SUMMING CIRCUITS

Electronic systems perform operations on information signals represented by electrical variables. Among these operations, one of the most simple, and therefore most common and widely used, is the *summation* or addition of signals, with a large range of variations.

Signal addition is a fundamental operation upon which many other simple and complex operators are based. For example, digital processing circuits perform most of their operations from some combination of addition and shifting.

Signal-processing circuits can be classified into two large groups, according to the way in which they perform their processing function: analog and digital processing systems. *Analog* processing systems operate on signals defined over a continuous range or interval. These signals are generally associated with the continuous value of some electrical magnitude of the circuit, like a voltage, current, charge, or magnetic flux defined in connection with some element or electrical nodes in the network. Analog summing operations emerge from the basic laws of electromagnetism, Maxwell equations, and in particular their derived result in circuit theory (Kirchhoff's Laws). On the other hand, *digital* processing systems operate over abstract number representations, generally codified as a sequence of digits, without a direct connection to a physical magnitude in the network. Possible values of each digit are discrete (two in the typical binary case), and each value is associated to some range of an electrical variable. In these cases, summation follows algorithmic rules not related to circuit theory.

ANALOG ADDERS

Introduction

An electronic analog adder operates over two or more electrical input signals ($x_1, x_2, \ldots, x_N$), providing an electrical output signal x_s, which is the sum of the former

$$x_S = x_1 + x_2 + \cdots + x_N \tag{1}$$

The electrical signals involved may be of the same or different nature (voltage, current, charge, flux). Often, input signals are all of the same type. If signals of different nature are combined, appropriate constants must be introduced to maintain a proper dimensionality (1). Such signal multiplication by a constant or *scaling,* is often found as a constitutive operator of analog summing circuits, as shall be seen, even when all input signals are of the same nature. The involved constants are always related to physical properties of one or more circuit elements and, in some cases, to other electrical signals.

The number of analog summing circuits reported in the literature is extremely large. Therefore, a classification is mandatory for their discussion. This classification can be made according to the nature of the signals to be added (current or voltage) and also by the nature of the output signal.

Another distinctive property is the way in which the time evolution of the input signals is taken into account: continuous-time or sampled-data systems. Although in either case the expected input and output signals are continuous in amplitude (i.e., analog), *continuous-time* signals are defined at any instant within some time interval, and the corresponding systems operate continuously in time. On the other hand, the information carried by *discrete-time* signals is defined only for discrete-time instants, and the corresponding sampled-data systems operate under the control of one or more periodic clock signals.

An alternative form of classification refers to the circuit technique employed to find the summing circuit, in particular to the type of elements required for its implementation: resistors, capacitors, analog switches, operational amplifiers, current conveyors, transistors, operational transconductance amplifiers, and the like.

Analog Summation Fundamentals

Analog summing circuits are essentially based on Kirchhoff Laws. To be precise, most analog summing circuits rely on Kirchhoff Current Law (KCL), which can be formulated as follows: The algebraic sum of currents flowing into any network node is zero. According to this law, the summation of any number of current signals is straightforward. It can be obtained by simply connecting the circuit branches carrying the currents to be added to a common node and allowing an additional branch for the sum of the currents to flow out of the node. Obviously, the current branches must have one of the nodes available to be connected to the summing node.

The Kirchhoff Voltage Law introduces a similar tool for voltage signals. It can be formulated as follows: The algebraic sum of the potential drops around any loop is zero. According to this law, the summation of voltage signals is also straightforward and can be obtained from a series connection of the pair of terminals between which the incoming voltage signals are defined. Because the incoming voltage sources are connected in series, they must be "floating," that is, the voltage drops must be insensitive to a shift of their terminal voltages with respect to an arbitrary reference level.

Although summing circuits based on either of the two approaches are possible, voltage signals found in most electronic circuits are defined as the potential at a particular node measured from a common ground reference: therefore, floating voltage signals are not usual. For this reason, most analog summing circuits rely on KCL, as stated previously.

Still, KCL-based summing circuits can operate on voltage signals, as long as the incoming voltages are transformed into currents, added at a common node, and the resulting current transformed again into a voltage. This is the most common operation principle of summing circuits, found in traditional and well-known structures.

Because summation is linear by definition, voltage-to-current and current-to-voltage transformations should be linear.

In its simplest form, these transformations can be obtained from linear two-terminal elements, whose constitutive equations are defined as

$$V = Z \cdot I \qquad \text{or} \qquad I = Y \cdot V \tag{2}$$

where Z and Y are the impedance and admittance of the two-terminals element, respectively. It is obvious that $Z = 1/Y$.

Figure 1 shows a conceptual example of voltage-signals summation based on KCL. Simple analysis results in the following expression for the output voltage:

$$V_o = \sum_{i=1}^{N} \frac{Z_F}{Z_i} V_i = \sum_{i=1}^{N} \frac{Y_i}{Y_F} V_i \tag{3}$$

Elements Z_1 to Z_N perform a linear transformation of the input voltages into currents. These currents are added at node N_s according to KCL. The resulting current I_s is reproduced by the current-controlled current-source (CCCS) and linearly transformed into the output voltage V_o with the help of element Z_F. Note that the output voltage represents a weighted summation of the input voltage signals, all of them defined with respect to a common voltage reference.

The intermediate transformation into currents results in different scaling factors or *weights* W_i for each input signal

$$W_i = \frac{Y_i}{Y_F} = \frac{Z_F}{Z_i} \tag{4}$$

allowing the output voltage to be expressed as

$$V_o = \sum_{i=1}^{N} W_i V_i \tag{5}$$

The impedance Z_F of the output element provides a "gain" control for the output signal. Scaling weights and global gain control introduce a large flexibility in these summing operators. Some of these capabilities rely on an active element (in this example a CCCS) to implement a virtual ground at node N_s, which in turn allows the transformation of current I_s into the output voltage without altering the individual currents

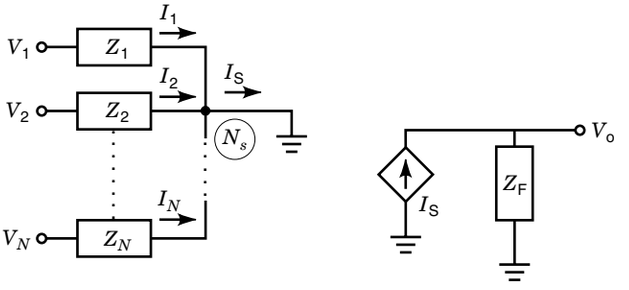


Figure 1. Conceptual example of voltage-signals summation based on KCL. This is the underlying fundamental of most analog summing circuits.

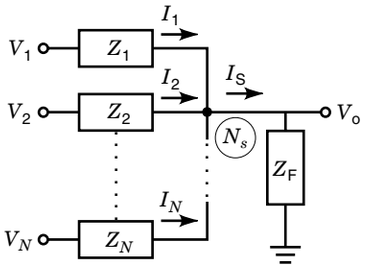


Figure 2. Voltage-signals summation without active element. Scaling factors become interdependent.

I_i. This will be illustrated in connection with Fig. 2, a simplified version of Fig. 1 in which the active element has been eliminated. Simple analysis results in the following expression for the new output voltage:

$$V_o = \frac{\sum_{i=1}^{N} Y_i V_i}{\sum_{i=1}^{N} Y_i + Y_F} \tag{6}$$

which is still a weighted summation of the input voltages. Now, however, the scaling factors associated with the different input signals cannot be controlled independently. Furthermore, in the simplest case in which all elements are resistors, the weighting factors will all be smaller than one. In addition, the effect of the external load on the output voltage must be carefully considered. Because of these and other related reasons, most analog summing circuits employ some type of active element.

Analog Summing Circuits with Operational Amplifiers

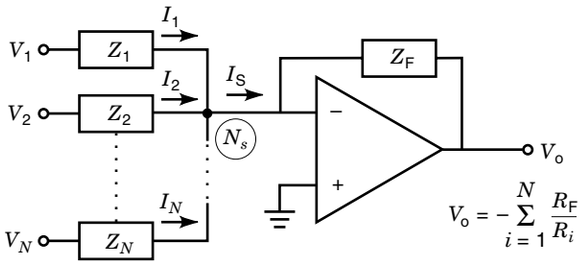
A simple and well-known alternative for a practical implementation of Fig. 1 uses operational amplifiers as active devices and linear resistors as two-terminal elements. Other simple two-terminal elements can also be employed (e.g., capacitors), although special considerations are required. In what follows, we assume the use of linear resistors $R = 1/G$ for simplicity.

The resulting circuit is shown in Fig. 3. In this circuit, current I_s flows from the virtual ground node toward the output terminal, resulting in a sign inversion in the summing operation. This summing circuit, in which all weights are negative, is commonly referred to as an inverting configuration.

Positive weights can be achieved by several means, the simplest of them being the use of an additional inverting configuration with a single input (i.e., an inverting amplifier), as shown in Fig. 4.

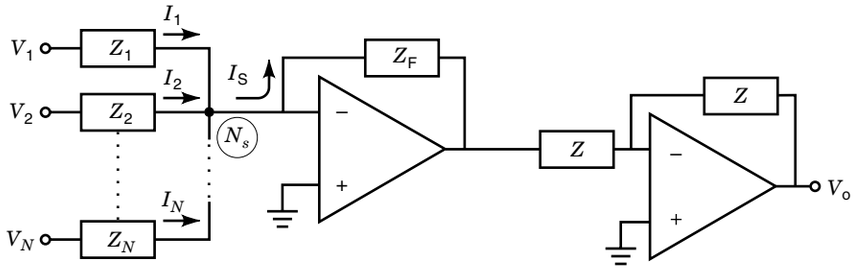
Another alternative results from the combination of the circuit in Fig. 2 with a noninverting amplifier. This solves the restriction on the weights to values smaller than unity and provides the necessary driving capability for the output voltage. The resulting circuit, commonly known as a noninverting configuration, is shown in Fig. 5. Note that the grounded input element is used as an additional input.

The circuits in Figs. 3–Fig. 5 are restricted to "same-sign" weights. When a summing circuit with both positive and neg-



$$V_o = -\sum_{i=1}^{N} \frac{R_F}{R_i} V_i = -\sum_{i=1}^{N} \frac{G_i}{G_F} V_i = -\sum_{i=1}^{N} W_i V_i$$

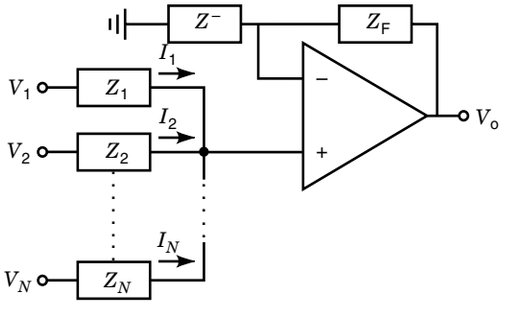
Figure 3. Voltage-summing circuit using operational amplifiers (inverting configuration).



$$V_o = \sum_{i=1}^{N} W_i V_i$$

$$W_i = \frac{G_i}{G_F} = \frac{Z_F}{Z_i}$$

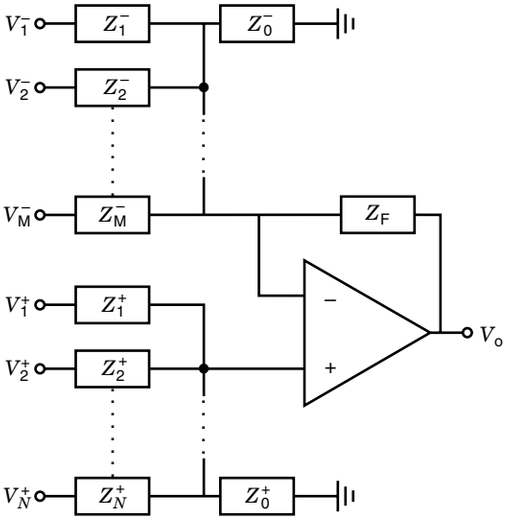
Figure 4. Positive-weight summing circuit using two inverting configurations.



$$V_o = \left(1 + \frac{G_-}{G_F}\right) \frac{\sum_{i=1}^{N} G_i V_i}{\sum_{i=1}^{N} G_i}$$

$$W_i = \left(1 + \frac{G_-}{G_F}\right) \frac{G_i}{\sum_{i=1}^{N} G_i}$$

Figure 5. Positive-weight summing circuit using a noninverting amplifier.



$$V_o = \frac{G_F + \sum_{j=0}^{M} G_j^-}{\sum_{i=0}^{N} G_i^+} \sum_{i=1}^{N} \frac{G_i^+}{G_F} V_i^+ - \sum_{j=1}^{M} \frac{G_j^-}{G_F} V_j^-$$

Figure 6. Generalized summing circuit based on an operational amplifier.

ative weights is required, some of the input signals can be inverted at the expense of an additional active element per inverted input. Another alternative is the use of the so-called *generalized adder,* obtained from a combination of the inverting and noninverting configurations in Figs. 3 and 5. The resulting circuit and its transfer function are shown in Fig. 6.

As with Fig. 3, negative weights are given by the ratio of the input G_j^- to the feedback G_F conductances and are independent of each other. On the other hand, positive weights depend on all input conductances. In order to eliminate this drawback, two additional elements Z_0^+ and Z_0^- are introduced to allow for the possibility of making

$$\frac{G_F + \sum_{j=0}^{M} G_j^-}{\sum_{i=0}^{N} G_i^+} = 1 \tag{7}$$

which allows the transfer function of the generalized adder to be expressed as

$$V_o = \sum_{i=1}^{N} W_i^+ V_i - \sum_{j=1}^{M} W_j^- V_j^- \tag{8}$$

with positive and negative weights having similar expressions

$$W_i^+ = \frac{G_i^+}{G_F} \qquad \text{and} \qquad W_j^- = \frac{G_j^-}{G_F} \tag{9}$$

Note that Eq. (7) can also be written as

$$1 + \sum_{j=1}^{M} W_j^- + \frac{G_0^-}{G_F} = \sum_{i=1}^{N} W_i^+ + \frac{G_0^+}{G_F} \tag{10}$$

Therefore, if

$$1 + \sum_{j=1}^{M} W_j^- < \sum_{i=1}^{N} W_i^+ \tag{11}$$

we can select $G_0^+ = 0$ and

$$G_0^- = G_F \left(\sum_{i=1}^{N} W_i^+ - \sum_{j=1}^{M} W_j^- - 1 \right) \tag{12}$$

On the other hand, if Eq. (11) is not true, we can select $G_0^- = 0$ and

$$G_0^+ = G_F \left(1 + \sum_{j=1}^{M} W_j^- - \sum_{i=1}^{N} W_i^+ \right) \tag{13}$$

Therefore, only one element among Z_0^+ and Z_0^- is actually required.

Summing Circuits Using Current Conveyors

The use of current conveyors (see CURRENT CONVEYORS) as the required active element in Fig. 1 results in a new family of summing circuits. Indeed, because current conveyors (CC) perform as current-controlled current-sources, the resulting circuits are in fact direct implementations of Fig. 1. As in the previous case, different configurations with positive and negative weights can be obtained.

Figure 7 shows a noninverting [Fig. 7(a)] and an inverting [Fig. 7(b)] configuration. Both of them are possible with either type I or type II CCs. The two configurations differ only in the "sign" of the CCs: CC^+ for the noninverting configuration and CC^- for the inverting counterpart.

If either CC^+ or CC^- are not available, the necessary sign inversion can be achieved at the expense of an additional CC of either sign, as shown in Fig. 8. Note that the sign-inverter in Fig. 8(a) operates on a voltage signal. Its input impedance is high; therefore, it may be connected to any circuit node without affecting its behavior. On the other hand, inverting stages in Fig. 8(b, c) operate on currents and thus should be inserted in series at the output of the CC in Fig. 7, whose loading element Z_F must be eliminated.

The combination of the circuits in Figs. 7 and 8 results in either inverting or noninverting summing circuits realizable with any type of CC. If positive and negative weights are required on the same summing device, we can use inverters at specific inputs, or a generalized adder architecture based on CCs. Figure 9(a) shows an implementation based on CCI of either sign. Indeed, the output of the CCI is not used. Note that its transfer function would be identical to Eq. (8) if

$$\sum_{i=0}^{N} G_i^+ - \sum_{j=0}^{M} G_j^- = G_F \tag{14}$$

where G_F is an arbitrary normalization conductance, not associated to any element, that plays the role of G_F in Eq. (9). Elements Z_0^+ and Z_0^- in Fig. 9(a) serve the purpose of achieving Eq. (14). As with opamp-based adders, only one of these elements is required.

The design equations required to obtain the different *Gi* values are identical to those obtained for the generalized adder in Fig. 6.

Figure 9(b) shows a generalized adder based on a CCII and its transfer function. The positive signed expression is obtained if a $CCII^+$ is employed, whereas the negative sign corresponds to the use of a $CCII^-$. In either case, the transfer function can again be expressed in the form of Eq. (8) if Z_0^+ and Z_0^- are chosen to verify

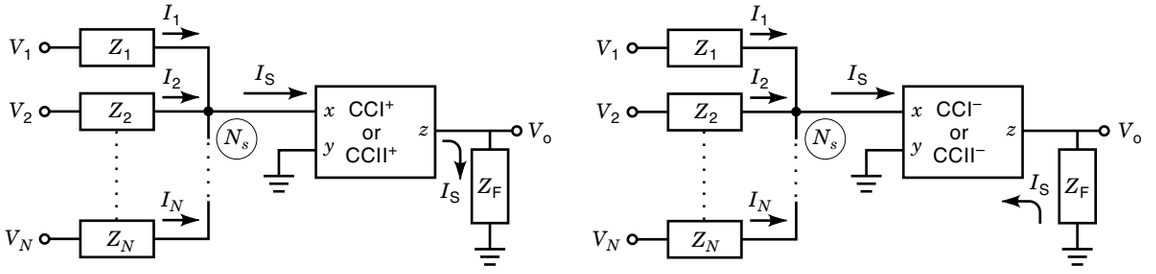
$$\sum_{i=0}^{N} G_i^+ = \sum_{j=0}^{M} G_j^- \tag{15}$$

As in the previous case, only one of both impedances are needed. If

$$\sum_{j=1}^{M} W_j^- < \sum_{i=1}^{N} W_i^+ \tag{16}$$

we can select $G_0^+ = 0$ and

$$G_0^- = G_F \left(\sum_{i=1}^{N} W_i^+ - \sum_{j=1}^{M} W_j^- \right) \tag{17}$$



$$V_o = \sum_{i=1}^{N} \frac{G_i}{G_F} V_i = \sum_{i=1}^{N} W_i V_i \qquad V_o = -\sum_{i=1}^{N} \frac{G_i}{G_F} V_i = -\sum_{i=1}^{N} W_i V_i$$

(**a**) (**b**)

Figure 7. Summing circuits using a single current conveyor: (a) noninverting configuration and (b) inverting configuration.

Otherwise, we can select $G_0^- = 0$ and

$$G_0^+ = G_F \left(\sum_{j=1}^{M} W_j^- - \sum_{i=1}^{N} W_i^+ \right) \tag{18}$$

An important remark concerning the presented summing circuits with CCs is that, because the output impedance of the current conveyor is high, any current drain from the output node V_o would result in deviations from the expected behavior. If a low impedance load is to be driven by V_o, a buffer will be required.

Summing Circuits Using Operational Transconductance Amplifiers

The use of operational transconductance amplifiers (OTA) as active elements constitutes by itself a general technique for the realization of analog circuits, with special relevance in integrated circuit (IC) realizations. Summing devices are easily realized with this circuit technique.

Figure 10(a) shows the symbol and transfer function of an OTA. Its differential input provides a large flexibility in the realization of most operators, including positive and negative resistors, as shown in Figs. 10(b, c). OTAs by themselves provide a direct transformation of voltages into currents, whereas the "resistor" configurations in Fig. 10 allow the inverse transformation. Therefore, we have all elements required to realize a summing structure. In addition, the differential input of the OTAs allow the realization of either-sign weights by simply swapping the input nodes.

Figure 11 shows a generalized adder structure, whose transfer function is given by

$$V_o = \sum_{i=1}^{N} W_i^+ V_i - \sum_{j=1}^{M} W_j^- V_j^- \tag{19}$$

where the weights are given by transconductances ratio

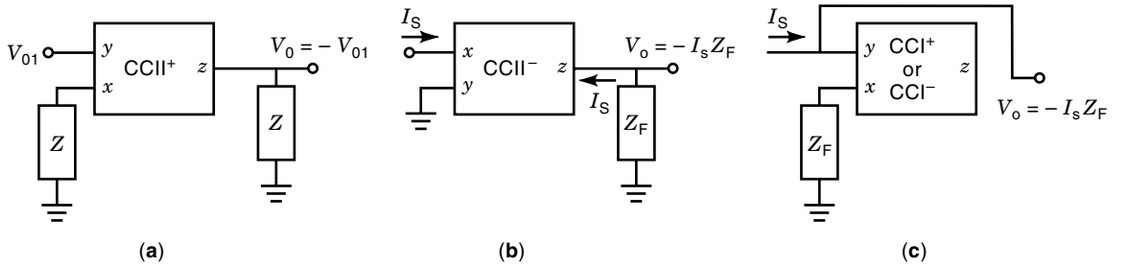
$$W_i^+ = \frac{g_{mi}^+}{g_{mF}} \qquad \text{and} \qquad W_j^- = \frac{g_{mj}^-}{g_{mF}} \tag{20}$$

which can be made highly insensitive to variations in the IC fabrication process. As with CC-based realizations, an output buffer will be required if low impedances are to be driven.

Discrete-Time Summing Circuits Using Switched-Capacitors Techniques

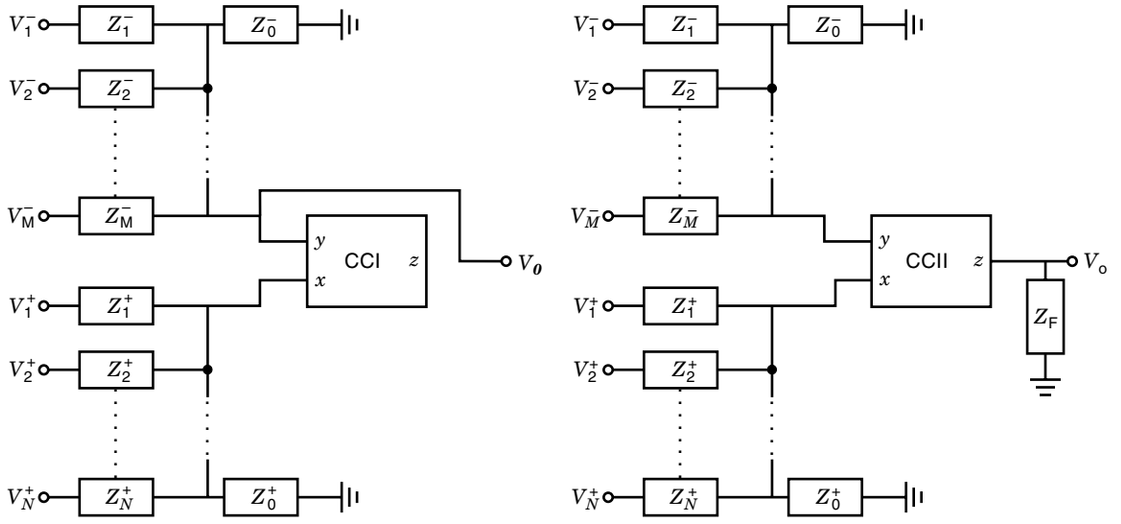
Switched-capacitor (SC) techniques substitute continuous-time current flow by periodic charge-package transferences. In their simplest and most common form, SC circuits are controlled by a pair of nonoverlapped clock signals defining two alternating configurations of the circuit. These two "phases" are often referred to as even and odd.

The operation of an SC summing circuit can be described in general as follows: given a set of voltage signals to be



(**a**) (**b**) (**c**)

Figure 8. Sign inverters using current conveyors: (a) inverter using a CCII+, (b) inverter using a CCII−, and (c) inverter using either a positive or a negative CCI.



$$V_o = \frac{\sum_{i=1}^{N} G_i^+ V_i^+ - \sum_{j=1}^{M} G_j^- V_j^-}{\sum_{i=0}^{N} G_i^+ - \sum_{j=0}^{M} G_j^-}$$

(a)

$$V_o = \pm \left(\sum_{i=1}^{N} \frac{G_i^+}{G_F} V_i^+ - \left(\frac{\sum_{i=0}^{N} G_i^+}{\sum_{j=0}^{M} G_j^-} \right) \sum_{j=1}^{M} \frac{G_j^-}{G_F} V_j^- \right)$$

(b)

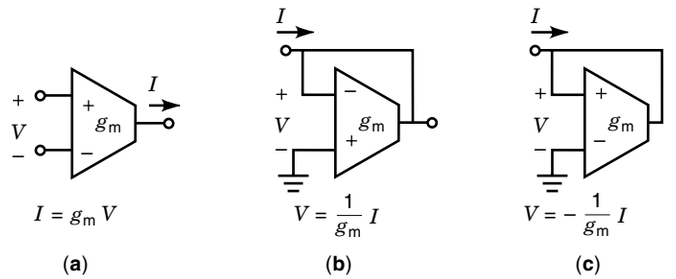
Figure 9. Generalized adders using current conveyors: (a) using CCI and (b) using CCII.

added, each of them is periodically sampled, during one of the clock phases, in a linear capacitor. The resulting charge packages are added in a common node and transformed into a voltage during the same or the next clock phase using another capacitor. Note that the underlying operation principle is identical to that of the previous continuous-time circuits, except that current flows are replaced by periodic discrete-time charge transferences, which indeed may be considered as a current flow from a time-averaged perspective.

Figure 12(a) shows a set of SC input branches. In the odd-to-even phase transition, a charge package ΔQ^{oe}, equal to the sum of the charge variations in each capacitor, flows to a virtual ground node. The value of ΔQ^{oe} is easily obtained applying the charge-conservation principle, yielding

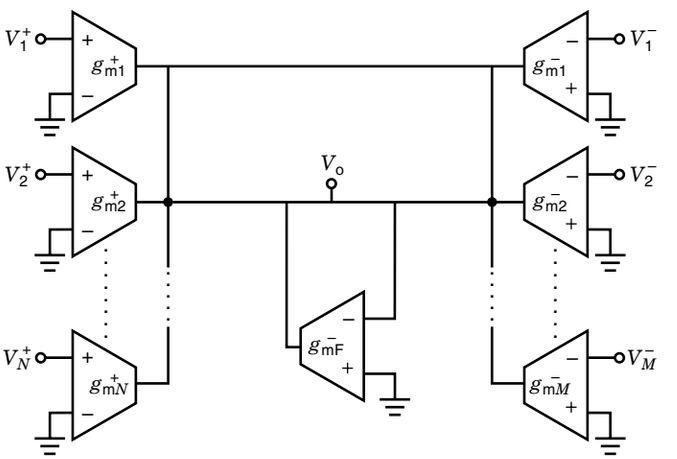
$$\Delta Q^{oe} = Q^e - Q^o = \sum_{i=1}^{N} C_i \left[V_{i2}^e(n) + V_{i3}^o(n - \tfrac{1}{2}) - V_{i1}^o(n - \tfrac{1}{2}) \right] \tag{21}$$

Note that signal information is conveyed in the difference of capacitor charges at the end of two consecutive odd and even



$I = g_m V$ (a) $V = \frac{1}{g_m} I$ (b) $V = -\frac{1}{g_m} I$ (c)

Figure 10. (a) Symbol and transfer function of an operational transconductance amplifier. (b) Implementation of a grounded resistor using an OTA with negative feedback. (c) Implementation of a grounded negative resistor using an OTA with positive feedback.



$$V_o = \sum_{i=1}^{N} \frac{g_{mi}^+}{g_{mF}} V_i^+ - \sum_{j=1}^{M} \frac{g_{mj}^-}{g_{mF}} V_j^-$$

Figure 11. Generalized adder circuit based on OTAs.

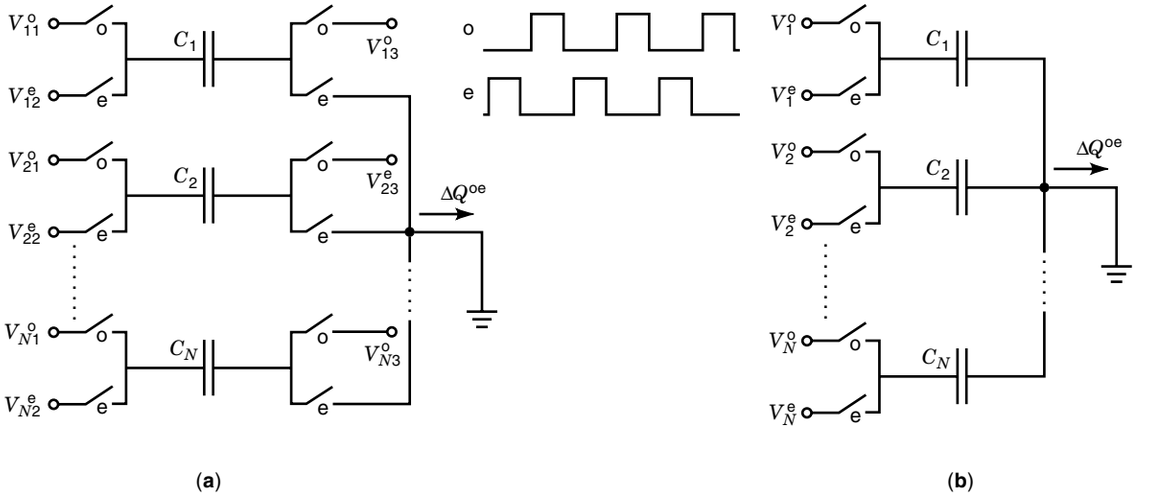


Figure 12. (a) Generic input branch of a SC summing circuit. (b) Input branch for SC structures insensitive to opamp offset-voltage.

phases, represented by time-instants $(n - \frac{1}{2})T$ and nT, respectively, where T is the clock-signal period. The discrete-time nature of the processing is therefore evident.

Each term in Eq. (21) is given by voltage differences at the two plates of the corresponding capacitor. Although the virtual ground node is nominally equivalent to ground, the input-referred offset voltage of the required active element introduces a small error. This error will be relevant if the required accuracy is in the range of 7–8 equivalent bits or above. Offset-voltage effects can be avoided if one of the plates of the capacitor is permanently connected to the virtual ground node, as shown in Fig. 12(b), which results in the following expression for ΔQ^{oe},

$$\Delta Q^{oe} = Q^{e} - Q^{o} = \sum_{i=1}^{N} C_i[V_i^{e}(n) - V_i^{o}(n - \tfrac{1}{2})] \qquad (22)$$

Results similar to Eqs. (21) and (22) can be obtained for the charge package ΔQ^{oe} originated just after the even-to-odd transition.

The transformation of these charge signals into a voltage requires a linear capacitor and an active element to implement the required virtual ground. An operational amplifier with capacitive feedback is the most common choice, as shown in Fig. 13(a). Note that one of the clock phases is used to discharge the feedback capacitor and to provide a current path to the virtual ground node. Therefore, only one of the charge packages originated by the input branches, ΔQ^{oe} in the example, is actually transformed into a voltage, and the output signal is valid only during one of the clock phases. Indeed, during the other clock phase, the output voltage is equal to the opamp offset voltage, resulting in large slewing requirements from the opamp. Figures 13(b, c) present alternative charge-sensing stages with lower slewing requirements at the expense of an increased complexity. Other relevant differences are related to their particular sensitivity to the finite gain and bandwidth of the opamps (1).

Combinations of the input branches in Fig. 12 and the sensing stages in Fig. 13 result in summing structures insensitive to parasitic capacitances. If Fig. 12(b) is employed, the result is also insensitive to the opamp offset, but only during one of the clock phases.

In every case, the output voltage can be obtained from

$$V_i^{e} = -\Delta Q^{oe}/C_o \qquad (23)$$

Note that a sign inversion takes place in the charge-sensing stage. The underlying voltage-charge-voltage transformations using capacitors result in the following expression for the ab-

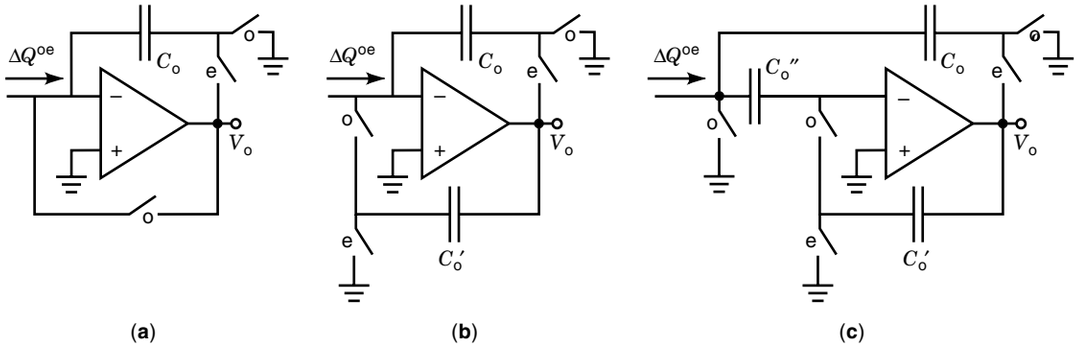


Figure 13. Charge-sensing stages insensitive to opamp offset-voltage and stray capacitances: (a) Gregorian stage (2), (b) Maloberti stage (3), and (c) Nagaraj stage (4).

solute value of the weighting coefficients:

$$W_{\mathrm{i}} = \frac{C_{\mathrm{i}}}{C_{\mathrm{o}}} \tag{24}$$

which can be made highly independent of IC fabrication technology variations. As seen from Eqs. (21) and (22), contributions of either sign are possible, depending on the particular switching configuration of the input branch. Note also that a delay of $T/2$ exists for some of the input signals. The delay is related to the sign of the weighting factor, in particular if the offset-insensitive branch is used.

The operational amplifiers in Figs. 12 and 13 can be replaced by operational transconductance amplifiers whenever the SC circuit drives a capacitive load. This is in fact a common practice in IC realizations, in which OTAs are usually advantageous in terms of area and power as compared to traditional, low output-impedance opamps.

Advanced Considerations in the Design of Analog Summing Circuits

The discussion of the preceding circuits has been made, as usual, on the basis of idealized descriptions of circuit elements: *models*. Practice, however, shows that real circuits operation is affected by several "second-order" effects. These include qualitative deviations of components behavior from their ideal model, systematic and random variations of electrical parameters from their nominal values, additional parasitic elements, and external interferences. Some of these error sources and their effects on analog summing circuits are described next.

Element Tolerances. Electrical parameters of real circuit components differ from their nominal values as a result of unavoidable manufacturing process inaccuracies. Such deviations are generally unpredictable and, therefore, commonly treated as statistical variables. In general, manufacturers provide a tolerance range for representative parameters of electronic components. Typical discrete-component tolerances are in the range of 1 to 20%.

Inaccuracies in component values result in deviations in circuit performances, which may be critical or not depending on specific sensitivities and acceptability margins.

Concerning analog summing circuits, element tolerances result in deviations from the desired weighting factors. In particular, we have seen that weighting factors are generally given by parameter ratios of same-type elements. Using the case of a resistor ratio $R_{\mathrm{o}}/R_{\mathrm{i}}$ as an example, the actual weight value can be expressed as

$$W_{\mathrm{i}} = \frac{R_{\mathrm{o}} + \Delta R_{\mathrm{o}}}{R_{\mathrm{i}} + \Delta R_{\mathrm{i}}} = \frac{R_{\mathrm{o}}}{R_{\mathrm{i}}} \left(\frac{1 + \dfrac{\Delta R_{\mathrm{o}}}{R_{\mathrm{o}}}}{1 + \dfrac{\Delta R_{\mathrm{i}}}{R_{\mathrm{i}}}} \right) \cong \frac{R_{\mathrm{o}}}{R_{\mathrm{i}}} \left(1 + \frac{\Delta R_{\mathrm{o}}}{R_{\mathrm{o}}} - \frac{\Delta R_{\mathrm{i}}}{R_{\mathrm{i}}} \right) \tag{25}$$

Note that relative weight deviations are given by the difference of the relative error of the two resistors. Therefore, extreme deviations may be as large as twice the tolerance of the components (assumed equal for simplicity), but they can also be extremely low. Assuming uncorrelated errors in the two resistors, it is easy to show that the standard deviation of the weight will be equal to $\sqrt{2}$ times the tolerance of the resistors. This is a reasonable assumption when discrete components are being employed. However, when integrated circuits are being formed, same-type components are fabricated simultaneously under extremely similar conditions; therefore, appreciable correlation exists. In this case, the two error terms tend to cancel each other, and it is not rare to obtain accuracies in the order of 0.1% with absolute value tolerances of about 20%.

Active Element Offset Voltage. Mismatch among ideally identical devices within active devices (opamps, current conveyors, OTAs) produce deviations in their performance as well. One of the most representative is the so-called input-referred offset voltage. Its effect can be analyzed using a random-value dc (direct current) voltage source at one of the input terminals. In analog summing structures, it produces at the output an additional term that is independent of the input signals (output offset). As an example, the analysis of the general adder in Fig. 6 results in the following expression for its output voltage:

$$V_{\mathrm{o}} = \sum_{i=1}^{N} W_i^+ V_i - \sum_{j=1}^{M} W_j^- V_j^- - \left(1 + \sum_{j=1}^{M} W_j^- \right) E_{\mathrm{os}} \tag{26}$$

where E_{os} is the opamp offset voltage.

Active Elements Dynamic Response. The dynamic response of active elements is reflected in the dynamic behavior of analog summing circuits. The corresponding analysis requires a dynamic model of the specific active device. We will consider again the opamp-based generalized adder in Fig. 6 as an example. Using a dominant-pole linear model for the opamp, we obtain the following expression:

$$V_{\mathrm{o}}(s) = \frac{1}{\left(1 + \dfrac{s}{s_{\mathrm{p}}} \right)} \left(\sum_{i=1}^{N} W_i^+ V_i(s) - \sum_{j=1}^{M} W_j^- V_j^-(s) \right) \tag{27}$$

where s_{p} is the dominant pole of the summing circuit given by

$$s_{\mathrm{p}} = \frac{-\mathrm{GB}}{\left(1 + \dfrac{\sum_{j=0}^{M} W_j^-}{} \right)} \tag{28}$$

and GB is the gain-bandwidth product of the opamp (see OPERATIONAL AMPLIFIERS).

The application of a step voltage at one of the input signals will thus result in an exponential response characterized by a time constant

$$\tau = \frac{-1}{s_{\mathrm{p}}} = \frac{1}{\mathrm{GB}} \left(1 + \sum_{j=0}^{M} W_j^- \right) = \frac{1}{\mathrm{GB}} \sum_{i=0}^{N} W_i^+ \tag{29}$$

where we have used the design Eq. (7). Note that system response-time increases with the sum of either positive or negative weights. This is a general statement valid for analog summing circuits based on other active devices.

An important nonlinear limitation of operational amplifiers, known as slew rate, establishes an upper bound for the

slope of the output voltage waveform. In cases where the linear model predicts faster variations, the opamp will respond with a constant slope independent of the input signal.

Finite Opamp Open-Loop Gain. Opamps finite low-frequency gain A_0 produces a uniform attenuation of summing-circuit weight values. As an example, analysis of the generalized adder in Fig. 6 yields the following result:

$$V_o = \frac{\sum_{i=1}^{N} W_i^+ V_i - \sum_{j=1}^{M} W_j^- V_j^-}{1 + \frac{1}{A_0}\left(1 + \sum_{j=0}^{M} W_j^-\right)} \tag{30}$$

Finite Output Impedance of CCs and OTAs. When current conveyors or operational transconductance amplifiers are used, their finite output impedance produces a result similar to that of the finite open loop gain of operational amplifiers. An analysis of the generalized OTA-based adder in Fig. 11 results in

$$V_o = \frac{\sum_{i=1}^{N} W_i^+ V_i - \sum_{j=1}^{M} W_j^- V_j^-}{1 + \alpha\left(1 + \sum_{i=1}^{N} W_i^+ + \sum_{j=1}^{M} W_j^-\right)} \tag{31}$$

where α is the ratio of OTAs output-conductance to transconductance, which is assumed equal for every OTA for simplicity. This is indeed a correct assumption on most practical cases. Similarly, the analysis of the generalized CC-based adder in Fig. 9(b) yields

$$V_o = \frac{\sum_{i=1}^{N} W_i^+ V_i^+ - \sum_{j=1}^{M} W_j^- V_j^-}{1 + G_p/G_F} \tag{32}$$

where G_p is the output conductance of the CCs.

Parasitic Devices. Any real circuit includes, in addition to the devices employed to implement the desired function, many other unwanted "devices" such as wiring and contact resistances, self and mutual inductances among wires, capacitive couplings, and transmission-lines. These parasitic elements may become relevant in certain circumstances, like high-frequency operation; in integrated circuit design; and, in general, whenever their electrical parameters are in the range of nominal devices. In these cases, special circuit techniques and careful routing should be considered.

Feedthrough. In switched-capacitor circuits, the MOS transistors employed as analog switches produce charge-injection effects on the capacitors employed for charge storage. These effects can be attenuated using small switches and large capacitors.

Other Error Sources. Nonlinearity of passive and active elements produce distortion on the output signal. A clear example is the output voltage- and current-saturation of active devices. Input (and output) impedance of active devices results in uniform weight deviations. Opamp input bias-currents produce output offset, and finite power-supply rejection-ratio (PSRR) and common-mode rejection-ratio (CMRR) result in spurious components at the output signal due to power-supply and common-mode signals coupling. High-order parasitic dynamic effects of active devices could produce stability problems. Finally, electronic noise from active devices and resistors may be relevant in certain applications cases.

DIGITAL ADDERS

Introduction

Previous sections focused on analog summing circuits. Digital adders operate with an essentially different codification of signals and are, therefore, different from their analog counterparts in practically every aspect.

Digital addition follows strict algorithmic rules on abstract numbers represented by a sequence of digits. In many senses, the process is identical to "human" summation of magnitudes represented by decimal (base 10) numbers. One major difference is the numerical *base,* which in the vast majority of digital systems is base 2. Numbers are therefore represented by strings of bits (i.e., digits), whose possible values are either 0 or 1. The most common form of representation of unsigned numbers is binary magnitudes, which follows the same conceptual rules as decimal representation. An additional difference is related to the representation of signed numbers. For the purpose of arithmetic operations, the so-called two's complement notation is the most widely used. Finally, an important constraint on digital summing circuits is their fixed word-length, imposed by hardware realizations, which may result in truncation or round-off errors depending on the representation employed: fixed-point or floating-point (see DIGITAL ARITHMETIC). In what follows, fixed-point arithmetic is assumed.

Binary-Magnitudes Arithmetic

The addition of two (positive) binary magnitudes A and B can be performed following the same conceptual rules employed in "hand" addition of decimal numbers. Fig. 14(a) illustrates the procedure, beginning from the right-hand side column, which represents the least significative bit (LSB) and proceeding toward the most significative bit (MSB) on the left-hand side.

The sum of any pair of bits a_i and b_i, each with values 0 or 1, ranges from 0 to 2 and must be represented by a two bits number commonly denoted as $(C_{i+1}\, S_i)$, as shown in Fig. 14(b). The sum-bit S_i is already part of the result $S = A + B$, whereas the carry-bit C_{i+1} must be added to the next column. Therefore, three bits must be added at each column i: a_i, b_i and the carry-bit from the previous column C_i. The sum of three bits ranges from 0 to 3 and can still be represented with a two-bits number $(C_{i+1}\, S_i)$, as shown in Fig. 14(c). Thus, the process can start from the LSB column, for which $C_0 = 0$ is assumed, and proceed toward the MSB in a repetitive manner. This procedure is the underlying fundamental of binary digital adders.

Basic Circuit Blocks. A digital circuit block realizing the truth table in Fig. 14(b) is commonly known as a half-adder

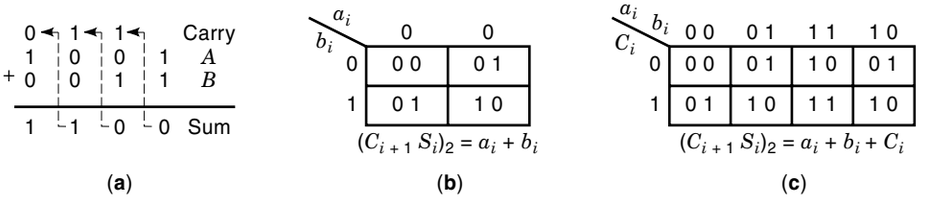


Figure 14. Examples of (a) binary addition process, (b) arithmetic addition of two bits, and (c) arithmetic addition of three bits.

(HA), whereas that defined by Fig. 14(c) is referred to as a full-adder (FA). Fig. 15(a, b) contain representations for these two basic building blocks of digital adders. Their implementation can be carried out following any general procedure for Boolean functions realization.

The functionality of a HA can be expressed as two Boolean functions:

$$S_i^{\mathrm{HA}} = \overline{a}_i \cdot b_i + a_i \cdot \overline{b}_i = a_i \oplus b_i \qquad C_{i+1}^{\mathrm{HA}} = a_i \cdot b_i \tag{33}$$

from which an implementation using two digital gates, an XOR and an AND, is straightforward, as shown in Fig. 16. Because the input-to-output signal path goes through just one gate, the propagation delay of this realization corresponds to one "gate level."

An FA can be implemented using two HAs and one OR gate, as shown in Fig. 17(a). This realization requires a reduced number of gates but, on the other hand, exhibits a delay of three gate levels. Other implementations can be built on the basis of the Boolean expressions for the two outputs of a FA,

$$\begin{aligned} S_i^{\mathrm{FA}} &= \overline{a}_i \cdot \overline{b}_i \cdot C_i + \overline{a}_i \cdot b_i \cdot \overline{C}_i + a_i \cdot \overline{b}_i \cdot \overline{C}_i + a_i \cdot b_i \cdot C_i \\ &= a_i \oplus b_i \oplus C_i \end{aligned} \tag{34}$$

$$C_{i+1}^{\mathrm{FA}} = a_i \cdot b_i + b_i \cdot C_i + a_i \cdot C_i \tag{35}$$

Figure 17(b) shows an implementation using double-rail input signals and two levels of NAND gates, and Fig. 17(c) shows an implementation using single-rail signals using just two-input NAND gates, with a propagation delay of six gate levels. Many other alternatives exist.

A careful evaluation of these alternatives in terms of cost and speed should take into account the diverse complexity (transistor count) and propagation delay of different gates. In general, NAND, NOR, and INV gates are faster and simpler, whereas AND, OR, and especially XOR and NXOR gates are more complex and slower. The requirement of double-rail inputs signals may result in additional cost and delay, depending on the specific case.

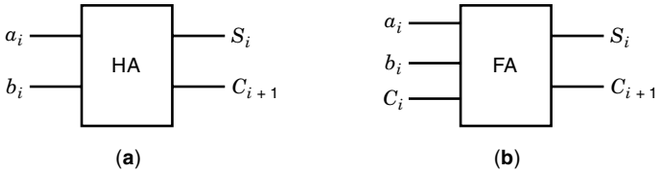


Figure 15. Basic digital-adder modules representation: (a) half adder and (b) full adder.

Serial and Parallel Adders. The addition of two n-bit binary numbers can be carried out serially or in parallel. A *serial adder,* shown in Fig. 18, contains one single FA and a flip-flop and is controlled by a clock signal. The two words A and B are sequentially added on a bit-to-bit basis, beginning with the LSB, for which the flip-flop must be initially set to zero. Consecutive clock cycles produce consecutive bits of the resulting sum S, as well as a carry-bit, which is stored in the flip-flop and employed as input to the FA in the next clock cycle. The summation of the two words requires n clock cycles. Therefore, serial adders constitute a slow solution in general. On the other hand, they are highly efficient in terms of hardware.

Parallel adders can be regarded as the opposite alternative, realizing fast additions at the expense of an increased hardware cost. Figure 19 shows an intuitive realization of a parallel adder, commonly known as serial-carry and also as ripple adder. Its hardware cost is of n full adders. The carry-bits are transmitted in a chain from the FA corresponding to the LSB toward the MSB. This signal path determines the response time of the parallel adder, which can be described as

$$t_{\mathrm{sp}} = nt_{\mathrm{FA}} \tag{36}$$

where t_{FA} is the response time of one FA. The response times of both the serial and the parallel adders are proportional to the number of bits. However, because the period of the clock signal employed in the serial adder must be at least several times larger than t_{FA}, it is clear that parallel adders are faster. Still, parallel adders may be too slow for certain applications, especially for long digital words (large n). Certain advanced architectures overcome these problems at the expense of additional hardware (see "High-Performance Digital Adders").

Addition and Substraction of Signed Numbers

Addition-Substraction Equivalence. The addition and the substraction of two signed numbers can both be formulated on the basis of a summing operation, by simply changing the sign of one of the operands when needed. Fig. 20 shows a conceptual flow diagram of an adder/substracter circuit based

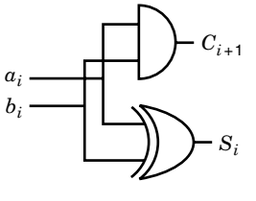


Figure 16. Half adder implementation.

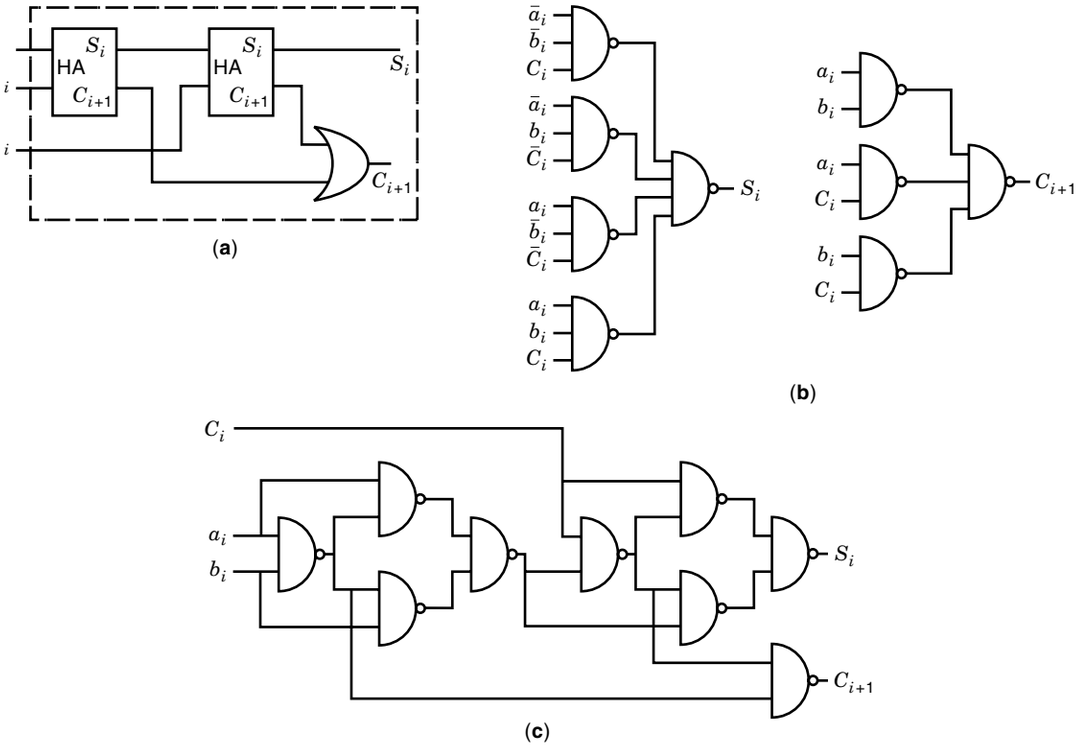


Figure 17. Alternative implementations of a full adder: (a) with two HAs and one OR gate, (b) with double-rail inputs and two levels of NAND gates, (c) with two-input NAND gates and six levels of delay.

on this approach. Signal $\overline{a}/s$ controls whether the sign of operand B is changed or not before the summation is performed. The specific meaning of a sign-inversion operation depends on the representation being used for the signed numbers, as described in Fig. 20.

In a sign-magnitude (SM) representation, a sign inversion is achieved by complementing just the sign bit, whereas the rest of the bits remain unchanged. In one's complement (C1) arithmetic, a sign inversion is obtained complementing every bit in the word. Finally, in two's complement (C2) arithmetic, a sign inversion requires adding one unit to the word obtained by complementing every bit.

Signed-Numbers Addition in Two's Complement Arithmetic. It can be shown that the addition of signed numbers in two's complement representation coincides with the (positive) binary magnitude summation of their digital codes. Fig. 21 shows examples covering the four possible cases of operand signs. The final (MSB) carry-bit C_n is neglected for the purpose of evaluating the result. It may be used, however, together with C_{n-1} to detect overflow. The possibility of using simple binary adders, like those described previously, for the summation of signed number has turned two's complement arithmetic the most widely used in digital operators, and the only one considered in what follows.

Two's Complement Adder-Subtracter Circuit. Figure 22(a) describes a digital adder-subtracter, controlled by signal $\overline{a}/s$, following the diagram illustrated in Fig. 20. It is based on a parallel binary-magnitude adder, and a parallel transfer/complement ($B/\overline{B}$) block. The realization and functionality of

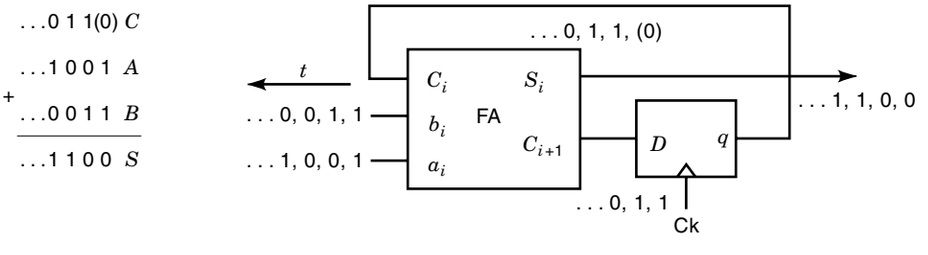


Figure 18. Serial adder and summation example. t indicates increasing time sequence.

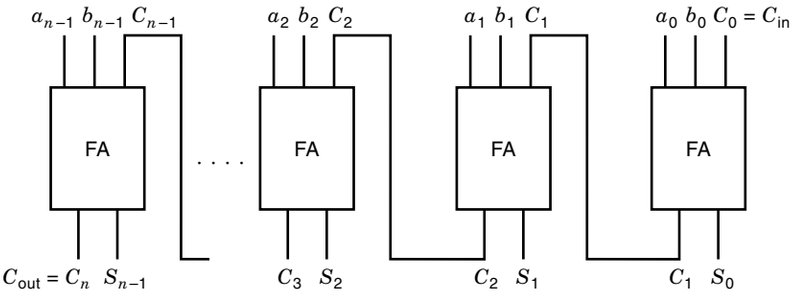


Figure 19. Parallel adder with serial carry (ripple adder).

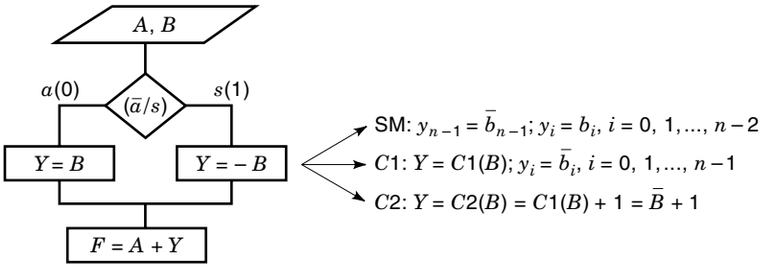


Figure 20. Flow diagram of a signed-number adder/subtracter circuit.

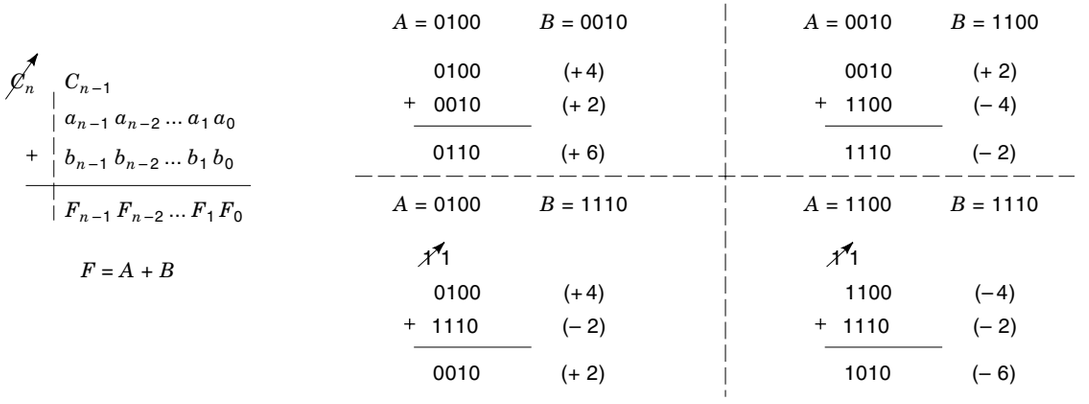


Figure 21. Four possible cases of arithmetic addition of signed numbers addition in two's complement representation. The result is correct in every case after neglecting the carry bit.

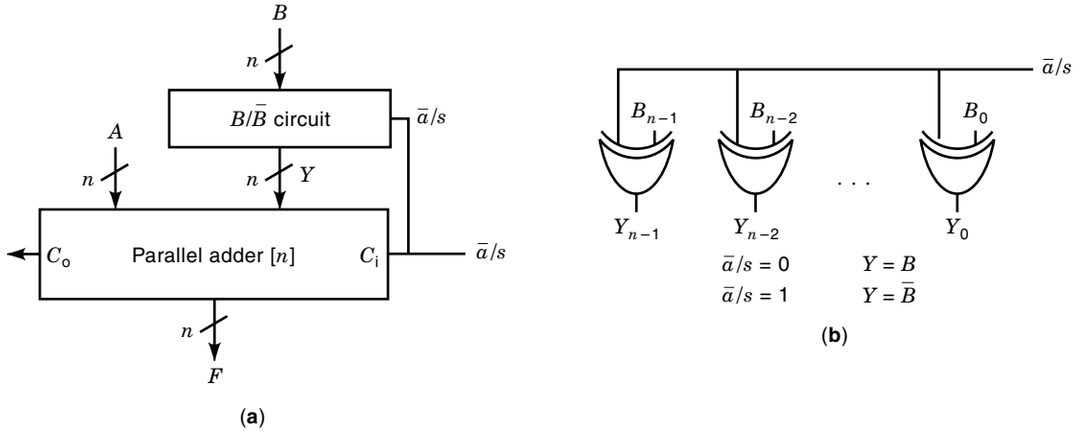


Figure 22. (a) Adder/subtracter circuit for signed numbers in two's complement notation and (b) transfer/complement ($B/\bar{B}$) circuit.

this last circuit block is described in Fig. 22(b). If $\overline{a}/s = 0$, the $B/\overline{B}$ circuit simply transfers its input B to its output, and the parallel adder, with the initial (LSB) carry-bit C_i set to 0, performs the addition of the two operands $A + B$. The result is correct for signed two's complement numbers and also for binary magnitudes. On the other hand, if $\overline{a}/s = 1$, the $B/\overline{B}$ circuit complements every bit b_i of operand B and transmits the result to the full adder, which now sees a 1 at C_i. The result is that A is added with $\overline{B} + 1$, that is, $-B$ in two's complement representation; therefore, the full adder produces the difference $A - B$ as expected.

Overflow Problems

An n-bits binary magnitude may take values ranging from 0 to $2^n - 1$, and the sum of two such magnitudes from 0 to $2^{n+1} - 2$. Similarly, the possible values of an n-bits signed-number in two's complement representation range from -2^{n-1} to $2^{n-1} - 1$, and the sum or difference of two such numbers from -2^n to $2^n - 2$. This means that the n-bits output of the adder-subtracter circuit in Fig. 22 will be unable to show the correct result in certain cases. This phenomenon is commonly known as overflow and is obviously something that should be detected. Furthermore, it would be convenient to determine the correct result also in these cases.

When unsigned binary magnitudes are being added, the MSB carry-bit C_n provides a flag-signal for overflow occurrences. It can also be shown that function

$$V = C_n \oplus C_{n-1} \tag{37}$$

which can be obtained at the expense of an additional XOR gate, constitutes a valid overflow flag for the addition and the subtraction of signed numbers in two's complement representation. Both signals are active when high.

In the event of overflow, and regardless the operation being performed, the correct result can be obtained, in its proper representation (binary magnitude or two's complement), from the $n + 1$ bits number compound by C_n (the new MSB or the new sign bit) and the n-bits output-word of the circuit in Fig. 22.

High-Performance Digital Adders

As in most digital processing circuits, the main concern in the optimization of digital adders is to increase their operation speed. Response time reductions can be achieved through improvements in the basic digital adder circuit block (the FA), through algorithmic or architectural modifications, or a combination of both. Parallel ripple adders are often used as a reference for the evaluation of advanced solutions, which in general focus on the elimination or at least the attenuation of the constraint imposed by the long signal path of carry signals from the least to the most significative bit.

Most modifications to the conventional implementations of the FA circuit block involve a reduction of the capacitive load of the carry signal [e.g., the so-called mirror, dynamic, and Manchester-adders (5)]. FA blocks with modified I/O signals are employed in the *carry-completion* adder architecture (6), yielding a reduced "average" response time.

Most architectural modifications rely on a segmentation of the bit-chain in smaller groups. One alternative, known as the *carry-bypass* adder (5), is based on evaluating within each group whether an eventual carry input C_i would propagate through the group. If it does, C_i is directly transferred to the carry output C_o. Otherwise, C_i can be ignored, and C_o is computed from the group input bits. A similar strategy results in the so-called *carry-skip* adder (6). An alternative architecture, the *linear carry-select* adder (5), computes within each group two results corresponding to the two possible values of C_i, and selects one of them after receiving its actual C_i value. A modification of this last architecture, the *square root carry-select* adder (5), employs increasing-length groups and results in a propagation type proportional to the square root of the number of bits.

Carry look-ahead adders (6,7) employ a significative different approach. Their architecture allows all carry-bits to be obtained simultaneously at the expense of a rapidly increasing complexity with the number of bits. In practice, this limits the number of bits to about four, forcing the use of combined approaches for larger word lengths.

Finally, a *pipeline* adder architecture (5) results in a high summing throughput, although the propagation time of individual summations may be larger than with the standard ripple adder.

Other alternatives relying on innovative circuit techniques (e.g., threshold gates, multivaluated-logic) or technologies (e.g., optical processors) do exist, but they will not be treated here.

BIBLIOGRAPHY

1. R. Unbehauen and A. Cichocki, *MOS Switched-Capacitor and Continuous-Time Integrated Circuits and Systems,* New York: Springer-Verlag, 1989.
2. R. Gregorian and G. C. Teme, *Analog MOS Integrated Circuits for Signal Processing,* New York: Wiley-Interscience, 1986.
3. F. Maloberti, Switched-capacitor building blocks for analogue signal processing, *Electronics Lett.,* **19**: 263–265. 1983.
4. K. Nagaraj et al., Reduction of finite-gain effect in switched-capacitor circuits, *Electron. Lett.,* **21**: 644–645, 1985.
5. Jan M. Rabaey, *Digital Integrated Circuit: A Design Perspective,* Upper Saddle River, NJ: Prentice-Hall, 1996.
6. Amos R. Omondi, *Computer Arithmetic Systems. Algorithms, Architectures and Implementations,* Upper Saddle River, NJ: Prentice-Hall, 1994.
7. N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design. A Systems Perspective,* Reading, MA: Addison-Wesley, 1985.

RAFAEL DOMÍNGUEZ-CASTRO
SERVANDO ESPEJO
ANGEL RODRÍGUEZ-VÁZQUEZ
CARMEN BAENA
MANUEL VALENCIA
University of Seville, IMSE-CNM-CSIC

SUNPHOTOMETERS. See PHOTOMETERS.

SUPERCAPACITORS. See CAPACITOR STORAGE.

SUPERCONDUCTING ANALOG AND DIGITAL MICROWAVE COMPONENTS. See SUPERCONDUCTING MICROWAVE TECHNOLOGY.