COLOR BURNER

Green Mountain Micro

THE COLOR BURNER

EPROM Programming System for the TRS-80 Color Computer

Designed by Dennis Bathory Kitsz

Distributed by Green Mountain Micro
Roxbury, Vermont 05669

****
Attention! Read This First!
****

Read this booklet first. If at any time within six weeks after the purchase date you decide that the Color Burner is not for you, for any reason, you may return it for a full refund of the purchase price. However, if you open the plastic bag containing your Color Burner and then wish to return it (unless it is defective), there is a $15 checkout and restocking fee. Defective Color Burners will be exchanged at no charge. No returns may be made for any reason without an authorization number; contact Green Mountain Micro by phone or mail.

****
Attention!
****

You are responsible for reading all warnings, cautions and attentions in this documentation.

****
Warranty and Disclaimer
****

Your Color Burner, an assembled and tested EPROM programmer, is warranted to be free from defects for a period of six months from time of purchase (1) when installed and used according to the instructions printed in this documentation, (2) when used on an otherwise unmodified TRS-80 Color Computer, Color Computer 2, or TDP-100, and (3) when used with Green Mountain Micro's Color Burner software. The user alone determines applicability and appropriateness of the Color Burner for the user's computer system, and by this purchase agrees to hold the manufacturer and distributor harmless for any damages, consequential or otherwise, arising from use of the Color Burner. This includes, but is not limited to, interruption of service; loss of profits, business, or anticipatory profits; or consequential damages arising from use of the Color Burner. Warranty does not cover physical damage, including breaking of battery connectors, damage caused by the use of external power sources other than those specified, or damage caused by the use of EPROMs other than those indicated in this documentation. This warranty does not apply to Color Burner kits or Color Burners modified by the user.

THE COLOR BURNER FROM
GREEN MOUNTAIN MICRO

****

IMPORTANT NOTICE ON
WARRANTY LIMITATIONS

1. Warranty does not cover damage to 6821 PIAs, which are under user control during EPROM programming. Replacement of damaged 6821s will be made at our regular parts cost, plus $10.00 installation, shipping and handling.
2. Warranty does not cover damage to unit unless COLOR BURNWARE has been used to program EPROMs. Our records must show that you have purchased BURNWARE or your Color Burner warranty is VOID.
3. Do NOT attempt repairs on your Color Burner. User-attempted repairs will VOID the warranty, except where such repairs were authorized by Green Mountain Micro technical support staff.
4. Use the lower 24 pins of the 28-pin socket for 2716, 2732 and 68764 EPROMs (see documentation, page 11). Use of the upper 24 pins may damage the 6821 PIAs and will VOID the warranty.
5. Do NOT attempt to program a Texas Instruments TMS2716. This is a multiple-voltage part NOT in the 27XX family of EPROMs. If you use a Texas Instruments EPROM, make sure it is a TMS2516. Use of a TMS2716 may damage the Color Burner and VOID the warranty.

Except as provided above, this notice does not express or imply changes to the warranty and disclaimer provisions printed in the Color Burner documentation.

****

****

Contents

****

Copying the Software

You are encouraged to make a backup copy of this software. After loading and running the program as described under "Loading and Running the Software" (page 3), tap <BREAK> until you get an "OK". Insert a fresh cassette in the recorder, set it to record, and type the following Basic program line:

```
FOR W = 1 TO 3 : CSAVE "BURNER" : FOR U = 1
TO 1000 : NEXT : CSAVEM"BLAST",&H3D00,&H4000,
&H3D27 : FOR U = 1 TO 5000 : NEXT : NEXT
```

```
**************************************************************
*                                                            *
*              T H E   C O L O R   B U R N E R               *
*   EPROM Programming System for the TRS-80 Color Computer   *
*                                                            *
*             Designed by Dennis Bathory Kitsz               *
*                                                            *
*          Distributed by Green Mountain Micro               *
*                Roxbury, Vermont 05669                      *
*                                                            *
**************************************************************
```

What is the Color Burner?

The Color Burner is plug-in module for the TRS-80 Color Computer that programs data into erasable, programmable, read-only memories (EPROMs). EPROMs are used for plug-in cartridge program packs, for replacement or correction of BASIC language ROMs, for video display character generators, and for storage of other vital data. The Color Burner requires operating software in addition to the program to be stored in the EPROM.

Color Burner hardware features:
* 21-volt and 25-volt programming pulses for compatability with most popular EPROMs.
* A "personality module", which directs voltages and programming pulses to the EPROMs, and allows expansion to new EPROMs.
* 24-pin or 28-pin EPROM capability for future expansion.
* Established circuit parts and low parts count for high reliability and long life.
* Inexpensive battery operation using three 9-volt alkaline batteries.
* Compatible with Color Computer, Color Computer 2, and TDP-100.

Color Burner software features:
* Expandable, with programming of 2716, 2732, 2764, 27128 and 68764 included.
* Pre-charge pulse to read newer EPROMs such as 68764.
* Read, write, verify and erase-verify modes for all EPROMs.
* Full or partial programming of EPROM memory contents from one to 16,384 bytes.
* Machine language monitor for display and editing of material to be programmed.
* Personality module design system.
* Extensive "Help" commands cuts time-consuming reference to documentation.

Installing the Color Burner

The Color Burner is a temporary addition to your Color Computer system, designed for use with cassette. Although you may be using a disk system, the Color Burner is electronically incompatible with a disk system.

Turn off your Color Computer, and remove any cartridges from the right-hand cartridge slot. With the components facing up and the gold edge connector pointing to the left, insert the Color Burner straight into the cartridge slot. Make sure the three 9-volt battery connectors are free and outside the computer's case. They must not lay on the Color Burner board.

Double check to make sure the Color Burner is inserted fully, that it is straight, and that no metal parts touch it. You may wish to support it two ways: with a piece of cardboard or a wood strip or, if you plan to use it often, with permanently attached felt or rubber feet.

To test the installation, turn on the power to the computer. You should see the usual green screen and sign-on message. If the screen does not appear correct, turn off the computer immediately! Double-check your insertion.

Attention! Kit Builders!

If your Color Burner does not work when tested, Green Mountain Micro will repair it for $12.00 plus $2.50 shipping and handling. Physically damaged Burners, Burners built with other than electronic (rosin core) solder, or otherwise built disregarding the instructions are not covered by this offer. Ship the Burner via UPS or First Class Mail, insured. Call for authorization before sending.

****
Using the Color Burner Software
****

Loading and Running the Software

The software consists of a Basic driver and a set of machine language subroutines. To load this program:

1. Turn on your 32K Extended Color Basic Color Computer.
2. Insert the tape into the player, make sure it is rewound, and set it to play.
3. Type:
   PMODE0 : PCLEAR1 : CLOAD <ENTER>
   The Basic driver program will load into the computer.
4. Keep the tape player on until the main menu appears.
5. When "OK" returns, type:
   RUN <ENTER>
   The machine language subroutines will be loaded.
6. The Main Menu will be presented.

```
::::::::::::::::::::::::::::::::::::::::::::::::
:       Color Burner Monitor/Blaster 2.2       :
:       Copyright (c) 1984   D. B. Kitsz       :
:                                              :
:                  >>> Menu <<<                :
:                                              :
:            1.  Set EPROM type.               :
:            2.  Load object tape.             :
:            3.  Examine object code.          :
:            4.  Save object tape.             :
:            5.  Blast EPROM.                  :
:            6.  Define new EPROM.             :
:            7.  Read EPROM program.           :
:                                              :
:       Touch number or "H"  for HELP.         :
::::::::::::::::::::::::::::::::::::::::::::::::
```

Remember that you may touch "H" from the main menu to receive information about each menu selection.

Main Menu Selection #1:
Setting the EPROM Type

A submenu is presented:

```
::::::::::::::::::::::::::::::::::::::::::::::::::
:       Color Burner Monitor/Blaster 2.2         :
:       Copyright (c) 1984   D. B. Kitsz         :
:                                                :
:        <*> EPROM DEFINED = NONE <*>            :
:                                                :
:          1.  2716 (2K X 8)                     :
:          2.  2732 (4K X 8)                     :
:          3.  2764 (8K X 8)                     :
:          4.  27128 (16K X 8)                   :
:          5.  68764 (8K X 8)                    :
:          6.  SPECIAL (MENU #6)                 :
:                                                :
:         Touch No. or "M" for Menu              :
::::::::::::::::::::::::::::::::::::::::::::::::::
```

An EPROM type must be selected in order to establish certain parameters within the machine language subroutines, and to provide appropriate prompts during memory loading, saving and examination. Once you have defined the EPROM type, it's a good idea to insert the personality module (see "Using the Color Burner Hardware" in this booklet).

Selection #6 is provided to permit you to work within a theoretical range of 64K bytes, and is included for future expansion of this software.

Main Menu Selection #2:
Loading Object Code

Object code -- binary program information or data -- up to 16K in length may be loaded into memory for editing, examination and burning into an EPROM. You must know the origin (first byte) of the object code. For example, plug-in ROM cartridges have an origin of $C000, Color Basic and origin of $A000, and Extended Color Basic an origin of $8000. In response to the prompt, enter the origin in hexadecimal.

The next prompt asks for the location in the EPROM where the object code is to be loaded. For example, you may have created several programs for storage in a large EPROM for a plug-in ROM pack, residing at $C000, $C10B, $C223, $CEA9 and $D002 in the Color Computer's memory. In the final programmed EPROM, these would reside at relative addresses $0000, $010B, $0223, $0EA9 and $1002. One of these latter values would be entered in response to the prompt.

You will then have the option of returning to the main menu, or loading the object code.

Main Menu Selection #3:
Examining Object Code

You may examine any area of memory within the Color Computer, and edit any area of RAM. The memory used for programming EPROMs resides at $4000 to $7FFF.

You will be asked for a starting address, in hexadecimal. After entering this value, the screen will display 16 rows of eight bytes, preceded by the memory address. A flashing cursor will be displayed.

Use the four keyboard arrows to move the cursor over the memory value you wish to change, and type in the new value (remember to use hexadecimal notation). When you have completed the changes, you have two options: (1) Pressing ENTER will confirm those changes and place them in memory; (2) Pressing "M" will return the main menu without making the memory changes.

Caution: You may edit ANY area of RAM using this editor. Be sure to be very careful when making changes in RAM below $4000, where Basic program variables, screen, plus key Basic vectors and tables exist.

Main Menu Selection #4:
Saving Object Code

The object code to be saved will reside in your RAM editing buffer from $4000 to $7FFF. This Color Burner software does not support saving blocks of Basic ROM, cartridge ROM, or low memory.

Enter an address from $0000 as a starting address within the EPROM you have defined, and then a larger value up to $4000 as the ending address. You have the option of saving to tape or returning to the main menu.

If you save, the block of binary information will be dumped to tape in standard CLOADM format, with an origin of $4000. It can therefore be loaded directly back into the Color Burner software for later editing and programming.

Main Menu Selection #5:
Burning an EPROM

Your EPROM must be installed in the socket, the proper EPROM must be defined through main menu selection #1, the personality module must be in place for that EPROM, and your object code must be ready and waiting in the memory buffer.

It is wise to check for an erased EPROM first, unless you are adding to a partially programmed EPROM. Press "Y" to check for erasure.

The starting address prompt will be displayed. You may program any block of memory within the EPROM; to program the entire EPROM, enter 0000 at this prompt.

The ending address prompt will be displayed. This is actually ending-address-plus-one. That is, if your are programming an entire 2716, you would enter 0000 as the starting address, and 0800 as the ending address. Remember, these values are in hexadecimal.

The programming will begin immediately, and continue until complete. DO NOT INTERRUPT THIS PROCESS BY PRESSING RESET OR TURNING THE MACHINE OFF! If you must interrupt the programming, disconnect one of the batteries, and ONLY THEN press the reset button.

Finally, you will be asked if you wish verification. Usually, you will want to do this, since verification can be performed quickly. A "Verify Fails" message will be displayed at each location not properly programmed, and the message will stay on the screen until "Verify Complete" is shown. This can be quick, so watch carefully.

You may return to the menu at this point.

Main Menu Selection #6:
Defining a New EPROM

The definition of a new EPROM is actually a reference for the personality module; programming timing definitions, pulse times, etc., are not supported in this version of the Color Burner Software (see the section entitled "Writing Your Own Software" in this booklet).

To define the personality module, you will need the information from the EPROM's data sheet. You will be asked a series of questions:

1. How many pins (24 or 28)?
2. Power (Vcc) on what pin?
3. Program (Vpp) on what pin? (This is the positive-going programming pulse voltage, not the program enable).
4. Low OE on what pin? This is the low-going output enable, present on some of the 2700 family of EPROMs. Enter 0 if there is no OE present on the EPROM.

5. Low CE on what pin? This is the low-going chip enable, present on some of the 2700 family of EPROMs. It is also called simply E. Enter 0 if there is no CE present on the EPROM.
6. A11 on what pin? This is address line 11, used depending on the EPROM's size. Enter 0 if there is no A11.
7. A12 on what pin? Same as above for address line 12.
8. A13 on what pin? Same as above for address line 13.
9. Low PGM on what pin? This is the low-going program-enable pulse present on some of the larger 2700 family of EPROMs. Enter 0 if there is no PGM present.

After answering these nine questions, you will be presented with a diagram of the personality module wiring. Should a wire go off the top of the module, this indicates that this EPROM cannot be programmed by the Color Burner.

For more information about personality modules, see the section entitled "Using the Color Burner Hardware" in this booklet.

Press Enter when you have copied the diagram of the personality module, and you will be returned to the main menu.

Main Menu Selection #7:
Reading an EPROM

To read an EPROM, you must have set its type, inserted the correct personality module, and have inserted the EPROM itself.

You will be prompted for the relative starting address within the EPROM, and the relative ending address. Use hexadecimal numbers, and remember that the ending address is the actual ending address plus one.

The EPROM will be read into the memory buffer, and you may then return to the main menu.

Hit RESET? Mess up? Make an error?
Need to recover SOMEHOW?? Try this:

GOTO3 <ENTER>

If variables are still intact, you will return to the Main Menu (Ah!)

Writing Your Own Software

The Color Burner is provided with two programming voltages, a 28-pin socket, and personality modules. Commercial EPROMs require programming and power voltages, several kinds of program pulses, and come in 24-pin and 28-pin packages. To accommodate these variations, all programming is done using software. The Color Burner software from Green Mountain Micro reads and burns 2716, 2732, 2764, 27128 and 68764 EPROMs.

You may need to write your own software to accommodate other types such as 25-series EPROMs, to take advantage of future enhancements of an existing series, or to incorporate one of the "intelligent" algorithms for faster programming of large EPROMs.

If you are using the Color Burner software, the information below will be useful.

Programming tables contain four bytes which are output through Color Burner port $FF46. Byte #1 is the value to turn the EPROM off; byte #2 turns the EPROM on to a read condition; byte #3 turns the EPROM on to a program-ready condition; byte #4 is the programming pulse.

2716 programming values	$3D07-$3D0A
2732 programming values	$3D0B-$3D0E
2764 programming values	$3D0F-$3D12
27128 programming values	$3D13-$3D16
68764 programming values	$3D17-$3D1A
Free programming table #1	$3D1B-$3D1E
Free programming table #2	$3D1F-$3D22
Free programming table #3	$3D23-$3D26

Major Subroutines:

Configure port A ($FF40) as input data (Y not used)	$3D27
Configure port A ($FF40) as output data (Y not used)	$3D34
Configure ports B, C and D ($FF42, 44, 46) as outputs	$3D42
Establish EPROM type and return address of table in Y register (X register saved, B not used)	$3D60
Display a 16-character message (X points to message)	$3D6D
Delay approx. 30 microseconds (B register saved)	$3D7B
Delay pulse time of N milliseconds (D register saved)	$3D85
16-bit delay value	$3D87-$3D88
50 mS delay = $18F8; 2 mS delay = $0100	
Display 16-bit value in hex (value in D; X and Y saved)	$3D92
Display 8-bit value in hex (value in A; X, B not used)	$3DA1
General setup to read EPROMs (Y saved, B not used)	$3DBE
Program EPROM using tables and subroutines	$3DCD
Check for erased EPROM using tables and subroutines	$3E4B
Verify correct programming using tables and subroutines	$3E8C

```
Read EPROM into memory at $4000 using tables and
        subroutines                                          $3EED
Message "Programming"                                        $3F2B
Message "Verifying"                                          $3F3B
Message "Erase Check"                                        $3F4B
Message "Verify Fails at"                                    $3F5B
Message "Erase Fails at"                                     $3F6B
Message "Burn Fails at"                                      $3F7B
Message "Burn Complete"                                      $3F8B
Message "EPROM Erased"                                       $3F9B
Message "Verify Complete"                                    $3FAB
Message "Reading EPROM"                                      $3FBB
Message "EPROM Read"                                         $3FCB
Message "Bad EPROM Type"                                     $3FDB
```

The creation of your own software is outside the scope of this documentation. However, the following information about the EPROM to be programmed will be necessary:

1. The programming voltage (21 or 25 volts).
2. The programming pulse width (2 to 50 mS).
3. Number of programming pulses required.
4. The EPROM physical size (24 or 28 pins).
5. The placement of variable pins including addresses A11, A12 and A13; chip enable (CE*), if any; output enable, (OE*) if any; program low (PGM*), if any; program pulse (Vpp); power supply voltage (Vcc).
6. The method of reading (continuous read or pre-charge).
7. The method of programming (byte or block).
8. Other timing considerations not describe above.

Essential data for several EPROMs is provided with this documentation, as well as a listing of the software developed by Green Mountain Micro. The comments in the Basic and source listings, together with the EPROM data, should provide the information you need to devise your own programming software for the Color Burner.

Software not Loading???

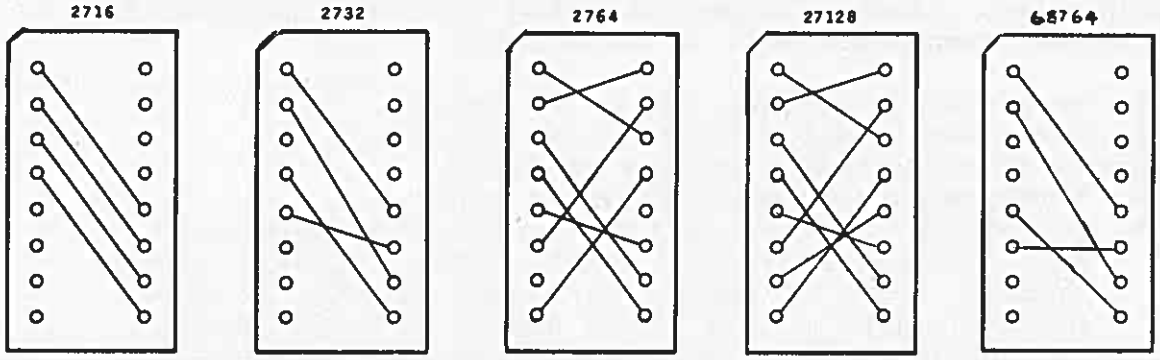
There are three copies of the Color Burner software on each side of the tape. Please try these copies before contacting Green Mountain Micro for exchange. Refer to your Color Basic manual for tape loading hints. Thanks.

****
Using the Color Burner Hardware
****

The Personality Module

The personality module directs programming information to the proper positions of the EPROM. Because of the wide variety of EPROMs and the expanability of the Color Burner software, the personality module is used as a simple alternative to expensive switching circuits.

You are provided with an unwired personality module with your Color Burner. To wire the module, choose the EPROM type you wish to program, and examine the drawings below:



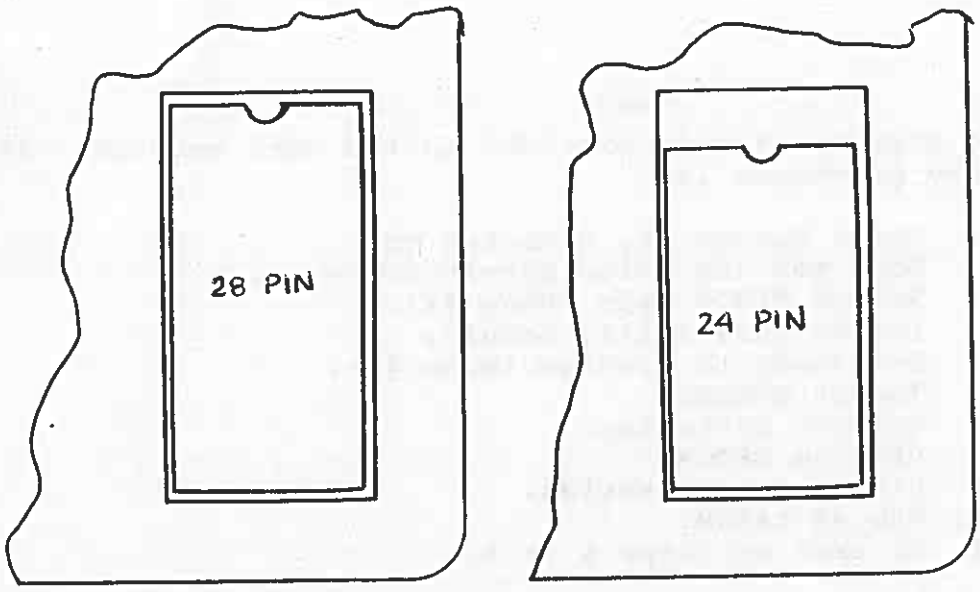
Hold the module in a small clamp, with the angled corner (marked pin 1) facing away from you. Run wires connecting the top, forked contacts. Be sure wires that cross do not touch; insulate them from each other. Solder quickly (don't melt the plastic!).

The personality module is inserted in the 16-pin socket at the far right side of the Color Burner board, with the angled corner facing the far side.

Insert the personality module as soon as you have selected the EPROM you wish to program (main menu selection #1). Keep it in place until you change EPROM types.

The Low-Insertion Force Socket

EPROMs are read and programmed by placing them in the low-insertion-force socket on the near right side of the Color Burner board. The socket has 28 contacts, and can be opened and closed for easy insertion and removal of EPROMs. All EPROMs are programmed with their identification notch facing the far side of the Color Burner board. 28-pin EPROMs fit right into the socket, taking up all the contact positions, whereas 24-pin EPROMs fit into the near 24 contacts. See the diagram on the next page:



Several different types of this socket have been provided with the Color Burner, some with screwdriver slots, some with handles. When using those with triangular slots marked "C-O" (manufactured by Kel-Am), be sure to insert a screwdriver of exactly the right width. Too small a screwdriver won't turn it and may deform the plastic, too large a screwdriver may snap of the plastic tongue. Never force a slotted socket closed. Some EPROMs have thick pins, and will make adequate contact with the Kel-Am sockets without closing the socket completely.

Insert the EPROM after you have selected your EPROM (main menu selection #1), and after you have inserted the personality module (see above), but before connecting the 9-volt batteries.

Disconnect one battery before removing the EPROM, when programming is complete.

The Battery Clips

For safety, the batteries are connected as the last step before programming; as you get proficient, you will be able to leave the batteries in place for the whole session.

Connect two batteries, but leave the third unconnected until an EPROM has been selected (main menu selection #1), the personality module has been inserted, and the EPROM has been placed in the socket. Then connect the last battery.

The battery clips are soldered to the Color Burner board with relatively fine wires. It's best to flex these wires as little as possible; place a small elastic band around the Color Burner when it's not being used.

Procedure

A detailed step-by-step is provided in the next section. In general, the procedure is:

1. Color Burner in, Computer on.
2. Load and run Color Burner software.
3. Select EPROM type (Menu #1).
4. Insert personality module.
5. Get ready to program (Menu #5).
6. Insert EPROM.
7. Connect batteries.
8. Program EPROM.
9. Disconnect batteries.
10. Remove EPROM.
11. Go back to steps 3 or 5.

Important things to remember:

1. Personality module faces the Burner's far side.
2. EPROM faces the Burner's far side.
3. 24-pin EPROMs sit in the near 24 socket contacts.
4. Use only fresh 9-volt alkaline batteries.

Battery Life

Battery life will vary, depending on the quality and shelf life of the batteries you buy. In using stock Radio Shack alkaline 9-volt batteries, we have found that you can program approximately 100K of EPROMs (fifty 2716s, twenty-five 2732s, twelve 68764s). You may be able to program more than that (we have programmed more than one hundred 2716s with one set of batteries), but be sure to verify correct programming after each one.

A T T E N T I O N

The Personality Module header is shipped in place in its socket.

Be sure to REMOVE the Personality Module before soldering to it!!

****

Sample Session: Step-by-Step EPROM Copying

****

Here is a step-by-step procedure for copying an EPROM. The example used will be the 68764. (Steps 16, 22 and 29 are recommended, but optional).

1. Install the Color Burner, component side up.
2. Turn on the 32K Extended Basic Color Computer.
3. Load and run the Color Burner software.
4. Touch main menu selection #1, Set EPROM type.
5. Touch submenu selection #5, 68764 (8K X 8).
6. Touch M to return to main menu.
7. Insert the 68764 Personality Module.
8. Touch main menu selection #7, Read EPROM program.
9. Open low-insertion-force socket (screw slot/handle).
10. Insert master 68764 EPROM in socket (near 24 pins).
11. Close low-insertion-force socket.
12. Connect the three 9-volt batteries in the clips.
13. Enter starting address 0000 in response to prompt.
14. Enter ending address 2000 in response to prompt.
15. EPROM will be read; touch M to return to main menu.
16. Disconnect one of the 9-volt batteries.
17. Open low-insertion-force socket.
18. Remove master 68764 EPROM from socket.
19. Touch main menu selection #5, Blast EPROM.
20. Insert blank 68764 EPROM in socket.
21. Close low-insertion-force socket.
22. Reconnect the remaining 9-volt battery.
23. Touch Y in response to prompt to verify erased EPROM.
24. If erase passes, press C to continue; if not, press M, remove the defective EPROM, and return to step 19.
25. Enter starting address 0000 in response to prompt.
26. Enter ending address 2000 in response to prompt.
27. Programming will complete; touch Y to verify burn.
28. Touch M to return to the main menu.
29. Disconnect one of the 9-volt batteries.
30. Open the low-insertion-force socket.
31. Remove the newly programmed 68764 EPROM.
32. If you wish to program another blank, go to step 19.
33. If you wish to program from a different 68764, go to step 8.
34. If you wish to program a different kind of EPROM, remove the personality module and go to step 4.
35. If you wish to complete the session, disconnect all batteries.
36. Remove the personality module from the Color Burner.
37. Turn off the Color Computer.
38. Remove the Color Burner Board from the Computer.

---

* Always use fresh 9-volt alkaline batteries for programming!

Color Burner Parts List

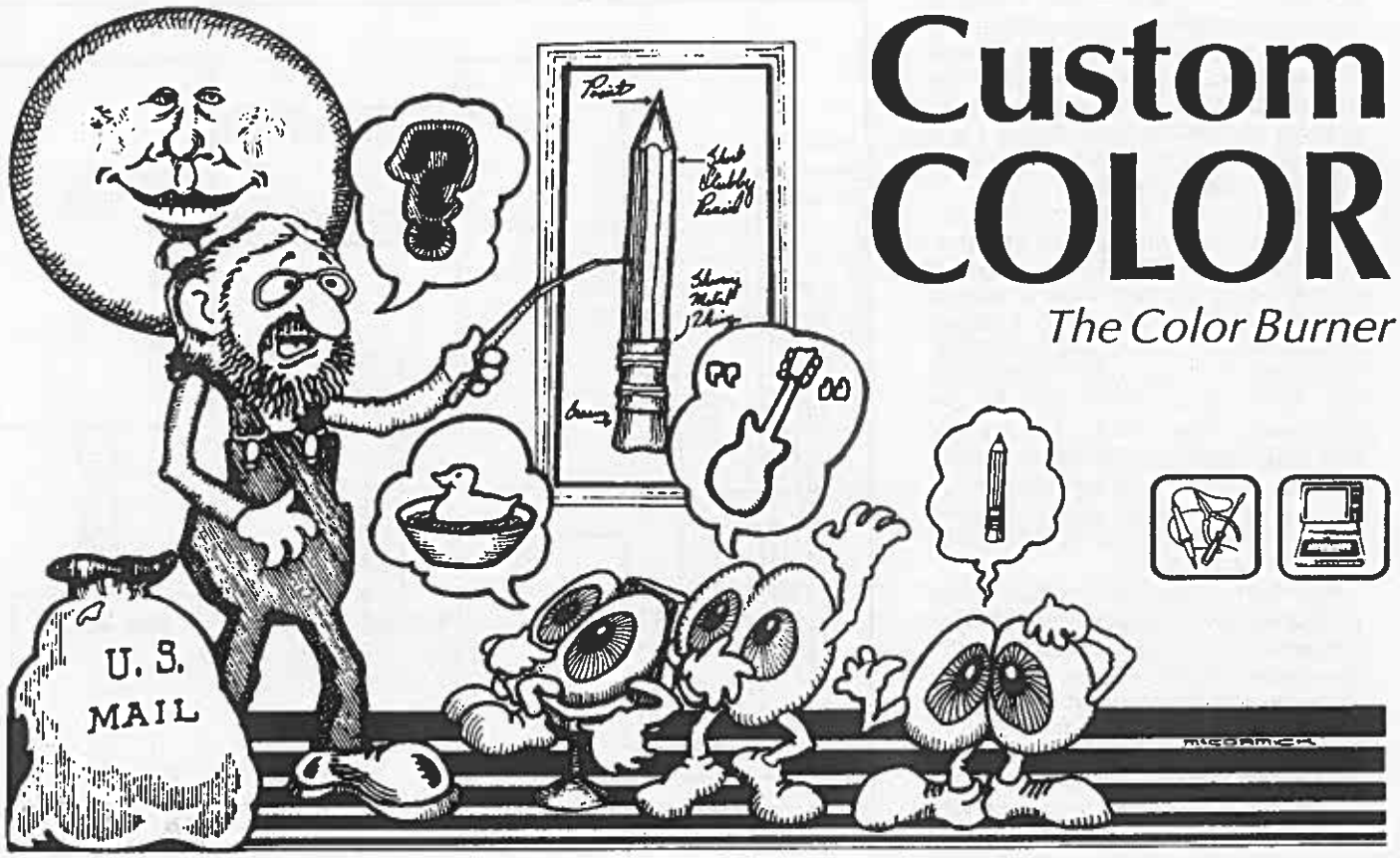
C1	0.1 mF monolithic or disc capacitor	.25
C2	0.1 mF monolithic or disc capacitor	.25
C3	47 mF, 35V radial aluminum electrolytic capacitor	.75
C5	0.1 mF monolithic or disc capacitor	.25
D1	1N4148 or equivalent small-signal diode	.20
D2	1N4148 or equivalent small-signal diode	.20
D3	1N4148 or equivalent small-signal diode	.20
D4	1N5253B 25-volt, 500 mW, 5% zener diode	1.00
D5	1N5251B 22-volt, 500 mW, 5% zener diode	1.00
Q1	2N3904 NPN small-signal switching transistor	.35
Q2	2N3906 PNP small-signal switching transistor	.35
Q3	2N3906 PNP small-signal switching transistor	.35
Q4	2N3906 PNP smal-signal switching transistor	.35
R1	100 ohms, 1/4-watt, 5% resistor	.10
R2	4.7K, 1/4-watt, 5% resistor	.10
R3	4.7K, 1/4-watt, 5% resistor	.10
R4	4.7K, 1/4-watt, 5% resistor	.10
R5	10K, 1/4-watt, 5% resistor	.10
R6	10K, 1/4-watt, 5% resistor	.10
R7	10K, 1/4-watt, 5% resistor	.10
R8	10K, 1/4-watt, 5% resistor	.10
R9	75 ohms, 1/4-watt, 5% resistor	.10
R10	1K, 1/4-watt, 5% resistor	.10
SO1	28-pin, low-insertion-force socket	7.50
SO2	16-pin integrated circuit socket	.75
U1	6821 Peripheral Interface Adapter	5.00
U2	6821 Peripheral Interface Adapter	5.00
U3	7406 open-collector hex inverter	4.50
--	16-pin header for personality modules	3.00
--	14-pin integrated circuit socket	1.00
--	40-pin integrated circuit sockets (2); each	1.00
--	9-volt battery clips (3); each	.15
--	Printed circuit board	20.00
--	Programming software	15.00
--	Complete kit of parts, without software	49.95
--	Complete kit of parts, with software	57.00
--	Assembled and tested, without software	64.95
--	Assembled and tested, with software	69.95

(Note: C4 and C6 are not used.)

The 2716 TEST EPROM is available for $5 plus a $10 deposit and $2.50 shipping and handling.
The 2732 TEST EPROM is available for $5 plus a $10 deposit and $2.50 shipping and handling.
Contact Green Mountain Micro.

Custom COLOR

The Color Burner



by Dennis Kitsz

THIS MONTH'S TOPIC is the Color burner, a project to program erasable, programmable read-only memories (EPROMs).

There's been a lot of interest in EPROM programmers among Color Computer users, mostly because of the computer's ability to use plug-in program-pack cartridges. The EPROM is the perfect device for program packs, for three big reasons: it's cheap ($3 to $4 for a 2K EPROM, $7 to $9 for a 4K EPROM, and prices on 8K and 16K versions coming down all the time); you can program it quickly and easily (less than a minute per 1K of memory); and it's re-usable after being erased under ultraviolet light.

Program-pack cartridges are convenient. They commonly contain machine language programs such as utilities or games, but they can contain Basic programs or just blocks of important, fixed data. There are 16,128 bytes available for the program-pack to use, making it a powerful extension of your Color Computer's software.

Another use for EPROMs is to replace Basic ROMs with other languages, or even to update your Basic ROMs to keep up with Radio Shack's changes and updates.

The main obstacle to using your own EPROMs is usually the fairly infrequent need to program one. Unlike software utilities, EPROM programmers (also called EPROM burners) are seldom used often enough to justify their $100 or higher cost. If that's your situation, this month's Color Burner is for you. For less than $15 in parts, you can put together a software-controlled EPROM programmer which will burn standard 5-volt EPROMs from 2K to 16K in size (2716 through 27128). With a simple change of software and a differently-wired "personality module," Basic ROM compatible 68764 EPROMs can be programmed.

How It Works

A memory device needs an address to access information. To read (get information from) any memory, the address of one memory cell is presented by the processor, together with memory-read control signals. The memory responds with data. To write to (store information in) ordinary read/write memory (RAM), an address is presented with memory-write control signals; data from the processor is stored into that memory cell.

Reading EPROMs is just this straightforward. But writing to — that is, programming — EPROMs is a special case. An address and data are presented together to the EPROM, and must be held steady (stable); while address and data are being held stable, special programming voltages and control signals are turned on for a very specific period of time. After this time, the control signals and voltages are turned off, and the next address and data are selected.

The actual programming time for most popular EPROMs is 50 milliseconds (mS), or roughly 1/20 of a second per cell, making the time per K (1,024 bytes) of memory about 51 seconds. An 8K EPROM the size of the Extended Basic, then, would take nearly seven minutes to program in its entirety. (A nice thing about EPROMs is that they can be programmed one byte at a time — program what you need, and leave the rest for expansion.)

The voltages needed for programming EPROMs are +5 volts (for power), +25 volts (for 2K and 4K EPROMs), and +21 volts (for 8K and larger EPROMs). Certain manufacturers also have self-identification information embedded into their EPROM that can be read using +11 volts, but the Color Burner does not use that feature.

The Color Burner uses three integrated circuits — two 6821 peripheral interface adapters (PIAs) to hold the address and data stable and provide control information, and one 7406 to activate the actual control voltages. Four transistors together with associated resistors and diodes are used by the 7406 to supply

ground, +5, +21 or +25 volts. A plug-in wiring block called a "personality module" routes correct address and voltage information to their respective points on the EPROM. A zero-insertion-force socket holds the EPROM itself. Figure 1 is a block diagram of the activity, and Figure 2 is the complete schematic.

Look at Figure 1. There are seven miscellaneous lines running from the computer to the two PIAs. Address lines A1 and A0 select the four internal control and data registers of the PIA (refer to "Custom Color," March and April 1983, for details of the operation of the 6821 PIA). SCS* (Spare Select) decodes the input/output memory area of the Color Computer (addresses $FF40 to $FF5F), and this, together with address line A2, activates each PIA when addresses $FF40 through $FF43 (PIA #1) or $FF44 through $FF47 (PIA #2) are used.

The RES* (reset) signal synchronizes the PIA and correctly establishes all registers when the power is turned on or the Reset button is pressed; the E clock synchronizes the input/output timing with the computer's processor; and the R/W* (read/write) line identifies a read-from-peripheral or write-to-peripheral state.

The remaining eight lines represent the data to be transmitted to the EPROM socket. The data fed through the first PIA's port A become actual data read from or written to the EPROM. The data fed through port B are used to hold stable the lower eight bits of the EPROM's address (A0 – A7).

Port A of the second PIA transmits the remaining six EPROM address bits; three (A8 – A10) complete the group of addresses common to all EPROMs from 2K to 16K bytes. Address A11 must be added to address 4K of memory, A12 is added to address 8K, and A13 is used for 16K EPROMs. These last three addresses are fed to a personality module, which routes them correctly as needed. Two bits of port A are not used.

The final port (PIA #2, port B) provides the necessary control signals. OE* (output enable), CE* (chip enable) and PGM* (program) are controls used in different ways by the four different EPROM sizes; not all are used by all EPROMs. Because of this, these must also be fed to the personality module for correct routing to the EPROM socket. Bit 3 of port B is not used by the Color Burner.

Bites 4 through 7 of port B select the voltage for programming or reading. For hardware simplicity and low cost, these are software controlled functions. Look at the schematic, Figure 2. When all bits are 0, the +5, +21 and +25 volt lines are turned off, and the transistor is opened up to ground. Combined with high levels on bits 0, 1 and 2, the EPROM can be entirely deselected in software. Table 1 presents the eight bits of port $FF46, and

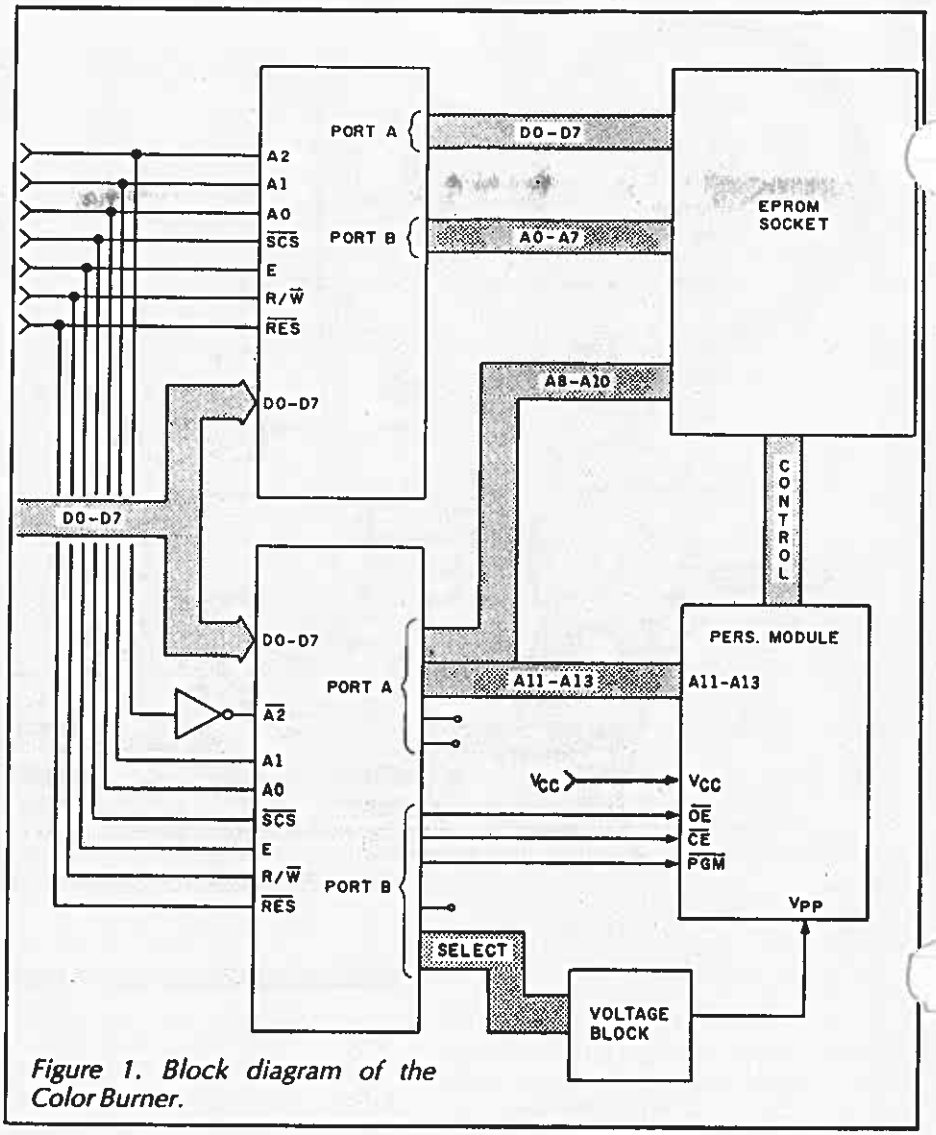


Figure 1. Block diagram of the Color Burner.

Table 1. PORT $FF46

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0V	5V	21V	25V	—	PGM*	CE*	DE*

Programming/Reading Voltages (V_{pp}):

0 = 0	0 = Z	0 = Z	0 = Z	
1 = Z	1 = 5V	1 = 21V	1 = 25V	

(Z = transitor turned off)

Control/Select Lines:

2764	All 2716
27128	2764
	27128

Table 2. Port Values for 2716 through 27128 EPROMs

Bits	7	6	5	4	3	2	1	0	
2716 off	0	0	0	0	(0)	(0)	1	1	= $03
2716 read	1	1	0	0	(0)	(0)	0	0	= $C0
2716 set prgm.	1	0	0	1	(0)	(0)	0	1	= $91
2716 pulse	1	0	0	1	(0)	(0)	010	1	= $91,$93,$91
2732 off	0	0	0	0	(0)	(0)	1	(0)	= $02
2732 read	0	0	0	0	(0)	(0)	0	(0)	= $00
2732 set prgm.	1	0	1	0	(0)	(0)	1	(0)	= $A2
2732 pulse	1	0	1	0	(0)	(0)	0	(0)	= $A2,$A0,$A2
2764 off	0	0	0	0	(0)	0	1	1	= $03
2764 read	1	1	0	0	(0)	1	0	0	= $C4
2764 set prgm.	1	0	1	0	(0)	1	0	0	= $A4
2764 pulse	1	0	1	0	(0)	101	0	0	= $A4,$A0,$A4

(010 = level zero, pulse one, level zero)
(101 = level one, pulse zero, level one)

how they are arranged, and Table 2 shows how the bits are used to select the off, EPROM read, EPROM voltage set, and EPROM program conditions.

Building the Color Burner

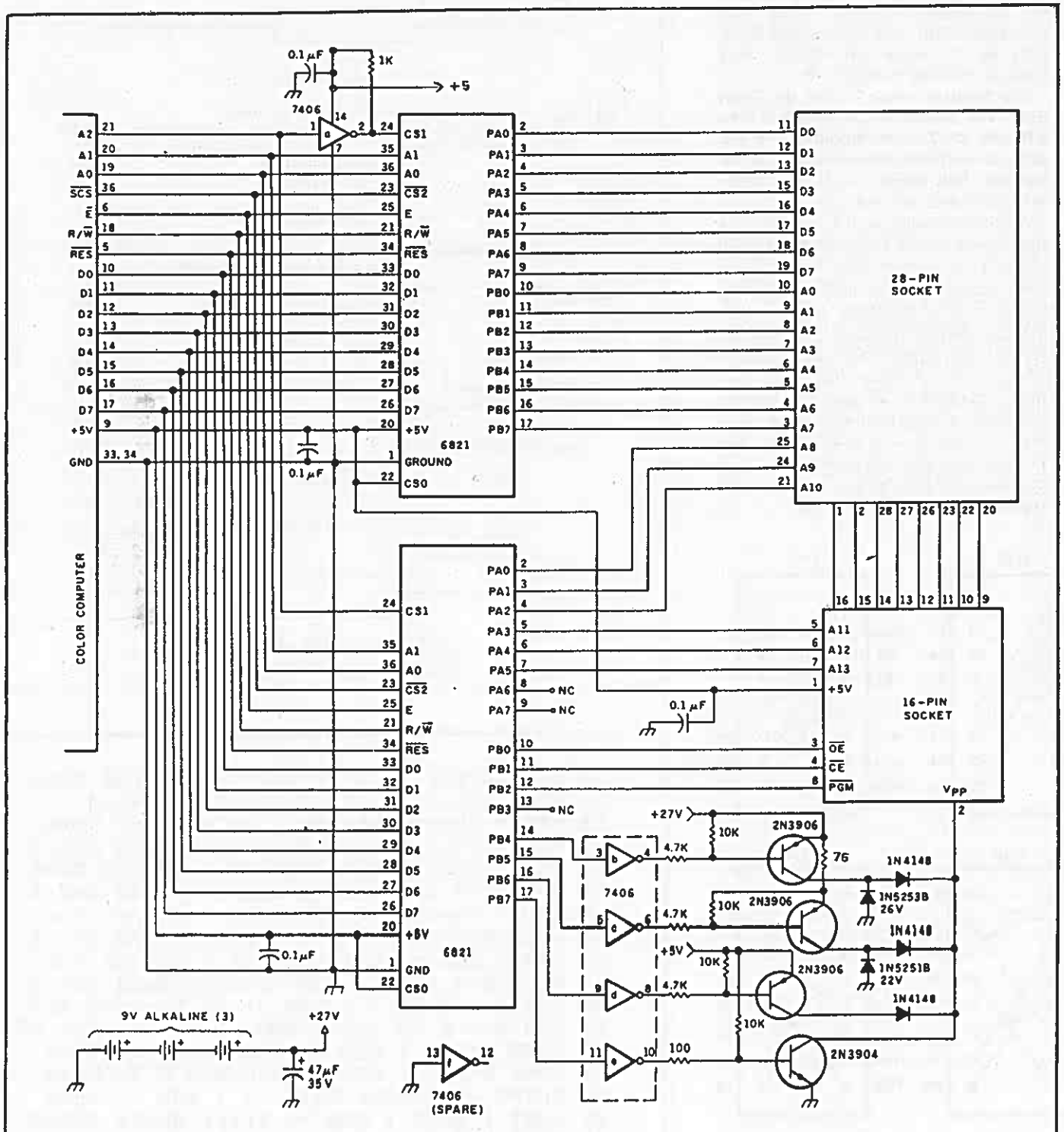
Any neat method of construction is fine for the Color Burner, including wire-wrapping, point-to-point soldering, or printed circuit board (see parts list for availability of boards and kits). Be sure to include the 0.1 mF decoupling capacitors near their respective integrated circuits to assure reliable operation.

Before using the Color Burner, be sure to test voltages, and be careful not to allow the three 9-volt batteries to touch other areas of the assembly. The results of point-to-point wiring should look something like the prototype shown in Photos 1 and 2.

Figure 3 shows personality modules for 2716 through 27128 EPROMs. Obtain 16-pin headers, and wire them as shown in the figure. Figure 4 shows the timing relationships required for reading and writing these EPROMs.

Reading an EPROM

Before attempting to program an EPROM, it's best to try to read one to



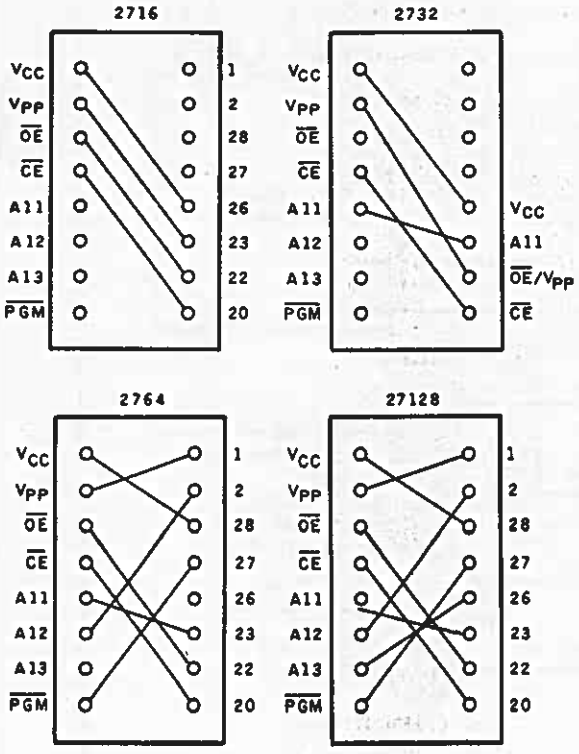
Complete schematic of the Color Burner.

make sure the Color Burner is working. Turn on the computer, insert a programmed 2716 EPROM and the 2716 personality module, and run the program in Listing 1. It should print the contents of the EPROM in hexadecimal.

Testing Voltages

Once the read mode tests correctly, remove the EPROM. POKE &HFF46,&HA0 and read approximately 21 volts at pin 2 of the personality module. Then POKE &HFF46,&H90 and read approximately 25 volts at the same pin. Finally, POKE &HFF46,0 to turn all voltages off.

The forward voltage drop of silicon diode and transistor junctions is 0.6 to 0.8 volts. For 21 volts programming voltage, use a 22-volt zener and a diode for isolation. This results in 22 volts minus two junctions, or approximately 20.5 volts. Test the voltage; if it is outside the high range of 22 volts, add a second diode. If it is outside the low range of 20 volts, obtain a zener diode of slightly higher value or tolerance, or use a germanium diode in place of the 1N4148. For 25 volts programming voltage, use a 27-volt zener plus two diodes for isolation. This results in 27 volts minus three junctions, or approximately 25.0 volts. Test voltage; if it is outside the high range of 26 volts, add a third diode; if it is outside the low range of 24 volts, remove one of the 1N4148 diodes.



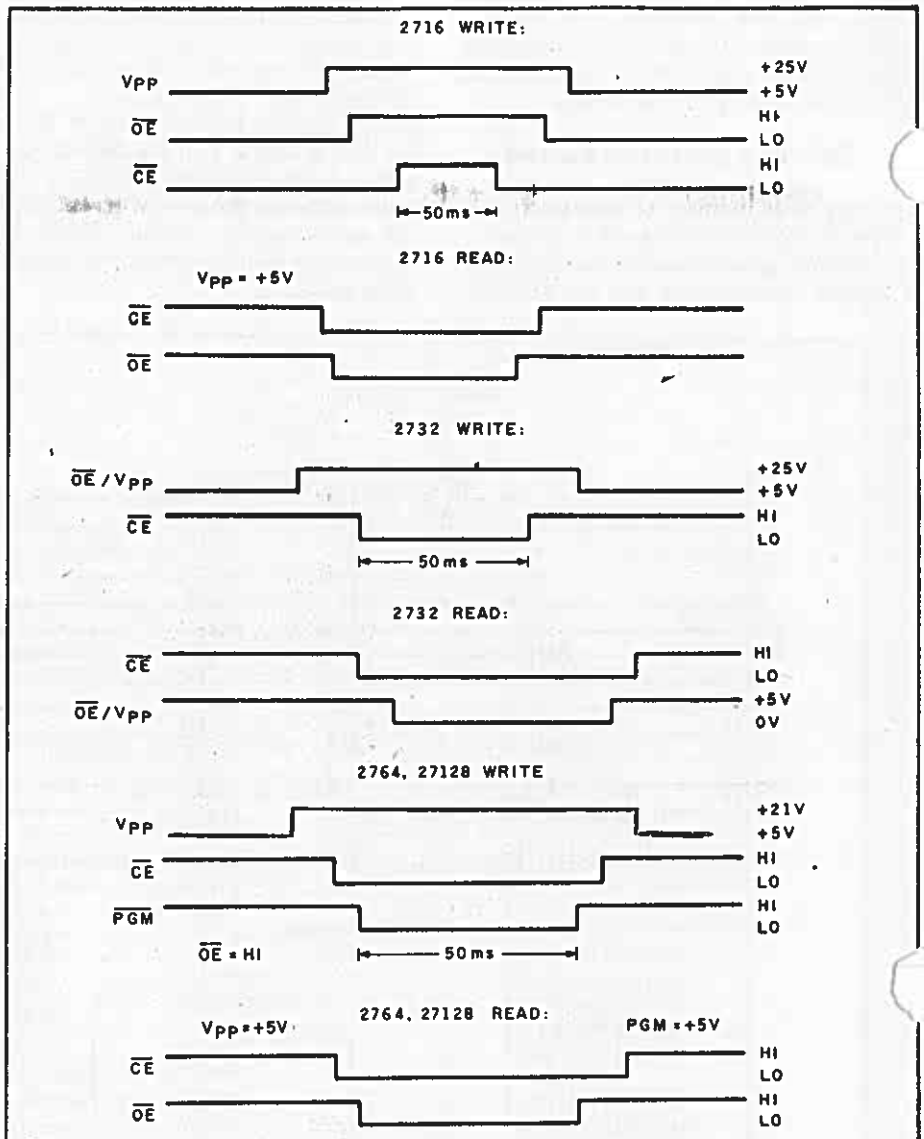


Figure 4. Timing diagrams for 2716 through 27128 EPROMs.

```
10 X = &HFF40 : REM * Point X to PIA Port
11 POKE X+1,0 : POKE X,0 : POKE X+1,4
12 REM * Above sets port 1A = Input Port
13 POKE X+3,0 : POKE X+2,255 : POKE X+3,4
14 REM * Above sets Port 1B = Output Port
15 POKE X+5,0 : POKE X+4,255 : POKE X+5,4
16 REM * Above sets Port 2A = Output Port
17 POKE X+7,0 : POKE X+6,255 : POKE X+7,4
18 REM * Above sets Port 2B = Output Port
19 POKE X+6,&HC0 : REM -> 2716 Read Setup
20 FOR Y = 0 TO 7 : REM -> A8 through A10
21 FOR Z = 0 TO 255 : REM -> A0 through A7
22 POKE X+2,Z : REM -> Address L.S. byte
23 POKE X+4,Y : REM -> Address M.S. byte
24 PRINT HEX$(PEEK(X))" "; : REM -> Data
25 NEXT : NEXT : REM -> Print Whole EPROM
26 POKE X+6,0 : REM -> Active Signals Off
```

****
Color Burner Kit Assembly Instructions
****

Tools you will need: Soldering iron (25-watt), fine solder, wire snips.

WARNING! This kit contains static-sensitive integrated circuits. Leave all parts in the black foam until ready to install them in the circuit, or permanent damage to the devices may result.

WARNING! Move the 28-pin zero-insertion force to the open position before soldering, or the contacts may be soldered closed!

CAUTION! Glass diodes are fragile. To prevent damage to the diodes and to avoid cuts, bend the leads gently -- not sharply -- around a pencil point before inserting in the printed circuit board.

CAUTION! Semiconductor devices can be damaged by heat. Use a 25-watt soldering iron and thin solder, and never solder a part so that it becomes to hot to hold. These devices are rated to be soldered at 300 degrees F for 10 seconds (maximum).

ATTENTION! Be careful not to crush or bend the pins when inserting the 40-pin integrated circuits in their sockets.

ATTENTION! Diodes and transistors look alike. Read the numbers on each part before installing on the printed circuit board.

PLEASE CHECK OFF EACH STEP AS YOU WORK.

Getting Started

( ) Identify the printed circuit board (PC board) solder side (bottom) and component side (top). The component side has component positions printed in white.

( ) Check the contents of the black foam pad against the following parts list:

	Quantity	Part	Description
( )	2	MC6821P	40-pin integrated circuit
( )	1	7406N	14-pin integrated circuit
( )	2	Socket	40-pin low profile
( )	1	Socket	16-pin low profile
( )	1	Socket	28-pin low-insertion force

( ) Check the contents of the plastic bag against the following parts list:

	Quantity	Part	Description
( )	3	Capacitors	0.1 mF monolithic (mkd 104K)
( )	1	Capacitor	47 mF 35V, radial electrolyic
( )	3	Diodes	1N4148 high-speed glass
( )	1	Zener diode	1N5253B, 25 volts, 500 mW
( )	1	Zener diode	1N5251B, 22 volts, 500 mW
( )	1	Transistor	2N3904, NPN, small-signal
( )	3	Transistors	2N3906, PNP, small-signal
( )	1	Resistor	100 ohms, 1/4 watt, 5% (brown-black-brown-gold)
( )	3	Resistors	4.7K, 1/4 watt, 5% (yellow-purple-red-gold)
( )	4	Resistors	10K, 1/4 watt, 5% (brown-black-orange-gold)
( )	1	Resistor	75 ohms, 1/4 watt, 5% (purple-green-black-gold)
( )	1	Resistor	1K, 1/4 watt, 5% (brown-black-red-gold)
( )	1	Header	16 pins
( )	3	Clips	9-volt battery type

Installing Parts

Attention! Solder parts only when you are told to do so!

( ) Place the printed circuit (PC) board with the marked side up, and the gold connector pointing to the left. The directions will be referred to as left, right, far (away from you) and near (closest to you).

( ) Pick up one of the 40-pin integrated circuit sockets. One end is distinguished from the other by a notch or dot.

( ) Insert the socket into the position on the PC board marked U1, at the near side of the board. The notch or dot should point in the same direction as the notch printed on the PC board.

( ) Bend over two diagonally opposite pins of the socket on the bottom of the board to hold the socket in place for soldering.

( ) Solder all 40 pins of the socket to the bottom of the PC board.

( ) Pick up, orient, and insert the other 40-pin socket in the position marked U2 on the PC board.

( ) Solder all 40 pins of the socket to the bottom of the PC board.

( ) Pick up the 16-pin integrated circuit socket, and locate the notched or dotted end.

( ) Insert this socket into the position marked SO2 on the PC board. Note that the notch points to the far side of the PC board.

( ) Bend over two diagonally opposite pins to hold the socket to the board.

( ) Solder all 16 pins of the socket to the bottom of the PC board.

( ) Pick up the 7406N integrated circuit, and locate the notched or dotted end.

( ) Insert this integrated circuit into the position marked U3 on the PC board. The notch points to the near side of the board.

( ) Bend over two diagonally opposite pins to hold the integrated circuit to the board.

( ) Solder all 14 pins of the integrated circuit to the bottom of the PC board.

( ) Using a pair of sharp, high-quality wire snips, clip the excess protruding through the bottom of the board from the three sockets and the integrated circuit. Don't cut all the way down; just clip the sharp points off.

( ) Pick up the 28-pin low-insertion-force integrated circuit socket, and identify the end with the handle, lever, or screwdriver slot (any one of several types of socket are shipped with the Color Burner).

( ) If the socket has a handle, orient the handle toward the near side of the board.

( ) If the socket has a screwdriver slot, orient the slot toward the far side of the board.

( ) Open the socket by lifting the lever or turning a screwdriver in the slot.

( ) Insert the socket into the position marked SO1.

( ) Solder all 28 pins of the socket to the bottom of the PC board.

( ) Pick up the 100-ohm resistor (brown-black-brown-gold) and hold it between thumb and forefinger of one hand. Pull the resistor between the thumb and forefinger of your other hand, bending the leads 90 degrees, along a gentle curve.

( ) Insert the 100-ohm resistor into the position marked R1.

( ) Hold the resistor in place, and solder it to the bottom of the board.

( ) Clip the excess leads from the bottom of the board.

( ) Pick up the 4.7K-ohm resistor (yellow-purple-red-gold), bend it as before, and insert it into the position marked R2.

( ) Hold in place, solder, and clip the excess leads.

( ) Pick up the next 4.7K-ohm resistor, bend, insert in the position marked R3, hold in place, solder and clip.

( ) Pick up the last 4.7K-ohm resistor, bend, insert in the position marked R4, hold in place, solder and clip.

( ) Pick up a 10K-ohm resistor (brown-black-orange-gold), bend it, and insert it into the position marked R5.

( ) Hold in place, solder, and clip the excess leads.

( ) Pick up, bend, and insert the second 10K-ohm resistor into the position marked R6.

( ) Hold in place, solder, and clip the excess leads.

( ) Pick up, bend, and insert the third 10K-ohm resistor into the position marked R7.

( ) Hold in place, solder, and clip the excess leads.

( ) Pick up, bend, and insert the last 10K-ohm resistor into the position marked R8.

( ) Hold in place, solder, and clip the excess leads.

( ) Pick up, bend, and insert the 75-ohm resistor (purple-green-black-gold) into the position marked R9.

( ) Hold in place, solder, and clip the excess leads.

( ) Pick up, bend, and insert the 1K-ohm resistor (brown-black-red-gold) into the position marked R10.

( ) Hold in place, solder, and clip the excess leads.

( ) Pick up one of the 0.1 mF capacitors (yellow or blue rectangles marked 104K).

( ) Insert it into the position marked C1, solder, and clip the excess leads.

( ) Pick up the second 0.1 mF capacitor, insert into position C2, solder, and clip the excess leads.

( ) Pick up the last 0.1 mF capacitor, insert into position C5, solder, and clip the excess leads.

( ) Pick up the 47 mF electrolytic capacitor, and

identify the negative and positive leads.

( ) Locate the position marked C3 on the PC board, and identify the positive hole.

( ) Insert the 47 mF capacitor into position C3, with the negative and positive leads oriented correctly.

The following components (transistors and diodes) are delicate and heat-sensitive. Re-read the warnings and cautions are the beginning of these instructions before continuing.

( ) Pick up the 2N3904 transistor, and identify the flat side.

( ) Insert the 2N3904 transistor in the position marked Q1, leaving about 1/4 inch of space between the body and the PC board.

( ) Be sure the flat side of the transistor matches the flat side on shown on the circuit board. Solder very carefully, being sure not to make a bridge of solder between the solder points.

( ) Clip the excess leads from the bottom of the board.

( ) Pick up and insert a 2N3906 transistor into the position marked Q2, leaving about 1/4 inch of space between the body and the PC board.

( ) Orient the flat side correctly, and solder.

( ) Clip the excess leads from the bottom of the board.

( ) Pick up and insert a 2N3906 transistor into the position marked Q3, leaving about 1/4 inch of space.

( ) Orient the flat side correctly, and solder.

( ) Clip the excess leads from the bottom of the board.

( ) Pick up and insert the last 2N3906 transistor into the position marked Q4, leaving about 1/4 inch of space.

( ) Orient the flat side, and solder.

( ) Clip the excess leads from the bottom of the board.

( ) Examine the 12 solder points you have just made. These are the tightest solder locations on the board, so be sure there are no solder splashes or bridges. If you find a problem, re-heat the connection to correct it.

( ) Pick up a 1N4148 diode and locate the banded end.

( ) Find position D1 on the PC board, and locate the banded end.

( ) Bend and insert the 1N4148 diode into position D1, orienting the banded end correctly.

( ) Solder and clip the excess leads from the bottom of the board.

( ) Pick up, bend, orient and insert the second 1N4148 diode into the position marked D2.

( ) Solder and clip the excess leads.

( ) Pick up, bend, orient and insert the last 1N4148 diode into the position marked D3.

( ) Solder and clip the excess leads.

( ) Pick up the 1N5253B diode, and identify the banded end.

( ) Bend, orient and insert the 1N5253B diode into the position marked D4.

( ) Solder and clip the excess leads.

( ) Pick up, bend, orient and insert the 1N5251B diode into the position marked D5.

( ) Solder and clip the excess leads.

( ) Pick up one of the 9-volt battery clips and identify the red and black leads.

( ) At the far right side of the board, behind the 16-pin integrated circuit socket, locate six holes. Call these holes, from left to right, #1 to #6.

( ) Insert the red lead in hole #1, marked "+", and solder.

( ) Insert the black lead in hole #2, marked "-", and solder

( ) Pick up the next battery clip, insert the red lead in hole #3 marked "+", and solder.

( ) Insert the black lead in hole #4 marked "-", and solder.

( ) Pick up the last battery clip, insert the red lead in hole #5 marked "+", and solder.

( ) Insert the black lead in hole #6 marked "-", and solder.

( ) Clip the excess of these six leads from the bottom of the board.

( ) Pick up one of the MC6821P integrated circuits, and identify the notched end.

( ) Orient this integrated circuit above the 40-pin socket U1, with the notch pointing in the same direction as the socket, toward the near side of the board.

( ) Being extremely careful not to bend or crush pins, insert the integrated circuit in the socket by pressing firmly and evenly on front and back of the integrated circuit.

( ) Pick up the other MC6821P integrated circuit, orient it, and insert it in socket U2.

( ) Your initial construction is complete; you should have one 16-pin header left.

( ) Positions C4 and C6 should be empty.

( ) Recheck all socket orientation.

( ) Recheck all integrated circuit orientation.

( ) Recheck all transistor numbers and orientation.

( ) Recheck all diode numbers and orientation.

( ) Recheck all resistor values.

( ) Recheck all capacitor values and the orientation of the 47 mF electrolytic.

( ) Recheck all solder points for complete, smooth connections.

( ) If possible, have a friend proofread your work.

Testing

Ideally, testing should be done with a voltmeter and an oscilloscope. Since the Color Burner has been designed to be relatively trouble-free when properly constructed, some of the testing suggested in this section is recommended but not essential.

Important: When following these directions, make NO assumptions. For example, do not install batteries until you are told to do so.

This testing section requires an Extended Color Basic computer.

( ) Turn the Color Computer OFF.

( ) Insert the Color Burner board, gold edge connector to the left, and component side up.

( ) Turn the Color Computer ON and OFF quickly, keeping it on for NO MORE THAN ONE SECOND. If the screen showed the usual green square, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Turn the Color Computer ON. Your sign-on message should appear (Extended Color Basic). If the screen shows the sign-on message, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Turn a digital voltmeter to the scale capable of displaying voltages between 3 and 30 volts.

( ) Connect the ground (black) lead to the Color Burner far side lead of D5 (opposite the banded end).

( ) Connect the signal (red) lead to the Color Burner right side lead of D1 (the banded end).

( ) Connect three FRESH 9-volt ALKALINE batteries to the battery clips.

( ) Your computer is still on. If it is not, turn it on now.

( ) Type the following Basic lines:

```
POKE&HFF47,0              (ENTER)
POKE&HFF46,255            (ENTER)
POKE&HFF47,4              (ENTER)
POKE&HFF46,0              (ENTER)
```

( ) The voltage should read 0 volts. If it reads 0 volts, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Type this Basic line:

```
POKE&HFF46,&HC0           (ENTER)
```

( ) The voltage should read between 3.5 and 5 volts. If it reads between 3.5 and 5 volts, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Type this Basic line:

```
POKE&HFF46,&HA0           (ENTER)
```

( ) The voltage should read between 20 and 22.5 volts. If it reads between 20 and 22.5 volts, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Type this Basic line:

```
POKE&HFF46,&H90           (ENTER)
```

( ) The voltage should read between 24 and 26 volts. If it reads between 24 and 26 volts, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Type this Basic line:
POKE&HFF46,0 <ENTER>

If the Color Burner passes the tests above, it has passed the basic electrical functions and will not damage itself or the computer. Complete testing requires programming software and an EPROM to program; an oscilloscope can be used for the optional tests. The following steps will copy a type 2716 EPROM using the standard Color Burner software; 32K Extended Color Basic is required, plus a TEST EPROM and the 2716 personality module.

( ) Insert the Color Burner software into the recorder, and set it to play.

( ) Type this Basic line:
PMODE0:PCLEAR1:CLOAD <ENTER>

( ) The software will load; when "OK" returns, type:
RUN <ENTER>

( ) The remaining software will load, and the menu will be presented. Touch 1; the EPROM selection menu will be presented.

( ) Touch 1, followed by M; the main menu will return. You have selected the type 2716 EPROM.

( ) Open the low-insertion-force socket.

( ) Insert the TEST EPROM in the socket. The notch faces the far side of the board, and the EPROM sits in the near 24 pins of the 28-pin socket.

( ) Close the low-insertion-force socket.

( ) Insert a 2716 personality module in the 16-pin socket. The angled edge of the module points to the far side of the board.

( ) Touch selection 7 to read the TEST EPROM.

( ) In response to start address, enter 0000.

( ) Be ready to read the screen. In response to end address, enter 0800.

( ) The EPROM will be read into memory. As you read the screen, you should see consecutive hexadecimal numbers flashing by.

( ) Touch M to return the main menu.

( ) Touch 3 to examine memory.

( ) In response to the prompt, enter 4000.

( ) The screen should display rows of hexadecimal numbers, four each of FF, four each of FE, etc. If you see these descending hexadecimal numbers, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLE-SHOOTING SECTION.

( ) Press M to return the menu.

( ) Open the low-insertion-force socket and remove the TEST EPROM.

( ) Insert a blank type 2716 EPROM, and close the socket.

( ) Press 5 to burn an EPROM.

( ) In response to the verify erasure prompt, press Y.

An erased EPROM should be verified. If it is not erased, replace it with an erased EPROM now.

( ) Press C to continue with the burning process.

( ) OPTIONAL STEP: Connect the ground lead of an oscilloscope to the far (unbanded) side of D5.

( ) OPTIONAL STEP: Connect the signal lead of an oscilloscope to the right (banded) side of D2.

( ) OPTIONAL STEP: Set the oscilloscope to 50 mS/ division horizontal, 20 volts/division vertical, DC, internal sync. If you have a storage scope, prepare it to store one sweep.

( ) In response to starting address, enter 0000.

( ) In response to ending address, enter 0800. The EPROM will begin programming, displaying the addresses as it goes. The process will take slightly less than 2 minutes.

( ) OPTIONAL STEP: The scope should display a 50 mS pulse of between 24 and 26 volts, falling to 5 and 0 volts between pulses. Store and display one sweep on a storage scope.

( ) When programming is complete, respond to the verify prompt by pressing Y. If the verification passes, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Open the socket and remove the EPROM.

( ) Remove the 2716 personality module.

The 25-volt programming section has been tested. The 21-volt programming section may be verified by programming a type 2732 EPROM. The following steps will copy a type 2732 EPROM using the standard Color Burner software; 32K Extended Color Basic is required, plus a TEST EPROM and the 2732 personality module. For these tests, it is assumed the Color Burner is in place and the software is operating from the previous tests.

( ) You are at the main menu. Touch 1 to select an EPROM.

( ) Touch 2, followed by M; the main menu will return. You have selected the type 2732 EPROM.

( ) Open the low-insertion-force socket.

( ) Insert the TEST EPROM in the socket. The notch faces the far side of the board, and the EPROM sits in the near 24 pins of the 28-pin socket.

( ) Close the low-insertion-force socket.

( ) Insert a 2732 personality module in the 16-pin socket. The angled edge of the module points to the far side of the board.

( ) Touch selection 7 to read the TEST EPROM.

( ) In response to start address, enter 0000.

( ) Be ready to read the screen. In response to end address, enter 1000.

( ) The EPROM will be read into memory. As you read the screen, you should see consecutive hexadecimal numbers flashing by.

( ) Touch M to return the main menu.

( ) Touch 3 to examine memory.

( ) In response to the prompt, enter 4000.

( ) The screen should display rows of hexadecimal numbers, four each of FF, four each of FE, etc. If you see these descending hexadecimal numbers, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLE-SHOOTING SECTION.

( ) Press M to return the menu.

( ) Open the low-insertion-force socket and remove the TEST EPROM.

( ) Insert a blank type 2732 EPROM, and close the socket.

( ) Press 5 to burn an EPROM.

( ) In response to the verify erasure prompt, press Y. An erased EPROM should be verified. If it is not erased, replace it with an erased EPROM now.

( ) Press C to continue with the burning process.

( ) OPTIONAL STEP: Connect the ground lead of an oscilloscope to the far (unbanded) side of D5.

( ) OPTIONAL STEP: Connect the signal lead of an oscilloscope to the right (banded) side of D2.

( ) OPTIONAL STEP: Set the oscilloscope to 50 mS/ division horizontal, 20 volts/division vertical, DC, internal sync. If you have a storage scope, prepare it to store one sweep.

( ) In response to starting address, enter 0000.

( ) In response to ending address, enter 1000. The EPROM will begin programming, displaying the addresses as it goes. The process will take approximately 4 minutes.

( ) OPTIONAL STEP: The scope should display a 50 mS pulse of between 20 and 22 volts, falling to 5 and 0 volts between pulses. Store and display one sweep on a storage scope.

( ) When programming is complete, respond to the verify prompt by pressing Y. If the verification passes, continue with the next step. Otherwise, your Color Burner has FAILED TESTING. Go to the TROUBLESHOOTING SECTION.

( ) Open the socket and remove the EPROM.

( ) Remove the 2732 personality module.

( ) Remove the three 9-volt batteries.

Your Color Burner is now tested and fully functioning. Refer to the main part of this manual for further information and operation instructions.

Troubleshooting

Problems with the Color Burner will be manifested in five major ways: locking up the computer; faulty EPROM reading; improper programming voltages; faulty EPROM programming; and "smoke and flame" during voltage testing.

Computer Lockup / Causes	Likelihood:
* Bridges, splashes or unsoldered connections.	High
* PC board not inserted correctly.	Low
* Defective MC6821 integrated circuit.	Low

Faulty EPROM Reading / Causes	Likelihood:
* EPROM inserted incorrectly.	High
* EPROM socket not fully closed.	High
* Incorrect personality module being used.	High
* Incorrect EPROM selected from the software.	Medium
* Missing or poor solder connections of EPROM socket.	Medium
* Missing or poor solder connections on MC6821 socket.	Medium
* EPROM is blank or damaged.	Low

Improper Programming Voltages / Causes	Likelihood:
* Shorts or poor solder connections on transistors.	High
* Shorts or poor solder connections on diodes.	High
* Incorrect personality module being used.	High
* Diodes inserted in reverse.	High
* Other than alkaline batteries being used.	High
* Shorts or poor solder connections on 7406N.	Medium
* Batteries weak.	Medium
* Typing error in Basic test lines.	Medium
* Heat damage to diodes or transistors.	Medium
* Meter incorrectly calibrated.	Low

Faulty EPROM Programming / Causes (where programming voltages are OK)	Likelihood:
* Shorts or poor solder connections on EPROM socket.	High
* Incorrect personality module being used.	High
* Wrong EPROM selected via software.	High
* EPROM not erased before programming.	High
* Program information incorrectly stored.	Medium
* Wrong area of EPROM selected to program.	Medium
* Batteries weakened (since voltage testing).	Medium
* Damaged MC6821.	Low
* Damaged EPROM.	Low

Smoke and Flame on Testing Voltages / Causes	Likelihood:
* Shorts on transistors.	High
* Shorts on diodes and/or zener diodes.	Medium
* Damaged 7406N.	Medium

****

PROGRAM LISTINGS

****

```
1 PMODE0:PCLEAR1:CLEAR200,15616:CLOADM"BLAST":A=128:A$=STRING$(32,A):B$=STRING$(9,A):C$=STRING$(3,A):D$=CH
R$(A):XP=&H3DCD:XE=&H3E4B:XV=&H3E8C:XR=&H3EED:V=&H3D00:FORX=0TO8:READE$(X):NEXT:DATANONE,2716,2732,2764,27
128,68764,SPCL1,SPCL2,SPCL3
2 S=&H4000:H$="&H":DIMD(28):FORX=0TO8:READE(X):NEXT:PO=&HFF46:FORX=0TO28:READD(X):NEXT:DATA0,2048,4096,819
2,16384,8192,0,0,0 ,0,25,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,165,0,145,125,0,0,105,85,65
3 POKEPO+1,0:POKEPO,255:POKEPO+1,4:POKEPO,0:W=0:POKEV+1,0:POKEV+2,0:POKEV+3,64:POKEV+4,0:GOSUB7:GOSUB8
4 GOSUB25:IFX$="H"ORX$="h"THEN26
5 ONVAL(X$)GOTO66,71,88,121,129,144,177
6 GOTO4
7 CLS0:PRINT@0,A$"Color Burner Monitor/Blaster 2.3Copyright (c) 1984  D. B. Kitsz"A$;:RETURN
8 PRINT@128,B$" ))) MENU ((( "B$A$C$" 1.  Set EPROM type.      "C$C$" 2.  Load object tape.    "C$C$" 3.
Examine object code. "C$C$" 4.  Save object tape.    "C$C$" 5.  Blast EPROM.         "C$;
9 PRINTC$" 6.  Define new EPROM.    "C$C$" 7.  Read EPROM program.  "C$A$" Touch number or "CHR$(34)"H"CHR
$(34)"  for HELP.";STRING$(31,A);:POKE1535,A:RETURN
10 PRINT"Starting Address (Hexadecimal):":INPUT"Address ----)";M$:MM=VAL(H$+M$)
11 PRINT"Ending Address (Hexadecimal):":INPUT"Address ----)";N$:NN=VAL(H$+N$)
12 IFE=0THEN23ELSEEE=E(E)
13 IFW=1THEN17
16 IFNN)EE THEN21ELSEIFMM)=NN THENGOSUB7:GOTO10
17 MS=INT(MM/256):MT=MM-256*MS
18 NS=INT(NN/256):NT=NN-256*NS
19 POKEV,(E-1)*4:POKEV+1,MS:POKEV+2,MT:POKEV+5,NS:POKEV+6,NT
20 RETURN
21 PRINT@384,"Program length exceeds size of  EPROM.  Please indicate correct size of EPROM (Use Menu #1).
   Press (ENTER) to return Menu.  ";:POKE1535,96
22 GOTO24
23 PRINT"EPROM not defined (Use Menu #1).Press (ENTER) to return Menu.  ";
24 GOSUB25:QQ=1:RETURN
25 X$=INKEY$:IFX$=""THEN25ELSERETURN
26 GOSUB7:PRINTA$C$" HELP is available for... "C$A$C$"1. Setting EPROM type.    "C$C$"2. Loading object co
de.   "C$C$"3. Examining object code. "C$C$"4. Saving object code.    "C$C$"5. Blasting an EPROM.    "C$;
27 PRINTC$"6. Defining a new EPROM.  "C$C$"7. Reading an EPROM.      "C$A$" Touch number or "CHR$(34)"M"CH
R$(34)" for MENU. ";
28 GOSUB25:IFX$="M"ORX$="m"THEN3
29 ONVAL(X$)GOTO31,34,40,44,47,56,62
30 GOTO28
31 CLS0:PRINTD$C$" 1.  SETTING EPROM TYPE "D$C$A$"You must define the type EPROM  you are using before it
can be  programmed.  This software will program five standard EPROMs:  2716 (2K), 2732 (4K), 2764 (8K),27
128 (16K) and 68764 (8K). For ";
32 PRINT"other types of EPROMs, an EPROM definition menu is provided (seeSelection #6).  To program any  E
PROM correctly, its respective personality module is plugged  into the Color Burner.        Touch M to
return the Menu.    ";
33 GOSUB25:IFX$="M"ORX$="m"THEN3ELSE33
34 CLS0:PRINTD$C$" 2. LOADING OBJECT CODE "D$C$A$"Object code is the binary code  burned into the EPROM.
The codeis prepared with a monitor or aneditor/assembler, and saved in  binary format for use with the  BA
SIC CLOADM command. The binary";
35 PRINT"code must have its origin (ORG) specified where you expect it toappear in the final EPROM.  The O
RG for Color Computer ROM packsis $C000 (49152).          Touch M for Menu, C to Continue.";
36 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$="C"ORX$="c"THEN37ELSE36
37 CLS0:PRINTD$C$" 2. LOADING OBJECT CODE "D$C$A$"You can also use Color Burner's hex monitor to prepare o
bject  code.  The memory range of the  monitor is hexadecimal 4000 to  7FFF, which represents 0000 to  3F
FF in the final EPROM.  The  ";
38 PRINT"final option for loading object code is by copying the contents of a programmed EPROM.        T
ouch M for Menu, R to Repeat.  ";
39 GOSUB25:IFX$="R"ORX$="r"THEN34ELSEIFX$="M"ORX$="m"THEN3ELSE39
40 CLS0:PRINTD$C$" 3. EXAMINE OBJECT CODE "D$C$A$"Object code is the binary data  which will be burned int
o the  EPROM.  It can be a program or  tables or other information. TheColor Burner monitor displays  th
e binary data in hexadecimal  ";
41 PRINT"format.  Move the cursor with  the four keyboard arrows.  Placethe cursor over the bytes you  w
ish to edit.  After completing any changes, press ENTER.  TouchM to return Menu with no change.";
42 PRINT"Press M to return to the Menu  ";
43 GOSUB25:IFX$="M"ORX$="m"THEN3ELSE43
44 CLS0:PRINTD$C$" 4.  SAVING OBJECT CODE "D$C$A$"After creating or editing the  binary object code, you
may wishto save it to tape.  This editedobject code will be saved using the address $4000, where it has be
en stored in the Color Burner ";
45 PRINT"memory buffer.  When reloading  edited Color Burner codes, you  MUST specify $4000 as the origin(
ORG) address.                Touch M to return the Menu.    ";
46 GOSUB25:IFX$="M"ORX$="m"THEN3ELSE46
47 CLS0:PRINTD$C$" 5.  BLAST (BURN) EPROM "C$D$A$"To burn an EPROM, you must be  certain that you have co
rrectly followed two instructions:    1.  You must have defined the      EPROM you are burning.    2.
  You must have inserted the  ";
48 PRINT"  correct personality module    in the socket on the Color    Burner.                  I
f you have not defined your  EPROM, the software will ask youto do so.                  Touch M for
 Menu, C to Continue.";
49 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$="C"ORX$="c"THEN50ELSE49
```

30-R0685

```
50 CLS0:PRINTD$C$" 5.  BLAST (BURN) EPROM "C$D$A$"You are responsible for using  the correct personality
module. USING THE INCORRECT MODULE MAY  PERMANENTLY DAMAGE THE EPROM!  Refer to your manual if you are un
sure of how to insert the     ";
51 PRINT"personality modules.  It is besttto insert the personality modulebefore beginning your program- m
ing session.  If you change   modules under power, do not pullthe Color Burner sideways.      Touch M for
 Menu, C to Continue.";
52 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$="C"ORX$="c"THEN53ELSE52
53 CLS0:PRINTD$C$" 5.  BLAST (BURN) EPROM "C$D$A$"You should insert the EPROM fromthe main Menu.  Insert t
he EPROMSLOWLY AND CAREFULLY, OR YOU MAYDAMAGE BOTH EPROM AND COMPUTER! Only THEN connect the batteries.Se
e the manual to identify which";
54 PRINT"way to insert the EPROM in the  socket.  NEVER PRESS RESET OR  TURN OFF THE POWER WHEN PROGRAM-M
ING IS IN PROGRESS!  You may  BREAK, but programming will stopat that point.                Press M for
 Menu, R to Repeat.  ";
55 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$="R"ORX$="r"THEN47ELSE55
56 CLS0:PRINTD$C$" 6.  DEFINE A NEW EPROM "C$D$A$"To define a new EPROM, you need the specifications of th
e EPROM and an understanding of how to  read them.  You will be asked todefine each pin of the EPROM.  To
uch M for Menu, C to Continue.";
58 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$="C"ORX$="c"THEN59ELSE58
59 CLS0:PRINTD$C$" 6.  DEFINE A NEW EPROM "C$D$A$"When the EPROM pins are defined,a wiring diagram for the
 header (personality module) will be    shown.  Continue when the module";
60 PRINT"has been wired and is in place  on the Color Burner.  Refer to  your documentation for EPROMs  n
ot included in this software.  Touch M for Menu, R to Repeat.  ";
61 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$="R"ORX$="r"THEN56ELSE61
62 CLS0:PRINTD$C$" 7.  READING AN EPROM   "C$D$A$"To read an already programmed  EPROM, you need to defin
e its  size (Menu #1).  After defining the size, indicate the start andending addresses of the EPROM tobe
 read into memory.  The data  ";
63 PRINT"will be read quickly into the  memory, where it can be examinedand edited.                  T
ouch M to return to the Menu.  ";
64 GOSUB25:IFX$="M"ORX$="m"THEN3ELSE64
66 GOSUB7:PRINT" (*) EPROM DEFINED = "E$(E)" (*)"
67 PRINTA$C$"1.  2716 (2K X 8)        "C$C$"2.  2732 (4K X 8)        "C$C$"3.  2764 (8K X 8)        "C$
C$"4.  27128 (16K X 8)       "C$C$"5.  68764 (8K X 8)       "C$C$"6.  SPECIAL (MENU #6)     "C$A$"  Touch
 No. or "CHR$(34)"M"CHR$(34)" for Menu.
68 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$("1"ORX$)"8"THEN68
69 E=VAL(X$):GOTO66
71 GOSUB7:PRINT"  (*) OBJECT CODE LOADING (*)   "A$;
72 IFE=0THENPRINT"YOUR EPROM HAS NOT BEEN DEFINED.PLEASE USE MENU SELECTION #1.";FORN=1TO2000:NEXT:GOTO3
73 PRINT"What is the origin of the objectcode on the tape?"
74 INPUT"Enter a hexadecimal value from  $0000 to $FFFF -----)";D$
75 O=G-VAL(H$+D$):IFO(0THENO=O+65536
76 PRINT"Where in your "E$(E)" EPROM is the object code to be loaded?"
77 INPUT"Enter a hexadecimal value from  $0000 to $FFFF -----)";P$
78 P=VAL(H$+P$)
79 PRINT"Insert tape and touch L to load,M to return the Menu.          ";:POKE1535,96
80 GOSUB25:IFX$="M"ORX$="m"THEN3ELSEIFX$="L"ORX$="l"THEN81ELSE80
81 GOSUB7:PRINT"  (*) OBJECT CODE LOADING (*)   "A$;
82 PRINT"Enter the filename of the objectcode; a filename is required."
83 INPUT"Filename -----)";F$
84 IFF$=""THEN82
85 CLOADM F$,O+P
86 PRINT"Object Code Loaded.";FORN=1TO1000:NEXT:GOTO3
88 GOSUB7:PRINT"   (*) MEMORY EXAMINATION (*)   "A$;
89 PRINT"Starting Address (Hexadecimal):";:INPUT"Address ----)";M$:MM=VAL(H$+M$):CLS:FORN=1TO16
90 AD$=HEX$(MM):IFLEN(AD$)(4THENAD$=STRING$(4-LEN(AD$),"0")+AD$
91 PRINTAD$" **";:FORT=0TO7
92 DD=PEEK(MM+T):DD$=HEX$(DD):IFLEN(DD$)(2THENDD$="0"+DD$
93 PRINT" "DD$;:NEXT
94 IFN(16THENPRINT" ";
95 MM=MM+8:NEXT
96 Z=1032
97 Z1=PEEK(Z):POKEZ,A
98 X$=INKEY$:IFX$=""THENPOKEZ,Z1:GOTO97
99 IFX$=CHR$(8)THENPOKEZ,Z1:GOTO108
100 IFX$=CHR$(9)THENPOKEZ,Z1:GOTO111
101 IFX$=CHR$(10)THENPOKEZ,Z1:GOTO114
102 IFX$=CHR$(94)THENPOKEZ,Z1:GOTO115
103 IFX$=CHR$(13)THENPOKEZ,Z1:GOTO116
104 IFX$="M"ORX$="m"THEN3
105 IFX$)="0"ANDX$(="9"THENZ1=ASC(X$)+64
106 IFX$)="A"ANDX$(="F"THENZ1=ASC(X$)
107 POKEZ,Z1:GOTO111
108 ZQ=(Z-1024)AND31:IFZQ(9THEN97
109 IFZQ=9ORZQ=12ORZQ=15ORZQ=18ORZQ=21ORZQ=24ORZQ=27ORZQ=30THENZ=Z-1ELSEZ=Z-2
```

31-R0685

```
110 GOTO97
111 ZQ=(Z-1024)AND31:IFZQ)29THEN97
112 IFZQ=8ORZQ=11ORZQ=14ORZQ=17ORZQ=20ORZQ=23ORZQ=26ORZQ=29THENZ=Z+1ELSEZ=Z+2
113 GOTO97
114 IFZ)1504THEN97ELSEZ=Z+32:GOTO97
115 IFZ(1056THEN97ELSEZ=Z-32:GOTO97
116 MM=MM-128:Z=1024:FORN=1TO16:FORT=8TO29STEP3:Z1=PEEK(T+Z):IFZ1)111THENZ1=Z1-64
117 Z2=PEEK(T+Z+1):IFZ2)111THENZ2=Z2-64
118 DD=VAL(H$+CHR$(Z1)+CHR$(Z2)):POKEMM,DD:MM=MM+1:NEXT:Z=Z+32:NEXT
119 GOTO3
121 GOSUB7:PRINT"   (*) SAVING OBJECT CODE (*)   ";A$;
122 W=1:GOSUB10:IFQQ=1THENQQ=0:GOTO3
123 PRINTA$;"Load tape and set to record.    Press S to Save, M for the Menu.";
124 GOSUB25:IFX$="S"ORX$="s"THEN125ELSEIFX$="M"ORX$="m"THEN3ELSE124
125 CLS0:PRINT@256,"";:INPUT"Enter Filename ----)";F$
126 CSAVEM F$,MM+6,NN+6,MM+6:GOTO3
129 GOSUB7:PRINT"    (*) BURNING AN EPROM (*)    ";A$;
130 PRINT"Do you wish to verify erasure?  Touch Y for yes, N for no."
131 GOSUB25:IFX$="Y"ORX$="y"THEN132ELSEIFX$="N"ORX$="n"THEN135ELSE131
132 GOSUB7:GOSUB10:IFQQ=1THENQQ=0:GOTO3:ELSECLS:EXEC XE
133 PRINT@256,"Press C to Continue, M for Menu"
134 GOSUB25:IFX$="C"ORX$="c"THEN135ELSEIFX$="M"ORX$="m"THEN3ELSE134
135 CLS:GOSUB7:GOSUB10:POKEV+3,MS+64:POKEV+4,MT:IFQQ=1THENQQ=0:GOTO3
136 IFE=5THENPOKE&H3D88,1:POKE&H3D89,0:CLS:FORQX=1TO25:EXECXP:NEXT:POKE&H3D88,24:POKE&H3D89,248:GOTO138:EL
SE137
137 CLS:EXECXP
138 PRINT@256,"PROGRAMMING IS COMPLETE.":PRINT"Do you wish burn verification?  (Touch Y for yes, N for no)
";
139 GOSUB25:IFX$="Y"ORX$="y"THEN140ELSEIFX$="N"ORX$="n"THEN3ELSE139
140 CLS:EXEC XV
141 PRINT@256,"Press "CHR$(34)"M"CHR$(34)" to return the Menu."
142 GOSUB25:IFX$="M"ORX$="m"THEN3ELSE142
144 GOSUB7:PRINT"  (*) DEFINING A NEW EPROM (*)  ";A$;
145 INPUT"How many pins (24 or 28)";D1:IFD1()24ANDD1()28THEN145
146 INPUT"Power (Vcc) what pin";D3:IFD3)D1 THEN145
147 INPUT"Program (Vpp) what pin";D4:IFD4)D1 THEN 145
148 INPUT"Lo OE what pin (0 if no OE)";D5:IFD5)D1 THEN145
149 INPUT"Lo CE what pin (0 if no CE)";D6:IFD6)D1 THEN145
150 INPUT"A11 what pin (0 if no A11)";D7:IFD7)D1 THEN145
151 INPUT"A12 what pin (0 if no A12)";D8:IFD8)D1 THEN145
152 INPUT"A13 what pin (0 if no A13)";D9:IFD9)D1 THEN145
153 INPUT"Lo PGM what pin (0 if no PGM)";DA:IFDA)D1 THEN145
154 PCLS:SCREEN1,1:LINE(80,10)-(180,180),PSET,B:LINE(120,10)-(140,10),PRESET
155 CIRCLE(130,10),10,3,1,0,.5
156 FORN=1TO8:CIRCLE(90,5+N*20),5,3:CIRCLE(170,5+N*20),5,3:NEXT
157 IFD1=28 THEN166
158 IFD3=0THEN159ELSED3=D3+2
159 IFD4=0THEN160ELSED4=D4+2
160 IFD5=0THEN161ELSED5=D5+2
161 IFD6=0THEN162ELSED6=D6+2
162 IFD7=0THEN163ELSED7=D7+2
163 IFD8=0THEN164ELSED8=D8+2
164 IFD9=0THEN165ELSED9=D9+2
165 IFDA=0THEN166ELSEDA=DA+2
166 IFD3=0THEN167ELSELINE(90,25)-(170,D(D3)),PSET
167 IFD4=0THEN168ELSELINE(90,45)-(170,D(D4)),PSET
168 IFD5=0THEN169ELSELINE(90,65)-(170,D(D5)),PSET
169 IFD6=0THEN170ELSELINE(90,85)-(170,D(D6)),PSET
170 IFD7=0THEN171ELSELINE(90,105)-(170,D(D7)),PSET
171 IFD8=0THEN172ELSELINE(90,125)-(170,D(D8)),PSET
172 IFD9=0THEN173ELSELINE(90,145)-(170,D(D9)),PSET
173 IFDA=0THEN174ELSELINE(90,165)-(170,D(DA)),PSET
174 GOSUB25:GOTO3
177 GOSUB7:PRINT"    (*) READING AN EPROM (*)    ";A$;
178 GOSUB10:IFQQ=1THENQQ=0:GOTO3
180 CLS:EXEC XR:PRINT@256,"EPROM READING IS COMPLETE.":PRINT"Press "CHR$(34)"M"CHR$(34)" to return the Men
u."
181 GOSUB25:IFX$="M"ORX$="m"THEN3ELSE181
```

```
                        00100 ************************
                        00110 * COLOR BURNER DRIVER 1.0
                        00120 * BY DENNIS BATHORY KITSZ
                        00130 * COPYRIGHT (C) 1984 BY
                        00140 * DENNIS BATHORY KITSZ
                        00150 ************************
                        00160 *
                        00170 *
3D00                    00180          ORG     $3D00
                        00190 *
                        00200 * OFFSET (0 4 8 12 16 20
                        00210 * 24 28) PLACED BY BASIC
                        00220 * TO INDICATE EPROM TYPE
                        00230 *
3D00    00              00240 OFFSET   FCB     $00
3D01    0000            00250 STTADR   FDB     $0000
3D03    4000            00260 STTDAT   FDB     $4000
3D05    0800            00270 ENDADR   FDB     $0800
                        00280 *
                        00290 * THE TABLES BELOW HOLD
                        00300 * THE PROGRAMMING DETAILS
                        00310 * FOR EIGHT EPROMS. FIXED
                        00320 * TABLES ARE PROVIDED FOR
                        00330 * 2716, 2732, 2764, 27128
                        00340 * AND 68764 EPROMS. OTHER
                        00350 * TYPES ARE FILLED IN BY
                        00360 * THE BASIC DRIVER.
                        00370 *
        3D07            00380 TAB16    EQU     *
3D07    03              00390 D2716    FCB     $03
3D08    C0              00400 R2716    FCB     $C0
3D09    91              00410 S2716    FCB     $91
3D0A    93              00420 P2716    FCB     $93
                        00430 *
        3D0B            00440 TAB32    EQU     *
3D0B    02              00450 D2732    FCB     $02
3D0C    C0              00460 R2732    FCB     $00
3D0D    A2              00470 S2732    FCB     $A2
3D0E    A0              00480 P2732    FCB     $A0
                        00490 *
        3D0F            00500 TAB64    EQU     *
3D0F    03              00510 D2764    FCB     $03
3D10    C4              00520 R2764    FCB     $C4
3D11    A5              00530 S2764    FCB     $A5
3D12    A1              00540 P2764    FCB     $A1
                        00550 *
        3D13            00560 TAB128   EQU     *
3D13    03              00570 D27128   FCB     $03
3D14    C4              00580 R27128   FCB     $C4
3D15    A5              00590 S27128   FCB     $A5
3D16    A1              00600 P27128   FCB     $A1
                        00610 *
        3D17            00620 TAB764   EQU     *
3D17    C0              00630 DSP1     FCB     $C0
3D18    00              00640 RSP1     FCB     $00
3D19    C0              00650 SSP1     FCB     $C0
3D1A    90              00660 PSP1     FCB     $90
                        00670 *
        3D1B            00680 TABSP2   EQU     *
3D1B    00              00690 DSP2     FCB     $00
3D1C    00              00700 RSP2     FCB     $00
3D1D    00              00710 SSP2     FCB     $00
3D1E    00              00720 PSP2     FCB     $00
                        00730 *
        3D1F            00740 TABSP3   EQU     *
3D1F    00              00750 DSP3     FCB     $00
3D20    00              00760 RSP3     FCB     $00
3D21    00              00770 SSP3     FCB     $00
3D22    00              00780 PSP3     FCB     $00
                        00790 *
        3D23            00800 TABSP4   EQU     *
3D23    00              00810 DSP4     FCB     $00
3D24    00              00820 RSP4     FCB     $00
3D25    00              00830 SSP4     FCB     $00
3D26    00              00840 PSP4     FCB     $00
                        00850 *
                        00860 * THIS ROUTINE CONFIGURES
                        00870 * PORT A AS AN INPUT PORT
                        00880 *
3D27 8E FF40            00890 CONAIN   LDX     #$FF40
3D2A C6 04              00900          LDB     #$04
3D2C 4F                 00910          CLRA
3D2D A7 01              00920          STA     1,X
3D2F A7 84              00930          STA     ,X
3D31 E7 01              00940          STB     1,X
3D33 39                 00950          RTS
                        00960 *
                        00970 * THIS ROUTINE CONFIGURES
                        00980 * THE A PORT AS OUTPUT
                        00990 *
3D34 8E FF40            01000 CONAOT   LDX     #$FF40
3D37 C6 04              01010          LDB     #$04
3D39 4F                 01020          CLRA
3D3A A7 01              01030          STA     1,X
3D3C 4A                 01040          DECA
3D3D A7 84              01050          STA     ,X
3D3F E7 01              01060          STB     1,X
3D41 39                 01070          RTS
                        01080 *
                        01090 * THIS ROUTINE CONFIGURES
                        01100 * PORTS B, C AND D AS
                        01110 * OUTPUTS.  B, C AND D
                        01120 * ARE USED TO PROVIDE A
                        01130 * BYTE OF DATA AND AN
                        01140 * ADDRESS TO THE EPROM
                        01150 *
3D42 C6 04              01160 CONBCD   LDB     #$04
3D44 8E FF40            01170          LDX     #$FF40
3D47 4F                 01180          CLRA
3D48 A7 03              01190          STA     3,X
3D4A 4A                 01200          DECA
3D4B A7 02              01210          STA     2,X
3D4D E7 03              01220          STB     3,X
3D4F 4F                 01230          CLRA
3D50 A7 05              01240          STA     5,X
3D52 4A                 01250          DECA
3D53 A7 04              01260          STA     4,X
3D55 E7 05              01270          STB     5,X
3D57 4F                 01280          CLRA
3D58 A7 07              01290          STA     7,X
3D5A 4A                 01300          DECA
3D5B A7 06              01310          STA     6,X
3D5D E7 07              01320          STB     7,X
3D5F 39                 01330          RTS
                        01340 *
                        01350 * THIS ROUTINE USES THE
                        01360 * OFFSET POKED BY BASIC
                        01370 * TO DETERMINE THE EPROM
                        01380 * TYPE AND SELECT THE
                        01390 * CORRECT TABLE FOR USE
                        01400 * BY PROGRAMMING SOFTWARE
                        01410 *
3D60 34 10              01420 ETYPE    PSHS    X
3D62 8E 3D07            01430          LDX     #TAB16
3D65 B6 3D00            01440          LDA     OFFSET
3D68 31 86              01450          LEAY    A,X
3D6A 35 10              01460          PULS    X
3D6C 39                 01470          RTS
                        01480 *
                        01490 * THIS ROUTINE DISPLAYS
                        01500 * MESSAGES DURING READ,
                        01510 * PROGRAM, VERIFY AND
                        01520 * ERASE CHECK.
                        01530 *
3D6D 108E 0460          01540 MESSER   LDY     #$0460
3D71 C6 10              01550          LDB     #$10
3D73 A6 80              01560 MLOOP1   LDA     ,X+
3D75 A7 A0              01570          STA     ,Y+
3D77 5A                 01580          DECB
3D78 26 F9              01590          BNE     MLOOP1
3D7A 39                 01600          RTS
                        01610 *
```

```
                    01620 * THIS ROUTINE PROVIDES
                    01630 * A SHORT DELAY TO ASSURE
                    01640 * THAT CONTROL SIGNALS
                    01650 * ARE PROPERLY RECEIVED
                    01660 * BY BOTH PIA AND EPROM
                    01670 *
3D7B 34   04        01680 DELAY1  PSHS    B
3D7D C6   04        01690         LDB     #$04
3D7F 5A             01700 LOOP1   DECB
3D80 26   FD        01710         BNE     LOOP1
3D82 35   04        01720         PULS    B
3D84 39             01730         RTS
                    01740 *
                    01750 * THIS ROUTINE CREATES A
                    01760 * 50 MS DELAY REQUIRED
                    01770 * FOR THE PROGRAMMING
                    01780 * PROCESS.  THE ACTUAL
                    01790 * DELAY CREATED HERE IS
                    01800 * APPROXIMATELY 50.1 MS
                    01810 *
3D85 34   06        01820 D50MS   PSHS    D
3D87 CC   18F8      01830         LDD     #$18F8
3D8A 83   0001      01840 LOOP50  SUBD    #$0001
3D8D 26   FB        01850         BNE     LOOP50
3D8F 35   06        01860         PULS    D
3D91 39             01870         RTS
                    01880 *
                    01890 * THIS ROUTINE DISPLAYS
                    01900 * A 16-BIT ADDRESS IN HEX
                    01910 * BY CALLING THE BYTE
                    01920 * DISPLAY ROUTINE TWICE
                    01930 *
3D92 34   30        01940 DISADR  PSHS    X,Y
3D94 108E 0470      01950         LDY     #$0470
3D98 8D   07        01960         BSR     SHOWAD
3D9A 1F   98        01970         TFR     B,A
3D9C 8D   03        01980         BSR     SHOWAD
3D9E 35   30        01990         PULS    X,Y
3DA0 39             02000         RTS
                    02010 *
                    02020 * THIS ROUTINE CONVERTS
                    02030 * ONE BYTE FROM BINARY
                    02040 * TO TWO HEXADECIMAL
                    02050 * DIGITS AND DISPLAYS
                    02060 * THEM ON THE SCREEN.
                    02070 *
3DA1 34   02        02080 SHOWAD  PSHS    A
3DA3 44             02090         LSRA
3DA4 44             02100         LSRA
3DA5 44             02110         LSRA
3DA6 44             02120         LSRA
3DA7 8D   0B        02130         BSR     CONVRT
3DA9 A7   A0        02140         STA     ,Y+
3DAB 35   02        02150         PULS    A
3DAD 84   0F        02160         ANDA    #$0F
3DAF 8D   03        02170         BSR     CONVRT
3DB1 A7   A0        02180         STA     ,Y+
3DB3 39             02190         RTS
3DB4 81   0A        02200 CONVRT  CMPA    #$0A
3DB6 24   03        02210         BCC     LETTER
3DB8 8B   70        02220         ADDA    #$70
3DBA 39             02230         RTS
3DBB 8B   37        02240 LETTER  ADDA    #$37
3DBD 39             02250         RTS
                    02260 *
                    02270 * THE FOLLOWING ROUTINE
                    02280 * DETERMINES THE EPROM
                    02290 * TYPE AND PREPARES IT TO
                    02300 * BE READ.
                    02310 *
3DBE 34   20        02320 SETRD   PSHS    Y
3DC0 8D   9E        02330         BSR     ETYPE
3DC2 A6   A4        02340         LDA     ,Y
3DC4 A7   06        02350         STA     6,X
3DC6 A6   21        02360         LDA     1,Y
3DC8 A7   06        02370         STA     6,X
3DCA 35   20        02380         PULS    Y
3DCC 39             02390         RTS
                    02400 *
                    02410 * THE EPROM PROGRAMMING
                    02420 * ROUTINE BEGINS HERE.
                    02430 * IT CONSISTS OF PORT
                    02440 * CONFIGURATION, SETTING
                    02450 * UP ADDRESS AND DATA TO
                    02460 * THE EPROM, AND ISSUING
                    02470 * PROGRAMMING PULSES.
                    02480 *
3DCD 8E   3F2B      02490 PROGMX  LDX     #MESSG1
3DD0 17   FF9A      02500         LBSR    MESSER
                    02510 *
                    02520 * THIS ROUTINE CONFIGURES
                    02530 * PORT A FOR OUTPUT TO
                    02540 * EPROM, PORTS B C D FOR
                    02550 * OUTPUT TO EPROM.
                    02560 *
3DD3 17   FF5E      02570 CONFIG  LBSR    CONAOT
3DD6 17   FF69      02580         LBSR    CONBCD
                    02590 *
                    02600 * D AND Y REGISTERS ARE
                    02610 * POINTED TO STARTING
                    02620 * ADDRESS OF THE EPROM
                    02630 * AND STARTING DATA IN
                    02640 * MEMORY TO PROGRAM.
                    02650 *
3DD9 FC   3D01      02660         LDD     STTADR
3DDC 10BE 3D03      02670         LDY     STTDAT
                    02680 *
                    02690 * THIS ROUTINE LATCHES
                    02700 * DATA INTO THE LOWER
                    02710 * AND UPPER HALVES OF THE
                    02720 * EPROM'S ADDRESS, GETS
                    02730 * THE DATA FROM MEMORY,
                    02740 * AND LATCHES THAT TO THE
                    02750 * EPROM THROUGH THE PIA.
                    02760 *
3DE0 E7   02        02770 AGAIN   STB     2,X
3DE2 A7   04        02780         STA     4,X
3DE4 34   06        02790         PSHS    D
3DE6 A6   A0        02800         LDA     ,Y+
3DE8 A7   84        02810         STA     ,X
                    02820 *
                    02830 * THE ADDRESS AND DATA
                    02840 * ARE DISPLAYED FOR THE
                    02850 * USER AS THE PROGRAMMING
                    02860 * GOES ON.
                    02870 *
3DEA 34   20        02880         PSHS    Y       *
3DEC 108E 0478      02890         LDY     #$0478
3DF0 17   FFAE      02900         LBSR    SHOWAD
3DF3 35   20        02910         PULS    Y
3DF5 35   06        02920         PULS    D
3DF7 34   06        02930         PSHS    D
3DF9 17   FF96      02940         LBSR    DISADR
3DFC 35   06        02950         PULS    D
3DFE 34   06        02960         PSHS    D
                    02970 *
                    02980 * THE ACTUAL PROGRAMMING
                    02990 * TAKES PLACE IN THIS
                    03000 * ROUTINE.  INTERRUPTS
                    03010 * ARE DISABLED TO KEEP
                    03020 * THE PROGRAMMING PULSE
                    03030 * OF THE CORRECT LENGTH,
                    03040 * THE EPROM TYPE IS
                    03050 * ESTABLISHED, AND A
                    03060 * SEQUENCE OF OFF-READ-
                    03070 * GET SET-PULSE-GET SET-
                    03080 * READ-OFF IS FOLLOWED.
                    03090 *
3E00 1A   50        03100 PROGRM  ORCC    #$50
3E02 34   20        03110         PSHS    Y
3E04 17   FF59      03120         LBSR    ETYPE
3E07 A6   A4        03130         LDA     ,Y
```

```
3E09 A7   06      03140          STA     6,X
3E0B 17   FF6D    03150          LBSR    DELAY1
3E0E A6   21      03160          LDA     1,Y
3E10 A7   06      03170          STA     6,X
3E12 17   FF66    03180          LBSR    DELAY1
3E15 A6   22      03190          LDA     2,Y
3E17 A7   06      03200          STA     6,X
3E19 17   FF5F    03210          LBSR    DELAY1
3E1C A6   23      03220          LDA     3,Y
3E1E A7   06      03230          STA     6,X
3E20 17   FF62    03240          LBSR    D50MS
3E23 A6   22      03250          LDA     2,Y
3E25 A7   06      03260          STA     6,X
3E27 17   FF51    03270          LBSR    DELAY1
3E2A A6   21      03280          LDA     1,Y
3E2C A7   06      03290          STA     6,X
3E2E 17   FF4A    03300          LBSR    DELAY1
3E31 A6   A4      03310          LDA     ,Y
3E33 A7   06      03320          STA     6,X
                  03330 *
                  03340 * ONE BYTE HAS BEEN
                  03350 * BURNED; A CHECK IS MADE
                  03360 * TO SEE IF THE PROCESS
                  03370 * IS COMPLETE.  IF NOT,
                  03380 * THE NEXT BYTE IS TAKEN
                  03390 * FROM MEMORY FOR BURNING
                  03400 * INTO THE EPROM.
                  03410 *
3E35 35   20      03420          PULS    Y
3E37 35   06      03430          PULS    D
3E39 C3   0001    03440          ADDD    #$0001
3E3C 10B3 3D05    03450          CMPD    ENDADR
3E40 26   9E      03460          BNE     AGAIN
                  03470 *
                  03480 * IF THE BURNING IS DONE,
                  03490 * A MESSAGE IS DISPLAYED
                  03500 * AND CONTROL IS RETURNED
                  03510 * TO BASIC.
                  03520 *
3E42 8E   3F8B    03530          LDX     #MESSG7
3E45 C6   0D      03540          LDB     #13
3E47 17   FF23    03550          LBSR    MESSER
3E4A 39           03560          RTS
                  03570 *
                  03580 * THE CHECK FOR AN ERASED
                  03590 * EPROM BEGINS HERE.  AN
                  03600 * ERASED EPROM CONTAINS
                  03610 * ALL FF (255) AS SENT
                  03620 * FROM THE FACTORY, OR
                  03630 * AFTER ERASURE UNDER
                  03640 * ULTRAVIOLET LIGHT. THIS
                  03650 * ROUTINE READS THE VALUE
                  03660 * IN THE EPROM.
                  03670 *
3E4B 8E   3F4B    03680 ERASEX  LDX     #MESSG3
3E4E 17   FF1C    03690          LBSR    MESSER
                  03700 *
                  03710 * SETUP WITH PORT A AS
                  03720 * INPUT AND PORTS B, C
                  03730 * AND D AS OUTPUT FOLLOWS
                  03740 *
3E51 17   FED3    03750          LBSR    CONAIN
3E54 17   FEEB    03760          LBSR    CONBCD
                  03770 *
                  03780 * D IS POINTED TO THE
                  03790 * STARTING ADDRESS IN THE
                  03800 * EPROM
                  03810 *
3E57 FC   3D01    03820          LDD     STTADR
                  03830 *
                  03840 * THE LSB AND MSB OF THE
                  03850 * ADDRESS ARE LATCHED
                  03860 * INTO THE EPROM, AND
                  03870 * THE ADDRESS AND DATA
                  03880 * ARE DISPLAYED FOR THE
                  03890 * USER.
                  03900 *
3E5A E7   02      03910 AGIN1   STB     2,X
3E5C A7   04      03920          STA     4,X
3E5E 34   06      03930          PSHS    D
3E60 17   FF2F    03940          LBSR    DISADR
3E63 17   FF58    03950          LBSR    SETRD
3E66 A6   84      03960          LDA     ,X
3E68 34   22      03970          PSHS    Y,A
3E6A 108E 0478    03980          LDY     #$0478
3E6E 17   FF30    03990          LBSR    SHOWAD
3E71 35   22      04000          PULS    A,Y
3E73 81   FF      04010          CMPA    #$FF
3E75 35   06      04020          PULS    D
3E77 26   0E      04030          BNE     EFAULT
3E79 C3   0001    04040          ADDD    #$0001
3E7C 10B3 3D05    04050          CMPD    ENDADR
3E80 26   D8      04060          BNE     AGIN1
                  04070 *
                  04080 * A CORRECT ERASURE OR
                  04090 * A FAILED ERASURE IS
                  04100 * DISPLAYED FOR THE USER.
                  04110 *
3E82 8E   3F9B    04120          LDX     #MESSG8
3E85 20   56      04130          BRA     OUT1
                  04140 *
3E87 8E   3F6B    04150 EFAULT  LDX     #MESSG5
3E8A 20   51      04160          BRA     OUT1
                  04170 *
                  04180 * THE VERIFY ROUTINE
                  04190 * (COMPARISON OF EPROM
                  04200 * WITH MEMORY CONTENTS)
                  04210 * BEGINS HERE.  THE SETUP
                  04220 * OF PORT A AS INPUT AND
                  04230 * PORTS B, C AND D AS
                  04240 * OUTPUT, PLUS EPROM
                  04250 * TYPING, IS DONE HERE.
                  04260 *
3E8C 8E   3F3B    04270 VERIFX  LDX     #MESSG2
3E8F 17   FEDB    04280          LBSR    MESSER
                  04290 *
3E92 17   FE92    04300          LBSR    CONAIN
3E95 17   FEAA    04310          LBSR    CONBCD
                  04320 *
                  04330 * D POINTS TO THE START
                  04340 * ADDRESS IN THE EPROM,
                  04350 * Y POINTS TO THE DATA
                  04360 * STORED IN MEMORY.
                  04370 *
3E98 FC   3D01    04380          LDD     STTADR
3E9B 10BE 3D03    04390          LDY     STTDAT
                  04400 *
                  04410 * A AND B ARE LATCHED AS
                  04420 * THE LSB AND MSB OF THE
                  04430 * EPROM'S ADDRESS, AND
                  04440 * ALL THE INFORMATION IN
                  04450 * PROGRESS IS DISPLAYED
                  04460 * FOR THE USER.
                  04470 *
3E9F E7   02      04480 AGIN2   STB     2,X
3EA1 A7   04      04490          STA     4,X
3EA3 34   06      04500          PSHS    D
3EA5 17   FEEA    04510          LBSR    DISADR
3EA8 17   FF13    04520          LBSR    SETRD
3EAB A6   84      04530          LDA     ,X
3EAD 34   20      04540          PSHS    Y
3EAF 108E 0478    04550          LDY     #$0478
3EB3 17   FEEB    04560          LBSR    SHOWAD
3EB6 86   7A      04570          LDA     #$7A
3EB8 A7   A0      04580          STA     ,Y+
3EBA 35   20      04590          PULS    Y
3EBC A6   A4      04600          LDA     ,Y
3EBE 34   20      04610          PSHS    Y
3EC0 108E 047C    04620          LDY     #$047C
3EC4 17   FEDA    04630          LBSR    SHOWAD
3EC7 35   20      04640          PULS    Y
3EC9 A6   84      04650          LDA     ,X
```

```
3ECB A1   A0      04660          CMPA    ,Y+
3ECD 35   06      04670          PULS    D
3ECF 26   10      04680          BNE     VFAULT
3ED1 C3   0001    04690 BACK     ADDD    #$0001
3ED4 10B3 3D05    04700          CMPD    ENDADR
3ED8 26   C5      04710          BNE     AGIN2
                  04720 *
                  04730 * EITHER A VERIFICATION
                  04740 * COMPLETE OR A VERIFY
                  04750 * FAILURE IS REPORTED IN
                  04760 * THE MESSAGES THAT
                  04770 * FOLLOW.
                  04780 *
3EDA 8E   3FAB    04790          LDX     #MESSG9
3EDD 17   FE8D    04800 OUT1     LBSR    MESSER
3EE0 39           04810          RTS
                  04820 *
3EE1 34   36      04830 VFAULT   PSHS    X,Y,D
3EE3 8E   3F5B    04840          LDX     #MESSG4
3EE6 17   FE84    04850          LBSR    MESSER
3EE9 35   36      04860          PULS    X,Y,D
3EEB 39   12      04870          RTS + NOP   (rev 2.22)
                  04880 *
                  04890 * THE ROUTINE TO READ THE
                  04900 * CONTENTS OF AN EPROM
                  04910 * INTO MEMORY BEGINS HERE
                  04920 *
3EED 8E   3FBB    04930 READRX   LDX     #MESSGA
3EF0 17   FE7A    04940          LBSR    MESSER
                  04950 *
                  04960 * THE USUAL SETUP OF PORT
                  04970 * A AS INPUT, PORTS B, C
                  04980 * AND D AS OUTPUT FOLLOWS
                  04990 *
3EF3 17   FE31    05000          LBSR    CONAIN
3EF6 17   FE49    05010          LBSR    CONBCD
                  05020 *
                  05030 * D POINTS TO THE EPROM
                  05040 * START ADDRESS, Y POINTS
                  05050 * TO THE STARTING PLACE
                  05060 * FOR DATA STORAGE IN
                  05070 * MEMORY.
                  05080 *
3EF9 FC   3D01    05090          LDD     STTADR
3EFC 10BE 3D03    05100          LDY     STTDAT
                  05110 *
                  05120 * ADDRESSES ARE LATCHED
                  05130 * INTO THE EPROM, AND
                  05140 * THE INFORMATION IS
                  05150 * DISPLAYED FOR THE
                  05160 * USER.
                  05170 *
3F00 E7   02      05180 AGIN3    STB     2,X
3F02 A7   04      05190          STA     4,X
3F04 34   06      05200          PSHS    D
3F06 17   FE89    05210          LBSR    DISADR
3F09 17   FEB2    05220          LBSR    SETRD
3F0C A6   84      05230          LDA     ,X
3F0E 34   22      05240          PSHS    A,Y
3F10 108E 0478    05250          LDY     #$0478
3F14 17   FE8A    05260          LBSR    SHOWAD
3F17 35   22      05270          PULS    Y,A
3F19 A7   A0      05280          STA     ,Y+
3F1B 35   06      05290          PULS    D
3F1D C3   0001    05300          ADDD    #$0001
3F20 10B3 3D05    05310          CMPD    ENDADR
3F24 26   DA      05320          BNE     AGIN3
                  05330 *
3F26 8E   3FCB    05340          LDX     #MESSGB
3F29 20   B2      05350          BRA     OUT1
                  05360 *
                  05370 *
                  05380 * THE MESSAGES TO BE
                  05390 * DISPLAYED FOLLOW HERE.
                  05400 * NOT ALL THE MESSAGES
                  05410 * ARE USED BY VERSION 1.0
                  05420 * OF THE COLOR BURNER
                  05430 * SOFTWARE.
                  05440 *
3F2B 50           05450 MESSG1   FCC     /PROGRAMMING/
     52
     4F
     47
     52
     41
     4D
     4D
     49
     4E
     47
3F36 6060         05460          FDB     $6060
3F38 6060         05470          FDB     $6060
3F3A 60           05480          FCB     $60
3F3B 56           05490 MESSG2   FCC     /VERIFYING/
     45
     52
     49
     46
     59
     49
     4E
     47
3F44 6060         05500          FDB     $6060
3F46 6060         05510          FDB     $6060
3F48 6060         05520          FDB     $6060
3F4A 60           05530          FCB     $60
3F4B 45           05540 MESSG3   FCC     /ERASE/
     52
     41
     53
     45
3F50 60           05550          FCB     $60
3F51 43           05560          FCC     /CHECK/
     48
     45
     43
     4B
3F56 6060         05570          FDB     $6060
3F58 6060         05580          FDB     $6060
3F5A 60           05590          FCB     $60
3F5B 56           05600 MESSG4   FCC     /VERIFY/
     45
     52
     49
     46
     59
3F61 60           05610          FCB     $60
3F62 46           05620          FCC     /FAILS/
     41
     49
     4C
     53
3F67 60           05630          FCB     $60
3F68 41           05640          FCC     /AT/
     54
3F6A 60           05650          FCB     $60
3F6B 45           05660 MESSG5   FCC     /ERASE/
     52
     41
     53
     45
3F70 60           05670          FCB     $60
3F71 46           05680          FCC     /FAILS/
     41
     49
     4C
     53
3F76 60           05690          FCB     $60
3F77 41           05700          FCC     /AT/
     54
3F79 6060         05710          FDB     $6060
```

```
3F7B 42 05720 MESSG6 FCC /BURN/
55
52
4E
3F7F 60 05730 FCB $60
3F80 46 05740 FCC /FAILS/
41
49
4C
53
3F85 60 05750 FCB $60
3F86 41 05760 FCC /AT/
54
3F88 6060 05770 FDB $6060
3F8A 60 05780 FCB $60
3F8B 42 05790 MESSG7 FCC /BURN/
55
52
4E
3F8F 60 05800 FCB $60
3F90 43 05810 FCC /COMPLETE/
4F
4D
50
4C
45
54
45
3F98 6060 05820 FDB $6060
3F9A 60 05830 FCB $60
3F9B 45 05840 MESSG8 FCC /EPROM/
50
52
4F
4D
3FA0 60 05850 FCB $60
3FA1 45 05860 FCC /ERASED/
52
41
53
45
44
3FA7 6060 05870 FDB $6060
3FA9 6060 05880 FDB $6060
3FAB 56 05890 MESSG9 FCC /VERIFY/
45
52
49
46
59
3FB1 60 05900 FCB $60
3FB2 43 05910 FCC /COMPLETE/
4F
4D
50
4C
45
54
45
3FBA 60 05920 FCB $60
3FBB 52 05930 MESSGA FCC /READING/
45
41
44
49
4E
47
3FC2 60 05940 FCB $60
3FC3 45 05950 FCC /EPROM/
50
52
4F
4D
3FC8 6060 05960 FDB $6060
3FCA 60 05970 FCB $60
3FCB 45 05980 MESSGB FCC /EPROM/
50
52
4F
4D
3FD0 60 05990 FCB $60
3FD1 52 06000 FCC /READ/
45
41
44
3FD5 6060 06010 FDB $6060
3FD7 6060 06020 FDB $6060
3FD9 6060 06030 FDB $6060
3FDB 42 06040 MESSGC FCC /BAD/
41
44
3FDE 60 06050 FCB $60
3FDF 45 06060 FCC /EPROM/
50
52
4F
4D
3FE4 60 06070 FCB $60
3FE5 54 06080 FCC /TYPE/
59
50
45
3FE9 6060 06090 FDB $6060
3FEB 44 06100 MESSGD FCC /D.B.KITSZ/
2E
42
2E
4B
49
54
53
5A
3FF4 60 06110 FCB $60
3FF5 52 06120 FCC /ROXBURY/
4F
58
42
55
52
59
3FFC 60 06130 FCB $60
06140 *
3FFD 06150 ZZZZZZ EQU *
06160 *
0000 06170 END
```

```
00000 TOTAL ERRORS
AGAIN  3DE0  MESSG4  3F5B  P2764   3D12  SSP3    3D21
AGIN1  3E5A  MESSG5  3F6B  PROGMX  3DCD  SSP4    3D25
AGIN2  3E9F  MESSG6  3F7B  PROGRM  3E00  STTADR  3D01
AGIN3  3F00  MESSG7  3F8B  PSP1    3D1A  STTDAT  3D03
BACK   3ED1  MESSG8  3F9B  PSP2    3D1E  TAB128  3D13
CONAIN 3D27  MESSG9  3FAB  PSP3    3D22  TAB16   3D07
CONADT 3D34  MESSGA  3FBB  PSP4    3D26  TAB32   3D0B
CONBCD 3D42  MESSGB  3FCB  R27128  3D14  TAB64   3D0F
CONFIG 3DD3  MESSGC  3FDB  R2716   3D08  TAB764  3D17
CONVRT 3DB4  MESSGD  3FEB  R2732   3D0C  TABSP2  3D1B
D50MS  3D85  MLOOP1  3D73  R2764   3D10  TABSP3  3D1F
DELAY1 3D7B  O27128  3D13  READRX  3EED  TABSP4  3D23
DISADR 3D92  O2716   3D07  RSP1    3D18  VERIFX  3E8C
EFAULT 3E87  O2732   3D0B  RSP2    3D1C  VFAULT  3EE1
ENDADR 3D05  O2764   3D0F  RSP3    3D20  ZZZZZZ  3FFD
ERASEX 3E4B  OFFSET  3D00  RSP4    3D24
ETYPE  3D60  OSP1    3D17  S27128  3D15
LETTER 3DBB  OSP2    3D1B  S2716   3D09
LOOP1  3D7F  OSP3    3D1F  S2732   3D0D
LOOP50 3D8A  OSP4    3D23  S2764   3D11
MESSER 3D6D  OUT1    3EDD  SETRD   3DBE
MESSG1 3F2B  P27128  3D16  SHOWAD  3DA1
MESSG2 3F3B  P2716   3D0A  SSP1    3D19
MESSG3 3F4B  P2732   3D0E  SSP2    3D1D
```

MCM2716

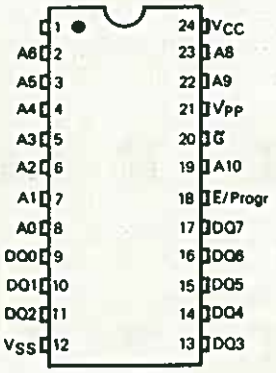
MOS

(N-CHANNEL, SILICON-GATE)

2048 × 8-BIT
UV ERASABLE PROM

- Single 5 V Power Supply
- Automatic Power-down Mode (Standby)
- Organized as 2048 Bytes of 8 Bits
- TTL Compatible During Read and Program
- Maximum Access Time = 450 ns MCM2716
- Pin Equivalent to Intel's 2716
- Pin Compatible to MCM68A316E
- Output Enable Active Level is User Selectable

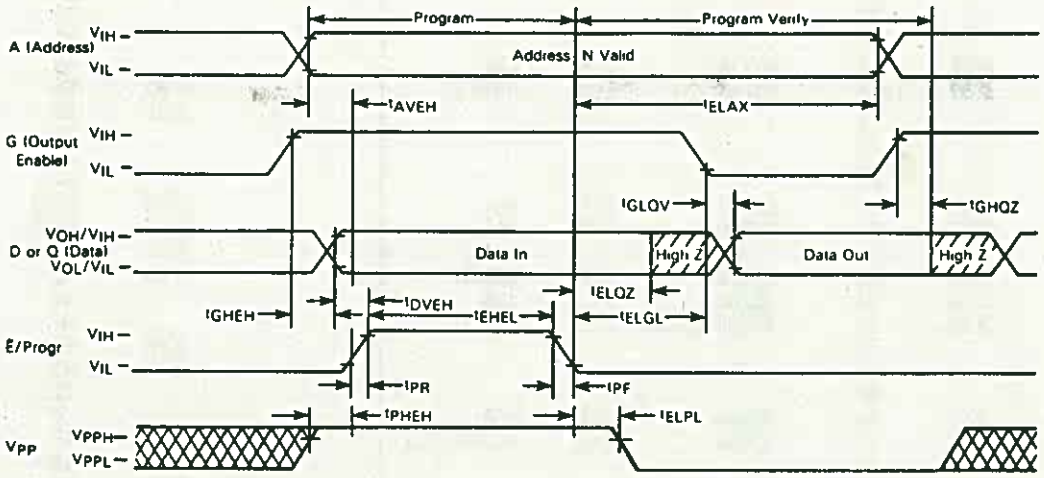
PIN ASSIGNMENT



*Pin Names	
A	Address
DQ	Data Input/Output
E/Progr	Chip Enable/Program
G	Output Enable

*New industry standard nomenclature

PROGRAMMING OPERATION TIMING DIAGRAM



PROGRAMMING INSTRUCTIONS

Before programming, the memory should be submitted to a full ERASE operation to ensure every bit in the device is in the "1" state (represented by Output High). Data are entered by programming zeros (Output Low) into the required bits. The words are addressed the same way as in the READ operation. A programmed "0" can only be changed to a "1" by ultraviolet light erasure.

To set the memory up for Program Mode, the V_{PP} input (Pin 21) should be raised to +25 V. The V_{CC} supply voltage is the same as for the Read operation and $\overline{G}$ is at V_{IH}. Programming data is entered in 8-bit words through the data out (DQ) terminals. Only "0 s" will be programmed when "0's" and "1 s" are entered in the 8-bit data word.

After address and data setup, a program pulse (V_{IL} to V_{IH}) is applied to the $\overline{E}$/Progr input. A program pulse is applied to each address location to be programmed. To minimize programming time, a 2 ms pulse width is recommended. The maximum program pulse width is 55 ms; therefore, programming must not be attempted with a dc signal applied to the $\overline{E}$/Progr input.

Multiple MCM2716s may be programmed in parallel by connecting together like inputs and applying the program pulse to the $\overline{E}$/Progr inputs. Different data may be programmed into multiple MCM2716s connected in parallel by using the PROGRAM INHIBIT mode. Except for the $\overline{E}$/Progr pin, all like inputs (including Output Enable) may be common.

The PROGRAM VERIFY mode with V_{PP} at 25 V is used to determine that all programmed bits were correctly programmed.

READ OPERATION

After access time, data is valid at the outputs in the READ mode. With stable system addresses, effectively faster access time can be obtained by gating the data onto the bus with Output Enable.

The Standby mode is available to reduce active power dissipation. The outputs are in the high impedance state when the $\overline{E}$/Progr input pin is high (V_{IH}) independent of the Output Enable input.

ERASING INSTRUCTIONS

The MCM2716 can be erased by exposure to high intensity shortwave ultraviolet light, with a wavelength of 2537 angstroms. The recommended integrated dose (i.e., UV-intensity X exposure time) is 15 Ws/cm^2. As an example, using the "Model 30-000" UV-Eraser (Turner Designs, Mountain View, CA 94043) the ERASE-time is 36 minutes. The lamps should be used without shortwave filters and the MCM2716 should be positioned about one inch away from the UV-tubes.

RECOMMENDED OPERATING PROCEDURES

After erasure and reprogramming of the EPROM, it is recommended that the quartz window be covered with an opaque self-adhesive cover. It is important that the self-adhesive cover not leave any residue on the quartz if it is removed to allow another erasure.

RECOMMENDED PROGRAMMING OPERATING CONDITIONS

Parameter	Symbol	Min	Nom	Max	Unit
Supply Voltage	V_{CC}, V_{PPL} V_{PPH}	4.75 24	5.0 25	5.25 26	V
Input High Voltage for Data	V_{IH}	2.2	–	V_{CC} + 1	V
Input Low Voltage for Data	V_{IL}	−0.1	–	0.8	V

PROGRAMMING OPERATION DC CHARACTERISTICS

Characteristic	Condition	Symbol	Min	Typ	Max	Unit
Address, G and E/Progr Input Sink Current	V_{in} = 5.25 V/0.45V	I_{LI}	–	–	10	μA
V_{PP} Programming Pulse Supply Current (V_{PP} = 25 V ± 1 V)	$\overline{E}$/Progr = V_{IH}	I_{PP2}	–	–	30	mA
V_{CC} Supply Current (Outputs Open)	–	I_{CC}	–	–	160	mA

AC PROGRAMMING OPERATING CONDITIONS AND CHARACTERISTICS

Characteristic	Symbol	Min	Max	Unit
Address Setup Time	t_{AVEH}	2.0	–	μs
Output Enable High to Program Pulse	t_{GHEH}	2.0	–	μs
Data Setup Time	t_{DVEH}	2.0	–	μs
Address Hold Time	t_{ELAX}	2.0	–	μs
Output Enable Hold Time	t_{ELGL}	2.0	–	μs
Data Hold Time	t_{ELQZ}	2.0	–	μs
V_{PP} Setup Time	t_{PHEH}	0	–	ns
V_{PP} to Enable Low Time	t_{ELPL}	0	–	ns
Output Disable to High Z Output	t_{GHQZ}	0	150	ns
Output Enable to Valid Data ($\overline{E}$/Progr = V_{IL})	t_{GLQV}	–	150	ns
Program Pulse Width	t_{EHEL}	1*	55	ms
Program Pulse Rise Time	t_{PR}	5	–	ns
Program Pulse Fall Time	t_{PF}	5	–	ns

*If shorter than 45 ms (min) pulses are used, the same number of pulses should be applied after the specific data has been verified to ensure that good programming levels have been written.

PROGRAMMING INSTRUCTIONS

Before programming, the memory should be submitted to a full erase operation to ensure that every bit is in the "1" state (represented by Output High). Data is entered by programming zeros (Output Low) into the required bits. The words are addressed the same way as in the READ operation. A programmed "0" can only be changed to a "1" by ultraviolet erasure.

To set the memory up for Program Mode, the $\overline{E}/V_{PP}$ input (Pin 20) should be between +2.0 and +6.0 V, which will three-state the outputs and allow data to be setup on the DQ terminals. The V_{CC} voltage is the same as for the Read operation. Only "0's" will be programmed when "0's" and "1's" are entered in the 8-bit data word.

After address and data setup, 25-volt programming pulse (V_{IH} to V_{IHP}) is applied to the $\overline{E}/V_{PP}$ input. The program pulse width is 2 ms and the maximum program pulse amplitude is 26 V.

Multiple MCM68764s may be programmed in parallel by connecting like inputs and applying the program pulse to the $\overline{E}/V_{PP}$ inputs. Different data may be programmed into multiple MCM68764s connected in parallel by selectively applying the programming pulse only to the MCM68764s to be programmed.

READ OPERATION

After access time, data is valid at the outputs in the Read mode. A single input ($\overline{E}/V_{PP}$) enables the outputs and puts the chip in active or standby mode. With $\overline{E}/V_{PP}$ = "0" the outputs are enabled and the chip is in active mode; with $\overline{E}/V_{PP}$ = "1" the outputs are three-stated and the chip is in standby mode. During standby mode, the power dissipation is reduced.

Multiple MCM68764s may share a common data bus with like outputs OR-tied together. In this configuration, only one $\overline{E}/V_{PP}$ input should be low and no other device outputs should be active on the same bus. This will prevent data contention on the bus.

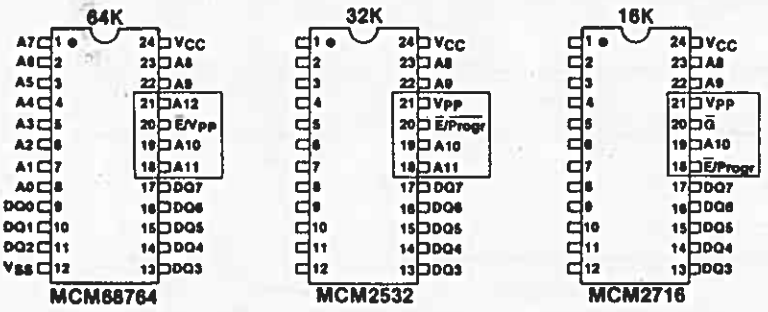
ERASING INSRUCTIONS

The MCM68764 can be erased by exposure to high intensity shortwave ultraviolet light, with a wavelength of 2537 angstroms. The recommended integrated dose (i.e., UV-intensity X exposure time) is 15 Ws/cm^2. As an example, using the "Model 30-000" UV-Eraser (Turner Designs, Mountain View, CA 94043) the ERASE-time is 36 minutes. The lamps should be used without shortwave filters and the MCM68764 should be positioned about one inch away from the UV-tubes.

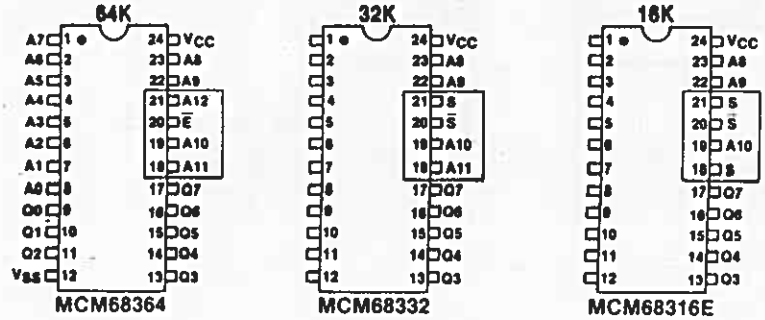
RECOMMENDED OPERATING PROCEDURES

After erasure and reprogramming of the EPROM, it is recommended that the quartz window be covered with an opaque self-adhesive cover. It is important that the self-adhesive cover not leave any residue on the quartz if it is removed to allow another erasure.

MOTOROLA'S PIN-COMPATIBLE EPROM FAMILY

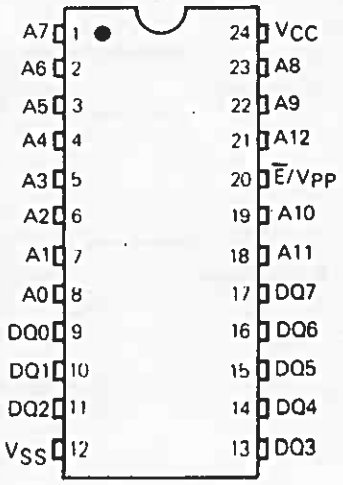


MOTOROLA'S PIN-COMPATIBLE ROM FAMILY



INDUSTRY STANDARD PINOUTS

PIN ASSIGNMENT



Pin Names	
A	Address
DQ	Data Input/Output
$\overline{E}/V_{PP}$ . . .	Chip Enable/Program

DC PROGRAMMING CONDITIONS AND CHARACTERISTICS

($T_A = 25 \pm 5°C$)

RECOMMENDED PROGRAMMING OPERATING CONDITIONS

Parameter	Symbol	Min	Nom	Max	Unit
Supply Voltage	V_{CC}	4.75	5.0	5.25	V
Input High Voltage for All Addresses and Data	V_{IH}	2.2	–	V_{CC} + 1	V
Input Low Voltage for All Addresses and Data	V_{IL}	−0.1	–	0.8	V
Program Pulse Input High Voltage	V_{IHP}	24	25	26	V
Program Pulse Input Low Voltage	V_{ILP}	2.0	V_{CC}	6.0	V

PROGRAMMING OPERATION DC CHARACTERISTICS

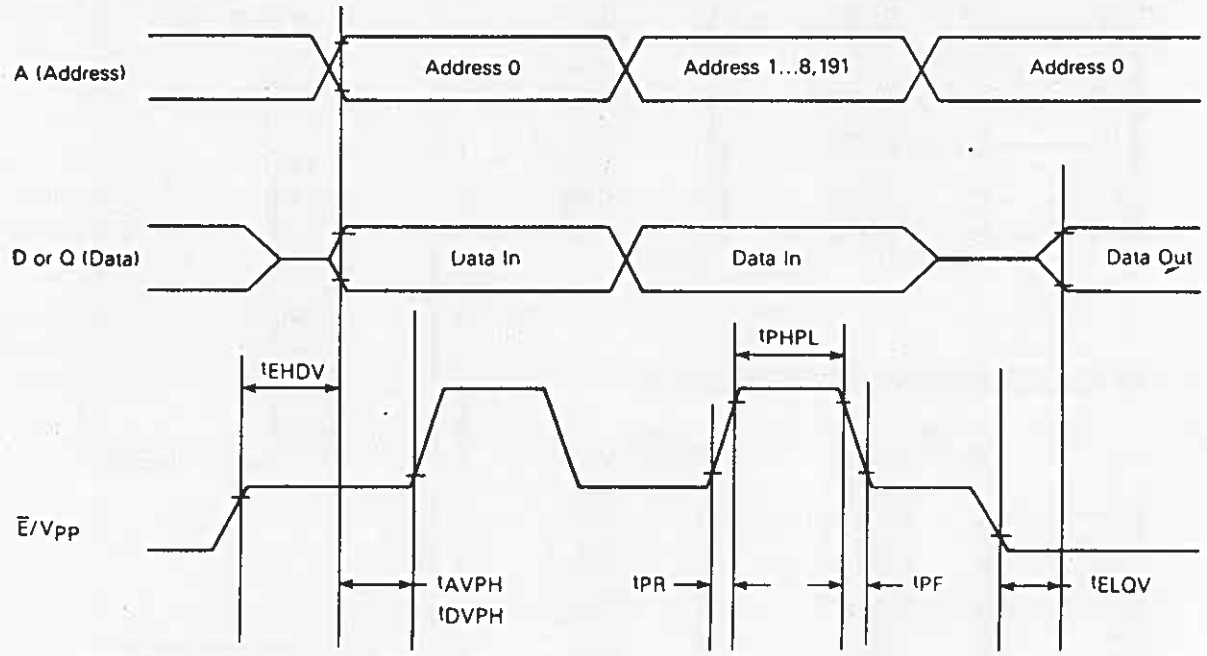
Characteristic	Condition	Symbol	Min	Typ	Max	Unit
Address Input Sink Current	V_{in} = 5.25 V	I_{LI}	–	–	10	μA
V_{PP} Program Pulse Supply Current (V_{PP} = 25 V ± 1 V)	–	I_{PH}	–	–	30	mA
V_{PP} Supply Current (V_{PP} = 2.4 V)	–	$I_{PL} = I_{EH}$	–	–	400	μA
V_{CC} Supply Current (V_{PP} = 5.0 V)	–	I_{CC}	–	–	160	mA

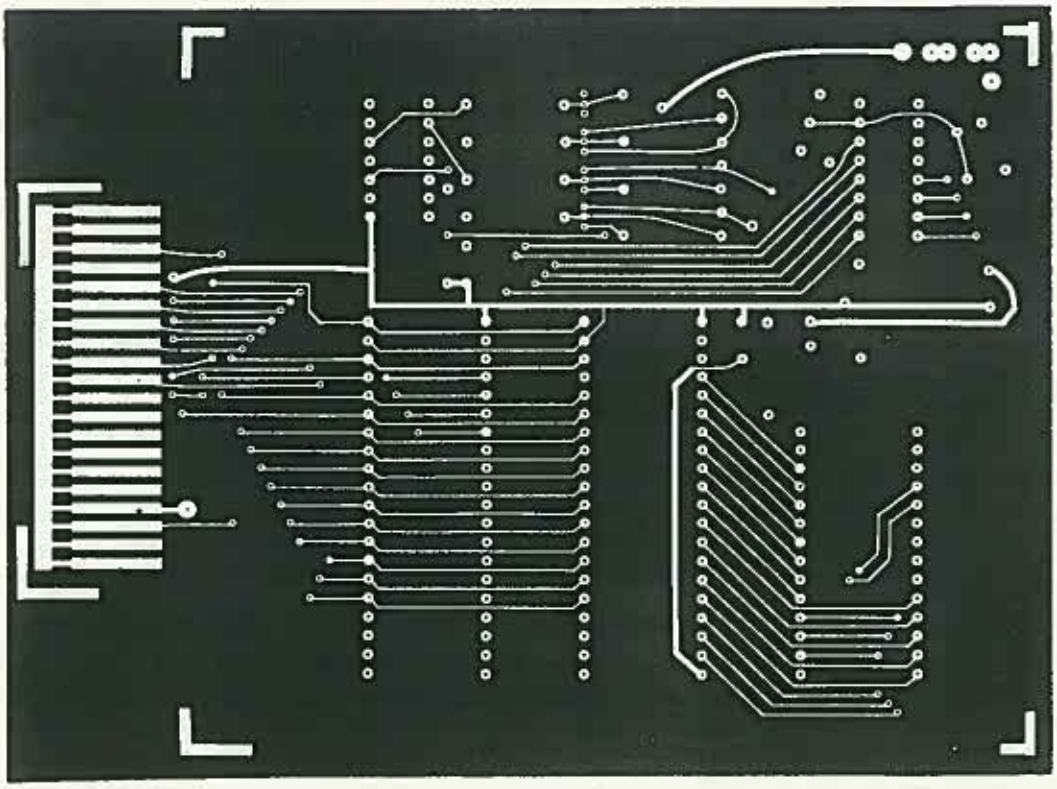
AC PROGRAMMING OPERATING CONDITIONS AND CHARACTERISTICS

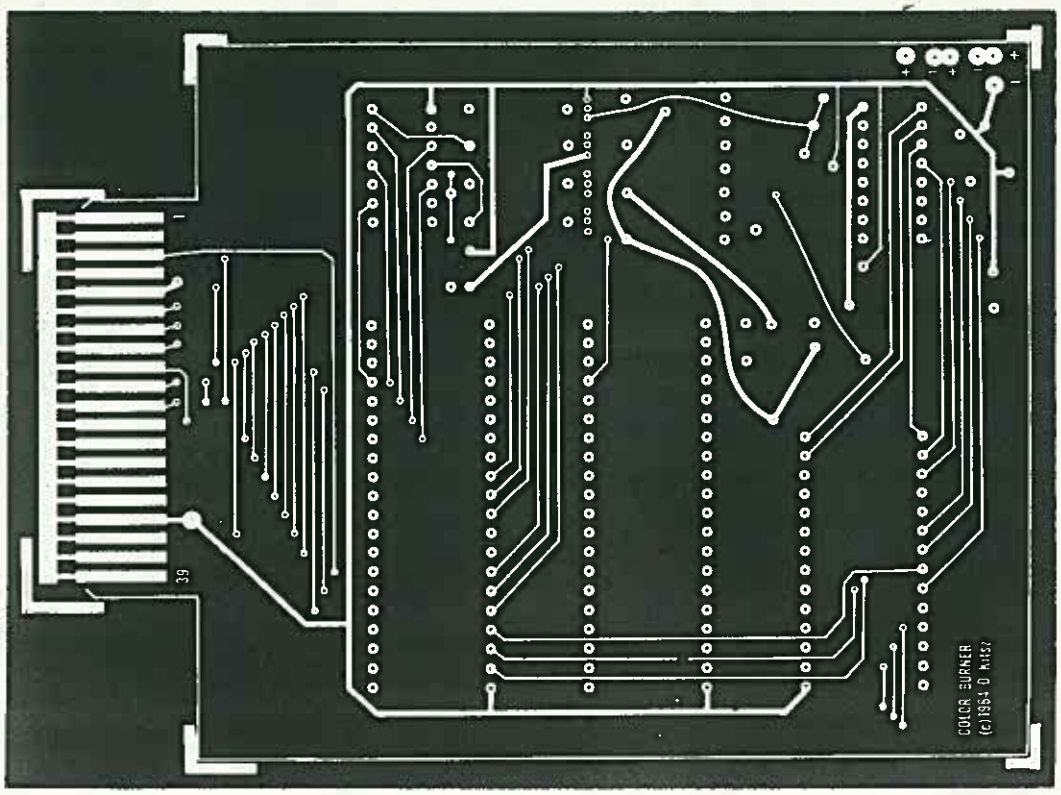
Characteristic	Symbol	Min	Max	Unit
Address Setup Time	t_{AVPH}	2.0	–	μs
Data Setup Time	t_{DVPH}	2.0	–	μs
Chip Enable to Valid Data	t_{ELQV}	450	–	ns
Chip Disable to Data In	t_{EHDV}	2.0	–	μs
Program Pulse Width	t_{PHPL}	1.9	2.1	ms
Program Pulse Rise Time	t_{PR}	0.5	2.0	μs
Program Pulse Fall Time	t_{PF}	0.5	2.0	μs
Cumulative Programming Time Per Word*	t_{CP}	12	50	ms

* Block mode programming must be used. Block mode programming is defined as one program pulse applied to each of the 8,192 address locations in sequence. Multiple blocks are used to accumulate programming time (t_{CP}). If less than 25 two millisecond pulses are required to verify programming, then 5 additional 2 millisecond pulses are required to ensure proper operating margins (i.e., 2 ms + 5 × 2 ms = 12 ms minimum t_{CP}).

PROGRAMMING OPERATION TIMING DIAGRAM






39
COLOR BURNER
(c)1984 D Hitz

USE YOUR COLOR BURNER TO CREATE AUTO-BOOT BASIC PROGRAMS

Starting a Basic program automatically from ROM is simple. Keep in mind that you can ROM about a 12K Basic program and 4K of special purpose machine language drivers with the simple routine presented here. By changing a few values, you can ROM nearly 16K of Basic.

First, you must have the Basic autostart program right at the beginning of the ROM you are going to use. Since it is short, use the Color Burner's machine language monitor (menu selection #3; enter 4000 in response to the prompt), and enter it as shown here:

```
4000 * 44 4B 86 55 97 71 CC 80
4008 * C0 DD 72 BD B9 5C 8E D0
4010 * 00 10 8E 0C 00 10 9F 19
4018 * 0C 1A EC 81 ED A1 10 83
4020 * 00 00 26 F6 ED A1 10 9F
4028 * 1B 8E 02 DD CC 52 55 ED
4030 * 81 CC 4E 00 ED 81 C6 04
4038 * 4F 97 70 8E 02 DC 7E AC
4040 * 7F 00 00 00 00 00 00 00
```

Burn this small program into several erased EPROMs. Use 2764 or larger EPROMs; the area from $4041 to about $40FF should be reserved for future enhancements of this routine. The area from $4100 to $4FFF is reserved for your own machine language programs (This routine is used in Green Mountain Micro's Data Gatherer, and all specialized machine language drivers are located in this area). Your Basic program will reside from $5000 to $5FFF (4K in a 2764) or from $5000 to $7EFF (about 12K in a 27128).

Keep these burned EPROMs handy, and follow the remaining instructions to burn and auto-boot your Basic programs.

NOTE: You can use your auto-boot Basic program in a ROMpack, or it can be wired internally if you are using your machine as a dedicated terminal, monitoring computer, assembly language development system or (as we do at Green Mountain Micro) as a Color Burner test station.

Internal wiring instructions are provided with this documentation.

PROCEDURE TO BURN A BASIC PROGRAM WITH THE COLOR BURNER

I. SAVING THE BASIC PROGRAM. Follow this procedure exactly!

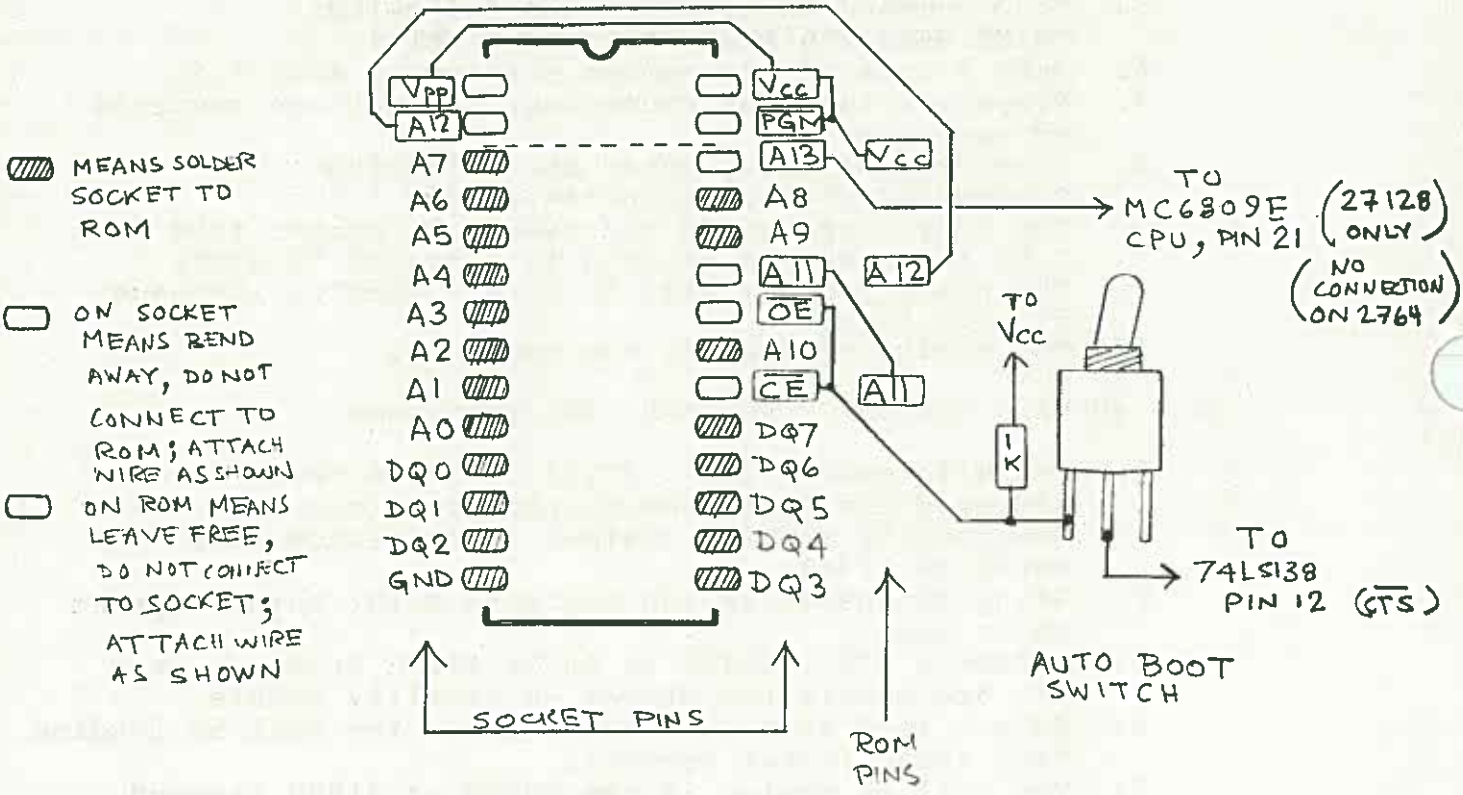
1. Turn the computer on. If it is on, turn it off, and and back on again. This is important.
2. Type and enter: PMODE0:PCLEAR1
3. CLOAD the program, list it to verify that it is correct. DO NOT RUN the program!
4. Important: even if you change PMODEs and PCLEARs later in your Basic program, we advise that you make the FIRST LINE of your program read: PMODE0:PCLEAR1. This will avoid the PCLEAR bug in Basic and set parameters correctly.
5. From command mode, enter the following: PRINT HEX$(256*PEEK(&H1B)+PEEK(&H1C))
6. Make a note of the number printed in step I.5.
7. Prepare a tape for recording, and set the controls to record mode.
8. From command mode, enter the following: CSAVEM"BASIC",&H0C00,&H****,&H80C0 The four stars above represent the number from step I.5; any name can be used in place of "BASIC".
9. The Basic program will save as a machine language file to tape.
10. The saving process is now complete.

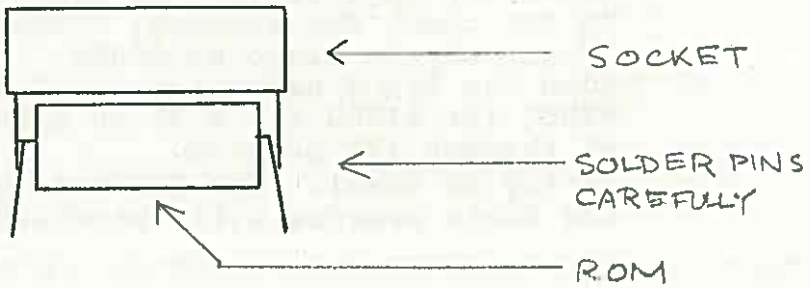
II. BURNING THE BASIC PROGRAM INTO THE EPROM

1. You will need a 2764, 27128 or 68766 EPROM prepared for Basic ROM downloading (the downloading routine resides in the EPROM from $0000 to $0040).
2. Bring the hardware and software Color Burner system up as usual.
3. Select a 2764, 27128 or 68766 EPROM from the menu #1, and insert the proper personality module.
4. Select load from tape (menu #2). You will be loading from $0C00 (first prompt).
5. You will be saving in the EPROM at $1000 (second prompt).
6. Prepare to load the tape; use the filename chosen in step I.8.
7. After the tape is loaded, prepare to burn the EPROM; do not check for erasure, since the download program is already in place at $0000.
8. Burn the EPROM beginning at $1000, and proceeding to $2000 (or $4000 for a 27128 EPROM). This will burn a 4K through 12K program.
9. Verify as usual. The EPROM is now ready to use, and the Basic program will autostart from a ROM pack.

A-2(43)
R 2/22/85

2764/27128 AUTO-BOOT ROM CONNECTED INTERNALLY (PIGGY-BACK ON COLOR OR EXTENDED COLOR ROM)





```
                        00100 ****************************
                        00110 * Special subroutine to *
                        00120 * block move a ROMmed   *
                        00130 * Basic program from    *
                        00140 * $D000 in ROM to Basic *
                        00150 * RAM, set pointers and *
                        00160 * do an autostart RUN   *
                        00170 *  USE WITH             *
                        00180 *  EXTENDED BASIC       *
                        00190 *  ONLY.                *
                        00200 *          Not to be    *
                        00210 * used with disk.       *
                        00220 ****************************
                        00230 *
            D000        00240 BASLST  EQU     $D000    * Basic stored here in ROM
            0019        00250 BASSTT  EQU     $0019    * Basic start stored here
            001B        00260 BASEND  EQU     $001B    * Basic end stored here
            0C00        00270 PCLER1  EQU     $0C00    * Start of PMODE0:PCLEAR1
                        00280 *
C000                    00290         ORG     $C000
                        00300 *
C000 86     55          00310 BPROGM  LDA     #$55     * Basic set up ID
C002 97     71          00320         STA     <$71     * Place Basic-set-up ID
C004 CC     80C0        00330         LDD     #$80C0   * Extended Basic restart
C007 DD     72          00340         STD     <$72     * Put vector in place
C009 BD     B95C        00350         JSR     $B95C    * Establish I/O parameters
                        00360 *
C00C 8E     D000        00370 BXFER   LDX     #BASLST  * Start of Basic-in-ROM
C00F 108E   0C00        00380         LDY     #PCLER1  * Start of Basic-in-RAM
C013 109F   19          00390         STY     <BASSTT  * Put start point in place
C016 0C     1A          00400         INC     <BASSTT+1        * Put NN01 as start
C018 EC     81          00410 MORE    LDD     ,X++     * Get ROM Basic byte
C01A ED     A1          00420         STD     ,Y++     * Store it in RAM Basic
C01C 1083   0000        00430         CMPD    #$0000   * End of program ID?
C020 26     F6          00440         BNE     MORE     * If not, get next byte
C022 ED     A1          00450         STD     ,Y++     * Else place four zeros
C024 109F   1B          00460         STY     BASEND   * Place end-of-program ID
                        00470 *
C027 8E     02DD        00480 RUNCR   LDX     #$02DD   * Point to keyboard buffer
C02A CC     5255        00490         LDD     #$5255   * Get letters "RU"
C02D ED     81          00500         STD     ,X++     * Place "RU" in buffer
C02F CC     4E00        00510         LDD     #$4E00   * Get letter "N" & delimiter
C032 ED     81          00520         STD     ,X++     * Place "N" and end buffer
C034 C6     04          00530         LDB     #$04     * Number of characters
C036 4F                 00540         CLRA             * Set Break flag off
C037 97     70          00550         STA     <$70     * Clear out I/O buffers
C039 8E     02DC        00560         LDX     #$02DC   * Front of keyboard buffer
C03C 7E     AC7F        00570         JMP     $AC7F    * Parse and interpret
                        00580 *
            C000        00590         END     $C000
00000 TOTAL ERRORS
BASEND   001B
BASLST   D000
BASSTT   0019
BPROGM   C000
BXFER    C00C
MORE     C018
PCLER1   0C00
RUNCR    C027
```

A-4 (45)
R 1/22/85

NEW SOCKET ANNOUNCEMENT: Color Burners shipped after 11/1/85 may be equipped with a "NIF" typ socket (black with orange tabs). To insert an EPROM, lift the tabs, set the EPROM in place, and press both tabs. Hold the board carefully To remove the EPROM, lift both tabs.

EPROM BURNER HOT SHEET

NOTE: The Color Burnware selection for 68764 EPROMs also programs 68766 EPROMs with no changes.

The latest version of Color Burnware contains the following groups of changes. Two are optional; one is mandatory.

Burnware Patch #1. This patch is MANDATORY.
Purpose: To correct low OE during 2764/128 programming, and to change 21V to 25V during 68764 programming.

CLOAD, RUN, THEN <BREAK>
QX=15616:POKEQX+17,165:POKEQX+18,161:POKEQX+21,165:
POKEQX+22,161:POKEQX+26,144 <ENTER>
Follow "COPYING THE SOFTWARE" on contents page of documentation, then use NEW COPY in future sessions.

FINAL PRINTING 2/23/87

Change the assembly listing as follows:

3D11	A5	00530	S2764	FCB	$A5
3D12	A1	00540	P2764	FCB	$A1 *
3D15	A5	00590	S27128	FCB	$A5
3D16	A1	00600	P27128	FCB	$A1 *
3D1A	90	00660	PSP1	FCB	$90

Burnware Patch #2. This patch is OPTIONAL.
Purpose: To correct run-on at "Verify Fails" message.

CLOAD, RUN, THEN <BREAK>
POKE16107,57:POKE16108,18 <ENTER>
Follow "COPYING THE SOFTWARE" on contents page of documentation, then use NEW COPY in future sessions.

Change the assembly listing as follows:

3EEB	39	04870		RTS	
3EEC	12	04875		NOP	

Burnware Patch #3. This patch is OPTIONAL.
Purpose: To change programming from 2732A (21-volt Vpp) to 2732 (25-volt Vpp). Consult your 2732 data sheet for programming voltage information.

CLOAD, RUN, THEN <BREAK>
QX=15616:POKEQX+13,146:POKEQX+14,144 <ENTER>
Follow "COPYING THE SOFTWARE" on contents page of documentation, then use NEW COPY in future sessions.

Change the assembly listing as follows:

3D0D	92	00470	S2732	FCB	$92
3D0E	90	00480	P2732	FCB	$90

*If you have revision 2.22, make this change only. Other changes have been made.

THIS SHEET REFLECTS REVISION 2.22A & REVISION 2.3

68764/68766 Users: Use the 68764 menu selection to program 68766 EPROMs. Also, note that a total of 25 PROGRAMMING PASSES MUST BE COMPLETED to fully program this EPROM. WAIT until programming is complete!!

FOR REFERENCE ONLY -- THESE CHANGES HAVE BEEN MADE IN YOUR COLOR BURNER & COLOR BURNWARE

FOR REFERENCE ONLY -- THESE CHANGES HAVE BEEN MADE IN YOUR COLOR BURNER & COLOR BURNWARE

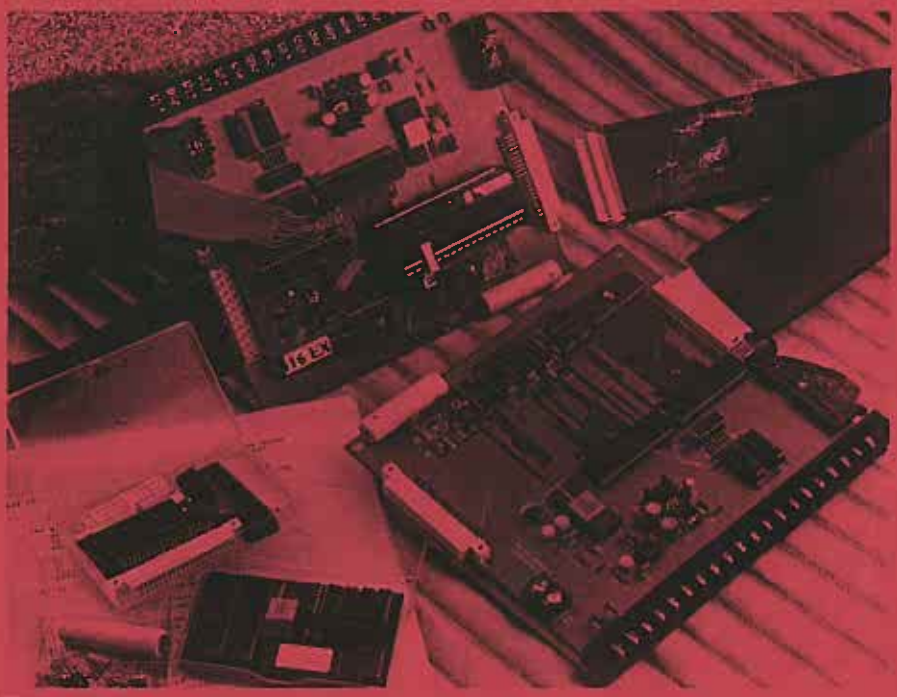
TO READ THIS

Rev-6 2/15/86
Rev-5 11/3/85
Repr. 10/10/85 n.c.
Repr. 9/24/85
Rep. 6/85
Rev.-4 3/25/85
Rev.-3 8/29/84
R-2 1/22/85
A-5 (46)

The Data Gatherer

New

A Professional 12-Bit, 16-Channel Data Acquisition System from Green Mountain Micro



Features:

- 16 analog input channels, each with 0-10 volt range.
- 12-bit (.025% accurate) analog-to-digital conversion.
- 20 sets of 16 complete A/D conversions per second!
- FAST conversion routine at 625 conversions per second.
- 12-bit digital-to-analog conversion output.
- 12,500 or more D/A conversions per second!
- Real-time clock/calendar with rechargeable battery backup.
- Parallel printer port (or use for other 10-bit I/O port).
- Operating system in ROM, with fully documented listings.
- Use with tape, disk or ROM packs, Basic or Assembly code.

Use for:

- **testing and measurement**
- **music synthesis**
- **laboratory control**
- **robotics**
- **temperature sensing**
- **monitoring systems**

Features high-performance, precision, laser-trimmed, 12-bit digital-to-analog converter.

$330.00 complete

Add $4.00 shipping and handling

For Professional and / or Industrial Use ONLY!

Complete, assembled and tested	$330
Complete with 32/64K computer	$550
Complete with 32/64K computer, EPROM programmer, and autostart software	$650
Parallel printer cable (optional)	$ 25
Complete kit of parts, board, manual	$220
Manual only (refundable with purchase)	$ 15

IMPORTANT! Specify:

16K Extended Basic version

32/64K Extended Basic version

32/64K Disk Basic version

R05038