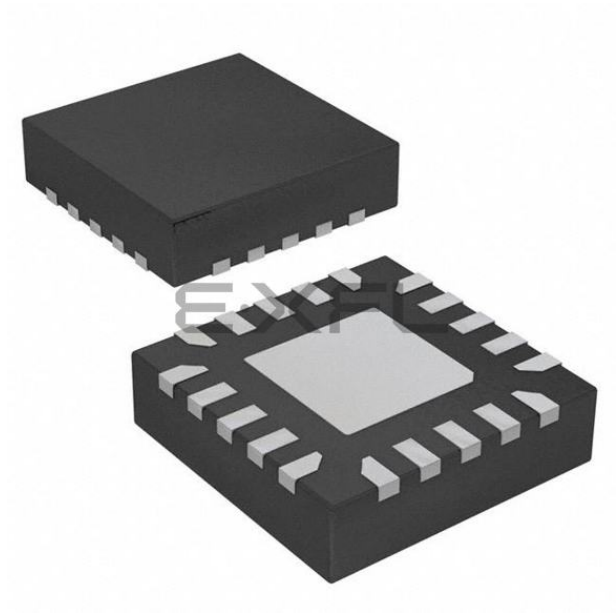


Welcome to E-XFL.COM

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	18
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	20-VFQFN Exposed Pad
Supplier Device Package	20-VQFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/attiny816-mn

9.1. Features....48
9.2. Overview....48
9.3. Functional Description....49
9.4. Register Summary - NVMCTRL....55
9.5. Register Description....55

10. CLKCTRL - Clock Controller....59
10.1. Features....59
10.2. Overview....59
10.3. Functional Description....61
10.4. Register Summary - CLKCTRL....66
10.5. Register Description....66

11. SLPCTRL - Sleep Controller....73
11.1. Features....73
11.2. Overview....73
11.3. Functional Description....74
11.4. Register Summary - SLPCTRL....77
11.5. Register Description....77

12. RSTCTRL - Reset Controller....78
12.1. Features....78
12.2. Overview....78
12.3. Functional Description....79
12.4. Register Summary - RSTCTRL....82
12.5. Register Description....82

13. CPUINT - CPU Interrupt Controller....84
13.1. Features....84
13.2. Overview....84
13.3. Functional Description....86
13.4. Register Summary - CPUINT....92
13.5. Register Description....92

14. EVSYS - Event System....95
14.1. Features....95
14.2. Overview....95
14.3. Functional Description....98
14.4. Register Summary - EVSYS....100
14.5. Register Description....100

15. PORTMUX - Port Multiplexer....106
15.1. Overview....106
15.2. Register Summary - PORTMUX....107
15.3. Register Description....107

16. PORT - I/O Pin Configuration....110
16.1. Features....110
16.2. Overview....110

Only the number of bits required to address the available data memory including external memory (up to 64KB) is implemented for each device. Unused bits will always read as zero.

The CPU.SPL and CPU.SPH register pair represents the 16-bit value, CPU.SP. The low byte [7:0] (suffix L) is accessible at the original offset. The high byte [15:8] (suffix H) can be accessed at offset + 0x01. For more details on reading and writing 16-bit registers, refer to Accessing 16-bit Registers.

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

Name: SP
Offset: 0x0D
Reset: 0xxxxx
Property: -

Bit	15	14	13	12	11	10	9	8
	SP[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	x	x	x	x	x	x	x	x

Bit	7	6	5	4	3	2	1	0
	SP[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	x	x	x	x	x	x	x	x

Bits 15:8 – SP[15:8]: Stack Pointer high byte
These bits hold the MSB of the 16-bit register.

Bits 7:0 – SP[7:0]: Stack Pointer low byte
These bits hold the LSB of the 16-bit register.

8.7.3 Status Register

The Status register contains information about the result of the most recently executed arithmetic or logic instruction. For details about the bits in this register and how they are affected by the different instructions, see the *Instruction Set Summary*.

Name: SREG
Offset: 0x0F
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	I	T	H	S	V	N	Z	C
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 7 – I: Global Interrupt Enable
Writing a '1' to this bit enable interrupts on the device.

Name: CTRLB
Offset: 0x01
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
							ADC0REFEN	DAC0REFEN
Access							R/W	R/W
Reset							0	0

Bit 1 – ADC0REFEN: ADC0 Reference Force Enable
Writing a '1' to this bit forces the voltage reference for the ADC0 to be running, even if it not requested.

Writing a '0' to this bit allows to automatic enable/disable of the reference source by the peripheral.

Bit 0 – DAC0REFEN: DAC0 and AC0 Reference Force Enable
Writing a '1' to this bit forces the voltage reference for the DAC0 and AC0 to be running, even if it not requested.

Writing a '0' to this bit allows to automatic enable/disable of the reference source by the peripheral.

Name	Description
Four ramp	Counter is reset to zero four times during a TCD cycle.
Dual ramp	Counter count both up and down between zero and selected top value.

22.3 Functional Description

22.3.1 Initialization and Disabling

To initialize the TCD:

1. Configure the static registers to the desired functionality.
2. Write desired initial values to the double-buffered registers.
3. Ensure that the Enable Ready bit (ENRDY) in the Status register (TCD.STATUS) is set to '1'.
4. Enable the TCD by writing a '1' to the ENABLE bit in the Control A register (TCD.CTRLA).

It is possible to disable the TCD in two different ways:

1. By writing a '0' to ENABLE in TCD.CTRLA. This disables the TCD instantly when synchronized to the TCD core domain.
2. By writing a '1' to the Disable at End of Cycle Strobe bit (DISEOC) in the Control E register (TCD.CTRLE). This disables the TCD at the end of the TCD cycle.

The bit fields in the TCD.CTRLA register are enable-protected, with exception of the ENABLE bit. They can only be written when ENABLE is written to '0' first.

Related Links

Register Synchronization Categories

22.3.2 Operation

22.3.2.1 Register Synchronization Categories

Most of the IO registers need to be synchronized to the asynchronous TCD core clock domain. This is done in different ways for different register categories:

- Command and Enable Control registers
- Doubled-buffered registers
- Static registers
- Normal IO and STATUS registers

See Table 22-3 for categorized registers.

Command and Enable Registers

Because of synchronization between the clock domains it is only possible to change the Enable bits while the Enable Ready bit (ENRDY) in the Status register (TCD.STATUS) is '1'.

The Control E register commands (TCD.CTRLE) are automatically synchronized to the TCD core domain when the TCD is enabled and as long as there not a synchronization ongoing already. Check in the Status register if the Command Ready bit (CCMDRDY) is '1' (TCD.STATUS) to ensure that it is possible to write a new command. TCD.CTRLE is a strobe register that will clear itself when the command is done.

The Control E register commands are:

WAVEGEN	DITHERSEL in TCD.DITCTRL	Additional TCD clock cycles to TCD cycle
	Dead-time B	0 (not supported)
	Dead-time A and B	0 (not supported)

The differences in the number of TCD clock cycles added to the TCD cycle is caused by the different number of compare values used by the TCD cycle. For example in One Ramp Mode, only CMPBCLR affects the TCD cycle time.

For DITHERSEL configurations where no extra cycles are added to the TCD cycles, compensation is reached by shortening the following output state.

> **Example:**
>
> In One Ramp Mode with DITHERSEL selecting Dead-time B, the Dead-time B will be increased by one cycle when dither overflow occurs. This reduces On-time B by one cycle.

22.3.2.6 TCD Counter Capture

Because the TCD counter is asynchronous to the system clock it is not possible to read out the counter value directly. It is possible to capture the TCD counter value, synchronized to the IO clock domain in two different ways.

- Capture value on input Events
- Software capture

The capture logic contains two separate capture blocks, CAPTUREA and CAPTUREB, that can capture and synchronize the TCD counter value to the IO clock domain. CAPTUREA/B can be triggered by input Event A/B or by software.

The capture values can be read by reading first TCD.CAPTUREXL and then TCD.CAPTUREXH registers.

Captures Triggered by Input Events

To enable capture on input Event, write a '1' to the ACTION bit in the respective Event Control x register (TCD.EVCTRL) when configuring an Event input.

When a capture has occurred, the TRIGA/B flag is raised in the Interrupt Flags register (TCD.INTFLAGS). The according TRIGA/B interrupt is executed if enabled by writing a '1' to the respective Trigger Interrupt x Enable bit (TRIGx) in the Interrupt Control register (TCD.INTCTRL). By polling TRIGx in TCD.INTFLAGS, the user knows that a CAPTUREx value is available, and can read out the value by reading first the TCD.CAPTUREXL and then TCD.CAPTURXH registers.

> **Example**
>
> In order to do PWM capture both event A and B should be connected to the same asynchronous event channel that contains the PWM signal. To get information on the PWM signal one event input should be configured to capture the rising edge of the signal. The other event input should be configured to capture the falling edge of the signal.

Capture Triggered by Software

Software can capture the TCD value by writing a '1' to respective Software Capture A/B Strobe bit (SCAPTUREx) in the Control E register (TCD.CTRLE). When this command is executed and the Command Ready bit (CMDRDY) in the Status register (TCD.STATUS) reads '1' again, the CAPTUREA/B

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

INTCTRL

PITINTCTRL

23.7 Sleep Mode Operation

The RTC will continue to operate in Idle sleep mode. It will run in Standby sleep mode if RTC.CTRLA.RUNSTDBY is set.

The PIT will continue to operate in any sleep mode.

Related Links

CTRLA

23.8 Synchronization

Both the RTC and the PIT are asynchronous, operating from a different clock source (CLK_RTC) independently of the main clock (CLK_PER). For control and count register updates, it will take a number of RTC clock and/or peripheral clock cycles before an updated register value is available in a register or until a configuration change has effect on the RTC or PIT, respectively. This synchronization time is described for each register in the Register Description.

For some RTC registers, a Synchronization Busy flag is available (CMPBUSY, PERBUSY, CNTBUSY, CTRLABUSY) in the Status register (RTC.STATUS).

For the RTC.PITCTRLA register, a Synchronization Busy flag (SYNCBUSY) is available in the PIT Status register (RTC.PITSTATUS).

Check for busy should be performed before writing to the mentioned registers.

Related Links

CLKCTRL - Clock Controller

23.9 Configuration Change Protection

Not applicable.

Writing a '1' to this bit clears the flag.

23.11.5 Temporary

The Temporary register is used by the CPU for single-cycle, 16-bit access to the 16-bit registers of this peripheral. It can also be read and written by software. See also Accessing 16-bit Registers. There is one common Temporary register for all the 16-bit registers of this peripheral.

Name: TEMP
Offset: 0x4
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	TEMP[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – TEMP[7:0]: Temporary

23.11.6 Debug Control

Name: DBGCTRL
Offset: 0x05
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
								DBGRUN
Access	R	R	R	R	R	R	R	R/W
Reset	0	0	0	0	0	0	0	0

Bit 0 – DBGRUN: Debug Run

Value	Description
0	The peripheral is halted in break debug mode and ignores events.
1	The peripheral will continue to run in break debug mode when the CPU is halted.

23.11.7 Clock Selection

Name: CLKSEL
Offset: 0x07
Reset: 0x00
Property: -

24.5.7 Control B

Name: CTRLB
Offset: 0x06
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	RXEN	TXEN		SFDEN	ODME	RXMODE[1:0]		MPCM
Access	R/W	R/W		R/W	R/W	R/W	R/W	R/W
Reset	0	0		0	0	0	0	0

Bit 7 – RXEN: Receiver Enable
Writing this bit to '1' enables the USART Receiver. The Receiver will override normal port operation for the RxD pin when enabled. Disabling the Receiver will flush the receive buffer invalidating the FERR, BUFOVF, and PERR flags. In GENAUTO and LINAUTO mode, disabling the receiver will reset the auto-baud detection logic.

Bit 6 – TXEN: Transmitter Enable
Writing this bit to '1' enables the USART Transmitter. The Transmitter will override normal port operation for the TxD pin when enabled. Disabling the Transmitter (writing TXEN to zero) will not become effective until ongoing and pending transmissions are completed, i.e., when the Transmit Shift Register and Transmit Buffer Register do not contain data to be transmitted. When the transmitter is disabled, it will no longer override the TxDn pin, and the pin direction is set as input automatically by hardware, even if it was configured as output by the user.

Bit 4 – SFDEN: Start Frame Detection Enable
Writing this bit to '1' enables the USART Start Frame Detection mode. The start frame detector is able to wake up the system from idle or standby sleep modes when a high (IDLE) to low (START) transition is detected on the RxD line.

Bit 3 – ODME: Open Drain Mode Enable
Writing this bit to '1' will make the TxD pin to have open-drain functionality. A pull-up resistor is needed to prevent the line from floating when a logic one is output to TxD pin.

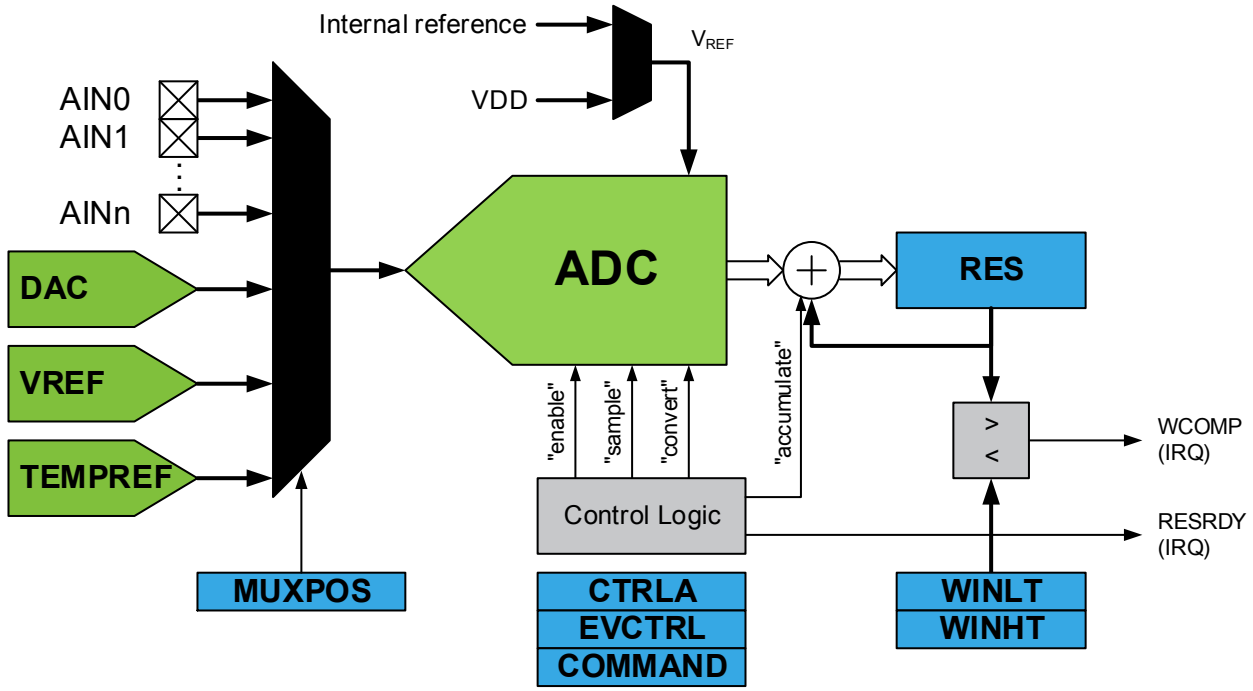
Bits 2:1 – RXMODE[1:0]: Receiver Mode
In CLK2X mode, the divisor of the baud rate divider will be reduced from 16 to 8 effectively doubling the transfer rate for asynchronous communication modes. For synchronous operation the CLK2X mode has no effect and RXMODE should always be written to zero. RXMODE must be zero when the USART Communication Mode is configured to IRCOM. Setting RXMODE to GENAUTO enables generic auto-baud where the SYNC character is valid when eight low and high bits has been registered. In this mode any SYNC character that gives a valid BAUD rate will be accepted. In LINAUTO mode the SYNC character is constrained and found valid if each two bits falls within 32 +/- 6 baud samples of the internal baud rate and match data value 0x55. The GENAUTO and LINAUTO mode is only supported for USART operated in asynchronous slave mode.

Value	Name	Description
0x0	NORMAL	Normal USART Mode, Standard Transmission Speed
0x1	CLK2X	Normal USART Mode, Double Transmission Speed

© 2017 Microchip Technology Inc.

30.2.1 Block Diagram

Figure 30-1. Block Diagram



The analog input channel is selected by writing to the MUXPOS bits in the MUXPOS register (ADC.MUXPOS). Any of the ADC input pins, GND, internal Voltage Reference (VREF), or temperature sensor, can be selected as single ended input to the ADC. The ADC is enabled by writing a '1' to the ADC ENABLE bit in the Control A register (ADC.CTRLA). Voltage reference and input channel selections will not go into effect before the ADC is enabled. The ADC does not consume power when the ENABLE bit in ADC.CTRLA is zero.

The ADC generates a 10-bit result which can be read from the Result Register (ADC.RES). The result is presented right adjusted.

30.2.2 Signal Description

Pin Name	Type	Description
AIN[11:0]	Analog input	analog input to be converted

Related Links

Configuration Summary

I/O Multiplexing and Considerations

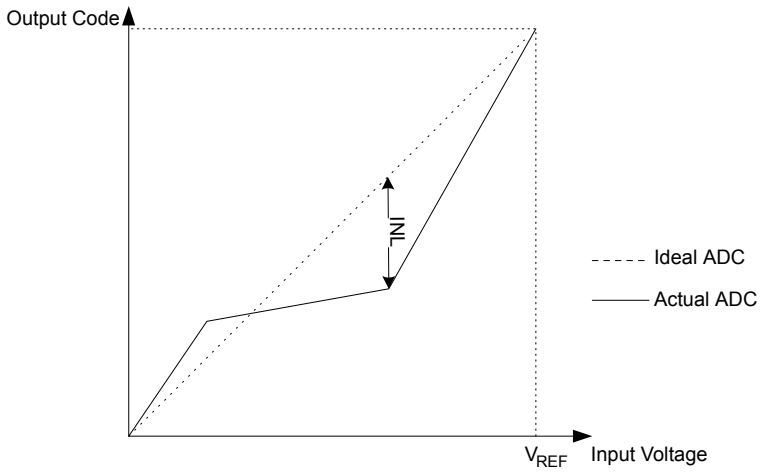
30.2.3 System Dependencies

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

Table 30-1. ADC System Dependencies

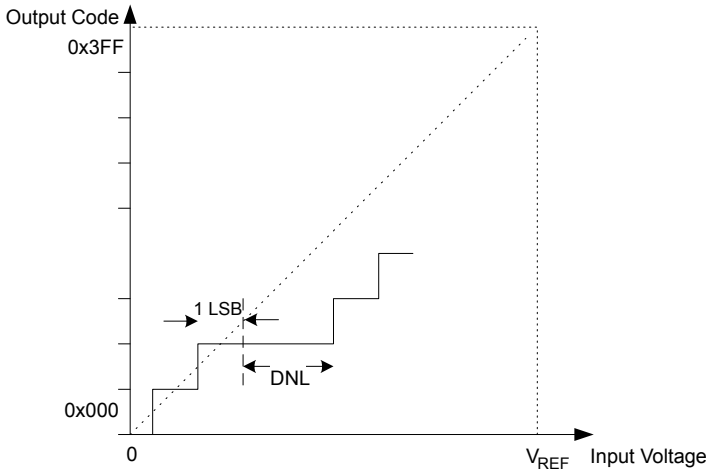
Dependency	Applicable	Peripheral
Clocks	Yes	CLKCTRL
I/O Lines and Connections	Yes	PORT

Figure 30-4. Integral Non-Linearity



Differential Non-Linearity (DNL) The maximum deviation of the actual code width (the interval between two adjacent transitions) from the ideal code width (1 LSB). Ideal value: 0 LSB.

Figure 30-5. Differential Non-Linearity



Quantization Error Due to the quantization of the input voltage into a finite number of codes, a range of input voltages (1 LSB wide) will code to the same value. Always ±0.5 LSB.

Absolute Accuracy The maximum deviation of an actual (unadjusted) transition compared to an ideal transition for any code. This is the compound effect of all aforementioned errors. Ideal value: ±0.5 LSB.

30.3 Functional Description

30.3.1 Initialization

The following steps are recommended in order to initialize ADC operation:

1. Configure the resolution by writing to the Resolution Selection bit (RESSEL) in the Control A register (ADC.CTRLA).
2. Optional: Enable the Free Running mode by writing a '1' to the Free Running bit (FREERUN) in ADC.CTRLA.

Bit	7	6	5	4	3	2	1	0
						WINCM[2:0]		
Access	R	R	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 2:0 – WINCM[2:0]: Window Comparator Mode
This field enable and define when the interrupt flag is set in Window Comparator mode. RESULT is the 16-bit accumulator result. WINLT and WINHT are 16-bit lower threshold value and 16-bit higher threshold value, respectively.

Value	Name	Description
0x0	NONE	No Window Comparison (default)
0x1	BELOW	*RESULT < WINLT*
0x2	ABOVE	*RESULT > WINHT*
0x3	INSIDE	*WINLT < RESULT < WINHT*
0x4	OUTSIDE	*RESULT < WINLT or RESULT >WINHT)*
Other	-	Reserved

30.5.6 Sample Control

Name: SAMPCTRL
Offset: 0x5
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
				SAMPLEN[4:0]				
Access	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 4:0 – SAMPLEN[4:0]: Sample Length
These bits extend the ADC sampling length in number of CLK_ADC cycles. By default the sampling time is two CLK_ADC cycles. Increasing the sampling length allows sampling sources with higher impedance. The total conversion time increased with the selected sampling length.

30.5.7 MUXPOS

Name: MUXPOS
Offset: 0x06
Reset: 0x00
Property: -

31. DAC - Digital to Analog Converter

31.1 Features

- 8-bit resolution
- Up to 350ksps conversion rate
- High drive capabilities
- Functioning as input to Analog Comparator (AC) or ADC
- One instances DAC0

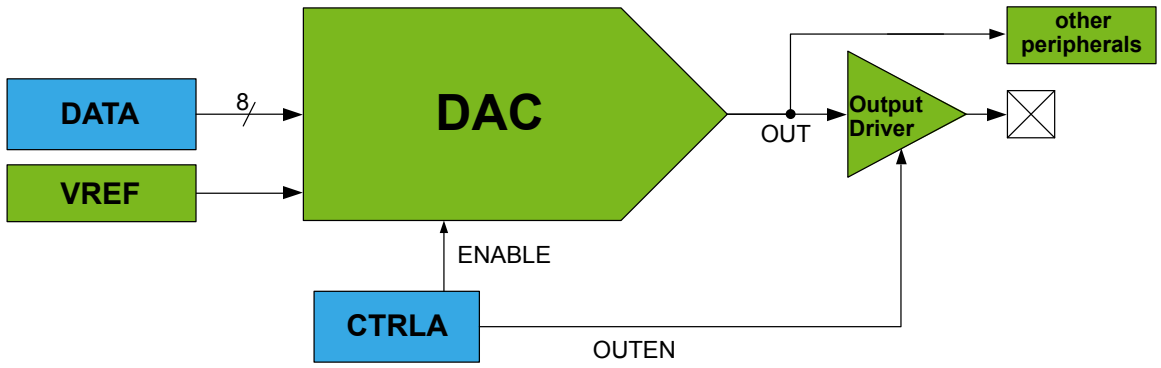
31.2 Overview

The Digital-to-Analog Converter (DAC) converts a digital value written to the Data register (DAC.DATA) to an analogue voltage. The conversion range is between GND and the selected reference voltage.

The DAC features an 8-bit Resistor String type DAC, capable of converting 350,000 samples per second (350ksps). The DAC uses the internal Voltage Reference (VREF) as upper limit for conversion. The DAC has one continuous time output with high drive capabilities, which is able to drive 5kΩ and/or 30pF load. The DAC conversion can be started from the application by writing to the data conversion registers.

31.2.1 Block Diagram

Figure 31-1. DAC Block Diagram



31.2.2 Signal Description

Signal	Description	Type
OUT	DAC output	Analog

Related Links

I/O Multiplexing and Considerations

31.2.3 System Dependencies

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

31.4 Register Summary - DAC

Offset	Name	Bit Pos.								
0x00	CTRLA	7:0	RUNSTDBY	OUTEN						ENABLE
0x01	DATA	7:0	DATA[7:0]							

31.5 Register Description

31.5.1 Control A

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

Bit	7	6	5	4	3	2	1	0
	RUNSTDBY	OUTEN						ENABLE
Access	R/W	R/W						R/W
Reset	0	0						0

Bit 7 – RUNSTDBY: Run in Standby Mode
If this bit is written to '1', the DAC or Output Buffer will not automatically be disabled when the device is entering Standby sleep mode.

Bit 6 – OUTEN: Output Buffer Enable
Writing a '1' to this bit enables the Output Buffer and sends the OUT signal to a pin.

Bit 0 – ENABLE: DAC Enable
Writing a '1' to this bit enables the DAC.

31.5.2 DATA

Name: DATA
Offset: 0x01
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	DATA[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – DATA[7:0]: Data
These bits contains the digital data which will be converted to an analog voltage.

33. UPDI - Unified Program and Debug Interface

33.1 Features

- Programming
  - External programming through UPDI 1-wire (1W) interface
    - Enable programming by 12V or fuse
    - Uses the $\overline{\text{RESET}}$ pin of the device for programming
    - No GPIO pins occupied during operation
    - Asynchronous Half-Duplex UART protocol towards the programmer
- Debugging
  - Memory mapped access to device address space (NVM, RAM, I/O)
  - No limitation on device clock frequency
  - Unlimited number of user program breakpoints
  - 2 Hardware Breakpoints
  - Run-time readout of program counter (PC), Stack Pointer (SP) and CPU Status register (CPU_SREG) for code profiling
  - Program Flow Control
    - Go, Stop, Reset, Step Into
  - Non-intrusive run-time chip monitoring without accessing system registers
    - Monitor CRC status and sleep status
- Unified Programming and Debug Interface (UPDI)
  - Built in error detection with error signature readout
  - Frequency Measurement of internal oscillators using the Event System

33.2 Overview

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

The UPDI supports programming of nonvolatile memory (NVM) space; FLASH, EEPROM, fuses, lockbits and the user row. In addition the UPDI can access the entire I/O and data space of the device. See the NVM Controller documentation for programming via the NVM controller and executing NVM controller commands.

Programming and debugging is done through the UPDI Physical interface (UPDI PHY), which is a 1-wire UART based half-duplex interface using the $\overline{\text{RESET}}$ pin for data reception and transmission. Clocking of UPDI PHY is done by an internal oscillator. Enabling of the 1-wire interface, by disabling the reset functionality, is either done by 12V programming or by fusing the $\overline{\text{RESET}}$ pin to UPDI by setting the $\overline{\text{RESET}}$ Pin Configuration (RSTPINCFG) bits in FUSE.SYSCFG0. The UPDI Access layer grants access to the Bus Matrix, with memory mapped access to system blocks such as Memories, NVM, and peripherals.

The Asynchronous System Interface (ASI) provides direct interface access to On-Chip Debugging (OCD), NVM and System Management features. This gives the debugger direct access to system information, without requesting bus access.

Related Links

34. Electrical Characteristics

34.1 Disclaimer

All typical values are measured at T = 25°C and V_{DD}=3V unless otherwise specified. All minimum and maximum values are valid across operating temperature and voltage unless otherwise specified.

34.2 Absolute Maximum Ratings

Stresses beyond those listed in this section 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 34-1. Absolute Maximum Ratings

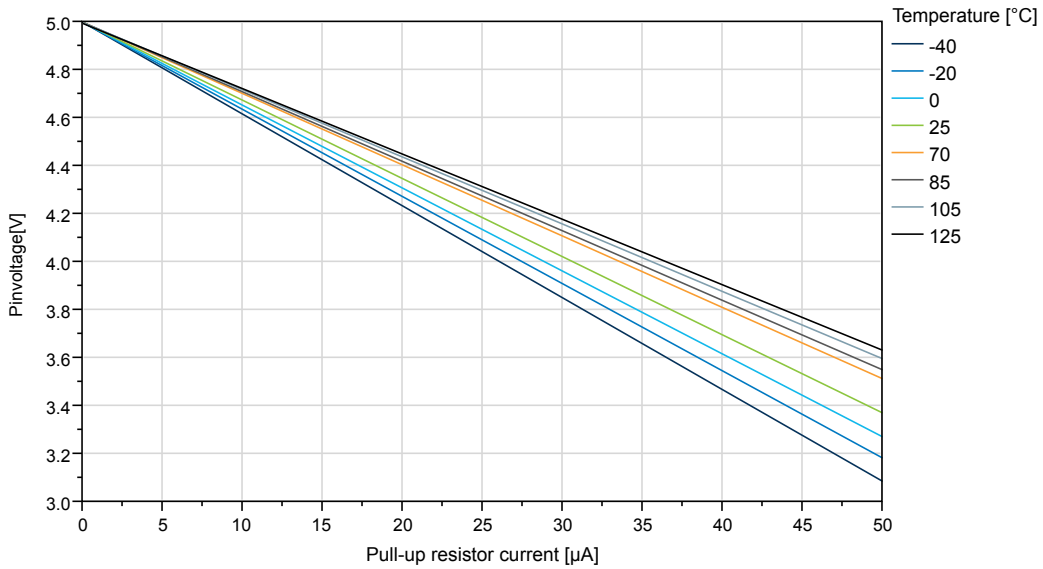
Symbol	Description	Conditions	Min.	Max.	Unit
V_{DD}	Power Supply Voltage		-0.5	6	V
I_{VDD}	Current into a V_{DD} pin	T=[-40, 85]°C	-	200	mA
		T=[85, 125]°C	-	100	mA
I_{GND}	Current out of a GND pin	T=[-40, 85]°C	-	200	mA
		T=[85, 125]°C	-	100	mA
V_{RST}	RESET pin voltage with respect to GND		-0.5	13	V
V_{PIN}	Pin voltage with respect to GND		-0.5	V_{DD}+0.5	V
I_{PIN}	I/O pin sink/source current		-40	40	mA
I_{c1}(1)	I/O pin injection current except RESET pin	V_{pin}<GND-0.6V or 5.5V<V_{pin}<=6.1V 4.9V<V_{DD}<=5.5V	-1	1	mA
I_{c2}(1)	I/O pin injection current except RESET pin	V_{pin}<GND-0.6V or V_{pin}<=5.5V V_{DD}<=4.9V	-15	15	mA
I_{ctot}	Sum of I/O pin injection current except RESET pin		-45	45	mA
$T_{storage}$	Storage temperature		-65	150	°C

Note:

1. – If Vpin is lower than GND-0.6V, then a current limiting resistor is required. The negative DC injection current limiting resistor is calculated as R = (GND-0.6V – V_{pin})/I_{Cn}.
   – If Vpin is greater than VDD+0.6V, then a current limiting resistor is required. The positive DC injection current limiting resistor is calculated as R = (V_{pin}-(VDD+0.6))/I_{Cn}.

© 2017 Microchip Technology Inc.

Figure 35-45. IO Pin Pull-Up Resistor Current vs. Input Voltage (V_{DD}=5.0V)



35.3 VREF Characteristics

Figure 35-46. Internal 0.55V Reference vs. Temperature

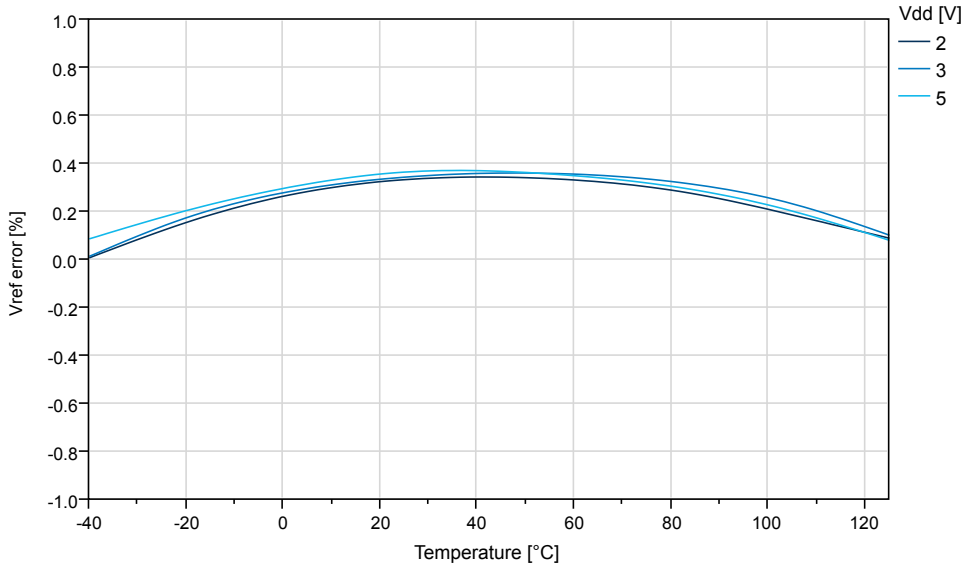
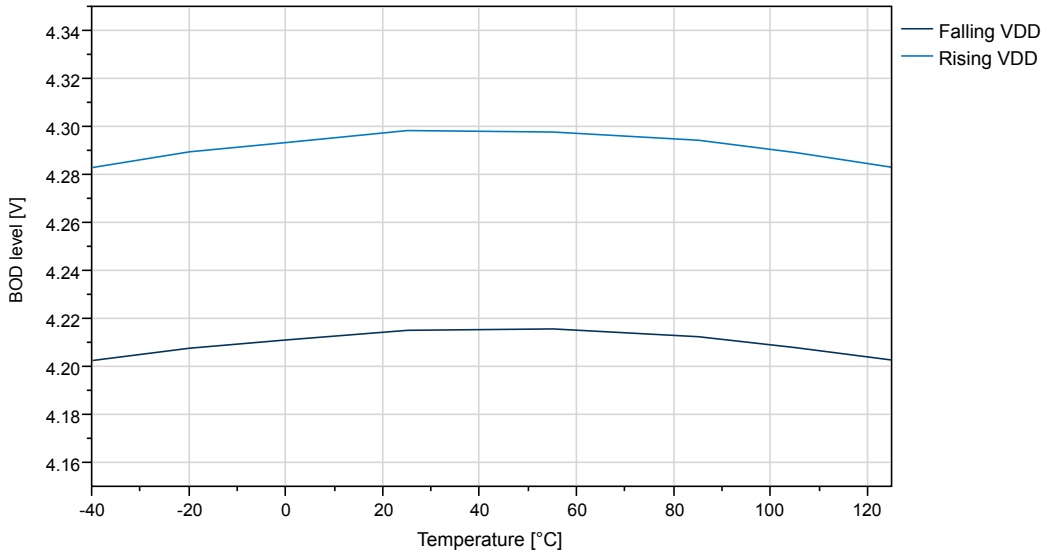


Figure 35-55. BOD Threshold vs. Temperature (Level 4.3V)



35.5 ADC Characteristics

Figure 35-56. Absolute Accuracy vs. V_{DD} (115ksps) at T=25°C

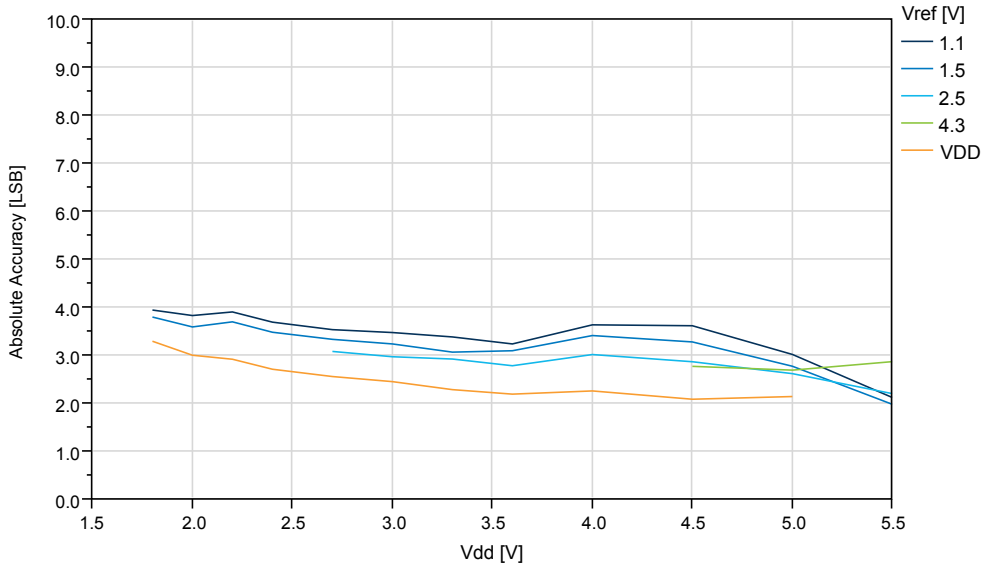


Figure 35-57. Absolute Accuracy vs. V_{ref} (V_{DD}=5.0V, 115ksps)

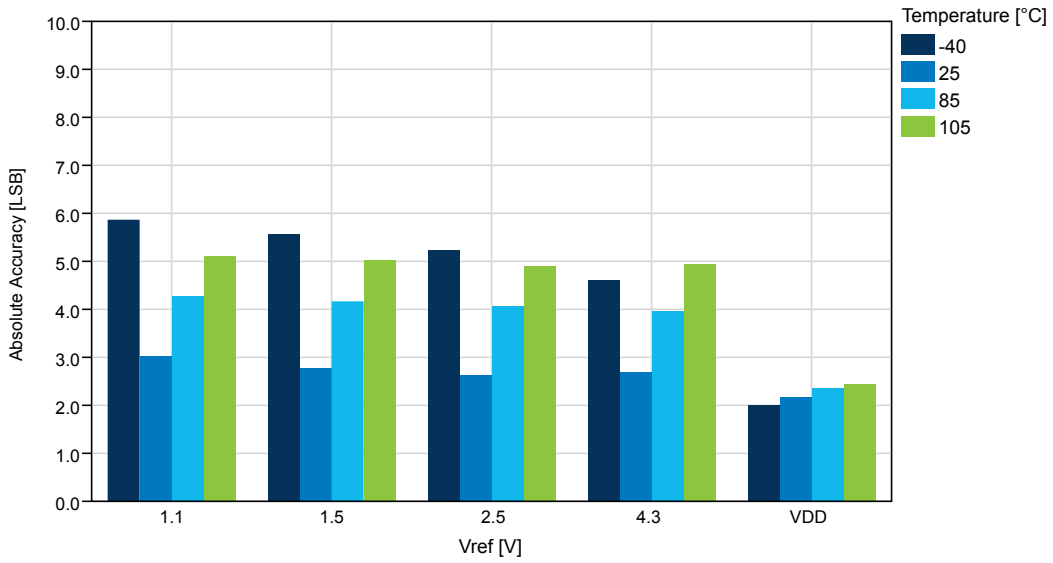


Figure 35-58. DNL Error vs. V_{DD} (115ksps) at T=25°C

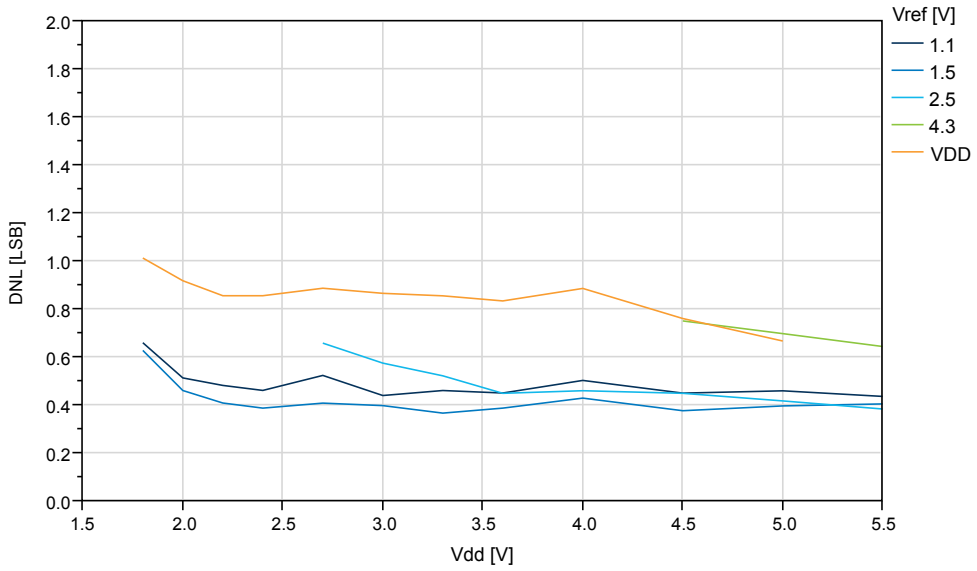
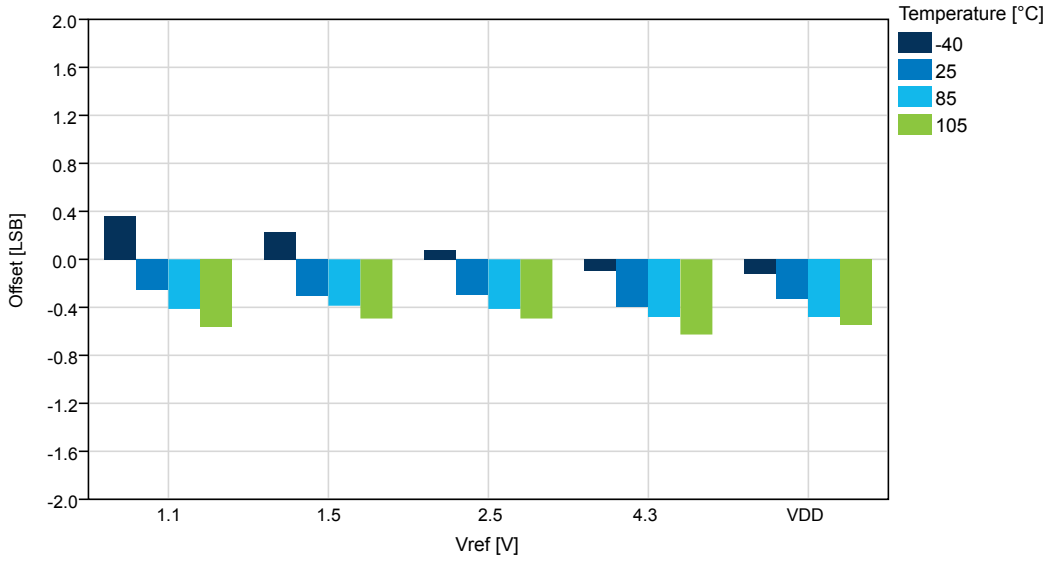


Figure 35-65. Offset Error vs. V_{ref} (V_{DD}=5.0V, 115ksps)



35.6 AC Characteristics

Figure 35-66. Hysteresis vs. V_{CM} - 10mV (V_{DD}=5V)

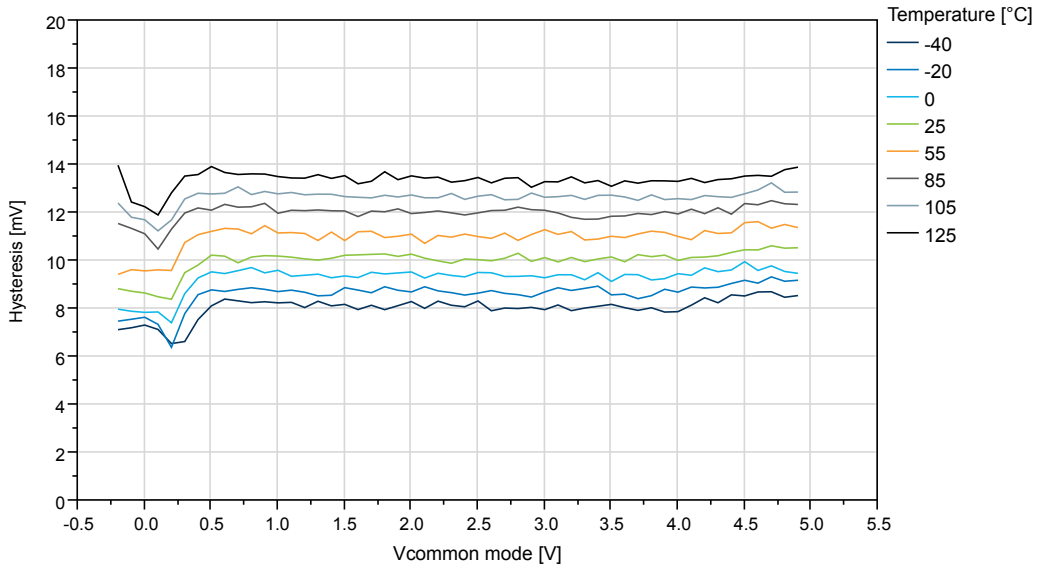
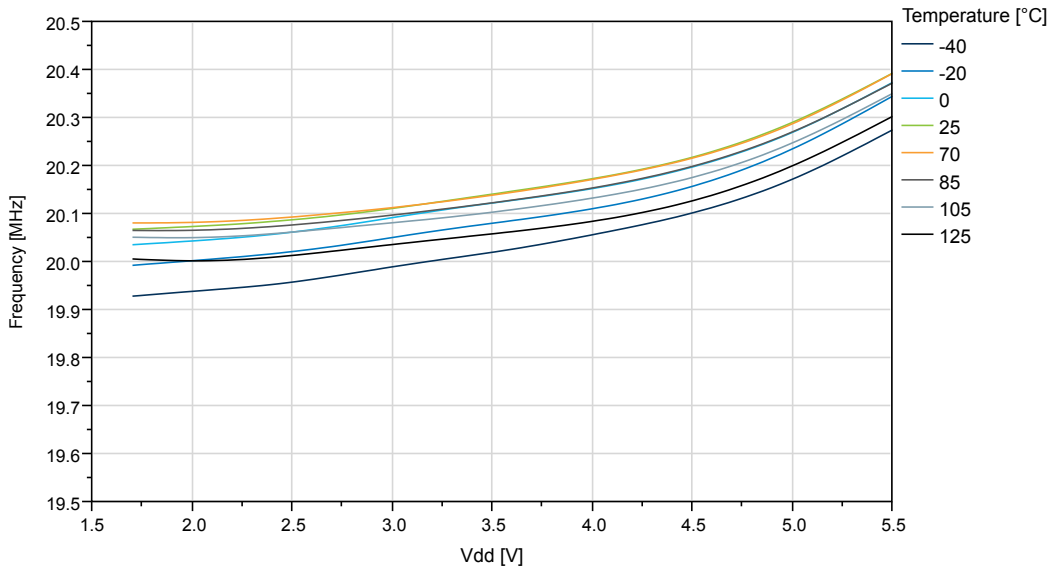


Figure 35-73. OSC20M Internal Oscillator: Frequency vs. V_{DD}



35.8 OSCULP32K Characteristics

Figure 35-74. OSCULP32K Internal Oscillator Frequency vs. Temperature