Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

Numonyx® P33-65nm Flash Memory

128-Mbit, 64-Mbit Single Bit per Cell (SBC)

Datasheet

Product Features

- High performance:
  - 60ns initial access time for Easy BGA
  - 70ns initial access time for TSOP
  - 25ns 8-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
  - 3.0V buffered programming at 1.8MByte/s (Typ) using 256-word buffer
  - Buffered Enhanced Factory Programming at 3.2MByte/s (typ) using 256-word buffer
- Architecture:
  - Asymmetrically-blocked architecture
  - Four 32-KByte parameter blocks: top or bottom configuration
  - 128-KByte main blocks
  - Blank Check to verify an erased block
- Voltage and Power:
  - V_{CC} (core) voltage: 2.3V – 3.6V
  - V_{CCQ} (I/O) voltage: 2.3V – 3.6V
  - Standby current: 35µA(Typ) for 64-Mbit, 50µA(Typ) for 128-Mbit
  - Continuous synchronous read current: 23mA (Typ) at 52 MHz
- Security:
  - One-Time Programmable Registers:
    - 64 OTP bits, programmed with unique information by Numonyx
    - 2112 OTP bits, available for customer programming
  - 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
  - Basic Command Set and Extended Function Interface (EFI) Command Set compatible
  - Common Flash Interface capable
- Density and Packaging:
  - 56-Lead TSOP package (128-Mbit, 64-Mbit)
  - 64-Ball Easy BGA package (128-Mbit, 64-Mbit)
  - 16-bit wide data bus
- Quality and Reliability:
  - JESD47E Compliant
  - Operating temperature: –40°C to +85°C
  - Minimum 100,000 erase cycles per block
  - 65nm process technology

Contents

1.0 Functional Description

1.1 Introduction

This document provides information about the Numonyx® P33-65nm Single Bit per Cell (SBC) Flash Memory and describes its features, operations, and specifications.

P33-65nm SBC device is offered in 64-Mbit and 128-Mbit densities. Benefits include high-speed interface NOR device, and support for code and data storage. Features include high-performance synchronous-burst read mode, a dramatical improvement in buffer program time through larger buffer size, fast asynchronous access times, low power, flexible security options, and two industry-standard package choices.

P33-65nm SBC device is manufactured using 65nm 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. The device features a 256-word buffer to enable optimum programming performance, which can improve system programming throughput time significantly to 1.8MByte/s.

The P33-65nm SBC device 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 $\leq V_{PPLK}$.

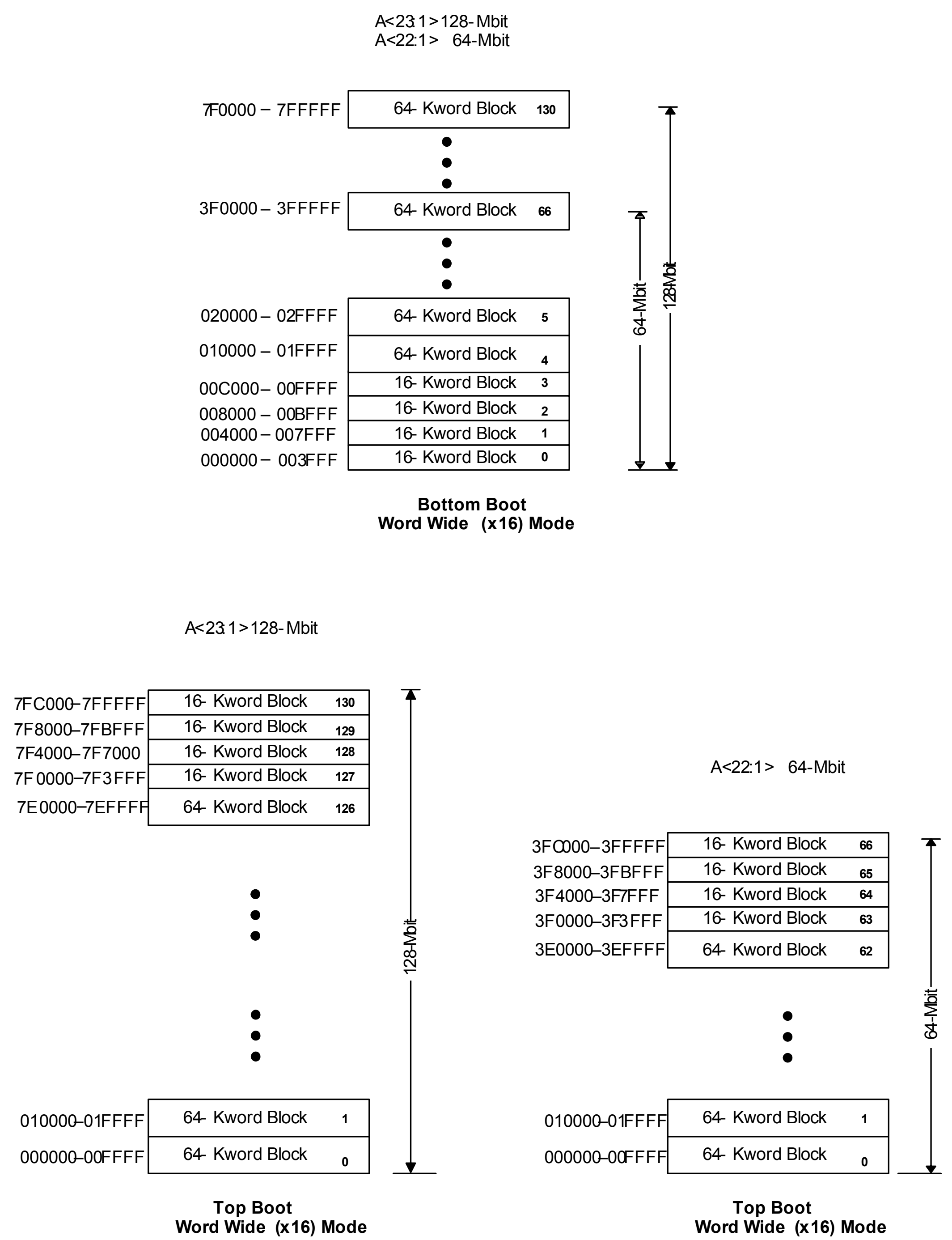
The Command User Interface is the interface between the system processor and all internal operations of the device. An internal Write State Machine automatically executes the algorithms and timings necessary for block erase and program. A Status Register indicates erase or program completion and any 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 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 SBC 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 P33-65nm SBC device could also provide the full-device OTP permanent lock feature.

1.3 Memory Maps

Figure 1: P33-65nm Memory Map (64-Mbit and 128-Mbit Densities)



2.0 Package Information

2.1 56-Lead TSOP

Figure 2: TSOP Mechanical Specifications

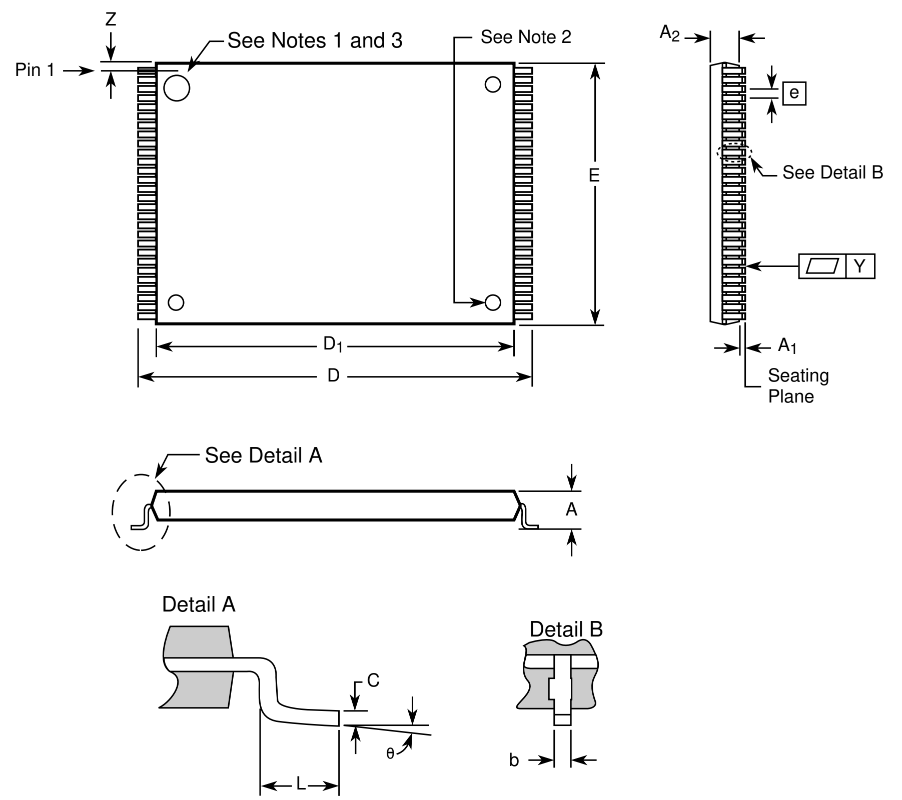


Table 1: TSOP Package Dimensions (Sheet 1 of 2)

Product Information	Symbol	Millimeters			Inches		
		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(4)	b	0.170	0.220	0.270	0.0067	0.0087	0.0106
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
Package Body Width	E	13.800	14.000	14.200	0.543	0.551	0.559
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 1: TSOP Package Dimensions (Sheet 2 of 2)

Product Information	Symbol	Millimeters			Inches		
		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.
4. For legacy lead width, 0.10mm(Min), 0.15mm(Typ) and 0.20mm(Max).

2.2 64-Ball Easy BGA Package

Figure 3: Easy BGA Mechanical Specifications (8x10x1.2 mm)

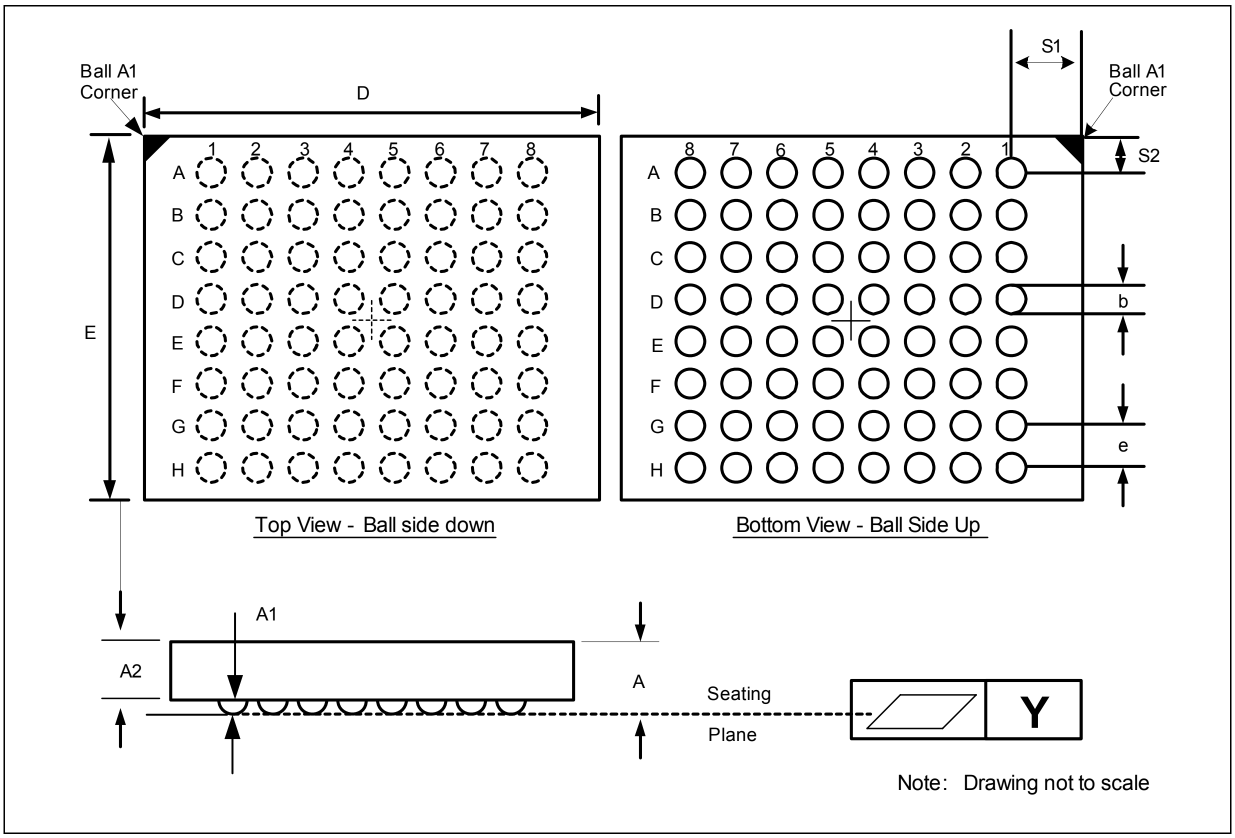


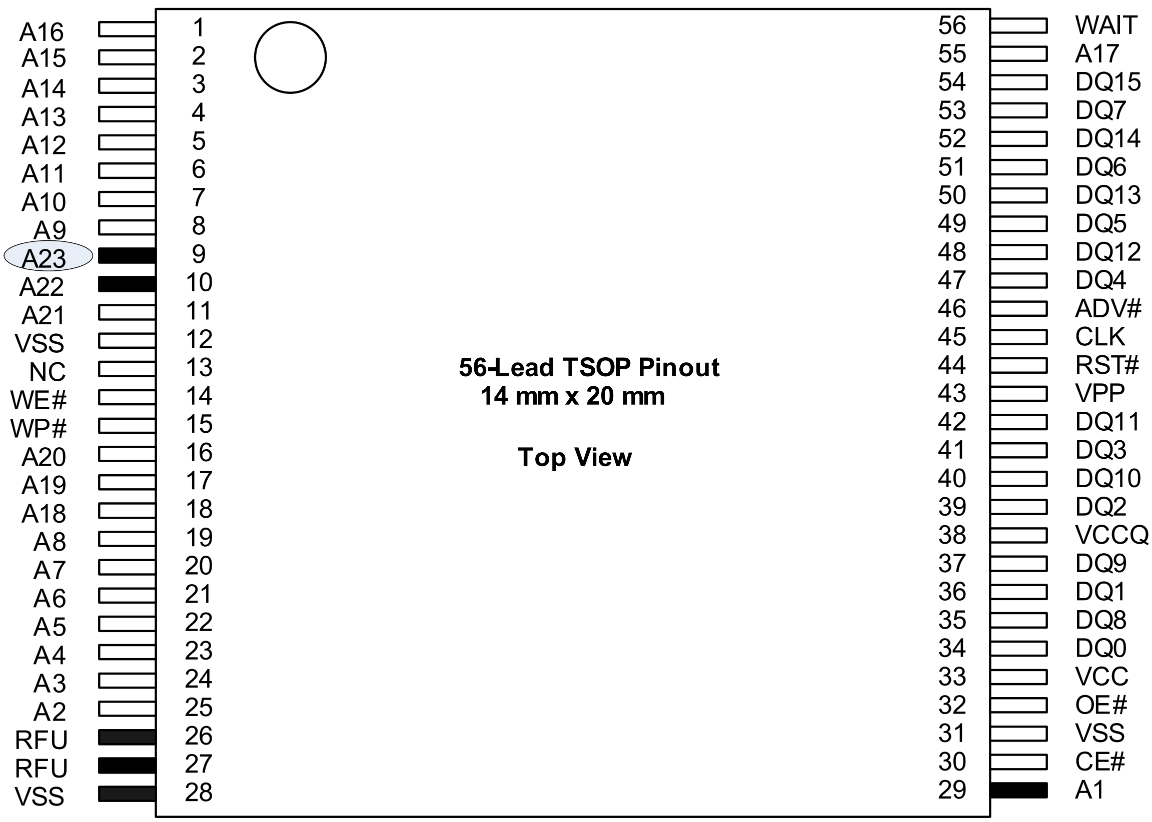
Table 2: Easy BGA Package Dimensions

Product Information	Symbol	Millimeters			Inches		
		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	0.310	0.410	0.510	0.0120	0.0160	0.0200
Package Body Width	D	9.900	10.000	10.100	0.3898	0.3937	0.3976
Package Body Length	E	7.900	8.000	8.100	0.3110	0.3149	0.3189
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
Corner to Ball A1 Distance Along E	S2	0.400	0.500	0.600	0.0157	0.0197	0.0236

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

3.0 Ballouts

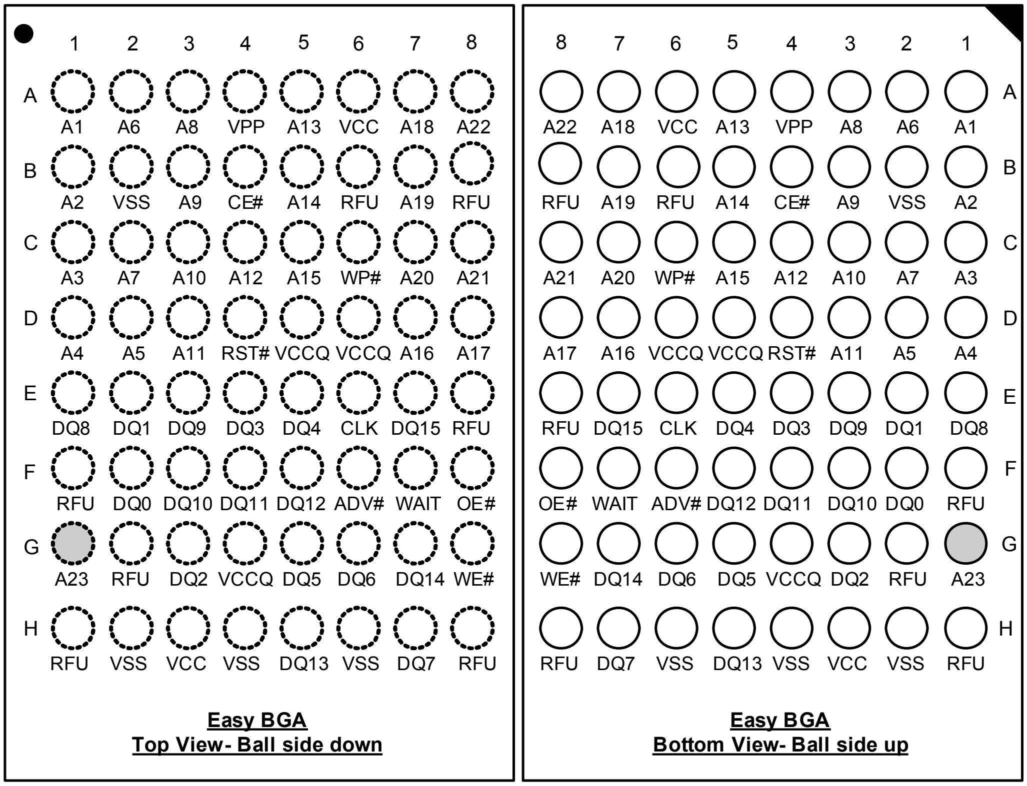
Figure 4: 56-Lead TSOP Pinout (64-Mbit and 128-Mbit Densities)



Notes:

1. A1 is the least significant address bit.
2. A23 is valid for 128-Mbit densities; otherwise, it is a no connect (NC).
3. A22 is valid for 64-Mbit densities and above; otherwise, it is a no connect (NC).
4. No Internal Connection on VCC Pin 13; it may be driven or floated. For legacy designs, pin can be tied to Vcc.
5. One dimple on package denotes Pin 1 which will always be in the upper left corner of the package, in reference to the product mark.

Figure 5: 64-Ball Easy BGA Ballout (64-Mbit and 128-Mbit Densities)



Notes:

1. A1 is the least significant address bit.
2. A23 is valid for 128-Mbit densities; otherwise, it is a no connect.
3. A22 is valid for 64-Mbit densities and above; otherwise, it is a no connect (NC).
4. One dimple on package denotes Pin 1 which will always be in the upper left corner of the package, in reference to the product mark.

4.0 Signals

Table 3: TSOP and Easy BGA Signal Descriptions (Sheet 1 of 2)

Symbol	Type	Name and Function
A[MAX:1]	Input	**ADDRESS INPUTS:** Device address inputs. 128-Mbit: A[23:1]; 64-Mbit: A[22:1]. WARNING: The active address pins unused in design should not be left float. Please tie them to VCCQ or VSS according to specific design requirements.
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# or CE#, whichever occurs first.
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. WARNING: Designs not using WP# for protection could tie it to VCCQ or VSS without additional capacitor.
VPP	Power/ Input	**ERASE AND PROGRAM POWER:** A valid voltage on this pin allows erasing or programming. Memory contents cannot be altered when VPP $\leq V_{PPLK}$. 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 $\leq V_{LKO}$. 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 3: 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.

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 SBC device conform to standard microprocessor bus operations. Table 4, "Bus Operations Summary"summarizes the bus operations and the logic levels that must be applied to the device control signal inputs.

Table 4: 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	2,3,4

Notes:
1. Refer to the Table 6, "Command Bus Cycles" on page 18 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.
4. Recommend to set CE# and WE# to V_{IH} on 65nm device during power-on/reset to avoid invalid commands written into flash accidently.

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 6, "Command Bus Cycles" on page 18 shows the bus cycle sequence for each of the supported device commands, while Table 5, "Command Codes and Definitions" on page 16 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 flash Command User Interface (CUI) provides access to device read, write, and erase operations. The CUI does not occupy an addressable memory location; it is part of the internal logic which allows the flash device to be controlled. The Write State Machine provides the management for its internal erase and program algorithms.

Commands are written to the CUI to control flash device operations. Table 5, "Command Codes and Definitions" describes all valid command codes.

For operations that involve multiple command cycles, the possibility exists that the subsequent command does not get issued in the proper sequence. When this happens, the CUI sets Status Register bits SR[5,4] to indicate a command sequence error.

Some applications use illegal or invalid commands (like 0x00) accidentally or intentionally with the device. An illegal or invalid command doesn't change the device output state compared with the previous operation on 130nm device. But the output will change to Read Status Register mode on 65nm device.

After an illegal or invalid command, software may attempt to read the device. If the previous state is read array mode before an illegal command, software will expect to read array data on 130nm device, such as 0xFFFF in an unprogrammed location. On the 65nm device, software may not get the expected array data and instead the status register is read.

Please refer to the legal and valid commands/spec defined in the Datasheet, such as for read mode, issue 0xFF to Read Array mode, 0x90 to Read Signature, 0x98 to Read CFI/ OTP array mode.

Table 5: Command Codes and Definitions (Sheet 1 of 3)

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.

Table 5: Command Codes and Definitions (Sheet 2 of 3)

Mode	Code	Device Mode	Description
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 256 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.
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 into the OTP array.

Table 5: Command Codes and Definitions (Sheet 3 of 3)

Mode	Code	Device Mode	Description
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[16:1]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	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 6, "Command Bus Cycles" on page 18. 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 6: 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	-	-	-

Table 6: 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)
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
Others	Blank Check	2	Write	BA	0xBC	Write	BA	D0
	Extended Function Interface(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
   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 256 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; $1 \le N \le 256$. 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 33).

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.

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} or V_{IL}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, eight 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 (RCR[13:11])" on page 34). 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 52
- Figure 21, "Continuous Burst Read, showing an Output Delay Timing" on page 53
- Figure 22, "Synchronous Burst-Mode Four-Word Read Timing" on page 53

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 Read Configuration Register data (see Section 6.2, "Device Command Bus Cycles" on page 18 for details on issuing the Read Device Identifier command). Table 7, "Device Identifier Information" on page 21 and Table 8, "Device ID codes" on page 21 show the address offsets and data values for this device.

Table 7: Device Identifier Information

Item	Address(1,2)	Data
Manufacturer Code	0x00	0x89h
Device ID Code	0x01	ID (see Table 8)
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	GPR data
Lock Register 0	0x80	PR-LK0
64-bit Factory-Programmed OTP Register	0x81–0x84	Numonyx 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 SBC, the GPR is used as read out register for Extended Function interface command.

Table 8: Device ID codes

ID Code Type	Device Density	Device Identifier Codes	
		–T (Top Parameter)	–B (Bottom Parameter)
Device Code	64-Mbit	881D	8820
	128-Mbit	881E	8821

7.4 Read CFI

The Read CFI command instructs the device to output Common Flash Interface data when read. See Section 6.1, "Device Command Codes" on page 16 for detail on issuing the CFI Query command. Section A.1, "Common Flash Interface" on page 60 shows CFI information and address offsets within the CFI database.

8.0 Program Operation

The device supports three programming methods: Word Programming (40h/10h), 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 29 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 29, "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 256-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 32, "Buffer Program Flowchart" on page 75).

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.

Note: *The device defaults to output SR data after the Buffered Programming Setup Command (E8h) is issued. CE# or OE# must be toggled to update Status Register. Don't issue the*

Read SR command (70h), which would be interpreted by the internal state machines as Buffer Word Count.

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 256-word boundary (A[8:1] = 0x00).

Note: If a misaligned address range is issued during buffered programming, the program region must also be within an 256-word aligned boundary.

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 flash programming. The enhanced programming algorithm used in BEFP eliminates traditional programming elements that drive up overhead in device programmer systems. (see Figure 33, “BEFP Flowchart” on page 76).

BEFP consists of three phases: Setup, Program/Verify, and Exit It uses a write buffer to spread flash program performance across 256 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 256 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 256-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

Table 9: BEFP Requirements

Parameter/Issue	Requirement	Notes
Case Temperature	T_C = 30°C ± 10°C	-
VCC	Nominal Vcc	-
VPP	Driven to V_{PPH}	-
Setup and Confirm	Target block must be unlocked before issuing the BEFP Setup and Confirm commands.	-
Programming	The first-word address (WA0) of 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.	-
Buffer Alignment	WA0 must align with the start of an array buffer boundary.	1

Note: Word buffer boundaries in the array are determined by A[8:1] (0x00 through 0xFF); the alignment start point is A[8:1] = 0x00.

Table 10: BEFP Considerations

Parameter/Issue	Requirement	Notes
Cycling	For optimum performance, cycling must be limited below 50 erase cycles per block.	1
Programming blocks	BEFP programs one block at a time; all buffer data must fall within a single block.	2
Suspend	BEFP cannot be suspended.	-
Programming the flash memory array	Programming to the flash memory array can occur only when the buffer is full.	3

Notes:
1. Some degradation in performance may occur is this limit is exceeded, but the internal algorithm continues to work properly.
2. If the internal address counter increments beyond the block's maximum address, addressing wraps around to the beginning of the block.
3. If the number of words is less than 256, 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 256 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 256, 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