Numonyx^{TM} StrataFlash$^{®}$ Embedded Memory (P33-65nm)

256-Mbit, 512-Mbit (256M/256M)

Datasheet

Product Features

- High performance:
  - — 95ns initial access time for Easy BGA
  - — 105ns initial access time for TSOP
  - — 25ns 16-word asynchronous-page read mode
  - — 52MHz with zero wait states, 17ns clock-to-data output synchronous-burst read mode
  - — 4-, 8-, 16-, and continuous-word options for burst mode
  - — Buffered Enhanced Factory Programming at 2.0MByte/s (typ) using 512-word buffer
  - — 3.0V buffered programming at 1.5MByte/s (Typ) using 512-word buffer
- Architecture:
  - — Multi-Level Cell Technology: Highest Density at Lowest Cost
  - — Asymmetrically-blocked architecture
  - — Four 32-KByte parameter blocks: top or bottom configuration
  - — 128-KByte main blocks
  - — Blank Check to verify an erase block
- Voltage and Power:
  - — V_{CC} (core) voltage: 2.3 V – 3.6 V
  - — V_{CCQ} (I/O) voltage: 2.3 V – 3.6 V
  - — Standby current: 65uA (Typ) for 256-Mbit
  - — Continuous synchronous read current: 21 mA (Typ)/24 mA (Max) at 52 MHz
- Security:
  - — One-Time Programmable Registers:
    - — 64 unique factory device identifier bits
    - — 2112 user-programmable OTP bits
  - — Absolute write protection: $V_{PP} = V_{SS}$
  - — Power-transition erase/program lockout
  - — Individual zero-latency block locking
  - — Individual block lock-down capability
  - — Password Access feature
- Software:
  - — 20µs (Typ) program suspend
  - — 20µs (Typ) erase suspend
  - — Numonyx™ Flash Data Integrator optimized
  - — Basic Command Set and Extended Function Interface Command Set compatible
  - — Common Flash Interface capable
- Density and Packaging
  - — 56-Lead TSOP package (256-Mbit only)
  - — 64-Ball Easy BGA package (256, 512-Mbit)
  - — 16-bit wide data bus
- Quality and Reliability
  - — Operating temperature: –40 °C to +85 °C
  - — Minimum 100,000 erase cycles per block
  - — 65nm ETOX™ X process technology

Apr 2009
Order Number: 320003-07

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH NUMONYX™ PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN NUMONYX'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, NUMONYX ASSUMES NO LIABILITY WHATSOEVER, AND NUMONYX DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF NUMONYX PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Numonyx products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

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

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

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

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

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

Numonyx, the Numonyx logo, and StrataFlash are trademarks or registered trademarks of Numonyx, B.V. or its subsidiaries in other countries.

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

Copyright © 2009, Numonyx, B.V., All Rights Reserved.

Contents

1.0 Functional Description

1.1 Introduction

This document provides information about the Numonyx™ StrataFlash® Embedded Memory (P33-65nm) device and describes its features, operations, and specifications.

P33-65nm is the latest generation of Numonyx™ StrataFlash® Embedded Memory (P33-65nm) devices. P33-65nm device will be offered in 64-Mbit up through 2-Gbit densities. This document covers specifically 256-Mbit and 512-Mbit (256M/256M) product information. Benefits include more density in less space, high-speed interface NOR device, and support for code and data storage. Features include high-performance synchronous-burst read mode, fast asynchronous access times, low power, flexible security options, and two industry-standard package choices.

P33-65nm is manufactured using Numonyx™ 65nm ETOX™ X process technology.

1.2 Overview

This family of devices provides high performance at low voltage on a 16-bit data bus. Individually erasable memory blocks are sized for optimum code and data storage.

Upon initial power-up or return from reset, the device defaults to asynchronous page-mode read. Configuring the RCR enables synchronous burst-mode reads. In synchronous burst mode, output data is synchronized with a user-supplied clock signal. A WAIT signal provides an easy CPU-to-flash memory synchronization.

In addition to the enhanced architecture and interface, the device incorporates technology that enables fast factory program and erase operations. Designed for low-voltage systems, the P33 Family Flash memory supports read operations with VCC at 3.0V, and erase and program operations with VPP at 3.0V or 9.0V. Buffered Enhanced Factory Programming provides the fastest flash array programming performance with VPP at 9.0V, which increases factory throughput. With VPP at 3.0V, VCC and VPP can be tied together for a simple, ultra low power design. In addition to voltage flexibility, a dedicated VPP connection provides complete data protection when VPP $\leq V_{PPLK}$.

The Command User Interface is the interface between the system processor and all internal operations of the device. An internal Write State Machine automatically executes the algorithms and timings necessary for block erase and program. A Status Register indicates erase or program completion and any errors that may have occurred.

An industry-standard command sequence invokes program and erase automation. Each erase operation erases one block. The Erase Suspend feature allows system software to pause an erase cycle to read or program data in another block. Program Suspend allows system software to pause programming to read other locations. Data is programmed in word increments (16 bits).

The P33 Family Flash memory one-time-programmable (OTP) register allows unique flash device identification that can be used to increase system security. The individual Block Lock feature provides zero-latency block locking and unlocking. The P33-65nm device adds enhanced protection via Password Access Mode which allows user to protect write and/or read access to the defined blocks. In addition, the P33 Family Flash memory may also provide the backward compatible OTP permanent lock feature.

1.3 Virtual Chip Enable Description

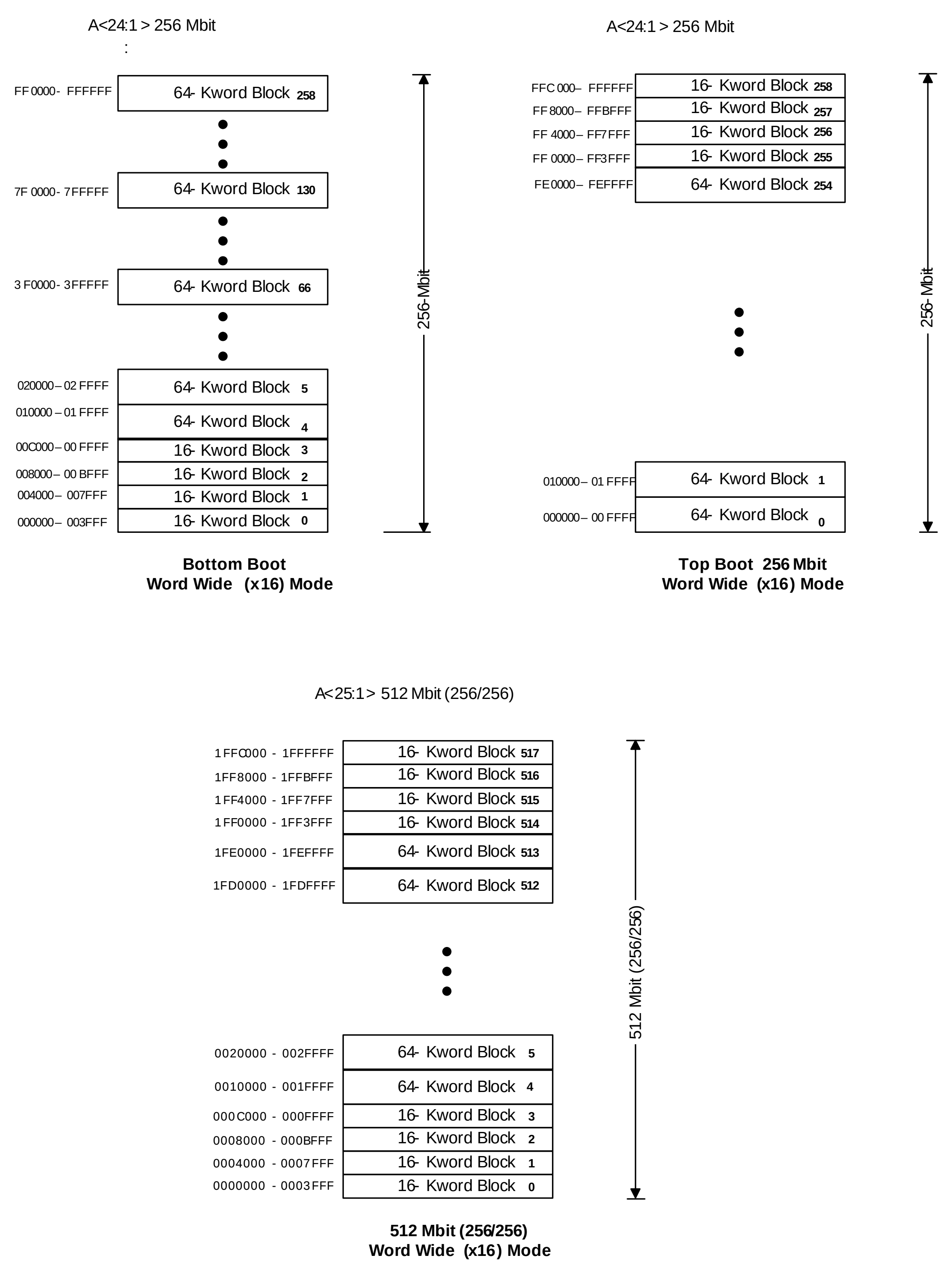
The 512 Mbit P33 Family Flash memory employs a Virtual Chip Enable which combines two 256-Mbit die with a common chip enable, CE# for Easy BGA packages. Address A25 is then used to select between the die pair with CE# asserted, depending upon the package option used. When chip enable is asserted and A25 is low (V_{IL}), The lower parameter die is selected; when chip enable is asserted and A25 is high (V_{IH}), the upper parameter die is selected.

Table 1: Flash Die Virtual Chip Enable Truth Table for 512 Mbit Easy BGA Package

Die Selected	CE#	A25
Lower Param Die	L	L
Upper Param Die	L	H

1.4 Memory Maps

Figure 1: P33-65nm Memory Map



2.0 Package Information

2.1 56-Lead TSOP

Figure 2: TSOP Mechanical Specifications (256-Mbit)

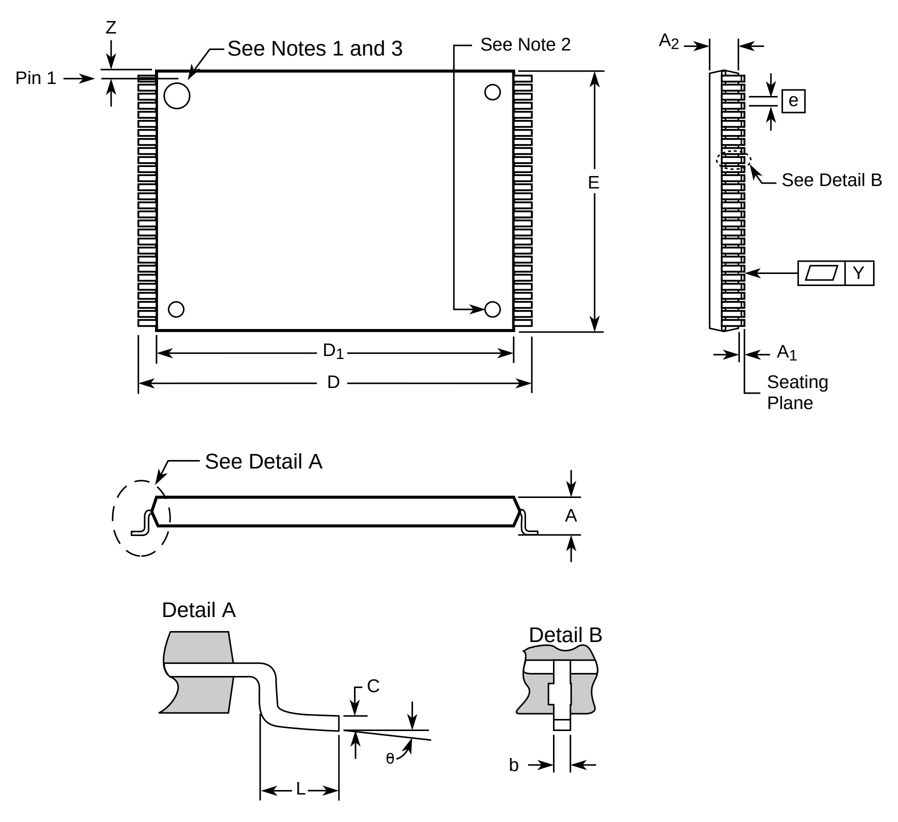


Table 2: TSOP Package Dimensions (Sheet 1 of 2)

Product Information	Symbol	Millimeters			Inches			Note
		Min	Nom	Max	Min	Nom	Max	
Package Height	A	-	-	1.200	-	-	0.047	
Standoff	A_1	0.050	-	-	0.002	-	-	
Package Body Thickness	A_2	0.965	0.995	1.025	0.038	0.039	0.040	
Lead Width	b	0.100	0.150	0.200	0.004	0.006	0.008	
Lead Thickness	c	0.100	0.150	0.200	0.004	0.006	0.008	
Package Body Length	D_1	18.200	18.400	18.600	0.717	0.724	0.732	Note
Package Body Width	E	13.800	14.000	14.200	0.543	0.551	0.559	Note
Lead Pitch	e	-	0.500	-	-	0.0197	-	
Terminal Dimension	D	19.800	20.00	20.200	0.780	0.787	0.795	
Lead Tip Length	L	0.500	0.600	0.700	0.020	0.024	0.028	

Table 2: TSOP Package Dimensions (Sheet 2 of 2)

Product Information	Symbol	Millimeters			Inches			Note
		Min	Nom	Max	Min	Nom	Max	
Lead Count	N	-	56	-	-	56	-	
Lead Tip Angle	θ	0°	3°	5°	0°	3°	5°	
Seating Plane Coplanarity	Y	-	-	0.100	-	-	0.004	
Lead to Package Offset	Z	0.150	0.250	0.350	0.006	0.010	0.014	

Notes:

1. One dimple on package denotes Pin 1.
2. If two dimples, then the larger dimple denotes Pin 1.
3. Pin 1 will always be in the upper left corner of the package, in reference to the product mark.

2.2 64-Ball Easy BGA Package

Figure 3: Easy BGA Mechanical Specifications (256/512-Mbit)

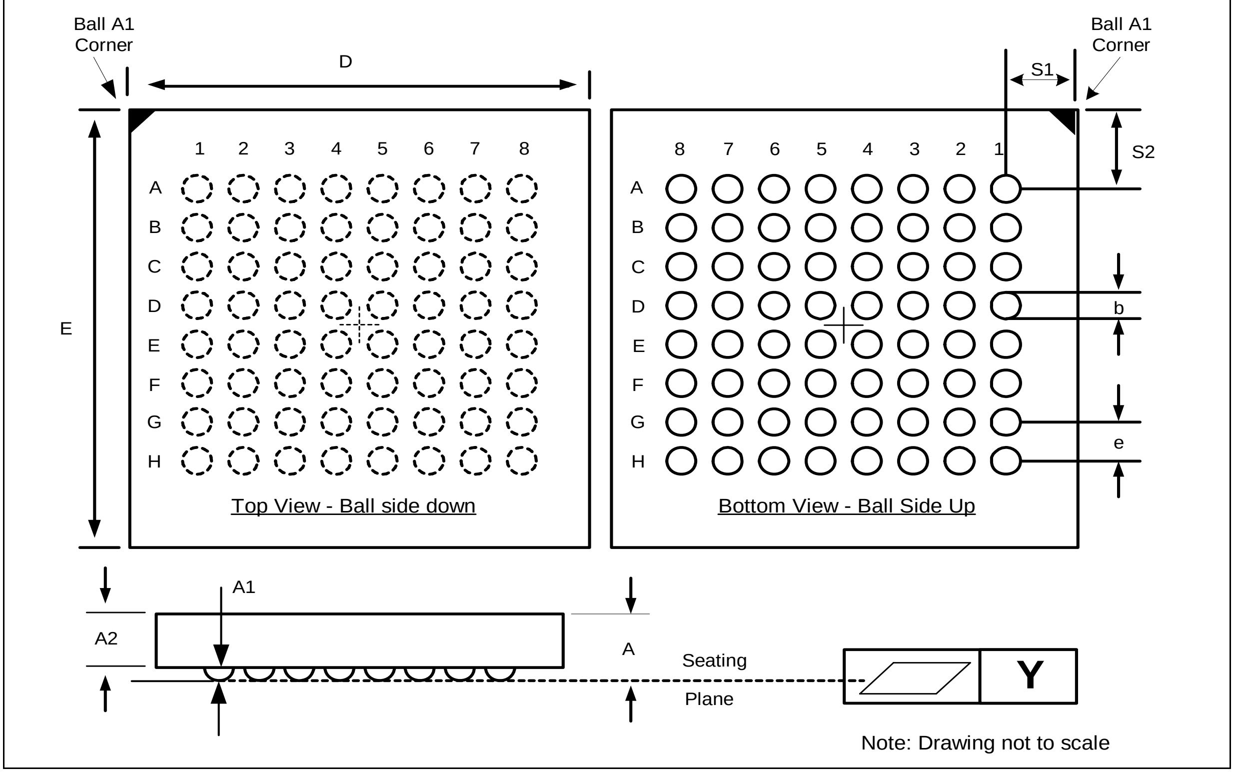


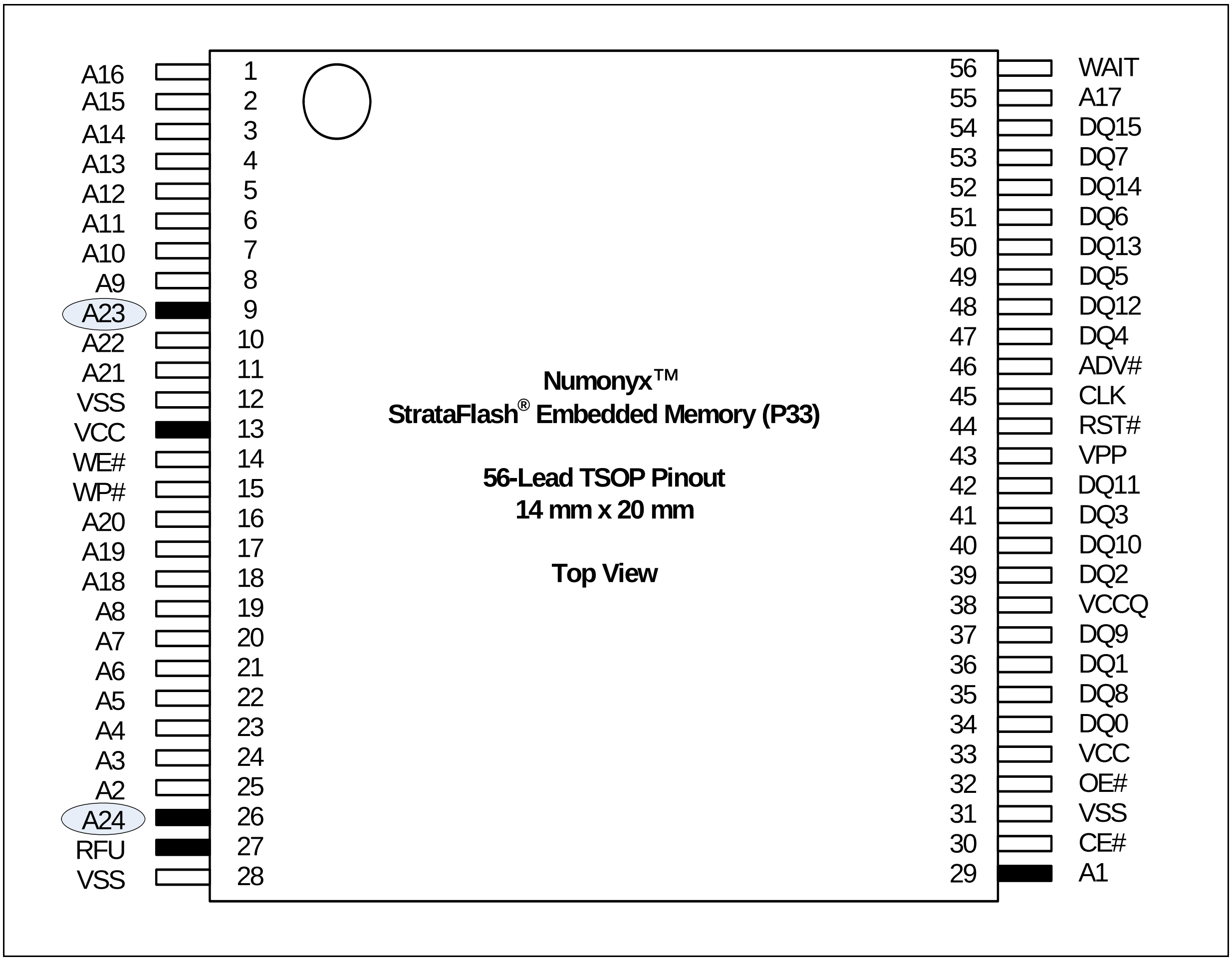
Table 3: Easy BGA Package Dimensions

Product Information	Symbol	Millimeters			Inches			Notes
		Min	Nom	Max	Min	Nom	Max	
Package Height	A	-	-	1.200	-	-	0.0472	
Ball Height	A1	0.250	-	-	0.0098	-	-	
Package Body Thickness	A2	-	0.780	-	-	0.0307	-	
Ball (Lead) Width	b	0.330	0.430	0.530	0.0130	0.0169	0.0209	
Package Body Width	D	9.900	10.000	10.100	0.3898	0.3937	0.3976	Note
Package Body Length	E	12.900	13.000	13.100	0.5079	0.5118	0.5157	Note
Pitch	[e]	-	1.000	-	-	0.0394	-	
Ball (Lead) Count	N	-	64	-	-	64	-	
Seating Plane Coplanarity	Y	-	-	0.100	-	-	0.0039	
Corner to Ball A1 Distance Along D	S1	1.400	1.500	1.600	0.0551	0.0591	0.0630	Note
Corner to Ball A1 Distance Along E	S2	2.900	3.000	3.100	0.1142	0.1181	0.1220	Note

Note: Daisy Chain Evaluation Unit information is at Numonyx™ Flash Memory Packaging Technology http://developer.numonyx.com/design/flash/packtech.

3.0 Ballouts

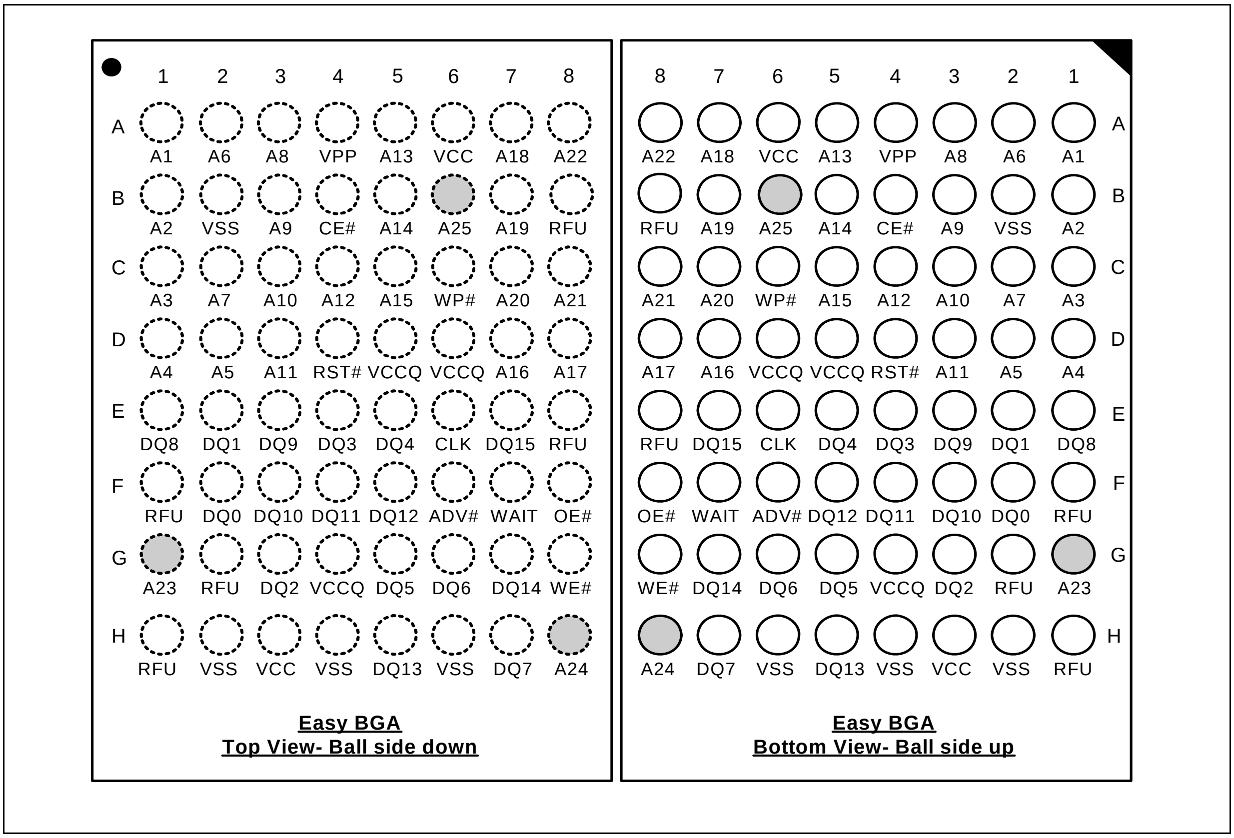
Figure 4: 56-Lead TSOP Pinout (256-Mbit)



Notes:

1. A1 is the least significant address bit.
2. A24 is valid for 256-Mbit densities; otherwise, it is a no connect (NC).
3. No Internal Connection on VCC Pin 13; it may be driven or floated. For legacy designs, pin can be tied to Vcc

Figure 5: 64-Ball Easy BGA Ballout (256/512-Mbit)



	1	2	3	4	5	6	7	8
A	A1	A6	A8	VPP	A13	VCC	A18	A22
B	A2	VSS	A9	CE#	A14	A25	A19	RFU
C	A3	A7	A10	A12	A15	WP#	A20	A21
D	A4	A5	A11	RST#	VCCQ	VCCQ	A16	A17
E	DQ8	DQ1	DQ9	DQ3	DQ4	CLK	DQ15	RFU
F	RFU	DQ0	DQ10	DQ11	DQ12	ADV#	WAIT	OE#
G	A23	RFU	DQ2	VCCQ	DQ5	DQ6	DQ14	WE#
H	RFU	VSS	VCC	VSS	DQ13	VSS	DQ7	A24

Easy BGA
Top View- Ball side down

8	7	6	5	4	3	2	1	
A22	A18	VCC	A13	VPP	A8	A6	A1	A
RFU	A19	A25	A14	CE#	A9	VSS	A2	B
A21	A20	WP#	A15	A12	A10	A7	A3	C
A17	A16	VCCQ	VCCQ	RST#	A11	A5	A4	D
RFU	DQ15	CLK	DQ4	DQ3	DQ9	DQ1	DQ8	E
OE#	WAIT	ADV#	DQ12	DQ11	DQ10	DQ0	RFU	F
WE#	DQ14	DQ6	DQ5	VCCQ	DQ2	RFU	A23	G
A24	DQ7	VSS	DQ13	VSS	VCC	VSS	RFU	H

Easy BGA
Bottom View- Ball side up

Notes:

1. A1 is the least significant address bit.
2. A24 is valid for 256-Mbit densities and above; otherwise, it is a no connect.
3. A25 is valid for 512-Mbit densities; otherwise, it is a no connect.

4.0 Signals

Table 4: TSOP and Easy BGA Signal Descriptions (Sheet 1 of 2)

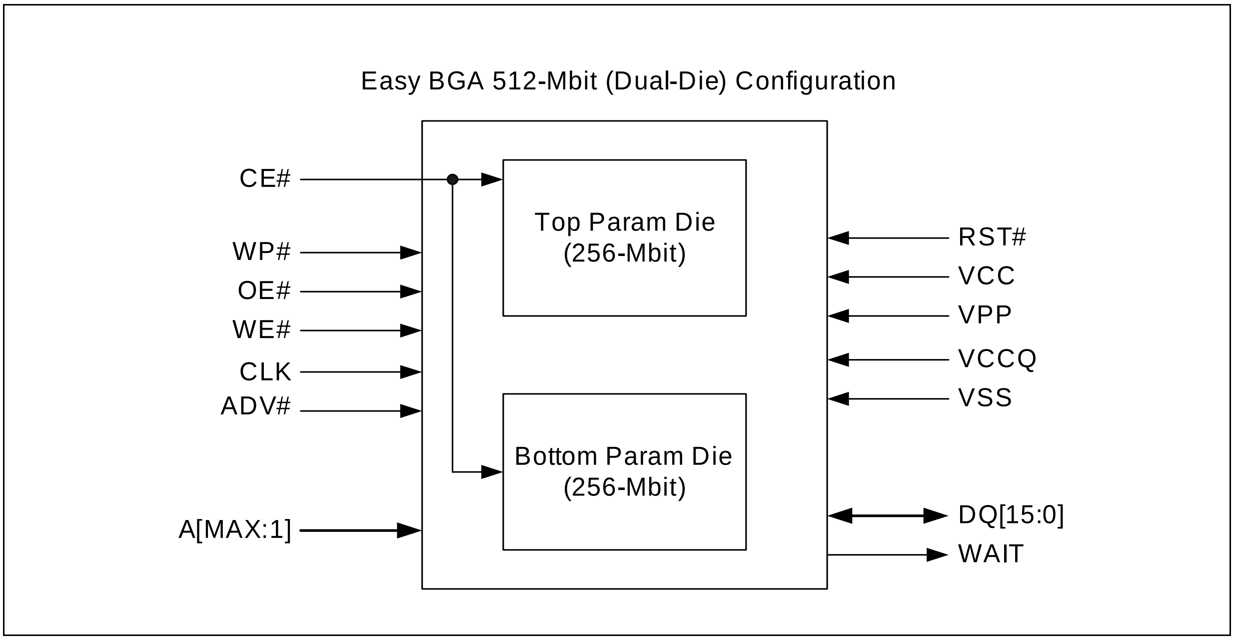
Symbol	Type	Name and Function
A[MAX:1]	Input	**ADDRESS INPUTS:** Device address inputs. 256-Mbit: A[24:1]; 512-Mbit: A[25:1]. *Note:* The virtual selection of the 256-Mbit “Top parameter” die in the dual-die 512-Mbit configuration is accomplished by setting A[25] high (V_{IH}).
DQ[15:0]	Input/ Output	**DATA INPUT/OUTPUTS:** Inputs data and commands during write cycles; outputs data during reads of memory, status register, OTP register, and read configuration register. Data balls float when the CE# or OE# are deasserted. Data is internally latched during writes.
ADV#	Input	**ADDRESS VALID:** Active low input. During synchronous read operations, addresses are latched on the rising edge of ADV#, or on the next valid CLK edge with ADV# low, whichever occurs first. In asynchronous mode, the address is latched when ADV# going high or continuously flows through if ADV# is held low. WARNING: Designs not using ADV# must tie it to VSS to allow addresses to flow through.
CE#	Input	**CHIP ENABLE:** Active low input. CE# low selects the associated flash memory die. When asserted, flash internal control logic, input buffers, decoders, and sense amplifiers are active. When deasserted, the associated flash die is deselected, power is reduced to standby levels, data and WAIT outputs are placed in high-Z state. WARNING: All chip enables must be high when device is not in use.
CLK	Input	**CLOCK:** Synchronizes the device with the system’s bus frequency in synchronous-read mode. During synchronous read operations, addresses are latched on the rising edge of ADV#, or on the next valid CLK edge with ADV# low, whichever occurs first. WARNING: Designs not using CLK for synchronous read mode must tie it to VCCQ or VSS.
OE#	Input	**OUTPUT ENABLE:** Active low input. OE# low enables the device’s output data buffers during read cycles. OE# high places the data outputs and WAIT in High-Z.
RST#	Input	**RESET:** Active low input. RST# resets internal automation and inhibits write operations. This provides data protection during power transitions. RST# high enables normal operation. Exit from reset places the device in asynchronous read array mode.
WAIT	Output	**WAIT:** Indicates data valid in synchronous array or non-array burst reads. RCR[10], (WT) determines its polarity when asserted. WAIT’s active output is V_{OL} or V_{OH} when CE# and OE# are V_{IL}. WAIT is high-Z if CE# or OE# is V_{IH}. • In synchronous array or non-array read modes, WAIT indicates invalid data when asserted and valid data when deasserted. • In asynchronous page mode, and all write modes, WAIT is deasserted.
WE#	Input	**WRITE ENABLE:** Active low input. WE# controls writes to the device. Address and data are latched on the rising edge of WE#.
WP#	Input	**WRITE PROTECT:** Active low input. WP# low enables the lock-down mechanism. Blocks in lock-down cannot be unlocked with the Unlock command. WP# high overrides the lock-down function enabling blocks to be erased or programmed using software commands.
VPP	Power/ Input	**ERASE AND PROGRAM POWER:** A valid voltage on this pin allows erasing or programming. Memory contents cannot be altered when VPP ≤ V_{PPLK}. Block erase and program at invalid VPP voltages should not be attempted. Set VPP = V_{PPL} for in-system program and erase operations. To accommodate resistor or diode drops from the system supply, the V_{IH} level of VPP can be as low as V_{PPL} min. VPP must remain above V_{PPL} min to perform in-system flash modification. VPP may be 0 V during read operations. V_{PPH} can be applied to main blocks for 1000 cycles maximum and to parameter blocks for 2500 cycles. VPP can be connected to 9 V for a cumulative total not to exceed 80 hours. Extended use of this pin at 9 V may reduce block cycling capability.
VCC	Power	**DEVICE CORE POWER SUPPLY:** Core (logic) source voltage. Writes to the flash array are inhibited when VCC ≤ V_{LKO}. Operations at invalid VCC voltages should not be attempted.
VCCQ	Power	**OUTPUT POWER SUPPLY:** Output-driver source voltage.
VSS	Power	**GROUND:** Connect to system ground. Do not float any VSS connection.

Table 4: TSOP and Easy BGA Signal Descriptions (Sheet 2 of 2)

Symbol	Type	Name and Function
RFU	—	**RESERVED FOR FUTURE USE:** Reserved by Numonyx for future device functionality and enhancement. These should be treated in the same way as a Don't Use (DU) signal.
DU	—	**DON'T USE:** Do not connect to any other signal, or power supply; must be left floating.
NC	—	**NO CONNECT:** No internal connection; can be driven or floated.

4.1 Dual-Die Configurations

Figure 6: 512-Mbit Easy BGA Block Diagram



Note: $A_{max} = V_{IH}$ selects the Top parameter Die; $A_{max} = V_{IL}$ selects the Bottom Parameter Die.

5.0 Bus Operations

CE# low and RST# high enable device read operations. The device internally decodes upper address inputs to determine the accessed block. ADV# low opens the internal address latches. OE# low activates the outputs and gates selected data onto the I/O bus.

In asynchronous mode, the address is latched when ADV# goes high or continuously flows through if ADV# is held low. In synchronous mode, the address is latched by the first of either the rising ADV# edge or the next valid CLK edge with ADV# low (WE# and RST# must be V_{IH}; CE# must be V_{IL}).

Bus cycles to/from the P33-65nm device conform to standard microprocessor bus operations. Table 5, "Bus Operations Summary"summarizes the bus operations and the logic levels that must be applied to the device control signal inputs.

Table 5: Bus Operations Summary

Bus Operation		RST#	CLK	ADV#	CE#	OE#	WE#	WAIT	DQ[15:0]	Notes
Read	Asynchronous	V_{IH}	X	L	L	L	H	Deasserted	Output	
	Synchronous	V_{IH}	Running	L	L	L	H	Driven	Output	
Write		V_{IH}	X	L	L	H	L	High-Z	Input	1
Output Disable		V_{IH}	X	X	L	H	H	High-Z	High-Z	2
Standby		V_{IH}	X	X	H	X	X	High-Z	High-Z	2
Reset		V_{IL}	X	X	X	X	X	High-Z	High-Z	2,3

Notes:

1. Refer to the Table 7, "Command Bus Cycles" on page 19 for valid DQ[15:0] during a write operation.
2. X = Don't Care (H or L).
3. RST# must be at V_{SS} ± 0.2 V to meet the maximum specified power-down current.

5.1 Read

To perform a read operation, RST# and WE# must be deasserted while CE# and OE# are asserted. CE# is the device-select control. When asserted, it enables the flash memory device. OE# is the data-output control. When asserted, the addressed flash memory data is driven onto the I/O bus.

5.2 Write

To perform a write operation, both CE# and WE# are asserted while RST# and OE# are deasserted. During a write operation, address and data are latched on the rising edge of WE# or CE#, whichever occurs first. Table 7, "Command Bus Cycles" on page 19 shows the bus cycle sequence for each of the supported device commands, while Table 6, "Command Codes and Definitions" on page 17 describes each command. See Section 15.0, "AC Characteristics" on page 48 for signal-timing details.

Note: Write operations with invalid VCC and/or VPP voltages can produce spurious results and should not be attempted.

5.3 Output Disable

When OE# is deasserted, device outputs DQ[15:0] are disabled and placed in a high-impedance (High-Z) state, WAIT is also placed in High-Z.

5.4 Standby

When CE# is deasserted the device is deselected and placed in standby, substantially reducing power consumption. In standby, the data outputs are placed in High-Z, independent of the level placed on OE#. Standby current, I_{CCS}, is the average current measured over any 5 ms time interval, 5 µs after CE# is deasserted. During standby, average current is measured over the same time interval 5 µs after CE# is deasserted.

When the device is deselected (while CE# is deasserted) during a program or erase operation, it continues to consume active power until the program or erase operation is completed.

5.5 Reset

As with any automated device, it is important to assert RST# when the system is reset. When the system comes out of reset, the system processor attempts to read from the flash memory if it is the system boot device. If a CPU reset occurs with no flash memory reset, improper CPU initialization may occur because the flash memory may be providing status information rather than array data. Flash memory devices from Numonyx allow proper CPU initialization following a system reset through the use of the RST# input. RST# should be controlled by the same low-true reset signal that resets the system CPU.

After initial power-up or reset, the device defaults to asynchronous Read Array mode, and the Status Register is set to 0x80. Asserting RST# de-energizes all internal circuits, and places the output drivers in High-Z. When RST# is asserted, the device shuts down the operation in progress, a process which takes a minimum amount of time to complete. When RST# has been deasserted, the device is reset to asynchronous Read Array state.

Note: If RST# is asserted during a program or erase operation, the operation is terminated and the memory contents at the aborted location (for a program) or block (for an erase) are no longer valid, because the data may have been only partially written or erased.

When returning from a reset (RST# deasserted), a minimum wait is required before the initial read access outputs valid data. Also, a minimum delay is required after a reset before a write cycle can be initiated. After this wake-up interval passes, normal operation is restored. See Section 15.0, “AC Characteristics” on page 48 for details about signal-timing.

6.0 Command Set

6.1 Device Command Codes

The system Central Processing Unit provides control of all in-system read, write, and erase operations of the device via the system bus. The on-chip WSM manages all block-erase and word-program algorithms.

Device commands are written to the CUI to control all flash memory device operations. The CUI does not occupy an addressable memory location; it is the mechanism through which the flash device is controlled. Table 6 shows valid device command codes and descriptions.

Table 6: Command Codes and Definitions (Sheet 1 of 2)

Mode	Code	Device Mode	Description
Read	0xFF	Read Array	Places the device in Read Array mode. Array data is output on DQ[15:0].
	0x70	Read Status Register	Places the device in Read Status Register mode. The device enters this mode after a program or erase command is issued. SR data is output on DQ[7:0].
	0x90	Read Device ID or Configuration Register	Places device in Read Device Identifier mode. Subsequent reads output manufacturer/device codes, Configuration Register data, Block Lock status, or OTP register data on DQ[15:0].
	0x98	Read Query	Places the device in Read Query mode. Subsequent reads output Common Flash Interface information on DQ[7:0].
	0x50	Clear Status Register	The WSM can only set SR error bits. The Clear Status Register command is used to clear the SR error bits.
Write	0x40	Word Program Setup	First cycle of a 2-cycle programming command; prepares the CUI for a write operation. On the next write cycle, the address and data are latched and the WSM executes the programming algorithm at the addressed location. During program operations, the device responds only to Read Status Register and Program Suspend commands. CE# or OE# must be toggled to update the Status Register in asynchronous read. CE# or ADV# must be toggled to update the SR Data for synchronous Non-array reads. The Read Array command must be issued to read array data after programming has finished.
	0xE8	Buffered Program	This command loads a variable number of words up to the buffer size of 512 words onto the program buffer.
	0xD0	Buffered Program Confirm	The confirm command is Issued after the data streaming for writing into the buffer is done. This instructs the WSM to perform the Buffered Program algorithm, writing the data from the buffer to the flash memory array.
	0x80	BEFP Setup	First cycle of a 2-cycle command; initiates the BEFP mode. The CUI then waits for the BEFP Confirm command, 0xD0, that initiates the BEFP algorithm. All other commands are ignored when BEFP mode begins.
	0xD0	BEFP Confirm	If the previous command was BEFP Setup (0x80), the CUI latches the address and data, and prepares the device for BEFP mode.
Erase	0x20	Block Erase Setup	First cycle of a 2-cycle command; prepares the CUI for a block-erase operation. The WSM performs the erase algorithm on the block addressed by the Erase Confirm command. If the next command *is not* the Erase Confirm (0xD0) command, the CUI sets Status Register bits SR [5,4], and places the device in Read Status Register mode.
	0xD0	Block Erase Confirm	If the first command was Block Erase Setup (0x20), the CUI latches the address and data, and the WSM erases the addressed block. During block-erase operations, the device responds only to Read Status Register and Erase Suspend commands. CE# or OE# must be toggled to update the Status Register in asynchronous read. CE# or ADV# must be toggled to update the SR Data for synchronous Non-array reads.

Table 6: Command Codes and Definitions (Sheet 2 of 2)

Mode	Code	Device Mode	Description
Suspend	0xB0	Program or Erase Suspend	This command issued to any device address initiates a suspend of the currently-executing program or block erase operation. The Status Register indicates successful suspend operation by setting either SR.2 (program suspended) or SR 6 (erase suspended), along with SR.7 (ready). The WSM remains in the suspend mode regardless of control signal states (except for RST# asserted).
	0xD0	Suspend Resume	This command issued to any device address resumes the suspended program or block-erase operation.
Protection	0x60	Block lock Setup	First cycle of a 2-cycle command; prepares the CUI for block lock configuration changes. If the next command is not Block Lock (0x01), Block Unlock (0xD0), or Block Lock-Down (0x2F), the CUI sets SR.5 and SR.4, indicating a command sequence error.
	0x01	Block lock	If the previous command was Block Lock Setup (0x60), the addressed block is locked.
	0xD0	Unlock Block	If the previous command was Block Lock Setup (0x60), the addressed block is unlocked. If the addressed block is in a lock-down state, the operation has no effect.
	0x2F	Lock-Down Block	If the previous command was Block Lock Setup (0x60), the addressed block is locked down.
	0xC0	Protection program setup	First cycle of a 2-cycle command; prepares the device for a OTP register or Lock Register program operation. The second cycle latches the register address and data, and starts the programming algorithm to program data the the OTP array.
Configuration	0x60	Read Configuration Register Setup	First cycle of a 2-cycle command; prepares the CUI for device read configuration. If the Set Read Configuration Register command (0x03) is not the next command, the CUI sets Status Register bits SR.5 and SR.4, indicating a command sequence error.
	0x03	Read Configuration Register	If the previous command was Read Configuration Register Setup (0x60), the CUI latches the address and writes A[15:0]to the Read Configuration Register. Following a Configure RCR command, subsequent read operations access array data.
blank check	0xBC	Blank Check	First cycle of a 2-cycle command; initiates the Blank Check operation on a main block.
	0xD0	Blank Check Confirm	Second cycle of blank check command sequence; it latches the block address and executes blank check on the main array block.
other	0xEB	Extended Function Interface command	This command is used in extended function interface. first cycle of a multiple-cycle command second cycle is a Sub-Op-Code, the data written on third cycle is one less than the word count; the allowable value on this cycle are 0 through 511. The subsequent cycles load data words into the program buffer at a specified address until word count is achieved.

6.2 Device Command Bus Cycles

Device operations are initiated by writing specific device commands to the CUI. See Table 7, "Command Bus Cycles" on page 19. Several commands are used to modify array data including Word Program and Block Erase commands. Writing either command to the CUI initiates a sequence of internally-timed functions that culminate in the completion of the requested task. However, the operation can be aborted by either asserting RST# or by issuing an appropriate suspend command.

Table 7: Command Bus Cycles (Sheet 1 of 2)

Mode	Command	Bus Cycles	First Bus Cycle			Second Bus Cycle		
			Oper	Addr[(1)]	Data[(2)]	Oper	Addr[(1)]	Data[(2)]
Read	Read Array	1	Write	DnA	0xFF	-	-	-
	Read Device Identifier	≥ 2	Write	DnA	0x90	Read	DBA + IA	ID
	Read CFI	≥ 2	Write	DnA	0x98	Read	DBA + CFI-A	CFI-D
	Read Status Register	2	Write	DnA	0x70	Read	DnA	SRD
	Clear Status Register	1	Write	DnA	0x50	-	-	-
Program	Word Program	2	Write	WA	0x40	Write	WA	WD
	Buffered Program[(3)]	> 2	Write	WA	0xE8	Write	WA	N - 1
	Buffered Enhanced Factory Program (BEFP)[(4)]	> 2	Write	WA	0x80	Write	WA	0xD0
Erase	Block Erase	2	Write	BA	0x20	Write	BA	0xD0
Suspend	Program/Erase Suspend	1	Write	DnA	0xB0	-	-	-
	Program/Erase Resume	1	Write	DnA	0xD0	-	-	-
Protection	Lock Block	2	Write	BA	0x60	Write	BA	0x01
	Unlock Block	2	Write	BA	0x60	Write	BA	0xD0
	Lock-down Block	2	Write	BA	0x60	Write	BA	0x2F
	Program OTP register	2	Write	PRA	0xC0	Write	OTP-RA	OTP-D
	Program Lock Register	2	Write	LRA	0xC0	Write	LRA	LRD
Configuration	Program Read Configuration Register	2	Write	RCD	0x60	Write	RCD	0x03

Table 7: Command Bus Cycles (Sheet 2 of 2)

Mode	Command	Bus Cycles	First Bus Cycle			Second Bus Cycle		
			Oper	Addr(1)	Data(2)	Oper	Addr(1)	Data(2)
Others	Blank Check	2	Write	BA	0xBC	Write	BA	D0
	Extended Function Interface command(5)	>2	Write	WA	0xEB	Write	WA	Sub-Op code

Notes:

1. First command cycle address should be the same as the operation's target address.
DBA = Device Base Address (NOTE: needed for dual-die 512Mbit device)
DnA = Address within the device.
IA = Identification code address offset.
CFI-A = Read CFI address offset.
WA = Word address of memory location to be written.
BA = Address within the block.
OTP-RA = OTP register address.
LRA = Lock Register address.
RCD = Read Configuration Register data on A[16:1].
2. ID = Identifier data.
CFI-D = CFI data on DQ[15:0].
SRD = Status Register data.
WD = Word data.
N = Word count of data to be loaded into the write buffer.
OTP-D = OTP register data.
LRD = Lock Register data.
3. The second cycle of the Buffered Program Command is the word count of the data to be loaded into the write buffer. This is followed by up to 512 words of data.Then the confirm command (0xD0) is issued, triggering the array programming operation.
4. The confirm command (0xD0) is followed by the buffer data.
5. The second cycle is a Sub-Op-Code, the data written on third cycle is N-1; 1=< N <=512. The subsequent cycles load data words into the program buffer at a specified address until word count is achieved, after the data words are loaded, the final cycle is the confirm cycle 0xD0)

7.0 Read Operation

The device can be in any of four read states: Read Array, Read Identifier, Read Status or Read Query. Upon power-up, or after a reset, the device defaults to Read Array mode. To change the read state, the appropriate read command must be written to the device (see Section 6.2, “Device Command Bus Cycles” on page 18). The following sections describe read-mode operations in detail.

The device supports two read modes: asynchronous page mode and synchronous burst mode. Asynchronous page mode is the default read mode after device power-up or a reset. The RCR must be configured to enable synchronous burst reads of the flash memory array (see Section 11.1, “Read Configuration Register” on page 34).

7.1 Asynchronous Page-Mode Read

Following a device power-up or reset, asynchronous page mode is the default read mode and the device is set to Read Array mode. However, to perform array reads after any other device operation (e.g. write operation), the Read Array command must be issued in order to read from the flash memory array.

Note: Asynchronous page-mode reads can only be performed when RCR.15 is set

The Clear Status Register command clears the status register. It functions independent of VPP. The WSM sets and clears SR[7,6,2], but it sets bits SR[5,3,1] without clearing them. The Status Register should be cleared before starting a command sequence to avoid any ambiguity. A device reset also clears the Status Register.

To perform an asynchronous page-mode read, an address is driven onto the address bus, and CE# and ADV# are asserted. WE# and RST# must already have been deasserted. WAIT is deasserted during asynchronous page mode. ADV# can be driven high to latch the address, or it must be held low throughout the read cycle. CLK is not used for asynchronous page-mode reads, and is ignored. If only asynchronous reads are to be performed, CLK should be tied to a valid V_{IH} level, WAIT signal can be floated and ADV# must be tied to ground. Array data is driven onto DQ[15:0] after an initial access time t_{AVQV} delay. (see Section 15.0, “AC Characteristics” on page 48).

In asynchronous page mode, sixteen data words are “sensed” simultaneously from the flash memory array and loaded into an internal page buffer. The buffer word corresponding to the initial address on the Address bus is driven onto DQ[15:0] after the initial access delay. The lowest four address bits determine which word of the 16-word page is output from the data buffer at any given time.

7.2 Synchronous Burst-Mode Read

To perform a synchronous burst-read, an initial address is driven onto the address bus, and CE# and ADV# are asserted. WE# and RST# must already have been deasserted. ADV# is asserted, and then deasserted to latch the address. Alternately, ADV# can remain asserted throughout the burst access, in which case the address is latched on the next valid CLK edge while ADV# is asserted.

During synchronous array and non-array read modes, the first word is output from the data buffer on the next valid CLK edge after the initial access latency delay (see Section 11.1.2, “Latency Count” on page 35). Subsequent data is output on valid CLK edges following a minimum delay. However, for a synchronous non-array read, the same word of data will be output on successive clock edges until the burst length requirements are satisfied. Refer to the following waveforms for more detailed information:

- Figure 20, “Synchronous Single-Word Array or Non-array Read Timing” on page 53

- Figure 21, “Continuous Burst Read, showing an Output Delay Timing” on page 53
- Figure 22, “Synchronous Burst-Mode Four-Word Read Timing” on page 54

7.3 Read Device Identifier

The Read Device Identifier command instructs the device to output manufacturer code, device identifier code, block-lock status, OTP register data, or configuration register data (see Section 6.2, “Device Command Bus Cycles” on page 18 for details on issuing the Read Device Identifier command). Table 8, “Device Identifier Information” on page 22 and Table 9, “Device ID codes” on page 22 show the address offsets and data values for this device.

Table 8: Device Identifier Information

Item	Address[1,2]	Data
Manufacturer Code	0x00	0x89h
Device ID Code	0x01	ID (see Table 9)
Block Lock Configuration: • Block Is Unlocked • Block Is Locked • Block Is not Locked-Down • Block Is Locked-Down	BBA + 0x02	Lock Bit: DQ0 = 0b0 DQ0 = 0b1 DQ1 = 0b0 DQ1 = 0b1
Read Configuration Register	0x05	RCR Contents
General Purpose Register[3]	DBA + 0x07	general data
Lock Register 0	0x80	PR-LK0
64-bit Factory-Programmed OTP register	0x81–0x84	Factory OTP register data
64-bit User-Programmable OTP Register	0x85–0x88	User OTP register data
Lock Register 1	0x89	OTP register lock data
128-bit User-Programmable OTP registers	0x8A–0x109	User OTP register data

Notes:
1. BBA = Block Base Address.
2. DBA = Device base Address, Numonyx reserves other configuration address locations
3. In P33-65nm, the GPR is used as read out register for Extended Functional interface command.

Table 9: Device ID codes

ID Code Type	Device Density	Device Identifier Codes	
		–T (Top Parameter)	–B (Bottom Parameter)
Device Code	256-Mbit	891F	8922

Note: The 512-Mbit devices do not have a Device ID associated with them. Each die within the stack can be identified by either of the 256-Mbit Device ID codes depending on its parameter option.

7.4 Read CFI

The Read CFI command instructs the device to output Common Flash Interface data when read.

8.0 Program Operation

The device supports three programming methods: Word Programming (40h), Buffered Programming (E8h, D0h), and Buffered Enhanced Factory Programming (80h, D0h). The following sections describe device programming in detail.

Successful programming requires the addressed block to be unlocked. If the block is locked down, WP# must be deasserted and the block must be unlocked before attempting to program the block. Attempting to program a locked block causes a program error (SR.4 and SR.1 set) and termination of the operation. See Section 10.0, "Security" on page 30 for details on locking and unlocking blocks.

8.1 Word Programming

Word programming operations are initiated by writing the Word Program Setup command to the device. This is followed by a second write to the device with the address and data to be programmed. The device outputs Status Register data when read. See Figure 33, "Word Program Flowchart" on page 72. VPP must be above V_{PPLK}, and within the specified V_{PPL} min/max values.

During programming, the WSM executes a sequence of internally-timed events that program the desired data bits at the addressed location, and verifies that the bits are sufficiently programmed. Programming the flash memory array changes "ones" to "zeros". Memory array bits that are zeros can be changed to ones only by erasing the block.

The Status Register can be examined for programming progress and errors by reading at any address. The device remains in the Read Status Register state until another command is written to the device.

Status Register bit SR.7 indicates the programming status while the sequence executes. Commands that can be issued to the device during programming are Program Suspend, Read Status Register, Read Device Identifier, Read CFI, and Read Array (this returns unknown data).

When programming has finished, Status Register bit SR.4 (when set) indicates a programming failure. If SR.3 is set, the WSM could not perform the word programming operation because VPP was outside of its acceptable limits. If SR.1 is set, the word programming operation attempted to program a locked block, causing the operation to abort.

Before issuing a new command, the Status Register contents should be examined and then cleared using the Clear Status Register command. Any valid command can follow, when word programming has completed.

8.2 Buffered Programming

The device features a 512-word buffer to enable optimum programming performance. For Buffered Programming, data is first written to an on-chip write buffer. Then the buffer data is programmed into the flash memory array in buffer-size increments. This can improve system programming performance significantly over non-buffered programming. (see Figure 35, "Buffer Program Flowchart" on page 74).

When the Buffered Programming Setup command is issued, Status Register information is updated and reflects the availability of the buffer. SR.7 indicates buffer availability: if set, the buffer is available; if cleared, the buffer is not available. To retry, issue the Buffered Programming Setup command again, and re-check SR.7. When SR.7 is set, the buffer is ready for loading.

On the next write, a word count is written to the device at the buffer address. This tells the device how many data words will be written to the buffer, up to the maximum size of the buffer.

On the next write, a device start address is given along with the first data to be written to the flash memory array. Subsequent writes provide additional device addresses and data. All data addresses must lie within the start address plus the word count. Optimum programming performance and lower power usage are obtained by aligning the starting address at the beginning of a 512-word boundary (A[9:1] = 0x00). The maximum buffer size would be 256-word if the misaligned address range is crossing a 512-word boundary during programming.

After the last data is written to the buffer, the Buffered Programming Confirm command must be issued to the original block address. The WSM begins to program buffer contents to the flash memory array. If a command other than the Buffered Programming Confirm command is written to the device, a command sequence error occurs and SR[7,5,4] are set. If an error occurs while writing to the array, the device stops programming, and SR[7,4] are set, indicating a programming failure.

When Buffered Programming has completed, additional buffer writes can be initiated by issuing another Buffered Programming Setup command and repeating the buffered program sequence. Buffered programming may be performed with VPP = V_{PPL} or V_{PPH} (see Section 13.2, “Operating Conditions” on page 45 for limitations when operating the device with VPP = V_{PPH}).

If an attempt is made to program past an erase-block boundary using the Buffered Program command, the device aborts the operation. This generates a command sequence error, and SR[5,4] are set.

If Buffered programming is attempted while VPP is below V_{PPLK}, SR[4,3] are set. If any errors are detected that have set Status Register bits, the Status Register should be cleared using the Clear Status Register command.

8.3 Buffered Enhanced Factory Programming

Buffered Enhanced Factory Programing (BEFP) speeds up Multi-Level Cell (MLC) flash programming. The enhanced programming algorithm used in BEFP eliminates traditional programming elements that drive up overhead in device programmer systems. (see Figure 36, “BEFP Flowchart” on page 75).

BEFP consists of three phases: Setup, Program/Verify, and Exit It uses a write buffer to spread MLC program performance across 512 data words. Verification occurs in the same phase as programming to accurately program the flash memory cell to the correct bit state.

A single two-cycle command sequence programs the entire block of data. This enhancement eliminates three write cycles per buffer: two commands and the word count for each set of 512 data words. Host programmer bus cycles fill the device’s write buffer followed by a status check. SR.0 indicates when data from the buffer has been programmed into sequential flash memory array locations.

Following the buffer-to-flash array programming sequence, the Write State Machine (WSM) increments internal addressing to automatically select the next 512-word array boundary. This aspect of BEFP saves host programming equipment the address-bus setup overhead.

With adequate continuity testing, programming equipment can rely on the WSM’s internal verification to ensure that the device has programmed properly. This eliminates the external post-program verification and its associated overhead.

8.3.1 BEFP Requirements and Considerations

BEFP requirements:

- Case temperature: T_C = 30 °C ± 10 °C
- Nominal VCC
- VPP driven to V_{PPH}
- Target block must be unlocked before issuing the BEFP Setup and Confirm commands
- The first-word address for the block to be programmed must be held constant from the setup phase through all data streaming into the target block, until transition to the exit phase is desired.
- The first-word address must align with the start of an array buffer boundary. Word buffer boundaries in the array are determined by A[8:0] (0x000 through 01FF); the alignment start point is A[8:0] = 0x000.

BEFP considerations:

- For optimum performance, cycling must be limited below 50 erase cycles per block. Some degradation in performance may occur is this limit is exceeded, but the internal algorithm continues to work properly.
- BEFP programs one block at a time; all buffer data must fall within a single block. If the internal address counter increments beyond the block’s maximum address, addressing wraps around to the beginning of the block.
- BEFP cannot be suspended
- Programming to the flash memory array can occur only when the buffer is full. If the number of words is less than 512, remaining locations must be filled with 0xFFFF.

8.3.2 BEFP Setup Phase

After receiving the BEFP Setup and Confirm command sequence, Status Register bit SR.7 (Ready) is cleared, indicating that the WSM is busy with BEFP algorithm startup. A delay before checking SR.7 is required to allow the WSM enough time to perform all of its setups and checks (Block-Lock status, VPP level, etc.). If an error is detected, SR.4 is set and BEFP operation terminates. If the block was found to be locked, SR.1 is also set. SR.3 is set if the error occurred due to an incorrect VPP level.

Note: Reading from the device after the BEFP Setup and Confirm command sequence outputs Status Register data. Do not issue the Read Status Register command; it will be interpreted as data to be loaded into the buffer.

8.3.3 BEFP Program / Verify Phase

After the BEFP Setup Phase has completed, the host programming system must check SR[7,0] to determine the availability of the write buffer for data streaming. SR.7 cleared indicates the device is busy and the BEFP program/verify phase is activated. SR.0 indicates the write buffer is available.

Two basic sequences repeat in this phase: loading of the write buffer, followed by buffer data programming to the array. For BEFP, the count value for buffer loading is always the maximum buffer size of 512 words. During the buffer-loading sequence, data is stored to sequential buffer locations starting at address 0x00. Programming of the buffer contents to the flash memory array starts as soon as the buffer is full. If the number of words is less than 512, the remaining buffer locations must be filled with 0xFFFF.

Caution: **The buffer must be completely filled for programming to occur. Supplying an address outside of the current block's range during a buffer-fill sequence causes the algorithm to exit immediately. Any data previously loaded into the buffer during the fill cycle is not programmed into the array.**

The starting address for data entry must be buffer size aligned, if not the BEFP algorithm will be aborted and the program fails and (SR.4) flag will be set.

Data words from the write buffer are directed to sequential memory locations in the flash memory array; programming continues from where the previous buffer sequence ended. The host programming system must poll SR.0 to determine when the buffer program sequence completes. SR.0 cleared indicates that all buffer data has been transferred to the flash array; SR.0 set indicates that the buffer is not available yet for the next fill cycle. The host system may check full status for errors at any time, but it is only necessary on a block basis after BEFP exit. After the buffer fill cycle, no write cycles should be issued to the device until SR.0 = 0 and the device is ready for the next buffer fill.

Note: Any spurious writes are ignored after a buffer fill operation and when internal program is proceeding.

The host programming system continues the BEFP algorithm by providing the next group of data words to be written to the buffer. Alternatively, it can terminate this phase by changing the block address to one outside of the current block’s range.

The Program/Verify phase concludes when the programmer writes to a different block address; data supplied must be 0xFFFF. Upon Program/Verify phase completion, the device enters the BEFP Exit phase.

8.3.4 BEFP Exit Phase

When SR.7 is set, the device has returned to normal operating conditions. A full status check should be performed at this time to ensure the entire block programmed successfully. When exiting the BEFP algorithm with a block address change, the read mode will not change. After BEFP exit, any valid command can be issued to the device.

8.4 Program Suspend

Issuing the Program Suspend command while programming suspends the programming operation. This allows data to be accessed from the device other than the one being programmed. The Program Suspend command can be issued to any device address. A program operation can be suspended to perform reads only. Additionally, a program operation that is running during an erase suspend can be suspended to perform a read operation (see Figure 34, “Program/Erase Suspend/Resume Flowchart” on page 73).

When a programming operation is executing, issuing the Program Suspend command requests the WSM to suspend the programming algorithm at predetermined points. The device continues to output Status Register data after the Program Suspend command is issued. Programming is suspended when Status Register bits SR[7,2] are set. Suspend latency is specified in Section 15.5, “Program and Erase Characteristics” on page 58.

To read data from the device, the Read Array command must be issued. Read Array, Read Status Register, Read Device Identifier, Read CFI, and Program Resume are valid commands during a program suspend.

During a program suspend, deasserting CE# places the device in standby, reducing active current. VPP must remain at its programming level, and WP# must remain unchanged while in program suspend. If RST# is asserted, the device is reset.

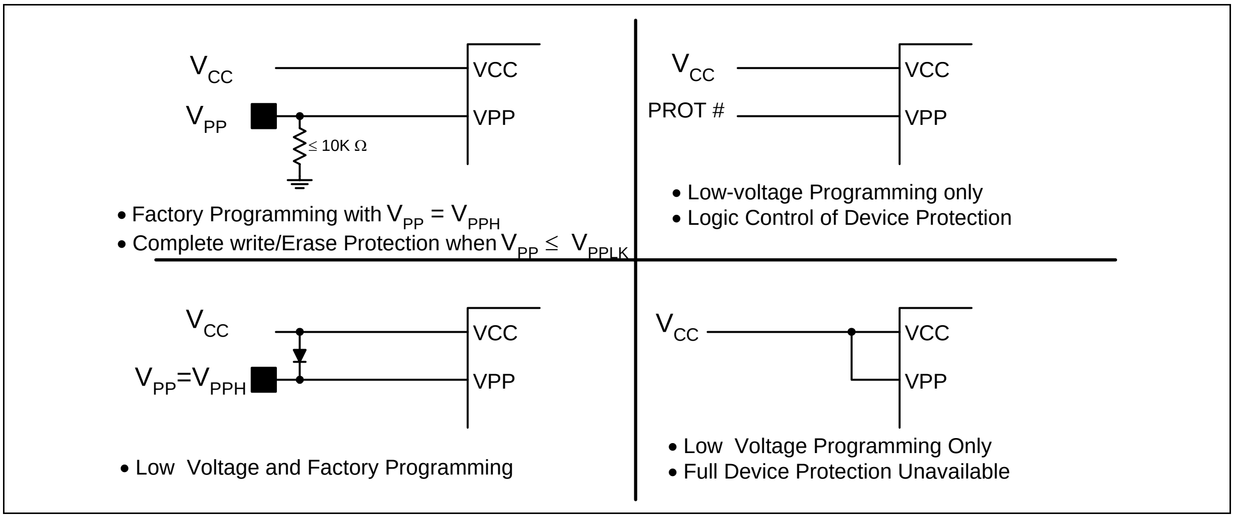
8.5 Program Resume

The Resume command instructs the device to continue programming, and automatically clears Status Register bits SR[7,2]. This command can be written to any address. If error bits are set, the Status Register should be cleared before issuing the next instruction. RST# must remain deasserted (see Figure 34, "Program/Erase Suspend/Resume Flowchart" on page 73).

8.6 Program Protection

When VPP = V_{IL}, absolute hardware write protection is provided for all device blocks. If VPP is at or below V_{PPLK}, programming operations halt and SR.3 is set indicating a VPP-level error. Block lock registers are not affected by the voltage level on VPP; they may still be programmed and read, even if VPP is less than V_{PPLK}.

Figure 7: Example VPP Supply Connections



9.0 Erase Operation

Flash erasing is performed on a block basis. An entire block is erased each time an erase command sequence is issued, and only one block is erased at a time. When a block is erased, all bits within that block read as logical ones. The following sections describe block erase operations in detail.

9.1 Block Erase

Block erase operations are initiated by writing the Block Erase Setup command to the address of the block to be erased (see Section 6.2, “Device Command Bus Cycles” on page 18). Next, the Block Erase Confirm command is written to the address of the block to be erased. If the device is placed in standby (CE# deasserted) during an erase operation, the device completes the erase operation before entering standby. VPP must be above V_{PPLK} and the block must be unlocked (see Figure 37, “Block Erase Flowchart” on page 76).

During a block erase, the WSM executes a sequence of internally-timed events that conditions, erases, and verifies all bits within the block. Erasing the flash memory array changes “zeros” to “ones”. Memory array bits that are ones can be changed to zeros only by programming the block.

The Status Register can be examined for block erase progress and errors by reading any address. The device remains in the Read Status Register state until another command is written. SR.0 indicates whether the addressed block is erasing. Status Register bit SR.7 is set upon erase completion.

Status Register bit SR.7 indicates block erase status while the sequence executes. When the erase operation has finished, Status Register bit SR.5 indicates an erase failure if set. SR.3 set would indicate that the WSM could not perform the erase operation because VPP was outside of its acceptable limits. SR.1 set indicates that the erase operation attempted to erase a locked block, causing the operation to abort.

Before issuing a new command, the Status Register contents should be examined and then cleared using the Clear Status Register command. Any valid command can follow once the block erase operation has completed.

9.2 Blank Check

The Blank Check operation determines whether a specified main block is blank (i.e. completely erased). Without Blank Check, Block Erase would be the only other way to ensure a block is completely erased. so Blank Check can be used to determine whether or not a prior erase operation was successful; this includes erase operations that may have been interrupted by power loss.

Blank check can apply to only one block at a time, and no operations other than Status Register Reads are allowed during Blank Check (e.g. reading array data, program, erase etc). Suspend and resume operations are not supported during Blank Check, nor is Blank Check supported during any suspended operations.

Blank Check operations are initiated by writing the Blank Check Setup command to the block address. Next, the Check Confirm command is issued along with the same block address. When a successful command sequence is entered, the device automatically enters the Read Status State. The WSM then reads the entire specified block, and determines whether any bit in the block is programmed or over-erased.

The status register can be examined for Blank Check progress and errors by reading any address within the block being accessed. During a blank check operation, the Status Register indicates a busy status (SR7 = 0). Upon completion, the Status Register indicates a ready status (SR7 = 1). The Status Register should be checked for any errors, and then cleared. If the Blank Check operation fails, which means the block is not completely erased, the Status

9.3 Erase Suspend

Issuing the Erase Suspend command while erasing suspends the block erase operation. This allows data to be accessed from memory locations other than the one being erased. The Erase Suspend command can be issued to any device address. A block erase operation can be suspended to perform a word or buffer program operation, or a read operation within any block except the block that is erase suspended (see Figure 34, "Program/Erase Suspend/Resume Flowchart" on page 73).

When a block erase operation is executing, issuing the Erase Suspend command requests the WSM to suspend the erase algorithm at predetermined points. The device continues to output Status Register data after the Erase Suspend command is issued. Block erase is suspended when Status Register bits SR[7,6] are set. Suspend latency is specified in Section 15.5, "Program and Erase Characteristics" on page 58.

To read data from the device (other than an erase-suspended block), the Read Array command must be issued. During Erase Suspend, a Program command can be issued to any block other than the erase-suspended block. Block erase cannot resume until program operations initiated during erase suspend complete. Read Array, Read Status Register, Read Device Identifier, Read CFI, and Erase Resume are valid commands during Erase Suspend. Additionally, Clear Status Register, Program, Program Suspend, Block Lock, Block Unlock, and Block Lock-Down are valid commands during Erase Suspend.

During an erase suspend, deasserting CE# places the device in standby, reducing active current. VPP must remain at a valid level, and WP# must remain unchanged while in erase suspend. If RST# is asserted, the device is reset.

9.4 Erase Resume

The Erase Resume command instructs the device to continue erasing, and automatically clears SR[7,6]. This command can be written to any address. If status register error bits are set, the Status Register should be cleared before issuing the next instruction. RST# must remain deasserted.

9.5 Erase Protection

When VPP = V_{IL}, absolute hardware erase protection is provided for all device blocks. If VPP is at or below V_{PPLK}, erase operations halt and SR.3 is set indicating a VPP-level error.

10.0 Security

The device features security modes used to protect the information stored in the flash memory array. The following sections describe each security mode in detail.

10.1 Block Locking

Individual instant block locking is used to protect user code and/or data within the flash memory array. All blocks power up in a locked state to protect array data from being altered during power transitions. Any block can be locked or unlocked with no latency. Locked blocks cannot be programmed or erased; they can only be read.

Software-controlled security is implemented using the Block Lock and Block Unlock commands. Hardware-controlled security can be implemented using the Block Lock-Down command along with asserting WP#. Also, VPP data security can be used to inhibit program and erase operations (see Section 8.6, "Program Protection" on page 27 and Section 9.5, "Erase Protection" on page 29).

The P33-65nm device also offers four pre-defined areas in the main array that can be configured as One-Time Programmable (OTP) for the highest level of security. These include the four 32 KB parameter blocks together as one and the three adjacent 128 KB main blocks. This is available for top or bottom parameter devices.

10.1.1 Lock Block

To lock a block, issue the Lock Block Setup command. The next command must be the Lock Block command issued to the desired block's address (see Section 6.2, "Device Command Bus Cycles" on page 18 and Figure 38, "Block Lock Operations Flowchart" on page 77). If the Set Read Configuration Register command is issued after the Block Lock Setup command, the device configures the RCR instead.

Block lock and unlock operations are not affected by the voltage level on VPP. The block lock bits may be modified and/or read even if VPP is at or below V_{PPLK}.

10.1.2 Unlock Block

The Unlock Block command is used to unlock blocks (see Section 6.2, "Device Command Bus Cycles" on page 18). Unlocked blocks can be read, programmed, and erased. Unlocked blocks return to a locked state when the device is reset or powered down. If a block is in a lock-down state, WP# must be deasserted before it can be unlocked (see Figure 8, "Block Locking State Diagram" on page 31).

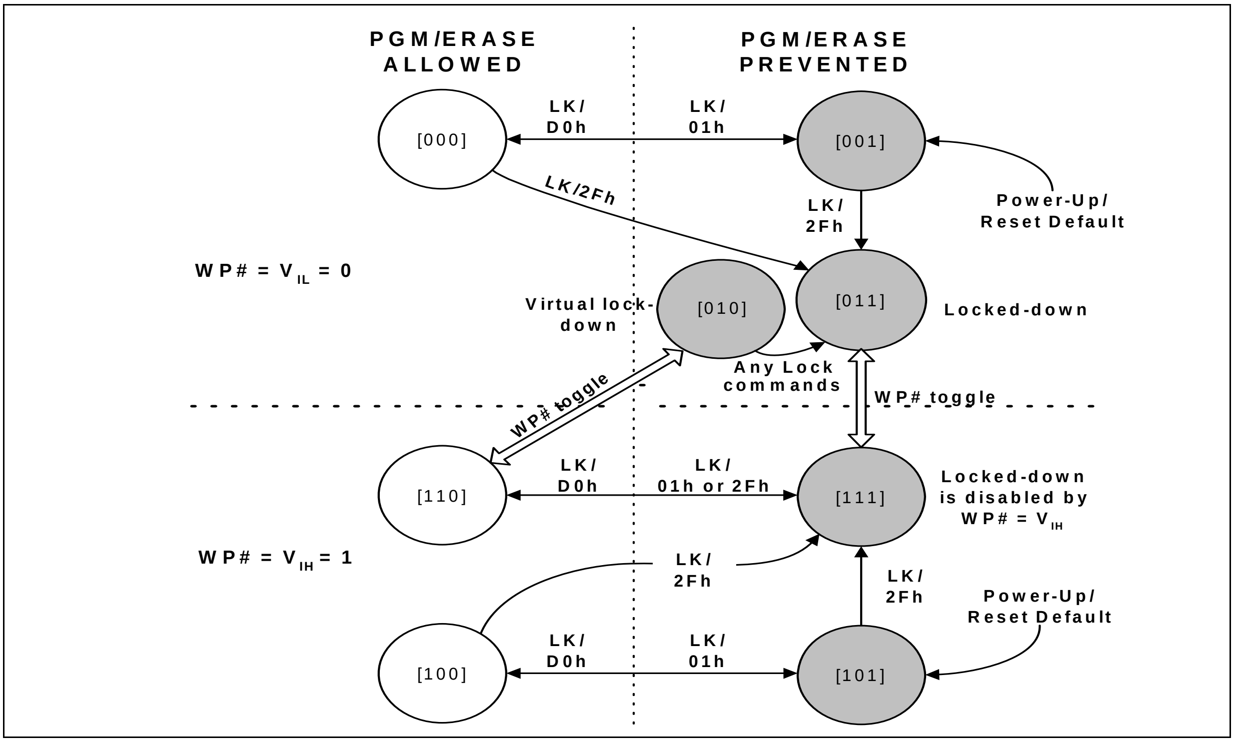
10.1.3 Lock-Down Block

A locked or unlocked block can be locked-down by writing the Lock-Down Block command sequence (see Section 6.2, "Device Command Bus Cycles" on page 18). Blocks in a lock-down state cannot be programmed or erased; they can only be read. However, unlike locked blocks, their locked state cannot be changed by software commands alone. A locked-down block can only be unlocked by issuing the Unlock Block command with WP# deasserted. To return an unlocked block to locked-down state, a Lock-Down command must be issued prior to changing WP# to V_{IL}. Locked-down blocks revert to the locked state upon reset or power up the device (see Figure 8, "Block Locking State Diagram" on page 31).

10.1.4 Block Lock Status

The Read Device Identifier command is used to determine a block's lock status (see Section 7.3, "Read Device Identifier" on page 22). Data bits DQ[1:0] display the addressed block's lock status; DQ0 is the addressed block's lock bit, while DQ1 is the addressed block's lock-down bit.

Figure 8: Block Locking State Diagram



Note: LK: Lock Setup Command, 60h; LK/D0h: Unlock Command; LK/01h: Lock Command; LK/2Fh: Lock-Down Command.

10.1.5 Block Locking During Suspend

Block lock and unlock changes can be performed during an erase suspend. To change block locking during an erase operation, first issue the Erase Suspend command. Monitor the Status Register until SR.7 and SR.6 are set, indicating the device is suspended and ready to accept another command.

Next, write the desired lock command sequence to a block, which changes the lock state of that block. After completing block lock or unlock operations, resume the erase operation using the Erase Resume command.

Note: A Lock Block Setup command followed by any command other than Lock Block, Unlock Block, or Lock-Down Block produces a command sequence error and set Status Register bits SR.4 and SR.5. If a command sequence error occurs during an erase suspend, SR.4 and SR.5 remains set, even after the erase operation is resumed. Unless the Status Register is cleared using the Clear Status Register command before resuming the erase operation, possible erase errors may be masked by the command sequence error.

If a block is locked or locked-down during an erase suspend of the *same* block, the lock status bits change immediately. However, the erase operation completes when it is resumed. Block lock operations cannot occur during a program suspend. See Appendix A, “Write State Machine” on page 80, which shows valid commands during an erase suspend.

10.2 Selectable OTP Blocks

Blocks from the main array may be optionally configured as OTP. Ask your local Numonyx representative for details about any of these selectable OTP implementations.

10.3 Password Access

Password Access is a security enhancement offered on the P33-65nm device.This feature protects information stored in main-array memory blocks by preventing content alteration or reads until a valid 64-bit password is received. Password Access may be combined with Non-Volatile Protection and/or Volatile Protection to create a multi-tiered solution. Please contact your Numonyx Sales for further details concerning Password Access.

11.0 Status Register

To read the Status Register, issue the Read Status Register command at any address. Status Register information is available to which the Read Status Register, Word Program, or Block Erase command was issued. SRD is automatically made available following a Word Program, Block Erase, or Block Lock command sequence. Reads from the device after any of these command sequences outputs the device's status until another valid command is written (e.g. the Read Array command).

The Status Register is read using single asynchronous-mode or synchronous burst mode reads. SRD is output on DQ[7:0], while 0x00 is output on DQ[15:8]. In asynchronous mode the falling edge of OE#, or CE# (whichever occurs first) updates and latches the Status Register contents. However, when reading the Status Register in synchronous burst mode, CE# or ADV# must be toggled to update SRD.

The Device Write Status bit (SR.7) provides overall status of the device. SR[6:1] present status and error information about the program, erase, suspend, VPP, and block-locked operations.

Table 10: Status Register Description

Status Register (SR)							**Default Value = 0x80**
Device Write Status	Erase Suspend Status	Erase Status	Program Status	VPP Status	Program Suspend Status	Block-Locked Status	BEFP Write Status
DWS	ESS	ES	PS	VPPS	PSS	BLS	BWS
7	6	5	4	3	2	1	0

Bit	Name	Description
7	Device Write Status (DWS)	0 = Device is busy; program or erase cycle in progress; SR.0 valid. 1 = Device is ready; SR[6:1] are valid.
6	Erase Suspend Status (ESS)	0 = Erase suspend not in effect. 1 = Erase suspend in effect.
5	Erase Status (ES)	0 = Erase successful. 1 = Erase fail or program sequence error when set with SR.4,SR.7.
4	Program Status (PS)	0 = Program successful. 1 = Program fail or program sequence error when set with SR.5,SR.7
3	VPP Status (VPPS)	0 = VPP within acceptable limits during program or erase operation. 1 = VPP < V_{PPLK} during program or erase operation.
2	Program Suspend Status (PSS)	0 = Program suspend not in effect. 1 = Program suspend in effect.
1	Block-Locked Status (BLS)	0 = Block not locked during program or erase. 1 = Block locked during program or erase; operation aborted.
0	BEFP Write Status (BWS)	After Buffered Enhanced Factory Programming (BEFP) data is loaded into the buffer: 0 = BEFP complete. 1 = BEFP in-progress.

Notes:

1. Always clear the Status Register prior to resuming erase operations. It avoids Status Register ambiguity when issuing commands during Erase Suspend. If a command sequence error occurs during an erase-suspend state, the Status Register contains the command sequence error status (SR[7,5,4] set). When the erase operation resumes and finishes, possible errors during the erase operation cannot be detected via the Status Register because it contains the previous error status
2. BEFP mode is only valid in main array.

11.0.1 Clear Status Register

The Clear Status Register command clears the status register. It functions independent of VPP. The WSM sets and clears SR[7,6,2], but it sets bits SR[5:3,1] without clearing them. The Status Register should be cleared before starting a command sequence to avoid any ambiguity. A device reset also clears the Status Register.

11.1 Read Configuration Register

The RCR is used to select the read mode (synchronous or asynchronous), and it defines the synchronous burst characteristics of the device. To modify RCR settings, use the Configure Read Configuration Register command (see Section 6.2, “Device Command Bus Cycles” on page 18).

RCR contents can be examined using the Read Device Identifier command, and then reading from offset 0x05 (see Section 7.3, “Read Device Identifier” on page 22).

The RCR is shown in Table 11. The following sections describe each RCR bit.

Table 11: Read Configuration Register Description (Sheet 1 of 2)

Read Configuration Register (RCR)															
Read Mode	Latency Count				WAIT Polarity	RES	WAIT Delay	Burst Seq	CLK Edge	RES	RES	Burst Wrap	Burst Length		
RM	LC[3:0]				WP	R	WD	BS	CE	R	R	BW	BL[2:0]		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit	Name	Description
15	Read Mode (RM)	0 = Synchronous burst-mode read 1 = Asynchronous page-mode read (default)
14:11	Latency Count (LC[3:0])	0010 =Code 2 0011 =Code 3 0100 =Code 4 0101 =Code 5 0110 =Code 6 0111 =Code 7 1000 =Code 8 1001 =Code 9 1010 =Code 10 1011 =Code 11 1100 =Code 12 1101 =Code 13 1110 =Code 14 1111 =Code 15 (default) (Other bit settings are reserved)
10	WAIT Polarity (WP)	0 =WAIT signal is active low (default) 1 =WAIT signal is active high
9	Reserved (R)	Default “0”, Non-changeable
8	WAIT Delay (WD)	0 =WAIT deasserted with valid data 1 =WAIT deasserted one data cycle before valid data (default)
7	Burst Sequence (BS)	Default “0”, Non-changeable
6	Clock Edge (CE)	0 = Falling edge 1 = Rising edge (default)
5:4	Reserved (R)	Default “0”, Non-changeable

Table 11: Read Configuration Register Description (Sheet 2 of 2)

3	Burst Wrap (BW)	0 =Wrap; Burst accesses wrap within burst length set by BL[2:0] 1 =No Wrap; Burst accesses do not wrap within burst length (default)
2:0	Burst Length (BL[2:0])	001 =4-word burst 010 =8-word burst 011 =16-word burst 111 =Continuous-word burst (default) (Other bit settings are reserved)

11.1.1 Read Mode

The Read Mode (RM) bit selects synchronous burst-mode or asynchronous page-mode operation for the device. When the RM bit is set, asynchronous page mode is selected (default). When RM is cleared, synchronous burst mode is selected.

11.1.2 Latency Count

The Latency Count (LC) bits tell the device how many clock cycles must elapse from the rising edge of ADV# (or from the first valid clock edge after ADV# is asserted) until the first valid data word is driven onto DQ[15:0]. The input clock frequency is used to determine this value and Figure 9 shows the data output latency for the different settings of LC. The maximum Latency Count for P33 would be Code 4 based on the Max clock frequency specification of 52 MHz, and there will be zero WAIT States when bursting within the word line. Please also refer to Section 11.1.3, “End of Word Line (EOWL) Considerations” on page 37 for more information on EOWL.

Refer to Table 12, “LC and Frequency Support” on page 36 for Latency Code Settings.

Figure 9: First-Access Latency Count

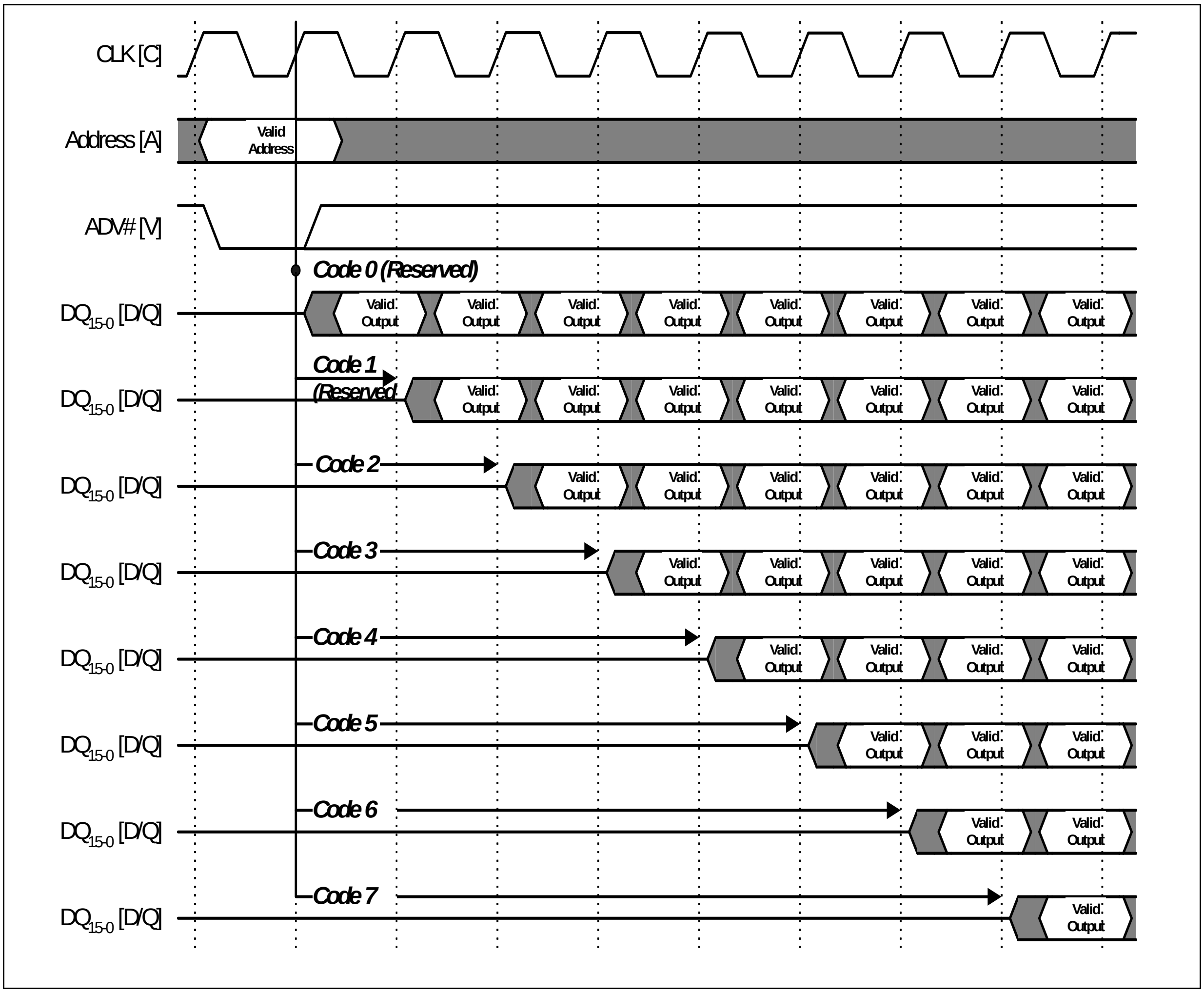
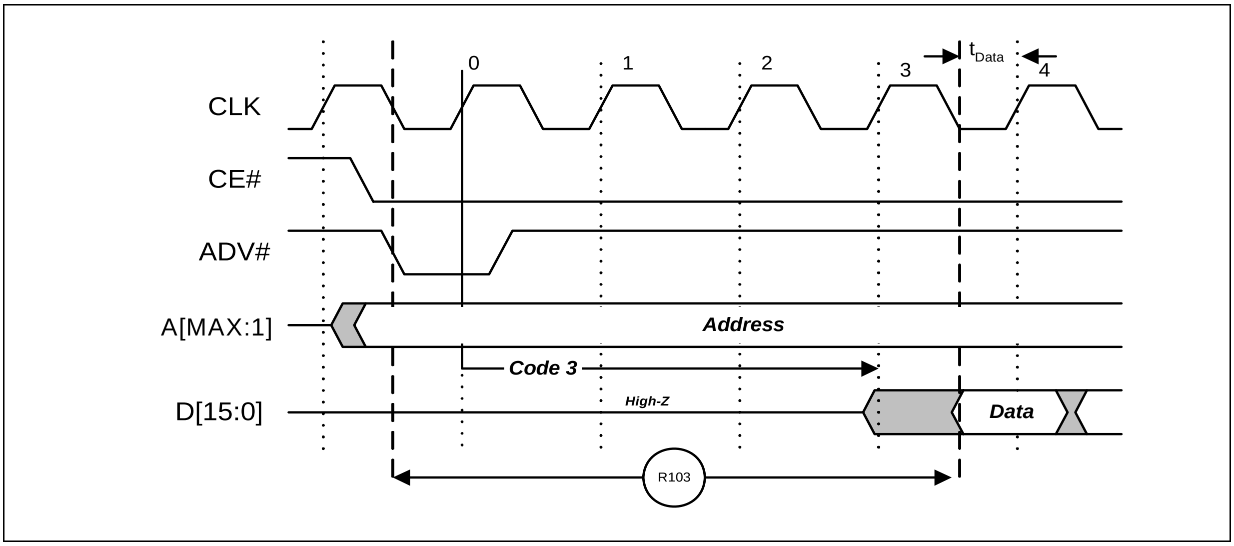


Table 12: LC and Frequency Support

Latency Count Settings	Frequency Support (MHz)
3	≤ 40
4	≤ 52

Figure 10: Example Latency Count Setting Using Code 3



11.1.3 End of Word Line (EOWL) Considerations

End of Wordline (EOWL) WAIT states can result when the starting address of the burst operation is not aligned to a 16-word boundary; that is, A[3:0] of start address does not equal 0x0. Figure 11, “End of Wordline Timing Diagram” on page 37 illustrates the end of wordline WAIT state(s), which occur after the first 16-word boundary is reached. The number of data words and the number of WAIT states is summarized in Table 13, “End of Wordline Data and WAIT state Comparison” on page 38for both P33-130nm and P33-65nm devices.

Figure 11: End of Wordline Timing Diagram

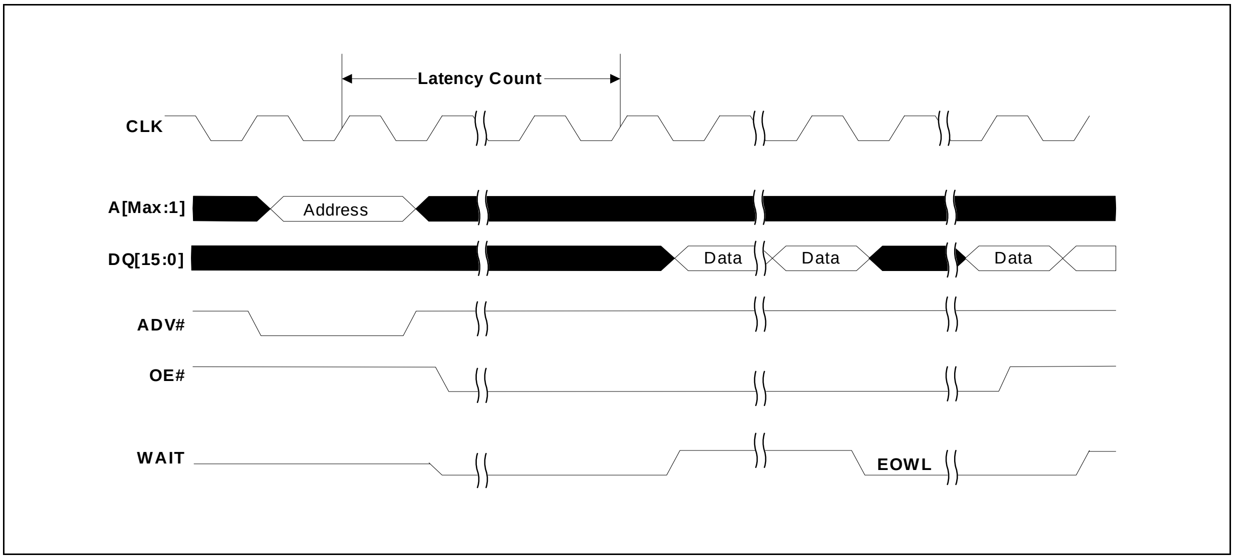


Table 13: End of Wordline Data and WAIT state Comparison

Latency Count	P33-130nm		P33-65nm	
	Data States	WAIT States	Data States	WAIT States
1	Not Supported	Not Supported	Not Supported	Not Supported
2	4	0 to 1	Not Supported	Not Supported
3	4	0 to 2	16	0 to 2
4	4	0 to 3	16	0 to 3
5	4	0 to 4	16	0 to 4
6	4	0 to 5	16	0 to 5
7	4	0 to 6	16	0 to 6
8	Not Supported	Not Supported	16	0 to 7
9			16	0 to 8
10			16	0 to 9
11			16	0 to 10
12			16	0 to 11
13			16	0 to 12
14			16	0 to 13
15			16	0 to 14

11.1.4 WAIT Polarity

The WAIT Polarity bit (WP), RCR.10 determines the asserted level (V_{OH} or V_{OL}) of WAIT. When WP is set, WAIT is asserted high. When WP is cleared, WAIT is asserted low (default). WAIT changes state on valid clock edges during active bus cycles (CE# asserted, OE# asserted, RST# deasserted).

11.1.5 WAIT Signal Function

The WAIT signal indicates data valid when the device is operating in synchronous mode (RCR.15=0). The WAIT signal is only “deasserted” when data is valid on the bus.

When the device is operating in synchronous non-array read mode, such as read status, read ID, or read query. The WAIT signal is also “deasserted” when data is valid on the bus.

WAIT behavior during synchronous non-array reads at the end of word line works correctly only on the first data access.

When the device is operating in asynchronous page mode, asynchronous single word read mode, and all write operations, WAIT is set to a deasserted state as determined by RCR.10. See Figure 18, “Asynchronous Single-Word Read (ADV# Latch)” on page 52, and Figure 19, “Asynchronous Page-Mode Read Timing” on page 52.

Table 14: WAIT Functionality Table

Condition	WAIT	Notes
CE# = '1', OE# = 'X' or CE# = '0', OE# = '1'	High-Z	1
CE# ='0', OE# = '0'	Active	1
Synchronous Array Reads	Active	1
Synchronous Non-Array Reads	Active	1
All Asynchronous Reads	Deasserted	1
All Writes	High-Z	1,2

Notes:
1. **Active:** WAIT is asserted until data becomes valid, then deasserts.
2. When OE# = V_{IH} during writes, WAIT = High-Z.

11.1.6 WAIT Delay

The WAIT Delay (WD) bit controls the WAIT assertion-delay behavior during synchronous burst reads. WAIT can be asserted either during or one data cycle before valid data is output on DQ[15:0]. When WD is set, WAIT is deasserted one data cycle *before* valid data (default). When WD is cleared, WAIT is deasserted *during* valid data.

11.1.7 Burst Sequence

The Burst Sequence (BS) bit selects linear-burst sequence (default). Only linear-burst sequence is supported. Table 15 shows the synchronous burst sequence for all burst lengths, as well as the effect of the Burst Wrap (BW) setting.

Table 15: Burst Sequence Word Ordering (Sheet 1 of 2)

Start Addr. (DEC)	Burst Wrap (RCR.3)	Burst Addressing Sequence (DEC)			
		4-Word Burst (BL[2:0] = 0b001)	8-Word Burst (BL[2:0] = 0b010)	16-Word Burst (BL[2:0] = 0b011)	Continuous Burst (BL[2:0] = 0b111)
0	0	0-1-2-3	0-1-2-3-4-5-6-7	0-1-2-3-4…14-15	0-1-2-3-4-5-6-…
1	0	1-2-3-0	1-2-3-4-5-6-7-0	1-2-3-4-5…15-0	1-2-3-4-5-6-7-…
2	0	2-3-0-1	2-3-4-5-6-7-0-1	2-3-4-5-6…15-0-1	2-3-4-5-6-7-8-…
3	0	3-0-1-2	3-4-5-6-7-0-1-2	3-4-5-6-7…15-0-1-2	3-4-5-6-7-8-9-…
4	0		4-5-6-7-0-1-2-3	4-5-6-7-8…15-0-1-2-3	4-5-6-7-8-9-10…
5	0		5-6-7-0-1-2-3-4	5-6-7-8-9…15-0-1-2-3-4	5-6-7-8-9-10-11…
6	0		6-7-0-1-2-3-4-5	6-7-8-9-10…15-0-1-2-3-4-5	6-7-8-9-10-11-12-…
7	0		7-0-1-2-3-4-5-6	7-8-9-10…15-0-1-2-3-4-5-6	7-8-9-10-11-12-13…
⋮	⋮	⋮	⋮	⋮	⋮
14	0			14-15-0-1-2…12-13	14-15-16-17-18-19-20-…
15	0			15-0-1-2-3…13-14	15-16-17-18-19-20-21-…
⋮	⋮	⋮	⋮	⋮	⋮

Table 15: Burst Sequence Word Ordering (Sheet 2 of 2)

0	1	0-1-2-3	0-1-2-3-4-5-6-7	0-1-2-3-4…14-15	0-1-2-3-4-5-6-…
1	1	1-2-3-4	1-2-3-4-5-6-7-8	1-2-3-4-5…15-16	1-2-3-4-5-6-7-…
2	1	2-3-4-5	2-3-4-5-6-7-8-9	2-3-4-5-6…16-17	2-3-4-5-6-7-8-…
3	1	3-4-5-6	3-4-5-6-7-8-9-10	3-4-5-6-7…17-18	3-4-5-6-7-8-9-…
4	1		4-5-6-7-8-9-10-11	4-5-6-7-8…18-19	4-5-6-7-8-9-10…
5	1		5-6-7-8-9-10-11-12	5-6-7-8-9…19-20	5-6-7-8-9-10-11…
6	1		6-7-8-9-10-11-12-13	6-7-8-9-10…20-21	6-7-8-9-10-11-12-…
7	1		7-8-9-10-11-12-13-14	7-8-9-10-11…21-22	7-8-9-10-11-12-13…
⋮	⋮	⋮	⋮	⋮	⋮
14	1			14-15-16-17-18…28-29	14-15-16-17-18-19-20-…
15	1			15-16-17-18-19…29-30	15-16-17-18-19-20-21-…

11.1.8 Clock Edge

The Clock Edge (CE) bit selects either a rising (default) or falling clock edge for CLK. This clock edge is used at the start of a burst cycle, to output synchronous data, and to assert/deassert WAIT.

11.1.9 Burst Wrap

The Burst Wrap (BW) bit determines whether 4, 8, or 16-word burst length accesses wrap within the selected word-length boundaries or cross word-length boundaries. When BW is set, burst wrapping does not occur (default). When BW is cleared, burst wrapping occurs.

11.1.10 Burst Length

The Burst Length bits (BL[2:0]) selects the linear burst length for all synchronous burst reads of the flash memory array. The burst lengths are 4-word, 8-word, 16-word, and continuous word.

Continuous burst accesses are linear only, and do not wrap within any word length boundaries (see Table 15, “Burst Sequence Word Ordering” on page 39). When a burst cycle begins, the device outputs synchronous burst data until it reaches the end of the “burstable” address space.

11.2 One-Time Programmable (OTP) Registers

The device contains 17 one-time programmable (OTP) registers that can be used to implement system security measures and/or device identification. Each OTP register can be individually locked.

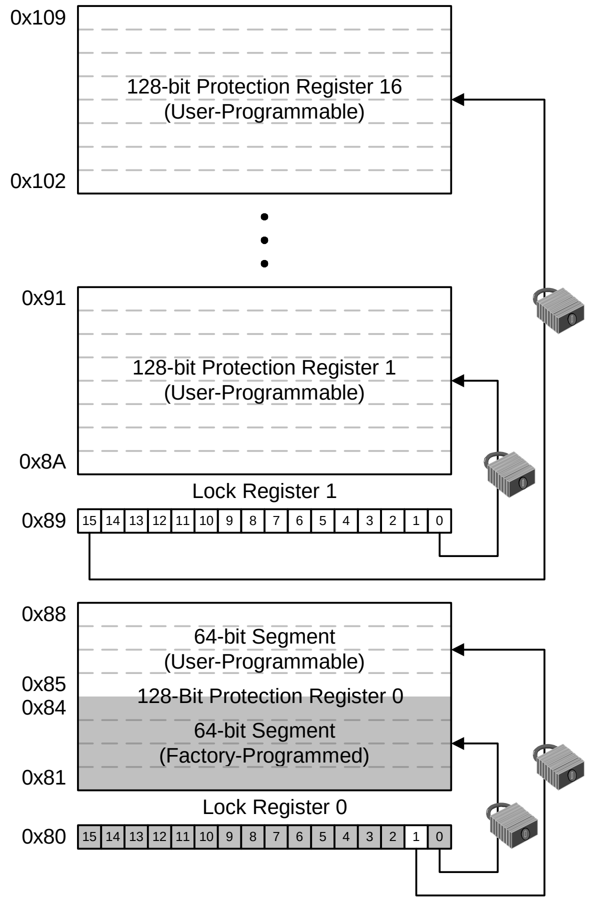
The first 128-bit OTP Register is comprised of two 64-bit (8-word) segments. The lower 64-bit segment is pre-programmed at the Numonyx factory with a unique 64-bit number. The other 64-bit segment, as well as the other sixteen 128-bit OTP Registers, are blank. Users can program these registers as needed. Once programmed, users can then lock the OTP Register(s) to prevent additional bit programming (see Figure 12, “OTP Register Map” on page 41).

The OTP Registers contain OTP bits; when programmed, PR bits cannot be erased. Each OTP Register can be accessed multiple times to program individual bits, as long as the register remains unlocked.

Each OTP Register has an associated Lock Register bit. When a Lock Register bit is programmed, the associated OTP Register can only be read; it can no longer be programmed. Additionally, because the Lock Register bits themselves are OTP, when programmed, Lock Register bits cannot be erased. Therefore, when a OTP Register is locked, it cannot be unlocked.

.

Figure 12: OTP Register Map



11.2.1 Reading the OTP Registers

The OTP Registers can be read from any address. To read the OTP Register, first issue the Read Device Identifier command at any address to place the device in the Read Device Identifier state (see Section 6.2, “Device Command Bus Cycles” on page 18). Next, perform a read operation using the address offset corresponding to the register to be read. Table 8, “Device Identifier Information” on page 22 shows the address offsets of the OTP Registers and Lock Registers. PR data is read 16 bits at a time.

11.2.2 Programming the OTP Registers

To program any of the OTP Registers, first issue the Program OTP Register command at the parameter's base address plus the offset to the desired OTP Register (see Section 6.2, “Device Command Bus Cycles” on page 18). Next, write the desired OTP Register data to the same OTP Register address (see Figure 12, “OTP Register Map” on page 41).

The device programs the 64-bit and 128-bit user-programmable OTP Register data 16 bits at a time (see Figure 39, “OTP Register Programming Flowchart” on page 78). Issuing the Program OTP Register command outside of the OTP Register's address space causes a program error (SR.4 set). Attempting to program a locked OTP Register causes a program error (SR.4 set) and a lock error (SR.1 set).

Note: When programming the OTP bits in the OTP registers for a **Top Parameter Device**, the following upper address bits must also be driven properly: A[Max:17] driven high (V_{IH}).

11.2.3 Locking the OTP Registers

Each OTP Register can be locked by programming its respective lock bit in the Lock Register. To lock a OTP Register, program the corresponding bit in the Lock Register by issuing the Program Lock Register command, followed by the desired Lock Register data (see Section 6.2, “Device Command Bus Cycles” on page 18). The physical addresses of the Lock Registers are 0x80 for register 0 and 0x89 for register 1. These addresses are used when programming the lock registers (see Table 8, “Device Identifier Information” on page 22).

Bit 0 of Lock Register 0 is already programmed during the manufacturing process at the “factory”, locking the lower, pre-programmed 64-bit region of the first 128-bit OTP Register containing the unique identification number of the device. Bit 1 of Lock Register 0 can be programmed by the user to lock the user-programmable, 64-bit region of the first 128-bit OTP Register. When programming Bit 1 of Lock Register 0, all other bits need to be left as ‘1’ such that the data programmed is 0xFFFD.

Lock Register 1 controls the locking of the upper sixteen 128-bit OTP Registers. Each of the 16 bits of Lock Register 1 correspond to each of the upper sixteen 128-bit OTP Registers. Programming a bit in Lock Register 1 locks the corresponding 128-bit OTP Register.

Caution: **After being locked, the OTP Registers cannot be unlocked.**

12.0 Power and Reset Specifications

12.1 Power-Up and Power-Down

Power supply sequencing is not required if VPP is connected to VCC or VCCQ. Otherwise VCC and VCCQ should attain their minimum operating voltage before applying VPP.

Power supply transitions should only occur when RST# is low. This protects the device from accidental programming or erasure during power transitions.

12.2 Reset Specifications

Asserting RST# during a system reset is important with automated program/erase devices because systems typically expect to read from flash memory when coming out of reset. If a CPU reset occurs without a flash memory reset, proper CPU initialization may not occur. This is because the flash memory may be providing status information, instead of array data as expected. Connect RST# to the same active low reset signal used for CPU initialization.

Also, because the device is disabled when RST# is asserted, it ignores its control inputs during power-up/down. Invalid bus conditions are masked, providing a level of memory protection.

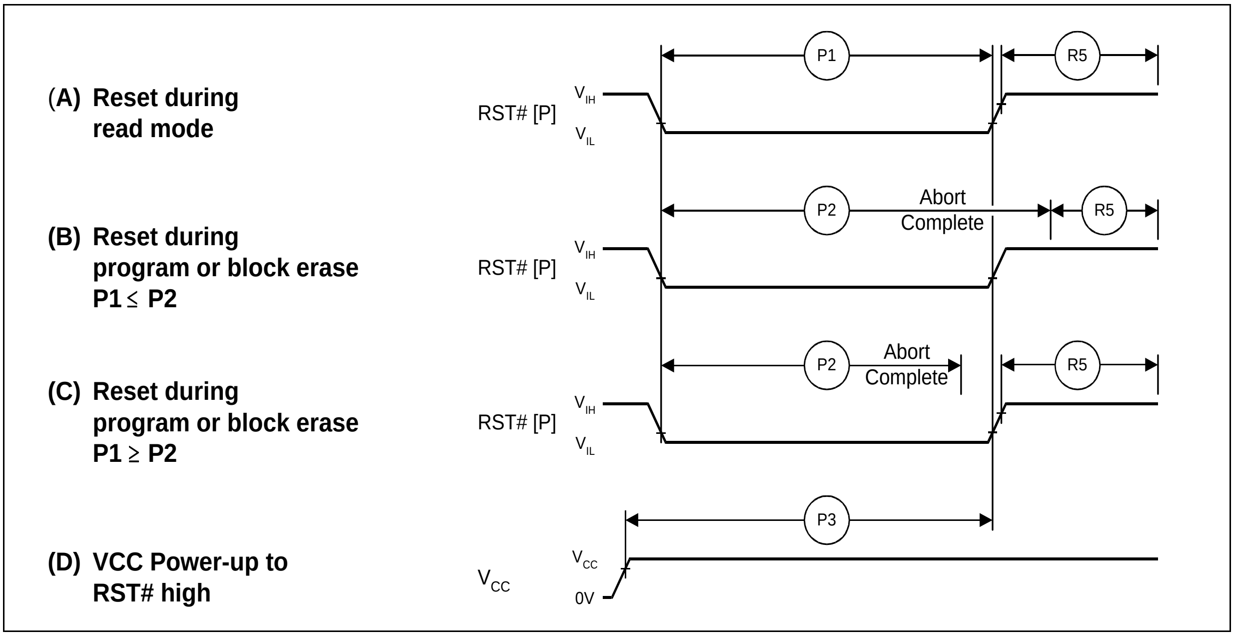
Table 16: Power and Reset

Num	Symbol	Parameter	Min	Max	Unit	Notes
P1	t_{PLPH}	RST# pulse width low	100	-	ns	1,2,3,4
P2	t_{PLRH}	RST# low to device reset during erase	-	25	µs	1,3,4,7
		RST# low to device reset during program	-	25		1,3,4,7
P3	t_{VCCPH}	VCC Power valid to RST# de-assertion (high)	300	-		1,4,5,6

Notes:

1. These specifications are valid for all device versions (packages and speeds).
2. The device may reset if t_{PLPH} is < t_{PLPH} Min, but this is not guaranteed.
3. Not applicable if RST# is tied to VCC.
4. Sampled, but not 100% tested.
5. When RST# is tied to the VCC supply, device will not be ready until t_{VCCPH} after VCC $\geq V_{CCMIN}$.
6. When RST# is tied to the VCCQ supply, device will not be ready until t_{VCCPH} after VCC $\geq V_{CCMIN}$.
7. Reset completes within t_{PLPH} if RST# is asserted while no erase or program operation is executing.

Figure 13: Reset Operation Waveforms



12.3 Power Supply Decoupling

Flash memory devices require careful power supply de-coupling. Three basic power supply current considerations are: 1) standby current levels; 2) active current levels; and 3) transient peaks produced when CE# and OE# are asserted and deasserted.

When the device is accessed, many internal conditions change. Circuits within the device enable charge-pumps, and internal logic states change at high speed. All of these internal activities produce transient signals. Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and correct de-coupling capacitor selection suppress transient voltage peaks.

Because Numonyx MLC flash memory devices draw their power from VCC, VPP, and VCCQ, each power connection should have a 0.1 µF ceramic capacitor to ground. High-frequency, inherently low-inductance capacitors should be placed as close as possible to package leads.

Additionally, for every eight devices used in the system, a 4.7 µF electrolytic capacitor should be placed between power and ground close to the devices. The bulk capacitor is meant to overcome voltage droop caused by PCB trace inductance.

13.0 Maximum Ratings and Operating Conditions

13.1 Absolute Maximum Ratings

Warning: Stressing the device beyond the Absolute Maximum Ratings may cause permanent damage. These are stress ratings only.

Table 17: Absolute Maximum Ratings

Parameter	Maximum Rating	Notes
Temperature under bias	–40 °C to +85 °C	-
Storage temperature	–65 °C to +125 °C	-
Voltage on any signal (except VCC, VPP and VCCQ)	–0.5 V to +4.1 V	1
VPP voltage	–0.2 V to +10 V	1,2,3
VCC voltage	–0.2 V to +4.1 V	1
VCCQ voltage	–0.2 V to +4.1 V	1
Output short circuit current	100 mA	4

Notes:

1. Voltages shown are specified with respect to V_{SS}. Minimum DC voltage is –0.5 V on input/output signals and –0.2 V on VCC, VCCQ, and VPP. During transitions, this level may undershoot to –2.0 V for periods less than 20 ns. Maximum DC voltage on VCC is VCC + 0.5 V, which, during transitions, may overshoot to VCC + 2.0 V for periods less than 20 ns. Maximum DC voltage on input/output signals and VCCQ is VCCQ + 0.5 V, which, during transitions, may overshoot to VCCQ + 2.0 V for periods less than 20 ns.
2. Maximum DC voltage on VPP may overshoot to +11.5 V for periods less than 20 ns.
3. Program/erase voltage is typically 2.3 V – 3.6 V. 9.0 V can be applied for 80 hours maximum total, to any blocks for 1000 cycles maximum. 9.0 V program/erase voltage may reduce block cycling capability.
4. Output shorted for no more than one second. No more than one output shorted at a time.

13.2 Operating Conditions

Note: Operation beyond the Operating Conditions is not recommended and extended exposure beyond the Operating Conditions may affect device reliability.

Table 18: Operating Conditions

Symbol	Parameter		Min	Max	Units	Notes
T_C	Operating Temperature		–40	+85	°C	1
VCC	VCC Supply Voltage		2.3	3.6	V	-
VCCQ	I/O Supply Voltage	CMOS inputs	2.3	3.6		-
		TTL inputs	2.4	3.6		
V_{PPL}	V_{PP} Voltage Supply (Logic Level)		1.5	3.6		2
V_{PPH}	Buffered Enhanced Factory Programming V_{PP}		8.5	9.5		
t_{PPH}	Maximum V_{PP} Hours	VPP = V_{PPH}	-	80	Hours	
Block Erase Cycles	Main and Parameter Blocks	VPP = V_{PPL}	100,000	-	Cycles	
	Main Blocks	VPP = V_{PPH}	100,000	-		
	Parameter Blocks	VPP = V_{PPH}	100,000	-		

Notes:

1. T_C = Case Temperature.
2. In typical operation VPP program voltage is V_{PPL}.

14.0 Electrical Specifications

14.1 DC Current Characteristics

Table 19: DC Current Characteristics (Sheet 1 of 2)

Sym	Parameter			CMOS Inputs (VCCQ = 2.3 V - 3.6 V)		TTL Inputs (VCCQ = 2.4 V - 3.6 V)		Unit	Test Conditions		Notes
				Typ	Max	Typ	Max				
I_{LI}	Input Load Current			-	±1	-	±2	µA	VCC = VCC Max VCCQ = VCCQ Max V_{IN} = VCCQ or V_{SS}		1,6
I_{LO}	Output Leakage Current	DQ[15:0], WAIT		-	±1	-	±10	µA	VCC = VCC Max VCCQ = VCCQ Max V_{IN} = VCCQ or V_{SS}		
I_{CCS}, I_{CCD}	VCC Standby, Power-Down		256-Mbit	65	210	65	210	µA	VCC = VCC Max VCCQ = VCC Max CE# =VCCQ RST# = VCCQ (for I_{CCS}) RST# = V_{SS} (for I_{CCD}) WP# = V_{IH}		1,2
			512-Mbit	130	420	130	420				
I_{CCR}	Average VCC Read Current	Asynchronous Single-Word f = 5 MHz (1 CLK)		26	31	26	31	mA	16-Word Read	VCC = VCC_{Max} CE# = V_{IL} OE# = V_{IH} Inputs: V_{IL} or V_{IH}	1
		Page-Mode Read f = 13 MHz (17 CLK)		12	16	12	16	mA	16-Word Read		
		Synchronous Burst f = 52 MHz, LC=4		19	22	19	22	mA	8-Word Read		
				16	18	16	18	mA	16-Word Read		
				21	24	21	24	mA	Continuous Read		
$I_{CCW,}$ I_{CCE}	VCC Program Current, VCC Erase Current			35	50	35	50	mA	VPP = V_{PPL}, Pgm/Ers in progress		1,3,5
				35	50	35	50		VPP = V_{PPH}, Pgm/Ers in progress		1,3,5
I_{CCWS}, I_{CCES}	VCC Program Suspend Current, VCC Erase Suspend Current		256-Mbit	65	210	65	210	µA	CE# = VCCQ; suspend in progress		1,3,4
			512-Mbit	70	225	70	225				
$I_{PPS,}$ $I_{PPWS,}$ IPPES	VPP Standby Current, VPP Program Suspend Current, VPP Erase Suspend Current			0.2	5	0.2	5	µA	VPP = V_{PPL}, suspend in progress		1,3,7
I_{PPR}	VPP Read			2	15	2	15	µA	VPP = V_{PPL}		1,3
I_{PPW}	VPP Program Current			0.05	0.10	0.05	0.10	mA	VPP = $V_{PPL,}$ program in progress		3
				0.05	0.10	0.05	0.10		VPP = $V_{PPH,}$ program in progress		
I_{PPE}	V_{PP} Erase Current			0.05	0.10	0.05	0.10	mA	VPP = $V_{PPL,}$ erase in progress		3
				0.05	0.10	0.05	0.10		VPP = $V_{PPH,}$ erase in progress		

Table 19: DC Current Characteristics (Sheet 2 of 2)

Sym	Parameter	CMOS Inputs (VCCQ = 2.3 V - 3.6 V)		TTL Inputs (VCCQ = 2.4 V - 3.6 V)		Unit	Test Conditions	Notes
		Typ	Max	Typ	Max			
I_{PPBC}	VPP Blank Check	0.05	0.10	0.05	0.10	mA	VPP = $V_{PPL,}$ erase in progress	3
		0.05	0.10	0.05	0.10		VPP = $V_{PPH,}$ erase in progress	

Notes:

1. All currents are RMS unless noted. Typical values at typical VCC, T_C = +25 °C.
2. I_{CCS} is the average current measured over any 5 ms time interval 5 µs after CE# is deasserted.
3. Sampled, not 100% tested.
4. I_{CCES} is specified with the device deselected. If device is read while in erase suspend, current is I_{CCES} plus I_{CCR}.
5. I_{CCW}, I_{CCE} measured over typical or max times specified in Section 15.5, “Program and Erase Characteristics” on page 58.
6. if V_{IN} > VCC the input load current increases to 10uA max.
7. the I_{PPS}, I_{PPWS}, I_{PPES} Will increase to 200uA when VPP/WP# is at V_{PPH}.

14.2 DC Voltage Characteristics

Table 20: DC Voltage Characteristics

Sym	Parameter	CMOS Inputs (VCCQ = 2.3 V – 3.6 V)		TTL Inputs (1) (VCCQ = 2.4 V – 3.6 V)		Unit	Test Conditions	Notes
		Min	Max	Min	Max			
V_{IL}	Input Low Voltage	-0.5	0.4	-0.5	0.6	V		2
V_{IH}	Input High Voltage	VCCQ – 0.4	VCCQ + 0.5	2.0	VCCQ + 0.5	V		
V_{OL}	Output Low Voltage	-	0.2	-	0.2	V	VCC = VCC Min VCCQ = VCCQ Min I_{OL} = 100 µA	-
V_{OH}	Output High Voltage	VCCQ – 0.2	-	VCCQ – 0.2	-	V	VCC = VCC Min VCCQ = VCCQ Min I_{OH} = –100 µA	-
V_{PPLK}	VPP Lock-Out Voltage	-	0.4	-	0.4	V		3
V_{LKO}	VCC Lock Voltage	1.5	-	1.5	-	V		-
V_{LKOQ}	VCCQ Lock Voltage	0.9	-	0.9	-	V		-
V_{PPL}	VPP Voltage Supply (Logic Level)	1.5	3.6	1.5	3.6	V		
V_{PPH}	Buffered Enhanced Factory Programming VPP	8.5	9.5	8.5	9.5	V		

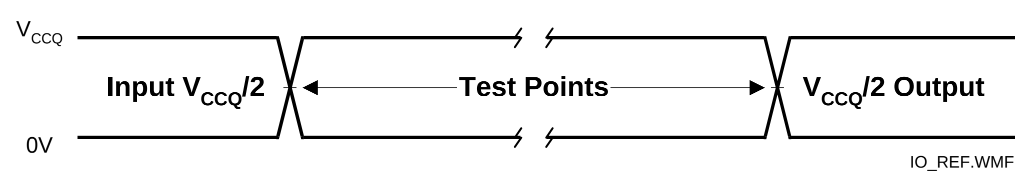
Notes:

1. Synchronous read mode is not supported with TTL inputs.
2. V_{IL} can undershoot to –0.4 V and V_{IH} can overshoot to VCCQ + 0.4 V for durations of 20 ns or less.
3. VPP $\leq$ V_{PPLK} inhibits erase and program operations. Do not use V_{PPL} and V_{PPH} outside their valid ranges.

15.0 AC Characteristics

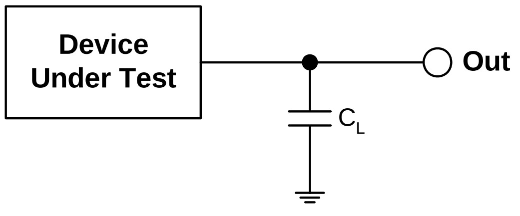
15.1 AC Test Conditions

Figure 14: AC Input/Output Reference Waveform



Note: AC test inputs are driven at VCCQ for Logic "1" and 0 V for Logic "0." Input/output timing begins/ends at VCCQ/2. Input rise and fall times (10% to 90%) < 5 ns. Worst-case speed occurs at VCC = VCCMin.

Figure 15: Transient Equivalent Testing Load Circuit

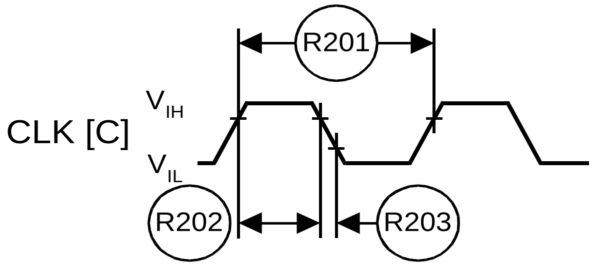


Notes:
1. See the following table for component values.
2. Test configuration component value for worst case speed conditions.
3. C_L includes jig capacitance

Table 21: Test Configuration Component Value for Worst Case Speed Conditions

Test Configuration	C_L (pF)
VCCQ Min Standard Test	30

Figure 16: Clock Input AC Waveform



15.2 Capacitance

Table 22: Capacitance

Symbol	Parameter	Signals	Min	Typ	Max	Unit	Condition	Note
C_{IN}	Input Capacitance	Address, Data, CE#, WE#, OE#, RST#, CLK, ADV#, WP#	2	6	7	pF	Typ temp = 25 °C, Max temp = 85 °C, VCC = (0 V - 3.6 V), VCCQ = (0 V - 3.6 V), Discrete silicon die	1,2,3
C_{OUT}	Output Capacitance	Data, WAIT	2	4	5	pF		

Notes:
1. Capacitance values are for a single die; for dual die, the capacitance values are doubled.
2. Sampled, not 100% tested.
3. Silicon die capacitance only, add 1 pF for discrete packages.

15.3 AC Read Specifications

Table 23: AC Read Specifications - (Sheet 1 of 3)

Num	Symbol	Parameter		Min	Max	Unit	Notes
Asynchronous Specifications							
R1	t_{AVAV}	Read cycle time	Easy BGA	95	-	ns	-
			TSOP	105		ns	-
R2	t_{AVQV}	Address to output valid	Easy BGA	-	95	ns	-
			TSOP		105	ns	-
R3	t_{ELQV}	CE# low to output valid	Easy BGA	-	95	ns	-
			TSOP		105	ns	-
R4	t_{GLQV}	OE# low to output valid		-	25	ns	1,2
R5	t_{PHQV}	RST# high to output valid		-	150	ns	1
R6	t_{ELQX}	CE# low to output in low-Z		0	-	ns	1,3
R7	t_{GLQX}	OE# low to output in low-Z		0	-	ns	1,2,3
R8	t_{EHQZ}	CE# high to output in high-Z		-	20	ns	1,3
R9	t_{GHQZ}	OE# high to output in high-Z		-	15	ns	
R10	t_{OH}	Output hold from first occurring address, CE#, or OE# change		0	-	ns	
R11	t_{EHEL}	CE# pulse width high		17	-	ns	1
R12	t_{ELTV}	CE# low to WAIT valid		-	17	ns	
R13	t_{EHTZ}	CE# high to WAIT high-Z		-	20	ns	1,3
R15	t_{GLTV}	OE# low to WAIT valid		-	17	ns	1
R16	t_{GLTX}	OE# low to WAIT in low-Z		0	-	ns	1,3
R17	t_{GHTZ}	OE# high to WAIT in high-Z		-	20	ns	
Latching Specifications							

Table 23: AC Read Specifications - (Sheet 2 of 3)

Num	Symbol	Parameter		Min	Max	Unit	Notes
R101	t_{AVVH}	Address setup to ADV# high		10	-	ns	1
R102	t_{ELVH}	CE# low to ADV# high		10	-	ns	
R103	t_{VLQV}	ADV# low to output valid	Easy BGA	-	95	ns	
			TSOP		105	ns	
R104	t_{VLVH}	ADV# pulse width low		10	-	ns	
R105	t_{VHVL}	ADV# pulse width high		10	-	ns	
R106	t_{VHAX}	Address hold from ADV# high		9	-	ns	1,4
R108	t_{APA}	Page address access		-	25	ns	1
R111	t_{phvh}	RST# high to ADV# high		30	-	ns	
Clock Specifications							
R200	f_{CLK}	CLK frequency		-	52	MHz	1,3,5,6
R201	t_{CLK}	CLK period		19.2	-	MHz	
R202	$t_{CH/CL}$	CLK high/low time		5	-	ns	
R203	$t_{FCLK/RCLK}$	CLK fall/rise time		0.3	3	ns	

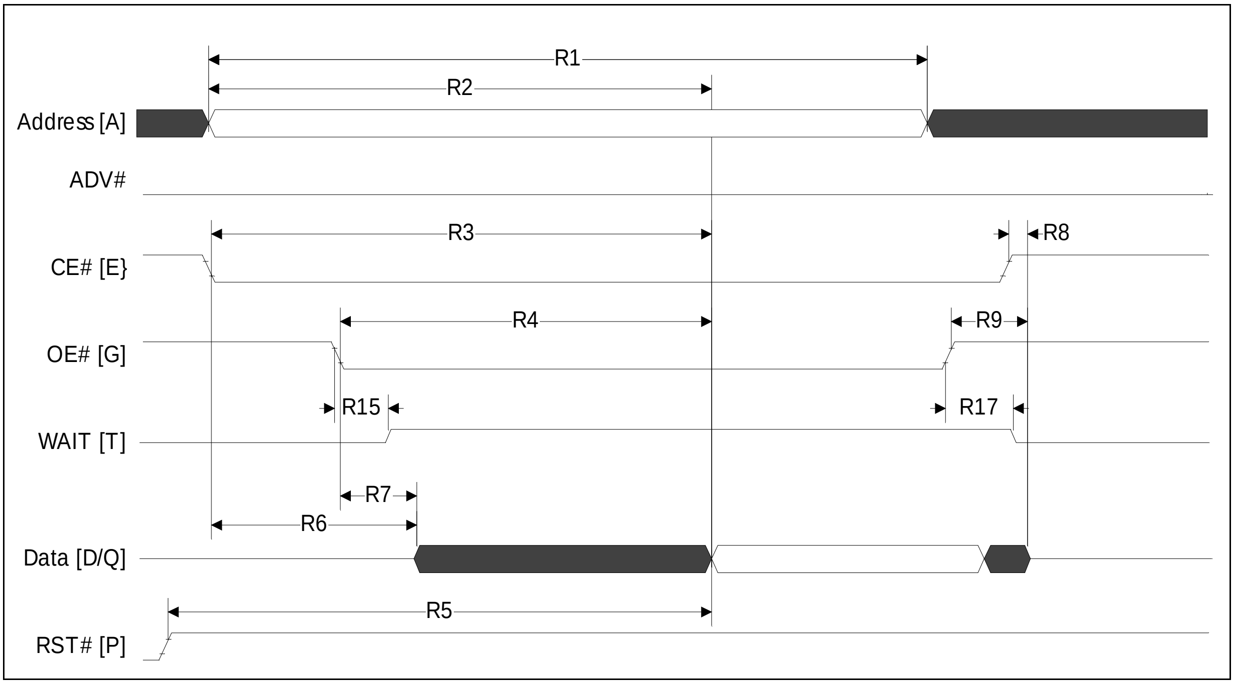
Table 23: AC Read Specifications - (Sheet 3 of 3)

Num	Symbol	Parameter	Min	Max	Unit	Notes
Synchronous Specifications$^{(5)}$						
R301	$t_{AVCH/L}$	Address setup to CLK	9	-	ns	1,6
R302	$t_{VLCH/L}$	ADV# low setup to CLK	9	-	ns	
R303	$t_{ELCH/L}$	CE# low setup to CLK	9	-	ns	
R304	t_{CHQV} / t_{CLQV}	CLK to output valid	-	17	ns	
R305	t_{CHQX}	Output hold from CLK	3	-	ns	1,6
R306	t_{CHAX}	Address hold from CLK	10	-	ns	1,4,6
R307	t_{CHTV}	CLK to WAIT valid	-	17	ns	1,6
R311	t_{CHVL}	CLK Valid to ADV# Setup	3	-	ns	1
R312	t_{CHTX}	WAIT Hold from CLK	3	-	ns	1,6

Notes:

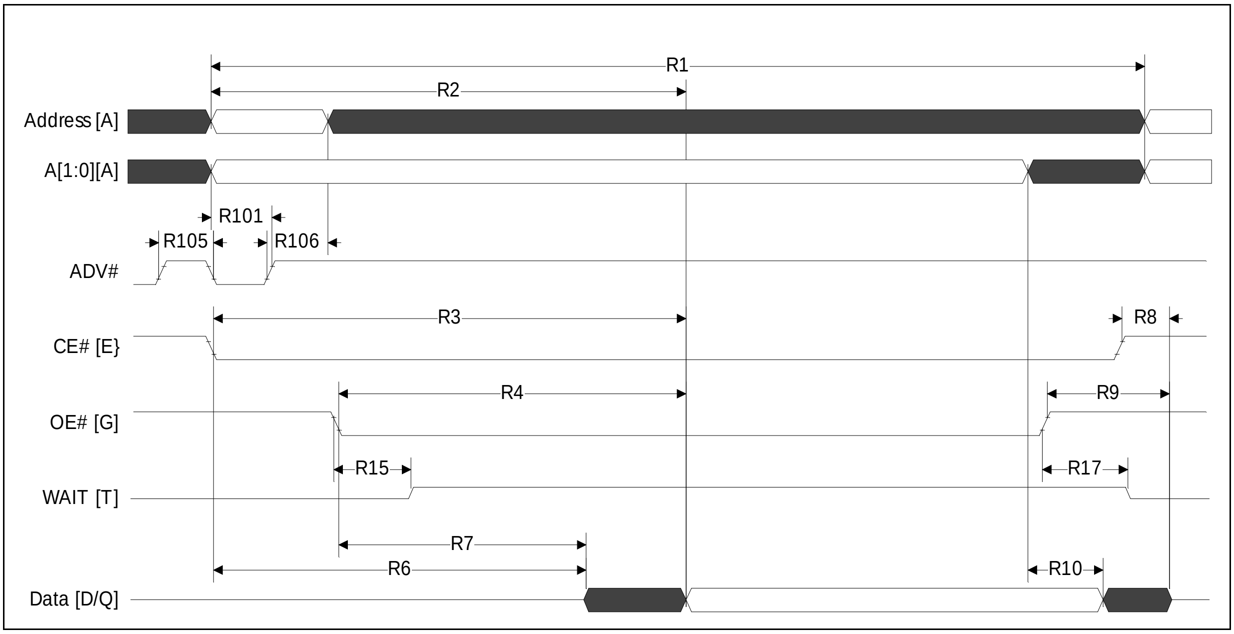
1. See Figure 14, “AC Input/Output Reference Waveform” on page 48 for timing measurements and max allowable input slew rate.
2. OE# may be delayed by up to t_{ELQV} – t_{GLQV} after CE#’s falling edge without impact to t_{ELQV}.
3. Sampled, not 100% tested.
4. Address hold in synchronous burst read mode is t_{CHAX} or t_{VHAX}, whichever timing specification is satisfied first.
5. Synchronous burst read mode is not supported with TTL level inputs.
6. Applies only to subsequent synchronous reads.

Figure 17: Asynchronous Single-Word Read (ADV# Low)



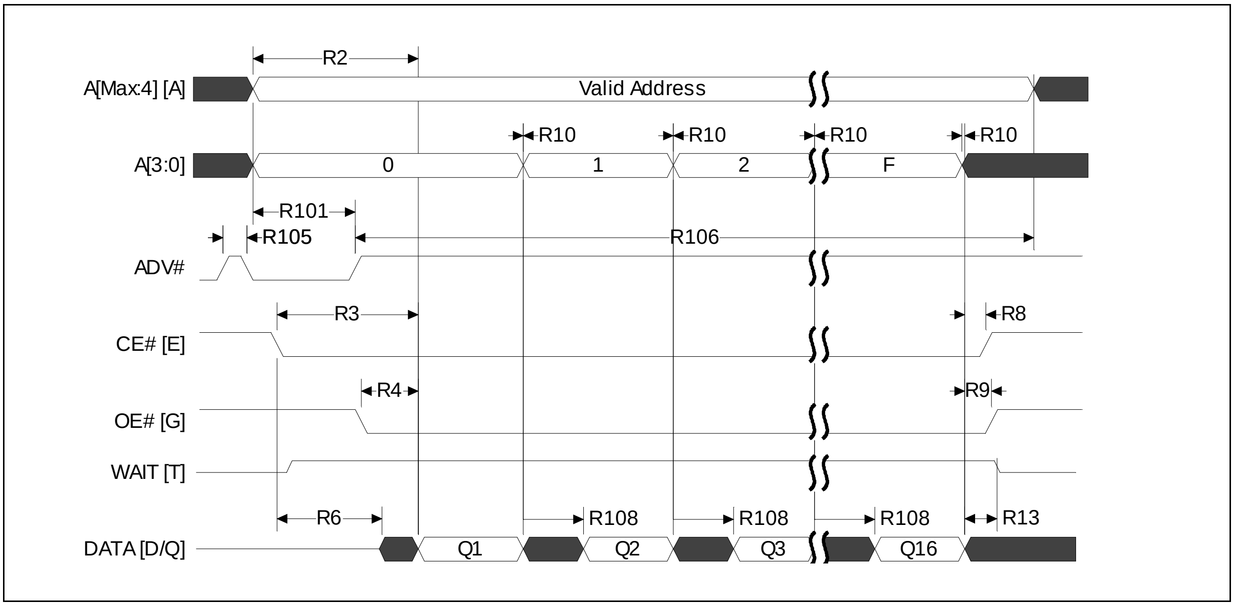
Note: WAIT shown deasserted during asynchronous read mode (RCR.10=0, WAIT asserted low).

Figure 18: Asynchronous Single-Word Read (ADV# Latch)



Note: WAIT shown deasserted during asynchronous read mode (RCR.10=0, WAIT asserted low)

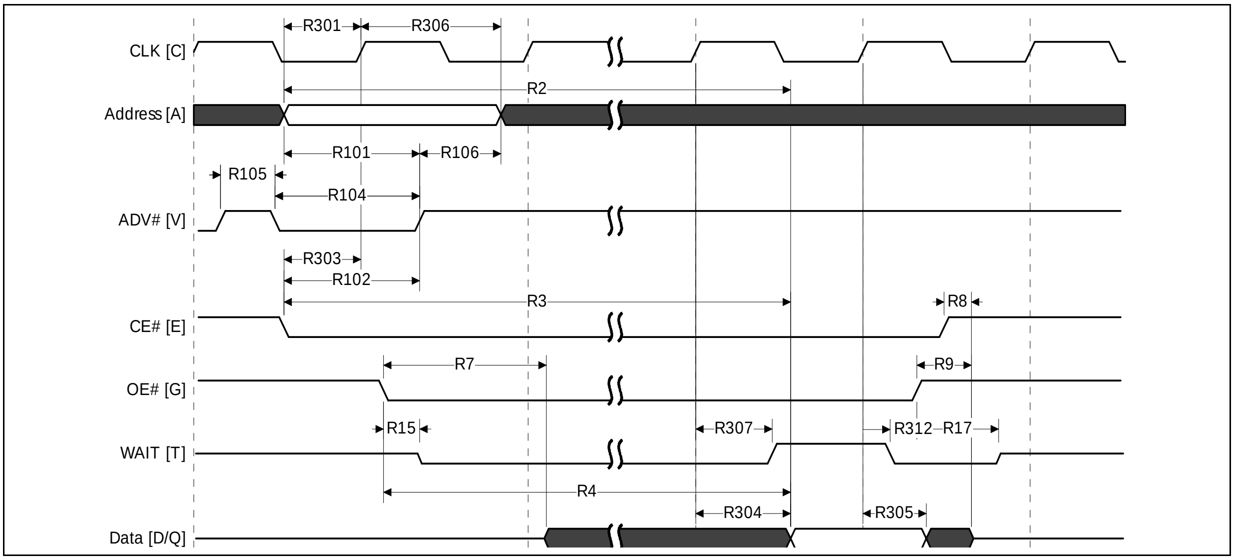
Figure 19: Asynchronous Page-Mode Read Timing



Note: WAIT shown deasserted during asynchronous read mode (RCR.10=0, WAIT asserted low).

.

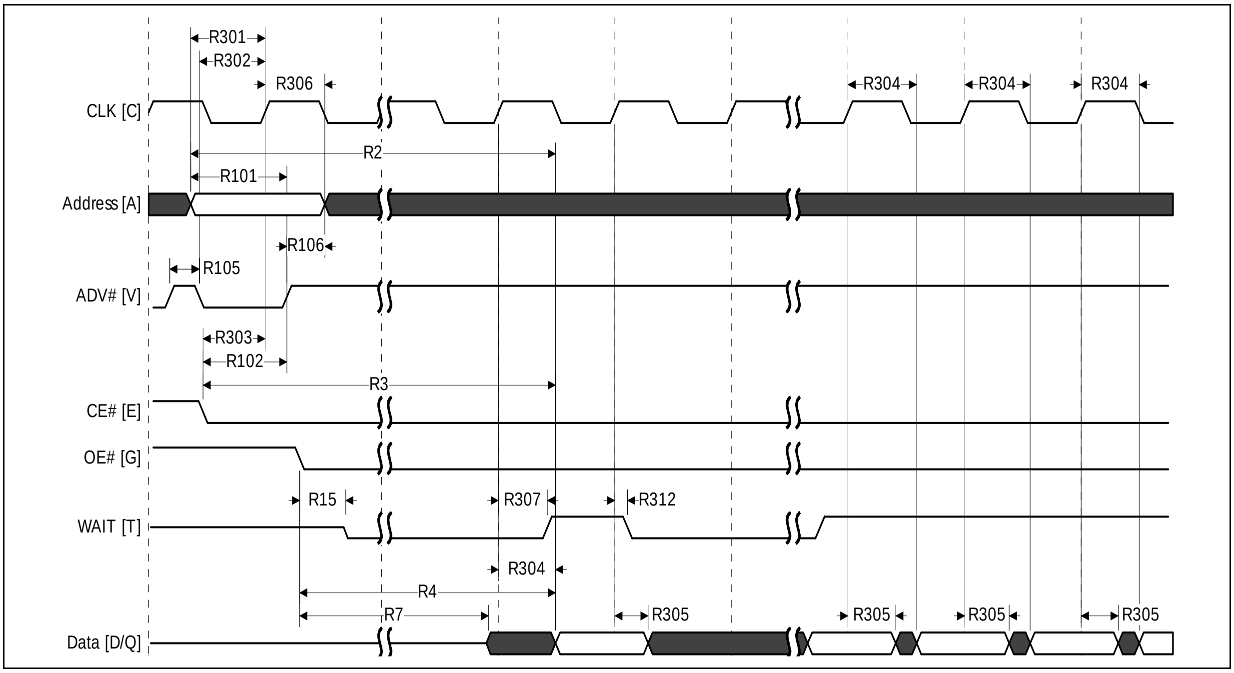
Figure 20: Synchronous Single-Word Array or Non-array Read Timing



Notes:

1. WAIT is driven per OE# assertion during synchronous array or non-array read, and can be configured to assert either during or one data cycle before valid data.
2. This diagram illustrates the case in which an n-word burst is initiated to the flash memory array and it is terminated by CE# deassertion after the first word in the burst.

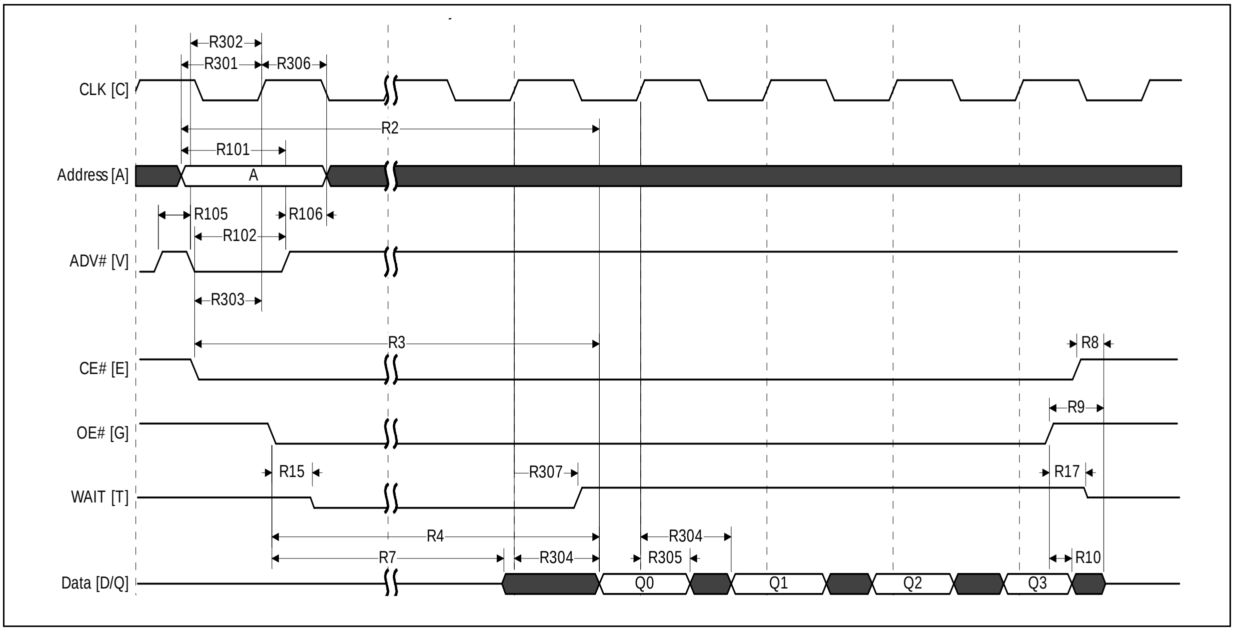
Figure 21: Continuous Burst Read, showing an Output Delay Timing



Notes:

1. WAIT is driven per OE# assertion during synchronous array or non-array read, and can be configured to assert either during or one data cycle before valid data.
2. At the end of Word Line; the delay incurred when a burst access crosses a 16-word boundary and the starting address is not 4-word boundary aligned. See Section 11.1.3, “End of Word Line (EOWL) Considerations” on page 37 for more information

Figure 22: Synchronous Burst-Mode Four-Word Read Timing



Note: WAIT is driven per OE# assertion during synchronous array or non-array read. WAIT asserted during initial latency and deasserted during valid data (RCR.10=0, WAIT asserted low).

15.4 AC Write Specifications

Table 24: AC Write Specifications (Sheet 1 of 2)

Num	Symbol	Parameter	Min	Max	Unit	Notes
W1	t_{PHWL}	RST# high recovery to WE# low	150	-	ns	1,2,3
W2	t_{ELWL}	CE# setup to WE# low	0	-	ns	1,2,3
W3	t_{WLWH}	WE# write pulse width low	50	-	ns	1,2,4
W4	t_{DVWH}	Data setup to WE# high	50	-	ns	1,2
W5	t_{AVWH}	Address setup to WE# high	50	-	ns	
W6	t_{WHEH}	CE# hold from WE# high	0	-	ns	
W7	t_{WHDX}	Data hold from WE# high	0	-	ns	
W8	t_{WHAX}	Address hold from WE# high	0	-	ns	
W9	t_{WHWL}	WE# pulse width high	20	-	ns	1,2,5
W10	t_{VPWH}	VPP setup to WE# high	200	-	ns	1,2,3,7
W11	t_{QVVL}	VPP hold from Status read	0	-	ns	
W12	t_{QVBL}	WP# hold from Status read	0	-	ns	1,2,3,7
W13	t_{BHWH}	WP# setup to WE# high	200	-	ns	
W14	t_{WHGL}	WE# high to OE# low	0	-	ns	1,2,9
W16	t_{WHQV}	WE# high to read valid	t_{AVQV} + 35	-	ns	1,2,3,6,10
Write to Asynchronous Read Specifications						
W18	t_{WHAV}	WE# high to Address valid	0	-	ns	1,2,3,6,8

Table 24: AC Write Specifications (Sheet 2 of 2)

Num	Symbol	Parameter	Min	Max	Unit	Notes
Write to Synchronous Read Specifications						
W19	$t_{WHCH/L}$	WE# high to Clock valid	19	-	ns	1,2,3,6,10
W20	t_{WHVH}	WE# high to ADV# high	19	-	ns	
W28	t_{WHVL}	WE# high to ADV# low	7	-	ns	
Write Specifications with Clock Active						
W21	t_{VHWL}	ADV# high to WE# low	-	20	ns	1,2,3,11
W22	t_{CHWL}	Clock high to WE# low	-	20	ns	

Notes:

1. Write timing characteristics during erase suspend are the same as write-only operations.
2. A write operation can be terminated with either CE# or WE#.
3. Sampled, not 100% tested.
4. Write pulse width low (t_{WLWH} or t_{ELEH}) is defined from CE# or WE# low (whichever occurs last) to CE# or WE# high (whichever occurs first). Hence, $t_{WLWH} = t_{ELEH} = t_{WLEH} = t_{ELWH}$.
5. Write pulse width high (t_{WHWL} or t_{EHEL}) is defined from CE# or WE# high (whichever occurs first) to CE# or WE# low (whichever occurs last). Hence, $t_{WHWL} = t_{EHEL} = t_{WHEL} = t_{EHWL}$).
6. t_{WHVH} or $t_{WHCH/L}$ must be met when transiting from a write cycle to a synchronous burst read.
7. VPP and WP# should be at a valid level until erase or program success is determined.
8. This specification is only applicable when transiting from a write cycle to an asynchronous read. See spec W19 and W20 for synchronous read.
9. When doing a Read Status operation following any command that alters the Status Register, W14 is 20 ns.
10. Add 10 ns if the write operations results in a RCR or block lock status change, for the subsequent read operation to reflect this change.
11. These specs are required only when the device is in a synchronous mode and clock is active during address setup phase.

Figure 23: Write-to-Write Timing

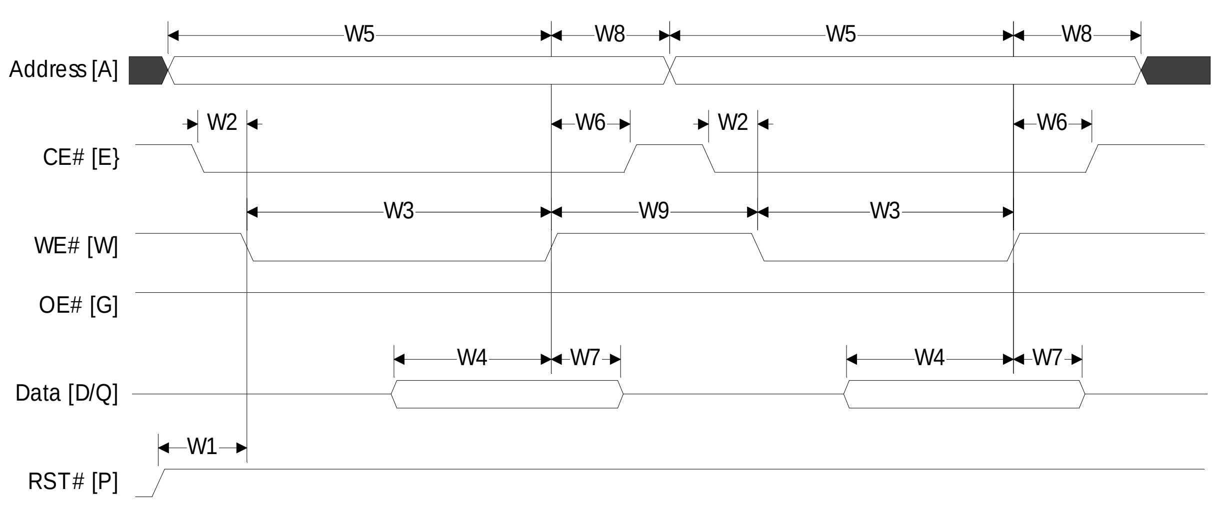
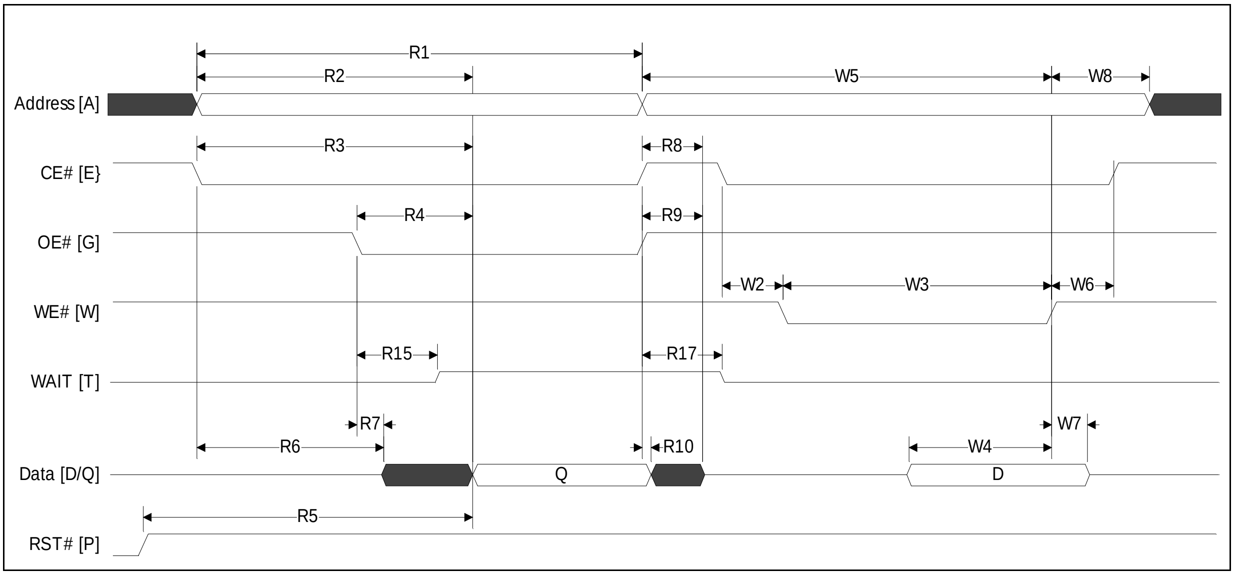


Figure 24: Asynchronous Read-to-Write Timing



Note: WAIT deasserted during asynchronous read and during write. WAIT High-Z during write per OE# deasserted.

Figure 25: Write-to-Asynchronous Read Timing

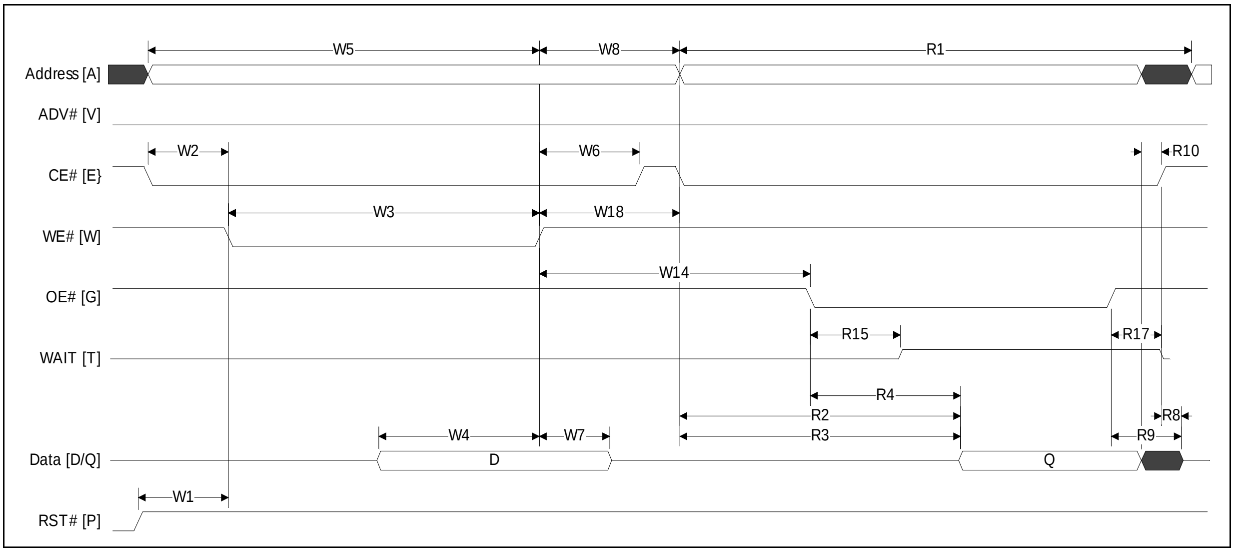
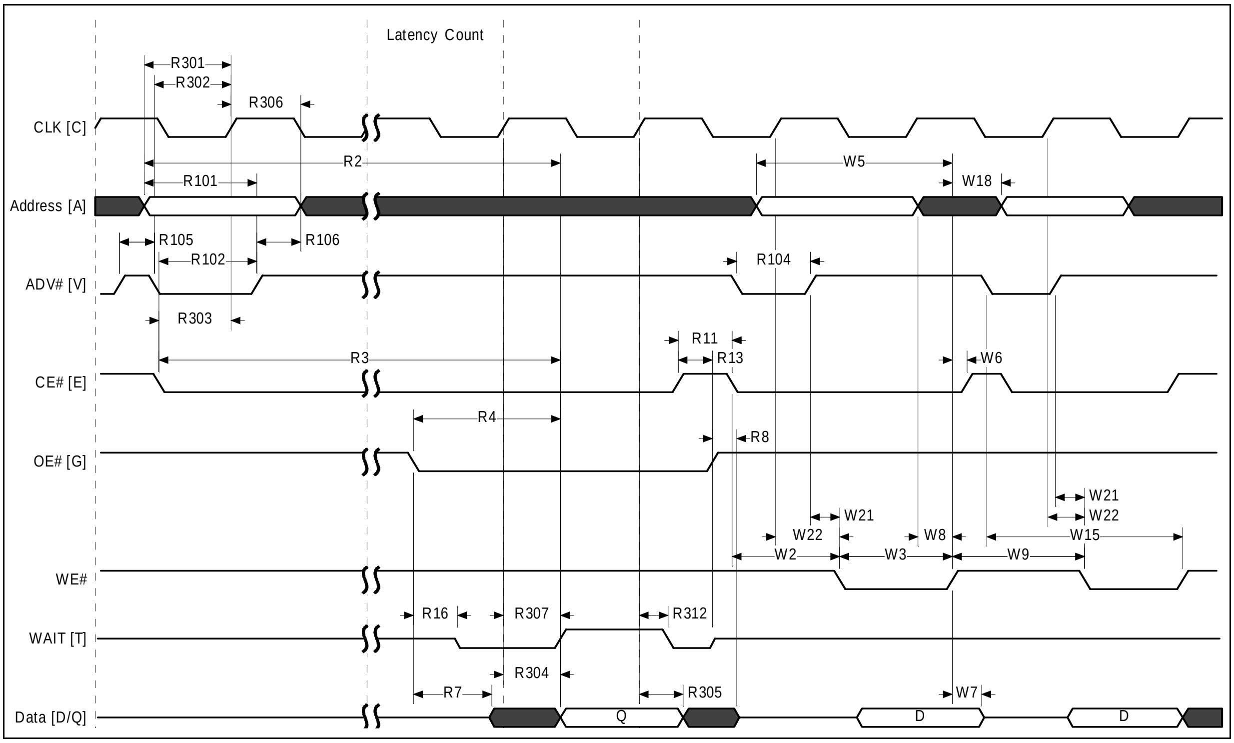
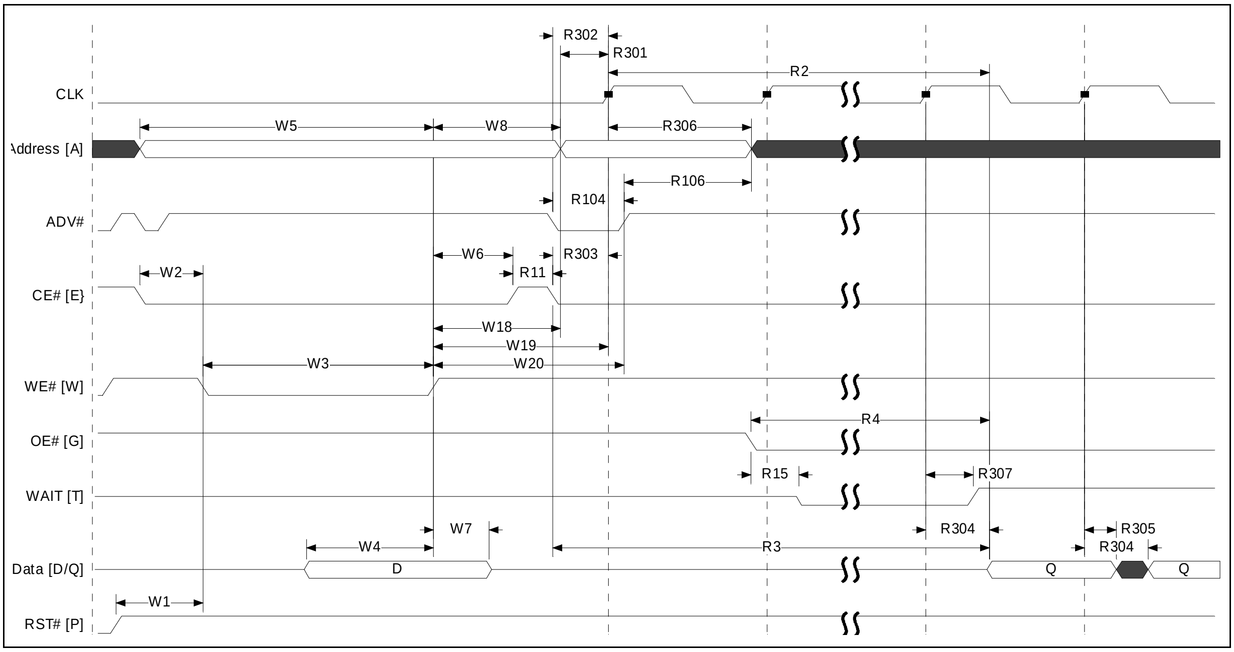


Figure 26: Synchronous Read-to-Write Timing



Note: WAIT shown deasserted and High-Z per OE# deassertion during write operation (RCR 10=0, WAIT asserted low). Clock is ignored during write operation.

Figure 27: Write-to-Synchronous Read Timing



Note: WAIT shown deasserted and High-Z per OE# deassertion during write operation (RCR.10=0, WAIT asserted low).

15.5 Program and Erase Characteristics

Table 25: Program and Erase Specifications

Num	Symbol	Parameter		V_{PPL}			V_{PPH}			Unit	Note
				Min	Typ	Max	Min	Typ	Max		
Conventional Word Programming											
W200	$t_{PROG/W}$	Program Time	Single word	-	150	456	-	150	456	µs	1
Buffered Programming											
W250	t_{PROG}	Program Time	Aligned 32-Wd, BP time (32 Words)	-	176	716	-	176	716	µs	1
			Aligned 64-Wd, BP time (64 Word)	-	216	900	-	216	900		
			Aligned 128-Wd, BP time (128 Words)	-	272	1140	-	272	1140		
			Aligned 256-Wd, BP time (256 Words)	-	396	1690	-	396	1690		
			one full buffer (512 Words)	-	700	3016	-	700	3016		
Buffered Enhanced Factory Programming											
W451	$t_{BEFP/B}$	Program	Single byte	n/a	n/a	n/a	-	0.5	-	µs	1,2
W452	$t_{BEFP/Setup}$		BEFP Setup	n/a	n/a	n/a	5	-	-		1
Erase and Suspend											
W500	$t_{ERS/PB}$	Erase Time	32-KByte Parameter	-	0.8	4.0	-	0.8	4.0	s	1
W501	$t_{ERS/MB}$		128-KByte Main	-	0.8	4.0	-	0.8	4.0		
W600	$t_{SUSP/P}$	Suspend Latency	Program suspend	-	20	25	-	20	25	µs	
W601	$t_{SUSP/E}$		Erase suspend	-	20	25	-	20	25		
W602	$t_{ERS/SUSP}$		Erase to Suspend	-	500	-	-	500	-		1,3
blank check											
W702	$t_{BC/MB}$	blank check	Main Array Block	-	3.2	-	-	3.2	-	ms	

Notes:
1. Typical values measured at T_C = +25 °C and nominal voltages. Performance numbers are valid for all speed versions. Excludes system overhead. Sampled, but not 100% tested.
2. Averaged over entire device.
3. W602 is the typical time between an initial block erase or erase resume command and the a subsequent erase suspend command. Violating the specification repeatedly during any particular block erase may cause erase failures.

16.0 Ordering Information

16.1 Discrete Products

Figure 28: Decoder for Discrete Products

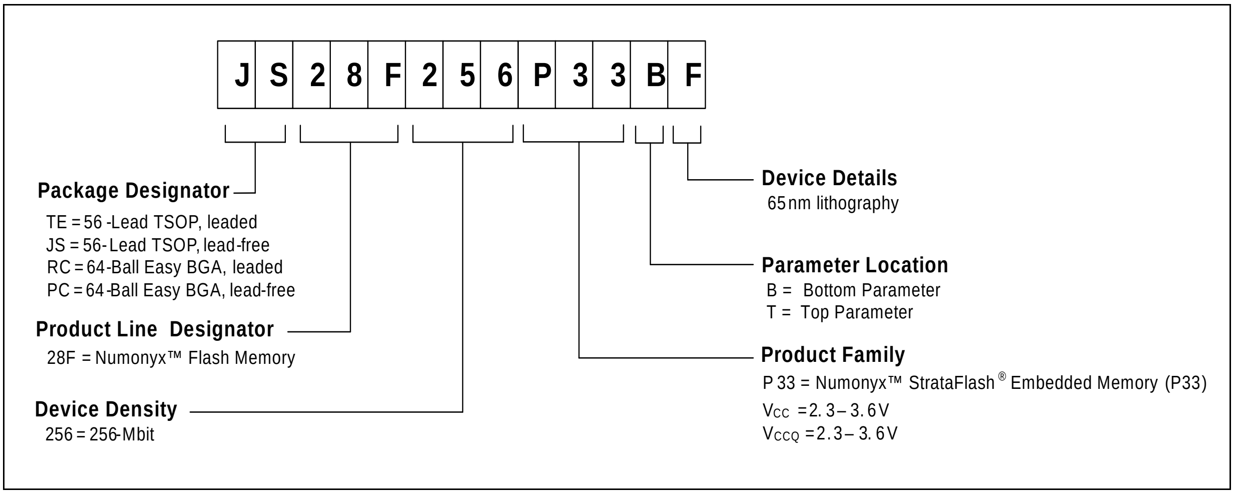


Table 26: Valid Combinations for Discrete Products

256-Mbit
RC28F256P33TF
RC28F256P33BF
PC28F256P33TF
PC28F256P33BF
TE28F256P33TF
TE28F256P33BF
JS28F256P33TF
JS28F256P33BF

16.2 SCSP Products

Figure 29: Decoder for SCSP Devices

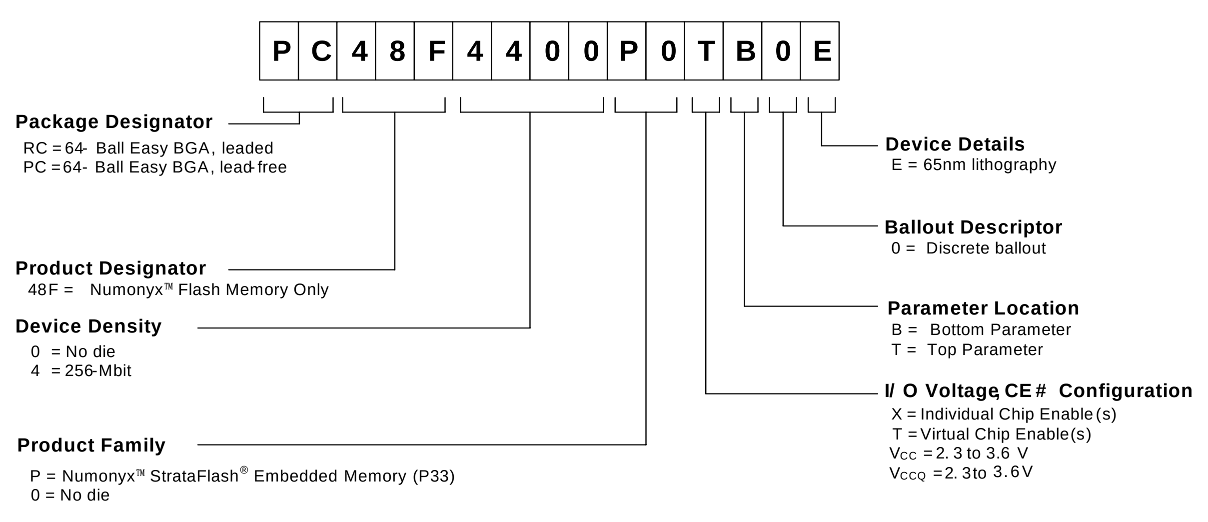


Table 27: Valid Combinations for Dual- Die Products

512-Mbit*
RC48F4400P0TB0E
PC48F4400P0TB0E

Note: The “B” parameter is used for Top(Die1)/Bot(Die2) stack option in the 512-Mbit density.

Appendix A Supplemental Reference Information

A.1 Common Flash Interface

The Common Flash Interface (CFI) is part of an overall specification for multiple command-set and control-interface descriptions. This appendix describes the database structure containing the data returned by a read operation after issuing the Read CFI command (see Section 6.2, “Device Command Bus Cycles” on page 18). System software can parse this database structure to obtain information about the flash device, such as block size, density, bus width, and electrical specifications. The system software will then know which command set(s) to use to properly perform flash writes, block erases, reads and otherwise control the flash device.

A.1.1 Query Structure Output

The Query database allows system software to obtain information for controlling the flash device. This section describes the device’s CFI-compliant interface that allows access to Query data.

Query data are presented on the lowest-order data outputs (DQ_{7-0}) only. The numerical offset value is the address relative to the maximum bus width supported by the device. On this family of devices, the Query table device starting address is a 10h, which is a word address for x16 devices.

For a word-wide (x16) device, the first two Query-structure bytes, ASCII “Q” and “R,” appear on the low byte at word addresses 10h and 11h. This CFI-compliant device outputs 00h data on upper bytes. The device outputs ASCII “Q” in the low byte (DQ_{7-0}) and 00h in the high byte (DQ_{15-8}).

At Query addresses containing two or more bytes of information, the least significant data byte is presented at the lower address, and the most significant data byte is presented at the higher address.

In all of the following tables, addresses and data are represented in hexadecimal notation, so the “h” suffix has been dropped. In addition, since the upper byte of word-wide devices is always “00h,” the leading “00” has been dropped from the table notation and only the lower byte value is shown. Any x16 device outputs have 00h on the upper byte in this mode.

Table 28: Summary of Query Structure Output as a Function of Device and Mode

Device	Hex Offset	Hex Code	ASCII Value
Device Addresses	00010: 00011: 00012:	51 52 59	"Q" "R" "Y"

Table 29: Example of Query Structure Output of x16 Devices

Offset	Hex Code	Value
A_X-A_1	D_{15}-D_0	
00010h	0051	"Q"
00011h	0052	"R"
00012h	0059	"Y"
00013h	P_ID_{LO}	PrVendor ID#
00014h	P_ID_{HI}	
00015h	P_{LO}	PrVendor TblAdr
00016h	P_{HI}	
00017h	A_ID_{LO}	AltVendor ID#
00018h	A_ID_{HI}	
...	...	...

A.1.2 Query Structure Overview

The Query command causes the flash component to display the Common Flash Interface (CFI) Query structure or *database*. Table 30 summarizes the structure sub-sections and address locations.

Table 30: Query Structure

00001-Fh	Reserved	Reserved for vendor-specific information
00010h	CFI query identification string	Command set ID and vendor data offset
0001Bh	System interface information	Device timing & voltage information
00027h	Device geometry definition	Flash device layout
$P^{(3)}$	Primary Numonyx-specific Extended Query	Vendor-defined additional information specific to the Primary Vendor Algorithm

Note:

1. Refer to the Query Structure Output section and offset 28h for the detailed definition of offset address as a function of device bus width and mode.
2. BA = Block Address beginning location (i.e., 08000h is block 1's beginning location when the block size is 32-KWord).
3. Offset 15 defines "P" which points to the Primary Numonyx-specific Extended Query Table.

A.1.3 Read CFI Identification String

The Identification String provides verification that the component supports the Common Flash Interface specification. It also indicates the specification version and supported vendor-specified command set(s).

Table 31: CFI Identification

Offset	Length	Description	Add.	Hex Code	Value
10h	3	Query-unique ASCII string “QRY“	10: 11: 12:	--51 --52 --59	"Q" "R" "Y"
13h	2	Primary vendor command set and control interface ID code. 16-bit ID code for vendor-specified algorithms	13: 14:	--01 --00	
15h	2	Extended Query Table primary algorithm address	15: 16:	--0A --01	
17h	2	Alternate vendor command set and control interface ID code. 0000h means no second vendor-specified algorithm exists	17: 18:	--00 --00	
19h	2	Secondary algorithm Extended Query Table address. 0000h means none exists	19: 1A:	--00 --00	

Figure 30: System Interface Information

Offset	Length	Description	Add.	Hex Code	Value
1Bh	1	V_{CC} logic supply minimum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 BCD volts	1B:	--23	2.3V
1Ch	1	V_{CC} logic supply maximum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 BCD volts	1C:	--36	3.6V
1Dh	1	V_{PP} [programming] supply minimum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 HEX volts	1D:	--85	8.5V
1Eh	1	V_{PP} [programming] supply maximum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 HEX volts	1E:	--95	9.5V
1Fh	1	“n” such that typical single word program time-out = 2^n µ-sec	1F:	--08	256µs
20h	1	“n” such that typical full buffer write time-out = 2^n µ-sec	20:	--0A	1024µs
21h	1	“n” such that typical block erase time-out = 2^n m-sec	21:	--0A	1s
22h	1	“n” such that typical full chip erase time-out = 2^n m-sec	22:	--00	NA
23h	1	“n” such that maximum word program time-out = 2^n times typical	23:	--01	512µs
24h	1	“n” such that maximum buffer write time-out = 2^n times typical	24:	--02	4096µs
25h	1	“n” such that maximum block erase time-out = 2^n times typical	25:	--02	4s
26h	1	“n” such that maximum chip erase time-out = 2^n times typical	26:	--00	NA

A.1.4 Device Geometry Definition

Figure 31: Device Geometry Definition

<table>
<tr><th>Offset</th><th>Length</th><th>Description</th><th>Add.</th><th>Hex Code</th><th>Value</th></tr>
<tr><td>27h</td><td>1</td><td>“n” such that device size = 2^n in number of bytes</td><td>27:</td><td colspan="2">See table below</td></tr>
<tr><td rowspan="2">28h</td><td rowspan="2">2</td><td rowspan="2">Flash device interface code assignment:
"n" such that n+1 specifies the bit field that represents the flash device width capabilities as described in the table:
<table>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>—</td><td>—</td><td>—</td><td>—</td><td>x64</td><td>x32</td><td>x16</td><td>x8</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td></tr>
</table></td><td>28:</td><td>--01</td><td rowspan="2">x16</td></tr>
<tr><td>29:</td><td>--00</td></tr>
<tr><td rowspan="2">2Ah</td><td rowspan="2">2</td><td rowspan="2">“n” such that maximum number of bytes in write buffer = 2^n</td><td>2A:</td><td>--0A</td><td rowspan="2">1024</td></tr>
<tr><td>2B:</td><td>--00</td></tr>
<tr><td>2Ch</td><td>1</td><td>Number of erase block regions (x) within device:
1. x = 0 means no erase blocking; the device erases in bulk
2. x specifies the number of device regions with one or more contiguous same-size erase blocks.
3. Symmetrically blocked partit</td><td>2C:</td><td colspan="2">See table below</td></tr>
<tr><td>2Dh</td><td>4</td><td>Erase Block Region 1 Information
bits 0–15 = y, y+1 = number of identical-size erase blocks
bits 16–31 = z, region erase block(s) size are z x 256 bytes</td><td>2D:
2E:
2F:
30:</td><td colspan="2">See table below</td></tr>
<tr><td>31h</td><td>4</td><td>Erase Block Region 2 Information
bits 0–15 = y, y+1 = number of identical-size erase blocks
bits 16–31 = z, region erase block(s) size are z x 256 bytes</td><td>31:
32:
33:
34:</td><td colspan="2">See table below</td></tr>
<tr><td>35h</td><td>4</td><td>Reserved for future erase block region information</td><td>35:
36:
37:
38:</td><td colspan="2">See table below</td></tr>
</table>

Address	256-Mbit	
	–B	–T
27:	--19	--19
28:	--01	--01
29:	--00	--00
2A:	--06	--06
2B:	--00	--00
2C:	--02	--02
2D:	--03	--FE
2E:	--00	--00
2F:	--80	--00
30:	--00	--02
31:	--FE	--03
32:	--00	--00
33:	--00	--80
34:	--02	--00
35:	--00	--00
36:	--00	--00
37:	--00	--00
38:	--00	--00

A.1.5 Numonyx-Specific Extended Query Table

Table 32: Primary Vendor-Specific Extended Query

Offset(1) P = 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+0)h	3	Primary extended query table	10A	--50	"P"
(P+1)h		Unique ASCII string "PRI"	10B:	--52	"R"
(P+2)h			10C:	--49	"I"
(P+3)h	1	Major version number, ASCII	10D:	--31	"1"
(P+4)h	1	Minor version number, ASCII	10E:	--35	"5"
(P+5)h	4	Optional feature and command support (1=yes, 0=no)	10F:	--E6	
(P+6)h		*bits 10–31 are reserved; undefined bits are "0." If bit 31 is*	110:	--09	
(P+7)h		*"1" then another 31 bit field of Optional features follows at*	111:	--00	
(P+8)h		*the end of the bit–30 field.*	112:	--40	
		bit 0 Chip erase supported	bit 0 = 0		No
		bit 1 Suspend erase supported	bit 1 = 1		Yes
		bit 2 Suspend program supported	bit 2 = 1		Yes
		bit 3 Legacy lock/unlock supported	bit 3 = 0		No
		bit 4 Queued erase supported	bit 4 = 0		No
		bit 5 Instant individual block locking supported	bit 5 = 1		Yes
		bit 6 Protection bits supported	bit 6 = 1		Yes
		bit 7 Pagemode read supported	bit 7 = 1		Yes
		bit 8 Synchronous read supported	bit 8 = 1		Yes
		bit 9 Simultaneous operations supported	bit 9 = 0		No
		bit 10 Extended Flash Array Blocks supported	bit 10 = 0		No
		bit 11 Permanent Block Locking of up to Full Main Array supported	bit 11 = 1		Yes
		bit 12 Permanent Block Locking of up to Partial Main Array supported	bit 12 = 0		No
		bit 30 CFI Link(s) to follow	bit 30 = 1		Yes
		bit 31 Another "Optional Features" field to follow	bit 31 = 0		No
(P+9)h	1	Supported functions after suspend: read Array, Status, Query Other supported operations are: bits 1–7 reserved; undefined bits are "0"	113:	--01	
		bit 0 Program supported after erase suspend	bit 0 = 1		Yes
(P+A)h	2	Block status register mask	114:	--03	
(P+B)h		*bits 2–15 are Reserved; undefined bits are "0"*	115:	--00	
		bit 0 Block Lock-Bit Status register active	bit 0 = 1		Yes
		bit 1 Block Lock-Down Bit Status active	bit 1 = 1		Yes
		bit 4 EFA Block Lock-Bit Status register active	bit 4 = 0		No
		bit 5 EFA Block Lock-Down Bit Status active	bit 5 = 0		No
(P+C)h	1	V_{CC} logic supply highest performance program/erase voltage bits 0–3 BCD value in 100 mV bits 4–7 BCD value in volts	116:	--18	1.8V
(P+D)h	1	V_{PP} optimum program/erase supply voltage bits 0–3 BCD value in 100 mV bits 4–7 HEX value in volts	117:	--90	9.0V

Address	Discrete		512-Mbit			
	–B	–T	–B		–T	
	–-	–-	die 1 (B)	die 2 (T)	die 1 (T)	die 2 (B)
112:	--00	--00	--40	--00	--40	--00

Table 33: OTP Register Information

Offset(1) P = 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+E)h	1	Number of Protection register fields in JEDEC ID space. "00h," indicates that 256 protection fields are available	118:	--02	2
(P+F)h	4	Protection Field 1: Protection Description	119:	--80	80h
(P+10)h		This field describes user-available One Time Programmable (OTP) Protection register bytes. Some are pre-programmed with device-unique serial numbers. Others are user programmable. Bits 0–15 point to the Protection register Lock byte, the section's first byte. The following bytes are factory pre-programmed and user-programmable. bits 0–7 = Lock/bytes Jedec-plane physical low address bits 8–15 = Lock/bytes Jedec-plane physical high address bits 16–23 = "n" such that 2^n = factory pre-programmed bytes bits 24–31 = "n" such that 2^n = user programmable bytes	11A:	--00	00h
(P+11)h			11B:	--03	8 byte
(P+12)h			11C:	--03	8 byte
(P+13)h	10	Protection Field 2: Protection Description	11D:	--89	89h
(P+14)h		Bits 0–31 point to the Protection register physical Lock-word address in the Jedec-plane.	11E:	--00	00h
(P+15)h			11F:	--00	00h
(P+16)h		Following bytes are factory or user-programmable.	120:	--00	00h
(P+17)h		bits 32–39 = "n" such that n = factory pgm'd groups (low byte)	121:	--00	0
(P+18)h		bits 40–47 = "n" such that n = factory pgm'd groups (high byte)	122:	--00	0
(P+19)h		bits 48–55 = "n" \ 2n = factory programmable bytes/group	123:	--00	0
(P+1A)h		bits 56–63 = "n" such that n = user pgm'd groups (low byte)	124:	--10	16
(P+1B)h		bits 64–71 = "n" such that n = user pgm'd groups (high byte)	125:	--00	0
(P+1C)h		bits 72–79 = "n" such that 2^n = user programmable bytes/group	126:	--04	16

Figure 32: Burst Read Information

Offset(1) P = 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+1D)h	1	Page Mode Read capability bits 0–7 = “n” such that 2^n HEX value represents the number of read-page bytes. See offset 28h for device word width to determine page-mode data output width. 00h indicates no read page buffer.	127:	--05	32 byte
(P+1E)h	1	Number of synchronous mode read configuration fields that follow. 00h indicates no burst capability.	128:	--04	4
(P+1F)h	1	Synchronous mode read capability configuration 1 Bits 3–7 = Reserved Bits 0–2 “n” such that 2n+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 bits 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.	129:	--01	4
(P+20)h	1	Synchronous mode read capability configuration 2	12A:	--02	8
(P+21)h	1	Synchronous mode read capability configuration 3	12B:	--03	16
(P+22)h	1	Synchronous mode read capability configuration 4	12C:	--07	Cont

Table 34: Partition and Erase Block Region Information

Offset(1) P = 10Ah		Description (Optional flash features and commands)	See table below		
				Address	
Bottom	Top		Len	Bot	Top
(P+23)h	(P+23)h	Number of device hardware-partition regions within the device. x = 0: a single hardware partition device (no fields follow). x specifies the number of device partition regions containing one or more contiguous erase block regions.	1	12D:	12D:

Table 35: Partition Region 1 Information (Sheet 1 of 2)

Offset[(1)]			**See table below**		
P = 10Ah		**Description**		**Address**	
Bottom	**Top**	**(Optional flash features and commands)**	**Len**	**Bot**	**Top**
(P+24)h	(P+24)h	Data size of this Parition Region Information field	2	12E:	12E
(P+25)h	(P+25)h	(# addressable locations, including this field)		12F	12F
(P+26)h	(P+26)h	Number of identical partitions within the partition region	2	130:	130:
(P+27)h	(P+27)h			131:	131:
(P+28)h	(P+28)h	Number of program or erase operations allowed in a partition bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	132:	132:
(P+29)h	(P+29)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Program mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	133:	133:
(P+2A)h	(P+2A)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Erase mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	134:	134:
(P+2B)h	(P+2B)h	Types of erase block regions in this Partition Region. x = 0 = no erase blocking; the Partition Region erases in bulk x = number of erase block regions w/ contiguous same-size erase blocks. Symmetrically blocked partitions have one blocking region. Partition size = (Type 1 blocks)x(Type 1 block sizes) + (Type 2 blocks)x(Type 2 block sizes) +…+ (Type n blocks)x(Type n block sizes)	1	135:	135:

Table 36: Partition Region 1 Information (Sheet 2 of 2)

Offset(1)		Description	See table below		
P = 10Ah				Address	
Bottom	Top	(Optional flash features and commands)	Len	Bot	Top
(P+2C)h	(P+2C)h	Partition Region 1 Erase Block Type 1 Information	4	136:	136:
(P+2D)h	(P+2D)h	bits 0–15 = y, y+1 = # identical-size erase blks in a partition		137:	137:
(P+2E)h	(P+2E)h	bits 16–31 = z, region erase block(s) size are z x 256 bytes		138:	138:
(P+2F)h	(P+2F)h			139:	139:
(P+30)h	(P+30)h	Partition 1 (Erase Block Type 1)	2	13A:	13A:
(P+31)h	(P+31)h	Block erase cycles x 1000		13B:	13B:
(P+32)h	(P+32)h	Partition 1 (erase block Type 1) bits per cell; internal EDAC	1	13C:	13C:
		bits 0–3 = bits per cell in erase region			
		bit 4 = internal EDAC used (1=yes, 0=no)			
		bits 5–7 = reserve for future use			
(P+33)h	(P+33)h	Partition 1 (erase block Type 1) page mode and synchronous mode capabilities defined in Table 10.	1	13D:	13D:
		bit 0 = page-mode host reads permitted (1=yes, 0=no)			
		bit 1 = synchronous host reads permitted (1=yes, 0=no)			
		bit 2 = synchronous host writes permitted (1=yes, 0=no)			
		bits 3–7 = reserved for future use			
		Partition Region 1 (Erase Block Type 1) Programming Region Information	6		
(P+34)h	(P+34)h	bits 0–7 = x, 2^x = Programming Region aligned size (**bytes**)		13E:	13E:
(P+35)h	(P+35)h	bits 8–14 = Reserved; bit 15 = Legacy flash operation (ignore 0:7)		13F:	13F:
(P+36)h	(P+36)h	bits 16–23 = y = Control Mode **valid** size in bytes		140:	140:
(P+37)h	(P+37)h	bits 24-31 = Reserved		141:	141:
(P+38)h	(P+38)h	bits 32-39 = z = Control Mode **invalid** size in bytes		142:	142:
(P+39)h	(P+39)h	bits 40-46 = Reserved; bit 47 = Legacy flash operation (ignore 23:16 & 39:32)		143:	143:
(P+3A)h	(P+3A)h	Partition Region 1 Erase Block Type 2 Information	4	144:	144:
(P+3B)h	(P+3B)h	bits 0–15 = y, y+1 = # identical-size erase blks in a partition		145:	145:
(P+3C)h	(P+3C)h	bits 16–31 = z, region erase block(s) size are z x 256 bytes		146:	146:
(P+3D)h	(P+3D)h			147:	147:
(P+3E)h	(P+3E)h	Partition 1 (Erase Block Type 2)	2	148:	148:
(P+3F)h	(P+3F)h	Block erase cycles x 1000		149:	149:
(P+40)h	(P+40)h	Partition 1 (erase block Type 2) bits per cell; internal EDAC	1	14A:	14A:
		bits 0–3 = bits per cell in erase region			
		bit 4 = internal EDAC used (1=yes, 0=no)			
		bits 5–7 = reserve for future use			
(P+41)h	(P+41)h	Partition 1 (erase block Type 2) page mode and synchronous mode capabilities defined in Table 10.	1	14B:	14B:
		bit 0 = page-mode host reads permitted (1=yes, 0=no)			
		bit 1 = synchronous host reads permitted (1=yes, 0=no)			
		bit 2 = synchronous host writes permitte			
		Partition Region 1 (Erase Block Type 2) Programming Region Information	6		
(P+42)h	(P+42)h	bits 0–7 = x, 2^x = Programming Region aligned size (**bytes**)		14C:	14C:
(P+43)h	(P+43)h	bits 8–14 = Reserved; bit 15 = Legacy flash operation (ignore 0:7)		14D:	14D:
(P+44)h	(P+44)h	bits 16–23 = y = Control Mode **valid** size in bytes		14E:	14E:
(P+45)h	(P+45)h	bits 24-31 = Reserved		14F:	14F:
(P+46)h	(P+46)h	bits 32-39 = z = Control Mode **invalid** size in bytes		150:	150:
(P+47)h	(P+47)h	bits 40-46 = Reserved; bit 47 = Legacy flash operation (ignore 23:16 & 39:32)		151:	151:

Table 37: Partition and Erase Block Region Information

Address	256-Mbit	
	–B	–T
12D:	--01	--01
12E:	--24	--24
12F:	--00	--00
130:	--01	--01
131:	--00	--00
132:	--11	--11
133:	--00	--00
134:	--00	--00
135:	--02	--02
136:	--03	--FE
137:	--00	--00
138:	--80	--00
139:	--00	--02
13A	--64	--64
13B:	--00	--00
13C:	--02	--02
13D:	--03	--03
13E:	--00	--00
13F:	--80	--80
140:	--00	--00
141:	--00	--00
142:	--00	--00
143:	--80	--80
144:	--FE	--03
145:	--00	--00
146:	--00	--80
147:	--02	--00
148:	--64	--64
149:	--00	--00
14A	--02	--02
14B:	--03	--03
14C:	--00	--00
14D:	--80	--80
14E:	--00	--00
14F:	--00	--00
150:	--00	--00
151:	--80	--80

Table 38: CFI Link Information

Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
4	CFI Link Field bit definitions	152:	See table below	
	Bits 0–9 = Address offset (within 32Mbit segment) of referenced CFI table	153:		
	Bits 10–27 = nth 32Mbit segment of referenced CFI table	154:		
	Bits 28–30 = Memory Type	155:		
	Bit 31 = Another CFI Link field immediately follows			
1	CFI Link Field Quantity Subfield definitions	156:	See table below	
	Bits 0–3 = Quantity field (n such that n+1 equals quantity)			
	Bit 4 = Table & Die relative location			
	Bit 5 = Link Field & Table relative location			
	Bits 6–7 = Reserved			

Address	Discrete		512-Mbit			
	–B	–T	–B		–T	
	–-	–-	die 1 (B)	die 2 (T)	die 1 (T)	die 2 (B)
152:	--FF	--FF	--10	--FF	--10	--FF
153:	--FF	--FF	--20	--FF	--20	--FF
154:	--FF	--FF	--00	--FF	--00	--FF
155:	--FF	--FF	--00	--FF	--00	--FF
156:	--FF	--FF	--10	--FF	--10	--FF

A.2 Flowcharts

Figure 33: Word Program Flowchart

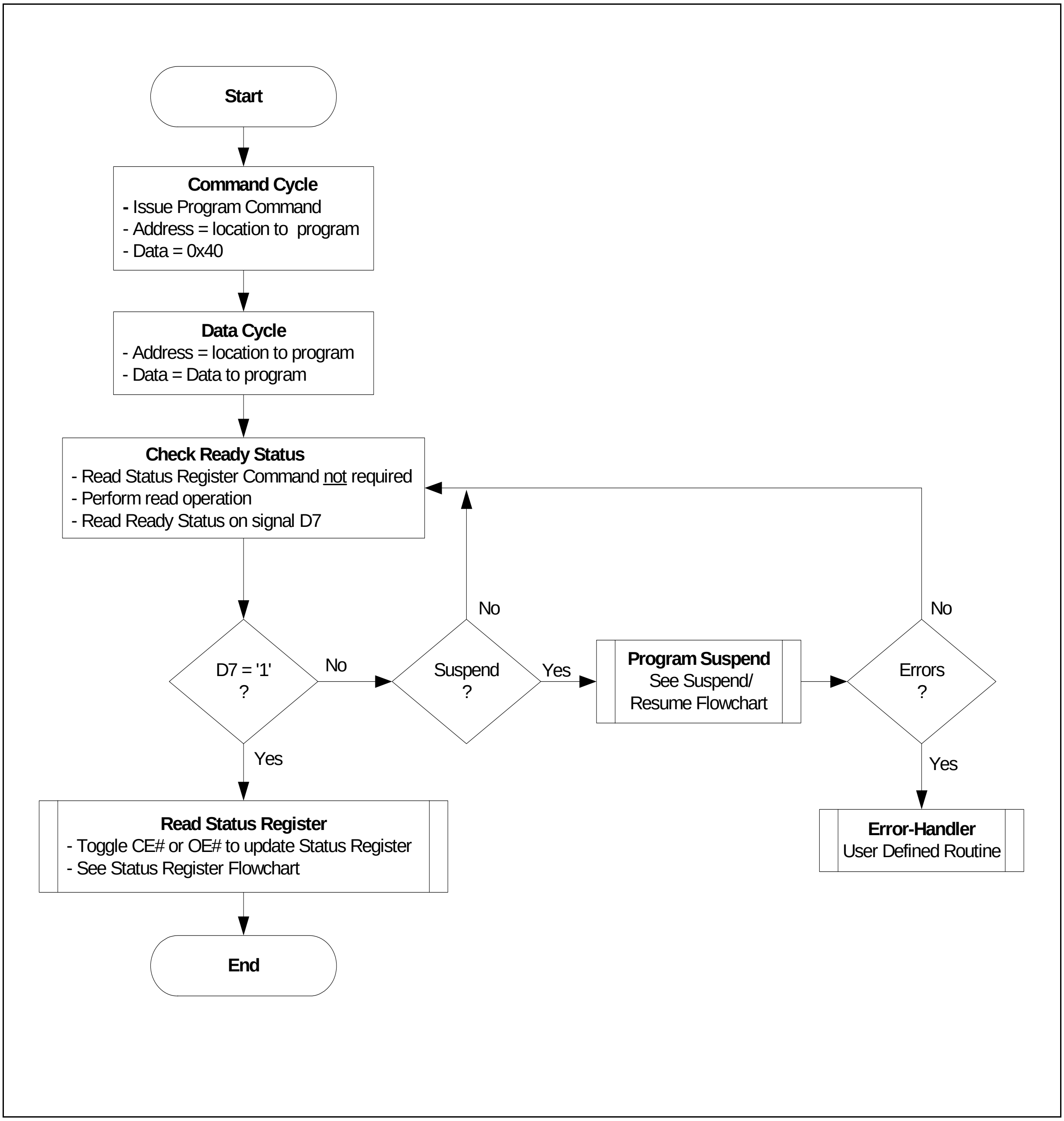


Figure 34: Program / Erase Suspend / Resume Flowchart

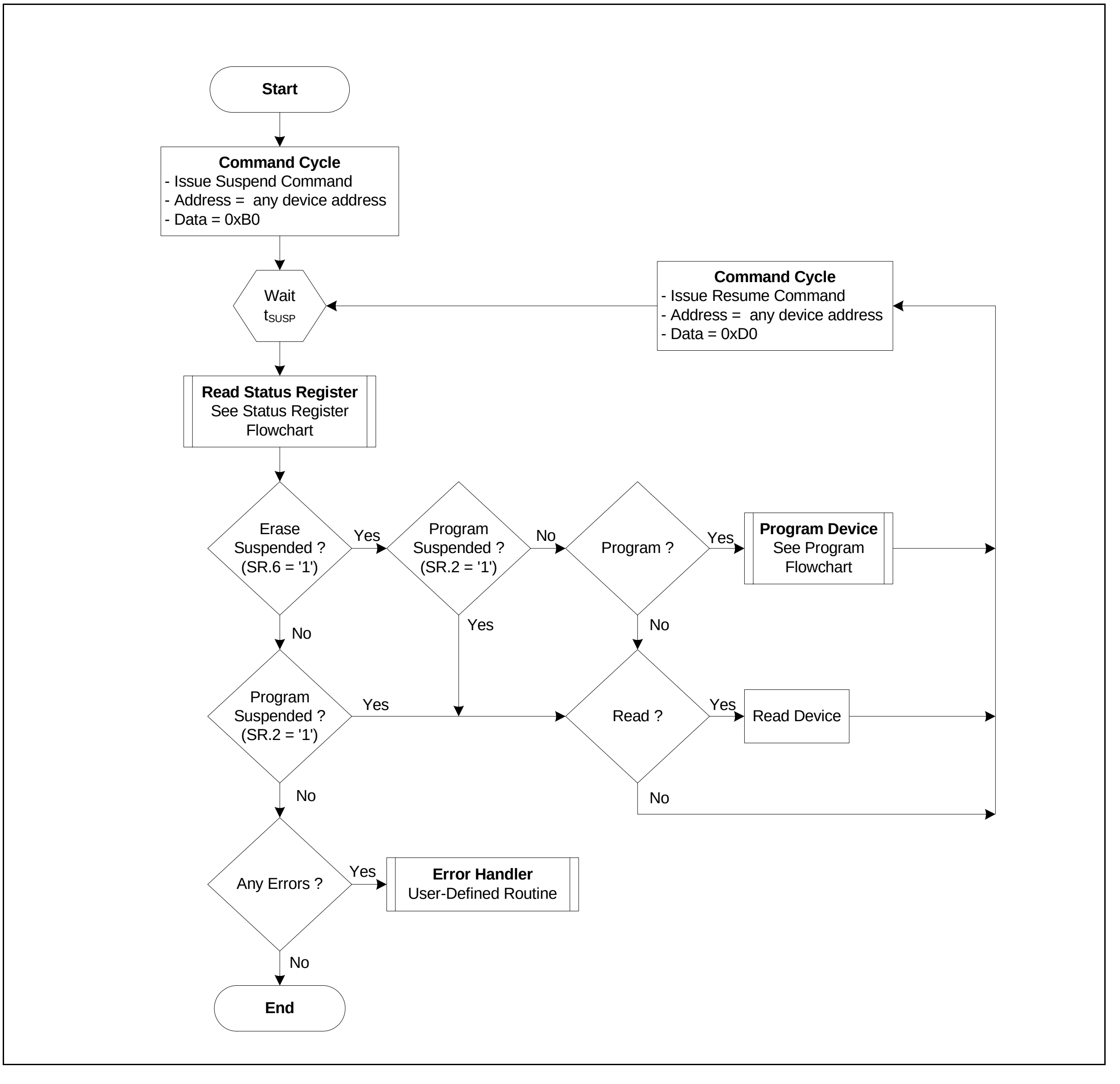
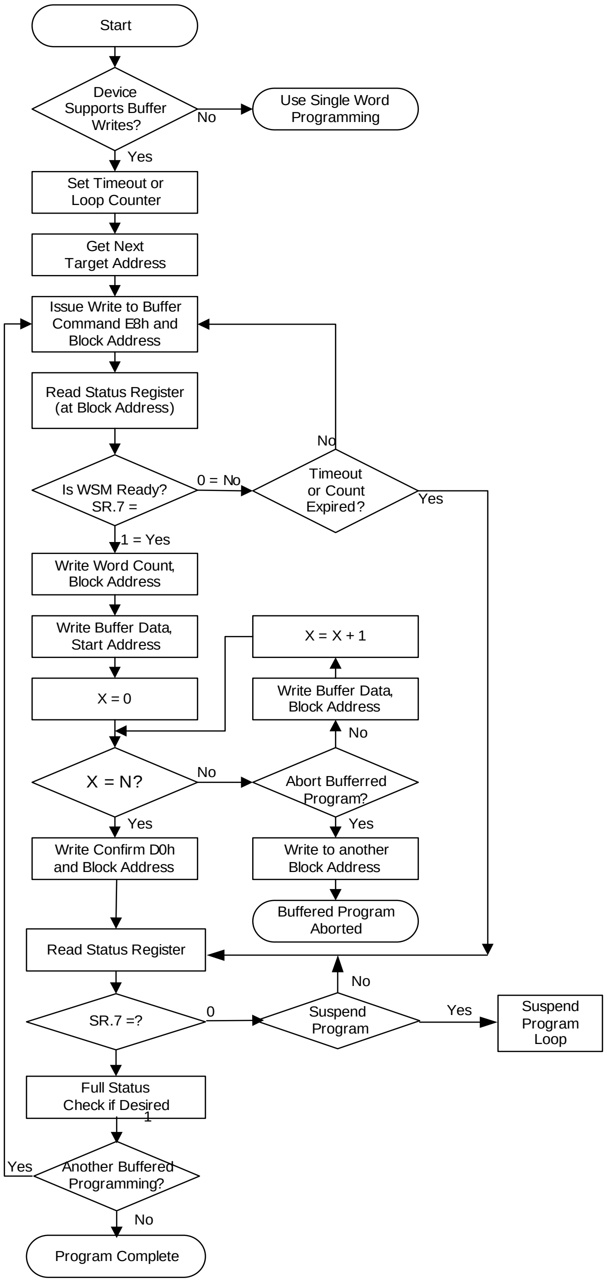


Figure 35: Buffer Program Flowchart



Bus Operation	Command	Comments
Write	Write to Buffer	Data = E8H Addr = Block Address
Read		SR.7 = Valid Addr = Block Address
Standby		Check SR.7 1 = Device WSM is Busy 0 = Device WSM is Ready
Write (Notes 1, 2)		Data = N-1 = Word Count N = 0 corresponds to count = 1 Addr = Block Address
Write (Notes 3, 4)		Data = Write Buffer Data Addr = Start Address
Write (Notes 5, 6)		Data = Write Buffer Data Addr = Block Address
Write	Program Confirm	Data = D0H Addr = Block Address
Read		Status register Data CE# and OE# low updates SR Addr = Block Address
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy

1. Word count values on DQ_0-DQ_7 are loaded into the Count register. Count ranges for this device are N = 0000h to 0001Fh.
2. The device outputs the status register when read.
3. Write Buffer contents will be programmed at the device start address or destination flash address.
4. Align the start address on a Write Buffer boundary for maximum programming performance (i.e., A_4–A_0 of the start address = 0).
5. The device aborts the Buffered Program command if the current address is outside the original block address.
6. The Status register indicates an "improper command sequence" if the Buffered Program command is aborted. Follow this with a Clear Status Register command.

Full status check can be done after all erase and write sequences complete. Write FFh after the last operation to reset the device to read array mode.

Figure 36: BEFP Flowchart

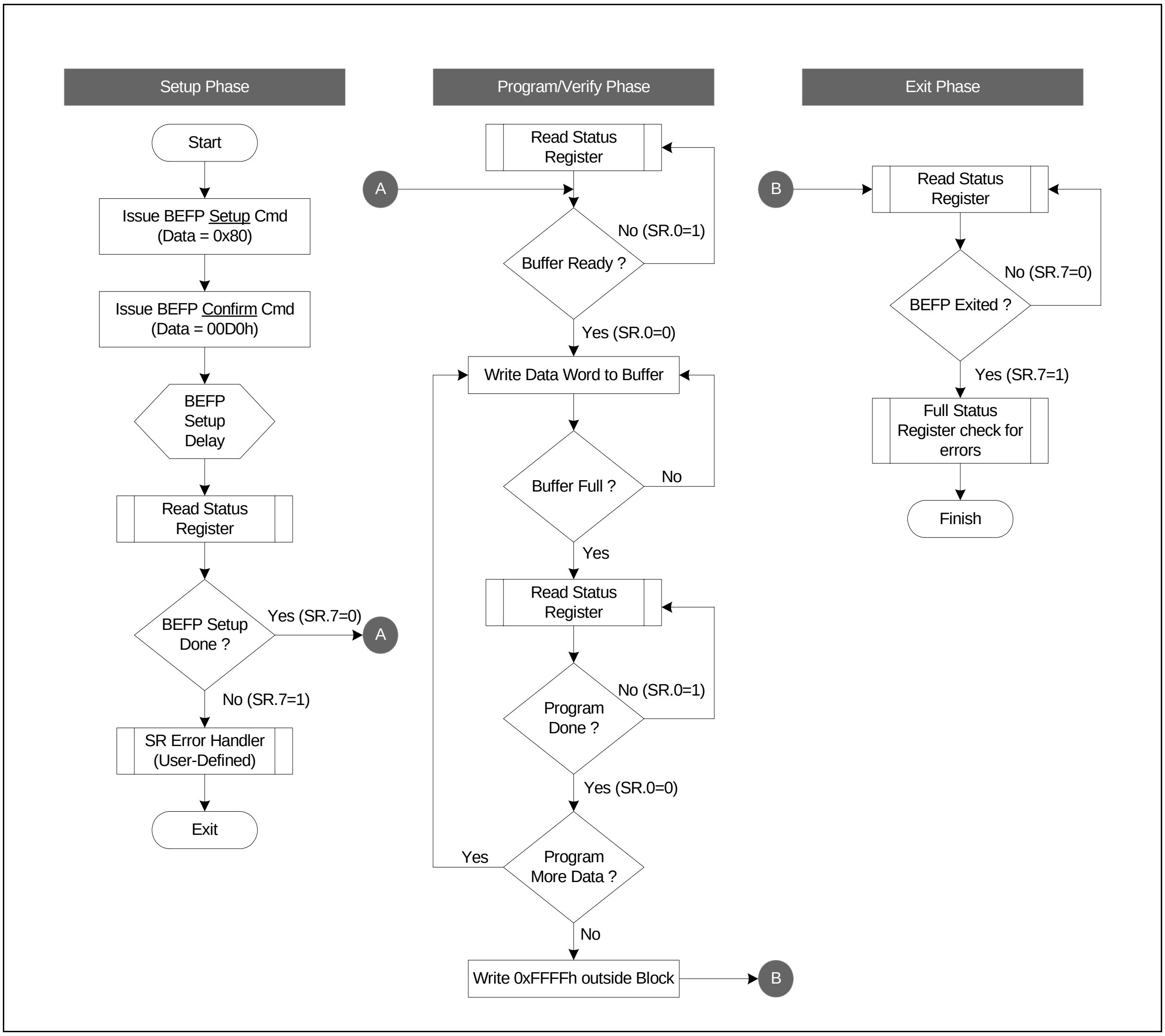


Figure 37: Block Erase Flowchart

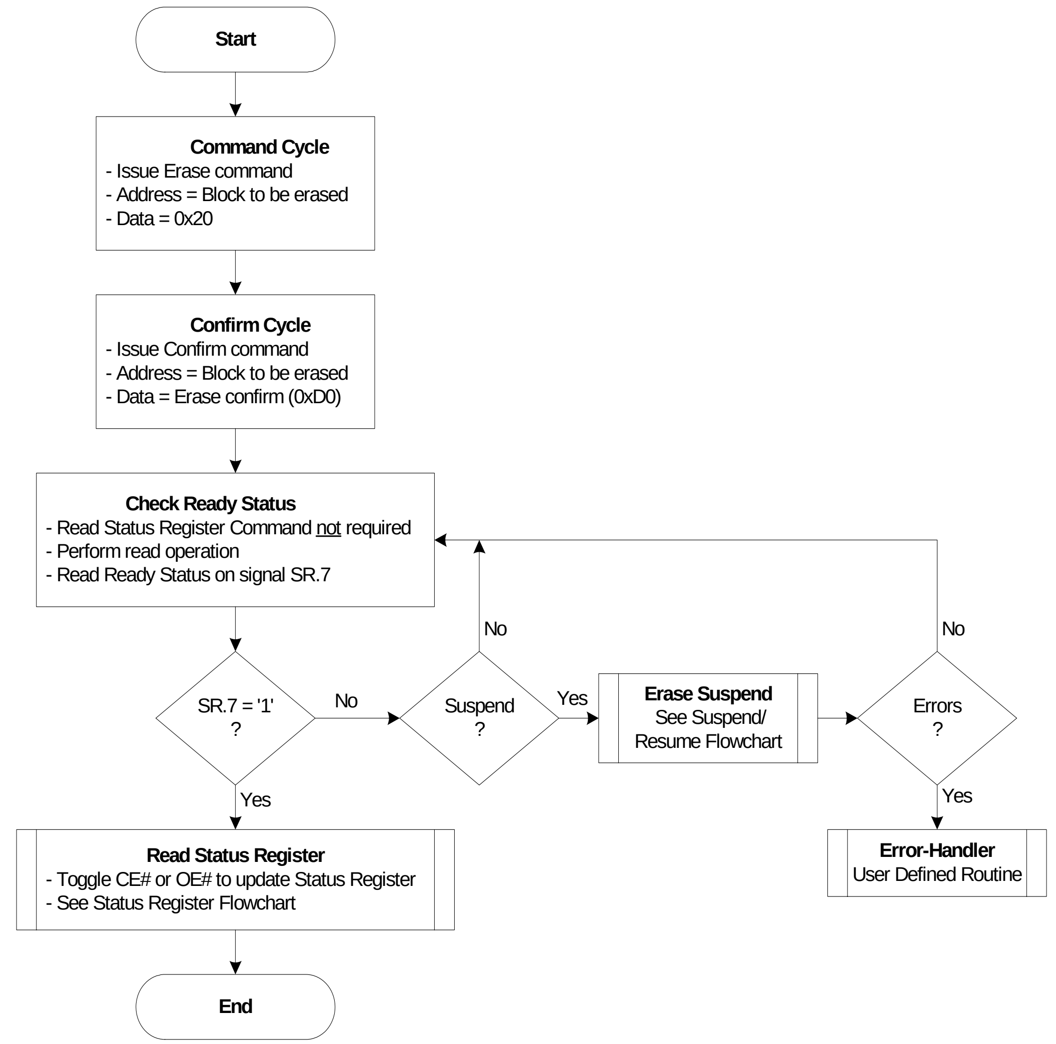


Figure 38: Block Lock Operations Flowchart

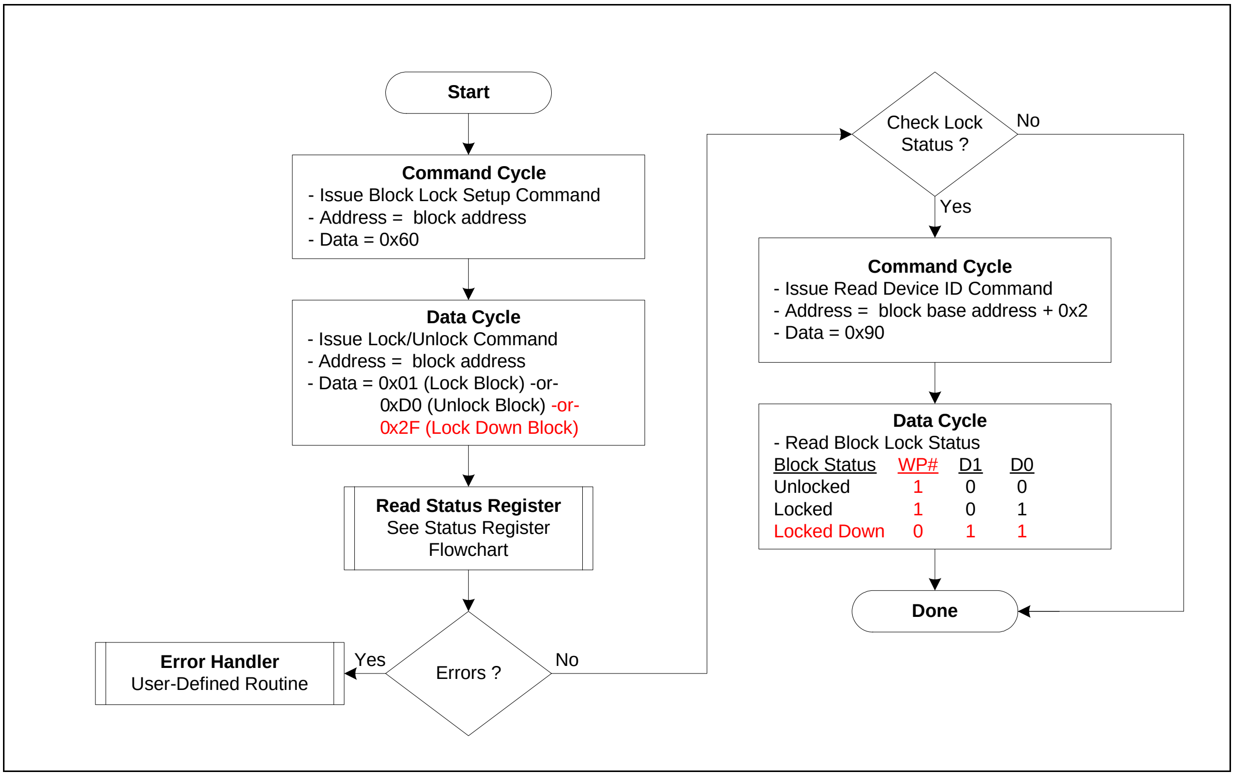


Figure 39: OTP Register Programming Flowchart

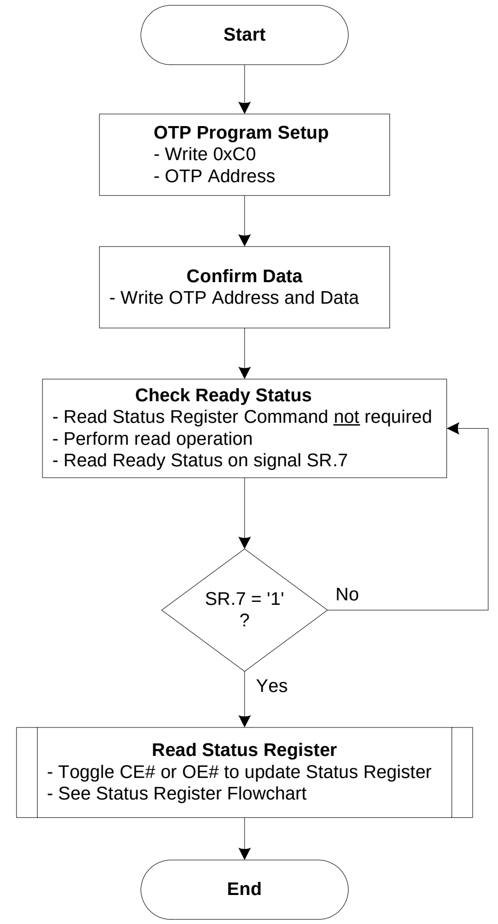
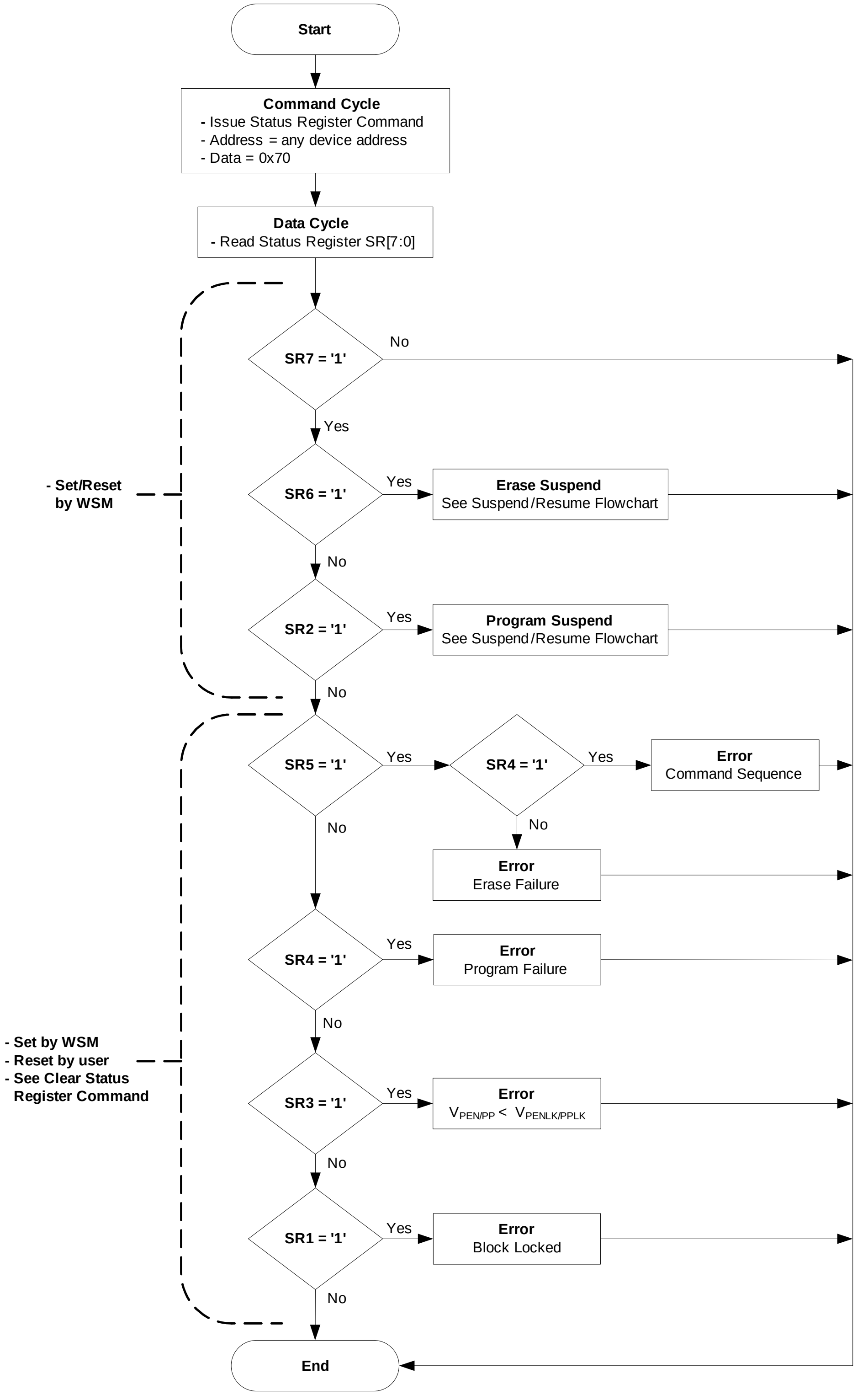


Figure 40: Status Register Flowchart



A.3 Write State Machine

Show here are the command state transitions (Next State Table) based on incoming commands. Only one partition can be actively programming or erasing at a time. Each partition stays in its last read state (Read Array, Read Device ID, Read CFI or Read Status Register) until a new command changes it. The next WSM state does not depend on the partition's output state.

Note: IS refers to Illegal State in the Next State Tables.

Table 39: Next State Table for P3x-65nm (Sheet 1 of 3)

Current Chip State		Command Input and Resulting Chip Next State(1)																			
		Array Read (3)	Word Pgm Setup (4,9)	BP Setup (8)	EFI Command Setup	Erase Setup (4,9)	BEFP Setup (6)	Confirm (7)	Pgm/Ers Suspend	Read Status	Clear SR (5)	Read ID/Query	Lock/RCR/ECR Setup	Blank Check	OTP Setup	Lock Blk Confirm (7)	Lock-down Blk Confirm (7)	Write ECR/RCR Confirm (7)	Block Address Change	Other Commands (2)	WSM Operation Completes
		(FFh)	(40h)	(E8h)	(EBh)	(20h)	(80h)	(D0h)	(B0)	(70h)	(50h)	(90h, 98h)	(60h)	(BCh)	(C0h)	(01h)	(2Fh)	(03h, 04h)		other	
Ready		Ready	Program Setup	BP Setup	EFI Setup	Erase Setup	BEFP Setup	Ready					Lock/RCR /ECR Setup	BC Setup	OTP Setup	Ready			N/A	Ready	N/A
Lock/RCR/ECR Setup		Ready (Lock Error [Botch])						Ready (Unlock Block)	Ready (Lock Error [Botch])						Ready (Lock Error [Botch])	Ready (Lock Block)	Ready (Lock down Block)	Ready (Set CR)	N/A	Ready (Lock Error [Botch])	N/A
OTP	Setup	OTP Busy				OTP Busy													N/A	OTP Busy	N/A
	Busy	OTP Busy	IS in OTP Busy	OTP Busy		IS in OTP Busy		OTP Busy					Illegal State in OTP Busy			OTP Busy			N/A	OTP Busy	Ready
	IS in OTP Busy	OTP Busy				OTP Busy															
Word Program	Setup	Word Program Busy																	N/A	Pgm Busy	N/A
	Busy	Pgm Busy	IS in Pgm Busy	Pgm Busy		IS in Pgm Busy		Pgm Busy	Pgm Susp	Word Pgm Busy			IS in Word Pgm Busy			Word Pgm Busy			N/A	Pgm Busy	Ready
	IS in Pgm Busy	Word Pgm Busy																			
	Suspend	Pgm Susp	IS in Pgm Susp	Pgm Suspend		IS in Pgm Susp		Pgm Busy	Pgm Susp		Pgm Susp (Er bits clear)	Word Pgm Susp	Illegal State in Pgm Suspend			Word Program Suspend			N/A	Word Pgm Susp	N/A
	IS in Pgm Suspend	Word Program Suspend																			
EFI	EFI Setup	Sub-function Setup																			N/A
	Sub-function Setup	Sub-op-code Load 1																			
	Sub-op-code Load 1	Sub-function Load 2 if word count >0, else Sub-function confirm																			
	Sub-function Load 2	Sub-function Confirm if data load in program buffer is complete, ELSE Sub-function Load 2																			
	Sub-function Confirm	Ready (Error [Botch])						S-fn Busy	Ready (Error [Botch])												
	Sub-function Busy	S-fn Busy	IS in S-fn Busy	S-fn Busy		Illegal State in S-fn Busy		S-fn Busy	S-fn Susp	S-fn Busy			IS in S-fn Busy			S-fn Busy			S-fn Busy		Ready
	IS in Sub-function Busy	Sub-function Busy																			
	Sub-function Susp	S-fn Susp	IS in S-fn Susp	Sub-function		Illegal State in S-fn Busy		S-fn Busy	S-fn Suspend		S-fn Susp (Er bits clear)	S-fn Susp	IS in S-fn Susp			S-fn Suspend			N/A	S-fn Susp	N/A
	IS in S-fn Susp	Sub-function Suspend																			

Table 39: Next State Table for P3x-65nm (Sheet 2 of 3)

Current Chip State		Command Input and Resulting Chip Next State(1)																			
		Array Read (3) (FFh)	Word Pgm Setup (4,9) (40h)	BP Setup (8) (E8h)	EFI Command Setup (EBh)	Erase Setup (4,9) (20h)	BEFP Setup (6) (80h)	Confirm (7) (D0h)	Pgm/Ers Suspend (B0)	Read Status (70h)	Clear SR (5) (50h)	Read ID/Query (90h, 98h)	Lock/RCR/ECR Setup (60h)	Blank Check (BCh)	OTP Setup (C0h)	Lock Blk Confirm (7) (01h)	Lock-down Blk Confirm (7) (2Fh)	Write ECR/RCR Confirm (7) (03h, 04h)	Block Address Change	Other Commands (2) other	WSM Operation Completes
Buffer Pgm (BP)	Setup	BP Load 1																			N/A
	BP Load 1 (8)	BP Load 2 if word count >0, else BP confirm																			
	BP Load 2 (8)	BP Confirm if data load in program buffer is complete, ELSE BP load 2																	Ready (Error [Botch])	BP Confirm if data load in program buffer is complete, else BP load 2	
	BP Confirm	Ready (Error [Botch])						BP Busy	Ready (Error [Botch])												
	BP Busy	BP Busy	IS in BP Busy	BP Busy		Illegal State in BP Busy		BP Busy	BP Susp	BP Busy			IS in BP Busy			BP Busy			BP Busy		Ready
	IS in BP Busy	BP Busy																			
	BP Susp	BP Susp	IS in BP Susp	BP Suspend		Illegal State in BP Busy		BP Busy	BP Suspend		BP Susp (Er bits clear)	BP Susp	IS in BP Susp			BP Suspend			N/A	BP Susp	N/A
	IS in BP Susp	BP Suspend																			
Erase	Setup	Ready (Error [Botch])						Erase Busy	Ready (Error [Botch])										N/A	Ready (Err Botch0])	N/A
	Busy	Erase Busy	IS in Erase Busy	Erase Busy		IS in Erase Busy		Erase Busy	Erase Susp	Erase Busy			IS in Erase Busy			Erase Busy			N/A	Ers Busy	
	IS in Erase Busy	Erase Busy																			Ready
	Suspend	Erase Susp	Word Pgm Setup in Erase Susp	BP Setup in Erase Susp	EFI Setup in Erase Susp	IS in Erase Suspend		Erase Busy	Erase Suspend		Erase Susp (Er bits clear)	Erase Susp	Lock/RCR/ECR Setup in Erase Susp	Erase Susp	IS in Erase Susp	Erase Suspend			N/A	Erase Susp	N/A
	IS in Erase Susp	Erase Suspend																			
Word Pgm in Erase Suspend	Setup	Word Pgm busy in Erase Suspend																	N/A	Word Pgm Busy in Ers Suspend	N/A
	Busy	Word Pgm busy in Erase Susp	IS in Pgm busy in Ers Susp	Word Pgm busy in Erase Susp		IS in Word Pgm busy in Ers Susp		Word Pgm busy in Erase Susp	Word Pgm Susp in Ers Susp	Word Pgm busy in Erase Susp			IS in Word Pgm busy in Ers Susp			Word Pgm busy in Erase Susp					Erase Susp
	Illegal state(IS) in Pgm busy in Erase Suspend	Word Pgm busy in Erase Suspend																			IS in Ers Susp
	Suspend	Word Pgm susp in Ers susp	iS in pgm susp in Ers Susp	Word Pgm susp in Ers susp		iS in pgm susp in Ers Susp		Word Pgm busy in Erase Susp	Word Pgm susp in Ers susp	Word Pgm susp in Ers susp	Word Pgm Susp in Ers Susp (Er bits clear)	Word Pgm susp in Ers susp	iS in Word Pgm susp in Ers Susp			Word Pgm susp in Ers susp			N/A		N/A
	Illegal State in Word Program Suspend in Erase Suspend	Word Pgm busy in Erase Suspend																			
BP in Erase Suspend	Setup	BP Load 1 in Erase Suspend																			N/A
	BP Load 1 (8)	BP Load 2 in Erase Suspend if word count >0, else BP confirm																			
	BP Load 2 (8)	BP Confirming Erase Suspend if data load in program buffer is complete, ELSE BP load 2 in Erase Suspend																	Ers Susp (Error [Botch])	BP Confirm in Erase Suspend when count=0, ELSE BP load 2	
	BP Confirm	Erase Suspend (Error [BotchBP])						BP Busy in Ers Susp	Erase Susp (Error [Botch BP])												
	BP Busy	BP Busy in Ers Susp	IS in BP Busy in Ers Susp	BP Busy in Erase Susp		Illegal State in BP Busy in Ers Susp			BP Susp in Ers Susp	BP Busy in Ers Susp			IS in BP Busy in Erase Suspend			BP Busy in Ers Susp			N/A	BP Busy in Ers Susp	Erase Susp
	IS in BP Busy	BP Busy in Erase Suspend																			IS in Ers Susp
	BP Susp	BP Susp in Ers Susp	IS in BP Susp in Ers Susp	BP Suspend in Erase Suspend		Illegal State in BP Busy in Ers Susp		BP Busy in Ers Susp	BP Susp in Ers Susp		BP Susp in Ers Susp (Er bits clear)	BP Susp in Ers Susp	IS in BP Busy in Erase Suspend			BP Susp in Ers Susp			N/A	BP Susp in Ers Susp	N/A
	IS in BP Suspend	BP Suspend in Erase Suspend																			

Table 39: Next State Table for P3x-65nm (Sheet 3 of 3)

Current Chip State		Command Input and Resulting Chip Next State(1)																			
		Array Read (3)	Word Pgm Setup (4,9)	BP Setup (8)	EFI Command Setup	Erase Setup (4,9)	BEFP Setup (6)	Confirm (7)	Pgm/Ers Suspend	Read Status	Clear SR (5)	Read ID/Query	Lock/RCR/ECR Setup	Blank Check	OTP Setup	Lock Blk Confirm (7)	Lock-down Blk Confirm (7)	Write ECR/RCR Confirm (7)	Block Address Change	Other Commands (2)	WSM Operation Completes
		(FFh)	(40h)	(E8h)	(EBh)	(20h)	(80h)	(D0h)	(B0)	(70h)	(50h)	(90h, 98h)	(60h)	(BCh)	(C0h)	(01h)	(2Fh)	(03h, 04h)		other	
EFI in Erase Suspend	EFI Setup	Sub-function Setup in Erase Suspend																			N/A
	Sub-function Setup	Sub-op-code Load 1 in Erase Suspend																			
	Sub-op-code Load 1	Sub-function Load 2 in Erase Suspend if word count >0, else Sub-function confirm in Erase Suspend																			
	Sub-function Load 2	Sub-function Confirm in Erase Suspend if data load in program buffer is complete, ELSE Sub-function Load 2																	Ers Susp (Error [Botch])	Sub-function Confirm if data load in program buffer is complete, ELSE Sub-function Load 2	
	Sub-function Confirm	Erase Suspend (Error [Botch])						S-fn Busy in Ers Susp	Erase Suspend (Error [Botch])												
	Sub-function Busy	S-fn Busy in Ers Susp	IS in S-fn Busy in Ers Susp	S-fn Busy in Ers Suspend		Illegal State in S-fn Busy in Ers Susp			S-fn Susp in Ers Susp	S-fn Busy in Ers Susp			IS in S-fn Busy in Ers Susp			S-fn Busy in Ers Susp			N/A	S-fn Busy in Ers Susp	Erase Susp
	IS in Sub-function Busy	Sub-function Busy in Ers Susp																			IS in Ers Susp
	Sub-function Susp	S-fn Susp in Ers Susp	IS in S-fn Susp in Ers Susp	S-fn Suspend in Ers Susp		Illegal State in S-fn Busy in Ers Susp		S-fn Busy in Ers Susp	S-fn Suspend in Ers Susp		S-fn Susp in Ers Susp (Er bits clear)	S-fn Susp in Ers Susp	IS in S-fn Susp in Ers Susp			S-fn Suspend in Ers Susp			N/A	S-fn Susp in Ers Susp	N/A
	IS in Phase-1 Susp	Sub-Function Suspend in Erase Suspend																			
Lock/RCR/ECR/Lock EFA Block Setup in Erase Suspend		Erase Suspend (Lock Error [Botch])						Ers Susp (Un-lock Block)	Ers Susp (Lock Error [Botch])						Ers Susp (Error [Botch])	Ers Susp Blk Lock	Ers Susp Blk Lk-Down	Ers Susp CR Set	N/A	Ers Susp (Error [Botch])	N/A
Blank Check	Setup	Ready (Error [Botch])						BC Busy	Ready (Error [Botch])										N/A	Ready (Error [Botch])	N/A
	Blank Check Busy	BC Busy	IS in BC Busy	BC Busy		IS in BC Busy		Blank Check Busy					IS in BC Busy			BC Busy				BC Busy	Ready
	IS in Blank Check Busy	BP Busy																			
BEFP	Setup	Ready (Error [Botch])						BEFP Load Data	Ready (Error [Botch])												N/A
	BEFP Busy	BEFP Program and Verify Busy (if Block Address given matches address given on BEFP Setup command). Commands treated as data. (7)																	Ready	BEFP Busy	Ready

Table 40: Output Next State Table for P3x-65nm

Current Chip State	Command Input to Chip and Resulting Output MUX Next State(1)																			
	Array Read (3)	Word Pgm Setup (4,9)	BP Setup (8)	EFI Command Setup	Erase Setup (4,9)	BEFP Setup (6)	Confirm (7)	Pgm/Ers Suspend	Read Status	Clear SR (5)	Read ID/Query	Lock/RCR/ECR Setup	Blank Check	OTP Setup	Lock Blk Confirm (7)	Lock-down Blk Confirm (7)	Write ECR/RCR Confirm (7)	Block Address Change	Other Commands (2)	WSM Operation Completes
	(FFh)	(40h)	(E8h)	(EBh)	(20h)	(80h)	(D0h)	(B0)	(70h)	(50h)	(90h, 98h)	(60h)	(BCh)	(C0h)	(01h)	(2Fh)	(03h, 04h)		other	
BEFPSetup, BEFP Pgm & Verify Busy, Erase Setup, OTP Setup, BP Setup, Load 1, Load 2 BP Setup, Load1, Load 2 - in Erase Susp. BP Confirm EFI Sub-function Confirm WordPgmSetup, Word Pgm Setup in Erase Susp, BP Confirm in Erase Suspend, EFI S-fn Confirm in Ers Susp, Blank Check Setup, Blank Check Busy	Status Read																			Output MUX does not Change
Lock/RCR/ECR Setup, Lock/RCR/ECR Setup in Erase Susp	Status Read																Array Read			
EFI S-fn Setup, Ld 1, Ld 2 EFI S-fn Setup, Ld1, Ld 2 - in Erase Susp.	Output MUX will not change																			
BP Busy BP Busy in Erase Suspend EFI Sub-function Busy EFI Sub-fn Busy in Ers Susp Word Program Busy, Word Pgm Busy in Erase Suspend, OTP Busy Erase Busy	Status Read	Status Read		Status Read	Status Read		Output MUX Does not Change		Status Read	Array Read	Status Read	Status Read			Output MUX does not Change					
Ready, Word Pgm Suspend, BP Suspend, Phase-1 BP Suspend, Erase Suspend, BP Suspend in Erase Suspend Phase-1 BP Susp in Ers Susp	Array Read			Output MUX doesn't Change							ID/Query Read									

Notes:

1. IS refers to *Illegal State* in the Next State Table.
2. "Illegal commands" include commands outside of the allowed command set.
3. The device defaults to "Read Array" on powerup.
4. If a "Read Array" is attempted when the device is busy, the result will be "garbage" data (we should not tell the user that it will actually be Status Register data). The key point is that the output mux will be pointing to the "array", but garbage data will be output. "Read ID" and "Read Query" commands do the exact same thing in the device. The ID and Query data are located at different locations in the address map.
5. The Clear Status command only clears the error bits in the status register if the device is not in the following modes:1. WSM running (Pgm Busy, Erase Busy, Pgm Busy In Erase Suspend, OTP Busy, BEFP modes) 2. Suspend states (Erase Suspend, Pgm Suspend, Pgm Suspend In Erase Suspend).
6. BEFP writes are only allowed when the status register bit #0 = 0 or else the data is ignored.
7. Confirm commands (Lock Block, Unlock Block, Lock-Down Block, Configuration Register and Blank Check) perform the operation and then move to the Ready State.
8. Buffered programming will botch when a different block address (as compared to the address given on the first data write cycle) is written during the BP Load1 and BP Load2 states.
9. All two cycle commands will be considered as a contiguous whole during device suspend states. Individual commands will not be parsed separately. (I.e. If an erase set-up command is issued followed by a D0h command, the D0h command will not resume the program operation. Issuing the erase set-up places the CUI in an "illegal state". A subsequent command will clear the "illegal state", but the command will be otherwise ignored.

Appendix B Conventions - Additional Documentation

B.1 Acronyms

BEFP:	Buffered Enhanced Factory Programming
CUI :	Command User Interface
MLC :	Multi-Level Cell
OTP :	One-Time Programmable
PLR :	one-time programmable Lock Register
PR :	one-time programmable Register
RCR :	Read Configuration Register
RFU :	Reserved for Future Use
SR :	Status Register
SRD	Status Register Data
WSM	Write State Machine

B.2 Definitions and Terms

VCC :	Signal or voltage connection
V_{CC} :	Signal or voltage level
h :	Hexadecimal number suffix
0b :	Binary number prefix
0x :	hexadecimal number prefix
SR.4 :	Denotes an individual register bit.
A[15:0] :	Denotes a group of similarly named signals, such as address or data bus.
A5 :	Denotes one element of a signal group membership, such as an individual address bit.
Bit :	Single Binary unit
Byte :	Eight bits
Word :	Two bytes, or sixteen bits
Kbit :	1024 bits
KByte :	1024 bytes
KWord :	1024 words
Mbit :	1,048,576 bits
MByte :	1,048,576 bytes
MWord :	1,048,576 words
K	1,000
M	1,000,000
3.0 V :	V_{CC} (core) and V_{CCQ} (I/O) voltage range of 2.3 V – 3.6 V
9.0 V :	VPP voltage range of 8.5 V – 9.5 V
Block :	A group of bits, bytes, or words within the flash memory array that erase simultaneously. The P33-65nm has two block sizes: 32 KByte and 128 KByte.

Main block :	An array block that is usually used to store code and/or data. Main blocks are larger than parameter blocks.
Parameter block :	An array block that may be used to store frequently changing data or small system parameters that traditionally would be stored in EEPROM.
Top parameter device :	A device with its parameter blocks located at the highest physical address of its memory map.
Bottom parameter device :	A device with its parameter blocks located at the lowest physical address of its memory map.

Appendix C Revision History

Date	Revision	Description
May 2008	01	Initial release
June 2008	02	Corrected SCSP order information
July 2008	03	Added W28 AC specification; Fixed Buffered Program Command error in figure 38; Updated block locking state diagram; Updated Address range in Memory Map figure; Changed LSB Address from A0 to A1 in figure7 under dual die configurations section; Changed LSB Address in ballout and pinout description from A0 back to A1 to match P33 130nm.
Sep 2008	04	Updated Axcell^{TM} trademark; Remove 64M related content;
Nov 2008	05	Update the Buffer program flowchart to reflect the read status register; minor wording modification; Return to StrataFlash; Update the buffer program comments for cross 512-Word boundary; Remove 128M related contents from this document; Correct A24 to A25 for virtual CE description in section 1.3; Remove Numonyx Confidential;
Dec 2008	06	Correct Page buffer address bits to Four on Section 7.1, “Asynchronous Page-Mode Read” on page 21. Correct VHH to V_{PPH} on Table 19, “DC Current Characteristics” on page 46 note.
Apr 2009	07	Add 512 Mbit (256/256) memory map in Figure 1, “P33-65nm Memory Map” on page 7 Correct RCR.4, RCR.5, RCR.7 and RCR.9 definitions in Table 11, “Read Configuration Register Description” on page 34 Correct A_0 to A_1 signal naming and remove invalid x8 information in Table 29, “Example of Query Structure Output of x16 Devices” on page 62