ELECTRONIC CALCULATORS

People have been making calculations with numbers for as long as there have been numbers. Many devices have been invented throughout history to make calculating easier. The abacus, which uses beads to keep track of numbers, was invented over 2000 years ago and is still used today. Blaise Pascal invented a "numerical wheel calculator," a brass box with dials for performing addition, in the 17th century (1). Gottfried Wilhelm von Leibniz soon created a version that could also multiply, but mechanical calculators were not widely used until the early 19th century, when Charles Xavier Thomas de Colmar invented a machine that could perform the four basic functions of addition, subtraction, multiplication, and division. Charles Babbage proposed a steam-powered calculating machine around 1822 that included many of the basic concepts of modern computers, but it was never built. A mechanical device that used punched cards to store data was invented in 1889 by Herman Hollerith and then used to compile the results of the U.S. census mechanically in only six weeks instead of ten years. A bulky mechanical calculator, with gears and shafts, was developed by Vannevar Bush in 1931 for solving differential equations (2).

The first electronic computers used technology based on vacuum tubes, resistors, and soldered joints, and thus were much too large for use in portable devices. The ENIAC (Electronic Numerical Integrator and Computer), completed in 1946, was one of the first general-purpose electronic computers. It was developed to compute artillery firing tables for World War II and could add, subtract, multiply, divide, and compute square roots. More than 17,000 vacuum tubes and 6,000 manual switches were used to build ENIAC, and it filled a large room. The invention of the transistor (replacing vacuum tubes) followed by the invention of the integrated circuit by Jack Kilby in 1958 led to the shrinking of electronic machinery until simple electronic computer functionality could be put into a package small enough to fit into a hand or a pocket.

Logarithms, developed by John Napier around 1600, can be used to solve multiplication and division problems with the simpler operations of addition and subtraction. Slide rules are mechanical, analog devices based on the idea of logarithms and use calibrated sticks or disks to perform multiplication and division to three or four significant figures. Slide rules were an indispensable tool for engineers until they were replaced by hand-held scientific calculators starting in the early 1970s (3).

CALCULATOR TYPES AND USES

Electronic calculators come in a variety of types: four-function (addition, subtraction, multiplication, division), desktop, printing, and scientific. Figure 1 shows various calculators with prices ranging from $3 to $265. Scientific calculators can calculate square roots, logarithms and exponents, and trigonometric functions. The scientific category includes business calculators, which have time-value-of-money, amortization, and other money management functions. Graphing calculators are a type of scientific calculator with a display that can show function plots. Advanced scientific and graphing calculators also have user programming capability that allows the user to enter and store programs. These programs can record and automate calculation steps, customize the calculator, or perform complicated or tedious algorithms. Some hand-held

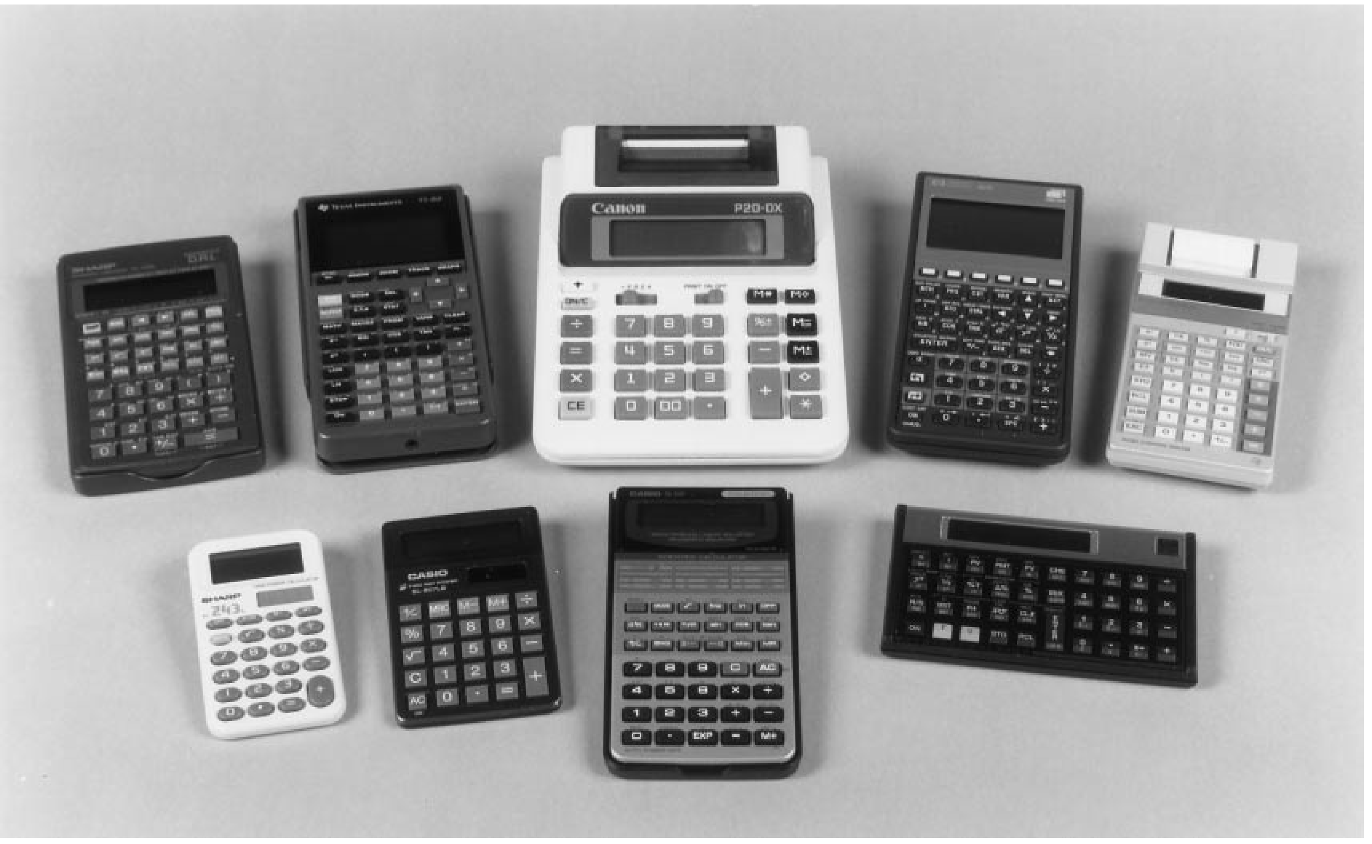

Figure 1. Various calculators with prices ranging from $3 to $265.

calculators are solar powered but most advanced scientific calculators are powered by batteries that last for many months without needing replacement.

People use electronic calculators for a variety of applications, from simple arithmetic operations for balancing a checkbook to complicated programs for collecting temperature samples from remote weather stations. Calculators are used in business and technical applications and they are used by students and professionals. Engineers and scientists use calculators in their work to check on initial results, convert measurements, and analyze data. Many banking, finance, and real estate professionals keep a financial calculator handy to calculate compound interest, loan and mortgage payments, and depreciation. Students use calculators for their mathematics and science homework, and more and more teachers are incorporating graphing calculators into their lessons. Calculators are very useful for quick, back-of-the-envelope types of problems. Even with the increasing use of computers in the workplace and in homes, people continue to buy and use calculators because they are handy, convenient, portable, and dedicated to performing a few mathematical functions well.

Scientific Calculators

Scientific calculators can perform trigonometric functions and inverse trigonometric functions (sin x, cos x, tan x, arcsin x, arccos x, arctan x) as well as hyperbolic and inverse hyperbolic functions (sinh x, cosh x, tanh x, arcsinh x, arccosh x, arctanh x). They can also find natural and common logarithms (ln x, log x), exponential functions (e^x, y^x, $\sqrt[x]{y}$), factorials ($n!$), and reciprocals ($1/x$). Scientific calculators contain a representation for the constant π, and they can convert angles between degrees and radians. Most scientific calculators accept numbers with 10 to 12 digits and exponents ranging from -99 to 99, although some allow exponents from -499 to 499.

Graphing Calculators

Graphing calculators were first developed in the late 1980s as larger liquid-crystal displays (LCDs) became available at lower cost. The pixels in an LCD display can be darkened individually and so can be used to plot function graphs. The user keys in a real-valued function of the form $y = f(x)$ and makes some choices about the scale to use for the plot and the set of values for x. Then the calculator evaluates $f(x)$ for each x value specified and displays the resulting $(x, f(x))$ pairs as a function graph. Graphing calculators can also plot polar and parametric functions, 3-D wireframe plots, differential equations, and statistics graphs such as scatter plots, histograms, and box-and-whisker plots (see Fig. 2). Once a graph has been displayed, the user can move a small cursor or crosshairs around the display by pressing the arrow or cursor keys and then obtain information about the graph, such as the coordinates of points, the x-intercepts, or the slope of the graph at a certain point. The user can also select an area of interest to zoom in on, and the calculator will re-plot the graph using a different scale (4).

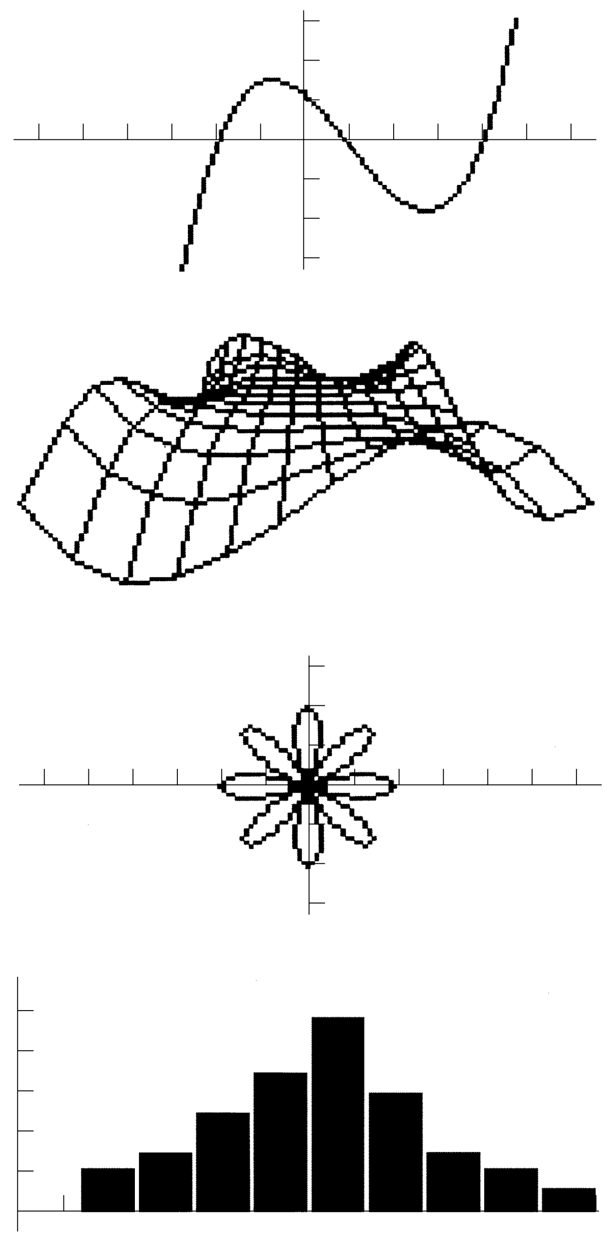

Figure 2. Graphing calculator plots (top to bottom): polynomial function $f(x) = x^3 - 3x^2 - 6x + 8$; wireframe plot of $z = x^3y - xy^3$; polar plot of $r = 2\cos(4\theta)$; histogram.

Programmable Calculators

If a series of steps is to be repeated using various inputs, it is convenient to be able to record those steps and replay them automatically. Simple programmable calculators allow the user to store a sequence of keystrokes as a program. More complicated programmable calculators provide programming languages with many of the components of high-level computer languages, such as branching and subroutines.

Given all these types and uses of calculators, what is it that defines a calculator? The basic paradigm of a calculator is: key per function. For example, one key is dedicated to the square root function on most scientific calculators. All the user has to do is input a number, then press one key, and the calculator performs a complicated series of steps to obtain an answer that users could not easily calculate on their own. Another way to say this is that there is an asymmetry of information flow: given a small amount of input, the calculator does something nontrivial and gives you back results that you

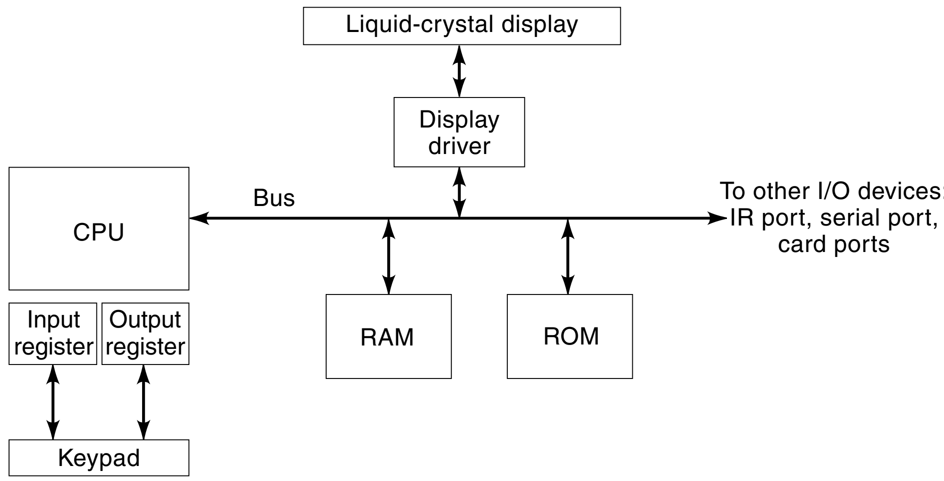

Figure 3. Block diagram of the system architecture of an advanced scientific graphing calculator.

could not have found easily in your head or with pencil and paper.

CALCULATOR HARDWARE COMPONENTS

Today's advanced scientific and graphing calculators have many similarities to computers. The block diagram in Figure 3 shows the system architecture of an advanced scientific graphing calculator (5). The two main components of a calculator are hardware and software. The hardware includes plastic and metal packaging, display, keypad, optional additional input/output devices (such as infrared, serial ports, card slots, and beeper parts to produce sound), power supply circuit, and an electronic subsystem. The electronic subsystem consists of a printed circuit board with attached electronic components and integrated circuits, including a central processing unit (CPU), display controllers, random-access memory (RAM), and the read-only memory (ROM) where the software programs are stored permanently.

The mechanical design of a calculator consists of subassemblies such as a top case with display and keypad, a bottom case, and a printed circuit or logic assembly. Figure 4 shows the subassemblies of a graphing calculator. A metal chassis in the top case supports the keypad, protects and frames the glass display, and provides a negative battery contact. The metal chassis is also part of the shielding that protects the electronic circuitry from electrostatic discharge (ESD). The bottom case may contain additional metal shielding, a piezoelectric beeper part, and circuitry for battery power. The subassemblies are connected electrically with flexible circuits (6).

Display

Early calculators used light-emitting diode (LED) displays, but liquid-crystal displays (LCDs) are used in most modern calculators because they have low voltage requirements, good visibility in high ambient light conditions, and they can produce a variety of character shapes and sizes (7). An LCD consists of two pieces of glass with a layer of liquid crystal in between that will darken in specific areas when a voltage signal is applied. These areas can be either relatively large segments that are combined a few at a time to represent a number or character, or a grid of very small rectangles (also called picture elements, or pixels) that can be darkened selectively to produce characters, numbers, and more detailed graphics. Basic calculators have one-line displays that show one row of numbers at a time, while today's more advanced calculators can display up to eight or more rows of characters with 22 or more characters per row, using a display with as many as 64 rows and 131 columns of pixels.

Keypad

Calculator keypads are made up of the keys the user presses, an underlying mechanism that allows the keys to be depressed and then to return to their initial state, and circuit traces that allow the system to detect a key press. When a key is pressed, an input register line and an output register line make contact, which causes an interrupt to be generated. This interrupt is a signal to the software to scan the keyboard

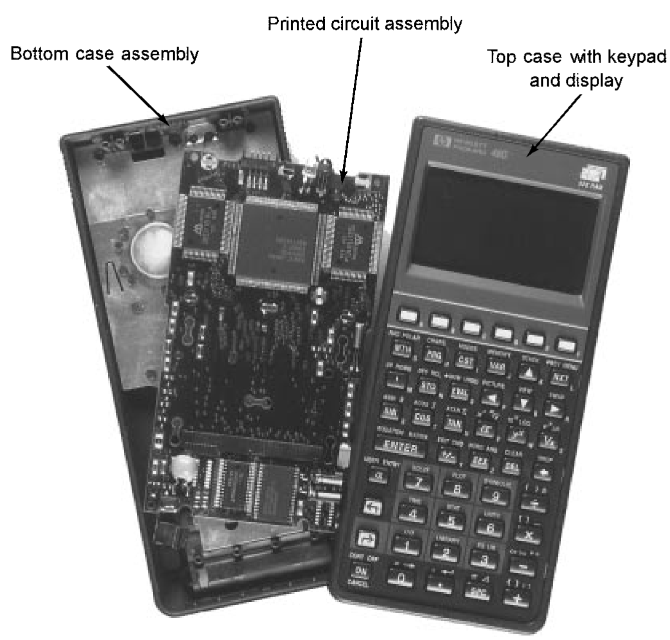

Figure 4. Graphing calculator subassemblies (bottom case, printed circuit assembly, and top case with keypad and display).

Figure 5. Hinged plastic keys provide tactile feedback to the user.

to see which key is pressed. Keypads have different tactile feel, depending on how they are designed and what materials they are made of. Hinged plastic keys (shown in Fig. 5) and dome-shaped underlying pads are used to provide feedback to the user with a snap when keys are pressed. An elastomer membrane separating the keys from the underlying contacts helps to protect the electronic system from dust (8).

The keypad is an input device, since it is a way for the user to provide information to the calculator. The display is an output device, since it allows the calculator to convey information to the user. Early calculators, and today's simple calculators, make do with only these input and output devices. But as more and more memory has been added to calculators, allowing for storage of more data and more extensive programs, the keypad and display have become bottlenecks. Various means have been developed to alleviate these bottlenecks. Small magnetic cards have been used, as well as RAM cards that can be plugged into a calculator. Infrared and serial cable ports allow some calculators to communicate with computers and with other calculators to transfer data and programs quickly and easily.

Circuits

The electronic components of a calculator form a circuit that includes small connecting wires that allow electric current to flow to all parts of the system. The system is made up of diodes, transistors, and passive components such as resistors, capacitors, and inductors, as well as conventional circuits and integrated circuits designed to perform certain tasks. One of these specialized circuits is an oscillator, which serves as a clock and is used to control the movement of bits of information through the system. Another type of specialized circuit is a logic circuit, or processor, which stores data in registers and performs manipulations such as addition.

Printed Circuit Assembly

A printed circuit board (PCB) forms the backbone of a calculator's electronic circuit system, allowing various components to be attached and connected to each other (9). Figure 6 shows a calculator printed circuit assembly with many of the electronic components labeled. Wires that transmit data and instructions between the logic circuits, the memory circuits, and the other components are called buses. The printed circuit assembly starts out as a piece of high-temperature laminate polyimide or modified polyimide. This material is used because it accepts high-speed gold thermosonic bonding. Copper tracings, which will form the circuits that connect the electronic components, are printed on the circuit board and are plated with nickel and gold to keep the copper from being exposed. High-purity, silver-filled epoxy is used to attach some components to the PCB because of its thermal and electrical conductivity. The bond wires of the integrated circuits are attached to the board traces with epoxy. Various integrated circuits may be used in a calculator, including a CPU, RAM and ROM circuits, memory controllers that allow the CPU to access the RAM and ROM, a controller for the display, quartz-crystal-controlled clocks, and controllers for optional additional input/output devices such as serial cable connectors and infrared transmitters and receivers. Depending on the design of the calculator and the choice of components, some of these pieces may be incorporated in a single integrated circuit called an application-specific integrated circuit (ASIC).

Central Processing Unit

The central processing unit of a calculator or computer is a complicated integrated circuit consisting of three parts: the arithmetic logic unit (ALU), the control unit, and the main storage unit. The ALU carries out the arithmetic operations of addition, subtraction, multiplication, and division and makes logical comparisons of numbers. The control unit receives program instructions and then sends control signals to different parts of the system; it can also jump to a different part of a program under special circumstances such as an arithmetic overflow. The main storage unit stores data and instructions used by the ALU or control unit.

Many calculators use custom microprocessors because commercially available microprocessors designed for larger computers do not take into account the requirements of a small, hand-held device. Calculator microprocessors must operate well under low power conditions, should not require too many support chips, and generally must work with a smaller system bus. This is because wider buses use more power and require additional integrated circuit (IC) pins, which increases part costs. Complementary metal-oxide semiconductor, or CMOS, technology is used for many calculator integrated circuits because it is well suited to very low power systems (10). CMOS has very low power dissipation and can retain data even with drastically reduced operating voltage. CMOS is also highly reliable and has good latch-up and electrostatic discharge (ESD) protection.

Memory

Random-access memory integrated circuits are made up of capacitors, which represent bits of information. Each bit may be

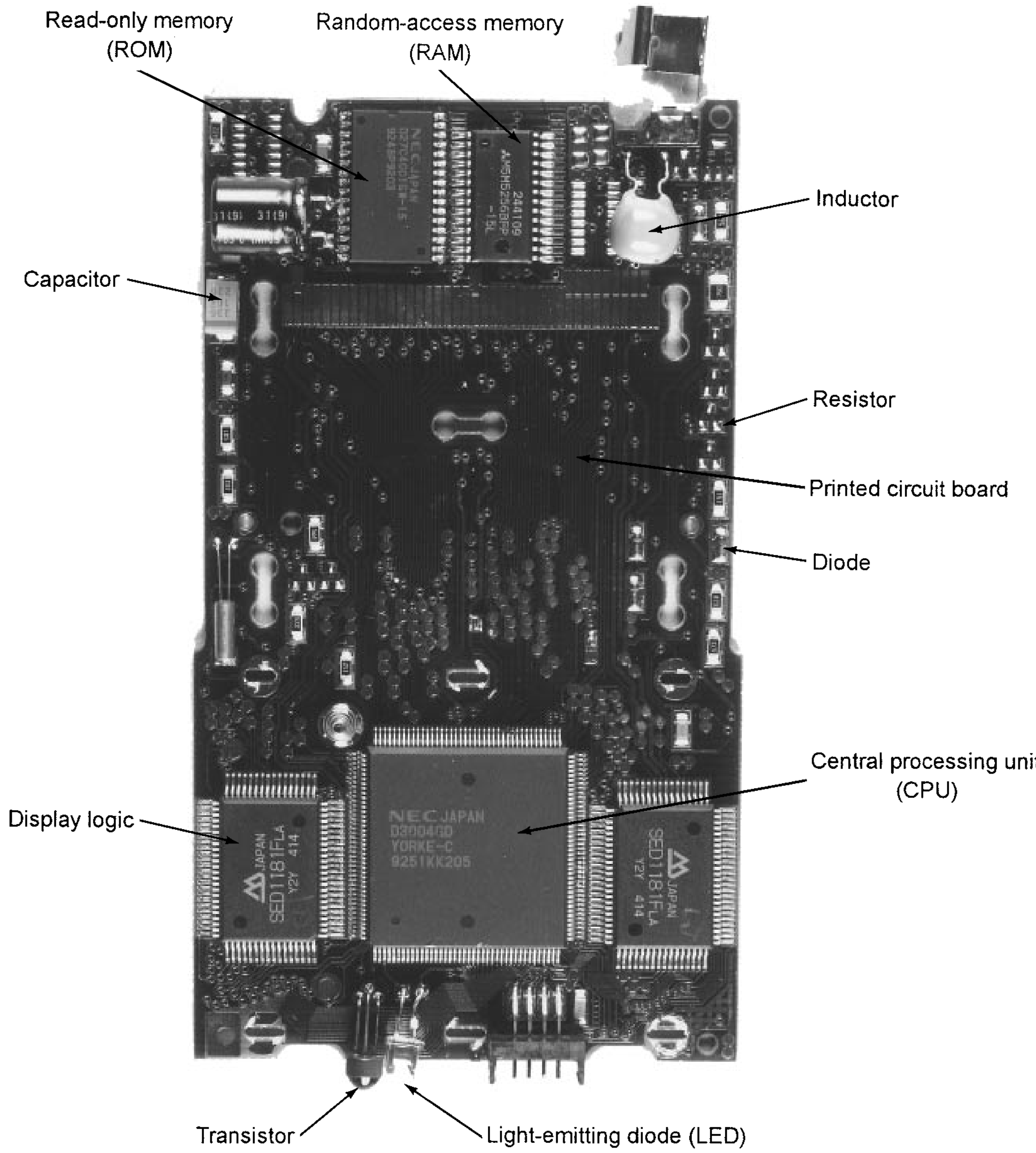

Figure 6. Electronic components are attached to a printed circuit board to form a printed circuit assembly.

in one of two possible states, depending on whether the capacitor is holding an electric charge or not. Any bit of information in RAM can be changed easily, but the information is only retained as long as power is supplied to the integrated circuit. In continuous memory calculators, the information in RAM is retained even when the calculator is turned off, because a small amount of power is still supplied by the system. The RAM circuits used in calculators have very low standby current requirements, and can retain their information for short periods of time without power, such as when batteries are being replaced.

Read-only memory circuits contain information that cannot be changed once it is encoded. Calculator software is stored in ROM because it costs less and has lower power requirements than RAM. Since software is encoded on ROM by the manufacturer and cannot be changed, the built-in software in calculators is often called firmware. When calculator firmware is operating, it must make use of some amount of RAM whenever values need to be recorded in memory. The RAM serves as a scratch pad for keeping track of inputs from the user, results of intermediate calculations, and final results. The firmware in most advanced scientific calculators consists of an operating system (which is a control center for coordinating the low-level input and output, memory access, and other system functions), user interface code, and the mathematical functions and other applications that the user is directly interested in.

OPERATING SYSTEM

The operation of a calculator can be broken down into three basic steps of input, processing, and output. For example, to find the square root of a number, the user enters a number on the keypad and chooses the function to be computed. This input generates electronic signals that are processed by the calculator's electronic circuits to produce a result. The result is then communicated to the user via the display. The processing step involves storing data using memory circuits and making changes to that data using logic circuits, as well as the general operation of the system, accomplished using control circuits.

A calculator performs many tasks at the system level that the user is not normally aware of. These include turning the calculator on or off, keeping track of how memory is being used, managing the power system, and all the overhead associated with getting input from the keypad, performing calculations, and displaying results. A calculator's operations are controlled by an operating system, which is a software program that provides access to the hardware computing resources and allows various application software programs to be run on the computer (or calculator).

The operating system deals with memory organization, data structures, and resource allocation. The resources it controls include CPU time, memory space, and input/output devices such as the keypad and display. The operating system is responsible for running an application program by scheduling slices of CPU time that can be used for executing the program steps, and also for overseeing handling of any interrupts that may occur while the program is executing. Interrupts are triggered by events that need to be dealt with in a timely fashion, such as key presses, requests from a program for a system-level service such as refreshing the display, or program errors. Some errors that may occur when a program is running are low power conditions, low memory conditions, arithmetic overflow, and illegal memory references. All of these conditions should be handled gracefully, with appropriate information given to the user. Operating systems provide convenience and efficiency: they make it convenient to execute application programs, and they manage system resources to get efficient performance from the computer or calculator (11).

USER INTERFACE

The user interface for one-line-display calculators is very simple, consisting of a single number shown in the display. The user may have some choice about the format of that number, such as how many digits to display to the right of the decimal point, or if the number should be shown using scientific notation. Error messages can be shown by spelling out short words in the display. Calculators more complicated than the simple four-function ones may not have enough keys on the keypad to use one for every operation the calculator can perform. Then it becomes necessary to provide a more extensive user interface than just a simple keypad. One way to increase the number of operations that the keypad can control is to add shifted keys. For example, one key may have the symbol $\sqrt{x}$ on the key, and the symbol x^2 printed just above the key, usually in a second color. If the user presses the $\sqrt{x}$ key, the square-root function is performed. But if the user first presses the Shift key and then the $\sqrt{x}$ key, the x-squared function will be performed instead.

Advanced scientific and graphing calculators provide systems of menus that let the user select operations. These menus may appear as lists of items in the display, which the user can scroll through using arrow or cursor keys and then select by pressing the Enter key. Changeable labels in the bottom portion of the display, which correspond to the top row of keys, can also be used to display menu choices. These are called soft keys, and they are much like the function keys on a computer. Methods for the user to enter information into the calculator depend on the type of calculator. On simple, one-line-display calculators, the user presses number keys and can see the corresponding number in the display. Graphing calculators, with their larger displays, can prompt the user for input and then display the input using dialog boxes like the ones used on computers (12). Figure 7 shows a graphing calculator dialog box used to specify the plot scale.

NUMBERS AND ARITHMETIC

The most basic level of functionality apparent to the calculator user is the arithmetic functions: addition, subtraction, multiplication, and division. All calculators perform these functions, and some calculators are limited to these four functions. Calculators perform arithmetic using the same types of circuits that computers use. Special circuits based on Boolean logic are used to combine numbers, deal with carries and overdrafts, and find sums and differences. Various methods have been developed to perform efficient multiplication and division with electronic circuits (13).

Binary Numbers

Circuits can be used to represent zeros or ones because they can take on two different states (such as on or off). Calculator (and computer) memory can be thought of as simply a large collection of zeros and ones. Zeros and ones also make up the binary, or base two, number system. For example, the (base ten) numbers 1, 2, 3, 4 are written in base two as 1, 10, 11, 100, respectively. Each memory circuit that can be used to represent a zero or one is called a binary digit, or bit. A collection of eight bits is called a byte (or a word). Some calculator systems deal with four bits at a time, called nibbles. If simple binary numbers were used to represent all numbers that could possibly be entered into a calculator, many bits of memory would be needed to represent large numbers. For example, the decimal number 2^n is represented by the binary number consisting of a 1 followed by n zeros, and so requires $n + 1$ bits of memory storage. To be able to represent very large numbers with a fixed number of bits, and also to optimize arithmetic operations for the design of the calculator, floating-point numbers are used in calculators and computers.

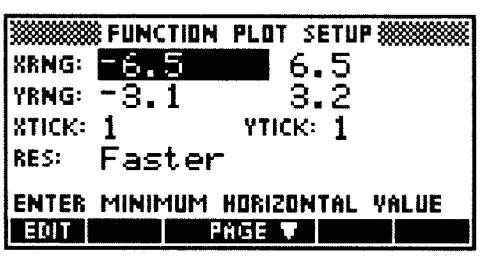

Figure 7. A dialog box on a graphing calculator is used to specify the plot scale.

Floating-Point Numbers

Floating-point numbers are numbers in which the location of the decimal point may move so that only a limited number of digits are required to represent large or small numbers. This eliminates leading or trailing zeros, but its main advantage for calculators and computers is that it greatly increases the range of numbers that can be represented using a fixed number of bits. For example, a number x may be represented as $x = (-1)^s \times F \times b^E$, where s is the sign, F is the significand or fraction, b is the base used in the floating-point hardware, and E is a signed exponent. A fixed number of bits are then used to represent each number inside the calculator. The registers in a CPU designed for efficient floating-point operations have three fields that correspond to the sign, significand, and exponent and can be manipulated separately.

Two types of errors can appear when a calculator returns an answer. One type is avoidable, and is caused by inadequate algorithms. The other type is unavoidable, and is the result of using finite approximations for infinite objects. For example, the infinitely repeating decimal representation for 2/3 is displayed as .6666666667 on a ten-decimal-place calculator. A system called binary-coded decimal (BCD) is used on some calculators and computers as a way to deal with rounding. Each decimal digit, 0, 1, 2, 3, . . ., 9 is represented by its four-bit binary equivalent: 0000, 0001, 0010, 0011, . . ., 1001. So rather than convert each base-ten number into the equivalent base-two number, the individual digits of the base-ten number are each represented with zeros and ones. When arithmetic is performed using BCD numbers, the methods for carrying and rounding follow base-ten conventions.

One way to improve results that are subject to rounding errors is to use extra digits for keeping track of intermediate results, and then do one rounding before the result is returned using the smaller number of digits that the user sees. For example, some advanced scientific calculators allow the user to input numbers using up to twelve decimal places, and return results in this same format, but fifteen-digit numbers are actually used during calculation.

Reverse Polish Notation and Algebraic Logic System

The Polish logician Jan Lukasiewicz demonstrated a way of writing mathematical expressions unambiguously without using parentheses in 1951. For example, given the expression $(2 + 3) \times (7 - 1)$, each operator can be written before the corresponding operands: $\times + 2\ 3 - 7\ 1$. Or, each operator can be written after its operands: $2\ 3\ +\ 7\ 1\ -\ \times$. The latter method has come to be known as Reverse Polish Notation, or RPN (14). Arithmetic expressions are converted to RPN before they are processed by computers because RPN simplifies the evaluation of expressions. In a non-RPN expression containing parentheses, some operators cannot be applied until after parenthesized subexpressions are first evaluated. Reading from left to right in an RPN expression, every time an operator is encountered it can be applied immediately. This means there is less memory and bookkeeping required to evaluate RPN expressions. Some calculators allow users to input expressions using RPN. This saves the calculator the step of converting the expression to RPN before processing it. It also means fewer keystrokes for the user since parentheses are never needed with RPN. Algebraic logic system (ALS) calculators require numbers and operators to be entered in the order they would appear in an algebraic expression. Parentheses are used to delimit subexpressions in ALS. An RPN calculator does not need to have an = key, but uses an Enter key to separate operands. To calculate the previous expression on an RPN calculator requires nine keystrokes: 2 Enter 3 + 7 Enter 1 − ×. The same expression would be keyed into an ALS calculator using twelve keystrokes: $(2 + 3) \times (7 - 1)$ =. The number of keystrokes required is not the only difference between the two input methods: RPN calculators display intermediate results (such as 2 + 3 in the previous example) and also allow the results of one calculation to be used in subsequent calculations without having to be keyed in.

User Memory

On many calculators, the user can store numbers in special memory locations or storage registers, and then perform arithmetic operations on the stored values. This process is called register arithmetic. On RPN calculators, memory locations are arranged in a structure called a stack. For each operation that is performed, the operands are taken from the stack and then the result is returned to the stack. Each time a new number is placed on the stack, the previous items that were on the stack are each advanced one level to make room for the new item. Whenever an item is removed from the stack, the remaining items shift back. A stack is a data structure similar to a stack of cafeteria trays, where clean trays are added to the top and as trays are needed, they are removed from the top of the stack. This scheme for placing and removing items is called last-in-first-out or LIFO.

ALGORITHMS

An algorithm is a precise, finite set of steps that describes a method for a computer (or calculator) to solve a particular problem. Many computer algorithms are designed with knowledge of the underlying hardware resources in mind, so that they can optimize the performance of the computer. Numerical algorithms for calculators take into account the way that numbers are represented in the calculator.

Square Root Algorithm

A simple approximation method is used by calculators to find square roots. The basic steps to finding $y = \sqrt{x}$ are first to guess the value of y, calculate y^2, and then find $r = x - y^2$. Then if the magnitude of r is small enough, return y as the answer. Otherwise, increase or decrease y (depending on whether r is positive or negative) and repeat the process. The number of intermediate calculations required can be reduced by avoiding finding y^2 and $x - y^2$ for each value of y. This can be done by first finding the value of the largest-place digit of y, then the next largest-place digit, and so on. For example, if calculating $\sqrt{54756}$, first find 200, then 30, and 4 to construct the answer y = 234. This method is similar to a method once taught in schools for finding square roots by hand (15).

Trigonometric Function Algorithms

The algorithms for computing trigonometric functions depend on using trigonometric identities and relationships to reduce arbitrarily difficult problems to more manageable problems. First, the input angle θ is converted to an angle in radians that is between 0 and 2π (or in some calculators, between 0 and $\pi/4$). Next θ is expressed as a sum of smaller angles. These smaller angles are chosen to be angles whose tangents are powers of ten: $\tan^{-1}(1) = 45°$, $\tan^{-1}(0.1)$, $\tan^{-1}(0.01)$, . . . and so on. A process called pseudo-division is used to express θ in this way: first $\tan^{-1}(1)$ is repeatedly subtracted from θ until an overdraft (or carry) occurs, then the angle being subtracted from is restored to the value it had right before the overdraft occurred, then the process is repeated by subtracting $\tan^{-1}(0.1)$ until an overdraft occurs, and so forth, until we are left with a remaining angle r that is small enough for the required level of accuracy of the calculator. Then θ can be expressed as:

$$\theta = q_0 \tan^{-1}(1) + q_1 \tan^{-1}(0.1) + \cdots + r \tag{1}$$

Vector geometry is the basis for the formulas used to compute the tangent of θ once it has been broken up into the sum of smaller angles. Starting with a vector with angle θ_1 then rotating it counter-clockwise by an additional angle of θ_2, Figure 8 illustrates the following relationships:

$$X_2 = X_1 \cos\theta_2 - Y_1 \sin\theta_2$$
$$Y_2 = Y_1 \cos\theta_2 + X_1 \sin\theta_2$$

Dividing both sides of these equations by $\cos\,\theta_2$ we obtain:

$$X_2/\cos\theta_2 = X_1 - Y_1 \tan\theta_2 = X_2' \tag{2}$$

$$Y_2/\cos\theta_2 = Y_1 + X_1 \tan\theta_2 = Y_2' \tag{3}$$

Since $Y_2/X_2 = \tan(\theta_1 + \theta_2)$, then by Eq. (2) and Eq. (3), we can see that $Y_2'/X_2' = \tan(\theta_1 + \theta_2)$. Eq. (2) and Eq. (3) can be used repeatedly to construct the tangent of θ, since θ has been broken down into a series of smaller angles, shown in Eq. (1). The initial X_1 and Y_1 correspond to the small residual angle r. Since r is a very small angle (in radians) $\sin(r)$ is close to r and $\cos(r)$ is close to 1, so if these values are close enough for our overall accuracy requirements, we can let Y_1 be r and X_1 be 1. Note Eq. (2) and Eq. (3) involve finding tangents, but since we expressed θ as a sum of angles of the form $\tan^{-1}(10^{-k})$, $\tan[\tan^{-1}(10^{-k})] = 10^{-k}$ so each evaluation of Eq. (2) or Eq. (3) will simply involve addition, subtraction, and multiplication by powers of 10. Since the only multiplication involved is by powers of ten, the calculations can be accomplished more quickly and simply using a process called pseudo-multiplication which involves only addition and the shifting of contents of registers to simulate decimal point shifts that correspond to multiplication by powers of ten. The iterative process of using Eq. (2) and Eq. (3) generates an X and Y proportional to the sine and cosine of the original angle θ. Then elementary operations can be used to find the values of the various trigonometric functions for θ (16).

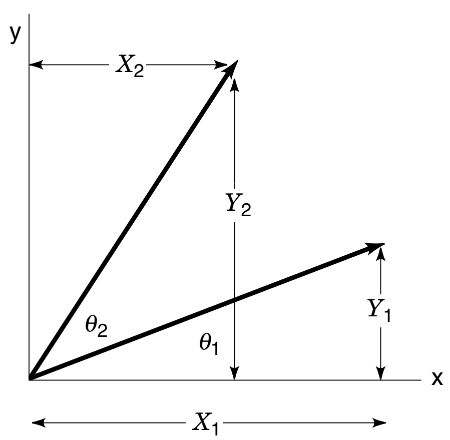

Figure 8. Algorithm for finding trigonometric functions depends on expressing an angle as a sum of smaller angles.

Logarithm Algorithms

Logarithms are found using a process similar to the approximation process used to compute trigonometric functions (17). It is a basic property of logarithms that $\ln(a_1 \times a_2 \times . . . \times a_n) = \ln(a_1) + \ln(a_2) + . . . + \ln(a_n)$. To find the logarithm of a number x, x is first expressed as the product of factors whose logarithms are known. The number x will be stored in the calculator using scientific notation $x = M \times 10^k$, where M is called the mantissa and M is ≥ 1 and <10. Since $\ln(M \times 10^k) = \ln(M) + k \times \ln(10)$, the problem of finding $\ln(x)$ is reduced to the problem of finding $\ln(M)$. Let a_j be numbers whose natural logarithms are known. Let $P = 1/M$. Then $-\ln(P) = \ln(M)$. Then express P as $P = P_n/r$ where $P_n = \mathbf{a}_0^{k0} \times \mathbf{a}_1^{k1} \times . . . \times \mathbf{a}_j^{kj}$ and r is a number close to 1. Note that $\ln(P) = \ln(P_n) - \ln(r)$, so now $\ln(M) = \ln(r) - \ln(P_n)$ and for r close to 1, $\ln(r)$ is close to 0. Also note that $M = 1/P = r/P_n$ implies that $M \times P_n = r$. So to find $\ln(M)$, we can first find P_n such that $M \times P_n$ is close to 1, where P_n is a product of specially chosen numbers a_j whose logarithms are known. To optimize this routine for a calculator's specialized microprocessor, values that give good results are $a_j = (1 + 10^{-j})$. Thus, for example, a_0, a_1, a_2, a_3, and a_4 would be 2, 1.1, 1.01, 1.001, and 1.0001. It turns out that M must first be divided by 10 in order to use these a_j choices. This choice of the a_j terms allows intermediate multiplications by each a_j to be accomplished by an efficient, simple shift of the digits in a register, similar to the pseudo-multiplication used in the trigonometric algorithm.

CALCULATOR DESIGN CHOICES AND CHALLENGES

The requirements for a hand-held calculator to be small, portable, inexpensive, and dedicated to performing computational tasks have driven many design choices. Custom ICs and the CMOS process have been used because of low power requirements. Calculator software has been designed to use mostly ROM and very little RAM because of part cost and power constraints. Specialized algorithms have been developed and refined to be optimized for calculator CPUs. As calculators become more complicated, ease-of-use becomes an important design challenge. As memory becomes less expensive and calculators have more storage space, the keypad and display become bottlenecks in the transfer of large amounts of data. Improved input/output devices such as pen input, better displays, and character and voice recognition could help to alleviate bottlenecks and make calculators easier to use.

A desktop personal computer (PC) does not fit the needs of personal portability, and is not very convenient to use as a calculator for quick calculations. Also, a PC is a generic platform rather than a dedicated appliance. The user must take the time to start up an application to perform calculations on a PC, so a PC does not have the back-of-the-envelope type of immediacy of a calculator. Hand-held PCs and palm-top PCs also tend to be generic platforms, only in smaller packages. So they are as portable as calculators, but they still do not have dedicated calculating functionality. Users must go out of their way to select and run a calculator application on a hand-held PC. The keypad of a hand-held PC has a QWERTY keyboard layout, and so does not have keys dedicated to calculator functions like sine, cosine, and logarithms. Hand-held organizers and personal digital assistants (PDAs) are closer to the calculator model, because they are personal, portable, battery-operated electronic devices dedicated to particular functionality, but they currently emphasize organizer functionality rather than mathematics functionality.

COMMUNICATION CAPABILITY

Some calculators have already had communication capability for many years, using infrared as well as serial cable and other types of cable ports. These have allowed calculators to communicate with other calculators, computers, printers, overhead display devices that allow an image of the calculator screen to be enlarged and projected for a roomful of people, data collection devices, bar code readers, external memory storage, and other peripheral devices. Protocols are standard formats for the exchange of electronic data that allow different types of devices to communicate with each other. For example, Kermit is a file transfer protocol developed at Columbia University. When this protocol is coded into a calculator, the calculator is able to communicate with a number of different computers by running a Kermit program on the computer.

TECHNOLOGY IN EDUCATION

Curriculum materials have changed with the increased use of graphing calculators in mathematics and science classrooms. Many pre-calculus and calculus textbooks and science workbooks now contain exercises that incorporate the use of calculators. This allows exercises more complicated than the types of problems easily solved with pencil and paper in a few minutes. With the use of calculators, more realistic, and thus more interesting and extensive, problems can be used to teach mathematics and science concepts. Calculators are becoming a requirement in many mathematics classes and on some standardized tests, such as the Scholastic Aptitude Test taken by most U.S. high school seniors who plan to attend college. Educational policy has, in turn, influenced the design of graphing calculators. In the U.S., the National Council of Teachers of Mathematics promotes the use of the symbolic, graphic, and numeric views for teaching mathematics. These views are reflected in the design of graphing calculators with keys dedicated to entering a symbolic expression, graphing it, and showing a table of function values. Figure 9 shows a graphing calculator display of the symbolic, graphic, and numeric views of $\sin(x)$ (18).

FUTURE NEED FOR CALCULATORS

Technical students and professionals will always need to do some back-of-the-envelope calculations quickly and conveniently. The key-per-function model of a calculator fits in nicely with this need. So does a device that is dedicated, personal, portable, low-cost, and has long battery life. Users' expectations will be influenced by improvements in computer speed and memory size. Also, video game users have higher expectations for interactivity, better controls, color, animation, quick responses, good graphic design, and visual quality. For the future, calculators can take advantage of advances in computer technology and the decreasing cost of electronic components to move to modern platforms that have the benefits of increased speed, more memory, better displays, color displays, more versatile input devices (such as pen and voice), and more extensive communication capability. With appropriate protocols, calculators could work with modems and gain access to the Internet. Or, calculators could be used as part of a network of computers and calculators in a classroom. Calculations could then be performed remotely on more powerful computers and answers sent back to the calculator. Calculators could also be used to receive lessons distributed over the Internet. Although these calculators would do much more than the simple four-function calculators, they are still consistent with the idea of a calculator as a personal, portable, specialized device, which is a handy tool for performing complicated functions quickly and easily.

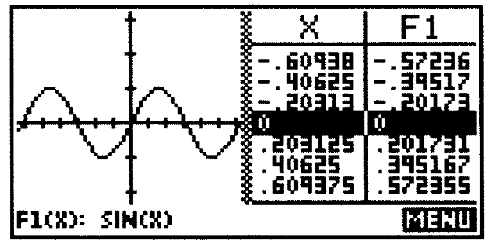

Figure 9. Graphing calculator display of the symbolic, graphic, and numeric views of $\sin(x)$.

BIBLIOGRAPHY

1. A. Ralston and E. D. Reilly, Jr. (eds.), *Encyclopedia of Computer Science and Engineering,* 2nd ed., New York: Van Nostrand Reinhold, 1983.
2. *Jones Telecommunications and Multimedia Encyclopedia,* Jones Digital Century, http://www.digitalcentury.com.
3. G. C. Beakley and R. E. Lovell, *Computation, Calculators, and Computers,* New York: Macmillan, 1983.
4. T. W. Beers, D. K. Byrne, G. L. Eisenstein, R. W. Jones, and P. J. Megowan, HP 48SX interfaces and applications, *Hewlett-Packard Journal,* **42** (3): 13–21, June 1991.
5. P. D. Brown, G. J. May, and M. Shyam, Electronic design of an advanced technical handheld calculator, *Hewlett-Packard Journal,* **38** (8): 34–39, August 1987.
6. M. A. Smith, L. S. Moore, P. D. Brown, J. P. Dickie, D. L. Smith, T. B. Lindberg, and M. J. Muranami, Hardware design of the HP 48SX scientific expandable calculator, *Hewlett-Packard Journal,* **42** (3): 25–34, June 1991.
7. C. Maze, The first HP liquid crystal display, *Hewlett-Packard Journal,* **31** (3): 22–24, March 1980.
8. T. Lindberg, Packaging the HP-71B handheld computer, *Hewlett-Packard Journal,* **35** (7): 17–20, July 1984.

9. B. R. Hauge, R. E. Dunlap, C. D. Hoekstra, C. N. Kwee, and P. R. Van Loan, A multichip hybrid printed circuit board for advanced handheld calculators, *Hewlett-Packard Journal,* **38** (8): 25–30, August 1987.
10. D. E. Hackleman, N. L. Johnson, C. S. Lage, J. J. Vietor, and R. L. Tillman, CMOSC: low-power technology for personal computers, *Hewlett-Packard Journal,* **34** (1): 23–28, January 1983.
11. J. L. Peterson and A. Silberschatz, *Operating System Concepts,* 2nd ed., Reading, MA: Addison-Wesley, 1985.
12. D. K. Byrne, C. M. Patton, D. Arnett, T. W. Beers, and P. J. McClellan, An advanced scientific graphing calculator, *Hewlett-Packard Journal,* **45** (4): 6–22, August 1994.
13. N. R. Scott, *Computer Number Systems and Arithmetic,* Englewood Cliffs, NJ: Prentice-Hall, 1985.
14. T. M. Whitney, F. Rode, and C. C. Tung, The 'powerful pocketful': an electronic calculator challenges the slide rule, *Hewlett-Packard Journal,* pp. 2–9, 1972.
15. W. E. Egbert, Personal calculator algorithms I: square roots, *Hewlett-Packard Journal,* pp. 22–24, May 1977.
16. W. E. Egbert, Personal calculator algorithms II: trigonometric functions, *Hewlett-Packard Journal,* pp. 17–20, June 1977.
17. W. E. Egbert, Personal calculator algorithms IV: logarithmic functions, *Hewlett-Packard Journal,* pp. 29–32, April 1978.
18. T. W. Beers, D. K. Byrne, J. A. Donnelly, R. W. Jones, and F. Yuan, A graphic calculator for mathematics and science classes, *Hewlett-Packard Journal,* **47** (3): 45–58, June 1996.

DIANA K. BYRNE
Texas Instruments Incorporated