EPROM PROGRAMMER - EMULATOR

Type: EP8000

Instruction Manual

INTRODUCTION

The EP8000 has been designed as a flexible system for programming and emulating all the current NMOS EPROMs It can be used as a straight forward programmer with ease of operator use, to give PASS/FAIL information for programming , verify and checksum.
The programming power can be expanded with the range of programming modules

At the same time, the machines emulation capability, editing facilities and input/output makes this an ideal system for aiding in the development of programs in a high level language, or assembly language on a larger system. Then downloading the code to the EP8000 and using the emulation mode to debug the final stages. Alternatively the program can be developed in hex, entirely on the machine and run from prototype hardware. In both cases when the program has been developed, it can be transferred to an EPROM for use by the outside system, independent of the EP8000

ABOUT THE MANUAL

The manual has been written as a guide to the use of the EP8000 EPROM Emulator Programmer. All aspects of the machine have been covered in detail, to allow anybody unfamiliar with EPROM programmers to be able to use it in the field.
Section 1 gives details on setting up the machine for use along with operating, care and maintenance instructions.
Section 2 gives a listing of status codes and a brief outline of the keyfunctions, controls and display.
Section 3 describes all the key functions along with examples of use.
Section 4 describes the operation of the cassette interface - setting up and using the port.
Section 5 gives details of the 8 bit handshake parallel port available.
Section 6 shows the method by which serial data transfers can be made, using any one of the 3 serial lines.
Section 7 details aspects of EPROM Emulation, giving 3 design examples to illustrate the use of this facility.
Section 8 gives general information on EPROM Programming and erasing and gives examples of the use of the programming facility, including block programming.

Appendix A gives brief details of the accessories available for the machine - please note that some of these are not yet in production, as indicated on the current price list.
Further appendices will be sent in the future to give users information on software and any further accessories which become available.

The design and specification of the machine are subject to continuous development, and improvement, consequently the machine may incorporate changes in detail from the information contained in the manual.
Any amendments will be issued at regular intervals between manual reprints.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont)

SECTION 1

OPERATING INSTRUCTIONS

1.00 GENERAL

Before switching the machine on, it is generally recommended that sections 1, 2, and 3 be read right through, and that some time be spent on becoming familiar with the controls.

1.01 SUPPLY VOLTAGE

Machines supplied in the UK are ready to operate from the 240v, 50Hz supply. A ready wired cable, complete with fused plug is supplied. The cores of the cable are colour coded as follows:

Line	Brown	Neutral	Blue	Earth	Green/Yellow

Note: If the case is to be opened for any reason - disconnect the unit from the mains supply before doing so.

1.02 OPERATING PRECAUTIONS

To ensure continued troble free operation, please note the following points.

1/ Operate the machine on a vibration free surface.
2/ Do not operate the machine in direct sunlight, or near any source of heat.
3/ Ensure the ventilation holes are not covered, and that external air flow is not restricted.
4/ Ensure no metal parts can fall into the machine - E.g. through the keypad aperture.
5/ Never switch the machine on or off with a device in the zero insertion force programming socket. This could destroy the device and possibly damage the machine.
6/ Always check the device selection , and key 'RESET' when inserting or removing EPROMs from the programming socket.
7/ Disconnect the unit from the mains supply when not in use.

1.03 CARE OF THE MACHINE

Treat the machine with the respect it, and any other item of equipment deserves. Observe the following points, to keep the machine in good condition.

1/ Periodically clean the ZIF with a stiff bristle brush, to ensure good contact and trouble free operation.
2/ Never force a device into or out of the ZIF - the socket is a zero insertion force device.

3/ If the outer surfaces require cleaning then use a damp cloth. Do not use volatile solvents - these will damage the plastic parts and the surface finish.

4/ Cover the machine when it is not in use, to prevent the ingress of dust and metal particles. (Disconnect from mains supply when not in use).

1.04 SETTING UP AND SWITCHING ON

Connect a TV or video monitor to the EP8000 (See section 2 for connection data). Connect the machine to the mains supply and switch on. The LED display will show C1 3000 FF to indicate mode C1, and that the cursor is at address 3000, and data at that address is FF.
Address 3000 is the start of the main RAM, and the entire RAM has been cleared to FF (erased state of EPROM).
Adjust the TV picture with the height, width and position controls to view all the information. No adjustment is required if you are using the Prinz video monitor.
The ZIF LED is off to indicate that the socket is powered down, and ready to receive an EPROM. Check the device type setting before inserting an EPROM into the socket.
The machine is now ready for use.

1.05 SERVICING AND MAINTENANCE

If the machine needs returning to us for repair or servicing under service contract or warranty, please ensure that:

1/ The machine is well packed for transit. Enclose it in a polythene cover and surround with soft packing material inside a strong cardboard box.

2/ Enclose a brief description of the fault.

3/ Enclose full name and address.

4/ State where and when purchased.

SECTION 2

2.00 Key labels & Status codes

The listing below explains the many status codes produced by the EP8000 and the abbreviations used for the keys. The list should be used as a guide when using the machine. The keyfunctions are described in detail in section 3.

	KEY	MEANING	Status code	Comments
	FN	function	FN	used to select uppercase keyfunctions
	TO	TO	TO	data relocation. object/target definition
	DMA	Direct memory access	--	EPROM or RAM emulation facility
	RST	reset	C1	halts internal microprocessor
*	PAD1	scratchpad 1	P1	define memory area 5C00-5FFF
	BLCK	block	bL	block memory area
*	DISP	displacement	dP	displacement from previous keypress of cursor
	MEM	Memory address	Ad	define 4 digit hex address
*	CLR	clear	CL	clear target to FF
	↑	Move cursor	--	move cursor up (16 address move)
*	PRINT	print	--	dump object to printer
	←	Move cursor	--	move cursor back
	DEF	define	dF	define (block) mode
	CSR	Cursor	C1	define on-screen cursor
*	SHIFT	shift	SH	shift (block) mode
	↓	move cursor	--	move cursor down (16 address move)
*	MTCH	match	HL	highlight (match) on-screen byte
*	S IN	serial in	Si(FL)	prepare to receive data
*	S OUT	serial out	(FL)	dump data via serial port
*	P IN	parallel in	Pi	receive data from parallel port
*	P OUT	parallel out	--	dump data to parallel port
*	C IN	cassette in	Ci(FL)	receive data from cassette
*	C OUT	cassette out	Co(FL)	dump data to cassette
*	RAM	Ram	rA	define RAM area 3000-4FFF
*	PROM	EPROM (in ZIF)	Pr	define EPROM area in accordance with device selection
*	EXEC	Execute	EC	execute (target) function
*	PROG	program	(PASS/FAIL)	program EPROM with RAM data
*	VFY	verify	(PASS/FAIL)	compare EPROM and RAM data
*	CHCK	checksum	CS	calulate EPROM checksum
*	STOR	store	(PASS/FAIL)	store EPROM data into RAM, then verify
	→	move cursor	--	move cursor forward
*	PAGE	page	PG	selects 256 bytes for viewing or PG Mode
*	INST	insert	--	inserts FF at cursor location
*	DEL	delete	--	deletes byte at cursor
*	DEV	device select	--(DE)	displays current device and allows new device to be selected

Upper case keys are labelled by *.
Hex keys are lower case labelled 0-9, A-F, used for direct entry.

2.01 SPECIAL STATUS CODES

Some special codes are generated during the course of operation, usually when the machine requires further information - these are:

Staus code	Meaning
Ad	request for 4 digit hex address for new C1 position
Pg	request for 2 digit page number. The page number is the first 2 digits of the address of the first byte to appear on screen
Fn	next key entry will be upper case
HL	enter 2 digits for highlighting byte (match function)
FL	request for input/output code. May be 2 or 4 digits.
no	indicates key sequence could corrupt internal data, and control sections, and is therefore not allowed
PASS	indicates good stores, verify, program or load
FAIL	indicates bad store, program, verify or load

A blank display indicates the machine is busy or in the EPROM emulation mode. When busy, the machine does not respond to key strokes except RESET (RST).

2.02 FAULT CODES

Certain operations on the EP8000 lead to the machine entering a FAULT mode. When this happens, the ZIF will power down and the display will flash 'FAULT X' where X is the fault code. To leave this mode, Key RESET (RST).

Fault Code	Fault
P	During programming, the machine checks that bytes set up to be programmed are correctly set-up. Fault P indicates a data pin fault. This would usually be caused by a short circuit within the EPROM.
I	Indicates that the device in the ZIF socket is not correctly inserted - either misplaced or reversed.
B	Indicates that the illegal bit check has found that the device is not programmable with the required data.

2.03 SCREEN STATUS CODES

Codes shown on the screen under the heading 'STATUS' provide guidence on the machines current operation.

CODE	MEANING
EDITING	The normal mode of operation
PROGRAM	The machine is carrying out a programming cycle
VERIFY	The machine is comparing the device in the ZIF with the corresponding RAM area
PASS/FAIL	Indicates a successful/unsuccessful check of data

CODE	MEANING
STORE	Indicates that the machine is copying data from the PROM into the RAM
SERIAL IN SERIAL OUT	The machine is engaged in a serial data transfer
PARALLEL IN PARALLEL OUT	The machine is engaged in a parallel data transfer
PRINT	Shows the machine is communicating with a printer
EXECUTING	Shows that the machine is executing a user originated program
EMULATING	The machine is in DMA or Emulation mode

Display and Connectors

2.04 PROGRAMMING SOCKET

The zero insertion force (ZIF) socket is a 28 pin lever operated device designed for the rapid and easy insertion of EPROMs without any damage. The position of pin 1 is shown on the panel with an appropriate diagram. See drawing below. Always ensure the EPROM is correctly orientated, and correctly seated before the lever is closed (flush with ZIF) otherwise the device, and the EP8000 may be damaged when the ZIF is powered up. When power is applied to the ZIF, the LED will light.

The socket can be powered down at any time by keying RESET. When using the socket to read or program EPROMs, always adopt the following procedure.

Device insertion

1/ Power down the socket by keying RESET

2/ Set the device selection to the required device type.

3/ Insert device and close the lever - (flush with ZIF and panel) - check the device is in the correct way round, not misplaced, and properly seated

4/ The socket will power automatically when any key is pressed

Device removal

1/ Power down the socket by keying RESET

2/ Remove the device

2.05 LED DISPLAY

The 8 digit seven segment green LED display gives a constant readout of information to the user. The display is divided into 3 sections -

1/ Status (2 digits)

Gives information regarding keypress and mode. A complete list of status codes is given in 2.00

2/ Address (4 digits)

Usually displays the address at which cursor 1 is pointing.

Also used for information request entries under the FL status code and displays PASS or FAIL on completion of a store, program, verify or data load from the input ports.

3/ Data (2 digits)

Usually displays the data at cursor 1 address as a pair of hex digits.

The 'NO' status code will appear in this section of the display.

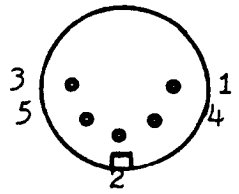
2.06 VIDEO DISPLAY

Two video outputs are provided

1/ Modulated video for connection to a T.V. receiver. The socket is an 'aerial' type, located at the side of the machine. Tune the T.V. to channel 35 for a clear sharp display.

2/ Composite video for connection to a video monitor. The socket is a DIN type located at the rear right of the machine (as viewed from machine front). The centre pin is 0v and the video signal is available from any other pin (0.5v-3.5v pp). A lot of information is presented by the EP8000 video, and the height, width, and position controls should be used to view it all. (If you are using the Prinz monitor then no adjustment should be necessary.)

5 pin DIN VIDEO SOCKET (pin connections)



Socket view

(pin numbers are DIN standard)

pin 1 - video out
pin 2 - 0v
pin 3 - video out
pin 4 - video out
pin 5 - video out
shield - 0v

The information presented on the screen consists of a band of user information across the top of the screen, with the rest of the screen being filled with hexadecimal data.

The user information consists of the following headings:

ADDRESS	-	The hexadecimal address of the cursor
DATA	-	The contents of the cursor
DEVICE	-	The device currently selected

STATUS - The machines operational status - the codes are explained in section 2.03

DISPLACEMENT:

FIX - The position of the cursor when the DISP key was pressed

FWD - The displacement from the cursor to the fixed point

BACK - The displacement from the fixed point to the cursor.

The hexadecimal information displayed is divided into three areas. The first being 64 bytes of information from the previous page. The main highlighted area consists of 256 bytes of information from the current page. This is the active area in which the cursor operates. The third area is another 64 byte area, the information being from the page following the active page.

The display is effectively a moveable window, which can be used to view any part of the memory. With the exception of the video RAM , this information is divided into 4K fields. This means that when page 20 is selected for instance, the information shown from the previous page at the top of the screen will be from page 1F. However, if the first page of RAM (PG 30) is selected no information is shown at the top of the screen. Additionally, when page 4F is selected no information is shown at the bottom of the screen.

2.07 EMULATION PLUG

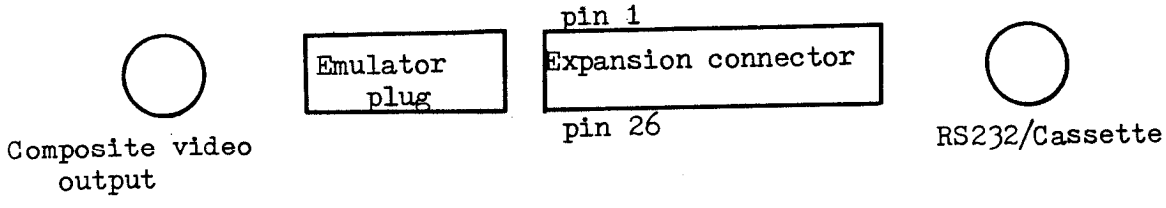
This is the 26 pin plug at the back of the machine, used for EPROM emulation. Input loading is 1 LS TTL load, and output leakage current is 10uA. The plug is only active when in the DMA mode. In order to achieve maximum speed of access time, no protection is provided on the plug, so care must be taken to ensure that the simulator cable is correctly inserted into the external host system. Take special care to plug the cable the correct way round.
It is recommended that you use the BSC8 cable when emulating with the EP8000.

2.08 EXPANSION CONNECTOR

This is the 50 way plug at the rear of the machine. It gives direct access to the machine buses, the parallel port, and the TTL and 20mA serial input/ output lines.

Pin connections:

Connections viewed from rear of the machine



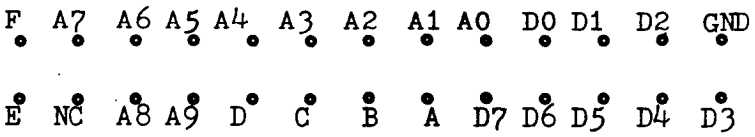
Pin connections: Expansion connector

Pin number	Name	Pin number	Name
1	A7	26	Vcc
2	A6	27	A8
3	A5	28	A9
4	A4	29	Vcc
5	A3	30	NMREQ
6	A2	31	A10
7	D7	32	A1
8	D6	33	A0
9	D5	34	D0
10	D4	35	D1
11	D3	36	D2
12	0v	37	SIM R/W
13	SIM OE	38	A11
14	NWDS	39	NRDS
15	PB7	40	SIM DIR
16	A15	41	A14
17	A13	42	A12
18	PB6	43	CDMA
19	PA7	44	PA0
20	PA6	45	PA1
21	PA5	46	PA2
22	PA4	47	PA3
23	---	48	---
24	TTL serial O/P	49	---
25	TTL serial I/P	50	---

KEY

Pin	Description	Input/Output
D0-7	Data bus	in/out
A0-A15	Address bus	out
PA0-PA7	Parallel port	in/out
PB6-PB7	Parallel handshake control	in/out
NWDS	Write strobe active low	out
NRDS	Read strobe active low	out
NMREQ	Memory request active low	out
CDMA	DMA control	out
SIM OE	Simulation RAM O/P enable	in
SIM R/W	Simulation RAM write enable	in
SIM DIR	Simulation DATA buffer direction control	in
Vcc	+5v supply	out
0v	zero volts	out
NC	not connected	

Pin connection View from machine rear



Key: A0-A9 address lines from host system. D0-D7 data bus

NC not connected Inputs A,B,C,D,E,f depends on selected device as shown on table overleaf.

CONFIGURATION OF EPROM PINS

Device Type	18 A	19 B	20 C	21 D	2 E	27 F
2704	Lo	+12V	$\overline{CS}$	-5V	NC	NC
2708	Lo	+12V	$\overline{CS}$	-5V	NC	NC
2716(3)	$\overline{CS}$	+12V	A10	-5V	NC	NC
2508	$\overline{CS}$	A10	$\overline{OE}$	Hi	NC	NC
2758A	$\overline{CS}$	Lo	$\overline{OE}$	Hi	NC	NC
2758B	$\overline{CS}$	Hi	$\overline{OE}$	Hi	NC	NC
2516/2716	$\overline{CS}$	A10	$\overline{OE}$	Hi	NC	NC
48016	$\overline{CS}$	A10	$\overline{OE}$	Hi	NC	NC
2532	A11	A10	$\overline{OE}$	Hi	NC	NC
2732/2732A	$\overline{CS}$	A10	$\overline{OE}$	A11	NC	NC
68732-0	A11	A10	$\overline{OE}$	Lo	NC	NC
68732-1	A11	A10	$\overline{OE}$	Hi	NC	NC
68764/68766	A11	A10	$\overline{OE}$	A12	NC	NC
2564	A11	A10	$\overline{OE}$	A12	$\overline{CS}1$	$\overline{CS}2$
2764	$\overline{CS}$	A10	$\overline{OE}$	A11	A12	CS

Pins A,B,C,D,E, and F are automatically configured by the EP8000 for the correct device type. The -5v and +12v for 3 rail devices are supplied from the host system, not the EP8000. These lines are used as reference levels by the EP8000.

CAUTION:

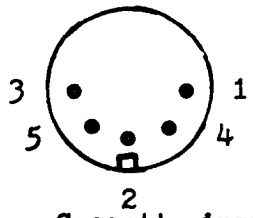
1/ Ensure the simulator cable is properly connected at both ends, ensuring correct orientation.

2/ Do not short any pin at any time.- especially the data pins - short circuit current will destroy the line driver.

2.09 RS232/ CASSETTE SOCKET

This is the 5 pin socket located at the rear right of the machine (as viewed from rear) close to the 50 way plug.

Connection data



Socket view
(pin numbers are DIN standard)

pin 1 - Cassette input
pin 2 - 0v
pin 3 - Cassette output
pin 4 - RS232 output (-5v, +5v)
pin 5 - RS232 input
shield - 0v

See 'serial data transfers' and 'cassette interface' for details

SECTION 3

3.00 Keyfunctions

This section gives a detailed description of the EP8000 keyfunctions, taken one key at a time. Examples are given in the use of each key by itself, and in conjunction with other keys.

3.01 Reset (RST)

Stops the internal microprocessor from executing the control program for as long as the key is held. The programming socket is powered down as indicated by the ZIF LED. All functions are stopped, including the emulation facility, Releasing the key starts the machine in normal mode, as indicated by C1 in the status window. The screen is rewritten clearing any highlights, but C1 remains at the same address. Any function modes such as block define, shift, program, input/output are stopped and cleared to C1 mode. RAM data is not affected. The serial output line will go to the mark condition (logic 1).
NOTE: If the ZIF contains an EPROM and reset is keyed, the socket will power down, so if you are viewing any page in the range 60-7F, because the socket is effectively empty (ZIF powered down), hex FF will be shown in all screen locations.
Powering the ZIF (by depressing any key) will have no effect, unless the page is rewritten by calling page 6X or 7X or by FN FN. (X = any hex key to define page number).

3.02 Hexadecimal Keys 0123456789ABCDEF

These are lower case keys for entering machine code at the C1 position shown on screen. The address of C1 is shown in the LED display.
These keys are also used for data entry when requested by the machine.
For example: Page selection, cassette file number, in/out speed and port number and cursor 1 address selection.
In normal mode when a 2 digit hex entry has been made at a particular C1 address, the cursor will automatically increment to the next address.

3.03 Function (FN)

This key is similar to shift or uppercase on a typewriter or terminal, and allows most of the keys to have 2 functions (except RST, TO and DMA).
Example: Key FN 9 will select 'RAM' as the keyfunction and this should be thought of as function RAM rather than FN 9 so that key sequences flow in a logical manner.

Double keying the function key i.e. FN FN is a function in itself which will rewrite the screen and clear any highlights. This function is similar to RESET except that the programming socket is not powered down.

3.04 Cursor Control Keys

These are the four arrow keys, and can be used in four possible modes:

1/ C1 Mode

In this mode, the arrow keys control the movement of the on-screen cursor whose address is shown on the LED display. The cursor can be moved forwards or backwards by 1 address or up/down (line jumping) by 16 addresses.

Example: Key CSR1 - the display blanks momentarily whilst a screen re-write is made. This key selects cursor 1 (C1) mode.

Now key ↓ and hold the key down. The cursor will jump a line at a time and fall off the screen bottom, reappearing at the top of the next 256 bytes after a screen rewrite. Releasing the key at any time will stop the cursor at the required address.

NOTE: It is not necessary to key CSR1 every time the cursor needs to be moved - this merely puts the machine into C1 mode, having cleared any other mode it may have been in.

2/ Page Mode

When the page mode is selected, the ↑ ↓ cursor keys increment/decrement the page being displayed. These keys enable the entire memory to be paged through without the limitations of the 4k fields. However, it is impossible to view pages 50-56 as these contain the video RAM and the I/O ports.

Example: Key FN PAGE - This puts the machine into Page Mode. By holding the ↓ key the screen is repeatedly rewritten, each rewrite being of the next consecutive Page in memory. To escape from PAGE mode, key RESET, FN FN, or CSR1. This will put the EP8000 into C1 (normal) mode.

3/ dF (define) Mode

A block of data can be 'defined' in this mode, where the block length limit is 8k x 8. A block so defined can be shifted, cleared, relocated or treated in much the same way as any other memory area.

Cursor movement backwards in this mode will define the required block by highlighting it (reverse video). C1 address is shown in the display and it is the start of the block. The end of the block is that address where C1 was when the define mode was entered.

Example: Key DEF and move the cursor back or up - The cursor encloses the required block. Moving the cursor forwards will shorten the block as needed.

4/ SH (shift) mode

In this mode the previously defined data block can be shifted with the cursor control keys. Data is shifted through RAM without overwriting or loss of data. Data in front of the block is transferred to the other side.

NOTE: The longer the block, the longer the delay before any apparant action takes place. This is because the data is moved in the RAM, and when complete the screen is updated to show the new block position. If no block has been defined, then the data at C1 will be shifted.
Example: Define a block as in (3) previously. Key shift ↓ .
The block will shift through memory line at a time for as long as the key is held. To escape from any mode, Key FN FN, RESET or select any other mode E.g. CSR (C1 mode).

3.05 TO - Data relocation

This is the data relocation key used to move data to or from an input/output port, or copy data to another memory area.
Example: PAGE 00 TO RAM will copy the data on Page 00 (256 bytes) to the RAM area starting at 3000 (H). The TO key is used to define PAGE 00 as the object and the RAM as the target area, to indicate the direction of data transfer. This key is useful for input/output control.
Example: SIN TO MEM 3010 will set the machine up to transfer data from the serial port to the RAM area, starting at address 3010 (H).

3.06 DMA Emulation Mode (Direct memory access)

When the DMA key is pressed and released the internal microprocessor is disabled and isolated from the main memory area.
The simulator cable buffers are enabled, to allow an external system to run the program in the EP8000's RAM. The machine now looks like an EPROM of the type selected by the device selector. The chip select and address lines have been automatically configured by the internal microprocessor to suit the selected device.
Prior to keying DMA, the external system should be held reset, or disabled so that when the EP8000's RAM is accessed, the program is executed in a logical manner when the external system is enabled.

3.07 MEM Memory address

This allows the cursor to be moved to any location within the 64k addressed by the internal microprocessor with the exception of the block from 5000-5C00. This block is reserved for the EPROM configuration for the ZIF and emulator.
Example: MEM 3201 will move C1 to address 3201 (H).
The correct page in which this address is located is now on screen.
MEM also serves as a specific memory area where the start of the area is C1 address and the end address is that of the 4k block in which C1 resides.
Example: PAGE 00 TO MEM 3200 will copy the data on page 00 to the RAM starting at address 3200. MEM has been defined as a target by the TO key.

Example: CLEAR MEM 3201 will clear all data at addresses 3201-3FFF (erased state of EPROM).
Example: MEM 3800 TO PROM will transfer data from 3800-3FFF to the EPROM in the ZIF. This is useful when transferring data from one 2732 device into two 2716's (The LED display will show when programming is complete).

3.08 PAGE Screen page selection

The screen can be described as a 'window' into the EP8000 memory. It displays a 256 byte page and 128 bytes of information on preceding and following pages. Any one of 240 pages of memory may be called to the screen.
Example: PAGE 12 will copy the data from 11C0 - 1340 to the screen, with the data from 1200 - 12FF being in the highlighted area.
Page can also be used as a memory area for relocation, programming etc.
Example: CLEAR PAGE 37 will write FF's to all locations on page 37 and at the same time call page 37 to the screen.
The high part of the page number describes a 4k block and the low part divides the 4k into 16 areas of 256 bytes.
If PAGE is not followed by a hex entry, the ↑ ↓ keys will allow the screen to be moved through memory, page at a time.

3.09 CURSOR (CSR)

Depressing this key will define the area from C1 address to the end of the 4k block in which C1 resides as an object or target area for data movement etc. At the same time the machine will go to C1 mode (i.e. Cursor is controlled by the arrow keys, all the shift or define modes are cleared).
Example: CLR CSR (clear cursor) will clear the memory area between C1 and the end of the 4k block provided C1 is in a RAM area.

3.10 SCRATCHPAD RAM (PAD)

Scratchpad RAM is a 1k block starting at 5C00. The PAD key will define this RAM as an object or target. Scratchpad is useful for the temporary storage of many data blocks for later recall to the main RAM area or for direct transfer to EPROM.
Example: Key PAGE 30 TO MEM 5E00 will copy page 30 to the scratchpad starting at address 5E00. In this way, up to 4 pages, or many previously defined data blocks can be temporarily stored for later retrieval.
If an object larger than 1k is copied to the PAD, only the 1st 1k will be copied.

3.11 RAM

Defines the 8k RAM area 3000-4FFF as an object or target.
This is the emulation RAM accessible via the 26 way simulator plug under the control of the DMA key.
Example: Key RAM TO PROM will copy the RAM contents to the ZIF socket. (I.e. program an 8k device).
Example: Key PROM TO RAM will transfer the PROM contents to the RAM starting at address 3000(H).
Example: Key CLR RAM (clear RAM) will clear the RAM to all FF.
(Erased state of EPROM).

3.12 PROM

This defines the programming socket addresses as a target area for EPROM programming or an object area for data movement from the ZIF.
The address limits of the PROM depend upon the device type selected.

Device	Address	Bytes
2704	6000-61FF	512
2708	6000-63FF	1k
2716 3R	6000-67FF	2k
2508	6000-63FF	1k
2758-A 2758-B	6400-67FF	1k
2516 2716 48016	6000-67FF	2k
2532 2732 2732A	6000-6FFF	4k
68764 68766	6000-7FFF	8k
68732-A	6000-6FFF	4k
68732-B	7000-7FFF	4k
2564 2764-A 2764-B	6000-7FFF	8k

Example: Key PROM TO RAM copies PROM data to the RAM starting at address 3000(H).
Example: Key PAGE 00 TO PROM will program the selected device with page 00 data.

3.13 BLOCK (BLCK)

This will fix the previously defined block as an object area only. Once the block has been relocated or cleared etc., the EP8000 reverts to normal C1 mode and the block limits disappear. If no block has been defined and block (BLCK) is keyed, the machine will show this by indicating NO in the display.

Example: Define a block as described in 3.04 (3).

Key BLCK TO MEM 5E 00. This key sequence will copy the defined block into the scratchpad RAM starting at address 5E00. The original data block has not been lost but the block limits have disappeared.

Example: During the course of program development it is often useful to transfer a small program patch into a blank or partially programmed EPROM.

Key BLOCK TO MEM6101 transfers the block data into an EPROM of selected type, the start address being 6101(H).

When the programming cycle is complete, the EP8000 will verify the block data only and indicate pass or fail.

Example: CLR BLCK (Clear block) will clear the defined block to FF.

If the data block defining is not required, it can be cleared by keying FN FN. (Block data is not lost -- only the defining limits).

3.14 EXTERNAL (EXT)

Defines a 2k block of memory, or devices external to the EP8000 at F000-F7FF. This is used by some programming modules but could contain a user program in ROM or extra RAM as a scratchpad.

3.15 DISPLACEMENT (DISP)

This is useful for calculating relative addresses for jumps etc.,

When keyed, the EP4000 will calculate the address difference between the current C1 position and the C1 position when DISP was last keyed. The displacement is shown in the address section of the LED display for 2 seconds. The display then reverts to showing C1 address and data.

The displacement is also displayed on the screen in the following way; When the DISP key is pressed the current cursor location is indicated by two asterisks. The current screen position of the cursor is displayed under the heading FIX which indicates the fixed point. As the cursor is moved away from the fixed point FWD shows the hexadecimal displacement from the cursor to the fixed point and BACK shows the displacement from the fixed point to the cursor.

3.16 PRINT

Outputs the object area to parallel printer in ASCII format. The output is via the expansion connector. Further information is contained in section 5.

Example: PRINT RAM prints the entire 8k of RAM.
Blocks, pages etc., can also be printed using the appropriate key sequences.

3.17 MATCH

Allows the user to highlight specific bytes on-screen for locating program areas. As many bytes as required may be highlighted.
Example: KEY MTCH FF (match FF byte).
All on-screen bytes which are at FF will be shown in reverse video.
The highlighting can be cleared by keying FN FN or RESET.

3.18 EXECUTE (EXEC)

This command key instructs the internal microprocessor to execute the program starting at the specified object address.
This facility allows the user to run programs from the programming socket, RAM area, INT, EXT, PAD or any address (MEM) for specific tasks. E.G. Serial data transfers to customer formats.
The program must start with an 08 (NOP) instruction in order to start execution. If the program starts with any other instrcution, the EP8000 will not execute the program, indicate NO and revert to C1 mode. The internal microprocessor used is the INS8060.
Example: EXEC MEM 6000 will start program execution from address 6000 (ZIF socket) provided this location contains an 08 byte. If the user program requires a return to the monitor, it should return using an XPPC3 instruction to 0000(H).

3.19 DEFINE (DEF)

Depressing the key identifies C1 address as the end of a datablock. Backward or upward cursor movement will extend the cursor to define the block. The current cursor position as shown in the LED display is the block start address. Moving the cursor forward will reduce the block length as required. The block so defined can now be cleared, shifted, relocated etc.
Example: Move C1 to address 3100.
Key DEF and move the cursor up to 3000. The block is between 3000 and 3010.
Now move the cursor to 3010. This shortens the block to 3010-3100.
The block can, for example, be cleared by keying CLR BLCK (clear block).
To exit from the define mode, key any other mode (C1,C2,SHIFT) or FN FN or RESET.

3.20 SHIFT

This transfers the cursor control keys to the shift mode to move a defined data block. Data is moved through the RAM area without overwriting - i.e. data is transferred from one to the other side of the block.

This is useful for shifting data to other areas for program rearrangement, and for shifting spaces into the program if data has to be inserted.
Example: Define the block as in 3.19. Key SHIFT and move the block forward with the cursor control keys.
The larger the data block being shifted, the longer will be the delay before any action is seen on-screen. This is due to the microprocessor shifting the data block and then updating the screen when done.

3.21 PROGRAM EPROM (PROG)

This is the EP8000 programming facility. When pressed, the machine will carry out an illegal bit check on the data and it's target in EPROM. If the device is not programmable, the display will show 'fault B' and reset must be keyed to continue.
The programming sequence transfers data starting from 3000 to the EPROM, the amount of data transferred dependent upon the device selected.
Once programming is complete, the machine verifies the data, showing PASS or FAIL and highlighting any discrepancy bytes.

3.22 VERIFY EPROM (VFY)

Compares the EPROM contents with it's specified RAM area. PASS will be shown on the LED display if the data is identical, else FAIL will be shown.
If a FAIL occurs the differences will be highlighted, but only if the page displayed is a RAM page corresponding to the EPROM.

3.23 CHECKSUM EPROM (CHCK)

This will add the value of all bytes in the EPROM defined by the device selection. It provides a compare function without having to modify the RAM contents.
The checksum is shown in the address section of the LED display (4 digits).
Example: With a master EPROM in the ZIF, key CHCK.
The checksum result is noted from the LED display. Replace the master with a copy and key CHCK. If the resulting checksum is the same, one can be fairly confident that the contents of the two devices are the same - you cannot be absolutely sure, because a byte for byte comparison has not been made, only a checksum.
This key also doubles as a 'blank check' since the checksum of blank devices is listed below.

Checksum Table:

Device	Blank Checksum
2704	FE00
2708	FC00
2716	F800
2508	FC00
2758A	FC00
2758B	FC00
2516	F800
2716	F800
48016	F800
2532	F000
2732	F000
2732A	F000
68764	E000
68766	E000
68732A	F000
68732B	F000
2564	E000
2764A	E000
2764B	E000

3.24 STORE EPROM DATA (STOR)

Copies the EPROM contents into the RAM area starting at 3000.

Device	Copied From
2704	6000-61FF
2708	6000-6300
2716	6000-6700
2508	6000-6300
2758A	6000-6300
2758B	6400-6700
2516	6000-6700
2716	6000-6700
48016	6000-6700
2532	6000-6F00
2732	6000-6F00
2732A	6000-6F00
68764	6000-7F00
68766	6000-7F00
68732A	6000-6F00
68732B	6000-7F00
2564	6000-7F00
2764A	6000-7F00
2764B	6000-7F00

When the data is copied, the EP8000 will verify that the data was correctly transferred, and indicate pass/fail in the LED display.

NOTE: An empty ZIF socket looks like a blank EPROM, so storing an empty socket will clear the RAM area in accordance with the device selected.

3.25 SERIAL INPUT (SIN)

Provides a method of loading data to the RAM area of the machine from the serial input line. The input is treated in a similar way to memory areas, where the target area is defined by the TO key.

Example: Key SIN TO RAM, the machine is now set to receive data, but requests further information by showing FL in the status window. The data required is a 4 digit hex 'input word' used to define the transfer speed, port and number of data bits. The table below should be used to select the correct word to your requirement.

INPUT WORD

DIGIT 1	BAUD RATE	DIGIT 2	PORT	DIGIT 3	DIGIT 4	DATA BITS
0	110	0	Cassette	0 GP Binary	1	1
1	300	1	RS232	1 Intel Ascii	2	2
2	600	2	TTL	2 Motorola Exorciser	3	3
3	1200				4	4
4	2400				5	5
5	4800				6	6
6	6400				7	7
7	9600				8	8

When the input word has been entered, the EP8000 will look at the serial line and wait for the first start bit. When the data has been received, the machine will do a checksum, indicating PASS or FAIL in the LED display.

The video display is updated when the transfer is complete.

3.26 SERIAL OUTPUT (SOUT)

Allows any length data block up to 8k to be transmitted via the serial line to peripheral devices. Data is transmitted in the format chosen by the output word.

Example: PAGE 30 TO SOUT. Prepares the machine to transfer the data on page 30 to the serial output line.

The machine requests an output word to describe the speed, format, and number of stop bits.

OUTPUT WORD

No port number is required because data is transmitted from all ports simultaneously. When the machine is outputting data, the LED display will blank to indicate 'busy'. When the transfer is complete the EP8000 will be in C1 mode.

OUTPUT WORD (Cont)

DIGIT 1	SPEED	DIGIT 2	DIGIT 3	DIGIT 4	STOP BITS	
8	110	Enter 0	8 GP Binary	9	1	
9	300	Enter 0	9 Intel Ascii	A	2	
A	600	Enter 0	A Motorola Exorciser	B	3	
B	1200	Enter 0		C	4	
C	2400	Enter 0		D	5	
D	4800	Enter 0		E	6	
E	6400	Enter 0		F	7	
F	9600	Enter 0				

3.27 PARALLEL INPUT (PIN)

Provides a high speed parallel data transfer facility.
The port is viewed by the EP8000 as a memory area, where the TO key defines it as the object area.
Example: Key PIN TO RAM. The display blanks to indicate 'busy' and is waiting for the first handshake transaction to take place.
Binary data should be set up on the port A input lines. When the strobe signal (PB7) is taken low by the peripheral device, data will be latched onto the port and the 'input buffer full' line (PB6) will go high to indicate data is received. When the data has been read from the port and stored at the target address, PB6 will go low to indicate that the next transaction can take place.

3.28 PARALLEL OUTPUT (POUT)

Data can be moved from a defined object area to the parallel port (which has been defined as the target by the TO key).
Example: MEM 3200 TO POUT transfers data from 3200-3FFF to the parallel port by handshake mode.
When the acknowledge (PB7) signal is low, the EP8000 will output binary data. 8 bits in parallel to port A.
It then issies an output buffer full (PB6) signal to strobe data into the peripheral device (active low). When the PB7 is taken high to indicate completion of the transfer, PB6 will also go high. When PB7 is again low, the next transfer will take place.

3.29 INSERT (INST)

This key allows FF bytes to be inserted at the cursor position. The ins function only operates in RAM. When used, the data to the right of the up to the first group of 3 'FF's' will be shifted right. If no block of can be found, the display will flash 'NO'. See example below:

Before INSERT

```
                                        csr
08  11  22  33  66  21  AE  43  7F  50  21  97  33  36  8F  4E
72  11  FF  FF  FF  08  C4  21  C9
          (3 FF's)
```

After INSERT

```
                                        csr
08  11  22  33  66  21  AE  43  7F  FF  50  21  97  33  36  8F
4E  72  11  FF  FF  08  C4  21  C9
          (2 FF's)
```

NOTE:

Caution should be exercised when using this function, as all code up to first block of 3 FF's will be moved. E.g. subroutines could easily be causing incorrect program operation. With this in mind, the function ha arranged so as to treat the RAM as two separate halves.

3.30 DELETE (DEL)

This key effectively deletes the contents of the cursor.
The key can only operate in RAM areas. It operates by shifting all data the cursor and the first block of 3 FF's, one place to the left.
See example below:

Before DELETE

```
08  C4  21  36  99  42  21  7F  33  FF  FF  FF  08
                csr
```

After DELETE

```
08  C4  21  36  42  21  7F  33  FF  FF  FF  FF  08
                csr
```

NOTE:

Caution should be exercised as it is possible to accidently move routine if each routine is not separated by at least 3 FF's.

SECTION 4

4.01 Device Selection

By pressing the DEV key, the device select mode is entered. In this mode, the video display shows a list of all possible device types, with the current device being highlighted. The current device type is also shown in the LED display. The device type may be changed by using the cursor keys to scroll through the list of devices. Pressing any other key will cause a return to the edit mode.

The Programmable devices are:

Device type Shown:

On video	On LED Display	Comments
2704 3R	2704 3R	3R indicates a triple
2708 3R	2708 3R	supply rail device
2716 3R	2716 3R	
2508	2508	
2758 A	A 2758	
2758 B	B 2758	
2516	2516	
2716	2716	
48016	8016	Electrically erasable PROM
2532	2532	
2732	2732	
2732 A	A 2732	21v Programming 2732
68764	8764	Motorola 24 pin devices
68766	8766	
68732 A	A 8732	
68732 B	B 8732	
2564	2564	
2764 A	A 2764	2764's other than Mostek
2764 B	B 2764	Mostek 2764