

Please note that Cypress is an Infineon Technologies Company.

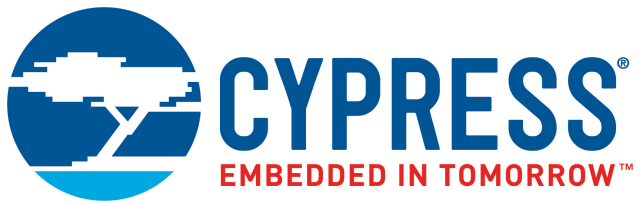
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content

The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers

Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.



THIS SPEC IS OBSOLETE

Spec No: 001-98510

Spec Title: AN98510 - DEVELOPING SYSTEM-LEVEL VALIDATION
ROUTINES

Replaced by: 002-17979, 001-99121, 001-98549

Developing System-Level Validation Routines

AN98510 discusses the guidelines for designing system-level test routines and validating the flash memory subsystem.

1 Introduction

During the system design and validation process, designers may require stress testing of their platforms and in system testing of Flash memory. Some test routines may inadvertently expose the systems flash memory components to stress far exceeding operational stress and result in incorrect analyses of system reliability. To better engineer system level test routines and to validate the flash memory subsystem, Cypress offers the following guidelines.

2 Systematic Approach

Systems using Flash memory for specialized applications commonly need to evaluate the Flash operational performance by simulating the final products durability under worst case conditions. When attempting to understand the long term behavior of Flash memory subsystems, laboratory test conditions can be used to extrapolate short-term test results, out in time. Using this evaluation method allows product lifetime under maximum runtime stress to be scrutinized. Understanding of Flash architecture, as it relates to endurance, can act as a guide for planning the evaluation process.

Erase time, for some applications, is a primary concern. Flash erase time exhibits variability during system test when the test program attempts multiple erase operations in an artificially short amount of time. By flooding the part with a large number of erase cycles and by repeatedly issuing erase commands within a single sector, the Flash memory will exhibit protracting of erase times which can be misunderstood as improper operation.

Cypress's MirrorBit™ Flash memory is based on a nitride isolation barrier bit cell architecture which utilizes hot electron injection for programming and hot hole injection for erasure. These processes manipulate minute numbers of electrons, which set charge levels representing logical states. During rapid erase cycling, hot holes, tend to over accumulate in the memory cell, the resulting net positive charge can lead to memory column leakage which may trigger programming failures from reduced programming efficiency. A process called annealing is used to reduce the excess positive charge in the memory cell. The annealing process occurs after a sector erase command and requires a finite amount of time to complete. Annealing returns the cell to a fully programmable state. The annealing time must be added to any erase time calculation, or time-out, in both the test program and the final application.

Flash memories are arranged in sectors or blocks of storage locations. A system test program should step sequentially, sector to sector in a round-robin fashion as the erase; write and read cycles are issued. By incrementally interrogating the sectors the programmer is optimally extending his test over the entirety of the memory array and allowing "self healing or relaxation time" to occur (much like "wear leveling").

To correctly evaluate a Cypress MirrorBit Flash device, a test program must take into consideration the operational details just covered by coding the following procedures into the test program.

3 Test Program Procedures

1. Proper use of the Cypress Flash device requires that each sector not be erased more that 1000 times per day (approximately once per 90 seconds). Test programs that erase individual sectors too rapidly may induce sector erase failures and cause rapid premature device wear out so individual sector erase cycling rates must be controlled.
2. Test routines should use the Flash's status bits (or RY/BY# output) to determine if the flash has completed a erase or program operation. Reliance on a software watchdog timer based on data sheet typical performance often leads to incorrectly flag failures, especially in systems that issue erase and program suspend commands.

3. System test programs and production code should employ wear leveling to distribute erases of individual sectors over a period of time, e.g. at least 90 seconds between erases of a specific sector. Each sector of the Flash is isolated from the other sectors during an erase operation. Therefore erasing sector 1 has no effect on sector 2, so by executing program/erase cycles sector by sector in a daisy-chained fashion, the number of erase cycles is extended and the system better optimizes annealing properties without incurring any increase in test time.

4 Summary

Test and validation program development should take into consideration the properties and characteristics of the Flash architecture for reliable test results. These programs should follow the test recommendations above. By following these guidelines, correct test programs can be developed for proper system level validation of the Flash memory subsystem.

5 Additional Documentation

[AN99121 - Practical Guide to Endurance and Data Retention](#)

Document History Page

Document Title: AN98510 - Developing System-Level Validation Routines Document Number: 001-98510				
Rev.	ECN No.	Orig. of Change	Submission Date	Description of Change
**	—	—	03/20/2007	Initial version
*A	—	—	09/15/2010	Corrected erroneous RDY reference; updated text
*B	4958456	MSWI	10/12/2015	Updated in Cypress template
*C	5868799	JHOE	08/31/2017	Updated link in Additional Information Updated template

Worldwide Sales and Design Support

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturers' representatives, and distributors. To find the office closest to you, visit us at [Cypress Locations](#).

Products

AARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

Cypress Developer Community

[Forums](#) | [WICED IOT Forums](#) | [Projects](#) | [Video](#) | [Blogs](#) |
[Training](#) | [Components](#)

Technical Support

cypress.com/go/support



Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2007-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spanion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spanion, the Spanion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.