28. ADC – Analog-to-Digital Converter

28.1 Features

- 12-bit resolution
- Up to two million samples per second
  - Two inputs can be sampled simultaneously using ADC and 1x gain stage
  - Four inputs can be sampled within 1.5µs
  - Down to 2.5µs conversion time with 8-bit resolution
  - Down to 3.5µs conversion time with 12-bit resolution
- Differential and single-ended input
  - Up to 16 single-ended inputs
  - 16x4 differential inputs without gain
  - 8x4 differential input with gain
- Built-in differential gain stage
  - 1/2x, 1x, 2x, 4x, 8x, 16x, 32x, and 64x gain options
- Single, continuous and scan conversion options
- Four internal inputs
  - Internal temperature sensor
  - DAC output
  - V_{CC} voltage divided by 10
  - 1.1V bandgap voltage
- Four conversion channels with individual input control and result registers
  - Enable four parallel configurations and results
- Internal and external reference options
- Compare function for accurate monitoring of user defined thresholds
- Optional event triggered conversion for accurate timing
- Optional DMA transfer of conversion results
- Optional interrupt/event on compare result

28.2 Overview

The ADC converts analog signals to digital values. The ADC has 12-bit resolution and is capable of converting up to two million samples per second (MSPS). The input selection is flexible, and both single-ended and differential measurements can be done. For differential measurements, an optional gain stage is available to increase the dynamic range. In addition, several internal signal inputs are available. The ADC can provide both signed and unsigned results.

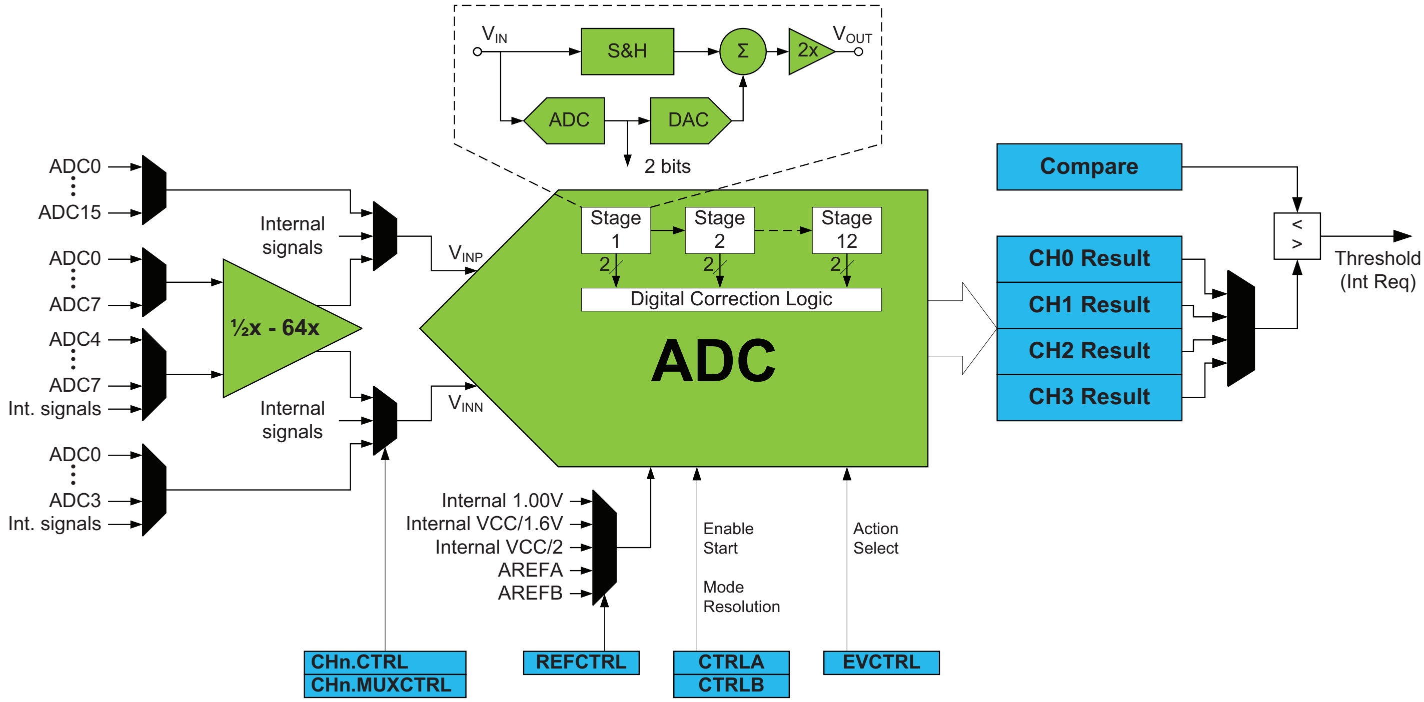
This is a pipelined ADC that consists of several consecutive stages. The pipelined design allows a high sample rate at a low system clock frequency. It also means that a new input can be sampled and a new ADC conversion started while other ADC conversions are still ongoing. This removes dependencies between sample rate and propagation delay.

The ADC has four conversion channels (0-3) with individual input selection, result registers, and conversion start control. The ADC can then keep and use four parallel configurations and results, and this will ease use for applications with high data throughput or for multiple modules using the ADC independently. It is possible to use DMA to move ADC results directly to memory or peripherals when conversions are done.

Both internal and external reference voltages can be used. An integrated temperature sensor is available for use with the ADC. The output from the DAC, $V_{CC}/10$ and the bandgap voltage can also be measured by the ADC.

The ADC has a compare function for accurate monitoring of user defined thresholds with minimum software intervention required.

Figure 28-1. ADC overview.



28.3 Input Sources

Input sources are the voltage inputs that the ADC can measure and convert. Four types of measurements can be selected:

- Differential input
- Differential input with gain
- Single-ended input
- Internal input

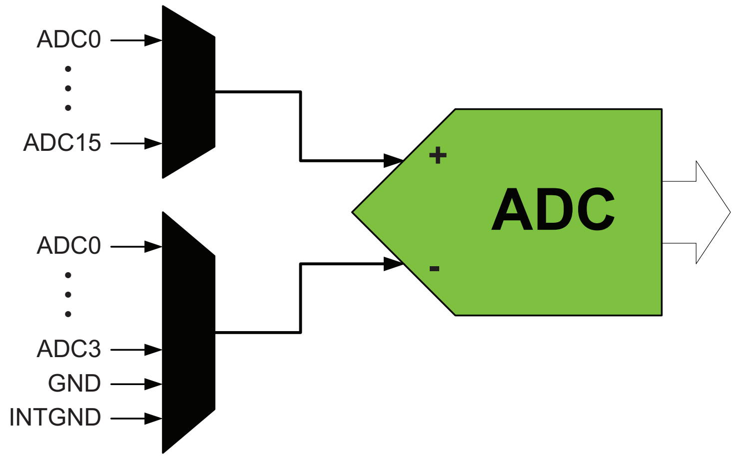
The input pins are used for single-ended and differential input, while the internal inputs are directly available inside the device. In devices with two ADCs, PORTA pins can be input to ADCA and PORTB pins can be input to ADCB. For AVR XMEGA devices with only one ADC, input pins may be available for ADCA on both PORTA and PORTB.

The ADC is differential, and so for single-ended measurements the negative input is connected to a fixed internal value. The four types of measurements and their corresponding input options are shown in Figure 28-2 on page 341 to Figure 28-6 on page 343.

28.3.1 Differential Input

When differential input is enabled, all input pins can be selected as positive input, and input pins 0 to 3 can be selected as negative input. The ADC must be in signed mode when differential input is used.

Figure 28-2. Differential measurement without gain.

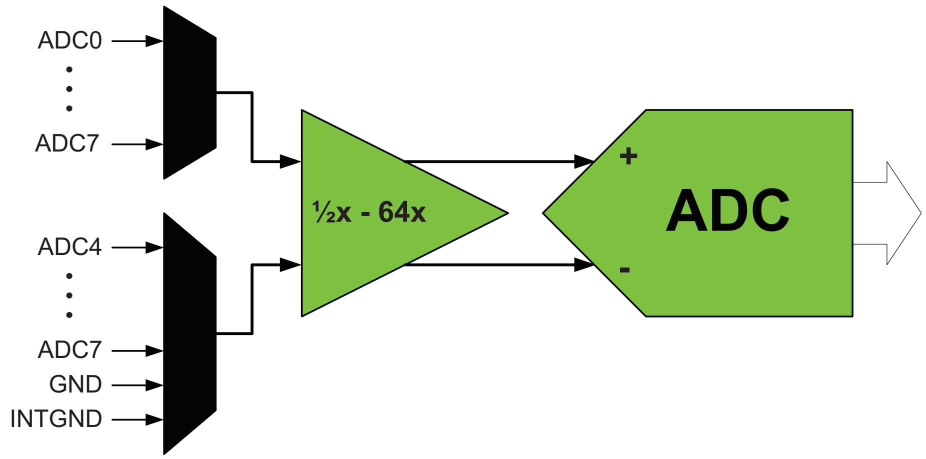


28.3.2 Differential Input with Gain

When differential input with gain is enabled, all input pins can be selected as positive input, and input pins 4 to 7 can be selected as negative input. When the gain stage is used, the differential input is first sampled and amplified by the gain stage before the result is fed into the ADC. The ADC must be in signed mode when differential input with gain is used.

The gain is selectable to 1/2x, 1x, 2x, 4x, 8x, 16x, 32x, and 64x gain.

Figure 28-3. Differential measurement with gain.

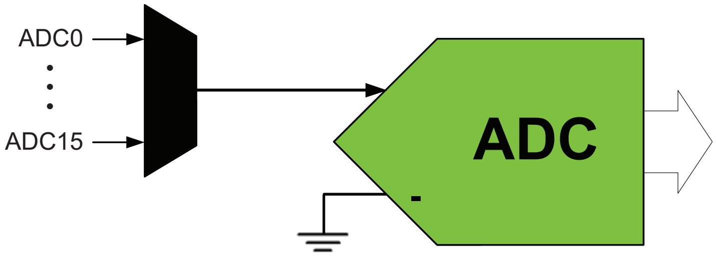


28.3.3 Single-ended Input

For single-ended measurements, all input pins can be used as inputs. Single-ended measurements can be done in both signed and unsigned mode.

The negative input is connected to internal ground in signed mode.

Figure 28-4. Single-ended measurement in signed mode.

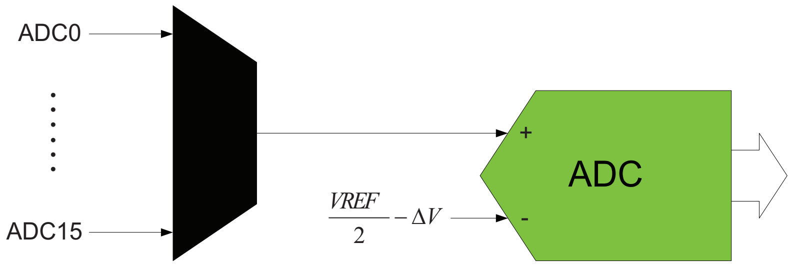


In unsigned mode, the negative input is connected to half of the voltage reference (VREF) voltage minus a fixed offset. The nominal value for the offset is:

$$\Delta V = VREF \times 0.05$$

Since the ADC is differential, the input range is VREF to zero for the positive single-ended input. The offset enables the ADC to measure zero crossing in unsigned mode, and allows for calibration of any positive offset when the internal ground in the device is higher than the external ground. See Figure 28-11 on page 345 for details.

Figure 28-5. Single-ended measurement in unsigned mode.



28.3.4 Internal Inputs

These internal signals can be measured or used by the ADC.

- Temperature sensor
- Bandgap voltage
- V_{CC} scaled
- DAC output
- Pad and Internal Ground

The temperature sensor gives an output voltage that increases linearly with the internal temperature of the device. One or more calibration points are needed to compute the temperature from a measurement of the temperature sensor. The temperature sensor is calibrated at one point in production test, and the result is stored to TEMPESENSE0 and TEMPSENSE1 in the production signature row. For more calibration condition details, refer to the device datasheet.

The bandgap voltage is an accurate internal voltage reference.

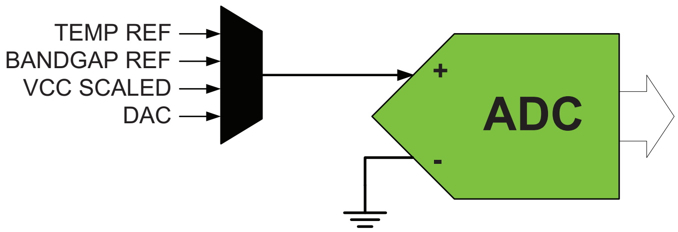
V_{CC} can be measured directly by scaling it down by a factor of 10 before the ADC input. Thus, a V_{CC} of 1.8V will be measured as 0.18V, and V_{CC} of 3.6V will be measured as 0.36V. This enables easy measurement of the V_{CC} voltage.

The internal signals need to be enabled before they can be measured. Refer to their manual sections for Bandgap and DAC for details of how to enable these. The sample rate for the internal signals is lower than that of the ADC. Refer to the ADC characteristics in the device datasheets for details.

For differential measurement Pad Ground (Gnd) and Internal Gnd can be selected as negative input. Pad Gnd is the gnd level on the pin and identical or very close to the external gnd. Internal Gnd is the internal device gnd level.

Internal Gnd is used as the negative input when other internal signals are measured in single-ended signed mode.

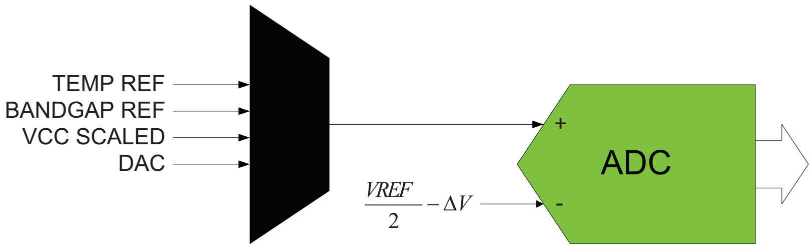
Figure 28-6. Internal measurements in single-ended signed mode.



To measure the internal signals in unsigned mode, the negative input is connected to a fixed value given by the formula below, which is half of the voltage reference (VREF) minus a fixed offset, as it is for single-ended unsigned input. Refer to Figure 28-11 on page 345 for details.

$VINN = VREF/2 - \Delta V$

Figure 28-7. Internal measurements in unsigned mode.



28.4 ADC Channels

To facilitate the maximum utilization of the ADC, it has four separate pairs of MUX control registers with corresponding result registers. Each pair forms an ADC channel. See Figure 28-1 on page 340. The ADC can then keep and use four parallel configurations of input sources and triggers. Each channel has dedicated result register, events and interrupts, and DMA triggers.

As an example of the ADC channel usage, one channel can be setup for single-ended measurements triggered by an event channel, the second channel can measure a differential input using a different event, and the two last channels can measure two other input sources started by the application software.

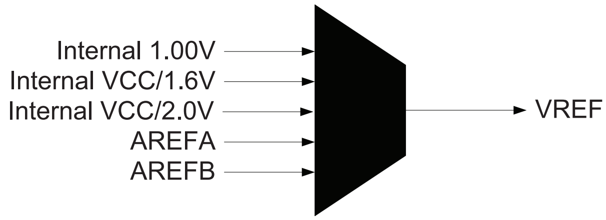
All the ADC channels use the same ADC pipeline for the conversions, and the pipeline enables a new conversion to be started for each ADC clock cycle. This means that multiple ADC measurements from different channels can be converted simultaneously and independently. The channels' result registers are individually updated and are unaffected by conversions on other channels. This can help reduce software complexity by allowing different software modules to start conversions and read conversion results fully independently of each other.

28.5 Voltage Reference Selection

The following voltages can be used as the reference voltage (VREF) for the ADC:

- Accurate internal 1.00V voltage generated from the bandgap
- Internal V_{CC}/1.6V voltage
- Internal V_{CC}/2V voltage
- External voltage applied to AREF pin on PORTA
- External voltage applied to AREF pin on PORTB

Figure 28-8. ADC voltage reference selection



28.6 Conversion Result

The result of the analog-to-digital conversion is written to the corresponding channel result registers. The ADC is either in signed or unsigned mode. This setting is global for the ADC and all ADC channels.

In signed mode, negative and positive results are generated. Signed mode must be used when any of the ADC channels are set up for differential measurements. In unsigned mode, only single-ended or internal signals can be measured. With 12-bit resolution, the TOP value of a signed result is 2047, and the results will be in the range -2048 to +2047 (0xF800 - 0x07FF).

The ADC transfer function can be written as:

$$\mathrm{RES} = \frac{\mathrm{VINP - VINN}}{\mathrm{VREF}} \cdot \mathrm{GAIN} \cdot (\mathrm{TOP} + 1)$$

VINP and VINN are the positive and negative inputs to the ADC.

For differential measurements, GAIN is 1/2 to 64. For single-ended and internal measurements, GAIN is always 1 and VINP is the internal ground.

In unsigned mode, only positive results are generated. The TOP value of an unsigned result is 4095, and the results will be in the range 0 to +4095 (0x0 - 0x0FFF).

The ADC transfer functions can be written as:

$$\mathrm{RES} = \frac{\mathrm{VINP} - (-\Delta V)}{\mathrm{VREF}} \cdot (\mathrm{TOP} + 1)$$

VINP is the single-ended or internal input.

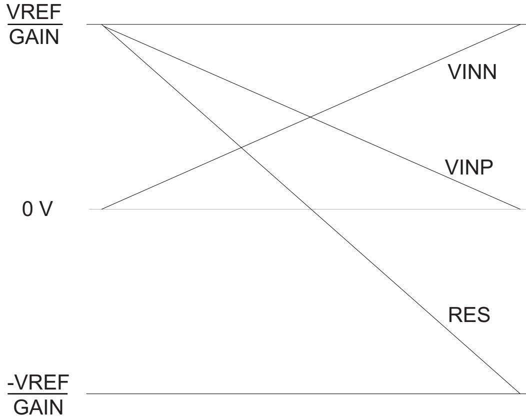
The ADC can be configured to generate either an 8-bit or a 12-bit result. A result with lower resolution will be available faster. See the "ADC Clock and Conversion Timing" on page 346 for a description on the propagation delay.

The result registers are 16 bits wide, and data are stored as right adjusted 16-bit values. Right adjusted means that the eight least-significant bits (lsb) are found in the low byte. A 12-bit result can be represented either left or right adjusted. Left adjusted means that the eight most-significant bits (msb) are found in the high byte.

When the ADC is in signed mode, the msb represents the sign bit. In 12-bit right adjusted mode, the sign bit (bit 11) is padded to bits 12-15 to create a signed 16-bit number directly. In 8-bit mode, the sign bit (bit 7) is padded to the entire high byte.

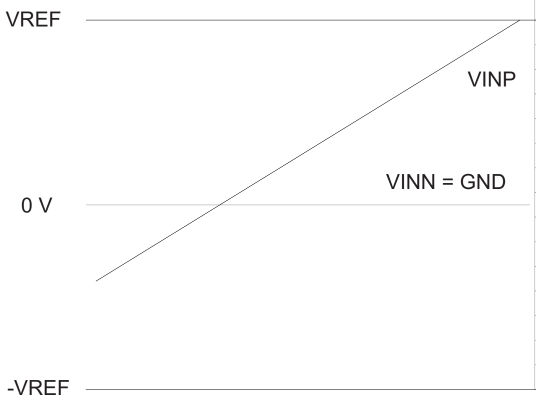
Figure 28-9 on page 345 to Figure 28-11 on page 345 show the different input options, the signal input range, and the result representation with 12-bit right adjusted mode.

Figure 28-9. Signed differential input (with gain), input range, and result representation.



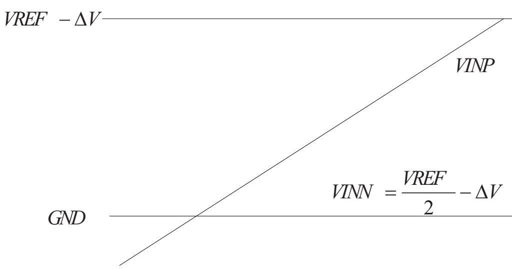
Dec	Hex	Binary	16-bit result register
2047	7FF	0111 1111 1111	0000 0111 1111 1111
2046	7FE	0111 1111 1110	0000 0111 1111 1110
2045	7FD	0111 1111 1101	0000 0111 1111 1101
...	...	...	...
3	3	0000 0000 0011	0000 0000 0000 0011
2	2	0000 0000 0010	0000 0000 0000 0010
1	1	0000 0000 0001	0000 0000 0000 0001
0	0	0000 0000 0000	0000 0000 0000 0000
-1	FFF	1111 1111 1111	1111 1111 1111 1111
-2	FFE	1111 1111 1110	1111 1111 1111 1110
...	...	...	...
-2045	803	1000 0000 0011	1111 1000 0000 0011
-2046	802	1000 0000 0010	1111 1000 0000 0010
-2047	801	1000 0000 0001	1111 1000 0000 0001
-2048	800	1000 0000 0000	1111 1000 0000 0000

Figure 28-10.Signed single-ended and internal input, input range, and result representation.



Dec	Hex	Binary	16-bit result register
2047	7FF	0111 1111 1111	0000 0111 1111 1111
2046	7FE	0111 1111 1110	0000 0111 1111 1110
2045	7FD	0111 1111 1101	0000 0111 1111 1101
...	...	...	...
3	3	0000 0000 0011	0000 0000 0000 0011
2	2	0000 0000 0010	0000 0000 0000 0010
1	1	0000 0000 0001	0000 0000 0000 0001
0	0	0000 0000 0000	0000 0000 0000 0000
-1	FFF	1111 1111 1111	1111 1111 1111 1111
-2	FFE	1111 1111 1110	1111 1111 1111 1110
...	...	...	...
-2045	803	1000 0000 0011	1111 1000 0000 0011
-2046	802	1000 0000 0010	1111 1000 0000 0010
-2047	801	1000 0000 0001	1111 1000 0000 0001
-2048	800	1000 0000 0000	1111 1000 0000 0000

Figure 28-11.Unsigned single-ended and internal input, input range, and result representation.



Dec	Hex	Binary	16-bit result register
4095	FFF	1111 1111 1111	0000 1111 1111 1111
4094	FFE	1111 1111 1110	0000 1111 1111 1110
4093	FFD	1111 1111 1101	0000 1111 1111 1101
...	...	...	...
203	0CB	0000 1100 1011	0000 0000 1100 1011
202	0CA	0000 1100 1010	0000 0000 1100 1010
201	0C9	0000 1100 1001	0000 0000 1100 1001
200	0C8	0000 1100 1000	0000 0000 1100 1000
...			
0	0	0000 0000 0000	0000 0000 0000 0000

28.7 Compare Function

The ADC has a built-in 12-bit compare function. The ADC compare register can hold a 12-bit value that represents a threshold voltage. Each ADC channel can be configured to automatically compare its result with this compare value to give an interrupt or event only when the result is above or below the threshold.

All four ADC channels share the same compare register.

28.8 Starting a Conversion

Before a conversion is started, the input source must be selected for one or more ADC channels. An ADC conversion for a channel can be started either by the application software writing to the start conversion bit for the channel or from any events in the event system. It is possible to write the start conversion bit for several channels at the same time, or use one event to trigger conversions on several channels at the same time. This makes it possible to scan several or all channels from one event. The scan will start from the lowest channel number.

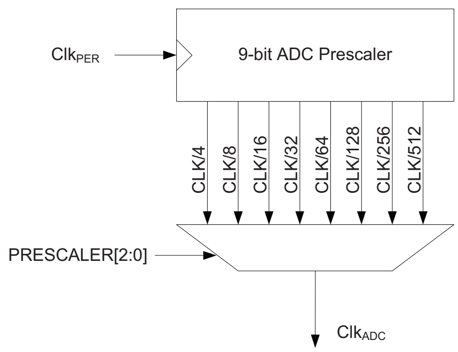
28.8.1 Input Source Scan

For ADC Channel 0 it is possible to select a range of consecutive input sources that is automatically scanned and measured when a conversion is started. This is done by setting the first (lowest) positive ADC channel input using the MUX control register, and a number of consecutive positive input sources. When a conversion is started, the first selected input source is measured and converted, then the positive input source selection is incremented after each conversion until it reaches the specified number of sources to scan.

28.9 ADC Clock and Conversion Timing

The ADC is clocked from the peripheral clock. The ADC can prescale the peripheral clock to provide an ADC Clock (clk_{ADC}) that matches the application requirements and is within the operating range of the ADC.

Figure 28-12.ADC prescaler.



The maximum ADC sample rate is given by the he ADC clock frequency (f_{ADC}). The ADC can sample a new measurement on every ADC clock cycle.

$$\text{Sample Rate} = f_{ADC}$$

The propagation delay of an ADC measurement is given by:

$$\text{Propagation Delay} = \frac{1 + \frac{RESOLUTION}{2} + GAIN}{f_{ADC}}$$

RESOLUTION is the resolution, 8 or 12 bits. The propagation delay will increase by one extra ADC clock cycle if the gain stage (GAIN) is used.

The propagation delay is longer than one ADC clock cycle, but the pipelined design means that the sample rate is limited not by the propagation delay, but by the ADC clock rate.

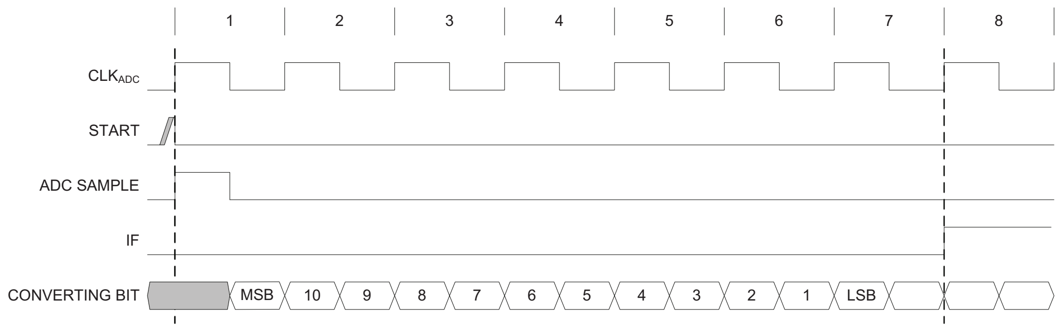
The most-significant bit (msb) of the result is converted first, and the rest of the bits are converted during the next three (for 8-bit results) or five (for 12-bit results) ADC clock cycles. Converting one bit takes a half ADC clock period. During the last cycle, the result is prepared before the interrupt flag is set and the result is available in the result register for readout.

28.9.1 Single Conversion without Gain

Figure 28-13 on page 347 shows the ADC timing for a single conversion without gain. The writing of the start conversion bit, or the event triggering the conversion (START), must occur at least one peripheral clock cycle before the ADC clock cycle on which the conversion starts (indicated with the grey slope of the START trigger).

The input source is sampled in the first half of the first cycle.

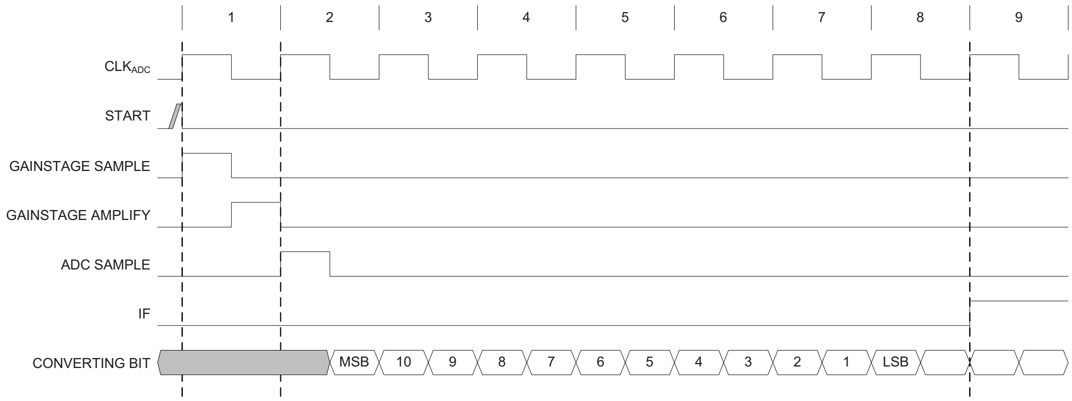
Figure 28-13.ADC timing for one single conversion without gain.



28.9.2 Single Conversion with Gain

Figure 28-14 on page 347 shows the ADC timing for one single conversion with gain. As seen in the "Overview" on page 339, the gain stage is placed prior to the actual ADC. The gain stage will sample and amplify the input source before the ADC samples it, and converts the amplified value. Compared to a single conversion without gain, this adds one ADC clock cycle (between START and ADC sample) for the gain stage sample and amplify. The sample time for the gain stage is one half ADC clock cycle.

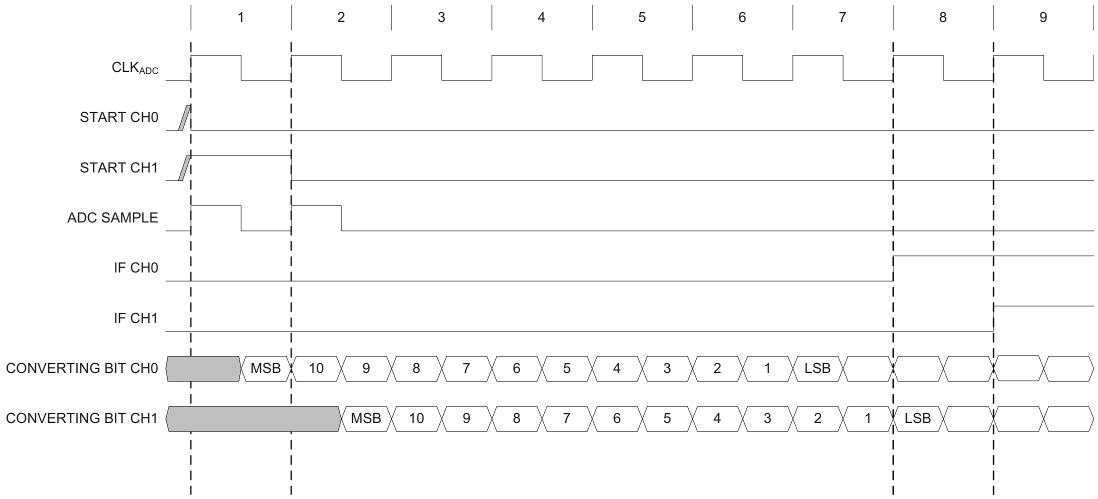
Figure 28-14.ADC timing for one single conversion with gain.



28.9.3 Single Conversions on Two ADC Channels

Figure 28-15 on page 348 shows the ADC timing for single conversions on two channels. The pipelined design enables the second conversion to start on the next ADC clock cycle after the first conversion has started. In this example, both conversions take place at the same time, but the conversion on ADC channel 1(CH1) does not start until the ADC samples and performs conversion on the msb on channel 0 (CH0).

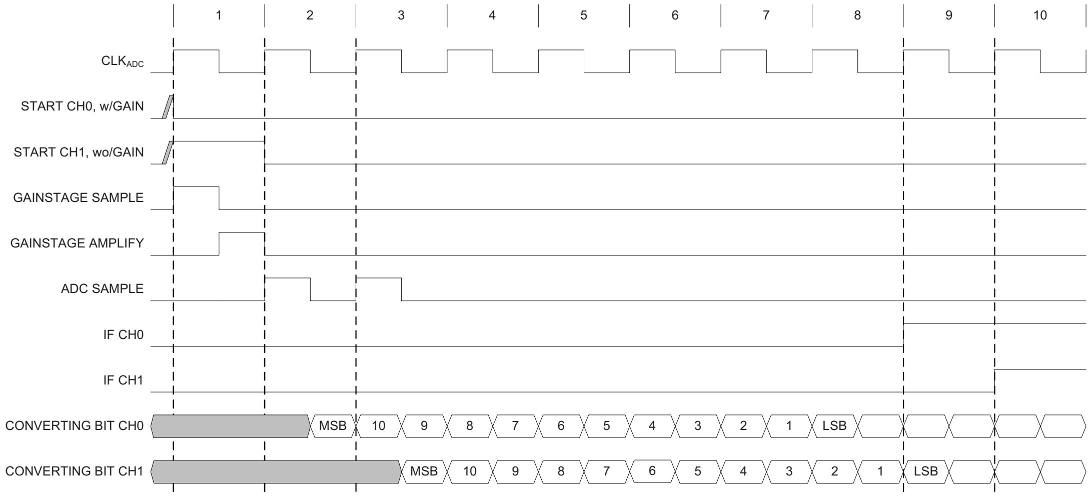
Figure 28-15.ADC timing for single conversions on two ADC channels.



28.9.4 Single Conversions on Two ADC Channels, CH0 with Gain

Figure 28-16 on page 348 shows the conversion timing for single conversions on two ADC channels where ADC channel 0 uses the gain stage. As the gain stage introduces one addition cycle for the gain sample and amplify, the sample for ADC channel 1 is also delayed one ADC clock cycle, until the ADC sample and msb conversion is done for ADC channel 0.

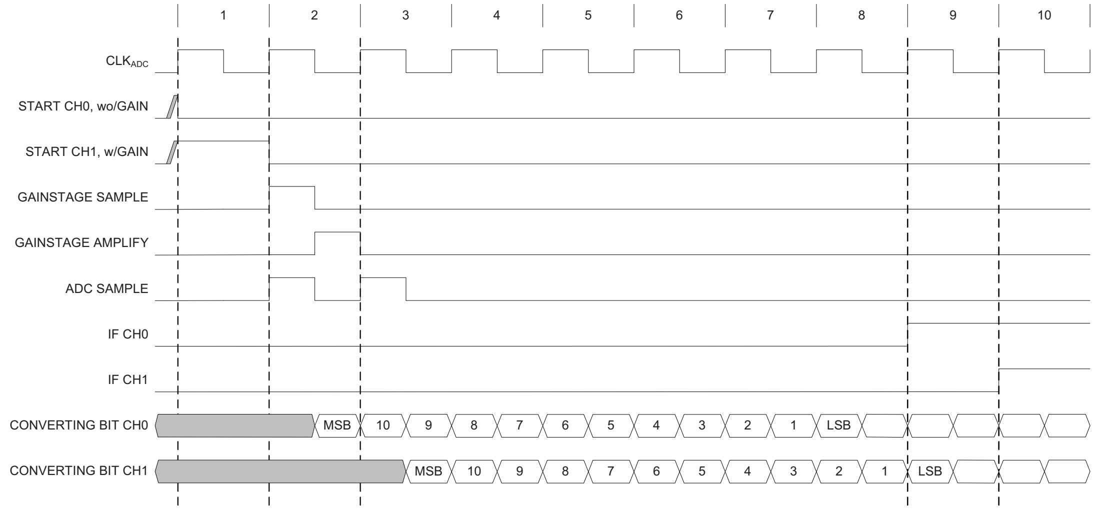
Figure 28-16.ADC timing for single conversion on two ADC channels, CH0 with gain.



28.9.5 Single Conversions on Two ADC Channels, CH1 with Gain

Figure 28-17 on page 349 shows the conversion timing for single conversions on two ADC channels where ADC channel 1 uses the gain stage.

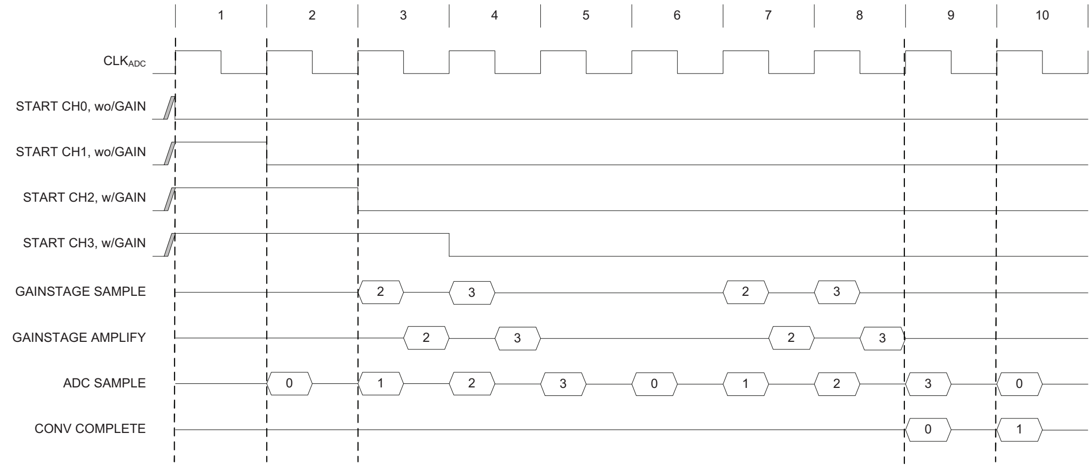
Figure 28-17.ADC timing for single conversion on two ADC channels, CH1 with gain.



28.9.6 Free Running Mode on Two ADC Channels with Gain

Figure 28-18 on page 349 shows the conversion timing for all four ADC channels in free running mode, CH0 and CH1 without gain and CH2 and CH3 with gain. When set up in free running mode, an ADC channel will continuously sample and do new conversions. In this example, all ADC channels are triggered at the same time, and each ADC channel samples and start converting as soon as the previous ADC channel is done with its sample and msb conversion. After four ADC clock cycles, all ADC channels have done the first sample and started the first conversion, and each ADC channels can then do the sample conversion start for their second conversion. After eight (for 12-bit mode) ADC clock cycles, the first conversion is done for ADC channel 0, and the results for the rest of the ADC channels are available in subsequent ADC clock cycles. After the next clock cycle (in cycle 10), the result from the second ADC channel is done and available, and so on. In this mode, up to eight conversions are ongoing at the same time.

Figure 28-18.ADC timing for free running mode.



28.10 ADC Input Model

The voltage input must charge the sample and hold (S/H) capacitor in the ADC in order to achieve maximum accuracy. Seen externally, the ADC input consists of an input resistance ($R_{in} = R_{channel} + R_{switch}$) and the S/H capacitor (C_{sample}). Figure 28-19 on page 350 and Figure 28-20 on page 350 show the ADC input channels.

Figure 28-19.ADC input for single-ended measurements.

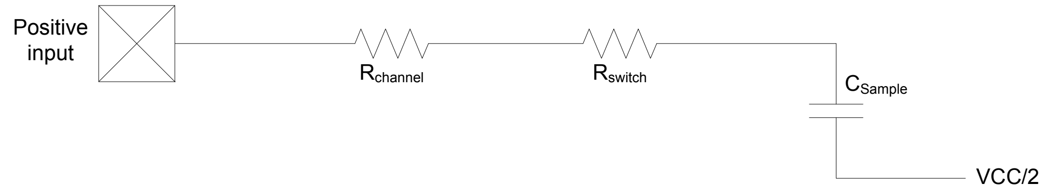
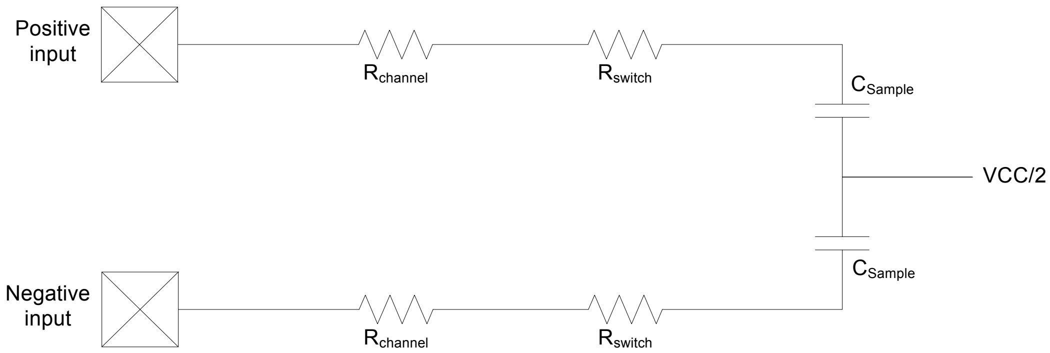


Figure 28-20.ADC input for differential measurements and differential measurements with gain.



In order to achieve *n* bits of accuracy, the source output resistance, R_{source}, must be less than the ADC input resistance on a pin:

$$R_{source} \le \frac{T_s}{C_{sample} \cdot \ln(2^{n+1})} - R_{channel} - R_{switch}$$

where the ADC sample time, T_S is one-half the ADC clock cycle given by:

$$T_s \le \frac{1}{2 \cdot f_{ADC}}$$

For details on $R_{channel}$, R_{switch}, and C_{sample}, refer to the ADC and ADC gain stage electrical characteristic in the device datasheet.

28.10.1 Gain Stage Impedance mode

To support applications with very high source output resistance, the gain stage has a high impedance mode. In this mode the charge on the S/H capacitor is kept after each sample, and the S/H capacitor can be fully charged by doing multiple samples on the same input channel. When low impedance mode is used, the S/H capacitor charge is flushed after each sample.

28.11 DMA Transfer

The DMA controller can be used to transfer ADC conversion results to memory or other peripherals. A new conversion result for any of the ADC channels can trigger a DMA transaction for one or several ADC channels. Refer to "DMAC - Direct Memory Access Controller" on page 53 for more details on DMA transfers.

28.12 Interrupts and Events

The ADC can generate interrupt requests and events. Each ADC channel has individual interrupt settings and interrupt vectors. Interrupt requests and events can be generated when an ADC conversion is complete or when an ADC measurement is above or below the ADC compare register value.

28.13 Calibration

The ADC has built-in linearity calibration. The value from the production test calibration must be loaded from the signature row and into the ADC calibration register from software to achieve specified accuracy. User calibration of the linearity is not needed, hence not possible. Offset and gain calibration must be done in software.

28.14 Channel Priority

Since the peripheral clock is faster than the ADC clock, it is possible to set the start conversion bit for several ADC channels within the same ADC clock period. Events may also trigger conversions on several ADC channels and give the same scenario. In this case, the ADC channel with the lowest number will be prioritized. This is shown the timing diagrams in "ADC Clock and Conversion Timing" on page 346.

28.15 Synchronous Sampling

The ADC can be configured to do synchronous sampling in three different ways.

1. Sample two input channels at the same time
2. Sample two ADCs at the same time
3. Sample on external trigger

28.15.1 Synchronous sampling of two ADC inputs

The ADC supports sampling of two input channels at the same time. This is achieved by setting up channel n to not use gain and channel n+1 to use 1x gain. The converted result from the channel using gain will be ready one ADC clock cycle after the other channel. See "Single Conversions on Two ADC Channels, CH1 with Gain" on page 348 for detailed timing diagram.

28.15.2 Synchronous sampling on event

Starting an ADC conversion can cause an unknown delay between the start trigger or event and the actual conversion start, since conversions of higher priority ADC channels may be pending, or since the peripheral clock is faster than the ADC clock. To start an ADC conversion immediately on an incoming event, it is possible to flush the ADC of all measurements, reset the ADC clock, and start the conversion at the next peripheral clock cycle (which then will also be the next ADC clock cycle). If this is done, all ongoing conversions in the ADC pipeline will be lost.

The ADC can be flushed from software, or an incoming event can do this automatically. When this function is used, the time between each conversion start trigger must be longer than the ADC propagation delay to ensure that one conversion is finished before the ADC pipeline is flushed and the next conversion is started.

It is also important to clear pending events or start ADC conversion commands before doing a flush. If not, pending conversions will start immediately after the flush.

28.15.3 Synchronous sampling of two ADCs

In devices with two ADC peripherals, it is possible to start two ADC samples synchronously in the two ADCs by using the same event channel to trigger both ADC.

28.16 Register Description – ADC

28.16.1 CTRLA – Control register A

Bit	7	6	5	4	3	2	1	0
+0x00	DMASEL[1:0]		CHSTART[3:0]				FLUSH	ENABLE
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:6 – DMASEL[1:0]: DMA Request Selection**
  To allow one DMA channel to serve more than one ADC channel, the DMA request from the channels can be combined into a common DMA request. See Table 28-1 for details.

Table 28-1. DMA request selection.

DMASEL[1:0]	Group configuration	Description
00	OFF	No combined DMA request
01	CH01	Common request for ADC channels 0 and 1
10	CH012	Common request for ADC channels 0, 1, and 2
11	CH0123	Common request for ADC channels 0, 1, 2, and 3

- **Bit 5:2 – CHSTART[3:0]: Channel Start Single Conversion**
  Setting any of these bits will start a conversion on the corresponding ADC channel. Setting several bits at the same time will start conversions on all selected ADC channels, starting with the channel with the lowest number. These bits are cleared by hardware when the conversion has started.

- **Bit 1 – FLUSH: Pipeline Flush:**
  Setting this bit will flush the ADC pipeline. When this is done, the ADC clock is restarted on the next peripheral clock edge, and all conversions in progress are aborted and lost.

  After the flush and the ADC clock restart, the ADC will resume where it left off; i.e., if a channel sweep was in progress or any conversions were pending, these will enter the ADC pipeline and complete.

- **Bit 0 – ENABLE: Enable**
  Setting this bit enables the ADC.

28.16.2 CTRLB – ADC Control register B

Bit	7	6	5	4	3	2	1	0
+0x01	IMPMODE	CURRLIMIT[1:0]		CONVMODE	FREERUN	RESOLUTION[1:0]		–
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7 – IMPMODE: Gain Stage Impedance Mode**
  This bit controls the impedance mode of the gain stage.

  See GAIN setting in ADC channel register description for more information ("CTRL – Channel Control register" on page 359).

Table 28-2. Gain stage impedance mode.

IMPMODE	Group configuration[(1)]	Description
0	HIGHIMP	For high-impedance sources; charge will remain on input
1	LOWIMP	For low impedance sources

Note: 1. This is either high or low impedance. While high impedance mode is only available for 1x, 2x, 4x, and 8x, for all other it will be forced to low impedance mode. See Table 28-10 on page 359.

- **Bit 6:5 – CURRLIMIT[1:0]: Current Limitation**
  These bits can be used to limit the current consumption of the ADC by reducing the maximum ADC sample rate. The available settings are shown in Table 28-3 on page 353. The indicated current limitations are nominal values. Refer to the device datasheet for actual current limitation for each setting.

Table 28-3. ADC current limitations.

CURRLIMIT[1:0]	Group configuration	Description
00	NO	No limit
01	LOW	Low current limit, max. sampling rate 1.5MSPS
10	MED	Medium current limit, max. sampling rate 1MSPS
11	HIGH	High current limit, max. sampling rate 0.5MSPS

- **Bit 4 – CONVMODE: Conversion Mode**
  This bit controls whether the ADC will work in signed or unsigned mode. By default, this bit is cleared and the ADC is configured for unsigned mode. When this bit is set, the ADC is configured for signed mode.

- **Bit 3 – FREERUN: Free Running Mode**
  When the bit is set to one, the ADC is in free running mode and the ADC channels defined in the EVCTRL register are swept repeatedly.

- **Bit 2:1 – RESOLUTION[1:0]: Conversion Result Resolution**
  These bits define whether the ADC completes the conversion at 12- or 8-bit result resolution. They also define whether the 12-bit result is left or right adjusted within the 16-bit result registers. See Table 28-4 on page 353 for possible settings.

Table 28-4. ADC conversion result resolution.

RESOLUTION[1:0]	Group configuration	Description
00	12BIT	12-bit result, right adjusted
01		Reserved
10	8BIT	8-bit result, right adjusted
11	LEFT12BIT	12-bit result, left adjusted

- **Bit 0 – Reserved**
  This bit is unused and reserved for future use. For compatibility with future devices, always write this bit to zero when this register is written.

28.16.3 REFCTRL – Reference Control register

Bit	7	6	5	4	3	2	1	0
+0x02	–	REFSEL[2:0]			–	–	BANDGAP	TEMPREF
Read/Write	R	R/W	R/W	R/W	R	R	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7 – Reserved**
  This bit is unused and reserved for future use. For compatibility with future devices, always write this bit to zero when this register is written.
- **Bits 6:4 – REFSEL[2:0]: Reference Selection**
  These bits selects the reference for the ADC according to Table 28-5 on page 354.

Table 28-5. ADC reference selection.

REFSEL[2:0]	Group configuration	Description
000	INT1V	10/11 of bandgap (1.0V)
001	INTVCC	$V_{CC}/1.6$
010	AREFA	External reference from AREF pin on PORT A
011	AREFB	External reference from AREF pin on PORT B
100	INTVCC2	$V_{CC}/2$
101 - 111		Reserved

- **Bit 3:2 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.
- **Bit 1 – BANDGAP: Bandgap Enable**
  Setting this bit enables the bandgap for ADC measurement. Note that if any other functions are already using the bandgap, this bit does not need to be set when the internal 1.00V reference is used for another ADC, the DAC or if the brownout detector is enabled.
- **Bit 0 – TEMPREF: Temperature Reference Enable**
  Setting this bit enables the temperature sensor for ADC measurement.

28.16.4 EVCTRL – Event Control register

Bit	7	6	5	4	3	2	1	0
+0x03	SWEEP[1:0]		EVSEL[2:0]			EVACT[2:0]		
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:6 – SWEEP[1:0]: Channel Sweep**
  These bits control which ADC channels are included in a channel sweep triggered by the event system or when in free running mode. See Table 28-6 on page 355.

Table 28-6. ADC channel select.

SWEEP[1:0]	Group configuration	Active ADC channels for channel sweep
00	0	Only ADC channel 0
01	01	ADC channels 0 and 1
10	012	ADC channels 0, 1, and 2
11	0123	ADC channels 0, 1, 2, and 3

- **Bit 5:3 – EVSEL[2:0]: Event Channel Input Select**
  These bits select which event channel will trigger which ADC channel. Each setting defines a group of event channels, where the event channel with the lowest number will trigger ADC channel 0, the next event channel will trigger ADC channel 1, and so on. See Table 28-7 on page 355.

Table 28-7. ADC event channel select.

EVSEL[2:0]	Group configuration	Selected event lines
000	0123	Event channel 0, 1, 2, and 3 as selected inputs
001	1234	Event channel 1, 2, 3, and 4 as selected inputs
010	2345	Event channel 2, 3, 4, and 5 as selected inputs
011	3456	Event channel 3, 4, 5, and 6 as selected inputs
100	4567	Event channel 4, 5, 6, and 7 as selected inputs
101	567	Event channel 5, 6, and 7 as selected inputs
110	67	Event channel 6and7 as selected inputs
111	7	Event channel 7 as selected input

- **Bit 2:0 – EVACT[2:0]: Event Mode**
  These bits select and limit how many of the selected event input channel are used, and also further limit the ADC channels triggers. They also define more special event triggers as defined in Table 28-8 on page 355.

Table 28-8. ADC event mode select.

EVACT[2:0]	Group configuration	Selected input operation mode
000	NONE	No event inputs
001	CH0	Event channel with the lowest number defined by EVSEL triggers conversion on ADC channel 0
010	CH01	Event channels with the two lowest numbers defined by EVSEL trigger conversions on ADC channels 0 and 1, respectively
011	CH012	Event channels with the three lowest numbers defined by EVSEL trigger conversions on ADC channels 0, 1, and 2, respectively
100	CH0123	Event channels defined by EVSEL trigger conversion on ADC channels 0, 1, 2, and 3, respectively

EVACT[2:0]	Group configuration	Selected input operation mode
101	SWEEP	One sweep of all ADC channels defined by SWEEP on incoming event channel with the lowest number defined by EVSEL
110	SYNCSWEEP	One sweep of all active ADC channels defined by SWEEP on incoming event channel with the lowest number defined by EVSE. In addition the ADC is flushed and restarted for accurate timing
111		Reserved

28.16.5 PRESCALER – Clock Prescaler register

Bit	7	6	5	4	3	2	1	0
+0x04	–	–	–	–	–	PRESCALER[2:0]		
Read/Write	R	R	R	R	R	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:3 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.

- **Bit 2:0 – PRESCALER[2:0]: Prescaler Configuration**
  These bits define the ADC clock relative to the peripheral clock according to Table 28-9 on page 356.

Table 28-9. ADC prescaler settings.

PRESCALER[2:0]	Group configuration	Peripheral clock division factor
000	DIV4	4
001	DIV8	8
010	DIV16	16
011	DIV32	32
100	DIV64	64
101	DIV128	128
110	DIV256	256
111	DIV512	512

28.16.6 INTFLAGS – Interrupt Flag register

Bit	7	6	5	4	3	2	1	0
+0x06	–	–	–	–	CH[3:0]IF			
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:4 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.

- **Bit 3:0 – CH[3:0]IF: Interrupt Flags**
  These flags are set when the ADC conversion is complete for the corresponding ADC channel. If an ADC channel is configured for compare mode, the corresponding flag will be set if the compare condition is met. CHnIF is automatically cleared when the ADC channel n interrupt vector is executed. The flag can also be cleared by writing a one to its bit location.

28.16.7 TEMP – Temporary register

Bit	7	6	5	4	3	2	1	0
+0x07	TEMP[7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:0 – TEMP[7:0]: Temporary bits**
  This register is used when reading 16-bit registers in the ADC controller. The high byte of the 16-bit register is stored here when the low byte is read by the CPU. This register can also be read and written from the user software.

  For more details on 16-bit register access, refer to "The combined EIND + Z register." on page 12.

28.16.8 CALL – Calibration Value register

The CALL and CALH register pair hold the 12-bit calibration value. The ADC pipeline is calibrated during production programming, and the calibration value must be read from the signature row and written to the CAL register from software.

Bit	7	6	5	4	3	2	1	0
+0x0C	CAL[7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:0 – CAL[7:0]: ADC Calibration value**
  These are the eight lsbs of the 12-bit CAL value.

28.16.9 CALH – Calibration Value register

Bit	7	6	5	4	3	2	1	0
+0x0D	–	–	–	–	CAL[11:8]			
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 3:0 – CAL[11:8]: Calibration value**
  These are the four msbs of the 12-bit CAL value.

28.16.10 CHnRESH – Channel n Result register High

The CHnRESL and CHnRESH register pair represents the 16-bit value, CHnRES. For details on reading 16-bit registers, refer to "The combined EIND + Z register." on page 12.

Bit	7	6	5	4	3	2	1	0
12-bit, left	CHRES[11:4]							
12-bit, right	–	–	–	–	CHRES[11:8]			
8-bit	–	–	–	–	–	–	–	–
Read/Write	R	R	R	R	R	R	R	R
Initial Value	0	0	0	0	0	0	0	0

28.16.10.1 12-bit Mode, Left Adjusted

- **Bit 7:0 – CHRES[11:4]: Channel Result high byte**
  These are the eight msbs of the 12-bit ADC result.

28.16.10.2 12-bit Mode, Right Adjusted

- **Bit 7:4 – Reserved**
  These bits will in practice be the extension of the sign bit, CHRES11, when the ADC works in differential mode, and set to zero when the ADC works in signed mode.

- **Bit 3:0 – CHRES[11:8]: Channel Result high byte**
  These are the four msbs of the 12-bit ADC result.

28.16.10.3 8-bit Mode

- **Bit 7:0 – Reserved**
  These bits will in practice be the extension of the sign bit, CHRES7, when the ADC works in signed mode, and set to zero when the ADC works in single-ended mode.

28.16.11 CHnRESL – Channel n Result register Low

Bit	7	6	5	4	3	2	1	0
12-/8-bit, right	CHRES[7:0]							
12-bit, left	CHRES[3:0]				–	–	–	–
Read/Write	R	R	R	R	R	R	R	R
Initial Value	0	0	0	0	0	0	0	0

28.16.11.1 12-/8-bit Mode

- **Bit 7:0 – CHRES[7:0]: Channel Result low byte**
  These are the eight lsbs of the ADC result.

28.16.11.2 12-bit Mode, Left Adjusted

- **Bit 7:4 – CHRES[3:0]: Channel Result low byte**
  These are the four lsbs of the 12-bit ADC result.

- **Bit 3:0 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.

28.16.12 CMPH – Compare register High

The CMPH and CMPL register pair represents the 16-bit value, CMP. For details on reading and writing 16-bit registers, refer to "The combined EIND + Z register." on page 12.

Bit	7	6	5	4	3	2	1	0
+0x19	CMP[15:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:0 – CMP[15:0]: Compare Value high**
  These are the eight msbs of the 16-bit ADC compare value. In signed mode, the number representation is 2's complement, and the msb is the sign bit.

28.16.13 CMPL – Compare register Low

Bit	7	6	5	4	3	2	1	0
+0x18	CMP[7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:0 – CMP[7:0]: Compare Value Low**
  These are the eight lsbs of the 16-bit ADC compare value. In signed mode, the number representation is 2's complement.

28.17 Register Description – ADC Channel

28.17.1 CTRL – Channel Control register

Bit	7	6	5	4	3	2	1	0
+0x00	START	–	–	GAIN[2:0]			INPUTMODE[1:0]	
Read/Write	R/W	R	R	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7 – START: START Conversion on Channel**
  Setting this bit will start a conversion on the channel. The bit is cleared by hardware when the conversion has started. Setting this bit when it already is set will have no effect. Writing or reading this bit is equivalent to writing the CH[3:0]START bits in "CTRLA – Control register A" on page 352.

- **Bit 6:5 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.

- **Bit 4:2 – GAIN[2:0]: Gain Factor**
  These bits define the gain factor for the ADC gain stage.

See Table 28-10 on page 359. Gain is valid only with certain MUX settings. See "MUXCTRL – ADC Channel MUX Control registers" on page 360.

Table 28-10. ADC gain factor.

GAIN[2:0]	Group configuration	Gain factor
000	1X	1x
001	2X	2x
010	4X	4x
011	8X	8x
100	16X	16x
101	32X	32x
110	64X	64x
111	DIV2	½x

- **Bit 1:0 – INPUTMODE[1:0]: Channel Input Mode**

These bits define the channel mode. Changing input mode will corrupt any data in the pipeline.

Table 28-11. Channel input modes, CONVMODE=0 (unsigned mode).

INPUTMODE[1:0]	Group configuration	Description
00	INTERNAL	Internal positive input signal
01	SINGLEENDED	Single-ended positive input signal
10		Reserved
11		Reserved

Table 28-12. Channel input modes, CONVMODE=1 (signed mode).

INPUTMODE[1:0]	Group configuration	Description
00	INTERNAL	Internal positive input signal
01	SINGLEENDED	Single-ended positive input signal
10	DIFF	Differential input signal
11	DIFFWGAIN	Differential input signal with gain

28.17.2 MUXCTRL – ADC Channel MUX Control registers

The MUXCTRL register defines the input source for the channel.

Bit	7	6	5	4	3	2	1	0
+0x01	–	MUXPOS[3:0]				MUXNEG[2:0]		
Read/Write	R	R/W	R/W	R/W	R/W	R	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7 – Reserved**
  This bit is unused and reserved for future use. For compatibility with future devices, always write this bit to zero when this register is written.

- **Bit 6:3 – MUXPOS[3:0]: MUX Selection on Positive ADC Input**
  These bits define the MUX selection for the positive ADC input. Table 28-13 on page 360 and Table 28-14 on page 361 show the possible input selection for the different input modes.

Table 28-13. Channel input modes, CONVMODE=1 (unsigned mode).

MUXPOS[3:0]	Group configuration	Description
0000	TEMP	Temperature reference
0001	BANDGAP	Bandgap voltage
0010	SCALEDVCC	1/10 scaled V_{CC}
0011	DAC	DAC output
0100-1111		Reserved

Table 28-14. ADC MUXPOS configuration when INPUTMODE[1:0] = 01 (single-ended) or INPUTMODE[1:0] = 10 (differential) is used.

MUXPOS[3:0]	Group configuration	Description
0000	PIN0	ADC0 pin
0001	PIN1	ADC1 pin
0010	PIN2	ADC2 pin
0011	PIN3	ADC3 pin
0100	PIN4	ADC4 pin
0101	PIN5	ADC5 pin
0110	PIN6	ADC6 pin
0111	PIN7	ADC7 pin
1000	PIN8	ADC8 pin
1001	PIN9	ADC9 pin
1010	PIN10	ADC10 pin
1011	PIN11	ADC11 pin
1100	PIN12	ADC12 pin
1101	PIN13	ADC13 pin
1110	PIN14	ADC14 pin
1111	PIN15	ADC15 pin

Table 28-15. ADC MUXPOS configuration when INPUTMODE[1:0] = 11 (differential with gain) is used.

MUXPOS[3:0]	Group configuration	Description
0000	PIN0	ADC0 pin
0001	PIN1	ADC1 pin
0010	PIN2	ADC2 pin
0011	PIN3	ADC3 pin
0100	PIN4	ADC4 pin
0101	PIN5	ADC5 pin
0110	PIN6	ADC6 pin
0111	PIN7	ADC7 pin
1XXX		Reserved

Depending on the device pin count and feature configuration, the actual number of analog input pins may be less than 16. Refer to the device datasheet and pin-out description for details.

- **Bit 2:0 – MUXNEG[2:0]: MUX Selection on Negative ADC Input**
  These bits define the MUX selection for the negative ADC input when differential measurements are done. For internal or single-ended measurements, these bits are not used.

Table 28-16 on page 362 and Table 28-17 on page 362 show the possible input sections.

Table 28-16. ADC MUXNEG configuration, INPUTMODE[1:0] = 10, differential without gain.

MUXNEG[2:0]	Group configuration	Analog Input
000	PIN0	ADC0 pin
001	PIN1	ADC1 pin
010	PIN2	ADC2 pin
011	PIN3	ADC3 pin
100	-	Reserved
101	GND	PAD ground
110	-	Reserved
111	INTGND	Internal ground

Table 28-17. ADC MUXNEG configuration, INPUTMODE[1:0] = 11, differential with gain.

MUXNEG[2:0]	Group configuration	Analog Input
000	PIN4	ADC4 pin
001	PIN5	ADC5 pin
010	PIN6	ADC6 pin
011	PIN7	ADC7 pin
100	INTGND	Internal ground
101	-	Reserved
110	-	Reserved
111	GND	PAD ground

28.17.3 INTCTRL – Channel Interrupt Control registers

Bit	7	6	5	4	3	2	1	0
+0x02	–	–	–	–	INTMODE[1:0}		INTLVL[1:0]	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bits 7:4 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.
- **Bit 3:2 – INTMODE: Interrupt Mode**
  These bits select the interrupt mode for the channel according to Table 28-18.

Table 28-18. ADC channel select.

INTMODE[1:0]	Group configuration	Interrupt mode
00	COMPLETE	Conversion complete
01	BELOW	Compare result below threshold
10		Reserved
11	ABOVE	Compare result above threshold

- **Bits 1:0 – INTLVL[1:0]: Interrupt Priority Level and Enable**
  These bits enable the ADC channel interrupt and select the interrupt level, as described in "Interrupts and Programmable Multilevel Interrupt Controller" on page 131. The enabled interrupt will be triggered for conditions when the IF bit in the INTFLAGS register is set.

28.17.4 INTFLAGS – ADC Channel Interrupt Flag registers

Bit	7	6	5	4	3	2	1	0
+0x03	–	–	–	–	–	–	–	IF
Read/Write	R	R	R	R	R	R	R	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:1 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.

- **Bit 0 – IF: Channel Interrupt Flag**
  The interrupt flag is set when the ADC conversion is complete. If the channel is configured for compare mode, the flag will be set if the compare condition is met. IF is automatically cleared when the ADC channel interrupt vector is executed. The bit can also be cleared by writing a one to the bit location.

28.17.5 RESH – Channel n Result register High

For all result registers and with any ADC result resolution, a signed number is represented in 2's complement form, and the msb represents the sign bit.

The RESL and RESH register pair represents the 16-bit value, ADCRESULT. Reading and writing 16-bit values require special attention. Refer to "The combined EIND + Z register." on page 12 for details.

	Bit	7	6	5	4	3	2	1	0
12-bit, left.		RES[11:4]							
12-bit, right	+0x05	–	–	–	–	RES[11:8]			
8-bit		–	–	–	–	–	–	–	–
	Read/Write	R	R	R	R	R	R	R	R
	Initial Value	0	0	0	0	0	0	0	0

28.17.5.1 12-bit Mode, Left Adjusted

- **Bit 7:0 – RES[11:4]: Channel Result High**
  These are the eight msbs of the 12-bit ADC result.

28.17.5.2 12-bit Mode, Right Adjusted

- **Bit 7:4 – Reserved**
  These bits will in practice be the extension of the sign bit, CHRES11, when the ADC works in differential mode, and set to zero when the ADC works in signed mode.

- **Bits 3:0 – RES[11:8]: Channel Result High byte**
  These are the four msbs of the 12-bit ADC result.

28.17.5.3 8-bit Mode

- **Bit 7:0 – Reserved**
  These bits will in practice be the extension of the sign bit, CHRES7, when the ADC works in signed mode, and set to zero when the ADC works in single-ended mode.

28.17.6 RESL – Channel n Result register Low

	Bit	7	6	5	4	3	2	1	0
12-/8-bit, right	+0x04	RES[7:0]							
12-bit, left.		RES[3:0]				–	–	–	–
	Read/Write	R	R	R	R	R	R	R	R
	Initial Value	0	0	0	0	0	0	0	0

28.17.6.1 12-/8-bit Mode

- **Bit 7:0 – RES[7:0]: Channel Result Low**
  These are the eight lsbs of the ADC result.

28.17.6.2 12-bit Mode, Left Adjusted

- **Bit 7:4 – RES[3:0]: Channel Result Low**
  These are the four lsbs of the 12-bit ADC result.

- **Bit 3:0 – Reserved**
  These bits are unused and reserved for future use. For compatibility with future devices, always write these bits to zero when this register is written.

28.17.7 SCAN – Channel Scan register

Scan is enabled when COUNT is set differently than 0. This register is available only for ADC channel 0.

Bit	7	6	5	4	3	2	1	0
+0x06	OFFSET[3:0]				COUNT[3:0]			
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 7:4 – OFFSET[3:0]: Positive MUX Setting Offset**
  The channel scan is enabled when COUNT != 0 and this register contains the offset for the next input source to be converted on ADC channel 0 (CH0). The actual MUX setting for positive input equals MUXPOS + OFFSET. The value is incremented after each conversion until it reaches the maximum value given by COUNT. When OFFSET is equal to COUNT, OFFSET will be cleared on the next conversion.

- **Bit 3:0 – COUNT[3:0]: Number of Input Channels Included in Scan**
  This register gives the number of input sources included in the channel scan. The number of input sources included is COUNT + 1. The input channels included are the range from MUXPOS to MUXPOS + COUNT.

28.18 Register summary – ADC

This is the register summary when the ADC is configured to give standard 12-bit results. The register summaries for 8-bit and 12-bit left adjusted will be similar, but with some changes in the result registers, CHnRESH and CHnRESL.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
+0x00	CTRLA	DMASEL[1:0]		CH[3:0]START				FLUSH	ENABLE	352
+0x01	CTRLB	IMPMODE	CURRLIMIT[1:0]		CONVMODE	FREERUN	RESOLUTION[1:0]		–	352
+0x02	REFCTRL	–	REFSEL[2:0]			–	–	BANDGAP	TEMPREF	354
+0x03	EVCTRL	SWEEP[1:0]		EVSEL[2:0]			EVACT[2:0]			354
+0x04	PRESCALER	–	–	–	–	–	PRESCALER[2:0]			356
+0x05	Reserved	–	–	–	–	–	–	–	–	
+0x06	INTFLAGS	–	–	–	–	CH[3:0]IF				356
+0x07	TEMP	TEMP[7:0]								357
+0x08	Reserved	–	–	–	–	–	–	–	–	
+0x09	Reserved	–	–	–	–	–	–	–	–	
+0x0A	Reserved	–	–	–	–	–	–	–	–	
+0x0B	Reserved	–	–	–	–	–	–	–	–	
+0x0C	CALL	CAL[7:0]								357
+0x0D	CALH	–	–	–	–	CAL[11:8]				357
+0x0E	Reserved	–	–	–	–	–	–	–	–	
+0x0F	Reserved	–	–	–	–	–	–	–	–	
+0x10	CH0RESL	CH0RES[7:0]								358
+0x11	CH0RESH	CH0RES[15:8]								357
+0x12	CH1RESL	CH1RES[7:0]								358
+0x13	CH1RESH	CH1RES[15:8]								357
+0x14	CH2RESL	CH2RES[7:0]								358
+0x15	CH2RESH	CH2RES[15:8]								357
+0x16	CH3RESL	CH3RES[7:0]								358
+0x17	CH3RESH	CH3RES[15:8]								357
+0x18	CMPL	CMP[7:0]								359
+0x19	CMPH	CMP[15:8]								358
+0x1A	Reserved	–	–	–	–	–	–	–	–	
+0x1B	Reserved	–	–	–	–	–	–	–	–	
+0x1C	Reserved	–	–	–	–	–	–	–	–	
+0x1D	Reserved	–	–	–	–	–	–	–	–	
+0x1E	Reserved	–	–	–	–	–	–	–	–	
+0x1F	Reserved	–	–	–	–	–	–	–	–	

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
+0x20	CH0 Offset	–	–	–	–	–	–	–	–	
+0x28	CH1 Offset	–	–	–	–	–	–	–	–	
+0x30	CH2 Offset	–	–	–	–	–	–	–	–	
+0x38	CH3 Offset	–	–	–	–	–	–	–	–	

28.19 Register summary – ADC channel

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
+0x00	CTRL	START	–	–	GAIN[2:0]			INPUTMODE[1:0]		359
+0x01	MUXCTRL	–	MUXPOS[3:0]				MUXNEG[2:0]			360
+0x02	INTCTRL	–	–	–	–	INTMODE[1:0]		INTLVL[1:0]		362
+0x03	INTFLAGS	–	–	–	–	–	–	–	IF	363
+0x04	RESL	RES[7:0]								364
+0x05	RESH	RES[15:8]								363
+0x06	SCAN	OFFSET				COUNT				363
+0x07	Reserved	–	–	–	–	–	–	–	–	

28.20 Interrupt vector summary

Table 28-19. Analog-to-digital converter interrupt vectors and their word offset address.

Offset	Source	Interrupt Description
0x00	CH0	Analog-to-digital converter channel 0 interrupt vector
0x02	CH1	Analog-to-digital converter channel 1 interrupt vector
0x04	CH2	Analog-to-digital converter channel 2 interrupt vector
0x06	CH3	Analog-to-digital converter channel 3 interrupt vector