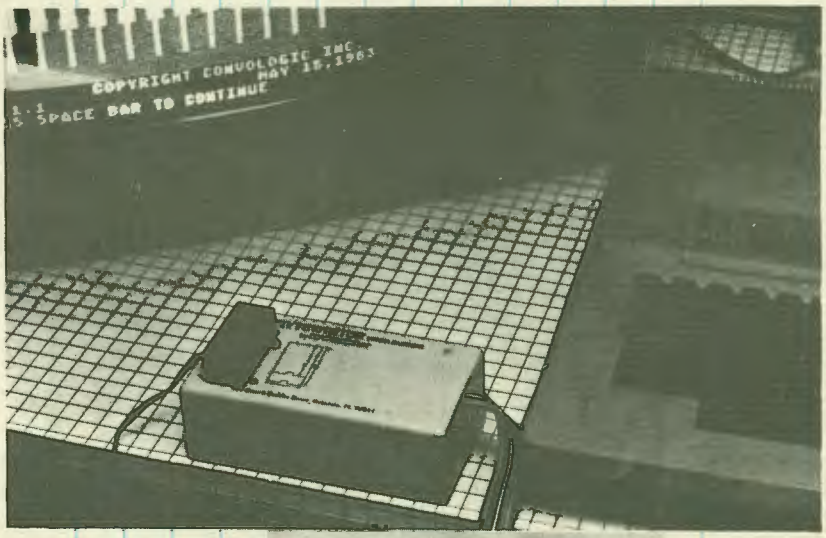
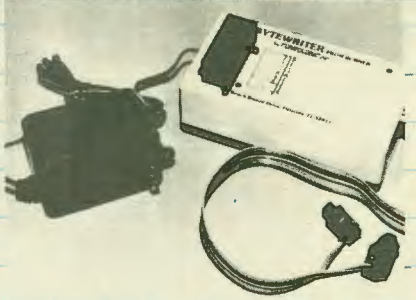
SUPERIOR QUALITY PERIPHERALS





THE BYTEWRITER

by CONVOLOGIC, INC.

INSTRUCTIONS

421 BAY TREE LANE • LONGWOOD

GLOSSARY

K - refers to 1024 bytes of memory,i.e. 8K=8192
$ - refers to hexadecimal numbers
RAM - random access memory; memory which can be changed
ROM - read only memory; memory which cannot be changed
(Carriage Return) - press carriage return key on keyboard

INTRODUCTION

The BYTEWRITER is designed to read, write and verify 2716, 2516, 2732, 2732A, 2532, 2764, 2564, 68764 and 27128 EPROMS. It will also read the ROMS used in the computer operating system, the disk drive and many game cartridges. Using the BYTEREADER option, it will read a cartridge and also produce a autoloading WAM disk or cassette file.

HARDWARE SETUP

The BYTEWRITER has a 28 pin ZIF socket on the top. Do not power up the BYTEWRITER with a EPROM in the ZIF socket. Do not insert or remove EPROMS or ROMS from the ZIF socket until the MAIN MENU is displayed. Lift the socket handle up to insert or remove EPROMS. Put the socket handle down to secure the EPROM in place for reading or writing. To insert 28 pin EPROMS, align the notched end of the EPROM (the end containing pin 1) with the handle of the ZIF socket. To insert a 24 pin EPROM, position the notched end of the EPROM toward the handle of the ZIF socket but place the EPROM in the ZIF with the unused ZIF pin sockets at the handle end. The 24 pin EPROM should have the same notch orientation as the 28 pin EPROM but be placed away from the handle by 2 rows of socket pins. The diagram next to the ZIF socket shows the orientation of the 24 pin

indexing notch in comparison to the 28 pin indexing notch.
The BYTEREADER option for the BYTEWRITER is a circuit card with a cartridge connector on one side and a 28 pin socket on the other side. Place the cartridge firmly down on the cartridge connector and, after the BYTEWRITER has been plugged into the wall socket, insert the pins of the BYTEREADER into the 28 pin ZIF on the BYTEWRITER. Orient the handle of the ZIF socket with the cutout in the BYTEREADER's circuit card. The BYTEWRITER cable has two joystick plugs at the end. One plug has a white rear cap and the other has a black rear cap. The white rear cap is the plug to insert into the high number joystick port. Remember "Put White to the Right".

EXAMPLE: the black capped plug inserted into joystick port 1
the white capped plug inserted into joystick port 2

or

the black capped plug inserted into joystick port 3
the white capped plug inserted into joystick port 4

The BYTEWRITER power supply is plugged into a wall outlet.

PROGRAM MEMORY LAYOUT

EXECUTIVE	$1F00 - $31E7
OPENING DISPLAY	$3B00 - $3FFB
BYTEREAD	$4000 - $4500

To maximize available buffer RAM in a 16K ATARI, the OPENING DISPLAY and the BYTEREAD programs can be sacrificed i.e., you can use this memory for

data.

SETUP FOR DISK

The disk shipped with the BYTEWRITER contains a DOS formated file, BYTEWRIT. The file must be copied onto a disk that already contains a full DOS and a MEM.SAV file. Instructions for formatting a blank disk and installing DOS and MEM.SAV are given in the ATARI DISK OPERATING SYSTEM manual.

A self-loading disk can be built by renaming the DOS file BYTEWRIT to AUTORUN.SYS. On booting, DOS will load the AUTORUN.SYS file and transfer control to the BYTEWRITER software.

LOADING THE PROGRAM

DISK

Remove all cartridges
Boot DOS from disk
Type L (carriage return)
Type BYTEWRIT(carriage return)

The file will load and run displaying an introductory display. Press the space bar to continue.

CASSETTE

Remove all cartridges
Load cassette in recorder
Rewind cassette if necessary
Press PLAY
Turn computer on while depressing START key.
Press the space bar, the tape will read in.
After a successful load, the cassette will stop and an introductory display will appear.

PROGRAM OPERATION

indexing notch in comparison to the 28 pin indexing notch.
The BYTEREADER option for the BYTEWRITER is a circuit card with a cartridge connector on one side and a 28 pin socket on the other aide. Place the cartridge firmly down on the cartridge connector and, after the BYTEWRITER has been plugged into the wall socket, insert the pins of the BYTEREADER into the 28 pin ZIF on the BYTEWRITER. Orient the handle of the ZIF socket with the cutout in the BYTEREADER's circuit card. The BYTEWRITER cable has two joystick plugs at the end. One plug has a white rear cap and the other has a black rear cap. The white rear cap is the plug to insert into the high number joystick port. Remember "Put White to the Right".

EXAMPLE: the black capped plug inserted into joystick port 1
the white capped plug inserted into joystick port 2

or

the black capped plug inserted into joystick port 3
the white capped plug inserted into joystick port 4

The BYTEWRITER power supply is plugged into a wall outlet.

PROGRAM MEMORY LAYOUT

EXECUTIVE	$1F00 - $31E7
OPENING DISPLAY	$3B00 - $3FFB
BYTEREAD	$4000 - $4500

To maximize available buffer RAM in a 16K ATARI, the OPENING DISPLAY and the BYTEREAD programs can be sacrificed i.e., you can use this memory for

data.

SETUP FOR DISK

The disk shipped with the BYTEWRITER contains a DOS formated file, BYTEWRIT.
The file must be copied onto a disk that already contains a full DOS and a MEM.SAV file.
Instructions for formatting a blank disk and installing DOS and MEM.SAV are given in the ATARI DISK OPERATING SYSTEM manual.
A self-loading disk can be built by renaming the DOS file BYTEWRIT to AUTORUN.SYS. On booting, DOS will load the AUTORUN.SYS file and transfer control to the BYTEWRITER software.

LOADING THE PROGRAM

DISK

Remove all cartridges
Boot DOS from disk
Type L (carriage return)
Type BYTEWRIT(carriage return)
The file will load and run displaying an introductory display. Press the space bar to continue.

CASSETTE

Remove all cartridges
Load cassette in recorder
Rewind cassette if necessary
Press PLAY
Turn computer on while depressing START key.
Press the space bar, the tape will read in.
After a successful load, the cassette will stop and an introductory display will appear.

PROGRAM OPERATION

indexing notch in comparison to the 28 pin indexing notch.
The BYTEREADER option for the BYTEWRITER is a circuit card with a cartridge connector on one side and a 28 pin socket on the other side. Place the cartridge firmly down on the cartridge connector and, after the BYTEWRITER has been plugged into the wall socket, insert the pins of the BYTEREADER into the 28 pin ZIF on the BYTEWRITER. Orient the handle of the ZIF socket with the cutout in the BYTEREADER's circuit card. The BYTEWRITER cable has two joystick plugs at the end. One plug has a white rear cap and the other has a black rear cap. The white rear cap is the plug to insert into the high number joystick port. Remember "Put White to the Right".

EXAMPLE: the black capped plug inserted into joystick port 1
the white capped plug inserted into joystick port 2

or

the black capped plug inserted into joystick port 3
the white capped plug inserted into joystick port 4

The BYTEWRITER power supply is plugged into a wall outlet.

PROGRAM MEMORY LAYOUT

EXECUTIVE	$1F00 - $31E7
OPENING DISPLAY	$3B00 - $3FFB
BYTEREAD	$4000 - $4500

To maximize available buffer RAM in a 16K ATARI, the OPENING DISPLAY and the BYTEREAD programs can be sacrificed i.e., you can use this memory for

data.

SETUP FOR DISK

The disk shipped with the BYTEWRITER contains a DOS formated file, BYTEWRIT. The file must be copied onto a disk that already contains a full DOS and a MEM.SAV file. Instructions for formatting a blank disk and installing DOS and MEM.SAV are given in the ATARI DISK OPERATING SYSTEM manual.

A self-loading disk can be built by renaming the DOS file BYTEWRIT to AUTORUN.SYS. On booting, DOS will load the AUTORUN.SYS file and transfer control to the BYTEWRITER software.

LOADING THE PROGRAM

DISK

Remove all cartridges
Boot DOS from disk
Type L (carriage return)
Type BYTEWRIT(carriage return)

The file will load and run displaying an introductory display. Press the space bar to continue.

CASSETTE

Remove all cartridges
Load cassette in recorder
Rewind cassette if necessary
Press PLAY
Turn computer on while depressing START key.
Press the space bar, the tape will read in.
After a successful load, the cassette will stop and an introductory display will appear.

PROGRAM OPERATION

While the introductory display is on the screen, press the space bar to continue to the next menu.

```
TYPE
     1 - BYTEWRITER IN PORTS 1+2
     3 - BYTEWRITER IN PORTS 3+4
```

Type 1 (carriage return) if you have connected the BYTEWRITER to joystick ports 1 and 2. Type 3 (carriage return) if you have connected the BYTEWRITER to joystick ports 3 and 4.

After typing a 1 or 3, the SELECT PROM TYPE menu will be displayed.

```
SELECT PROM TYPE

     0 - NEC 2716*
     1 - NEC 2732*
     2 - FUJITSU 2732A*
     3 - INTEL 2764
     4 - INTEL 27128
     5 - MOTOROLA 68764
     6 - TEXAS INSTR 2516
     7 - TEXAS INSTR 2532
     8 - TEXAS INSTR 2564
     9 - SYSTEM ROM D OR F
     A - SYSTEM ROM E
     B - CARTRIDGE 8K ROM
     C - CARTRIDGE RIGHT 4K ROM
     D - CARTRIDGE LEFT 4K ROM
     E - DISK ROM
     F - CARTRIDGE READER


* - EPROM is INTEL compatible
```

Each of the named prom types represents a pin out configuration for a EPROM or ROM and may be supplied by many different manufacturers. Check the EPROM data sheet for your PROM to determine

which PROM type applies. Only use the CARTRIDGE READER prom type (F in the menu) when creating a WAM file or reading a cartridge into memory via the BYTEREADER option.

After typing in a prom type (carriage return), the MAIN MENU will be displayed. To return to the MAIN MENU from another menu, type Q (carriage return). The MAIN MENU display follows:

```
                                   PROM TYPE
                MAIN MENU
TYPE
     1 - READ A PROM
     2 - VERIFY A PROM AGAINST MEMORY
     3 - VERIFY A PROM IS CLEARED TO FF
     4 - WRITE A PROM
     5 - SELECT A NEW PROM TYPE
     6 - GO TO DOS
     7 - READ/WRITE DISK/CASSETTE
     8 - DISPLAY AND CHANGE MEMORY
     9 - READ A CARTRIDGE
     A - CREATE A WAM FILE
```

An explanation of each MAIN MENU follows.

1 - READ A PROM

From the MAIN MENU type 1 (carriage return). The following will be displayed

```
                                                PROM TYPE
                READ MENU
TYPE
     P - TURN PRINTER ON/OFF
     S - TURN SCREEN OUTPUT ON/OFF
     M1234 - READ PROM TO MEMORY
     SSSS EEEE - READ PROM DATA BETWEEN PROM
ADDRESSES
                    SSSS AND EEEE.
```

Typing P (carriage return) will turn the printer output enable on. Type P (carriage return) again

which PROM type applies. Only use the CARTRIDGE READER prom type (F in the menu) when creating a WAM file or reading a cartridge into memory via the BYTEREADER option.

After typing in a prom type (carriage return), the MAIN MENU will be displayed. To return to the MAIN MENU from another menu, type Q (carriage return). The MAIN MENU display follows:

```
                                        PROM TYPE
                MAIN MENU
TYPE
     1 - READ A PROM
     2 - VERIFY A PROM AGAINST MEMORY
     3 - VERIFY A PROM IS CLEARED TO FF
     4 - WRITE A PROM
     5 - SELECT A NEW PROM TYPE
     6 - GO TO DOS
     7 - READ/WRITE DISK/CASSETTE
     8 - DISPLAY AND CHANGE MEMORY
     9 - READ A CARTRIDGE
     A - CREATE A WAM FILE
```

An explanation of each MAIN MENU follows.

1 - READ A PROM

From the MAIN MENU type 1 (carriage return). The following will be displayed

```
                                        PROM TYPE
                READ MENU
TYPE
     P - TURN PRINTER ON/OFF
     S - TURN SCREEN OUTPUT ON/OFF
     M1234 - READ PROM TO MEMORY
     SSSS EEEE - READ PROM DATA BETWEEN PROM
ADDRESSES
                    SSSS AND EEEE.
```

Typing P (carriage return) will turn the printer output enable on. Type P (carriage return) again

to toggle the printer output enable off. The printer is initialized to the disable condition by the MAIN MENU.

Typing S (carriage return) will toggle the screen output similar to the printer output described above. The screen is initialized by the MAIN MENU to enabled.

Typing M4000 will set the memory pointer to 4000. The memory pointer is necessary to read a EPROM'S contents into memory for subsequent use. If the M function is not performed, then a EPROM read will be displayed to the screen/printer but will not be retained in memory. The memory pointer is disabled after each read and must be re-enabled by a M function if more data is to be read into memory.

Typing 100 13F (carriage return) will cause the contents of the EPROM or ROM to be read. The P,S, and M functions above determine whether the data read is displayed or retained in computer memory.

To return to the MAIN MENU , type Q (carriage return).

2 - VERIFY A PROM AGAINST MEMORY

From the MAIN MENU type 2 (carriage return). The following display will occur:

```
                                        PROM TYPE
                VERIFY MENU
TYPE
     SSSS EEEE MMMM
          SSSS - STARTING ADDRESS OF PROM
          EEEE - ENDING ADDRESS OF PROM
          MMMM - STARTING ADDR OF DATA IN MEMORY
```

Typing O 1FFF 4000 (carriage return) will compare each sequential byte of the EPROM or ROM starting

to toggle the printer output enable off. The printer is initialized to the disable condition by the MAIN MENU.

Typing S (carriage return) will toggle the screen output similar to the printer output described above. The screen is initialized by the MAIN MENU to enabled.

Typing M4000 will set the memory pointer to 4000. The memory pointer is necessary to read a EPROM'S contents into memory for subsequent use. If the M function is not performed, then a EPROM read will be displayed to the screen/printer but will not be retained in memory. The memory pointer is disabled after each read and must be re-enabled by a M function if more data is to be read into memory.

Typing 100 13F (carriage return) will cause the contents of the EPROM or ROM to be read. The P,S, and M functions above determine whether the data read is displayed or retained in computer memory.

To return to the MAIN MENU , type Q (carriage return).

2 - VERIFY A PROM AGAINST MEMORY

From the MAIN MENU type 2 (carriage return). The following display will occur:

```
                                        PROM TYPE
              VERIFY MENU
TYPE
     SSSS EEEE MMMM
          SSSS - STARTING ADDRESS OF PROM
          EEEE - ENDING ADDRESS OF PROM
          MMMM - STARTING ADDR OF DATA IN MEMORY
```

Typing 0 1FFF 4000 (carriage return) will compare each sequential byte of the EPROM or ROM starting

at EPROM address 0 with each sequential byte of memory starting at memory address 4000. The 8K of EPROM of this example will be compared with the 8K of memory and all differences found will be displayed. The display of the differences can be terminated by typing a Q.

3 - VERIFY A PROM IS CLEARED TO FF

From the MAIN MENU, type 3 (carriage return). The followng will be dislayed:

```
                                        PROM TYPE
              VERIFY MENU
TYPE
     SSSS EEEE
          SSSS - STARTING ADDRESS OF PROM
          EEEE - ENDING ADDRESS OF PROM
```

A EPROM is "cleaned" by exposing its quartz window to strong ultra-violet light for 30 minutes. By "cleaning" is meant that all bytes in the EPROM are set to FF's. Every byte must be an FF because the programming of an EPROM consists of setting the ones to zeros in the proper order.

To verify that a 2K EPROM is all FF's before attempting to program it, type 0 7FF (carriage return). The EPROM's contents will be checked to contain FF's. All locations not containing FF's will be displayed.

To check greater capacity EPROMS, specify the correct higher ending address for the EPROM.

To return to the MAIN MENU type Q.

4 - WRITE A PROM

Note: ROMS cannot be written to, only EPROMS can. PROM types above 8 can only be read.

From the MAIN MENU type 4 (carriage return). The screen will clear and then display:

```
                                        PROM TYPE
              WRITE MENU
TYPE
     SSSS EEEE MMMM
        SSSS - STARTING ADDRESS OF PROM
        EEEE - ENDING ADDRESS OF PROM
        MMMM - STARTING ADDR OF DATA IN MEMORY
```

Typing 0 1FFF 4000 will "write" an EPROM from the EPROM's address 0 to 1FFF with data from the computer's memory address 4000 to 5FFF. A single byte can be written as well as the entire EPROM or any number of bytes in between.

During a EPROM "write" the number of 256 byte pages left to be "written" are displayed at the top of the screen. The following display will be presented at the end of a successful EPROM "write".

```
PROM BURN IS VERIFIED
PRESS Q TO RETURN TO MAIN MENU
```

If the EPROM write is unsuccessful, the location of error will be displayed.

```
THIS PROM LOCATION WILL NOT PROGRAM
PROM ADR    PROM DATA    MEMORY DATA
OEF1        84             95

PRESS Q TO RETURN TO THE MAIN MENU
```

5 - SELECT A NEW PROM TYPE

From the MAIN MENU type 5 (carriage return). The screen will clear and display the followng:

```
SELECT PROM TYPE

  0 - NEC 2716*
```



```
  1 - NEC 2732*
  2 - FUJITSU 2732A*
  3 - INTEL 2764
  4 - INTEL 27128
  5 - MOTOROLA 68764
  6 - TEXAS INSTR 2516
  7 - TEXAS INSTR 2532
  8 - TEXAS INSTR 2564
  9 - SYSTEM ROM D OR E
  A - SYSTEM ROM E
  B - CARTRIDGE 8K ROM
  C - CARTRIDGE RIGHT 4K ROM
  D - CARTRIDGE LEFT 4K ROM
  E - DISK ROM
  F - CARTRIDGE READER

* EPROM is INTEL compatible
```

Each of the named PROM types represents a pin out configuration for a EPROM or ROM and may be supplied by many different manufacturers. Check the EPROM data sheet for your PROM to determine which PROM type applies.

After typing in a prom type (carriage return), the MAIN MENU will be displayed.

6 - Go to DOS

From the MAIN MENU type 6 (carriage return). The computer will load DUP.SYS from the disk (replacing the BYTEWRIT program) and will display the DOS menu. All of DOS's functions are available. Binary memory dumps can be saved to a DOS file thru DOS's K command (See DOS manual for more information).

7 - READ/WRITE DISK/CASSETTE

From the MAIN MENU type 7 (carriage return).

The screen will clear and then display:

```
TYPE
     C - CASSETTE I/O
     D - DISK I/O
```

CASSETTE OPERATION

Type C (carriage return) will be followed by the following display:

```
TYPE MEMORY STARTING ADDRESS FOR
    I/O TRANSFER
```

Type in the computer starting memory address where data is to be read into or written out of. After typing the (carriage return),the following will be displayed:

```
TYPE

  R -  READ FROM
  W1234 - WRITE TO LAST ADDRESS 1234
```

Typing W4100 (carriage return) will setup a cassette write from address specified as the memory starting address to $4100.

After a W function, described above, the following display occurs:

```
TYPE

 S - FOR SHORT INTER-RECORD GAPS
 L - FOR LONG INTER-RECORD GAPS
```

Typing S (carriage return) will select a cassette write with short inter-record gaps (similar to BOOT cassette files).

Typing L (carriage return) will select a cassette

write with long inter-record gaps.

After choosing a inter-record gap size, the computer "BEEPS" twice. The 410 recorder should have a cassette inside and be rewound. Pressing the space bar will allow the cassette file to be written to the tape. The cassette motor will stop after a successful write and the MAIN MENU will be displayed.

Typing R (carriage return) will cause the computer to display the following:

```
TYPE

 S - FOR SHORT INTER-RECORD GAPS
 L - FOR LONG INTER-RECORD GAPS
```

Typing S (carriage return) will select a cassette read with short inter-record gaps (similar to BOOT cassette files).

Typing L (carriage return) will select a cassette read with long inter-record gaps.

After choosing an inter-record gap size,the computer will "BEEP". The 410 should have a tape loaded and rewound. Pressing the space bar will allow the cassette file to be loaded into memory starting at the memory pointer specified by the previous M function.

A list of possible error codes and their meaning are listed in APPENDIX A.

DISK OPERATION

Typing D (carriage return) will be followed by the following display:

```
TYPE  DISK FILE NAME
```

EXAMPLE D:THISFILE.ASM

Type in the disk file name to be referenced. Any legal DOS catalog file specification is acceptable. Follow the file name with a (carriage return).

After the file name is typed in the following display occurs:

```
TYPE MEMORY STARTING ADDRESS FOR
    I/O TRANSFER
```

Type in the computer starting memory address where data is to be read into or written out of. After typing (carriage return), the following will be displayed:

```
TYPE
  R - READ FROM
  W1234 - WRITE TO LAST ADDRESS 1234
```

Typing W607F (carriage return) will allow the disk to write the file already specified, in this example, with data from the specified starting memory address to $607F. At the completion of a successful disk write the computer displays the MAIN MENU.

A list of possible error codes and their meaning are listed in APPENDIX A.

A note on DOS file formats - To write a simple DOS compatible file, a 6 byte header must be added by INSPECT AND CHANGE in front of the memory data to be written to the disk file. The first two bytes are FF's. The second two bytes are the starting RAM address that DOS will use to reload the data into memory.
The last two bytes are the ending RAM address for the last byte to be loaded. These above bytes do not have to correspond to the current memory

region the program resides in.
Example: To write a DOS file of the data in RAM from $4000 to $5FFF that I want DOS to reload at $A000 to $BFFF insert the followng patch via INSPECT AND CHANGE:

RAM ADDRESS	NEW DATA
3FFA	FF
3FFB	FF
3FFC	00
3FFD	A0
3FFE	FF
3FFF	BF

After inserting the above patch, use a M3FFA and a W5FFF to write the disk file. This file will later reload from DOS into RAM addresses $A000 to $BFFF.

If the CASSETTE/DISK I/O is later used to reload the above example file into memory, the file will be loaded where the M function specifies and the possibly unwanted 6 byte patch will be the first 6 bytes read in.

8 - DISPLAY AND CHANGE MEMORY

From the MAIN MENU type 8 (carriage return). The screen will clear and then display:

```
TYPE

    P-TURN PRINTER ON/OFF
    I1234 - DISPLAY ADDRESS 1234
       TYPE 56 TO CHANGE CONTENTS TO 56
    L1234 5678 - DISPLAY DATA FROM
       ADDRESSES 1234 to 5678
    M1234 5678 9ABC - MOVE MEMORY
       STARTING AT ADDRESS 1234 AND
       ENDING AT 5678 TO MEMORY
       STARTING ADDRESS 9ABC
```

S12 3456 789A - STORE CONSTANT
12 IN MEMORY FROM ADDRESS
3456 TO ADDRESS 789A

By typing P (carriage return) the printer output will be enabled for the L function but not the I function. There is no hardcopy capability for the I function. Typing P (carriage return) will disable the printer output again.

Typing I1F00 (carriage return) will display the hexadecimal contents of memory location $1F00. A sample display is below:

```
I1F00
 1F00  AD
```

Typing any hexadecimal character or characters (carriage return) will then replace the original contents of the memory location, here shown as 1F00, with the value typed in. The next sequential memory location will then be displayed. To examine a location other than the next sequential one, type I and the new memory address (carriage return).

Typing L1F00 1F03(carriage return) will display the hexadecimal memory contents from $1F00 to $1F03. A sample display follows:

```
      +0  +1  +2  +3  +4  +5  +6  +7
2000  42  03
```

To terminate a display before it is finished,or to return to the MAIN MENU, type Q(carriage return).

Typing M4000 5000 6000 will take the data between address $4000 to $5000 and copy it to addresses $6000 to $7000. This function can be used to copy the system ROMs to memory for examination or saving to disk.

Typing SFF 4000 5000 will store the hexadecimal constant FF from address $4000 to $5000. This function can be used to clear areas of data to $FF before a EPROM write function.

9 - READ A CARTRIDGE

From the MAIN MENU type 9(carriage return). The screen will then display:

```
PLACE CARTRIDGE ON CONNECTOR
 PLACE CONNECTOR IN ZIF SOCKET
  THEN TYPE
   1 - 8K CARTRIDGE
   2 - 16K CARTRIDGE
```

Firmly press the cartridge on the BYTEREADER connector and insert the pins of the BYTEREADER into the 28 pin ZIF on the BYTEWRITER. Orient the handle of the ZIF socket with the cutout in the BYTEREADER's circuit card. Secure the BYTEREADER in the ZIF socket by pressing the ZIF handle down level with the BYTEWRITER's housing. To read a 8K cartridge into memory type 1(carriage return). To read a 16K cartridge into memory type 2(carriage return). Refer to APPENDIX D for the procedure to determine a 8K cartridge from a 16K cartridge.

The following display will appear:

```
TYPE STARTING ADDRESS WHERE
 CARTRIDGE DATA IS TO BE STORED
```

Type in the starting HEX address where the data is to be stored in memory, address 5000 is a suitable choice. A 8K cartridge will occupy 5000 - 6FFF. A 16K cartridge will occupy 5000 - 8FFF. APPENDIX D explains the cartridge memory layout and the important addresses used by the Operating System to startup and initialize the cartridge program.

After the starting address is typed in, the MAIN MENU will appear on the screen when the cartridge data has been read into memory.

A - CREATE A WAM FILE

From the MAIN MENU type A(carriage return). The screen will clear and then display:

```
PLACE CARTRIDGE ON EXAMINER'S
 CONNECTOR
  THEN TYPE
   1 - FOR A 8K CARTRIDGE
   2 - FOR A 16K CARTRIDGE
```

After typing 1(carriage return) or 2(carriage return) depending on the size of the cartridge (refer to APPENDIX D), the following display will appear:

```
TYPE
  C - TO WRITE A CARTRIDGE TO CASSETTE
  D - TO WRITE A CARTRIDGE TO DISK
```

To create a autoboot cassette file of the cartridge, type C(carriage return). The computer will beep twice. The 410 recorder should have a cassette inside, be rewound, and have the PLAY and RECORD buttons depressed. Pressing the space bar
will allow the cartridge file to be written to the tape. The cassette motor will stop after a successful write and the MAIN MENU will be displayed.

To create a autorun DOS disk file of the cartridge, type D(carriage return). The following display will appear:

```
TYPE IN FILE NAME WANTED
   FOR EXAMPLE D2:MYFILE.NAM
```


Type in the disk file name wanted for the cartridge and follow with a (carriage return). The autorun file will be written to the disk. After the file is written, the MAIN MENU will be displayed.

WARRANTY

CONVOLOGIC warrants to the original consumer/purchaser that CONVOLOGIC'S BYTEWRITER (not including the computer programs) shall be free from any defects in material or workmanship for a period of 90 days from the date of purchase. If a defect is discovered during this 90 day warranty period, CONVOLOGIC will repair or replace the BYTEWRITER at CONVOLOGIC'S option, provided the BYTEWRITER and proof of purchase is delivered or mailed, postage prepaid, to CONVOLOGIC INC.

This warranty shall not apply if the BYTERWRITER (1) has been misused or shows signs of excessive wear, or (2) if the purchaser causes or permits the BYTEWRITER to be serviced or modified by anyone other than CONVOLOGIC INC. Any applicable implied warranties, including warranties of merchantability and fitness, are hereby limited to 90 days from the date of purchase. Consequential or incidental damages resulting from a breach of any applicable express or implied warranties are hereby excluded.

NOTICE

All CONVOLOGIC INC. computer programs are distributed on an "as is" basis without warranty of any kind. The entire risk as to the quality and performance of such programs is with the purchaser. Should the programs prove defective following their purchase, the purchaser and not the manufacturer, distributor, or retailer

assumes the entire cost of all necessary servicing or repair.

CONVOLOGIC INC. shall have no liability or responsibility to a purchaser, customer, or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by computer programs or hardware sold through CONVOLOGIC INC. This includes but is not limited to any interruption of service, loss of business or anticipatory profits or consequential damages resulting from the use or operation of such computer programs or hardware.

The provisions of the foregoing warranty are subject to the laws of the state in which the BYTEWRITER is purchased. Such laws may broaden the warranty protection available to the purchaser of the BYTEWRITER.

APPENDIX A

I/O ERROR CODES

CODE	MEANING
8A	Timeout
8C	Serial framing error
8E	Serial frame overrun error
8F	Serial frame checksum error
90	Device done error
A0	Disk drive number error
A2	Disk full
A3	Disk I/O error
A4	File number mismatch
A5	File name error
A7	File locked
A9	Directory full
AA	File not found

APPENDIX B

EPROM PINOUTS

28 pins	24 pins	NEC 2716	NEC 2732	FUJITSU 2732A	INTEL 2764	INTEL 27128
1					VPP	VPP
2					A12	A12
3	1	A7	A7	A7	A7	A7
4	2	A6	A6	A6	A6	A6
5	3	A5	A5	A5	A5	A5
6	4	A4	A4	A4	A4	A4
7	5	A3	A3	A3	A3	A3
8	6	A2	A2	A2	A2	A2
9	7	A1	A1	A1	A1	A1
10	8	A0	A0	A0	A0	A0
11	9	O0	O0	O0	O0	O0
12	10	O1	O1	O1	O1	O1
13	11	O2	O2	O2	O2	O2
14	12	GND	GND	GND	GND	GND
15	13	O3	O3	O3	O3	O3
16	14	O4	O4	O4	O4	O4
17	15	O5	O5	O5	O5	O5
18	16	O6	O6	O6	O6	O6
19	17	O7	O7	O7	O7	O7
20	18	^CE	^CE	^CE	^CE	^CE
21	19	A10	A10	A10	A10	A10
22	20	^OE	VP/^OE	VP/^OE	^OE	^OE
23	21	VPP	A11	A11	A11	A11
24	22	A9	A9	A9	A9	A9
25	23	A8	A8	A8	A8	A8
26	24	VCC	VCC	VCC	NC	A13
27					^PGM	^PGM
28					VCC	VCC

28 pins	24 pins	MOTOR 68764	TEXAS INSTR 2516	TEXAS INSTR 2532	TEXAS INSTR 2564	SYSTEM ROM D or F
1					VPP	
2					^CS1	
3	1	A7	A7	A7	A7	A7
4	2	A6	A6	A6	A6	A6
5	3	A5	A5	A5	A5	A5
6	4	A4	A4	A4	A4	A4
7	5	A3	A3	A3	A3	A3
8	6	A2	A2	A2	A2	A2
9	7	A1	A1	A1	A1	A1
10	8	A0	A0	A0	A0	A0
11	9	O0	O0	O0	O0	O0
12	10	O1	O1	O1	O1	O1
13	11	O2	O2	O2	O2	O2
14	12	GND	GND	GND	GND	GND
15	13	O3	O3	O3	O3	O3
16	14	O4	O4	O4	O4	O4
17	15	O5	O5	O5	O5	O5
18	16	O6	O6	O6	O6	O6
19	17	O7	O7	O7	O7	O7
20	18	A11	PD/PGM	A11	A11	A11
21	19	A10	A10	A10	A10	A10
22	20	^E/VP	^CS	PD/^PG	PD/^PG	^CS
23	21	A12	VPP	VPP	A12	CS
24	22	A9	A9	A9	A9	A9
25	23	A8	A8	A8	A8	A8
26	24	VCC	VCC	VCC	NC	VCC
27					^CS2	
28					VCC	

28 pins	24 pins	SYSTEM ROM E	CART 8K ROM	CART RIGHT 4K ROM	CART LEFT 4K ROM	DISK ROM
1						
2						
3	1	A7	A7	A7	A7	A7
4	2	A6	A6	A6	A6	A6
5	3	A5	A5	A5	A5	A5
6	4	A4	A4	A4	A4	A4
7	5	A3	A3	A3	A3	A3
8	6	A2	A2	A2	A2	A2
9	7	A1	A1	A1	A1	A1
10	8	A0	A0	A0	A0	A0
11	9	O0	O0	O0	O0	O0
12	10	O1	O1	O1	O1	O1
13	11	O2	O2	O2	O2	O2
14	12	GND	GND	GND	GND	GND
15	13	O3	O3	O3	O3	O3
16	14	O4	O4	O4	O4	O4
17	15	O5	O5	O5	O5	O5
18	16	O6	O6	O6	O6	O6
19	17	O7	O7	O7	O7	O7
20	18	A11	A11	A11	A11	CS
21	19	A10	A10	A10	A10	A10
22	20	^CS	^CS	^CS	^CS	CS
23	21	^CS	A12	CS	^CS	CS
24	22	A9	A9	A9	A9	A9
25	23	A8	A8	A8	A8	A8
26	24	VCC	VCC	VCC	VCC	VCC
27						
28						

Legend:

^	Negative logic(low = enabled)
A0 - A13	Address bits
O0 - O7	Data output bits
GND	Ground
VCC	+5 volts
VPP	Programming voltage(25 or 21 volts)
VP	Same as VPP
CE	Chip enable
OE	Output enable
PGM	Programming enable
PG	Same as PGM
E	Chip enable
CS	Chip select
PD	Power down
CS1	Chip select 1
CS2	Chip select 2
NC	No connection to this pin

APPENDIX C
PROCEDURE FOR THE CONSTRUCTION OF A CARTRIDGE

The 2764 EPROM board has two sockets.The sockets are on the component side of the board. Looking at the component side with the edge connector down,the left socket will contain the program code from $8000 to $9FFF. The right socket will contain the program code from $A000 to $BFFF. A 8K cartridge program is contained only in the right socket. A 16K cartridge program is contained in both sockets.

To construct a 16K cartridge, place your cartridge image in computer memory in a 16K block, for this example, from $4000 to $7FFF. Using the BYTEWRITER's MAIN MENU select a PROM TYPE of the INTEL 2764 (number 5 of the MENU) and then select item 4 - WRITE A PROM. The screen will display

PR
OM TYPE
WRITE MENU
TYPE
SSSS EEEE MMMM
SSSS - STARTING ADDRESS OF PROM
EEEE - ENDING ADDRESS OF PROM
MMMM - STARTING ADDR OF DATA IN MEMORY

Type 0 1FFF 4000 (carriage return)

The screen will display the number of 128 byte groups to be burned. When finished,the display will indicate
PROM BURN IS VERIFIED
Press space bar and the MAIN MENU will be displayed.Remove the 2764 EPROM that was burned and insert this EPROM into the left socket of the EPROM board. Left and right sockets are determined by placing the EPROM board on a surface with the socket side up and the gold edge

connector closest to you. In the correct position the sockets will have a small half-circle indentation away from you on the top edge (between pins 1 and 28).To insert the EPROM into the socket, align the half-circle indentation of the socket and the EPROM. Place one row of the EPROM's legs in the spring loaded holes and compress the other row of legs toward the first row until they are aligned with the socket holes. Press down firmly. The EPROM will seat firmly with all legs gripped by the springs of the socket holes.
After the left socket is in place, place a clean 2764 EPROM in the BYTEWRITER's ZIF. Secure the EPROM by pressing down the ZIF handle. Repeat the steps at the keyboard to WRITE A PROM. This time the addresses to be burned are
0 1FFF 6000
because the upper 8K of the 16K program is to be programmed into the second 2764. When the burn is complete, remove the EPROM and place it into the EPROM board's right socket.
To test the board, open the cartridge door of the computer and place the board into the left cartridge slot of the computer.
NOTE: the component side of the cartridge board should face the rear of the computer, not the front. If you can clearly see the EPROMs while seated before your computer, then the board is in wrong - turn the board around before applying power.
After the cartridge board's position in the cartridge slot has been verified, close the cartridge door. Apply power to the computer and your cartridge program should operate.
To build a 8K cartridge, only program one 2764 EPROM and place it into the right EPROM board socket. To allow a 8K memory deselect instead of a 16K deselect, place the EPROM board on a flat surface with the component side face down. Cut the right most edge connector trace just above the point where the "A" edge connector meets a

circuit land. This land runs from the edge connector marked "A"for about 3/8 of an inch towards the top of the board and ends at a feed through. Any knife cut through this trace will cause the computer to only deselect 8K of memory. CAUTION - do not cut this trace for 16K programs. To place the EPROM board in the shell, place the half shell with the 4 male pins on a flat surface and place the EPROM board, component side up, in the half shell with the two lower male pins passing through the aligning holes in the board. These two holes are closest to the board's gold edge connector. The board should slide down the male pins and the board's center hole should rest on the center pin of the half shell. Place the other half shell on the four male pins and squeeze the two half shells together. Insert and tighten the metal screw in the cases screw hole.

APPENDIX D

ELEMENTARY CARTRIDGE INFORMATION

Cartridges come in 8K and 16K varieties. To determine an 8K from a 16K, depress the edge connector guard if one exists and count the gold traces on the card edge. An 8K cartridge has 12 traces on the front of the card and 13 traces on the back of the card. A 16K cartridge has 13 traces on the front of the card and 14 traces on the back of the card.

ADDRESS	FUNCTION
$9FFA,$9FFB	lower cartridge run address
$9FFC	lower cartridge available byte
$9FFD	lower cartridge option flag
$9FFE,$9FFF	lower cartridge initialization address
$BFFA,$BFFB	upper cartridge run address
$BFFC	upper cartridge available byte
$BFFD	upper cartridge option flag
$BFFE,$BFFF	upper cartridge initialization address

HOW TO RUN A CARTRIDGE

16K CARTRIDGES

Examine the upper cartridge option flag, if bit 7 is set (i.e. number is $80 or greater) then the cartridge starts execution at the address specified in the cartridge initialization address.
Examine the lower cartridge available byte, if the byte equals 0 then an initialization subroutine call is executed at the address specified in the lower cartridge initialization address. Examine the upper cartridge available byte. If the byte equals 0 then an

initialization subroutine call is executed at the address specified in the upper cartridge initialization address.
All of the cartridges analyzed to date have started execution at the address specified in the upper cartridge initialization address with no subroutine calls needed. However future cartridges will probably use all possible address combinations explained herein just to confuse the analyst.
Examine bit 0 (the number will be odd if bit 0 set) in the upper and lower cartridge option flag. If either bit is set (=1) then a DOS boot is required. A DOS boot will only be required on programs that use the disk like BASICS, ASSEMBLER/EDITORS, WORD PROCESSORS etc. A DOS boot can best be handled by a subroutine call to the OS program of your computer. The particular location to call is dependent on which computer and which OS ROM version you have. For OS ROM A and ROM B in the 400/800 machine, a correct address to call as a subroutine is $F2CF. (See example 2)
Examine the upper cartridge available byte, if the byte equals 0 then examine the upper cartridge option flag. If bit 2 is set (i.e. has some version of 4 through 7 as its lower nibble) then the cartridge program's starting address is contained in the upper cartridge run address.
If the upper cartridge available byte is not equal to 0 then examine the lower cartridge available byte. If this byte equals 0, then examine the lower cartridge option flag. If bit 2 is set then the cartridge program's starting address is contained in the lower cartridge run address.
If neither of the above conditions are true, the starting address for the program is contained in addresses $A,$B in memory.

8K CARTRIDGES

Follow the instructions for a 16K cartridge but omit all references to the lower cartridge items because nothing below address $A000 is valid in a 8K cartridge.

EXAMPLES

Example 1:

```
$BFFC=0
$BFFD=$FF
$BFFE=$15
$BFFF=$90
```

The upper cartridge option flag ($BFFD) has bit 7 set so the starting address of the cartridge is contained in the upper cartridge initialization address ($BFFE,$BFFF) and equals $9015.To create a self-starting DOS file of this cartridge image,use the DOS K command from the DOS MENU.For example

```
K(carriage return)

  PROGRAM,8000,BFFF,,9015(carriage return)
```

The above example assumes the cartridge image was loaded from $8000 to $BFFF in memory.

Example 2:

```
$BFFA=$25
$BFFB=$9F
$BFFC=0
$BFFD=5
$BFFE=00
$BFFF=$A0
```

The upper cartridge option flag does not have bit 7 set. The upper cartridge available byte equals 0. Therefore an initialization call to $A000 must be made. The upper cartridge option flag

does have bit 0 set so a subroutine call for a DOS boot must be made. The upper cartridge option flag does have bit 2 set indicating the cartridge run address is $9F25. A series of instructions will be needed to satisfy the above conditions:

ASSEMBLY LANGUAGE	MACHINE CODE
JSR $A000	20 00 A0
JSR $F2CF	20 CF F2
JMP $9F25	4C 25 9F

Using Inspect and Change (MAIN MENU item 6) insert the machine code instructions in memory.

For example:
$7000 20
$7001 00
$7002 A0
$7003 20
$7004 CF
$7005 F2
$7006 4C
$7007 25
$7008 9F

then go to DOS (MAIN MENU item 5) and store above instructions as a load file. When in the DOS MENU

K(carriage return)

FILENAME,7000,7008,,7000 (carriage return)

This will create a DOS file that will load into $7000 and start itself at $7000.
Create the cartridge image file by typing the following:

K(carriage return)

IMAGE,8000,BFFF(carriage return)

The above example assumes the cartridge image was

loaded from $8000 to $BFFF in memory.
Using DOS, first load the cartridge image then load FILENAME. The Program will then start executing at $7000 and will run the cartridge program.

loaded from $8000 to $BFFF in memory.
Using DOS, first load the cartridge image then load FILENAME. The Program will then start executing at $7000 and will run the cartridge program.

APPENDIX E

BOOT CASSETTE FORMAT

A bootable cassette has the following specified bytes at the beginning of the cassette.

Explanation	Length
00	1 byte
number of 128 byte files in record	1 byte
starting address of file load area	2 bytes
initialization address of file	2 bytes
LDA #$3C	2 bytes
STA $D302	3 bytes
JP START	3 bytes

The first byte of the boot header shown above is always 0.

The second byte (number of 128 byte files in the record) is calculated by taking the last address to write to the tape and subtracting the starting address to write to the tape. Add 127 to this difference and divide the answer by 128. For example, to calculate the number of 128 byte files for a tape write from $8000 to $BFFF setup the following equation:

((\$BFFF - \$8000) + 127) / 128.

\$BFFF - \$8000 = 16383

16383 + 127 = 16510

APPENDIX E

BOOT CASSETTE FORMAT

A bootable cassette has the following specified bytes at the beginning of the cassette.

Explanation	Length
00	1 byte
number of 128 byte files in record	1 byte
starting address of file load area	2 bytes
initialization address of file	2 bytes
LDA #$3C	2 bytes
STA $D302	3 bytes
JP START	3 bytes

The first byte of the boot header shown above is always 0.

The second byte (number of 128 byte files in the record) is calculated by taking the last address to write to the tape and subtracting the starting address to write to the tape. Add 127 to this difference and divide the answer by 128. For example, to calculate the number of 128 byte files for a tape write from $8000 to $BFFF setup the following equation:

(($BFFF - $8000) + 127) / 128.

$BFFF - $8000 = 16383

16383 + 127 = 16510

16510/128 = 128.98

Drop the decimal part of the number so 128.98 = 128. Therefore the number of 128 byte files = 128 for a 16K memory write to tape. Consequently for a 8K program memory the number of 128 byte files = 64. Note that 128 in hexadecimal = $80 and 64 in hexadecimal = $40.

The third and fourth bytes of the boot header are the 2 byte starting address of where the file loads into memory. In the above example the starting address is ($8000 - $E) = $7FF2. Therefore byte 3 will equal $F2 and byte 4 will contain $7F.

Bytes 5 and 6 of the boot header are the 2 byte initialization address. These 2 bytes are 0 and 0 for a single stage boot load file format.

Bytes 7 thru 11 are always the following:

```
$A9
$3C
$8D
$02
$D3
```

These bytes turn off the cassette motor after a successful load.

Bytes 12 thru 14 are a jump to the starting address of the program obviously somewhere between the start and end addresses that are to be written to tape. If the starting address is $B123 then these last 3 bytes would be:

```
4C
23
B1
```

Using the Inapect and Change function (6 in the

MAIN MENU) insert these bytes in front of the area of memory to be written to tape.

Example: The memory area of $A000 to $BFFF (8K) is to be written to tape in a boot formatted file on cassette. The starting address to run the program is $A1B4. Using Inspect and Change the following bytes are first placed into memory.

$9FF2	00
9FF3	40
9FF4	F2
9FF5	9F
9FF6	00
9FF7	00
9FF8	A9
9FF9	3C
9FFA	8D
9FFB	02
9FFC	D3
9FFD	4C
9FFE	B4
9FFF	A1

Then using CASSETTE/DISK I/O (7 in the MAIN MENU),specify a M9FF2 and a WBFFF with short inter-record gaps. This will create a boot file format cassette tape of memory from $A000 to $BFFF with an autostart address of $A1B4.