Basics of Analog-to-digital conversation

In electronics, an analog-to-digital converter (ADC, A/D) is a system that converts an analog signal, such as a sound picked up by a microphone or light entering a digital camera, into a digital signal (Fig.1). An ADC may also provide an isolated measurement such as an electronic device that converts an input analog voltage or current to a digital number representing the magnitude of the voltage or current. Typically the digital output is a two's complement binary number that is proportional to the input, but there are other possibilities. There are several ADC architectures. Due to the complexity and the need for precisely matched components, all but the most specialized ADCs are implemented as integrated circuits (ICs). A digital-to-analog converter (DAC) performs the reverse function; it converts a digital signal into an analog signal.

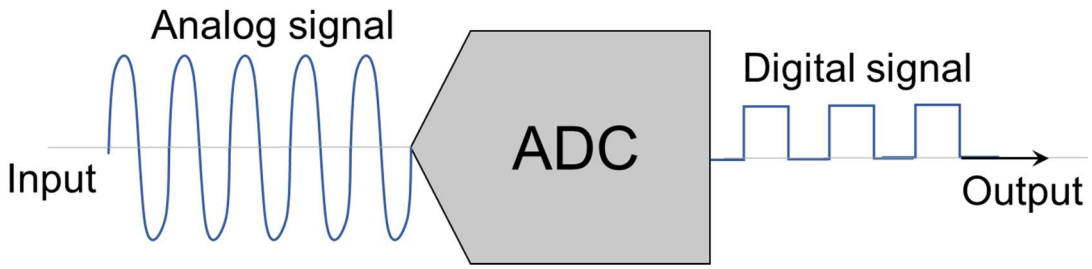

Figure 1 – An ADC conversation

ADC converts a continuous-time and continuous-amplitude analog signal to a discrete-time and discrete-amplitude digital signal. The conversion involves quantization of the input, so it necessarily introduces a small amount of error or noise. Furthermore, instead of continuously performing the conversion, an ADC does the conversion periodically, sampling the input, limiting the allowable bandwidth of the input signal.

The performance of an ADC is primarily characterized by its **bandwidth** and **signal-to-noise ratio** (SNR). The bandwidth of an ADC is characterized primarily by its sampling rate. The SNR of an ADC is influenced by many factors, including the resolution, linearity and accuracy (how well the quantization levels match the true analog signal), aliasing and jitter. The SNR of an ADC is often summarized in terms of its effective number of bits (ENOB), the number of bits of each measure it returns that are on average not noise. An ideal ADC has an ENOB equal to its resolution. ADCs are chosen to match the bandwidth and required SNR of the signal to be digitized. If an ADC operates at a sampling rate greater than twice the bandwidth of the signal, then per the Nyquist–Shannon sampling theorem, perfect reconstruction is possible. The presence of quantization error limits the SNR of even an ideal ADC. However, if the SNR of the ADC exceeds that

of the input signal, its effects may be neglected resulting in an essentially perfect digital representation of the analog input signal.

Resolution

The resolution of the converter indicates the number of discrete values it can produce over the range of analog values. The resolution determines the magnitude of the quantization error and therefore determines the maximum possible average signal-to-noise ratio for an ideal ADC without the use of oversampling. The values are usually stored electronically in binary form, so the resolution is usually expressed as the audio bit depth. In consequence, the number of discrete values available is assumed to be a power of two. For example, an ADC with a resolution of 8 bits (Fig. 2) can encode an analog input to one in 256 different levels ($2^8 = 256$). The values can represent the ranges from 0 to 255 (i.e. unsigned integer) or from −128 to 127 (i.e. signed integer), depending on the application.

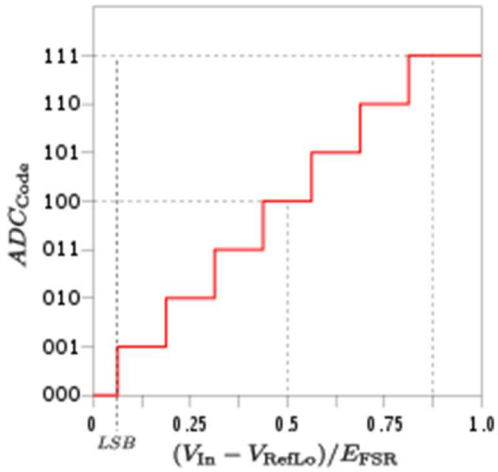

Figure 2 – An 8-level ADC coding scheme

Some ADCs have pseudo-differential configuration, two pins (VIN+ and VIN-) are used for the signal input. With a pseudo-differential input, the second input pin provides the reference for the signal. The distinction between pseudo-differential inputs and standard differential inputs is that the signal on the VIN- can only deviate a small range from the voltage of the VSS supply rail. Although this restriction requires that a single-ended source is connected to the ADC, the input stage maintains the ability to cancel small common-mode fluctuations on the input pins. The voltage reference for the ADC may be provided internally or by an external source. Since the accuracy of the measurement results is directly affected by the reference, it is important that the reference source is stable over time and temperature.

For low cost converters, the reference input is often implemented as a single-ended input. In this case, one pin is used for the reference input and the input voltage range for the converter is determined by VSS and VREF. For higher performance converters, two voltage reference pins are typically

provided. The input voltage range for these converters is determined by the voltage difference between VREF+ and VREF-. In either case, the voltage range for the reference inputs is usually restricted by the VDD and VSS power supply rails. Although a "real world" ADC will have higher resolution, a theoretical 3-bit ADC will be used here to demonstrate the performance of the ideal converter and the various sources of error. The figure shows the transfer function of the ideal 3-bit ADC. As the transfer function indicates, the ideal 3-bit ADC provides eight equally spaced digital output codes over the analog input voltage range. Each digital output code represents a fractional value of the reference voltage. The largest value that can be obtained from the ADC is (n-1)/n, where n is the resolution in bits.

Resolution can also be defined electrically, and expressed in volts. The change in voltage required to guarantee a change in the output code level is called the least significant bit (LSB) voltage. The resolution Q of the ADC is equal to the LSB voltage. The voltage resolution of an ADC is equal to its overall voltage measurement range divided by the number of intervals:

$$Q = \frac{E_{\mathrm{FSR}}}{2^M}$$

where M is the ADC's resolution in bits and E_{FSR} is the full scale voltage range (also called 'span'). E_{FSR} is given by

$$E_{\mathrm{FSR}} = V_{\mathrm{RefHi}} - V_{\mathrm{RefLow}},$$

where V_{RefHi} and V_{RefLow} are the upper and lower extremes, respectively, of the voltages that can be coded. Normally, the number of voltage intervals is given by

$$N = 2^M,$$

where M is the ADC's resolution in bits. That is, one voltage interval is assigned in between two consecutive code levels.

Sampling and quantization

An ADC carries out two processes, sampling and quantization. The ADC represents an analog signal, which has infinite resolution, as a digital code that has finite resolution. The ADC produces 2N digital values where N represents the number of binary output bits. The analog input signal will fall between the quantization levels because the converter has finite resolution resulting in an inherent uncertainty or quantization error. That error determines the maximum dynamic range of the converter.

The sampling process represents a continuous time domain signal with values measured at discrete and uniform time intervals. This process determines the maximum bandwidth of the sampled signal in accordance with the Nyquist Theory. This theory states that the signal frequency must be less than or equal to one half the sampling frequency to prevent aliasing. Aliasing is a condition in which frequency signals outside the desired signal band will, through the sampling process, appear within the bandwidth of interest. However, this aliasing process can be exploited in communications systems

design to down-convert a high frequency signal to a lower frequency. This technique is known as under-sampling. A criterion for under-sampling is that the ADC has sufficient input bandwidth and dynamic range to acquire the highest frequency signal of interest.

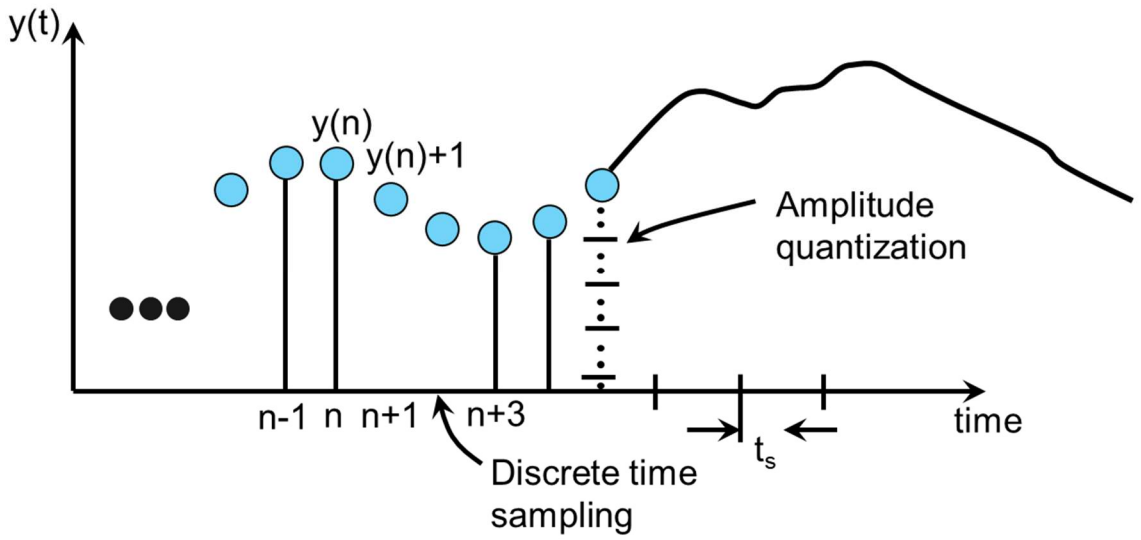

Figure 3 – Sampling Process

Sampling and quantization are important concepts because they establish the performance limits of an ideal ADC. In an ideal ADC, the code transitions are exactly 1 least significant bit (LSB) apart. So, for an N-bit ADC, there are 2N codes and 1 LSB = FS/2N, where FS is the full-scale analog input voltage. However, ADC operation in the real world is also affected by non-ideal effects, which produce errors beyond those dictated by converter resolution and sample rate. These errors are reflected in a number of AC and DC performance specifications associated with ADCs.

ADC classification

Converting signals from analog to digital or digital to analog is an unavoidable task for today's engineer. There are many different kinds of analog to digital converters and digital to analog converters. While these differ in their architecture, they all work to achieve a similar end. Since digital signal processing cannot be done with analog values, it would be analogous to a French speaking person trying to speak with a German speaking person. It would not work without a translator. ADC and DAC devices can help to act like that translator. When an ADC sees an analog voltage, its job is to turn the analog voltage into a binary code at a given period of time. This means the ADC will sample the analog voltage at an instant, and then it determines what the value would be in binary on the output side of the ADC. The amount of samples that the device takes every second will be called out on its

documentation. By being able to take that many samples in a second, it is possible to accurately log what the analog voltage looked like by using a binary interpretation.

Sometimes the sample rate of an ADC is not high enough to accurately recreate its input which causes aliasing. This is where signals start to be indistinguishable from one another, or aliases of one another. Imagine a video camera that can take 24 frames per second when recording. For most applications this will be fine; however, if trying to view something that is moving very fast, this can distort the image. Recall the effect of watching a television on a recording in the late nineties. The image on the television would be flickering. This is because the refresh rate on the television itself is much faster than the recording can capture at its given frames per second. The image would be distorted because the video is actually a succession of pictures. There is more happening between each picture than is actually being represented on the video. The same kind of effect can happen with an ADC. To avoid this, it is pertinent to make sure the sample rate is at least two times higher than the highest frequency that needs to be transmitted. This is referred to as the **Nyquist rate**.

Thus, **speed** and **accuracy** are two critical measures of ADC performance. As such, they provide a means for broadly categorizing today's monolithic ADCs. ADC chips can be grouped along these lines as **general-purpose**, **high-speed**, or **precision**. Converters with eight- to fourteen-bit resolution and conversion rates below ten Megasamples/second are typically considered as general-purpose ADCs. Those with conversion rates above ten Megasamples/second usually get the high-speed moniker, while those with sixteen bits or more of resolution fall into the precision ADC category. These definitions, however, are somewhat arbitrary and largely reflect the current state-of-the-art.

Within these broad categories, ADCs may also be grouped according to **converter architecture**. The three most popular ADC architectures are **Successive Approximation Registers** (SAR), **Delta-Sigma** (ΔΣ), and **Flash** converters. Each architecture offers certain advantages with respect to conversion speed, accuracy, and other parameters. The characteristics associated with each architecture help determine its suitability for a given application. Over time, the migration of ADC designs to CMOS processes with smaller geometries has increased the possibilities for performance enhancements, while also allowing higher levels of integration.

Each of these will convert an analog signal into a digital output, but there are slight differences in how this is done. The SAR will sample an analog input and hold it, turn it into a digital signal, then pass it off. Delta-Sigma converters will average the sample over the time it takes to convert it into a digital signal. Pipelined converters divide the conversion into different stages

allowing for very fast conversion speeds. Each of these will have positives and negatives. The SAR architecture will be easy to use, typically use lower power, and have low latency time with high accuracy. The Sigma-Delta will have a very high resolution and high stability at low power and low cost; however, it will work at much lower speeds than the SAR and Pipeline architectures. The Pipeline ADC will work at the higher speeds and higher bandwidth than the previous examples, but will have a lower resolution and require more power to run.

Flash (parallel) ADC

In the flash or parallel ADC architecture, an array of 2 in power N comparators converts an analog signal to digital with a resolution of N bits (Fig. 4). The comparators receive the analog signal on one input and a unique fraction of the reference voltage on the other. The reference voltage for each comparator is often a tap off a resistive voltage divider, whereby the comparators are biased in voltage increments equivalent to 1 least significant bit. The comparator array is clocked simultaneously. The comparators with reference voltages less than the analog input will output a digital one. The comparators with reference voltages greater than the analog input will output a digital zero. When read together, the outputs present a "thermometer code," which the output logic converts to standard binary code.

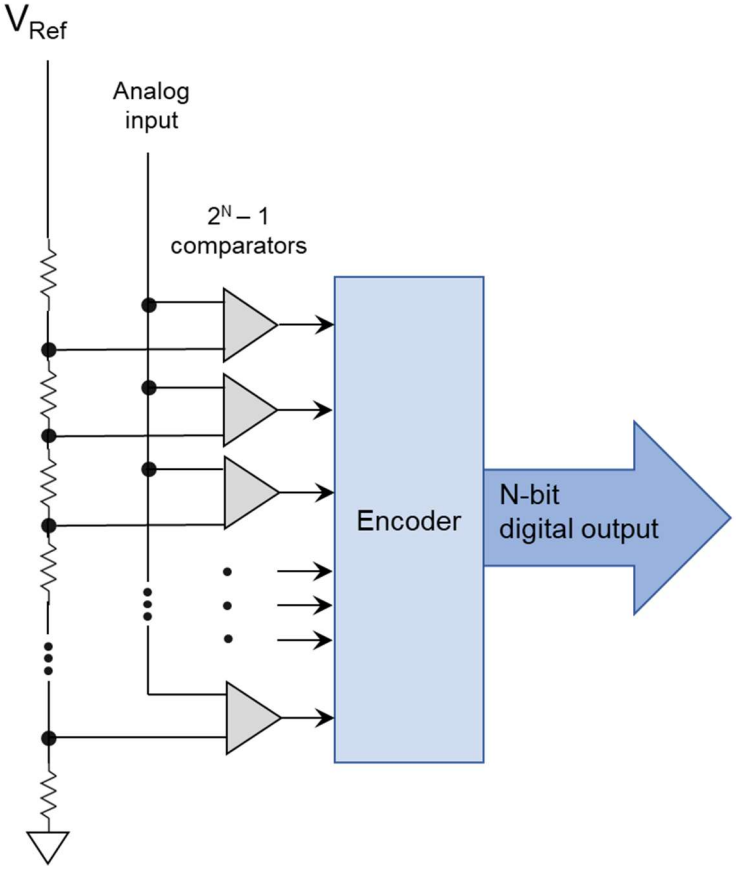

Figure 4 – Flash ADC block diagram.

The advantage of Flash ADC – their speed, as they converts result in one clock cycle. The main problem – they requires many comparators. The physical limits of monolithic integration generally allow only up to 8 bits of resolution for one ADC chip.

Successive approximation-register converter

The successive approximation-register (SAR) converter works like a balance scale that compares an unknown weight against a series of known weights. SAR converter compares the analog input voltage against a series of successively smaller voltages representing each of the bits in the digital output code. These voltages are fractions of the full-scale input voltage.

The first comparison is made between the analog input voltage and a voltage representing the most significant bit (MSB). If that analog input voltage is greater than the MSB voltage, the value of the MSB is set to 1, otherwise it's set to 0. The second comparison is made between the analog input voltage and a voltage representing the sum of the MSB and the next most significant bit. The value of the second most significant bit is then set accordingly. The third comparison is made between the analog input voltage and the voltage representing the sum of the three most significant bits. At this point, the value of the third most significant bit is set.

The process repeats until the value of the LSB is established. The main advantage of SAR ADC that it uses a simple architecture to achieve high resolution (Fig. 5), so it's usually used as a common built in part of any microcontroller. But ADC of this type requires N comparisons to achieve N-bit resolution, which decreasing their sampling rate and cause limitations in frequency range.

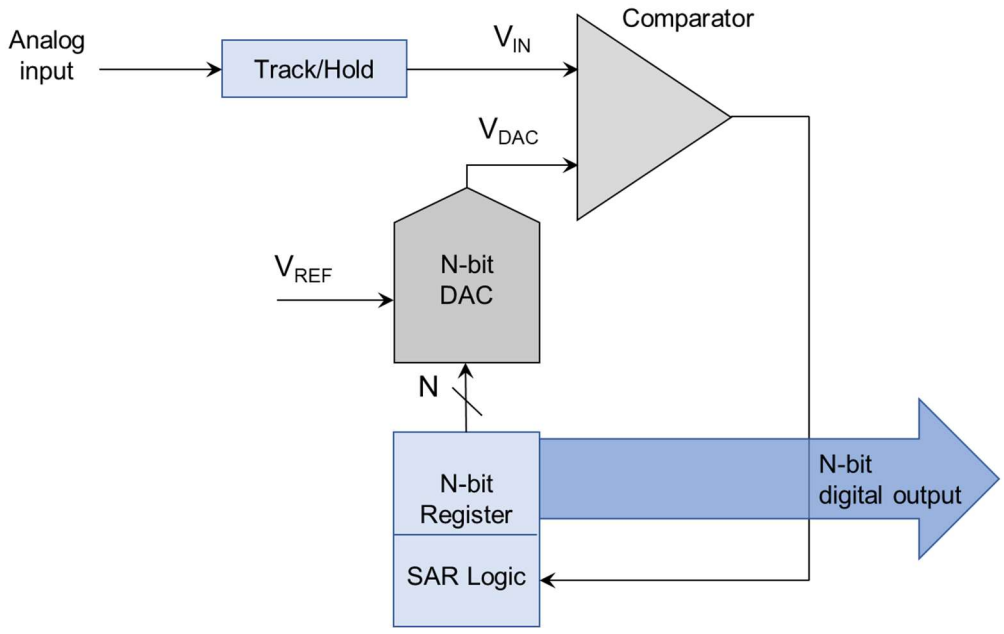

Figure 5 – Successive approximation-register ADC block diagram.

Sigma-Delta ADC

The basic elements of Sigma-Delta architecture are integrator, comparator, and one-bit digital-to-analog converter (DAC), which together form a sigma-delta modulator (Fig. 6). The modulator subtracts the DAC from the analog input signal and then feeds the signal to the integrator. The output of the integrator then goes to a comparator, which converts the signal to a one-bit digital output.

The resulting bit is fed to the DAC, which produces an analog signal to be subtracted from the input signal. The process repeats at a very fast oversampled rate. The modulator produces a binary stream in which the ratio of ones to zeros is a function of the input signal's amplitude. By digitally filtering and decimating this stream of one and zeroes, a binary output representing the value of the analog input is obtained.

Sigma-Delta ADC yields the highest precision for lower input-bandwidth applications. From the other hand, oversampling and latency cause large time of single conversation, so we can use this type only for slow signals (up to 100 kHz).

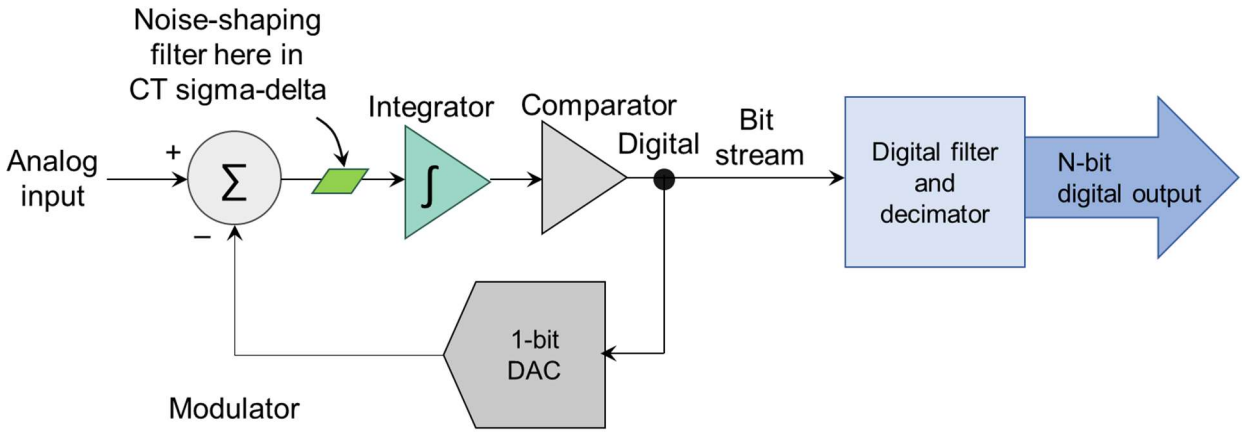

Figure 6 – Successive approximation ADC functional block diagram.

More information about ADC basics:

- https://en.wikipedia.org/wiki/Analog-to-digital_converter
- https://wiki.analog.com/university/courses/electronics/text/chapter-20
- https://www.digikey.com/en/articles/techzone/2017/sep/adc-dac-tutorial
- http://microchipdeveloper.com/adc:start

- https://www.rohm.com/electronics-basics/ad-da-converters/ad-converter-configurations
- https://www.ti.com/seclit/ml/slyc139/slyc139.pdf
- http://www.vra.be/teksten/Analog%20design.pdf