

Technical Note

NAND Flash Performance Increase with PROGRAM PAGE CACHE MODE Command

Overview

NAND Flash devices are designed for applications requiring nonvolatile, high-density, solid-state storage memory. While NAND Flash may meet speed requirements of traditional applications, many newer applications require faster data processing capabilities. For these applications, Micron[®] NAND Flash delivers higher data throughput using PROGRAM PAGE CACHE MODE operation.

Cache programming works by pipelining data to the NAND Flash device. Pipelining involves inputting a page (2KB) of data to the cache register and then transferring this data to the data register when the cache programming command is issued. When the transfer is complete, the cache register is available to receive new data input while the data register simultaneously programs the NAND Flash array.

Typical page-programming time (^tPROG) for Micron NAND Flash devices is approximately 300 μ s for a 2KB page of data. PROGRAM PAGE CACHE MODE operation can improve data throughput by as much as 35 percent for x8 devices and 17 percent for x16 devices.

This technical note discusses the benefits of PROGRAM PAGE CACHE MODE operations over normal PROGRAM PAGE operations. It provides specific timing examples and instructions for performing PROGRAM PAGE CACHE MODE operations.

Functionality

Normal PROGRAM PAGE Operation

Micron NAND Flash devices have two command latch cycles and five address latch cycles, which are followed by up to 2,112 bytes or 1,056 words of data in a normal PROGRAM PAGE operation. Following the ADDRESS LATCH cycles, R/B# goes LOW for ^tPROG (a typical time of 300 μ s, up to a maximum time of 700 μ s). Timing details are provided in Tables 2 and 3 on page 5 for x8 devices and Tables 4 and 5 on page 6 for x16 devices.

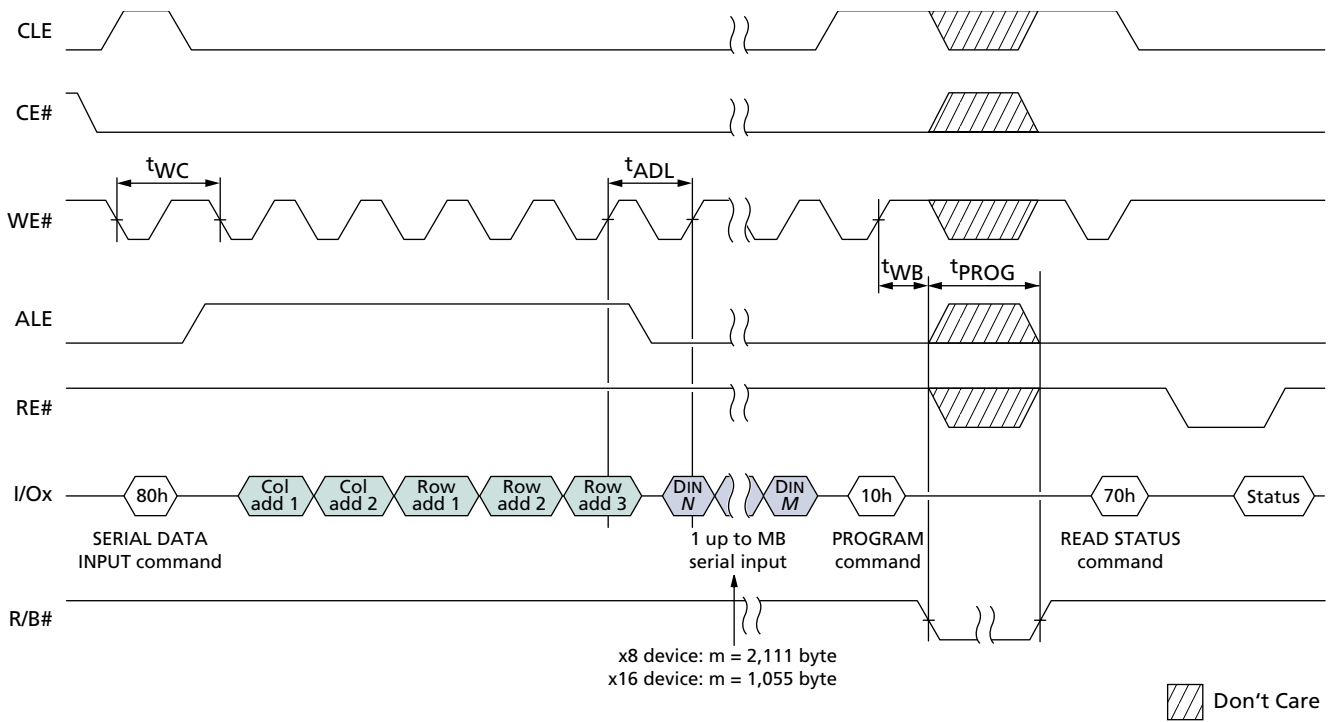
Data can be programmed sequentially, one byte or word per cycle, depending on the width configuration of the NAND Flash device. Assuming 50ns cycles, timing is shown in Figure 1 on page 2.

Figure 1: Normal PROGRAM PAGE Timing

x8 Configurations
 $(2 \times 50\text{ns}) \text{ commands} + (5 \times 50\text{ns}) \text{ address} + (2,112 \times 50\text{ns}) \text{ data} + 300\mu\text{s } t_{\text{PROG}}$
 $= 405.95\mu\text{s per page}$

x16 Configurations
 $(2 \times 50\text{ns}) \text{ commands} + (5 \times 50\text{ns}) \text{ address} + (1,056 \times 50\text{ns}) \text{ data} + 300\mu\text{s } t_{\text{PROG}}$
 $= 353.15\mu\text{s per page}$

Figure 2: NAND Flash Typical PROGRAM PAGE Operation Timing



PROGRAM PAGE CACHE MODE Operation

PROGRAM PAGE CACHE MODE operation begins with a PROGRAM PAGE CACHE MODE (80h-15h) command. Initially, data is copied into the cache register. When the 15h command completes, the data is transferred to the data register. Assuming that a page 0 address is issued in the PROGRAM PAGE CACHE MODE command, page 0 programming from the data register into the NAND Flash array begins when R/B# returns HIGH.

When R/B# returns HIGH, another PROGRAM PAGE CACHE MODE command sequence can be issued to write new data to the cache register. The time that R/B# stays LOW is determined by the actual programming time. On the first programming pass, R/B# stays LOW for the time it takes to transfer data from the cache register to the data register. On subsequent passes prior to the last page, the data in the data register must be programmed into the NAND Flash memory array before additional data can be transferred from the cache register.

When R/B# is used to determine programming status during PROGRAM PAGE CACHE MODE, a PROGRAM PAGE (80h-10h) command is used to program the last page of data. For the last cache programming sequence, R/B# stays LOW for t_{LPROG} . Table 1 below and Figure 3 on page 4 provide signal and timing information required for PROGRAM PAGE CACHE MODE operations.

For programming a single page, the PROGRAM PAGE CACHE MODE operation takes the same time as a regular PROGRAM PAGE operation (see tech note TN-29-01); however, for two or more pages, there is a significant performance gain using PROGRAM PAGE CACHE MODE.

The following examples assume t_{CBSY} is equal to t_{PROG} (TYP) after the first t_{CBSY} .

Table 1: Timing Delays

Parameter	Description	Typ	Max	Unit
t_{PROG}	Data transfer from data register to NAND Flash array	300	700	μs
t_{CBSY}	Busy time for PROGRAM PAGE CACHE MODE	3	700	μs
t_{LPROG}^1	Last-page programming time for PROGRAM PAGE CACHE MODE (see 2Gb NAND Flash data sheet)	–	–	–

Notes: 1. $t_{LPROG} = t_{PROG}$ (last page) + t_{PROG} ((last - 1) page) - command load time (last page) - address load time (last page) - data load time (last page).

Figure 3: PROGRAM PAGE CACHE MODE Operation Timing

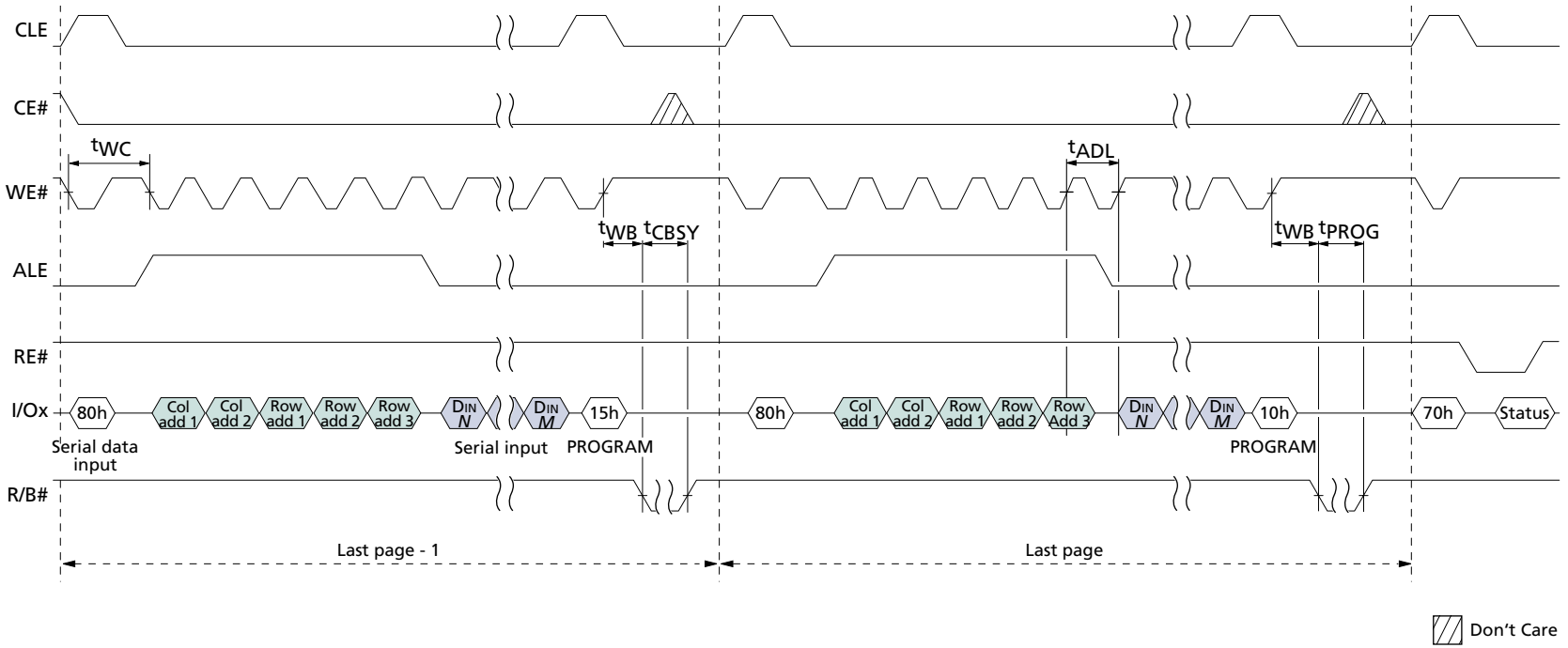


Table 2: Single-Block PROGRAM PAGE Operation Timing (x8)

Process	Repetitions	50ns Cycle Time		Unit
		Time	Total Time	
Command latch (80h)	1	50	50	ns
Address latch	5	50	250	ns
Program data cycles	2,112	50		ns
Program data cycles total page			105.6	µs
Command latch (10h)	1	50	50	ns
R/B# LOW (^t PROG) typical	1	300	300	µs
Total time to program a page			405.95	µs
Total time to program a block	64	0.40595	25.98	ms
Speed (programming 64 pages in a block)			5.2	MB/s

Table 3: Single-Block PROGRAM PAGE CACHE MODE Operation Timing (x8)

Process	Repetitions	50ns Cycle Time		Unit
		Time	Total Time	
Command latch (80h)	1	50	50	ns
Address latch	5	50	250	ns
Program data cycles	2,112	50		ns
Program data cycles total page			105.6	µs
Command latch (15h)	1	50	50	ns
R/B# LOW (1st ^t CBSY)	1	3	3	µs
Time to program page 0			108.95	µs
R/B# LOW (^t CBSY) typical	1	300	300	µs
Time to program page 1			300	µs
Time to program pages 2:62	61	0.300	18.3	ms
R/B# LOW (^t LPROG)	1	494.65	494.65	µs
Time to program page 63	1		494.65	µs
Total time to program a block			19.20	ms
Speed (programming 64 pages in a block)			7.04	MB/s

Table 4: Single-Block PROGRAM PAGE Operation Timing (x16)

Process	Repetitions	50ns Cycle Time		Unit
		Time	Total Time	
Command latch (80h)	1	50	50	ns
Address latch	5	50	250	ns
Program data cycles	1,056	50		ns
Program data cycles total page			52.50	μs
Command latch (10h)	1	50	50	ns
R/B# LOW (^t PROG)	1	300	300	μs
Total time to program a page			353.15	μs
Total time to program a block	64	0.35315	22.60	ms
Speed (programming 64 pages in a block)			5.98	MB/s

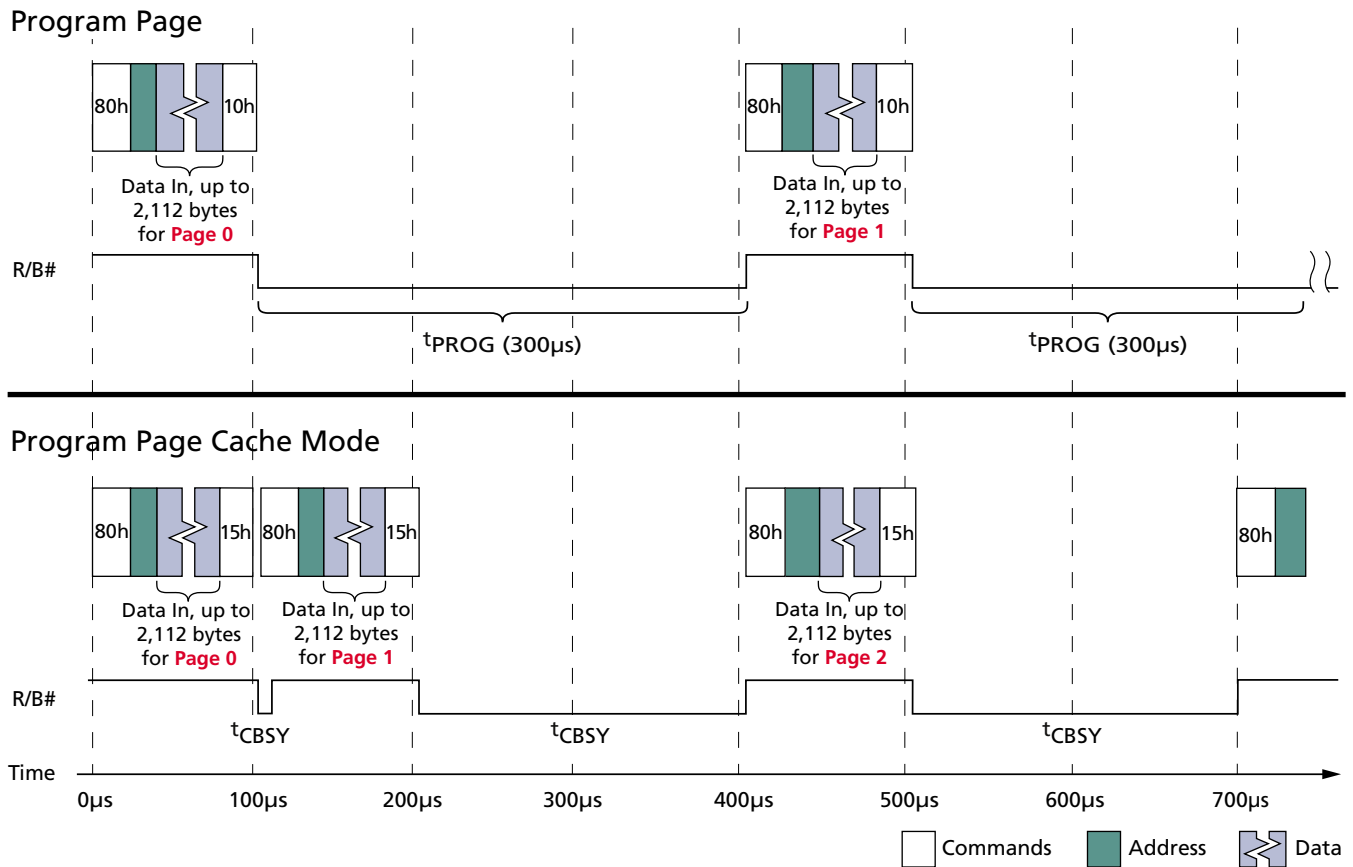
Table 5 shows timing calculations for 50ns PROGRAM CLOCK cycles with PROGRAM PAGE CACHE MODE.

Table 5: Single-Block PROGRAM PAGE CACHE MODE Operation Timing (x16)

Process	Number of Cycles	50ns Cycle Time		Units
		Time/Cycle	Total Time	
Timing for page 0 programming				
Command latch (80h)	1	50	50	ns
Address latch	5	50	250	ns
Program data cycles	1,056	50		ns
Program data cycles total page			52.50	μs
Command latch (15h)	1	50	50	ns
R/B# LOW (1st ^t CBSY)	1	3	3	μs
Time to program page 0			56.15	μs
R/B# LOW (^t CBSY)	1	300	300	μs
Time to program page 1			300	μs
Time to program pages 2:62	61	0.300	18.3	ms
R/B# LOW (^t LPROG)	1	543.85	543.85	μs
Time to program page 63			543.85	μs
Total time to program a block			19.2	ms
Speed (programming 64 pages in a block)			7.04	MB/s

Figure 4 on page 7 is a visual example that demonstrates the time savings associated with using PROGRAM PAGE CACHE MODE.

Figure 4: PROGRAM PAGE CACHE MODE Time Savings



Summary

By using the PROGRAM PAGE CACHE MODE operation available in Micron NAND Flash devices, customers can realize programming performance gains and increase data throughput in their systems.



8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900

prodmktg@micron.com www.micron.com Customer Comment Line: 800-932-4992

Micron, the M logo, and the Micron logo are trademarks of Micron Technology, Inc. All other trademarks are the property of their respective owners.

Revision History

Rev. B	5/07
<ul style="list-style-type: none">• “Overview” on page 1: Updated description.• “Normal PROGRAM PAGE Operation” on page 1: Added cross references.• “PROGRAM PAGE CACHE MODE Operation” on page 3: Revised description and cross reference.• Former “Head-to-Head Program Modes Comparison” and former Figure 5 on page 8: Deleted text and figure.• Former “Performance Increases” and former Figure 6 on page 9: Deleted text and figure.	
Rev. A	8/06
<ul style="list-style-type: none">• Initial release.	