Construction Project:

AN IMPROVED DSO ADAPTOR FOR PC'S - 1

In the February 1993 issue, we published the design for a low cost digital sampling scope adaptor for PC's, which has been very popular. However various people have asked if we could describe a similar unit but with additional features, such as a calibrated vertical amplifier and timebase, and more flexible triggering facilities. So here it is: the Mark 2 DSO Adaptor.

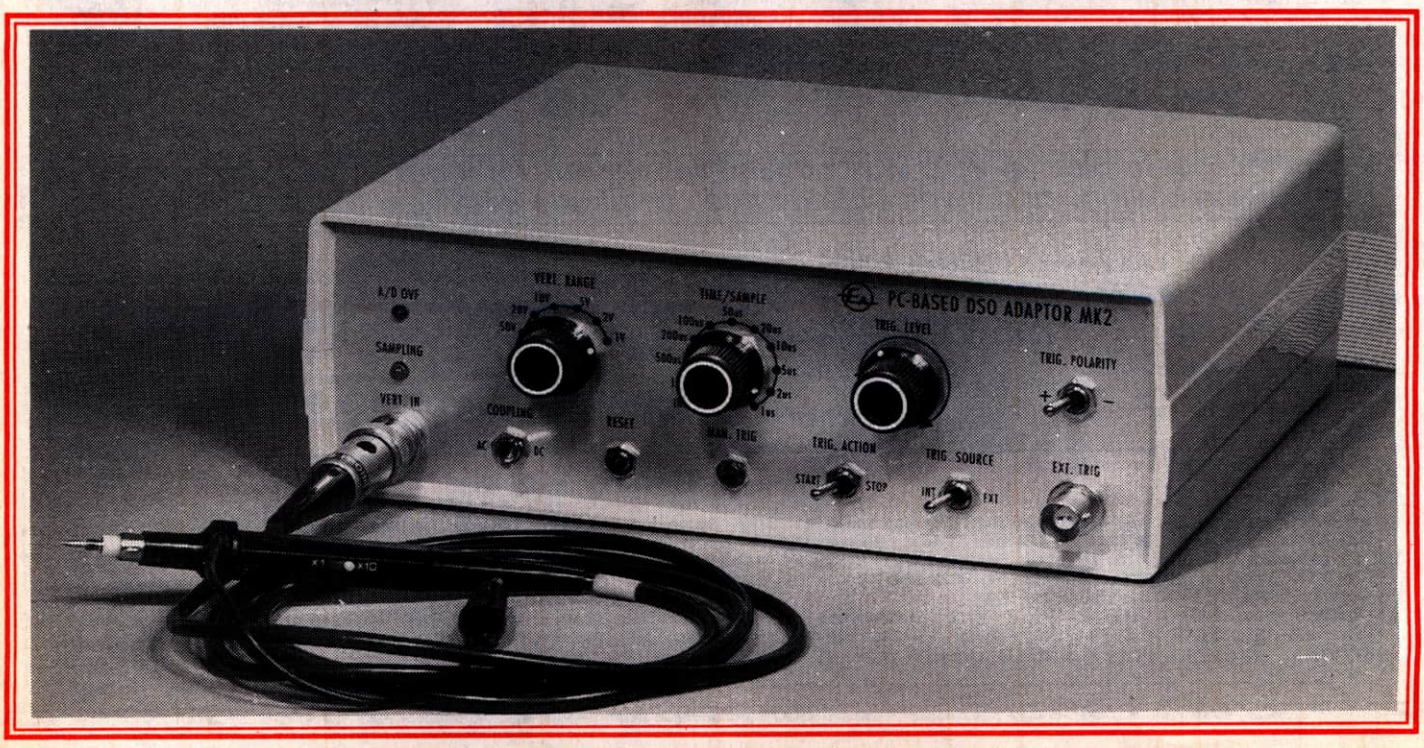
by JIM ROWE

It's really not surprising that David Jones' original DSO design was so popular, because it provided a very low cost way to provide yourself with an audio DSO — using your PC for the display, storage and printout of sampled waveforms. The hardware design was also quite elegant, using only a small number of low-cost ICs. Most commercial DSO's are quite complex and expensive instruments, with even the new handheld LCD-readout models generally carrying a price tag of around $2000. The alternative approach of adapting an existing PC to do the job for less than $100 obviously appealed to many hundreds of readers, judging from what we've heard regarding the numbers of kits sold...

All the same, the conscious 'lowest possible cost' approach taken by Mr Jones did result in a few limitations in terms of the adaptor's facilities. Neither the vertical amplifier nor the sampling timebase was calibrated, both being adjusted by means of simple pots, and the triggering facilities were also rather restricted. For those used to using modern 'scopes, these limitations did tend to grate somewhat — hence the requests to come up with something just a *little* more pretentious.

That's the background, then, to this new DSO Adaptor design. Although based on the David Jones original, and capable of being used with both his original and more recent software, it provides a number of enhanced measurement facilities, including:

- A calibrated vertical amplifier, with a maximum input sensitivity of 1V full scale, an effective frequency response of DC - 600kHz and a risetime of around 200ns.
- A calibrated sampling timebase, with 11 crystal-derived sampling clock periods from 1us to 1ms, in a 1/2/5 sequence.
- A more flexible triggering circuit, offering a choice of either internal or external triggering source, positive- or negative-slope triggering, software triggering from the PC and selection of triggering to either 'start' or 'stop' sampling (for capturing events either after, or before the trigger event).
- Circuitry which allows the sampling write/read mode switching to be con-



trolled directly by the PC software, instead of a manual switch. This makes operation faster and more convenient. (However a manual switch can be fitted if desired, to make the new unit compatible with the original software.)

- Similarly, there is now also circuitry to allow the PC software to supply the adaptor's A/D converter directly with sampling clock pulses — making possible full software- controlled sampling, for applications such as long-term data logging.

Another difference between the Mk2 adaptor and the earlier design is that it is now able to take advantage of a new and faster A/D converter chip, the ADC08061. This is capable of sampling at rates up to 1.5MS/s (megasamples per second), roughly twice the limit of the ADC0820 chip used in the original design.

It's thanks to this faster converter that the new unit is able to have a maximum sampling timebase range of 1us (i.e., 1MS/s). This gives the DSO adaptor an effective bandwidth of about 100kHz, on the basis of at least 10 samples per signal period.

I should point out, though, that because the ADC08061 is at least 50% more expensive than the slower chip, some readers may still prefer to use the ADC0820 — especially if they don't need a bandwidth of more than 65kHz or so. Luckily the two chips have the same pinouts, so the slower chip can still be used if you wish; all that is needed is to change the value of one resistor, as I'll explain later. But of course you won't be able to use the fastest (1us) timebase range, if you do this.

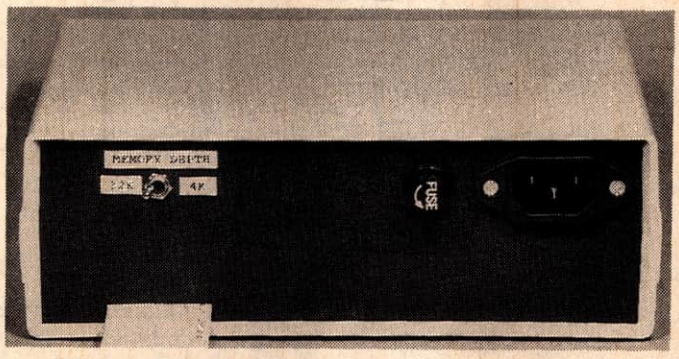
By the way, contributor David Jones had already made many significant improvements to his original DSOA software, and on hearing that I had produced this new version of the hardware, he very kindly volunteered to adapt his latest version of the software to suit it. So there's a very nice software package now available to suit the new design, directly from David himself. You'll find more details about this in a box, later in the article.

As with the original DSO adaptor, the basic idea is that when triggered, the circuit hardware automatically takes a series of digital samples of the input signal, and stores or 'writes' them sequentially in an on-board memory. This is all done in 'real time', at a rate determined by the sampling timebase clock you've selected.

After the set of samples have been stored, they are then read back out of the memory, again in sequence, and passed back to the PC via the normal Centronics-type parallel printer port. This second 'read' phase is performed under the control of software running in the PC, and the software can also display the sampled waveform on screen, zoom in to display details, save the waveform to disk, retrieve it again and so on.

Circuit description

Now let's have a look at the circuitry which does the hardware side of these functions, in the new adaptor. As before, the heart of the adaptor is formed by U12, the A/D converter chip, and U16 which is the memory — a fast static RAM. U12 is where our analog input signal is converted into a series of digital samples, and U16 is where they're then stored.



The rear view is not particularly exciting. At upper right are the mains input and fuse, with the PC interface cable exiting at lower left. The optional switch visible at upper left allows selecting either 4K or 32K of effective sample memory.

Like the ADC0820 A/D converter used in the original adaptor, the ADC08061 is a compromise between the super-fast 'full flash' type and the slower 'successive approximation' type. In this case it uses a multi-step technique, to deliver a full eight-bit sample in less than 700ns — using only 14 internal comparators, plus some fancy multiplexing. As a result, it can comfortably perform over one million conversions per second.

The SRAM chip used for U16 is the same is that used in the original design: a 62256-10, which is a 100ns access time 256K bit device, organised as 32K (32,768) eight-bit words.

A faster chip such as the 62256-8 (80ns) can be used instead, but not a slower one. As you can see, the data outputs of the A/D chip are connected to the data I/O lines of the SRAM.

Each conversion cycle is triggered by the arrival of a negative-going pulse at pin 6 of U12. Pin 9 of U12 goes to a logic high during the conversion, and then goes low at the end. This 'INT' pulse is fed via inverter U17e and gate U18a to the Write Enable-bar input of U16, to cause the memory chip to write the sample into the next available memory.

The same pulse is then fed via U10d to the clock input of counter chip U14, which together with U13 forms the SRAM's memory address register. As a result the memory address is incremented ready for the next write cycle.

Each 'sample clock' pulse fed to pin 6 of U12 initiates one of these conversion/write cycles, so by controlling the pulses we determine not only when sampling takes place, but also the rate at which samples are taken.

Control of the sampling is made by 'master write control' flipflop U9b, whose Q output (pin 9) enables gate U6c. The input pulses to U6c are shaped by C21, R21 and D6, fed in turn with a timebase signal selected by timebase switch S8 and buffered by gate U6b.

The purpose of the shaping circuit is to ensure that the pulses fed to U12 are narrow enough to allow reliable operation at sampling rates up to 1MS/s, but wider than the ADC08061's requirement for 100ns minimum pulse width. The values shown give a width of 150ns — which is increased to 680ns if the slower ADC0820 is used, by increasing R21 to 680 ohms.

Crystal timebase

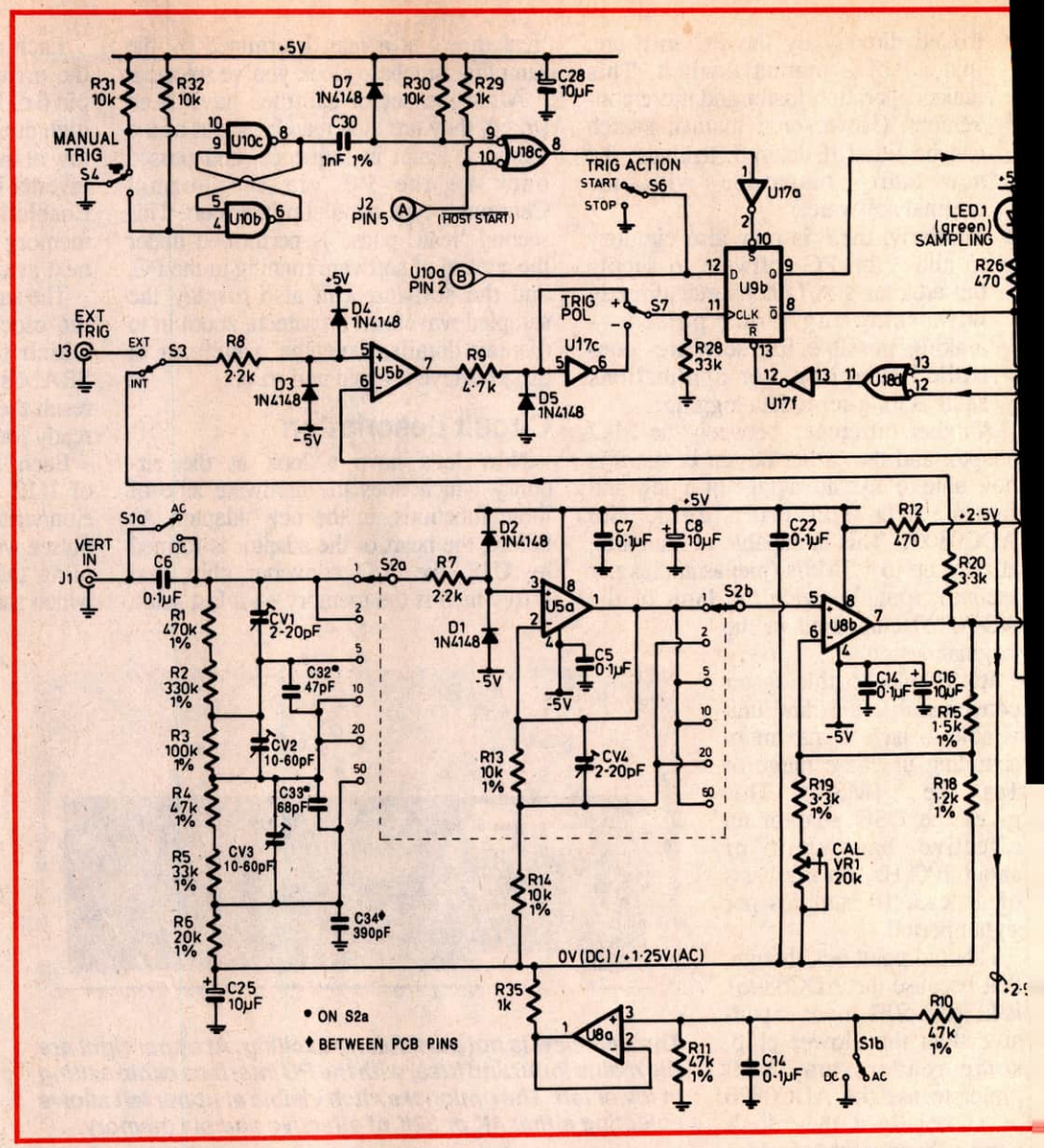
Most of the timebase signals selected by S8 are derived from a crystal divider circuit, to give accurately known sampling rates. The crystal oscillator uses a low cost 2MHz crystal X1, together with gate U6a. Trimmer capacitor CV5 is used to adjust the crystal to exact frequency, for timebase calibration.

The 2MHz signal from U6a is divided by the two cascaded sections of U7, a 74HC74 dual flipflop, to provide the 1us and 2us timebase signals. The 1us signal is then passed to the various sections of U3 and U4 (both 74HC390 dual decade counters) to produce the remaining signals in a 1/2/5 sequence, down to 1ms.

The remaining 'timebase' signal selected by S8, in the 'host' position, comes from the PC itself via pin 4 of connector J2. This allows the PC software to initiate each A/D conversion

DSO adaptor for PC's - 1

Here is the schematic for most of the new DSO adaptor — the crystal derived time- base and power supply are shown separately. Features added to this version include a calibrated vertical amplifier with ranges from 1V to 50V full scale, a calibrated sampling timebase with clock rates from 1us to 1ms, and more extensive triggering facilities — such as selection of positive or negative slope triggering, internal or external triggering, and the ability to either start or stop sampling with the trigger event. The PC can also provide sampling pulses directly, for long-term data logging.



cycle directly, for applications like data logging.

Like most A/D converter chips, U12 essentially operates by matching the analog input voltage, fed into pin 1, with a digitally switched fraction of an analog reference voltage, fed into it via pins 12 and 11. In this case the reference voltage is an accurately known 2.5V DC, developed via shunt reference source U11. Although U11 is shown in the schematic as a zener diode, it is in fact an IC in a TO-92 package, designed to develop a particularly accurate and stable voltage.

The LM4040CIZ-2.5 device specified develops 2.5V +/-0.5%, stable to within 100ppm/°C. (Alternative devices are the LM336Z-2.5 or the LM4041CIZ-ADJ, which in both cases also require additional divider resistors R36 and R37 — both 10k.)

Since the converter's reference voltage is +2.5V DC, the analog input voltage to be sampled by U12 can only vary between 0V and +2.5V — 0V produces the converter's minimum digital output of 00 hex, while +2.5V produces its maximum output of FF hex. So our vertical input amplifier, designed around op-amps U5a and U8b, is arranged to deliver its output in this range.

Vertical amplifier

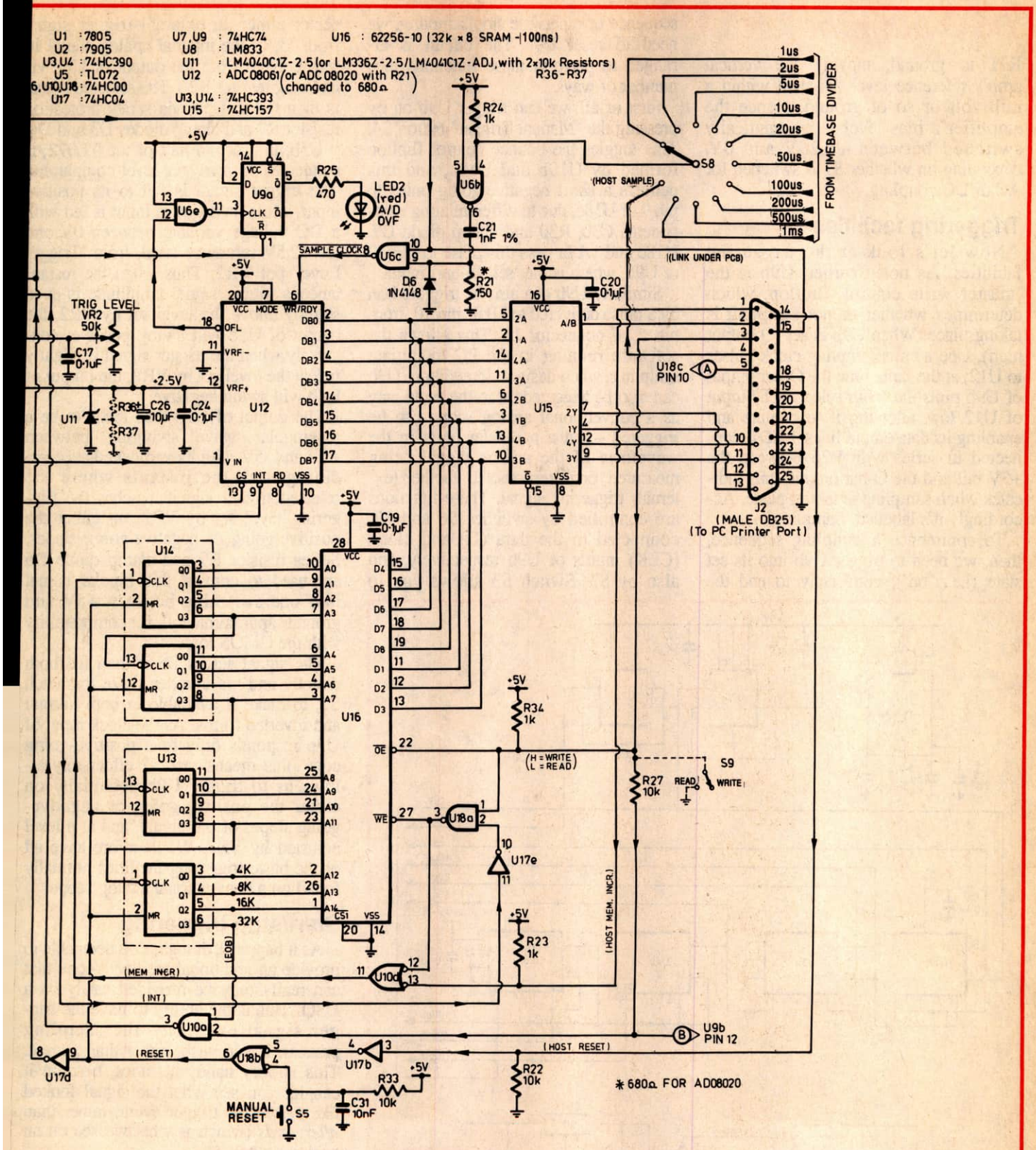
U5a is half a TL072 FET-input device, used as a unity gain buffer. This allows us to provide the vertical amplifier with a standard input impedance of 1M, so it is directly compatible with standard scope probes. Resistors R1 - R6 are connected as an input divider, and this together with a further divider formed by R13 and R14 at the output of the buffer provides a range of input sensitivities while also limiting the signal amplitude handled by U5a, to prevent distortion. Resistor R7 and diodes D1 and D2 are used to protect U5a against damage from gross overload.

Note that there are six ranges, selected by the two poles of switch S2 and labelled here in terms of the peak-to-peak signal amplitude corresponding to 'full scale deflection' (not the volts per major deflection division, as is more usual with analog scopes). Trimmer caps CV1 - CV4 are used to adjust the frequency compensation for the various ranges.

The signal selected by S2b is fed to U8b, which is half of an LM833 low noise dual op-amp. This is connected as a wideband amplifier stage, with a gain set to approximately 2.5 times by the negative feedback divider formed by R15, R18, R19 and trimpot RV1. The trimpot is used to balance the gain of the stage against the exact reference voltage provided by U11, for calibrating the adaptor's vertical sensitivity.

As you can see, the basic vertical amplifier chain formed by U5a and U8b is DC coupled, right through to the A/D chip input.

To provide the adaptor with the ability to measure AC signals separately from any DC component, switch S1a allows



coupling capacitor C6 to be switched in series with the input. However when this is done, the DC reference level of the vertical amplifier must be raised above ground, to allow the signal to swing in either direction without leaving the 0V - 2.5V 'input window' range of the A/D converter chip. In fact the reference level needs to be moved up to +1.25V — halfway up the range, in order to allow equal positive and negative swings.

We do this by means of U8a, wired again as a unity gain buffer. Divider resistors R10 and R11 are used to derive an accurate +1.25V from the 2.5V reference produced by U11, and this is duplicated at the low-impedance output of U8a. The output is then fed via stopper resistor R35 to the 'bottom end' of R6, R14, VR1 and R18, so that it establishes the quiescent bias level of the complete vertical amplifier channel. Capacitor C25 is used to ensure that the bias line is at ground potential for all AC frequencies of interest.

In the DC position of S1, its second pole S1b shorts the junction of R10 and

R11 to ground, moving the vertical amp's reference level down to within a millivolt or so of ground. Hence the amplifier's bias level is automatically switched between +1.25V and 0V, depending on whether S1 is switched for AC or DC coupling.

Triggering facilities

Now let's look at the triggering facilities. As noted earlier, U9b is the 'master write control' flipflop, which determines whether or not sampling is taking place. When U9b is set (Q output high), U6c admits sampling clock pulses to U12; at the same time the Q-bar output of U9b pulls the 'chip select' (CS) input of U12 low, allowing it to operate and enabling its data output lines. LED1, connected in series with R26 between the +5V rail and the Q-bar output, thus indicates when sampling is taking place. Accordingly it's labelled 'Sampling'.

To commence a sampling sequence, then, we need to trigger U9b into its set state (i.e., 'on'); conversely to end the sequence or otherwise stop sampling, we need to reset U9b. The circuit is arranged to perform these functions in a number of ways.

First of all, we can trigger U9b on by pressing the 'Manual Trigger' button S4. This toggles the bounce control flipflop formed by U10b and U10c, and this delivers a short negative-going pulse to pin 9 of U18c, due to differentiating components C30, R30 and clamp diode D7. U18c and U17a pass this pulse to pin 10 of U9b, which is the 'set' (S-bar) input.

Similarly U9b can also be triggered on by a pulse delivered to U18c pin 10, from pin 5 of connector J2. This allows the software running in the PC to initiate sampling, when desired. In addition, U9b can also be triggered on in the same way as a conventional analog scope can be triggered — at a particular point in the waveform of the analog signal being measured, or in response to another (external) triggering signal. These functions are controlled by switches S6 and S7, connected to the data (D) and clock (CLK) inputs of U9b respectively, and also by S3. Switch S3 allows you to select either an external trigger signa from J3, or the internal analog signal it self, derived from the output of U8b vi isolating resistor R20. The selected signa is then fed to U5b, via series protectio resistor R7 and clamp diodes D3 and D4

U5b, the second half of the TL072, i connected as a trigger level comparato The trigger signal is fed to its positive input, while its negative input is fed with a DC voltage variable between 0V and the +2.5V reference level, from 'Trigger Level' pot VR2. Thus when the instantaneous trigger signal amplitude is even slightly *above* the level set by VR2, the output of U5b will swing positive; conversely when the trigger signal is slightly *below* the level set by VR2, the output of U5b will swing negative.

The output of U5b thus tends to be a rectangular wave swinging between +5V and -5V, with its vertical edges coinciding with the instants where the selected trigger signal matches the triggering level set by VR2, on either the positive-going or negative-going slopes. Series resistor R9 and clamp diode D5 are used to convert this bipolar signal into one switching between +5V and ground (approximately), for compatibility with the CMOS logic.

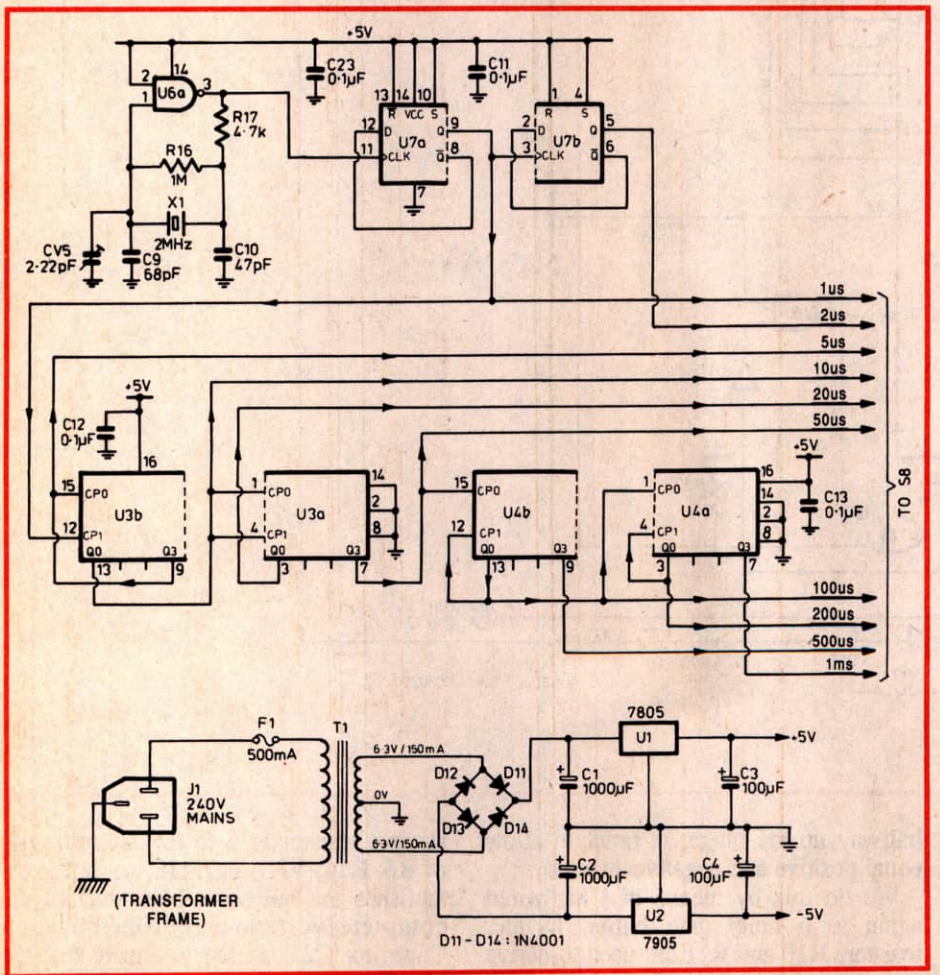
The signal across D5 is then fed both directly and via inverter U17c to switch S7, to make it available in both normal and inverted forms. As the clock input of U9b responds only to a positive-going edge, this means that S7 effectively allows us to trigger U9b potentially on either the positive-going or negative-going slopes of our signal, and at a level adjusted by VR2. We therefore have all of the basic triggering facilities normally found on a conventional analog scope.

Start/stop triggering

As it happens, though, I've been able to provide an additional facility — one that can really only be provided easily on a DSO. This is the ability to have the trigger signal event *stop* the sampling process, if you wish, rather than *start* it. This is very handy at times, because it can let you see what the signal looked like *before* the trigger event, rather than *afterwards* (which is what we see on an analog scope).

The additional facility is provided by using 'Trigger Action' switch S6 to manipulate the data (D) input of U9b — which of course determines what state the flipflop assumes, when a triggering edge is fed into the CLK input.

For the more usual 'start' mode of triggering, S6 is open circuit and the D input (pin 12) of U9b is pulled to the high logic level by resistor R27, which connects to



The schematic for the remaining crystal timebase divider chain and power supply sections of the DSO adaptor. Only three divider ICs and a single-gate oscillator are used to generate 10 different sampling clock frequencies in a 1/2/5 sequence.

PARTS LIST

Resistors

R1	470k 1% metal film
R2	330k 1% metal film
R3	100k 1% metal film
R4	47k 1% metal film
R5	33k 1% metal film
R6	20k 1% metal film
R7,R8	2.2k 5% 1/4W
R9,R17	4.7k 5% 1/4W
R10,R11	47k 1% metal film
R12,R25,R26	470 ohms 5% 1/4W
R13,R14,R22,R27,R30,R31,R32,R33	10k 5% 1/4W
R15	1.5k 1% metal film
R16	1M 1% metal film
R18	1.2k 1% metal film
R19	3.3k 1% metal film
R20	3.3k 5% 1/4W
R21	150 ohms 1% metal film (or 680 ohms: see text)
R23,R24,R29,R34,R35	1k 5% 1/4W
R28	33k 5% 1/4W
R36,R37	10k 1% metal film (optional: see text)
VR1	20k linear trimpot, horiz PCB mount (small)
VR2	50k linear pot

Capacitors

C1,C2	2200uF 16VW RB electrolytic
C3,C4	100uF 16VW RB electrolytic
C5,C7,C11,C12,C13,C14,C15,C17,C18, C19,C20,C22,C23,C24,C29	0.1uF monolithic ceramic
C6	0.1uF 100VW metallised polyester or MKT
C8,C16,C25,C26,C28	10uF 25VW TAG tantalum
C9,C33	68pF NPO ceramic
C10,C32	47pF NPO ceramic
C21,C30	1nF 1% 63V polystyrene
C31	10nF monolithic ceramic
C34	390pF ceramic
CV1,4,5	2-22pF plastic trimcap
CV2,3	10-60pF plastic trimcap

Semiconductors

D1-7	1N4148 or 1N914 signal diode
D11-14	1N4001 or similar power diode
U1	7805 positive 5V regulator (TO-220)
U2	7905 negative 5V regulator (TO-220)
U3,U4	74HC390 dual decade counter
U5	TL072 dual FET input op amp
U6,U10,U18	74HC00 quad NAND gate
U7,U9	74HC74 dual D-type flipflip
U8	LM833 dual low noise op amp
U11	LM4040CIZ-2.5V reference (or LM336Z-2.5 + R36,R37)
U12	ADC08061 A/D converter (or ADC0820: see text)
U13,U14	74HC393 dual four-bit counter
U15	74HC157 quad two-input multiplexer
U16	62256-10 SRAM 32K x 8, 100ns
U17	74HC04 hex inverter

Miscellaneous

S1	DPDT miniature toggle switch
S2	Two pole, 6 position rotary switch
S3,S6,S7	SPDT miniature toggle switch
S4,S5	Miniature pushbutton, SPDT momentary
S8	Single pole, 11 position rotary switch
S9	SPDT miniature toggle switch (optional: see text)
LED1	3mm round LED, green
LED2	3mm round LED, red
X1	Quartz crystal, 2MHz (10 x 3.5 x 13mm case)
T1	Power transformer, 12.6V CT at 150mA
PCB	PC board, 143 x 123mm, code 94dso5
J1,J3	BNC coaxial socket, single hole mounting
J2	13 x 2 length of DIL pin header
J4	IEC 240V male chassis connector
F1	Screw-in 3AG cartridge fuse-holder, with 500mA cartridge fuse

Plastic instrument case, 200 x 160 x 70mm (or 200 x 160 x 65mm); front panel, photo-sensitive aluminium or silk-screened; three instrument knobs, 20mm diameter; 51 x PCB pins; 2 x 8-pin DIP sockets; 8 x 14-pin DIP sockets; 3 x 16-pin DIP sockets; 1 x 20- pin DIP socket; 1 x 28-pin DIP socket (0.6"); 100 x 30mm piece of 1mm aluminium sheet; 100 x 35mm piece of tinplate, for switch shield; 80 x 45mm piece of blank PCB laminate, for PCB shield; 1 x 26-way DIL IDC connector; 1 x DB25 plug, IDC type; 2m length of 25-way IDC ribbon cable; 4 x 3mm x 10mm machine screws with nuts and lock-washers; 2 x 3mm x 6mm PK screws; hookup wire, TC wire, sleeving, etc.

the Write/Read-bar control line driven by pin 3 of connector J2 (or optional switch S9 — more about this shortly). In the adaptor's sampling or 'Write' mode, the W/R-bar control line is pulled high by R34.

With the D input at logic high, U9b therefore switches into the set state when the triggering edge occurs, and sampling begins. To achieve the opposite effect, S6 is simply switched to the 'stop' position, shorting pin 12 of U9b to ground (logic low). This means that if we begin sampling say manually, by pressing the Manual Trigger button S4, the effect of the triggering edge sent to the CLK input will now be to force U9b to *reset*, stopping the sampling (and leaving in memory the samples that have just been taken).

How do we arrange for sampling to stop normally, in 'start' mode? Basically this is done in the same way as the original design, by arranging for our 'master write control' flipflop U9b to be reset when we reach the end of our memory (i.e., when we have a full sequence of samples). This is done by gate U10a, which takes an 'end of buffer' (EOB) signal from a selected output of memory address counter chip U13, and sends a reset signal back to U9b via OR gate U18d and inverter U17f.

By selecting different outputs from U13, to derive our EOB signal, we can stop the sampling after differing numbers of samples have been taken — effectively varying the 'length' of our sample memory.

A choice of four outputs is provided on the board, from the full 32K available in the SRAM (used for maximum resolution) down to 4K (for faster 'snapshot' sample sequences). If desired, an optional switch can be provided on the adaptor's rear panel, to select whichever memory length is desired.

Note that when S6 is switched to 'stop' mode, this resetting of U9b by the EOB signal is disabled, because S6 also controls the logic level at pin 2 of gate U10a. So in this mode, the sampling is *not* stopped when the end of the memory is reached; it continues indefinitely, with samples being stored in the full 32K of memory continuously until U9b is reset by the selected trigger event.

Actually there *is* one other way that sampling can be stopped, at any time. This is by deliberately resetting U9b, either manually or by means of a 'host reset' pulse sent from the PC software. Manual resetting is performed using pushbutton S5, which pulls pin 4 of OR gate U18b low (power-on reset components R33 and C31 normally hold this pin high). Similarly the PC reset pulse enters the adaptor via pin 14 of connector J2, and after inversion by U17b is applied to pin 5 of U18b.

In either case, U18b not only resets the master write control flipflop U9b via U17d, U18d and U17f as before; it also resets the memory address counter chips U13 and U14, so the memory is 'sent back to the start'. This is fine if we've taken the samples in 'start' mode, because we'd normal want to read them back out again from the start anyway. But if we have taken them in 'stop' mode, resetting the memory counters will lose our stopping address and we won't know where the sampling sequence ended...

Overload indicator

There's one more section of the circuit, involved in the sampling or 'write' mode operation, which we haven't yet mentioned: that involving U6e, wired as an inverter, and flipflop U9a. These are used to indicate if the A/D converter has been overloaded and 'sent off scale' during the sampling.

If we have set our input range switch S2 to a range that causes the A/D chip to overload at some point in the signal waveform, the chip responds by driving its OFL-bar output (normally high) down to logic low, at the end of the conversion cycle(s) concerned. However as this out-

put then goes high again at the start of the next conversion cycle (which might begin within a microsecond), we can't use this output to drive an indicator directly. Instead we have to use it to trigger a latch, which will remain on long enough for us to *see* that overload has occurred.

U9a is the latch, triggered on by U12's OFL-bar signal after it has been inverted by U6e. In turn the Q output (pin 5) of U9a drives LED2 via R25, so that if LED2 (labelled 'A/D OVF') glows during a sampling sequence, we know that the converter is being driven into overload and we need to set S2 to a higher range. Note that U9a is only reset by the manual or 'host reset' signals from U18b, via U17d. So after an overload, the complete triggering circuit (and memory address counter) must be reset either manually or by the PC software, before taking another sample sequence.

Reading the samples

Now let's look at the *second* main phase of the adaptor's operation — where the stored samples are read back out of the memory, and sent to the PC...

To change the circuitry into this mode, the Write/Read-bar control line is pulled down to logic low level, either by the PC software (via pin 3 of J2), or by means of optional write/read mode switch S9. This does a number of things, one being to disable gate U18a, so that its output goes high along with pin 27 (WE-bar) of memory chip U16 — preventing any further writing into the memory.

At the same time, pin 22 (OE-bar) of the memory chip is taken low, enabling its data output buffers so that they are ready to provide the read out data. And finally, the logic low on the Write/Read-bar line also pulls the D input of U9b (pin 12), via R27, so that the master write control flipflop cannot be triggered on. This prevents U12 from being enabled, for any further sampling.

The stored sample data is then read from the memory chip and back into the PC, under software control. As there are only five 'input' lines on a standard Centronics parallel port, multiplexer chip U15 (a 74HC157) is used to read each sample byte back as two four-bit 'nibbles', one after the other, via pins 10, 11, 12 and 13 of J2. (The fifth port input, at J2 pin 15, is connected to the Q output of U9b, so the PC software can tell when sampling has stopped.)

The control pin of U15 is pin 1, and to read the data the software first pulls this line low via pin 2 of J2. This causes U15 to make available the 'low nibble' (i.e., bits D0 - D3) of the data byte from the current memory address, at pins 10 - 13 of J2. Then when the software has read this nibble, it toggles pin 1 of U15 high, which makes the 'high nibble' (bits D4 - D7) available instead. This completes the actual read operation, but the final step is for the software to send down a negative-going pulse via pin 1 of J2, to increment the memory address counters U14 and U13 ready to read the data in the next memory location.

This 'read high nibble/read low nibble/increment memory' sequence is therefore repeated over and over, until the PC has read all of the sample data from the memory. Obviously the software has to be told in advance how much memory is being used (i.e., 32K or 4K, etc), so it knows how many bytes are to be retrieved.

So that's basically all there is to reading the sampled data back into the computer, for display and further processing. To ensure that the readout process starts at the beginning of memory, in the case of a sequence of samples made in 'start' triggering mode, the software can send a 'host reset' pulse at the beginning, if desired. However this can't be done in the case of a sequence made in 'stop' triggering mode, because the readout has to begin at the very next memory address from the one where sampling stopped.

In any case, an initial reset pulse is really only necessary when a memory length shorter than the full 32K has been selected (for faster setup), and the samples have been taken in 'start' triggering mode. If the full 32K memory has been used, sampling will have stopped at the end of memory anyway — ready for readout from the start again.

Power supply

The power required by the DSO adaptor's circuitry is very modest, thanks to the use of CMOS devices. The overall drain is typically less than 100mA, in fact, so we are able to power it from one of the smallest transformers in common use: a 2VA unit supplying 12.6V centre-tapped at 150mA (type 2851 or similar).

By grounding the centre-tap and using a full-wave bridge rectifier, we use the transformer to produce raw DC outputs of both +9V and -9V with respect to ground, across reservoir caps C1 and C2. Regulators U1 and U2 are then used to produce regulated supply rails of +5V and -5V, to power all of the adaptor circuitry. The negative rail is used only for the op-amps, to ensure in particular that those in the vertical amplifier can handle the required voltage swing without distortion.

By the way, as the DSO adaptor can be used to take samples away from the PC, its low current drain would allow it to be powered from a battery pack, if you wish. All you'd need is a set of say 12 'D' size cells, connected in series so that they produce a centre-tapped 18V DC. This can be fed into the adaptor's PCB power input pins, instead of the AC from the transformer.

And that's about all we have space for, this month. In the second of these articles I'll describe the construction of the adaptor, as well as how it's set up and used.

(To be continued) ❖

Software for the DSO Adaptor Mk2

As mentioned elsewhere in this article, David Jones (the designer of the original DSO Adaptor) of Tronnort Technology has written a new and expanded version of the updated software for his own adaptor, especially to suit our new model. The new Version 3.0 DSOA software takes advantage of the extra features of the Mk2 Adaptor, and also offers many new 'user friendly' enhancements of its own, including:

- A full on-screen measurement graticule, which allows you to update its legends according to the DSO ranges chosen.
- Much faster waveform display updating than previous versions.
- The ability to perform one-shot measurements from the PC keyboard, with a single keystroke.
- 'Real time' repetitive sampling and screen updating, again controlled very conveniently from the PC keyboard.
- The ability to take individual samples under PC software control at programmed intervals, for data logging.
- Storage of printer port address, DSO buffer size, etc. in a config file, so the software 'doesn't forget'.

At the time of writing, the software doesn't yet provide for direct screen printout, to produce a 'hard copy' version of any display, but David Jones is hoping to have completed this very useful feature by the time the May issue is published.

In short, the new DSOA Version 3.0 software is an excellent program, and one which allows you to make very good use of our new DSO Adaptor Mk2. At the price being asked — only $30 plus $5 postage anywhere in Australia — it seems to us to represent really good value for money, considering how much work has been involved in writing it.

The Version 3.0 software is available from Tronnort Technology, 12 Copeland Road, Lethbridge Park 2770.

We understand that some kits for the DSO Adaptor Mk2 may include this software, by arrangement with Tronnort. However constructors may still wish to register with Tronnort itself as a user of the software, so David Jones can keep you posted on any further upgrades he releases in the future.