ELAN DIGITAL SYSTEMS LTD

"E9C" SERIES PROGRAMMER OPERATING INSTRUCTIONS

Rev: 1.1 15.9.86 SW EB8 HW5

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

CHAPTER 1, GENERAL INFORMATION

POWER REQUIREMENTS

The programmer can be supplied pre-wired for 240v, 220v, 120v or 110v 50/60 Hz and consumes approximately 40 watts.

For operation at 240v/220v use a 500mA SLO-BLO fuse 20mm length.
For operation at 120v/110v use a 1A SLO-BLO fuse 20mm length.

When the power is switched on, the system self-tests the integrity of the memory.

SOCKETS

The master and copy sockets are fully protected and powered down during insertion. All are 28 pin zero insertion force (ZIF) sockets. Power is not applied until a device operation cycle has started and is removed when the cycle is completed.

28 Pin Devices: Pin No. 1 is at the top left hand corner.

24 Pin Devices: These devices must be inserted in the lower part of the socket (bottom justified), with pin no.1 uppermost.

CAUTION: Only insert or remove devices in sockets when the system is in its idle mode, i.e., when display is static showing device type.

Do not operate these systems in high static areas unless antistatic precautions have been taken.

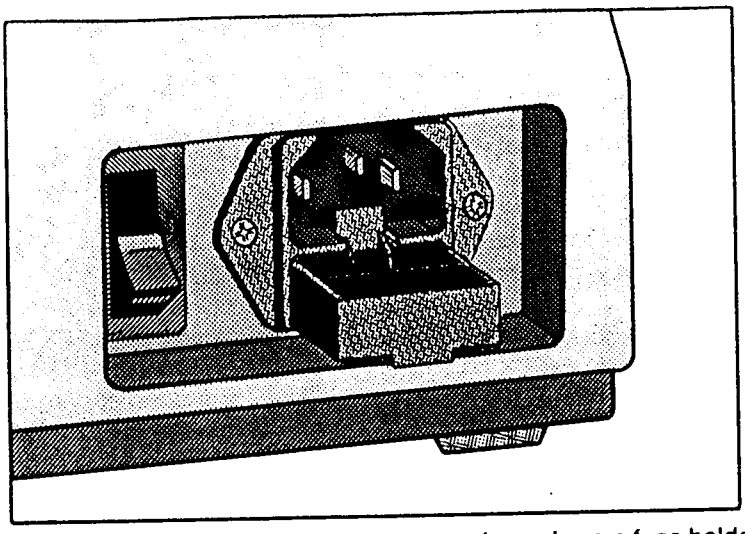
Do not turn power on or off when devices are in any sockets.

BATTERY BACK-UP

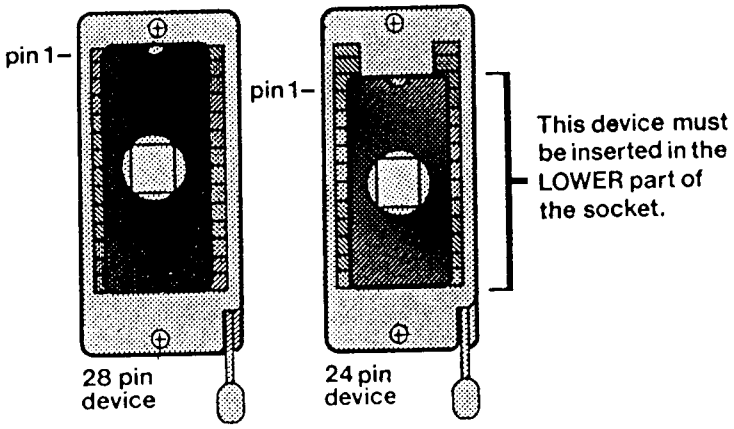
The SCRATCHPAD RAM (not DATA RAM) is supported by a trickle- charged Nickel Cadium cell. Selected defaults for device type, access time, number of sockets, programming mode and serial/parallel configurations are automatically held for periods in excess of six months when powered down.

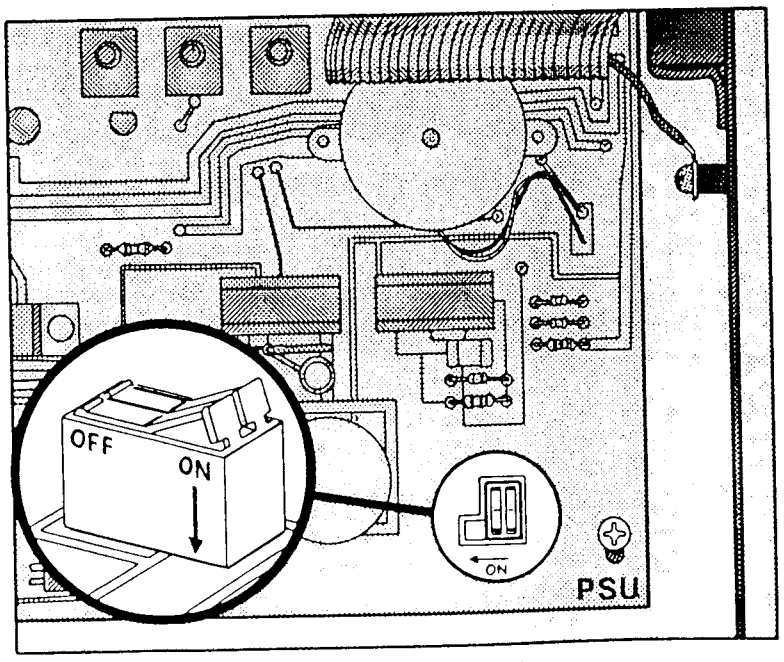
Similarly, when the label printing option is installed, label codes are saved in the powered down state.

If the programmer is not to be used for periods in excess of one year, the battery should be switched off, using the switch adjacent to the Ni-Cad cell on the PSU board.



I.E.C. supply socket with integral fuse carrier and spare fuse holder.
(front compartment)





Location of battery switch on the PSU board.

PRIMARY CONTROL MODES

PUSHBUTTONS

PROG VFY BLK READ <-STEP-> RST ENTER

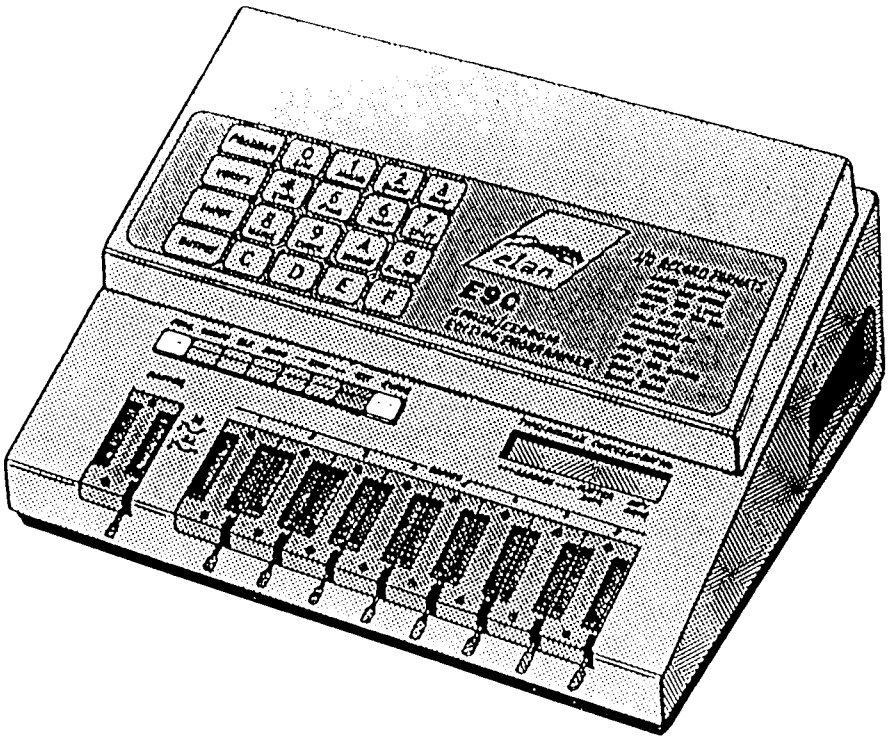
PROG	-	Program from master
VFY	-	Verify from master
BLK	-	Blank check (* NOTE 1)
READ	-	Read master device into RAM
STEP <-	-	Step backwards
STEP ->	-	Step forwards
RST	-	Reset operation (* NOTE 2)
ENTER	-	Enter operation (* NOTE 3)

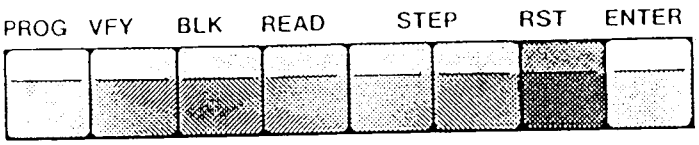
RAM KEYPAD CONTROL

PROGRAM	0	1	2	3
VERIFY	4	5	6	7
INPUT	8	9	A	B
OUTPUT	C	D	E	F

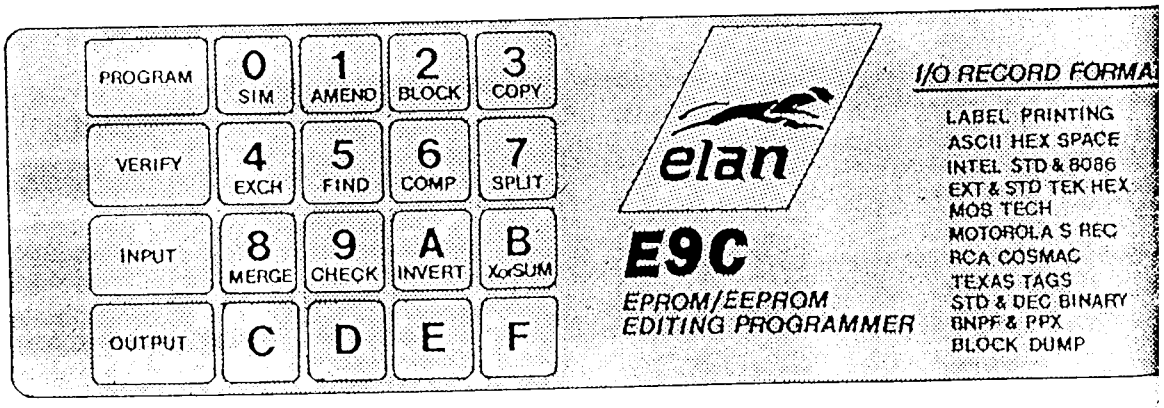
PROGRAM	-	Program from RAM
VERIFY	-	Verify from RAM
INPUT	-	Input data to RAM
OUTPUT	-	Output data from RAM
0 to F	-	Hexadecimal numbers (Editing mode)
0	-	Simulate (with E11 adapter)
1	-	Amend data
2	-	Block change to set value
3	-	Copy block of data
4	-	Exchange string of characters
5	-	Find string of characters
6	-	Compare RAM with master
7	-	Split even/odd bytes
8	-	Merge top and bottom halves
9	-	Calculate checksum
A	-	Convert RAM to one's complement
B	-	Calculate EXCLUSIVE - OR Checksum
C-F	-	Unallocated

* Note 1: See Page 4-2 "Chip erase EEPROMs" for second function.
* Note 2: See Appendix D "E11 Simulator" for secondary function.
* Note 3: See Appendix D "E11 Simulator" for secondary function.





PROG	Program copy socket(s) with data from master socket
VFY	Verify copy socket(s) with data from master socket
BLK	Check copy socket(s) for blank eproms, erases E^2 proms
READ	Read master socket data to RAM
←STEP	Step backwards when viewing data
STEP→	Step forwards when viewing data
RST	Reset – Resets from mode or permits selection of defaults
ENTER	Performs entry of data when editing



PROGRAM	Program copy socket(s) with data from RAM
VERIFY	Verify copy socket(s) with data from RAM
INPUT	Input serial data in selected format
OUTPUT	Output serial or parallel data in selected format

0 – F	Hexadecimal numbers
0	Simulator
1	Amend data
2	Block change to set value
3	Copy block of data
4	Exchange string of characters
5	Find string of characters
6	Compare RAM with master
7	Split odd and even bytes
8	Merge top and bottom halves
9	Calculate checksum
A	Convert RAM to ONE's Complement
B	Calculate Exclusive – or Checksum

PROGRAMMING MODES

Mode A

This is the program routine normally used.
Verify where possible, each byte of the device immediately after it has been programmed.
Complete program cycle as follows:-

a. Illegal bit test of whole device. To confirm that the required data can be stored in the device.

b. Set up next address (start zero).

c. If data correct, go to step b.

d. Program this address with data.

e. If data correct, go to step b.

f. Verify whole device with required data.

Mode B

Inhibit verify during program cycle of standard 50ms pulse device selections. This mode should not normally be used. It has only been incorporated to allow early Texas 2764 devices to be programmed. These devices do not meet specifications because they will not verify during program cycle.

Complete program cycle as follows:

a. Illegal bit test of whole device. To confirm the required data can be stored in the device.

b. Program all locations in the device(s).

c. Verify whole device with required data.

DISPLAY

The system has an alpha numeric display. The messages are:

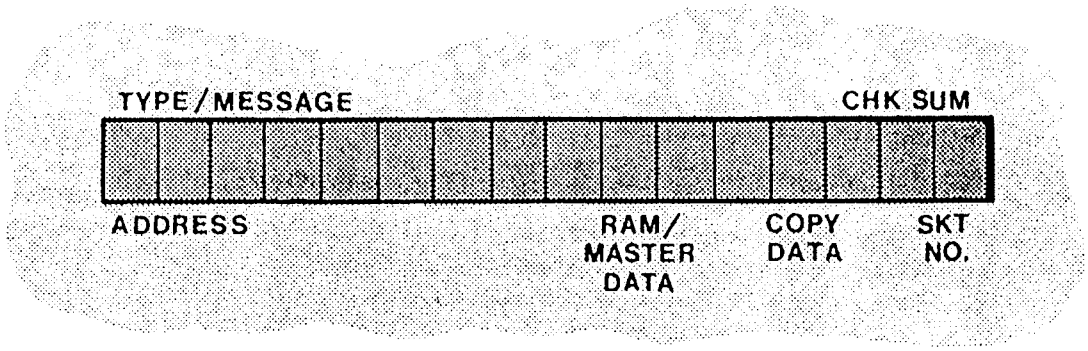
DIGIT	DESCRIPTION
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	
2 7 6 4 I n t	- Device type selected
2 5 0	- Access time in nano seconds
8	- Number of sockets
A	- Programming mode
E n d C C C C	- End and checksum of all devices
C h s u m S k n C C C C	- Checksum of socket 'n'
F a i l m e s s a g e	- Error message
A A A A	- Indicates current address
D D	- Indicates master data
D D	- Indicates copy data
W a i t	- Indicates system busy

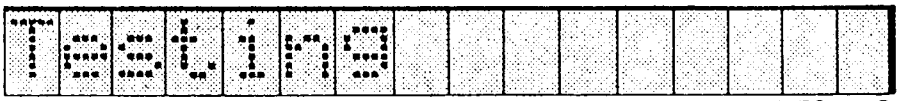
CAUTION

Disconnect power cord before opening the case.

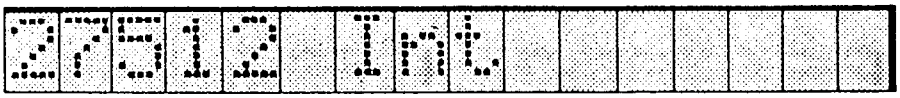
The intensity of the Liquid Crystal Display, to suit any particular viewing angle, can be adjusted by means of a control mounted underneath the main control board. (Accessible by removing the two securing screws at either side of the case front and hingeing the case upwards.)

DISPLAY EXAMPLES

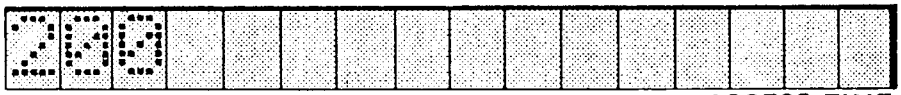




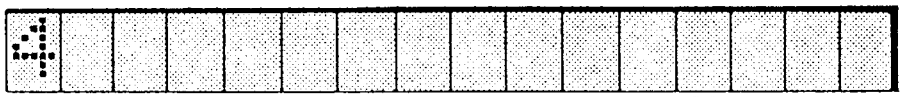
POWER UP TESTING



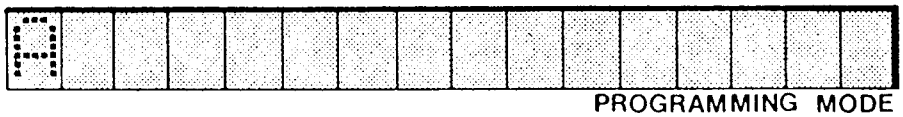
DEVICE TYPE



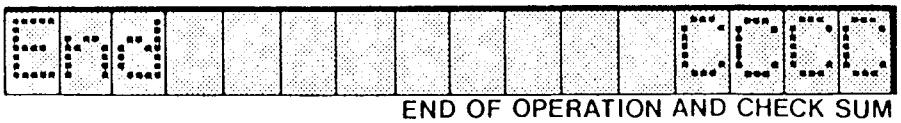
ACCESS TIME



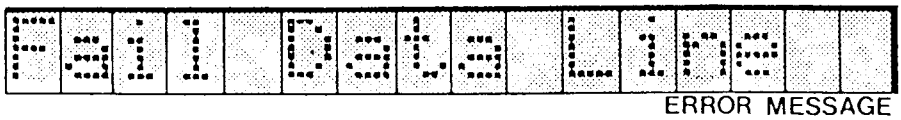
NO. OF COPY SOCKETS



PROGRAMMING MODE



END OF OPERATION AND CHECK SUM



ERROR MESSAGE

ADDRESS: 1234
MASTER DATA: C3
COPY DATA: FF
SKT NO: 3

ERROR CODES

Fail scratchpad	-	System internal scratchpad error - Return unit for service
Fail RAM	-	System internal RAM error - Return unit for service
Fail Master Read	-	Master socket device read error - Check device pins are clean and properly secured in Z1F socket - device may be faulty
Fail Prog Sk n	-	Verify error during program cycle - Check device pins properly secured in Z1F socket - device may be faulty
Fail Data Line	-	Device data line fail - Device may be faulty
Fail Vcc Power	-	Vcc fail - Device may be faulty
Fail Vpp Power	-	Vpp fail - Device may be faulty
Fail RS232 fram	-	RS232 framing - See Page 6-3
Fail RS232 parit	-	RS232 parity - See Page 6-3
Fail RS232 oflow	-	RS232 overflow - See Page 6-3
Fail RS232	-	RS232 combination - See Page 6-3
Fail Rec Ch Sum	-	RS232 checksum - See Page 6-3
Fail ID Man Code	-	Intelligent Identifier manufacturer code error - Check device is specified to contain Intelligent Identifier
Fail ID Dev Code	-	Intelligent Identifier device code error - Check device is specified to contain Intelligent Identifier
Fail Acc Time	-	Access time fail during verify - Access time may be set too fast
Fail Verify	-	Verify fail during access time test - Master data not equal to copy data
Fail No. of Skts	-	More devices in copy sockets than number selected
Fail Blank Ch n	-	Blank fail on socket n in set mode
Fail Verify n	-	Verify fail on socket n in set mode
Fail Illegal bit n	-	Illegal bit fail on socket n in set mode - Bit set to zero where master data requires a 1
Fail Dev Insert	-	Device incorrectly inserted
Fail Prog E4/7	-	E4/E7 adapter device failed to program
Fail E4 Insert	-	E4 adapter device incorrectly inserted
Fail Read E4/7	-	E4/E7 adapter device read fail
Fail E11 Data Ln	-	E11 output data fail
Fail Read E5 Dev	-	Fail to read device in E5 adapter
Fail Prog E5 Dev	-	Fail to program device in E5 adapter
Fail Read E13	-	Fail to read device in E13 adapter
Fail Test E13	-	Device in E13 has failed. Could be Vcc, Vpp or Data

DEVICE SELECTION GUIDE

NOTE

"*" These devices are verified during the program cycle. Therefore the number of sockets in use must be selected.

*	2508	
*	2716	Includes 2516,27C16 (single voltage type only)
	2532	
*	2732	Includes 27C32
*	2732A	
	2564	
*	2764	INTEL using 50ms program pulse
*	2764 Int	INTEL using INTEL intelligent program algorithm
*	2764 Int ID	INTEL intelligent identifier & program algorithm
*	2764 Fuj	FUJITSU "QUICKPRO" intelligent program algorithm
*	2764A	INTEL using INTEL intelligent program algorithm Vpp=12.5V
*	QP2764A Int	INTEL using INTEL 'Quick-Pulse' Programming (tm)' intelligent program algorithm. Vpp = 12.75v
*	QP2764A Int ID	INTEL intelligent identifier & 'Quick-Pulse Programming (tm)' Vpp = 12.75v
*	27C64 Int	INTEL using INTEL intelligent program algorithm Vpp=12.5v
*	27C64 Int ID	INTEL intelligent identifier & program algorithm Vpp=12.5v
*	87C64 Int	INTEL using INTEL intelligent program algorithm Vpp=12.5v
*	87C64 Int ID	INTEL intelligent identifier & program algorithm Vpp=12.5v
*	QP87C64 Int	INTEL using INTEL 'Quick-Pulse' Programming (tm)' intelligent program algorithm. Vpp = 12.75v
*	QP87C64 Int ID	INTEL intelligent identifier & 'Quick-Pulse Programming (tm)' Vpp = 12.75v
	68764	MOTOROLA
	27128	INTEL using 50ms program pulse
	27128 Int	INTEL using INTEL intelligent program algorithm
	27128 Int ID	INTEL intelligent identifier & program algorithm
	27128 Fuj	FUJITSU "QUICKPRO" intelligent program algorithm
	27128A	INTEL using INTEL intelligent program algorithm Vpp=12.5v
*	QP27128A Int	INTEL using INTEL 'Quick-Pulse Programming (tm)' intelligent program algorithm. Vpp = 12.75v
*	QP27128A Int ID	INTEL intelligent identifier & 'Quick-Pulse Programming (tm)' Vpp = 12.75v
*	27256 Int	INTEL using INTEL intelligent program algorithm Vpp=12.5v
*	27256 Int ID	INTEL intelligent identifier & program algorithm Vpp=12.5v

* QP27256	Int	INTEL using INTEL 'Quick-Pulse Programming (tm)' intelligent program algorithm. Vpp - 12.75v
* QP27256	Int ID	INTEL intelligent identifier & 'Quick-Pulse Programming (tm)' Vpp - 12.75v
* 27C256	Fuj 21v	FUJITSU "QUICKPRO" intelligent program algorithm Vpp=21V
* 27512	Int	INTEL using INTEL intelligent program algorithm Vpp=12.5v
* 27512	Int ID	INTEL intelligent identifier & program algorithm Vpp=12.5v
* QP27512	Int	INTEL using INTEL 'Quick-Pulse Programming (tm)' intelligent program algorithm. Vpp -12.75v
* QP27512	Int ID	INTEL intelligent identifier & 'Quick-Pulse Programming (tm)' Vpp - 12.75v
* 27512	AMD	AMD intelligent identifier & program algorithm Vpp=12.5v
* 27513	Int	INTEL using INTEL intelligent program algorithm Vpp=12.5v
* 27513	Int ID	INTEL intelligent identifier & program algorithm Vpp=12.5v
* QP27513	Int	INTEL using INTEL 'Quick-Pulse Programming (tm)' intelligent program algorithm. Vpp - 12.75v
* QP27513	Int ID	INTEL intelligent identifier & 'Quick-Pulse Programming (tm)' Vpp - 12.75v
* 27011	Int	INTEL using INTEL intelligent program algorithm Vpp - 12.5v
* 27011	Int ID	INTEL intelligent identifier & program algorithm Vpp - 12.5v
* QP27011	Int	INTEL using INTEL 'Quick-Pulse Programming (tm)' intelligent program algorithm. Vpp - 12.75v
* QP27011	Int ID	INTEL intelligent identifier & 'Quick-Pulse Programming (tm)' Vpp - 12.75v
* 2815	]	Programming Intel, Motorola and Xycor devices and
* 2816	]	erasing Intel devices. For erasing Motorola and Xycor devices contact Elan for advice.
48016		HITACHI E^2 device Chip erase feature supported.
* X2864A	Xic	Xicor E^2 device
* 2864B	Int	Intel E^2 device
* 2864B	Int ID	Intel E^2 device using Intel intelligent identifier.
* 58064P	Hit	Hitachi E^2 device.Chip erase feature supported.

(SEE NOTES ON PAGE 1-8)

ADAPTER DEVICE SELECTION GUIDE

8741A	E4	Requires E4 adapter
8742	E4	" " "
8744	E7	Requires E7 adapter
8748	E4	Requires E4 adapter
8748H	E4	" " "
8749H	E4	" " "
8755A	E4	" " "
8751	E7	Requires E7 adapter
8751H	E7	" " "
8752	E7	" " "
68701	E5	Requires E5 adapter
2704	E13	Requires E13 adapter, for 3 rail versions of this device
2708	E13	" " " " " " " " " " "
2716	E13	" " " " " " " " " " "

NOTES

Int. Fast programming using intelligent programming algorithm specified by INTEL. This reduces programming time for the 2764 from 7 minutes to approximately 50 seconds and the 27128 from 13 minutes to approximately 100 seconds, and pro-rata for larger devices. During fast programming Vcc is raised from 5v to 6v, and reduced to 5v when verifying.

Int ID. Intelligent identifier and programming algorithm specified by INTEL. The intelligent identifier mode allows the reading out of binary code from a hidden area in an EPROM not forming part of the user memory and enables the programmer to identify the EPROM manufacturer and device type. Not all EPROMs have this code and damage could be caused to the EPROM if this mode is used incorrectly. Check with your distributor or EPROM manufacturer.

'Quick-Pulse Programming (tm)' Fast programming using intelligent programming algorithm specified by Intell for one time programmable (otp) devices with pre-fix P eg P2764A and for 32 pin plastic leaded chip carrier (plcc) devices with pre-fix N eg N27C64.

PROGRAMMING TIME IS REQUIRED TO APPRX. 2 SECONDS for a P2764A (not including pre & post programming checks).

Fuj. Fast programming using "QUICKPRO" programming algorithm specified by FUJITSU. This reduces programming time for the 2764 from 7 minutes to approximately 20 seconds and the 27128 from 13 minutes to approximately 40 seconds. During fast programming Vcc is raised from 5v to 6v, and reduced to 5v when verifying.

AMD ID. Intelligent identifier and programming algorithm specified by AMD. The intelligent identifier mode allows the reading out of binary code from a hidden area in an EPROM not forming part of the user memory and enables the programmer to identify the EPROM manufacturer and device type. Not all EPROMs have this code and damage could be caused to the EPROM if this mode is used incorrectly. Check with your distributor or EPROM manufacturer.

CHAPTER 2, SETTING UP THE PROGRAMMER

TO SELECT DEVICE TYPE, ACCESS TIME, NUMBER OF SOCKETS AND PROGRAMMING MODE

	DISPLAY
1. The display indicates the current device type setting.	[27128]
2. Press RST. The system beeps and the display flashes the current device type.	[27128]
a. Press STEP-> and hold to scroll forward through the device types, or	[27256]
Press STEP<- and hold to scroll backwards through the device types.	[27128]
3. When the device type is correct, press RST. The system beeps and the display flashes the current access time test limit in nano seconds.	[350]
a. Press STEP-> and hold to scroll faster times, or	[300]
Press STEP<- and hold to scroll slower times.	[400]
4. When the access time is correct, press RST. The system beeps and the display flashes the current number of devices.	[3]
a. Press STEP-> and hold to scroll forward, or	[4]
Press STEP<- and hold to scroll backward	[2]
5. When the number of devices/set mode selected is correct, press RST. The system beeps and flashes the current programming mode.	[A]
a. Press STEP-> to change mode if required.	[B]
6. When all selections are correct, press ENTER The system beeps and returns to the idle mode.	[27128]

TO ACTIVATE SYSTEM VARIABLES

1. Turn programmer off.

2. Press both STEP buttons while the programmer is switched on and hold until the system beeps. When the display clears of the message "Testing" it will display the device type followed by "Sys".

3. Press PROGRAM button and release quickly. The display will show two digits representing the variables currently set.

   To leave variables unchanged:

   a. Press RST.

   To change variables:

   a. Press ENTER.
   b. Key in required value (see table on page 2-3)
   c. Press ENTER.

4. Programmer will go into Testing mode then revert to device type display in idle mode and is then ready for use.

SYSTEM VARIABLES AVAILABLE (2 DIGIT CODE)

LEFT DIGIT

INHIBIT LINE FEED	SPECIAL (ALWAYS NO) *	REMOTE INHIBIT RTN/LF AFTER >	REMOTE INHIBIT RTN/LF BEFORE >	LEFT DIGIT
NO	NO	NO	NO	0
NO	NO	NO	YES	1
NO	NO	YES	NO	2
NO	NO	YES	YES	3
NO	YES	NO	NO	4
NO	YES	NO	YES	5
NO	YES	YES	NO	6
NO	YES	YES	YES	7
YES	NO	NO	NO	8
YES	NO	NO	YES	9
YES	NO	YES	NO	A
YES	NO	YES	YES	B
YES	YES	NO	NO	C
YES	YES	NO	YES	D
YES	YES	YES	NO	E
YES	YES	YES	YES	F

RIGHT DIGIT

ENABLE REMOTE RESPONSE DELAY	RCA INHIBIT RETURN AFTER !M	RCA WITH MICRO MONITOR	SERIAL TIME OUT	RIGHT DIGIT
NO	NO	NO	NO	0
NO	NO	NO	YES	1
NO	NO	YES	NO	2
NO	NO	YES	YES	3
NO	YES	NO	NO	4
NO	YES	NO	YES	5
NO	YES	YES	NO	6
NO	YES	YES	YES	7
YES	NO	NO	NO	8
YES	NO	NO	YES	9
YES	NO	YES	NO	A
YES	NO	YES	YES	B
YES	YES	NO	NO	C
YES	YES	NO	YES	D
YES	YES	YES	NO	E
YES	YES	YES	YES	F

NOTE: For EASYCOM IBM remote control, set code to 00.

For INTEL MDS remote control, set code to 90.

* "SPECIAL" is a special protocol, it should always be set to NO.

CHAPTER 3, EPROM/EEPROM FUNCTIONS

TO READ MASTER DEVICE INTO RAM AND SCROLL MASTER DATA

		DISPLAY
1.	Select device type and place device in master socket.	[27128]
2.	Press READ. The system displays start address 0000.	[0000]
3.	Key in required start address.	[XXXX]
4.	Press ENTER. The system beeps, pauses momentarily and displays "wait" while reading the whole device into RAM. The system beeps when finished reading and displays the checksum.	[CCCC]
5.	Press READ to read the data in the device. The system beeps and the display indicates address 0000 and data dd.	[0000 dd]
6.	Press STEP-> to scroll forwards through the addresses of the device, or	[0001 dd]
	press STEP<- to scroll backwards through the addresses of the device.	[3FFF dd]
	NOTE: If the STEP button is held for 8 continuous address steps, the next significant address digit will start to scroll. This enables you to quickly read any area of the device.	
7.	Press RST to terminate the operation and power down the socket. The system returns to the idle mode.	[27128]
8.	Remove the device from the socket.	
	NOTE: Steps 5 and 6 enable the copier to examine data in devices. However, the alternative method of using the editing controls may be more convenient.	

TO BLANK CHECK DEVICES

		DISPLAY
1.	Select device type and place devices into copy sockets.	[27128]
2.	Press BLK. The system blank checks the copy sockets consecutively. The display indicates the system is busy and the socket number being checked.	[Wait n]
3.	If the devices are blank, the display indicates end.	[End]
	If the device is not blank, the display indicates the address AAAA, data blank FF, device data dd and the socket number n.	[AAAA FF dd n]
	a. Press STEP-> to continue blank check on the remaining addresses of the device.	
	b. Press ENTER to continue blank check on the next copy socket device.	[Wait n]
	c. Press RST to terminate operation.	[27128]

TO CHIP ERASE EEPROMS

1.	Select device type and place device into copy socket.	[2815]
2.	Press BLK and hold for two beeps. The system erases the copy socket devices and proceeds to blank check the devices consecutively. The display indicates the system is busy and the copy socket number currently being processed.	[Wait n]
3.	If all of the devices are blank, the display indicates end.	[End]
	If a device is not blank, the display indicates the address AAAA, data blank FF, device data dd, and the socket number n.	[AAAA FF dd n]
	a. Press STEP-> to continue blank check on the remaining addresses of the device.	
	b. Press ENTER to continue blank check on the next copy socket device.	[Wait n]
	c. Press RST to terminate operation.	[2815]

TO VERIFY COPIES WITH MASTER DEVICE

	DISPLAY
1. Select device type, place master into master socket and copies into copy sockets.	[27128]
2. Press VFY. The system verifies each copy device consecutively. The display indicates the system is busy and the copy socket number currently being processed.	[Wait n]
3. If all of the devices verify correctly, the display indicates end.	[End]
If a device has different data, the display indicates the address AAAA, master data MM, device data dd, and socket number n.	[AAAA MM dd n]
a. Press STEP-> to continue verify check on the rest of the device, or Press STEP<- to display the data at the previous address, or	[AAAA MM dd]
Press ENTER to continue verify check on the next copy socket device, or	[Wait n]
Press RST to terminate operation.	[27128]
If a device fails to verify at the set access time but can be read correctly at a slower speed, the display indicates the error message and the socket number.	[Fail Acc Time n]
a. Press ENTER to continue verify check on the next copy socket device.	[Wait n]
b. Press RST to terminate operation.	[27128]

TO PROGRAM COPIES FROM MASTER

1. Select device type, place master into master socket and blank devices into copy sockets.	[27128]
2. Press PROG and hold for two beeps. The system interrogates the copy devices to confirm the master data can be programmed into the copy devices. The display indicates the system is busy and the socket number under test.	[Wait n]
If a device has an illegal bit programmed, the system will beep continuously and display the address AAAA, master data MM, and the copy data dd.	[AAAA MM dd]

TO PROGRAM COPIES FROM MASTER (continued)

	DISPLAY
a. Press ENTER to continue test on the next device and program regardless.	[Wait n]
b. Press RST to terminate the operation; replace the offending device, and start again.	[27128]
If the copy device is not blank but can be programmed the system beeps and displays message. Press ENTER to continue the operation to program the copy devices.	[Copy Not Blank] [AAAA]
To reduce programming time, the system skips bytes containing blank data and continues to display the last address programmed.	
Where possible (see device selection guide), the system verifies the devices before and after applying the program pulse. The system does not program bytes already containing the correct data. If a device fails to program, the system stops programming and displays the error message. Where n represents the socket number in error, press ENTER to step to the next error.	[Fail Prog Sk n]
At the end of the program cycle, the system verifies the master with the copy devices. The display indicates the system is busy and the socket number under test.	[Wait n]
If all of the devices program correctly, the display indicates End and the checksum CCCC. The system will beep periodically until RST is pressed.	[End CCCC]
If a device has different data, the display indicates the address AAAA, master data MM, device data dd, and socket number n.	[AAAA MM dd n]
a. Press STEP-> to continue verify check on the remaining addresses of the device.	
b. Press STEP<- to display the data at the previous address.	
c. Press ENTER to continue verify check on the next copy socket device.	[Wait n]
d. Press RST to terminate operation.	[27128]

TO PROGRAM COPIES FROM RAM

		DISPLAY
1.	Select device type and place devices into copy sockets.	[27128]
2.	Press RAM PROGRAM. The display indicates start address.	[0000]
3.	Key required start address.	[XXXX]
4.	Press ENTER. The system interrogates the copy devices to confirm the RAM data can be programmed into the copy devices. The display indicates the system is busy and the socket number under test.	[Wait n]
	If a device has an illegal bit programmed, the system will beep continuously and display the address AAAA, RAM data RR and the socket number n.	[AAAA RR n]
	a. Press ENTER to continue test on the next device and program, regardless.	[Wait n]
	b. Press RST to terminate the operation.	[27128]
	If the copy device is not blank but can be programmed the system beeps and displays message. Press enter to continue the operation to programme the copy devices.	[Copy Not Blank] [AAAA]
	To reduce programming time, the system skips bytes containing blank data and continues to display the last address programmed.	
	Where possible (see device selection guide), the system verifies the devices before and after applying the program pulse. The system does not program bytes already containing the correct data. If a device fails to program, the system stops programming and displays the error message. Where n represents the socket number in error. Press ENTER to step to the next error.	[Fail Prog Sk n]
	At the end of the program cycle, the system verifies the RAM with the copy devices. The display indicates the system is busy and the socket number under test.	[Wait n]
	If all of the devices program correctly, the display indicates End and the checksum CCCC.	[End CCCC]

TO PROGRAM COPIES FROM RAM (continued)

	DISPLAY
If a device has different data, the display indicates the address AAAA, RAM data RR, device data dd, and socket number n.	[AAAA RR dd n]
a. Press STEP-> to continue verify check on the remaining addresses of the device.	
b. Press STEP<- to display the data at the previous address.	
c. Press ENTER to continue verify check on the next copy socket device.	[Wait n]
d. Press RST to terminate operation.	[27128]

TO VERIFY COPIES WITH RAM DATA

1.	Select device type and place copy devices into copy sockets.	[27128]
2.	Press RAM VERIFY. The display indicates start address.	[0000]
3.	Key in required start address.	[XXXX]
4.	Press ENTER. The system verifies the copy device consecutively. The display indicates the system is busy and the socket number currently being processed.	[Wait n]
5.	If the devices verify correctly, the display indicates End.	[End]
	If a device has different data, the display indicates the address AAAA, RAM data RR, device data dd, and socket number n.	[AAAA RR dd n]
	a. Press STEP-> to continue verify check on the rest of the device.	
	b. Press STEP<- to display the data at the previous address.	
	c. Press ENTER to continue verify check on the next copy socket device.	[Wait n]
	d. Press RST to terminate operation.	[27128]

TO VERIFY COPIES WITH RAM DATA (continued)

	DISPLAY
If a device fails to verify at the preset access time, but can be read correctly at a slower speed, the display indicates the error message and the socket number.	[Fail Acc Time n]
a. Press ENTER to continue the verify check on the next copy socket.	[Wait n]
b. Press RST to terminate operation.	[27128]

TO FIND ACCESS TIME OF COPIES

FROM MASTER

1. Select device type, place master into master sockect and copies into copy sockets.	[27128]
2. Press VFY and hold for double beep. The system verifies one copy at a time with master to identify the fastest speed the device can be read. When calculated, the display shows the access time and the socket number.	[Wait n] [Acc Time 200 n]
If the master data is different to the copy data, the display indicates the error message and the socket number.	[Fail Verify n]
a. Press ENTER to continue the test on the next socket.	[Acc Time 200 n]
b. Press RST to terminate the operation.	[27128]

FROM RAM

1. Select device type, enter data into RAM, and insert copies into copy sockets.	
2. Press RAM VERIFY. The display indicates RAM start address.	[0000]
3. Key required RAM start address.	[XXXX]
4. Press ENTER and hold for double beep. The system verifies one copy at a time with RAM to identify the fastest speed the device can be read. When calculated, the display shows the access time and the socket number.	[Acc Time 200 n]

TO FIND ACCESS TIME OF COPIES (continued)

	DISPLAY
If the master data is different to the copy data, the display indicates the error message and the socket number.	[Fail Verify n]
a. Press ENTER to continue the test on the next socket.	[Acc Time 200 n]
b. Press RST to terminate the operation.	[27128]

CHAPTER 4, RAM EDITING FUNCTIONS

EDITING FUNCTIONS USING RAM (64K BYTE STANDARD IN E9C) (128K BYTE OPTIONAL)

1. AMEND DATA
2. BLOCK CHANGE
3. COPY BLOCK
4. EXCHANGE STRING
5. FIND STRING
6. COMPARE RAM
7. SPLIT RAM
8. MERGE RAM
9. CHECKSUM
A. ONE'S COMPLEMENT
B. X OR SUM

(1) AMEND DATA

To change data currently in RAM	DISPLAY
a. Press hex keyboard key 1. Display indicates address 0000.	[0000]
b. Enter required address through keyboard or,	[XXXX]
Press RST to exit from AMEND mode and display device type.	[27128]
c. Press ENTER button to enter the address number and to display the current data at that address. The display shows the address AAAA, and current data DD.	[AAAA DD]
d. Enter required data through keyboard. Display indicates address AAAA, current data DD and required data RR.	[AAAA DD RR]
e. Press STEP-> to store the amended data in RAM. This will also step to the next address and display details, or	[XXXX DD]
press STEP<- to store the amended data in RAM and step to the previous address and display details. If this address is to be amended, repeat steps d and e, or	[XXXW DD]
press ENTER to store the amended data. This will also reset the address to zero. Key in the next address to be amended and repeat the procedure.	
f. Press RST to exit from amend mode and display device type.	[27128]

(1) AMEND DATA (continued)

EXAMPLE: To amend RAM address 0123 from 45 to 67, address 0124 from A1 to A2, and address 0126 from C2 to 11. Leave address 0125 with FF.

	DISPLAY
Select 1 for amend.	[0000]
Key in address 0123.	[0123]
Press ENTER.	[0123 45]
Key in required data 67.	[0123 45 67]
Press STEP->.	[0124 A1]
Key in required data A2.	[0124 A1 A2]
Press STEP->	[0125 FF]
Press STEP->.	[0126 C2]
Key in required data 11.	[0126 C2 11]
Press STEP->.	[0127 XX]
Press RESET.	

(2) BLOCK CHANGE

To change a block of RAM to the same value.

a.	Press key 2. Display indicates address 0000.	[0000]
b.	Enter start address through keyboard.	[XXXX]
c.	Press ENTER. Display indicates last address 3FFF.*	[3FFF]
d.	Enter last address through keyboard.	[XXXX]
e.	Press ENTER. Display indicates required data "FF".	[FF]
f.	Enter required data through keyboard.	[DD]
g.	Press ENTER to carry out the operation. The display will revert back to device type.	[Wait n] [27128]

* Note: The value of the last address displayed is a function of the device selected. In this example the programmer would be configured for a 27128.

EXAMPLE: To set all locations of RAM, address 0340 to address 0672, to value 28.

Select 2 for block change.	[0000]
Key in address 0340.	[0340]
Press ENTER.	[3FFF]
Key in address 0672.	[0672]
Press ENTER.	[FF]
Key in data "28".	[28]
Press ENTER.	[27128]

(3) COPY BLOCK

To copy a block of data from one area of RAM to another.

		DISPLAY
a.	Press key 3. Display indicates the start address of block.	[0000]
b.	Enter the start address of block to be copied.	[XXXX]
c.	Press ENTER. The display indicates last address of block.	[0000]
d.	Enter the last address of the block to be copied.	[XXXX]
e.	Press ENTER. Display indicates destination.	[0000]
f.	Enter destination address.	[XXXX]
g.	Press ENTER to carry out copy.	[27128]

EXAMPLE: To copy the block of data (address 0100 to 01FF) to address 0E00.

Select 3 for copy.	[0000]
Key in start address 0100.	[0100]
Press ENTER.	[0000]
Key in last address 01FF.	[01FF]
Press ENTER.	[0000]
Key in destination address 0E00.	[0E00]
Press ENTER.	[27128]

(4) EXCHANGE STRING

To exchange a string of characters (1 to 8 characters long) for another string of characters (1 to 8 characters long)

a.	Press key 4. Display indicates start address 0000 of search area.	[0000]
b.	Enter the start address of search area.	[XXXX]
c.	Press ENTER. Display indicates last address of search area.	[3FFF]
d.	Enter the last address of search area.	[XXXX]

(4) EXCHANGE STRING (continued)

		DISPLAY
e.	Press ENTER.	[-]
f.	Enter string of characters to be replaced (1 to 8 characters long). If a particular character is masked, press ENTER to skip to next position.	[SSSSSSSS]
g.	After the eighth character is entered the system is ready to input the required replacement characters.	[-]
h.	Enter string of required characters (1 to 8 characters long). If a particular character is masked, press ENTER to skip to next position.	[SSSSSSSS]
i.	After the eighth digit is input the system carries out the exchange. The display reverts back to the device type when done.	[27128]

EXAMPLE: Exchange all occurrences of the string of data 12131415 to ABCD between addresses 0600 and 07FF.

Assume the contents of each byte of RAM is the least significant 2 characters of the address, i.e.,for address 110,data =10, for address 111, data =11 etc.

Select 4 for exchange.	[0000]
Key in address 0600.	[0600]
Press ENTER.	[3FFF]
Key in address 07FF.	[07FF]
Key in string 12131415.	[-]
Key in string ABCD.	[ABCD-]
Press ENTER four times.	[27128]

Address	Data Before Exchange	Data After Exchange
612	12	AB
613	13	CD
614	14	14
615	15	15
712	12	AB
713	13	CD
714	14	14
715	15	15

(5) FIND STRING

To find a specified string of characters in RAM (1 to 8 characters long) with the option of inspecting and amending bytes in the vicinity.

		DISPLAY
a.	Press key 5. Display indicates start address 0000 of search area.	[0000]
b.	Enter the start address.	[XXXX]
c.	Press ENTER. Display indicates last address of seach area.	[3FFF]
d.	Enter the last address.	[XXXX]
e.	Press ENTER.	[-]
f.	Key in required string of characters (1 to 8 characters long). If a particular character is masked, press ENTER to skip to next position. At the first occurence of the string of data, the operation will stop and display the address.	[AAAA]
g.	Press ENTER to locate the next occurence of the string, or	
	press key 1 to amend the address. Display indicates address and data.	[AAAA DD]
h.	Enter required data through the keyboard. Display indicates address, current data and required data.	[AAAA DD RR]
i.	Press STEP-> to store the amended data in RAM. This will also step to the next address and display details, or	[AAAA DD]
	Press STEP<- to store the amended data in RAM, step to the previous address and display details.	[AAAA DD]
	If this address is to be amended, repeat steps h and i.	
j.	Press RST to exit from AMEND/FIND mode and continue search for next occurrence of the string.	[AAAA]

(5) FIND STRING (continued)

		DISPLAY
k.	When the search is completed, the display will revert back to the device type.	[27128]

EXAMPLE: Find all occurences of the string of data 12131 between the addresses 0300 and 04FF. Examine the data surrounding the string and change the first occurence of data 10 to F0. Assume the contents of each byte of RAM are the least significant 2 characters of the address, i.e, change as on page 5-4.

Select 5.	[0000]
Key in address 0300.	[0300]
Press ENTER.	[3FFF]
Key in address 04FF.	[04FF]
Press ENTER.	[-]
Key in string 12131.	[12131-]
Press ENTER three times.	[0312]
Select 1.	[0312 12]
Press STEP<-.	[0311 11]
Press STEP<-.	[0311 10]
Key in data F0.	[0310 10 F0]
Press STEP->.	[0311 11]
Press STEP->.	[0312 12]
Press STEP->	[0313 13]
Press STEP->	[0314 14]
Press STEP->	[0315 15]
Press RESET	[0412]
Press ENTER	[27128]

Thus, the string occurred at addresses 0312 and 0412

(6) COMPARE RAM

The unit will compare an area of RAM data with a device giving details of differences.

a.	Select device type and place in master socket.	[27128]
b.	Press button 6. Display indicates RAM start address.	[0000]
c.	Enter required RAM start address.	[XXXX]

DISPLAY

d. Press ENTER.

i) If the two data areas are the same, the display will indicate End. [End]

ii) If a mismatch is found, the display indicates thus:- [AAAA DD RR]

press STEP-> to continue to next mismatch. If no mismatch is found, the display will indicate End, or

Press STEP<- to see previous address.

EXAMPLE: Compare two 27128 EPROMs. Assume the two have the same data, except for:

Address	EPROM X	EPROM Y
110	01	02
111	F2	3E
5AF	47	AB
580	96	69
746	22	44
7F1	5A	5B

First load EPROM X into RAM:	[27128]
Select Read.	[0000]
Press ENTER.	[End CCCC]
Press RST.	[27128]
Now compare the two EPROMs	[27128]
Place EPROM Y into master socket.	
Select 6.	[0000]
Press ENTER.	[0110 02 01]
Press STEP->.	[0111 3E F2]
Press STEP->.	[05AF AB 47]
Press STEP->.	[0580 69 96]
Press STEP->.	[0746 44 22]
Press STEP->.	[0741 5B 5A]
Press STEP->.	[End]

(7) SPLIT RAM

Split even and odd bytes of RAM (place even bytes into bottom half and odd bytes into top half of RAM).

		DISPLAY
a.	Press button 7. The display indicates it is busy until the operation is complete.	[Wait]

(8) MERGE RAM

To merge the top and bottom half of RAM (the bottom half becomes the even bytes and the top half the odd bytes of RAM):

a.	Press button 8. The display indicates it is busy until the operation is complete.	[Wait]

(9) CHECKSUM and (B) Exclusive - OR Checksum

To calculate checksum of data between any two addresses in RAM.

a.	Press button 9. Display indicates start address.	[0000]
b.	Key in required start address.	[XXXX]
c.	Press ENTER. Display indicates last address.	[3FFF]
d.	Key in required last address.	[YYYY]
e.	Press ENTER. The display indicates the 4-character checksum after a short wait.	[CCCC]

(A) ONE'S COMPLEMENT

To set RAM to one's complement

a.	Press button A and hold for two beeps. The system will convert each byte of RAM to its one's complement and display the new checksum.	[Wait] [CCCC]

CHAPTER 5, INPUT/OUTPUT COMMUNICATIONS

INPUT/OUTPUT DATA CONFIGURATION

The system has 3 levels to configure:

level 1 - format
level 2 - baud rate
level 3 - data stream

DATA STREAM SELECTION

RECORD FORMAT	BAUD RATE
Label Printing	50
ASCII Hex Space	75
INTEL std & 8086	110
Binary - no header	135
Tek Hex	150
Mos Tech	200
Motorola S Rec	300
Dec Binary	600
Binary - header	1200
Block Dump	1800
RCA Cosmac	2400
PPX	4800
Texas Tags	9600
ASCII BNPF	19200
Ext Tek Hex	

DATA STREAM

Parity	Display
even	ev/par
odd	od/par
no	no/par
Data Bits	
7	7 Db
8	8 Db
Stop Bits	
1	1 St
2	2 St

Example:
8 Db 2 St no/par -
8 Data Bits, 2 Stop Bits, No Parity.

TO SELECT COMMUNICATIONS CONFIGURATION

		DISPLAY
1.	The display indicates current device type.	[27128]
2.	Press RST and hold for two beeps. The display flashes format.	[Intel std & 8086]
3.	Press STEP-> and hold to scroll forward through the selections, or Press STEP<- and hold to scroll backwards through the selection.	[Binary-no header] [ASCII Hex Space]
4.	When the configuration is correct, press ENTER. The system beeps and the display stops flashing, or if format is correct but the baud rate needs changing, press RST. The system beeps and the display flashes the baud rate.	[27128] [Baud Rate 9600]
5.	(Repeat step 3) - with display -	[Baud Rate 4800] [Baud Rate 19200]
6.	When the configuration is correct, press ENTER. The system beeps and the display stops flashing, or if the baud rate is correct but the data stream needs changing, press RST. The system beeps and the display flashes the data stream currently selected.	[27128] [8 Db 1 St ev/par]
7.	(Repeat step 3) - with display -	[8 Db 1 St od/par] [7 Db 2 St no/par]
8.	When the data stream is correct press ENTER.	[27128]

At power down the communications configuration that has been selected will be saved for a period of at least six months.

SERIAL TEST ROUTINE

Quite often the development system serial configuration is unknown and not even available to the user. If this is the case the programmer can be used to narrow down the possibilities quite quickly.

1. Estimate the baud rate. This can be done quite easily by dumping a file from the development system to the serial port and timing it. Then compare the time with the table of values below. A 1k dump in format ASCII Hex Space takes the following times:

Time in Seconds	Baud Rate	Time in Seconds	Baud Rate
2	19200	120	300
4	9600	200	200
8	4800	240	150
15	2400	270	135
22	1800	320	110
30	1200	480	75
60	600	740	50

2. Set the programmer serial configuration code to Binary - no header, Baud Rate R, 8 Db, 1 St, no/par, where R is the correct baud rate. This format will store all data received.

3. Set the programmer into input mode, ready to receive data. Send a small known file to it.

a. If the programmer detects an error the data sent must be 8 data bits and either odd or even parity or 7 data bits, no parity and 1 stop bit. Try them out to confirm this.

b. If the programmer receives the data without detecting an error you have either found the correct configuration the first time or the data is 7 bits with parity. Select the AMEND mode on the programmer to examine the data stored and identify the ASCII code.

EXAMPLE: If an ASCII 0 and 1 has been stored the data will appear as follows:

Stored Data	Diagnosis
30/31	7 data bits, no parity or 8 data bits
30/B1	7 data bits and even parity
B0/31	7 data bits and odd parity

TO INPUT DATA TO THE PROGRAMMER

General -

Some serial formats are designed to handle very large memory address ranges. As the programmer has a finite memory space (user RAM) it is necessary for the programmer to be instructed to select and store the required window in the data stream.

		DISPLAY
1.	Select required communications configuration code.	
2.	Press INPUT.	
3a.	With ASCII Hex. Space, Binary-no header, Dec. Binary Binary-header and ASCII BNPF formats the display indicates start address zero.	[0000]
	i. Enter start address.	[XXXX]
	ii. Press ENTER. It is now ready to receive data.	
3b.	With Tek Hex, Mos Tech, RCA Cosmac, PPX and Texas Tags, the display indicates it is ready to receive data.	[Wait]
3c.*	With Intel Std & 8086 format the display prompts for the entry of the most significant address displacement when using 16 bit format (8086). The Hex digit represents address bits 16 to 19.	[0]
	i. Key most significant address. It is now ready to receive data.	[X]
3d.*	With Motorola S Record format the display prompts for the entry of the most significant address when using S2 and S3 records. The Hex digits represend address bits 16 to 31.	
	i. Key up to 4 digits.	
3e.	With Ext.Tek.Hex format the display prompts for the entry of the start address of the required window in the data stream.	
	i. Key up to 4 digits.	
4.	Transfer data from the development system to the programmer. The display will indicate the address currently being loaded. Note: This address is not updated every record and hence could be misleading.	[AAAA]
5.	At the end of transmission, the display indicates End and the checksum CCCC.	[End CCCC]

* Note: This is to select the correct 64K page of data.

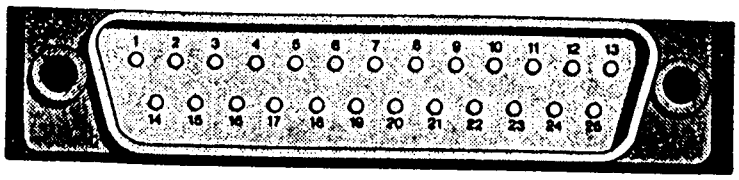
TO OUTPUT DATA FROM THE PROGRAMMER

		DISPLAY	
1.	Select required communications configuration code.	[27128	]
2.	Press OUTPUT. Display indicates start RAM address zero.	[0000	]
3.	Enter required start RAM address if different from 0000.	[XXXX	]
4.	Press ENTER. Display indicates last RAM address.	[3FFF	]
5.	Enter required last RAM address.	[YYYY	]
6.	Press ENTER.		
7. *	For formats 2 and 6 only, key most significant address digits. (See 3c and 3d on page 6-3)	[0 [X	]]
8.	The programmer will now transmit data. The display indicates the current address.	[A000	]
9.	At the end of transmission, the display indicates End and the checksum CCCC.	[End	CCCC]

* Note: This is to select the correct 64K page of data.

RS232 CONNECTIONS

The I/O connector on the back of the programmer is a standard 25 pin "D" type male connector with the data and signal line connections as follows:



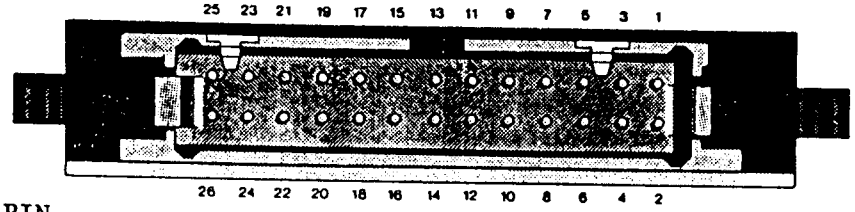
PIN NO.	DESCRIPTION	CLASSIFICATION
1	Chassis Ground	---
2	Serial data out	Output
3	Serial data in	Input
4	Request To Send	Output
5	Clear To Send	Input
6	Data Set Ready	Input
7	Signal ground	---
20	Data Terminal Ready	Output

Although bi-directional handshaking is provided, the programmer can operate without these signals being connected.

The system will also respond to X ON/X OFF while outputting data.

PARALLEL PORT CONNECTIONS

The 26-pin ribbon cable connector on the back of the programmer carries data and handshaking lines for the 16 Bit parallel port.



26 PIN RIBBON CABLE CONNECTOR	DESCRIPTION		CLASSIFICATION	CENTRONICS 36 PIN CONNECTOR
4	Data	0	Output	2
6		1	"	3
8		2	"	4
10		3	"	5
12		4	"	6
14		5	"	7
16		6	"	8
18		7	"	9
1		0V	Common	16
5		Strobe	Output	1
17		Busy	Input	11

Do not make connections to pins not listed in first column. These are used with "E" Series adapters. A mode up cable with a centronics 36 pin connector is available from Elan.

TO SELECT REMOTE CONTROL

1. Switch system on.

2. Select device type (this can be re-selected under remote control.)

3. Select serial configuration.

4. Press and hold ENTER button. The display will show the device type followed by the letters "Rem" on the right.

5. The system is now ready to respond to the commands which are shown starting on page 6-8.

6. Press RST to terminate remote control.

REMOTE CONTROL COMMANDS

COMPUTER COMMAND/ CONTROL COMMAND	NAME	DESCRIPTION
RETURN		Execute last command.
Z	Terminate	Programmer operates in stand-alone mode.
PROGRAMMER STATUS INQUIRY		
D	Odd parity	Programmer confirms parity compatible.
E	Even parity	" "
N	No parity	" "
J	1-stop bit	Programmer confirms stop bit compatible.
K	2-stop bit	" "
X	Error code	Programmer returns last code.
x	Error code inquiry	Programmer returns error code list.
H	Handshake	Programmer returns prompt.
R	EPROM status	Programmer indicates status of EPROM selected. AAAA/B/C, where AAAA = device word limit, B = byte size, and C = VOL/VOH status (1 = VOL; 0 = VOH).
g	Programmer software release	Programmer generation number. (a 4 charecter response)

DEVICE COMMANDS

B	Blank check	Check if EPROM is erased.
b	Erase EEPROM	
T	Illegal bit check	Check data can be programmed into device.
L	Load master	Reads copy socket 1 into RAM.
V	Verify	Verify RAM with copy socket.
P	Program	Program copy socket from RAM
p	Program	8751H security bit using E7 adapter

RAM COMMANDS

I	Input	Input data from computer to RAM. See NOTE below
O	Output	Output data from RAM to computer (up to the word limit of selected EPROM).
S	Checksum	Programmer calculates the two byte checksum of RAM data up to the word limit of the selected EPROM.
C	Compare	Compare input data from computer with RAM.See NOTE below.
nn Y	Fill RAM	Fill RAM within RAM start and end addresses with data "nn".
c	Complement	Convert all RAM to its one's complement.
m	Merge RAM	
s	Split RAM	

NOTE: After an I or C command a 10 millisecond delay must occur before sending the data records.

CONFIGURATION COMMANDS

nn M	Record length	Select output record length nn Hex.
nn U	Nulls	Enables output of nn Hex Nulls after carriage rtn.

CONFIGURATION COMMANDS

nn A	* Select format	Select I/O record format.
nn f	* Select format	Select I/O record format.
nnnnn W	+ Virtual address displacement	Sets RAM address to required system base address. (Only applicable to systems with less than 64K RAM.)
nnnnn w	$ Virtual address displacement	Where nnnn is the Hex value of address lines 16 TO 31.
nnnn :	+ Device start address	
nnnnn <	+ RAM start address	
nnnnn ;	+ RAM end address	Specifies highest RAM address nnnn-1. Defaults to device size. 0000 will set end address FFFF.
nnnn #	+ O/P data start address	
n a	* Access time	
n n	Number of sockets	Where the 1st n = number of sockets
e	Access time inquiry	
k	Number of sockets/set mode inquiry	
r	RAM size inquiry	
nn t	* Select device type	
d	Device type inquiry	
nnnn @	* Select device type	
(	Device type inquiry	

NOTE: The spaces shown in the multiple commands such as "nn t" are for clarity and must be omitted in practice.

* - See following tables for values of "n".

+ - These addresses are reset to device defaults on selection of a device.

$ - This command is for use with formats which have greater than e.g. (i) 64K address capability. For Intel Extended record format which has 20 address lines, the command 000Fw would select address range F0000H to FFFFFH in the programmer.
e.g.(ii) For Motorola S Record format which can accommodate up to 32 address lines the command F000w would select address range F0000000H to F001FFFF in the programmer.

PROGRAMMER RESPONSES	DESCRIPTION
RETURN line feed	a. To indicate command received. (except for commands I and C where no indication is given)
> RETURN line feed	b. On successful completion of command.
F RETURN line feed	b. On unsuccessful completion of command.
? RETURN line feed	b. Command not understood.

a. A software option switch can be set to inhibit this response.

b. A software switch can be set to inhibit the return/line feed after the response ">", "F", and "?".

a & b A software option switch can be set to inhibit all line feeds.

REMOTE ACCESS TIME CODES

ACCESS TIME	a CODE
450	0
400	1
350	2
300	3
250	4
200	5
150	6
100	7

REMOTE DEVICE CODES

TYPE	t CODE	@ CODE
2508	00	1922
2716	01	1923
2532	02	3125
2732	03	1924
2732A	04	2724
2564	05	3130
2764	06	3533
2764 Int	07	7933
2764 Int ID	08	
2764 Fuj	09	4533
2764A	0A	9333
68764	0C	2529
27128	0D	3551
27128 Int	0E	7951
27128 Int ID	0F	
27128 Fuj	10	4551
27128A	11	9351
27256 Int	12	9332
27256 Int ID	13	
27256 Fuj	14	
27512 Int	15	
27512 Int ID	16	
27512 AMD	17	
2815	18	
2816	19	3723
48016	1A	
8741	1B	5654
8742	1C	
8748	1D	5256
8748H	1E	5056
8749H	1F	5057
8755	20	4755
8751	21	
8752	22	
68701	23	
2704	2B	
2708	2C	
TMS 2716	2D	

REMOTE FORMAT CODES

FORMAT	f CODE	A CODE
ASCII hex space	01	50
INTEL LOADER	02	83
Optional	03	-
TEK HEX	04	86
MOS TECH	05	81
MOTOROLA S REC	06	82
DEC BINARY	07	-
Binary	08	10
Block dump	09	-
RCA COSMAC	0A	-
PPX	0B	-
TEXAS TAGS	0C	-
ASCII BNPF FORMAT	0D	-
EXT.TEK.HEX	0E	-

REMOTE ERROR CODES

CODE	DESCRIPTION
01	E series configuration out of range
mm20	Blank check fail
mm21	Illegal bit fail
mm22	Program fail
mm23	Verify fail
29	Read fail
31	Data line fail
37	None EE device
38	Device fail
81	Serial stream error
82	Serial I/O error

Where "mm" is a mask indicating the socket numbers which fail the test. Note: This will only be sent on an "x" inquiry, not an "X" inquiry.

E.g.,

mm	skt number
80	1
40	2
20	3
10	4
08	5
04	6
02	7
01	8
FF	all
C0	1 & 2
81	1 & 8
etc.	

CHAPTER 6, EXTRA OPTIONS

LABEL PRINTING (OPTIONAL)

The details of eight labels can be entered and stored by the programmer. 64 characters of memory are allocated to each label area. Special print control characters can be entered to select different print formats (e.g., condensed or bold characters).

The programmer will output the characters from the first location until it finds a carriage return (OD hex). It will then repeat this text the number of times selected across the page. It will then output the next line of text until it finds the end of label character 00 hex.

NOTE: a. Each line of text must terminate with a carriage return.

b. Each label must terminate with a zero.

c. All lines of text must contain the same number of printable characters.

		DISPLAY
To enter the labels into the programmer from keyboard:		
1.	Select input format Label Printing	[Label Printing]
2.	Press INPUT.	[0]
3.	Enter label number (0 to 7). The display indicates	[0 00]
4.	Press ENTER. Current data DD.	[0 00 DD]
5.	Key in required data.	[0 00 DD RR]
6.	Press STEP->.	[0 01 DD]
	Repeat steps 5 and 6 until the label details are complete.	
7.	Press RST.	[27128]

TO ENTER THE LABELS INTO THE PROGRAMMER FROM RS232 OR EPROM

1. Load text into user RAM.

LABEL NUMBER	START ADDRESS	END ADDRESS
0	0000	003F
1	0040	007F
2	0080	00BF
3	00C0	00FF
4	0100	013F
5	0140	017F
6	0180	01BF
7	01C0	01FF

	DISPLAY
2. Select input format "label printing".	
3. Press INPUT.	[0]
4. Press ENTER.	[0 00]
5. Press READ. The contents of user RAM addresses 0000 to 01FF is copied into label RAM and saved.	[27128]

TO PRINT LABELS

1. Select output format "label printing".	
2. Press OUTPUT.	[0000]
3. Enter 4-digit print control word.	[1824]

1st digit = label number.
2nd digit = number of labels across page.
3rd & 4th digits = decimal count of number of labels to print.

4. Press ENTER. The required labels will be output to printer.

TO ENABLE SELECTIONS AND RAM EDITING WITH CODE LOCK (OPTIONAL)

1. Switch programmer off.

2. Press the E button while the programmer is switched on and hold until the display clears the 8's. The display shows 0000. [0000]

3. Enter security code through keyboard. [****]

4. Press ENTER. If the code is accepted the system will beep and display the device type. [27128]

```
**************************************
*                                    *
*                                    *
*     RAM KEY LOCK CODE - 0B0B       *
*   (unless otherwise specified)     *
*                                    *
*                                    *
**************************************
```

APPENDIX B, SERIAL COMMUNICATION FORMATS

Serial Formats - All output formats terminate with CONTROL Z.

1. ASCII HEX SPACE FORMAT

CHARACTER	DESCRIPTION
1	Block mark: "CONTROL A" & or "CONTROL B" used to identify the beginning of a block.
2 to N	Data: Each byte of data is represented by a two-character hex number followed by a space character. The high order character precedes the low order. The system reads only the two characters that proceeded a space. Therefore, the sequence: AB space, carriage return, line feed, CD space (Note: commas included to improve legibility - they are not transmitted) would result in the two two 8-bit words described by AB and CD being stored.
N + 1	Terminate mark: "CONTROL C" is used to identify the end of the block.

2. INTEL LOADER FORMAT (Standard and Extended 16 bit Address records)

1	Record mark: A colon is used to mark the beginning of a record.
2 - 3	Record length: A two-character hex representation of the number of bytes of data in the record, a length of zero (00) indicates an end-of-file. Character 2 is high order record length of characters.
4 - 7	Load address: A four-character hex address at which the first data bytes are loaded into successive (higher) memory locations. Character 4 is the high order address digit. In an end-of-file record, the load address is taken as the starting address.
8 - 9	Record type: A two-character hex code specifying the record type. All data records are type 00. The most significant digit is character 8. Extended address records are type 02.
10 - N	Data: Each byte of data is represented by a two-character hex humber. The high order character precedes the low order.
N + 1 - N + 2	Checksum: A two-character hex checksum, which is the negative sum of all bytes in the record except the colon and checksum, evaluated modulo 256. The sum of all bytes in the record plus the checksum must be zero.
N + 3	Carriage return
N + 4	Line feed

3. OPTIONAL

1 - N	Binary data

4. TEKTRONIX HEXADECIMAL FORMAT

1.	Record mark: A slash is to mark the beginning of a record.
2 - 5	Load address: A four-character hex address at which the first data bytes are loaded into successive (higher) memory locations. Character 2 is the high order address digit. In an end-of-file record, the load address is taken as the starting address.
6 - 7	Record length: A two-character hex representation of the number of bytes of data in the record. A record length of zero (00) indicates an end-of-file. Character 6 is the high order record length of character.
8 - 9	Header checksum: A two-character number representing the eight-bit sum, modulo 256 of the hex values of the six characters 2 - 7.
10 - N	Data: Each bytes of data is represented by a two-character hex number. The high order character precedes the low order.
N + 1 - N + 2	Data checksum: A two-character number representing the eight-bit sum, modulo 256, of the hex values of the digits that make up the N data bytes.
N + 3 - N + 4	Carriage return Line feed

5. MOS TECHNOLOGY

1	Record mark: A semicolon is used to mark the beginning of a record.
2 - 3	Record length: A two-character hex representation of the number of bytes of data in the record. A record length of zero (00) indicates end-of-file, Character 2 is the high order record.
4 - 7	Load address: A four-character hex address at which the first data bytes must be loaded. Ensuing data bytes are loaded into successive (higher) memory locations. Character 4 is the high order address digit. In an end-of-file record, the load address is taken as the starting address.
8 - N	Data
N + 1 - N + 6	Checksum: A four-character hex checksum which is the sum of all bytes in the record except the semicolon and checksum.
N + 5 - N + 6	Carriage return Line feed

6. MOTOROLA S RECORD

The programmer will action S0, S1, S2, S3, S7, S8 and S9 records.

where:- S0 is the header record.
S1 is data from 16 bit address record.
S2 is data from 24 bit address record.
S3 is data from 32 bit address record.
S7 is file terminator.
S8 is file terminator.
S9 is file terminator.

7. DEC BINARY

CHARACTER	DESCRIPTION
1 to (H-1)	Header: Binary value 1111 1111 is used to identify header.
H	Terminate header: Binary value 0000 0000 is used to identify last character of header.
D - N	Binary data characters: Terminates on selected device boundary.

8. BINARY

1	Header: Binary value 1111 1111
2 - N	Binary data characters: Terminate on selected device boundary.

9. BLOCK DUMP (output only)

Address	16 bytes hex dec. data	16 bytes ADCII data
AAAA	HH HH . . . . .HH HH	AA . . . . . .AA AA
"	" " " "	" "
"	" " " "	" "
"	" " " "	"
AAAA	HH HH . . . . .HH HH	AA . . . . . .AA

A. RCA COSMAC FORMAT

```
!M (CR)(LF)
AAAA HHHHHH . . . . . .HH;(CR)(LF)
AAAA HH . . . . . .HH,(CR)(LF)
HHHH  . . . . . .HH(CR)(LF)
```

!M	- Start character
AAAA	- Address field
HH	- Two hexadecimal digits
;	- End of record character if followed by expressed address
,	- End of record character (no address following)
(CR)(LF)	- Non-printing carriage return line feed
(CR)	- End of file (when a record terminates without a, or ;)

Set system variable inhibit return after !M value 04 to allow the address to be on the same line as start character, e.g., !MAAAA.

B. PPX - ASCII HEX SPACE

1	Block mark * used to identify the beginning of a block.
2,3	Carriage return Line feed
4 to 7	Load address: A four-character hex address at which the first bytes must be loaded. Ensuing data bytes are loaded into successive (higher) memory locations.
8	Space
9 to N	Data: Each byte of data is represented by a two-character hex number followed by a space character. The high order character precedes the low order. The system reads only the two characters that preceeded a space. Therefore, the sequence: AB space, carriage return, line feed, CD space (Note: Commas included to improve legibility - they are not transmitted) would result in the two 8-bit words described by AB and CD being stored.
N + 1, N + 2	Record terminate mark: Carriage return, Line feed, or
N + 1	Block terminate mark: Dollar

C. TEXAS TAGS

TAG CHAR.	HEXADECIMAL (FOUR CHARACTERS)	SECOND FIELD	MEANING	PRGMR
0	Length of all relocatable code	8-character program identifier	Program start	READ
1	Entry address	None	Absolute entry address	READ
2	Entry address	None	Relocatable entry address	IGNORED
3	Location of last appearance of symbol	6-character symbol	External ref. last used in relocatable code	IGNORED
4	Location of last appearance of symbol	6-character symbol	External ref. last used in absolute code	READ
5	Location	6-character symbol	Relocatable external definition	IGNORED
6	Location	6-character symbol	Absolute external definition	READ
7	Checksum for current record	None	Checksum	ACTIOND
8	Ignore checksum	None	Do not checksum for error	ACTIOND
9	Load address	None	Absolute load address	ACTIOND
A	Load address	None	Relocatable address	IGNORED

C. TEXAS TAGS (continued)

TAG CHAR.	HEXADECIMAL (FOUR CHARACTERS)	SECOND FIELD	MEANING	PRGMR
B	Data	None	Absolute data	ACTIOND
C	Data	None	Relocatable data	IGNORED
D	Load bias value	None	Load point specifier	IGNORED
F	None	None	End-of-rec.	ACTIOND
G	Location	6-character symbol	Relocatable symbol definition	IGNORED
H	Location	6-character symbol	Absolute symbol definition	READ

D. ASCII BNPF FORMAT

CHAR.	DESCRIPTION
1	START OF FIELD; "B" used to identify the begining of a word.
2 - 9	Each bit within the data word is represented by a "P" if a "1" or an "N" if a "0".
10	END OF FIELD. "F" used to identify the end of a word.
n	TERMINATE MARK: "ETX" (non-printable) used to identify the end of a block.

Any character except "B" or "ETX" after an "F" and before a "B" (eg. line feed, carriage return etc.) will be ignored on input. For output a carriage return and line feed are sent after every eighth word.

E. EXTENDED TEKTRONIK HEXADECIMAL FORMAT

HEADER FIELDS

1 Record Mark: A percent (%) specifies that the block is in extended TEKHEX format.

2 -3 Block Count: The number of ASCII characters in the block This count does not include the leading % or the end of line.

4 Block Type: 6 - Data Block
8 - Termination Block
3 - Symbol Block (not supported thus ignored)

5 -6 Checksum: A two digit HEX no. representing the sum, MOD 256, of all the characters in the block, excluding the leading %, the checksum digits and the end of line.

EXTENDED TEKHEX DATA BLOCK FORMAT

1 -6 Header: As defined above with a block type of 6.

7 -n Load Address: This is a variable length number, range is from 2 to 17 ASCII characters.

The first ASCII character indicates the length of the rest of the field. The value of 0 (zero) indicates a length of 16 ASCII characters.

N.B. Only the least 8 significant characters of the address are actioned.

n +1 DATA

Carriage Return
Line Feed

EXTENDED TEKHEX TERMINATION BLOCK FORMAT

1 -6 Header: As defined above with a block type of 8.

7 -n Transfer Address: The address where a program execution is to begin, a variable length no.

N.B. On input transfer address is ignored
On output transfer address is 0 (zero).

APPENDIX C, ASCII CHARACTER CODES

DEC	HEX	CHR	DEC	HEX	CHR	DEC	HEX	CHR
000	00H	NUL	043	2BH	+	086	56H	V
001	01H	SOH	044	2CH	,	087	57H	W
002	02H	STX	045	2DH	-	088	58H	X
003	03H	ETX	046	2EH	.	089	59H	Y
004	04H	EOT	047	2FH	/	090	5AH	Z
005	05H	ENQ	048	30H	0	091	5BH	[
006	06H	ACK	049	31H	1	092	5CH	/
007	07H	BEL	050	32H	2	093	5DH	]
008	08H	BS	051	33H	3	094	5EH	^
009	09H	HT	052	34H	4	095	5FH	_
010	0AH	LF	053	35H	5	096	60H	'
011	0BH	VT	054	36H	6	097	61H	a
012	0CH	FF	055	37H	7	098	62H	b
013	0DH	CR	056	38H	8	099	63H	c
014	0EH	SO	057	39H	9	100	64H	d
015	0FH	SI	058	3AH	:	101	65H	e
016	10H	DLE	059	3BH	;	102	66H	f
017	11H	DC1	060	3CH	<	103	67H	g
018	12H	DC2	061	3DH	=	104	68H	h
019	13H	DC3	062	3EH	>	105	69H	i
020	14H	DC4	063	3FH	?	106	6AH	j
021	15H	NAK	064	40H	@	107	6BH	k
022	16H	SYN	065	41H	A	108	6CH	l
023	17H	ETB	066	42H	B	109	6DH	m
024	18H	CAN	067	43H	C	110	6EH	n
025	19H	EM	068	44H	D	111	6FH	o
026	1AH	SUB	069	45H	E	112	70H	p
027	1BH	ESCAPE	070	46H	F	1132	71H	q
028	1CH	FS	071	47H	G	114	72H	r
029	1DH	GS	072	48H	H	115	73H	s
030	1EH	RS	073	49H	I	116	74H	t
031	1FH	US	074	4AH	J	117	75H	u
032	20H	SPACE	075	4BH	K	118	76H	v
033	21H	?	076	4CH	L	119	77H	w
034	22H	"	077	4DH	M	120	78H	x
035	23H	*	078	4EH	N	121	79H	y
036	24H	$	079	4FH	0	122	7AH	z'
037	25H	%	080	50H	P	123	7BH	(
038	26H	&	081	51H	Q	124	7CH	)
039	27H	'	082	52H	R	125	7DH	}
040	28H	(	083	53H	S	126	7EH	~
041	29H	)	084	54H	T	127	7FH	DEL
042	2AH	*	085	55H	U			

DEC - Decimal, HEX - Hexadecimal (H), CHR - Character
LF - Line feed, FF - Form feed, CR - Carriage return,
DEL -Rubout

APPENDIX D, E11 SIMULATOR

GENERAL

The E11 Simulator is designed to speed up program/hardware development by simulating the target system EPROM. Small program changes can be quickly made and instantly checked. A single E11 will simulate the current range of JDEC standard 24 pin and 28 pin devices, i.e., 2716, 2732, 2764, 27128, and 27256. Two E11s in a slave/master configuration provide twin EPROM simulation or 16-bit simulation.

Data is edited in the main programmer and transferred to the E11 via the 26-pin ribbon cable. Data from the E11 can be read back or verified with data in the main programmer.

Battery backup in the E11 supports the memory for approximately 10 minutes so that, if required, the E11 can be disconnected from the programmer and used separately.

-- 30-pin ribbon cable connects the E11 to the target system terminating in a 28-pin or 24-pin DIP plug. An additional 5-pin connector at the rear of the E11 has an optional Ov connection and RESET/RESET connections. Normally, RESET connection will be used to disable the target system when out of simulation and to synchronize the target system with the E11. Failure to use this control will almost certainly result in misoperation of the target system. Alternatively, a power up reset synchronization can be used without the reset connection (see "Operation without Reset Control").

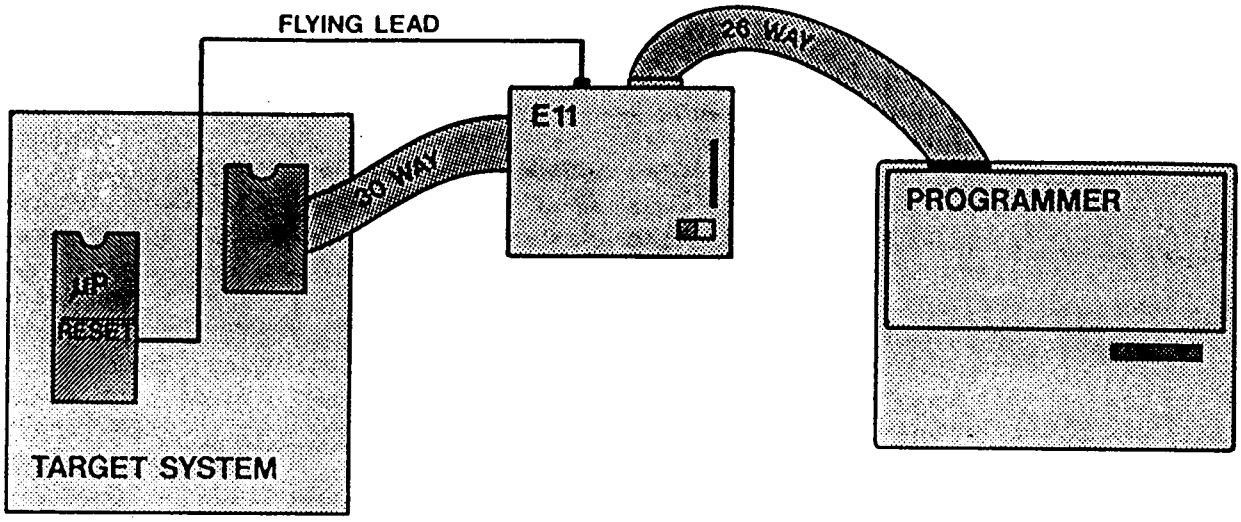
POWER

The E11 draws power from the main programmer or target system. Consumption is usually less than the EPROM replaced - typically 10mA standby, 45mA active. The RAM Nickel Cadium battery is automatically trickle-charged when the E11 is connected to the main programmer or target system. Approximately 10 minutes after the E11 has been disconnected from the target system and the main programmer, the battery support to RAM is automatically switched off.

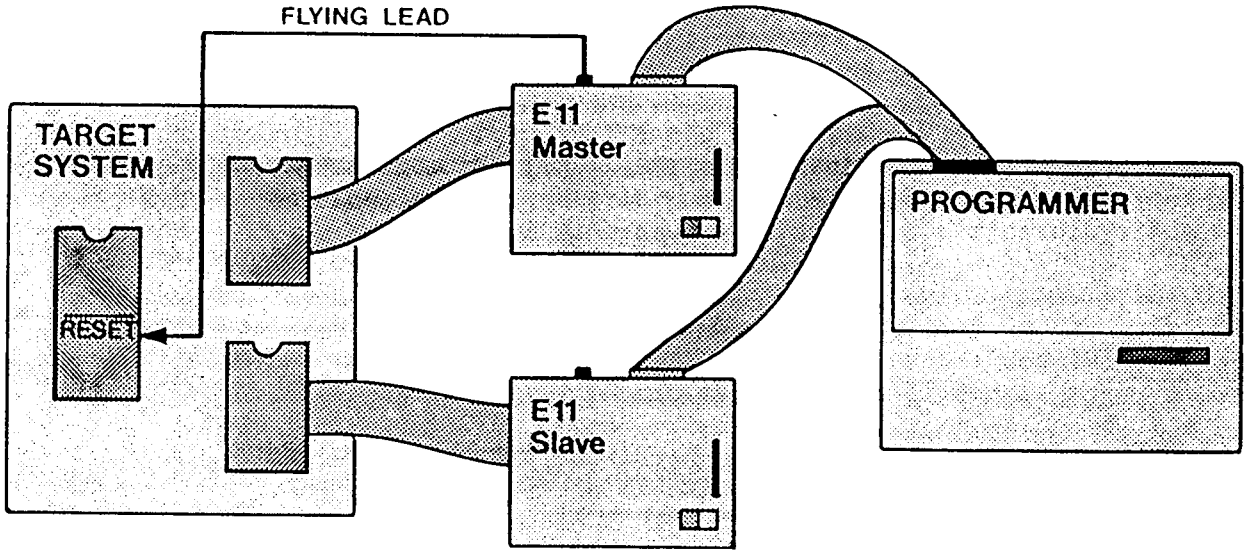
CONNECTIONS

26-Pin Connector at Rear of E11:

Connect to the 26-pin connector on the rear of the main programmer using the ribbon cable supplied. This connection can be made with or without the programmer powered up.



26 WAY CABLE : Connects to Programmer for Data Transfers
30 WAY CABLE : Connects to Target System eprom socket
1 FLYING LEAD : Micro-Processor control signal



Programmer selects Master or Slave for Data Transfers

E11 Master's pushbuttons (Reset and Simulate) controls Slave and Master E11's L.E.D. displays on both E11 units display device type and simulate status

One Flying Lead from MASTER is required to control µP

30-Pin Connector at side of E11 :

Connect to the target system EPROM socket using the 24-pin or 28-pin DIP plug cable supplied. Observe orientation and do not plug into a live system.

5-Pin Connector at Rear of E11:

	PIN NO.	SIGNAL	COMMENTS
Left hand pin	1	Ov	Not normally needed
	2		No connection
	3		No connection
	4	$\overline{\text{RESET}}$	) Always use one of these signals to control micro-processor in target system if system control is via E11.
Right hand pin	5	RESET	

SYSTEM SYNCHRONIZATION USING RESET/$\overline{\text{RESET}}$ CONTROL

The target system micro-processor must be halted or reset under control of the E11 for two reasons:

With the EPROM removed and the E11 in circuit but not simulating, the target system would try to run accessing a disabled tri-state buffer on the E11. The target system would therefore run in an indeterminate uncontrolled manner which may be potentially dangerous.

The RESET signal generated by the E11 is rather like a power on reset which normally initializes the micro-processor. Internally, tri-state address and data buffers are enabled and synchronized to the target system CE and OE signals.

Both RESET and $\overline{\text{RESET}}$ are open collector transistors pulled up to +5v with 4K7 resistors. When connecting to the target system, there may be conflict between the E11 reset signal and the target system reset signal if this is an active circuit. In this case, it may be possible to connect to the effective point of reset remote from the actual micro-processor reset pin. Always consult your circuit diagram before making RESET or $\overline{\text{RESET}}$ connections.

OPERATION WITHOUT RESET/$\overline{\text{RESET}}$ CONNECTION

Although the most useful and convenient method of operation involves the use of either a RESET or $\overline{\text{RESET}}$ connection, it is possible to obtain synchronization and correct running without this connection. This is achieved by putting the E11 into simulate mode before the target system is powered up. When the target system powers up, it's internal reset circuitry effects a power on reset which synchronizes it to the E11. However, the disadvantage of this method is that if amendments to the program are required and the E11 is taken out of simulate mode, the whole procedure of turning off the target system, re-entering simulate mode, and turning back on the target system must be carried out to effect resynchronization. It is therefore much more convenient to use the automatic reset/resynchronization provided by the reset or reset connection.

E11 CONTROLS AND INDICATORS

Simulate switch (SIM)	Puts the E11 into simulate mode. The reset signals are inactive, i.e. RESET is high; $\overline{\text{RESET}}$ is low. In this mode, data cannot be transferred between the E11 and the main programmer.
Reset mode (RST)	Takes the E11 out of simulate mode. The reset signals are active, i.e., RESET is low; $\overline{\text{RESET}}$ is high. In this mode, data cannot be simulated but data transfers between the E11 and the main programmer may take place.
Simulate indicator	When lit, indicates that the E11 is in simulate mode.
Device type indicator	When lit, indicates the selected simulated device type.

To conserve battery backup power, the LED indicators are only illuminated when the E11 is connected to a powered-up programmer or target system. If the E11 has lost data and device selection (i.e., if it has been disconnected from a source of power for periods greater than 10 minutes), then none of the indicators will be illuminated until this information is restored.

OPERATING PROCEDURE

A. Using the 26-pin ribbon cable connector, connect the E11 rear jack to the 26-pin jack at the rear of the programmer (the programmer may or may not be powered up).

b. With the target system switched off, remove EPROM to be simulated and insert either 24-pin or 28-pin DIP plug into empty socket. Observe orientation. Connect the other end of the DIP plug lead into the side jack on the E11.

c. Connect RESET or $\overline{\text{RESET}}$ to the micro-processor reset control (see Section 1.4 and 1.5 for guidance).

d. Write programmer data into E11 (see Section 1.9 for guidance).

e. Switch on target system.

f. Press simulate (SIM) button on E11.

To escape from simulate and restore the target system, push RST button then use the above steps from e, in reverse order.

CONTROL OF THE E11 FROM THE MAIN PROGRAMMER

The E Series programmers can carry out three functions with the E11 simulator.

a. Write data into E11.
b. Read data from E11.
c. Verify programmer RAM with E11 RAM.

To simplify the operation, data is always written to or read from start address zero in the programmer.

To enable E11 functions, press hex key "0" [Simulate]
The display will flash to indicate E11 mode (device type)

This gives the three RAM control keys VERIFY, INPUT, and OUTPUT the following double functions:

VERIFY - Verify programmer RAM with E11 RAM
INPUT - Read E11 RAM into programmer
OUTPUT - Write programmer RAM into E11 RAM

While a function is being carried out, the display indicates: [Sim Wait]

Press RST on programmer to exit from E11 mode [Device type]

APPENDIX D (continued)

WRITING DATA TO THE E11

a. Press hex key '0' to enable E11 functions	[Device type] [Simulate]
b. Press key OUTPUT. The programmer will write the correct amount of data for the selected device type from RAM start address zero to the E11 simulator.	[Sim Wait]
c. Press RST	[Simulate] (Device type)

READING DATA FROM THE E11

a. Press hex key '0' to enable E11 function	[Device type] [Simulate]
Press key INPUT. The programmer will read the correct amount of data for the selected device type from the E11 and place it in RAM starting at address zero.	[Sim Wait]
At the end of data transfer, the checksum is displayed.	[CCCC]
c. Press RST.	

VERIFYING DATA

Verify programmer RAM with E11 RAM

a. Press hex key '0' to enable E11 functions	[Device type] [Simulate]
b. Press RAM VERIFY key. The system verifies the two sets of data. The display indicates the function is being carried out.	[Sim Wait]
c. If the data verifies correctly, the display indicates End.	[End]
If the system has different data, the display indicates the address AAAA, RAM data MM and the E11 data dd.	[AAAA MM dd]
Press STEP> to continue verify check, or	

Press STEP< to display the data at the previous address, or

Press RST to terminate operation [Simulate]

EXAMPLE - SIMULATING A 27128 DEVICE

a. Select device type 27128

b. Load the data via RS232, or load master EPROM into programmer RAM start address zero.

c. Press hex key '0'.

d. Press key OUTPUT to write data to E11. When the transfer is complete, the display indicates the checksum.

[Simulate]
[27128]
[Sim Wait]
[CCCC]

e. Press RST.

f. Press E11 SIM button.

[Simulate]
[27128]

The target system will now run if powered up. See page D3 for operation with or without RESET/RESET control. Monitor the required target functions.

g. Press E11 RST button to terminate simulation.

h. Press RST on programmer.

i. Amend the memory address of the function and repeat Steps 3 to 9 as required.

TWO E11's IN MASTER/SLAVE CONFIGURATION

GENERAL

An additional E11 may be used as a slave unit (if internally linked), to extend the single EPROM socket simulate facility to two sockets.

When used in a master/slave configuration, the E11 slave pushbuttons are not used - control of reset or simulate being under control of the master E11. Accordingly, one single reset or reset connection is required from the master E11 to the target system.

A dedicated slave E11 can still be used independently in exactly the same way as a standard master E11 provided that slave data transfers have been selected on the main programmer.

DATA TRANSFERS TO MASTER OR SLAVE

The operating system of the main programmer assumes, unless commanded, that data transfers are to a master E11. To select the slave unit for data transfers, carry out the following procedures:

a. Enter simulate operating mode in usual way by pressing key "0" on hex keypad.

[Simulate flashing (device type)]

b Press ENTER key once on lower keypad. The 1 on the right hand side of the display indicates slave selection.

[Simulate flashing (device type) 1

c. Carry out write/read/verify in usual way.

To revert to master E11 data transfers, press ENTER key once; the "1" on the right hand side of the display will disappear, indicating master selection.

[Simulate flashing (device type)]

APPENDIX E, E4/E7 ADAPTER
OPERATING INSTRUCTIONS FOR E4 ADAPTER (8741/48/48H/49/55) & E7 ADAPTER (8744/51/52)

CAUTION: Irrevocable damage to the device may be caused if the following sequence is not strictly adhered to.

CAUTION:The E4 in standard form will program Intel 8741/48/48H/49/55. A modification is required on the E4 to program NEC 8748H and NEC 8749H parts and is available to special order.

1. Switch on E series programmer. Verify E series adapter is powered down.

2. Connect E series adapter with a 26-pin ribbon cable to the ribbon cable jack on the rear of the programmer.

3. Power up E series adapter. L.E.D. on the top of the unit should be illuminated.

4. Select device type (8748/49/55, etc.) on the E series programmer.

5. When the device type has been entered (i.e., when the display has stopped flashing the type), the corresponding device may be inserted into the 40-pin zero insertion force socket. Carefully observe orientation of the socket - reversal may result in instant destruction of the device.

6. Programming, reading, and verification may be carried out in the usual manner using the lower set of pushbuttons on the programmer. To program the security bit in the 8751, program the device, then when the display indicates end [End CCCC], press PROG and hold for two beeps. After this operation, it is impossible to reprogram, read, verify, or blank check the device until erased.

7. Only remove the device at the end of an operation. The RST button may be safely used to abort a programming, verification, or blank check operation.

8. Power down sequence is the reverse of power up. Ensure 40-pin socket is empty. Switch off power to adapter, and remove 26-pin ribbon cable from programmer.

9. Note that the unit will function normally with other EPROM types while the adapter is connected. The RS232 interface is operational, but the parallel output connection is utilized by the adapter.

10. Unlike normal EPROMs and the 8755, a "clean" 8748 or 8749 will have "00" in each location instead of "FF". To reduce unnecessary programming time, the programmer does not program blank locations with blank data and therefore a device with only a few locations of data and many blank locations will be programmed in a much shorter time than the maximum for a device which has data in all locations.

APPENDIX F, E5 ADAPTER

OPERATING INSTRUCTIONS FOR E5 ADAPTER (68701)

1. Switch on 'E' series programmer.

2. Switch on E5 Adapter. The L.E.D visible from the top of the unit should be illuminated.

3. Connect E5 Adapter to programmer via 26 way ribbon connector.

4. Select device type on programmer : 68701 E5.

5. When the device type has been entered (ie when the display has stopped flashing the type) the E5 Adapter is ready for use. CAREFULLY observe orientation of the 40 pin socket - incorrect insertion of the 68701 may cause damage to the device. The 40 pin socket is safely "powered down" in the idle mode.

6. Programming, Reading, Blank check and Verification may be carried out by using the LOWER set of pushbuttons on the programmer. These functions are similar in operation to normal EPROM programming except that data is always loaded into and read from RAM base address 0000 in the main programmer.

7. At the end of an operation the adapter is in the idle mode and it is then safe to remove or insert a device. The RST button on the programmer may be safely used to abort any operation and place the adapter in the idle mode. Under some situations the adapter may enter a "lock up" state and fail to initialize an operation. To escape press RST once and repeat the required operation.

8. Power down sequence is the reverse of power up.

   Ensure 40 pin socket is empty
   remove 26 way ribbon cable
   remove power supply from the adapter

9. Note that the main programmer will function normally with other EPROM types whilst the adapter is connected. The RS232 interface is operational but the parallel output connection is utilized by the adapter.

10. Unlike normal EPROMS the 68701 has '00' in unprogrammed locations instead of the usual 'FF'. To reduce unneccessary programming time the programmer does not programme blank locations or locations which are already programmed with identical data to RAM and therefore a device with only a few locations of data to change will be programmed in a much shorter time than the maximum for a device which has data to be programmed in all locations.

APPENDIX G, E13 ADAPTER

OPERATING INSTRUCTIONS FOR E13 ADAPTER (2704,2708,2716 3 RAIL)

CAUTION

Irrevocable damage to the device may be caused if the following sequence is not strictly adhered to:-

1. Connect the adapter to 26 pin ribbon cable socket at the right hand rear of the programmer.

2. Switch on E Series Programmer. The L.E.D visible from the top of the E13 should be illuminated.

3. Select device type (2704, 2708, 2716 3 RAIL) on the main programmer.

4. When the device type has been entered (i.e. when the display has stopped flashing the type) the corresponding device may be inserted into the 24 pin zero insertion force socket on the E13 adapter. Carefully observe orientation of the socket - reversal may result in destruction of the device.

5. Programming, Reading and Verification may be carried out in the usual manner using the lower set of push buttons on the main unit.

6. Only remove the device at the end of an operation. The RST button may be safely used to abort a programming, verification or blank check operation.

7. Power down sequence is the reverse of power up:
   Ensure 24 pin ZIF socket is empty.
   Remove 26 way connector from the main unit when it is switched off.

8. Note that the programmer will function normally with other EPROM types whilst the adapter is connected. The RS232 interface is operational but the parallel output connection is utilized by the adapter.

EASYCOM Remote Driver (Elan Asynchronous Communications Program)- OPTION

"EAsyCom" is an optional program for controlling any E-Series EPROM editing programmer or editing copier, the C41 programmer or Universe 1000, all manufactured by Elan Digital Systems. The program operates on IBM PCs (TM) and compatible computers under the MS-DOS Operating System versions 2.0 or higher.

The organization of these instructions corresponds to the steps to take in operating the EASYCOM program. First the Elan programmer must be connected to the computer, the power turned on, and the programmer put in a mode that will enable it to communicate with the EASYCOM program. These steps are described under 'Set up Programmer'. Then the EASYCOM program is initiated, as described under 'Start Program'. After the initial EASYCOM informational screens have been displayed, all control of the Elan programmer takes place from the Programmer Control Screen.

Set Up Programmer

Set up the input/output communications configuration of the Elan programmer. The serial port should be transmitting and receiving at 4800 baud rate, no parity, 8 data bits and 1 stop bit. Use the I/O code setting 2CC for E Series or EA Series programmers, or I/O code setting 8 Db 1 St no/par for EB Series programmers or the C41 programmer. The data format to use will be set depending on the type of data file to be transmitted, using the Programmer Control Screen, and need not be set up at this time.

Before initiating EASYCOM, the Elan Programmer must be connected to the computer, its power must be "ON", and its initial diagnostic sequence must have been completed. A device type (e.g. "2764") must be displayed without blinking on the Elan Programmer's display. The programmer must be put into the remote control mode by holding down the "ENTER" key before EASYCOM will operate correctly. This will result in a display "REM" in the rightmost characters of the LCD in the EB or C41 programmers, or a display of "C" in the rightmost character of the E or EA Series programmers. Connect the Elan programmer's serial I/O to port COM1 on the computer. See appendix I.

EASYCOM will return control of the Elan programmer to the manual mode when you have terminated its activity and it exits to the operating system.

Start Program

Before loading EASYCOM you must first boot up your operating system. Then place the EASYCOM disk in the appropriate disk drive and start the program by typing EASYCOM at the MS-DOS prompt (>). You will see a copyright notice followed by directions to press function keys F1, F2 or F9. Use of function key F2 takes the program directly to the Elan programmer control screen and should only be used by operators already familiar with the use of EASYCOM. Users not experienced in the use of this program should depress

function Key F9 and proceed through the instruction screens as directed. If you have entered EASYCOM in error, press F1 to exit back to the Operating System.

The Elan programmer will emit one or more "beeps" if F2 is depressed, indicating it is receiving information from EASYCOM.

If the "No Programmer Response" error appears when an attempt is made to first enter the programmer control screen, this indicates that the Elan programmer is not properly connected to the computer, it is not turned on, or it is not in a mode that allows it to communicate with the EASYCOM program (it may not be in Remote Control Mode, or it may be executing a command or performing start-up diagnostics).Check that the SYSTEM VARIABLES are set to zero.(page 2-3).

Use of various control and function keys

Enter "Carriage Return"
Press this key when the contents of the current field are correct. The program advances to the next field after Enter is pressed.

Function Key F1 "Previous Screen/Exit"
"EASYCOM" Pressing this key displays the screen previously examined. Exit from the intial copyright screen or from the Programmer Control Screen.

Function Key F2 "Previous Field"
Displays the next screen to be examined.

Function Key F3 "Previous Field"
Selects the field immediately after the current one.

Function Key F5 "Previous Choice"
Press once to select the previous value in the current field. Hold down to scroll the previous values rapidly.

Function Key F6 "Next Choice"
Press once to select the next value in the current field. Hold down to scroll next values rapidly.

Function Key F8 "Clear Field"
Clears the current field if it is an entry field.

Use of various control and function keys (Cont.d)

Function Key F9 "Help"
Displays the "Help" Screen.

Function Key F10 "Do" or Execute"
Causes the Elan programmer to execute the function indicated in the "Programmer Function" field. This field doesn't need to be the current field when this Function Key is pressed. The programmer "beeps" when this key is pressed.

Pg Up "First Field"
The first field on the programmer control screen is selected.

Pg Dn "Last Field"
The last field on the programmer control screen is selected.

Entry Field Editing
If the current field is an entry field, the cursor control keys on the numeric keypad can be used to edit or control the field's contents.

Left arrow "Cursor left"
Cursor moves one character to the left in the current field.

Right arrow Curosr right"
Cursor moves one character to the right in the current field.

Del "Delete"
Deletes the character at the current cursor position.

Ins "Overwrite"/"Insert"
Changes from the Overwrite to Insert mode, or vice-versa. This controls the action that occurs when a character is typed at the cursor position. In the "Overwrite" mode, the typed character replaces the character at the cursor position; in the "Insert" mode, the typed character moves all the other characters in the field to the right without replacing any. The "default" condition when "EAsyCom" is entered is the "Overwrite" mode.

Backspace
Deletes the character to the left of the cursor.

Home
Moves the cursor to the first character of the current field.

End
Moves the cursor to the last character of the current field.

APPENDIX I, SERIAL COMMUNICATIONS CABLE ELAN/IBM

ELAN SERIAL I/O connector in a CANNON 'D' type 25 way plug

IBM PC & IBM XT COMM1 connector in a CANNON 'D' type 9 way plug. (use converter cable IBM part no. 0745 769 to convert to 25 way plug).

Interconnecting Cable with 25 way CANNON 'D' type socket at each end, reversable pin interconnections

1	connect to	1
2	"	3
3	"	2
4	"	5
5	"	4
6	"	20
7	"	7
20	"	6

AFTER SALES SUPPORT AND SERVICE

In case of operating difficulties (and before making any returns) please contact:-

For U.K. and rest of the world (excluding U.S.A.):

i) Your Distributor or

ii) Customer Support Engineering at Elan Digital Systems Ltd
Tel: (0293) 510448

For U.S.A. the Technical Support Representitive, Tel: (408) 734 2226

Advice can be given on all aspects of the programmer's operation and the problems encountered when interfacing with other systems.

In the event of a return being necessary please use the original packing material or pack very carefully to minimise damage in transit. Equipment received in inadequate packing will be returned in new packing which will be charged for.