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, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	34
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.6V ~ 3.6V
Data Converters	A/D 12x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	0°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atxmega16a4u-anr

9. Event System

9.1 Features

- System for direct peripheral-to-peripheral communication and signaling
- Peripherals can directly send, receive, and react to peripheral events
  - CPU and DMA controller independent operation
  - 100% predictable signal timing
  - Short and guaranteed response time
- Eight event channels for up to eight different and parallel signal routing configurations
- Events can be sent and/or used by most peripherals, clock system, and software
- Additional functions include
  - Quadrature decoders
  - Digital filtering of I/O pin state
- Works in active mode and idle sleep mode

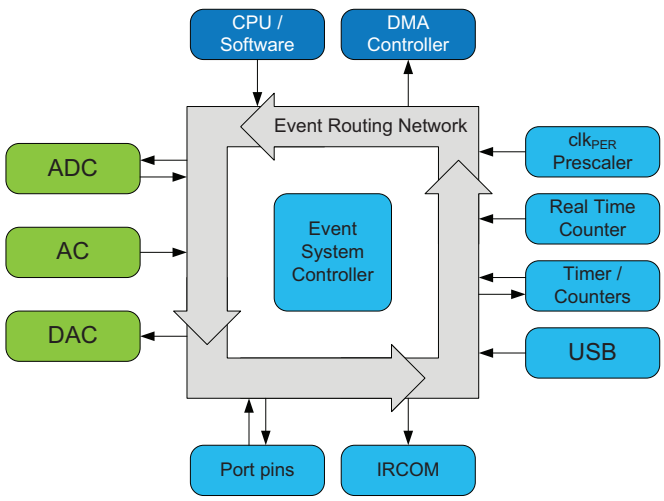
9.2 Overview

The event system enables direct peripheral-to-peripheral communication and signaling. It allows a change in one peripheral's state to automatically trigger actions in other peripherals. It is designed to provide a predictable system for short and predictable response times between peripherals. It allows for autonomous peripheral control and interaction without the use of interrupts, CPU, or DMA controller resources, and is thus a powerful tool for reducing the complexity, size and execution time of application code. It also allows for synchronized timing of actions in several peripheral modules.

A change in a peripheral's state is referred to as an event, and usually corresponds to the peripheral's interrupt conditions. Events can be directly passed to other peripherals using a dedicated routing network called the event routing network. How events are routed and used by the peripherals is configured in software.

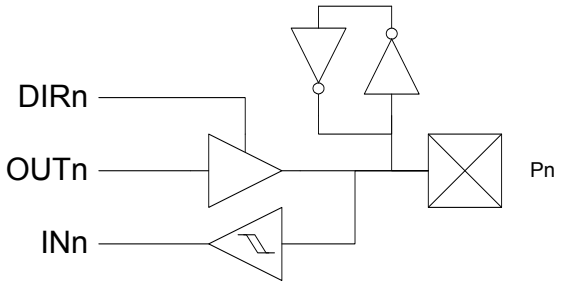
Figure 9-1 on page 20 shows a basic diagram of all connected peripherals. The event system can directly connect together analog and digital converters, analog comparators, I/O port pins, the real-time counter, timer/counters, IR communication module (IRCOM), and USB interface. It can also be used to trigger DMA transactions (DMA controller). Events can also be generated from software and the peripheral clock.

Figure 9-1. Event system overview and connected peripherals.



The event routing network consists of eight software-configurable multiplexers that control how events are routed and used. These are called event channels, and allow for up to eight parallel event routing configurations. The maximum routing latency is two peripheral clock cycles. The event system works in both active mode and idle sleep mode.

Figure 15-4. I/O configuration - Totem-pole with bus-keeper.



15.3.5 Others

Figure 15-5. Output configuration - Wired-OR with optional pull-down.

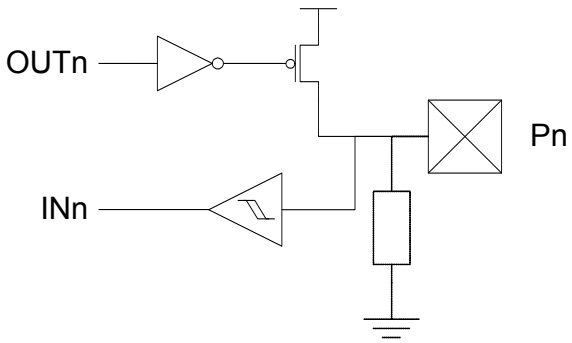
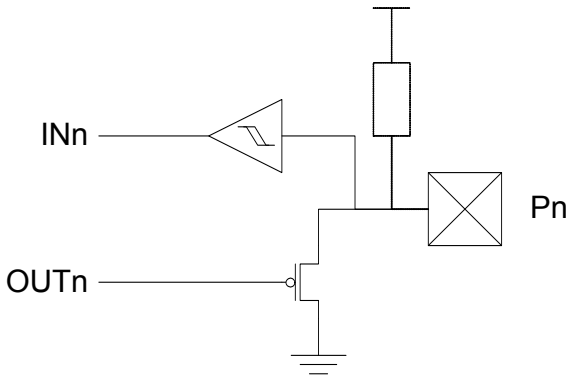


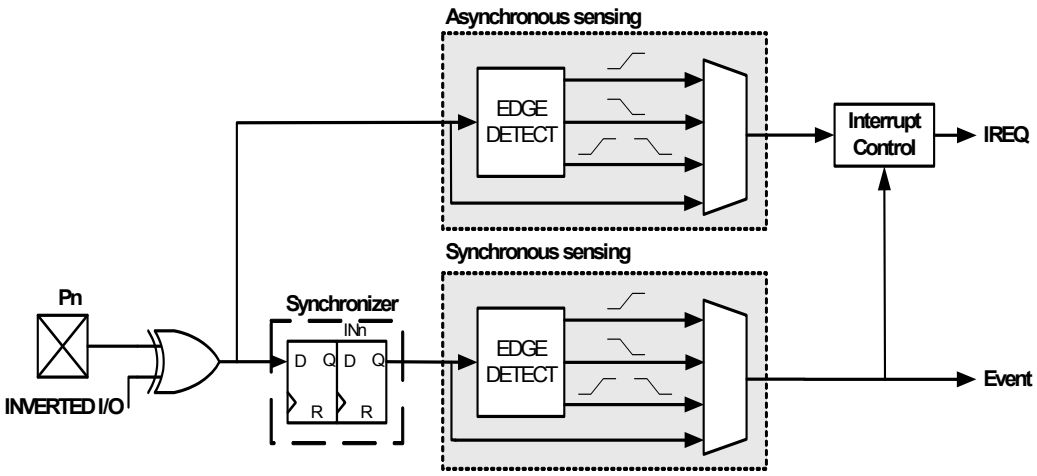
Figure 15-6. I/O configuration - Wired-AND with optional pull-up.



15.4 Input sensing

Input sensing is synchronous or asynchronous depending on the enabled clock for the ports, and the configuration is shown in Figure 15-7.

Figure 15-7. Input sensing system overview.



When a pin is configured with inverted I/O, the pin value is inverted before the input sensing.

15.5 Alternate Port Functions

Most port pins have alternate pin functions in addition to being a general purpose I/O pin. When an alternate function is enabled, it might override the normal port pin function or pin value. This happens when other peripherals that require pins are enabled or configured to use pins. If and how a peripheral will override and use pins is described in the section for that peripheral. “Pinout and Pin Functions” on page 55 shows which modules on peripherals that enable alternate functions on a pin, and which alternate functions that are available on a pin.

19. Hi-Res – High Resolution Extension

19.1 Features

- Increases waveform generator resolution up to 8x (three bits)
- Supports frequency, single-slope PWM, and dual-slope PWM generation
- Supports the AWeX when this is used for the same timer/counter

19.2 Overview

The high-resolution (hi-res) extension can be used to increase the resolution of the waveform generation output from a timer/counter by four or eight. It can be used for a timer/counter doing frequency, single-slope PWM, or dual-slope PWM generation. It can also be used with the AWeX if this is used for the same timer/counter.

The hi-res extension uses the peripheral 4x clock (Clk_{PER4}). The system clock prescalers must be configured so the peripheral 4x clock frequency is four times higher than the peripheral and CPU clock frequency when the hi-res extension is enabled.

There are 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.

22. TWI – Two-Wire Interface

22.1 Features

- Two 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
- 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 frequency is 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.

24. USART

24.1 Features

- Five identical USART peripherals
- Full-duplex operation
- Asynchronous or synchronous operation
  - Synchronous clock rates up to 1/2 of the device clock frequency
  - Asynchronous clock rates up to 1/8 of the device clock frequency
- Supports serial frames with 5, 6, 7, 8, or 9 data bits and 1 or 2 stop bits
- Fractional baud rate generator
  - Can generate desired baud rate from any system clock frequency
  - No need for external oscillator with certain frequencies
- Built-in error detection and correction schemes
  - Odd or even parity generation and parity check
  - Data overrun and framing error detection
  - Noise filtering includes false start bit detection and digital low-pass filter
- Separate interrupts for
  - Transmit complete
  - Transmit data register empty
  - Receive complete
- Multiprocessor communication mode
  - Addressing scheme to address a specific devices on a multidevice bus
  - Enable unaddressed devices to automatically ignore all frames
- Master SPI mode
  - Double buffered operation
  - Operation up to 1/2 of the peripheral clock frequency
- IRCOM module for IrDA compliant pulse modulation/demodulation

24.2 Overview

The universal synchronous and asynchronous serial receiver and transmitter (USART) is a fast and flexible serial communication module. The USART supports full-duplex communication and asynchronous and synchronous operation. The USART can be configured to operate in SPI master mode and used for SPI communication.

Communication is frame based, and the frame format can be customized to support a wide range of standards. The USART is buffered in both directions, enabling continued data transmission without any delay between frames. Separate interrupts for receive and transmit complete enable fully interrupt driven communication. Frame error and buffer overflow are detected in hardware and indicated with separate status flags. Even or odd parity generation and parity check can also be enabled.

The clock generator includes a fractional baud rate generator that is able to generate a wide range of USART baud rates from any system clock frequencies. This removes the need to use an external crystal oscillator with a specific frequency to achieve a required baud rate. It also supports external clock input in synchronous slave operation.

When the USART is set in master SPI mode, all USART-specific logic is disabled, leaving the transmit and receive buffers, shift registers, and baud rate generator enabled. Pin control and interrupt generation are identical in both modes. The registers are used in both modes, but their functionality differs for some control settings.

An IRCOM module can be enabled for one USART to support IrDA 1.4 physical compliant pulse modulation and demodulation for baud rates up to 115.2Kbps.

PORTC and PORTD each has two USARTs. PORTE has one USART. Notation of these peripherals are USARTC0, USARTC1, USARTD0, USARTD1 and USARTE0, respectively.

36.1.14 Clock and Oscillator Characteristics

36.1.14.1 Calibrated 32.768kHz Internal Oscillator characteristics

Table 36-22. 32.768kHz internal oscillator characteristics.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
	Frequency			32.768		kHz
	Factory calibration accuracy	T = 85°C, V_{CC} = 3.0V	-0.5		0.5	%
	User calibration accuracy		-0.5		0.5	%

36.1.14.2 Calibrated 2MHz RC Internal Oscillator characteristics

Table 36-23. 2MHz internal oscillator characteristics.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
	Frequency range	DFLL can tune to this frequency over voltage and temperature	1.8		2.2	MHz
	Factory calibrated frequency			2.0		MHz
	Factory calibration accuracy	T = 85°C, V_{CC}= 3.0V	-1.5		1.5	%
	User calibration accuracy		-0.2		0.2	%
	DFLL calibration stepsize			0.21		%

36.1.14.3 Calibrated and tunable 32MHz internal oscillator characteristics

Table 36-24. 32MHz internal oscillator characteristics.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
	Frequency range	DFLL can tune to this frequency over voltage and temperature	30		55	MHz
	Factory calibrated frequency			32		MHz
	Factory calibration accuracy	T = 85°C, V_{CC}= 3.0V	-1.5		1.5	%
	User calibration accuracy		-0.2		0.2	%
	DFLL calibration step size			0.22		%

36.1.14.4 32kHz Internal ULP Oscillator characteristics

Table 36-25. 32kHz internal ULP oscillator characteristics.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
	Output frequency			32		kHz
	Accuracy		-30		30	%

Table 36-63. SPI timing characteristics and requirements.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
t_{SCK}	SCK period	Master		(See Table 21-4 in XMEGA AU Manual)		ns
t_{SCKW}	SCK high/low width	Master		0.5×SCK		
t_{SCKR}	SCK rise time	Master		2.7		
t_{SCKF}	SCK fall time	Master		2.7		
t_{MIS}	MISO setup to SCK	Master		10		
t_{MIH}	MISO hold after SCK	Master		10		
t_{MOS}	MOSI setup SCK	Master		0.5×SCK		
t_{MOH}	MOSI hold after SCK	Master		1.0		
t_{SSCK}	Slave SCK Period	Slave	4×t Clk_{PER}			
t_{SSCKW}	SCK high/low width	Slave	2×t Clk_{PER}			
t_{SSCKR}	SCK rise time	Slave			1600	
t_{SSCKF}	SCK fall time	Slave			1600	
t_{SIS}	MOSI setup to SCK	Slave	3.0			
t_{SIH}	MOSI hold after SCK	Slave	t Clk_{PER}			
t_{SSS}	$\overline{SS}$ setup to SCK	Slave	21			
t_{SSH}	$\overline{SS}$ hold after SCK	Slave	20			
t_{SOS}	MISO setup SCK	Slave		8.0		
t_{SOH}	MISO hold after SCK	Slave		13		
t_{SOSS}	MISO setup after $\overline{SS}$ low	Slave		11		
t_{SOSH}	MISO hold after $\overline{SS}$ high	Slave		8.0		

36.3.10 Brownout Detection Characteristics

Table 36-81. Brownout detection characteristics.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
V_{BOT}	BOD level 0 falling V_{CC}		1.50	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			1.2		%

36.3.11 External Reset Characteristics

Table 36-82. External reset characteristics.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
t_{EXT}	Minimum reset pulse width		1000	95		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		0.50×V_{CC}		
		V_{CC} = 1.6 - 2.7V		0.40×V_{CC}		
R_{RST}	Reset pin Pull-up Resistor			25		kΩ

36.3.12 Power-on Reset Characteristics

Table 36-83. 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+}.

Figure 37-3. Active mode supply current vs. V_{CC}.

f_{SYS} = 32.768kHz internal oscillator.

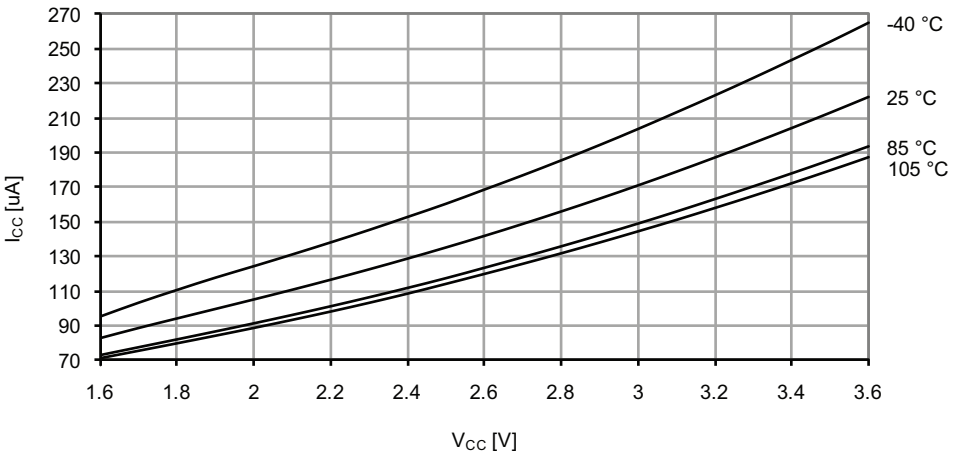
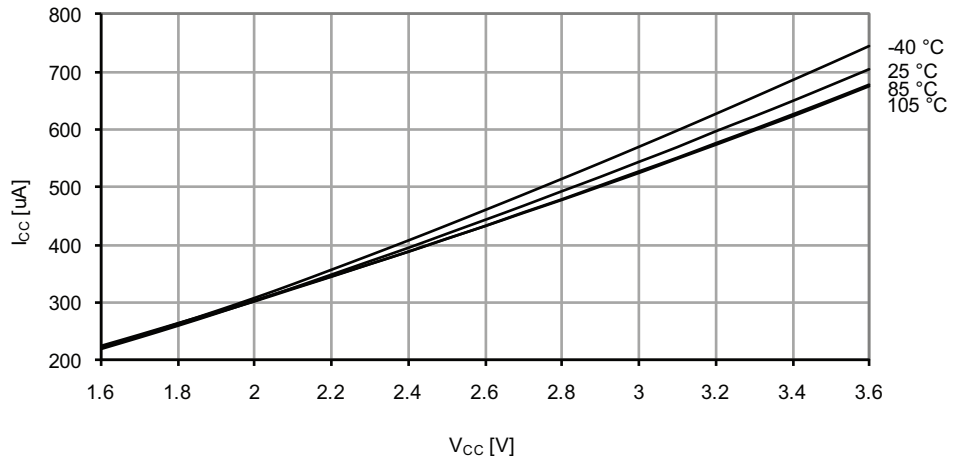


Figure 37-4. Active mode supply current vs. V_{CC}.

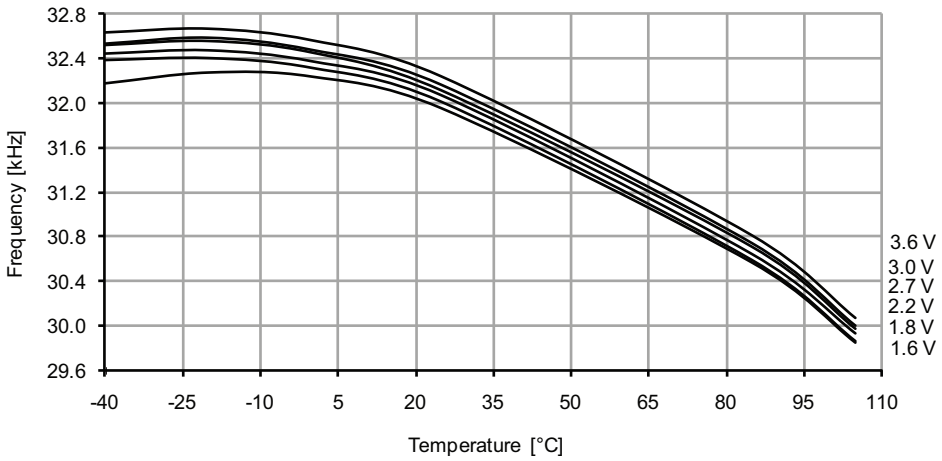
f_{SYS} = 1MHz external clock.



37.1.10 Oscillator Characteristics

37.1.10.1 Ultra Low-Power internal oscillator

Figure 37-70. Ultra Low-Power internal oscillator frequency vs. temperature.



37.1.10.2 32.768kHz Internal Oscillator

Figure 37-71. 32.768kHz internal oscillator frequency vs. temperature.

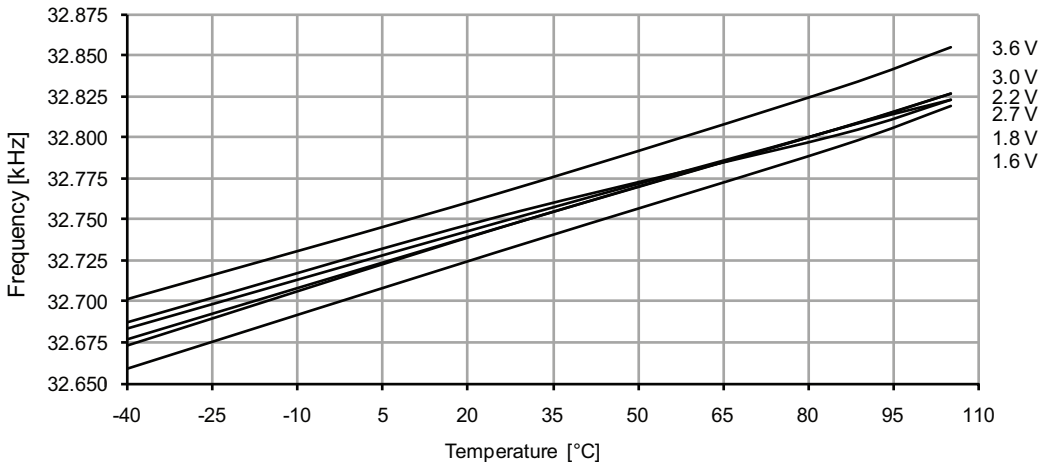


Figure 37-97. Idle mode supply current vs. V_{CC}.

f_{SYS} = 32MHz internal oscillator prescaled to 8MHz.

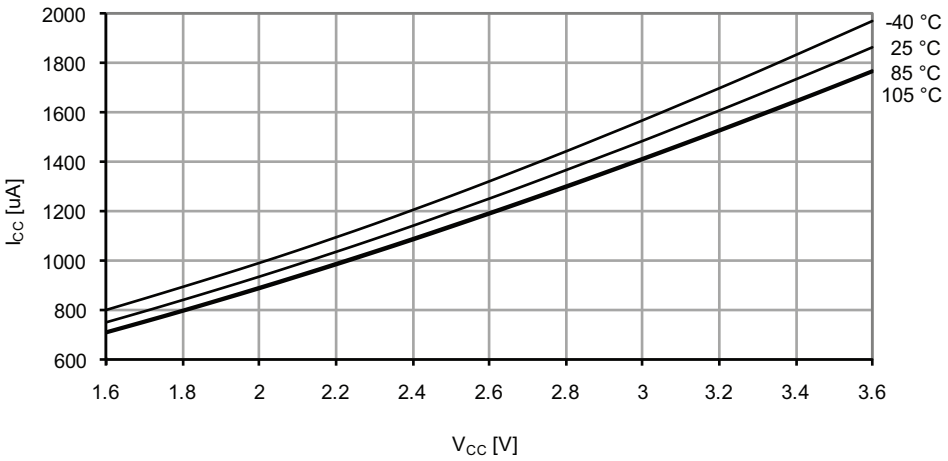


Figure 37-98. Idle mode current vs. V_{CC}.

f_{SYS} = 32MHz internal oscillator.

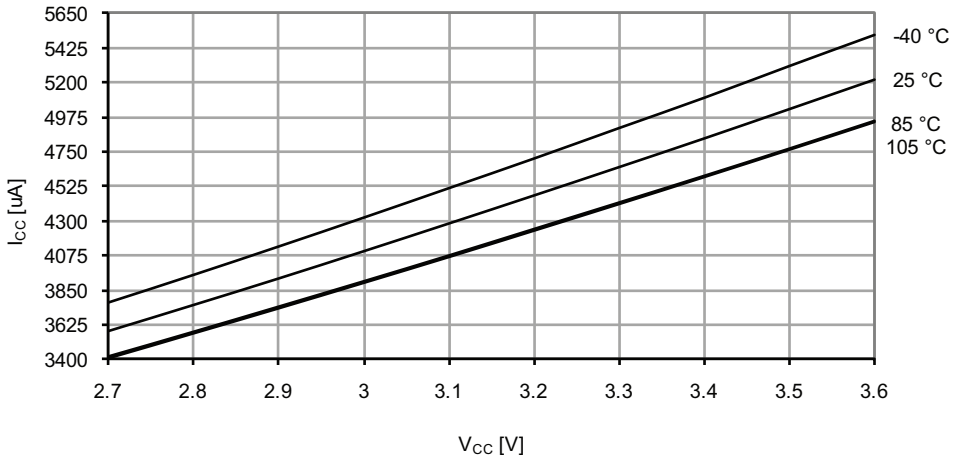


Figure 37-130. Offset error vs. V_{CC}.

T = 25°C, V_{REF} = external 1.0V, ADC sampling speed = 500ksps.

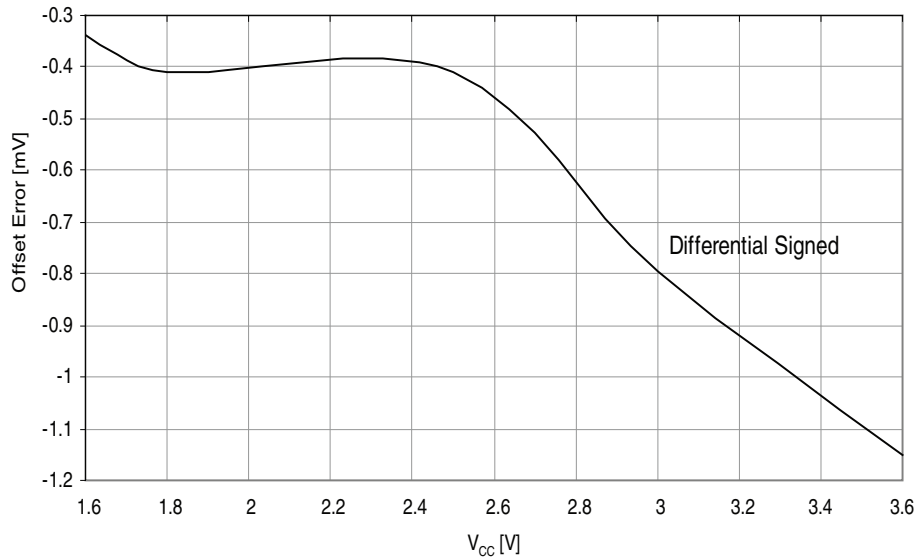


Figure 37-131. Noise vs. V_{REF}.

T = 25°C, V_{CC} = 3.6V, ADC sampling speed = 500ksps.

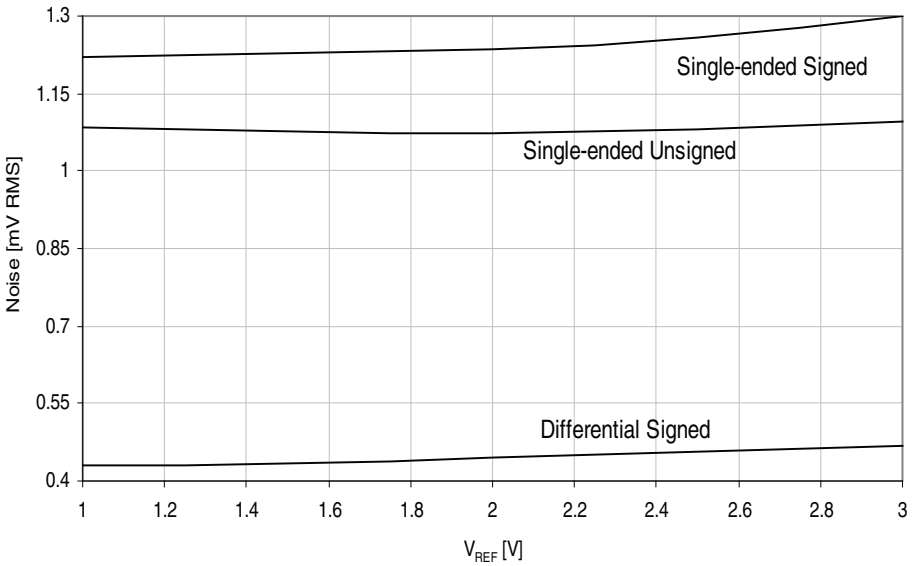


Figure 37-225. Analog comparator current source vs. calibration value.

V_{CC} = 3.0V.

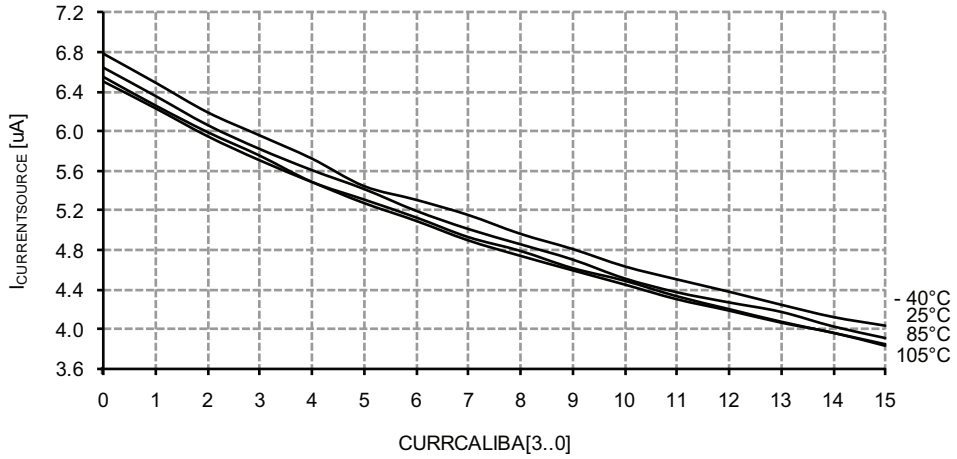
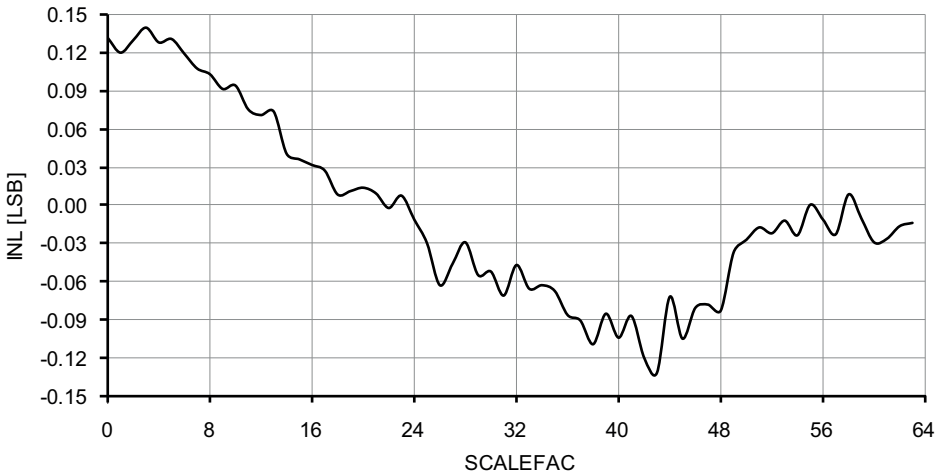


Figure 37-226. Voltage scaler INL vs. SCALEFAC.

T = 25°C, V_{CC} = 3.0V.



37.3.10.3 2MHz Internal Oscillator

Figure 37-241. 2MHz internal oscillator frequency vs. temperature.

DFLL disabled.

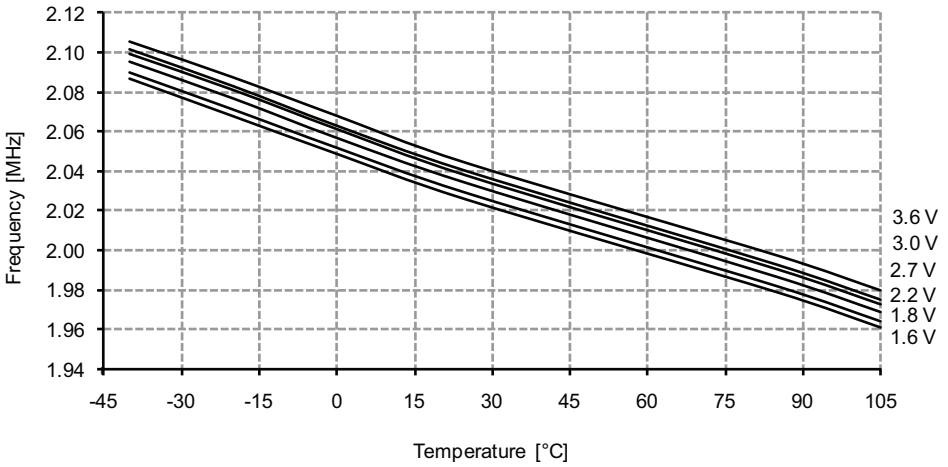
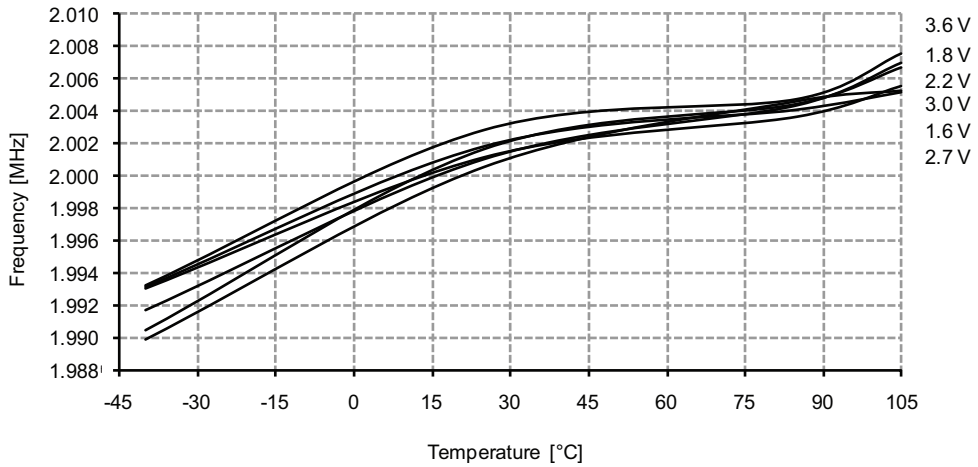


Figure 37-242. 2MHz internal oscillator frequency vs. temperature.

DFLL enabled, from the 32.768kHz internal oscillator .



37.4 ATxmega128A4U

37.4.1 Current consumption

37.4.1.1 Active mode supply current

Figure 37-253. Active supply current vs. frequency.

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

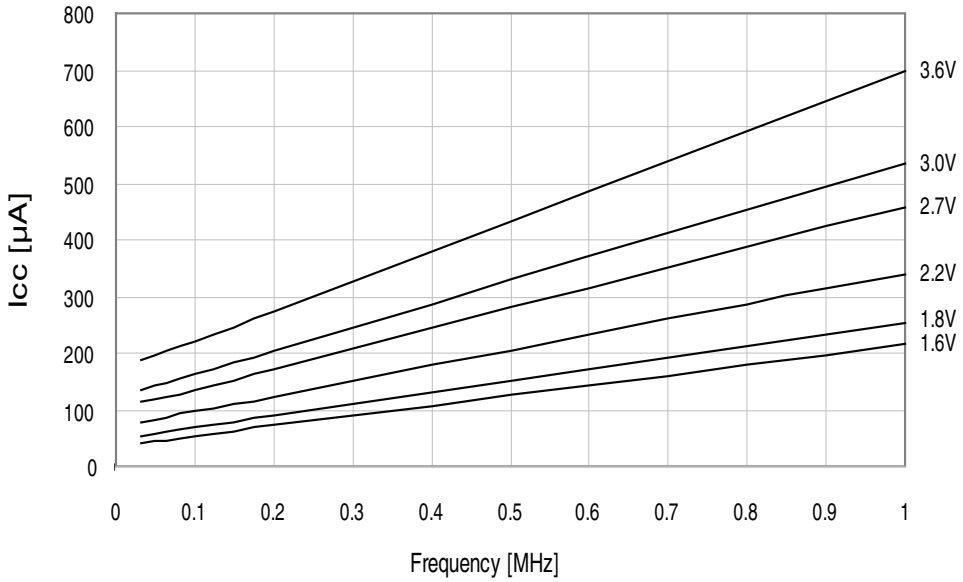


Figure 37-254. Active supply current vs. frequency.

f_{SYS} = 1 - 32MHz external clock, T = 25°C.

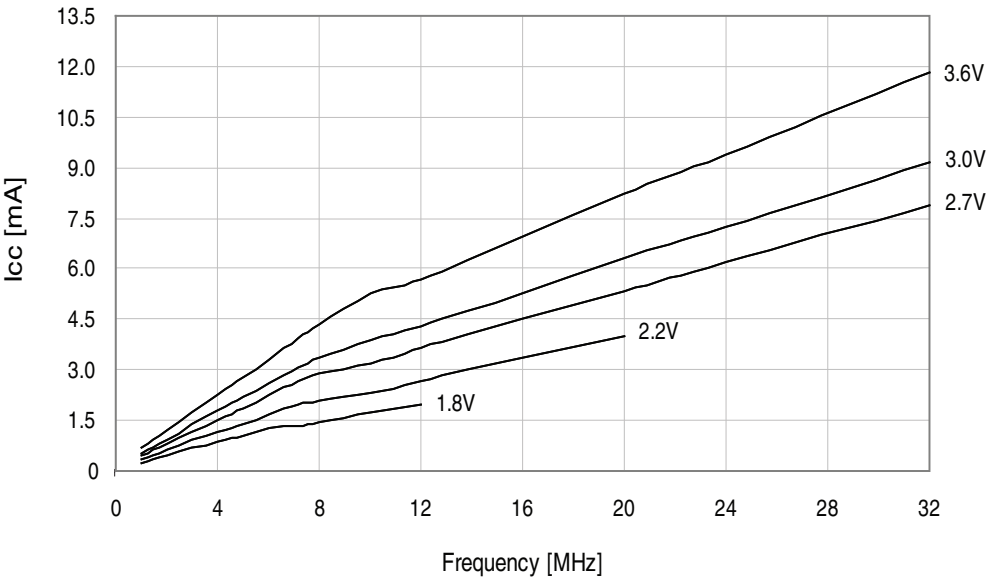


Figure 37-293. DNL error vs. input code

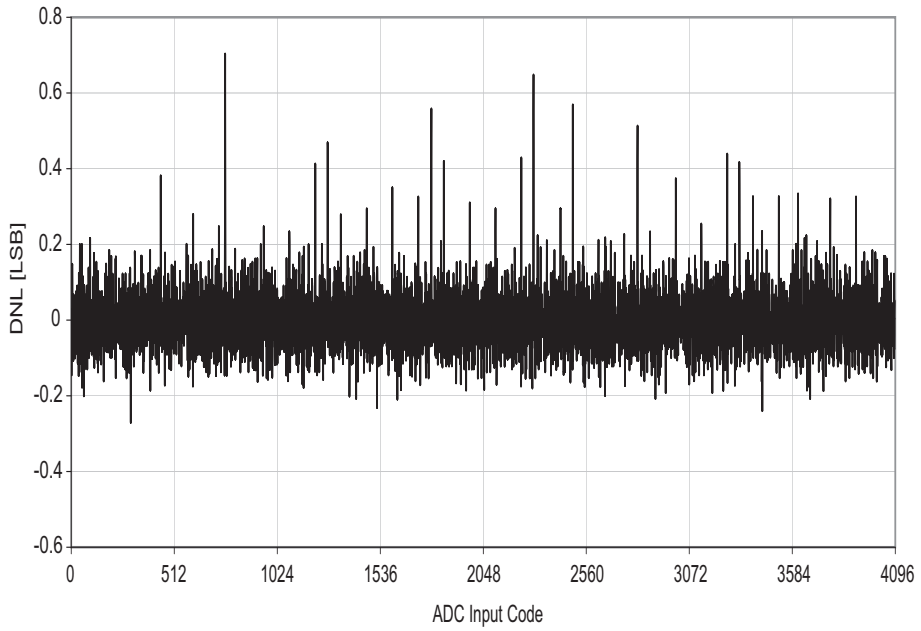


Figure 37-294. Gain error vs. V_{REF}

T = 25°C, V_{CC} = 3.6V, ADC sampling speed = 500ksps

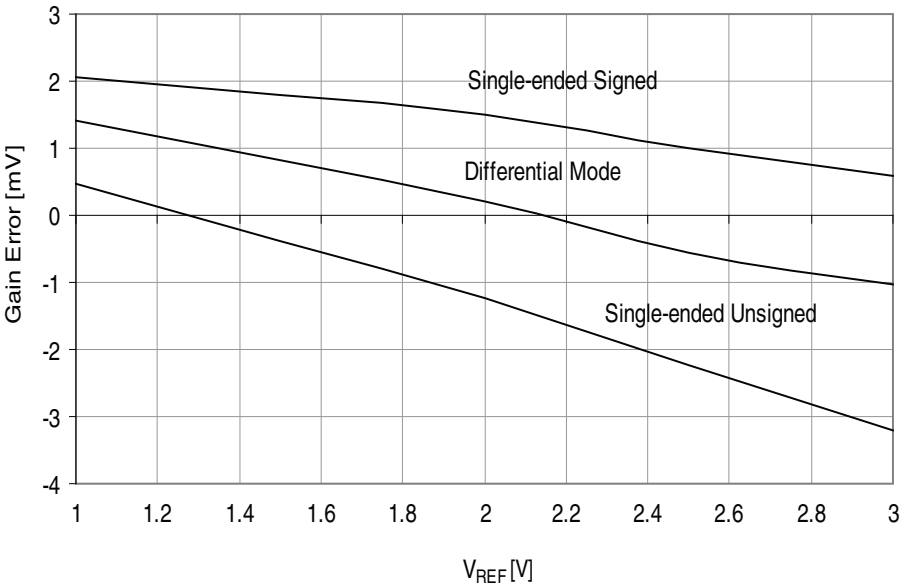


Figure 37-305. Analog comparator hysteresis vs. V_{CC}.

Low power, small hysteresis

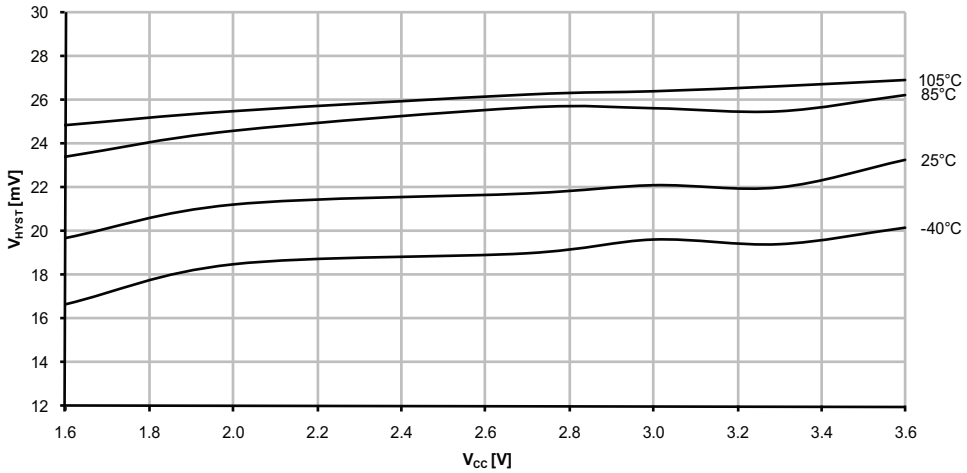


Figure 37-306. Analog comparator hysteresis vs. V_{CC}.

High-speed mode, large hysteresis

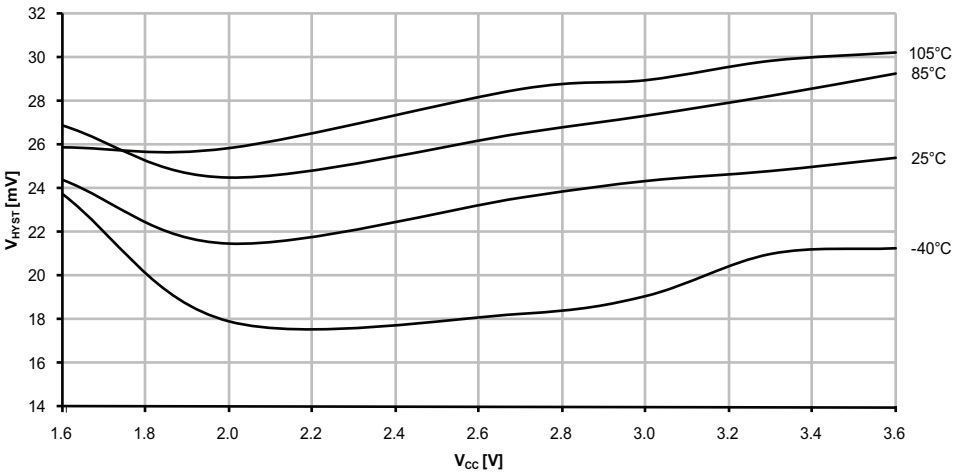
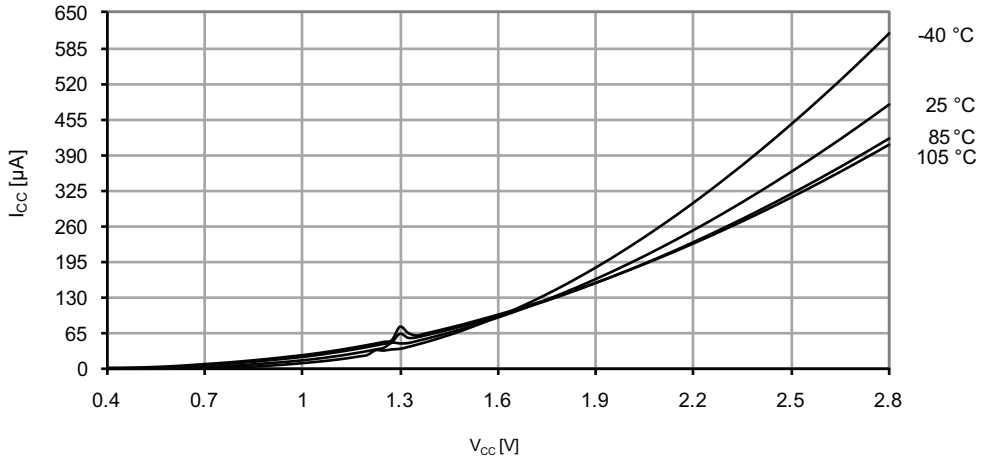


Figure 37-321. Power-on reset current consumption vs. V_{CC}

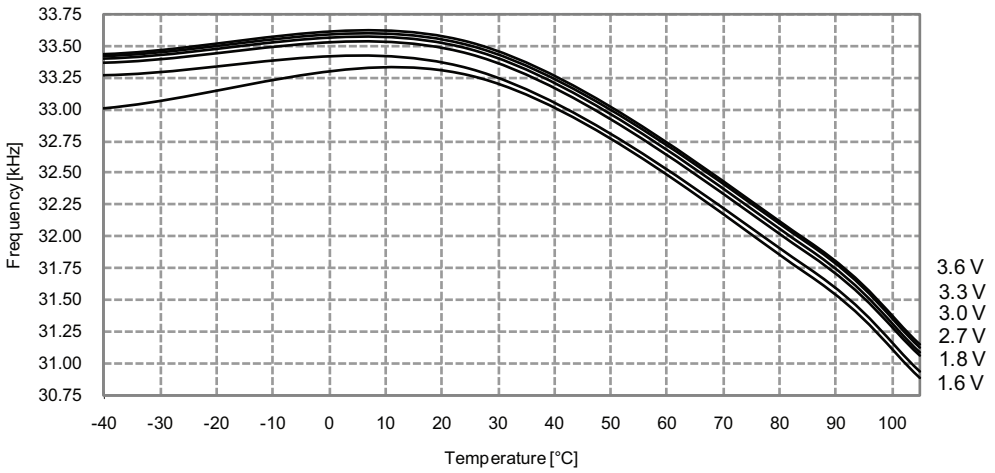
BOD level = 3.0V, enabled in sampled mode



37.4.10 Oscillator Characteristics

37.4.10.1 Ultra Low-Power internal oscillator

Figure 37-322. Ultra Low-Power internal oscillator frequency vs. temperature.



38.2 ATxmega32A4U

38.2.1 Rev. E

- ADC may have missing codes in SE unsigned mode at low temp and low Vcc
- CRC fails for Range CRC when end address is the last word address of a flash section
- AWeX fault protection restore is not done correct in Pattern Generation Mode

1. **ADC may have missing codes in SE unsigned mode at low temp and low Vcc**
   The ADC may have missing codes i single ended (SE) unsigned mode below 0C when Vcc is below 1.8V.

   Problem fix/Workaround
   Use the ADC in SE signed mode.

2. **CRC fails for Range CRC when end address is the last word address of a flash section**
   If boot read lock is enabled, the range CRC cannot end on the last address of the application section. If application table read lock is enabled, the range CRC cannot end on the last address before the application table.

   Problem fix/Workaround
   Ensure that the end address used in Range CRC does not end at the last address before a section with read lock enabled. Instead, use the dedicated CRC commands for complete applications sections.

3. **AWeX fault protection restore is not done correct in Pattern Generation Mode**
   When a fault is detected the OUTOVEN register is cleared, and when fault condition is cleared, OUTOVEN is restored according to the corresponding enabled DTI channels. For Common Waveform Channel Mode (CWCM), this has no effect as the OUTOVEN is correct after restoring from fault. For Pattern Generation Mode (PGM), OUTOVEN should instead have been restored according to the DTLSBUF register.

   Problem fix/Workaround
   For CWCM no workaround is required.

   For PGM in latched mode, disable the DTI channels before returning from the fault condition. Then, set correct OUTOVEN value and enable the DTI channels, before the direction (DIR) register is written to enable the correct outputs again.

38.2.2 Rev. A - D

Not sampled.