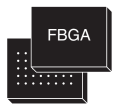
numonyx

M58WR064HU
M58WR064HL

64 Mbit (4Mb x16, Mux I/O, Multiple Bank, Burst) 1.8V supply Flash memories

Feature summary

- Supply voltage
  - V_{DD} = 1.7V to 2V for Program, Erase and Read
  - V_{DDQ} = 1.7V to 2V for I/O Buffers
  - V_{PP} = 12V for fast Program (9V tolerant)
- Multiplexed address/data
- Synchronous / asynchronous read
  - Synchronous Burst Read mode: 66MHz
  - Random Access: 70ns
- Synchronous burst read suspend
- Programming time
  - 8µs by Word typical for Fast Factory Program
  - Double/Quadruple Word Program option
  - Enhanced Factory Program options
- Memory blocks
  - Multiple Bank Memory Array: 4 Mbit Banks
  - Parameter Blocks (Top or Bottom location)
- Dual operations
  - Program Erase in one Bank while Read in others
  - No delay between Read and Write operations
- Block locking
  - All blocks locked at Power up
  - Any combination of blocks can be locked
  - $\overline{WP}$ for Block Lock-Down
- Security
  - 128 bit user programmable OTP cells
  - 64 bit unique device number
- Common Flash Interface (CFI)
- 100,000 program/erase cycles per block



VFBGA44
7.7 x 9mm (ZB)
7.5 × 5mm (ZA)

- Electronic signature
  - Manufacturer Code: 20h
  - Top Device Code, M58WR064HU: 88C0h
  - Bottom Device Code, M58WR064HL: 88C1h
- Package
  - ECOPACK®

Contents

List of tables

List of figures

1 Summary description

The M58WR064HU/L are 64 Mbit (4 Mbit x16) non-volatile Flash memories. They may be erased electrically at block level and programmed in-system on a Word-by-Word basis using a 1.7V to 2V V_{DD} supply for the circuitry and a 1.7V to 2V V_{DDQ} supply for the Input/Output pins. An optional 12V (9V tolerant) V_{PP} power supply is provided to speed up customer programming.

The first sixteen address lines are multiplexed with the Data Input/Output signals on the multiplexed address/data bus ADQ0-ADQ15. The remaining address lines A16-A21 are the Most Significant Bit addresses.

The device features an asymmetrical block architecture.

The M58WR064HU and M58WR064HL have an array of 135 blocks, and are divided into 4 Mbit banks. There are 15 banks each containing 8 main blocks of 32 KWords, and one parameter bank containing 8 parameter blocks of 4 KWords and 7 main blocks of 32 KWords.

The Multiple Bank Architecture allows Dual Operations, while programming or erasing in one bank, Read operations are possible in other banks. Only one bank at a time is allowed to be in Program or Erase mode. It is possible to perform burst reads that cross bank boundaries. The bank architecture is summarized in *Table 2*, and the memory maps are shown in *Figure 3* The Parameter Blocks are located at the top of the memory address space for the M58WR064HU, and at the bottom for the M58WR064HL.

Each block can be erased separately. Erase can be suspended, in order to perform program in any other block, and then resumed. Program can be suspended to read data in any other block and then resumed. Each block can be programmed and erased over 100,000 cycles using the supply voltage V_{DD}. There are two Enhanced Factory programming commands available to speed up programming.

Program and Erase commands are written to the Command Interface of the memory. An internal Program/Erase Controller takes care of the timings necessary for program and erase operations. The end of a program or erase operation can be detected and any error conditions identified in the Status Register. The command set required to control the memory is consistent with JEDEC standards.

The device supports synchronous burst read and asynchronous read from all blocks of the memory array; at power-up the device is configured for asynchronous read. In synchronous burst mode, data is output on each clock cycle at frequencies of up to 66MHz. The synchronous burst read operation can be suspended and resumed.

The device features an Automatic Standby mode. When the bus is inactive during Asynchronous Read operations, the device automatically switches to the Automatic Standby mode. In this condition the power consumption is reduced to the standby value I_{DD4} and the outputs are still driven.

The M58WR064HU/L feature an instant, individual block locking scheme that allows any block to be locked or unlocked with no latency, enabling instant code and data protection. All blocks have three levels of protection. They can be locked and locked-down individually preventing any accidental programming or erasure. There is an additional hardware protection against program and erase. When $V_{PP} \leq V_{PPLK}$ all blocks are protected against program or erase. All blocks are locked at Power- Up.

The device includes a Protection Register to increase the protection of a system's design. The Protection Register is divided into two segments: a 64 bit segment containing a unique device number written by Numonyx, and a 128 bit segment One-Time-Programmable (OTP) by the user. The user programmable segment can be permanently protected. *Figure 4*, shows the Protection Register Memory Map.

The M58WR064HU/L is available in VFBGA44, 7.7 x 9mm and VFBGA44 7.5 × 5, 10x4 active ball array, 0.5mm pitch packages.

The memories are supplied with all the bits erased (set to '1').

Figure 1. Logic diagram

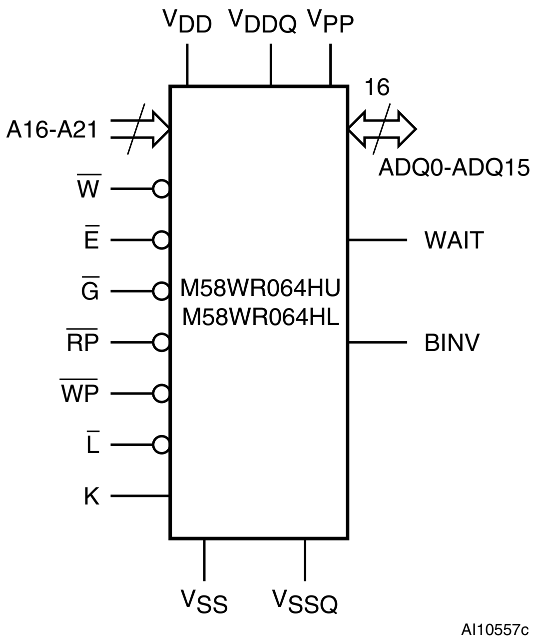


Table 1. Signal names

A16-A21	Address Inputs
ADQ0-ADQ15	Data Input/Outputs or Address Inputs, Command Inputs
$\overline{E}$	Chip Enable
$\overline{G}$	Output Enable
$\overline{W}$	Write Enable
$\overline{RP}$	Reset/Power-down
$\overline{WP}$	Write Protect
K	Clock
$\overline{L}$	Latch Enable
WAIT	Wait
BINV	Bus Invert
V_{DD}	Supply Voltage
V_{DDQ}	Supply Voltage for Input/Output Buffers
V_{PP}	Optional Supply Voltage for Fast Program & Erase
V_{SS}	Ground
V_{SSQ}	Ground Input/Output Supply
NC	Not Connected Internally

Figure 2. VFBGA connections (top view through package)

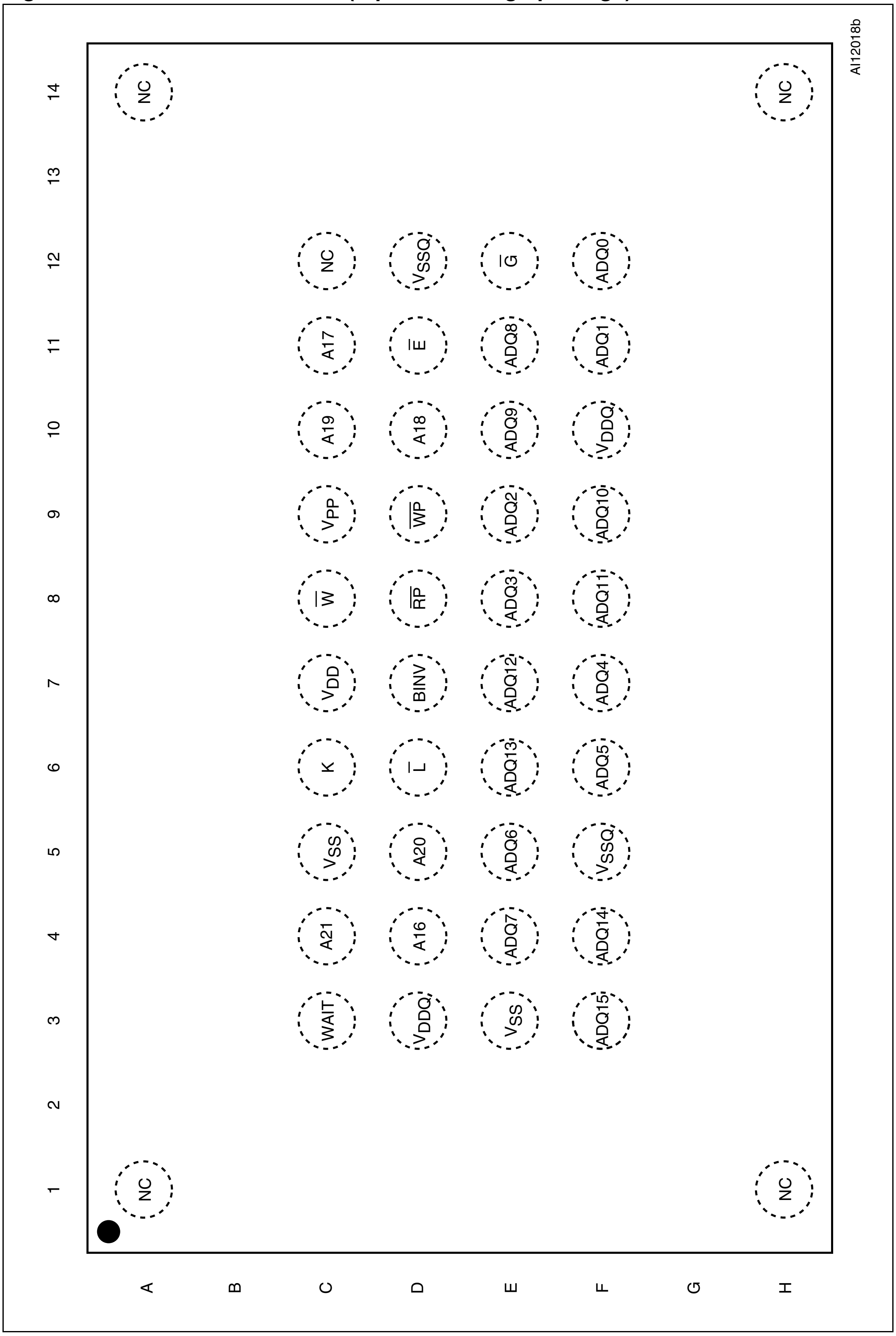
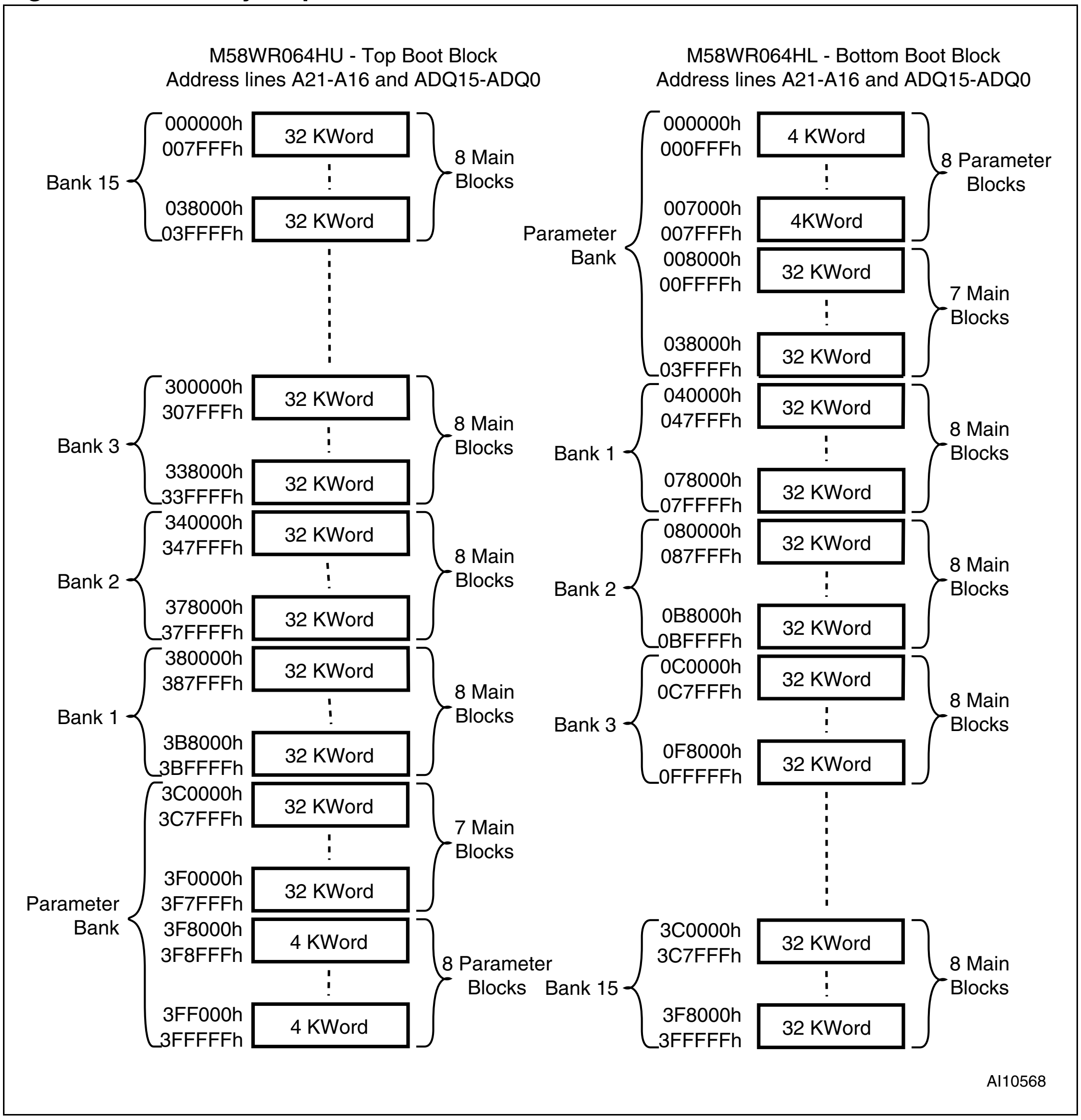


Table 2. Bank architecture

Number	Bank Size	Parameter Blocks	Main Blocks
Parameter Bank	4 Mbit	8 blocks of 4 KWord	7 blocks of 32 KWord
Bank 1	4 Mbit	-	8 blocks of 32 KWord
Bank 2	4 Mbit	-	8 blocks of 32 KWord
Bank 3	4 Mbit	-	8 blocks of 32 KWord
⋮	⋮	⋮	⋮
Bank 14	4 Mbit	-	8 blocks of 32 KWord
Bank 15	4 Mbit	-	8 blocks of 32 KWord

Figure 3. Memory map



2 Signal descriptions

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

2.1 Address Inputs (ADQ0-ADQ15, A16-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 Input/Output (ADQ0-ADQ15)

The Data I/O outputs the data stored at the selected address during a Bus Read operation or inputs a command or the data to be programmed during a Bus Write operation.

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

The Chip Enable input activates the memory control logic, input buffers, decoders and sense amplifiers. When Chip Enable is at V_{IL} and Reset is at V_{IH} the device is in active mode. When Chip Enable is at V_{IH} the memory is deselected, the outputs are high impedance and the power consumption is reduced to the stand-by level.

2.4 Output Enable ($\overline{G}$)

The Output Enable controls data outputs during the Bus Read operation of the memory.

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

The Write Enable controls the Bus Write operation of the memory's Command Interface. The data is latched on the rising edge of Chip Enable or Write Enable whichever occurs first.

2.6 Write Protect ($\overline{WP}$)

Write Protect is an input that gives an additional hardware protection for each block. When Write Protect is at V_{IL}, the Lock-Down is enabled and the protection status of the Locked-Down blocks cannot be changed. When Write Protect is at V_{IH}, the Lock-Down is disabled and the Locked-Down blocks can be locked or unlocked. (refer to *Table 15: Lock status*).

2.7 Reset/Power-Down ($\overline{RP}$)

The Reset/Power-Down input provides a hardware reset of the memory, and/or power-down functions, depending on the settings in the Configuration Register. When Reset/Power-Down is at V_{IL}, the memory is in reset mode: the outputs are high impedance and the current consumption is reduced to the Standby Supply Current I_{DD3}, or to the Reset/Power-Down Supply Current I_{DD2} if the Power-Down function is enabled. Refer to *Table 20: DC Characteristics - Currents*, for the value of I_{DD2} and I_{DD3}. After reset all blocks are in the Locked state and the bits of the Configuration Register are reset except for Power-Down bit CR5. When Reset/Power-Down is at V_{IH}, the device is in normal operation. Exiting reset mode the device enters Asynchronous Read mode, but a negative transition of Chip Enable or Latch Enable is required to ensure valid data outputs.

The Reset pin can be interfaced with 3V logic without any additional circuitry. It can be tied to V_{RPH} (refer to *Table 21: DC characteristics - voltages*).

2.8 Latch Enable ($\overline{L}$)

Latch Enable latches the ADQ0-ADQ15 and A16-A21 address bits on its rising edge. The address latch is transparent when Latch Enable is at V_{IL} and it is inhibited when Latch Enable is at V_{IH}.

2.9 Clock (K)

The clock input synchronizes the memory to the microcontroller during synchronous read operations; the address is latched on a Clock edge (rising or falling, according to the configuration settings) when Latch Enable is at V_{IL}. Clock is don't care during asynchronous read and in write operations.

2.10 Wait (WAIT)

Wait is an output signal used during synchronous read to indicate whether the data on the output bus are valid. This output is high impedance when Chip Enable is at V_{IH} or Reset is at V_{IL}. It can be configured to be active during the wait cycle or one clock cycle in advance. The WAIT signal is forced deasserted when Output Enable is at V_{IH}.

2.11 Bus Invert (BINV)

Bus invert is an input/output signal used to reduce the amount of power required to switch the external address/data bus. Power is saved by inverting the data on ADQ0-ADQ15 each time the inversion results in a reduced number of pin transitions. Data is inverted when BINV is at V_{IH} (i.e. if the data is AAAAh and BINV is at V_{IH}, AAAAh becomes 5555h). BINV is high impedance when Chip Enable or Output Enable is at V_{IH} or when Reset/Power Down is at V_{IL}.

2.12 V_{DD} Supply Voltage

V_{DD} provides the power supply to the internal core of the memory device. It is the main power supply for all operations (Read, Program and Erase).

2.13 V_{DDQ} Supply Voltage

V_{DDQ} provides the power supply to the I/O pins and enables all Outputs to be powered independently from V_{DD}. V_{DDQ} can be tied to V_{DD} or can use a separate supply.

2.14 V_{PP} Program Supply Voltage

V_{PP} is both a control input and a power supply pin. The two functions are selected by the voltage range applied to the pin.

If V_{PP} is kept in a low voltage range (0V to V_{DDQ}) V_{PP} is seen as a control input. In this case a voltage lower than V_{PPLK} gives an absolute protection against program or erase, while V_{PP} in the V_{PP1} range enables these functions (see Tables 20 and 21, DC Characteristics for the relevant values). V_{PP} is only sampled at the beginning of a program or erase; a change in its value after the operation has started does not have any effect and program or erase operations continue.

If V_{PP} is in the range of V_{PPH} it acts as a power supply pin. In this condition V_{PP} must be stable until the Program/Erase algorithm is completed.

2.15 V_{SS} Ground

V_{SS} ground is the reference for the core supply. It must be connected to the system ground.

2.16 V_{SSQ} Ground

V_{SSQ} ground is the reference for the input/output circuitry driven by V_{DDQ}. V_{SSQ} must be connected to V_{SS}.

Note: Each device in a system should have V_{DD}, V_{DDQ} and V_{PP} decoupled with a 0.1µF ceramic capacitor close to the pin (high frequency, inherently low inductance capacitors should be as close as possible to the package). See Figure 8: AC measurement load circuit. The PCB track widths should be sufficient to carry the required V_{PP} program and erase currents.

3 Bus operations

There are six standard bus operations that control the device. These are Bus Read, Bus Write, Address Latch, Output Disable, Standby and Reset. See *Table 3: Bus operations*, for a summary.

Typically glitches of less than 5ns on Chip Enable or Write Enable are ignored by the memory and do not affect Bus Write operations.

3.1 Bus Read

Bus Read operations are used to output the contents of the Memory Array, the Electronic Signature, the Status Register and the Common Flash Interface. Both Chip Enable and Output Enable must be at V_{IL} in order to perform a read operation. The Chip Enable input should be used to enable the device. Output Enable should be used to gate data onto the output. The data read depends on the previous command written to the memory (see *Section 4: Command interface*). See Figures *9*, *10* and *11* Read AC Waveforms, and Tables *22* and *23* Read AC Characteristics, for details of when the output becomes valid.

3.2 Bus Write

Bus Write operations write Commands to the memory or latch Input Data to be programmed. A bus write operation is initiated when Chip Enable and Write Enable are at V_{IL} with Output Enable at V_{IH}. Commands and Input Data are latched on the rising edge of Write Enable or Chip Enable, whichever occurs first. The addresses must also be latched prior to the write operation by toggling Latch Enable (when Chip Enable is at V_{IL}). The Latch Enable must be tied to V_{IH} during the bus write operation.

See Figures *14* and *15*, Write AC Waveforms, and Tables *24* and *25*, Write AC Characteristics, for details of the timing requirements.

3.3 Address Latch

Address latch operations input valid addresses. Both Chip enable and Latch Enable must be at V_{IL} during address latch operations. The addresses are latched on the rising edge of Latch Enable.

3.4 Output Disable

The outputs are high impedance when the Output Enable is at V_{IH}.

3.5 Standby

Standby disables most of the internal circuitry allowing a substantial reduction of the current consumption. The memory is in stand-by when Chip Enable and Reset are at V_{IH}. The power consumption is reduced to the stand-by level and the outputs are set to high impedance, independently from the Output Enable or Write Enable inputs. If Chip Enable switches to V_{IH} during a program or erase operation, the device enters Standby mode when finished.

3.6 Reset/Power-Down

During reset mode the memory is deselected and the outputs are high impedance. The memory is in reset mode when Reset/Power-Down is at V_{IL}. The power consumption is reduced to the Standby level, or to the Reset/Power-Down level if the Power-Down function is enabled, independently of the Chip Enable, Output Enable or Write Enable inputs. If Reset/Power-Down is pulled to V_{SS} during a Program or Erase, this operation is aborted and the memory content is no longer valid.

Table 3. Bus operations

Operation(1)	$\overline{E}$	$\overline{G}$	$\overline{W}$	$\overline{L}$	$\overline{RP}$	WAIT(2)	ADQ15-ADQ0
Bus Read	V_{IL}	V_{IL}	V_{IH}	V_{IH}	V_{IH}		Data Output
Bus Write	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{IH}		Data Input
Address Latch	V_{IL}	V_{IH}	x	V_{IL}	V_{IH}		Address Input
Output Disable	V_{IL}	V_{IH}	V_{IH}	V_{IH}	V_{IH}		Hi-Z
Standby	V_{IH}	x	x	x	V_{IH}	Hi-Z	Hi-Z
Reset/Power-Down	x	x	x	x	V_{IL}	Hi-Z	Hi-Z

1. x = Don't care.
2. WAIT signal polarity is configured using the Set Configuration Register command.

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. An internal Program/Erase Controller handles all timings and verifies the correct execution of the Program and Erase commands. The Program/Erase Controller provides a Status Register whose output may be read at any time to monitor the progress or the result of the operation.

The Command Interface is reset to read mode when power is first applied, when exiting from Reset or whenever V_{DD} is lower than V_{LKO}. Command sequences must be followed exactly. Any invalid combination of commands will reset the device to read mode.

Refer to *Table 4: Command codes*, and *Appendix D*, Tables *42*, *43*, *44* and *45*, Command Interface States - Modify and Lock Tables, for a summary of the Command Interface.

The Command Interface is split into two types of commands: Standard commands and Factory Program commands. The following sections explain in detail how to perform each command.

Table 4. Command codes

Hex Code	Command
01h	Block Lock Confirm
03h	Set Configuration Register Confirm
10h	Alternative Program Setup
20h	Block Erase Setup
2Fh	Block Lock-Down Confirm
30h	Enhanced Factory Program Setup
35h	Double Word Program Setup
40h	Program Setup
50h	Clear Status Register
56h	Quadruple Word Program Setup
60h	Block Lock Setup, Block Unlock Setup, Block Lock Down Setup and Set Configuration Register Setup
70h	Read Status Register
75h	Quadruple Enhanced Factory Program Setup
80h	Bank Erase Setup
90h	Read Electronic Signature
98h	Read CFI Query
B0h	Program/Erase Suspend
C0h	Protection Register Program
D0h	Program/Erase Resume, Block Erase Confirm, Bank Erase Confirm, Block Unlock Confirm or Enhanced Factory Program Confirm
FFh	Read Array

5 Command interface - Standard commands

The following commands are the basic commands used to read, write to and configure the device. Refer to *Table 5: Standard commands*, in conjunction with the following text descriptions.

5.1 Read Array command

The Read Array command returns the addressed bank to Read Array mode. One Bus Write cycle is required to issue the Read Array command and return the addressed bank to Read Array mode. Subsequent read operations will read the addressed location and output the data. A Read Array command can be issued in one bank while programming or erasing in another bank. However if a Read Array command is issued to a bank currently executing a Program or Erase operation the command will be executed but the output data is not guaranteed.

5.2 Read Status Register command

The Status Register indicates when a Program or Erase operation is complete and the success or failure of operation itself. Issue a Read Status Register command to read the Status Register content. The Read Status Register command can be issued at any time, even during Program or Erase operations.

The following read operations output the content of the Status Register of the addressed bank. The Status Register is latched on the falling edge of $\overline{E}$ or $\overline{G}$ signals, and can be read until $\overline{E}$ or $\overline{G}$ returns to V_{IH}. Either $\overline{E}$ or $\overline{G}$ must be toggled to update the latched data. See *Table 8* for the description of the Status Register Bits. This mode supports asynchronous or single synchronous reads only.

5.3 Read Electronic Signature command

The Read Electronic Signature command reads the Manufacturer and Device Codes, the Block Locking Status, the Protection Register, and the Configuration Register.

The Read Electronic Signature command consists of one write cycle to an address within one of the banks. A subsequent Read operation in the same bank will output the Manufacturer Code, the Device Code, the protection Status of the blocks in the targeted bank, the Protection Register, or the Configuration Register (see *Table 6*).

Dual operations between the Parameter bank and the Electronic Signature location are not allowed (see *Table 14: Dual operation limitations*).

If a Read Electronic Signature command is issued in a bank that is executing a Program or Erase operation the bank will go into Read Electronic Signature mode, subsequent Bus Read cycles will output the Electronic Signature data and the Program/Erase controller will continue to program or erase in the background. This mode supports asynchronous or single synchronous reads only, it does not support synchronous burst reads.

5.4 Read CFI Query command

The Read CFI Query command is used to read data from the Common Flash Interface (CFI). The Read CFI Query Command consists of one Bus Write cycle, to an address within one of the banks. Once the command is issued subsequent Bus Read operations in the same bank read from the Common Flash Interface.

If a Read CFI Query command is issued in a bank that is executing a Program or Erase operation the bank will go into Read CFI Query mode, subsequent Bus Read cycles will output the CFI data and the Program/Erase controller will continue to Program or Erase in the background. This mode supports asynchronous or single synchronous reads only, it does not support synchronous burst reads.

The status of the other banks is not affected by the command (see *Table 12*). After issuing a Read CFI Query command, a Read Array command should be issued to the addressed bank to return the bank to Read Array mode.

Dual operations between the Parameter Bank and the CFI memory space are not allowed (see *Table 14: Dual operation limitations*).

See *Appendix B: Common Flash Interface*, Tables *32*, *33*, *34*, *35*, *36*, *37*, *38*, *39*, *40* and *41* for details on the information contained in the Common Flash Interface memory area.

5.5 Clear Status Register command

The Clear Status Register command can be used to reset (set to ‘0’) error bits SR1, SR3, SR4 and SR5 in the Status Register. One bus write cycle is required to issue the Clear Status Register command. After the Clear Status Register command the bank returns to read mode.

The error bits in the Status Register do not automatically return to ‘0’ when a new command is issued. The error bits in the Status Register should be cleared before attempting a new Program or Erase command.

5.6 Block Erase command

The Block Erase command can be used to erase a block. It sets all the bits within the selected block to '1'. All previous data in the block is lost. If the block is protected then the Erase operation will abort, the data in the block will not be changed and the Status Register will output the error. The Block Erase command can be issued at any moment, regardless of whether the block has been programmed or not.

Two Bus Write cycles are required to issue the command.

- The first bus cycle sets up the Erase command.
- The second latches the block address to the Program/Erase Controller and starts it.

If the second bus cycle is not Write Erase Confirm (D0h), Status Register bits SR4 and SR5 are set and the command aborts. Erase aborts if Reset turns to V_{IL}. As data integrity cannot be guaranteed when the Erase operation is aborted, the block must be erased again.

Once the command is issued the device outputs the Status Register data when any address within the bank is read. At the end of the operation the bank will remain in Read Status Register mode until a Read Array, Read CFI Query or Read Electronic Signature command is issued.

During Erase operations the bank containing the block being erased will only accept the Read Array, Read Status Register, Read Electronic Signature, Read CFI Query and the Program/Erase Suspend command, all other commands will be ignored. Refer to *Section 10: Dual operations and multiple bank architecture* for detailed information about simultaneous operations allowed in banks not being erased. Typical Erase times are given in *Table 16: Program, erase times and endurance cycles*.
See *Appendix C*, *Figure 25: Block Erase flowchart and pseudo code*, for a suggested flowchart for using the Block Erase command.

5.7 Program command

The memory array can be programmed word-by-word. Only one Word in one bank can be programmed at any one time. If the block being programmed is protected then the Program operation will abort, the data in the block will not be changed and the Status Register will output the error.

Two bus write cycles are required to issue the Program Command.

- The first bus cycle sets up the Program command.
- The second latches the Address and the Data to be written and starts the Program/Erase Controller.

After programming has started, read operations in the bank being programmed output the Status Register content.

During Program operations the bank being programmed will only accept the Read Array, Read Status Register, Read Electronic Signature, Read CFI Query and the Program/Erase Suspend command. Refer to *Section 10: Dual operations and multiple bank architecture* for detailed information about simultaneous operations allowed in banks not being programmed. Typical Program times are given in *Table 16: Program, erase times and endurance cycles*.

Programming aborts if Reset goes to V_{IL}. As data integrity cannot be guaranteed when the program operation is aborted, the memory location must be reprogrammed.
See *Appendix C*, *Figure 21: Program flowchart and pseudo code*, for the flowchart for using the Program command.

5.8 Program/Erase Suspend command

The Program/Erase Suspend command is used to pause a Program or Block Erase operation. A Bank Erase operation cannot be suspended.

One bus write cycle is required to issue the Program/Erase command. Once the Program/Erase Controller has paused bits SR7, SR6 and/ or SR2 of the Status Register will be set to '1'. The command can be addressed to any bank.

During Program/Erase Suspend the Command Interface will accept the Program/Erase Resume, Read Array (cannot read the suspended block), Read Status Register, Read Electronic Signature and Read CFI Query commands. Additionally, if the suspend operation was Erase then the Clear status Register, Program, Block Lock, Block Lock-Down or Block Unlock commands will also be accepted. The block being erased may be protected by issuing the Block Lock, Block Lock-Down or Protection Register Program commands. Only the blocks not being erased may be read or programmed correctly. When the Program/Erase Resume command is issued the operation will complete. Refer to the *Section 10: Dual operations and multiple bank architecture* for detailed information about simultaneous operations allowed during Program/Erase Suspend.

During a Program/Erase Suspend, the device can be placed in standby mode by taking Chip Enable to V_{IH}. Program/Erase is aborted if Reset turns to V_{IL}.

See *Appendix C*, *Figure 24: Program Suspend & Resume flowchart and pseudo code*, and *Figure 26: Erase Suspend & Resume flowchart and pseudo code*, for flowcharts for using the Program/Erase Suspend command.

5.9 Program/Erase Resume command

The Program/Erase Resume command can be used to restart the Program/Erase Controller after a Program/Erase Suspend command has paused it. One Bus Write cycle is required to issue the command. The command can be written to any address.

The Program/Erase Resume command does not change the read mode of the banks. If the suspended bank was in Read Status Register, Read Electronic signature or Read CFI Query mode the bank remains in that mode and outputs the corresponding data. If the bank was in Read Array mode subsequent read operations will output invalid data.

If a Program command is issued during a Block Erase Suspend, then the erase cannot be resumed until the programming operation has completed. It is possible to accumulate suspend operations. For example: suspend an erase operation, start a programming operation, suspend the programming operation then read the array. See *Appendix C*, *Figure 24: Program Suspend & Resume flowchart and pseudo code*, and *Figure 26: Erase Suspend & Resume flowchart and pseudo code*, for flowcharts for using the Program/Erase Resume command.

5.10 Protection Register Program command

The Protection Register Program command is used to Program the 128 bit user One-Time-Programmable (OTP) segment of the Protection Register and the Protection Register Lock. The segment is programmed 16 bits at a time. When shipped all bits in the segment are set to ‘1’. The user can only program the bits to ‘0’.

Two write cycles are required to issue the Protection Register Program command.

- The first bus cycle sets up the Protection Register Program command.
- The second latches the Address and the Data to be written to the Protection Register and starts the Program/Erase Controller.

Read operations output the Status Register content after the programming has started.

The segment can be protected by programming bit 1 of the Protection Lock Register (see *Figure 4: Protection Register memory map*). Attempting to program a previously protected Protection Register will result in a Status Register error. The protection of the Protection Register is not reversible.

The Protection Register Program cannot be suspended. Dual operations between the Parameter B ank and the Protection Register memory space are not allowed (see *Table 14: Dual operation limitations*). See *Appendix C*, *Figure 28: Protection Register Program flowchart and pseudo code*, for a flowchart for using the Protection Register Program command.

5.11 Set Configuration Register command

The Set Configuration Register command is used to write a new value to the Configuration Register which defines the burst length, type, X latency, Synchronous/Asynchronous Read mode and the valid Clock edge configuration.

Two Bus Write cycles are required to issue the Set Configuration Register command.

- The first cycle writes the setup command and the address corresponding to the Configuration Register content.
- The second cycle writes the Configuration Register data and the confirm command.

Once the command is issued the memory returns to Read mode.

The values of the Configuration Register must always be presented on ADQ15-ADQ0. CR0 is on ADQ0, CR1 on ADQ1, etc.; the other address bits are ignored.

5.12 Block Lock command

The Block Lock command is used to lock a block and prevent Program or Erase operations from changing the data in it. All blocks are locked at power-up or reset.

Two Bus Write cycles are required to issue the Block Lock command.

- The first bus cycle sets up the Block Lock command.
- The second Bus Write cycle latches the block address.

The lock status can be monitored for each block using the Read Electronic Signature command. *Table 15* shows the Lock Status after issuing a Block Lock command.

The Block Lock bits are volatile, once set they remain set until a hardware reset or power-down/power-up. They are cleared by a Block Unlock command. Refer to *Section 11: Block locking*, for a detailed explanation. See *Appendix C*, *Figure 27: Locking Operations flowchart and pseudo code*, for a flowchart for using the Lock command.

5.13 Block Unlock command

The Block Unlock command is used to unlock a block, allowing the block to be programmed or erased. Two Bus Write cycles are required to issue the Block Unlock command.

- The first bus cycle sets up the Block Unlock command.
- The second Bus Write cycle latches the block address.

The lock status can be monitored for each block using the Read Electronic Signature command. *Table 15* shows the protection status after issuing a Block Unlock command. Refer to *Section 11: Block locking*, for a detailed explanation and *Appendix C*, *Figure 27: Locking Operations flowchart and pseudo code*, for a flowchart for using the Unlock command.

5.14 Block Lock-Down command

A locked or unlocked block can be locked-down by issuing the Block Lock-Down command. A locked-down block cannot be programmed or erased, or have its protection status changed when $\overline{WP}$ is low, V_{IL}. When $\overline{WP}$ is high, $V_{IH,}$ the Lock-Down function is disabled and the locked blocks can be individually unlocked by the Block Unlock command.

Two Bus Write cycles are required to issue the Block Lock-Down command.

- The first bus cycle sets up the Block Lock command.
- The second Bus Write cycle latches the block address.

The lock status can be monitored for each block using the Read Electronic Signature command. Locked-Down blocks revert to the locked (and not locked-down) state when the device is reset on power-down. *Table 15* shows the Lock Status after issuing a Block Lock-Down command. Refer to *Section 11: Block locking*, for a detailed explanation and *Appendix C*, *Figure 27: Locking Operations flowchart and pseudo code*, for a flowchart for using the Lock-Down command.

Table 5. Standard commands

Commands	Cycles	Bus Operations(1)					
		1st Cycle			2nd Cycle		
		Op.	Add	Data	Op.	Add	Data
Read Array	1+	Write	BKA	FFh	Read	WA	RD
Read Status Register	1+	Write	BKA	70h	Read	BKA(2)	SRD
Read Electronic Signature	1+	Write	BKA	90h	Read	BKA(2)	ESD
Read CFI Query	1+	Write	BKA	98h	Read	BKA(2)	QD
Clear Status Register	1	Write	X	50h			
Block Erase	2	Write	BKA or BA(3)	20h	Write	BA	D0h
Program	2	Write	BKA or WA(3)	40h or 10h	Write	WA	PD
Program/Erase Suspend	1	Write	X	B0h			
Program/Erase Resume	1	Write	X	D0h			
Protection Register Program	2	Write	PRA	C0h	Write	PRA	PRD
Set Configuration Register	2	Write	CRD	60h	Write	CRD	03h
Block Lock	2	Write	BKA or BA(3)	60h	Write	BA	01h
Block Unlock	2	Write	BKA or BA(3)	60h	Write	BA	D0h
Block Lock-Down	2	Write	BKA or BA(3)	60h	Write	BA	2Fh

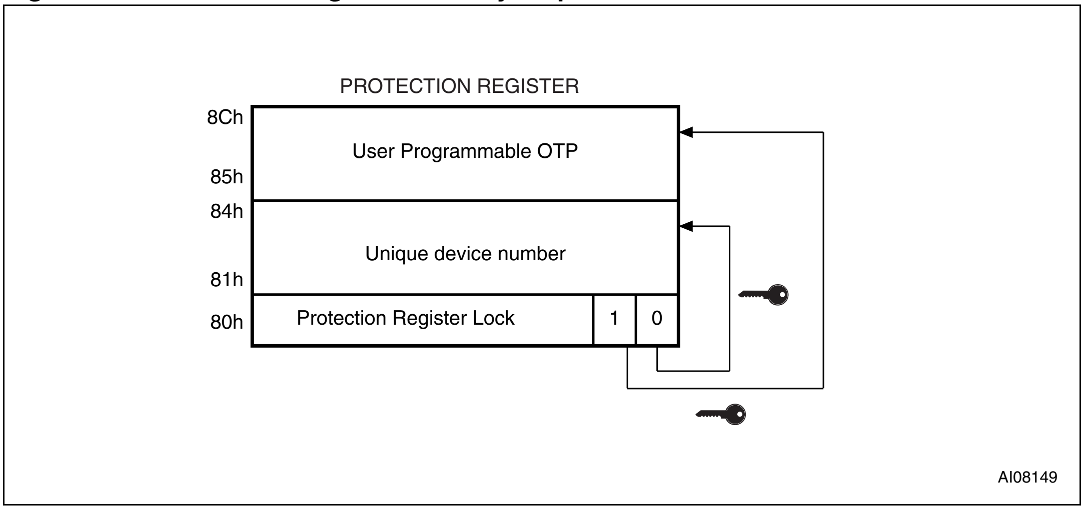
1. X = Don't Care, WA=Word Address in targeted bank, RD = Read Data, SRD = Status Register Data, ESD = Electronic Signature Data, QD = Query Data, BA = Block Address, BKA = Bank Address, PD = Program Data, PRA = Protection Register Address, PRD = Protection Register Data, CRD = Configuration Register Data.
2. Must be same bank as in the first cycle. The signature addresses are listed in *Table 6*.
3. Any address within the bank can be used.

Table 6. Electronic Signature Codes

Code		Address (h)	Data (h)
Manufacturer Code		Bank Address + 00	0020
Device Code	Top (M58WR064HU)	Bank Address + 01	88C0
	Bottom (M58WR064HL)	Bank Address + 01	88C1
Block Protection	Locked	Block Address + 02	0001
	Unlocked		0000
	Locked and Locked-Down		0003
	Unlocked and Locked-Down		0002
Die Revision Code		Bank Address + 03	DRC(1)
Configuration Register		Bank Address + 05	CR(2)
Protection Register Lock	Numonyx Factory Default	Bank Address + 80	0002
	OTP Area Permanently Locked		0000
Protection Register		Bank Address + 81 Bank Address + 84	Unique Device Number
		Bank Address + 85 Bank Address + 8C	OTP Area

1. DRC = Die Revision Code.
2. CR = Configuration Register.

Figure 4. Protection Register memory map



6 Command interface - Factory program commands

The Factory Program commands are used to speed up programming. They require V_{PP} to be at V_{PPH}. Refer to *Table 7: Factory Program commands*, in conjunction with the following text descriptions.

6.0.1 Bank Erase command

The Bank Erase command can be used to erase a bank. It sets all the bits within the selected bank to '1'. All previous data in the bank is lost. The Bank Erase command will ignore any protected blocks within the bank. If all blocks in the bank are protected then the Bank Erase operation will abort and the data in the bank will not be changed. The Status Register will not output any error. Bank Erase operations can be performed at both

$V_{PP} = V_{PPH}$ and $V_{PP} = V_{DD}$

Two Bus Write cycles are required to issue the command.

- The first bus cycle sets up the Bank Erase command.
- The second latches the bank address in the Program/Erase Controller and starts it.

If the second bus cycle is not Write Bank Erase Confirm (D0h), Status Register bits SR4 and SR5 are set and the command aborts. Erase aborts if Reset turns to V_{IL}. As data integrity cannot be guaranteed when the Erase operation is aborted, the bank must be erased again.

Once the command is issued the device outputs the Status Register data when any address within the bank is read. At the end of the operation the bank will remain in Read Status Register mode until a Read Array, Read CFI Query or Read Electronic Signature command is issued.

During Bank Erase operations the bank being erased will only accept the Read Array, Read Status Register, Read Electronic Signature and Read CFI Query command, all other commands will be ignored. A Bank Erase operation cannot be suspended.

Dual Operations are not supported during Bank Erase operations and the command cannot be suspended. Typical Erase times are given in *Table 16: Program, erase times and endurance cycles*

6.1 Double Word Program command

The Double Word Program command improves the programming throughput by writing a page of two adjacent words in parallel. The two words must differ only for the address ADQ0.

If the block being programmed is protected then the Program operation will abort, the data in theblock will not be changed and the Status Register will output the error.

V_{PP} must be set to V_{PPH} during the Double Word Program, otherwise the command will be ignored and the Status register will not output any errors

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.

Read operations in the bank being programmed output the Status Register content after the programming has started.

During Double Word Program operations the bank being programmed will only accept the Read Array, Read Status Register, Read Electronic Signature and Read CFI Query command, all other commands will be ignored. Dual operations are not supported during Double Word Program operations and the command cannot be suspended. Typical Program times are given in *Table 16: Program, erase times and endurance cycles*.

Programming aborts if Reset goes to V_{IL}. As data integrity cannot be guaranteed when the program operation is aborted, the memory locations must be reprogrammed.

See *Appendix C*, *Figure 22: Double Word Program flowchart and pseudo code*, for the flowchart for using the Double Word Program command.

6.2 Quadruple Word Program command

The Quadruple Word Program command improves the programming throughput by writing a page of four adjacent words in parallel. The four words must differ only for the addresses ADQ0 and ADQ1.

V_{PP} must be set to V_{PPH} during the Quadruple Word Program, otherwise the command will be ignored and the Status register will not output any errors.

If the block being programmed is protected then the Program operation will abort, the data in the block will not be changed and the Status Register will output the error.

Five bus write cycles are necessary to issue the Quadruple 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.
- 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.

Read operations to the bank being programmed output the Status Register content after the programming has started.

Programming aborts if Reset goes to V_{IL}. As data integrity cannot be guaranteed when the program operation is aborted, the memory locations must be reprogrammed.

During Quadruple Word Program operations the bank being programmed will only accept the Read Array, Read Status Register, Read Electronic Signature and Read CFI Query command, all other commands will be ignored.

Dual operations are not supported during Quadruple Word Program operations and the command cannot be suspended. Typical Program times are given in *Table 16: Program, erase times and endurance cycles*.

See *Appendix C*, *Figure 23: Quadruple Word Program flowchart and pseudo code*, for the flowchart for using the Quadruple Word Program command.

6.3 Enhanced Factory Program command

The Enhanced Factory Program command can be used to program large streams of data within any one block. It greatly reduces the total programming time when a large number of Words are written to a block at any one time.

If the block being programmed is protected then the Program operation will abort, the data in the block will not be changed and the Status Register will output the error.

The use of the Enhanced Factory Program command requires certain operating conditions.

- V_{PP} must be set to V_{PPH}
- V_{DD} must be within operating range
- Ambient temperature T_A must be 30°C ± 10°C
- The targeted block must be unlocked

Dual operations are not supported during the Enhanced Factory Program operation and the command cannot be suspended.

For optimum performance the Enhanced Factory Program commands should be limited to a maximum of 100 program/erase cycles per block. If this limit is exceeded the internal algorithm will continue to work properly but some degradation in performance is possible. Typical Program times are given in *Table 16*.

The Enhanced Factory Program command has four phases: the Setup Phase, the Program Phase to program the data to the memory, the Verify Phase to check that the data has been correctly programmed and reprogram if necessary and the Exit Phase. Refer to *Table 7: Factory Program commands*, and *Figure 29: Enhanced Factory Program flowchart*.

6.3.1 Setup Phase

The Enhanced Factory Program command requires two Bus Write operations to initiate the command.

- The first bus cycle sets up the Enhanced Factory Program command.
- The second bus cycle confirms the command.

The Status Register P/E.C. Bit SR7 should be read to check that the P/E.C. is ready. After the confirm command is issued, read operations output the Status Register data. The read Status Register command must not be issued as it will be interpreted as data to program.

If the second bus cycle is not EFP confirm(D0h), the Status Register bits SR4 and SR5 are set and the command will be aborted.

V_{PP} value must be in the V_{PPH} range during the Confirm command, otherwise SR4 and SR3 will be set and the command will be aborted.

6.3.2 Program Phase

The Program Phase requires n+1 cycles, where n is the number of Words (refer to *Table 7: Factory Program commands* and *Figure 29: Enhanced Factory Program flowchart*).

Three successive steps are required to issue and execute the Program Phase of the command.

1. Use one Bus Write operation to latch the Start Address and the first Word to be programmed. The Status Register Bank Write Status bit SR0 should be read to check that the P/E.C. is ready for the next Word.
2. Each subsequent Word to be programmed is latched with a new Bus Write operation. The address can either remain the Start Address, in which case the P/E.C. increments the address location or the address can be incremented in which case the P/E.C. jumps to the new address. If any address that is not in the same block as the Start Address is given with data FFFFh, the Program Phase terminates and the Verify Phase begins. The Status Register bit SR0 should be read between each Bus Write cycle to check that the P/E.C. is ready for the next Word.
3. Finally, after all Words have been programmed, write one Bus Write operation with data FFFFh to any address outside the block containing the Start Address, to terminate the programming phase. If the data is not FFFFh, the command is ignored.

The memory is now set to enter the Verify Phase.

6.3.3 Verify Phase

The Verify Phase is similar to the Program Phase in that all Words must be resent to the memory for them to be checked against the programmed data. The Program/Erase Controller checks the stream of data with the data that was programmed in the Program Phase and reprograms the memory location if necessary.

Three successive steps are required to execute the Verify Phase of the command.

1. Use one Bus Write operation to latch the Start Address and the first Word, to be verified. The Status Register bit SR0 should be read to check that the Program/Erase Controller is ready for the next Word.
2. Each subsequent Word to be verified is latched with a new Bus Write operation. The Words must be written in the same order as in the Program Phase. The address can remain the Start Address or be incremented. If any address that is not in the same block as the Start Address is given with data FFFFh, the Verify Phase terminates. Status Register bit SR0 should be read to check that the P/E.C. is ready for the next Word.
3. Finally, after all Words have been verified, write one Bus Write operation with data FFFFh to any address outside the block containing the Start Address, to terminate the Verify Phase.

If the Verify Phase is successfully completed the memory remains in Read Status Register mode. If the Program/Erase Controller fails to reprogram a given location, the error will be signaled in the Status Register.

6.3.4 Exit Phase

Status Register P/E.C. bit SR7 set to '1' indicates that the device has returned to Read mode. A full Status Register check should be done to ensure that the block has been successfully programmed. See the section on the *Status Register* for more details.

6.4 Quadruple Enhanced Factory Program command

The Quadruple Enhanced Factory Program command can be used to program one or more pages of four adjacent words in parallel. The four words must differ only for the addresses ADQ0 and ADQ1. V_{PP} must be set to V_{PPH} during the Quadruple Enhanced Factory Program, otherwise the command will be ignored and the Status register will not output any errors.

If the block being programmed is protected then the Program operation will abort, the data in the block will not be changed and the Status Register will output the error

It has four phases: the Setup Phase, the Load Phase where the data is loaded into the buffer, the combined Program and Verify Phase where the loaded data is programmed to the memory and then automatically checked and reprogrammed if necessary and the Exit Phase. Unlike the Enhanced Factory Program it is not necessary to resubmit the data for the Verify Phase. The Load Phase and the Program and Verify Phase can be repeated to program any number of pages within the block.

6.4.1 Setup Phase

The Quadruple Enhanced Factory Program command requires one Bus Write operation to initiate the load phase. After the setup command is issued, read operations output the Status Register data. The Read Status Register command must not be issued as it will be interpreted as data to program.

6.4.2 Load Phase

The Load Phase requires 4 cycles to load the data (refer to *Table 7: Factory Program commands*, and *Figure 30: Quadruple Enhanced Factory Program flowchart*). Once the first Word of each Page is written it is impossible to exit the Load phase until all four Words have been written.

Two successive steps are required to issue and execute the Load Phase of the Quadruple Enhanced Factory Program command.

1. Use one Bus Write operation to latch the Start Address and the first Word of the first Page to be programmed. For subsequent Pages the first Word address can remain the Start Address (in which case the next Page is programmed) or can be any address in the same block. If any address with data FFFFh is given that is not in the same block as the Start Address, the device enters the Exit Phase. For the first Load Phase Status Register bit SR7 should be read after the first Word has been issued to check that the command has been accepted (bit SR7 set to ‘0’). This check is not required for subsequent Load Phases.
2. Each subsequent Word to be programmed is latched with a new Bus Write operation. The address is only checked for the first Word of each Page as the order of the Words to be programmed is fixed.

The memory is now set to enter the Program and Verify Phase.

6.4.3 Program and Verify Phase

In the Program and Verify Phase the four Words that were loaded in the Load Phase are programmed in the memory array and then verified by the Program/Erase Controller. If any errors are found the Program/Erase Controller reprograms the location. During this phase the Status Register shows that the Program/Erase Controller is busy, Status Register bit SR7 set to '0', and that the device is not waiting for new data, Status Register bit SR0 set to '1'. When Status Register bit SR0 is set to '0' the Program and Verify phase has terminated.

Once the Verify Phase has successfully completed subsequent pages in the same block can be loaded and programmed. The device returns to the beginning of the Load Phase by issuing one Bus Write operation to latch the Address and the first of the four new Words to be programmed.

6.4.4 Exit Phase

Finally, after all the pages have been programmed, write one Bus Write operation with data FFFFh to any address outside the block containing the Start Address, to terminate the Load and Program and Verify Phases.

Status Register bit SR7 set to '1' and bit SR0 set to '0' indicate that the Quadruple Enhanced Factory Program command has terminated. A full Status Register check should be done to ensure that the block has been successfully programmed. See the section on the *Status Register* for more details.

If the Program and Verify Phase has successfully completed the memory returns to Read mode. If the P/E.C. fails to program and reprogram a given location, the error will be signaled in the Status Register.

Table 7. Factory Program commands

Command	Phase	Cycles	Bus Write Operations[1]									
			1st		2nd		3rd		Final -1		Final	
			Add	Data	Add	Data	Add	Data	Add	Data	Add	Data
Bank Erase		2	BKA	80h	Write	BKA	D0h	PD2				
Double Word Program[2]		3	BKA or WA1[3]	35h	WA1	PD1	WA2	PD2				
Quadruple Word Program[4]		5	BKA or WA1[3]	56h	WA1	PD1	WA2	PD2	WA3	PD3	WA4	PD4
Enhanced Factory Program [5]	Setup, Program	2+n +1	BKA or WA1[3]	30h	BA or WA1[6]	D0h	WA1[8]	PD1	WAn[7]	PAn	NOT WA1[8]	FFFF h
	Verify, Exit	n+1	WA1[8]	PD1	WA2[7]	PD2	WA3[7]	PD3	WAn[7]	PAn	NOT WA1[8]	FFFF h
Quadruple Enhanced Factory Program [4],[5]	Setup, first Load	5	BKA or WA1[3]	75h	WA1[8]	PD1	WA2[9]	PD2	WA3[9]	PD3	WA4[9]	PD4
	First Program & Verify		Automatic									
	Subsequent Loads	4	WA1i[8]	PD1i	WA2i [9]	PD2i	WA3i[9]	PD3i			WA4i[9]	PD4i
	Subsequent Program & Verify		Automatic									
	Exit	1	NOT WA1[8]	FFFFh								

1. WA = Word Address in targeted bank, BKA = Bank Address, PD = Program Data, BA = Block Address.
2. Word Addresses 1 and 2 must be consecutive Addresses differing only for A0.
3. Any address within the bank can be used.
4. Word Addresses 1,2,3 and 4 must be consecutive Addresses differing only for A0 and A1.
5. A Bus Read must be done between each Write cycle where the data is programmed or verified to read the Status Register and check that the memory is ready to accept the next data. n = number of Words, i = number of Pages to be programmed.
6. Any address within the block can be used.
7. Address can remain Starting Address WA1 or be incremented.
8. WA1 is the Start Address. NOT WA1 is any address that is not in the same block as WA1.
9. Address is only checked for the first Word of each Page as the order to program the Words in each page is fixed so subsequent Words in each Page can be written to any address.

7 Status Register

The Status Register provides information on the current or previous Program or Erase operations. Issue a Read Status Register command to read the contents of the Status Register, refer to *Section 5.2: Read Status Register command* for more details. To output the contents, the Status Register is latched and updated on the falling edge of the Chip Enable or Output Enable signals and can be read until Chip Enable or Output Enable returns to V_{IH}. The Status Register can only be read using single asynchronous or single synchronous reads. Bus Read operations from any address within the bank, always read the Status Register during Program and Erase operations.

The various bits convey information about the status and any errors of the operation. Bits SR7, SR6, SR2 and SR0 give information on the status of the device and are set and reset by the device. Bits SR5, SR4, SR3 and SR1 give information on errors, they are set by the device but must be reset by issuing a Clear Status Register command or a hardware reset. If an error bit is set to ‘1’ the Status Register should be reset before issuing another command. SR7 to SR1 refer to the status of the device while SR0 refers to the status of the addressed bank.

The bits in the Status Register are summarized in *Table 8: Status Register bits*. Refer to *Table 8* in conjunction with the following text descriptions.

7.0.1 Program/Erase Controller Status Bit (SR7)

The Program/Erase Controller Status bit indicates whether the Program/Erase Controller is active or inactive in any bank. When the Program/Erase Controller Status bit is Low (set to ‘0’), the Program/Erase Controller is active; when the bit is High (set to ‘1’), the Program/Erase Controller is inactive, and the device is ready to process a new command.

The Program/Erase Controller Status is Low immediately after a Program/Erase Suspend command is issued until the Program/Erase Controller pauses. After the Program/Erase Controller pauses the bit is High.

During Program, Erase, operations the Program/Erase Controller Status bit can be polled to find the end of the operation. Other bits in the Status Register should not be tested until the Program/Erase Controller completes the operation and the bit is High.

After the Program/Erase Controller completes its operation the Erase Status, Program Status, V_{PP} Status and Block Lock Status bits should be tested for errors.

7.0.2 Erase Suspend Status Bit (SR6)

The Erase Suspend Status bit indicates that an Erase operation has been suspended or is going to be suspended in the addressed block. When the Erase Suspend Status bit is High (set to ‘1’), a Program/Erase Suspend command has been issued and the memory is waiting for a Program/Erase Resume command.

The Erase Suspend Status should only be considered valid when the Program/Erase Controller Status bit is High (Program/Erase Controller inactive). SR7 is set within the Erase Suspend Latency time of the Program/Erase Suspend command being issued therefore the memory may still complete the operation rather than entering the Suspend mode.

When a Program/Erase Resume command is issued the Erase Suspend Status bit returns Low.

7.0.3 Erase Status Bit (SR5)

The Erase Status bit can be used to identify if the memory has failed to verify that the block or bank has erased correctly. When the Erase Status bit is High (set to ‘1’), the Program/Erase Controller has applied the maximum number of pulses to the block or bank and still failed to verify that it has erased correctly. The Erase Status bit should be read once the Program/Erase Controller Status bit is High (Program/Erase Controller inactive).

Once set High, the Erase Status bit can only be reset Low by a Clear Status Register command or a hardware reset. If set High it should be reset before a new Program or Erase command is issued, otherwise the new command will appear to fail.

7.0.4 Program Status Bit (SR4)

The Program Status bit is used to identify either a Program failure, or an attempt to program a ‘1’ to an already programmed bit when $V_{PP} = V_{PPH}$.

When the Program Status bit goes High (set to ‘1’) after a Program failure, the Program/Erase Controller has applied the maximum number of pulses to the byte and still failed to verify that it has programmed correctly.

After an attempt to program a ‘1’ to an already programmed bit, the Program Status bit SR4 only goes High (set to ’1’) if $V_{PP} = V_{PPH}$ (if $V_{PP} \neq V_{PPH}$, SR4 remains Low (set to ‘0’) and the attempt is not shown).

The Program Status bit should be read once the Program/Erase Controller Status bit is High (Program/Erase Controller inactive).

Once set High, the Program Status bit can only be reset Low by a Clear Status Register command or a hardware reset. If set High it should be reset before a new command is issued, otherwise the new command will appear to fail.

7.0.5 V_{PP} Status Bit (SR3)

The V_{PP} Status bit can be used to identify an invalid voltage on the V_{PP} pin during Program and Erase operations. The V_{PP} pin is only sampled at the beginning of a Program or Erase operation. Indeterminate results can occur if V_{PP} becomes invalid during an operation.

When the V_{PP} Status bit is Low (set to ‘0’), the voltage on the V_{PP} pin was sampled at a valid voltage; when the V_{PP} Status bit is High (set to ‘1’), the V_{PP} pin has a voltage that is below the V_{PP} Lockout Voltage, V_{PPLK}, the memory is protected and Program and Erase operations cannot be performed.

Once set High, the V_{PP} Status bit can only be reset Low by a Clear Status Register command or a hardware reset. If set High it should be reset before a new Program or Erase command is issued, otherwise the new command will appear to fail.

7.0.6 Program Suspend Status Bit (SR2)

The Program Suspend Status bit indicates that a Program operation has been suspended in the addressed block. When the Program Suspend Status bit is High (set to ‘1’), a Program/Erase Suspend command has been issued and the memory is waiting for a Program/Erase Resume command. The Program Suspend Status should only be considered valid when the Program/Erase Controller Status bit is High (Program/Erase Controller inactive). SR2 is set within the Program Suspend Latency time of the Program/Erase Suspend command being issued therefore the memory may still complete the operation rather than entering the Suspend mode.

When a Program/Erase Resume command is issued the Program Suspend Status bit returns Low.

7.0.7 Block Protection Status Bit (SR1)

The Block Protection Status bit can be used to identify if a Program or Block Erase operation has tried to modify the contents of a locked block.

When the Block Protection Status bit is High (set to ‘1’), a Program or Erase operation has been attempted on a locked block.

Once set High, the Block Protection Status bit can only be reset Low by a Clear Status Register command or a hardware reset. If set High it should be reset before a new command is issued, otherwise the new command will appear to fail.

7.0.8 Bank Write/Multiple Word Program Status Bit (SR0)

The Bank Write Status bit indicates whether the addressed bank is programming or erasing. In Enhanced Factory Program mode the Multiple Word Program bit shows if a Word has finished programming or verifying depending on the phase. The Bank Write Status bit should only be considered valid when the Program/Erase Controller Status SR7 is Low (set to ‘0’).

When both the Program/Erase Controller Status bit and the Bank Write Status bit are Low (set to ‘0’), the addressed bank is executing a Program or Erase operation. When the Program/Erase Controller Status bit is Low (set to ‘0’) and the Bank Write Status bit is High (set to ‘1’), a Program or Erase operation is being executed in a bank other than the one being addressed.

In Enhanced Factory Program mode if Multiple Word Program Status bit is Low (set to ‘0’), the device is ready for the next Word, if the Multiple Word Program Status bit is High (set to ‘1’) the device is not ready for the next Word.

Note: Refer to *Appendix C: Flowcharts and pseudo codes*, for using the Status Register.

Table 8. Status Register bits

Bit	Name	Type	Logic Level (1)	Definition	
SR7	P/E.C. Status	Status	'1'	Ready	
			'0'	Busy	
SR6	Erase Suspend Status	Status	'1'	Erase Suspended	
			'0'	Erase In progress or Completed	
SR5	Erase Status	Error	'1'	Erase Error	
			'0'	Erase Success	
SR4	Program Status	Error	'1'	Program Error	
			'0'	Program Success	
SR3	V_{PP} Status	Error	'1'	V_{PP} Invalid, Abort	
			'0'	V_{PP} OK	
SR2	Program Suspend Status	Status	'1'	Program Suspended	
			'0'	Program In Progress or Completed	
SR1	Block Protection Status	Error	'1'	Program/Erase on protected Block, Abort	
			'0'	No operation to protected blocks	
SR0	Bank Write Status	Status	'1'	SR7 = '1'	Not Allowed
				SR7 = '0'	Program or erase operation in a bank other than the addressed bank
			'0'	SR7 = '1'	No Program or erase operation in the device
				SR7 = '0'	Program or erase operation in addressed bank
	Multiple Word Program Status (Enhanced Factory Program mode)	Status	'1'	SR7 = '1'	Not Allowed
				SR7 = '0'	the device is NOT ready for the next word
			'0'	SR7 = '1'	the device is exiting from EFP
				SR7 = '0'	the device is ready for the next Word

1. Logic level '1' is High, '0' is Low.

8 Configuration Register

The Configuration Register is used to configure the type of bus access that the memory will perform. Refer to *Section 9: Read modes* for details on read operations.

The Configuration Register is set through the Command Interface. After a Reset or Power-Up the device is configured for asynchronous read (CR15 = 1). The Configuration Register bits are described in *Table 10*. They specify the selection of the burst length, burst type, burst X latency and the Read operation. Refer to Figures *5* and *6* for examples of synchronous burst configurations.

8.1 Read Select Bit (CR15)

The Read Select bit, CR15, is used to switch between asynchronous and synchronous Bus Read operations. When the Read Select bit is set to '1', read operations are asynchronous; when the Read Select bit is set to '0', read operations are synchronous. Synchronous Burst Read is supported in both parameter and main blocks and can be performed across banks.

On reset or power-up the Read Select bit is set to'1' for asynchronous access.

8.2 Bus Invert Configuration (CR14)

The Bus Invert Configuration bit is used to enable the BINV functionality. When the functionality is enabled, if the BINV pin operates as an input pin (during write bus operations), the BINV signal must always be driven; if it operates as an output pin (during read bus operations), the functionality is valid only during synchronous read operations.

8.3 X-Latency Bits (CR13-CR11)

The X-Latency bits are used during Synchronous Read operations to set the number of clock cycles between the address being latched and the first data becoming available. Refer to *Figure 5: X-latency and data output configuration example*.

For correct operation the X-Latency bits can only assume the values in *Table 10: Configuration Register*.

Table 9 shows how to set the X-Latency parameter, taking into account the speed class of the device and the Frequency used to read the Flash memory in Synchronous mode.

Table 9. X-Latency Settings

fmax	$t_K$min	X-Latency min
		Speed 70ns
30MHz	33ns	2
40MHz	25ns	3
54MHz	19ns	4
66MHz	15ns	4

8.4 Wait Polarity Bit (CR10)

In synchronous burst mode the Wait signal indicates whether the output data are valid or a WAIT state must be inserted. The Wait Polarity bit is used to set the polarity of the Wait signal. When the Wait Polarity bit is set to '0' the Wait signal is active Low. When the Wait Polarity bit is set to '1', the Wait signal is active High.

8.5 Data Output Configuration Bit (CR9)

The Data Output Configuration bit determines whether the output remains valid for one or two clock cycles. When the Data Output Configuration Bit is '0' the output data is valid for one clock cycle, when the Data Output Configuration Bit is '1' the output data is valid for two clock cycles.

The Data Output Configuration depends on the condition:

- $t_K > t_{KQV} + t_{QVK\_CPU}$

where t_K is the clock period, t_{QVK_CPU} is the data setup time required by the system CPU and t_{KQV} is the clock to data valid time. If this condition is not satisfied, the Data Output Configuration bit should be set to '1' (two clock cycles). Refer to *Figure 5: X-latency and data output configuration example*.

8.6 Wait Configuration Bit (CR8)

In burst mode the Wait bit controls the timing of the Wait output pin, WAIT. When WAIT is asserted, Data is Not Valid and when WAIT is deasserted, Data is Valid.

When the Wait bit is '0' the Wait output pin is asserted during the wait state. When the Wait bit is '1', the Wait output pin is asserted one clock cycle before the wait state.

8.7 Burst Type Bit (CR7)

The Burst Type bit is used to configure the sequence of addresses read as sequential or interleaved. When the Burst Type bit is '0' the memory outputs from interleaved addresses; when the Burst Type bit is '1', the memory outputs from sequential addresses. See *Table 11: Burst type definition*, for the sequence of addresses output from a given starting address in each mode.

8.8 Valid Clock Edge Bit (CR6)

The Valid Clock Edge bit, CR6, is used to configure the active edge of the Clock, K, during Synchronous Burst Read operations. When the Valid Clock Edge bit is '0' the falling edge of the Clock is the active edge; when the Valid Clock Edge bit is '1' the rising edge of the Clock is active.

8.9 Power-Down Bit (CR5)

The Power-Down bit is used to enable or disable the Power-Down function. When it is set to ‘0’ the Power-Down function is disabled. If the Reset/Power-Down, $\overline{RP}$, pin goes Low (V_{IL}), the device is reset and the supply current I_{DD} is reduced to the Standby value I_{DD3}. When the Power-Down bit is set to ‘1’ the Power-Down function is enabled. If the Reset/Power-Down, $\overline{RP}$, pin goes Low (V_{IL}) the device switches to the Power-Down state and the supply current I_{DD} is reduced to the Reset/Power-Down value, I_{DD2}.

The recovery time after a Reset/Power-Down, $\overline{RP}$, pulse is significantly longer when Power-Down is enabled (see *Table 26*).

8.10 Wrap Burst Bit (CR3)

The burst reads can be confined inside the 4, 8 or 16 Word boundary (wrap) or overcome the boundary (no wrap). The Wrap Burst bit is used to select between wrap and no wrap. When the Wrap Burst bit is set to ‘0’ the burst read wraps; when it is set to ‘1’ the burst read does not wrap.

8.11 Burst length Bits (CR2-CR0)

The Burst Length bits set the number of Words to be output during a Synchronous Burst Read operation as result of a single address latch cycle. They can be set for 4 words, 8 words, 16 words or continuous burst, where all the words are read sequentially.

In continuous burst mode the burst sequence can cross bank boundaries.

In continuous burst mode or in 4, 8, 16 words no-wrap, depending on the starting address, the device asserts the WAIT output to indicate that a delay is necessary before the data is output.

If the starting address is aligned to a 4 word boundary no wait states are needed and the WAIT output is not asserted.

If the starting address is shifted by 1,2 or 3 positions from the four word boundary, WAIT will be asserted for 1, 2 or 3 clock cycles when the burst sequence crosses the first 16 word boundary, to indicate that the device needs an internal delay to read the successive words in the array. WAIT will be asserted only once during a continuous burst access. See also *Table 11: Burst type definition*.

CR4 is reserved for future use.

Table 10. Configuration Register

Bit	Description	Value	Description
CR15	Read Select	0	Synchronous Read
		1	Asynchronous Read (Default at power-on)
CR14	Bus invert configuration	0	BINV (power save) disabled (default)
		1	BINV (power save) enabled
CR13-CR11	X-Latency	010	2 clock latency
		011	3 clock latency
		100	4 clock latency
		101	5 clock latency
		111	Reserved (default)
		Other configurations reserved	
CR10	Wait Polarity	0	WAIT is active Low (default)
		1	WAIT is active high
CR9	Data Output Configuration	0	Data held for one clock cycle
		1	Data held for two clock cycles (default)
CR8	Wait Configuration	0	WAIT is active during wait state (default)
		1	WAIT is active one data cycle before wait state
CR7	Burst Type	0	Interleaved
		1	Sequential (default)
CR6	Valid Clock Edge	0	Falling Clock edge
		1	Rising Clock edge (default)
CR5	Power-Down Configuration	0	Power-Down disabled (default)
		1	Power-Down enabled
CR4	Reserved		
CR3	Wrap Burst	0	Wrap
		1	No Wrap (default)
CR2-CR0	Burst Length	001	4 words
		010	8 words
		011	16 words
		111	Continuous (CR7 must be set to ‘1’) (default)

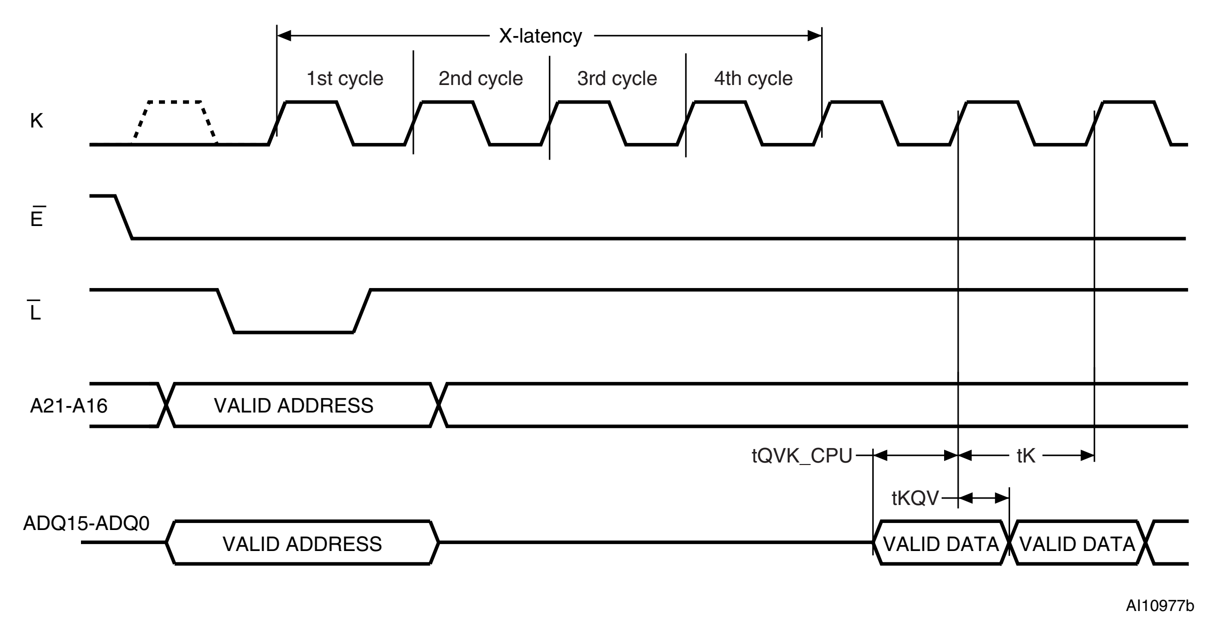
Table 11. Burst type definition

Mode	Start Add	4 Words		8 Words		16 Words		Continuous Burst
		Sequential	Interleaved	Sequential	Interleaved	Sequential	Interleaved	
Wrap	0	0-1-2-3	0-1-2-3	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-7-8-9-10-11-12-13-14-15	0-1-2-3-4-5-6-7-8-9-10-11-12-13-14-15	0-1-2-3-4-5-6...
	1	1-2-3-0	1-0-3-2	1-2-3-4-5-6-7-0	1-0-3-2-5-4-7-6	1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-0	1-0-3-2-5-4-7-6-9-8-11-10-13-12-15-14	1-2-3-4-5-6-7-...15-WAIT-16-17-18...
	2	2-3-0-1	2-3-0-1	2-3-4-5-6-7-0-1	2-3-0-1-6-7-4-5	2-3-4-5-6-7-8-9-10-11-12-13-14-15-0-1	2-3-0-1-6-7-4-5-10-11-8-9-14-15-12-13	2-3-4-5-6-7...15-WAIT-WAIT-16-17-18...
	3	3-0-1-2	3-2-1-0	3-4-5-6-7-0-1-2	3-2-1-0-7-6-5-4	3-4-5-6-7-8-9-10-11-12-13-14-15-0-1-2	3-2-1-0-7-6-5-4-11-10-9-8-15-14-13-12	3-4-5-6-7...15-WAIT-WAIT-WAIT-16-17-18...
	...							
	7	7-4-5-6	7-6-5-4	7-0-1-2-3-4-5-6	7-6-5-4-3-2-1-0	7-8-9-10-11-12-13-14-15-0-1-2-3-4-5-6	7-6-5-4-3-2-1-0-15-14-13-12-11-10-9-8	7-8-9-10-11-12-13-14-15-WAIT-WAIT-WAIT-16-17...
	...							
	12							12-13-14-15-16-17-18...
	13							13-14-15-WAIT-16-17-18...
	14							14-15-WAIT-WAIT-16-17-18....
	15							15-WAIT-WAIT-WAIT-16-17-18...

Table 11. Burst type definition (continued)

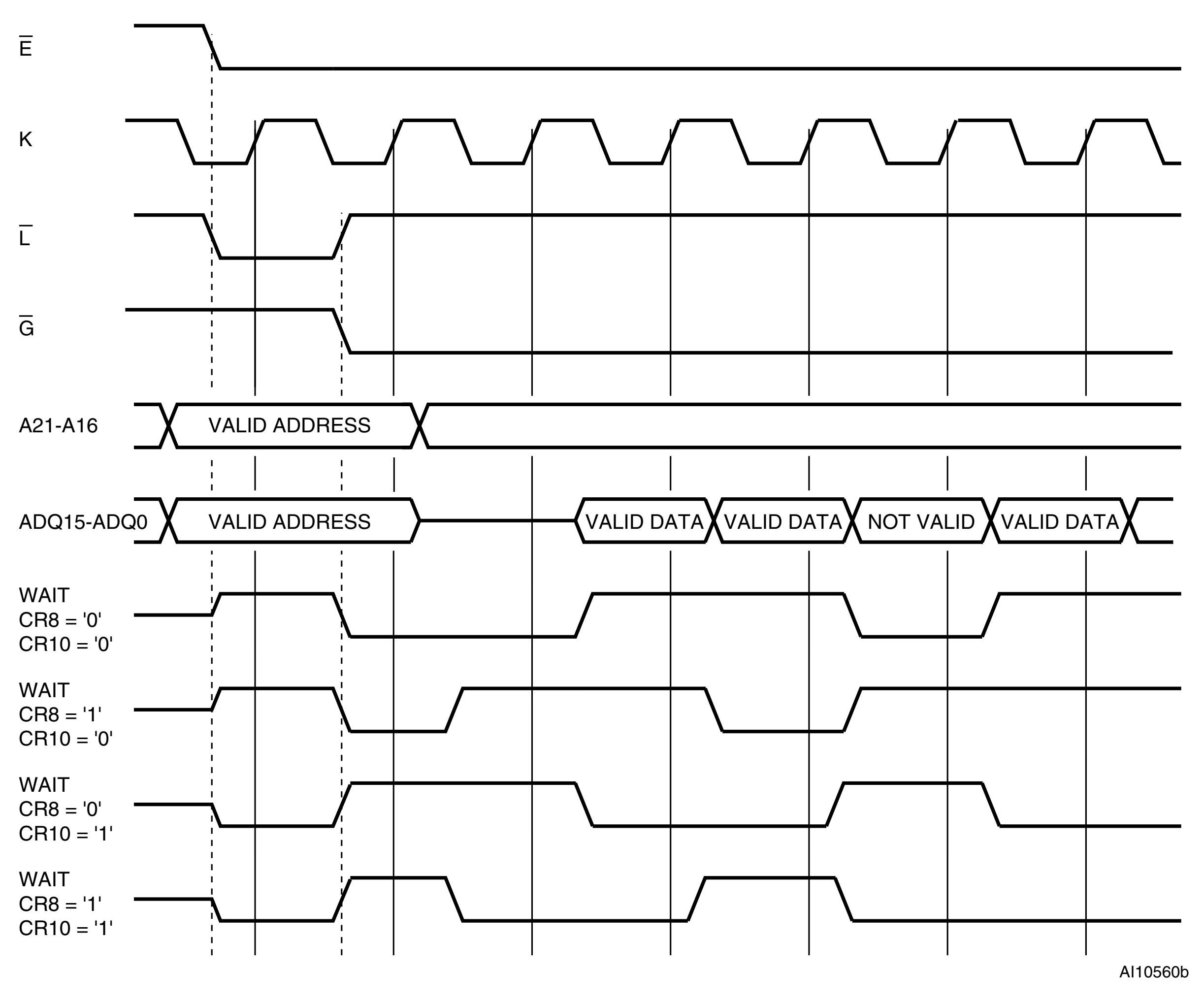
Mode	Start Add	4 Words		8 Words		16 Words		Continuous Burst
		Sequential	Interleaved	Sequential	Interleaved	Sequential	Interleaved	
No-wrap	0	0-1-2-3		0-1-2-3-4-5-6-7		0-1-2-3-4-5-6-7-8-9-10-11-12-13-14-15		Same as for Wrap (Wrap /No Wrap has no effect on Continuous Burst)
	1	1-2-3-4		1-2-3-4-5-6-7-8		1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-WAIT-16		
	2	2-3-4-5		2-3-4-5-6-7-8-9...		2-3-4-5-6-7-8-9-10-11-12-13-14-15-WAIT-WAIT-16-17		
	3	3-4-5-6		3-4-5-6-7-8-9-10		3-4-5-6-7-8-9-10-11-12-13-14-15-WAIT-WAIT-WAIT-16-17-18		
	...							
	7	7-8-9-10		7-8-9-10-11-12-13-14		7-8-9-10-11-12-13-14-15-WAIT-WAIT-WAIT-16-17-18-19-20-21-22		
	...							
	12	12-13-14-15		12-13-14-15-16-17-18-19		12-13-14-15-16-17-18-19-20-21-22-23-24-25-26-27		
	13	13-14-15-WAIT-16		13-14-15-WAIT-16-17-18-19-20		13-14-15-WAIT-16-17-18-19-20-21-22-23-24-25-26-27-28		
	14	14-15-WAIT-WAIT-16-17		14-15-WAIT-WAIT-16-17-18-19-20-21		14-15-WAIT-WAIT-16-17-18-19-20-21-22-23-24-25-26-27-28-29		
	15	15-WAIT-WAIT-WAIT-16-17-18		15-WAIT-WAIT-WAIT-16-17-18-19-20-21-22		15-WAIT-WAIT-WAIT-16-17-18-19-20-21-22-23-24-25-26-27-28-29-30		

Figure 5. X-latency and data output configuration example



1. Settings shown: X-latency = 4, Data Output held for one clock cycle.

Figure 6. Wait configuration example



9 Read modes

Read operations can be performed in two different ways depending on the settings in the Configuration Register. If the clock signal is 'don't care' for the data output, the read operation is Asynchronous; if the data output is synchronized with clock, the read operation is Synchronous.

The Read mode and data output format are determined by the Configuration Register. (See *Section 8: Configuration Register* for details). All banks supports both asynchronous and synchronous read operations. The Multiple Bank architecture allows read operations in one bank, while write operations are being executed in another (see Tables *12* and *13*).

9.1 Asynchronous Read mode

In Asynchronous Read operations the clock signal is 'don't care'. The device outputs the data corresponding to the address latched, that is the memory array, Status Register, Common Flash Interface or Electronic Signature depending on the command issued. CR15 in the Configuration Register must be set to '1' for Asynchronous operations.

In Asynchronous Read mode, the WAIT signal is always deasserted.

The device features an Automatic Standby mode. During asynchronous read operations, after a bus inactivity of 150ns, the device automatically switches to the Automatic Standby mode. In this condition the power consumption is reduced to the standby value I_{DD4} and the outputs are still driven.

See *Table 22: Asynchronous read AC characteristics*, and *Figure 9: Asynchronous random access read AC waveforms*.

9.2 Synchronous Burst Read mode

In Synchronous Burst Read mode the data is output in bursts synchronized with the clock. It is possible to perform burst reads across bank boundaries.

Synchronous Burst Read mode can only be used to read the memory array. For other read operations, such as Read Status Register, Read CFI and Read Electronic Signature, Single Synchronous Read or Asynchronous Random Access Read must be used.

In Synchronous Burst Read mode the flow of the data output depends on parameters that are configured in the Configuration Register.

A burst sequence is started at the first clock edge (rising or falling depending on Valid Clock Edge bit CR6 in the Configuration Register) after the falling edge of Latch Enable. Addresses are internally incremented and after a delay of 2 to 5 clock cycles (X latency bits CR13-CR11) the corresponding data are output on each clock cycle.

The number of Words to be output during a Synchronous Burst Read operation can be configured as 4, 8 or 16 Words or Continuous (Burst Length bits CR2-CR0). The data can be configured to remain valid for one or two clock cycles (Data Output Configuration bit CR9).

The order of the data output can be modified through the Burst Type and the Wrap Burst bits in the Configuration Register. The burst sequence may be configured to be sequential or interleaved (CR7). The burst reads can be confined inside the 4, 8 or 16 Word boundary (Wrap) or overcome the boundary (No Wrap). If the starting address is aligned to the Burst Length (4, 8 or 16 Words), the wrapped configuration has no impact on the output sequence. Interleaved mode is not allowed in Continuous Burst Read mode or with No Wrap sequences.

A WAIT signal may be asserted to indicate to the system that an output delay will occur. This delay will depend on the starting address of the burst sequence; the worst case delay will occur when the sequence is crossing a 16 word boundary and the starting address was at the end of a four word boundary.

WAIT is asserted during X-latency, the Wait state and at the end of a 4, 8 and 16 Word burst. It is only deasserted when output data are valid or when $\overline{G}$ is at V_{IH}. In Continuous Burst Read mode a Wait state will occur when crossing the first 16 Word boundary. If the burst starting address is aligned to a 4 Word Page, the Wait state will not occur.

The WAIT signal can be configured to be active Low or active High by setting CR10 in the Configuration Register.

See *Table 23: Synchronous read AC characteristics*, and *Figure 10: Synchronous burst read AC waveforms*, for details.

9.2.1 Synchronous Burst Read Suspend

A Synchronous Burst Read operation can be suspended, freeing the data bus for other higher priority devices. It can be suspended during the initial access latency time (before data is output), or after the device has output data. When the Synchronous Burst Read operation is suspended, internal array sensing continues and any previously latched internal data is retained. A burst sequence can be suspended and resumed as often as required as long as the operating conditions of the device are met.

A Synchronous Burst Read operation is suspended when $\overline{E}$ is low and the current address has been latched (on a Latch Enable rising edge or on a valid clock edge). The clock signal is then halted at V_{IH} or at V_{IL}, and $\overline{G}$ goes high.

When $\overline{G}$ becomes low again and the clock signal restarts, the Synchronous Burst Read operation is resumed exactly where it stopped.

WAIT being gated by $\overline{E}$ remains active and will not revert to high-impedance when $\overline{G}$ goes high. So if two or more devices are connected to the system's READY signal, to prevent bus contention the WAIT signal of the Flash memory should not be directly connected to the system's READY signal.

See *Table 23: Synchronous read AC characteristics*, and *Figure 12: Synchronous burst read suspend AC waveforms*, for details.

9.3 Single Synchronous Read mode

Single Synchronous Read operations are similar to Synchronous Burst Read operations except that only the first data output after the X latency is valid.

Synchronous Single Reads are used to read the Electronic Signature, Status Register, CFI, Block Protection Status, Configuration Register Status or Protection Register. When the addressed bank is in Read CFI, Read Status Register or Read Electronic Signature mode, the WAIT signal is deasserted when Output Enable, $\overline{G}$, is at V_{IH} or for the one clock cycle during which output data is valid. Otherwise, it is asserted.
See *Table 23: Synchronous read AC characteristics*, and *Figure 11: Single synchronous read AC waveforms*, for details.

10 Dual operations and multiple bank architecture

The Multiple Bank Architecture of the M58WR064HU/L provides flexibility for software developers by allowing code and data to be split with 4Mbit granularity. The Dual Operations feature simplifies the software management of the device and allows code to be executed from one bank while another bank is being programmed or erased.

The Dual operations feature means that while programming or erasing in one bank, Read operations are possible in another bank with zero latency (only one bank at a time is allowed to be in Program or Erase mode). If a Read operation is required in a bank which is programming or erasing, the Program or Erase operation can be suspended. Also if the suspended operation was Erase then a Program command can be issued to another block, so the device can have one block in Erase Suspend mode, one programming and other banks in Read mode. Bus Read operations are allowed in another bank between setup and confirm cycles of program or erase operations. The combination of these features means that read operations are possible at any moment.

Dual operations between the Parameter Bank and the CFI, OTP or Electronic Signature memory space, are not allowed. *Table 14: Dual operation limitations* shows which dual operations are allowed between the CFI, OTP, Electronic Signature locations and the memory array. Tables *12* and *13* show the dual operations possible in other banks and in the same bank. Note that only the commonly used commands are represented in these tables. For a complete list of possible commands refer to *Appendix D: Command interface state tables*.

Table 12. Dual operations allowed in other banks

Status of bank	Commands allowed in another bank							
	Read Array	Read Status Register	Read CFI Query	Read Electronic Signature	Program	Block Erase	Program/ Erase Suspend	Program/ Erase Resume
Idle	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Programming	Yes	Yes	Yes	Yes	–	–	Yes	–
Erasing	Yes	Yes	Yes	Yes	–	–	Yes	–
Program Suspended	Yes	Yes	Yes	Yes	–	–	–	Yes
Erase Suspended	Yes	Yes	Yes	Yes	Yes	–	–	Yes

Table 13. Dual operations allowed in same bank

Status of bank	Commands allowed in same bank							
	Read Array	Read Status Register	Read CFI Query	Read Electronic Signature	Program	Block Erase	Program/ Erase Suspend	Program/ Erase Resume
Idle	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Programming	–(1)	Yes	Yes	Yes	–	–	Yes	–
Erasing	–(1)	Yes	Yes	Yes	–	–	Yes	–
Program Suspended	Yes(2)	Yes	Yes	Yes	–	–	–	Yes
Erase Suspended	Yes(2)	Yes	Yes	Yes	Yes(2)	–	–	Yes

1. The Read Array command is accepted but the data output is not guaranteed until the Program or Erase has completed.
2. Not allowed in the Block or Word that is being erased or programmed.

Table 14. Dual operation limitations

Current Status		Commands allowed			
		Read CFI / OTP / Electronic Signature	Read Parameter Blocks	Read Main Blocks	
				Located in Parameter Bank	Not Located in Parameter Bank
Programming / Erasing Parameter Blocks		No	No	No	Yes
Programming / Erasing Main Blocks	Located in Parameter Bank	Yes	No	No	Yes
	Not Located in Parameter Bank	Yes	Yes	Yes	In Different Bank Only
Programming OTP		No	No	No	No

11 Block locking

The M58WR064HU/L features an instant, individual block locking scheme that allows any block to be locked or unlocked with no latency. This locking scheme has three levels of protection.

- Lock/Unlock - this first level allows software-only control of block locking.
- Lock-Down - this second level requires hardware interaction before locking can be changed.
- $V_{PP} \leq V_{PPLK}$ - the third level offers a complete hardware protection against program and erase on all blocks.

The protection status of each block can be set to Locked, Unlocked, and Lock-Down. *Table 15*, defines all of the possible protection states ($\overline{WP}$, DQ1, DQ0), and *Appendix C*, *Figure 27*, shows a flowchart for the locking operations.

11.1 Reading a block's lock status

The lock status of every block can be read in the Read Electronic Signature mode of the device. To enter this mode write 90h to the device. Subsequent reads at the address specified in *Table 6*, will output the protection status of that block. The lock status is represented by DQ0 and DQ1. DQ0 indicates the Block Lock/Unlock status and is set by the Lock command and cleared by the Unlock command. It is also automatically set when entering Lock-Down. DQ1 indicates the Lock-Down status and is set by the Lock-Down command. It cannot be cleared by software, only by a hardware reset or power-down.

The following sections explain the operation of the locking system.

11.2 Locked state

The default status of all blocks on power-up or after a hardware reset is Locked (states (0,0,1) or (1,0,1)). Locked blocks are fully protected from any program or erase. Any program or erase operations attempted on a locked block will return an error in the Status Register. The Status of a Locked block can be changed to Unlocked or Lock-Down using the appropriate software commands. An Unlocked block can be Locked by issuing the Lock command.

11.3 Unlocked state

Unlocked blocks (states (0,0,0), (1,0,0) (1,1,0)), can be programmed or erased. All unlocked blocks return to the Locked state after a hardware reset or when the device is powered-down. The status of an unlocked block can be changed to Locked or Locked-Down using the appropriate software commands. A locked block can be unlocked by issuing the Unlock command.

11.4 Lock-Down state

Blocks that are Locked-Down (state (0,1,x))are protected from program and erase operations (as for Locked blocks) but their protection status cannot be changed using software commands alone. A Locked or Unlocked block can be Locked-Down by issuing the Lock-Down command. Locked-Down blocks revert to the Locked state when the device is reset or powered-down.

The Lock-Down function is dependent on the $\overline{WP}$ input pin. When $\overline{WP}$=0 (V_{IL}), the blocks in the Lock-Down state (0,1,x) are protected from program, erase and protection status changes. When $\overline{WP}$=1 (V_{IH}) the Lock-Down function is disabled (1,1,x) and Locked-Down blocks can be individually unlocked to the (1,1,0) state by issuing the software command, where they can be erased and programmed. These blocks can then be re-locked (1,1,1) and unlocked (1,1,0) as desired while $\overline{WP}$ remains high. When $\overline{WP}$ is Low, blocks that were previously Locked-Down return to the Lock-Down state (0,1,x) regardless of any changes made while $\overline{WP}$ was High. Device reset or power-down resets all blocks, including those in Lock-Down, to the Locked state.

11.5 Locking operations during Erase Suspend

Changes to block lock status can be performed during an erase suspend by using the standard locking command sequences to unlock, lock or lock-down a block. This is useful in the case when another block needs to be updated while an erase operation is in progress.

To change block locking during an erase operation, first write the Erase Suspend command, then check the status register until it indicates that the erase operation has been suspended. Next write the desired Lock command sequence to a block and the lock status will be changed. After completing any desired lock, read, or program operations, resume the erase operation with the Erase Resume command.

If a block is locked or locked-down during an erase suspend of the same block, the locking status bits will be changed immediately, but when the erase is resumed, the erase operation will complete. Locking operations cannot be performed during a program suspend. Refer to *Appendix D: Command interface state tables*, for detailed information on which commands are valid during erase suspend.

Table 15. Lock status

Current Protection Status(1) ($\overline{WP}$, ADQ1, ADQ0)		Next Protection Status(1) ($\overline{WP}$, ADQ1, ADQ0)			
Current State	Program/Erase Allowed	After Block Lock Command	After Block Unlock Command	After Block Lock-Down Command	After $\overline{WP}$ transition
1,0,0	yes	1,0,1	1,0,0	1,1,1	0,0,0
1,0,1(2)	no	1,0,1	1,0,0	1,1,1	0,0,1
1,1,0	yes	1,1,1	1,1,0	1,1,1	0,1,1
1,1,1	no	1,1,1	1,1,0	1,1,1	0,1,1
0,0,0	yes	0,0,1	0,0,0	0,1,1	1,0,0
0,0,1(2)	no	0,0,1	0,0,0	0,1,1	1,0,1
0,1,1	no	0,1,1	0,1,1	0,1,1	1,1,1 or 1,1,0(3)

1. The lock status is defined by the write protect pin and by DQ1 (‘1’ for a locked-down block) and DQ0 (‘1’ for a locked block) as read in the Read Electronic Signature command with A1 = V_{IH} and A0 = V_{IL}.
2. All blocks are locked at power-up, so the default configuration is 001 or 101 according to $\overline{WP}$ status.
3. A $\overline{WP}$ transition to V_{IH} on a locked block will restore the previous DQ0 value, giving a 111 or 110.

12 Program and erase times and endurance cycles

The Program and Erase times and the number of Program/ Erase cycles per block are shown in *Table 16*. In the M58WR064HU/L the maximum number of Program/ Erase cycles depends on the voltage supply used.

Table 16. Program, erase times and endurance cycles(1)

Supply	Parameter	Condition		Min	Typ	Typical after 100k W/E Cycles	Max	Unit
$V_{PP} = V_{DD}$	Erase	Parameter Block (4 KWord)(2)			0.3	1	2.5	s
		Main Block (32 KWord)	Preprogrammed		0.8	3	4	s
			Not Preprogrammed		1		4	s
		Bank (4Mbit)	Preprogrammed		4.5			s
			Not Preprogrammed		6			s
	Program(3)	Word			12	12	100	µs
		Parameter Block (4 KWord)			40			ms
		Main Block (32 KWord)			300			ms
	Suspend Latency	Program			5		10	µs
		Erase			5		20	µs
	Program/Erase Cycles (per Block)	Main Blocks		100,000				cycles
		Parameter Blocks		100,000				cycles
$V_{PP} = V_{PPH}$	Erase	Parameter Block (4 KWord)			0.25		2.5	s
		Main Block (32 KWord)			0.8		4	s
		Bank (4Mbit)			6			s
	Program(3)	Word/ Double Word/ Quadruple Word(4)			10		100	µs
		Parameter Block (4 KWord)	Quad-Enhanced Factory		11			ms
			Enhanced Factory		38			ms
			Quadruple Word(4)		8			ms
			Word		32			ms
		Main Block (32 KWord)	Quad-Enhanced Factory		88			ms
			Enhanced Factory		300			ms
			Quadruple Word(4)		64			ms
			Word		256			ms
		Bank (4Mbit)	Quad-Enhanced Factory(4)		0.7			s
			Quadruple Word(4)		0.5			s
	Program/Erase Cycles (per Block)	Main Blocks					1000	cycles
		Parameter Blocks					2500	cycles

1. T_A = –40 to 85°C; $V_{DD} = V_{DDQ}$ = 1.7V to 2V.
2. The difference between Preprogrammed and not preprogrammed is not significant (‹30ms).
3. Values are liable to change with the external system-level overhead (command sequence and Status Register polling execution).
4. Measurements performed at 25°C. T_A = 30°C ±10°C for Quadruple Word, Double Word and Quadruple Enhanced Factory Program.

13 Maximum rating

Stressing the device above the rating listed in the Absolute Maximum Ratings table may cause permanent damage to the device. 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. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability. Refer also to the Numonyx SURE Program and other relevant quality documents.

Table 17. Absolute maximum ratings

Symbol	Parameter	Value		Unit
		Min	Max	
T_A	Ambient Operating Temperature	–40	85	°C
T_{BIAS}	Temperature Under Bias	–40	125	°C
T_{STG}	Storage Temperature	–65	155	°C
V_{IO}	Input or Output Voltage	–0.5	V_{DDQ}+0.6	V
V_{DD}	Supply Voltage	–0.2	2.45	V
V_{DDQ}	Input/Output Supply Voltage	–0.2	2.45	V
V_{PP}	Program Voltage	–0.2	14	V
I_O	Output Short Circuit Current		100	mA
t_{VPPH}	Time for V_{PP} at V_{PPH}		100	hours

14 DC and AC parameters

This section summarizes the operating measurement conditions, and the DC and AC characteristics of the device. The parameters in the DC and AC characteristics Tables that follow, are derived from tests performed under the Measurement Conditions summarized in *Table 18: Operating and AC measurement conditions*. Designers should check that the operating conditions in their circuit match the operating conditions when relying on the quoted parameters.

Table 18. Operating and AC measurement conditions

Parameter	M58WR064HU/L		Units
	70		
	Min	Max	
V_{DD} Supply Voltage	1.7	2	V
V_{DDQ} Supply Voltage	1.7	2	V
V_{PP} Supply Voltage (Factory environment)	8.5	12.6	V
V_{PP} Supply Voltage (Application environment)	–0.4	V_{DDQ}+0.4	V
Ambient Operating Temperature	–40	85	°C
Load Capacitance (C_L)	30		pF
Input Rise and Fall Times		5	ns
Input Pulse Voltages	0 to V_{DDQ}		V
Input and Output Timing Ref. Voltages	V_{DDQ}/2		V

Figure 7. AC measurement I/O waveform

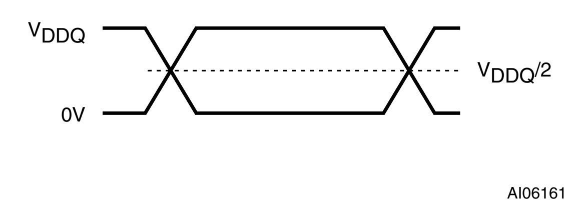


Figure 8. AC measurement load circuit

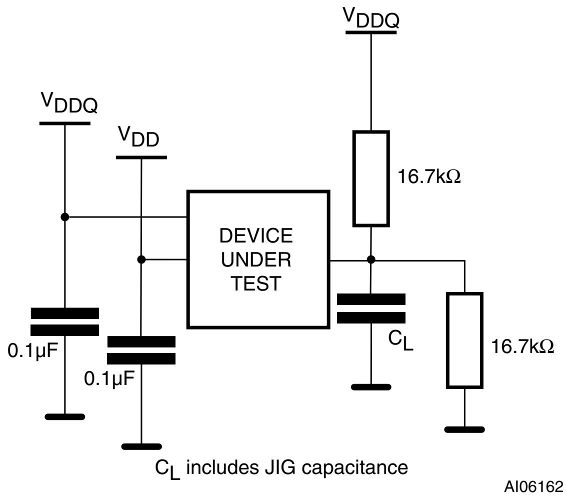


Table 19. Capacitance(1)

Symbol	Parameter	Test Condition	Min	Max	Unit
C_{IN}	Input Capacitance	V_{IN} = 0V	6	8	pF
C_{OUT}	Output Capacitance	V_{OUT} = 0V	8	12	pF

1. Sampled only, not 100% tested.

Table 20. DC Characteristics - Currents

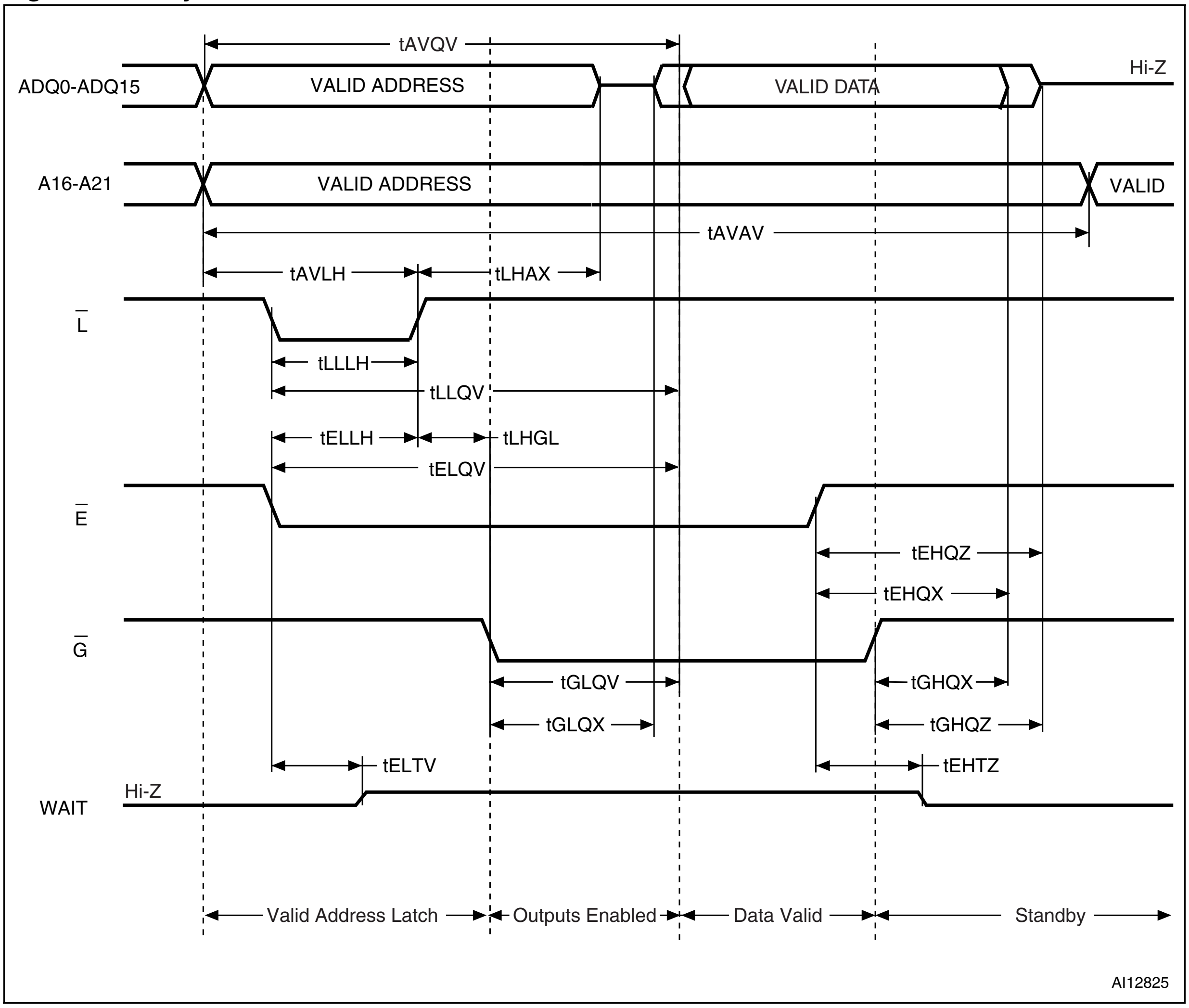
Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
I_{LI}	Input Leakage Current	$0V \leq V_{IN} \leq V_{DDQ}$			±1	µA
I_{LO}	Output Leakage Current	$0V \leq V_{OUT} \leq V_{DDQ}$			±1	µA
I_{DD1}	Supply Current Asynchronous Read (f=6MHz)	$\overline{E} = V_{IL}$, $\overline{G} = V_{IH}$		10	20	mA
	Supply Current Synchronous Read (f=66MHz)	4 Word		18	20	mA
		8 Word		20	22	mA
		16 Word		25	27	mA
		Continuous		28	30	mA
I_{DD2}	Supply Current (Reset/Power-Down)	$\overline{RP} = V_{SS} \pm 0.2V$		2	10	µA
I_{DD3}	Supply Current (Standby)	$K=V_{SS}$ $E = V_{DDQ} \pm 0.2V$		22	50	µA
I_{DD4}	Supply Current (Automatic Standby)	$\overline{E} = V_{IL}$, $\overline{G} = V_{IH}$		22	50	µA
I_{DD5}(1)	Supply Current (Program)	$V_{PP} = V_{PPH}$		8	15	mA
		$V_{PP} = V_{DD}$		10	20	mA
	Supply Current (Erase)	$V_{PP} = V_{PPH}$		8	15	mA
		$V_{PP} = V_{DD}$		10	20	mA
I_{DD6}(1)(2)	Supply Current (Dual Operations)	Program/Erase in one Bank, Asynchronous Read in another Bank		20	40	mA
		Program/Erase in one Bank, Synchronous Read (continuous burst 66MHz) in another Bank		38	50	mA
I_{DD7}(1)	Supply Current Program/ Erase Suspended (Standby)	$K=V_{SS}$ $E = V_{DDQ} \pm 0.2V$		22	50	µA
I_{PP1}(1)	V_{PP} Supply Current (Program)	$V_{PP} = V_{PPH}$		5	10	mA
		$V_{PP} = V_{DD}$		0.2	5	µA
	V_{PP} Supply Current (Erase)	$V_{PP} = V_{PPH}$		5	10	mA
		$V_{PP} = V_{DD}$		0.2	5	µA
I_{PP2}	V_{PP} Supply Current (Read)	$V_{PP} = V_{PPH}$		100	400	µA
		$V_{PP} \leq V_{DD}$		0.2	5	µA
I_{PP3}(1)	V_{PP} Supply Current (Standby)	$V_{PP} \leq V_{DD}$		0.2	5	µA

1. Sampled only, not 100% tested.
2. V_{DD} Dual Operation current is the sum of read and program or erase currents.

Table 21. DC characteristics - voltages

Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
V_{IL}	Input Low Voltage		–0.5		0.4	V
V_{IH}	Input High Voltage		V_{DDQ} –0.4		V_{DDQ} + 0.4	V
V_{OL}	Output Low Voltage	I_{OL} = 100µA			0.1	V
V_{OH}	Output High Voltage	I_{OH} = –100µA	V_{DDQ} –0.1			V
V_{PP1}	V_{PP} Program Voltage-Logic	Program, Erase	1.3		2.4	V
V_{PPH}	V_{PP} Program Voltage Factory	Program, Erase	8.5	12	12.6	V
V_{PPLK}	Program or Erase Lockout				0.9	V
V_{LKO}	V_{DD} Lock Voltage				1	V
V_{RPH}	$\overline{RP}$ pin Extended High Voltage				3.3	V

Figure 9. Asynchronous random access read AC waveforms



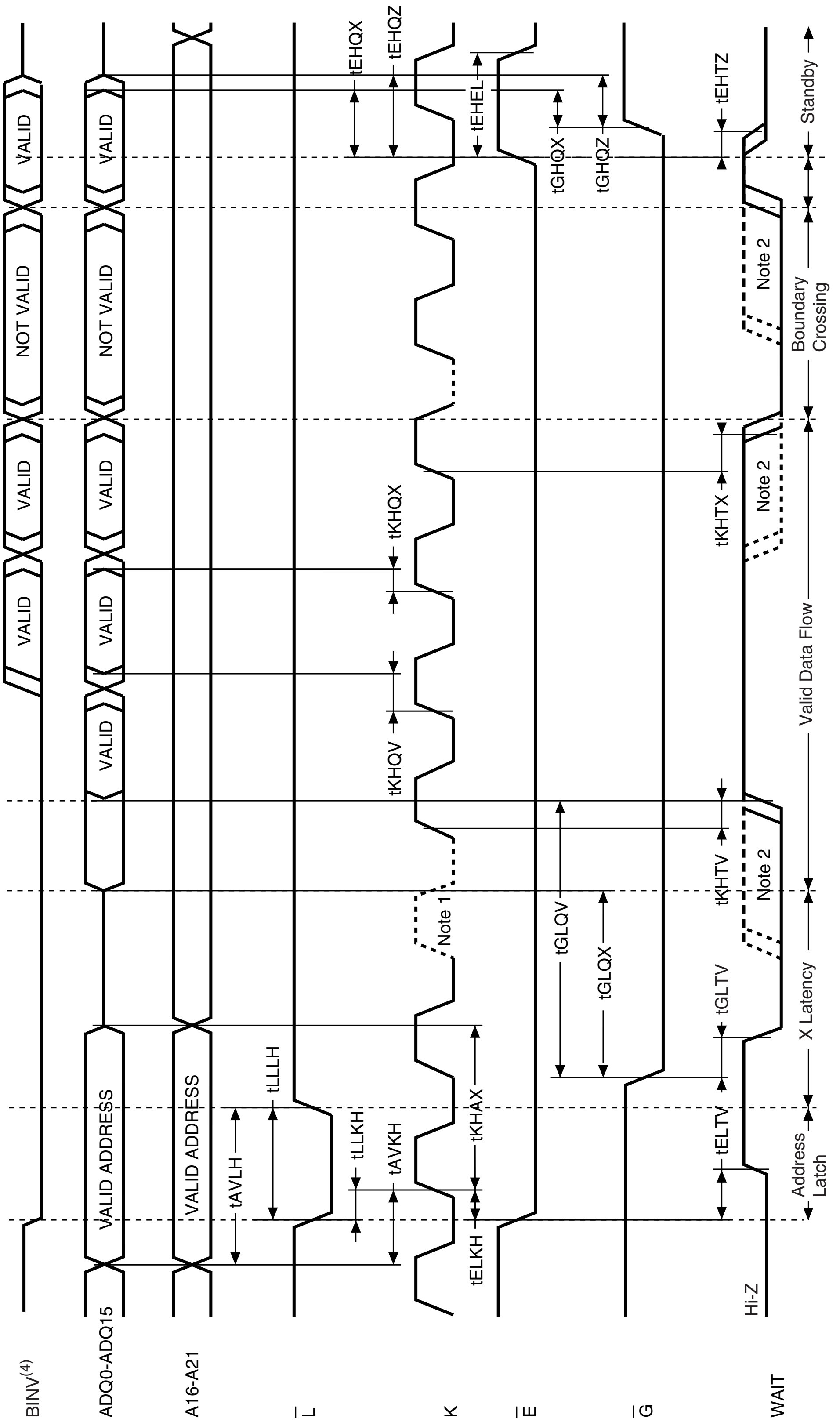
1. Write Enable, $\overline{W}$, is High, WAIT is active Low.

Table 22. Asynchronous read AC characteristics

	Symbol	Alt	Parameter		M58WR064HU/L 70	Unit
Read Timings	t_{AVAV}	t_{RC}	Address Valid to Next Address Valid	Min	70	ns
	t_{AVQV}	t_{ACC}	Address Valid to Output Valid (Random)	Max	70	ns
	t_{ELTV}		Chip Enable Low to Wait Valid	Max	11	ns
	t_{ELQV}(2)	t_{CE}	Chip Enable Low to Output Valid	Max	70	ns
	t_{EHTZ}		Chip Enable High to Wait Hi-Z	Max	14	ns
	t_{EHQX}(1)	t_{OH}	Chip Enable High to Output Transition	Min	0	ns
	t_{EHQZ}(1)	t_{HZ}	Chip Enable High to Output Hi-Z	Max	14	ns
	t_{GLQV}(2)	t_{OE}	Output Enable Low to Output Valid	Max	20	ns
	t_{GLQX}(1)	t_{OLZ}	Output Enable Low to Output Transition	Min	0	ns
	t_{GHQX}(1)	t_{OH}	Output Enable High to Output Transition	Min	0	ns
	t_{GHQZ}(1)	t_{DF}	Output Enable High to Output Hi-Z	Max	14	ns
Latch Timings	t_{AVLH}	t_{AVADVH}	Address Valid to Latch Enable High	Min	7	ns
	t_{ELLH}	t_{ELADVH}	Chip Enable Low to Latch Enable High	Min	10	ns
	t_{LHAX}	t_{ADVHAX}	Latch Enable High to Address Transition	Min	7	ns
	t_{LLLH}	$t_{ADVLADVH}$	Latch Enable Pulse Width	Min	7	ns
	t_{LLQV}	t_{ADVLQV}	Latch Enable Low to Output Valid (Random)	Max	70	ns
	t_{LHGL}	t_{ADVHGL}	Latch Enable High to Output Enable Low	Min	5	ns

1. Sampled only, not 100% tested.
2. $\overline{G}$ may be delayed by up to t_{ELQV} - t_{GLQV} after the falling edge of $\overline{E}$ without increasing t_{ELQV}.

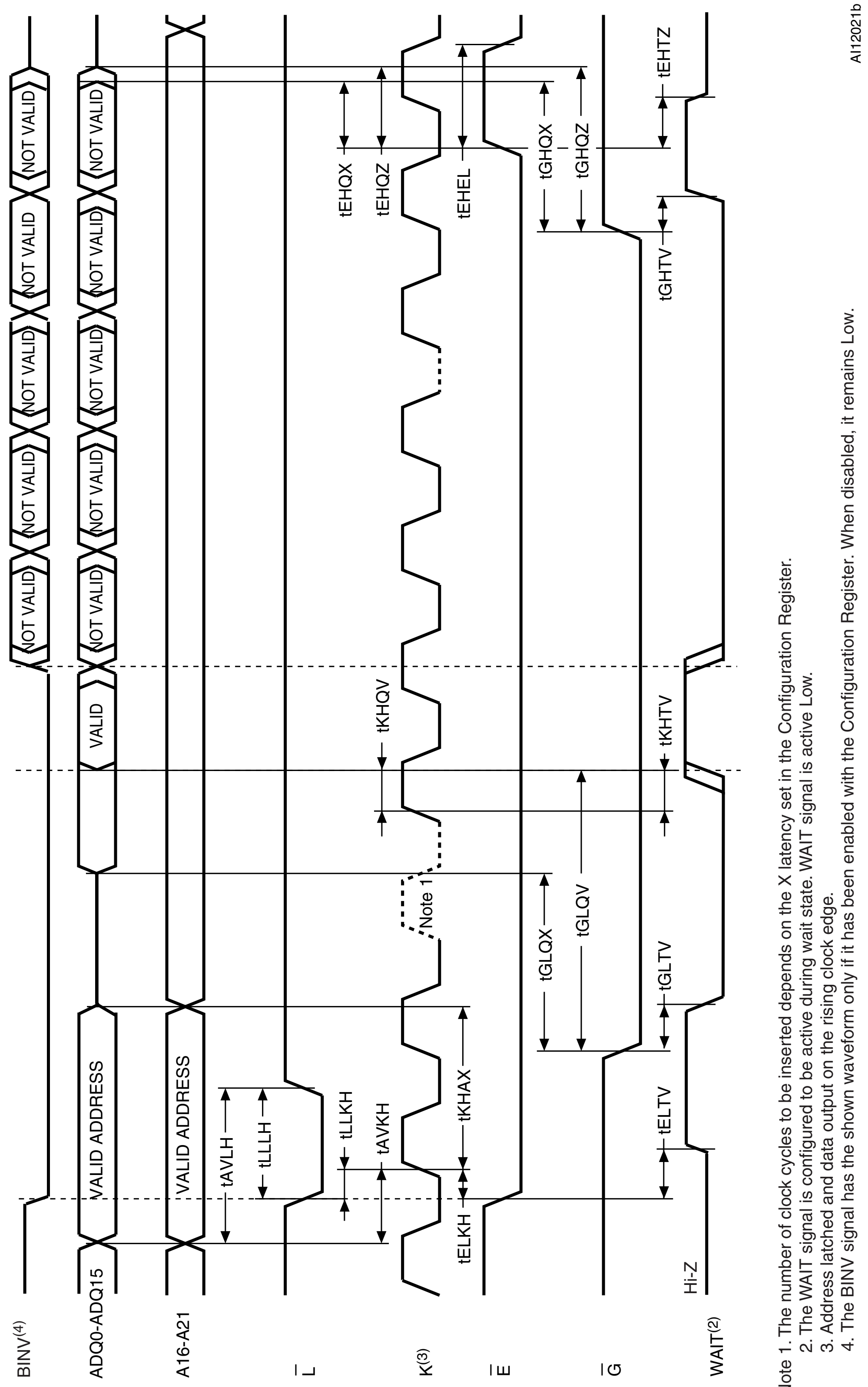
Figure 10. Synchronous burst read AC waveforms



Note 1. The number of clock cycles to be inserted depends on the X latency set in the Configuration Register.
2. The WAIT signal can be configured to be active during wait state or one cycle before. WAIT signal is active Low.
3. Address latched and data output on the rising clock edge.
4. The BINV signal has the waveform shown only if it has been enabled with the Configuration Register. If it is disabled, it remains Low.

AI12020b

Figure 11. Single synchronous read AC waveforms



Note 1. The number of clock cycles to be inserted depends on the X latency set in the Configuration Register.
2. The WAIT signal is configured to be active during wait state. WAIT signal is active Low.
3. Address latched and data output on the rising clock edge.
4. The BINV signal has the shown waveform only if it has been enabled with the Configuration Register. When disabled, it remains Low.

Figure 12. Synchronous burst read suspend AC waveforms

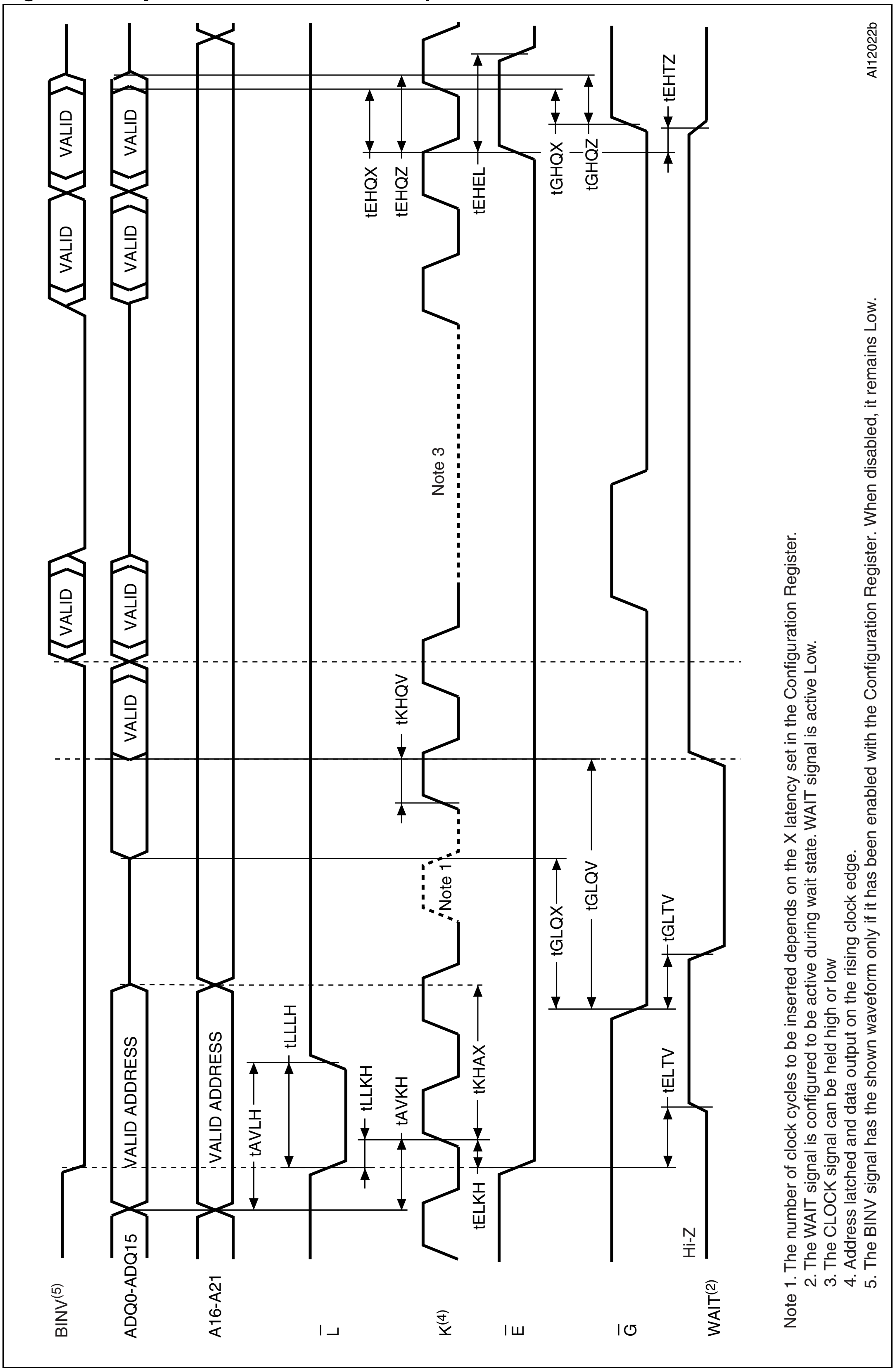


Figure 13. Clock input AC Waveform

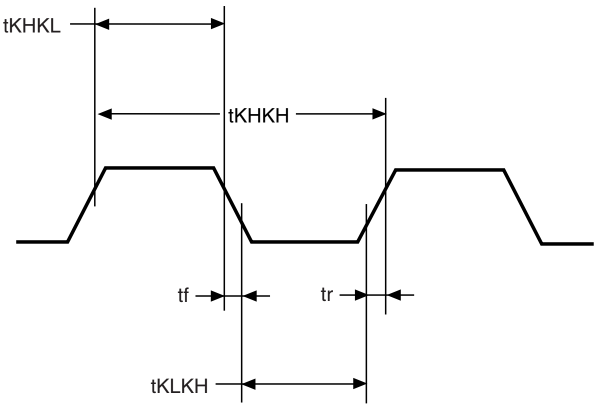


Table 23. Synchronous read AC characteristics

	Symbol	Alt	Parameter		M58WR064HU/L 70	Unit
Synchronous Read Timings	t_{AVKH}	t_{AVCLKH}	Address Valid to Clock High	Min	5	ns
	t_{ELKH}	t_{ELCLKH}	Chip Enable Low to Clock High	Min	5	ns
	t_{ELTV}		Chip Enable Low to Wait Valid	Max	11	ns
	t_{EHEL}		Chip Enable Pulse Width (subsequent synchronous reads)	Min	14	ns
	t_{EHTZ}		Chip Enable High to Wait Hi-Z	Max	14	ns
	t_{GHTV}		Output Enable High to Wait Valid	Min	11	ns
	t_{GLTV}		Output Enable Low to Wait Valid	Max	11	ns
	t_{KHAX}	t_{CLKHAX}	Clock High to Address Transition	Min	7	ns
	t_{KHQV} t_{KHTV}	t_{CLKHQV}	Clock High to Output Valid Clock High to WAIT Valid	Max	11	ns
	t_{KHQX} t_{KHTX}	t_{CLKHQX}	Clock High to Output Transition Clock High to WAIT Transition	Min	3	ns
	t_{LLKH}	$t_{ADVLCLKH}$	Latch Enable Low to Clock High	Min	5	ns
Clock Specifications	t_{KHKH}(1)	t_{CLK}	Clock Period (66MHz)	Min	15	ns
	t_{KHKL} t_{KLKH}		Clock High to Clock Low Clock Low to Clock High	Min	3.5	ns
	t_f t_r		Clock Fall or Rise Time	Max	3	ns

1. The device can support jitters of +/-5% on clock frequency.
2. Sampled only, not 100% tested. For other timings please refer to *Table 22: Asynchronous read AC characteristics*.

Figure 14. Write AC waveforms, Write Enable controlled

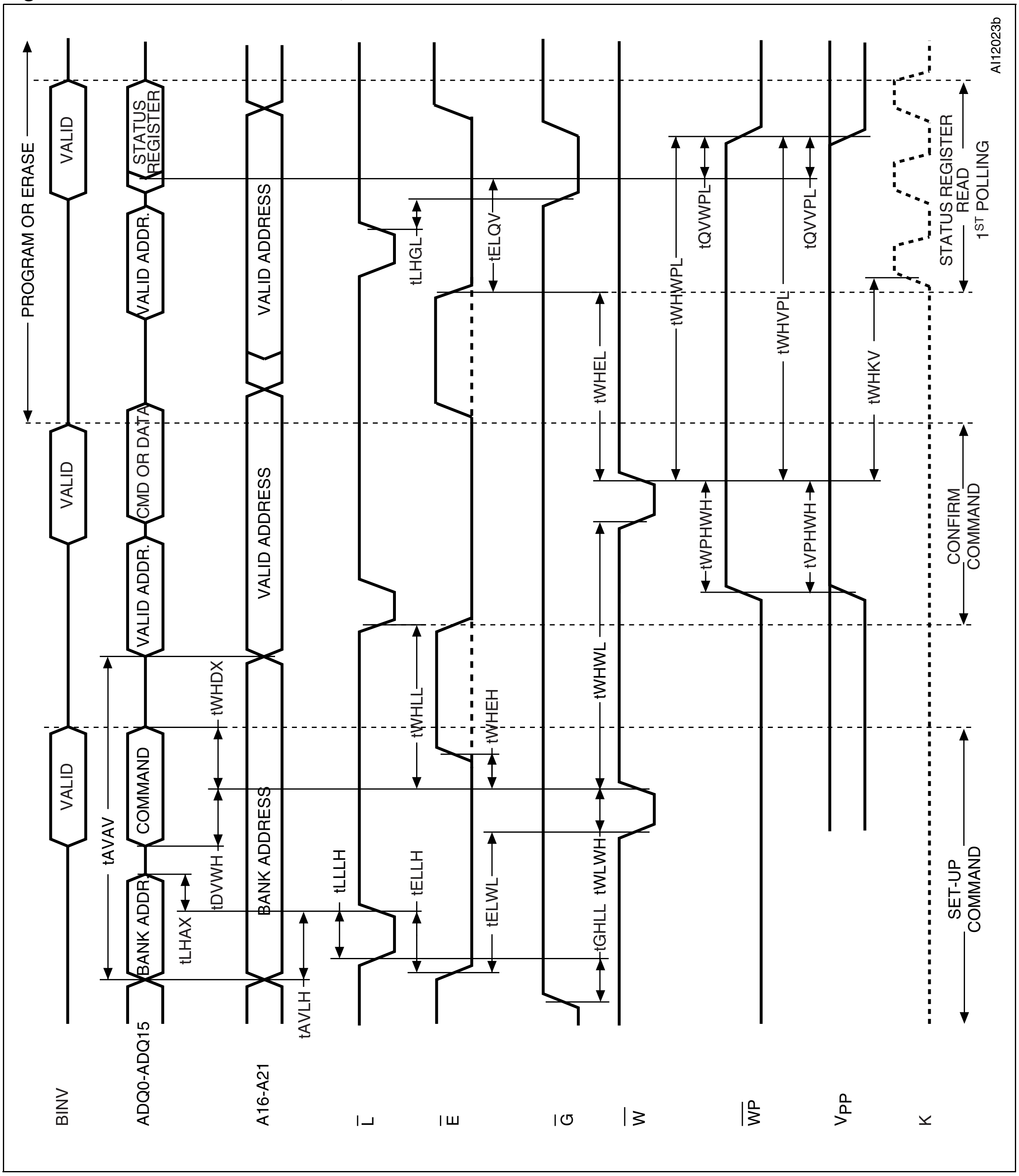


Table 24. Write AC characteristics, Write Enable controlled

	Symbol	Alt	Parameter		M58WR064HU/L 70	Unit
Write Enable Controlled Timings	t_{AVAV}	t_{WC}	Address Valid to Next Address Valid	Min	70	ns
	t_{AVLH}		Address Valid to Latch Enable High	Min	7	ns
	t_{DVWH}	t_{DS}	Data Valid to Write Enable High	Min	40	ns
	t_{ELLH}		Chip Enable Low to Latch Enable High	Min	10	ns
	t_{ELWL}	t_{CS}	Chip Enable Low to Write Enable Low	Min	0	ns
	t_{ELQV}		Chip Enable Low to Output Valid	Min	70	ns
	t_{GHLL}		Output Enable High to Latch Enable Low	Min	20	ns
	t_{GHWL}		Output Enable High to Write Enable Low	Min	20	ns
	t_{LHAX}		Latch Enable High to Address Transition	Min	7	ns
	t_{LHGL}		Latch Enable High to Output Enable Low	Min	7	ns
	t_{LLLH}		Latch Enable Pulse Width	Min	7	ns
	t_{WHDX}	t_{DH}	Write Enable High to Input Transition	Min	0	ns
	t_{WHEH}	t_{CH}	Write Enable High to Chip Enable High	Min	0	ns
	t_{WHEL}(2)		Write Enable High to Chip Enable Low	Min	25	ns
	t_{WHGL}		Write Enable High to Output Enable Low	Min	0	ns
	t_{WHLL}		Write Enable High to Latch Enable Low	Min	0	ns
	t_{WHWL}	t_{WPH}	Write Enable High to Write Enable Low	Min	25	ns
	t_{WLWH}	t_{WP}	Write Enable Low to Write Enable High	Min	45	ns
Protection Timings	t_{QVVPL}		Output (Status Register) Valid to V_{PP} Low	Min	0	ns
	t_{QVWPL}		Output (Status Register) Valid to Write Protect Low	Min	0	ns
	t_{VPHWH}	t_{VPS}	V_{PP} High to Write Enable High	Min	200	ns
	t_{WHVPL}		Write Enable High to V_{PP} Low	Min	200	ns
	t_{WHWPL}		Write Enable High to Write Protect Low	Min	200	ns
	t_{WPHWH}		Write Protect High to Write Enable High	Min	200	ns

1. Sampled only, not 100% tested.
2. t_{WHEL} has the values shown when reading in the targeted bank or when reading following a Set Configuration Register command. System designers should take this into account and may insert a software No-Op instruction to delay the first read in the same bank after issuing any command and to delay the first read to any address after issuing a Set Configuration Register command. If the first read after the command is a Read Array operation in a different bank and no changes to the Configuration Register have been issued, t_{WHEL} is 0ns.

Figure 15. Write AC waveforms, Chip Enable controlled

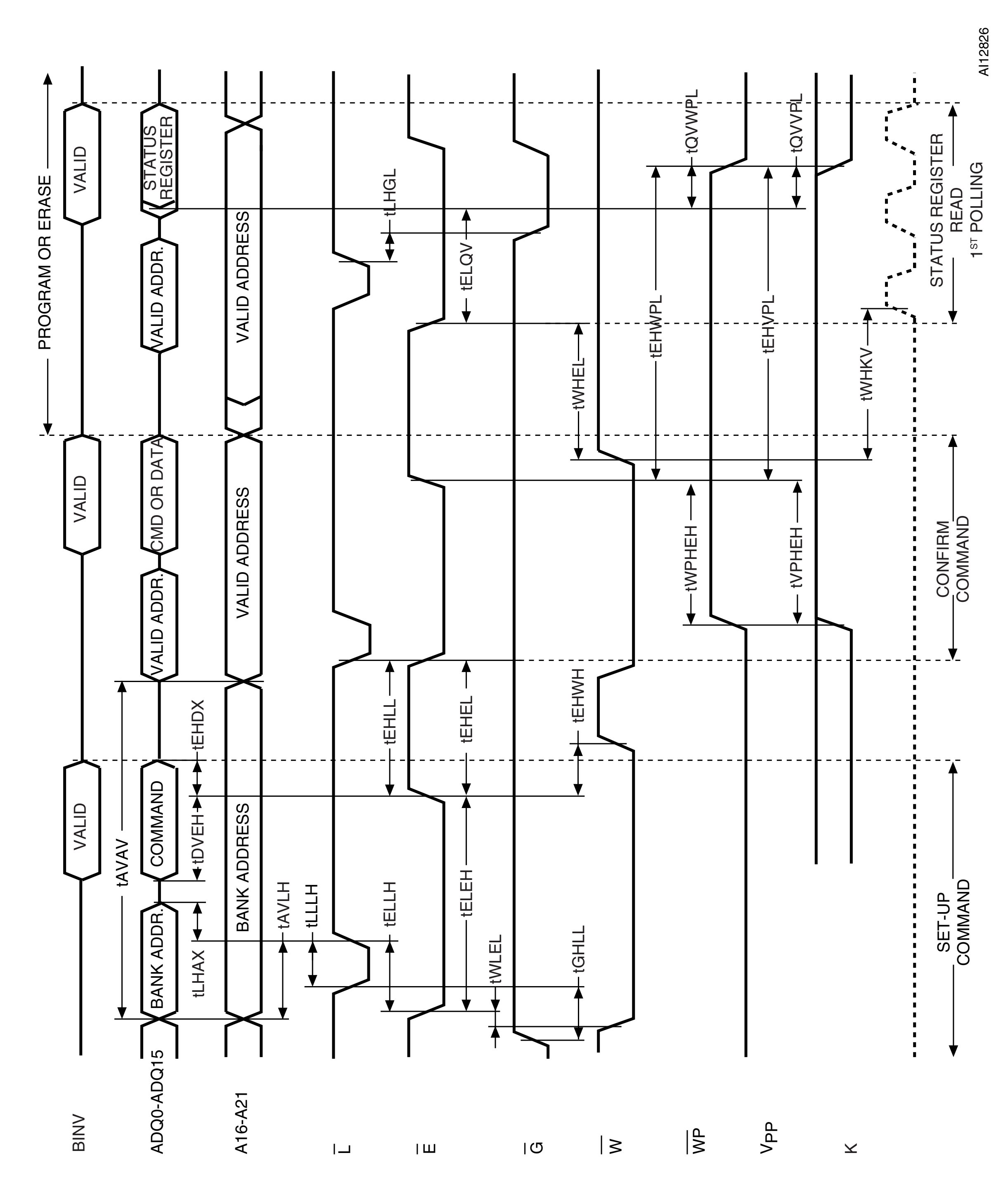


Table 25. Write AC characteristics, Chip Enable controlled

	Symbol	Alt	Parameter		M58WR064HU/L 70	Unit
Chip Enable Controlled Timings	t_{AVAV}	t_{WC}	Address Valid to Next Address Valid	Min	70	ns
	t_{AVLH}		Address Valid to Latch Enable High	Min	7	ns
	t_{DVEH}	t_{DS}	Data Valid to Chip Enable High	Min	40	ns
	t_{EHDX}	t_{DH}	Chip Enable High to Input Transition	Min	0	ns
	t_{EHEL}	t_{WPH}	Chip Enable High to Chip Enable Low	Min	25	ns
	t_{EHLL}		Chip Enable High to Latch Enable Low	Min	0	ns
	t_{EHWH}	t_{CH}	Chip Enable High to Write Enable High	Min	0	ns
	t_{ELEH}	t_{WP}	Chip Enable Low to Chip Enable High	Min	45	ns
	t_{ELLH}		Chip Enable Low to Latch Enable High	Min	10	ns
	t_{ELQV}		Chip Enable Low to Output Valid	Min	70	ns
	t_{GHEL}		Output Enable High to Chip Enable Low	Min	20	ns
	t_{GHLL}		Output Enable High to Latch Enable Low	Min	20	ns
	t_{LHAX}		Latch Enable High to Address Transition	Min	7	ns
	t_{LHGL}		Latch Enable High to Output Enable Low	Min	7	ns
	t_{LLLH}		Latch Enable Pulse Width	Min	7	ns
	t_{WHEL} (1)		Write Enable High to Chip Enable Low	Min	25	ns
	t_{WLEL}	t_{CS}	Write Enable Low to Chip Enable Low	Min	0	ns
Protection Timings	t_{EHVPL}		Chip Enable High to V_{PP} Low	Min	200	ns
	t_{EHWPL}		Chip Enable High to Write Protect Low	Min	200	ns
	t_{QVVPL}		Output (Status Register) Valid to V_{PP} Low	Min	0	ns
	t_{QVWPL}		Output (Status Register) Valid to Write Protect Low	Min	0	ns
	t_{VPHEH}	t_{VPS}	V_{PP} High to Chip Enable High	Min	200	ns
	t_{WPHEH}		Write Protect High to Chip Enable High	Min	200	ns

1. t_{WHEL} has the values shown when reading in the targeted bank or when reading following a Set Configuration Register command. System designers should take this into account and may insert a software No-Op instruction to delay the first read in the same bank after issuing any command and to delay the first read to any address after issuing a Set Configuration Register command. If the first read after the command is a Read Array operation in a different bank and no changes to the Configuration Register have been issued, t_{WHEL} is 0ns.

3. Sampled only, not 100% tested.

Figure 16. Reset and Power-up AC waveforms

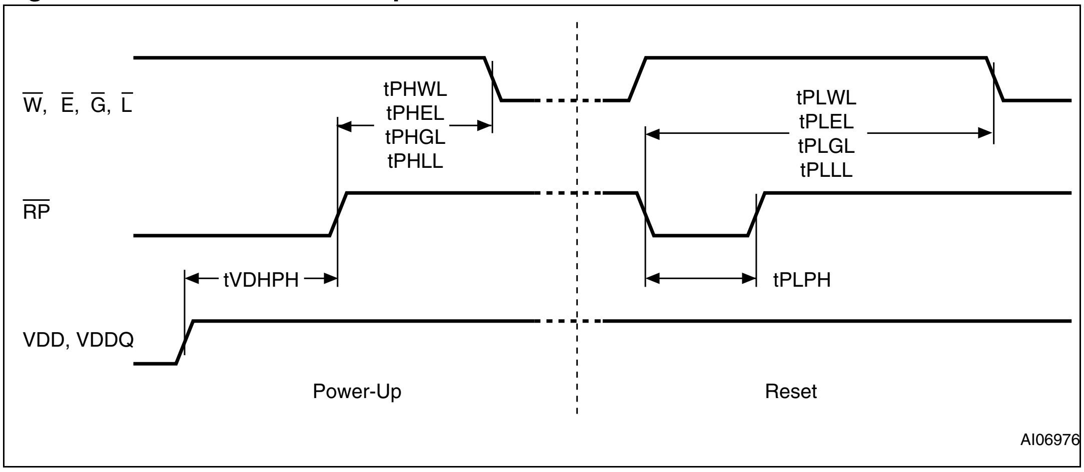


Table 26. Reset and Power-up AC characteristics

Symbol	Parameter	Test Condition		70	Unit
t_{PLWL} t_{PLEL} t_{PLGL} t_{PLLL}	Reset Low to Write Enable Low, Chip Enable Low, Output Enable Low, Latch Enable Low	During Program	Min	10	µs
		During Erase	Min	20	µs
		After Power-Down	Min	50	µs
		Other Conditions	Min	80	ns
t_{PHWL} t_{PHEL} t_{PHGL} t_{PHLL}	Reset High to Write Enable Low Chip Enable Low Output Enable Low Latch Enable Low		Min	30	ns
t_{PLPH}(1)(2)	RP Pulse Width		Min	50	ns
t_{VDHPH}(3)	Supply Voltages High to Reset High		Min	50	µs

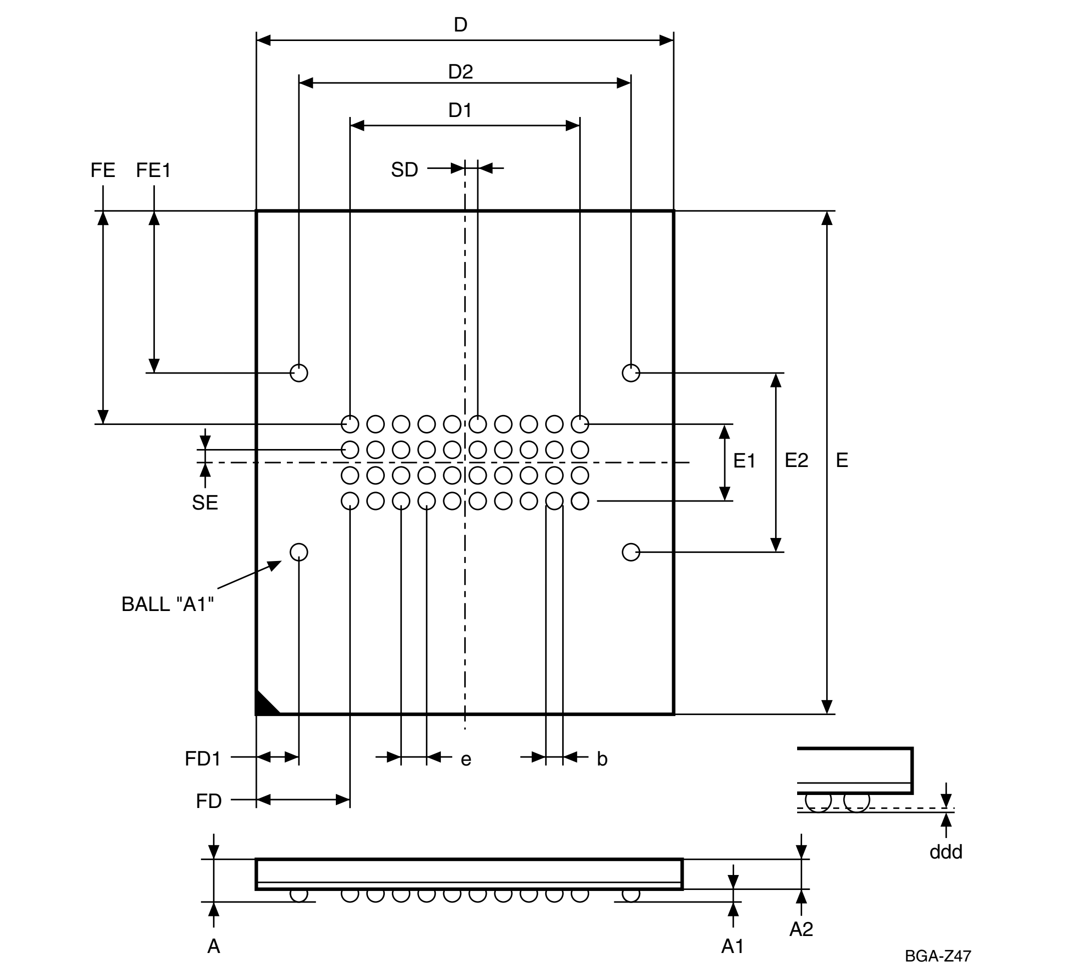
1. The device Reset is possible but not guaranteed if t_{PLPH} < 50ns.
2. Sampled only, not 100% tested.
3. It is important to assert RP in order to allow proper CPU initialization during Power-Up or Reset.

15 Package mechanical

In order to meet environmental requirements, Numonyx offers these devices in ECOPACK® packages. These packages have a Lead-free second-level interconnect. 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 17. VFBGA44 - 7.7x9mm, 10x4 ball array, 0.5mm pitch, Bottom View Package Outline

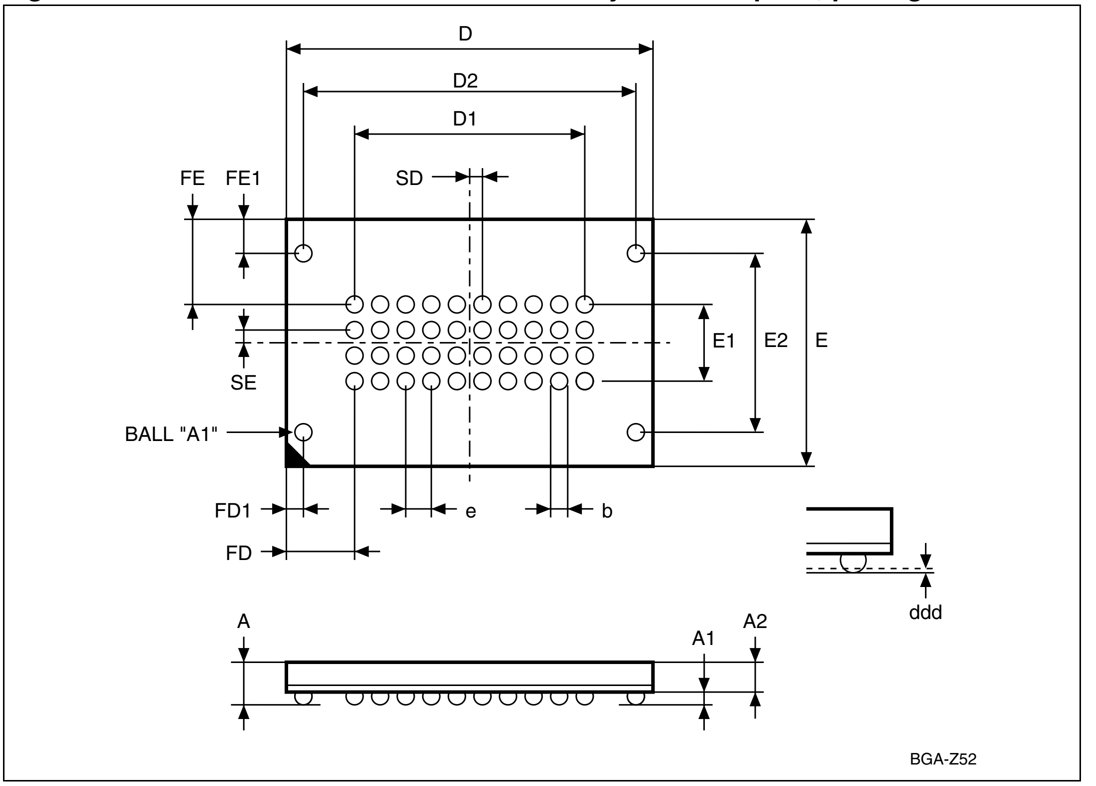


1. Drawing is not to scale.

Table 27. VFBGA44 - 7.7x9mm, 10x4 ball array, 0.5mm pitch package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.00			0.039
A1		0.15			0.006	
A2	0.66			0.026		
b	0.32	0.27	0.37	0.013	0.011	0.015
D	7.70	7.60	7.80	0.303	0.299	0.307
D1	4.50			0.177		
D2	6.50			0.256		
ddd			0.08			0.003
E	9.00	8.90	9.10	0.354	0.350	0.358
E1	1.50			0.059		
E2	3.50			0.138		
e	0.50	–	–	0.020	–	–
FD	1.60			0.063		
FD1	0.60			0.024		
FE	3.75			0.148		
FE1	2.75			0.108		
SD	0.25	–	–	0.010	–	–
SE	0.25	–	–	0.010	–	–

Figure 18. VFBGA44 7.5 × 5mm - 10x4 ball array - 0.50mm pitch, package outline



1. Drawing is not to scale.

Table 28. VFBGA44 7.5 × 5mm - 10x4 ball array - 0.50mm pitch, package mechanical data

Symbol	millimeters			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.000			0.0394
A1		0.150			0.0059	
A2	0.660			0.0260		
b	0.300	0.250	0.350	0.0118	0.0098	0.0138
D	7.500	7.400	7.600	0.2953	0.2913	0.2992
D1	4.500			0.1772		
D2	6.500			0.2559		
ddd			0.080			0.0031
E	5.000	4.900	5.100	0.1969	0.1929	0.2008
E1	1.500			0.0591		
E2	3.500			0.1378		
e	0.500	–	–	0.0197	–	–
FD	1.500			0.0591		
FD1	0.500			0.0197		
FE	1.750			0.0689		
FE1	0.750			0.0295		
SD	0.250			0.0098		
SE	0.250			0.0098		

Figure 19. VFBGA44 Daisy Chain - Package Connections (Top view through package)

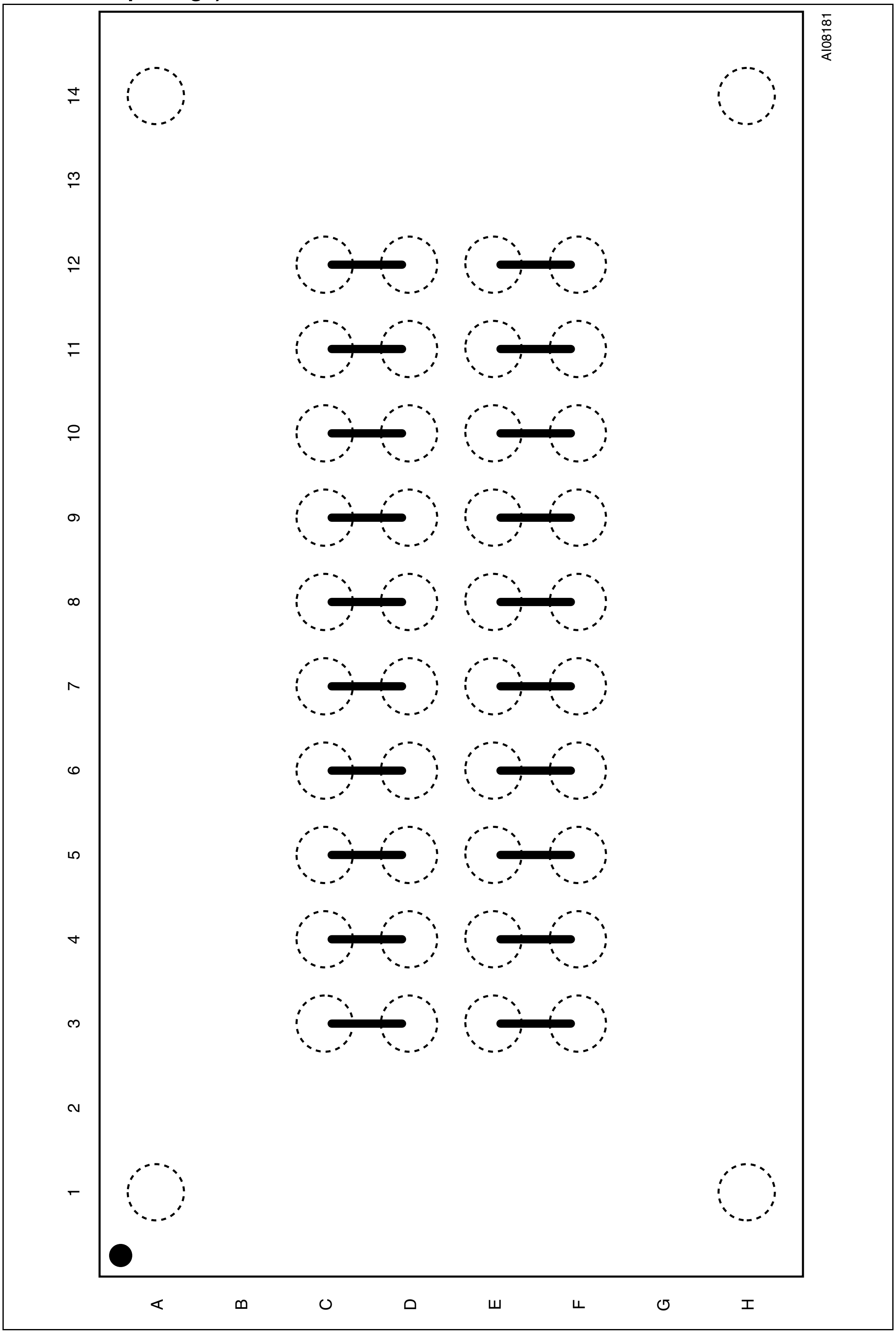
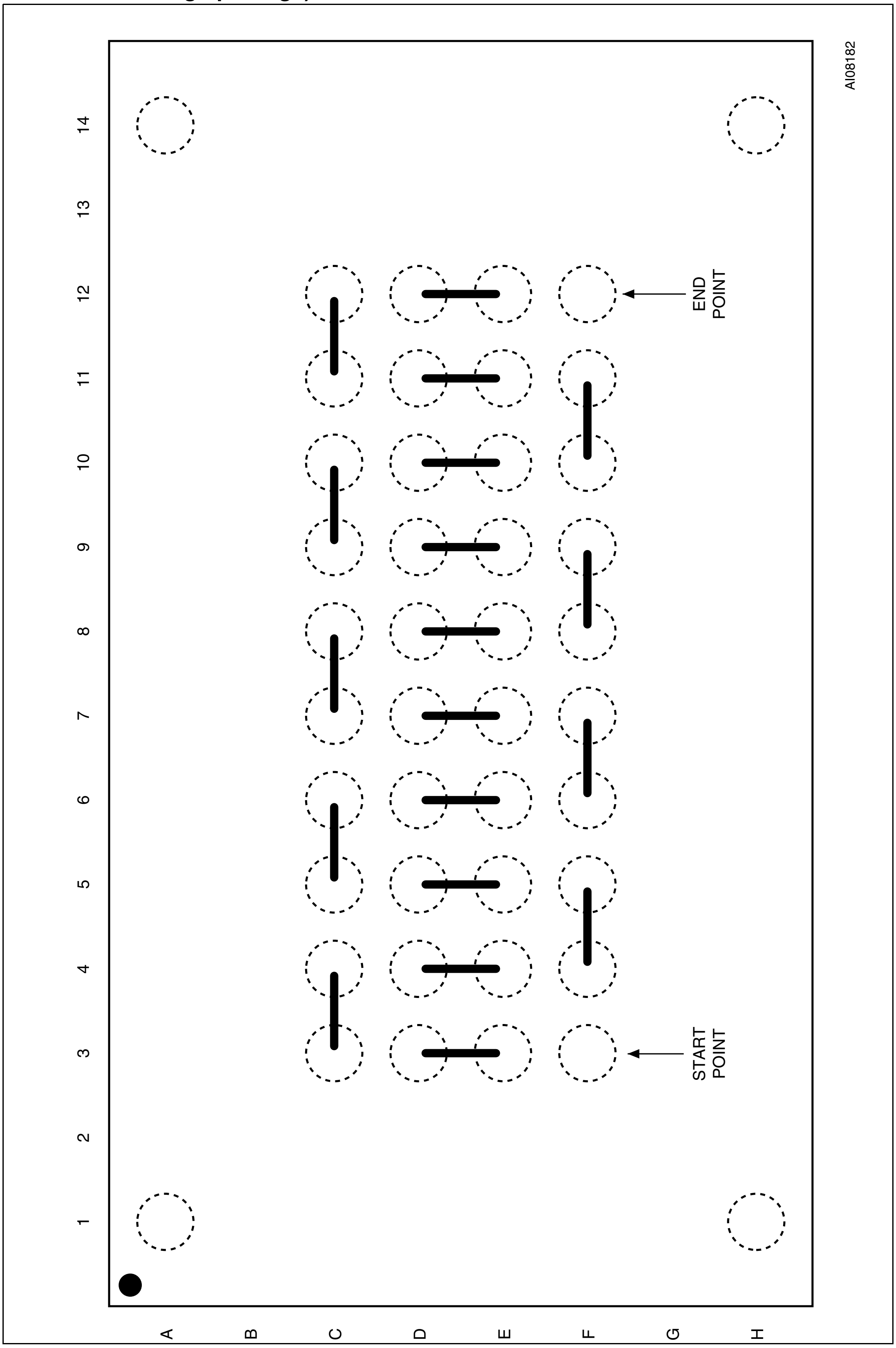


Figure 20. VFBGA44 Daisy Chain - PCB connection proposal (top view through package)



16 Part numbering

Table 29. Ordering information scheme

Example: M58WR064HU 70 ZB 6 U

Device Type

M58

Architecture

W = Multiple Bank, Burst Mode

Operating Voltage

R = V_{DD} = V_{DDQ} = 1.7V to 2V

Density

064 = 64 Mbit

Technology

H = 90nm technology

Parameter Bank Location

U = Top Boot, Multiplexed I/O

L = Bottom Boot, Multiplexed I/O

Speed

70 = 70ns

Package

ZB = VFBGA44: 7.7 x 9mm, 0.5mm pitch

Temperature Range

6 = –40 to 85°C

Option

E = ECOPACK® Package, Standard Packing

U = ECOPACK® Package, Tape & Reel Packing, 16mm

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

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

Appendix A Block address tables

Table 30. Top boot block addresses, M58WR064HU

Bank[(1)]	#	Size (KWord)	Address Range
Parameter Bank	0	4	3FF000-3FFFFF
	1	4	3FE000-3FEFFF
	2	4	3FD000-3FDFFF
	3	4	3FC000-3FCFFF
	4	4	3FB000-3FBFFF
	5	4	3FA000-3FAFFF
	6	4	3F9000-3F9FFF
	7	4	3F8000-3F8FFF
	8	32	3F0000-3F7FFF
	9	32	3E8000-3EFFFF
	10	32	3E0000-3E7FFF
	11	32	3D8000-3DFFFF
	12	32	3D0000-3D7FFF
	13	32	3C8000-3CFFFF
	14	32	3C0000-3C7FFF
Bank 1	15	32	3B8000-3BFFFF
	16	32	3B0000-3B7FFF
	17	32	3A8000-3AFFFF
	18	32	3A0000-3A7FFF
	19	32	398000-39FFFF
	20	32	390000-397FFF
	21	32	388000-38FFFF
	22	32	380000-387FFF
Bank 2	23	32	378000-37FFFF
	24	32	370000-377FFF
	25	32	368000-36FFFF
	26	32	360000-367FFF
	27	32	358000-35FFFF
	28	32	350000-357FFF
	29	32	348000-34FFFF
	30	32	340000-347FFF

Table 30. Top boot block addresses, M58WR064HU (continued)

Bank[(1)]	#	Size (KWord)	Address Range
Bank 3	31	32	338000-33FFFF
	32	32	330000-337FFF
	33	32	328000-32FFFF
	34	32	320000-327FFF
	35	32	318000-31FFFF
	36	32	310000-317FFF
	37	32	308000-30FFFF
	38	32	300000-307FFF
Bank 4	39	32	2F8000-2FFFFF
	40	32	2F0000-2F7FFF
	41	32	2E8000-2EFFFF
	42	32	2E0000-2E7FFF
	43	32	2D8000-2DFFFF
	44	32	2D0000-2D7FFF
	45	32	2C8000-2CFFFF
	46	32	2C0000-2C7FFF
Bank 5	47	32	2B8000-2BFFFF
	48	32	2B0000-2B7FFF
	49	32	2A8000-2AFFFF
	50	32	2A0000-2A7FFF
	51	32	298000-29FFFF
	52	32	290000-297FFF
	53	32	288000-28FFFF
	54	32	280000-287FFF
Bank 6	55	32	278000-27FFFF
	56	32	270000-277FFF
	57	32	268000-26FFFF
	58	32	260000-267FFF
	59	32	258000-25FFFF
	60	32	250000-257FFF
	61	32	248000-24FFFF
	62	32	240000-247FFF
Bank 7	63	32	238000-23FFFF
	64	32	230000-237FFF
	65	32	228000-22FFFF
	66	32	220000-227FFF
	67	32	218000-21FFFF
	68	32	210000-217FFF
	69	32	208000-20FFFF
	70	32	200000-207FFF

Table 30. Top boot block addresses, M58WR064HU (continued)

Bank[1]	#	Size (KWord)	Address Range
Bank 8	71	32	1F8000-1FFFFF
	72	32	1F0000-1F7FFF
	73	32	1E8000-1EFFFF
	74	32	1E0000-1E7FFF
	75	32	1D8000-1DFFFF
	76	32	1D0000-1D7FFF
	77	32	1C8000-1CFFFF
	78	32	1C0000-1C7FFF
Bank 9	79	32	1B8000-1BFFFF
	80	32	1B0000-1B7FFF
	81	32	1A8000-1AFFFF
	82	32	1A0000-1A7FFF
	83	32	198000-19FFFF
	84	32	190000-197FFF
	85	32	188000-18FFFF
	86	32	180000-187FFF
Bank 10	87	32	178000-17FFFF
	88	32	170000-177FFF
	89	32	168000-16FFFF
	90	32	160000-167FFF
	91	32	158000-15FFFF
	92	32	150000-157FFF
	93	32	148000-14FFFF
	94	32	140000-147FFF
Bank 11	95	32	138000-13FFFF
	96	32	130000-137FFF
	97	32	128000-12FFFF
	98	32	120000-127FFF
	99	32	118000-11FFFF
	100	32	110000-117FFF
	101	32	108000-10FFFF
	102	32	100000-107FFF
Bank 12	103	32	0F8000-0FFFFF
	104	32	0F0000-0F7FFF
	105	32	0E8000-0EFFFF
	106	32	0E0000-0E7FFF
	107	32	0D8000-0DFFFF
	108	32	0D0000-0D7FFF
	109	32	0C8000-0CFFFF
	110	32	0C0000-0C7FFF

Table 30. Top boot block addresses, M58WR064HU (continued)

Bank(1)	#	Size (KWord)	Address Range
Bank 13	111	32	0B8000-0BFFFF
	112	32	0B0000-0B7FFF
	113	32	0A8000-0AFFFF
	114	32	0A0000-0A7FFF
	115	32	098000-09FFFF
	116	32	090000-097FFF
	117	32	088000-08FFFF
	118	32	080000-087FFF
Bank 14	119	32	078000-07FFFF
	120	32	070000-077FFF
	121	32	068000-06FFFF
	122	32	060000-067FFF
	123	32	058000-05FFFF
	124	32	050000-057FFF
	125	32	048000-04FFFF
	126	32	040000-047FFF
Bank 15	127	32	038000-03FFFF
	128	32	030000-037FFF
	129	32	028000-02FFFF
	130	32	020000-027FFF
	131	32	018000-01FFFF
	132	32	010000-017FFF
	133	32	008000-00FFFF
	134	32	000000-007FFF

1. There are two Bank Regions: Bank Region 1 contains all the banks that are made up of main blocks only; Bank Region 2 contains the banks that are made up of the parameter and main blocks (Parameter Bank).

Table 31. Bottom boot block addresses, M58WR064HL

Bank(1)	#	Size (KWord)	Address Range
Bank 15	134	32	3F8000-3FFFFF
	133	32	3F0000-3F7FFF
	132	32	3E8000-3EFFFF
	131	32	3E0000-3E7FFF
	130	32	3D8000-3DFFFF
	129	32	3D0000-3D7FFF
	128	32	3C8000-3CFFFF
	127	32	3C0000-3C7FFF

Table 31. Bottom boot block addresses, M58WR064HL (continued)

Bank[(1)]	#	Size (KWord)	Address Range
Bank 14	126	32	3B8000-3BFFFF
	125	32	3B0000-3B7FFF
	124	32	3A8000-3AFFFF
	123	32	3A0000-3A7FFF
	122	32	398000-39FFFF
	121	32	390000-397FFF
	120	32	388000-38FFFF
	119	32	380000-387FFF
Bank 13	118	32	378000-37FFFF
	117	32	370000-377FFF
	116	32	368000-36FFFF
	115	32	360000-367FFF
	114	32	358000-35FFFF
	113	32	350000-357FFF
	112	32	348000-34FFFF
	111	32	340000-347FFF
Bank 12	110	32	338000-33FFFF
	109	32	330000-337FFF
	108	32	328000-32FFFF
	107	32	320000-327FFF
	106	32	318000-31FFFF
	105	32	310000-317FFF
	104	32	308000-30FFFF
	103	32	300000-307FFF
Bank 11	102	32	2F8000-2FFFFF
	101	32	2F0000-2F7FFF
	100	32	2E8000-2EFFFF
	99	32	2E0000-2E7FFF
	98	32	2D8000-2DFFFF
	97	32	2D0000-2D7FFF
	96	32	2C8000-2CFFFF
	95	32	2C0000-2C7FFF

Table 31. Bottom boot block addresses, M58WR064HL (continued)

Bank[(1)]	#	Size (KWord)	Address Range
Bank 10	94	32	2B8000-2BFFFF
	93	32	2B0000-2B7FFF
	92	32	2A8000-2AFFFF
	91	32	2A0000-2A7FFF
	90	32	298000-29FFFF
	89	32	290000-297FFF
	88	32	288000-28FFFF
	87	32	280000-287FFF
Bank 9	86	32	278000-27FFFF
	85	32	270000-277FFF
	84	32	268000-26FFFF
	83	32	260000-267FFF
	82	32	258000-25FFFF
	81	32	250000-257FFF
	80	32	248000-24FFFF
	79	32	240000-247FFF
Bank 8	78	32	238000-23FFFF
	77	32	230000-237FFF
	76	32	228000-22FFFF
	75	32	220000-227FFF
	74	32	218000-21FFFF
	73	32	210000-217FFF
	72	32	208000-20FFFF
	71	32	200000-207FFF
Bank 7	70	32	1F8000-1FFFFF
	69	32	1F0000-1F7FFF
	68	32	1E8000-1EFFFF
	67	32	1E0000-1E7FFF
	66	32	1D8000-1DFFFF
	65	32	1D0000-1D7FFF
	64	32	1C8000-1CFFFF
	63	32	1C0000-1C7FFF

Table 31. Bottom boot block addresses, M58WR064HL (continued)

Bank(1)	#	Size (KWord)	Address Range
Bank 6	62	32	1B8000-1BFFFF
	61	32	1B0000-1B7FFF
	60	32	1A8000-1AFFFF
	59	32	1A0000-1A7FFF
	58	32	198000-19FFFF
	57	32	190000-197FFF
	56	32	188000-18FFFF
	55	32	180000-187FFF
Bank 5	54	32	178000-17FFFF
	53	32	170000-177FFF
	52	32	168000-16FFFF
	51	32	160000-167FFF
	50	32	158000-15FFFF
	49	32	150000-157FFF
	48	32	148000-14FFFF
	47	32	140000-147FFF
Bank 4	46	32	138000-13FFFF
	45	32	130000-137FFF
	44	32	128000-12FFFF
	43	32	120000-127FFF
	42	32	118000-11FFFF
	41	32	110000-117FFF
	40	32	108000-10FFFF
	39	32	100000-107FFF
Bank 3	38	32	0F8000-0FFFFF
	37	32	0F0000-0F7FFF
	36	32	0E8000-0EFFFF
	35	32	0E0000-0E7FFF
	34	32	0D8000-0DFFFF
	33	32	0D0000-0D7FFF
	32	32	0C8000-0CFFFF
	31	32	0C0000-0C7FFF

Table 31. Bottom boot block addresses, M58WR064HL (continued)

Bank[(1)]	#	Size (KWord)	Address Range
Bank 2	30	32	0B8000-0BFFFF
	29	32	0B0000-0B7FFF
	28	32	0A8000-0AFFFF
	27	32	0A0000-0A7FFF
	26	32	098000-09FFFF
	25	32	090000-097FFF
	24	32	088000-08FFFF
	23	32	080000-087FFF
Bank 1	22	32	078000-07FFFF
	21	32	070000-077FFF
	20	32	068000-06FFFF
	19	32	060000-067FFF
	18	32	058000-05FFFF
	17	32	050000-057FFF
	16	32	048000-04FFFF
	15	32	040000-047FFF
Parameter Bank	14	32	038000-03FFFF
	13	32	030000-037FFF
	12	32	028000-02FFFF
	11	32	020000-027FFF
	10	32	018000-01FFFF
	9	32	010000-017FFF
	8	32	008000-00FFFF
	7	4	007000-007FFF
	6	4	006000-006FFF
	5	4	005000-005FFF
	4	4	004000-004FFF
	3	4	003000-003FFF
	2	4	002000-002FFF
	1	4	001000-001FFF
	0	4	000000-000FFF

1. There are two Bank Regions: Bank Region 2 contains all the banks that are made up of main blocks only; Bank Region 1 contains the banks that are made up of the parameter and main blocks (Parameter Bank).

Appendix B Common Flash Interface

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

When the Read CFI Query Command is issued the device enters CFI Query mode and the data structure is read from the memory. Tables *32*, *33*, *34*, *35*, *36*, *37*, *38*, *39*, *40* and *41* show the addresses used to retrieve the data. The Query data is always presented on the lowest order data outputs (DQ0-DQ7), the other outputs (DQ8-DQ15) are set to 0.

The CFI data structure also contains a security area where a 64 bit unique security number is written (see *Figure 4: Protection Register memory map*). 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. Issue a Read Array command to return to Read mode.

Table 32. Query structure overview(1)

Offset	Sub-section Name	Description
00h	Reserved	Reserved for algorithm-specific information
10h	CFI Query Identification String	Command set ID and algorithm data offset
1Bh	System Interface Information	Device timing & voltage information
27h	Device Geometry Definition	Flash device layout
P	Primary Algorithm-specific Extended Query table	Additional information specific to the Primary Algorithm (optional)
A	Alternate Algorithm-specific Extended Query table	Additional information specific to the Alternate Algorithm (optional)
80h	Security Code Area	Lock Protection Register Unique device Number and User Programmable OTP

1. The Flash memory display the CFI data structure when CFI Query command is issued. In this table are listed the main sub-sections detailed in Tables *33*, *34*, *35* and *36*. Query data is always presented on the lowest order data outputs.

Table 33. CFI Query Identification String

Offset	Sub-section Name	Description		Value
00h	0020h	Manufacturer Code		Numonyx
01h	88C0h 88C1h	Device Code	M58WR064HU M58WR064HL	Top Bottom
02h	reserved	Reserved		
03h	DRC	Die Revision Code		
04h-0Fh	reserved	Reserved		
10h	0051h	Query Unique ASCII String "QRY"		"Q"
11h	0052h			"R"
12h	0059h			"Y"
13h 14h	0003h 0000h	Primary Algorithm Command Set and Control Interface ID code 16 bit ID code defining a specific algorithm		
15h 16h	offset = P = 0039h 0000h	Address for Primary Algorithm extended Query table (see *Table 35*)		p = 39h
17h 18h	0000h 0000h	Alternate Vendor Command Set and Control Interface ID Code second vendor - specified algorithm supported		NA
19h 1Ah	value = A = 0000h 0000h	Address for Alternate Algorithm extended Query table		NA

Table 34. CFI Query system interface information

Offset	Data	Description	Value
1Bh	0017h	V_{DD} Logic Supply Minimum Program/Erase or Write voltage bit 7 to 4 BCD value in volts bit 3 to 0 BCD value in 100 millivolts	1.7V
1Ch	0020h	V_{DD} Logic Supply Maximum Program/Erase or Write voltage bit 7 to 4 BCD value in volts bit 3 to 0 BCD value in 100 millivolts	2V
1Dh	00B4h	V_{PP} [Programming] Supply Minimum Program/Erase voltage bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 millivolts	11.4V
1Eh	00C6h	V_{PP} [Programming] Supply Maximum Program/Erase voltage bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 millivolts	12.6V
1Fh	0004h	Typical time-out per single byte/word program = 2^n µs	16µs
20h	0000h	Typical time-out for multi-Byte programming = 2^n µs	NA
21h	000Ah	Typical time-out per individual block erase = 2^n ms	1s
22h	0000h	Typical time-out for full chip erase = 2^n ms	NA
23h	0003h	Maximum time-out for word program = 2^n times typical	128µs
24h	0000h	Maximum time-out for multi-Byte programming = 2^n times typical	NA
25h	0002h	Maximum time-out per individual block erase = 2^n times typical	4s
26h	0000h	Maximum time-out for chip erase = 2^n times typical	NA

Table 35. Device geometry definition

	Offset Word Mode	Data	Description	Value
	27h	0017h	Device Size = 2^n in number of bytes	8 MBytes
	28h 29h	0001h 0000h	Flash Device Interface Code description	x16 Async.
	2Ah 2Bh	0000h 0000h	Maximum number of bytes in multi-byte program or page = 2^n	NA
	2Ch	0002h	Number of identical sized erase block regions within the device bit 7 to 0 = x = number of Erase Block Regions	2
M58WR064HU	2Dh 2Eh	007Eh 0000h	Region 1 Information Number of identical-size erase blocks = 007Eh+1	127
M58WR064HU	2Fh 30h	0000h 0001h	Region 1 Information Block size in Region 1 = 0100h * 256 byte	64 KByte
M58WR064HU	31h 32h	0007h 0000h	Region 2 Information Number of identical-size erase blocks = 0007h+1	8
M58WR064HU	33h 34h	0020h 0000h	Region 2 Information Block size in Region 2 = 0020h * 256 byte	8 KByte
M58WR064HU	35h 38h	Reserved for future erase block region information		NA
M58WR064HL	2Dh 2Eh	0007h 0000h	Region 1 Information Number of identical-size erase block = 0007h+1	8
M58WR064HL	2Fh 30h	0020h 0000h	Region 1 Information Block size in Region 1 = 0020h * 256 byte	8 KByte
M58WR064HL	31h 32h	007Eh 0000h	Region 2 Information Number of identical-size erase block = 007Eh+1	127
M58WR064HL	33h 34h	0000h 0001h	Region 2 Information Block size in Region 2 = 0100h * 256 byte	64 KByte
M58WR064HL	35h 38h	Reserved for future erase block region information		NA

Table 36. Primary algorithm-specific extended query table

Offset	Data	Description	Value
(P)h = 39h	0050h	Primary Algorithm extended Query table unique ASCII string "PRI"	"P"
	0052h		"R"
	0049h		"I"
(P+3)h = 3Ch	0031h	Major version number, ASCII	"1"
(P+4)h = 3Dh	0033h	Minor version number, ASCII	"3"
(P+5)h = 3Eh	00E6h	Extended Query table contents for Primary Algorithm. Address (P+5)h contains less significant byte.	
	0003h	bit 0 Chip Erase supported (1 = Yes, 0 = No)	No
(P+7)h = 40h	0000h	bit 1 Erase Suspend supported (1 = Yes, 0 = No)	Yes
(P+8)h = 41h	0000h	bit 2 Program Suspend supported (1 = Yes, 0 = No) bit 3 Legacy Lock/Unlock supported (1 = Yes, 0 = No) bit 4 Queued Erase supported (1 = Yes, 0 = No) bit 5 Instant individual block locking supported (1 = Yes, 0 = No) bit 6 Protection bits supported (1 = Yes, 0 = No) bit 7 Page mode read supported (1 = Yes, 0 = No) bit 8 Synchronous read supported (1 = Yes, 0 = No) bit 9 Simultaneous operation supported (1 = Yes, 0 = No) bit 10 to 31 Reserved; undefined bits are '0'. If bit 31 is '1' then another 31 bit field of optional features follows at the end of the bit-30 field.	Yes No No Yes Yes Yes Yes Yes
(P+9)h = 42h	0001h	Supported Functions after Suspend Read Array, Read Status Register and CFI Query bit 0 Program supported after Erase Suspend (1 = Yes, 0 = No) bit 7 to 1 Reserved; undefined bits are '0'	Yes
(P+A)h = 43h	0003h	Block Protect Status Defines which bits in the Block Status Register section of the Query are implemented.	
(P+B)h = 44h	0000h	bit 0 Block protect Status Register Lock/Unlock bit active(1 = Yes, 0 = No) bit 1 Block Lock Status Register Lock-Down bit active (1 = Yes, 0 = No) bit 15 to 2 Reserved for future use; undefined bits are '0'	Yes Yes
(P+C)h = 45h	0018h	V_{DD} Logic Supply Optimum Program/Erase voltage (highest performance) bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 mV	1.8V
(P+D)h = 46h	00C0h	V_{PP} Supply Optimum Program/Erase voltage bit 7 to 4 HEX value in volts bit 3 to 0 BCD value in 100 mV	12V

Table 37. Protection Register information

Offset	Data	Description	Value
(P+E)h = 47h	0001h	Number of protection register fields in JEDEC ID space. 0000h indicates that 256 fields are available.	1
(P+F)h = 48h	0080h	Protection Field 1: Protection Description	0080h
(P+10)h = 49h	0000h	Bits 0-7 Lower byte of protection register address Bits 8-15 Upper byte of protection register address	
(P+11)h = 4Ah	0003h	Bits 16-23 2^n bytes in factory pre-programmed region	8 Bytes
(P+12)h= 4Bh	0004h	Bits 24-31 2^n bytes in user programmable region	16 Bytes

Table 38. Burst read information

Offset	Data	Description	Value
(P+13)h = 4Ch	0003h	Page-mode read capability bits 0-7'n' such that 2^n HEX value represents the number of read-page bytes. See offset 28h for device word width to determine page-mode data output width.	8 Bytes
(P+14)h = 4Dh	0004h	Number of synchronous mode read configuration fields that follow.	4
(P+15)h = 4Eh	0001h	Synchronous mode read capability configuration 1 bit 3-7 Reserved bit 0-2 'n' such that 2^{n+1} HEX value represents the maximum number of continuous synchronous reads when the device is configured for its maximum word width. A value of 07h indicates that the device is capable of continuous linear bursts that will output data until the internal burst counter reaches the end of the device's burstable address space. This field's 3-bit value can be written directly to the read configuration register bit 0-2 if the device is configured for its maximum word width. See offset 28h for word width to determine the burst data output width.	4
(P+16)h = 4Fh	0002h	Synchronous mode read capability configuration 2	8
(P+17)h = 50h	0003h	Synchronous mode read capability configuration 3	16
(P+18)h = 51h	0007h	Synchronous mode read capability configuration 4	Cont.

Table 39. Bank and erase block region information(1) (2)

M58WR064HU		M58WR064HL		Description
Offset	Data	Offset	Data	
(P+19)h = 52h	02h	(P+19)h = 52h	02h	Number of Bank Regions within the device

1. The variable P is a pointer which is defined at CFI offset 15h.
2. Bank Regions. There are two Bank Regions, see *Appendix A*, *Table 30* and *Table 31*

Table 40. Bank and erase block region 1 information(1)

M58WR064HU		M58WR064HL		Description
Offset	Data	Offset	Data	
(P+1A)h = 53h	0Fh	(P+1A)h = 53h	01h	Number of identical banks within Bank Region 1
(P+1B)h = 54h	00h	(P+1B)h = 54h	00h	
(P+1C)h = 55h	11h	(P+1C)h = 55h	11h	Number of program or erase operations allowed in Bank Region 1: Bits 0-3: Number of simultaneous program operations Bits 4-7: Number of simultaneous erase operations
(P+1D)h = 56h	00h	(P+1D)h = 56h	00h	Number of program or erase operations allowed in other banks while a bank in same region is programming Bits 0-3: Number of simultaneous program operations Bits 4-7: Number of simultaneous erase operations
(P+1E)h = 57h	00h	(P+1E)h = 57h	00h	Number of program or erase operations allowed in other banks while a bank in this region is erasing Bits 0-3: Number of simultaneous program operations Bits 4-7: Number of simultaneous erase operations
(P+1F)h = 58h	01h	(P+1F)h = 58h	02h	Types of erase block regions in Bank Region 1 n = number of erase block regions with contiguous same-size erase blocks. Symmetrically blocked banks have one blocking region.(2)
(P+20)h = 59h	07h	(P+20)h = 59h	07h	Bank Region 1 Erase Block Type 1 Information Bits 0-15: n+1 = number of identical-sized erase blocks Bits 16-31: n×256 = number of bytes in erase block region
(P+21)h = 5Ah	00h	(P+21)h = 5Ah	00h	
(P+22)h = 5Bh	00h	(P+22)h = 5Bh	20h	
(P+23)h = 5Ch	01h	(P+23)h = 5Ch	00h	
(P+24)h = 5Dh	64h	(P+24)h = 5Dh	64h	Bank Region 1 (Erase Block Type 1) Minimum block erase cycles × 1000
(P+25)h = 5Eh	00h	(P+25)h = 5Eh	00h	
(P+26)h = 5Fh	01h	(P+26)h = 5Fh	01h	Bank Region 1 (Erase Block Type 1): BIts per cell, internal ECC Bits 0-3: bits per cell in erase region Bit 4: reserved for “internal ECC used” BIts 5-7: reserved 5Eh 01 5Eh 01
(P+27)h = 60h	03h	(P+27)h = 60h	03h	Bank Region 1 (Erase Block Type 1): Page mode and synchronous mode capabilities Bit 0: Page-mode reads permitted Bit 1: Synchronous reads permitted Bit 2: Synchronous writes permitted Bits 3-7: reserved

Table 40. Bank and erase block region 1 information[1] (continued)

M58WR064HU		M58WR064HL		Description
Offset	Data	Offset	Data	
		(P+28)h = 61h	06h	Bank Region 1 Erase Block Type 2 Information
		(P+29)h = 62h	00h	Bits 0-15: n+1 = number of identical-sized erase blocks Bits 16-31: n×256 = number of bytes in erase block region
		(P+2A)h = 63h	00h	
		(P+2B)h = 64h	01h	
		(P+2C)h = 65h	64h	Bank Region 1 (Erase Block Type 2) Minimum block erase cycles × 1000
		(P+2D)h = 66h	00h	
		(P+2E)h = 67h	01h	Bank Regions 1 (Erase Block Type 2): BIts per cell, internal ECC Bits 0-3: bits per cell in erase region Bit 4: reserved for “internal ECC used” BIts 5-7: reserved
		(P+2F)h = 68h	03h	Bank Region 1 (Erase Block Type 2): Page mode and synchronous mode capabilities Bit 0: Page-mode reads permitted Bit 1: Synchronous reads permitted Bit 2: Synchronous writes permitted Bits 3-7: reserved

1. The variable P is a pointer which is defined at CFI offset 15h.
2. Bank Regions. There are two Bank Regions, see *Appendix A*, *Table 30* and *Table 31*.

Table 41. Bank and erase block region 2 information(1)

M58WR064HU		M58WR064HL		Description
Offset	Data	Offset	Data	
(P+28)h = 61h	01h	(P+30)h = 69h	0Fh	Number of identical banks within Bank Region 2
(P+29)h = 62h	00h	(P+31)h = 6Ah	00h	
(P+2A)h = 63h	11h	(P+32)h = 6Bh	11h	Number of program or erase operations allowed in Bank Region 2: Bits 0-3: Number of simultaneous program operations Bits 4-7: Number of simultaneous erase operations
(P+2B)h = 64h	00h	(P+33)h = 6Ch	00h	Number of program or erase operations allowed in other banks while a bank in this region is programming Bits 0-3: Number of simultaneous program operations Bits 4-7: Number of simultaneous erase operations
(P+2C)h = 65h	00h	(P+34)h = 6Dh	00h	Number of program or erase operations allowed in other banks while a bank in this region is erasing Bits 0-3: Number of simultaneous program operations Bits 4-7: Number of simultaneous erase operations
(P+2D)h = 66h	02h	(P+35)h = 6Eh	01h	Types of erase block regions in Bank Region 2 n = number of erase block regions with contiguous same-size erase blocks. Symmetrically blocked banks have one blocking region.(2)
(P+2E)h = 67h	06h	(P+36)h = 6Fh	07h	Bank Region 2 Erase Block Type 1 Information Bits 0-15: n+1 = number of identical-sized erase blocks Bits 16-31: n×256 = number of bytes in erase block region
(P+2F)h = 68h	00h	(P+37)h = 70h	00h	
(P+30)h = 69h	00h	(P+38)h = 71h	00h	
(P+31)h = 6Ah	01h	(P+39)h = 72h	01h	
(P+32)h = 6Bh	64h	(P+3A)h = 73h	64h	Bank Region 2 (Erase Block Type 1) Minimum block erase cycles × 1000
(P+33)h = 6Ch	00h	(P+3B)h = 74h	00h	
(P+34)h = 6Dh	01h	(P+3C)h = 75h	01h	Bank Region 2 (Erase Block Type 1): BIts per cell, internal ECC Bits 0-3: bits per cell in erase region Bit 4: reserved for “internal ECC used” BIts 5-7: reserved
(P+35)h = 6Eh	03h	(P+3D)h = 76h	03h	Bank Region 2 (Erase Block Type 1): Page mode and synchronous mode capabilities (defined in *Table 38*) Bit 0: Page-mode reads permitted Bit 1: Synchronous reads permitted Bit 2: Synchronous writes permitted Bits 3-7: reserved
(P+36)h = 6Fh	07h			Bank Region 2 Erase Block Type 2 Information Bits 0-15: n+1 = number of identical-sized erase blocks Bits 16-31: n×256 = number of bytes in erase block region
(P+37)h = 70h	00h			
(P+38)h = 71h	20h			
(P+39)h = 72h	00h			

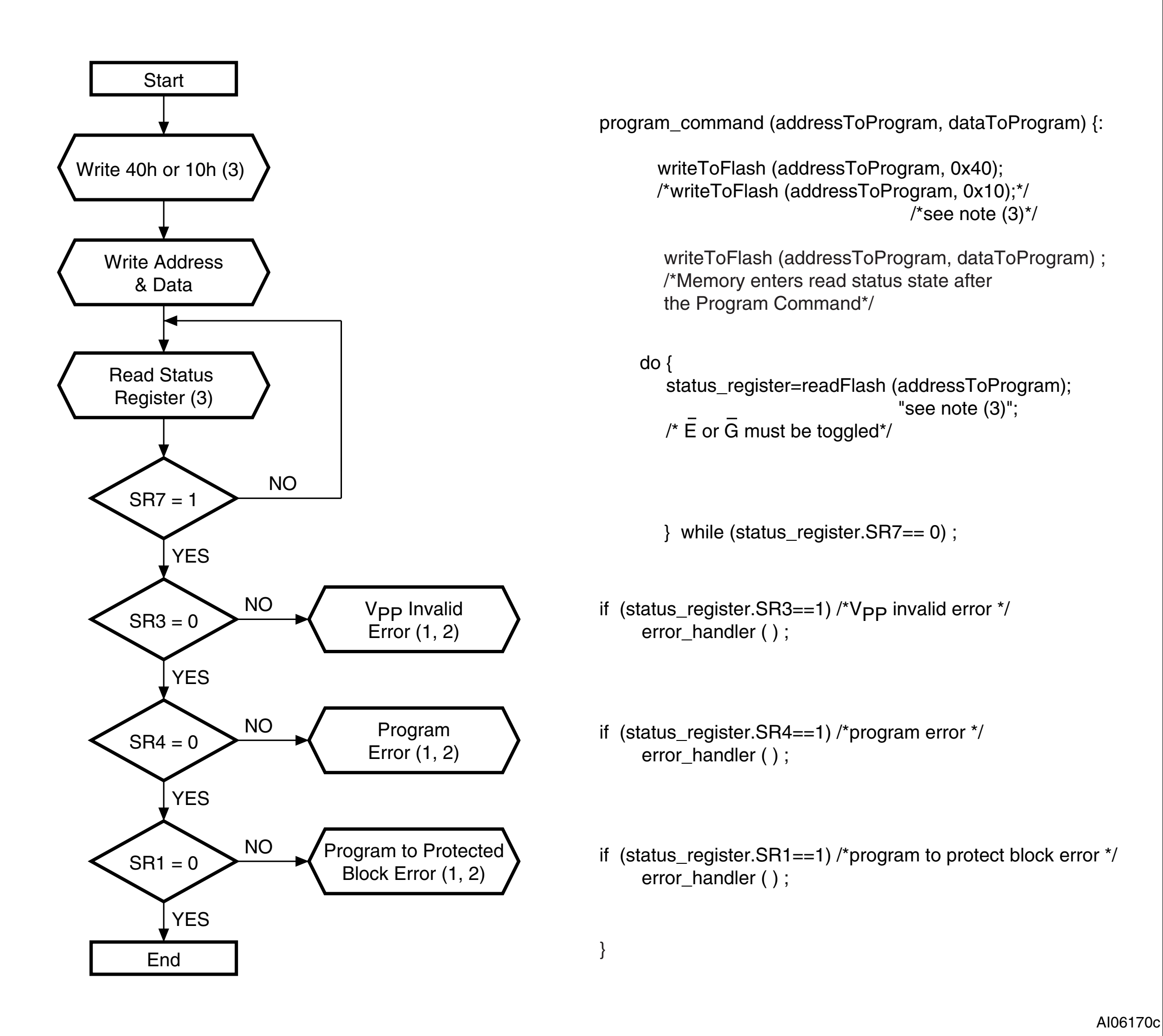
Table 41. Bank and erase block region 2 information(1) (continued)

M58WR064HU		M58WR064HL		Description
Offset	Data	Offset	Data	
(P+3A)h = 73h	64h			Bank Region 2 (Erase Block Type 2) Minimum block erase cycles × 1000
(P+3B)h = 74h	00h			
(P+3C)h = 75h	01h			Bank Region 2 (Erase Block Type 2): BIts per cell, internal ECC Bits 0-3: bits per cell in erase region Bit 4: reserved for “internal ECC used” BIts 5-7: reserved
(P+3D)h = 76h	03h			Bank Region 2 (Erase Block Type 2): Page mode and synchronous mode capabilities (defined in *Table 38*) Bit 0: Page-mode reads permitted Bit 1: Synchronous reads permitted Bit 2: Synchronous writes permitted Bits 3-7: reserved
(P+3E)h = 77h		(P+3E)h = 77h		Feature Space definitions
(P+3F)h = 78h		(P+3F)h = 78h		Reserved

1. The variable P is a pointer which is defined at CFI offset 15h.
2. Bank Regions. There are two Bank Regions, see *Appendix A*, *Table 30* and *Table 31*.

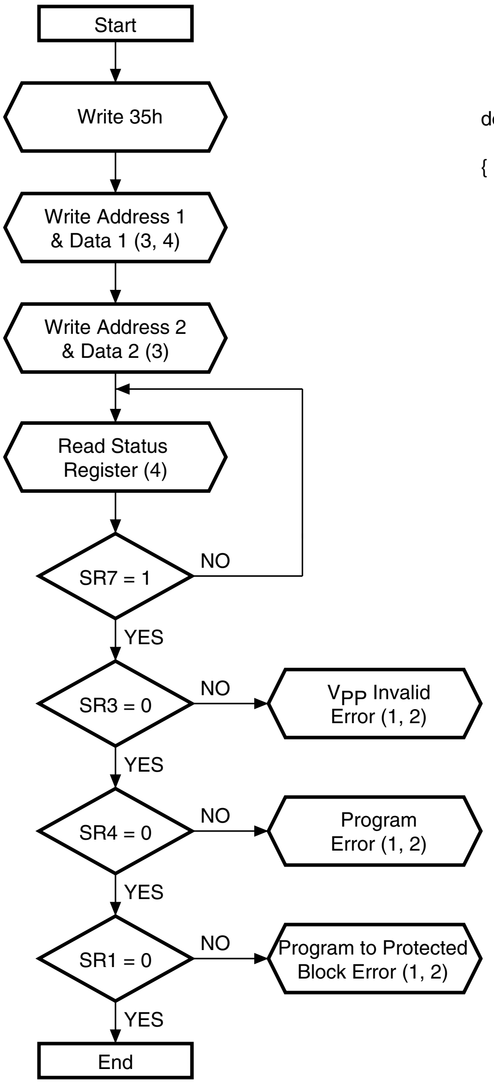
Appendix C Flowcharts and pseudo codes

Figure 21. Program flowchart and pseudo code



1. Status check of SR1 (Protected Block), SR3 (V_{PP} Invalid) and SR4 (Program Error) can be made after each program operation or after a sequence.
2. If an error is found, the Status Register must be cleared before further Program/Erase Controller operations.
3. Any address within the bank can equally be used.

Figure 22. Double Word Program flowchart and pseudo code

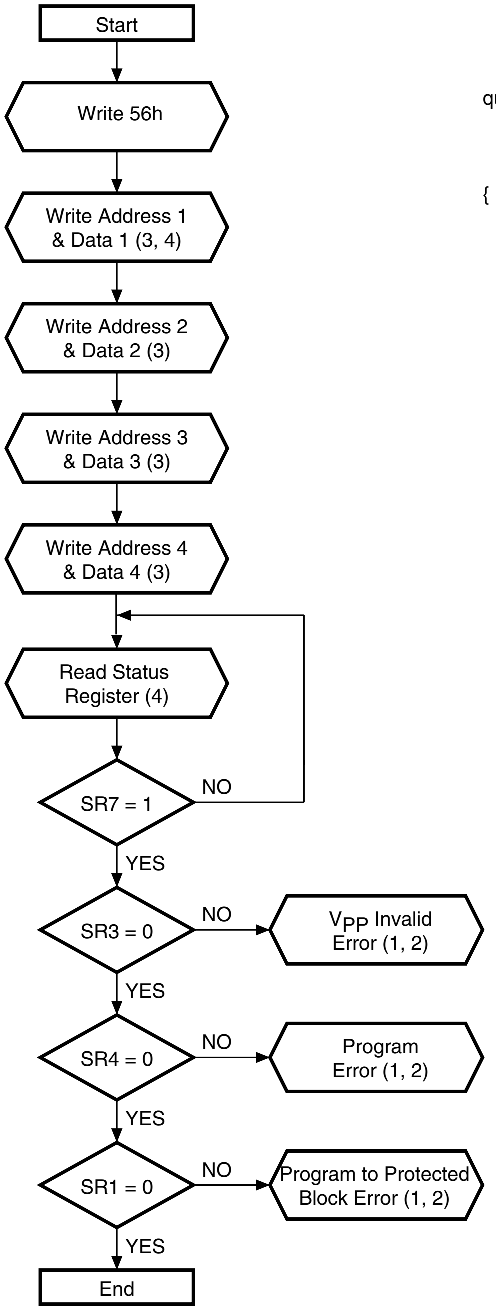



```
double_word_program_command (addressToProgram1, dataToProgram1,
                             addressToProgram2, dataToProgram2)
{
        writeToFlash (addressToProgram1, 0x35);
                    /*see note (4)*/
        writeToFlash (addressToProgram1, dataToProgram1) ;
                    /*see note (3) */
        writeToFlash (addressToProgram2, dataToProgram2) ;
                    /*see note (3) */
        /*Memory enters read status state after
        the Program command*/

      do {
        status_register=readFlash (addressToProgram) ;
                    "see note (4)"
        /* E or G must be toggled*/

        } while (status_register.SR7== 0) ;

  if (status_register.SR3==1) /*VPP invalid error */
        error_handler ( ) ;

  if (status_register.SR4==1) /*program error */
        error_handler ( ) ;

  if (status_register.SR1==1) /*program to protect block error */
        error_handler ( ) ;

  }
```

AI06171b

1. Status check of b1 (Protected Block), b3 (V_{PP} Invalid) and b4 (Program Error) can be made after each program operation or after a sequence.
2. If an error is found, the Status Register must be cleared before further Program/Erase operations.
3. Address 1 and Address 2 must be consecutive addresses differing only for bit A0.
4. Any address within the bank can equally be used.

Figure 23. Quadruple Word Program flowchart and pseudo code

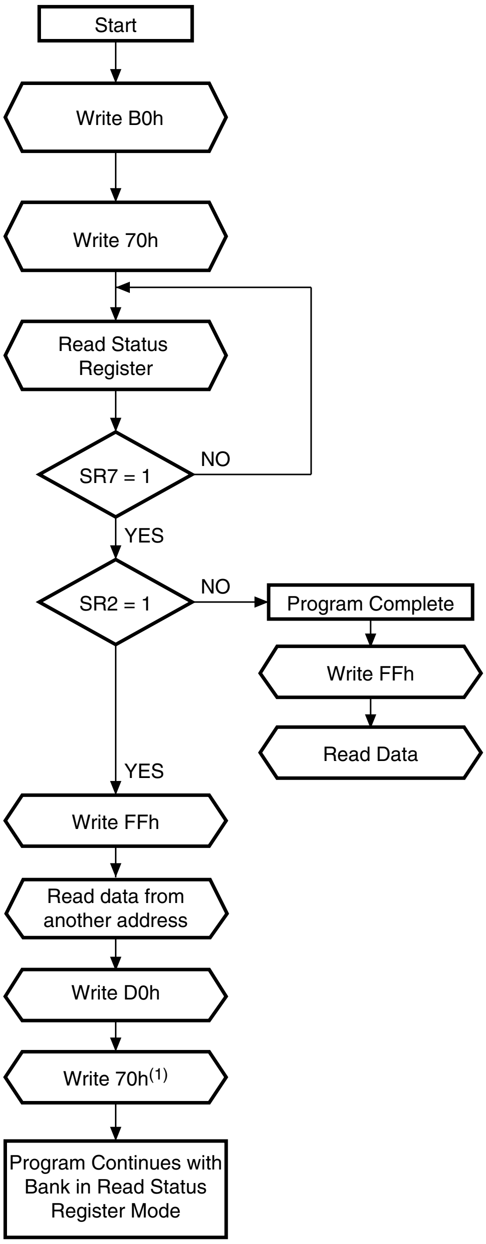



```
quadruple_word_program_command (addressToProgram1, dataToProgram1,
                                addressToProgram2, dataToProgram2,
                                addressToProgram3, dataToProgram3,
                                addressToProgram4, dataToProgram4)
{
        writeToFlash (addressToProgram1, 0x56);
                        /*see note (4) */

        writeToFlash (addressToProgram1, dataToProgram1) ;
                        /*see note (3) */

        writeToFlash (addressToProgram2, dataToProgram2) ;
                        /*see note (3) */

        writeToFlash (addressToProgram3, dataToProgram3) ;
                        /*see note (3) */

        writeToFlash (addressToProgram4, dataToProgram4) ;
                        /*see note (3) */

        /*Memory enters read status state after
        the Program command*/
      do {
        status_register=readFlash (addressToProgram) ;
                        /"see note (4) "/
        /* E or G must be toggled*/

        } while (status_register.SR7== 0) ;

  if (status_register.SR3==1) /*VPP invalid error */
      error_handler ( ) ;

  if (status_register.SR4==1) /*program error */
      error_handler ( ) ;

  if (status_register.SR==1) /*program to protect block error */
      error_handler ( ) ;

  }
```

AI06977b

1. Status check of SR1 (Protected Block), SR3 (V_{PP} Invalid) and SR4 (Program Error) can be made after each program operation or after a sequence.
2. If an error is found, the Status Register must be cleared before further Program/Erase operations.
3. Address 1 to Address 4 must be consecutive addresses differing only for bits A0 and A1.
4. Any address within the bank can equally be used.

Figure 24. Program Suspend & Resume flowchart and pseudo code

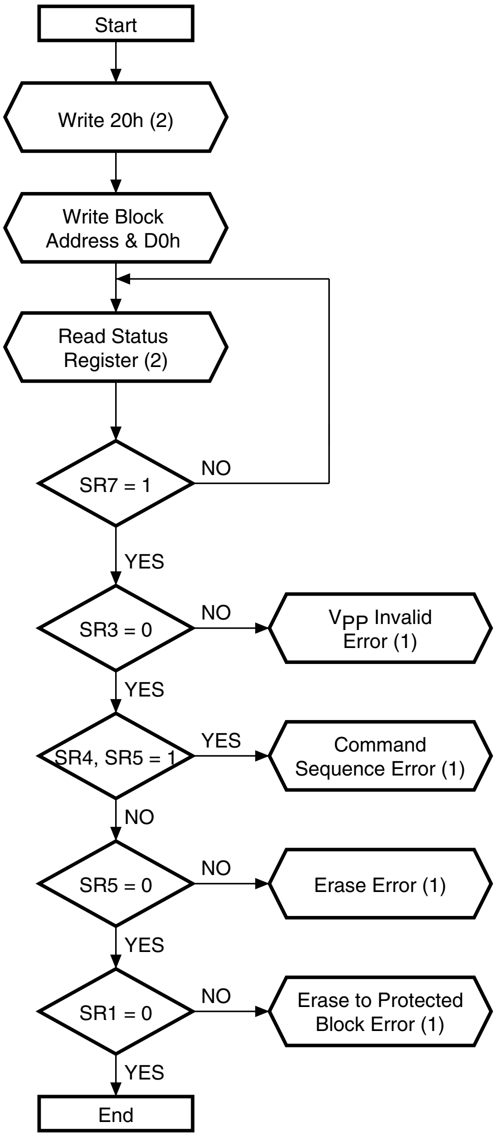



```
program_suspend_command  ( ) {
      writeToFlash (any_address, 0xB0) ;

      writeToFlash (bank_address, 0x70) ;
      /* read status register to check if
      program has already completed */

  do {
      status_register=readFlash (bank_address) ;
      /* Ē or Ḡ must be toggled*/

      }  while (status_register.SR7== 0) ;

if  (status_register.SR2==0)  /*program completed */

      {  writeToFlash (bank_address, 0xFF)  ;
         read_data ( ) ;
         /*The device returns to Read Array
         (as if program/erase suspend was not issued).*/
       }
else
      {  writeToFlash (bank_address, 0xFF)  ;

         read_data ( ); /*read data from another address*/

         writeToFlash (any_address, 0xD0) ;
         /*write 0xD0 to resume program*/

         writeToFlash (bank_address, 0x70) ;
         /*read status register to check if program has completed */

       }
}
```

AI10117b

1. The Read Status Register command (Write 70h) can be issued just before or just after the Program Resume command.

Figure 25. Block Erase flowchart and pseudo code

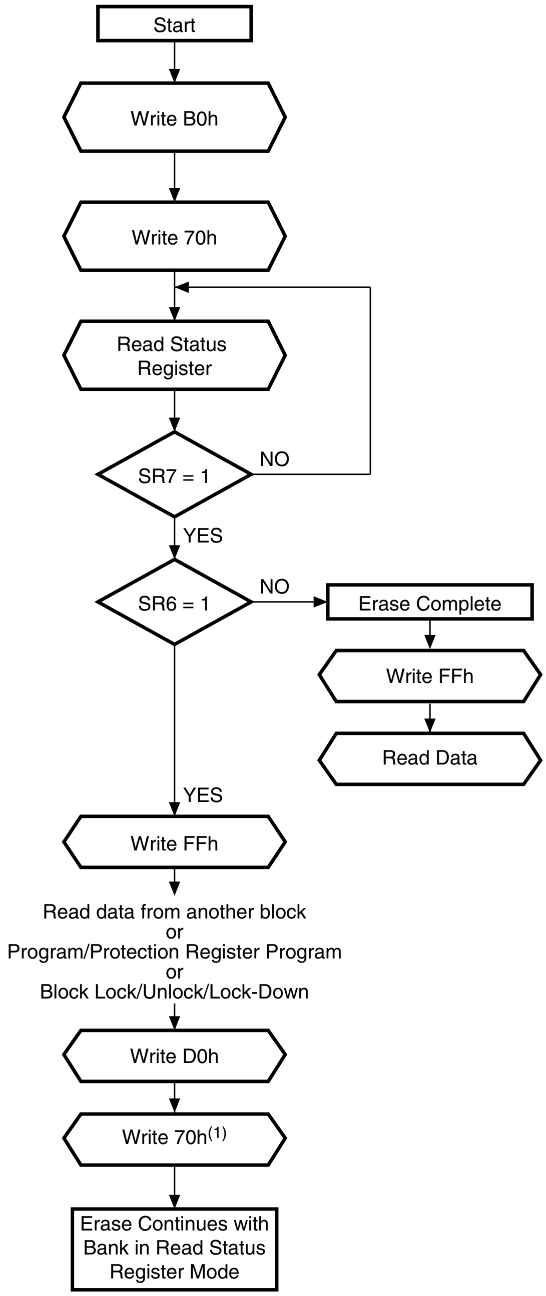



```
erase_command  ( blockToErase ) {
       writeToFlash (blockToErase, 0x20) ;
                              /*see note (2) */

       writeToFlash (blockToErase, 0xD0) ;
       /* only ADQ12-ADQ15 and A16-A21 significant */
       /* Memory enters read status state after
       the Erase Command */

    do {
       status_register=readFlash (blockToErase) ;
                                          /* see note (2) */
       /* Ē or Ḡ must be toggled*/

       }  while (status_register.SR7== 0) ;

if  (status_register.SR3==1) /*VPP invalid error */
      error_handler ( ) ;

if  ( (status_register.SR4==1) && (status_register.SR5==1) )
/* command sequence error */
      error_handler ( ) ;

if  ( (status_register.SR5==1) )
/* erase error */
      error_handler ( ) ;

if  (status_register.SR1==1) /*program to protect block error */
      error_handler ( ) ;

}
```

AI10567b

1. If an error is found, the Status Register must be cleared before further Program/Erase operations.
2. Any address within the bank can equally be used.

Figure 26. Erase Suspend & Resume flowchart and pseudo code

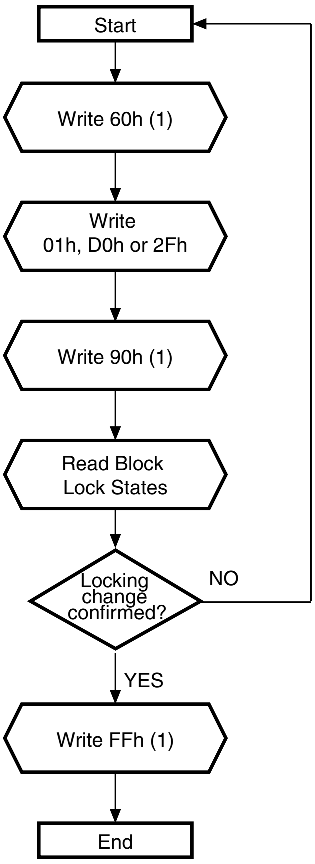



```
erase_suspend_command  ( ) {
      writeToFlash (bank_address, 0xB0) ;

      writeToFlash (bank_address, 0x70) ;
      /* read status register to check if
      erase has already completed */

    do {
       status_register=readFlash (bank_address) ;
       /* Ē or Ḡ must be toggled*/

      }  while (status_register.SR7== 0) ;

if  (status_register.SR6==0)  /*erase completed */

      {  writeToFlash (bank_address, 0xFF)  ;

        read_data ( ) ;
        /*The device returns to Read Array
        (as if program/erase suspend was not issued).*/
      }
else
      {  writeToFlash (bank_address, 0xFF)  ;

        read_program_data ( );

        /*read or program data from another block*/

        writeToFlash (bank_address, 0xD0) ;
        /*write 0xD0 to resume erase*/

        writeToFlash (bank_address, 0x70) ;
        /*read status register to check if erase has completed */
      }
}
```

AI10116b

1. The Read Status Register command (Write 70h) can be issued just before or just after the Erase Resume command.

Figure 27. Locking Operations flowchart and pseudo code

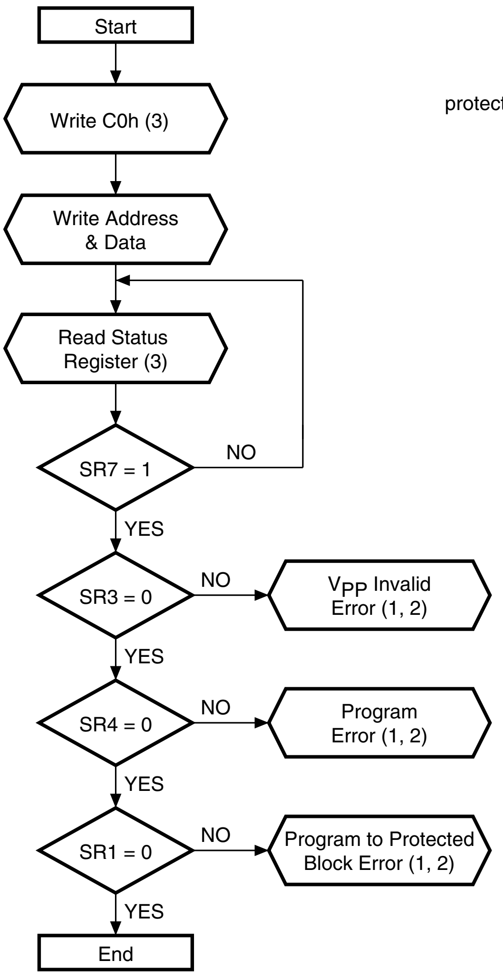



```
locking_operation_command   (address, lock_operation)  {
        writeToFlash (address, 0x60) ; /*configuration setup*/
                                        /* see note (1) */

        if  (lock_operation==LOCK)  /*to protect the block*/
                writeToFlash (address, 0x01)  ;
        else if  (lock_operation==UNLOCK)  /*to unprotect the block*/
                writeToFlash (address, 0xD0)  ;
        else if  (lock_operation==LOCK-DOWN)  /*to lock the block*/
                writeToFlash (address, 0x2F)  ;

                writeToFlash (address, 0x90) ;
                                /*see note (1) */

        if  (readFlash (address) ! = locking_state_expected)
                error_handler () ;
                /*Check the locking state (see Read Block Signature table )*/

        writeToFlash (address, 0xFF) ; /*Reset to Read Array mode*/
                                /*see note (1) */

}
```

AI06176b

1. Any address within the bank can equally be used.

Figure 28. Protection Register Program flowchart and pseudo code

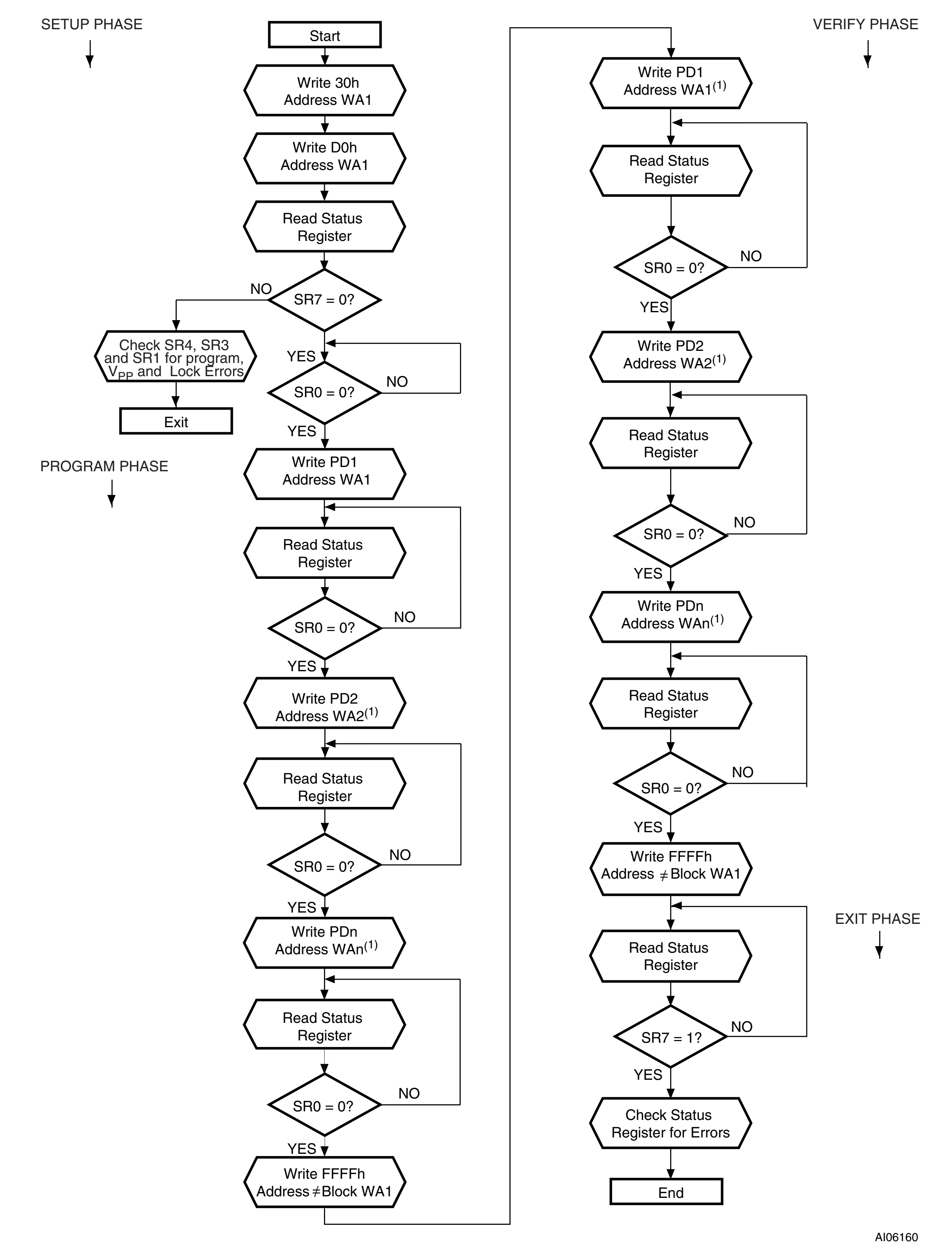



```
protection_register_program_command (addressToProgram, dataToProgram) {:
        writeToFlash (addressToProgram, 0xC0) ;
                /*see note (3) */

        writeToFlash (addressToProgram, dataToProgram) ;
        /*Memory enters read status state after
        the Program Command*/

    do {
        status_register=readFlash (addressToProgram) ;
                /* see note (3) */
        /* E or G must be toggled*/

    } while (status_register.SR7== 0) ;

    if (status_register.SR3==1) /*VPP invalid error */
        error_handler ( ) ;

    if (status_register.SR4==1) /*program error */
        error_handler ( ) ;

    if (status_register.SR1==1) /*program to protect block error */
        error_handler ( ) ;

}
```

AI06177b

1. Status check of SR1 (Protected Block), SR3 (V_{PP} Invalid) and SR4 (Program Error) can be made after each program operation or after a sequence.
2. If an error is found, the Status Register must be cleared before further Program/Erase Controller operations.
3. Any address within the bank can equally be used.

Figure 29. Enhanced Factory Program flowchart

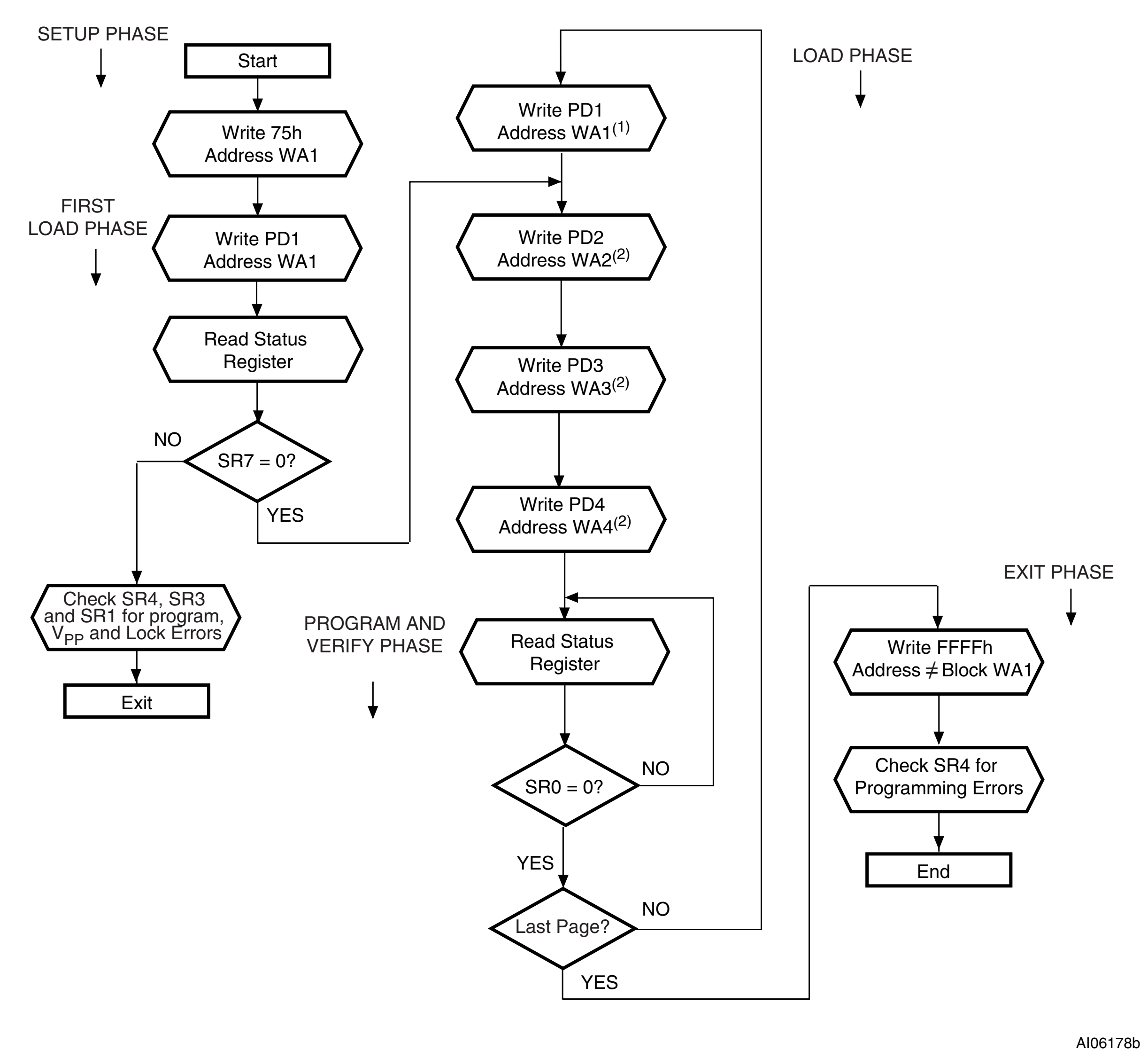


1. Address can remain Starting Address WA1 or be incremented.

16.1 Enhanced factory program pseudo code

```
efp_command(addressFlow,dataFlow,n)
/* n is the number of data to be programmed */
{
   /* setup phase */
   writeToFlash(addressFlow[0],0x30);
   writeToFlash(addressFlow[0],0xD0);
   status_register=readFlash(any_address);
   if (status_register.SR7==1){
      /*EFP aborted for an error*/
      if (status_register.SR4==1) /*program error*/
         error_handler();
      if (status_register.SR3==1) /*VPP invalid error*/
         error_handler();
      if (status_register.SR1==1) /*program to protect block error*/
         error_handler();
   }
   else{
      /*Program Phase*/
      do{
         status_register=readFlash(any_address);
         /* E̅ or G̅ must be toggled*/
      } while (status_register.SR0==1)
      /*Ready for first data*/
      for (i=0; i++; i< n){
         writeToFlash(addressFlow[i],dataFlow[i]);
         /* status register polling*/
         do{
            status_register=readFlash(any_address);
            /* E̅ or G̅ must be toggled*/
         } while (status_register.SR0==1);
         /* Ready for a new data */
      }
      writeToFlash(another_block_address,FFFFh);

      /* Verify Phase */
      for (i=0; i++; i< n){
         writeToFlash(addressFlow[i],dataFlow[i]);
         /* status register polling*/
         do{
            status_register=readFlash(any_address);
            /* E̅ or G̅ must be toggled*/
         } while (status_register.SR0==1);
         /* Ready for a new data */
      }
      writeToFlash(another_block_address,FFFFh);
      /* exit program phase */

      /* Exit Phase */
      /* status register polling */
      do{
         status_register=readFlash(any_address);
         /* E̅ or G̅ must be toggled */
      } while (status_register.SR7==0);
      if (status_register.SR4==1) /*program failure error*/
         error_handler();
      if (status_register.SR3==1) /*VPP invalid error*/
         error_handler();
      if (status_register.SR1==1) /*program to protect block error*/
         error_handler();
   }
}
```

Figure 30. Quadruple Enhanced Factory Program flowchart



1. Address can remain Starting Address WA1 (in which case the next Page is programmed) or can be any address in the same block.
2. The address is only checked for the first Word of each Page as the order to program the Words is fixed, so subsequent Words in each Page can be written to any address.

16.2 Quadruple Enhanced Factory Program pseudo code

```
quad_efp_command(addressFlow,dataFlow,n)
/* n is the number of pages to be programmed.*/
            {
      /* Setup phase */
   writeToFlash(addressFlow[0],0x75);
      for (i=0; i++; i< n){
                              /*Data Load Phase*/

      /*First Data*/
      writeToFlash(addressFlow[i],dataFlow[i,0]);
            /*at the first data of the first page, Quad-EFP may be aborted*/
      if (First_Page) {
                  status_register=readFlash(any_address);
                  if (status_register.SR7==1){
            /*EFP aborted for an error*/
                        if (status_register.SR4==1) /*program error*/
               error_handler();
                        if (status_register.SR3==1) /*VPP invalid error*/
               error_handler();
                        if (status_register.SR1==1) /*program to protect block
error*/
               error_handler();
         }
      }
      /*2nd data*/
            writeToFlash(addressFlow[i],dataFlow[i,1]);

      /*3rd data*/
                  writeToFlash(addressFlow[i],dataFlow[i,2]);

      /*4th data*/
            writeToFlash(addressFlow[i],dataFlow[i,3]);

      /* Program&Verify Phase */
      do{
         status_register=readFlash(any_address);
         /* E̅ or G̅ must be toggled*/
      }while (status_register.SR0==1)
   }
   /* Exit Phase */
   writeToFlash(another_block_address,FFFFh);
      /* status register polling */
   do{
      status_register=readFlash(any_address);
      /* E̅ or G̅ must be toggled */
   } while (status_register.SR7==0);
         if (status_register.SR1==1) /*program to protected block error*/
      error_handler();
      if (status_register.SR3==1) /*VPP invalid error*/
      error_handler();
      if (status_register.SR4==1) /*program failure error*/
      error_handler();
   }
}
```

Appendix D Command interface state tables

Table 42. Command interface states - modify table, next state

<table>
<tr><th colspan="2" rowspan="2">Current CI State</th><th colspan="11">Command Input[1]</th></tr>
<tr><th>Read Array[2] (FFh)</th><th>WP setup[3][4] (10/40h)</th><th>DWP, QWP Setup[3][4] (35h, 56h)</th><th>Block Erase, Bank Erase Setup[3][4] (20h, 80h</th><th>EFP Setup (30h)</th><th>Quad-EFP Setup (75h)</th><th>Erase Confirm P/E Resume, Block Unlock confirm, EFP Confirm (D0h)</th><th>Program/ Erase Suspend (B0h)</th><th>Read Status Register (70h)</th><th>Clear status Register[5] (50h)</th><th>Read Electronic signature, Read CFI Query (90h, 98h)</th></tr>
<tr><td colspan="2">Ready</td><td>Ready</td><td>Program Setup</td><td>Program Setup</td><td>Erase Setup</td><td>EFP Setup</td><td>Quad-EFP Setup</td><td colspan="5">Ready</td></tr>
<tr><td colspan="2">Lock/CR Setup</td><td colspan="6">Ready (Lock Error)</td><td>Ready</td><td colspan="4">Ready (Lock Error)</td></tr>
<tr><td rowspan="2">OTP</td><td>Setup</td><td colspan="11" rowspan="2">OTP Busy</td></tr>
<tr><td>Busy</td></tr>
<tr><td rowspan="3">Program</td><td>Setup</td><td colspan="11">Program Busy</td></tr>
<tr><td>Busy</td><td colspan="7">Program Busy</td><td>Program Suspended</td><td colspan="3">Program Busy</td></tr>
<tr><td>Suspend</td><td colspan="6">Program Suspended</td><td>Program Busy</td><td colspan="4">Program Suspended</td></tr>
<tr><td rowspan="3">Erase</td><td>Setup</td><td colspan="6">Ready (error)</td><td>Erase Busy</td><td colspan="4">Ready (error)</td></tr>
<tr><td>Busy</td><td colspan="7">Erase Busy</td><td>Erase Suspended</td><td colspan="3">Erase Busy</td></tr>
<tr><td>Suspend</td><td>Erase Suspended</td><td>Program in Erase Suspend</td><td colspan="4">Erase Suspended</td><td>Erase Busy</td><td colspan="4">Erase Suspended</td></tr>
<tr><td rowspan="3">Program in Erase Suspend</td><td>Setup</td><td colspan="11">Program Busy in Erase Suspend</td></tr>
<tr><td>Busy</td><td colspan="7">Program Busy in Erase Suspend</td><td>Program Suspend in Erase Suspend</td><td colspan="3">Program Busy in Erase Suspend</td></tr>
<tr><td>Suspend</td><td colspan="6">Program Suspend in Erase Suspend</td><td>Program Busy in Erase Suspend</td><td colspan="4">Program Suspend in Erase Suspend</td></tr>
<tr><td colspan="2">Lock/CR Setup in Erase Suspend</td><td colspan="6">Erase Suspend (Lock Error)</td><td>Erase Suspend</td><td colspan="4">Erase Suspend (Lock Error)</td></tr>
<tr><td rowspan="3">EFP</td><td>Setup</td><td colspan="6">Ready (error)</td><td>EFP Busy</td><td colspan="4">Ready (error)</td></tr>
<tr><td>Busy</td><td colspan="11">EFP Busy[6]</td></tr>
<tr><td>Verify</td><td colspan="11">EFP Verify[6]</td></tr>
<tr><td rowspan="2">Quad EFP</td><td>Setup</td><td colspan="11">Quad EFP Busy[6]</td></tr>
<tr><td>Busy</td><td colspan="11">Quad EFP Busy[6]</td></tr>
</table>

1. CI = Command Interface, CR = Configuration Register, EFP = Enhanced Factory Program, Quad EFP = Quadruple Enhanced Factory Program, DWP = Double Word Program, QWP = Quadruple Word Program, P/E. C. = Program/Erase Controller.
2. At Power-Up, all banks are in Read Array mode. A Read Array command issued to a busy bank, results in undetermined data output.
3. The two cycle command should be issued to the same bank address.
4. If the P/E.C. is active, both cycles are ignored.
5. The Clear Status Register command clears the Status Register error bits except when the P/E.C. is busy or suspended.
6. EFP and Quad EFP are allowed only when Status Register bit SR0 is set to '0'.EFP and Quad EFP are busy if Block Address is first EFP Address. Any other commands are treated as data.

Table 43. Command interface states - modify table, next output[1]

<table>
<tr><th rowspan="2">Current CI State</th><th colspan="11">Command Input [2]</th></tr>
<tr><th>Read Array[3] (FFh)</th><th>WP Setup[4][5] (10/40h)</th><th>DWP, QWP Setup[4][5] (35h, 56h)</th><th>Block Erase, Bank Erase Setup[4][5] (20h, 80h)</th><th>EFP Setup (30h)</th><th>Quad-EFP Setup (75h)</th><th>Erase Confirm P/E Resume, Block Unlock confirm, EFP Confirm (D0h)</th><th>Program/ Erase Suspend (B0h)</th><th>Read Status Register (70h)</th><th>Clear status Register[6] (50h)</th><th>Read Electronic signature, Read CFI Query (90h, 98h)</th></tr>
<tr><td>Program Setup</td><td colspan="11" rowspan="11">Status Register</td></tr>
<tr><td>Erase Setup</td></tr>
<tr><td>OTP Setup</td></tr>
<tr><td>Program in Erase Suspend</td></tr>
<tr><td>EFP Setup</td></tr>
<tr><td>EFP Busy</td></tr>
<tr><td>EFP Verify</td></tr>
<tr><td>Quad EFP Setup</td></tr>
<tr><td>Quad EFP Busy</td></tr>
<tr><td>Lock/CR Setup</td></tr>
<tr><td>Lock/CR Setup in Erase Suspend</td></tr>
<tr><td>OTP Busy</td><td rowspan="7">Array</td><td colspan="5" rowspan="7">Status Register</td><td colspan="2" rowspan="7">Output Unchanged</td><td rowspan="7">Status Register</td><td rowspan="7">Output Unchanged</td><td>Status Register</td></tr>
<tr><td>Ready</td><td rowspan="6">Electronic Signature/CFI</td></tr>
<tr><td>Program Busy</td></tr>
<tr><td>Erase Busy</td></tr>
<tr><td>Program/Erase</td></tr>
<tr><td>Program Busy in Erase Suspend</td></tr>
<tr><td>Program Suspend in Erase Suspend</td></tr>
</table>

1. CI = Command Interface, CR = Configuration Register, EFP = Enhanced Factory Program, Quad EFP = Quadruple Enhanced Factory Program, DWP = Double Word Program, QWP = Quadruple Word Program, P/E. C. = Program/Erase Controller.
2. The output state shows the type of data that appears at the outputs if the bank address is the same as the command address. A bank can be placed in Read Array, Read Status Register, Read Electronic Signature or Read CFI Query mode, depending on the command issued. Each bank remains in its last output state until a new command is issued. The next state does not depend on the bank's output state.
3. At Power-Up, all banks are in Read Array mode. A Read Array command issued to a busy bank, results in undetermined data output.
4. The two cycle command should be issued to the same bank address.
5. If the P/E.C. is active, both cycles are ignored.
6. The Clear Status Register command clears the Status Register error bits except when the P/E.C. is busy or suspended.

Table 44. Command interface states - lock table, next state[1]

<table>
<tr><th colspan="2" rowspan="2">Current CI State</th><th colspan="8">Command Input</th></tr>
<tr><th>Lock/CR Setup[2] (60h)</th><th>OTP Setup[2] (C0h)</th><th>Block Lock Confirm (01h)</th><th>Block Lock-Down Confirm (2Fh)</th><th>Set CR Confirm (03h)</th><th>EFP Exit, Quad EFP Exit[3]</th><th>Illegal Command[4]</th><th>P/E. C. Operation Completed</th></tr>
<tr><td colspan="2">Ready</td><td>Lock/CR Setup</td><td>OTP Setup</td><td colspan="5">Ready</td><td>N/A</td></tr>
<tr><td colspan="2">Lock/CR Setup</td><td colspan="2">Ready (Lock error)</td><td colspan="3">Ready</td><td colspan="2">Ready (Lock error)</td><td>N/A</td></tr>
<tr><td rowspan="2">OTP</td><td>Setup</td><td colspan="7" rowspan="2">OTP Busy</td><td>N/A</td></tr>
<tr><td>Busy</td><td>Ready</td></tr>
<tr><td rowspan="3">Program</td><td>Setup</td><td colspan="7">Program Busy</td><td>N/A</td></tr>
<tr><td>Busy</td><td colspan="7">Program Busy</td><td>Ready</td></tr>
<tr><td>Suspend</td><td colspan="7">Program Suspended</td><td>N/A</td></tr>
<tr><td rowspan="3">Erase</td><td>Setup</td><td colspan="7">Ready (error)</td><td>N/A</td></tr>
<tr><td>Busy</td><td colspan="7">Erase Busy</td><td>Ready</td></tr>
<tr><td>Suspend</td><td>Lock/CR Setup in Erase Suspend</td><td colspan="6">Erase Suspended</td><td>N/A</td></tr>
<tr><td rowspan="3">Program in Erase Suspend</td><td>Setup</td><td colspan="7">Program Busy in Erase Suspend</td><td>N/A</td></tr>
<tr><td>Busy</td><td colspan="7">Program Busy in Erase Suspend</td><td>Erase Suspended</td></tr>
<tr><td>Suspend</td><td colspan="7">Program Suspend in Erase Suspend</td><td>N/A</td></tr>
<tr><td colspan="2">Lock/CR Setup in Erase Suspend</td><td colspan="2">Erase Suspend (Lock error)</td><td colspan="3">Erase Suspend</td><td colspan="2">Erase Suspend (Lock error)</td><td>N/A</td></tr>
<tr><td rowspan="3">EFP</td><td>Setup</td><td colspan="7">Ready (error)</td><td>N/A</td></tr>
<tr><td>Busy</td><td colspan="5">EFP Busy[5]</td><td>EFP Verify</td><td>EFP Busy[5]</td><td>N/A</td></tr>
<tr><td>Verify</td><td colspan="5">EFP Verify[5]</td><td>Ready</td><td>EFP Verify[5]</td><td>Ready</td></tr>
<tr><td rowspan="2">QuadEFP</td><td>Setup</td><td colspan="7">Quad EFP Busy[5]</td><td>N/A</td></tr>
<tr><td>Busy</td><td colspan="5">Quad EFP Busy[5]</td><td>Ready</td><td>Quad EFP Busy[5]</td><td>Ready</td></tr>
</table>

1. CI = Command Interface, CR = Configuration Register, EFP = Enhanced Factory Program, Quad EFP = Quadruple Enhanced Factory Program, P/E. C. = Program/Erase Controller.
2. If the P/E.C. is active, both cycles are ignored.
3. EFP and Quad EFP exit when Block Address is different from first Block Address and data is FFFFh.
4. Illegal commands are those not defined in the command set.
5. EFP and Quad EFP are allowed only when Status Register bit SR0 is set to ‘0’. EFP and Quad EFP are busy if Block Address is first EFP Address. Any other commands are treated as data.

Table 45. Command interface states - lock table, next output[1]

<table>
<tr><th rowspan="2">Current CI State</th><th colspan="8">Command Input</th></tr>
<tr><th>Lock/CR Setup[2] (60h)</th><th>OTP Setup[2] (C0h)</th><th>Block Lock Confirm (01h)</th><th>Block Lock-Down Confirm (2Fh)</th><th>Set CR Confirm (03h)</th><th>EFP Exit, Quad EFP Exit[3]</th><th>Illegal Command[4]</th><th>P/E. C. Operation Completed</th></tr>
<tr><td>Program Setup</td><td colspan="7" rowspan="9">Status Register</td><td rowspan="18">Output Unchanged</td></tr>
<tr><td>Erase Setup</td></tr>
<tr><td>OTP Setup</td></tr>
<tr><td>Program in Erase Suspend</td></tr>
<tr><td>EFP Setup</td></tr>
<tr><td>EFP Busy</td></tr>
<tr><td>EFP Verify</td></tr>
<tr><td>Quad EFP Setup</td></tr>
<tr><td>Quad EFP Busy</td></tr>
<tr><td>Lock/CR Setup</td><td colspan="4" rowspan="2">Status Register</td><td rowspan="2">Array</td><td colspan="2" rowspan="2">Status Register</td></tr>
<tr><td>Lock/CR Setup in Erase Suspend</td></tr>
<tr><td>OTP Busy</td><td colspan="2" rowspan="7">Status Register</td><td colspan="3" rowspan="7">Output Unchanged</td><td rowspan="7">Array</td><td rowspan="7">Output Unchanged</td></tr>
<tr><td>Ready</td></tr>
<tr><td>Program Busy</td></tr>
<tr><td>EraseBusy</td></tr>
<tr><td>Program/Erase</td></tr>
<tr><td>Program Busy in Erase Suspend</td></tr>
<tr><td>Program Suspend in Erase Suspend</td></tr>
</table>

1. CI = Command Interface, CR = Configuration Register, EFP = Enhanced Factory Program, Quad EFP = Quadruple Enhanced Factory Program, P/E. C. = Program/Erase Controller.
2. If the P/E.C. is active, both cycles are ignored.
3. EFP and Quad EFP exit when Block Address is different from first Block Address and data is FFFFh.
4. Illegal commands are those not defined in the command set.

17 Revision history

Table 46. Document revision history

Date	Revision	Changes
15-Oct-2004	0.1	First Issue.
31-May-2005	0.2	*Table 14: Dual operation limitations* added. t_{WHQV} removed throughout document. 80ns speed class removed. X-latency setting clarified. Bank Erase moved to Factory Program command sections. Test conditions modified in *DC and AC parameters*.
05-July-2005	0.3	Programming command clarified in *Command interface - Factory program commands*. *Table 9: X-Latency Settings*, *Table 23: Synchronous read AC characteristics* and Table 31., M58WR128HU - Parameter Bank Block Addresses corrected.
15-Nov-2005	1	M58WR128HU and M58WR128HL removed, M58WR032HU and M58WR32HL added. VFBGA44 7.5 x 5mm package added. *Appendix A: Block address tables* revised. All packages are ECOPACK®. Document promoted from Target specification to full datasheet status.
18-Nov-2005	2	Missing Tables (*Table 3*, *Table 32* and *Table 33*) and *Figure 4* added back.
18-Apr-2006	3	M58WR032HU and M58WR32HL part numbers removed. Small text changes. Removed VFBGA44, 7.5 x 5mm package.
04-May-2006	4	VFBGA44 7.5 × 5 package added back (see *Section 15: Package mechanical*). Small text changes. Daisy chain ordering scheme table removed.
12-Nov-2007	5	Applied Numonyx branding.