Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

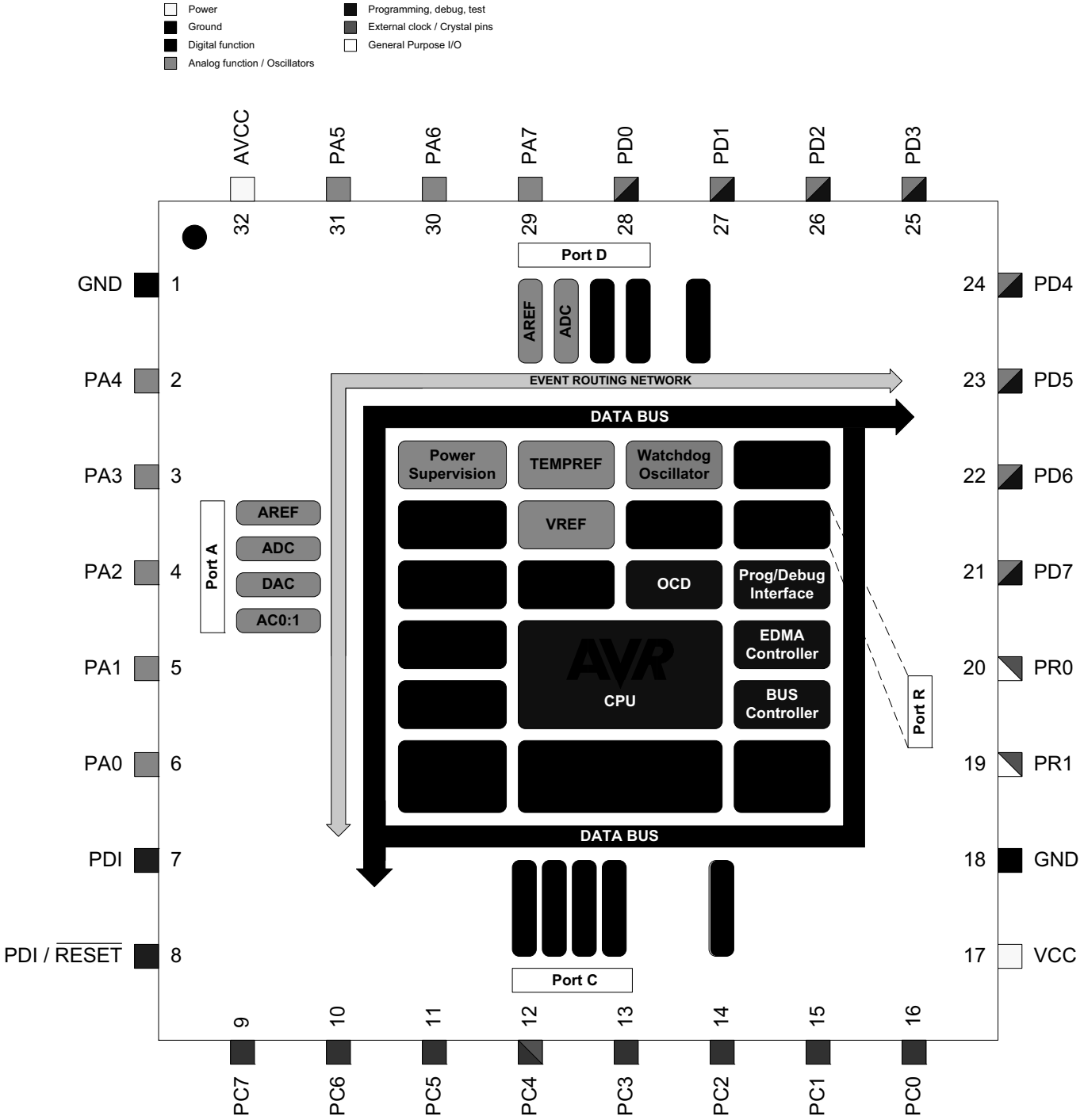
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	AVR
Core Size	8/16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	26
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.6V ~ 3.6V
Data Converters	A/D 16x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-TQFP
Supplier Device Package	32-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atxmega32e5-au

1. Ordering Information

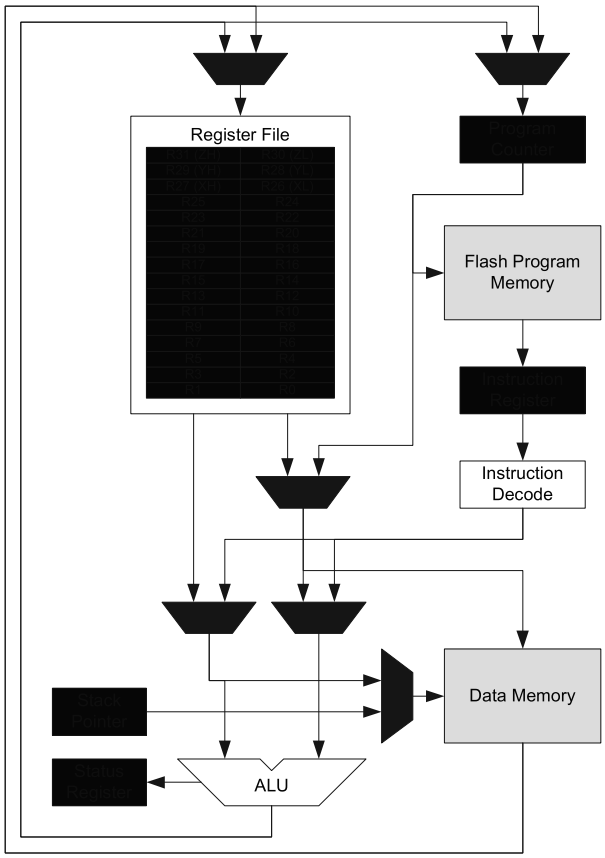
Ordering Code	Package[1][2][3]	Flash [Bytes]	EEPROM [Bytes]	SRAM [Bytes]	Speed [MHz]	Power supply [V]	Temp. [°C]
ATxmega8E5-AU	32A (7x7mm TQFP)	8K + 2K	512	1K	32	1.6 – 3.6	-40 – 85
ATxmega8E5-AUR[4]							
ATxmega8E5-MU	32Z (5x5mm VQFN)						
ATxmega8E5-MUR[4]							
ATxmega8E5-M4U	32MA (4x4mm UQFN)						
ATxmega8E5-M4UR[4]							
ATxmega16E5-AU	32A (7x7mm TQFP)	16K + 4K	512	2K	32	1.6 – 3.6	-40 – 85
ATxmega16E5-AUR[4]							
ATxmega16E5-MU	32Z (5x5mm VQFN)						
ATxmega16E5-MUR[4]							
ATxmega16E5-M4U	32MA (4x4mm UQFN)						
ATxmega16E5-M4UR[4]							
ATxmega32E5-AU	32A (7x7mm TQFP)	32K + 4K	1K	4K	32	1.6 – 3.6	-40 – 85
ATxmega32E5AUR[4]							
ATxmega32E5-MU	32Z (5x5mm VQFN)						
ATxmega32E5-MUR[4]							
ATxmega32E5-M4U	32MA (4x4mm UQFN)						
ATxmega32E5-M4UR[4]							
ATxmega8E5-AN	32A (7x7mm TQFP)	8K + 2K	512	1K	32	1.6 – 3.6	-40 – 105
ATxmega8E5-ANR[4]							
ATxmega8E5-MN	32Z (5x5mm VQFN)						
ATxmega8E5-MNR[4]							
ATxmega8E5-M4UN	32MA (4x4mm UQFN)						
ATxmega8E5-M4UNR[4]							
ATxmega16E5-AN	32A (7x7mm TQFP)	16K + 4K	512	2K	32	1.6 – 3.6	-40 – 105
ATxmega16E5-ANR[4]							
ATxmega16E5-MN	32Z (5x5mm VQFN)						
ATxmega16E5-MNR[4]							
ATxmega16E5-M4UN	32MA (4x4mm UQFN)						
ATxmega16E5-M4UNR[4]							

3. Pinout and Block Diagram



Notes: 1. For full details on pinout and alternate pin functions refer to “Pinout and Pin Functions” on page 57.

Figure 7-1. Block Diagram of the AVR CPU Architecture



The arithmetic logic unit (ALU) supports arithmetic and logic operations between registers or between a constant and a register. Single-register operations can also be executed in the ALU. After an arithmetic operation, the status register is updated to reflect information about the result of the operation.

The ALU is directly connected to the fast-access register file. The 32 x 8-bit general purpose working registers all have single clock cycle access time allowing single-cycle arithmetic logic unit (ALU) operation between registers or between a register and an immediate. Six of the 32 registers can be used as three 16-bit address pointers for program and data space addressing, enabling efficient address calculations.

The memory spaces are linear. The data memory space and the program memory space are two different memory spaces.

The data memory space is divided into I/O registers, SRAM, and memory mapped EEPROM.

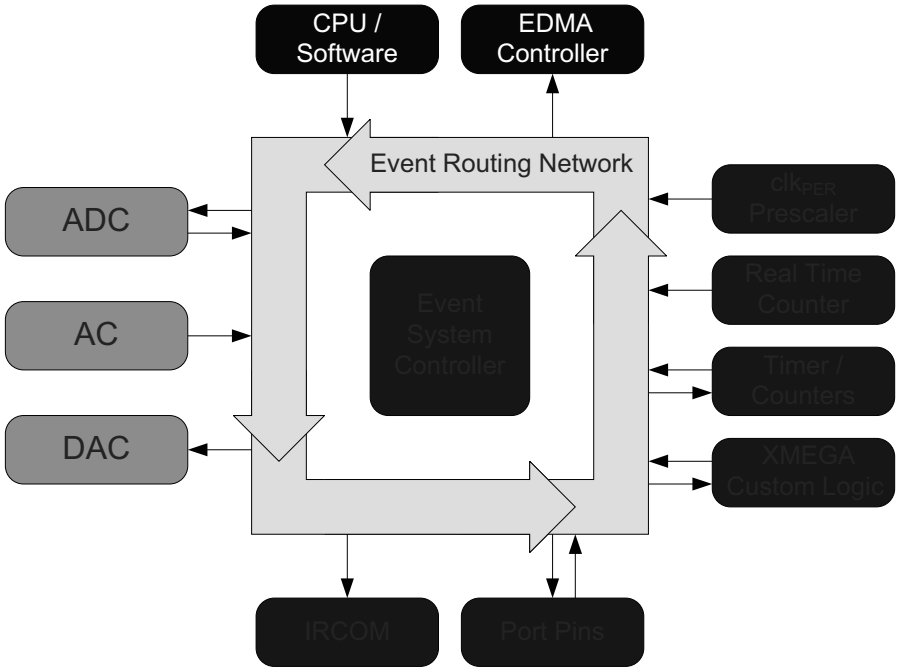
All I/O status and control registers reside in the lowest 4KB addresses of the data memory. This is referred to as the I/O memory space. The lowest 64 addresses can be accessed directly, or as the data space locations from 0x00 to 0x3F. The rest is the extended I/O memory space, ranging from 0x0040 to 0x0FFF. I/O registers here must be accessed as data space locations using load (LD/LDS/LDD) and store (ST/STS/STD) instructions.

The SRAM holds data. Code execution from SRAM is not supported. It can easily be accessed through the five different addressing modes supported in the AVR architecture. The first SRAM address is 0x2000.

Data addresses 0x1000 to 0x1FFF are reserved for EEPROM.

The program memory is divided in two sections, the application program section and the boot program section. Both sections have dedicated lock bits for write and read/write protection. The SPM instruction that is used for self-programming of the application flash memory must reside in the boot program section. The application section contains an application table section with separate lock bits for write and read/write protection. The application table section can be used for save storing of nonvolatile data in the program memory.

Figure 10-1. 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 up to eight parallel event configurations and routing. The maximum routing latency of an external event is two peripheral clock cycles due to re-synchronization, but several peripherals can directly use the asynchronous event without any clock delay. The event system works in all power sleep modes, but only asynchronous events can be routed in sleep modes where the system clock is not available.

12.3.3 Power-save Mode

Power-save mode is identical to power down, with one exception. If the real-time counter (RTC) is enabled, it will keep running during sleep, and the device can also wake up from either an RTC overflow or compare match interrupt. Low power mode option of 8MHz internal oscillator enables instant oscillator wake-up time. This reduces the MCU wake-up time or enables the MCU wake-up from UART bus.

12.3.4 Standby Mode

Standby mode is identical to power down, with the exception that the enabled system clock sources are kept running while the CPU, peripheral, and RTC clocks are stopped. This reduces the wake-up time. The low power option of 8MHz internal oscillator can be enabled to further reduce the power consumption.

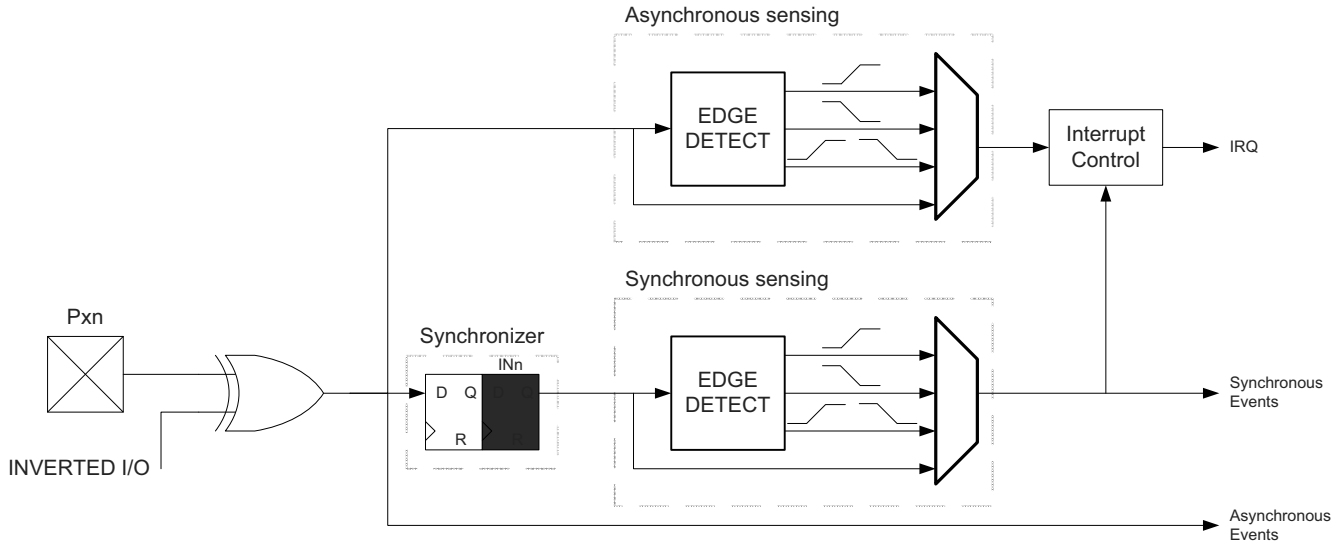
12.3.5 Extended Standby Mode

Extended standby mode is identical to power-save mode, with the exception that the enabled system clock sources are kept running while the CPU and peripheral clocks are stopped. This reduces the wake-up time. The low power option of 8MHz internal oscillator can be enabled to further reduce the power consumption.

16.4 Input Sensing

Input sensing is synchronous or asynchronous depending on the enabled clock for the ports, and the configuration is shown in Figure 16-7.

Figure 16-7. Input Sensing System Overview



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

16.5 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 57 shows which modules on peripherals that enable alternate functions on a pin, and which alternate functions that are available on a pin.

18. WeX – Waveform Extension

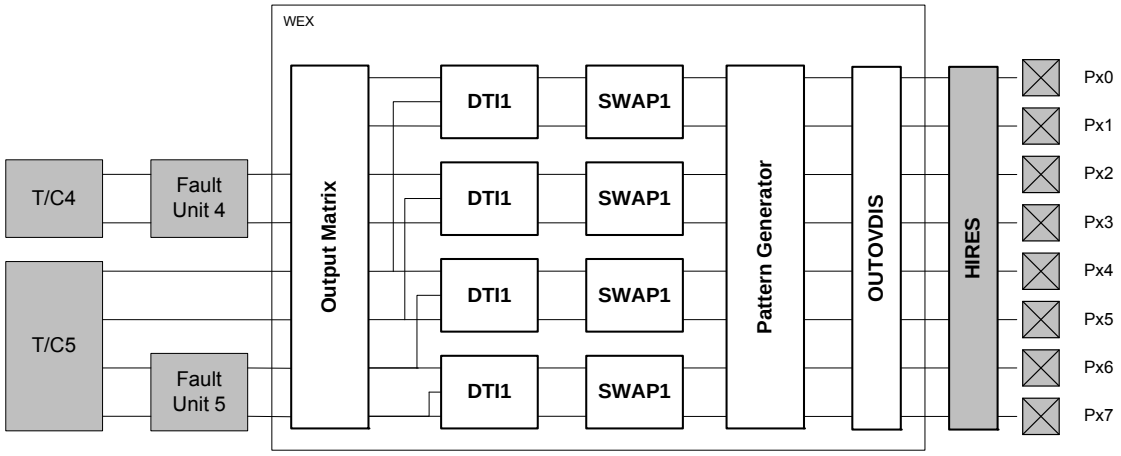
18.1 Features

- Module for more customized and advanced waveform generation
  - Optimized for various type of motor, ballast, and power stage control
- Output matrix for timer/counter waveform output distribution
  - Configurable distribution of compare channel output across port pins
  - Redistribution of dead-time insertion resource between TC4 and TC5
- Four dead-time insertion (DTI) units, each with
  - Complementary high and low side with non overlapping outputs
  - Separate dead-time setting for high and low side
  - 8-bit resolution
- Four swap (SWAP) units
  - Separate port pair or low high side drivers swap
  - Double buffered swap feature
- Pattern generation creating synchronized bit pattern across the port pins
  - Double buffered pattern generation

18.2 Overview

The waveform extension (WEX) provides extra functions to the timer/counter in waveform generation (WG) modes. It is primarily intended for motor control, ballast, LED, H-bridge, power converters, and other types of power control applications. The WEX consist of five independent and successive units, as shown in Figure 18-1.

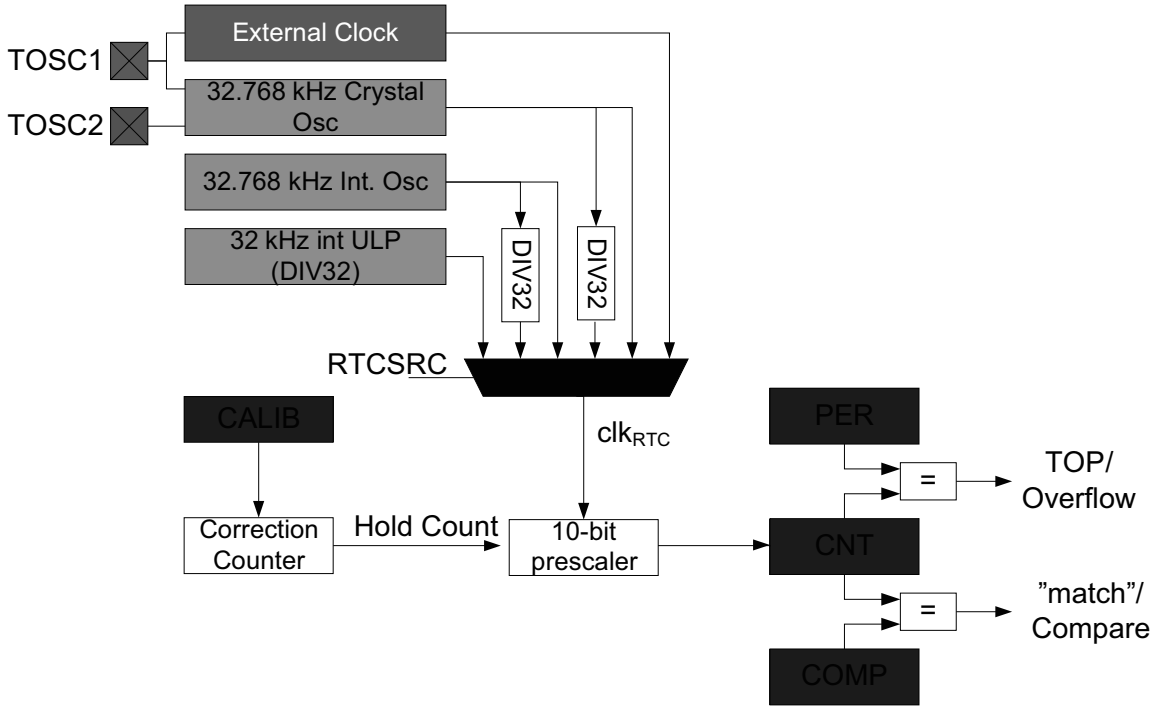
Figure 18-1. Waveform Extension and Closely Related Peripherals



The output matrix (OTMX) can distribute and route out the waveform outputs from timer/counter 4 and 5 across the port pins in different configurations, each optimized for different application types. The dead time insertion (DTI) unit splits the four lower OTMX outputs into a two non-overlapping signals, the non-inverted low side (LS) and inverted high side (HS) of the waveform output with optional dead-time insertion between LS and HS switching.

The swap (SWAP) unit can swap the LS and HS pin position. This can be used for fast decay motor control. The pattern generation unit generates synchronized output waveform with constant logic level. This can be used for easy stepper motor and full bridge control.

Figure 21-1. Real-time Counter Overview



The RTC also supports correction when operated using external 32.768 kHz crystal oscillator. An externally calibrated value will be used for correction. The calibration can be done by measuring the default RTC frequency relative to a more accurate clock input to the device as system clock. The RTC can be calibrated to an accuracy of ±0.5ppm. The RTC correction operation will either speed up (by skipping count) or slow down (adding extra cycles) the prescaler to account for the crystal oscillator error.

22. TWI – Two-Wire Interface

22.1 Features

- One two-wire 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
  - Bridge mode with independent and simultaneous master and slave operation
- 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, 400kHz, and 1MHz 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)
- Supports SMBUS Layer 1 timeouts
- Configurable timeout values
- Independent timeout counters in master and slave (Bridge mode support)

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. One bus can have many slaves and one or several masters that can take control of the bus.

The TWI module supports master and slave functionality. The master and slave functionality are separated from each other, and can be enabled and operate simultaneously and separately. The master module supports multi-master bus operation and arbitration. It contains the baud rate generator. Quick command and smart mode can be enabled to auto-trigger operations and reduce software complexity. The master can support 100kHz, 400kHz, and 1MHz bus frequency.

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. By using the bridge option, the slave can be mapped to different pin locations. The master and slave can support 100kHz, 400kHz, and 1MHz bus frequency.

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.

25. IRCOM – IR Communication Module

25.1 Features

- Pulse modulation/demodulation for infrared communication
- IrDA compatible for baud rates up to 115.2Kbps
- Selectable pulse modulation scheme
  - 3/16 of the baud rate period
  - Fixed pulse period, 8-bit programmable
  - Pulse modulation disabled
- Built-in filtering
- Can be connected to and used by any USART

25.2 Overview

Atmel AVR XMEGA devices contain an infrared communication module (IRCOM) that is IrDA compatible for baud rates up to 115.2Kbps. It can be connected to any USART to enable infrared pulse encoding/decoding for that USART.

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ICALL		Indirect Call to (Z)	PC(15:0) ← Z, PC(21:16) ← 0	None	2 / 3[1]
EICALL		Extended Indirect Call to (Z)	PC(15:0) ← Z, PC(21:16) ← EIND	None	3[1]
CALL	k	call Subroutine	PC ← k	None	3 / 4[1]
RET		Subroutine Return	PC ← STACK	None	4 / 5[1]
RETI		Interrupt Return	PC ← STACK	I	4 / 5[1]
CPSE	Rd,Rr	Compare, Skip if Equal	if (Rd = Rr) PC ← PC + 2 or 3	None	1 / 2 / 3
CP	Rd,Rr	Compare	Rd - Rr	Z,C,N,V,S,H	1
CPC	Rd,Rr	Compare with Carry	Rd - Rr - C	Z,C,N,V,S,H	1
CPI	Rd,K	Compare with Immediate	Rd - K	Z,C,N,V,S,H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if (Rr(b) = 0) PC ← PC + 2 or 3	None	1 / 2 / 3
SBRS	Rr, b	Skip if Bit in Register Set	if (Rr(b) = 1) PC ← PC + 2 or 3	None	1 / 2 / 3
SBIC	A, b	Skip if Bit in I/O Register Cleared	if (I/O(A,b) = 0) PC ← PC + 2 or 3	None	2 / 3 / 4
SBIS	A, b	Skip if Bit in I/O Register Set	If (I/O(A,b) =1) PC ← PC + 2 or 3	None	2 / 3 / 4
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then PC ← PC + k + 1	None	1 / 2
BRBC	s, k	Branch if Status Flag Cleared	if (SREG(s) = 0) then PC ← PC + k + 1	None	1 / 2
BREQ	k	Branch if Equal	if (Z = 1) then PC ← PC + k + 1	None	1 / 2
BRNE	k	Branch if Not Equal	if (Z = 0) then PC ← PC + k + 1	None	1 / 2
BRCS	k	Branch if Carry Set	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRCC	k	Branch if Carry Cleared	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRSH	k	Branch if Same or Higher	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRLO	k	Branch if Lower	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRMI	k	Branch if Minus	if (N = 1) then PC ← PC + k + 1	None	1 / 2
BRPL	k	Branch if Plus	if (N = 0) then PC ← PC + k + 1	None	1 / 2
BRGE	k	Branch if Greater or Equal, Signed	if (N ⊕ V= 0) then PC ← PC + k + 1	None	1 / 2
BRLT	k	Branch if Less Than, Signed	if (N ⊕ V= 1) then PC ← PC + k + 1	None	1 / 2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then PC ← PC + k + 1	None	1 / 2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then PC ← PC + k + 1	None	1 / 2
BRTS	k	Branch if T Flag Set	if (T = 1) then PC ← PC + k + 1	None	1 / 2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then PC ← PC + k + 1	None	1 / 2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then PC ← PC + k + 1	None	1 / 2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then PC ← PC + k + 1	None	1 / 2
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ← PC + k + 1	None	1 / 2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1 / 2
Data transfer instructions					
MOV	Rd, Rr	Copy Register	Rd ← Rr	None	1
MOVW	Rd, Rr	Copy Register Pair	Rd+1:Rd ← Rr+1:Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1

Mnemonics	Operands	Description	Operation	Flags	#Clocks
SPM	Z+	Store Program Memory and Post-Increment by 2	(RAMPZ:Z) ← R1:R0, Z ← Z + 2	None	-
IN	Rd, A	In From I/O Location	Rd ← I/O(A)	None	1
OUT	A, Rr	Out To I/O Location	I/O(A) ← Rr	None	1
PUSH	Rr	Push Register on Stack	STACK ← Rr	None	1(1)
POP	Rd	Pop Register from Stack	Rd ← STACK	None	2(1)
XCH	Z, Rd	Exchange RAM location	Temp ← Rd, Rd ← (Z), (Z) ← Temp	None	2
LAS	Z, Rd	Load and Set RAM location	Temp ← Rd, Rd ← (Z), (Z) ← Temp v (Z)	None	2
LAC	Z, Rd	Load and Clear RAM location	Temp ← Rd, Rd ← (Z), (Z) ← ($FFh – Rd) ● (Z)	None	2
LAT	Z, Rd	Load and Toggle RAM location	Temp ← Rd, Rd ← (Z), (Z) ← Temp ⊕ (Z)	None	2
Bit and bit-test instructions					
LSL	Rd	Logical Shift Left	Rd(n+1) ← Rd(n), Rd(0) ← 0, C ← Rd(7)	Z,C,N,V,H	1
LSR	Rd	Logical Shift Right	Rd(n) ← Rd(n+1), Rd(7) ← 0, C ← Rd(0)	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	Rd(0) ← C, Rd(n+1) ← Rd(n), C ← Rd(7)	Z,C,N,V,H	1
ROR	Rd	Rotate Right Through Carry	Rd(7) ← C, Rd(n) ← Rd(n+1), C ← Rd(0)	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ← Rd(n+1), n=0..6	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	Rd(3..0) ↔ Rd(7..4)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
SBI	A, b	Set Bit in I/O Register	I/O(A, b) ← 1	None	1
CBI	A, b	Clear Bit in I/O Register	I/O(A, b) ← 0	None	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	T	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ← T	None	1
SEC		Set Carry	C ← 1	C	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1

Mnemonics	Operands	Description	Operation	Flags	#Clocks
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
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.

Symbol	Parameter	Condition[2]		Min.	Typ.	Max.	Units
DNL[1]	Differential non-linearity	Differential mode	16ksps, V_{REF} = 3V		1		lsb
			16ksps, V_{REF} = 1V		2		
			300ksps, V_{REF} = 3V		1		
			300ksps, V_{REF} = 1V		2		
		Single ended unsigned mode	16ksps, V_{REF} = 3.0V		1	1.5	
			16ksps, V_{REF} = 1.0V		2	3	
	Offset Error	Differential mode			8		mV
			Temperature drift		0.01		mV/K
			Operating voltage drift		0.25		mV/V
	Gain Error	Differential mode	External reference		-5		mV
			AV_{CC}/1.6		-5		
			AV_{CC}/2.0		-6		
			Bandgap		±10		
			Temperature drift		0.02		mV/K
			Operating voltage drift		2		mV/V
	Gain Error	Single ended unsigned mode	External reference		-8		mV
			AV_{CC}/1.6		-8		
			AV_{CC}/2.0		-8		
			Bandgap		±10		
			Temperature drift		0.03		mV/K
			Operating voltage drift		2		mV/V

Notes: 1. Maximum numbers are based on characterisation and not tested in production, and valid for 10% to 90% input voltage range.
2. Unless otherwise noted all linearity, offset and gain error numbers are valid under the condition that external V_{REF} is used.

Table 36-10. Gain Stage Characteristics

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
R_{in}	Input resistance	Switched		4.0		kΩ
C_{sample}	Input capacitance	Switched		4.4		pF
	Signal range	Gain stage output	0		AV_{CC}- 0.6	V
	Propagation delay	ADC conversion rate	1/2	1	3	Clk_{ADC} cycles
	Clock rate	Same as ADC	100		1800	kHz

36.13.5 Internal Phase Locked Loop (PLL) Characteristics

Table 36-24. Internal PLL Characteristics

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
f_{IN}	Input frequency	Output frequency must be within f_{OUT}	0.4		64	MHz
f_{OUT}	Output frequency (1)	V_{CC}= 1.6 - 1.8V	20		48	
		V_{CC}= 2.7 - 3.6V	20		128	
	Start-up time			25		µs
	Re-lock time			25		

Note: 1. The maximum output frequency vs. supply voltage is linear between 1.8V and 2.7V, and can never be higher than four times the maximum CPU frequency.

36.13.6 External Clock Characteristics

Figure 36-3. External Clock Drive Waveform

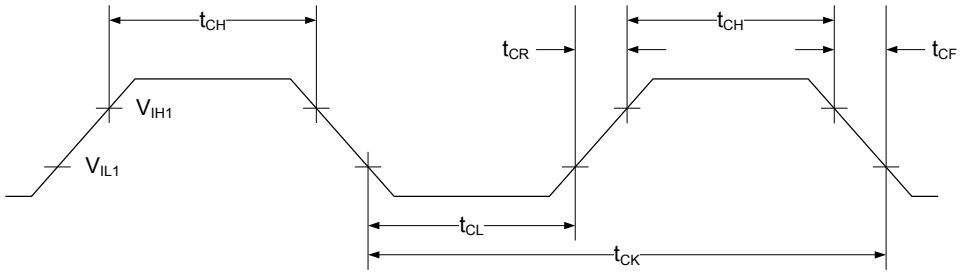


Table 36-25. External Clock used as System Clock without Prescaling

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
$1/t_{CK}$	Clock Frequency (1)	V_{CC} = 1.6 - 1.8V	0		12	MHz
		V_{CC} = 2.7 - 3.6V	0		32	
t_{CK}	Clock Period	V_{CC} = 1.6 - 1.8V	83.3			ns
		V_{CC} = 2.7 - 3.6V	31.5			
t_{CH}	Clock High Time	V_{CC} = 1.6 - 1.8V	30.0			
		V_{CC} = 2.7 - 3.6V	12.5			
t_{CL}	Clock Low Time	V_{CC} = 1.6 - 1.8V	30.0			
		V_{CC} = 2.7 - 3.6V	12.5			
t_{CR}	Rise Time (for maximum frequency)	V_{CC} = 1.6 - 1.8V			10	
		V_{CC} = 2.7 - 3.6V			3	
t_{CF}	Fall Time (for maximum frequency)	V_{CC} = 1.6 - 1.8V			10	
		V_{CC} = 2.7 - 3.6V			3	
Δt_{CK}	Change in period from one clock cycle to the next				10	%

Note: 1. 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.

Figure 37-31.I/O Pin Output Voltage vs. Sink Current

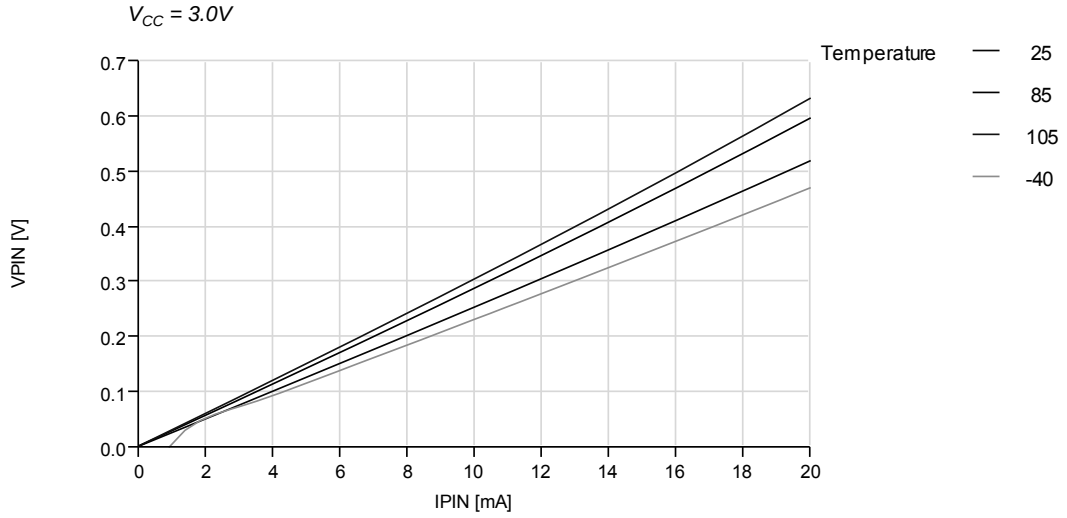


Figure 37-32.I/O Pin Output Voltage vs. Sink Current

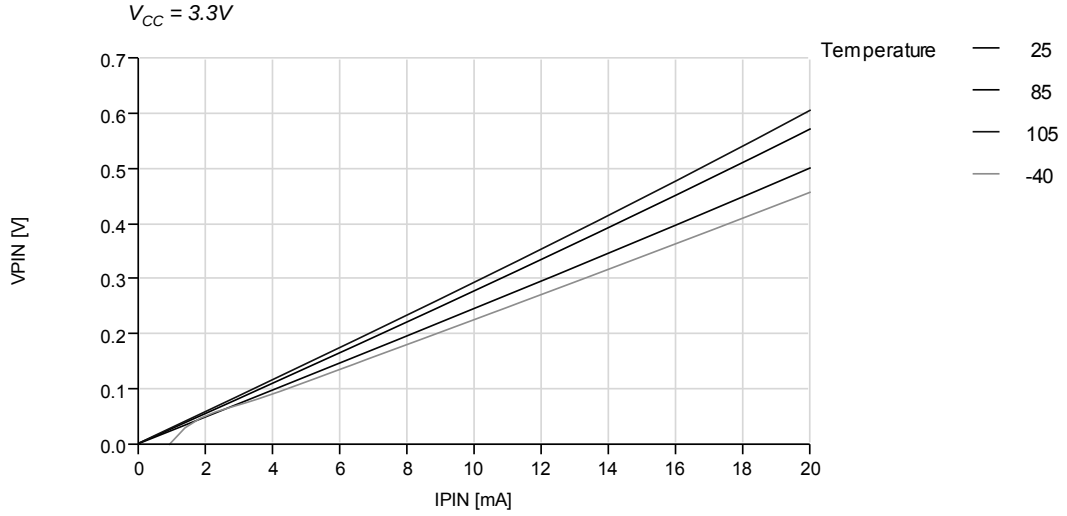


Figure 37-41. ADC Gain Error vs. V_{CC}

T = 25°C, V_{REF} = 1.0V, ADC sample rate = 300ksps

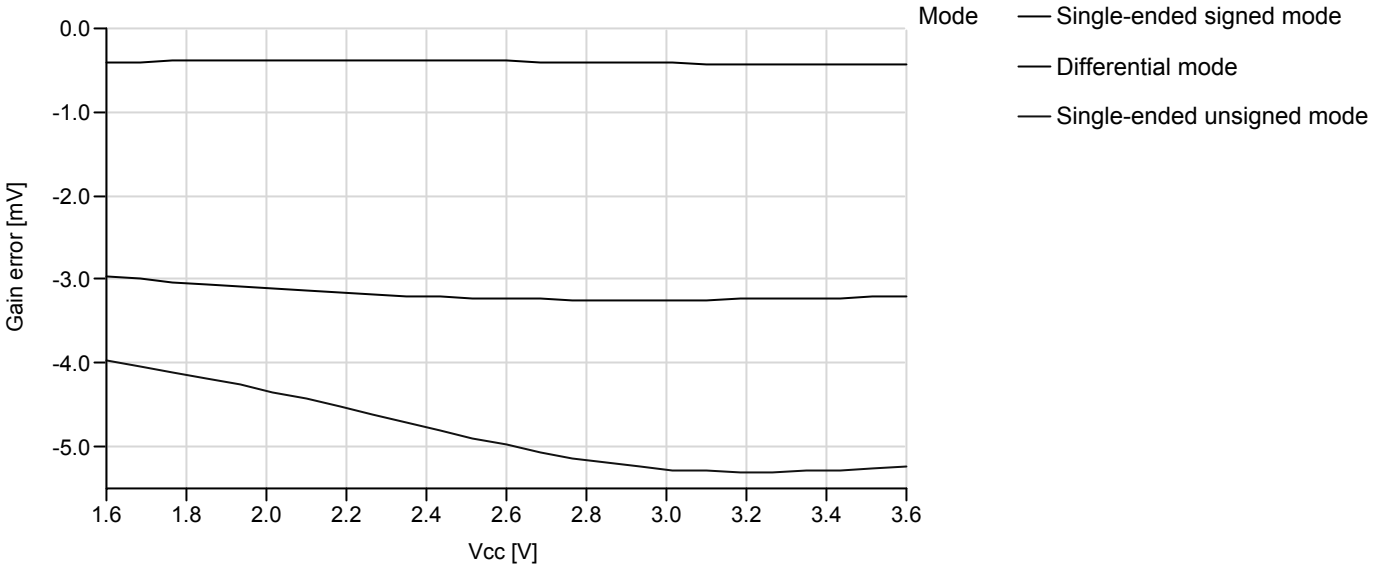


Figure 37-42. ADC Gain Error vs. V_{REF}

T = 25°C, V_{CC} = 3.6V, ADC sample rate = 300ksps

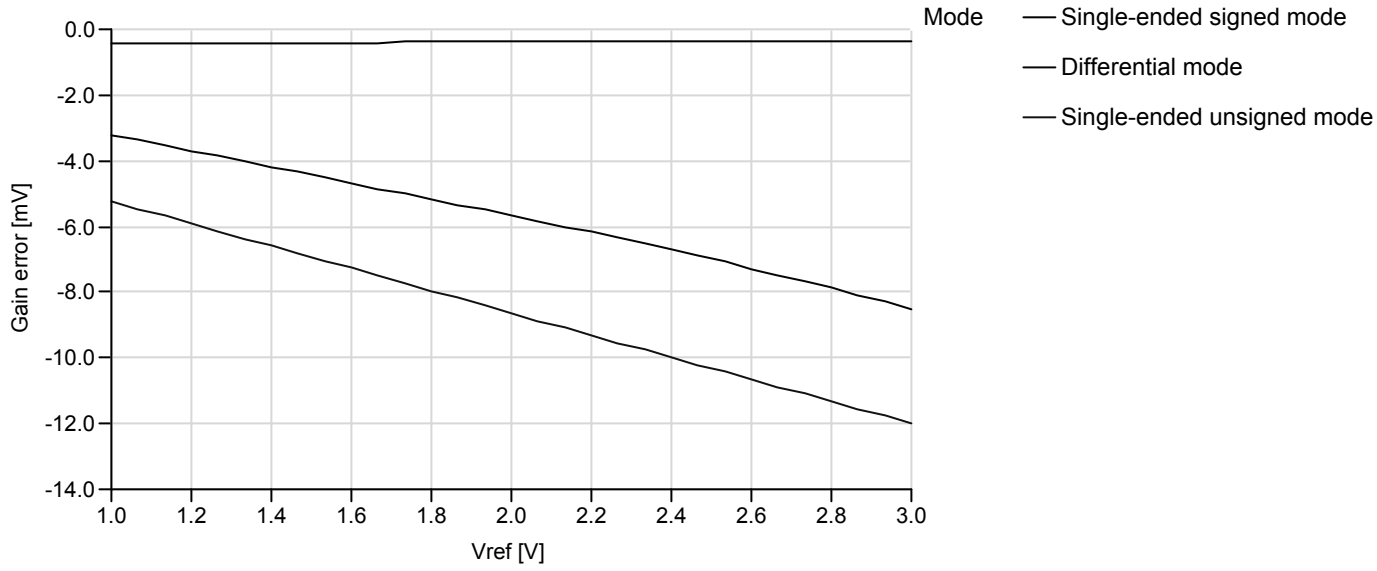


Figure 37-57.Analog Comparator Source vs. Calibration Value

V_{CC} = 3.0V

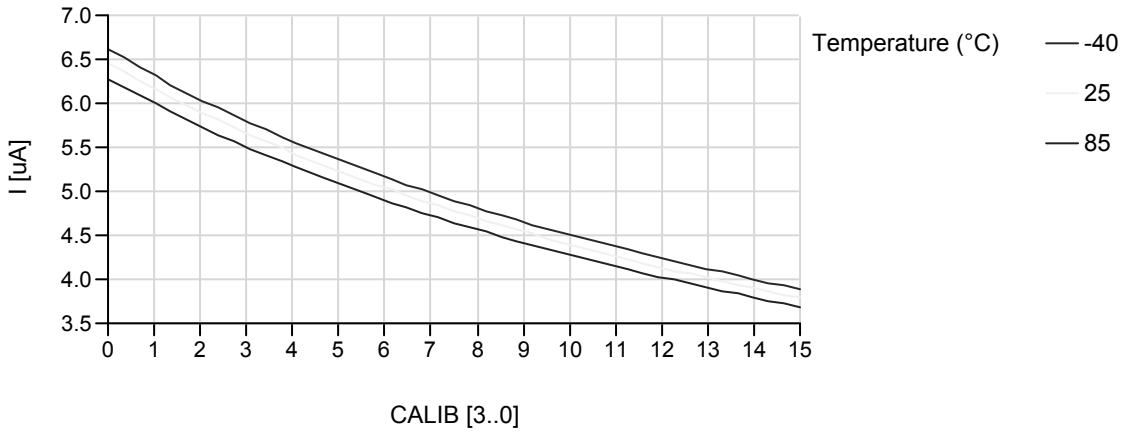
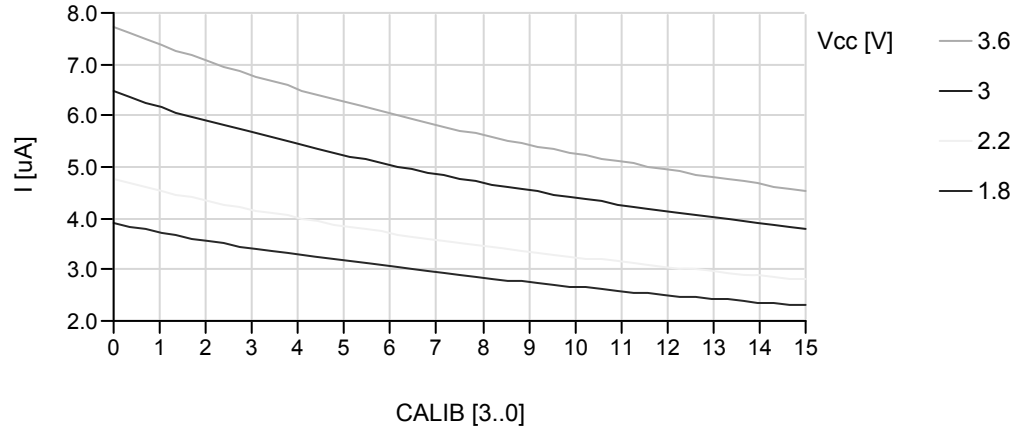


Figure 37-58.Analog Comparator Source vs. Calibration Value

T = 25°C



Issue: TWI SM bus level one Master or slave remembering data

If a write is made to Data register, prior to Address register, the TWI design sends the data as soon as the write to Address register is made. But the send data will be always 0x00.

Workaround:

Since single interrupt line is shared by both timeout interrupt and other TWI interrupt sources, there is a possibility in software that data register will be written after timeout is detected but before timeout interrupt routine is executed. To avoid this, in software, before writing data register, always ensure that timeout status flag is not set.

Issue: Temperature sensor not calibrated

Temperature sensor factory calibration is not implemented on devices before date code 1324.

Workaround:

None.

Issue: Automatic port override on PORT C

When Waveform generation is enabled on PORT C Timers, Automatic port override of peripherals other than Tc may not work even though the pin is not used as waveform output pin.

Workaround:

No workaround.

Issue: Sext timer is not implemented in slave mode

In slave mode, only Ttout timer is implemented. Sext timer is needed in slave mode to release the SCL line and to allow the master to send a STOP condition. If only master implements Sext timer, slave continues to stretch the SCL line (up to the Ttout timeout in the worse case). Sext = Slave cumulative timeout.

Workaround:

No workaround.