ADC Architectures VII: Counting ADCs

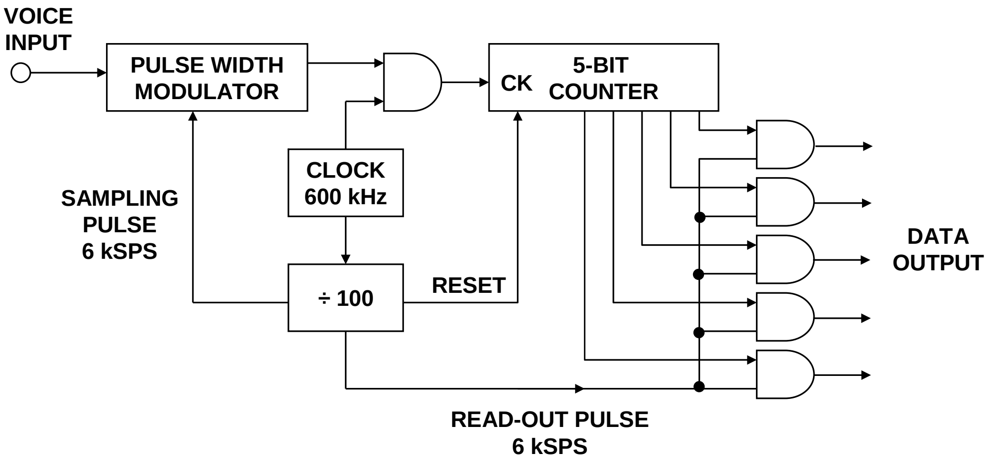
by Walt Kester and James Bryant

INTRODUCTION

Although counting-based ADCs are not well suited for high speed applications, they are ideal for high resolution low frequency applications, especially when combined with integrating techniques such as dual-, triple-, and quad-slope.

A. H. REEVES' 5-BIT COUNTING ADC

Counting ADCs had their origins in early experimental pulse code modulation (PCM) systems during the late 1930s. The first ADC suitable for PCM applications was the one documented by A. H. Reeves in his comprehensive 1939 PCM patent (Reference 1). A simplified diagram of the ADC is repeated here in Figure 1. The early ADCs for PCM typically had 5 to 7 bits of resolution and sampling rates of 6 kSPS to 10 kSPS. Interestingly enough, Reeves' ADC was based on a counting technique, probably because of his general interests in counters—the Eccles-Jordan bistable multivibrator had been invented only a few years earlier. However other architectures such as flash (Tutorial MT-020), successive approximation (Tutorial MT-021), subranging and pipelined (Tutorial MT-024), and bit-per-stage (Tutorial MT-025) were much more widely used in later PCM applications.



Adapted from: Alec Harley Reeves, "Electric Signaling System," *U.S. Patent 2,272,070*, Filed November 22, 1939, Issued February 3, 1942

Figure 1: A. H. Reeves' 5-bit Counting ADC

The counting ADC technique basically uses a sampling pulse to take a sample of the analog signal, set an R/S flip-flop, and simultaneously start a controlled ramp voltage. The ramp voltage is compared with the input, and when they are equal, a pulse is generated which resets the R/S flip-flop. The output of the flip-flop is a pulse whose width is proportional to the analog signal at the sampling instant. This pulse width modulated (PWM) pulse controls a gated oscillator, and the number of pulses out of the gated oscillator represents the quantized value of the analog signal. This pulse train can be easily converted to a binary word by driving a counter. In Reeves' system, a master clock of 600 kHz was used, and a 100:1 divider generated the 6-kHz sampling pulses. The system uses a 5-bit counter, and 31 counts (out of the 100 counts between sampling pulses) therefore represents a full-scale signal. The technique can obviously be extended to higher resolutions.

CHARGE RUN-DOWN ADC

The charge run-down ADC architecture (see Reference 2) shown in Figure 24 first samples the analog input and stores the voltage on a fixed capacitor. The capacitor is then discharged with a constant current source, and the time required for complete discharge is measured using a counter. Notice that in this approach, the overall accuracy is dependent on the quality and magnitude of the capacitor, the magnitude of the current source, as well as the accuracy of the timebase.

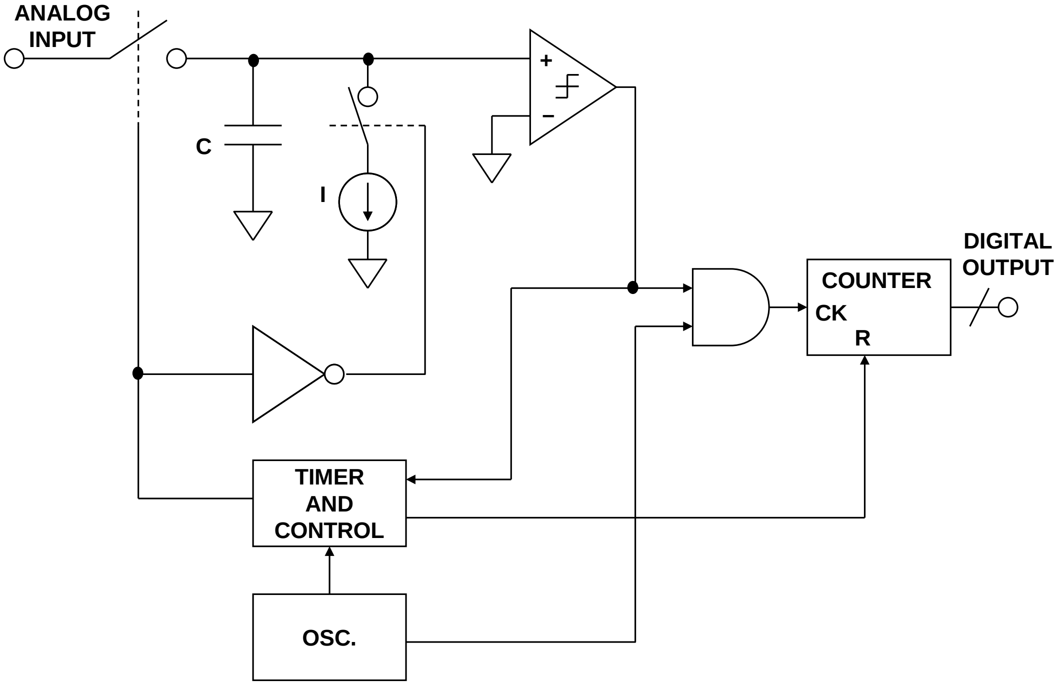


Figure 2: Charge Run-Down ADC

RAMP RUN-UP ADC

In the ramp run-up architecture shown in Figure 3 (see Reference 3), a ramp generator is started at the beginning of the conversion cycle. The counter then measures the time required for the ramp voltage to equal the analog input voltage. The counter output is therefore proportional to the value of the analog input. In an alternate version (shown dotted in Figure 3), the ramp voltage generator is replaced by a DAC which is driven by the counter output. The advantage of using the ramp is that the ADC is always monotonic, whereas overall monotonicity is determined by the DAC when it is used as a substitute.

The accuracy of the ramp run-up ADC depends on the accuracy of the ramp generator (or the DAC) as well as the oscillator. In order to process ac signals, a sample-and-hold must be used such that the analog input is stable during the conversion cycle. Note that the ramp run-up architecture is quite similar to the Reeves' counting architecture shown in Figure 1.

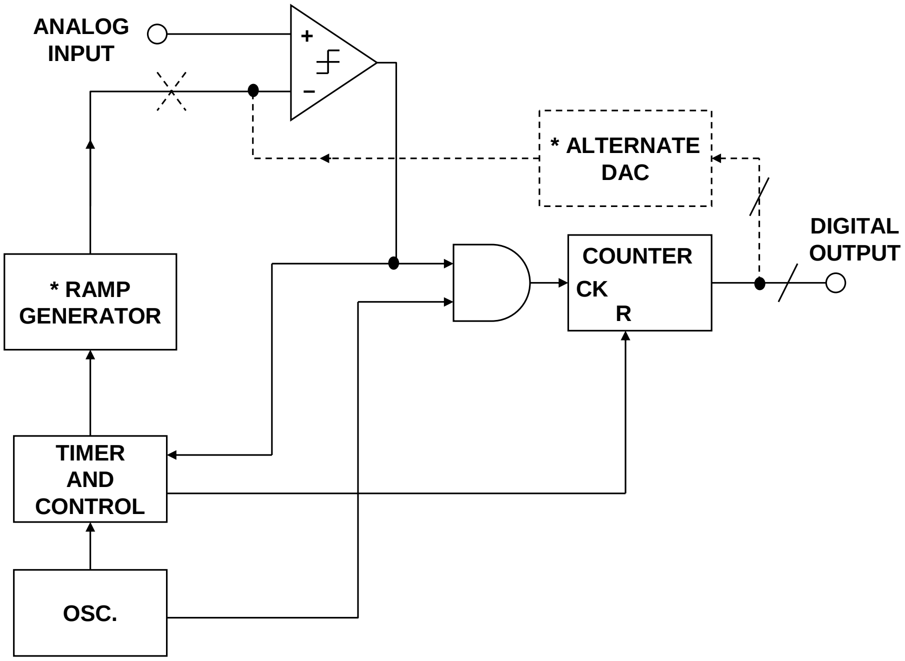


Figure 3: Ramp Run-Up ADC

TRACKING ADC

The tracking ADC architecture shown in Figure 4 (see References 4 and 5) continually compares the input signal with a reconstructed representation of the input signal. The up/down counter is controlled by the comparator output. If the analog input exceeds the DAC output, the counter counts up until they are equal. If the DAC output exceeds the analog input, the counter counts down until they are equal. It is evident that if the analog input changes slowly, the counter will follow, and the digital output will remain close to its correct value. If the analog input suddenly undergoes a large step change, it will be many hundreds or thousands of clock cycles before the output is again valid. The tracking ADC therefore responds quickly to slowly changing signals, but slowly to a quickly changing one.

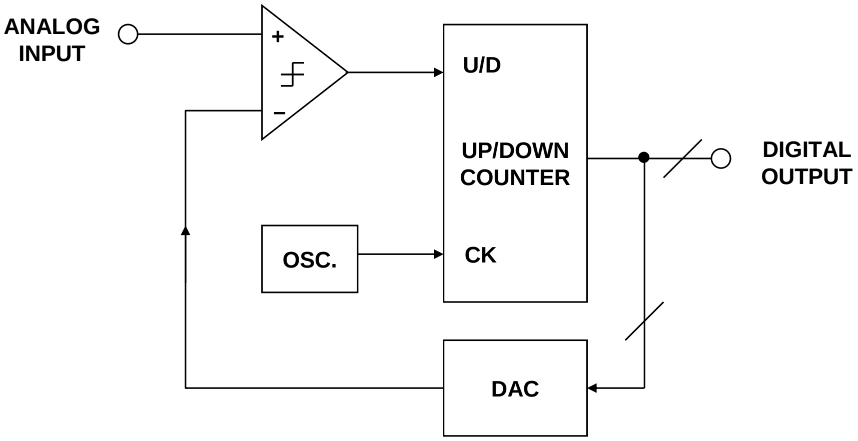


Figure 4: Tracking ADC

The simple analysis above ignores the behavior of the ADC when the analog input and DAC output are nearly equal. This will depend on the exact nature of the comparator and counter. If the comparator is a simple one, the DAC output will cycle by 1 LSB from just above the analog input to just below it, and the digital output will, of course, do the same—there will be 1 LSB of flicker. Note that the output in such a case steps every clock cycle, irrespective of the exact value of analog input, and hence always has unity Mark/Space ratio. In other words, there is no possibility of taking a mean value of the digital output and increasing resolution by oversampling.

A more satisfactory, but more complex arrangement would be to use a window comparator with a window 1-2 LSB wide. When the DAC output is high or low the system behaves as in the previous description, but if the DAC output is within the window, the counter stops. This arrangement eliminates the flicker, provided that the DAC DNL never allows the DAC output to step across the window for 1 LSB change in code.

Tracking ADCs are not very common. Their slow step response makes them unsuitable for many applications, but they do have one asset: their output is *continuously* available. Most ADCs perform conversions: i.e., on receipt of a "start convert" command (which may be internally generated), they perform a conversion and, after a delay, a result becomes available. Providing that the analog input changes slowly, the output of a tracking ADC is always available. This is valuable in synchro-to-digital and resolver-to-digital converters (SDCs and RDCs), and this is the application where tracking ADCs are most often used (see Tutorial MT-030). Another valuable characteristic of tracking ADCs is that a fast transient on the analog input causes the output to change only one count. This is very useful in noisy environments. Notice the similarity between a tracking ADC and a successive approximation ADC. Replacing the up/down counter with SAR logic yields the architecture for a successive approximation ADC.

REFERENCES

1. Alec Harley Reeves, "Electric Signaling System," *U.S. Patent 2,272,070*, filed November 22, 1939, issued February 3, 1942. Also *French Patent 852,183* issued 1938, and *British Patent 538,860* issued 1939. *(the ground-breaking patent on PCM. Interestingly enough, the ADC and DAC proposed by Reeves are counting types, and not successive approximation).*

2. Bernard M. Oliver and Claude E. Shannon, "Communication System Employing Pulse Code Modulation," *U.S. Patent 2,801,281*, filed February 21, 1946, issued July 30, 1957. *(charge run-down ADC and Shannon-Rack DAC).*

3. Arthur H. Dickinson, "Device to Manifest an Unknown Voltage as a Numerical Quantity," *U.S. Patent 2,872,670*, filed May 26, 1951, issued February 3, 1959. *(ramp run-up ADC).*

4. K. Howard Barney, "Binary Quantizer," *U.S. Patent 2,715,678*, filed May 26, 1950, issued August 16, 1955. *(tracking ADC).*

5. Bernard M. Gordon and Robert P. Talambiras, "Information Translating Apparatus and Method," *U.S. Patent 2,989,741*, filed July 22, 1955, issued June 20, 1961. *(tracking ADC).*