DIGITAL–TO-ANALOG CONVERTERS:

V-F, and F-V converters, performance specifications, D-A conversion techniques (R-2R & binary weighted) multiplying DACapplications. A-D conversion techniques (flash, successive approximation, single slope, dual slope), over sampling converters.

The function of a ***voltage-to-frequency converter*** (VFC) is to accept an analog input v_I and generate a pulse train with frequency,

$$f_O = kv_I$$

where *k* is the VFC ***sensitivity,*** in hertz per volt. As such, the VFC provides a simple form of analog-to-digital conversion.

The primary reason for this type of conversion is that a pulse train can be transmitted and decoded much more accurately than an analog signal, especially if the transmission path is long and noisy.

If electrical isolation is also desired, it can be accomplished without loss of accuracy

using inexpensive optocouplers or pulse transformers. Moreover, combining a VFC with

a binary counter and digital readout provides a low-cost digital voltmeter.

VFCs usually have more stringent performance specifications than VCOs. Typical requirements are:

- wide dynamic range (four decades or more),
- the ability to operate to relatively high frequencies (hundreds of kilohertz, or higher),
- Low linearity error (less than 0.1% deviation from the straight line going from zero to the full scale),
- high scale-factor accuracy and stability with temperature and supply voltage.

The output waveform, on the other hand, is of secondary concern as long as its levels are

compatible with standard logicsignals. VFCs fall into two categories: ***wide-sweep***

multivibrators and ***charge-balancing VFCs***

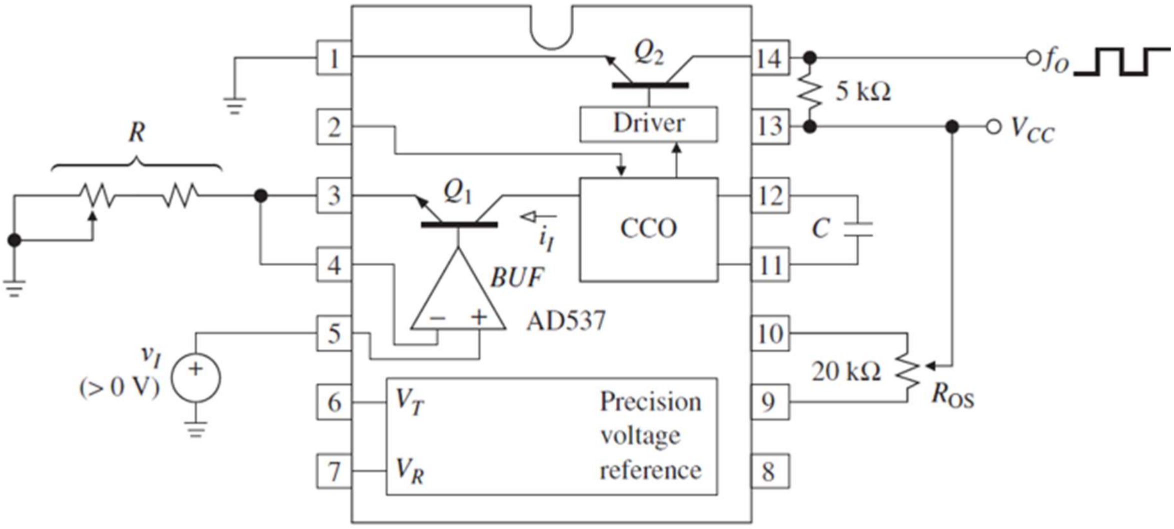
Wide-Sweep Multivibrator VFC

These circuits are essentially voltage-controlled astable multivibrators designed with VFC performance specifications in mind.

A popular product in this category is the **AD537** The op amp and Q_1 form a buffer *V-I* converter that converts v_I to the current drive i_I for the CCO according to $i_I = v_I / R$.

The CCO(current controlled oscillator) parameters have been chosen so that $f_O = i_I / (10C)$, or

$$f_O = \frac{v_I}{10RC}$$



The AD537 voltage-to-frequency converter

This relationship holds fairly accurately over a dynamic range of at least four decades, up to a full-scale current of 1 mAand a full-scale frequency of 100 kHz. For instance, with **C = 1 nF, R = 10 kΩ, and VCC = 15 V,** varying v_I from 1 mV to 10 V varies i_I **from 0.1 μA to 1 mA and f_O from 10 Hz to 100 kHz**.

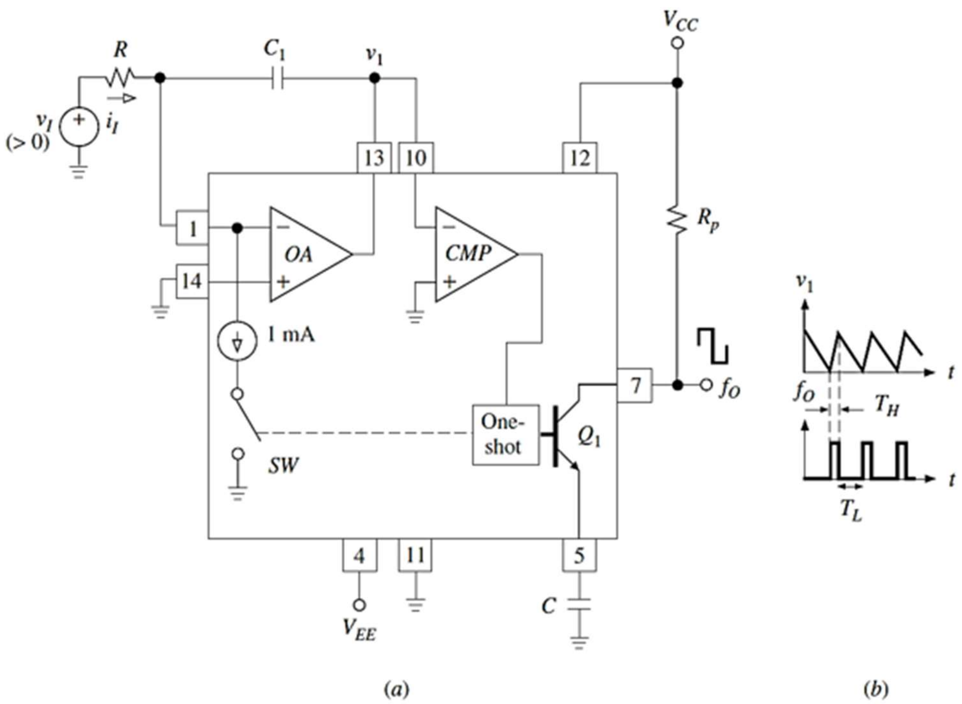
The device can also function as a **current-to-frequency converter** (CFC) if we make the control current flow out of theinverting input node. For instance, grounding pin 5 and replacing *R* by a photodetector diode current sink will convert light intensity to frequency.

The AD537 also includes an on-chip precision voltage reference to stabilize the CCO scale factor. This yields a typical **thermal stability of 30 ppm/◦C.** To further enhance the versatility of the device, two nodes of the reference circuitry aremade available to the user, namely, V_R and V_T .

Figure also shows another useful feature of the AD537, namely, the ability to transmit information over a twisted pair. This pair serves the dual purpose of supplying power to the device and carrying frequency data in the form of current modulation.

Charge-Balancing VFC

The **charge-balancing technique** supplies a capacitor with continuous charge at a rate that is linearly proportional to the input voltage v_I , while simultaneously pulling discrete charge packets out of thecapacitor at a rate f_O such that the net charge flow is always zero. The result is $f_O = kv_I$



OA converts v_I to a current $i_1 = v_1/R$ flowing into the summing junction; the value of *R* is chosen such that we always have $i_I < 1$ mA. With *SW* open, i_I flows into *C*1 and causes *v*1 to ramp downward.

As soon as $v1$ reaches 0 V, *CMP* fires and triggers a precision one-shot that closes *SW* and turns on *Q*1 for a time interval T_H set by *C*. The one-shot, whose details have been omitted for simplicity, uses a threshold of 7.5 V and a charging current of 1 mA to give

$$T_H = \frac{7.5\ \text{V}}{1\ \text{mA}} C$$

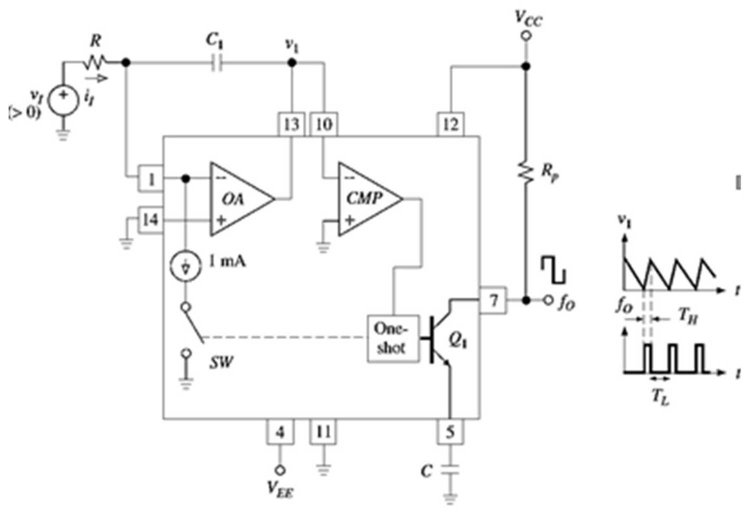
The closure of *SW* causes a net current of magnitude $(1\ \text{mA} - i_I)$ to flow out of the summing junction of *OA*. Consequently, during T_H, v_1 ramps upward by an amount $\Delta v_1 = (1\ \text{mA} - i_I)T_H/C_1$. After the one-shot times out, *SW* is opened and v_1 resumes ramping downward at a rate again set by i_I. The time T_L it takes for v_1 to return to zero is such that $T_L = C_1 \Delta v_1 / i_I$. Eliminating Δv_1 and letting $f_O = 1/(T_L + T_H)$ gives, with the help of Eq. (10.29),

$$f_O = \frac{v_I}{7.5RC}$$

where f_O is in hertz, v_I in volts, R in ohms, and C in farads. As desired, f_O is linearly proportional to v_I. Moreover, the duty cycle $D(\%) = 100 \times T_H/(T_H + T_L)$ is readily found to be

$$D(\%) = 100 \frac{v_I}{R \times 1\ \text{mA}}$$

and it is also proportional to v_I. For best linearity, the data sheets recommend designing for a maximum duty cycle of 25%, which corresponds to $i_{I(\max)} =$ 0.25 mA.



(a) (b)

The absence of *C*1 from the above equations indicates that thetolerance and drift of this capacitor are not critical, so its value can be chosen arbitrarily. However, for optimum performance, the data sheets recommend using the value of *C*1 that yields *v*1~= 2.5 V. *C*, on the other hand, so it must be a low-drift type, such as NPO ceramic. If *C* and *R* have equal but opposing thermal coefficients, the overall drift can be reduced to as little as **20 ppm/◦C**. Foraccurate operation to low values of v_I, the input offset voltage of OA must be nulled

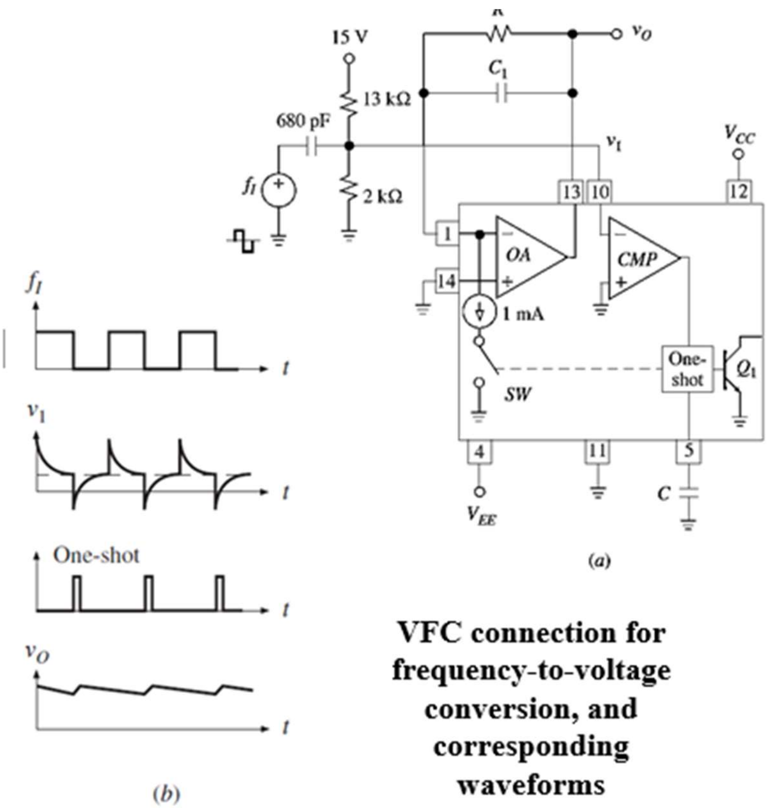
Frequency-to-Voltage Conversion

The ***frequency-to-voltage converter*** (FVC) performs the inverse operation, namely, it accepts a periodic waveform of frequency f_I and yields an analog output voltage:

$$v_O = kf_I$$

where ***k*** is the FVC sensitivity, in volts per hertz. FVCs find application as tachometers in motor speed control and rotational measurements. Moreover, they are used in conjunction with VFCs to convert the transmitted pulse train back to an analog voltage

A charge balancing VFC can easily be configured as an FVC by applying the periodic input to the comparator and deriving the output from the op amp, which now has the resistance R in the feedback path (see fig)



VFC connection for frequency-to-voltage conversion, and corresponding waveforms

The input signal usually require proper conditioning to produce a voltage with reliable zero crossing for CMP. Shown in the figure is a high pass network to accommodate inputs of the TTL and CMOS type.

On each negative spike of v_1, *CMP* triggers the one-shot, closing *SW* and pulling 1 mA out of *C*1 for a duration *TH.* In response to this train of current pulses, v_O builds up until the current pulled out of the summing junction of *OA* in 1-mA packets is exactly counterbalanced by that injected by v_o via R continuously, or f I x 10^{-3} x TH = v_o/R

olving for V_o gives

$$v_O = 7.5RCf_I$$

The value of C is determined on the basis of a maximum duty cycle of 25 percent, as discussed earlier, while *R* now establishes the full-scale value of v_O. As in the VFC case, the input offset voltage of *OA* should be nulled to avoid degrading the conversion accuracy at the low end of the range.

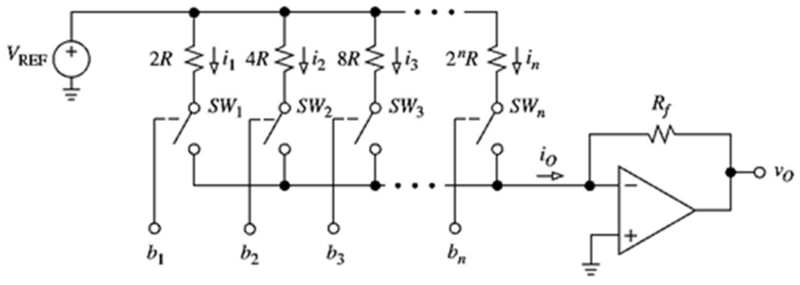
D-A CONVERSIONTECHNIQUES

Weighted Resistor DAC

The functions required to implement an ***n*-bit DAC** are n switch and n binary- weighted variables to synthesize the terms $b_k 2^{-k}$, ***k* = 1, 2, . . . , *n*;** moreover, we need an *n*-input summer, and a reference

The DAC of Fig uses an op amp to sum *n* binary-weighted currents derived from V_{REF} via the current-scaling resistances 2*R,* 4*R,* 8*R,* . . . , 2^n *R*. Whether the current $i_k = V_{REF}/2^k R$ appears in the sum depends on whether the corresponding switch is closed (b_k = 1) or open (b_k = 0). Writing $v_O = -R_f i_O$ gives:

$$v_O = (-R_f/R)V_{REF}(b_1 2^{-1} + b_2 2^{-2} + \cdots + b_n 2^{-n})$$



Weighted-resistor DAC

The conceptual simplicity of the weighted-resistor DAC is offset by two drawbacks, namely, the nonzero resistances of the switches, and a spread in the current-setting resistances that increases exponentially with n.

The effect of switch resistances is to disrupt the binary-weighted relationships of the currents, particularly in the most significant bit positions, where the current-setting resistances are smaller.

These resistances can be made sufficiently large to swamp the switch resistances; however, this may result in unrealistically large resistances in the least significant positions.

For instance, an 8-bit DAC requires resistances ranging from 2R to 256R. The difficulty in ensuring accurate ratios over a range this wide, especially in monolithic form, restricts the practicality of resistor-weighted DACs below 6 bits.

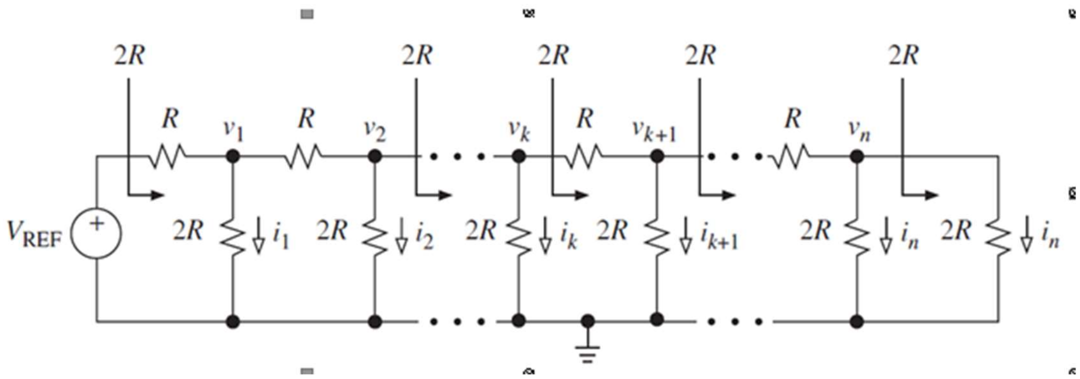
R-2*R* LadderDAC

Most DAC architectures are based on the popular *R*-2*R* ladder depicted in Fig.

The current flowing downward, away from each node, is equal to the current flowing toward the right; moreover, twice thiscurrent enters the node from the left. The currents and, hence, the node voltages are binary-weighted,

$$i_{k+1} = \frac{1}{2} i_k \qquad v_{k+1} = \frac{1}{2} v_k$$

where $k = 1, 2, \ldots, n - 1$.



With a resistance spread of only 2-to-1, *R*-2*R* ladders can be fabricated monolithically to a high degree of accuracy and stability. Thin-film ladders, fabricated by deposition on the oxidized silicon surface, lend themselves to accurate laser trimming for DACs with $n \geq 12$. For DACS with a lower number of bits, diffused or ion-implanted ladders are often adequate. Depending on how the ladder is utilized, different DAC architectures result.

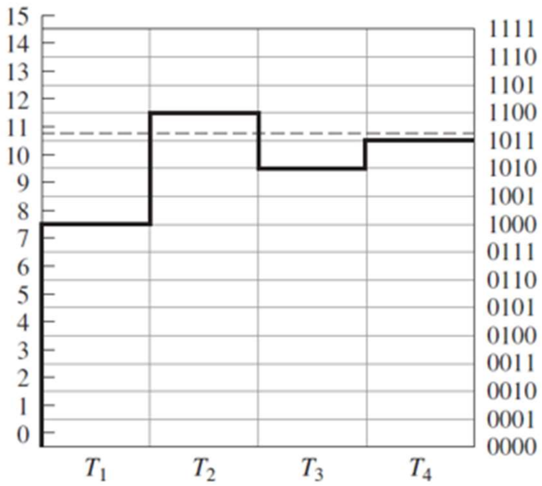
A-D CONVERSIONTECHNIQUES

Successive-Approximation Converters (SA ADCs)

This technique uses the register as a successive-approximation register (SAR) to find each bit by trial and error. Starting from the MSB, the SAR inserts a trial 1 and then interrogates the comparator to find whether this causes v_O to rise above v_I.

If it does, the trial bit is changed back to 0; otherwise, it is left as 1. The procedure is then repeated for all subsequent bits, one bit at a time.

Figure illustrates how a 10.8-V input is converted to a 4-bit code with V_{FSR} = 16 V. The analog range, in volts, is at the left, and the digital codes at the right**. To ensure correct results, the DAC output must be offset by −12 LSB, or −0.5 V in our example.**



The conversion takes place as follows:

Following the arrival of the START command, the SAR sets *b*1 to 1 with all remaining bits at 0 so that the trial code is 1000. This causes the DAC to output $\boldsymbol{v_O = 16\,(1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3} + 0 \times 2^{-4}) - 0.5 = 7.5}$ **V**. At the end of clock period *T*1, v_O is compared against v_I , and since 7.5 < 10.8, b_1 is left at 1.

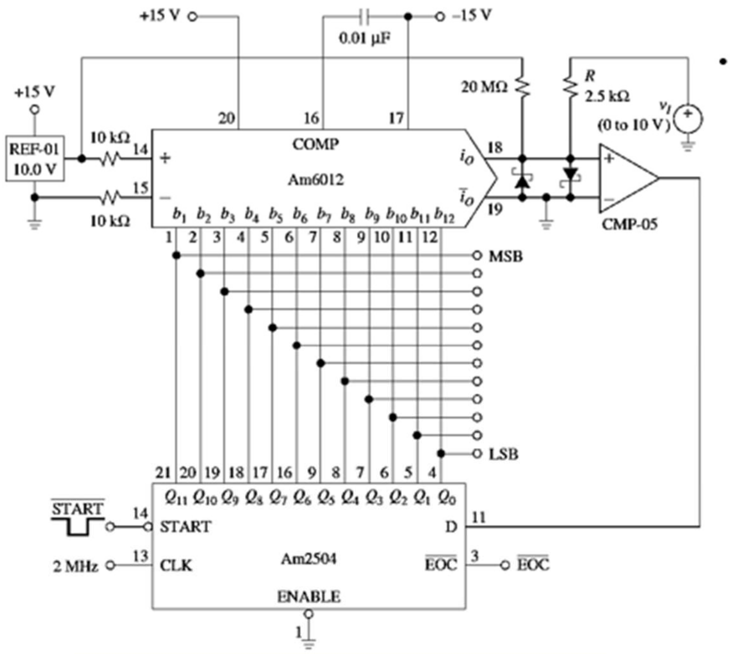
At the beginning of *T*2, b_2 is set to 1, so the trial code is now 1100 and $\boldsymbol{v_O = 16(2^{-1} + 2^{-2}) - 0.5 = 11.5}$ **V**. Since 11.5 >10.8, b_2 is changed back to 0 at the end of *T*2.

At the beginning of *T*3, b_3 is set to 1, so the trial code is 1010 and $\boldsymbol{v_O = 10 - 0.5 = 9.5}$ **V**. Since 9.5 < 10.8, b_3 is left at 1.

At the beginning of T4, b_4 is set to 1, so the trial code is 1011 and v_O = 11 − 0.5 = 10.5 V. Since 10.5 < 10.8, b4 is left at 1. Thus, when leaving T4, the SAR has generated the code 1011, which ideally corresponds to 11 V. Note that any voltage in the range 10.5 V < v_I < 11.5 V would have led to the same code.

Figure shows an actual implementation using the Am2504 SAR and the Am6012 bipolar DAC

SA ADCs are available from a variety of sources and in a wide range of performance characteristics and prices. Conversion times typically range from under 1 µs for the faster 8-bit units to tens of microseconds for the high- resolution (n ≥14) types. SA ADCs equipped with an on-chip SHA are referred to as sampling ADCs.A popular example is the AD1674 12-bit, 100-kilosamples per second (ksps) SA ADC.



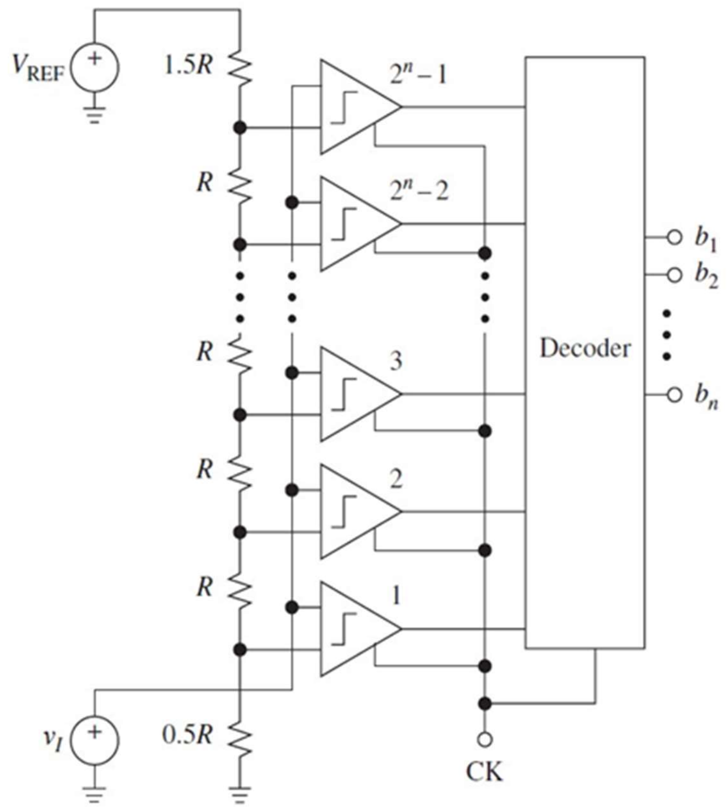
12-bit, 6-µs successive-approximation ADC.

FlashConverters

The circuit of Figure uses a resistor string to create 2n − 1 reference levels separated from each other by 1 LSB, and a bank of 2n − 1 high- speed latched comparators to simultaneously compare vI against each level.

Note that to position the analog signal range properly, the top and bottom resistors must be 1.5R and 0.5R, as shown. As the comparators are strobed by the clock, the ones whose reference levels are below vI will output a logic 1, and the remaining ones a logic 0.

The result, referred to as a bar graph, or also as a thermometer code, is then converted to the desired output code b1 . . . bn by a suitable decoder, such as a priority encoder.



Since input sampling and latching take place during the first phase of the clock period, and decoding during the second phase, the entire conversion takes only one clock cycle, so this ADC is the fastest possible.

Aptly called a flash converter, it is used in high-speed applications, such as video and radar signal processing, where conversion rates on the order of millions of samples per second (Msps) are required, and SA ADCs are generally not fast enough. The high-speed and inherent-sampling advantages of flash ADCs are offset by the fact that 2n−1 comparators are required.

For instance, an 8-bit converter requires 255 comparators. The exponential increase with n in die area, power dissipation, and stray input capacitance makes flash converters impractical for n > 10.

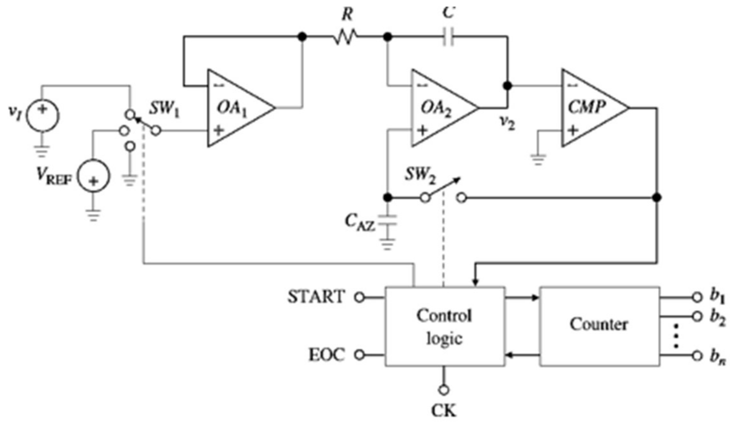
Flash ADCs are available in bipolar or in CMOS technology, with resolutions of 6, 8, and 10 bits, sampling rates of tens to hundreds of Msps, depending on resolution, and power dissipation ratings on the order of 1 W or less.

Dual-slopeADC

Integrating-type converters perform A-D conversion indirectly by converting the analog input to a linear function of time and hence to a digital code. The two most common converter types are the charge- balancing and dual-slope ADCs.

As shown in the functional diagram of Fig, a dual-slope ADC, also called a dual-ramp ADC, is based on a high- input-impedance buffer, a precision integrator, and a voltage comparator.

The circuit first integrates the input signal vI for a fixed duration of 2n clock periods, and then it integrates an internal reference VREF of opposite polarity until the integrator output is brought back to zero.



Functional diagram of a dual-slope ADC

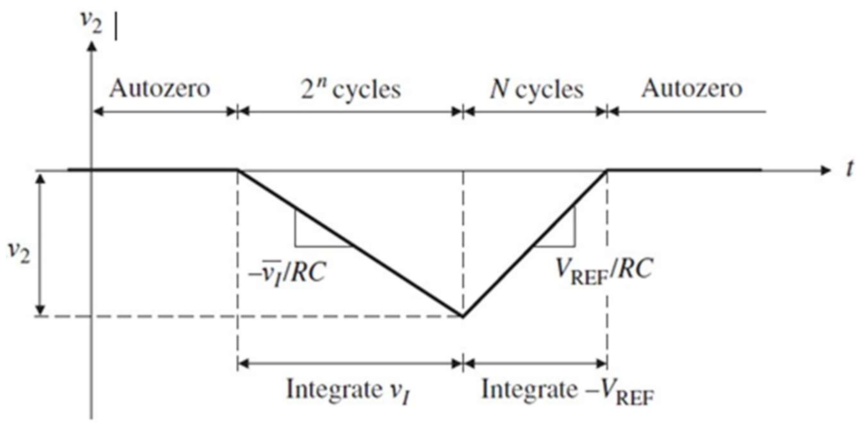
The number N of clock cycles required to return to zero is proportional to the value of vI averaged over the integration period. Consequently, N represents the desired output code.

With reference to the waveform diagram of Fig., following is a detailed description of how the circuit operates.

Prior to the arrival of the START command, SW1 is connected to ground and SW2 closes a loop around the integrator-comparator combination. This forces the autozero capacitance CAZ to develop whatever voltage is needed to bring the output of OA2 right to the comparator's threshold voltage and leave it there. This phase, referred to as the autozero phase, provides simultaneous compensation for

the input offset voltages of all three amplifiers. During the subsequent phases, when SW2 opens, CAZ acts as an analog memory to hold the voltage required to keep the net offset nulled.

At the arrival of the START command, the control logic opens SW2, connects SW1 to vI (which we assume to be positive), and enables the counter, starting from zero. This phase is called the signal integrate phase. As the integrator ramps downward, the counter counts until, 2n clock periods later, it overflows. This marks the end of the current phase.



As the overflow condition is reached, the counter resets automatically to zero and SW1 is connected to −VREF, causing v2 to ramp upward. This is called the deintegrate phase. Once v2 again reaches the comparator threshold, the comparator fires to stop the counter and issues an EOC command. The accumulated count N is such that CΔv2 = (VREF/R)NTCK. Since CΔv2 is the same during the two phases, we get

$$N = 2^n \frac{\overline{v_I}}{V_{\mathrm{REF}}}$$