AVR120: Characterization and Calibration of the ADC on an AVR

8-bit AVR® Microcontrollers

Application Note

Features

- **Understanding Analog to Digital Converter (ADC) characteristics**
- **Measuring parameters describing ADC characteristics**
- **Temperature, frequency and supply voltage dependencies**
- **Compensating for offset and gain error**

Introduction

This application note explains various ADC (Analog to Digital Converter) characterization parameters given in the datasheets and how they effect ADC measurements. It also describes how to measure these parameters during application testing in production and how to perform run-time compensation for some of the measured deviations.

A great advantage with the Flash memory of the AVR is that calibration code can be replaced with application code after characterization. Therefore, no code space is consumed by calibration code in the final product.

Rev. 2559C-AVR-05/04

Theory

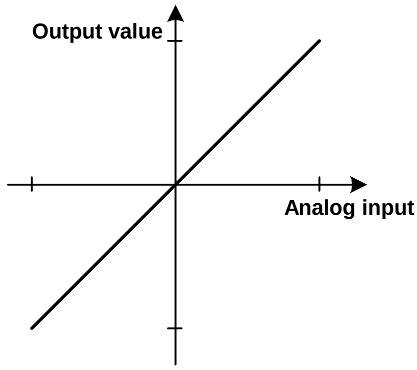
Before getting into the details, some central concepts need to be introduced. The following section (General ADC concepts) can be skipped if quantization, resolution, and ADC transfer functions are familiar to the reader.

General ADC concepts

The ADC translates an analog input signal to a digital output value representing the size of the input relative to a reference. To get a better basis for describing a general ADC, this document distinguishes between *ideal*, *perfect* and *actual* ADCs.

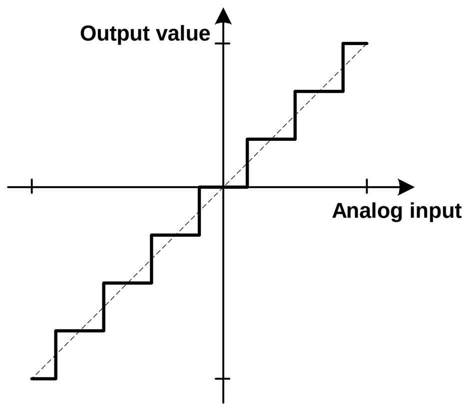
An *ideal* ADC is just a theoretical concept, and cannot be implemented in real life. It has infinite resolution, where every possible input value gives a unique output from the ADC within the specified conversion range. An ideal ADC can be described mathematically by a straight-line transfer function, as shown in Figure 1.

Figure 1. Transfer function of an ideal ADC



To define a *perfect* ADC, the concept of quantization must be introduced. Due to the digital nature of an ADC, continuous output values are not possible. The output range must be divided into a number of steps, one for each possible digital output value. This means that one output value does not correspond to a unique input value, but a small range of input values. This results in a staircase transfer function. The resolution of the ADC equals the number of unique output values. For instance, an ADC with 8 output steps has a resolution of 8 levels or, in other words, 3 bits. The transfer function of an example 3-bit perfect ADC is shown in Figure 2 together with the transfer function of an ideal ADC. As seen on the figure, the perfect ADC equals the ideal ADC on the exact midpoint of every step. This means that the perfect ADC essentially rounds input values to the nearest output step value.

Figure 2. Transfer function of a 3-bit perfect ADC



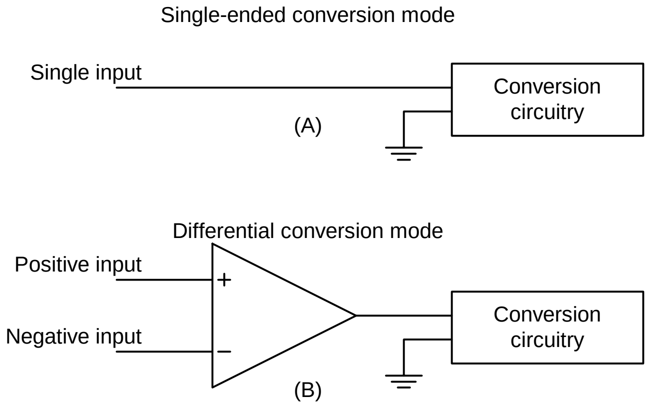
The maximum error for a perfect ADC is ±½ step. In other words, the maximum quantization error is always ±½ LSB, where LSB is the input voltage difference corresponding to the Least Significant Bit of the output value. Real ADCs have other sources of errors, described later in this document.

Conversion ranges

The ADC in Atmel's AVR devices can be configured for single-ended or differential conversion. Single-ended mode is used to measure input voltages on a single input channel, while differential mode is used to measure the difference between two channels. Regardless of conversion mode, input voltages on any channel must stay between GND and AV_{CC}.

When using single-ended mode, the voltage relative to GND is converted to a digital value. Using differential channels, the output from a differential amplifier (with an optional gain stage) is converted to a digital value (possibly negative). A simplified illustration of the input circuitry is shown in Figure 3.

Figure 3. Simplified ADC input circuitry



To decide the conversion range, the conversion circuitry needs a voltage reference (V_{REF}) to indicate the voltage level corresponding to the maximum output value. According to the datasheets, V_{REF} should be at least 2.0 V for standard devices, while devices operating down to 1.8V can use a reference voltage down to 1.0V. This applies both to single-ended and differential mode. Consult the datasheet for details.

Single-ended conversion range

Single-ended conversion feeds the input channel directly to the conversion circuitry, as shown in Figure 3A. The 10-bit ADC of the AVR therefore converts continuous input voltages from GND to V_{REF} to discrete output values from 0 to 1023.

Differential conversion range

Differential conversion feeds the two input channels to a differential amplifier with an optional gain stage. The output from the amplifier is then fed to the conversion logic, as shown in Figure 3B. Voltage differences from $-V_{REF}$ to $+V_{REF}$ therefore results in discrete output values from –512 to +511. The digital output is represented in 2's complement form. Even when measuring negative voltage differences, the voltages applied to the input channels themselves must stay between GND and AV_{CC}.

Note that some devices cannot measure negative differences, e.g. ATtiny26.

The need for calibration

The total error of the actual ADC comes from more than just quantization error. This document describes offset and gain errors and how to compensate for them. It also describes two measures for non-linearity, namely differential and integral non-linearity.

For most applications, the ADC needs no calibration when using single ended conversion. The typical accuracy is 1-2 LSB, and it is often neither necessary nor practical to calibrate for better accuracies.

However, when using differential conversion the situation changes, especially with high gain settings. Minor process variations are scaled with the gain stage and give large parameter differences from part to part. The uncompensated error is typically above 20 LSB. These variations must be characterized for every device and compensated for in software.

At first sight 20 LSB seems to be a large value, but does not mean that differential measurements are impractical to use. Using simple calibration algorithms, accuracies of typically 1-2 LSB can be achieved.

Absolute error

The absolute error is the maximum deviation between the ideal straight line and the actual transfer function, including the quantization steps. The minimum absolute error is therefore ½ LSB, due to quantization.

Absolute error or absolute accuracy is the total uncompensated error and includes quantization error, offset error, gain error and non-linearity. Offset, gain and non-linearity are described later in this document.

Absolute error can be measured using a ramp input voltage. In this case all output values are compared against the input voltage. The maximum deviation gives the absolute error.

Note that absolute error cannot be compensated for directly, without using e.g. memory-expensive lookup tables or polynomial approximations. However, the most significant contributions to the absolute error - the offset and gain error - can be compensated for.

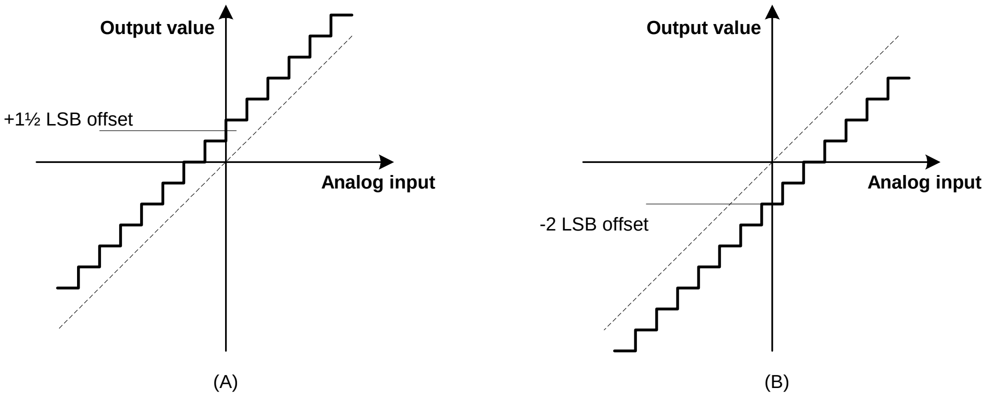
Be aware that the absolute error represent a reduction in the ADC range, and one should therefore consider the margins to the minimum and maximum input values to avoiding clipping keeping the absolute error in mind.

Offset error

The offset error is defined as the deviation of the actual ADC's transfer function from the ideal straight line at zero input voltage.

When the transition from output value 0 to 1 does not occur at an input value of ½ LSB, then we say that there is an offset error. With positive offset errors, the output value is larger than 0 when the input voltage approaches ½ LSB from below. With negative offset errors, the input value is larger than ½ LSB when the first output value transition occurs. In other words, if the actual transfer function lies below the ideal line, there is a negative offset and vice versa. Negative and positive offsets are shown in Figure 4.

Figure 4. Examples of positive (A) and negative (B) offset errors



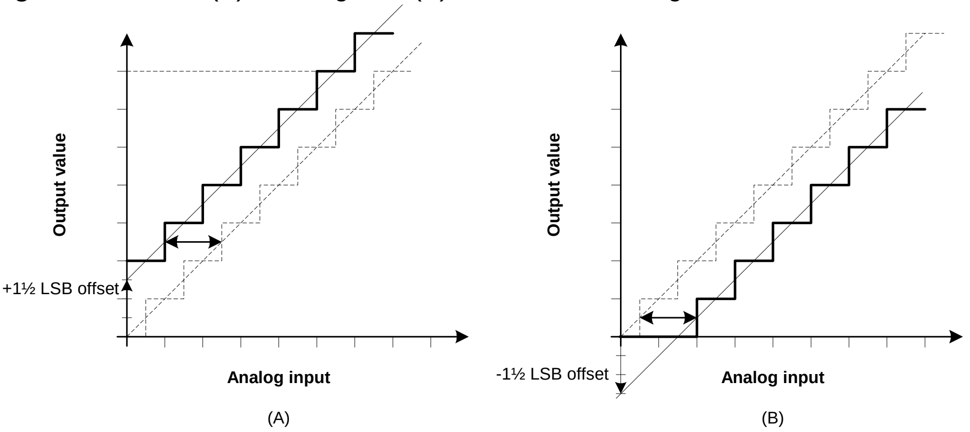
Since single-ended conversion gives positive results only, the offset measurement procedures are different when using single-ended and differential channels.

Offset error – single-ended channels

To measure the offset error, increase the input voltage from GND until the first transition in the output value occurs. Calculate the difference between the input voltage for which the perfect ADC would have shown the same transition and the input voltage corresponding to the actual transition. This difference, converted to LSB, equals the offset error.

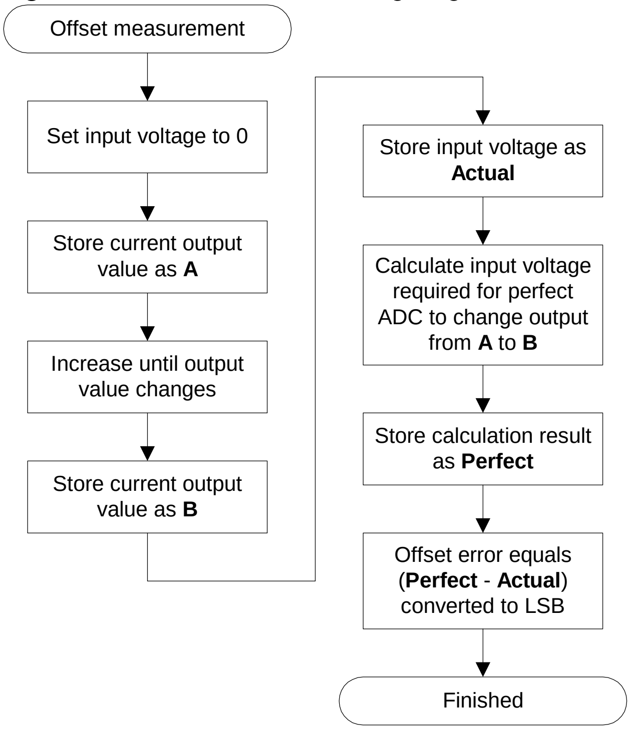
In Figure 5A, the first transition occurs at 1 LSB. The transition is from 2 to 3, which equals an input voltage of 2½ LSB for the perfect ADC. The difference is +1½ LSB, which equals the offset error. The double-headed arrows show the differences. The same procedure applies to Figure 5B. The first transition occurs at 2 LSB. The transition is from 0 to 1, which equals an input voltage of ½ LSB for the perfect ADC. The difference is -1½ LSB, which equals the offset error.

Figure 5. Positive (A) and negative (B) offset errors in single-ended mode



The measurement procedure can be formalized as described in Figure 6.

Figure 6. Flowchart for measuring single ended offset errors



To compensate for offset errors when using single ended channels, subtract the offset error from every measured value. Be aware that offset errors limit the available range for the ADC. A large positive offset error causes the output value to saturate at maximum before the input voltage reaches maximum. A large negative offset error gives output value 0 for the smallest input voltages.

Offset error - differential channels

With differential channels, the offset measurement can be performed much easier since no external input voltage is required. The two differential inputs can be connected to the same voltage internally and the resulting output value is then the offset error. Since this method gives no exact information on where the first transition occurs, it gives an error of ½ to 1 LSB (worst case).

To compensate for offset errors when using differential channels, subtract the offset error from every measured value.

Gain error

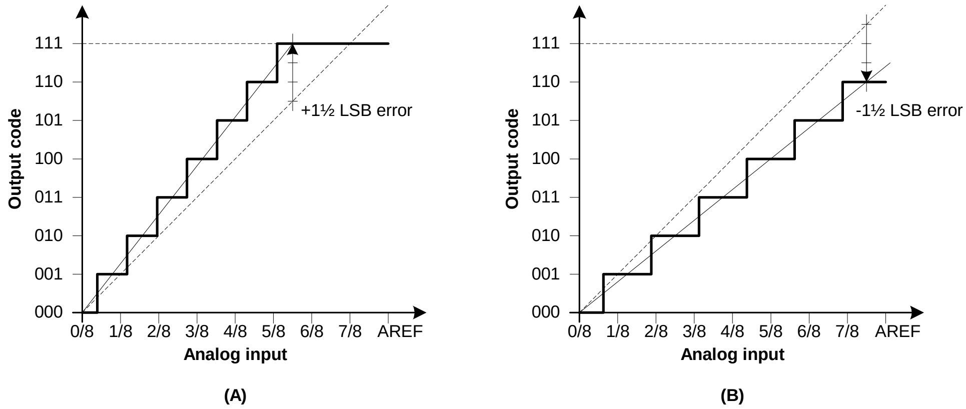
The gain error is defined as the deviation of the last output step's midpoint from the ideal straight line, after compensating for offset error.

After compensating for offset errors, applying an input voltage of 0 always give an output value of 0. However, gain errors cause the actual transfer function slope to deviate from the ideal slope. This gain error can be measured and compensated for by scaling the output values.

Run-time compensation often uses integer arithmetic, since floating point calculation takes too long to perform. Therefore, to get the best possible precision, the slope deviation should be measured as far from 0 as possible. The larger the values, the better precision you get. This is described in detail later in this document.

The example of a 3-bit ADC transfer functions with gain errors is shown in Figure 7. The following description holds for both single-ended and differential modes.

Figure 7. Examples of positive (A) and negative (B) gain errors

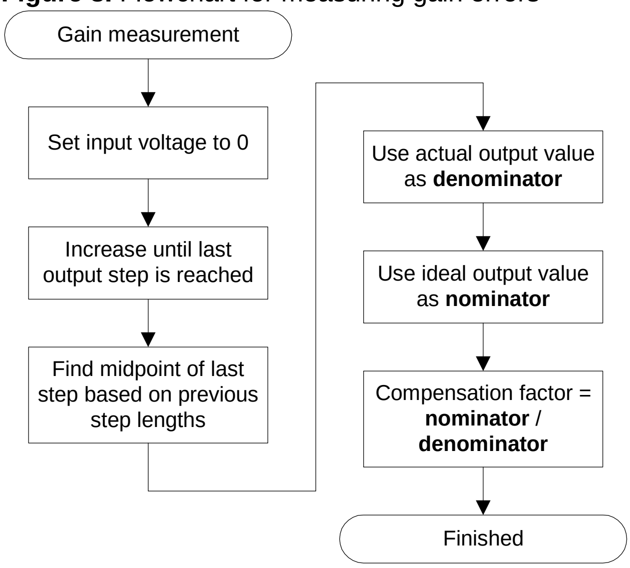


To measure the gain error, the input value is increased from 0 until the last output step is reached. The scaling factor for gain compensation equals the ideal output value for the midpoint of the last step divided by the actual value of the step.

In Figure 7A, the output value saturates before the input voltage reaches its maximum. The vertical arrow shows the midpoint of the last output step. The ideal output value at this input voltage should be 5.5, and the scaling factor equals 5.5 divided by 7. In Figure 7B, the output value has only reached 6 when the input voltage is at its maximum. This results in a negative deviation for the actual transfer function. The ideal output value for the midpoint of the last step is 7.5 in this case. The scaling factor now equals 7.5 divided by 6.

The measurement procedure is illustrated in Figure 8.

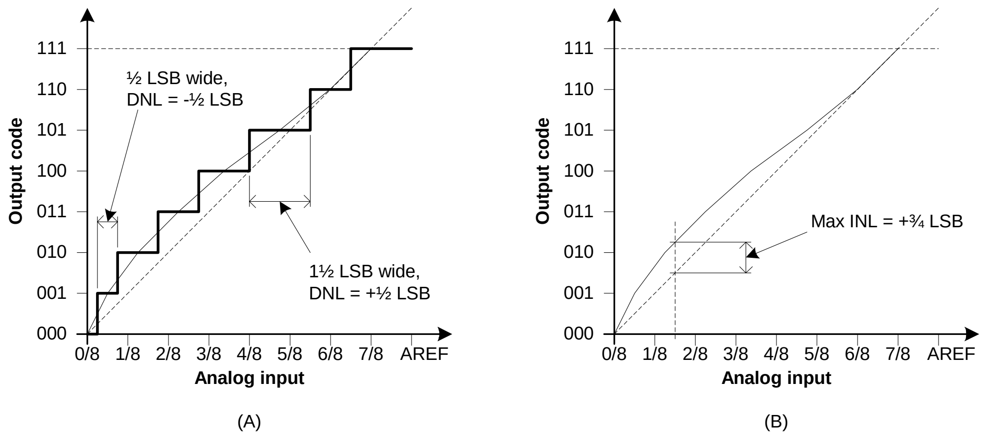
Figure 8. Flowchart for measuring gain errors



Non-Linearity

When offset and gain errors are compensated for, the actual transfer function should be equal to the transfer function of perfect ADC. However, non-linearity in the ADC may cause the actual curve to deviate slightly from the perfect curve, even if the two curves are equal around 0 and at the point where the gain error was measured. There are two methods for measuring non-linearity, both described below. Figure 9 shows examples of both measures.

Figure 9. Example of a non-linear ADC conversion curve



Differential non-linearity

Differential Non-Linearity (DNL) is defined as the maximum and minimum difference between the step width and the perfect width (1 LSB) of any output step.

Non-linearity produces quantization steps with varying widths. All steps should be 1 LSB wide, but some are narrower or wider.

To measure DNL, a ramp input voltage is applied and all output value transitions are recorded. The step lengths are found from the distance between the transitions, and the most positive and negative deviations from 1 LSB are used to report the maximum and minimum DNL.

Integral non-linearity

Integral Non-Linearity (INL) is defined as the maximum vertical difference between the actual and the perfect curve.

INL can be interpreted as a sum of DNLs. E.g. several consecutive negative DNLs raise the actual curve above the perfect curve as shown in Figure 9A. Negative INLs indicate that the actual curve is below the perfect curve.

The maximum and minimum INL are measured using the same ramp input voltage as in DNL measurement. Record the deviation at each conversion step midpoint and report the most positive and negative deviations as maximum and minimum INL.

Measurements and compensation

It is important that DNL and INL values are measured after offset and gain error compensation. If not, the results will be infected by the offset and gain error and thus not reveal the true DNL and INL.

Non-linearity cannot be compensated for with simple calculations. Polynomial approximations or table lookups can be used for that purpose. However, the typical DNL and INL values are ½ LSB for the 10-bit ADC of the AVR, and are rarely of any concern in real life applications.

Temperature, frequency and supply voltage dependencies

When using the internal voltage reference with the ADC, the accuracy of this reference must be considered. The internal voltage reference is proportional to the bandgap voltage, which is characterized in the devices' datasheets. The characteristics show that the bandgap voltage is slightly dependent on supply voltage and operating temperature.

The ADC accuracy also depends on the ADC clock. The recommended maximum ADC clock frequency is limited by the internal DAC in the conversion circuitry. For optimum performance, the ADC clock should not exceed 200 kHz. However, frequencies up to 1 MHz do not reduce the ADC resolution significantly.

Operating the ADC with frequencies greater than 1 MHz is not characterized.

Bandwidth and input impedance

When using single-ended mode, the ADC bandwidth is limited by the ADC clock speed. Since one conversion takes 13 ADC clock cycles, a maximum ADC clock of 1 MHz means approximately 77k samples per second. This limits the bandwidth in single-ended mode to 38.5 kHz, according to the Nyquist sampling theorem.

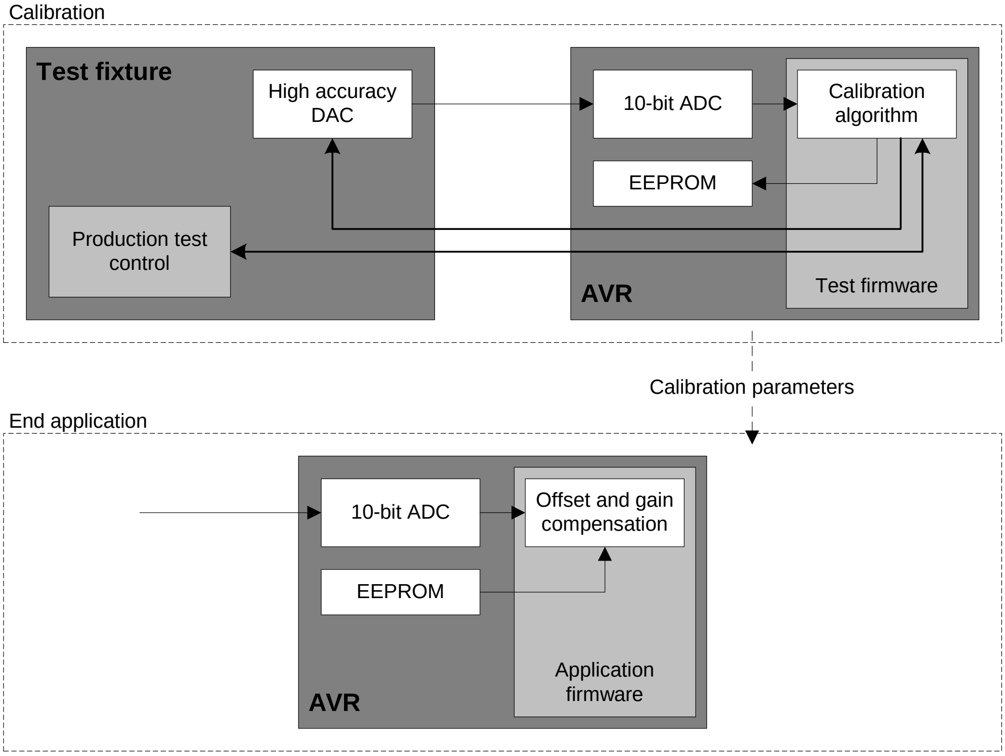
When using differential mode, the bandwidth is limited to 4 kHz by the differential amplifier. Input frequency components above 4 kHz should be removed by an external analog filter, to avoid non-linearity's.

The input impedance to V_{CC} and GND is typically 100 MΩ. Together with the output impedance of the signal source, this creates a voltage divider. The signal source should therefore have sufficiently low output impedance to get correct conversion results.

Implementation

An example setup for calibrating the ADC is shown in Figure 10.

Figure 10. Production calibration setup



During the production test phase, each device's ADC must be characterized using a test setup similar to this. When the test fixture is ready for calibrating the AVR, the tester signals the AVR to start calibrating itself. The AVR uses the test fixture's high accuracy DAC (e.g. 16-bit resolution) to generate input voltages to the calibration algorithm. When calibration is finished, the parameters for offset and gain error compensation are programmed into the EEPROM for later use, and the AVR signals that it is ready for other test phases.

Note that this requires the EESAVE fuse to be programmed, so that the Flash memory can be reprogrammed without erasing the EEPROM. Otherwise, the ADC parameters must be temporally stored by the programmer while erasing the device.

Fixed-point arithmetic for offset and gain error compensation

Floating-point arithmetic is not an efficient way of scaling the ADC values. However, the scaling factor for gain error compensation will be somewhere close to 1, and needs a certain precision to yield good compensated ADC values. Therefore, fixed-point numbers represented by integers can be used.

Since the gain compensation factor certainly never exceeds 2, it can be scaled by a factor of 2^{14} to fit exactly in a signed 16-bit word. In other words, the scaling factor can be represented by two bytes as a 1:14 signed fixed-point number.

The equation for both offset and gain error compensation is shown in Equation 1.

Equation 1.

$$realvalue = (adcvalue - offset) \cdot gainfactor$$

When the calculation result is truncated to an ordinary integer later, it is always truncated to the largest integer less than or equal to the result. In order to achieve correct rounding to the nearest integer, 0.5 must be added before truncating.

Adding 0.5, scaling the equation by 2^{14} and moving the offset correction outside the parentheses gives Equation 2.

Equation 2.

$$2^{14} \cdot realvalue = 2^{14} \cdot adcvalue \cdot gainfactor + 2^{14} \cdot 0.5 - 2^{14} \cdot offset \cdot gainfactor$$

Since the gain factor and offset correction value are constants, further optimization can be achieved. In addition, if the result is scaled by 2^2, giving a total scale of 2^{16}, the upper two bytes of the result equals the truncated integer without the need for a 16-step right-shift.

We introduce some constants and summarise it all in Equation 3.

Equation 3.

$$factor = 2^{14} \cdot gainfactor\ ,$$

$$correction = 2^{14} \cdot (0.5 - offset \cdot gainfactor)\ ,$$

$$2^{16} \cdot realvalue = 2^{2} \cdot (adcvalue \cdot factor + correction)$$

Using this method, the calibration software calculates the constants $factor$ and $correction$ and stores them in EEPROM memory. At execution time the compensation firmware only needs one integer multiplication, one addition and two left shift operations to correct the ADC values. Using the IAR C compiler with highest optimization for speed, this takes 42 CPU cycles.

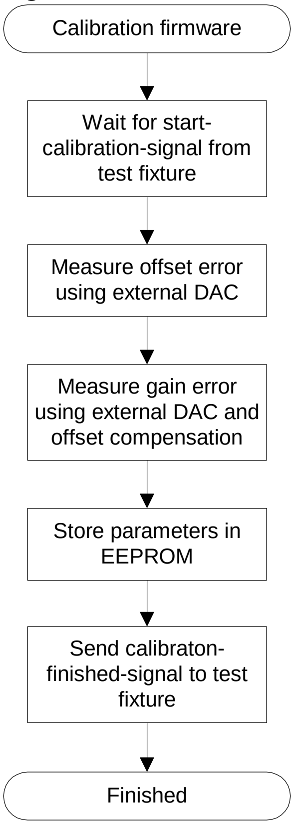
Calibration

The test fixture design is outside the scope of this application note. However, a flowchart of a calibration firmware in the AVR is given. It uses the external DAC through the test fixture and runs its own calibration algorithm.

There is no need for using several ADC channels, only switching between single ended and differential conversion. The ADC parameters are the same regardless of which channel is used. The multiplexer does not introduce any errors.

The firmware could be implemented as shown in Figure 11.

Figure 11. Flowchart for calibration firmware

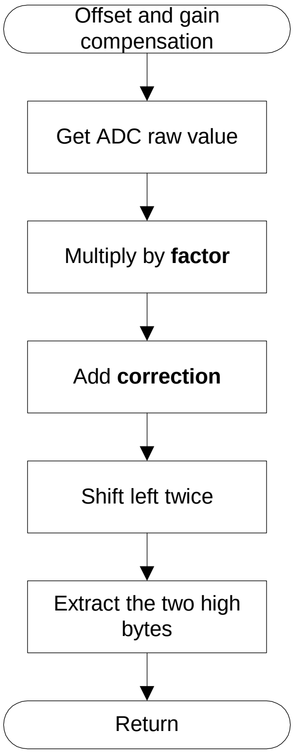


This piece of firmware is programmed into the AVR prior to calibration, and is replaced by the actual application firmware afterwards. Once again, the EESAVE fuse must be programmed to preserve the calibration parameters in EEPROM during Flash reprogramming.

Compensation

The run-time compensation code can be implemented as a small function. Every ADC measurement is run through this function, which uses the constants $factor$ and $correction$.

Figure 12. Flowchart for offset and gain compensation



This calculation in Figure 12 can be implemented using the following C function, or alternatively as a macro:

```
signed int adc_compensate( signed int adcvalue,
                           signed int factor,
                           signed long correction )
{
    return (((((signed long)adcvalue*factor)+correction)<<2)>>16);
}
```

The parameter constants stored in EEPROM could be copied to SRAM variables during startup for quicker access later.

Literature references

- Robert Gordon – A Calculated Look at Fixed-Point Arithmetic http://www.embedded.com/98/9804fe2.htm
- Application Note AVR210 – Using the AVR Hardware Multiplier http://www.atmel.com/dyn/resources/prod_documents/DOC1631.PDF

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/ High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature