

CARD OF THE MONTH

CROMEMCO TV DAZZLER™

By Roger Edelson

This month I will cover a non-essential but fun card, the Cromemco TV Dazzler™. By now most of you have seen various computer displays, games and motion art and are probably interested in duplicating them. The Dazzler™ allows you to put graphics, games and animated shows on your T.V. in full color or in black and white for merely a little more than you would pay for a black and white terminal. The Dazzler™ costs only \$215 for the complete kit or \$350 for a fully assembled and tested version.

How is this accomplished? The Dazzler maps your computer memory or content onto your color T.V. screen. The device uses DMA (Direct Memory Access) to scan the computer memory. It then formats each memory bit into a point on the T.V. screen to give a 128 x 128 — element picture. Depending on the mode selected, later to be described, the Dazzler requires 512 or 2K bytes of memory — not an excessive amount.

Let us first take a look at the design of the Dazzler™. The unit is somewhat complex, using over 70 MOS and TTL integrated circuits on two printed circuit boards. The two Dazzler boards may be plugged directly into two adjacent slots in the Altair computer. A 16-conductor ribbon cable with DIP plugs on each end is used to interconnect the two boards. Board #2 may also be mounted "piggyback" on Board #1 so that only one slot of the Altair is required for the Dazzler. In this configuration seven additional jumper wires must be used to interconnect Board #1 with Board #2.

The functional block diagram of the Dazzler is shown in Figure 1. Board #2 is used to communicate with the computer and Board #1 generates the video output signal.

The Dazzler uses high-speed DMA to read the

memory of the host computer and translates the information in the memory into a T.V. signal. The key advantage of using DMA is that the Dazzler can display a picture simultaneously while the computer is executing either a related or an unrelated program. Dazzler causes only 15% slowdown in computer execution.

A 3.579545 MHz crystal oscillator is used to generate the color video subcarrier. The video output amplifier delivers a standard 1V negative sync NTSC video signal into a 52 ohm load. A six-foot length of 52 ohm coax cable is supplied with the unit.

The circuit diagram for Board #1 is shown in Figure 2. The oscillator uses a standard chroma crystal to generate the 3.579545 MHz signal. Three inverters and R-L phase-shifting components are used to provide the red and green signals. A group of NAND gates is used to mix the various signals to produce a video output. Vertical and horizontal sync are generated on Board #1.

While we are on the subject of the diagrams, I would like to digress to an editorial comment. I wish that Cromemco would put their parts identification on the schematics. They are *nowhere* to be found except in the board layout. Please, Cromemco; it would really help troubleshooting.

Figure 3 is the circuit of Board #2, the one used to communicate with the computer. This board uses a 1-Megabyte DMA channel with "handshaking" control, i.e. the Dazzler control issues a *Hold* command when it is ready to access the computer memory. When the computer finishes the current machine cycle, it issues a *Hold A knowledge* signal to begin DMA.

In addition to the high-speed DMA two 8-bit output ports of the computer are used to set the

I hereby propose the development of THE QUBE

First, an optically clear silicon *buole*, doped neutrally with a photosensitive dopant, is formed in the same way as for conventional chip construction. *Buole* is the term applied to the large mass of mono-crystalline silicon produced by 'drawing' silicon from a molten state. A cube is then cut from the *buole* and all sides of the cube are polished to optical quality.

The photosensitive dopant is sensitive only to light of two particular wavelengths rated above a certain candlepower. When illuminated above that threshold the dopant changes from neutral to positive or negative, depending upon the wavelength, and stabilizes, just as a normal photographic emulsion changes its molecular state in the presence of particular colors. Masks are prepared as if they were to be used for normal photo-etching on silicon chips.

Two laser beams are used in this method, each one capable of emitting the same two discrete wavelengths such as alpha and beta. Alpha and beta are the chosen wavelengths which turn the dopant to negative and positive, respectively. One beam is a collimated plane of light. It is a straight beam as wide as the cube but only a few millionths of an inch thick—about the same thickness as the epitaxial layers on a chip. The beam's cross section is a rectangle as wide as the cube and as high as a typical epitaxial layer. The second laser is a collimated beam whose width and thickness are both equal to the width of the silicon cube. The beam's cross section is a square.

The cube is placed on a stand with the wide beam illuminating it from above, perpendicular to the top surface and the narrow beam illuminating it from the side with the beam parallel to the top surface. Each beam, by itself, is less powerful than the minimum necessary to affect the photosensitive dopant. However, where the two beams cross the sum is high enough to cause the change from neutral to positive or negative, depending upon the wavelength of the laser light.

The cube is treated as if it were a stack of layers, each one having its own photo-mask as in normal etching methods. The narrow beam is directed through the layer to be 'etched', using the alpha wavelength. The correct mask is placed between the cube and the upper laser, and the upper laser is turned on to the same wavelength, thus 'etching' the layer with the positive mask. The lasers are turned off and the next mask is put in place. Then the negative mask is 'etched' using the beta wavelength. Last the narrow laser is positioned to the next layer and the process repeated. This is continued until the entire cube is 'etched' with the proper pattern of neutral, negative, and positive areas to produce the desired circuitry. At this point it is no longer a simple cube of silicon but a *Qube*, the newest generation of computer hardware.

One *Qube* could contain, with proper masking, several CPU's, all necessary ROM and RAM, and

even I/O interfaces. Contacts for the outside world and power supply would be made just as with conventional chips by providing pads for mechanical connections.

A major problem would be reliability since even with today's chips the reject rate can be high. I would suggest that backup systems be employed wherever possible within the *Qube*. Each memory bit would actually be three bits. The state of each tri-bit would be 1 if at least two of its bits were 1 and 0 if at least two of its bits were 0. In other words, majority rule. The CPU would also use redundancy, perhaps utilizing three CPUs with a majority rule standard.

One problem in developing the *Qube* is finding a suitable photosensitive dopant. Another would be overcoming the tendency of the photo-mask to act as a diffraction grating causing interference patterns rather than the mask pattern to propagate through the cube. A possible solution to this would be to use two crossing narrow beams of lasers whose positions are computer controlled and whose cross sections are each circles with radii equal to the thickness of a typical layer. The point of intersection would be a minute sphere rather than a plane and the etching would proceed from point to point within the cube rather than layer-to-layer as in the above method. With this method no masks would be required and changes could be made to the circuitry design with great ease. It would also facilitate the production of experimental, small quantity, or one-of-a-kind *Qubes*.

The developer would need only specify a circuit description to the controlling computer, which might rely upon a library of standard circuit designs. OEM users would be able to make their specifications in general terms such as 'two 8080 CPU's, 64k RAM, ROM operating system number AB32, and 7 A/D convertors'.

I believe that the advantages of the *Qube* would be well worth the effort. Initial costs would probably be high as with much of today's semiconductor industry. Yet costs would surely drop as production rose. The *Qube* would use less silicon per element, providing more efficient use of silicon real estate, take less time and would require less handling in production.

The *Qube* can even be tested during production since it would not be necessary to wait for the whole thing to cool to be cut as with conventional chips. After the first layers are 'etched' connections could be made and diagnostics run as the remaining layers are 'etched'. When faulty elements are found a new layer is 'etched' to replace it and the old one 'erased' by bypassing its bus lines.

Imagine a complete multi-processing system with several high level languages fitting into the palm of your hand! There would be no need for time-sharing since every user could have his own dedicated system. By connecting additional non-destructive RAM *Qubes* we could eliminate the need for mass storage devices like tapes and disks. The field of artificial intelligence would get the hardware it needs to implement the theories which are now only dreams.

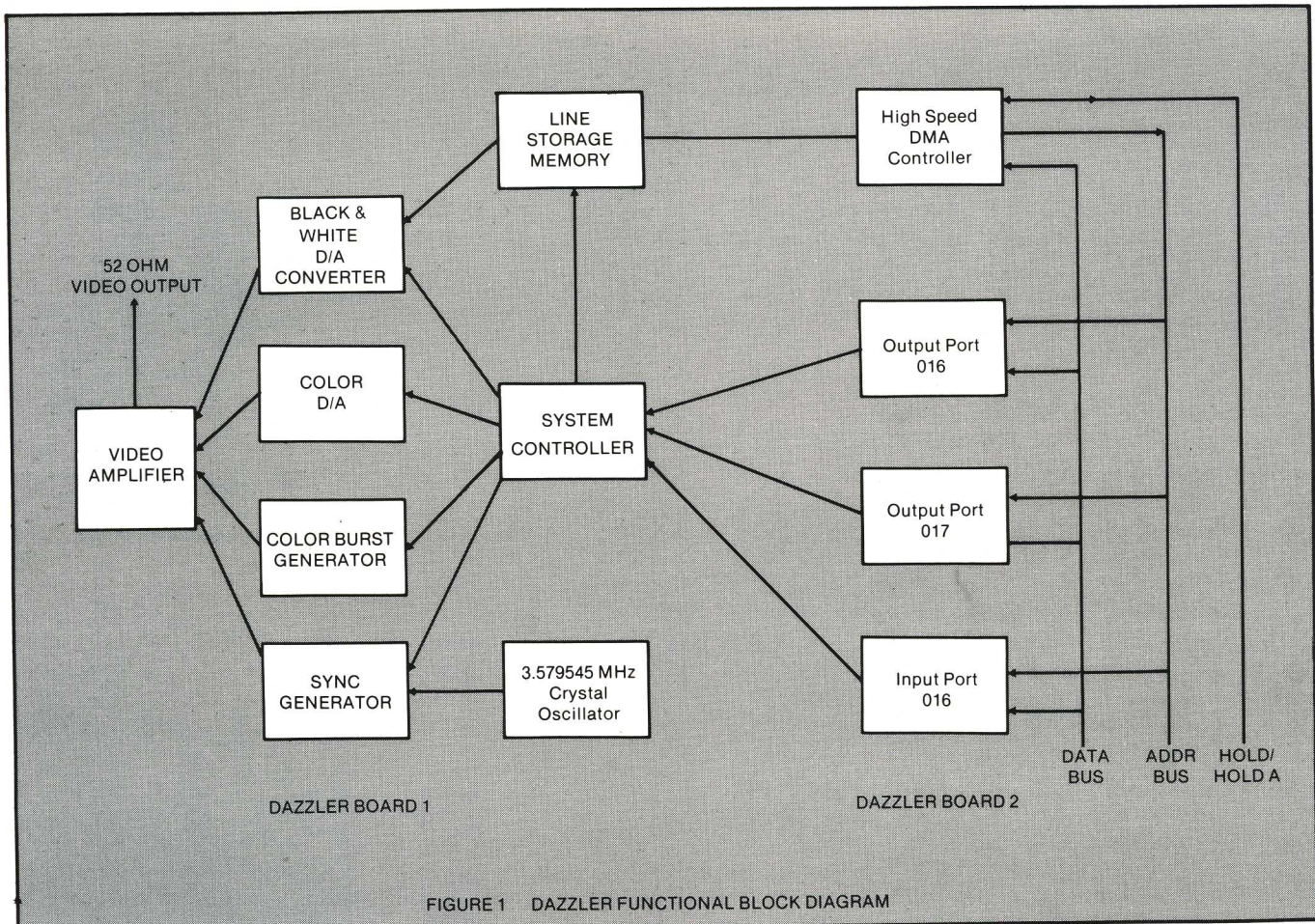
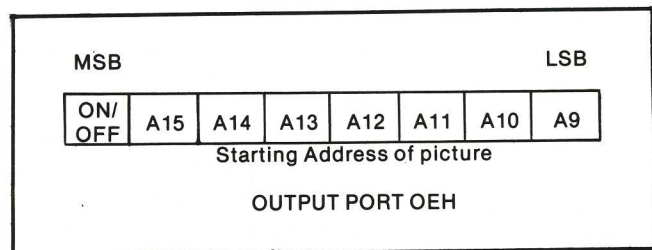


FIGURE 1 DAZZLER FUNCTIONAL BLOCK DIAGRAM

Dazzler modes and addressing. Two bits of the computer input ports are utilized to preset time and framing information to the computer.

The most significant bit of output port OCH is used to enable or disable the Dazzler. A "1" at this bit location turns the Dazzler on. A "0" turns the Dazzler off. The Dazzler can also be turned off manually by depressing the *clear* switch on the Altair front panel. The remaining seven bits of output port OCH are used to set the starting location of the picture to be displayed in the computer's memory. As will be discussed below, the picture may require 512 bytes of memory or 2K bytes of memory depending on the mode in which the Dazzler is operating. In any event this must be **static** memory with an access time of 1 microsecond or faster. A summary of the use of output port OEH is shown below:



Computer output port OF4 is used to set the format of the TV picture. The function of each of the eight bits of this output port is shown in this table:

DAZZLER OUTPUT PORT OFH

- D7 -Not used.
 - D6 = 1-Resolution x 4. Color and intensity set by D4 to -0:
 - D6 = 0-Normal resolution. (32X32 for 512-byte picture. 64X64 for 2K-byte picture.) Color and intensity of each picture element set by 4-bit words in the computer memory.
 - D5 = 1-Picture in 2K bytes of memory.
 - D5 = 0-Picture in 512 bytes of memory.
 - D4 = 0-Black-and-White picture.
 - D4 = 1-Color picture.
- D3 = 1-High intensity color.

D3 = 0-Low intensity color.

D2 = 1-Blue

D2 = 0-No blue

D1 = 1-Green

D1 = 0-No green

D0 = 1-Red

D0 = 0-No red
- In resolution X 4 black-and-white mode, D3 is the most significant bit of 4-bit intensity control word.

⋮

Bit D0 is the least, significant bit.

Bit D7 is not used. Bit D6 is used to set normal resolution (32 x 32 element picture for 512 bytes or 64 x 64 element picture for 2K bytes) or resolution x 4 (64 x 64 element picture for 512 bytes or 128 x 128 element picture for 2K bytes). Bit D5 sets the amount of computer memory (starting at the location given to output port OEH) allocated to the picture. When D5 is "1" 2K bytes of memory are used. When D5 is "0" 512 bytes are used. Bit D4 is used to select between a black and white display and a color display. Bits D3 - D0 are used in resolution x 4 mode to set the color of a color picture or the intensity of a black and white picture. Bits D3 - D0 are not used in normal resolution mode.

Only two bits of Computer input port OEH are used. Bit D7 is low during odd lines and high during even lines. Bit D6 goes low for 4 milli-

- LEGEND**
 A. □ 100 PIN ALTAIR BUS DESIGNATION
 B. □ 16 PIN INTERCONNECT
1. VIDEO DATA
 2. VERT. ENABLE
 3. VERT. SYNC
 4. VIDEO DATA
 5. BOARD No. 2
 6. BOARD No. 2
 7. HORIZ. SYNC
 8. HORIZ. SYNC
 9. HORIZ. SYNC
 10. COLOR ENABLE
 11. ENABLE DAZZLER
 12. 64*64 ENABLE
 13. 1.79 MHz DMA CLOCK
 14. HORIZ. SYNC
 15. HORIZ. SYNC
 16. BYTE - BYTE DEMUX CONTROL

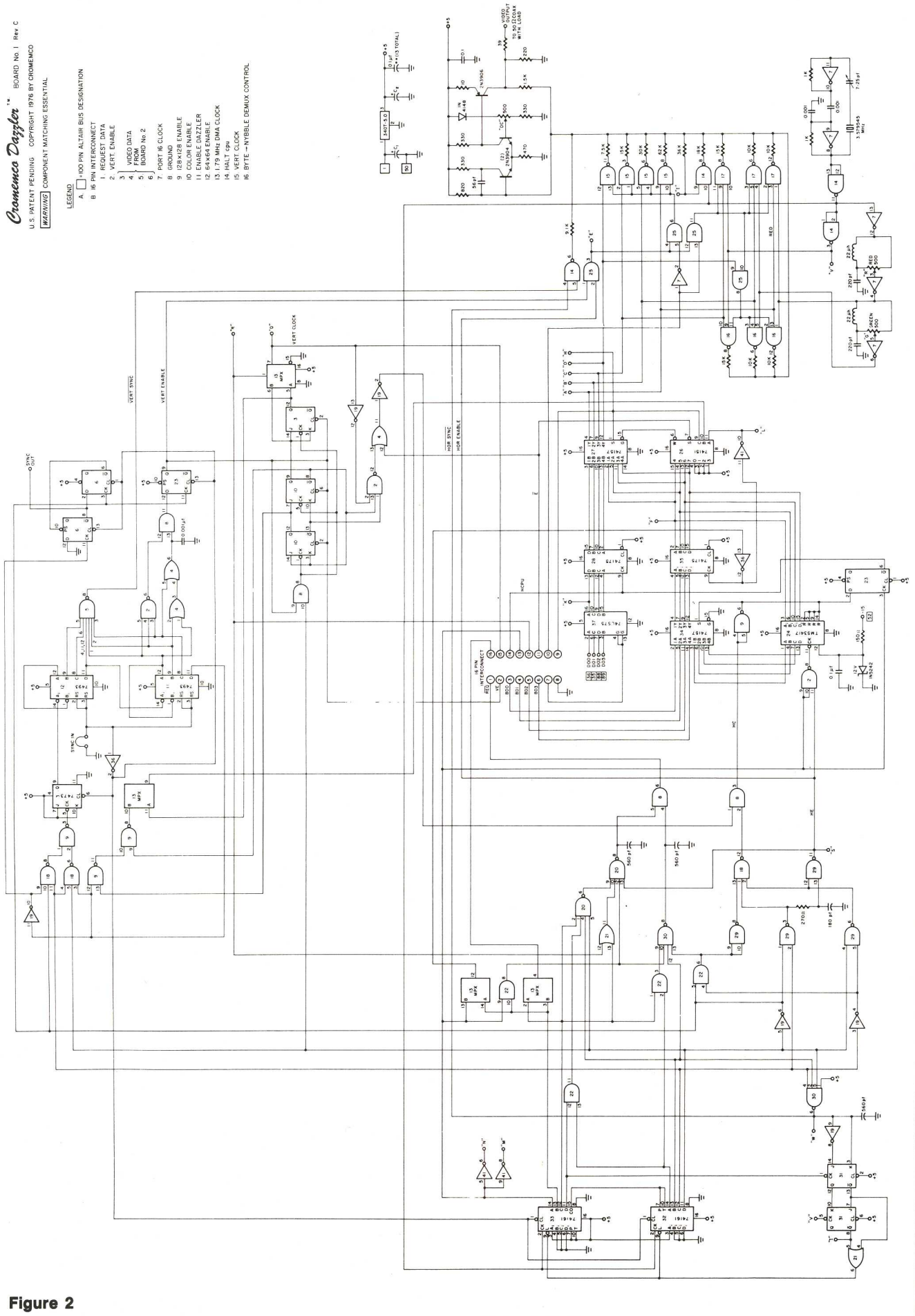


Figure 2

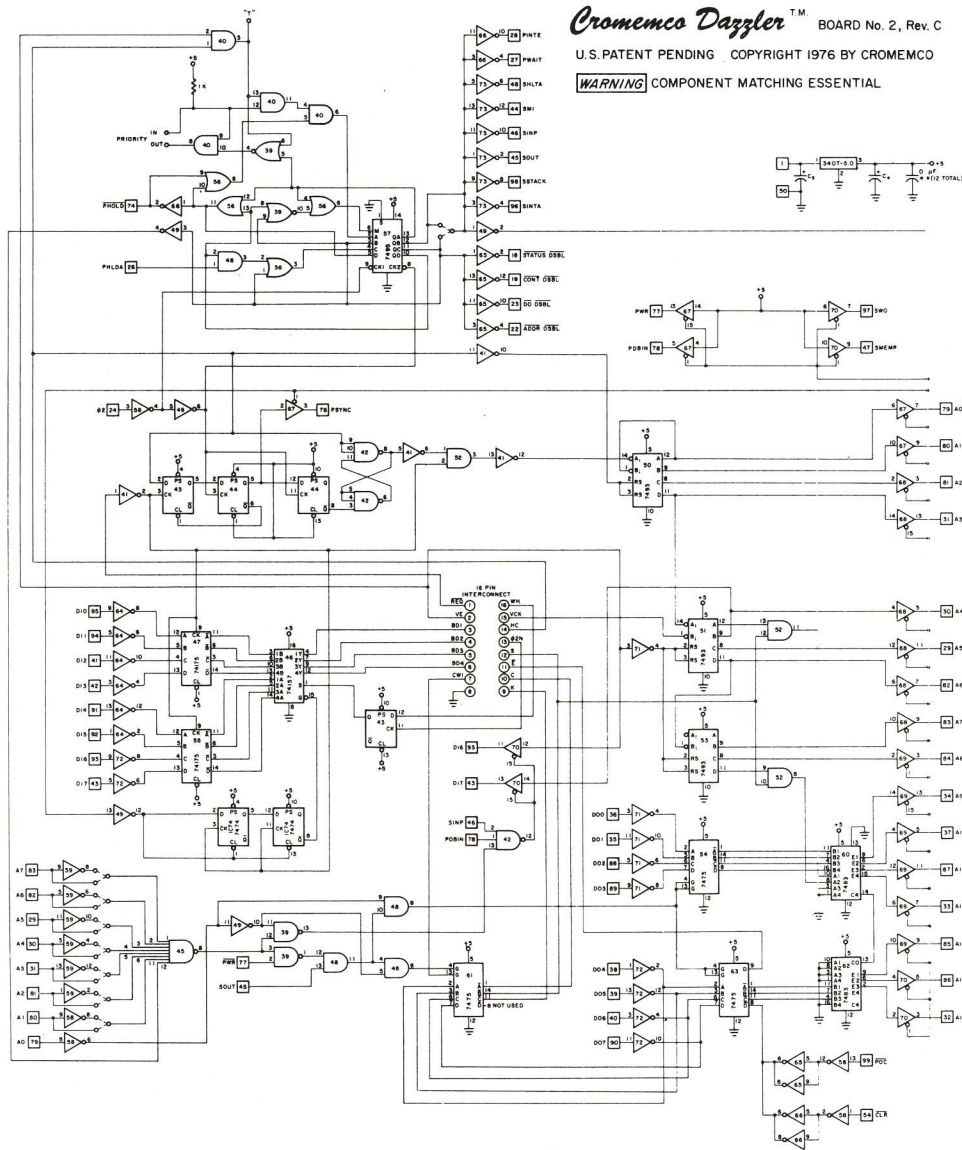


Figure 3

seconds between frames to indicate end of frame.



In order to generate a TV picture with Dazzler, the information that the Dazzler reads from the computer memory must be properly formatted. In resolution x 4 mode each point on the TV screen is controlled by just one bit in the computer memory. When that bit is a "1" the corresponding element of the picture is on. When that bit is a "0" the picture element is off. In resolution x 4 mode the color and intensity of the picture is set by bits D0 to D3 of the control word at output port 017. For full color in resolution x 4 mode, multiple frames of different colors must be interleaved.

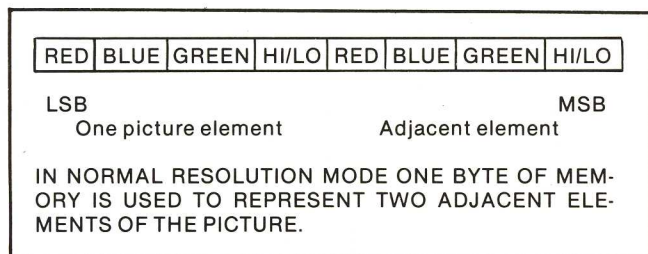
In normal resolution mode the color and intensity of each element of the TV picture are controlled by a four-bit "nybble" in the computer

memory. Two elements of the picture are thus stored in each byte (8 bits) of memory. (For this reason a 64 x 64 picture in normal resolution mode requires 2K of memory.) The lowest order bit of each nybble (D0) determines whether the corresponding element of the picture contains red (1) or no red (0). Similarly D1 controls green, D2 controls blue, and D3 sets either high intensity or low intensity color. In black and white mode these four bits are instead used to specify one of 16 levels of gray.

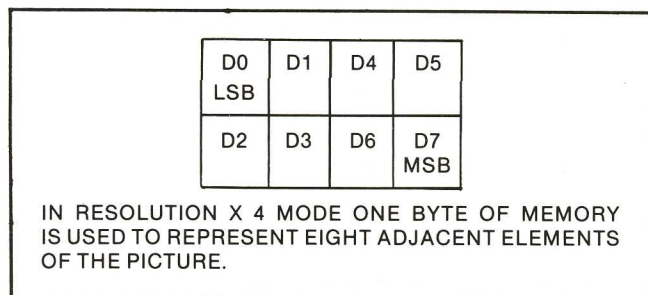
When writing programs for Dazzler displays it is important to remember that the TV picture is stored as a specially coded sequence in the computer memory. The Dazzler simply interprets this code to form a T.V. picture. Two different codes are used depending on whether the Dazzler is used in normal resolution mode or in resolution x 4 mode (as set by the control word sent to output port 017).

In normal resolution mode four bits of computer

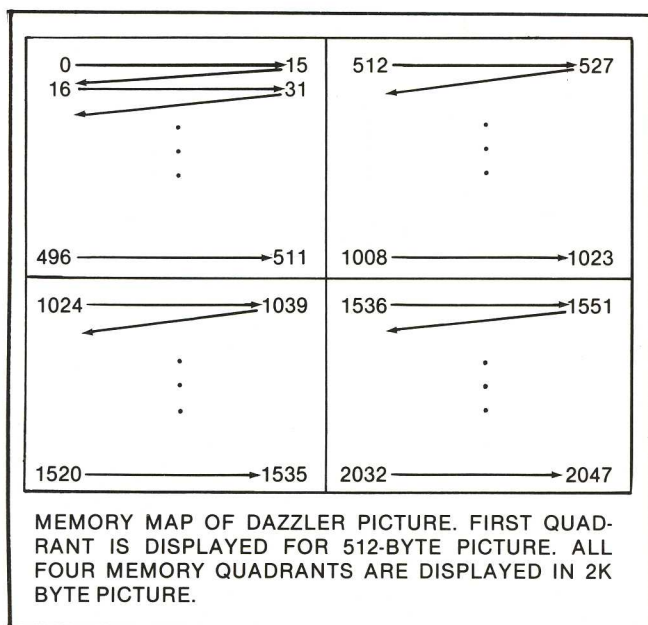
memory are used to code each element of the picture. A 32 x 32 picture requires 512 bytes of memory. A 64 x 64 picture requires 2K bytes of memory. The following diagram shows how the 4-bit code works:



In resolution x 4 mode each bit of the memory is used to either turn on or off a single element of the picture. The eight picture elements controlled by a single byte have the following geometric relationship:



The 2K byte Dazzler picture is stored in memory as four quadrants. Each quadrant of the picture occupies one 512-byte page of memory. Only one page of memory is displayed for a 512-byte picture. The sequence in which memory is scanned in making a 2K byte picture is shown below:



Now that we have looked at the operation of the Dazzler, let us look at the kit itself. As I mentioned earlier, it is a two-board kit containing over 70 ICs. Board #2 has 37 ICs and no other component except the 5V regulator and some noise and filter capacitors. Board #1 also contains 37 ICs in addition to the oscillator and various other discrete components.

Both boards are high quality with gold-plated edge board connector pins. The kit is fully socketed for all ICs — point which always gladdens my heart. The boards are equipped with very high quality solder masks which even surround each individual IC pin — a great help. Component silk-screen masking is excellent with all identifications clearly readable. The adjustment potentiometers and oscillator trimmer are placed at the top of Board #1. For easy access I would prefer a coax cable connector in place of merely soldering the cable to the board. Since the cable is the small LG-174/U, a small connector could easily have been placed in the area where the cable exits. I soldered an OSM 244-2 connector in that upper right corner on its side as a later modification.

Both boards are easy to build and the instructions, though seeming sparse, are adequate. No notes are included on board cleaning which should be done after completion of the construction phase.

The two boards are connected by a flexible flat cable and take two slots in the mother board. Since a method of “piggy-backing” is available, I would like to see hardware designed to hard-mount the two boards together as a single unit which would plug in to two slots. This would make handling the Dazzler an easier task.

A test program is included for initial operation and tuneup. Addresses are in OCTAL. To operate this program you must have 2K bytes of static memory beginning at location zero in memory.

Memory location	Memory contents	Comments
000 000	076	Move immediate into the accumulator
000 001	200	
000 002	323	Output to port number 016
000 003	016	
000 004	333	Input from the sense switches
000 005	377	
000 006	323	Output to port number 017
000 007	017	
000 010	303	Jump to memory location 000 000
000 011	000	
000 012	000	

This program begins by putting the bit sequence “10000000” into the accumulator and outputting it to port 016. This turns on the Dazzler and tells it that the picture that it is to display begins at location zero in memory. Next the program uses

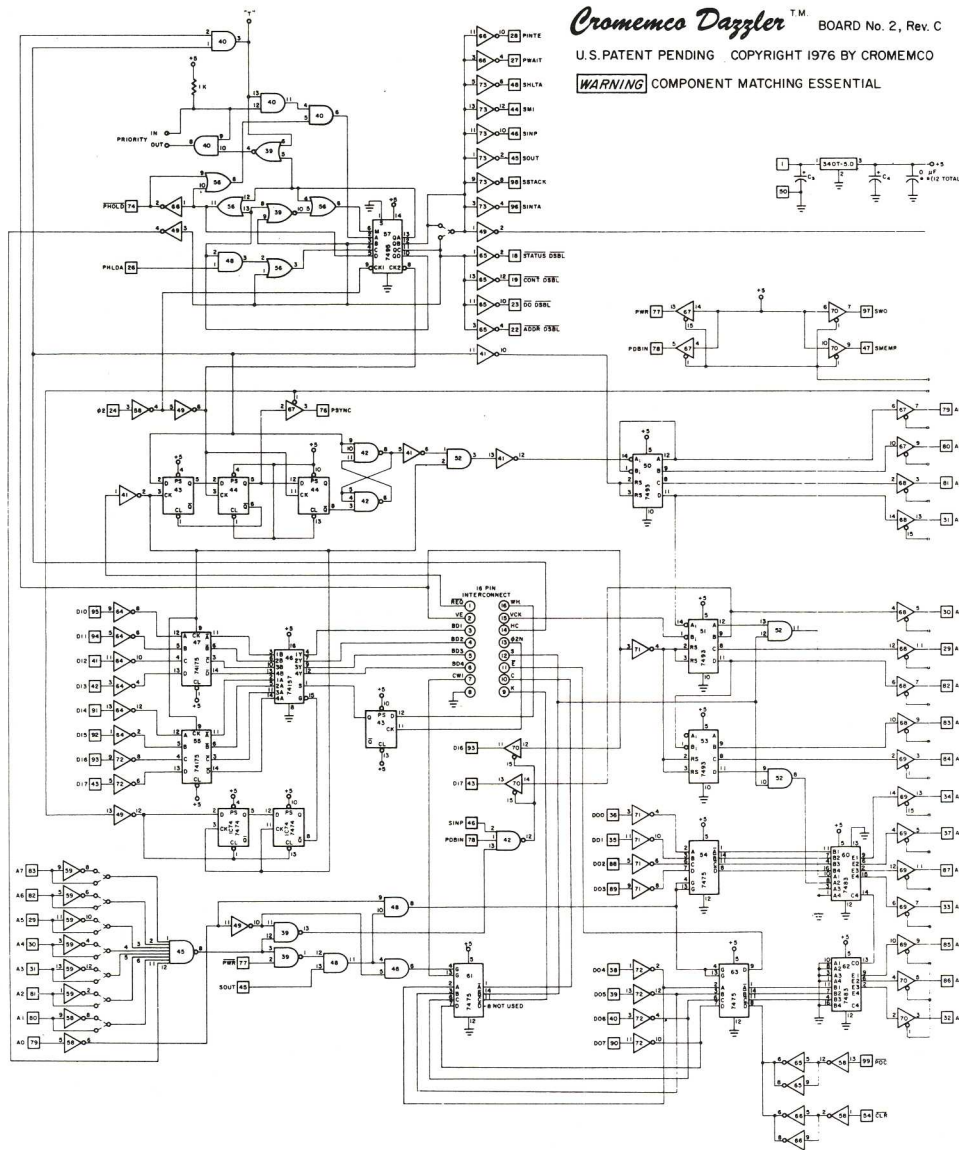


Figure 3

seconds between frames to indicate end of frame.

D7 ODD LINE/ EVEN LINE	D6 END OF FRAME	ONLY TWO BITS OF COMPU- TER INPUT PORT OEH ARE USED.
------------------------------	-----------------------	------------------------------------------------------------

In order to generate a TV picture with Dazzler, the information that the Dazzler reads from the computer memory must be properly formatted. In resolution x 4 mode each point on the TV screen is controlled by just one bit in the computer memory. When that bit is a "1" the corresponding element of the picture is on. When that bit is a "0" the picture element is off. In resolution x 4 mode the color and intensity of the picture is set by bits D0 to D3 of the control word at output port 017. For full color in resolution x 4 mode, multiple frames of different colors must be interleaved.

In normal resolution mode the color and intensity of each element of the TV picture are controlled by a four-bit "nybble" in the computer

memory. Two elements of the picture are thus stored in each byte (8 bits) of memory. (For this reason a 64 x 64 picture in normal resolution mode requires 2K of memory.) The lowest order bit of each nybble (D0) determines whether the corresponding element of the picture contains red (1) or no red (0). Similarly D1 controls green, D2 controls blue, and D3 sets either high intensity or low intensity color. In black and white mode these four bits are instead used to specify one of 16 levels of gray.

When writing programs for Dazzler displays it is important to remember that the TV picture is stored as a specially coded sequence in the computer memory. The Dazzler simply interprets this code to form a T.V. picture. Two different codes are used depending on whether the Dazzler is used in normal resolution mode or in resolution x 4 mode (as set by the control word sent to output port 017).

In normal resolution mode four bits of computer

the front panel sense switches to set the control word sent to output port 017. This allows the user to experiment with different output formats and appreciate the fantastic versatility of the Dazzler display. When sense switch A12 is raised the Dazzler goes into color mode, and a colorful quilt-like pattern will appear on the TV screen.

Another personal gripe — I wish Cromemco had done everything in HEX rather than OCTAL. I feel much more familiar with HEX and hate to switch back and forth.

The troubleshooting guide is very comprehensive with 49 items included in the kit manual. The test equipment requirements are a VTVM (or VOM) and a *triggered* sweep-scope (DC-5MHz) with an external synch input.

Fortunately my Dazzler came up running the first time. I assume that to be luck, not skill for I am too "experienced" to follow instructions. I've connected the Dazzler to a small black and white set which is at present adequate for my uses. I would ultimately like to connect it to a color set, but at the moment would have to overrule my family's objections to "butchering" our one color T.V.

There is an alternate answer to this problem. The Dazzler can be connected to the antenna terminal of the T.V. rather than to the video amplifier, but to do this an R.F. modulator is required. Such a modulator is the Pixieverter Model XPXV-2A available for \$8.50 from ATV RESEARCH, 13th and Broadway, Dakota City, Nebraska 68731. If you use a Pixieverter you *must* also use a matching transformer between the output of the Pixieverter and the TV set. A standard 75 ohm to 300 ohm matching transformer, available at TV supply stores, will provide the required matching. (Radio Shack stores carry such a transformer for \$2.59, model #15-1140).

While on the subject of T.V. connection, it would be useful if Cromemco included some typical video amplifier connection diagrams to help the process. However, most of our readers have already seen the excellent articles on T.V. hookups in BYTE Magazine.

Cromemco has various programs already available for the Dazzler. They are:

1. DAZZelmation - an animation program
2. KALEIDOSCOPE - a random colorful picture generator
3. DAZZelwriter - an extensive alphanumeric display
4. THE BONE OF LIFE - a full-color version of LIFE

For those of you with joystick analog inputs for your computer, Cromemco has DAZZLEDOODLE and CHASE available. Hopefully the Cromemco Joystick and D-7A module will be the subject of another article.

In summary the Dazzler is a well-planned design, capable of extending the versatility and capabilities of your computer while providing enjoyment for you and your family with this exciting art form.

TO SERVE YOU BETTER!

BITS N BYTES

Announces:

BITS N BYTES #2

Opening soon in San Diego!

Call (714) 422-0889 for info

Our Product Lines:

Digital Group	Solid State Music
IMSAI	Soroc
Lear-Siegler	Tarbell
Okidata	TDL
Pickles & Trout	Vector Graphic
PolyMorphic Systems	

BITS N BYTES

679 "D" South State College Blvd.

(College Business Park)

Fullerton, CA 92631

(714) 879-8386

Hours: M-F 6 p.m.-9 p.m., Sat. 12-6 p.m.

CIRCLE INQUIRY NO. 65

THE NEW

BYTE SHOP™ OF MIAMI

the affordable computer store

7825 BIRD ROAD
(305) 264-2983 MIAMI, FL
DIAL 264-BYTE 33155

#1 IN SOUTH FLORIDA

Microcomputer kits, tools, computer system, books, hobby magazines, software (programs), expert advice, repairs, parts, chips, etc. All kits guaranteed. For hobby, business, office, schools or profession.

Join our classes and workshops—whether a beginner or an expert. Come in and browse or play real computer games. Let us show you how to use the new microcomputers for any application. Computer club information.

SPECIALIZING IN SYSTEMS

OPEN: 10-6 Daily & Sat.
10-9 Thur.

CIRCLE INQUIRY NO. 66