# Wireless Gecko
# EFR32FG28 Errata

This document contains information on the EFR32FG28 errata. The latest available revision of this device is revision A.

Errata that have been resolved remain documented and can be referenced for previous revisions of this device.

The device data sheet explains how to identify the chip revision, either from the package marking or electronically.

Errata effective date: February, 2023.

# 1. Errata Summary

The table below lists all known errata for the EFR32FG28 and all unresolved errata of the EFR32FG28.

**Table 1.1. Errata Overview**

| Designator | Title/Problem | Workaround Exists | Exists on Revision: |
|:---:|:---|:---:|:---:|
| | | | A |
| LCD_E301 | LOADBUSY Status Goes Inactive Early With Prescaled Clock | Yes | X |

## 2. Current Errata Descriptions

### 2.1 LCD_E301 — LOADBUSY Status Goes Inactive Early With Prescaled Clock

| Description of Errata |
| --- |
| The LCD_STATUS_LOADBUSY bit erroneously reports completion of writes to the LCD_BACTRL, LCD_AREGA, LCD_AREGB, and LCD_SEGDn registers before synchronization is complete when LCD_CTRL_PRESCALE > 3. |

| Affected Conditions / Impacts |
| --- |
| If LOADBUSY is used to gate consecutive writes to one of the affected registers, only the data associated with the last write is guaranteed to be latched into the register. |

| Workaround |
| --- |
| For each write to one of the affected registers, insert a delay equal to LCD_CTRL_PRESCALE ÷ $f_{LCDCLK}$ after LOADBUSY transitions from 1 to 0 before issuing the next write to the same register. |
| **Note:** LOADBUSY reports when data written from the PCLK register domain into the LCD controller's low-frequency clock domain has been synchronized. It does not indicate when data written into one of the affected registers is actually driven on the LCD controller pins. |
| In cases where writes to these registers, such as LCD_SEGDn, are intended to have the change in pin state be observable on the connected display, LOADBUSY should not be used to gate consecutive writes. Instead, the CPU should issue the register write and wait to issue the next write until a display update event or frame counter update event occurs as reported by the LCD_IF register DISPLAY or FC flag bits. Interrupts associated with these flags can and should be enabled in such cases to minimize energy use by keeping the CPU in a low-energy mode (e.g., EM2) between such consecutive register writes. |

| Resolution |
| --- |
| There is currently no resolution for this issue. |

# 3. Revision History

**Revision 0.1**

February, 2023
- Initial release.

# Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

**IoT Portfolio**
www.silabs.com/IoT

**SW/HW**
www.silabs.com/simplicity

**Quality**
www.silabs.com/quality

**Support & Community**
www.silabs.com/community

**Silicon Laboratories Inc.**
**400 West Cesar Chavez**
**Austin, TX 78701**
**USA**

# SILICON LABS

**www.silabs.com**