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 ~ 105°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-cnr

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 API's 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.

The available memory size configurations are shown in “Ordering Information” on page 2. In addition each device has a flash memory signature rows for calibration data, device identification, serial number etc.

7.3 Flash Program Memory

The Atmel AVR XMEGA devices contain on-chip, in-system reprogrammable flash memory for program storage. The flash memory can be accessed for read and write from an external programmer through the PDI or from application software running in the device.

All AVR CPU instructions are 16 or 32 bits wide, and each flash location is 16 bits wide. The flash memory is organized in two main sections, the application section and the boot loader section. The sizes of the different sections are fixed, but device-dependent. These two sections have separate lock bits, and can have different levels of protection. The store program memory (SPM) instruction, which is used to write to the flash from the application software, will only operate when executed from the boot loader section.

The application section contains an application table section with separate lock settings. This enables safe storage of nonvolatile data in the program memory.

Figure 7-1. Flash program memory (Hexadecimal address).

Word Address			
ATxmega128A1U		**ATxmega64A1U**	
0		0	Application Section (bytes) (128K/64K)
			...
EFFF	/	77FF	
F000	/	7800	Application Table Section (bytes) (8K/4K)
FFFF	/	7FFF	
10000	/	8000	Boot Section (bytes) (8K/4K)
10FFF	/	87FF	

7.3.1 Application Section

The Application section is the section of the flash that is used for storing the executable application code. The protection level for the application section can be selected by the boot lock bits for this section. The application section can not store any boot loader code since the SPM instruction cannot be executed from the application section.

7.3.2 Application Table Section

The application table section is a part of the application section of the flash memory that can be used for storing data. The size is identical to the boot loader section. The protection level for the application table section can be selected by the boot lock bits for this section. The possibilities for different protection levels on the application section and the application table section enable safe parameter storage in the program memory. If this section is not used for data, application code can reside here.

7.3.3 Boot Loader Section

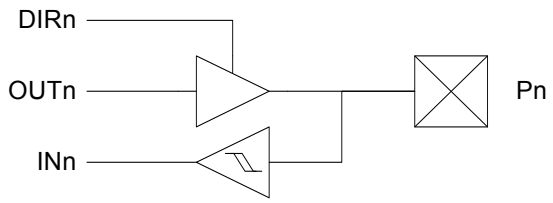
While the application section is used for storing the application code, the boot loader software must be located in the boot loader section because the SPM instruction can only initiate programming when executing from this section. The SPM instruction can access the entire flash, including the boot loader section itself. The protection level for the boot loader section can be selected by the boot loader lock bits. If this section is not used for boot loader software, application code can be stored here.

15.3 Output Driver

All port pins (Pn) have programmable output configuration. The port pins also have configurable slew rate limitation to reduce electromagnetic emission.

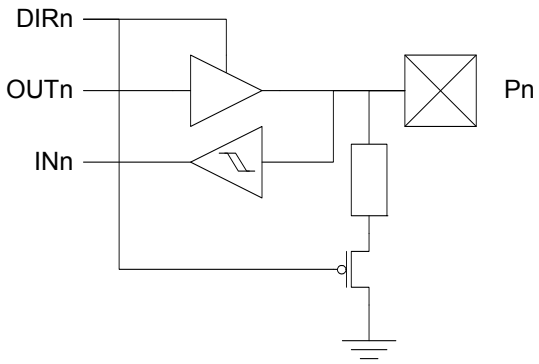
15.3.1 Push-pull

Figure 15-1. I/O configuration - Totem-pole.



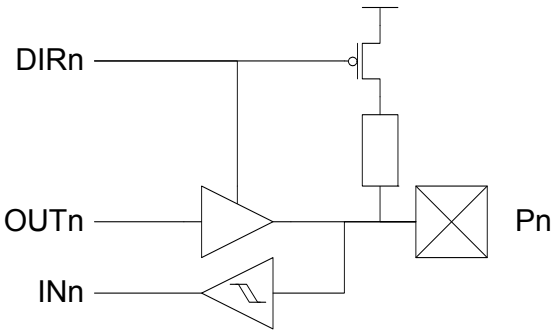
15.3.2 Pull-down

Figure 15-2. I/O configuration - Totem-pole with pull-down (on input).



15.3.3 Pull-up

Figure 15-3. I/O configuration - Totem-pole with pull-up (on input).



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 four hi-res extensions that each can be enabled for each timer/counters pair on PORTC, PORTD, PORTE and PORTF. The notation of these peripherals are HIRESC, HIRESD, HIRESE and HIRESF, respectively.

PORTC, PORTD, PORTE, and PORTF each has one TWI. Notation of these peripherals are TWIC, TWID, TWIE, and TWIF.

23. SPI – Serial Peripheral Interface

23.1 Features

- Four identical SPI peripherals
- Full-duplex, three-wire synchronous data transfer
- Master or slave operation
- Lsb first or msb first data transfer
- Eight programmable bit rates
- Interrupt flag at the end of transmission
- Write collision flag to indicate data collision
- Wake up from idle sleep mode
- Double speed master mode

23.2 Overview

The Serial Peripheral Interface (SPI) is a high-speed synchronous data transfer interface using three or four pins. It allows fast communication between an Atmel AVR XMEGA device and peripheral devices or between several microcontrollers. The SPI supports full-duplex communication.

A device connected to the bus must act as a master or slave. The master initiates and controls all data transactions.

PORTC, PORTD, PORTE, and PORTF each has one SPI. Notation of these peripherals are SPIC, SPID, SPIE, and SPIF.

32. PDI – Programming and Debugging

32.1 Features

- Programming
  - External programming through PDI or JTAG interfaces
    - Minimal protocol overhead for fast operation
    - Built-in error detection and handling for reliable operation
  - Boot loader support for programming through any communication interface
- Debugging
  - Nonintrusive, real-time, on-chip debug system
  - No software or hardware resources required from device except pin connection
  - Program flow control
    - Go, Stop, Reset, Step Into, Step Over, Step Out, Run-to-Cursor
  - Unlimited number of user program breakpoints
  - Unlimited number of user data breakpoints, break on:
    - Data location read, write, or both read and write
    - Data location content equal or not equal to a value
    - Data location content is greater or smaller than a value
    - Data location content is within or outside a range
  - No limitation on device clock frequency
- Program and Debug Interface (PDI)
  - Two-pin interface for external programming and debugging
  - Uses the Reset pin and a dedicated pin
  - No I/O pins required during programming or debugging
- JTAG interface
  - Four-pin, IEEE Std. 1149.1 compliant interface for programming and debugging
  - Boundary scan capabilities according to IEEE Std. 1149.1 (JTAG)

32.2 Overview

The Program and Debug Interface (PDI) is an Atmel proprietary interface for external programming and on-chip debugging of a device.

The PDI supports fast programming of nonvolatile memory (NVM) spaces; flash, EEPOM, fuses, lock bits, and the user signature row.

Debug is supported through an on-chip debug system that offers nonintrusive, real-time debug. It does not require any software or hardware resources except for the device pin connection. Using the Atmel tool chain, it offers complete program flow control and support for an unlimited number of program and complex data breakpoints. Application debug can be done from a C or other high-level language source code level, as well as from an assembler and disassembler level.

Programming and debugging can be done through two physical interfaces. The primary one is the PDI physical layer, which is available on all devices. This is a two-pin interface that uses the Reset pin for the clock input (PDI_CLK) and one other dedicated pin for data input and output (PDI_DATA). A JTAG interface is also available on most devices, and this can be used for programming and debugging through the four-pin JTAG interface. The JTAG interface is IEEE Std. 1149.1 compliant, and supports boundary scan. Any external programmer or on-chip debugger/emulator can be directly connected to either of these interfaces. Unless otherwise stated, all references to the PDI assume access through the PDI physical layer.

Table 33-3. Port C - alternate functions.

PORT C	PIN#	INTERRUPT	TCC0[1][2]	AWEXC	TCC1	USARTC0[3]	USARTC1	SPIC[4]	TWIC	CLOCKOUT[5]	EVENTOUT[6]
GND	13										
VCC	14										
PC0	15	SYNC	OC0A	$\overline{\text{OC0ALS}}$					SDA		
PC1	16	SYNC	OC0B	OC0AHS		XCK0			SCL		
PC2	17	SYNC/ASYNC	OC0C	$\overline{\text{OC0BLS}}$		RXD0					
PC3	18	SYNC	OC0D	OC0BHS		TXD0					
PC4	19	SYNC		$\overline{\text{OC0CLS}}$	OC1A			$\overline{\text{SS}}$			
PC5	20	SYNC		OC0CHS	OC1B		XCK1	MOSI			
PC6	21	SYNC		$\overline{\text{OC0DLS}}$			RXD1	MISO		clk_{RTC}	
PC7	22	SYNC		OC0DHS			TXD1	SCK		clk_{PER}	EVOUT

Notes:
1. Pin mapping of all TC0 can optionally be moved to high nibble of port. Refer to Pin Remap register in I/O Ports in the XMEGA AU Manual.
2. If TC0 is configured as TC2 all eight pins can be used for PWM output. Refer to Pin Remap register in I/O Ports in the XMEGA AU Manual..
3. Pin mapping of all USART0 can optionally be moved to high nibble of port. Refer to Pin Remap register in I/O Ports in the XMEGA AU Manual..
4. Pins MOSI and SCK for all SPI can optionally be swapped.Refer to Pin Remap register in I/O Ports in the XMEGA AU Manual.
5. CLKOUT can optionally be moved between port C, D and E and between pin 4 and 7. Refer to CLKEVOUT register in I/O port configuration in the XMEGA AU Manual.
6. EVOUT can optionally be moved between port C, D and E and between pin 4 and 7. Refer to CLKEVOUT register in I/O port configuration in the XMEGA AU Manual.

Table 33-4. Port D - alternate functions.

PORT D	PIN#	INTERRUPT	TCD0	TCD1	USBD	USARTD0	USARTD1	SPID	TWID	CLOCKOUT	EVENTOUT
GND	23										
VCC	24										
PD0	25	SYNC	OC0A						SDA		
PD1	26	SYNC	OC0B			XCK0			SCL		
PD2	27	SYNC/ASYNC	OC0C			RXD0					
PD3	28	SYNC	OC0D			TXD0					
PD4	29	SYNC		OC1A				$\overline{\text{SS}}$			
PD5	30	SYNC		OC1B			XCK1	MOSI			
PD6	31	SYNC			D-		RXD1	MISO			
PD7	32	SYNC			D+		TXD1	SCK		clk_{PER}	EVOUT

Table 33-5. Port E - alternate functions.

PORT E	PIN #	INTERRUPT	TCE0	AWEXE	TCE1	USARTE0	USARTE1	SPIE	TWIE	CLOCKOUT	EVENTOUT
GND	33										
VCC	34										
PE0	35	SYNC	OC0A	$\overline{\text{OC0ALS}}$					SDA		
PE1	36	SYNC	OC0B	OC0AHS		XCK0			SCL		

34. Peripheral Module Address Map

The address maps show the base address for each peripheral and module in XMEGA A1U. For complete register description and summary for each peripheral module, refer to the XMEGA AU manual.

Table 34-1. Peripheral module address map.

Base address	Name	Description
0x0000	GPIO	General Purpose IO Registers
0x0010	VPORT0	Virtual Port 0
0x0014	VPORT1	Virtual Port 1
0x0018	VPORT2	Virtual Port 2
0x001C	VPORT3	Virtual Port 3
0x0030	CPU	CPU
0x0040	CLK	Clock Control
0x0048	SLEEP	Sleep Controller
0x0050	OSC	Oscillator Control
0x0060	DFLLRC32M	DFLL for the 32MHz Internal RC Oscillator
0x0068	DFLLRC2M	DFLL for the 2MHz RC Oscillator
0x0070	PR	Power Reduction
0x0078	RST	Reset Controller
0x0080	WDT	Watch-Dog Timer
0x0090	MCU	MCU Control
0x00A0	PMIC	Programmable Multilevel Interrupt Controller
0x00B0	PORTCFG	Port Configuration
0x00C0	AES	AES Module
0x00D0	CRC	CRC Module
0x0100	DMA	DMA Controller
0x0180	EVSYS	Event System
0x01C0	NVM	Non Volatile Memory (NVM) Controller
0x0200	ADCA	Analog to Digital Converter on port A
0x0240	ADCB	Analog to Digital Converter on port B
0x0300	DACA	Digital to Analog Converter on port A
0x0320	DACB	Digital to Analog Converter on port B
0x0380	ACA	Analog Comparator pair on port A
0x0390	ACB	Analog Comparator pair on port B
0x0400	RTC	Real Time Counter

Mnemonics	Operands	Description	Operation	Flags	#Clocks
SEV		Set Two's Complement Overflow	V ← 1	V	1
CLV		Clear Two's Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	H	1
CLH		Clear Half Carry Flag in SREG	H ← 0	H	1
MCU control instructions					
BREAK		Break	(See specific descr. for BREAK)	None	1
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR)	None	1

Notes:
1. Cycle times for Data memory accesses assume internal memory accesses, and are not valid for accesses via the external RAM interface.
2. One extra cycle must be added when accessing Internal SRAM.

37.1.5 I/O Pin Characteristics

The I/O pins complies with the JEDEC LVTTL and LVCMOS specification and the high- and low level input and output voltage limits reflect or exceed this specification.

Table 37-7. I/O pin characteristics.

Symbol	Parameter	Condition		Min.	Typ.	Max.	Units
I_{OH} (1)/ I_{OL} (2)	I/O pin source/sink current			-20		20	mA
V_{IH}	High level input voltage	V_{CC} = 2.7 - 3.6V		2		V_{CC}+0.3	V
		V_{CC} = 2.0 - 2.7V		0.7*V_{CC}		V_{CC}+0.3	
		V_{CC} = 1.6 - 2.0V		0.7*V_{CC}		V_{CC}+0.3	
V_{IL}	Low level input voltage	V_{CC} = 2.7- 3.6V		-0.3		0.3*V_{CC}	
		V_{CC} = 2.0 - 2.7V		-0.3		0.3*V_{CC}	
		V_{CC} = 1.6 - 2.0V		-0.3		0.3*V_{CC}	
V_{OH}	High level output voltage	V_{CC} = 3.0 - 3.6V	I_{OH} = -2mA	2.4	0.94*V_{CC}		
		V_{CC} = 2.3 - 2.7V	I_{OH} = -1mA	2.0	0.96*V_{CC}		
			I_{OH} = -2mA	1.7	0.92*V_{CC}		
		V_{CC} = 3.3V	I_{OH} = -8mA	2.6	2.9		
		V_{CC} = 3.0V	I_{OH} = -6mA	2.1	2.6		
		V_{CC} = 1.8V	I_{OH} = -2mA	1.4	1.6		
V_{OL}	Low level output voltage	V_{CC} = 3.0 - 3.6V	I_{OL} = 2mA		0.05*V_{CC}	0.4	
		V_{CC} = 2.3 - 2.7V	I_{OL} = 1mA		0.03*V_{CC}	0.4	
			I_{OL} = 2mA		0.06*V_{CC}	0.7	
		V_{CC} = 3.3V	I_{OL} = 15mA		0.4	0.76	
		V_{CC} = 3.0V	I_{OL} = 10mA		0.3	0.64	
		V_{CC} = 1.8V	I_{OL} = 5mA		0.3	0.46	
I_{IN}	Input leakage current				<0.001	0.1	µA
R_P	I/O pin Pull/Buss keeper resistor				25		kΩ
R_{RST}	Reset pin pull-up resistor				25		
t_r	Pad rise time	No load			4.0		ns
			slew rate limitation		7.0		

Notes:
1. The sum of all I_{OH} for PORTA, PORTC, PORTD, PORTF, PORTH, PORTJ, PORTK must for each port not exceed 200mA.
   The sum of all I_{OH} for PORTB must not exceed 100mA.
   The sum of all I_{OH} for PORTQ, PORTR and PDI must not exceed 100mA.
2. The sum of all I_{OL} for PORTA, PORTC, PORTD, PORTF, PORTH, PORTJ, PORTK must for each port not exceed 200mA.
   The sum of all I_{OL} for PORTB must not exceed 100mA.
   The sum of all I_{OL} for PORTQ, PORTR and PDI must not exceed 100mA.

Figure 37-8. Maximum Frequency vs. V_{CC}.

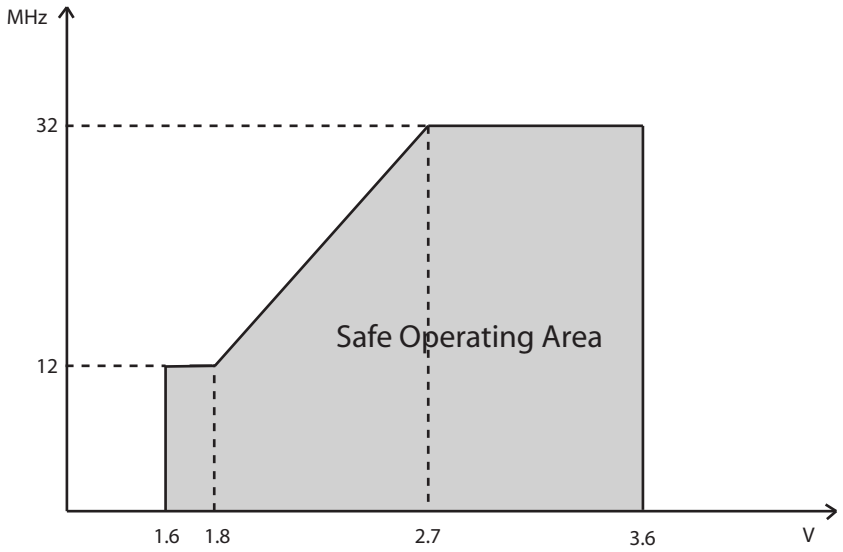


Figure 38-27.I/O pin output voltage vs. sink current.

V_{CC} = 1.8V.

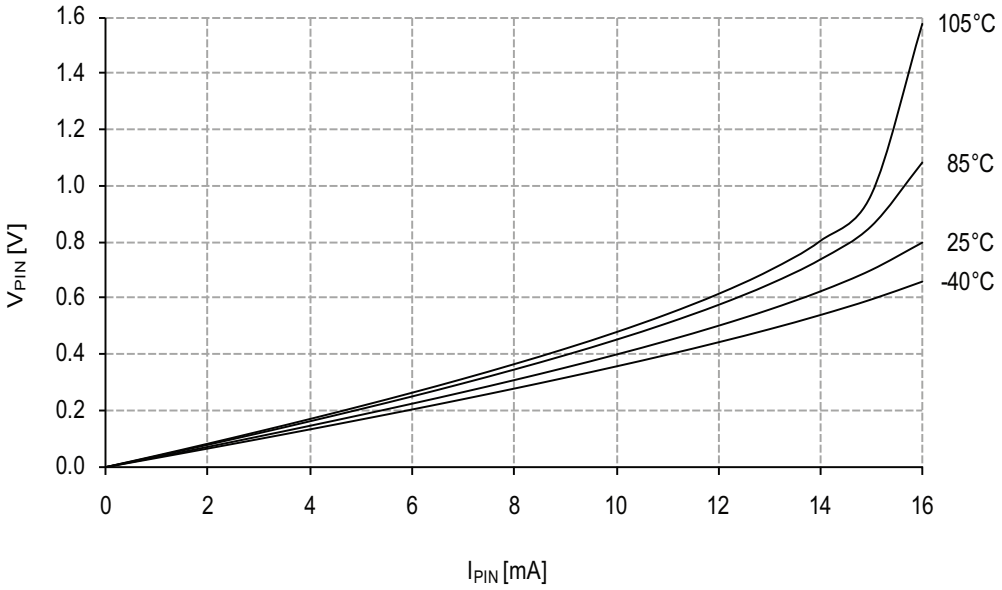


Figure 38-28.I/O pin output voltage vs. sink current.

V_{CC} = 3.0V.

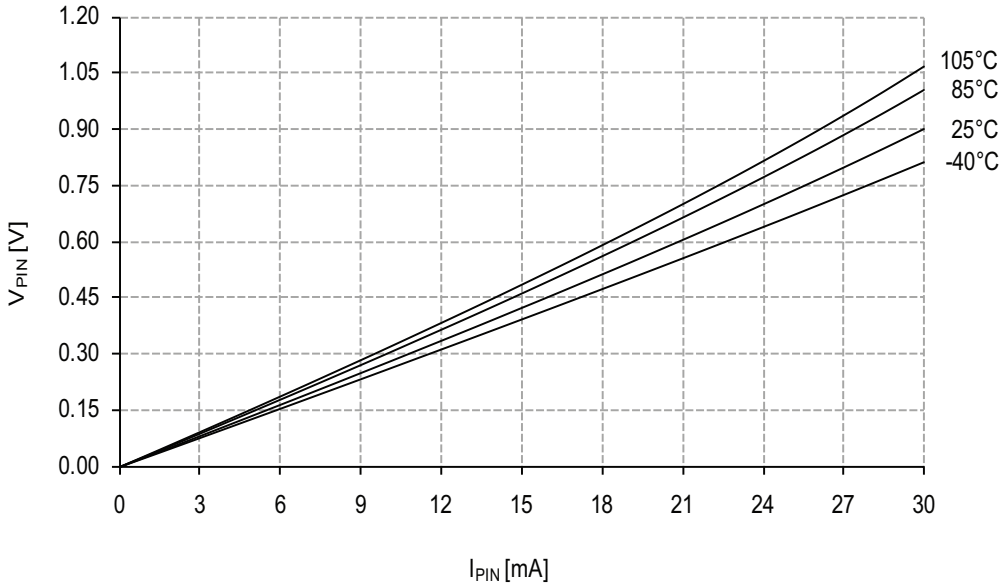


Figure 38-29.I/O pin output voltage vs. sink current.

V_{CC} = 3.3V.

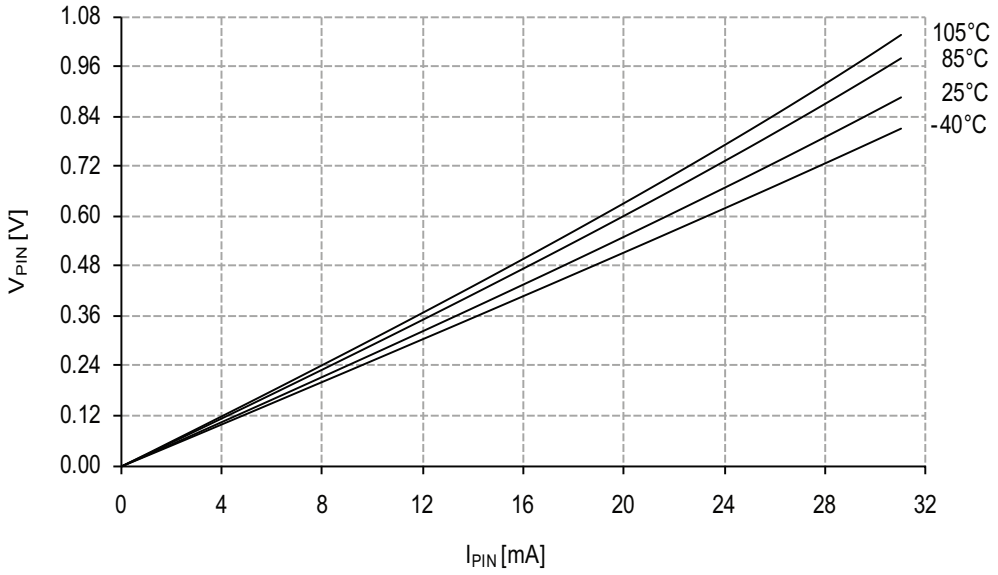


Figure 38-30.I/O pin output voltage vs. sink current.

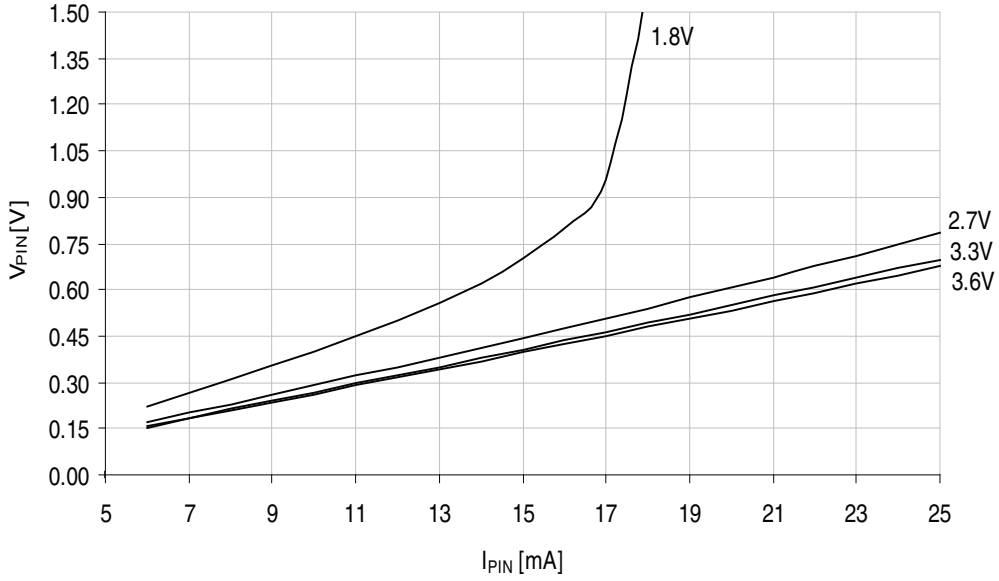


Figure 38-39.DNL error vs. sample rate.

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

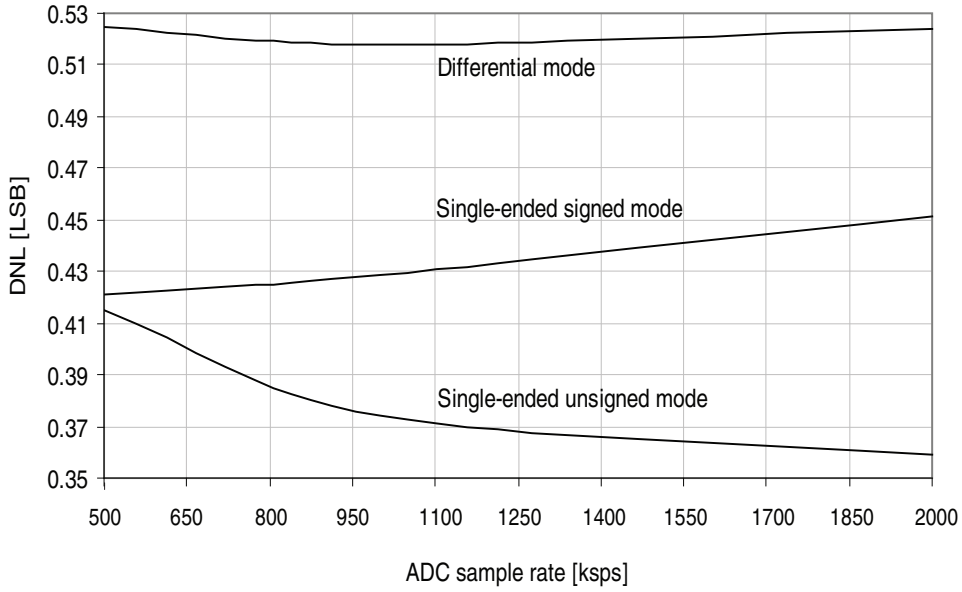


Figure 38-40.DNL error vs. input code.

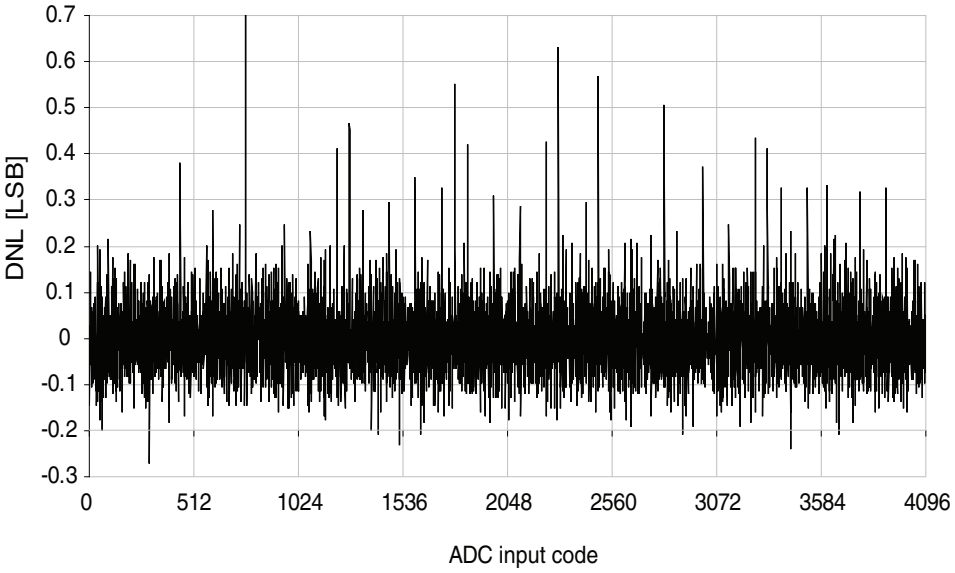


Figure 38-109.I/O pin output voltage vs. sink current.

V_{CC} = 1.8V.

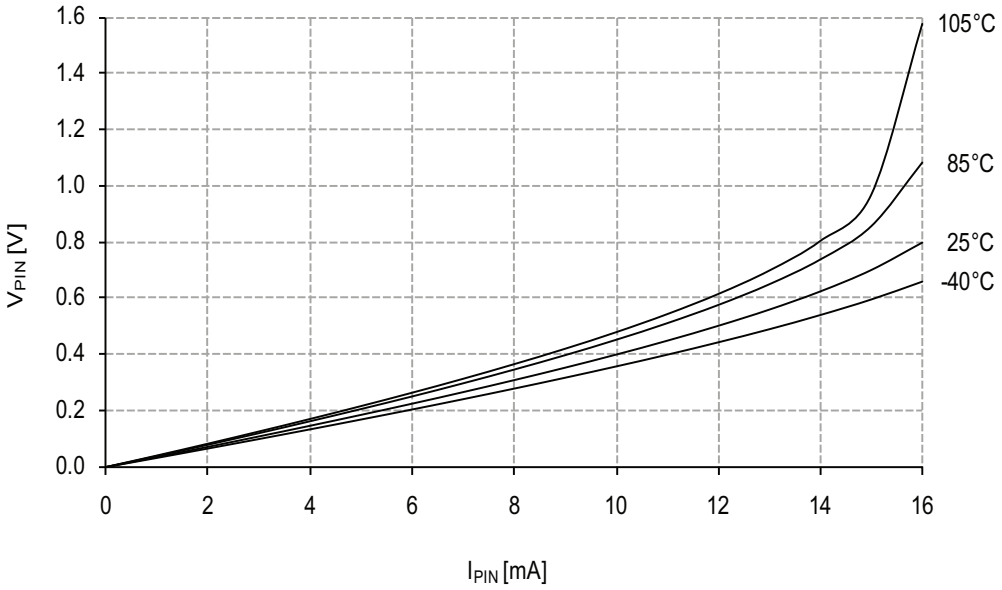


Figure 38-110.I/O pin output voltage vs. sink current.

V_{CC} = 3.0V.

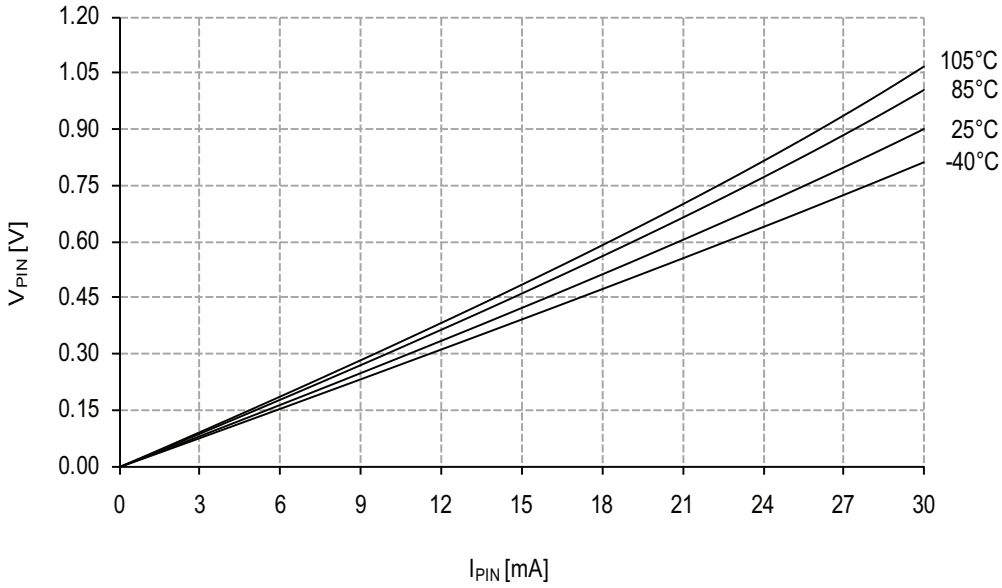


Figure 38-121.DNL error vs. sample rate.

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

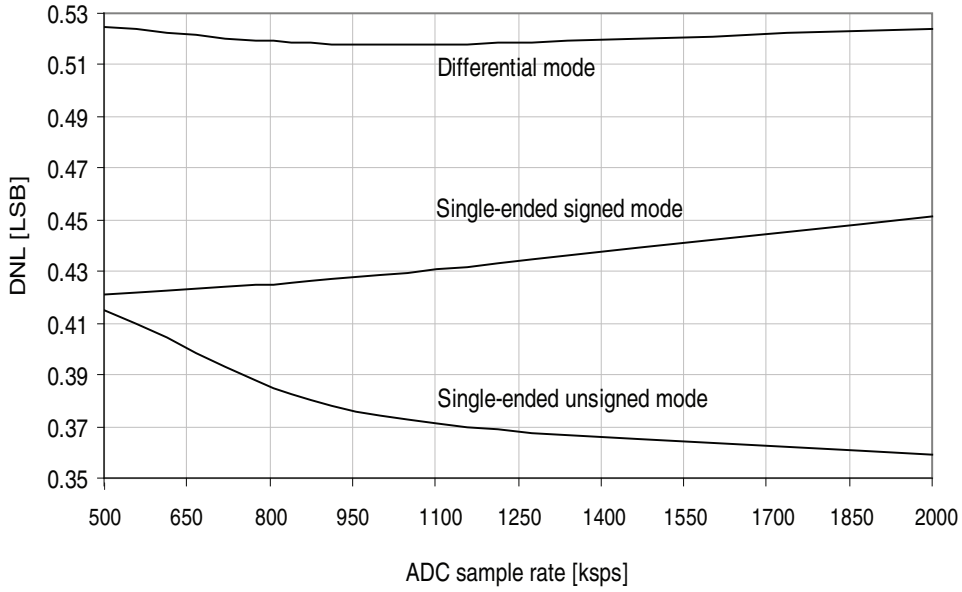


Figure 38-122.DNL error vs. input code.

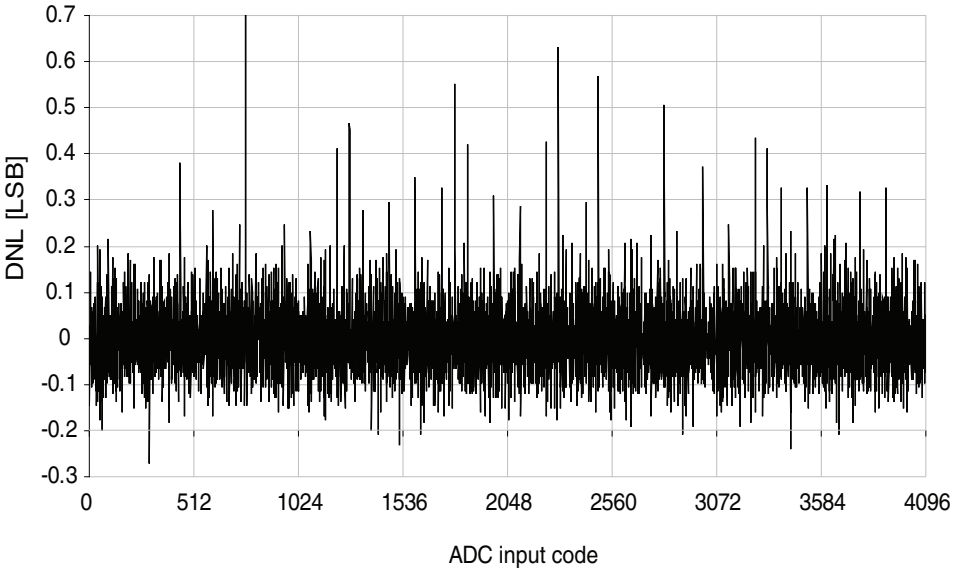
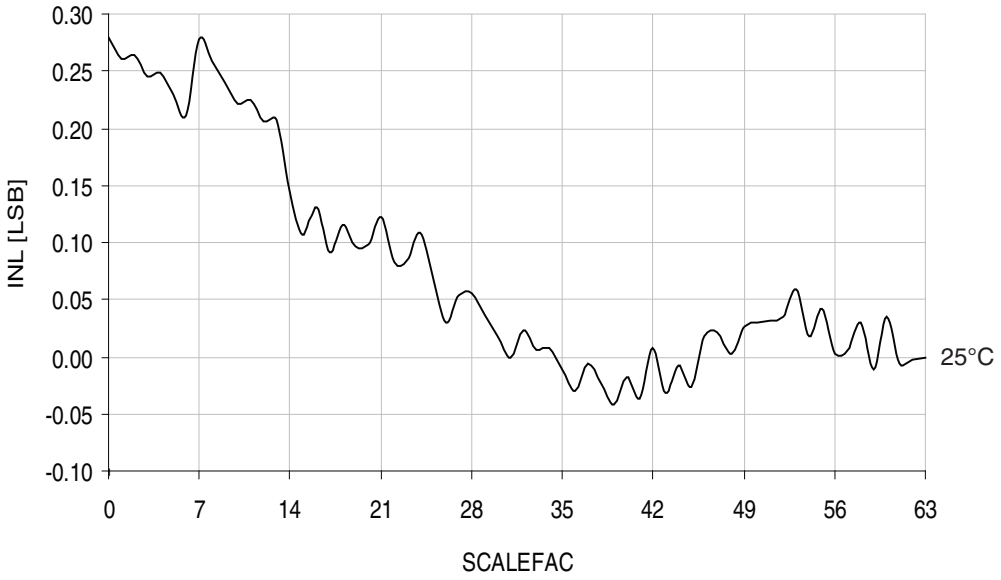


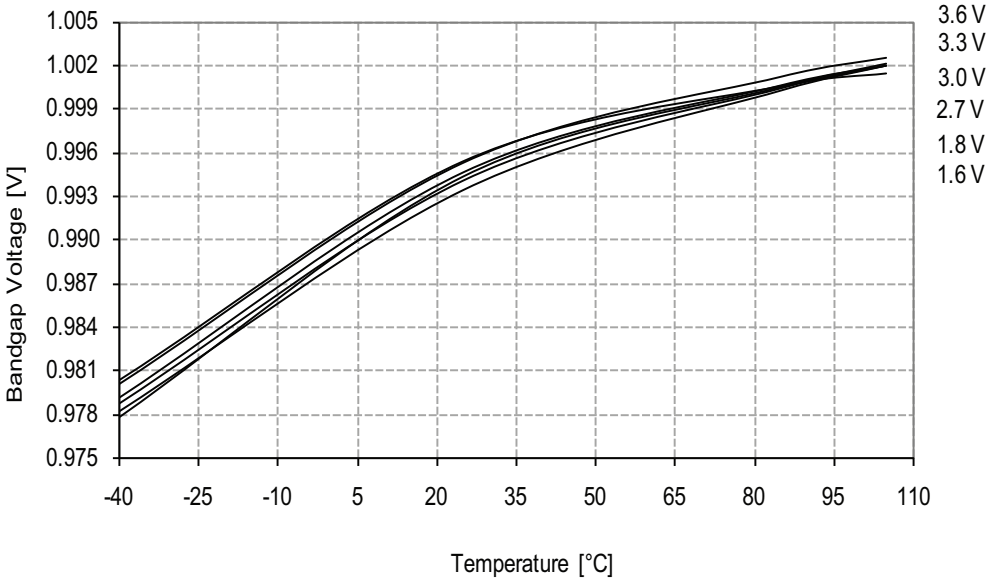
Figure 38-139.Voltage scaler INL vs. SCALEFAC.

T = 25°C, V_{CC} = 3.0V.



38.2.6 Internal 1.0V reference Characteristics

Figure 38-140. ADC/DAC Internal 1.0V reference vs. temperature.



38.2.9 Power-on Reset Characteristics

Figure 38-149.Power-on reset current consumption vs. V_{CC}.

BOD level = 3.0V, enabled in continuous mode.

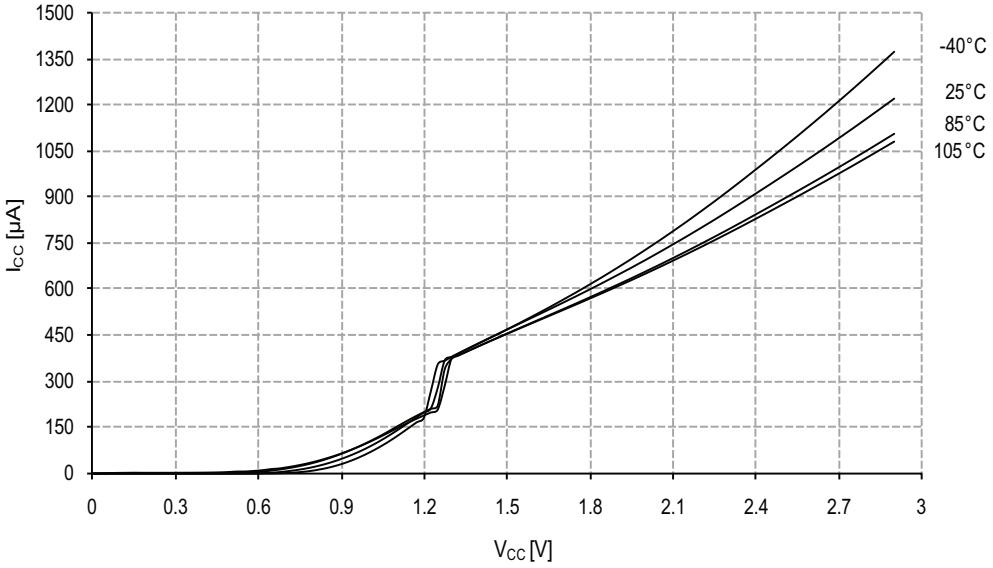
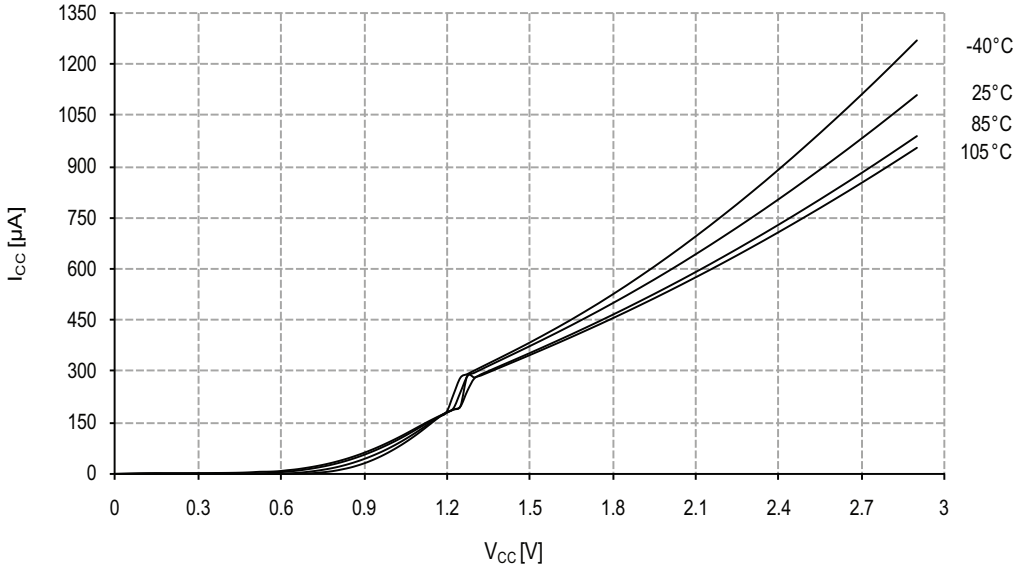


Figure 38-150.Power-on reset current consumption vs. V_{CC}.

BOD level = 3.0V, enabled in sampled mode.



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.

Problem fix/Workaround

Configure PWM and CWCM according to the Table 39-1 on page 203.

Table 39-1. PWM and CWCM configuration.

PGM	CWCM	Description
0	0	PGM and CWCM disabled
0	1	PGM enabled
1	0	PGM and CWCM enabled
1	1	PGM enabled

5. **AWEX PWM output after fault restarted with wrong values**

When recovering from fault state, the PWM output will drive wrong values to the port for up to two CLK_{PER} + one CLK_{PER4} cycles.

Problem fix/Workaround

The following seqence can be used in Latched Mode:

a. Disable DTI outputs (Write DTICCxEN to 0)
b. Clear fault flag
c. Wait for Overflow
d. Re-enable DTI (Write DTICCxEN to 1)
e. Set pin direction to Output

This will remove the glitch, but the following period will be shorter. In Cycle-by-cycle mode the same procedure can be followed as long as the Pattern Generation Mode is not enabled.

For Pattern generation mode, there is no workaround.

6. **RTC Counter value not correctly read after sleep**

If a real time counter (RTC) interrupt is used wake up the device from sleep, and bit 0 of RTC count register (CNT) has the same value as when the device entered sleep, CNT will not be read correctly during the first prescaled RTC clock cycle after wakeup. The value read will be the same as the value in the register was when entering sleep.

Problem fix/Workaround

Wait at least one prescaled RTC clock cycle before reading CNT.

7. **RTC clock output option is non-functional**

The real time counter (RTC) as clock output option is non-functional, and setting the RTCOUT bit in the clock and event out register (CLKEVOUT) will have no effect.

Problem fix/Workaround

None

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.