April/May 1983
The magazine for the serious Sinclair user
95p

SINCLAIR PROJECTS

FIVE GREAT PROJECTS TO IMPROVE YOUR MACHINE

CENTRAL HEATING CONTROL

FIRE BUTTON

DECODER

SPECTRUM RS 232 INTERFACE

AID FOR DISABLED

COMPUTER AIDED DESIGN

"IT LOOKS NICE BUT WHAT THE HELL CAN I DO WITH IT?"

Every ZX SPECTRUM Print'n'Plotter Jotter has 100 pages of finely printed screen grids.

50 for the high resolution screen. 50 for the normal character screen.

With these at your disposal you can plan practically any graphics print-out to program into your computer.

The high-resolution PLOT grid shows every one of the 45,060 pixels! Every one printed. Every one with its co-ordinate numbers. This gives you enormous graphics power to DRAW, PLOT, CIRCLE, PLOT OVER and so on in any position or screen building up graphic drawings, charts, maps . . . in fact anything without the complications of guesswork, integer out of range, or wrongly positioned pixel colours which change PRINTed INK characters!

The normal character PRINT grids on the other hand will allow you to be specific about PRINT AT, TAB, PRINT OVER, SCREEN$ and INK/PAPER in direct co-ordination with PLOT . . . you see every page is printed on **high quality tracing paper** . . . ideal to overlay on to illustrations and 'copy' or co-ordinate.

And there's another bonus, because each page contains 24 User-definable grids — **2400 per pad!**

With 50 pages of PLOT grids, 50 pages at PRINT grids 2400 user-definable grids, a set of colour pens, a printed PIXEL RULER and our Special Offer of demo programs, IT'S THE BEST VALUE IN ZX GRAPHICS PROGRAMMING.

"PRINTER PAPER THAT WORKS? I DON'T BELIEVE IT!"

Five rolls of our PRINTER PAPER will only cost you £10.95. And you will gain a great deal more! It prints beautifully. It's not too thick. It's not too shiny. Print is black — not grey. It actually feeds through the machine! See for yourself. Only £10.95 for 5 rolls. Beware of expensive imitations!

"NOW WHICH KEY DO I PRESS TO SHOOT DOWN THE THING FROM MARS?"

Ever forgotten which key to press when playing a game?
Do you suffer from a mind-boggling mass of programmed keys?
Do you write programs that use different keys to perform various functions? If so, we have just the thing for your ZX SPECTRUM.

A pack of Print'n'Plotter Keyboard Overlays.

Just write the function or functions under each key you program and keep the overlay for the next time you play the game.
Of course there's lots of more uses you will find for our OVERLAYS, user defined characters, keyboard scanners, mathematical programs, business uses — to name just a few.

There's TEN OVERLAYS to a pack — so you can program with impunity! And they fit perfectly onto your standard ZX SPECTRUM KEYBOARD. Each OVERLAY is printed with the cursor movement key directions and there's room for program name etc.

"ZX81 GRAPHICS BETTER THAN SQUIGGLES AND BLOBS?"

Disappointed with ZX81 Graphics? Perhaps you haven't explored the vast possibilities.

Why not invest in a ZX81 JOTTER, FILM and our guide "ZX81 Graphics programming made easy"?

The JOTTER is a 100 page pad of PRINT grids and PLOT grids with all numbered co-ordinates. The FILM is a re-usable matt transparent polyester version of the grids which can be drawn-on and used again . . . ideal for overlaying and copying.

The 24 page full colour guide will show you how to use the JOTTER and FILM to produce perfect low resolution graphics on your ZX81!

So stop squiggling. Get a set now!

"WHERE CAN I GET THEM?"

Post now or call at one of our retailers!
Post to Print 'n' Plotter Products, 19 Borough High St., London SE1 9SE.

Name: ....

Address: ....

....

.... 5

ZX SPECTRUM JOTTER @ £9.95 each.
ZX SPECTRUM KEYBOARD OVERLAYS @ £2.95 per pack
ZX SPECTRUM DEMO CASSETTE @ 95p each.
ZX81 JOTTER PADS @ £3.50 each.
ZX81 FILMS @ £2.25 each.
ZX81 "GRAPHICS PROGRAMMING GUIDE" @ £1.50 each.
ZX PRINTER PAPER @ £10.95 per five rolls.

Remittance enclosed. Please bill my Access/Barclaycard/Visa/Mastercard No:

Please note: Price quoted include VAT, P&P for UK deliveries.
Overseas order please add 25% for additional Surface Mail.

Print'n'Plotter Products

You can see and buy most of our products at
W. H. Smith (Computer Stores)
Buffer Micro (Streatham)
Microware (Leicester)
Dennys Bookshops (London EC1)
Personal Computer Services (Darwen, Lancs)
Telford Electronics & Computing (Shifnall)
Georges Bookshop (Bristol)

SINCLAIR PROJECTS

FROM THE EDITOR

WELCOME to the third edition of *Sinclair Projects.* We are now having an increasing number of projects submitted each month, which means we can select a wider range of articles. It is becoming increasingly clear that the vast majority of readers have little or no background in electronics, and we have decided that a section of this issue shall be devoted to answering some of the problems which readers encountered in the first two issues.

The five main projects in this issue include an RS232 printer interface for the Spectrum, constructed on a Eurocard and Veroboard instead of the usual proper strip board. We have also included our first printed circuit board design. It was submitted by Ferranti and demonstrates the use of one of its new integrated circuits.

The PCB layout is given in the text but being a double-sided board, we have made arrangements for them to be purchased by prospective users. The design is the first of a series showing how you can use a ZX-81 to control things, for example central heating. We will be following-up these ideas in future issues.

A number of our readers work in education and industry and we have published an interesting project looking at how you can use a computer to simulate electronic circuits. It is an introduction to computer-aided design techniques which are used constantly in industry. We hope it will stimulate further interest in this area.

We have also entered the field of aids for the disabled. Our simple project using easily-available components allows severely handicapped people to communicate using the relatively cheap facilities of the ZX-81 and the ZX printer.

As additional help for the many people who like to keep up-to-date with what is happening on the hardware side of the Sinclair market, we include a number of pages of news, as well as book reviews to assist in the understanding of what happens in the small black boxes from Sinclair Research.

Finally, as the start of an extended tutorial on how the ZX-81 works, we have the first of a series of articles, dealing with how you address the various bits of memory in the computer.

At the rear of the magazine is a series of guidelines to be used when submitting articles. No doubt the use of British standard logic symbols will cause discussion and any comments would be welcome.

Editor Nigel Clark **Consultant editor** Robin Bradbeer **Technical assessors** Dave Buckley, Bazyle Butcher **Production editor** Harold Mayes MBE **News writer** Stephen Adams **Design** Elaine Bishop and Bill Scolding **Editorial director** John Sterlicchi **Advertisement manager** John Ross **Advertisement Executive** Annette Burrows **Editorial/production assistant** Margaret Hawkins **Managing director** Terry Cartwright **Chairman** Richard Hease

Sinclair Projects is published bi-monthly by ECC Publications Ltd. It is in no way connected with Sinclair Research Ltd.

Telephone, all departments: 01-359 7481. If you would like to contribute to any of the *Sinclair User* group of publications please send programs, articles or ideas for hardware projects to Sinclair User and Projects, ECC Publications, 30-31 Islington Green, London N1 8BJ. We pay £50 per 1,000 words for each article used.

trum

ZX Spectrum software on cassettes – available now

The first 21 software cassettes are now available directly from Sinclair. Produced by ICL and Psion, subjects include games, education, and business/household management. Galactic Invasion…Flight Simulation…Chess…History…Inventions…VU-CALC…VU-3D …47 programs in all. There's something for everyone, and they all make full use of the Spectrum's colour, sound and graphics capabilities. You'll receive a detailed catalogue with your Spectrum.

RS232/network interface board

This interface, available later this year, will enable you to connect your ZX Spectrum to a whole host of printers, terminals and other computers.

The potential is enormous. And the astonishingly low price of only £20 is possible only because the operating systems are already designed into the ROM.

sinclair

Sinclair Research Ltd, Stanhope Road, Camberley, Surrey GU15 3PS.
Tel: Camberley (0276) 685311.

The ZX Printer – available now

Designed exclusively for use with the Sinclair ZX range of computers, the printer offers ZX Spectrum owners the full ASCII character set – including lower-case characters and high-resolution graphics.

A special feature is COPY which prints out exactly what is on the whole TV screen without the need for further instructions. Printing speed is 50 characters per second, with 32 characters per line and 9 lines per vertical inch.

The ZX Printer connects to the rear of your ZX Spectrum. A roll of paper (65ft long and 4in wide) is supplied, along with full instructions. Further supplies of paper are available in packs of five rolls.

The ZX Microdrive – coming soon

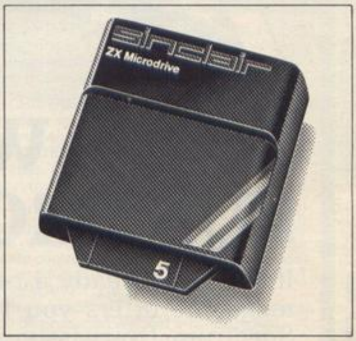
The new Microdrives, designed especially for the ZX Spectrum, are set to change the face of personal computing.

Each Microdrive is capable of holding up to 100K bytes using a single interchangeable microfloppy.

The transfer rate is 16K bytes per second, with average access time of 3.5 seconds. And you'll be able to connect up to 8 ZX Microdrives to your ZX Spectrum.

All the BASIC commands required for the Microdrives are included on the Spectrum.

A remarkable breakthrough at a remarkable price. The Microdrives are available later this year, for around £50.



How to order your ZX Spectrum

BY PHONE – Access, Barclaycard or Trustcard holders can call 01-200 0200 for personal attention 24 hours a day, every day. BY FREEPOST – use the no-stamp needed coupon below. You can pay by cheque, postal order, Access, Barclaycard or Trustcard.

EITHER WAY – please allow up to 28 days for delivery. And there's a 14-day money-back option, of course. We want you to be satisfied beyond doubt – and we have no doubt that you will be.

To: Sinclair Research, FREEPOST, Camberley, Surrey, GU15 3BR. **Order**

Qty	Item	Code	Item Price £	Total £
	Sinclair ZX Spectrum – 16K RAM version	100	125.00	
	Sinclair ZX Spectrum – 48K RAM version	101	175.00	
	Sinclair ZX Printer	27	59.95	
	Printer paper (pack of 5 rolls)	16	11.95	
	Postage and packing: orders under £100	28	2.95	
	orders over £100	29	4.95	
			Total £	

Please tick if you require a VAT receipt ☐

*I enclose a cheque/postal order payable to Sinclair Research Ltd for £ ______

*Please charge to my Access/Barclaycard/Trustcard account no.

*Please delete/complete as applicable

Signature

PLEASE PRINT

Name: Mr/Mrs/Miss

Address

MEG812

FREEPOST – no stamp needed. Prices apply to UK only. Export prices on application.

Sinclair Programs

TWO GREAT WAYS TO GET MORE FUN FROM YOUR SINCLAIR

If you are already a reader of *Sinclair Programs* you will appreciate that this magazine offers you fantastic new ways of enjoying your Sinclair ZX-81 or Spectrum. Every issue contains 40 brand new ready-to-key programs. And every one has been tested by our staff of experts — so you can be sure they will work.

Now you can get double the fun. Because from May onwards *Sinclair Programs* will be published monthly. Demand for our bi-monthly editions has been so strong that we have responded to our readers' demands by doubling the frequency. Still the same number of quality programs. Still only 95p per copy.

Where else can you find such value?

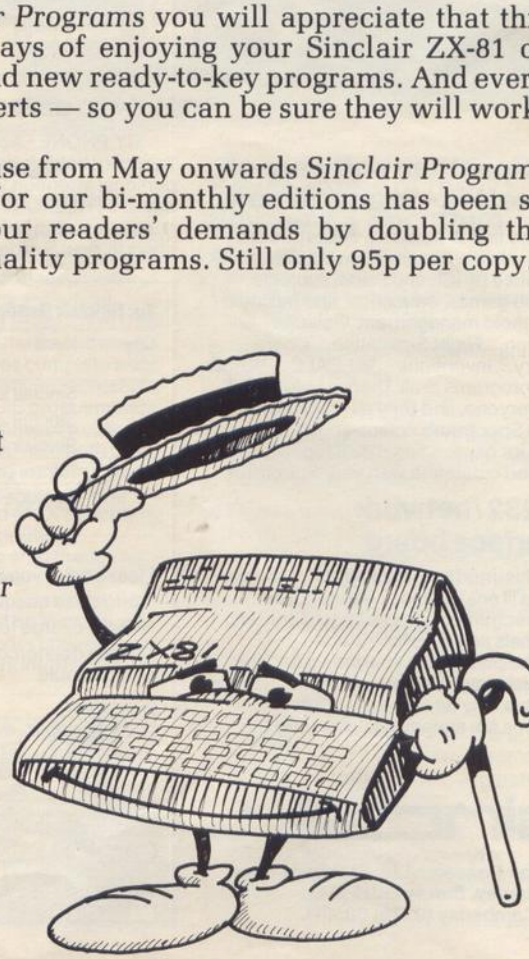
From us!

Because we're also publishing **The Best of Sinclair Programs**

To be available in time for Easter, this will contain **the 100 most popular programs** we have published in *Sinclair Programs* in the last year.
And all for just £1.95.

Make sure you get the monthly *Sinclair Programs* and *The Best of Sinclair Programs* by placing your order *now*. Both will be available from all leading newsagents or by post from ECC Publications Ltd., 30-31 Islington Green, London N1 8BJ. Prices: *Sinclair Programs* 95p per copy (plus 35p P & P); subscriptions — £13.20 for 12 issues including P & P. *The Best of Sinclair Programs* £1.95 per copy (plus 50p P & P). Cheques/POs with orders please, payable to ECC Publications Ltd.

NB. These rates apply to U.K. readers only. Overseas rates available on request.



Commercial RTTY permission needed

After our first two issues your letters are arriving regularly, giving your comments on the magazine and the world of Sinclair. We answer some queries while others are dealt with in depth in Advice Corner.

CONCERNING the article in *Sinclair Projects*—February/March—on RTTY which was of particular interest to me as a licensed amateur. I would like to point out that if any of your readers are interested in copying commercial RTTY, they might like to know that, in the last few years, there has been a number of successful prosecutions.

Neither the broadcast receiving licence nor the U.K. amateur licence permits the reception of these transmissions, but the Home Office Radio Regulatory Department has indicated that permission might be granted if it is first obtained from the various press agencies. To be safe, try the following frequencies in the amateur bands:

80m	3.60Mhz ± 20Khz
40m	7.04Mhz ± 5Khz
20m	14.09Mhz ± 10Khz
15m	21.10Mhz ± 20Khz
10m	28.10Mhz ± 50Khz
4m	70.30Mhz
2m	144.60Mhz(FSK)
2m	145.30Mhz(AFSK)
70cm	432.60Mhz(FSK)
70cm	433.30Mhz(AFSK)

Your article refers only to Audio Frequency Shift Keying; also used is Frequency Shift Keying, which shifts the transmitted carrier frequency—170Hz amateur, 850Hz commercial—for each mark to space transmission.

Two errors also crept in. On page 12, the Murray code character in figure six under the numbers column 'none' should be '+'; page 13, the mark/space tones should be either 1.275KHz or 1.275Hz(1.445KHz–1.445Hz) and on page 14 some amateurs use 50 baud.

How about a similar article for the Spectrum? Many thanks for a very good magazine.

John Lawrence,
Totton,
Southampton.

Port query

THANK YOU for publishing my project and, now that it has come through the wash, I would like to make some comments which you may wish to print.

Towards the end of the text on the first page, $\overline{Q}_a$–$\overline{Q}_d$ is quoted twice but the second mention should be Q_a–Q_d.

In figure one, pin 8 of the 74LS93 is connected to pins 2 and 3 of this integrated circuit and pin 13 of the 74LS42. Pin 1 on the 74LS93 is connected to pin 12 of the 74LS42 only. C4 is a 0.1uF capacitor and R2 should be connected to ground at one end.

In figure six, where it says 74LS93, it should have read 'from ICI where ICI is the ULA inside the ZX-81'.

Figures two and three have probably succeeded in mystifying everybody but they should not have been there. When the original was built, the two spare gates in the package were used as Schmitt trigger oscillators with 22k potentiometers inserted between holes 1,5 and 1,6, 1,7 and 1,8. The outputs were fed into the keyboard input—KB3 & KB4—via the DPDT switch—the rectangle with a circle in it above the integrated circuit in figure two.

They were to be used as joysticks but the software needed to be sufficiently advanced. Therefore the idea did not progress very far but the more imaginative experimenters may like to develop it further.

Stephen Huckstepp,
Colchester.

Business use

I AM very impressed with the content of your magazine and congratulate you on filling the need for Sinclair owners who like to delve into the electronic aspects of their hobby.

I use a ZX-81 as a business machine for calculating heat losses and unit costs for bills of quantities. A program on which I am working will help small businessmen to obtain more information from balance sheets and another will proceed logically through the various options to display the probable reason why a domestic heating system has stopped or is not working correctly.

The Money Maker program by David Nowotnik in the February/March issue is of great interest but after spending last night loading the program I have concluded that a number of lines are missing. Should there be a line 760 and 5050? There may be more.

John Burgess,
Lichfield,
Staffs.

● *You are correct; two lines are missing:*
5030 PRINT AT TI,2R:D8
5040 RETURN
5050 as 5030 etc
Line 760 does not exist.

Differences

I UNKNOWINGLY missed the first issue of *Sinclair Projects* so I missed the graphic board for the ZX-81 which I would like to have had. Now that I have the second issue I am very pleased at what it offers. It is different from all the others because it shows the circuits you can make.

M Ward,
Camberley, Surrey.

Radio blocks

I READ with interest the article on the ZX-81 radio teleprinter in the February/March issue. The author mentions that a terminal unit must be used and mentions two circuits in various magazines. I do not understand the significance of the blocks marked in dashed lines with pin numbers beside them.

Apart from this I find your magazine very enjoyable; it is certainly a refreshing change from software-based magazines.

R L Brooks,
Bishop's Stortford,
Herts.

● *The dotted blocks are for optional buffer clips to be used if your ZX-81 does not like being coupled directly to the 8251. If you*

LETTERS

read the text you will find that reference is made to two articles in other magazines which show you how to build a terminal unit.

Basic logic

I WRITE with regard to an article in the February/March issue, Operating logically to improve Basic.

There are machine code routines mentioned for typing-in but no listings are given at all. As I buy magazines to use the articles contained therein, this has somewhat perplexed me.

It is a pity that, in common with many of your competitors, the magazine is not completing its articles to make them technically viable, as many of the articles are very useful.

S F Wiltshire,
Bournemouth.

● *Unfortunately you are correct. The missing line of machine code is:*
0, 0, 38, 64, 46, 130, 136, 35, 166, 119, 201.

Graphics

IN THE December/January issue of *Sinclair Projects* you have the instructions on how to make a graphics generator. My 13-year-old son saved his pocket money, bought the parts needed, and started to make one.

He got on well, until he came to connect the +5V and the OV to pins 1B and 4B of the edge connector. There is a +5V shown in the diagram on page 30 but no OV and no written word to tell him where it is supposed to be on the graphics board, and where it has to be connected to the computer.

As there are four technical assessors to help, according to your magazine, my son telephoned before Christmas to solve his problem. He was told by the respondent that he was not qualified to help him. The man then put down the telephone, leaving my son's problem unanswered and with no advice as to how to obtain an answer.

When I telephoned I was connected to a woman, who said after a few minutes that she could not help, but took my telephone number and said she would contact someone who could help to return my call.

I do not know where your technical assessors were or why my son was not given help, but surely someone on your staff knows something about computers and could have offered help.

Mrs H E Ross,
Gillingham, Kent.

● *Issue two contains a component layout for the circuit of the character generator. Unfortunately it, too, contains errors.*

The OV and 5V to which you refer are the power supplies to the integrated circuits. The OV is connected to pin 7 of each IC and the 5V is connected to pin 14 of each IC. We have sent you a correct drawing and hope this enables your son to complete the project.

Interface

I REFER to the February/March edition and the article on the RS232 interface. A friend who is to make the interface requires the following information before continuing:

The 12F capacitor shown in figure two (top of PCB) is not shown on figure three (circuit diagram). Where on the diagram should it be?

I do not know where to obtain BZX51 diodes; is the number correct? If the number is correct could you advise an equivalent?

R L Hartley,
Enfield, Middlesex.

● *The answer to problem one is that the capacitor is across the power supply rails and therefore not on the circuit diagram. Problem two is answered in the letter from David Dunn.*

Good joystick

I WOULD like to congratulate you on *Sinclair Projects*. At last somebody with limited electronics knowledge can build useful hardware for a computer.

I built the joystick controller which works very well. Would it be possible for you to publish a project to add a fire button to the controller?

K J Gouldstone,
Wallington,
Surrey.

Better artwork

I FEEL I must write to congratulate you on publishing *Sinclair Projects*. You have certainly filled a vacuum for people like myself who wish to uprate the capabilities of the ZX-81 with projects at a reasonable cost.

After initial teething problems, I am certain the magazine will establish itself among the more serious users and the technically-inclined.

I could easily afford to buy the latest developed home system but I find it far more educational and satisfying to improve the ZX-81 and I am sure there must be thousands like me who wish to approach computing in this way. Here's wishing you every success.

There is only one thing which I believe detracts from *Sinclair Projects* and that is the circuit schematics.

I am, by profession, a printed circuit design engineer for a large telecomms company and I am afraid that the schematics in the first two issues never gave me sufficient confidence to want to try building any of the projects. I shall build the I/O port once I have laundered the schematics.

Without being disrespectful to your draughtsmen, I feel they have very little, if any, electronics experience and that the magazine would benefit greatly if the artwork was of a higher standard.

Meanwhile, thank you for an interesting magazine and I hope you have the success you deserve.

J V Salmon,
London N18.

I WAS prepared to tolerate the errors in issue one but what bothered me more, if anything, was the presentation of the circuits.

The Rotring drawing instrument—microprocessor-controlled draughtsman's aid—can be rented, I think, from an office equipment firm. **M Winibury,**
Coventry.

RS 232 diode

IN THE February/March edition, page 37 gives a PCB which contains reference to a diode BZX 51. We have searched electronics catalogues and made enquiries from various sources of supply and can find no trace of such a diode. Is there in existence a BZX 61? **David Dunn,**
Watford, Herts.

● *We are afraid we made an error and, according to the author, the diode which should be used is the Zener BZX 61 7.5V.*

NEWS

Battery RAM from Hunter

HUNTER PRODUCTS, an American firm, has produced a battery-backed RAM board which can be moved about in the memory of a ZX-81 to any 8K section. The board is available as a kit and is very easy to build from the detailed instructions.

The lithium battery included means that the 6116 RAM chips keep the program even when you switch off the machine, just like a ROM. That means that machine code routines and even Basic can be stored in the on-board RAM, to be recalled at any time without having to load them from tape.

A simple 2K RAM is provided. ROMs and RAM can be mixed on the board but that involves some cutting of tracks. One problem is that the board cannot decode the last address line A15, so that another chip—not provided—has to be soldered on to the original if you use more than 16K of memory from Basic.

One advantage is that you can replace the 8K ROM with a battery-backed copy and then add your own commands or correct the mistakes in the ROM.

The kit costs £19.95 inc. VAT from Hunter Electronics, PO Box 5, Axminster, Devon EX13 5AS. That is a postal address as the company is in the U.S.

Contoured RAM pack

CHEETAH MARKETING has produced a 16K RAM pack for the ZX-81 which stops any RAM pack wobble without using glue, tape or screws. The front has been contoured to fit the curved back of the ZX-81 and fits it like a glove.

Vigorous tests were carried out, such as thumping the keyboard heavily, but it refused to crash. The cost is £19.75 inc VAT and postage.

Cheetah also makes a 64K RAM pack for £44.75. Printer owners may be pleased to know that it also works well with the Sinclair printer.

Cheetah Marketing is at 359 Strand, London WC2. Tel: 01-240 7939.

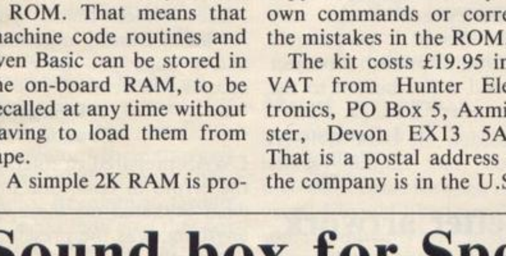
Sound box for Spectrum

BI-PAK has now made the ZON X-81 sound box available for the Spectrum as well as the ZX-81. It contains an 8912 three-channel sound chip with a powerful amplifier, loudspeaker and volume control in a 6in. × 3in. × 2in. black box. Everything is powered from the computer and some complicated sounds can be generated which will repeat without the aid of the computer.

As the device is I/O-mapped it requires machine code on the ZX-81 but an ingenious method of making it compatible with other REM-located machine code makes it easy. The Spectrum must use an additional board which contains a 3.5MHz oscillator to stabilise the frequency used. That is about 2in. long and fits between the Spectrum and the ZON 81.

One problem with the decoding makes it incompatable with the Microdrive and RS232 intended for the Spectrum as it uses A4, which Sinclair has already declared will be used by its devices. Apart from that it works well on both machines, providing a cheap and easy alternative to the BEEP command.

Bi-Pak sells the ZON X for £25.95 and the Spectrum board for £6.80. Bi-Pak is at PO Box 6, 63a High Street, Ware, Herts. Tel: 0920 3182.

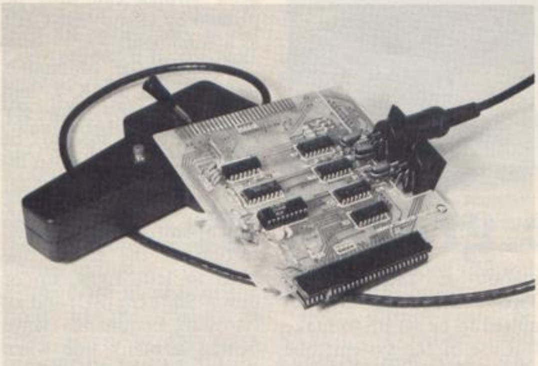


Reading is made easier

BRANDON PRODUCTS stocks the easy reader computer stand for holding magazines and books. The metal stand is very stable and tilts back a magazine at an angle for easy viewing. The magazine is held by a large clip and there are extension arms on both top and right-hand side for large pages. It is very useful for copying-in programs; also it folds flat for storage.

Contact Brandon Products, Forest Home, Normansand, Salisbury, Wilts SP5 2BN. Tel:079439-336. Cost is £14.95 inc. VAT and postage. Brandon also has a range of T shirts and sweat shirts with ZX-81 and Spectrum logos for £3.25 and £6.99 respectively.

Joystick fits both Sinclairs



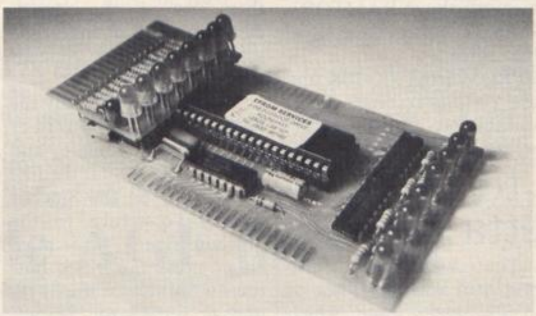
MIDWICH has produced an analogue joystick board which fits both the ZX-81 and the Spectrum. It can take two joysticks by plugging them into two six-way DIN sockets on the side of the bare board. It is input/output-mapped and requires the use of a small section of machine code on the ZX-81.

The board uses a ZN499E A/D converter and a switch to select which direction to measure. The results vary from 0 to 255 for each direction on the joystick—N/S and E/W—and the "FIRE" button operates one bit each of a second port, one giving 0 or 128 and the other 0 or 64; the higher number indicates that the button is pressed. Both pressed gives 128=64 (191).

Inserting the power plug is difficult, as not sufficient space has been left between that and the joystick sockets on the Spectrum version.

Midwich sells the board as a kit for £17.20 and the assembled version for £22.95. The joysticks cost £4.50 uncased with fire button. The kits are well-designed with good instructions. Midwich Computer Co Ltd, Rickinghall House, Richinghall, Suffolk 1P22 1HH. Tel: 0379-898571.

32-way port built by Eprom



EPROM SERVICES produces a 32-way port for the ZX-81 or Spectrum which has LEDs to indicate the state of its outputs. The device is based on an 8255 chip which is I/O-mapped to give 24 lines. One of the outputs is then fed to a set of eight flip-flops which produce the other eight lines.

A flip-flop gives two outputs which are always the opposite of each other, so if one is 0 the other is 1. That allows you to drive more devices but does not give more than 24 controllable two-state lines.

One of the facilities provided is for using the port to select other devices. The EPROM boards by the same firm can be selected in turn by this method, called bank switching. All that is required is to OUTPUT the correct number for the bank you require. An extra port testing board can also be added to check the other inputs or outputs for £6.

The PCB has no edge connector, so it has to be used with a motherboard or back-to-back connector.

Eprom Services is at 3 Wedgewood Drive, Roundhay, Leeds LS8 1EF. It charges £15 for the port and £24 for a motherboard for the Spectrum.

DCP puts packs into one box

DCP INTERSPEC pack is a combination of several other packs produced by the same firm into one box which fits the ZX-81 or the Spectrum.

It contains an 8-bit input and output port, four relay-isolated outputs rated at 24V-1 amp, four switch inputs—for joysticks, thermostats—and an analogue-to-digital converter with eight channels. That is an appreciable amount in a box only 4½in. long by 3in. wide and 2in. tall.

The Interspec has an expansion bus at the back which provides the data, read and write control signals, plus two device-select pins to add to your own interfaces.

All are fully-decoded, to the limits allowed on eight address lines, and work very well. The output connectors are well-documented and are standard school 2mm. sockets and 0.1mm. Molex connectors.

The instructions also give examples of what devices can be used with the Interspec. The cost of the unit is £39.95 inc. VAT and postage.

One thing about it is that it will not allow the use of any expansion on the ZX-81, including the Sinclair RAM pack, without it having either a PCB on the back of the RAM pack or a mother board. DCP Micro Developments Ltd, 2 Station Close, Lingwood, Norwich NR13 4RX. Tel: 0603-712482.

NEWS

Memotech range grows

MEMOTECH has been producing RAM packs for some time but also produces a keyboard for the ZX-81 and a hi-res pack. The keyboard does not require the user to dive into the machine to fit. Its interface module goes into the back of the ZX-81 like all the other modules and the keyboard is on a 7in. cable emerging from the end of the metal box. It provides only 40 keys—no space bar— and the keys tend to echo inside the box but it works well. It would be very useful to the non-technical user and does not invalidate the guarantee.

The hi-res pack—HRG—requires the use of at least a 16K RAM pack in which to store its hi-res pages. Depending on the size of memory, more than one 6,337-byte page can be stored. The pages must be stored outside the normal memory above RAMTOP.

The routines to control the screen are in a 2K EPROM from 8K-10K and use certain reserved Basic variables to transfer information to the routines. Twelve variables are required to be set up to make full use of the system; one of the Z$ is used to contain one of 30 commands.

The screen must first be set up by allocating variable V to the start of the video page, Z$ to "STARCH" and calling the user routine via RAND USR 8912. That then sets up the screen, clears it and changes the screen from the Sinclair version.

You can then see only the hi-res screen, no INPUTs or commands are visible, and PRINT, PLOT and the like will appear on the Sinclair screen. To return to the Sinclair screen you must either press the black button on the side of the hi-res unit or set Z$ to "Basic" and call the user routine again.

All commands are given in that way through the same USR call. X and Y variables determine the plot position, X being in the range 0-247 and Y is 0-191. Lines can be drawn from point P/Q to X/Y using "LINE"; "UNLINE" unplots them. Drawing in black or white a vertical line from X/Y until a bit of the same colour is reached can be done by "BLINE" or "WLINE". No shading is provided. Characters can be defined in C$ and printed at X/Y; normal Sinclair character strings can also be printed by using "SINCH" as a command.

Four of the commands test the bits at X/Y and return a result by using LET A$\frac{3}{8}$USR 8192 instead of RAND USR 8192. Also included in the command set is RU RD SD SU SL SR which shift a pre-defined section of the screen one pixel to left, right, up or down. Shift Up (SU) and all the shift commands leave behind a blank line when they move, whereas Roll Up and Down will wrap around the section.

Using those commands can become complicated, as each change requires two or three lines of Basic and a USR call.

It is a useful device but one which is not easy to learn to use. Many variables have to be set and re-set during the program for it to work. Error codes are given but things like the scroll commands can crash the system if they are not used properly.

The casing is stylish and strong and Memotech includes Velcro strips to stick the packs to the ZX-81 to stabilise them. All the Memotech packs are moulded to fit the backs of the ZX-81 and each other.

The Memotech keyboard costs £49.95 and the HRG pack £39.90. They can be purchased from Memotech Ltd, Witney, Oxon OX8 6BX. Tel: 0993 2977, or from larger W H Smith branches.

Mindware to make second Rosetta award

THE SECOND annual Rosetta Stone Award will be presented sometime in April by Mindware Inc for outstanding intellectual achievement on a ZX machine. The award was presented last year to Dr Ian Logan for his work on the ZX-81 ROM. The award is worth $200 and one year's free subscription to an American computer journal of the winner's choice.

The winner is also presented with a replica of the Rosetta Stone which puzzled archeologists with its codes for years.

Nominations are being taken at present and the judging will be done by a panel of representatives of American and British magazines. Nominations should be sent as soon as possible to Mindware Inc, 15 Tech Circle, Natick, MA 01760, U.S.A.

Extra high-resolution ZX-81 graphics

A HIGH-RESOLUTION graphics pack designed to work on a ZX-81 with at least 8K of memory is the G007 produced by the Nottingdale Technology Centre. By extending certain commands in Basic such as SLOW, FAST, CLS and PLOT, it gives the ZX-81 a 256–192 dot screen—it is slightly bigger but to keep the calculations simple it is limited to that.

The extra number at the front of the commands gives access to the extra commands, i.e., CLS 3 inverts the whole screen without clearing it. The routines available allow the plotting of individual pixels on the screen, drawing lines, triangles and parts of circles, as well as shading them.

PRINTing may be done to any pixel on the screen, as the PRINTing corresponds to the graphics pointer. That enables you to print a character, number or letter at any place on the hi-res screen. The SLOW command determines whether it is printed in black or white and what the background will be.

The use of the hi-res screen is enhanced further by the fact that the edit line is constantly on the screen, so INPUT and commands can be given without having to switch backwards and forwards between screens. Both the Sinclair screen and the hi-res version are separate and so clearing one will not effect the other.

Both screens can be SAVEd, LOADed and COPYed from Basic. The hi-res screen is stored as the last program line—line G007—and can be deleted only by a USR routine. The Sinclair Basic will not allow you to SAVE it as a named program, however, so "LOAD" must be used to get it back.

The program is stored in a 2K (2716) EPROM inside the unit and uses the internal memory of the ZX-81—1K or 2K—to store its system variables. All of those are described in the manual with the unit, so that the machine code programmer can also use them. There are three example programs, only one of which failed to work.

The device goes directly on to the back of the ZX-81, as it has to decode the internal memory into using the 8K–12K section of memory on the ZX-81.

That means it could be incompatible with some ZX-81 RAM packs or addons. Using it is very easy, as one soon becomes used to adding the extra numbers—they are all fully syntax-checked, too—and the display is rock steady.

Perhaps the next project for the group will be a colour board, as it would make it a very cheap BBC machine, but using only 6K of memory for the screen.

To order the device, send £37.55 to Nottingdale Technology Centre Ltd, TFL (ZX-81), 189 Freston Road, London W10 6TH. Tel: 01-969 8942.

Versatile AGF joystick

AGF HARDWARE has produced a joystick interface which fits on the expansion port of a ZX-81 or Spectrum and imitates the keyboard. Both versions simulate the cursor keys and the 0 key on joystick one and keys T-Y-U-I-P on joystick two.

There is no soldering to do as the board is plugged into the expansion port and any other devices plug in after it. The joystick is very easy to use and it speeds games enormously. The other advantage is that while it is enabled—via a switch—the rest of the keyboard is inoperable, thus preventing errors. Several programs already work with the joysticks, though some require the disabling of the joystick while speeds are chosen. The only disadvantage is the cheap edge connector and the fact that you cannot select which keys are simulated—that would allow an arcade game to be played.

Joystick interface costs £15.96, Joystick £7.54 inc. VAT, from AGF Hardware, 26 Van Gogh Place, Bognor Regis, West Sussex PO22 9BY. Interfaces are sent with a free cassette of programs and instructions how to alter your own.

Analogue port for the ZX-81

AN ANALOGUE port from University Computers is a memory-mapped device and is designed to be used by the ZX-81 Basic instructions PEEK and POKE.

It is decoded into 64 locations just below the start of Basic RAM—15488-15552.

The first eight locations are used as an eight-channel analogue input port and digital eight-bit output port. That is an unusual combination and, together with the manual, it can provide a complete demonstration unit for schools.

The board is despatched without a box but with an expansion connector at the back for plugging-in the 16K RAM pack.

Three LEDs are also provided on the board to test the digital port, as is a light sensor to check the analogue input. There is also an amplifier so that voltages in the millivolt range can be measured.

A full circuit diagram is provided and the manual takes you through a very easy course on how to use the port.

The same firm can also supply various sensors for heat, light, sound and gas to fit on to the board. It can also supply various relays up to 25 amp 250V for control applications.

A very good teaching tool at £39.95 inc. VAT and postage from University Computers, 5 St. Barnabus Road, Cambridge.

NEWS

More routines from Eprom

EPROM SERVICES has produced a Spectrum version of its EPROM board for the 16K Spectrum. It is in the top 8K of memory from 56K to 64K which is normally empty. It consists of four sockets on a board, to which can be added static RAMs—6116 type—or 2K EPROM, 2716 +5V type.

As yet there are no EPROMs available for the Spectrum but some will soon be available.

EPROM I contains 10 small routines for program editing such as block delete, delete up/down from a specified line re-numbering.

Only the lines, no GOTOs or GOSUBs, and various checks on the memory, such as program length.

The next 12 routines are for entering machine code into REM statements, in hex only, and extending and modifying both the machine code and the REM statement. There is also a very useful hex dump facility to view an area of memory which gives eight columns of hex addresses and the data in them.

Sixteen more routines contained in the same EPROM deal with the screen display and together with a 2K RAM stored at 14K-16K on the same board allow you to store machine code or TV screens in non-Basic RAM to be recalled later. That makes a total of 38 routines in one 2K EPROM.

EPROM II concerns tape routines, including a RAPID load and SAVE. The other routines allow you to verify and identify a Sinclair SAVEd program.

EPROM IV contains a very good ZX-81 monitor program. It requires also that the board be fitted with 2K of RAM at 14K-16K to store the monitor screen stack and scratchpad. All the system variables are saved and the various Z-80A registers can be set before entering a machine code routine.

Two more useful instructions include a duplicate of the Spectrum instructions IN and OUT for getting data transferred via the input/output ports. Machine code can be saved and loaded from the monitor under the name "M". The monitor will also work on the minimum 1K basic ZX-81 and does not need a 16K RAM pack.

Breakpoints and the dumping of the Z-80A registers is done in the 2K RAM, so that a system crash does not result in the loss of the code—a re-set key must be used, not pulling-out the plug.

The Spectrum EPROM board also has the facility to be switched out of the memory map by input/output port. It has a blank 2K EPROM fitted—EPROM I—and costs £19. EPROMs II to IV cost £10 each and are available only for the ZX-81 at present. Eprom Services is at 3 Wedgewood Drive, Roundhay, Leeds LS8 1EF. Tel: 0532 667183.

Switch-type joystick

KEMPSTON (Micro) Electronics has a joystick of the switch type for the ZX-81 or Spectrum. It is a Competition-Pro joystick and is input/output-mapped by A6 being low. The joystick has a solid metal shaft and is spring-loaded with a black balled top. The two red fire buttons are about one inch in diameter and both produce the same result. The joystick can be in eight positions as the leaf springs used will both operate if moved in a diagonal direction. The joystick gives 8 for down, 4 for up, 1 for left and 2 for right. So up/right will give 6 and the joystick fire buttons both give 16.

The interface is "dead-ended", so it must go on after any other equipment, but it is attached to the joystick by 5ft. of cable, so that should not be a problem. Software is always important to an item like this and Kempston already has four programs which can use the joystick, from Abbex, Quicksilva and Softek.

Gulpman has also been converted for use with the joystick. When you buy the joystick you also get the listings of three programs provided free—one Space Invaders, a hi-res drawing program and a program to mix with the Bat and Ball game provided on the Horizon tape.

The joystick costs £25 inc. VAT from Kempston Electronics, Bedford Road, Kempston, Bedford MK42 8B2. Tel: 0234 852997.

Modulated sound for Sinclairs

TELESOUND 84 is the latest version of the sound modulator for Spectrum, ZX-81 and the BBC computer. There is a reduction in size to $\frac{3}{4}$ in. × $\frac{1}{2}$ in. × $\frac{1}{2}$ in. and the length of the leads which connect it to the computer is much-improved. The sound quality is also better and because of the connection by clips to the circuit it can be removed easily.

The Spectrum BEEP is then amplified by the TV set and is loud enough for anyone. The cost of the Telesound 84 is £9.95 from Compusound, 32 Langley Close, Redditch, Worcs B98 0ET. Tel: 0527 21438.

NEWS

Good tracing

ONE OF the most outstanding achievements in the graphics field is the RD tracer, a device which allows you to draw across a piece of paper with a cross hair—like a rifle sight—and have the computer copy the same thing on to the screen. The three-part arm needs to be fixed to the board on which you are working by the sticky strip provided—or Blutack. A template is provided to allow you to line up this and the graph-type map over which the arm works.

The man is marked in pixels and PRINT positions, so that letters and numbers can be added to the drawing. The interface black box, which contains an edge connector and PCB continuing strip, plugs into the expansion port of the ZX-81 or Spectrum. As it fits both computers it has only a 23-way edge connector and must therefore be used in the same way as the printer—placed after any Spectrum equipment.

On the ZX-81 it is memory-mapped into location 15391; on the 48K Spectrum is it input/output-mapped (port 31), but if required on the 16K Spectrum if can also be memory-mapped via an adaptor. LEDs are provided on the module to indicate when it is being used and also if it is being used on the input/output map.

The tracer tape has ZX-81 and Spectrum programs on the same side. That is satifactory for the Spectrum but ZX-81 owners should locate the ZX-81 program before attempting to LOAD the program.

The tracer has four programs for the Spectrum and two for the ZX-81. For the Spectrum the tracer can be used to create lines, circles or just to follow the trace of the arm in all the colours available. The drawing can then have areas "painted" in or shaded in the same or a different colour.

One thing to remember is that each character square can have only two colours. All the commands are available by one key-press and the entire program is written in Basic so that it can be customised easily.

The drawing can be scaled to give a 3D effect, with the program taking care of the perspective. There are amazing possibilities, as with user-definable graphics plus text and the tracer producing any other shapes all kinds of pictures can be created.

The other use of the tracer is to input data in the form of co-ordinates for graphs; they can be stored as strings and the tracing program made to execute the co-ordinates one after another.

The pictures can be saved on to tape from within the program and can also be loaded back for modification. The cost of the tracer is £55.50 with 10 percent discount for cash. It is available from RD Labs, 5 Kennedy Road, Dane End, Ware, Herts SG12 0LU. Tel: 0920 84380.

Games expansion unit

PROGRAM POWER has produced what seems to be an all-in-one module which contains joystick, soundboard and amplifier. The amplifier can also be used to boost the weak BEEP of the Spectrum up to a full two watts via the volume control. The board is not cased and fits ZX-81 or Spectrum, although the ZX-81 user will have to write some machine code to use it.

The placing of the board is the same as in the tracer, but the board does not have a continuing strip at the back, so place it last.

A demonstration tape is provided with every purchase and allows you via Basic to hear a selection of pre-set sounds and to be able easily to alter the sound registers one by one. The sound chip also contains an input and output port and they are used to run the joystick.

The joystick is not the switched type, however, and gives a position back in the form of two numbers X and Y. The fire button results in setting a variable to 1 if pressed and all the joystick commands are accessed by a machine code routine for greater speed.

At present you have to put together the joystick, as it arrives as a kit. The instructions are easy enough to follow but it requires soldering and the screws which hold the case together seem difficult to fit.

The system works very well and as it contains its own loudspeaker and is driven from the Spectrum or ZX-81 power supply, it should do very well.

Micro Power sells the sound board for £24.43 and the joystick for £6.53 inc. case or £4.55 without, all inc. VAT. Micro Power is at 8/8a Regent Street, Chapel Allerton, Leeds LS7 4PE. Tel. 0532 683186.

Auto repeat for ZX-81

AN AUTO KEY repeat key module is available for the ZX-81 which works in the same way as the Spectrum—1.5 sec. delay before repeating. It is soldered on to the ZX-81 via nine wires and rests inside the case underneath the keyboard.

It is not the same design as the Haven Hardware module.

TV Services of Cambridge makes the device and it is sold by that company and Kempston Electronics at £6.50 including postage. TV Services is at Chesterton Mill, Frenches Road, Cambridge CB4 3NP. Tel: 0223 358366.

Attempting to get good characters

Following a number of queries which have required more serious consideration we have given David Buckley more space in which to give advice and help you through some of the more difficult problems which have arisen.

A FEW PEOPLE have written about the character generator described in the first issue; some say it works well, others that to make it work they had to connect wires in a different manner, and still others who cannot make it work at all.

The character generator board shown in the photographs was built by *Sinclair Projects* and I am told it worked perfectly, although I never saw it. Unfortunately the particular ZX-81 and board have since been lost and it looks as if I shall have to build one to settle the arguments.

The purpose of the character generator board is to get the ULA to enable selectively either the ROM or the CHRS RAM and hence the signal from the board should look as if it is from the ULA. The instructions given in the article for connecting the ROMCS input and ROMCS output from the board do not seem to do that. To those who are having difficulty making the unit work, I suggest you try the following instructions.

Solder a thin piece of tinned copper wire over any track breaks you may have made on the ZX-81, except that in the RAMCS line by position ZA of the edge connector.

Make a break in the track from the ULA to R28 near to R28—that is the opposite side of R28 to the ROM—connect the ROM ROMCS line from the characters board to the end of R28 by the track break, connect the edge connector ROMCS line to the other side of the track. Connect all other lines as shown in the article.

That should solve the problems. I will build one to make sure and let you know the result in another issue.

The drawing on page 49 of issue two is not very clear and some of the connections seem to terminate between pins. If you use a pencil and ruler to draw vertical lines through the IC pins on the diagram, most connections lie nearest the line to which they should go but the following do not:

The connection from the 74LS08 pin 14 should go to pin 14 on both 74LS00s and so to +5V. All the pins 7 should be connected together and to OV. The right-hand end of the 10K resistor should go through an unbroken Vero track to the link to pin 14 of the 74LS00.

Now I will try and answer some specific points. Stanley McKeown of Belfast writes:

"I constructed the character generator board and I am delighted with this useful modification. I would like, however, to make the following points:

"The circuit would work only after connecting ROMCS input to the right-hand side of R28 and not pin 23B.

"The PCB layout of issue two, page 49 is inconsistent with the circuit shown in figure one, page 29, issue one and the photograph on page 32, issue one, although both constitute viable circuits.

"Paragraph one, page 49, issue two, Graphics generator adaptation for ZX-81 with 4118 RAM chip, omitted to state that CH$ RAMCS output should be connected to pin 20 of the RAM IC.

"My ZX-81 now seems to be more susceptible to white-outs; that appears to be caused by increased susceptibility to the spikes in the power supply". He suggests some ZX-81 topics for future issues, such as extra power supply to drive future projects; high-resolution graphics package, say 256×196 pixels, PLOT, DRAW, CIRCLE, based possibly on the graphics generator board of issue one or more extensive additional hardware and software; hardware/software to improve the loading accuracy and speed; the major disadvantage of the ZX-81.

If you follow the tracks on the issue one ZX-81 PCB the right-hand side of R28 which is connected to the ROM pin 20 is also connected directly to position 23B of the edge connector, so connecting ROMCS to either place should have the same effect.

If anyone has a high-resolution graphics package or any other project you think will be suitable, please write.

Will Epson 310 work with ZX-81?

"Could you please tell me if the Epson 310 printer could be interfaced with the ZX-81 using your interface in *Sinclair Projects* or any other interface?", asks Neil Martin. "Does it have some hidden snag?"

I have no information on the printer but I would expect it to be calculator-sized, and although it probably could be interfaced to the ZX-81 using the latch card, that may be difficult, needing extra hardware.

I suggest you write to the company advertising it and ask for details.

"I have a query regarding the EPROM blower. Once you have blown the chip can you leave the EPROM in the socket and use it as additional ROM or do you need an additional board?" asks Mark Sullivan of Canterbury.

Once the EPROM is blown, to use it as ROM it must be plugged into a memory expansion board. It cannot be used as ROM while in the EPROM blower. What would be possible, though, would be to use the 8255 to read the contents of the ROM and transfer them to a protected area of RAM.

ADVICE CORNER

Mark Webster, of Kings Heath, Birmingham is another reader with a query on the character generator. The information he needs is that it does not matter which address pin of a RAM is connected to which address line; whichever way they are connected each address will still select a specific byte position within the RAM.

CS and CE are equivalent. Regarding the connection of the ROMCS, see the earlier reply.

Non-standard chips on board

Tom Dawson of Enfield, Middlesex was one reader with non-standard chips. He says: "I was pleased to see your magazine on my local bookstall and I was impressed by the contents. Subsequently I spent £5.89 on the necessary parts for the graphic generator, only to find that my ZX-81 does not contain the 2114 RAM ICs but a simple 8117 (Mostek MK 4118N-4) RAM IC in their place. That chip is not in a socket but soldered directly to the board, making the bending-up of the pins impossible. If the only possibility is to buy two 2114 RAM ICs I will be happy to do so, especially if I can retain my internal 1K RAM as well.

"I also wish to point out that the OV connection to the logic circuit is not shown and a circuit diagram, even a Veroboard layout, would be very helpful on this and other projects, especially to a novice like myself."

He goes on to suggest that in a future issue we might publish a circuit diagram for the ZX-81—and the other less important Sinclairs—or, at least, a diagram of each chip showing the numbers or references of each terminal.

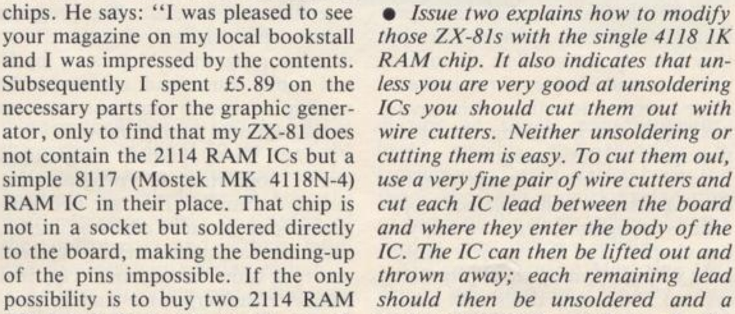
"That applies equally to your very helpful edge connector diagram. If ROMCS is pin 23B—presumably B stands for bottom, and A for top how can it be that there are only 22 pins each side? Is the slot pin 3? Finally, it is not possible for the graphics generator to work on the character codes 192 to 255? Those characters are accessible from the keyboard and yet are not used normally once a program is running. If they can be used there will be no loss of characters."

● *Issue two explains how to modify those ZX-81s with the single 4118 1K RAM chip. It also indicates that unless you are very good at unsoldering ICs you should cut them out with wire cutters. Neither unsoldering or cutting them is easy. To cut them out, use a very fine pair of wire cutters and cut each IC lead between the board and where they enter the body of the IC. The IC can then be lifted out and thrown away; each remaining lead should then be unsoldered and a socket soldered in place to take the new chip.*

Originally if you had one 4118 chip and you just cut the leads flush with the component side of the board you can fit two sockets for 2114s without having to unsolder all the remaining leads, and vice versa if originally you had two 2114 chips.

About the edge connector numbering; the slot is counted as a position and the edge connector diagram at the back of the magazine should help you find the correct location of the signals. Do not solder direct to the edge connector, as that will prevent you fitting the RAM pack; instead, solder to the tracks leading to the edge connector.

Once the board is working and initialised as set out in the article, you can alter any of the normal video characters, which includes the graphics characters 1 to 10. The words called-up by character codes 192 to 255 do not exist as such in the ROM and so it is not possible to alter them.



Getting power on the card

D Barnett of St Austell, Cornwall had problems with the latch card and power card projects in our first issue.

"I sent for the components and sat down to begin the construction. After a long time of trying to follow the circuit diagram I gave up both projects because of the number of mistakes.

● *The parts list for the latch card omitted a 20-pin 74LS373, which was also incorrectly labelled as a 74LS133 in figure five. Table four should have read*

Edge connector	*74LS133*
A15	*3*
A9	*11*
A8	*12*
A7	*13*
A6	*14*
A5	*15*

The circuit diagram was re-drawn from the original to conform to British Standards which the magazine has adopted for circuit diagrams.

The Vero tracks should run the way they do on the Veroboard, i.e., right to left in figure one, not as drawn in figure three, which was re-drawn in issue two together with the circuit diagram.

The wiring of SK1 and SK2 is shown correctly in figure two. The long link by SK1 should stop by pin 9 and not go beyond the top edge of the socket.

Simple introductions to using Sinclairs

Robin Bradbeer reviews three important books for people who wish to learn more about what to do with their computers and need careful well-written guides.

RANDLE HURLEY is well-known to readers of *Sinclair Projects* as the writer of the article which appeared in the first issue of the magazine on word processing on the Spectrum. *The Sinclair ZX-81,* Randle Hurley, Macmillan Press Ltd, 162 pages, contains a series of interesting programs.

None of the programs is really hardware-orientated; they are aimed at exploring the software capabilities of the system. To quote from the introduction:

"The person I had in mind when writing was not a computer expert. My target reader had finished all the material in the Sinclair manual and probably did not really understand those bits about Procrustean Assignment and Unary Functions. He or she will probably have read a book or two of games programs and perhaps will have been interested in some of the ideas presented in the pages of magazines devoted to the ZX-81.

"My reader may have shared my frustration that these ideas, while fascinating, are given in isolation and seldom related to any real task the ZX-81 may be called on to do. The person I am writing for may be a teacher like myself, hoping to use the machine for academic organisation or as an educational tool. On the other hand, I may be writing for someone interested in using the ZX-81 to help run a small business or a social club. All these possible needs have been borne in mind during the writing."

That quotation explains very clearly his aims and, to a considerable extent, he has more than succeeded. Unlike a number of other writers, he tries to introduce the concept of flowcharting at the beginning of the book. The early programs are really designed to overcome some of the more annoying habits of the ZX-81. Some interesting software is given for error-detecting routines, crash-proofing subroutines and even auto start-up for cassette saving.

Some interesting ideas are presented on storing numbers so that they take up the smallest amount of memory, and chapter three contains some very interesting time-saving techniques for increasing the execution speed of Basic programs.

After the introductory chapters there are seven well-explained applications programs, each occupying one chapter. The first project considers a very elementary word processing system. The second deals with money matters. As that normally means performing very simple routines many times over, a great deal of time is spent getting the program structure correct.

That leads to a personal finance program, which is claimed to run all your household accounts. Four options are considered — updating the balance; setting-up the system; changing standing orders; and filing the information. That allows six aspects of personal finances to be considered in detail — filing details of regular payments and the dates they fall due; remembering which cheque has to be deducted next; remembering which is the next paying-in slip to be added; deciding when salary cheques must be added and standing orders deducted; allowing for the possibility of special into and out of the account; and making provision for changes in standing orders.

Following the financial theme, Hurley simulates a bank for teaching purposes. At his school there is a bank for students. It is not only for the convenience of the users but is considered to be important for educational purposes.

Another chapter considers storing data in the system so that comprehensive files can be built. The program was written originally for the ZX-80 as a fun program, which ran a mythical rabbit farm and organised the reading to ensure maximum success at the National Bunny Show.

From those humble beginnings some sophisticated programming points are made. The last two programs look at more education tasks, such as examination results analysis and rank order calculations. The final chapter considers some simple hardware modifications to the ZX-81, so that it is able to withstand the hammering that is expected in the educational world.

For anybody interested in serious programming using Basic, the book contains some very interesting ideas. Although the programs are fairly specialised and orientated towards education, there is much to be learned.

By the guru of microprocessor applications, *Practical Microprocessor Systems,* Ian Sinclair, Newnes Microcomputer Books; 140 pages; covers a range of more popular microprocessors used for hobbyist-based systems.

He looks at the National Semiconductor INS-8060 (SC/MP II), the MOS Technology 6502 and the Zilog Z-80. The latter two microprocessors are very common, being used in computers such as the Pet, Apple, BBC micro and Oric, and the TRS-80, ZX-80 and 81, Spectrum and RML 380-Z respectively.

Although the book concentrates

more on the INS-8060 and the 6502 at the expense of the Z-80, most of the information is fairly processor-independent. Sinclair looks at how the microprocessor works, the action of the control data and address buses and elementary instruction codes.

Address decoding and addressing methods are considered in the second and third chapters. Again, although the material is written around a number of support chips for the 6502/6800 system, it is of sufficient general interest to be of use to any microprocessor.

Chapter four is a comprehensive review of buffers and interfaces and looks at one-state, two-state and three-state buffers, PIAs and input/output ports. Two minimal systems are considered in detail. One based on the Sinclair Research — no relation — precursor of the ZX-80, the MK-14 and the other on the Commodore precursor of Pet, the Kim 1.

Again, like the rest of the book the lessons learned in exploring the systems are very relevant to the current single-board computers such as the ZX-80 and ZX-81. The last chapter is a very clear introduction and explanation of assembly language programming.

The book is a good general introduction to simple microprocessor-based systems and therefore is recommended as a starter book for those who want a quick explanation of some of the more technical terms used in constructional articles.

The comprehensive *Mastering Machine Code on Your ZX-81* by Tony Baker, Interface; 180 pages; is not intended for the real beginner. Although Tim Hartnell's introduction indicates that it is easy to understand, you need to know your way around at least the elements of Z-80 machine code programming before it makes sense at all.

The introductory chapters deal with hexadecimal numbers, simple arithmetic and PEEKing and POKEing. Fortunately Baker gives a hex machine code loader on the first page of the second chapter, which makes life easy for entering the programs in the remainder of the book.

After a comprehensive look at the LD instruction, a number of short programs are considered for adding numbers and for moving information around the Z-80 registers. A section looks at the CARRY flag and that introduces the idea of subtraction.

The PEEKing and POKEing section explains clearly what happens when you try to use Basic to enter machine code directly into a memory location and that chapter also considers looping and loading registers directly with data. So far new and old ROMS are considered together and that is repeated throughout the book.

Consequently most of the machine code programs can be used on the ZX-80 as well as the ZX-81. A very clear explanation of the use of REM statements for storing information is also included in the fifth chapter.

One of the most powerful attributes of the Z-80 microprocessor is its stack operations. A chapter is devoted to that and playing with the stack pointer produces some very interesting programs.

The seventh chapter considers the ZX screen-handling facilities and the problems involved in using the rather peculiar method Sinclair designed for printing on the screen. A number of interesting programs are given in this chapter which really ends the introductory part of the book.

Then it becomes more a reference work, allowing you to dip into various sections whenever you require specific information.

The succeeding chapters are fairly self-contained. For example, chapter eight is a dictionary of all the Z-80 machine code extractions and chapter nine explains how to debug machine code programs. The operation of the keyboard is dealt with in chapter 10.

Three chapters deal with the development of a draughts game. It is clever, as Baker had obviously penned his thoughts as he developed the program, and that makes for fascinating reading.

Some of the other sample programs include obtaining sound from the television speaker and some very elementary high-resolution graphics. The final chapters look at disassembling the ROM and also a list of useful arithmetic subroutines.

It is a useful reference work for anyone interested in developing machine code programs for the ZX-81 and is recommended to those serious readers of *Sinclair Projects,* especially if they are having difficulty understanding what is happening when machine code is encountered.

Sinclair and calculus combine to aid computer design

As an introduction for ways in which the ZX-81 can help in design, Philip Lawton describes a way of predicting the waveforms which come from the tape recorder socket. We deal with the necessary electrical theory, the method of obtaining the equations, the program, and the results.

ONE OF the more novel uses of computers is to help to design other computers. As an introduction to the concept, I shall begin by showing how the ZX-81 can be used with calculus to predict waveforms and similar circuit analysis.

The signal simulated is that from the tape recorder socket of the ZX-81. The waveform of that signal, which is produced during SAVE, can be predicted using 16 lines of Basic running on the ZX-81 or another computer.

The ideas can be applied to many problems and are the foundations of modelling, simulation and systems theory. Equations involving rates of change have to be deduced and then solved numerically; that is easy when fundamental equations, laws, and a computer are used.

The ZX-81 output circuit to be analysed is shown in figure one; the required signal is V2.

As it is an electrical circuit problem, appropriate fundamental equations and laws are required. The equations must refer to the rate of change of the capacitor voltage with respect to time. They are stated in figure two.

In the following equations, the rates of change of capacitor voltage have been written as SV1 and SV2 respectively. Also the resistances of the logic output and the tape recorder input have been combined with R29 and R27 respectively.

Using the voltage law and the capacitor equation

E = R29*I + V1 + V2 and I = C12*SV1

by substituting for I and re-arranging for SV1 gives equation A

SV1 = (E − V1 − V2)/(R29*C12)

....

Using the current law and the capacitor equation

I = C12*SV1 = V2/R27 + C11*SV2

re-arranging for SV2 to get equation B

SV2 = (R27*C12*SV1 − V2)/(R27*C11)

....

Equation B could be amended by eliminating SV1 using equation A, but that is not necessary.

The program used to undertake the necessary calculus is shown in figure three. Lines 90 and 100 contain the equations for the rates of change of capacitor voltage, i.e., equations A and B.

Figure 1: ZX-81 output circuit for SAVE.

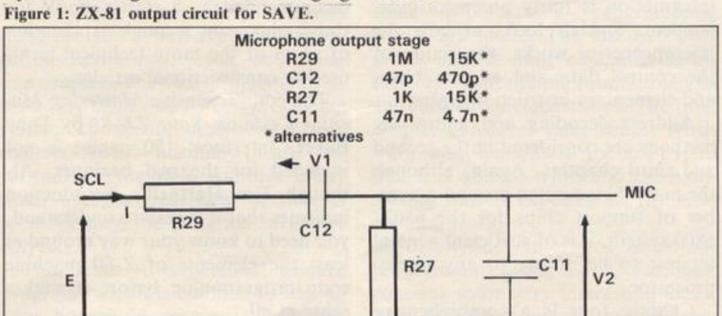
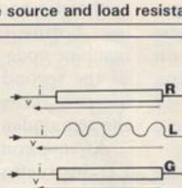


Figure 2: Electrical equations and laws.

The following four equations apply to electrical elements:

resistance	v = Ri	
inductance	v = Li′	(i′ means di/dt)
conductance	i = Gv	
capacitance	i = Cv′	(v′ means dv/dt)



The following two laws apply to electrical circuits:

the addition and subtraction of all the electrical potential differences in a closed mesh is ZERO;

the addition and subtraction of all the electrical currents at a junction is ZERO.

Also non-linear effects can be included.

Lines 110 and 120 undertake the calculus; they represent numerical integration using Euler's single step method.

To understand line 110, SV1 needs to be evaluated at T = 0. Using line 90 and the ZX-81 as a direct calculator, SV1 is 21277 volts per second.

Now assume that this rate of change remains constant for a very short time interval—represented by H. Then for H = 1E − 6 (line 30, one-millionth of a second) the change in voltage (line 110) will be approximately 0.02 volts. That change is summed with the predicted potential (line 110) and the sum becomes the next predicted potential.

The accuracy of the predictions depends on the selection of the time interval H; it has been selected to produce a small initial change (0.02 volts) compared to the final voltage (1 volt). More accurate and faster numerical integration routines are available and are used in the large simulation programs.

To predict the output voltage (V2) when the input (E) is a square wave of +2 to +4 volts the additional lines of code—figure three—should be incorporated in the program.

The results are shown in figure four and predict a peak output of ±0.6mV.

Figure 4: The results.

ZX-81 MIC output stage V2

Results for V2 when E = 1

Time (sec)	Volts
0	0
.00001	.00017
.00002	.00028
.00003	.00034
.00004	.00036
.00005	.00037
.00006	.00035
.00007	.00033
.00008	.00031
.00009	.00028
.00010	.00025
.00011	.00022
.00012	.00019
.00013	.00017
.00014	.00014
.00015	.00012

V2 when E = 4, 2, 4, 2, at 150μS intervals

0	0
.00005	.00074
.00010	.00050
.00015	.00025
.00020	−.00062
.00025	−.00045
.00030	−.00023
.00035	.00063
.00040	.00045
.00045	.00023
.00050	−.00063
.00055	−.00045
.00060	−.00023

Figure 3: Simulation program.

ZX-81 MIC output stage V2

Program for V2 when E = 1

```
10 REM ZX81 MIC OUTPUT 23/4/82
20 LET T=0
30 LET V1=0
40 LET V2=0
50 LET E=1
60 LET H=1E-6
70 FOR A=1 TO 15
80 FOR B=1 TO 10
90 LET SV1=(E-V1-V2)/47E-6
100 LET SV2=(47E-9*SV1-V2)/47E-6
110 LET V1=SV1*H+V1
120 LET V2=SV2*H+V2
130 LET T=T+1
140 NEXT B
150 PRINT T*H,V2
160 NEXT A
```

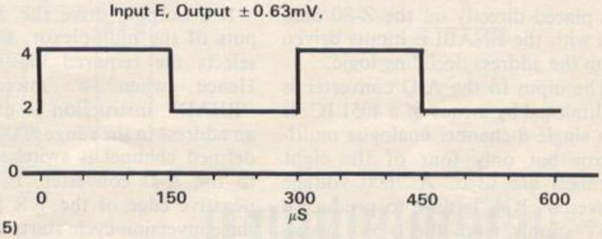
Additions for E = 4, 2, 4, 2, at 150μS intervals.

```
30 LET V1=2
42 LET F=-1
44 FOR C=1 TO 4
46 LET F=-1*F
50 LET E=3+F
145 SCROLL
170 NEXT C
```

Alternatives (SLOW or FAST ZX81 only)

```
145
150 PLOT INT (T/10+0.5), INT (V2*1E5/4+20+0.5)
```

Input E, Output ±0.63mV.



Improve your heating efficiency with ZX-81

In the first part of our two-part article on controlling your central heating system Owen Bird shows how to make a high-performance data aquisition interface. This is the most important part of the system. We will be showing how to connect the rest of the equipment in a future edition.

A SIMPLE, low-cost, high-performance data acquisition interface can be made for the ZX-81. A block diagram of the interface is shown in figure one. The interface connects directly to the expansion edge connector at the back of the ZX-81 and provides four input channels and two output channels to an accuracy of eight bits.

At the heart of the interface is a new IC from Ferranti Electronics, the ZN477. It is an 8-bit successive approximation A-D converter designed for easy interfacing to microprocessors. All active circuitry is contained on-chip, including a clock generator and a stable 2.5V reference.

A conversion cycle is initiated by the application of a negative-going pulse to the $\overline{WR}$ input; it can be completely asynchronous with the device clock signal. After a period of between 7.5 and 8.5 clock cycles from the end of the $\overline{WR}$ pulse, the data on the outputs of the ZN477 will be valid. Data can be read out by taking the $\overline{RD}$ input low, thus enabling the three stage outputs. Those outputs are connected directly to the data bus from the Z-80 microprocessor.

Two Ferranti ZN428s are used to produce the dual analogue output channels. The ZN428 is a fast 8-bit D-A converter with latched inputs, which are enabled by taking the $\overline{ENABLE}$ input LOW. The ZN428s are placed directly on the Z-80 data bus with the $\overline{ENABLE}$ inputs driven from the address decoding logic.

The input to the A-D converter is multiplexed by means of a 4051 IC. It is a single 8-channel analogue multiplexor but only four of the eight channels are used. A 7600 voltage converter, IC4, is used to produce a −5V supply from the +5V, necessary for both the multiplexor and A-D converter ICs.

ICs 6, 7 and 8 comprise the address-decoding logic — see figure four. They decode the unused ROM area between 8K-16K to produce the memory addresses for the system as defined in table one. Note that to keep the logic to a minimum, full address decoding is not used, nor is it necessary, hence table one will be repeated for every eight addresses in the address range 8K-16K, i.e., 8008-8015, 8016-8023 and the like.

IC7, which is a 3-8 line decoder/demultiplexor, generates the $\overline{WR}$ and $\overline{RD}$ pulses for the ZN447 and the $\overline{ENABLE}$ pulses for the ZN428s. The ZN447 $\overline{WR}$ signal is also inverted to produce the CLOCK signal for the dual D type latch, IC6. That latches the state of the two address lines A0, A1 on the negative edge of the ZN447 $\overline{WR}$ pulse.

IC6 outputs drive the control inputs of the multiplexor, IC5, which selects the required input channel. Hence when a microprocessor "READ" instruction is executed at an address in the range 8000-8003, the defined channel is switched through to the A-D converter, IC1, on the negative edge of the $\overline{WR}$ pulse, and the conversion cycle starts on the first negative clock edge after the positive edge of the $\overline{WR}$ pulse. The width of the $\overline{WR}$ pulse, approximately 1μs, allows the analogue switch to turn on prior to the converter starting a measurement.

To use the address space 8K-16K on the ZX-81 it is necessary to force the $\overline{ROM\ CS}$ pin on the ZX-81 connector to the high state to disable the ZX-81 on-chip ROM. That is accomplished by means of the NAND function formed with diodes D1-D3, gate IC8(b) and transistor Q1.

The nominal 0-2.56 voltage range (10mv/bit), of the A-D converter is used, without any pre-scaling or zero trim. The purpose is to allow standard joystick controls to be connected to the four input channels with no additional power supplies. Since the sweep angle of the standard joystick is only of the order of ±25 degrees of the potentiometer 270 degrees travel, then with ±5V across the potentiometer the input voltage to the A-D converter can be adjusted to be within its input range of 0-2.56V. The voltage swing from the joystick pot. will be about 2.0V.

Software calibration can then be used as part of the operating program to scale the inputs. If different input ranges are wanted, one of the simple resistor networks, shown in figures five or six, can be used in the circuit between the multiplexor and the A-D converter. Note that the internal reference of the ZN447 is used to supply the reference voltage to the ZN428s. That saves components and produces improved matching between converters. The basic output range of the ZN428s is nominally 0-2.56V with an output resistance of 4kohm. Suitable buffer amplifiers can be added externally if necessary.

Control of the interface can be performed in either Basic or machine code. With Basic it is a matter of using the PEEK and POKE commands directed at the appropriate hardware addresses as defined in

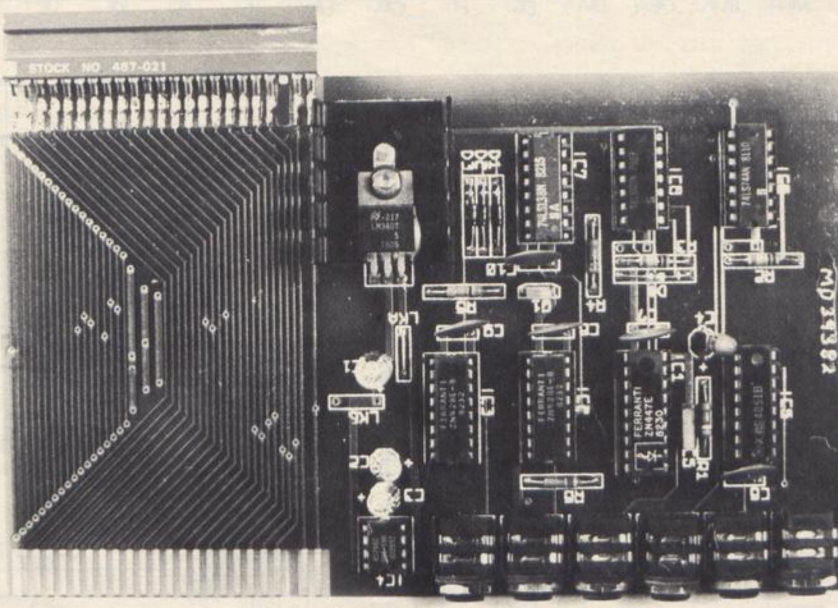
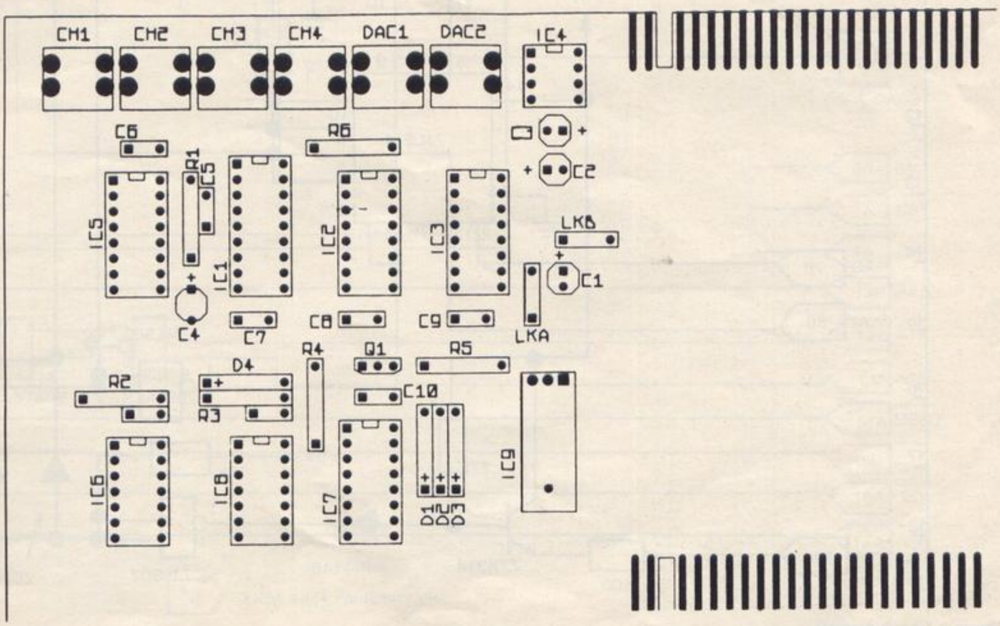


Figure 1: Component layout.



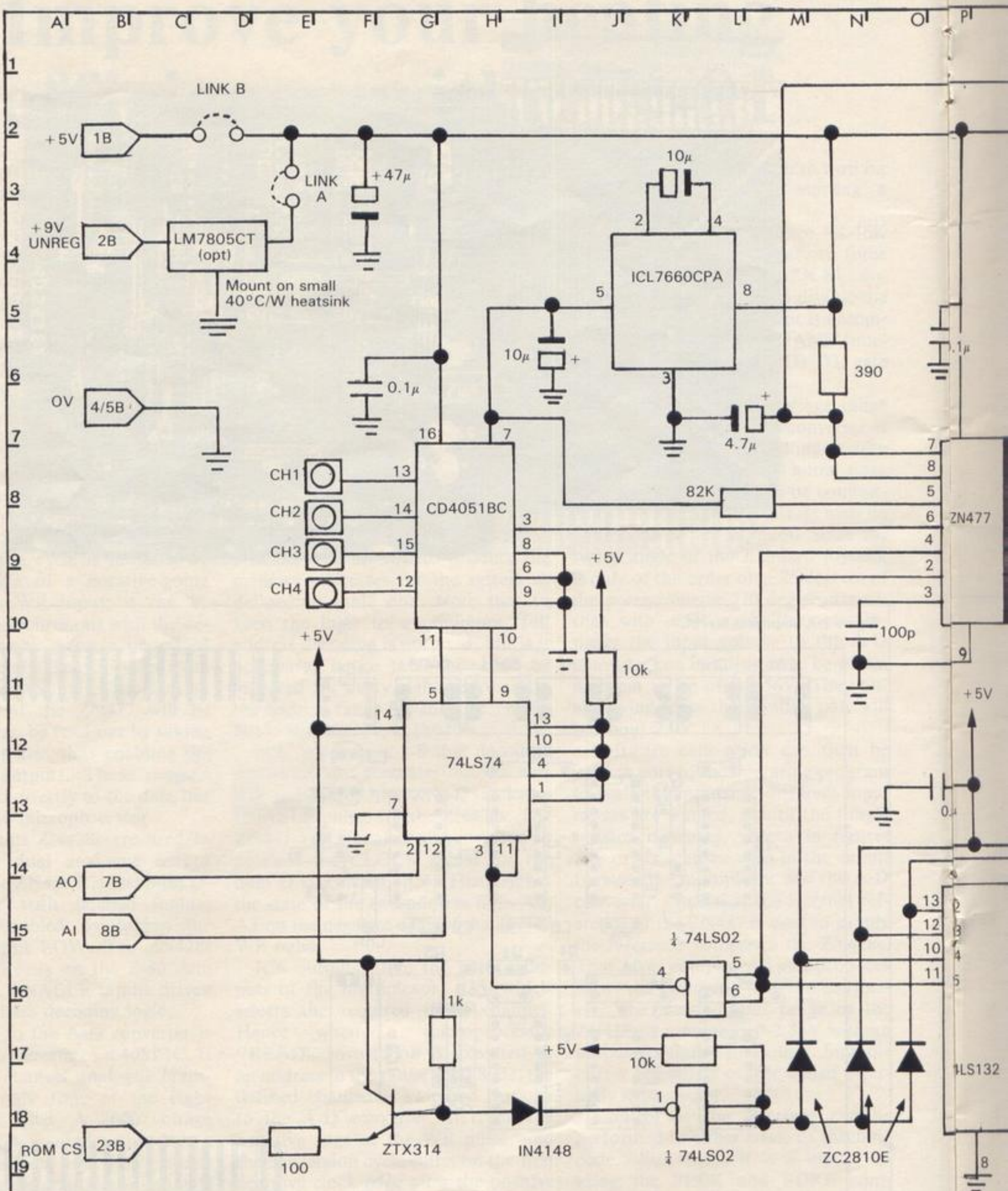
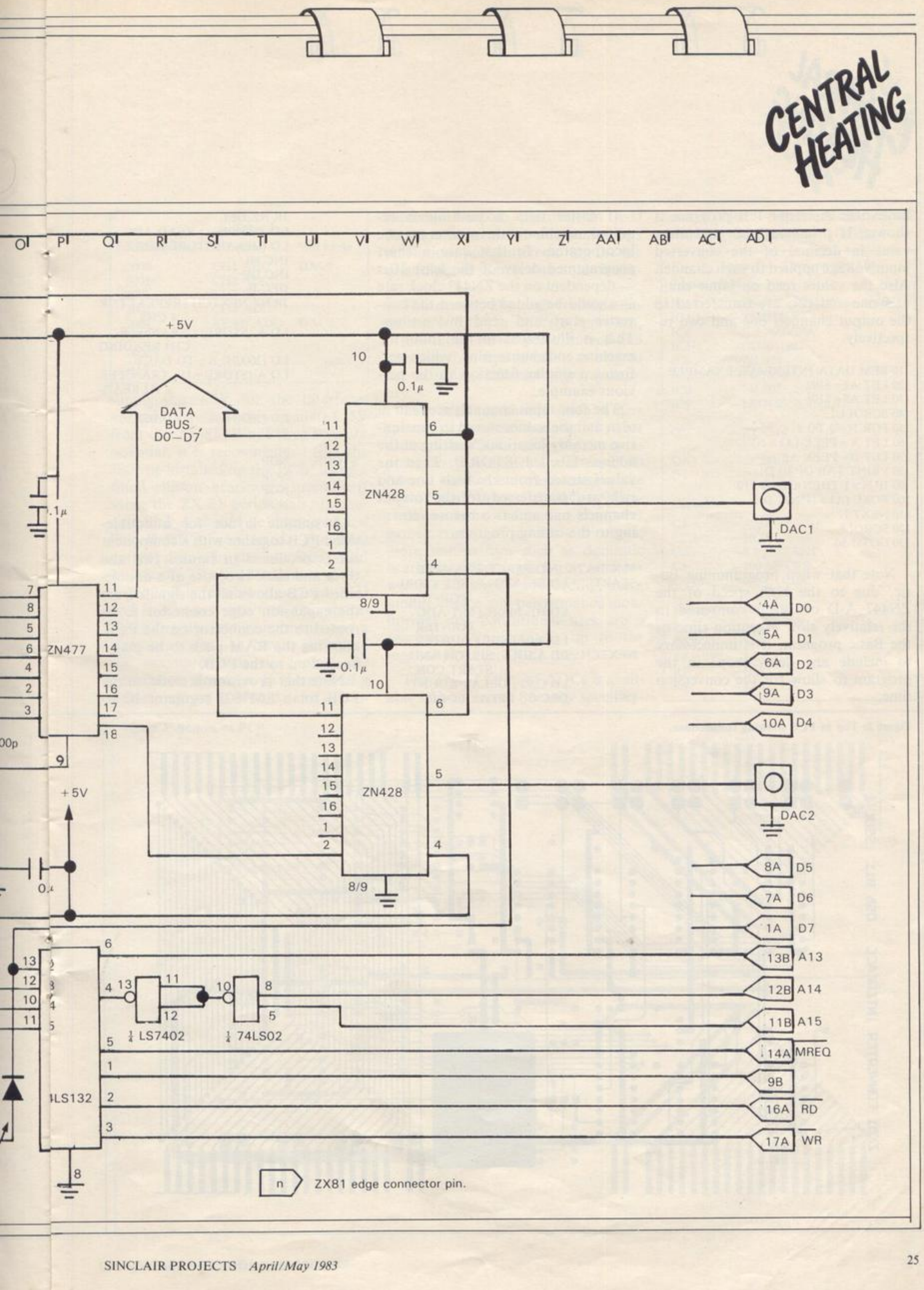


Figure 4: Circuit diagram.


O
P
Q
R
S
T
U
V
W
X
Y
Z
AA
AB
AC
AD
+5V
DATA BUS D0'–D7'
10
0.1μ
11
12
13
14
15
16
1
2
6
5
4
8/9
ZN428
ZN477
17
18
9
DAC1
DAC2
4A D0
5A D1
6A D2
9A D3
10A D4
8A D5
7A D6
1A D7
13B A13
12B A14
11B A15
14A MREQ
9B
16A RD
17A WR
¼ LS7402
¼ 74LS02
LS132
n ZX81 edge connector pin.

CENTRAL HEATING

table one. A simple test program is shown. It prints-out the equivalent value in decimal of the converted input voltage applied to each channel. Also the values read on input channels one and two are transferred to the output channels one and two respectively.

```
10 REM DATA INTERFACE EXAMPLE
20 LET A1=8192
30 LET A2=8196
40 SCROLL
50 FOR N=0 TO 3
60 LET X=PEEK (A1+N)
70 LET D=PEEK A2
80 PRINT TAB (N*8); D;
90 IF N>1 THEN GOTO 110
100 POKE (A1+(4*N)), D
110 NEXT N
120 SCROLL
130 GOTO 50
```

Note that when programming Basic, due to the high speed of the ZN447 A-D converter compared to the relatively slow execution time of the Basic program, it is unnecessary to include any delay loops in the program to allow for the conversion time.

If faster data acquisition is required, machine code routines may be incorporated. In that case a short programmed delay of the least 10μs — dependent on the ZN447 clock rate — should be added between the converter start and read instructions. That is illustrated in the following machine code subroutine, which performs a similar function to the previous example.

The four input channels are read in turn and the values stored in consecutive memory locations, starting at the address labelled 'STORE'. Then the values stored from channels one and two are transferred to the output channels one and two before returning to the calling program.

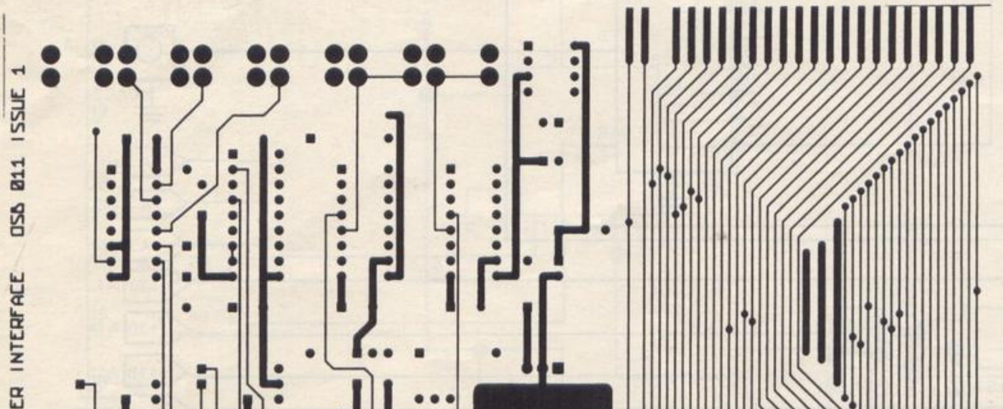
```
M/C DATA INTERFACE EXAMPLE
START:   LD DE, STORE; SET STORE
                        POINTER
         LD HL,2000H ; SET ADC
                        POINTER
         LD B,04 ; SET COUNTER
NEXTCH:  LD A,(HL) ; SEL CH AND
                        START CON
         LD C,82 ; DELAY COUNT
DEL:     DEC C ; DELAY LOOP
         JR NZ,DEL
         LD A,(2004H) ; READ ADC
         LD (DE),A ; STORE RESULT
         INC HL
         INC DE
         DEC B
         JR NZ,NEXTCH ; REPEAT FOR
                        4 CHS
         LD A,(STORE) ; TRANSFER
                        CH1 READING
         LD (2000H),A ; TO DAC1
         LD A,(STORE+1) ; TRANSFER
                        CH2 READ-
                        ING
         LD (2004H),A ; TO DAC2
         RET

STORE:   NOP
         NOP
         NOP
         NOP
```

A suitable layout for a double-sided PCB together with a component list is produced in figures two and three and table two. Use of a double-sided PCB allows all the signals from the expansion edge connector to be routed to the connector on the PCB, allowing the RAM pack to be piggy-backed on to the PCB.

Note that provision is made on the PCB for a LM7805 regulator IC to

Figure 2: Top of PCB showing connections.



CENTRAL HEATING

Table 1

Address (hex)	RD cycle	WR cycle
2000	CH1 ADC	DAC1
2001	CH2 ADC	
2002	CH3 ADC	
2003	CH4 ADC	
2004	READ ADC	DAC2
2005		
2006		
2007		

supply the +5V for the interface, instead of using the regulated 5V from the ZX-81. While probably not essential, it is recommended that the 7805 be installed on the PCB, using a small clip-on heatsink, especially if using the ZX-81 peripherals. In that case, link A should be connected and link B left unconnected. The total current drawn by the interface board is typically 75mA, 125mA maximum.

While it is feasible to perform A-D and D-A conversion with a comparator, a few discrete components and a microprocessor, it means tieing-up the microprocessor to control the conversion sequence. In the case of a Basic program it would take a considerable time to perform an A-D function to a useful number of bits resolution, probably requiring the use of additional sample and hold circuitry.

This design, although costing a little more than the discrete approach, offers much better performance and versatility, as well as being as easy to use as simply addressing extra memory.

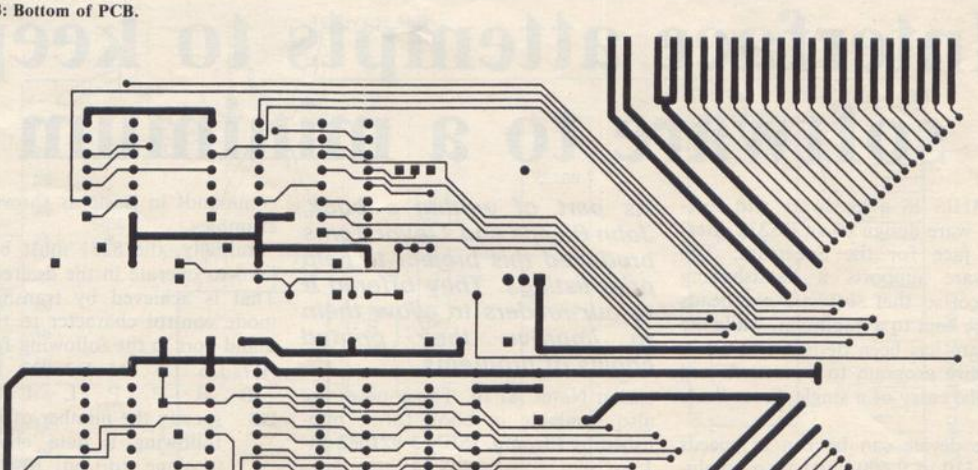
The interface should open a new sphere of interest and activity for users of the ZX-81, allowing the computer to communicate with and to control analogue transducers in the "real" analogue world. Such applications as more interactive computer games; control of models and robots; more serious uses such as domestic heating and lighting control; amateur weather station data recording; automobile engine and performance monitoring; and educational uses are a few ideas. The limit is up to the ingenuity of the ZX-81 user.

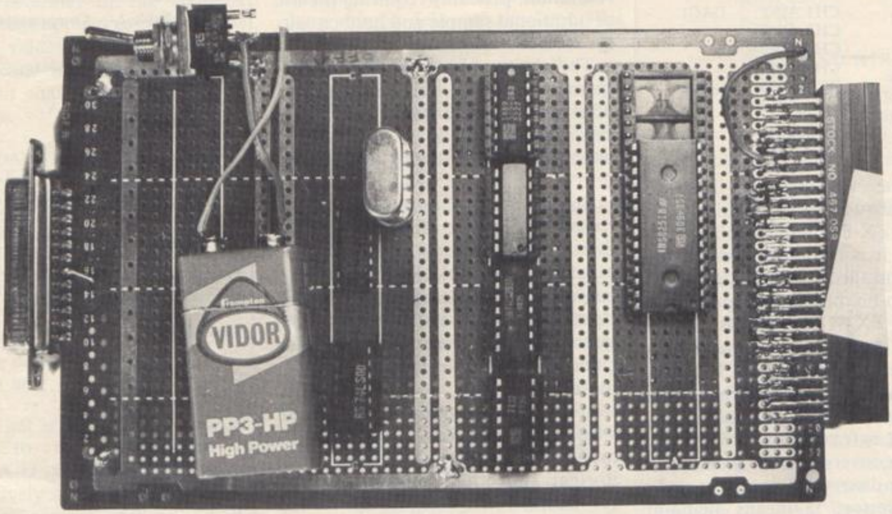
The ZN447 series of ADCs is available in a choice of linearity specifications, ZN447 — quarter LSB, ZN448 — half LSB, ZN449 — 1 LSB.

Table 2 — Components List

Semiconductors	
IC1	—ZN447E, ZN448E or ZN449E
IC2,3	—ZN428E-8
IC4	—ICL7660CPA
IC5	—CD4051BC
IC6	—74LS74
IC7	—74LS138
IC8	—74LS02
IC9	—LM7805CT (opt.)
Q1	—ZTX314
D1-3	—ZC2810E
D4	—IN4148
Capacitors	
C1	—47 μF
C2,3	—10 μF
C4	—4.7 μF TANT
C5	—100 pF CER
C6-10	—0.1 μF CER
Resistors — all 1/4W 5%	
R1	—390
R2,3	—10 k
R4	—1 k
R5	—100
R6	—82 k

Figure 3: Bottom of PCB.





Interface attempts to keep software to a minimum

As part of writing a book, John Phipps and Trevor Toms produced this project to help print listings. They offered it to our readers to allow them to improve their printed copies of programs.

THIS IS a hardware and software design of an RS232 interface for the Spectrum. The hardware supports a handshaking protocol so that software overheads may be kept to a minimum, while the software has been designed to allow an entire program to be transmitted with the entry of a single direct Basic line.

The device can be run at speeds from 110 to 9,600 baud by a combination of software and hardware strap alterations. It is based on the Intel 8251 UART and requires the use of a 9V battery to provide the 10–15V rail, since the − 12V connector on the Spectrum backplane does not supply − 12V.

Details of the 8251 operating modes can be found in Intel Application Notes AP16. Those notes are also available in book form, published by Elcomp, ISBN 3-921682-54-1.

The baud rate is generated by a 4702B; pins S0–S3 (output on pins 11–14) decide the appropriate rate value as shown in the table. The 8251 is installed at port addresses 73F hex (1855 decimal) for the command port and 71F hex (1823 decimal) for the data port. Those ports can be accessed directly using the IN and OUT commands in Basic as shown in the examples.

Initially, the 8251 must be initialised to operate in the desired mode. That is achieved by transmitting a mode control character to the command port in the following format:

D7	D6	D5	D4	D3	D2	D1	D0
B	B	P	P	L	L	R	R

BB specify the number of stop bits following a data character— 01 = one stop bit, 10 = 1.5 stop bits, 11 = two stop bits.

PP determine the parity of data— 00 or 10 mean no parity is to be checked, 01 = odd parity, 11 = parity.

LL indicates the data character length in bits—00 = 5 bits, 01 = 6 bits, 10 = 7 bits, 11 = 8 bits.

RR determine the baud rate factor—01 means switch rate, 10 = switch rate * 64.

To ensure that the 8251 is prepared to accept a mode control character, first send three null characters:

```
LET mode control = BIN BBPPLLRR
OUT 1855,0: OUT 1855,0: OUT 1855,0
OUT 1855, mode control
```

Then the UART can be used to receive or transmit data as desired. That is determined by sending at any time a command instruction to the command port, in the following format:

D7	D6	D5	D4	D3	D2	D1	D0
EH	IR	RTS	ER	SBK	RXE	DTR	TXE

EH not used in async. protocols.
IR 1 indicates that the 8251 is to be re-set.
RTS Request to send. Setting this bit to 1 will cause the output $\overline{RTS}$ to be forced to zero.
ER Error re-set. 1 re-sets all error flags (PE, OE, FE — see text).
SBK 1 forces TXD pin low.
RXE 1 enables data reception.
DTR 1 forces the $\overline{DTR}$ pin low (enabling data terminal ready).
TXE 1 enables data transmission.

At any time, the command port can be read to ascertain the current 8251 status. That should always be done prior to transmitting or receiving any data.

```
LET reply = IN 1855
```

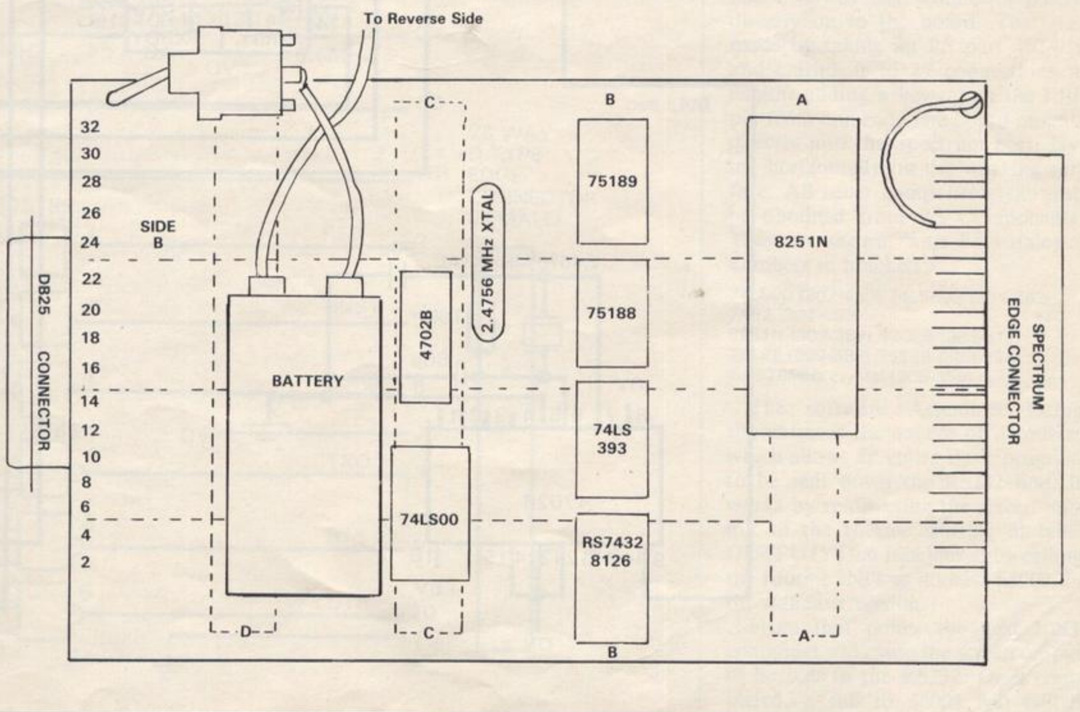
gives the following additive value in the variable "reply":

1 TXRDY—i.e., a character may be sent for transmission.
2 RXRDY—i.e., a character is waiting in the 8251 buffer to be read.
4 TXE—transmitter empty. That means another character may be loaded into the 8251 data port.
8 PE—parity error has been detected. That does not affect any operation; it serves only as a warning.
16 OE—over-run error. A character has been received by the 8251 before the previous one was taken from the buffer. That should not occur if the interfacing cable has RTS and DTR connected for handshake.
32 FE—framing error. A stop bit has not been detected. Again, this is only a warning.
64 SYNDET—not used in async. mode.
128 DSR—Data set ready. This contains the setting of the DSR pin on the 8251.

The following program transmits data at 9,600 baud to an external device, checking that transmission has been undertaken successfully. Normally, the device would be a fast printer.

```
10  LET initialise = 9700: LET transmit = 9800
    ....
    ....
100 GO SUB initialise
110 LET d$ = "string to transmit": GO SUB transmit
```

Figure 1: Top of the PCB



SPECTRUM RS232

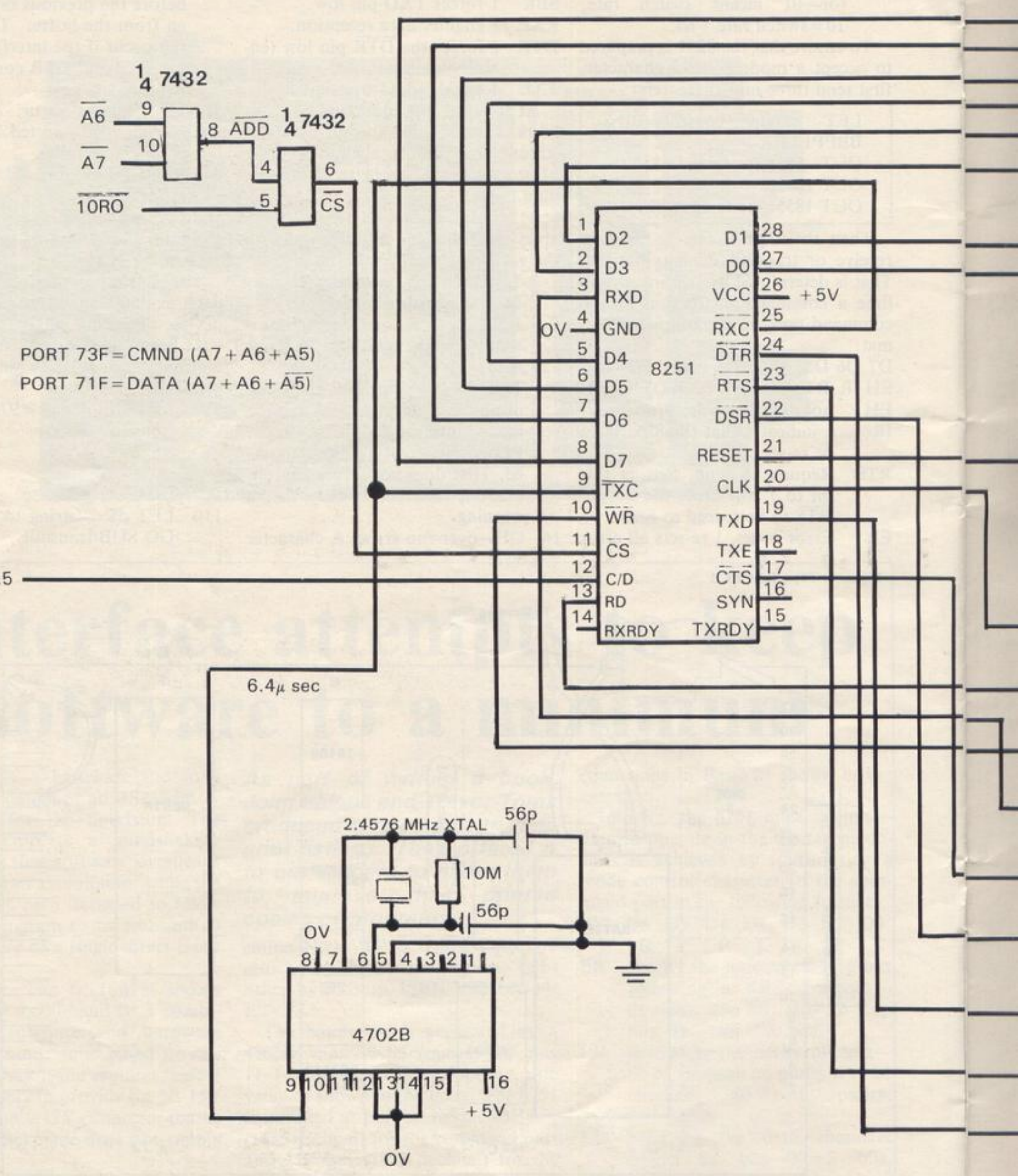
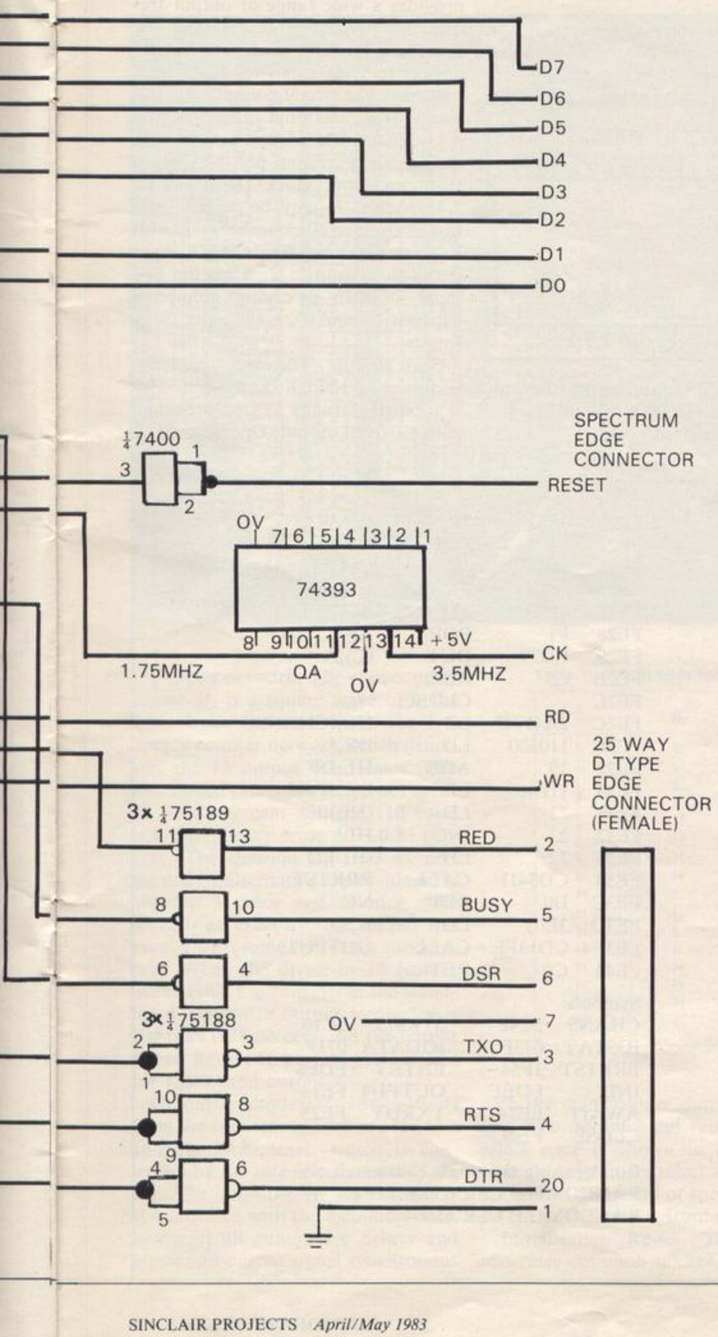


Figure 2: Circuit diagram

SPECTRUM RS232




```
9700 REM initialise 8251
9710 OUT 1855,0: OUT 1855,0:
     OUT 1855,0
9720 OUT 1855,BIN 11001110: OUT
     1855,BIN 00110111
9730 RETURN
9800 REM transmit d$
9810 FOR x=1 TO LEN d$
9820 LET reply=IN 1855
9830 LET reply=reply-INT (reply/
     8)*8
9840 IF reply<>5 AND reply<>7
     THEN GO TO 9820
9850 OUT 1823,CODE d$(x)
9860 NEXT x
9870 RETURN
```

Line 9820 loads the current 8251 status.

Line 9830 isolates the three least significant bits—TXE, RXRDY, TXRDY.

Line 9840 allows a character t be transmitted (line 9850) only when both TXE and TXRDY are true, regardless of the setting of RXRDY.

Our interface is mounted on an RS Eurocard — part no. 434-267 — and has a 44-way edge connector placed directly on to the board. That was made by taking an RS part 467-059 and cutting it to 27 connections in length, adding a keyway at the fifth pin from one end. The board mounts directly into the Spectrum port, laying horizontally on the working surface. All other components can also be obtained from RS Components. They are shown, with RS catalogue numbers in brackets:

74LS00 (307-480) 74LS393 (304-863)
7432 (304-093)
8251N (309-357) 4702B (303-517)
75188 (309-587) 75189 (309-593)
2.4576MHz crystal (303-359)

The software Assembler listing shown gives the source of a routine which allows an entire Basic program to be sent down the RS232 line. It works by re-directing the screen output to the routine starting at label OUTPUT. That is achieved by calling the routine INIT at address 65000, in the standard version.

From that point, the next LIST command will cause the screen output to be sent to the RS232. Once completed, a call to 65004 will switch

SPECTRUM RS232

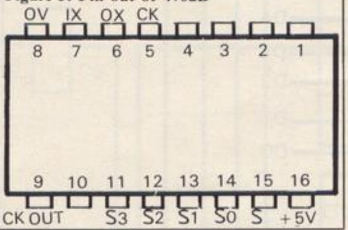
output back to the screen. Additionally that transmits three ETX characters to signify to any receiving device that the end has been reached.

Note that the ROM printing routines also re-direct screen output when a TAB control character is issued; thus this method cannot be used to re-direct PRINT statement output.

The routine was designed originally for use with a CP/M-based microcomputer so that all programs developed for the *Spectrum Pocket Book* could be transmitted for accurate printing ready for publication. The CP/M computer used a conversion program to translate all the keyword code values into their expanded form—e.g., a byte containing the decimal value 237 was converted into the text GO SUB—and additionally forced the printout to be shown in bold type.

This Assembler program can be entered using the Assembler given in the *Spectrum Pocket Book* or, alternatively, loaded with a simple hex loader program, using the column marked HEX as a guide to the code to be entered.

Figure 3: Pin-out of 4702B



The 4702B baud rate generator provides a wide range of output frequencies ranging from 50 baud for interfacing with electromechanical devices to 9,600 baud for high-speed modems. The circuit contains the following five functional subsystems:

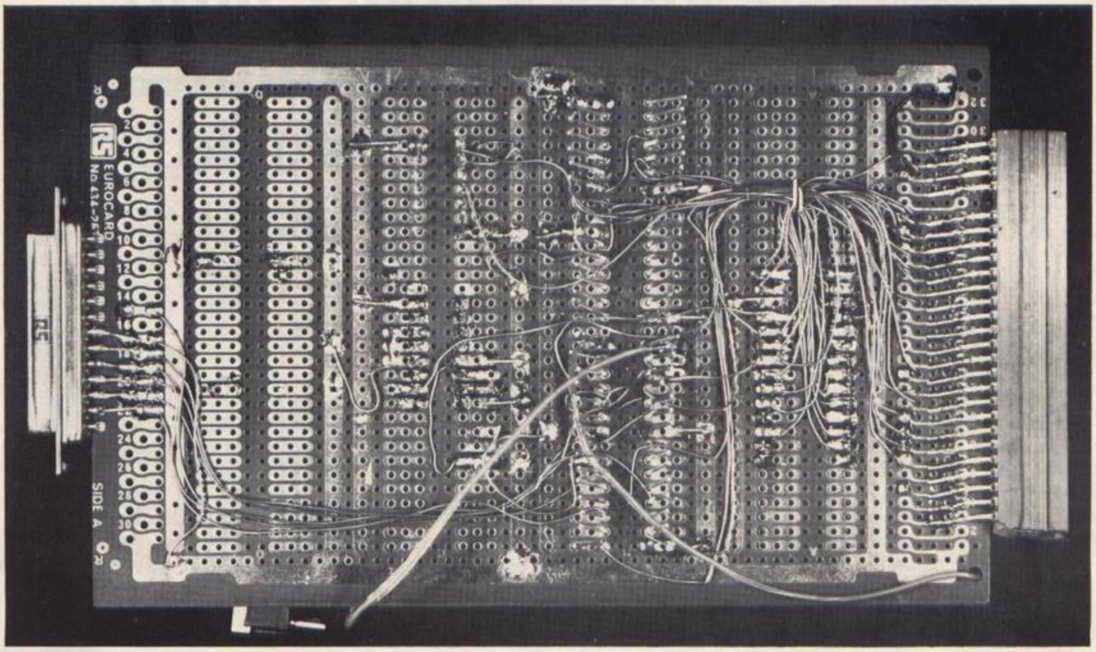
Oscillator: This circuit generates 16 output clock pulses per bit period from an input clock frequency of 2.4576MHz. It may be driven from two alternative clock sources—1, with E_{CK} at a logic low level the CK input is the clock source; 2, when the E_{CK} input is high a crystal connected between 1X and 0X is the input clock source.

Scan counter: The clock frequency is output on the CK output pin. This is applied internally to a ÷ 8 pre-scaler with buffered outputs Q_O, Q_1 and Q_2.

```
FDE8  1800           JR     INIT
FDEA  1800           JR     OUTPUT
FDEC  1800           JR     CLOSE
FDEE                 INIT:
FDEE  013F07         LD     BC,IOSTAT
FDF1  AF             XOR    A
FDF2  ED79           OUT    (C),A
FDF4  ED79           OUT    (C),A
FDF6  ED79           OUT    (C),A
FDF8  3E40           LD     A,40H
FDFA  ED79           OUT    (C),A
FDFC  3ECE           LD     A,0CEH
FDFE  ED79           OUT    (C),A
FE00  3E37           LD     A,37H
FE02  ED79           OUT    (C),A
FE04  2A4F5C         LD     HL,(CHANS)
FE07  110500         LD     DE,5
FE0A  19             ADD    HL,DE
FE0B  110000         LD     DE,OUTPUT
FE0E  73             LD     (HL),E
FE0F  23             INC    HL
FE10  72             LD     (HL),D
FE11  23             INC    HL
FE12  C9             RET
FE13                 OUTPUT:
FE13  F5             PUSH   AF
FE14                 AWAIT:
FE14  013F07         LD     BC,IOSTAT
FE17  ED768          IN     A,(C)
FE19  E604           AND    4
FE18  2000           JR     NZ,TXRDY
FE1D  CD541F         CALL   BRKTST
FE20  38F2           JR     C,AWAIT
FE22  F1             POP    AF
FE23  1800           JR     CLOSE
FE25                 TXRDY:
FE25  011F07         LD     BC,IODATA
FE28  F1             POP    AF
FE29  ED79           OUT    (C),A
FE2B  C9             RET
FE2C                 CLOSE:
FE2C  2A4F5C         LD     HL,(CHANS)
FE2F  110500         LD     DE,5
FE32  19             ADD    HL,DE
FE33  11F409         LD     DE,9F4H
FE36  73             LD     (HL),E
FE37  23             INC    HL
FE38  72             LD     (HL),D
FE39  CD541F         CALL   BRKTST
FE3C  D0             RET    NC
FE3D  3E03           LD     A,3
FE3F  CD13FE         CALL   OUTPUT
FE42  C9             RET

Symbols:
CHANS   5C4F    STRMS   5C10
IOSTAT  073F    IODATA  071F
BRKTST  1F54    ENTRY   FDE8
INIT    FDEE    OUTPUT  FE13
AWAIT   FE14    TXRDY   FE25
CLOSE   FE2C
```

Run by using the single command:
RANDOMIZE USR 65000:LIST:
RANDOMIZE USR 65004



Counter network: The scan counter output Q_2 is a square wave of $\frac{1}{8}$ the input frequency and drives the frequency counter network which generates the 13 output baud rates. The 134.5 baud rate has a frequency error of −0.87 percent and the 110 baud rate a frequency error of −0.83 percent. The division of 16/3 is performed by alternating the divide ratio between 5 twice and 6 once. The result is an exact average output frequency with some frequency modulation. When the divide-by-16 feature of the UART is considered the resulting distortion in output frequency is less than 0.78 percent. All the output signals have a 50 percent cycle except for 1,800 baud output.

Table 1: Truth table for baud rate select inputs.

S_3	S_2	S_1	S_0	Baud rate output
L	L	L	L	Multiplexed input (I_M)
L	L	L	H	Multiplexed input (I_M)
L	L	H	L	50
L	L	H	H	75
L	H	L	L	134.5
L	H	L	H	200
L	H	H	L	600
L	H	H	H	2400
H	L	L	L	9600
H	L	L	H	4800
H	L	H	L	1800
H	L	H	H	1200
H	H	L	L	2400
H	H	L	H	300
H	H	H	L	150
H	H	H	H	110

Output multiplexer: The outputs from the counter network are fed to a 16-input multiplexer, which is controlled by the rate select inputs (S_0 to S_3). The multiplexer output is re-synchronised with the incoming clock to cancel all cumulative delays and present an output signal synchronous with the scan counter outputs (Q_0 to Q_2). The output baud rate for each select code is shown in table one. Note that two codes select an I_M input allowing a zero baud or special rate to be fed to the output from this input.

Initialisation Re-set: This circuit generates common master re-set signal for all the internal flip-flops. This signal is derived from a digital differentiator which senses the first high level on the CK input after E_{CK} goes low. When E_{CK} is high, selecting the crystal input, CK must be low. A high level on CK would apply a continuous re-set.

Sinclair allows many more handicapped to communicate with the outside world

There are many aids which allow severely disabled people to write but most are expensive. Here, using an easily-available interface this cost is reduced substantially.

WITH THE ADVENT of a cheap printer from Sinclair and a cheap input/output port module from Technomatic, both of which are very cost-effective, a printing/sketching aid for disabled/handicapped people can be produced. The device would be orders of magnitude cheaper than other aids of that type and thus make it available to the many, rather than the few.

The design aims were single-switch operation; simple connection to the port and minimum component count; capable of being tailored to individual disabilities; minimum physical effort.

Single-switch operation is achieved through cunning software design, in which symbols on the screen do all the work, rather than multiple switch presses or sequences. The user stops a moving asterisk above the appropriate function symbol, e.g., [c] (inverse c) clears the screen.

Simple connection is achieved by orientating the software the proper way. With nothing connected, PEEKing the port yields 255, i.e., all bits high. A push-to-make switch is connected between channel 0 and ground. When pushed it yields a port value of 254. That avoids the need for pull up/pull down 1k resistors connected in profusion, grounding the other channels. In that way, using active low, no components are needed except a single push-to-make switch.

The software speed—the speed of movement of the asterisk—can be altered to almost any speed to suit the comfort of the person. The single switch can be operated by blowing down a tube, nodding your head through a light beam, using the teeth or the lips. So long as the disabled person can flex one muscle group, he can use the device. Minimum physical effort is achieved by allowing the asterisk to move automatically and have key depression only when a character is selected. That creates less strain than holding down the switch and releasing to select character.

The author did his best to incorporate those aims and believes that the resultant device will find many applications in the homes of the disabled, because of both its effectiveness and its cheapness. It is also felt that many establishments caring for disabled children or adults might like to review their equipment.

A typical situation where the program is being used is shown in figure two—a copy of the VDU screen. The bottom two lines on the screen are for character selection and the rest of the screen is for character printing. The asterisk moves automatically around the line of characters. Pressing the switch to stop the asterisk opposite a particular character prints that character at the top of the screen in the cursor position (*) and moves on the cursor. Holding down the switch causes auto repeat.

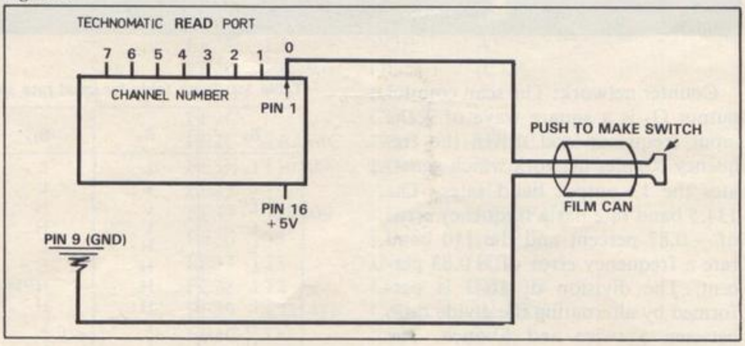
The black square after the inverse D prints a white blank space at the top of the screen. Holding down the switch with the asterisk above that black square causes auto space repeat. The inverse B causes a back space to delete last character and can be held down to delete whole words or lines.

Stopping the asterisk above the inverse D causes the asterisk to jump to the line below, to access the less-used characters and the graphics for drawing. The inverse C in that line causes the screen to clear and re-starts the asterisk on the top line automatically.

A complete screen copy is achieved by stopping the asterisk above the inverse P on the bottom line. Only the text is copied—the two character select lines are not.

To return to the top line, stop the asterisk above the inverse U and it reverts to the top line. It is much simpler than it sounds and, in a mat-

Figure 1.



AID FOR DISABLED

Figure 2: Representation of screen.

```
  10 CLS
  15 LET A=0
  16 LET B=0
  20 PRINT AT 19,0;"sPace,invers
e D,inverse sPaceABCDEFGHIJKLMNO
PQRSTUVWXYZ,. inverse B"
  30 PRINT AT 21,0;"2*sPace,inve
rse U1234567890+-*/( ),inverse U,
9raPhic 2,9raPhic 8,9raPhic A,9r
aPhic W,9raPhic D,9raPhic T,9raP
hic G,sPace,inverse CPB"
  35 LET C=10
  36 LET D=0
  50 PRINT AT C,D;" *";
  52 FOR J=1 TO 10
  53 IF PEEK 11000=254 THEN GOTO
 130
  63 NEXT J
  64 IF PEEK 11000=255 THEN LET
D=D+1
  65 IF D=31 THEN PRINT AT C,D;"
 ";
 100 IF D=31 THEN LET D=0
 120 GOTO 45
 130 LET Z=PEEK 16396+256*PEEK 1
6397+(C+1)*33+D+2
 140 LET Y=PEEK Z
 141 IF Y=169 THEN PRINT AT C,D;
"  "
 142 IF Y=169 THEN LET C=C+2
 143 IF Y=169 THEN GOTO 50
 144 IF Y=186 THEN PRINT AT C,D;
"  "
 145 IF Y=186 THEN LET C=C-2
 146 IF Y=186 THEN GOTO 50
 147 IF Y=167 THEN LET B=B-1
 148 IF Y=167 THEN PRINT AT A,B;
" *"
 149 IF Y=167 THEN GOTO 230
 150 IF Y=168 THEN GOTO 10
 160 IF Y=181 THEN GOTO 240
 165 IF Y=0 THEN GOTO 40
 170 IF Y=128 THEN LET Y=0
 180 PRINT AT A,B;CHR$ (Y);
 190 LET B=B+1
 200 IF B=31 THEN LET A=A+1
 210 IF B=31 THEN LET B=0
 220 IF A=18 THEN LET A=0
 225 PRINT AT A,B;"*";
 230 GOTO 40
 240 PRINT AT 19,0;"
                                "
 250 PRINT AT 21,0;"
                                "
 260 COPY
 270 GOTO 10
```

Program 1.

ter of minutes, the relative novice is able to write quickly on the screen and to use graphics to produce simple pictures.

The Technomatic port is available as a kit or ready-built from Technomatic Ltd although any port could be used. The kit arrives well-packed and with some good explanatory applications notes. Construction is relatively easy. The only criticism is that the board is not through-plated and soldering through the board takes time. Soldering on the RAM pack extension edge connector can be difficult. The single switch required to operate the device was attached as in figure one.

The switch is connected between pin 9 (ground) and pin 1 (channel 0) of the READ port. The port pushes on to the edge connector at the back of the ZX-81. Then push the printer on to the port edge connector and finally push on the RAM pack. That sandwich is more stable than it sounds and the author has experienced no trouble with poor connections. The decoding provided by Technomatic puts the port at address 11000 and a single byte can be PEEKed from or POKEd there. Switch status polling is achieved through PEEKing the byte at the port. With the switch operated, i.e., channel zero grounded, PEEK yields 254 with the switch released, PEEK yields 255. It is as simple as that and if you use a different memory-mapped port you change the PEEK address.

A listing of the software is shown in program one. At lines 240 and 250 there are 32 spaces between the quotation marks. Lines 52, 62 and 63 poll the switch status; if depressed, line 62 causes a jump to line 130. At that point the asterisk has stopped opposite a particular character. That character is PEEKed—lines 130 and 140—and Y is given that value.

Lines 141 to 170 test Y, to see if it is a function character, rather than a character to be printed, and the program then takes the appropriate action for that function. For example, if Y has the value 181 (inverse P), the program jumps to the copy screen routine starting line 240.

The speed of traverse of the asterisk is altered by changing the range of the J values in line 52. In that way the speeds at which different individuals work can be accommodated.

Those without an input/output port module can connect two wires to the keyboard matrix connectors inside the computer and replace line 62 with:

62 IF INKEY$ = "A" THEN GO TO 130 where A is the letter for the keyboard connections chosen. Then replace line 64 with:

64 IF INKEY$ = "" THEN LET D = D + 1

The program/hardware configuration should bring printing/drawing within the reach of many disabled people. The author would also be pleased to answer queries from bona-fide disabled people and related establishments on any implementation problems experienced while using the system.

Decoding the memory to find where data goes

To help with the understanding of how to make a variety of future projects, Ian Mellor describes how information is accepted, travels through the computer and is stored. Articles based on this will follow in subsequent issues.

MANY READERS have asked us for the details of how a computer works and a number of other subjects. This series will take you through ideas, such as analogue-to-digital conversion, and how the screen and keyboard work. We begin with an explanation of how the memory works and how a particular piece of data reaches the correct memory location without becoming lost.

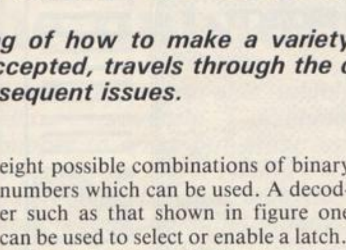
A description of the architecture of a microcomputer shows that it consists of a microprocessor — or CPU — memory — ROM and RAM — and input/output ports. The memory is used to store instructions and information and the input/output ports are used to exchange information with the outside world via items such as keyboards, and VDU. They are connected by the address and data buses and some control lines.

Information travels to and from the microprocessor along the data bus which in the ZX-81 is eight bits (lines) wide. The information consists of a binary word which is sent in parallel — one bit down each of the eight lines. From where that information emanates or where it goes depends on the "address" on the address bus, which in the example is 16 bits wide, i.e., it has 16 lines; and the signals on various other lines called the control lines. We consider only the read ($\overline{RD}$) and write ($\overline{WR}$) control lines.

A line or bar drawn over the letter(s) of a mnemonic indicates that it is active low, in other words a low voltage on the $\overline{WR}$ line signals that the CPU is performing a write instruction. If we consider a three-bit address bus for simplicity, there are eight possible combinations of binary numbers which can be used. A decoder such as that shown in figure one can be used to select or enable a latch.

The data bus is bi-directional; information can be sent from the CPU along the data bus to be stored in one of the latches or read in from the latch. When data is being sent the $\overline{WRITE}$ line goes low and that is used by the latching devices to switch them into input mode. Any latch which is then addressed or enabled will store whatever data is in the data bus at the time; all previous data stored in that latch will, of course, be overwritten or 'lost'.

Figure 1A



Four to 16-line decoder. Any book on digital electronics or TTL data book will have an example.

When the CPU is reading data it reads whatever is on the data bus while the READ line is low or the WRITE line — or R/$\overline{W}$ line — is high. The addressed latch will be in output mode. The information stored in the latch can be read over and over again and will not change until new information is written into the latch. Some latches are one-way and act as input or output ports.

An output is produced when the microprocessor sends data to an output location as if it were writing data to a memory location. The 5-bit data word is available on the five output lines from the latch to control relays, motors, display unity. Information from the outside keyboard is fed on to the data bus via a latch — or tri-state buffer — as though the microprocessor were reading data from a memory address.

It is often desirable to know whether a particular address refers to ROM, RAM or input/output and there are so many memory locations that it is difficult to grasp the overall organisation. If all the 16 address lines are fully-decoded there are 65536 (64K) addresses. Note that 1K means 1,024 memory locations. An effective way of picturing the memory space in a computer is to use a memory map as shown in figure five. That can be labelled in decimal and hexadecimal to allow cross-references to be made. In many microcomputers not all of the address space is decoded. This is because not all of the address space is used and areas are left to allow flexibility of expansion. Also, to save money, only essential decoding is done.

Memory traditionally has been partitioned into $\frac{1}{4}$, $\frac{1}{2}$ or 1K blocks or banks. The rest of the decoding is included inside the IC package with the memory. There may well be 1,024 or more bytes of memory and all the necessary decoding on a single IC. The address in the memory map where the 1,024 bytes are located depends on how the chip enable line has been decoded. A block of $\frac{1}{4}$K of memory is sometimes called a page

DECODER

— 0-255 decimal = 0-FF hex; on the memory map diagram each one-millimetre division represents 256 bytes or one page of memory. There are 256 pages in 64K.

The shaded parts of the columns represent a low on the address line, thus when A13 is high it selects memory in blocks:

8,192–16,383 (8–16K)
24,767–32,768 (24–32K)
40,960–49,152 (40–48K)
and 57,344–65,536 (56–64K)

Consider a case where 256×8 RAMs and ROMs are being used — 256×8 means 256 bytes. Because each RAM or ROM contains 256 locations, eight address bits must be used to select the correct location within each chip ($2^8 = 256$). If we want 1K bytes of ROM and 3K bytes of RAM a total of four ROM chips and 12 RAM chips are needed.

The addressing process must select the correct chip. The chip-select (CS) put to each RAM or ROM can be used. Up to eight memory chips can be selected using linear addressing; each address line A8-A15 selects one IC. That is deep because no decoders are needed but it limits further expansion.

Before deciding where and how to add further inputs or outputs to a ZX-81 it is necessary to understand how the system handles its inputs and outputs. Then a position on the memory map can be selected to place the I/O ports. The present inputs and outputs are screen display (out); keyboard (in); cassette (in & out); printer (in & out).

The single Z-80 microprocessor chip in a ZX-81 handles all the input and output control as well as performing the processing. Further economies are made in the way the decoding is performed to minimise the hardware required.

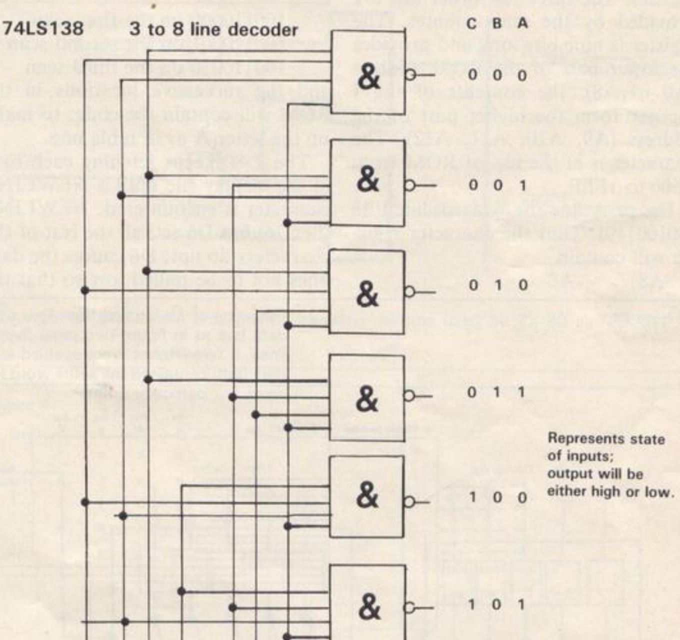
The display file is an area of RAM containing lines of text or other characters, each line being terminated by the NEWLINE character, i.e., decimal 118 (hex 76..01110110). The allowable character codes are the numbers 0 to 63 (hex 0 to 3F).

The monitor initiates a scan by jumping to the first byte of the line with A15 of the program counter set and with the interrupt enabled. That is an op code fetch cycle. Line A15 is so connected within the Sinclair logic chip, a Ferranti ULA, that the data lines are pulled low so that the CPU sees zeros on the data bus and executes a NOP instruction (code 00).

After executing the NOP instruction, the Z-80 increments the program counter. The CPU then addresses the next item of data from the display file in RAM. The CPU samples data on the data bus on the rising edge of the T3 clock pulse and that same edge is used to turn off RD and MREQ

The SLC does two things; first, it latches the data into the character address register on the SLC; then it

Figure 1



Simplified — omit enable line.
Only one of the eight output lines can be low at any time. The codes on each line show the states of the inputs A, B, C for that line to be active (low). This is usually expressed in a truth table. See data sheets for pin-out diagram and truth table and compare to this diagram.

drives the data bus low before T3 so that the CPU will 'see' a NOP instruction.

Each line is scanned eight times to produce the necessary eight rows for the character matrix. Each line is scanned 16 times; two scan lines produce one row of dots. The same codes, representing the character, are loaded into the character register each time.

Bits 0,1,2,3,4 and 5 of the code are placed into bits 3,4,5,6,7 and 8 of the register. The three low-order bits are provided by the scan counter. The register is nine bits long and provides the lower part of the ROM address (A0 to A8); the contents of the I register form the higher part of the address (A9, A10, A11, A12). The character is at the top of ROM from 1E00 to 1FFF.

Hence A has the code decimal 38 (00100110). Thus the character register will contain

A8 A0

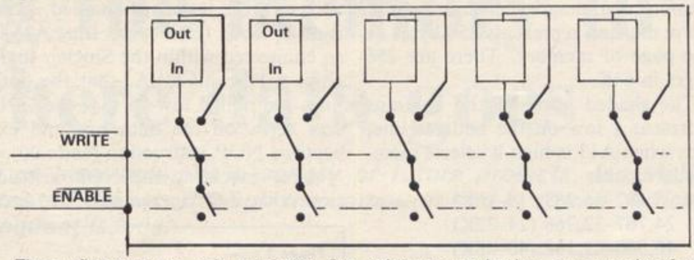
100110000 on the first scan
100110001 on the second scan
100110010 on the third scan

and the successive locations in the ROM will contain the codes to make up the letter A as in table one.

The Z-80 keeps fetching each byte in the display file until a NEWLINE character is encountered. NEWLINE (hex 76) has D6 set; all the rest of the characters do not. D6 causes the data lines not to be pulled low so that the CPU 'sees' 76 and halts, because 76h is the Z-80 HALT instruction.

A HALT instruction causes the CPU to execute endless NOP instructions until an interrupt is received. Each cycle in the HALT state is a normal M1 (op-code fetch) cycle except that the CPU ignores the data bus and a NOP is forced internally to the CPU. The refresh register continues to be incremented during a HALT state.

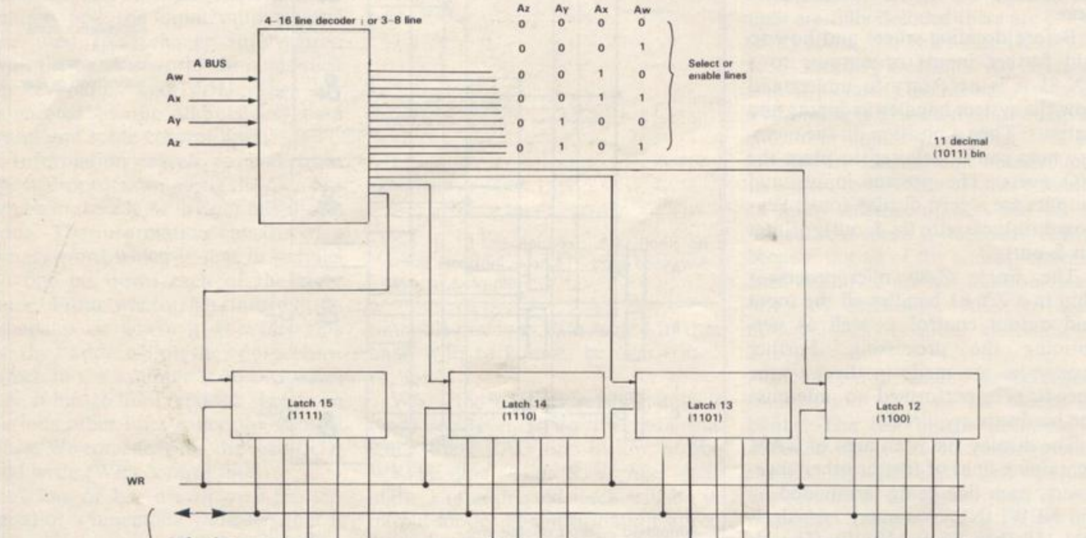
Figure 3: A five-bit memory location or latch



These diagrams are only symbolic. See reference on latches or memories for operation.

Figure 2

Decoding of 16 latches from four address lines. All the latches are connected to the data bus as in figure two; only five lines are shown when it would consist of eight lines. If two latches were enabled at the same time there would be a conflict, unless they both contained the same word (data). It is important that only one latch is turned on at any particular time.



DECODER

Just before beginning the line scan, the monitor loads the refresh counter with a number. That counter is incremented every M1 cycle. The number is such that after 32 M1 cycles the A6 line will go low. That line is connected to the INT line and so causes an interrupt terminating the HALT.

The interrupt subroutine produces the video line sync pulse and the character register counter is incremented ready to scan the line again. After 16 scans—two TV line scans are needed for each dot—the system loops to the next line to be displayed.

The clock frequency for the ZX-81 was chosen so that each line takes 64μs to be scanned, the time required between line sync pulses in a TV. At the end of the display file a frame sync pulse is generated.

After displaying a screenful of information the monitor jumps to the keyboard scan routine. The Z-80 instructions

LD BC(FEFE)
IN A(C)

put the code 11111110 (B reg.) on to address lines A8 to A15. The fact that IN A(FE) causes A0 to go low is used to latch the keyboard/cassette input port. The B register is shifted left and the routine repeated so that each address line goes low in turn.

The resistors RP3 hold the lines KBD0-KBD4 high unless a key has been depressed, in which case the combination of the number loaded into the A register and the contents of the B register will form the code of the key depressed.

Bit 7 of this port is used as the serial input from the cassette during a cassette load operation.

Summarising the use of address lines in I/O instructions:

A0 is used to enable the keyboard and cassette input latch using address (FE).

A1 is used to cause a TV display in SLOW mode using the NMI and hardware enabled by address (FD).

A2 is used to enable the printer input/output latch at address (FB).

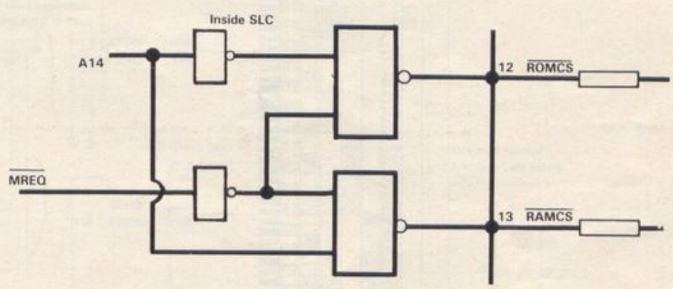
A8 to A15 are used for keyboard output.

It is not necessary to decode an I/O port fully or even a section of mem-

Table 1

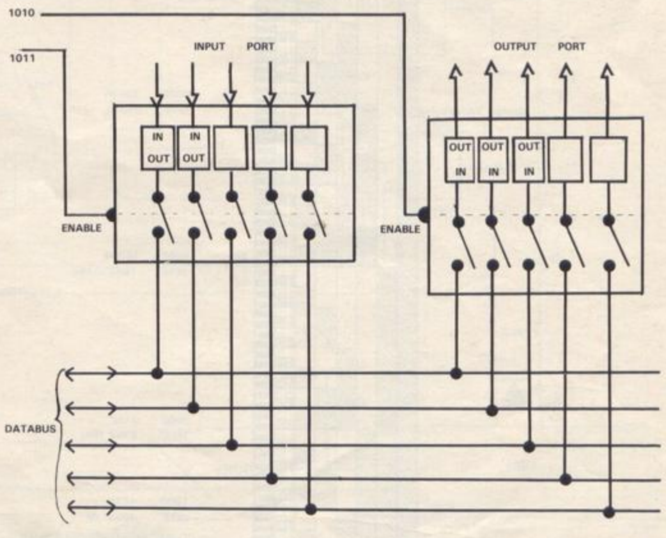
Decimal	Hex			Hex	Decimal
7984	IF30	0001111	100110000-00000000	00	0
7985	IF31		100110001-00011100	1C	28
7986	IF32	upper 7 bits	100110010-00100010	22	34
7987		of address	100110011-00100010	22	34
		are IE hex	100110100-00111110	3E	62
		supplied	100110101-00100010	22	34
		by I	100110110-00100010	22	34
		register	100110111-00000000	00	0

Figure 6



Example of simple and inexpensive decoding scheme used by ZX-80 and ZX-81 computers.

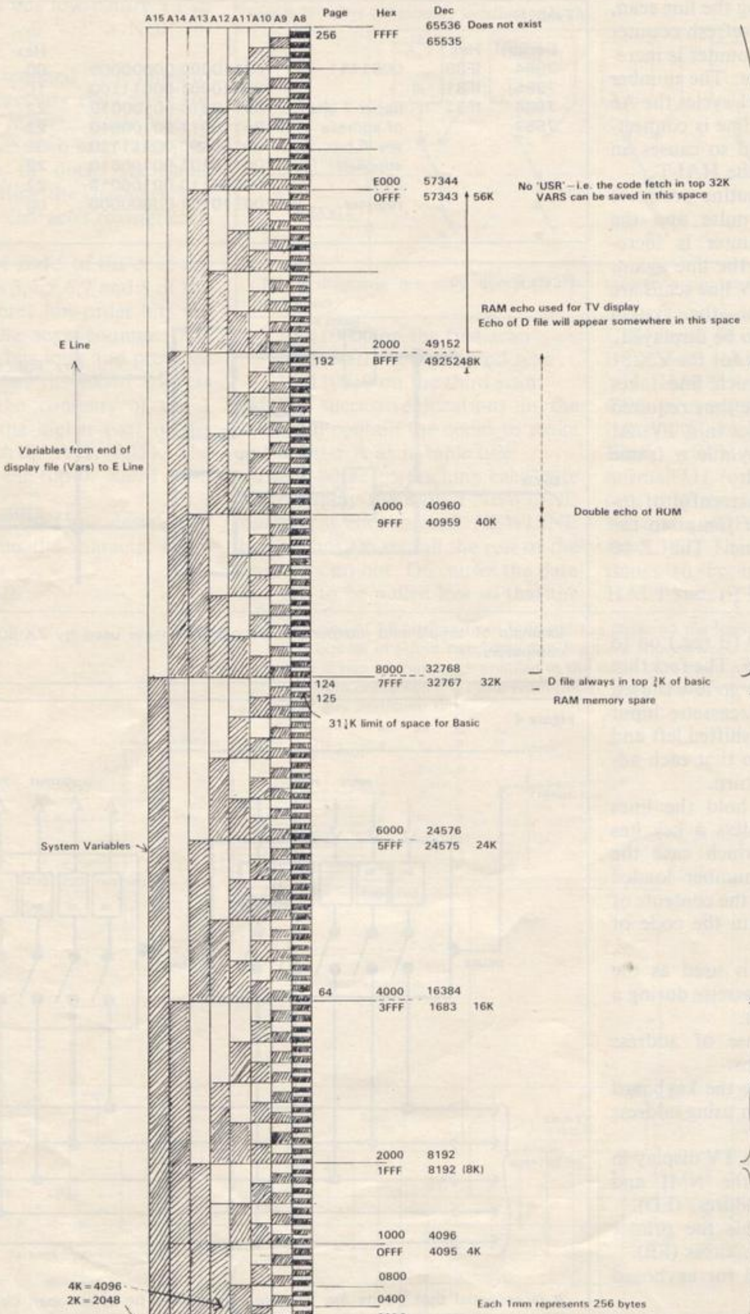
Figure 4



It is essential that inputs are tri-state. They must be in an open circuit or high-impedance state until they are enabled.

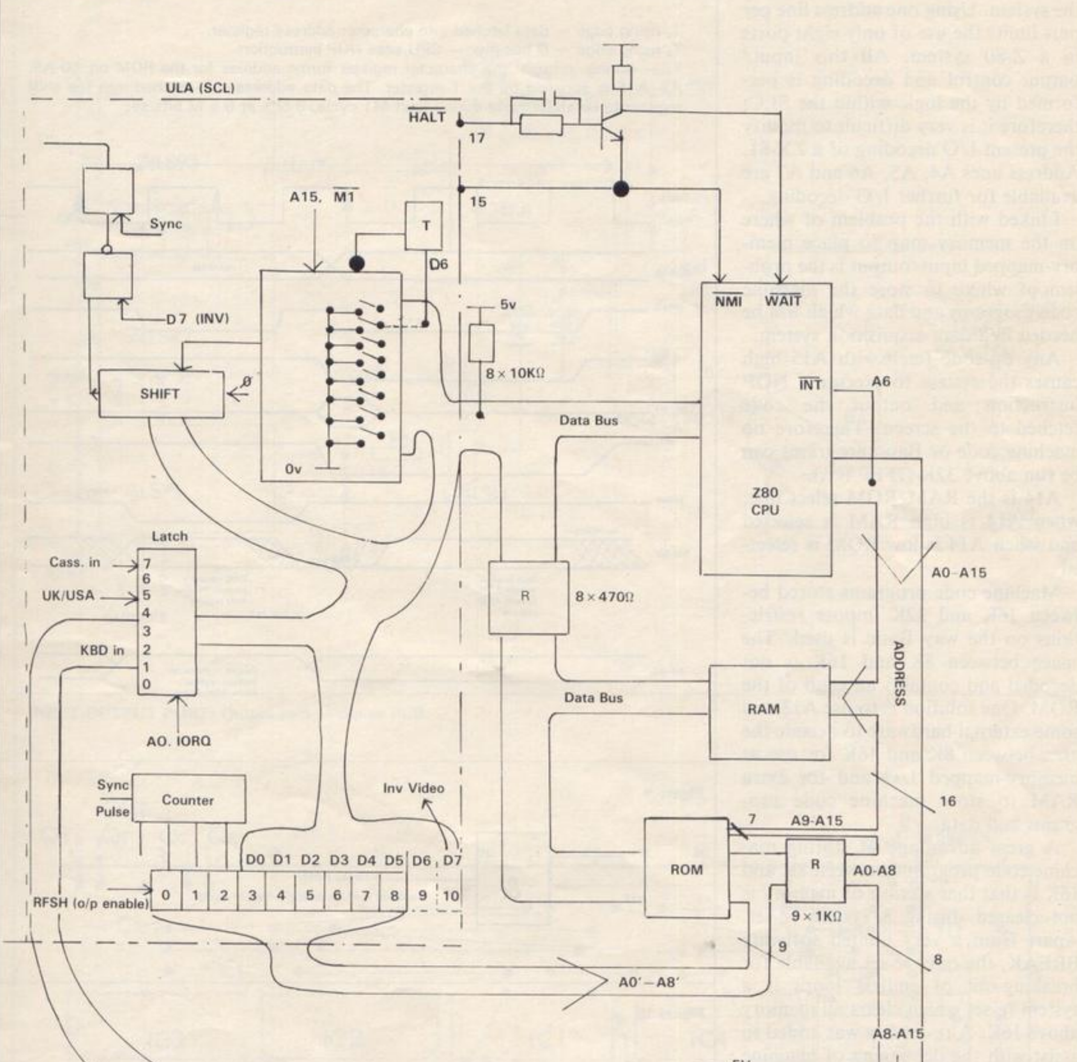
DECODER

Figure 5



DECODER

Figure 7L ZX-81 input/output organisation



ZX-81 I/O Organisation

DECODER

ory so long as there are no clashes in the system. Using one address line per port limits the use of only eight ports in a Z-80 system. All this input/output control and decoding is performed by the logic within the SLC; therefore it is very difficult to modify the present I/O decoding of a ZX-81. Address lines A4, A5, A6 and A7 are available for further I/O decoding.

Linked with the problem of where on the memory map to place memory-mapped input/output is the problem of where to store the machine code programs and data which will be needed in a data acquisition system.

Any op-code fetch with A15 high causes the system to execute a NOP instruction and output the code fetched to the screen. Therefore no machine code or Basic programs can be run above 32K (7FFF hex).

A14 is the RAM/ROM select line; when A14 is high RAM is selected and when A14 is low ROM is selected.

Machine code programs stored between 16K and 32K impose restrictions on the way Basic is used. The space between 8K and 16K is not decoded and contains an echo of the ROM. One solution is to use A13 and some external hardware to decode the area between 8K and 16K for use as memory-mapped I/O and for extra RAM to store machine code programs and data.

A great advantage of storing machine code programs between 8K and 16K is that that section of memory is not cleared during a system re-set. Apart from a very limited software BREAK, the only re-set available for breaking-out of endless loops is a system re-set which clears all memory above 16K. A re-set key was added to assist with the debugging of machine code programs.

Memory above 32K can be used by the Basic interpreter for the storage of variables or by machine code programs for the storage of data. To ensure that there is no clash between data above 48K and the echo of the display file it is necessary to add the circuit shown in figure nine. Data will not be accessed during the fetch instruction used by the display routine.

Figure 8

T_2 rising edge — data latched into character address register.
T_3 rising edge — D bus low — CPU sees NOP instruction.
T_4 — during 'refresh' the character register forms address for the ROM on A0-A8. A9–A15 is supplied by the T register. The data addressed is latched into the shift register to be shifted and during next M1 cycle, 8 bits at 6.5 M bits/sec.

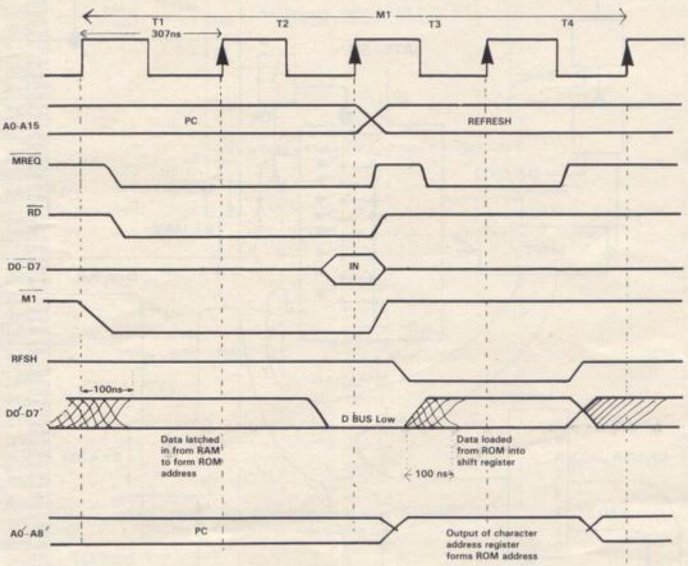


Figure 9

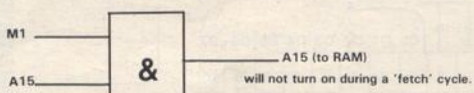
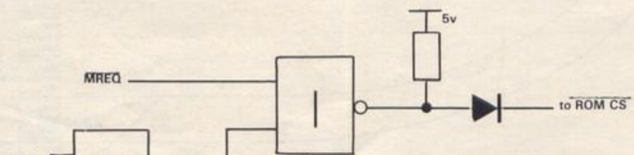


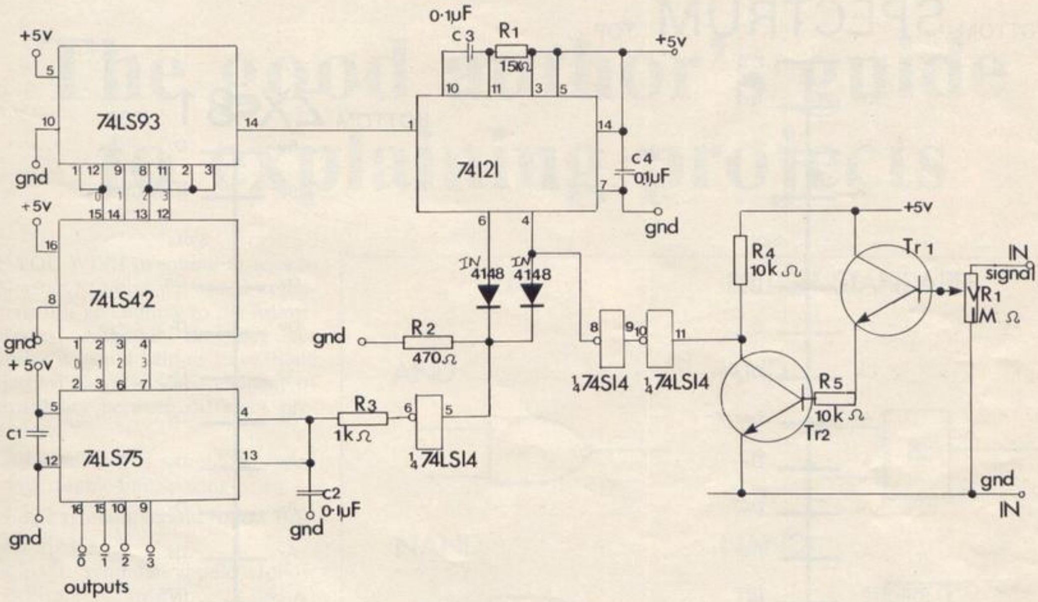
Figure 10



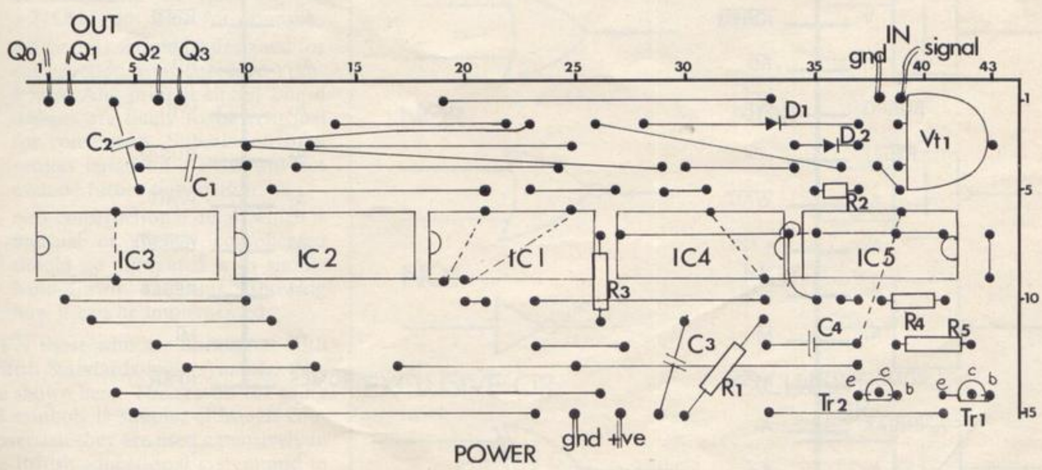
Circuit to disable ROM in space 8K–16K

The circuits are usually incorporated into the appropriate add-on units.

INPUT-OUTPUT PORT: Output port — circuit diagram



INPUT-OUTPUT PORT: Output port — top or PCB



EDGE CONNECTOR

Edge Connector signal allocation

SPECTRUM

BOTTOM	TOP
A_{14}	A_{15}
A_{12}	A_{13}
5v	D_7
9v	
Slot	Slot
Ov	D_0
Ov	D_1
CK	D_2
A_0	D_6
A_1	D_5
A_2	D_3
A_3	D_4
$\overline{\text{IORQGE}}$	$\overline{\text{INT}}$
Ov	$\overline{\text{NMI}}$
Video	$\overline{\text{HALT}}$
Y	$\overline{\text{MEMREQ}}$
V	$\overline{\text{IOREQ}}$
U	$\overline{\text{RD}}$
$\overline{\text{BUSRQ}}$	$\overline{\text{WR}}$
$\overline{\text{RESET}}$	−5v
A_7	$\overline{\text{WAIT}}$
A_6	+12v
A_5	−12v
A_4	$\overline{\text{MI}}$
$\overline{\text{ROMCS}}$	$\overline{\text{RFSH}}$
$\overline{\text{BUSAK}}$	A_8
A_9	A_{10}
A_{11}	

ZX-81

BOTTOM	TOP
5v	D_7
9v	$\overline{\text{RAM CS}}$
Slot	Slot
Ov	Do
Ov	D_1
$\overline{\varnothing}$	D_2
A_0	D_6
A_1	D_5
A_2	D_3
A_3	D_4
A_{15}	$\overline{\text{INT}}$
A_{14}	$\overline{\text{NMI}}$
A_{13}	$\overline{\text{HALT}}$
A_{12}	$\overline{\text{MREQ}}$
A_{11}	$\overline{\text{IORQ}}$
A_{10}	$\overline{\text{RD}}$
A_9	$\overline{\text{WR}}$
A_8	$\overline{\text{BUSAK}}$
A_7	$\overline{\text{WAIT}}$
A_6	$\overline{\text{BUSRQ}}$
A_5	$\overline{\text{RESET}}$
A_4	$\overline{\text{MI}}$
$\overline{\text{ROMCS}}$	$\overline{\text{RFSH}}$

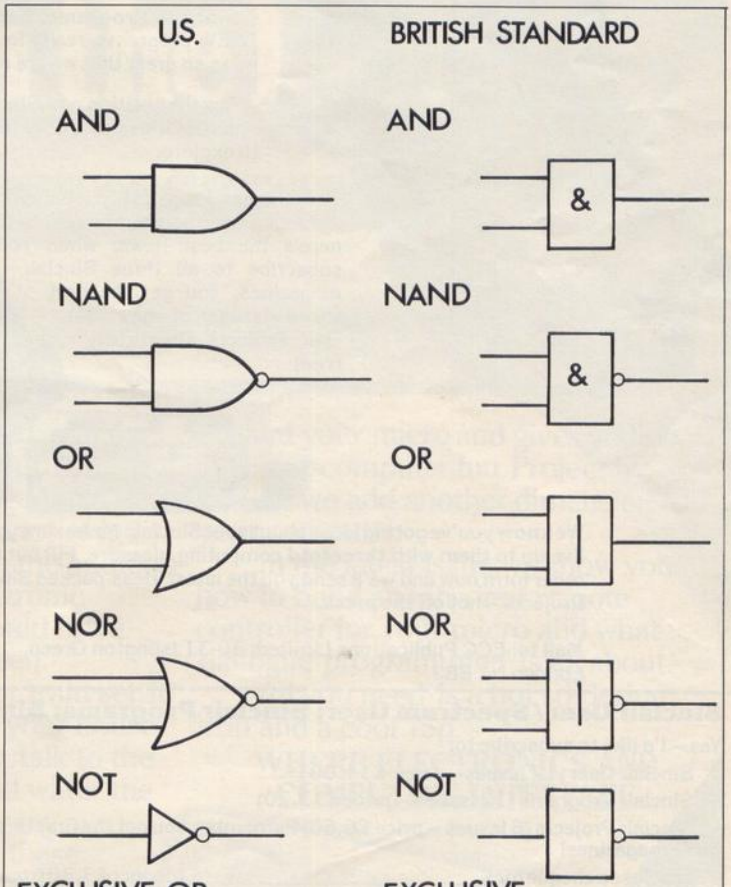
ARTICLE OUTLINE

The good author's guide to explaining projects

IF YOU WISH to submit articles to *Sinclair Projects* we would appreciate it if you adhere to the following rules. Although they are not exclusive it would help us to evaluate projects if there is some element of compatibility between different presentations.

- All manuscripts should be typed with double-line spacing.
- Logic symbols should follow British Standards.
- Circuit symbols should follow British Standards.
- Circuit diagrams should have the values of the components shown, not a reference to a component table.
- Parts of integrated circuits should be designated in this way — ¼ 74LS14, not IC5a for example.
- All circuits should be designed for construction using standard Veroboard. Any printed circuit board designs are likely to be returned for conversion. Submission of a project based on a PCB will not exclude future publication.
- Any constructional detail which is unusual or slightly complicated should be illustrated with simple hand-drawn diagrams, showing how it can be implemented.

For those who are unfamiliar with British Standards logic symbols, they are shown here. The reason for using BS symbols is simple, although controversial; they are used extensively in the British educational system and in British industry. As most of our readership is in those areas we use circuit symbols which are most familiar.




IBM
IBM

MAIL ORDER ADVERTISING

British Code of Advertising Practice

Advertisements in this publication are required to conform to the British Code of Advertising Practice. In respect of mail order advertisements where money is paid in advance, the code requires advertisers to fulfil orders within 28 days, unless a longer delivery period is stated. Where goods are returned undamaged within seven days, the purchaser's money must be refunded. Please retain proof of postage/despatch, as this may be needed.

Mail Order Protection Scheme

If you order goods from Mail Order advertisements in this magazine and pay by post in advance of delivery, *Sinclair User* will consider you for compensation if the Advertiser should become insolvent or bankrupt, provided:

(1) You have not received the goods or had your money returned; and

(2) You write to the Publisher of *Sinclair User* summarizing the situation not earlier than 28 days from the day you sent your order and not later than two months from that day.

Please do not wait until the last moment to inform us. When you do write, we will tell you how to make your claim and what evidence of payment is required.

We guarantee to meet claims from readers in accordance with the above procedure as soon as possible after the Advertiser has been declared bankrupt or insolvent (up to a limit of £4,250 per annum for any one Advertiser so affected and up to £10,000 per annum in respect of all insolvent Advertisers. Claims may be paid for higher amounts, or when the above procedure has not been complied with at the discretion of *Sinclair User*, but we do not guarantee to do so in view of the need to set some limit to this commitment and to learn quickly of readers' difficulties).

This guarantee covers only advance payment sent in direct response to an advertisement in this magazine (not, for example, payment made in response to catalogues etc, received as a result of answering such advertisements). Classified advertisements are excluded.