60p

YOUR COMPUTER

MAY 1982

Vol.2 No.5

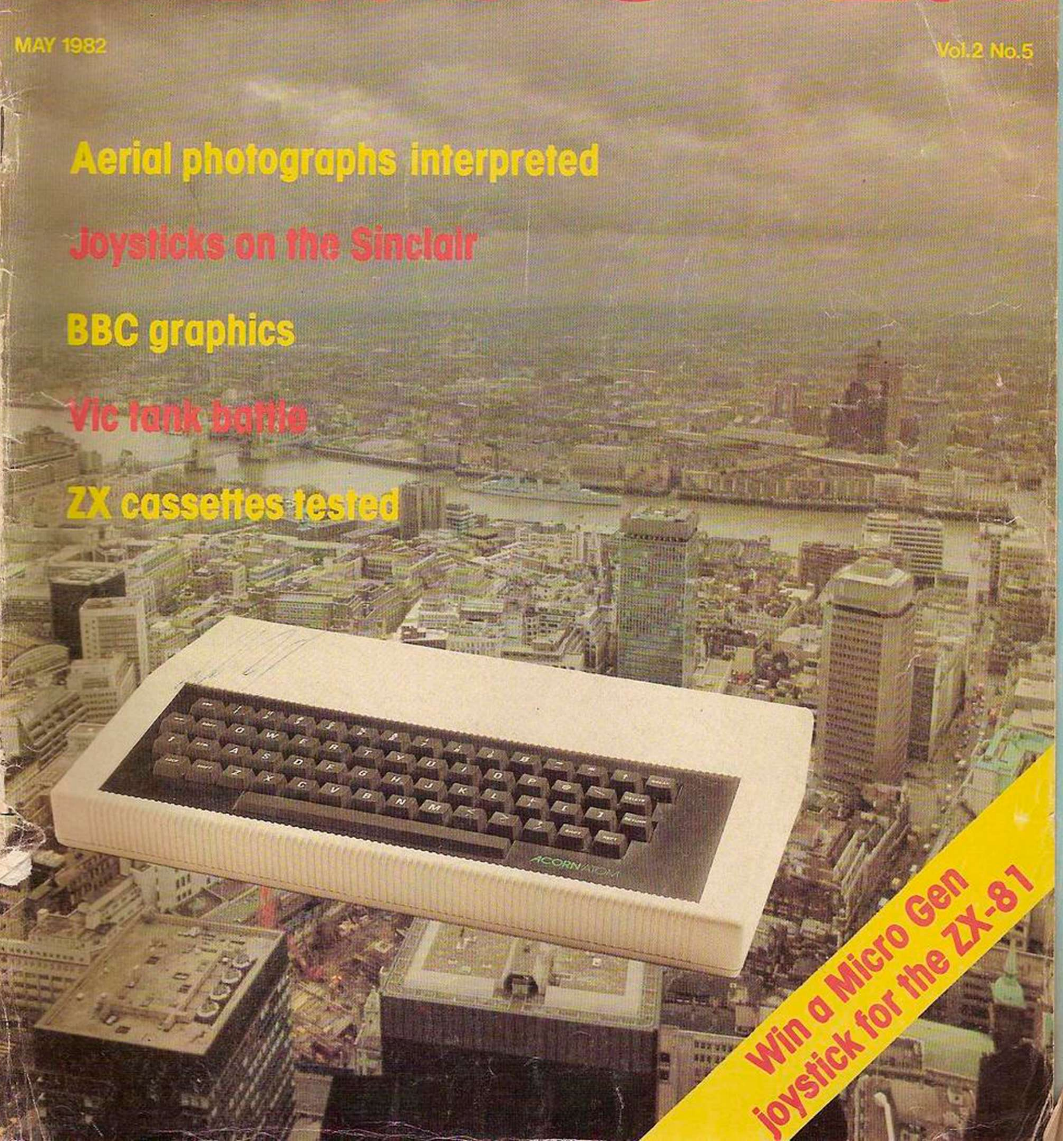
Aerial photographs interpreted

Joysticks on the Sinclair

BBC graphics

Vic tank battle

ZX cassettes tested



Win a Micro Gen joystick for the ZX-81

YOUR COMPUTER

Cover photograph by Stephen Oliver.

Editor
TOBY WOLPE

Assistant Editor
BRENDON GORE

Staff Writer
BILL BENNETT

Sub-editors
MEIRION JONES
JOHN LIEBMANN

Editorial Secretary
LYNN COWLING

Editorial: 01-661 3144

Advertisement Manager
PHILIP KIRBY 01-661 3127

Advertisement Executives
BILL ARDLEY 01-661 3127
PETER RICE 01-661 3127

Midlands Office
DAVID HARVETT 021-356 4838

Northern Office
RON SOUTHALL 061-872 8861

Publishing Director
CHRIS HIPWELL

Your Computer, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS.
Typesetting by In-Step Ltd, London EC1.
Printed by Riverside Press Ltd, Whitstable, Kent.
Subscriptions: U.K. £8 for 12 issues.
©IPC Business Press Ltd 1982
ISSN 0263-0885

Published by IPC Electrical-Electronic Press Ltd, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS. Tel: 01-661 3500 Telex/grams: 892084 BIPRESG.

EDITORIAL

NEXT MONTH, with ZX-81 software on flexidisc offered free with every issue, *Your Computer* is presenting a new idea in program storage to the micro market — an idea which could have a dramatic effect on the currently expensive business of buying software. The flexidisc method eliminates the time-consuming chore of entering a program line by line, and if adopted commercially could reduce manufacturing costs to such a point that micro users would benefit from a fall in program prices to one-quarter of their present levels.

The flexidisc is the size of the ubiquitous 7in. forty-five or single, and is made of pliable plastic. In its grooves, in the form of high- and low-frequency sine waves, it can contain the kind of program that would occupy one side of a conventional software cassette. You transfer the software that the flexidisc contains on to one of your own cassettes simply by playing the disc on your record player and recording it. Once the program is safely committed to cassette the flexidisc is stored away as the master copy, to be brought out only if you wish to record another duplicate. Of course, in next month's *Your Computer* there will be full step-by-step instructions on how to use the disc, plus a thorough account of its workings on the technical level.

When the micro enthusiast, hungry for novel applications for his machine, sets about buying new software it is clearly not the cassette itself that concerns him but the quality of the software it contains. Given that the cassette can satisfy the essential loading requirements, all that it becomes is a container for programs — and compared with the flexidisc, an expensive one at that. Reduce the cost of the container but not the quality of the product held in it, and very soon you find yourself on the brink of a software revolution.

For a better idea of the finances of manufacturing cassette software, we could cite one program-producer who recently revealed that a cassette which sells for £5 costs an estimated 22p to make. For that manufacturer to be able to obtain the same percentage profit margin, the selling price for a flexidisc would be 66p on a production price of 3p.

In the ZX-81's short 1½-year life, software prices for the machine have fallen considerably. Micro users look to commercial software for new ideas, and if the flexidisc means that software becomes even more affordable, new ideas will develop quickly and it will herald a breakthrough which will have an impact on both the user's pocket and the development of micro programming.

NEXT MONTH FREE ZX-81 GAME ON FLEXIDISC

THE ZX–81 UNDER

BUILD YOUR OWN

Joysticks give an added dimension to most games software. David Griffin shows how you can benefit through building your own joysticks for the ZX-81.

JOYSTICKS GIVE an extra novelty to games. Using them is easy compared with the player's usual confrontation with 40 rather meaningless keys. Most hardware additions such as joysticks are available only on larger, more expensive machines, and then only to those fitted with ports — so that would normally rule out the basic ZX-81.

The joystick I shall describe is eight-directional, but after connection to the ZX-81 only four of these directions can be obtained from Basic. If we want to use the other directions, we must resort to a very simple machine-code subroutine.

The joystick is very easy to construct, made from pieces of wood and metal — see figure 1. Cut a hole in the centre of a suitably sized sheet of metal. This hole should be slightly larger than the bolt being used. This sheet is the base of the joystick.

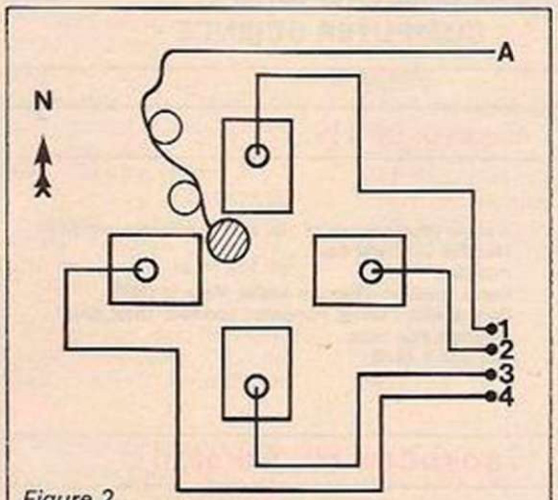
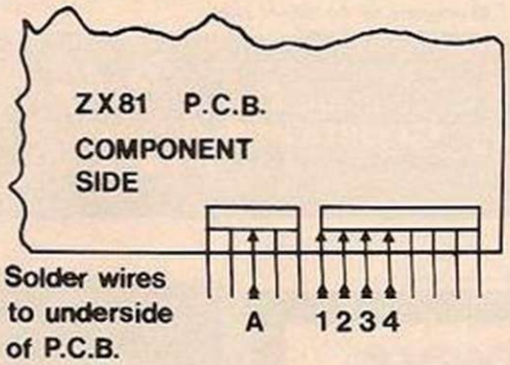


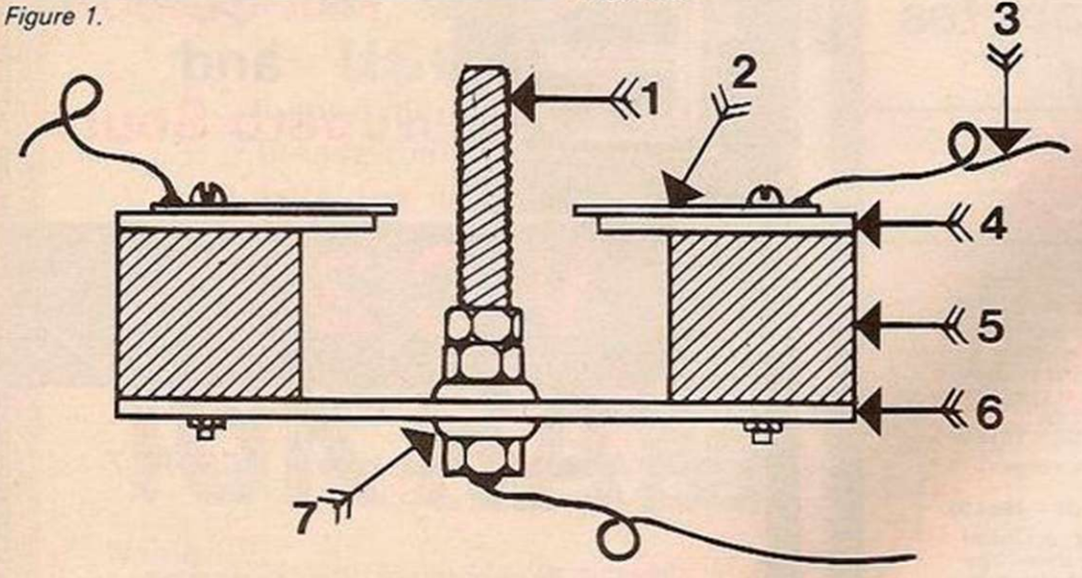
Figure 2.



Solder wires to underside of P.C.B.

Figure 3.

Figure 1.



1 bolt
2 copper plate
3 wire soldered to copper
4 wood with a square hole cut from the centre
5 wooden block
6 metal sheet with a hole cut in the middle
7 old scalectric tyre, or a piece cut from rubber tubing.

Fit an old Scalextric tyre — or a piece cut from rubber tubing — around the bolt, and then into the hole. Screw two nuts around the tyre to hold the bolt in place to give a spring-like effect to the motions of the bolt in the hole. Then position wooden blocks around the bolt and metal sheet, as shown in figure 1. The copper plates can be obtained from old piping if you cannot find any in sheet form. Alternatively, use another good conducter in its place.

Arrange the plates as in figure 2. It will be necessary to use five wires to make the connections to your ZX-81; four of the wires should be soldered to the four copper plates, the fifth to the base of the bolt.

After the joystick has been made, attach it to your computer. To do this, the wires from the joystick must be soldered to the keyboard connections on the printed-circuit board. Remove the back from your ZX-81. If you find this difficult, first locate the three screws under three of the rubber feet supporting it. These feet are held in place by sticky tape and should pull off easily.

There is a ribbon cable joining the keyboard to the printed-circuit board. It is to the underside of these connections that the wires from the joystick will be soldered — see figure 3. The wire A from the bolt is to be soldered to one of the five connections on the left-hand terminal. The four wires 1, 2, 3 and 4 will be soldered to any of the eight connections on the right.

Ideally, a ribbon cable should be soldered to all of the connections, and then an edge connector soldered to the other end. Then, any additions such as the joystick should be fixed to a piece of Veroboard slotted into this connector. It is important that your keyboard is fully functional after these modifications.

The joystick works by connecting two wires of the keyboard terminals to give the same result as if a key had been pressed. Depending on which lines your joystick's wires were soldered to, different results will occur to the following:

```
10 SCROLL
20 PRINT INKEY$
30 GOTO 10
```

Type in this little program and run it, testing each position north, south, east and west to see the response. If you used the same positions as in figure 3 the results should be as follows:

```
3: north
8: south
E: east
D: west
```

As you will have discovered if you tried to push the bolt into one of the diagonals, no result will occur. The reason that the diagonals cannot be used in Basic is that two

JOYSTICK CONTROL

keys cannot be pressed at once to create a response on the screen — that is, except with Shift. The problem can be overcome with a little machine-code subroutine.

```
1 REM xxxxxx
POKE 16514,205
POKE 16515,187
POKE 16516,2
POKE 16517,68
POKE 16518,77
POKE 16519,201
```

This program calls a routine in ROM, which returns the value of the key being pressed. This is used in the statement Inkey$. It also registers when two, three or even more keys are pressed at once. This is of use for our purposes when two keys — the diagonals — are pressed at the same time.

To discover which values correspond to the connections on the printed-circuit board you have chosen, type:

```
 1 REM . . . (machine code)
10 SCROLL
20 PRINT USR 16514
30 GOTO 10
```

and run it, testing each position of your joystick, and noting down the numbers for each. Let:

A = north value
B = north-east value
C = east value
D = south-east value
E = south value
F = south-west value
G = west value
H = north-west value

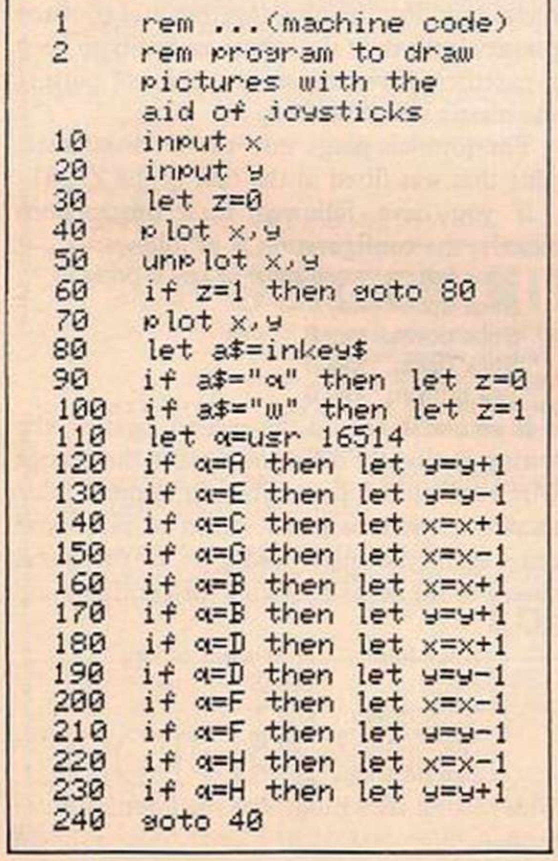
Substitute the letters in lines 120 to 230 of program 1 for these values. Program 1 allows you to draw on the screen. First, the ZX-81 asks for two inputs; the first is the x co-ordinate of the starting point — 0 to 63 — and the second is the y co-ordinate of the starting point — 0 to 43. Then proceed to draw in all the directions of the joystick. To rub out at any time, type W and continue to move in the usual directions. To draw again press Q.

If you have only 1K, you must erase lines 160 to 230, but you will be able to move only in the four main directions.

If you want, you can build a button. This can have many applications in games — for firing lasers, for example. This should be made from a keyswitch, like those used in keyboards. The two connections of this should be fixed to the keyboard terminals, one to each, left and right.

You could also construct an additional joystick. Connect the four wires from the copper plates to any of the connections on the right keyboard terminal. Solder the wire from the bolt to one of the connections of the left terminal. This must be a different one to that used by the first joystick. If you do not take this precaution you will find that the joysticks may register the same result when in certain positions. ■

Program 1.



```
1     rem ...(machine code)
2     rem program to draw
      pictures with the
      aid of joysticks
10    input x
20    input y
30    let z=0
40    plot x,y
50    unplot x,y
60    if z=1 then goto 80
70    plot x,y
80    let a$=inkey$
90    if a$="q" then let z=0
100   if a$="w" then let z=1
110   let α=usr 16514
120   if α=A then let y=y+1
130   if α=E then let y=y-1
140   if α=C then let x=x+1
150   if α=G then let x=x-1
160   if α=B then let x=x+1
170   if α=B then let y=y+1
180   if α=D then let x=x+1
190   if α=D then let y=y-1
200   if α=F then let x=x-1
210   if α=F then let y=y-1
220   if α=H then let x=x-1
230   if α=H then let y=y+1
240   goto 40
```

CONVERTING ATARI STICKS

Patrick Norris explains how you can convert Atari joysticks to work with the ZX-81.

AFTER READING an article by Ian MacLean in the January 1982 issue on interfacing the ZX-81, I conceived the idea of connecting Atari joysticks to the ZX-81. Construction costs are negligible and it took me only one evening to complete the project. Obviously either one or two joysticks are required. I chose Atari sticks because they are robust and, at £6.50 each, not too expensive.

The constructor will require the joysticks, two male and two female in-line five-pin DIN plugs, and about 5ft. of multi-strand wire — the thin, plastic-covered type. Do not use single-strand wire because it is too brittle.

Perhaps surprisingly, the hardest part of the project is making the holes in the side of the ZX-81, to allow access to the male plugs. I achieved this by the judicious use of a wooden dowel, slightly smaller than the outside diameter of the plugs, with emery paper wrapped around it to file the holes to the required size.

There is just enough space to enclose the male plugs under the keyboard. Access to the plugs is on the right-hand side of the case.

Be very careful that the rearmost plug does not touch the printed-circuit board when the two halves of the ZX-81 case are put back together.

Figure 2.

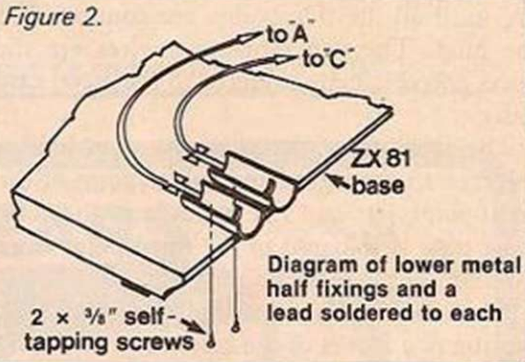


Diagram of lower metal half fixings and a lead soldered to each

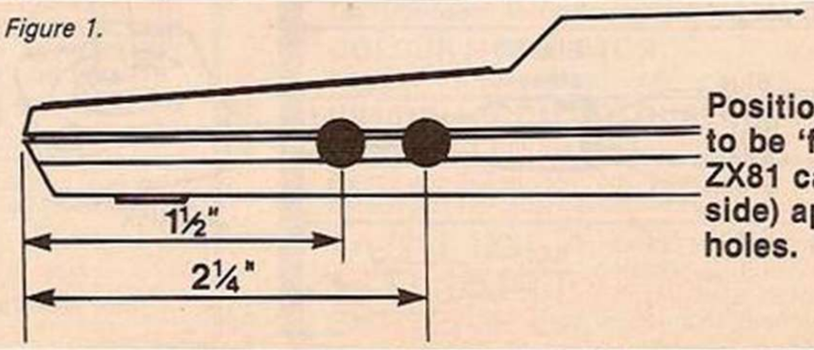
The lower halves of the metal sections of the male plugs were held to the base of the ZX with 3/8 in. self-tapping screws — see figure 2. Solder one wire to the rearmost metal half and one to the foremost.

The next stage is to wire up the two DIN plugs. Take a wire from pin 1 of one plug to pin 1 of the other, pin 2 to pin 2 and so on, until each plug is linked to the other by five wires. Then solder an extra five wires to the plug that will be in the rearmost position — one wire to pin 1, the second wire to pin 2 and so on.

At this stage you should have two plugs

(continued on next page)

Figure 1.



Positions of holes to be 'filed' in the ZX81 case (right hand side) approx. ½" diam holes.

60p

[illegible]OUR COMPUTER

JULY 1982

Vol.2 No.7

Spectrum graphics and sound

Reviews:

ZX-81 colour board

Atom software

Generating BBC sound

Vic memory game

Disassembler for ZX-81

YOUR LETTERS

CHESS FALLACY

I enjoyed John White's article on Chess in May's *Your Computer*. However, there is a fallacy in the article. White gives as one reason for programming in machine code the inability of Basic to perform recursion. Whatever the other reasons for programming in machine code it is certainly not universally true that Basic will not support recursion. Recursion is supported by Sharp Basic on the Sharp MZ-80K. Consider the following simple program, for example:

```
   10 A=0
   20 GOSUB 100
   30 PRINT A
   40 END
  100 A=A+1
  110 IF A<4 THEN GOSUB 100
  120 A=A+A
  130 RETURN
```

The MZ-80K happily runs this program obtaining the correct value of 64 for A. If the Gosub in line 110 is replaced by Goto the final value of A is only 8. The MZ-80K will accept 15 levels of subroutines and it does not seem to matter even if they are all the same one. John's lapse is all the more surprising since his excellent articles suggest that he uses a Sharp MZ-80K.

Alan Stevens,
Alvaston,
Derby.

MISCAST RUNES

While entering "Cast the Runes" from the May edition of *Your Computer*, I found two errors. A semi-colon was missing in line 1050. After the line number a semi-colon is required before LN. Also, line 1130 should read:

```
A=ASC (C$):A=A -64: IF A>26 OR
  A<1 THEN PRINT 4 etc.
```

You printed a zero instead of an 'o' in the word or, which as it was jammed up against another number, made things very confusing. Congratulations on a great magazine. Keep it up.

John Bell,
Quandon,
Derby.

PURSUIT PRAISE

S A Nicholls has struck again! I spent 1½ hours cautiously entering the Hot Pursuit program, from March Software File, in hexadecimal.

The game was a revelation — responsive, fast, flicker free and what a finale. Could I humbly suggest saving the program in segments — e.g., 16514, 16678, 16574, 16835, 16514 and 16971. There are two reasons for this. Firstly it is very tiring keying in with such concentration. Secondly if a lead comes loose, then a backup is near to hand. Name the Saves as "PART", "PART 2", "MAZE". Do not forget that if loading one of the saves, such as "PART 2" that the Let statement in line 10 should be updated, that is Let X = 16836. Prior to saving the last segment change to Slow. Otherwise it is annoying to change it after every load. But this is nit picking — once again congratulations to S A Nicholls.

David Miskimin,
Astley,
Manchester.

TURNING TABLES

I own a 1K Sinclair ZX-81 and was very much impressed by Charles Chambers' article "How to impress with your ZX-81", in May's issue. Although the majority of the programs ran smoothly, I came across a hitch with Program 3 creating the children's multiplication tables. On running, the program provided a series of numbers which continued until the memory was full but not the expected tables. As a result, I turned my hand to writing a simple 1K program for the tables. It runs as follows:

```
   5 REM TABLES *12 BY H CORRIN,
     1982.
  10 LET Y=1
  20 LET X=1
  30 PRINT X;" X ";Y;" = ";X*Y
  40 LET X=X+1
  50 IF X=13 THEN GOTO 70
  60 GOTO 30
  70 PAUSE 300
  80 CLS
  90 LET Y=Y+1
 100 IF Y=13 THEN GOTO 120
 110 GOTO 20
 120 STOP
```

Howard Corrin,
Harrogate,
North Yorkshire.

BBC ROM BUG

I have made a rather odd discovery. The BBC, unlike the Pet or 380-Z, does not set all variables to zero when it is switched on; thus if one types "PRINT A" the machine will reply apologetically "No such variable". However, if one types "LET A = A", without predefining A, it does not give an error message. In fact, if one types "PRINT A" it now contains 0. If this is not intended on Acorn's part, do I have the honour of finding the first, and hopefully the only, bug in the BBC's ROM?

Simon Pick,
Eskdaleside,
North Yorkshire.

KEYBOARD SURVEY

Of the many hundreds of Protos keyboards we have sold, the one reviewed in the June survey was the first to suffer damage in the post. It may be a coincidence, but this was the first to be sent by first-class post.

I would also like to point out some factual inaccuracies in the article. First, the Protos does not require soldering to the ZX-81 board, as may be construed from the fact that it was not included among those that do not. The keyboard is fitted with a copy of the ZX-81 connector which pushes into the ZX-81 keyboard socket. This is not a ribbon cable but an industrial flexible connector costing about 60 times that of ribbon cable.

Secondly, the key legends are not only very legible, but are an enormous improvement on the Sinclair-style ones. The problem of colour blindness is one which is of concern to all manufacturers, but cannot be the over-riding factor in use of colour. The same criticism could be levelled at countless others — including Sinclair with the Spectrum keyboard.

As for being "awkward to use" we have found that the use of an index finger pressed on one of the keytops has the same effect on the Protos keyboard as on any other keyboard. However, as the manufacturers of the key-switches we use assure us that they are good for up to 10 million or so operations you would be able to press your key-switch for about 20 times as long as the majority of other manufacturers' boards made for the ZX-81.

Mike Mepham,
Protos,
Frome, Somerset.

BBC ON SPEC

I wonder if BBC micro owners have noticed the many differences between the hardware and the expensive and detailed BBC specification. The PAL encoded video output will not work unless you add a 470pF capacitor to the board to connect the PAL encoder to the video output circuit. The Break key has not been put on the back as specified but is alongside the function and cursor-control keys, and can be hit accidentally; however Old will restore the status quo, and the key can be programmed as *Key10. The Centronics interface drawing in the manual is inaccurate; the ACK connection having moved two places closer to the Data lines, and the interpreter is in EPROM.

Nevertheless, after a month's hard use I can honestly say that the Beeb is the best and most versatile micro I have used, and very user-friendly. After a hard night's programming you can always fry your breakfast bacon and eggs on the power supply.

Geoff Cox,
Gillingham,
Kent.

ZX-81 SOFTWARE

Last month you did a survey of ZX-81 software, and wereunable to load Monster Maze. What a pity, as it was the finest piece of software in your selection. If I may be so bold as to rate it for you:

A	B	C	D	E	F	G	H
-	3	5	5	-	5	5	5

I noticed an article on the Vic-20 in the May issue, in which the marketing manager said: "75 percent of ZX owners had exhausted the possibilities of their machines". I have been attempting to exhaust the possibilities of my ZX-81 for almost a year. The more I find out, the more there is to know. I thought I was almost familiar with its workings, but since our Cardiff ZX-81 club formed we have discovered ways of scrolling the listings, and producing new graphics.

When my Sinclair Spectrum arrives I intend to interface it to my old ZX-81.

Mike Hayes,
The '81 Club,
Grangetown,
Cardiff.

RESISTANCE

I encountered a few problems with the Resistor Finder on page 79 of the May issue on entering it into a Vic-20.

A resistor of colours Red, Red, Black is 22 ohms, that is 2, 2, MULTIPLIER. ONE and the result was a zero. In the resistor colour code bands 1 and 2, if black are zeros, but if band 3 is black, it is a multiplier of Value 1.

I changed the following lines:

```
270 IF A$ = "BLACK" THEN A = 0
271 IF B$ = "BLACK" THEN B = 0
272 IF C$ = "BLACK" THEN C = 1
 15 PRINT " " (Clear Screen)
195 INPUT "PRESS RETURN";X$
140 )
155 ) Deleted
190 )
196 GOTO 15
```

Line 20 change TAB (165) to: TAB (155) for formatting.

I am new to computing so I hope I have not overlooked anything.

Anthony Roland,
Northwich,
Cheshire.

MONITOR BYTE

There is an error in John Sylvester's machine-code monitor program for the ZX-81 which appeared in the June issue. The byte at address 40B7, listed as 84, should in fact be 8A. With this amendment, the program runs perfectly.

D F Hewin,
Tamworth,
Staffordshire.

JOYSTICK SHORT

I have received several letters about my article "The ZX-81 under joystick control — build your own" in the May issue. They brought to my notice a mistake in figure 1, which will cause the joystick to short.

The four bolts which secure the four copper plates to the joystick should be substituted with screws, which will screw halfway into the wooden blocks, so eliminating the problem. An alternative solution is to insulate the bolts from the upper plates with small rubber washers.

David Griffin,
Warley,
West Midlands. ■