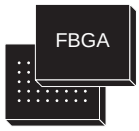
# M58CR064C, M58CR064D
# M58CR064P, M58CR064Q

## 64 Mbit (4Mb x 16, Dual Bank, Burst )
## 1.8V Supply Flash Memory

**FEATURES SUMMARY**

- SUPPLY VOLTAGE
  - $V_{DD}$ = 1.65V to 2V for Program, Erase and Read
  - $V_{DDQ}$ = 1.65V to 3.3V for I/O Buffers
  - $V_{PP}$ = 12V for fast Program (optional)
- SYNCHRONOUS / ASYNCHRONOUS READ
  - Synchronous Burst Read mode : 54MHz
  - Asynchronous/ Synchronous Page Read mode
  - Random Access: 85, 100, 120ns
- PROGRAMMING TIME
  - 10µs by Word typical
  - Double/Quadruple Word Program option
- MEMORY BLOCKS
  - Dual Bank Memory Array: 16/48 Mbit
  - Parameter Blocks (Top or Bottom location)
- DUAL OPERATIONS
  - Program Erase in one Bank while Read in other
  - 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
  - One parameter block permanently lockable
- COMMON FLASH INTERFACE (CFI)
- 100,000 PROGRAM/ERASE CYCLES per BLOCK

**Figure 1. Packages**




TFBGA56 (ZB)
6.5 x 10mm

- ELECTRONIC SIGNATURE
  - Manufacturer Code: 20h
  - Top Device Code, M58CR064C: 88CAh
  - Bottom Device Code, M58CR064D: 88CBh
  - Top Device Code, M58CR064P: 8801h
  - Bottom Device Code, M58CR064Q: 8802h

## TABLE OF CONTENTS

**SUMMARY DESCRIPTION . . . 6**
Figure 2. Logic Diagram . . . 7
Table 1. Signal Names . . . 7
Figure 3. TFBGA Connections (Top view through package) . . . 8
Table 2. Bank Architecture . . . 8
Figure 4. Memory Map . . . 9

**SIGNAL DESCRIPTIONS . . . 10**
Address Inputs (A0-A21). . . . 10
Data Input/Output (DQ0-DQ15). . . . 10
Chip Enable (E). . . . 10
Output Enable (G). . . . 10
Write Enable (W). . . . 10
Write Protect (WP). . . . 10
Reset/Power-Down (RP). . . . 10
Latch Enable (L). . . . 10
Clock (K). . . . 10
Wait (WAIT). . . . 10
$V_{DD}$ Supply Voltage . . . 10
$V_{DDQ}$ Supply Voltage. . . . 10
$V_{PP}$ Program Supply Voltage . . . 10
$V_{SS}$ Ground. . . . 10
$V_{SSQ}$ Ground. . . . 10

**BUS OPERATIONS. . . . 11**
Bus Read. . . . 11
Bus Write. . . . 11
Address Latch. . . . 11
Output Disable. . . . 11
Standby. . . . 11
Reset. . . . 11
Table 3. Bus Operations . . . 11

**COMMAND INTERFACE . . . 12**
Table 4. Command Codes. . . . 12

**COMMAND INTERFACE - STANDARD COMMANDS . . . 13**
Read Array Command . . . 13
Read Status Register Command . . . 13
Read Electronic Signature Command . . . 13
Read CFI Query Command . . . 13
Clear Status Register Command . . . 13
Block Erase Command . . . 13
Bank Erase Command . . . 14
Program Command . . . 14

Program/Erase Suspend Command . . . 14
Program/Erase Resume Command . . . 15
Protection Register Program Command . . . 15
Set Configuration Register Command. . . . 15
Block Lock Command . . . 15
Block Unlock Command . . . 16
Block Lock-Down Command . . . 16
Table 5. Standard Commands. . . . 17
Table 6. Electronic Signature Codes . . . 18
Figure 5. Security Block and Protection Register Memory Map . . . 18

**COMMAND INTERFACE - FACTORY PROGRAM COMMANDS . . . 19**
Double Word Program Command . . . 19
Quadruple Word Program Command . . . 19
Table 7. Factory Program Commands . . . 20

**STATUS REGISTER . . . 20**
Program/Erase Controller Status Bit (SR7) . . . 20
Erase Suspend Status Bit (SR6) . . . 20
Erase Status Bit (SR5) . . . 20
Program Status Bit (SR4) . . . 21
$V_{PP}$ Status Bit (SR3) . . . 21
Program Suspend Status Bit (SR2) . . . 21
Block Protection Status Bit (SR1) . . . 21
Reserved Bit (SR0) . . . 21
Table 8. Status Register Bits . . . 22

**CONFIGURATION REGISTER . . . 23**
Read Select Bit (CR15) . . . 23
X-Latency Bits (CR13-CR11) . . . 23
Power-Down Bit (CR10) . . . 23
Wait Configuration Bit (CR8) . . . 23
Burst Type Bit (CR7) . . . 23
Valid Clock Edge Bit (CR6) . . . 23
Wrap Burst Bit (CR3). . . . 23
Burst length Bits (CR2-CR0) . . . 24
Table 9. Configuration Register . . . 24
Table 10. Burst Type Definition . . . 25
Figure 6. X-Latency and Data Output Configuration Example . . . 26
Figure 7. Wait Configuration Example . . . 26

**READ MODES. . . . 27**
Asynchronous Read Mode . . . 27
Synchronous Burst Read Mode. . . . 27
Single Synchronous Read Mode . . . 27

**DUAL OPERATIONS AND MULTIPLE BANK ARCHITECTURE . . . 28**

Table 11. Dual Operations Allowed In Other Bank . . . . . 28
Table 12. Dual Operations Allowed In Same Bank . . . . . 28

**BLOCK LOCKING . . . . . 29**
Reading a Block's Lock Status . . . . . 29
Locked State . . . . . 29
Unlocked State . . . . . 29
Lock-Down State . . . . . 29
Locking Operations During Erase Suspend . . . . . 29
Table 13. Lock Status . . . . . 30

**PROGRAM AND ERASE TIMES AND ENDURANCE CYCLES . . . . . 31**
Table 14. Program, Erase Times and Program, Erase Endurance Cycles . . . . . 31

**MAXIMUM RATING . . . . . 32**
Table 15. Absolute Maximum Ratings . . . . . 32

**DC and AC PARAMETERS . . . . . 33**
Table 16. Operating and AC Measurement Conditions . . . . . 33
Figure 8. AC Measurement I/O Waveform . . . . . 33
Figure 9. AC Measurement Load Circuit . . . . . 33
Table 17. Capacitance . . . . . 33
Table 18. DC Characteristics - Currents . . . . . 34
Table 19. DC Characteristics - Voltages . . . . . 35
Figure 10. Asynchronous Random Access Read AC Waveforms . . . . . 36
Figure 11. Asynchronous Page Read AC Waveforms . . . . . 37
Table 20. Asynchronous Read AC Characteristics . . . . . 38
Figure 12. Synchronous Burst Read AC Waveforms . . . . . 39
Figure 13. Single Synchronous Read AC Waveforms . . . . . 40
Table 21. Synchronous Read AC Characteristics . . . . . 41
Figure 14. Write AC Waveforms, Write Enable Controlled . . . . . 42
Table 22. Write AC Characteristics, Write Enable Controlled . . . . . 43
Figure 15. Write AC Waveforms, Chip Enable Controlled . . . . . 44
Table 23. Write AC Characteristics, Chip Enable Controlled . . . . . 45
Figure 16. Reset and Power-up AC Waveforms . . . . . 46
Table 24. Reset and Power-up AC Characteristics . . . . . 46

**PACKAGE MECHANICAL . . . . . 47**
Figure 17. TFBGA56 6.5x10mm - 8x7 ball array, 0.75 mm pitch, Bottom View Package Outline . . 47
Table 25. TFBGA56 6.5x10mm - 8x7 ball array, 0.75 mm pitch, Package Mechanical Data . . . . . 47

**PART NUMBERING . . . . . 48**
Table 26. Ordering Information Scheme . . . . . 48

**REVISION HISTORY . . . . . 49**
Table 27. Document Revision History . . . . . 49

**APPENDIX A. BLOCK ADDRESS TABLES . . . . . 50**
Table 28. Top Boot Block Addresses, M58CR064C, M58CR064P . . . . . 50
Table 29. Bottom Boot Block Addresses, M58CR064D, M58CR064Q . . . . . 52

**APPENDIX B. COMMON FLASH INTERFACE . . . . . 54**
Table 30. Query Structure Overview . . . . . 54
Table 31. CFI Query Identification String . . . . . 54
Table 32. CFI Query System Interface Information . . . . . 55
Table 33. Device Geometry Definition . . . . . 55
Table 34. Primary Algorithm-Specific Extended Query Table . . . . . 56
Table 35. Burst Read Information . . . . . 57

**APPENDIX C. FLOWCHARTS AND PSEUDO CODES. . . . . 58**
Figure 18. Program Flowchart and Pseudo Code . . . . . 58
Figure 19. Double Word Program Flowchart and Pseudo code . . . . . 59
Figure 20. Quadruple Word Program Flowchart and Pseudo Code . . . . . 60
Figure 21. Program Suspend & Resume Flowchart and Pseudo Code . . . . . 61
Figure 22. Block Erase Flowchart and Pseudo Code . . . . . 62
Figure 23. Erase Suspend & Resume Flowchart and Pseudo Code. . . . . 63
Figure 24. Locking Operations Flowchart and Pseudo Code . . . . . 64
Figure 25. Protection Register Program Flowchart and Pseudo Code . . . . . 65

**APPENDIX D. COMMAND INTERFACE STATE TABLES . . . . . 66**
Table 36. Command Interface States - Lock table . . . . . 66
Table 37. Command Interface States - Modify Table . . . . . 67

## SUMMARY DESCRIPTION

The M58CR064 is a 64 Mbit (4Mbit x16) non-volatile Flash memory that may be erased electrically at block level and programmed in-system on a Word-by-Word basis using a 1.65V to 2V $V_{DD}$ supply for the circuitry and a 1.65V to 3.3V $V_{DDQ}$ supply for the Input/Output pins. An optional 12V $V_{PP}$ power supply is provided to speed up customer programming. In M58CR064C and M58CR064D the $V_{PP}$ pin can also be used as a control pin to provide absolute protection against program or erase. In M58CR064P and M58CR064Q this feature is disabled.

The device features an asymmetrical block architecture. M58CR064 has an array of 135 blocks, and is divided into two banks, Banks A and B. The Dual Bank Architecture allows Dual Operations, while programming or erasing in one bank, Read operations are possible in the other bank. 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 4. The Parameter Blocks are located at the top of the memory address space for the M58CR064C and M58CR064P, and at the bottom for the M58CR064D and M58CR064Q.

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 $V_{DD}$.

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 54MHz.

The M58CR064 features 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. In M58CR064C and M58CR064D 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 and a Security Block 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 ST, and a 128 bit segment One-Time-Programmable (OTP) by the user. The user programmable segment can be permanently protected. The Security Block, parameter block 0, can be permanently protected by the user. Figure 5, shows the Security Block and Protection Register Memory Map.

The memory is offered in a TFBGA56, 6.5 x 10mm, 0.75 mm ball pitch package and is supplied with all the bits erased (set to '1').

**Figure 2. Logic Diagram**

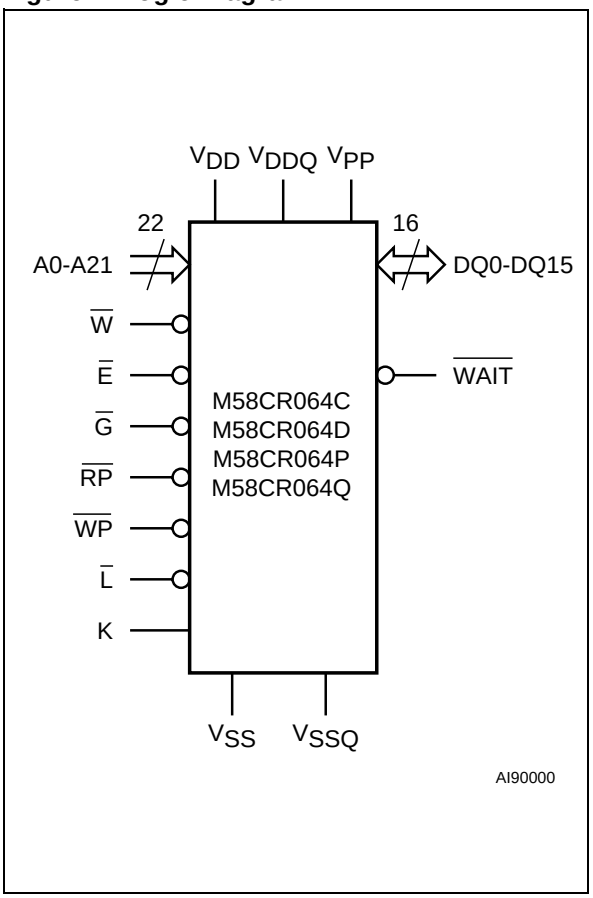



**Table 1. Signal Names**

| | |
|---|---|
| A0-A21 | Address Inputs |
| DQ0-DQ15 | Data Input/Outputs, 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 |
| $\overline{WAIT}$ | Wait |
| $V_{DD}$ | 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 3. TFBGA Connections (Top view through package)**

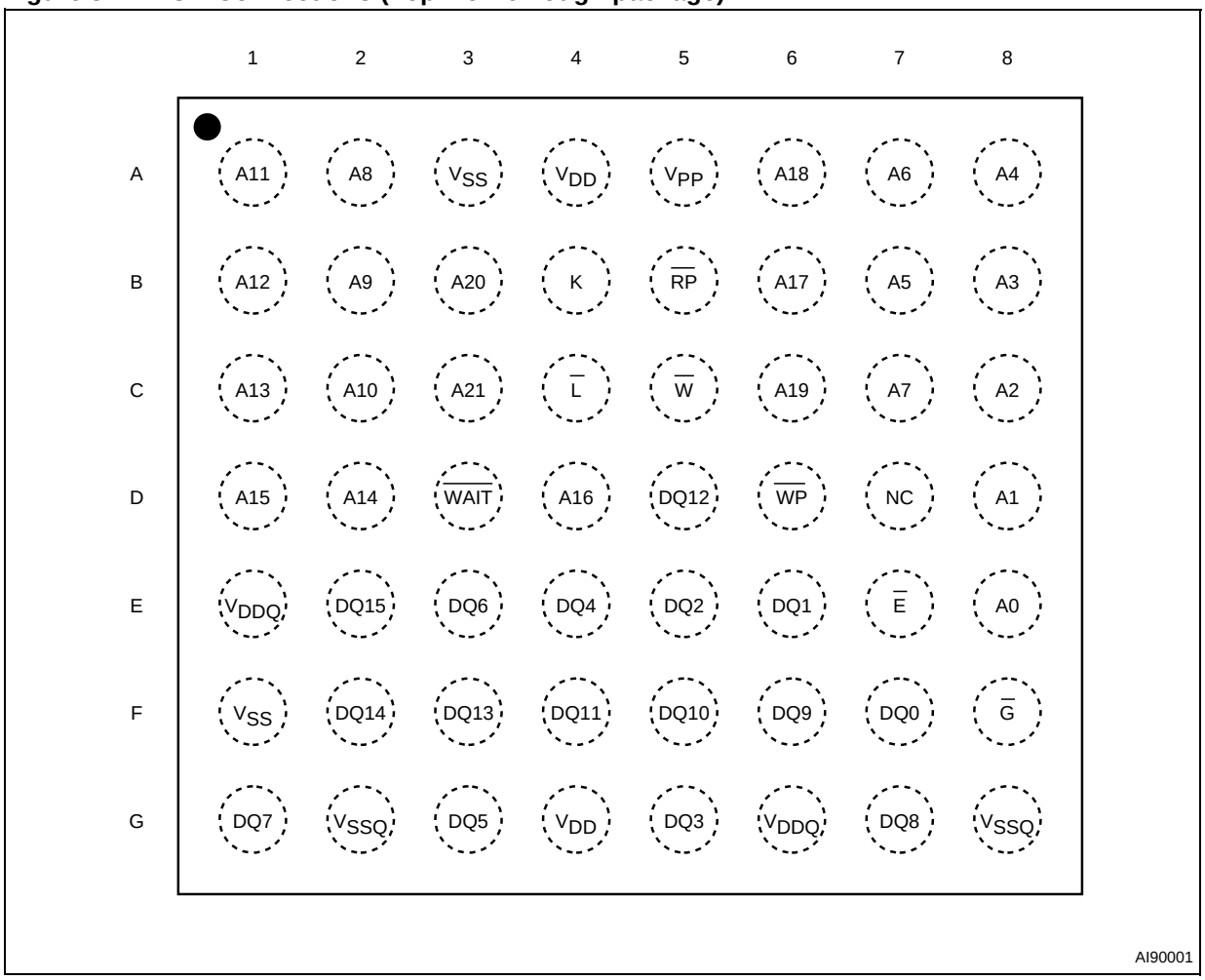
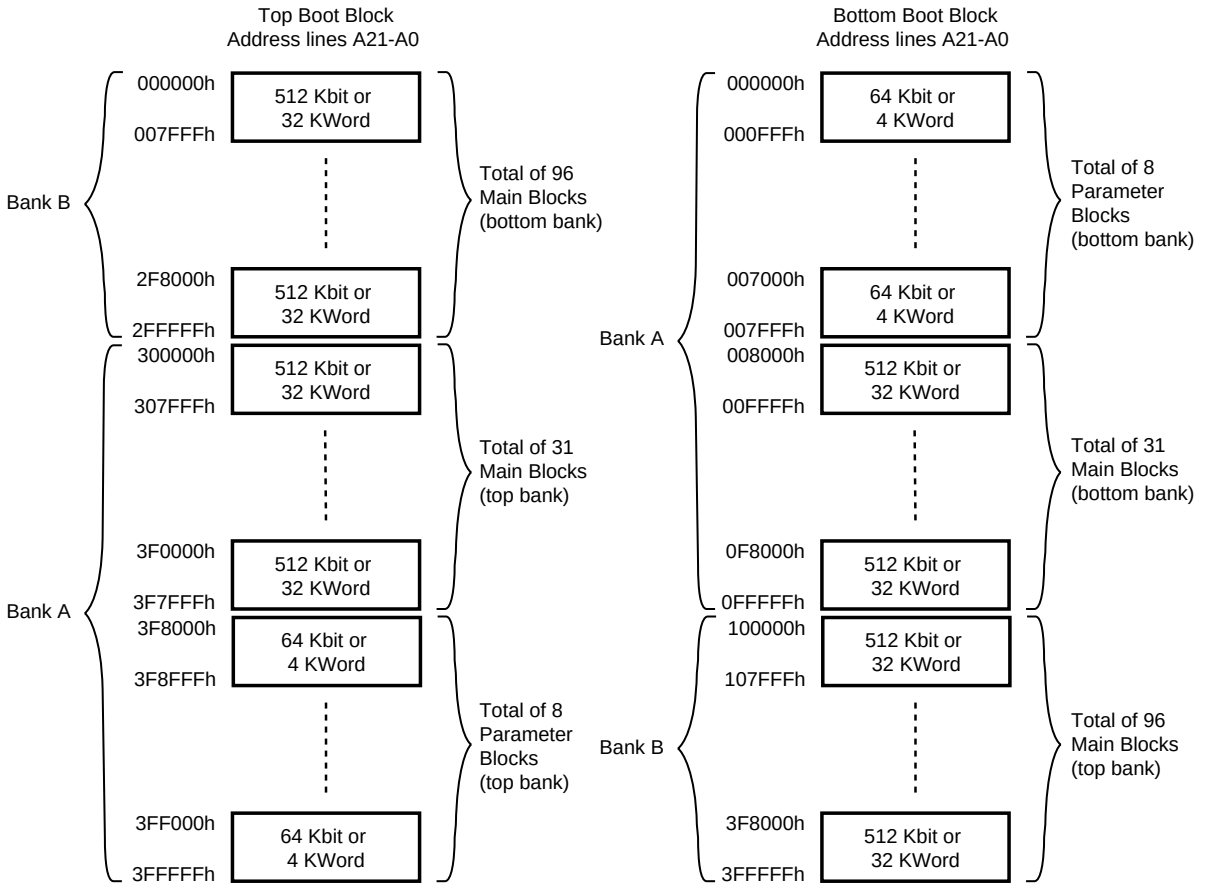



**Table 2. Bank Architecture**

| | Bank Size | Parameter Blocks | Main Blocks |
|---|---|---|---|
| Bank A | 16 Mbit | 8 blocks of 4 KWord | 31 blocks of 32 KWord |
| Bank B | 48 Mbit | - | 96 blocks of 32 KWord |

**Figure 4. Memory Map**



## SIGNAL DESCRIPTIONS

See Figure 2 Logic Diagram and Table 1,Signal Names, for a brief overview of the signals connected to this device.

**Address Inputs (A0-A21).** The Address Inputs select the cells in the memory array to access during Bus Read operations. During Bus Write operations they control the commands sent to the Command Interface of the internal state machine.

**Data Input/Output (DQ0-DQ15).** 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.

**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/Power-Down 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.

**Output Enable ($\overline{G}$).** The Output Enable controls the outputs during the Bus Read operation of the memory.

**Write Enable ($\overline{W}$).** The Write Enable controls the Bus Write operation of the memory's Command Interface. The data and address inputs are latched on the rising edge of Chip Enable or Write Enable whichever occurs first.

**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 13, Lock Status).

**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 Configuration Register status. When Reset/Power-Down is at $V_{IL}$, the memory is in reset mode: the outputs are high impedance and if the Power-Down function is enabled the current consumption is reduced to the Reset Supply Current $I_{DD2}$. Refer to Table 2, DC Characteristics - Currents for the value of $I_{DD2}$. After Reset all blocks are in the Locked state and the Configuration Register is reset. 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/Power-Down pin can be interfaced with 3V logic without any additional circuitry. It can be tied to $V_{RPH}$ (refer to Table 19, DC Characteristics).

**Latch Enable ($\overline{L}$).** Latch Enable latches the 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 $V_{IH}$. Latch Enable can be kept Low (also at board level) when the Latch Enable function is not required or supported.

**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.

**Wait ($\overline{WAIT}$).** 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 or Output Enable are at $V_{IH}$ or Reset/Power-Down is at $V_{IL}$. It can be configured to be active during the wait cycle or one clock cycle in advance.

**$V_{DD}$ 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).

**$V_{DDQ}$ 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.

**$V_{PP}$ Program Supply Voltage.** $V_{PP}$ is both a control input and a power supply pin. In M58CR064C/D the two functions are selected by the voltage range applied to the pin. In the M58CR064P/Q the control feature is disabled.

In M58CR064C/D 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 $V_{PP} > V_{PP1}$ enables these functions (see Tables 18 and 19, 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.

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

**$V_{SSQ}$ 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 9, AC Measurement Load Circuit. The PCB trace widths should be sufficient to carry the required $V_{PP}$ program and erase currents.**

## 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.

**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 Command Interface section). Refer to the Read AC Waveform figures and Characteristics tables in the DC and AC Parameters section for details of when the output becomes valid.

**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, Input Data and Addresses are latched on the rising edge of Write Enable or Chip Enable, whichever occurs first. The addresses can also be latched prior to the write operation by toggling Latch Enable. In this case the Latch Enable should be tied to $V_{IH}$ during the bus write operation.

See Figures 14 and 15, Write AC Waveforms, and Tables 22 and 23, Write AC Characteristics, for details of the timing requirements.

**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.

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

**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/Power-Down 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.

**Reset.** 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, independently from the Chip Enable, Output Enable or Write Enable inputs. If Reset 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 | $\overline{E}$ | $\overline{G}$ | $\overline{W}$ | $\overline{L}$ | $\overline{RP}$ | $\overline{WAIT}$ | DQ15-DQ0 |
|---|---|---|---|---|---|---|---|
| Bus Read | $V_{IL}$ | $V_{IL}$ | $V_{IH}$ | $V_{IL}$ (2) | $V_{IH}$ | | Data Output |
| Bus Write | $V_{IL}$ | $V_{IH}$ | $V_{IL}$ | $V_{IL}$ (2) | $V_{IH}$ | Hi-Z | Data Input |
| Address Latch | $V_{IL}$ | X | $V_{IH}$ | $V_{IL}$ | $V_{IH}$ | | Data Output or Hi-Z (3) |
| Output Disable | $V_{IL}$ | $V_{IH}$ | $V_{IH}$ | X | $V_{IH}$ | Hi-Z | Hi-Z |
| Standby | $V_{IH}$ | X | X | X | $V_{IH}$ | Hi-Z | Hi-Z |
| Reset | X | X | X | X | $V_{IL}$ | Hi-Z | Hi-Z |

Note: 1. X = Don't care.
2. $\overline{L}$ can be tied to $V_{IH}$ if the valid address has been previously latched.
3. Depends on $\overline{G}$.

## 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 36 and 37, 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 | Double Word Program Setup |
| 40h | Program Setup |
| 50h | Clear Status Register |
| 55h | Quadruple Word Program Setup |
| 60h | Block Lock Setup, Block Unlock Setup, Block Lock Down Setup and Set Configuration Register Setup |
| 70h | Read Status Register |
| 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 |
| FFh | Read Array |

## 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.

### 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 the other bank. However if a Read Array command is issued to a bank currently executing a Program or Erase operation the command will be ignored.

### Read Status Register Command

A Bank's 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 of the addressed Bank. The Read Status Register command can be issued at any time, even during Program or Erase operations.

The following Bus 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.

### 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 the bottom bank. A subsequent read operation in the address of the bottom bank will output the Manufacturer Code, the Device Code, the protection Status of Blocks of the bottom bank, the Die Revision Code, the Protection Register, or the Read Configuration Register (see Table 6).

If the first write cycle of Read Electronic Signature command is issued to an address within the top bank, a subsequent read operation in an address of the top bank will output the protection Status of blocks of the top bank. The status of the other bank is not affected by the command (see Table 11). This mode supports asynchronous or single synchronous reads only, it does not support page mode or synchronous burst reads.

### Read CFI Query Command

The Read CFI Query command is used to read data from the Common Flash Interface (CFI) memory area located in the bottom bank. The Read CFI Query Command consists of one Bus Write cycle, to an address within the bottom bank. 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 Status Register mode, subsequent Bus Read cycles will output the Status Register and the Program/Erase controller will continue to Program or Erase in the background. When the Program or Erase operation has finished the device will enter Read CFI Query mode.

This mode supports asynchronous or single synchronous reads only, it does not support page mode or synchronous burst reads.

The status of the other banks is not affected by the command (see Table 11). After issuing a Read CFI Query command, a Read Array command should be issued to the addressed bank to return the bank to read mode.

See Appendix C, Common Flash Interface, Tables 30, 31, 32, 33, 34 and 35 for details on the information contained in the Common Flash Interface memory area.

### Clear Status Register Command

The Clear Status Register command can be used to reset (set to '0') error bits 1, 3, 4 and 5 in the Status Register of the addressed bank. One bus write cycle is required to issue the Clear Status Register command. After the Clear Status Register command the bank returns to Read Array 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.

### 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 in the internal state machine and starts the Program/ Erase Controller.

If the second bus cycle is not Write Erase Confirm (D0h), Status Register bits 4 and 5 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 Status Register and the Program/Erase Suspend command, all other commands will be ignored. Refer to Dual Operations section for detailed information about simultaneous operations allowed in banks not being erased. Typical Erase times are given in Table 14, Program, Erase Times and Program/Erase Endurance Cycles.

See Appendix C, Figure 22, Block Erase Flowchart and Pseudo Code, for a suggested flowchart for using the Block Erase command.

**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.

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 internal state machine and starts the Program/ Erase Controller.

If the second bus cycle is not Write Bank Erase Confirm (D0h), Status Register bits b4 and b5 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 Status Register command, all other commands will be ignored. A Bank Erase operation cannot be suspended.

Refer to Dual Operations section for detailed information about simultaneous operations allowed in banks not being erased. Typical Erase times are given in Table 14, Program, Erase Times and Program/Erase Endurance Cycles.

**Program Command**

The memory array can be programmed word-by-word. Only one Word in one bank can be programmed at any one time. 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 Status Register and the Program/Erase Suspend command. Refer to Dual Operations section for detailed information about simultaneous operations allowed in banks not being programmed. Typical Program times are given in Table 14, Program, Erase Times and Program/Erase 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 18, Program Flowchart and Pseudo Code, for the flowchart for using the Program command.

**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 Suspend command. Once the Program/Erase Controller has paused bits 7, 6 and/ or 2 of the Status Register will be set to '1'. The command must be addressed to the bank containing the Program or Erase operation.

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 Protection Program commands will also be accepted. The block being erased may be pro-

tected 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 Dual Operations section 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 21, Program Suspend & Resume Flowchart and Pseudo Code, and Figure 23, Erase Suspend & Resume Flowchart and Pseudo Code for flowcharts for using the Program/Erase Suspend command.

**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 must be written to the bank containing the Program or Erase Suspend.

The Program/Erase Resume command changes the read mode of the target bank to Read Status Register mode.

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 21, Program Suspend & Resume Flowchart and Pseudo Code, and Figure 23, Erase Suspend & Resume Flowchart and Pseudo Code for flowcharts for using the Program/Erase Resume command.

**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. 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. Bit 1 of the Protection Lock Register also protects bit 2 of the Protection Lock Register. Programming bit 2 of the Protection Lock Register will result in a permanent protection of Parameter Block #0 (see Figure 5, Security Block and 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 and/or the Security Block is not reversible.

The Protection Register Program cannot be suspended. See Appendix C, Figure 25, Protection Register Program Flowchart and Pseudo Code, for a flowchart for using the Protection Register Program command.

**Set Configuration Register Command.**

The Set Configuration Register command is used to write a new value to the Configuration Control 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 value for the Configuration Register is always presented on A0-A15. CR0 is on A0, CR1 on A1, etc.; the other address bits are ignored.

**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. 13 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 the section, Block Locking, for a detailed explanation. See Appendix C, Figure

24, Locking Operations Flowchart and Pseudo Code, for a flowchart for using the Lock command.

**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 13 shows the protection status after issuing a Block Unlock command. Refer to the section, Block Locking, for a detailed explanation and Appendix C, Figure 24, Locking Operations Flowchart and Pseudo Code, for a flowchart for using the Unlock command.

**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. 13 shows the Lock Status after issuing a Block Lock-Down command. Refer to the section, Block Locking, for a detailed explanation and Appendix C, Figure 24, Locking Operations Flowchart and Pseudo Code, for a flowchart for using the Lock-Down command.

**Table 5. Standard Commands**

| Commands | Cycles | Bus Operations | | | | | |
|---|---|---|---|---|---|---|---|
| | | 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 | BBKA or BKA(3) | 90h | Read | BBKA or BKA(2,3) | ESD(3) |
| Read CFI Query | 1+ | Write | BBKA | 98h | Read | BBKA(2) | QD |
| Clear Status Register | 1 | Write | BKA | 50h | | | |
| Block Erase | 2 | Write | BKA | 20h | Write | BA | D0h |
| Bank Erase | 2 | Write | BKA | 80h | Write | BKA | D0h |
| Program | 2 | Write | BKA | 40h or 10h | Write | WA | PD |
| Program/Erase Suspend | 1 | Write | BKA | B0h | | | |
| Program/Erase Resume | 1 | Write | BKA | 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 | 60h | Write | BA | 01h |
| Block Unlock | 2 | Write | BKA | 60h | Write | BA | D0h |
| Block Lock-Down | 2 | Write | BKA | 60h | Write | BA | 2Fh |

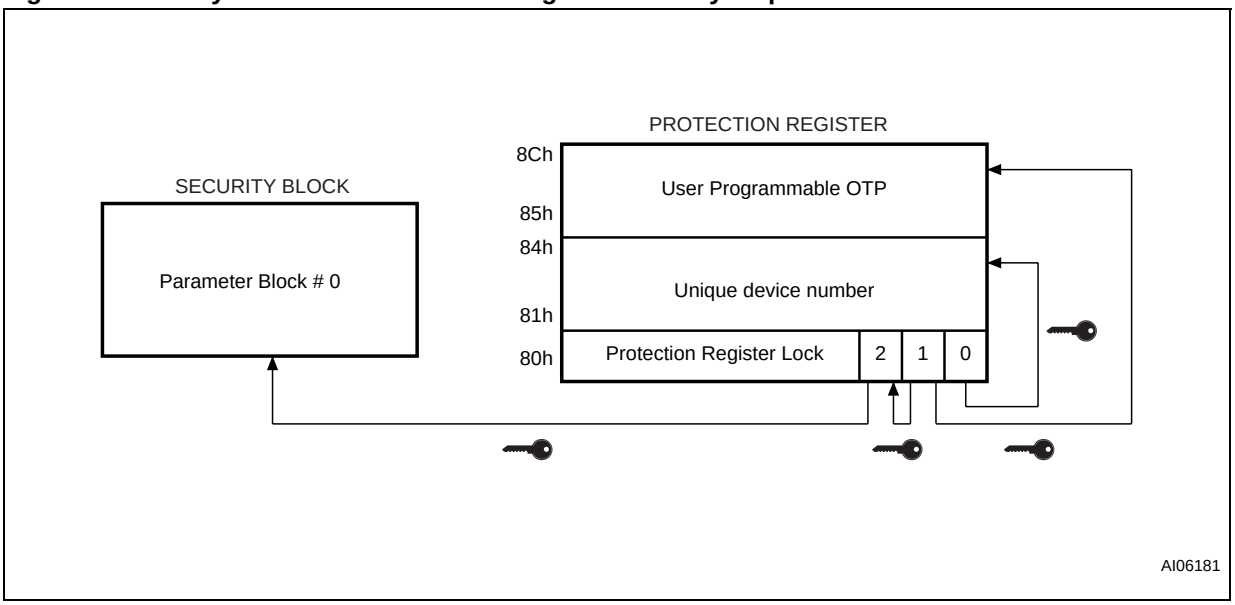
Note: 1. X = Don't Care, WA=Word Address in targeted bank, RD=Read Data, SRD=Status Register Data, ESD=Electronic Signature Data, QD=Query Data, BA=Block Address, BKA= Bank Address, BBKA= Bottom Bank Address, PD=Program Data, PRA=Protection Register Address, PRD=Protection Register Data, CRD=Configuration Register Data.
2. Must be same bank as in the first cycle. The signature addresses are listed in Table 6.
3. When addressed to a block in the Top Bank, reads Block Protection data only.

**Table 6. Electronic Signature Codes**

| Code | | Address (h) | Data (h) |
|---|---|---|---|
| Manufacturer Code | | Bottom Bank Address + 00 | 0020 |
| Device Code | Top (M58CR064C) | Bottom Bank Address + 01 | 88CA |
| | Bottom (M58CR064D) | | 88CB |
| | Top (M58CR064P) | | 8801 |
| | Bottom (M58CR064Q) | | 8802 |
| Block Protection | Lock | Block Address + 02 | 0001 |
| | Unlocked | | 0000 |
| | Locked and Locked-Down | | 0003 |
| | Unlocked and Locked-Down | | 0002 |
| Reserved | | Bottom Bank Address + 03 | Reserved |
| Configuration Register | | Bottom Bank Address + 05 | CR |
| Protection Register Lock | ST Factory Default | Bottom Bank Address + 80 | xx06 |
| | Security Block Permanently Locked | | xx02 |
| | OTP Area Permanently Locked | | xx04 |
| | Security Block and OTP Area Permanently Locked | | xx00 |
| Protection Register | | Bottom Bank Address + 81<br>Bottom Bank Address + 84 | Unique Device Number |
| | | Bottom Bank Address + 85<br>Bottom Bank Address + 8C | OTP Area |

Note: CR=Configuration Register.

**Figure 5. Security Block and Protection Register Memory Map**



## 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.

### 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 A0.

Programming should not be attempted when $V_{PP}$ is not at $V_{PPH}$. The command can be executed if $V_{PP}$ is below $V_{PPH}$ but the result is not guaranteed.

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 Status Register command, all other commands will be ignored. Dual operations are not supported during Double Word Program operations. It is not recommended to suspend the Double Word Program command. Typical Program times are given in Table 14, Program, Erase Times and Program/Erase 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 19, Double Word Program Flowchart and Pseudo Code, for the flowchart for using the Double Word Program command.

### 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 A0 and A1.

Programming should not be attempted when $V_{PP}$ is not at $V_{PPH}$. The command can be executed if $V_{PP}$ is below $V_{PPH}$ but the result is not guaranteed.

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 Status Register command, all other commands will be ignored.

Dual operations are not supported during Quadruple Word Program operations. It is not recommended to suspend the Quadruple Word Program command. Typical Program times are given in Table 14, Program, Erase Times and Program/Erase Endurance Cycles.

See Appendix C, Figure 20, Quadruple Word Program Flowchart and Pseudo Code, for the flowchart for using the Quadruple Word Program command.

**Table 7. Factory Program Commands**

| Command | Phase | Cycles | Bus Write Operations | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 1st | | 2nd | | 3rd | | 4th | | 5th | |
| | | | Add | Data | Add | Data | Add | Data | Add | Data | Add | Data |
| Double Word Program(2) | | 3 | BKA | 30h | WA1 | PD1 | WA2 | PD2 | | | | |
| Quadruple Word Program(3) | | 5 | BKA | 55h | WA1 | PD1 | WA2 | PD2 | WA3 | PD3 | WA4 | PD4 |

Note: 1. WA=Word Address in targeted bank, BKA= Bank Address, PD=Program Data, WA1 is the Start Address.
2. Word Addresses 1 and 2 must be consecutive Addresses differing only for A0.
3. Word Addresses 1,2,3 and 4 must be consecutive Addresses differing only for A0 and A1.

## STATUS REGISTER

The M58CR064 has two Status Registers, one for each bank. The Status Registers provide information on the current or previous Program or Erase operations executed in each bank. Issue a Read Status Register command to read the contents of the Status Register, refer to Read Status Register Command section 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 and SR2 give information on the status of the bank 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.

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.

**Program/Erase Controller Status Bit (SR7).** The Program/Erase Controller Status bit indicates whether the Program/Erase Controller is active or inactive in the addressed 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.

**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 30µs 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.

**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.

**Program Status Bit (SR4).** The Program Status bit is used to identify a Program failure. When the Program Status bit is High (set to '1'), the Program/Erase Controller has applied the maximum number of pulses to the byte and still failed to verify that it has programmed correctly. 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.

**$V_{PP}$ 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.

**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 5µs 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.

**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.

**Reserved Bit (SR0).** SR0 is reserved. Its value must be masked.

**Table 8. Status Register Bits**

| Bit | Name | Type | Logic Level | 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 | Reserved | | | |

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

## CONFIGURATION REGISTER

The Configuration Register is used to configure the type of bus access that the memory will perform. Refer to Read Modes section 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 page read (CR15 = 1). The Configuration Register bits are described in Table 9. They specify the selection of the burst length, burst type, burst X latency and the Read operation. Refer to Figures 6 and 7 for examples of synchronous burst configurations.

### 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.

### 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. For correct operation the X-Latency bits can only assume the values in Table 9, Configuration Register.

The correspondence between X-Latency settings and the maximum sustainable frequency must be calculated taking into account some system parameters. Two conditions must be satisfied:

1. Depending on whether $t_{AVK_CPU}$ or $t_{DELAY}$ is supplied either one of the following two equations must be satisfied:

   $(n + 1)\ t_K \geq t_{ACC} - t_{AVK_CPU} + t_{QVK_CPU}$

   $(n + 2)\ t_K \geq t_{ACC} + t_{DELAY} + t_{QVK_CPU}$

2. and also

   $t_K > t_{KQV} + t_{QVK_CPU}$

where

n is the chosen X-Latency configuration code

$t_K$ is the clock period

$t_{AVK_CPU}$ is clock to address valid, $\overline{L}$ Low, or $\overline{E}$ Low, whichever occurs last

$t_{DELAY}$ is address valid, $\overline{L}$ Low, or $\overline{E}$ Low to clock, whichever occurs last

$t_{QVK_CPU}$ is the data setup time required by the system CPU,

$t_{KQV}$ is the clock to data valid time

$t_{ACC}$ is the random access time of the device.

Refer to Figure 6, X-Latency and Data Output Configuration Example.

### Power-Down Bit (CR10)

The Power-Down bit is used to enable or disable the power-down function.

When the Power-Down bit 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 goes into the power-down state and the supply current, $I_{DD}$, is reduced to the 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 24).

After a reset the Power-Down Bit is set to '0'.

### Wait Configuration Bit (CR8)

In burst mode the Wait bit controls the timing of the Wait output pin, $\overline{WAIT}$. When the Wait bit is '0' the Wait output pin is asserted during the wait state. When the Wait bit is '1' (default) the Wait output pin is asserted one clock cycle before the wait state.

$\overline{WAIT}$ is asserted during a continuous burst and also during a 4 or 8 burst length if no-wrap configuration is selected. $\overline{WAIT}$ is not asserted during asynchronous reads, single synchronous reads or during latency in synchronous reads.

### 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' (default) the memory outputs from sequential addresses. See Tables 10, Burst Type Definition, for the sequence of addresses output from a given starting address in each mode.

### 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.

### Wrap Burst Bit (CR3)

The burst reads can be confined inside the 4 or 8 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.

**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 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 words no-wrap, depending on the starting address, the device asserts the $\overline{WAIT}$ 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 $\overline{WAIT}$ output is not asserted.

If the starting address is shifted by 1,2 or 3 positions from the four word boundary, $\overline{WAIT}$ will be asserted for 1, 2 or 3 clock cycles when the burst sequence crosses the first 64 word boundary, to indicate that the device needs an internal delay to read the successive words in the array. $\overline{WAIT}$ will be asserted only once during a continuous burst access. See also Table 10, Burst Type Definition.

**CR14, CR9, CR5** and **CR4** are reserved for future use.

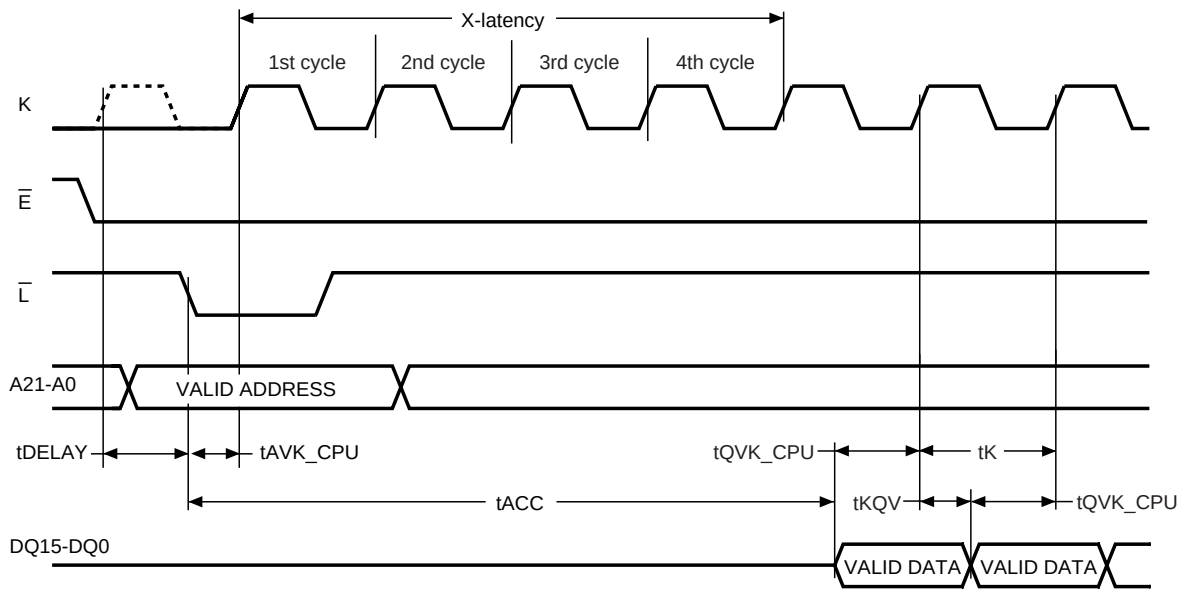
**Table 9. Configuration Register**

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

**Table 10. Burst Type Definition**

| Mode | Start Address | 4 Words | | 8 Words | | Continuous Burst |
|---|---|---|---|---|---|---|
| | | 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... |
| | 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... |
| | 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... |
| | 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... |
| | ... | | | | | |
| | 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... |
| | ... | | | | | |
| | 60 | | | | | 60-61-62-63-64-65-66... |
| | 61 | | | | | 61-62-63-WAIT-64-65-66... |
| | 62 | | | | | 62-63-WAIT-WAIT-64-65-66... |
| | 63 | | | | | 63-WAIT-WAIT-WAIT-64-65-66... |
| | | **Sequential** | **Interleaved** | **Sequential** | **Interleaved** | |
| No-wrap | 0 | 0-1-2-3 | | 0-1-2-3-4-5-6-7 | | 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 | | |
| | 2 | 2-3-4-5 | | 2-3-4-5-6-7-8-9... | | |
| | 3 | 3-4-5-6 | | 3-4-5-6-7-8-9-10 | | |
| | ... | | | | | |
| | 7 | 7-8-9-10 | | 7-8-9-10-11-12-13-14 | | |
| | ... | | | | | |
| | 60 | 60-61-62-63 | | 60-61-62-63-64-65-66-67 | | |
| | 61 | 61-62-63-WAIT-64 | | 61-62-63-WAIT-64-65-66-67-68 | | |
| | 62 | 62-63-WAIT-WAIT-64-65 | | 62-63-WAIT-WAIT-64-65-66-67-68-69 | | |
| | 63 | 63-WAIT-WAIT-WAIT-64-65-66 | | 63-WAIT-WAIT-WAIT-64-65-66-67-68-69-70 | | |

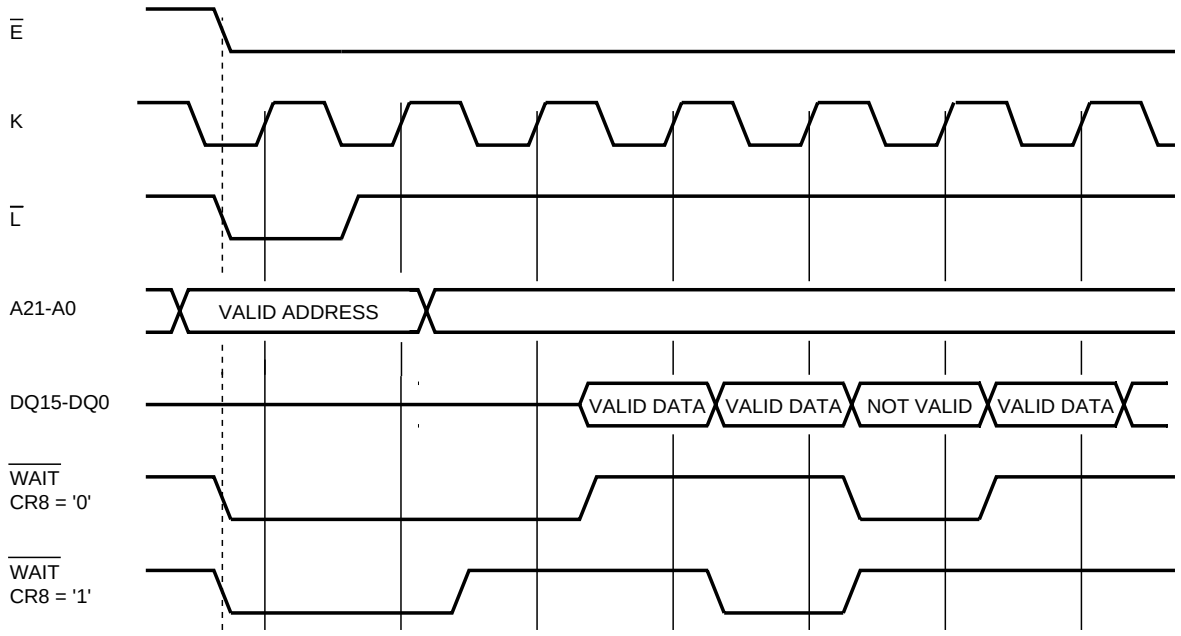
**Figure 6. X-Latency and Data Output Configuration Example**




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

AI90005

**Figure 7. Wait Configuration Example**




AI90006

## 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 Configuration Register section for details). All banks supports both asynchronous and synchronous read operations. The Dual Bank architecture allows read operations in one bank, while write operations are being executed in the other (see Tables 11 and 12).

### 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 a Page of data is internally read and stored in a Page Buffer. The Page has a size of 4 Words and is addressed by A0 and A1 address inputs. The address inputs A0 and A1 are not gated by Latch Enable in Asynchronous Read mode.

The first read operation within the Page has a longer access time ($T_{acc}$, Random access time), subsequent reads within the same Page have much shorter access times. If the Page changes then the normal, longer timings apply again.

Asynchronous Read operations can be performed in two different ways, Asynchronous Random Access Read and Asynchronous Page Read. Only Asynchronous Page Read takes full advantage of the internal page storage so different timings are applied.

See Table 20, Asynchronous Read AC Characteristics, Figure 10, Asynchronous Random Access Read AC Waveform and Figure 11, Asynchronous Page Read AC Waveform for details.

### 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 or 8 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 or 8 Word boundary (Wrap) or overcome the boundary (No Wrap). If the starting address is aligned to a 4 Word Page 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 $\overline{\text{WAIT}}$ 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 64 word boundary and the starting address was at the end of a four word boundary.

$\overline{\text{WAIT}}$ is asserted during X latency and the Wait state and is only deasserted when output data are valid. In Continuous Burst Read mode a Wait state will occur when crossing the first 64 Word boundary. If the burst starting address is aligned to a 4 Word Page, the Wait state will not occur.

The $\overline{\text{WAIT}}$ signal is active Low. The $\overline{\text{WAIT}}$ signal is meaningful only in Synchronous Burst Read mode, in other modes, $\overline{\text{WAIT}}$ is not asserted (except for Read Array mode).

See Table 21, Synchronous Read AC Characteristics and Figure 12, Synchronous Burst Read AC Waveform for details.

### 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. Other Configuration Register parameters have no effect on Single Synchronous Read operations.

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 always asserted.

See Table 21, Synchronous Read AC Characteristics and Figure 13, Single Synchronous Read AC Waveform for details.

## DUAL OPERATIONS AND MULTIPLE BANK ARCHITECTURE

The Dual Operations feature simplifies the software management of the device and allows code to be executed from one bank while the other bank is being programmed or erased.

The Dual operations feature means that while programming or erasing in one bank, Read operations are possible in the other 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 the other bank in Read mode. Bus Read operations are allowed in the other 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.

Tables 11 and 12 show the dual operations possible in the other bank 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 11. Dual Operations Allowed In Other Bank**

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

**Table 12. 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 | Erase | Program/ Erase Suspend | Program/ Erase Resume |
| Idle | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Programming | – | Yes | – | – | – | – | Yes | – |
| Erasing | – | Yes | – | – | – | – | Yes | – |
| Program Suspended | Yes(1) | Yes | Yes | Yes | – | – | – | Yes |
| Erase Suspended | Yes(1) | Yes | Yes | Yes | Yes(1) | – | – | Yes |

Note: 1. Not allowed in the Block or Word that is being erased or programmed.

## BLOCK LOCKING

The M58CR064 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 (M58CR064C/D only).

The first two levels (Lock/Unlock and Lock-Down) are available in M58CR064C/D and M58CR064P/Q. The third level ($V_{PP} \leq V_{PPLK}$) is only available for the M58CR064C/D versions, in the M58CR064P/Q this feature has been disabled.

For all devices the protection status of each block can be set to Locked, Unlocked, and Lock-Down. Table 13, defines all of the possible protection states ($\overline{WP}$, DQ1, DQ0), and Appendix C, Figure 24, shows a flowchart for the locking operations.

### 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.

### 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.

### 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.

### 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.

### 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 Table, for detailed information on which commands are valid during erase suspend.

**Table 13. Lock Status**

| Current Protection Status(1) ($\overline{WP}$, DQ1, DQ0) | | Next Protection Status(1) ($\overline{WP}$, DQ1, DQ0) | | | |
|---|---|---|---|---|---|
| **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) |

Note: 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.

## 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 14. In the M58CR064 the maximum number of Program/ Erase cycles depends on the voltage supply used.

**Table 14. Program, Erase Times and Program, Erase Endurance Cycles**

| | Parameter | Condition | Min | Typ | Typical after 100k W/E Cycles | Max | Unit |
|---|---|---|---|---|---|---|---|
| $V_{PP} = V_{DD}$ | Parameter Block (4 KWord) Erase(2) | | | 0.3 | 1 | 2.5 | s |
| | Main Block (32 KWord) Erase | Preprogrammed | | 0.8 | 3 | 4 | s |
| | | Not Preprogrammed | | 1.1 | | 4 | s |
| | Bank A (16Mbit) Erase | Preprogrammed | | 11 | | | s |
| | | Not Preprogrammed | | 18 | | | s |
| | Bank B (48Mbit) Erase | Preprogrammed | | 33 | | | s |
| | | Not Preprogrammed | | 54 | | | s |
| | Parameter Block (4 KWord) Program(3) | | | 40 | | | ms |
| | Main Block (32 KWord) Program(3) | | | 300 | | | ms |
| | Word Program (3) | | | 10 | 10 | 100 | µs |
| | Program Suspend Latency | | | 5 | | 10 | µs |
| | Erase Suspend Latency | | | 5 | | 20 | µs |
| | Program/Erase Cycles (per Block) | Main Blocks | 100,000 | | | | cycles |
| | | Parameter Blocks | 100,000 | | | | cycles |
| $V_{PP} = V_{PPH}$ | Parameter Block (4 KWord) Erase | | | 0.3 | | 2.5 | s |
| | Main Block (32 KWord) Erase | | | 0.9 | | 4 | s |
| | Bank A (16Mbit) Erase | | | 13 | | | s |
| | Bank B (48Mbit) Erase | | | 39 | | | s |
| | 4Mbit Program | Quadruple Word | | 510 | | | ms |
| | Word/ Double Word/ Quadruple Word Program(3) | | | 8 | | 100 | µs |
| | Parameter Block (4 KWord) Program(3) | Quadruple Word | | 8 | | | ms |
| | | Word | | 32 | | | ms |
| | Main Block (32 KWord) Program(3) | Quadruple Word | | 64 | | | ms |
| | | Word | | 256 | | | ms |
| | Program/Erase Cycles (per Block) | Main Blocks | | | | 1000 | cycles |
| | | Parameter Blocks | | | | 2500 | cycles |

Note: 1. $T_A$ = –40 to 85°C; $V_{DD}$ = 1.65V to 2V; $V_{DDQ}$ = 1.65V to 3.3V.
2. The difference between Preprogrammed and not preprogrammed is not significant (‹30ms).
3. Excludes the time needed to execute the command sequence.

## 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 STMicroelectronics SURE Program and other relevant quality documents.

**Table 15. Absolute Maximum Ratings**

| | | Value | | |
|---|---|---|---|---|
| **Symbol** | **Parameter** | **Min** | **Max** | **Unit** |
| $T_A$ | Ambient Operating Temperature | –40 | 85 | °C |
| $T_{BIAS}$ | Temperature Under Bias | –40 | 125 | °C |
| $T_{STG}$ | Storage Temperature | –55 | 155 | °C |
| $V_{IO}$ | Input or Output Voltage | –0.5 | $V_{DDQ}$+0.5 | V |
| $V_{DD}$ | Supply Voltage | –0.5 | 2.7 | V |
| $V_{DDQ}$ | Input/Output Supply Voltage | –0.5 | 3.6 | V |
| $V_{PP}$ | Program Voltage | –0.5 | 13 | V |
| $I_O$ | Output Short Circuit Current | | 100 | mA |
| $t_{VPPH}$ | Time for $V_{PP}$ at $V_{PPH}$ | | 100 | hours |

## 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 16, 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 16. Operating and AC Measurement Conditions**

| Parameter | M58CR064C, M58CR064D, M58CR064P, M58CR064Q | | | | | | |
|---|---|---|---|---|---|---|---|
| | 85 | | 100 | | 120 | | Units |
| | Min | Max | Min | Max | Min | Max | |
| $V_{DD}$ Supply Voltage | 1.8 | 2.0 | 1.65 | 2.0 | 1.65 | 2.0 | V |
| $V_{DDQ}$ Supply Voltage | 1.8 | 3.3 | 1.65 | 3.3 | 1.65 | 3.3 | V |
| $V_{PP}$ Supply Voltage (Factory environment) | 11.4 | 12.6 | 11.4 | 12.6 | 11.4 | 12.6 | V |
| $V_{PP}$ Supply Voltage (Application environment) | -0.4 | $V_{DDQ}$ +0.4 | -0.4 | $V_{DDQ}$ +0.4 | -0.4 | $V_{DDQ}$ +0.4 | V |
| Ambient Operating Temperature | – 40 | 85 | – 40 | 85 | – 40 | 85 | °C |
| Load Capacitance ($C_L$) | 30 | | 30 | | 30 | | pF |
| Input Rise and Fall Times | | 4 | | 4 | | 4 | ns |
| Input Pulse Voltages | 0 to $V_{DDQ}$ | | 0 to $V_{DDQ}$ | | 0 to $V_{DDQ}$ | | V |
| Input and Output Timing Ref. Voltages | $V_{DDQ}/2$ | | $V_{DDQ}/2$ | | $V_{DDQ}/2$ | | V |

**Figure 8. AC Measurement I/O Waveform**

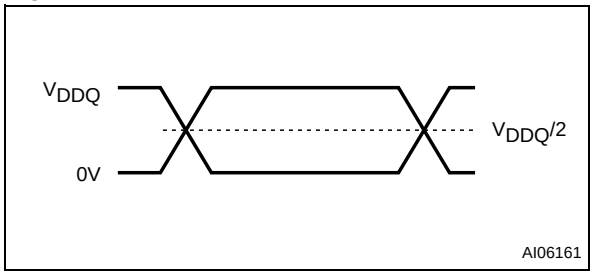



**Figure 9. AC Measurement Load Circuit**

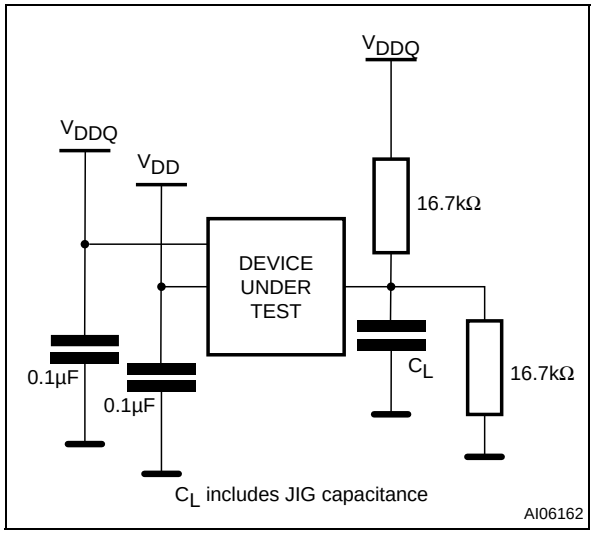



**Table 17. Capacitance**

| 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 |

Note: Sampled only, not 100% tested.

**Table 18. DC Characteristics - Currents**

| Symbol | Parameter | Test Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Leakage Current | $0V \le V_{IN} \le V_{DDQ}$ | | | ±1 | µA |
| $I_{LO}$ | Output Leakage Current | $0V \le V_{OUT} \le V_{DDQ}$ | | | ±1 | µA |
| $I_{DD1}$ | Supply Current Asynchronous Read (f=6MHz) | $\overline{E} = V_{IL}$, $\overline{G} = V_{IH}$ | | 3 | 6 | mA |
| | Supply Current Synchronous Read (f=40MHz) | 4 Word | | 6 | 13 | mA |
| | | 8 Word | | 8 | 14 | mA |
| | | Continuous | | 6 | 10 | mA |
| | Supply Current Synchronous Read (f=54MHz) | 4 Word | | 7 | 16 | mA |
| | | 8 Word | | 10 | 18 | mA |
| | | Continuous | | 13 | 25 | mA |
| $I_{DD2}$ | Supply Current (Power-Down) | $\overline{RP} = V_{SS} \pm 0.2V$ | | 2 | 10 | µA |
| $I_{DD3}$ | Supply Current (Standby) | $\overline{E} = V_{DD} \pm 0.2V$ | | 10 | 50 | µA |
| $I_{DD4}$ (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 |
| $I_{DD5}$ (1,2) | Supply Current (Dual Operations) | Program/Erase in one Bank, Asynchronous Read in another Bank | | 13 | 26 | mA |
| | | Program/Erase in one Bank, Synchronous Read in another Bank | | 16 | 30 | mA |
| $I_{DD6}$(1) | Supply Current Program/ Erase Suspended (Standby) | $\overline{E} = V_{DD} \pm 0.2V$ | | 10 | 50 | µA |
| $I_{PP1}$(1) | $V_{PP}$ Supply Current (Program) | $V_{PP} = V_{PPH}$ | | 2 | 5 | mA |
| | | $V_{PP} = V_{DD}$ | | 0.2 | 5 | µA |
| | $V_{PP}$ Supply Current (Erase) | $V_{PP} = V_{PPH}$ | | 2 | 5 | mA |
| | | $V_{PP} = V_{DD}$ | | 0.2 | 5 | µA |
| $I_{PP2}$ | $V_{PP}$ Supply Current (Read) | $V_{PP} = V_{PPH}$ | | 100 | 400 | µA |
| | | $V_{PP} \le V_{DD}$ | | 0.2 | 5 | µA |
| $I_{PP3}$(1) | $V_{PP}$ Supply Current (Standby) | $V_{PP} \le V_{DD}$ | | 0.2 | 5 | µA |

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

**Table 19. 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 | 1.8 | 1.95 | V |
| $V_{PPH}$ | $V_{PP}$ Program Voltage Factory | Program, Erase | 11.4 | 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 10. Asynchronous Random Access Read AC Waveforms**

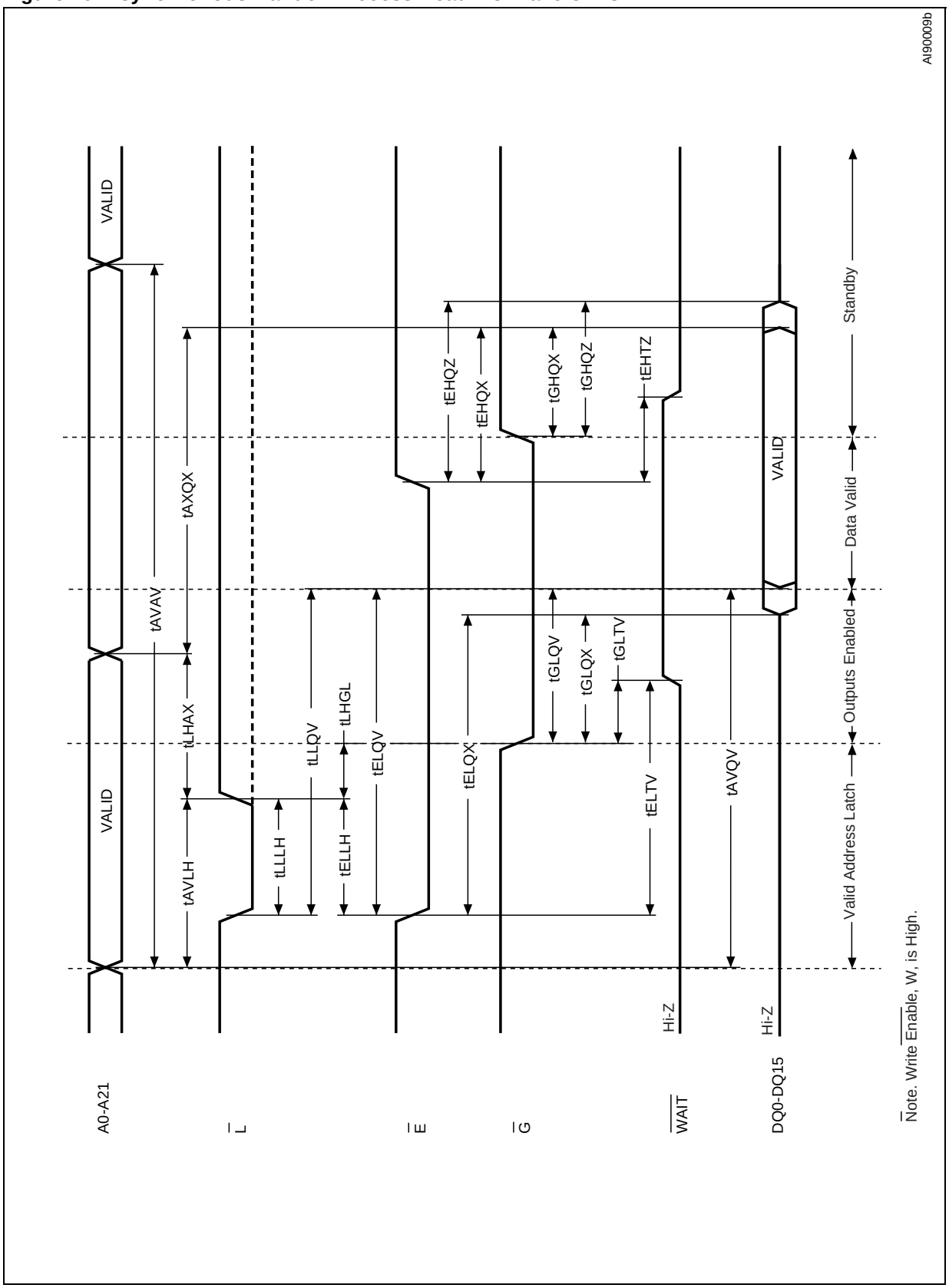


**Figure 11. Asynchronous Page Read AC Waveforms**

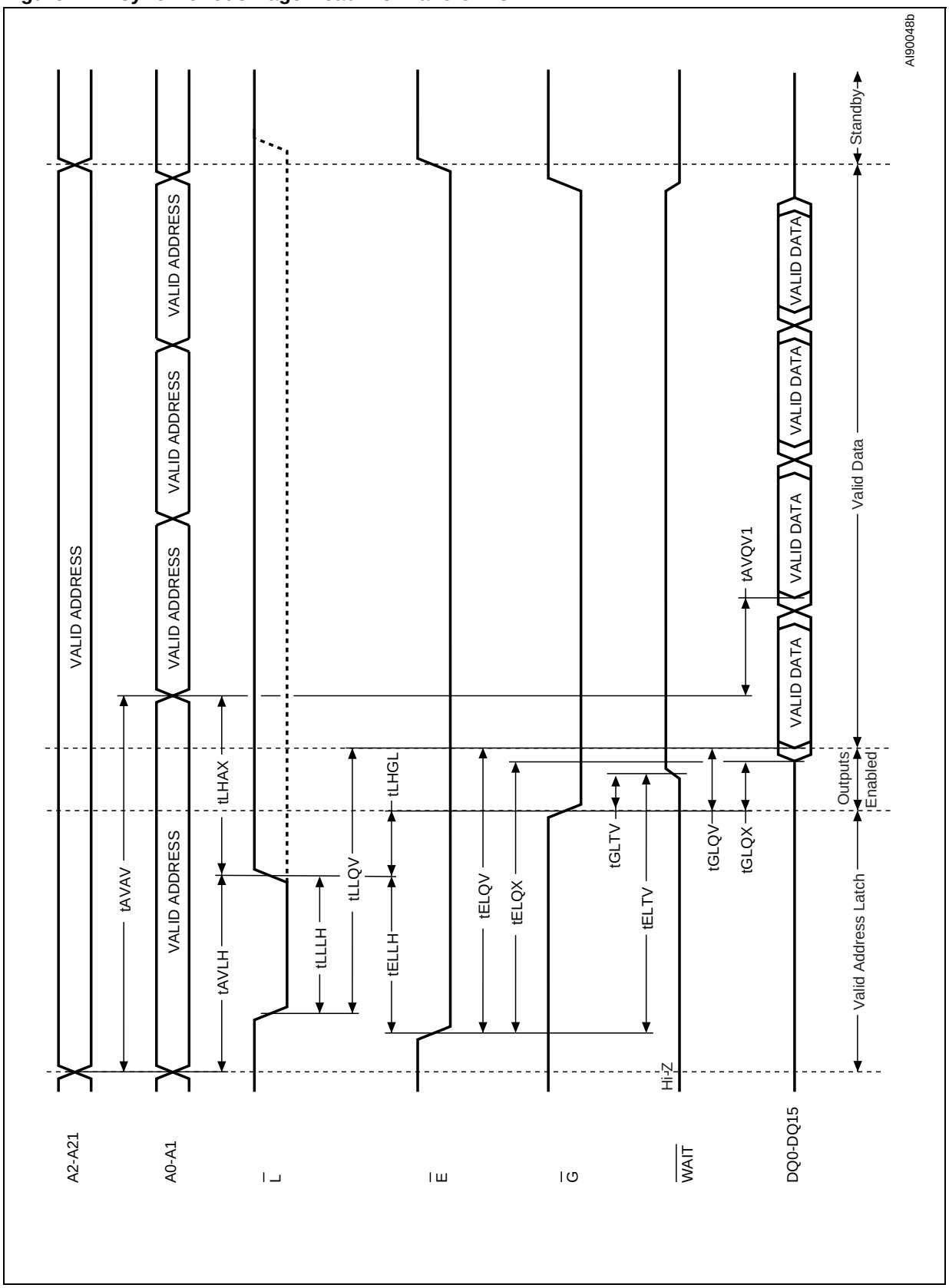
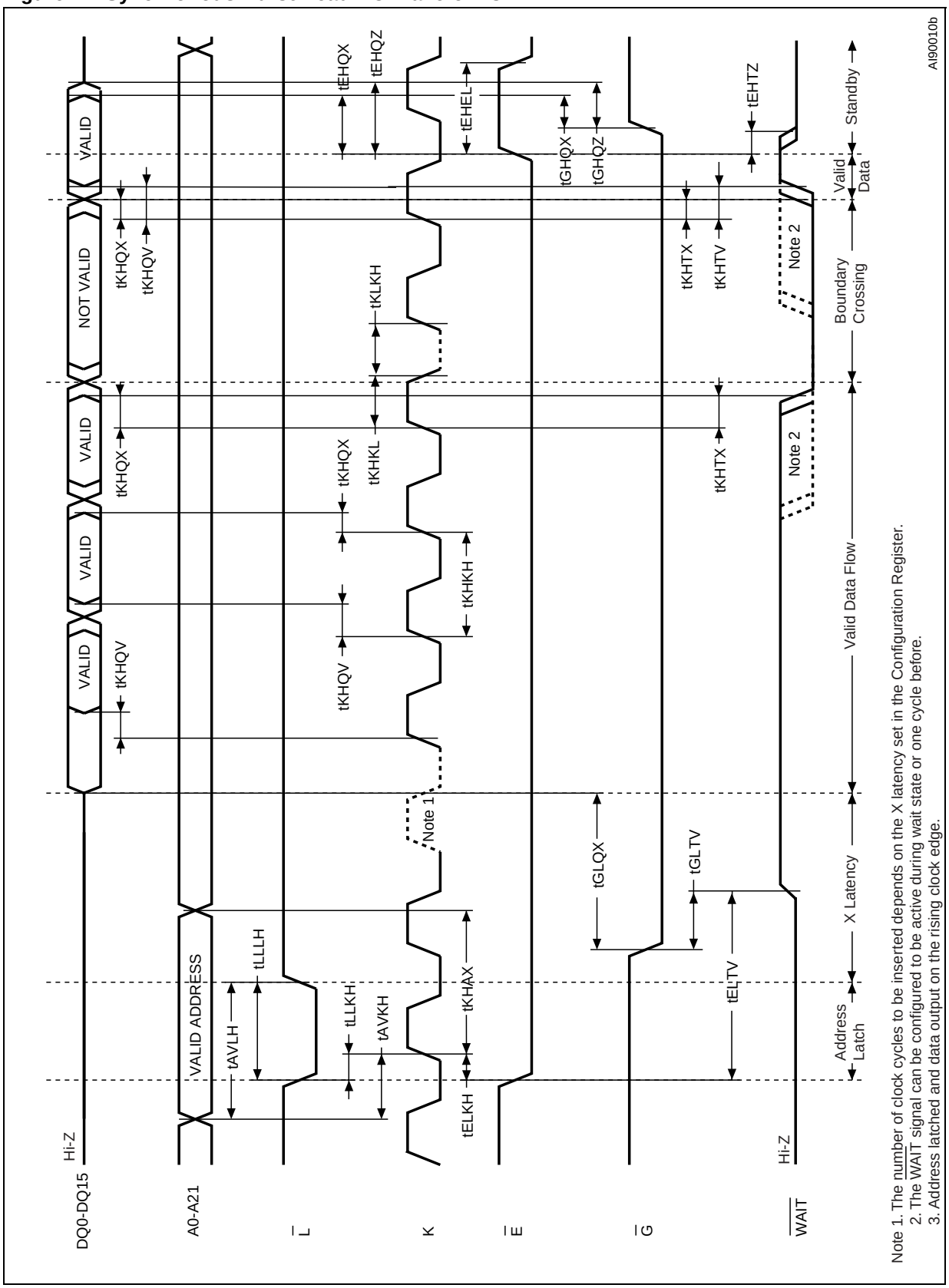


**Table 20. Asynchronous Read AC Characteristics**

| Symbol | | Alt | Parameter | | M58CR064 | | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | | | | 85 | 100 | 120 | |
| Read Timings | $t_{AVAV}$ | $t_{RC}$ | Address Valid to Next Address Valid | Min | 85 | 100 | 120 | ns |
| | $t_{AVQV}$ | $t_{ACC}$ | Address Valid to Output Valid (Random) | Max | 85 | 100 | 120 | ns |
| | $t_{AVQV1}$ | $t_{PAGE}$ | Address Valid to Output Valid (Page) | Max | 30 | 45 | 45 | ns |
| | $t_{AXQX}$ (1) | $t_{OH}$ | Address Transition to Output Transition | Min | 0 | 0 | 0 | ns |
| | $t_{ELTV}$ | | Chip Enable Low to Wait Valid | Max | 14 | 14 | 18 | ns |
| | $t_{ELQV}$ (2) | $t_{CE}$ | Chip Enable Low to Output Valid | Max | 85 | 100 | 120 | ns |
| | $t_{ELQX}$ (1) | $t_{LZ}$ | Chip Enable Low to Output Transition | Min | 0 | 0 | 0 | ns |
| | $t_{EHTZ}$ | | Chip Enable High to Wait Hi-Z | Max | 20 | 20 | 20 | ns |
| | $t_{EHQX}$ (1) | $t_{OH}$ | Chip Enable High to Output Transition | Min | 0 | 0 | 0 | ns |
| | $t_{EHQZ}$ (1) | $t_{HZ}$ | Chip Enable High to Output Hi-Z | Max | 20 | 20 | 20 | ns |
| | $t_{GLQV}$ (2) | $t_{OE}$ | Output Enable Low to Output Valid | Max | 25 | 25 | 25 | ns |
| | $t_{GLQX}$ (1) | $t_{OLZ}$ | Output Enable Low to Output Transition | Min | 0 | 0 | 0 | ns |
| | $t_{GLTV}$ | | Output Enable Low to Wait Valid | Max | 14 | 14 | 18 | ns |
| | $t_{GHQX}$ (1) | $t_{OH}$ | Output Enable High to Output Transition | Min | 0 | 0 | 0 | ns |
| | $t_{GHQZ}$ (1) | $t_{DF}$ | Output Enable High to Output Hi-Z | Max | 20 | 20 | 20 | ns |
| Latch Timings | $t_{AVLH}$ | $t_{AVADVH}$ | Address Valid to Latch Enable High | Min | 10 | 10 | 10 | ns |
| | $t_{ELLH}$ | $t_{ELADVH}$ | Chip Enable Low to Latch Enable High | Min | 10 | 10 | 10 | ns |
| | $t_{LHAX}$ | $t_{ADVHAX}$ | Latch Enable High to Address Transition | Min | 10 | 10 | 10 | ns |
| | $t_{LLLH}$ | $t_{ADVLADVH}$ | Latch Enable Pulse Width | Min | 10 | 10 | 10 | ns |
| | $t_{LLQV}$ | $t_{ADVLQV}$ | Latch Enable Low to Output Valid (Random) | Max | 85 | 100 | 120 | ns |
| | $t_{LHGL}$ | $t_{ADVHGL}$ | Latch Enable High to Output Enable Low | Min | 10 | 10 | 10 | ns |

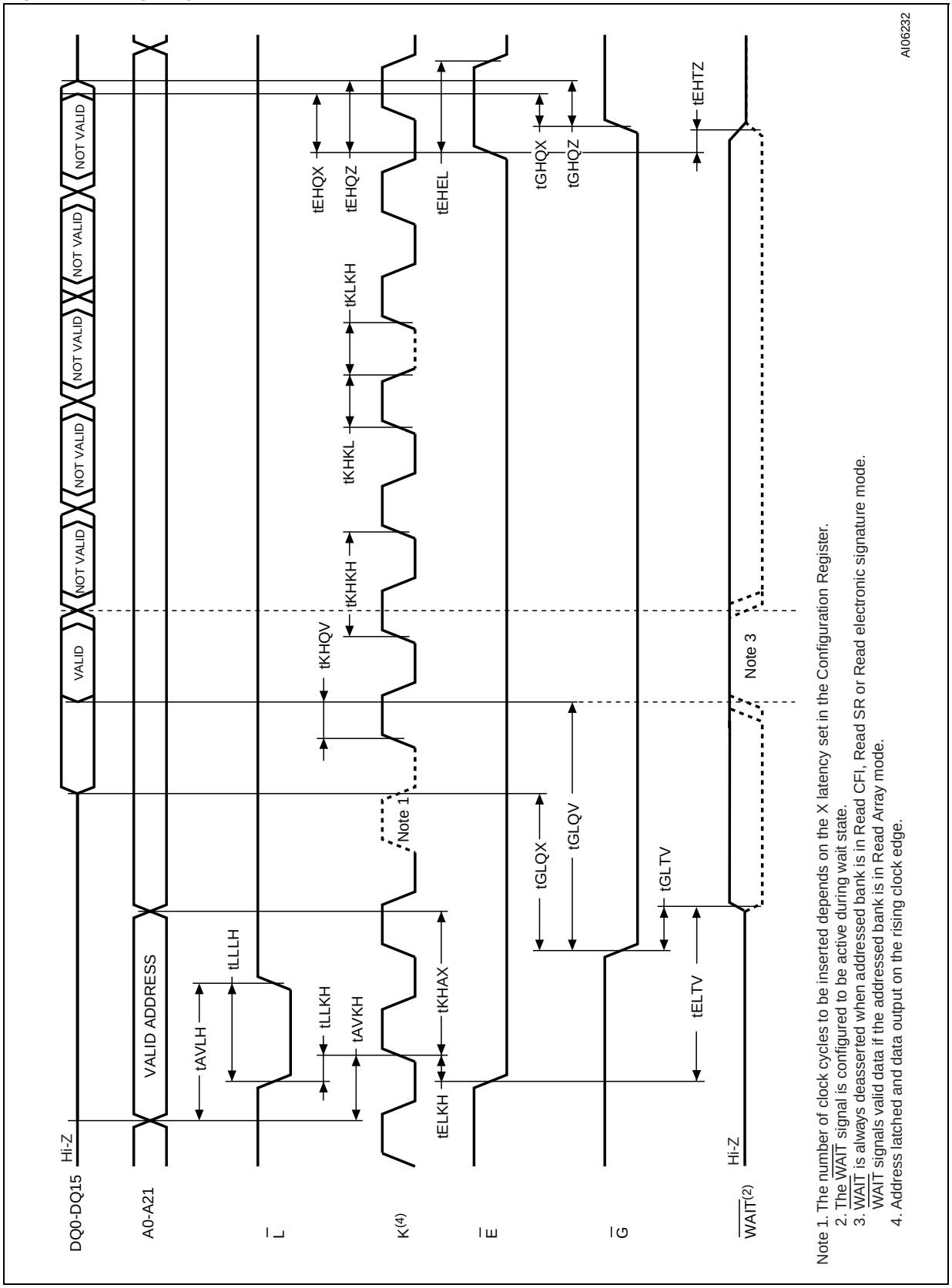
Note: 1. Sampled only, not 100% tested.
2. $\overline{G}$ may be delayed by up to $t_{ELQV}$ - $t_{GLQV}$ after the falling edge of $\overline{E}$ without increasing $t_{ELQV}$.

**Figure 12. 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.
3. Address latched and data output on the rising clock edge.

**Figure 13. 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.
3. WAIT is always deasserted when addressed bank is in Read CFI, Read SR or Read electronic signature mode. WAIT signals valid data if the addressed bank is in Read Array mode.
4. Address latched and data output on the rising clock edge.

**Table 21. Synchronous Read AC Characteristics**

| Symbol | | Alt | Parameter | | M58CR064 | | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | | | | 85 | 100 | 120 | |
| Synchronous Read Timings | $t_{AVKH}$ | $t_{AVCLKH}$ | Address Valid to Clock High | Min | 7 | 7 | 7 | ns |
| | $t_{ELKH}$ | $t_{ELCLKH}$ | Chip Enable Low to Clock High | Min | 7 | 7 | 7 | ns |
| | $t_{ELTV}$ | | Chip Enable Low to Wait Valid | Max | 14 | 14 | 18 | ns |
| | $t_{EHEL}$ | | Chip Enable Pulse Width (subsequent synchronous reads) | Min | 20 | 20 | 20 | ns |
| | $t_{EHTZ}$ | | Chip Enable High to Wait Hi-Z | Max | 20 | 20 | 20 | ns |
| | $t_{KHAX}$ | $t_{CLKHAX}$ | Clock High to Address Transition | Min | 10 | 10 | 10 | ns |
| | $t_{KHQV}$<br>$t_{KHTV}$ | $t_{CLKHQV}$ | Clock High to Output Valid<br>Clock High to WAIT Valid | Max | 14 | 14 | 18 | ns |
| | $t_{KHQX}$<br>$t_{KHTX}$ | $t_{CLKHQX}$ | Clock High to Output Transition<br>Clock High to WAIT Transition | Min | 4 | 4 | 4 | ns |
| | $t_{LLKH}$ | $t_{ADVLCLKH}$ | Latch Enable Low to Clock High | Min | 7 | 7 | 7 | ns |
| Clock Specifications | $t_{KHKH}$ | $t_{CLK}$ | Clock Period (f=40MHz) | Min | | | 25 | ns |
| | | | Clock Period (f=54MHz) | Min | 18 | 18 | | ns |
| | $t_{KHKL}$ | $t_{CLKHCLKL}$ | Clock High to Clock Low | Min | 5 | 5 | 5 | ns |
| | $t_{KLKH}$ | $t_{CLKLCLKH}$ | Clock Low to Clock High | Max | 5 | 5 | 5 | ns |

Note: 1. Sampled only, not 100% tested.
2. For other timings please refer to Table 20, Asynchronous Read AC Characteristics.

**Figure 14. Write AC Waveforms, Write Enable Controlled**

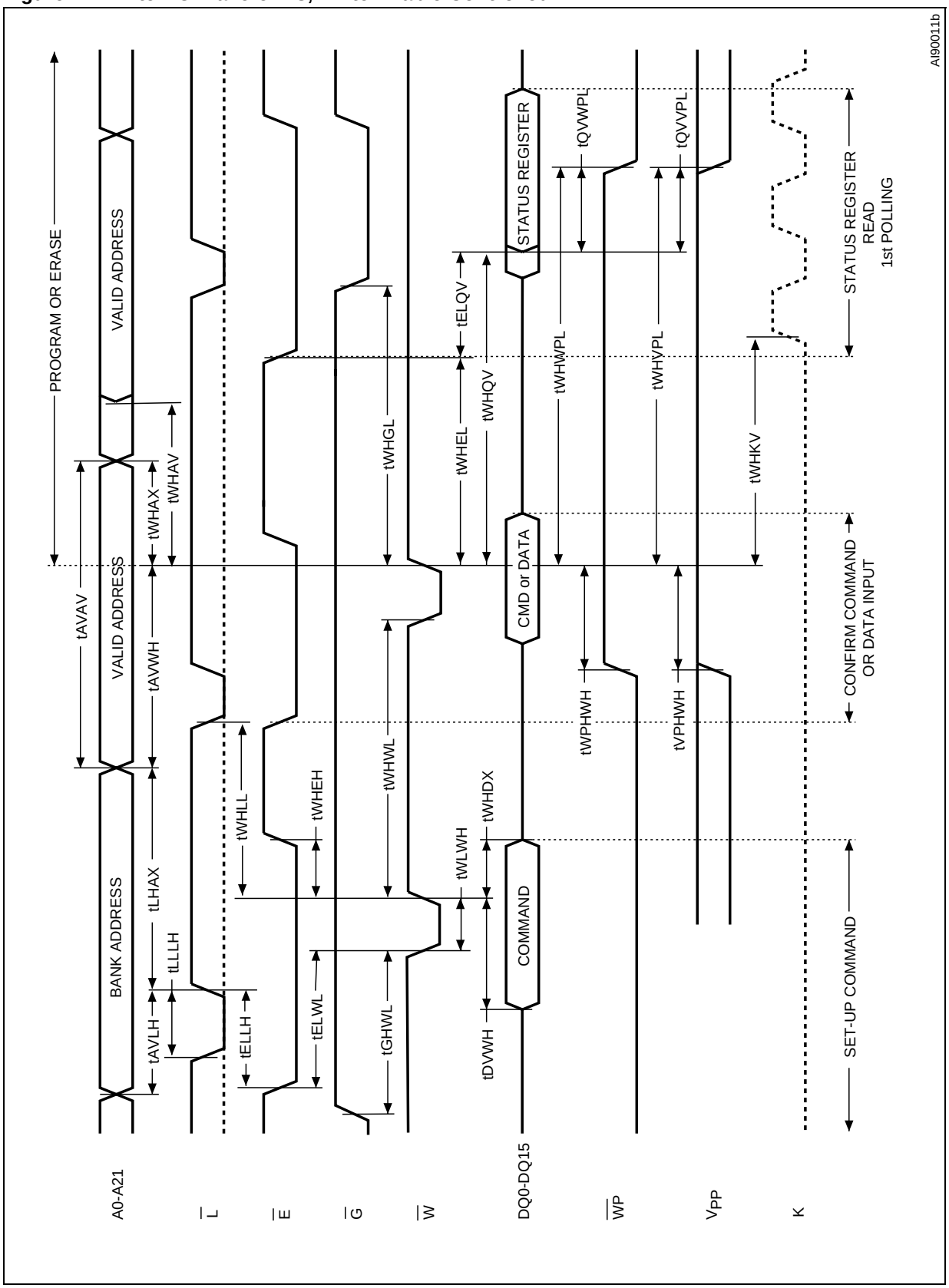


**Table 22. Write AC Characteristics, Write Enable Controlled**

| | Symbol | Alt | Parameter | | M58CR064 | | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | | | | 85 | 100 | 120 | |
| Write Enable Controlled Timings | $t_{AVAV}$ | $t_{WC}$ | Address Valid to Next Address Valid | Min | 85 | 100 | 120 | ns |
| | $t_{AVLH}$ | | Address Valid to Latch Enable High | Min | 10 | 10 | 10 | ns |
| | $t_{AVWH}$ | $t_{WC}$ | Address Valid to Write Enable High | Min | 60 | 60 | 60 | ns |
| | $t_{DVWH}$ | $t_{DS}$ | Input Valid to Write Enable High | Min | 40 | 40 | 40 | ns |
| | $t_{ELLH}$ | | Chip Enable Low to Latch Enable High | Min | 10 | 10 | 10 | ns |
| | $t_{ELWL}$ | $t_{CS}$ | Chip Enable Low to Write Enable Low | Min | 0 | 0 | 0 | ns |
| | $t_{ELQV}$ | | Chip Enable Low to Output Valid | Min | 85 | 100 | 120 | ns |
| | $t_{GHWL}$ | | Output Enable High to Write Enable Low | Min | 20 | 20 | 20 | ns |
| | $t_{LHAX}$ | | Latch Enable High to Address Transition | Min | 10 | 10 | 10 | ns |
| | $t_{LLLH}$ | | Latch Enable Pulse Width | Min | 10 | 10 | 10 | ns |
| | $t_{WHAV}$ | | Write Enable High to Address Valid | Min | 0 | 0 | 0 | ns |
| | $t_{WHAX}$ | $t_{AH}$ | Write Enable High to Address Transition | Min | 0 | 0 | 0 | ns |
| | $t_{WHDX}$ | $t_{DH}$ | Write Enable High to Input Transition | Min | 0 | 0 | 0 | ns |
| | $t_{WHEH}$ | $t_{CH}$ | Write Enable High to Chip Enable High | Min | 0 | 0 | 0 | ns |
| | $t_{WHEL}$ (2) | | Write Enable High to Chip Enable Low | Min | 50 | 50 | 50 | ns |
| | $t_{WHGL}$ | | Write Enable High to Output Enable Low | Min | 0 | 0 | 0 | ns |
| | $t_{WHLL}$ | | Write Enable High to Latch Enable Low | Min | 0 | 0 | 0 | ns |
| | $t_{WHKV}$ | | Write Enable High to Clock Valid | Min | 25 | 25 | 25 | ns |
| | $t_{WHWL}$ | $t_{WPH}$ | Write Enable High to Write Enable Low | Min | 30 | 30 | 30 | ns |
| | $t_{WHQV}$ | | Write Enable High to Output Valid | Min | 105 | 120 | 140 | ns |
| | $t_{WLWH}$ | $t_{WP}$ | Write Enable Low to Write Enable High | Min | 50 | 50 | 50 | ns |
| Protection Timings | $t_{QVVPL}$ | | Output (Status Register) Valid to $V_{PP}$ Low | Min | 0 | 0 | 0 | ns |
| | $t_{QVWPL}$ | | Output (Status Register) Valid to Write Protect Low | Min | 0 | 0 | 0 | ns |
| | $t_{VPHWH}$ | $t_{VPS}$ | $V_{PP}$ High to Write Enable High | Min | 200 | 200 | 200 | ns |
| | $t_{WHVPL}$ | | Write Enable High to $V_{PP}$ Low | Min | 200 | 200 | 200 | ns |
| | $t_{WHWPL}$ | | Write Enable High to Write Protect Low | Min | 200 | 200 | 200 | ns |
| | $t_{WPHWH}$ | | Write Protect High to Write Enable High | Min | 200 | 200 | 200 | ns |

Note: 1. Sampled only, not 100% tested.
2. $t_{WHEL}$ has the values shown when reading in the targeted bank. 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 a command. If the read operation is in a different bank $t_{WHEL}$ is 0ns.

**Figure 15. Write AC Waveforms, Chip Enable Controlled**

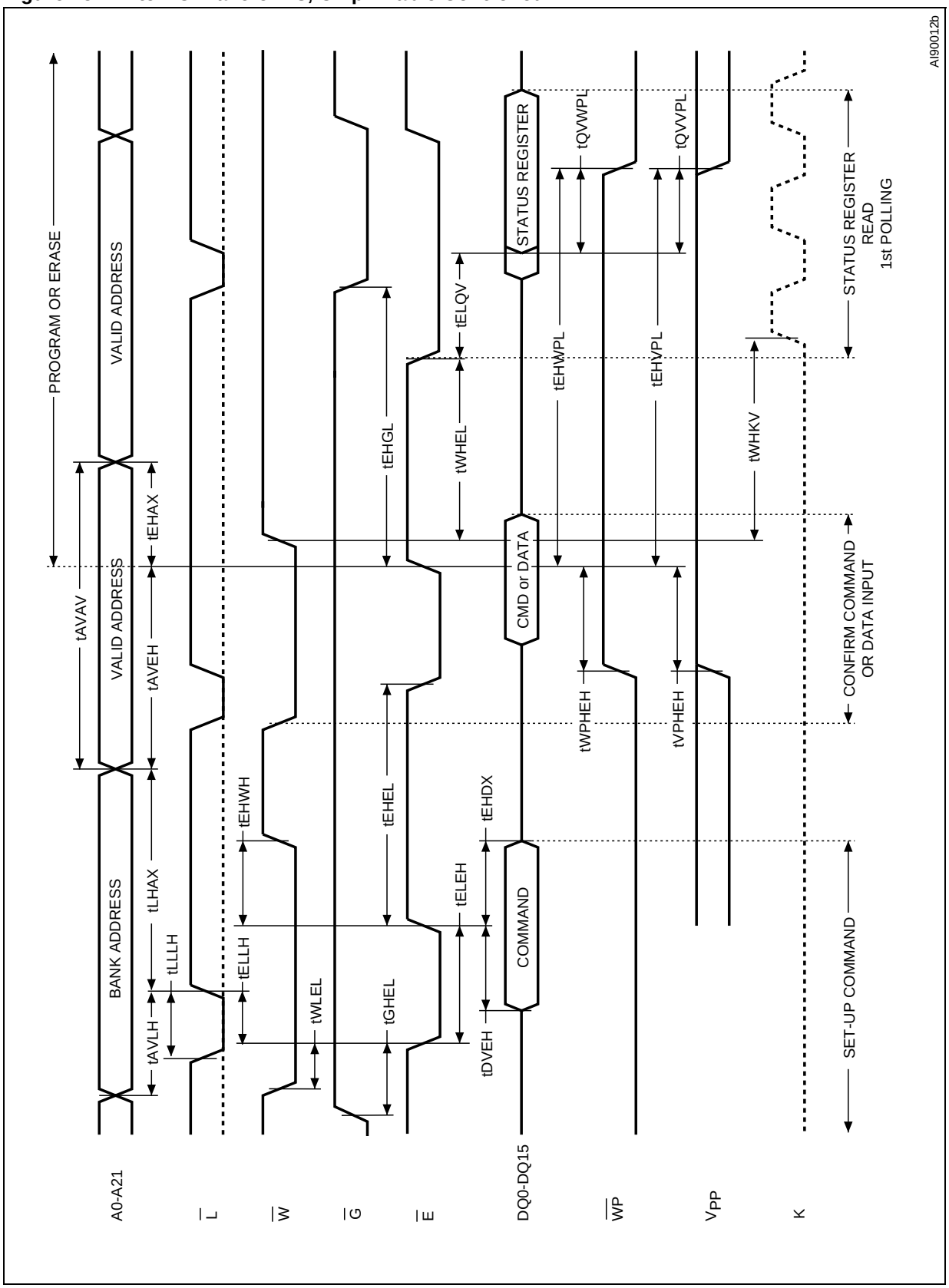


**Table 23. Write AC Characteristics, Chip Enable Controlled**

| Symbol | | Alt | Parameter | | M58CR064 | | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | | | | 85 | 100 | 120 | |
| Chip Enable Controlled Timings | $t_{AVAV}$ | $t_{WC}$ | Address Valid to Next Address Valid | Min | 85 | 100 | 120 | ns |
| | $t_{AVEH}$ | $t_{WC}$ | Address Valid to Chip Enable High | Min | 60 | 60 | 60 | ns |
| | $t_{AVLH}$ | | Address Valid to Latch Enable High | Min | 10 | 10 | 10 | ns |
| | $t_{DVEH}$ | $t_{DS}$ | Input Valid to Write Enable High | Min | 40 | 40 | 40 | ns |
| | $t_{EHAX}$ | $t_{AH}$ | Chip Enable High to Address Transition | Min | 0 | 0 | 0 | ns |
| | $t_{EHDX}$ | $t_{DH}$ | Chip Enable High to Input Transition | Min | 0 | 0 | 0 | ns |
| | $t_{EHEL}$ | $t_{WPH}$ | Chip Enable High to Chip Enable Low | Min | 30 | 30 | 30 | ns |
| | $t_{EHGL}$ | | Chip Enable High to Output Enable Low | Min | 0 | 0 | 0 | ns |
| | $t_{EHWH}$ | $t_{CH}$ | Chip Enable High to Write Enable High | Min | 0 | 0 | 0 | ns |
| | $t_{ELEH}$ | $t_{WP}$ | Chip Enable Low to Chip Enable High | Min | 60 | 60 | 60 | ns |
| | $t_{ELLH}$ | | Chip Enable Low to Latch Enable High | Min | 10 | 10 | 10 | ns |
| | $t_{ELQV}$ | | Latch Enable Low to Output Valid | Min | 85 | 100 | 120 | ns |
| | $t_{GHEL}$ | | Output Enable High to Chip Enable Low | Min | 20 | 20 | 20 | ns |
| | $t_{LHAX}$ | | Latch Enable High to Address Transition | Min | 10 | 10 | 10 | ns |
| | $t_{LLLH}$ | | Latch Enable Pulse Width | Min | 10 | 10 | 10 | ns |
| | $t_{WHEL}$ (2) | | Write Enable High to Chip Enable Low | Min | 50 | 50 | 50 | ns |
| | $t_{WHKV}$ | | Write Enable High to Clock Valid | Min | 25 | 25 | 25 | ns |
| | $t_{WLEL}$ | $t_{CS}$ | Write Enable Low to Chip Enable Low | Min | 0 | 0 | 0 | ns |
| Protection Timings | $t_{EHVPL}$ | | Chip Enable High to $V_{PP}$ Low | Min | 200 | 200 | 200 | ns |
| | $t_{EHWPL}$ | | Chip Enable High to Write Protect Low | Min | 200 | 200 | 200 | ns |
| | $t_{QVVPL}$ | | Output (Status Register) Valid to $V_{PP}$ Low | Min | 0 | 0 | 0 | ns |
| | $t_{QVWPL}$ | | Output (Status Register) Valid to Write Protect Low | Min | 0 | 0 | 0 | ns |
| | $t_{VPHEH}$ | $t_{VPS}$ | $V_{PP}$ High to Chip Enable High | Min | 200 | 200 | 200 | ns |
| | $t_{WPHEH}$ | | Write Protect High to Chip Enable High | Min | 200 | 200 | 200 | ns |

Note: 1. Sampled only, not 100% tested.
2. $t_{WHEL}$ has the values shown when reading in the targeted bank. 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 a command. If the read operation is in a different bank $t_{WHEL}$ is 0ns.

**Figure 16. Reset and Power-up AC Waveforms**

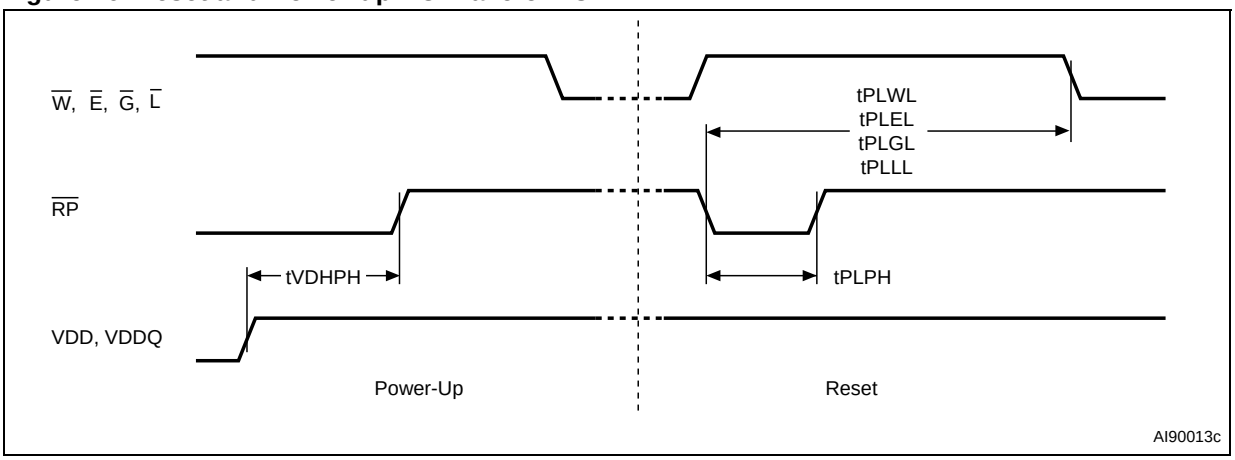


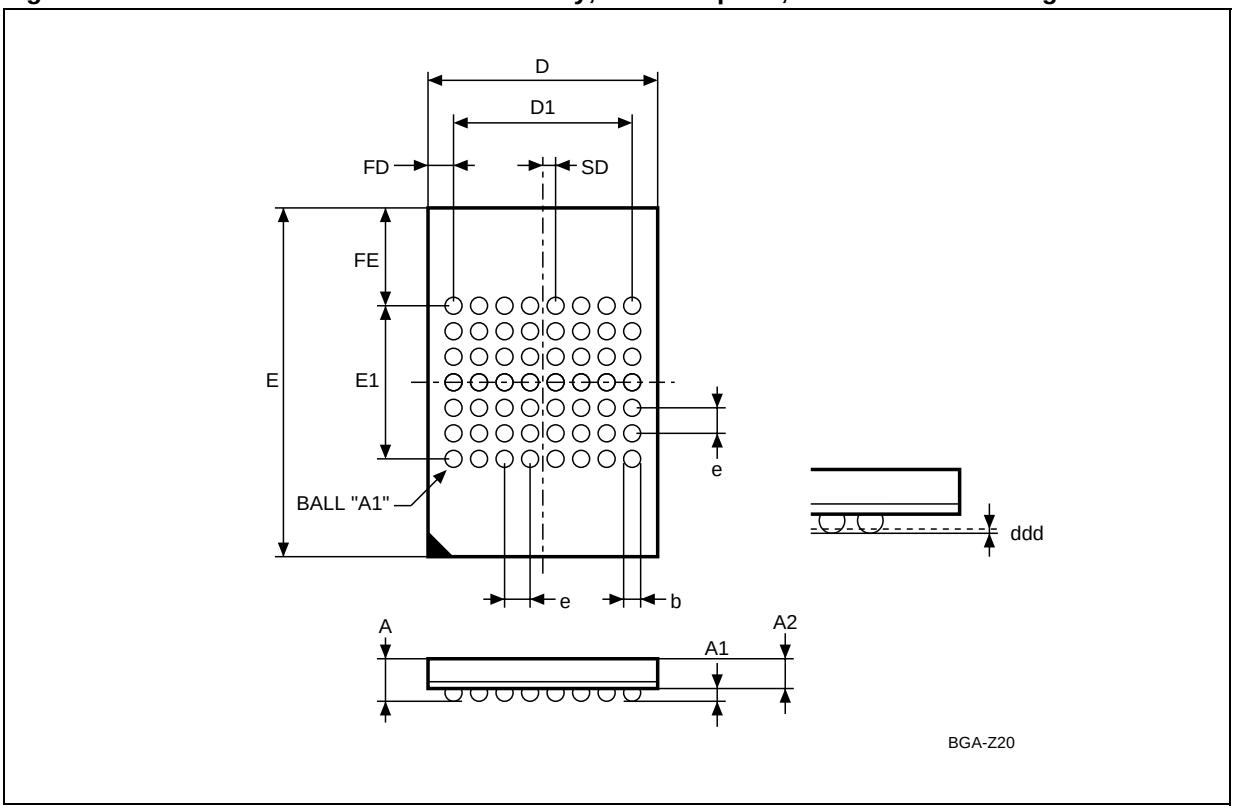

**Table 24. Reset and Power-up AC Characteristics**

| Symbol | Parameter | Test Condition | | 85 | 100 | 120 | 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 | 10 | 10 | µs |
| | | During Erase | Min | 20 | 20 | 20 | µs |
| | | Other Conditions | Min | 80 | 80 | 80 | ns |
| $t_{PLPH}$ (1,2) | $\overline{RP}$ Pulse Width | | Min | 50 | 50 | 50 | ns |
| $t_{VDHPH}$ (3) | Supply Voltages High to Reset High | | Min | 50 | 50 | 50 | µs |

Note: 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 $\overline{RP}$ in order to allow proper CPU initialization during Power-Up or Reset.

## PACKAGE MECHANICAL

**Figure 17. TFBGA56 6.5x10mm - 8x7 ball array, 0.75 mm pitch, Bottom View Package Outline**




Note: Drawing is not to scale.

**Table 25. TFBGA56 6.5x10mm - 8x7 ball array, 0.75 mm pitch, Package Mechanical Data**

| Symbol | millimeters | | | inches | | |
|---|---|---|---|---|---|---|
| | Typ | Min | Max | Typ | Min | Max |
| A | | 1.010 | 1.200 | | 0.0398 | 0.0472 |
| A1 | | 0.250 | 0.400 | | 0.0098 | 0.0157 |
| A2 | 0.790 | | | 0.0311 | | |
| b | 0.400 | 0.350 | 0.450 | 0.0157 | 0.0138 | 0.0177 |
| D | 6.500 | 6.400 | 6.600 | 0.2559 | 0.2520 | 0.2598 |
| D1 | 5.250 | – | – | 0.2067 | – | – |
| ddd | | | 0.100 | | | 0.0039 |
| E | 10.000 | 9.900 | 10.100 | 0.3937 | 0.3898 | 0.3976 |
| E1 | 4.500 | – | – | 0.1772 | – | – |
| e | 0.750 | – | – | 0.0295 | – | – |
| FD | 0.625 | – | – | 0.0246 | – | – |
| FE | 2.750 | – | – | 0.1083 | – | – |
| SD | 0.375 | – | – | 0.0148 | – | – |

## PART NUMBERING

### Table 26. Ordering Information Scheme

Example: M58CR064C 85 ZB 6 T

**Device Type**
M58

**Architecture**
C = Dual Bank, Burst Mode

**Operating Voltage**
R = $V_{DD}$ = 1.65V to 2.0V, $V_{DDQ}$ = 1.65V to 3.3V

**Device Function**
064C = 64 Mbit (x16), Dual Bank: 1/4-3/4 partitioning, Top Boot
064D = 64 Mbit (x16), Dual Bank: 1/4-3/4 partitioning, Bottom Boot
064P = 64 Mbit (x16), Dual Bank: 1/4-3/4 partitioning, Top Boot, $V_{PP}$ protection feature disabled
064Q = 64 Mbit (x16), Dual Bank: 1/4-3/4 partitioning, Bottom Boot, $V_{PP}$ protection feature disabled

**Speed**
85 = 85 ns
100 = 100 ns
120 = 120 ns

**Package**
ZB = TFBGA56: 6.5 x 10mm, 0.75 mm pitch

**Temperature Range**
6 = –40 to 85°C

**Option**
T = Tape & Reel packing

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

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

## REVISION HISTORY

**Table 27. Document Revision History**

| Date | Version | Revision Details |
|---|---|---|
| November 2000 | -01 | First Issue |
| 12/20/00 | -02 | Protection/Security clarification<br>Memory Map diagram clarification (Figure 4)<br>Single Synchronous Read clarification (Figure 6)<br>Identifier Codes clarification (Table 6)<br>X-Latency configuration clarification<br>CFI Query Identification String change (Table 31)<br>Synchronous Burst Read Waveforms change (Figure 12)<br>Reset AC Characteristics clarification (Table 24)<br>Program Time clarification (Table ) |
| 1/08/01 | -03 | Reset AC Characteristics clarification (Table 24)<br>Reset AC Waveforms diagram change (Figure 1) |
| 3/02/01 | -04 | Document type: from Target Specification to Product Preview<br>Read Status Register clarification<br>Read Electronic Signature clarification<br>Protection Register Program clarification<br>Write Configuration Register clarification<br>Wait Configuration Sequence change (Figure 7)<br>CFI Query System Interface clarification (Table 32)<br>CFI Device Geometry change (Table 33)<br>Asynchronous Read AC Waveforms change (Figure 10)<br>Page Read AC Waveforms added (Figure 11)<br>Write AC Waveforms W Contr. and E Contr. change (Figure 14, 15)<br>Reset and Power-up AC Characteristics and Waveform change (Table 24, Figure 1)<br>TFBGA Package Mechanical Data and Outline added (Table 25, Figure 17) |
| 4/05/01 | -05 | TFBGA Connections change<br>X-Latency Configuration Sequence change<br>Reset and Power-up AC Characteristics clarification<br>$V_{DDQ}$ clarification |
| 23-Jul-2001 | -06 | Complete rewrite and restructure |
| 23-Oct-2001 | -07 | 85ns speed class added, document classified as Preliminary Data |
| 15-Mar-2002 | -08 | Part numbers M58CR064P/Q added. CFI information clarified: Table 31,data modified at Offset 13h. Table 32, data modified at Offsets 20h, 23h, 24h and 25h. Table 35, Offset addresses modified. DC Characteristics table modified, Program, Erase Times and Program, Erase Endurance Cycles table modified. |
| 23-May-2002 | -09 | Document changed to new structure |
| 27-Aug-2002 | 9.1 | Revision numbering modified: a minor revision will be indicated by incrementing the digit after the dot, and a major revision, by incrementing the digit before the dot. (revision version 09 equals 9.0).<br>Document status changed from Preliminary Data to Datasheet.<br>Minimum $V_{DD}$ and $V_{DDQ}$ supply voltages for 85ns speed class changed to 1.8V in Table16, Operating and AC Measurement Conditions. |

## APPENDIX A. BLOCK ADDRESS TABLES

**Table 28. Top Boot Block Addresses, M58CR064C, M58CR064P**

| Bank | # | Size (KWord) | Address Range |
|---|---|---|---|
| Bank A | 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 |
| | 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 |
| | 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 |
| | 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 B | 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 |
| | 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 |
| | 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 |
| | 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 |
| | 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 B | 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 |
| | 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 |
| | 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 |
| | 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 |

| | | | |
|---|---|---|---|
| Bank B | 110 | 32 | 0C0000-0C7FFF |
| | 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 |
| | 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 |
| | 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 |

**Table 29. Bottom Boot Block Addresses, M58CR064D, M58CR064Q**

| Bank | # | Size (KWord) | Address Range |
|---|---|---|---|
| Bank B | 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 |
| | 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 |
| | 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 |
| | 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 |
| | 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 |
| Bank B | 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 |
| | 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 |
| | 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 |
| | 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 |
| | 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 B | 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 |
| | 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 A | 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 |
| | 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 A | 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 |
| | 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 |

## 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 30, 31, 32, 33, 34 and 35 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 Table , Security Code area). This area can be accessed only in Read mode by the final user. It is impossible to change the security number after it has been written by ST. Issue a Read Array command to return to Read mode.

**Table 30. Query Structure Overview**

| 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<br>Unique device Number and<br>User Programmable OTP |

Note: 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 31, 32, 33, 34 and 35. Query data is always presented on the lowest order data outputs.

**Table 31. CFI Query Identification String**

| Offset | Sub-section Name | Description | Value |
|---|---|---|---|
| 00h | 0020h | Manufacturer Code | ST |
| 01h | 88CAh<br>88CBh<br>8801h<br>8802h | Device Code (M58CR064C/D/P/Q) | Top<br>Bottom |
| 02h | reserved | Reserved | |
| 03h | reserved | Reserved | |
| 04h-0Fh | reserved | Reserved | |
| 10h | 0051h | Query Unique ASCII String "QRY" | "Q" |
| 11h | 0052h | | "R" |
| 12h | 0059h | | "Y" |
| 13h | 0003h | Primary Algorithm Command Set and Control Interface ID code 16 bit ID code defining a specific algorithm | |
| 14h | 0000h | | |
| 15h | offset = P = 0039h | Address for Primary Algorithm extended Query table (see Table 33) | p = 39h |
| 16h | 0000h | | |
| 17h | 0000h | Alternate Vendor Command Set and Control Interface ID Code second vendor - specified algorithm supported | NA |
| 18h | 0000h | | |

| Offset | Sub-section Name | Description | Value |
|---|---|---|---|
| 19h<br>1Ah | value = A = 0000h<br>0000h | Address for Alternate Algorithm extended Query table | NA |

**Table 32. CFI Query System Interface Information**

| Offset | Data | Description | Value |
|---|---|---|---|
| 1Bh | 0017h | $V_{DD}$ Logic Supply Minimum Program/Erase or Write voltage<br>bit 7 to 4 BCD value in volts<br>bit 3 to 0 BCD value in 100 millivolts | 1.7V |
| 1Ch | 0020h | $V_{DD}$ Logic Supply Maximum Program/Erase or Write voltage<br>bit 7 to 4 BCD value in volts<br>bit 3 to 0 BCD value in 100 millivolts | 2.0V |
| 1Dh | 0017h | $V_{PP}$ [Programming] Supply Minimum Program/Erase voltage<br>bit 7 to 4 HEX value in volts<br>bit 3 to 0 BCD value in 100 millivolts | 1.7V |
| 1Eh | 00C0h | $V_{PP}$ [Programming] Supply Maximum Program/Erase voltage<br>bit 7 to 4 HEX value in volts<br>bit 3 to 0 BCD value in 100 millivolts | 12V |
| 1Fh | 0004h | Typical time-out per single byte/word program = $2^n$ µs | 16µs |
| 20h | 0003h | Typical time-out for quadruple word program = $2^n$ µs | 8µs |
| 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 | 0004h | Maximum time-out for quadruple word = $2^n$ times typical | 128µs |
| 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 33. Device Geometry Definition**

| Offset Word Mode | Data | Description | Value |
|---|---|---|---|
| 27h | 0017h | Device Size = $2^n$ in number of bytes | 8 MByte |
| 28h<br>29h | 0001h<br>0000h | Flash Device Interface Code description | x16<br>Async. |
| 2Ah<br>2Bh | 0003h<br>0000h | Maximum number of bytes in multi-byte program or page = $2^n$ | 8 Byte |
| 2Ch | 0002h | Number of identical sized erase block regions within the device<br>bit 7 to 0 = x = number of Erase Block Regions | 2 |

| Offset Word Mode | | Data | Description | Value |
|---|---|---|---|---|
| M58CR064C/P | 2Dh<br>2Eh | 007Eh<br>0000h | Region 1 Information<br>Number of identical-size erase blocks = 007Eh+1 | 127 |
| | 2Fh<br>30h | 0000h<br>0001h | Region 1 Information<br>Block size in Region 1 = 0100h * 256 byte | 64 KByte |
| | 31h<br>32h | 0007h<br>0000h | Region 2 Information<br>Number of identical-size erase blocks = 000Eh+1 | 8 |
| | 33h<br>34h | 0020h<br>0000h | Region 2 Information<br>Block size in Region 2 = 0020h * 256 byte | 8 KByte |
| | 35h<br>38h | 0000h | Reserved for future erase block region information | NA |
| M58CR064D/Q | 2Dh<br>2Eh | 0007h<br>0000h | Region 1 Information<br>Number of identical-size erase block = 0007h+1 | 8 |
| | 2Fh<br>30h | 0020h<br>0000h | Region 1 Information<br>Block size in Region 1 = 0020h * 256 byte | 8 KByte |
| | 31h<br>32h | 007Eh<br>0000h | Region 2 Information<br>Number of identical-size erase block = 007Eh+1 | 127 |
| | 33h<br>34h | 0000h<br>0001h | Region 2 Information<br>Block size in Region 2 = 0100h * 256 byte | 64 KByte |
| | 35h<br>38h | 0000h | Reserved for future erase block region information | NA |

**Table 34. 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 | 0030h | Minor version number, ASCII | "0" |
| (P+5)h = 3Eh | 00E6h | Extended Query table contents for Primary Algorithm. Address (P+5)h contains less significant byte. | |
| | 0003h | | |
| (P+7)h = 40h | 0000h | bit 0 Chip Erase supported (1 = Yes, 0 = No) | No |
| | | bit 1 Erase Suspend supported (1 = Yes, 0 = No) | Yes |
| (P+8)h = 41h | 0000h | bit 2 Program Suspend supported (1 = Yes, 0 = No) | Yes |
| | | bit 3 Legacy Lock/Unlock supported (1 = Yes, 0 = No) | No |
| | | bit 4 Queued Erase supported (1 = Yes, 0 = No) | No |
| | | bit 5 Instant individual block locking supported (1 = Yes, 0 = No) | Yes |
| | | bit 6 Protection bits supported (1 = Yes, 0 = No) | Yes |
| | | bit 7 Page mode read supported (1 = Yes, 0 = No) | Yes |
| | | bit 8 Synchronous read supported (1 = Yes, 0 = No) | Yes |
| | | bit 9 Simultaneous operation supported (1 = Yes, 0 = No) | Yes |
| | | 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. | |

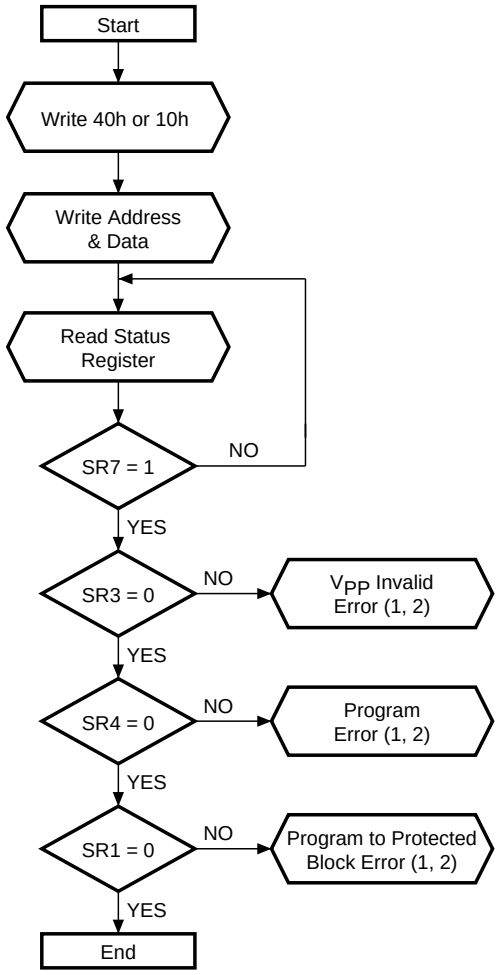
| Offset | Data | Description | Value |
|---|---|---|---|
| (P+9)h = 42h | 0001h | Supported Functions after Suspend<br>Read Array, Read Status Register and CFI Query<br>bit 0 Program supported after Erase Suspend (1 = Yes, 0 = No)<br>bit 7 to 1 Reserved; undefined bits are '0' | Yes |
| (P+A)h = 43h<br>(P+B)h = 44h | 0003h<br>0000h | Block Protect Status<br>Defines which bits in the Block Status Register section of the Query are implemented.<br>bit 0 Block protect Status Register Lock/Unlock bit active (1 = Yes, 0 = No)<br>bit 1 Block Lock Status Register Lock-Down bit active (1 = Yes, 0 = No)<br>bit 15 to 2 Reserved for future use; undefined bits are '0' | <br><br><br><br>Yes<br>Yes |
| (P+C)h = 45h | 0018h | $V_{DD}$ Logic Supply Optimum Program/Erase voltage (highest performance)<br>bit 7 to 4HEX value in volts<br>bit 3 to 0BCD value in 100 mV | 1.8V |
| (P+D)h = 46h | 00C0h | $V_{PP}$ Supply Optimum Program/Erase voltage<br>bit 7 to 4HEX value in volts<br>bit 3 to 0BCD value in 100 mV | 12V |
| (P+E)h = 47h<br>(P+F)h = 48h<br>(P+10)h = 49h<br>(P+11)h = 4Ah<br>(P+12)h = 4Bh | 0000h | Reserved | |

**Table 35. Burst Read Information**

| Offset | Data | Description | Value |
|---|---|---|---|
| (P+13)h = 4Ch | 0003h | Page-mode read capability<br>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 | 0003h | Number of synchronous mode read configuration fields that follow. | 3 |
| (P+15)h = 4Eh | 0001h | Synchronous mode read capability configuration 1<br>bit 3-7 Reserved<br>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 | 0007h | Synchronous mode read capability configuration 3 | Cont. |
| (P+18)h = 51h | 0036h | Max operating clock frequency (MHz) | 54 MHz |
| (P+19)h = 52h | 0001h | Supported handshaking signal ($\overline{WAIT}$ pin)<br>bit 0 during synchronous read (1 = Yes, 0 = No)<br>bit 1 during asynchronous read (1 = Yes, 0 = No) | <br>Yes<br>No |

## APPENDIX C. FLOWCHARTS AND PSEUDO CODES

### Figure 18. Program Flowchart and Pseudo Code

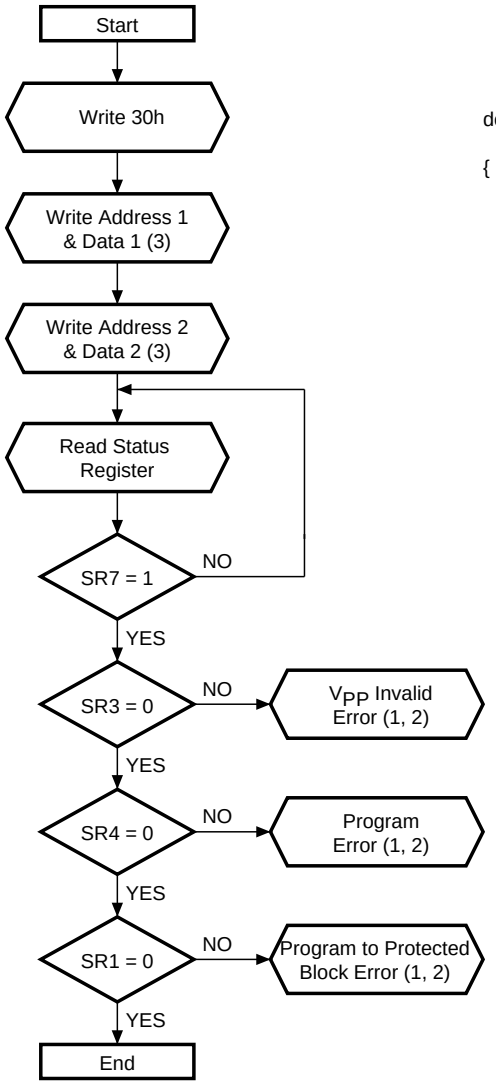



```
program_command (addressToProgram, dataToProgram) {:
        writeToFlash (bank_address, 0x40) ;
        /*or writeToFlash (bank_address, 0x10) ; */

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

      do {
        status_register=readFlash (bank_address) ;
        /* 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 ( ) ;

}
```

AI06170

Note: 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.

**Figure 19. Double Word Program Flowchart and Pseudo code**

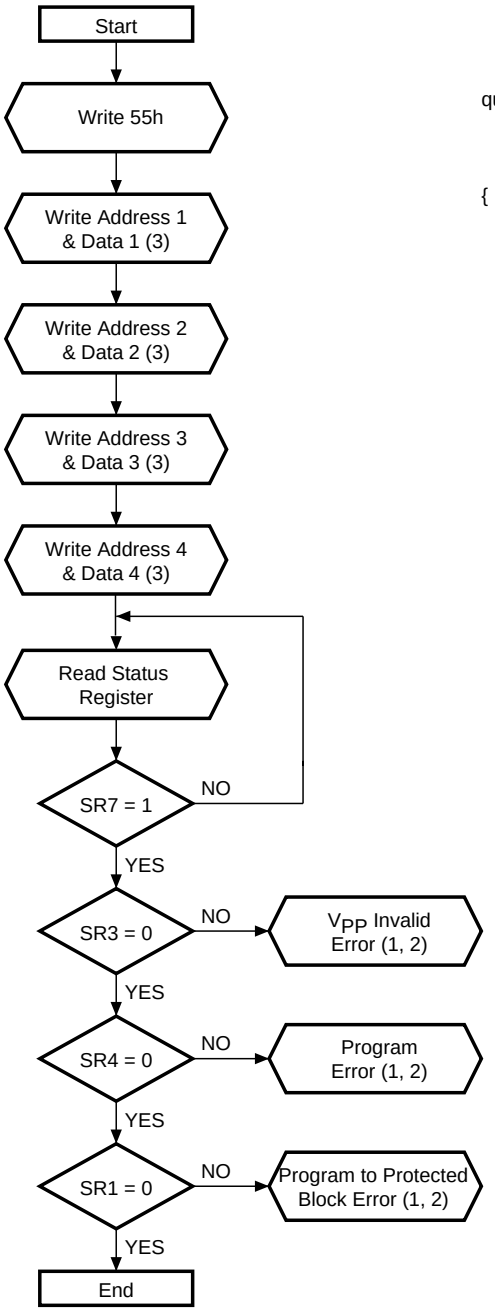



```
double_word_program_command (addressToProgram1, dataToProgram1,
                                        addressToProgram2, dataToProgram2)
{
        writeToFlash (bank_address, 0x30) ;
        writeToFlash (addressToProgram1, dataToProgram1) ;
                                        /*see note (3) */
        writeToFlash (addressToProgram2, dataToProgram2) ;
                                        /*see note (3) */
         /*Memory enters read status state after
        the Program command*/

      do {
        status_register=readFlash (bank_address) ;
        /* E or G must be toggled*/

        } while (status_register.b7== 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 ( ) ;

}
```

AI06171

Note: 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.

**Figure 20. Quadruple Word Program Flowchart and Pseudo Code**

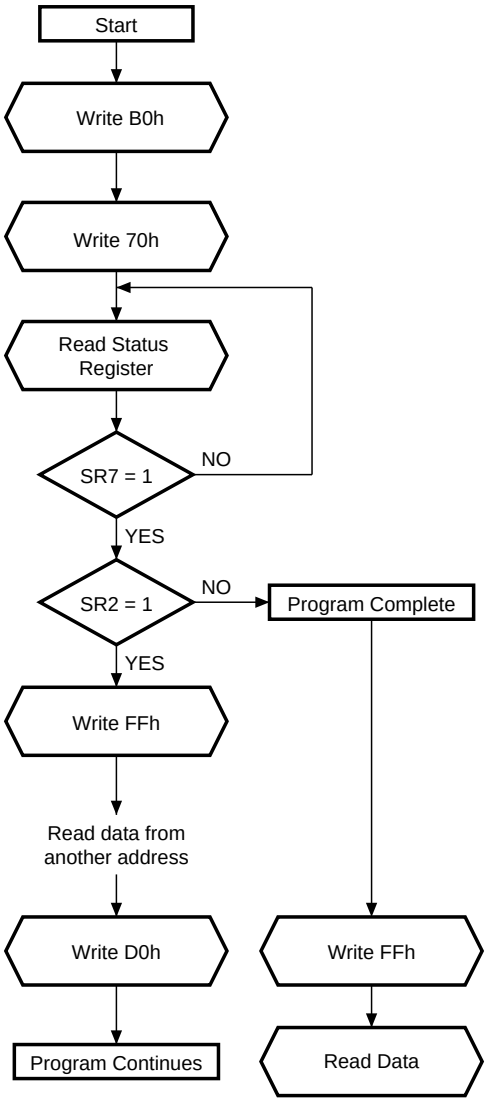



```
quadruple_word_program_command (addressToProgram1, dataToProgram1,
                                addressToProgram2, dataToProgram2,
                                addressToProgram3, dataToProgram3,
                                addressToProgram4, dataToProgram4)
{
        writeToFlash (bank_address, 0x55) ;

        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 (bank_address) ;
        /* E or G must be toggled*/

        } while (status_register.b7== 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 ( ) ;

}
```

AI06172

Note: 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.

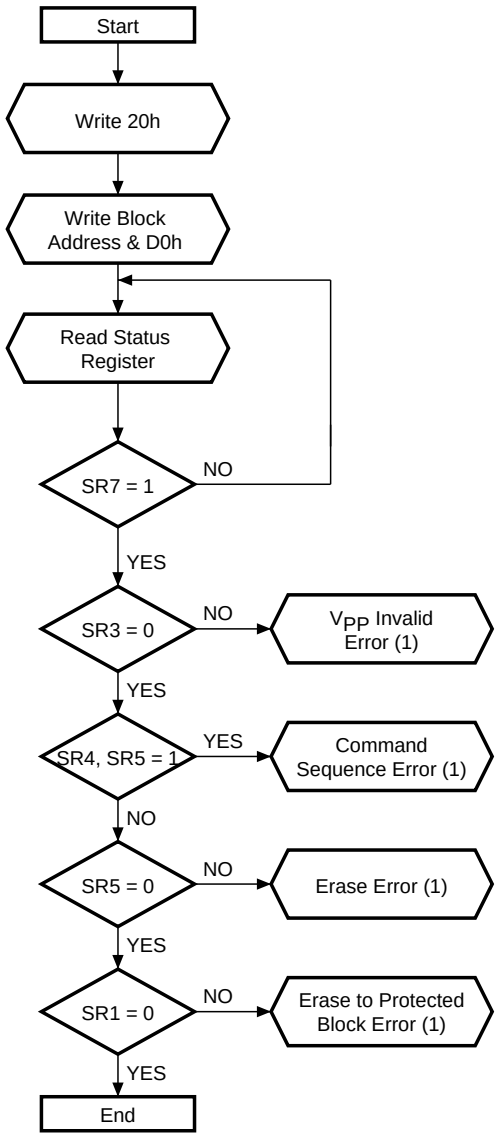
**Figure 21. Program Suspend & Resume Flowchart and Pseudo Code**




```
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) ;
      /* E or G must be toggled*/




      } while (status_register.SR7== 0) ;



if  (status_register.SR2==0)  /*program completed */
      {  writeToFlash (bank_address, 0xFF)  ;
        read_data ( ) ; /*read data from another block*/
        /*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*/
       }
}
```

AI06173

**Figure 22. Block Erase Flowchart and Pseudo Code**

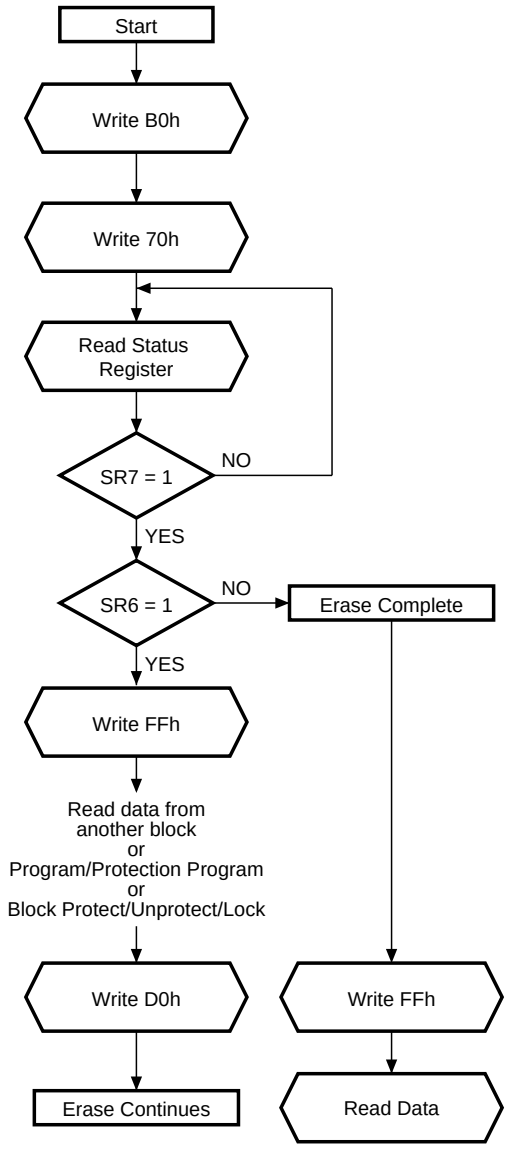



```
erase_command   ( blockToErase )  {
        writeToFlash (bank_address, 0x20) ;

        writeToFlash (blockToErase, 0xD0) ;
        /* only A12-A20 are significannt */
        /* Memory enters read status state after
        the Erase Command */


    do {
        status_register=readFlash (bank_address) ;
         /* 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) && (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 ( ) ;


}
```

AI06174

Note: If an error is found, the Status Register must be cleared before further Program/Erase operations.

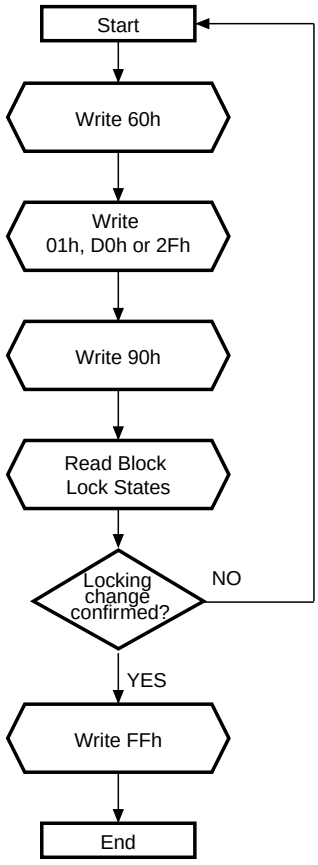
**Figure 23. Erase Suspend & Resume Flowchart and Pseudo Code**




```
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) ;
        /* E or G must be toggled*/


        } while (status_register.SR7== 0) ;


if  (status_register.SR6==0)  /*erase completed */
        {  writeToFlash (bank_address, 0xFF)  ;


          read_data ( ) ;
          /*read data from another block*/
          /*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 address*/
          writeToFlash (bank_address, 0xD0) ;
          /*write 0xD0 to resume erase*/
        }
}
```

AI06175

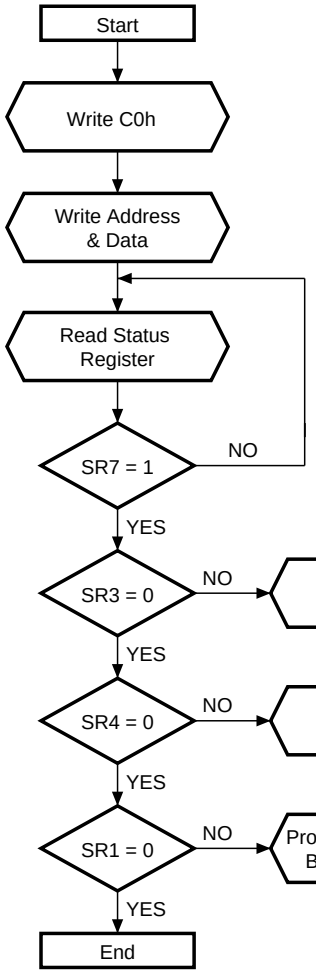
**Figure 24. Locking Operations Flowchart and Pseudo Code**




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

        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 (bank_address, 0x90)  ;

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

        writeToFlash (bank_address, 0xFF) ; /*Reset to Read Array mode*/

}
```

AI06176

**Figure 25. Protection Register Program Flowchart and Pseudo Code**




```
protection_register_program_command (addressToProgram, dataToProgram) {:
                    writeToFlash (bank_address, 0xC0) ;

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

                do {
                    status_register=readFlash (bank_address) ;
                    /* 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 ( ) ;

}
```

AI06177

Note: 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.

## APPENDIX D. COMMAND INTERFACE STATE TABLES

### Table 36. Command Interface States - Lock table

| Current State of Other Bank | Current State of the Current Bank | | Command Input to the Current Bank (and Next State of the Current Bank) | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Mode | State | Others | Read Array (FFH) | Erase Confirm P/E Resume Unlock Confirm (D0h) | Read Status Register (70h) | Clear Status Register (50h) | Read Electronic Signature (90h) | Read CFI Query (98h) | Block Lock Unlock Lock-Down setup Set CR setup (60h) | Block lock Confirm (01h) | Block Lock-Down Confirm (2Fh) | Set CR Confirm (03h) |
| Any State | Read | Array<br>CFI<br>Electronic Signature<br>Status | SEE MODIFY TABLE | Read Array | Read Array | Read Status Register | Read Array | Read Elect. Sign. | Read CFI | Block Lock, Unlock, Lock-Down, Set CR Setup | Read Array | Read Array | Read Array |
| Any State | Lock Unlock Lock-Down CR | Setup | Block Lock Unlock Lock-Down Error, Set CR Error | Block lock Unlock Lock-Down Error, Set CR Error | Block Lock Unlock Lock-Down Block | Block Lock Unlock Lock-Down Error, Set CR Error | Block Lock Unlock Lock-Down Error, Set CR Error | Block Lock Unlock Lock-Down Error, Set CR Error | Block Lock Unlock Lock-Down Error, Set CR Error | Block Lock Unlock Lock-Down Error, Set CR Error | Block Lock Unlock Lock-Down Block | Block Lock Unlock Lock-Down Block | Set CR |
| | | Error<br>Lock Unlock Lock-Down Block<br>Set CR | SEE MODIFY TABLE | Read Array | Read Array | Read Status Register | Read Array | Read Elect. Sign. | Read CFI | Block LocK Unlock Lock-Down Setup, Set CR Setup | Read Array | Read Array | Read Array |
| Any State | Protection Register | Done | SEE MODIFY TABLE | Read Array | Read Array | Read Status Register | Read Array | Read Elect. Sign. | Read CFI | Block LocK Unlock Lock-Down Setup, Set CR Setup | Read Array | Read Array | Read Array |
| Any State | Program-Double/ Quadruple Program | Done | SEE MODIFY TABLE | Read Array | Read Array | Read Status Register | Read Array | Read Elect. Sign. | Read CFI | Block LocK Unlock Lock-Down Setup, Set CR Setup | Read Array | Read Array | Read Array |
| Setup<br>Idle<br>Erase Suspend | Program Suspend | Read Array, CFI, Elect. Sign., Status | SEE MODIFY TABLE | PS Read Array | Program (Busy) | PS Read Status Register | PS Read Array | PS Read Elect. Sign. | PS Read CFI | PS Read Array | PS Read Array | PS Read Array | PS Read Array |
| Idle | Block/ Bank Erase | Setup | Erase Error | Erase Error | Erase (Busy) | Erase Error | Erase Error | Erase Error | Erase Error | Erase Error | Erase Error | Erase Error | Erase Error |
| Any State | | Error<br>Done | SEE MODIFY TABLE | Read Array | Read Array | Read Status Register | Read Array | Read Elect. Sign. | Read CFI | Block LocK Unlock Lock-Down Setup, Set CR Setup | Read Array | Read Array | Read Array |
| Setup<br>Busy<br>Idle<br>Program Suspend | Erase Suspend | Read Array, CFI, Elect. Sign., Status | SEE MODIFY TABLE | ES Read Array | Erase (Busy)<br>ES Read Array<br>Erase (Busy)<br>ES Read Array | ES Read Status Register | ES Read Array | ES Read Elect. Sign. | ES Read CFI | Block LocK Unlock Lock-Down Setup, Set CR Setup | ES Read Array | ES Read Array | ES Read Array |

Note: PS = Program Suspend, ES = Erase Suspend.

**Table 37. Command Interface States - Modify Table**

| Current State of the Other Bank | Current State of the Current Bank | | Command Input to the Current Bank (and Next State of the Current Bank) | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Mode | State | Others | Program Setup (10h/40h) | Block Erase Setup (20h) | Program-Erase Suspend (B0h) | Protection Register Program Setup (C0h) | Double/ Quadruple Program Setup (30h/55h) | Bank Erase Setup (80h) |
| Setup | Read | Array, CFI, Electronic Signature, Status Register | SEE LOCK TABLE | Read Array | Read Array | Read Array | Read Array | Read Array | Read Array |
| Busy | | | | | | | | | |
| Idle | | | | Program setup | Block Erase Setup | | Protection Register Setup | Double/ Quadruple Program Setup | Bank Erase Setup |
| Erase Suspend | | | | | Read Array | | Read Array | | Read Array |
| Program Suspend | | | | Read Array | | | | Read Array | |
| Setup | Lock Unlock Lock-Down CR | Error, Lock Unlock Lock-Down Block, Set CR | SEE LOCK TABLE | Read Array | Read Array | Read Array | Read Array | Read Array | Read Array |
| Busy | | | | | | | | | |
| Idle | | | | Program setup | Block Erase Setup | | Protection Register Setup | Double/ Quadruple Program Setup | Bank Erase Setup |
| Erase Suspend | | | | | Read Array | | Read Array | | Read Array |
| Program Suspend | | | | Read Array | | | | Read Array | |
| Idle | Protection Register | Setup | Protection Register (Busy) | Protection Register (Busy) | Protection Register (Busy) | Protection Register (Busy) | Protection Register (Busy) | Protection Register (Busy) | Protection Register (Busy) |
| Setup | | Busy | | | | | | | |
| Busy | | Done | SEE LOCK TABLE | Read Array | Read Array | Read Array | Read Array | Read Array | Read Array |
| Idle | | | | Program Setup | Block Erase Setup | | Protection Register Setup | Double/ Quadruple Program Setup | Bank Erase Setup |
| Erase Suspend | | | | | Read Array | | Read Array | | Read Array |
| Program Suspend | | | | Read Array | | | | Read Array | |
| Any State | Program Double/ Quadruple Word Program | Setup | Program (Busy) | Program (Busy) | Program (Busy) | Program (Busy) | Program (Busy) | Program (Busy) | Program (Busy) |
| Idle | | Busy | | | | PS Read Status Register | | | |
| Setup | | Done | SEE LOCK TABLE | Read Array | Read Array | Read Array | Read Array | Read Array | Read Array |
| Busy | | | | | | | | | |
| Idle | | | | Program Setup | Block Erase Setup | | Protection Register Setup | Double/ Quadruple Program Setup | Bank Erase Setup |
| Erase Suspend | | | | | Read Array | | Read Array | | Read Array |
| Program Suspend | | | | Read Array | | | | Read Array | |
| Setup | Program Suspend | Read Array, CFI, Elect. Sign., Status Register | SEE LOCK TABLE | PS Read Array | PS Read Array | PS Read Array | PS Read Array | PS Read Array | PS Read Array |
| Idle | | | | | | | | | |
| Erase Suspend | | | | | | | | | |
| Idle | Block/ Bank Erase | Setup | SEE LOCK TABLE | Erase Error | Erase Error | Erase Error | Erase Error | Erase Error | Erase Error |
| | | Busy | Erase (Busy) | Erase (Busy) | Erase (Busy) | ES Read Status Register | Erase (Busy) | Erase (Busy) | Erase (Busy) |
| Setup | Erase Suspend | Read Array, CFI, Elect. Sign., Status Register | SEE LOCK TABLE | ES Read Array | ES Read Array | ES Read Array | ES Read Array | ES Read Array | ES Read Array |
| Busy | | | | | | | | | |
| Idle | | | | Program Setup | | | | Double/ Quadruple Program Setup | |
| Program Suspend | | | | ES Read Array | | | | ES Read Array | |

Note: PS = Program Suspend, ES = Erase Suspend.

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is registered trademark of STMicroelectronics

All other names are the property of their respective owners

© 2002 STMicroelectronics - All Rights Reserved

STMicroelectronics group of companies

Australia - Brazil - Canada - China - Finland - France - Germany - Hong Kong -

India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States.

**www.st.com**