AN ARGUS SPECIALIST PUBLICATION

electronics today

INTERNATIONAL SEPT 1986 £1.20

·TO·BUILD·

LOW COST FRAMESTORE

BATTERY POWER – THERE'S LIFE IN THE OLD CELLS YET

CLEAN UP YOUR MAINS

JLLH ON BIAS

LOW COST FRAMESTORE

Want to grab some of the action? Need a frozen frame for the hot weather? Dan Ogilvie of Oggitronics has the wherewithal — an affordable video frame store with all the trimmings allowing storage and processing of medium resolution images.

Over the next few issues of ETI we will cover the design and construction of an affordable video framestore that can capture and process video images, whether they be from cameras, video tape recorders or broadcast in real time. A full kit of parts will be available for the construction of the PCB, or the plated through hole PCB can be purchased alone for those wishing to make their own way. A cased, tested and complete framestore is also available for those whose feet tend to be level with their head for the greater proportion of the time.

The techniques of interfacing the framestore to a microprocessor will be explained by means of an example in the form of an add-on RS232 interface which provides a comprehensive, albeit slow, access to the store. Again a kit of parts and PCB will be made available. Finally some EPROMs will be available with some simple and not so simple image processing algorithms in them to run on the MPU board.

Decision Time

At the outset, some decisions had to be made as to how much memory and what type to use, what resolution is satisfactory and affordable and, generally, what to include and what to leave out of the specification. To appreciate the decisions that were made, it's useful to have some idea what the incoming video looks like.

Video is sent as a sequence of lines of information, each one slightly below the previous one and each lasting 64 microseconds, of which 52 μs is visible. The other 12μs is time used to get the trace back to the beginning of the next line.

In all, 625 lines are sent to describe the complete image. They are sent in two goes, 312½ each time, the half ensuring the second lot lie in between, or 'interlace' with, the first lot. Each lot is called a field, 312½ lines of 64μs each, lasting a total of 20ms. The two fields together are called a frame. About 575 of the 625 lines are visible, the others being blanked during flyback, when the spot gets back to the beginning of a field.

Figure 1 illustrates what the composite video signal looks like. There are two obvious parts to it. The first is down the bottom of the signal (about 0.3V of it usually) and is the synchronizing information.

This consists of 4 μs pulses to mark the beginning of each line and longer pulses to mark the beginning of each field. Within the field pulse is information which enables us to determine which field is which within a frame.

Although a monochrome signal would not have it, I've shown where the colour burst pops up. This performs the same job as the line and field syncs for the colour decoder. On top of the sync signals and about 0.7V in amplitude is the video itself. Where the syncs stop and the video starts (on the back porch) is defined as black. As the video signal approaches 0.7V, it would appear progressively whiter.

Memories

To store one video image, we are going to break it up into little packets (pixels), which represent the brightness of the image at each point on a notional screen and hold them in a digital memory. The more pixels there are in memory, the better the reconstituted image will look.

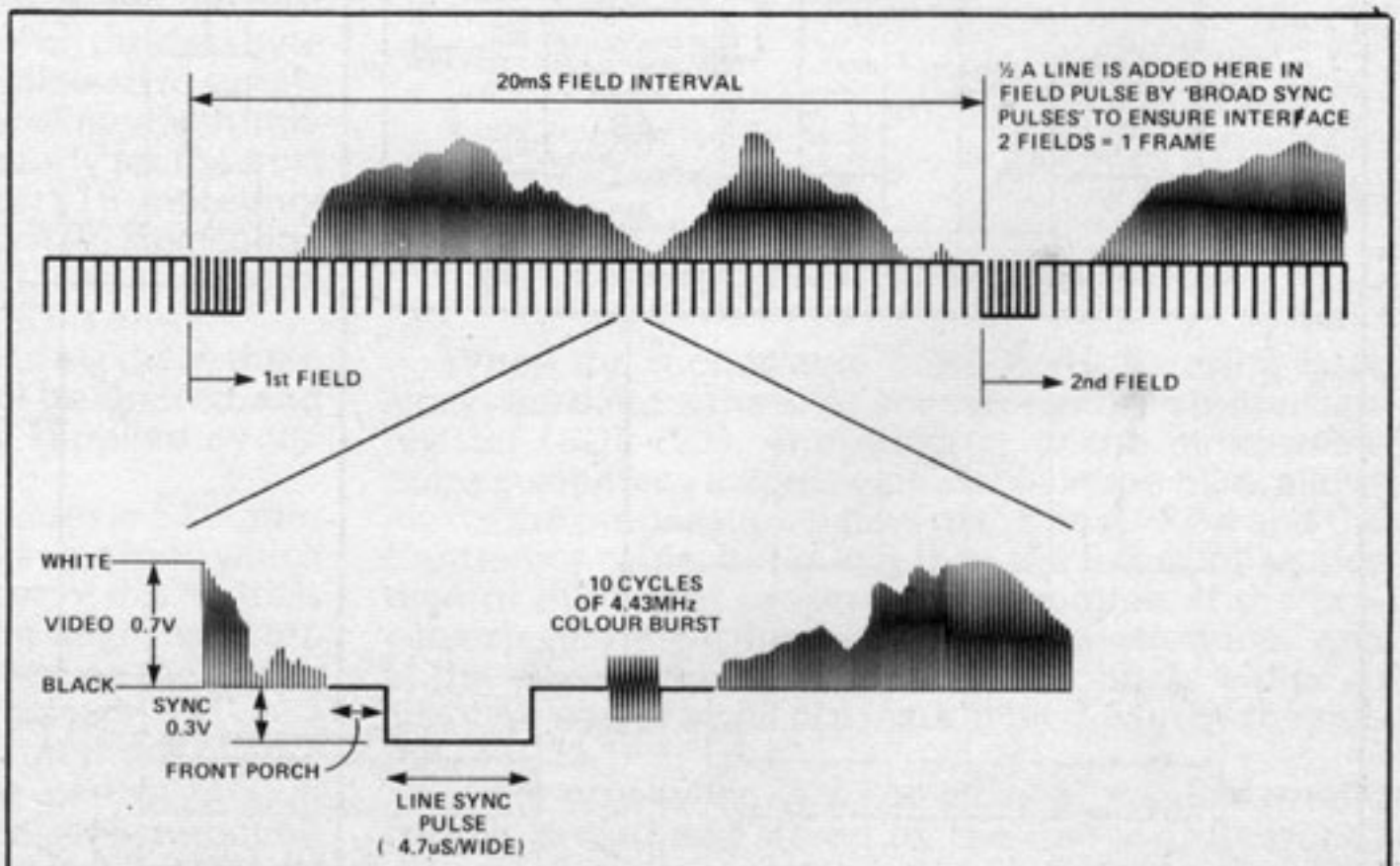


Fig. 1 The video signal.

To make full use of memory ICs, we must choose the number of pixels to be stored with care. Nearly all memories store 2^n bits of information. We could store 575 x ½, or 287½ lines per field. The nearest power of 2, gives 256 (28) lines. No real problem here, 64 μs are available to store each line and this is plenty of time for modern RAMs.

How many pixels per line should be stored? Well what about 256 again? Sounds reasonable, doesn't it? This means that one image will be composed of 256 x 256, or 64K, pixels. The pixels will require a certain number of bits to represent brightness levels, all of which means a fair amount of memory which could turn out quite expensive. Cost indicates dynamic RAM.

What should the access time of the RAM be? To make full use of the RAM, we should only store what is known as the active period of the line, the 52 μs that doesn't comprise flyback. So each pixel is 52/256 μs or 203ns wide. Let's call it 200ns and use a 5MHz sample clock — nice round numbers.

Just a small snag here though — 200ns is too fast for dynamic RAM. The popular 64Kx1, 150ns DRAM actually requires 260ns to access it, allowing time between accesses for the DRAM to settle down a little (the precharge time). One way around this is to store two successive pixels temporarily in a register and then present them to two dynamic RAMs (one for each pixel). Now 400ns are available for storage, which is plenty, and the slight increase in hardware complexity is negligible.

As far as accuracy is concerned, one factor overrides all. As I only have 200ns to convert a signal level video into a binary number, a parallel or flash converter must be used. Until recently, a 4 bit (16-level) flash converter could cost £40 or so — the 6-bit (64-level) device being over £100. To our rescue comes STC's recently introduced 8-bit combined ADC and DAC in a single package costing about £40 complete. The IC will be looked at in more detail next month — suffice to say that the price is a result of plastic packaging and anticipated high volume production for television and video equipment.

Unfortunately, to use more than four bits the amount of hardware (including memory) would have to be doubled, most TTL ICs being four or eight bits wide. To keep costs down. It was decided to use four bits with an option to expand this later.

The framestore showing an untreated image.

Although we're only using four-bits, since two RAMs are required for each storage operation we still need eight RAMs altogether. Having settled on DRAM and having 400ns available for an access, it was decided to use a read-modify-write cycle instead of a conventional read or write cycle.

By delaying the application of the write pulse, the DRAM will assume it is in a read cycle and will present the information at its Q outputs — that is, at the address requested. We take this information and display it. We also send it to an arithmetic processor which has access to the data waiting to be written into memory. The output of the arithmetic unit is then presented to the D inputs of the DRAM after any required processing. This enables the frame store to do things like add or subtract previous and present information before writing it into RAM.

Another simple benefit is that an image is still displayed even if continuously writing into the store. To do this with static RAM would require a read followed by a write cycle: 300ns in total, which is too long. A read-modify-write cycle for DRAM takes only 25ns longer than a conventional read or write cycle. Also, 8Kx8 static RAM (the most cost effective at the moment) uses common I/O pins so that the read information must be latched and held for the arithmetic logic unit (ALU) — which means more hardware.

In the actual circuit (Fig. 2), the hardware to interface to DRAM allows a simple upgrade to 256Kx1 memories (by adding one address like) which enables up to eight, fields to be stored. Also, since the contents of the DRAMs are being read all the time, the stored image is always viewable and the DRAMs can be refreshed automatically in the read-modify-write cycle as long as the address present when row address strobe ($\overline{RAS}$) goes low is the lowest order byte of the full address. This changes 128 times a line or 256 times every 128 microseconds.

The DRAMs are not accessed during the flyback period when nothing is being stored. This represents 312½-256 = 56.5 lines per field or 56.5 x 64μs, which is 3.616ms. This is the longest time the DRAMs are required to hold data without refresh. This is too long for some DRAMs which require 128 cycle refresh every 2ms. DRAMs which only need a 256 cycle refresh every 4ms present no need to worry about refresh at all. The last row to be refreshed will have to wait 3.616ms plus 128μs, or 3.744ms, which is fine.

However, 2ms DRAMs will probably work if used at room temperature or thereabouts so that

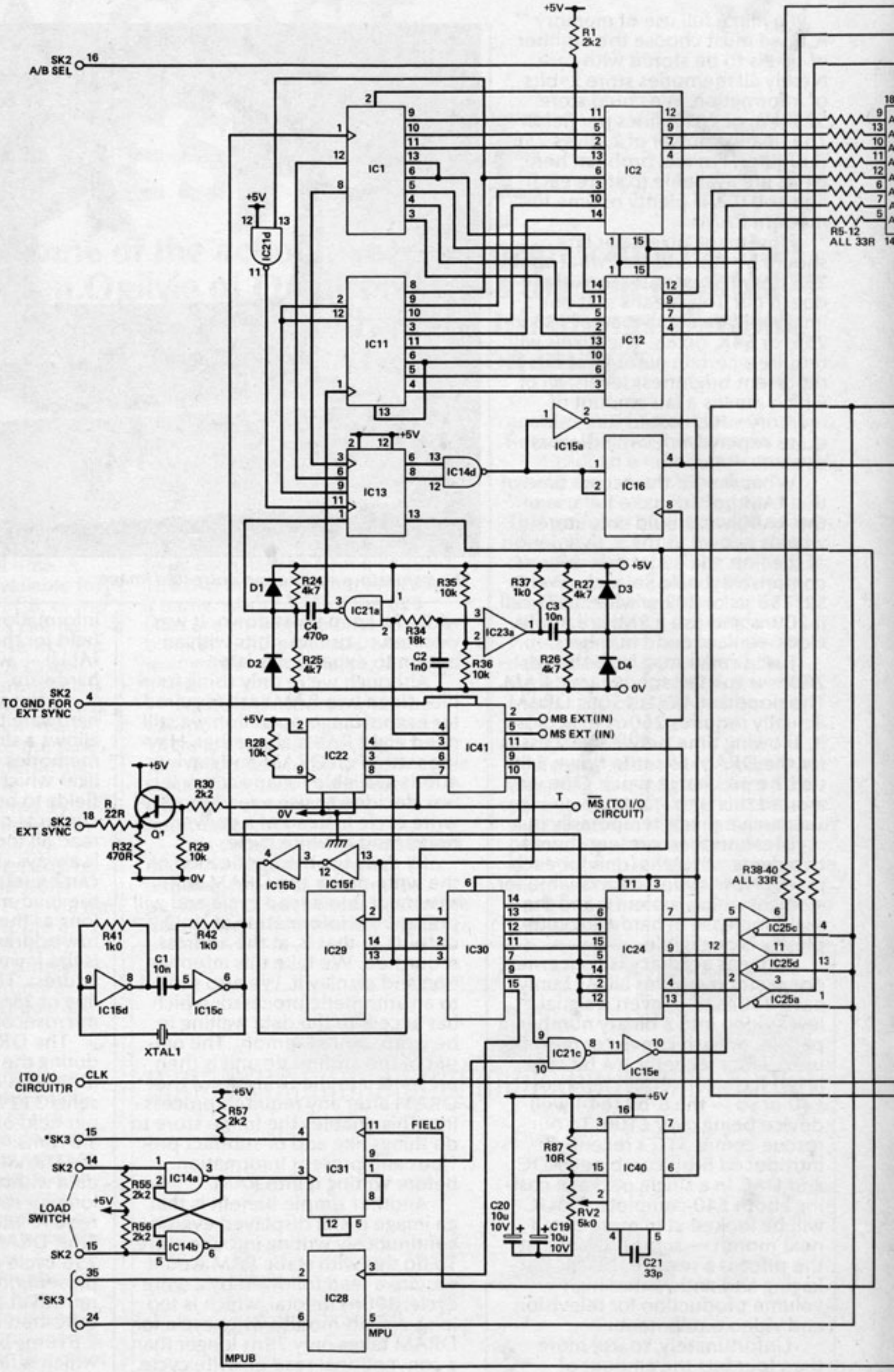
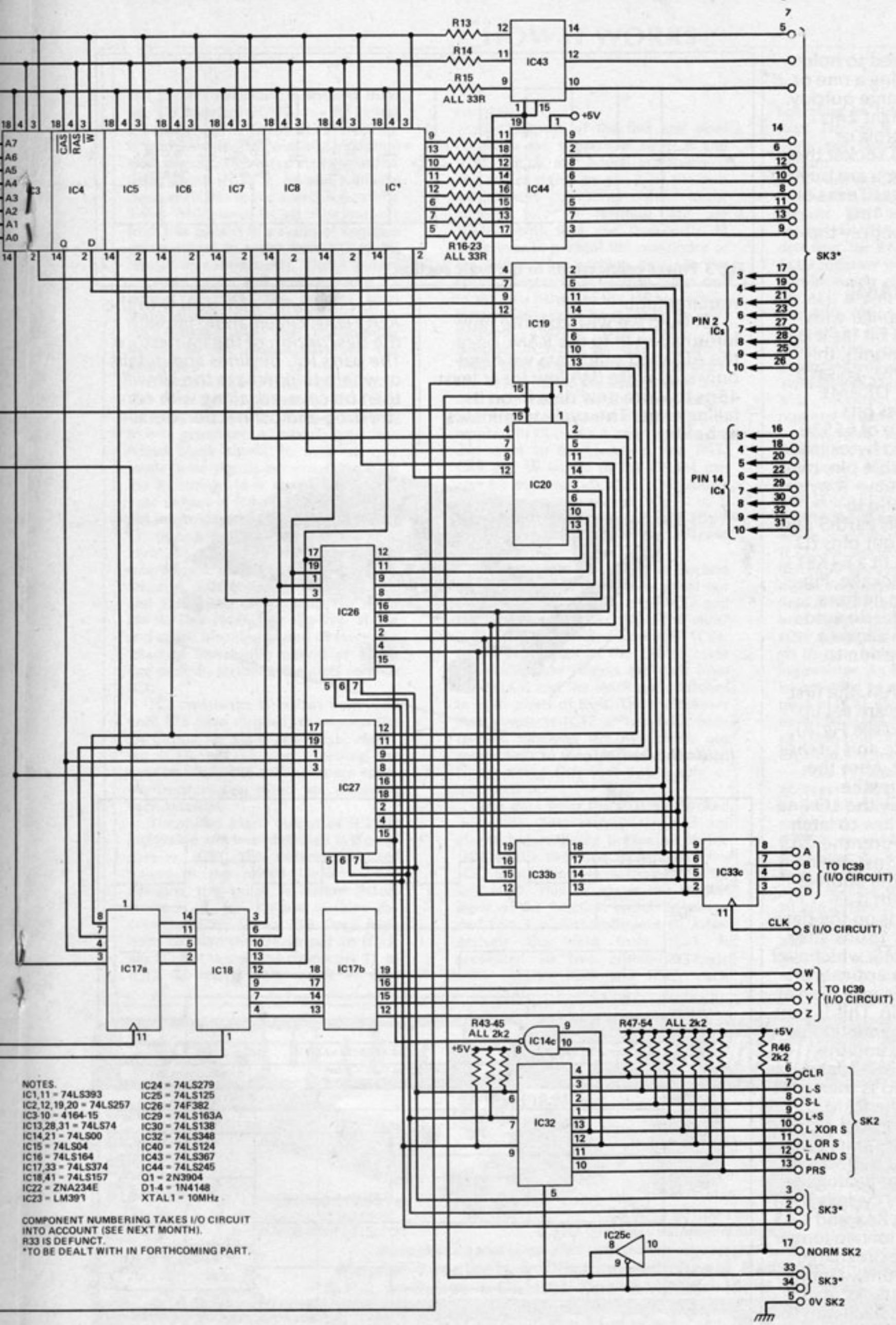


Fig. 2 The circuit diagram for the logic section of the framestore.



NOTES.
IC1,11 = 74LS393
IC2,12,19,20 = 74LS257
IC3-10 = 4164-15
IC13,28,31 = 74LS74
IC14,21 = 74LS00
IC15 = 74LS04
IC16 = 74LS164
IC17,33 = 74LS374
IC18,41 = 74LS157
IC22 = ZNA234E
IC23 = LM391
IC24 = 74LS279
IC25 = 74LS125
IC26 = 74F382
IC29 = 74LS163A
IC30 = 74LS138
IC32 = 74LS348
IC40 = 74LS124
IC43 = 74LS367
IC44 = 74LS245
Q1 = 2N3904
D1-4 = 1N4148
XTAL1 = 10MHz

COMPONENT NUMBERING TAKES I/O CIRCUIT INTO ACCOUNT (SEE NEXT MONTH).
R33 IS DEFUNCT.
*TO BE DEALT WITH IN FORTHCOMING PART.

the tiny capacitors used to hold the charge representing a one or zero bit do not discharge quickly. If you already have eight 2ms RAMs (Hitachi, Motorola or Toshiba, for instance) socket them and try them out. If you are buying RAMs from scratch, get Texas or Samsung or any other 4ms devices. The kit will supply the correct RAMs.

Just A Wee DRAM

Those who are familiar with DRAMs may skip this bit (as it is the last section this month, this has distinct attractions). To save on packaging, 64Kx1 DRAMs receive the 16 address bits required to select one of 65536 (216) locations as two bytes into the RAM. A write enable pin, two supply pins (watch out — they're the opposite way round to everything else on this earth!) and separate data in and out pins (D and Q) mean we can fit a 64Kx1 RAM into a 16 pin package. Pin one is unused (a little lie here, since it can be used for an auto refresh function) and allows a simple fourfold expansion to 256K.

To access the DRAM, the first (and lower) eight bits are set up and $\overline{RAS}$ is taken low (see Fig. 4). This initiates the cycle and latches the first address byte. After the $\overline{RAS}$ address hold time, the address is changed, by the LO line here, and $\overline{CAS}$ taken low to latch the high order byte. After the $\overline{CAS}$ address hold time (45ns), we no longer need to hold the address steady and can do with it as we will. $\overline{CAS}$ also turns on the data output drivers. Some 150ns after $\overline{RAS}$ or 100ns after $\overline{CAS}$, whichever is the later, valid data appears on Q, representing the information at the location latched in. This is sent to the ALU and some function performed to combine it and the incoming converted ADC data. The output of the ALU is then sent to the data input of the RAM.

If the RAM is to be read as in a normal display access cycle, the output data is just latched so it is ready for the digital-to-analogue converter. The access cycle is then terminated by taking $\overline{RAS}$ and $\overline{CAS}$ high. The RAM may now no longer be accessed until the precharge time has elapsed (100ns). During this time the data that was read is automatically written back into the location from wherever it was destructively read. Hence the lack of an additional refresh requirement.

Should we wish to write new information in to the RAM, once the ALU had valid data we need only take write ($\overline{W}$) low for at least 45ns to write new data in on its falling edge. The cycle terminates as before.

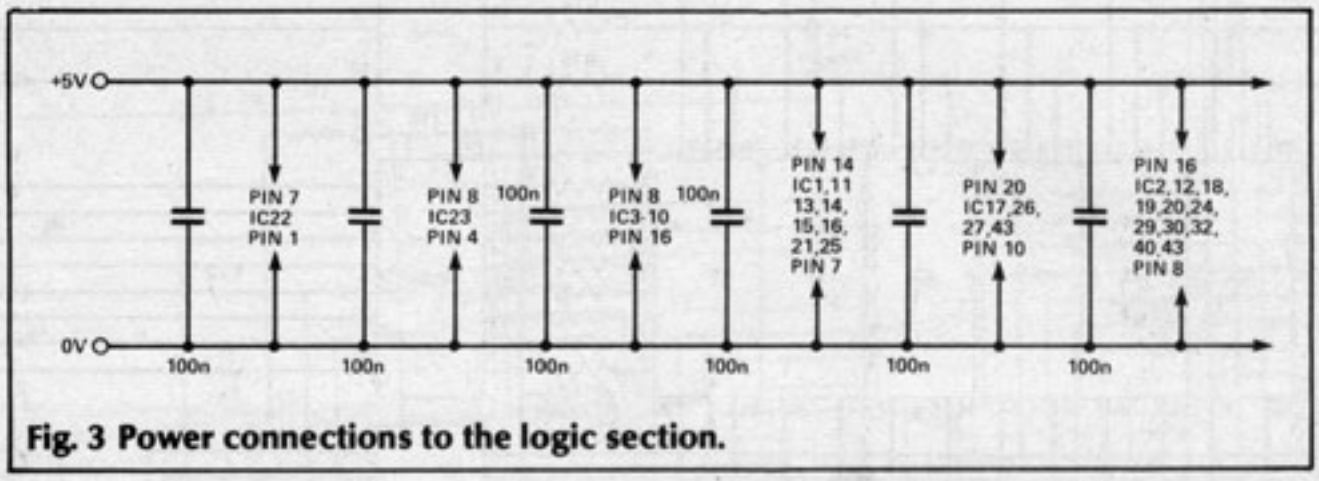
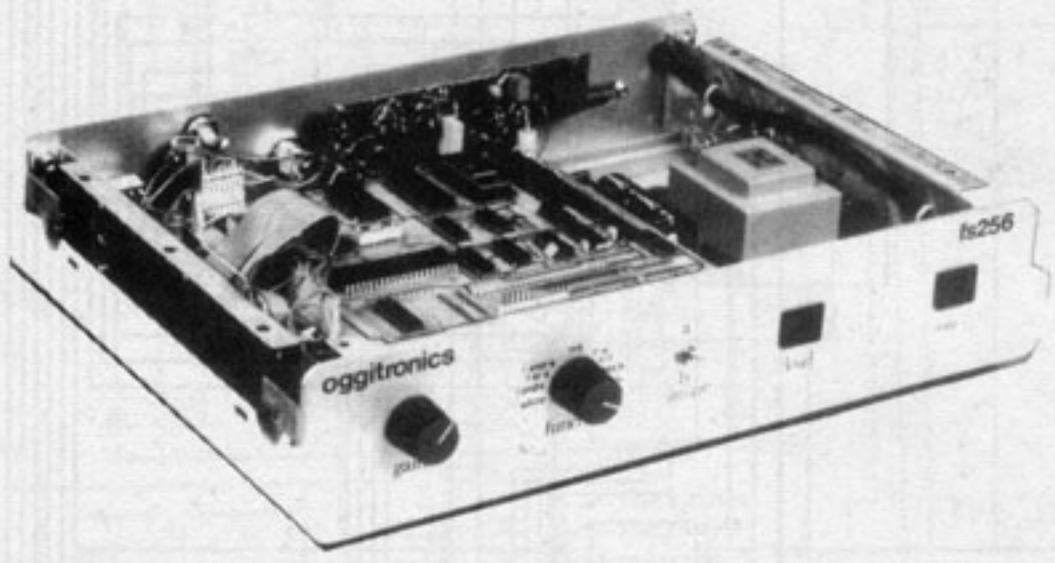


Fig. 3 Power connections to the logic section.

Next month we'll deal with the ADC/DAC circuit and complete the description of the framestore. The parts list, buylines and details of where to purchase the kit will then be covered, along with construction and connection details.



Inside the framestore.

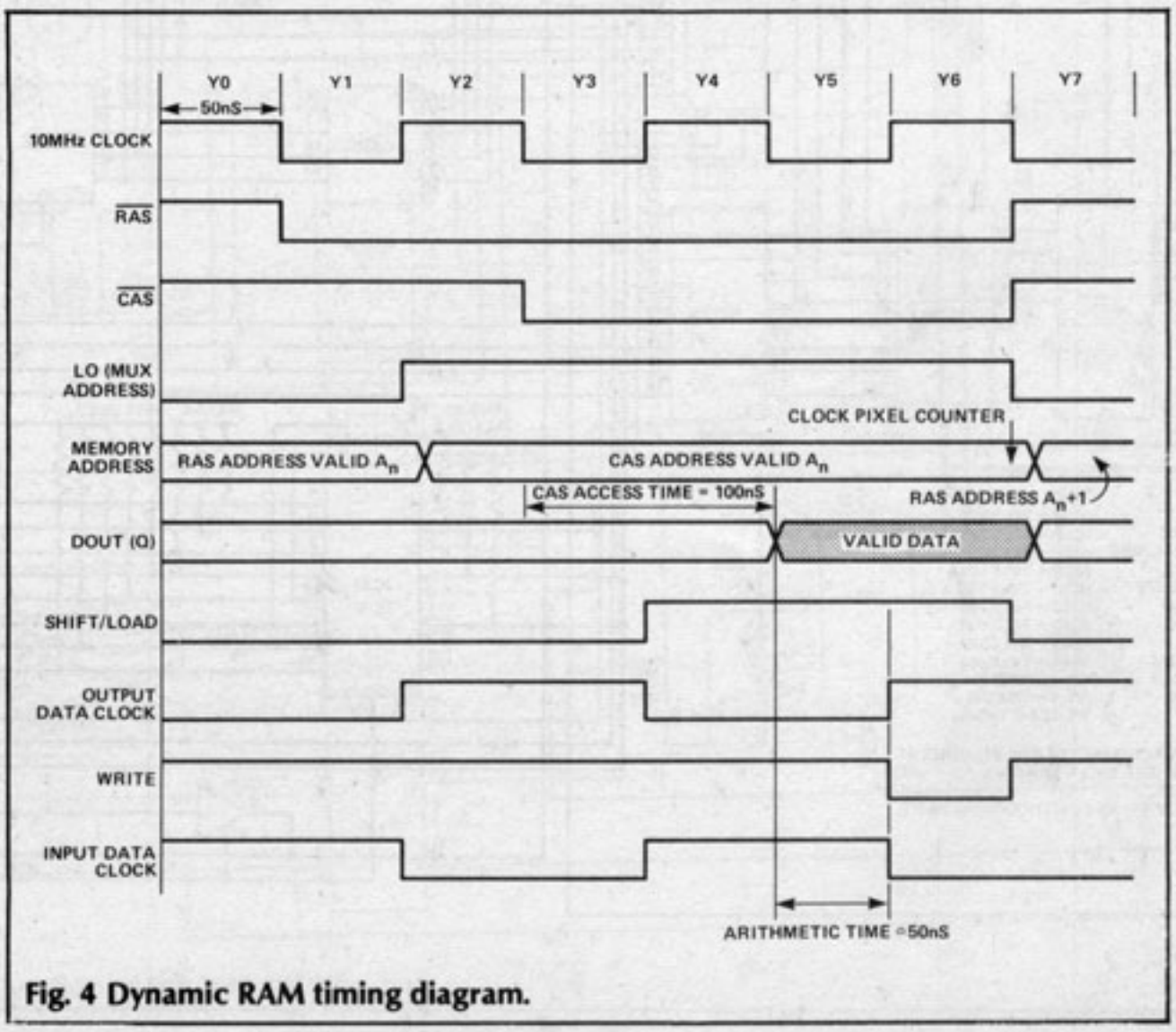


Fig. 4 Dynamic RAM timing diagram.

HOW IT WORKS

The master oscillator is formed from two TTL inverters (IC15b, c — Fig 2) and a 10MHz crystal (XTAL 1). (An option is also provided to feed in an external clock signal). This is fed via two further delay gates (IC15 a, d) and a binary counter (IC29) to the select inputs of a 3-to-8 mulitplexer (IC30). The outputs from this consist of a series of negative going pulses in order from YO to Y7 repeating continuously. These pulses are used to set and reset the four RS latches of IC24. The outputs from these form the $\overline{RAS}$ and $\overline{CAS}$ signals for the DRAMs and the multiplexing signals for the output data and the addresses.

The 2.5MHz from IC29 clocks IC22, the sync pulse generator. This chip (actually designed to produce test patterns) generates a mixed sync and mixed blank signal. In external sync mode these signals are generated from the incoming video signal. The mixed sync output is fed directly to the output mixer stage and is also buffered by Q1 to provide a 75R drive to the sync input of a camera if required. The mixed blank output is fed via an inverter and differentiating network (IC2 and associated components) to IC13, pin 1. This resets the flip-flop at the end of the blanking period of every line (start of the display period or active line period), enabling the pixel counter IC1.

IC1 counts the Y7 pulses from IC30 until 128 have elapsed whereupon the Q4 ouptut on pin 8 goes high, clocking IC13, pin 32, and resetting the counter. Only 128 addresses are counted because two pixels are stored in each location.

The mixed blank output of IC22 is integrated and level detected in a comparator, IC23. This detects the field pulses in the mixed blank output. Similarly this pulse is differentiated, clears IC13, pin 13 and enables the counter IC11. When 256 lines have been counted the Q4 output on IC11, pin 8, returns low and clocks pin 11 of IC13 via IC21d. This resets the line counter.

The outputs of the line and pixel counters are presented to IC2 and IC12, which are 2-to-1 multiplexers. The select signal on pin 1 of the two multiplexers presents the lower address bits for latching into the DRAMs with $\overline{RAS}$ and then switches them over to present the remainder of the address for latching by $\overline{CAS}$. The upper address bit is used to select one of the two halves of the DRAM, allowing two images to be stored. The multiplexed address lines drive the DRAMs, IC3-10, via 33R resistors. These help to prevent damaging negative overshoot of the address and control lines caused by the steep edges and high capacitance of the DRAM inputs. Pin 15 of ICs 2 and 12 allow the addresses to be tri-stated. The $\overline{RAS}$, $\overline{CAS}$ and $\overline{W}$ inputs of the DRAM can also be tri-stated via IC25. This allows external memory access. ICs 3 to 6 store the 4 bits of one pixel, and ICs 7 to 10 store the 4 bits of the adacent pixel.

The data from the DRAM is clocked into a 4-bit wide parallel in, serial out shift register formed by latch IC17 and the 2-to-1 multiplexer IC18. The shift/load signal is provided by pin 13, IC24. In the latter part of the DRAM cycle the multiplexer selects the data from the DRAM and the clock pulse latches in both pixels of data. The multiplexer then connects IC17 as two 4-bit wide latches, clocking the two pixels out sequentially to the DAC whilst the RAM is accessing the next two pixels of information.

The data from the ADC is clocked into IC33. Two sequential pixels are stored before being presented to the DRAM. The data may be passed from IC33 unmodified via multiplexers IC19 and IC20. This occurs if the NORM input of the function switch is grounded (pin 1 of the multiplexers). Alternatively the data from IC33 is presented to two arithmetic logic units (ALUs), IC26 and IC27, each operating on one pixel of the latched data. The other input to the ALU is from the appropriate RAM chip's output.

A read-modify-write cycle is used by the DRAMs. This allows them to present the previous stored data before rewriting occurs. The output data from the RAMs is presented to the ALUs together with the corresponding-new-incoming data. The output from the ALU is sent to the RAM via IC19 and IC20. The function of the ALU is selected by IC32. This chip provides a 3-bit binary code in response to grounding one of its eight inputs. The 'unmodified/ALU' select line and the ALU function can be tri-stated if required to allow external control.

The write signal to the RAM is derived from IC30. The load switch is debounced by IC14 and the leading edge of the pulse is latched into one half of IC31. The Q output of this is sent to the D input of the second half which is clocked by field pulses from IC23. An integral number of fields is therefore always stored, although no distinction between first and second field is made, either being stored. The load enable output gates pulses from IC30, pin 9, which are sent to the RAM via tri-state buffer, IC25b, and a damping resistor. An external load input can be provided by momentarily grounding pin 4 of IC31. Holding this pin low continuously loads new data into the RAM so that the TV monitor shows a digitized live image.

IC28 synchronizes external accesses to the RAM by, for example, an MPU, ensuring no conflict with the precharge or other critical timings. No refresh is provided for the DRAMs. By presenting the least significant address lines to the RAMs on the $\overline{RAS}$ clock edge, reading the RAMs ensures that all 256 cycles are refreshed within the 4ms requirement. Incidentally, IC22 also provides crosshatch, dot and greyscale outputs that can be used in setting up the framestore or monitor.

LOW COST FRAMESTORE

Dan Ogilvie discusses the ADC/DAC and video circuitry

In the last exciting episode I looked at the desing of a real-time storage video framestore, and in particular at the logic and memory interfacing. This month I shall describe the video input and output stages and the external synchronizing circuit.

I mentioned briefly last month that the analogue-to-digital conversion has to be performed within 200ns and that a new and cost-effective integrated circuit from STC will be used for this. It will be pertinent to look in more detail at this IC — the UVC3101 — now (see Fig. 5).

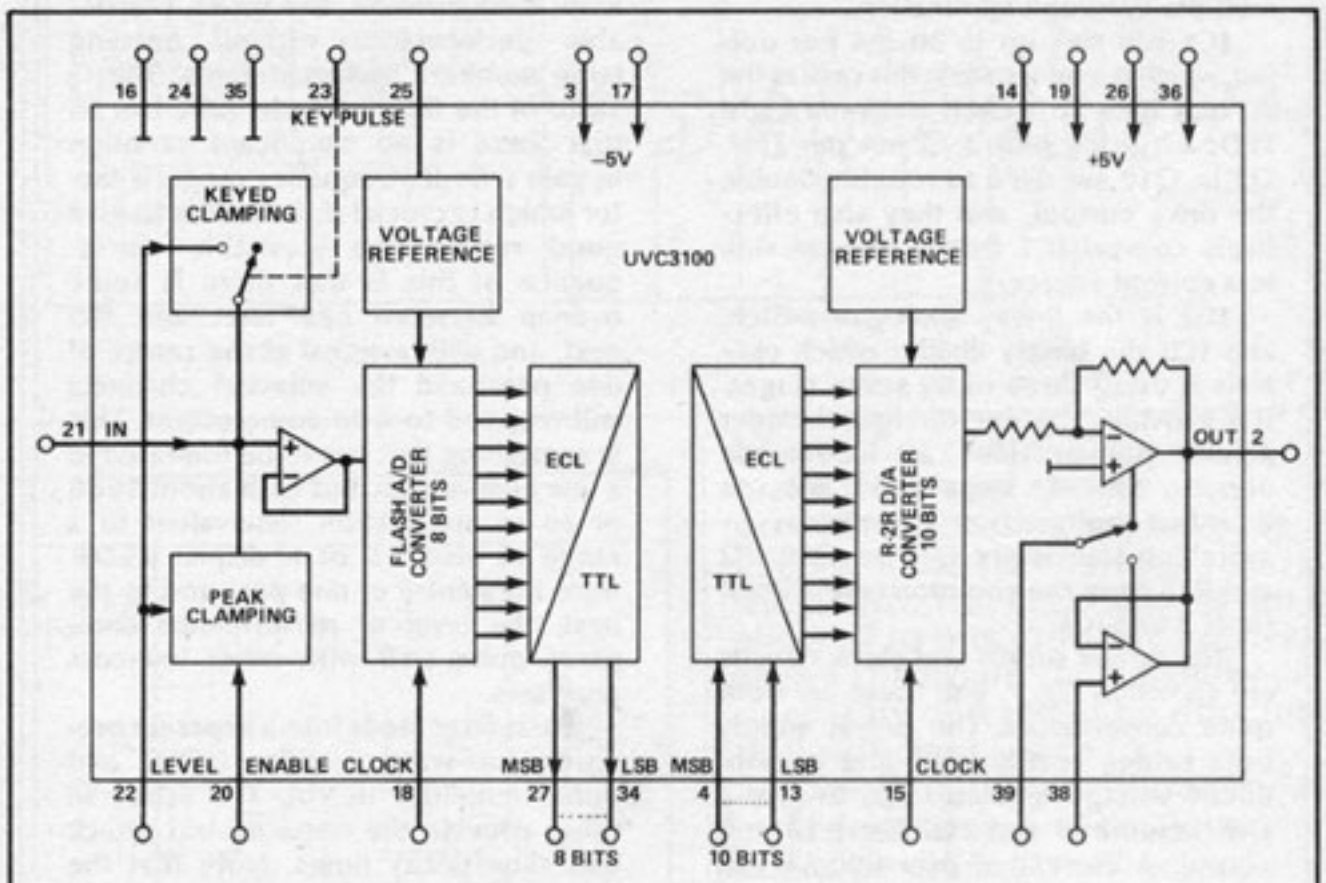


Fig. 5 Block diagram of the combined ADC-DAC.

High Speed ADC

Flash converters, or parallel converters as they are sometimes called, have been covered in ETI recently so I shall only briefly mention the principle involved.

An n-bit flash converter will contain 2^n comparators. One input to these comparators will be the analogue input, the other input will be a tap on a resistor ladder. The resistor ladder is connected in this case beteen ground and a voltage reference of 2V, provided on the chip.

The higher the analogue input the more comparators that change state so the output from the converter is a kind of bar graph. Our converter has 8-bit resolution, so 256 comparators are used. The 256 outputs are fed to a priority encoding network which converts them to 8-bit binary form.

One useful feature of the UVC3101 is the provision of an input amplifier for the ADC. The input to a flash converter usually sees a large number of comparator inputs and hence a large capacitance. Worse still, the impedance changes depending on the state of the comparators which, in turn, depends on the voltage input. We therefore normally need to provide a low impedance output driver for the ADC capable of charging and discharging the 100pF or so input capacitance at 5MHz.

The input amplifier saves us the trouble, its input impedance being 100k in parallel with 10pF. This can be comfortably driven from most op-amps.

The input amplifier also allows us to black-level clamp the input signal. We mentioned last month that the back porch of the line sync is defined as black. By generating a pulse in this part of the signal and using it to DC restore the incoming video we can ensure that the black in the image is always stable.

Jumping DAC Flash

The UVC3101 is not finished yet. Also fabricated on chip is a 10-bit R-2R DAC. An internal 2V reference for this is also provided as are two output amplifiers. One provides buffering for the DAC, the other allows us to mix in an additional video signal. Switching between amplifiers is accomplished by a TTL level signal on pin 39.

Genlock

A composite sync output is provided from the framestore board. It is meant to drive the genlock input to a video camera. This input will synchronize the camera to the framestore and ensure stable pictures are obtained when loading. Switching between internal and genlock mode may be automatic or manually achieved.

Camera genlock is the preferred mode of operation, but not all cameras have a genlock input, while video tape recorders and TV broadcasts never have the facility. The BBC are unlikely to accept a request to synchronize all their studio equipment from your framestore, so provision has been made for the framestore to lock to the incoming video. Switching is performed by pulling low pin 1 of IC41, the select input of a 2:1 multiplexer (see Fig. 2, p. 38, ETI September, 1986).

The incoming video has the sync pulses stripped from it. Mixed sync and blanking signals are derived from it and replace the

HOW IT WORKS

IC34 (Fig. 6) is a high speed op-amp which amplifies the incoming video from 0.7V to 2V. This is AC coupled into the ADC by C5. The video input is terminated in 75R by R65, but this may be removed if necessary. R63 and C6 form a low-pass filter to reduce the amplitude of high frequency signals and prevent aliasing. The AC coupled input to the ADC is clamped internally by a pulse on pin 23. IC38 generates this pulse which is arranged to lie in the back porch of the line sync pulse — black by definition. This ensures the blacks in the picture do not shift with changing light levels or with different camera iris settings. The top 4 bits of the 8-bit converter are used and are present on pins 27-30 of IC39. Pin 27 is the MSB.

The top four DAC bits are on pins 4 to 7 of IC39 (MSB first). Resistors R58-R61 pull these inputs low when their driver, IC17, is tristated. This occurs in the blanked areas of the picture. The DAC has two output amplifiers which are selected by a pulse on pin 39 — the mixed sync signal. One of these amplifiers buffers the DAC output while the other buffers an input on pin 38 of the IC which, in this case, forms a variable sync pulse level. The mixed sync pulses are used to switch between the DAC output (video) and the sync pulse level — set by RC1. The resulting composite output is on pin 2 which is buffered by Q3 to enable it to drive 75R.

The video input is also presented to one quarter of IC35 which amplifies and inverts it. The output is sent to the +ve input of a comparator IC36 and also to a peak detector formed by D5 and C9. C9 holds the peak level of the signal, in this case the voltage corresponding to the sync pulse tips. R68 and R69 discharge C9, forming the time constant of the peak detector. They also form a potential divider allowing some 90% of peak detector output to form the -ve input or reference input to IC36.

One further quarter of IC35 buffers the high impedance of the peak detector. The comparator inputs see both the sync pulses and a reference voltage derived from them at some 90% of their peak value. By using this as the comparator, the derived sync pulses are formed on pin 1 of IC36. These are inverted by Q4 to form the genlock sync input. They are also used to trigger half of IC37 to form the line sync blanking pulse, and are integrated by R74 and C10 which, with IC36, detect the field sync pulses. The line and field blanking signals are then mixed by D6 and D7 and inverted by Q2 to form the external blanking signal.

These two pulses are switched in to replace the internally generated signals when pin 1 of IC41 (a 2:1 multiplexor) is pulled low (see Fig. 2, p.38, ETI September 1986). The crystal oscillator is then also replaced by IC40 (a VCO) which is locked by the inhibit signal on pin 6 to ensure it starts in the same position in its cycle every line. RV2 sets the oscillator frequency for IC40 — which should be 10MHz. If you have no oscilloscope, set the picture width equal to that of a non-genlocked signal.

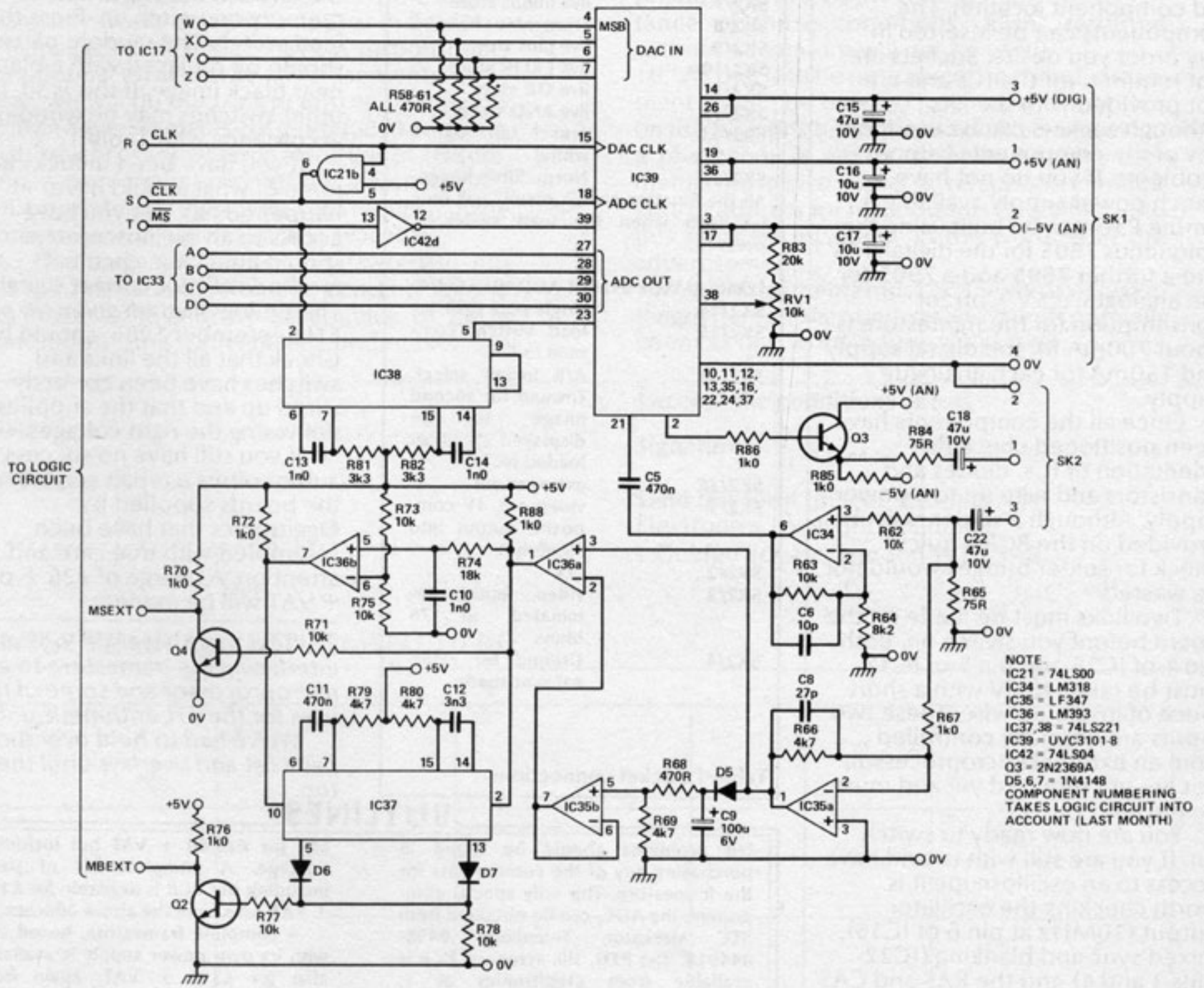


Fig. 6 The video circuit.

normal internally generated signals. The clock oscillator is also locked to the incoming video, ensuring that it starts at the same point with every line. (Incidentally, Fig. 2 fails to show the connection between pin 14, IC29, and the CLK line. This was an oversight.)

This circuit will provide a stable lock from nearly all video inputs. However, because the switching between the two modes of operation is not automatic if the input is removed in genlock mode, refresh to the memory is not guaranteed and picture corruption may occur if this condition persists.

Construction

It is recommended that the plated-through-hole PCB provided with the kit (see below) is used to construct the framestore, although a wirewrapped prototype performed satisfactorily. The board has a silk screen legend to aid component location. The components can be inserted in any order you desire. Sockets are not required for the ICs and are not provided with the kits, although sockets can be used for any of the components without problems. If you do not have a bench power supply available a simple PSU can be built using the ubiquitous 7805 for the digital +5V and a further 7805 and a 7905 for the analogue ±5V. Current consumption for the framestore is about 700mA for the digital supply and 150mA for each analogue supply.

Once all the components have been positioned check the orientation of ICs, diodes and transistors and wire up to a power supply. Although a solder resist is provided on the PCB a quick check for solder bridges would not be wasted.

Two links must be made on the board before you switch on. Both pin 1 of IC28 and pin 5 of IC32 must be taken to 0V with a short piece of insulated wire. These two inputs are normally controlled from an external microprocessor, but are not required yet and must be linked.

You are now ready to switch on. If you are still with us and have access to an oscilloscope it is worth checking the oscillator output (10MHz at pin 6 of IC15), mixed sync and blanking (IC22 pins 3 and 4) and the RAS and CAS signals. If all is well (or if you were unable to check) switch off and wire up the load switch, PL2 pin 17, to 0V and the video output to a TV monitor. Switch on again. If possible use a scope to set the sync pulse amplitude on the output to 0.3V into 75 ohms with RV1. Alternatively set it to obtain a stable lock on the monitor.

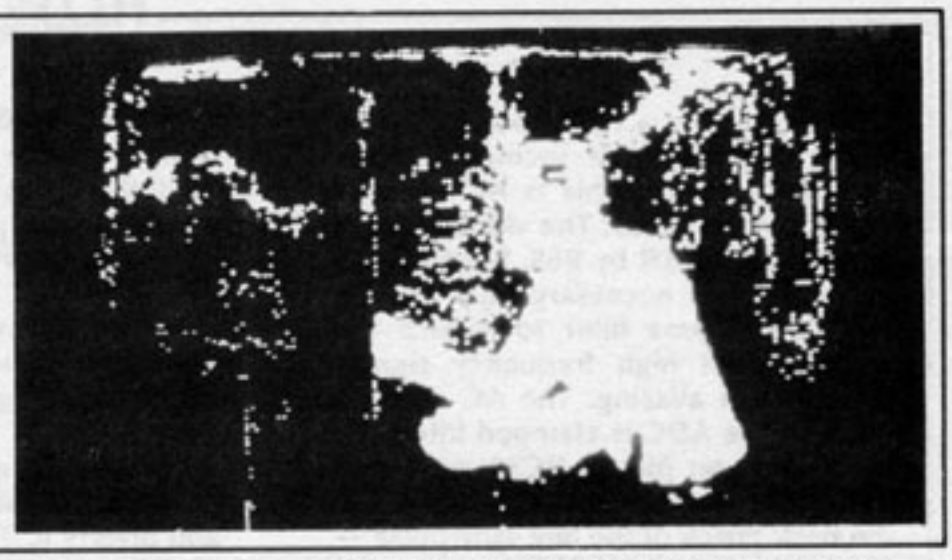
If necessary use the horizontal and vertical hold controls on the monitor to obtain a stable picture. Some form of fixed pattern should be obtained on the screen. This is the random data held in the memory on switch on. Press the load switch. The random pattern should be replaced with a black or near black image. If this is so, the other switches may be wired up in accordance with Table 1.

If you have been unlucky and none of what should have happened has, but you have access to an oscilloscope, you should check the circuit methodically for correct signals. The timing diagram given on p. 40, ETI September 1986, should help. Check that all the links and switches have been correctly wired up and that the supplies are delivering the right voltages.

If you still have no success, the author offers a repair service on the boards supplied by Oggitronics that have been assembled with due care and attention. A charge of £25 + parts + VAT will be made.

Next month we will look at interfacing the framestore to a microprocessor and some of the uses for the TTL arithmetic unit.

We've had to hold over the parts list and overlays until then, too.



The framestore will reveal differences between two images.

POWER	
SK1/1	+5V analog
SK1/2	-5V analog
SK1/3	+5V digital
SK1/4	OV
FUNCTION	
SK2/5	OV (ground one of PL2/6-PL2/13 or PL2/17 via 9 way rotary switch to select function)
SK2/6	clear function (all zeroes = black)
SK2/7	live minus store
SK2/8	store minus live
SK2/9	live plus store
SK2/10	live EXOR store
SK2/11	live OR store
SK2/12	live AND store
SK2/13	preset (all '1's = white)
SK2/17	Norm. Store image.
All the function switch results will only be seen when the load switch is pressed.	
LOAD, IMAGE SELECT AND VIDEO I/O	
SK2/14	single pole C/O
SK2/15	load switch, common to OV.
SK2/16	A/B image select. Ground for second image to be displayed or loaded to.
SK2/18	ext. sync out
SK2/1	video out, IV composite output into 75 ohms
SK2/2	OV
SK2/3	video input, terminated in 75 ohms
SK2/4	Ground for external sync mode

Table 1 Socket connections

BUYLINES

No problems should be found in purchasing any of the components for the framestore. The only special component, the ADC, can be obtained from STC Mercator, Telephone 0493-844911. The PTH, silk screened PCB is available from Oggitronics at 7, Saywell Brook, Chelmsford Essex CM1 6RJ for £25.00 + VAT but including postage. A complete kit of parts including the PCB is available for £170 + VAT, also from the above address.

A complete framestore, boxed and with its own power supply is available also for £395 + VAT, again from Oggitronics. **ETI**

LOW COST FRAMESTORE

Get those frames stored and fields frozen with the final part of Dan Ogilivies's project.

In the two previous episodes of this saga we have described the design and construction of a real time storage video field store. This month, we shall look at the operation of the store and describe how to get the best from it. You'll also find the overlay diagram and parts list.

OOPS!

Firstly, one or two errors have arisen in the previous two articles which I shall correct now:

- R36 should be changed from 10k to 3k3, as in parts list. This ensures no break through of line sync pulses into the field detection circuit.
- An error in the logic circuit diagram (Fig. 2, ETI September 1986, p.38) has appeared in the write circuitry (in fact, it is far from write!). IC15e does not actually exist at all and should simply be removed, leaving IC21 pin 8 connected directly to IC25 pin 5.
- Pins 2 and 3 of the IC23 are shown reversed on the logic circuit diagram.
- The output video transistor (Q3 on the video circuit diagram, Fig. 6, ETI October, 1986, p.49) is now removed and the video output is taken directly from IC39, pin 2, via R86 (changed to 56R), R84 (still 75R) and C18. The two resistors ensure that the 2V output of IC39 is attenuated to 0.7V into a 75 ohm load. R85 should be omitted.

These changes have been made, where applicable, to the parts list and the PCB supplied by Oggitronics (who now have a new address — see Buylines). If you're intent on making your own PCB from our foil pattern, link Q3 base and emitter and omit Q3 and R85.

In The Frame

The field store provides a 1V composite video output into 75 ohms which is compatible with most TV monitors. Should you wish to show the output on a television set, sufficient unterminated output is provided to drive the SCART connector.

The field store accepts a 1V composite video input which it terminates in 75 ohms. To lock the framestore from an external source, the sync signals must be present on the incoming signal.

A 2V composite sync output is provided which is intended to lock up video cameras that will operate in this genlock mode. This is the preferred mode of operation in that it will provide a more stable image in the majority of cases. The INT/EXT switch (pin 4 of SK2) can be left to select this mode.

If you wish to store images from sources that do not accept a genlock input, such as VTRs or television tuners then SK2/4 must be grounded to select the external sync mode. Otherwise, the framestore and its source will just free run and attempts to load an image will produce effects similar to the loss of lock on a TV when the vertical and horizontal hold controls are rotated too far.

The switching between internal and external sync is manual. If the video should fail (for example, at the end of a broadcast or a film on a VTR), no synchronizing signals will be generated in the framestore and, therefore, there will be no refresh to the dynamic RAM and this will corrupt the image. Once the image has been captured reverting to internal sync will prevent this happening.

Frame After Frame

Having connected up the framestore to a TV or monitor, and to a source of video, and having sorted out the sync, you are ready to load an image.

Switch on the framestore, ensuring that the FUCNTION switch is set to NORM and the A/B select image is set to A. A random pattern should appear on the monitor screen. If necessary, adjust the hold controls on the monitor to achieve a stable picture. If nothing happens or you get an unstable picture, check the

RESISTORS (all ¼W, 5% unless otherwise stated)

R1, 30, 43–46 55–57	2k2
R13, 14, 15, 38 39, 40	33R
R24–27, 66, 69 79, 80	4k7
R28, 29, 35, 62, 63, 71, 73 75, 77, 78	10k
R31	22R
R32, 58–61, 68	470R
R34, 74	18k
R37, 41, 42, 67, 70, 72, 76, 88	1k0
R64	8k2
R65, 85	75R
R36, 81, 82	3k3
R83	20k
R86	56R
R87	10R
R5–12, 16–23	8-way DIL 33R packs (separate)
R47–54	8-way DIL 2k2 packs (common pin)
RV1	10k
RV2	5k0

CAPACITORS

C1, 3	10n
C2, 10, 13, 14	1n0
C4	470p
C5, 11	470n
C6	10p
C7	100n
C8	27p
C9	100µ 6 V elect.
C12	3n3
C15, 18, 22	47µ 10V elect
C16, 17, 29, 20	10µ 10V elect
C21	33p

100n decoupling capacitors for every IC. All electrolytics are PCB mounting radial.

SEMICONDUCTORS

IC1, 11	74LS393
IC2, 12, 19, 20	74LS257A
IC3–10	TMS4164–15
IC13, 28, 31	74LS74A
IC14, 21	74LS00
IC15, 42	74LS04
IC16	74LS164
IC17, 33	74LS374
IC18, 41	74LS157
IC22	ZNA234E
IC23, 36	LM393
IC24	74LS279
IC25	74LS125
IC26, 27	74F382
IC29	74LS163A
IC30	74LS138
IC32	74LS348
IC34	LM318
IC35	LF347
IC37, 38	74LS221
IC39	UVC3101–8
IC40	74S124
IC43	74LS367
IC44	74LS245
Q1, Q2, Q4	2N3904
D1–D7	1N4148

MISCELLANEOUS

XTAL1 10MHz; SK1, SK2, 0.1" SIL crimp connectors; PCB FS256/4 available from Oggitronics; power supply and case to suit.

(NOTE: R85 and Q3 are missing — see text for these and other changes).

setting of the INT/EXT sync select.

Press the load button. The random pattern should have been replaced with the next complete field of video. Ground SK2/16 to view the other image. This should still contain a random pattern which can be loaded to in the same way as the first image. The two images may be switched between at field rate if required enabling storing of a 256x512 resolution image.

Don't alter the setting of the switch during loading of an image unless you want to corrupt the images for some reason. If there is little or no difference between two images it may not appear that a load has occurred at all. This is because the read-modify-write (RMW) operation on the DRAM always reads old information before writing new. The viewed image, therefore, does not appear to flash when a new image is loaded in. By holding the external load input low a continuous digitised live image can be viewed.

One And One Is ...

The RMW memory cycle allows us to modify the incoming data in some way before writing it in to memory. Setting the function switch to NORM by-passes the arithmetic unit. Selecting black or white overrides the data in the memory by setting it to all zeroes or ones. When the load button is pressed the data in memory is replaced with a black or white image.

Two subtraction functions are provided. They either subtract live from the store or store from the live image. The results are seen when the load button is pressed. The arithmetic unit — ICs 26 and 27 — operates all the time and it is possible, for example, to count the number of different pixels between two images by accessing the carry outputs of the arithmetic units. Further details of the operation of the ICs — 74F382 types — can be found in the Fairchild FAST data book.

The live and stored image can also be added together by setting the function switch to L+S. Remember that unless the images are low contrast the field store will quickly overload creating some unusual if useless results. An additional bit of memory to store the carry is necessary to prevent this happening.

The three logical operations provided work on each of the four individual bits of the image individually. That is to say there is no interaction between the results of each bit and care must be taken if you wish to interpret the results correctly.

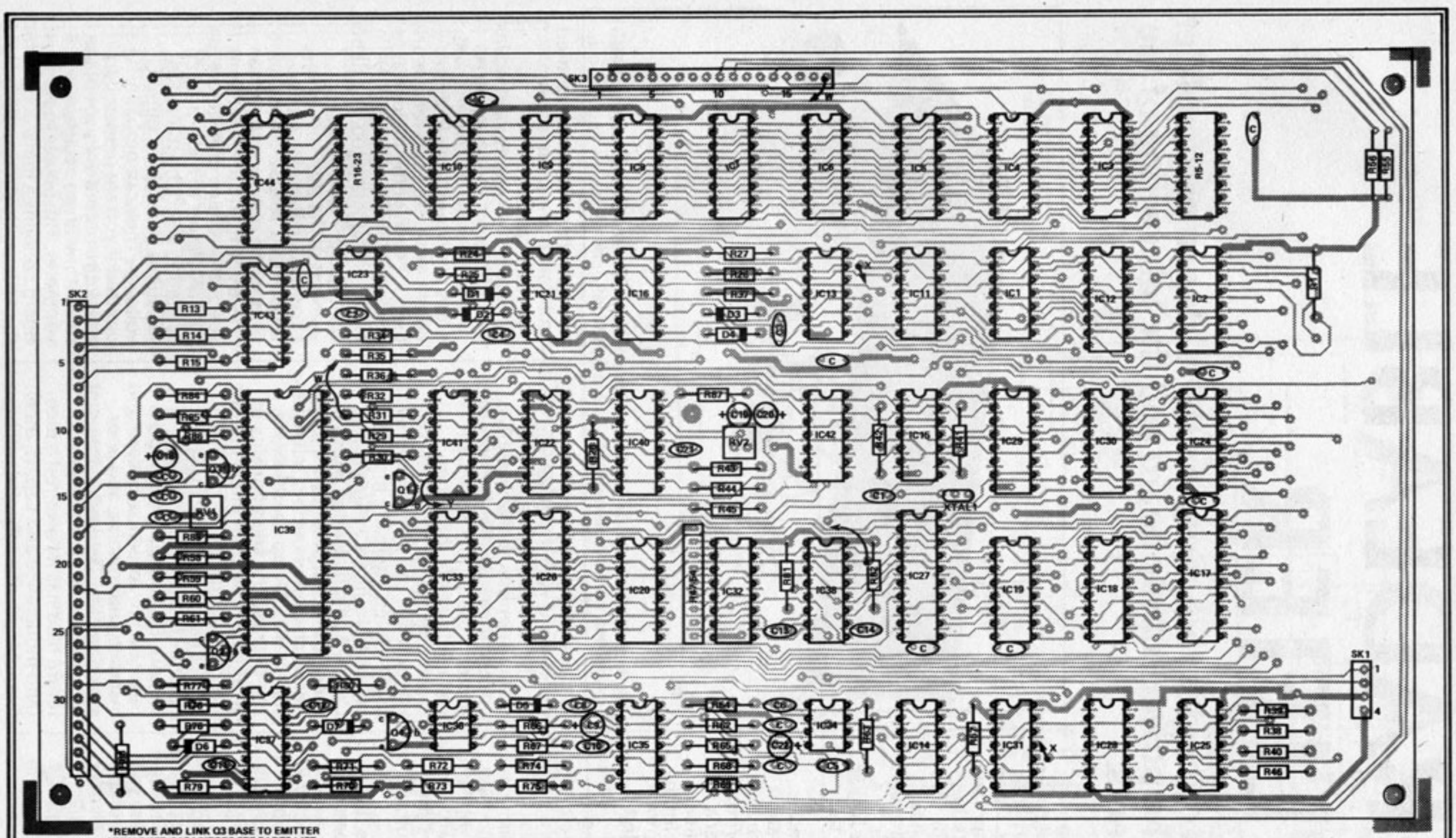
Consider the L AND S function. Only if both bits are a '1' will the answer be a '1'. If the live data is white (value, 15 or 1111H) and the stored data is a mid-grey (value, 8 or 1000H), ANDing the two will result in the lower (darker) of the two values.

The exclusive OR function reveals differences between the two images. By counting the number of pixels that are different between the live and stored image and establishing a suitable threshold the framestore makes a useful intruder detector. The count might be achieved by taking the MSB, bit 3, from the arithmetic units to a hardware counter during an XOR.

The above information should help you to understand the field store a little more and to get the best from it. In a future episode, we plan to introduce an RS232 interface to the store, enabling most home computers to access and control the framestore.

BUYLINES

A complete kit of parts for the framestore is available from Oggitronics, who have recently moved in to new premises at Poole House, 37 High Street, Maldon, Essex (telephone 0621–50378). This is by far the easiest way of building it yourself, and will cost you £170 plus VAT but inclusive of postage (as with the other prices). Oggitronics will also supply you with a PTH, silk-screened PCB for £25, or a complete unit, built and cased with its own power supply for £395. If you build your own using an Oggitronics PCB, the author offers to stop you pulling your hair out if you fail to get it up and running. He will fix any kits built on the Oggitronics board for £25 plus parts. Naturally, he can be reached at the Oggitronics address.



*REMOVE AND LINK Q3 BASE TO EMITTER
UNMARKED CAPACITORS SEE PARTS LIST

Fig. 7 Component overlay for the framestore (note: the board is double-sided through-hole plated and only one side is shown here. The complete artwork is reproduced on our foil patterns page. If you make your own board, use wire links to connect both sides of the board wherever necessary.)