MirrorBit® Flash Write Buffer Programming and Page Buffer Read

About this document

Scope and purpose

AN98479 discusses the specific set of commands to accomplish write buffer programming in Cypress GL MirrorBit® flash devices fabricated on 90-nm to 45-nm process technology.

Associated Part Family

S29GL-P, S29GL-S, S29GL-T

Table of contents

1 Overview

The write buffer in Cypress MirrorBit® Flash memory devices is designed to reduce the overall system programming time when writing to the device. The host system issues a Write to Buffer command, fills the write buffer with data, and then issues the Programming Buffer to Flash command. The flash device programs the buffered data in parallel, and thus reduces the programming time compared to programming bytes or words one at a time. Depending on the device design, a write buffer can store up to a maximum write buffer size of data that can then be written using a single programming operation. MirrorBit devices use a specific set of commands to accomplish write buffer programming.

Refer to the appropriate Cypress data sheet for the complete list of device commands. The page buffer read function accelerates read operations for addresses within a specific range. **Table 1** shows Write Buffer size and Page Buffer size for GL-P, GL-S, and GL-T.

Table 1 Write buffer, page buffer size

Product	Write buffer size	Page buffer size
GL-P	32 word/64 byte	8 word/16 byte
GL-S (64Mb)	128 word/256 byte	8 word/16 byte
GL-S (≥128Mb)	256 word (512 byte)	16 word (32 byte) [1]
GL-T	256 word/512 byte	16 word/32 byte

[1] Byte mode is not available on the GL-S (≥128 Mb).

2 Write buffer programming

2.1 Benefits of write buffer programming

The main benefit for the using the write buffer is programming speed. Data can be written into the write buffers at the Write Cycle Time (t_{WC}). In GL-P, t_{WC} is same as speed option of the part (e.g. a 90-ns part can have new data written into the buffer every 90 ns). In the case of GL-S and the GL-T, t_{WC} is 60 ns regardless of the speed option. GL-S and GL-T enable faster programming than GL-P. The write buffer programs the flash memory array in parallel, greatly decreasing the time needed to write to the Flash. **Figure 1** shows that using the write buffers with the maximum payload size is the most efficient.

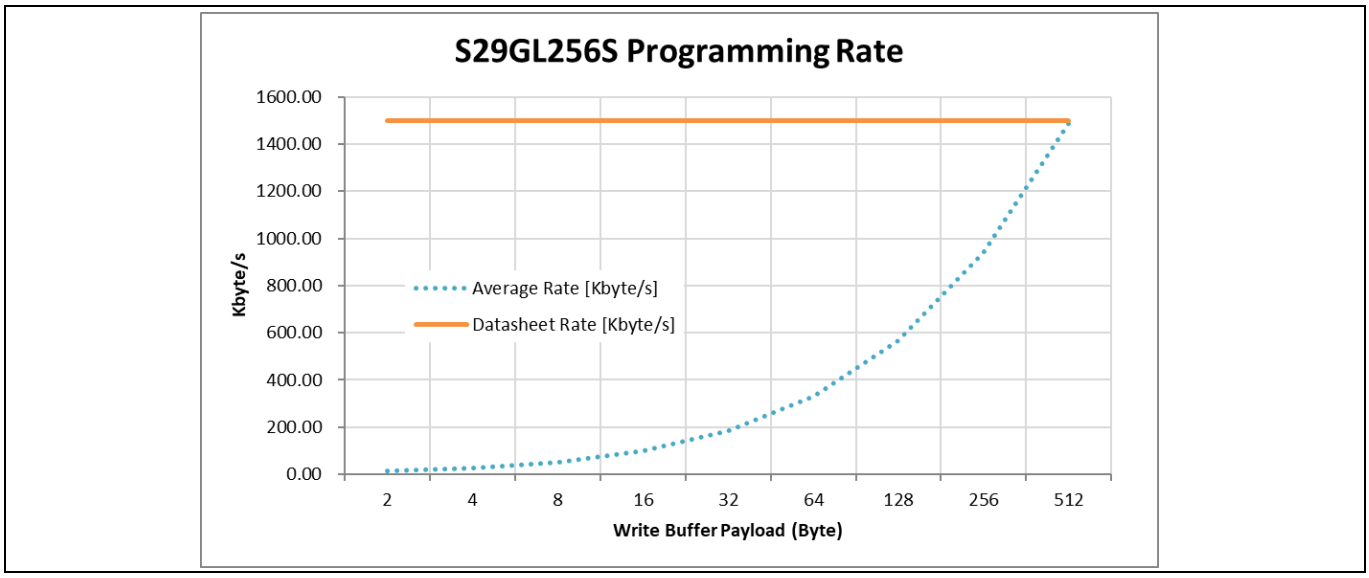

Figure 1 Program Rate (KB/s) vs. Payload Size (B)

2.2 Write buffer operation

Write buffer programming is only available through the Write to Buffer and Program Buffer to Flash command sequences. The Write-to-Buffer Abort Reset command sequence is used to exit out of the Write-Buffer-Abort state. **Table 2** lists all software program sequences associated with the write buffer.

Table 2 Write buffer operation

Command sequence	Interface	Bus cycles											
		First		Second		Third		Fourth		Fifth		Sixth	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Write to buffer	Word	555	AA	2AA	55	SA	25	SA	WC[2]	PA	PD	WBL[2]	PD
	Byte[3]	AAA		555									
Program buffer to flash (Confirm)	Word	SA	29										
	Byte[3]												
Write-to-buffer abort reset	Word	555	AA	2AA	55	XXX	F0						
	Byte[3]	AAA		555									

[2] The sixth cycle must be repeated to complete the number of writes specified by the WC in cycle four.

[3] Byte mode is not available on GL-S (≥128 Mb).

Write buffer programming

Legend:

PA = Program Address of the memory location to be programmed. This can be any address within the target write buffer page.

PD = Program Data to be programmed at location PA.

SA = Sector Address containing locations to be programmed. This can be any valid address within the sector.

WC = Write Count is the number of write buffer locations to load minus one (to load 7 locations the WC would be 6).

WBL = Write Buffer Location. The address must be within the same write buffer page or Line.

In **Table 2**, the host system first loads data at any location in the target write buffer page. Subsequent write buffer locations do not need to be loaded in any particular order as long as they reside in the same write buffer page and sector. In the case of GL-S (≥128 Mb), the write buffer page is defined as the address from xxx0h to xxFFh (any sequence of address where AMAX - A8 do not change).

Note that the internal write counter decrements for every data load operation, not for each unique write buffer address location. If the same write buffer location is loaded multiple times, the internal write counter will decrement after each load operation. The last data loaded into a given write-buffer location will be programmed into the device after the Program Buffer to Flash (confirm) command. The host system must therefore account for the effects of loading a write-buffer location more than once.

When the Write to Buffer command programming sequence has been completed, the Program Buffer to Flash command (confirm) must be issued to move the data from the write buffer into the flash memory array.

2.3 Write buffer programming abort

As stated earlier, write buffer programming cannot be performed across multiple write buffer pages, lines, or across multiple sectors. If this is attempted, the write buffer programming operation will be automatically aborted. The abort condition is detected by performing a status read operation, in which the data shows DQ1 = 1, DQ7 = DATA# (for the “Last Loaded Address”), DQ6 = TOGGLE, DQ5 = 0, and also bit3 in the status register on GL-S and GL-T. The write buffer programming sequence aborts in the following cases:

- Loading a value that is greater than the write buffer size (write-buffer-page) during the “fourth cycle (WC) Numbers of Locations to Program” step.
- Writing to an address in a sector that is different than the one specified during the Write-Buffer-Load command.
- Writing an Address/Data pair to a different write buffer page than the one selected by the starting address during the “write buffer data loading” stage of the operation.
- Writing data other than the Program Buffer to Flash command after loading the specified number of write buffer locations.

Note that the Write-to-Buffer Abort Reset command sequence must be written to the device, after a write buffer program abort, to return the device to READ mode.

2.4 Write buffer programming in S29GL-P, S29GL-S, and S29GL-T devices

In Cypress S29GL-P, S29GL-S, and S29GL-T devices, write buffer programming with program and erase suspend/resume functionality allows the system to write to a maximum of 32 words (GL-P), 128 words (64-Mb GL-S), 256 words (≥128-Mb GL-S and GL-T) in one programming operation. This provides faster programming performance than word programming. GL-S (≥128-Mb) can program data at 1.5 MB/s, and includes a status register to check the status of embedded operation. **Figure 2** shows write buffer program procedure with

Write buffer programming

ordinary data polling method, and **Figure 3** shows the write buffer program with status register. To ensure correct operation, please refer to the appropriate Cypress data sheet for a review of the full write buffer operation.

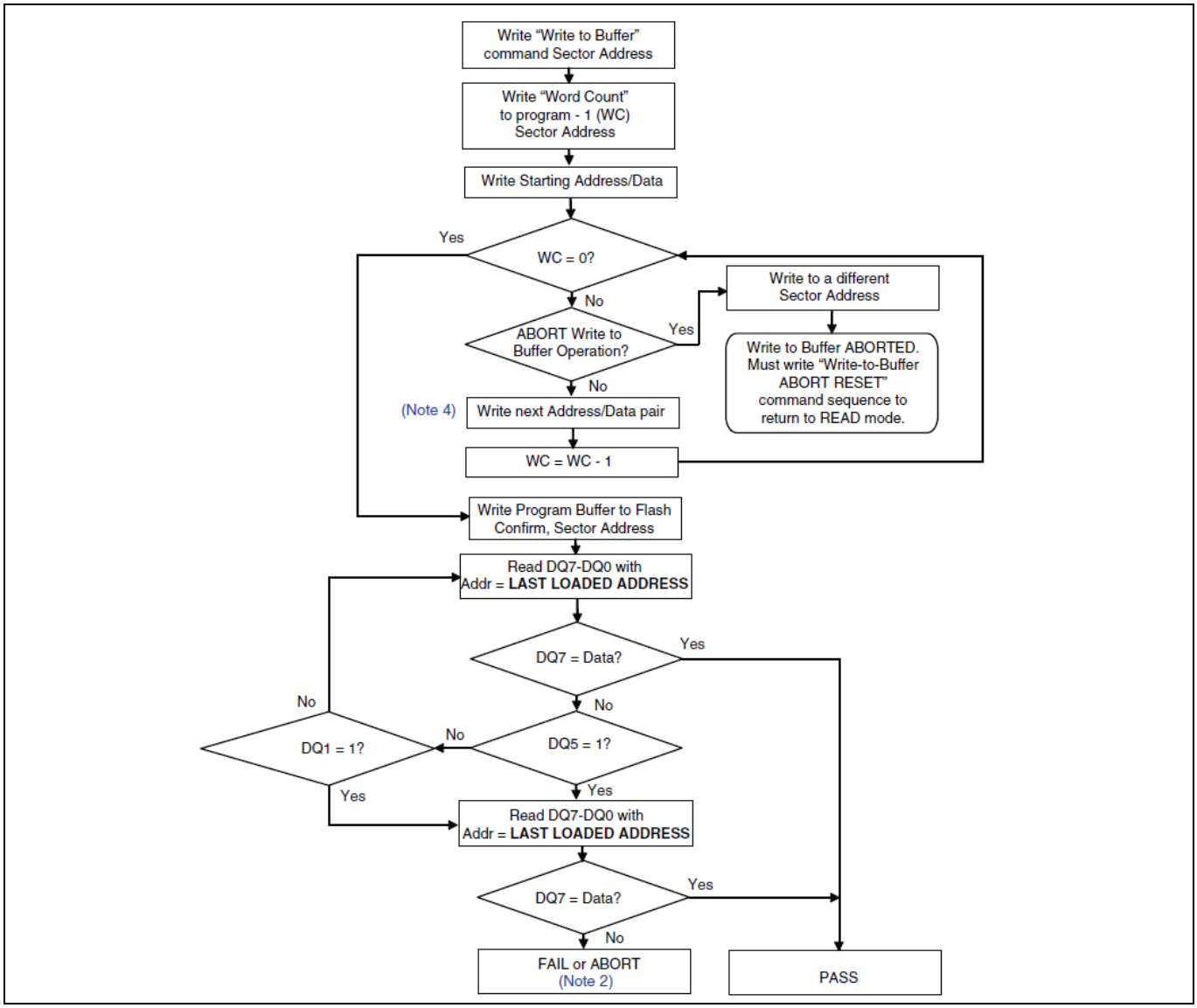

Figure 2 Write buffer programming operation with data polling

Notes:

1. *DQ7 should be rechecked even if DQ5 = 1 because DQ7 may change simultaneously with DQ5.*
2. *If this flowchart location was reached because DQ5 = 1, then the device FAILED. If this flowchart location was reached because DQ1 = 1, then the Write Buffer operation was ABORTED. In either case the proper RESET command must be written to the device to return the device to READ mode. Write-Buffer-Programming-Abort-Rest if DQ1 = 1, either Software RESET or Write-Buffer-Programming-Abort-Reset if DQ5 = 1.*
3. *See **Table 2** for the command sequence as required for Write Buffer Programming.*
4. *When Sector Address is specified, any address in the selected sector is acceptable. However, when loading Write-Buffer address locations with data, all addresses MUST fall within the selected Write-Buffer Page.*

Write buffer programming

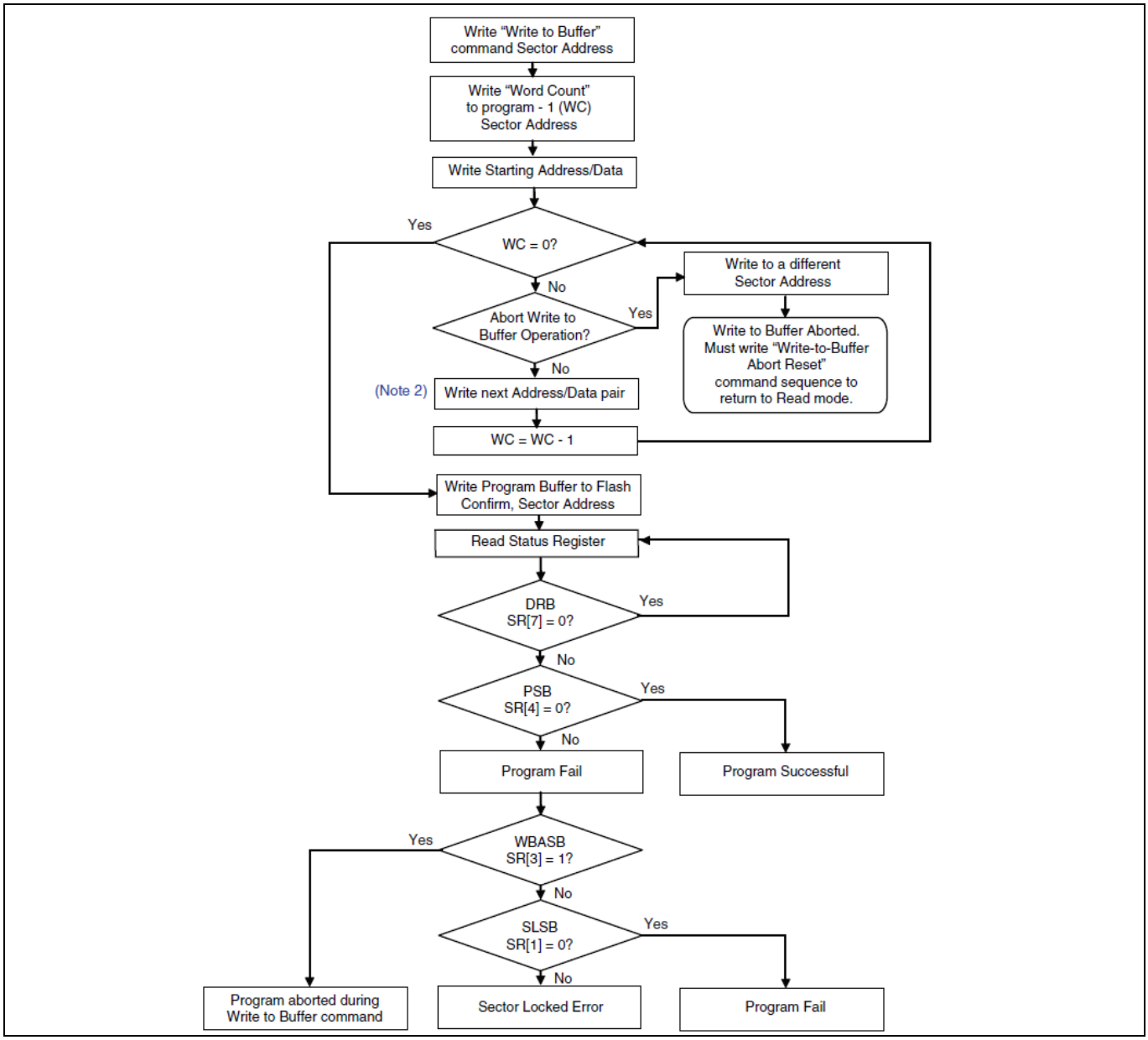

Figure 3 Write buffer programming operation with status register

Notes:

1. *See Command Definitions in the datasheet for the command sequence as required for Write Buffer Programming.*
2. *When Sector Address is specified, any address in the selected sector is acceptable. However, when loading Write-Buffer address.*

3 Page buffer reads

3.1 Page buffer read instruction

Whenever the host system changes a “page address” (or toggles CE# during a read), the device performs a “random access”. During this “random access” the read page buffer is loaded in parallel with data within the selected read-page boundaries. Subsequent intra-page accesses are 3 to 8 times faster than random accesses because the data are already available in the buffer. Therefore, read performance is significantly improved.

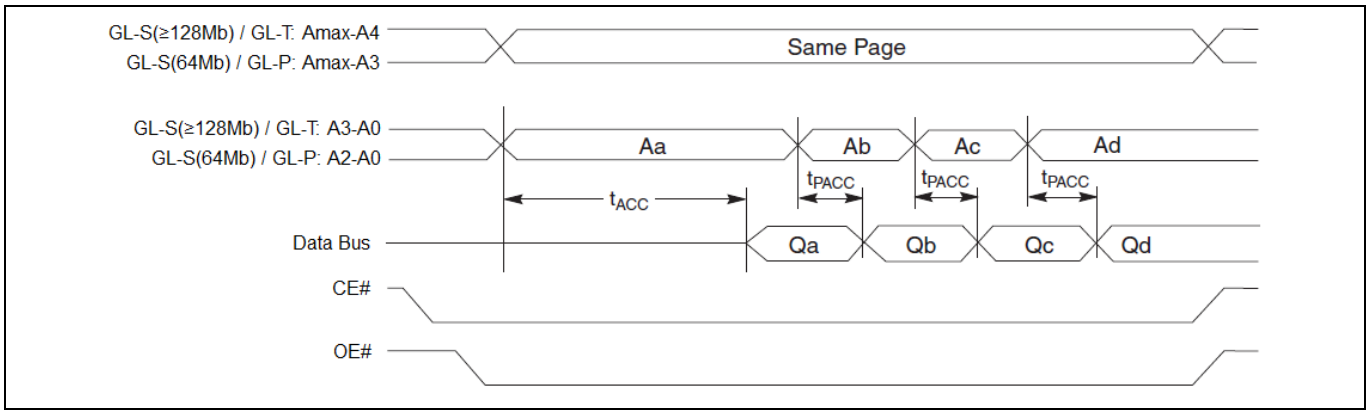

Figure 4 Page buffer read timing diagram

Note: The timing diagram shows device in word mode. Addresses A2–A-1 are used during byte mode except GL-S (≥128 Mb).

3.2 Read buffer operation with a 8-word/16-word page buffer

Cypress page mode flash devices support the use of a multi-word read page buffer. GL-P and GL-S (64Mb) has an 8-word (16-byte) read page buffer. GL-S (≥128 Mb) and GL-T has a 16-word (32-byte) read page buffer. For a page buffer read operation, the user must issue a read address, or “RA”, for any memory location. During the initial access time (t_{CE}/t_{ACC}) a page starting from a 16-bytes/32-bytes boundary, is read into the page buffer. If the device is in word mode, address bits A3-A0 or A2-A0 can then be used to access any address within the page with a reduced page access time (t_{PACC}). When the device is operated in byte mode, A-1 is also used to access any of the data in the read buffer page.

A depiction of the command sequence definition for read accesses is shown in **Table 3**.

Table 3 Write buffer operation

Command sequence	Interface	Bus cycles									
		First		Second		Third		Fourth		Fifth	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read	Word	RA	RD	RA	RD	RA	RD	RA	RD	RA	RD
	Byte[4]										

Legend:

RA = Read Address

[4] Byte mode is not available on the GL-S (≥128 Mb).

Page buffer reads

Note: For reading bytes, sixteen or thirty-two consecutive memory locations can be read, compared to eight or sixteen memory locations for reading words. “Intra-read page” locations can be accessed in any order.

A depiction of the device bus operation for read accesses is shown in **Table 4**.

Table 4 Device bus operation for read access

Operation	CE#	OE#	WE#	RESET#	WP#	Address	Data
Read	L	L	H	H	X	A_{IN}	D_{OUT}

During page buffer read operations, the CE# pin must be kept at voltage level VIL during all fast page mode accesses. If the CE# pin toggles or changes state during a page buffer read operation, the current data transfer will automatically be aborted and another initial page access is started. This will result in an unnecessary delay in read timings.

4 Conclusion

The write buffer programming feature of MirrorBit flash memory devices can decrease the programming time by over 20%, when compared to single word programming. Furthermore, the GL-S and GL-T can reduce programming time over 90% due to its new enhanced architecture called MirrorBit Eclipse™ Flash memory. Write buffer programming is enabled via a simple addition of three commands to the standard embedded algorithm bus command set.

The read page buffer feature of MirrorBit flash memories can increase performance significantly. Following each random (inter-page) access, all locations of the referenced 8-word or 16-word page are available for fast access. When multiple read accesses are grouped within a page the average read performance can be increased by 3 to 8 times.

Revision history

Document version	Date of release	Description of changes
**	2012-06-28	New application note.
*A	2015-10-09	Updated to Cypress template.
*B	2017-07-17	Updated Cypress Logo and Copyright.
*C	2018-07-12	Added Associated Part Family as “S29GL-P, S29GL-S, S29GL-T”. Added GL-S (64Mb) and GL-T products related information in all instances across the document. Added S29GL-T device related information in all instances across the document.
*D	2021-07-01	Updated to Infineon template.

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2021-07-01
Published by
Infineon Technologies AG
81726 Munich, Germany

Do you have a question about this document?
Go to www.cypress.com/support

Document reference
001-98479 Rev. *D

IMPORTANT NOTICE
The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.