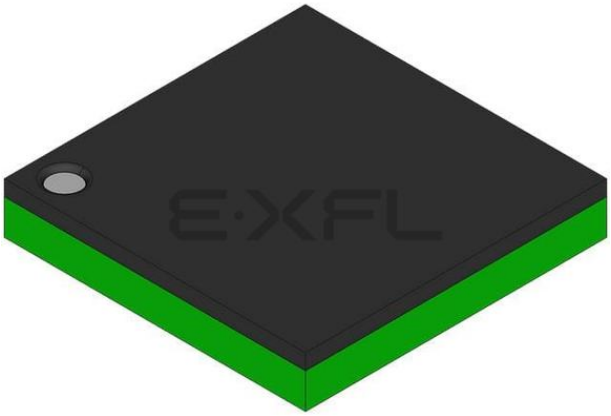


**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | AVR |
| Core Size | 8/16-Bit |
| Speed | 32MHz |
| Connectivity | I²C, IrDA, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, DMA, POR, PWM, WDT |
| Number of I/O | 34 |
| Program Memory Size | 32KB (16K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | 1K x 8 |
| RAM Size | 4K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.6V ~ 3.6V |
| Data Converters | A/D 12x12b; D/A 2x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 49-VFBGA |
| Supplier Device Package | 49-VFBGA (5x5) |
| Purchase URL | https://www.e-xfl.com/product-detail/atmel/atxmega32a4-cu |

# 3. Overview

The Atmel® AVR® XMEGA™A4 is a family of low power, high performance and peripheral rich CMOS 8/16-bit microcontrollers based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the XMEGA A4 achieves throughputs approaching 1 Million Instructions Per Second (MIPS) per MHz allowing the system designer to optimize power consumption versus processing speed.

The AVR CPU combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction, executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs many times faster than conventional single-accumulator or CISC based microcontrollers.

The XMEGA A4 devices provide the following features: In-System Programmable Flash with Read-While-Write capabilities, Internal EEPROM and SRAM, four-channel DMA Controller, eight-channel Event System, Programmable Multi-level Interrupt Controller, 34 general purpose I/O lines, 16-bit Real Time Counter (RTC), five flexible 16-bit Timer/Counters with compare modes and PWM, five USARTs, two Two Wire Serial Interfaces (TWIs), two Serial Peripheral Interfaces (SPIs), AES and DES crypto engine, one Twelve-channel, 12-bit ADC with optional differential input with programmable gain, one Two-channel 12-bit DAC, two analog comparators with window mode, programmable Watchdog Timer with separate Internal Oscillator, accurate internal oscillators with PLL and prescaler and programmable Brown-Out Detection.

The Program and Debug Interface (PDI), a fast 2-pin interface for programming and debugging, is available.

The XMEGA A4 devices have five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, DMA Controller, Event System, Interrupt Controller and all peripherals to continue functioning. The Power-down mode saves the SRAM and register contents but stops the oscillators, disabling all other functions until the next TWI or pin-change interrupt, or Reset. In Power-save mode, the asynchronous Real Time Counter continues to run, allowing the application to maintain a timer base while the rest of the device is sleeping. In Standby mode, the Crystal/Resonator Oscillator is kept running while the rest of the device is sleeping. This allows very fast start-up from external crystal combined with low power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run. To further reduce power consumption, the peripheral clock to each individual peripheral can optionally be stopped in Active mode and in Idle sleep mode.

The device is manufactured using Atmel's high-density nonvolatile memory technology. The program Flash memory can be reprogrammed in-system through the PDI. A Bootloader running in the device can use any interface to download the application program to the Flash memory. The Bootloader software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8/16-bit RISC CPU with In-System Self-Programmable Flash, the Atmel XMEGA A4 is a powerful microcontroller family that provides a highly flexible and cost effective solution for many embedded applications.

The XMEGA A4 devices are supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, programmers, and evaluation kits.

# 7. Memories

## 7.1 Features

- **Flash Program Memory**
  - **One linear address space**
  - **In-System Programmable**
  - **Self-Programming and Bootloader support**
  - **Application Section for application code**
  - **Application Table Section for application code or data storage**
  - **Boot Section for application code or bootloader code**
  - **Separate lock bits and protection for all sections**
  - **Built in fast CRC check of a selectable flash program memory section**
- **Data Memory**
  - **One linear address space**
  - **Single cycle access from CPU**
  - **SRAM**
  - **EEPROM**
    - **Byte and page accessible**
    - **Optional memory mapping for direct load and store**
  - **I/O Memory**
    - **Configuration and Status registers for all peripherals and modules**
    - **16 bit-accessible General Purpose Register for global variables or flags**
  - **Bus arbitration**
    - **Safe and deterministic handling of CPU and DMA Controller priority**
  - **Separate buses for SRAM, EEPROM, I/O Memory and External Memory access**
    - **Simultaneous bus access for CPU and DMA Controller**
- **Production Signature Row Memory for factory programmed data**
  - **Device ID for each microcontroller device type**
  - **Serial number for each device**
  - **Oscillator calibration bytes**
  - **ADC, DAC and temperature sensor calibration data**
- **User Signature Row**
  - **One flash page in size**
  - **Can be read and written from software**
  - **Content is kept after chip erase**

## 7.2 Overview

The AVR architecture has two main memory spaces, the Program Memory and the Data Memory. In addition, the XMEGA A4 features an EEPROM Memory for non-volatile data storage. All three memory spaces are linear and require no paging. The available memory size configurations are shown in ”Ordering Information” on page 2. In addition each device has a Flash memory signature row for calibration data, device identification, serial number etc.

Non-volatile memory spaces can be locked for further write or read/write operations. This prevents unrestricted access to the application software.

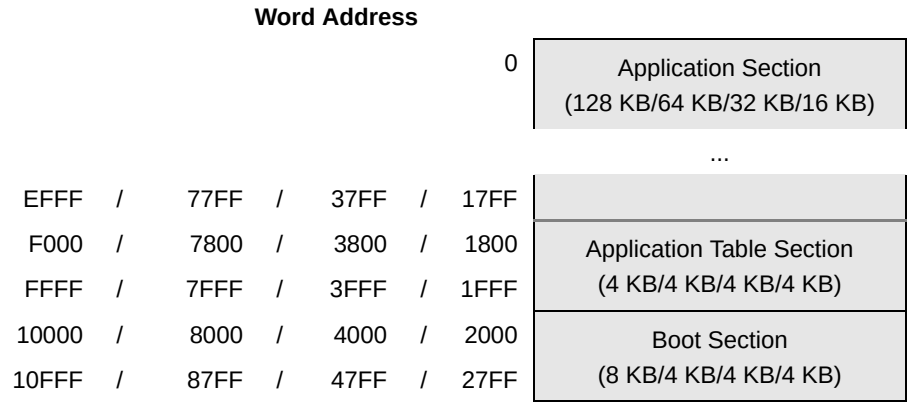
## 7.3 In-System Programmable Flash Program Memory

The XMEGA A4 devices contain On-chip In-System Programmable Flash memory for program storage, see Figure 7-1 on page 11. Since all AVR instructions are 16- or 32-bits wide, each Flash address location is 16 bits.

The Program Flash memory space is divided into Application and Boot sections. Both sections have dedicated Lock Bits for setting restrictions on write or read/write operations. The Store Program Memory (SPM) instruction must reside in the Boot Section when used to write to the Flash memory.

A third section inside the Application section is referred to as the Application Table section which has separate Lock bits for storage of write or read/write protection. The Application Table section can be used for storing non-volatile data or application software.

**Figure 7-1.** Flash Program Memory (Hexadecimal address)




The Application Table Section and Boot Section can also be used for general application software.

## 7.7 Flash and EEPROM Page Size

The Flash Program Memory and EEPROM data memory are organized in pages. The pages are word accessible for the Flash and byte accessible for the EEPROM.

Table 7-2 on page 14 shows the Flash Program Memory organization. Flash write and erase operations are performed on one page at a time, while reading the Flash is done one byte at a time. For Flash access the Z-pointer (Z[m:n]) is used for addressing. The most significant bits in the address (FPAGE) give the page number and the least significant address bits (FWORD) give the word in the page.

**Table 7-2.** Number of words and Pages in the Flash.

| Devices | Flash Size | Page Size (words) | FWORD | FPAGE | Application | | Boot | |
|---|---|---|---|---|---|---|---|---|
| | | | | | Size | No of Pages | Size | No of Pages |
| ATxmega16A4 | 16 KB + 4 KB | 128 | Z[6:0] | Z[13:7] | 16 KB | 64 | 4 KB | 16 |
| ATxmega32A4 | 32 KB + 4 KB | 128 | Z[6:0] | Z[14:7] | 32 KB | 128 | 4 KB | 16 |
| ATxmega64A4 | 64 KB + 4 KB | 128 | Z[6:0] | Z[15:7] | 64 KB | 128 | 4 KB | 16 |
| ATxmega128A4 | 128 KB + 8 KB | 256 | Z[7:0] | Z[16:8] | 128 KB | 256 | 8 KB | 16 |

Table 7-3 on page 14 shows EEPROM memory organization for the XMEGA A4 devices. EEPROM write and erase operations can be performed one page or one byte at a time, while reading the EEPROM is done one byte at a time. For EEPROM access the NVM Address Register (ADDR[m:n]) is used for addressing. The most significant bits in the address (E2PAGE) give the page number and the least significant address bits (E2BYTE) give the byte in the page.

**Table 7-3.** Number of Bytes and Pages in the EEPROM.

| Devices | EEPROM Size | Page Size (Bytes) | E2BYTE | E2PAGE | No of Pages |
|---|---|---|---|---|---|
| ATxmega16A4 | 1 KB | 32 | ADDR[4:0] | ADDR[10:5] | 32 |
| ATxmega32A4 | 1 KB | 32 | ADDR[4:0] | ADDR[10:5] | 32 |
| ATxmega64A4 | 2 KB | 32 | ADDR[4:0] | ADDR[10:5] | 64 |
| ATxmega128A4 | 2 KB | 32 | ADDR[4:0] | ADDR[10:5] | 64 |

### 10.3.3 32.768 kHz Crystal Oscillator

The 32.768 kHz Crystal Oscillator is a low power driver for an external watch crystal. It can be used as system clock source or as asynchronous clock source for the Real Time Counter.

### 10.3.4 0.4 - 16 MHz Crystal Oscillator

The 0.4 - 16 MHz Crystal Oscillator is a driver intended for driving both external resonators and crystals ranging from 400 kHz to 16 MHz.

### 10.3.5 2 MHz Run-time Calibrated Internal Oscillator

The 2 MHz Run-time Calibrated Internal Oscillator is a high frequency oscillator. It is calibrated during production to provide a default frequency which is close to its nominal frequency. The oscillator can use the 32 kHz Calibrated Internal Oscillator or the 32 kHz Crystal Oscillator as a source for calibrating the frequency run-time to compensate for temperature and voltage drift hereby optimizing the accuracy of the oscillator.

### 10.3.6 32 MHz Run-time Calibrated Internal Oscillator

The 32 MHz Run-time Calibrated Internal Oscillator is a high frequency oscillator. It is calibrated during production to provide a default frequency which is close to its nominal frequency. The oscillator can use the 32 kHz Calibrated Internal Oscillator or the 32 kHz Crystal Oscillator as a source for calibrating the frequency run-time to compensate for temperature and voltage drift hereby optimizing the accuracy of the oscillator.

### 10.3.7 External Clock input

The external clock input gives the possibility to connect a clock from an external source.

### 10.3.8 PLL with Multiplication factor 1 - 31x

The PLL provides the possibility of multiplying a frequency by any number from 1 to 31. In combination with the prescalers, this gives a wide range of output frequencies from all clock sources.

# 11. Power Management and Sleep Modes

## 11.1 Features

- **5 sleep modes**
  - **Idle**
  - **Power-down**
  - **Power-save**
  - **Standby**
  - **Extended standby**
- **Power Reduction registers to disable clocks to unused peripherals**

## 11.2 Overview

The XMEGA A4 provides various sleep modes tailored to reduce power consumption to a minimum. All sleep modes are available and can be entered from Active mode. In Active mode the CPU is executing application code. The application code decides when and what sleep mode to enter. Interrupts from enabled peripherals and all enabled reset sources can restore the microcontroller from sleep to Active mode.

In addition, Power Reduction registers provide a method to stop the clock to individual peripherals from software. When this is done, the current state of the peripheral is frozen and there is no power consumption from that peripheral. This reduces the power consumption in Active mode and Idle sleep mode.

## 11.3 Sleep Modes

### 11.3.1 Idle Mode

In Idle mode the CPU and Non-Volatile Memory are stopped, but all peripherals including the Interrupt Controller, Event System and DMA Controller are kept running. Interrupt requests from all enabled interrupts will wake the device.

### 11.3.2 Power-down Mode

In Power-down mode all system clock sources, and the asynchronous Real Time Counter (RTC) clock source, are stopped. This allows operation of asynchronous modules only. The only interrupts that can wake up the MCU are the Two Wire Interface address match interrupts, and asynchronous port interrupts, e.g pin change.

### 11.3.3 Power-save Mode

Power-save mode is identical to Power-down, with one exception: If the RTC is enabled, it will keep running during sleep and the device can also wake up from RTC interrupts.

### 11.3.4 Standby Mode

Standby mode is identical to Power-down with the exception that all enabled system clock sources are kept running, while the CPU, Peripheral and RTC clocks are stopped. This reduces the wake-up time when external crystals or resonators are used.

# 15. I/O Ports

## 15.1 Features

- **Selectable input and output configuration for each pin individually**
- **Flexible pin configuration through dedicated Pin Configuration Register**
- **Synchronous and/or asynchronous input sensing with port interrupts and events**
  - **Sense both edges**
  - **Sense rising edges**
  - **Sense falling edges**
  - **Sense low level**
- **Asynchronous wake-up from all input sensing configurations**
- **Two port interrupts with flexible pin masking**
- **Highly configurable output driver and pull settings:**
  - **Totem-pole**
  - **Pull-up/-down**
  - **Wired-AND**
  - **Wired-OR**
  - **Bus-keeper**
  - **Inverted I/O**
- **Optional Slew rate control**
- **Configuration of multiple pins in a single operation**
- **Read-Modify-Write (RMW) support**
- **Toggle/clear/set registers for Output and Direction registers**
- **Clock output on port pin**
- **Event Channel 0 output on port pin 7**
- **Mapping of port registers (virtual ports) into bit accessible I/O memory space**

## 15.2 Overview

The XMEGA A4 devices have flexible General Purpose I/O Ports. A port consists of up to 8 pins, ranging from pin 0 to pin 7. The ports implement several functions, including synchronous/asynchronous input sensing, pin change interrupts and configurable output settings. All functions are individual per pin, but several pins may be configured in a single operation.

## 15.3 I/O configuration

All port pins (Pn) have programmable output configuration. In addition, all port pins have an inverted I/O function. For an input, this means inverting the signal between the port pin and the pin register. For an output, this means inverting the output signal between the port register and the port pin. The inverted I/O function can be used also when the pin is used for alternate functions.

# 18. Hi-Res - High Resolution Extension

## 18.1 Features

- **Increases Waveform Generator resolution by 2-bits (4x)**
- **Supports Frequency, single- and dual-slope PWM operation**
- **Supports the AWEX when this is enabled and used for the same Timer/Counter**

## 18.2 Overview

The Hi-Resolution (Hi-Res) Extension is able to increase the resolution of the waveform generation output by a factor of 4. When enabled for a Timer/Counter, the Fast Peripheral clock running at four times the CPU clock speed will be as input to the Timer/Counter.

The High Resolution Extension can also be used when an AWEX is enabled and used with a Timer/Counter.

XMEGA A4 devices have three Hi-Res Extensions that each can be enabled for each Timer/Counters pair on PORTC, PORTD and PORTE. The notation of these are HIRESC, HIRESD and HIRESE, respectively.

# 19. RTC - 16-bit Real-Time Counter

## 19.1 Features

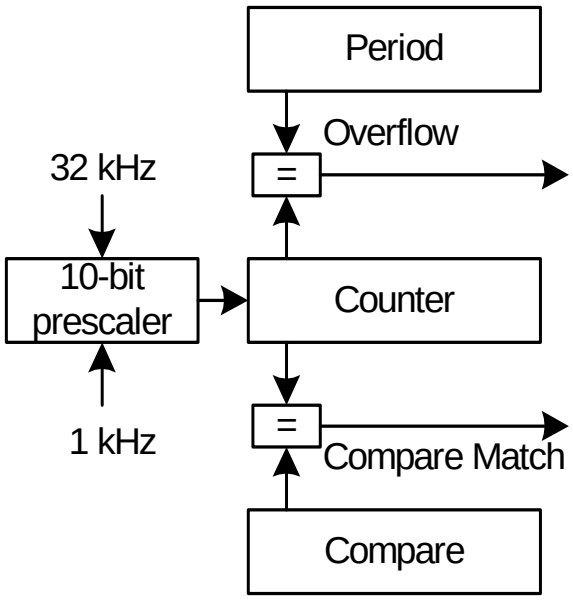
- **16-bit Timer**
- **Flexible Tick resolution ranging from 1 Hz to 32.768 kHz**
- **One Compare register**
- **One Period register**
- **Clear timer on Overflow or Compare Match**
- **Overflow or Compare Match event and interrupt generation**

## 19.2 Overview

The XMEGA A4 includes a 16-bit Real-time Counter (RTC). The RTC can be clocked from an accurate 32.768 kHz Crystal Oscillator, the 32.768 kHz Calibrated Internal Oscillator, or from the 32 kHz Ultra Low Power Internal Oscillator. The RTC includes both a Period and a Compare register. For details, see Figure 19-1.

A wide range of Resolution and Time-out periods can be configured using the RTC. With a maximum resolution of 30.5 µs, time-out periods range up to 2000 seconds. With a resolution of 1 second, the maximum time-out period is over 18 hours (65536 seconds).

**Figure 19-1.** Real Time Counter overview



# 29. Program and Debug Interfaces

## 29.1 Features

- **PDI - Program and Debug Interface (Atmel proprietary 2-pin interface)**
- **Access to the OCD system**
- **Programming of Flash, EEPROM, Fuses and Lock Bits**

## 29.2 Overview

The programming and debug facilities are accessed through PDI physical interface. The PDI physical interface uses one dedicated pin together with the Reset pin, and no general purpose pins are used.

## 29.3 PDI - Program and Debug Interface

The PDI is an Atmel proprietary protocol for communication between the microcontroller and Atmel's development tools.

### 30.1.5 Communication functions

| | |
|---|---|
| SCL | Serial Clock for TWI |
| SDA | Serial Data for TWI |
| XCKn | Transfer Clock for USART n |
| RXDn | Receiver Data for USART n |
| TXDn | Transmitter Data for USART n |
| $\overline{SS}$ | Slave Select for SPI |
| MOSI | Master Out Slave In for SPI |
| MISO | Master In Slave Out for SPI |
| SCK | Serial Clock for SPI |

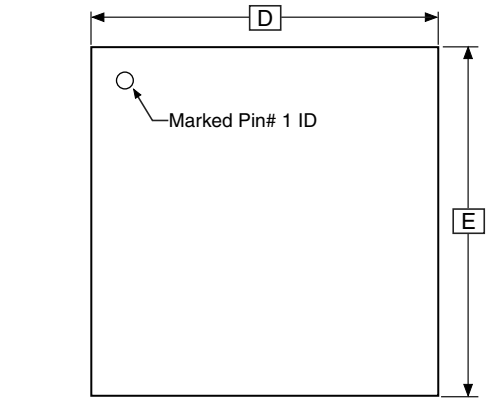
### 30.1.6 Oscillators, Clock and Event

| | |
|---|---|
| TOSCn | Timer Oscillator pin n |
| XTALn | Input/Output for inverting Oscillator pin n |

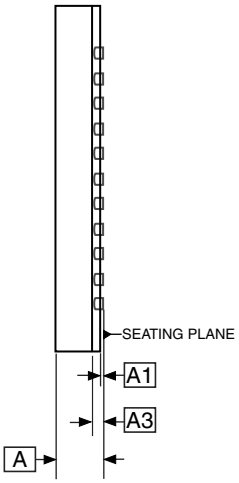
### 30.1.7 Debug/System functions

| | |
|---|---|
| $\overline{RESET}$ | Reset pin |
| PDI_CLK | Program and Debug Interface Clock pin |
| PDI_DATA | Program and Debug Interface Data pin |

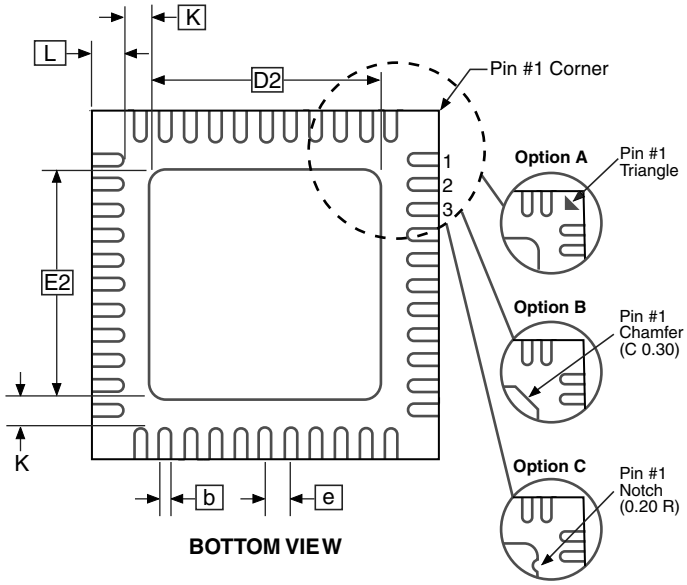
## 33.2 44M1




TOP VIEW




SIDE VIEW




BOTTOM VIEW

Note: JEDEC Standard MO-220, Fig. 1 (SAW Singulation) VKKD-3.

**COMMON DIMENSIONS**
(Unit of Measure = mm)

| SYMBOL | MIN | NOM | MAX | NOTE |
|---|---|---|---|---|
| A | 0.80 | 0.90 | 1.00 | |
| A1 | – | 0.02 | 0.05 | |
| A3 | | 0.20 REF | | |
| b | 0.18 | 0.23 | 0.30 | |
| D | 6.90 | 7.00 | 7.10 | |
| D2 | 5.00 | 5.20 | 5.40 | |
| E | 6.90 | 7.00 | 7.10 | |
| E2 | 5.00 | 5.20 | 5.40 | |
| e | | 0.50 BSC | | |
| L | 0.59 | 0.64 | 0.69 | |
| K | 0.20 | 0.26 | 0.41 | |

9/26/08

| | | | | |
|---|---|---|---|---|
| **Package Drawing Contact:** packagedrawings@atmel.com | **TITLE** **44M1,** 44-pad, 7 x 7 x 1.0mm body, lead pitch 0.50mm, 5.20mm exposed pad, thermally enhanced plastic very thin quad flat no lead package (VQFN) | **GPC** ZWS | **DRAWING NO.** 44M1 | **REV.** H |

**Figure 35-19.** Reset Pull-up Resistor Current vs. Reset Pin Voltage

$V_{CC}$ = 3.0V.

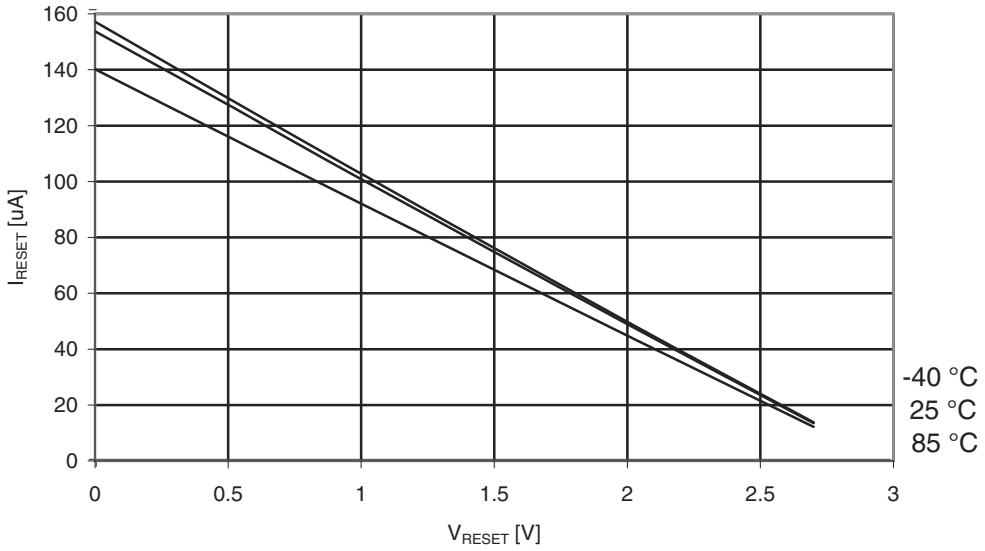
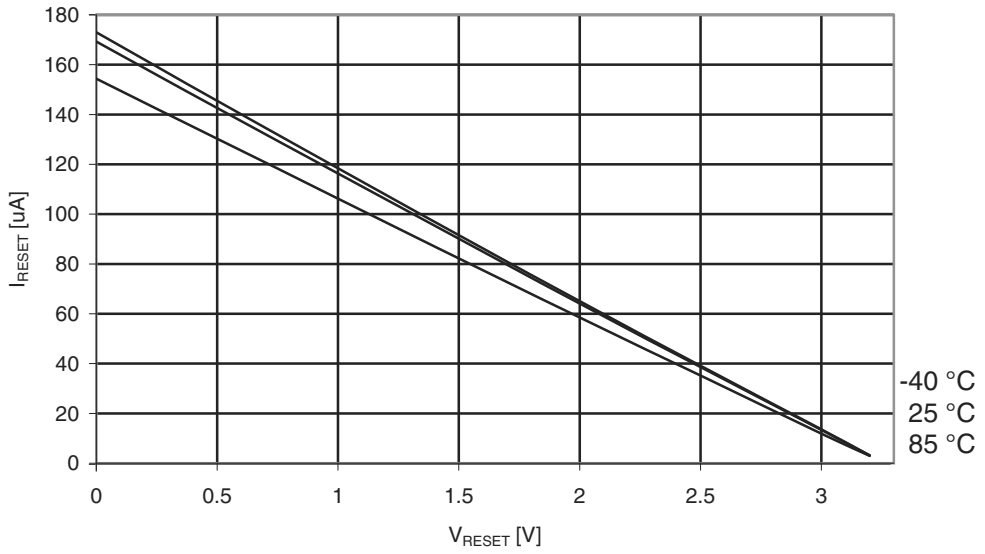



**Figure 35-20.** Reset Pull-up Resistor Current vs. Reset Pin Voltage

$V_{CC}$ = 3.3V.



## 35.7 Pin Thresholds and Hysteresis

**Figure 35-27.** I/O Pin Input Threshold Voltage vs. $V_{CC}$

*$V_{IH}$ - I/O Pin Read as "1".*

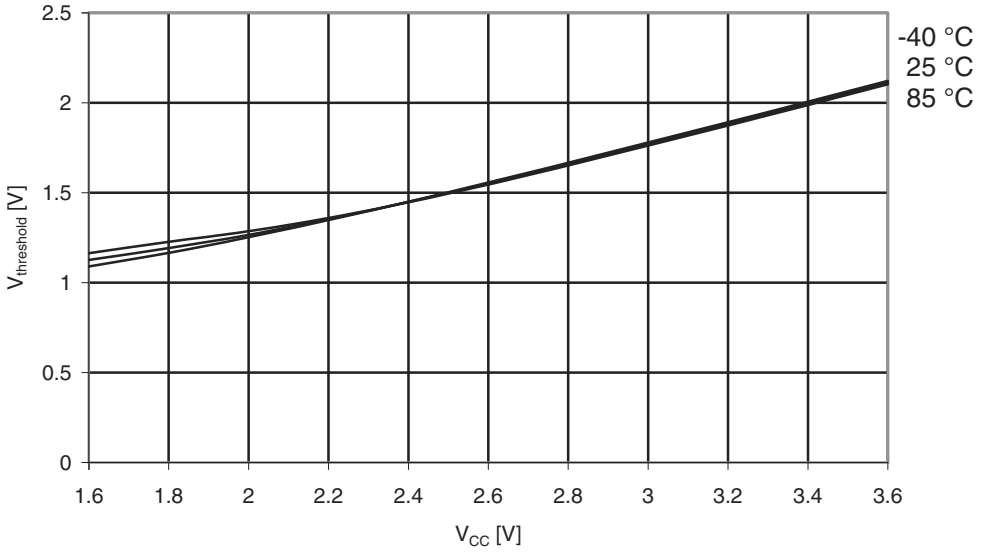



**Figure 35-28.** I/O Pin Input Threshold Voltage vs. $V_{CC}$

*$V_{IL}$ - I/O Pin Read as "0".*

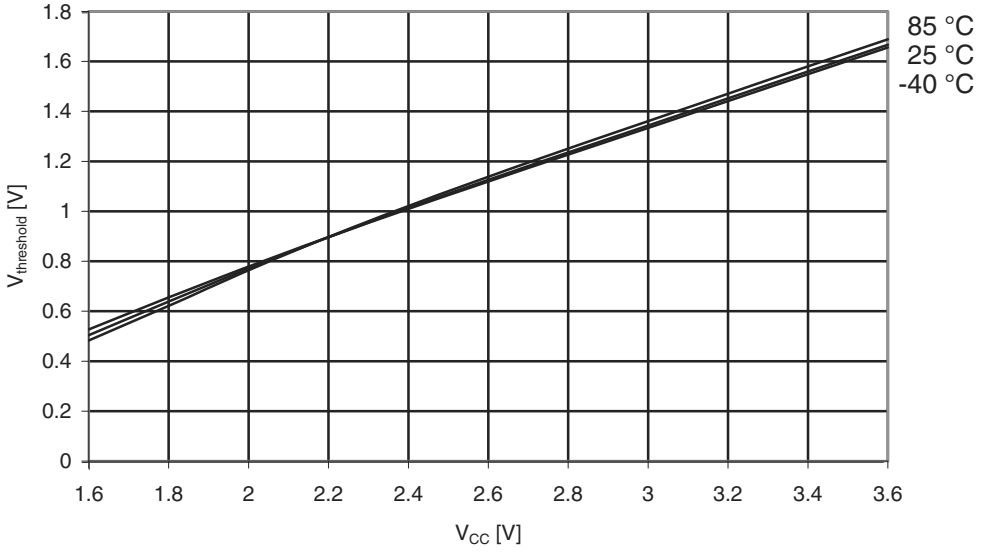


**Figure 35-29.** I/O Pin Input Hysteresis vs. $V_{CC}$.

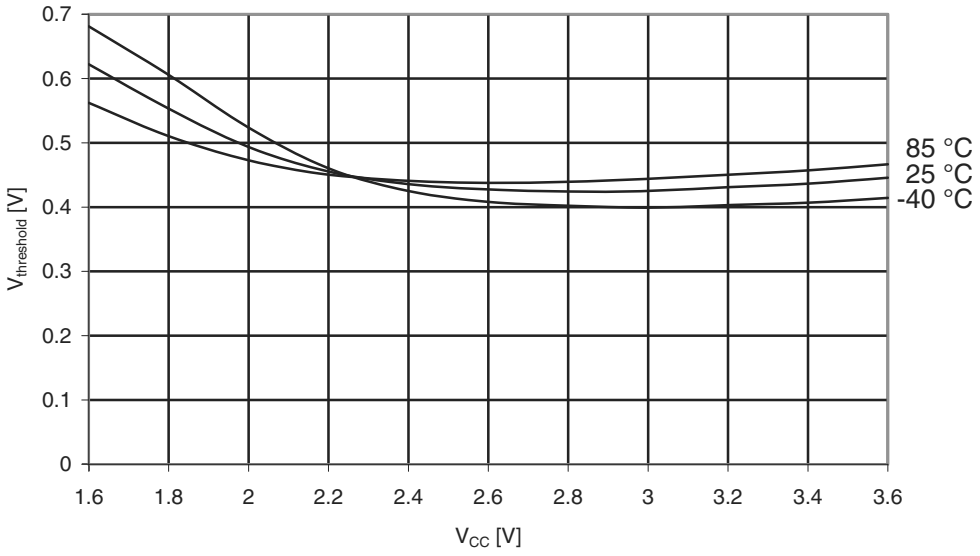



**Figure 35-30.** Reset Input Threshold Voltage vs. $V_{CC}$

*$V_{IH}$ - I/O Pin Read as "1".*

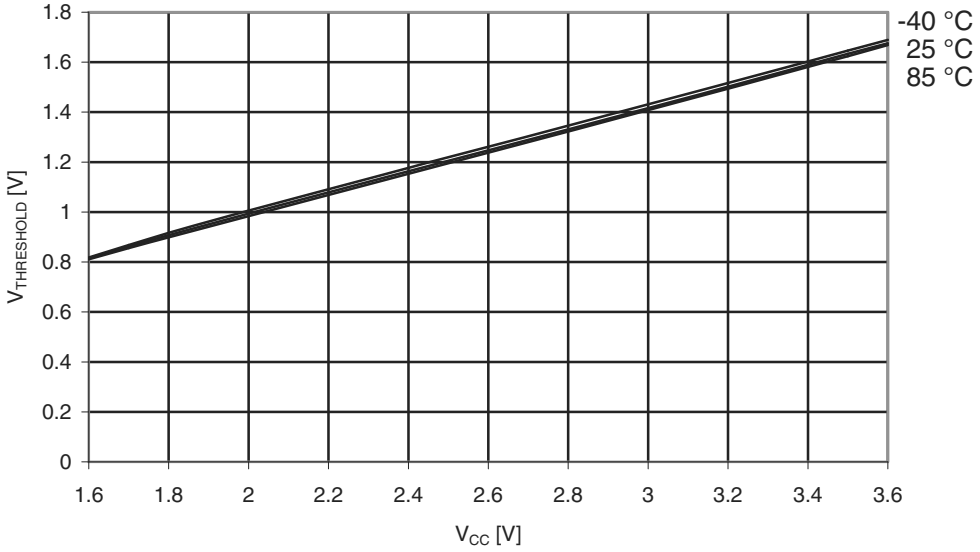
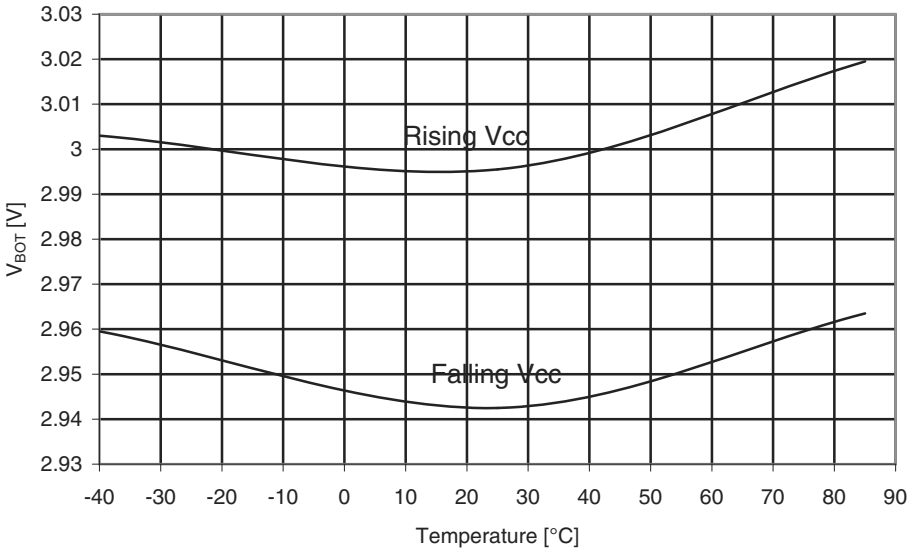


**Figure 35-33.** BOD Thresholds vs. Temperature

*BOD Level = 2.9V.*




## 35.9 Analog Comparator

**Figure 35-34.** Analog Comparator Hysteresis vs. $V_{CC}$

*High-speed, Small hysteresis.*

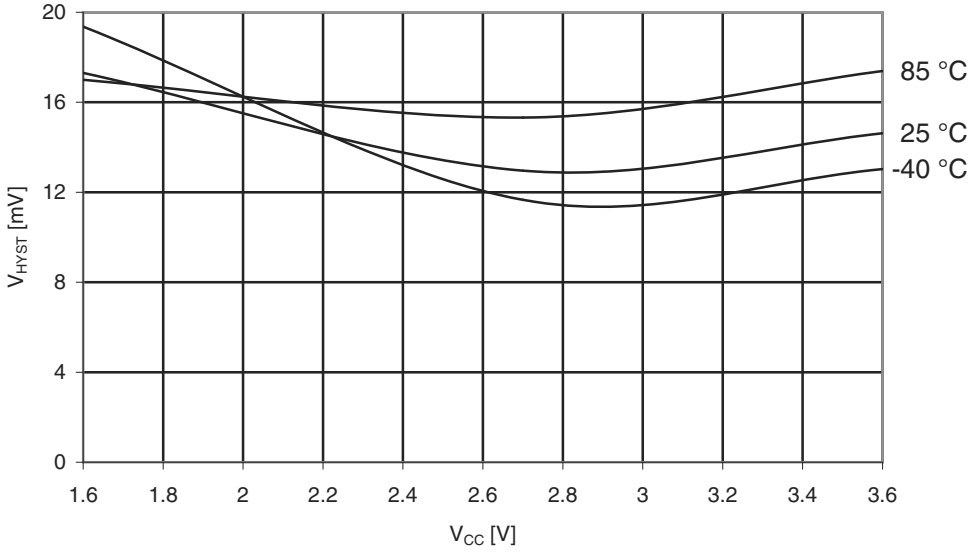


**Figure 35-35.** Analog Comparator Hysteresis vs. $V_{CC}$

*High-speed, Large hysteresis.*

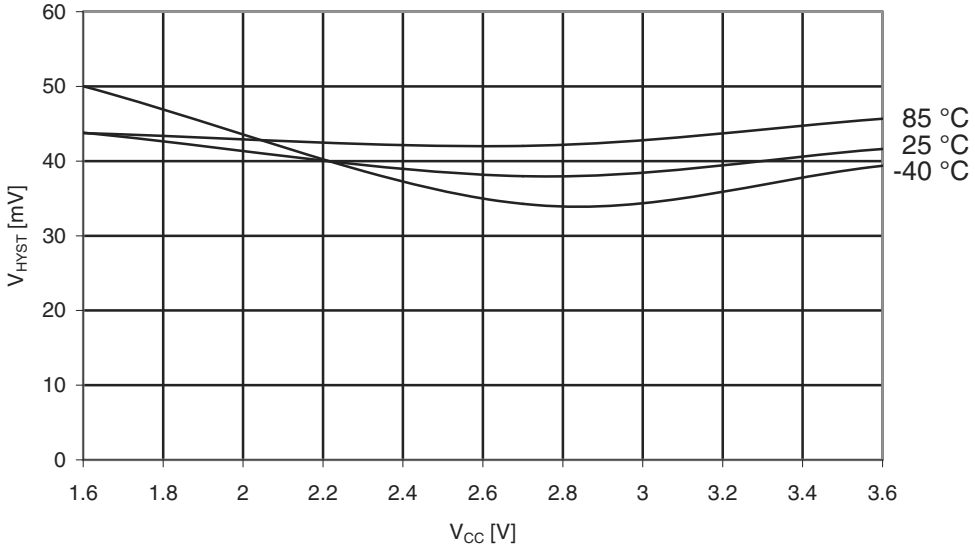



**Figure 35-36.** Analog Comparator Propagation Delay vs. $V_{CC}$

*High-speed.*

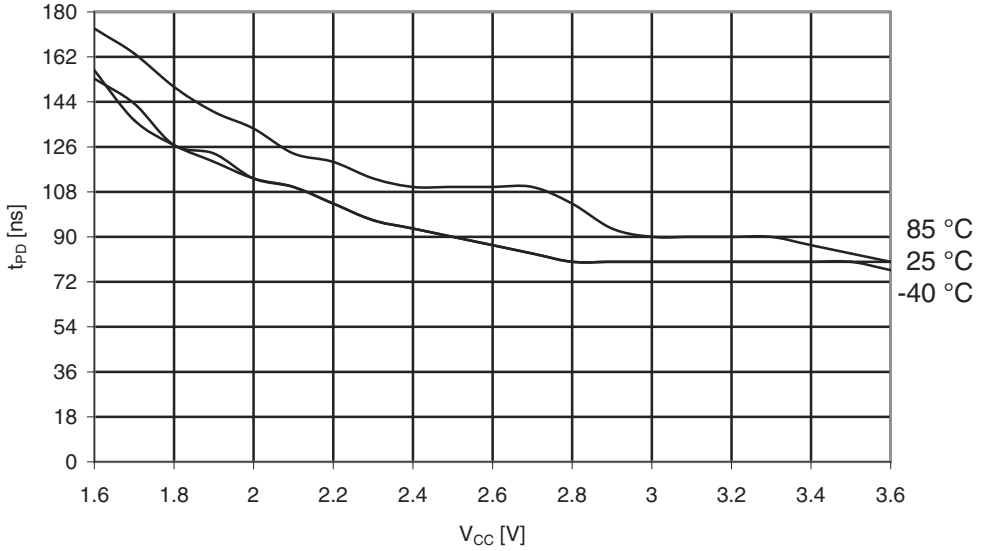
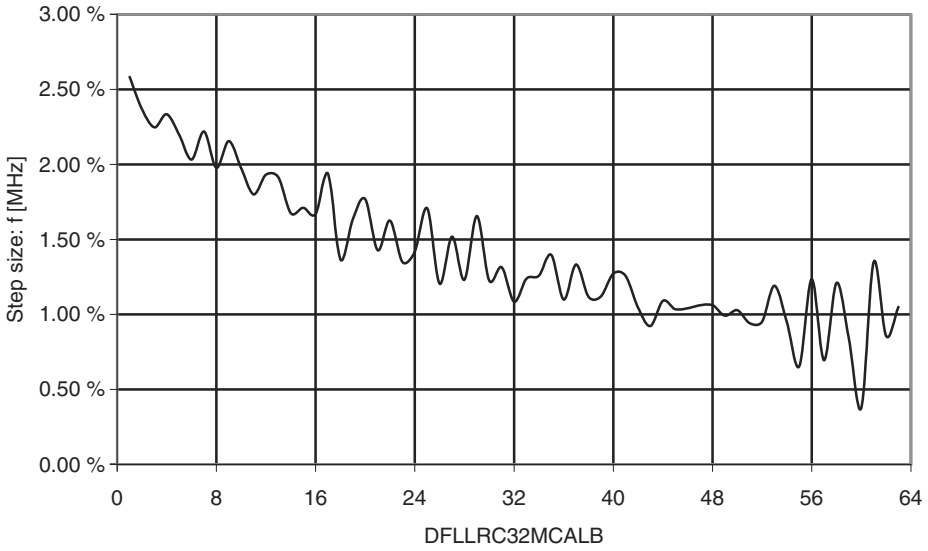


**Figure 35-41.** Internal 32 MHz Oscillator CALB Calibration Step Size

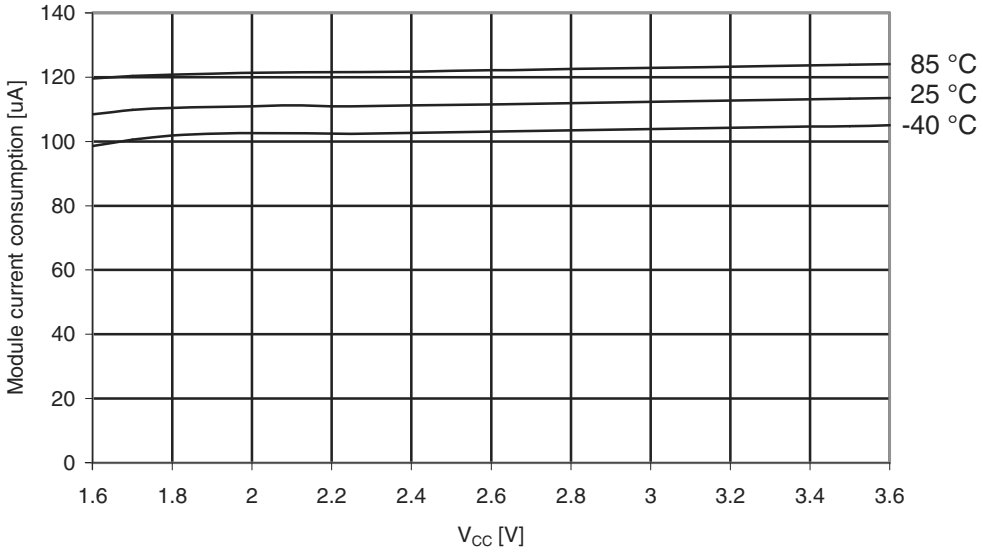
*T = -40 to 85°C, $V_{CC}$ = 3V.*




## 35.11 Module current consumption

**Figure 35-42.** AC current consumption vs. Vcc

*Low-power Mode.*



### 23. Clearing TWI Stop Interrupt Flag may lock the bus

If software clears the STOP Interrupt Flag (APIF) on the same Peripheral Clock cycle as the hardware sets this flag due to a new address received, CLKHOLD is not cleared and the SCL line is not released. This will lock the bus.

**Problem fix/Workaround**

Check if the bus state is IDLE. If this is the case, it is safe to clear APIF. If the bus state is not IDLE, wait for the SCL pin to be low before clearing APIF.

**Code:**

```
/* Only clear the interrupt flag if within a "safe zone". */
while ( /* Bus not IDLE: */
       ((COMMS_TWI.MASTER.STATUS & TWI_MASTER_BUSSTATE_gm) !=
         TWI_MASTER_BUSSTATE_IDLE_gc)) &&
         /* SCL not held by slave: */
         !(COMMS_TWI.SLAVE.STATUS & TWI_SLAVE_CLKHOLD_bm)
      )
{
    /* Ensure that the SCL line is low */
    if ( !(COMMS_PORT.IN & PIN1_bm) )
        if ( !(COMMS_PORT.IN & PIN1_bm) )
            break;
}
/* Check for an pending address match interrupt */
if ( !(COMMS_TWI.SLAVE.STATUS & TWI_SLAVE_CLKHOLD_bm) )
{
    /* Safely clear interrupt flag */
    COMMS_TWI.SLAVE.STATUS |= (uint8_t)TWI_SLAVE_APIF_bm;
}
```

### 24. TWI START condition at bus timeout will cause transaction to be dropped

If Bus Timeout is enabled and a timeout occurs on the same Peripheral Clock cycle as a START is detected, the transaction will be dropped.

**Problem fix/Workaround**

None.

### 25. TWI Data Interrupt Flag erroneously read as set

When issuing the TWI slave response command CMD=0b11, it takes 1 Peripheral Clock cycle to clear the data interrupt flag (DIF). A read of DIF directly after issuing the command will show the DIF still set.

**Problem fix/Workaround**

Add one NOP instruction before checking DIF.

### 26. WDR instruction inside closed window will not issue reset

When a WDR instruction is execute within one ULP clock cycle after updating the window control register, the counter can be cleared without giving a system reset.

**Problem fix/Workaround**

Wait at least one ULP clock cycle before executing a WDR instruction.

## 37.10 8069I – 03/09

1. Updated ”Electrical Characteristics” on page 61.
2. Updated ”Typical Characteristics” on page 70.

## 37.11 8069H – 11/08

1. Updated ”Ordering Information” on page 2.
2. Added VFBGA to ”Pinout/Block Diagram” on page 3.
3. Updated ”Block Diagram” on page 6.
4. Updated feature list in ”Memories” on page 10.
5. Added 49-balls VFBGA to ”Packaging information” on page 58.

## 37.12 8069G – 10/08

1. Updated ”Features” on page 1.
2. Updated ”Ordering Information” on page 2.
3. Replaced the package drawing ”44M1” on page 59 by a rev H update.

## 37.13 8069F – 09/08

1. Updated ”Features” on page 1.
2. Updated ”Ordering Information” on page 2.
3. Updated ”Features” on page 10 by removing “External Memory...”.
4. Updated Figure 7-1 on page 11 and Figure 7-2 on page 12.
5. Updated Table 7-2 on page 14 and Table 7-3 on page 14.
6. Updated ADC ”Features” on page 41 and ”Overview” on page 41.
7. Removed “Interrupt Vector Summary” section from datasheet.

## 37.14 8069E – 08/08

1. Changed Figure 2-1’s title to “Bock Diagram and TQFP/QFN pinout” .
2. Updated Table 30-6 on page 52.