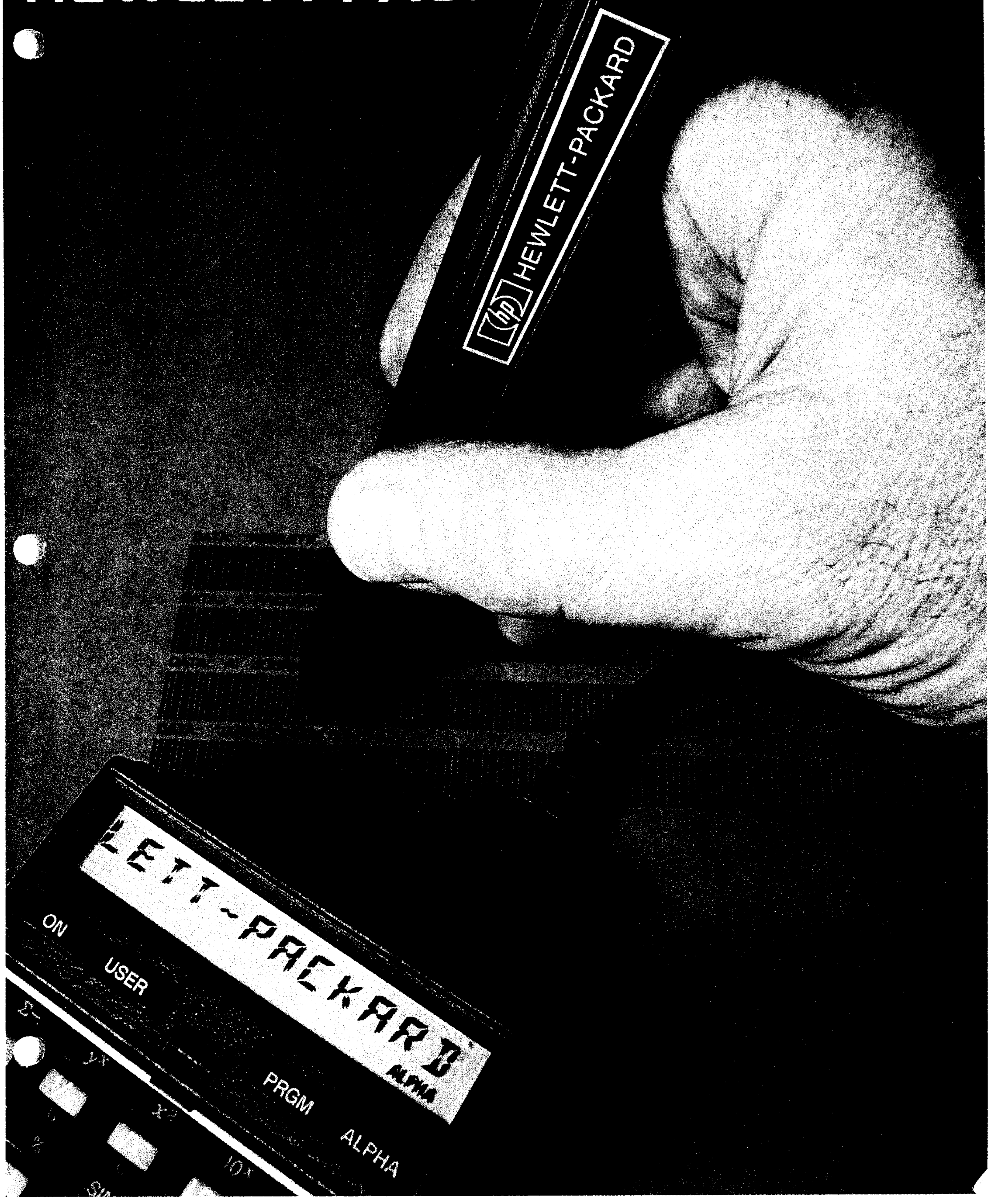
JANUARY 1981

HEWLETT-PACKARD JOURNAL



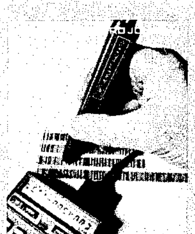
HEWLETT-PACKARD JOURNAL

Technical Information from the Laboratories of Hewlett-Packard Company

Contents:

JANUARY 1981 Volume 32 • Number 1

In this Issue:



Computer application programs tell a computer how to accomplish specific tasks. Typically, such tasks involve the processing of information that the computer finds stored in its memory or coming in from a terminal or other input device. The articles in this issue deal with ways of getting that information into the computer.

Featured on our cover is a new digital bar-code wand. You've seen bar codes on the labels of products at the supermarket. The bars and spaces in these codes represent numbers or letters, and the bar-code wand converts them into electrical signals a computer can read. It's a fast, reliable, relatively cheap and easy-to-use method of putting data or programs into a computer. The wand comes in two versions, Model HEDS-3000 (page 3) for general use, and Model 82153A (cover and page 11) for use with the HP-41C Calculator, featured in these pages in March 1980. HP-41C solutions books give you not only the program listings but also the equivalent bar code, so all you have to do to program the calculator is scan the pages of the book with the wand. Easy.

The article on page 15 describes another computer input device, Model 9111A Graphics Tablet. With this useful tool you can draw the computer a picture. For example, an engineer can draw circuit diagrams or mechanical structures and see them appear on a display or plotter as they are drawn. Compared to describing pictures to the computer in terms of coordinates, the tablet is a lot more convenient.

An article about a new data capture software package, DATACAP/1000, begins on page 25. DATACAP is a package of programs for HP 1000 Computers. Its purpose is to help manufacturing companies collect data from factories—information about inventories, work in process, time and attendance, distribution. Guided by DATACAP, factory personnel enter this data into HP data capture terminals using keyboards, cards, or bar-code wands. DATACAP receives this data, checks it, and stores it in a data base in the computer's memory, so that report-generating programs can process it and produce timely and accurate reports for management. DATACAP is designed to adapt easily to each individual factory's needs. It asks its owner questions to find out what those needs are and generates a tailor-made information-gathering system to meet them. This helps managers increase operating efficiency, minimize inventory investment, and improve customer service.

-R. P. Dolan

Editor, Richard P. Dolan • Associate Editor, Kenneth A. Shaw • Art Director, Photographer, Arvid A. Danielson
Illustrator, Nancy S. Vanderbloom • Administrative Services, Typography, Anne S. LoPresti • European Production Manager, Dick Leeksma

Handheld Scanner Makes Reading Bar Codes Easy and Inexpensive

This lightweight wand contains the light source, reflected-light sensor, and digital signal shaping circuitry needed for scanning bar-code patterns reliably.

by John J. Uebbing, Donald L. Lubin, and Edward G. Weaver, Jr.

BAR CODES ARE A RAPIDLY GROWING method of manual data entry that can be used as an effective alternative to keyboards. A bar code is a self-contained message with information encoded in the physical widths of bars and spaces in a printed pattern. Hewlett-Packard's new HEDS-3000 Digital Bar Code Wand (Fig. 1) is a reliable interface between these printed bar codes and a digital decoding system. When the handheld wand is used to scan the bar code, it converts the light reflected from the printed bars and spaces into TTL or CMOS-compatible logic levels. The resulting digital signal is available for input to a digital decoding system.

The wand contains a precision optical sensor, an analog amplifier, a digitizing circuit and an output driver. The integration of the emitter, detector, and optics of the optical sensor into a single package makes the scanner rugged and reliable. The output of the sensor is proportional to the reflectance of a 0.2-mm (0.008 in) diameter spot in front of the opening in the wand tip. The sensor signal is amplified and converted into a logic-level output by a circuit contained in the wand body. This output is a logic high (1) level when the sensor is looking at a black bar and a logic low (0) level when it is looking at a reflecting white space. The output of the wand is connected to the user's digital processor, which typically measures the time intervals corresponding to the widths of these bars and spaces as the wand is scanned over the bar code. The user's decoding algorithm can then decode these time intervals into binary, numeric, or alphanumeric information depending on the bar-code format. Parity and check-sum information can be used to verify that the read operation was error-free before the information is entered into the computer.

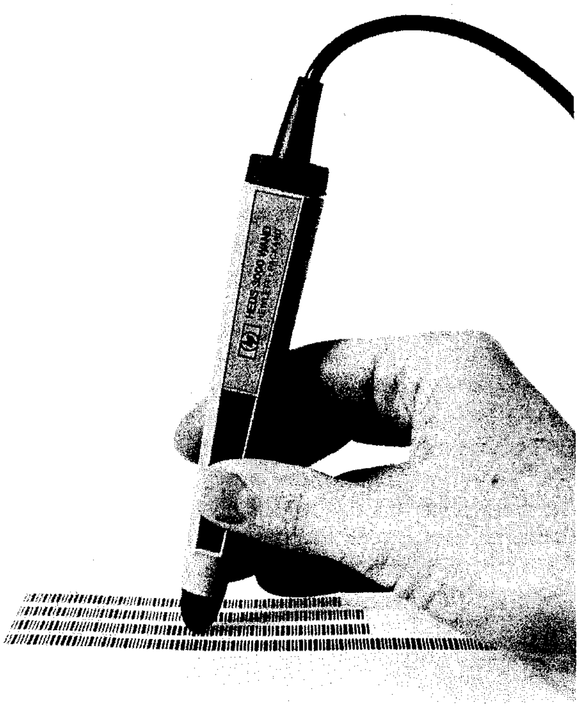


Fig. 1. *The HEDS-3000 Digital Bar Code Wand contains all of the components necessary to convert a printed bar-code pattern into a digital signal for use in data processing.*

The optical sensor has a spot-size resolution that allows bar-code widths as small as 0.3 mm (0.012 in) to be read reliably. This resolution is ideal for dot-matrix printed bar codes. In addition, the 700-nm wavelength of the sensor light source enables sensing of many colored bar codes, although the HEDS-3000 is primarily intended for black-and-white patterns.

The circuit in the HEDS-3000 bar-code scanner uses a push-to-read switch to save power in battery-operated systems. Another battery-oriented feature is the wide range of operating voltage. The wand circuit is designed to use a single power supply within the range of 3.6 to 5.75 volts. At maximum voltage the wand will draw less than 50 mA when the switch is depressed. The circuit's open-collector transistor output allows the wand to interface with either TTL or CMOS circuits.

The HEDS-3000 is packaged in a rugged ABS-plastic case. A strain-relieved one-metre cord on the wand is terminated in a nine-pin D-style subminiature connector with an integral squeeze-to-release retention mechanism. The low-friction tip unscrews for cleaning the sensor window or for replacement in the event of excessive wear.

A key specification of the HEDS-3000 is the accuracy with which the wand can measure the bar and space widths of bar-code patterns. This width-error specification is compatible with the specifications of bar-code printers to allow the system designer to evaluate the trade-offs in the design of a bar-code system. The wand can typically measure the width of the first bar in a code pattern within 0.1 mm (0.004 in) and the interior data bars and spaces with an accuracy of 0.05 mm (0.002 in). The wand is designed to read bar codes in all handheld orientations within a cone of

30° from the normal to the bar-code pattern. The wand will also operate over hand-scanned speeds ranging from 76 to 760 mm/s (3 to 30 in/s) and over an operating temperature range of 0 to 55°C (32 to 130°F).

Reflectance Sensor

The high-resolution reflectance sensor that is an integral part of the HEDS-3000 Bar Code Digital Wand is also available as a separate component (HEDS-1000) for use in other sensing applications. These include pattern recognition, object sizing, optical limit switching, tachometry, defect detection, dimensional monitoring, line location, paper-edge sensing and bar-code scanning.

To be useful in a low-cost, portable bar-code scanner the reflected-light sensor must be able to detect 0.25-mm (0.01-in) wide bars and spaces with high sensitivity, have low power consumption, exhibit good reliability, and have low manufacturing cost. The design of the HEDS-1000 meets all of these requirements.

Several optical configurations were considered for the reflectance sensor arrangement, including half-silvered mirrors, coaxial source and detector arrangements, and separate packages for the source and the detector. The bifurcated side-by-side approach (Fig. 2) was selected because it provides a compact structure that fits into the tip of a wand and allows both the source and detector to be mounted on the same substrate. One of the possible drawbacks of this configuration is that stray light reflected from the split lens system can generate a photocurrent when there is no object to be sensed. This is judged not significant because of the low level of the stray light relative to the signal and because the signal conditioning circuit can compensate for the presence of stray light.

To obtain high sensitivity, a large aperture is needed so that a substantial amount of light is focused on the bar-code pattern and a large portion of the reflected light can be collected and focused on the detector. Early experiments determined that spherical lenses with the necessary aperture exhibited spherical aberrations that were too great to allow 0.25-mm width resolution. Thus, to provide a numerical aperture of 0.4 and a sharp focus, an aspheric lens design was used. The two plastic lenses have hyperbolic surfaces and are precision molded together as a pair.

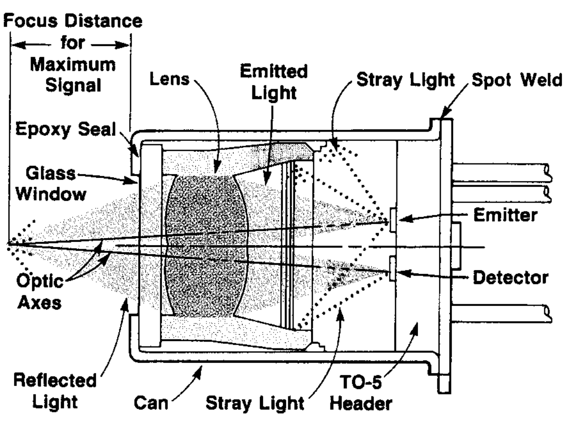


Fig. 2. *The HEDS-1000 High-Resolution Reflectance Sensor used in the HEDS-3000 is also available as a separate product. This sensor contains both an emitter and detector in a single package. A bifurcated lens design focuses the light from the sensor's emitter onto the area to be sensed and focuses the reflected light back onto the detector area in the sensor.*

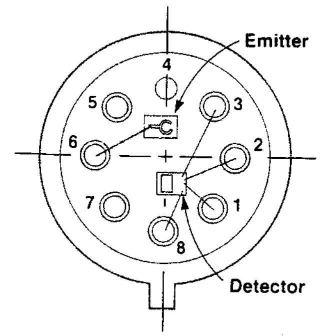


Fig. 3. *The emitter and detector chips in the HEDS-1000 are mounted on a standard TO-5 header as shown.*

The source is a small (0.18-mm diameter) light-emitting diode (LED) with a wavelength of 700 nm. The wavelength of the LED was chosen to provide the best light-generating efficiency combined with the ability to sense dye-based ink patterns with good contrast. The detector consists of a silicon photodiode and transistor integrated on a single chip. The reflectance signal can be obtained directly from the photodiode or through the integral transistor configured as a high-gain amplifier.

One half of the lens system focuses the light from the LED onto the paper. The other half focuses the reflected light back onto the integrated photodetector chip. The area of the photodiode is somewhat larger than the LED to compensate for assembly tolerance and to improve the depth range over which good sensing can occur. Letting the LED size determine the resolution in this way minimizes the LED power consumption for a given photocurrent level. This is important for battery-powered operation.

A standard TO-5 header used for the substrate provides a compact method for bringing out the leads (Fig. 3). Gluing the lens directly to the header was first evaluated, but the moisture-fogging and temperature-cycling test results were inferior with this arrangement. The configuration adopted uses a tall metal can with an adhesive-bonded glass window for good moisture resistance. The molded bifurcated plastic lens pair is bonded to the can with a soft silicone adhesive for good temperature cycling performance. Both the LED and the detector are die-attached with a eutectic alloy to the header using a precision collet tool that attaches the chips with placement accuracy better than 0.05 mm.

Each HEDS-1000 Reflectance Sensor is individually tested by an electrical test system controlled by an HP 9825A Computer/Controller.[1] The signal levels, transistor parameters and other data sheet values for the HEDS-1000 are guaranteed by this system.

Bar-Code Reader Circuit

The circuit in the HEDS-3000 is designed to convert the low-level analog signals from the photodiode in the sensor module to a compatible logic level that can be easily interfaced to digital systems.

If the optical sensor module can resolve widths much narrower than the minimum bar width and there is a great

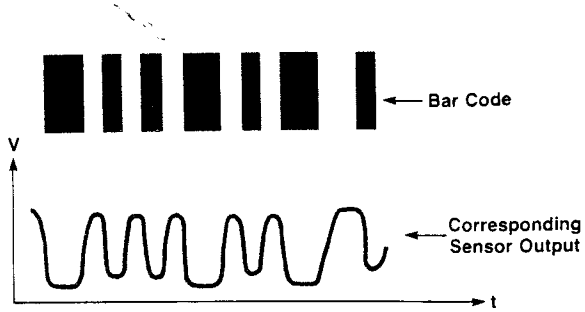


Fig. 4. *A typical bar-code pattern and corresponding waveform resulting from a scan of the pattern.*

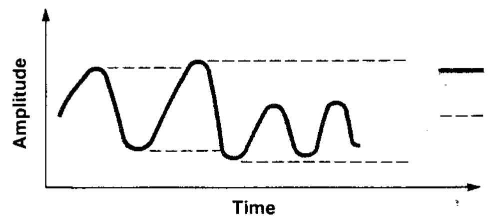


Fig. 5. *When the signal maxima and/or minima change as shown, the peak detectors must be reset frequently to correctly detect signal peaks.*

deal of contrast between bars and spaces, it would appear that only a comparator connected to the photodiode would be needed to perform the digitizing function. Unfortunately, this is not often the case and some type of circuit is required to determine where a bar ends and a space begins. A typical analog waveform from a bar-code scan is shown in Fig. 4. The reason why the signal changes gradually when going from a dark to a light region is that the finite spot size viewed by the sensor integrates the light and dark areas.

There are several techniques for determining precisely where a light-to-dark or dark-to-light transition occurs. One approach is to differentiate the signal, because the maximum rate of change occurs at the transition. The disadvantage of this technique is that it is very sensitive to both electrical and optical noise and is heavily dependent on scan rate. Some designs use ac coupled clipping circuits that amplify the signal to the point where it looks like a pulse train. This approach is very sensitive to changes in average signal value and does not digitize accurately enough for many requirements.

A potentially accurate technique is to detect the positive and negative signal peaks and set a threshold halfway between the two peak values. This point corresponds to the sensor viewing area being positioned half on the bar and half on the space. This works well if the signal level remains constant or the signal maxima increase or the signal minima decrease. In a typical scan, this is not the case because the height above the bar-code pattern (tag) and the wand angle can vary considerably during the scan, which changes the dc level as well as the modulation amplitude. If this technique is used, a means for resetting the peak detectors after each transition is needed to allow for the case of decreasing signal maxima or increasing minima (Fig. 5). Previous solutions of this problem have either compromised performance by using rapidly decaying peak detectors or have required the extensive use of digital circuitry to perform this function.

Another problem with peak detection, and some other schemes as well, is how to handle static conditions when there are no transitions and reliable references as to what is black and what is white. Arbitrary logic default states and fixed black-to-white thresholds could be used. The problem is that arbitrary states will not always be correct and fixed thresholds are subject to too many error sources, such as ambient light and supply voltage variations, sensor and amplifier circuit drift, and manufacturing tolerances.

The design of the bar-code reader circuit had to deal with these processing problems while complying with a number of other constraints. One major constraint is size, because the entire circuit has to fit in the wand body. To accomplish this with previous circuit approaches requires the design of a complex analog/digital integrated circuit. This was the approach initially taken before a simpler signal processing technique was found. Another constraint is the requirement for operation over a supply voltage range of 3.6 to 5.75 volts to be compatible with the HP-41C Calculator. Economic constraints dictate that the solution use low-cost components and assembly techniques. This eliminates approaches that require trimming for proper operation.

The circuit can be broken into three major blocks—amplifier, signal processor, and digital output. Fig. 6 shows the circuit schematic partitioned into these functions. The

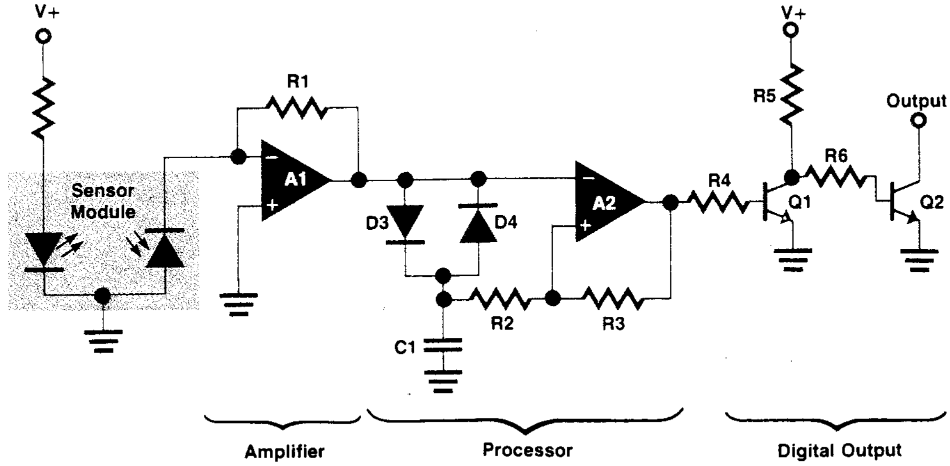


Fig. 6. *Circuit schematic for the HEDS-3000.*

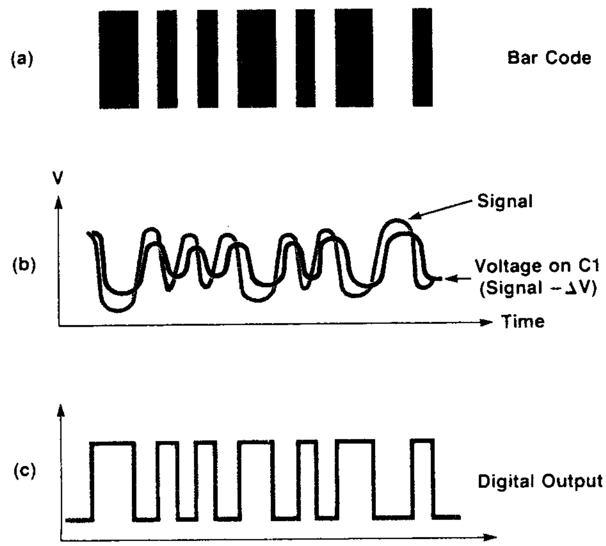


Fig. 7. *Operation of the processing circuit in the HEDS-3000. When the bar code in (a) is scanned, peak detection converts the resulting voltage across C1 (b) into a logic-compatible signal (c). Note: ΔV=diode forward voltage drop.*

amplifier increases the signal from the photodiode up to a processable level by converting a current on the order of 100 nanoamperes to a voltage of about 1 volt. An amplifier with a transimpedance of about 10 megohms is used to do this.

The processing circuit uses peak detection to decode bars and spaces. A simple positive and negative peak detector uses two diodes (D3 and D4) and a capacitor (C1). A positive voltage peak minus a diode drop is stored on C1. The comparator circuit (A2, R2, R3) compares this value with the output of amplifier A1. When the signal drops below the value stored on C1, the comparator changes state. As the signal goes through its negative excursion, D4 starts conducting and the negative peak voltage minus a diode drop is stored on C1. As the signal becomes more positive than the voltage on C1, the comparator changes state again. Diode D3 then conducts until the positive peak voltage minus a diode drop is stored on C1 again and the process repeats itself. Fig. 7 shows this operation. Notice that this circuit does not change state when the photodiode viewing area is located half on black and half on white. Instead it changes state before that point, introducing a leading phase shift which has little adverse effect. The advantage of this approach is that the peak detector always resets itself so that it can detect the peak of a decreasing signal maximum or an increasing signal minimum. It does this without requiring complex circuitry to track the signal correctly.

The comparator in the processing section provides two functions. First, it digitizes the signal by comparing the signal level to the voltage on C1. Resistors R2 and R3 provide hysteresis for this function to insure clean transitions in the presence of a noisy signal. The second, and much more subtle function of A2, R2, and R3 is to keep the output in the correct state under static conditions. As mentioned earlier, this is a problem with most decoding approaches. This scheme solves many of the problems associated with static conditions in a simple manner. It requires no absolute reference level or adjustments. The only constraint is to define what state the wand is in after power up. The circuit can be made to power up in either state depending on circuit details. Once the power-up condition is reached, the wand will correctly track all transitions dynamically and retain the correct state information statically.

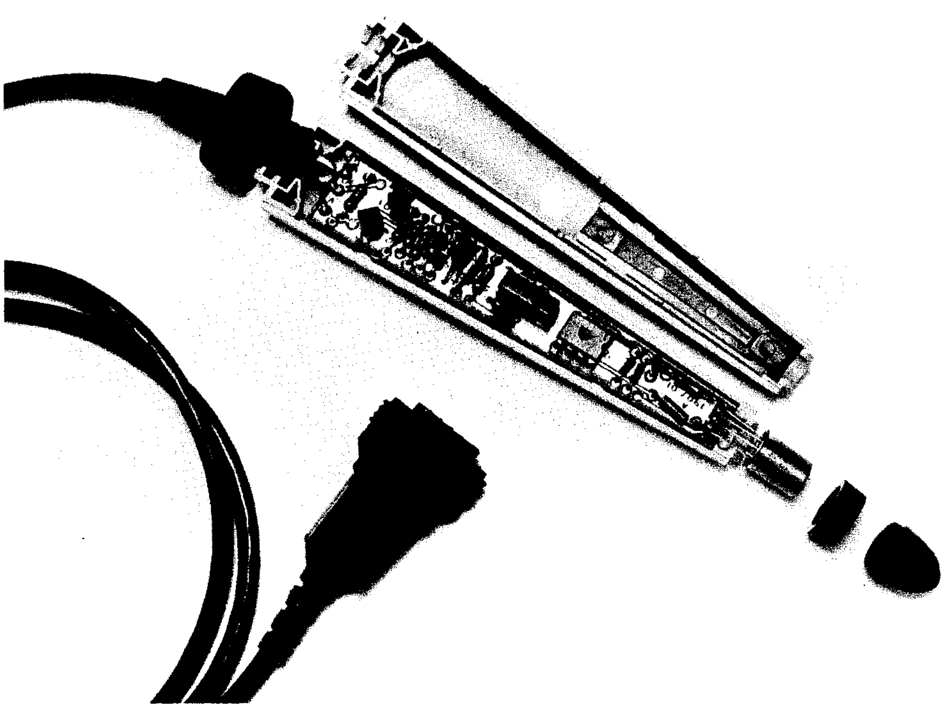


Fig. 8. *Wand case before assembly. The molded cylindrical body contains the sensor and the electronics and is held together by two rings.*

What Is a Bar Code?

Bar codes are messages with data encoded in the widths of printed bars and spaces on a piece of paper. If you consider that standard printed characters are two-dimensional, then bar codes are one-dimensional characters that have been stretched in the vertical dimension. From this perspective, bar-code scanning can be seen as the one-dimensional version of optical character recognition (OCR). The simplicity of scanning and decoding one-dimensional patterns is one main reason that bar-code systems are more prevalent than OCR systems and are expected to remain so for some time. An analogy for how bar-code systems work is to consider a printed bar code as a pulse-code-modulated (PCM) signal where linear distance on the paper is equivalent to time and the white and black reflectance levels of the bar-code pattern (tag) on the paper are equivalent to the high and low logic levels of the electrical signal.

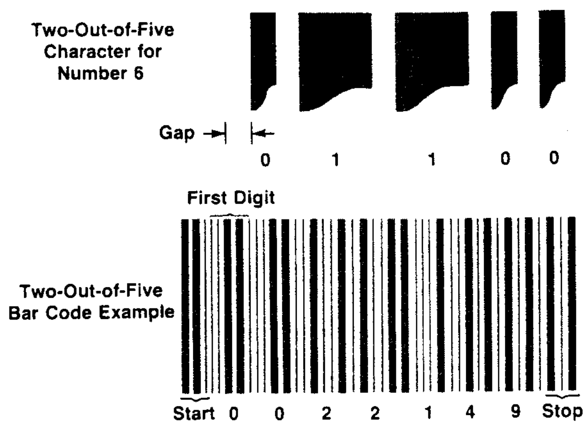


Fig. 1. *Illustration of two-out-of-five bar code. Each character is represented by a combination of five bars that are each one or three units wide (see Table I).*

The number of different possible coding schemes for bar codes is endless. However, the majority of schemes now in use can be classified as two-level codes. In a two-level code a wide bar or space represents a binary one and a narrow bar or space represents a binary zero (or vice versa). Usually the first two bars of a tag are used to define the initial value of a narrow width, then all bars and spaces read by the wand are compared to this standard value and assigned values of either one or zero depending on their respective widths.

Variations of the two-level code include versions where only bars carry information or where groups of bars and/or spaces (e.g., five bars and four spaces) represent single coded characters with internal parity checks. Bar codes can be read either bidirectionally or from one direction only. Almost all codes include a checksum digit encoded at the end of the bar code to provide security against improperly decoded characters. Code 39™ (developed by Interface Mechanisms) and two-out-of-five bar codes are examples of alphanumeric and numeric two-level character bar codes. Paperbyte™ and the HP-41C bar codes are two-level binary codes. Another common bar code is the four-level Universal Product Code (UPC) that is used for identifying grocery products. In this code a numeric character is defined by two bars and two spaces. Each bar and space is either one, two, three, or four modules wide and the total character width is constrained to be seven modules wide (a module is a unit of bar-code width). Security of the UPC code is further insured by internal parity checks and a checksum digit encoded at the end of the tag.

To get a better idea of exactly how a bar code works let's examine the two-out-of-five bar code more closely. Two-out-of-five code is a numeric code with ten digit-characters, a start symbol, and a stop symbol. The version described here is a

Fig. 2. *In this industrial application each assembly is identified by bar code and the QA defect code is read directly by a bar code wand from a code index, eliminating copying errors.*

Table I

Two-out-of-five bar-code character decoding. (0 = narrow bar, 1 = wide bar, M = margin, a very wide space at the beginning and end of a tag.)

Bar-Code Pattern	Character
00110	0
10001	1
01001	2
11000	3
00101	4
10100	5
01100	6
00011	7
10010	8
01010	9
M110	Start
010M	Stop

bar-only code with no information in the spaces. Narrow bars represent zero bits and wide bars (typically three times the width of the narrow bars) represent one bits. Each numeric character is a set of five consecutive bars of which two bars out of the five are wide. The character table and coding for this code is shown in Table I. A typical character and tag for two-out-of-five bar code is shown in Fig. 1. There are typically three levels of self checking that insure that the decoded two-out-of-five data is valid before it is entered into the computer. First, each character is checked to see that there are two and only two wide bars. Second, there must be an integer multiple of five bars between the start and stop symbols. Any extra or missing bars will cause the decoding system to void the read. Finally, the last character should be a software-determined checksum number to verify that the decoded data is the same as the encoded data. For example, if the last digit of the sum of the data values is encoded as your checksum digit, then it should also be the last digit of the sum of the decoded data values. If it is not, then the read is invalid.

The main applications for bar codes are as alternatives to keyboards. Bar codes allow fast and accurate hand entry of small amounts of data by minimally trained operators. The commercial market for bar codes is currently divided into three major types of users. The first and most widespread application of bar codes is in point-of-sale (POS) computer systems in grocery stores. Merchandise labeled with bar codes is scanned at the checkout counter to automatically enter the price and item on the customer's bill and update the store's inventory. This system allows faster checkouts, fewer errors, and more effective inventory management. The second major application of bar codes is in industrial data entry (Fig. 2). This category includes warehouse inventory control, identification of assemblies for process monitoring or work scheduling, and remote data collection. The purpose of bar codes in these industrial environments is to provide a simple error-proof means to hand-enter data into the central computer. The third and newest area of bar-code applications is low-cost data entry for microcomputers. Computer software in bar-code form can be mass produced inexpensively by the printing industry and distributed to a broad base of users. Bar-code-formatted software can be used for programming appliances, intelligent instruments, or personal computers. The bar-code option on the HP-41C described in the article on page 11 and the Paperbyte™ bar-code programs published by BYTE Magazine are excellent examples of the use of bar codes for low-cost consumer-oriented data entry.

If the voltage on C1 is greater than the output of A1, the output of A2 will switch to a high state. If the signal then remains constant, (static condition) and if R3 is not connected from the output to the noninverting input, C1 eventually charges via D3, D4 and the input bias current of A1 to a level comparable to the signal and the output state is indeterminate. With R3 connected, C1 charges through R2 and R3 toward the output level of A2, keeping the output of A2 in the high state. If the voltage on C1 becomes less than the signal, the output of A2 switches to a low state and under static conditions R2 and R3 discharge C1 toward that level, keeping the output in the low state. The circuit thus functions as a latch when static conditions are encountered.

The digital output section consists of transistors Q1 and Q2 and resistors R4, R5, and R6. The transistors serve to buffer the output of A2 and provide logic-compatible levels. The output stage shown in Fig. 6 provides a logic zero output for a dark-to-light transition. If the inverse is desired, the output can be taken from the collector of Q1, and Q2, R5, and R6 can be deleted.

Development of the Wand Package

After considering a number of alternatives such as a rectangular-box-type reader and a molded tube wand structure, the basic configuration shown in Fig. 1 and Fig. 8 was developed. The slim tip allows the operator to see the code while it is being read. The long, somewhat bulky body not only accommodates the printed circuit board, but feels comfortable in the operator's hand, somewhat like the handle of a paint brush. The switch actuation plate is long and comes up the side of the wand body so that the switch can be actuated by the operator's thumb or forefinger even though the wand is grasped in a number of different ways. The case itself is constructed of two separate halves, with the printed circuit board fitting in between. To provide aesthetic lines for the product, the case halves are held together with two molded rings of an ABS-and-polycarbonate-plastic alloy for maximum impact resistance and good temperature range. A vinyl strain relief is molded onto the end of the cord. The detailing at the end of the strain relief fits into grooves in the clamshell halves. The strain relief and cord system have successfully passed one thousand cycles of heavily loaded 180°-bend tests. Fig. 8 shows the wand case before assembly.

The tip of the wand has a slim profile and still accommodates the reflective sensor. The hole in the tip is large enough so that paper dust and other foreign material does not accumulate inside the tip, but falls out in normal use. The tip is molded of teflon-filled acetal since this material gives the smoothest ride over the paper and suffers the least wear. When the tip does wear out it can be replaced easily because it screws into the wand body.

Specification and Testing of the Wand

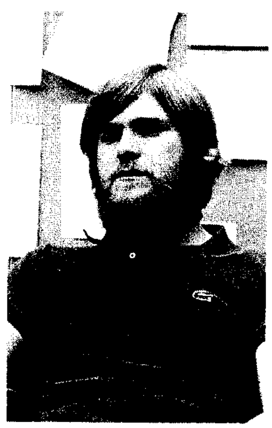
The HEDS-3000 uses a new approach to specifying bar-code wand performance. Historically, bar-code wands were analog devices and previous specifications dealt with such analog signal characteristics as amplitude modulation and noise. A bar-code wand with a digital output requires definition of a new set of performance measures. This digital performance specification properly describes the effects of mechanical, electrical, and optical parameters upon the wand's performance. This product specification then defines the conditions under which the wand will read bar codes successfully in a customer's system. However, different customers are interested in different specifications for different types of systems. Each system has requirements based on the specific bar code, bar-code printer, and decoding algorithm chosen by the system designer. The effect of the wand on the readability of the bar code in each of these systems is slightly different.

The desire to specify the performance of the wand in a manner that makes sense to a system designer led to the concept of system width error. The width error is a modification of the specifications used to describe bar-code printing tolerances. Bar-code printers generally specify edge resolution. The width of a printed bar or space can vary from the desired width by the printing uncertainty of the edges. Since these edge errors are independent of the width of the bar, printer performance for any bar is specified by a width error (e.g., narrow bars = 0.25±0.05 mm and wide bars = 0.50±0.05 mm). Like bar-code printing errors, wand reading errors are primarily edge errors. The performance of the wand is characterized by the accuracy with which it can measure bar widths. The wand bar-width error can then be summed with the width errors of the bar-code printer to determine the total width error at the start of the system decoding algorithm. A paper analysis of system performance accounting for printer errors, wand errors, and the specific software algorithm can then be done. Further, by specifying the bar-width errors and space-width errors separately, the designer can create software to compensate for offset errors characteristic of both wands and printers. Bars that consistently appear wider and spaces that appear narrower are examples of offset errors. When the specification is in terms of width error, the system designer can easily understand the trade-offs in bar-code system design.

The width error is the difference between the calculated bar or space width and the optically measured bar or space width. A two-level black-and-white code on photographic paper is used as the standard tag for characterization. The bars and spaces of these tags are optically measured with a toolmaker's microscope. When this standard tag is read by a wand under test at a constant scan velocity, the wand-measured bar width can be calculated from the duration of the bar output level from the wand. The width errors are separated into bar and space errors and into maximum and minimum errors. Because the magnitude of the width error is dependent on the width of the preceding space(bar) as well as the measured bar(space), the width errors are also sorted into bar/space and space/bar categories to give the system designer a more complete description of the wand performance. This information can be used to analyze the functionality of the designer's decoding software. For example, the system designer can see immediately that decoding schemes comparing bars with bars and spaces with spaces will cancel the systematic or offset errors that make bars appear wider and spaces appear narrower, while decoding software comparing bars with spaces will magnify these errors. To provide as complete a picture as possible of the HEDS-3000 performance in a variety of operating conditions, the wand's width errors are characterized as a function of wand height, angle and orientation, tempera-

Edward G. Weaver, Jr.

Eddie Weaver received the BSEE degree in 1975 and the MSEE degree in 1977, both from Rice University. He joined HP shortly after and has worked as a development engineer in the Optoelectronics Division since then. His work has included the testing and characterization of the HEDS-3000 Wand. Eddie is the co-author of three papers on the generation of continuous-wave ultraviolet radiation and optical effects in nonlinear crystals. He is a member of the IEEE and the Optical Society of America. Eddie is a native of Cortez, Colorado and now lives in Sunnyvale, California. He enjoys travel, hiking, backpacking, and gardening.



Donald L. Lubin

Don Lubin was born in Cleveland, Ohio and grew up in Yonkers, New York. He has been with HP since 1973 and has led projects dealing with optocouplers, fiber optics and the circuit design for the HEDS-3000 Wand. Don is currently the section manager for optocoupler, emitter, and detector products development. He was awarded the BSEE and MSEE degrees by Rensselaer Polytechnic Institute in 1972 and 1973, respectively. Don and his wife make their residence in Los Altos, California. Outside of work Don enjoys photography, running, sailing, and collecting clocks.

John J. Uebbing

John Uebbing is the author of a variety of papers on photocathodes, magnifiers, hybrid substrates and other optoelectronic topics. He came to HP in 1973 and has worked on alphanumeric and monolithic displays, was the manager of packaging and emitter detector groups, and is now involved with R&D work on advanced displays. John's previous experience included work on III-V semiconductor photocathodes and electron spectrometers. He has a BSEE degree awarded in 1960 by the University of Notre Dame, a MSEE degree awarded in 1962 by the Massachusetts Institute of Technology, and a PhD degree awarded by Stanford University in 1967. John is a member of the International Society for Hybrid Microelectronics and the Society for Information Display. He is a native of Chicago, Illinois and he and his wife and two children live in Palo Alto, California. John is interested in religious education and enjoys sailing, backpacking, and playing bridge.

ture, scan velocity, minimum bar-width size, and supply voltage.

The width errors of every wand are tested in production before packaging and shipment. The standard photographic test tag is attached to a wheel rotating at a constant speed. The wand is fixtured into the specified test position relative to the tag and the output of the wand is measured by a time-interval-measurement circuit and fed into a 9825A Computer/Controller. The 9825A compares the wand-output bar and space widths with the stored values of the optically measured bar and space widths. The 9825A then calculates the width errors for all the bars and spaces in the test tag, reduces this data, and stores the relevant information. Besides this final performance test, each printed circuit board is tested before final assembly to guarantee frequency response, output levels, and functionality. Each optical sensor is also pretested to guarantee its performance before final assembly. Finally, wands sampled from production lots are tested to ensure product operating life and humidity resistance. The design is characterized for mechanical integrity (shock, strain relief, etc.), operating reliability (temperature cycling, operating life, humidity, etc.), and performance quality (electromagnetic interference (EMI) and width error over diverse operating conditions).

Acknowledgments

The authors would like to thank Perry Jeung and John Lee for their project leadership, Ed Liljenwall, Fred Goodman, Matt Stein, Jim Casciani, and Bob Teichner for package development, Carl Trautman and Ray Wong for circuit work, Nate Walker and John Dunse for manufacturing engineering, Walt Heinzer for reliability testing and John Sien and Julian Elliott for marketing.

Reference

1. D.E. Morris, C.J. Christopher, G.W. Chance, and D.B. Barney, "Third Generation Programmable Calculator Has Computer-Like Capabilities," Hewlett-Packard Journal, June 1976.

SPECIFICATIONS

HEDS-3000 Digital Bar Code Wand

POWER SUPPLY:
V_S: 3.6-5.75 volts
I_S: 50 mA maximum.

DATA OUTPUT:
LOGIC LEVEL: TTL and CMOS compatible
WIDTH ERRORS:

First Bar	0.1 mm typical
Interior Bar	0.05 mm typical
Interior Space	0.05 mm typical

SCAN VELOCITY: 7.6-76 cm/s.
ILLUMINATION WAVELENGTH: 700 nm
TEMPERATURE RANGE
OPERATING: 0°C to +55°C.
STORAGE: −20°C to +55°C.
DIMENSIONS: 133 × 23 × 20 mm (5.2 × 0.9 × 0.8 in). Cable 1 m long.

HEDS-1000 High-Resolution Reflectance Sensor

POWER SUPPLY:
V_d, V_c, V_e: 20 volts maximum, 5 volts typical.
I_{LED}: 50 mA maximum average, 75 mA maximum peak.
POWER DISSIPATION: 120 mW maximum.
PHOTOCURRENTS:
PHOTOCURRENT (white surface): 120 nA, typical.
STRAY PHOTOCURRENT: 20 nA, typical.
FOCAL PROPERTIES:
IMAGE SIZE AT FOCUS (distance for 10-90% response over black-white transition): 0.17 mm.
DEPTH OF FOCUS (to 50% of maximum photocurrent): 1.2 mm.
MAXIMUM SIGNAL POINT: 4.3 mm from front of can.
SOURCE PEAK WAVELENGTH: 700 nm.
TEMPERATURE RANGE:
OPERATING: −20°C to +70°C.
STORAGE: −40°C to +75°C.
PACKAGE: 8-pin TO-5 style package, 12.9 mm long.
PRICES IN U.S.A.:
HEDS-3000 Digital Bar Wand, $99.50 each in small (1-99) quantities.
HEDS-1000 High Resolution Reflectance Sensor, $28.75 in small (1-9) quantities.
MANUFACTURING DIVISION: OPTOELECTRONICS DIVISION
640 Page Mill Road
Palo Alto, California 94304 U.S.A.

HP Model 82153A Wand

PHYSICAL SPECIFICATIONS: Same as for HEDS-3000.
ELECTRICAL SPECIFICATIONS: Supplied with interface plug-in for use with HP-41C Calculator.
PRICE IN U.S.A.
82153A Wand: $125.
MANUFACTURING DIVISION: CORVALLIS DIVISION
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

Reading Bar Codes for the HP-41C Programmable Calculator

by David R. Conklin and Thomas L. Revere III

A SPECIAL VERSION of the HEDS-3000 Digital Bar Code Wand is supplied to Hewlett-Packard's Corvallis Division for use in the 82153A Wand (Fig. 1), an accessory to the HP-41C programmable calculator. Corvallis Division attaches an interface module containing two integrated circuits—a custom wand interface chip, and a 4096-word microcode ROM. In this article we describe the wand interface chip, the bar-code formats recognized by the HP-41C, and uses for the bar code and wand in the HP-41C calculator system.

The wand interface chip is a CMOS integrated circuit that converts the electrical signals from the wand into binary data, apportions the decoded data into eight-bit bytes, stores the byte(s) for retrieval by the HP-41C, and interfaces with the HP-41C bus lines to transfer the data to the calculator's CPU. The interface chip is located in the HP-41C-compatible plug which is attached to the 82153A Wand by a cable. To read a row of bar code, the wand is scanned across the bar code to generate a time-varying electrical signal that corresponds to the widths of the bars and spaces. The bar-code encoding scheme represents a logic zero by a bar with a relative width of one unit and a logic one by a bar with a relative width of two units. All spaces are one unit wide. A row of bar code may contain up to 16 bytes of data.

Fig. 1. *The 82153 Wand accessory to the HP-41C Calculator provides easy entry of data and programs printed in bar code.*

Decoding is done by counting the HP-41C system clock cycles (~360 kHz) between space-to-bar and bar-to-space transitions and comparing the result to a reference derived from the counts for the previous bar and space. Because inherent wand width bias, acceleration, rotation and other scanning irregularities introduce error into bar and space counts, the definitions of a one and a zero must include significant margins to reduce the possibility of an erroneous decode. A logic zero is defined as any bar less than 3/2 times the unit width established by the previous bar and space while a logic one bar must be greater than or equal to 3/2 times the unit width. The reference is created by first adding 1/2 the count for the previous space to 1/2 the count for the previous bar (1/4 the count for the bar, if the bar was decoded as a logical one or two-units-wide bar). The result is then added to 1/2 of itself to create a 3/2 unit-width reference.

The logic used to decode and store bar codes in one of two identical 16-byte buffers is controlled by a 64-state ROM machine which has 16 instructions and eight branch qualifiers. The decoding algorithm is illustrated by the flow chart in Fig. 2. The tests for bars and spaces include a test for maximum count ($\geq 2^{14}$) that branches back to the start state if the count is exceeded.

Maximum and minimum acceptable bar widths are determined by the scan speed and the interface chip. The maximum count for a bar is limited to less than 2^{14} HP-41C clock cycles. For a clock period of 2.63 μs, this is equal to a duration of about 43 ms. If the scan speed is 76.2 mm/s (3 in/s), the maximum bar width is about 3.2 mm (0.125 in). The minimum bar width is set by the time required to decode a bar, establish a new reference and store eight bits in a buffer. Seventy-six clock cycles are required, corresponding to a minimum bar or space width of about 0.17 mm (0.007 in) at a high scan speed of 762 mm/s (30 in/s).

In addition to the data bars, the interface chip requires that a row of bar code (Fig. 3) have four additional bars—one unit-width bar at each end of a row of bar code to be used in conjunction with the adjacent space to establish an initial reference and another bar next to each reference bar to determine scan direction. The direction bar encountered first in a right-to-left scan is two units wide; in a left-to-right scan it is one unit wide. The direction bars enable the chip to determine which direction the user is scanning and therefore in what order the decoded data should be sent back to the system CPU. The CPU always receives the least significant bit (LSB) of the leftmost byte first. The interface chip accomplishes this by conditionally storing the de-

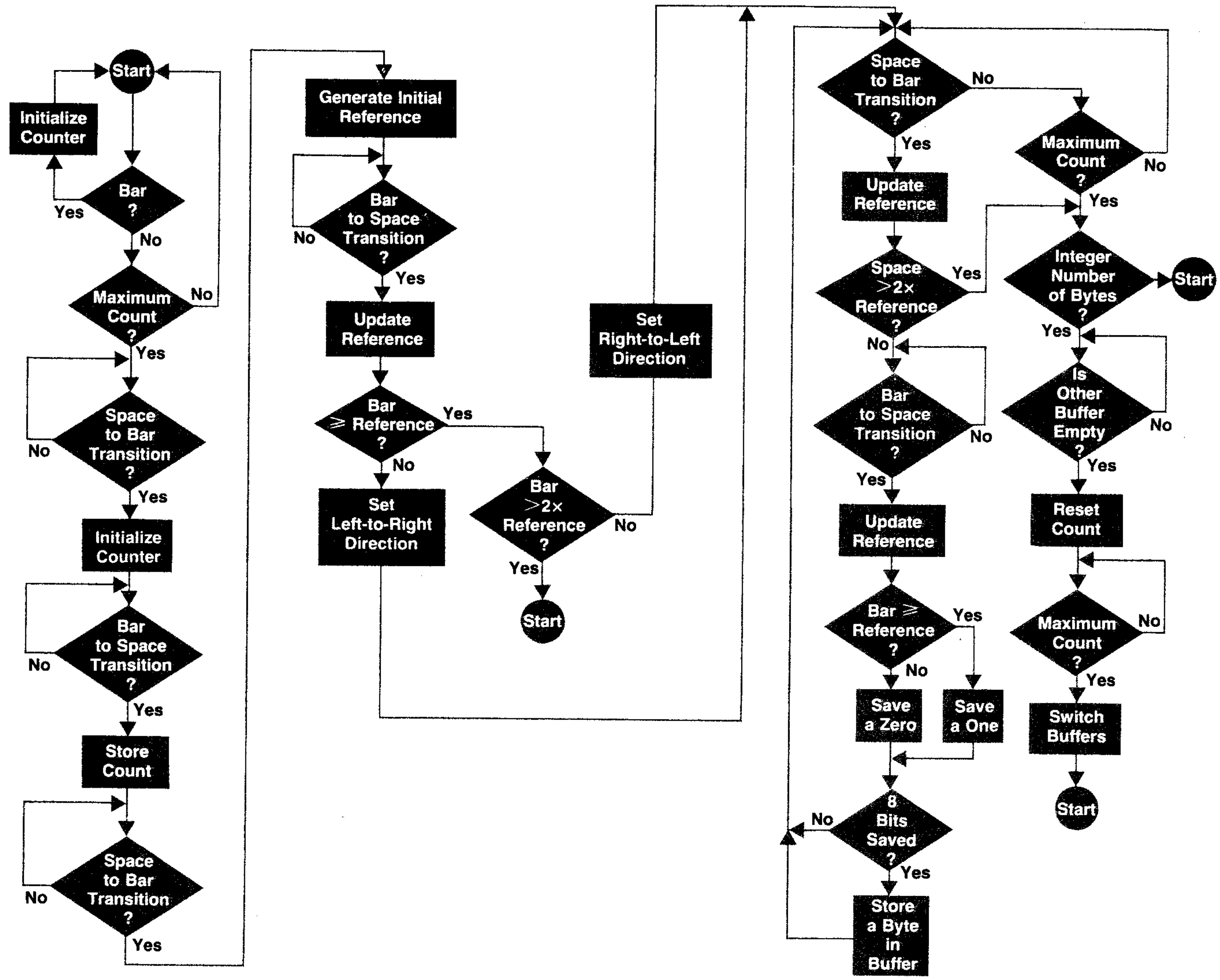


Fig. 2. *Flow chart of bar-code decoding algorithm used by the 82153A Wand interface chip.*

coded bytes in one of two sequences in a 16-byte buffer.

The location in which each byte is stored in a buffer is determined by an address pointer that is reset to the first location in the buffer at the beginning of each scan and is incremented if the scan is right-to-left and decremented if the scan is left-to-right. When a read operation is started by the CPU, and the scan is left-to-right, the pointer is reset to the first location in the buffer and is decremented for each subsequent byte. The pointer is not reset if the scan is right-to-left but is still decremented during readback.

During the time that decoded bar code is being stored in one of the buffers, the other buffer will send a byte from the location indicated by its pointer in response to an HP-41C CPU instruction requesting data from the interface chip. After a line of bar code has been successfully read and the other buffer has been emptied the newly filled buffer is allowed to communicate with the CPU. Two of the HP-41C's input flags are reserved for the wand. Because the wand chip uses the system clock for all internal functions, it prevents the calculator from returning to the clockless standby state by activating flag zero when the wand is turned on and pointed at a white surface and therefore presumably may be in the process of decoding bars. Flag two is used to signal the CPU that data is available and flag two will remain active until the buffer is emptied. The interface chip also wakes up the HP-41C from the off or standby states by pulling on the ISA bus line when the wand is turned on and pointed at a white surface.

Bar code generated for the HP-41C falls into one of four logical types—program bar code, data bar code, and bar code representing keystrokes (paper-keyboard bar code) or complete key phrases (direct-execution bar code). See Fig. 4 for a diagram of the bar code types. When the wand microcode sees a row of bar code only one or two bytes in length, the wand assumes that it is paper-keyboard bar code. One-byte rows have four bits for data and four bits for a checksum that is mirror symmetric to the data pattern. This convention makes one-byte rows immune to errors in decoding the direction bits. Two-byte rows have twelve bits of data and a four-bit checksum that is computed as a sum of the data in four-bit nibbles with end-around carry.

Rows of more than two bytes are assumed to be data,

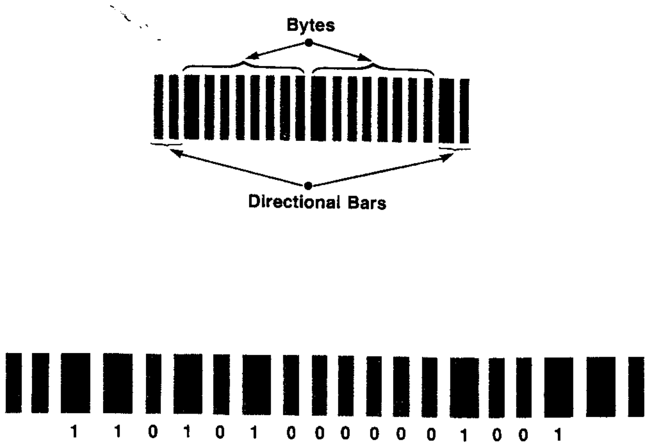


Fig. 3. *Basic format for bar codes read by the 82153A Wand in the HP-41C Calculator system.*

program, or direct-execution bar code. In each case, the leftmost byte is an eight-bit checksum and is followed by a four-bit type specifier. Program bar code includes a four-bit sequence number and two four-bit quantities that record the numbers of leading and trailing partial function code bytes (i.e., how much of the row is taken up with function codes that begin or end in other rows). The overhead—checksum, type, sequence number, and partial function code information—amounts to three bytes, which leaves up to thirteen bytes for the program itself. The sequence number, since it is four bits wide, only defines the sequence of the row within the surrounding sixteen rows; but this is adequate to warn the user when a row has been skipped or read twice. The checksum is a running eight-bit-wide sum with end-around carry of the current row and all preceding rows.

A data bar-code row may contain either a number or an alphanumeric string. The number may have up to ten digits of mantissa and two digits of exponent; the alphanumeric string may be as long as fourteen bytes.

Direct-execution bar code represents a complete key phrase (e.g., XEQ A or STO 12). After two bytes of overhead, the key phrase itself may be from one to nine bytes long.

Loading programs is the primary use envisioned for the 82153A Wand in the HP-41C Calculator system. The wand is less expensive than the card reader, although not as fast. It is much faster, more reliable, and less tiring than hand keying in a program from a program listing. To load a program with the wand, the user simply begins by scanning the program at the first bar-code row. It is not necessary to execute any function or do any other initialization beforehand. The medium of the printed page is widely available and inexpensive. A single standard sheet typically contains eighteen rows of bar code—the equivalent of both tracks of one magnetic card. This is comparable to the amount of space taken up by the printed listing of the program. It is our hope that in the future wherever calculator programs are printed in listing form, the bar code for the programs will also be printed (e.g., in textbooks, technical journal articles, newsletters, etc.).

Data bar codes make another important application possible. A number of large organizations—corporations and

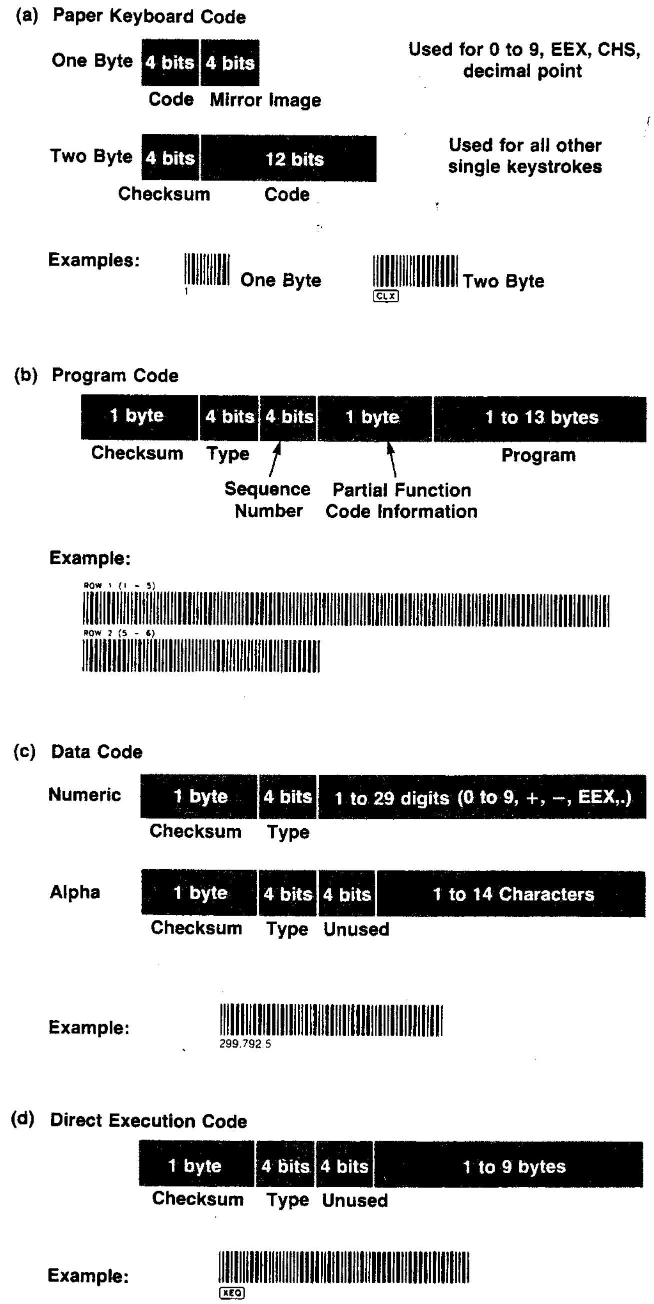


Fig. 4. *Four logical types of bar code are used by the HP-41C. (a) Paper keyboard code. (b) Program code. (c) Data code. (d) Direct execution code.*

government agencies—have standard sets of calculator programs developed for in-house use. Frequently these programs use data that is changed periodically. In such cases, the data can be printed and disseminated in bar code form.

The most novel applications for the 82153A Wand make use of the opportunity to mix machine-readable bar code with human-readable text in formats specifically adapted to the problem to be solved. The advantage in this sort of application is that the legends on the bar code can be written in the natural terms of the problem, and need not bear any resemblance to the technical meaning of the bar code to

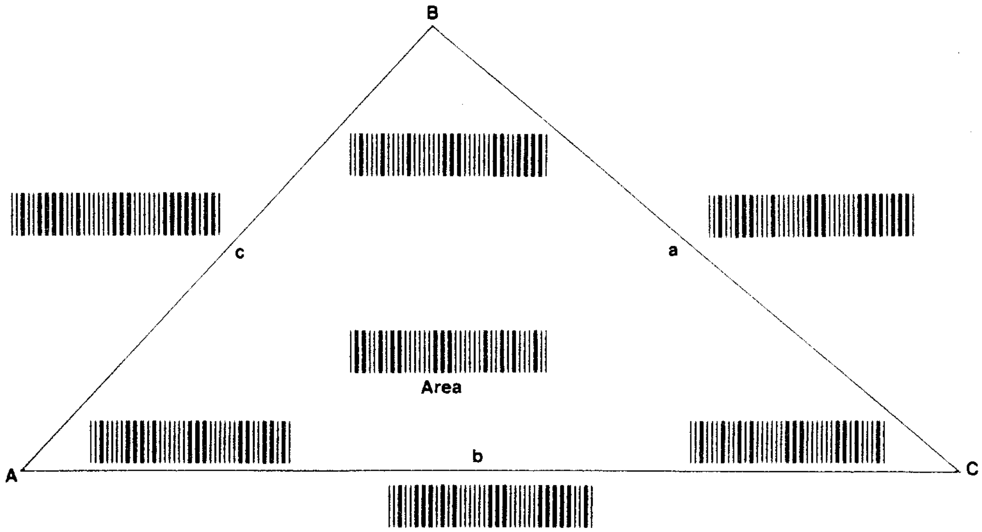


Fig. 5. *Problems can be easily solved by using bar codes to enter known values and to execute routines to determine the unknown value. An example for finding an unknown side or angle of a triangle is shown to the left.*

the calculator. In Fig. 5, this technique is used for an easy-to-use program for solving triangle problems. The user enters the known data about the triangle, indicating which part of the triangle it applies to by scanning the corresponding bar-code row. When all the known data is entered, the user queries the calculator for an unknown value simply by scanning the bar code for that part. The user never has to know that the bar-code rows on the diagram actually mean XEQ A, XEQ B, etc.

The paper-keyboard type of bar code derives its name from a paper keyboard provided in the box with the wand. The HP-41C calculator has available many more functions than keys, so a user occasionally must spell out the name of a function to invoke it from the keyboard. The paper keyboard, however, has bar-code rows for every function in the HP-41C catalog, and for the card reader, printer, and wand functions as well. A single sweep of the wand can replace a half dozen or more keystrokes when the user wants to invoke a function not currently assigned to a key.

During the project, it became clear to the design team that we would not be able to anticipate all the uses of the HP-41C/82153A Wand/bar code system. For this reason, a function (WNDSCN) was added that allows the HP-41C to read any bar code meeting the requirements of the interface chip. The data from the bar-code row is placed into registers 01 through n, one byte in each register. The WNDSCN function does not perform a checksum or any other consistency test. A submarine hunt game that makes use of this function is included in the owner's manual.

Acknowledgments

Bernie Musch conceived the product and provided inspiration and continuity for the project. Rich Whicker and Dave Lowe did the initial hardware design. Alan Peterson and Sheshadri Iyengar helped complete hardware development. Gaye Daniels, Steve Chou, and John Van Boxtel created the microcode. Jerry Hackett is the production engineer. Eric Henderson wrote the owner's manual. Tycho Howle was product manager. Other important members of the project team include: John Allen, Bob Dunlap, Earle Ellis, Alan Gill, Steve Gregg, Don Hale, and Judy Thompson. Special thanks goes to the PPC (a user's group) for help in testing the product, especially to Jake Schwartz for suggestions leading to the paper keyboard layout. The triangle program in Fig. 4 was adapted by Pam Raby from John Kennedy's program for the HP-67.

Thomas L. Revere III

Tom Revere joined HP in 1975 after working in the semiconductor industry for four years. He initially worked on IC design and is currently involved with continuing development of the HP-41C product system. Tom is a native of Mobile, Alabama and attended Brigham Young University, where he was awarded both the BESEE and MEEE degrees in 1971. He is a member of the IEEE and he, his wife, and five children live in Corvallis, Oregon. Tom's outside interests include home computing, camping, photography, and collecting science-fiction books.

David R. Conklin

Dave Conklin was born in Washington, D.C. and attended both the University of California at Berkeley, earning a BA degree in mathematics in 1967, and the University of Santa Clara, earning an MSCS degree in 1975. He worked for HP from 1973 to 1975 at the Santa Clara Division and joined the company's Corvallis Division in 1977 after working for a time elsewhere on process control programming. Dave has worked on the microprogramming for the HP-41C Calculator and is the project manager for the 82153A Wand. He is a member of the IEEE and the Association for Computing Machinery. Dave and his wife live in Monroe, Oregon.