TV & VIDEO ✻ AUDIO ✻ COMPUTERS ✻ AUTO ELECTRONICS!

PAY TV: HOW GOOD IS THE VIDEO/AUDIO QUALITY?

NDD

Electronics

AUSTRALIA WITH Professional Electronics & ETI

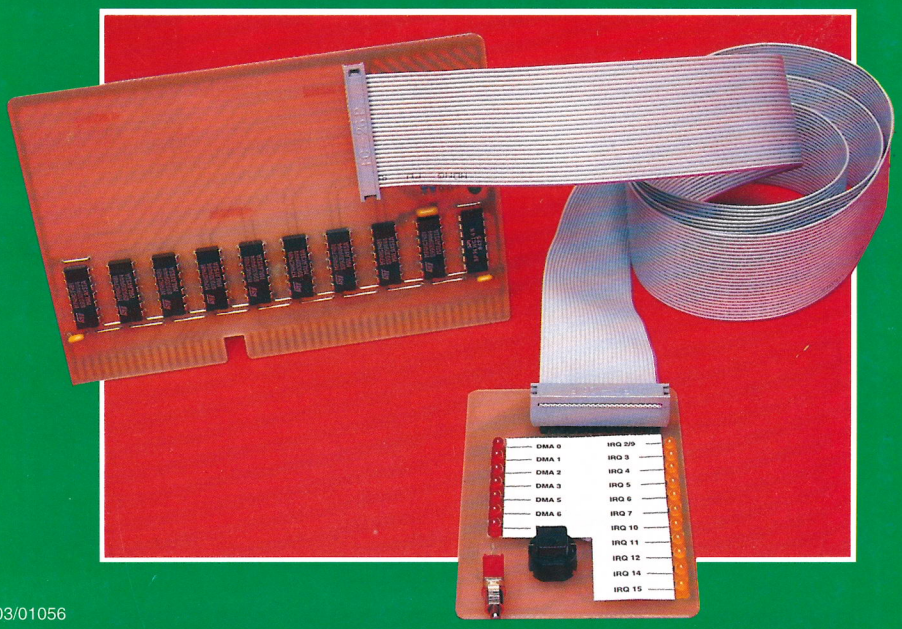
OCTOBER 1996 $5.50
NZ $7.20 incl GST

SONY'S NEW JA3ES MINIDISC DECK:

"Most outstanding consumer recorder ever..."

PROJECTS TO BUILD:

- PC-BUS 'SLEUTH' CARD FOR RESOLVING IRQ & DMA CONFLICTS
- 32-CHANNEL 40MHz PC BASED LOGIC ANALYSER



9 312966 123633 10

Print Post Approved PP255003/01056

Electronics AUSTRALIA with Professional Electronics & ETI

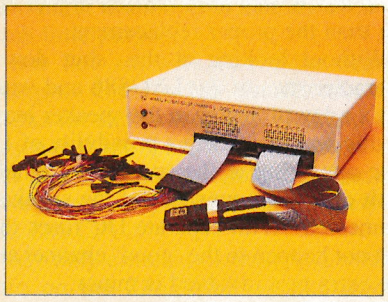
Volume 58, No.10
October 1996

AUSTRALIA'S LARGEST SELLING ELECTRONICS MAGAZINE — ESTABLISHED IN 1922

But what of the *quality*?

There's been a lot of media coverage about the range of programming on the new Pay TV networks, but very little indeed on the actual image and sound quality they're delivering. We asked Barrie Smith to see what he could find out, and the results make interesting reading. See page 22.

40MS/s Logic Analyser



If you need a logic analyser for troubleshooting in digital circuits, commercial models carry price tags starting in the thousands of dollars and rising rapidly — especially if you want 32 input channels and sampling at up to 40MHz. But you can now build one yourself, for only a couple of hundred dollars. See page 90...

On the cover

Sony's new JA3ES MiniDisc Recording Deck features the latest ATRAC technology. Reviewer Louis Challis found its performance so good that he's rated it the best piece of domestic audio gear he's ever tested. (See page 10). Also shown is our exciting new PC Bus Sleuth project. (Photos by Phil Aynsley, Ben Duncan.)

Video and Audio

Features

Projects and Technical

Columns and Comments

Departments

Professional Electronics

Construction Project:

A PC-BASED 32CH LOGIC ANALYSER - 1

PC-based test instrument designs make a lot of sense, because they can often deliver a high order of performance at a significantly lower cost. Here's a PC-based Logic Analyser which offers 32 TTL/CMOS input channels, sampling at up to 40MHz and fully maskable 32-channel triggering — all under software control.

by DAVID L. JONES and DAVID BULFONI

Although many readers of *EA* would be aware of what a logic analyser is and what it does, how many actually *have* one — or have even used one?

I wouldn't be surprised if I didn't see too many hands for the second part of this question, and even fewer for the first part. Why? Because the humble logic analyser is quite an exotic, expensive, and sometimes intimidating piece of test gear.

Just what *is* a logic analyser, anyhow? Well, the simplest answer is that it can be considered as a bunch of logic probes, all simultaneously capable of reading digital data and storing it in memory. The captured data is then displayed as a series of digital 'waveforms' which show the time relationship between the digital input signals.

Some new (read expensive) logic analysers also have very fast analog to digital converters to provide an oscilloscope type display of what the *actual* waveshape is, as well.

The main use of a logic analyser is when you need to measure more than a few digital input signals simultaneously. In fact, nothing can really take its place in this kind of situation.

With the proliferation of microprocessor controlled equipment these days, debugging the hardware aspects of new designs can be a nightmare without the use of a logic analyser.

But what makes the logic analyser so special, and better than an oscilloscope and/or logic probe for some digital troubleshooting? There are three main reasons, the first being that the logic analyser is capable of capturing single-shot or 'non repetitive' data, that can't be viewed on a normal non-storage oscilloscope. Of course, a digital storage oscilloscope is capable of capturing single-shot digital events, but they are generally expensive and can only handle a few channels at most.

The second advantage of a logic analyser is that it can capture dozens and even hundreds of channels at the same time. This allows you to analyse different events happening at the same time, and shows the relationship of one event to another. It also allows you to analyse the data and address buses of microprocessors.

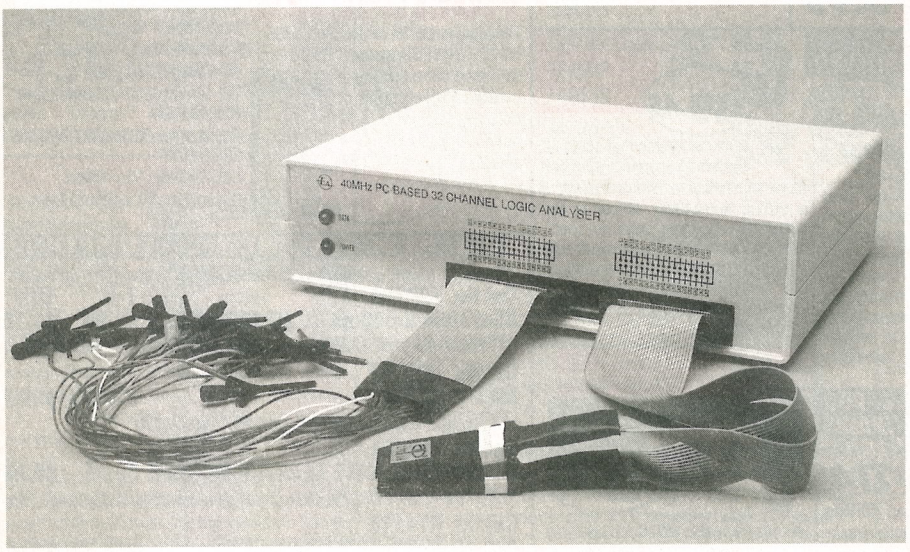
The third advantage is that a logic analyser can be triggered on a certain 'event', or combination of input conditions. For example, you can set it up to trigger off a certain microprocessor memory address, and capture the data before and after that address is accessed.

As good as a logic analyser is, many people still tend to use just a logic probe and oscilloscope for digital troubleshooting and circuit verification most of the time. This is because a logic analyser is generally quite 'fiddly' to use and time consuming to set up, especially if you know exactly what the circuit is supposed to do.

But when it comes to multiple channel non-repetitive digital signals, the logic analyser will become one of the most valuable piece of test equipment on your bench.

Projects for test equipment have been very popular in *EA* over the years — which is not surprising, considering that it's one field where you can still make a lot of your own equipment for much less than the cost of commercial units. But because of its cost and complexity, the logic analyser has been the one noticeable exception.

Commercial logic analysers tend to start in the four-digit price range and work their way up. One of the main rea-



The author's prototype for the new Logic Analyser. With 32 fully maskable input channels and able to sample at up to 40MHz, it's very suitable for troubleshooting in many digital systems.

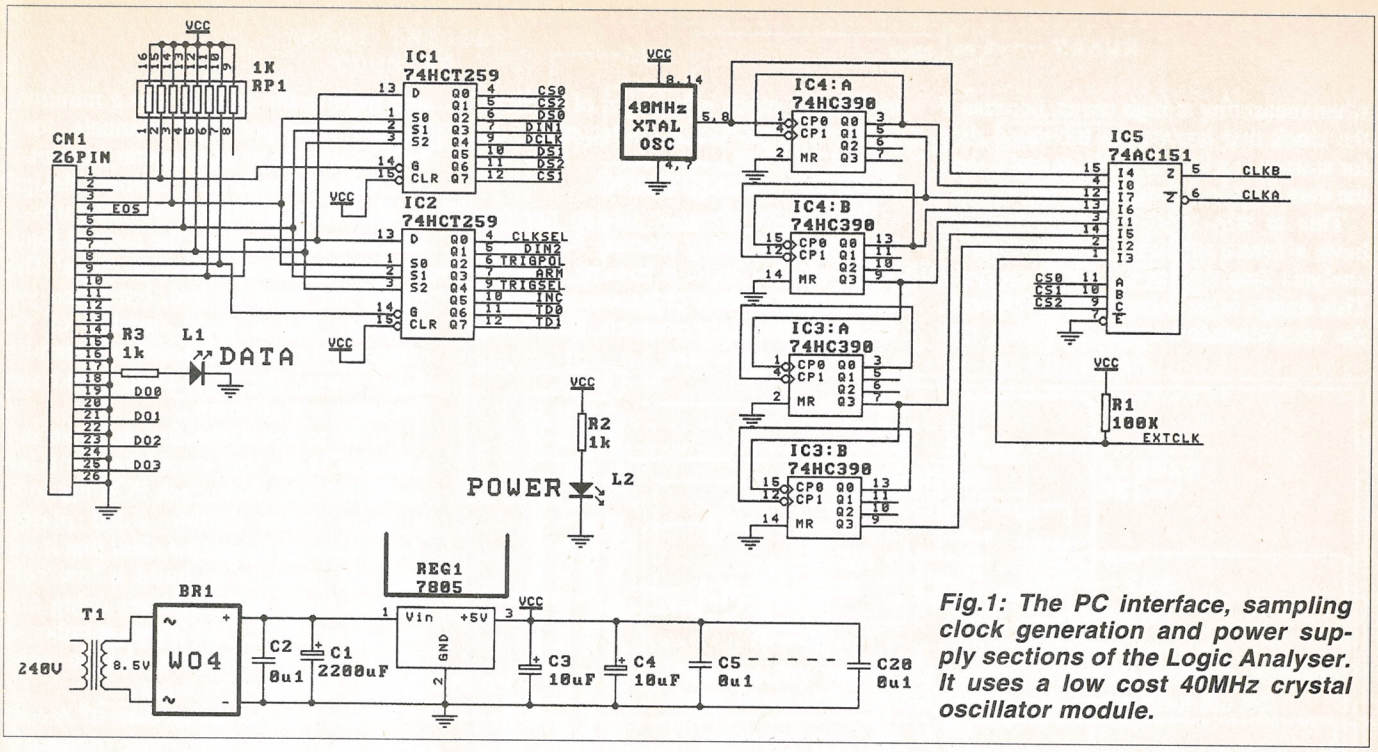


Fig.1: The PC interface, sampling clock generation and power supply sections of the Logic Analyser. It uses a low cost 40MHz crystal oscillator module.

sons for the high price is that most commercial units have their own screen and firmware to drive it, which puts the price out of the reach of the hobbyist — and many organizations as well!

In fact, though, it's not very difficult to design your own logic analyser. The basic operation is quite straightforward; all that is needed is some memory, a word trigger circuit and some simple control logic. A standard PC can be used to retrieve the stored information and display the required waveforms.

So why hasn't a simple logic analyser design been published before? The main reason is the sheer number of chips needed. To be really useful, a logic analyser must have many channels. Eight or 16 channels is fine for simple circuits, but if you're dealing with microprocessor buses, then 32 channels is much more useful. So, although the basic concept is simple enough, multiply it by 16 or 32 channels and you have a lot of circuitry!

New design

The new EA PC-Based Logic Analyser (or PCLA) to be described here, while being very simple in concept, has most of the features required for serious debugging of most digital circuitry. It provides 32 TTL-level input channels, as well as external clock and trigger inputs. The external clock and trigger inputs can be inverted under software control. The trigger circuitry is capable of setting each of the 32 channels to trigger off a HIGH, LOW, or DON'T CARE condition. The trigger signal can also be delayed by two, four, or eight clock cycles.

There are seven different internal sampling rates of 40MHz, 20MHz, 10MHz, 5MHz, 1MHz, 100kHz, and 10kHz. The maximum sampling rate is 40MHz internal and 20MHz external (more about this later).

At the same time the project is simple to construct, on one single-sided PC board. It connects to a standard PC parallel port, which controls all aspects of the project. As such, there are no front panel switches or controls. This also lowers the cost of the project, as a control chip is cheaper than a switch and wiring.

Two 34-way levered IDC header connectors are used for the 32 input channels (16 each), along with the external clock and trigger inputs. A five volt supply output is also provided on the connector to allow the addition of external level converter and buffer probes.

SPECIFICATION

Number of input channels:	32
Compatibility:	TTL/CMOS
Control:	Fully PC controlled
Sample Rate	Internal: 40, 20, 10, 5 and 1MHz, 100 and 10kHz
	External: Up to 20MHz positive/negative (see text)
Triggering:	Latched trigger word/mask
	External — positive/negative
	Internal — 32 channels, fully maskable
	Optional glitch capture (see text)
Software:	Address/data disassembly Measurement cursors
Options:	External buffer/trigger boxes

Types of analyser

Before we go any further, it's important to understand the two different types of measurements a logic analyser can make. One is called TIMING analysis, and the other is called STATE analysis. They both capture basically the same information, but it depends on what you want to measure that determines what mode you will use.

The simplest description is that TIMING analysis is *asynchronous* to the data being measured, and STATE analysis is *synchronous*. In other words, TIMING analysis uses an internal sample clock independent of the circuit under test, whereas STATE analysis uses a sample clock from the circuit under test.

So why not just call the two type of analysis internal or external clock? The terms come from the way the captured data is interpreted. In TIMING mode, you can actually measure the TIME difference between two data points or events, exactly like an oscilloscope. In STATE mode you are looking at what is happening in the circuit as it changes from one state to another, in step with the master clock from the circuit under test.

STATE analysis does have one disadvantage, in that it is difficult (read 'complex and expensive') to design the logic analyser to match the delay of the sample clock and the data. This is important

when measuring at high speed, because if the clock that latches the incoming data for storage is a little too late or early, then the data may no longer be valid.

This is why logic analysers usually have a lower specification for STATE analysis than that of TIMING analysis. This design is no exception, and as such, the recommended upper sampling rate in STATE mode is half that of TIMING mode. (More about this later, too.)

Design background

When I set about to design my own logic analyser using discrete 74xx series combinatorial logic, it became clear that a minimum of three chips was needed for each group of eight channels, just to store and retrieve the data — plus a minimum of five chips per eight channels, for a fully maskable word triggering circuit.

I really wanted a 32 channel design, so this meant a total of over 32 chips — and when you include the control logic, it's getting close to 50 or so chips! This might not seem so bad, but when you consider that a good number of the chips need to be connected with parallel 8-bit buses, the PCB design and size becomes a nightmare...

I tried to break the design down into separate PCBs that would handle 8-bits each, but even then, each PCB has to be a quite dense double sided plated-though board. It became clear that it was just not possible to build a low cost, easy to build logic analyser using normal 74xx logic. The sheer number of chips and the complex double sided PCB(s) simply ruled it out.

So how has *this* design done it? A complete 32 channel fully mask triggerable, 40MHz, delayed triggering, PC controlled logic analyser that fits onto ONE self-contained single sided PCB? OK, so the design doesn't use all discrete logic, but I'm sure you'll agree it was worth it.

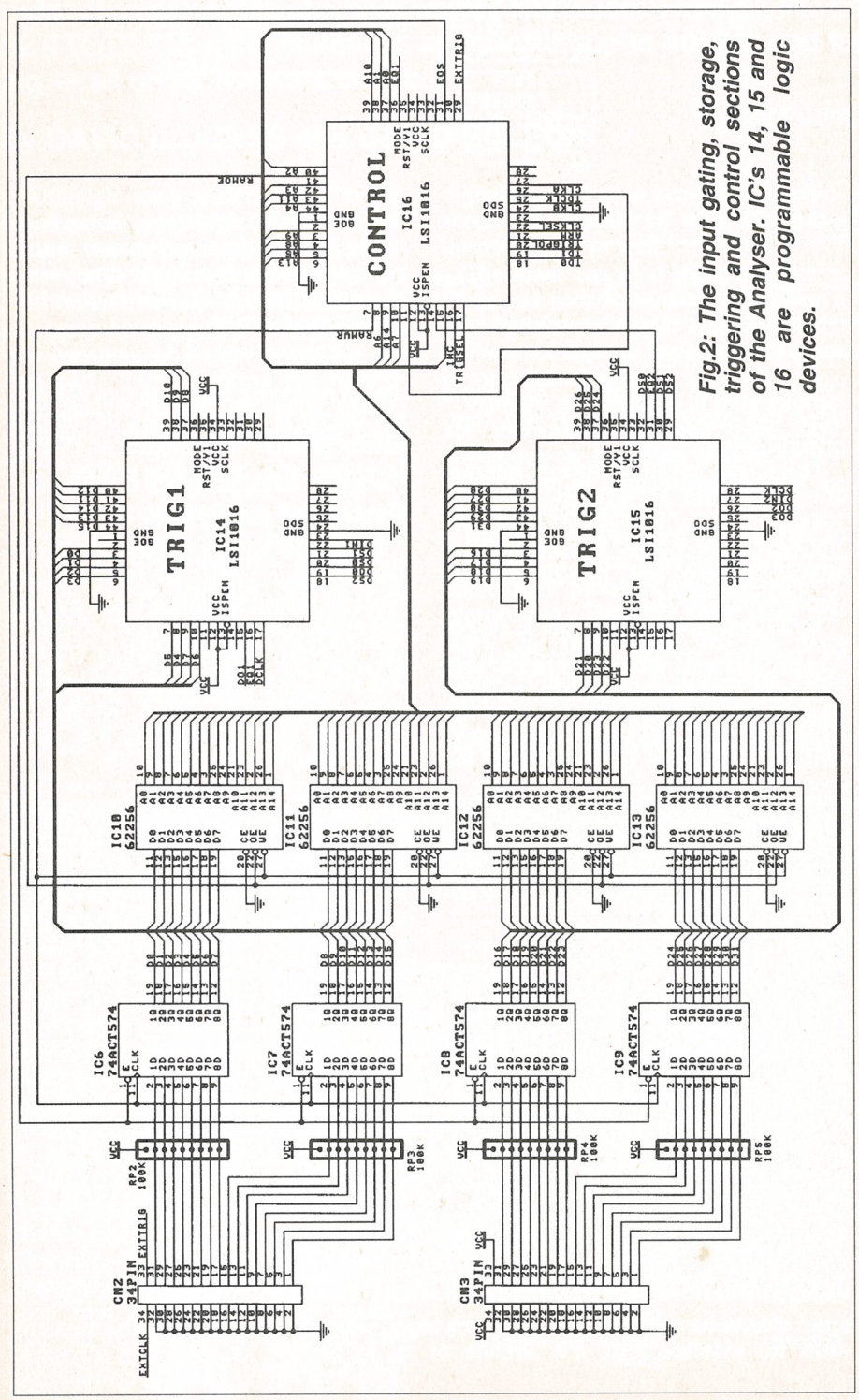


Fig.2: The input gating, storage, triggering and control sections of the Analyser. IC's 14, 15 and 16 are programmable logic devices.

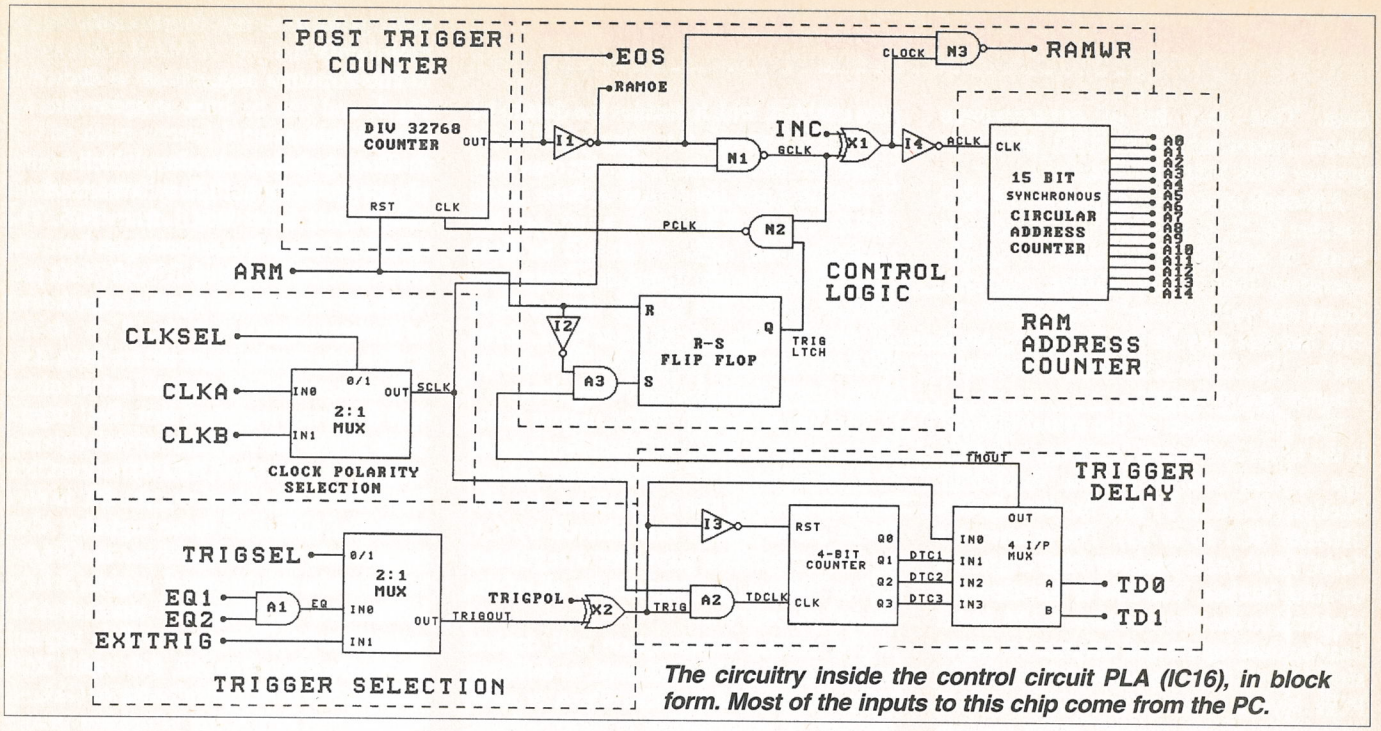
Programmable chips

If you take a look at the main circuit (Figs.1 and 2), you'll see how the design has been compressed onto a single sided PCB. Three Lattice Semiconductor large scale integration (LSI) programmable logic devices (PLDs) have been used for most of the control circuitry and all of the maskable trigger circuitry.

The Lattice 1016 PLD's contain the equivalent of about 2000 standard logic gates. They are also fully in-system programmable, which means that they can be re-programmed whilst in circuit using a dedicated serial bus. This feature is not included in this design however.

The good thing with this arrangement is that virtually no hardware is required to program the devices, they are simply connected to a 5V supply and connected to a normal PC parallel port. They are then programmed using the download software available from Lattice.

There are actually two versions of the LSI device. One is the ispLSI1016 which is in-system programmable, and can be reprogrammed over a 1000 times. The other is the LSI1016 which must be programmed with a commercial programmer, and is only programmable once. There is a substantial cost saving by using the one-time programmable LSI1016, but if you program it incorrectly or want to



The circuitry inside the control circuit PLA (IC16), in block form. Most of the inputs to this chip come from the PC.

modify it, then you're up for the cost of a new chip! So here it is preferable to use the reprogrammable ispLSI device...

The bottom of the range 1016 (ispLSI1016-80) is rated at 80MHz, which is more than adequate for our 40MHz application.

Most of the 44 pins on the 1016 are programmable as either inputs or outputs (or both) — which provides flexibility in PCB design, and is the main reason why the design can be fitted onto a single sided PCB.

The use of PLDs also reduces the cost of the project, even if you just take into account the cost of the chips that they replace. This is not to mention the cost saving on the PCB.

However, the other great advantage of PLDs over discrete logic is that if you have multiple gates connected in series, the total propagation delay is much smaller with the PLD. This is because the PLD connects all the gates together very closely on the same piece of silicon. Whereas with discrete logic, the propagation delay increases proportionally with each gate added.

This aspect of PLDs is very important for the trigger circuit, as the final trigger signal has to have passed through about seven gates before it registers the 'trigger' in the trigger latch. Even using very fast and expensive '74F' series logic, it would be hard to make a 40MHz analyser. But the PLDs lower this propagation delay to effectively two gates.

Fully pre-programmed PLD devices for this project are available (see elsewhere), or you can program and modify your own devices using the low cost (about $150) PDS LSI starter kit, available from Lattice Semiconductor dealers. Full internal schematics are provided here, and the actual LSI design files are available for downloading via the EA BBS or on disk from the *EA* Reader Information Service.

Circuit description

The main schematic of the discrete logic can be broken down into four main sections. The first section is the parallel port interface, comprising ICs 1 and 2 (Fig.1). These 74HCT259 devices are 8-bit addressable latches, which are used to give a total of 15 control lines to operate the PCLA. This was required because the parallel port only has 12 output lines, which was not enough for the functions I wanted to add. They also have one other purpose in that they provide a debounce function for all of the control lines. This is needed to make the lines immune to 'ringing', which can cause multiple clocking on those lines that are used as clock inputs.

It would have been nice to incorporate the functions of IC1 and IC2 into the main control LSI, but the lack of additional internal resources forced the use of external chips. Anyway, you didn't want EVERYTHING to be inside those little square chips, did you?

The second section of the circuit is the clock generation and selection section, comprising ICs 3, 4, and 5. The purpose of this section is to provide a range of clock frequencies which can be used as the sample clock in TIMING mode. Alternatively, an external clock can be selected, for use in STATE analysis mode. The 74HC390 devices used in the divider stage are dual decade counters. Actually, each four bit counter consists of a separate /2 and /5 counter. The output of the /2 counter is tied to the clock input of the /5 counter to produce a decade counter.

A TTL crystal oscillator module is used to generate the 40MHz master clock which is fed to the first clock input of IC4a, to produce 20MHz and 10MHz clock outputs on pins 3 and 5 respectively. The 10MHz clock then clocks the next counter, which gives outputs of 5MHz and 1MHz; and so on with IC3, which generates a 100kHz and 10kHz clock. The PCB can cater for both 8 pin and 14 pin oscillator modules.

All of these clock frequencies are fed into IC5, a 74AC151 8-channel multiplexer controlled directly from the PC with control lines CS0, CS1, and CS2. The multiplexer generates both normal and inverted outputs of the selected clock signal, CLKA and CLKB. The polarity of the output makes no difference in TIMING mode, because the clock is asynchronous with the input data. But in STATE mode, when the external clock input is selected, the polarity is important as this determines what edge of the clock the data will be sampled on.

A 74HC151 can be used instead of the AC series device is desired, but the HC device will provide greater 'skew', or delay between the external input clock and the actual sampling of the

data. This may be of consequence when operating in STATE mode at greater than 10MHz or so.

Although many brands of the 74HC390 counter (IC3, 4) are not 'guaranteed' to work at 40MHz, most have a typical operating frequeny of 50MHz. I haven't found one yet that won't work in this circuit, but this may be something to watch out for. The 390 is not readily available in a higher speed version.

The third section of the circuit (Fig.2) is the input buffering, latching, and storage section comprising the four 8-bit latches (ICs 6 to 9) and the four 32KB SRAMs (ICs 10 to 13). The latches are 74ACT574 octal positive edge triggered tri-state latches. The 574 is functionally identical to the more common 374, but it has the data inputs and outputs on opposite sides of the chip — which made the PCB layout easier. The ACT series is essentially a low power CMOS version of the FAST series used in very high speed designs. 74HCT574 devices can be substituted for the ACT devices if necessary, but the HCT574 is not guaranteed to work at 40MHz.

ACT devices should be used instead of AC devices, as this provides compatibility for both CMOS and TTL inputs.

The 100k pull-up resistor on each input line is used to keep unused inputs from 'floating', which may cause undesirable effects. This also sets the effective value of the input resistance.

The latches capture the input data on the positive edge of the clock input, which is common to all four 8-bit latches. During sampling, the tri-state latches are permanently enabled by bringing the common enable line low. This also means that the RAM outputs must be disabled during sampling, otherwise both the latches and RAMs will be outputing data at the same time. To do this, the OUTPUT ENABLE line of the RAMs must be held high during sampling (the inverse of the latch enable line).

Writing the data into the RAMs is accomplished when the WRITE ENABLE line of the RAM is LOW. This must occur some time after the data has been latched, to allow for the propagation delay of the data through the latches. This is easily accomplished by tying the latch clock to the WE of the RAM. In this way, the data is latched and allowed to settle on the positive part of the clock, and then written into the RAM on the negative part of the clock.

During data retrieval the latch outputs are disabled (tri-state) and the RAM outputs enabled, which enables the data to be read back through the trigger LSI chips which are also connected to the data bus.

The SRAMs used here are the same 32KB cache RAMs as used on PC motherboards, so there should be little problem obtaining them at a very reasonable price. The RAMs should have an access time of 20ns or less. 25ns SRAMs have not been tried, but should work from looking at the data sheet.

The fourth section of the circuit is the power supply (Fig.1), which comprises a 2155-type 8.5V transformer whose output is full wave rectified by the WO4 diode bridge and filtered by C1. A 7805 regulator provides a regulated 5V at 500mA, as required by all of the digital circuitry.

The current consumption will depend on the brand of SRAMS used, as different types can range from 50-100mA or more per chip. A small finned TO-220 heatsink must be used for the regulator, to provide adequate heat dissipation. A larger heatsink may be required depending on the voltage at the input to the regulator.

The rest of the circuitry is contained within the three LSI chips (Fig.2), which we'll now look at more closely.

LSI control chip

The main control LSI (IC16) contains all of the control circuitry required to operate the PCLA. Most of the inputs to the LSI come directly from the PC, which has control over the functionality of the entire project.

The circuitry inside IC16 is shown in Fig.3, and can be broken down into six main functions :

1. Clock polarity selection, which is simply a two-input multiplexer which selects either CLKA or CLKB from IC5. The CLKSEL input comes straight from the PC. The output of the MUX (SCLK) provides the main sampling clock for the rest of the circuitry.
2. The trigger logic section, which does two things. The first is to select either the internal trigger from the trigger LSIs (EQ1 and EQ2), or the external trigger signal (EXTTRIG) from CN1. The TRIGSEL line from the PC is used to select either of those signals. AND gate A1 is used to combine the trigger signals from the two trigger chips, effectively creating a 32-channel AND gate. The MUX output is then fed into controlled inverter X2, a 2-input XOR gate. The trigger polarity signal (TRIGPOL) from the PC either inverts or buffers the TRIGOUT signal to always produce a positive-going master trigger signal called TRIG.
3. The trigger delay section. This takes the TRIG signal, along with SCLK, and can delay the trigger signal by 0, 2, 4 or 8 clock cycles. A 4-input MUX selects which of the four delay settings will be selected, using control lines TD0 and TD1 from the PC.

A delay of zero is chosen by just feeding the TRIG signal straight into the MUX. The other delay times are generated by a four-bit counter, which is incremented by SCLK only when the TRIG signal is active (HIGH) by virtue of AND gate A2. The counter is automatically reset when the TRIG line goes low, which restarts the delay counter ready for next time.

(The name TRIGGER DELAY is a bit deceiving, as the trigger signal is not always 'delayed'. If the trigger signal does not stay high for the duration of the delay time, then the trigger signal is ignored. This can be used to stop spurious signals from triggering the analyser.)

4. The RAM address counter, which as the name implies generates the address for all four of the RAM chips. It acts as a 15-bit circular counter, which means it never gets reset. This is to allow the data to slowly overwrite itself, which is needed to provide the PRE- and POST-trigger capability. The counter is of the synchronous type, which ensures that all of the outputs change at once.
5. The 15-bit post trigger counter, which is effectively a divide by 32,768 counter. The 15th output line goes high 16,384 counts after the trigger event,

Getting the PLDs and Software

The designers of this project have generously provided the internal coding for the PLD chips used in the design, for us to make available for readers wishing to program their own devices. The necessary files will be available on the EA Reader Service BBS for free downloading, and will also be available by mail for those who prefer to send a formatted HD 3.5" floppy disk, plus a cheque or money order for $7.50 (made out to Electronics Australia) to cover copying and return postage.

Alternatively, for those who prefer not to program their own chips, a set of three preprogrammed LSI1016 PLD devices will be available from Tronnort Technology, of 12 Copeland Road, Lethbridge Park NSW 2770. The set of three programmed chips is priced at $55 including postage.

Tronnort Technology can also supply the PC software needed to control and run the Logic Analyser, for only $35 including postage.

The software and programmed PLDs can be purchased together, for a discounted total cost of $80 including postage.

which stops any further sampling. This allows the RAM to be half filled with data before the trigger event, and half after the trigger event.

6. The main control logic, which coordinates all of the above sections to provide the functionality of the logic analyser.

To understand how it all works, let's start by assuming that the ARM control line from the PC is high. This resets the post trigger counter and the RS flip-flop, which we'll call the trigger latch. The SET input of the trigger latch is held low by inverter I2 and AND gate A3. The trigger latch will therefore ignore all trigger signals coming from the trigger delay section via the TMOUT line.

With the trigger latch held in the reset state, the Q output (TRIGLTCH) disables NAND gate N2, keeping the output (PCLK) high. Because the post trigger counter is not incrementing, the End Of Sample (EOS) line will never become high. This will allow the sample clock (SCLK) to pass through NAND gate N1.

Assuming that the PC is keeping the manual address increment line (INC) low, the GCLK signal is fed straight through XOR gate X1, supplying the sample clock to the RAM address counter and the RAM write line RAMWR.

What this all means is that when the PC keeps the ARM line high, the PCLA is continuously writing data into the RAM, overwriting any previous data once the RAM address wraps around to the start again.

The CLK line of the input latches is connected to the RAMWR line. This allows the data to be latched on the positive edge of this clock and written into the current address of the RAM on the negative half of the cycle.

The RAM will continue to be filled with data, and when the address counter reaches 32,767, it will just loop back to the first address and continue to overwrite the old data in the RAM. This is called PRE TRIGGER sampling, as all of the current data in the RAM is before (or PRE) the trigger point. Effectively, you (and the software) never know, nor care what the current RAM address is; all the software needs to know is that there are 32,768 addresses.

The PC must keep the ARM line HIGH long enough for the entire RAM to be filled. This is to ensure that no data is left in the RAM from a previous aquisition. The software handles this automatically when using internal sampling as it knows the sampling rate and can thus determine the delay needed.

But when using an external clock, the PC has no way to know the sample rate and just adds a delay of one second.

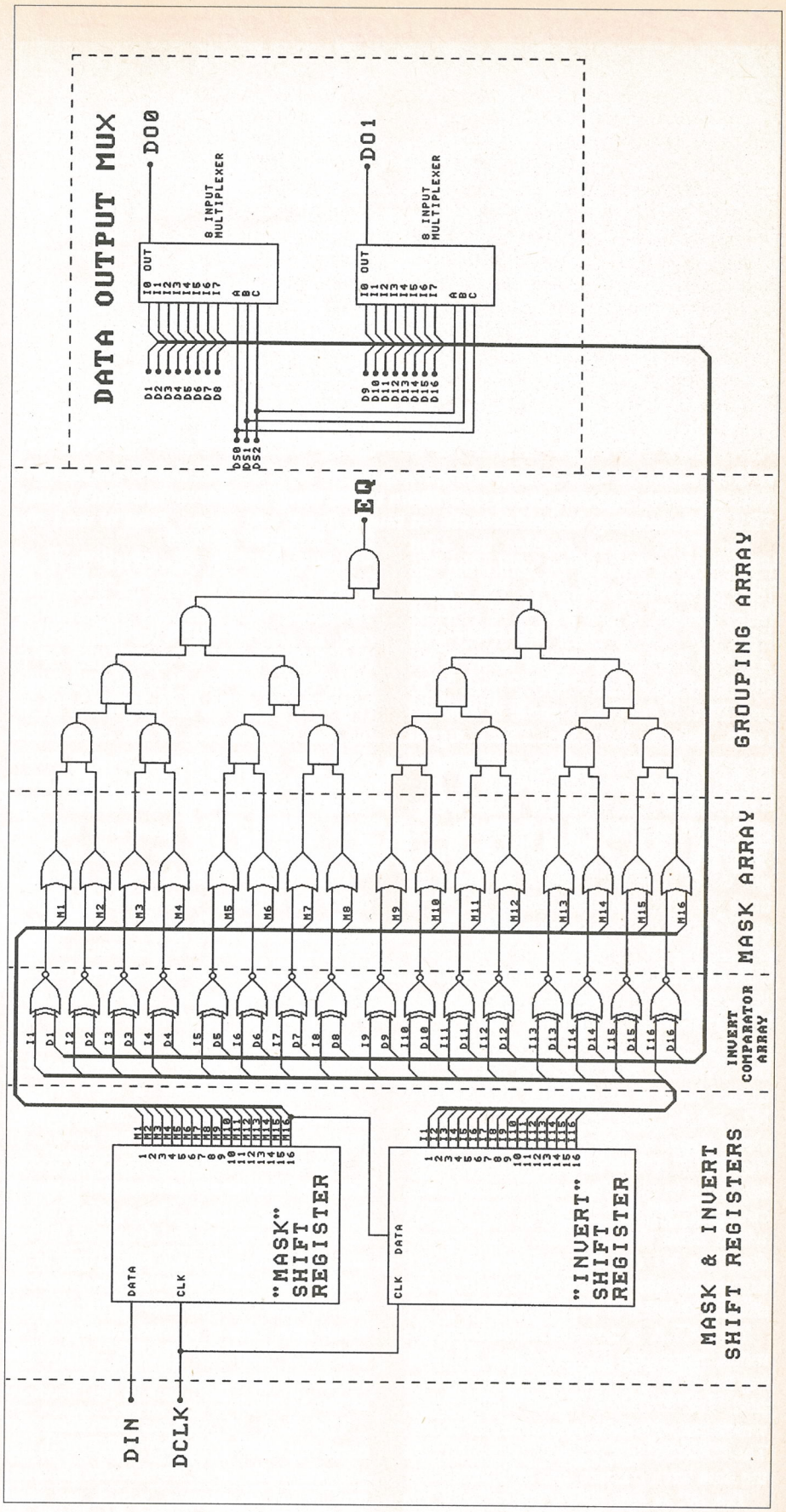


Fig.4: The circuitry inside each of the two trigger circuit PLAs (IC14 and IC15). Each chip provides a 16-channel fully maskable triggering circuit plus a pair of 8:1 data multiplexers, which are used to feed the sampled data back to the PC.

This delay is enough to handle a minimum external clock rate of 32kHz; any less than that, and you will have to manually hold off the trigger until the required time is reached.

When the PC sets the ARM line LOW, the PCLA can now accept a trigger signal by virtue of gate A3 being enabled. If both the EQ1 and EQ2 inputs go high, this means the input data has matched the trigger data in the two trigger LSIs, and a trigger condition is satisfied. Assuming internal trigger has been selected by the TRIGSEL line, the trigger signal (TRIGOUT) will go high and pass into A2 (TRIG). The trigger inverter X2 is only used for external trigger mode. When internal triggering is selected, the TRIGPOL line must be held low.

The TRIG signal enables the 4-bit trigger delay counter by removing the reset signal from inverter I3. The trigger delay counter is now able to be clocked from the sample clock via AND gate A2. The amount of trigger delay is set by the trigger delay MUX lines TD0 and TD1. If the first channel is selected (zero delay), then the TRIG signal bypasses the trigger delay counter and passes straight into TMOUT.

The other three channels select a trigger delay of either 2, 4, or 8 clock cycles. After the trigger signal enables the trigger delay counter, it starts to count up until the TRIG signal returns LOW. If the TRIG signal stays HIGH long enough for the counter to reach the value chosen by the MUX, then the TMOUT line goes HIGH and the trigger delay time period has been met.

When a trigger signal is received on the TMOUT line, the Q output of the trigger latch is set HIGH. This enables N2, which then supplies the sample clock signal to the post address counter. The PCLA is now said to be in the POST sampling mode. The post counter starts counting until it reaches 16,384, at which point the EOS line goes high — disabling N1 and N3, and stopping any further sample clocks from reaching the address counter or post counter.

The PCLA is now in End Of Sample mode, which the PC can detect by reading the EOS line on pin 15 of the parallel port. The RAMWR signal is now high, which puts the RAMs into READ mode. The RAMOE line is also low, which puts the data latch outputs into tri-state mode.

The PC is now able to read back the data from the RAMs, using the INC line to increment the RAM address counter. The RAM address counters are never reset when End Of Sample mode is reached, which means the first sample the PC reads back will be the oldest data sample in the RAM. The PC will continue to read the rest of the data until it counts to 32768, at which point the data will be the most recent sampled. The 16,384th data sample will be the trigger point.

After all of the data is retrieved, the PC sets the ARM line back high again, which restarts the whole cycle.

The PCLA therefore has three different modes of operation: ARM mode, where data is continuously sampled; POST mode, when the PCLA has been triggered; and finally End Of Sample mode, when the PC retrieves the data.

LSI trigger chips

Each trigger LSI chip (IC14, 15) comprises all of the circuitry for a complete 16-channel fully maskable triggering circuit, as well as two 8-bit multiplexers. One LSI is used for each group of 16 input channels. The internal schematic (Fig.4) is purely for the purpose of showing the functionality of the circuit, and does not necessarily match the actual internal layout of the LSI. This is because the LSI compiler is given Boolean equations, which it then generates into the most usable form for use within the LSI.

The purpose of each trigger LSI is to allow the software to individually invert, and mask (turn off) each of the 16 data inputs. This allows it to generate a trigger signal when the input data matches the exact data specified by the PC software.

Both trigger LSIs have exactly the same circuitry, but they do differ in their pinouts, and so are not interchangeable.

The internal circuit of each trigger LSI can be broken down into five separate sections:

1. The mask storage section, which contains two 16 bit cascaded serial-in parallel-out shift registers. This is effectively a 32-bit shift register fed from the PC via one data and one clock line. There is no need for a reset line, as the unwanted bits just 'vanish' off the end of the register when new data is pushed in. The PC is therefore able to directly write the required trigger mask information to each of the 32 outputs.

The first 16 outputs are used for masking each of the 16 channels in the sequence shown. Likewise for the second lot of 16 outputs, which are used for generating matching data for word comparison.

2. The invert comparators or word comparison circuit, which is simply 16 2-input exclusive-NOR gates, each of which acts as a 2-bit comparator. One XNOR gate is used for each of the 16 input channels. Input data is fed into one input of the XNOR gate (from the data lines, via the output MUX section) and the corresponding data line from the shift register is fed into the other input. If the data from the data register matches the input data, then XNOR gate will give a positive output. Likewise, if the register data does not match the input data, the XNOR output will be low. If all of the inputs match the 16-bit data on the data register outputs, then all of the XNOR outputs will be high, which signifies a match.

3. The masking array. Here the 16 channels from the comparator array are fed into 16 corresponding OR gates, along with the corresponding outputs of the mask shift register. This allows the PC software to individually mask out (or turn off) each of the 16 channels. A high output from the mask shift register will make the corresponding OR gate output high regardless of the data input. A low output from the mask register will allow the OR gate to pass the input data unaffected.

(Masking out a channel effectively 'matches' that input channel regardless of the incoming data. This allows the software to ignore certain input channels, and only trigger off the desired channels.)

4. The grouping array. This is effectively a 16-input AND gate, which generates a positive trigger signal EQ when all of the 16 channel inputs match the data register and/or are masked.

5. The data output MUX section. This final section comprises two 8-input multiplexers which are fed directly from the trigger LSI's data inputs (connected to the PCLA's internal data buses). The MUXes allow the PC to read back data captured in the RAMs. Each multiplexer handles 8 bits of data from its associated 16-bit data bus, and sends a single bit at a time directly back to the PC. Bit selection is handled by control lines DS0, DS1 and DS2, which come directly from the PC, and are common to both multiplexers.

The four multiplexed data lines DO0-DO4 (two from each LSI) are fed directly back to the PC parallel port on pins 10, 11, 12 and 13. The software therefore retrieves one bit of data from four channels at the same time. Eight more read operations are required at the same address in order to read all 32 channels.

Hopefully the foregoing should give you a good idea of the way the new PC-Driven Logic Analyser works. In the second of these articles, we'll describe its construction and use.

(To be continued.) ❖

TV & VIDEO ✲ AUDIO ✲ COMPUTERS ✲ AUTO ELECTRONICS!

SPECIAL FEATURE ON QUEENSLAND INDUSTRY

NDD

Electronics

AUSTRALIA WITH Professional Electronics & ETI

NOVEMBER 1996 $5.50
NZ $7.20 incl GST

TEK'S NEW TDS 220 LOW COST 100MHZ SCOPE, WITH 1GS/S REAL-TIME SAMPLING

Review inside...

'MIRACLE' ANTENNA FOR AM RECEPTION

9 312966 123633 11

ADAPTOR LETS YOU SAFELY VIEW MAINS CURRENTS

Print Post Approved PP255003/01056

Electronics

AUSTRALIA WITH Professional Electronics & ETI

Volume 58, No.11
November 1996

AUSTRALIA'S LARGEST SELLING ELECTRONICS MAGAZINE — ESTABLISHED IN 1922

No ordinary CD player

This month Louis Challis has been testing the new Onkyo Integra DX-7911 high-end CD player, which incorporates technology claimed to eliminate jitter and offer dramatically improved tracking. His test results confirm that this is indeed no ordinary CD player — see Louis' review, which starts on page 12.

An eye for your PC...

The new Compro D-Cam digital camera is relatively low in cost, and hooks up very easily to your PC via the Centronics printer port — so there's none of the usual hassles with IRQ's, DMA's or I/O address conflicts. It takes quite respectable 640 x 480 pixel colour images, too, as Graham Cattley found when he tried one out this month. You'll find his review starting on page 18...

On the cover

The new Tektronix TDS 220 and its 60MHz sibling the TDS 210 look set to woo many low-end scope buyers away from traditional analog models, as Tek has effectively halved the price of DSOs while simultaneously upping performance and reducing size by about 75%. See our review on page 77. (Photo by Phil Aynsley)

Video and Audio

Features

Projects and Technical

Columns and Comments

Departments

Professional Electronics

Construction Project:

A PC-BASED 32CH LOGIC ANALYSER - 2

Here's the second of two articles describing this new instrument, which offers sampling at up to 40MHz and fully maskable 32-channel triggering — all under software control. In this article the authors cover its construction, testing and use.

by DAVID L. JONES and DAVID BULFONI

The PCLA has been designed to be as easy to assemble as possible. All of the components, with the exception of the D25 connector, transformer, and two LEDs are mounted on a single sided PCB measuring 230mm x 115mm.

The project is neatly housed in an instrument case measuring 260 x 180 x 65mm. The PCB is butted up against the front panel, to allow the right-angle mounting test probe connectors to protrude from the front panel, eliminating any internal cabling. The only other components mounted on the front panel are two LEDs — one for power, and the other for data indication. A combined IEC mains input connector and fuseholder, along with a DB25 connector for the cable to the PC, are the only items mounted on the back panel.

After checking the PCB for the usual etching problems, work can commence on PCB assembly by installing the 50 or so wire links required. It pays to keep some of the longer links as straight as possible, to make the board look much neater. Take note of the angled link between the crystal and IC5.

Install the three resistors and five resistor packs next. Don't confuse RP1 (1k) with the other four 100k networks, and be sure to match the 'common' end of each network with that marked with a square on the overlay.

Now come all of the IC sockets. It is highly recommended that all of the ICs be mounted in sockets, especially the input latches and the RAMs. Using sockets, the repetitiveness of the circuit will make troubleshooting much easier at a later stage. About the only IC's which aren't either expensive or connected to an outside circuit which can cause damage are ICs 3, 4, and 5.

It may be difficult to obtain the 28-way 'skinny DIP' sockets required for the RAMs, but in this case two 14-pin sockets can be connected end to end. In fact, two 14-pin sockets may also be cheaper than one 28-way one.

The three 44-way PLCC sockets rate a special mention. Due to Murphy's law, it is extremely easy to solder in a PLCC socket the wrong way around! A PLCC chip will only go into the socket in one of the four possible orientations, so it is vital to get the socket around the correct way.

The PLCC socket has a bevelled edge on one corner, and this must match the bevelled edge on the component overlay. Also make sure that all of the pins protrude through their holes. If you are using a homemade PCB and drill a few holes off centre, then it can be very frustrating trying to insert the socket!

Install all of the capacitors next, with the exception of C1. Watch the orientation of the two tantalum capacitors. Also install the five PCB stakes and bridge rectifier.

Next comes the three IDC connectors CN1, CN2, and CN3. The best way to install CN3 is to keep the header plug attached to the header pins. This will keep the header pins straight and aligned while they are being soldered. CN1 and CN2 should also be bolted to the PCB with M2.5 nuts and bolts. This will stop pressure from being exerted on the solder joints when the test probe leads are connected and disconnected from the front panel.

Lastly, install C1 and the regulator. Bolt the regulator and heatsink to the PCB before soldering the leads.

Attention can now be turned to the mechanical side of things. If you don't have pre-punched panels, then cutouts will need to be made for the mains connector, DB25 connector, the LEDs and probe connectors. The only cutouts which will need alignment are those for the probe connectors. The probe cutouts can extend down to the bottom of the panel if you wish, but some may prefer to take a bit more time to make the cutouts only just big enough to accept the connector.

Holes will also need to be drilled for the PCB and transformer. The PCB should be as close to the front panel as possible, to allow the probe lead connectors to extend out. The PCB is just wide enough to fit between the two front support posts.

The transformer should be mounted in the middle of the back part of the case, with the mains input towards the rear panel. Securely connect the transformer frame to the main earth pin on the IEC socket. All mains wiring connections should be properly insulated with electrical tape and heatshrink.

Connect the 8.5V winding of the transformer to the AC input pins on the PCB, and connect the LEDs to their respective PCB pins.

The LEDs can now be wired to the PCB. Only three wires are required for this, one for the anode of each LED and their common ground connection. The cathode of both LEDs will have to be connected together on the front panel and connected to the ground lead.

Finally, assemble the IDC female DB25 connector and 26-way IDC header to a short length of ribbon cable, just long enough to connect from the PCB to the rear panel.

Before installing any of the ICs, apply power and ensure that 5V is available on the supply pins of each IC socket. The power LED should also be on. Disconnect the power, and install all of the ICs — paying close attention to the correct orientation. Be sure to use relevant anti-static procedures to avoid damaging any of the ICs. The LSI devices will be labeled 'LACC', 'LATC1' and 'LATC2'. Ensure that these devices are installed in their correct sockets as follows: LATC1 is IC14, LATC2 is IC15 and LACC is IC16.

Carefully check the orientation of

each LSI device before pushing it down into the socket. The bevelled corner on the chip must match the bevelled corner on the socket. Trying to force the chip in with the wrong orientation may damage the IC and socket pins.

Re-apply the power and check the 5V supply rail on all of the chips. If the heatsink gets overly hot, then something may be loading down the supply. If this is the case, then it is likely to be a PCB short somewhere. Start by removing the ICs one by one and repowering until you track down the fault.

Assuming that the hardware hasn't gone up in smoke, it's time to connect the PCLA to the PC. Use a 25-way ribbon cable with a DB25 IDC male on one end, and a DB25 IDC female on the other end. In fact, it is best to make up a 'universal' cable with both female and male DB25 connectors and a 26-way IDC DIL header on each end. This will also you to use the cable for all sorts of other projects as well. The cable should be kept as short as possible, with an absolute maximum length of about two metres.

Probe construction

One of the most difficult and expensive parts of building a logic analyser would have to involve the test probes. Unlike an oscilloscope or multimeter, the PCLA has 32 inputs, and if you want to fully use all 32 channels, then you will have to make 32 or more test probes!

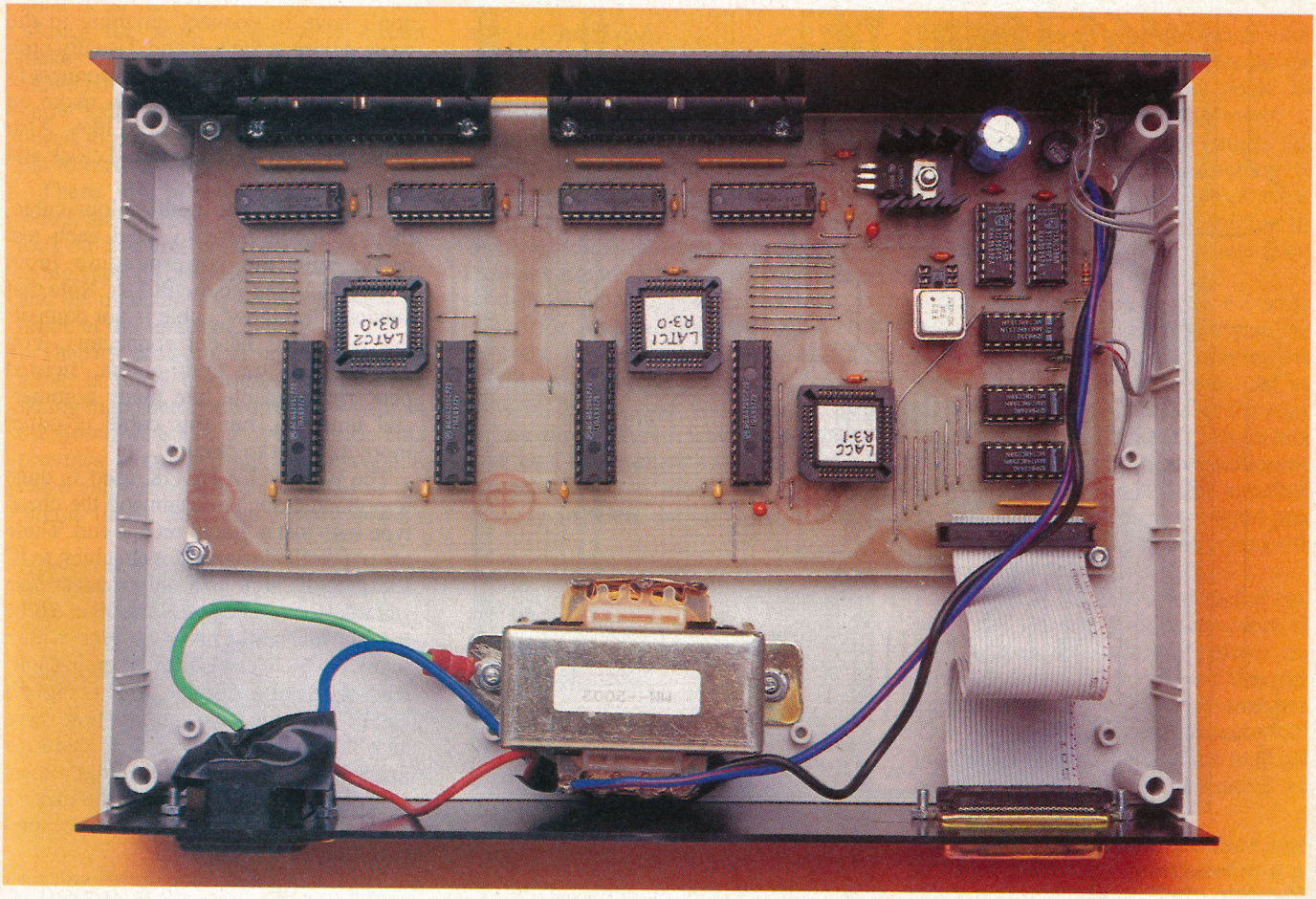
Like most commercial logic analysers, the PCLA uses levered dual row IDC header connectors for the probe inputs, which provides a low cost and compact solution. Unfortunately, there is no standard pinout or connector size for logic analysers. Almost every manufacturer uses their own custom pinout, so don't rush out and buy a set of Joe Blow logic analyser probes...

The PCLA has two 34-way connectors which provide all of the 32 channel inputs, external clock, external trigger, ground and an external +5V supply signals. The front panel label shows the pinouts of the two connectors. The pinouts are the same for pins 1-32 on both connectors, but pins 33 and 34 differ. One connector provides the clock and trigger inputs, while the other provides a +5V output on both pins.

So what kind of probes do you need? It all depends on what you want to measure. The photographs show two of the most common types of probes. One of them uses a commonly available 16-way DIL test connector, which simply clips over a DIL IC in-circuit. This sort of probe is useful for quick testing of all the pins on a single IC.

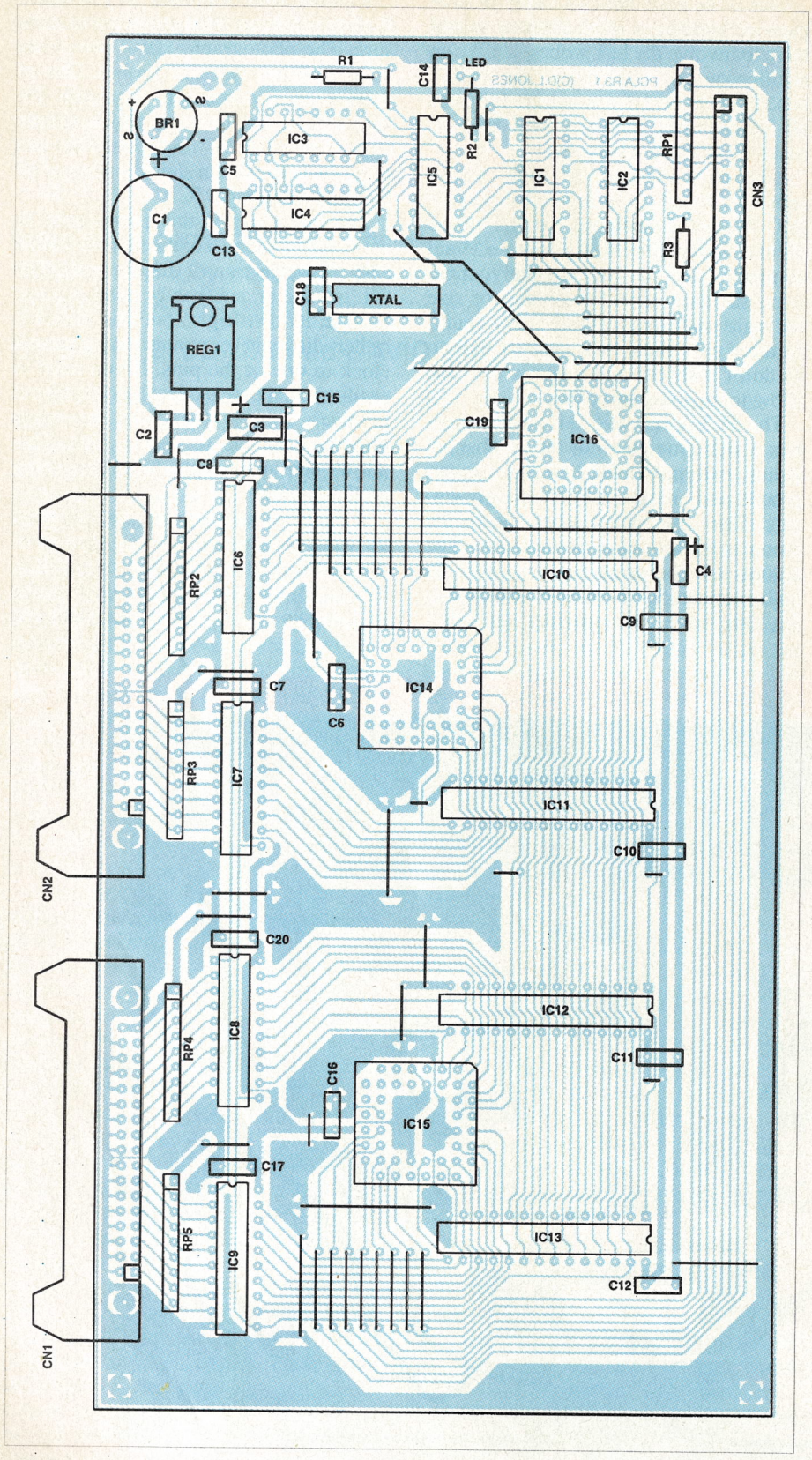
The prototype of this probe used 8-way SIL header connectors to allow easy disconnection of the test head. The SIL connector can then be used as a test head in its own right, but this makes it rather difficult to connect an external clock to one of the pins. If you need to easily connect the external clock, then it is much simpler to solder the wires directly onto the test head pins. This allows you to connect a clock line using an EZ-HOOK.

It might be wise to make up a 28-way, 20-way, 16-way and 14-way version of this test head, for general testing of ICs. With 32 channels available on the PCLA, two 16- or 14-way test heads can



Inside the new Logic Analyser. Thanks to the use of three preprogrammed PLD chips, everything fits on a single-sided PC board. The input probe leads connect directly to the two 34-way IDC DIL connectors visible at the upper left, which protrude through the front panel.

A PC-Based 32Ch Logic Analyser — 2



Use this overlay diagram as a guide, along with the interior photo, when you are wiring up the PCB for your own analyser. The author recommends that sockets are used for all ICs, and suggests that you take particular care with the orientation of the sockets for the PLD chips IC14, IC15 and IC16.

be used at the same time.

When making one of these connectors, it is recommended to match the actual IC pin number with the corresponding input channel on the PCLA connector. This just makes it a lot easier to remember which input goes to which pin, when using the software.

The other photograph shows a 16-channel EZ-HOOK test probe. I actually had a good quantity of these premade leads in my junk box from an old logic analyser, complete with 'claw' type EZ-HOOKs. They used 10-pin header connectors, so I just snipped off the connector and attached the individual leads to a short length of ribbon cable. The probes are arranged in the resistor colour code sequence to aid in identification.

You can make your own EZ-HOOK probes using a 34-way IDC DIL header plug, about 300mm of ribbon cable and a few dozen EZ-HOOKs. It's a simple matter of connecting the IDC connector to the ribbon cable, and then connecting an EZ-HOOK to each input wire.

When splitting the cable into individual wires, leave a ground wire attached to the side of each input wire — but you don't have to connect anything to the ground wire at the test end. Usually only one ground wire is enough to connect to the circuit, although you can make up extra ground leads if you like. Also make up leads for the external clock and trigger inputs.

If you don't use multi-colour cable, then you will have to label each test probe with the corresponding input channel number on the PCLA. Note that standard 'rainbow' cable is not compatible with IDC header connectors. You can also use the multi-colour 'twisted pair' ribbon cable like many commercial units do, but this cable is rather more difficult to obtain.

The type of EZ-HOOK used is quite important. I don't recommend the cheap type with just a 'hook' on one end. These are not really designed for attaching to IC leads; their large size only makes them suitable for resistors and the like. To be useful, you really need the type with a 'claw', which can securely grab an IC lead with little chance of slipping off and/or shorting two pins. These type aren't exactly cheap, but then again it's never going to be cheap to build a logic analyser probe! If you want to make a good range of test probes, then be prepared to spend over $100.

These general purpose probes will cater for most of the common circuits, but if you want to test surface mount or large PLCC packages, then you'll have little

choice but to make up specialised probes.

It may be handy to make up some probes with a two-pin header on each of the channels. This is a common type of probe used in a lot of commercial equipment. In fact, many designs have built-in logic probe test points, usually of the two-pin header variety, where one wire is the signal, and the other a ground.

If you are designing a circuit of your own and think you may use the logic analyser for evaluation or troubleshooting, then it is a wise idea to add some two-pin header logic probe test points. As header connectors are very cheap, the main problem will either be lack of space for the connector, or lack of room to route the tracks to the connector.

If you've ever seen a commercial PCB with rows of unused two-pin connectors, or PCB pads without any connector, you'll now know what they are for!

Who knows, the way things are going, TV and VCR service manuals of the future may have logic diagrams instead of CRO shots...

Taking measurements

Using a logic analyser has many traps for the unwary. Unlike an oscilloscope, there is no easy way to tell what effect adding the logic probe to a circuit makes. The only output you have is either HIGH or LOW, and if you don't know exactly what the circuit is supposed to do, then you have little choice but to trust the logic analyser.

There are however two problems which are the most common and easy to look out for. One is the use of the PCLA at high speeds (greater than a few MHz), and the other involves data misinterpretation at the sampling point.

The first problem, of high speed operation, is fairly obvious. Usually when taking high speed measurements with an oscilloscope, a x10 probe is used. This provides a much lower probe capacitance and hence less disturbance of the circuit under test. The PCLA doesn't have the same capability of using a x10 low capacitance probe, and all measurements are taken with a directly connected input. In our case, we're using a 74ACT series gate. Here the amount of probe capacitance will depend on the length and type of cable being used.

The probe capacitance has a dramatic loading effect on high speed circuits, so this is something to be wary of if you are not getting the result you expect.

The rule is to either keep the cable as short as possible —generally less then 300mm — or to use a buffered probe. A buffered probe is an external box with a buffer chip as close to the probe inputs as possible. You can then use a much longer cable to connect to the PCLA. The PCLA provides a +5V output which can be used to power external buffer probes.

Another use of a buffer probe box is to provide different types of input matching. The PCLA has 74ACT series logic on the input, and can therefore accept either TTL or 5V CMOS signal levels. If you want to measure 4000 series logic of a higher voltage, or the newer 3.3V logic, or even ECL, then you will have to make an appropriate buffer box. The box should convert the incoming signal to a TTL level acceptable for the PCLA.

The second problem concerns the fact that the PCLA has to 'latch' the data at some discrete point. If the input happens to be changing from one state to another at this same point, then the PCLA will not register the data correctly and may interpret the data as either logic level. This can readily be seen by measuring a constant square wave of lower frequency than the internal sample clock.

Say for example that the input is a constant 10MHz square wave, and the PCLA is using its internal clock of 40MHz. In this case you would expect to see the captured waveform change state every four clock samples, and this is what you get — most of time!

Every so often, the input may change state right at the point the data is clocked into the latch. If the latch misinterprets the data, then the display will show the input pulse longer or shorter than it really is. There is no real way to avoid this when using the internal PCLA clock (TIMING mode), so be wary of it.

In effect, this is similar to the quantisation error in A to D converters. Just as they can only measure a finite number of levels, so too the PCLA in TIMING mode will always have a probable error of one sample clock period.

A similar problem exists in STATE analysis mode, but this time it is possible to prevent it. Because the external clock on the PCLA has to pass through IC5 and the control chip, it will inevitably have some delay or 'skew' associated with it compared to the input data.

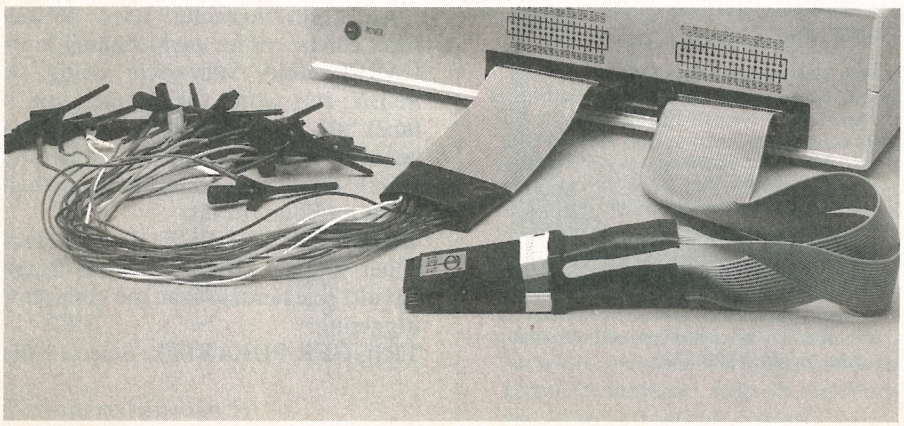
If you are not getting the results you expect in STATE mode, then try inverting the external clock polarity in the software. This will usually fix the problem, but at high speeds the skew may be too much to capture data reliably. This is why the PCLA is arbitrarily limited to a speed of 20MHz in STATE analysis mode.

You can push the speed to the full 40MHz if you feel confident enough that the clock skew will not be a problem. Alternatively, you can use an external buffer probe to try and delay the data to match the delay of the clock. But you're on your own with this one!

Expensive commercial units get around the problem by switching the external clock straight through to the latch clock input. However, adding this feature to the PCLA would have overly complicated the design.

It is also important to select a proper ground when measuring high speed signals. In such circuits, moving the ground lead just a few inches can add significant 'ground bounce' to the signal being measured. Although this is more important (vital!) with an oscilloscope than a logic analyser, it's still something to watch out for. It's always good practice to connect to the ground pin of the chip you are testing.

The PCLA triggers off data fed from the input latches. This means that the PCLA does not have any 'glitch capture' capability like some of the more expensive commercial logic analysers. This may be a problem if you are looking for a particular fault which may only occur for a short period of time. If you



For general work, two different kinds of input probe are most useful: a set of individual EZ-HOOK leads like those at left, and a 16-pin DIL IC clip like that at right.

A PC Based 32-Ch Logic Analyser — 2

set the PCLA to trigger off the fault condition then depending on the sample rate, the input can match the fault condition, but if the data is not being latched and sampled at that time, the trigger circuit will never see it.

However, there is a partial solution to this. The 74ACT574 input latches can be replaced with 74ACT573 transparent latches. These will allow the data inputs to pass 'transparently' through the latch when the clock input is HIGH. This will at least give the PCLA glitch capture capability for at least half of the sample clock, which is better than nothing. The 573's will latch the data on the negative edge of the clock, but this is the same time as the data is written into the RAM. If the data changes just before the latching takes place, then the RAM may not have enough setup time required to successfully write the data into the RAM.

Using the 573's was the original intention for the PCLA, but the prototype proved a bit touchy at the higher sample rates. But if you're after a glitch capture capability, then this may be a worthwhile modification.

To design the PCLA to incorporate full glitch capture capability would have required the trigger inputs to be permanently connected to the inputs. This would require a set of buffer chips, and certainly a double-sided PCB.

Frankly I don't believe the lack of a glitch capture capability is a major disadvantage in a low cost design such as this. In fact, it's sometimes beneficial to have a design that will ONLY trigger off the data that is actually captured and displayed.

Using the software

The minimum requirements for the software are a DOS based 286 or higher IBM compatible, with a VGA screen and parallel port. Any standard printer which supports the IBM extended character set can be used for hardcopy printout, as the software prints using text mode. It is recommended to use a second parallel port if the printer is to be used at the same time as the PCLA.

After installing the software, start the program by running the PCLA batch file. A main menu will appear with the various options in the middle of the screen, and the channel information box on the left hand side.

The channel information box will always remain on screen, and provides the channel number, on/off control, trigger setting, channel type, and user defined label for each of the 32 channels.

The main menu provides the following options :

TIMIMG DISPLAY: This displays a timing diagram waveform view of the most recently captured data. It allows you to zoom and expand the waveform window, and scroll the entire 32KB buffer on all 32 channels. Two cursors are available to measure the time interval and number of clock periods between two points.

DISASSEMBLY DISPLAY: This routine decodes the most recent data into separate address and data fields specified by the channel type information. This is used for decoding microprocessor buses and the like. The information is display in two columns, one containing the address, and next to it the data.

CAPTURE DATA: Starts the PCLA sampling data using the current trigger information and options specified. It then retrieves the data from the PCLA after sampling has finished. The DATA LED on the front panel will change state according to what the PCLA is doing. During PRE-TRIGGER sampling, the LED will be fully ON. During data retrieval the LED will be blinking rapidly at half brightness, and the LED will be OFF when data retrieval is complete.

EDIT DATA: Allows editing of all the trigger and channel options. The USED check box turns each channel on and off. Turning off a channel will only stop it from being displayed, but it will still be retrieved and triggered from. Any unused channels should be turned off, as this will speed up the display refreshing.

The trigger check box can be set to HIGH, LOW, or DON'T CARE. This sets the desired trigger setting for that channel. Remember to set all unused channels to DON'T CARE (X). The default state is DON'T CARE for all of the channels.

The TYPE box allows the user to define that channel as a particular data bit or address bit, for use with the disassembly function. Both address and data can be up to 31 bits wide. You do not need to specify all of the bits for the disassembly function to work; you can skip unused bits and only specify the ones you are interested in. For example, you may only want to decode address bits 2, 5, 6, 7 and the first four data bits.

An eight-character user defined label can be set for each channel in the LABEL field. Whenever using the PCLA, it's worth taking the time to label all used channels, as it can become very difficult to remember which input you connected to which point on your circuit.

TRIGGER SOURCE: This selects either internal (from the word trigger circuit) or external (from the connector) triggering.

TRIGGER POLARITY: Selects either

(Continued on page 97)

PARTS LIST

Semiconductors

IC1,2	74HCT259 8 bit adressable latch
IC3,4	74HC390 dual decade counter
IC5	74AC151 8 bit multiplexer
IC6-9	74ACT574 octal 3-state latch (or 74ACT573 — see text)
IC10-13	32Kx8 20ns cache SRAM (any brand) (skinny DIP package as used on PC motherboards).
IC14	ispLSI1016-80 (LATC1_30.JED)*
IC15	ispLSI1016-80 (LATC2_30.JED)*
IC16	ispLSI1016-80 (LACC31.JED)*
REG1	7805 (TO-220) 5V regulator
DB1	WO4 1A diode bridge
XTAL	40MHz TTL XTAL oscillator (8/14 pin DIL)
L1	5mm green LED
L2	5mm red LED

Resistors

R1	100k 0.25W 5%
RP1	1k x 7 SIP network
RP2-5	100k x 8 SIP network

Capacitors

C1	2200uF 16V RB electrolytic
C3,4	10uF 10V tantalum
C2,5-20	0.1uF 0.2 pitch monolithic ceramic

Miscellaneous

2115 type 8.5V 1A mains transformer
PCB, single sided (PCLA31)
DB25 female IDC connector
100mm length of 25-way ribbon cable
ABS case, 260 x 180 x 65mm (W x D x H), Jaycar type HB5984 or similar
Small fined TO-220 heatsink
Three 44-way PLCC sockets; eight 14-way DIL sockets; five 16-way DIL sockets; four 20-way DIL sockets; two 34-way R/A IDC levered headers; one 26-way dual row straight pin header; one 26-way IDC header plug; two 5mm LED bezels; six M3 nuts, bolts and washers; four M2.5 nuts and bolts; one IEC mains cord; set of four PCB mounting spacers; fused panel mount IEC mains connector; tinned copper wire; mains rated hookup wire; solder, etc.

PLD, Software availability:
*The programmed logic devices used in this project (IC14, 15 and 16), and also the PC software needed to control it, are available from Tronnort Technology, of 12 Copeland Road, Lethbridge Park 2770. The quoted prices (including postage within Australia) are:
Software only.....$35
Three programmed 1016 PLDs.....$55
Software and programmed PLDs.....$80
Tronnort Technology can accept phone/fax orders and/or enquiries only after business hours, on (02) 9628 1223.

Here is the artwork for the Logic Analyser's front panel (left) and the PCB (above), reproduced actual size for those who like to make their own. The large rectangular area on the front panel with crossed diagonal lines is for the input connector cutout.

A PC Based 32-Ch Logic Analyser — 2

(Continued from page 84)

somewhat surprised to find that 5BG, 6AC and 3BQ were already on the air in Australia using the same technique.

Max's remarks about his early efforts to produce an appropriate RF piezoelectric slab from a lump of crystalline quartz are quite fascinating:

"After much trial and tribulation hacking quartz crystal about with a carborundum strip on the reverse side of a hacksaw blade, I put a little diamond saw in our lathe — much to my brother's disgust. He had visions of the diamond dust getting into the bearings and ruining them."

"So, between us, we made up a separate grinding head to take the drive via a belt, to keep any dust well away from the lathe. With that I hacked my way through various pieces of quartz until I achieved a satisfactory one".

Royal Flying Doctor

As it happened, Max's venture into crystal control for the restored 3BQ had a surprising outcome. Around 1930, he was visited at Canterbury by the Rev. John Flynn and Alfred Traeger, of the Royal Flying Doctor Service. Max was lavish in his praise of Traeger, (a) for his resourcefulness in devising the pedal generator to power the RFDS's emergency transceivers, and (b) for adapting a typewriter keyboard to produce Morse code from such a transceiver, intelligible to a central operator.

Anyone who could switch on a transmitter and spell out the necessary words could send a distress message in Morse. What the RDF needed now was a means of stabilising the transmitter frequency, so that scattered transceivers would all come up on the allotted spot on the band — notwithstanding temperature extremes and/or how erratically the generator was being pedalled.

At that point in time, Max Howden undertook to do his best to provide Alf Traeger with the crystals he needed. More about that in the next issue.

(To be continued) ❖

WHEN I THINK BACK...

(Continued from page 41)

positive or negative triggering when external triggering is selected. This setting will automatically be set to positive when internal triggering is selected.

TRIGGER DELAY: A trigger delay of 0, 2, 4 or 8 cycles is selectable. This determines how long the trigger signal must be present in order for the PCLA to register the trigger event. It can be used to avoid false triggering, and also for a legitimate use when you want to distinguish between short and long trigger events.

CLOCK SELECT: Toggles through all of the internal TIMING mode clock sample rates, and also selects the STATE mode external clock.

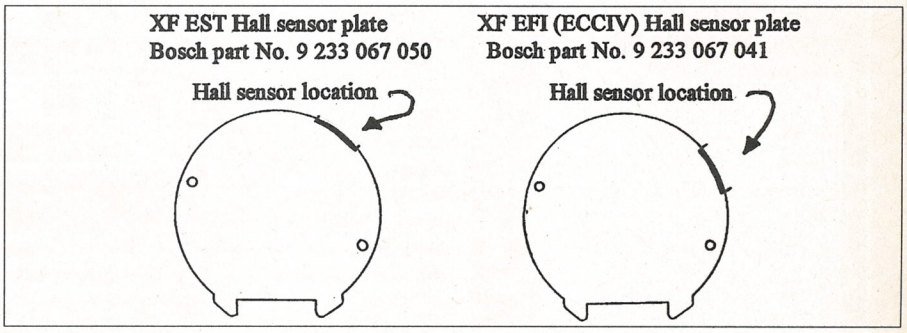
CLOCK POLARITY: Determines if the data will be sampled on the positive or negative edge of the external clock. This has no effect when in TIMING mode.

That's about all there is to the software. As you can see it's quite straightforward. Further information is available in the software documentation provided on the disk.

The software and pre-programmed PLDs will be available from Tronnort Technology. Please refer to the note in the parts list for more details.

That brings to an end the description of the new PC based logic analyser. I hope you find it a valuable addition to your test gear collection. Happy triggering! ❖

NOTES & ERRATA



Auto Electronics (October 1996): In Fig.7, on page 45, the Hall Sensor plates for XF Falcon EST and EFI (EECIV) systems were unfortunately swapped. The corrected diagram is shown above.