AN0018.0: Supply Voltage Monitoring

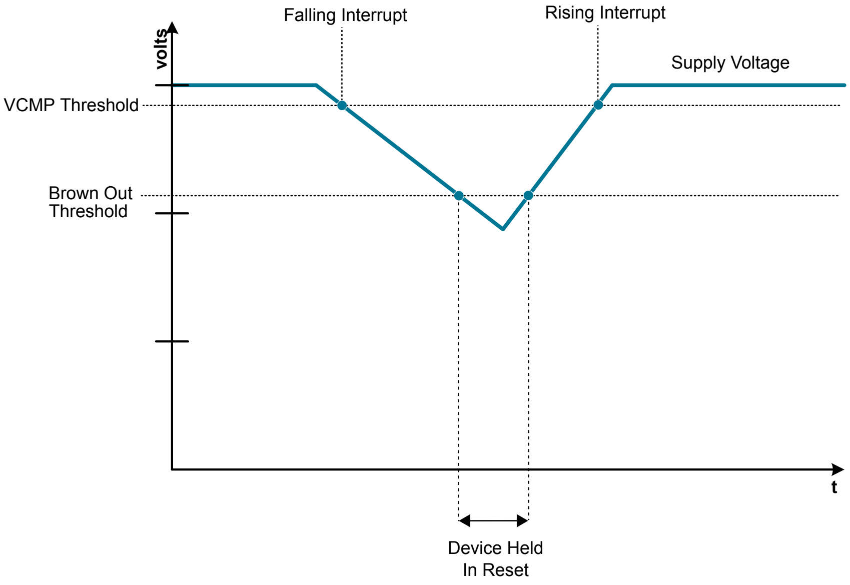
This application note will demonstrate how to use the EFM32 Reset Management Unit and the Voltage Comparator to read the reset cause register and monitor the supply voltage.

This application note includes:

- This PDF document
- Source files (zip)
  - Example c-code
  - Multiple IDE projects

KEY POINTS

- Observe RMU to determine if a brown-out has occurred
- Monitor power rails for low voltage conditions with VCMP
- Low-current voltage monitoring down to EM3 with VCMP



1. Device Compatibility

This application note supports multiple device families, and some functionality is different depending on the device.

MCU Series 0 consists of:

- EFM32 Gecko (EFM32G)
- EFM32 Giant Gecko (EFM32GG)
- EFM32 Wonder Gecko (EFM32WG)
- EFM32 Leopard Gecko (EFM32LG)
- EFM32 Tiny Gecko (EFM32TG)
- EFM32 Zero Gecko (EFM32ZG)
- EFM32 Happy Gecko (EFM32HG)

Wireless MCU Series 0 consists of:

- EZR32 Wonder Gecko (EZR32WG)
- EZR32 Leopard Gecko (EZR32LG)
- EZR32 Happy Gecko (EZR32HG)

2. Reset Management Unit

The Reset Management Unit (RMU) monitors the reset lines and ensures reliable operation by resetting the microcontroller if the any of the supply voltages are insufficient or if certain events occur. Such events are activation of the external reset pin, watchdog timeout, kernel lockup or a system reset request. In addition, the RMU sets the reset cause register which can be used to determine the cause of the last reset at startup.

2.1 Power-on Reset and Brown-out Detection

During power up, the device is held in reset until the supply voltage exceeds the threshold voltage, $V_{BODextthr+}$, which is approximately 1.8V. Below this threshold, the voltage levels are undefined and safe operation can not be guaranteed. Once the voltage level rises above $V_{BODextthr+}$, the power-on reset line is released and the microcontroller is powered up.

If the supply voltage falls below the threshold voltage during operation, $V_{BODextthr-}$, a "brown-out" has occurred. The brown-out detector will pull the microcontroller into reset to prevent unexpected program execution and data corruption. EFM32 and EZR32 Series 0 devices feature two brown-out detectors, one for the external unregulated power supply and one for the regulated 1.8V rail.

Once the RMU initiates a reset, the reset cause register is written to specify the reset source.

2.2 Reading the Reset Cause Register

The reset cause register indicates which reset source triggered the last reset. Because several bits can be set at once and multiple reset causes can occur simultaneously, it is important to look for the reset cause in the following order to discover the main cause of reset:

1. Power-on reset
2. Unregulated power brown-out reset
3. Regulated power brown-out reset
4. External and watchdog reset
5. Lockup and system request reset

When following this list from the smallest to largest number, the first indicated reset cause will be the actual reset cause. The reset cause register needs to be cleared after it has been read, otherwise it may indicate a wrong cause of reset at the next startup. To clear the reset cause register a 1 has to be written to the RCCLR bit in RMU_CMD and the HRCCLR bit in EMU_AUXCTRL.

The EFM32 software library contains functions to read and clear the reset cause register.

3. Voltage Comparator

The voltage comparator (VCMP) provides an easy way to warn if the supply voltage reaches a critical level. While the RMU automatically initiates a reset if the voltage level is too low, the voltage comparator compares the supply voltage to an internal bandgap reference. The supply voltage can be scaled in 64 steps to trigger the comparator output on voltage levels between 1.8 and 3.8 V. By adjusting the trigger level it is possible to take action before the microcontroller enters a brown-out reset or to verify that the supply voltage is within a specified range.

The voltage comparator output can be read from the VCMPOUT bit in VCMP_STATUS. VCMP can also be configured to give interrupt on either falling and/or rising edge when the supply voltage crosses the threshold voltage. The voltage comparator is fully functional in EM0 to EM3.

3.1 Warm-up

When enabling the voltage comparator, the output is undefined for a period of time, which is called the warm-up time (typically 10 µs). To avoid any glitches on the output during this period, the value of INACTVAL is selected as output value instead of the actual comparator output until warm-up is complete. An edge interrupt will occur if the output level toggles in the transition from INACTVAL to the voltage comparator output value. Setting the INACTVAL bit to the expected comparator output value therefore avoids this initial interrupt. Note that the device must stay in EM0 or EM1 during the warm-up.

3.2 Low Power Reference

Enabling the low power reference (LPREF) will significantly reduce the power consumption compared to active mode at the cost of accuracy. Refer to the device datasheet for typical values. To ensure that a correct reference value is set, LPREF should be disabled until comparator warm-up is complete. When LPREF is enabled it is recommended to enable the half bias bit and set the bias configuration register to one of the lowest settings. Too high bias current can cause the reference value to drift and thus trigger the comparator output at higher voltage level than intended.

3.3 Voltage threshold levels

The trigger level is configured by setting the TRIGLEVEL bits in the VCMP_INPUTSEL register. VCMP can be set to trigger at any voltage level supported by the EFM32 in steps of 0.034v, according to:

$$\text{TRIGLEVEL} = \frac{\left(V_{\text{trig}} - 1.667\right)}{0.034}$$

Figure 3.1. VCMP Trigger Level Equation

If the MCU supply voltage is higher than V_{trig}, the comparator output is 1 and 0 otherwise.

The EFM32 software library includes a function to calculate TRIGLEVEL from a given voltage. Note that during type conversion the resulting TRIGLEVEL is rounded down to the nearest integer.

3.4 Response Time

Changing the trigger level can be done while VCMP is enabled, but it is necessary to wait for the new input value to propagate through the comparator before using VCMPOUT. This delay is called the comparator response time and depends on the bias current configuration. Higher bias settings will shorten the response time. The bias settings are controlled by the BIASPROG and HALFBIAS bits in VCMP_CTRL. Checking if the comparator output value has been updated can be done by re-enabling VCMP. Disabling and enabling the voltage comparator will cause a new warm-up cycle to begin.

3.5 Threshold hysteresis

Small fluctuations in the supply voltage can be filtered out by enabling hysteresis on the comparator input. If enabled, the comparator will not trigger before the supply voltage level is approximately 20mV above or below the reference voltage.

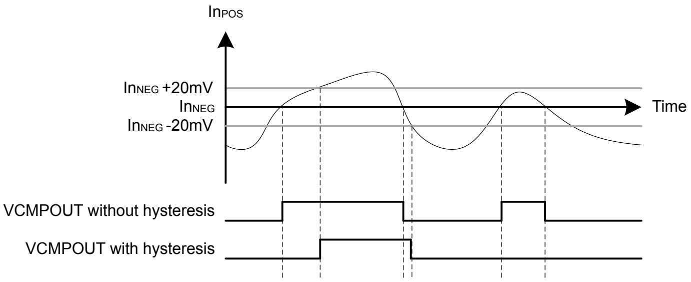


Figure 3.2. Voltage Comparator Hysteresis

3.6 Energy Optimization Considerations

The voltage comparator consumes as little as 100nA while enabled with the lowest bias settings. Using an interrupt driven approach is simple and ensures that a critical low voltage will be detected before the MCU is forced into a brown-out reset. However, some energy sensitive applications running on battery will usually see a slowly decreasing supply voltage. Having the voltage comparator constantly enabled is therefore unnecessary. A solution is to enable the voltage comparator on regular intervals to check the output voltage level and leave the voltage comparator disabled in between. If the application features a real-time clock or similar, the voltage comparator can be sampled at specified intervals.

4. Software Examples

The following examples shows how to read and clear the reset cause register and use the voltage comparator to monitor the supply voltage.

4.1 Example 1: Reset cause register and interrupt driven VCMP

In this example the reset cause register is read and cleared immediately after startup. The voltage comparator is initialized and the MCU is entering EM1 to wait for comparator warm-up to finish. Because the warm-up period is given as a number of clock cycles, a lower energy mode can not be used as HFPERCLK is disabled below EM1. The MCU then enters EM2 waiting for the supply voltage to trigger an interrupt.

This example is limited to EM2 to allow the LCD to be active. Without the LCD-part this example will also work in EM3.

4.2 Example 2: Polled supply voltage monitoring

This example constantly polls the voltage comparator output to check if the supply voltage is at a certain level. In a real application, the polling interval can be increased to reduce the current consumption. The trigger level alternates between a high and low limit to determine if the supply voltage is within the allowed range. After changing the trigger level it is necessary to wait for the comparator response time to pass to ensure that the correct comparator output value is used.

5. Revision History

Revision 1.09

November, 2017

- Split appnote into Series 0 and Series 1 versions.
- Updated to new appnote format.

Revision 1.08

May, 2014

- Updated example code to CMSIS 3.20.5.
- Changed to Silicon Labs license on code examples.
- Added example projects for Simplicity IDE.
- Removed example makefiles for Sourcery CodeBench Lite.
- Updated project files with shorter filenames.

Revision 1.07

October, 2013

- New cover layout.

Revision 1.06

May, 2013

- Added software projects for ARM-GCC and Atollic TrueStudio.

Revision 1.05

November, 2012

- Adapted software projects to new kit-driver and bsp structure.
- Added software support for Tiny and Giant Gecko STKs.

Revision 1.04

April, 2012

- Adapted software projects to new peripheral library naming and CMSIS_V3.

Revision 1.03

October, 2011

- Updated IDE project paths with new kits directory.

Revision 1.02

May, 2011

- Updated projects to align with new bsp version.

Revision 1.01

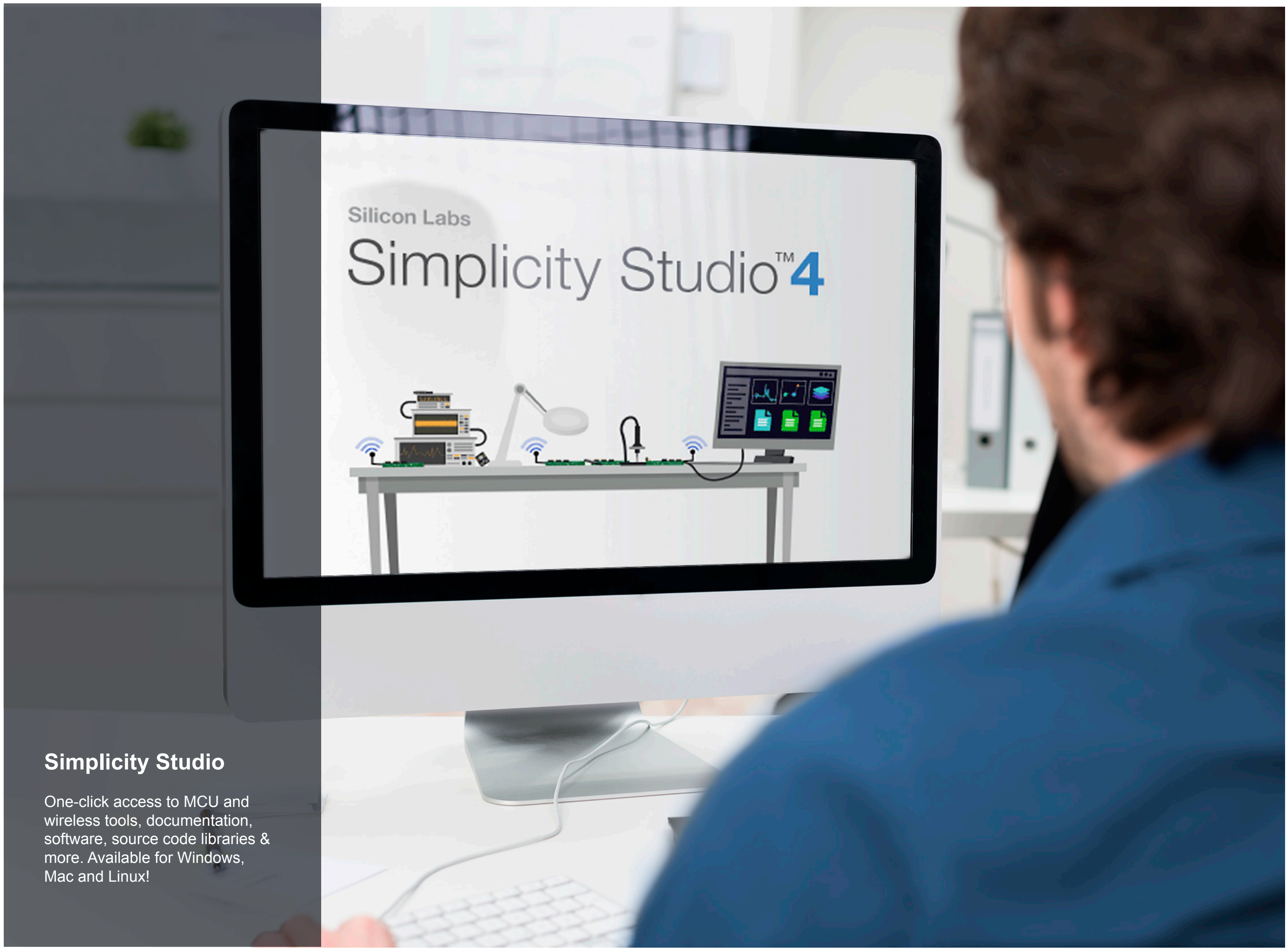
November, 2010

- Changed example folder structure, removed build and src folders.
- Added chip-init function.
- Updated register defines in code to match newest efm32lib release.

Revision 1.00

September, 2010

- Initial revision.



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 and Community
community.silabs.com

Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information

Silicon Laboratories Inc.® , Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOmodem®, Micrium, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress®, Zentri and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.

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

http://www.silabs.com