A New Description of MOS Circuits at Switch-Level With Applications

Massoud PEDRAM

Department of Electrical Engineering-Systems, University of Southern California,

Los Angeles, CA 90007-2562, USA

Xunwei WU*

Department of Electronic Engineering, Hangzhou University,

Hangzhou, Zhejiang 310028, China

(* Xunwei Wu is presently on leave at University of Southern California.)

SUMMARY After analyzing the limitations of the traditional description of CMOS circuits at the gate level, this paper introduces the notions of switching and signal variables for describing the switching states of MOS transistors and signals in CMOS circuits, respectively. Two connection operations for describing the interaction between MOS transistors and signals and a new description for MOS circuits at the switch level are presented. This new description can be used to express the functional relationship between inputs and the output at the switch level. It can also be used to describe the circuit structure composed of MOS switches. The new description can be effectively used to design both CMOS circuits and nMOS pass transistor circuits.

Key words: *MOS, LSI, pass-transistor logic, switching theory, low power* ***design***

1. Traditional Description of CMOS Circuits at Gate Level

The traditional description of CMOS circuits is based on Boolean algebra. Its elementary points are:

(1) Boolean variables are used to represent signals in circuits. The two values of a variable, 1 and 0 are physically represented by two levels of a signal, for example 5V and 0V.

(2) The basic operations among variables in Boolean algebra are NOT, AND and OR operations. Usually, two composite operations, NAND and NOR, are also introduced. These operations are realized by the corresponding basic circuit units called gates, such as NOT gate (inverter), AND gate, OR gate, NAND gate and NOR gate.

(3) NOT, AND and OR operations form a complete set and can be used to express any functions. Besides, NAND alone (or NOR alone) can form a complete basis by itself.

Therefore, as long as we get the function expression we can obtain its corresponding circuit configuration by using gates from the complete set. For example, the Exclusive-OR function is given by its truth table shown in Fig.1 (a). Based on the Boolean algebra, we can describe the Exclusive-OR function of *x* and *y* by the following expression:

$$x \oplus y = (x \,\mathrm{I}\;\bar{y})\,\mathrm{Y}(\bar{x}\,\mathrm{I}\;y) = \overline{\overline{(x\,\mathrm{I}\;\bar{y})}\,\mathrm{I}\;\overline{(\bar{x}\,\mathrm{I}\;y)}},$$

Which yields the corresponding circuits in Fig. 1(b) and (c), respectively. This example explains how the description is used for both the function and the circuit structure at the gate level.

Figure 1

Fig.1 Definition and circuits for Exclusive-OR operation (a) truth table, (b) circuit composed of NOT, AND and OR gates, (c) circuit composed of NOT and NAND gates.

We should also point out the following limitations of the traditional description.

(1) The internal structure of a gate cannot be described, or derived from the function expression. For example, Boolean algebra cannot describe the internal structures of CMOS inverter and CMOS NAND gate shown in Fig. 2 (a) and (b).

(2) A compound gate, which also realizes the Exclusive-OR function, is shown in Fig. 2 (c). Boolean algebra cannot describe the structure because it is not composed of gates. Reference [1] introduces the following procedure to derive the structure from its Boolean expression. First, we obtain the inverted expression: $x \oplus y = \overline{(x \mathrm{I}\ y)\mathrm{Y}(\bar{x} \mathrm{I}\ \bar{y})}$. For the n-side of the CMOS structure, we take the non-inverted expression $(x \mathrm{I}\ y)\mathrm{Y}(\bar{x} \mathrm{I}\ \bar{y})$. Here the operations I and Y may be considered connections of nMOS transistors in series and in parallel. After having the n-structure, a dual p-structure can be derived and the whole configuration is obtained as shown in Fig. 2 (c). Obviously, the procedure is not included in Boolean algebra.

(3) NOT, AND and OR operations in Boolean algebra form a complete set, but they cannot be used to describe the relationship of the output to the inputs of a CMOS circuit that has a high-impedance state ϕ. The output of a simple CMOS transmission gate shown in Fig. 2 (d) is expressed as $f = (c \mathrm{I}\ x)\mathrm{Y}(\Phi \mathrm{I}\ \bar{x})$ in some textbooks. However, we do not define the operations related to the high-impedance state F in Boolean algebra.

Figure 2

Fig .2 Switch structure of a few CMOS gates (a) inverter, (b) NAND gate, (c) composed Exclusive-OR gate, (d) transmission gate.

According to the above discussion, Boolean algebra can be used to describe the CMOS circuit structure at the gate level, but cannot be used to describe the switching states of MOS transistors in the circuit and the circuit structure at switch level. The description at switch level is however desired for the switch-level techniques [2,3].

In the recent past, there has been renewed interest in pass-transistor logic circuits because they comprise of fewer transistors and exhibit smaller stray capacitance, compared conventional static CMOS circuits [4,5], and hence lead to lower power dissipation. A number of papers on pass-transistor logic have been published, for example, CPL (Complementary Pass-transistor Logic), DCVSPG (Differential Cascade Voltage Switch with Pass-Gate), DPL (Double pass-transistor Logic), SRPL (Swing Restored Pass-transistor Logic), and SAPL (Sense-Amplifying Pass-transistor Logic) [6-10]. However, pass transistor logic has not succeeded in playing a major role in practical LSI designs. In particular, pass transistors are used only in a small portion of arithmetic macros or XOR logic. One reason is that no synthesis tools are available for pass-transistor based design. The other reason is that the nMOS pass-transistor circuits are generally perceived to have poor low-voltage performance compared to their full static CMOS counterparts which raises doubts about their future applicability.

Reference [11] investigates various issues relative to the design of nMOS pass-transistor circuits. A cell library and an automatic synthesis tool utilizing the functionality of pass-transistors are constructed. The preliminary results appear to be quite positive. By designing some small arithmetic circuits, the others demonstrate excellent quality of pass-transistor logic in terms of area, delay time, and power dissipation. In addition, the authors find that the low-voltage performance of nMOS-based pass-transistor circuits is better than the conventional CMOS circuits down to 1V, when the threshold voltage is 0.4V. However, a number of key issues related to direct synthesis of pass-transistor circuits from a Boolean description remain.

Up until now, the pass-transistor design and the traditional fully complementary logic design have been considered to be very different design methods [12]. In this paper we will show that these methods are based on the same switch-level view of the MOS transistor. Therefore, by focusing on the switching process, we derive a relationship between the two methods, which then enables us to solve the problem of synthesizing pass-transistor circuits. The remainder of this paper is organized as follows. Section 2 gives a new description that describes both signal and switching state of transistors in MOS circuits and reflects the circuit structure, where each MOS transistor is recognized as a switch for transmitting its source signal to its drain. Sections 3 and 4 present application of the new description to the design of CMOS and nMOS pass-transistor circuits. Section 5 contains our concluding remarks.

2. Description of MOS Circuits at Switch Level

To describe the MOS circuit structure, we introduce an additional variable that describes switches in circuits; we should distinguish the new variable from the variable that is used to describe signals.

(1) Assume α, β … are switching variables that take two values, T and F, which in turn represent the two opposite states of on and off for a MOS transistor. The basic operations related to switching variables are NOT, AND and OR. Their definitions are as follows:

NOT operation

$$\tilde{\alpha} = \begin{cases} \mathrm{T} & \text{if} \quad \alpha = \mathrm{F}, \\ \mathrm{F} & \text{if} \quad \alpha = \mathrm{T}; \end{cases} \tag{1}$$

AND operation

$$\alpha \cdot \beta = \begin{cases} \mathrm{T} & \text{if} \quad \alpha = \beta = \mathrm{T} \\ \mathrm{F} & \text{otherwise} \end{cases} \tag{2}$$

OR operation

$$\alpha + \beta = \begin{cases} \mathrm{F} & \text{if} \quad \alpha = \beta = \mathrm{F} \\ \mathrm{T} & \text{otherwise} \end{cases} \tag{3}$$

Based on the above basic operations, a binary *switching algebra* is established.

(2) Assume *x*, *y*, … are binary signal variables. They take two values, 1 and 0, which represent the two signal levels, high and low, in a circuit. They have a precise magnitude and can be identified by comparing their magnitude with a threshold value, denoted by 0.5. The basic operations that are related to binary signal variables are Complement, Minimum and Maximum. Their definitions are as follows:

Complement operation

$$\bar{x} = \begin{cases} 1 & \text{if} \quad x = 0 \\ 0 & \text{if} \quad x = 1 \end{cases} \tag{4}$$

Minimum operation

$$x \cap y = \begin{cases} 1 & \text{if} \quad x = y = 1 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Maximum operation

$$x \cup y = \begin{cases} 0 & \text{if} \quad x = y = 0 \\ 1 & \text{otherwise} \end{cases} \tag{6}$$

Based on the above basic operations, binary *signal algebra* is formed.

Therefore, we have two kinds of binary algebra systems; it can be shown that the two systems are isomorphic. However, in the existing literature they are always substituted for one another confused without considering their essential differences and their isomorphism. Therefore, gates that realize Complement, Minimum, and Maximum operations are traditionally named NOT, AND and OR gates.

Taking the inverter in Fig. 2(a) as an example to explain two kinds of variables, we can use x, $\bar{x}$, α_p and α_n to express the input and output signal, and the switching states of pMOS and nMOS transistors, respectively. Their relationship is given in Table 1.

Table 1 Relationship between signals and switching states in a CMOS inverter.

Table 1

We can further introduce operations between two kinds of variables for describing the connection between the on-off states of switching elements and the voltage levels of the signals, as shown in Fig. 3. They are:

Connection operation I – which describes the physical process of how the binary signal controls the on-off state of an element.

Connection operation II – which describes the physical process of how the on-off state of an element controls the transmission of the binary signal.

Figure 3

Fig. 3 Connections between binary switching variables and binary signal variables.

In a CMOS digital circuit, the on-off state of a MOS transistor is dependent on the comparison between the gate signal and the threshold. Therefore, we define the connection operation I as follow:

High-threshold comparison operation

$$^{0.5}x = \begin{cases} \mathrm{T} & \text{if} \quad x > 0.5(i.e., x = 1) \\ \mathrm{F} & \text{if} \quad x < 0.5(i.e., x = 0) \end{cases}$$

(7)

Low-threshold comparison operation

$$x^{0.5} = \begin{cases} \mathrm{T} & \text{if} \quad x < 0.5 \ \text{(i.e., x = 0)} \\ \mathrm{F} & \text{if} \quad x > 0.5 \ \text{(i.e., x =1)} \end{cases}$$

(8)

In Eqs. (7) and (8), the gate signal $x \in \{0,\ 1\}$ and 0.5 indicates that the detection threshold is set in the middle of two logic levels, 0 and 1. If *x* assumes an intermediate value other than 0 or 1, then the switching state of the MOS transistor will be indeterminate. This case is **not allowed**. Note that these two equations represent the high-active switching characteristic of a nMOS transistor and the low-active switching characteristic of a pMOS transistor, respectively.

The following properties can be easily verified by use of the above definitions:

$$\bar{x}^{0.5} = {}^{0.5}x = \widetilde{x^{0.5}}\ , \tag{9}$$

$${}^{0.5}\bar{x} = x^{0.5} = \widetilde{{}^{0.5}x}\ ,$$

(10)

Which state that the two threshold comparison operations can be transformed by complementing the signal variable.

$${}^{0.5}(x \mathrm{I}\ y) = {}^{0.5}x \cdot {}^{0.5}y\ , \tag{11}$$

$${}^{0.5}(x \mathrm{Y} y) = {}^{0.5}x + {}^{0.5}y\ . \tag{12}$$

In Eqs. (10)-(12) the corresponding relationships between Complement, Minimum and Maximum in binary signal algebra and NOT, AND and OR in switching algebra are established by use of the high-threshold comparison operations. Furthermore, the high-threshold comparison operation penetrates through a function $f(x, y, \Lambda\ ; -, \mathrm{I}\ , \mathrm{Y})$ in binary signal algebra as follows:

$${}^{0.5}f\,(x, y, \Lambda\ ; -, \mathrm{I}\ , \mathrm{Y}) = f\,({}^{0.5}x, {}^{0.5}y, \Lambda\ ; \sim,\ \cdot\ , +)\ . \tag{13}$$

On the other hand, the on-off state of a MOS transistor determines whether the source signal is transmitted to the drain or not. Therefore, we may introduce the connection operation II as follows.

Transmission operation

$$c * \alpha = \begin{cases} \mathrm{c} & \text{if} \quad \alpha = \mathrm{T} \\ \Phi & \text{if} \quad \alpha = \mathrm{F} \end{cases} \tag{14}$$

Where the binary variable *c* is called transmitted source signal, and *a* represents the switching state of a transmission switch network. If α = T, signal *c* is transmitted to the output; if α = F, the switch network is off and its output is in the high-impedance state, denoted by symbol ϕ. The switch network is composed of a nMOS transistor, if *c* = 0, or a pMOS transistor, if *c* = 1.

To denote the joining of the outputs of two (or more) transmission branches, we define the following operation further.

Union Operation

$$c_1 * \alpha_1 \# c_2 * \alpha_2 = \begin{cases} c_1 * \alpha_1 & \text{if} \quad c_2 * \alpha_2 = \Phi \\ c_2 * \alpha_2 & \text{if} \quad c_1 * \alpha_1 = \Phi \end{cases} \tag{15}$$

In Eq. (15), the transmission operation * takes priority over the Union operation #. Note that 1 # Φ = 1 and 0 # Φ = 0. We notice that if $c_1 \neq c_2$ there must be $\alpha_1 \cdot \alpha_2 = \mathrm{F}$, so that a voltage conflict between sources c_1 and c_2 is avoided. As an example, we can use the above operations to re-express $\bar{x}$ in Eq. (4) at switch level:

$$\bar{x} = 1 * x^{0.5} \# 0 * {}^{0.5}x \,, \tag{16}$$

Which exactly describes the circuit structure in Fig.1(a) and its working process shown in Table 1. We find that the exclusive relationship between two switching functions ($x^{0.5}$, ${}^{0.5}x$) is naturally conformed.

It can be proved that the following laws related to the transmission operation and union operation hold.

Serial transmission law

$$(c*\alpha_1)*\alpha_2 = c*(\alpha_1 \cdot \alpha_2)\,, \tag{17}$$

Parallel transmission law

$$c*\alpha_1 \# c*\alpha_2 = c*(\alpha_1 + \alpha_2)\,, \tag{18}$$

Commutation law

$$c_1*\alpha_1 \# c_2*\alpha_2 = c_2*\alpha_2 \# c_1*\alpha_1\,, \tag{19}$$

Associative law

$$(c_1*\alpha_1 \# c_2*\alpha_2) \# c_3*\alpha_3 = c_1*\alpha_1 \# (c_2*\alpha_2 \# c_3*\alpha_3) = c_1*\alpha_1 \# c_2*\alpha_2 \# c_3*\alpha_3\,, \tag{20}$$

Distributive law

$$(c_1*\alpha_1 \# c_2*\alpha_2)*\alpha_3 = c_1*(\alpha_1 \cdot \alpha_3) \# c_2*(\alpha_2 \cdot \alpha_3)\,, \tag{21}$$

We can use the connection operations to derive a new canonical function form. For example, a two-variable function *f* (*x*, *y*) has the following canonical expansion form at the switch level:

$$f(x,y) = f(0,0)*(x^{0.5} \cdot y^{0.5}) \# f(0,1)*(x^{0.5} \cdot {}^{0.5}y) \# f(1,0)*({}^{0.5}x \cdot y^{0.5}) \# f(1,1)*({}^{0.5}x \cdot {}^{0.5}y) \tag{22}$$

In comparison, the two-variable function *f* (*x*, *y*) has its traditional min-term expansion at the gate level:

$$f(x,y) = [\,f(0,0) \,\mathrm{I}\, \bar{x} \,\mathrm{I}\, \bar{y}]\mathrm{Y}[\,f(0,1) \,\mathrm{I}\, \bar{x} \,\mathrm{I}\, y]\mathrm{Y}[\,f(1,0) \,\mathrm{I}\, x \,\mathrm{I}\, \bar{y}]\mathrm{Y}[\,f(1,1) \,\mathrm{I}\, x \,\mathrm{I}\, y] \tag{23}$$

Equation (23) shows how the circuit is realized by using gates, which could be renamed Complement gate (inverter), Minimum gate (AND gate) and Maximum gate (OR gate), as show in Fig.4 (a). However, Eq. (22) explains how four signals, *f* (*i*, *j*), are transmitted to the output through two switches in series, as shown in Fig.3 (b). These equations illustrate the difference in philosophy between switch-level and gate-level descriptions. In fact, the structure shown in Fig.4 (b) is relative to the pass transistor circuits. Therefore, the above new description for CMOS circuits is also suitable to nMOS pass transistor circuits.

Figure 4

Fig. 4 Circuit realizations of a two-variable function (a) at gate level, (b) at switch level.

3. Application to Designing CMOS Circuits

Since in Eq. (22) the expansion coefficient $f\ (i,\ j) \in \{0,1\}$, we can factor coefficients 1 and 0, respectively, and obtain the following form:

$$f = 1{*}^{0.5}f\#0{*}\, f^{\,0.5}\,, \tag{24}$$

Where $^{0.5}f$ and $f^{\,0.5}$ are complementary. They are the switching functions of source 1 and source 0, respectively. By using Eq. (9) the above equation can be rewritten as

$$f = 1{*}^{0.5}f\#0{*}^{0.5}\bar{f}\ . \tag{25}$$

If we have the simplified function expression $\overline{f(x,y,\Lambda\,;-,\mathrm{I}\,,\mathrm{Y})}$ in traditional binary signal algebra, we can easily derive its corresponding switch-level expression $^{0.5}\overline{f(x,y,\Lambda\,;-,\mathrm{I}\,,\mathrm{Y})}$ by using Eq. (13). The latter shows how to use serial and parallel nMOS switch connections for controlling the transmission of source 0. According to $^{0.5}\bar{f} = \widetilde{{}^{0.5}f}$ in Eq. (25) and De Morgan's Law, the two switch-level expressions, $^{0.5}f$ and $^{0.5}\bar{f}$, are dual. And that is the principle of the design procedure presented in [1].

Taking $f_1 = \overline{x\,\mathrm{I}\ y}$ and $f_2 = x \oplus y$ as examples, we have $\bar{f}_1 = x\,\mathrm{I}\ y$ and $\bar{f}_2 = (x\,\mathrm{I}\ y)\mathrm{Y}(\bar{x}\,\mathrm{I}\ \bar{y})$. Then the following switch-level expressions can be obtained:

$$^{0.5}\bar{f}_1{=}^{0.5}x{\cdot}^{0.5}y \quad \text{And} \quad {}^{0.5}\bar{f}_2{=}^{0.5}x{\cdot}^{0.5}y{+}^{0.5}\bar{x}{\cdot}^{0.5}\bar{y}\,. \tag{26}$$

The above two expressions describe the n-branches in Fig.2 (b) and (c). By De Morgan's Law, we have the following dual expressions for describing the corresponding p-branches:

$$^{0.5}f_1 = x^{0.5} + y^{0.5} \quad \text{And} \quad {}^{0.5}f_2 = (x^{0.5} + y^{0.5})\cdot(\bar{x}^{0.5} + \bar{y}^{0.5})\,. \tag{27}$$

In fact, the duality between p-part and n-part in a CMOS circuit is unnecessary. Because $f_2 = x \oplus y$ also can be expressed as $f_2 = (x\,\mathrm{I}\ \bar{y})\mathrm{Y}(\bar{x}\,\mathrm{I}\ y)$, we have the following expression instead:

$$^{0.5}f_2{=}^{0.5}x{\cdot}^{0.5}\bar{y}{+}^{0.5}\bar{x}{\cdot}^{0.5}y = \bar{x}^{0.5}\cdot y^{0.5} + x^{0.5}\cdot\bar{y}^{0.5}\,. \tag{28}$$

The above expression will guide a new pMOS connection model, which would eliminate the internal connection in the p-branch in Fig. 2(c).

Another example is given in design of a combinational adder as follows. For a circuit with output inverting buffers, we can design a circuit with inverse outputs, $\overline{C_+}$ and $\bar{S}$, first. From Eq. (25) we have

$$\overline{C_+} = 1{*}^{0.5}\overline{C_+}\#0{*}^{0.5}C_+\,, \tag{29}$$

$$\bar{S} = 1{*}^{0.5}\bar{S}\#0{*}^{0.5}S\ . \tag{30}$$

Based on the traditional Boolean algebra the following expressions are derived:

$$C_+ = (A\,\mathrm{I}\ B)\mathrm{Y}[(A\,\mathrm{Y}\,B)\,\mathrm{I}\ C], \tag{31}$$

$$\overline{C_+} = (\bar{A}\,\mathrm{I}\ \bar{B})\mathrm{Y}[(\bar{A}\,\mathrm{Y}\,\bar{B})\,\mathrm{I}\ \bar{C}], \tag{32}$$

$$S = [(A\,\mathrm{Y}\,B\,\mathrm{Y}\,C)\,\mathrm{I}\ \overline{C_+}]\mathrm{Y}(A\,\mathrm{I}\ B\,\mathrm{I}\ C), \tag{33}$$

$$\bar{S} = [(\bar{A}\,\mathrm{Y}\,\bar{B}\,\mathrm{Y}\,\bar{C})\,\mathrm{I}\ C_+]\mathrm{Y}(\bar{A}\,\mathrm{I}\ \bar{B}\,\mathrm{I}\ \bar{C})\,. \tag{34}$$

Note that in the above expressions C_+ and $\overline{C_+}$, S and $\bar{S}$ are symmetric rather than dual with each other. By using Eq. (13), their corresponding expressions at switch-level are:

$$^{0.5}\overline{C_+} = A^{0.5} \cdot B^{0.5} + (A^{0.5} + B^{0.5}) \cdot C^{0.5}\,, \tag{35}$$

$$^{0.5}C_+ = {}^{0.5}A \cdot {}^{0.5}B + ({}^{0.5}A + {}^{0.5}B) \cdot {}^{0.5}C\,, \tag{36}$$

$$^{0.5}\overline{S} = (A^{0.5} + B^{0.5} + C^{0.5}) \cdot \overline{C_+}^{0.5} + A^{0.5} \cdot B^{0.5} \cdot C^{0.5}\,, \tag{37}$$

$$^{0.5}S = ({}^{0.5}A + {}^{0.5}B + {}^{0.5}C) \cdot {}^{0.5}\overline{C_+} + {}^{0.5}A \cdot {}^{0.5}B \cdot {}^{0.5}C\,. \tag{38}$$

Therefore we obtain the corresponding circuit design at switch level, as shown in Fig. 5 [1]. Obviously, the p-branch and n-branch in the optimized schematic are symmetric.

Figure 5

Fig. 5 CMOS design of a full adder.

In addition to transformation from the traditional Boolean expression, the switch-level expression can be also derived directly from truth table or Karnaugh map. Figure 6 shows three Karnaugh maps for outputs of the NAND gate, the transmission gate, and the Exclusive-OR gate. From the mapping synthesis shown in Fig.6 (a), we directly obtain the switch-level expression:

$$\overline{x \,\mathrm{I}\, y} = 1 * (x^{0.5} + y^{0.5}) \# 0 * ({}^{0.5}x \cdot {}^{0.5}y)\,. \tag{39}$$

The above expression describes how the source 1 (V_{DD}) transmitted through two p-transistors in parallel to the output, and the source 0 (Ground) transmitted through two n-transistors in series to the output, as shown in Fig. 6(a).

According to the mapping in Fig. 6(b), we use the variable *c* as its transmitted source and obtain the switch-level expression:

$$f = c * {}^{0.5}x\,. \tag{40}$$

The above expression shows that the source *c* is transmitted by an n-transistor. To avoid the poor transmission for *c* = 1 we rewrite the expression as:

$$f = c * ({}^{0.5}x + \overline{x}^{0.5})\,, \tag{41}$$

Where two terms, ${}^{0.5}x$ and $\overline{x}^{0.5}$, are equal based on Eq. (9). However, they describe that the variable source is transmitted by a complementary MOS construction, as shown in Fig. 6(b).

Figure 6

Fig. 6 Mapping synthesis of some CMOS circuits at switch-level (a) NAND gate, (b) Transmission gate, (c) Exclusive-OR gate.

Based on the previous example, we can synthesize Exclusive-OR by its Karnaugh map in Fig. 6(c):

$$x \oplus y = y * (x^{0.5} + {}^{0.5}\overline{x}) \# x(1) * y^{0.5} \# \overline{x}(0) * {}^{0.5}y\,. \tag{42}$$

Notice that the parts overlapped in Karnaugh map have been realized by $y * (x^{0.5} + {}^{0.5}\overline{x})$, and therefore the branch with source *x* (term $x(1) * y^{0.5}$) only has to transmit a 1 and never a 0, and the branch with source $\overline{x}$ (term $\overline{x}(0) * {}^{0.5}y$) only has to transmit a 0 and never a 1. Hence each of these transmission branches can be realized with a single MOS transistor as shown in Fig. 6(c). This simple circuit realization previously has been considered as something that "… does not follow from any systematic (design) method" [13]. Comparing all three designs of Exclusive-OR gate in Fig. 1(c), Fig. 2(c) and Fig. 6(c) we find that the number of transistors in the circuits are 16, 12 and 6, respectively. Besides, the numbers of internal nodes in circuits are 5, 3 and 2, respectively. This means that the design based on switch-level description may lead to a circuit with a simpler structure as well as higher quality (delay and power).

It should be pointed out that the description of MOS circuits at switch-level also offers a new method for analyzing and optimizing circuits. However, since there are physical capacitance and resistance associated with the MOS transistors, the permutation of the inputs to a series chain of transistors will lead to different input pin loads and pin dependent delays. It is well known that the signal to pin assignment in a CMOS logic gate has a sizable impact on the propagation delay through the gate [14]. Besides, researchers have pointed out that the assignments of input signal with different probability of assuming a controlling value (zero for nMOS and one for pMOS), or input signal with different switching activity when all other inputs are set to their non-controlling values must be considered for power reduction [14,15]. Therefore, we have to describe the circuit more accurately at switch level. It means that each transistor in series-connected structure should be located exactly. In fact, by weakening the communication law of switching variables, those two switching variables in Eq. (17) cannot be permuted. Thus, the order of AND'ed switching variables will represent corresponding location accurately in the series-connected structure. Taking the Exclusive-OR gate shown in Fig. 2(c) as the example, its output can be expressed as

$$x \oplus y = 1{*}[(x^{0.5} + y^{0.5}) \cdot (\bar{x}^{0.5} + \bar{y}^{0.5})]\#0{*}[{}^{0.5}x \cdot {}^{0.5}y + {}^{0.5}\bar{x} \cdot {}^{0.5}\bar{y}]\,. \tag{43}$$

The above expression indicates that the two nMOS transistors, which are controlled by signals x and $\bar{x}$ are close to Ground since their corresponding switching variables, are near the source 0 in the above expression. Obviously, the other two nMOS transistors are close to the output terminal. Therefore, the proposed description of CMOS circuits is modified to locate each transistor in the circuit. It is expected that the new switch-level description will provide a basis for analyzing and optimizing delay and power dissipation of CMOS circuits.

4. Application to Designing nMOS Pass Transistor Circuits

An nMOS pass-transistor circuit, as shown in Fig.7 (a) has a structure similar to the one in Fig.4 (b) except for the following differences:

(1) Each switch in the transmission branches is a nMOS transistor independent of the signal value, 0 or 1, to be transmitted. In contrast, if the transmitted signal f (i, j) in Fig.4 (a) is a variable source, each switch in the branch should be a transmission gate, which consists of a pair of complementary MOS transistors.

(2) Since nMOS transistor has a poor transmission for high level, it is necessary to restore the output level using an output inverter, as shown in Fig.7 (a). This inverter consists of three MOS transistors, as shown in Fig.7 (b). The feedback to the pull-up pMOS transistor improves the slope of transient pass-transistor output waveform if the input of the inverter is at high level. If the load capacitance is extremely large, an extra feedback inverter should be added to avoid its influence on the feedback action [11]. We notice that in Fig.7 (a) all source signals are inverted because of the output inverter.

Figure 7

Fig. 7 (a) A nMOS pass-transistor circuit, (b) output inverter.

By explaining the differences between conventional CMOS transmission circuit and nMOS pass-transistor circuit we can modify the new description for CMOS circuit and use it to describe and design nMOS pass-transistor circuits as follows.

In the design of CMOS circuits, since source 1 should be transmitted by pMOS transistors, ${}^{0.5}f$ in Eq. (25) have to be expressed by arguments $x^{0.5}$, $y^{0.5}$, etc. However, source 1 in nMOS pass-transistor circuit is also transmitted by nMOS transistors, therefore, the low threshold comparison operations $x^{0.5}$, $y^{0.5}$, etc. should not appear. For example, consider $f_1 = \overline{x \mathrm{I}\ y} = \bar{x} \mathrm{Y} \bar{y}$ and $f_2 = x \oplus y = (x \mathrm{I}\ \bar{y}) \mathrm{Y} (\bar{x} \mathrm{I}\ y)$ as in the last section we obtain following expressions, in place of Eqs. (27). (Note that Eq. (26) remains unchanged).

$$ {}^{0.5}f_1 = {}^{0.5}\bar{x} + {}^{0.5}\bar{y} \text{ And } {}^{0.5}f_2 = {}^{0.5}\bar{x} \cdot {}^{0.5}y + {}^{0.5}x \cdot {}^{0.5}\bar{y}, \qquad (44) $$

Figure 8

Fig.8 nMOS pass-transistor circuits with constant source (a) $\overline{x \mathrm{I}\ y}$, (b) $x \oplus y$.

The above expressions describe the nMOS pass-transistor circuits for realizing f_1 and f_2, as show in Fig.8 (a) and (b). If we add an output inverter to restore the logic level, the source signals should be inverted. If comparing the above nMOS designs with CMOS designs in Fig.2 (b) and (c), we cannot see any advantage with the nMOS design. In fact, the feature of nMOS pass-transistor circuits is that nMOS transistor is used to transmit both source 0 and source 1. This means that a single nMOS transistor can be used to transmit a variable source. The use of variable signal sources rather than constant sources frequently allows simpler circuit to be designed. As an example, we can rewrite $x \oplus y$ as

$$ x \oplus y = (\bar{x} \mathrm{I}\ y) \mathrm{Y} (x \mathrm{I}\ \bar{y}) = \begin{cases} y, & \text{if } x = 0; \\ \bar{y}, & \text{if } x = 1. \end{cases} \qquad (45) $$

By using the high threshold comparison operation the above equation can be transformed into a switching expression:

$$ x \oplus y = y * x^{0.5} \# \bar{y} * {}^{0.5}x = y * {}^{0.5}\bar{x} \# \bar{y} * {}^{0.5}x \,. \qquad (46) $$

The corresponding nMOS pass-transistor circuit is shown in Fig.9 (a), which is much simpler than the one in Fig.8 (b). (In fact, if the two nMOS transistors controlled by x and the two nMOS transistors controlled by $\bar{x}$ in Fig.8 (b) are merged respectively, we get the design of Fig.9 (a).) The example indicates that as long as a Boolean function $f(x, y, \Lambda\ , \mathrm{I}\ , \mathrm{Y})$ is expressed in a MUX form by using the Shannon expansion:

$$ f(x, y, \Lambda\ , \mathrm{I}\ , \mathrm{Y}) = [f(0, y, \Lambda\ , \mathrm{I}\ , \mathrm{Y}) \mathrm{I}\ \bar{x}] \mathrm{Y} [f(1, y, \Lambda\ , \mathrm{I}\ , \mathrm{Y}) \mathrm{I}\ x] $$

(47)

We will have its corresponding switching expression:

$$ f(x, y, \Lambda\ , \mathrm{I}\ , \mathrm{Y}) = f(0, y, \Lambda\ , \mathrm{I}\ , \mathrm{Y})^{*0.5}\bar{x} \# f(1, y, \Lambda\ , \mathrm{I}\ , \mathrm{Y})^{*0.5}x, $$

(48)

Which will lead its realization with nMOS pass-transistor MUX. Notice that in the above expression the exclusive relationship of two switching functions is met by ${}^{0.5}\bar{x} \cdot {}^{0.5}x = \mathrm{F}$.

Taking conventional two-variable Boolean operations, $x \mathrm{I}\ y$ and $x \mathrm{Y} y$ as examples we have

$$x \mathrm{I}\ y = (0 \mathrm{I}\ \bar{y}) \mathrm{Y} (x \mathrm{I}\ y) = 0*^{0.5}\bar{y} \# x*^{0.5}y\,, \tag{49}$$

$$x \mathrm{Y} y = (x \mathrm{I}\ \bar{y}) \mathrm{Y} (1 \mathrm{I}\ y) = x*^{0.5}\bar{y} \# 1*^{0.5}y\,. \tag{50}$$

The corresponding nMOS pass-transistor circuits are shown in Fig.9 (b) and (c). If we add the output inverter to restore level, all transmitted sources in these circuits should be inverted.

Figure 9

Fig.9 nMOS pass-transistor circuits with variable source (a) $x \oplus y$, (b) $x \mathrm{I}\ y$, and (c) $x \mathrm{Y} y$.

Furthermore, we take the full adder as a practical design example. If we introduce $C_p = A \oplus B$, which is the carry propagation term, as an intermediate variable, we obtain the following Boolean expressions:

$$C_p = A \oplus B = (B \mathrm{I}\ \bar{A}) \mathrm{Y} (\bar{B} \mathrm{I}\ A)\,, \tag{51}$$

$$C_+ = (A \mathrm{I}\ \bar{C}_p) \mathrm{Y} (C \mathrm{I}\ C_p)\,, \tag{52}$$

$$S = (C \mathrm{I}\ \bar{C}_p) \mathrm{Y} (\bar{C} \mathrm{I}\ C_p)\,. \tag{53}$$

Based on the above design procedure we can construct the nMOS pass transistor circuit with simple nMOS MUX, as shown in Fig.10 (a). In the circuit only 14 MOS transistors are used. However, the simple nMOS MUX without output inverter cannot work well. If we use the inverted nMOS MUX with output inverter in Fig.7 (b) instead, we can obtain the improved design by inserting inverters into the circuit of Fig.10 (a), as shown in Fig.10 (b). The basic building cell is framed by dotted line in Fig.10 (a). The conventional design shown in Fig.5 requires 28 MOS transistors. However, our nMOS pass-transistor design only needs 23 MOS transistors and has an extra carry propagation output C_P, which is useful for carry look ahead technique.

Figure 10

Fig. 10 nMOS pass-transistor design of a full adder

(a) With simple nMOS MUX, (b) with inverted nMOS MUX.

The above design procedure of nMOS pass-transistor network is based on the binary division expressed by Eq. (48). The exclusive relationship between two switching functions is guaranteed for each division operation. This procedure makes the design of nMOS pass-transistor circuits as same as the synthesis for MUX networks. In fact, this procedure also conforms to the representation of BDD. Therefore, the designed circuit construction based on Eq. (48) can be described by its corresponding BDD representation. For example, for circuits in Fig.9 and Fig.10 (a) we have their corresponding BDDs, as shown in Fig.11. An effective synthesis method of pass-transistor network based on BDD has been developed [16]. As long as the reduced BDD is obtained it can be mapped to nMOS transistor circuit easily (e.g. we derive circuits in Fig.9 and Fig.10 (a) from BDDs in Fig.11). We notice that in Fig.11 (d) we adopt super-node labeled by C_P in the BDD [17]. It can reduce number of transistors and delay.

Figure 11

Fig.11 BDDs (a) $x \oplus y$, (b) $x \mathrm{I}\ y$, (c) $x \mathrm{Y} y$, (d) C_{+}, S and C_{P}.

However, we should point out that synthesis of nMOS pass transistor network does not have to be confined to binary division given by Eq. (48). For example, we can express $x \mathrm{I}\ y \mathrm{I}\ z$ as follows:

$$x \mathrm{I}\ y \mathrm{I}\ z = x *^{0.5}(y \mathrm{I}\ z) \# 0 *^{0.5}(\overline{y \mathrm{I}\ z}) = x * ({}^{0.5}y \times {}^{0.5}z) \# 0 * ({}^{0.5}\overline{y} \times {}^{0.5}\overline{z})$$

(54)

The corresponding circuit realization is shown in Fig.12 (a). This circuit is not mapped from BDD. Therefore, the synthesis of nMOS transistor network based on BDD is just one canonical design method by using the new description of MOS circuits at switch level. A bridge-connected circuit shown in Fig.12 (b) gives another example. It is difficult to analyze its function by traditional method. However, we can describe it at switch level as follows. If $c = 1$, the above bridge transistor conducts and the below one shuts off. Based on the switch connection we can get

$$f(c=1) = 1 * [({}^{0.5}a + {}^{0.5}b) \cdot ({}^{0.5}d + {}^{0.5}e)] \# 0 * [{}^{0.5}\overline{a} \cdot {}^{0.5}\overline{b} + {}^{0.5}\overline{d} \cdot {}^{0.5}\overline{e}]. \qquad (55)$$

Similarly, if $c = 0$, we have

$$f(c=0) = 1 * [{}^{0.5}a \cdot {}^{0.5}d + {}^{0.5}b \cdot {}^{0.5}e] \# 0 * [({}^{0.5}\overline{a} + {}^{0.5}\overline{d}) \cdot ({}^{0.5}\overline{b} + {}^{0.5}\overline{e})].$$

(56)

It is easy to prove that two switching functions in Eq. (55) (or Eq. (56)) are exclusive with each other. It guarantees that the circuit shown in Fig.12 (b) is source-conflict free.

Figure 12

Fig.12 (a) Three AND gate, (b) nMOS circuit with bridge connection.

From Eq. (25) we obtain

$${}^{0.5}f = {}^{0.5}c \cdot [({}^{0.5}a + {}^{0.5}b) \cdot ({}^{0.5}d + {}^{0.5}e)] + {}^{0.5}\overline{c} \cdot [{}^{0.5}a \cdot {}^{0.5}d + {}^{0.5}b \cdot {}^{0.5}e].$$

Thus, we have

$$\begin{aligned} f &= \{c \mathrm{I}\, [(a \mathrm{Y} b) \mathrm{I}\ (d \mathrm{Y} e)]\} \mathrm{Y} \{\overline{c} \mathrm{I}\, [(a \mathrm{I}\ d) \mathrm{Y} (b \mathrm{I}\ e)]\} \\ &= \{c \mathrm{I}\, [(a \mathrm{Y} e) \mathrm{I}\ (b \mathrm{Y} d)]\} \mathrm{Y} [(a \mathrm{I}\ d) \mathrm{Y} (b \mathrm{I}\ e)] \end{aligned} \qquad (57)$$

With the traditional AND-OR operation symbols the above equation can be expressed as

$$f = c \times (a \times e + b \times d) + (a \times d + b \times e).$$

Obviously, the BDD representation for the above function will have a different and more complicated form compared to the circuit in Fig.12 (b). From the above discussion, we demonstrate that the new description for MOS transistor promises a more comprehensive circuit construction than BDD. Besides, we find that in the circuit two bridges transistors are used to connect two signals. Thus, the bi-directional characteristic of MOS transistor is used in the design.

5. Conclusion

The traditional description of CMOS circuits is based on Boolean algebra, where three basic operations, NOT, AND and OR, are used to describe functional relationship between inputs and the output, and to describe the circuit structure composed of gates. However, it cannot be used to describe the internal structure of MOS transistor switches. Besides, there exist some problems with Boolean algebra when describing complex gates or gates with high-impedance state. This paper introduced another variable to describe the switching state of transistors in addition to original variable, which describes signal in the circuit. The two variables have their own independent operations. Since there exists a mutual relationship between the on-off states of switch elements and the signals, we proposed two connection operations for describing their interaction, whereby a new description for CMOS circuits at the switch level is presented. Based on the new description the design of CMOS circuits at switch level can be realized. For CMOS circuits the traditional inverting-logic stage design and pass-transistor design have been considered to be two different design methods. However, the new description proposed in this paper can unite the two and overcome other difficulties in the traditional theory. The theory is also applicable to the description and design of nMOS pass-transistor circuits.

Acknowledgment

This work was supported in part by DARPA under contract #F33615-95-C-1627 and NNSF of China.

References

[1] N. H. E. Weste and K. Eshraghian, Principles of CMOS VLSI Design: A Systems Perspective, 2nd Edition, Addison-Wesley Publishing Company, New York, 1993.

[2] J. P. Shen and S. Hirschhorn, "Switch-level techniques," IEEE Design & Test of Computers, Vol.4, No.4, pp.15-16, 1987.

[3] P. Hares, "An introduction to switch-level modeling," IEEE Design & Test of Computers, Vol.4, No.4, pp.18-25, 1987.

[4] T. Kuroda and T. Sakurai, "Overview of low-power ULSI circuit techniques," IEICE Trans. Electorn., Vol.78-C, no.4, pp.334-343, April 1995.

[5] D. Radhakrishnan, S. R. Whitaker and G. K. Maki, "Formal design procedures for pass transistor switching circuits," IEEE J. Solid-state Circuits, Vol.SC-20, no.2, pp.531-536, April 1985.

[6] K. Yano, T. Yamanaka, T. Nishida, M. Saito, K. Shimohigashi and A. Shimizu, "A 3.8-ns CMOS 16*16-b multiplier using complementary pass-transistor logic," IEEE J. Solid-state Circuits, Vol.SC-25, no.2, pp.388-395, April 1990.

[7] F. S. Lai, W. Hwang, "Differential cascade voltage switch with the pass-gate (DCVSPG) logic tree for high performance CMOS digital systems," Proc. IEEE VLSITSA, pp.358-362, 1993.

[8] M. Suzuki, N. Ohkubo, T. Yamanaka, A. Shimuzu and K. Sasaki, "A 1.5ns 32b CMOS ALU in double pass-transistor logic," ISSCC Dig. Tech. Papers, pp.90-91, Feb. 1993.

[9] A. Parameswar, H. Hara and T. Sakurai, "A high based multiply and accumulate circuit for multimedia applications," Proc. IEEE CICC, pp.278-281, May 1994.

[10] M. Matsui, H. Hara, K. Seta, Y. Uetani, L.-S. Kim, T. Ohto, Y. Watanabe, F. Sano, A. Chiba, K. Matsuda and T. Sakurai, "200 MHz video compression Macrocells using low-swing differential logic," ISSCC Dig. Tech. Papers, pp.76-77, Feb. 1994.

[11] K. Yano, Y. Sasaki, K. Rikino and K. Seki, "Top-down pass-transistor logic design," IEEE J. Solid-state Circuits, Vol.SC-31, no.6, pp.792-803, June 1996.

[12] D. Radhakrishnan, S. R. Whitaker and G. K. Maki, "Formal design procedures for pass transistor switching circuits," *IEEE J. Solid-state Circuits*, Vol.SC-20, pp.531-536, 1985.

[13] A. Mukherjee, Introduction to nMOS and CMOS VLSI System Design, Englewood Cliffs, NJ: Prentice-Hall, 1986.

[14] M. Pedram, "Power minimization in IC Design: Principles and applications," ACM Transactions on Design Automation of Electronic Systems, vol. 1, no.1, pp.3-56, Jan. 1996.

[15] S. C. Prasad and K. Roy, "Circuit optimization of power consumption under delay constraint," in *Proceedings of the 1994 International Workshop on Low Power Design*, pp.15-20, April 1994.

[16] T. Sakurai, B. Lin, and A. R. Newton, "Multiple-output shared transistor logic (MOSTL) family synthesized using binary decision diagram," Memorandum UCB/ERL M90/21, Electronics Research Laboratory, College of Engineering, University of California, Berkeley, March 1990.

[17] N. Calazans, Q. Zhang, R. Jacobi, B. Yernaux and A.-M. Trullemans, "Advanced ordering and manipulation techniques for binary decision diagrams," *Proc. Eur. DAC*, pp.452-457, 1992.

Authors’ biography

Massoud Pedram received the BS degree in Electrical Engineering from California Institute of Technology and MS and Ph.D. degrees in Electrical Engineering and Computer Sciences from the University of California, Berkeley in 1989 and 1991, respectively. He then joined the Department of Electrical Engineering -Systems at the University of Southern California where he is currently an associate professor. He is a recipient of the National Science Foundation's Young Investigator Award in 1994 and a Presidential Early Career Award for Scientists and Engineers in 1996. His research has received a number of awards including one DAC Best Paper Award, one ICCD Best Paper Award, and the IEEE Trans. VLSI Systems Best Paper Award. He has served on the technical program committee of the DAC since 1992 and was the co-founder and General Chair of the 1994 International Workshop on Low Power Design. He is currently the General Co-chair of the International Symposium on Low Power electronics and Design. His research interests include computer-aided design of VLSI circuits and systems. His present research focuses on developing methodologies and techniques for lowering the power dissipation in electronic circuits and on layout-driven logic synthesis. He is a Member of IEEE - Circuits and Systems Society and ACM - SIGDA.

Xunwei Wu graduated in Physics from Hangzhou University, Hangzhou, China, in 1962. He then joined the Department of Physics (1962) and Department of Electronic Engineering (1989) at the Hangzhou University. Since 1990 he has been a Professor in Electronic Engineering. From 1986 to 1987 he was a visiting professor in Computer Science Department at Indiana University, USA, and from 1996 to the present he is a visiting professor in Department of Electrical Engineering -Systems at the University of Southern California, USA, to join Prof. Pedram’s research group. He is a recipient of the National Education Commission’s Science-Technology Progress Award (1988, 1994), National Natural Science Award (1995) and National Outstanding Scientist Award (1995) in China. He has served on the committee of Multiple-Valued Logic and Fuzzy Logic, Chinese Computer Federation since 1990 and is the vice-director of the committee from 1996. His research interests include logic synthesis, multiple-valued logic, switch-level design, current-mode circuits, and low-power analysis and design. He is a senior member of IEEE - Computer Society, a Fellow of Chinese Institute of Electronics, and a Fellow of Chinese Institute of Communication.

Figure captions

Fig.1 Definition and circuits for Exclusive-OR operation (a) truth table, (b) circuit composed of NOT, AND and OR gates, (c) circuit composed of NOT and NAND gates.

Fig .2 Switch structure of a few CMOS gates (a) inverter, (b) NAND gate, (c) composed Exclusive-OR gate, (d) transmission gate.

Fig. 3 Connections between binary switching variables and binary signal variables.

Fig. 4 Circuit realizations of a two-variable function (a) at gate level, (b) at switch level.

Fig. 5 CMOS design of a full adder.

Fig. 6 Mapping synthesis of some CMOS circuits at switch-level (a) NAND gate, (b) Transmission gate, (c) Exclusive-OR gate.

Fig. 7 (a) A nMOS pass-transistor circuit, (b) output inverter.

Fig.8 nMOS pass-transistor circuits with constant source (a) $\overline{x \mathrm{I}\ y}$, (b) $x \oplus y$.

Fig.9 nMOS pass-transistor circuits with variable source (a) $x \oplus y$, (b) $x \mathrm{I}\ y$, and (c) $x \mathrm{Y} y$.

Fig. 10 nMOS pass-transistor design of a full adder

(a) With simple nMOS MUX, (b) with inverted nMOS MUX.

Fig.11 BDDs (a) $x \oplus y$, (b) $x \mathrm{I}\ y$, (c) $x \mathrm{Y} y$, (d) C_+, S and C_P.

Fig.12 (a) Three AND gate, (b) nMOS circuit with bridge connection.

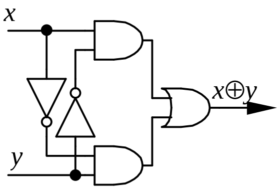
Table captions

Table 1 Relationship between signals and switching states in a CMOS inverter.

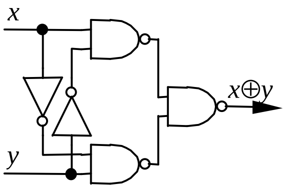
Figure 1

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

(a)

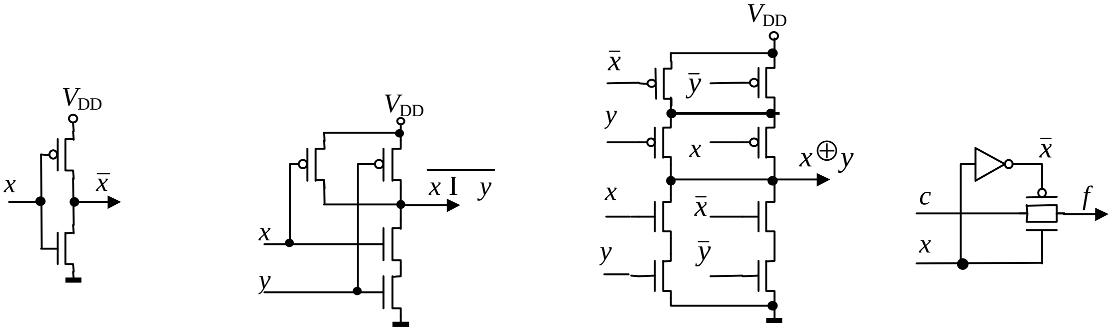


(b)



(c)

Figure 2



(a) (b) (c) (d)

Figure 3

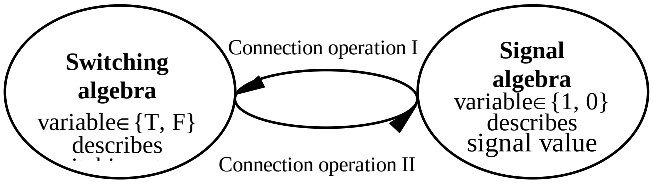

Switching algebra
variable∈{T, F}
describes
Connection operation I
Connection operation II
Signal algebra
variable∈{1, 0}
describes
signal value

Figure 4

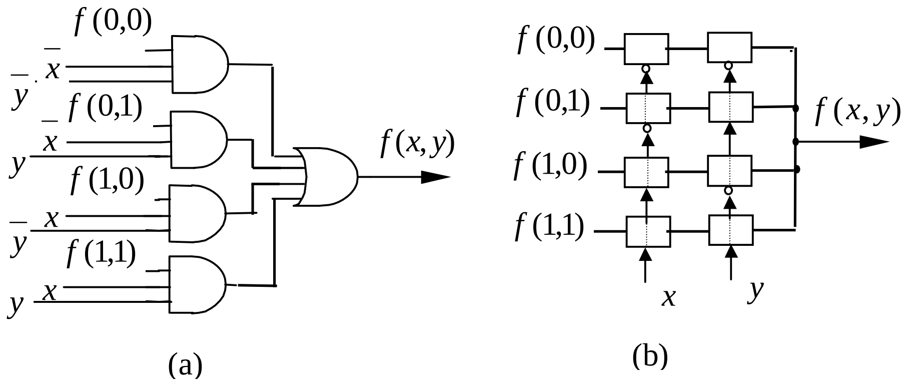

$f(0,0)$
$\bar{x}$
$\bar{y}$
$f(0,1)$
$\bar{x}$
y
$f(1,0)$
x
$\bar{y}$
$f(1,1)$
x
y
$f(x,y)$
(a)
$f(0,0)$
$f(0,1)$
$f(1,0)$
$f(1,1)$
$f(x,y)$
x
y
(b)

Figure 5

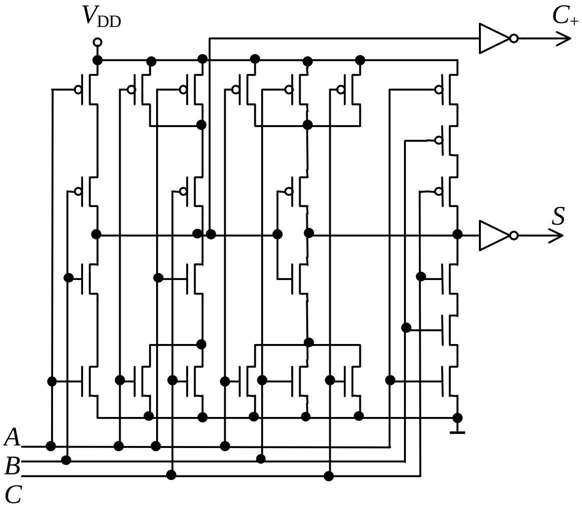


Figure 6

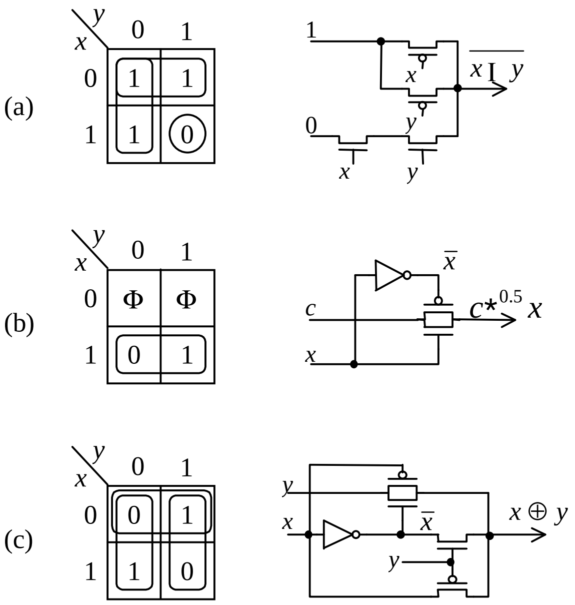
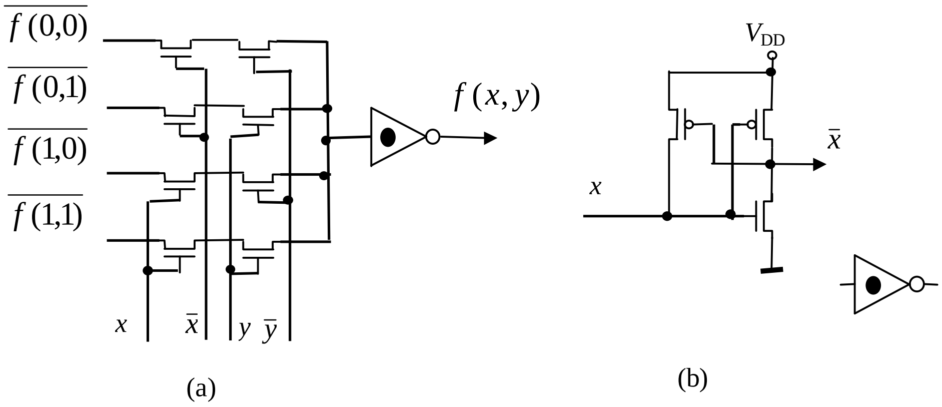

(a)
y
x
0
1
0
1
1
1
1
0
1
0
x
y
$\overline{x \text{ I } y}$
x
y
(b)
y
x
0
1
0
1
Φ
Φ
0
1
$\overline{x}$
c
x
$c *^{0.5} x$
(c)
y
x
0
1
0
1
0
1
1
0
y
x
$\overline{x}$
y
$x \oplus y$

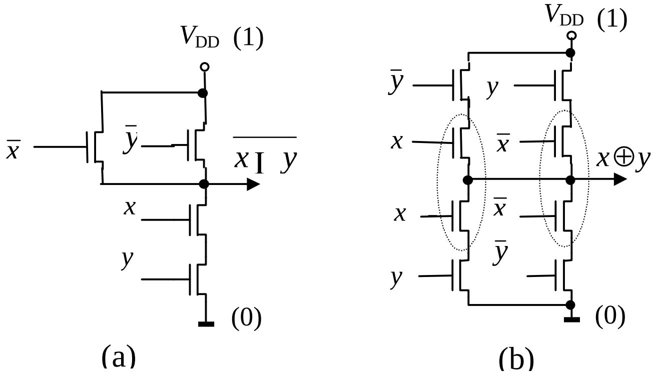
Figure 7


$\overline{f(0,0)}$
$\overline{f(0,1)}$
$\overline{f(1,0)}$
$\overline{f(1,1)}$
$f(x,y)$
x
$\bar{x}$
y
$\bar{y}$
V_{DD}
$\bar{x}$
x

(a)

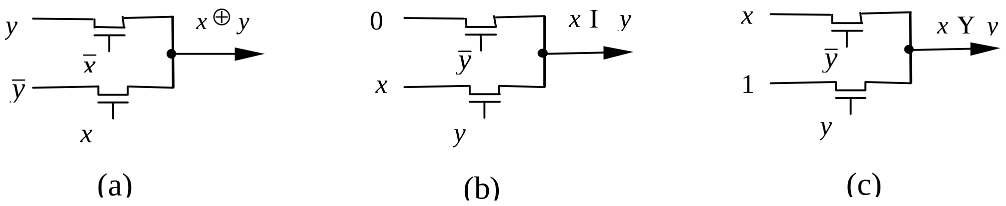
(b)

Figure 8



V_{DD} (1)
$\bar{x}$
$\bar{y}$
$\overline{x \mathrm{I}\ y}$
x
y
(0)
(a)
V_{DD} (1)
$\bar{y}$
y
x
$\bar{x}$
$x \oplus y$
x
$\bar{x}$
$\bar{y}$
y
(0)
(b)

Figure 9



(a)

(b)

(c)

Figure 10

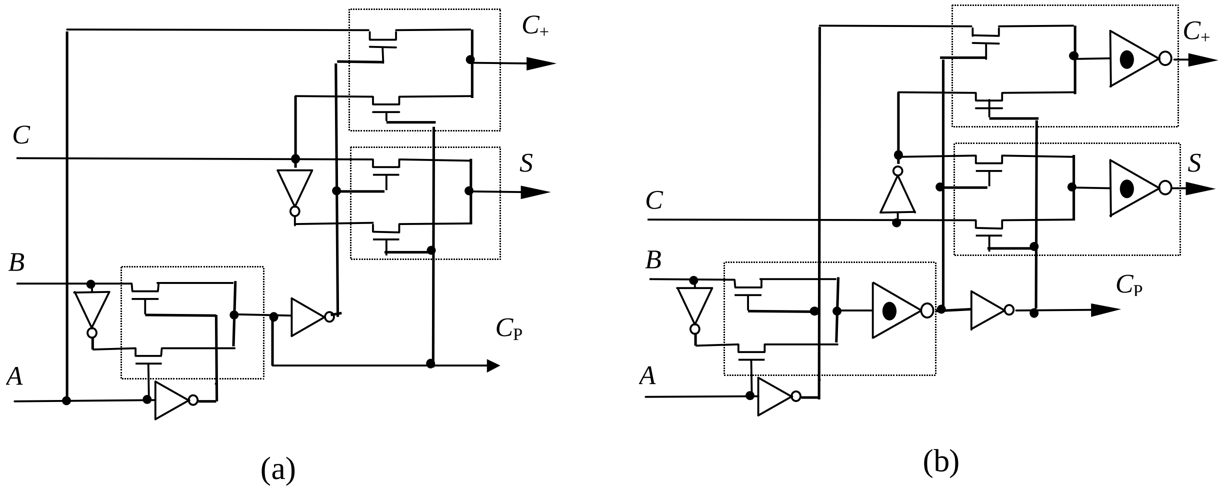


Figure 11

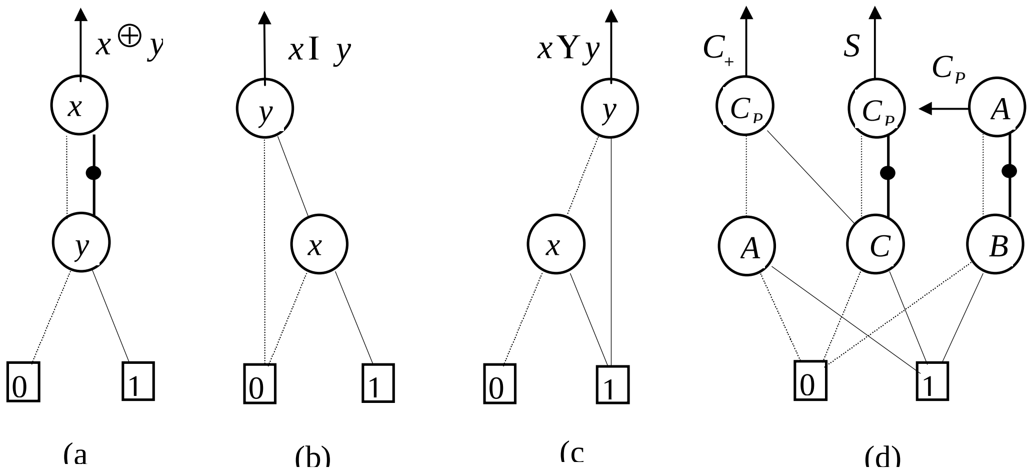
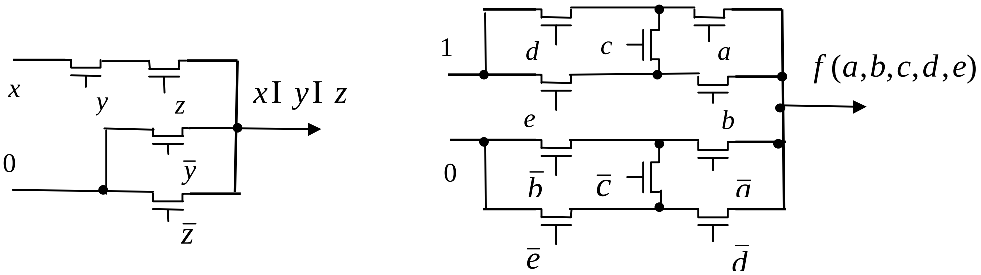


Figure 12


x
y
z
xI yI z
0
$\bar{y}$
$\bar{z}$
1
d
c
a
$f(a,b,c,d,e)$
e
b
0
$\bar{b}$
$\bar{c}$
$\bar{a}$
$\bar{e}$
$\bar{d}$

(a)

(b)

Table 1

x	α_{p}	α_{n}	$\bar{x}$
0 (low level)	T (on)	F (off)	1 (high level)
1 (high level)	F (off)	T (on)	0 (low level)