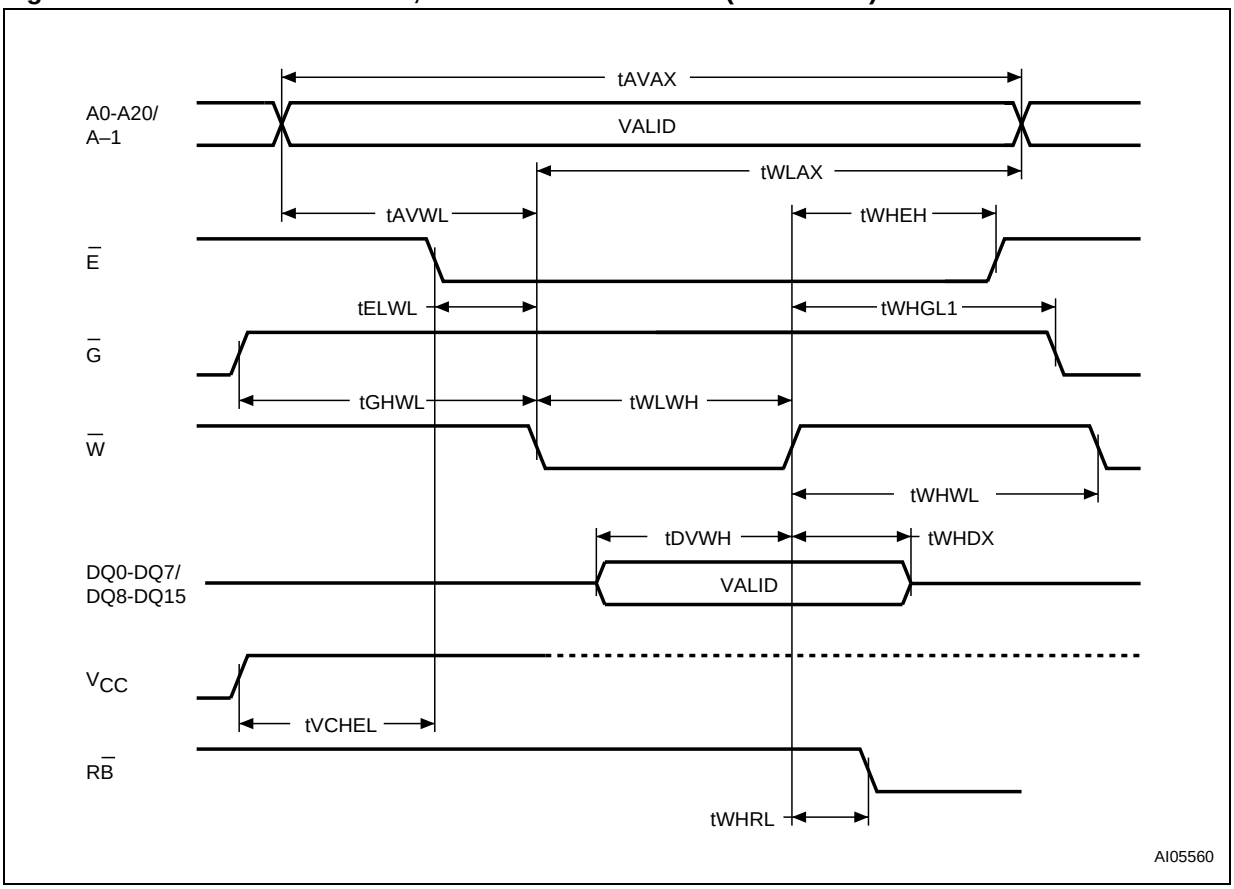


Figure 16. Write AC waveforms, chip enable controlled (8-bit mode)

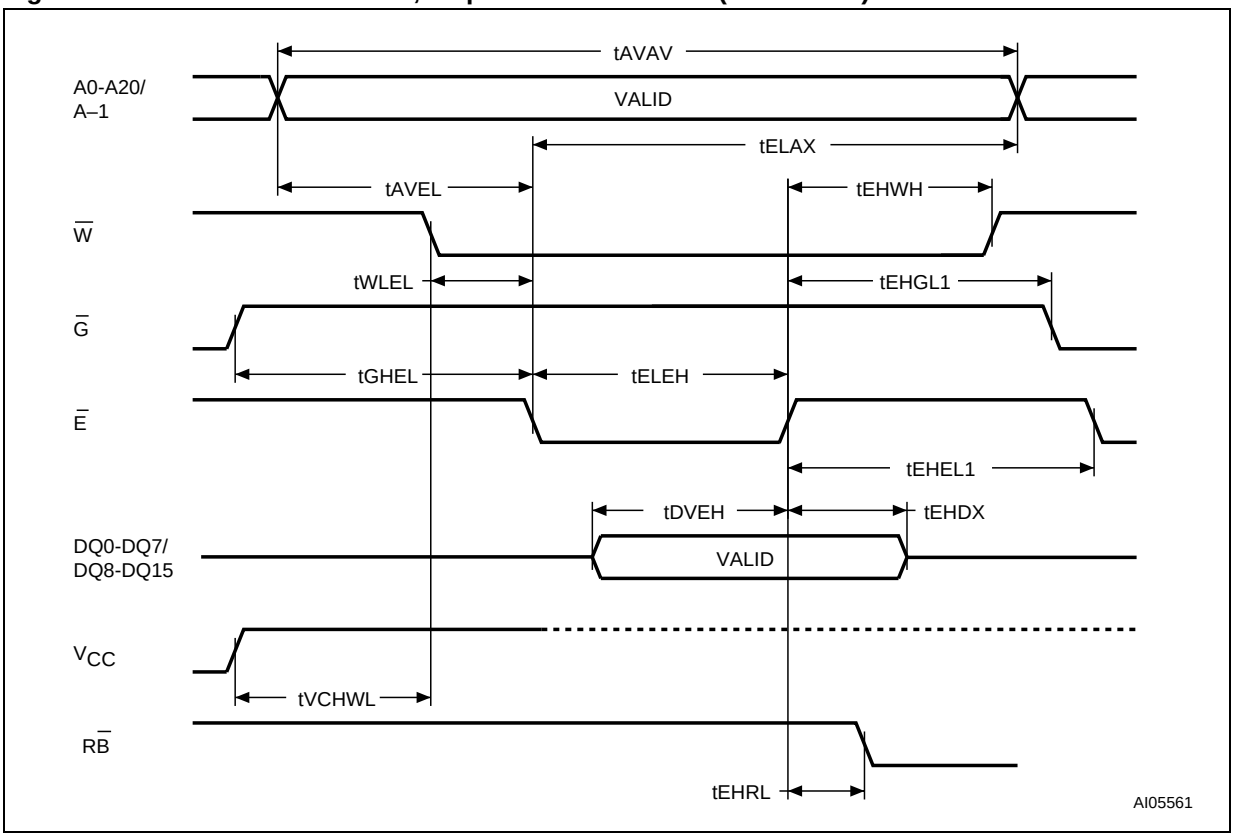


Table 19. Write AC characteristics

Symbol	Alt	Parameter		M29W640GT, M29W640GB, M29W640GH, M29W640GL			Unit
				60	70	90	
t_{AVAX}	t_{WC}	Address Valid to Next Address Valid	Min	60	70	90	ns
t_{ELWL}	t_{CS}	Chip Enable Low to Write Enable Low	Min	0	0	0	ns
t_{WLEL}	t_{WS}	Write Enable Low to Chip Enable Low	Min	0	0	0	ns
t_{WLWH}	t_{WP}	Write Enable Low to Write Enable High	Min	35	35	35	ns
t_{ELEH}	t_{CP}	Chip Enable Low to Chip Enable High	Min	35	35	35	ns
t_{DVWH} t_{DVEH}	t_{DS}	Input Valid to Write Enable or Chip Enable High	Min	30	30	30	ns
t_{WHDX} t_{EHDX}	t_{DH}	Write Enable or Chip Enable High to Input Transition	Min	0	0	0	ns
t_{WHEH}	t_{CH}	Write Enable High to Chip Enable High	Min	0	0	0	ns
t_{EHWH}	t_{WH}	Chip Enable High to Write Enable High	Min	0	0	0	ns
t_{WHWL}	t_{WPH}	Write Enable High to Write Enable Low	Min	25	25	25	ns
t_{WHGL1} t_{EHGL1}	t_{OEH}	Output Enable Hold time	Min	0	0	0	ns
t_{EHEL1}	t_{CPH}	Chip Enable High to Chip Enable Low	Min	25	25	25	ns
t_{AVWL} t_{AVEL}	t_{AS}	Address Valid to Write Enable or Chip Enable Low	Min	0	0	0	ns
t_{WLAX} t_{ELAX}	t_{AH}	Write Enable or Chip Enable Low to Address Transition	Min	45	45	45	ns
t_{GHWL}	t_{GHWL}	Output Enable High to Write Enable Low	Min	0	0	0	ns
t_{GHEL}	t_{GHEL}	Output Enable High to Chip Enable Low	Min	0	0	0	ns
t_{WHRL} (1) t_{EHRL}	t_{BUSY}	Program/Erase Valid to $R\overline{B}$ Low	Max	0	0	0	ns
t_{VCHEL} t_{VCHWL}	t_{VCS}	V_{CC} High to Chip Enable Low	Min	50	50	50	µs

1. Sampled only, not 100% tested.

Figure 17. Reset/Block Temporary Unprotect AC waveforms

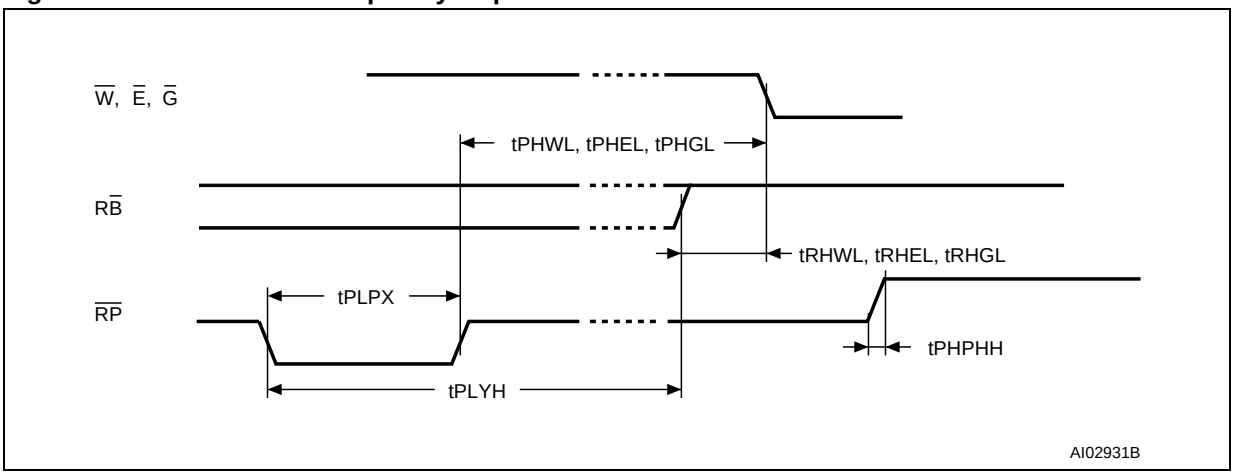


Figure 18. Accelerated program timing waveforms

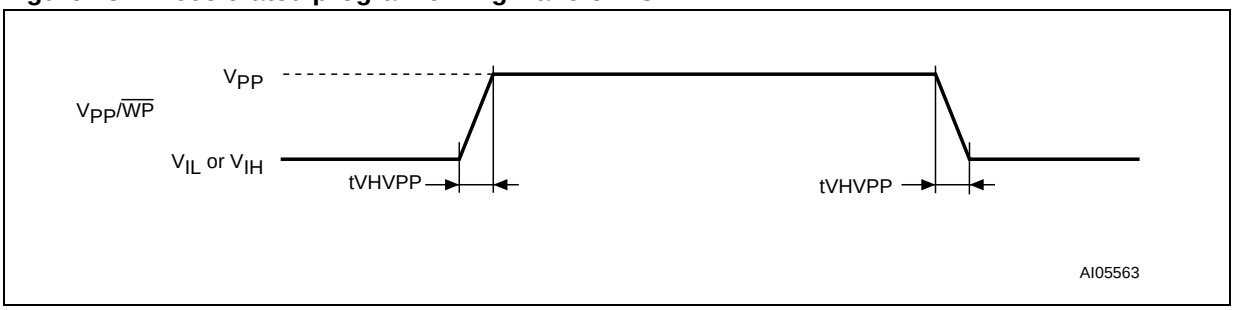
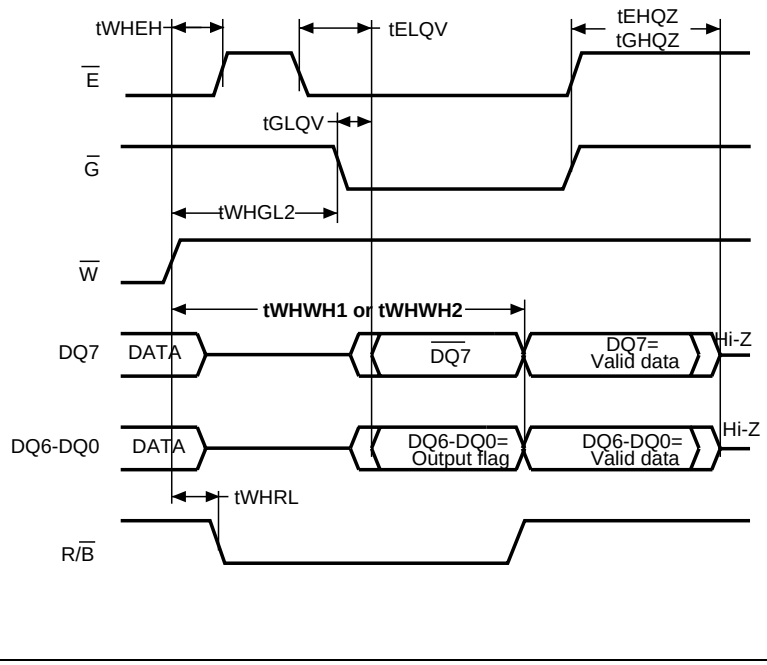


Table 20. Reset/Block Temporary Unprotect AC characteristics

Symbol	Alt	Parameter		M29W640GT, M29W640GB, M29W640GH, M29W640GL	Unit
t_{PHWL}(1) t_{PHEL} t_{PHGL}(1)	t_{RH}	RP High to Write Enable Low, Chip Enable Low, Output Enable Low	Min	50	ns
t_{RHWL}(1) t_{RHEL}(1) t_{RHGL}(1)	t_{RB}	RB High to Write Enable Low, Chip Enable Low, Output Enable Low	Min	0	ns
t_{PLPX}	t_{RP}	RP pulse width	Min	500	ns
t_{PLYH}	t_{READY}	RP Low to read mode	Max	50	µs
t_{PHPHH}(1)(2)	t_{VIDR}	RP rise time to V_{ID}	Min	500	ns
t_{VHVPP}(1)		V_{PP} rise and fall time	Min	500	ns

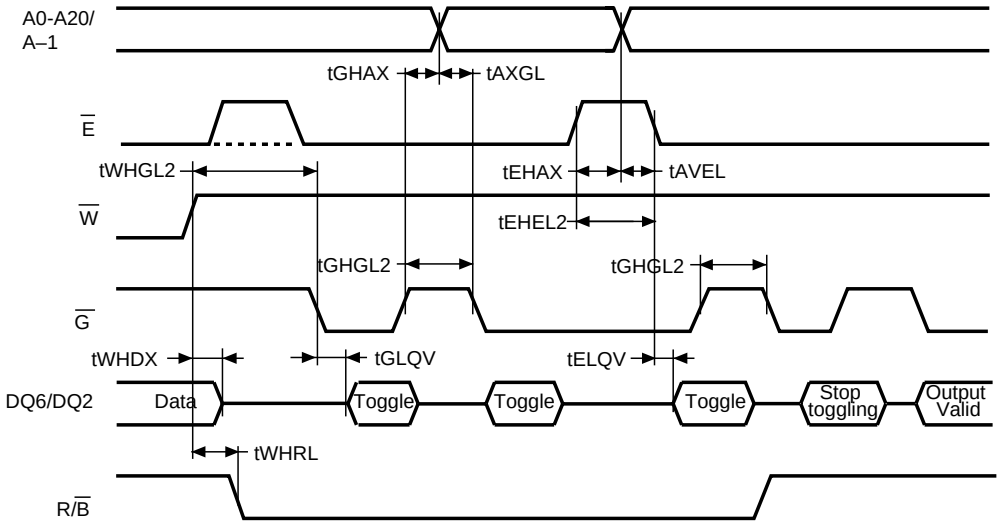
1. Sampled only, not 100% tested.
2. For fast program operations using V_{PP}/WP at 12 V.

Figure 19. Data polling AC waveforms



1. DQ7 returns valid data bit when the ongoing Program or Erase command is completed.
2. See *Table 21: Data polling and data toggle AC characteristics* and *Table 18: Read AC characteristics* for details on the timings.

Figure 20. Toggle/alternative toggle bit polling AC waveforms (8-bit mode)



1. DQ6 stops toggling when the ongoing Program or Erase command is completed. DQ2 stops toggling when the ongoing Chip Erase or Block Erase command is completed.
2. See *Table 21: Data polling and data toggle AC characteristics* and *Table 18: Read AC characteristics* for details on the timings.

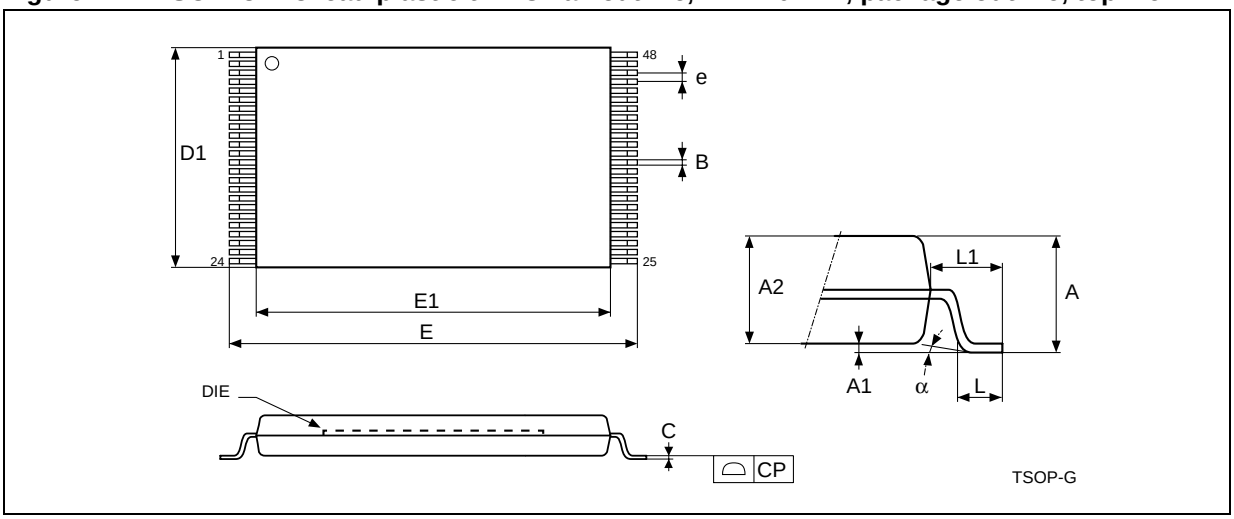
Table 21. Data polling and data toggle AC characteristics

Symbol	Alt	Parameter		M29W640GT, M29W640GB, M29W640GH, M29W640GL			Unit
				60	70	90	
t_{AXGL}	t_{ASO}	Address setup time to Output Enable Low during toggle bit polling	Min	10	10	10	ns
t_{GHAX} t_{EHAX}	t_{AHT}	Address hold time from Output Enable during toggle bit polling	Min	10	10	10	ns
t_{EHEL2}	t_{CEPH}	Chip Enable High during toggle bit polling	Min	10	10	10	ns
t_{WHGL2} t_{GHGL2}	t_{OEH}	Output hold time during data and toggle bit polling	Min	20	20	20	ns

8 Package mechanical

To meet environmental requirements, Numonyx offers these devices in RoHS compliant packages, which 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.

Figure 21. TSOP48 –48 lead plastic thin small outline, 12 x 20 mm, package outline, top view

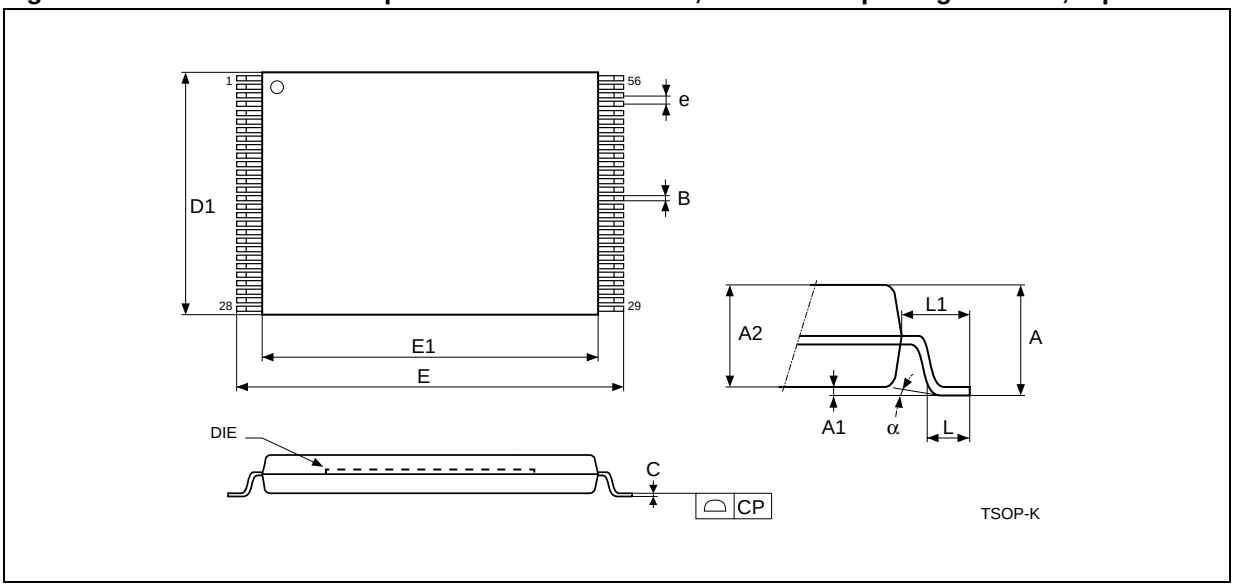


1. Drawing is not to scale.

Table 22. TSOP48 – 48 lead plastic thin small outline, 12 x 20 mm, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.20			0.047
A1	0.10	0.05	0.15	0.004	0.002	0.006
A2	1.00	0.95	1.05	0.039	0.037	0.041
B	0.22	0.17	0.27	0.009	0.007	0.011
C		0.10	0.21		0.004	0.008
CP			0.10			0.004
D1	12.00	11.90	12.10	0.472	0.468	0.476
E	20.00	19.80	20.20	0.787	0.779	0.795
E1	18.40	18.30	18.50	0.724	0.720	0.728
e	0.50	–	–	0.020	–	–
L	0.60	0.50	0.70	0.024	0.020	0.028
L1	0.80			0.031		
α	3°	0°	5°	3°	0°	5°

Figure 22. TSOP56 – 56 lead plastic thin small outline, 14 x 20 mm package outline, top view

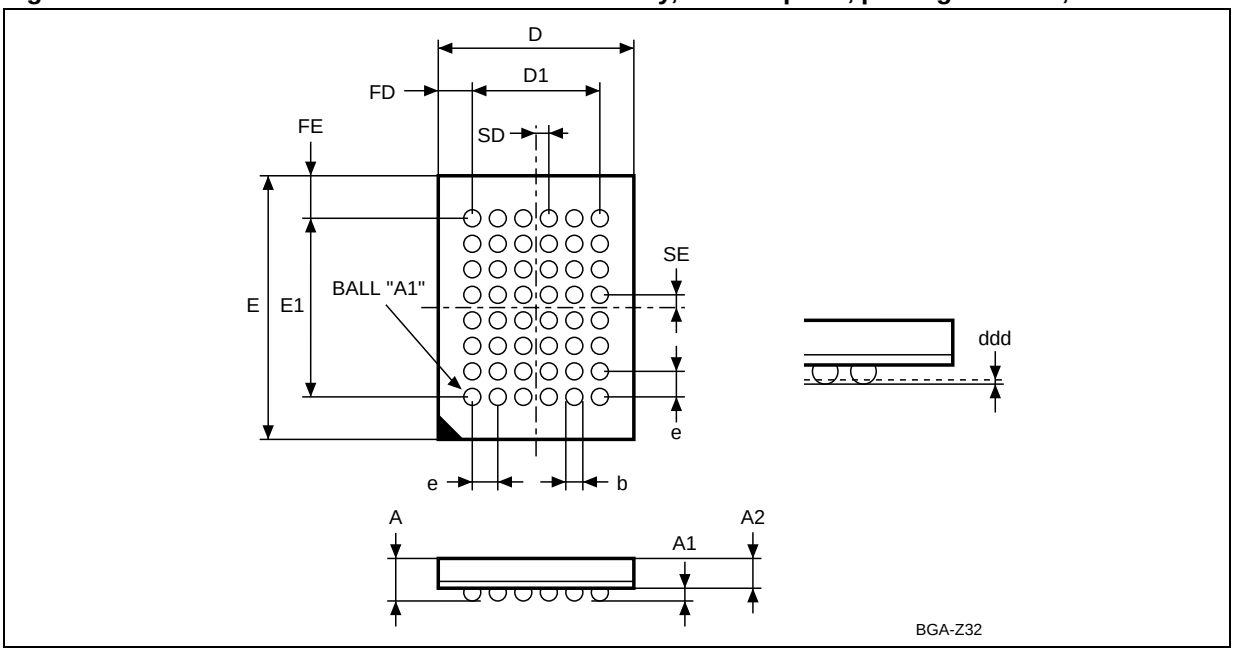


1. Drawing is not to scale.

Table 23. TSOP56 – 56 lead plastic thin small outline, 14 x 20 mm, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.20			0.047
A1	0.10	0.05	0.15	0.004	0.002	0.006
A2	1.00	0.95	1.05	0.039	0.037	0.041
B	0.22	0.17	0.27	0.009	0.007	0.011
C		0.10	0.21		0.004	0.008
CP			0.10			0.004
D1	14.00	13.90	14.10	0.551	0.547	0.555
E	20.00	19.80	20.20	0.787	0.780	0.795
E1	18.40	18.30	18.50	0.724	0.720	0.728
e	0.50	–	–	0.020	–	–
L	0.60	0.50	0.70	0.024	0.020	0.028
α	3°	0°	5°	3°	0°	5°

Figure 23. TFBGA48 6 x 8 mm-6 x 8 active ball array, 0.8 mm pitch, package outline, bottom view

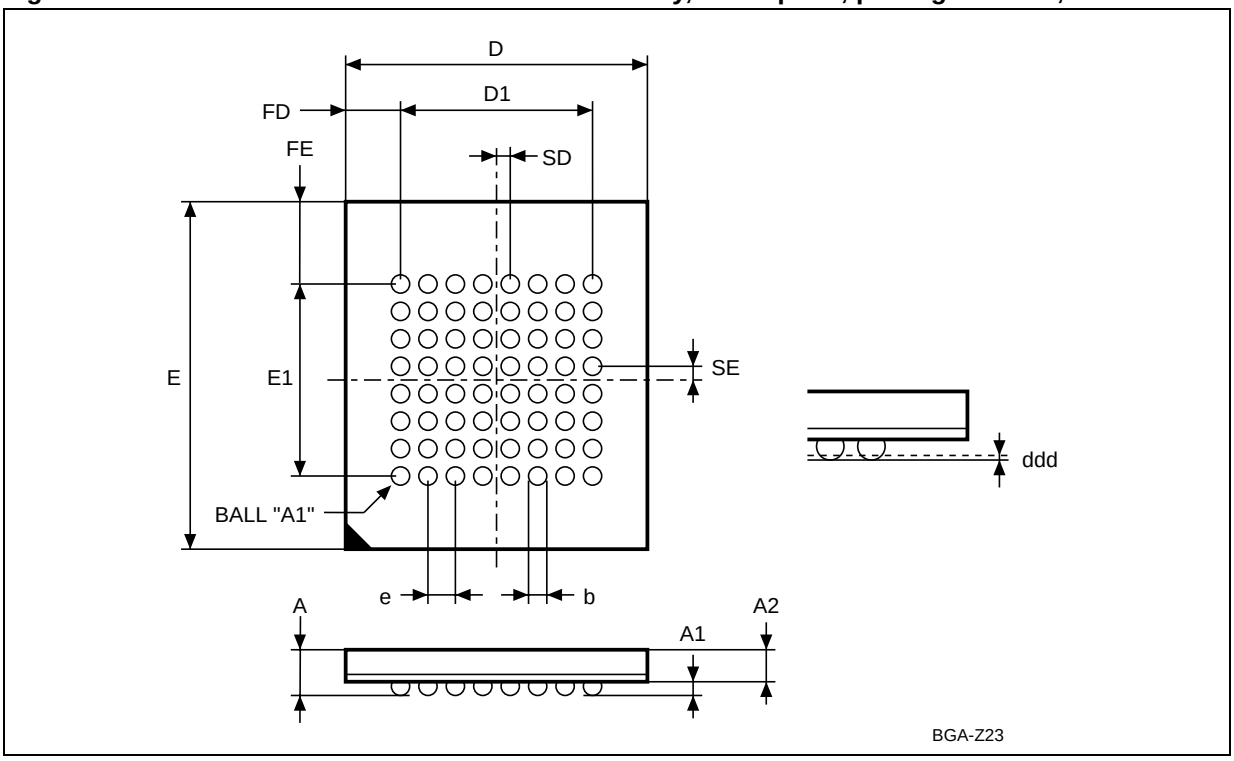


1. Drawing is not to scale.

Table 24. TFBGA48 6 x 8 mm - 6 x 8 active ball array, 0.8 mm pitch, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.20			0.047
A1		0.26			0.010	
A2			0.90			0.035
b		0.35	0.45		0.014	0.018
D	6.00	5.90	6.10	0.236	0.232	0.240
D1	4.00	–	–	0.157	–	–
ddd			0.10			0.004
E	8.00	7.90	8.10	0.315	0.311	0.319
E1	5.60	–	–	0.220	–	–
e	0.80	–	–	0.031	–	–
FD	1.00	–	–	0.039	–	–
FE	1.20	–	–	0.047	–	–
SD	0.40	–	–	0.016	–	–
SE	0.40	–	–	0.016	–	–

Figure 24. TBGA64 10 x 13 mm-8 x 8 active ball array, 1 mm pitch, package outline, bottom view

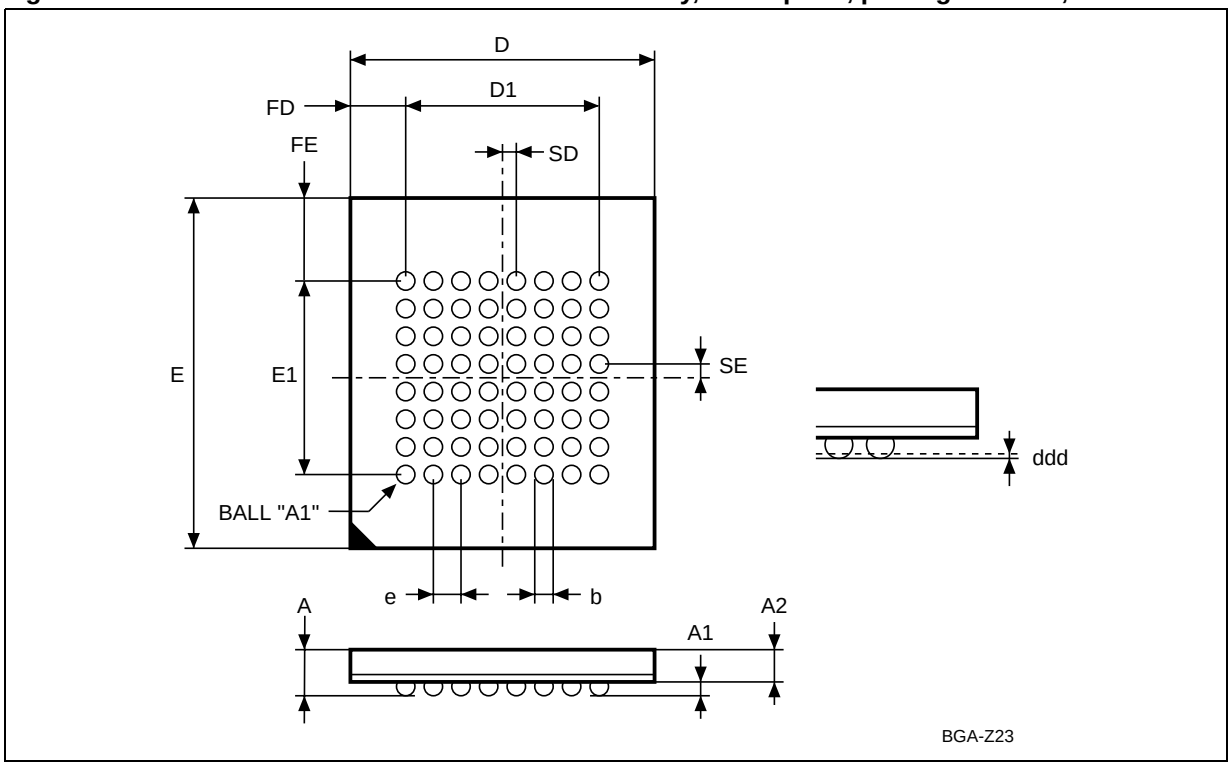


1. Drawing is not to scale.

Table 25. TBGA64 10 x 13 mm - 8 x 8 active ball array, 1 mm pitch, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.20			0.0472
A1	0.30	0.20	0.35	0.012	0.008	0.014
A2	0.80			0.031		
b		0.35	0.50		0.014	0.020
D	10.00	9.90	10.10	0.394	0.390	0.398
D1	7.00	–	–	0.276	–	–
ddd			0.10			0.004
e	1.00	–	–	0.039	–	–
E	13.00	12.90	13.10	0.512	0.508	0.516
E1	7.00	–	–	0.276	–	–
FD	1.50	–	–	0.059	–	–
FE	3.00	–	–	0.118	–	–
SD	0.50	–	–	0.020	–	–
SE	0.50	–	–	0.020	–	–

Figure 25. FBGA64 11 x 13 mm - 8 x 8 active ball array, 1 mm pitch, package outline, bottom view



1. Drawing is not to scale.

Table 26. FBGA64 11 x 13 mm—8 x 8 active ball array, 1 mm pitch, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A	—	—	1.40	—	—	0.055
A1	0.48	0.43	0.53	0.018	0.016	
A2	0.80	—	—	0.031	—	—
b	—	0.55	0.65	—	0.021	0.025
D	11.00	10.90	11.10	0.433	0.429	0.437
D1	7.00	—	—	0.275	—	—
ddd	—	—	0.15	—	—	0.0059
e	1.00	—	—	0.039	—	—
E	13.0	12.90	13.10	0.511	0.507	0.515
E1	7.00	—	—	0.275	—	—
FD	2.00	—	—	0.078	—	—
FE	3.00	—	—	0.118	—	—
SD	0.50	—	—	0.0196	—	—
SE	0.50	—	—	0.0196	—	—

9 Ordering information

Table 27. Ordering information scheme

Example: M29W640GT 70 NA 6 F

Device type

M29

Operating voltage

W = V_{CC} = 2.7 to 3.6 V

Device function

640G = 64-Mbit (x8/x16), boot block, uniform or boot block

Array matrix

T = top boot

B = bottom boot

H = last block protected by $V_{PP}/\overline{WP}$

L = first block protected by $V_{PP}/\overline{WP}$

Speed

60 = 60 ns

70 = 70 ns

90 = 90 ns

7A = 70 ns Automotive -40 °C to 85 °C Certified Part, used in conjiuction with temperature range option 6 to distinguish the Automotive Part.

Package

NA = TSOP48: 12 x 20 mm

NB = TSOP56: 14 x 20 mm(1)

ZA= TFBGA48: 6 x 8 mm - 0.8 mm pitch

ZF = TBGA64: 10 x 13 mm -1 mm pitch(1)

ZS = FBGA64: 11 x 13 mm, 1 mm pitch

Temperature range

6 = –40 to 85 °C

3 = –40 to 125 °C

Option

E = RoHS package, standard packing

F = RoHS package, tape & reel packing

1. Packages only available upon request.

Note: This product is also available with the extended block factory locked. 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 28. M29W640GH and M29W640GL block addresses

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
0	64/32	Protection group	000000h–00FFFFh(1)	000000h–007FFFh(1)
1	64/32	Protection group	010000h–01FFFFh(1)	008000h–00FFFFh
2	64/32	Protection group	020000h–02FFFFh(1)	010000h–017FFFh(1)
3	64/32	Protection group	030000h–03FFFFh(1)	018000h–01FFFFh(1)
4	64/32	Protection group	040000h–04FFFFh	020000h–027FFFh
5	64/32		050000h–05FFFFh	028000h–02FFFFh
6	64/32		060000h–06FFFFh	030000h–037FFFh
7	64/32		070000h–07FFFFh	038000h–03FFFFh
8	64/32	Protection group	080000h–08FFFFh	040000h–047FFFh
9	64/32		090000h–09FFFFh	048000h–04FFFFh
10	64/32		0A0000h–0AFFFFh	050000h–057FFFh
11	64/32		0B0000h–0BFFFFh	058000h–05FFFFh
12	64/32	Protection group	0C0000h–0CFFFFh	060000h–067FFFh
13	64/32		0D0000h–0DFFFFh	068000h–06FFFFh
14	64/32		0E0000h–0EFFFFh	070000h–077FFFh
15	64/32		0F0000h–0FFFFFh	078000h–07FFFFh
16	64/32	Protection group	100000h–10FFFFh	080000h–087FFFh
17	64/32		110000h–11FFFFh	088000h–08FFFFh
18	64/32		120000h–12FFFFh	090000h–097FFFh
19	64/32		130000h–13FFFFh	098000h–09FFFFh
20	64/32	Protection group	140000h–14FFFFh	0A0000h–0A7FFFh
21	64/32		150000h–15FFFFh	0A8000h–0AFFFFh
22	64/32		160000h–16FFFFh	0B0000h–0B7FFFh
23	64/32		170000h–17FFFFh	0B8000h–0BFFFFh
24	64/32	Protection group	180000h–18FFFFh	0C0000h–0C7FFFh
25	64/32		190000h–19FFFFh	0C8000h–0CFFFFh
26	64/32		1A0000h–1AFFFFh	0D0000h–0D7FFFh
27	64/32		1B0000h–1BFFFFh	0D8000h–0DFFFFh
28	64/32	Protection group	1C0000h–1CFFFFh	0E0000h–0E7FFFh
29	64/32		1D0000h–1DFFFFh	0E8000h–0EFFFFh
30	64/32		1E0000h–1EFFFFh	0F0000h–0F7FFFh
31	64/32		1F0000h–1FFFFFh	0F8000h–0FFFFFh

Table 28. M29W640GH and M29W640GL block addresses (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
32	64/32	Protection group	200000h–20FFFFh	100000h–107FFFh
33	64/32		210000h–21FFFFh	108000h–10FFFFh
34	64/32		220000h–22FFFFh	110000h–117FFFh
35	64/32		230000h–23FFFFh	118000h–11FFFFh
36	64/32	Protection group	240000h–24FFFFh	120000h–127FFFh
37	64/32		250000h–25FFFFh	128000h–12FFFFh
38	64/32		260000h–26FFFFh	130000h–137FFFh
39	64/32		270000h–27FFFFh	138000h–13FFFFh
40	64/32	Protection group	280000h–28FFFFh	140000h–147FFFh
41	64/32		290000h–29FFFFh	148000h–14FFFFh
42	64/32		2A0000h–2AFFFFh	150000h–157FFFh
43	64/32		2B0000h–2BFFFFh	158000h–15FFFFh
44	64/32	Protection group	2C0000h–2CFFFFh	160000h–167FFFh
45	64/32		2D0000h–2DFFFFh	168000h–16FFFFh
46	64/32		2E0000h–2EFFFFh	170000h–177FFFh
47	64/32		2F0000h–2FFFFFh	178000h–17FFFFh
48	64/32	Protection group	300000h–30FFFFh	180000h–187FFFh
49	64/32		310000h–31FFFFh	188000h–18FFFFh
50	64/32		320000h–32FFFFh	190000h–197FFFh
51	64/32		330000h–33FFFFh	198000h–19FFFFh
52	64/32	Protection group	340000h–34FFFFh	1A0000h–1A7FFFh
53	64/32		350000h–35FFFFh	1A8000h–1AFFFFh
54	64/32		360000h–36FFFFh	1B0000h–1B7FFFh
55	64/32		370000h–37FFFFh	1B8000h–1BFFFFh
56	64/32	Protection group	380000h–38FFFFh	1C0000h–1C7FFFh
57	64/32		390000h–39FFFFh	1C8000h–1CFFFFh
58	64/32		3A0000h–3AFFFFh	1D0000h–1D7FFFh
59	64/32		3B0000h–3BFFFFh	1D8000h–1DFFFFh
60	64/32	Protection group	3C0000h–3CFFFFh	1E0000h–1E7FFFh
61	64/32		3D0000h–3DFFFFh	1E8000h–1EFFFFh
62	64/32		3E0000h–3EFFFFh	1F0000h–1F7FFFh
63	64/32		3F0000h–3FFFFFh	1F8000h–1FFFFFh

Table 28. M29W640GH and M29W640GL block addresses (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
64	64/32	Protection group	400000h–40FFFFh	200000h–207FFFh
65	64/32		410000h–41FFFFh	208000h–20FFFFh
66	64/32		420000h–42FFFFh	210000h–217FFFh
67	64/32		430000h–43FFFFh	218000h–21FFFFh
68	64/32	Protection group	440000h–44FFFFh	220000h–227FFFh
69	64/32		450000h–45FFFFh	228000h–22FFFFh
70	64/32		460000h–46FFFFh	230000h–237FFFh
71	64/32		470000h–47FFFFh	238000h–23FFFFh
72	64/32	Protection group	480000h–48FFFFh	240000h–247FFFh
73	64/32		490000h–49FFFFh	248000h–24FFFFh
74	64/32		4A0000h–4AFFFFh	250000h–257FFFh
75	64/32		4B0000h–4BFFFFh	258000h–25FFFFh
76	64/32	Protection group	4C0000h–4CFFFFh	260000h–267FFFh
77	64/32		4D0000h–4DFFFFh	268000h–26FFFFh
78	64/32		4E0000h–4EFFFFh	270000h–277FFFh
79	64/32		4F0000h–4FFFFFh	278000h–27FFFFh
80	64/32	Protection group	500000h–50FFFFh	280000h–287FFFh
81	64/32		510000h–51FFFFh	288000h–28FFFFh
82	64/32		520000h–52FFFFh	290000h–297FFFh
83	64/32		530000h–53FFFFh	298000h–29FFFFh
84	64/32	Protection group	540000h–54FFFFh	2A0000h–2A7FFFh
85	64/32		550000h–55FFFFh	2A8000h–2AFFFFh
86	64/32		560000h–56FFFFh	2B0000h–2B7FFFh
87	64/32		570000h–57FFFFh	2B8000h–2BFFFFh
88	64/32	Protection group	580000h–58FFFFh	2C0000h–2C7FFFh
89	64/32		590000h–59FFFFh	2C8000h–2CFFFFh
90	64/32		5A0000h–5AFFFFh	2D0000h–2D7FFFh
91	64/32		5B0000h–5BFFFFh	2D8000h–2DFFFFh
92	64/32	Protection group	5C0000h–5CFFFFh	2E0000h–2E7FFFh
93	64/32		5D0000h–5DFFFFh	2E8000h–2EFFFFh
94	64/32		5E0000h–5EFFFFh	2F0000h–2F7FFFh
95	64/32		5F0000h–5FFFFFh	2F8000h–2FFFFFh

Table 28. M29W640GH and M29W640GL block addresses (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
96	64/32	Protection group	600000h–60FFFFh	300000h–307FFFh
97	64/32		610000h–61FFFFh	308000h–30FFFFh
98	64/32		620000h–62FFFFh	310000h–317FFFh
99	64/32		630000h–63FFFFh	318000h–31FFFFh
100	64/32	Protection group	640000h–64FFFFh	320000h–327FFFh
101	64/32		650000h–65FFFFh	328000h–32FFFFh
102	64/32		660000h–66FFFFh	330000h–337FFFh
103	64/32		670000h–67FFFFh	338000h–33FFFFh
104	64/32	Protection group	680000h–68FFFFh	340000h–347FFFh
105	64/32		690000h–69FFFFh	348000h–34FFFFh
106	64/32		6A0000h–6AFFFFh	350000h–357FFFh
107	64/32		6B0000h–6BFFFFh	358000h–35FFFFh
108	64/32	Protection group	6C0000h–6CFFFFh	360000h–367FFFh
109	64/32		6D0000h–6DFFFFh	368000h–36FFFFh
110	64/32		6E0000h–6EFFFFh	370000h–377FFFh
111	64/32		6F0000h–6FFFFFh	378000h–37FFFFh
112	64/32	Protection group	700000h–70FFFFh	380000h–387FFFh
113	64/32		710000h–71FFFFh	388000h–38FFFFh
114	64/32		720000h–72FFFFh	390000h–397FFFh
115	64/32		730000h–73FFFFh	398000h–39FFFFh
116	64/32	Protection group	740000h–74FFFFh	3A0000h–3A7FFFh
117	64/32		750000h–75FFFFh	3A8000h–3AFFFFh
118	64/32		760000h–76FFFFh	3B0000h–3B7FFFh
119	64/32		770000h–77FFFFh	3B8000h–3BFFFFh
120	64/32	Protection group	780000h–78FFFFh	3C0000h–3C7FFFh
121	64/32		790000h–79FFFFh	3C8000h–3CFFFFh
122	64/32		7A0000h–7AFFFFh	3D0000h–3D7FFFh
123	64/32		7B0000h–7BFFFFh	3D8000h–3DFFFFh
124	64/32	Protection group	7C0000h–7CFFFFh	3E0000h–3E7FFFh
125	64/32	Protection group	7D0000h–7DFFFFh	3E8000h–3EFFFFh
126	64/32	Protection group	7E0000h–7EFFFFh	3F0000h–3F7FFFh
127	64/32	Protection group	7F0000h–7FFFFFh	3F8000h–3FFFFFh

1. Used as the Extended Block Addresses in Extended Block mode.

Table 29. Top boot block addresses, M29W640GT

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
0	64/32	Protection group	000000h–00FFFFh(1)	000000h–007FFFh(1)
1	64/32		010000h–01FFFFh(1)	008000h–00FFFFh(1)
2	64/32		020000h–02FFFFh(1)	010000h–017FFFh(1)
3	64/32		030000h–03FFFFh(1)	018000h–01FFFFh(1)
4	64/32	Protection group	040000h–04FFFFh	020000h–027FFFh
5	64/32		050000h–05FFFFh	028000h–02FFFFh
6	64/32		060000h–06FFFFh	030000h–037FFFh
7	64/32		070000h–07FFFFh	038000h–03FFFFh
8	64/32	Protection group	080000h–08FFFFh	040000h–047FFFh
9	64/32		090000h–09FFFFh	048000h–04FFFFh
10	64/32		0A0000h–0AFFFFh	050000h–057FFFh
11	64/32		0B0000h–0BFFFFh	058000h–05FFFFh
12	64/32	Protection group	0C0000h–0CFFFFh	060000h–067FFFh
13	64/32		0D0000h–0DFFFFh	068000h–06FFFFh
14	64/32		0E0000h–0EFFFFh	070000h–077FFFh
15	64/32		0F0000h–0FFFFFh	078000h–07FFFFh
16	64/32	Protection group	100000h–10FFFFh	080000h–087FFFh
17	64/32		110000h–11FFFFh	088000h–08FFFFh
18	64/32		120000h–12FFFFh	090000h–097FFFh
19	64/32		130000h–13FFFFh	098000h–09FFFFh
20	64/32	Protection group	140000h–14FFFFh	0A0000h–0A7FFFh
21	64/32		150000h–15FFFFh	0A8000h–0AFFFFh
22	64/32		160000h–16FFFFh	0B0000h–0B7FFFh
23	64/32		170000h–17FFFFh	0B8000h–0BFFFFh
24	64/32	Protection group	180000h–18FFFFh	0C0000h–0C7FFFh
25	64/32		190000h–19FFFFh	0C8000h–0CFFFFh
26	64/32		1A0000h–1AFFFFh	0D0000h–0D7FFFh
27	64/32		1B0000h–1BFFFFh	0D8000h–0DFFFFh
28	64/32	Protection group	1C0000h–1CFFFFh	0E0000h–0E7FFFh
29	64/32		1D0000h–1DFFFFh	0E8000h–0EFFFFh
30	64/32		1E0000h–1EFFFFh	0F0000h–0F7FFFh
31	64/32		1F0000h–1FFFFFh	0F8000h–0FFFFFh

Table 29. Top boot block addresses, M29W640GT (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
32	64/32	Protection group	200000h–20FFFFh	100000h–107FFFh
33	64/32		210000h–21FFFFh	108000h–10FFFFh
34	64/32		220000h–22FFFFh	110000h–117FFFh
35	64/32		230000h–23FFFFh	118000h–11FFFFh
36	64/32	Protection group	240000h–24FFFFh	120000h–127FFFh
37	64/32		250000h–25FFFFh	128000h–12FFFFh
38	64/32		260000h–26FFFFh	130000h–137FFFh
39	64/32		270000h–27FFFFh	138000h–13FFFFh
40	64/32	Protection group	280000h–28FFFFh	140000h–147FFFh
41	64/32		290000h–29FFFFh	148000h–14FFFFh
42	64/32		2A0000h–2AFFFFh	150000h–157FFFh
43	64/32		2B0000h–2BFFFFh	158000h–15FFFFh
44	64/32	Protection group	2C0000h–2CFFFFh	160000h–167FFFh
45	64/32		2D0000h–2DFFFFh	168000h–16FFFFh
46	64/32		2E0000h–2EFFFFh	170000h–177FFFh
47	64/32		2F0000h–2FFFFFh	178000h–17FFFFh
48	64/32	Protection group	300000h–30FFFFh	180000h–187FFFh
49	64/32		310000h–31FFFFh	188000h–18FFFFh
50	64/32		320000h–32FFFFh	190000h–197FFFh
51	64/32		330000h–33FFFFh	198000h–19FFFFh
52	64/32	Protection group	340000h–34FFFFh	1A0000h–1A7FFFh
53	64/32		350000h–35FFFFh	1A8000h–1AFFFFh
54	64/32		360000h–36FFFFh	1B0000h–1B7FFFh
55	64/32		370000h–37FFFFh	1B8000h–1BFFFFh
56	64/32	Protection group	380000h–38FFFFh	1C0000h–1C7FFFh
57	64/32		390000h–39FFFFh	1C8000h–1CFFFFh
58	64/32		3A0000h–3AFFFFh	1D0000h–1D7FFFh
59	64/32		3B0000h–3BFFFFh	1D8000h–1DFFFFh
60	64/32	Protection group	3C0000h–3CFFFFh	1E0000h–1E7FFFh
61	64/32		3D0000h–3DFFFFh	1E8000h–1EFFFFh
62	64/32		3E0000h–3EFFFFh	1F0000h–1F7FFFh
63	64/32		3F0000h–3FFFFFh	1F8000h–1FFFFFh

Table 29. Top boot block addresses, M29W640GT (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
64	64/32	Protection group	400000h–40FFFFh	200000h–207FFFh
65	64/32		410000h–41FFFFh	208000h–20FFFFh
66	64/32		420000h–42FFFFh	210000h–217FFFh
67	64/32		430000h–43FFFFh	218000h–21FFFFh
68	64/32	Protection group	440000h–44FFFFh	220000h–227FFFh
69	64/32		450000h–45FFFFh	228000h–22FFFFh
70	64/32		460000h–46FFFFh	230000h–237FFFh
71	64/32		470000h–47FFFFh	238000h–23FFFFh
72	64/32	Protection group	480000h–48FFFFh	240000h–247FFFh
73	64/32		490000h–49FFFFh	248000h–24FFFFh
74	64/32		4A0000h–4AFFFFh	250000h–257FFFh
75	64/32		4B0000h–4BFFFFh	258000h–25FFFFh
76	64/32	Protection group	4C0000h–4CFFFFh	260000h–267FFFh
77	64/32		4D0000h–4DFFFFh	268000h–26FFFFh
78	64/32		4E0000h–4EFFFFh	270000h–277FFFh
79	64/32		4F0000h–4FFFFFh	278000h–27FFFFh
80	64/32	Protection group	500000h–50FFFFh	280000h–287FFFh
81	64/32		510000h–51FFFFh	288000h–28FFFFh
82	64/32		520000h–52FFFFh	290000h–297FFFh
83	64/32		530000h–53FFFFh	298000h–29FFFFh
84	64/32	Protection group	540000h–54FFFFh	2A0000h–2A7FFFh
85	64/32		550000h–55FFFFh	2A8000h–2AFFFFh
86	64/32		560000h–56FFFFh	2B0000h–2B7FFFh
87	64/32		570000h–57FFFFh	2B8000h–2BFFFFh
88	64/32	Protection group	580000h–58FFFFh	2C0000h–2C7FFFh
89	64/32		590000h–59FFFFh	2C8000h–2CFFFFh
90	64/32		5A0000h–5AFFFFh	2D0000h–2D7FFFh
91	64/32		5B0000h–5BFFFFh	2D8000h–2DFFFFh
92	64/32	Protection group	5C0000h–5CFFFFh	2E0000h–2E7FFFh
93	64/32		5D0000h–5DFFFFh	2E8000h–2EFFFFh
94	64/32		5E0000h–5EFFFFh	2F0000h–2F7FFFh
95	64/32		5F0000h–5FFFFFh	2F8000h–2FFFFFh

Table 29. Top boot block addresses, M29W640GT (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
96	64/32	Protection group	600000h–60FFFFh	300000h–307FFFh
97	64/32		610000h–61FFFFh	308000h–30FFFFh
98	64/32		620000h–62FFFFh	310000h–317FFFh
99	64/32		630000h–63FFFFh	318000h–31FFFFh
100	64/32	Protection group	640000h–64FFFFh	320000h–327FFFh
101	64/32		650000h–65FFFFh	328000h–32FFFFh
102	64/32		660000h–66FFFFh	330000h–337FFFh
103	64/32		670000h–67FFFFh	338000h–33FFFFh
104	64/32	Protection group	680000h–68FFFFh	340000h–347FFFh
105	64/32		690000h–69FFFFh	348000h–34FFFFh
106	64/32		6A0000h–6AFFFFh	350000h–357FFFh
107	64/32		6B0000h–6BFFFFh	358000h–35FFFFh
108	64/32	Protection group	6C0000h–6CFFFFh	360000h–367FFFh
109	64/32		6D0000h–6DFFFFh	368000h–36FFFFh
110	64/32		6E0000h–6EFFFFh	370000h–377FFFh
111	64/32		6F0000h–6FFFFFh	378000h–37FFFFh
112	64/32	Protection group	700000h–70FFFFh	380000h–387FFFh
113	64/32		710000h–71FFFFh	388000h–38FFFFh
114	64/32		720000h–72FFFFh	390000h–397FFFh
115	64/32		730000h–73FFFFh	398000h–39FFFFh
116	64/32	Protection group	740000h–74FFFFh	3A0000h–3A7FFFh
117	64/32		750000h–75FFFFh	3A8000h–3AFFFFh
118	64/32		760000h–76FFFFh	3B0000h–3B7FFFh
119	64/32		770000h–77FFFFh	3B8000h–3BFFFFh
120	64/32	Protection group	780000h–78FFFFh	3C0000h–3C7FFFh
121	64/32		790000h–79FFFFh	3C8000h–3CFFFFh
122	64/32		7A0000h–7AFFFFh	3D0000h–3D7FFFh
123	64/32		7B0000h–7BFFFFh	3D8000h–3DFFFFh
124	64/32	Protection group	7C0000h–7CFFFFh	3E0000h–3E7FFFh
125	64/32		7D0000h–7DFFFFh	3E8000h–3EFFFFh
126	64/32		7E0000h–7EFFFFh	3F0000h–3F7FFFh
127	8/4	Protection group	7F0000h–7F1FFFh	3F8000h–3F8FFFh
128	8/4	Protection group	7F2000h–7F3FFFh	3F9000h–3F9FFFh
129	8/4	Protection group	7F4000h–7F5FFFh	3FA000h–3FAFFFh
130	8/4	Protection group	7F6000h–7F7FFFh	3FB000h–3FBFFFh

Table 29. Top boot block addresses, M29W640GT (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
131	8/4	Protection group	7F8000h–7F9FFFh	3FC000h–3FCFFFh
132	8/4	Protection group	7FA000h–7FBFFFh	3FD000h–3FDFFFh
133	8/4	Protection group	7FC000h–7FDFFFh	3FE000h–3FEFFFh
134	8/4	Protection group	7FE000h–7FFFFFh	3FF000h–3FFFFFh

1. Used as the extended block addresses in extended block mode.

Table 30. Bottom boot block addresses, M29W640GB

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
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	004000h–004FFFh
5	8/4	Protection group	00A000h-00BFFFh	005000h–005FFFh
6	8/4	Protection group	00C000h-00DFFFh	006000h–006FFFh
7	8/4	Protection group	00E000h-00FFFFh	007000h–007FFFh
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
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

Table 30. Bottom boot block addresses, M29W640GB (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
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
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

Table 30. Bottom boot block addresses, M29W640GB (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
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
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
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

Table 30. Bottom boot block addresses, M29W640GB (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
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
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

Table 30. Bottom boot block addresses, M29W640GB (continued)

Block	Kbytes/Kwords	Protection block group	(x8)	(x16)
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	64/32		7F0000h-7FFFFFh	3F8000h–3FFFFFh

1. Used as the 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 CFI Query command is issued the device enters CFI query mode and the data structure is read from the memory. Tables *31*, *32*, *33*, *34*, *35*, and *36*, show the addresses 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 36: 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 31. Query structure overview(1)

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 32. CFI query identification string(1)

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 35*)	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 33. 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 4BCD value in volts bit 3 to 0BCD value in 100 mV	2.7 V
1Ch	38h	0036h	V_{CC} logic supply maximum program/erase voltage bit 7 to 4BCD value in volts bit 3 to 0BCD value in 100 mV	3.6 V
1Dh	3Ah	00B5h	V_{PP} [programming] supply minimum program/erase voltage bit 7 to 4HEX value in volts bit 3 to 0BCD value in 100 mV	11.5 V
1Eh	3Ch	00C5h	V_{PP} [programming] supply maximum program/erase voltage bit 7 to 4HEX value in volts bit 3 to 0BCD value in 100 mV	12.5 V
1Fh	3Eh	0004h	Typical timeout per single Byte/Word Program = 2^n µs	16 µs
20h	40h	0004h	Typical timeout for minimum size write buffer program = 2^n µs	16 µs
21h	42h	000Ah	Typical timeout per individual Block Erase = 2^n ms	1 s
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	0004h	Maximum timeout for Write Buffer Program = 2^n times typical	200 µs
25h	4Ah	0003h	Maximum timeout per individual Block Erase = 2^n times typical	8 s
26h	4Ch	0000h	Maximum timeout for Chip Erase = 2^n times typical	NA

Table 34. Device geometry definition(1)

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	0005h 0000h		Maximum number of bytes in multi-byte program or page = 2^n	32 bytes
2Ch	58h	M29W640GH, M29W640GL	0001h	Number of erase block regions. It specifies the number of regions containing contiguous Erase blocks of the same size.	1
		M29W640GT, M29W640GB	0002h		2
2Dh 2Eh	5Ah 5Ch	M29W640GH, M29W640GL	007Fh 0000h	Region 1 information Number of erase blocks of identical size = 007Fh+1	128
2Fh 30h	5Eh 60h		0000h 0001h	Region 1 information Block size in region 1 = 0100h * 256 byte	64 Kbytes
2Dh 2Eh	5Ah 5Ch	M29W640GT, M29W640GB	0007h 0000h	Region 1 information Number of erase blocks of identical size = 0007h+1	8
2Fh 30h	5Eh 60h		0020h 0000h	Region 1 information Block size in region 1 = 0020h * 256 byte	8 Kbytes
31h 32h	62h 64h	M29W640GT, M29W640GB only	007Eh 0000h	Region 2 information Number of erase blocks of identical size= 007Eh+1	127
33h 34h	66h 68h		0000h 0001h	Region 2 information Block size in region 2 = 0100h * 256 byte	64 Kbytes
35h 36h 37h 38h	6Ah 6Ch 6Eh 70h		0000h 0000h 0000h 0000h	Region 3 information Number of erase blocks of identical size=007Fh+1 Region 3 information Block size in region 3 = 0000h * 256 byte	0 0
39h 3Ah 3Bh 3Ch	72h 74h 76h 78h		0000h 0000h 0000h 0000h	Region 4 information Number of erase blocks of identical size=007Fh+1 Region 4 information Block size in region 4 = 0000h * 256 byte	0 0

1. For bottom boot devices, erase block region 1 is located from address 000000h to 007FFFh and erase block region 2 from address 008000h to 3FFFFFh.
For top boot devices, erase block region 1 is located from address 000000h to 3F7FFFh and erase block region 2 from address 3F8000h to 3FFFFFh.

Table 35. Primary algorithm-specific extended query table

Address x16	Address x8	Data	Description		Value
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) 00h = required, 01h = not required Silicon revision number (bits 7 to 2)		Yes
46h	8Ch	0002h	Erase Suspend 00h = not supported, 01h = Read only, 02 = Read and Write		2
47h	8Eh	0004h	Block Protection 00h = not supported, x = number of blocks per protection group		4
48h	90h	0001h	Temporary Block Unprotect 00h = not supported, 01h = supported		Yes
49h	92h	0004h	Block Protect /Unprotect		04
4Ah	94h	0000h	Simultaneous operations, 00h = not supported		No
4Bh	96h	0000h	Burst mode: 00h = not supported, 01h = supported		No
4Ch	98h	0001h	Page mode: 00h = not supported, 01h = 4 page word, 02h = 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		11.5 V
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		12.5 V
4Fh	9Eh	0002h 0003h 0004h 0005h	M29W640GB M29W640GT M29W640GL M29W640GH	Top/bottom boot block flag 02h = bottom boot device 03h = top boot device 04h = uniform blocks bottom $V_{PP}/\overline{WP}$ protect 05h = uniform blocks top $V_{PP}/\overline{WP}$ protect	–
50h	A0h	0001h	Program Suspend 00h = not supported 01h = supported		Supported

Table 36. 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 M29W640G 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 memory block verify code’ in *Table 7: Bus operations, BYTE = VIL* and *Table 8: Bus operations, BYTE = VIH*, 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 the *Section 4.3.1: Enter Extended Block command* and *Section 4.3.2: Exit Extended Block command*, and to *Table 10* and *Table 11: Commands, 8-bit mode, BYTE = VIL*.

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 37: 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*10 Section D.2: In-system technique* and to the corresponding flowcharts, *Figure 28* and *Figure 29*, 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*10*, *Section D.1: Programmer technique* and to the corresponding flowcharts, *Figure 26* and *Figure 27*, 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 37. Extended block address and data

Address		Data	
x8	**x16**	**Factory locked**	**Customer lockable**
000000h-00007Fh	000000h-00003Fh	Security identification number	Determined by customer
000080h-0000FFh	000040h-00007Fh	Unavailable	

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: Block addresses*, *Table 29* and *Table 30* 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, $\overline{RP}$; this is described in the *Section 2: Signal descriptions*.

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 26: 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 27: Programmer equipment chip unprotect flowchart*. *Table 38: Programmer technique bus operations, BYTE = VIH or VIL*, 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, $\overline{RP}$(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 28: 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 29: 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.

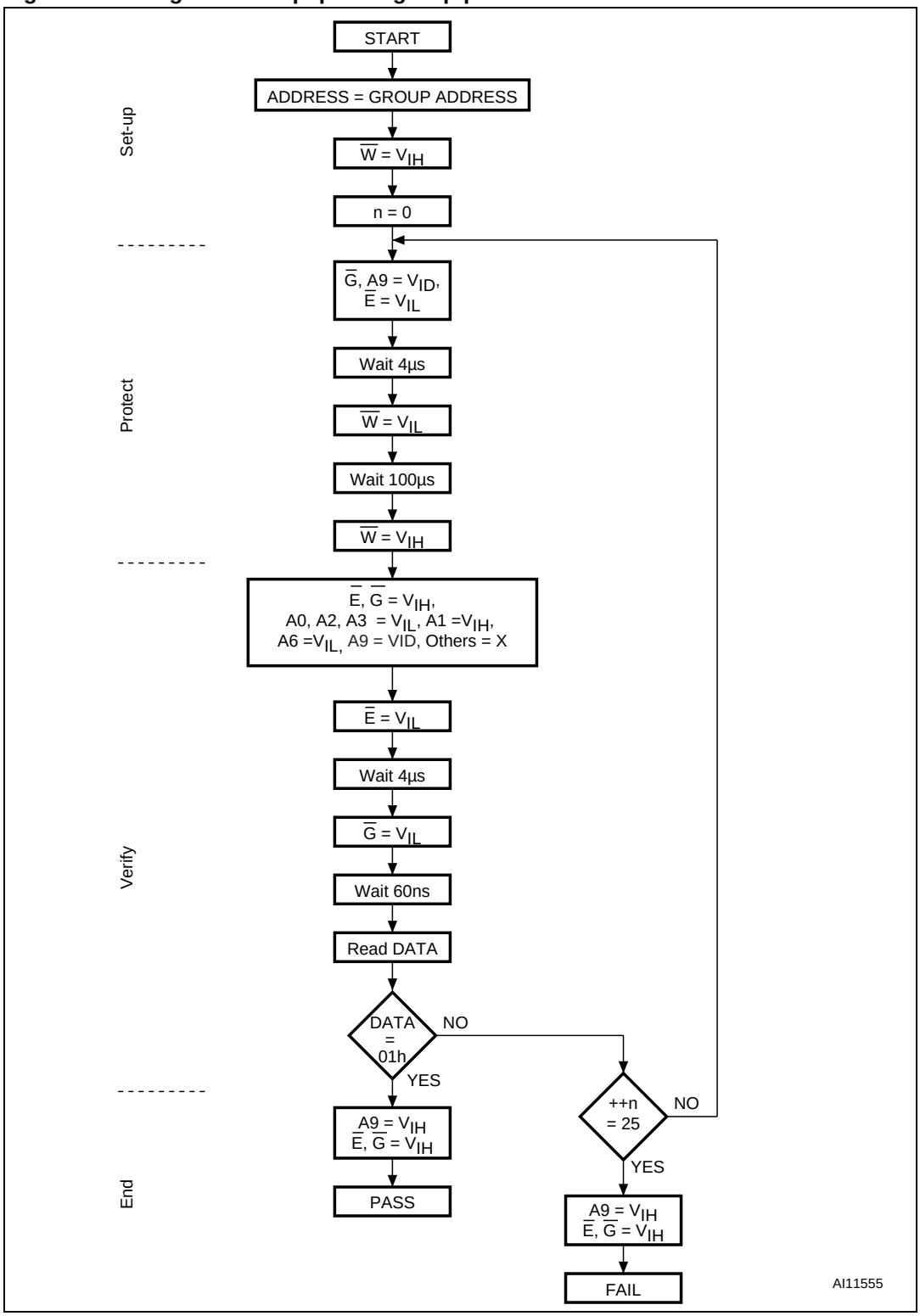
Note: $\overline{RP}$ can be either at V_{IH} or at V_{ID} when using the in-system technique to protect the extended block.

Table 38. Programmer technique bus operations, $\overline{BYTE}$ = V_{IH} or V_{IL}

Operation	$\overline{E}$	$\overline{G}$	$\overline{W}$	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) Protection Verify	V_{IL}	V_{IL}	V_{IH}	A0, A2, A3 = V_{IL}, A1 = V_{IH}, A6 = V_{IL}, A9 = V_{ID}, A12-A21 = Block Address Others = X	Pass = XX01h Retry = XX00h
Block (Group) Unprotection Verify	V_{IL}	V_{IL}	V_{IH}	A0, A2, A3 = V_{IL}, A1 = V_{IH}, A6 = V_{IH}, A9 = V_{ID}, A12-A21 = Block Address Others = X	Retry = XX01h Pass = XX00h

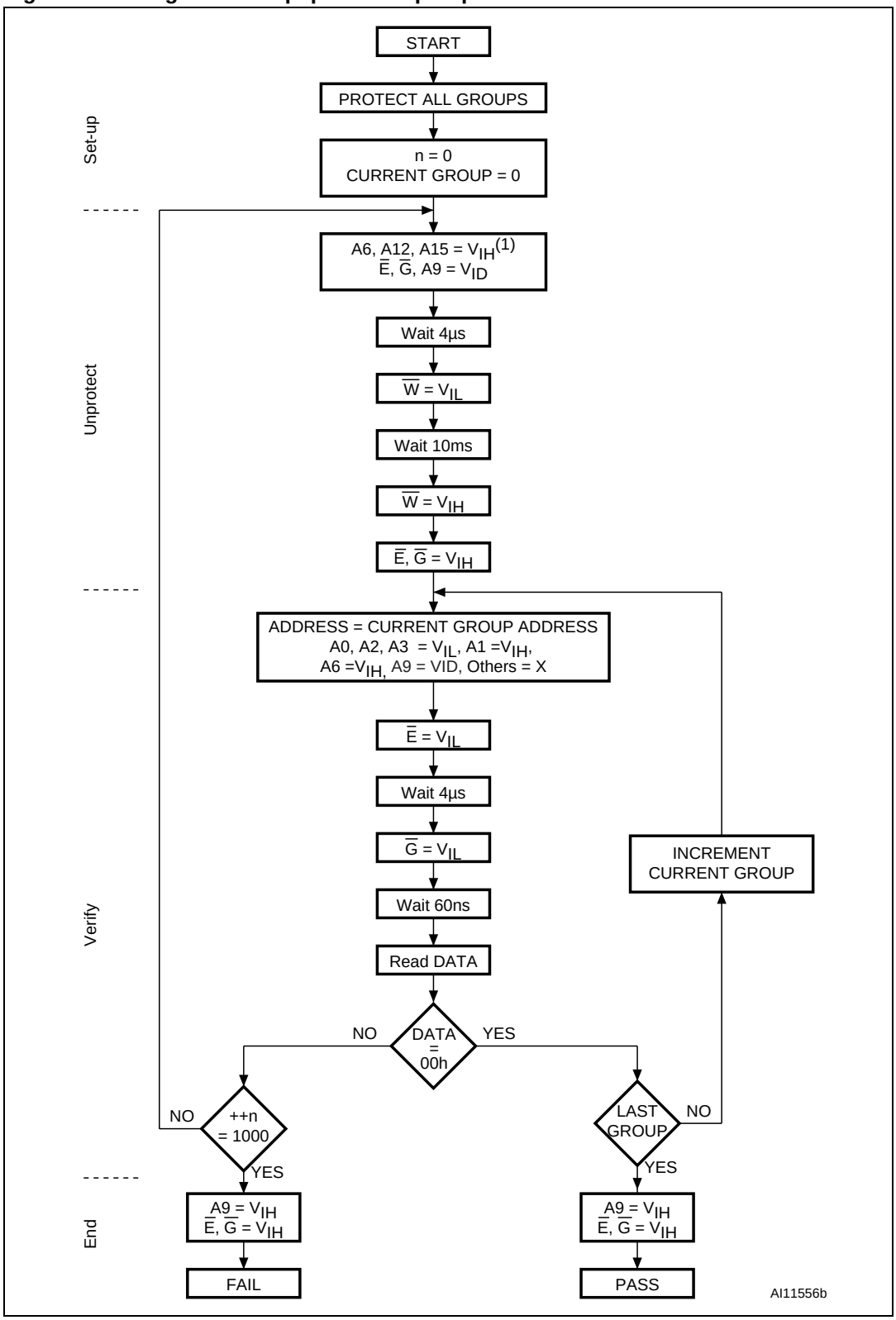
1. Block protection groups are shown in *Appendix A*, Tables *29* and *30*.

Figure 26. Programmer equipment group protect flowchart



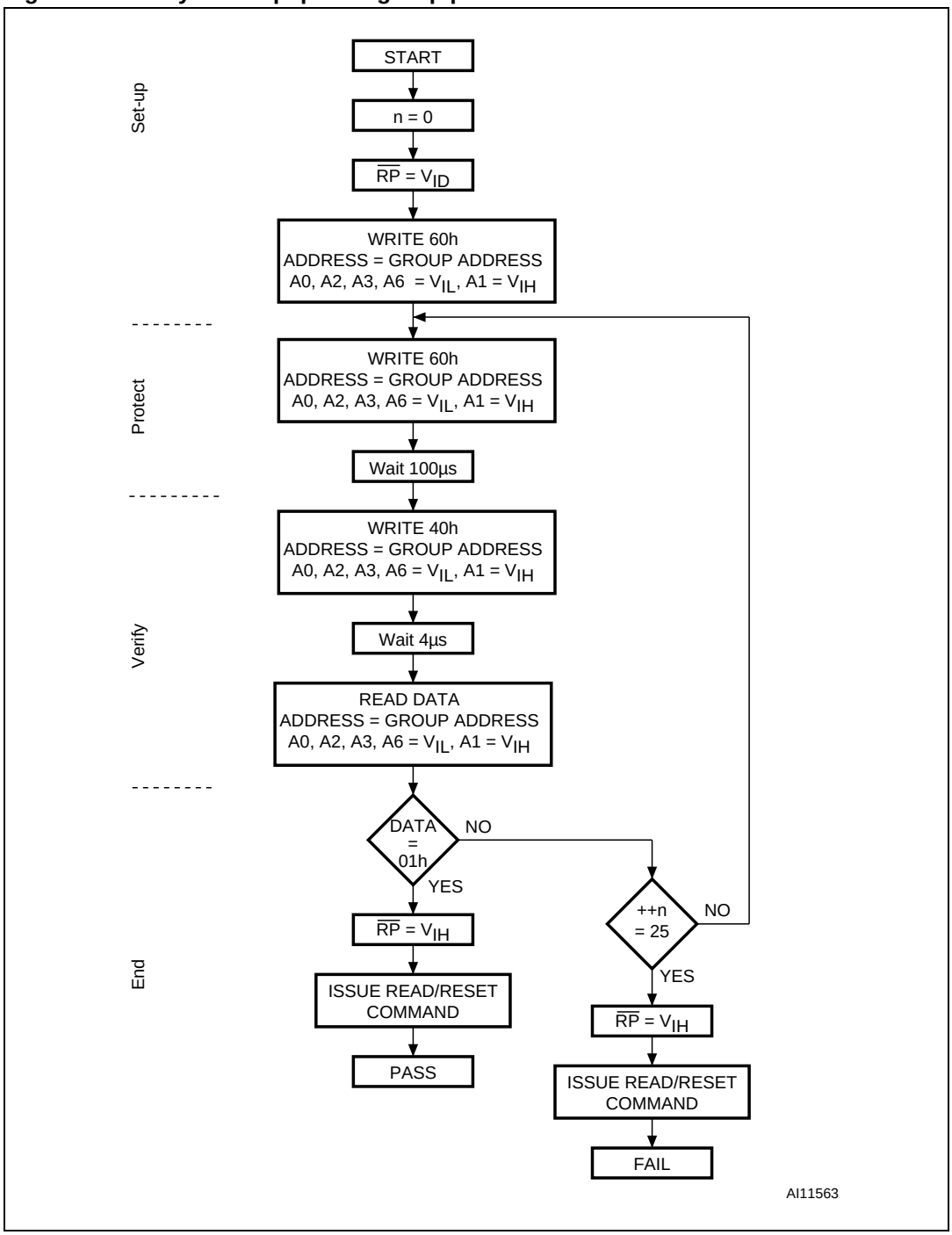
1. Block protection groups are shown in *Appendix A*, Tables *29* and *30*.

Figure 27. Programmer equipment chip unprotect flowchart



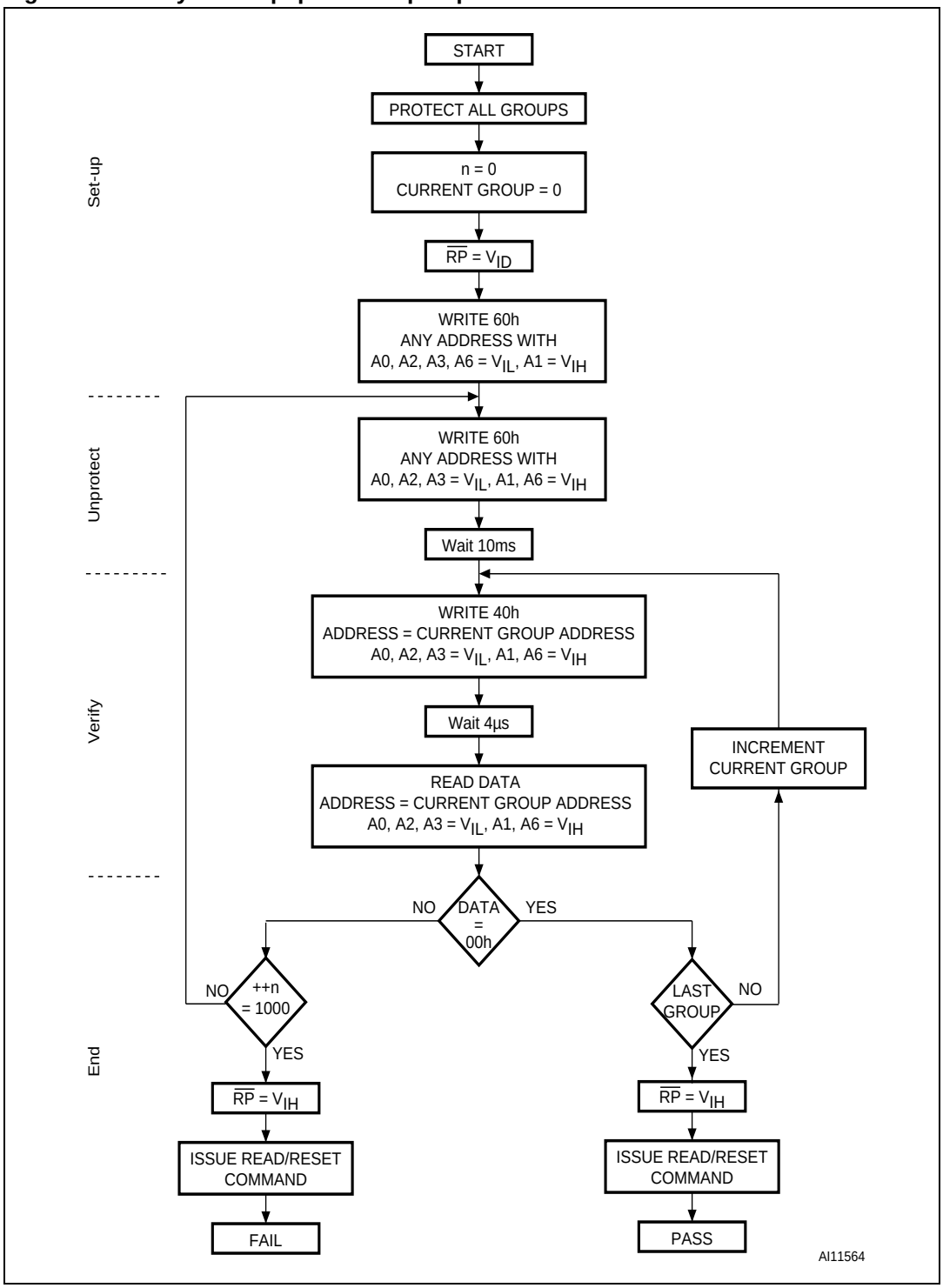
1. Block protection groups are shown in *Appendix A*, Tables *29* and *30*.

Figure 28. In-system equipment group protect flowchart



2. Block protection groups are shown in *Appendix A*, Tables *29* and *30*.
3. $\overline{RP}$ can be either at V_{IH} or at V_{ID} when using the In-system technique to protect the extended block.

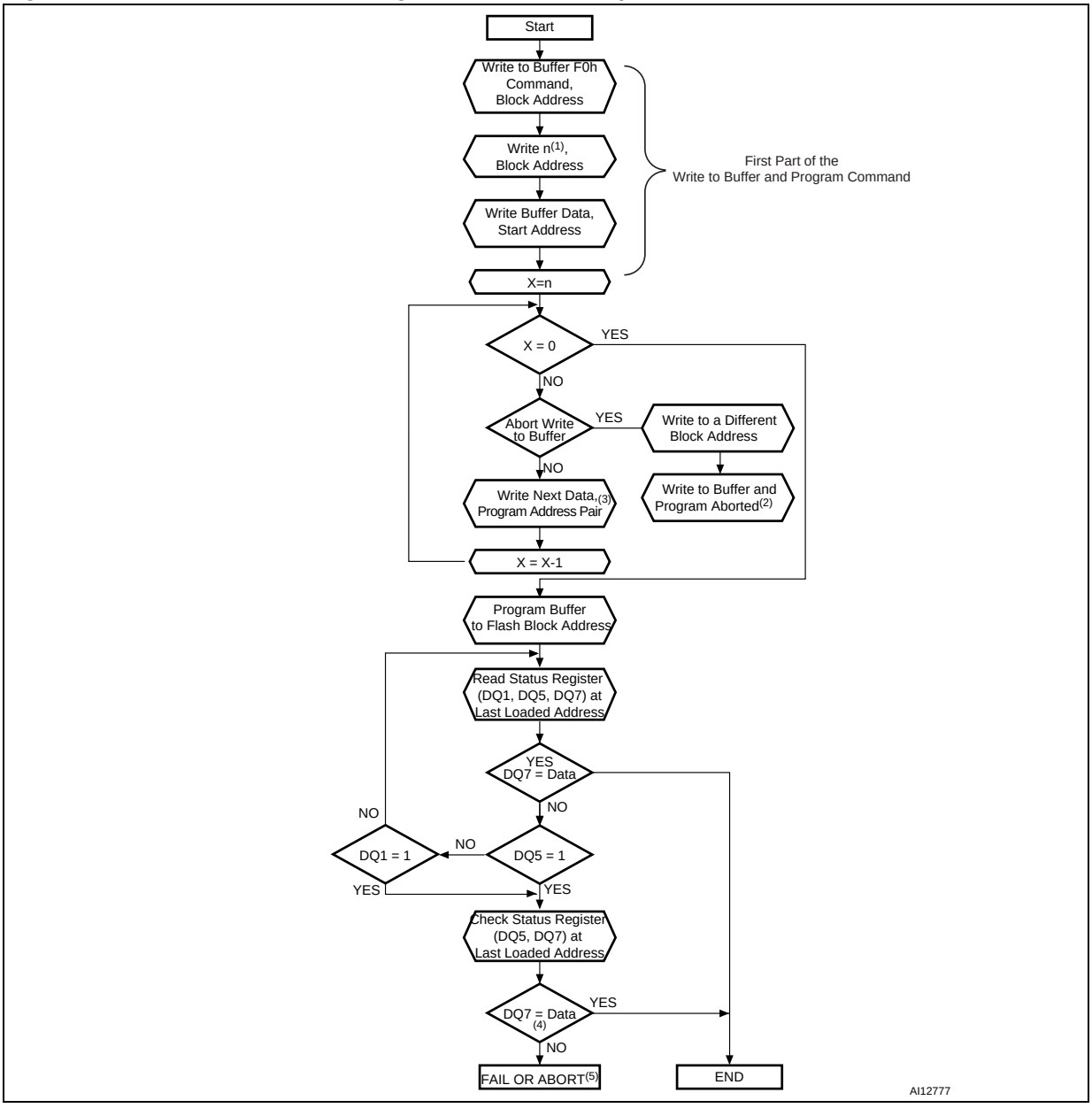
Figure 29. In-system equipment chip unprotect flowchart



1. Block protection groups are shown in *Appendix A*, Tables *29* and *30*.

Appendix E Flowchart

Figure 30. Write to Buffer and Program flowchart and pseudocode



1. n+1 is the number of addresses to be programmed.
2. A Write to Buffer and Program Abort and Reset must be issued to return the device in read mode.
3. When the block address is specified, any address in the selected block address space is acceptable. However when loading write buffer address with data, all addresses must fall within the selected write buffer page.
4. DQ7 must be checked since DQ5 and DQ7 may change simultaneously.
5. If this flowchart location is reached because DQ5='1', then the Write to Buffer and Program command failed. If this flowchart location is reached because DQ1='1', then the Write to Buffer and Program command aborted. In both cases, the appropriate reset command must be issued to return the device in read mode: a Reset command if the operation failed, a Write to Buffer and Program Abort and Reset command if the operation aborted.
6. See *Table 10* and *Table 11*, for details on Write to Buffer and Program command sequence.

10 Revision history

Table 39. Document revision history

Date	Version	Changes
20-Jul-2006	1	Initial release.
21-Aug-2006	2	Datasheet status updated to full datasheet; added an explanation of how to abort the Write Buffer Programming Sequence in *Section 4.2.9: Write to Buffer and Program command*; amended text of *4.2.11: Write to Buffer and Program Abort and Reset command*.
25-Oct-2006	3	*Table 13: Status Register bits* updated.
22-Feb-2007	4	90 ns access time added.
27-Mar-2008	5	Applied Numonyx branding.
09-Jun-2008	6	Updated: *Section 1: Description*, *Section 2.8: VPP/Write Protect (VPP/WP)*, *Section 2.9: Reset/Block Temporary Unprotect (RP)*, and *Table 6: Hardware protection*. Minor text changes.
16-Dec-2008	7	Added the following: – To cover page, bullet stating: Automotive Certified Parts Available for Version M29W640GT/M29W640GB. – To *Table 27.: Ordering information scheme*, under “Speed”: 7A = 70 ns Automotive -40C to 85C Certified Part.
11-March-2009	8	Added FBGA (ZS) package information.
26-March-2009	9	Updated CFI addresses 20h and 24h with the correct timeout value for write buffer program. Devices with date code 913 (ww13-2009) or later include this CFI correction.
20-July-2009	10	Corrected tPHWL from 200 to 50 in *Table 20.: Reset/Block Temporary Unprotect AC characteristics*.
8-October-2009	11	Made the following revision to cover page: – Removed specific part numbers from the bullet, Automotive Certified Parts Available. Made the following revision to Order Information: – Add temperature range, option: 3 = -40 °C to 125 °C – Added to speed: 7A = 70 ns Automotive -40 °C to 85 °C Certified Part, used in conjiuction with temperature range option 6 to distinguish the Automotive Part"

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.

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Flash Memory ***category:***

Click to view products by Micron ***manufacturer:***

Other Similar products are found below :

MBM29F200TC-70PFTN-SFLE1 MBM29F400BC-70PFTN-SFLE1 MBM29F800BA-90PF-SFLE1 8 611 200 906 9990933135 AM29F200BB-90DPI 1 AT25DF021A-MHN-Y AT25DF256-SSHN-T EAN62691701 N25Q512A83G1240F P520366230636 8 905 959 076T 8 905 959 252 8 925 850 296 260332-002 04 S29AL008J55BFIR20 S29AL008J55TFIR23 S29AL008J70BFI010 S29AL008J70BFI013 S29AL032D90TFA040 S29AS016J70BHIF40 S29GL064N90TFI013 S29PL064J55BFI120 S76MSA90222AHD000 S99AL016D0019 9990932415 A2C53026990 SST39VF400A-70-4I-MAQE AM29F400BB-55SF0 AM29F400BB-55SI MBM29F400BC-90PFVGTSFLE1 MBM29F800BA-70PFTN-SFLE1 MBM29F800TA-90PFCN-SFLE1 AT25DF011-MAHN-T AT25DN011-MAHF-T AT45DQ161-SHFHB-T RP-SDCCTH0 S29AL016J70TFN013 S29CD016J0MQFM110 S29GL032N90BFI042 S29GL032N90FAI033 S29GL064N90TFI023 S29GL128S10GHIV20 S29PL127J70BAI020 S34ML01G200GHI000 S34ML02G200TFI003 S34MS02G200BHI000 S34MS02G200TFI000 S71VS256RC0AHK4L0 AT25SF041-MHD-T