NAND CIRCUITS

Electronic circuits can be subdivided into analog and digital circuits. Analog circuits involve currents and voltages that are continuous in time, whereas digital circuits involve discontinuous signals, or pulses. Digital signals are thus considered to be binary in nature [i.e., a signal is either ON (high or 1) or OFF (low or 0)]. The binary nature of digital signals allows the concepts of Boolean algebra to be used to facilitate digital circuit design (also known as logic design). In logic design, circuit input and output signals are considered to be purely digital. Logic gates are the fundamental building blocks that form the foundation of digital logic circuitry. Examples of logic functions include NOT, AND, OR, NAND (NOT-AND), NOR (NOT-OR), and XOR (exclusive OR). Most of these operations can be achieved using one type of logic circuit. As will be discussed, the NAND circuit is often an optimal choice for implementing logic functions. Any digital combinational logic circuit (i.e., digital circuits without memory) can be constructed using the appropriate combination of NAND gates (or NOR gates). NANDs can also be combined with memory devices to form sequential logic circuits (i.e., digital circuits with memory). Specific applications involve circuit testability and design. Future trends involve improving the speed, power consumption, power handling, and temperature performance of NAND gates. NOR circuits are described in the article entitled NOR CIRCUITS.

DIGITAL LOGIC BACKGROUND

In Boolean algebra, logical inputs (true or false) are combined using logical operators to provide a specific function (1). For example, the NAND operation yields a True for all input combinations except for the case where all inputs are True. Logic circuits are often referred to as logic gates and are physical implementations of Boolean expressions. These digital circuits provide an output signal that is a predetermined function of the input signals. Logic gates perform the operations of Boolean algebra applied to two-state (binary) systems. Logic gates can have two or more inputs and typically have one output. In digital electronics, voltage levels are assigned to logic states which correspond to the True and False conditions. These are commonly referred to as 1 and 0, respectively. A NAND is a logic gate or Boolean operation that performs the NOT-AND function.

The predetermined logical outputs based upon various input combinations are typically expressed in truth tables. The most common two-input logic gate symbols and truth tables are shown in Fig. 1. The two-input NAND gate symbol and truth table are shown in Fig. 1(d). As suggested by the name, the NAND gate yields the inverse output as compared to the AND gate. Note that this can also be surmised from the NAND circuit symbol because it is a serial combination of the AND and NOT symbols.

Most commercially available logic gates are often realized using NAND or NOR gates (2). For power conservation, it is desirable to employ normally OFF devices so that the standby current is minimized. This results in logic implementations that impose an inverse operation. Thus, the inverting nature of NAND and NOR gates facilitates the use of inverting components for logic design. Figure 2 shows how combinations of NAND gates can be configured to realize NOT, AND, and OR functions.

NAND CIRCUIT TECHNOLOGIES

NANDs and other logic gates are composed of discrete electronic components. Logic gates require switching elements used to select either a 0 or 1 output level as a function of the appropriate input combinations. Historically, NANDs were created using vacuum tubes as switching elements. However, the advent of solid state electronics had a tremendous impact on logic gate fabrication and a variety of solid state components have been used to form logic families as shown in Table 1. Table 1 shows that logic gates within a given logic family are synthesized from only one type of active electronic device. This is caused by the nature of the wafer-scale fabrication process used to manufacture these devices. Logic gates are easily interconnected to other gates of the same family, and thus more complex digital circuits can be created at the wafer scale (i.e., prior to packaging electronic components). The ability to integrate logic gates at the wafer scale has resulted in high-density digital circuitry as found in personal computers and other electronic devices.

NAND gates can be synthesized using each of these logic families. Examples illustrating two-input NAND gate synthesis are shown in Fig. 3.

The operation of a NAND circuit depends upon the specific implementation. This can be better understood by further examining the CMOS implementation in Fig. 3(d), as an example. For the case where signal A is low (0 V), then transistor Q_1 is OFF (resulting in an open circuit between the source and drain terminals) and transistor Q_2 is ON (forming a short circuit path between the drain and source terminals). Thus, the voltage at node Y (circuit output) is tied through the drain-source of Q_2 to V_{dd}. If the signal A is high (5 V), then Q_1 is ON and Q_2 is OFF. The output voltage at node Y will depend then on signal B. When signal B is low, transistor Q_3 is OFF and transistor Q_4 is ON. This ties the output node Y to V_{dd} as was the case for signal A. When signal B is high, Q_3 is ON and Q_4 is OFF. This ties the node between the source of Q_1 and drain of Q_3 to ground through Q_3. If signal A is also high, Q_1 is ON, and node Y is tied to ground through Q_1 and Q_3. Thus for either signal A or signal B, if both are low or if either is low, the output node Y is tied to V_{dd} resulting in a high output signal. Only when *both* A and B are high does the output node Y become low. As can be seen, this circuit provides the NAND operation.

As highlighted in Table 1, logic gates can be constructed using either diodes or transistors as the switching devices required to realize both high and low logic states. Even though diodes were used in early gate developments, transistor technologies are much more prevalent as gate components. In general, BJTs are used for high-power and high-speed de-

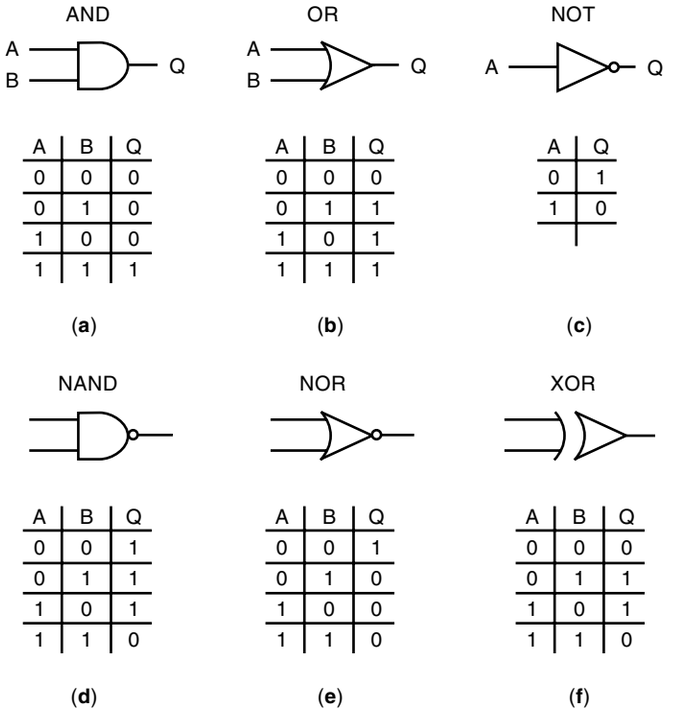

Figure 1. Schematic symbols and truth tables for basic digital logic gates: (a) AND, (b) OR, (c) NOT, (d) NAND, (e) NOR, and (f) XOR. Typically, NAND and NOR gates are the most widely utilized in circuit design (4).

vices, whereas MOSFETs are used for low-power consumption and more general-purpose applications. Even though logic family selection is application-specific, TTL and CMOS technologies are the most widely used in commercial electronic components (4,5).

NAND CHARACTERISTICS

NANDs and other logic gates are often characterized according to transfer characteristics. The factors used to characterize gate behavior include noise margins, power dissipation, fan-in, fan-out, propagation delay, and the delay-power product. These quantities are considered figures of merit and can be used for comparisons between technologies as well as their implementations. Noise margins are the minimum allowable voltage ranges that are defined for logic 1 or 0. The degree to which the noise margins are controlled represent one of the limits for low-voltage operation. Fan-in and fan-out refer to the number of possible inputs and outputs to a given logic gate. The ability of a logic gate to drive or sink current often determines the fan-in and fan-out. Propagation delay is the time delay for a signal to pass through the logic gate and ultimately determines the speed of operation (3).

NANDs and other individual logic gates are typically packaged in a plastic or ceramic housing known as a dual-in-line package (DIP). A DIP consists of a plastic compound encapsulating the integrated circuit, and metallic pins are utilized for breadboard or PC board connections.

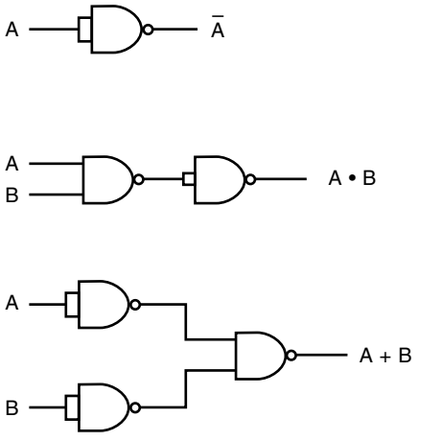

Figure 2. Realization of NOT, AND, and OR logic functions using NAND logic gates. This illustrates how NAND gates can be utilized exclusively to design the nonmemory components of digital logic circuits.

APPLICATIONS

Logic gates are combined to form more complex digital circuits. A variety of logic gate combinations can be used to design a digital circuit having a desired output. However, any digital circuit having a desired output can be designed using NAND or NOR gates. Digital circuit synthesis using NAND or NOR gates exclusively is standard due to increased efficiency of design based upon standard device manufacturing and packaging techniques. Thus, NAND and NOR gates, in practice, are the true building blocks of digital electronic circuitry and are found in most integrated logic circuits.

NANDs and other logic gates find primary applications in combinational and sequential logic circuits. Combinational logic circuits have outputs that are functions of present values of the input signals (i.e. have no memory). Examples of combinational logic circuits are gate arrays, multiplexers, decoders, adders, etc. Sequential logic circuits have outputs that are functions of past and present values of input signals, thus requiring memory components. Examples of sequential logic circuits are flip-flops and counters. NAND gates can be used exclusively in synthesizing any combinational logic circuit, whereas NAND gates plus some form of memory (e.g., flip-flops, delay lines, and inertial delays) are required to construct sequential logic circuits. Digital circuits can also be integrated with analog circuits to create "smart sensors," control systems, analog-to-digital conversion, and digital-to-analog conversion circuits (1).

In today's fast-paced marketplace, decreasing product time-to-market is of key interest to both manufacturers and consumers. Thus, simulation of digital circuit performance is often employed to decrease prototype costs and turnaround time. Models at various circuit levels are used to simulate digital circuit performance. Circuit models are typically selected at the transistor level, gate level, or a higher functional level. NAND gate modeling is common because digital circuits can be realized exclusively with NAND logic. Such NAND models are effectively defined as input and output arrays (i.e., truth tables) and used in circuit simulation via computers.

Even though simulating the performance of properly functioning circuits is of initial interest, it is equally important to simulate how circuits malfunction (6). Any condition that

Table 1. Digital Logic Family Characteristics (3–5)

Logic Family	Family Description	Properties and Characteristics
DTL	Diode–Transistor Logic	Popular in the 1960s Slow responses Ultimately replaced by TTL
RTL	Resistor–Transistor Logic	An early logic technology Narrow noise margins Large power dissipation
TTL	Transistor–Transistor Logic	Logic gates contain NPN BJTs, diodes, and resistors Devices driven to cutoff and saturation to achieve high/low logic states Used for moderately high computational speed Less susceptible to electrostatic discharge damage (ESD) Ideal for laboratory breadboarding
ECL	Emitter-Coupled Logic	BJTs and resistors connected in differential amplifier topology Devices driven below cutoff and saturation Used for ultra-high-speed switching applications
NMOS (*n*-channel metal-oxide semiconductor)	*n*-channel enhancement mode and depletion mode metal oxide semiconductor field effect transistors	Use transistors only (no resistors) High packing density Moderately high computational speeds Low-load current driving capability Not as standard as TTL or ECL More susceptible to ESD than BJT-based gates (requires anti-static packaging)
CMOS (complementary metal-oxide semiconductor)	*n*-channel and *p*-channel enhancement mode metal oxide semiconductor field effect transistors	Use transistors only (no resistors) High packing density Low power consumption General purpose: low power and high speeds More susceptible to ESD than BJT-based gates (requires anti-static packaging)

causes a logic device to malfunction is known as a fault. Thus, fault modeling, detection, and prevention is of major importance from both a simulation and measurement perspective. Faults can be permanent, intermittent, or transient, and models can be derived for each type. An example of a common fault model is the stuck-at fault model that simulates a wire being pinned at a high or low logic level. Network redundancy must be considered to determine the testability of various fault types. Fault detection involves determining either the existence or location of a fault. As an example, a pin-fault model is used to ignore faults internal to logic gates and only consider those associated with interconnections between logic gates. Volumes of research and industrial literature contain further information on fault modeling and detection that is beyond the scope of this article (1,6).

CURRENT RESEARCH IN LOGIC CIRCUITS

Current research efforts in NANDs and other logic gates have been used to increase switching speeds, decrease power consumption, increase power handling capability, increase packing density, increase fan-out, and increase temperature performance. Future research and advances in NAND gates should occur in these same areas as well as embarking upon new technologies for designing and fabricating NAND gates.

Semiconducting Diamond and Silicon Carbide Logic Circuits

Ever-increasing demands on performance of integrated circuits have spawned the development of new semiconductor materials such as silicon carbide and diamond. Diamond, for example, has a unique set of properties that make it an ideal candidate for use in extreme environments. Diamond possesses a large band-gap, is chemically inert, radiation-resistant, and robust as a material. Silicon carbide has similar properties.

Dreifus et al. (7) demonstrated diamond MOSFETs that operated at temperatures as high as 500°C as shown in Fig. 4 (7,8). Holmes et al. combined these devices into simple digital logic circuits (9). Operation of depletion-mode (normally on) diamond NAND circuits operating at 350°C and 400°C is shown in Fig. 5. Siergiej et al. (10) have shown operation of SiC NOR circuits using depletion-mode MOSFETs. Logic circuit operation of SiC logic circuits was demonstrated up to 300°C.

With the development of electronic devices that operate in extreme situations such as high temperature, radiation, and corrosive environments, designers can begin to generate autonomous sensing and control circuitry at the points of interest. These developments remove the necessity of extensive cabling to remote sensors that add appreciably to the weight and cost of monitoring environmental conditions in harsh environments such as aerospace and automobile engine performance, chemical processing, and nuclear reactors.

Fault Tolerance and NAND Fault Trees

A generalized fault tree is an analytical technique used for evaluating reliability, safety, and maintainability of systems. These systems can be either processes or products. Factors affecting the outcome of a given system are inputs, and intermediate subsystem and overall system operation are the out-

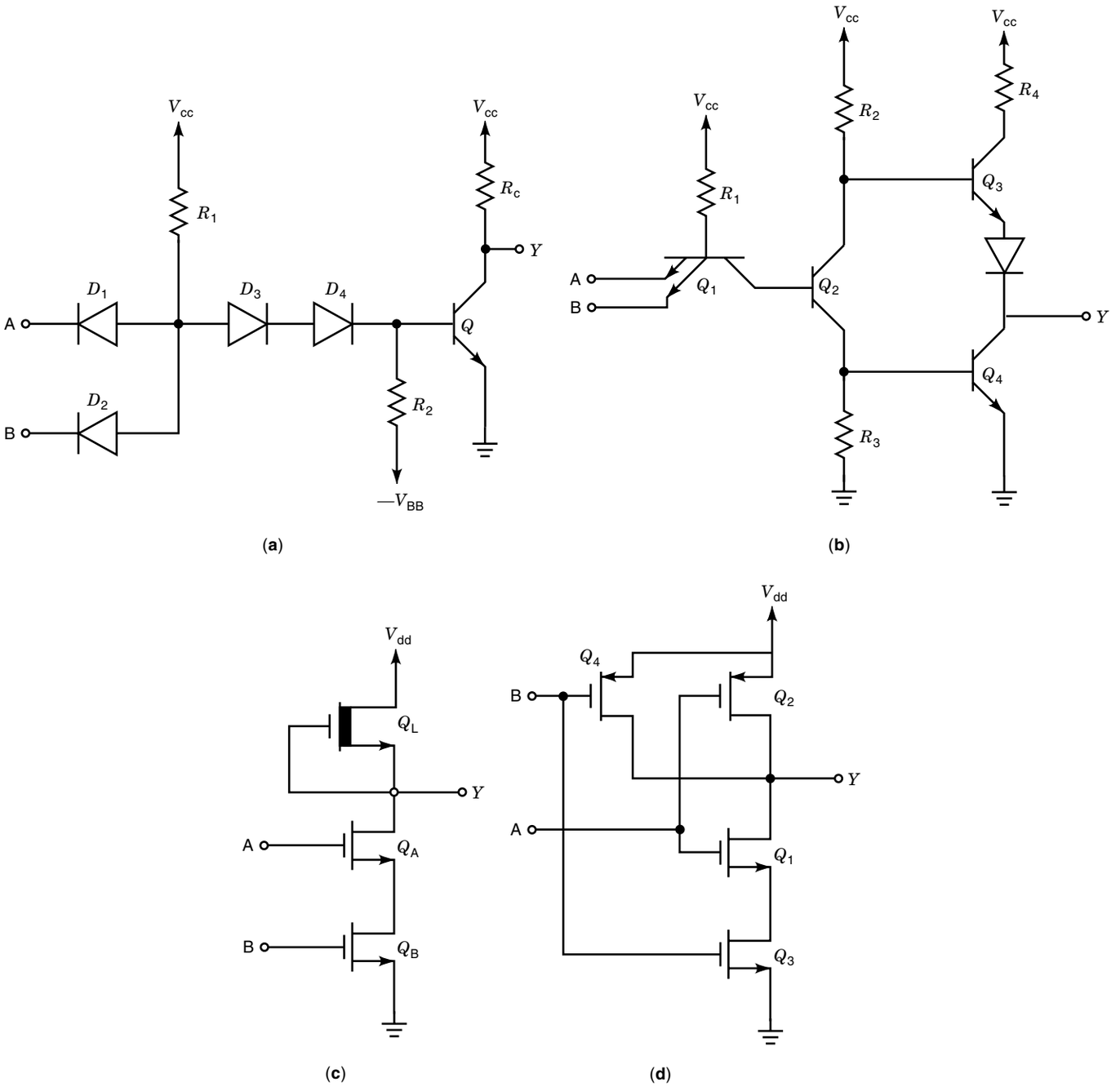

Figure 3. Actual NAND circuit implementations of (a) DTL, (b) TTL, (c) depletion load NMOS, and (d) CMOS. TTL and CMOS technologies are most commonly used to fabricate NAND gates (3).

puts. This method provides an easy way to define the complex relationship that exists between the hardware, software, and other components of a system. When all factors or parts are functioning properly, the system state is True. Fault-tolerant systems can provide operation, even if one or more of the factors is not functioning properly and this technique quite similar to the concept of pin-faults discussed earlier. When the system reaches a point where it can no longer be configured to operate safely or properly, the system state is False, and corrective action can be taken if necessary. In fault-tolerant systems, no single event or failure can be allowed to cause system failure or loss of control over the process. When additional redundancy is required, such as control operations in hazardous environments, system design can require tolerance of multiple faults or failures. Examples would include controlling a blast furnace used in the forging of steel, controlling and monitoring nuclear reactor operations, or operating complex combat systems. In these applications, damaged compo-

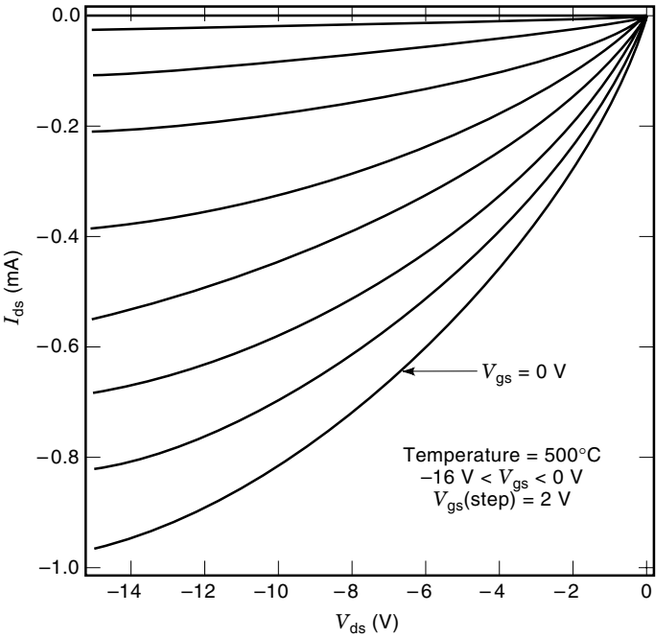

Figure 4. Drain-to-source current-voltage characteristics for depletion-mode metal-oxide semiconducting diamond field-effect transistor operating at 500°C. Transistors based on novel electronic materials may well be utilized in logic design for high temperature applications. Silicon-based transistors lose functionality above −200°C.

nents or operational variations could lead to disastrous consequences.

Both positive and negative factors are commonly used as inputs. NAND gates can be used to define contingencies in a system fault tree. As has been discussed, NAND circuits can be used to generate the other logical operators. In addition to the overall system response, the output of the various sections of the tree can be used to reconfigure the system and thus allow operation under conditions that would otherwise result in system failure.

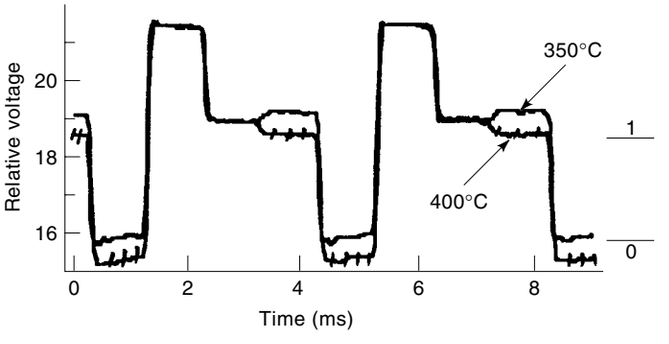

Figure 5. Operation at 350°C and 400°C of a NAND circuit employing depletion-mode metal-oxide-semiconducting diamond field-effect transitors. Simple logic structures are often utilized to demonstrate the utility of novel technologies for digital electronics.

CONCLUSIONS

Simple NAND circuits are in widespread use today in a variety of areas such as combinational logic and microprocessors. NAND circuits are finding increasing use in fault-trees, pin faults, and smart-sensor applications. NAND logic gates can be realized in multiple technologies, and the NAND circuit can be used to generate the other Boolean logic operations. NAND circuits are one of the easiest to realize in normally off transistor technologies. In the near future, applications operating in extreme environments will require both sensors and digital logic functionality for autonomous systems. With the development of new materials such as diamond and silicon carbide, which are nearing commercial viability, one of the first types of digital logic gates to be implemented will be most likely to be the NAND or NOR circuit.

BIBLIOGRAPHY

1. H. T. Nagle, Jr., B. D. Carroll, and J. D. Irwin, *An Introduction to Computer Logic.* Englewood Cliffs, NJ: Prentice-Hall, 1975.
2. H. Taub and D. Schilling, *Digital Integrated Circuits.* New York: McGraw-Hill, 1977.
3. A. S. Sedra and K. C. Smith, *Microelectronic Circuits,* 2nd ed. New York: Holt, Rinehardt and Winston, 1987.
4. P. Horowitz and W. Hill, *The Art of Electronics,* 2nd ed., New York: Cambridge Univ. Press, 1990.
5. M. N. Hornstein, *Microelectronic Circuits and Devices,* Englewood Cliffs, NJ: Prentice-Hall, 1990.
6. M. Abromovici, M. A. Breuer, and A. D. Friedman, *Digital Systems Testing and Testable Design.* New York: Computer Science, 1990.
7. D. L. Dreifus et al., High temperature operation of diamond field-effect transistors, *High Temperature Electronics Conf.,* Charlotte, NC, 1994.
8. J. S. Holmes and D. L. Dreifus, Field-effect transistors and circuits fabricated from semiconducting diamond thin films, *IEDM Conf.,* San Francisco, CA, 1994, pp. 423–426.
9. J. S. Holmes, A. J. Tessmer, and D. L. Dreifus, High temperature operation of diamond digital logic structures, *High Temperature Electronics Conf.,* Charlotte, NC, 1994.
10. R. R. Siergiej et al., Novel silicon carbide MOSFETs for monolithic integrated circuits, *High Temperature Electronics Conf.,* Charlotte, NC, 1994.

DAVID L. DREIFUS
JOSEPH S. HOLMES
Kobe Steel USA, Inc.

NANOCOMPUTING. See NONCONVENTIONAL COMPUTERS.

NANOSCALE MAGNETS. See MAGNETIC PARTICLES.