



Welcome to [E-XFL.COM](https://www.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	I ² C, IrDA, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	50
Program Memory Size	256KB (128K x 16)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	1.6V ~ 3.6V
Data Converters	A/D 16x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atxmega256a3u-aur

3. For packaging information, see ["Packaging information" on page 71](#).
4. Tape and Reel.

Package Type	
64A	64-lead, 14 x 14mm body size, 1.0mm body thickness, 0.8mm lead pitch, thin profile plastic quad flat package (TQFP)
64M2	64-pad, 9 x 9 x 1.0mm body, lead pitch 0.50mm, 7.65mm exposed pad, quad flat no-lead package (QFN)

Typical Applications

Industrial control	Climate control	Low power battery applications
Factory automation	RF and ZigBee®	Power tools
Building control	USB connectivity	HVAC
Board control	Sensor control	Utility metering
White goods	Optical	Medical applications

4. Resources

A comprehensive set of development tools, application notes and datasheets are available for download on <http://www.atmel.com/avr>.

4.1 Recommended reading

- Atmel AVR XMEGA AU manual
- XMEGA application notes

This device data sheet only contains part specific information with a short description of each peripheral and module. The XMEGA AU manual describes the modules and peripherals in depth. The XMEGA application notes contain example code and show applied use of the modules and peripherals.

All documentations are available from www.atmel.com/avr.

5. Capacitive touch sensing

The Atmel QTouch library provides a simple to use solution to realize touch sensitive interfaces on most Atmel AVR microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully debounced reporting of touch keys and includes Adjacent Key Suppression® (AKS®) technology for unambiguous detection of key events. The QTouch library includes support for the QTouch and QMatrix acquisition methods.

Touch sensing can be added to any application by linking the appropriate Atmel QTouch library for the AVR microcontroller. This is done by using a simple set of APIs to define the touch channels and sensors, and then calling the touch sensing APIs to retrieve the channel information and determine the touch sensor states.

The QTouch library is FREE and downloadable from the Atmel website at the following location: www.atmel.com/qtouchlibrary. For implementation details and other information, refer to the [QTouch library user guide](#) - also available for download from the Atmel website.

6.7 Stack and Stack Pointer

The stack is used for storing return addresses after interrupts and subroutine calls. It can also be used for storing temporary data. The stack pointer (SP) register always points to the top of the stack. It is implemented as two 8-bit registers that are accessible in the I/O memory space. Data are pushed and popped from the stack using the PUSH and POP instructions. The stack grows from a higher memory location to a lower memory location. This implies that pushing data onto the stack decreases the SP, and popping data off the stack increases the SP. The SP is automatically loaded after reset, and the initial value is the highest address of the internal SRAM. If the SP is changed, it must be set to point above address 0x2000, and it must be defined before any subroutine calls are executed or before interrupts are enabled.

During interrupts or subroutine calls, the return address is automatically pushed on the stack. The return address can be two or three bytes, depending on program memory size of the device. For devices with 128KB or less of program memory, the return address is two bytes, and hence the stack pointer is decremented/incremented by two. For devices with more than 128KB of program memory, the return address is three bytes, and hence the SP is decremented/incremented by three. The return address is popped off the stack when returning from interrupts using the RETI instruction, and from subroutine calls using the RET instruction.

The SP is decremented by one when data are pushed on the stack with the PUSH instruction, and incremented by one when data is popped off the stack using the POP instruction.

To prevent corruption when updating the stack pointer from software, a write to SPL will automatically disable interrupts for up to four instructions or until the next I/O memory write.

After reset the stack pointer is initialized to the highest address of the SRAM. See [Figure 7-3 on page 16](#).

6.8 Register File

The register file consists of 32 x 8-bit general purpose working registers with single clock cycle access time. The register file supports the following input/output schemes:

- One 8-bit output operand and one 8-bit result input
- Two 8-bit output operands and one 8-bit result input
- Two 8-bit output operands and one 16-bit result input
- One 16-bit output operand and one 16-bit result input

Six of the 32 registers can be used as three 16-bit address register pointers for data space addressing, enabling efficient address calculations. One of these address pointers can also be used as an address pointer for lookup tables in flash program memory.

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 A3U devices are shown in [Table 14-1](#). 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](#). The program address is the word address.

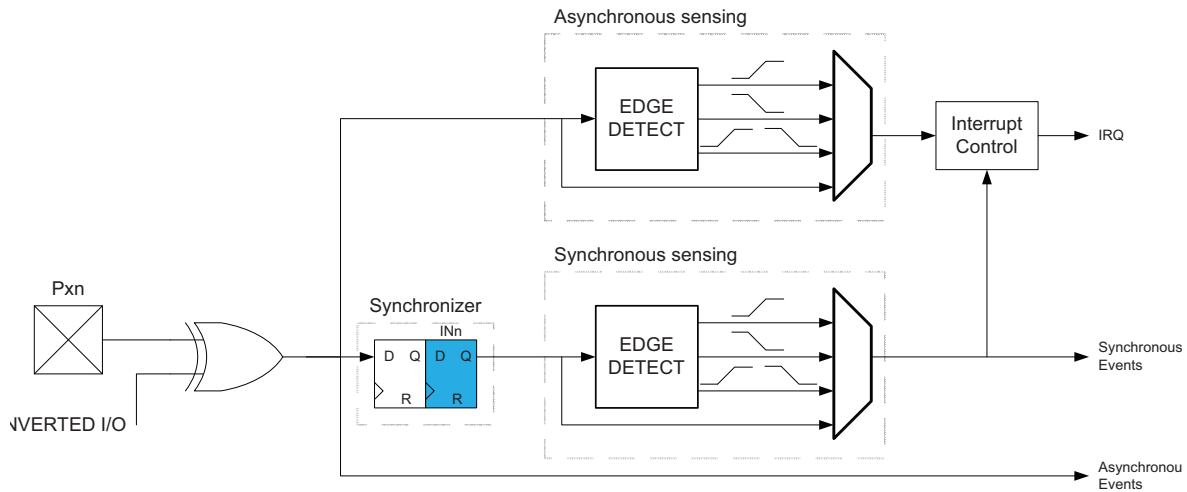
Table 14-1. Reset and interrupt vectors.

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

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 59](#) shows which modules on peripherals that enable alternate functions on a pin, and which alternate functions that are available on a pin.

36.2.4 Wake-up time from sleep modes

Table 36-38. Device wake-up time from sleep modes with various system clock sources.

Symbol	Parameter	Condition	Min.	Typ. ⁽¹⁾	Max.	Units
t _{wakeup}	Wake-up time from Idle, Standby, and Extended Standby mode	External 2MHz clock		2		µs
		32.768kHz internal oscillator		120		
		2MHz internal oscillator		2		
		32MHz internal oscillator		0.2		
	Wake-up time from Power-save and Power-down mode	External 2MHz clock		4.5		µs
		32.768kHz internal oscillator		320		
		2MHz internal oscillator		9		
		32MHz internal oscillator		5		

Note: 1. The wake-up time is the time from the wake-up request is given until the peripheral clock is available on pin, see Figure 36-2. All peripherals and modules start execution from the first clock cycle, except the CPU that is halted for four clock cycles before program execution starts.

Figure 36-9. Wake-up time definition.

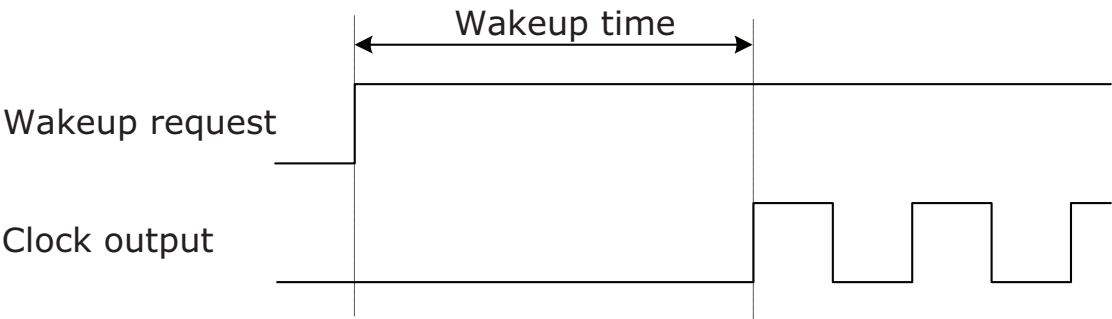


Table 36-69. Current consumption for modules and peripherals.

Symbol	Parameter	Condition ⁽¹⁾	Min.	Typ.	Max.	Units
I _{CC}	ULP oscillator			1.0		μA
	32.768kHz int. oscillator			27		μA
	2MHz int. oscillator			85		μA
		DFLL enabled with 32.768kHz int. osc. as reference		115		
	32MHz int. oscillator			270		μA
		DFLL enabled with 32.768kHz int. osc. as reference		460		
	PLL	20x multiplication factor, 32MHz int. osc. DIV4 as reference		220		μA
	Watchdog Timer			1		μA
	BOD	Continuous mode		138		μA
		Sampled mode, includes ULP oscillator		1.2		
	Internal 1.0V reference			100		μA
	Temperature sensor			95		μA
	ADC	250ksps V _{REF} = Ext ref		3.0		mA
			CURRLIMIT = LOW	2.6		
			CURRLIMIT = MEDIUM	2.1		
			CURRLIMIT = HIGH	1.6		
	DAC	250ksps V _{REF} = Ext ref No load	Normal mode	1.9		mA
			Low Power mode	1.1		
	AC	High Speed Mode		330		μA
		Low Power Mode		130		
	DMA	615KBps between I/O registers and SRAM		115		μA
	Timer/Counter			16		μA
	USART	Rx and Tx enabled, 9600 BAUD		2.5		μA
	Flash memory and EEPROM programming			4		mA

Note: 1. All parameters measured as the difference in current consumption between module enabled and disabled. All data at V_{CC} = 3.0V, Clk_{SYS} = 1MHz external clock without prescaling, T = 25°C unless other conditions are given. All parameters measured as the difference in current consumption between module enabled and disabled. All data at V_{CC} = 3.0V, Clk_{SYS} = 1MHz external clock without prescaling, T = 25°C unless other conditions are given.

36.4 ATxmega256A3U

36.4.1 Absolute Maximum Ratings

Stresses beyond those listed in [Table 36-1](#) under may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Table 36-97. Absolute maximum ratings.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
V_{CC}	Power Supply Voltage		-0.3		4	V
I_{VCC}	Current into a V_{CC} pin				200	mA
I_{GND}	Current out of a Gnd pin				200	mA
V_{PIN}	Pin voltage with respect to Gnd and V_{CC}		-0.5		$V_{CC}+0.5$	V
I_{PIN}	I/O pin sink/source current		-25		25	mA
T_A	Storage temperature		-65		150	°C
T_j	Junction temperature				150	°C

36.4.2 General Operating Ratings

The device must operate within the ratings listed in [Table 36-2](#) in order for all other electrical characteristics and typical characteristics of the device to be valid.

Table 36-98. General operating conditions.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
V_{CC}	Power Supply Voltage		1.60		3.6	V
AV_{CC}	Analog Supply Voltage		1.60		3.6	V
T_A	Temperature range	85 °C	-40		85	°C
		105 °C	-40		105	
T_j	Junction temperature	85°C	-40		105	°C
		105°C	-40		125	

Table 36-99. Operating voltage and frequency.

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
Clk_{CPU}	CPU clock frequency	$V_{CC} = 1.6V$	0		12	MHz
		$V_{CC} = 1.8V$	0		12	
		$V_{CC} = 2.7V$	0		32	
		$V_{CC} = 3.6V$	0		32	

36.4.6 ADC characteristics

Table 36-104. 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		$AV_{CC} - 0.6$	V
R_{in}	Input resistance	Switched		4.0		k Ω
C_{sample}	Input capacitance	Switched		4.4		pF
R_{AREF}	Reference input resistance	(leakage only)		>10		M Ω
C_{AREF}	Reference input capacitance	Static load		7		pF
V_{IN}	Input range		-0.1		$AV_{CC} + 0.1$	V
	Conversion range	Differential mode, $V_{inP} - V_{inN}$	$-V_{REF}$		V_{REF}	V
V_{IN}	Conversion range	Single ended unsigned mode, V_{inP}	$-\Delta V$		$V_{REF} - \Delta V$	V
ΔV	Fixed offset voltage			190		LSB

Table 36-105. 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	μ s
	Conversion time (latency)	(RES+2)/2+(GAIN !=0) 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	

37.1.1.3 Power-down mode supply current

Figure 37-15. Power-down mode supply current vs. V_{CC} .
All functions disabled.

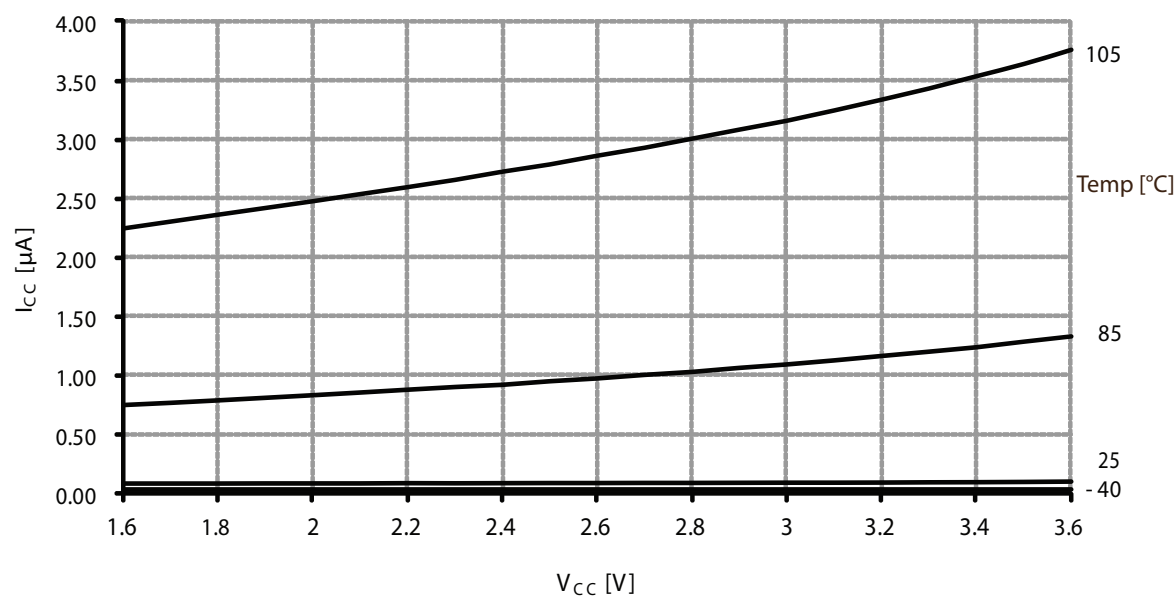


Figure 37-16. Power-down mode supply current vs. V_{CC} .
Watchdog and sampled BOD enabled.

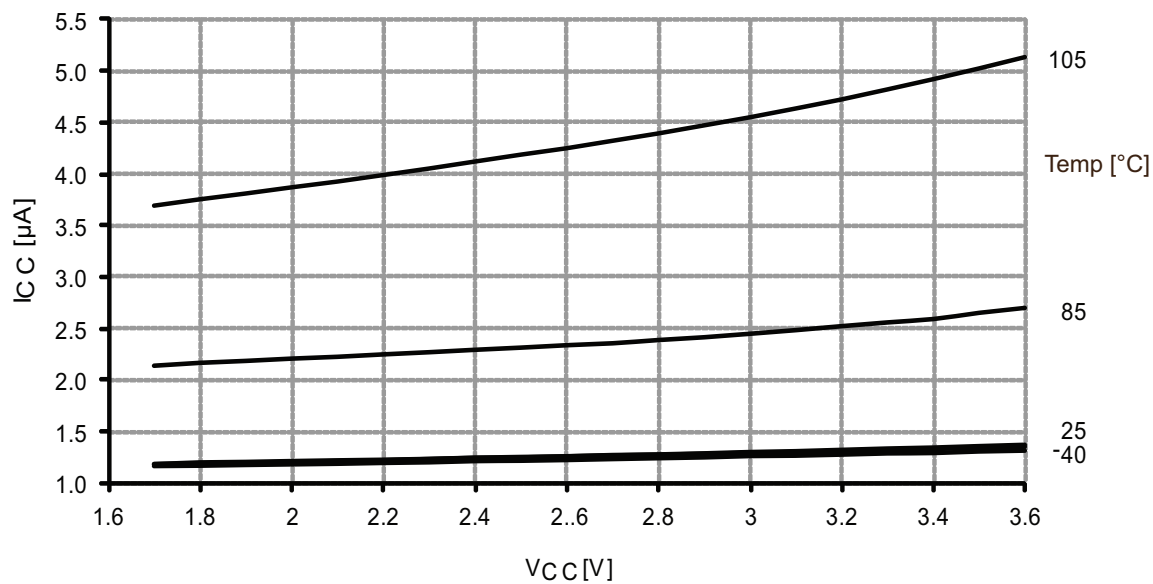


Figure 37-37. INL error vs. input code.

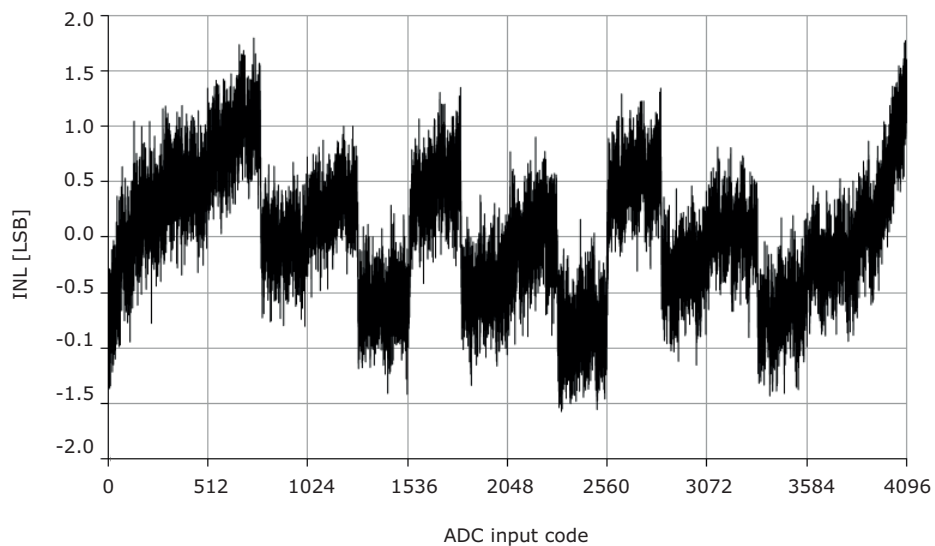


Figure 37-38. DNL error vs. external V_{REF} .
 $T = 25^{\circ}\text{C}$, $V_{CC} = 3.6\text{V}$, external reference.

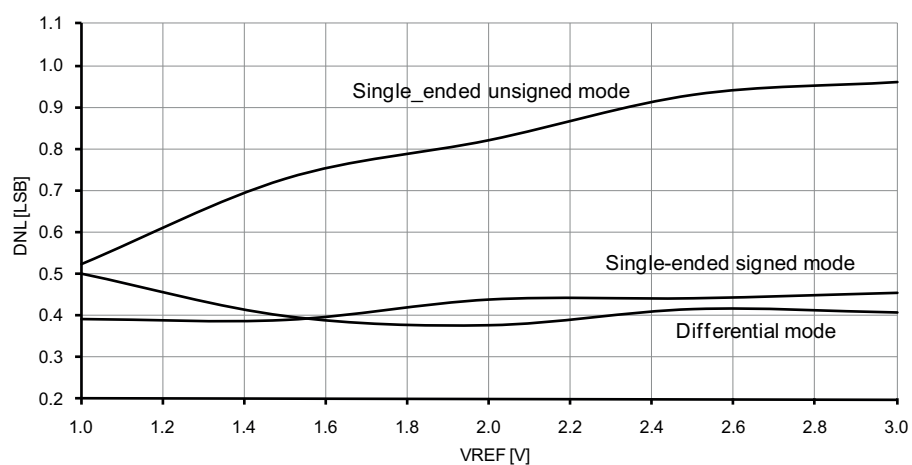


Figure 37-112. I/O pin output voltage vs. sink current.

$V_{CC} = 3.3V$.

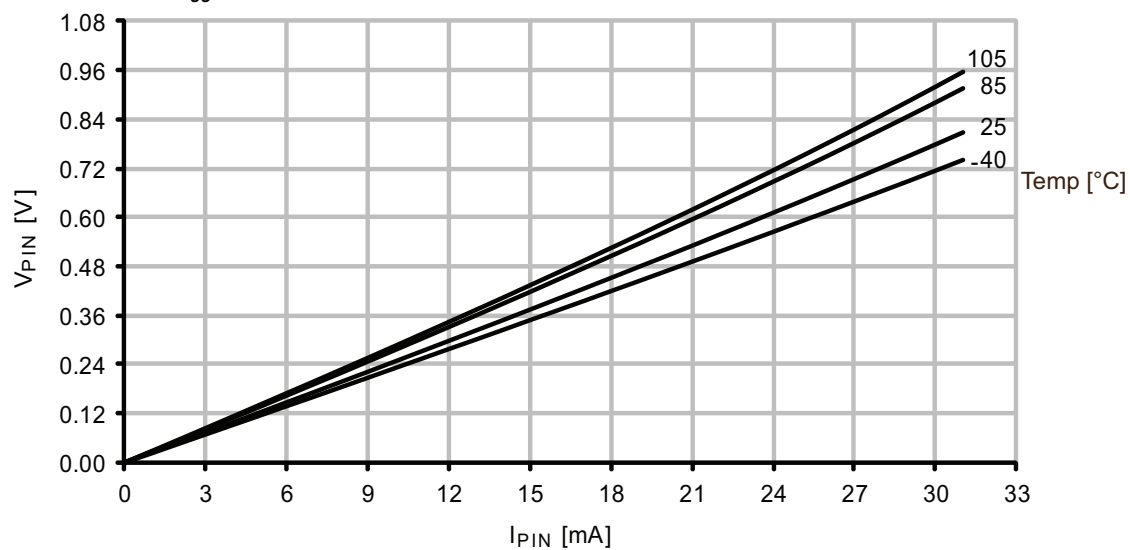


Figure 37-113. I/O pin output voltage vs. sink current.

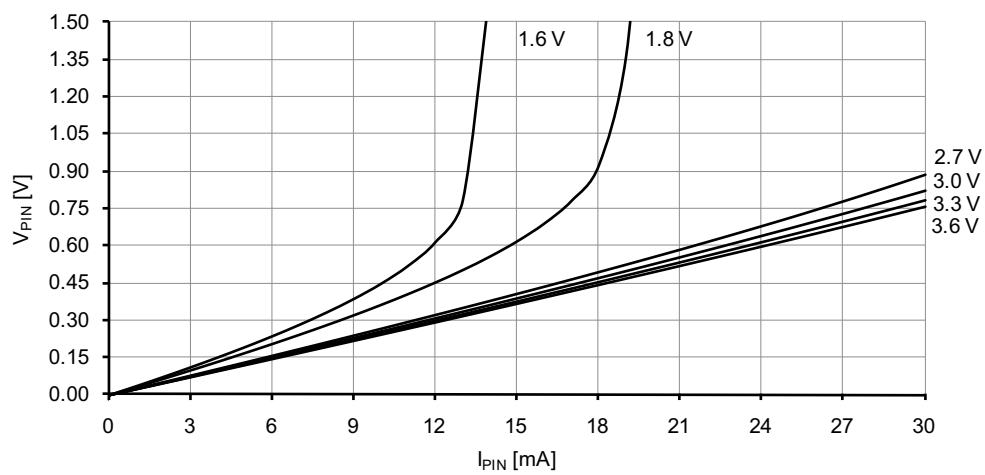


Figure 37-215. DNL error vs. V_{REF} .
 $T = 25^{\circ}\text{C}$, $V_{CC} = 3.6\text{V}$.

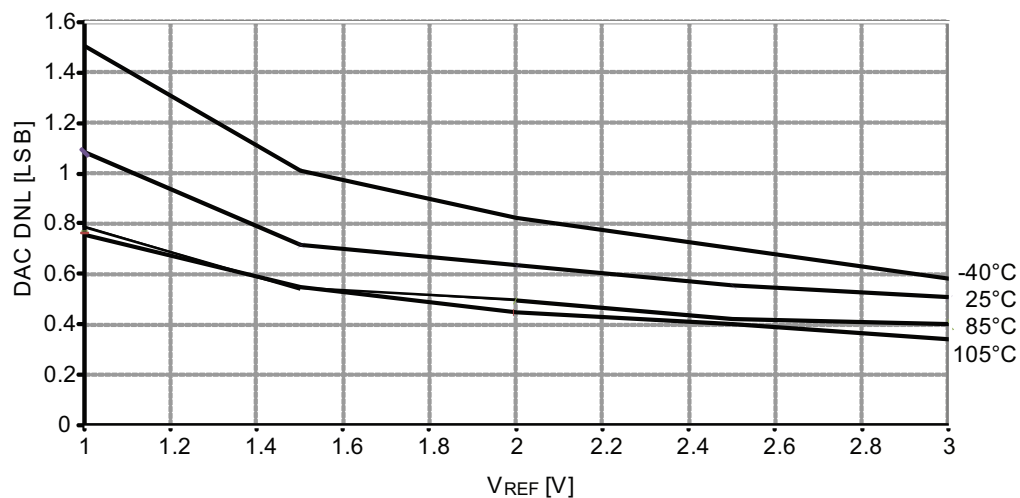


Figure 37-216. DAC noise vs. temperature.
 $V_{CC} = 3.3\text{V}$, $V_{REF} = 2.0\text{V}$.

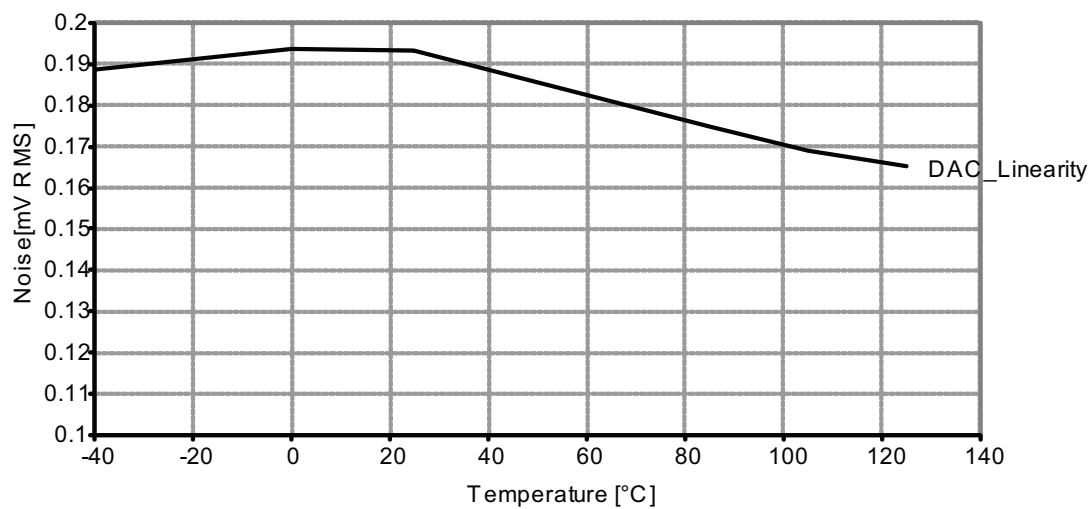
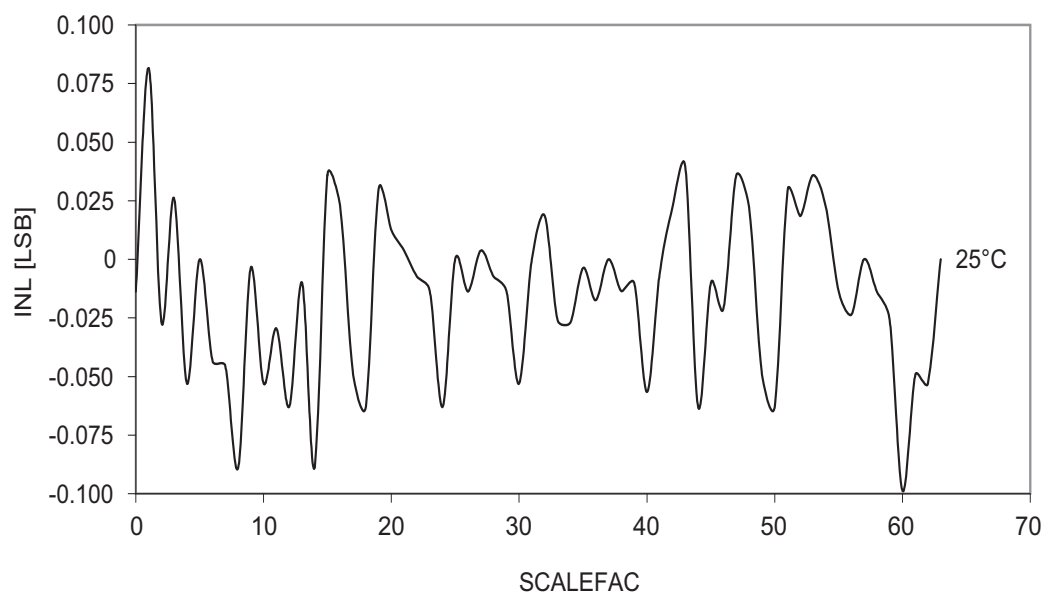


Figure 37-223. Voltage scaler INL vs. SCALEFAC.

$T = 25^{\circ}\text{C}$, $V_{CC} = 3.0\text{V}$.



37.3.6 Internal 1.0V reference Characteristics

Figure 37-224. ADC/DAC Internal 1.0V reference vs. temperature.

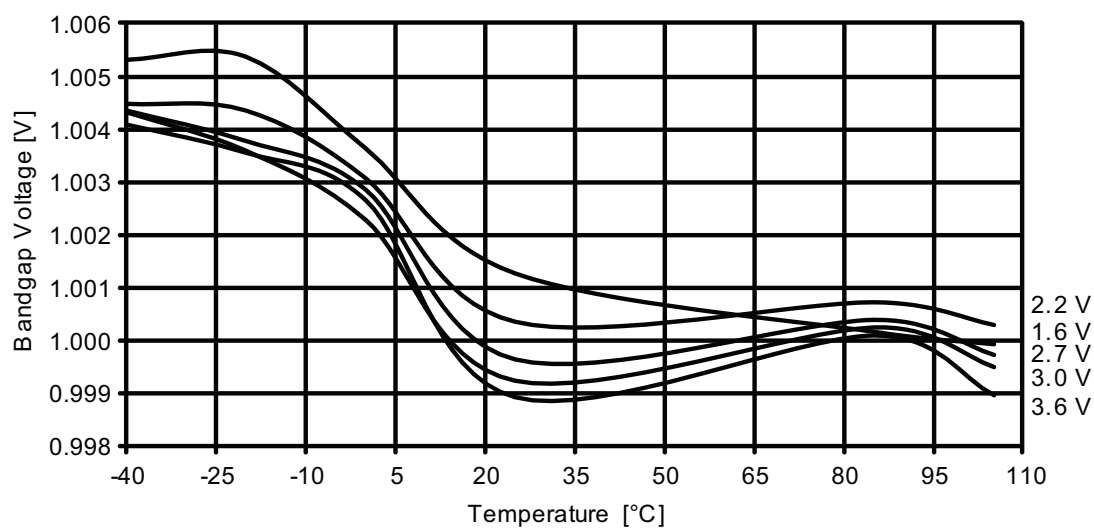
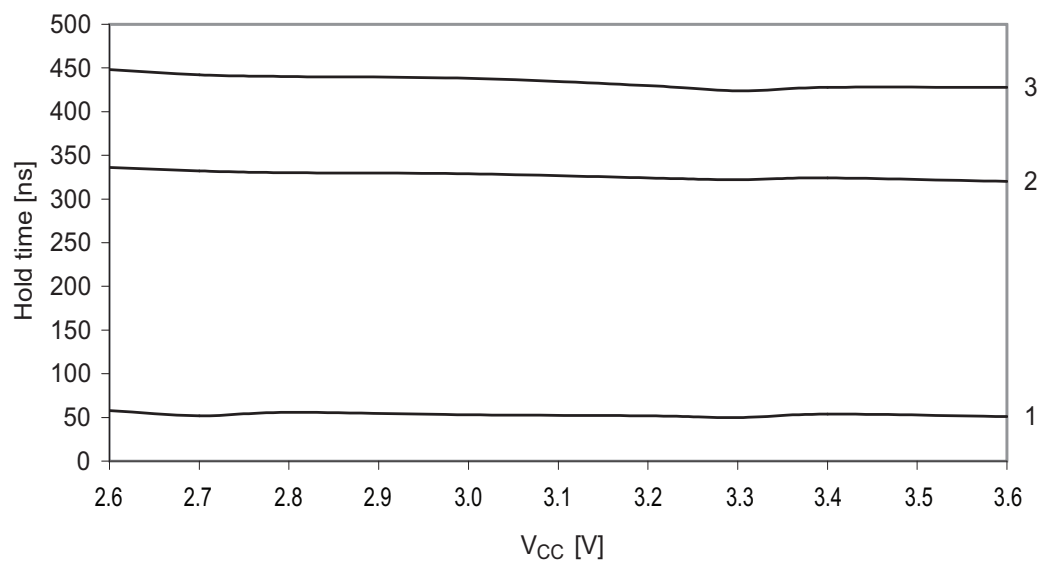


Figure 37-248. SDA hold time vs. supply voltage.



37.3.12 PDI characteristics

Figure 37-249. Maximum PDI frequency vs. V_{CC} .

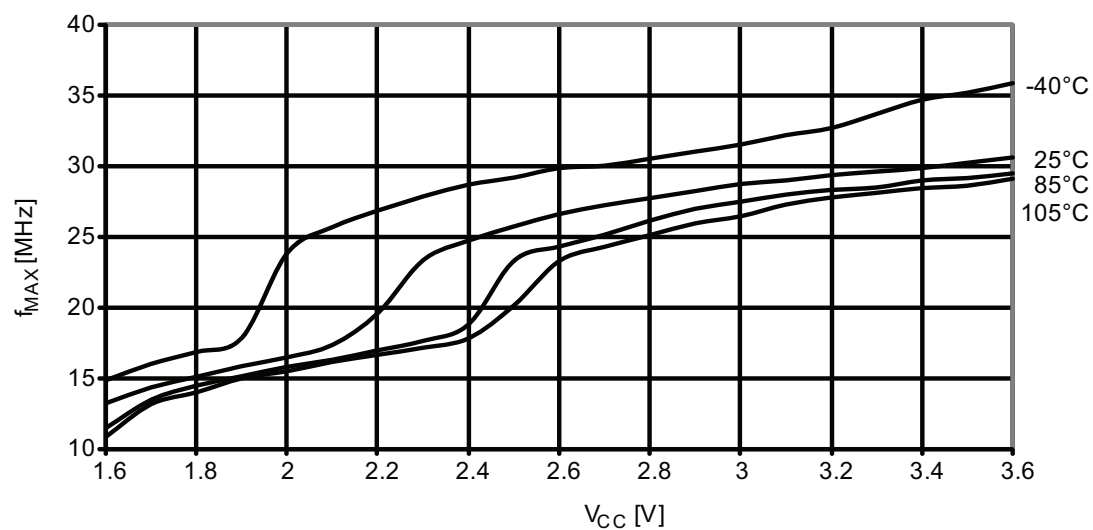


Figure 37-286. INL error vs. input code.

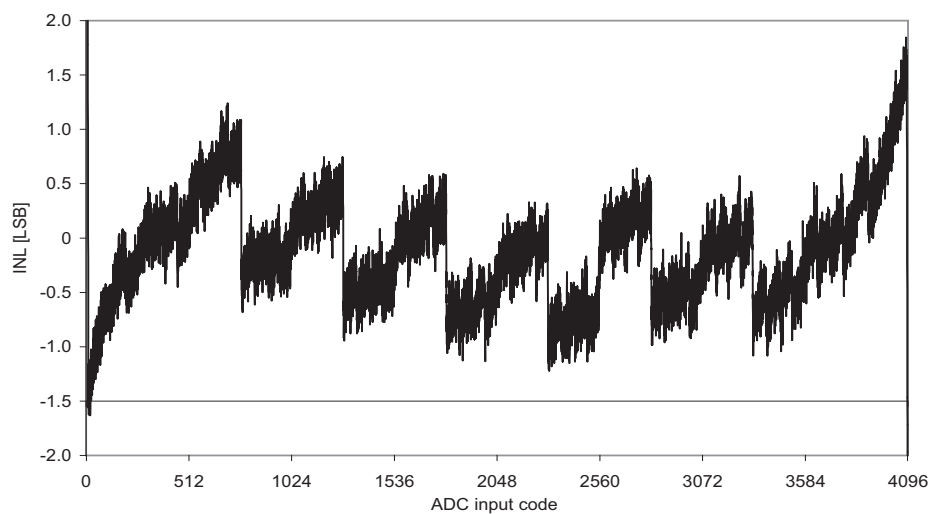


Figure 37-287. DNL error vs. external V_{REF} .
 $T = 25^{\circ}\text{C}$, $V_{CC} = 3.6\text{V}$, external reference.

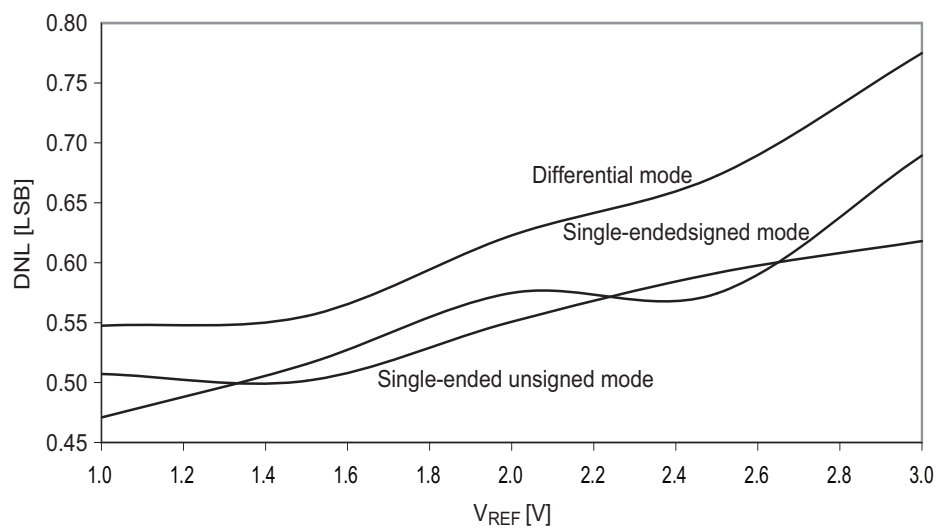


Figure 37-290. Gain error vs. V_{REF} .

$T = 25^{\circ}\text{C}$, $V_{CC} = 3.6\text{V}$, ADC sampling speed = 500kps.

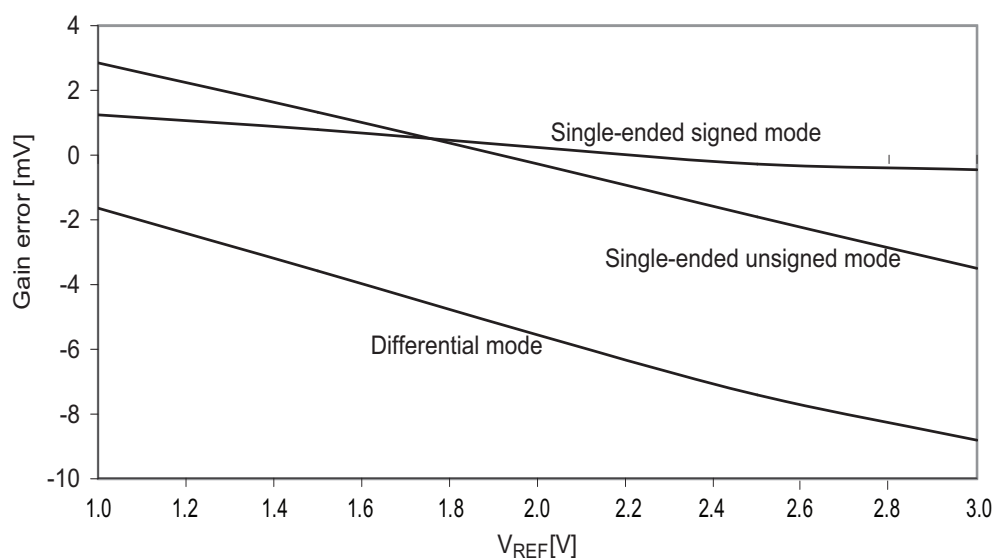


Figure 37-291. Gain error vs. V_{CC} .

$T = 25^{\circ}\text{C}$, $V_{REF} = \text{external } 1.0\text{V}$, ADC sampling speed = 500kps.

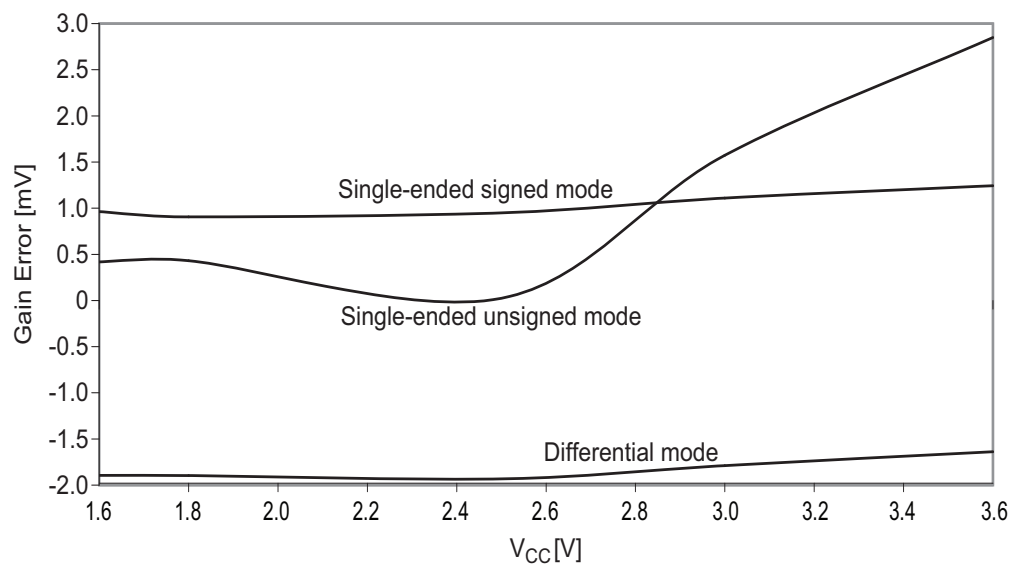


Figure 37-304. Analog comparator current source vs. calibration value.
Temperature = 25°C.

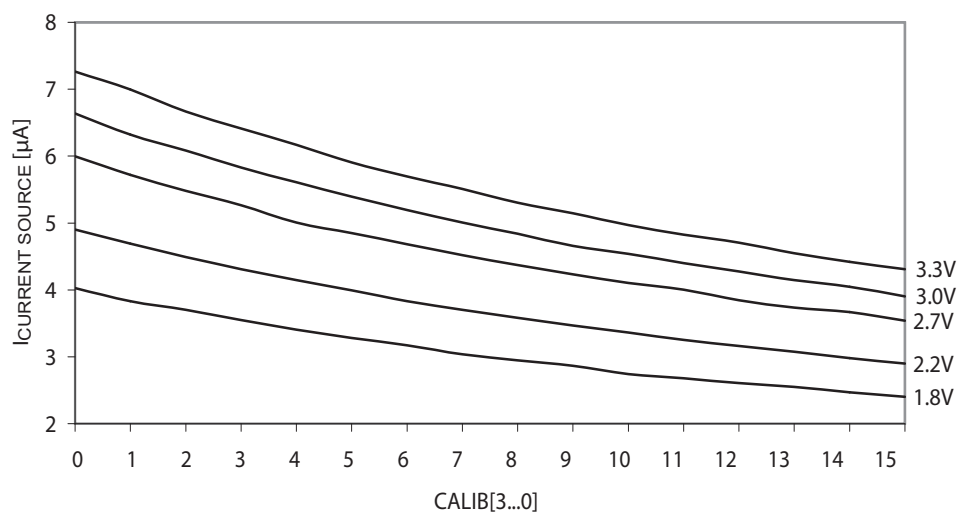


Figure 37-305. Analog comparator current source vs. calibration value.
 $V_{CC} = 3.0\text{V}$.

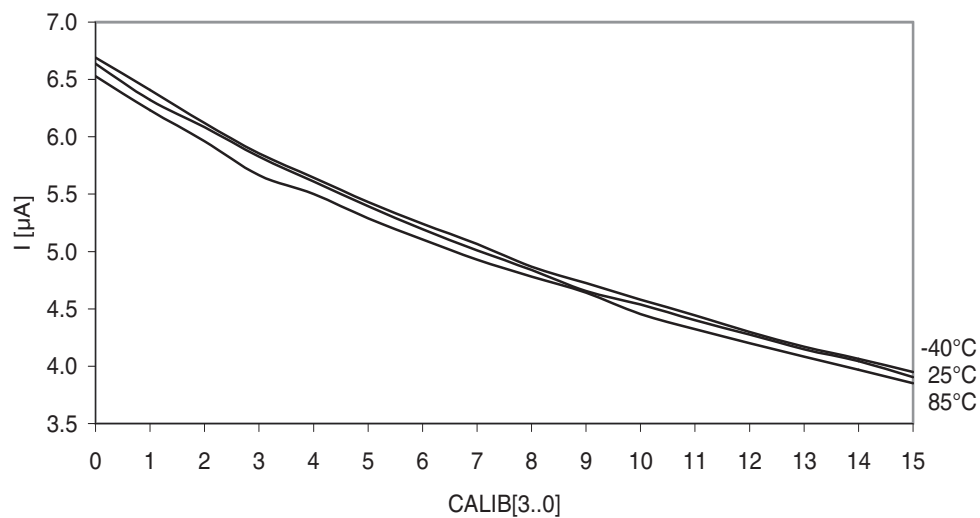
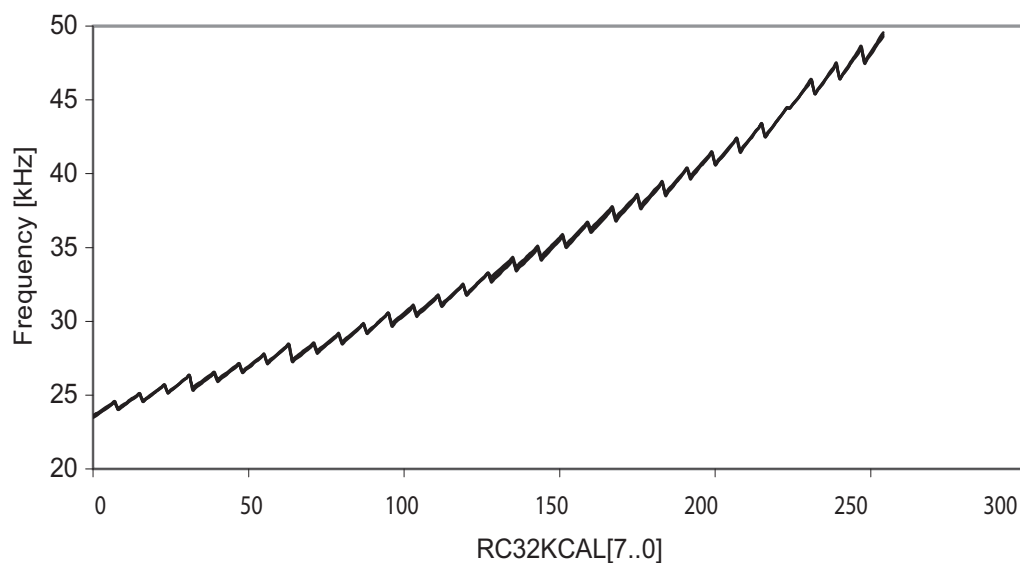


Figure 37-319. 32.768kHz internal oscillator frequency vs. calibration value.

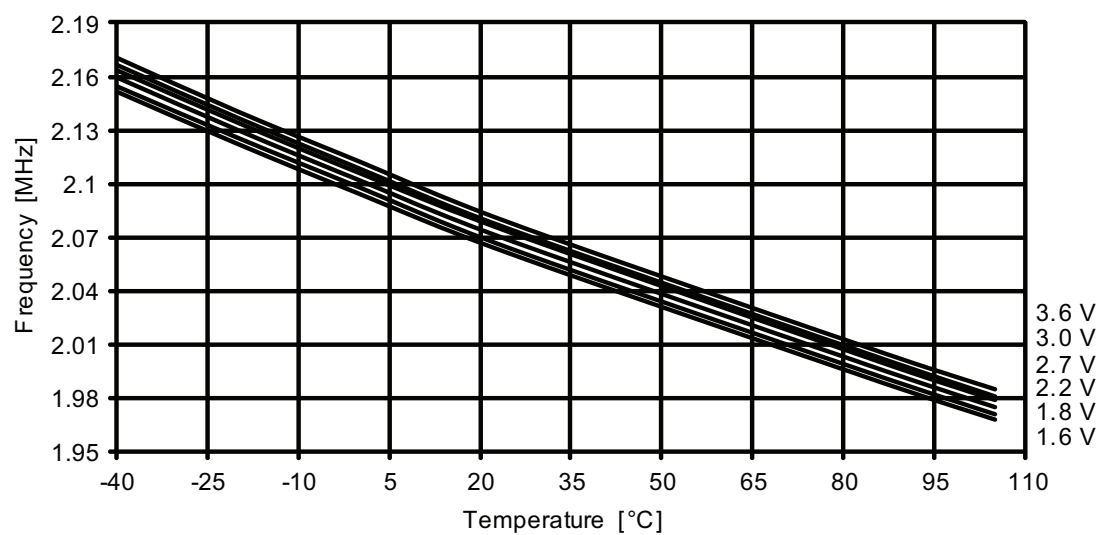
$V_{CC} = 3.0V$, $T = 25^{\circ}C$.



37.4.10.3 2MHz Internal Oscillator

Figure 37-320. 2MHz internal oscillator frequency vs. temperature.

DPLL disabled.



38. Errata

38.1 ATxmega64A3U, ATxmega128A3U, ATxmega192A3U, ATxmega256A3U

38.1.1 Rev. G

- The DAC Channel 1 has not been calibrated in the Xmega devices released prior to April 2012.
- AWeX fault protection restore is not done correct in Pattern Generation Mode.

1. 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

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.

For PGM in cycle-by-cycle mode there is no workaround.

38.1.2 Rev. A-F

Not sampled.