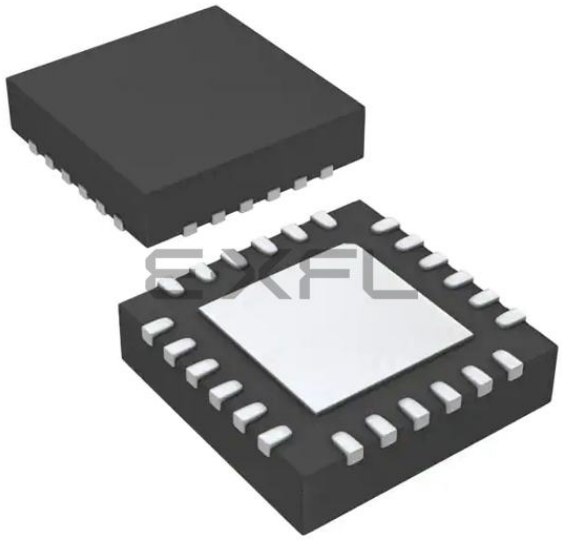


Welcome to E-XFL.COM

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-Bit
Speed	20MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	22
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 12x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	24-VFQFN Exposed Pad
Supplier Device Package	24-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/attiny817-mnr

2. Ordering Information

2.1 ATtiny417

Table 2-1. ATtiny417 Ordering Codes

Ordering Code[1]	Flash	Package Type (GPC)	Leads	Power Supply	Operational Range	Carrier Type
ATtiny417-MNR	4KB	VQFN 4x4 (ZHA)	24	1.8V - 5.5V	Industrial (-40°C +105°C)	Tape & Reel
ATtiny417-MFR	4KB	VQFN 4x4 (ZHA)	24	1.8V - 5.5V	Industrial (-40°C +125°C)	Tape & Reel

1. Pb-free packaging complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.

2.2 ATtiny817

Table 2-2. ATtiny817 Ordering Codes

Ordering Code[1]	Flash	Package Type (GPC)	Leads	Power Supply	Operational Range	Carrier Type
ATtiny817-MNR	8KB	VQFN 4x4 (ZHA)	24	1.8V - 5.5V	Industrial (-40°C +105°C)	Tape & Reel
ATtiny817-MFR	8KB	VQFN 4x4 (ZHA)	24	1.8V - 5.5V	Industrial (-40°C +125°C)	Tape & Reel

Note:

1. Pb-free packaging complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.

6. Memories

6.1 Overview

The main memories are SRAM data memory, EEPROM data memory, and Flash program memory. In addition, the peripheral registers are located in the I/O memory space.

Table 6-1. Physical Properties of Flash Memory

Property	ATtiny417	ATtiny817
Size	4KB	8KB
Page size	64B	64B
Number of pages	64	128
Start address	0x8000	0x8000

Table 6-2. Physical Properties of SRAM

Property	ATtiny417	ATtiny817
Size	256B	512B
Start address	0x3F00	0x3E00

Table 6-3. Physical Properties of EEPROM

Property	ATtiny417	ATtiny817
Size	128B	128B
Page size	32B	32B
Number of pages	4	4
Start address	0x1400	0x1400

Related Links

I/O Memory

6.2 Memory Map

6.3 In-System Reprogrammable Flash Program Memory

The ATtiny417/817 contains 4/8KB On-Chip In-System Reprogrammable Flash memory for program storage. Since all AVR instructions are 16 or 32 bits wide, the Flash is organized as 4K x 16. For write protection, the Flash Program memory space can be divided into three sections: Boot Loader section, Application code section and Application data section, with restricted access rights among them.

The program counter is 11/12 bits wide to address the whole program memory. The procedure for writing Flash memory is described in detail in the documentation of the Non-Volatile Memory Controller (NVMCTRL) peripheral.

6.9.3 Fuse Summary - FUSE

Offset	Name	Bit Pos.								
0x00	WDTCFG	7:0	WINDOW[3:0]				PERIOD[3:0]			
0x01	BODCFG	7:0	LVL[2:0]			SAMPFREQ	ACTIVE[1:0]		SLEEP[1:0]	
0x02	OSCCFG	7:0	OSCLOCK						FREQSEL[1:0]	
0x03	Reserved									
0x04	TCD0CFG	7:0	CMPDEN	CMPCEN	CMPBEN	CMPAEN	CMPD	CMPC	CMPB	CMPA
0x05	SYSCFG0	7:0	CRCSRC[1:0]				RSTPINCFG[1:0]			EESAVE
0x06	SYSCFG1	7:0						SUT[2:0]		
0x07	APPEND	7:0	APPEND[7:0]							
0x08	BOOTEND	7:0	BOOTEND[7:0]							
0x09	Reserved									
0x0A	LOCKBIT	7:0	LOCKBIT[7:0]							

6.9.4 Fuse Description

6.9.4.1 Watchdog Configuration

Name: WDTCFG
Offset: 0x00
Reset: -
Property: -

Bit	7	6	5	4	3	2	1	0
	WINDOW[3:0]				PERIOD[3:0]			
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 7:4 – WINDOW[3:0]: Watchdog Window Timeout Period
This value is loaded into the WINDOW bit field of the Watchdog Control A register (WDT.CTRLA) during Reset.

Bits 3:0 – PERIOD[3:0]: Watchdog Timeout Period
This value is loaded into the PERIOD bit field of the Watchdog Control A register (WDT.CTRLA) during Reset.

Related Links
Register Summary - WDT
RSTCTRL - Reset Controller

6.9.4.2 BOD Configuration

The settings of the BOD will be reloaded from this Fuse after a Power-On Reset. For all other Resets, the BOD configuration remains unchanged.

Bit	7	6	5	4	3	2	1	0
	CMPDEN	CMPCEN	CMPBEN	CMPAEN	CMPD	CMPC	CMPB	CMPA
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 4, 5, 6, 7 – CMPAEN, CMPBEN, CMPCEN, CMPDEN: Compare x Enable

Value	Description
0	Compare x output on Pin is disabled
1	Compare x output on Pin is enabled

Bits 0, 1, 2, 3 – CMPA, CMPB, CMPC, CMPD: Compare x

This bit selects the default state of Compare x after Reset, or when entering debug if FAULTDET is '1'.

Value	Description
0	Compare x default state is 0
1	Compare x default state is 1

Related Links

Register Summary - TCD

RSTCTRL - Reset Controller

6.9.4.5 System Configuration 0

Name: SYSCFG0
Offset: 0x05
Reset: 0xC4
Property: -

Bit	7	6	5	4	3	2	1	0
	CRCSRC[1:0]				RSTPINCFG[1:0]			EESAVE
Access	R	R			R	R		R
Reset	1	1			0	1		0

Bits 7:6 – CRCSRC[1:0]: CRC Source

See CRC description for more information about the functionality.

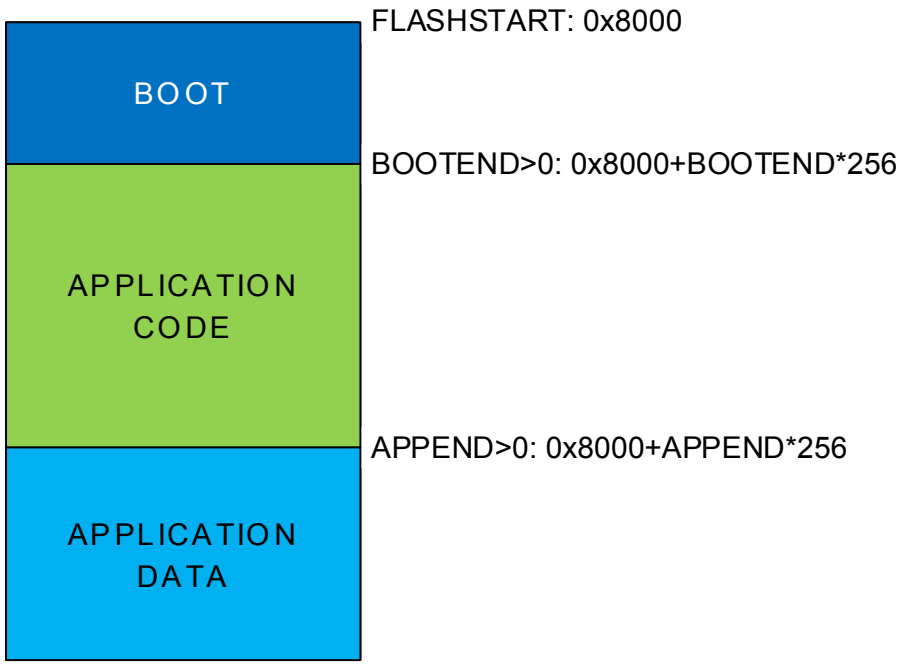
Value	Name	Description
00	FLASH	CRC of full Flash (boot, application code and application data)
01	BOOT	CRC of boot section
10	BOOTAPP	CRC of application code and boot sections
11	NOCRC	No CRC

Bits 3:2 – RSTPINCFG[1:0]: Reset Pin Configuration

These bits select the Reset/UPDI pin configuration.

Value	Description
0x0	GPIO
0x1	UPDI
0x2	RESET

Figure 9-2. Flash Sections



Section Sizes

The sizes of these sections are set by the Boot Section End fuse (FUSE.BOOTEND) and Application Code Section End fuse (FUSE.APPEND).

The fuses select the section sizes in blocks of 256 bytes. As shown in Figure 9-2, the BOOT section stretches from the start of the Flash until BOOTEND. The APPCODE section runs from BOOTEND until APPEND. The remaining area is the APPDATA section. If APPEND is written to 0, the APPCODE section runs from BOOTEND to the end of Flash (removing the APPDATA section). If BOOTEND and APPEND are written to 0, the entire Flash is regarded as BOOT section. APPEND should either be set to 0 or a value greater or equal than BOOTEND.

Table 9-2. Setting up Flash sections

BOOTEND	APPEND	BOOT section	APPCODE section	APPDATA section
0	0	0 to FLASHEND	-	-
> 0	0	0 to 256*BOOTEND	256*BOOTEND to FLASHEND	-
> 0	== BOOTEND	0 to 256*BOOTEND	-	256*BOOTEND to FLASHEND
> 0	> BOOTEND	0 to 256*BOOTEND	256*BOOTEND to 256*APPEND	256*APPEND to FLASHEND

Note:

- Also see BOOTEND and APPEND descriptions.
- Interrupt vectors are by default located after the BOOT section. This can be changed in the interrupt controller. Refer to Interrupt Vector Locations

or event clocking for TCA configured by EVACT in TCA.EVCTRL. Event clocking from TCA0 is only used if enable count on event (CNTEI) is set to '1' in TCA.EVCTRL.

When the Synchronize Update bit (SYNCUPD) in the Control A register (TCB.CTRLA) is written to '1', the TCB counter will restart when the TCA0 counter restarts.

Related Links

Block Diagram

21.3.4 Events

If TCB is connected to the Event System as an event generator, the TCB will generate a strobe on the connected event channel when the configured event condition is met.

The peripheral accepts one event input. If the Capture Event Input Enable bit (CAPTEI) in the Event Control register (TCB.EVCTRL) is written to '1', incoming events will result in an event action as defined by the Event Edge bit (EDGE) in TCB.EVCTRL. A change on event line needs to be held for at least on system clock cycle to guarantee action on event except single shot mode.

Related Links

EVCTRL

EVSYS - Event System

21.3.5 Interrupts

Table 21-4. Available Interrupt Vectors and Sources

Offset	Name	Vector Description	Conditions
0x00	CAPT	TCB interrupt	Depending on operating mode. See description of CAPT in TCB.INTFLAG.

When an interrupt condition occurs, the corresponding Interrupt Flag is set in the Interrupt Flags register of the peripheral (*peripheral*.INTFLAGS).

An interrupt source is enabled or disabled by writing to the corresponding enable bit in the peripheral's Interrupt Control register (*peripheral*.INTCTRL).

An interrupt request is generated when the corresponding interrupt source is enabled and the Interrupt Flag is set. The interrupt request remains active until the Interrupt Flag is cleared. See the peripheral's INTFLAGS register for details on how to clear Interrupt Flags.

Related Links

CPUINT - CPU Interrupt Controller

INTFLAGS

21.3.6 Sleep Mode Operation

TCB will halt operation in the Power Down Sleep mode. Standby sleep operation is dependent on the Run in Standby bit (RUNSTDBY) in the Control A register (TCB.CTRLA).

21.3.7 Synchronization

Not applicable.

21.3.8 Configuration Change Protection

Not applicable.

23. RTC - Real Time Counter

23.1 Features

- 16-bit resolution
- Selectable clock source
  - 32.768kHz external crystal (XOSC32K)
  - External clock
  - 32KHz internal ULP oscillator (OSCULP32K)
  - OSCULP32K divided by 32
- Programmable 15-bit clock prescaling
- One compare register
- One period register
- Clear timer on period overflow
- Optional interrupt/Event on overflow and compare match
- Periodic interrupt and Event

23.2 Overview

The RTC peripheral offers two timing functions: the Real-Time Counter (RTC) and a Periodic Interrupt Timer (PIT).

The PIT functionality can be enabled independent of the RTC functionality.

RTC - Real-Time Counter

The RTC counts (prescaled) clock cycles in a Counter register, and compares the content of the Counter register to a Period register and a Compare register.

The RTC can generate both interrupts and Events on compare match or overflow. It will generate a compare interrupt and/or Event at the first count after the counter equals the Compare register value, and an overflow interrupt and/or Event at the first count after the counter value equals the Period register value. The overflow will also reset the Counter value to zero.

The RTC peripheral typically runs continuously, including in low-power sleep modes, to keep track of time. It can wake up the device from sleep modes and/or interrupt the device at regular intervals.

The reference clock is typically the 32.768kHz output from an external crystal. The RTC can also be clocked from an external clock signal, the 32KHz internal Ultra-Low Power oscillator (OSCULP32K), or the OSCULP32K divided by 32.

The RTC peripheral includes a 15-bit programmable prescaler that can scale down the reference clock before it reaches the counter. A wide range of resolutions and time-out periods can be configured for the RTC. With a 32.768kHz clock source, the maximum resolution is 30.5µs, and time-out periods can be up to 2 seconds. With a resolution of 1s, the maximum time-out period is more than 18 hours (65536 seconds). The RTC can give a compare interrupt and/or Event when the counter equals the compare register value, and an overflow interrupt and/or Event when it equals the period register value.

PIT - Periodic Interrupt Timer

Using the same clock source as the RTC function, the PIT can request an interrupt or trigger an output Event on every *n*-th clock period. *n* can be selected from {4, 8, 16,.. 32768} for interrupts, and from {64, 128, 256,... 8192} for Events.

The PIT uses the same clock source (CLK_RTC) as the RTC function .

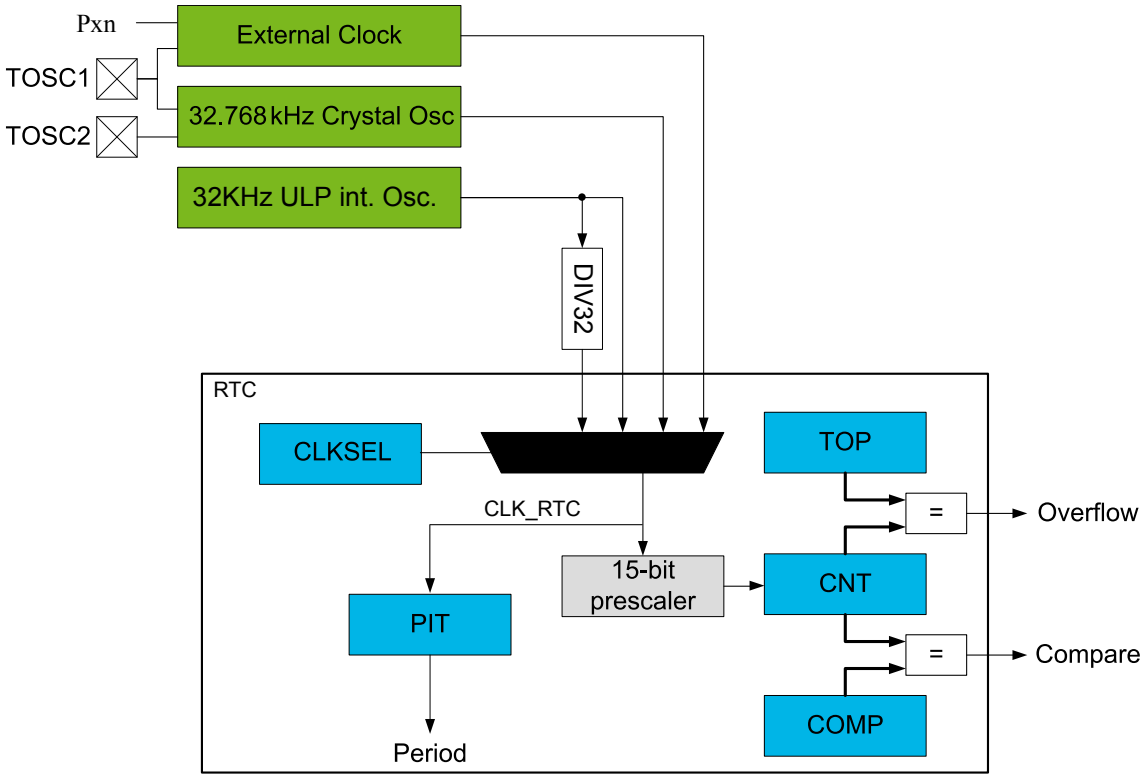
Related Links

RTC Functional Description

PIT Functional Description

23.2.1 Block Diagram

Figure 23-1. Block Diagram



23.2.2 Signal Description

Not applicable.

23.2.3 System Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

Table 23-1. RTC System Dependencies

Dependency	Applicable	Peripheral
Clocks	Yes	CLKCTRL
I/O Lines and Connections	Yes	PORT
Interrupts	Yes	CPUINT

- All LD instructions (LD, LDS, LDD with X/Y/Z pointer)
- POP
- RET/RETI
- CALL/JMP (not RJMP/RCALL/IJMP/ICALL)
- All MUL instructions (MUL, MULS, MULSU, FMUL...)
- ADIW/SBIW

If the cache is enabled, the CRC will get access as long as the CPU is executing code from a cached loop, with the same timing assumptions as for priority mode.

27.3.3 Interrupts

Table 27-3. Available Interrupt Vectors and Sources

Offset	Name	Vector Description	Conditions
0x00	NMI	Non-Maskable Interrupt	Generated on CRC failure

When the interrupt condition occurs, the OK flag in the Status register (CRCSCAN.STATUS) is cleared to '0'.

An interrupt is enabled by writing a '1' to the respective Enable bit (NMIEN) in the Control A register (CRCSCAN.CTRLA), but can only be disabled with a system Reset. An NMI is generated when the OK flag in CRCSCAN.STATUS is cleared and the NMIEN bit is '1'. The NMI request remains active until a system Reset, and can not be disabled.

A non-maskable interrupt can be triggered even if interrupts are not globally enabled.

Related Links

CTRLA

STATUS

CPUINT - CPU Interrupt Controller

27.3.4 Sleep Mode Operation

In all CPU sleep modes, the CRCSCAN peripheral is halted, and will resume operation when the CPU wakes up.

It is possible to enter sleep mode after setting up the CRCSCAN to start a PRIORITY check (see CTRLB for more information), but before the actual check is started. If the CPU is able enter sleep mode before the check starts and a Priority check was configured, the check will start and complete (halting the CPU) immediately after waking up, and before entering any interrupt handler.

27.3.5 Configuration Change Protection

Not applicable.

Gated D-Latch (DLATCH)

The D-input is driven by the even LUT output (LUT0), and the G-input is driven by the odd LUT output (LUT1).

Figure 28-9. D-Latch

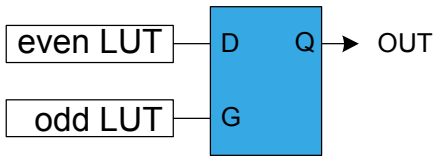


Table 28-5. D-Latch Characteristics

G	D	OUT
0	X	Hold state (no change)
1	0	Clear
1	1	Set

RS Latch (RS)

The S-input is driven by the even LUT output (LUT0), and the R-input is driven by the odd LUT output (LUT1).

Figure 28-10. RS-Latch

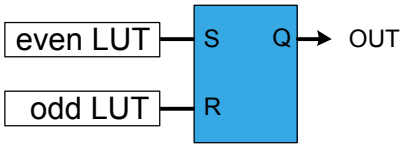


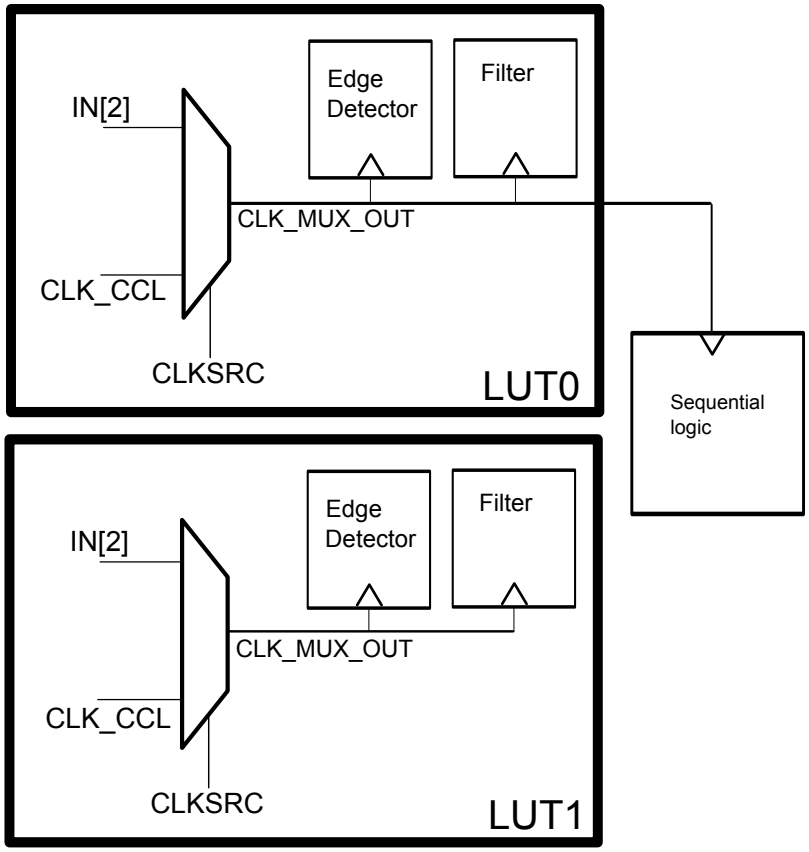
Table 28-6. RS-latch Characteristics

S	R	OUT
0	0	Hold state (no change)
0	1	Clear
1	0	Set
1	1	Forbidden state

28.3.2.7 Clock Source Settings

The Filter, Edge Detector and Sequential logic are by default clocked by the system clock (CLK_PER). It is also possible to use the LUT input 2 (IN[2]) to clock these blocks (CLK_MUX_OUT in figure Figure 28-11). This is configured by writing the Clock Source bit (CLKSRC) in the LUT Control A register (CCL.LUTnCTRLA) to '1'.

Figure 28-11. Clock Source Settings



When the Clock Source bit (CLKSRC) is '1', IN[2] is used to clock the corresponding Filter and Edge Detector (CLK_MUX_OUT). The Sequential logic is clocked by CLK_MUX_OUT of the even LUT in the pair. When CLKSRC bit is '1', IN[2] is treated as MASKed (low) in the TRUTH table.

The CCL peripheral must be disabled while changing the clock source to avoid undetermined outputs from the peripheral.

28.3.3 Events

The CCL can generate the following output events:

- LUTnOUT: Lookup Table Output Value

The CCL can take the following actions on an input event:

- INx: The event is used as input for the TRUTH table.

Related Links

EVSYS - Event System

28.3.4 Sleep Mode Operation

Writing the Run In Standby bit (RUNSTDBY) in the Control A register (CCL.CTRLA) to '1' will allow the system clock to be enabled in Standby sleep mode.

If RUNSTDBY is '0' the system clock will be disabled in Standby sleep mode. If the Filter, Edge Detector or Sequential logic are enabled, the LUT output will be forced to zero in Standby sleep mode. In Idle sleep mode, the TRUTH table decoder will continue operation and the LUT output will be refreshed accordingly, regardless of the RUNSTDBY bit.

The RESRDY interrupt flag in ADC.INTFLAG will be set even if the specific interrupt is disabled, allowing software to check for finished conversion by polling the flag. A conversion can thus be triggered without causing an interrupt.

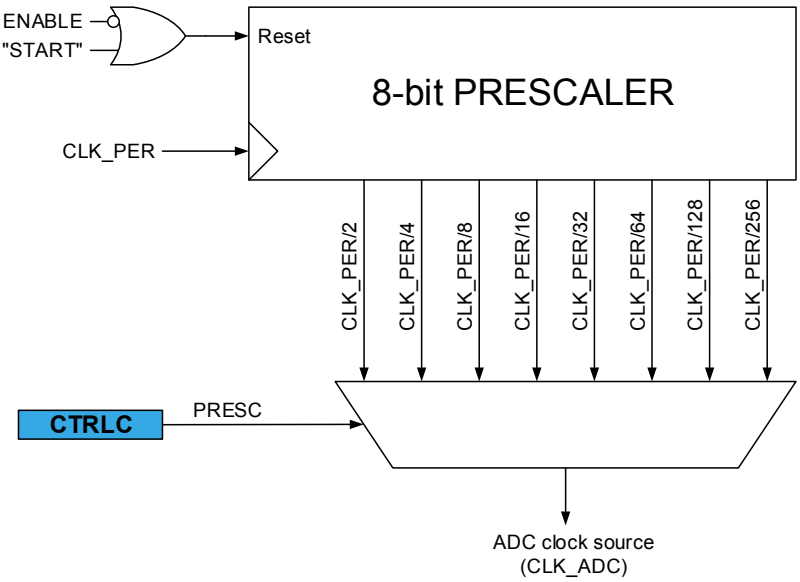
Alternatively, a conversion can be triggered by an event, this is enabled by writing a '1' to the Start Event Input bit (STARTEI) in the Event Control register (ADC.EVCTRL). Any incoming event routed to the ADC thorugh the Event System (EVSYS) will trigger an ADC conversion. This provides a method to start conversions at predictable intervals or at specific conditions.

The event trigger input is edge sensitive, when an event occurs, STCONV in ADC.COMMAND is set. STCONV will be cleared when the conversion is complete.

In Free Running mode, the first conversion is started by writing the STCONV bit to '1' in ADC.COMMAND. A new conversion cycle is started immediately after the previous conversion cycle has completed. A conversion complete will set the RESRDY flag in ADC.INTFLAGS.

30.3.2.2 Clock generation

Figure 30-6. ADC Prescaler



The ADC requires an input clock frequency between 50kHz and 1.5MHz for maximum resolution. If a lower resolution than 10 bits is selected, the input clock frequency to the ADC can be higher than 1.5MHz to get a higher sample rate.

The ADC module contains a prescaler which generates the ADC clock (CLK_ADC) from any CPU clock (CLK_PER) above 100kHz. The prescaling is selected by writing to the Prescaler bits (PRESC) in the Control C register (ADC.CTRLC). The prescaler starts counting from the moment the ADC is switched on by writing a '1' to the ENABLE bit in ADC.CTRLA. The prescaler keeps running as long as the ENABLE bit is one, the prescaler counter is reset to zero when the ENABLE bit is zero.

When initiating a conversion by writing a '1' to the Start Conversion bit (STCONV) in the Command register (ADC.COMMAND) or from event, the conversion starts at the following rising edge of the CLK_ADC clock cycle. The prescaler is kept reset as long as there is no ongoing conversion. This assures a fixed delay from the trigger to the actual start of a conversion in CLK_PER cycles as:

$$\text{StartDelay} = \frac{\text{PRESC}_{\text{factor}}}{2} + 2$$

30.3.3 Events

An ADC conversion can be triggered automatically by an event input if the Start Event Input bit (STARTEI) in the Event Control register (ADC.EVCTRL) is written to '1'.

See also the description of the Asynchronous User Channel n Input Selection in the Event System (EVSYS.ASYNCUSERn).

Related Links

ASYNCUSER0, ASYNCUSER1, ASYNCUSER2, ASYNCUSER3, ASYNCUSER4, ASYNCUSER5, ASYNCUSER6, ASYNCUSER7, ASYNCUSER8, ASYNCUSER9, ASYNCUSER10

30.3.4 Interrupts

Table 30-2. Available Interrupt Vectors and Sources

Offset	Name	Vector Description	Conditions
0x00	RESRDY	Result Ready interrupt	The conversion result is available in the Result register (ADC.RES).
0x02	WCOMP	Window Comparator interrupt	As defined by WINCM in ADC.CTRLE.

When an interrupt condition occurs, the corresponding Interrupt Flag is set in the Interrupt Flags register of the peripheral (*peripheral*.INTFLAGS).

An interrupt source is enabled or disabled by writing to the corresponding enable bit in the peripheral's Interrupt Control register (*peripheral*.INTCTRL).

An interrupt request is generated when the corresponding interrupt source is enabled and the Interrupt Flag is set. The interrupt request remains active until the Interrupt Flag is cleared. See the peripheral's INTFLAGS register for details on how to clear Interrupt Flags.

30.3.5 Sleep Mode Operation

The ADC is by default disabled In *Standby sleep mode*.

The ADC can stay fully operational in Standby sleep mode if the Run in Standby bit (RUNSTDBY) in the Control A register (ADC.CTRLA) is written to '1'.

When the device is entering Standby sleep mode when RUNSTDBY is one, the ADC will stay active, hence any ongoing conversions will be completed, and interrupts will be executed as configured.

In Standby sleep mode an ADC conversion must be triggered via the Event System (EVSYS), or the ADC must be in free-running mode with the first conversion triggered by software before entering sleep. The peripheral clock is requested if needed and is turned off after the conversion is completed.

When an input Event trigger occurs, the positive edge will be detected, the Start Conversion bit (STCONV) in the Command register (ADC.COMMAND) set, and the conversion will start. When the conversion is completed, the Result Ready Flag (RESRDY) in the Interrupt Flags register (ADC.INTFLAGS) is set and the STCONV bit in ADC.COMMAND is cleared.

The reference source and supply infrastructure need time to stabilize when activated in Standby sleep mode. Configure a delay for the start of the first conversion by writing a non-zero value to the Initial Delay bits (INITDLY) in the Control D register (ADC.CTRLD).

In *Power Down sleep mode*, no conversions are possible. Any ongoing conversions are halted and will be resumed when going out of sleep. At end of conversion, the Result Ready Flag (RESRDY) will be set, but the content of the result registers (ADC.RES) is invalid since the ADC was halted in the middle of a conversion.

1. Enter the USERROW-Write KEY located in Table 33-6 by using the `KEY` instruction. See Table 33-6 for the UROWWRITE signature.
2. **Optional:** Read the UROWWRITE bit field in UPDI.ASI_KEY_STATUS to see that the KEY has been activated.
3. Write the Reset signature into the UPDI.ASI_RESET_REQ register. This will issue a System Reset.
4. Write 0x00 to the Reset signature in UPDI.ASI_RESET_REQ register to clear the System Reset.
5. Read UROWPROG bit in UPDI.ASI_SYS_STATUS.
6. User Row Programming can start when UROWPROG == 1. If UROWPROG == 0, go to point 5 again.
7. The writable area has a size of one EEPROM page, 32 byte, and it is only possible to write User Row data to the first 32 byte addresses of the RAM. Addressing outside this memory range will result in a non executed write. The data will map 1:1 with the User Row space, when the data is copied into the User Row upon completion of the Programming sequence.
8. When all User Row data has been written to the RAM, write the UROWWRITEFINAL bit in UPDI.ASI_SYS_CTRLA.
9. Read the UROWPROG bit in UPDI.ASI_SYS_STATUS.
10. The User Row Programming is completed when UROWPROG == 0. If UROWPROG == 1, go to point 9 again.
11. Write the UROWWRITE bit in UPDI.ASI_KEY_STATUS.
12. Write the Reset signature into the UPDI.ASI_RESET_REQ register. This will issue a System Reset.
13. Write 0x00 to the Reset signature in UPDI.ASI_RESET_REQ register to clear the System Reset.
14. User Row Programming is complete.

It is not possible to read back data from the SRAM in this mode. Only writes to the first 32 bytes of the SRAM is allowed.

33.3.8 Events

The UPDI is connected to the Event System (EVSYS) as described in ASYNCCH0, ASYNCCH1, ASYNCCH2, ASYNCCH3.

The UPDI can generate the following output events:

- SYNCH Character Positive Edge Event

This Event is set on the UPDI clock for each detected positive edge in the SYNCH character, and it is not possible to disable this event from the UPDI. The recommended application for this Event is system clock frequency measurement through the UPDI. Section System Clock Measurement with UPDI provides the details on how to setup the system for this operation.

Related Links

EVSYS - Event System

33.3.9 Sleep Mode Operation

The UPDI physical layer runs independently of all sleep modes, and the UPDI is always accessible for a connected debugger independent of the device sleep mode. If the system enters a sleep mode that turns off the CPU clock, the UPDI will not be able to access the system bus and read memories and peripherals. The UPDI physical layer clock is unaffected by the sleep mode settings, as long as the UPDI is enabled. By reading the INSLEEP bit in UPDI.ASI_SYS_STATUS it is possible to monitor if the system domain is in sleep mode. The INSLEEP bit is set if the system is in IDLE sleep mode or deeper.

It is possible to prevent the system clock from stopping when going into sleep mode, by writing the CLKREQ bit in UPDI.ASI_SYS_CTRL to '1'. If this bit is set the system sleep mode state is emulated, and

Table 34-13. 32.768kHz Internal Oscillator (OSCULP32K) Characteristics

Symbol	Description	Condition	Condition	Min.	Typ.	Max.	Unit
$f_{OSCULP32K}$	Accuracy	Factory calibrated	T=25°C, 3.0V	-3	-	3	%
			T=[0, 70]°C, V_{DD}=[1.8, 3.6]V(1)	-10	-	+10	
			Full operation range(1)	-30	-	+30	
DC	Duty cycle			-	50	-	%
T_{start}	Start-up time			-	250	-	µs

Note:

1. These values are based on characterization and not covered by production testlimits.

Table 34-14. 32.768kHz External Crystal Oscillator (XOSC32K) Characteristics

Symbol	Description	Condition	Min.	typ	Max.	Unit
F_{out}	Frequency		-	32.768	-	kHz
T_{start}	Startup time	C_L=7.5pF	-	300	-	ms
C_L	Crystal load capacitance		7.5	-	12.5	pF
C_{TOSC1}	Parasitic capacitor load		-	5.5	-	pF
C_{TOSC2}			-	5.5	-	pF
ESR	Equivalent Series Resistance - Safety Factor=3	C_L=7.5pF	-	-	80	kΩ
		C_L=12.5pF	-	-	40	

Figure 34-2. External Clock Waveform Characteristics

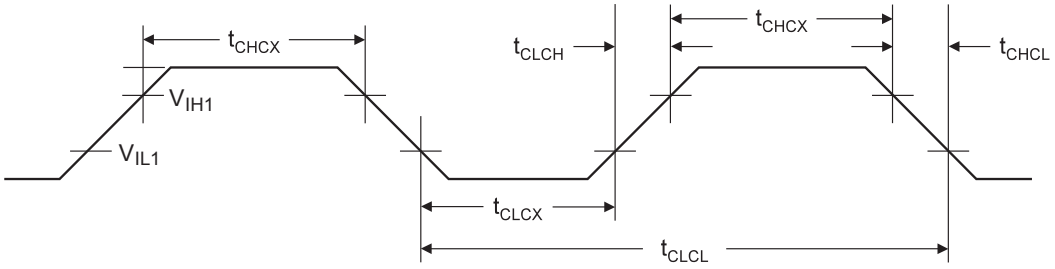


Table 34-15. External Clock Characteristics

Symbol	Description	Condition	V_{DD}=[1.8, 5.5]V		V_{DD}=[2.7, 5.5]V		V_{DD}=[4.5, 5.5]V		Unit
			Min.	Max.	Min.	Max.	Min.	Max.	
f_{CLCL}	Frequency		0	5.0	0.0	10.0	0.0	20.0	MHz
t_{CLCL}	Clock Period		200	-	100	-	50	-	ns

Symbol	Description	Condition	Min.	Typ.	Max.	Unit
f_{SSCK}	Slave SCK clock frequency	Slave	-	-	5	MHz
t_{SSCK}	Slave SCK Period	Slave	4*t Clkper	-	-	ns
t_{SSCKW}	SCK high/low width	Slave	2*t Clkper	-	-	ns
t_{SSCKR}	SCK rise time	Slave	-	-	1600	ns
t_{SSCKF}	SCK fall time	Slave	-	-	1600	ns
t_{SIS}	MOSI setup to SCK	Slave	3.0	-	-	ns
t_{SIH}	MOSI hold after SCK	Slave	t_{Clkper}	-	-	ns
t_{SSS}	SS setup to SCK	Slave	21	-	-	ns
t_{SSH}	SS hold after SCK	Slave	20	-	-	ns
t_{SOS}	MISO setup to SCK	Slave	-	8.0	-	ns
t_{SOH}	MISO hold after SCK	Slave	-	13	-	ns
t_{SOSS}	MISO setup after SS low	Slave	-	11	-	ns
t_{SOSH}	MISO hold after SS low	Slave	-	8.0	-	ns

34.13 TWI

Figure 34-7. TWI - Timing Requirements

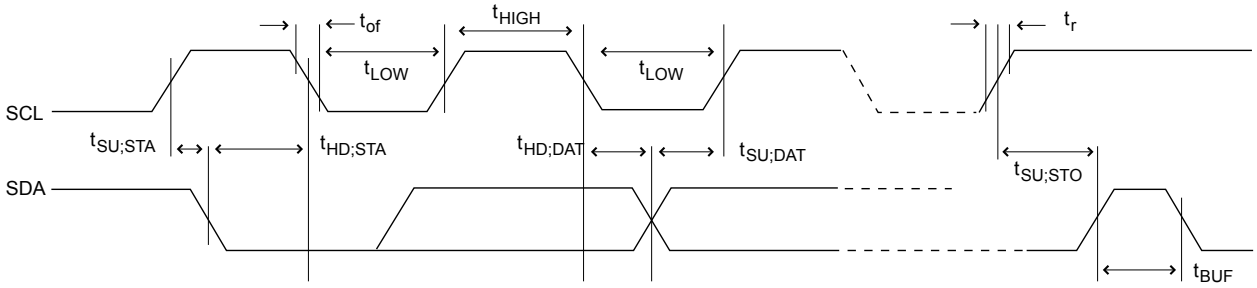
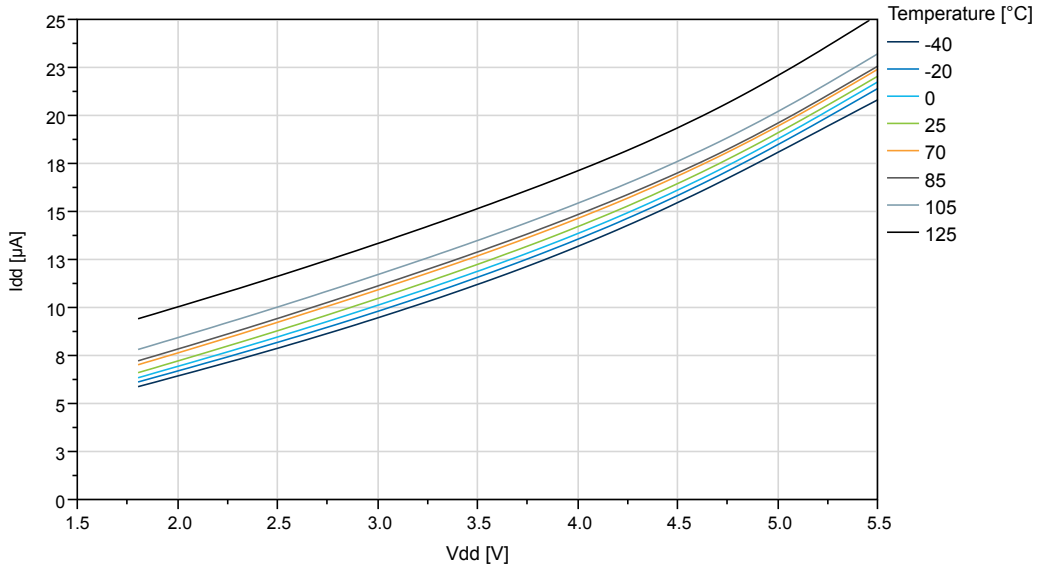


Table 34-19. TWI - Timing Characteristics

Symbol	Description	Condition	Min.	Typ.	Max.	Unit
f_{SCL}	SCL clock frequency		0	-	1000	kHz
V_{IH}	Input high voltage		$0.7 \times V_{DD}$	-	-	V
V_{IL}	Input low voltage		-	-	$0.3 \times V_{DD}$	V
V_{HYS}	Hysteresis of Schmitt trigger inputs		$0.1 \times V_{DD}$		$0.4 \times V_{DD}$	V
V_{OL}	Output low voltage	I_{load}=20mA, Fast mode+	-	-	$0.2V_{DD}$	V
		I_{load}=3mA, Normal mode, V_{DD}>2V	-	-	0.4V	

Figure 35-5. : Active Supply Current vs. V_{DD} (f=32KHz OSCULP32K)



35.1.2 Supply Currents in Idle Mode

Figure 35-6. Idle Supply Current vs. Frequency (1-20MHz) at T=25°C

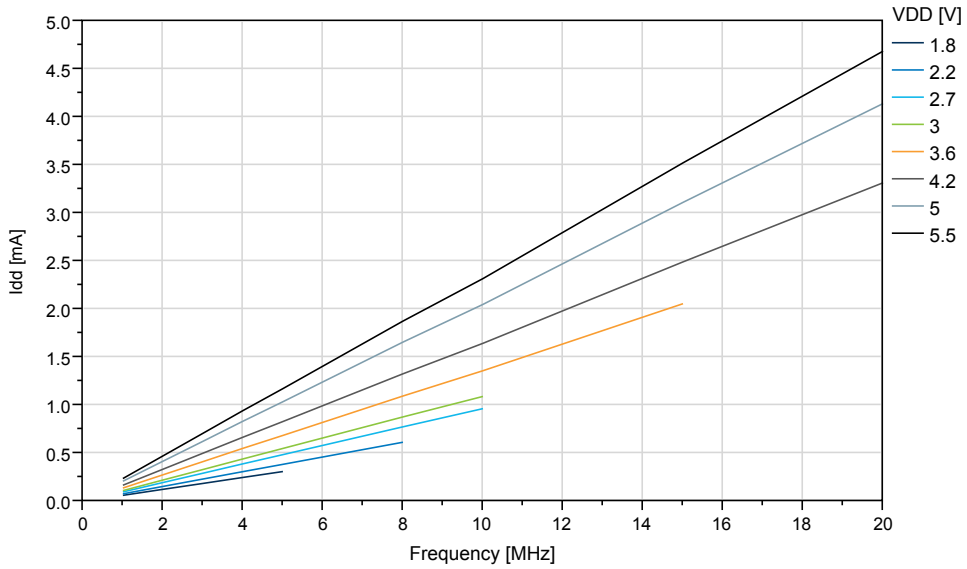


Figure 35-13. Standby Mode Supply Current vs. V_{DD} (RTC running with internal OSCULP32K)

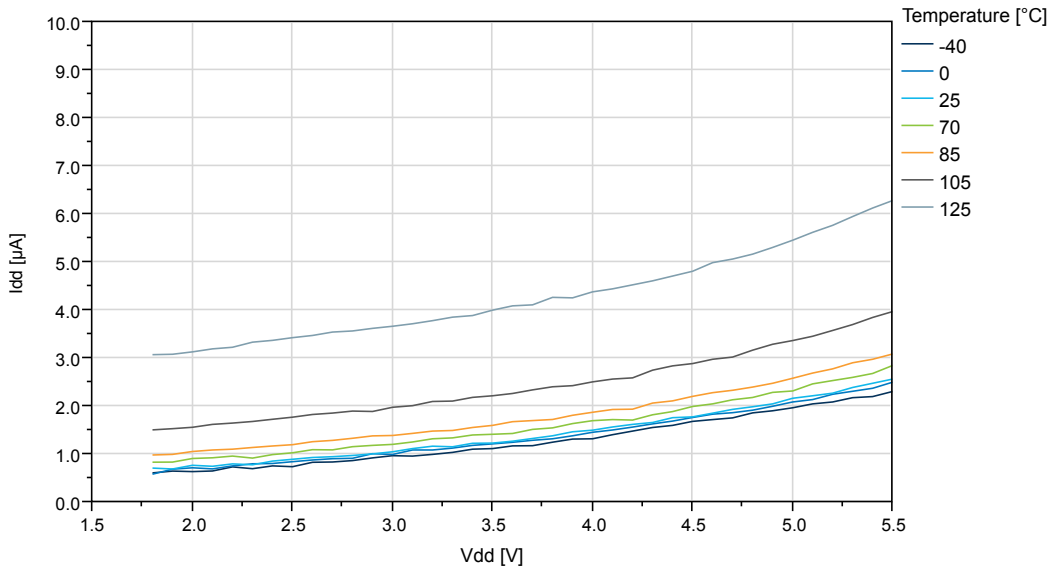
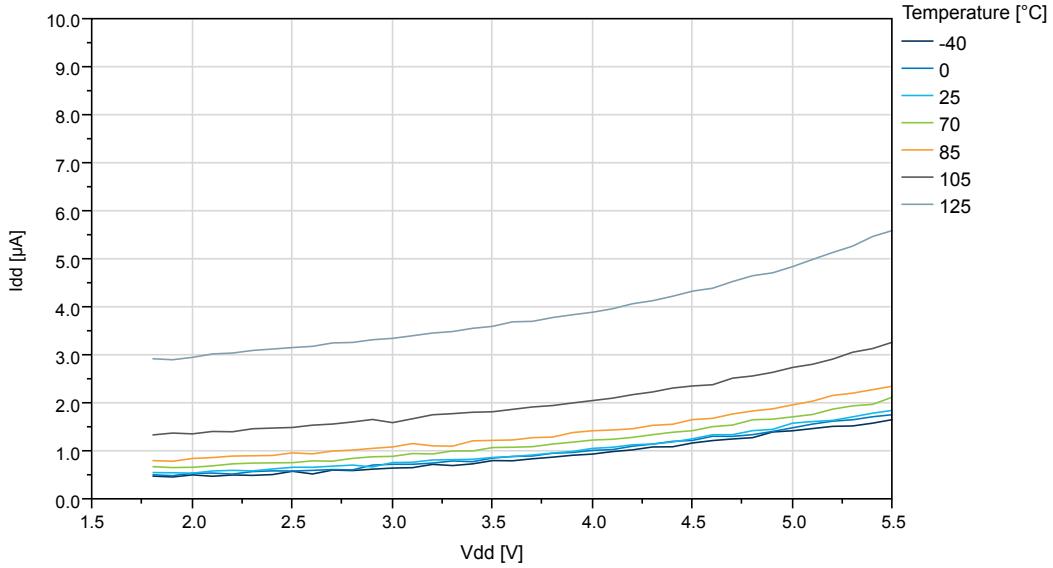


Figure 35-14. : Standby Mode Supply Current vs. V_{DD} (Sampled BOD running at 125Hz)



35.8 OSCULP32K Characteristics

Figure 35-59. OSCULP32K Internal Oscillator Frequency vs. Temperature

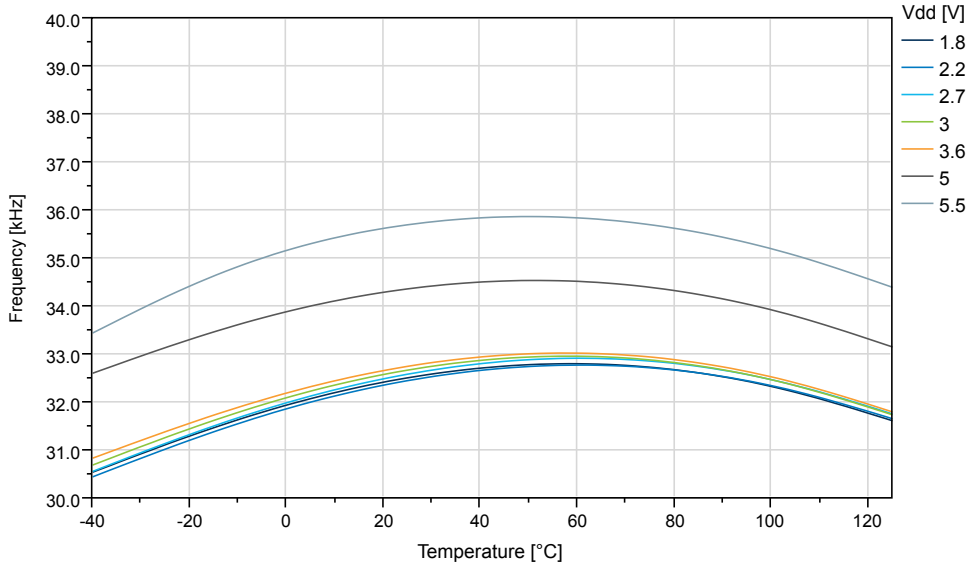
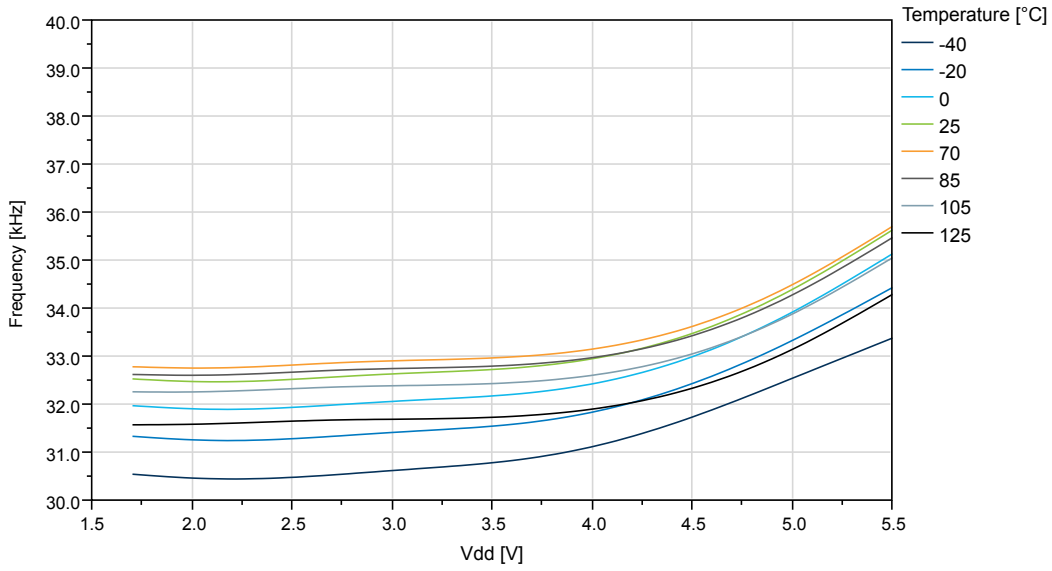


Figure 35-60. OSCULP32K Internal Oscillator Frequency vs. V_{DD}



38. Conventions

38.1 Numerical Notation

Table 38-1. Numerical Notation

Symbol	Description
165	Decimal number
0b0101	Binary number (example 0b0101 = 5 decimal)
'0101'	Binary numbers are given without prefix if unambiguous.
0x3B24	Hexadecimal number
X	Represents an unknown or don't care value
Z	Represents a high-impedance (floating) state for either a signal or a bus

38.2 Memory Size and Type

Table 38-2. Memory Size and Bit Rate

Symbol	Description
KB (kbyte)	kilobyte (2^{10} = 1024)
MB (Mbyte)	megabyte (2^{20} = 1024*1024)
GB (Gbyte)	gigabyte (2^{30} = 1024*1024*1024)
b	bit (binary '0' or '1')
B	byte (8 bits)
1kbit/s	1,000 bit/s rate (not 1,024 bit/s)
1Mbit/s	1,000,000 bit/s rate
1Gbit/s	1,000,000,000 bit/s rate
word	16 bit

38.3 Frequency and Time

Symbol	Description
kHz	1kHz = 10^3Hz = 1,000Hz
KHz	1KHz = 1,024Hz, 32KHz = 32,768Hz
MHz	10^6 = 1,000,000Hz
GHz	10^9 = 1,000,000,000Hz