logic analyser II

Last month, the basic principles of the logic analyser were explained with the aid of block diagrams. Now the moment has arrived to see what the actual circuit diagrams look like. Again, the unit has been split up into two sections: the logic analyser itself and the cursor circuit. This makes it easier to 'place' the various parts previously shown in the block diagrams.

The logic analyser block diagram was by no means straightforward and so it is not surprising that it leads to a rather elaborate circuit diagram. For this reason, it is advisable to look up the previous article and have the block diagrams at hand, so as to see which components represent which block. Otherwise, the circuit diagram is bound to cause confusion.

To start with, a few general remarks. The logic analyser consists of three main sections: the analyser, the cursor and of course the power supply. Wherever possible, LS TTL ICs have been used to limit current consumption. The printed circuit boards and the constructional details will be dealt with next month. The circuits are more than enough for now.

The logic analyser circuit diagram

For a change, we'll start in the middle of figure 1 (that is, the lower left-hand corner, to be precise) — at the heart of the logic analyser. This consists of the clock oscillator and the time base switch. With the given capacitor values, the voltage controlled oscillator IC9 produces a frequency of 4 MHz. The oscillator's stability can be improved considerably by replacing capacitors C7 and C8 by a 4 MHz crystal. With the aid of divide-by-two and divide-by-five stages (IC12 . . . 14a) different sampling rates are obtained from the oscillator frequency. The desired rate can be selected with switch S1. S2 enables the division ratios of the dividers to be changed, thereby extending the number of rates. The final position of S1 (K) is connected to gate N12. An external clock can be connected to this. The position of S3 will then determine whether the circuit reacts to the positive or to the negative edge of the external signal.

Table 1 lists the sampling rates for the different positions of S1 and S2. Gates N20, N22 and N23 are used to switch from the selected sampling frequency to the fixed scan frequency, and vice versa. For this purpose one input of N20 is connected to the Q output of FF2 and one of N22's inputs is connected to the $\overline{Q}$ output. Thus, the state of FF2 will determine which signal is passed.

What about the inputs? The eight data inputs are connected to the latch (IC1). This transfers the input data to the outputs, at the sampling rate determined by S1 and S2. The delay time that elapses between the sampling pulse and the data transfer can be preset with the monostable multivibrator MMV1. The delay switch S19 allows for two alternatives. When it is in position a, the delay will be fixed at 50 ns; when it is in position b, the delay can be adjusted between 150 and 500 ns by means of P1. The A input of MMV1 is connected to the output of N22 which produces the sampling frequency during data entry. In the display mode, N22 is blocked so that the latch will not receive any sampling pulses either.

The memory consists of two 256 x 4 bit RAMs, 2101A-2 (IC2 and IC3) which have to perform their utmost in this design. This is because the shortest sampling time is 250 ns which happens to be the shortest time that the RAMs can process. The data at the latch outputs is passed to the memory ICs, four lines leading to IC2 and another four to IC3. The addresses are provided by IC4 and IC5. Together they constitute the 8 bit counter A shown in the block diagram. This counter continually scans all the addresses in the memory, as its clock inputs receive the sampling or scan frequency from N23.

Back to the trigger section. After passing the latch, the data is also applied to the word recognizer. This consists of es N1 . . . N10 and switches S5 . . . S14. The open collector outputs of the gates are all interconnected and are linked to the positive supply through R11. This means FF1 will only be provided with a trigger pulse when all gate outputs are logic 1. Two external trigger inputs are included via N1 and N2. One input of each of the remaining gates (N3 . . . N10) is connected to one of IC1's outputs.

The other input of every EXNOR is connected to +5 V via a 5k6 resistor and to the centre contact of a three-way switch. When the switch is on a, the input connected to it will be logic zero; logic one corresponds to the centre position, and in position b the two inputs of the gate will be linked. The three switch positions are labelled 'L', 'H' and 'x' (both in the circuit and on the front panel). In position L, the output of the corresponding gate will only become high if that of the latch is low. Similarly, 'H' indicates that the output from the latch must be high; finally, in the 'x' (don't care) position the EXNOR output will always be logic one, regardless of the latch's state. The switches can therefore be used to preset an eight-bit pattern, or 'word'. As soon as this word appears at the outputs of the latch, FF1 will be triggered by the word recognizer. That is, assuming the two switches for the external trigger inputs are on 'x'.

When FF1 is triggered, either by the word recognizer or by the 'manual trigger' key (S15), the analyser is switched to the 'display' mode. The reset key S16 resets the flipflop and thus the analyser. LED D1, driven from the $\overline{Q}$ output, will light as soon as the analyser is triggered.

The data outputs of the RAMs are connected to the inputs of the 8-to-1 multiplexer IC6. Counter C (IC14b) determines which input of the multiplexer is connected through to the

Table 1

Sampling rates

S1	S2	
	a	b
a	250 ns	250 ns
b	500 ns	500 ns
c	500 ns	1 µs
d	2.5 µs	5 µs
e	5 µs	10 µs
f	25 µs	50 µs
g	50 µs	100 µs
h	250 µs	500 µ
i	500 µ	1 ms
j	2.5 ms	5 ms
k	EXT	EXT

1

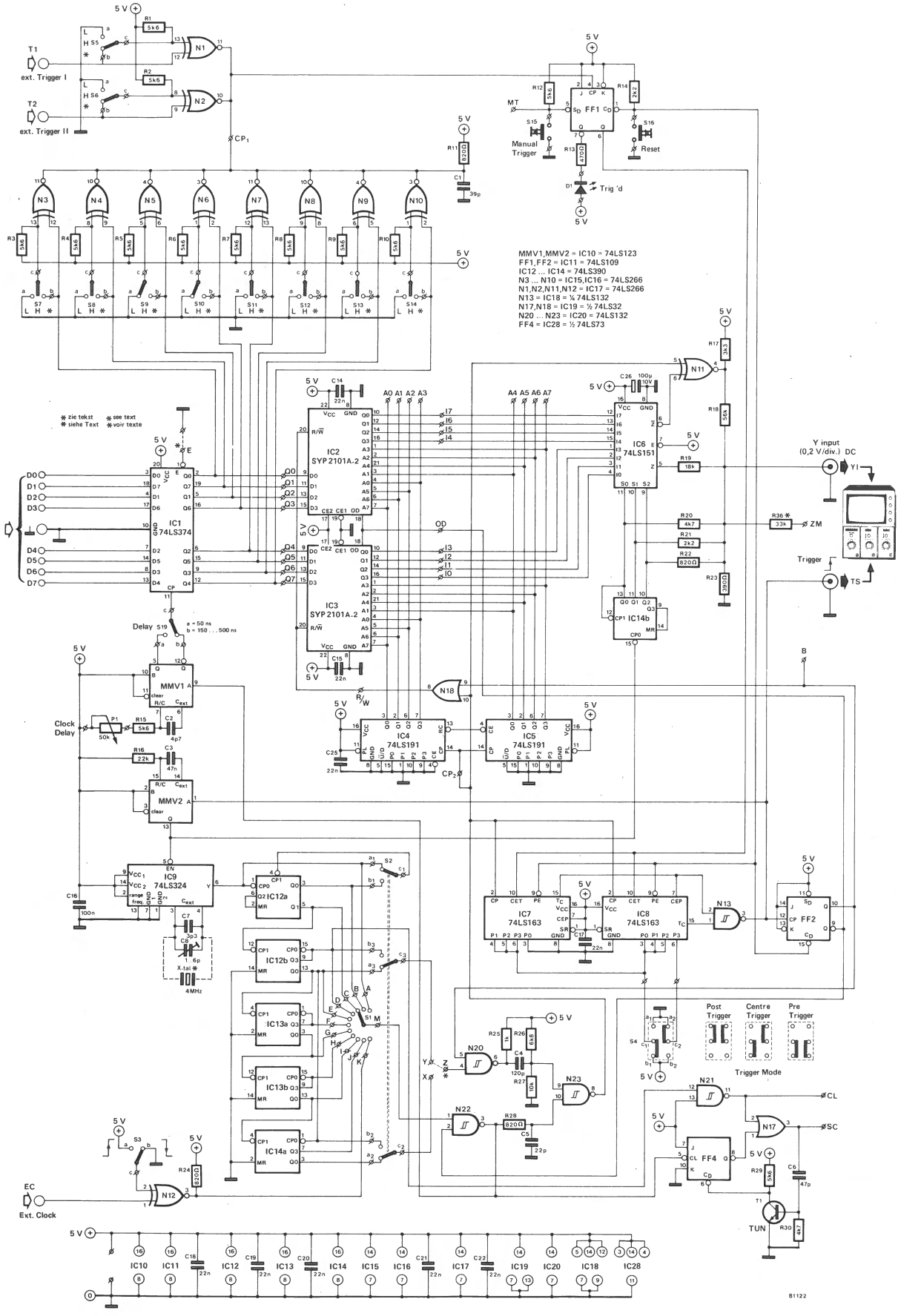

Figure 1. The logic analyser circuit diagram does seem rather elaborate, but then it offers quite a few interesting facilities.

2

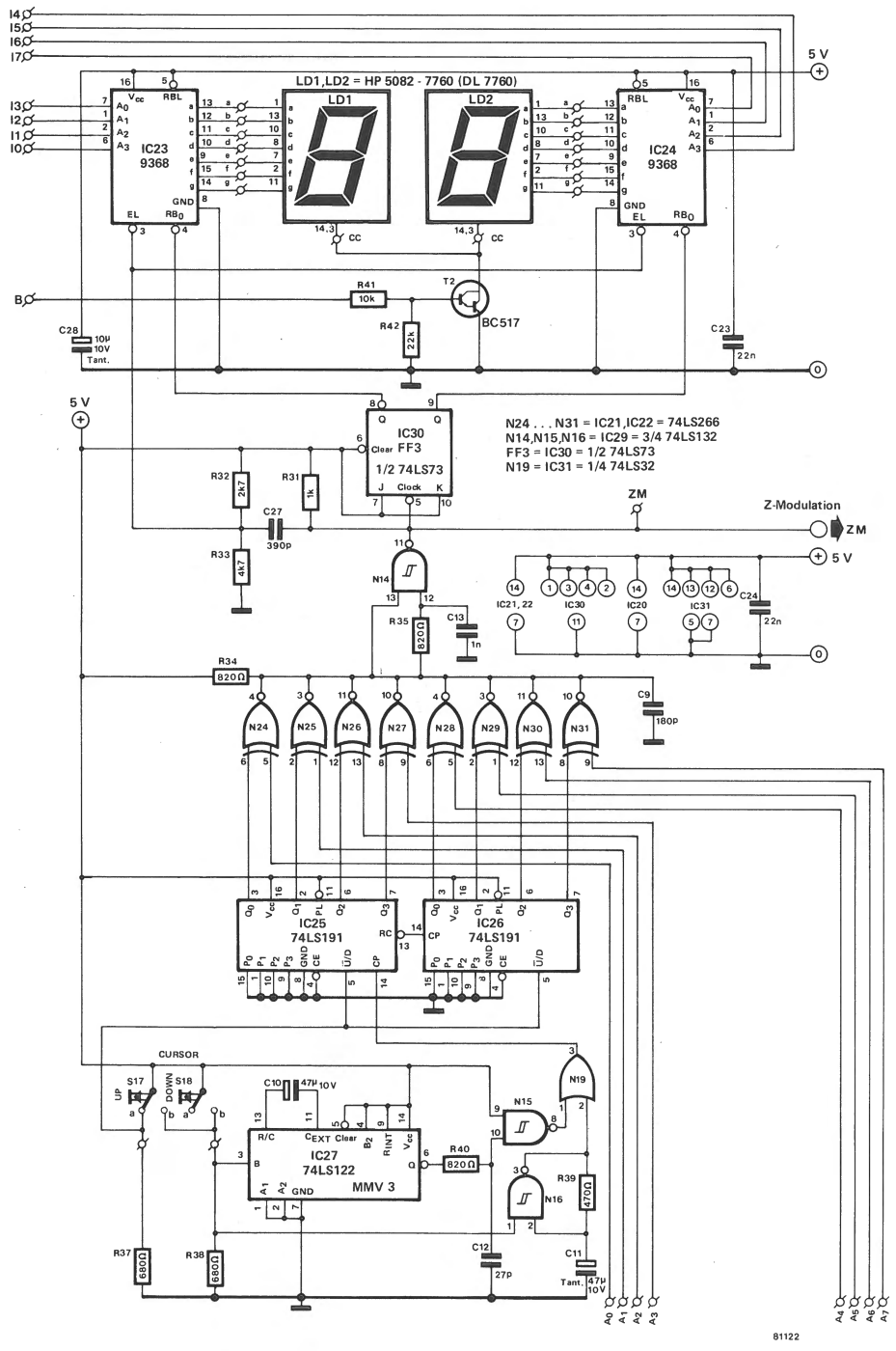

Figure 2. The cursor circuit, a great help when reading data on the screen.

output. The D/A converter shown in the block diagram is not immediately apparent in the circuit. In actual fact, it consists of R19 . . . R23. These resistors sum the data output from the multiplexer and the outputs of counter C, all in the correct proportion to obtain the analogue voltage required for the y input of the 'scope'. A further signal is also mixed in at this point (via N11 and R18) to produce a 'dotted line' between the logic 0 and logic 1 levels, as can be seen in the photos of the display.

Counter B in the block diagram appears as IC7 and IC8 in the circuit. From the trigger signal and the preset of S4 it ascertains when the data entry into RAM is to stop. Furthermore, the counter ensures that the memory is read out again in the same (correct) manner. The 'trigger mode' switch S4 sets the 'preset' inputs of the counter to 0, 126 or 254, as explained last month. When counter B generates a carry pulse, FF2 is triggered; in turn, this flipflop switches the system from 'sampling' to 'scan' mode.

To sum it up

Having located all the sections, running briefly through a complete load and display cycle may help to clarify things. Initially, the desired sampling rate is set by S1; a 'trigger word' is programmed into S7 . . . S14 (S5 and S6 are set to 'x'); S4 is set, say, to 'centre trigger'.

Operating the reset key (S16) clears FF1 and FF2, and causes the preset (126 in this example) to be loaded into counter B (IC7 and IC8). The data now starts to appear 'byte-by-byte' at the memory inputs, via IC1; the memory ICs (IC2 and IC3) are set to 'write' mode and counter A (IC4, IC5) cycles continuously through the full address range, so that each incoming 'sample'

is stored at the next higher address.
If and when the incoming data is identical to the combination set up on S7 . . . S14, the word recognizer will trigger FF1. LED D1 now lights, and counter B (IC7, IC8) is enabled. This counter starts to count sampling pulses, starting from the preset number (126), until it reaches 255. Depending on the preset a further 1, 129 or 255 samples are required. It then gives a 'carry', toggling FF2.
Simultaneously, MMV2 is triggered; this stops the clock oscillator (IC9) for a short period. The circuit is now in 'display' mode: as soon as the clock is started again, the memory will be 'read' at the fixed scan frequency (200 kHz).
During the first scan, one of the data lines is selected by the multiplexer and is displayed on the screen. At the end of this scan (after 256 bits, in other words), counter B again produces a carry pulse. As before, this stops the 'clock', counter C (IC14b) is incremented, select the next data line, the clock is re-started and the 'scope receives the next trigger pulse.
From the above it will be clear that the eight traces are not displayed simultaneously on the screen — how could they, on a single-channel 'scope'? However, they are 'multiplexed' at such a high rate (less than 10 milliseconds for a complete 8-channel display) that they all 'appear' at the same time.
So much for figure 1. The handful of components in the lower right-hand corner (N17, N21, FF4, etc.) will not be dealt with here. They are part of an extension circuit that converts the logic analyser into a 'storage oscilloscope front-end'. Hold your horses! We'll get to that in two or three months.

The cursor

The cursor circuit is shown in figure 2. The wire links that connect it to the main circuit in figure 1 are labelled A0 . . . A7, I0 . . . I7 and B. In the block diagram these correspond to the connections to RAM, counter A and FF2.
The two displays LD1 and LD2 are controlled by IC23 and IC24. These binary-to-seven segment converters convert 8-bit data into two hexadecimal numbers. Each converter is connected to four data lines in memory. The common cathodes of the two displays are switched by T2. The base of this transistor is connected to the Q output of FF2. As a result, the displays only light if data is being written onto the screen.
The idea behind the cursor is that an address can be selected; the data at this address must appear in the displays, with some position indication on the screen.
The circuit that recognizes the preset address is very similar to the word recognizer circuit. Gates N24 . . . N31 compare the contents of counter A (IC4 and IC5) to the contents of counter D (IC25 and IC26). When they are identical, the output of the comparator circuit will become logic one. Via N14, this causes the display decoders to read the data. FF3 switches at every pulse generated by the comparator circuit: once for each complete memory scan. This multiplexes the displays.
The output of N14 can be connected to the Z modulation input of the oscilloscope. As a result, eight brighter dots appear on the screen in a vertical line, one on each scan. These dots indicate the position of the data being displayed as two hexadecimal digits.
The contents of counter D, which determine the position of the dots on the screen and the data on the displays, can be preset with pushbuttons S17 and S18. They operate the 'cursor control' which produces the clock pulses and the up/down signal for counters IC25 and IC26. When S17 (up) is depressed the up/down signal becomes logic zero; the counter will now count the pulses that appear at its clock input. If, on the other hand, S18 is operated (down), the up/down signal becomes logic one and the counter will count down.
The up/down pulse generator may look a little complicated but this does provide some interesting facilities. If either S17 or S18 is depressed for less than 0.5 seconds, only one pulse will be sent to the counter and the cursor will only shift one position. If, however, one of the two switches is held down for longer, a 25 Hz frequency will start to appear at the output of N19 and the cursor then moves left or right across the screen at a much higher speed. This is achieved as follows. When a key is operated, the output of the oscillator around N16 immediately becomes logic zero. At the same time, MMV3 is activated to make its $\overline{Q}$ output '0' as well. The R40/C12 combination briefly delays N15 from reacting to the '0'. Consequently, N19 passes N16's logic zero to the counter. After this short interval, the output of N15 becomes logic one and the oscillator signal is therefore blocked by N19. When the MMV time (0.5 seconds) has elapsed, the output of N15 becomes low again and the oscillator frequency of 25 Hz is passed to the counters. Figure 3 should help to clarify this.
Z modulation can take place in various ways. If the oscilloscope has a suitable connection, the corresponding output can be connected to it. In some cases this may require an inverter, depending on the type of oscilloscope used. If, however, the oscilloscope does not have a Z modulation input, the cursor can be made visible by including resistor R36 in the circuit. The cursor will then be represented on the screen as a slight dent on every line.

3

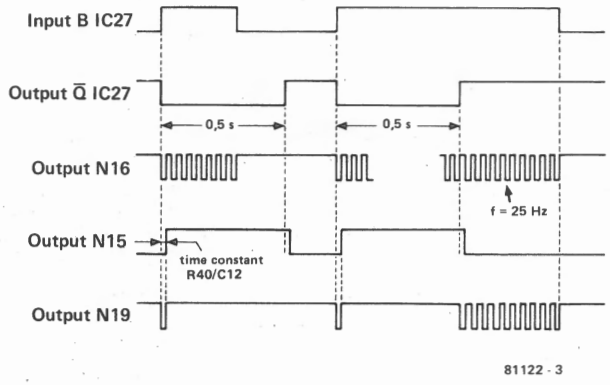

Figure 3. This pulse diagram gives a good idea of the way in which the pulse generator in the cursor works.

The power supply

The Junior Computer's supply (Elektor, May 1980, p. 5-11) turned out to suit the circuit perfectly, so there was no need to design a new version. The logic analyser only requires the +5 V section, but the +12 V and −5 V supplies will be used for the 'storage scope' extension board we mentioned earlier.

In the next episode . . .

Obviously, not everything in the logic analyser circuit diagram could be discussed in full detail. Nevertheless, we hope the circuit's operation will now be clearly understood. In the next issue, the final episode of the logic analyser saga will include constructional details, printed circuit boards and a front panel design.
After that, we will come to the 'storage oscilloscope'.