VOL. 2 No. 2 February 1985

COMPUTER DIGEST

NEW KIND OF MAGAZINE FOR ELECTRONICS PROFESSIONALS

SEIKO'S NEW WRIST-WATCH COMPUTER SYSTEM

An in-depth look at the world's first wrist-watch that forms part of a complete computer system.

HIGH-RESOLUTION MONITORS

How to unscramble all those specifications.

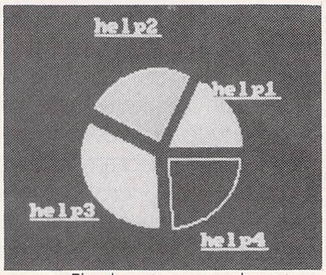
A GERNSBACK PUBLICATION

RECEIVING COMPUTER PROGRAMS VIA BROADCAST RADIO

Standards are being proposed that will enable you to receive computer programs from your radio receiver.

CONTENTS

Vol. 2 No. 2 February 1985



Find out more about high-resolution displays, low-resolution displays, and everything in between. The story begins on **page 13.**

The conclusion of Machine Code Development System for the Timex Sinclair *1000* will appear in the March issue of Computer Digest.

ON THE COVER

Just about everyone knows about Dick Tracy's wrist radio. Well, a new innovation from the people of Seiko has left that once speculative device in the dust. The innovation is a tiny "wrist computer" complete with an LCD wrist-watch-like readout, a pocket-sized keyboard, and 2K of RAM. **See page 7.**

ComputerDigest is published monthly as an insert in Radio-Electronics magazine by Gernsback Publications, Inc., 200 Park Avenue South, New York, NY 10003. Second-Class Postage Paid at New York, N.Y. and additional mailing offices.

LETTERS

NO COMMODORE PRINTER PROBLEMS

Our company, Martec Associates, Inc., sells a dot-matrix printer together with the Cardco, Inc. Card + interface for the Commodore 64 and VIC 20 computers. Naturally, we were very interested by Herb Friedman's article in the September issue of **ComputerDigest**, "Printer Delay for the Commodore 64."

A phone call to Cardco yielded the following information. In 1983, Cardco changed their ROM due to a problem like the one described in the article. However, since then, they have not had any problems. Is it possible that Herb Friedman used an old Cardco CARD?—Arthur Kingsnorth, Vice President, Martec Associates, Elk Grove Village, Illinois

Although both the Commodore computer and Cardco interface were bought shortly before the article was prepared, it is possible that the Cardco interface was sitting around in stock for some time and therefore contained the old ROM. The Customer Service Department at Cardco assures us that all problems have been corrected with the printer adapter. If anyone runs into a situation like the one Herb Friedman did, Cardco will gladly replace the defective interface if it is sent to the Customer Service Department at Cardco (300 S. Topeka, Wichita, KS 67202) along with proof of purchase and a description of the problem.

TI COMPUTERS

You say that you're not "just another computer magazine." Well, so far all you do is talk about the same things that the other computer magazines do.

I have a TI 99/4A, and I haven't seen anything written about it in **ComputerDigest**. For that matter, I haven't seen much written about it in any of the other magazines. That is one of the reasons that I'm writing this letter. I would like to know how to make a phone modem, and a few other things for my computer. I'm sure that there are many more people like me that have a TI 99/4 computer.

I enjoy **Radio-Electronics,** but if **ComputerDigest** is going to be like the rest of the computer magazines, then we can do without it.—Brian E. Sparling, Northfield, IL

If you're a TI owner and think you have an article for publication, consider this an open invitation for you to let us know about it.

PUBLISH YOUR PROGRAMS

The Blacksburg Group writes and produces books about using small computers. Since 1977, we have developed over 60 titles on electronics and computer subjects, and our books are published by major U.S. publishers.

One of our new projects is the collection and publication of useful engineering and scientific routines and subroutines. We know that many people have written interesting programs to solve a specific problem or because they couldn't find a special routine they needed. Our book gives these people a way to share their programs with other scientists and engineers so that others can benefit.

Useful programs include those that do graphing, numerical analysis, statistics, equation solving, 3-D plotting, controlling real-time clocks, controlling analog converters, and so on. We are interested in almost *all* programs that could be used by scientists and engineers to answer specific or general needs. We know from

COMPUTER DIGEST

Hugo Gernsback (1884-1967) founder
M. Harvey Gernsback, editor-in-chief
Larry Steckler, CET, publisher
Art Kleiman, editorial director
Byron G. Wels, editor
Brian C. Fenton, technical editor
Carl Laron, associate editor
Robert A. Young, assistant editor
Ruby M. Yee, production manager
Robert A. W. Lowndes, production associate
Dianne Osias, production assistant
Karen Tucker, production assistant
Jacqueline P. Weaver, circulation director
Arline R. Fishman, advertising coordinator

Gernsback Publications, Inc.
200 Park Ave. South
New York, NY 10003
Chairman of the Board: M. Harvey Gernsback
President: Larry Steckler

ADVERTISING SALES 212-777-6400
Larry Steckler
Publisher

EAST/SOUTHEAST
Stanley Levitan
Radio-Electronics
200 Park Ave. South
New York, NY 10003
212-777-6400

MIDWEST/Texas/Arkansas/Okla.
Ralph Bergen
Radio-Electronics
540 Frontage Road—Suite 325
Northfield, Illinois 60093
312-446-1444

PACIFIC COAST
Mountain States
Marvin Green
Radio-Electronics
15335 Morrison St., Suite 227,
Sherman Oaks, CA 91403
818-986-2001

LETTERS *continued*

personal experience that most people don't want to become professional programmers just to be able to use a computer, so we strongly feel that a collection of useful program segments will be invaluable.

If readers are interested, they should write or call us first, and we will send them guidelines and other information about submitting a program or subroutine. Our number is 703-951-9030 (which is in the Eastern time zone).

Right now, we're particularly interested in BASIC-language programs for popular desktop computers. This is NOT a vanity publication, and authors retain the copyrights to their material.—Jonathan A. Titus, Ph.D, President, The Blacksburg Group, Inc. PO Box 242, Blacksburg, VA 24060

Thanks for the information—we're happy to pass it along to our readers.

Competition

I'd like to make what I think is an important point that we'd all better understand. I know nothing at all about spectator sports, and could not care less. I think that it was for that reason alone that my young son developed his own interest, and learned to quote the batting averages of every football player in the major leagues. It gave him the opportunity to know something that I didn't. He once challenged me to ask him any question about baseball, so I asked him how far it was between first base and home plate, and he became angry. "What kind of question is *that?*"

Being an old electronics nut myself, he shied away from that field until he realized that my knowledge of computers was miniscule. He jumped on that too. He became an overnight computer "expert" and kept asking questions of a technical nature that he already knew the answers to. I quickly caught on to the game, and began to seriously read—and wonder of wonders—to understand, thanks to

ComputerDigest.

Things are beginning to settle down now around the house. We're starting to learn more together, and we're enjoying it more all of the time. Just thought you'd like to know. Keep up the good work.—J.P., San Francisco, CA.

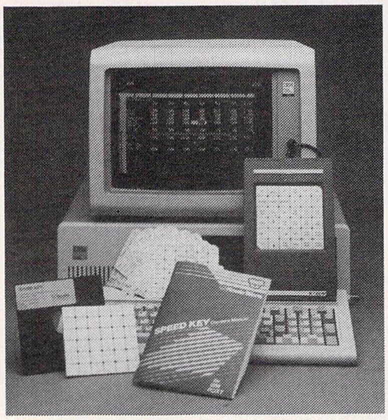
COMPUTER PRODUCTS

For more details use the free information card inside the back cover

PRODUCTIVITY TOOL, *Speed Key,* is designed for business software programs and languages. It makes them easier to learn and to be customized to each user's special needs. *Speed Key* supports the following business-application programs: *Lotus 1-2-3, WordStar, MultiPlan, SuperCalc, VisiCalc, dBASE II,* and IBM's DOS and BASIC programming language.

With *Speed Key* and Koala's touch tablet, a user can bypass the standard computer keyboard to run the leading business-software programs. *Speed Key* converts the Koala touch tablet into a custom keyboard with up to 37 programmable function keys and a cursor controller with the features of a "mouse" pointing device.

Each *Speed Key* overlay has 36 squares, with each square representing a softkey designed to give specific



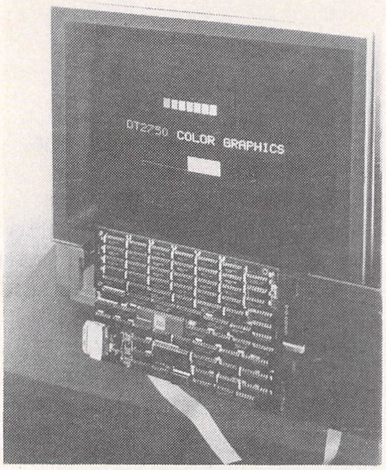
CIRCLE 21 ON FREE INFORMATION CARD

instructions to the IBM PC or IBM PC-XT. When a softkey on the overlay is pressed, the computer and application software will think that the user is typing on the computer keyboard. Thus, the user does not have to remember lengthy commands and multiple keystrokes required by conventional keyboards.

Speed Key has a suggested retail price of $99.00.—**Koala Technologies Corp.,** 3100 Patrick Henry Drive, Santa Clara, CA 95052-8100.

GRAPHICS SUBSYSTEM, the model *DT2750,* is a full-color raster subsystem that allows any Q-Bus processor to generate and display mixed graphics and alphanumerics on a monochrome or standard RGB color monitor.

The model *DT2750* contains a 192K graphics-display memory that organized as two independent display buffers with 512 × 512 × 3 pixels each. The framesize may be jumpered to 512 × 156 × 3 pixels. The 3-bit depth permits display of eight colors at one



CIRCLE 22 ON FREE INFORMATION CARD

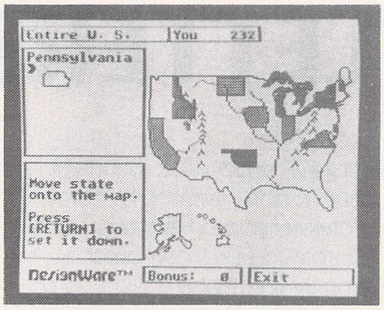
time, selected via a lookup table from a palette of 512 colors. When used with a monochrome monitor, there are eight levels of gray scale available.

Applications for the model *DT1750* graphics board are found in process monitoring and control, medical electronics, business graphics, and visual network control. It is priced at $1995.00.—**Data Translation,** 100 Locke Drive, Marlboro, MA 01752.

EDUCATIONAL SOFTWARE GAME, *States & Traits,* challenges families and children ages nine and above to sharpen their knowledge of United States geography, history, and current trivia.

The map-maker/user has two options: In "states," he or she charts states into their proper locations on a colorful master map. In "traits," the cartographer's challenge is to plot topographical features into their correct geographic positions and to answer questions on U.S. landmarks, history, and trivia.

Players can choose to test their geographic knowledge of the whole United States, or to concentrate on one of four regions. If the player selects "states," the labeled outline of a state appears on the screen next to a map of the total U.S. or region. Using a joystick or keyboard, the player then leads the state to its correct position

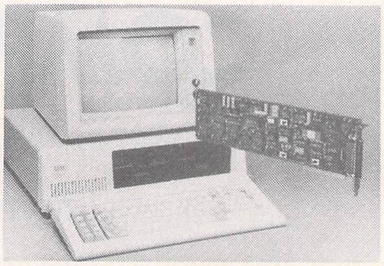


CIRCLE 23 ON FREE INFORMATION CARD

on the map. The state appears to "march" across the country as the player moves and charts its course.

In the "traits" portion of the game, a map and a question appear on the screen. State capitals, neighboring states, historical facts, and current events are all fair game. To answer, the user draws an arrow to the correct state. For example: "What state claims fame as the boyhood home of Abraham Lincoln?" The player must point to the state of Illinois. *States & Traits* is priced at $44.95.—**Designware,** 185 Berry Street, San Francisco, CA 94107.

DATA-ACQUISITION BOARD, the *DASH-16,* is a high-speed, plug-in data-acquisition board for the IBM *PC* and other bus-compatible computers.



CIRCLE 24 ON FREE INFORMATION CARD

DASH-16 provdes 12-bit A/D conversion and speeds up to 40,000 samples per second, with transfer to memory at that speed using DMA (level 2 or 3). Sixteen single-ended or eight differential analog input channels are available (switch selectable), as is an instrumentation amplifier with switch-selectable gains of 0.5, 1, 2, 5, 10, and a special "user gain" for specific application gain requirements. Data conversions may be initiated by the program, an internal timer, or by an external trigger. Converted data may be transfered by program interrupt or DMA. The interrupt and DMA modes support background operation. Input-voltage range is ±10 to ±0.5 volts.

The *DASH-16* is priced at $895.00.—**MetraByte Corp.** 254 Tosca Drive, Stoughton, MA 02072.

SEIKO'S NEW WRIST COMPUTER

It had to happen, and Seiko did it... Here's a look at the new Wrist Computer!

Marc Stern

■If you've been following Dick Tracy through the years, you've probably seen him talking to Sam, his sidekick, on his wrist radio. Well, today that can be updated. Instead of talking to Sam on his wrist radio, Dick Tracy will probably use his wrist computer to exchange information with him.

Although that may sound like science fiction, it really isn't because of a recent development from Seiko, the people who normally bring you high-quality watches and mini-televisions, among other things.

Seiko does it

The Seiko *Datagraph* system, which consists of a wrist module, pocket keyboard, and controller is the result of several years of development by the Japanese electronics giant Hattori Corp. It relies on large scale integration (LSI) techniques and the first use of inductive wireless transmission technology in the computer industry.

When you first look at the wrist module, you'll notice that it isn't especially unique-looking. In fact, it looks just like a watch (which is what you'd expect from Seiko). It's what the company has done with technology and how they have managed to squeeze some computing power into a wrist-sized package that sets it apart from all the other wrist watches of the world.

The wrist module

The basic component of the information system is the wrist module. It contains five CMOS LSI IC's that include a four-bit central processing unit; a 2K RAM, and three display drivers for the liquid-crystal display. A block diagram of the wrist module is shown in Fig. 1.

The LCD readout consists of a 10-column by four-row matrix that has a resolution of 1,400-pixels. The LCD not only serves as the display for the computer, but—as you would expect of a watch manufacturer—it also displays all the timekeeping functions, including day, date, chronograph, and alarm. It is powered by a lithium battery that Seiko claims will supply power to the wrist module for 1.5 years.

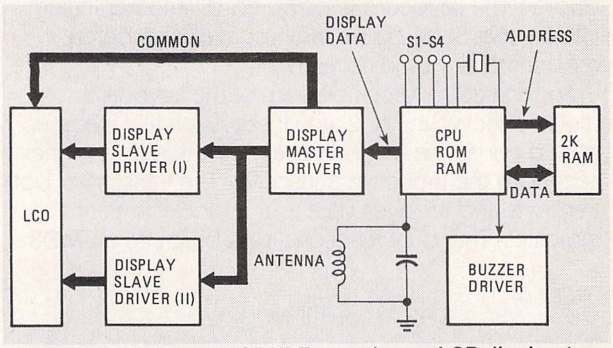


FIG. 1—THE DISPLAY MODULE contains an LCD display, two display drivers, a master display driver, CPU and 2K of RAM, a loop antenna and a buzzer.

The keyboard module

Data is entered into the wrist module by a pocket-sized keyboard unit. The keyboard measures only 5½ × 2¼ × 5⁄16 inches, and contains 61-keys through which data is input to the wrist module. The wrist module itself has only four buttons that are used to access pre-programmed functions. So, as you can see, the keyboard is an important device. You can use the keyboard unit to input notes, calculations, etc. into the wrist module. You can even store telephone numbers, appointments, and just about anything else in the device.

Data is transferred from the keyboard unit to the

THIS IS NOT a wrist watch! It is the display module for the Seiko *Datagraph* computer.

wrist module through induction. The wrist module fits on a small plate at the left of the keyboard and communicates with the controller at a rate of 2048 baud (bits per second). The communications is duplex. The keyboard is powered by another lithium battery that Seiko claims will last five years before replacement will be necessary.

Looking at the block diagram of the keyboard module shown in Fig. 2, you'll see it is really made up of two parts. The first is the keyboard circuit and the second is the inductive transceiver The transceiver both transmits and receives data as the block diagram indicates. That dual functionality is built into a CMOS

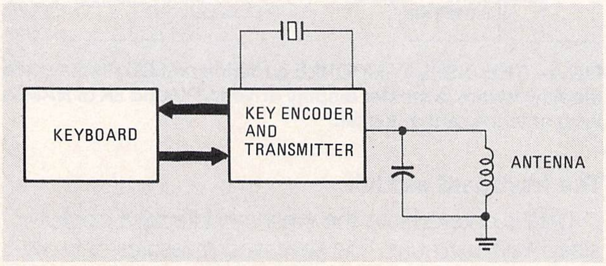
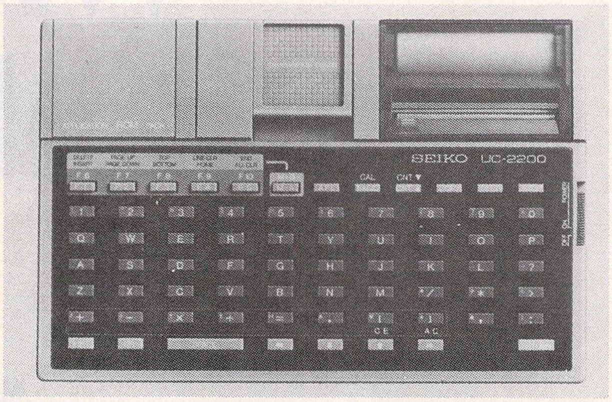


FIG. 2—THE KEYBOARD MODULE contains the keyboard, a keyboard encoder and transmitter, and a loop antenna.

LSI IC that handles not only key scan, but also character generation, as well as wireless transmission. It runs at a clock rate of 32.768 kHz.

Master control

Rounding out the *Datagraph* system is the module that turns this system into a true 8-bit microcomputer, the *UC-2200* controller. Communicating inductively, the controller uses the wrist module as its display device. As shown in Fig. 3, the *UC-2200* controller contains an 8-bit CMOS Z80-equivalent microprocessor. It boasts not only a complete, typewriter-style QWERTY keyboard with function keys, but also a dot-matrix mini-printer. The controller also features 4K of RAM



THE *UC-2200* CONTROLLER turns the display module into an 8-bit micro-computer. The controller is shown here with an application ROM pack in place.

memory and a ROM applications pack that includes scheduling, a 26K BASIC interpreter, as well as other programs. The ROM can contain as much as 32K. The entire unit measures only 5 × 7-inches and it can be powered by alkaline batteries.

The importance of the Seiko *Datagraph* system lies equally in its size and capability, as well as its method of transferring data from one device to another. It is the first system in the industry to employ inductive transmission and reception techniques.

It brings major computer capability to a unit whose key part—the wrist module—is intended to be worn

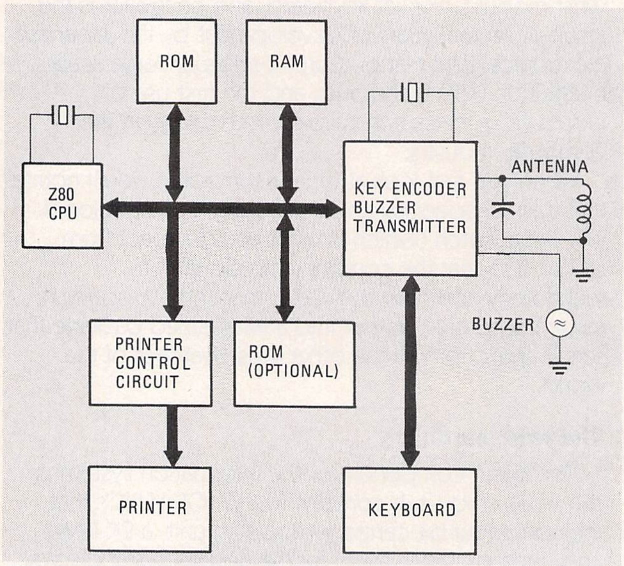


FIG. 3—THE CONTROLLER MODULE contains a CPU, RAM, ROM, an optional ROM containing applications software, a keyboard encoder, buzzer driver and transmitter, a keyboard, printer controller, printer, buzzer, and loop antenna.

on your your wrist. And, the system itself is small enough that it can be easily used by the "man on the go." Ten years ago, when the first microcomputers were making their appearance, it was impossible to have a computer at your fingertips wherever you went. Of course, that situation changed about four years ago with the introduction of such transportable computers as the Osborne I and the Kaypro. But, let's face it, those computers were far from conveniently sized for hauling. Briefcase computers did put true computing power at your fingertips wherever you went, but they were still fairly large. However, the Seiko *Datagraph* system now puts computing power on your *wrist*—a very convenient package.

Inductive coupling

Rather than relying on traditional radio transmission techniques, Seiko chose to use inductive coupling to transfer data between the various modules. Figure 4 shows the basic approach.

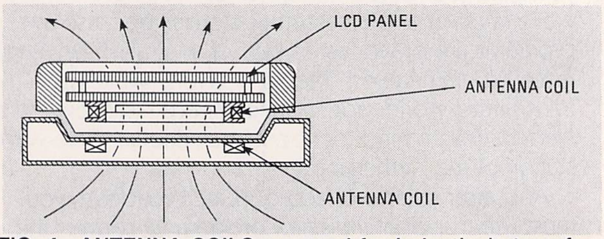


FIG. 4—ANTENNA COILS are used for inductively transferring data between the various modules.

Each module contains an antenna coil that resonates at 32 kHz. That frequency was chosen because the other clock frequencies contained within each of the modules could be easily filtered out. The bit stream modulates a 32-kHz signal that is applied to the transmitting antenna. If the bit is at a logic 1 level, then the 32-kHz signal is applied to the antenna, if the bit is at a logic 0 level, then no signal is applied to to the transmitting antenna. The magnetic field produced by the transmitting antenna cuts across the windings in the receiving antenna. The receiving antenna produces a current in response to the magnetic field. The output of the receiving antenna is an exact duplicate of the original signal applied to the transmitting antenna. The 32-kHz signal is filtered out of the received signal and the original bit stream is recovered.

An eight-bit digital word is transferred in very much the same format that is used when two computers communicate via a modem. The actual transfer of the eight-bit digital word consists of a start bit, followed by eight data bits, a parity bit and finally, a stop bit.

Seiko chose inductive technology for several reasons. First, it could keep everything inside a hermetically sealed unit so it is humidity resistant. Second, it helps to keep the unit portable because of its small size. Next, the company chose this method because of the ease of interfacing the units. There are no cables to worry about and it allows duplex communication through one loop. Last, but not least, the technique is easy to implement and since the circuitry needed to accomplish it is simple, few parts are needed. In fact, the data transmission and receiving circuits can be connected to one antenna coil.

Packaging

None of this would have been possible without developments in LSI packaging. For instance, all parts of the transmitting-receiving circuits, other than the coils and tuning capacitors, are located on a single integrated circuit.

Look at Fig. 5 and you'll see how the wrist module is put together. The main substrate—epoxy glass resin—contains three LSI packages. In turn, a ceramic substrate, which contains the wrist module's 4-bit CPU and 2K RAM, is soldered to the rear. In other words,

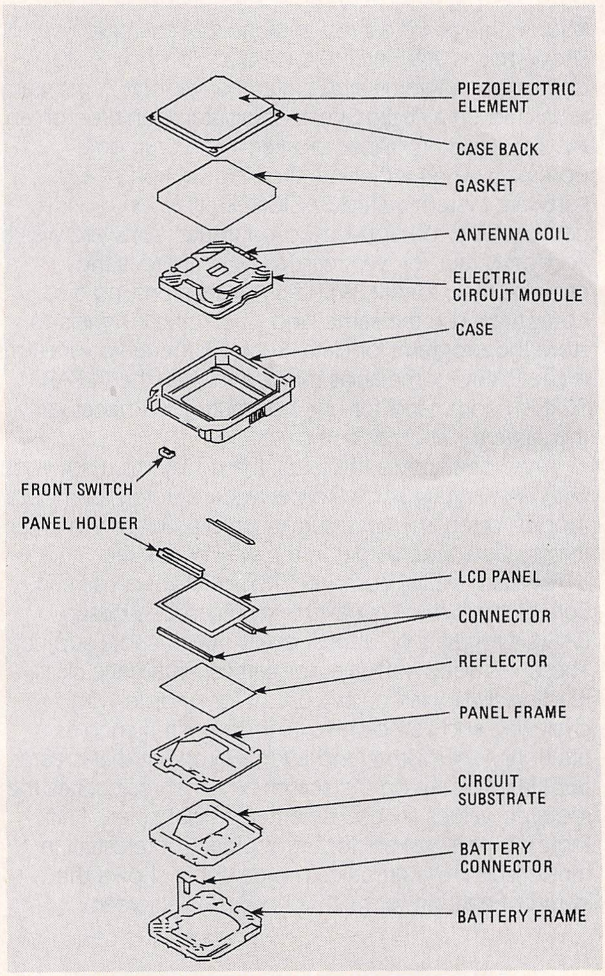


FIG. 5—CONSTRUCTION OF THE WRIST MODULE.

two substrates contain the five LSI chips needed for the entire module. This was made possible by a breakthrough that allowed Seiko to combine two display and control functions onto an individual substrate.

Even the pickup loops have been carefully merged into this plan. They are wound around the battery frame, eliminating the need for extra space.

With all this, the day of the wrist information system has dawned and just like Dick Tracy, we can now have our own "wrist radios," but they're really computers, of course.

BASICODE

You can get free software via shortwave radio.

HERB FRIEDMAN

■Some things are carved in stone. For example, we all know that as a general rule, cassette-based BASIC computer programs aren't interchangeable. A program recorded on a Radio Shack computer won't run on an Apple, while an Apple program won't run on a Commodore 64, etc. Until the introduction of a software system called BASICODE, the non-compatibility of tape-based computer software was an accepted fact for two reasons: First, there is the cassette tape format. With few exceptions, no two computers use the same kind of electrical signals to store the programs or data. Second, there are variations in the BASIC commands themselves; ie., the CLEAR SCREEN command for one computer isn't necessarily the same for another computer.

When one knows the facts, the problems associated with exchanging software between different computer models often appear insurmountable. But as with many things, facts tend to get in the way of real life.

For many years, computer hobbyists and users in Europe have routinely exchanged cassette-based BASIC programs on almost every imagineable subject. These included software for games, arithmetic skills, reading skills (particularly useful for people with dyslexia), and some rather high-level stuff such as titrations (chemistry), and even a program that creates a 555 timer circuit on the screen and then calculates the required values for user-selected frequencies. And there's even software for computerists interested in music, such as a graphics program that shows the correct fingering for guitar chords. (The screen photographs that appear later in this article illustrate two of the typical exhange programs.)

"Aha!," you say, "This is too good to be true. There must be a catch!" Yes, there is a catch. Here in the U.S. you can't exchange software unless you have a short-wave receiver and a cassette recorder, because the programs are broadcast as part of an English language radio program called "Media Network,"which is transmitted worldwide via short-wave radio by *Radio Nederlands*, the independent international short-wave station of the Netherlands.

And after you have recorded the broadcast, you must use a special *translator* program to convert the broadcast into the hardware and software format required by your computer.

Difficult? No. It only appears to be difficult. For most commonly-used computers, it's as simple as pressing the PLAY button on the cassette machine. The translator program—which is available for many popular computers—works on two distinct levels; hardware and software. It processes the received electrical signals into the format required by your computer, and then interprets the handful of non-standard BASIC statements into the format required by your computer's particular version of BASIC.

Hobbyscoop

The translator software is part of a system called NOS-BASICODE, which was developed by Dutch and other European hobbyists for the "Hobbyscoop" (Hobbyscope) radio program NOS—the Dutch

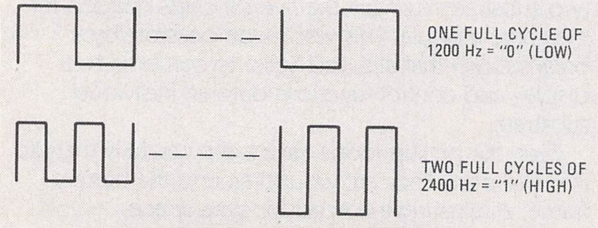


FIG. 1—REPRESENTING ONE FULL CYCLE of 1200 Hz, is a "0" (low), while a "1" (high) is represented by two full cycles of 2400 Hz. A byte of data uses standard teletypewriter format of 1 start bit (logic 0), 8 data bits (logic 1), 2 stop bits (logic 1).

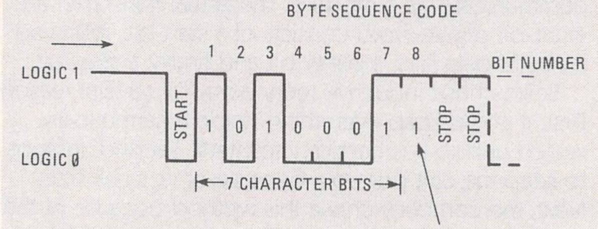


FIG. 2—THE LETTER 'E" shows the byte sequence code. You'll find a start bit, seven character bits, the eighth bit and two start bits, a total of eleven bits.

Broadcasting Corporation. Hobbyscoop features news of new and unusual developments in electronics which would be of interest to hobbyists. When personal computers came along in 1977, it was natural for Hobbyscoop to cover the subject and to broadcast software for the few home/hobby computers available at that time.

Originally, Hobbyscoop broadcast indvidual computer programs. A single program for three different computers required three separate broadcasts. As more computer models were introduced, the amount of time needed to broadcast a single program for the various computers became burdensome, and so a "universal language" called BASICODE, which incorporated a *translator*, was developed. The purpose of the BASICODE system was to use a single broadcast to transmit a program to many different computers, all having different electrical and software requirements. Eventually, BASICODE would translate for 20 different computers.

The broadest tones

Transmitted at 1200 baud, the tone frequencies of 1200 Hz and 2400 Hz are used to broadcast the software to the receiver/cassette recorder. As shown in Figure 1, a "0" (low) is represented by one full cycle of 1200 Hz, while a "1" (high) is represented by two full cycles of 2400 Hz. A byte of data uses the standard teletypewriter format of 1 start bit (logic 0), 8 data bits (logic 1), and 2 stop bits (logic 1).

All characters are represented in ASCII (American Standard Code for Information Interchange). Since only seven bits are used to represent an ASCII character, the eighth bit is always automatically set high (1). The complete character consists of a start bit, seven character bits, the eighth bit, and two start bits. A total of eleven bits; exactly the same format as used for a standard 110 baud teletypewriter. Figure 2 shows how the character "E" would be transmitted. Figure 3 shows the tone sequence used for the actual broadcast transmission, which is eventually recorded by the user. The complete off-the-air signal consists of: Leader, 5 seconds of 2400 Hz; Start text, ASCII "Start Text" (Hex 82); Program, BASIC in ASCII; Checksum, and Trailer, 5 seconds of 2400Hz. The checksum at the end is derived from the bit indication of exclusive-OR of the previous bytes, and is expressed as an 8-bit term. The purpose of the Checksum is to allow the user to test whether the program has been read from the tape without error. Even if the Checksum is not correct—indicating an error—BASICODE will load the program and permit a listing of what has been read. Since the program is in BASIC and not op code (binary), the user can correct the errors and then SAVE the program in the usual way.

The translator converts

Since no computer reads the the transmitted cassette format directly, the user must first load the translator program into his or her computer. (The translator is written in the precise software and hardware format required by the computer.) Then the BASICODE cassette program is loaded. The translator, which is already in the computer, causes the computer to read the BASICODE cassette signals, and then converts the BASICODE into the format required by the computer.

Obviously, not all BASICODE cassette programs can be loaded in the usual way because some of the cassette interfaces used in home computers are going to reject anything that isn't precisely formatted, and the translator can't do anything until the signals get into the computer.

Generally, if the computer refuses to load non-conforming cassette signals, a hardware accessory will get the received cassette signal into the computer. For example, while the BASICODE cassette tape can be fed directly into an Apple or Commodore computer, a TRS-80 requires a simple interface such as the one shown in Figure 4. On the other hand, an OSI *Model 1P* computer (an early hobbyist model) requires only the the addition of a three-way switch and a wire jumper, while for CP/M computers, the BASICODE signal is fed into what is normally the computer's parallel printer output.

A few computers require a somewhat extensive accessory interface. Circuits for those computers that require accessory interfaces are given in the BASICODE Handbook. We'll tell you how to get one later in this article.

Once the BASICODE program has been loaded into the computer and translated it can be SAVED as a standard cassette file—even as a disk file. As far as the computer is concerned, the translated program is a "standard" program for that particular computer and can be used on any similar computer without need for the translator.

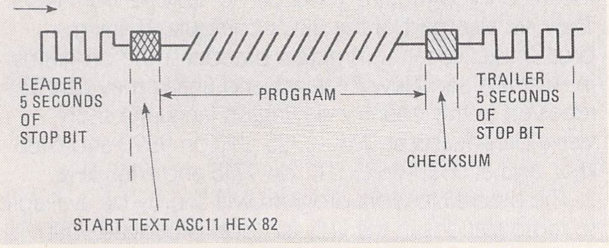


FIG. 3—THE TONE SEQUENCE used for the actual broadcast transmission, which is eventually recorded by the user.

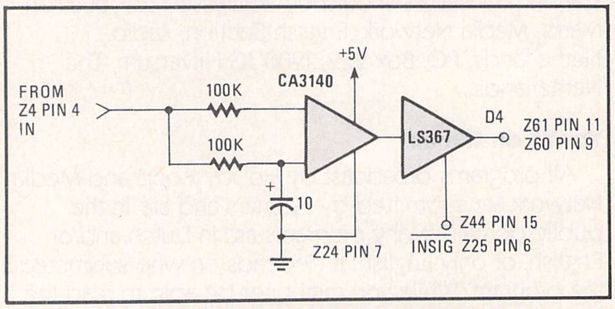


FIG. 4—A SAMPLE INTERFACE for the TRS Model I/III as shown in the BASICODE Handbook.

Universal basic

BASICODE is not an "...all bells and whistles" version of BASIC; instead, BASICODE is more like the high-performance *time-share* BASIC from the era when schools used teletypewriter terminals connected via the telephone system to mainframe computers. The supported statements in BASICODE are: ABS, AND, ASC, ATN, CHR$, COS, DATA, DIM, END, EXP, FOR, GOSUB (GO SUB), GOTO, IF, INPUT, INT, LEFT$, LEN, LET, LOG, MID$, NEXT, NOT, ON, OR, PRINT, READ, REM, RESTORE, RETURN, RIGHT$, RUN, SGN, SIN, SQR, STEP, STOP, TAB, TAN, THEN, TO, and VAL.

The translator automatically accommodates the various ideosyncracies of BASIC through special BASICODE software routines located in the reserved area of program lines 0-999. (Since lines 0-999 are reserved for the translator, user written code starts at line 1010.)

The program translation works this way: Assume you want to write a program that will be used by others having different computers and a BASICODE translator. You want your program to clear the screen first. Since the CLEAR SCREEN statement varies from computer to computer. A BASICODE program would not use the "normal" CLEAR SCREEN command for your computer, instead, the program would use the statement GOSUB 100.

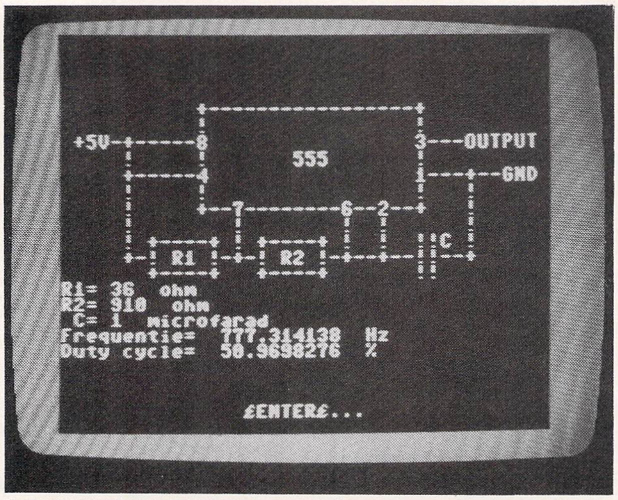
Each translator program has the correct CLEAR SCREEN routine for a specific computer at line 100. The GOSUB command sends the program to line 100 for the correct CLEAR SCREEN statement, and then returns to the program. Another example is a random number variable. That's a GOSUB 260, which the translator uses to generate the correct "random" statement for each computer. Actually, there's not more than a handful of conversions. Having lines 0-999 available insures that BASICODE has room to grow.

It's all in a kit

A kit consisting of the BASICODE handbook (with English translation) and an English-language cassette of translators for 17 popular or commonly-used computers (the ZX-81 is not one of them) is available (sent airmail) for f 38,- (f = Dutch guilders), payment in an international money order (IMO). The kit can be ordered from: BASICODE, Administratie Algemeen Secretariaat, NOS, P.O. Box 10, 1200 JB Hilversum, The Netherlands. Since the translators are updated periodically, information regarding the availability of specific translators should be addressed to: Jonathan Marks, Media Network, English Section, Radio Netherlands, P.O. Box 222, 1200 JG Hilversum, The Netherlands.

You're on the air

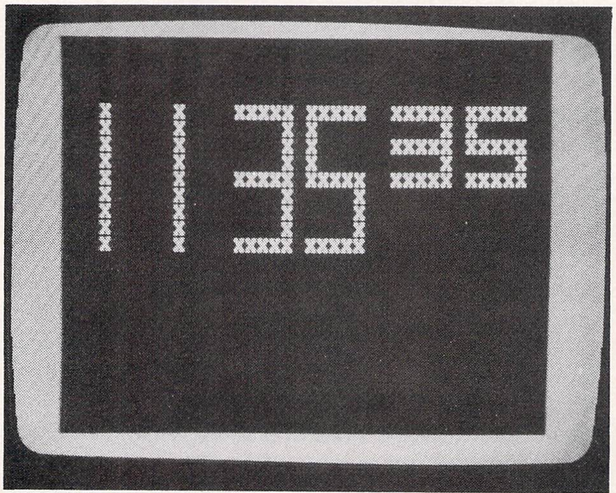
All programs broadcast by Hobbyscoop and Media Network are submitted by listeners and are in the public domain. Many programs are in Dutch and/or English, or only English. It depends on who submitted the program. While you might not be able to read the Dutch labels of non-English programs, you can certainly understand or figure out what's going on. You can also



A BASICODE PROGRAM created this schematic of a 555 timer and then asked for the desired frequency and duty cycle. In the blink of an eye, part values were displayed for the nearest standard values. The requested frequency was 800Hz.

rewrite the listed programs because the BASIC function statements—not necessarily the PRINT statements—are usually in English.

If you have some favorite programs you would like to share with others around the world you can submit them to Media Network, or Hobbyscoop for consideration. The BASICODE programs are presently



JUST FOR FUN, BASICODE has broadcast a program that creates a video clock.

received throughout a good part of Europe from NOS. They are also part of the BBC's computer-training broadcasts. Local transmitters broadcast the programs in Australia and New Zealand; and finally, they can be received in this country via English language short-wave broadcasts at 0230-0325 UTC on 9590 and 6165 kHz, and at 0530-0625 UTC on 9715 and 6165 kHz.

The Media Network program will shortly be available to radio stations in the U.S. for local broadcast, but until then, you'll have to get your "free" software via shortwave.

HIGH RESOLUTION COLOR MONITORS

There's more to computer-display resolution than meets the eye!

HERB FRIEDMAN

■Almost from the beginning of personal computing, we have referred to color displays in terms of resolution; there is low resolution, medium resolution, and high resolution. The individual picture elements that make up the display have been described in terms of dots, pixels, PELS, and total resolution of dots × lines.

What is resolution?

In fact, the picture resolution—meaning the number of individual picture elements that can be displayed—is relative to the existing technology: Yesterday's "high resolution" is today's "medium resolution" while today's "high resolution" is tomorrow's "low resolution."

Unlike a monochrome monitor, whose *apparent* visual sharpness depends almost entirely on the unit's bandwidth, color-monitor resolution is presently a combination of several variables, the most important being the size of the triad (a triad is made up of one red, one green, and one blue phosphor), the number of triads in a picture element, and the size of the displayed characters in terms of the number of active scanned lines. "Active lines" are those lines used for the computer display; it does not include the lines left unused to compensate for overscanning.

For multi-color reproduction, approximately 320 individual horizontal picture-elements per line is about the best we can do under normal circumstances. (That used to be considered *high-res*, but today, it's medium resolution.) The primary limitation on the number of horizontal elements is the size of a single dot of a character's matrix on standard color CRT's used for personal computer monitors, which translates into the maximum number of horizontal characters.

For example, characters for the so-called 80 × 25 screen are usually formed from an 8 × 8 dot matrix. In non-technical terms, it means each character is 7 dots high and 6 or 7 dots wide; the unused dots providing the spacing between characters and rows. Therefore, 80 columns of characters require 80 × 8 or 640 dots per horizontal line. A 40-character display requires 40 × 8 or 320 dot resolution.

It is the same thing going vertically. 25 lines requires 25 × 8 or 200 horizontal lines. (Aha! Now you see how screen resolution values are derived.)

Therefore, a high-res screen—80 characters × 25 lines—requires 640 × 200 resolution. A medium-res screen—40 characters × 25 lines—requires 320 × 200 resolution. Anything less, such as 32 characters × 16 lines is low resolution.

Standards

How did we come to more or less standardize on 32 × 16, 40 × 25, etc.? Mostly, to accommodate existing monitor equipment such as overscanned TV sets; sometimes because it's the most that can be safely put on the screen. To increase the size of the characters by going to a 9 × 9 or 10 × 10 matrix would reduce the number of columns or rows, or would require special monitors if the display wasn't to be "stretched" right off the top, bottom, and sides of the screen.

TV is a compromise

Until very recently, personal computer color displays were intended to be viewed on a TV receiver (through an RF modulator), or on a composite monitor originally intended for TV viewing; hence, the computer's color resolution was designed to function within the limitations of TV equipment, for which a resolution of 240 elements (dots) per line was—and still is—excellent. In relative terms, however, it is "low resolution."

Also, a computer's vertical resolution is designed for approximately 100–200 lines, depending on whether it is intended for viewing on a TV set or a video monitor. (For our discussion, let's assume 200 lines.) That creates an even greater visual conception of low resolution because of the larger-than-normal spacing between the lines that make up the display.

Half a frame

A standard TV picture uses interlaced scanning. The receiver first scans a field (1/60th of a second) of 262.5 lines. During the vertical retrace interval a series of equalizing pulses transmitted by the TV station delays the start of the next field's 262.5 lines so that they start—and fall—in the spaces between the previously scanned lines. The lines from the second field *interlace* with those of the first field. Because of the CRT's persistence, and the retentive characteristics of the eye, the two fields are combined by the brain into a single frame. Because some horizontal lines occur during the CRT's vertical retrace, the actual number of displayed field lines is less than 262.5.

That, of course, gives us the familiar 525 lines of a TV frame. But, due to overscanning, computer displays make use of less than half those lines.

TV transmitting standards call for a picture aspect ratio of 4 × 3 (width to height). But, because there is little picture information in the corners, and because manufacturers of TV sets want a TV picture to be "larger-than-life" even on a small-sized screen, most have designed their sets to "stretch" the picture so that it overscans the edges of the CRT. For TV, what's lost usually isn't missed, unless you're watching an old movie and happen to see someone talking to a nose, an eyebrow, or an ear.

But imagine what would happen to a computer display that tried to use the full screen: Part of the display would be lost in the overscan, while the display in the corners would be distorted—possibly beyond recognition.

Somewhere along, it was determined that 200 lines would not only accommodate 25 lines of 8 × 8 matrix characters, it would also fit on most monitors regardless of how much the display was stretched "to fill the screen." (Of course, now we can get ten scientific reasons why 200 lines is a "natural phenomenon throughout the universe," but it really all started as a way to insure that part of the computer display didn't flow off the top and bottom of the screen.)

When the computer produces the second field of 200 lines, it lays them down right over the first field; there is no interlace of the scanning lines, so the spacing between the lines is greater than it is for standard TV. That produces the illusion of reduced definition.

Another factor to be considered is reduced image convergence at the corners of a color display. Normally, at least in the lower-cost monochrome and color displays, there is a loss of focusing at the extreme sides and in the corners of the CRT. That added pincushioning and the misconvergence of the color guns at the circumference of the display, which under the best of circumstances is an error of 100%. The picture resolution gets pretty murky at the sides and in the corners.

Seeing all of it

In order to provide maximum edge-to-edge picture resolution and minimum distortion, some manufacturers

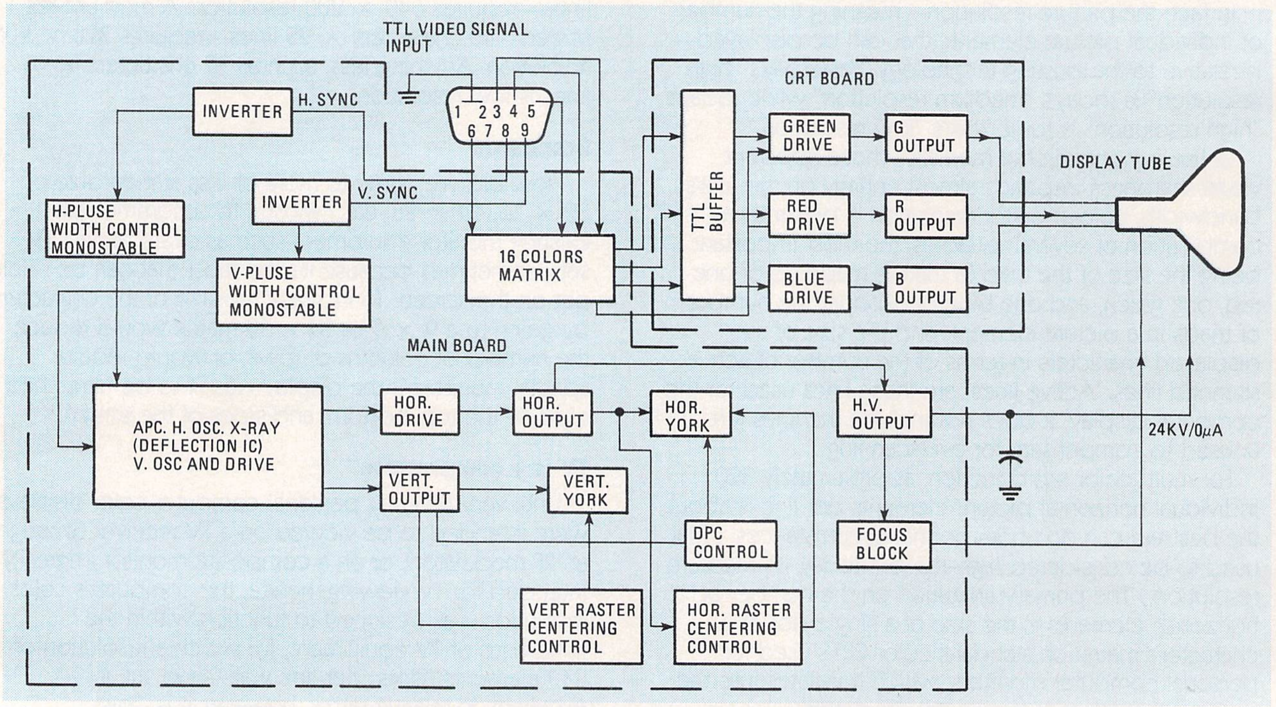


FIG. 1—IN THE RGB SYSTEM, the red, green blue and intensity data are fed out of the computer into individual inputs of the color monitor. A color matrix in the monitor combines the intensity level with the RGB signals so that 16, rather than 8 levels of color can be displayed.

"full frame" the display; that is, the monitor's scanning is adjusted so the actual corners of the display barely touches the sides of the CRT, creating a complete picture rectangle on the CRT with dark (no picture) areas at the top, bottom and sides. If you turn up the monitor's brightness or contrast controls, you will actually see a complete rectangle on the CRT, right down to the corners. (Try this on a modern Radio Shack or IBM monitor—you'll see why their displays are uniformly sharp from corner to corner.)

By keeping the color guns away from the circumference of the CRT, the focusing and overall convergence error is reduced. And while the display might be smaller in size than an overscan on the same CRT, the overall definition is higher.

By the way, the 200-line non-interlaced scanning is how one form of "highlighting" is created. Imagine for a moment a screen display of a word processing document. The normal character intensity is produced by scanning the characters for two fields per frame. For "highlighting"—actually reduced intensity—the highlighted characters are only scanned for one field. Simple but effective.

Good but not great

While the lower-cost computers—those usually called "home and family"—use TV sets or composite-video monitors for the computer's display, the business computers, almost without exception, provide for the so-called RGB monitor, even if they also have a composite video output connection. The problem with using composite monitors with computers is that they use analog video signals. While they can do a good job of creating computer displays, they are not the equal of an RGB monitor whose three color guns are controlled individually by the computer.

The RGB monitor

As shown in Figure 1—a block diagram of the RGB high-performance, high-resolution Princeton *SR-12* monitor—the computer outputs digital data for each individual color gun along with a digital signal representing intensity. A matrix interpolates the digital data into individual R, G, and B drive signals, which produce 8 colors: black, green, red, blue, brown, cyan, magenta and white. (There's no error in the preceeding. While RGB should produce yellow, not brown, in a high-performance monitor such as an IBM, a Quadram *HX-12*, or a Princeton *SR-12*, the base color is brown, and the intensity-modified brown is yellow.)

When the intensity bit is combined with the computer's RGB data in the monitor's color matrixing circuit, the resultant colors are: dark grey (intensified black), light blue, light green, light cyan, light red (which is the only red provided by some monitors), light magenta, yellow (a true yellow), and pure white (a super white). Altogether, 16 colors.

Tiny dots

But whether the monitor is mono or color, the average monitor is still 200 lines, so the apparent sharpness is determined by the size of each dot, and the smaller the dot, the sharper the apparent image.

High-resolution monitors appear to be sharper than conventional high-performance monitors because the dot size is smaller. For example, the Quadram *HX-12* and Princeton *SR-12* have a 0.31mm dot compared to IBM's 0.43mm dot size. Thus, the Quadram/Princeton characters appear sharper than those of the IBM monitor. Unfortunately, both Quadram and Princeton stretch the screen display to make the characters larger than they would appear when full-framed, which puts some of the display into the corners of the CRT—the last place you would want a high-resolution display because of reduced focus and convergence, and diagonal stretch caused by even minimal pincushioning.

400 lines

Within the limits of moderate cost, there is only so much resolution that can be attained from a conventional high-performance monitor. For a further increase in *apparent* resolution, we must still fill in the gaps in the display so the eye is tricked into believing the display has more information than is actually there. This is accomplished through a 400-line display.

Since the computer's normal character and graphics generators produce 200 lines per field, we obtain 400 lines by generating artificial lines of information. The "extra" lines are positioned directly between the normal scanning lines in the normal "interlace" location. The inherent "bloom" of each dot blends the lines together, and the eye sees a continuous character.

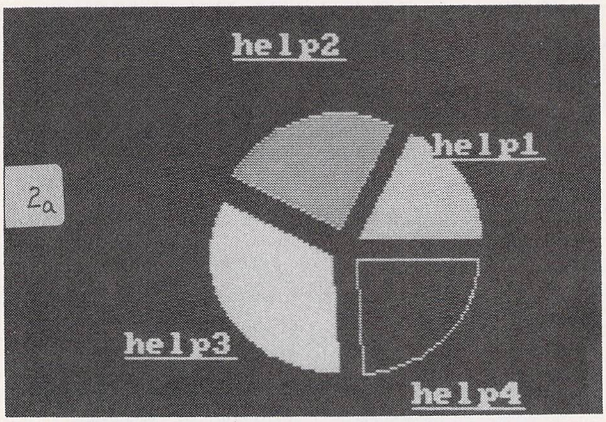


FIG. 2—A DISPLAY as seen on the screen of a Quadram *HX-12* 640 × 200 high-performance monitor shows that though the characters are "sharp as a tack" (for a color monitor) there is an illusion of unsharpness because of the space between the scanning lines.

Figures 2 and 3, photographs from two high-performance, moderate-cost monitors, show how it's done. Figure 2 is from a Quadram *HX-12*, one of the highest performance 200-line monitors: By any standards, it is sharp! Figure 3 is the same display from the Princeton *SR-12*. Note that no scanning lines can be seen. While the edges of the characters displayed on the *SR-12* are not as sharp as those of the *HX-12*, the *SR-12*'s *apparent* sharpness is greater because the normal scanning lines are "filled in."

A good question at this point, is "How can the *SR-12*

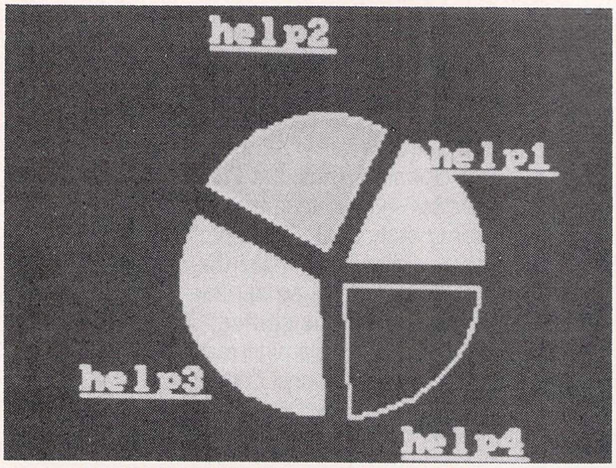


FIG. 3—THE SAME DISPLAY from a Princeton *SR12* 640 × 400 high-performance monitor shows no spaces because an additional 200 lines repeating the same data has been interleaved with the original 200 lines. While the edges of the characters are not as sharp as those on the 200-line monitor, visually the display appears sharper and seems to have greater resolution because there are no spaces between the scanning lines.

get sufficient time to scan two lines in the conventional horizontal scanning period?" By using a device known as a *scan doubler*—A plug-in adapter board for IBM-compatible computers (see Fig. 4).

It works this way. The horizontal scan rate of the *SR-12* monitor runs at a nominal 31.4 kHz, which is twice the conventional horizontal scanning rate of 15.75 kHz. During the conventional horizontal scan period, the *SR-12* scans two lines. But the computer still puts out its signal at 15.75 kHz: If fed directly to the *SR-12*, the single-line display would be spread over two lines and the screen would be a scramble of "garbage."

That's where the scan doubler comes in. Instead of the computer's RGB signal being fed to the monitor, it is fed to the scan doubler, which contains two memory banks and an output switcher. The computer's first line of data is stored in the scan doubler's first memory bank. When memory is full, its data is output twice to the *SR-12* at twice the normal horizontal scanning frequency, so two complete lines are scanned in the same time period of one standard scan; hence, the computer's first display line is scanned on two sequential monitor lines—scan line 1, and what we will call scan line R1—for repeat line 1.

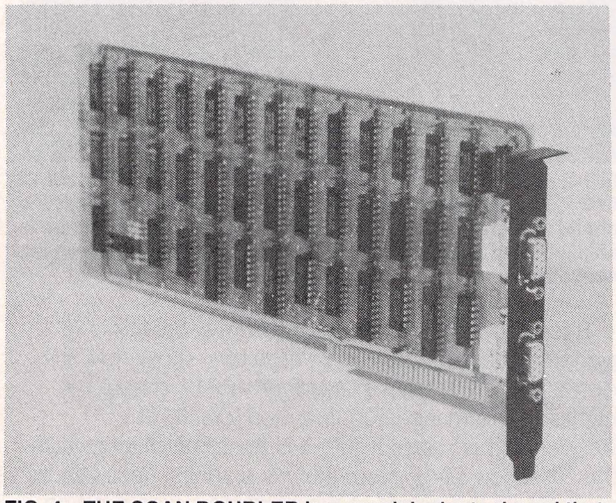


FIG. 4—THE SCAN DOUBLER is a special adapter board that generates 400 lines of display from what is normally a 200-line screen. The computer's normal signal is fed into the lower jack instead of to a monitor. The synthesized 400-line display exits at the top connector to the monitor.

While the monitor is scanning line 1 and R1—each of which corresponds to a single line of data from the computer—the scan doubler's second memory bank is being filled by the computer with the second line of data. When the monitor is finished scanning the first line of data for the second time, the scan doubler switches its output to its second memory bank, which provides the data for monitor lines 2 and R2.

As the scan doubler's second memory bank is being dumped into the monitor, the first memory bank is overwritten by the computer with the data for line 3 and R3.

By continuously switching its output between memories 1 and 2, the scan doubler creates a 400-line monitor display from only 200 lines of data.

The visual effect is spectacular. The display has no visible lines; it is continuous.

Voodoo resolution

Nothing stands still. If equipment has reached the stage beyond which there is no improvement in performance, then someone will invent improvements.

For reasons we have mentioned earlier, a business computer display is more or less standardized at a maximum of 640 dots × 200 lines in order to put everything on the screen within the area of maximum definition. Naturally, if it were possible to fill each complete horizontal line end-to-end, and if it were possible to use the unused vertical lines at the top and bottom of the screen, theoretically, the screen would resolve more dots. And that's precisely the assumption being used to create the illusion of even greater monitor resolution. Manufacturers are now starting to include the "phantom" data in the overall screen specifications, and we get claimed resolutions of 690 dots, 900 dots, 360 lines, 480 lines.

Yes, all that is possible, but not with conventional "business" computer equipment. Some specially-designed graphics and "text" adapters are capable of generating those seemingly unbelievable displays, but their common use is somewhere down the road. We'll cover them when they are generally available and cost-effective. Meanwhile, keep those *magic figures* in mind: *640 × 200 and 320 × 200*. For "medium resolution," multi-color displays, a 320 × 200 resolution is required. And though we call that "medium" resolution, it is actually high resolution for color, because the so-called high resolution display of 640 × 200 is used for only two colors, which happens to be monochrome (black-and-white is two colors, as is any monochrome display). Presently, within the constraints of moderate costs, we cannot generate higher multi-color resolution than approximately 320 × 200 lines unless we artificially create a 400-line display through the use of a scan doubler.