Orginal! Please return soon

# COMPLEMENT AND ADD

October/November 1979 No. 5

Greetings!
First of all, thank you for your interesting correspondance, contributions for the newsletter and membership fees.

The page that will interest most is the review of the S. of C. VDU kit courtesy of Paul Kaufman.

Also, the major worry of MK14 users seems to be the addition of a keyboard, so I have given the wiring diagram of a keyboard to the edge-connector using the September 1978 E.T.I. magazine.

S. of C. have kindly sent a brochure and price list of their current range.I thought it might be nice to include a photo or diagram in the newsletter, just to brighten it up!

## STATUS

The status of the group is that the numbers are steadying out. I am still recieving mail from interesting sources (Even from foreign parts such as Richmond), but I have dropped one or two names either because they requested it or I am sure that the mail is getting lost at their address. For instance someone may have written in February, his name then would have been included on the lists, and then I may never have heard from him again! One of the members has disappeared off to America for a year without telling me, so next year when you are expecting newsletter no. 11, you may get no.3 instead!!

## ACCUMULATOR

Two cassette tape manufacturing companies have written offering club discount of some sort or other; one's prices were abominable (like my typing) and seemed to be more for the PET type user, the other's prices are summarised below:

10% discount to members. (i.e. say when writing that 'you are a memb of the MK14-SC/MP users group')
RACAL-ZONAL C-12's 5 - £3.20, 10 - £5.30, 50 - £23.90, 100 - £45.80.
T.D.K. Head demagnetiser £11.75
Head Cleaner cassette 40p.
Books: Comp. programs that work £2.90, Introduction to Personal & Business Computing £5.00,Micros from Chips to Systems £7.50,

ACCUMULATOR contd.

Programming the 6502 (Sacrilige!!!!!) £8.00
6502 Applications book £9.00
The address is : DJM Services, 82 Hilden Park Road,Hildenborough, Kent, TN11 9BN (0732) 832815
All prices inclusive of everything.
Talking of cassettes, on December 1st (or the 2nd if it is a Sunday) I shall start a circulating cassette (the mind boggles) - all I need is a round Jiffy bag!? No seriously, what I will do is send out a good quality cassette (C60) with a list of people to recieve it and some details (which will increase as the cassette gets filled up) of what is where on the tape.
The standard MK14 cassette routine will be used, so would all those interested shout 'aye' via a letter!
One of the things one never sees is a short pencil! All my pencils disappear into obscurity after their first week of existance, or more likely someone's coat pocket. No, I'm not starting a circulating pencil, (you can send a cassette by post, but a pencil must be lead (led)) but in a fit of madness I ordered a 100 pencils which say on them : 'THIS IS A SC/MP PROGRAMMING PENCIL'.
If anyone wants one, then send me a 10p and a 5p stamp to cover return postage and the cost of the pencil (in that order).

EXTENSION

Clive Isbell, who provided the tempature prog. last time, would like to form an EAST ANGLIAN branch of the MK14 group.

Anyone interested should write to C. Isbell, Trewint House Chelmsford Road, Felsted, Dunmow,Essex CM6 3LS.

Those in the LONDON area may like to meet informally at Martin Stamp's home in Hyde Park, perhaps I'd better rephrase that, no I won't bother, anyway Martin has kindly offered to put up with us for an hour or two at 7.30 on Thurs 8th November.

His address is Flat 2, 9 Pembridge Villas, London, W11 3EW.

Incidentally, wait around for Breadboard Tickets, I will try to get some at a discount for the Club.

Science of Cambridge have said that any club member will get priority on their products, but cannot offer discount.

Any beginners to SC/MP programming might like to read a shrunk down (4 to 2) general note on how to use different addressing modes. If anyone wants a copy then send me a couple of 10p stamps (Not Martin Stamps!) and I will send one.

Also I have some notes on my Maze program published in C.T. earlier this year, and will provide some copies for a couple of 10p stamps.

No circulation List this time, I'm afraid, I'm sure it isn't needed every month.

The two manuals which are soon to be published by S. of C. will be reviewed next time, if I can get some advance issues.

The next newsletter will be sent out early, so that you get them before Breadboard and Christmas! The reason for the latter is that I have been sent some Carols for the 'Music Box' program!

As a supplement to the circulating newsletters, I have put in a contribution from John Leach, to whom I am v. grateful, this being a description of a lights board, or a Target game for the lights board.

Incidentally, the Computing Today which presented my Maze prog. was May 1979, I have just found the issue.

Apologies to anyone if I have forgotton something important, but the newsletter must go on, any more than 3 pages of news, and you are going to get bored!

Oh yes, I have got a listing of a SC/MP assembler, written in Basic, by John Leach. Once again, if anyone wants a copy, then send me 20p worth of stamps, and some praise to John Leach!!

Warning, the Basic is very weird with things like SEG£ and OPEN/CLOSE statements.

Until next newsletter then, goodbye from me, the rest of the newsletter is comprised of individual items.

G.P. 7/10/79

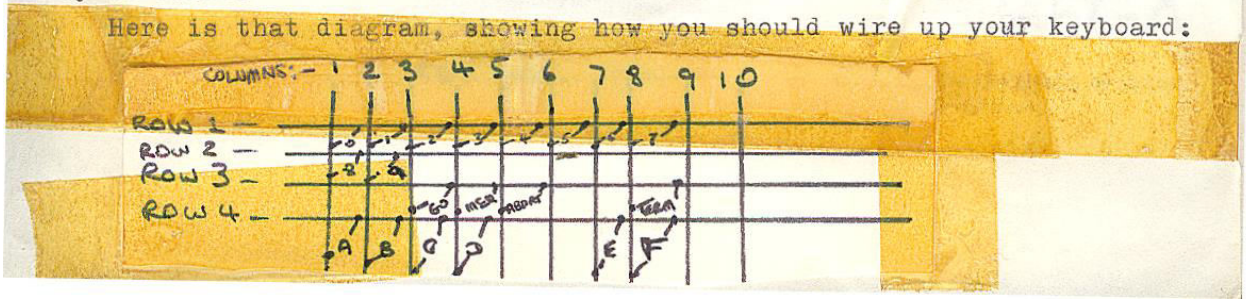
Konnecting Up a Keyboard

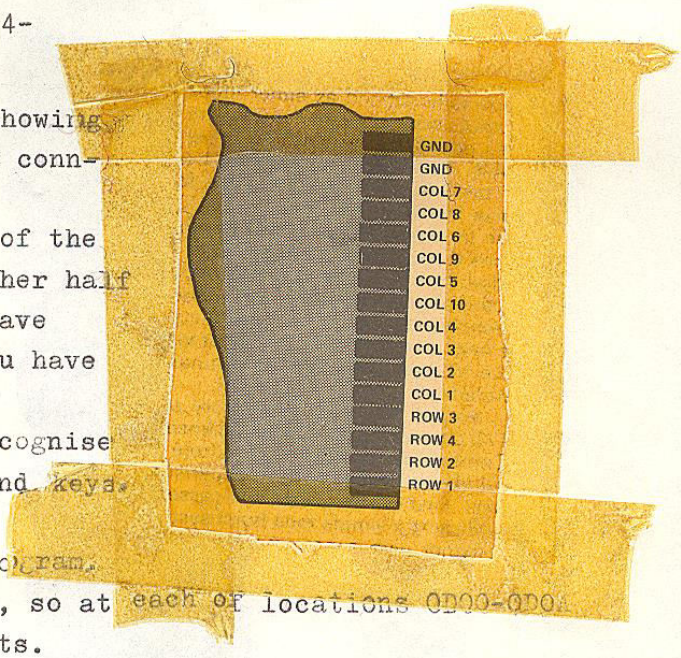
The best way of adding a proper keyboard to the MK14 is to remove all the electronic bits from the keyboard, leaving just the individual switches themselves.

Then, the keyboard should be wired up in columns and rows as per diagram. When I converted my ex-IBM keyboard, I did not worry about getting the right key in the right place, but shuffled the keytops around to suit the wiring.

If you have got an old QWERTY keybd to use, then you can accept up to 40 different key presses (10 columns, 4 rows) and decode them uniquely by software.

Here is that diagram, showing how you should wire up your keyboard:






On the right is a diagram showing the format of the MK14 edge connecter.

So if you group one half of the switches in tens and the other half in groups of 4 , you will have 40 unique keys (assuming you have 40 keys on the keyboard)
Now the monitor can only recognise hex codes (0-F) and 4 command keys.

The rest will have to be interpreted via your own program.
Each column contains 4 rows, so at each of locations 0D00-0D09 you can get 4 possible inputs.
Each row connects to one of the higher bits of the data bus.
Row 1 is the highest (bit 7) to row 4 which is the lowest.
By pressing say row 1, you are connecting 1 bit to 0V, leaving the rest at level 1. So the data value input is going to be 7F for row 1, (0111 1111), BF for row 2, DF for row 3, EF for row 4.

The address you load from is: Column no. - 1 plus 0D00.
So to decide whether ABORT is pressed, load the data value from 0D04 and test for DF.

The other considerations for this are; what happens if the program misses a key press, and how do you know that the key is released?

Well, the usual way of getting a key press, is to wait in a tight loop for the key to change from a value of FF (unpressed) to some other value. Then in the same way, wait in a tight loop until the key reverts back to 'FF'.
The best way to do this is by ILD'ing the location required, and testing for 00 with a JZ (jump zero) instruction.
Beware! You cannot use ILD with an operand of 80 - i.e. using the extension register as a changing offset from 0D00!
If you are scanning the keyboard, then just load the value and exclusive-or with X'FF' then test for zero.
Anyway, enough talk, here is an example to test when the key GO is pressed and to display 'GO' for a while when it is!

(relocatable)

```
0B00 LDI X'0D'  Xpah (1)     C4 0D 35    Set P1 to Keyboard address.
0B03 LDI X'00'  Xpal (1)     C4 00 31          (0D00)
0B06 ILD 0D02          (1)   A9 02       Test if key pressed.
0B08 JZ  0B06                98 FC       If not then try again.
0B0A XOR X'E0'               E4 E0       Test if row 3, (DF plus 1)
```

```
OB0C JNZ OB00          9C F2     If not, then try again
OB0E LDI X'6F'         C4 6F     Load C'G'
OB10 ST  0D02  (1)     C9 02     Store on display.
OB12 DLY 08            8F 08     Delay a while.
OB14 LDI X'3F'         C4 3F     Load C'O'
OB16 ST  0D01  (1)     C9 01     Display for while
OB18 DLY 08            8F 08     Delay.
OB1A DLD OB20          B8 05     loop 256 times using (0B20) as a count
OB1C JNZ OB0E          9C F0     If not zero, goto OB0E.
OB1E JMP 0B00          90 E0     Back again for next time
```

Pattern Search Program  by Terry Wyatt

This is a nice little program which occupies a ridiculously low number of bytes.

```
OF20 Address of memory to be searched (high, low)
OF22 Byte to be looked for
OF23 Byte to replace it.
OF24   start:

OF24  LDI OF XPAH2              C4 OF 36
OF27  LDI 00 XPAL2              C4 00 32
OF2A  LD  (OF20)                C0 F5
OF2C  JZ  (OF44)                98 16
OF2E  ST  (2) E                 CA 0E
OF30  XPAH1                     35
OF31  LD (OF21) ST (2) 0C       C0 EF CA 0C
OF35  XPAH1                     31
OF36  LD@ (1) 1                 C5 01
OF38  XPAH1 ST (OF20)           35 C8 E6
OF3B  XPAH1  XPAL1              35 31
OF3D  ST (OF21) XPAL1           C8 E3 31
OF40  XR OF22 JNZ (OF2A)        E0 E1 9C E6
OF44  LD (1) -1 ST (2) 0D       C1 FF CA 0D
OF48  LDI 01 XPAH3              C4 01 37
OF4B  LDI 3F XPAL3              C4 3F 33
OF4E  XPPC 3                    3F
OF4F  JMP (OF2A)                90 D9
OF51  LD (OF20) XPAH1           C0 CE 35
OF54  LD (OF21) XPAL1           C0 CC 31
OF57  LD (OF23) ST (1) -1       C0 CB C9 FF
OF5B  ST (2) 0D                 CA 0D
OF5D JMP (OF48)                 90 E9
OF5F  END.
```

I had to amend Terry's original program as there was a program bug, in that the high order digit would be 1 too large when the low order byte is FF.

The program searches from the address at OF20,21 until a match with (OF22) is found. Then, if a number key is pressed, the value at OF23 is moved in to replace the contents of this location.

If a command key is depressed, then the search continues to the next byte, until OFFF is reached.

E.g. to change all 08's to 03's from 0B00-0BFF, set 0f20 to 0B )OF21 to 00, OF22 to 08, OF23 to 03.

This program highlights the best method to display numbers using the monitor. Note the unconditional jump at OF4F, this will only be invoked when a command key is pressed, otherwise the jump will be missed.

Label Assembler for SC/MP by G.Phillips.

The following program fits easily into the 0F00 block and allows you to have any jump (JZ, JMP etc) refer to a label rather than an address calculated at program-writing time.
The advantage of this is that if you store your source program, with its labels uncalculated onto tape then if you want to add or remove the odd byte, you don't have to recalculate the jump operands.
It still leaves the problem of PC relative stores, but if you adopt the normal practice of using stack pointers, this should provide no bother.

Coding of Labels

Labels are included as part of the program, and coded as A0 nn, where nn is a unique code representing a label.

Any jumps,whose operand is equal to the nn label, will have their jump displacement calculated, but if no label is found, the jump operand will remain unchanged.

So a simple endless loop could be : A0 77 90 77 meaning Go to label77 Upon compilation, the A0 will be removed, so be careful that you allow for this in your program writing.

Compilation

You can compile up to 4K of a program at a time.
As mentioned above, the compilation will first take a note of all labels (up to 47 labels can be held at any one time)
and then go through converting any instruction : 9X to have the correct displacement.

The A0's will not be carried forward, and the compilation will stop when an '88' is reached.

Usually the program will be written out to the same area as used for the input, but a different area of ram may be specified.

The addresses are set up at: 0F12-3 Start address for output user prog.
0F14-5 Start address for input user prog.

Run from 0F17
If more than 47 labels are likely to be encountered, then change the stack area to somewhere other than 0000.

Program Listing

```
0F12-3 OH,OL                Output Address
0F14-5 IH,IL                Input  Address
0F16                        Count
0F17   LDI 00 XPAH2         C4 00 36
0F1A   LDI 00 XPAL2         C4 00 32
0F1D   LDI 00 ST count      C4 00 C8 F6
0F21   LD OH XPAH1          C0 F2 35
0F24   LD oL XPAL1          C0 F0 31
0F27   ILD count LD@1 1     A8 EE C5 01
0F2B   JP (0F27) XRI 88     94 FA E4 88
0F2F   JZ (0F45)            98 14
0F31   XRI 28 JZ 06         E4 28 98 06
0F35   ILD count Ld@1 1     A8 E0 C5 01
0F39   JMP (0f27)           90 EC
0F3B   LD@1 1 ST@(2) -1     C5 01 CE FF
0F3F   DLD count ST@2 -1    B8 D6 CE FF
0F43   JMP (0F27)           90 E2
0F45   ST@(2) -1            CE FF
0F47   NOP                  08
End of phase 1, all labels stored.
```

```
phase 2.
OF48  LD (OF12) XPAH3 LD (OF13) XPAL3      C0 C9 37 C0 C7 33
OF4E  LD (OF14) XPAH2 LD (OF15) XPAL2      C0 C5 36 C0 C3 32
OF54  LDI 00 ST count                      C4 00 C8 BF
OF58  ILD count LD@2 1                     A8 BD C6 01
OF5C  ST@(3) 1 JP -8                       CF 01 94 F8
OF60  XRI 88  JZ OF9C                      E4 88 98 37
OF64  XRI 28  JZ 2D                        E4 28 98 2D
OF68  ANI F0  XRI 30 JZ 08                 D4 F0 E4 30 98 08
OF6E  ILD count LD@2 1 ST@3 1              A8 A7 C6 01 CF 01
OF74  JMP OF58                             90 E2
OF76  ILD count LDI 00 XPAH1               A8 9F C4 00 35
OF7B  LDI 1 XPAL1                          C4 01 31
OF7E  LD@1 -2  JZ OF8D                     C5 FE 98 0B
OF82  XR(2) 00 JNZ -8                      E2 00 9C F8
OF86  LD(1) -1 SCL CAD count               C1 FF 03 F8 8C
OF8B  JMP 2 LD(2) 00                       90 02 c2 00
OF8F  ST@3 1 LD@2 1 JMP (OF58)             CF 01 C6 01 90 C3
OF95  LD@3 -1 LD@2 1 JMP (OF5A)            C7 FF C6 01 90 BF
OF9B  LDI 00 XPAH3 LDI 02 XPAL3 XPPC 3     C4 00 37 C4 02 33 3F
```

Example

Program to bleep every time go is pressed (sets flag, thus also lights cassette interface led)

```
0B00 A0 01                 label no.1
0B02 C4 0F 07              Load flag
0B05 A0 02                 label no.2
0B07 8F 03                 delay
0B09 B8 20                 count to 256
0B0B 9C 02                 loop back to label 2
0B0D 07 3F 90 01 88        reset flag,return to monitor, then go to 1.
```

This will compile to:

0B00 C4 0F 07 8F 03 B8 20 9C FA 07 3F 90 F3.

Paul Kaufman Tips

If you suffer from slow reset time (sneeze 4 boards mostly), then read on, else skip a few bytes.

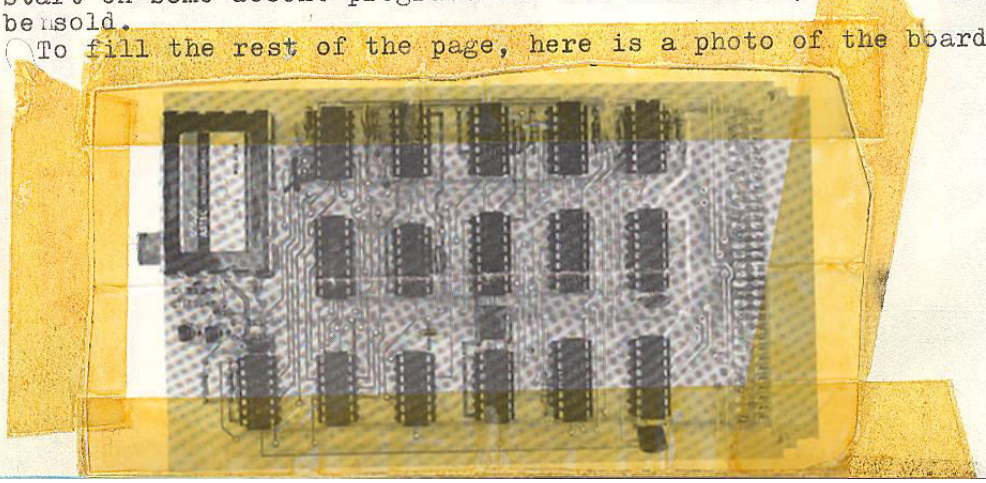
Change R11 to 1.2K; change C6 to .1uF (not electrolytic).

The reset time is now halved

Science of Cambridge VDU kit by Paul Kaufman

Paul must have one of the first VDU kits, when mine arrives I will start on some decent programs for the newsletter, as at £33, a lot will be sold.

To fill the rest of the page, here is a photo of the board.



Review of Science of Cambridge VDU by P. Kaufman

To go with this review, the next newsletter will contain info. to help you add another 1k of memory to the MK14, which will be a necessity for programs greater than 80 bytes! (Editors note)

Finally, after a wait of almost 3 months, my VDU board was delivered. On ripping open the Jiffy bag, I diseovered a pile of I.c.s, resistors, several transistors,PCB and instruction leaflet.

The double-sided glass-fibre PCB is probably the best yet of S.of C.s range - all the holes drilled properly and compnnent positions clearly marked.

For those waiting delivery, here is a list of non-optional extras.

1) 3M of Co-ax. cable with a standard phono plug at one end and a TV type co-ax plug at the other.
2) A 4Mhz crystal for sneeze boards below 5, this replaces the MK14's 4.4Mhz where supplied!!
3) 10 off 14pin i.c. sockets
4) 6 off 16pin i.c. sockets
5) one 4.7K resistor
6) 32 way double sided edge connecter,(unless you are untidy like me and connect things up without connectors- ed.)
7) Couple of yards of ribbon cable (16 way)

Construction s no problem if you keep to the board markings. I suggest mounting the discrete components first, leaving the I.C.s & modulator til last. Be careful with the diode, as its marking easily peels off (?!!) Note at the edge of the board there are two rows of connectors, row 'A' is the outer one, row 'B' is the inner one.Be careful to avoid solder bridges here as you will otherwise get some v. strange results on switch on.

The most consuming task is the wiring up of the data and address lines - be careful to get them right even if it means taking your time. There are several control lines which may be hardwired or connected to the flags. These are used to slect the memory pages to be displayed and the display formats.

I suggest they should be connected as follows:

1) B13 to flag 1 turns vdu on and off
2) B14 to flag 2 alternates between graphics or ASCII.
3) B16 to flag 0 changes Black on white to white on black.

The cassette interface can be left alone.
The VDU displays 32 lines of 16 characters in ASCII mode, or 64 lines of 8 lumps (8 bits to a lump),.
I found the picture to be very stable and exceptionally clear in all modes
Some fiddling was needed to the TV to get the picture central on the screen.
The manual gives 3 useful programs for the VDU and some explanation of it functioning. I didn't get a circuit diagram, but a S.A.E to S. of C. soon got me one. The only serious critisism is that the VDU does not give you an extra memory, and relies on existing memory to function. What this means is that if you want to use the whole screen (two 256 byte areas) you are only left with 80 bytes (hex) to write programs with.

It is relatively easy to add 1k using 2 2114's, even if you decide you don't need the VDU, occupying 0400--7FF. To be prepared for next newsletter, order 2 2114's and a couple (only one for issue 4 or 5) of 74LS00's.

**The Memory extension board** provides the gating to select another 1½K bytes of memory in whatever form is desired, the board has sockets for 2111 ram, 2114 ram and 74S571 PROM. Extra current will be drawn by this module dependent upon the choice of memory device.

**The VDU interface** is connected between the MK14 and the aerial socket of any UHF 625 line domestic television. It presents on the screen a maximum of 512 bytes of MK14 memory, any two ¼K sections of memory may be selected (including the monitor) and the bit mapped display may be represented graphically or in upper case ascii characters. Connections have to be made on the reverse of the MK14 p.c.b. but the VDU module will run from the MK14's own stabilised supply rail. Programs are provided for listing hex programs, moving spots about for video game development etc.

[Re oriented from the next page for OCR to attempt this!]

The following may useful to those with cassette interface problems, as the older C.I. instructions were not so helpful.

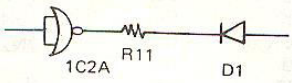
**Extra Notes on Construction**

1) The white band on D1 is the positive end.
2) On some boards two positions are marked for C4. The position in the centre of the board is for C5.
3) The connections for the transistor are as shown below.
4) In some boards the +ve sign for the LEDS is the wrong way round. It should be on the left.
5) The polarity of the LEDS is as shown below.


2N2926
Transistor


LED

6) On later boards there is an extra 27K resistor in series with D1. This improves the noise immunity.




7) Cheap foreign recorders may require an input of about 100mV. To get this larger signal increase R6 to 2K7.
8) If the output level from your tape recorder is less than 200mV try decreasing R1.
9) If data is corrupted to towards 00 signals are too weak and if towards FF then the signals are too strong.
10) C2 must be fitted on the MK14 board. It should be about 20$\mu$F with the +ve sign to the left.
11) This interface is capable of running at teletype speeds.

**The Memory extension board** provides the gating to select another 1½K bytes of memory in whatever form is desired, the board has sockets for 2111 ram, 2114 ram and 74S571 PROM. Extra current will be drawn by this module dependent upon the choice of memory device.

**The VDU interface** is connected between the MK14 and the aerial socket of any UHF 625 line domestic television. It presents on the screen a maximum of 512 bytes of MK14 memory, any two ½K sections of memory may be selected (including the monitor) and the bit mapped display may be represented graphically or in upper case ascii characters. Connections have to be made on the reverse of the MK14 p.c.b. but the VDU module will run from the MK14's own stabilised supply rail. Programs are provided for listing hex programs, moving spots about for video game development etc.

I've also included the latest price list for S.of C.'s accessories.

| | |
|---|---|
| Extra ram (256 bytes) | £ 3.60 plus VAT |
| Ram I/O (8154) | £ 7.80 plus VAT |
| Cassette interface module with software and instructions | £ 5.95 plus VAT |
| Prom programmer | £ 9.95 plus VAT |
| Blank DM72LS571 (two) for programming for dedicated application | £ 5.95 plus VAT |
| Mains power supply unit | £ 4.95 plus VAT |
| V.D.U. interface | £29.00 plus VAT |
| V.D.U. interface less character generator | £23.00 plus VAT |

COMPLEMENT AND ADD

October/November 1979
No. 5
[Scans of original pages follow]

Greetings! First of all, thank you for your interesting correspondence, contributions for the newsletter and membership fees.
The page that will interest most is the review of the S. of C. VDU kit courtesy of Paul Kaufman.
Also, the major worry of MK14 users seems to be the addition of a keyboard, so I have given the wiring diagram of a keyboard to the edge connector using the September 1978 E.T.I. magazine.
S. of C. have kindly sent a brochure and price list of their current range. I thought it might be nice to include a photo or diagram in the newsletter, just to brighten it up!

STATUS
The status of the group is that the numbers are steadying out. I am still receiving mail from interesting sources (Even from foreign parts such as Richmond), but I have dropped one or two names either because they requested it or I am sure that the mail is getting lost at their address. For instance someone may have written in February, his name then would have been included on the lists, and then I may never have heard from him again! One of the members has disappeared off to America for a year without telling me, so next year when you are expecting newsletter no. ll, you may get no.3 instead!

ACCUMULATOR
Two cassette tape manufacturing companies have written offering club discount of some sort or other; one's prices were abominable (like my typing) and seemed to be more for the PET type user, the other's prices are summarised below:

10% discount to members. (i.e. say when writing that you are a member of the MK14-SC/MP users group)
RACAL-ZONAL C-12's 5 - £3.20, 10 - £5.30, 50 - £23.90, 100 - 245.80.
T.D.K. Head demagnetiser £11.75 Head Cleaner cassette 40p
Books: Comp. programs that work £2.90, Introduction to Personal & Business Computing £5.00, Micros from Chips to Systems £7.50,

2)
ACCUMULATOR contd.

Programming the 6502 (Sacrilige!!!!!) £8.00, 6502 Applications book £9.00

The address is : DJM Services, 82 Hilden Park Road ,Hildenborough, Kent, TN11 9BN (0732) 832815
All prices inclusive of everything. Talking of cassettes, on December 1st (or the 2nd if it is a Sunday) I shall start a circulating cassette (the mind boggles) - all I need is a round Jiffy bag!? No seriously, what I will do is send out a good quality cassette (C60) with a list of people to receive it and some details (which will increase as the cassette gets filled up) of what is where on the tape. The standard MK14 cassette routine will be used, so would all those interested shout aye! via a letter!

One of the things one never sees is a short pencil! All my pencils disappear into obscurity after their first week of existence, or more likely someone's coat pocket. No, I'm not starting a circulating pencil, (you can send a cassette by post, but a pencil must be lead (led)) but in a fit of madness I ordered a 100 pencils which say on them : 'THIS IS A SC/MP PROGRAMMING PENCIL'.
If anyone wants one, then send me a 10p and a 5p stamp to cover return postage and the cost of the pencil (in that order).

EXTENSION
Clive Isbell, who provided the temperature prog. last time, would like to form an EAST ANGLIAN branch of the MK14 group. Anyone interested should write to C. Isbell, [Address obliterated]
Those in the LONDON area may like to meet informally at Martin Stamp's home in Hyde Park, perhaps I'd better rephrase that, no I won't bother, anyway Martin has kindly offered to put up with us for an hour or two at 7.30 on Thurs 8th November.
His address is [erased]
Incidentally, wait around for Breadboard Tickets, I will try to get some at a discount for the Club.
Science of Cambridge have said that any club member will get priority on their products, but cannot offer discount.
Any beginners to SC/MP programming might like to read a shrunk down (4 to 2) general note on how to use different addressing modes. If anyone wants a copy then send me a couple of 10p stamps (Not Martin Stamps!) and I will send one.

3)

Also I have some notes on my Maze program published in C.T. earlier this year, and will provide some copies for a couple of 10p stamps.
No circulation List this time, I'm afraid, I'm sure it isn't needed every month.
The two manuals which are soon to be published by S. of c. will be reviewed next time, if I can get some advance issues.
The next newsletter will be sent out early, so that you get them before Breadboard and Christmas! The reason for the latter is that I have been sent some Carols for the 'Music Box' program!
As a supplement to the circulating newsletters, I have put in a contribution from John Leach, to whom I am v. grateful, this being a description of a lights board, or a Target game for the lights board.
Incidentally, the Computing Today which presented my Maze prog. was May 1979, I have just found the issue.

Apologies to anyone if I have forgotten something important, but the newsletter must go on, any more than 3 pages of news, and you are going to get bored!
Oh yes, I have got a listing of a SC/MP assembler, written in Basic, by John Leach. Once again, if anyone wants a copy, then send me 20p worth of stamps, and some praise to John Leach!!

Warning, the Basic is very weird with things like SEG$ and OPEN/CLOSE statements.
Until next newsletter then, goodbye from me, the rest of the newsletter is comprised of individual items.

Konnecting Up a Keyboard
The best way of adding a proper keyboard to the MK14 is to remove all the electronic bits from the keyboard, leaving just the individual switches themselves.
Then, the keyboard should be wired up in columns and rows as per diagram. When I converted my ex-IBM keyboard, I did not worry about getting the right key in the right place, but shuffled the key tops around to suit the wiring.
If you have got an old QWERTY keybd to use, then you can accept up to 40 different key presses (10 columns, 4 rows) and decode them uniquely by software.
Here is that diagram, showing how you should wire up your keyboard:

4)

On the right is a diagram showing the format of the MK14 edge connecter.

So if you group one half of the switches in tens and the other half in groups of 4 , you will have 40 unique keys (assuming you have 40 keys on the keyboard) Now the monitor can only recognise hex codes (0-F) and 4 command keys.
The rest will have to be interpreted via your own program. Each column contains 4 rows, so at each of locations ODOO to 0D0A you can get 4 possible inputs. Each row connects to one of the higher bits of the data bus. Row 1 is the highest (bit 7) to row 4 which is the lowest. By pressing say row 1, you are connecting 1 bit to 0v, leaving the rest at level 1. So the data value input is going to be 7F for row 1, (0111 1111), BF for row 2, DF for row 3, EF for row 4.
The address you load from is: Column no. - 1 plus ODOO. So to decide whether ABORT is pressed, load the data value from ODO4 and test for DF.
The other considerations for this are; what happens if the program misses a key press, and how do you know that the key is released?
Well, the usual way of getting a key press is to wait in a tight loop for the key to change from a value of FF (unpressed) to some other value. Then in the same way, wait in a tight loop until the key reverts back to 'FF'.

The best way to do this is by ILD'ing the location required, and testing for 00 with a JZ (jump zero) instruction. Beware! You cannot use ILD with an operand of 80 - i.e. using the extension register as a changing offset from ODOO! If you are scanning the keyboard, then just load the value and exclusive-or with X'FF' then test for zero. Anyway, enough talk, here is an example to test when the key Go is pressed and to display 'GO' for a while when it is
(relocatable)

```
OBOO LDI X'OD' Xpah (1) C4 OD 35 Set Pl to Keyboard address.
OBO3 LDI X'00' Xpal (1) C4 00 31 (0D00)
OB06 ILD ODO2           A9 02 Test if key pressed.
OB08 JZ OB06            98 FC If not then try again.
OBOA XOR X'EO'          E4 EO Test if row 3, (DF plus 1)
```

5)

```
0BOC JNZ OBOO           9C F2 If not, then try again
OBOE LDI X'6F'          C4 6F load C'G
OB10 ST ODO2            C9 02 store on display
OB12 DLY 08             8F 08 Delay a while
OB14 LDI X'3F'          C4 3F load c'O'
OB16 ST OD01            C9 01 Display a while
OB18 DLY 08             8F 08 Delay
OB1A DLD OB20           B8 05 Loop 256 times using B20 as count
OBIC JNZ OBOE           9C F0 if not zero go to 0B0E
OB1E JMP OBOO           90 E0. Back again for next time
```

Pattern Search Program
by Terry Wyatt
This is a nice little program which occupies a ridiculously low number of bytes.
OF20 Address of memory to be searched (high, low)
OF22 Byte to be looked for
OF23 Byte to replace it.

```
0F24 start:
OF24 LDI OF XPAH2         C4 0F 36
0F27 LDI 00 XPAL2         C4 00 32
OF2A LD (OF20)            C0 F5
OF2C JZ (OF44)            98 16
OF2E ST (2) E             CA 0E
OF30 XPAHI                35
OF31 LD (OF21) ST (2) OC. C0 EF CA 0C
OF35 XPAL1                31
OF36 LD @(1) 1            C5 01
OF38 XPAHI ST (OF20)      35 C8 E6
OF3B XPAHI XPALI          35 31
```

```
OF3D ST (OF21) XPAL1      C8 E3 31
OF40 XR OF22 JNZ (OF2A)   E0 E1 9C E6
OF44 LD (1) -1 ST (2) OD  C1 FF CA 0D
OF48 LDI 01 XPAH3         C4 01 37
OF4B LDI 3F XPAL3         C4 3F 33
OF4E XPPC 3               3F
OF4F JMP (OF2A)           90 D9
OF51 LD (OF20) XPAH1      C0 CE 35
0F54 LD (OF21) XPAL1      C0 CC 31
OF57 LD (OF23) ST (1) -1  C0 CB C9 FF
OF5B ST (2) OD            CA 0D
OF5D JMP (OF48)           90 E9
OF5F END.
```

I had to amend Terry's original program as there was a program bug, in that the high order digit would be 1 too large when the low order byte is FF.
The program searches from the address at OF20,21 until a match with (OF22) is found. Then, if a number key is pressed, the value at OF23
is moved in to replace the contents of this location. If a command key is depressed, then the search continues to the next byte, until OFFF is reached. E.g. to change all 08's to 03's from OBOO-OBFF, set Of 20 to OB OF21 to 00, OF22 to 08, OF23 to 03. This program highlights the best method to display numbers using the monitor. Note the unconditional jump at OF4F, this will only be invoked when a command key is pressed, otherwise the jump will be missed.

6)
Label Assembler for SC/MP  by G. Phillips.

The following program fits easily into the OF00 block and allows you to have any jump (JZ, JMP etc) refer to a label rather than an address calculated at program-writing time. The advantage of this is that if you store your source program, with its labels uncalculated onto tape then if you want to add or remove the odd byte, you don't have to recalculate the jump operands. It still leaves the problem of PC relative stores, but if you adopt the normal practice of using stack pointers, this should provide no bother.

Coding of Labels

Labels are included as part of the program, and coded as AO nn, where nn is a unique code representing a label. Any jumps, whose operand is equal to the nn label, will have their jump displacement calculated, but if no label is found, the jump operand will remain unchanged.
So a simple endless loop could be : A0 77 90 77 meaning Go to label77 Upon compilation, the AO will be removed, so be careful that you allow for this in your program writing.

Compilation
You can compile up to 4K of a program at a time. As mentioned above, the compilation will first take a note of all labels (up to 47 labels can be held at any one time) and then go through converting any instruction : 9X to have the correct displacement.

The AO's will not be carried forward, and the compilation will stop when an '88' is reached.
Usually the program will be written out to the same area as used for the input, but a different area of ram may be specified. The addresses are set up at:
OF12-3 Start address for output user prog.
OF14-5 Start address for input user prog.
Run from OF17

If more than 47 labels are likely to be encountered, then change the stack area to somewhere other than 0000.

Program Listing

```
OF12-3 OH, OL Output Address
OF14-5 IH, IL Input Address
OF16 Count
OF17 LDI 00 XPAH2          C4 00 36
OFIA LDI 00 XPAL2          C4 00 32
OFID LDI 00 ST count       C4 00 C8 F6
OF21 LD OH X???1           CO F2 35
OF24 LD OL XPALI           CO FO 31
OF27 ILDI Count LD@l 1     A8 EE C5 01
OF2B JP (OF27) XRI 88      94 FA E4 88
OF2F JZ (OF45)             98 14
OF31 XRI 28 JZ 06          E4 28 98 06
OF35 ILD count Id@l 1      A8 EO C5 01
OF39 JMP (OF27)            90 EC
OF3B LD@1 1 STO(2)         C5 01 CE FF
OF3F DLD count ST@2 -1     B8 D6 CE FF
OF43 JMP (OF27)            90 E2
OF45 ST@(2) -1             CE FF
OF47 NOP                   08
End of phase 1, all labels stored.
```

7)

phase 2.

```
OF48 LD (OF12) XPAH3 LD (OF13) XPAL3       C0 C9 37 C0 C7 33
OF4E LD (OF14) XPAH2 LD (F15) XPAL2        C0 C5 36 C0 C3 32
OF54 LDI 00 ST count                       C4 00 C8 BF
OF58 ILD count LD@2 1                      A8 BD C6 01
OF5C ST@(3) 1 JP -8                        CF 01 94 F8
OF60 XRI 88 JZ OF9C                        E4 88 98 37
OF64 XRI 28 JZ 2D                          E4 28 98 2D
OF68 ANI FO XRI 30 JZ 08                   D4 F0 E4 30 98 08
OF6E ILD count LDQ2 1 ST@3 1               A8 A7 C6 01 CF 01
OF74 JMP OF58                              90 E2
OF76 ILD count LDI 00 XPAHI                A8 9F C4 00 35
OF7B LDI 1 XPAL1                           C4 01 31
OF7E LD@1 -2 JZ OF8D                       C5 FE 98 0B
OF82 XR(2) 00 JNZ -8                       E2 00 9C F8
OF86 LD 1) -1 SCL CAD count                C1 FF 03 F8 8C
OF8B JMP 2 LD(2) 00                        90 02 C2 00
OF8F STQ3 1 LD@2 1 JMP (OF58)              CF 01 C6 01 90 C3
OF95 LDQ3 -] LD@2 1 JMP (OF5A)             C7 FF C6 01 90 BF
OF9B LDI 00 XPAH3 LDI 02 XPAL3 XPPC 3      C4 00 37 C4 02 33 3F
```

Example Program to bleep every time go is pressed (sets flag, thus also lights cassette interface led)

```
OBOO AO 01     label no.1
OBO2 C4 OF 07  load flag
OB05 AO 02     label no.2
OBO7 8F 03     delay
OBO9 B8 20     count to 256
OBOB 9C 02     loop back to label 2
OBOD 07 3F 90 01 88  reset flag, return to monitor, then go to 1.
```

This will compile to: OBOO C4 OF 07 8F 03 B8 20 9C FA 07 3F 90 F3.

Paul Kaufman Tips

If you suffer from slow reset time (sneeze 4 boards mostly), then read on, else skip a few bytes.
Change Rll to 1.2K; change c6 to 0.1uF (not electrolytic). The reset time is now halved

Science of Cambridge VDU kit by Paul Kaufman

Paul must have one of the first VDU kits, when mine arrives I will start on some decent programs for the newsletter, as at £33, a lot will be sold.

To fill the rest of the page, here is a photo of the board.

8)

-8
Review of Science of Cambridge VDU
by P. Kaufman

To go with this review, the next newsletter will contain info. to help you add another lk of memory to the MK14, which will be a necessity for programs greater than 80 bytes! (Editors note)

Finally, after a wait of almost 3 months, my VDU board was delivered. On ripping open the Jiffy bag, I discovered a pile of ICs, resistors, several transistors, PCB and instruction leaflet.
The double-sided glass-fibre PCB is probably the best yet of S.of C.s range - all the holes drilled properly and component positions clearly marked. For those waiting delivery, here is a list of non-optional extras:

1) 3M of Co-ax. cable with a standard phono plug at one end and a TV type co-ax plug at the other.
2) A 4Mhz crystal for sneeze boards below 5, this replaces the MK14's 4.4Mhz where supplied!!
3) 10 off 14pin i.c. sockets
4) 6 off 16pin i.c. sockets
5) one 4.7K resistor
6) 32 way double sided edge connecter, (unless you are untidy like me and connect things up without connectors- ed.) (7) Couple of yards of ribbon cable (16 way)
Construction s no problem if you keep to the board markings. I suggest mounting the discrete components first, leaving the ICs & modulator til last. Be careful with the diode, as its marking easily peels off (?!!) Note at the edge of the board there are two rows of connectors, row 'A is the outer one, row 'B' is the inner one. Be careful to avoid solder bridges here as you will otherwise get some v. strange results on switch on.
The most consuming task is the wiring up of the data and address lines - be careful to get them right even if it means taking your time. There are several control lines which may be hardwired or connected to the flags. These are used to select the memory pages to be displayed and the display formats.
I suggest they should be connected as follows:
1) B13 to flag 1 turns vdu on and off
2) B14 to flag 2 alternates between graphics or ASCII.
3) B16 to flag O changes Black on white to white on black.
The cassette interface can be left alone.
The VDU displays 32 lines of 16 characters in ASCII mode, or 64 lines of 8 lumps (8 bits to a lump), . I found the picture to be very stable and exceptionally clear in all modes.

Some fiddling was needed to the TV to get the picture central on the screen. The manual gives 3 useful programs for the VDU and some explanation of it

functioning. I didn't get a circuit diagram, but a S.A.E to S. of C. soon got me one. The only serious criticism is that the VDU does not give you an extra memory, and relies on existing memory to function. What this means is that if you want to use the whole screen (two 256 byte areas) you are only left with 80 bytes (hex) to write programs with,
It is relatively easy to add lk using 2 2114's, even if you decide you don't need the VDU, occupying 0400--7FF. To be prepared for next newsletter, order 2 2114's and a couple (only one for issue 4 or 5) of 74LSOO's.

9)

The Memory extension board provides the gating to select another 1.5K bytes of memory in whatever form is desired, the board has sockets for 2111 ram, 2114 ram and 745571 PROM. Extra current will be drawn by this module dependent upon the choice of memory device.
The VDU interface is connected between the MK14 and the aerial socket of any UHF 625 line domestic television. It presents on the screen a maximum of 512 bytes of MK14 memory, any two 1/2K sections of memory may be selected (including the monitor) and the bit mapped display may be represented graphically or in upper case ascii characters. Connections have to be made on the reverse of the MK 14 p.c.b. but the VDU module will run from the MK14's own stabilised supply rail. Programs are provided for listing hex programs, moving spots about for video game development etc.

The following may useful to those with cassette interface problems, as the older C.I. instructions were not so helpful.
Extra Notes on Construction
1) The white band on D1 is the positive end.
2) On some boards two positions are marked for C4. The position in the centre of the board is for C5.
3) The connections for the transistor are as shown below.
4) In some boards the +ve sign for the LEDS is the wrong way round. It should be on the left.
5) The polarity of the LEDS is as shown below.
6) On later boards there is an extra 27 K resistor in series with D1.This improves the noise immunity.
7) Cheap foreign recorders may require an input of about 100mV. To get this larger signal increase R6 to 2K7.
8) If the output level from your tape recorder is less than 200mV try decreasing R1.
9) If data is corrupted to towards 00 signals are too weak and if towards FF then the signals are too strong.
10) C2 must be fitted on the MK 14 board. It should be about 20uF with the +ve sign to the left.
11) This interface is capable of running at teletype speeds.

I've also included the latest price list for S.of C's accessories.

| | |
|---|---|
| Extra ram (256 bytes) | £ 3.60 plus VAT |
| Ram 1/0 (8154) | £ 7.80 plus VAT |
| Cassette interface module with software and instructions | £5.95 plus VAT |
| Prom programmer | £9.95 plus VAT |
| Blank DM72LS571 (two) for programming for dedicated application | £5.95 plus VAT |
| Mains power supply unit | £4.95 plus VAT |
| V.D.U. interface | £29.00 plus VAT |
| V.D.U. interface less character generator | £23.00 plus VAT |