

With thanks to H.P. Diehl and
H.D. De Mülder

junior cookbook

a few healthy recipes to keep your computer in shape

Our Elektor staff desks are covered in piles of letters, the telephone never stops ringing and even telexes are streaming in by the dozen. If this carries on, the offices will have to be evacuated . . . And all because of the Junior Computer! Book One whetted people's appetites to such an extent that they just cannot wait for Book Two; they are craving to try out more Junior recipes . . . NOW!

Far be it from us to let anyone starve, so here are a few hints, alternatives and other useful 'brain food' to keep you and your computer going until the main course, Book Two, arrives.

Menu 1: Decimal arithmetic

As you will remember from Book 1, subtractions and additions may be carried out both in binary and in decimal. Whenever instruction SED (F8) for decimal appears in the program, problems will arise if the computer jumps back to the monitor (IC00) via a BRK at the end of the program or when operating in single-step mode. What

increment it as often as necessary by depressing the + key function (key C!). . . Like going from London to Liverpool via Newcastle. But how can this happen? During the monitor subroutine GETKEY the key value is determined by adding 07 nought times, once or twice to a basic value. Everything works out fine provided this takes place in binary, but when the computer tries to do it in decimal the key codes get mixed up. One solution to the problem is to make sure that the SAVE routine of the monitor contains the instruction CLD (opcode D8), like the RESET routine.

happens is that if you attempt to return to the monitor when the D flag is 1, keys F, +, AD, DA, PC and GO will have become ineffective; the key functions A . . . F will be lost. The key function AD will be taken over by key A, B has become DA, C is now the + function, D is GO and E will assume PC's function. In other words, the numeric key functions A . . . F are no longer available. Addresses containing A . . . F cannot be entered directly, but only by means of a slight detour. The solution is to type in the nearest lower address below the one required that uses decimals 0 . . . 9 only, and

This will do no harm to the program, as the D=1 situation is kept in the saved P register (00F1) and it will be restored upon the computer's return from the monitor (section GOEXEC). This will mean changing the EPROM as follows:

```
1C1A 4C 32 1C JMP-START
1C31 78      SEI
1C32 D8      CLD D=0 START (tempo-
              rary) binary arithmetic
```

As a result, the central START section of the monitor will now begin with CLD. After SAVE the machine will 'convert' back to binary and all keys will retain their normal functions. By the way,

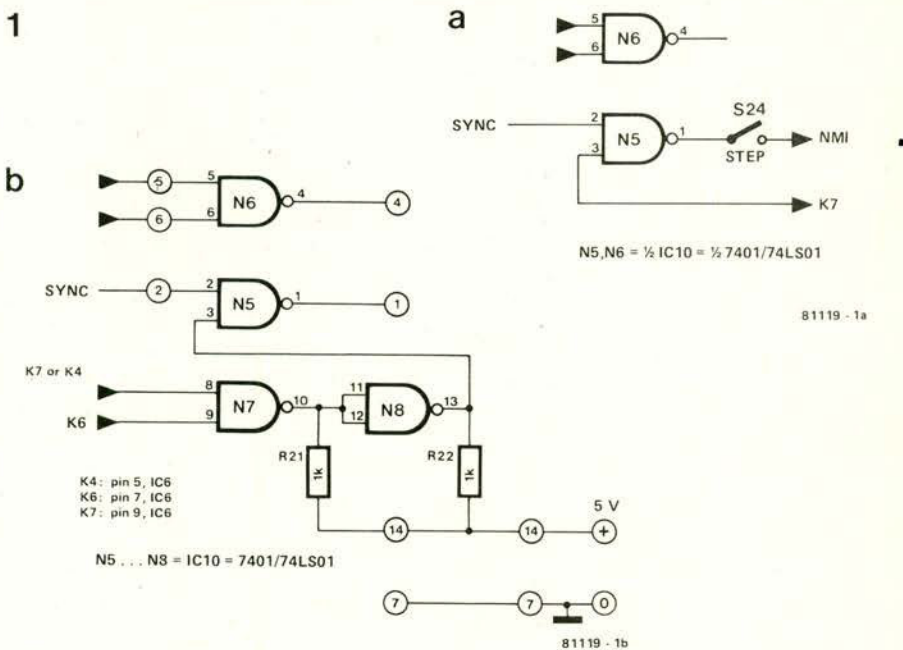


Figure 1. The hardware necessary to hold up the step-by-step procedure whenever the computer is either in the monitor (1a) or inside one of the two memory ranges activated by the two chip select signals, K4 or K7/K6 respectively (1b).

there are no plans to start supplying modified EPROMS, as there are several other alternative solutions to the problem.

It might be useful at this stage to look at an example of decimal arithmetic. For this we will use the adapted addition program given on page 68 in Book 1:

```
0100 18      CLC
0101 A9 13 LDA # 13
0103 F8      SED      D=1 decimal arith-
                    metic
0104 69 08 ADC 08
0106 D8      CLD      D=0 binary arithmetic
0107 00      BRK
1A7E 00      IRQ vector points to monitor
1A7F 1C
```

Once the start address has been entered and GO has been depressed, the program is run; it leads to a jump to the monitor. The keys may now be used as normal, as after the addition the computer went back to binary. Typing in address 00F3 will give the result of the addition (21).

Alternatively:

```
0100 18      CLC
0101 A9 13    LDA # 13
0103 F8      SED      D=1 decimal arithmetic
0104 69 08    ADC 08
0106 00      BRK
1A7E 00      IRQ vector points to 1A00
1A7F 1A
1A00 D8      CLD      D=0 binary arithmetic
1A01 4C 00 1C JMP-SAVE jump to monitor
```

At the end of the program the BRK will bring the computer to 1A00 by way of the IRQ jump vector. After going back to binary it will jump to the monitor. If the program is to be run in one go, the method we have just described is not so suitable; if it is stepped through, however, from 0103, this is in fact the only feasible method.

With reference to the latter (step mode), a few aspects have to be considered. The hardware will have to be modified, for instance. This is because it is prohibited to step through the monitor. The monitor's task involves executing a large number of instructions

in a continuous cycle (display multiplexing, waiting for a key to be depressed, etc.). This explains gate N5 in the circuit diagram on page 14/15 in Book 1. Provided signal K7 is high (EPROM is not addressed) an NMI will occur after every SYNC pulse (generated during the op-code phase of an instruction). Once the current instruction has been processed, this will cause the computer to jump to the monitor (provided the NMI jump vector is pointing to 1C00). If, on the other hand, K7 is low (monitor is addressed), no NMI will occur.

The forthcoming expansion possibilities include a printer monitor comprising the address range 1000...13FF. This is selected by K4. The printer monitor may not be stepped through either; in other words, the circuit around N5 will have to be expanded. The details are given in figure 1b. Now there are two ways in which to block an NMI via SYNC. Either K4 or K7 will be used for

2

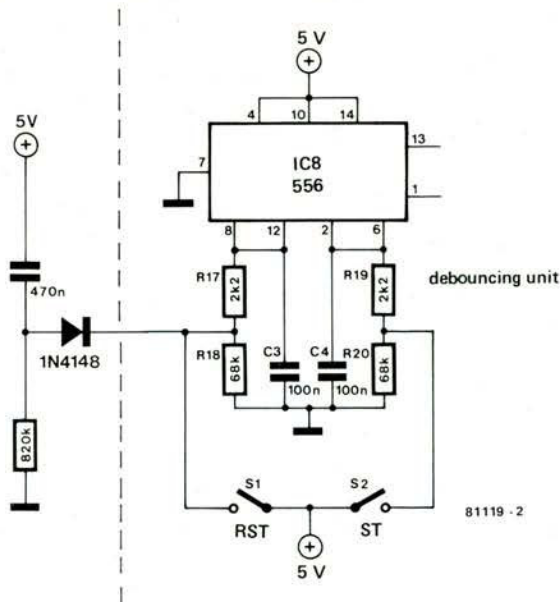


Figure 2. The 'automatic alarm clock', formed by the addition of three components, to wake up the Junior Computer when it is to be started automatically (RESET) after being switched on.

3

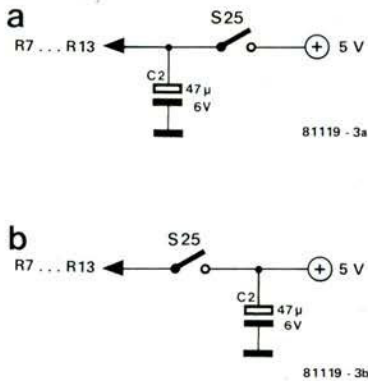


Figure 3. When the positive end of C2 is moved to another place on the board, causing the display on and off will not cause the memory contents to be lost.

STOP/NMI key. The NMI jump vector (1A7A and 1A7B) is now pointed at address 1A00, where the following program is located:

MIN	1A00	48	PHA	save accumulator on stack	
1A01	A5	FA	LDAZ-POINTL	fetch low order address byte	
1A03	D9	02	BNE ADL	on to ADL if A≠00000000	
1A05	C6	FB	DECZ-POINTH	decrement POINTH by 1	
ADL	1A07	C6	FA	DECZ-POINTL	decrement POINTL by 1
1A09	68		PLA	restore A to original value	
1A0A	40		RTI	return to monitor	

Thus, depressing the STOP/NMI key during the monitor allows the address in the display to be decremented by one. It is now also possible to enter data via the DA key in the reverse order, that is to say, per decremented address, as the minus key, like the plus key, operates irrespective of whether the machine is in the data or in the address mode.

Menu 3: Automatic start

Depress the RST key as soon as the Junior Computer is switched on and the monitor will be ready for use. Using the circuit designed by Mr. H. Diehl which is shown in figure 2, the RST does not need to be depressed, because the computer is reset automatically. It does mean adding another three components to the main board, but these should be able to be squeezed in near the RST key.

Menu 4: Display switch

The display switch (S25 in the main circuit diagram on pages 14/15 in Book 1) enables the display to be switched off. If, for instance, you've been having a 'computer session' until past midnight and you'd like to call it a day, but wish to continue on the following evening without having to type in all the data again, the Junior Computer may be left on, but the display must be switched off. Leaving

one signal and K6 for the other. (Figure 1b itself will be dealt with later). The point is here that by connecting K6 a program in page 1A cannot be stepped through either. The decimal addition program mentioned previously can now be stepped through.

Menu 2: Add a minus key

It is common knowledge that the address to be displayed during the monitor routine can be incremented by one by depressing the plus key. This can be done as often as is required. When checking a program, the user may well like to know what the previous address contained.

On such occasions it would be an advantage to have a *minus key* function. This can be achieved by using the

L \ H	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4																
5																
6																
7																

81119-4

Figure 4. A useful table showing the full 128 segment pattern possibilities.

the display 'on' might have a detrimental effect on its lifespan. When ready for another session, the display is switched on again. (Obviously, this will all be superfluous, once the cassette interface can be connected and all the data can be stored on tape).

It rarely happens, but something could go wrong when the display is switched on again. Capacitor C2 (figure 3a) must then be recharged to top supply level. Closing S25 can in some cases lead to a considerable spike on the +5 V supply lines. As a result, the RAM memory contents, so carefully preserved until then, may be lost. Then everything will depend on *your* memory whether the damage can be undone!

This headache can be remedied by connecting the positive end of C2 to the other side of S25. C2 is shown on the board next to the S25 connections.

Menu 5: Another Junior text dish

In response to the 'text display on the Junior Computer' article published last month in Elektor, Mr. H. De Mulder has kindly sent us a useful table (see figure 4). This presents the full 128 different possible combinations of segment patterns. Patterns in the same row have

an identical most significant nibble (H) and patterns in the same column all have an identical least significant nibble (L). What it comes down to is that the bit corresponding to a segment that is lit must be *zero*

Menu 6: A few constructive recipes

To start with, let's look at the pin assignments of the ICs belonging to the main board (top view) and to the seven segment displays, as shown in figure 5. The pin assignments of the expansion connector and the port connector were already dealt with in Appendix 4 of Book 1. They are useful aids to have when servicing the computer or when tracking down errors.

Several thousand Junior Computers have been built by now, both in the UK and abroad, and so far remarkably few problems have come to light. This does not mean, however, that the Junior Computer is fool-proof, as the following description plus photographs illustrates.

The mind boggles . . .

Does anything in these photographs strike you as odd? Is it a U.F.O.

perhaps? Take another swig, readjust your glasses and look again. Surely not . . . ???!! Yes, right second time. It's a slightly unusual Junior Computer power supply, seen from a 'different' angle. It was sent to our editorial staff for 'repair' by a reader who invested a fair amount of artistic ingenuity and good faith into its construction. The work of art which is currently on show at the Museum of Modern Art, New York, will not work for some reason. However, it is full of surprises:

- The unconventionally placed heat sink is a real treat (or threat?) to the eye. Being rather ebullient in nature, it was deemed necessary to 'pin' it down lest it should escape — heat tending to rise. The nails chosen for the purpose are highly effective, as they are 10 cm long and ½ cm in diameter. This however did not satisfy the artist who thought 'better safe than sorry' and soldered them to the board. As a result, the heat sink now maternally screens the voltage stabiliser IC, LM 309, from all outside influence.

- IC1 (79L12) unhappily tried to take his (78L12) brother's rightful place on the board. For this heinous offence he was executed on the spot.

1

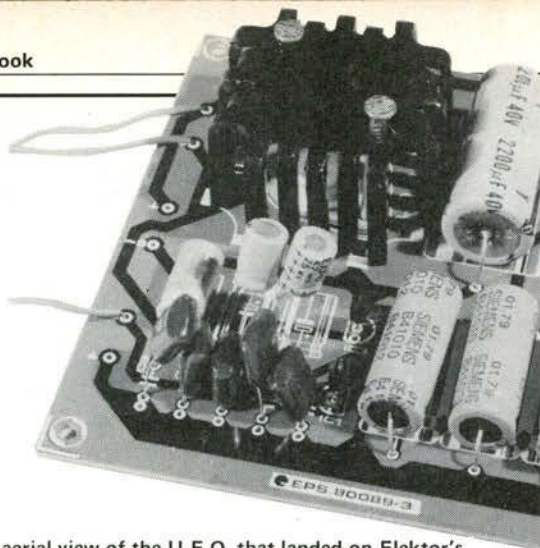


Figure 6. An aerial view of the U.F.O. that landed on Elektor's doorstep, seen from the east at a safe distance.

2

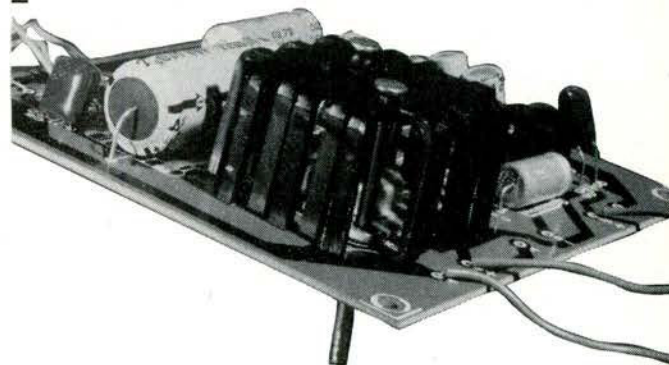


Figure 7. The same mysterious object viewed from the south-west.

5

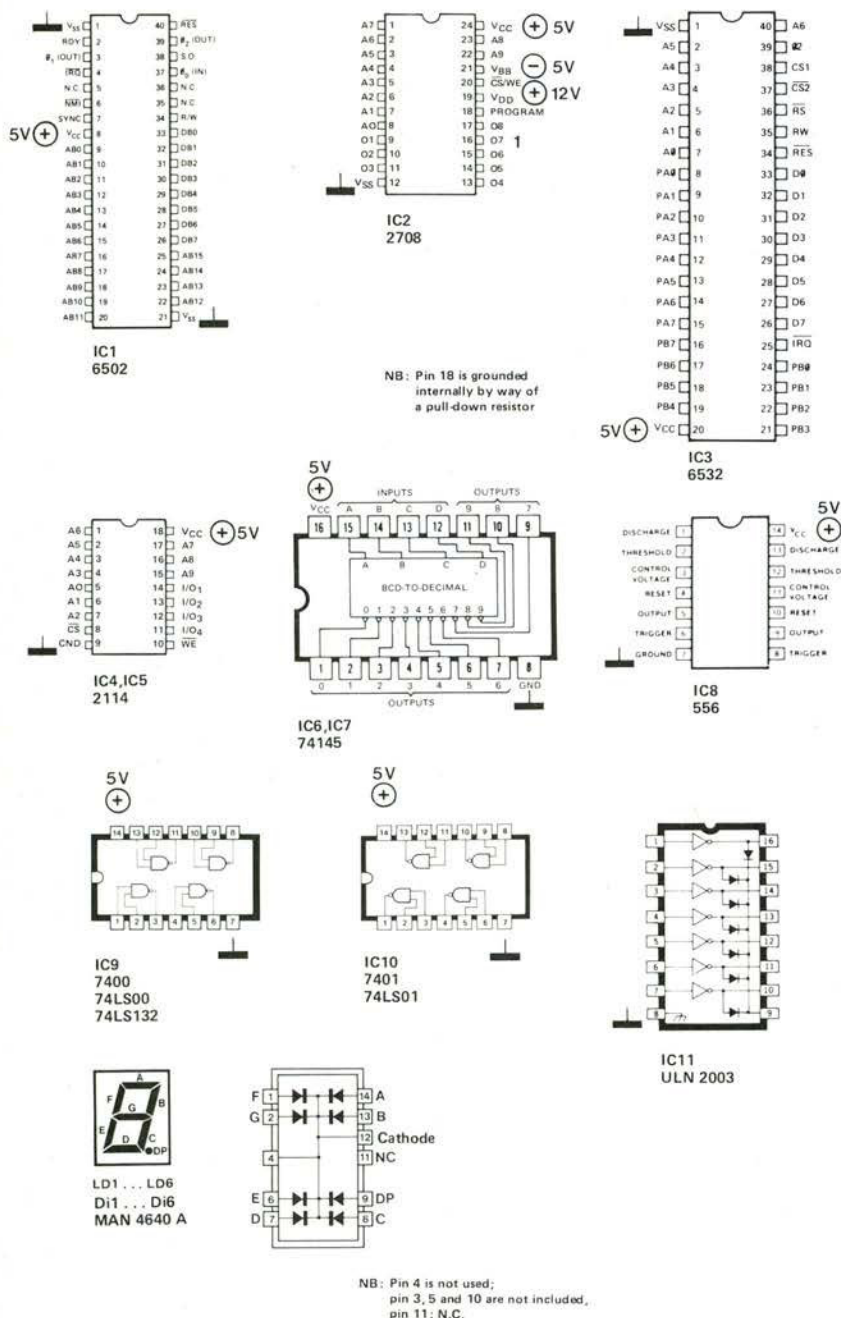


Figure 5. The pinning (top view) of all the ICs used on the Junior Computer's main and display boards.

● As can be seen, the dark capacitor family is very close and has such high aspirations that it now hovers some 10 mm above the board. Seriously though, we all make mistakes at some time or another... Regrettably, refusing to ask for advice means components have to die an untimely (and expensive!) death. This can be avoided by ringing up 'Technical Queries'. Why muddle through and let yourselves in for unpleasant and often explosive shocks? If in doubt, give a shout! Our editorial staff will be pleased to assist you with your projects at all times.

Earlier articles:

1. *The Junior Computer*; May 1980, p. 5-08. Introducing the computer.
2. *8K RAM + 4, 8 or 16K EPROM on a single card*; Elektor September 1980, p. 9-04. Universal memory expansion.
3. *Elektor μ P's*; Elektor May 1980, p. 5-04. The various microcomputer systems published in Elektor.
4. *The Junior Computer memory card*; Elektor, October 1980, p. 10-22. How to connect the card to the Junior Computer.
5. *The Junior Computer Book 1*, the practical introduction to a powerful system; Elektor Publishers Ltd. ISBN 0 905705 05x. Copyright 1980.
6. *Junior's growing up*; Elektor, February 1981, p. 2-16. The short and long-term outlook.
7. *Text display on the Junior Computer*; Elektor March 1981, p. 3-36. A seven-segment alphabet enables written texts to be displayed.