06.1 Analog–to–digital and digital–to–analog conversion

Most sensors and actuators, which are necessary components for control systems, have as input and output analog signals. These are continuous in time and can take on a virtually infinite number of real values. In preceding chapters, we have learned that computers work with binary digital signals, which communicate information by changing voltage between conventional levels representing logical true and false. How can a binary digital signal represent an analog signal and vice versa?

Analog-to-digital conversion and analog outputs

Analog-to-digital conversion (ADC) is the process by which a binary digital signal is made to represent an analog signal. This proceeds in three operations, as illustrated in Fig. 06.1:

sampling the analog signal is sampled: measured at discerete moments in time, usually at a fixed sample period T (i.e. sample rate 1/T or angular sample rate 2π/T);

quantization the sampled measurement is quantized: represented by one of a finite set of values limited by the number of bits available in binary conversion; and

binary conversion the quantized measurement is converted to binary: given a binary representation such that it can be represented by a binary digital signal.

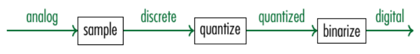
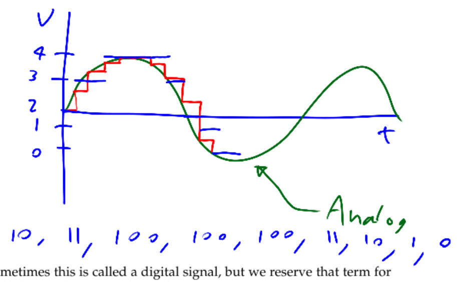


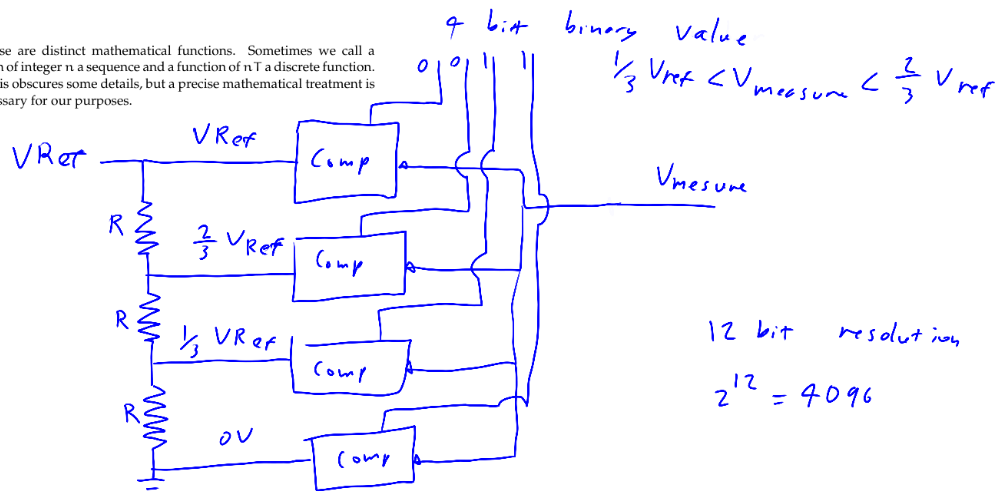
Figure 06.1: the operations required to convert an analog to a binary digital signal.



It is convenient to give a name to the measurement after it is sampled, but before it is quantized: a discrete signal,[1] which is represented mathematically as a sequence of real numbers paired with corresponding time intervals (usually fixed). These sequences are either denoted as functions of an integer n or its product nT with a fixed sample period T.[2]

A microcontroller, such as the myRIO, frequently has at least a few analog inputs (AIs): voltage measurement channels with ADC. The ADC takes time, so a lag is introduced in the process. Microcontrollers with an FPGA (like the myRIO) can frequently use it perform much faster ADC than those without.

For details about programming the myRIO analog inputs, see Resource 14.



12 bit resolution

$2^{12} = 4096$

Digital-to-analog conversion and analog outputs

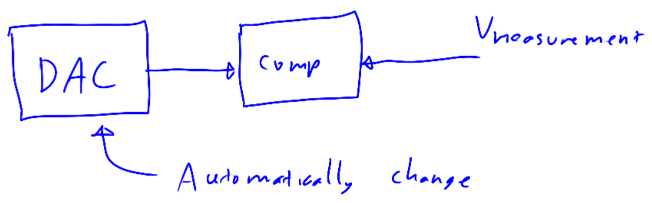
Microcontrollers also commonly have at least a few analog outputs (AOs): channels that can programmatically produce an analog signal voltage in some range (usually something like $[0, 5]$ V or $[-10, +10]$ V). But, as we know, that continuous output once had a digital representation, so it can only construct a continuous representation of some finite number of voltage levels.

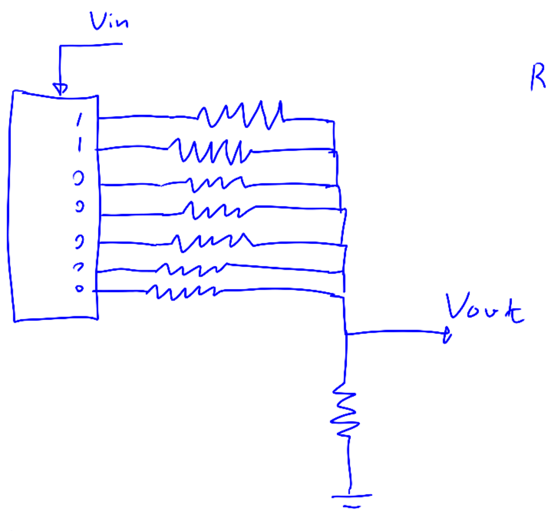
The process of converting a digital signal to an analog signal is called digital-to-analog conversion (DAC). It is essentially the opposite process as ADC; the key move is to convert a discrete signal to a continuous signal.

Considered broadly, it is the approximation of a discrete function by a continuous one, which is called "curve fitting": an attempt to fill-in continuous values in the intermediate time between discrete samples.

There is a tradeoff here between accuracy and speed because curve fitting requires computation time. In embedded computing applications, it is frequently more-important to immediately represent the realtime output than to be accurate with intermediate approximations, especially if (as is usually the case) output resolution is sufficient. This suggests the ubiquitous method of the zero-order hold, which simply maintains the previous sample value throughout the intermediate sample period, yielding a step-like analog signal. While this introduces some high-frequency noise, it is usually the best option.

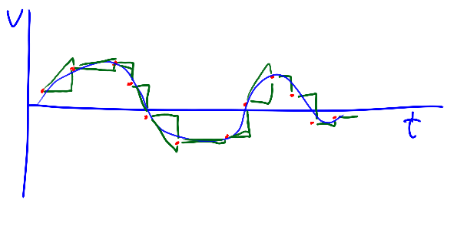
For details about programming the myRIO analog outputs, see Resource 14.





$$R_{eq} = \frac{1}{\frac{1}{R} + \frac{1}{R} + \cdots \frac{1}{R}} = \frac{1}{\frac{n}{R}} = \frac{R}{n}$$

$$V_{out} = V_{in} \frac{R/n}{R/n + R} = V_{in} \frac{R}{R + nR} = V_{in} \frac{1}{1+n}$$



---

[1] Sometimes this is called a digital signal, but we reserve that term for (usually binary-) quantized signals.

[2] These are distinct mathematical functions. Sometimes we call a function of integer n a sequence and a function of nT a discrete function. Even this obscures some details, but a precise mathematical treatment is unnecessary for our purposes.