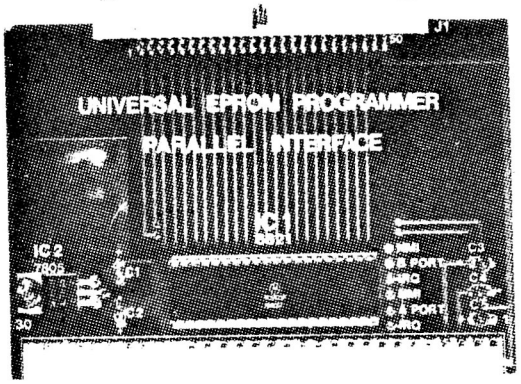
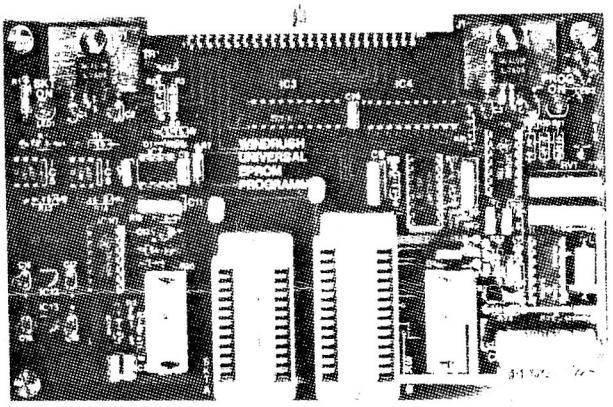
WINDRUSH

ALL-IN-TWO

UNIVERSAL EPROM PROGRAMMER





1.0 INTRODUCTION

The Windrush Universal EPROM Programmer has been designed to program the most common (and some not so common) EPROMS in general use. A parallel interface board is supplied which plugs directly into any of the S-30 slots. The programmer board is extended to the work area via a four foot IDC cable. This arrangement precludes the necessity of requiring continuous access to the interior of the Micro-computer which often hampers the forced air cooling. The EPROM programmer board can also be interfaced with a standard 6821 PIA elsewhere in the system if the S-30 bus is not available.

2.0 APPLICATIONS

The range of BYTE WIDE EPROMS covered by the programmer and it's associated software are as follows:

2508 (single voltage) (1K)	2532 (4K)	2764 (8K)
2708 (triple voltage) (1K)	2732 (4K)	2528 (16K)
2516 (single voltage) (2K)	2732A (4K)	
2716 (triple voltage) (2K)	2564 (8K)	

WINDRUSH UNIVERSAL EPROM PROGRAMMER

TABLE OF CONTENTS

3.0 SOFTWARE

The software provided with this programmer has been developed to be as user kindly as possible. Consequently the program requires just over 6K of RAM to run. A fairly large proportion of this requirement (4K) is consumed by the various messages and prompts that are presented to the operator during use.

The memory map of the software, as provided, is as follows:

$0000 - $3FFF	EPROM BUFFER AREA	(16K)
$4000 - $5840	PROGRAMMER SOFTWARE	(Just over 6K)
$5880 - $58E0	SCRATCHPAD RAM	($60 BYTES)

The software is not relocatable in it's binary form and is not re-entrant. The software can be relocated by changing the two ORIGIN statements at the beginning of the listing and re-assembling the source files. The software can reside in PROM provided that the required RAM is located elsewhere.

4.0 DISC OPERATING SYSTEMS

The software can be supplied for a variety of disc operating systems as follows:

MICROWARE OS9 (level 1 and level 2)
SSB DOS68 V 5.1 ($6000, $A000, $C000) (CHIEFTAIN & BFD-LFD VERSIONS)
SSB DOS68D V 6.0 on up
SSB DOS69 V 1.0 on up
SSB DOS69D V 2.0 on up
TSC FLEX 2.0
TSC FLEX 09
TSC UNIFLEX

5.0 DISC FILES

Five files will be provided on the disc supplied with the EPROM programmer. The file names are as follows:

A. UPRMQ .TXT (EPROM PROGRAMMER EQUATES FILE)
B. UPRM1x.TXT (EPROM PROGRAMMER SOURCE FILE PART ONE)
C. UPRM2x.TXT (EPROM PROGRAMMER SOURCE FILE PART TWO)
D. UPRM3x.TXT (EPROM PROGRAMMER SOURCE FILE PART THREE)
E. UPROM .BIN (EPROM PROGRAMMER OBJECT CODE FILE)
F. UPROM .CMD (UPROM.$ for SSB DOS)....(EPROM PROGRAMMER COMMAND FILE)

The 'x' will be replaced by a letter code (A - Z) which represents the current version of the EPROM programmer software. As modifications or enhancements are made to the software this letter code will be incremented.

The source file has been split into an equates file and three text files in order to facilitate editing. Most editors cannot accomodate the entire source file in one lump.

SSB DOS 5.1 users must use either the 'A' DOS or the 'C' DOS when editing the source files. DOS at $6000 does not provide enough RAM for the source files.

View (list) a file named UPROM.INF for further information.

6.0 BEFORE YOU START

Before you attempt to instal this software on your system, make a working copy of the disc we have supplied using your DOS 'COPY' or 'BACKUP' command. Store the original disc in a safe place. Once you have made a working copy do not use the original disc for anything whatsoever except to make another working copy.

WE CANNOT BE HELD RESPONSIBLE FOR FATAL DAMAGE TO DISCS DUE TO OPERATOR ERROR OR HARDWARE FAULTS, NOR WILL WE. WE DESIRE TO MAKE MONEY BUT DO NOT WISH TO DO SO BY SELLING YOU UN-NEEDED COPIES OF THE SOFTWARE PACKAGE.

7.0 HARDWARE DEPENDANT SYSTEM VECTORS

In order to cope with the considerable variation in 6800 and 6809 development systems the software uses DOS routines whenever possible. The software has been structured to includes a jump table at the beginning of the program. This enables the software to be reconfigured by simply patching the binary file. This is by far the easiest way to adapt the software to your system. This method will be discussed in detail in the next section.

We have also supplied the source files for the programmer software. This will enable you to add additional routines to suit your particular needs.

THE FOUR VECTORS AND ONE ADDRESS USED BY THE PROGRAMMER SOFTWARE ARE DOS AND HARDWARE DEPENDANT. THE ADDDESSES GIVEN IN THIS SECTION ARE PRESENTED FOR GUIDANCE ONLY. ALL ADDRESSES SHOULD BE CHECKED AGAINST THE DOCUMENTATION SUPPLIED WITH YOUR COMPUTER SYSTEM.

7.1 INCH

This is the routine that looks at the keyboard in the system terminal and returns the keyboard character in the A register. We recommend using the DOS routines as this will make the software a bit more portable. The vectors for various Disc Operating Systems are detailed in the table below.

SMOKE		SIGNAL		BROADCASTING		TSC	
DOS68 $6000	DOS68 $A000	DOS68 $C000	DOS68D $C000	DOS69 $C000	DOS69D $C000	FLEX 2.0 $A000	FLEX 09 $C000
$72C4	$B2C4	$D2C4	$D2C4	$D2C4	$D2C4	$AD15	$CD15

This routine is called 'ZGETCH' in SSB DOS and does not provide an 'ECHO'. This necessitates a 'JSR' instruction in the jump table rather than a 'JMP' instruction. This routine is called 'GETCHR' in TSC FLEX and will automatically ECHO. This requires a 'JMP' instruction in the jump table. For further details see section 8.1.

7.2 OUTCH

This is the routine that outputs the character in the A register to the system terminal. Again we recommend using the DOS vectors given in the table below.

DOS68 $6000	DOS68 $A000	DOS68 $C000	DOS68D $C000	DOS69 $C000	DOS69D $C000	FLEX 2.0 $A000	FLEX 09 $C000
$72C1	$B2C1	$D2C1	$D2C1	$D2C1	$D2C1	$AD18	$CD18

This routine is called 'ZOUTEEE' in SSB DOS and 'PUTCHR' in TSC FLEX.

7.3 MONITOR RESTART

One of the options in the 'OPERATIONS MENU' is to return control of the system to the system monitor. We recommend using the DOS vectors given below.

DOS68 $6000	DOS68 $A000	DOS68 $C000	DOS68D $C000	DOS69 $C000	DOS69D $C000	FLEX 2.0 $A000	FLEX 09 $C000
$728D	$B28D	$D28D	$D28D	$D28D	$D28D	$B3B5	$D3F3

The addresses given above include the +1 offset required to use the SSB DOS jump table 'ZMON' as a vector (the programmer software uses an indexed jump rather than a direct jump through the DOS jump table). The vector for FLEX 2.0 has been confirmed for SSB versions only. If you have any difficulty in locating this vector in your FLEX 2.0 DOS refer to the instructions in the disc file entitled 'UPROM.INF'

7.4 DOS WARM START

Another of the options in the 'OPERATIONS MENU' is to return control of the system to the Disc Operating System. The various DOS vectors are as follows:

DOS68 $6000	DOS68 $A000	DOS68 $C000	DOS68D $C000	DOS69 $C000	DOS69D $C000	FLEX 2.0 $A000	FLEX 09 $C000
$7283	$B283	$D283	$D283	$D283	$D283	$AD03	$CD03

7.5 PIA PORT ADDRESS

Most users of our EPROM programmer will be using it in conjunction with the S-30 interface board supplied. The base addess of the S-30 I/O section varies between $8000 (SWTP, MSI and GIMIX 6800 systems), $E000 (SWTP and GIMIX 6809 systems), $F7E0/$F780 (SSB 6800 & 6809 systems). The number of addresses per port can also vary between 4 and 16.

7.5 PIA PORT ADDRESS (continued)

In order to cope with the vast array of actual addresses covered by the aforementioned hardware variations the EPROM programmer software will prompt you for the base address of the 6821 PIA. Use the tables below or the documentation supplied with your system to acertain the actual base address of a particular S-30 I/O port.

S-30 BUS WITH FOUR ADDRESSES PER PORT

BASE ADDRESS	PORT 0	PORT 1	PORT 2	PORT 3	PORT 4	PORT 5	PORT 6	PORT 7
$8000	$8000	$8004	$8008	$800C	$8010	$8014	$8018	$801C
$E000	$E000	$E004	$E008	$E00C	$E010	$E014	$E018	$E01C
$F7E0	$F7E0	$F7E4	$F7E8	$F7EC	$F7F0	$F7F4	$F7F8	$F7FC

S-30 BUS WITH SIXTEEN ADDRESSES PER PORT

BASE ADDRESS	PORT 0	PORT 1	PORT 2	PORT 3	PORT 4	PORT 5	PORT 6	PORT 7
$8000	$8000	$8010	$8020	$8030	$8040	$8050	$8060	$8070
$E000	$E000	$E010	$E020	$E030	$E040	$E050	$E060	$E070
$F780	$F780	$F790	$F7A0	$F7B0	$F7C0	$F7D0	$F7E0	$F7F0

INTERFACING WITH A PIA

If you are considering interfacing the EPROM programmer board with a PIA elsewhere in the system ensure that the PIA addresses stack up as follows:

BASE ADDRESS + 0: 'A' SIDE DATA PORT
BASE ADDRESS + 1: 'A' SIDE CONTROL PORT
BASE ADDRESS + 2: 'B' SIDE DATA PORT
BASE ADDRESS + 3: 'B' SIDE CONTROL PORT

NOTE: ENSURE THAT ALL THREE UNREGULATED SUPPLIES ARE BROUGHT OUT TO THE PROGRAMMER BOARD. THE SYSTEM RESET LINE SHOULD ALSO BE BROUGHT OUT TO THE PROGRAMMER BOARD. THIS WILL ENSURE THAT THE LATCHES CLEAR ON HARDWARE RESET AND SYSTEM POWER UP.

8.0 PATCHING THE SOFTWARE

8.1 Make a working copy of the disc we have supplied using your DOS 'COPY' or 'BACKUP' commands if you have not already done so. See section 6.0.

USE THE WORKING COPY FOR ALL SUBSEQUENT WORK. UNDER NO CIRCUMSTANCES USE THE ORIGINAL DISC AS A WORKING DISC. AN ACCIDENT COULD BE EXPENSIVE!

8.2. Insert the work disc you have just created into your work drive. Use the DOS 'GET' command to load the binary file entitled 'UPROM .BIN'.

GET 1:UPROM .BIN (SSB DOS SYNTAX ASSUMING WORK DISC IS IN DRIVE # 1)

GET 1.UPROM .BIN (FLEX DOS SYNTAX ASSUMING WORK DISC IS IN DRIVE # 1)

8.3 EXIT to the system monitor.

E <CR> (SSB DOS SYNTAX)

MON <CR> (FLEX DOS SYNTAX)

8.4. Use the system monitor memory examine and change command, 'M' in most system monitors, to inspect the jump table which starts at $4000. This table is structured as follows:

$4000 7E 4012 JUMPS OVER THE FOLLOWING TABLE (DO NOT ALTER!)

$4003 BD 0000 'INCH' (See section 7.1)

$4006 7E 0000 'OUTCH' (See section 7.2)

$4009 7E 0000 'DOS WARM START' (See section 7.4)

$400C 0000 'MONITOR RESTART' (INDEXED JUMP) (See section 7.3)

$400E 0008 TRI-VOLT EPROM PROGRAMMING REST TIME (See section 8.1)

$4010 0008 5 VOLT EPROM PROGRAMMING REST TIME (See section 8.1)

8.5 Use the examine and change command to patch the jump table with the appropriate addresses given in the sections indicated.

8.6 Return to DOS. (JUMP TO THE DOS WARM START VECTOR INDICATED IN SECTION 7.4)

8.7 Use the DOS 'FIND' ('MAP') command to find the starting and ending addresses of the file entitled (UPROM.BIN).

8.8..Use the DOS 'SAVE' command to save the contents of memory between the two memory locations indicated in 8.7 above.

SAVE UPROM.$ 4000 5FFF 4000 (SSB DOS SYNTAX, SAVES THE FILE TO DRIVE 0)

SAVE UPROM.CMD 4000 5FFF 4000 (FLEX DOS SYNTAX, SAVES THE FILE TO DRIVE 0)

8.9 This completes the patching. You should now have a command called 'UPROM' which can be executed from DOS. Proceed to section 9.0

8.1 NOTES ON PATCHING AND SYSTEM OPERATION

One point to keep in mind if your system documentation indicates different addresses for the various DOS routines is that SSB DOS uses a JUMP table (the address of each routine is preceeded by '7E') and that TSC DOS uses a VECTOR table (only the routine addresses are in the table).

There are two distinct differences in the jump tables for the EPROM PROGRAMMER software configured for SSB DOS and TSC FLEX. The instruction at $4003 is a 'JSR' ('BD') for SSB DOS and is a 'JMP' ('7E') for TSC FLEX. These differences cater for the fact that the SSB DOS 'ZGETCH' routine does not ECHO whilst the TSC FLEX 'GETCHR' automatically provides an ECHO.

The programmer software uses two count values to establish the rest time between programming pulses. The values used ($0008) have been selected to provide a 300 microsecond rest time between pulses in a 2 MHZ system. This count value provides the fastest programming operation possible consistant with the manufacturers recommendations. If your system clock runs slower than 2 MHZ these count values can be decreased to optimise the programming time.

IMPORTANT NOTICE ON DUTY CYCLE LIMITATIONS

The WINDRUSH UNIVERSAL EPROM PROGRAMMER has been designed for use in development systems. It is NOT intended for production programming of EPROMS. Ensure that the progammer is allowed to cool for at least two minutes between programming operations. Alternatively you can increase the count values for the two programming rest periods to $0FFF which will significantly increase the programming time but will also prevent misuse and possible damage to the programmer.

9.0 SYSTEM OPERATION

The software used to drive the EPROM programmer is driven by two menus which prompt the operator for an activity or information. This approach will enable you to master the many functions of the software with very little effort. If an error occurs the software will print an error message and re-prompt you for the information. Some errors will return you to the OPERATION MENU.

All addresses, address ranges, and byte counts must be given in HEX. Responses to YES - NO questions are 'Y' for YES and any other key for 'NO'. If you make a mistake during data entry or wrongly select a particular operation a carriage return <CR> will return you to the operation select menue. Backspaces are not allowed during data entry.

One activity that has been intentionally left outside of the EPROM programmer software is loading binary files into RAM from disc. This activity was considered to be beyond the scope of the programmer software package. If you intend to program an EPROM from data that is stored on disc you should use your DOS 'GET' command to load the binary file into RAM. You can load it where it would normally run providing it does not fall into the region between $4000 and $58E0, or you can load it with an offset so that it loads directly into the EPROM programmer buffer area which starts at $0000 and ends at $3FFF.

9.1 STARTING UP

Turn off the power to your system. Plug the S-30 interface board into the designated I/O slot. Plug the extension cable into the interface board and the programmer ensuring the connectors are oriented correctly. Turn on the power and 'BOOT' up your system. Type in the command 'UPROM' followed by a carriage return <CR>. If you have used a different name when you saved the file, type in the name you have given to the command file.

A startup banner and the prompt 'ENTER BASE ADDRESS OF INTERFACE PIA' will appear on the terminal. If it does not appear check the address you have specified for 'OUTCH'. Type in the base address of the PIA followed by a carriage return. The prompt 'HIT ANY KEY TO CONTINUE' will appear on the terminal.

Hit any key on the keyboard. Two things should happen at this point. First the 'EPROM TYPES' menu should appear on the terminal, second the 'SKT ON' LED on the EPROM programmer board should illuminate for 1-3 seconds. If neither of these responses takes place check the address you have specified for 'INCH'. If the 'EPROM TYPES' menue appears but the LED does not illuminate check the base address you have specifed for the PIA.

9.2 EPROM TYPES MENU

The EPROM TYPES MENU provides a convenient method for EPROM type selection. Simply look up the EPROM type number in the table and type in the number (0-9) adjacent to it. The software will immediately prompt you to set up the switches on the EPROM programmer board and to place an EPROM in the programmer socket.

NOTE: You do not have to insert the EPROM at this point if you do not desire to do so. If you do not insert the EPROM at this point do not insert it until the software prompts you to do so during the 'COPY', 'VERIFY' or 'PROGRAM' operations.

9.2 EPROM TYPES MENU (continued)

A quick note is in order on 2716 EPROMS. As you are probably aware there is a bit of confusion in the industry over this type number. We have opted to use the TEXAS INSTRUMENTS designations. If you have a 2716 EPROM that only requires a 5 Volt supply, i.e. it is not a TRI-VOLT type, select the '2516' EPROM (number 2 in the MENU).

9.3 OPERATIONS MENU

The operations menu comes up with a banner at the top which indicates the EPROM type you have selected and the starting and ending addresses of the EPROM as related to the buffer area.

There are ten operations available. Each operation will be discussed in detail in the following sections.

To select a particlar operation simply type in the number (0-9) adjacent to the description of the operation you require.

When you have selected a particular operation a banner will appear confirming the operation has been selected. The software will then prompt you for further information or provide further instructions.

If an error is made during data entry one of three error messages will appear:

1. <*> NOT A HEX NUMBER <*> appears if any key other than 0 - 9 or A - F is pressed.

2. <*> THE ADDRESS INPUT IS OUT OF RANGE! RE-INPUT. <*> appears if either the FROM/START or the TO/END address specified falls outside of the buffer area.

3. <*> BUFFER LIMITS EXCEEDED! CHECK ADDRESSES. <*> appears during the 'MOVE' operation if the requested move will overflow the buffer.

9.3.0 FILL BUFFER AREA WITH A HEX CHARACTER

This operation is provided to fill the EPROM buffer area between $0000 and $3FFF with a specified HEX byte.

The software will prompt you for the HEX FILL BYTE, the STARTING ADDRESS, and the END ADDRESS.

a. The HEX FILL BYTE must be within $00 - $FF.
b. The STARTING ADDRESS must be within $0000 -$3FFF.
c. The END ADDRESS must be within $0000 - $3FFF and must also be higher than or equal to STARTING ADDRESS.

Two errors can occur during data entry:

1. <*> NOT A HEX NUMBER <*>
2. <*> THE ADDRESS INPUT IS OUT OF RANGE! RE-INPUT. <*>

9.3.0 FILL BUFFER AREA WITH A HEX CHARACTER (continued)

If no errors occur during data entry the 'FUNCTION COMPLETED' message and the prompt 'HIT ANY KEY TO CONTINUE' will appear. Pressing any key will return you to the OPERATIONS MENU.

9.3.1 MOVE DATA TO THE BUFFER

This operation allows you to move up to 16K of code from any part of the 64K memory map to the buffer area between $0000 and $3FFF.

The software will prompt you for a FROM ADDRESS, a TO ADDRESS and the NUMBER OF BYTES to be moved.

a. FROM ADDRESS can be anywhere between $0000 and $FFFF.
b. TO ADDRESS must be between $0000 and $3FFF.
c. NUMBER OF BYTES must not exceed the limits of the buffer area. ($4000 bytes)

Note: The number of bytes to be moved when added to the TO ADDRESS must not exceed $4000. The number of bytes to be moved must be specified in HEX. $0000 = 0 bytes moved, $0800 = 1K bytes moved, $1000 = 2K bytes moved, etc.

Three errors can occur during data entry:

1. <*> NOT A HEX NUMBER <*>
2. <*> THE ADDRESS INPUT IS OUT OF RANGE! RE-INPUT. <*>
3. <*> BUFFER LIMITS EXCEEDED! CHECK ADDRESSES. <*>

If no errors occur during data entry the 'FUNCTION COMPLETED' message and the prompt 'HIT ANY KEY TO CONTINUE' will appear. Pressing any key will return you to the OPERATIONS MENU.

9.3.2 COPY A PROM TO THE BUFFER

This operation enables you to copy a MASTER PROM into the buffer. You may then VERIFY other PROMS against the buffer, PROGRAM other PROMS from the buffer, or return to DOS and save the buffer area onto disc.

You can also alter the buffer area by returning to the OPERATIONS MENU and using the MOVE operation, FILL BUFFER operation, or the EXAMINE & CHANGE operation after you have copied the MASTER PROM. This enables minor modifications to PROM'd code to be accomplished with ease.

This operation will prompt you to place the MASTER EPROM into the socket and then ask you if you are ready to continue. A 'Y' (YES) response will result in the EPROM being copied into the buffer, any other key being pressed will return you to the OPERATIONS MENU.

Once the copy of the EPROM has been made the 'FUNCTION COMPLETED' message and the prompt 'HIT ANY KEY TO CONTINUE' will appear. Pressing any key will return you to the OPERATIONS MENU.

No error messages will appear during this operation.

NOTE: THE 'SKT ON' LED WILL ILLUMINATE BRIEFLY DURING THIS OPERATION.

9.3.3 VERIFY A PROM AGAINST THE BUFFER

This operation compares the contents of a PROM against the data in the buffer. The data in the buffer can originate from a MASTER PROM using the 'COPY' operation, can be moved to the buffer area by using the 'MOVE' operation, or can be loaded into the buffer area beforehand by using the DOS 'GET command.

This operation will prompt you to place an EPROM into the socket and then ask you if you are ready to continue. A 'Y' (YES) response will result in the EPROM being compared with the data in the buffer, any other key being pressed will return you to the OPERATIONS MENU.

Once the EPROM has been verified the 'FUNCTION COMPLETED' message and the prompt 'HIT ANY KEY TO CONTINUE' will appear. Pressing any key will return you to the OPERATIONS MENU.

The only error message than can occur during this operation is:

<*> VERIFY FAILED AT ADDRESS $XXXX

This message indicates the first address at which the data in the buffer is not identical to the data in the PROM.

NOTE: THE 'SKT ON' LED WILL ILLUMINATE BRIEFLY DURING THIS OPERATION.

9.3.4 PROGRAM A PROM FROM THE BUFFER

This operation enables you to program an erased or partially erased EPROM from data contained in the buffer. The data in the buffer can originate from a MASTER PROM using the 'COPY' operation, can be moved to the buffer area by using the 'MOVE' operation, or can be loaded into the buffer area beforehand by using the DOS 'GET command.

Initially this operation will prompt you to place an EPROM in the socket and then ask you if you are ready to continue. A 'Y' (YES) response will result in the address range prompts appearing, any other key being pressed will return you to the OPERATIONS MENU.

NOTE: DUE TO SPECIAL PROGRAMMING REQUIREMENTS TRI-VOLT EPROMS (2708 & 2716) DO NOT HAVE AN ADDRESS RANGE PROMPT.

The address range prompts allow to program the entire EPROM or just a specific portion of it. The 'PROGRAM EPROM' banner indicates which EPROM type has been selected and what it's full address range is. The software will prompt you for a START and END address. The EPROM will only be programmed between these limits.

a. The START ADDRESS must fall within the range indicated in the banner.
b. The END ADDRESS must also fall within the range indicated in the banner and must not be lower than the START ADDRESS.

Two errors can occur during data entry:

1. <*> NOT A HEX NUMBER <*>
2. <*> THE ADDRESS INPUT IS OUT OF RANGE! RE-INPUT. <*>

9.3.4 PROGRAM A PROM FROM THE BUFFER (continued)

If no errors occur during data entry the software will verify that the EPROM is erased between the limits specified. IF all addresses that are to be programmed are verifed to be erased the software will inform you of this and ask you if you want to START PROGRAMMING. If you answer NO (any key) you will be returned to the OPERATIONS MENU. If you answer 'Y' (YES) the software will initiate the programming cycle.

If the EPROM is not erased at any address between the specified limits the following message, which indicates the first address that is found not to be erased, will appear:

<*> PROM NOT ERASED AT ADDRESS $XXXX

PROGRAM ANYWAY? (Y/N)

If you answer NO (any key) you will be returned to the OPERATIONS MENU. If you answer 'Y' (YES) the software will initiate the programming cycle.

When the pprogramming cycle has been initiated the message 'PROGRAMMING IN PROGRESS' will appear.

Once programming is complete the software will automatically verify the entire EPROM (not just the range programmed) against the buffer. If all is well the message 'PROGRAMMING & VERIFICATION COMPLETED' will appear. If all is not well the following message, which indicates the first address that verify failure occured, will appear:

<*> VERIFY FAILED AT ADDRESS $XXXX

The software will then prompt 'HIT ANY KEY TO CONTINUE'. Pressing any key will return you to the OPERATIONS MENU.

NOTE: DURING THE ERASE CHECK THE 'SKT ON' LED WILL ILLUMINATE BRIEFLY. ONCE THE PROGRAMMING SEQUENCE IS INITIATED THE 'SKT ON' LED AND THE 'PROG ON' LED WILL ILLUMINATE. ONCE PROGRAMMING IS COMPLETE BOTH LED'S WILL EXTINGUISH THEN THE 'SKT ON' LED WILL ILLUMINATE BRIFELY ONCE AGAIN AS THE PROGRAMMED EPROM IS VERIFIED AGAINST THE BUFFER

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* CAUTION: OBSERVE DUTY CYCLE LIMITATIONS. (SEE PAGE 7) *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

9.3.5 EXAMINE OR CHANGE THE BUFFER

This operation will provide a MEMORY EXAMINE and CHANGE function very similar to the one your system monitor gives you. Hitting the 'SPACE' bar will increment the address being examined, hitting 'U' or '^' will decrement the address being examined. The contents of the memory location being examined may be changed by typing in the desired HEX byte.

Selecting this operation will prompt you for a STARTING address. The STARTING address can be anywhere between $0000 and $FFFF, although you would normally only use this operation within the buffer area between $0000 and $3FFF.

9.3.5 EXAMINE OR CHANGE THE BUFFER (continued)

Only one error can occur during the prompt for the starting address or an attempt to modify a memory location:

<*> NOT A HEX NUMBER <*>

Hitting carriage return <CR> at any point will return you to the OPERATIONS MENU.

9.3.6 FORMATTED DUMP OF THE BUFFER

The formatted dump requires a terminal with 80 characters per line. Any HEX byte within the range of $20 to $7E will have the equivalent ASCII character displayed to the right of the HEX code. Any code that falls outside of this range will result in a '.' being displayed

Selecting this operation will prompt you for a STARTING address. The STARTING address can be anywhere between $0000 and $FFFF, although you would normally only use this operation within the buffer area between $0000 and $3FFF.

Once a START address is entered 256 bytes will be dumped and you will be prompted for 'MORE?'. Hitting the 'Y' (YES) key will dump the next 256 bytes. Hitting any key other than 'Y' will return you to the OPERATIONS MENU.

9.3.7 RETURN TO EPROM SELECT MENU

Selecting this operation will return you to the EPROM SELECT (TYPES) MENU described in section 9.2.

9.3.8 EXIT TO SYSTEM MONITOR

Selecting this operation will return control of the system to the system monitor via the MONITOR RESTART VECTOR.

9.3.9 RETURN TO DISC OPERATING SYSTEM

Selecting this operation will return control of the system to the DOS via the DOS WARM START VECTOR.

10.0 HOW IT ALL WORKS

The Windrush Universal Eprom Programmer comprises two PCB assemblies. One is an S-30 Parallel Interface board, the other is the EPROM PROGRAMMER itself.

10.1 PARALLEL INTERFACE BOARD

This board contains a standard 6821 PIA and simply brings out all of the PIA signals, the system RESET line, +/- 12 Volt unregulated supplies and the +5 Volt unregulated supply to a 50 way IDC connector.The 50 way IDC connector has been arranged to accomodate twisted pair planar cable. This allows the programmer board to be extended by up to ten feet without necessitating buffers. An on board regulator supplies regulated +5 Volts to the PIA. On board tantalum bead capacitors pre-filter the D.C. supplies for the programmer board.

Link pads have been provided for NMI and IRQ interrupts. The software we supply does not require any interrupts.

10.2 EPROM PROGRAMER BOARD

This board is best described by dividing it into the three sections: POWER SUPPLY SWITCHING, PROGRAM PULSE GENERATION & LOGIC LEVEL CONTROL, and EPROM SOCKETS.

10.2.0 POWER SUPPLY SWITCHING

Refer to the schematic diagram on sheet 3 of the drawings which follow this section.

+5 VOLTS (UNSWITCHED)

This supply is provided by IC2 in conjunction with C3, C4, C6 and C26. (C26 is on sheet 4). This supply provides power to IC3, IC4, IC7, IC8, IC9, and IC10.

MISSING PULSE DETECTOR

IC7, a 555, is configured as a missing pulse detector. The time constant of R7/C11 is on the order of 1-3 seconds. Whenever the software is strobing IC4 via the 'CA2' line of the PIA the output at pin 3 of the 555 will be at +5 Volts (Logic 1). The software will only be strobing the 'CA2' line during EPROM read and programming operations. When the software is not performing these operations the output of the 555 will drop to 0 Volts (Logic 0) after a delay of 1-3 seconds. The normal steady state of the 555 output is 0 volts. This keeps the four switched supplies turned off.

Note: Worth mentioning is that 'CA2' is strobed when the PIA is initialized. This is the reason that you must never insert an EPROM into the sockets until the software prompts you to do so.

10.2.0 POWER SUPPLY SWITCHING (continued)

+12 VOLTS (SWITCHED)

The unregulated +12 Volt supply is switched via IC5, an opto isolator. Whenever the 555 output goes to +5 Volts the internal LED will illuminate causing the internal darlington transistor to switch on. The unregulated supply is then regulated by IC13 in conjunction with C18 and C19.

-5 VOLTS (SWITCHED)

The unregulated -5 Volt supply is switched via IC6, an opto isolator. Whenever the 555 output goes to +5 Volts the internal LED will illuminate causing the internal darlington transistor to switch on. The unregulated supply is then regulated by IC12 in conjunction with C16 and C17.

+5 VOLTS (SWITCHED)

The +5 Volt supply is switched by TR1 via IC10. Whenever the 555 output goes to logic 1 IC10 will invert the signal and pull down the base of TR1 towards 0 Volts forward biasing it via R2. R3 shunts the base leakage current of TR1 and assists in turn off. The incoming unregulated supply is then regulated by IC1 in conjunction with C1, C2, and C5. The unregulated switched supply also illuminates LED1 ('SKT ON') informing the operator that power is applied to the EPROM socket.

+21 AND +25 VOLTS (SWITCHED)

The unregulated +5 Volt supply that is switched by TR1 also supplies IC11, a switching regulator. The +5 Volt supply is stepped up to +21/25 Volts by IC11 and the surrounding components. RV1 and RV2 are factory set to provide +21 VOLTS and +25 VOLTS at point 'F' during programming. These adjustments must be set with an accurately calibrated oscilloscope with an EPROM actually being programmed to ensure that the voltage drops of TR2 and D2/D4 will be compensated for under normal load conditions.

SWA '10' and SWB '10' determine whether the output of the switching regulator is +21 or +25 Volts. With both switches open the output will be +21 Volts, with either switch closed the output will be +25 Volts.

The +21 Volt supply is required to program 2732A and 2764 EPROMS manufactured by INTEL and second sourced by several other manufacturers.

10.2.1 PROGRAM PULSE GENERATION & LOGIC LEVEL CONTROL

Refer to the schematic diagram on sheet 4 of the drawings which follow this section.

ADDRESS & MODE LATCHES

IC3 and IC4 are two data latches which hold the data present on PA0 through PA7 whenever the software strobes CA2 or CB2 respectively. The primary function of these devices is to generate thirteen address lines for the ERPOMS from the eight incoming PIA data lines. These latches also generate a PROGRAM/VERIFY line when A13 is not used (e.g. 2708 & 2716 EPROMS), provide a trigger for the program pulse monostable and controls the +25 Volt programming supply.

PROGRAM PULSE GENERATION

IC8 is a dual CMOS monostable, chosen for it's high stability and high input impedence which enables accurate long duration pulses to be generated with relatively small components. The two monostables are enabled via pins 3 & 13 whenever a program pulse is required. The two monostables are triggered via pins 4 & 12. The output at pin 6 is a 1 Millisecond pulse whilst the output at pins 9 & 10 is a 50 Millisecond pulse. The driving software monitors the pulse transitions via PIA lines CA1 and CA2 respectively.

IC9 and the surrounding components generate a 1 Millisecond 21/25 programming pulse for TRI-VOLT EPROMS or a 21/25V D.C. level for all other types depending on the status of the 25V ON/OFF line.

IC10 and the surrounding components control the 50 Millisecond pulse and a +12 volt level for TRI-VOLT EPROMS.

10.2.2 EPROM SOCKETS

Refer to the schematic diagram on sheets 5 & 6 of the drawings which follow this section.

These drawings detail the connections to the EPROM sockets and the various signals controlled by the MODE SELECT switches SWA & SWB. This arrangement precludes the costly 'PERSONALITY MODULES' that other EPROM programmer manufacturers use to compensate for the general lack of standardization between EPROM manufacturers. It's amazing that they all managed to get VCC and GROUND on the same pins!

11.0 PARALLEL INTERFACE BOARD PARTS LIST

ITEM	QUAN	REF.	DESCRIPTION
1	1	PCB	PRINTED CIRCUIT BOARD
2	3	-	10 Way Molex 2145-10AG
3	1	-	40 Pin IC Socket
4	1	C1	4u7 x 25 Volt Tant Bead
5	1	C2	4u7 x 25 Volt Tant Bead
6	1	C3	4u7 x 25 Volt Tant Bead
7	1	C4	4u7 x 25 Volt Tant Bead
8	1	C5	4u7 x 25 Volt Tant Bead
9	1	IC1	6821 PIA
10	1	IC2	7805 5 V, 1 A Regulator
11	1	J1	50 Way IDC 3M 3433-5302

INTERFACE CABLE ASSEMBLY PARTS LIST

ITEM	QUAN	REF.	DESCRIPTION
1	2	-	50 Way IDC 3M 3425-6000
2	2	-	Strain rlf 3M 3448-3050
3	6'	-	50 Way Twist-N-Flat

EPROM PROGRAMMER BOARD PARTS LIST

ITEM	QUAN	REF.	DESCRIPTION
1	1	PCB	PRINTED CIRCIT BOARD
2	1	-	10 Gauge Alum plate
3	4	-	3/8" x 4 BA ALL THD
4	4	-	1/4" x 4 BA C/S
5	4	-	1/4" x 4 BA R/H
6	4	-	12.7 mm Stick on feet
7	2	-	3/8" x 4 BA R/H (Brass)
8	2	-	4 BA Nut (Brass)
9	2	-	4 BA Flat Washer (Brass)
10	2	-	4 BA Star Washer (Cad)
11	2	-	6 pin IC SKT
12	1	-	8 pin IC SKT
13	3	-	14 pin IC SKT
14	1	-	16 pin IC SKT
15	2	-	20 pin IC SKT
16	1	C1	4u7 x 25 Volt Tant Bead
17	1	C2	4u7 x 25 Volt Tant Bead
18	1	C3	4u7 x 25 Volt Tant Bead
19	1	C4	4u7 x 25 Volt Tant Bead
20	1	C5	4u7 x 25 Volt Tant Bead

11.0

EPROM PROGRAMMER BOARD PARTS LIST

ITEM	QUAN	REF.	DESCRIPTION
21	1	C6	4u7 x 25 Volt Tant Bead
22	1	C7	10 N (.01 uf) 20%
23	1	C8	10 N (.01 uf) 20%
24	1	C9	100 N (.01 uf) 5%
25	1	C10	100 N (.01 uf) 5%
26	1	C11	100 N (.01 uf) 10%
27	1	C12	4u7 x 25 Volt Tant Bead
28	1	C13	10 N (.01 uf) 20%
29	1	C14	4N7 (4700 pf) 20%
30	1	C15	DELETED
31	1	C16	4u7 x 25 Volt Tant Bead
32	1	C17	4u7 x 25 Volt Tant Bead
33	1	C18	4u7 x 25 Volt Tant Bead
34	1	C19	4u7 x 25 Volt Tant Bead
35	1	C20	10 N (.01 uf) 20%
36	1	C21	1 N (1000 pf) 20%
37	1	C22	10 N (.01 uf) 20%
38	1	C23	10 N (.01 uf) 20%
39	1	C24	N15 (150 pf) 2%
40	1	C25	100 uf x 35 Volt
41	1	C26	10 N (.01 uf) 20%
42	1	D1	1N4148 diode
43	1	D2	1N4001 diode
44	1	D3	1N4001 diode
45	1	D4	1N4001 diode
46	1	IC1	7805 5 V, 1 A Regulator
47	1	IC2	7805 5 V, 1 A Regulator
48	1	IC3	74LS273 Octal Latch
49	1	IC4	74LS273 Octal Latch
50	1	IC5	4N33 Opto Isolator
51	1	IC6	4N33 Opto Isolator
52	1	IC7	555 Timer
53	1	IC8	4528 CMOS Dual Mono
54	1	IC9	7406 Hex Inv Buf (O/C)
55	1	IC10	7406 Hex Inv Buf (O/C)
56	1	IC11	TL497 Switching Reg.
57	1	IC12	79L05 -5 V, 100 MA Reg
58	1	IC13	78L12 +12 V, 100 MA Reg
59	1	J1	50 Way IDC 3M 3433-5302
60	1	L1	220 uH x 200 MA Ind
61	1	LED1	.2" Red LED
62	1	LED2	.2" Red LED
63	1	R1	680 Ohms, 1/4 W, 5%
64	1	R2	1KO, 1/4 W, 5%
64	1	R3	10K, 1/4 W, 5%
65	1	R4	1MO, 1/4 W, 5%
66	1	R5	1MO, 1/4 W, 5%
67	1	R6	4K7, 1/4 W, 5%
68	1	R7	10M, 1/4 W, 5%
69	1	R8	4K7, 1/4 W, 5%
70	1	R9	4K7, 1/4 W, 5%
71	1	R10	4K7, 1/4 W, 5%
72	1	R11	330 Ohms, 1/4 W, 5%

11.0 EPROM PROGRAMMER BOARD PARTS LIST

ITEM	QUAN	REF.	DESCRIPTION
73	1	R12	0.47 Ohms, 2 1/4 W, 10%
74	1	R13	68 Ohms, 1/4 W, 5%
75	1	R14	68 Ohms, 1/4 W, 5%
76	1	R15	4K7, 1/4 W, 5%
77	1	R16	22K, 1/4 W, 5%
78	1	R17	1M1, 1/4 W, 5%
79	1	R18	1KO, 1/4 W, 5%
80	1	R19	4K7, 1/4 W, 5%
81	1	R20	4K7, 1/4 W, 5%
82	1	R21	47 Ohms, 1/4 W, 5%
83	1	R22	2K7 1/4 W, 5%
84	1	R23	24K, 1/4 W, 5%
85	1	R24	10K, 1/4 W, 5%
86	1	R25	10K, 1/4 W, 5%
87	1	R26	4K7, 1/4 W, 5%
88	1	R27	100 Ohms, 1/4 W, 5%
89	1	RV1	1K Trimmer (3/4")
90	1	RV2	1K Trimmer (3/4")
91	1	RV3	10K Min cermet trimmer
92	1	RV4	500K Min cermet trimmer
93	1	SKTA	24 Pin Z.I.F. IC Skt
94	1	SKTA	28 Pin Z.I.F. IC Skt
95	1	SWA	10 Way SPDT, ERG SCSO-023
96	1	SWB	10 Way SPDT, ERG SCSO-023
97	1	TR1	BC327 (2N4403) GP PNP
98	1	TR2	BC327 (2N4403) GP PNP
99	2	-	TO-220 Heatsink for IC1

BARE BOARD KIT PARTS LIST

ITEM	QUAN	REF.	DESCRIPTION
1	1	PCB	INTERFACE PCB
2	1	PCB	PROGRAMMER PCB
3	1	-	10 Gauge Alum plate
4	4	-	3/8" x 4 BA ALL THD
5	4	-	1/4" x 4 BA C/S
6	4	-	1/4" x 4 BA R/H
7	4	-	12.7 mm Stick on feet

HARD TO GET PARTS KIT

ITEM	QUAN	REF.	DESCRIPTION
1	1	L1	220 uH x 200 MA Ind
2	1	R12	0.47 Ohms, 2 1/4 W, 10%
3	1	RV3	10K Min cermet trimmer
4	1	RV4	500K Min cermet trimmer
5	1	SWA	10 Way SPDT, ERG SCSO-023
6	1	SWB	10 Way SPDT, ERG SCSO-023

12.0 TEST AND ALIGNMENT PROCEDURE (for bare board customers)

1. Complete the software patching as outlined in section 8.0

2. Assemble the two printed circuit boards.

3. Plug the PARALLEL INTERFACE board into the designated S-30 slot.

4. Connect the PROGRAMMER BOARD to the PARALLEL INTERFACE board via the 50 way cable. (OBSERVE THE INDEX MARKS ON THE CONNECTORS!)

5. Turn RV1 and RV2 on the PROGRAMMER board fully anti-clockwise.
   Turn RV3 and RV4 to mid rotation.

6. Apply power to your system.

7. Measure the voltage across C4 on the PROGRAMMER board. It should be +5 Volts (+/- 0.25 Volts). If it is not rectify the problem before proceeding.

8. Follow the startup procedure outlined in section 9.1

9. Providing that all indications are as described in section 9.1 proceed as follows, if the indications are not as described locate and rectify the problem before proceeding.

10. Select a 2716 EPROM ('3 <CR>'). Ignore the switch setting instructions. Move all switches on SWA and SWB to the right. Hit any key. The OPERATIONS MENU should now be present on your terminal.

11. Use the 'FILL BUFFER' operation to fill the buffer area between $0000 and $3FFF with $00.

12. The following operations must be performed relatively quickly and possibly repeated several times before all tests can be completed. Select the 'PROGRAM EPROM' operation ('4 <CR>'). When the 'NOT ERASED' error message appears type in 'Y'. Both LED's on the programmer board should be illuminated at this point. If they are not there is a probably a fault on the PROGRAMMER board. Ensure both LED's are illuminated before proceeding.

13. When the 'VERIFY FAILED' message appears. Simply hit any key and repeat the above procedure until you have completed steps 14 & 15.

14. Measure the D.C. voltage across C2. It should be +5 Volts, +/- 0.25 Volts. If it is not within the range indicated rectify the problem before proceeding.

15. Using an oscilloscope observe the waveform at pin 6 of IC8 (4528 Dual Monostable). A 1 Millisecond (+/- 50 Microseconds) positive pulse should be present. The period between pulses should be approximately 300 Microseconds. If this pulse width is not within the limits specified adjust RV3 until the 1 Ms pulse is within specifcations.

16. Return to the OPERATIONS MENU. Select 'RETURN TO EPROM SELECT MENU' ('7 <CR>'). Select a 2516 EPROM ('4 <CR>'). Ignore the switch setting Instructions (leave all switches to the right). Hit any key to enter the OPERATIONS MENU.

12.0 TEST AND ALIGNMENT PROCEDURE (continued)

17. Select the 'PROGRAM EPROM' operation ('4 <CR>'). Specify a programming range of $0000 to $07FF. When the 'NOT ERASED' error message appears type in 'Y'. Again both LED's should be illuminated.

18. When the 'VERIFY FAILED' message appears. Simply hit any key and repeat the above procedure until you have completed step 19.

19. Using an oscilloscope observe the waveform at pin 10 of IC8 (4528 Dual Monostable). A 50 Millisecond (+/- 2 Millisesonds) positive pulse should be present. The period between pulses should be approximately 300 Microseconds. If this pulse width is not within the limits specified adjust RV4 until the 50 Ms pulse is within specifcations.

20. Repeat steps 17 & 18 as often as necessary to complete steps 21 & 23.

21. Measure the D.C. voltage across C24 (the large electrolytic in the lower right hand corner of the PROGRAMMER board) with an accurate DVM. Adjust RV1 (+21 Volt Adjust) until the voltage across C24 is 21.5 Volts +/- 0.1 Volts. (The extra 0.5 volts compensates for the voltage drops across TR2 and the associated steering diodes under normal load conditions.)

22. Move switch number 10 on SWA to the left.

23. Measure the D.C. voltage across C24 again. Adjust RV2 (+25 Volt Adjust) until the voltage across C24 is 25.5 Volts +/- 0.1 Volts.

24. Measure the voltage across C19. It should be +12 Volts, +/- 0.25 Volts.

25. Measure the voltage across C17. It should be -5 Volts, +/- 0.25 Volts.

26. This concludes the test procedure. The programmer is now ready for use.

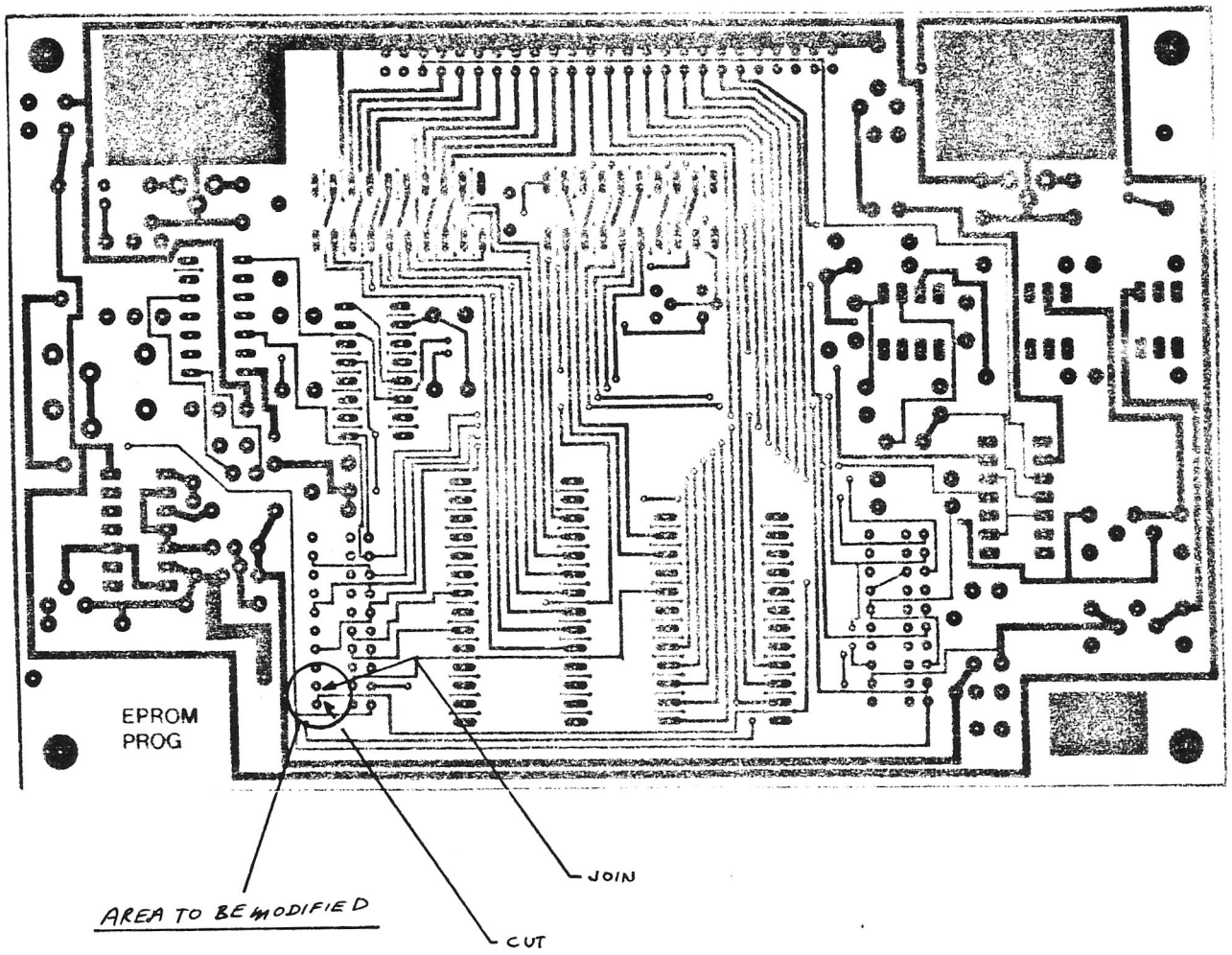
13.0 NOTES FOR PURCHASERS OF BARE BOARD KITS

These comments apply to 'REV B' boards only.

There is a trackwork error on the 'non component' side of the programmer PCB assembly.

To locate the error view the reverse side of the programmer PCB with the 50 way IDC connector uppermost. There is a track going to the left side of SWB (10) that originates from the centre contact of SWA (4). The track that is connected to the left side of SWB (10) should be cut at the pad for SWB (10) and connected to SWB (9) directly above it.

This modification can be performed before or after the programmer board has been assembled.



WINDRUSH MICRO SYSTEMS LIMITED

UNIVERSAL EPROM PROGRAMMER VERSION 2.05 ENHANCEMENTS

This revision of the software removed the two bugs reported on Version 2.01 and Version 2.04. i.e. troubles with 1.0 MHZ systems and the prompt for the 2732A eprom switch settings.

The most noticeable changes are a re-structuring of the software to allow the buffer to be positioned anywhere in memory whilst the prompts for the EPROM buffer area remain referenced to the base address of the EPROM. Thus if you wished to locate the buffer at say $4000 to $9FFF and have the software located at $A000 the buffer area would still be regarded as $0000 to $3FFF.

This eliminates any confusion in calculating offsets. The only time an offset becomes involved is when you are in your disk operating system using the 'GET' or 'SAVE' commands (or the equivelant). This is the only time that the hardware location of the buffer becomes involved in any calculations.

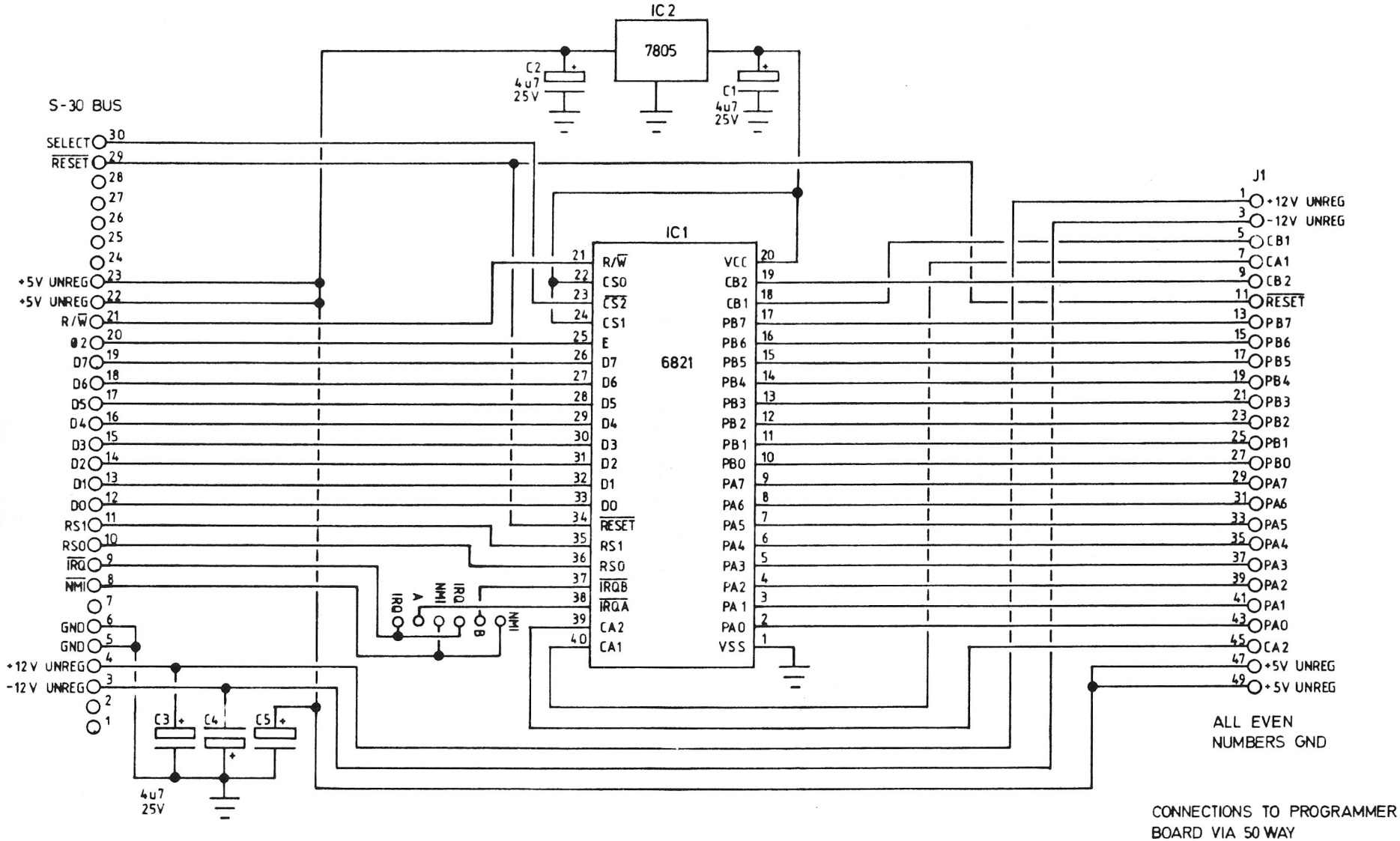
As a part of this enhancement all buffer operations i.e. 'MOVE DATA', 'EXAMINE & CHANGE' and 'FORMATTED DUMP' are now restricted to the allocated buffer area. This prevents corrupting data outside of the buffer area.

This release is accompanied by the initial release of our EPROM programmer software for OS-9 level one and Motorola MDOS Version 3.00 or higher. An interface board is now available for the Motorla EXORciser II also.

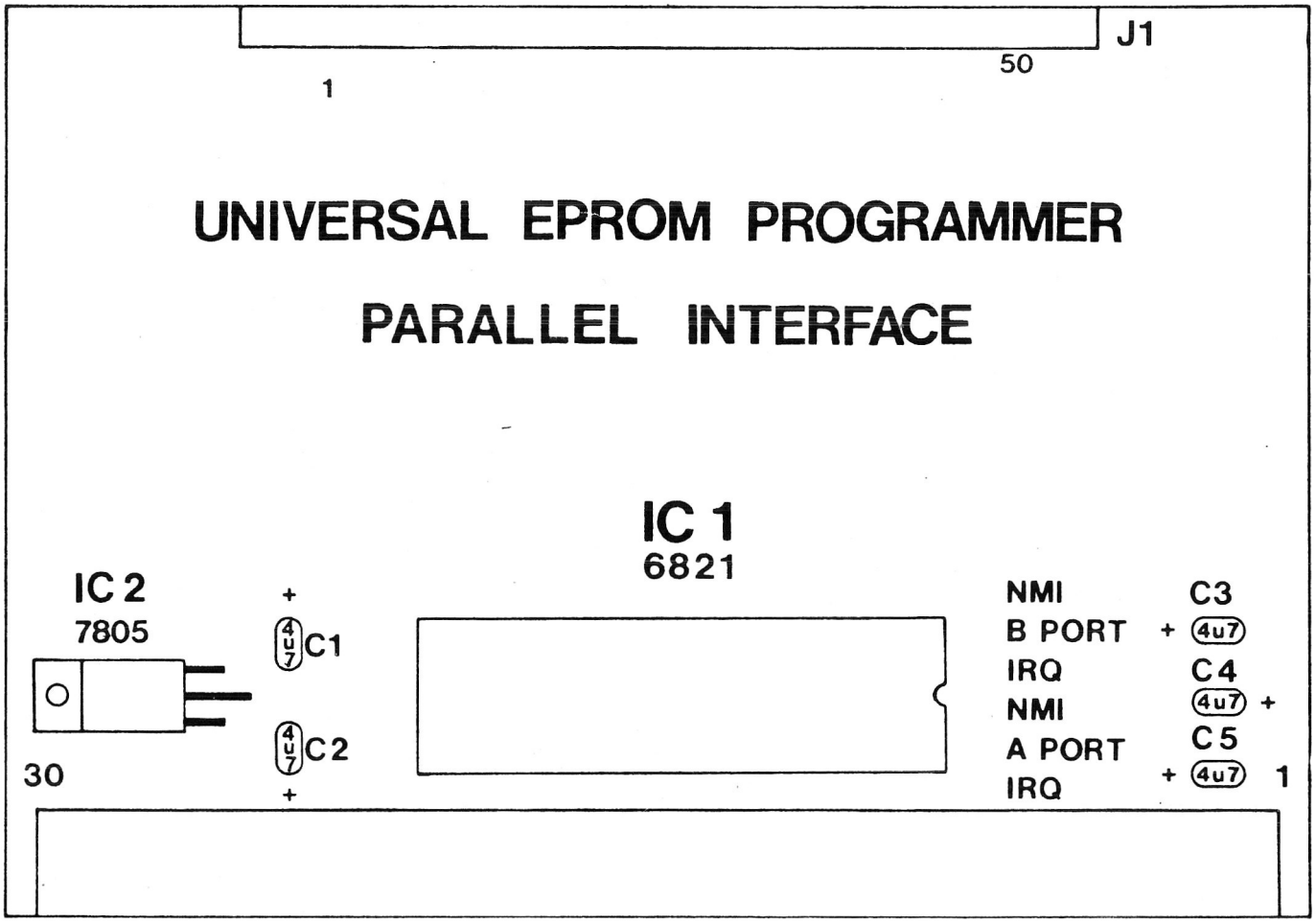
We hasten to point out that the version of our programmer software for OS-9 level one positively will not operate on OS-9 level two. This is due to the memory management/protection scheme used by Microware. This scheme prevents a user program from accessing memory outside of his program's allocated limits. Since our software attempts to make a direct access to the PIA (it is not using a device descriptor) this portion of the software is ignored by OS-9 level two.

We are currently working on a version of this software that will operate on OS-9 level two. It should be available in the third quarter of '82.

We will upgrade any software package supplied earlier or will supply additional software for different operating systems for 25.00 U.S. Dollars/15.00 pounds sterling per disk. This amount includes REGISTERED AIR MAIL costs to anywhere in the world.



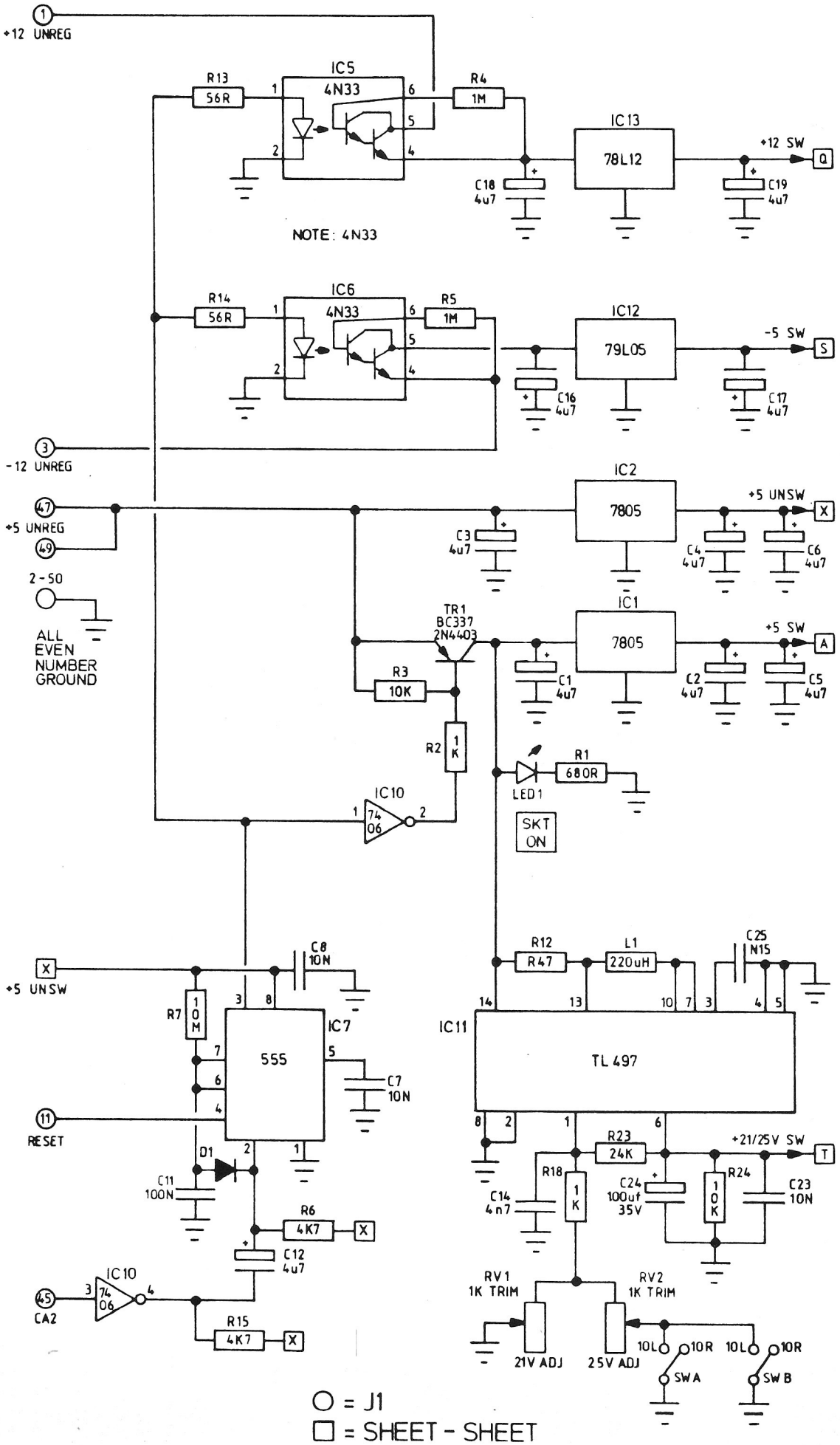
REV	DATE	REV	DATE	DWN			
A	13 OCT 81			HE	UNIVERSAL EPROM PROGRAMMER, PARALLEL INTERFACE	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.	EPROM PROGRAMMER SHEET 1 OF 8



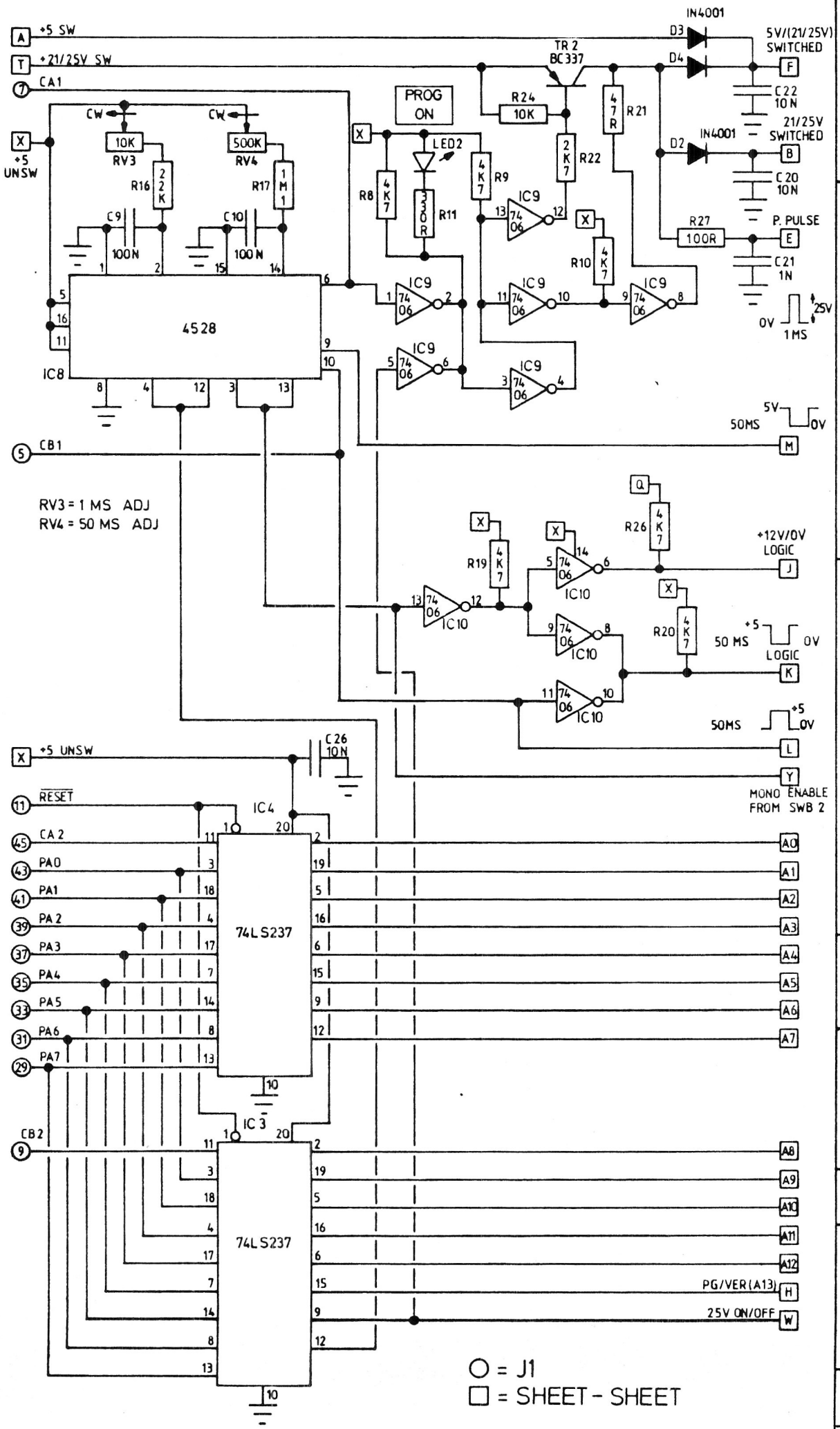
REV	DATE	REV	DATE	DWN		
A	7 OCT 81			HE	COMPONENT OVERLAY FOR UNIVERSAL EPROM PROGRAMMER PARALLEL INTERFACE	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.

EPROM PROGRAMMER

SHEET 2 OF 8



REV	DATE	REV	DATE	DWN	POWER SUPPLY DETAILS FOR UNIVERSAL EPROM PROGRAMMER	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.	EPROM PROGRAMMER SHEET 3 OF 8
A	21 OCT 81			HE			



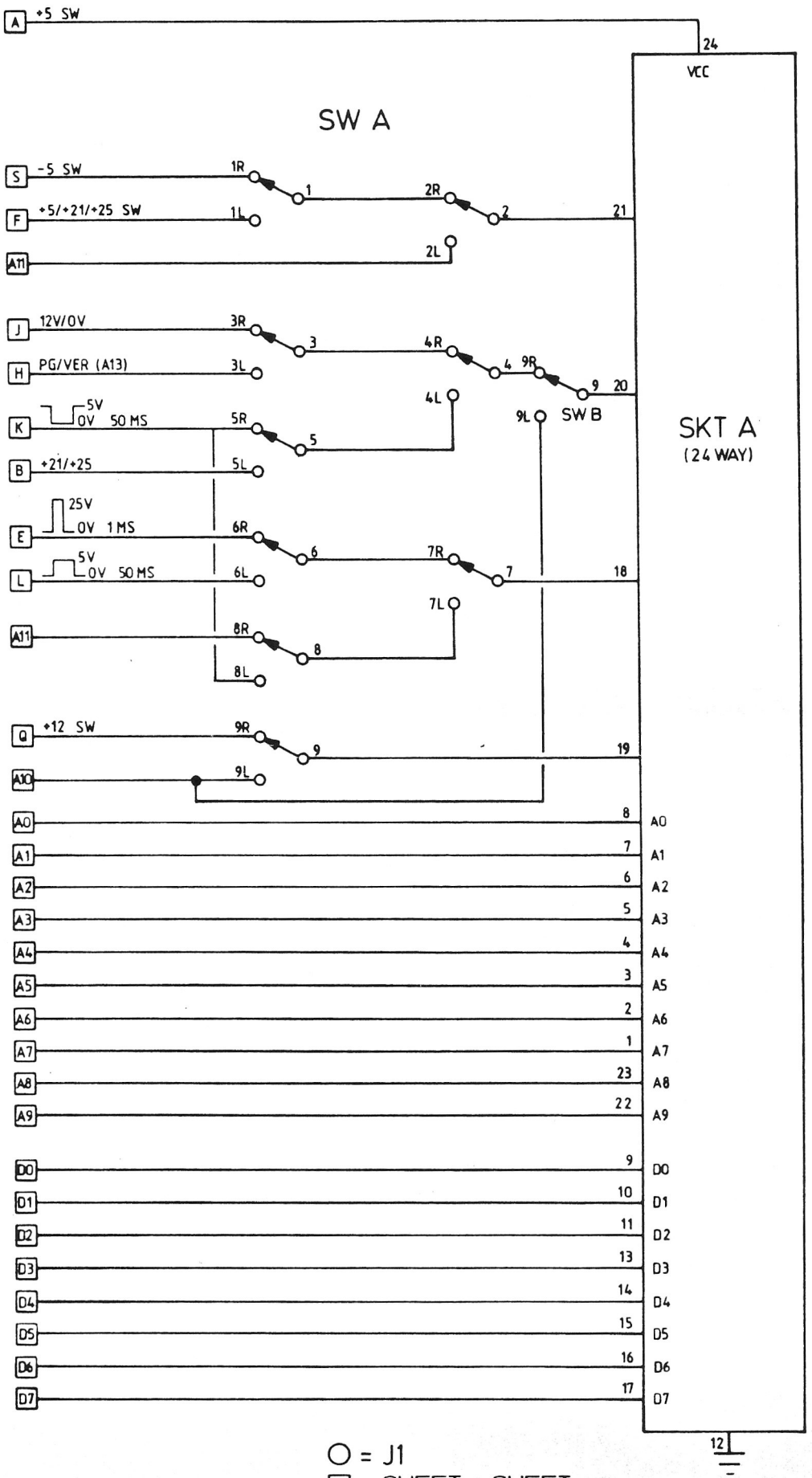
EPROM PROGRAMMER SHEET 4 OF 8

WINDRUSH MICRO DESIGNS
GAYMERS WAY, NORTH
WALSHAM, NORFOLK.

INPUT LATCH MONOSTABLE & MODE SWITCH DETAILS
UNIVERSAL EPROM PROGRAMMER

DWN HE

REV	DATE
A	21 OCT 81
B	14 MAR 82



REV	DATE
A	21 OCT 81

REV	DATE

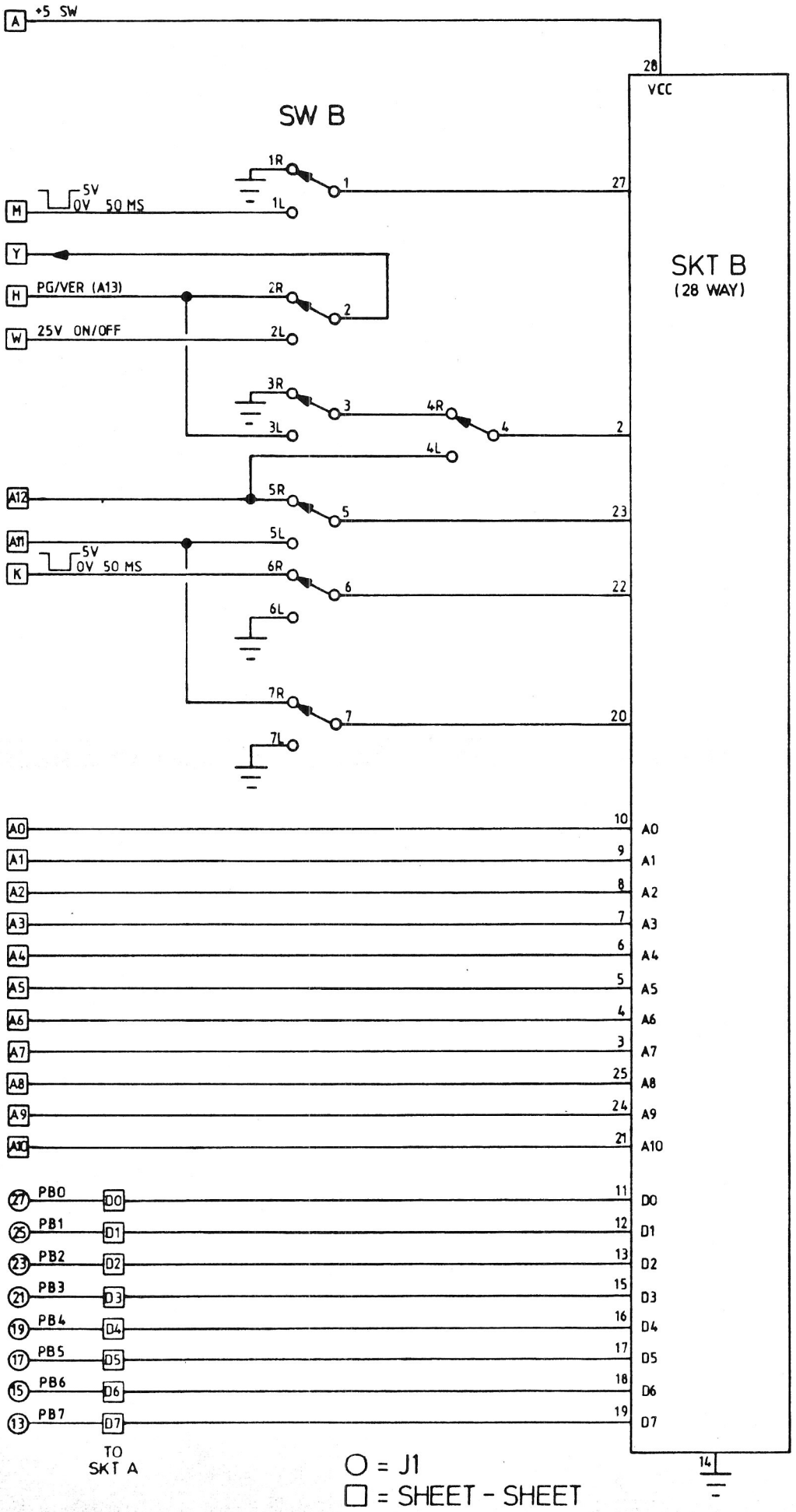
DWN
HE

SOCKET A & SWITCH A
UNIVERSAL EPROM
PROGRAMMER

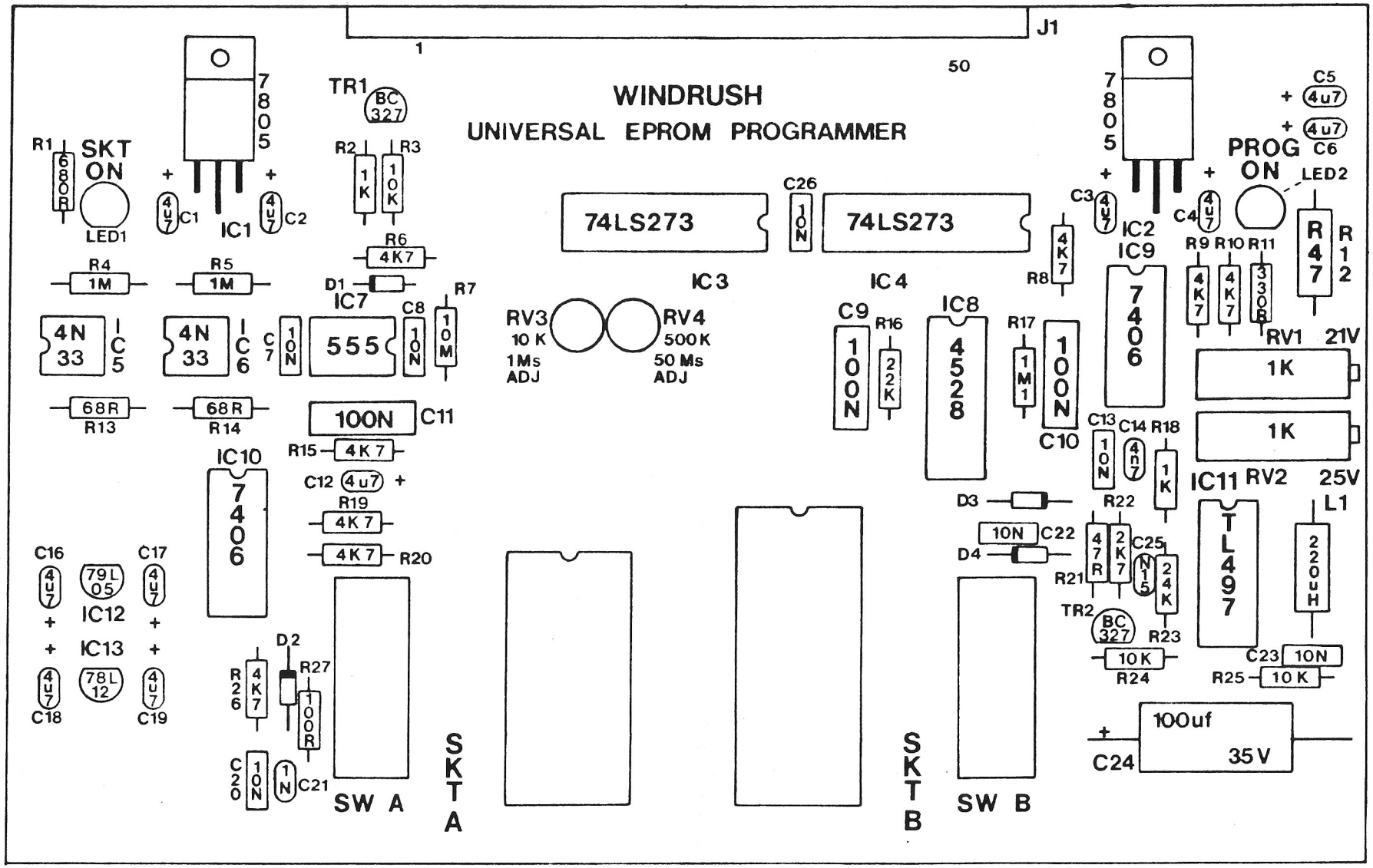
WINDRUSH MICRO DESIGNS
GAYMERS WAY, NORTH
WALSHAM, NORFOLK.

EPROM
PROGRAMMER

SHEET 5 OF 8



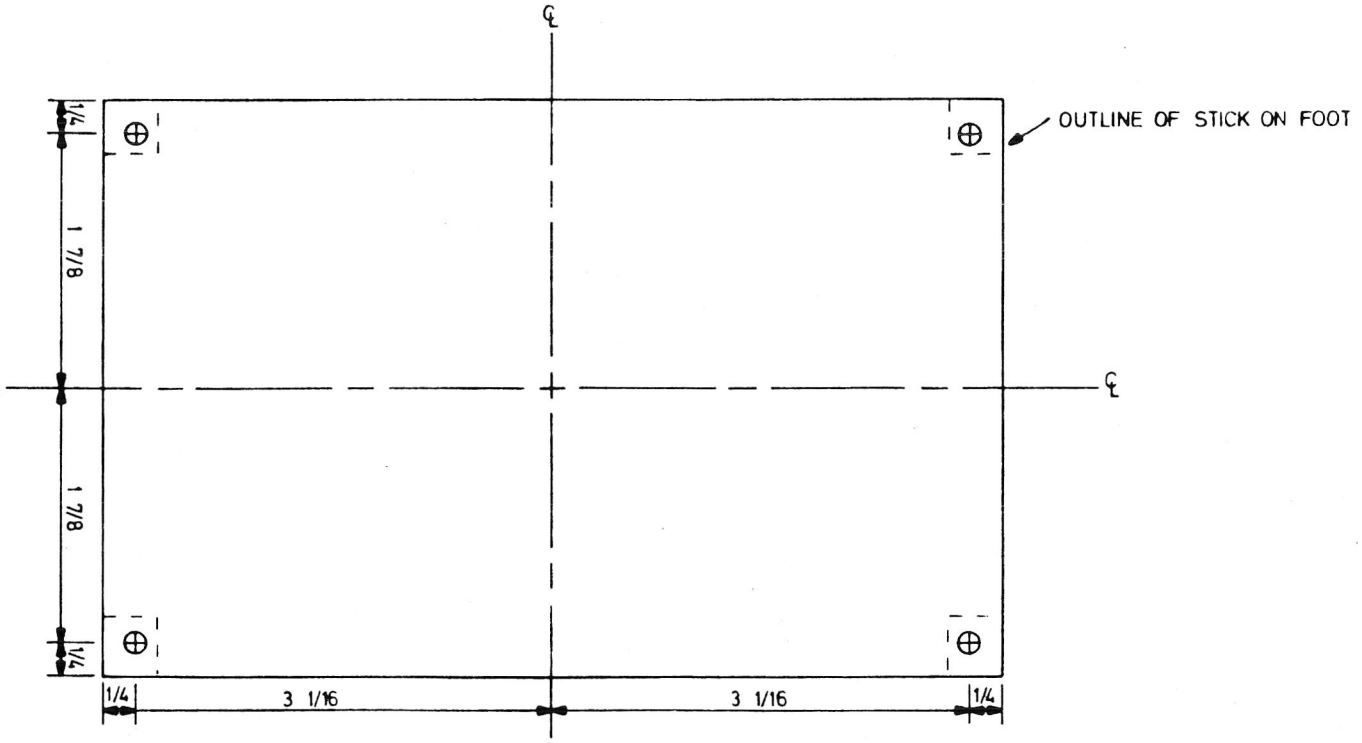
REV	DATE	REV	DATE	DWN	SOCKET B & SWITCH B UNIVERSAL EPROM PROGRAMMER	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.	EPROM PROGRAMMER SHEET 6 OF 8
A	21 OCT 81			HE			



REV	DATE	REV	DATE	DWN		
A	8 OCT 81			HE	COMPONENT OVERLAY FOR THE WINDRUSH UNIVERSAL EPROM PROGRAMMER	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.
B	22 JAN 82					

EPROM PROGRAMMER

SHEET 7 OF 8



HOLES TO BE DRILLED
5/32 & COUNTERSUNK
TO 4BA
MATERIAL - 1/8" THICK ANODIZING QUALITY
ALUMINIUM

FINISH - BLACK ANODIZE

4 STICK ON FEET FITTED UNDER
C/S 4BA SCREWS
MOUNTED TO PROGRAMMER BOARD
BY FOUR OFF 3/8" ALL THD 4BA
SPACERS.

REV	DATE	REV	DATE	DWN	BASE PLATE DETAILS FOR EPROM PROGRAMMER	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.	EPROM PROGRAMMER SHEET 8 OF 8
A	20 OCT 81			HE			

WINDRUSH MICRO SYSTEMS LIMITED
==========

UNIVERSAL EPROM PROGRAMMER VERSION 2.04 HARDWARE BUGS REPORTED AS OF 1 JUNE 1982
==========

1. This problem concerns Texas Instruments TMS2716 TRI-Volt EPROMS. These devices require that the VCC pin be raised to +12 volts during program. The design of the programmer did not cater for this unusual requirement and thus will not program this particular manufactures device without modifying the board. The modifications are detailed for PCB assembly REV B. The REV A boards are virtually identical except that the connections to SWB, 6 are reversed.

   The attached sheets detail the minor modifications required. The only change that requires a bit more explanation is the modifications to the switched +5 Volt regulator. The modification requires that this regulator be insulated from it's associated heat sink and that the common pin (the one in the centre) be completely isolated from earth. The common pin is cut back to within 1/8" of the regulator and a 1N4148 diode placed in series with the connection between the regulator common and ground. This modification compensates for the additional 0.7 volts lost in diodes D3 and D5.

   This modification takes the +25.7 volt 1Ms pulses present during programming and produces a +12 volt steady DC voltage across the 220uf capacitor. This voltage is switched to the VCC pin of the EPROM via SWB, 8 when the 2716 device is selected. D5 prevents the +12 volts from effecting the +5 volt supply.

   If you do not intend to program the TMS2716 devices this modification notice can be ignored. If you decide to modify the programmer the following ammendment to the software is also required:

   Look for 'INMSG3' in part 3 of the programmer software. Change the line refering to setting SWB to the following:

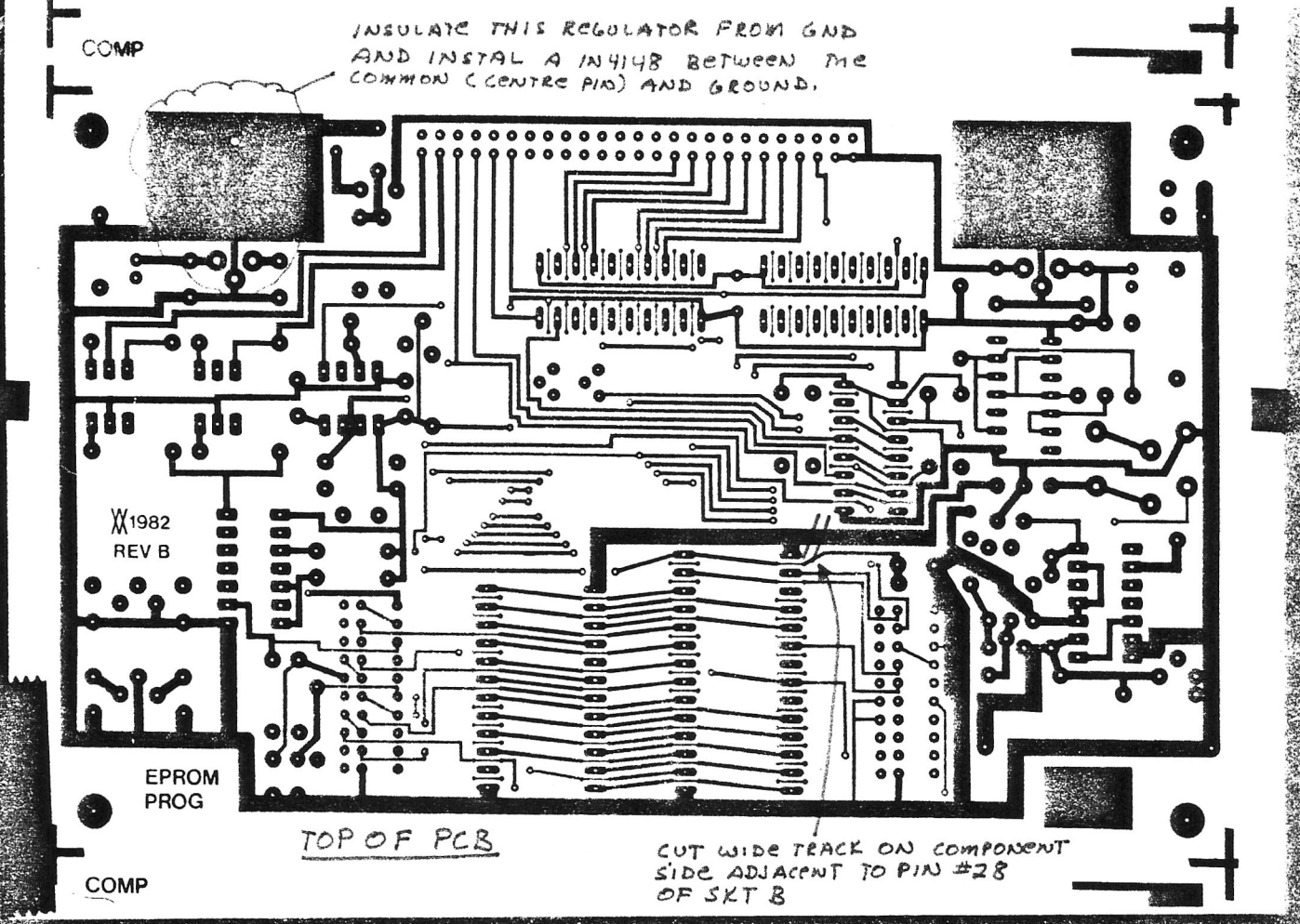
```
FCC 'SET SWB = 8,9 LEFT. ALL OTHERS RIGHT.'
```

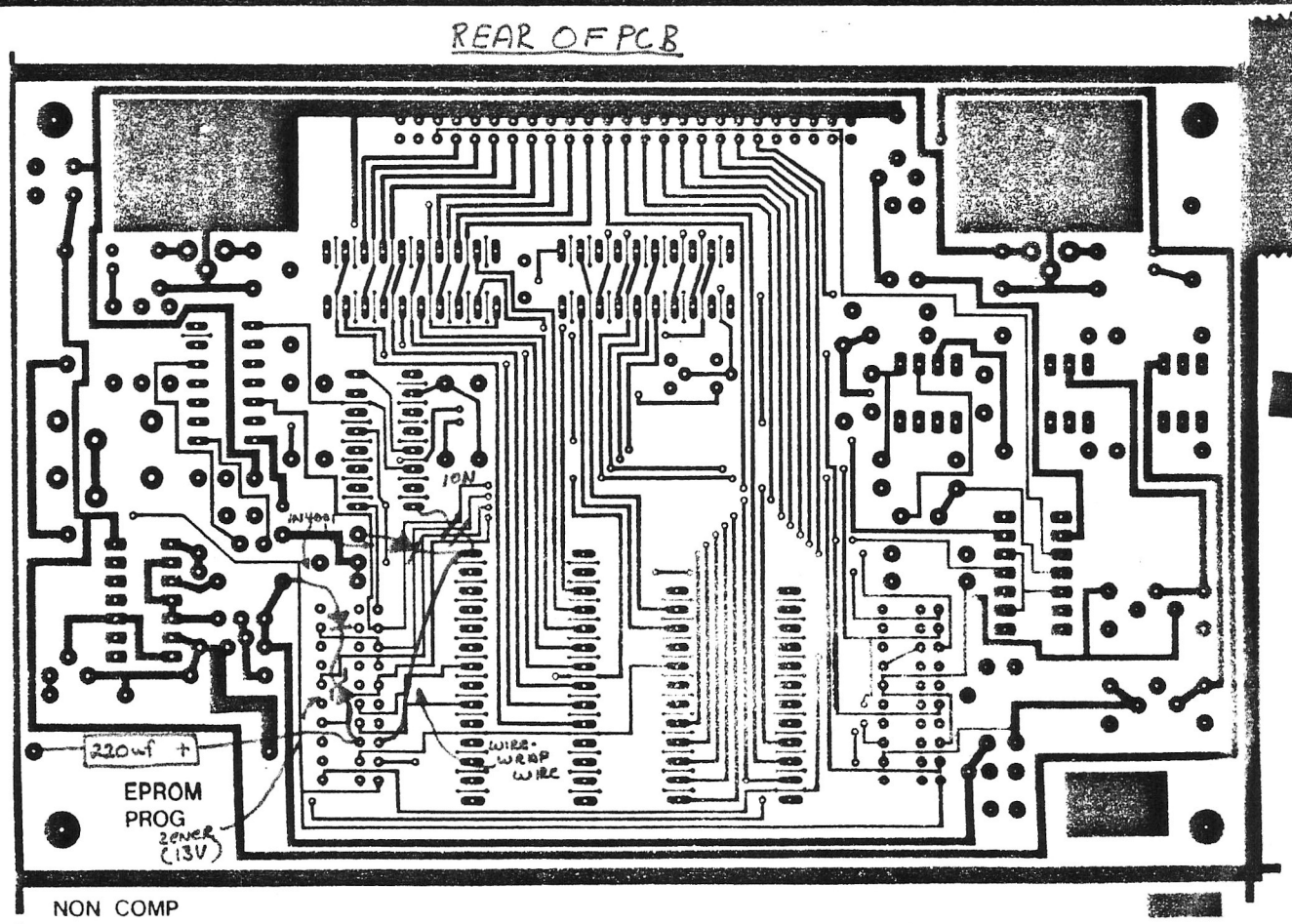
   Look for 'INMSG8' in part 3 of the programmer software. Change the line referring to setting SWB to the following:

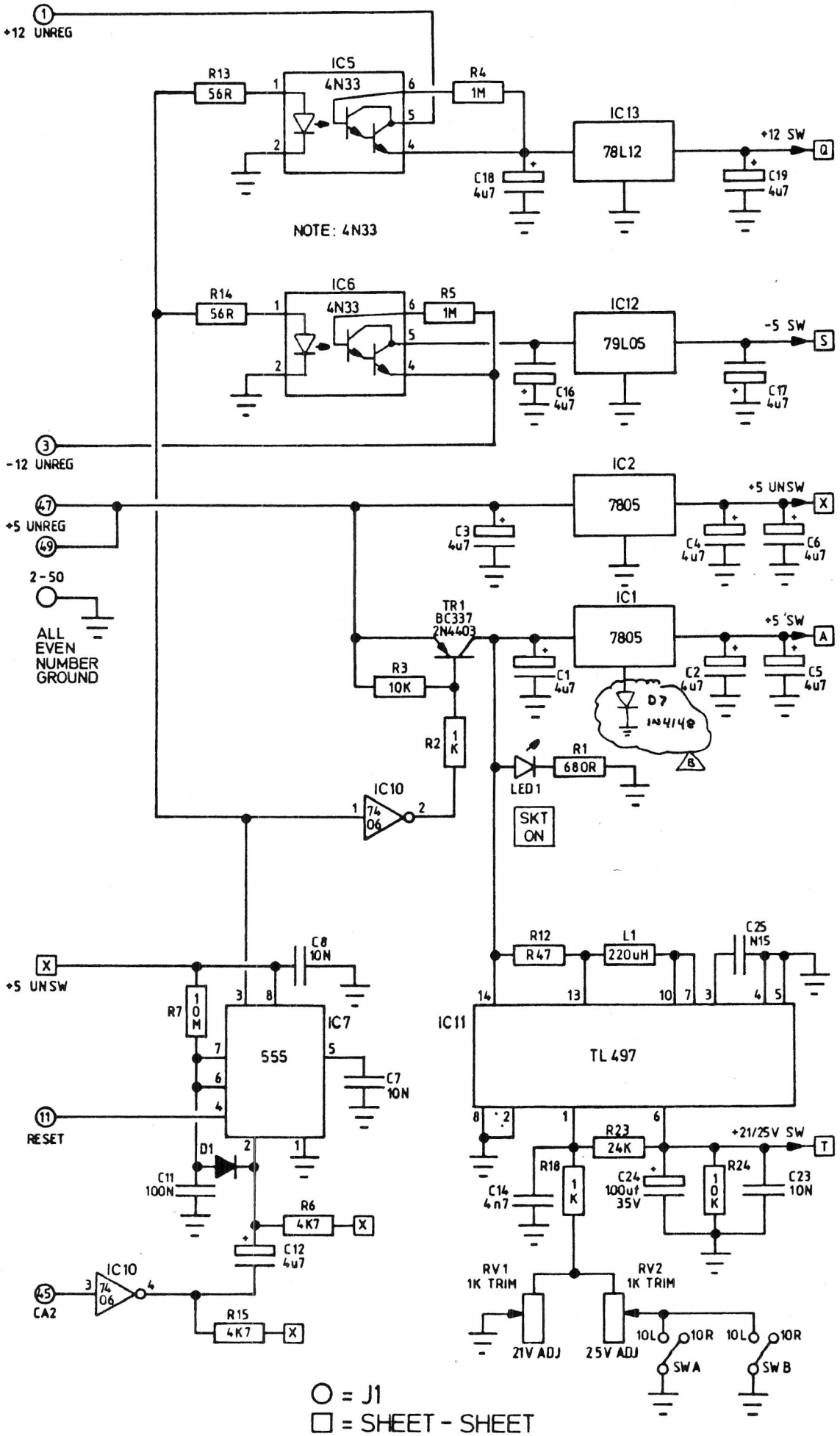
```
FCC  'SET SWB = 8,9,10 RIGHT. ALL OTHERS LEFT.'
```

   The following parts are required to implemement this modification. Customers who have purchased the factory assembled programmer may return the unit to our works for modification free of charge or may write to us for a free modification kit.

   A. One mica insulator & bush for TO-220 devices.
   B. One screw, nut, shakeproof washer, and two flat washers to suit above bush
   C. One 1N4148 diode
   D. Two 1N4001 (or equiv) 1 amp x 50 PIV diode
   E. One 13 Volt, 1.3 watt zener
   F. One 10N (.01uf) x 63 Volt ceramic disc capacitor, bypass quality
   G. One 220uf x 16 Volt electrolytic
   H. Short length of wire wrap wire.


COMP
INSULATE THIS REGULATOR FROM GND AND INSTAL A IN4148 BETWEEN THE COMMON (CENTRE PIN) AND GROUND.
1982
REV B
EPROM
PROG
TOP OF PCB
CUT WIDE TRACK ON COMPONENT SIDE ADJACENT TO PIN #28 OF SKT B
COMP


REAR OF PCB
ION
IN4004
220uf +
EPROM
PROG
ZENER (13V)
WIRE-WRAP WIRE
NON COMP



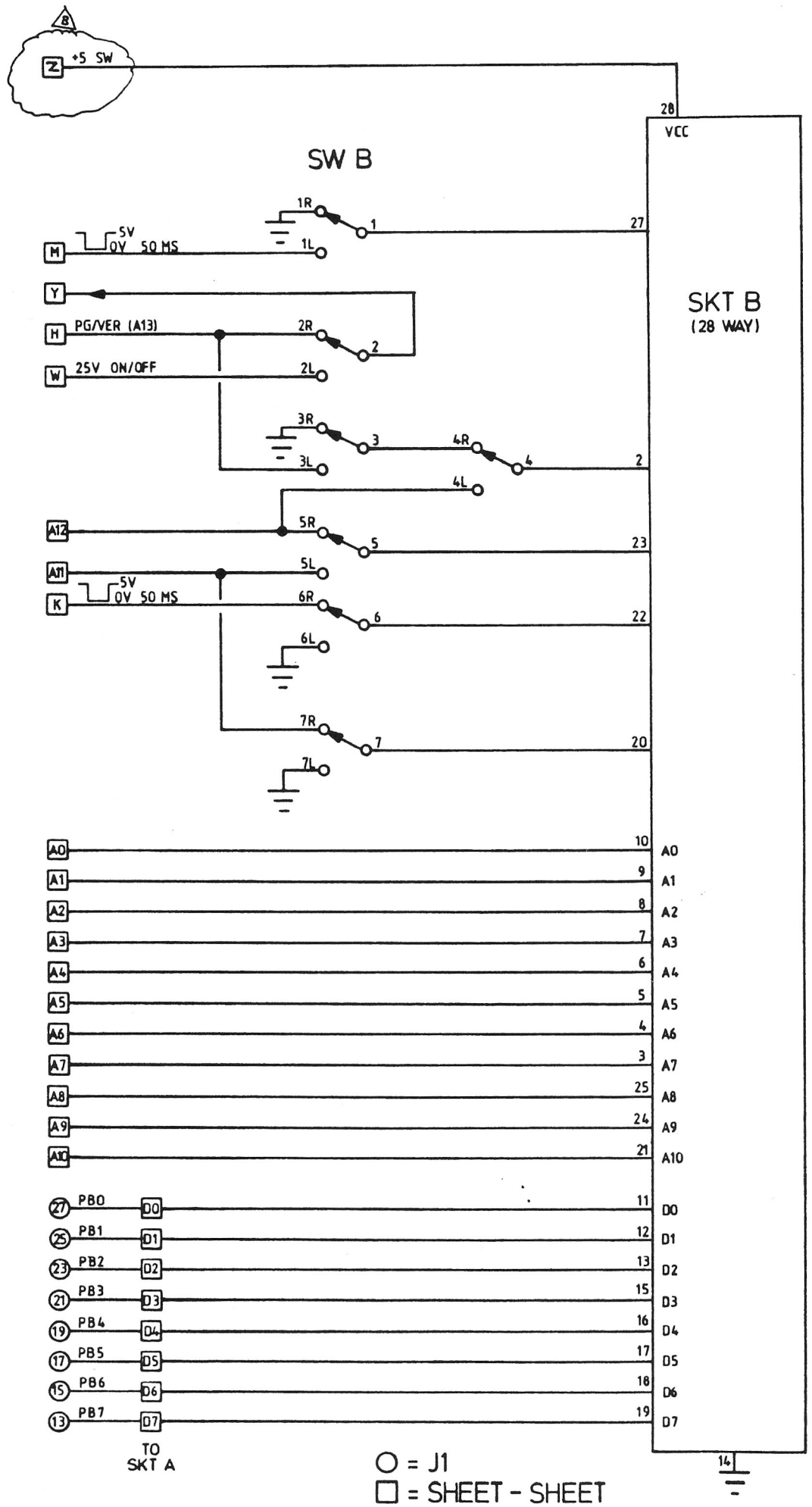
REV	DATE
A	21 OCT 81
B	1 JUN 82

REV	DATE	DWN
		HE

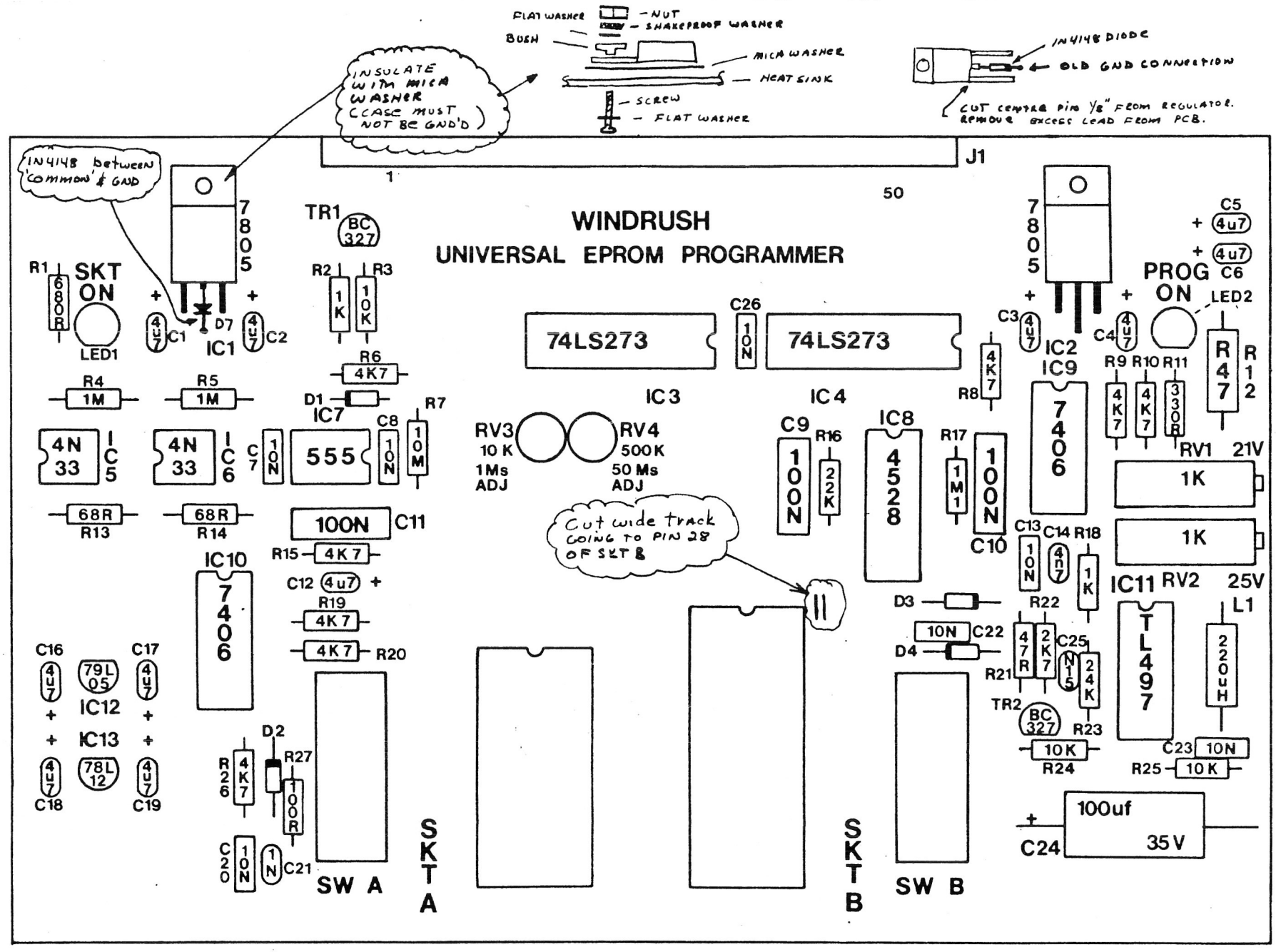
WINDRUSH MICRO DESIGNS
GAYMERS WAY, NORTH
WALSHAM, NORFOLK.

POWER SUPPLY DETAILS FOR
UNIVERSAL EPROM
PROGRAMMER

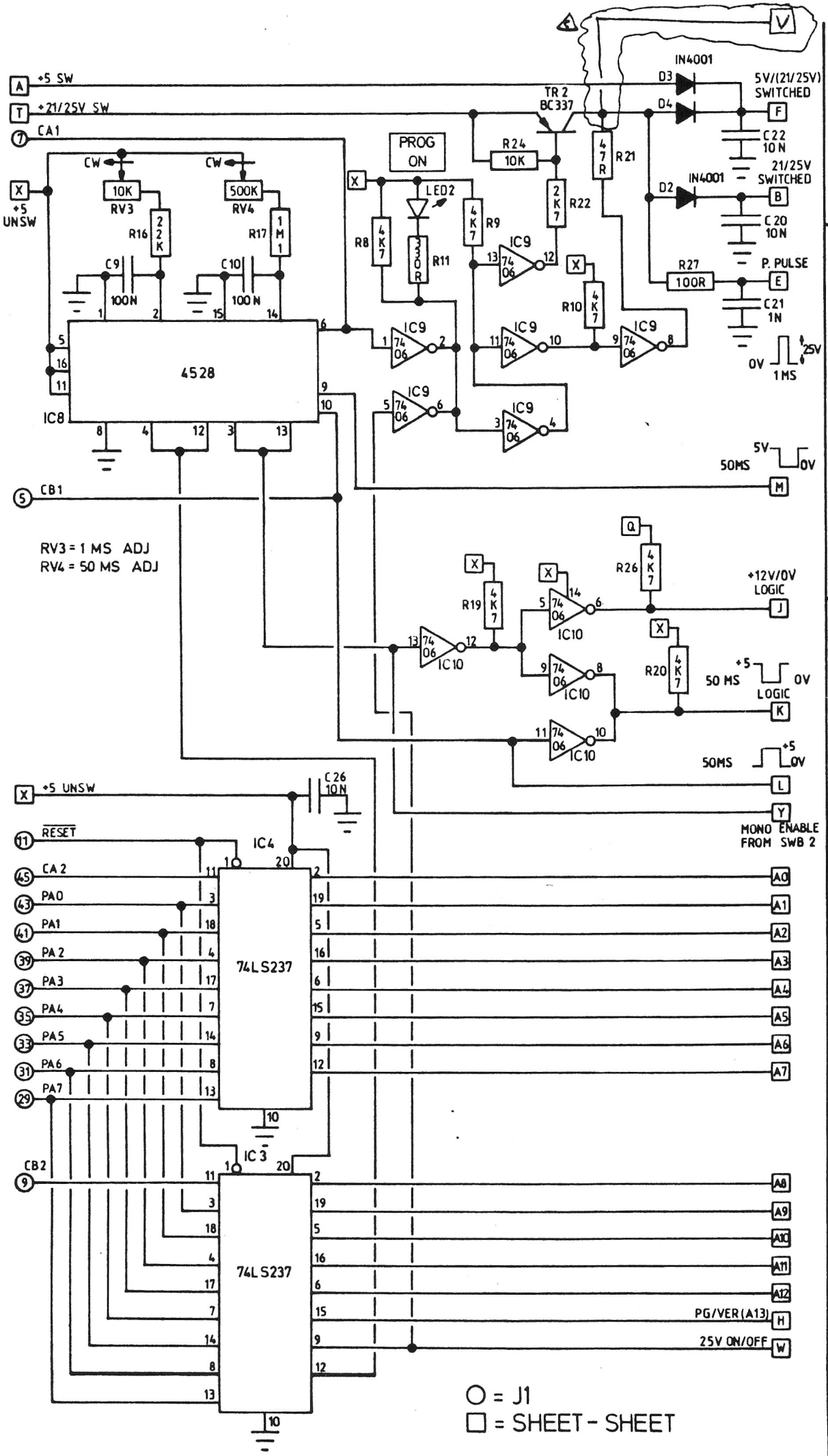
EPROM
PROGRAMMER
SHEET 3 OF 8



REV	DATE	REV	DATE	DWN	SOCKET B & SWITCH B UNIVERSAL EPROM PROGRAMMER	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.	EPROM PROGRAMMER SHEET 6 OF 8
A	21 OCT 81			HE			
B	1 JUN 82						



REV	DATE	REV	DATE	DWN		
	18 OCT 81			HE	COMPONENT OVERLAY FOR THE WINDRUSH UNIVERSAL EPROM PROGRAMMER	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH



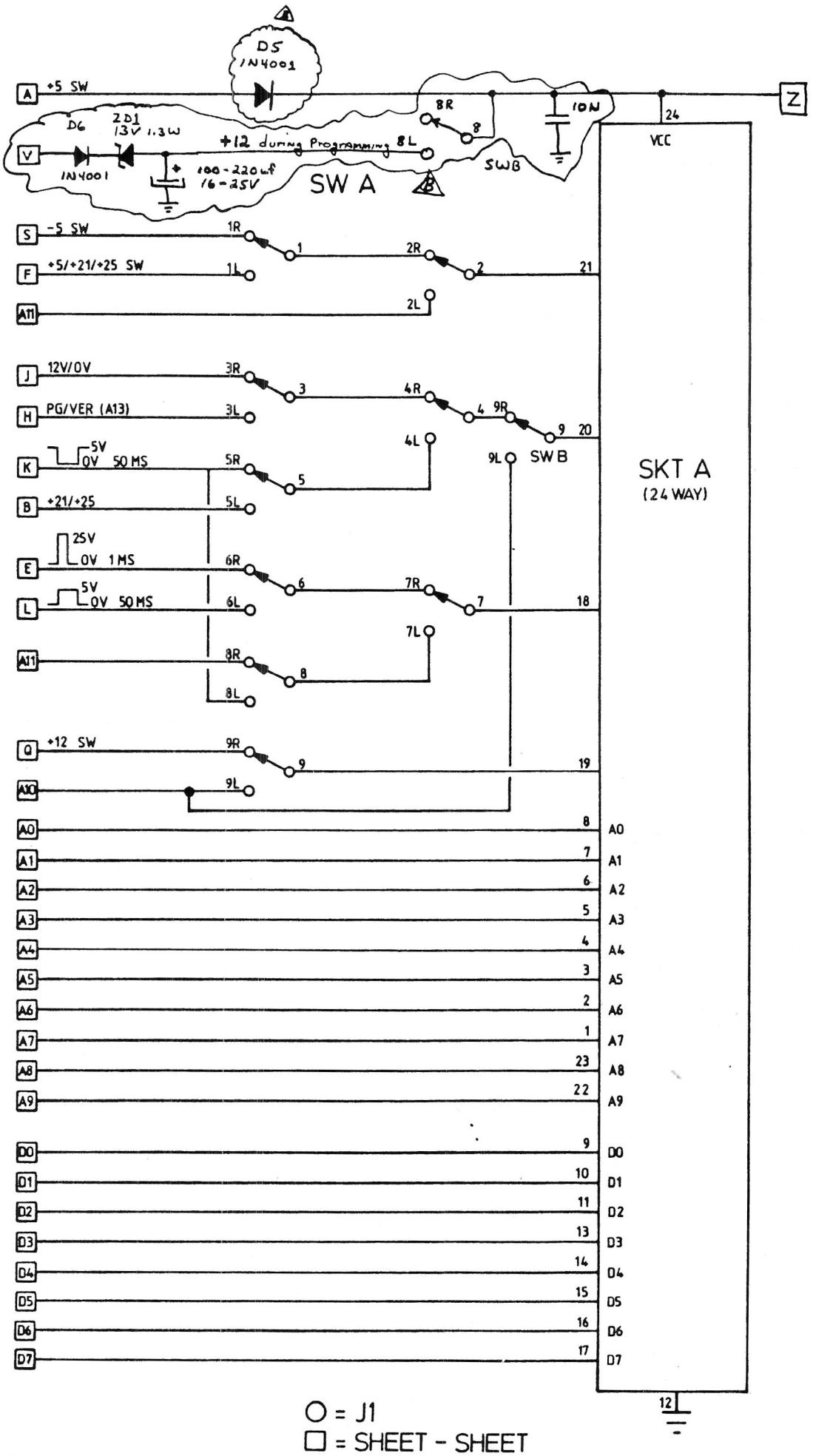
EPROM PROGRAMMER SHEET 4 OF 8

WINDRUSH MICRO DESIGNS
GAYMERS WAY, NORTH
WALSHAM, NORFOLK.

INPUT LATCH MONOSTABLE & MODE SWITCH DETAILS
UNIVERSAL EPROM PROGRAMMER

DWN HE

REV	DATE	REV	DATE
A	21 OCT 81		
B	14 MAR 82		



REV	DATE	REV	DATE	DWN	SOCKET A & SWITCH A UNIVERSAL EPROM PROGRAMMER	WINDRUSH MICRO DESIGNS GAYMERS WAY, NORTH WALSHAM, NORFOLK.	EPROM PROGRAMMER SHEET 5 OF 8
A	21 OCT 81			HE			