



The following document contains information on Cypress products. Although the document is marked with the name "Spansion" and "Fujitsu", the company that originally developed the specification, Cypress will continue to offer these products to new and existing customers.

Continuity of Specifications

There is no change to this document as a result of offering the device as a Cypress product. Any changes that have been made are the result of normal document improvements and are noted in the document history page, where supported. Future revisions will occur when appropriate, and changes will be noted in a document history page.

Continuity of Ordering Part Numbers

Cypress continues to support existing part numbers. To order these products, please use only the Ordering Part Numbers listed in this document.

For More Information

Please contact your local sales office for additional information about Cypress products and solutions.

About Cypress

Cypress (NASDAQ: CY) delivers high-performance, high-quality solutions at the heart of today's most advanced embedded systems, from automotive, industrial and networking platforms to highly interactive consumer and mobile devices. With a broad, differentiated product portfolio that includes NOR flash memories, F-RAM™ and SRAM, Traveo™ microcontrollers, the industry's only PSoC® programmable system-on-chip solutions, analog and PMIC Power Management ICs, CapSense® capacitive touch-sensing controllers, and Wireless BLE Bluetooth® Low-Energy and USB connectivity solutions, Cypress is committed to providing its customers worldwide with consistent innovation, best-in-class support and exceptional system value.

Errata

This errata sheet is for MB9A310K/110K Series FLASH PROGRAMMING MANUAL
Rev.1v0 (MN706-00017-1v0-E).

32-BIT MICROCONTROLLER
FM3 Family
MB9A310K/110K Series
FLASH PROGRAMMING MANUAL

2012.4.19

: Corrected part

Date	Page	Item	Description
2012/4/19	48	CHAPTER 2: 3.3.2.	<p>Figure 3-2 should be changed as indicated by the shading below.</p> <p>(Error)</p> <pre> graph TD Start([Start of writing]) --> SetASZ[Set the ASZ bit of WorkFlash access size register (WFASZR) to "0"] SetASZ --> ReadWFASZR[Read WorkFlash access size register (WFASZR) (Dummy)] ReadWFASZR --> WriteCmd[Write command sequence 1. Addr:000X_1550 Data:XXAA 2. Addr:000X_XAA8 Data:XX55 3. Addr:000X_1550 Data:XXA0 4. Write Address Write Data] WriteCmd --> Start </pre> <p>(Correct)</p> <pre> graph TD Start([Start of writing]) --> SetASZ[Set the ASZ bit of WorkFlash access size register (WFASZR) to "0"] SetASZ --> ReadWFASZR[Read WorkFlash access size register (WFASZR) (Dummy)] ReadWFASZR --> WriteCmd[Write command sequence 1. Addr:000X_XAA8 Data:XXAA 2. Addr:000X_X554 Data:XX55 3. Addr:000X_XAA8 Data:XXA0 4. Write Address Write Data] WriteCmd --> Start </pre> <p>[mcu doc1160]</p>

Date	Page	Item	Description
2012/4/19	52	CHAPTER 2: 3.3.4.	<p>Figure 3-3 should be changed as indicated by the shading below.</p> <p>(Error)</p> <pre> graph TD Start([Start of erase]) --> SetASZ[Set the ASZ bit of WorkFlash access size register (WFASZR) to "0"] SetASZ --> ReadWFASZR[Read WorkFlash access size register (WFASZR) (Dummy)] ReadWFASZR --> Seq[Sector erase command sequence 1. Addr:000X_1550 Data:XXAA 2. Addr:000X_XAA8 Data:XX55 3. Addr:000X_1550 Data:XX80 4. Addr:000X_1550 Data:XXAA 5. Addr:000X_XAA8 Data:XX55] Seq --> ReadWFASZR </pre> <p>(Correct)</p> <pre> graph TD Start([Start of erase]) --> SetASZ[Set the ASZ bit of WorkFlash access size register (WFASZR) to "0"] SetASZ --> ReadWFASZR[Read WorkFlash access size register (WFASZR) (Dummy)] ReadWFASZR --> Seq[Sector erase command sequence 1. Addr:000X_XAA8 Data:XXAA 2. Addr:000X_X554 Data:XX55 3. Addr:000X_XAA8 Data:XX80 4. Addr:000X_XAA8 Data:XXAA 5. Addr:000X_X554 Data:XX55] Seq --> ReadWFASZR </pre> <p style="text-align: right;">[mcu_doc1160]</p>
2012/4/19	58	CHAPTER 2: 4.1.	<p>Notes should be changed as indicated by the shading below.</p> <p>(Error)</p> <hr/> <p><Notes></p> <ul style="list-style-type: none"> • When ASZ is set to "01", always perform writes to flash using half-word access (16-bit access). • Do not change this register using an instruction that is contained in the flash memory. Overwrite this register from a program in any other area except for flash memory. • Perform a dummy read to register, after changing this register. <hr/> <p>(Correct)</p> <hr/> <p><Notes></p> <ul style="list-style-type: none"> • When ASZ is set to "0", always perform writes to flash using half-word access (16-bit access). • Do not change this register using an instruction that is contained in the flash memory. Overwrite this register from a program in any other area except for flash memory. • Perform a dummy read to register, after changing this register. <hr/> <p style="text-align: right;">[mcu_doc1160]</p>