Technische dokumentatie

TRADECOM INTERNATIONAL B.V.

HONDSDIJK 1-A - POSTBUS 60 - 2396 ZH KOUDEKERK A/D RIJN - TEL. 01714-4300* - TELEX 39466

NewBrain CP/M system description issue 3.0

1) Introduction

The NewBrain CP/M system operates with two memory configurations - the 32K and 64K systems. The 32K system requires a NewBrain A or AD and a floppy disc controller with at least one floppy disc drive. The 64K system requires the expansion box fitted with 64K extra RAM in addition. Although the two systems work in the same way, you may find that some of the CP/M programs you wish to run will only operate in the full 64K system. For example WORDSTAR will only run on a 64K system. The same discs can be used for either system.

2) Entering the CP/M system

With the 32K CP/M system this can done by using the comand "CPM" from BASIC or by pressing the stop key on powerup. The 64K system can only be entered by selecting "CP/M 2.2" from the system menu. The system information will be read from drive A and the copyright message will be displayed. At this point if an initial command has been set by the SETINIT program it will be executed. Otherwise the CP/M prompt "A>" will be displayed.

3) The keyboard

The keyboard operates in a slightly different way than under normal conditions. The first thing is that control codes will operate as is usual under CPM.. This means that, for example it is not immediately possible to change the mode of the screen using cntrl/W. To do this the codes have to be sent to the screen. A trick to get round this is to use the fact that the command program reflects unrecognisable commands. If you wish say to make the screen black on white type following "A>" cntrl/W, cntrl/Q and newline. The screen will change followed by a "?" to indicate that the control program doesn't understand. The desired effect will however have been achieved.

Since the cursor control, home, insert and escape keys are really control keys, these will also not have the expected effect, unless the program being run will interprete them. One effect is that cursor left often means delete the character on the left, and shift cresor left will often mean delete everything on the left upto the prompt (if any). Many programs (e.g. WORDSTAR) will use different control keys for cursor movements.

The NewBrain keyboard has no 'delete' key. When the delete key is refered to in CP/M documenation or in CP/M program documentation the control key should be used with the comma key instead.

The control 0 to 9 functions now have new meanings. Cntrl/0 to cntrl/7 have no effect (they are to have special meanings under MP/M). Cntrl/8 will set and unset the 'telytype caps'. When this mode is set all letters keyed in will be made into capitals whether or not the shift key is pressed. This is similar to the old cntrl/1 function. Whenever this mode is set, the mode will be displayed on the single line display in model AD NewBrains. To obtain the meanings the cntrl/0 to cntrl/9 keys had previously, cntrl/9 should be pressed beforehand.

The stop key has no effect.

The the keyboard operates on a type-ahead buffer, which means that anything typed in will be stored until the next input is required.

4) Screen codes

The screen used by CP/M is similar to the normal screen editor except it is much simpler. It operates on a 25 by 80 screen using the normal character set. Characters whose top bit are set will display according to the current TV mode. When the last column has been typed to the cursor stays on the last character but changes from a line to a block to indicate that the next character typed will appear on the next line. Those control codes not listed below will perform no action at the moment, but they may be made to do something at a later time and should be avoided.

00H - Null. No action.

01H - Insert line. The lines below the current line (if any) are scrolled down. The current line is cleared and the cursor is moved to the left hand side.

02H - Delete line. The current line is deleted and the lines below (if any) are scrolled up. A blank line is inserted at the bottom of the screen.

03H - Delete to the end of line. The character at the cursor and all characters to the right of it (if any) are converted to spaces.

05H - Cursor off. The cursor will cease to flash.

06H - Cursor on. The cursor will start to flash.

08H - Backspace. The cursor is moved one space to the left, or if it was on the left hand side to the right hand side but one of the above line (the 80th character). If the cursor was at the top-left corner, it is moved to the top-right corner.

09H - Tab. Spaces are printed until the column number is a multiple of 8. A new line is taken if necessary.

0AH - Cursor down. The cursor is moved one space down, and if the cursor was at the bottom of the screen, the screen is scrolled up and a blank line inserted at the bottom.

0BH - Cursor up. The cursor is moved up one line, unless it was already at the top line.

0CH - Cursor right. The cursor is moved one position to the right, or if it was beyond the rightmost column (with block cursor) it is moved to the second column on the next line.

0DH - Carraige return. The cursor is moved to the left of the current line.

11H - Insert mode. Further non-contol characters are inserted at the cursor position, with the character previously at the cursor position and characters to the right of it moved to the right. Characters overflowing the right hand side of the screen are lost. This mode is cancelled when another control character is sent.

14H - Escape. The following character is not interpreted as a control code, except for 0, which is always ignored.

15H - Clear line. The current line is written with spaces and the cursor moved to the left hand side.

16H - Set cursor position. Following this control code the next two characters sent are interpreted as the column and row number to which the cursor should be set. The column number should be sent first. The top row is row 1 and the leftmost column is column 1.

17H - Set TV mode. Following this control code the next character sent will set the TV control register whenever that screen is being displayed. Only bits 0 and 1 are interpreted. Bit 0 determines the background colour (0 means background dark, 1 background bright). Bit 1 determines how characters with bit 7 set are interpreted (0 gives reverse characters, 1 allows 256 character set).

18H - Delete left. The character to the left of the cursor is deleted and subsequent characters are moved one position to the left. The cursor is moved one position leftwards. If the cursor was in the leftmost column, no action is taken.

19H - Delete. The character at the cursor is deleted and subsequent characters are moved one position to the left.

1AH - Clear screen. The screen is cleared and the cursor is moved to the top left corner.

1BH - Escape2. If this is followed by anything other than '=' then no action is taken at the moment (these sequences are reserved fopr expansion). If it is followed by '=' then a conventional cursor positioning takes place, with the row address followed by the column address. In this case the top left corner is at 20H,20H.

1CH - Cursor home left. As carraige return.

1DH - Cursor home right. The cursor is moved to the right most column in the line (with block cursor).

1EH - Cursor home. The cursor is moved to the top left corner.

1FH - Clear screen. As 1AH.

5) Other devices

In the 32K system the PTR: device is always set to the printer port, and the RDR: and PUN: devices are set to the communications port. These are set to work at 9600 baud. In the 64K system the ports used will be in the expansion box will be used instead, but these may be changed by using the CONFIGUR program.

CSP 1/6/83

CP/M utility documentation issue 2.0

1) Introduction

This document refers to the programs provided on the NewBrain CP/M system disc that are produced by Grundy Business Systems. These programs are particular to the NewBrain CP/M system and will ot work on any other CP/M system. In addition they will not work with the NewBrain MP/M system (programs which work with both will be issued with the MP/M system). These programs should not be used through a networked system.

2) FORMAT

Blank discs must be formatted with this program before use. When it is invoked it will ask which drive is to be used for the formmatting operation. If you have two or more drives, it is recommended to use a different one to A:. Answer A, B, C, or D as appropriate and press newline. The program will then ask you to place a blank disc in the stated drive and press newline. If you have a single drive system it is important at this point to remove your system disc and place a blank disc in the drive, otherwise your system disc will become formated. To avoid accidents it is a good idea to write-protect your system disc. If anything other than newline is pressed at this point control will return to the first question. Once newline is pressed the disc will be formatted. At the end the first question will be reasked and you can format another disc. If you wish to exit the program press newline. Before you do this make sure that a system disc is in A: again.

3) SYSGEN

This program copies the system information on the reserved tracks form one disc to another. This should usually be done after a formatting a new disc. The information copied will include the configuration information and the initial command line. When activated it will first ask for the letter of the drive from which the system information is to be read. Enter the letter of the drive and press newline. The system information will then be read from that drive. It will then asked where the information is to be written to. Answer this in the same way, and the information will then be written. This question will then be repeated. To exit the program press newline or cntrl/C. At any point discs in the drives may be removed or replaced as required.

4) SETINIT

This program will set up an initial command that will be executed by the CP/M system when cold-booting the system. For example, you may wish always to use WORDSTAR. When the program is activated it will prompt with '>'. Type in the command you wish e.g. "WS" and press newline. If you wish to clear the initial command just press newline. The initial command will then replace the previous one. To leave the command unchanged press cntrl/C. Note that the command will be set on the disc in drive A: only. To place it on other discs you must either repeat the program or SYSGEN to the other discs.

5) EXIT

This program exits the CP/M system. In the case of the 32K CP/M system the screen will go blank temporarily before restarting BASIC. While this is occuring you may press the stop key to reenter CP/M instead (as on powerup). With the 64K system the main menu will reappear and you can then select your next program.

6) SCOPY

This is a program that copys a disc track by track when only a single drive is available. The method is to fill the RAM with as much as possible from the source disc and then write this to the destination disc. When the program is started up it will say "Place source disc in drive A:". This will then be read, and then it will say "Place destination disc in drive B:". This will then be written to. The process will be repeated, until at the end it will say "Type C to continue with another copy operation or newline to return to the system".

7) DCOPY

This is a program to copy a disc when to drives are available. When it starts up it will say "Place source disc in drive A: and destination disc in drive B: and type S (or newline to quit)". Any other key will cause the program to abort. A track by track operation will then take place, and the original prompt will reappear. This program will only work on two similarly configured drives (PIP should be used otherwise).

CSP 13/5/83

CONFIGURE program documentation issue 1.0

1) Introduction

The NewBrain CP/M CONFIGURE program which will run under both 32K and 64K CP/M systems allows the user to set his CP/M system according to the type of disc drives and devices he wishes to use. The information is processed and written to the first sector of the first track of the drive specified at the end. This information is thus copied during SYSGEN and does not affect the files on the disc. The information is only read in at cold boot time, so will not take force until that occurs. When a cold boot occurs, the configuration information is read off the disc in physical drive 0 (usually A:). The information on discs in any other drives is ignored. The configuration information is used in both CP/M systems and in both the paged and non-paged DISCID systems.

2) Starting up

From the CCP prompt (A>) enter CONFIGUR and press newline. When running the program, at any point the user may abandon the program by typing cntrl/C, and no change will be made. When answering a question the CP/M line editor can be used to correct answers. Once newline has been pressed it is not possible to correct any errors. The XSUB utility may prove useful.

3) Configuring the discs

3.1) How many disc drives do you wish to configure

Enter the number of disc drives there are in your system. This will usually be a maximum of 4, but may be more if you have a hard disc or have a RAM disc facility. If you specify more than there are the floppy disc controller will not object, but there will be a strange delay of a few seconds on cold booting while the controller attempts to recalibrate the unattached drives. From now on the questions will be asked for each drive.

3.2) What is the physical drive number (Usually n)

This gives the hardware drive number to be used to access the drives. Usually you should respond with the suggested value, but if you have hard discs or RAM disc facilitys they may be different.

3.3) What is the gap length (usually 42)

You should always enter 42 unless you have other disc controllers.

3.4) What is the gap length on formatting

You should always enter 40 unless you have strange disc drives.

3.5) What is the special parameter byte of the drive (usually 0)

This will be 0 usually though setting certain bits have certain interesting effects. Setting bit 0 will cause data to be read and written to the disc in inverted format. This can be used for example when reading many SuperBrain disc formats. Setting bit 1 will cause the disc contoller to double the track number when seeking a certain track (though the cylinder number in the sector searched for will remain unchanged). This can be useful for reading a 48 tracks per inch (TPI) formatted disc with a 96 TPI drive. It is not recommended to write to a disc in this mode as it may not be read by a 48 TPI drive.

3.6) What is the seek rate for the disc in milliseconds

Consult the technical manual for the drive you are using and enter in the correct value. If in doubt 32 is safe but slow. If the drive has been supplied by Grundy Business Systems the value you require should be supplied.

3.7) What is the sector size (512 or 1024)

The disc controller and supports two sizes of physical sector size, 512 and 1024. However the software will only support a 512 byte sector unless an expansion box is used. Usually a 512 byte sector will be used.

3.8) How many sectors are there per track

This question refers to the number of physical sectors (NOT CP/M sectors) there are per track. If 5.25 inch discs are being used there is a maximum of 10 (which is the usual value) for 512 byte sectors and 5 for 1024 byte sectors.

3.9) What is the skew factor (usually 0)

Normally you will enter 0. This is because when you format a disc the controller will hard-format the most optimum skew factor onto the disc. However you may wish to use a software skew to read other formats of disc. For example many SuperBrain formats use a skew factor of 2. The skew factor is one greater than the number of physical sectors skipped between reading sectors on a track in the logically correct order.

3.10) How many sides does the drive have (1 or 2)

Enter 1 for single sided drives and 2 for double-sided drives. You may of course enter 1 for a double-sided drive if you wish to read a single sided disc in it. If you answer 2 you will then be asked:

3.10.1) Are the tracks numbered .....

If you have a double-sided drive you can either use the tracks on one side of the disc followed by the tracks on the other side, or you can use first a track on one side, then a track on the other. If you wish to use the former then reply 1, if the latter (which is more efficient and is the usual NewBrain format) then reply 2. Many alien formats (e.g. SuperBrain) do however use the former scheme.

3.11) How many tracks per side are there on the disc

To answer this question you should refer to the technical manual of the disc drive. For a 5.25 inch drive normally it will be 40 for a 48 TPI drive and 80 for a 96 TPI drive.

3.12) How many track are reserved (Usually 2)

The usual answer will be 2. The system information in the NewBrain system tracks take up 7K bytes, and the reserved tracks should be enough to hold this.

3.17) What is the blocksize in Kbytes ([1,]2,4,8, or 16)

Usually it will be 2 for 40 TPI single/double sided drives and 4 for 80 TPI double sided drives. Other format discs may have other blocksizes. If introducing a wierd format a rule of thumb is to use a blocksize that will give you between 128 and 255 blocks. The maximum number of blocks is 512. Once you have answered this question the program will tell you the disc capacity. This may be less than the advertised capacity for two reasons. Firstly it does not include the space for the system

tracks and secondly it will be a multiple of the blocksize and any odd amount is unused. This will be the value displayed when doing a STAT DSK:.

3.18) How many entries are there in the directory (multiple of n)

Usually this will be the lowest number possible. If it is greater than 128 then a message will be output which states that only 128 entries will be checked. This means that the CP/M system will only sumcheck the first 128 directory entries in its system which detects whether the discs have been changed surreptitiously.

3.19) Is the disc removable (1=yes, 2=no)

Usually answer 1, but for non-removable hard discs and RAM disc facilities it may be more efficient to answer 0. If you answer 0 the CP/M system will not use its directory sumcheck system to detect whether the discs have been changed.

4) Configuring devices

This information is only used in the 64K CP/M system, to describe the devices that are attached to the CP/M logical devices (IOBYTE is not used). Stream 0 refers to the keyboard, stream 1 refers to the screen, stream 2 refers to the printer, stream 3 refers to the reader/punch, and stream 4 is for keyboard status.

You should respond as follows (null means just newline):

Stream 0 - 19 (may be 6 on some prototypes), 0, null
Stream 1 - 18, 0, null
Stream 2 - 16, 0, LO
Stream 3 - 17, 0, *0
Stream 4 - 3, 0, null

5) Finishing at the end it will ask you which drive you wish the configuration information to be written to. This is your last chance to press cntrl/C. If A then reply 0, if B then 1 etc. The information will then be written to the disc and the program will finish.

CSP 13/4/83

Description of the NewBrain DISCIO system issue 1.0

Introduction

The NewBrain DISCIO system provides the ability of assembler programs or high-level language programs (e.g. BASIC, COMAL) to use the facilities of the CP/M 2.2 disc filing system, without actually operating under CP/M. This is done by providing 4 device drivers, devices 12-15, which when called in the correct manner provide the program with serial and random access to disc files, and the ability to read and update the directory. These devices are available under both the paged and non-paged NewBrain operating systems, provided that the disc controller box is present in the system.

DISCIO files

Since DISCIO uses the CP/M 2.2 disc filing system, the filenames used are the same as used by CP/M 2.2. That is an optional drive identifier which is a drive letter followed by a colon, a 1-8 character filename optionaly followed by an extension which is a dot followed by 1-3 characters. Extra characters in either field are ignored. Examples of filenames are FRED, PROGRAM.BAS, and B:CODE.COM. When opening a file you will have to specify the filename. In some special circumstances you can use ambiguous filenames. In these cases, a '?' is a 'wildcard' character, and a '*' specifys that the rest of the field is to be completed with wildcard characters. For example "Z?" will match Z1, ZT, Z2 etc., and "*.COM" will match CODE.COM, ASM.COM, STAT.COM etc. "*.*" matches all files. When an unambiguous filename is required there should be no occurrences of "*" or "?". All letters in filenames are converted to capital letters automatically.

Device 12 - BDISCIO

This provides a completely binary file serial I/O facility. When opening the device the parameter string should contain an unambiguous filename. The bytes of the file can then be read or written, depending on whether the file was opened for input or output. An attempt to read from a file opened for output or write to a file opened for input will result in no action. An attempt to read past the end of file will result in an error. Note however that the CP/M filing system does not make note of the end of file and so the end of file error will not be raised until the last sector has been entirely read. Device 12 is recommended for saving and loading BASIC and COMAL programs, and for reading in assembler programs assembled by a CP/M assembler into a COM file. All output file streams must be closed. Closing an input file has no effect but it is advised that programs should do so.

Device 13 - TDISCIO

Device 13 is extremely similar to device 12, but operates in a way so that text files created by a CP/M editor or word processor can be read or written. When outputing to a file opened by device 13 a line-feed is inserted after every carraige return, and 1AH is treated as a file terminator which is output when closing the file. Likewise when reading from a file opened by device 13 line-feeds are ignored and 1AH is treated as an end of file indicator. When end-of-file is reached 1AH is returned when INPUT is called.

Device 14 - RDISCIO

Device 14 provides the facility of random file access. Its power is however far greater than that of CP/M random file facility as it provides for records of variable size from 1 byte to the maximum 8 Mbytes to be read or written at any position within the file. Files are opened in the same manner as in devices 12 and 13, though in this case it is immaterial as to whether the device is open for input or output.

Once a file has been opened by device 14, any sequence of reads or writes can be performed. Before each is done a six-byte header sequence must be sent to the device. This consists of two three byte binary numbers of which the most significant bytes should be sent first. The first number specifys the position within file that the I/O is to commence from, and then the length of area that is to be transputed. Neither of these numbers should have their most significant bit set. Whether the transputation is an input or an output is determined by the next I/O to be performed. The program then must either read or write the number of bytes that he specified the length of the transputation to be.

When transactions with a file have been completed, then it must be closed if it has been updated, and should in any case be closed. It is worth noting that the CP/M filing system does not provide for the deletion of areas no longer required in a random file, so if device 14 is to be used for a data base facility then either the data base system should reuse redundant areas or else should provide for copying to a new restructured file and deletion of the old file.

Example:

```
10 OPEN#14,14,"TEMP"
20 INPUT("STARTING ADDRESS - ")X:GOSUB100
30 INPUT("LENGTH           - ")X:GOSUB100
40 LINPUT("INPUT OR OUTPUT?")A$:IFA$="O" THEN 60
50 FORA=1TOX:GET#14,Y:PUTY:NEXTA:PUT13:GOTO20
60 LINPUT ("STRING?")A$:FORA=0 TO X-1
70 PUT#14,MID$(A$,A+1-(INT(A/LEN(A$))*LEN(A$))):NEXTA:GOTO20
100 PUT#14,X/65536,(X/256)AND255,X-(INT(X/256)*256):RET
```

Device 15 - SDISCIO

This provides the facility of reading and updating the directory. When opened no parameters are required and the port number should be 0. The program interacts with device driver by means of conducting transactions. The program will first write a decimal digit to the device to indicate what kind of transaction is required. Each has its own number. When passing parameters to device 15, spaces, TABs, and newlines are valid separators. When receiving results from device 15, a newline character will mark the end of the result. The rest of the transaction depends on the number. The various transactions are:

Type 0 - Deletefile. Takes as argument an ambiguous filename and deletes all files matching it. No result is returned.

Type 1 - Renamefile. Takes as arguments two unambiguous filenames, and causes the first file to be renamed as the second filename. No result is returned.

Type 2 - Select disc. This takes a capital letter as argument indicating the new default drive for all subsequent disc operations. No result is returned.

Type 3 - Search for first. This initiates a directory search process. This takes an ambiguous filename as argument and returns the first filename matching it. If no match is found then a null string is returned.

Type 4 - Search for next. This must only be called after a type 3 or 4 transaction. It takes no argument but returns the next matching filename. If no further match is found then a null string is returned.

Type 5 - Return default disc. This takes no argument but returns a letter giving the current default disc.

Type 6 - Reset disc system. This takes no argument and no result. This resets the CP/M filing system.

Separators between arguments can be commas, newlines, or tabs. Results returned will be terminated by a newline.

Example of use: To obtain a full directory of a drive.

```
10 LINPUT ("Which drive do wish a directory of ? ")a$
20 a$=CHR$(ASC(A$) OR 32)
30 IF a$<"a" OR a$>"d" THEN PRINT "Drive must be A t  .":33TI ..:
40 OPEN#15,15:PRINT#15,3,a$+":*.*"
50 LINPUT#15,a$:IF a$<>""THEN PRINT a$,:PRINT#15,4:GOTO 50
60 CLOSE#15:END
```

DISCIO Modularity

The DISCIO module requires three subroutines to be provided as external routines. INITCPM, ENDCPM, and BDOS. The first two are called when any of devices 12 to 15 are opened or closed. BDOS is expected to perform the same function as the CP/M BDOS function. This provides for the use of the DISCIO module with a CP/NOS system to allow access to the central filing system within a network, or whatever CP/M compatible DOS system is used.

Technical note

When any DISCIO device is opened the CP/M filing system is introduced, and by calling INITCPM or ENDCPM a use count will be maintained. If the use count falls to 0, the space taken by the filing system may be relinquished. There will not be two copies of the filing system at any one time. In the non-paged system the filing system will take up about 4.5 Kbytes. In the paged system the DISCIO system claims an entire page for its own use.

Error numbers

150 - Bad filename. The filename provided as parameter when opening device 12, 13, or 14 or a device 15 argument is syntactically incorrect, or is ambiguous when it should not be.

151 - Input error. Either a physical disc error has occurred or the end-of-file has been reached on device 12.

152 - Output error. Either a physical disc error has occurred or the disc has become full.

153 - Directory error. A file requested does not exist in the directory, the directory is full, or there was a physical disc error when reading the directory.

154 - Initialisation error. The CP/M filing system could not be loaded, either due to lack of memory, or a disc fault.

155 - Transaction error. With devices 14 and 15, inputting when an output was expected, or vice versa.

156 - Not port zero. Opening device 15 with a non-zero port.

157 - Filename too long. A filename presented to device 15 is too long.

158 - Transaction type error. An transaction number presented to device 15 is illegal.

159 - Random transput error. Attempt to access a file beyond 8 Mbytes, or a length of zero has been specified.

With some physical disc errors, a message may be output to the console in addition to the error condition being raised.

CSP (6/5/83)

Issue ϕ.1 9/4/83 This supercedes all previous issues.

NewBrain paged memory operating system

1) Introduction

The NewBrain paged memory operating system allows the NewBrain to use more memory than the normal Z80 64K address space allows. Much of the design in terms of I/O has been taken from the non-paged operating system. The design was based on the necessity for highly efficient memory access and management, and for efficient program environment switching.

2) Hardware

The hardware which enables the NewBrain to access paged memory is held in the NewBrain expansion box. Memory whether ROM or RAM is divided into contiguous PAGEs of 8K. A distinction is made in the hardware, which is reflected in the software between ROM and RAM pages. The Z80 can access at any one time 64K of memory, which is divided into 8 8K SLOTs. Any page can be brought into any slot. No movement of data takes place when this occurs, but the hardware is made to dynamically change the hardware implemented address translation function which converts the Z80 generated address into a system highway address.

The hardware address translation function takes an additional bit as input. This bit is called A16 and may be generated by any active hardware element (i.e. main processor or DMA) using the hardware address translation function. This provides for an additional 8 slots called the ALTERNATIVE SLOT SET. In the case of the main processor the setting and unsetting of this bit causes a complete change in the pages that it can access. The use of these alternative slots is analagous to the Z80 alternative registers and they are reserved for system use.

3) Fundamentals

3.1) Slot allocation

The allocation of slots changes dynamically according to the required environment of the currently executing program. It is usual however for slot 0 always to contain the SYSTEM PAGE. This is a page of RAM whose data structures are fundamental to the system operation. In what follows it is assumed that this page is always in slot 0.

3.2) Preservation of program state

Normally the state of a program is saved when a subroutine is called by pushing the registers onto the hardware stack. In a paged system this is not adequate, as the subroutine may wish to access its own pages of workspace. In order that the pages that the calling routine was using are still accessible when the subroutine has been finished, the subroutine may decide to save the pages that the calling code was using on the stack and restore them afterwards. It is not possible however to read the hardware to discover which pages are currently switched in. To get round this there are in the system page 8 2-byte locations called S0 to S7 where the numbers of the pages which are currently resident are kept. All programs are must therefore update these locations if they bring in new pages and perform system calls (see below). Also when performing a system call a program must be aware of which slots the system call preserves, destroys or passes information in.

3.3) Format of locations S0 to S7

Each Sn location contains a two byte quantity which contains what was written to the page store when the page was placed in slot n. Bits 13 to 15 must always contain the value n. Bit 12 is always 0. Bits 0 to 11 contain the page number. When writing to the page store the two byte quantity is presented on A15-A8 and D7-D0.

Example: Preserving and restoring a page in slot 7.

```
     LD HL,(S7)
     PUSH HL
     .
     .                    ;Code that uses slot 7
     .
     POP HL
     LD B,H
     LD C,PAGEREG
     OUT (C),L
     LD (S7),HL
```

3.4) The concept of a page number

As stated above the page number is a 12 bit quantity. However in the case of ROM, since Z80 code is not relocatable, any given routine will only run correctly when placed in a certain slot. It is therefore customary to use the term page number of a routine to mean a 16 bit quantity, containing the appropriate slot information in the top 3 bits. In the case of RAM, the slot bits are set to 0 and are added as needed when bringing in a page.

3.5) Operating system calls

The facilities of the operating system are made available to the program by means of a SYSTEM CALL. There may be up to 256 available system calls, each identified by a ZCODE. A program may make a system call by using the Z80 restart instruction RST 32, followed by the Zcode. Control is passed to the byte following the Zcode on completion of the system call. If a RST 40 is used, it has the effect of a CALL NC i.e. the system call is made only if the carry flag is clear, otherwise there is no action taken. The alternative registers are destroyed in a system call and cannot be used for sending parameters or receiving results. There are also some system facilities which can always be accessed by means of a normal Z80 CALL instruction, but unless otherwise stated, all the system calls are performed by the restarts. System calls that are performed with the restart are all brought into slot 7, which is always preserved. Slot 7 cannot therefore be used for passing information in these cases.

4) Memory management

The paged memory operating system allows a rich variety of ways in which a program may obtain workspace and construct data structures, though it is envisaged that most requirements will be satisfied by the use of the memory management system or MMS.

4.1) The Memory Management System

The MMS performs memory management operations allowing the program to conveniently use the full paged memory capabilities of the NewBrain. The fundamental concept of the MMS is the OBJECT. An object is a data record of any size between 0 and 65520 bytes. The size of the object is fixed once it has been created but it contents may be changed at any time.

Associated with each object is an OBJECT NUMBER, which is between 1 and 65535 (there cannot be an object whose number is 0 so 0 can be used as a 'null' pointer in creating data structures). An object will always have the same number until it is deleted, and can only be refered to by means of that number. Object numbers are, however, purely transient quantities. That is to say that is a data structure is stored on a backing store and then restored, the MMS is unlikely to allocate the same numbers to the objects. Therefore when such a data structure is being placed on the backing store a different system of refering to the constituent parts of the data structure should be employed if the structure of the data structure is to be maintained (this does not of course invalidate the use of data structures on backing stores themselves e.g. indexed sequential as the pointers used on these are positions on the backing store medium).

An object may be referenced by the system calls GETOB and IGETOB (used by a Z80 CALL instruction). For GETOB, on entry HL is the object number. On exit the object is placed in slots 1 and 2, and the Z80 location of the first byte of the object is in HL. Registers BCDE are preserved. Only the first 8180 bytes of the object are guaraunteed to be available. IGETOB performs much the same function as GETOB but takes an extra parameter in DE which is the required offset within the object. On exit HL is the Z80 location of the object and its offset. Only the subsequent 8180 bytes of the object are then available. The object (or part of) only remains available until a subsequent call to the MMS, or until a system call that requires memory allocation (e.g. opening a stream).

An object is created by NEWOB, which requests the creation of an object of size BC bytes. On exit carry will be clear and HL will be the object number given by the MMS to that object. Also a NEWOB also performs an implied GETOB and the location of the first byte of the object is in DE on exit. BC is preserved. If however the carry flag is set on exit then an error has occurred and an error number is left in A (the object will not have been created).

An object is deleted by DELOB, which takes in HL the object number of the object to be deleted. BCDEHL are preserved.

All MMS calls destroy slots 1 and 2.

Although the MMS has a simple interface it has great power, and is used by the paged input/output system (see below) for creating buffers for device drivers, and all the data structures for the NewBrain COMAL system other than its run-time stack.

4.2) Video memory allocation

Only certain parts of the RAM have the necessary hardware associated with it so that it can be used for screen display files. Thus a separate system is used to allocate video memory areas (other memory allocation systems will use non-video memory before using video memory). Like the MMS there are NEWVIDEO, DELVIDEO and GETVIDEO functions available. In addition there are one-byte video object numbers. It is impossible for there to be a video area number 255. It is recommended that video memory allocation only be used for video memory purposes. The size of areas requested will be rounded up to multiples of 128 bytes.

GETVIDEO brings into slot 4,5,6, and 7 the video area whose number is in A, returning in HL the location of it. BCDE preserved. GETVIDEO resides in the system page and is called with a Z80 CALL instruction.

NEWVIDEO allocates a video memory area of size BC bytes, returning with carry set and A=error number on failure, otherwise A=video object number. BCDEHL preserved.

DELVIDEO deletes video area number A. In addition certain pointers in the system page are updated if they point into video areas moved as a result. BCDEHL preserved.

4.3) Page allocation

This is a facility whose use should be extremely sparing. There will be times when a program will require a page of workspace, e.g. MMS when its current pages are exhausted. A system call REQUESTPAGE is provided for this purpose. It should be stressed that this is the lowest level of overall page allocation and when this facility is used the program has to bear in mind two things: 1) The program maintains to perform his own garbage collection and may be required to move his pages from one place to another on request from the operating system (the exception to this being the user program) and 2) Page operations can be extremely slow - a page request can cause a complete system garbage collection to take place.

4.3.1) The region system

All RAM pages allocated, other than the system page, all belong to a region. There are 16 general purpose regions (numbered 0-15). The pages belonging to each region have contiguous numbers, and have higher numbers than those of a lesser numbered region. When a request for a page is made, the region number must always be specified. A special region (number 255) used by the video memory allocator, which is entirely separate as the video supporting RAM has the highest page numbers.

4.3.2) The region table

The region table is kept in the system page and contains the status of each region. Each region has a six-byte entry. The first two bytes are the page number of the lowest page belonging to that region (the end of the region is found from the subsequent entry). The other four bytes are the real address of the region manager (page number followed by location in page). The region manager consists of two routines, GARBAGECOLLECTOR and MOVEUP. The MOVEUP routine is assumed to be at a location three bytes greater than that of the region manager, which is where the GARBAGECOLLECTOR routine resides. The MOVEUP routine may be omitted in the case of region 0. There is no entry for region 255. At the end of the table is a dummy entry to mark the end of region 15. This is also called REGTOP.

4.3.3) Responsibilities of the region manager

The MOVEUP routine is called with A equal to the number of pages that the region must move up. The region manager should merely move his pages up the requested amount and update its data structures if necessary. When called the base pointer will not have been updated though the pointer for the regions above it will have. The main registers can be destroyed.

The GARBAGECOLLECTOR is called with A equal to the number of pages the region has to move down. The region manager should move the region down that number of pages, and also attempt to reduce the memory required by it by performing a garbage collection or whatever is appropriate. When called the base pointer will have been updated but not that of the region above. On exit A should be the number of pages that the end of the region has moved (this is then passed on to the region above for its garbage collector), but the other main registers can be destroyed.

4.3.4) Operation of the page allocator

When REQUESTPAGE is called, A is the number of required pages and B is the region number. On exit the carry flag is set if there was a failure, and A contains an error number. Otherwise no result is returned. BCDEHL are preserved.

The algorithm used by the page allocator is 1) Check that the space exists 2) If not call all the region managers (low regions first) to collect garbage and check again (if there is still insufficient space then error) 3) Call the region managers of those regions with higher region number than that of the region in which the page is to be allocated to move up (high regions first). If a region has no pages then the region manager is not called.

A common error in writing region managers is that the data structure in the region is left in an inconsistent state when calling REQUESTPAGE for the region, and a possible resulting call to the GARBAGECOLLECTOR of the region manager operates incorrectly. Attention should be paid to this point.

4.3.5) Region number allocation

There is no dynamic region allocator - region numbers are fixed. Region 0 is reserved for the user program. Region 1 is used by the DISCIO system. Region 8 is used by the MMS. Other regions with even numbers are reserved for future use, and odd numbered regions are for user use.

5) Paged I/O system (PIOS)

The PIOS is very similar to its non-paged counterpart IOS, particularly in terms of the various interfaces and the available system calls. The PIOS provides access from a program to the various device drivers. The central concept in this is that of the stream. Programs transact with a stream, and the PIOS translates the transactions into physical device driver calls.

5.1) User program interface

OPENIN and OPNOUT opens device A port D on stream E with parameter string length BC at HL. The length of the string should not be greater than 256 and should not reside in slot 7. On exit if carry is set then the open was a failure and A contains an error number. Otherwise BC is the length of the returned result string at HL. DE preserved.

INPUT, OUTPUT and CLOSE perform the said operation on stream E, receiving data or returning data in A if appropriate. If carry is set on exit then an error has occurred and A contains an error number.

BLKIN and BLKOUT input or output a string to or from HL length BC from or to stream E. No part of the string should reside in slot 7. DE preserved. If an error occurred carry set on exit and A contains an error number.

5.2) Interface from PIOS to device driver

All device drivers will be brought into and entered on slot 4. A device driver has a number of entry points which can be called by PIOS or specified not to exist. The first byte of a device driver should be one less than the highest entry number. The following bytes should contain the offsets to the actual entry points of the routines. These should be in turn OPENIN, OPENOUT, INPUT, OUTPUT and CLOSE (MOVE is not used by PIOS). For example:

```
DRIVER:   DEFB 4                ;Number of entry points less 1
          DEFB DOPENIN-$        ;DOPENIN is code for open for input
          DEFB DOPENOUT-$       ;DOPENOUT is for open for output
          DEFB DINPUT-$         ;DINPUT reads a byte from device
          DEFB DOUTPUT-$        ;DOUTPUT outputs a byte
          DEFB DCLOSE-$         ;DCLOSE close the device
```

All the routines may destroy IX.

INPUT, OUTPUT and CLOSE are called with HL pointing to the stream buffer, A is the data if appropriate, E is the stream number and D is the port number. OUTPUT returns a result in A. If they wish to report an error they should leave carry set and an error number in A on exit.

OPENIN and OPENOUT are called with D as the port number and E the stream number. Also BC is the length of the parameter string at HL, and one can similarly be passed back on return. The maximum length of this string is 256 bytes. If they return no string set BC to 0 on exit. If they wish to report an error they should leave carry set and an error number in A on exit.

The system call MKBUFF may be done by the OPENIN and OPENOUT to create for themselves a buffer. The maximum size of this buffer is 8180 bytes. On entry E should contain the stream number and BC the size of buffer required. On exit HL will point to the created buffer and BCDE preserved.

All device drivers must preserve all the slots.

There is an additional system call GETSTREAM, which brings in the buffer of stream E into slots 1 and 2, and leaving its location in HL. On exit E is preserved, B is the device number, D is the port number and A and C are swapped from what they were on entry. Carry is set on exit if the stream was not open.

6) System modularity

There are two orthogonal levels of modularity in the system. The first is the manner in which the elements of the system act independantly of each other - the page allocation, allocation within regions (e.g. the MMS), PIOS, device drivers etc. The second is the modularity of each ROM. When a ROM is added to a system, its presence is found by the system initialisation routine and its facilities are augmented to the system as a whole irrespective of whatever other systems are available. To this end it is crucial that the code of each ROM makes no assumptions about the hardware and software environment other than those allowed. These are the existence of system calls, device drivers, and the contents of the system page are all that should be assumed.

6.1) ROM headers

To aid the initialisation process, each ROM should have a header. The format of the header is that of a list of items. The first byte of each item is the type of the item and is followed by arguments.

Type 1 is followed by an address of a subroutine. This subroutine should be in slot 5 code and when entered DE contains the page number.

Type 2 defines a system call and is followed by the 2code and the address of the routine (system call routines should be in slot 7 code).

Type 3 defines a device driver and is followed by the device number and the address of the device driver (device drivers should be in slot 4 code).

Type 4 is followed by a one byte argument and the initialisation routine skips that many pages. Type 0 and 6 to 127 causes the following two bytes to be skipped. 128 to 255 signifies the end of the header and causes the next page to be moved to.

Type 5 defines a user program and has a two byte argument which points to a string giving the title of the program (which is terminated by a byte with bit 7 set), and after the string is the entry point of the user program.

Type 6 defines a library routine and has a two byte argument. See below for the format of this header.

Any item begining with 0 or 7-127 is assumed to have a two byte argument and is ignored. (0 is used for ROMS which act under the non-paged operating system as well). Any beginning with 128-255 marks the end of the ROM header.

6.2) Scanning ROMs

An aid to ROM scanning is provided by the system call NEXTPAGE. On entry if carry is set then it returns in DE the page number of the first ROM, otherwise if carry is clear on entry DE contains the a page number and on exit DE is the next page number (carry set on exit if there are no more ROM pages). In addition the page is placed in slot 5 and on exit HL is poining to the first byte of the page. The page number will always be a slot 5 page number. This routine can be run with interrupts enabled or disabled as the IFF flags are preserved.

.) How to write a user program

7.1) Slot allocation

When you write a user program to run under the paged memory operating system, the first design decision you must make is how to allocate your slots. If you want you can use slot 0, but if you do you have to make up for its absence. This is done for example when CP/M is being run. This is not recommended unless absolutely necessary. So we shall assume that the system page is left in slot 0.

If you wish to use the memory management system then you will normally leave slots 1 and 2 to access the MMS objects.

7.4) Aids to user programs

7.4.1) EXIT

There is a system call EXIT which does not return but causes the user program to be abandoned the main menu to be returned to (or whatever the initial program is). A garbage collector will be set up to deal with the left over pages. All the streams are then automatically closed and SP reset.

7.4.2) USERPAGE

The user program has the option of using a system call USERPAGE to obtain region 0 pages. If this is not done the user program must run region 0 itself by providing a region manager, bearing in mind that region 0 is not necessarily empty when a user program starts up. On entry HL should contain the number of required pages. On exit carry is set if there was an error, or else HL contains the page number of the lowest numbered region 0 page. Subsequent calls to USERPAGE should have in HL the total number of required pages. New pages given to region 0 will have page numbers higher than those already given, which cannot move.

8) Library facilities

8.1) ROM headers

A library routine is introduced to the system by a ROM header type 6 item. This takes a two byte argument which points to a string giving the name of the routine (the last character has its top bit set) which should be in capital letters, starting with a letter and being followed by a sequence of characters which are either letters or digits or '.' or '_'. When choosing a name of a library routine it is worth bearing in mind that because of the curious way in which BASIC works, any part of the name which contains a BASIC keyword will become entokenised, thus causing that routine not to be accessible from BASIC.

Following the name is the parameter definition string. The first byte is 0 if there is no result otherwise bit 7 is set and bit 6 indicates if the result is a string rather than numeric. The second byte contains the number of parameter (the length of the string is two greater than this number). The following bytes describe the nature of the parameters. Bit 7 is set if the parameter is to deliver a result and bit 6 is set if the parameter is of string rather than numeric type.

Following the parameter definition string is the first byte of the library routine.

8.2) FINDLIBRARY

A system call FINDLIBRARY accessed via RST 20H takes as parameter HL which points to a non-slot 7 string length C. This string contains a library name. On exit carry will be set if the library search fails. Otherwise DEHL will be the real address of the library, BC points to a string whose first byte has bit 7 set if a result is returned (i.e. the call can be used as a function) in which case bit 6 indicates the type (0=numeric, 1=string), the second byte is the number of parameters and subsequent bytes indicates for each parameter whether it can be used to return a result (bit 7 set) and whether the type of the parameter is string or numeric (bit 6 set if string).

8.3) ENTERLIBRARY

When the library function is entered by a system page call ENTERLIBRARY which preserves slot 7 takes parameters BCDE is real address of the library routine HL points to the parameter area. This will consist of a list of two-byte pointers (one per parameter with the first one first) each of which will be followed by a length if they are strings. The pointers point to either a six-byte floating point number or string. The library can overwrite these at will and if it returns results via parameters should on exit leave the parameter area accordingly. If the call is a function it will return in HL a pointer to the result (length in BC if result is a string). Naturally the function cannot overwrite a parameter string with a longer one unless it gets it room elsewhere by some means or other for the new string. If the library returns an error carry will be set and A will contain an error number. All pointers are non-slot 7 pointers.

CSP 9/4/83

System Page description issue 0.0 26/4/83

To PARS add :

E1 - E3 DISCSTATUS

System page description - issue 0.0

Introduction

This document describes the contents of the system page under the NewBrain paged memory operating system and also the fixed memory areas used by the NewBrain in the non-paged operating system.

Notation and nomenclature

OS - non-paged operating system.
POS - paged memory operating system.
Bn - bit number n where 0 is the least significant bit.
P - paged.
NP - non-paged.
MMS - memory management system.
IOS - input/output system (in OS).
PIOS - paged input/output system (in POS).
Object - a memory area allocated by the MMS.
COP - the COP 420 input/output microprocessor.
Page - an 8K area of the paged memory highway address space, all of which is addressible by a single hardware page address.
Page number - a value which is used to access a unique page. This is a two byte number of which B12-B15 are 0.
Slot - one of the eight 8K areas in the Z80 address space into which a page may be brought in.
System page - a page which is usually in slot 0 containing fundamental system information.
Slot n page number - A two byte number where B0-B11 are the page number and B13-B15 are n. B12 is 0.
Location - the Z80 address which can be used to access an area of memory when the page it is on has been placed in an appropriate slot.
Region - a set of contiguous RAM pages used for a similar purpose as allocated by the page allocator.
Video address - a location within the 32K of video RAM. B15 is usually 1.
Streched format - the bytes of a table entry are spaced 256 bytes apart. The first bytes of consecutive entries are adjacent.
Not used - reserved for future expansion.

PARS

The PARS area exists between locations 0 and FF in both OS and POS. This area is allocated in the following way:

System page description - issue 0.0

0 DI and 1 - 3 TPON. NP contains the instructions DI, JP PONINT (an address in the EF system ROM). P contains DI, JP 0. This is designed for the use of NewBrains with power-down facility. POS does not support this facility, and is not supported by the hardware in models A and AD.

4 - 5 B3PRM. NP is the address of the start of the RAM available to BASIC and is adjusted by OS as the IO buffer area increases and decreases. P is available for use by the user program. BASIC sets it to 2000H.

6 - 7 B4. NP is the address of the end of the RAM available to BASIC. P is available for use by the user program. BASIC sets it to C000H.

8 - A TRST8. Available for temporary use by any module. 8 contains C3 (JP) and must be preserved.

B DEV0. Contains the default console device number which may be used by user programs when opening up a console.

C EXCESS. This is set to 24 and is only used in OS and should not be changed.

D TV0. Cursor flash clock used by the frame interrupt handler.

E TV2. TV display status flags. NP B7 set means display screen, B6 means display cursor, B5 means cursor is underline not block, B4 means switch off video for next frame only, B3-B0 unused. P B7 set means display cursor, B6 means cursor is underline not block, B5 means switch off video for next frame only, B4-B0 any set will force video to be switched off (at the moment B0 is used by the video device drivers and B1 is used by the video memory allocator and the others are unused).

F TV1. Contains the character at the cursor position.

10 - 12 TEXM. Restart for use by user program (used by BASIC). 10 is set to C3 and should be preserved.

13 DEV2. Contains default backup storage device number for use of user programs (e.g. BASIC when doing LOAD "program").

14 SAVE2. NP Temporary use by IOS. P Not used.

16 SAVE3. NP Temporary use by IOS. P not used.

18 - 1A TEXMNC. Restart for use by user program (used by BASIC). 18 is set to C3 and should be preserved.

1B PLLEN. Line length of stream 0 device (if not supported will be 0).

1C PHPOS. Print head position of stream 0 device (left hand side is 1). This is only operative if (PLLEN~=0).

1D PZLEN. Print zone length of stream 0 device. This is only operative if (PLLEN~=0).

1E SAVE1. NP OS workspace. P not used.

20 - 22 TRST32. System call restart location.

23 BRKBUF. If this is non-zero all breaks (STOP key and * during cassette lead-in read) will be ignored.

24 ENREGMAP. The last byte sent to enable register 1 (see hardware definition).

25 IOPUC. Use count of open devices that require IOPOWER.

26 - 27 UP. Location of user program. NP this will usually be BASIC. P this is the location of the main menu program which is always a slot 6 location.

28 - 2A TRST40. System call on carry clear restart location.

2B KBMODE. This is used by KLOOK when performing keyboards decoding and is used by all open device 0, 3 and 4 streams, which share a common keyboard mode.
2C - 2D GSPR. NP Address of user program make space routine. P not used.
2E - 2F RSPR. NP Address of user program return space routine. P not used.
30 - 32 TRST48. Restart reserved for interrupt handling routines. 30 contains C3.
33 - BUFLG. B7 set means the paged operating system is in operation. The other bits are unused except B0 in NP which is set by IOS when buffers are being moved causing the video to be switched off.
34 - 35 DPSP. Location of default console device open parameter string.
36 - 37 DPSL. Length of default console device open parameter string.
38 - 3A TINT. Interrupt mode 1 interrupt handler. 38 must be kept as C3.
3B COPCTL. Acknowledge byte sent to the COP immediately after the next COP interrupt.
3C COPST. COP status. B7 unused, B6 set if keyboard data present in COPBUFF that has not been read in, B5 set if no command is waiting to be sent to the COP, B4 is set if a cassette error has occurred, B3 is set if * can cause a cassette break, B2 indicates status of break key, B1 indicates a cassette break or battery low, B0 is 0.
3D COPBUFF. Contains keyboard and cassette data, and cassette error numbers.
3E - 4D OUTBUFF. Data to be sent to single line display.
4E - 4F TIMBUFF. Not used.
50 - 51 CHKSUM. Tape checksum count.
52 - 55 CLOCK/CLACK. COP regular interrupt count. Used to seed random number generator. Least significant byte is first.
56 - 57 STRTAB. NP address of IOS stream table. P not used.
58 - 59 DEVTAB. NP address of IOS device table. P not used.
5A - 5B TVCUR. NP address of cursor position. P video address of cursor position.
5C - 5D TVRAM. NP address of IOS buffer of currently displayed screen. P video address of currently displayed window of currently displayed screen.
5E - 5F OTHER1. Unused but is updated by IOS in NP if points into a moved buffer and by the video memory allocator in P if is a video address in a video area that is moved.
60 - 61 OTHER2. Same as OTHER1.
62 - 63 IOSRAM. NP Address of base of IOS buffer area. P not used.
64 - 65 STRTOP. NP Address of end of IOS stream table. P not used.
66 - 68 TNMI. Non-maskable interrupt handler. 66 contains C3.
69 - 6B FICKLM. Frame interrupt count providing accurate 50 Hz clock. Most significant byte is first.
6C - TBRP. Default transmit baud rate parameter. (Divide 19200 by the value to give the baud rate). Optionally used by some device drivers.

6D - RBRP. Default receive baud rate parameter. As TBRP.
6E - 70 IOPON. Increment IOPOWER use count routine.
71 - 73 IOPOFF. Decrement IOPOWER use count routine.
74 - 75 DEFNF. Default numerical output format. This value is a format as required by the maths pack OUT routine.
76 STR11. Default high resolution graphics stream number.
77 - 78 CHRROM. NP address/P slot 1 location of character definition table used by high resolution graphics text routines.

The following are only used in the paged operating system:

79 - 7A DISCBUFFER. Location of disc command block in whichever slot set it is supposed to be in.
7B - 7D INTSERVICE. Non-COP interrupt handling routine (in alternative slot set).
7E - 80 GETOB. MMS object retrieval routine.
81 - 83 IGETOB. MMS object retrieval with index routine.
84 - 86 GETVIDEO. Video memory area retrieval routine.
87 - 88 PARAMLEN. Temporary workspace by PIOS.
89 - 8A IOSTEMP. Temporary workspace by PIOS.
8B - 9A S0,S1,S2,S3,S4,S5,S6,S7. Sn contains the slot n page number of the page currently in slot n.
9B - 9E ENDOFOBJECTS. Used by MMS to mark the real address of the end of the last object relative to the beginning of the region.
9F - A0 VIDEOTOP. Page number of end of video RAM pages.
A1 - A2 VIDEOBASE. Page number of first video page.
A3 - A4 AVIDEOTOP. Address of the end of the last used video area relative to the beginning of the video area.
A5 - A6 UPPAGE. Slot 6 page number of page containing the main menu program.
A7 TV3. Contains the byte to be sent to the TV control register (see hardware definition) for the current screen display.
A8 - A9 ESYSPAGE. Location in system page of first unused byte.
AA - AC DISC. Disc operation activation routine.
AD OLDOS. Identifys the version of the non-paged operating system. 0 means 1.4, 1 means 1.9, 2 means 1.91, 3 means 1.9 FRENCH, 4 means 2.0.
AE - AF CHRPAGE. Slot 1 page number of page containing the character definitions used by the high resolution graphics text routines.
B0 - B1 INABROM. Slot 5 page number of internal AB ROM.
B2 - B3 INCDROM. Slot 6 page number of internal CD ROM.
B4 - B5 INEFROM. Slot 7 page number of internal EF ROM.
B6 ENREG2MAP.
B7 INTFLGS.
B8 - B9 BILL.
BA TV4. Contains video area number of currently displayed screen.
BB - BC PUSER. Number of region 0 pages that are required.
BD A16COUNT. Contains use count of alternative slot set.
BE - C0 ENTERLIBRARY. Routine to enter external library routines.
C1 - C3 INTBUFFER.
C4 - C6 GETBUFFER.
C7 - D6 BOXTAB.

System page description - issue 0.0

D7 - D9 SERVECOPINT - COP interrupt service routine.
DA - DB LOWBUFFERPAGE.
DC - DD HIGHBUFFERPAGE.
DE BUFMARGIN.
DF - E0 V2INTBUFFER.

The rest of the non-paged system fixed areas

100 - 19F FPRAM. Used by the maths pack as workspace.
1A0 - 267 STKRAM. Used as the hardware stack.

The rest of the paged system fixed areas

100 - 19F FPRAM. Used by the maths pack as workspace.
1A0 - 4FF STKRAM. Used as the hardware stack.
500 - 5FF MORPARS. Expansion to PARS area initialised to 0.
600 - 6FF STATUSTABLE.
700 - AFF TABLEZERO. This is the base table of the MMS data structure, which contains the real addresses of the secondary tables. See MMS documentation for details.
B00 - CFF STREAMTABLE. The entries are in streched format. An entry of 0 indicates the stream is not open. Otherwise the entry contains the object number of the buffer object. The first byte of this contains the device number, the second contains the port number and the rest of the object is the own memory area.
D00 - DFF PARAMAREA. This is an area used as workspace by PIOS and FINDLIBRARY.
E00 - 11FF PDEVTAB. This is the table of device drivers, which is in streched format. The first two bytes of each entry give the slot 4 page number of the device driver and the second two bytes give its location.
1200 - 15FF ZPTABLE. This is the table of system routines, which is in streched format. The first two bytes of each entry give the slot 7 page number of the device driver and the second two bytes give its location.
1600 - 167F VIDTABLE. This gives the video addresses and lengths of the 32 possible video areas. The first two bytes of each entry contains the video address and the second two contain the length. All video areas are multiples of 128 in size.
1680 - 169F ENTRYAVAILABLETABLE. This is a bit map table auxiliary to MMS. The bits are ordered with B0 of the bytes being for 0 to 31, B1 being for 32 to 63 etc. If a bit is a one it means that the secondary MMS table it corresponds to exists and has a free entry. See MMS documentation.
16A0 - 1701 REGIONTABLE. Each region has a six byte entry. The first two bytes correspond to the first page belonging to the region. The second two are the slot 4 page number of the region manager (if the most significant byte is 0 it means there is none), and the third two are the location of the region manager. The last two bytes of REGIONTABLE (REGTOP) are a dummy entry to mark the end of region 16.

1702 - (ESYSPAGE)-1 SLOTZEROCODE. This area contains various important system routines. If it is required to put in a new one then put it in at (ESYSPAGE) and increment (ESYSPAGE past it).

CSP. 26/4/83

SYSTEM PAGE

The SYSTEM PAGE is a page of RAM which normally resides in Slot Ø. It contains all the data structures necessary to system operation including:

1. the Status Table
2. the Page Store, i.e. the numbers of the pages currently resident.
3. the Region Table
4. access to operating system routines
5. GETVIDEO

STATUS TABLE

The status table is a 256 byte contiguous block ofmemory in the system page, the start address of which is fixed at Ø6ØØH.

The n'th byte of the table is owned bythe device driver and any associated interrupt handling routine currently connected to stream n, and is called the "status word" for stream n. It may be peeked by anything, but only altered by its owners. The status word is set to zero on system initialisation and whenever its owner is closed, if its owner supports the status word. (We say that a device "supports the status word" if it might alter it). The status word for stream n will have value zero if stream n is not open or if stream n does not support the status word or if the value of its bits happen to be zero.

The meaning of the bits are as follows:

bit	mnemonic	meaning if set
Ø	IEST	Input error status: the next call to input from this stream will produce an error in accordance with the usual behaviour of the device connected.
1	OEST	output error status: the next call to output to this stream will produce an error in accordance with the usual behaviour of the device connected.
2	NRIST	not ready for input: the next call to input from this stream may result in a long wait, or if the device is the sort which has returns immediately and has a conventional return for "no data returned" then this return will occur.
3	NROST	not ready for output: the next call to output to this stream may result in a long wait, or a loss of data.

bit	mnemonic	meaning if set
4	EOFST	end-of-file: the next input from this stream will result in a readpast end-of-file error.
5		
6		reserved for expansion.
7		

Note that a meaning is systemically attached to a status word bit only if the bit is set - the precise interpretation will depend on the device: for example, an error may still occur on reading even if IEST is not set. This also means that some devices might only partially support a status word: for example, unless there are other compelling reasons it is not necessary to assume that a device opened for input only, which would produce an error if an attempt were made to output to it, should keep OEST set.

KBBFIO documentation issue 1.0

1) Introduction

KBBFIO is configured as device 19 in the paged memory operating system and is only available when the disc controller module is connected. Its function is to provide an enhanced keyboard driver with an interrupt buffer, and the ability to use the control/0 to control/9 as additional function keys.

2) OPEN

Opening for input has the same effect as opening for output. The port number is ignored. A parameter string containing a decimal number may be given to select the size of buffer. If this is zero error 110 will be returned. Otherwise a buffer will be created out of that number of 256 byte blocks from the interrupt buffer management system. If no parameter string is given then a default buffer size of 256 bytes is used. An attempt to open device 19 twice will fail as the interrupt buffer management system will raise a fault.

3) INPUT

When reading from the device the characters will be read from the buffer according and returned. If no character is available the device will wait until a key is pressed. If a 0 is returned this indicates an escape and the device should be read again. If a further 0 is returned this indicates either a genuine character 0 or alternatively an invalid keyboard combination. If a 0 is followed by 90H to 97H, 8EH or 8FH, this indicates that control 0 to 9 respectively has been pressed. On some keyboards (e.g. English) control and '*' will yield a 0 followed 98H.

4) OUTPUT

Outputing to device 19 will change the keyboard mode as in device 5 and 6. However in this case bits 7, 5 and 4 are reserved for expansion, and bit 6 is an additional TTYCAPS bit. This will effect any characters which are already in the buffer, because characters are stored untranslated in the buffer. This does not effect the keyboard modes of any open device 5 or a stream nor screen editors common keybord mode byte.

5) CLOSE

Closing device 19 will deallocate the buffer. Any data in it is lost.

6) Polling

The status of the buffer may be tested by testing bit NRIST (2) of the appropriate byte in the system page status table. If this is 0 a character is available in the buffer and can be returned by device 19. If it is 1 the buffer is empty. This method of polling provides a much more efficient facility than afforded by device 6.

7) Interaction with devices 5 and 6

Keyboard data can still be read using devices 5 and 6 and the screen editor, without removing data from that which can be read by device 19. Thus keyboard input can be read twice over.

EXPANDED SYSTEM ERROR NUMBERS ISSUE 1.0 10.5.83.

This replaces all previous issues

1. CP/M

150 Bad filename. The filename provided as parameter when opening device 12, 13 or 14 or a device 15 argument is synatically incorrect.

151 Input error. Either a physical disc error has occured or the end-of-file has been reached on device 12.

152 Output error. Either a physical disc error has occured or the disc has become full.

153 Directory error. A file requested does not exist in the directory, the directroy is full, or there was a physical disc error when reading the directory.

154 Initialisation error. The CPM filing system could not be loaded, either due to lack of memory or a disc fault.

155 Transaction error. With devices 14 and 15, inputting when an output was expected, or vice versa.

156 Not port zero. Opening device 15 with a non-zero port.

157 Filename too long. A filename presented to device 15 is too long.

158 Transaction type error. A transaction number pesented to device 15 is illegal.

159 Random transput error. Attempt to access a file beyond 9 Mbytes, or a length of zero has been specified.

With some physical disc errors, a message may be output to the console in addition to the error code.

2. Paged Operating System

PIOS

104 String too long. The parameter string supplied to OPEN is longer than 256 bytes.

105 Stream not open. Stream refered to by INPUT, OUTPUT, BLKIN, or BLKOUT is not open.

106 Attempt to OPEN a non-existant device.

107 Attempt tp OPEN a device/port that is already open.

108 Attempt to open a stream that is already open.

109 Attempt to use a device function that does not exist.

MMS

160 Attempt to create more than 65025 MMS objects.

161 Run out of memory on NEWOB

162 Request for object larger than 65531 bytes.

VMA

170 Attempt to open more than 32 video areas.

171 Attempt to use more than 32K of video area or space available.

PAGE ALLOCATOR

199 Insufficient pages in system for REQUESTPAGE request.

SERIAL DEVICE

192 Read Past E.O.F.

193 Insufficient memory

CENTRONICS PRINTER

199 Printer Number out of range

120 Error in parameter string

206 Buffer does not exist

207 Buffer already in use

208 Buffer full or data could not be output to ACIA

209 Illegal use of multiplexed ACIA attempted.

210 Buffer empty or no data to input from ACIA

211 Attempt to run two ACIA's on the same peripheral interface at different speeds.

212 Attempt to run multiplexed ACIA at high speed.

213 Not used at present

214 Illegal buffer number

215 Illegal combination of data and stop bits and parity requested

216 ACIA framing error

217 ACIA overrun error

218 ACIA parity error

219 ACIA loss of carrier detected.

DEVICE DRIVERS AVAILABLE ON EXPANDED SYSTEM

ISSUE 1.0
10.5.83.

This replaces all previous issues.

1. CP/M

Device 12	BDISCIO Binary file serial I/O facility
Device 13	TDISCIO Text file serial I/O facility
Device 14	RDISCIO Random file access I/O
Device 15	SDISCIO Direct read and update

2. Expansion Interface Module

Device 16	ACLPIO ACIA Line Printer Device
Device 17	ACVIO ACIA Driver
Device 18	SSEIO Simple Screen Editor
Device 19	Not yet available Keyboard driver with buffers.
Device 20	SMDD Serial Memory Device Driver
Device 21	CENTR Centronics Printer Driver
Device 22	APIO Analogue I/O Driver

BASIC extensions with peripherals - issue 1.0

1) Expansion box

1.1) Memory

When running BASIC under the paged memory operating system the storage available for programs and data is 40K. This is not affected by the amount of I/O facilities used. This means that a large screen (up to 80 by 255) or a large high resolution graphics area, or the DISCIO system can be used without affecting the 40K available to the user. This will be between 2000H and BFFFH. Space may be RESERVEd as before at the top of the area which can be used for machine code routines as before. Such routines can make system calls as described in the systems documentation without pre- or postamble. The pages will always be present so that PEEK, POKE and CALL can be used as before. In addition the system page will always be present between 0 and 1FFFH and can be PEEKed and POKEd.

1.2) SAVF

A new command SAVF is available (syntax the is the same as SAVE). This will save the current program in such a way that it will LOAD or MERGE faster. Programs saved using SAVF will not VERIFY. Programs saved using SAVF can be LOADed by any BASIC. It will be found that the speed increase obtained will be greater the larger the program is.

1.3) EXIT

A new command EXIT will cause a return to the main system menu. The current program and data are lost.

1.4) EXT[ERNAL]

A new command EXTERNAL and functions EXTERNAL and EXTERNAL$ (EXT is a synonym for EXTERNAL) exist. This facility allows access to subroutines and functions that are in add-on ROM librarys. The keyword is always followed by the name of the routine. Any parameters should be enclosed in brackets and separated by commas. EXTERNAL routines are unusual in that the parameters may be used to return results. In this case the parameters must be variables. It is not possible to use a function with a string result parameter in a DEF statement.

Examples:

```
EXTERNAL shoot

EXT sound(x,y,z)

EXTERNAL complexadd (a(1),a(2),b(1),b(2))

PRINT EXT e, EXTERNAL parity(x), EXT$ spaces(20)

EXT exchange(a$,b$,c$)
```

The following new error numbers may be given:

58) Use of an EXTERNAL subroutine as a function.

59) Use of an EXTERNAL function as a subroutine.

60) The name of the requested EXTERNAL routine cannot be found in the attached library modules.

61) The EXTERNAL function being used delivers the wrong type of result.

62) Attempt to use a function with a string result parameter within a DEF statement.

64) The type of parameter is wrong.

2) Disc controller

When the disc controller on its own is connected, BASIC works exactly as it does in a NewBrain on its own. There are two additional commands, SAVF which is as described above, and CPM which immediately enters the 32K CP/M system, losing the current program and data.

When the disc controller is connected to a NewBrain with an expansion box no further extension than is described above occurs (the CPM command does not exist). To enter CP/M use EXIT and then enter CP/M from the paged system menu.

When the disc controller is connected to the system, with or without an expansion box, devices 12-15 become available (see DISCIO documentation). In addition commands like LOAD "PROGRAM" will use the disc system (device 12) to load the program rather than the cassette port. In this case the name of the program will not be displayed while loading and a filename must be given (simply saying LOAD will not do).

To load or save programs from cassette, device 1 has to opened explicitly. E.g. to load a program:

```
OPENIN#1,1,"PROGRAM"
LOAD#1
CLOSE#1
```

And to save a program:

```
OPENOUT#1,1,"PROGRAM"
SAVE#1:REM Could use SAVE here!
CLOSE#1
```

CSP 18/4/83

21-7-83 15

GRUNDY BUSINESS SYSTEMS LTD

SERVICE MANUAL

1. CONTENTS

MODIFICATIONS TO ISSUE 6 MAIN BOARD (NB 12117/*/6)

1. Relay 416 is type HE751A05-10.
2. Sil pads to insulate tracks from power transistor leads (8,22).
3. Resistor 121 is 330R.
4. Capacitor 50 is 33pF.
5. There is no capacitor 40 to be used.
6. Check sticky pad holds the crystal to the board.
7. Capacitor 104, -ve terminal to near IC417 pin 9, +ve terminal to near IC417 pin 10.
8. Resistor pack 134 4.7K add on IC421, resistor pack pins 2 to 9. connected to IC421 pins 2 to 9, resistor pack pin 1 connected to IC421 pin 20.
9. 330pF capacitor soldered to IC449 pins 4 and 8.
10. 470pF capacitor soldered to IC463 pins 3 and 7.
11. 330pF capacitor soldered to IC434 pins 7 and 11. (if vertical lines on graphics fade then change to 220pF).
12. Cut track leaving IC402 pin 5.
13. Cut track leaving IC402 pins 10 and 11.
14. Cut track joining IC413 pin 1 to IC413 pin 16.
15. Wire IC402 pin 5 to IC428 pin 4.
16. Wire IC402 pins 10 and 11 to IC428 pin 7.
17. Connect IC413 pin 1 to adjacent track (bend pin 1).
18. Replace resistor 25 with 1K2, connected from resistor 25 hole near diode 31, to capacitor 29 wire near Extention connector. Sleeve to avoid short circuits to other components.
19. Resistor 4.7K added to IC425 pin 10 and 14.

Read instructions on EPROMS/NEC ROMS in conjunction with this sheet.

ISSUE A

MODIFICATIONS TO ISSUE 7 MAIN BOARD (NB 12117/*/7)

1. Relay 416 is type HE751A05-10.
2. Silpads to insulate tracks from power transistor leads (8,22).
3. Resistor 121 is 330R.
4. Capacitor 50 is 33pF.
5. There is no capacitor 40 to be used.
6. Check sticky pad holds the crystal to the board.
7. Capacitor 104, -ve terminal to near IC417 pin 9, +ve terminal to near IC417 pin 10.
8. 330pF capacitor soldered to IC449 pins 4 and 8.
9. Replace resistor 25 with 1K2, connected from resistor 25 hole near diode 31, to capacitor 29 wire near Extention connector. Sleeve to avoid short circuits to other components.

Read instructions on EPROMS/NEC in conjunction with this sheet.

ISSUE A

RAM BOARD MODIFICATIONS (NB 12118/*/5)
(NB 12118/*/6)

1. 33R resistors are to be inserted in series with the tracks from pin 2 of the following IC's 409, 410, 411, 412, 413, 414, 415, 416. This is achieved by cutting the tracks from the pin 2 and soldering the resistor from pin 2 to the plated through hole of the cut track.

2. Solder a 100 pF capacitor from IC422 pin 7 to the plated through hole on the track connected to IC422 pin 13.

3. Solder a diode (1N914/4148 type) across resistor 48 (150R). Band mark to the left.

4. Add a Sil pad to the top of the board to insulate around the fixing hole between IC417 and IC410.

5. Link IC423 pin 3 to the track running between IC 423 pins 2 and 3.

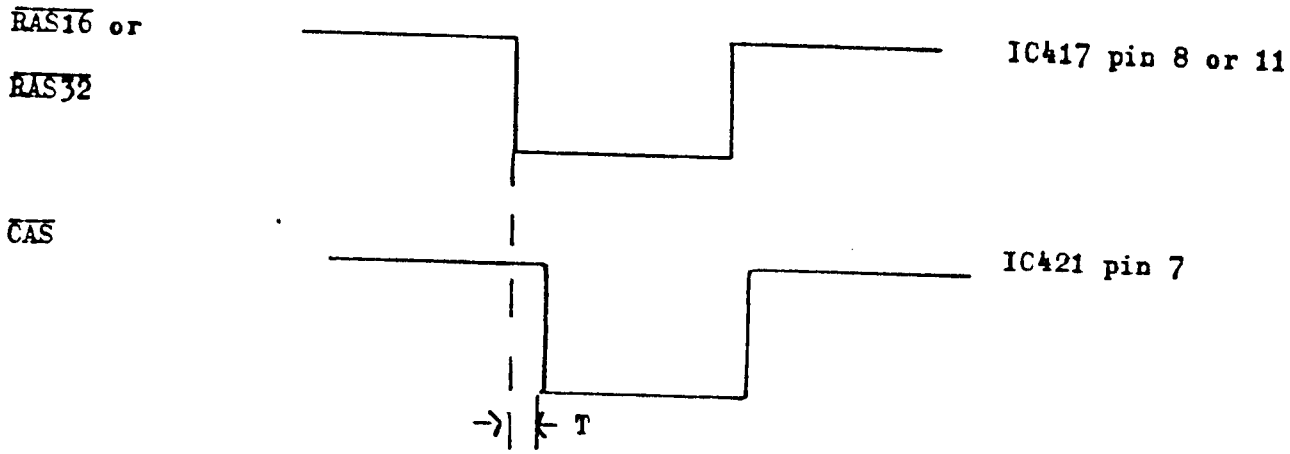
6. Cut track from IC 423 pins 3 and 4

7. Add resistor pack (4.7K commoned) on IC 421 on main board. Resistor pack pins 2 to 9 soldered to IC 421 pins 18 to 11 in sequence. Also solder resistor pack pw 1 to IC 421 pin 20.

Check $\overline{RAS}$, $\overline{CAS}$ timing

ISSUE A

RAM BOARD TIMING ($\overline{RAS}$,$\overline{CAS}$)

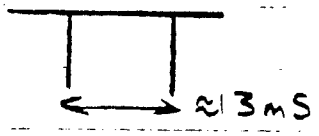
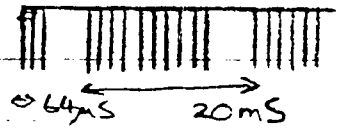
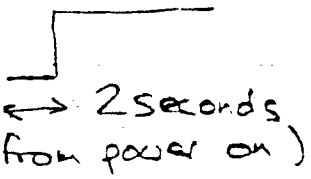


Capacitor * value is modified if nessesary to trim T (the $\overline{RAS}$ to $\overline{CAS}$ timing) to 80 - 140nS measured on the -ve going edges.

Oscilloscope beam 1 on IC417 pin 11 ($\overline{RAS16}$). Beam 2 on IC421 pin 7 ($\overline{CAS}$). Trigger from IC418 pin 8 (RFRSHB+CPRQ).

(If beam 1 does not display correct waveform try IC417 pin 8($\overline{RAS32}$)).

Issue	Capacitor *
5	49
6	49
8	57

IN	INPUT	FUNCTION	SIGNAL
6	CLK	Processor clock for Z8Ø	4MHz 
6	$\overline{INT}$	Interupt request Ø requests Z8Ø interupt action	
7	$\overline{NMI}$	Non maskable interupt Ø interupts Z8Ø (restart@ØØ66)	+5V
4	$\overline{WAIT}$	Wait Ø slows Z8Ø I/O action	+5V
5	$\overline{BUSREQ}$	Bus request Ø to request control of A,D buses	
6	$\overline{RESET}$	Reset Ø clears Z8Ø, restarts @ ØØØØ	

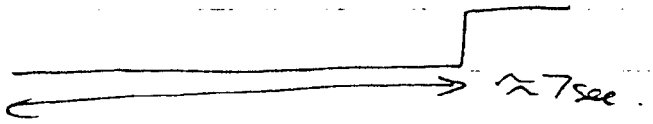
Fails to light Line display or give Monitor/TV signal

Check Power supply voltages.

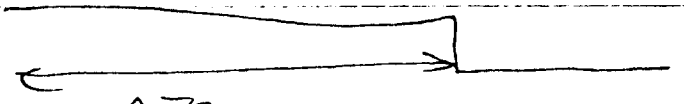
ZN404 = 2.45V.

IC 452 Power up resets.

Pin 6 

Pin 4 

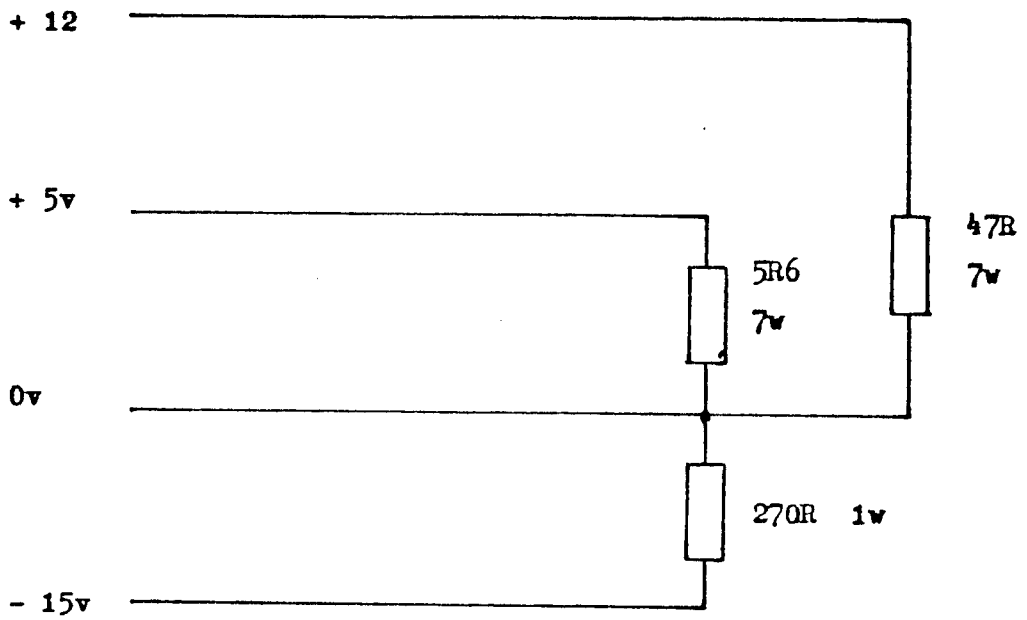
IC 432 Reset buffer.

Pin 4 

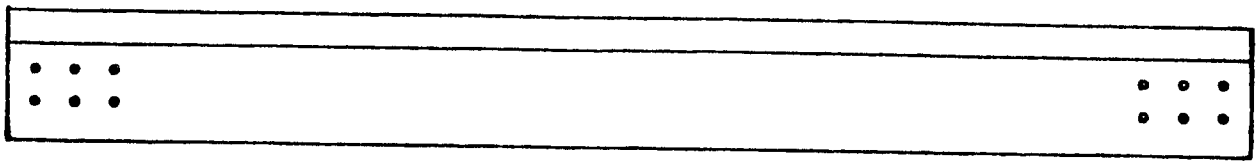
Clocks.

441	pin 11	16MHz
439	pin 4	8MHz
441	pin 8	8MHz
441	pin 6	8MHz
432	pin 8	4MHz

POWER SUPPLY TEST LOAD



EXPANSION CONNECTOR



49 47 45
50 48 46

5 3 1
6 4 2

1	GND		2	1/8 C	(4MHz)
3	A14	*	4	A13	*
5	D5	*	6	RMSL	
7	D4	*	8	D3	*
9	D6	*	10	D7	*
11	A11	*	12	A10	*
13	A8	*	14	A9	*
15	A12	*	16	A7	*
17	A3	*	18	A2	*
19	A1	*	20	A0	*
21	DØ	*	22	D1	*
23	Reserved		24	D2	*
25	$\overline{A5}$	*	26	A6	*
27	$\overline{RAMENB}$		28	EXRM2	
29	EXRM1		30	$\overline{EXRMØ}$	
31	ROMOV		32	$\overline{BUSRQ}$	*
33	$\overline{M1}$	*	34	$\overline{RESET}$	*
35	$\overline{RPSH}$	*	36	$\overline{WAIT}$	*
37	A4	*	38	$\overline{BUSAK}$	*
39	A15	*	40	$\overline{WR}$	*
41	$\overline{INT}$	*	42	$\overline{RD}$	*
43	$\overline{NMI}$	*	44	$\overline{HALT}$	*
45	$\overline{MREQ}$	*	46	$\overline{IORQ}$	*
47	PRTOV		48	RAMINH	
49	+5		50	BUSRQ	

* Connected to Z80

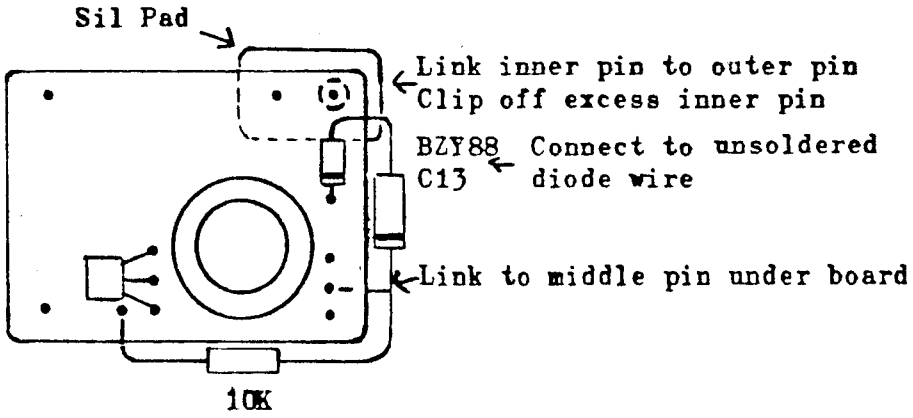
ISSUE A

ISSUE A

KEYBOARD MODIFICATIONS

1. Capacitor 4 (100μF) is replaced by a ZN404 (note polarity, + to +) A InF capacitor is soldered across the ZN404.

2. Replace resistor 8 (3K9) by a 680R resistor from resistor 8 + hole to capacitor 5 wire near the edge of the board.



3. The convertor (voltage) should be modified as in the above diagram. A Sil pad should be used on the main board at the top right corner of the convertor to insulate the wire link.

EPROM/ROM LINKS ISSUE 6 MAIN BOARD (NB 12117/*/6)

1. If using CD and EF EPROMS in IC407/408 positions use a wire link from 5V to the hole near IC408 pin 28. (If the board previously held an NEC ROM then remove the link IC408 pin 26 to IC428 pin 4 and repair track IC428 pin 4 to resistor pack 54 pin 4).

   If using NEC ROM in IC408 (No CD in IC407) use a 4K7 resistor in place of the wire link. Then wire the end of the resistor connected to IC408 pin 26 to IC428 pin 4. cut track IC428 pin 4 to resistor pack 54 pin 4.

2. If using character EPROM in IC453 then use two long links (horizontal) near IC453 pin 28.

   If using NEC ROM in IC453 then use two short links (vertical) near IC453 pin 28.

EPROM/ROM LINKS ISSUE 7 MAIN BOARD (NB 12117/*/7)

1. If using NEC ROM in IC408 (No CD in IC407) use a 4K7 resistor from 5V to the hole near IC408 pin 28. Wire from resistor pack 54 pin 4 to the end of the 4K7 resistor connected to the ROM IC408 pin 26 (near pin 28). Remove link next to resistor pack 54.

   If using CD and EF EPROMS in IC407/408 positions use a wire link from 5V to the hole near IC408 pin 28 (There is no wire on the back of the board as is used with NEC ROM). Replace link next to resistor pack 54.

2. If using character EPROM in IC453 then use two short links (vertical) near IC453 pin 28.

   If using NEC ROM in IC453 then use two long links (horizontal) near IC453 pin 28.

3. Connect Pin 14 of IC408 (EF) to the adjacent ground track.

RAM BOARD MODIFICATIONS (NB 12118/*/8)

1. Solder a 100pF capacitor from IC422 pin 7 to the plated through hole on the track connected to IC422 pin 13.

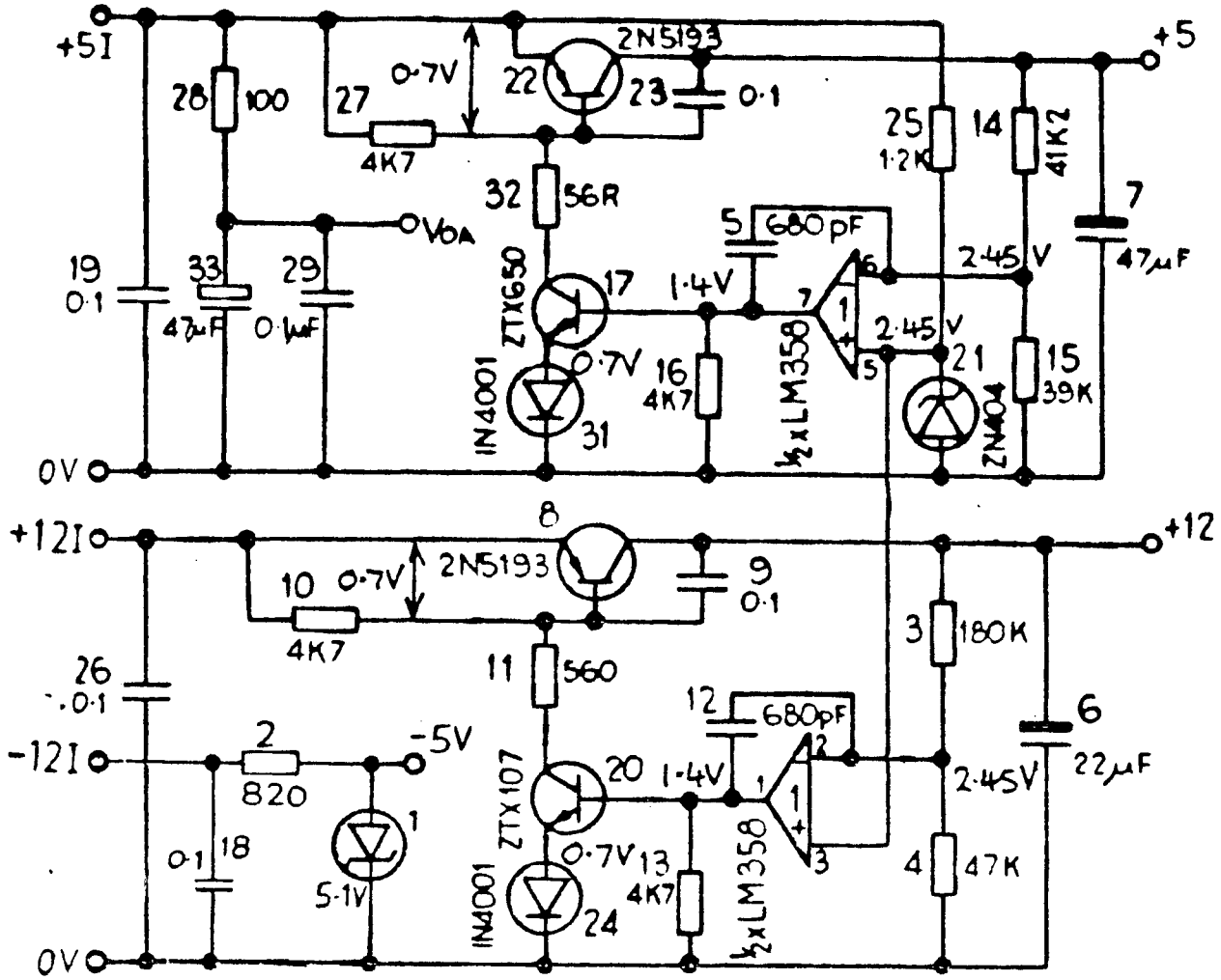
2. Solder a diode (1N914/4148 type) across resistor 56 (150R). Band mark to the left.

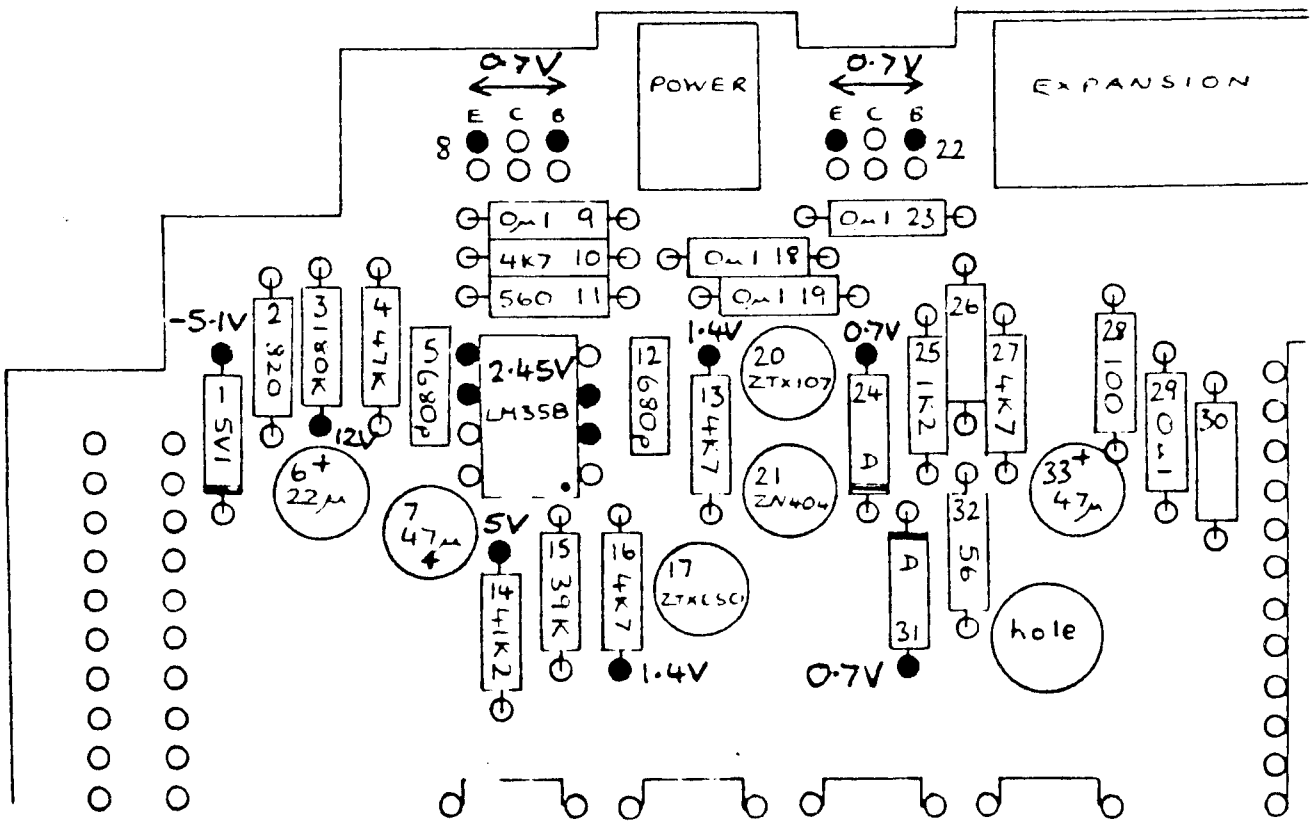
3. There should not be a resistor pack (4K7) Soldered to IC421 pins 11 to 18 and 20 on the main board.

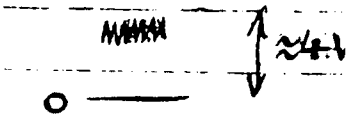
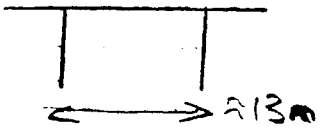
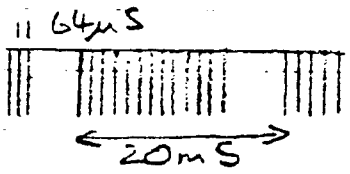
4. Solder a resistor (470R) from IC401 pin 6 to IC401 pin 9.

Check $\overline{RAS}$, $\overline{CAS}$ timing

ISSUE A


+5I
28 100
27 0·7V
4K7
22 2N5193
23 0·1
+5
25 1·2K
14 41K2
7 47µF
32 56R
VOA
5 680pF
19 0·1
33 47µF
29 0·1µF
17 1·4V
ZTX650
IN4001
0·7V
31
16 4K7
½xLM358
2·45V
21 ZN404
15 39K
0V
+12I
8
10 0·7V
4K7
2N5193
9 0·1
+12
3 180K
26 0·1
11 560
12 680pF
6 22µF
-12I
2 820
-5V
1
5·1V
0·1 18
20 1·4V
ZTX107
IN4001
0·7V
24
13 4K7
½xLM358
2·45V
4 47K
0V


0·7V
POWER
0·7V
EXPANSION
E C B
8
22
0m1 9
4K7 10
560 11
0m1 23
0m1 18
0m1 19
-5·1V
2 820
3 180K
4 47K
5 680pF
2·45V
LM358
12 680pF
1·4V
20 ZTX107
13 4K7
0·7V
24
21 ZN404
25 1·2K
26
27 4K7
28 100
29 0·1
30
33 47µ
32 56
6 22µ
7 47µ
5V
15 39K
16 4K7
17 ZTX650
1·4V
4 41K2
A
31
0·7V
hole

PIN	OUTPUT	FUNCTION	SIGNAL
18	$\overline{HALT}$	Halt state Ø indicates that Z8Ø is in halt state awaiting $\overline{INT}$ or $\overline{NMI}$ = Ø	
19	$\overline{MREQ}$	Memory request Ø indicates that address is valid	pulses
20	$\overline{IORQ}$	Input/Output Request Ø indicates valid I/O address	
21	$\overline{RD}$	Memory read Ø indicates Z8Ø requests input data	pulses
22	$\overline{WR}$	Memory write Ø indicates data bus holds valid data	pulses
23	$\overline{BUSAK}$	Bus acknowledge 1 indicates Z8Ø has control of buses	
27	$\overline{M1}$	Machine cycle one Ø indicates opcode fetch	pulses
28	$\overline{RFSH}$	Refresh Ø indicates the refresh address is valid	pulses

ROM BOX INSTRUCTIONS

FOUR WAY STANDARD CONNECTIONS. (Fig 3.0)

This unit is capable of placing four 2764 EPROMS anywhere within the 64K address range of the Z80. Chip 404 (74LS138) divides the top three address lines into individual chip select lines each of 8K bytes width. Links L4 through to L7 connect to the chip select on the four sockets. With straight linking on the board

EPROM socket 7 will be selected in the address range E000 HEX to FFFF HEX.

EPROM socket 6 will be selected in the address range C000 HEX to OFFF HEX.

EPROM socket 5 will be selected in the address range A000 HEX to BFFF HEX.

EPROM socket 4 will be selected in the address range 8000 HEX to 9FFF HEX.

To achieve any other 8K byte address range it will be necessary to either strap across the links or (if address below 8000H is wanted) link directly to the LS138 chip. (See Fig 1.0)

Another feature of this board is the ability to 'power up' into an EPROM on this board. Normally the machine 'powers up' at E000H internally, but by linking L1 and L2 the power up condition is lengthened and moved to a device in the ROM BOX. Links L13 to L15 and G13 to G15 determine which device is selected in power up. Strapping L13 to L15 will make the machine power up at EF. For other strapping option see Fig 2.0

SOCKET	E/F	C/D	A/B	8/9	0/7	4/5	2/3	0/1
7	L7-A7	L7-A6	L7-A5	L7-A4	L7-A3	L7-A2	L7-A1	Not Recommended
6	L6-A7	L6-A6	L6-A5	L6-A4	L6-A3	L6-A2	L6-A1	
5	L5-A7	L5-A6	L5-A5	L5-A4	L5-A3	L5-A2	L5-A1	
4	L4-A7	L4-A6	L4-A5	L4-A4	L4-A3	L4-A2	L4-A1	

Fig 1.0

* A3-A1 are pins on chip 404

ADDR	STRAP				
E/F	L1	L2	L13	L14	L15
C/D	L1	L2	G13	L14	L15
A/B	L1	L2	L13	G14	L15
8/9	L1	L2	G13	G14	L15
6/7	L1	L2	L13	L14	G15
4/5	L1	L2	G13	L14	G15
2/3	L1	L2	L13	G14	G15

Fig 2.0.

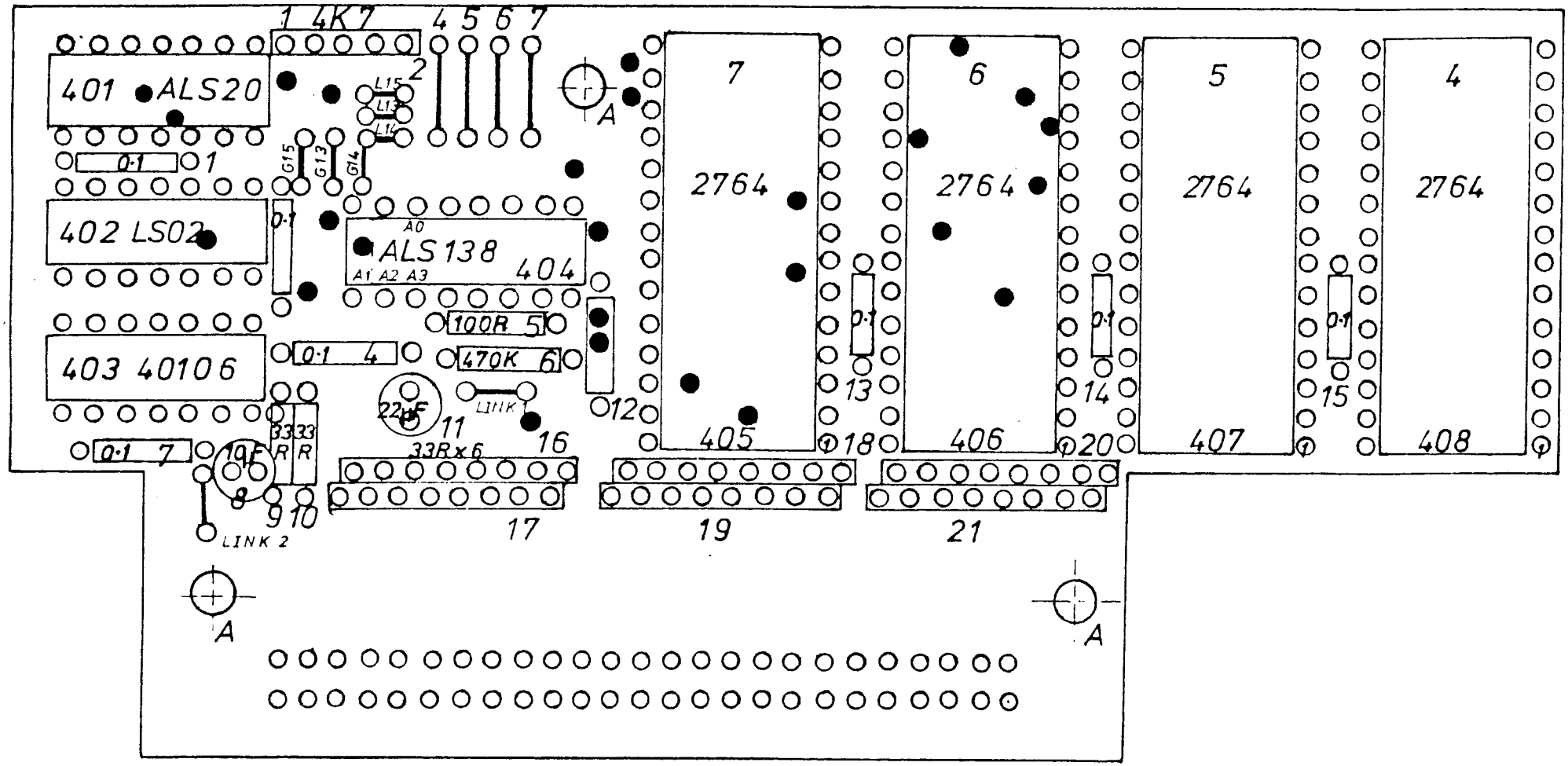
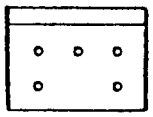

1 4K7
4 5 6 7
401 ALS20
2
L15
L13
L14
A
7
2764
6
2764
5
2764
4
2764
0·1
1
G15
G13
G14
402 LS02
0·1
A0
ALS 138
A1 A2 A3
404
100R 5
403 40106
0·1 4
470K 6
22µF
11
LINK
12
13
14
15
0·1
0·1
0·1
0·1
0·1 7
10µF
8
3333
R
3333
R
9 10
33R x 6
16
LINK 2
17
405
18
406
19
20
407
21
408
A
A

Fig 3.0

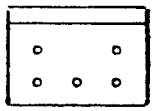
CONNECTORS

POWER



5	3	1	0v	-15v	n/c
6		2	+5v		+12v

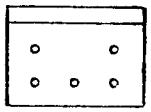
PRINTER



5		1	n/c		$\overline{\text{n/c}}$
6	4	2	0v	PO	$\overline{\text{CTSP}}$

$\overline{\text{CTSP}}$: Not clear to send printer date (input)
PO : Printer data (output)

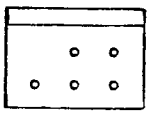
MODEM (COMMS)



5		1	$\overline{\text{RTSD}}$		$\overline{\text{RDDK}}$
6	4	2	0v	DO	$\overline{\text{CTSD}}$

RDDK : Read data (input)
$\overline{\text{CTSD}}$: Not clear to send data (input)
DO : Data out (output)
$\overline{\text{RTSD}}$: Not ready to send data (output)

TAPE (1&2)



	3	1			IFT
6	4	2	0v		OTT

IFT : Input from tape
OTT : Output to tape
—/— : Control relay

Interfaces. Signal Description and Connector Pinout.

PIN	SIGNAL TYPE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
Power			
1	O		Ground for supply flex screen.
2	I	+12I	Source for 12V internal rail. Accepts 13.0 to 14.5V applied. Consumes about 110MA.
3	I	-12I	Source for -12V and-5V internal rails. Accepts -14.0 to -12V applied. Consumes about 25mA.
4			Absent (Accepts polarising blank in power lead plug).
5	I	GND	Ground return for all internal rails.
6	I	+5I	Source for 5V internal rail. Accepts 6.5V to 7.5V applied. Consumes about 900MA.
Tape 1 and 2			
1	I	IFT1	Input from tape recorder earphone jack. Loaded by 10nF in series with 39K. Input voltage >500mV required
2	O	OTT1	Output to tape recorder microphone jack. 25mV nominal. Slew rate 4msec/V. 250 ohms source impedance.
3		TMS1	Relay contact for tape motor. Open circuit DC voltage must be < 20V.
4		$\overline{\text{TMS1}}$	Relay contact for tape motor. Open circuit DC voltage must be < 20V.
5		TDTPLPO	Absent (Accepts polarising blank in tape lead plug).
6		GRND	Ground return for tape signals via flex screen to microphone jack ground only.
Communications			
1	I	SRDDK	Received serial data. Input > 2.5V (binary zero) or < 2.5V (binary one).
2	I	$\overline{\text{SCTSD}}$	Nominally not clear to send ($\overline{\text{CTS}}$ or $\overline{\text{RFS}}$). Used as handshake by software. Characteristics as for pin 1.
3			Absent (Accepts polarising blank in communications lead plug).
4	O	SDO	Transmitted serial data. Output > 8V (binary zero) or < -8V (binary one) for 3K ohms load. Power up leaves binary zero (contra V24 spec) until stream opened to device.
5	O	$\overline{\text{SRTSD}}$	Nominally not request to send ($\overline{\text{RTS}}$). Used as handshake by software. Characteristics as for pin 4 (No power up caveat).

PIN	SIGNAL TYPE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
9	T	D6	D6 from Z80A.
10	T	D7	D7 from Z80A.
11	T	A11	A11 from Z80A.
12	T	A10	A10 from Z80A.
13	T	A8	A8 from Z80A.
14	T	A9	A9 from Z80A.
15	T	A12	A12 from Z80A.
16	T	A7	A7 from Z80A.
17	T	A3	A3 from Z80A.
18	T	A2	A2 from Z80A.
19	T	A1	A1 from Z80A.
20	T	A0	A0 from Z80A.
21	T	D0	D0 from Z80A.
22	T	D1	D1 from Z80A.
23			Unused. Early machines have a signal connected to this pin. Connect 10K ohm pull-up to 5V.*(HP11).
24	T	D2	D2 from Z80A.
25	T	A5	A5 from Z80A.
26	T	A6	A6 from Z80A.
27	I	$\overline{\text{RAMENB}}$	Not RAM enable. Logic zero routes Z80 memory request to RAM, conditional on pin 48 signal. Unconnected state biased to logic one. *(HP3).
28	I	EXRM2	External A15 signal. Used by paging circuits in the expansion interface module *(HP2).
29	I	EXRM1	External A14 signal. Used as under pin 28 *(HP1).
30	I	EXRM0	External A13 signal. Used as under pin 28 *(HP0).

NEWBRAIN, INPUT/OUTPUT, PORT MAP

NOTE:

1. The NewBrain without expansion interface module decodes port address lines A0 - A4 inclusive. Reference to port 'nx' where x (integral) lies between 0 and31 inclusive produces the same result for all integral values of n from 1 to 8 inclusive.

2. The addition of the expansion interface module causes decoding of port address lines A0 - A7 inclusive. There is then no 'wrap-around'.

3. Grundy Business Systems will make every effort not to use ports from 32 to 63 inclusive.

4. Some ports are further decoded by the address lines A8 - A15. This is indicated in the relevant port description below.

Abbreviations:

P	Processor modules A or AD
EI	Expansion Interface module
DC	Disc Controller module
NC	Network Controller module
API	Autonomous Peripheral Interface module
ME	Memory Expansion module
R	Read
W	Write
D0 - D7	Data bits 0 to 7
A0 - A15	Address bit 0 to 15

Port No.	Read/ Write Status	Module	Description
0	R/W	EI	Not used.
1	W	EI	Load enable register 2. R reads and loads all ones. D0: zero enables user data bus interrupt and also parallel latched data output (or centronics printer) interrupt. D1: one enables ADC conversion complete interrupt and also calling indicator interrupt. D2: one enables serial receive clock into multiplier input of DAC and signals data terminal not ready. D3: one enables 50K Baud serial data rate to be obtained ie. CTC input clock of 800KHz. Zero selects 307.6923KHz. D4: one enables serial receive clock to sound output summer, and also selects serial input from the printer port. Zero selects serial input from the comms port. D5: one enables second bank of four analogue inputs (voltage, non-ratiometric), ie. ch4-7,and enables sound output, zero selects ch0-3. D6: one enables serial transmit clock to sound output summer, and also selects serial output to the printer port. Zero selects serial output to the comms port. D7: ninth output bit for centronics printer port.
2	W	EI	Load page registers. R reads all ones, but has no other effect. Which of sixteen 8 bit registers is written to is selected by address bits A12, A15, A14, A13, (most to least significant). Contents written are given by $\overline{A11}$, $\overline{D6}$, $\overline{D5}$, $\overline{D4}$, $\overline{D3}$, $\overline{D2}$, $\overline{D1}$, $\overline{D0}$.
3	W	EI	Load latched parallel output register (centronics printer port data output). R reads and loads all ones.

Port No.	Read/Write Status	Module	Description
18	R/W	EI	As 16 for ANIN2 if ch0 - 3 selected or ANINV unamplified if ch4 - 7 selected.
19	R/W	EI	As 16 for ANIN3 if ch0 - 3 selected or ANINV amplified if ch4 - 7 selected.
20	R	P & EI	Read status register 1.W causes bus contention. D0: fixed at one - indicates excess of 24 or 48. Obsolete (P). D1: one indicates power up from 'cold' - necessary in battery machines with power switching (P). D2: one indicates analogue or calling indicator interrupts. D3: zero indicates centronics printer (latched output data) port interrupt. D4: zero indicates parallel data bus port interrupt. D5: zero indicates frame frequency clock interrupt. D6: zero indicates ACIA interrupt. D7: zero indicates interrupt from micro-controller COP420M.
21	R	EI	Read status register 2. W causes bus contention. If EI not connected appears identical with port address 20. D2: one selects normal video on power up (white on black), zero selects reversed video (appears as D0 on the first two hundred EI's). D3: one indicates power is being taken from the mains supply. D4: one indicates that 40 column video is selected on power up. Zero selects 80 column video. D6: one indicates that a video display is required on power up.
22	R	D	Read data input register 1 (user input register 1). W causes bus contention. D0: received serial data from communications port. D1: zero indicates 'clear-to-send' condition at communications port. D5: logic level tape input. D7: zero indicates 'clear-to-send' condition at printer port.
23	R/W	P & EI	Read or write to user data bus. If EI not connected appears identical with port address 21 and W will cause bus contention.
24	R/W	EI	Read status and load control registers of the ACIA.
25	R/W	EI	Read the receive data register and load the transmit data register of the ACIA.
26		EI	Unused.
27		EI	Unused.
28	R/W	EI	Communicate with channel 0 of the CTC.
29	R/W	EI	Communicate with channel 1 of the CTC.
30	R/W	EI	Communicate with channel 2 of the CTC.

Interfaces. Signal Description and Connector Pinout

Connector	Pin No	Signal Type	Circuit Mnemonic	Description
Power				
	1	O		Ground for supply flex screen.
	2	I	+12I	Source for 12V internal rail. Accepts 13.0 to 14.5V applied Consumes about
	3	I	-12I	Source for -12 and -5V internal rails. Accepts -14.0 to -12.0V applied. Consumes about
	4			Absent (Accepts polarising blank in power lead plug).
	5	I	GRND	Ground return for all internal rails.
	6	I	+5I	Source for 5V internal rail. Accepts 6.5V to 7.5V applied. Consumes about
Communications 1				
	1	I	SRDV24	Received serial data. Input >2.5V (binary zero) or <-2.5V (binary one). Load impedance about 4K. Unloaded input is binary 0.
	2	I	$\overline{\text{SCTSV24}}$	Not clear to send ($\overline{\text{CTS}}$ or $\overline{\text{RFS}}$). Characteristics as for pin 1.
	3			Absent. (Accepts polarising blank in communications or printer lead plug).
	4	O	STDV24	Transmitted serial data. Output >8V (binary zero) of <-8V (binary one) for 3K load.
	5	O	$\overline{\text{SRTSV24}}$	Not request to send ($\overline{\text{RTS}}$). Characteristics as for pin 4.
	6			Ground return for communications signals via flex screen.
Communications 2				
	1	I	SRDPTR	Received serial data or calling indicator. Input >2.5V (binary zero) or <-2.5V (binary one). Load impedance about 4K. Unloaded input is binary 0.
	2	I	$\overline{\text{SCTSPTR}}$	Not clear to send ($\overline{\text{CTS}}$ of $\overline{\text{RFS}}$). Characteristics as for pin 1.
	3			Absent (Accepts polarising blank in communications or printer lead plug).

Connector	Pin No.	Signal Type	Circuit Mnemonic	Description
	5	T	DB4	Data bit 4. LSTTL level with 270 ohms series register.
	6	T	DB5	Data bit 5. LSTTL level with 270 ohms series register.
	7	T	DB6	Data bit 6. LSTTL level with 270 ohms series register.
	8	T	DB7	Data bit 7. LSTTL level with 270 ohms series register.
	9	O	$\overline{\text{USBS}}$	Negative strobe pulse about 625n sec enabling input or output. LSTTL output level.
	10	I	$\overline{\text{USRINT}}$	Negative input causes Z80A interrupt if enable bit is set. LSTTL level input.
	11	O	$\overline{\text{RDUSR}}$	Negative strobe pulse roughly synchronous with that on pin 9 indicating interface is inputting data to the system. LSTTL level output.
	12		GRND	Ground return for signals via flex screen.
Analogue				
	1	I	ANIN0	High impedance unprotected input to channel 0 of the multiplexed ADC. Normally used for ratiometric inputs only.
	2	I	ANIN1	As for pin 1 but connected to channel 1.
	3	I	ANIN2	As for pin 1 but connected to channel 2.
	4	I	ANIN3	As for pin 1 but connected to channel 3.
	5	I	ANINV	High impedance analogue input protected by 4K7 series resistor and catch diodes to ±5V. Input may be from 0 to 5V or -2.5V to +2.5V. Amplification up to 40 dB is available for lower level signals.
	6	O	ANOF	Buffered analogue output from DAC. Output range is either 0 to 5V or -2.5V to +2.5V.
	7	O	ANOV	Buffered analogue output from DAC with variable attenuation. Output range down to 0 to 50mV or -25mV to +25mV.
	8	O	SND	Buffered 'sound' output from a programmable mix of two variable frequency sources and the DAC output.

Connector	Pin No	Signal Type	Circuit Mnemonic	Description
	15	T	HA12	Buffered A12 address line, as pin 3.
	16	T	HA7	Buffered A7 address line, as pin 3.
	17	T	HA3	Buffered A3 address line, as pin 3.
	18	T	HA2	Buffered A2 address line, as pin 3.
	19	T	HA1	Buffered A1 address line, as pin 3.
	20	T	HA0	Buffered A0 address line, as pin 3.
	21	T	HD0	Buffered D0 data line, as pin 5.
	22	T	HD1	Buffered D1 data line, as pin 5.
	23	T	HP11	Buffered P11 page address line.
	24	T	HD2	Buffered D2 data line, as pin 5.
	25	T	HA5	Buffered A5 address line, as pin 3.
	26	T	HA6	Buffered A6 address line, as pin 3.
	27	T	HP3	Buffered P3 page address line.
	28	T	HP2	Buffered P2 page address line.
	29	T	HP0	Buffered P1 page address line.
	30	T	HPO	Buffered P0 page address line.
	31	I	$\overline{GRQ}$	External page address request. Tristates page address drives in expansion interface module.*
	32	OC	$\overline{HBUSRQ}$	Not bus request. Used in autonomous data interchanges via the paged expansion highway. To Z80A.*
	33	O	$\overline{HMI}$	Not M1 from Z80A.*

Connector

Pin No	Signal Type	Circuit Mnenomic	
34	O	$\overline{\text{RST}}$	Not reset as supplied to Z80A and other circuits.
35	O	$\overline{\text{HRFRSH}}$	Not refresh from Z80A.
36	I	$\overline{\text{HWAIT}}$	Not wait. Memory type dependent. To Z80A.
37	T	HA4	Buffered A4 address line, as pin 3.
38	O	$\overline{\text{HBUSAK}}$	Not bus acknowledge (external). Used in conjunction with signal on pin 32. From Z80A.*
39	T	HA15	Buffered A15 address line, as pin 3.
40	T	$\overline{\text{HWR}}$	Buffered not write signal. From Z80A. Direction reversed by HBUSAK.
41	OC	$\overline{\text{HINT}}$	Not interrupt.*
42	T	$\overline{\text{HRD}}$	Buffered not read signal. As pin 40.
43	O	$\overline{\text{HISLT}}$	Not highway isolate. Used in creating virtual page map and in multiprocessing mode.
44	I	MPSL	Select multi processing. Used to expand page addressing from 8 to 12 bits.*
45	T	$\overline{\text{HMREQ}}$	Not memory request. Strobe for activating memory circuits. As pin 40.
46	T	$\overline{\text{HIORQ}}$	Not input/output request. Strobe for activating input or output ports. As pin 40.
47	T	HP6	Buffered P6 page address line.
48	T	HP5	Buffered P5 page address line.
49	O	5V	Unprotected 5V rail.*
50	T	HP4	Buffered P4 page address line.

Abbreviations:

D0 - D7	Data bits 0 to 7
A0 - A15	Address bits 0 to 15
I	Input
O	Output
T	Tristate
OC	Open Collector
*	Signal function changes or is modified on multi processing expansion highway.

PIN	SIGNAL TYRE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
6			Ground return for communications signals via flex screen.
Printer			
1			Unused.
2	I	$\overline{\text{SCTSP}}$	Not clear to send ($\overline{\text{CTS}}$ or $\overline{\text{RFS}}$). Characteristics as for pin 1, communications interface
3			Absent (Accepts polarising blank in printer lead plug).
4	O	SPO	Transmitted serial data. Characteristics as for pin 4, communications interface.
5			Unused.
6			Ground return for printer signals via flex screen.
Monitor			
	O		Composite video to CCITT specification. Sourced at 75 ohm. Open circuit peak white voltage 2V, Black 1V, sync 0V. Frame 50Hz, line 15.625KHz, H sync 4µsec, V sync. 140 µsecs. Video blanking 4µsecs before and after H sync, and 8 µsecs before, 840 µsecs after V sync. Programmable video starts 11.5 µsec after and finishes 6.4 µsec before H sync and starts 2.4 m sec after, finishes 1.6 m sec before V sync.
UHF			
	O		UHF output modulated by video. Carrier 591.25 MHz (Ch 36), output 0- 1.5mV sourced at 75 ohms (RF). Output is halved when the monitor output is terminated by 75 ohms.
Expansion			
1		GRND	Ground. Signal ground connecting to screen on ribbon cable from other modules.
2	O	1/8C	4MHz clock. Z80A clock is formed by inversion of this signal.
3	T	A14	A14 from Z80A.
4	T	A13	A13 from Z80A.
5	T	D5	D5 from Z80A.
6		RMSL	Address modifier. Unused signal.*(HA 16).
7	T	D4	D4 from Z80A.
8	T	D3	D3 from Z80A.

PIN	SIGNAL TYPE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
31	I	$\overline{\text{ROMOV}}$	Not address overide. Logic zero causes A13, A14, A15, to be replaced by external A13,A14,A15.*(GRQ).
32	OC	$\overline{\text{BUSRQ}}$	Not Bus request as supplied to Z80A ($\overline{\text{BUSRQ}}$).
33	O	$\overline{\text{M1}}$	Not M1 from Z80A ($\overline{\text{M1}}$)
34	O	$\overline{\text{RST}}$	Not reset as supplied to Z80A ($\overline{\text{RST}}$).
35	O	$\overline{\text{RFRSH}}$	Not refresh from Z80A ($\overline{\text{RFRSH}}$)
36	I	$\overline{\text{WAIT}}$	Not wait as supplied to Z80A ($\overline{\text{WAIT}}$).
37	T	A4	A4 from Z80A.
38	O	$\overline{\text{BUSAK}}$	Not bus acknowledge from Z80A ($\overline{\text{BUSAK}}$).
39	T	A15	A15 from Z80A.
40	T	$\overline{\text{WR}}$	Not write from Z80A ($\overline{\text{WR}}$)
41	OC	$\overline{\text{INT}}$	Not interrupt as supplied to Z80A ($\overline{\text{INT}}$).
42	T	$\overline{\text{RD}}$	Not read from Z80A ($\overline{\text{RD}}$)
43	I	$\overline{\text{NMI}}$	Not non-maskable interrupt as supplied to Z80A ($\overline{\text{NMI}}$) *($\overline{\text{HISLT}}$).
44	O	$\overline{\text{HALT}}$	Not halt from Z80A ($\overline{\text{HALT}}$) *(MPSL).
45	T	$\overline{\text{MREQ}}$	Not memory request from Z80A ($\overline{\text{MREQ}}$).
46	T	$\overline{\text{IORQ}}$	Not input/output request from Z80A ($\overline{\text{IORQ}}$).
47	I	PRTOV	Logic one inhibits internal IO port decode. Unconnected state biased to logic zero *(HP6).
48	I	RAMINH	Logic one inhibits Z80A request to internal RAM. Unconnected state biased to logic zero. *(HP5)
49	O		± 5volts (up to 200mA).
50	O	BUSRQ	Bus request as generated by video circuit *(HP4)

Abbreviations:

D0-D7	Data bits 0 to 7	T	Tristate
A0-A15	Address bits 0 to 15	OC	Open collector
I	Input	*	Signal function changes on paged expansion highway.
O	Output		

Port No.	Read/Write Staus	Module	Description
4	W	P	Reset clock interrupt. R likewise and reads ones. Data irrelevant.
5	W	EI	Load DAC. R reads and loads all ones. On the first 200 modules the data bits are reversed. i.e. D7 is the least significant bit and D0 the most significant bit.
6	R/W	P	Communicate with COP 420M micro-controller.
7	W	P	Load enable register 1. R reads and loads all ones. D0: zero enables frame frequency clock interrupts. D2: one enables video display. D4: zero asserts 'request to send' on communications port. D5: one asserts binary one data on communications port. D7: one asserts binary one data on printer port. D1, D3 and D6 are not currently used.
8	W	P	Sets ninth bit of video address counter. Data is loaded into first 8 bits of video address counter. Likewise R reads and loads all ones.
9	W	P	Loads first 8 bits of video address counter. R loads and reads all ones.
10	W	P	As 8
11	W	P	As 9
12	W	P	Loads video control register. R loads and reads all ones. D0: one reverses video over entire field, ie. black on white. D1: zero generates 128 characters and 128 reverse field characters from 8 bit character code. One generates 256 characters from 8 bit character code. D2: zero generates 320 or 640 horizontal dots in pixel graphics mode. One generates 256 or 512 horizontal dots in pixel graphics mode. D3: zero selects 256 characters expressed in an 8 x 10 matrix, and 25 lines (max) displayed. One selects 256 characters in an 8 x 8 matrix, and 31 lines (max) displayed. D6: zero selects 40 character line length. One selects 80 character line length D4, D5, D7 are currently unused.
13	W	P	As 12
14	W	P	As 12
15	W	P	As 12
16	R/W	EI	W starts conversion of channel 0 or 4 (same ratiometric input ANINO). R reads conversion result of ch0 or 4 (ANINO).
17	R/W	EI	As 16 for ANIN1 (ch1 or 5).

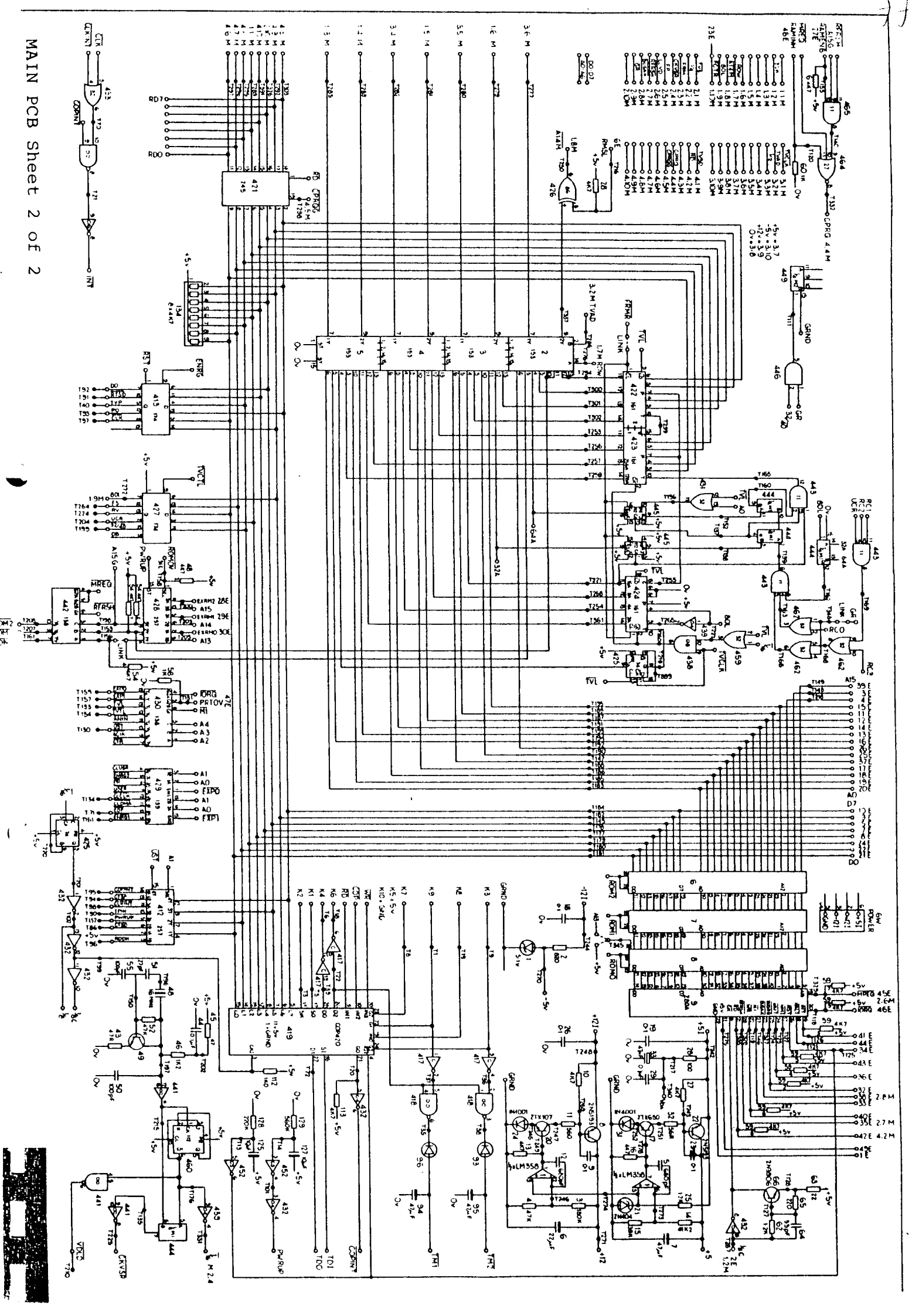
Port No.	Read/Write Status	Module	Description
31	R/W	EI	Communicate with channel 3 of the CTC.
255	W	EI/DC/ NC	Load paging status register. R reads all ones. Modules each contain a paging status register. EI is selected when A8 is one, DC when A9 is one and NC when A10 is one.

D0: one enables paging circuits.
D1: unused.
D2: one sets local A16 to one (ie. causes second set of 8 page registers to select addressed memory).
D3: zero selects multi-processing mode. Among other effects this extends the page registers from 8 to to 12 bits in length.
D4: one isolates the local machine. This is used in multi-processing mode.

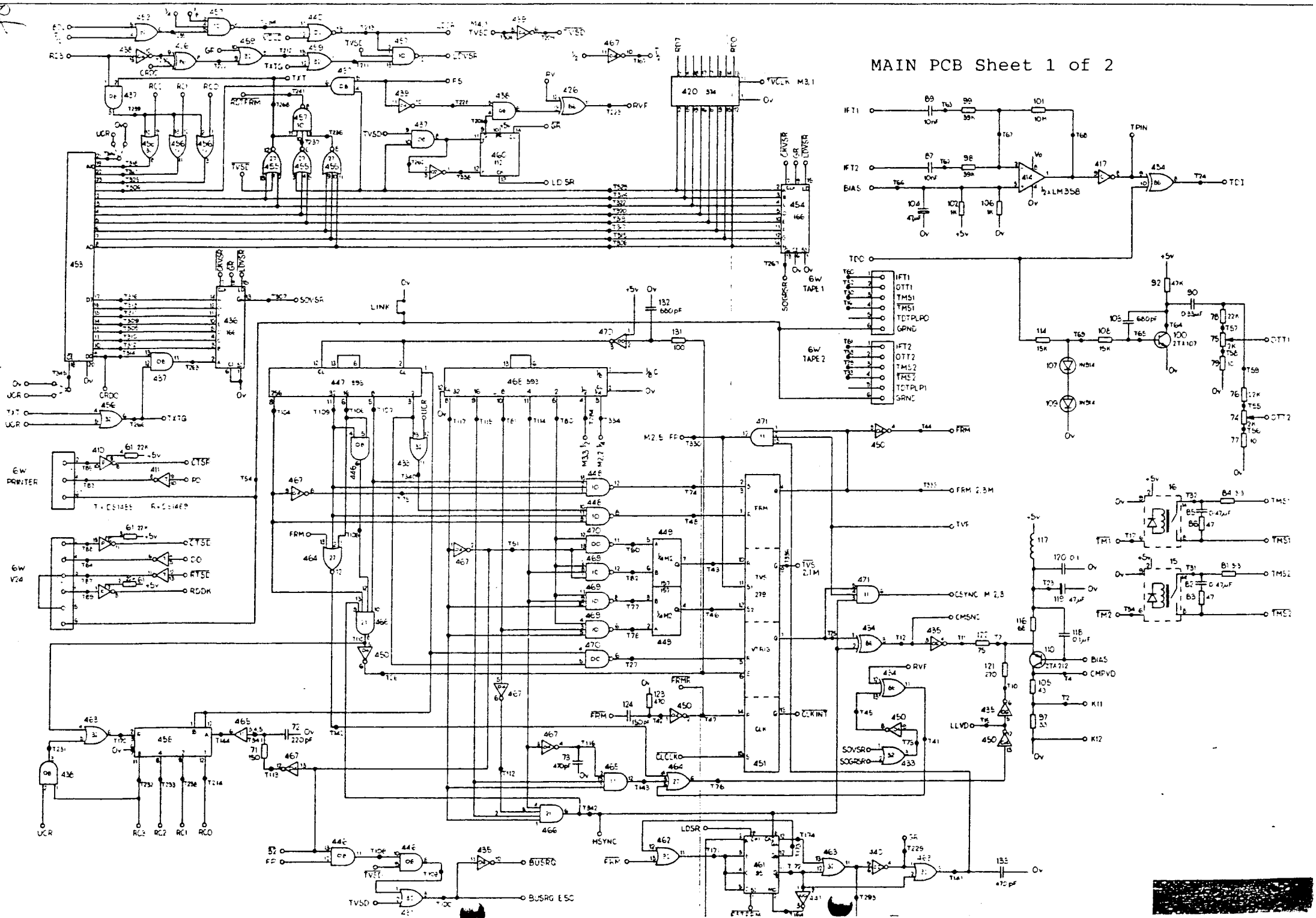
Connector	Pin No.	Signal Type	Circuit Mnemonic	Description
	4	O	STDPTR	Transmitted serial data. Output >8V (binary zero) or <-8V (binary one) for 3K load.
	5	O	$\overline{\text{SRTSPTR}}$	Not request to send ($\overline{\text{RTS}}$) or not data terminal ready ($\overline{\text{DTR}}$). Characteristics as for pin 4.
	6			Ground return for communications signals via flex screen.
Parallel Printer				
	1	O	PD0	Data bit 0, LSTTL output level.
	2	O	PD1	Data bit 1, LSTTL output level.
	3	O	PD2	Data bit 2, LSTTL output level.
	4	O	PD3	Data bit 3, LSTTL output level.
	5	O	PD4	Data bit 4, LSTTL output level.
	6	O	PD5	Data bit 5, LSTTL output level.
	7	O	PD6	Data bit 6, LSTTL output level.
	8	O	PD7	Data bit 7, LSTTL output level.
	9	O	$\overline{\text{USER}}$	Negative clock pulse of about 625nsec loading output register on the rising edge. LSTTL output level.
	10	I	$\overline{\text{PINT}}$	Negative input pulse sets internal interrupt latch on falling edge. Pulse should not be less than 20n sec. Latch is cleared internally. LSTTL input level.
	11	O	PD8	Data bit 8. LSTTL output level.
	12		GRND	Ground return for signals via flex screen.
Data Bus				
	1	T	DB0	Data bit 0. LSTTL level with 270 ohms series register.
	2	T	DB1	Data bit 1. LSTTL level with 270 ohms series register.
	3	T	DB2	Data bit 2. LSTTL level with 270 ohms series register.
	4	T	DB3	Data bit 3. LSTTL level with 270 ohms series register.

Connector	Pin No.	Signal Type	Circuit Mnemonic	Description
	9	O	REF2	5V output reference for use with ratiometric inputs and DC shift of input and output ranges.
	10	I	ANIRI	Connected to pin 12 gives 0 - 5V input range on pin 5 and connected to pin 9 gives -2.5V to 2.5V range.
	11	I	ANORI	Connected to pin 12 gives 0 - 5V output range on pin 6 (attenuated on pin 7), and connected to pin 9 gives -2.5V to 2.5V range.
	12		ANGND	Reference ground for analogue signals.
Paged Expansion				
	1		GRND	Ground. Signal ground connecting to screen on ribbon cable from other modules.
	2	O	HCLK	4MHz clock, buffered. Z80A clock is formed by the inversion of the buffer input signal.
	3	T	HA14	Buffered A14 address line from Z80A, direction reversed by HBUSAK.
	4	T	HA13	Buffered A13 address line, as pin 3.
	5	T	HD5	Buffered D5 data line, direction determined by ($\overline{\text{HRD}}$) XOR (HBUSAK)
	6	T	HA16	Buffered A16 address line, as pin 3.
	7	T	HD4	Buffered D4 data line, as pin 5.
	8	T	HD3	Buffered D3 data line, as pin 5.
	9	T	HD6	Buffered D6 data line, as pin 5.
	10	T	HD7	Buffered D7 data line, as pin 5.
	11	T	HA11	Buffered A11 address line, as pin 3.
	12	T	HA10	Buffered A10 address line, as pin 3.
	13	T	HA8	Buffered A8 address line, as pin 3.
	14	T	HA9	Buffered A9 address line, as pin 3.

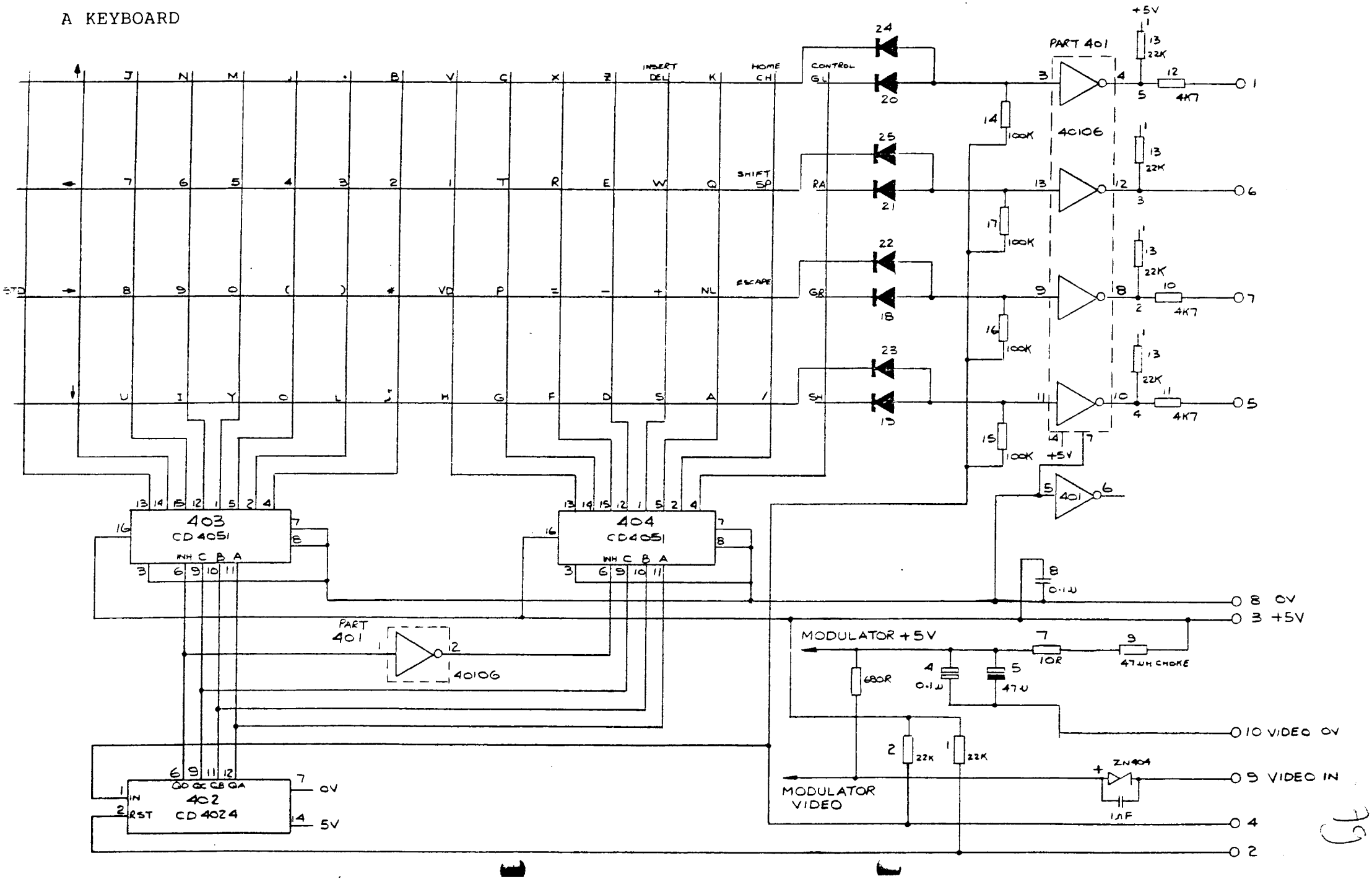
MAIN PCB Sheet 2 of 2

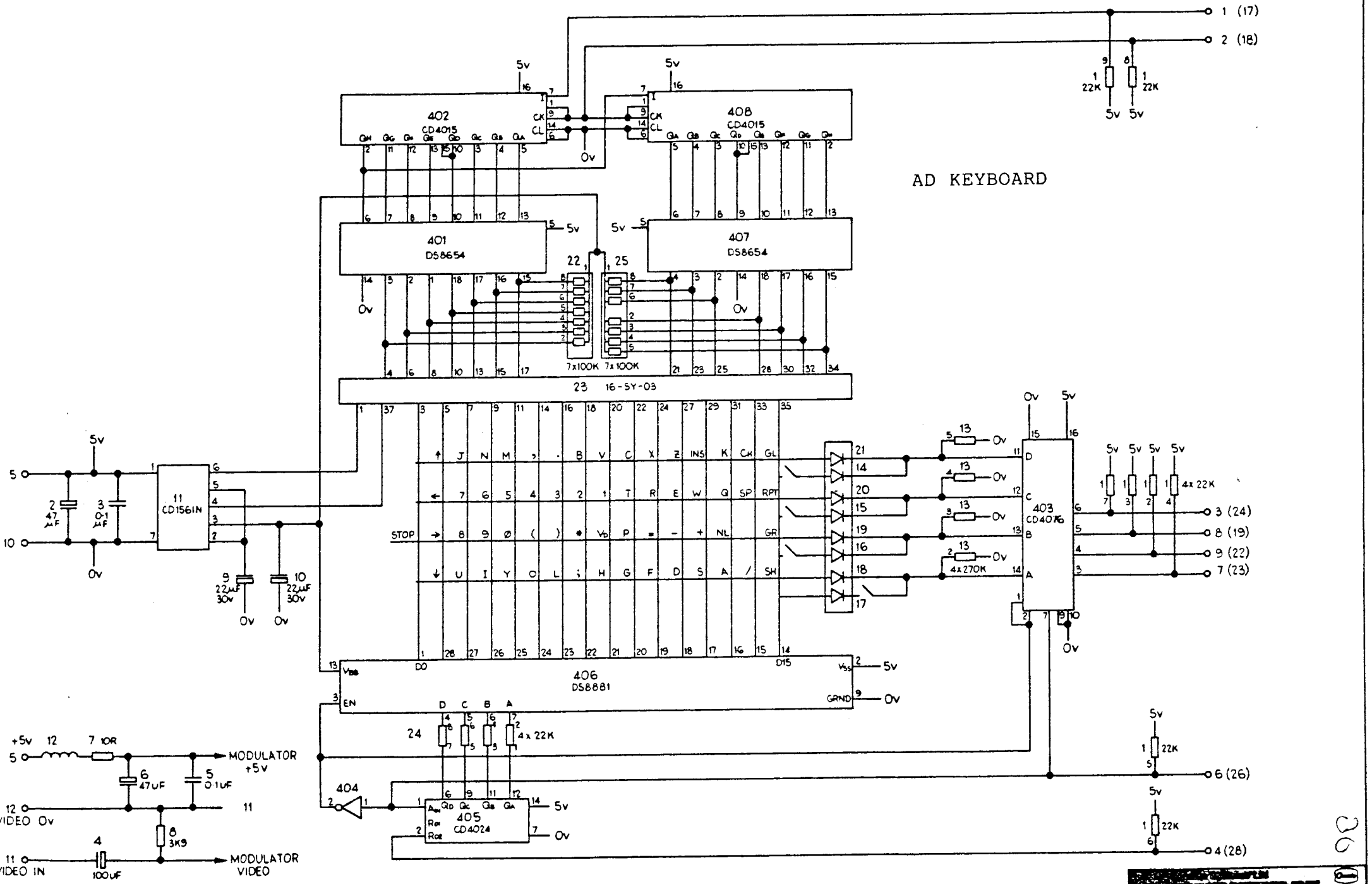


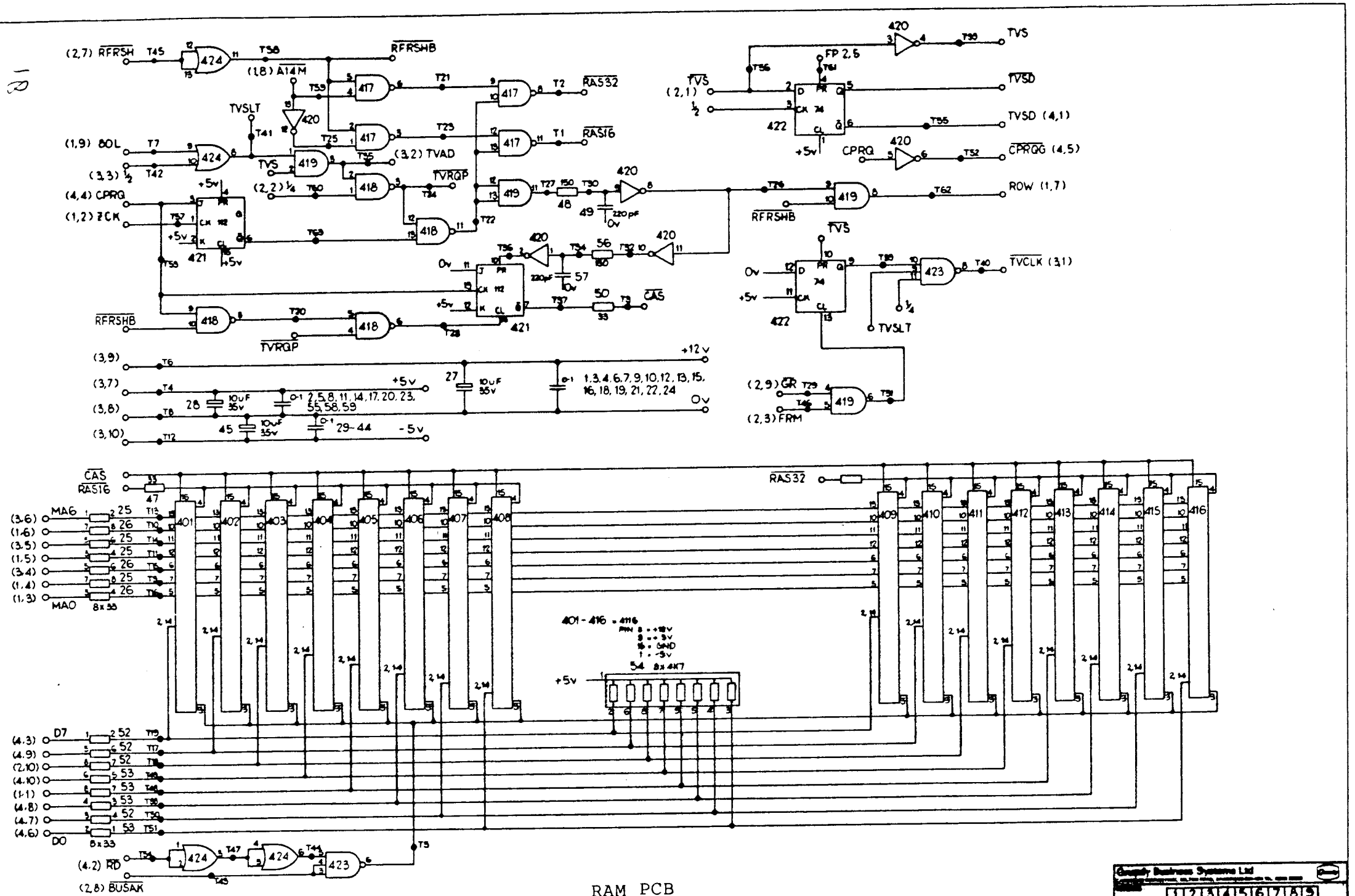
MAIN PCB Sheet 1 of 2



A KEYBOARD


J N M , . B V C X Z INSERT DEL K HOME CH CONTROL GL
7 6 5 4 3 2 1 T R E W Q SHIFT SP RA
8 9 0 ( ) * VD P = - + NL ESCAPE GR
U I Y O L ; H G F D S A / SH
24
20
25
21
22
18
23
19
PART 401
4010G
+5V
13 22K
12 4K7
14 100K
17 100K
16 100K
15 100K
10 4K7
11 4K7
1
6
7
5
403
CD4051
INH C B A
404
CD4051
INH C B A
401
8 0.1U
8 0V
3 +5V
PART 401
4010G
MODULATOR +5V
7 10R
9 47UH CHOKE
680R
4 0.1U
5 47U
10 VIDEO 0V
2 22K
1 22K
ZN404
1NF
9 VIDEO IN
MODULATOR VIDEO
4
2
402
CD4024
IN
RST
QD QC QB QA
7 0V
14 5V


AD KEYBOARD
402 CD4015
408 CD4015
401 DS8654
407 DS8654
7x100K
7x100K
23 16-SY-03
406 DS8881
403 CD4076
405 CD4024
404
11 CD1561N
4x 22K
4x270K
MODULATOR +5V
VIDEO Ov
VIDEO IN
100uF
MODULATOR VIDEO
1 (17)
2 (18)
3 (24)
8 (19)
9 (22)
7 (23)
6 (26)
4 (28)



RAM PCB

Gregory Business Systems Ltd

1	2	3	4	5	6	7	8	9	

ID-7-82 B

Interfaces. Signal Description and Connector Pinout.

PIN	SIGNAL TYPE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
Power			
1	O		Ground for supply flex screen.
2	I	+12I	Source for 12V internal rail. Accepts 13.0 to 14.5V applied. Consumes about 110MA.
3	I	-12I	Source for -12V and-5V internal rails. Accepts -14.0 to -12V applied. Consumes about 25mA.
4			Absent (Accepts polarising blank in power lead plug).
5	I	GND	Ground return for all internal rails.
6	I	+5I	Source for 5V internal rail. Accepts 6.5V to 7.5V applied. Consumes about 900MA.
Tape 1 and 2			
1	I	IFT1	Input from tape recorder earphone jack. Loaded by 10nF in series with 39K. Input voltage >500mV required
2	O	OTT1	Output to tape recorder microphone jack. 25mV nominal. Slew rate 4msec/V. 250 ohms source impedance.
3		TMS1	Relay contact for tape motor. Open circuit DC voltage must be < 20V.
4		$\overline{\text{TMS1}}$	Relay contact for tape motor. Open circuit DC voltage must be < 20V.
5		TDTPLPO	Absent (Accepts polarising blank in tape lead plug).
6		GRND	Ground return for tape signals via flex screen to microphone jack ground only.
Communications			
1	I	SRDDK	Received serial data. Input > 2.5V (binary zero) or < 2.5V (binary one).
2	I	$\overline{\text{SCTSD}}$	Nominally not clear to send ($\overline{\text{CTS}}$ or $\overline{\text{RFS}}$). Used as handshake by software. Characteristics as for pin 1.
3			Absent (Accepts polarising blank in communications lead plug).
4	O	SDO	Transmitted serial data. Output > 8V (binary zero) or < -8V (binary one) for 3K ohms load. Power up leaves binary zero (contra V24 spec) until stream opened to device.
5	O	$\overline{\text{SRTSD}}$	Nominally not request to send ($\overline{\text{RTS}}$). Used as handshake by software. Characteristics as for pin 4 (No power up caveat).

PIN	SIGNAL TYRE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
6			Ground return for communications signals via flex screen.
Printer			
1			Unused.
2	I	$\overline{\text{SCTSP}}$	Not clear to send ($\overline{\text{CTS}}$ or $\overline{\text{RFS}}$). Characteristics as for pin 1, communications interface
3			Absent (Accepts polarising blank in printer lead plug).
4	O	SPO	Transmitted serial data. Characteristics as for pin 4, communications interface.
5			Unused.
6			Ground return for printer signals via flex screen.
Monitor			
	O		Composite video to CCITT specification. Sourced at 75 ohm. Open circuit peak white voltage 2V, Black 1V, sync 0V. Frame 50Hz, line 15.625KHz, H sync 4µsec, V sync. 140 µsecs. Video blanking 4µsecs before and after H sync, and 8 µsecs before, 840 µsecs after V sync. Programmable video starts 11.5 µsec after and finishes 8.4 µsec before H sync and starts 2.4 m sec after, finishes 1.6 m sec before V sync.
UHF			
	O		UHF output modulated by video. Carrier 591.25 MHz (Ch 36), output 0- 1.5mV sourced at 75 ohms (RF). Output is halved when the monitor output is terminated by 75 ohms.
Expansion			
1		GRND	Ground. Signal ground connecting to screen on ribbon cable from other modules.
2	O	1/8C	4MHz clock. Z80A clock is formed by inversion of this signal.
3	T	A14	A14 from Z80A.
4	T	A13	A13 from Z80A.
5	T	D5	D5 from Z80A.
6		RMSL	Address modifier. Unused signal.*(HA 16).
7	T	D4	D4 from Z80A.
8	T	D3	D3 from Z80A.

PIN	SIGNAL TYPE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
9	T	D6	D6 from Z80A.
10	T	D7	D7 from Z80A.
11	T	A11	A11 from Z80A.
12	T	A10	A10 from Z80A.
13	T	A8	A8 from Z80A.
14	T	A9	A9 from Z80A.
15	T	A12	A12 from Z80A.
16	T	A7	A7 from Z80A.
17	T	A3	A3 from Z80A.
18	T	A2	A2 from Z80A.
19	T	A1	A1 from Z80A.
20	T	A0	A0 from Z80A.
21	T	D0	D0 from Z80A.
22	T	D1	D1 from Z80A.
23			Unused. Early machines have a signal connected to this pin. Connect 10K ohm pull-up to 5V.*(HP11).
24	T	D2	D2 from Z80A.
25	T	A5	A5 from Z80A.
26	T	A6	A6 from Z80A.
27	I	$\overline{\text{RAMENB}}$	Not RAM enable. Logic zero routes Z80 memory request to RAM, conditional on pin 48 signal. Unconnected state biased to logic one. *(HP3).
28	I	EXRM2	External A15 signal. Used by paging circuits in the expansion interface module *(HP2).
29	I	EXRM1	External A14 signal. Used as under pin 28 *(HP1).
30	I	EXRM0	External A13 signal. Used as under pin 28 *(HP0).

PIN	SIGNAL TYPE	CIRCUIT MNEMONIC	SIGNAL DESCRIPTION
31	I	$\overline{\text{ROMOV}}$	Not address overide. Logic zero causes A13, A14, A15, to be replaced by external A13,A14,A15.*(GRQ).
32	OC	$\overline{\text{BUSRQ}}$	Not Bus request as supplied to Z80A ($\overline{\text{BUSRQ}}$).
33	O	$\overline{\text{M1}}$	Not M1 from Z80A ($\overline{\text{M1}}$)
34	O	$\overline{\text{RST}}$	Not reset as supplied to Z80A ($\overline{\text{RST}}$).
35	O	$\overline{\text{RFRSH}}$	Not refresh from Z80A ($\overline{\text{RFRSH}}$)
36	I	$\overline{\text{WAIT}}$	Not wait as supplied to Z80A ($\overline{\text{WAIT}}$).
37	T	A4	A4 from Z80A.
38	O	$\overline{\text{BUSAK}}$	Not bus acknowledge from Z80A ($\overline{\text{BUSAK}}$).
39	T	A15	A15 from Z80A.
40	T	$\overline{\text{WR}}$	Not write from Z80A ($\overline{\text{WR}}$)
41	OC	$\overline{\text{INT}}$	Not interrupt as supplied to Z80A ($\overline{\text{INT}}$).
42	T	$\overline{\text{RD}}$	Not read from Z80A ($\overline{\text{RD}}$)
43	I	$\overline{\text{NMI}}$	Not non-maskable interrupt as supplied to Z80A ($\overline{\text{NMI}}$) *($\overline{\text{HISLT}}$).
44	O	$\overline{\text{HALT}}$	Not halt from Z80A ($\overline{\text{HALT}}$) *(MPSL).
45	T	$\overline{\text{MREQ}}$	Not memory request from Z80A ($\overline{\text{MREQ}}$).
46	T	$\overline{\text{IORQ}}$	Not input/output request from Z80A ($\overline{\text{IORQ}}$).
47	I	PRTOV	Logic one inhibits internal IO port decode. Unconnected state biased to logic zero *(HP6).
48	I	RAMINH	Logic one inhibits Z80A request to internal RAM. Unconnected state biased to logic zero. *(HP5)
49	O		± 5volts (up to 200mA).
50	O	BUSRQ	Bus request as generated by video circuit *(HP4)

Abbreviations:

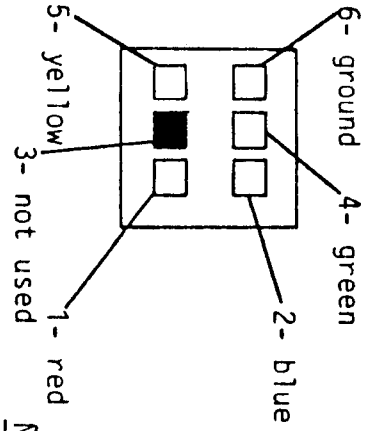
D0-D7	Data bits 0 to 7
A0-A15	Address bits 0 to 15
I	Input
O	Output
T	Tristate
OC	Open collector
*	Signal function changes on paged expansion highway.

TECHNICAL DIAGRAMS

1. Serial Interface with Microline 8Ø (& 82A)
2. Wiring to Smith Corona TP-1
3. NewBrain Software Serial port
4. Wiring to Superbrain
5. Location of functions/pin numbers
6. NewBrain connection to Infoscribe 1000 (2 page.)
7. " " " Triumph-Adler DRM80/PERTEC P80
8. " " " Amber 2400 (2 pages)
9. " " " Walters WM2000 Printer lead (2 pages)
10. " " " Dataproducts M-100 with Serial interface
11. " " " Penny and Giles Matrix 80 printer (2 pages)
12. " " "Penny and Giles Matrix 132 printer (2 pages)
13. " " " Microline 82A with high speed Serial Board (2 pages)
14. " " " Epson Mk 80 Type 3 Serial Interface Board (3 pages)
15. " " " Microline 80 and 82A with Low Speed serial board (2 p.)
16. " " " Epson RX80 with Serial Interface
17. " " " Olivetti DY 211
18 " " " Epson FX80 with serial interface.

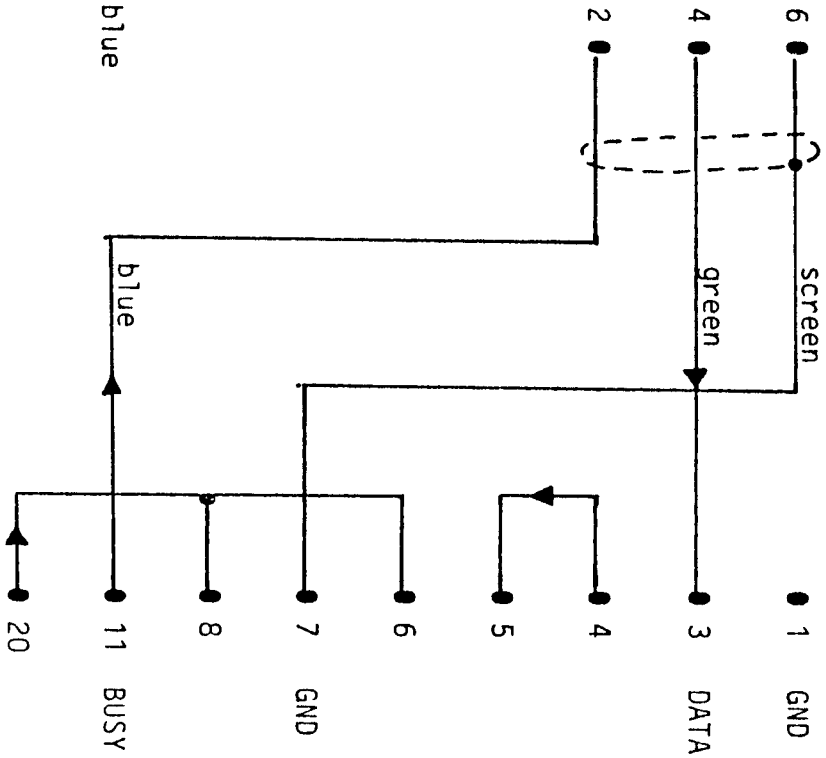
Technical Diagram No 1 Issue 1

This document supercedes all previous ones.
The colour coding refers to the leads in the stores in Cambridge on this date, 5 July 1982.

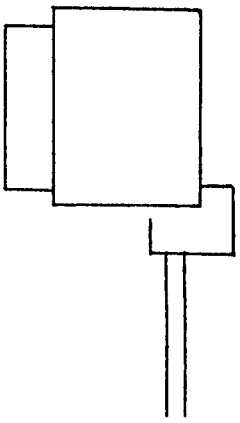


NEWBRAIN

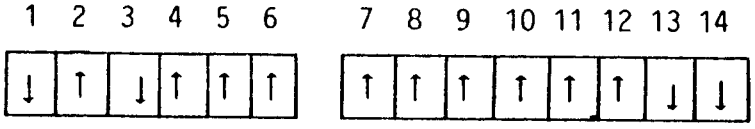
yellow & red are not used



25 way D-type plug



MICROLINE 80 (& 82A)
(Serial interface)



↑ ON
↓ OFF

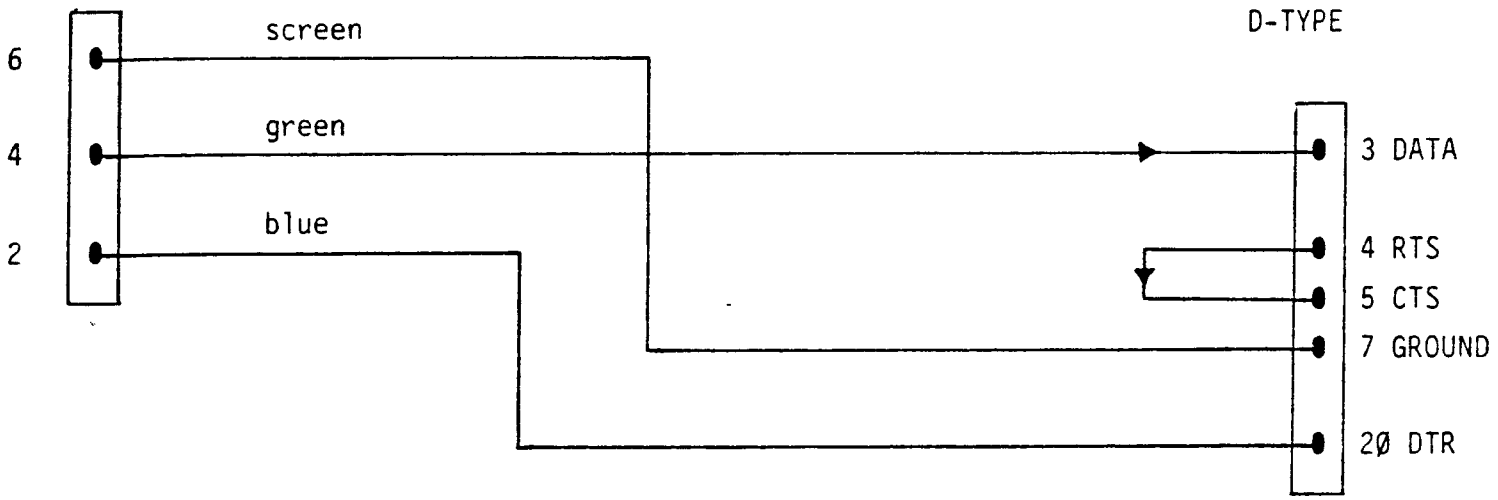
LINE 80 SWITCH SETTINGS
(on serial interface box)

NEWBRAIN

6 WAY
BERG

SMITH CORONA TP-I

25-WAY
D-TYPE

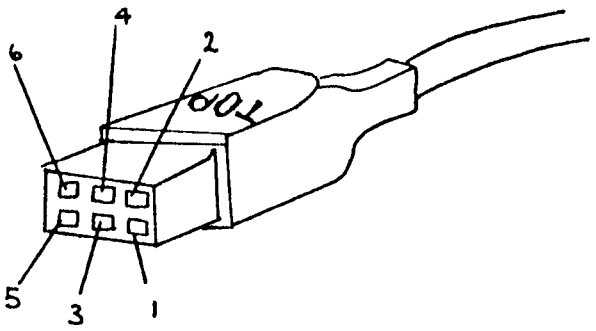


The Smith Corona is usually set to 1200 band on delivery, so to initiate communications at NewBrain end:
OPEN # 8,8,"1200".

Technical Diagram No. 3 Issue 1

NEWBRAIN SOFTWARE SERIAL PORT

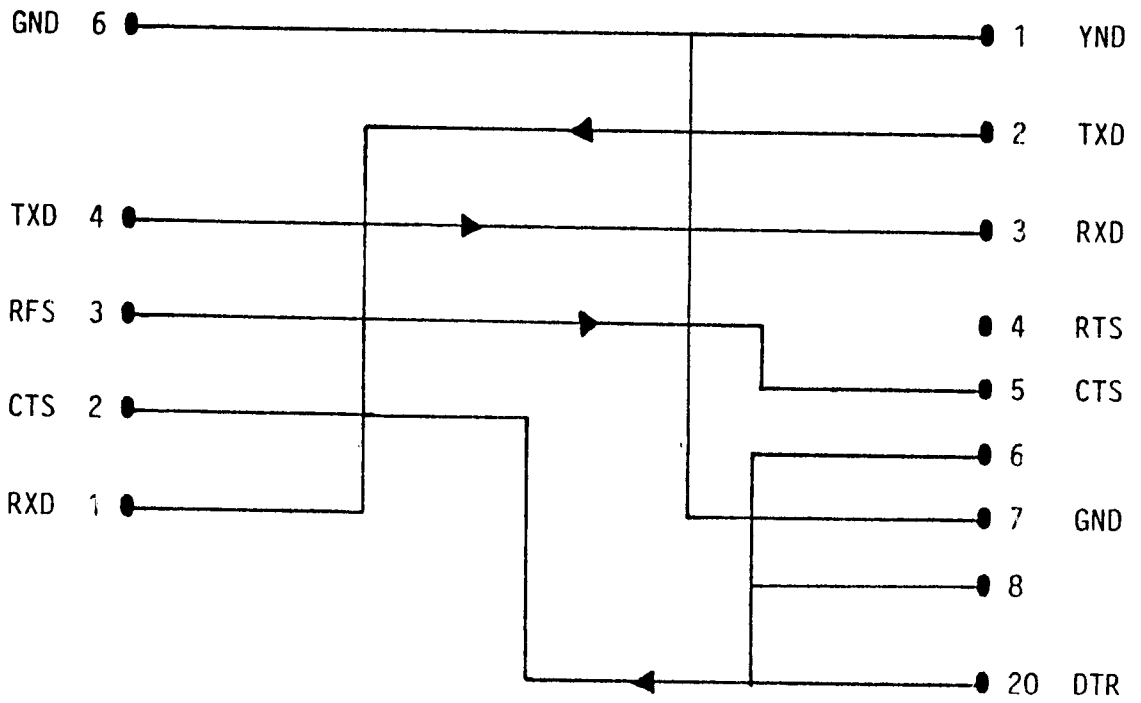
5 August 1982



1 -	RED	RX data
2 -	BLUE	CTS
3 -	not used	
4 -	GREEN	TX data
5 -	YELLOW	DTR*
6 -	SCREEN	GROUND

* Referred to elsewhere as RTS, RFS, SSD

6 April 1982



NEWBRAIN

6 WAY BERG
CONNECTOR

SUPERBRAIN

25 WAY
D-TYPE

Technical Diagram 5

November 1982

PRINTER

Pin Number	Function
2	Clear to send
3	POL
4	DATA OUT
6	GROUND

Output to printer is 8 bits, no parity with 2 stop bit, and 1 start bit.

After each byte (including carriage return) the NewBrain waits until the printer gives clear to send before sending the next byte.

On signal to the NewBrain is at >3V and off at <-3v
On signal from the NewBrain is at >9v and off at <-9v

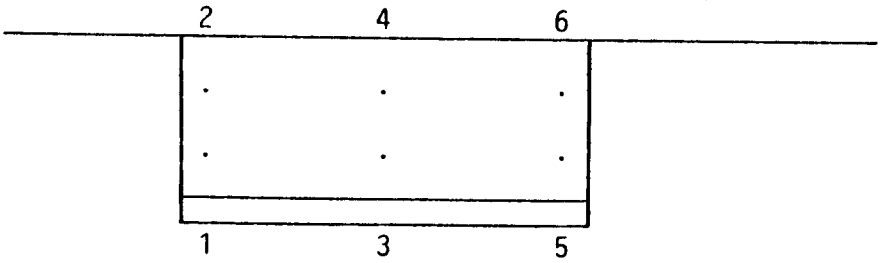
Communications

Pin Number	Function
1	DATA IN
2	CTS
3	POL
4	DATA OUT
5	RTS
6	GROUND

Cassette Ports

Pin Number	Function
1	INPUT
2	OUTPUT
3	TMS 1 (2)) relay contacts
4	TMS 1 (2))
5	POL
6	GROUND

Signal to and from the cassette is on two frequencies 1200 and 2400 Hg and conforms to NewBrain's own standard.



92

Technical Diagram 5

BUS EXTENSION CONNECTOR MODEL A NB12117/3 November 1982

PIN		FUNCTION	PIN		FUNCTION
1		GND	26	Z80	A6
2		1/8	27		$\overline{\text{RAMENB}}$
3	Z80	A14	28		EXRM2
4	Z80	A13	29		EXRM1
5	Z80	D5	30		EXRM0
6		RMSL	31		$\overline{\text{ROMOV}}$
7	Z80	D4	32	Z80	$\overline{\text{BUSRQ}}$
8	Z80	D3	33	Z80	$\overline{\text{M1}}$
9	Z80	D6	34	Z80	$\overline{\text{RST}}$
10	Z80	D7	35	Z80	$\overline{\text{RFRSH}}$
11	Z80	A11	36	Z80	WAIT
12	Z80	A10	37	Z80	A4
13	Z80	A8	38	Z80	$\overline{\text{BUSAK}}$
14	Z80	A9	39	Z80	A15
15	Z80	A12	40	Z80	$\overline{\text{WR}}$
16	Z80	A7	41	Z80	$\overline{\text{INT}}$
17	Z80	A3	42	Z80	$\overline{\text{RD}}$
18	Z80	A2	43	Z80	$\overline{\text{NMI}}$
19	Z80	A1	44	Z80	$\overline{\text{HALT}}$
20	Z80	A0	45	Z80	$\overline{\text{MREQ}}$
21	Z80	D0	46	Z80	$\overline{\text{IORQ}}$
22	Z80	D1	47		PRTOV
23		-spare to RAM board	48		RAMINH
24	Z80	D2	49		+5
25	Z80	A5	50		BUSRQ

CONNECTION OF FACIT 4510 TO NEWBRAIN

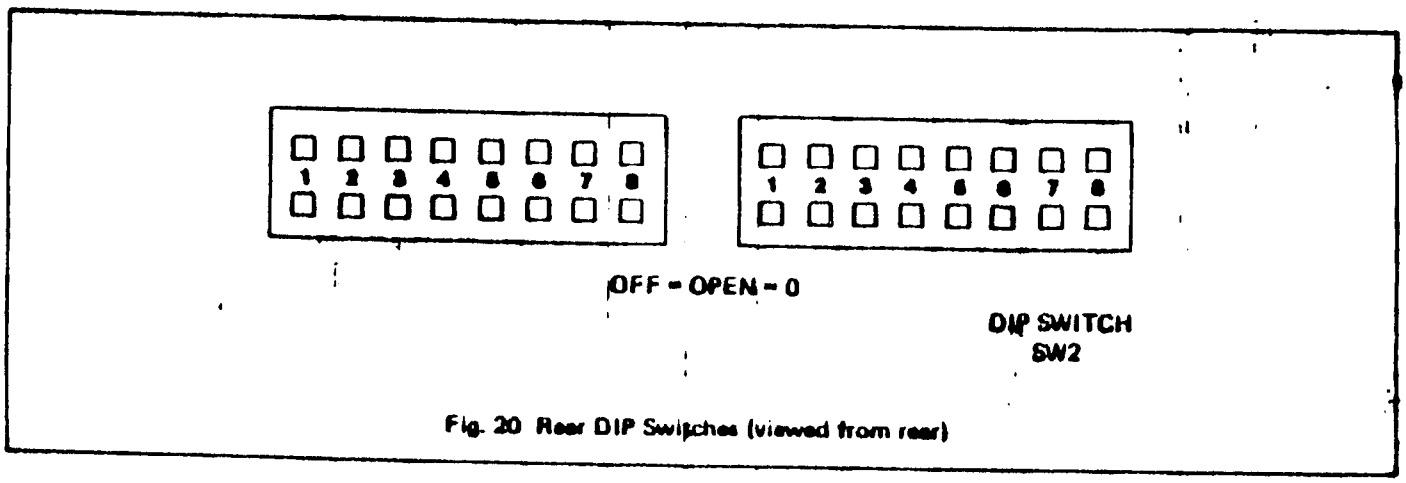


Fig. 20 Rear DIP Switches (viewed from rear)

REAR DIP SWITCHES

DIP SWITCH 1

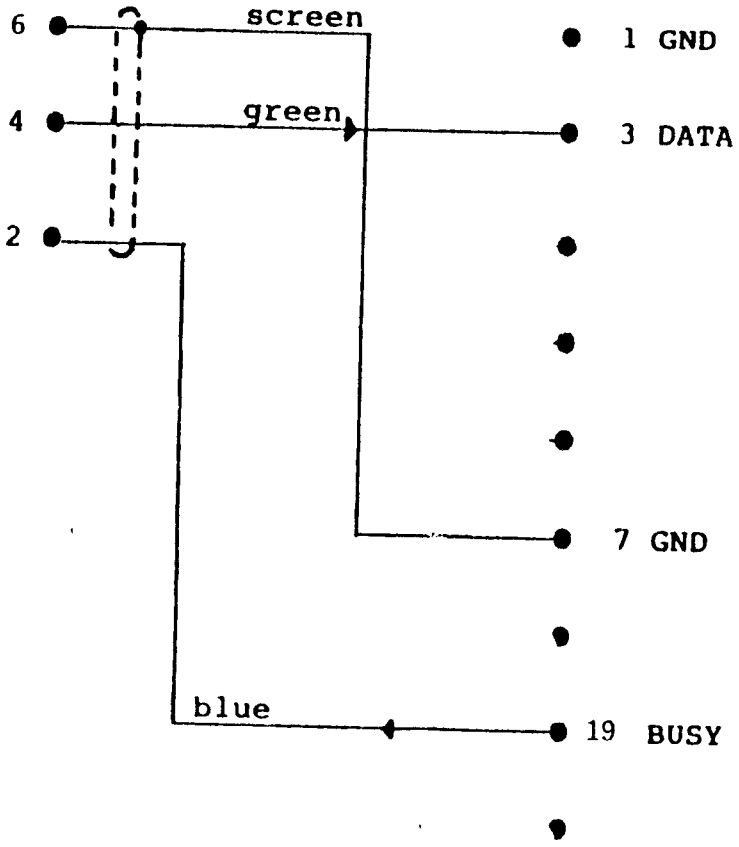
1	2	3	4	5	6	7	8
0	0	1	1	0	0	0	0

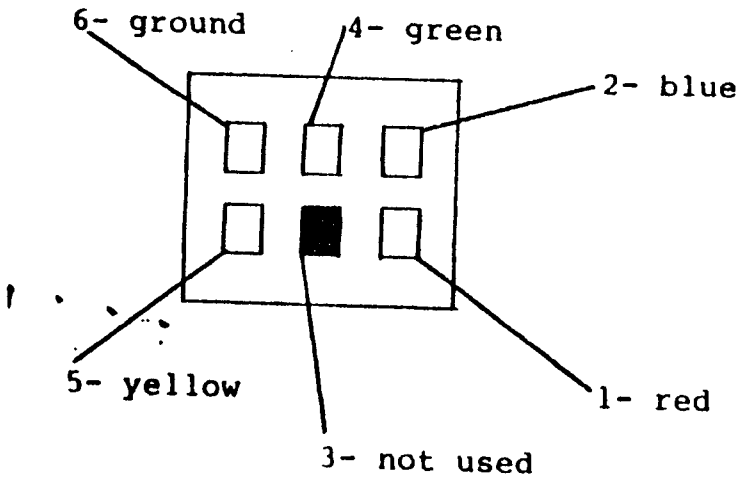
DIP SWITCH 2

1	2	3	4	5	6	7	8
0	1	1	1	0	0	0	0

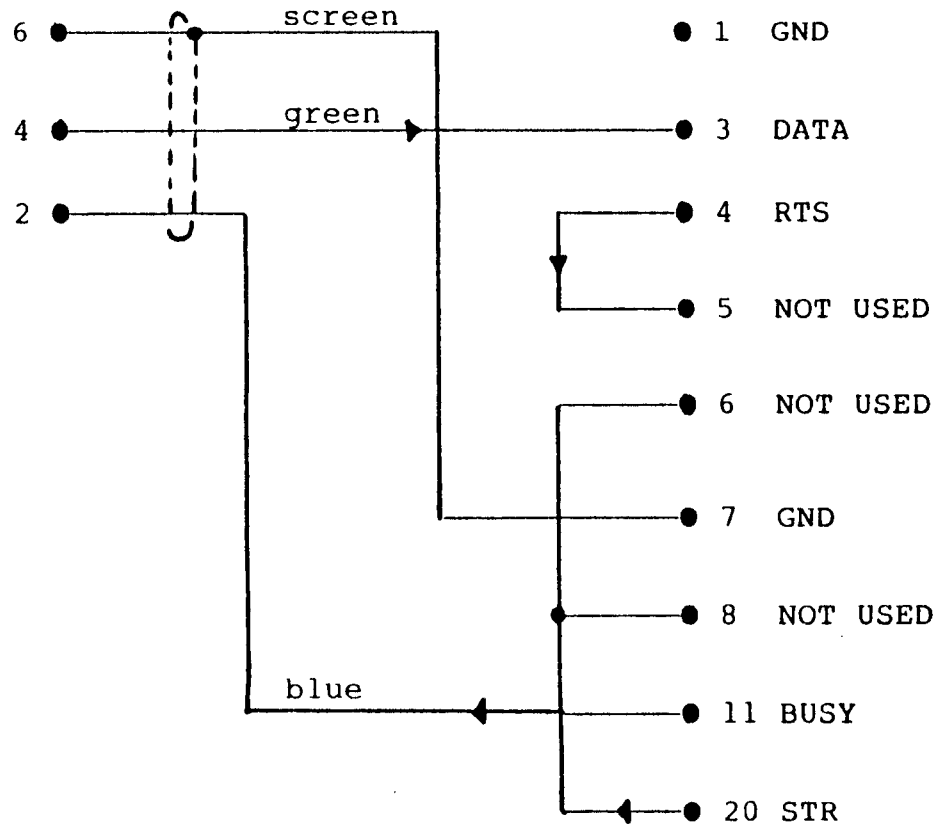
Technical Diagram

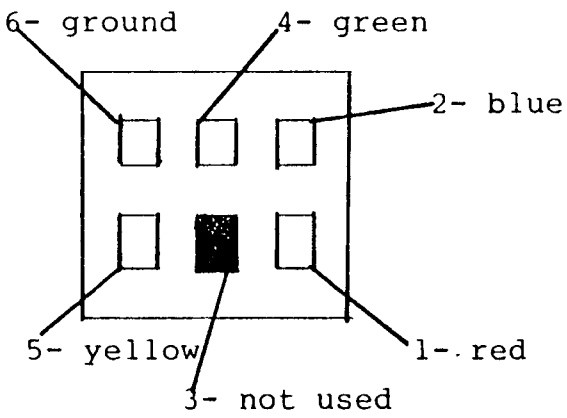
CONNECTION OF FACIT 4510





CONNECTION OF NEWBRAIN TO INFOSCRIBE 1000





STANDARD 25-WAY CANNON DC-25-P

Figure 5-1. control Panel Switch/Indicator Indentification.

Ensure that switch 5 is off and switch 6 is on

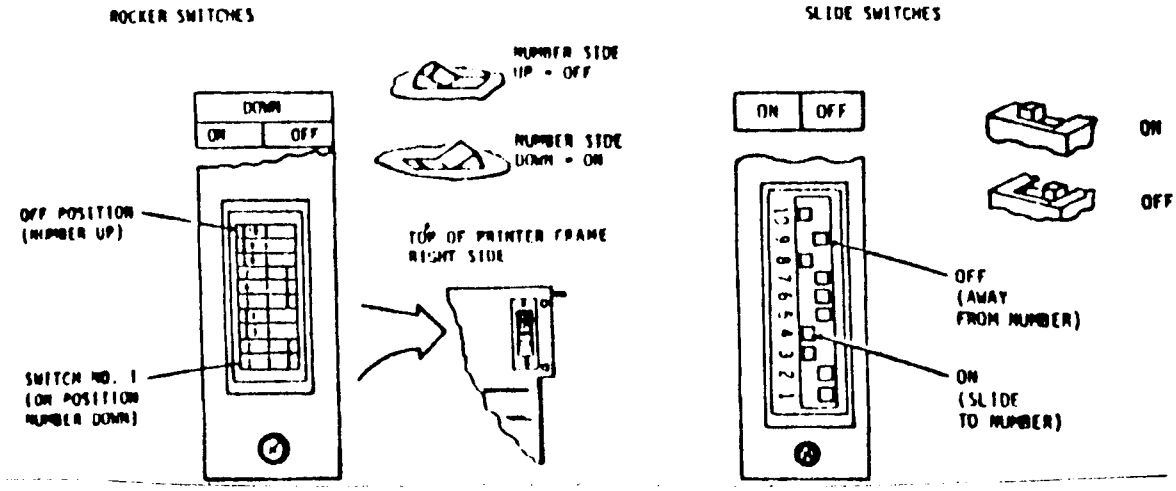


Figure 5-2. Format Switch Identification

CONNECTION TO INFOSCRIBE 1000

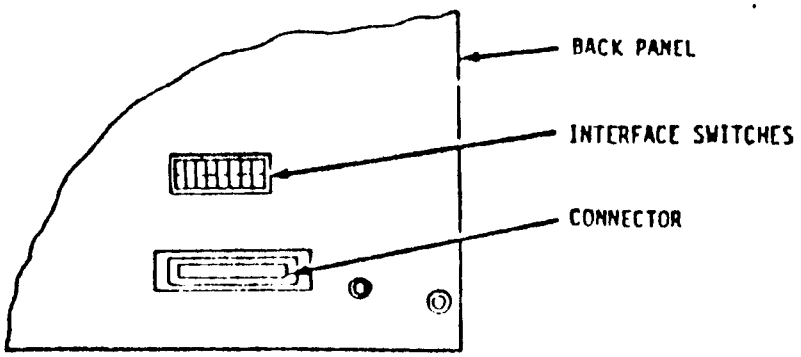


Figure 4-6. Interface Switch Functions

1. Serial Interface Switch Settings (PN 9030022 and 9030215)

SWITCH POSITION

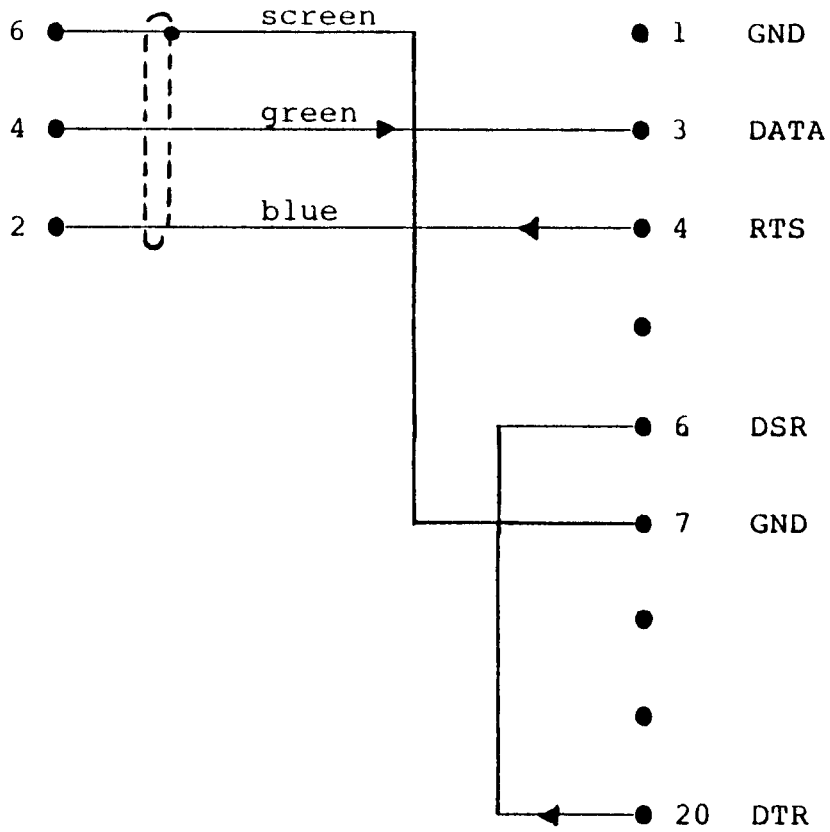
1	2	3	4	5	6	7	8
ON	OFF	OFF	OFF	OFF	OFF	OFF	OFF

2. Communications Interface Switch U9 (PN 9030168

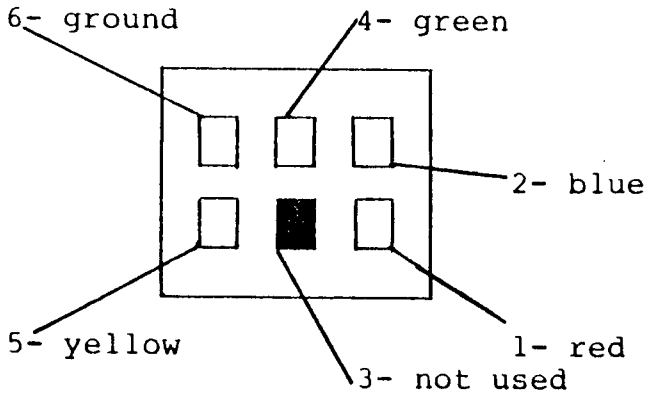
SWITCH POSITION

1	2	3	4	5	6	7	8	9	0
OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF	OFF

CONNECTION TO TRIUMPH-ADLER DRM80/PERTEC P 80



STANDARD 25-WAY CANNON DC-25-P

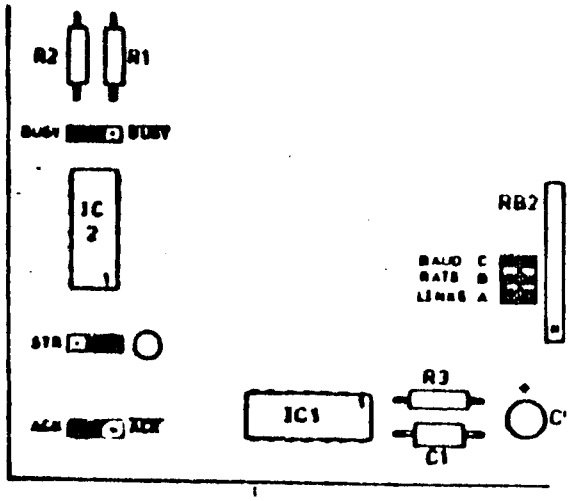


CONNECTION TO AMBER 2400:

Links required as shown in Fig 1.

Amber 2400 Links

"BUSY" LINKED BUSY



BAUD C: LINKED

RATE B: LINKED

LINKS A: LINKED

"STROBE" LINKED TO NOT STROBE

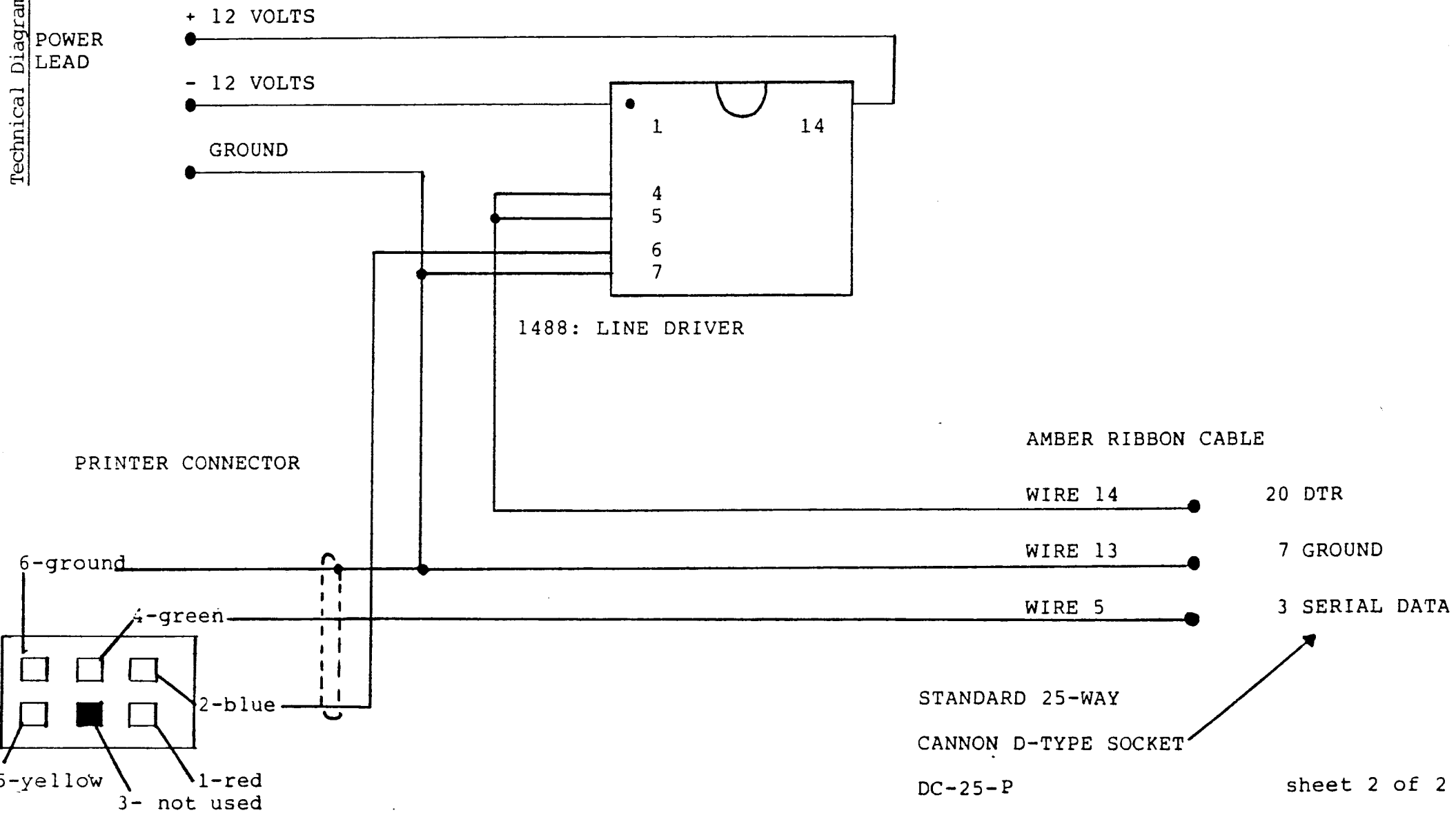
ie. $\overline{ST}$

FIGURE 1

"ACKNOWLEGE" LINKED TO ACK

Technical Diagram 8

CONNECTION OF AMBER 2400:



CONNNECTION TO WALTERS WM2000

PRINTER CONTROL SWITCHES (SWB)

The second group of switches are used for Printer control, and are constantly monitored, unless otherwise specified. They provide the following options: -

SWB (ON = + side of switch depressed)

S8	S7	S6	S5	S4	S3	S2	S1
OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF

COMMUNICATIONS CONTROL SWITCHES (SWA)

16 switches are located at the top of the main PCB internally. The first 8 switches (SWA) are read when power is switched on, and when switching from NOT SEL to SEL. They are used for interface control and have the following functions:

SWA (ON = + side of switch depressed)

S8	S7	S6	S5	S4	S3	S2	S1
ON	OFF	OFF	OFF	ON	ON	ON	ON

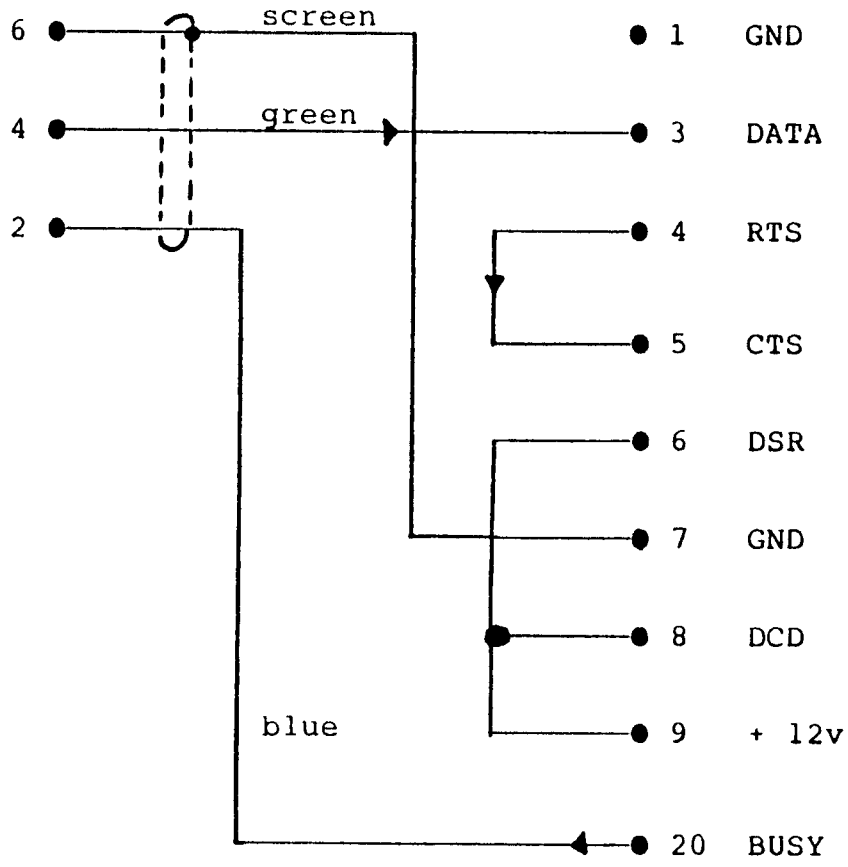
NB

This sets the printer to receive at its fastest rate (ie. 19,200 baud) and requires the printer port on the NewBrain to opened thus:

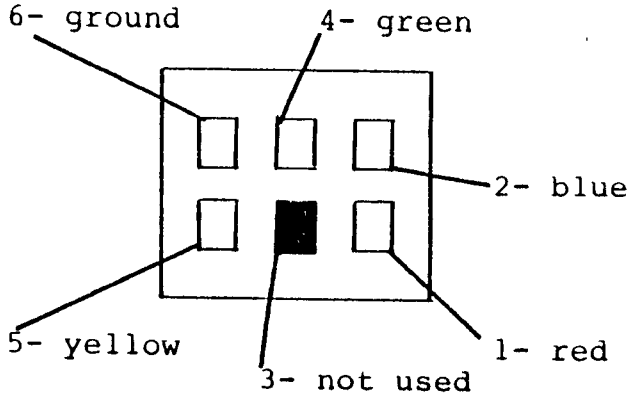
eg. OPEN#8,8"19200"

Technical Diagram 9

CONNECTION TO WALTERS WM2000 PRINTER LEAD



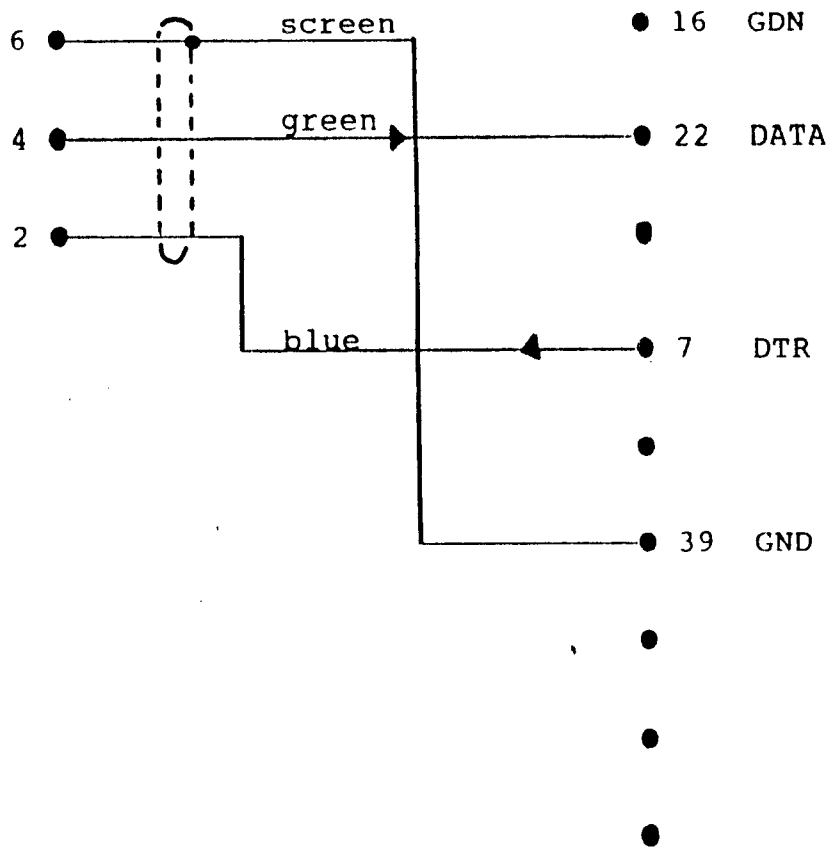
STANDARD 25 PIN CANNON PLUG
DC-25-P



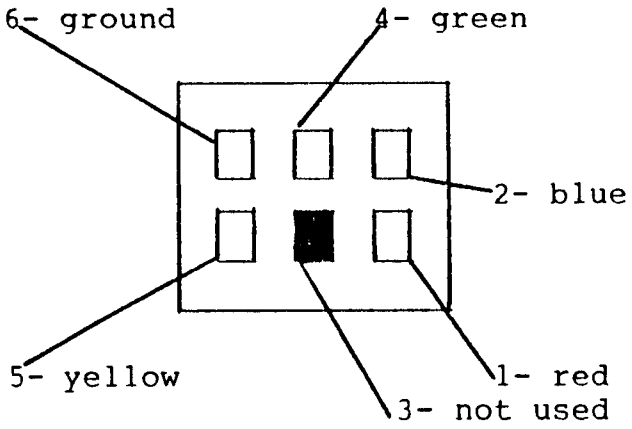
Technical Diagram 10

CONNECTION TO DATAPRODUCTS M-100 WITH SERIAL INTERFACE

:PRINTER LEAD TO 25-PIN AMP CONNECTOR



50-PIN AMP CONNECTOR



CONNNECTION OF NEWBRAIN TO PENNY AND GILES MATRIX 80 PRINTER

AUXILIARY AND CHARACTER SET SWITCH SETTINGS

SO1 OFF	SO2 OFF	SO3 OFF	SO4 OFF	SO5 OFF	SO6 ON
SO7 OFF	SO8 ON	SO9 OFF	SO10 OFF	SO11 OFF	SO12 OFF

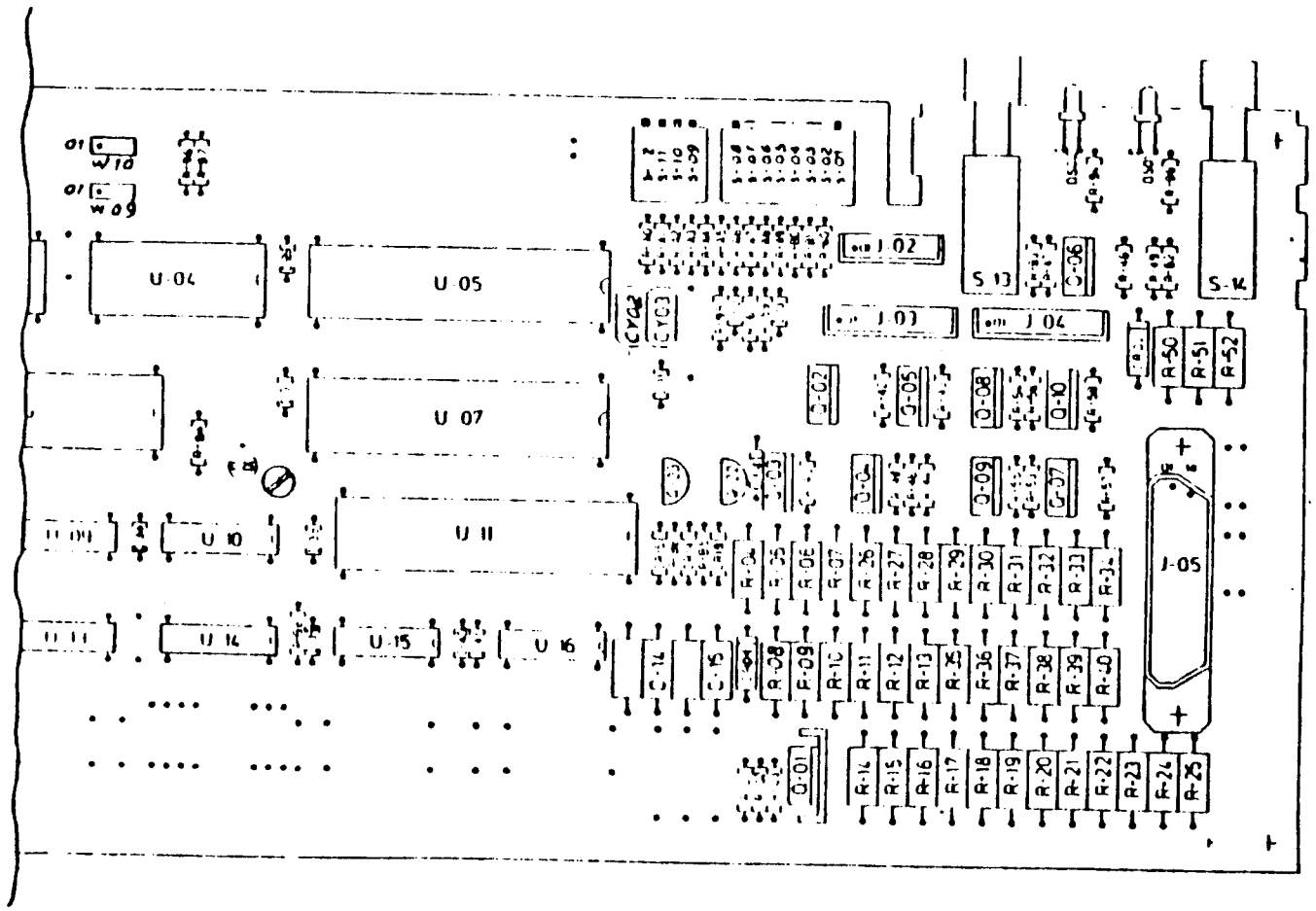
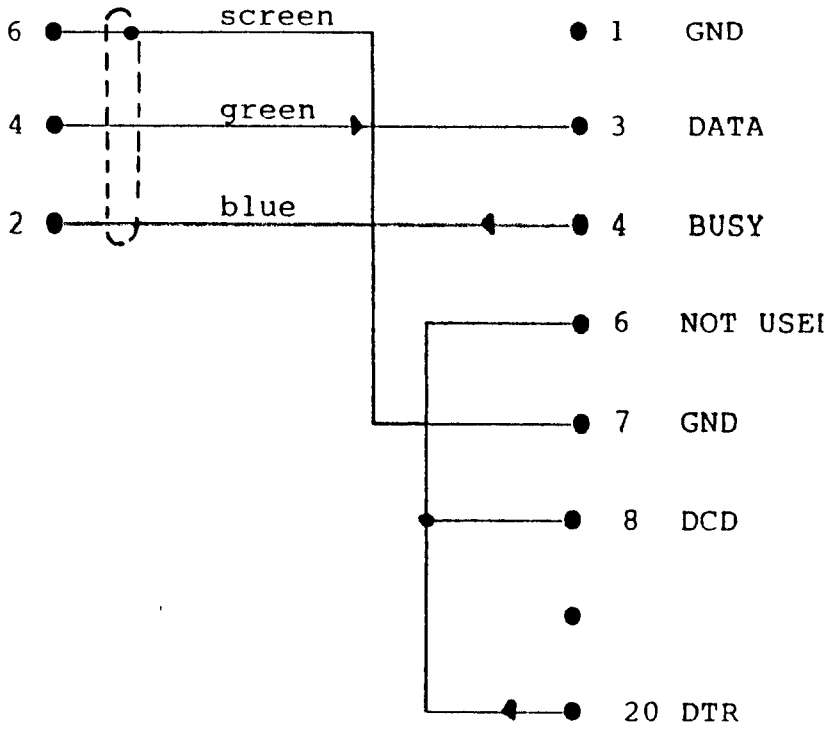
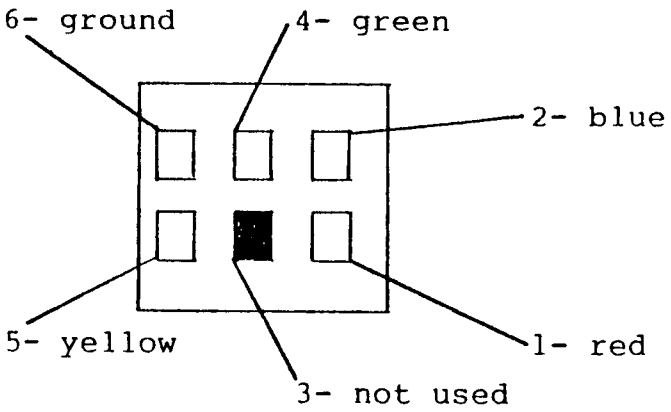


Figure 3-1 Switch Position Layout

CONNECTION OF NEWBRAIN TO PENNY AND GILES MATRIX 80

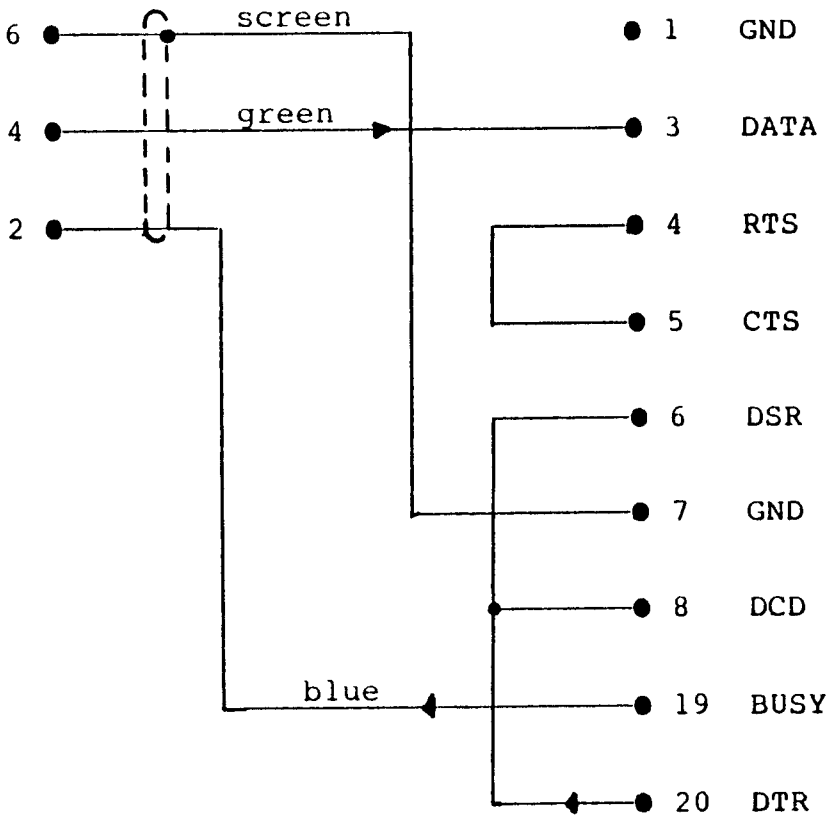


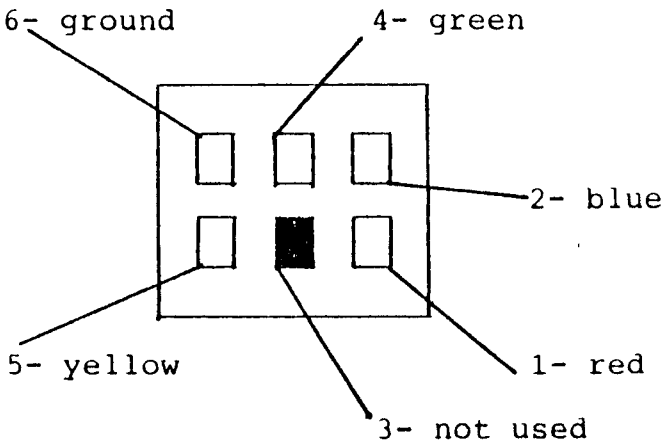
STANDARD 25- WAY CANNON
DC-25-P



Technical diagram 12 (1)

CONNECTION OF NEWBRAIN TO PENNY & GILES MATRIX 132 PRINTER





CONNECTION OF NEWBRAIN TO PENNY & GILES MATRIX 132 PRINTER

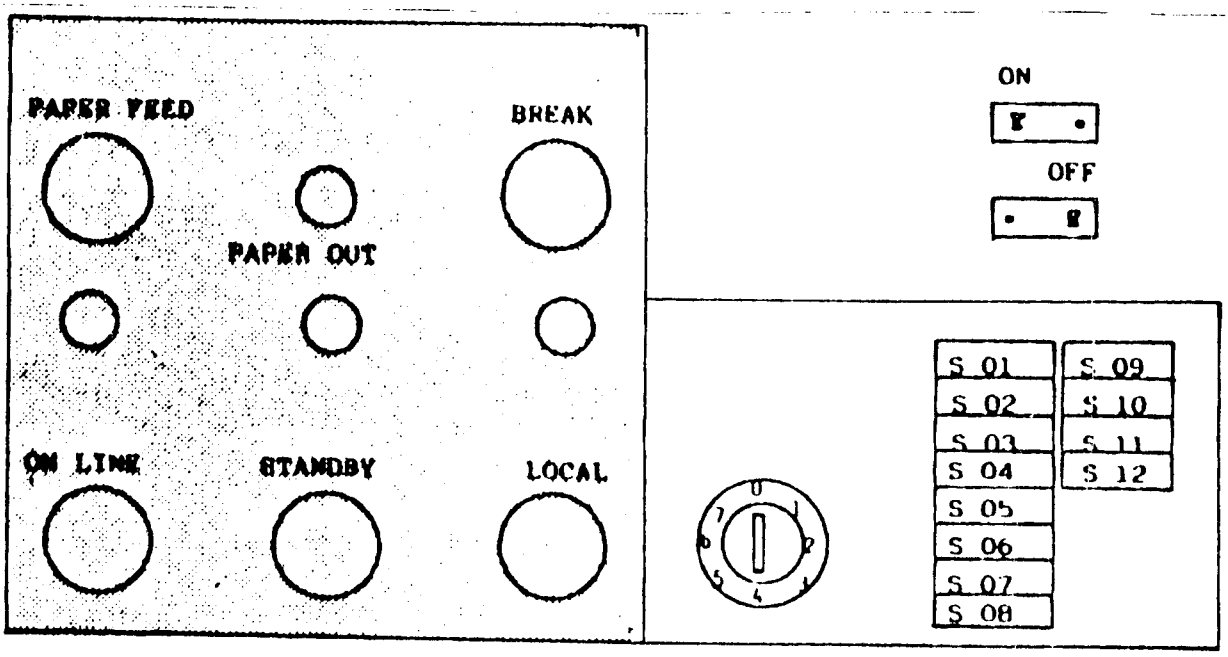


Figure 3-1. Layout of Protected Part of the Operator Panel

SWITCH SETTINGS

SO1	OFF	SO7	OFF
SO2	OFF	SO8	ON
SO3	OFF	SO9	ON
SO4	ON	SO10	ON
SO5	ON	SO11	ON
SO6	OFF	SO12	ON

CONNECTION OF MICROLINE 82A to NEWBRAIN WITH HIGH SPEED SERIAL BOARD

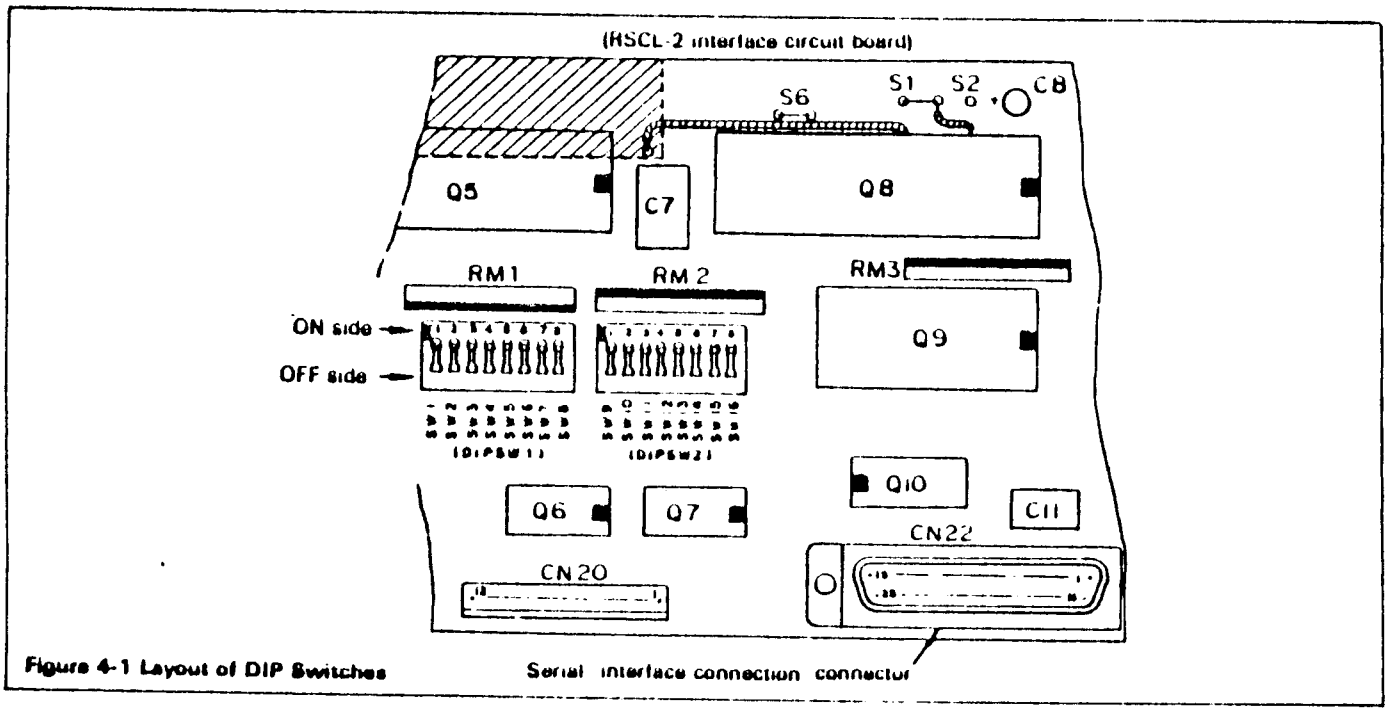


Figure 4-1 Layout of DIP Switches

SWITCH SETTINGS REQUIRED: RM1 and RM2 (Fig 4.1)

	1	2	3	4	5	6	7	8
RM1	OFF	ON	OFF	ON	ON	ON	ON	ON
RM2	ON	ON	ON	ON	OFF	OFF	OFF	ON

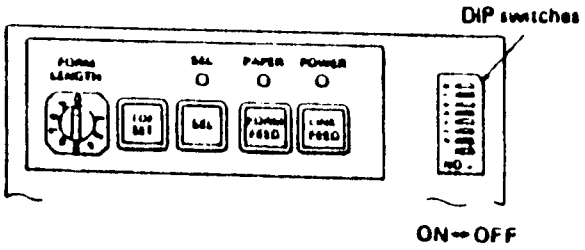
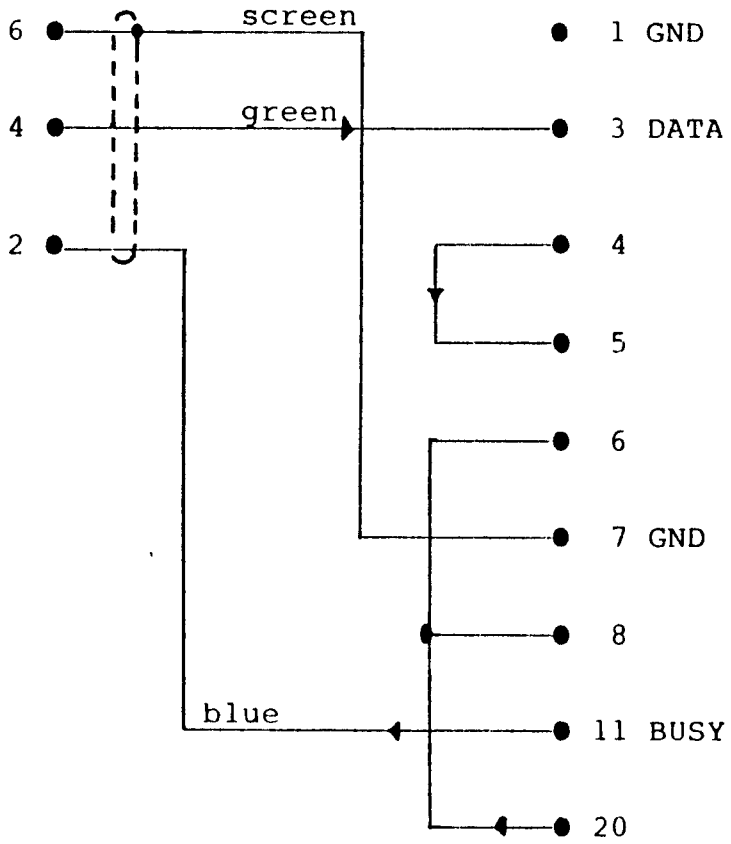


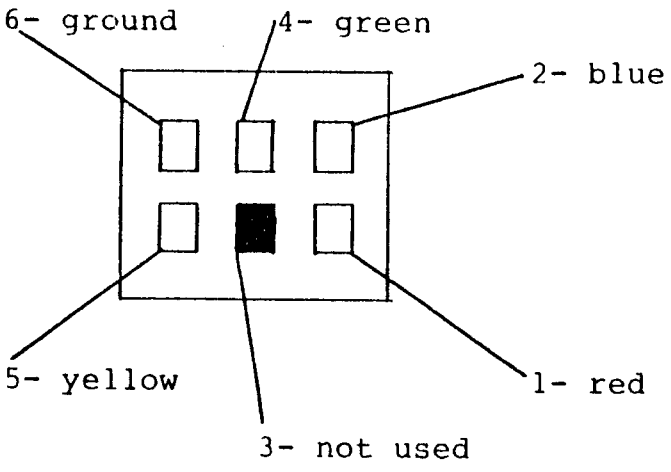
Figure 6-3 DIP Switches on Operating Panel

DIP SWITCH SETTINGS REQUIRED

8	OFF
7	OFF
6	OFF
5	ON
4	OFF
3	OFF
2	OFF
1	OFF

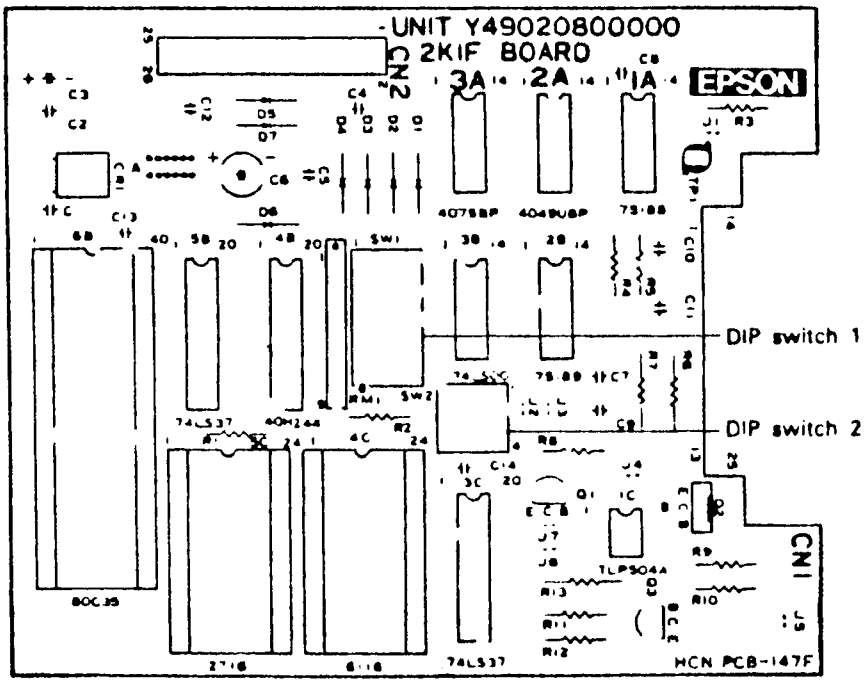
CONNECTION OF MIRCOLINE 82A WITH HIGH SPEED SERIAL BOARD





CONNECTION TO EPSON MK 80 TYPE 3 SERIAL INTERFACE BOARD

PARTS LOCATIONS Fig 6 Shows the component layout of the Serial Interface Board (Cat no 8145).



DIP SWITCH 1

1	ON
2	OFF
3	OFF
4	OFF
5	OFF
6	OFF
7	ON
8	OFF

DIP SWITCH 2

1	OFF
2	ON
3	OFF
4	ON

Fig 6 Component Layout

SETTING OF JUMPERS (FACTORY SET CONDITION)

J1	ON	J5	OFF
J2	ON	J6	OFF
J3	OFF	J7	OFF
J4	OFF		

NB
"on" denotes the connection of the jumper, while "off" denotes the disconnection of the jumper.

CONNECTION TO EPSON MX80 TYPE 3

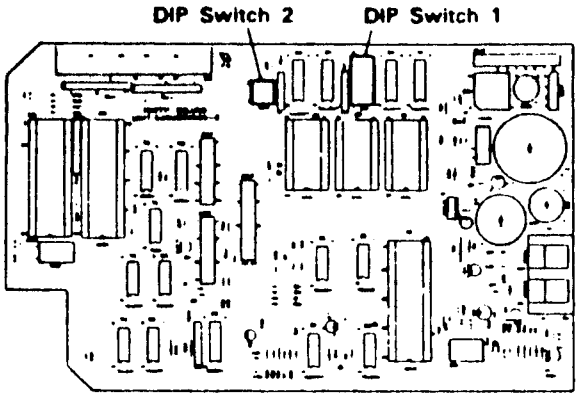


Fig. 53 Location of DIP Switches

There are two "DIP" (DUEL IN-LINE PACKAGE) switches in the HMTP board. (see Fig 53). The switches set to the left are ON to the right are OFF (see Fig 54)

DIP SWITCH 1
(as shown in Fig 54

8	ON	
7	ON	
6		OFF
5		OFF
4		OFF
3		OFF
2		OFF
1		OFF

DIP SWITCH 2

4		OFF
3		OFF
2	ON	
1	ON	

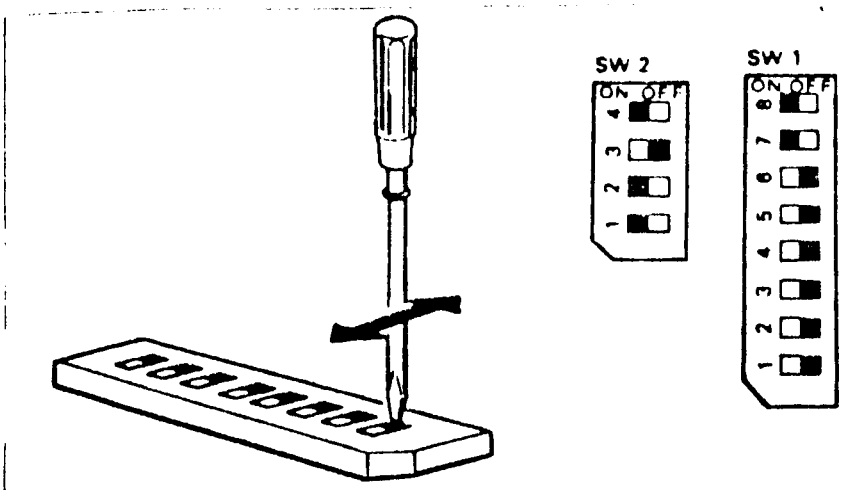
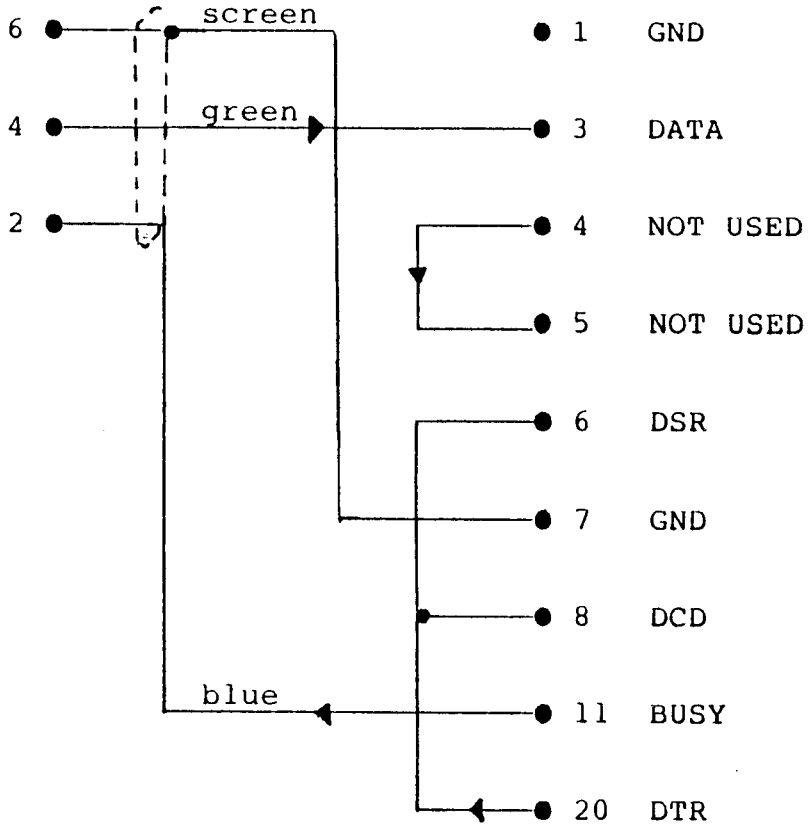
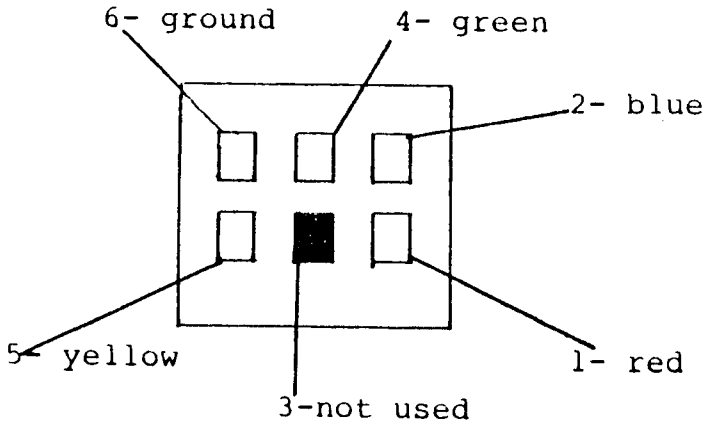


Fig. 54 Setting DIP Switches

CONNECTION TO EPSON MX80 TYPE 3 SERIAL INTERFACE BOARD:
PRINTER LEAD

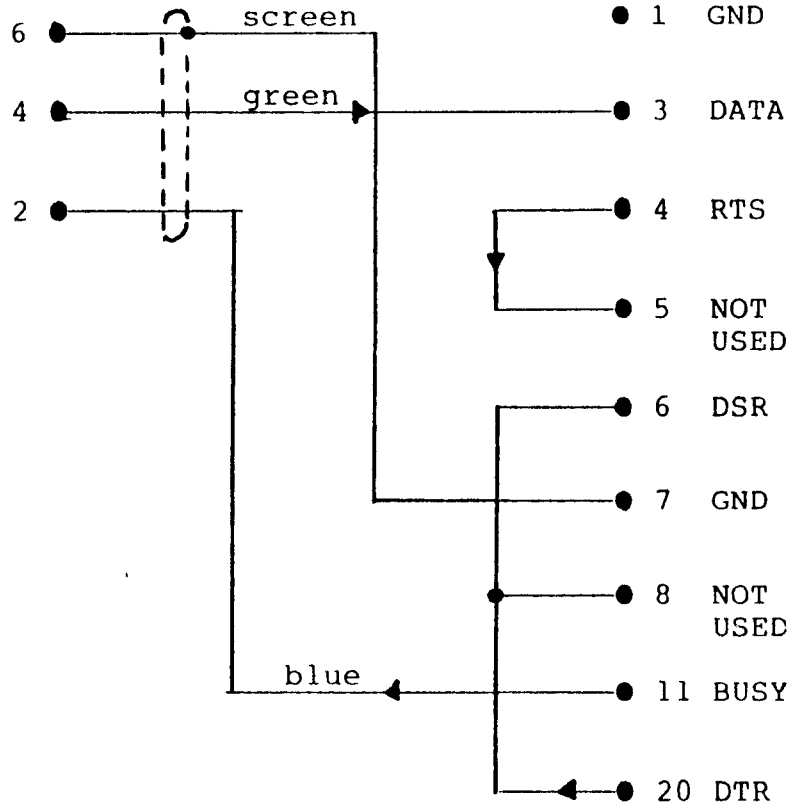


STANDARD 25-PIN CANNON PLUG
DC 25-P

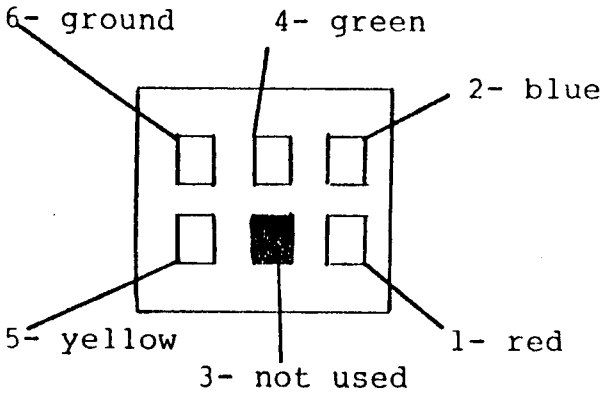


Technical Diagram 15 (1)

CONNECTION TO MICROLINE 80 and 82A WITH LOW SPEED SERIAL BOARD



STANDARD 25 PIN CANNON PLUG
DC-25-P



DIP SWITCH SETTINGS REQUIRED

8	ON	
7	ON	
6		OFF
5		OFF
4		OFF
3		OFF
2		OFF
1		OFF

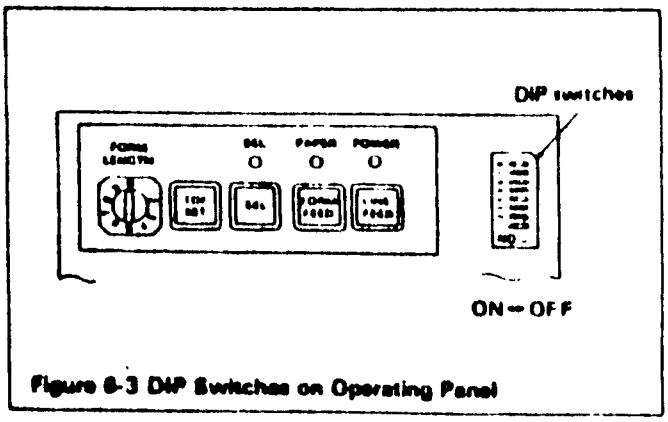


Figure 6-3 DIP Switches on Operating Panel

CONNECTION OF NEWBRAIN TO EPSON RX80 WITH SERIAL INTERFACE

MAIN BOARD: SWITCH 1

1	2	3	4	5	6	7	8
OFF	OFF	ON	OFF	OFF	ON	ON	ON

SWITCH 2

1	2	3	4
ON	ON	OFF	OFF

SERIAL INTERFACE - SWITCH 1

1	2	3	4	5	6	7	8
ON	OFF	OFF	OFF	OFF	OFF	ON	OFF

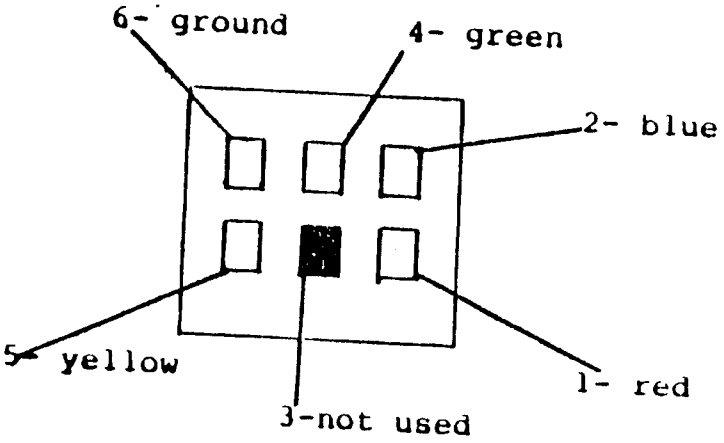
JUMPER CONNECTIONS (FACTORY SET CONDITION)

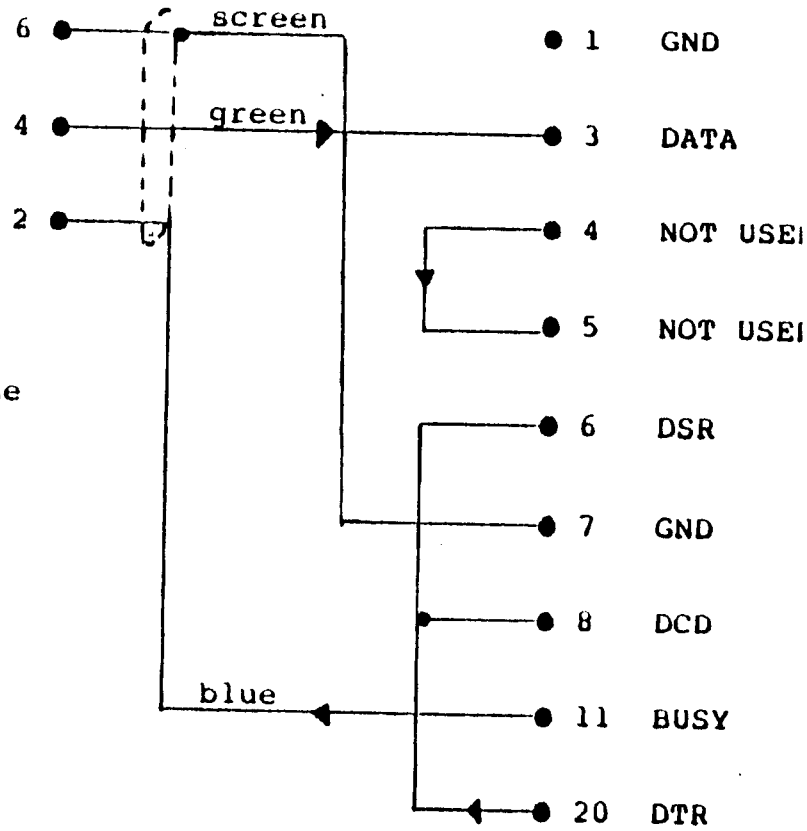
J1	J2	J3	J4	J5	JRS	JC	JNOR	JREV
OFF	OFF	OFF	OFF	ON	ON	OFF	ON	OFF

SERIAL INTERFACE - SWITCH 2

1	2	3	4
OFF	ON	OFF	ON

CONNECTION TO EPSON RX80 TYPE 3 SERIAL INTERFACE BOARD: PRINTER LEAD





STANDARD 25-PIN CANNON PLUG
DC 25-P

CONNECTION OF OLIVETTI DY 211 TO NEWBRAIN

Mode Select		Self Test	
2	0	2	0
1	0	1	0

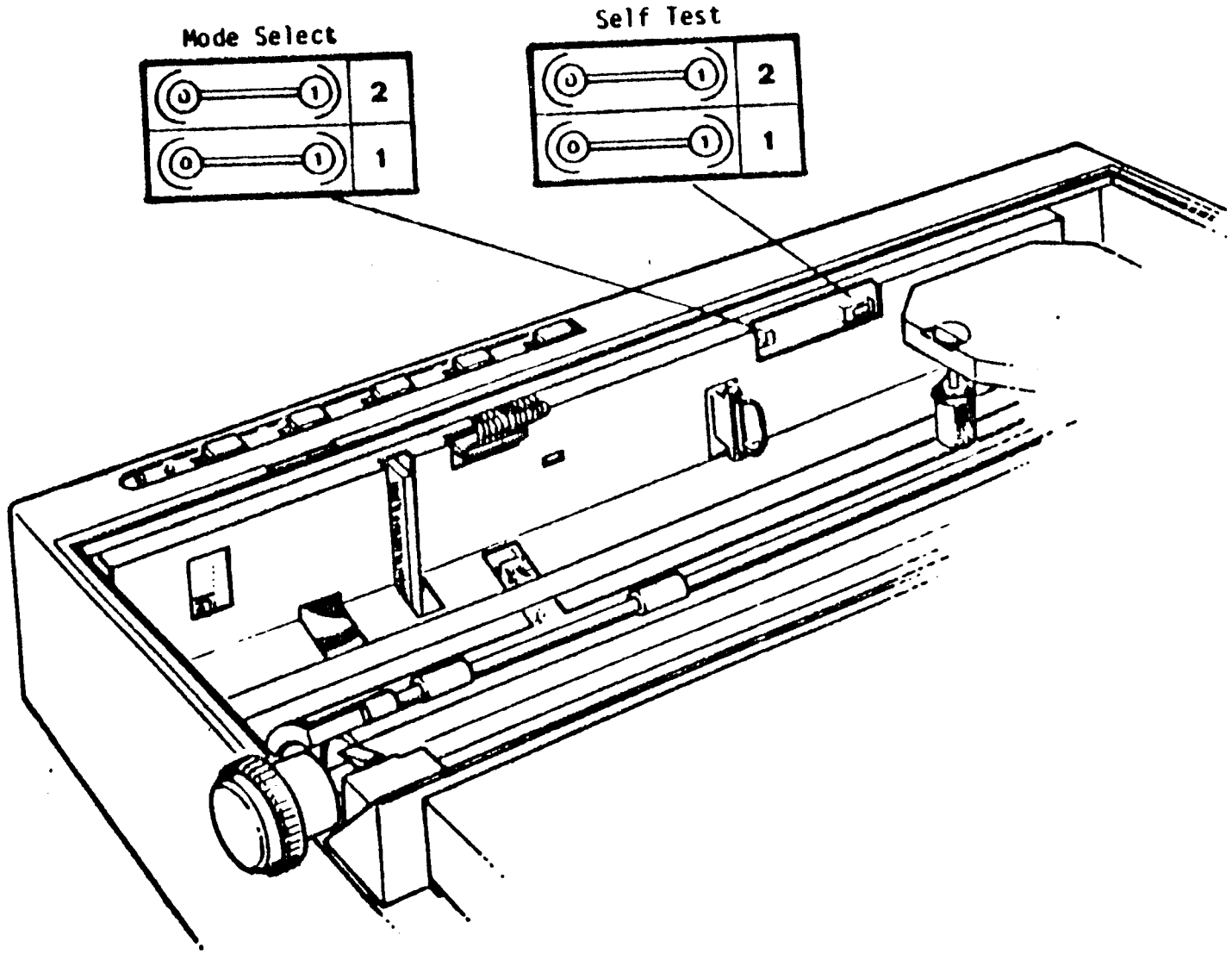
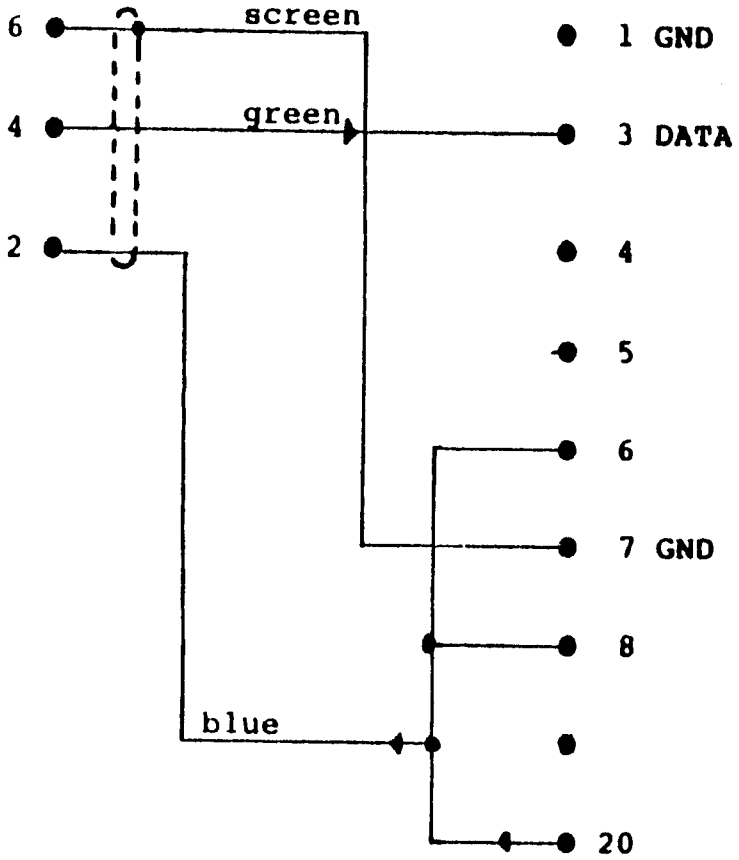
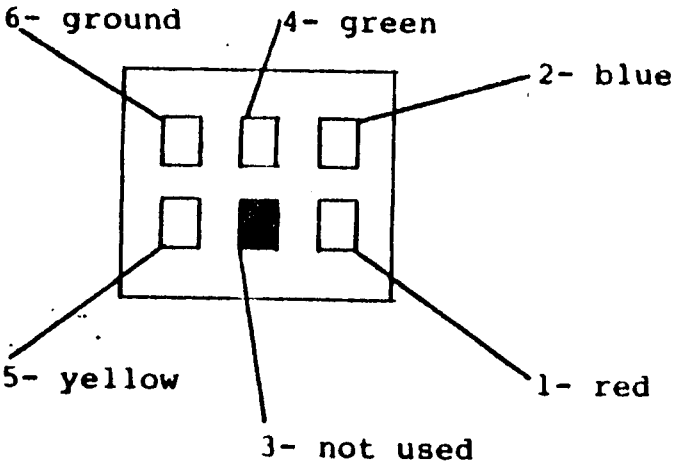


Figure 8. Mode Select and Self Test Switches
(as viewed from the rear of the printer)

CONNECTION OF OLIVETTI DY211 TO NEWBRAIN





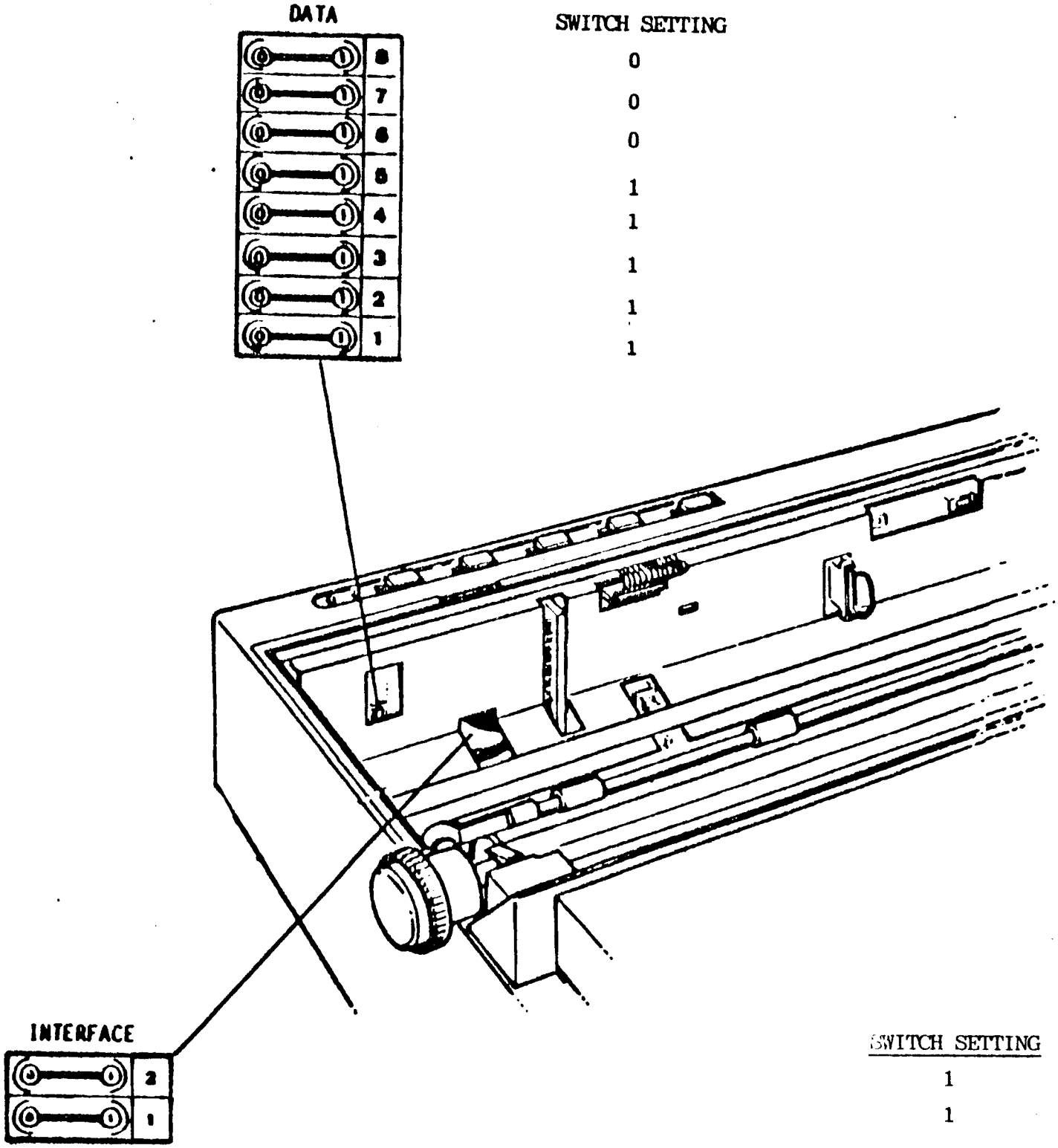


Figure 26. Location of DIP Switches for Serial Communication Options

CONNECTION OF NEWBRAIN TO EPSON FX80 WITH SERIAL INTERFACE

MAIN BOARD: SWITCH 1

1	2	3	4	5	6	7	8
OFF	ON	OFF	OFF	OFF	ON	ON	ON

SWITCH 2

1	2	3	4
ON	ON	OFF	OFF

SERIAL BOARD: DIP SWITCHES

1	2	3	4	5	6	7	8
ON	OFF	OFF	OFF	OFF	OFF	ON	OFF

JUMPER CONNECTIONS (FACTORY SET CONDITION)

J1	J2	J3	J4	J5	JRS	JC	JNOR	JREV
OFF	OFF	OFF	OFF	ON	ON	OFF	ON	OFF

SERIAL BOARD: SWITCH 2

1	2	3	4
OFF	ON	OFF	ON