

32-BIT MICROCONTROLLER  
**MB91460 SERIES**

---

**PRECAUTIONS ON READING  
FLASH STATUS DURING  
PROGRAMMING**

CUSTOMER INFORMATION

## Revision History

Date	Issue
2008-08-14	V0.1, Initial Version, JWa
2008-08-20	V0.2, review, JWa
2008-08-20	V1.0, release, JWa

This document contains 5 pages.

### Abbreviations:

FME                Fujitsu Microelectronics Europe GmbH  
 MCU                Microcontroller

---

# Contents

<b>REVISION HISTORY .....</b>	<b>2</b>
<b>CONTENTS .....</b>	<b>3</b>
<b>1 PROBLEM DESCRIPTION .....</b>	<b>4</b>
<b>2 PROBLEM CONDITIONS .....</b>	<b>4</b>
<b>3 AFFECTED DEVICES.....</b>	<b>4</b>
<b>4 ROOT CAUSE .....</b>	<b>4</b>
<b>5 WORKAROUND .....</b>	<b>5</b>
5.1 Use DPOLL rather than TOGGLE .....	5
5.2 Use software workaround .....	5

## 1 Problem Description

A problem was found in the embedded flash auto program algorithms using the TOGGLE bit function for write/erase determination on all MB91460 flash series MCUs. The problem occurs when the status of the hardware sequence flags is read twice by two consecutive accesses. In that case, the TOGGLE bit (bit6) does not toggle, although the auto write/erase algorithm is not finished. A function may then falsely detect the end of the auto algorithm and go on with further programming/erasing which may lead to corrupt data.

The problem does not occur if at least one CLKB cycle between the two read accesses is inserted (e.g. NOP).

## 2 Problem Conditions

Problem occurs if all of the following conditions are met:

- Auto write or erase algorithm of the internal flash is started by the appropriate command sequence
- Two consecutive read operations are done from the same or different address within this flash (sector). The read instructions are not interrupted by other instructions. For example:

LD @R4, R0 ; R4 holds the address of erased/programmed flash sector

LD @R4, R1

→ TOGGLE Bit (bit6) of R0 and R1 content is not toggled in this case

- The problem is independent of the core clock (CLKB).

## 3 Affected Devices

All MB91460 flash devices

## 4 Root Cause

The bit toggle generation is derived from the flash access signal OEX. This signal goes low with each flash access, goes back to high when the access is finished. If two consecutive flash accesses occur, the OEX signal does not perform a low high low transition – it stays low for the two accesses. Caused by this, the TOGGLE bit of the flash status is not modified.

## 5 Workaround

### 5.1 Use DPOLL rather than TOGGLE

Instead of checking the TOGGLE bit by comparing two status values, the DPOLL bit can be used to check the completion of an auto erase/program algorithm. For the usage of the DPOLL bit, please refer to the latest hardware manual.

### 5.2 Use software workaround

To prevent the occurring of the described behaviour, simply insert one NOP instruction between the two consecutive read accesses.

```
LD @R4, R0 ; R4 holds the address of erased/programmed flash sector
```

```
NOP
```

```
LD @R4, R1
```

This should also be done, if the code is written in C language. Just to be sure, that the optimized code will not produce this behaviour.