July 1, 2009

C8051F55x/56x/57x Revision A Errata

Errata Status Summary

This document summarizes all known errata with Revision A of the C8051F55x/56x/57x devices.

Errata #	Title	Impact	Status
			Revision A Device
1	SMBus Hardware ACK behavior	Major	Issue Exists
2	Set FLSCL when Flash writing/erasing	Minor	Issue Exists
3	VDD Monitor/VDD Regulator Interaction	Minor	Issue Exists

Impact Definition: Each erratum is marked with an impact, as defined below:

- Minor—Workaround exists without any impact to device capability.
- Major—Workaround does not exist, or the workaround limits the capabilities of the device.
- Information—The device behavior is not ideal but acceptable. Typically, the data sheet will be changed to match the device behavior.

Errata Details

1. **Description**: The Address Hardware Acknowledge mechanism of the SMBus peripheral can cause an unexpected SMBus interrupt or cause an incorrect SMBus state transition. The behavior depends on the EXTHOLD bit in the SMB0CF register.

   a) When Hardware Acknowledge is enabled (EHACK = 1b, SMB0ADM) and SDA setup and hold times are not extended (EXTHOLD = 0, SMB0CF), the SMBus hardware will generate an SMBus interrupt, whether or not the address on the bus matches the hardware address match conditions. The expected behavior is that an interrupt is only generated when the address matches. When the MCU enters the interrupt service routine, the SMBus peripheral will be in the appropriate state and indicate the reception of a slave address.

   b) When Hardware Acknowledge is enabled (EHACK = 1b, SMB0ADM) and SDA setup and hold times are extended (EXTHOLD = 1, SMB0CF) the SMBus hardware will incorrectly clear the Start bit (STA) on reception of a slave address, which causes the firmware to interpret the state as the "Slave Receiver -- Data Byte received" state. This will only happen when the address match conditions determined by the SMB0ADR and SMB0MASK registers are met by the address presented on the bus.

   c) When Hardware Acknowledge is enabled (EHACK = 1b, SMB0ADM) and the ACK bit is set to 1, an unaddressed slave may cause interference on the SMBus by driving SDA low during an ACK cycle. The ACK bit may be set to 1 if any device on the bus generates an ACK.

   Impacts:

   a) Once the CPU enters the interrupt service routine, SCL will be asserted low until SI is cleared. Incompliant SMBus masters that do not support SCL clock stretching will not recognize that the clock is

being stretched. If the received address does not match the conditions of SMB0ADR and SMB0MASK, the slave will generate a NACK. If the CPU issues a write to SMB0DAT, it will have no effect on the bus. No data collisions will occur.

b) Once the hardware has matched an address and entered the interrupt service routine, the firmware will not be able to use the Start bit to distinguish between the reception of an address byte versus the reception of a data byte. However, the hardware will still correctly acknowledge the address byte (SLA+R/W).

c) The SMBus master and addressed slave are not able to generate a NACK since the unaddressed slave is holding SDA low during the ACK cycle. There is a potential for the SMBus to lock up.

Workaround:

a) The SMBus interrupt service routine should verify an address when it is received and clear SI as soon as possible if the address does not match.

b) It is recommended that setup and hold times should not be extended when Hardware Acknowledge is enabled. Contact mcuapps@silabs.com for alternate workarounds if these two features are required.

c) Schedule a timer interrupt to clear the ACK bit at an interval shorter than 7 bit periods when the slave is not being addressed. For example, on a 400 kHz SMBus, the ACK bit should be cleared every 17.5 µs (or at 1/7 the bus frequency, 57 kHz).

As soon as a matching slave address is detected, the timer which clears the ACK bit should be stopped and its interrupt flag cleared. The timer should be re-started once a stop condition is detected.

Contact mcuapps@silabs.com for a code example demonstrating this workaround.

Resolution: All issues will be corrected in Revision B.

2. **Description**: When writing or erasing Flash from application firmware, the lower four bits of the FLSCL register must be set to 0010b instead of 0000b as stated in the data sheet. This ensures that the Flash is programmed properly.

   Impact: The Flash Write and Erase times are extended when FLSCL is written to 0010b. The extended times are as follows:

Parameters	Conditions	Min	Typ	Max	Units
Erase Cycle Time	25 MHz System Clock	28	30	45	ms
Write Cycle Time	25 MHz System Clock	79	84	125	µs

   Workaround: None.

   Resolution: In Revision B, the lower four bits of FLSCL can be safely set to 0000b and the Flash write/erase times will match the data sheet.

3. **Description**: There is an interaction between the VDD Monitor and the Voltage Regulator that causes some devices to be held in reset. The VDD Monitor threshold has a low setting (default) and a high setting. The VDD Monitor threshold setting is persistent after all device resets except for a power-on reset. The Voltage Regulator output is self-calibrated after a device is released from reset.

   On the affected devices, the uncalibrated Voltage Regulator output is below the high VDD monitor threshold. On these devices, with the VDD Monitor configured to the high threshold and enabled as a reset source, when a reset occurs, the device is held in reset by the VDD Monitor until a power-on reset.

The uncalibrated Voltage Regulator output triggers the VDD Monitor to hold the device in reset, preventing the calibration sequence of the regulator.

Workaround:

A. Preventing the Issue

One option to prevent this issue is to not use the high setting for the VDD Monitor in conjunction with the internal regulator. Note that the high setting for the VDD Monitor is required to perform in-application Flash programming. If in-application Flash programming is required, perform the following steps:

1. Disable the VDD Monitor as a reset source in SFR RSTSRC.
2. Configure the VDD Monitor to the high threshold using SFR VDDMON.
3. Wait for the required VDD monitor stabilization time.
4. Check the VDDSTAT bit in VDDMON to confirm that the VDD voltage is above the threshold.
5. If the VDD voltage is above the threshold, enable the VDD Monitor as a reset source in RSTSRC.
6. Perform the Flash write/erase.
7. Disable the VDD Monitor as a reset source.
8. Configure the VDD Monitor to the low threshold.
9. Re-enable the VDD Monitor as a reset source in RSTSRC.

Contact MCU Applications at www.silabs.com/support for example firmware.

The issue is also preventable by using an external voltage source for VDD instead of the output of the Voltage Regulator. When an external voltage source is used for VDD, firmware should disable the internal voltage regulator. In this configuration, it is safe to set the VDD Monitor to the high threshold.

B. Recovering a Device

On the affected devices, it is not possible to connect to or reprogram the MCU using the Silicon Labs IDE once firmware is loaded that sets the VDD monitor to the high threshold. The Silicon Labs IDE C2 connection sequence first resets device before putting the device into the debug state. This C2 reset triggers the issue, preventing the IDE from connecting to the device.

To recover a device that will not connect to the Silicon Labs IDE due to this issue, override the regulator using an external voltage supply. Use an external voltage supply that is higher than the high VDD monitor threshold. This will prevent the supply monitor from holding the device in reset, and the Silicon Labs IDE can then connect to the device and erase code space using the Tools → Erase Code Space menu option.

Another option to recover a device is to use the Silicon Labs Device Erase program. Contact MCU Applications at www.silabs.com/support to obtain the program.

Resolution: The next revision of the C8051F55x/56x/57x data sheet will include details of the issue and the workarounds. The current version of the data sheet is Revision 0.5.