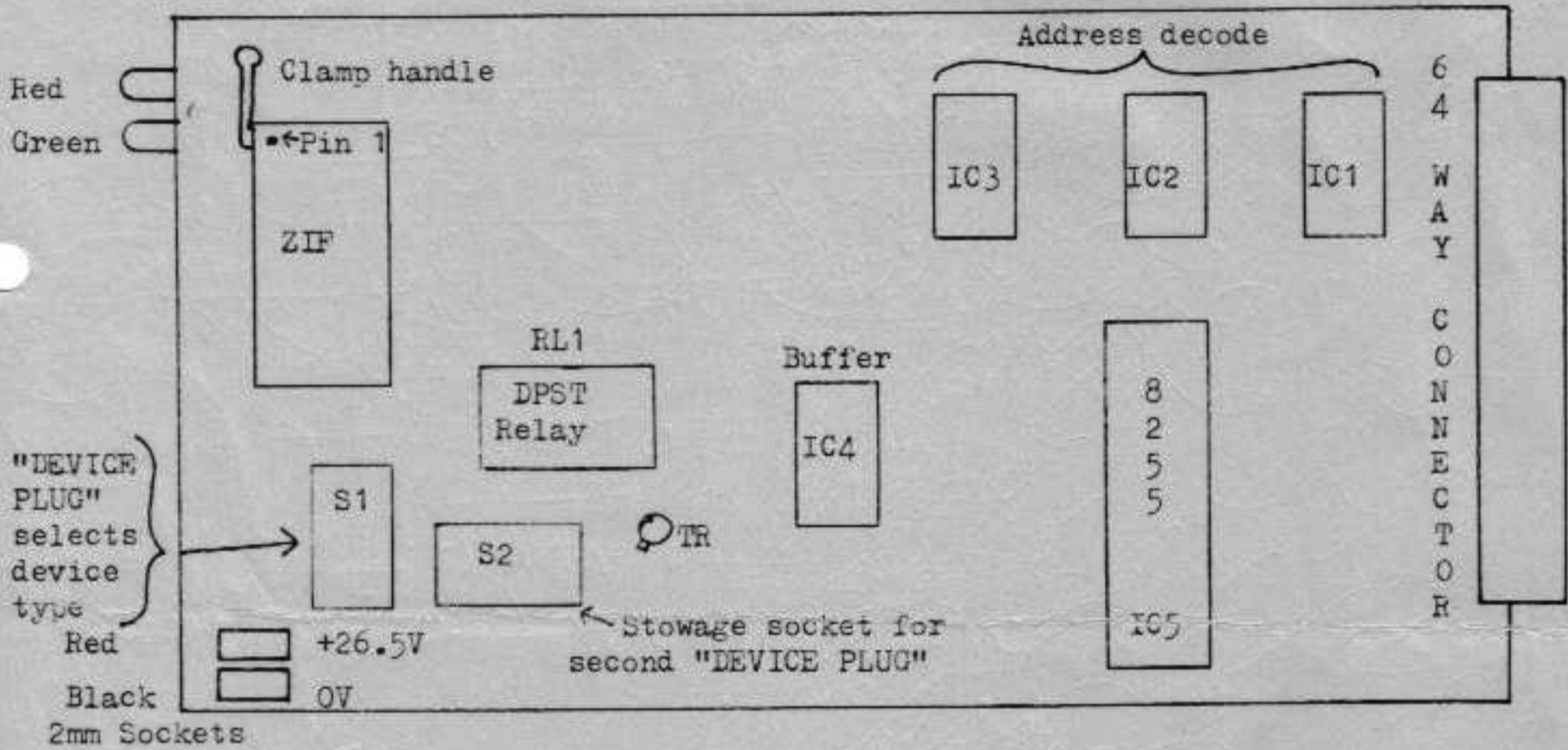


MICROTANIC EPROM PROGRAMMER

Introduction

The board and associated eprom based software provide the capability to program four types of single supply eproms, viz:- 2516, 2716, 2532 & 2732. Also included is the facility to enter routines/data etc into "blank" areas of previously programmed devices.

Board Layout



The board is supplied with two 14 pin DIL "DEVICE PLUG" headers marked on the top as follows:-



With a "DEVICE PLUG" inserted in S1, the legend at the top refers to the eprom type to be dealt with. The 'R' and 'P' after '2532' on the second plug refer to eprom read operations (ie Blank check, copy & Verify operations) and the program operation respectively.

Notes:-

- 1) No harm will befall a 2532 device if programming is attempted with the "DEVICE PLUG" in the read position and vice versa.
- 2) BUT programming of a 25/2716 or 2732 device with the "DEVICE PLUG" inserted the wrong way round will result in the destruction of the eprom.

DOUBLE CHECK BEFORE PROGRAMMING

Installation

Insert the board into one of the 'additional' motherboard slots and the 2K eeprom into the spare location (E2) on Tanex, or into one of the sockets on the appropriate Microtanic 8K Eeprom Extension Boards for Tanex E2. Connect a 26.5 volt DC supply, capable of 50mA, to the red and black 2mm sockets, (+26.5V & 0V respectively), using the ready made cable supplied.

Some hardware notes

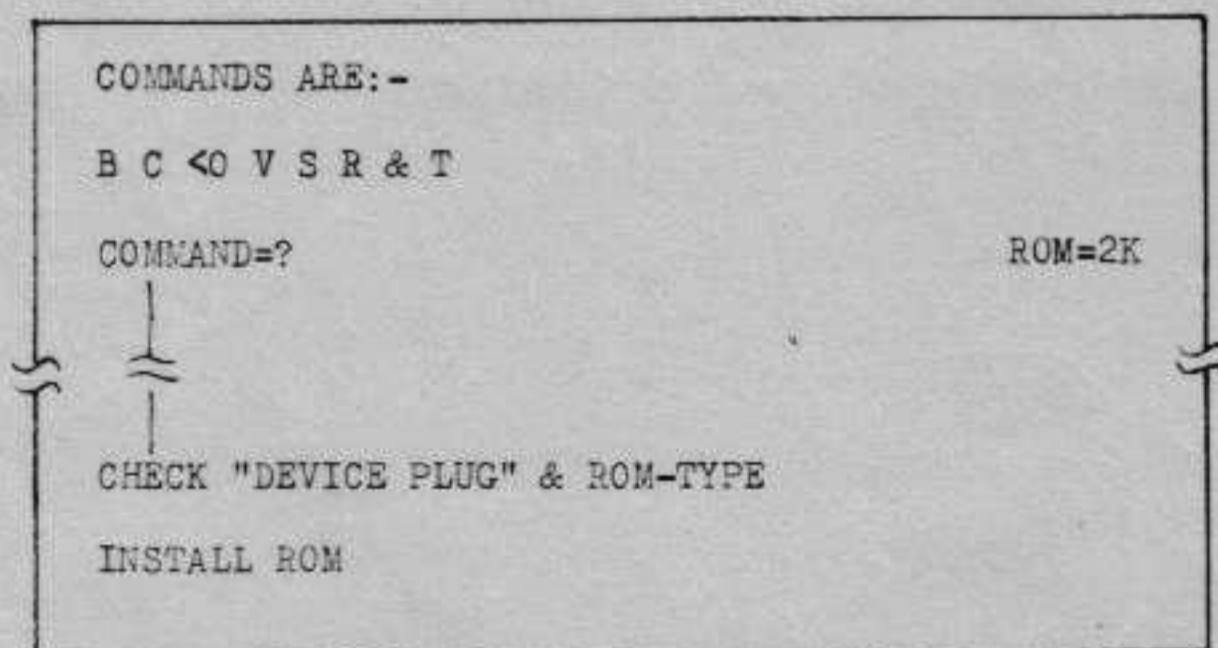
- 1) The specified programming voltage (Vpp) for all the eeproms the unit is capable of programming is 25.0V (26.5V absolute maximum).
- 2) 26.5V input is necessary due to the 'volts drop' across the transistor TR when it is conducting.
- 3) The red LED lights whenever the transistor TR is switched on, to allow 25.0V to the zero-insertion force (ZIF) socket for programming, AND the +5V is applied.
- 4) The green LED lights whenever +5V is applied to the ZIF.
- 5) The ZIF is only powered (i.e. +5V or +5V and 25V applied) for the duration of a command operation (see 'Command Descriptions'); it is safe to insert/remove an eeprom at all other times.
- 6) A system reset leaves an eeprom installed in the ZIF completely safe.

Some software notes

- 1) The board is located at B000-B003 hex in the Tangerine I/O memory map.
- 2) The software uses zero-page locations E8-FA, and 400-BFF (2K) or 400-13FF (4K) on Tanex to copied data or data to be programmed.
- 3) The 50ms programming pulse is produced by Timer 2 of the first 6522 VIA located at A2 on Tanex.

Getting it going

Entry to the programmer software is at E800 and the following will appear on the screen:-



Notes:-

- 1) The '<' symbol is used to denote a control character key, in this case control 0.
- 2) <0 is used since <P is already used in Tanbug V2.3.
- 3) When the '?' flashes a command may be entered, a return to that state shows that execution of the command operation selected is complete.
- 4) The program defaults to the 2K device on entry.

It is suggested that a 'dry-run' of the command operations be performed before actually doing anything with an eeprom installed in the ZIF.

Command descriptions

B = BLANK CHECK - checks that all eeprom locations are un-programmed, i.e. at FF hex, and lists the address of any location that is not.

NB. Since the screen configuration can only display 7 addresses, any more than this will be scrolled. This is not really a problem since an eeprom with this many 'poor' locations would be un-useable anyhow.

C = COPY - copies eeprom contents into RAM on Tanex.

<O = PROGRAM - programs the eeprom with data stored in RAM, i.e. from 400-BFF (2K) or from 400-13FF (4K).

NB. If any sections of the eeprom will remain unused, leave the 'spare' locations at FF hex to facilitate later entry of routines into these areas using the 'S' command.

V = VERIFY - checks the eeprom contents against data in RAM, can be used after a 'COPY' as well as after a 'PROGRAM' command.

NB. The eeprom address, contents of that address and the corresponding byte of RAM data of any differences/errors are listed. If there are more than 7, the message "M TO CONT" (i.e. press the M key to list the next set) will appear. Pressing 'M' displays the next 7 differences/errors. The message will be displayed until they have all been listed, signified by the 'V' after "COMMAND=" being replaced by the flashing "?". Successful verification is indicated by the response "OK".

This routine can be used to check the differences between one version of an eeprom and another.

R = ROM CHANGE - changes the software action to accommodate 4K eeproms, pressing

'R' again reverts back to 2K. After pressing 'R' the user is prompted to:-
"CHECK "DEVICE PLUG" & ROM-TYPE"
"INSTALL ROM"

NB. On program entry the software defaults to 2K.

S = SINGLE STEP PROGRAM - allows programming of individual locations or block

of locations (up to 255 bytes), after entry of the required eeprom start address and the data to be programmed into that location/block of locations. The data is entered serially and as hex op-codes/operands.

This facility can be used for entering routines into spare un-programmed sections of previously programmed devices.

The following errors will be flagged on address/data entry:-

- a) "NOT HEX!" - if any non-hex character is entered.
- b) "NOT IN RANGE!" - 1) if the start address entered is not within the device range i.e. 0-7FF for 2K and 0-FFF for 4K eeproms.
2) if the start address plus the number of bytes of data entered exceeds the range of the device as in 1) above.
- c) "NO DATA!" - if programming is attempted before any data has been entered.

Notes:-

- 1) A '<E' at any time after entry of the start address causes a return to 'command mode', "E" being displayed at the bottom of the screen.
- 2) A 'CR' response to "ROM START ADDR?" results in a zero address being entered.
- 3) A 'CR' response at data entry causes a zero byte to be entered and the "BYTES" counter to increment, but ~~0~~ will not appear on the screen although the display will scroll.
- 4) Entering more than 2 characters for a data byte causes only the last 2 to be stored, as in HEXPACK.
- 5) '<O' causes the entered data to be programmed, verification is automatically performed afterwards, any errors being listed as for the 'V' command.

If it is necessary to go back to Tanbug during single step data entry, this can be simply achieved by returning to the 'command mode' with a "<E" and then to the monitor with a "T" command, (see "T" command description).

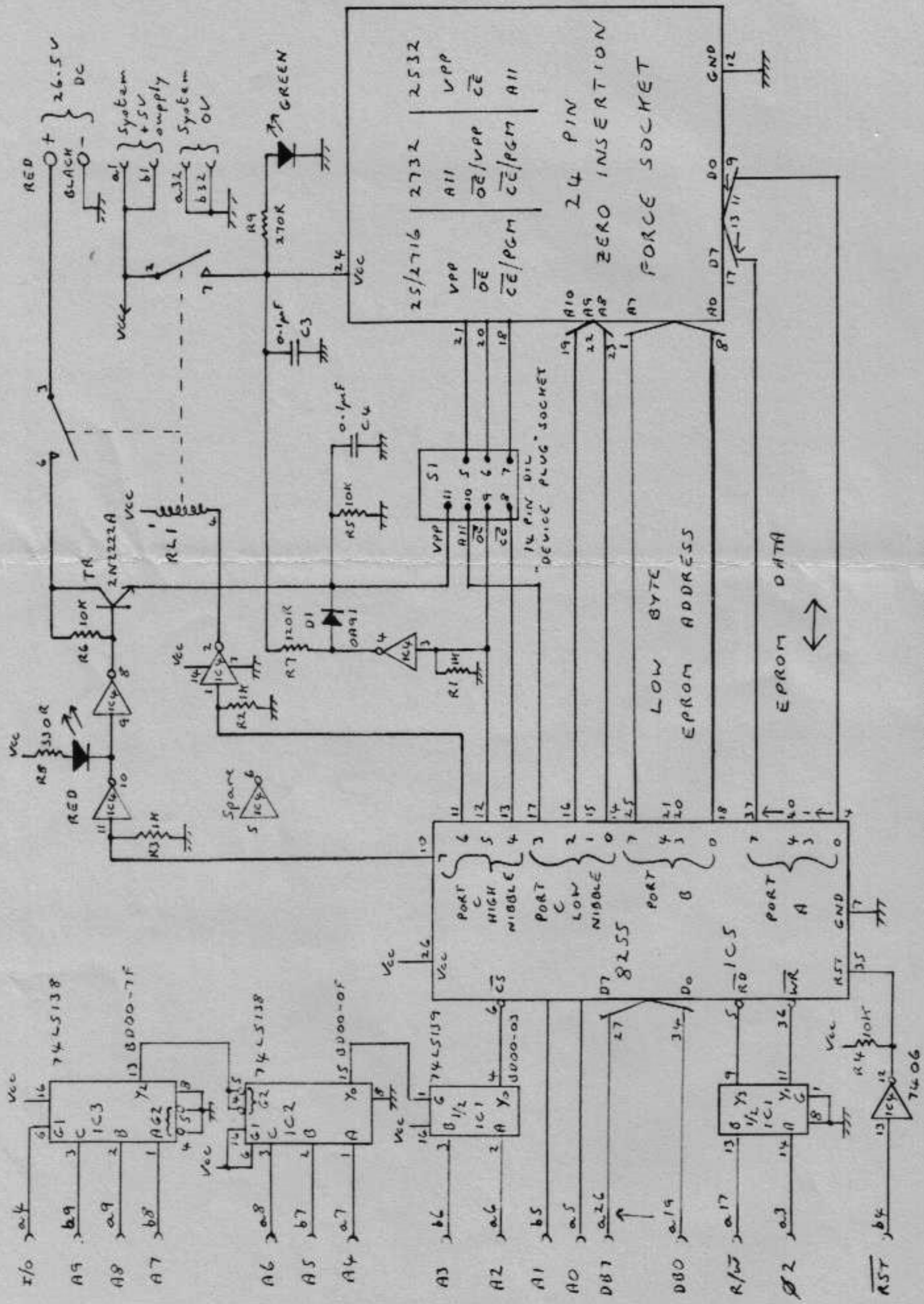
Re-entry to the single step routine may be accomplished by a GEA62, this returns to exactly the same state as at exit, provided that locations E8-FA in zero-page have not been corrupted.

T = TERMINATE - causes a return to Tanbug at FC4B, after clearing the screen and displaying "T".

NB. The stack pointer will be at the same value as at program entry.




Programming procedure example - for a 25/2716 or 2732 device.



- 1) Get data to be programmed into RAM from 400 upwards.
This could be:-
 - a) from cassette, in which case ensure that the program is relocated to reside at the address at which the eprom will finally 'live'. This could be done using either the Microtanic 'Relocate program' or the '2-Pass Assembler'.
 - b) downloaded from a system resident piece of firmware using the Tanbug 'copy' facility.
 - c) copied into RAM from an eprom installed in the ZIF using the programmer 'COPY' command.
 Assuming case a) or b) above:-
 - 2) Enter the programmer software at E800.
 - 3) Check that the "ROM-TYPE" and "DEVICE PLUG" are correctly set.
 - 4) Insert 'blank' eprom into ZIF, pin 1/notch in case to the top, and clamp in position i.e. clamp handle flush with board.
 - 5) Do a "BLANK" check, "OK" will be displayed if all locations are erased.
 - 6) Check correct connection of the +26.5V supply.
 - 7) Press '<O', both green and red LED's will light.
Programming takes 1M 43Sec for 2K and 3M 26Sec for a 4K device. Completion is indicated by both LED's going out and the response "OK".
 - 8) Verify with a 'V' command, "OK" indicating successful verification.
- NB. If programming a 2532 device remember to change round the "DEVICE PLUG" before step 7 and back again before verification in step 8 (see note 1 under Board Layout).



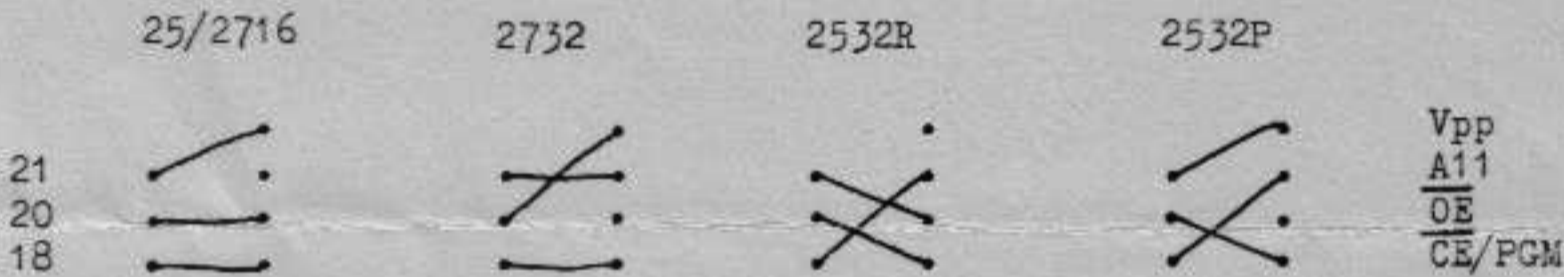
MICROINAMIC EPROM PROGRAMMER

"DEVICE PLUG" links

	ZIF pin no.	25/2716	2732	2532
R E A D	18	\overline{CE}/PGM VL	\overline{CE}/PGM VL	A11 A11
	20	\overline{OE} VL	V_{pp}/\overline{OE} VL	\overline{CE}/PGM VL
	21	V_{pp} VH	A11 A11	V_{pp} VH
P R O G R A M	18	\overline{CE}/PGM 	\overline{CE}/PGM 	A11 A11
	20	\overline{OE} VH	$V_{pp}/\overline{OE} +25V$	\overline{CE}/PGM 
	21	$V_{pp} +25V$	A11 A11	$V_{pp} +25V$

Notes:-
 \overline{CE}/PGM = not chip enable/program (active low)
 \overline{OE} = not output enable (active low)
 V_{pp} = programming voltage
 A11 = address line A11
 /  = 50mSec programming pulse
 VL = logic '0'
 VH = logic '1'

Links in header plugs as seen from above when correctly orientated:-



Designation of 8255 Peripheral Interface Adaptor (PIA) ports

	I/O ADDRESS
PORT A = ZIF DATA	BDO0
PORT B = ZIF ADDRESS A0-A7	BDO1
PORT C = ZIF ADDRESS A8-A11	BDO2
PORT C L	
PC4 = program pulse	
PC5 = \overline{OE}	BDO2
PORT C U	
PC6 = ZIF +5V control	
PC7 = ZIF V_{pp} control	
8255 MODE CONTROL	BDO3

INCLUSION OF MICROTANIC MACHINE CODE RELOCATOR PROGRAM

The 'MACHINE CODE RELOCATOR' software has now been included in the same eeprom as the 'Programmer', to further enhance the capabilities of the unit.

The instructions for use are as per the original software except for the following changes:-

- 1) Start address = EDDF
- 2) A return to Tanbug may be made at any time with a '<T' (control T) input.
- 3) The program returns to Tanbug after the actual relocation has been performed, successful or otherwise.
- 4) Successful relocation is indicated by the response "OK".