**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 | EBI/EMI, I²C, IrDA, SPI, UART/USART, USB |
| Peripherals | Brown-out Detect/Reset, DMA, POR, PWM, WDT |
| Number of I/O | 78 |
| Program Memory Size | 128KB (64K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | 2K x 8 |
| RAM Size | 8K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.6V ~ 3.6V |
| Data Converters | A/D 16x12b; D/A 4x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 100-TFBGA |
| Supplier Device Package | 100-CBGA (9x9) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/atxmega128a1u-cur |

**Figure 7-2. Data memory map (hexadecimal address).**

| Byte Address | ATxmega64A1U |
|---|---|
| 0 | I/O Registers (4K) |
| FFF | |
| 1000 | EEPROM (2K) |
| 17FF | |
| | RESERVED |
| 2000 | Internal SRAM (4K) |
| 2FFF | |
| 4000 | External Memory (0 to 16MB) |
| FFFFFF | |

| Byte Address | ATxmega128A1U |
|---|---|
| 0 | I/O Registers (4KB) |
| FFF | |
| 1000 | EEPROM (2K) |
| 17FF | |
| | RESERVED |
| 2000 | Internal SRAM (8K) |
| 3FFF | |
| 3000 | External Memory (0 to 16MB) |
| FFFFFF | |

## 7.6 EEPROM

XMEGA AU devices have EEPROM for nonvolatile data storage. It is either addressable in a separate data space (default) or memory mapped and accessed in normal data space. The EEPROM supports both byte and page access. Memory mapped EEPROM allows highly efficient EEPROM reading and EEPROM buffer loading. When doing this, EEPROM is accessible using load and store instructions. Memory mapped EEPROM will always start at hexadecimal address 0x1000.

## 7.7 I/O Memory

The status and configuration registers for peripherals and modules, including the CPU, are addressable through I/O memory locations. All I/O locations can be accessed by the load (LD/LDS/LDD) and store (ST/STS/STD) instructions, which is used to transfer data between the 32 registers in the register file and the I/O memory. The IN and OUT instructions can address I/O memory locations in the range 0x00 - 0x3F directly. In the address range 0x00 - 0x1F, single- cycle instructions for manipulation and checking of individual bits are available.

The I/O memory address for all peripherals and modules in XMEGA A1U is shown in the “Peripheral Module Address Map” on page 63.

### 7.7.1 General Purpose I/O Registers

The lowest 16 I/O memory addresses are reserved as general purpose I/O registers. These registers can be used for storing global variables and flags, as they are directly bit-accessible using the SBI, CBI, SBIS, and SBIC instructions.

## 7.8 External Memory

Four ports can be used for external memory, supporting external SRAM, SDRAM, and memory mapped peripherals such as LCD displays. Refer to “EBI – External Bus Interface” on page 48. The external memory address space will always start at the end of internal SRAM.

## 7.9 Data Memory and Bus Arbitration

Since the data memory is organized as four separate sets of memories, the different bus masters (CPU, DMA controller read and DMA controller write, etc.) can access different memory sections at the same time.

# 11. Power Management and Sleep Modes

## 11.1 Features

- Power management for adjusting power consumption and functions
- Five sleep modes
  - Idle
  - Power down
  - Power save
  - Standby
  - Extended standby
- Power reduction register to disable clock and turn off unused peripherals in active and idle modes

## 11.2 Overview

Various sleep modes and clock gating are provided in order to tailor power consumption to application requirements. This enables the Atmel AVR XMEGA microcontroller to stop unused modules to save power.

All sleep modes are available and can be entered from active mode. In active mode, the CPU is executing application code. When the device enters sleep mode, program execution is stopped and interrupts or a reset is used to wake the device again. The application code decides which sleep mode to enter and when. 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 modes and enables much more fine-tuned power management than sleep modes alone.

## 11.3 Sleep Modes

Sleep modes are used to shut down modules and clock domains in the microcontroller in order to save power. XMEGA microcontrollers have five different sleep modes tuned to match the typical functional stages during application execution. A dedicated sleep instruction (SLEEP) is available to enter sleep mode. Interrupts are used to wake the device from sleep, and the available interrupt wake-up sources are dependent on the configured sleep mode. When an enabled interrupt occurs, the device will wake up and execute the interrupt service routine before continuing normal program execution from the first instruction after the SLEEP instruction. If other, higher priority interrupts are pending when the wake-up occurs, their interrupt service routines will be executed according to their priority before the interrupt service routine for the wake-up interrupt is executed. After wake-up, the CPU is halted for four cycles before execution starts.

The content of the register file, SRAM and registers are kept during sleep. If a reset occurs during sleep, the device will reset, start up, and execute from the reset vector.

### 11.3.1 Idle Mode

In idle mode the CPU and nonvolatile memory are stopped (note that any ongoing programming will be completed), but all peripherals, including the interrupt controller, event system and DMA controller are kept running. Any enabled interrupt will wake the device.

### 11.3.2 Power-down Mode

In power-down mode, all clocks, including the real-time counter clock source, are stopped. This allows operation only of asynchronous modules that do not require a running clock. The only interrupts that can wake up the MCU are the two-wire interface address match interrupt, asynchronous port interrupts, and the USB resume interrupt.

# 13. WDT – Watchdog Timer

## 13.1 Features

- Issues a device reset if the timer is not reset before its timeout period
- Asynchronous operation from dedicated oscillator
- 1kHz output of the 32kHz ultra low power oscillator
- 11 selectable timeout periods, from 8ms to 8s
- Two operation modes:
  - Normal mode
  - Window mode
- Configuration lock to prevent unwanted changes

## 13.2 Overview

The watchdog timer (WDT) is a system function for monitoring correct program operation. It makes it possible to recover from error situations such as runaway or deadlocked code. The WDT is a timer, configured to a predefined timeout period, and is constantly running when enabled. If the WDT is not reset within the timeout period, it will issue a microcontroller reset. The WDT is reset by executing the WDR (watchdog timer reset) instruction from the application code.

The window mode makes it possible to define a time slot or window inside the total timeout period during which WDT must be reset. If the WDT is reset outside this window, either too early or too late, a system reset will be issued. Compared to the normal mode, this can also catch situations where a code error causes constant WDR execution.

The WDT will run in active mode and all sleep modes, if enabled. It is asynchronous, runs from a CPU-independent clock source, and will continue to operate to issue a system reset even if the main clocks fail.

The configuration change protection mechanism ensures that the WDT settings cannot be changed by accident. For increased safety, a fuse for locking the WDT settings is also available.

# 14. Interrupts and Programmable Multilevel Interrupt Controller

## 14.1 Features

- Short and predictable interrupt response time
- Separate interrupt configuration and vector address for each interrupt
- Programmable multilevel interrupt controller
  - Interrupt prioritizing according to level and vector address
  - Three selectable interrupt levels for all interrupts: low, medium and high
  - Selectable, round-robin priority scheme within low-level interrupts
  - Non-maskable interrupts for critical functions
- Interrupt vectors optionally placed in the application section or the boot loader section

## 14.2 Overview

Interrupts signal a change of state in peripherals, and this can be used to alter program execution. Peripherals can have one or more interrupts, and all are individually enabled and configured. When an interrupt is enabled and configured, it will generate an interrupt request when the interrupt condition is present. The programmable multilevel interrupt controller (PMIC) controls the handling and prioritizing of interrupt requests. When an interrupt request is acknowledged by the PMIC, the program counter is set to point to the interrupt vector, and the interrupt handler can be executed.

All peripherals can select between three different priority levels for their interrupts: low, medium, and high. Interrupts are prioritized according to their level and their interrupt vector address. Medium-level interrupts will interrupt low-level interrupt handlers. High-level interrupts will interrupt both medium- and low-level interrupt handlers. Within each level, the interrupt priority is decided from the interrupt vector address, where the lowest interrupt vector address has the highest interrupt priority. Low-level interrupts have an optional round-robin scheduling scheme to ensure that all interrupts are serviced within a certain amount of time.

Non-maskable interrupts (NMI) are also supported, and can be used for system critical functions.

## 14.3 Interrupt vectors

The interrupt vector is the sum of the peripheral's base interrupt address and the offset address for specific interrupts in each peripheral. The base addresses for the Atmel AVR XMEGA A1U devices are shown in Table 14-1 on page 27. Offset addresses for each interrupt available in the peripheral are described for each peripheral in the XMEGA AU manual. For peripherals or modules that have only one interrupt, the interrupt vector is shown in Table 14-1 on page 27. The program address is the word address.

**Table 14-1. Reset and interrupt vectors.**

| Program address (base address) | Source | Interrupt description |
|---|---|---|
| 0x000 | RESET | |
| 0x002 | OSCF_INT_vect | Crystal oscillator failure interrupt vector (NMI) |
| 0x004 | PORTC_INT_base | Port C interrupt base |
| 0x008 | PORTR_INT_base | Port R interrupt base |
| 0x00C | DMA_INT_base | DMA controller interrupt base |
| 0x014 | RTC_INT_base | Real time counter interrupt base |
| 0x018 | TWIC_INT_base | Two-Wire interface on port C interrupt base |
| 0x01C | TCC0_INT_base | Timer/counter 0 on port C interrupt base |

# 16. TC0/1 – 16-bit Timer/Counter Type 0 and 1

## 16.1 Features

- Eight 16-bit timer/counters
  - Four timer/counters of type 0
  - Four timer/counters of type 1
- 32-bit timer/counter support by cascading two timer/counters
- Up to four compare or capture (CC) channels
  - Four CC channels for timer/counters of type 0
  - Two CC channels for timer/counters of type 1
- Double buffered timer period setting
- Double buffered capture or compare channels
- Waveform generation:
  - Frequency generation
  - Single-slope pulse width modulation
  - Dual-slope pulse width modulation
- Input capture:
  - Input capture with noise cancelling
  - Frequency capture
  - Pulse width capture
  - 32-bit input capture
- Timer overflow and error interrupt events
- One compare match or input capture interrupt event per CC channel
- Can be used with event system for:
  - Quadrature decoding
  - Count and direction control
  - Capture
- Can be used with DMA and trigger DMA transactions
- High-resolution extension
  - Increases frequency and waveform resolution by 4x (2-bit) or 8x (3-bit)
- Advanced waveform extension
  - Low and high-side output with programmable dead-time insertion (DTI)
  - Event controlled fault protection for safe disabling of external drivers

## 16.2 Overview

Atmel AVR XMEGA devices have a set of eight flexible 16-bit timer/counters (TC). Their capabilities include accurate program execution timing, frequency and waveform generation, and input capture with time and frequency measurement of digital signals. Two timer/counters can be cascaded to create a 32-bit timer/counter with optional 32-bit capture.

A timer/counter consists of a base counter and a set of compare or capture (CC) channels. The base counter can be used to count clock cycles or events. It has direction control and period setting that can be used for timing. The CC channels can be used together with the base counter to do compare match control, frequency generation, and pulse width waveform modulation, as well as various input capture operations. A timer/counter can be configured for either capture or compare functions, but cannot perform both at the same time.

A timer/counter can be clocked and timed from the peripheral clock with optional prescaling or from the event system. The event system can also be used for direction control and capture trigger or to synchronize operations.

There are two differences between timer/counter type 0 and type 1. Timer/counter 0 has four CC channels, and timer/counter 1 has two CC channels. All information related to CC channels 3 and 4 is valid only for timer/counter 0. Only Timer/Counter 0 has the split mode feature that split it into 2 8-bit Timer/Counters with four compare channels each.

# 22. TWI – Two-Wire Interface

## 22.1 Features

- Four identical two-wire interface peripherals
- Bidirectional two-wire communication interface
  - Phillips $I^2C$ compatible
  - System Management Bus (SMBus) compatible
- Bus master and slave operation supported
  - Slave operation
  - Single bus master operation
  - Bus master in multi-master bus environment
  - Multi-master arbitration
- Flexible slave address match functions
  - 7-bit and general call address recognition in hardware
  - 10-bit addressing supported
  - Address mask register for dual address match or address range masking
  - Optional software address recognition for unlimited number of addresses
- Slave can operate in all sleep modes, including power-down
- Slave address match can wake device from all sleep modes, including power-down
- 100kHz and 400kHz bus frequency support
- Slew-rate limited output drivers
- Input filter for bus noise and spike suppression
- Support arbitration between start repeated start and data bit (SMBus)
- Slave arbitration allows support for address resolve protocol (ARP) (SMBus)

## 22.2 Overview

The two-wire interface (TWI) is a bidirectional, two-wire communication interface. It is $I^2C$ and System Management Bus (SMBus) compatible. The only external hardware needed to implement the bus is one pull-up resistor on each bus line.

A device connected to the bus must act as a master or a slave. The master initiates a data transaction by addressing a slave on the bus and telling whether it wants to transmit or receive data. One bus can have many slaves and one or several masters that can take control of the bus. An arbitration process handles priority if more than one master tries to transmit data at the same time. Mechanisms for resolving bus contention are inherent in the protocol.

The TWI module supports master and slave functionality. The master and slave functionality are separated from each other, and can be enabled and configured separately. The master module supports multi-master bus operation and arbitration. It contains the baud rate generator. Both 100kHz and 400kHz bus frequencies are supported. Quick command and smart mode can be enabled to auto-trigger operations and reduce software complexity.

The slave module implements 7-bit address match and general address call recognition in hardware. 10-bit addressing is also supported. A dedicated address mask register can act as a second address match register or as a register for address range masking. The slave continues to operate in all sleep modes, including power-down mode. This enables the slave to wake up the device from all sleep modes on TWI address match. It is possible to disable the address matching to let this be handled in software instead.

The TWI module will detect START and STOP conditions, bus collisions, and bus errors. Arbitration lost, errors, collision, and clock hold on the bus are also detected and indicated in separate status flags available in both master and slave modes.

It is possible to disable the TWI drivers in the device, and enable a four-wire digital interface for connecting to an external TWI bus driver. This can be used for applications where the device operates from a different $V_{CC}$ voltage than used by the TWI bus.

**Table 33-11. Port R - alternate functions.**

| PORT R | PIN # | INTERRUPT | PDI | XTAL |
|---|---|---|---|---|
| **PDI** | 89 | | PDI_DATA | |
| $\overline{\text{RESET}}$ | 90 | | PDI_CLOCK | |
| **PRO** | 91 | SYNC | | XTAL2 |
| **PR1** | 92 | SYNC | | XTAL1 |

| Base address | Name | Description |
|---|---|---|
| 0x0A80 | AWEXE | Advanced Waveform Extension on port E |
| 0x0A90 | HIRESE | High Resolution Extension on port E |
| 0x0AA0 | USARTE0 | USART 0 on port E |
| 0x0AB0 | USARTE1 | USART 1 on port E |
| 0x0AC0 | SPIE | Serial Peripheral Interface on port E |
| 0x0B00 | TCF0 | Timer/Counter 0 on port F |
| 0x0B40 | TCF1 | Timer/Counter 1 on port F |
| 0x0B90 | HIRESF | High Resolution Extension on port F |
| 0x0BA0 | USARTF0 | USART 0 on port F |
| 0x0BB0 | USARTF1 | USART 1 on port F |
| 0x0BC0 | SPIF | Serial Peripheral Interface on port F |

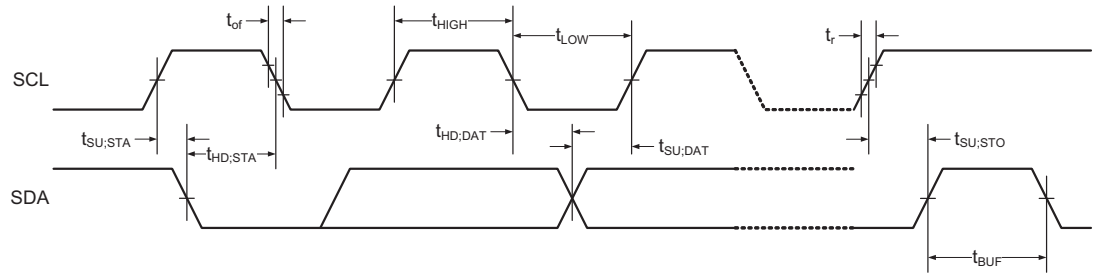
**Table 37-33. EBI SDRAM characteristics and requirements.**

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $t_{ClkPER2}$ | SDRAM clock period | | $0.5*t_{ClkPER}$ | | | ns |
| $t_{AH}$ | SDRAM address hold time | | | $0.5*t_{ClkPER2}$ | | |
| $t_{AS}$ | SDRAM address setup time | | | $0.5*t_{ClkPER2}$ | | |
| $t_{CH}$ | SDRAM clock high-level width | | | $0.5*t_{ClkPER2}$ | | |
| $t_{CL}$ | SDRAM clock low-level width | | | $0.5*t_{ClkPER2}$ | | |
| $t_{CKH}$ | SDRAM CKE hold time | | | $0.5*t_{ClkPER2}$ | | |
| $t_{CKS}$ | SDRAM CKE setup time | | | $0.5*t_{ClkPER2}$ | | |
| $t_{CMH}$ | SDRAM CS, RAS, CAS, WE, DQM hold time | | | $0.5*t_{ClkPER2}$ | | |
| $t_{CMS}$ | SDRAM CS, RAS, CAS, WE, DQM setup time | | | $0.5*t_{ClkPER2}$ | | |
| $t_{DRH}$ | SDRAM data in hold after CLK high | | 0 | | | |
| $t_{AC}$ | SDRAM access time from CLK | | | | $t_{ClkPER}$-5 | |
| $t_{DWH}$ | SDRAM data out hold after CLK high | | | $0.5*t_{ClkPER2}$ | | |
| $t_{DWS}$ | SDRAM data out setup before CLK high | | | $0.5*t_{ClkPER2}$ | | |

### 37.1.17 Two-Wire Interface Characteristics

Table 37-34 on page 96 describes the requirements for devices connected to the Two-Wire Interface Bus. The Atmel AVR XMEGA Two-Wire Interface meets or exceeds these requirements under the noted conditions. Timing symbols refer to Figure 37-7.

**Figure 37-7. Two-Wire Interface bus timing.**



### 37.2.6 ADC characteristics

**Table 37-42. Power supply, reference and input range.**

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $AV_{CC}$ | Analog supply voltage | | $V_{CC}$- 0.3 | | $V_{CC}$+ 0.3 | V |
| $V_{REF}$ | Reference voltage | | 1 | | $AV_{CC}$- 0.6 | |
| $R_{in}$ | Input resistance | Switched | | 5.0 | | kΩ |
| $C_{in}$ | Input capacitance | Switched | | 5.0 | | pF |
| $R_{AREF}$ | Reference input resistance | (leakage only) | | >10 | | MΩ |
| $C_{AREF}$ | Reference input capacitance | Static load | | 7 | | pF |
| $V_{IN}$ | Input range | | -0.1 | | $AV_{CC}$+0.1 | V |
| | Conversion range | Differential mode, Vinp - Vinn | $-V_{REF}$ | | $V_{REF}$ | |
| $V_{IN}$ | Conversion range | Single ended unsigned mode, Vinp | -ΔV | | $V_{REF}$-ΔV | |
| ΔV | Fixed offset voltage | | | 190 | | LSB |

**Table 37-43. Clock and timing.**

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $Clk_{ADC}$ | ADC Clock frequency | Maximum is 1/4 of peripheral clock frequency | 100 | | 2000 | kHz |
| | | Measuring internal signals | 100 | | 125 | |
| $f_{ADC}$ | Sample rate | Current limitation (CURRLIMIT) off | 100 | | 2000 | ksps |
| | | CURRLIMIT = LOW | 100 | | 1500 | |
| | | CURRLIMIT = MEDIUM | 100 | | 1000 | |
| | | CURRLIMIT = HIGH | 100 | | 500 | |
| | Sampling time | 1/2 $Clk_{ADC}$ cycle | 0.25 | | 5 | µs |
| | Conversion time (latency) | (RES+2)/2+(GAIN !=0)<br>RES (Resolution) = 8 or 12 | 5 | | 8 | $Clk_{ADC}$ cycles |
| | Start-up time | ADC clock cycles | | 12 | 24 | $Clk_{ADC}$ cycles |
| | ADC settling time | After changing reference or input mode | | 7 | 7 | $Clk_{ADC}$ cycles |
| | | After ADC flush | | 1 | 1 | |

| Symbol | Parameter | Condition | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| $t_{delay}$ | Propagation delay | $V_{CC}$ = 3.0V, T= 85°C | mode = HS | | 60 | 90 | ns |
| | | mode = HS | | | 60 | | |
| | | $V_{CC}$ = 3.0V, T= 85°C | mode = LP | | 130 | | |
| | Current source calibration range | Single mode | | 2 | | 8 | µs |
| | | Double mode | | 4 | | 16 | |
| | 64-Level Voltage Scaler | Integral non-linearity (INL) | | | 0.3 | 0.5 | lsb |

### 37.2.9 Bandgap and Internal 1.0V Reference Characteristics

**Table 37-50. Bandgap and Internal 1.0V reference characteristics.**

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| | Startup time | As reference for ADC or DAC | 1 $Clk_{PER}$ + 2.5µs | | | µs |
| | | As input voltage to ADC and AC | | 1.5 | | |
| | Bandgap voltage | | | 1.1 | | V |
| INT1V | Internal 1.00V reference for ADC and DAC | T= 85°C, after calibration | 0.99 | 1 | 1.01 | |
| | Variation over voltage and temperature | Relative to T= 85°C, $V_{CC}$ = 3.0V | | ±1.0 | | % |

### 37.2.10 Brownout Detection Characteristics

**Table 37-51. Brownout detection characteristics.**

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $V_{BOT}$ | BOD level 0 falling $V_{CC}$ | | 1.60 | 1.62 | 1.72 | V |
| | BOD level 1 falling $V_{CC}$ | | | 1.8 | | |
| | BOD level 2 falling $V_{CC}$ | | | 2.0 | | |
| | BOD level 3 falling $V_{CC}$ | | | 2.2 | | |
| | BOD level 4 falling $V_{CC}$ | | | 2.4 | | |
| | BOD level 5 falling $V_{CC}$ | | | 2.6 | | |
| | BOD level 6 falling $V_{CC}$ | | | 2.8 | | |
| | BOD level 7 falling $V_{CC}$ | | | 3.0 | | |
| $t_{BOD}$ | Detection time | Continuous mode | | 0.4 | | µs |
| | | Sampled mode | | 1000 | | |
| $V_{HYST}$ | Hysteresis | | | 1.4 | | % |

### 37.2.11 External Reset Characteristics

**Table 37-52. External reset characteristics.**

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $t_{EXT}$ | Minimum reset pulse width | | | 86 | 1000 | ns |
| $V_{RST}$ | Reset threshold voltage ($V_{IH}$) | $V_{CC}$ = 2.7 - 3.6V | | 0.60*$V_{CC}$ | | V |
| | | $V_{CC}$ = 1.6 - 2.7V | | 0.60*$V_{CC}$ | | |
| | Reset threshold voltage ($V_{IL}$) | $V_{CC}$ = 2.7 - 3.6V | | 0.40*$V_{CC}$ | | |
| | | $V_{CC}$ = 1.6 - 2.7V | | 0.40*$V_{CC}$ | | |

### 37.2.12 Power-on Reset Characteristics

**Table 37-53. Power-on reset characteristics.**

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $V_{POT-}$ (1) | POR threshold voltage falling $V_{CC}$ | $V_{CC}$ falls faster than 1V/ms | 0.4 | 1.0 | | V |
| | | $V_{CC}$ falls at 1V/ms or slower | 0.8 | 1.0 | | |
| $V_{POT+}$ | POR threshold voltage rising $V_{CC}$ | | | 1.3 | 1.59 | |

Note: 1. $V_{POT-}$ values are only valid when BOD is disabled. When BOD is enabled $V_{POT-}$ = $V_{POT+}$.

### 37.2.13 Flash and EEPROM Memory Characteristics

**Table 37-54. Endurance and data retention.**

| Symbol | Parameter | Condition | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| | Flash | Write/Erase cycles | 25°C | 10K | | | Cycle |
| | | | 85°C | 10K | | | |
| | | | 105°C | 2K | | | |
| | | Data retention | 25°C | 100 | | | Year |
| | | | 85°C | 25 | | | |
| | | | 105°C | 10 | | | |
| | EEPROM | Write/Erase cycles | 25°C | 100K | | | Cycle |
| | | | 85°C | 100K | | | |
| | | | 105°C | 30K | | | |
| | | Data retention | 25°C | 100 | | | Year |
| | | | 85°C | 25 | | | |
| | | | 105°C | 10 | | | |

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $t_{CL}$ | Clock Low Time | $V_{CC}$ = 1.6 - 1.8V | 4.5 | | | ns |
| | | $V_{CC}$ = 2.7 - 3.6V | 2.4 | | | |
| $t_{CR}$ | Rise Time (for maximum frequency) | $V_{CC}$ = 1.6 - 1.8V | | | 1.5 | ns |
| | | $V_{CC}$ = 2.7 - 3.6V | | | 1.0 | |
| $t_{CF}$ | Fall Time (for maximum frequency) | $V_{CC}$ = 1.6 - 1.8V | | | 1.5 | ns |
| | | $V_{CC}$ = 2.7 - 3.6V | | | 1.0 | |
| $\Delta t_{CK}$ | Change in period from one clock cycle to the next | | | | 10 | % |

Notes: 1. System Clock Prescalers must be set so that maximum CPU clock frequency for device is not exceeded.
2. The maximum frequency vs. supply voltage is linear between 1.6V and 2.7V, and the same applies for all other parameters with supply voltage conditions.

### 37.2.14.7 External 16MHz crystal oscillator and XOSC characteristics

**Table 37-63. External 16MHz crystal oscillator and XOSC characteristics.**

| Symbol | Parameter | Condition | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| | Cycle to cycle jitter | FRQRANGE=0 | | | <10 | | ns |
| | | FRQRANGE=1, 2, 3 | | | <1 | | |
| | Frequency error | FRQRANGE=0 | | | <0.5 | | % |
| | | FRQRANGE=1 | | | <0.05 | | |
| | | FRQRANGE=2 | | | <0.005 | | |
| | | FRQRANGE=3 | | | <0.005 | | |
| | Duty cycle | FRQRANGE=0 | | | 50 | | |
| | | FRQRANGE=1 | | | 50 | | |
| | | FRQRANGE=2 | | | 50 | | |
| | | FRQRANGE=3 | | | 50 | | |

**Figure 38-33.I/O pin input threshold voltage vs. $V_{CC}$.**

***$V_{IL}$ I/O pin read as "0".***

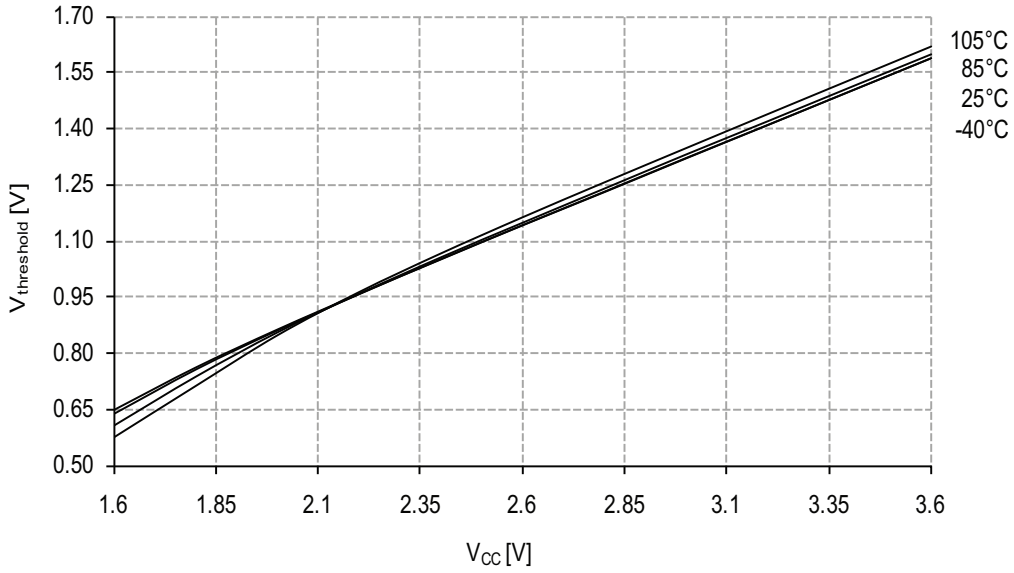
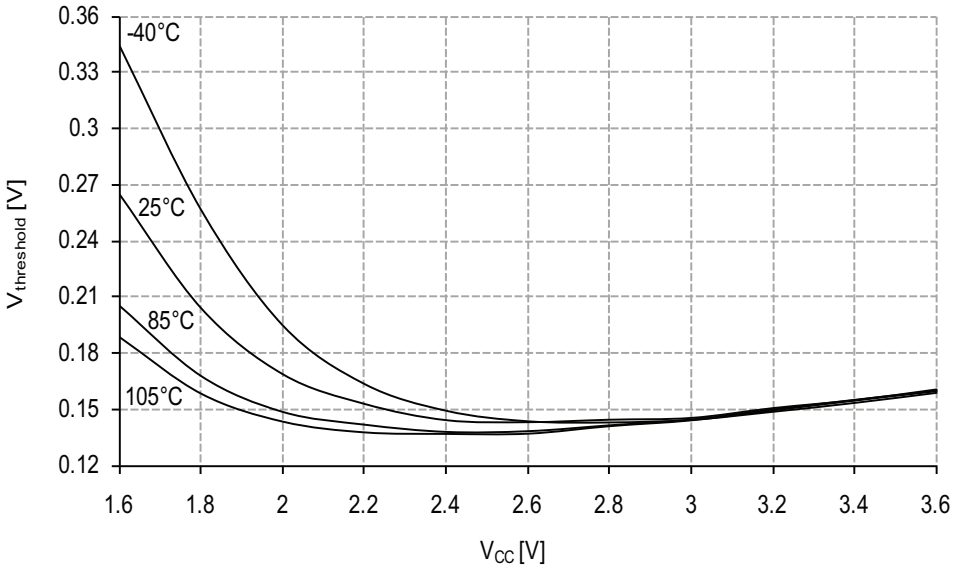



**Figure 38-34.I/O pin input hysteresis vs. $V_{CC}$.**



### 38.1.3 ADC Characteristics

**Figure 38-35.INL error vs. external $V_{REF}$.**

***T = 25°C, $V_{CC}$ = 3.6V, external reference.***

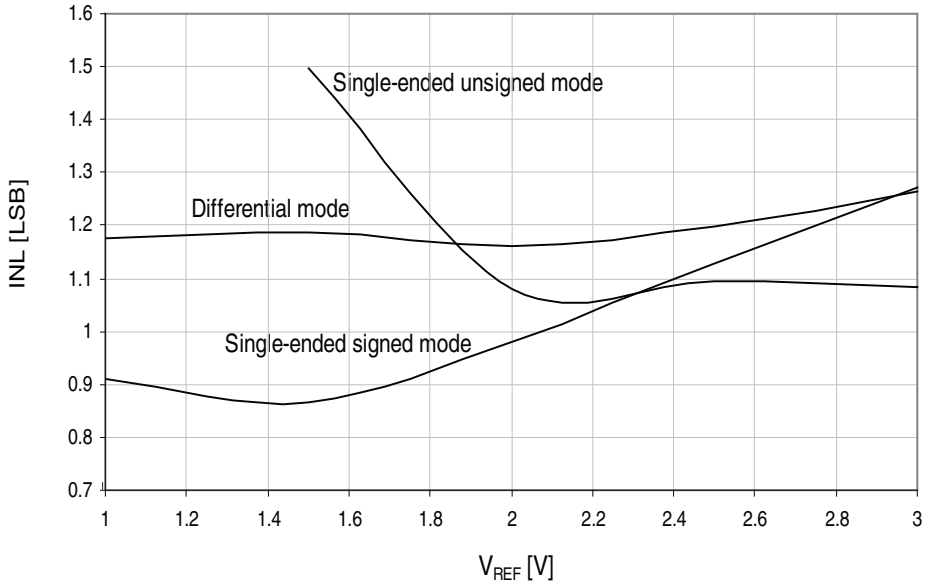



**Figure 38-36.INL error vs. sample rate.**

***T = 25°C, $V_{CC}$ = 3.6V, $V_{REF}$ = 2.0V external.***

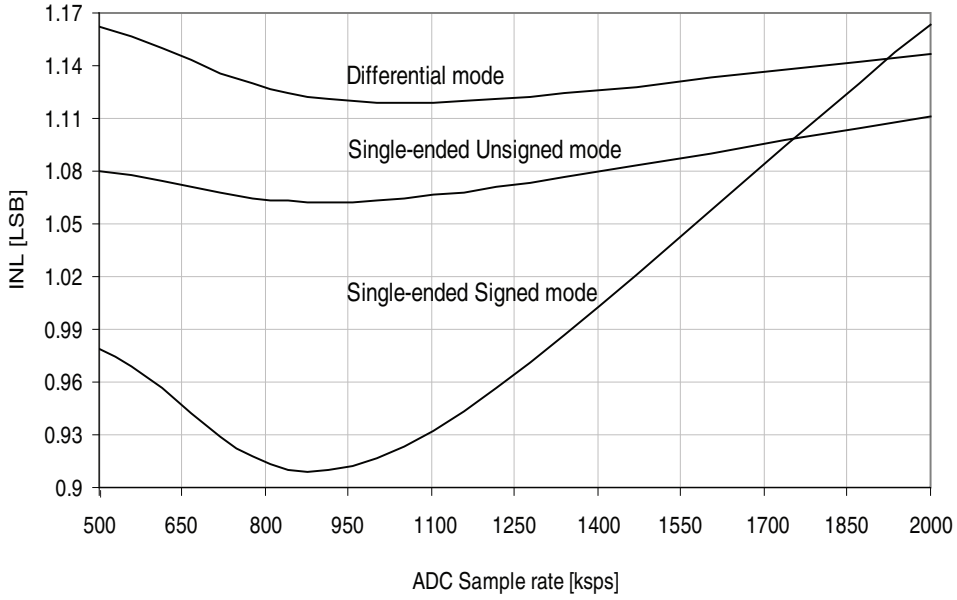


**Figure 38-53.Analog comparator hysteresis vs. $V_{CC}$.**

***High-speed mode, large hysteresis.***

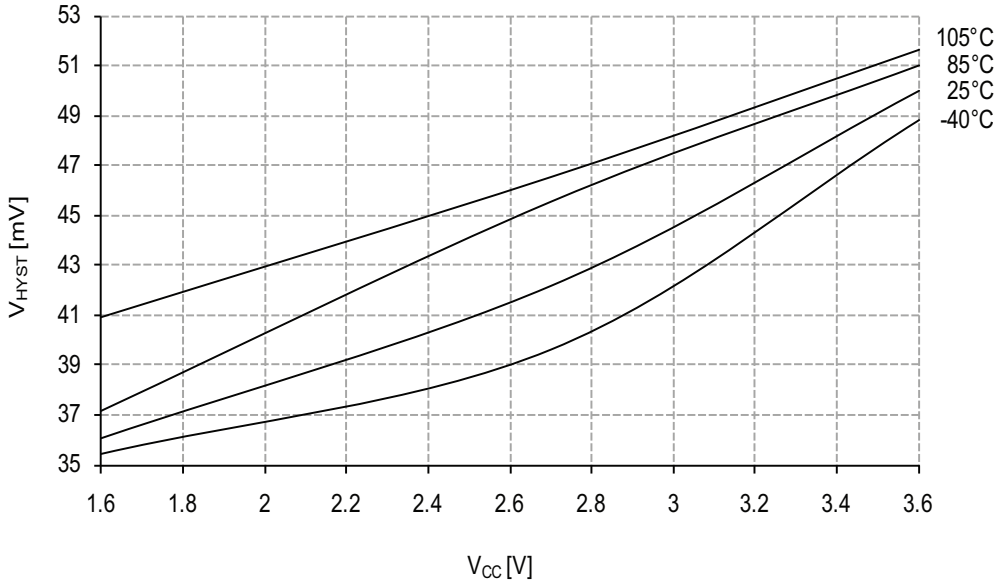



**Figure 38-54.Analog comparator hysteresis vs. $V_{CC}$.**

***Low power, large hysteresis.***

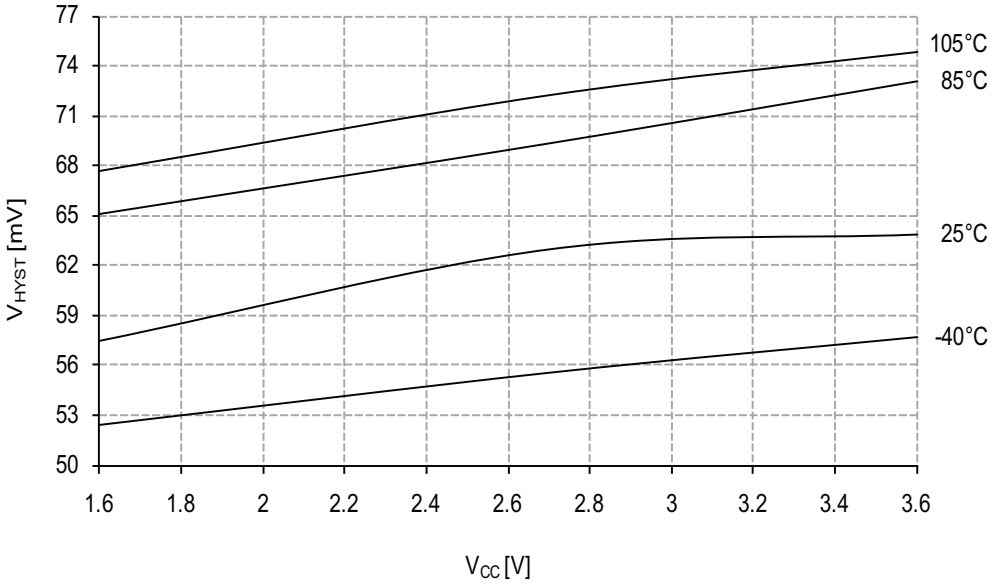
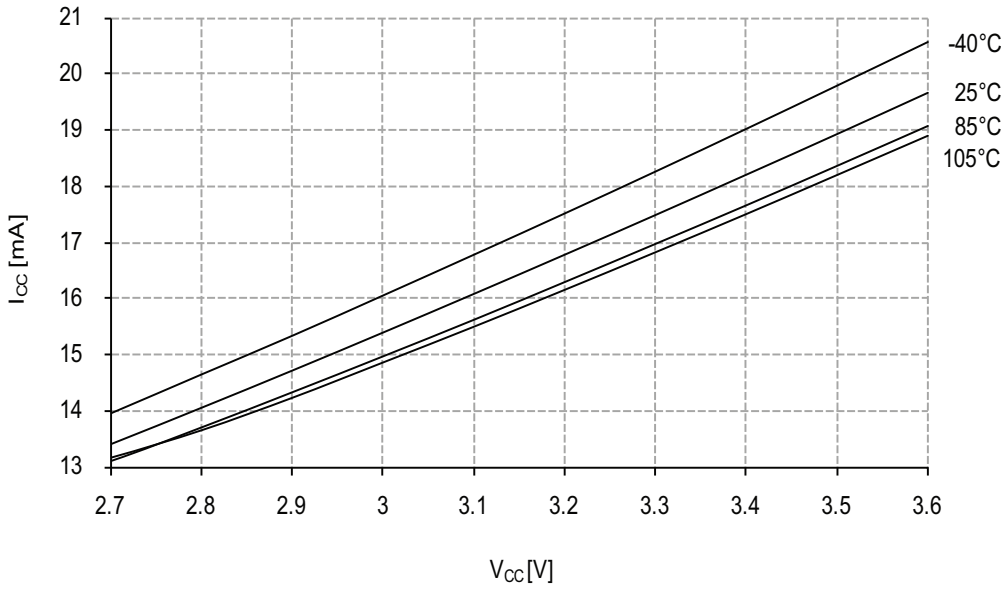


**Figure 38-89. Active mode supply current vs. $V_{CC}$.**

***$f_{SYS}$ = 32MHz internal oscillator.***




### 38.2.1.2 Idle mode supply current

**Figure 38-90.Idle mode supply current vs. frequency.**

***$f_{SYS}$ = 0 - 1MHz external clock, T = 25°C.***

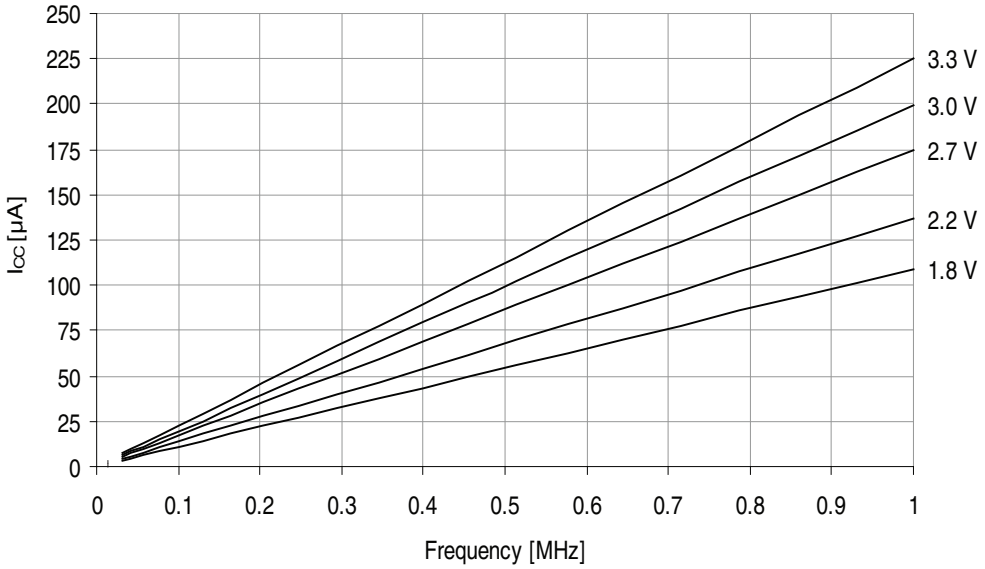


**Figure 38-103. I/O pin pull-up resistor current vs. pin voltage.**

***$V_{CC}$ = 3.0V.***

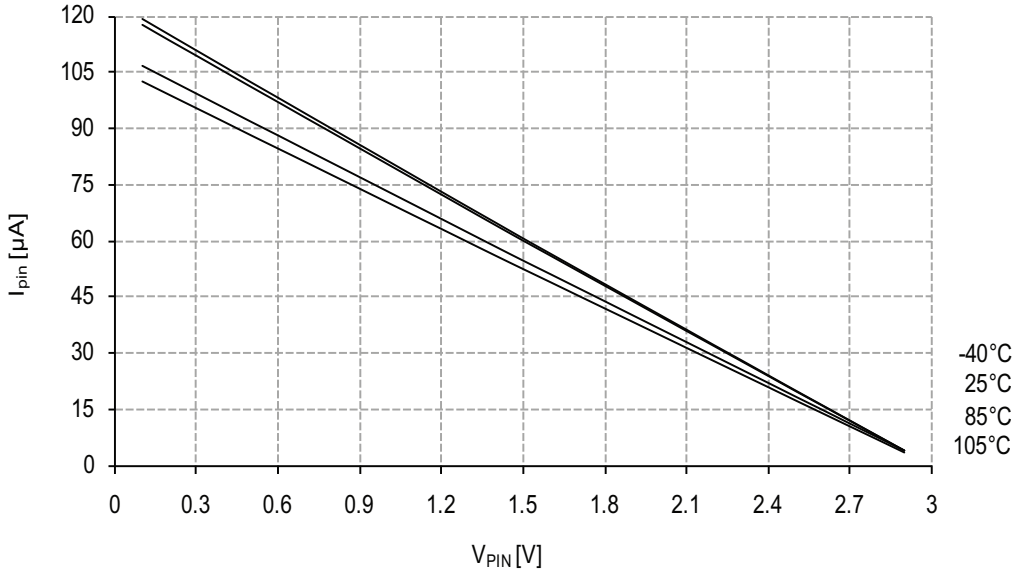
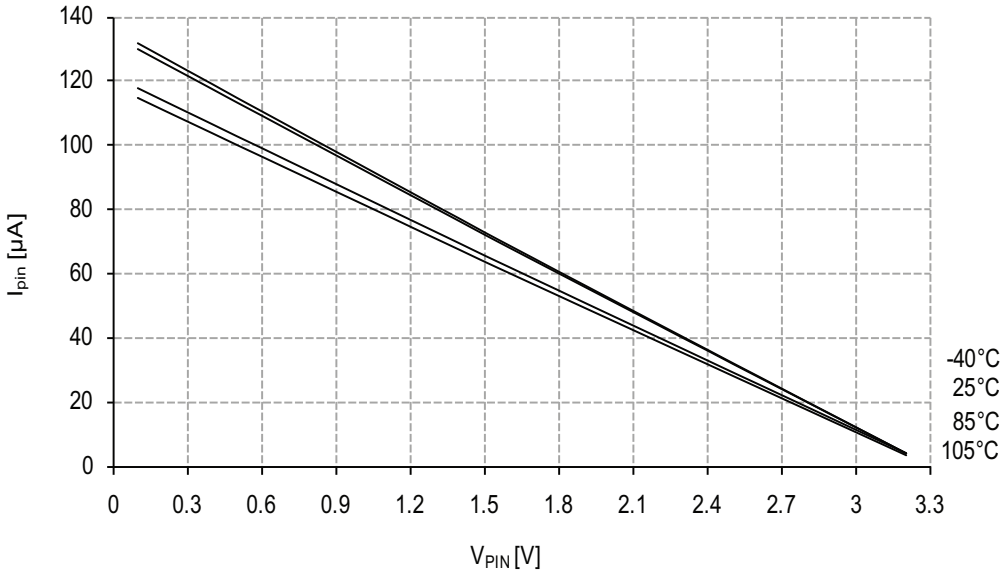



**Figure 38-104. I/O pin pull-up resistor current vs. pin voltage.**

***$V_{CC}$ = 3.3V.***



8. **USB, when receiving 1023Byte length isochronous frame, it will corrupt 1024th SRAM location**

When USB is configured for isochronous operation and 1023Byte data payload size, the 1024th RAM location that is directly after the endpoint RAM buffer will be corrupted.

**Problem fix/Workaround**

Allocate 1024bytes RAM buffer when using 1023 isochronous endpoint. This workaround is implemented in all USB software and source code delivered from Atmel in the AVR Software Framework.

9. **USB endpoint table is 16-byte alignment**

The USB endpoint table uses 16-byte alignment, instead of 16-bit alignment.

**Problem fix/Workaround**

Align the endpoint configuration table pointer in SRAM to a 16-byte. This workaround is implemented in all USB software and source code delivered from Atmel in the AVR Software Framework.

10. **USB Auto ZLP feature is non-functional**

The Auto ZLP feature is non-functional and can not be used.

**Problem fix/Workaround**

None.

11. **Disabling the USART transmitter does not automatically set the TxD pin direction to input**

If the USART transmitter is idle with no frames to transmit, setting TXEN to zero will not automatically set the TxD pin direction to input.

**Problem fix/Workaround**

The TxD pin direction can be set to input using the Port pin direction to input using the port direction (DIR) register. When the port pin direction is input, it will be immediate and ongoing transmissions will be truncated.

12. **TWI, SDAHOLD configuration in the TWI CTRL register is one bit**

The SDAHOLD configuration in the TWI Control register (CTRL) is one bit. Due to this the SDA hold time can be configured for maximum ~50ns when enabled. Configuring for longer hold time will have no effect.

**Problem fix/Workaround**

If longer SDA hold time than 50nS is required it must be handled in software.

13. **ADC has increased INL error in when used in SE unsigned mode at low temperatures**

When the ADC is used on single ended (SE) unsigned mode, -INL error is increased up to +/- 5 LSB in temperatures below -20C.

**Problem fix/Workaround**

Use the ADC in single ended signed mode.

14. **ADC is non-functional in SE unsigned mode with $V_{REF}$ below 1.8V**

When the ADC is used on single ended unsigned mode and $V_{REF}$ is below 1.8V, INL and DNL error is increased above +/- 10LSB, i.e. the ADC have missing codes under this condition.

## 39.2 ATxmega128A1U

### 39.2.1 Rev. L

- Register ANAINIT in MCUR will always read as zero
- Enabling DFLL with illegal reference oscillator will lock the DFLL
- XOSCPWR configuration is non-functional
- Configuration of PGM and CWCM is not as described in XMEGA AU Manual
- AWEX PWM output after fault restarted with wrong values
- RTC Counter value not correctly read after sleep
- RTC clock output option is non-functional
- USB, when receiving 1023 byte length isochronous frame, it will corrupt 1024th SRAM location
- USB endpoint table is 16-byte alignment
- USB Auto ZLP feature is non-functional
- Disabling the USART transmitter does not automatically set the TxD pin direction to input
- TWI, SDAHOLD configuration in the TWI CTRL register is one bit
- ADC has increased INL error in when used in SE unsigned mode at low temperatures
- ADC is non-functional in SE unsigned mode with VREF below 1.8V
- ADC has increased linearity error when using the gain stage above 500ksps
- DAC Offset calibration range too small when using AVCC as reference
- DAC clock noise
- Internal 1V reference has noise at low temperature

**1 Register ANAINIT in MCUR will always read as zero**

The ANAINIT register in the MCUR module will always be read as zero even if written to a value. The actual content of the register is correct.

**Problem fix/Workaround**

Do not use software that reads these registers to get the Analog Initialization configuration.

**2. Enabling DFLL with illegal reference oscillator will lock the clock system**

If external crystal is selected as reference for DFLL, but no crystal is connected and DFLL is enabled, the DFLL will be locked until reset is issued.

**Problem fix/Workaround**

Do not enable DFLL before reference clock is present, enabled and ready.

**3. XOSCPWR configuration is non-functional**

The Crystal oscillator drive (XOSCPWR) option in the XOSC Control register is non-functional.

**Problem fix/Workaround**

None.

**4. Configuration of PGM and CWCM is not as described in XMEGA AU Manual**

Configuration of common waveform channel mode (CWCM) and pattern generation mode (PGM), is not as described in the XMEGA AU manual.