

APPARAT PROM BLASTING SYSTEM DOCUMENTATION ADDENDUM  
SOFTWARE VERSION 3.4                      JUNE 2, 1982

This document describes the differences between APB software version 3.1 (as described in the manual) and version 3.4.

The APB software is now stored in a modified format binary file with a quick loader. This increases the size of the file slightly, but it will now load via a DOS "BRUN" command approximately twice as fast as the previous version. It should be noted that this prevents copying the file by the DOS "BLOAD" and "BSAVE" commands, however FID and most other generalized file copy programs will copy it properly. The easiest way to backup the software now is to use its own ~ command, as described below.

- II.            The main menu ~ command and the disk menu have been dropped. In its place are now the "L" (load) and "S" (save) commands of the main menu. These commands function in essentially the same manner as the old disk menu "L" and "S" commands, however the default disk slot and drive are now automatically set to those from which the software was run.
  
- III.           The main 'menu "S" command and the setup menu are now replaced by the main menu "A" (APB slot) and "P" (PROM type) commands. The "A" command will prompt you for a new APB card slot number, and the "P" command will allow you to select one of the ten supported PROM types. The copy of the APB software supplied on the distribution disk is configured to default to APB slot 7 and PROM type 2716 (single supply).
  
- IV.           A new command "U" (update) has been added to the main menu. This command will allow you to select disk slot and drive numbers and display the catalog of the disk in the same manner as the "L" and "5" commands. You will then be prompted for a filename. If you specify a filename, the APB software will write itself into the specified binary file. This can be used to backup the APB software. It should also be noted that the default APB slot and PROM type of the software written by the U' command will be those which were in effect at the time the "U" command was executed. If you use your APB card in slot 2, for example, you would probably find it advantageous to use the "A" command to select slot 2 and then use "U" to rewrite the software to disk. The new copy will then default to slot 2.

V. The C', "R", "R" and "V" commands (check erasure, read, blast, and verify PROM, respectively) no longer require the input of PROM and array start and end offsets, as partial PROM operations were found to cause problems in certain circumstances. Partial erasure of EEPROMS is not supported in this release. The new procedure for partial blasts is as follows:

1. Use the "R" (read) command to read the former contents of the PROM.
2. Make any desired changes to the array using the "E" (edit) and "L" (load) commands.
3. Reblast the PROM using the "B" command.

In order to prevent prom damage it is important that no bits that were clear (0) in the PROM originally are set (1) in the array when the blast command is used.

VI. The "S" (save) command no longer prompts for a reload address.

VII. The "L" (load) command will now display the length of the specified binary file and give you the option of only loading part of it.

VIII. The "F" (fill) command no longer is limited to 16 byte sequences; indeed, you may enter bytes until you have completely filled the array range you have specified if you so desire.

Ix. The "E" (edit) command now has two new subcommands: CTRL-B and CTRL-E. These commends move the cursor to the start and end of the array, repectively.

X. The manual neglects to mention at the bottom of page 7 that after using the Apple monitor to work on the array, the CTRL-Y monitor command is used to re-enter the APB software. Also, after entering the monitor with the "Q" (quit) command or by pressing reset, BASIC may be entered by the monitor command "3D0G".

The A.P.B.

-----

APPARAT PROM BLASTING SYSTEM

Version 3.1 Software for Apple DOS 3.3

INTRODUCTION

-----

The A.P.B. system is the most versatile and cost effective PROM programmer on the market. The system will program virtually all of the common 24 pin EPROMs and EEPROMS. Each PROM type is selected by a special personality module that adapts the programmer to the PROM. The following PROMs can be programmed:

Prom type	Prom size bits (bytes)	3 Supply	5 Volt only
-----	-----	-----	-----
EPROM:	4K(512)	2704	
	8K(1024)	2708	2508, 2758
	16K(2048)	2716(TI)	2516, 2716
	32K(4096)		2532, 2732q 2732A
	64K(8192)		68764
EEPROM:	16K (2048)		2815, 2816

The PROM programmer is a complete package. It includes powerful and easy to use software and personality modules for all of the PROMs listed above with the exception of the 2815 and 2816. The wiring diagram of all of the modules including the 2815/2816 is in the back of this manual; there are no extra modules to buy.

With the unique combination of personality modules and sophisticated software, the PROM programming system will perform many operations impossible with other systems. Here is a list of operations that can be performed:

- Erase EEPROMs
- Verify PROM is erased
- Read PROMs
- Fully or partially copy PROMs, even to different PROM types
- Fully or partially program PROMs, even 2704 and 2708
- Verify programming is correct
- Load or save PROM data to disk
- Program directly from computer memory
- Examine and/or edit working memory
- Preset working memory with any byte or sequence of bytes

The PROM programmer package consists of an interface card that

plugs into one of the Apple ][ expansion slots, a complete set of personality modules, disk based software for Apple DOS, and instruction manual.

#### USING THE A.P.B.

-----

Before you proceed with the operation of the A.P.B.. it is necessary to make a copy of the disk, as the one supplied is only partially formatted. This can be accomplished by using the FID program to copy the file "APPARAT PROM BLASTER" from the distribution disk to a formatted DOS 3.3 disk. It is also advisable to make some practice runs through the software to familiarize yourself with its operation.

#### GENERAL

-----

The A.P.B. package consists of three items: the programmer card, a set of personality modules, and disk based software. The personality modules plug into the programmer card and adapt it to the characteristics of the PROM type to be used. PROMs are programmed by inserting them into the zero insertion force socket on the programmer card, selecting and inserting the appropriate personality module, and running the software.

It should be noted that all inputs and outputs of the A.P.B. software are in hexadecimal.

\*\*\*\*\*

WARNING!!!

PROMs can be damaged by even small discharges of static electricity. It is recommended that you ground yourself before handling PROMs. This can be easily accomplished by touching the chassis of the Apple power supply.

Never install or remove A.P.B. card with power applied. This can cause damage to the A.P.B. card and your Apple.

Most PROM manufacturers do not specify whether it is allowable to remove or install PROMs into a circuit with power applied. Many of our customers change PROMs with power applied routinely, but this practice has been known to cause surges which can damage the A.P.B. card, so we do not recommend it.

A number of our customers have inquired as to why we have not put a power switch on our board as is present on some other PROM blasters. On the other boards we have examined, some use a switch to turn off the programming voltage to the PROM. The A.P.B. does not apply the programming voltage when not programming, so no such switch is needed. Other boards switch ground or YCC to the PROM, but it is not clear whether this really provides any protection unless the address, data, and control lines a-f the PROM are tristated. Also, on the 3 supply PROMs it would require a multiple pole switch to switch all of the supplies to the PROM. While we could in fact have included the switch and circuitry needed to implement these features, this would have pushed the price up considerably. We feel that since no other PROM programmers in the price range of the A.P.B. provided what we would consider adequate power-down protection, that the inconvenience of having to power down between PROMs was worth the lower cost.

If you feel you need to have power down capability, it should be noted that for all of the 5 volt only PROMs which can be used with the A.P.B., power can be removed by unplugging the personality module, but this is not recommended.

\*\*\*\*\*

Programming a PROM consists of eight steps:

- 1) Erase (EPROM)
- 2) Set up programmer
- 3) Erase (EEPROM)
- 4) Load data to be blasted
- 5) Do actual blast
- 6) Remove PROM

## ERASE EPROM

EPROMs are erased by exposing them to ultraviolet light through a transparent (sometimes translucent) window in the middle of the top side. Erasure sets all bytes of the EPROM to hex FF. Ultraviolet EPROM erasers are available from most computer stores. Erasing an EPROM takes from 20 minutes to an hour, depending on the type of EPROM and the intensity of the eraser. To insure complete erasure, follow the manufacturer's instructions.

## SET UP PROGRAMMER

With the control lever in the open (upward) position, install the PROM to be programmed in the zero insertion force socket. Be sure that pin one of the PROM is aligned with the index mark on the socket. Lock the PROM into the socket by moving the control lever to the closed (down) position.

Select the proper personality module from the chart below and install it in socket U1. Be sure that the index mark lines up with pin one.

### PERSONALITY MODULE TABLE

ROM TYPE	MODULE TO USE
2704 2708 .....	2708
TI 2716 (3 supply).....	2716 (3 voltage)
2508, 2516, Intel 2716, 2758...	12716 (5 volt only)
2532.....	2532
2732.....	2732
2732A .....	2732A (21 volt)
68764 .....	68764
2815, 2816 .....	2816 (Not supplied)

With the power off, install the A.P.B. card into an empty slot in the Apple. You may use any slot but zero. If you are going to be programming several PROMs, install the programmer into a slot that gives clear access to the PROM socket.

## BOOT SOFTWARE

Turn the power on and boot your disk containing a copy of the A.P.B. software. Type "BRUN APPARAT PROM BLASTER'. This will load and run the software. The software will first ask you which slot your A.P.B. is plugged into. After responding with a number

between 1 and 7 you will be presented with a list of PROM types and the program will ask which type you will be using. If the A.P.B. software should give **you** an error message at any point after the software is booted, check the following:

- Card in wrong slot
- PROM in socket backwards
- Wrong personality module
- Personality module inserted backwards
- Try erasing for a longer period of time

If these steps fail to correct the problem, the PROM is probably defective, try another one.

#### ERASE EEPROM

EEPROMs are erased by applying special signals to certain of their pins. If you are using an EEPROM type "E" for erase. The program will ask you for a PROM START OFFSET and a PROM END OFFSET at which times you may simply press <RETURN> to erase entire EEPROM. or start and end addresses to specify a range of bytes to be erased.

#### LOAD DATA

The data to be written in the PROM must be loaded into a working array inside the program. To make things flexible, the data can be loaded from several different sources:

- from disk file
- from another PROM (see Copy section)
- directly from memory
- from the keyboard

The use of each of these sources of data is described in detail in other sections of the manual. Choose the appropriate method for your application.

#### DO ACTUAL BLAST

After you have loaded the data and are at the main menu, select "B" for Blast. You will be asked for an ARRAY START OFFSET (default zero). Normally you will just press return to program the PROM from data starting at the beginning of the array. If you wanted to blast the PROM from another address in the array, for instance if you wanted to blast a 2716 from the second half of a data file for a 2732, you can enter the relative address in the array of the first byte you want blasted into PROM. You will next be asked for the PROM START OFFSET and the PROM END OFFSET,

which default to zero and the end of PROM, respectively. Again, you would normally press <RETURN> in response to both questions, however if you only wanted to blast part of the PROM you could specify the start and end of the range you wanted to blast.

After specifying the blast parameters, the program will check to see if the PROM is erased in the range to be blasted. If not, the program will ask you whether you wish to proceed. In general, unless you wanted to blast over the bytes already in the PROM (having a net effect of a logical AND), you should enter "N" and erase the PROM.

At this point the actual blast will begin. The multi pass PROMs (2704, 2708. 66764) require somewhat more time to blast than the single pass types. The software will indicate the start of each pass. The following are approximate programming times for each PROM type:

2816 . . . . .	25 sec
2704, 2508, 2758. 2816 . . . . .	55 sec
2708, 2516. Intel 2716. 2815 . . .	1 min 45sec
TI 2716, 2532. 2732(A), 68764 . .	3 min 25sec

Once the blast is complete, the contents of the PROM will be compared with the array to verify that data has been correctly programmed into the PROM. If any discrepancies are noted, the program will give you the option to display the addresses and data in question on the screen.

#### REMOVE PROM

Turn off the Apple ````

Then remove the PROM from the card. If you are going to program several PROMs you can leave the card in the slot. If this is the only PROM you are going to blast you may remove the card before removing the PROM. To prolong the life of the slat edge connector, you should minimize insertions and removals from the slot.

#### COPYING PROMS

Copying a PROM is similar to the programming process described above. The process involves reading the PROM to be copied into memory, then saving the data to disk. Once it is saved one or more copies can be made as described above.

Here are the steps:

- 1) Set up the card and install the PROM to be copied. This is described in detail in step two of the programming sequence above.

- 2) Boot the software and enter the slot and PROM type.
- 3) Select "R" to read the PROM into the array. You will be prompted for the PROM START OFFSET, the PROM END OFFSET, and the ARRAY START OFFSET. To read the entire PROM into the beginning of the array, just press return in response to each prompt.
- 4) Now select the "D" option for disk I/O. Select "S" to save the array to disk. Loading and saving data is described in detail in the software section of this manual.
- 5) Remove the PROM as described in step seven of the programming instructions above.
- 6) You can now make one or more copies of the PROM using the normal programming sequence above.

#### THE WORKING ARRAY

Many of the operations of the A.P.B. package are performed between the A.P.B. card and an area of memory called the working array. The working array is simply memory space set aside by the program to hold PROM data. The working array can be loaded from disk, saved to disk, read from memory, read from PROM, blasted into PROM, and edited.

Sometimes it is necessary to perform operations on the working array not provided by this software, I.E. disassembly, printing, etc. In these cases one can use the "0" command to exit the software, and then use monitor commands or any other available means to perform the desired operations. The working array is 8K in length and resides from addresses \$4000 to \$5FFF

## USING THE SOFTWARE

-----

After the software is booted as described in the programming instructions above, and the A.P.B. slot number and PROM type have been entered, the main menu will be displayed and you will be asked to pick a selection:

\*\*\*\*\* MAIN MENU \*\*\*\*\*

- S) SETUP PROM TYPE OR BLASTER SLOT
- C) CHECK PROM ERASURE OR ERASE EEPROM
- R) READ PROM INTO ARRAY
- B) BLAST PROM FROM ARRAY
- V) VERIFY PROM AGAINST ARRAY
- D) DISK I/O TO/FROM ARRAY
- E) EDIT ARRAY
- M) MOVE MEMORY INTO ARRAY
- F) FILL ARRAY
- Q) QUIT

SELECT OPTION:

### SETUP PROM TYPE OR BLASTER SLOT

-----

If for any reason you would want to change the PROM type or A.P.B. slot number, you can use this option to do so.

### CHECK PROM ERASURE OR ERASE EEPROM

This option will allow you to specify a start and end PROM offset, and will check the PROM for erasure in this range, listing any discrepancies. Most EPROM erasure errors are due to under erasure. When an EPROM is under erased, the data may be temperature sensitive.

When using this option with EEPROMs. you will be asked whether to erase the specified range of addresses. Note that it takes much longer to erase a of the EEPROM than the whole thing, as EEPROMs provide a fast bulk erase made.

### READ PROM INTO ARRAY

After prompting you for a range of PROM addresses and an array start offset, this will read the range of the PROM into the array starting at the specified offset. The data can then be edited, saved to disk, etc.

## BLAST PROM FROM ARRAY

-----

This option will prompt you for a range of array addresses and a PROM start offset. It will then proceed to blast bytes from the array range into the PROM starting at the specified offset. The multi-pass PROMs (2704, 2708, 68764) take much more time to blast than the single pass types, so the software will indicate which pass number it is performing. The single pass PROMs program much faster, so no pass information is displayed.

After the blast is completed, an automatic verify is performed. Any discrepancies will be displayed.

## VERIFY PROM AGAINST ARRAY

-----

After prompting you for a PROM address range and an array start offset, this option will perform a byte by byte comparison of the PROM and array data. If all bytes are correct the message "GOOD BLAST" will be displayed. Otherwise a list of offending addresses and data bytes will be displayed.

## DISK I/O TO/FROM ARRAY

-----

This option will display the following disk I/O submenu, and prompt you to select an option:

\*\*\*\*\* DISK MENU \*\*\*\*\*

- C) CATALOG DISK
- 6) SAVE ARRAY TO DISK
- L) LOAD ARRAY FROM DISK
- D) DELETE FILE
- L) LOCK FILE
- U) UNLOCK FILE
- V) VERIFY FILE
- R) RETURN TO MAIN MENU

SELECT OPTION:

## CATALOG DISK

-----

This option will prompt you for slot and drive numbers, and display the catalog of the selected drive. This drive becomes the new default for the other disk commands.

SAVE ARRAY TO DISK

This selection will prompt you for a range of array addresses, a file name. and a reload address (default \$800). The specified range of array bytes will be written into the file on the default drive. The reload address is written into the file so DOS will have an address to load the file into with its "BLOAD" and "BRUN" commands.

LOAD ARRAY FROM DISK

-----

After specifying a filename and a array start offset, this command will load the file into the array at that offset. The data can then be edited, saved to disk, etc.

DELETE, UNLOCK, LOCK, VERIFY FILE

-----

These commands will prompt for a filename and then delete, unlock, lock, or verify the specified file.

RETURN TO MAIN MENU

-----

Entering this selection will return you to the main menu.

COPY MEMORY INTO ARRAY

-----

This command will allow you to copy data from any part of memory directly into the array by specifying the memory address range and array start offset.

FILL ARRAY

-----

This option will ask you for an array address range, then will allow you to enter from 1 to 16 bytes, After the last byte entered, just press return. The specified series of bytes will then be repeated to fill the array range.

## EDIT ARRAY

-----

This option takes you into edit mode. In edit mode the screen will display one full page (256 bytes) of memory at a time in the following format:

```
XXXX:00112233 44556677 B899AABB CCDDEEFF
```

where XXXX is the array offset and 00 through FF are data bytes. the current byte will be displayed in inverse. The current byte can be changed by directly entering hex data. After every two hex digits entered the cursor will move to the next byte. The following commands may be used in edit mode:

cursor:	up	G) Go: prompts for new cursor offset
	I	R) Return to main menu
left J	K right	
	M	Left arrow: move cursor one page
	down	Right arrow: move cursor forward one page

## HARDWARE NOTES

-----

The A.P.B. board has been designed to blast all of the common 24 pin PROMs. Most of the pinouts on these PROMs are identical. The only differences exist on pins 18, 19, 20, 21, and 24. These pins require various voltage levels depending upon the PROM type. These pins are connected to the personality module socket on the card. Other pins on the personality module socket provide special logic levels and voltages needed by the different PROM types. By installing headers with appropriate jumpers, the proper logic levels can be connected to the proper PROM pins for each PROM type.

The following special signals are available at the personality module socket:

All	Address bit 11, provides most significant address bit for 4K PROMs
A10	Address bit 10, provides most significant address bit for 2K PROMs
PPH	Program Pulse High, can be switched between 0, 5, and 26 volts
PPL	Program Pulse Low, can be switched between 0, 5, and 12 volts
TTL1	TTL level pulse, can be used for chip select, output enable, etc.
-5v, +5V, +12V	Power supply lines

## ROM ADDRESSING

-----

The address lines for the PROM are provided by a 12 bit binary counter on the A.P.B. card. This takes the PROM off of the Apple's address bus and allows the programmer to program large PROMs without interfering with the Apple's PROM space. The counter can be reset and incremented under software control.

## THE PIA

-----

Interfacing of the A.P.B. card to the Apple bus is provided by a 6821 PIA. The PIA registers are addressed as follows, where X is the A.P.B. slot number plus 6:

Port A	Data Direction Register	\$COX0
	Data Port	\$COX0
	Control Port	\$COX1
Port B	Data Direction Register	\$COX2
	Data Port	\$COX2
	Control Port	\$COX3

Port B of the PIA is connected directly to the data bus of the PROM. Since the port can be programmed as either an inputs or outputs. data can be either read or written to the PROM.

Port A is used to control the address counter and the special personality module logic lines. Each bit of port A is set up as an output and controls the following functions:

Bit:	7	6	5	4	3	2	1	0
	PPH1	PPHO	PPL1	PPLO	TTL1	-	RESET	COUNT
PPH1	PPHO	PPH Voltage		PPL1	PPLO	PPL Voltage		
----	----	-----		----	----	-----		
0	0	Ground		0	1	Ground		
0	1	+5		0	1	Ground		
1	0	+26		1	0	+12		
1	1	Tristate		1	1	+5		

Bit 3 controls the TTL1 line.

A high pulse on bit 1 resets the address counter.

A high pulse on bit 0 increments the address counter.

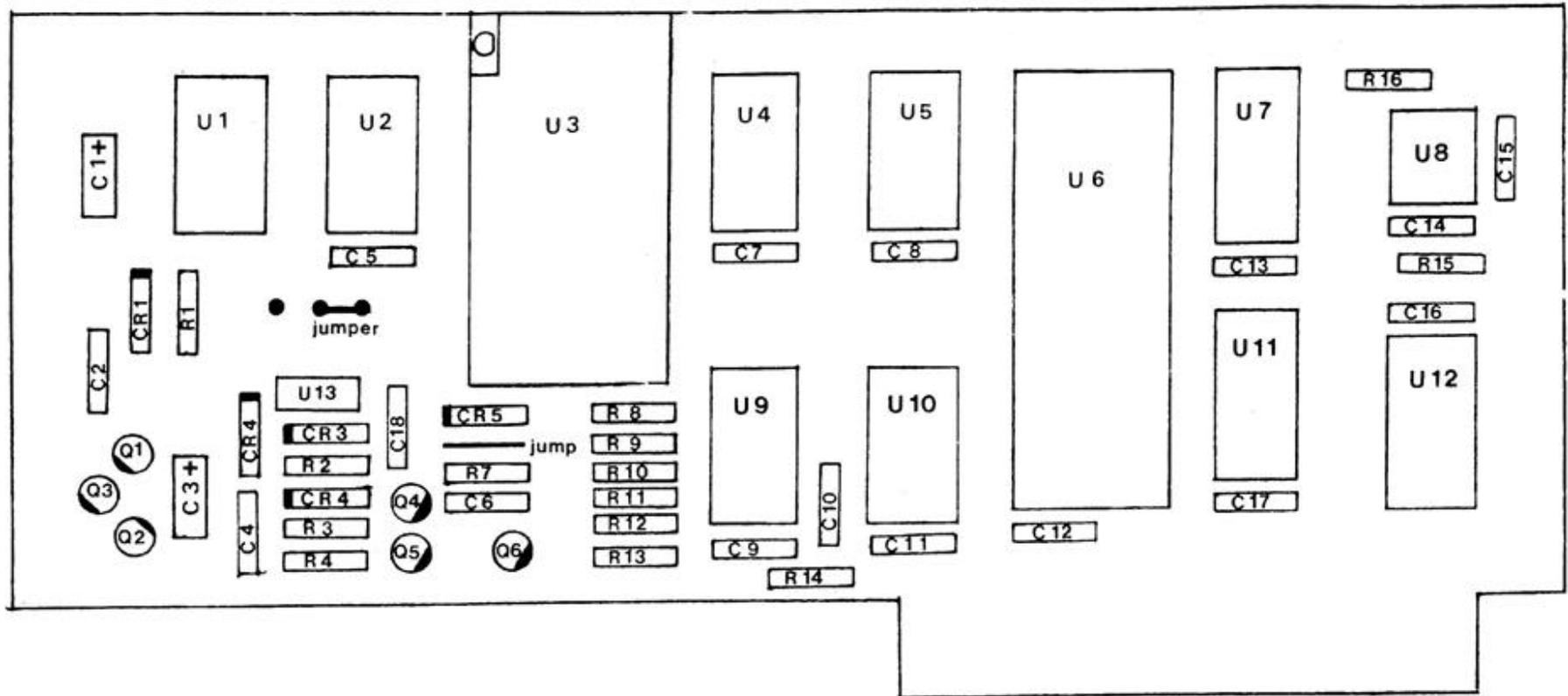
The 26 volt programming pulse is provided by a charge pump which converts +12V and -12V into +36V. The +36V supply is regulated down to +26V and fed to the pulse control logic.

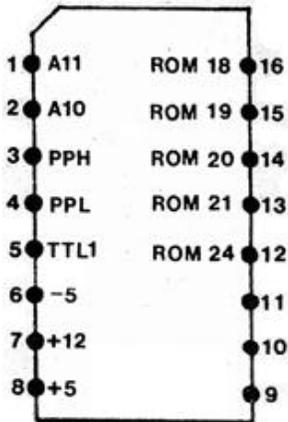
PARTS LIST

-----

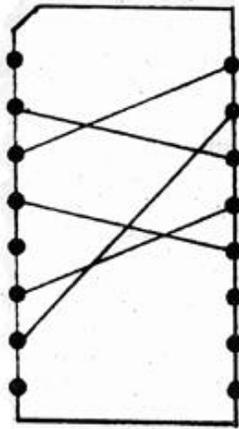
U1	Personality Module
U2	4040
U3	ZIF PROM socket
U4	74LS02
U5	74LS04
U6	6520, 6820, or 6821
U7, U11, U12	74LS367
U8	555
U9	7407
U10	74LS00
U13	7824
C1, C3	3.3
C2, C5-C9, C11-C14, C16-C18	.1
C4	.06
C10	.22
C15	.001
R1, R15	1K
R2, R12, R13	10K
R3	10
R4	50
R7	390
R8	6.8K
R9	2.2K
R10, R11	2.7K
R14	1.5M
R16 47K	
CR1, CR2, CR4	1N4001
CR3	3.2V Zener
CR5	5V Zener
Q1-Q3, Q6	2N2222 NPN
Q4, Q5	2N2907 PNP

Note: Unless otherwise specified, all capacitances are in Mfd, resistances are in ohms, and all resistors are 1/4 Watt.

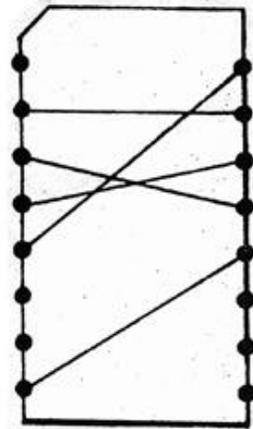




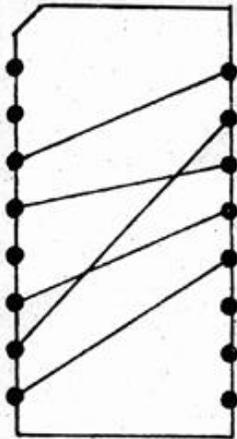
SIGNALS



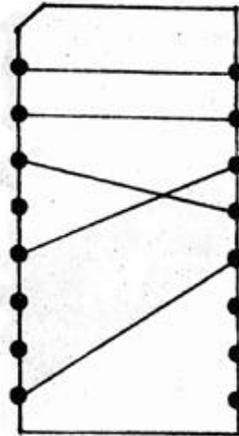
2716  
3 Voltage



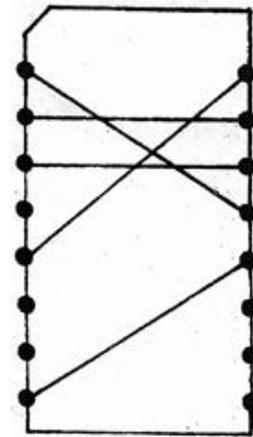
2716  
5 Volt Only



2708

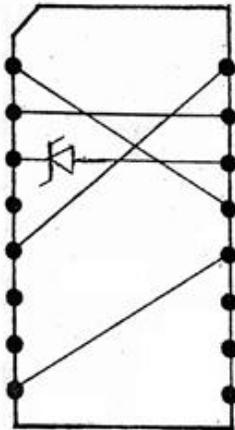


2532

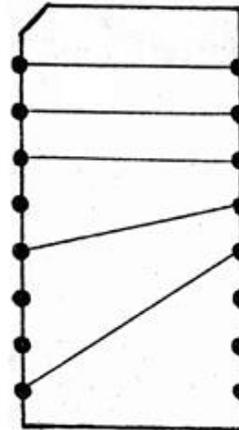


2732

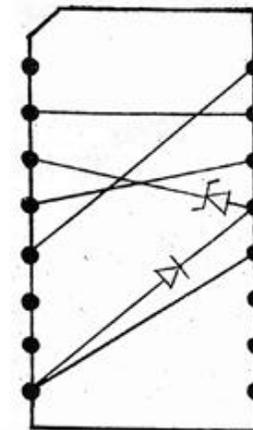
Zeners are 1N5231 Diodes are 1N4001



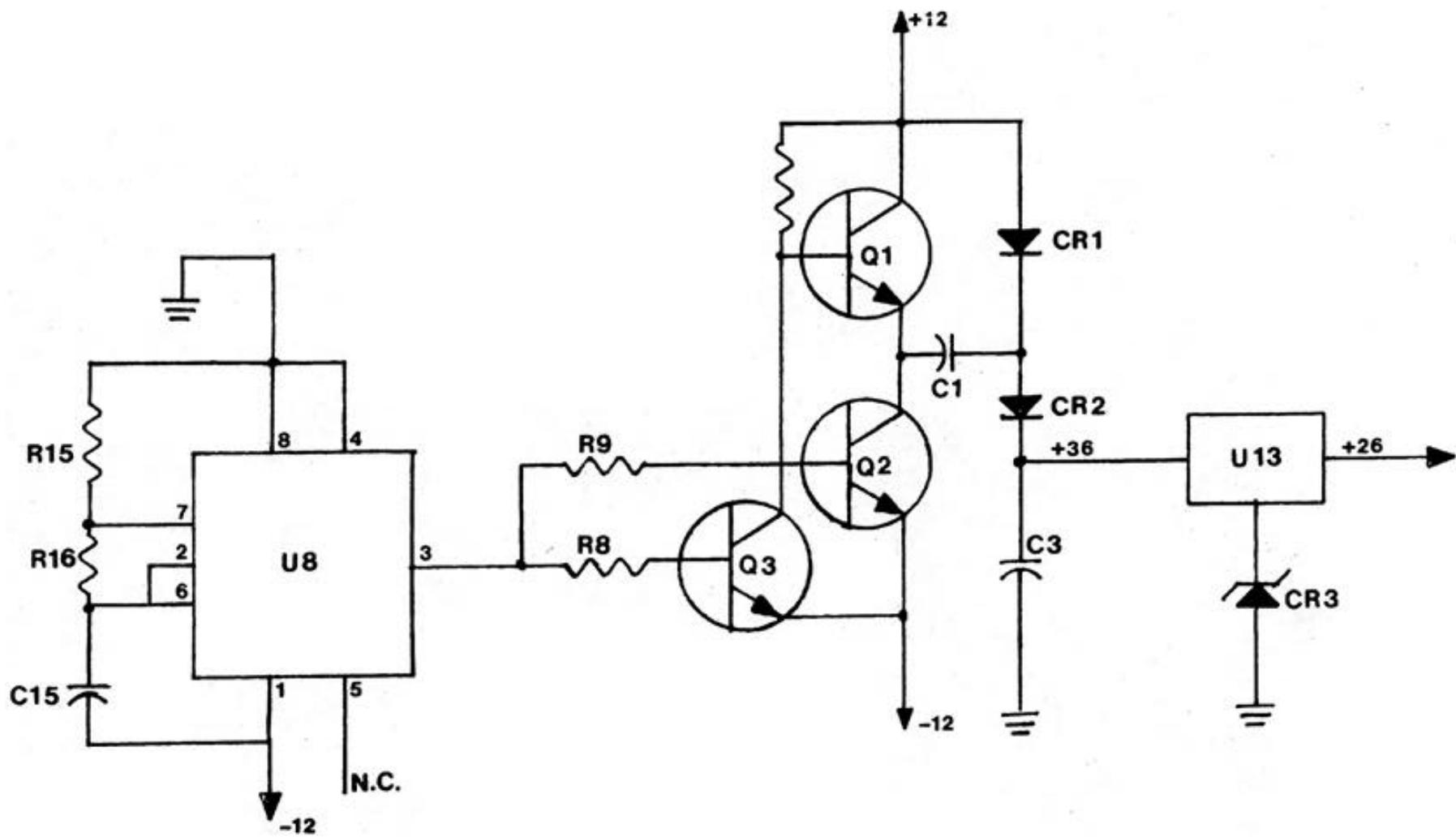
2732A

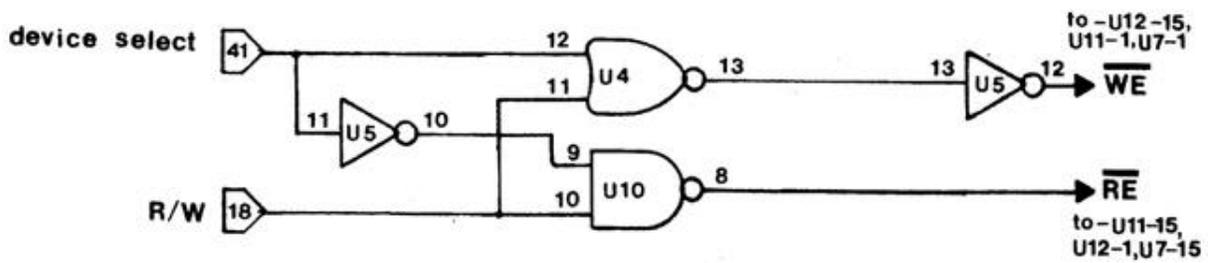
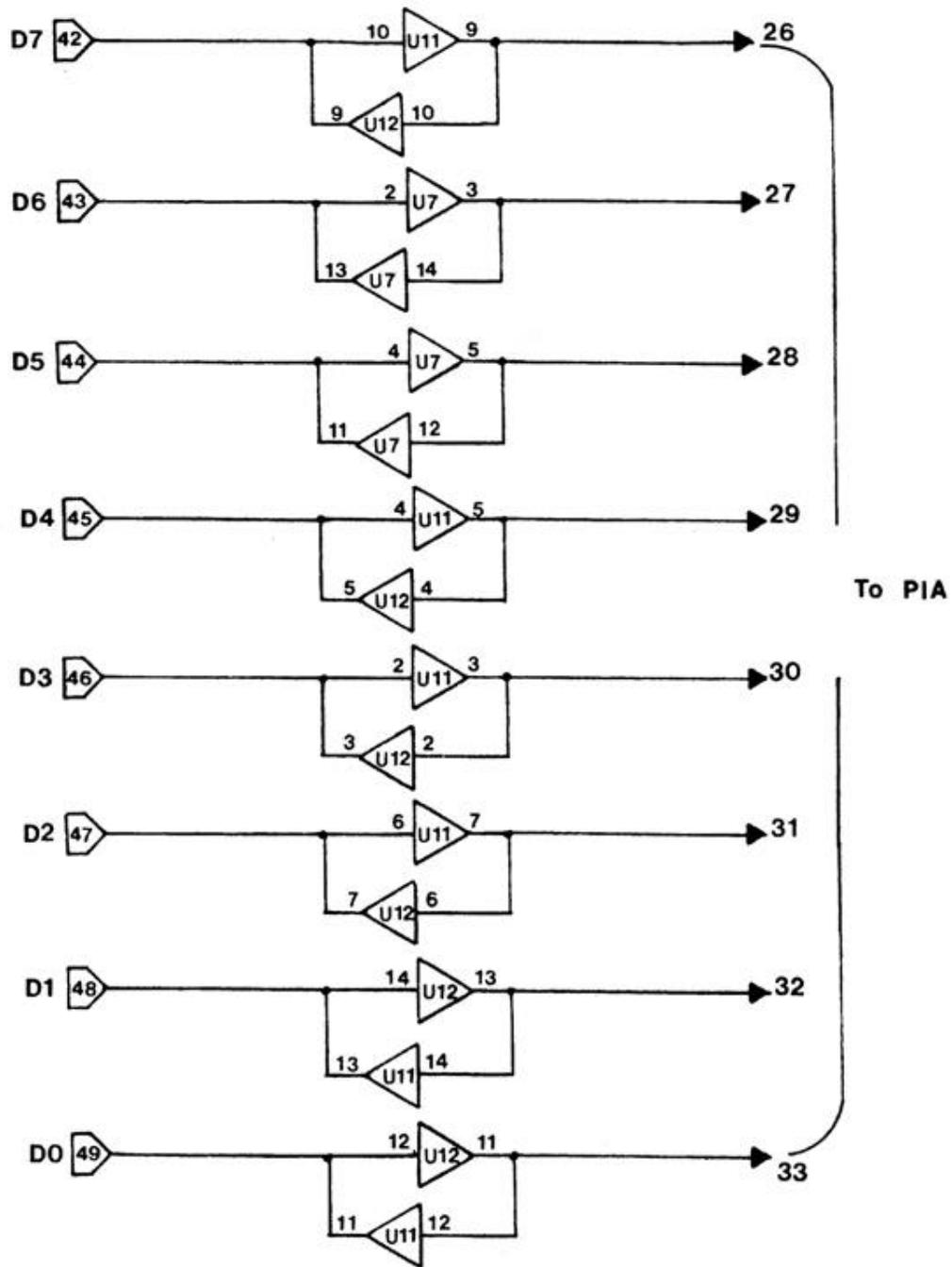


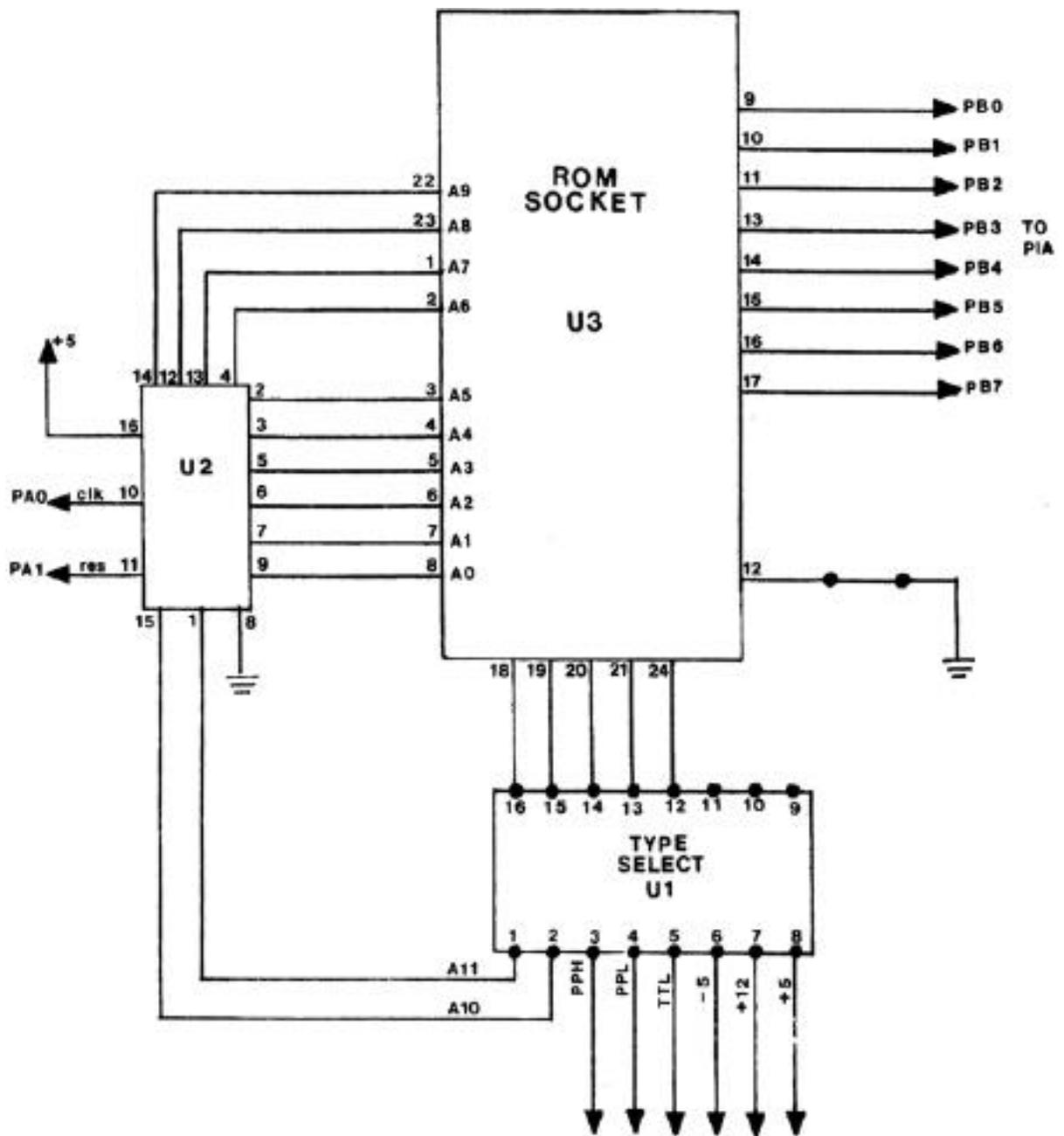
68764

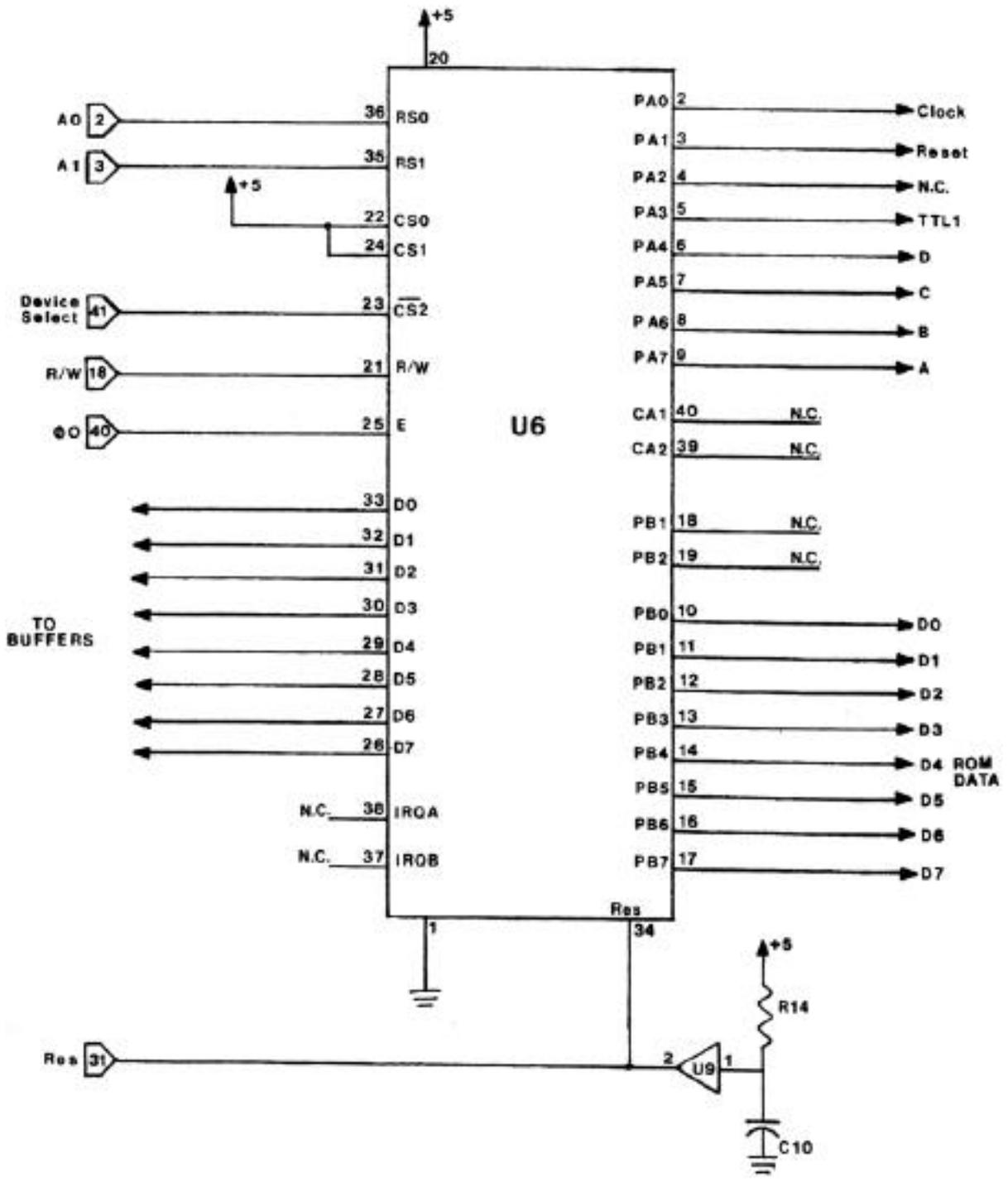


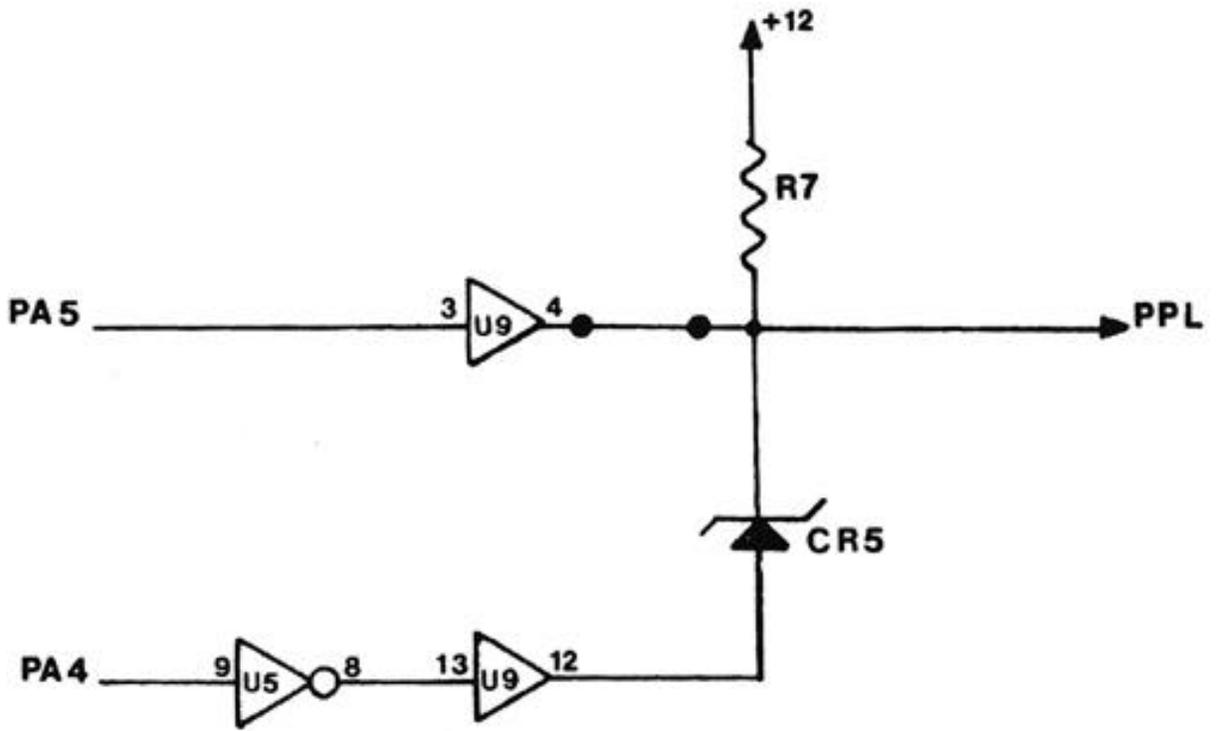
2815/2816



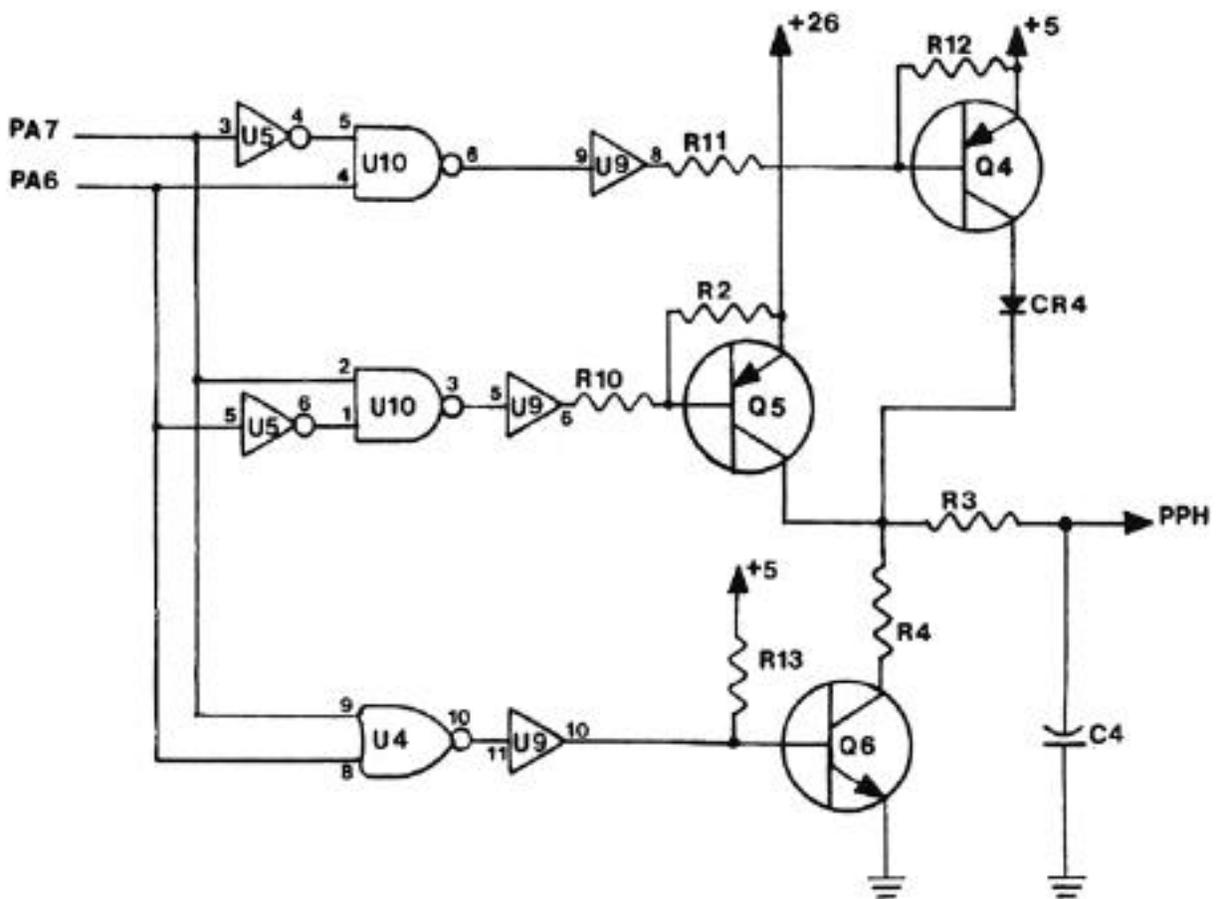








PA5	PA4	PPL
0	0	0
0	1	0
1	0	12v
1	1	5v



PA-7	PA-6	PPH
O	O	GND
O	I	+5
I	O	+26
I	I	TRI

The A.P.B.

APPARAT PROM BLASTING SYSTEM

INTRODUCTION

The A.P.B. system is the most versatile and cost effective EPROM programmer on the market. The system will program virtually all of the common 24 pin EPROMs. Each EPROM type is selected by a special personality module that adapts the programmer to the EPROM. The following EPROMs can be programmed:

2704	2708
2716	2716 (5 volt)
2732	2508
2516	2532
2758	68764

The EPROM programmer includes powerful and easy to use software and a complete set of personality modules. With the unique combination of personality modules and sophisticated software, the EPROM programming system will perform many operations impossible with other systems. A list of operations that can be performed follows:

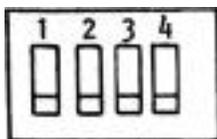
- Verify EPROM is erased
- Read EPROMs
- Copy EPROMs
- Copy between different EPROM types
- Read or save EPROM data on disk
- Program EPROMs
- Verify that EPROM was programmed correctly
- Program directly from computer memory
- Examine and/or modify memory
- Preset memory

The EPROM programmer package consists of an interface card, a complete, set of personality modules, software on disk, and an instruction manual. The software will run under IBM DOS.

The IBM PROM Burner is designed to be operated in one of the slots of the IBM Personal Computer. The EPROM programmer will operate on any slot. (It is recommended that the slot next to the EPROM socket remain empty to allow easy access to the socket.)

Always turn off the computer before inserting or removing any card. (Make sure the card is firmly seated before turning the power on.)

## SELECT SWITCH



### Select Switch

The switch on the card configures the address of the PROM Burner in the computer's port space. The card uses 4 bytes of the port space. The exact address depends on the switch settings - the PROM Blaster can reside anywhere between 280H and 2BFH. The exact address is:

$280H + 4*n$  where  $n$  is a four bit number  
determined by the switch settings.

Switch 3 is bit 0, switch 0 is bit 3, etc. When a switch is 'on' the bit is a zero and when it is 'off' the bit is a one.

Example:

Switch3	Bit	0	=	0
Switch2	Bit	1	=	1
Switch1	Bit	2	=	0
Switch0	Bit	3	=	0

$n = 2 \Rightarrow \text{Address} = 280 + 4*n$   
 $= 280 + 4(2)$   
 $= 288H$

The supplied software will work only if the select switches are all set to 'on'.

## USING THE A.P.B.

Before you proceed with the operation of the A.P.B., it is advisable to make a copy of the disk and set it aside. You may use a DOS disk to make a backup. It is also advisable to make some practice runs through the software to familiarize yourself with its operation.

## GENERAL

The IBM PROM Blaster consists of three items: the Programmer Card, a set of Personality Modules, and disk-based software. The Personality Modules are 16 pin plug-in adapters that configure the programming socket to match the type of EPROM being programmed. The Personality Module plugs into the Programming Card and adapts it to the characteristics of each EPROM type. EPROMs are programmed by inserting the EPROM into the card, selecting the proper personality module and running the software.

\*\*\*\*\*

!!!!           WARNING       !!!!

          EPROMs can be damaged by static electricity.  
          NEVER install or remove EPROM with the power on.  
          NEVER remove or install IBM PROM Blaster card  
          with the power on.

\*\*\*\*\*

## SOFTWARE PACKAGE

The software is included on a 5-inch disk as a part of the A.P.B. package. The software will run under the IBM DOS. Instructions for using the software are provided later in this manual.

## WORKING MEMORY

Many of the operations of the IBM EPROM Programmer are performed between the EPROM Programmer card and an area of memory termed "working memory". The working memory can be read from disk, written to disk, read from memory, read from EPROM, written to EPROM and edited.

## PROCEDURE

Programming an EPROM consists of five steps:

- 1) Erase EPROM.
- 2) Set up programmer.
- 3) Load data to be blasted.
- 4) Verify that the EPROM is erased. Do actual blast and verify correct blast.
- 5) Remove EPROM.

## ERASE EPROM

All of the EPROMs programmed by the IBM PROM Blaster system are erased using ultraviolet light. Each EPROM has a small glass window. Shining ultraviolet light through this window erases the EPROM. In the erased state every byte in the EPROM is set to hex FF.

Ultraviolet EPROM erasers are available from many computer stores. Erasing an EPROM takes from a half hour to an hour, depending on the intensity of the eraser. To ensure complete erasure, follow the EPROM manufacturer's instructions.

## SETTING UP IBM PROM BLASTER TO PROGRAM

With the control lever in the open position, install the EPROM to be programmed in the zero insertion force socket. Be sure that Pin 1 of the EPROM is aligned with the index mark on the socket. Lock the EPROM in the socket by moving the control lever to the closed position.

Select the proper personality module from the chart below and install it in Socket IC12. Be sure that the index mark lines up with Pin 1.

EPROM TYPE	PERSONALITY MODULE TABLE MODULE TO USE	DRIVER NAME
2704,2708	2708	I2708
TI 2716 (3 voltage)	2716 (3 voltage)	T12716
2508	I2716 (5 volt)	I2716
2516	I2716 (5 volt)	I2716
Intel 2716 (5 volt)	I2716 (5 volt))	I2716
2532	2532	T12532
2732	2732	I2732
2732A	2732A (21 volt)	I2732
68764	68764	M68764

## DO ACTUAL BLAST

Load PROM data into working memory by using EDIT or by reading another EPROM, then return to the main menu. Select the 'Write EPROM' option. The amount of time required to blast an EPROM depends upon the EPROM type. The 3 voltage EPROMs (2704, 2708, and T12716) require much more time because they require one hundred passes to program, while the single voltage EPROMs require only a single pass.

If there are errors during the write, check the following:

- EPROM in socket backwards.
- Wrong personality module.
- Personality module in backwards.
- Try erasing for a longer period of time.

If these fail to correct the problem, the EPROM is probably defective: try another EPROM.

REMOVE PROM (Turn Off the Computer!!!)

Remove the EPROM from the card. If you are going to program several EPROMs, you can leave the card in the slot. If this is the only EPROM you are going to program, you may remove the card before removing the EPROM. To prolong the life of the slot edge connector, you should minimize insertions and removals from the slot.

## SOFTWARE OPERATION

\*\*\*\*\*  
All Values in Hexidecimal  
\*\*\*\*\*

Turn the power on and boot the EPROM disk. Then run the program by typing:

A>E PROM

The next step is to load the prom data. Make sure the correct disk is mounted and then do option 'G' from the MAIN MENU. If you wish to use a different prom than the 'current prom' then type 'E' to enter EDIT MENU and 'C' to change prom. (See page 8 for details.)

If you wish to verify that an EPROM contains a certain set of data or compare one prom to another then you can load the working array with the data and VERIFY WRITE.

Verify prom is blank:

This option verifies that the EPROM contains 'FF's in the range specified. Choose 'B' to verify that the EPROM is blank. The computer will then ask for the start and end addresses: these addresses refer to absolute prom addresses. The addresses are inclusive and are entered in hexadecimal form.

Read prom:

This reads the data of an EPROM into the working memory. It again asks for an EPROM range as above and also asks for a Buffer start address. The Buffer start address is where the first byte in the prom will be stored in the working array. The Buffer start + number of bytes being read cannot exceed X'2000'.

Write prom:

This writes a prom from the working array. It asks for a prom range and Buffer start: 'Buffer start' being the first byte to read. If the EPROM being written is a multi-pass EPROM then the number of passes remaining will be displayed. After the write is completed, the contents of the EPROM will be verified and any discrepancies will be noted in the form:

ERROR at XXXX-YY found: ZZ expected (bell)

X 'XXXX': absolute prom address of error  
X'YY' : byte prom contains  
X'ZZ' : byte written from working array

Verify write:

This will compare a portion of the working array with a prom. It asks for a prom range and Buffer start: 'Buffer start' is the first byte to be verified. Any differences are reported in the form of a write error.

Input data file to buffer:

This reads a file into the working array. The first prompt is 'What file to read?'. Make sure the correct disk is mounted and enter the file name in the correct DOS format. It will then ask you 'Concatenate files?'. If you answer no then the file will be read into the start of the working array. If you answer yes then the computer will ask you 'Start where?' - your answer will be the place in the working array where the first byte read will be stored. The file is read until the EOF is reached ( the whole file) or until the buffer is full (X'2000' bytes) or until a read error occurs. The data previously in the buffer that is not overwritten then remains intact.

Output buffer to data file:

This function outputs the buffe>r to a data file. The first question is 'What file to write?' - make sure a disk with enough room is mounted. The next question is 'How many bytes?' - this is the first byte to write to the disk file. Next - 'Buffer start address?'. The file is written until the data is saved or until a disk error occurs.

Edit:

See EDIT MENU.

Get prom data from disk:

This reads the file 'PROM.DAT'. The file contains all the pertinent information necessary to read and write EPROMs. Unless you wish to create a new prom (see EDIT) you must do this before any prom operations.

Put prom data to disk:

After creating a new prom, or editing an old one, execute this to save your changes to disk for next time.

To quit and return to DOS:

Type CTRL-C or CTRL-Scroll/Lock.

EDIT MENU:

Type 'E' from main menu to enter 'EDIT MENU'.

Edit buffer:

This allows you to enter/edit the working array. The computer asks you for a range: this is the first and last byte to be edited. The format is:

XXXX:YY? ZZ

where X'XXXX' is the address in working memory.  
X'YY' is the current value at location X'XXXX'.  
zz is your input, the new value.  
'.' alone exits edit and returns to Edit

Menu.

Move memory to buffer:

This is to move computer memory into working memory. The computer first asks you for a memory address in Segment/Offset format and then for the 'Buffer start'.

Fill buffer:

This will fill a certain range of the buffer with a certain sequence of bytes. The first query is that of the range to be filled. Next, how many bytes, referring to the length of your sequence. You are then prompted for each individual byte.

Set TOB:

This sets the Top of Buffer. Usually when you are working with a certain EPROM the TOB is set to the length of the prom, but for some applications you need extra room. This option lets you set the length of the buffer. The maximum is X'2000'.

Change current prom:

This lets you change the EPROM you are reading, writing, or verifying. It will prompt you for an EPROM name -- enter the name exactly. TOB is adjusted to the length of the EPROM. If you have data past the new TOB simply change back.

Create new prom:

If new compatible (see hardware notes) EPROMs are released in the future, this allows you to create new prom specs.

The questions -

<S>ingle or <M>ultiple passes: This is by which method the prom programs. (See hardware notes)

Read byte : This is the byte sent before a read.(See hardware notes)

Write1, Write2 : These are the bytes sent before a write. (See hardware notes)

Length : This is the length of the prom; e.g., 800 for a 2708.

Return:

Return to Main Menu.

Port Change:

This should not be necessary. But if there is a conflict between the EPROM Blaster and another card then it might be necessary to change the communication ports on our card. This requires both hardware and software changes. Under EDIT MENU type '0'. It will then ask you for the new ports. See hardware notes for details.

NOTE TO 64K RAM USERS:

The program EPROM" is too big to fit in 64K of memory. Because of this Apparatus has split the functions of EPROM into three separate programs — these are: "READ", "BLAST", and "EDIT". READ reads an EPROM and stores its contents into a disk file; BLAST reads a disk file and then, after confirming that the EPROM is blank, blasts the data to the prom. EDIT is used to both edit files for reading, writing, and editing the EPROM data file (files that contain information as to how to blast specific EPROMs). As with EPROM, files to be blasted can be created in other ways as long as the resulting file is in the form of a 'COM' file (e.g., LINK.COM). Examples are DEBUG or an assembler file.

To invoke READ or BLAST the user types:

```
A:read filename
      or
A:blast filename
```

The 'A:' is the DOS prompt for drive 'A'. READ and BLAST can, of course, be loaded from any other drive. The file 'filename' does not have to be on the same drive as READ or BLAST; if it is not, just add a drive prefix to either READ or BLAST or to filename:

```
B:a:read filename
      or
A:read b:filename
```

After invoking, the EPROM drivers (READ and BLAST) will display the list of EPROMs that are currently in the EPROM data file and then they will ask for a 'PROM NAME' • The user then types in the type of EPROM he wishes to blast (e.g., 'i2716', for an Intel 2716 type EPROM). The name must be exact and upper and lower case are distinguished. The drivers then ask for a first and last address. The first address is the address of the first byte of the EPROM to read or write. READ will then read until it has read enough bytes to be at 'last', it will write the data to 'filename'. BLAST will next read in the data from 'filename', and then write enough bytes from the data in 'filename' until it reaches 'last'. BLAST always starts with the first byte in 'filename'! After writing an EPROM, BLAST will verify that the write was successful by comparing the EPROM's data to 'filename's data; any discrepancies are displayed to the user as in EPROM.

To invoke EDIT, type:

```
A:edit filename
```

All disk operations are now done on 'filename'. For the basic editing commands see the EPROM EDIT documentation. The 'C' option will change the current working file. Read and Write will read and write from the current file. The Top of Buffer (TOB) will be set to the maximum (X'2000'). Remember that to change or add to the prom data file you must first read in the old data with 'G'. As stated above DEBUG or an assembler can be used in place of edit.

## COPYING EPROMS

Copying an EPROM is similar to the programming process described above. The process involves reading the EPROM to be copied into memory and then, if desired, saving the data on disk. Once it is saved, one or more copies can be made as described above.

Here are the steps:

- 1) Set up the card and install the EPROM to be copied. This is described in detail in the programming sequence above.
- 2) Boot the disk, enter EPROM, load PROM data, and select the correct EPROM driver.
- 3) Select the 'READ EPROM' option. This will read the EPROM into working memory.
- 4) Now, if you want to save the data to a file, select the appropriate option. Reading and saving data is described in detail in the software section of this manual.
- 5) Remove the EPROM as described in the programming instructions above.
- 6) You can now make one or more copies of the data using the normal programming sequence described above.

## HARDWARE NOTES

The A.P.B. board has been designed to blast all of the common 24 pin EPROMs. Most of the pins on these EPROMs are identical. The only differences exist on pins 18, 19, 20, 21, and 24. These pins require various voltages and logic levels depending on the EPROM type. These pins are connected to the personality module socket on the card. Other pins in the personality module socket provide special logic levels and voltages needed by the different EPROM types.

The following special signals are available at the personality module socket:

- |     |  |
|-----|--|
| A11 | Address bit 11, provides most significant bit for 4K EPROMs. |
| A10 | Address bit 10, provides most significant bit for 2K EPROMs. |

PPH High Program Pulse, can be switched between 0, 5, and 26 volts.

PPL Low Level Program Pulse, can be switched between 0, 5, and 12 volts.

TTL1 TTL level pulse, used for chip select, output enable, etc.

-5V Minus 5 volt power supply line.

+12 Plus 12 volt power supply line.

+5V Plus 5 volt power supply line.

#### EPROM ADDRESSING

The programming and reading addresses for the EPROM are provided by a 12 bit binary address register. This takes the EPROM off the computer address bus and allows the programmer to program large EPROMs without interfering with the EPROM space. The register is loaded with a byte address under software control.

The EPROM Blaster uses 4 port based registers to control EPROM programming. The registers are defined as follows:

Port Address	Register
280H+ 4(n)H	Control
281H+ 4(n)H	EPROM Address (Low)
282H+ 4(n)H	EPROM Address (High)
283H+ 4(n)H	EPROM Data
(n = switch setting)	

Control Register: The control register is used to set the logic states and to control the voltages to the personality modules.

#### Control Register

Bit	7	6	5	4	3	2	1	0
Assignment	Not Used	Not Used	TTL 1	Data Reg Enable	PPL D	PPH C	PPH B	PPH A

#### PPH

B A	Voltage
0 0	0
0 1	5 volts
1 0	26 volts
1 1	Undef.

## PPL

D	C	Voltage
0	0	0
0	1	0
1	0	5 volts
1	1	12 volts

Address Registers: Ports 281H+4(n) and 28211+4(n) are used for the EPROM Address Register. Port 28111+4(n) is used for the lower 8 bits of the EPROM address. Port 281H+4(n) is used for the high 4 bits of the EPROM address. Writing to 281H+4(n) and 282H+4(n) will establish the address of the byte to be programmed by the EPROM Blaster.

The Address Register will hold the address information until written to again. The Address Register cannot be read by the CPU.

Data Register: The Data Register holds the data that is being programmed into the EPROM. The Data Register is enabled to the EPROM by setting bit 4 of the control register. The Data Register can be written to by writing to Port 28311+4(n) and can be read only when bit 4 of the control register is set. The Data Register can be read on Ports 28011+4(n) through 28311+4(n).

### Procedure for Writing Data to EPROM:

First the low order address byte should be written to the Address Register. Then the high order address bits are written to the Address Register. The data to be programmed into the EPROM is written to the Data Register. Then bit 4 of the control register is set and the data will be placed on the EPROM data bus. Next, the rest of the bits in the control register are set or reset as necessary to program the byte of data.

### Read Data:

To read data from an EPROM, first the low and high order address bits are written to their respective registers. Bit 4 of the control register is reset to disable the Data Register. Next set the control register so the EPROM is in read mode. Then input the data through Port 28011+4(n).

### Charge Pump:

Most EPROMs require a +26 volt programming voltage to the Programmer. The board must generate this voltage. A charge pump triples the 12 volt supply to provide approximately 36 volts, which is then regulated down to 26 volts.