POWERTRAN cybernetics ltd

cortex II

16-BIT, 16-COLOUR MICROCOMPUTER

1983

Contents

We strongly recommend that this manual be thoroughly read and digested before construction is commenced.

POWERTRAN cybernetics ltd

PORTWAY INDUSTRIAL ESTATE, ANDOVER, HANTS SP10 3NN

ANDOVER (0264) 64455

CORTEX PART 1

Announcing the ETI Cortex, in all its 16-bit splendour. This advanced design uses up-to-the-minute technology and forms the basis of a powerful home or business system.

Setting records is getting to be a habit at ETI. We were the first magazine to publish a DIY computer (the Triton). And we were the first magazine to publish a full-feature 16-bit computer, and at a price that makes a lot of commercial machines look a bit sick. The Cortex forms the basis of a versatile computer system that expands with your imagination and is based on state-of-the-art VLSI technology; the 'why use five chips if one will do?' philosophy.

Processing Power

The processor is a high-speed 16-bit device which possesses a unique system of RAM-based registers. The Cortex kit is supplied with a full 64K bytes of dynamic RAM (ie 32K words of 16 bits), and 24K of BASIC and assembler in an overlaid memory organisation (we'll explain what that means later). The high definition colour VDU has a separate 16K of RAM outside the CPU's 64K memory map and some extraordinary features that result in superb graphics capabilities. Disc drives may be interfaced easily as the controller chips fit on the PCB, and the resident BASIC can be overwritten by disc-based languages; the first that will be available is UCSD Pascal. The latter features will make the system particularly attractive to business users, and Cortexes (Corti?) will be available ready-built as well as in kit form.

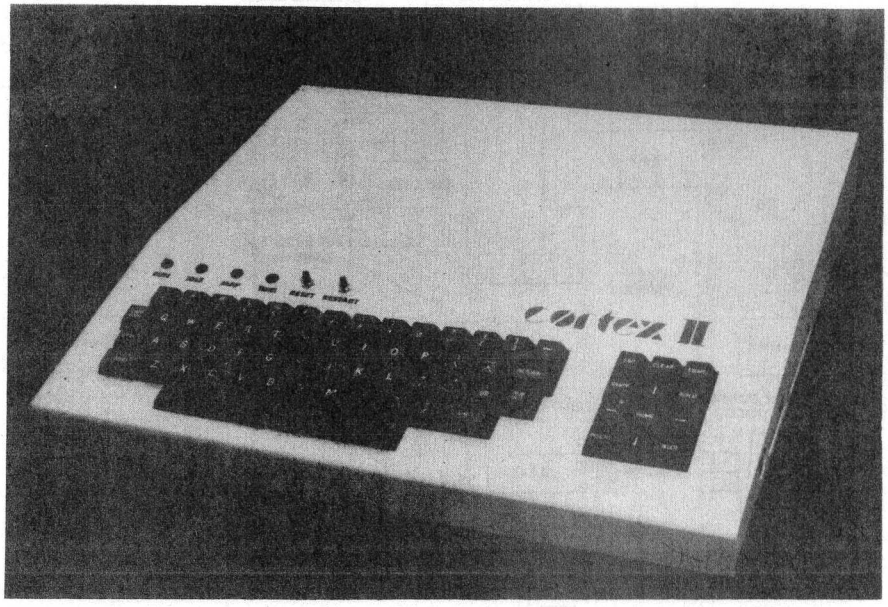
The heart of the Cortex is the TMS9995 CPU. As with all the components in this project it was selected from the wide range of currently available CPUs on a price/performance basis. The 9995 is based on the unique memory-to-memory architecture of the TMS9900. Thus it has the same architectural features of this powerful 16-bit processor and an enhancement of its rich, mini-computer style, instruction set. It is fabricated in state-of-the-art N-channel silicon gate MOS technology, enabling single 5 V operation and high speed (12 MHz) to be achieved in a compact silicon area.

As well as the 16-bit CPU, fabricated onto the same chip are a number of extra features that make this device the obvious choice for a large number of general purpose applications. 256 bytes (128 words) of on-chip RAM enables four complete, fast access register files (workspaces) to be implemented in full speed memory. A clock generator is also included to minimise the number of external components. Also available are a timer/event counter, a prioritised Interrupt Interface and a 16-bit flag register which can be output on the I/O control bus. The I/O bus is completely separate from the main memory map, and enables 32K of individual I/O bits to be manipulated individually or simultaneously in groups.

Artful Architecture

The term 'memory-to-memory' implies the fundamental difference between traditional eight-bit CPU architectures and that of the 9995. That is, all transfers in the machine are from one main memory location to another. Only three 16-bit registers exist on the CPU itself; the Program Counter, the Status Register and the Workspace Pointer.

The Workspace Pointer contains the address of the start of a block of RAM anywhere in the main memory map. This address is designated register zero; the next 15 contiguous memory locations are designated Registers 1–15. These registers may then be used by the



programmer as scratch registers. A large number of instructions exist that make maximum use of the reduced addressing needed to access these. For example MPY R4, R5, performs a 16-bit multiply of the contents of Register 4 with Register 5 and stores the 32-bit result in Register 5 (most significant word) and Register 6.

What advantage does this give a programmer over other architectures? Well, in addition to providing no restriction on the choice of addressing modes (register 0 can still be accessed as memory location 7XXXX) the power comes when an interrupt or other subroutine call occurs. It is obviously desirable, especially on receipt of an interrupt, to be able to react as rapidly as possible. On a register-oriented machine it is necessary to save the contents of the working registers in main memory, a slow operation requiring a large number of reads and writes to push the registers onto the stack. On the 9995 the context switch occurs by changing the value in the Workspace Pointer. This points to a new area of fresh registers ready to process the interrupt. The full context of the previous operation is retained in the previous workspace registers, which, being in main memory, are still preserved intact. A return is similarly implemented easily and rapidly by restoring the old Workspace Pointer value. In many real time control situations with a large number of external events occurring this architecture is the only one that allows for efficient processing.

External Affairs

The system clock is externally generated and fed to the CPU via the XTAL2 input. A clockout signal is also provided that is one-fourth of the crystal frequency (3 MHz). The 64K bytes of system memory are directly addressed by A0-A15. Here the convention is that A0 is the most significant bit. A0-A14 are also used to directly address the separate I/O structure of the Communication Register Unit (CRU). The bit to be accessed is held on A0-A14 and the data is present on A15. The data is then clocked out by CRUCLK. Reading a CRU bit into the CPU is achieved in the same manner but is read into the CRUIN line.

The data bus is multiplexed from the internal 16-bit architecture to eight bits externally, D0-D7. A memory access is signalled by MEMEN and data direction is controlled by DBIN. WE indicates a memory write. For the control of slow memories a READY line signals to the CPU if the memory is ready to complete the current access. If not, the CPU waits for another CLKOUT cycle before continuing.

In the Cortex all these features are used to provide the fastest BASIC available to a home constructor. Even then the speed is still limited by I/O transfers. This can be seen as evidenced by the amount of time the 'idle' LED is on. This LED is directly driven by a status decoder that indicates when the CPU is no longer executing any instructions but is waiting for an external interrupt.

Dynamic Designing

We don't doubt that all you digital dabblers have watched the plummeting price of memory in our advertisers' pages, and the most dramatic trend has been in dynamic memory (DRAM). This is partly due to the simplicity of the basic cell design in a DRAM; and this simplicity means in turn that the most dramatic cell density increases have been and will be made in DRAMs. With 16K DRAMs prices have pretty much levelled out at the bottom of the curve, but the 64K DRAMs are in hot pursuit.

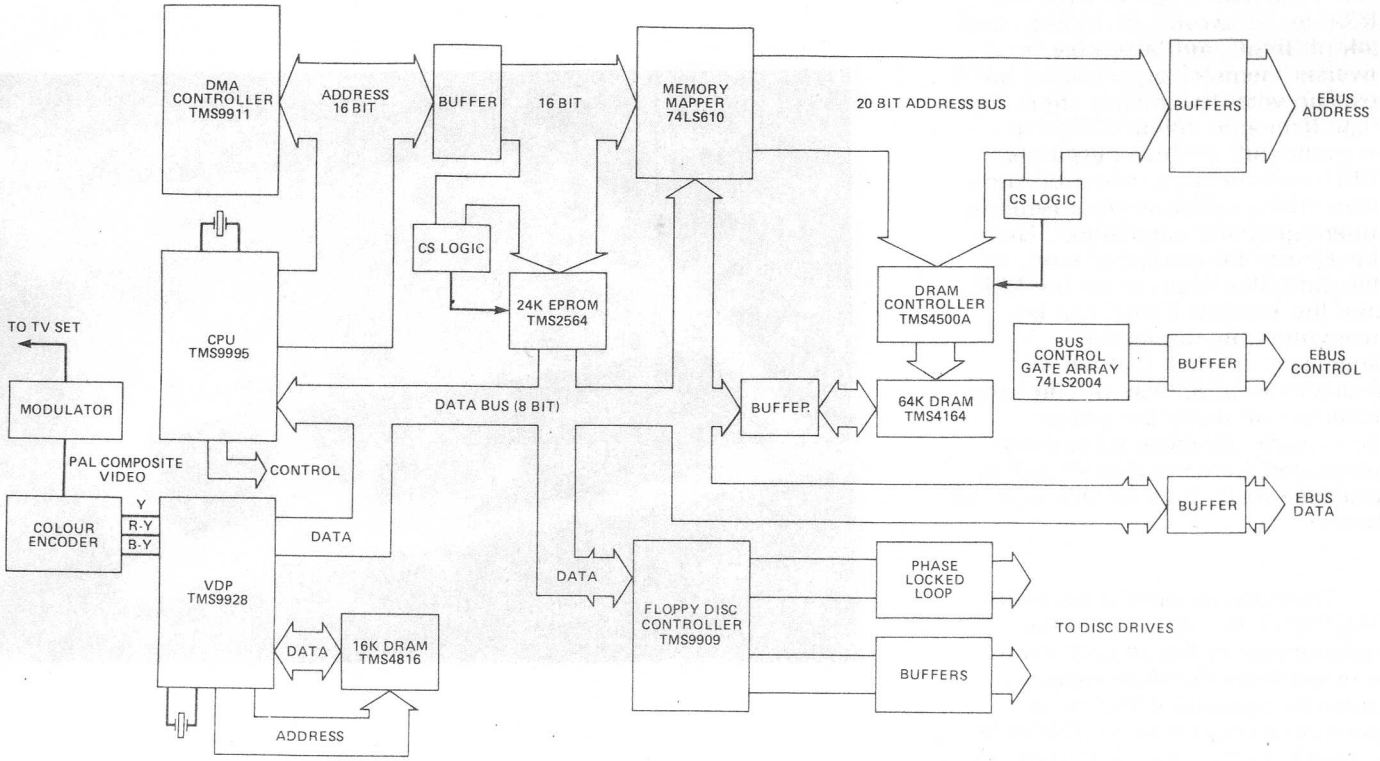


Fig. 1 Block diagram for the complete Cortex.

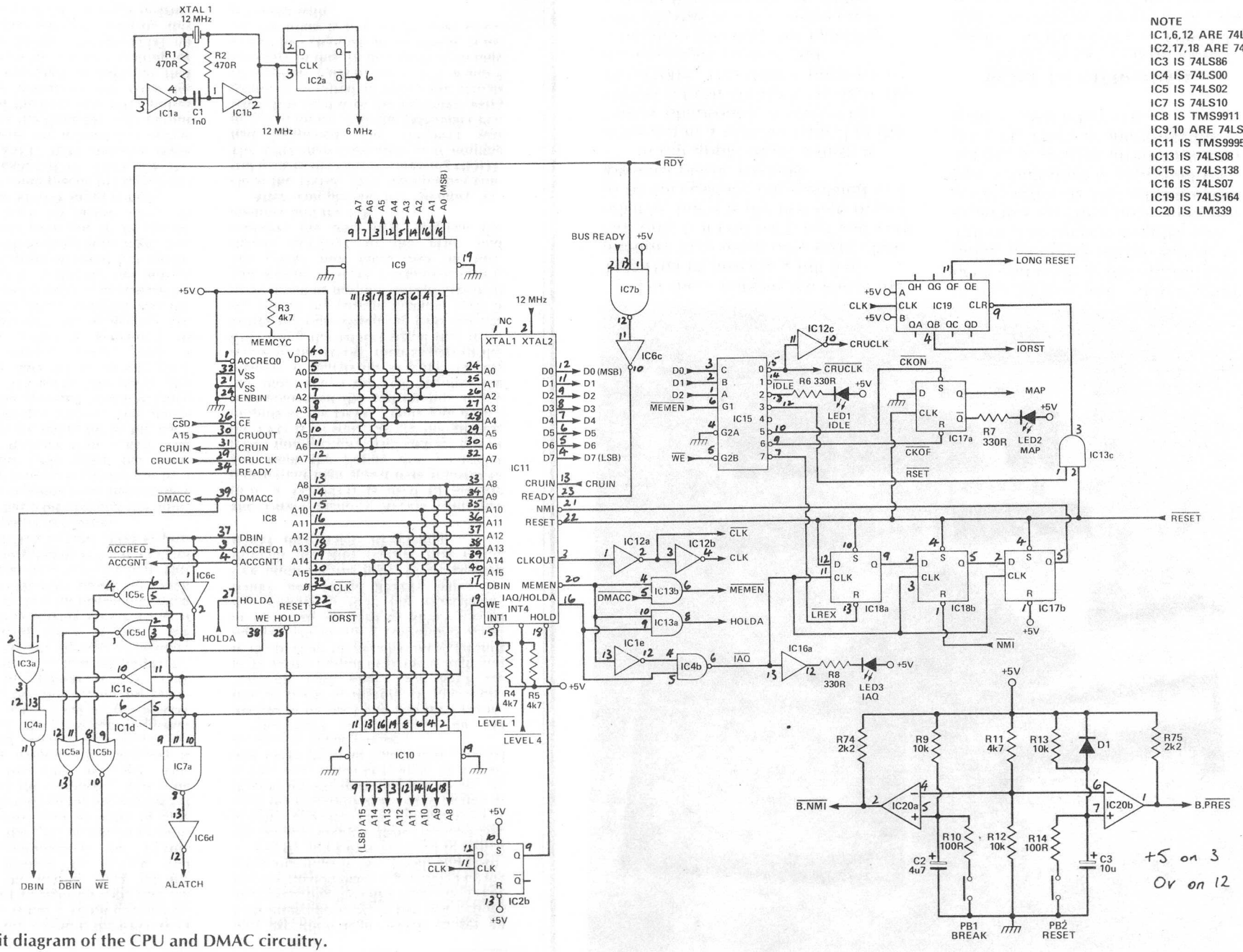


Fig. 2 Circuit diagram of the CPU and DMAC circuitry.

Cortex II - Page 3

HOW IT WORKS — CPU AND DMAC

The heart of the system is the CPU, IC11 (a TMS9995). It has a 16-bit internal architecture and an eight-bit wide external data path. The master clock for the system is formed by IC1a,b and associated components; the 12 Mhz clock rate of the CPU enables it to complete a memory read or write in only 166 nS! This is too fast for present DRAM technology so the automatic wait state feature of the CPU is used. This automatically assumes that memory is not ready and extends the memory access to 500 nS. The cycle can be further extended by a low level on the READY input to the CPU; this occurs, for example, when the DRAM is not ready because a refresh cycle is taking place.

The CPU signals the type of memory cycle by driving either DBIN or WE (write) low after driving MEMEN low. If the memory cycle is an instruction fetch then the IAQ/HOLDA signal goes high until both bytes have been fetched. This condition is decoded by IC6e, IC14a and buffered by IC16a to light LED3 to provide a front panel indication.

The CPU has a bit-mapped I/O interface which is separate from the memory data bus; the process is carried out by a section of the CPU called the Communications Register Unit (CRU). The data transfers are serial, bit by bit, each bit having a unique address. This allows 32K bits to be accessed (not 64K since address line A15 carries the data). The value of the data bit is on CRUOUT (A15) for output cycles and a WE/CRUCLK pulse is generated to strobe the data into the I/O devices. On input cycles the data is sampled from the CRUIN line and a pulse is generated on the DBIN line to enable the bus buffers and so on. During all serial I/O operations the MEMEN signal stays high. Any number of bits from one to 16 can be transferred, each bit taking 500 nS to transfer if the READY input is high.

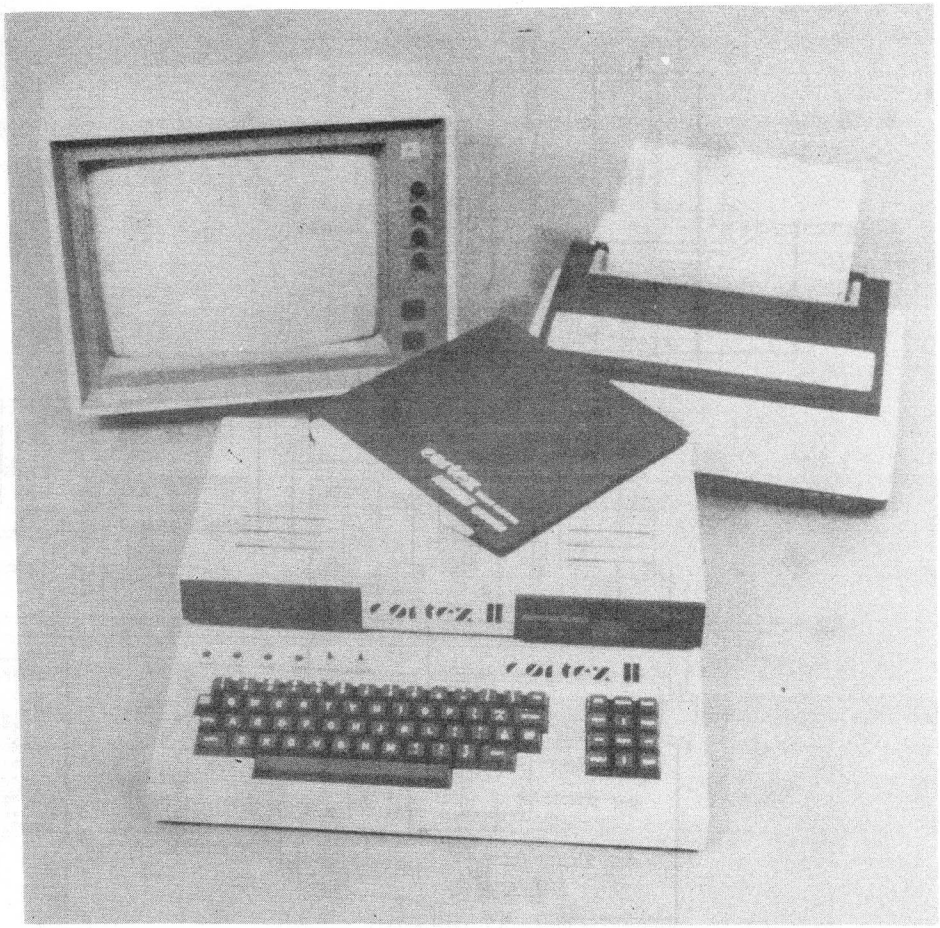
There are some special I/O signals for control use called IDLE, LREX, CKON, CKOF and RSET. IC15 decodes these separately from the normal I/O operations by using the three-bit code output on D0–D2 of the data bus. IDLE pulses continuously whenever the IDLE instruction is executed; it indicates that the CPU is in an internal loop waiting for interrupts (ie doing nothing). LED1 on the front panel lights to indicate this state. LREX is used for the single instruction execution logic which causes an NMI interrupt to occur two instructions after executing the LREX instruction. The two-instruction delay is generated by the series flip-flops IC18a, IC18b and IC2b.

CKON and CKOF are used to switch the memory mapper device from passive to active and vice versa: the signals set or reset the Q output of IC17a to enable or disable the memory mapper (IC26) via IC24a, 25a. When Q is high, $\overline{Q}$ is low, and LED2 lights to signal that external memory is being accessed.

The RSET signal causes all I/O devices to be reset and sets the CPU interrupt mask to disable all interrupts. Normally both RESET and RSET are high, so the output of IC13c is high and IC19, an eight-bit parallel-out serial shift register, clocks out a continuous series of 1s. A low on RSET or RESET sets all the outputs of IC19 low (specifically IORST and LONG RESET); when the CLR input returns high, 1s are clocked through the shift register first taking IORST then LONG RESET high.

The Direct Memory Access controller (IC8, a TMS9911) is used to provide transparent high speed data transfer to and from the floppy disc controller (FDC) into memory. The address bus of the CPU is tri-state, as are the address outputs of the DMAC. Only one device is in control of the address bus at any one time. When the FDC requires the memory it signals on ACCREQ (access request); the DMAC then signals to the CPU using the HOLD signal that it requires the bus. When the CPU reaches the end of the current memory cycle it tri-states all its outputs (except MEMEN) and signals HOLDA ('acknowledged'). The DMAC now takes over the bus, signals ACCGNT to the FDC, and transfers the data byte between the memory and the FDC.

After completing the memory cycle(s) the DMAC then relinquishes control back to the CPU by releasing HOLD. The CPU then continues as if nothing had happened. The TMS9911 was designed for use with the TMS9900 CPU; when it is used with the TMS9995, extra gating is required to make the signals compatible. Parts of ICs 3, 4, 5, 6 and 7 take care of this. In this application only one of two channels in the DMAC is used; the other is free for the user to experiment with.



For these reasons we chose the TMS4164 to provide a full 64K memory map using only eight chips. Not only is it compact, fast and very reliable, but it is the first 64K device to be successfully encapsulated in a low-cost plastic package.

In our application, refresh is achieved in a manner typical of the Cortex philosophy; a single-chip DRAM refresh controller is used, the TMS4500A. This device provides all the necessary control and arbitration functions for handling 64K DRAMs in a microprocessor system. It accepts the 16 address lines, A0–15, and multiplexes them to row address and column address (RAS and CAS) at the appropriate times, generates refresh signals for 128 or 256-cycle memories and arbitrates synchronously between access requests and refresh cycles. Synchronisation is important for achieving reliable arbitration, and the CPU clock is utilised as the main timing reference.

ROM To Manoeuvre

Similar design criteria were applied to the choice of EPROMs to store the system firmware. In order to economically store the large number of bytes required to provide

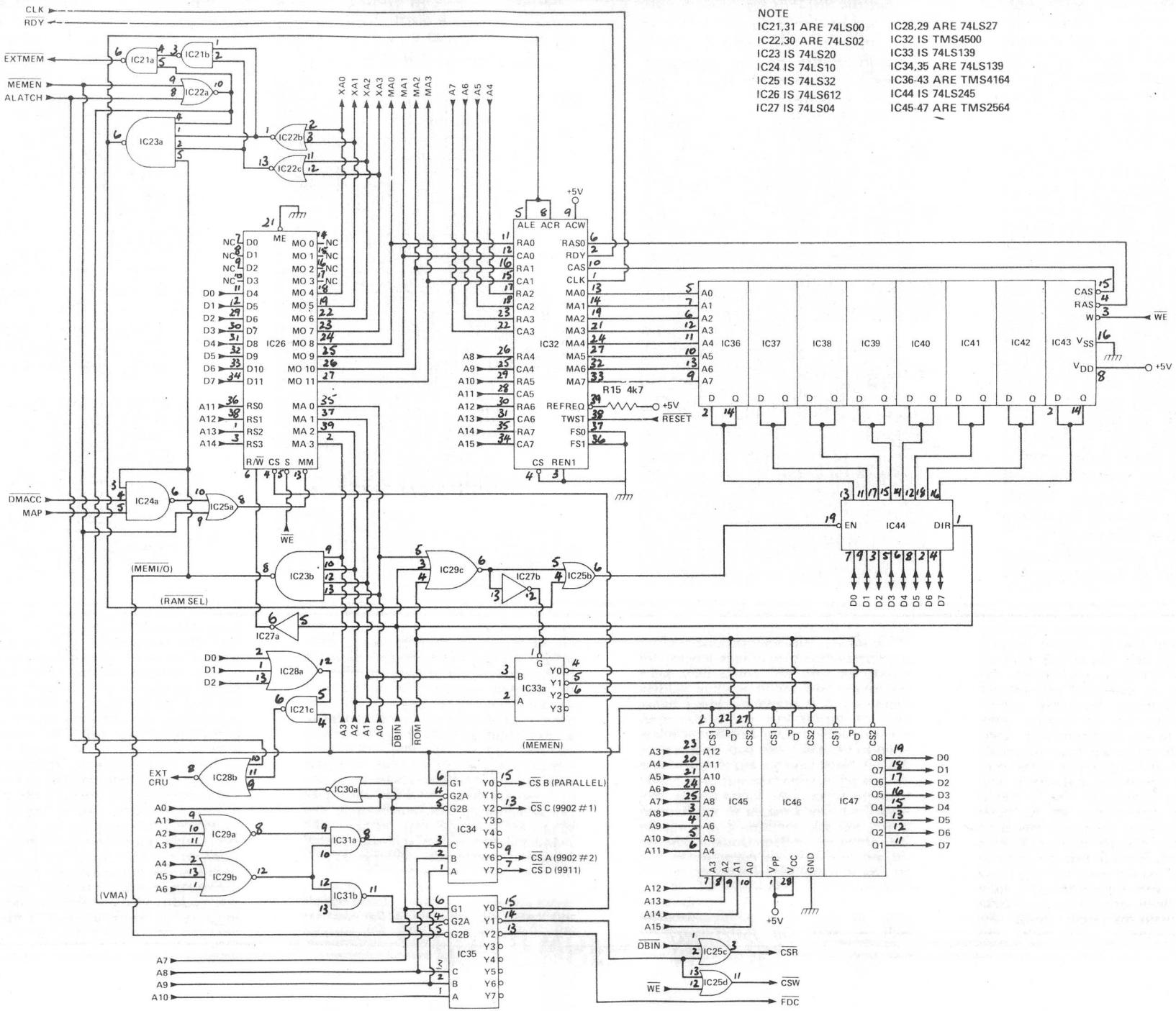


Fig. 3 Circuit diagram of the memory section

HOW IT WORKS — MEMORY

The 24K of EPROM (IC45, 46, 47) contains the assembly language support and the BASIC interpreter. The EPROMs are switched in and out of the memory map by the I/O bit 'ROM' (see I/O section). This signal powers up in the active (low) state, with the EPROMs on. The DRAM (ICs 36–43) is also accessed during a read of the EPROMs but the data buffer IC44 is not enabled; this means that any write while the EPROMs are on is put into the DRAM, so that it behaves as a 'phantom' or ghost. The BASIC interpreter copies itself into the DRAM and then switches the EPROMs off. This has two advantages; first, during execution the interpreter overlays sections of code and then re-copies the relevant section of EPROM back to conserve memory. Second, to enable disc-based operating systems (eg Pascal) to be used, the system needs to be able to operate RAM-resident.

The addresses for DRAM accesses are passed through the memory mapper device, IC26. This segments the CPU's 64K address map into 16 pages of 4K; each page has a 12 bit register (MO0 to MO11) of which only eight bits are used (MO4 to MO11). The outputs replace the top four bits from the CPU and add four more. Thus each 4K block can be anywhere in the 1M total address reach (20 bits). The CPU at any one time still only has a 64K address map but by dynamically loading the mapper during program execution the full 1M can be used. The mapper registers are loaded or read as 16 memory locations in the memory-mapped I/O area in high memory.

The chip select logic defines the first 64K as on-board memory and the remaining memory map as off-board, using the E-BUS interface (see later). The bottom 32K of memory has the 'phantom' DRAM under the EPROM although only 24K of this is used. The DRAM occupies 60K of the memory map; the top 4K is sub-divided into 256 bytes of high-speed internal to the CPU, and then a memory-mapped I/O region for the Video Display Processor, the memory mapper and the floppy disc controller. Eight memory-mapped slots are decoded, leaving some spare chip selects for user experimentation and expansion.

The DRAM controller (IC32) takes the 16-bit least significant address signals and multiplexes them on outputs MA0–7 to supply the memory devices with the correctly-timed waveforms. Once an access request is made, the addresses are first latched and then the controller arbitrates between a refresh cycle and an access cycle. If the controller is busy refreshing the memory then it signals a 'not ready' state to the CPU on the READY line, which suspends operation until the signal returns high again. The controller has to use 'cycle-steal' refresh, as the CPU is nearly always accessing memory and not enough free time can be guaranteed. The refresh cycle obviously slows down the CPU, but by less than 10%, which is a small penalty to pay for the large amount of memory at a low cost.

a comprehensive and versatile BASIC language, high capacity devices were required. The cost and capacity trends in EPROM technology have followed a similar path to DRAMs; thus the TMS2564 was chosen to complement the TMS4164 in the system. These devices, like all the major devices in the project, operate off a single 5V rail.

Each 2564 can store 8K of program organised in the now industry-standard 8K×8 format. Three chips are used to store the firmware: 24K in all. However, an important design feature of the Cortex should be mentioned here; the EPROMs phantom the DRAMs in the memory map. At power-up the EPROMs are enabled, and after checking the DRAMs the program then copies the full operating system from EPROM into RAM. Once this has been completed the EPROMs are disabled. Thus the operating system is running in RAM, allowing changes to be made or sections deleted to create extra space. This is most noticeable when you're only operating the assembler section, since the whole of the BASIC interpreter may then be eliminated from memory, freeing an extra 20K for the user! Thus the Cortex is the only computer offering a full 64K of RAM to the user as a standard feature!

Graphic Descriptions.

No home computer is considered worthy of the name these days unless it features impressive graphics capabilities, and the Cortex certainly won't disappoint you in this respect (just look at our cover and the photos in this article for some examples of the display power). What's more, all this is controlled with just one IC, the TMS9929! This chip is a memory-mapped video controller, which produces all the relevant video signals internally; it's a very versatile device capable of producing extremely complex displays, including 3D simulations. In order to achieve this its mode of operation is very different to the majority of display devices currently in use.

For simple text displays the video output of a computer may be produced by using a small random access memory to address a larger read-only memory that stores the character shape. Teletext displays fall into this category. The next step up from this is to have a display where each displayed dot is a bit (or bit pattern) stored in RAM. The first method means that shapes can be positioned very rapidly but the repertoire of shapes is limited to those in the graphics/text ROM. The second scheme is slower but allows more complex shapes to be created and lines to be plotted. However, problems occur when trying to overlay shapes as everything must be done in software — making things even slower.

The 9929 uses variants on both these schemes to produce extremely complex shapes with the minimum of software overhead. Tables are designated in an area of RAM to define pattern shapes, characters or graphics. A separate table area is designated to define attributes to the shape, such as colour and size. Finally an area of RAM akin to the Teletext RAM contains pointers for each screen location that point to the desired shape and its associated attributes. The advantage of this system is that large, dramatic changes can be made to the display very rapidly by making simple changes in the pattern look-up table. Further, all displays of one particular shape can be modified simultaneously by updating the pattern generator table.

A distinction is made in hardware between graphics and text modes, though text is available in graphics mode. They may be interchanged using the BASIC commands TEXT and GRAPH. In GRAPH mode the resolution is 256×192 (48K pixels), and the colour of each pixel can be set to any of the 16 available display colours, the only limitation being that only two colours per eight pixels may be used horizontally. To put it another way, the background may be any one of the 16 colours, while the colours of the pixels (foreground) can be set to one of 16 colours in blocks of 32×192. Control is exercised from BASIC using the COLOUR and PLOT commands.

In TEXT mode the screen format is 24 rows of 40 characters per row, each character being defined by a 6×8 bit matrix. In this case an attribute table is not required; instead the character colour is defined by a colour register on the 9929 itself that stores one foreground and one background colour. The character shapes are held in RAM in the VDP's memory

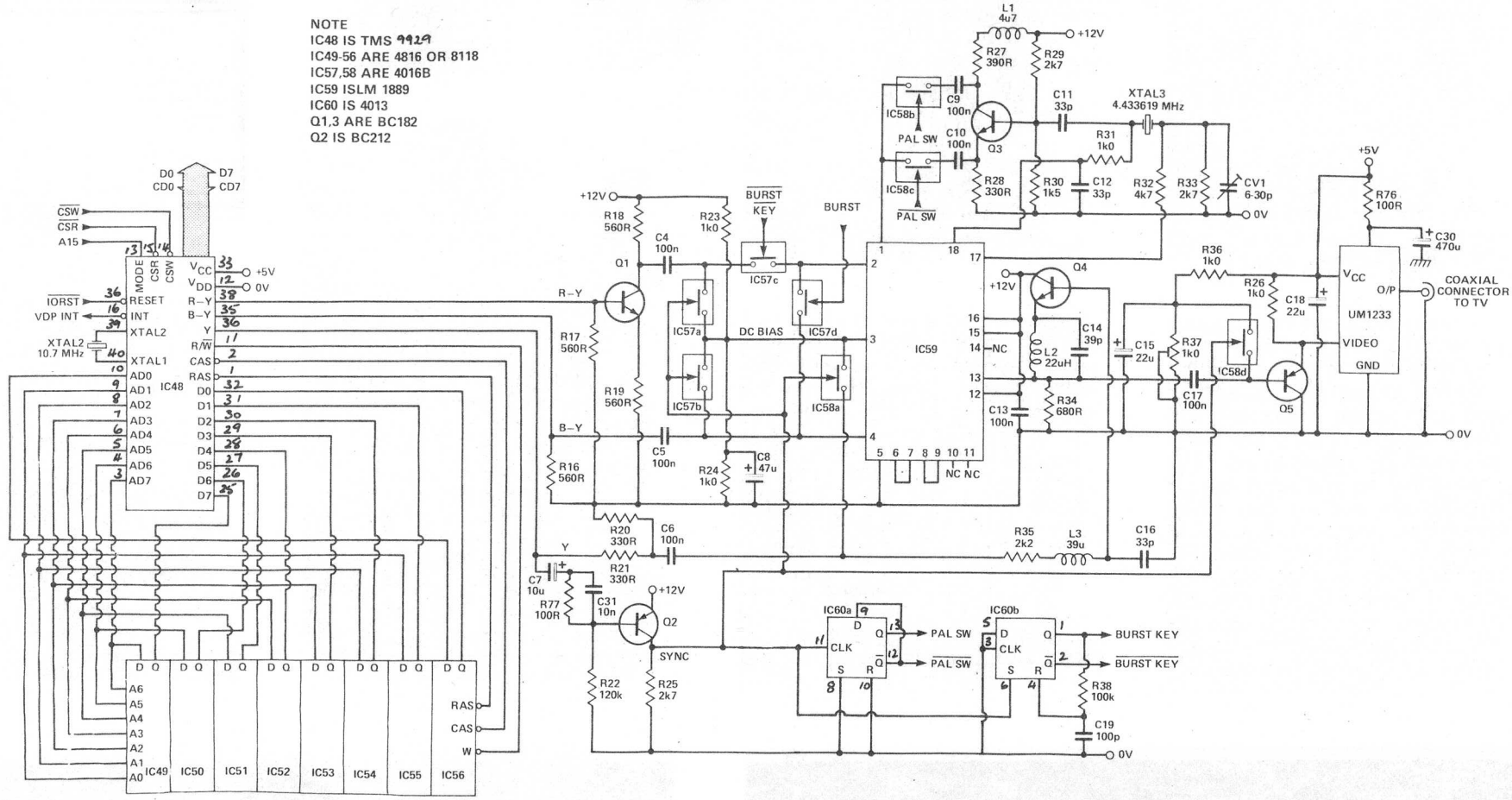


Fig. 4 Circuit diagram of the video display circuitry.

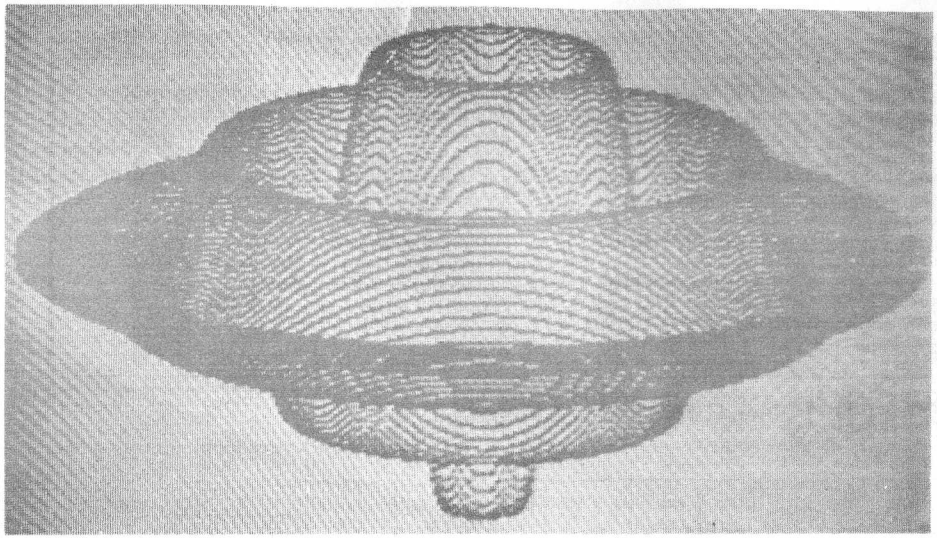
HOW IT WORKS — VIDEO DISPLAY PROCESSOR AND PAL ENCODER

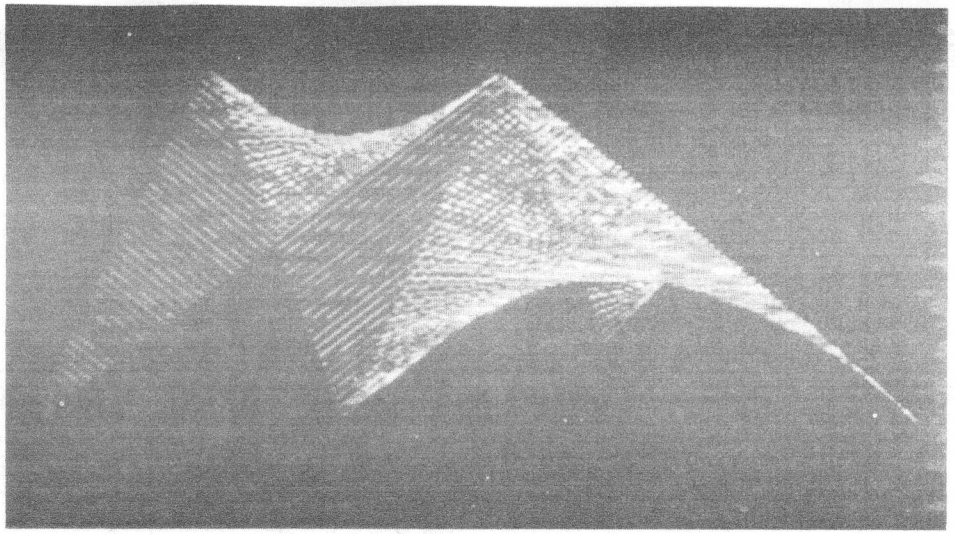
The TMS9929 (IC48) is a 625-line non-interlaced video display processor; it directly drives 16K of memory which is completely separate from the main CPU memory. The VDP fetches data from its DRAM (ICs 49–56) at such a rate that the DRAM is automatically refreshed many times over. There's very little else to say about this section of the circuitry — IC48 does everything internally! The VDP outputs a composite luminance and sync waveform on the Y output and colour difference signals on the R–Y and B–Y outputs. These signals contain all the information needed to produce either R–G–B or PAL-encoded colour. The VDP is controlled by a two-byte memory-mapped slot in high memory.

The sync is separated from the Y (luminance) signal by Q2 and associated circuitry, and used to drive DC restoration clamps IC57a,b,58a; these charge capacitors C4, 5, 6 to a reference voltage during the sync pulse. The sync is also used to toggle the PAL (Phase Alternate Line) switch, IC60a, which gates an inverted or non-inverted chroma oscillator signal into one of the two analogue multipliers in IC59. The chroma oscillator is built around Q3 and XTAL3, a crystal whose resonant frequency is 4.433619MHz, that of the PAL colour subcarrier. The non-inverted signal is taken from the collector of Q3 via C9 and IC58b, while the inverted signal is taken from Q3 emitter via C10 and IC58c.

The second flip-flop, IC60b, is used as a monostable which, at the beginning of each line, connects the burst pulse which occurs on the B–Y signal to the R–Y input of IC59. This switching is done by IC57d. The inverting amplifier Q1 on the R–Y line from the VDP, IC48, is to match the direction of the burst pulse with the direction of the video signal to yield the correct colours.

The luminance signal is low-pass-filtered by R28, L3, C16 and then summed with the chrominance output of IC59 via the chroma trap L2, C14; this filter removes colour fringing effects. The signal is then DC-shifted by another DC restoration clamp (IC58d) to feed the RF modulator.





and so may be easily user-modified. The CHAR command allows any of the 256 possible character definitions to be altered.

Table 1 shows the 16 colours which are available; this 'palette' has been arranged to give not only a good colour display, but also a good monochrome display, as the colours produce an even grey scale on a black-and-white TV. Eight grey levels are generated.

One peculiarity may have caught your eye in Table 1: what is the point of a transparent colour? A transparent object will allow you to see what's behind it, but in most graphic displays 'behind' is meaningless. However, the VDP in the Cortex considers its display to consist of 36 planes prioritised one above the other. When you look at the screen you're seeing an image which can be considered analogous to holding 36 colour slides, one above the other in a stack, and peering through them all.

TABLE 1

Code	Colour	Code	Colour
0	transparent	8	medium red
1	black	9	light red
2	medium green	10	dark yellow
3	light green	11	light yellow
4	dark blue	12	dark green
5	light blue	13	magenta
6	dark red	14	grey
7	cyan	15	white

The rearmost plane is black to allow images to be built up over it. The next plane is for external video and need not concern us here. On top of this is the backdrop plane which lies directly behind the text/graphic plane. This defines the border colour as well. Since this plane defines the colour of the whole screen, it is now obvious that the only way to see the external video input or the black rearmost plane is to set the backdrop to transparent. The text/graphic plane is written to by the TEXT and GRAPH commands discussed earlier.

This leaves another 32 planes sitting in front of the four mentioned above; these are called the sprite planes. A sprite is a graphic shape that can be user-defined from BASIC with the SPRITE command. Sprites can be displayed in a variety of sizes depending on the size and magnification flags; these give four possible modes. SIZE 0 means that a block of 8×8 bits is used to define the sprite, while SIZE 1 uses 16×16 bits (but reduces the total number of different shapes from 256 to 64). The display size can be varied with the MAG command; MAG 0 maps one bit in the shape onto one pixel while MAG 1 maps one bit onto a 2×2 block of pixels on the screen.

Each sprite has four attributes associated with it; its plane (or priority), its colour and its X and Y screen coordinates. Again, each sprite can be one of 16 colours, and those bits set to 1 in the sprite definition adopt the defined colour while the other bits are set to transparent. The screen coordinates define the position of the top left-hand corner of the sprite, and sprite positions can therefore be rapidly changed by simply altering two bytes in memory. The colour can be changed equally quickly by altering one byte. Because the planes are prioritised, 0 to 31, if any shape is positioned coincident with another shape, only the one with the highest priority plane will be displayed. This gives rise to simple 3D simulation. A status flag is set to indicate when any two sprites 'touch' each other. Any point in the text or graphic plane will only be seen if all the points directly above it on the 32 sprite planes are transparent.

All these features mean you can generate very versatile and complex displays; but they also use up a reasonable amount of memory. We don't believe that screen RAM should be stolen from the user RAM, so the VDP only occupies two bytes in the CPU memory map. These two registers are all that's required to write all the relevant information through the VDP chip and into its own 16K of DRAM.

CORTEX PART 2

Build yourself a better brain: in this article we explain the remaining Cortex circuitry and the construction of the main board.

Serial I/O on the Cortex is handled by a versatile UART, the 9902. The CPU communicates with the UART via its serial I/O bus, based on the Communication Register Unit or CRU, which requires only three wires; thus the device fits easily into an 18-pin package. The 9902 is fully programmable and the range of variations is so great that it's outside the scope of this article. In the Cortex the chip is configured to handle RS232 eight-bit codes with even parity and 1½ stop bits; the communication rate can be set from BASIC using the BAUD command and the device is activated using the UNIT statement. The parameter for UNIT is a 16-bit word, each bit corresponding to a channel that can be either on (1) or off (0).

Channel 0 is the keyboard/screen channel; channel 1 is the 9902 that is already wired into the PCB. Channels 2–15 are implemented in software and only require the addition of extra 9902s on the CRU bus. The Cortex powers up set to UNIT 1. Executing UNIT 2 disables the keyboard and passes control to the 9902. UNIT 3 enables both simultaneously.

TABLE 1

DDEN	SIZE	TRANSFER RATE (kHz)	DIVISION RATIO (IC87)	MONOSTABLE PERIOD (uS)	COMMENTS
0	0	125	12	2·0	5¼" single density
1	0	250	6	1·0	5¼" double density
0	1	250	6	1·0	8" single density
1	1	500	3	0·5	8" double density

Construction

The main board and the keyboard both have plated-through PCBs, ie there are tracks on both sides and connections between the sides are made by the copper that has been plated onto the sides of each hole. There are therefore no track-link pins; it is, however, good practice to apply solder to EVERY hole to reinforce the connections which in some cases carry power. This happens automatically when boards are 'flow soldered' by

HOW IT WORKS—I/O

The I/O map space is split into two regions; the bottom region is for on-board I/O devices and the top region causes an off-board access. (The CPU has an internal I/O area of 16 bits, some of which is reserved for specific hardware functions; the rest is free for the user.) The on-board I/O area of the Cortex is decoded by IC34 into eight 32-bit slots, of which only four are used. Two slots (CS A and CS C) are used for the Asynchronous Communications Controllers (ACCs), the third (CS B) for the parallel I/O for the keyboard data, flags and control lines (such as 'ROM', mentioned in the Memory section), and the fourth for the DMA controller IC8 (CS D).

NOTE
IC16 IS 74LS07
IC61 IS 74LS74
IC62,63 ARE 74LS251
IC64 IS 74LS259
IC65 IS 74LS32

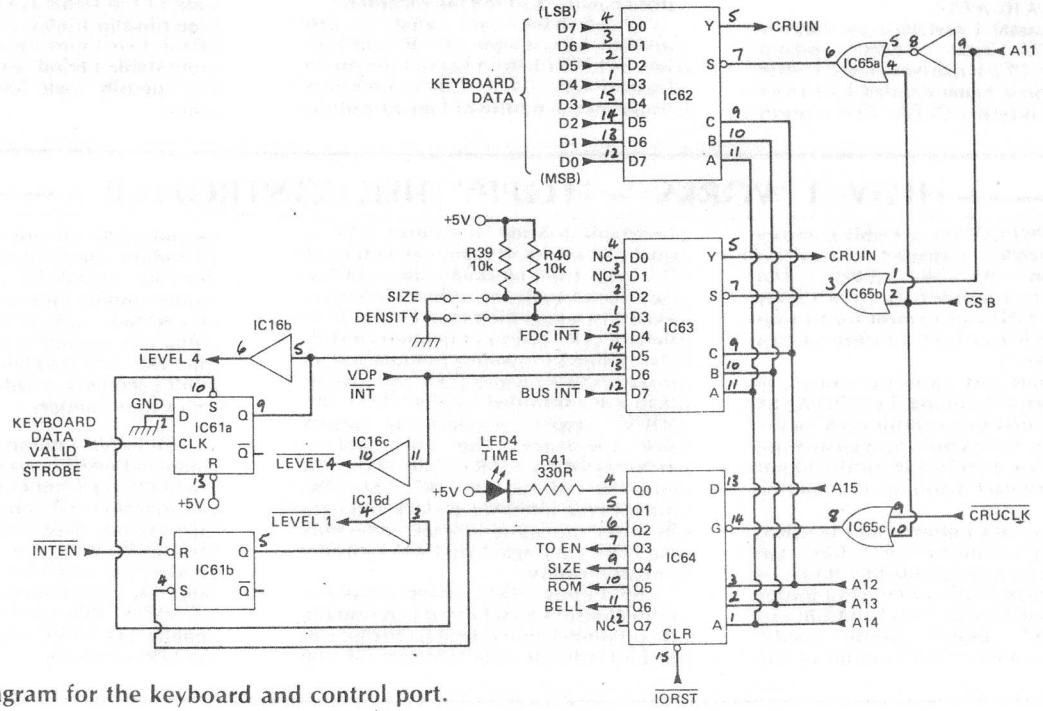


Fig. 1 Circuit diagram for the keyboard and control port.

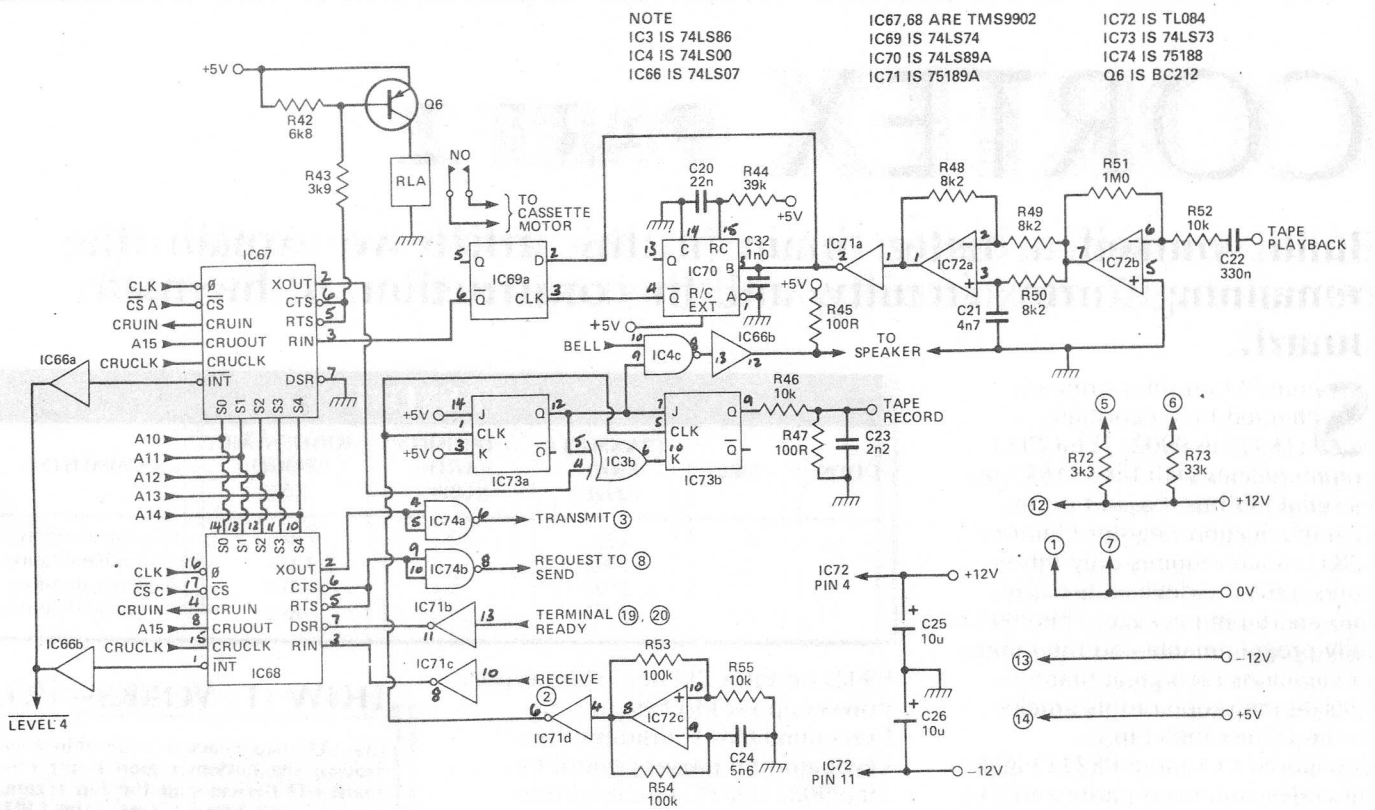


Fig. 2 Circuit diagram of the RS232 and cassette interfaces. Circled numbers are RS232C socket numbers.

HOW IT WORKS — RS232 AND CASSETTE PORT

The RS232 port consists of IC68, a TMS9902 Asynchronous Communications Controller (ACC) and the TTL-to-RS232 signal level shifters (IC74a,b and IC71b,c). IC68 is a completely software-controlled device; its baud rate can be set at anything from 46 baud to over 100,000 baud. The number of bits to be transmitted or received can also be changed, as can the type, the parity and number of stop bits. The CPU drives the ACC through the serial I/O (CRU) bus. The ACC is decoded as a 32-bit block, each bit being selected by the five address lines A10–A14.

The cassette interface uses another ACC, IC67. First a 4.8kHz op-amp oscillator (IC72c) drives a level shifter (IC71d) before being divided by two in the first flip-flop (IC73a). This ensures that the waveform has a unity mark-space ratio. The serial output from IC67 then controls the action of the second flip-flop, IC73b, via the EXOR gate IC3b. When the output is high, IC73b acts as a shift register, passing through the 2.4kHz tone; however, when the ACC output goes low then synchronously at the next clock pulse, IC73b starts to divide by two, hence generating 1.2kHz. The key point here is the synchronous switch from one tone to the other. The signal is high-pass-filtered and attenuated by R46, R47 and C23 before passing to the tape recorder.

On playback the signal is first amplified by a factor of 100 and buffered in IC72b before going through an all-pass filter, IC72c. This is necessary because of the nature of tape recording. When square waves are recorded on tape they are accurately captured; however, on playback frequency equalisation is carried out in the tape recorder but the phase relationship is destroyed, resulting in a 'spiky' sine wave. This is corrected by the linear phase-shift-versus-frequency characteristic of the all-pass filter. Thus the original square wave shape is recovered at the output of IC72a. This is then level-shifted by IC71a and used to trigger a monostable (IC70a). At the end of the monostable period (312.5uS) the state of the signal is sampled by the D-type flip-flip IC69a. As the half-periods of the two tones lie either side of the monostable period, each tone generates the opposite logic level at the sample point.

HOW IT WORKS — FLOPPY DISC CONTROLLER

The TMS9909 (IC76) is a highly complex micro-controller, designed to work in conjunction with the TMS9911 DMA controller to transfer data from floppy discs. The FDC can control up to four drives which can be a mixture of two sizes or types.

All signals that go to the drives are open-collector buffered by IC80,82,83 and terminated by a resistor pack on the last drive in the chain. The signals from the drives are terminated on the board by a resistor pack and then buffered by IC84.

The raw data pulses from the drive, after being buffered by IC84a, are stretched by a monostable (IC70b) by an amount dependent on the data transfer rate selected by the 'SIZE' I/O bit and the 'DDEN' (double density enable) signal (see Table 1). The output of the monostable is used to control IC77, a digital phase-locked loop. The output of IC77 is, in the unlocked state, half the input clock frequency. When the loop is locked to a signal then the PLL inserts or deletes clock pulses in the pulse stream, thus shifting the average frequency. The programmable divider IC87 and divider IC69b are controlled by the 'SIZE' and 'DDEN' signals to select the correct clock frequency. The raw data is synchronised by IC88 to the PLL clock and then fed to the FDC. The FDC separates the interleaved clock and data bits from the pulse stream and sends data bytes via single byte DMA transfers to main memory.

Mini-floppy ($5\frac{1}{4}$") drives require a motor control signal to start and stop the disc rotating. Upon starting, the disc will not be ready for data transfers for one second while the disc gets up to speed. To reduce the time required to access the disc repeatedly IC79b keeps the motor running for five seconds after it is de-selected and IC79a provides the initial one second 'not ready' signal to the FDC. For standard (8") drives that don't generate a 'ready' signal there is a set of four jumpers.

The BASIC interpreter has a 'BOOT' command which causes the FDC to read the first track from disc 1 and execute it as a machine code program. This could, for example, then search for and load the UCSD interpreter. In order that the system can boot from any type of disc there are two jumpers called 'SIZE' and 'DENSITY' which are read by IC63. This enables the BASIC interpreter to set up the FDC correctly.

NOTE
IC3 IS 74LS86
IC69 IS 74LS74
IC70 IS 74LS89A
IC76 IS TMS9909
IC77 IS 74LS08
IC78 IS 74LS32
IC79 IS LM339
IC80,82,83 ARE 74LS07
IC81 IS 74LS04
IC84 IS 74LS244
IC85 IS 74LS139
IC86 IS 74LS297
IC87 IS 74LS163
IC88 IS 74LS74

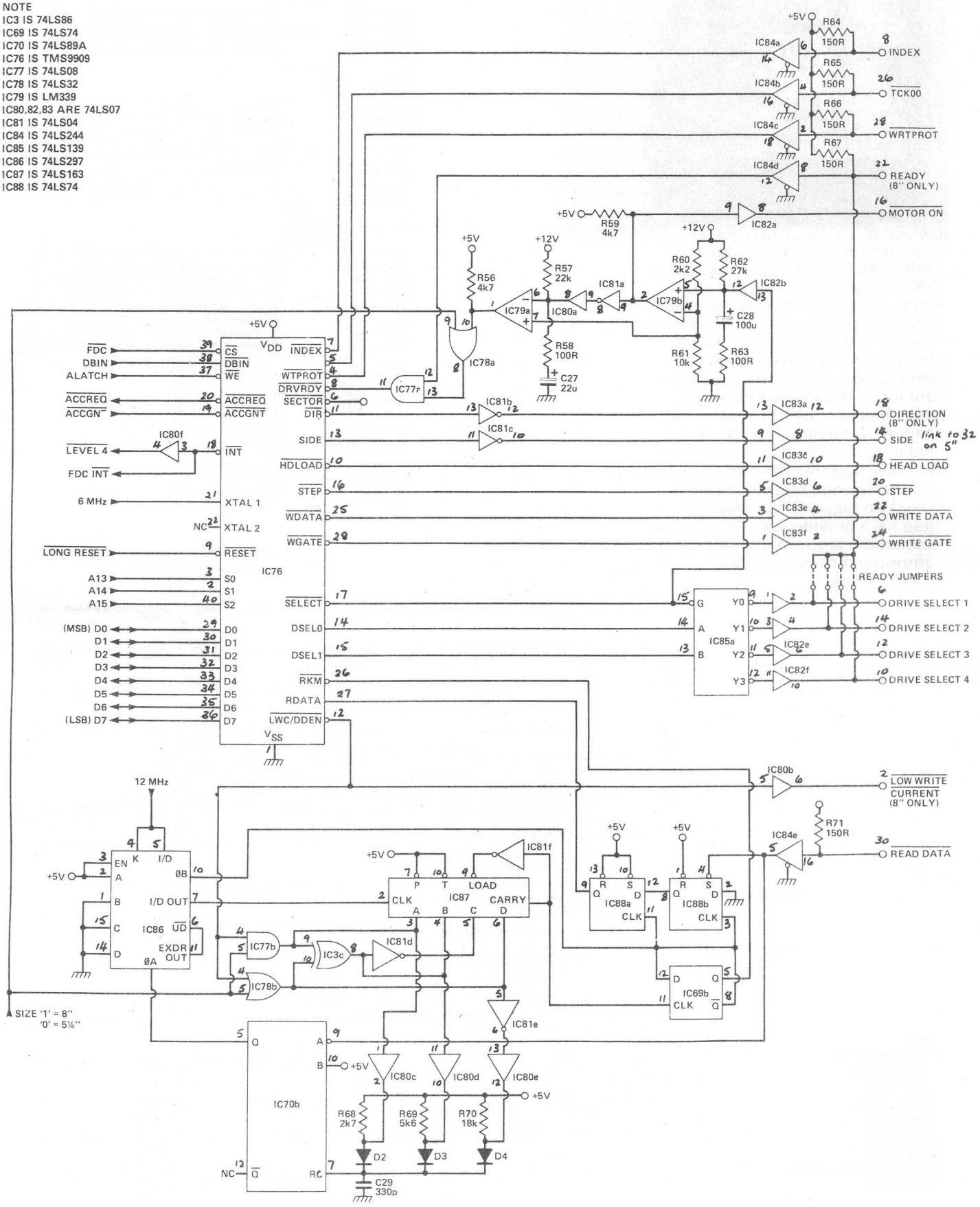
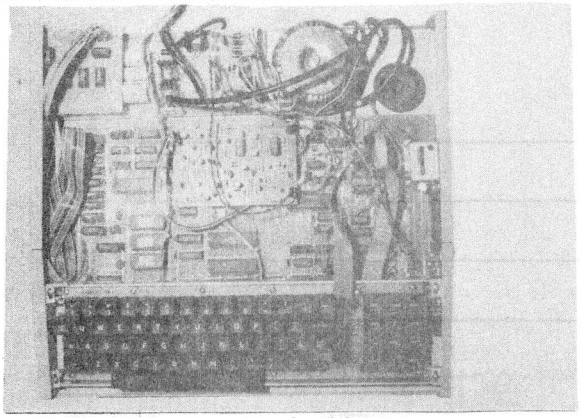


Fig. 3 Circuit diagram for the floppy disc controller section.



passing over a wave of solder in a solder bath during factory assembly. With plated-through boards it is particularly important not to make errors of construction as removal of soldered-in parts is more difficult than on conventional boards and the chances of this being required are much reduced by fitting ALL parts before soldering — if the last part left for fitting is not the one required for the last space you can be pretty sure that the required part is in the wrong holes! IC sockets should be regarded as essential; these are provided with the kits and should be fitted with the index mark corresponding with the index mark on the overlay.

NOTE
IC12,27 ARE 74LS04
IC13 IS 74LS08
IC16 IS 74LS07
IC66 IS 74LS07
IC89 IS 74LS2001
IC90 IS 74LS08
IC91 IS 74LS32
IC92 IS 74LS74
IC93 IS 74LS00
IC94-96,98,99 ARE 74LS244
IC97 IS 74LS245

Fig. 5 Circuit diagram for the E-BUS interface. Since this circuit was first published, difficulties over the supply of IC89 (74LS2001) have arisen, and a revised but less versatile circuit has been described on pages 19-21

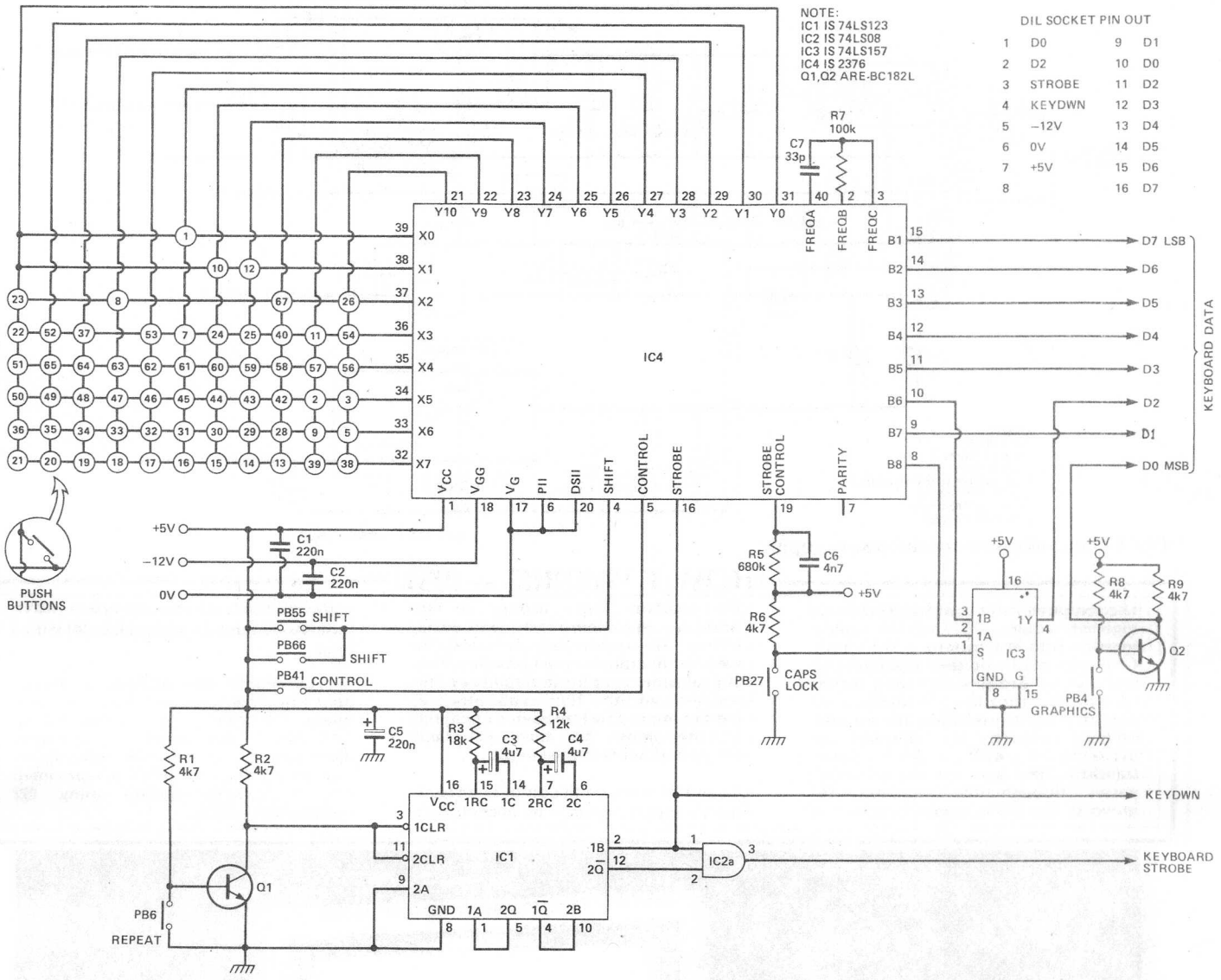


Fig. 6 Circuit of the keyboard; note that the component numbering is separate from all the previous circuitry.

HOW IT WORKS — E-BUS

The E-BUS is a powerful and compact bus which allows many intelligent cards to share a common resource of memory and I/O cards. In order to share out the resources on the bus, each card has a priority according to its position. This is done by passing a signal down the bus which goes into each card as GRANTIN and comes out as GRANTOUT to form the GRANTIN of the next card. A second signal, BUSY, tells each card if the bus is in use or free. If the bus is free and a card requires the bus, it disables the lower priority cards with the GRANTOUT signal and if the GRANTIN signal and BUSY are OK it asserts BUSY and enables its data and address bus buffers.

Once the bus transfers are complete or if a higher priority card requires the bus, then the card will relinquish control. All these events are synchronised by a backplane clock, BUSCLK. Each data transfer that takes place must signal its completion using READY.

The 74LS2001 gate array (IC89) contains the bus arbitration and control logic to gain and release the bus with timeouts upon error conditions. If the card cannot gain control of the bus after 128 clock cycles, it aborts with a timeout interrupt. Also, if after 16 clock cycles the transfer has not been signalled as complete using the READY line, the controller completes and issues a timeout interrupt.

The E-BUS has provision for a multibit interrupt code signalled by the INTEN signal. This interface only provides a single interrupt level using the INTEN signal. The data, address and interrupt signal are multiplexed onto the same pins to conserve connections. The ALATCH signal is used to enable the address latches when the address is on the bus. Then either DEN or WE will be signalled, to show that either a data read or write is occurring and that data is now on the bus. The INTEN signal can be used to latch the interrupt code.

KEYBOARD

The keyboard is a separate unit providing a fully encoded output. Most of the work is carried out by the 2376 keyboard encoder (IC4). This IC contains a 50 kHz oscillator and two ring counters of eight and 11 stages, the outputs of which form an XY matrix across which the switches are connected. By this means each key is sequentially scanned. The closing of one of the switches for a sufficient length of time for switch bounce to be completed causes the scanning to stop; a 'valid' signal now appears on the strobe output. The encoder also contains a 2376-bit ROM (hence the IC name) arranged as three groups of 88 words of nine bits. The shift and control inputs select one of the three groups and the individual word is addressed by the ring counters.

IC3 is a data selector. D2 is either the output B6 or B8 depending on whether upper or lower case characters are selected by the CAPS LOCK switch. Repeated entry of a character is accomplished by multiple strobe signals from IC1, which is a dual monostable arranged as an oscillator and is enabled by a high level on the clear inputs.

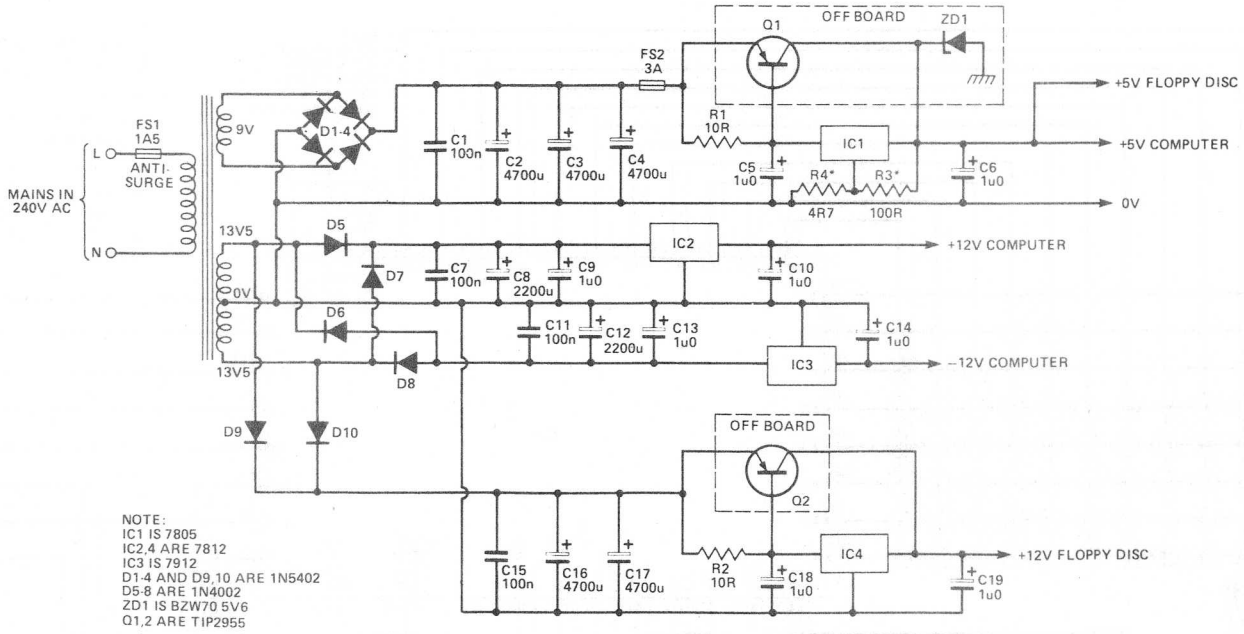


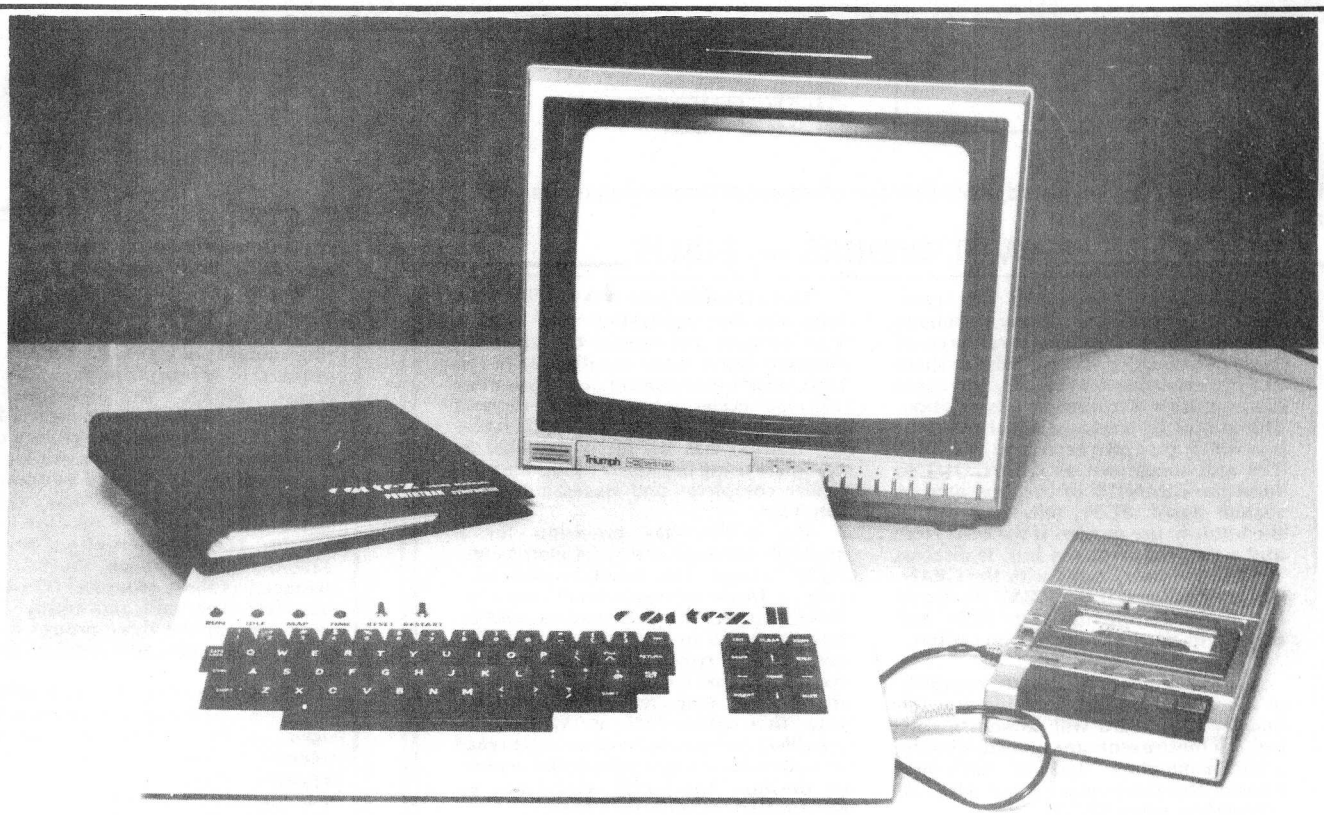
Fig. 7 Circuit diagram for the power supply.

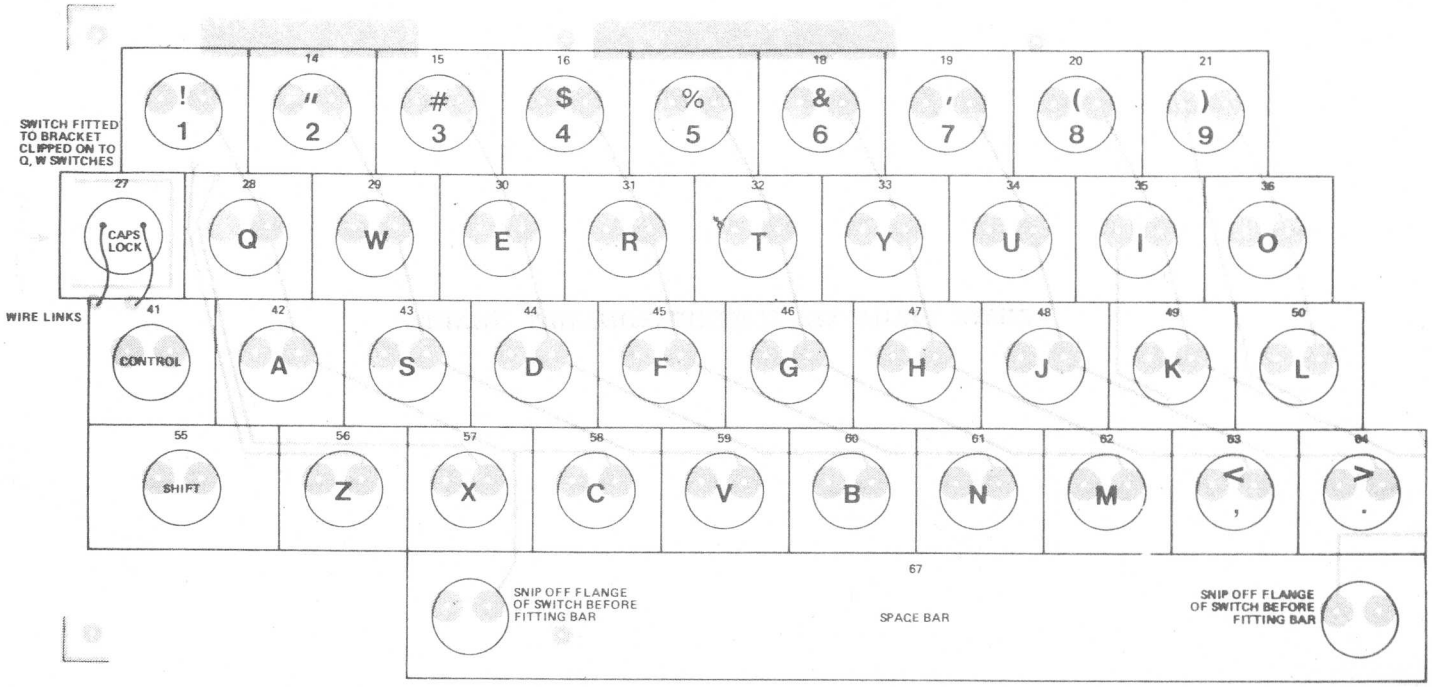
HOW IT WORKS — PSU

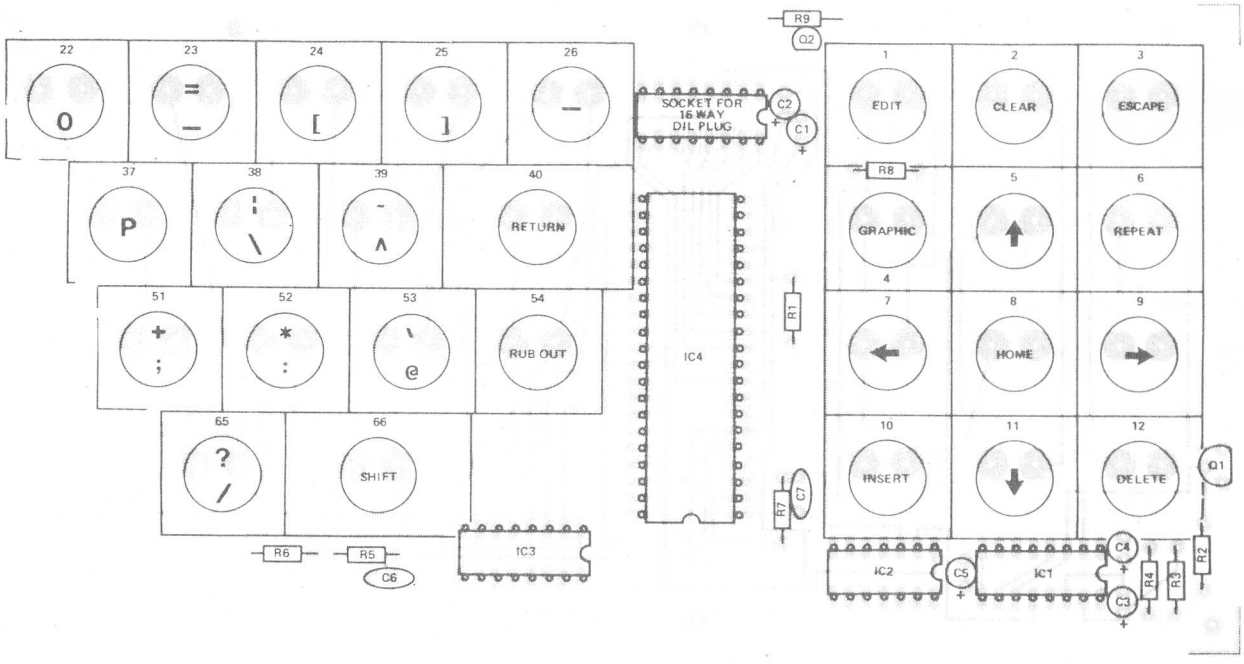
The computer main board and keyboard together require a 5 V at 3 A supply, together with low current ± 12 V rails. One amp plastic voltage regulators on small finned heatsinks are used for the 12 V supplies; for the 5 V supply a 1 A regulator is also used but the current-carrying capacity is boosted by bypassing it with a 15 A power transistor, the base current of which passes through the regulator. R1 prevents the off-load input current of the regulator from turning on the transistor when there is no load during testing. The resistor also increases the speed of operation of the transistor. The 1uf capacitors are for the stability of the regulator and the 100nF capacitors are used to remove fast transients orginating from the mains. The zener will clamp any spikes that reach the output.

R3 and R4 increase the output voltage of the 5V supply and may be needed if the volts at IC48 are low (less than 4.8V); otherwise omit R3 and replace R4 with a link.

To simplify the addition of floppy discs these are powered from the same board. The drivers require about 0A7 at 5 V which is also supplied by Q1; they also require + 12 V at 1A6 with higher surges at switch-on, and this is provided by a separate section using Q2 controlled by IC4.







The keyboard overlay, in two halves.

THIS PAGE IS INTENTIONALLY BLANK

CORTEX CENTRONICS INTERFACE

Any follow-up on the Cortex is long overdue, for which we apologise. We're now about to start setting this to rights.

The Cortex 16-bit computer was designed in late 1982 and was featured by ETI in the November 1982, December 1982 and January 1983 editions. It is based on the TMS9995 16-bit microprocessor with a full 64K byte main memory and a separate 16K byte memory for the colour graphics display. The machine can be expanded by simply adding chips to provide features such as floppy disc support and an expansion bus for extra memory, input and output. This article details how to use the EBUS bus expansion to plug an external board into your Cortex to provide a parallel data interface for a Centronics printer.

The E-BUS was developed to provide a compact, high-performance and flexible interface for both memory and input/output expansion. The E-BUS system multiplexes both addresses and data information onto its lines at different times, so as to keep the number of lines needed as low as possible. It can be shared by multiple microprocessors for access to common memory or I/O.

With regard to I/O expansion, some explanation of the computer's operation is necessary. When doing input and output operations, the CPU does not communicate in eight or 16-bit words, in contrast to most common processors which support dedicated I/O commands. Instead, the 9995 sends and receives just one bit at a time. This means that multi-bit input and output functions, like those needed for a Centronics printer for instance, have to be built up or broken down one bit at a time.

However, this operation is made simpler by the fact that each bit can be sent to or received from a unique address. So, to output eight bits to a peripheral, it is necessary to transfer the bits to a latch or similar device with eight successive single bit operations. Only then can the peripheral be told that the data is ready to be acted upon.

An input cycle occurs when the $\overline{\text{MEMEN}}$ (memory enable) control signal goes high (inactive); the bit address is output on the address/data bus and the single bit of data is sampled on the CRUIN signal. An output cycle is similar, except that the data bit is output on the CRUOUT signal and the $\overline{\text{CRUCLK}}$ control signal pulses active low. For multi-bit transfers, the cycle repeats with the address incrementing each time. A timing diagram for this is shown in Fig. 1.

During an I/O operation, the multiplexed address/data lines are forced to output the address throughout the I/O cycle so no address latches are required, consequently the ALATCH (address latch) control signal remains high. The data bit to be sent appears as the signal CRUOUT on the least significant address line (A15 by the TI convention) when an output operation is performed. The remaining 15 address lines define 32768 addresses to which it can be sent, although the Cortex and the 9995 CPU use some of these addresses internally (see Fig. 2, the Cortex I/O map).

A data bit to be read in is taken via the CRUIN line. Up to 16 bits can be input or output by a single instruction, the specified number of bits being processed serially to or from the target addresses.

The E-BUS

The kernel of the E-BUS interface on the Cortex is the 74LS-2001 gate array. This device is used to control access to the bus

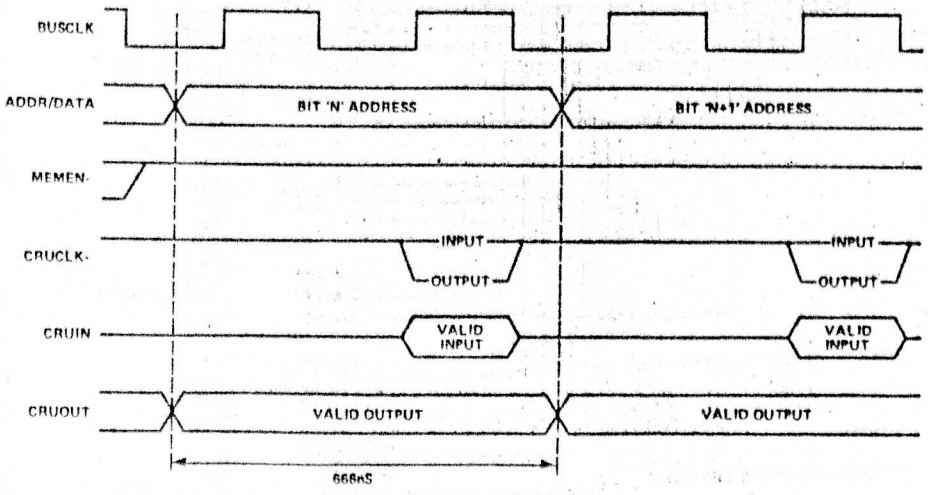


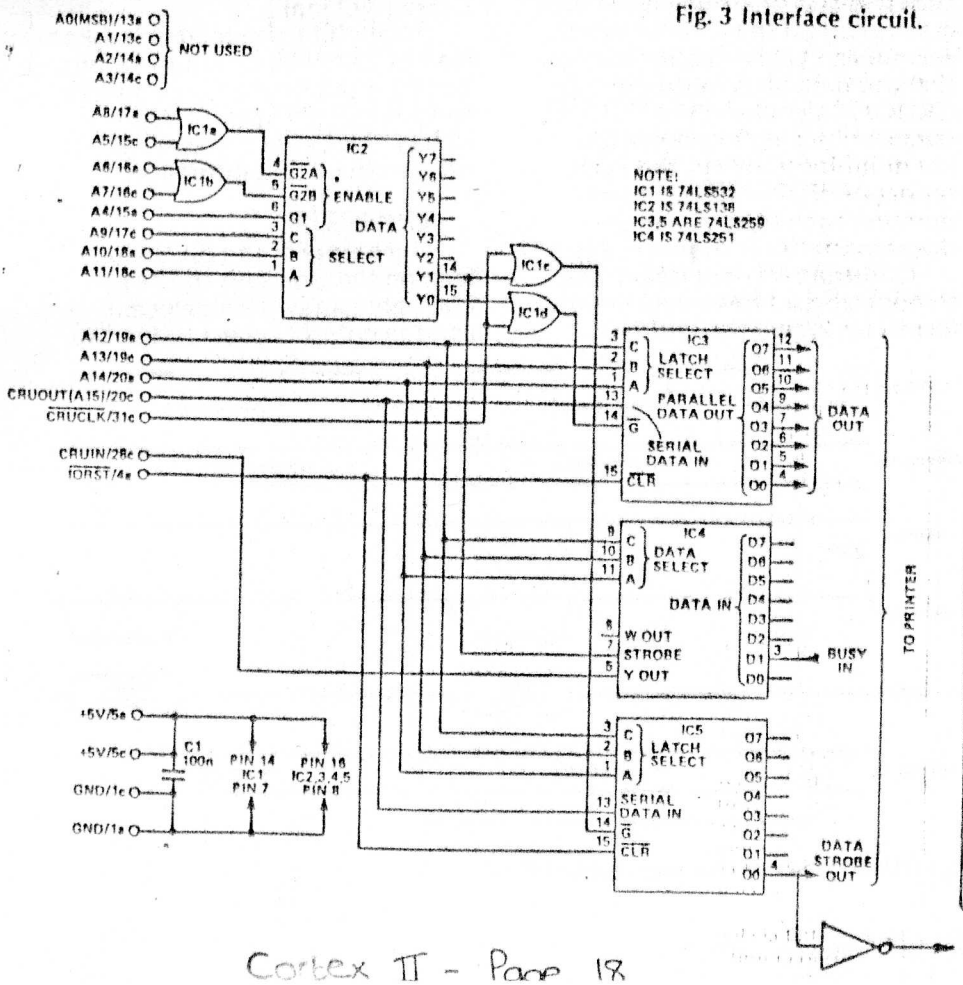
Fig. 1 Timing diagram for multi-bit transfers.

Address	Input Function	Output Function
0000	NOT USED*	"BASIC" LED
0002	NOT USED	KEYBOARD ACKNOWLEDGE
0004	DISK SIZE JUMPER	EBUS INTERRUPT ACK
0006	DISK DENSITY JUMPER	EBUS TIMEOUT ENABLE
0008	FLOPPY INTERRUPT FLAG	DISK SIZE
000A	KEYBOARD INTERRUPT FLAG	EPROM ON/OFF
000C	DISPLAY INTERRUPT FLAG	"BELL" ENABLE
000E	EBUS TIMEOUT INTERRUPT FLAG	NOT USED
0010 to 001E	KEYBOARD DATA	DUPLICATE OF ABOVE
0020 to 003E	DUPLICATE OF ABOVE	DUPLICATE OF ABOVE
0040 to 007E	NOT USED	NOT USED
0080 to 00BE	TMS 9902	TMS 9902
00C0 to 017E	RS232 PORT	RS232 PORT
0180 to 01BE	NOT USED	NOT USED
01C0 to 01FE	TMS 9902 CASSETTE PORT TMS 9911	TMS 9902 CASSETTE PORT TMS9911
0200 to 07FE	DMA CONTROLLER external via EBUS	DMA CONTROLLER external via EBUS
0800 to 080E	external via EBUS	Centronics data via EBUS
0810	external via EBUS	Centronics strobe via EBUS
0812	Centronics status via EBUS	external via EBUS
0814 to 1EDE	external via EBUS	external via EBUS
1EE0	CPU internal	CPU internal timer control
1EE2	" "	" " timer interrupt
1EE4	" " level 1 interrupt flag	" " enable
1EE6	" " " 3 " "	" "
1EE8	" " " 4 " "	" "
1EEA	" " **	" " **
1EEC	" " ***	" " ***
1F00 to 1FD8	" " not used	" " not used
1FDA	CPU internal MID flag	CPU internal
1FDC to FFFE	external via EBUS	external via EBUS

* Some systems may have this used for floppy disk interface
** Cortex Basic uses this bit to enable the display on the TV screen of all control characters.
*** Cortex Basic uses this bit to disable the scrolling of a text display on the TV screen.

Fig. 2 (above) Cortex I/O map.

Fig. 3 Interface circuit.



for multiple microprocessors and to synchronise all data transfers, as well as to provide time-out controls to avoid a permanent lock-up of the bus. However, for simple expansion of the Cortex facilities, it is possible to avoid using this device. Four changes to the main PCB have to be made, as follows:

1. Cut the connection to IC99 pin 18 and connect this pin to IC94 pin 11;
2. Connect IC94 pin 9 to IC11 pin 13;
3. Connect IC94 pin 19 to IC27 pin 2;
4. Connect IC89 pin 18 to IC89 pin 19.

For this modification to work, the

HOW IT WORKS

IC1a, IC1b and IC2 decode address bits A4 to A11 inclusive. Two outputs from IC2 are used each of which will go low when one of their particular group of eight addresses is accessed. Note that the address decoding is not complete and may well respond at other locations (2K interval). The Y1 output from IC2 (address 0088h to 008Fh) enables the output of IC4 (74LS251) the 1 of 8 selector. This device selects the signal on the input specified by the least significant address lines and passes it to the CRUIN line. This would normally be the printer status line in this case and its state will signal whether the printer is ready to receive data.

The data to be output to the printer appears on the CRUOUT line in serial form. This has to be converted to parallel form before being presented to the printer. This is done by IC3 which is an eight bit addressable latch (74LS259). Address lines A12, 13 and 14 determine which bit is to be written into IC3 and the output Y0 of IC2 gated with CRUCLK in IC1d actually causes it to happen. To write a complete eight bits of new data requires eight output cycles. Once this has been done the data is ready to be acted upon by the printer. IC5 is another 74LS259 like IC3 and one bit of its output is used as a strobe signal to tell the printer that the data is ready. Output Q0 is used in this case; it is normally low and is pulsed high under software control. This happens when a 1 and then a 0 is output to 0088h.

For the transfer of one character the sequence is as follows: eight bits of data are written into the eight locations in IC3. The data then appears in parallel form at its output and is available to the printer. The DATA STROBE is taken high and then low to signal to the printer that valid data is ready. The printer signals that is is BUSY by taking the BUSY STATUS line high. This is read by the computer and no further action is taken until it goes low again. Now new data can be sent to IC3 and the process repeated.

Note that some printers give a low on the status line to indicate the BUSY condition, this can be accommodated by connecting the EBUS CRUIN line to IC4 pin 6 instead of pin 5. More difficult is when the STROBE has to be inverted, this requires an additional inverter.

following devices must be present on the PCB (if your Cortex does not have all the options available, then some of these may be missing, so you will need to check): IC90 (74LS08), IC91 (74LS32), IC92 (74LS74A), IC93 (74LS00), IC94,95,96 and 99 (all 74LS244).

If you wish to use memory expansion, then you must break the links next to IC26 and drill out the shorting links next to IC94, and ensure that you have IC97 (74LS245) and IC26 (74LS612). Note that all the component numbers used here are those given in the original Cortex article.

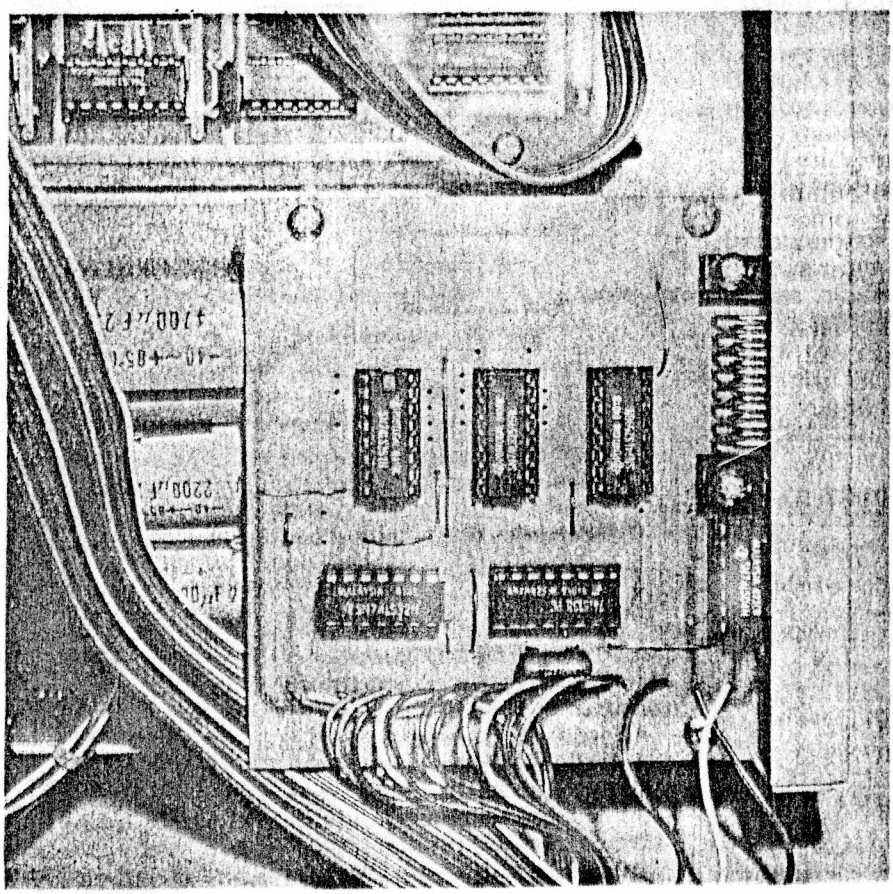
It's Already There

Although the Cortex has an RS232 interface for a printer or terminal, it also has the software necessary to drive a parallel data printer port. The hardware necessary to implement this was left off the main PCB to save space. Building the interface described here will therefore free the RS232 for other tasks, and make it possible to use the more common Centronics printers.

The circuit diagram for the interface is shown in Fig. 3 and the circuit itself is discussed in the 'How It Works' section.

I/O	ROW A	PIN	ROW C	I/O
	GND 0V	1	GND 0V	
I/O	PRES- (RESET-)	2	BUSCLK (3MHz)	O
	+12V	3	−12V	
O	IORST- (I/O RESET-)	4	NMI-(NON-MASKABLE INT)	I
	+5V	5	+5V	
	—	6	—	
	—	7	—	
	—	8	—	
	—	9	—	
I	INTEN- (LEVEL 1)	10	ALATCH (ADDRESS LATCH)	O
O	XA0 (MSB ADDRESS)	11	XA1	O
O	XA2	12	XA3	O
O	A0	13	A1	O
O	A2	14	A3	O
O	A4	15	A5	O
O	A6	16	A7	O
I/O	A8/D8 (DATA MSB)	17	A9/D9	I/O
I/O	A10/D10	18	A11/D11	I/O
I/O	A12/D12	19	A13/D13	I/O
I/O	A14/D14	20	A15/D15/CRUOUT (LSB)	I/O
	—	21	MEMEN- (MEMORY ENABLE)	O
O	DEN- (MEMORY READ)	22	READY- (MEMORY READY)	I
I	GRANTIN	23	GRANTOUT	O
	—	24	BUSY- (BUS BUSY)	I/O
	GND	25	GND	
	—	26	—	
	—	27	—	
	—	28	CRUIN (I/O INPUT)	I
O	WE- (MEMORY WRITE)	29	—	
	+5V	30	+5V	
	—	31	CRUCLK (I/O WRITE STROBE)	O
	GND	32	GND	

Fig. 4 EBUS signals and their uses.



CORTEX CENTRONICS INTERFACE

In the slightly delayed second part of this article, we present the construction and use details.

The overlay of the PCB is shown in Fig. 4. There have been three modifications between this and the original circuit given in the June ETI. Firstly, the address lines A4 and A8 were the wrong way round in the original circuit diagram and this has been corrected.

Secondly, an extra package, IC6, has been added, of which only one inverter gate is used. This is to provide a complement of STROBE as well as STROBE itself on the Centronics output; this is to increase flexibility, as some printers will require the complement rather than the original.

Finally, and also to increase flexibility, the W and Y outputs of IC4 are link-selectable; using the Y output, the BUSY IN line is inverted to become the CRUIN signal; using the W output, it is not; one or other of these will be appropriate to your printer. Needless to say, you should not use both links at the same time!

Using the special PCB, assembly of the circuit should be quite straightforward, but do make sure you get the links in the right places and be careful with IC orientation. Some clearances are a

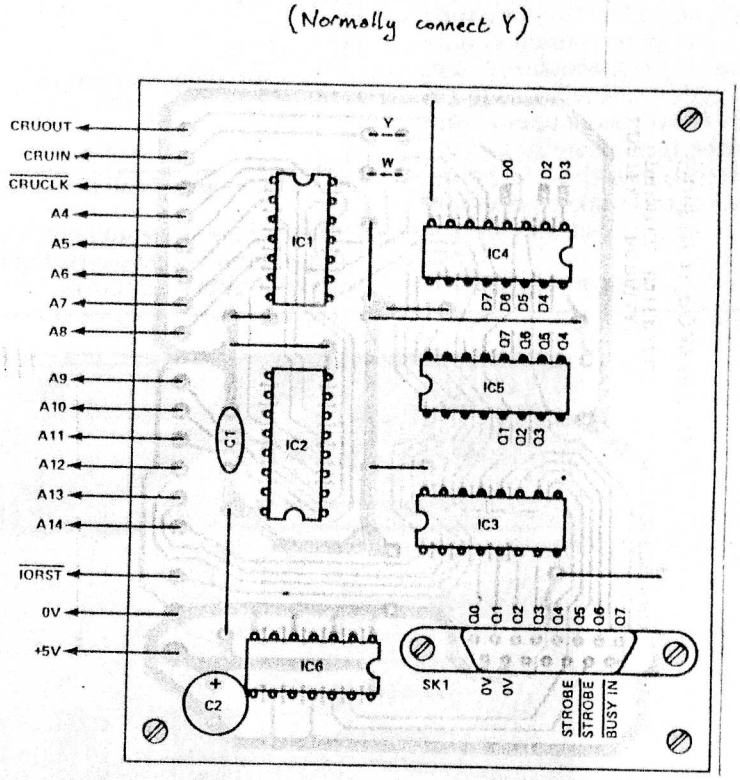


Fig. 4 Overlay diagram for the PCB. The points marked D0, D2, D3 etc and Q1, Q2, etc on the PCB (not the connector) are for the unused locations that readers may wish to make use of. Note that there are two additional decoupling capacitor location points, in the unlikely event of any supply line problems arising.

PARTS LIST

CAPACITORS

C1	100n ceramic
C2	10μPCB electrolytic

SEMICONDUCTORS

IC1	74LS32
IC2	74LS138
IC3,5	74LS259
IC4	74LS251
IC6	74LS04

MISCELLANEOUS

15-way D-type connector socket & plug; PCB; connector to suit printer; ribbon cable, etc.

BUYLINES

A full kit of parts for this project will be available through Powertran Cybernetics Ltd, Portway Industrial Estate, Andover, Hants SP10 3ET. Powertran hold the copyright on the PCB so it will be available only from them.

Fig. 5 Test program for the interface, to print a row of 'A's.

```
1Ø   BASE   Ø8ØØH          !  I/O BASE ADDRESS
2Ø   CRB (8) = Ø           !  RESET STROBE
3Ø   CRF (8) = Ø 41H       !  O/P ASCII 'A'
4Ø   IF CRB (9) = 1 THEN GOTO 4Ø     ! WAIT FOR FREE
5Ø   CRB (8) = 1 : CRB (8) = Ø       ! PULSE STROBE
6Ø   GOTO  3Ø                        ! LOOP
```

bit tight, so do check carefully for any solder bridges after you have finished.

In Use

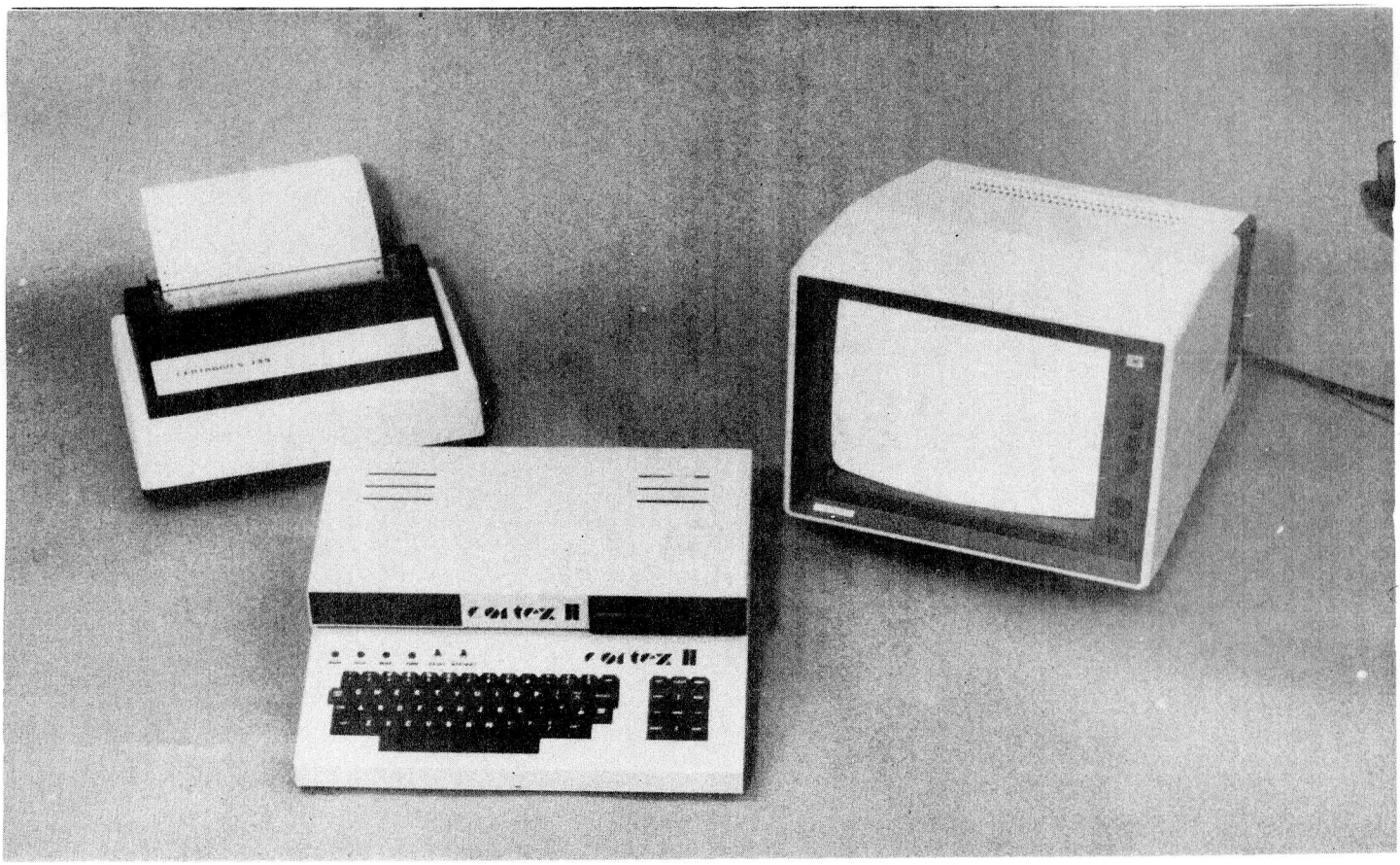
Once the interface is connected between the computer and the printer, then typing in the command UNIT 4 will enable printing, while the command UNIT — 4 will disable printing. If the printer fails to print or a paper-out condition arises, then pressing both the GRAPH and RUBOUT keys together will cause all output to be reset to UNIT 1 only.

The BASIC program shown in Fig. 5 can be used; this should print a stream of letter 'A's.

Having built the printer interface you will have noticed that there are seven spare I/O bits and if the printer is not in use then seven other parallel data ports with separate strobes and status bits can be used with a common data port. Also six other I/O address slots are decoded by IC2 as shown:—

O/P	ADDRESS	
Y0	0800	Parallel data output
Y1	0810	Single bit I/O
Y2	0820	
Y3	0830	
Y4	0840	Unused
Y5	0850	
Y6	0860	
Y7	0870	

We look forward to receiving project ideas from readers that make use of these!



A S S E M B L Y I N S T R U C T I O N S

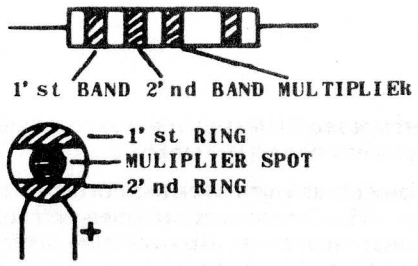
(Please read thoroughly before starting assembly)

General notes on kit construction

Identifying parts

Before assembling any part of the kit, familiarize yourself with the components. Clear yourself plenty of space in a spare room and sort out the parts according to the check list but do not mix up the packs. Read all the parts and put them in order.

Resistor Colour code



Colour	1st & 2nd band/ring	Multiplying factor
Brown	1	10
Red	2	100
Orange	3	1000
Yellow	4	10000
Green	5	100000
Blue	6	
Mauve	7	
Grey	8	
White	9	
Black	0	1
Gold		1/10

example: 390K = orange white yellow (39 × 10000 = 390000 ohms)
Note: K = kilohm (1000 ohms) M = megohms (1000000 ohms)

rather than writing 5.6K for 5600 ohms usually 5K6 is written as it avoids the risk of missing the dot and reading 56K

Capacitors: these may be marked as pico farads (P or pF or nothing at all) nano farad (n or nF) or micro farads (μF or μ or mF)

$1pF = 10^{-12}$ farad $= 10^{-6}\mu F = 10^{-3}n$

$1nF = 10^{-9}$ farad $= 10^{-3}\mu F = 0.001\mu F = 1000pF$

$1\mu F = 10^{-6}$ farad $= 1000n$

Some capacitors are colour coded. The bead tantalum types are coded in uF the striped polycarbonate types are coded in pF

Soldering

Soldering is really very easy but strangely if a constructor runs into problems with his kit it is the soldering which is nearly always at fault. To solder well it is best to understand the principles of soldering.
Solder is an alloy (of tin and lead) which is electrically conductive and has a fairly low melting point (about 190°C) making it suitable for joining together pieces of metal (except aluminium) both mechanically and electrically without causing heat damage. Solder supplied for electronic construction is in the form of wire which has small cavities containing 'flux' which is a chemical which when hot and freshly melted will remove from the pieces being joined the oxide layer and other dirt which otherwise would prevent the solder wetting the surfaces - if the solder does not wet the surfaces the joint will be unsound both mechanically and electrically - you then have a 'dry' joint.
To solder well the iron must be clean - wipe the tip on a damp sponge whenever it has a lot of black rubbish (burnt flux) on it. WARNING some beginners books talk of filing the bit or rubbing it on abrasive paper - good modern irons have iron plated bits which will be ruined by this procedure. To solder a component into a circuit board, insert its leads and bend them slightly to hold the component in place. We do not recommend flatenning the leads onto the board (except for the very thin leads of polystyrene capacitors - the ones which look like rolled up Sellotape) because doing that makes removal much more difficult if you later find you have fitted it into the wrong space. Next trim the leads to about 2mm from the board. Now comes the easy bit which so often goes wrong.

Some constructors cover the iron with melted solder and attempt to transfer this to the joint. Apart from the risk of solder spreading across the tracks, solder carried on the iron no longer has active flux to clean the joint which is necessary for the solder to be able to wet the components.

The way to make solder work for you is to touch the solder onto leads and tracks which are hot enough to melt the solder and release active flux onto the joint, so place the tip of the soldering iron onto the circuit board track touching the component lead at the same time, wait about a second then place the solder where the track, lead and soldering iron tip meet. The solder will then flow around the joint. Make sure the lead is surrounded by solder then remove the solder and iron. Now look at the joint and you will see that you DO *NOT* HAVE A BLOB OF SOLDER around the lead - you will see solder smoothly tapering off in thickness as it spreads away from the joint.

Soldering wires to component tags

On some components to which wires are attached there are holes in their tags. To solder a wire to these wrap the wire (insulation stripped off the last ¼" first) through the hole so as to hold the wire in place then heat tag and lead simultaneously and apply solder to where the lead, tag and iron meet and do not remove the iron and solder until the joint is smoothly covered which may take a few seconds with thick tags or thick wire.

Soldering wires to pins on PCB's

This calls for a different technique as it is not normal to make a mechanical joint before soldering. Holding a wire, a soldering iron and solder in place simultaneously requires 50% more hands than most people are born with so instead of attempting that, it is usual to pre-coat the pin and lead with solder and then join them with heat. This pre-coating is called 'tinning' - heat the bared wire with the iron and apply solder to where the wire and iron meet until the wire is coated - next repeat this for the pin. Hold the wire against the pin (pliers or tweezers help in some difficult to reach places), with the solder resting on the table, with the end bent up so it can be touched with the iron without burning the table melt a *SMALL* amount of solder onto the iron (this is the ONLY time loading the iron is permissible) and as fast as possible apply the iron to the joint and remove as soon as solder is seen to have joined the wire to the pin but do not yet remove your grip on the wire. As solder cools it passes through a semi molten stage (eutectic) when visually it may appear solid but is in fact very weak and if the joint is disturbed at this stage it will be permanently weakened and likely to fail later.

Cleaning of the circuit board

The importance of cleaning the track side of circuit boards cannot be overstressed. Unless all the resin has been cleaned from the joints it is very easy to miss spotting an unsoldered joint, a dry joint or shorted tracks.

To remove the flux brush the soldered joints with a stiff brush (a ½" wide paint brush with its bristles cut down to ½" length is ideal) and a suitable solvent. Proprietary board cleaners are available - RS Components are one such supplier and this can be obtained from many component dealers, however, cheaper alternatives are also available. Acetone can be purchased readily as solvent/brush cleaner for polyester resins from Strand Glass Co. the fibre glass specialist. Snopake thinners are available from stationery suppliers (Snopake is the white correcting fluid used by typists). Nail polish remover also gives reasonable results and Boots own brand is good value.

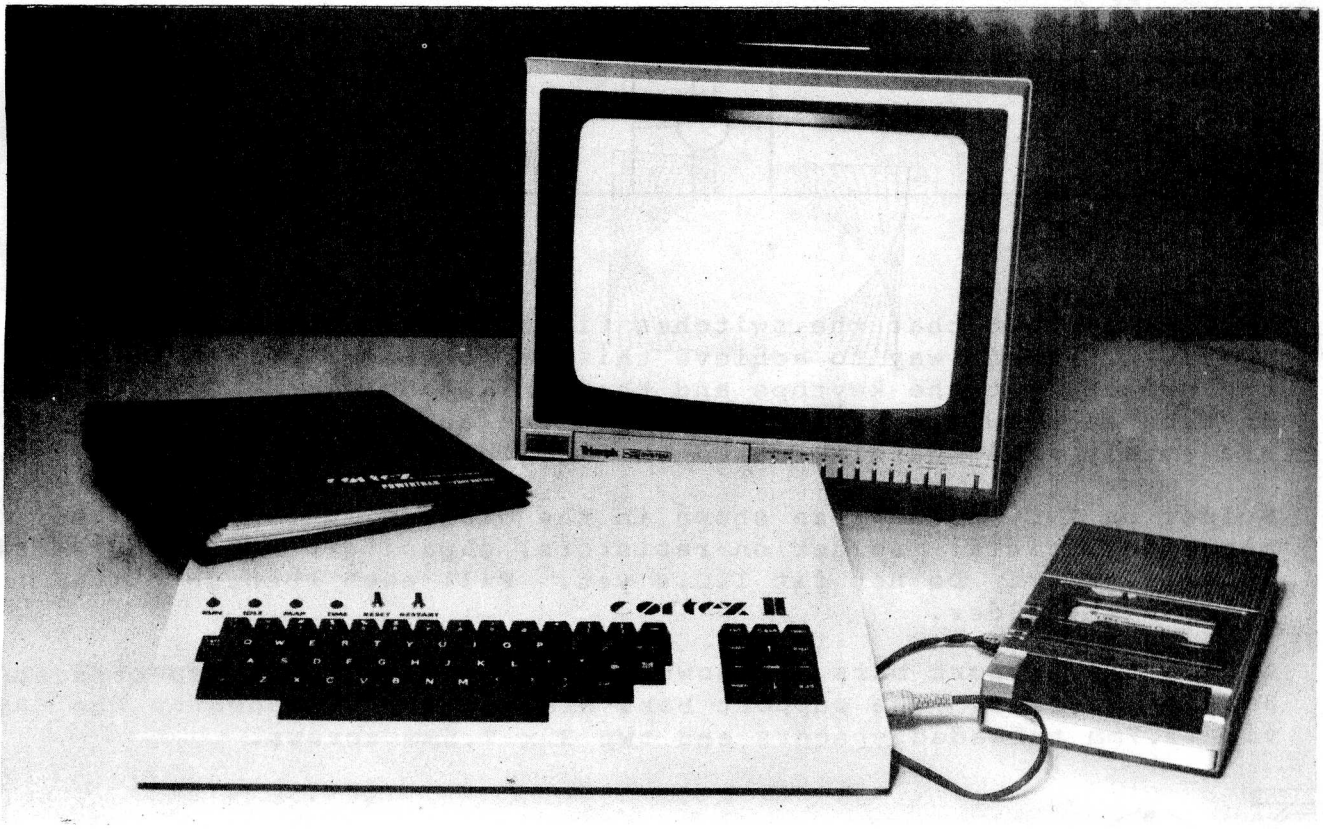
When cleaning, keep the solvent off the component side of the board as it could remove the markings of the components and possibly damage them (particularly likely with polystyrene capacitors). Solvent could flow through any spare holes on to the component side of the board so these should be soldered over before cleaning. Check the board when dry under a bright light (in our laboratory we use a 6' fluorescent fitting mounted 22" above the test bench) look very closely (a magnifying glass helps) at each joint and also all regions where tracks are close to each other. Sometimes very thin whisps of solder fall across a pair of tracks and these have to be looked for very closely. Even with very careful soldering a surprising number of faults can be revealed by this examination.

INTRODUCTION

The Cortex should be built in the following sequence:

1. Keyboard
2. Powerboard
3. Case
4. Main Board
5. Final assembly

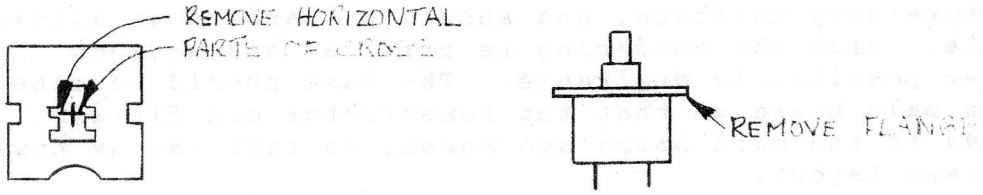
This sequence should be followed because the main board is by its nature very delicate, and should be handled as little as possible. Also the soldering is critical and as much practice as possible is desirable. The case should be constructed before the main board so that the constructor can fit all ancillaries to the main board and become as familiar as possible with the case layout.

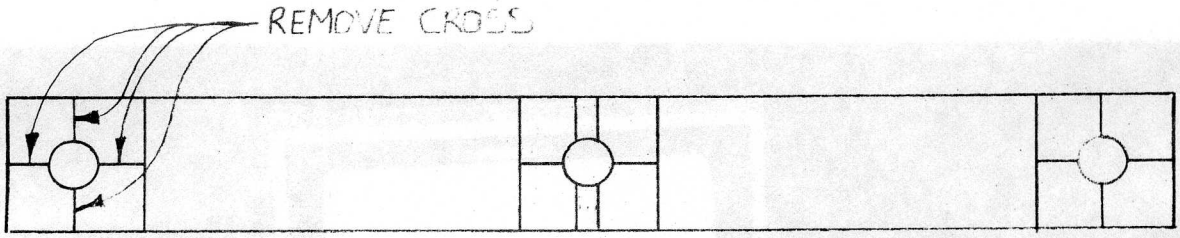


KEYBOARD ASSEMBLY

The CAPS LOCK switch is physically different from the rest and is wired in with wire links. The switch is supplied attached to a bracket which is secured to the Q, W switches. When soldering on the wire links be as quick as possible to avoid overheating which could damage the switch.

Before fitting the 2 switches for the space bar cut off the horizontal parts of the cross on the stem with side cutters and trim off any remainder with a razor blade or Stanley knife. If this is not done then the space bar will not fit securely. Also remove the square flange with side cutters to prevent it fouling with the space bar. In the space bar cut out the cross part where the switches fit in.

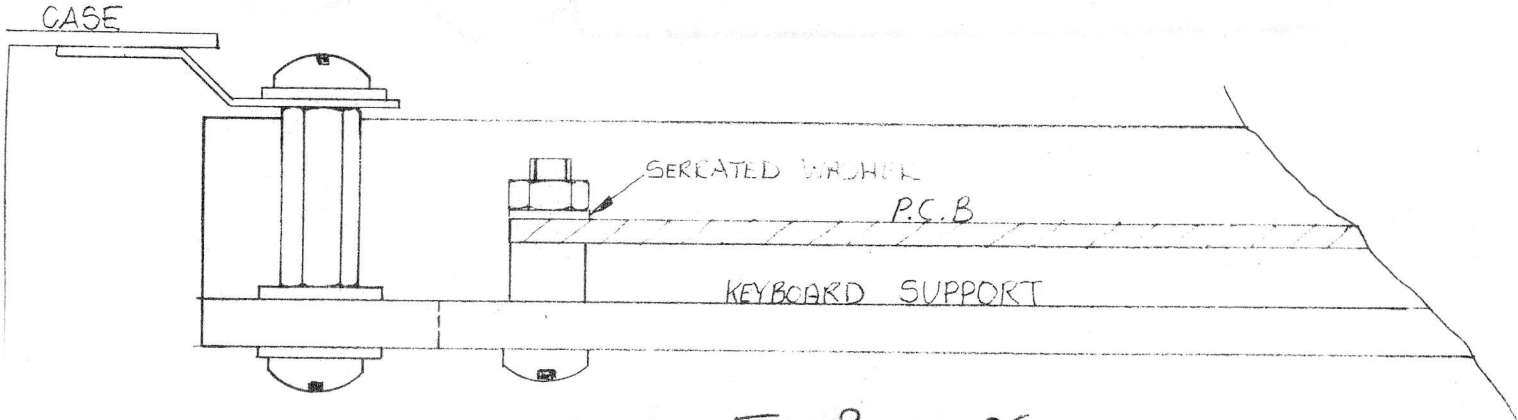




It is important that the switches fit squarely and evenly on the board. The best way to achieve this is to solder only one pin of the switch, fit the keytops and then whilst holding the board press in turn each switch whilst reheating the soldered joint. When all are evenly and squarely fitted the remaining pins can be soldered.

Solder on I.C. holders as shown in the diagram in the reprinted magazine article. Solder on resistors, capacitors and transistors in that order. Do not fit I.C.s yet. Fill each through plated hole with solder.

Attach the support bars as shown below using 4mm plain spacers and 3 x 12mm screws. The support bars are in turn attached to the case via 12.7mm threaded spacers and two 3 x 7.5mm screws.



KEYBOARD PARTS LIST

Resistors (all ¼w,5%)

R1,2,6,8,9	4k7
R3	18k
R4	12k
R5	680k
R7	100k

Capacitors

C1,2,5	220n 35V tantalum
C3,4	4u7 16V tantalum
C6	4n7 ceramic
C7	33p ceramic

Semiconductors

IC1	74LS123
IC2	74LS08
IC3	74LS157
IC4	2376
Q1,2	BC182L

Miscellaneous

PCB; 67 off momentary push-to-make switches; one off latching push-to-make switch; set of 67 double shot moulded key tops; IC sockets; switch mounting bracket; mounting pillars etc.

Fit keyboard to case with screws loose. Fit cover and ensure sufficient clearance around keys. Adjust as necessary. Remove cover and tighten threaded spacers to support bars in correct position. Remove keyboard by removing the top screws from the threaded spacers. The keyboard should now be easy to refit. Place keyboard in safe place.

POWER BOARD

1. Solder capacitors onto P.C.B. ensuring correct polarity.

2. Solder on resistors and plug pins.

3. Solder on diodes.

4. Bolt regulators to P.C.B. ensuring that the heat sinks are sandwiched between P.C.B. and regulators. Add a smear of silicone grease or heat sink compound between regulator and heat shunt. Solder power transistor flying leads to board.

PARTS LIST

POWER SUPPLY

Resistors (all ¼w, 5%)

R1, 2	10R
R3	100R
R4	4R7

Capacitors

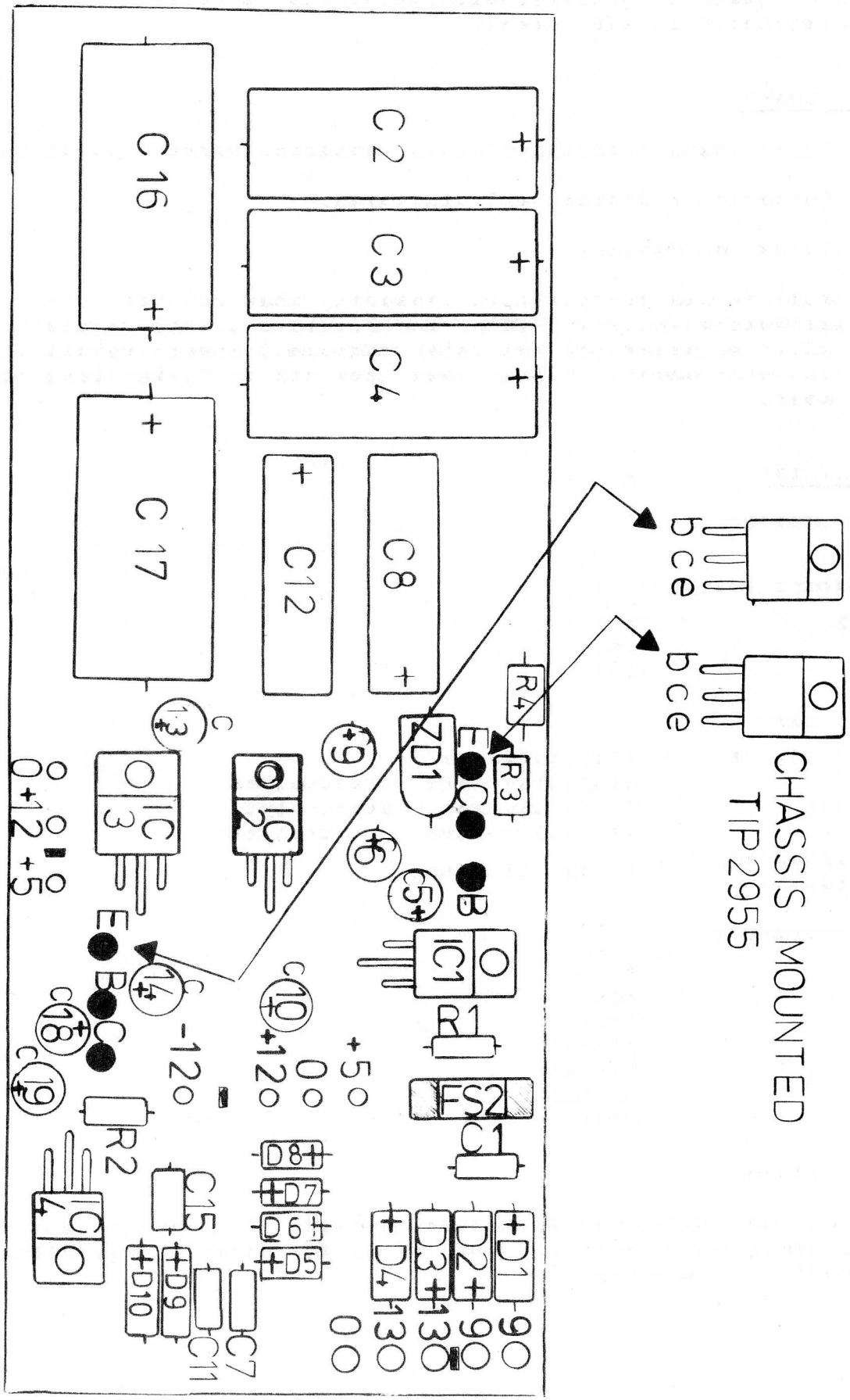
C1, 7, 11, 15	100n polyester
C2-4	4700u 16v axial electrolytic
C16, 17	4700u 25v axial electrolytic
C8, 12	2200u 25v axial electrolytic
C5, 6, 9, 10, 13, 14, 18, 19	1n 35v tantalum

Semiconductors

IC1	7805
IC2, 4	7812
IC3	7912
Q1, 2	TIP2955
D1-4, 9, 10	1N5402
D5-8	1N4002
ZD1	BCW70 5v6

Miscellaneous

PCB; one off three way connector; two off five way connectors; transformer (13.5-0-13.5 at 3A, 9V at 4A); fuse holder clips; four off TV5 heatsinks.

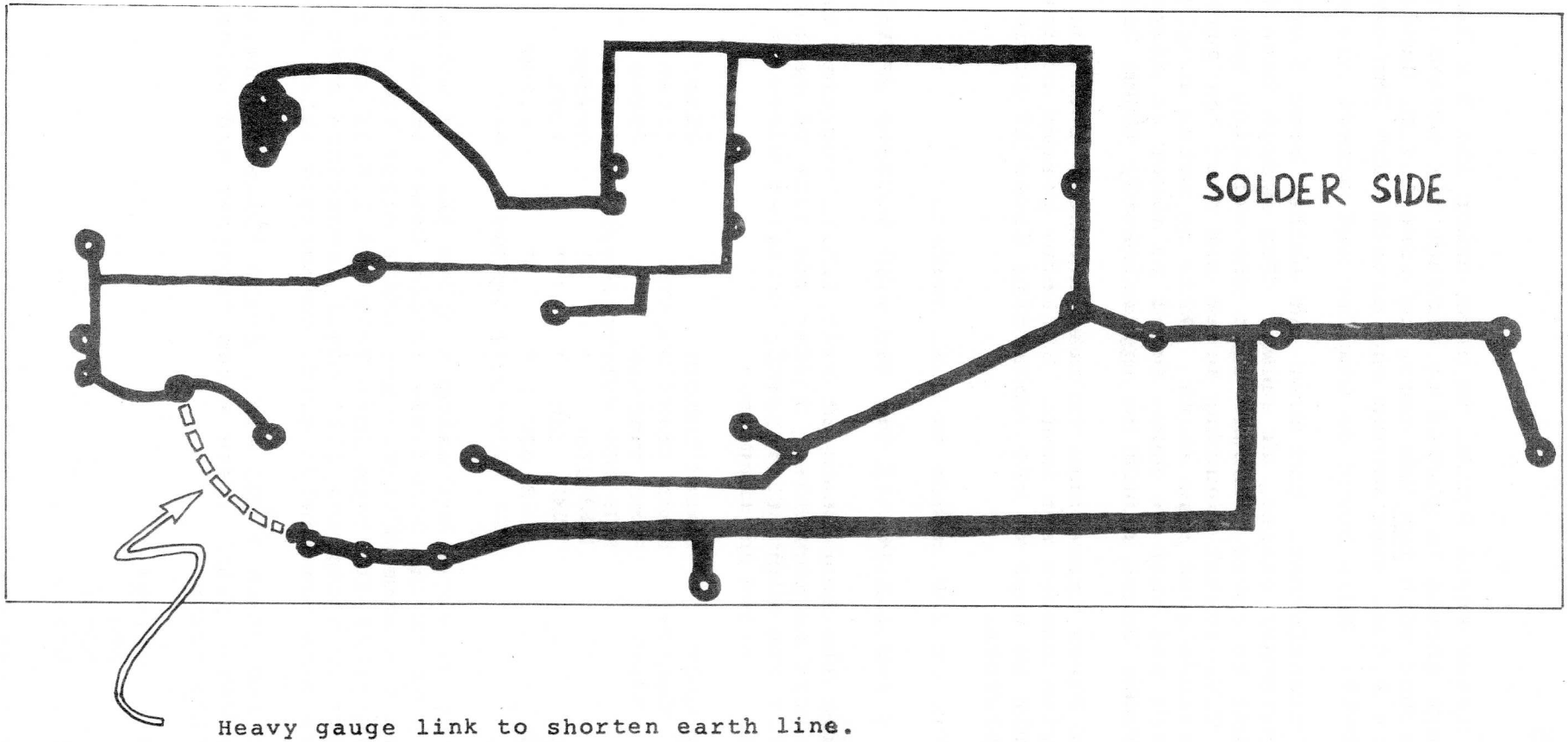


POWER SUPPLY BOARD

CORTEX II POWER SUPPLY BOARD

The tracks shown should be liberally tinned to reduce earth line voltage drop.

This ripple reduction eliminates bars on TV or monitor.



Heavy gauge link to shorten earth line.

CASE

Fit Power Board P.C.B. to case using six 3 x 16mm screws. These screws should be pushed up through the bottom of the case and attached with two 3mm nuts, to give P.C.B. sufficient clearance. Place P.C.B. over screws and attach with 3mm nuts and serrated washers. Main board is similar but screws are reversed.

Fit transformer. Cut down and solder onto 5 way plug the heavy transformer cables, as shown. Fit switch fuse holder and T.V. socket to cover. Ensure that the earthing tage is fitted under the T.V. socket mounting screw and that the paint is scraped away; to ensure good case earth. Wire up mains supply through the switch and fuse to Power Board as shown in diagram. Mains wire to Power Board should be approximately 400mm long.

Bolt Power transistor to case ensuring that the mica washer and plastic bushes are used. Silicone grease or heat sink compound should be used on all contacting faces of power transistors and regulators.

Solder on T.V. cable to T.V. socket.

Fit 4 red L.E.Ds and the two push buttons (P/Bs).

Solder the long leads of each L.E.D. together keeping the lead as short as possible. Solder one side of each P/B together. Solder the eight differently coloured wires to the L.E.Ds and P/Bs as shown on diagram:-

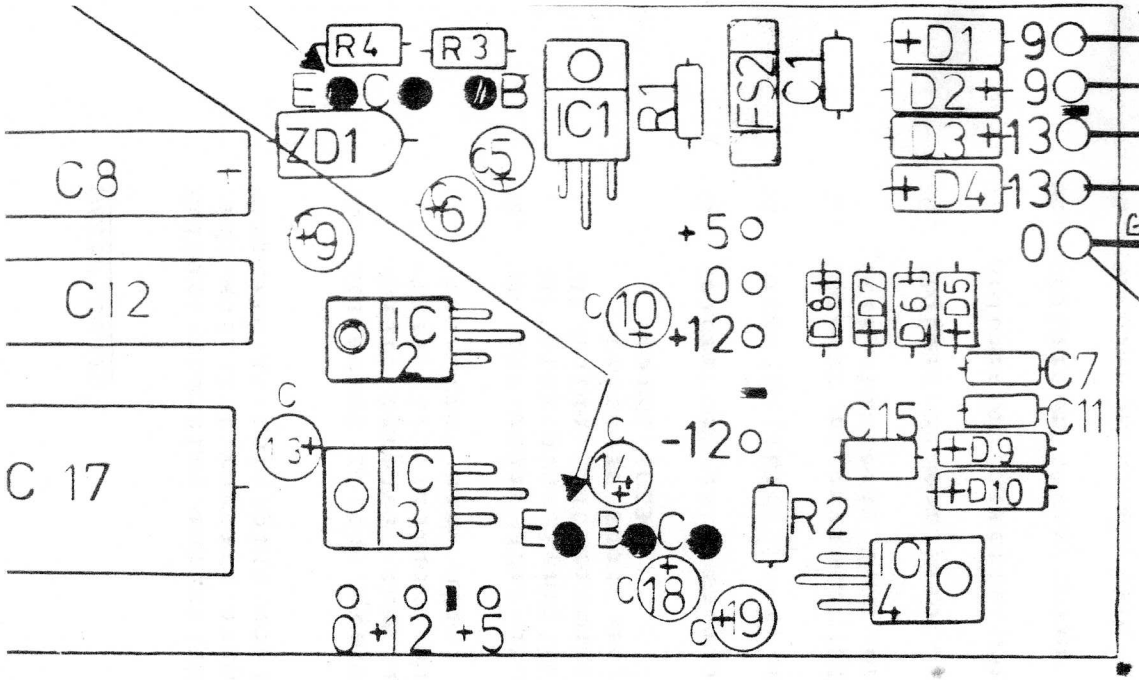
P/Bs Left	Reset button	Blue
" Centre	Warm start button	Green
" Right	Both buttons	Black
IAQ	RUN LED (short lead)	White
MAP	MAP LED "	Orange
TIME	TIME LED "	Pink
IDLE	IDLE LED "	Yellow
+5V	ALL LEDs(Long leads)	Red

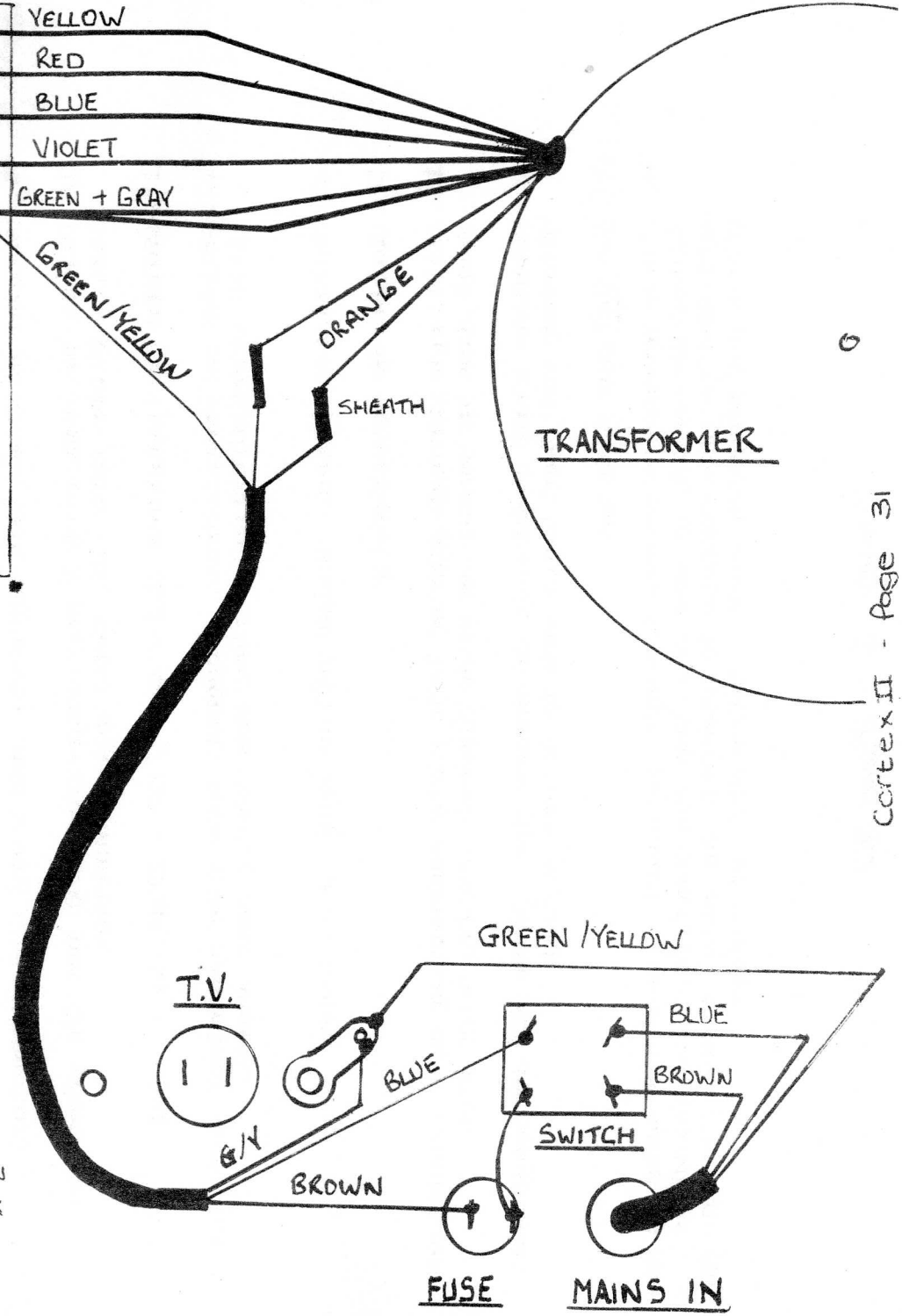
Gather up wires and using a nylon tie and adhesive holder attach the loom to the underside to the cover about 50mm from the L.E.Ds. Twist the loom slightly and add a nylon tie every 75mm. For the last 120mm split the loom into P/Bs and L.E.Ds and slide over sheathly. Solder or crimp the 0.1" shell connectors onto the end of each lead. Press into the shells until connectors click home.

Wire the other 5 way Power Board connector using the 350mm wires as shown. Twist these wires together and solder onto 4 way connectors in this order:-

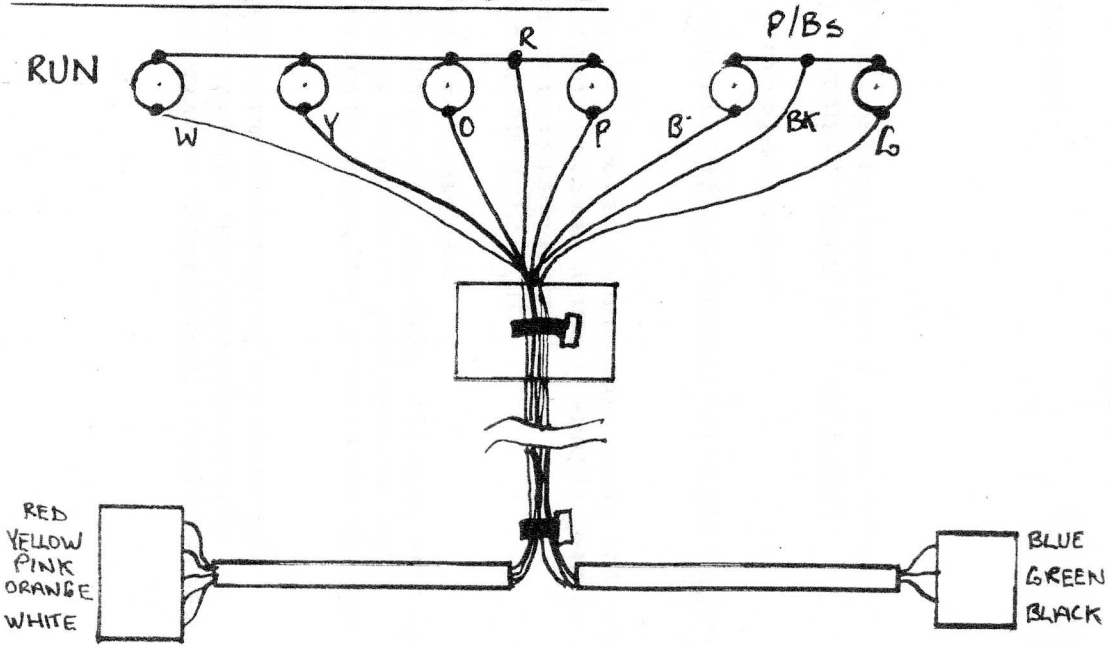
+12V	Orange
+5V	Red
0V	Black
-12V	Blue

Plug these leads onto Power Board. Solder the two 225mm orange leads to the loud speaker. Twist wires together and solder 0.1 connector onto to end. Push connector into 3 way shell using the outer two slots. Using the double-sided adhesive pads (3) stick the loudspeaker down to the case next to the transformer.





POWER SUPPLY BOARD



MAIN BOARD

Before starting ensure that you have a reasonably large, clean and well lit working area. Work slowly and methodically. A mistake at this pint can be very time consuming to rectify.

1. Refering to the magazine article note the I.Cs used in the basic computer they are marked with a B. Solder on all these I.C. holders ensuring that all the end cutouts are to the right hand side. Do not over solder.

2. On the under side of the board on I.C. 70:-
   A) Link pins 3 and 16 using insulated wire.
   B) Solder 1nf ceramic capacitor acress pins 1 and2, keeping leads as short as possible. Press capacitor flat against board. (C31).

3. Solder on all resistors. Solder C32 in parallel with R77.

4. Wire in links 1 - 4 using insulated wire.

5. Solder on crystals (XTAL 1 - 12MHz, XTAL 2 - 10.7MHz) Push the body flat against the board and wire into place.

6. Solder on modulator.

7. Solder on cassette port.

8. Solder on 4 way power connector.

9. Solder on 5 way LED connector and 3 way P/B connector.

10. Using the remaining 3 way connector clip out the short centre tag. Solder into the Piezo hole connector.

11. Solder on inductors (L1 = 4uH7, L2 = 22uH, L3 = 33uH)

12. Solder on capacitors. (Suggest make disc capacitor the last item to be soldered as they are easily bent over).

13. Solder on diodes. Broad yellow band is cathode.

14. **Solder on transistors**

15. **Fill each through plated hole (hole connecting only tracks on each side of board) on wide (power) tracks, with solder.**

16. **Examine board carefully to ensure that there are no components missing and that there are no solder errors.**

17. Do NOT put I.Cs in.

18. Place seven, M3 screws through the board, place two nuts on the underside to act as spacers from the chassis. The board is attached to the chassis by placing the seven screws through the slots and securing with a third nut and washer.

PART LIST

Main Board

Resistors (all ¼w, 5% except where stated)

R1,2	470R
R3-5,11,15 ,32,56,59	4K7
R6-8,20,21,28,41	330R
R9,12,13,39,40, 55,61,46,52 R10,14,45,47,58	10k
63,77	100R
R16-19	560R
R22	120k
R23,24,26,31,36	1k0
R25,29,33,68	2K7
R27	390R
R30	1k5
R34	680R
R35,60,74,75	2k2
R37	1k pre-set
R38,53,54	100K
R42	6k8
R43	3k9
R44	39k
R48-50	8k2
R51	1MO
R57	22k
R62	27k
R64-67,71	150R resistor array
R69	5K6
R70	12k
R72,73	3k3
R76	10R

Capacitors

C1	1n0 ceramic
C2,18	4u7 16V PCB electrolytic
C3,7,25,26	10u 16V PCB electrolytic
C4,5,6,9,10,13,17	100n polycarbonate
C8	47u 10V PCB electrolytic
C11,12,16	33p ceramic
C14	39p ceramic
C15,27	22u 16V PCB electrolytic
C19	100p ceramic
C20	22n polycarbonate
C21	4n7 polycarbonate
C22	330n polycarbonate
C24	5n6 polystyrene
C28	100u 16V PCB electrolytic
C29	330p ceramic
C30	470u 10V
C23	2n2 polycarbonate
C31	In ceramic
C32	10n ceramic
CV1	6-30p trimmer

Electrolytics of higher voltage rating may be used.

L1	4u7
L2	22u
L3	33u

Semicondutors

IC1,6,12,27,81	74LS04
IC2,17,18,61,69, 88	74LS74
IC3	74LS86
IC4,21,31	74LS00
IC5,22,30	74LS02
IC7,24	74LS10
IC8	TMS9911
IC9,10,84 98	74LS244
IC11	TMS9995
IC13,77	74LS08
IC15,34,35	74LS138
IC16,66,80,82,83	74LS07
IC19	74LS164
IC20,79	LM339
IC23	74LS20
IC25,65,78	74LS32
IC26	74LS612
IC28,29	74LS27
IC32	TMS4500
IC33,85	74LS139
IC36-43	TMS4164
IC44,97	74LS245
IC45-47	TMS2564
IC48	TMS9929
IC49-56	8118
IC57,58	4016B
IC59	LM1889
IC60	4013
IC62,63	74LS251
IC64	74LS259
IC67,68	TMS9902
IC70	74LS123
IC71	75189A
IC72	TL084
IC73	74LS73
IC74	75188
IC76	TMS9909
IC86	74LS297
IC87	74LS163

Q1,3,4	BC182
Q2,5	BC212
Q6	BC212
D1-5	1N4148

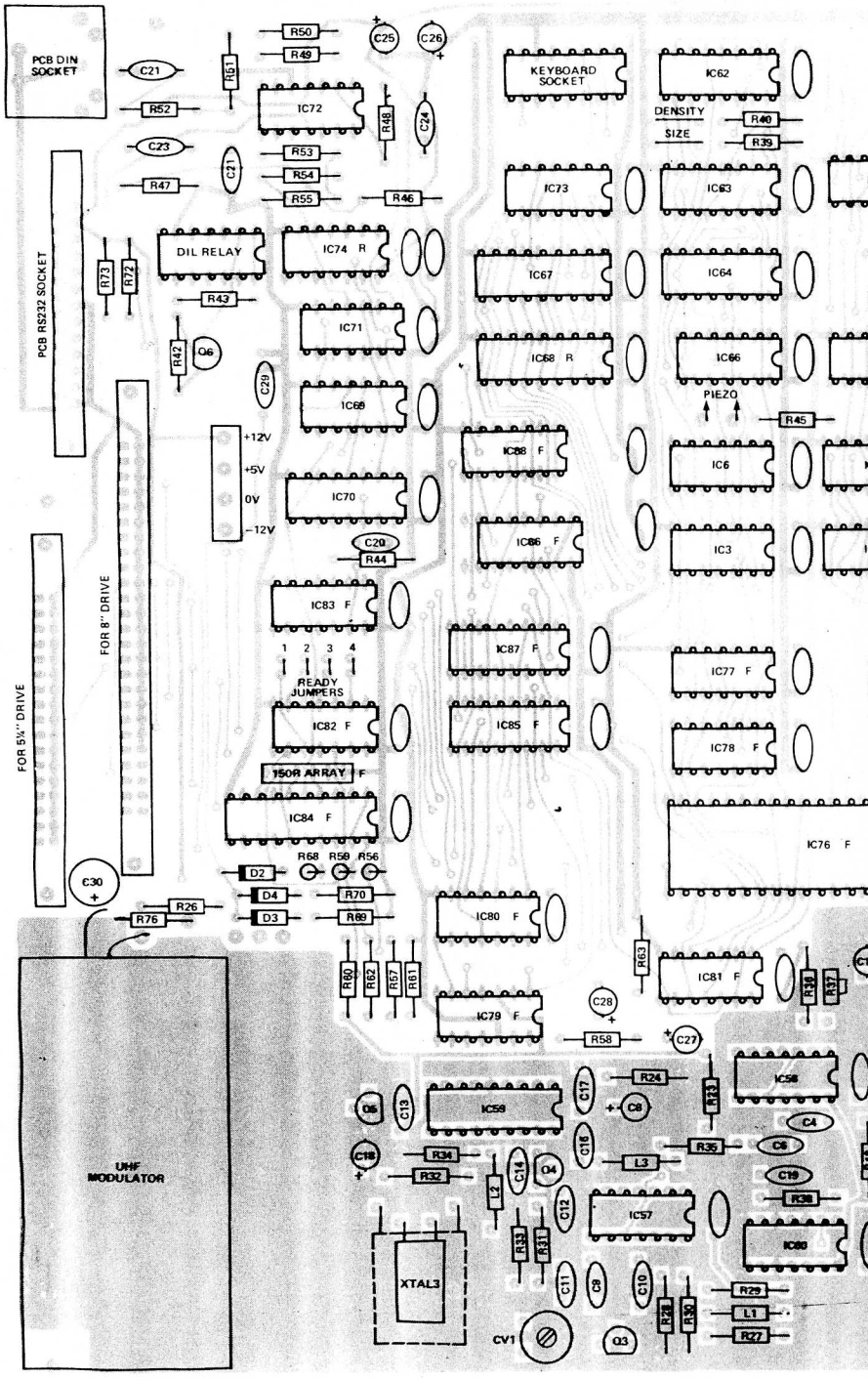
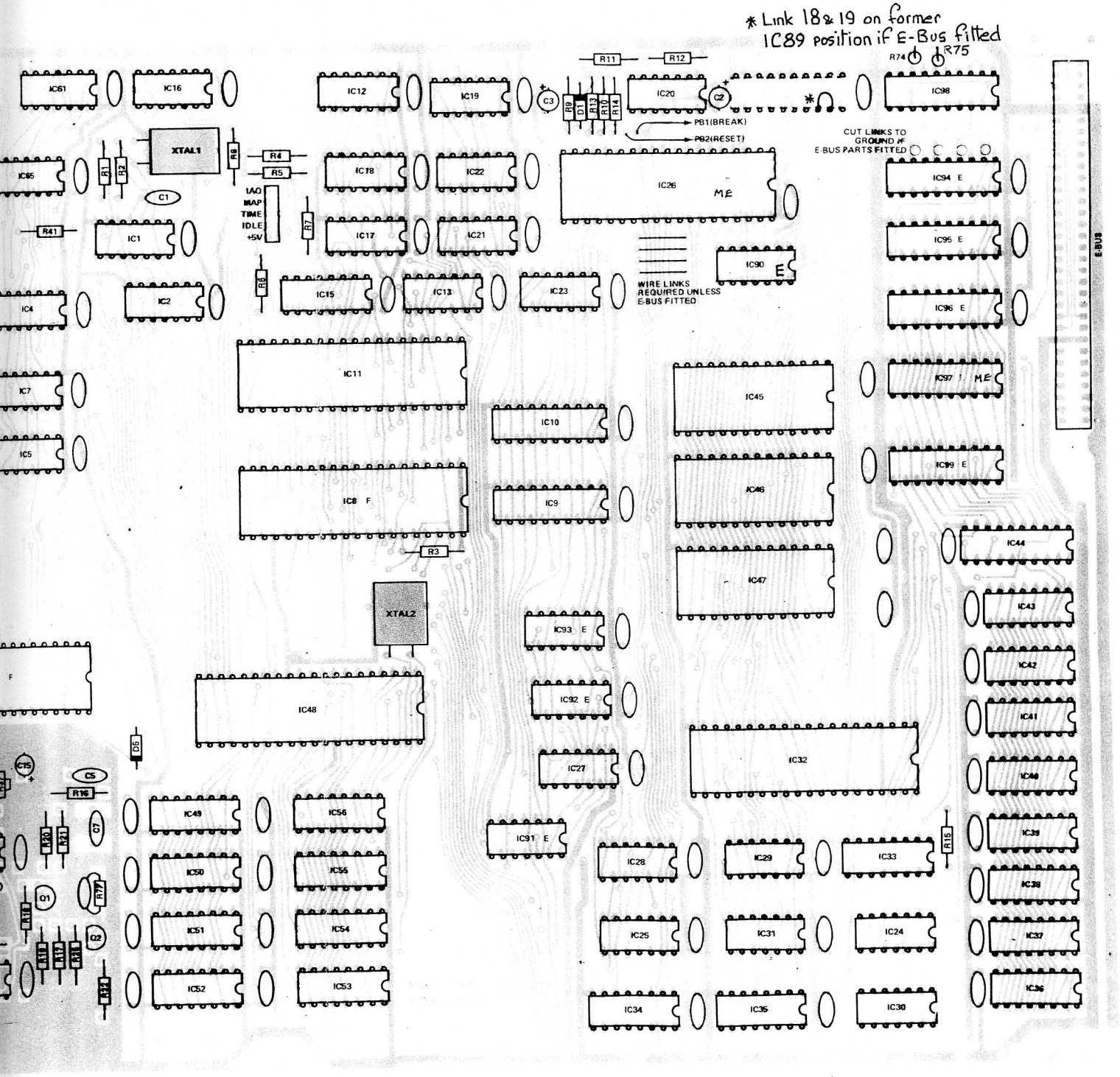


Fig. 4 Component overlay for the Cortex main board. The numerious unmarked unpolarised capacitors
ing capacitors and are 47n ceramic. The tracking shown is of the top foil of the earliest version and ma
boards (as may the patterns shown for the PSU and keyboard). ICs marked with various letters are for



itors are all supply line decoupl-
d may differ on later generation
e for expansion options and are

not supplied with the most basic version of the kit. The expansion options are as follows: R = RS232; F = floppy disc interface; E = E-BUS interface; ME = memory expansion

FINAL ASSEMBLY

Your machine should now be in the following condition:-

Case - with Power Board, transformer and speaker fitted.

Cover - attached to case by power transistor leads and mains lead.

Main Board - separate with no I.Cs fitted

Keyboard- " " " " "

Leaving main board and keyboard out, plug in and turn on machine. Using a Voltimeter check power supply output (+12,-12,+5). Switch off.

Place main board in case.

Plug in power supply, speaker, LEDs and P/Bs. Ensure correct orientation of each plug by reference to drawing of main board. Mark each plug and socket with a touch of paint to ensure correct future orientation.

Switch on. Check power supply output. Should be unchanged. Switch off. Disconnect power supply. Short power supply with 100R resistor.

Insert all I.Cs on main board and connect the power supply. Switch on. The **TIME** LED will flash about once per second. Check the supply voltages.

Connect a suitable T.V. receiver to the T.V. socket. Tune in receiver.

The television vertical hold control and the preset resistor on the main board may require slight adjustment for a steady display on wich will be written:-

```
CORTEX BASIC REV 1.1 C 1982
* Ready
```

If you seem to get nothing but a grey/blue screen, persevere with the adjustments (tuning, and preset resistor), you can do no damage by taking these to extremes.

When a steady display is achieved, switch off. Insert I.Cs into keyboard and connect to main board with cable provided.

Ensure caps lock is depressed. Switch on and type in the following:-

```
10   FOR Y = 0 TO 191
20   FOR C = 0 TO 15
30   COLOUR C,7
40   PLOT C*16, Y TO 7 + C*16,Y
50   NEXT C
60   NEXT Y

RUN
```

Adjust the tuner, preset resistor and preset capacitor for the best display.

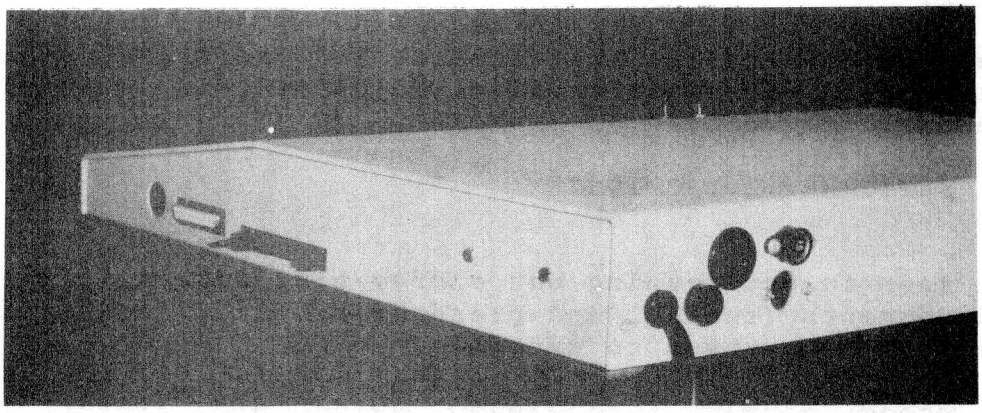
Switch off.

Attach keyboard to case. (The P/Bs connector may have to be bent over to avoid fouling).

Place cover on case and screw down.

FITTING THE RS232C INTERFACE

1. Carefully remove Main Board.
2. Solder on I.C. holders (I.C 74 & I.C 68). Referring to main board layout. I.Cs marked R.
3. Bolt right angled connector to board. Solder connector. Link connector pins 19 and 20.
4. Replace Main Board.
5. Fit I.Cs.
6. Test printer
7. Re-assemble machine.



CORTEX R.G.B. INTERFACE - ASSEMBLY INSTRUCTIONS

1. Assemble the components on the P.C. board, using the component list to identify the components.

2. Make up the connecting leads as shown in the diagram.

3. Remove the P.C. board from the Cortex.

   Secure the hexagonal threaded spacers in positions A and C.

   Solder in position D a five way P.C. mounting plug.

   Replace board in Cortex.

4. Plug the 5 pin connector on lead A on to the connector D on the Cortex P.C. board. The black wire goes to the pin nearest the transformer.

5. Mount the RGB board on the spacers, the top mounting screws go through the large holes on the edge of the board and into the spacers A and C.

6. Plug the 4 pin connector on lead A on to connector PL1, the black wire goes to pin 4.

7. Clip out the centre short pin on the three pin male connector. Solder to board in 12V - 0V position. Connect the red and black wires to the outside positions of the female part of the connector. Mark the two parts of the connectors to ensure correct orientation. Solder the other ends of the wires to the +12 and 0V lines on the power board.

8. Feed the 5 pin flat socket on lead B through the spare DIN socket hole on the rear panel of the Cortex, fix the DIN socket to the rear panel of the Cortex, and then plug the 5 pin flat socket on to PL2, the black wire goes to pin 1.

9. Make up a lead as in diagram C using the DIN plug supplied, to connect to the LINEAR input to your monitor.

10. Test using the last program on page 37 of the Cortex Construction Manual.

Note Some boards have errors in the component overlay:

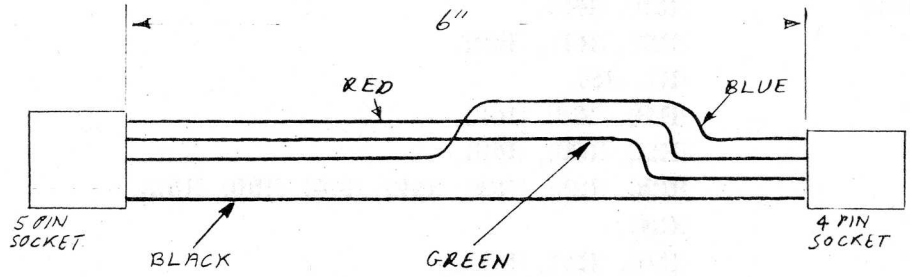
1. C19 should have its negative leg nearest the edge of the board.

2. TR13 should be mounted with the same orientation as TR12 - do not bend the legs.

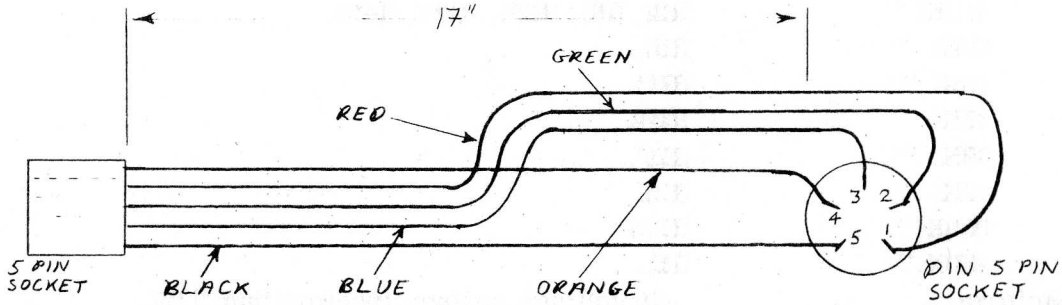
A
B
C
D
PIEZO
IC76
IC77
IC78
IC79
IC80
IC81
IC82
IC83
IC84
IC85
IC86
IC87
IC88
IC89
IC90
IC91
IC92
IC93
IC94
IC95
IC96
IC97
IC98
IC99
IC100
IC101
IC102
IC103
IC104
IC105
IC106
IC107
IC108
IC109
IC110
IC111
IC112
IC113
IC114
IC115
IC116
IC117
IC118
IC119
IC120
IC121
IC122
IC123
IC124
IC125
IC126
IC127
IC128
IC129
IC130
IC131
IC132
IC133
IC134
IC135
IC136
IC137
IC138
IC139
IC140
IC141
IC142
IC143
IC144
IC145
IC146
IC147
IC148
IC149
IC150
IC151
IC152
IC153
IC154
IC155
IC156
IC157
IC158
IC159
IC160
IC161
IC162
IC163
IC164
IC165
IC166
IC167
IC168
IC169
IC170
IC171
IC172
IC173
IC174
IC175
IC176
IC177
IC178
IC179
IC180
IC181
IC182
IC183
IC184
IC185
IC186
IC187
IC188
IC189
IC190
IC191
IC192
IC193
IC194
IC195
IC196
IC197
IC198
IC199
IC200
IC201
IC202
IC203
IC204
IC205
IC206
IC207
IC208
IC209
IC210
IC211
IC212
IC213
IC214
IC215
IC216
IC217
IC218
IC219
IC220
IC221
IC222
IC223
IC224
IC225
IC226
IC227
IC228
IC229
IC230
IC231
IC232
IC233
IC234
IC235
IC236
IC237
IC238
IC239
IC240
IC241
IC242
IC243
IC244
IC245
IC246
IC247
IC248
IC249
IC250
IC251
IC252
IC253
IC254
IC255
IC256
IC257
IC258
IC259
IC260
IC261
IC262
IC263
IC264
IC265
IC266
IC267
IC268
IC269
IC270
IC271
IC272
IC273
IC274
IC275
IC276
IC277
IC278
IC279
IC280
IC281
IC282
IC283
IC284
IC285
IC286
IC287
IC288
IC289
IC290
IC291
IC292
IC293
IC294
IC295
IC296
IC297
IC298
IC299
IC300
IC301
IC302
IC303
IC304
IC305
IC306
IC307
IC308
IC309
IC310
IC311
IC312
IC313
IC314
IC315
IC316
IC317
IC318
IC319
IC320
IC321
IC322
IC323
IC324
IC325
IC326
IC327
IC328
IC329
IC330
IC331
IC332
IC333
IC334
IC335
IC336
IC337
IC338
IC339
IC340
IC341
IC342
IC343
IC344
IC345
IC346
IC347
IC348
IC349
IC350
IC351
IC352
IC353
IC354
IC355
IC356
IC357
IC358
IC359
IC360
IC361
IC362
IC363
IC364
IC365
IC366
IC367
IC368
IC369
IC370
IC371
IC372
IC373
IC374
IC375
IC376
IC377
IC378
IC379
IC380
IC381
IC382
IC383
IC384
IC385
IC386
IC387
IC388
IC389
IC390
IC391
IC392
IC393
IC394
IC395
IC396
IC397
IC398
IC399
IC400
IC401
IC402
IC403
IC404
IC405
IC406
IC407
IC408
IC409
IC410
IC411
IC412
IC413
IC414
IC415
IC416
IC417
IC418
IC419
IC420
IC421
IC422
IC423
IC424
IC425
IC426
IC427
IC428
IC429
IC430
IC431
IC432
IC433
IC434
IC435
IC436
IC437
IC438
IC439
IC440
IC441
IC442
IC443
IC444
IC445
IC446
IC447
IC448
IC449
IC450
IC451
IC452
IC453
IC454
IC455
IC456
IC457
IC458
IC459
IC460
IC461
IC462
IC463
IC464
IC465
IC466
IC467
IC468
IC469
IC470
IC471
IC472
IC473
IC474
IC475
IC476
IC477
IC478
IC479
IC480
IC481
IC482
IC483
IC484
IC485
IC486
IC487
IC488
IC489
IC490
IC491
IC492
IC493
IC494
IC495
IC496
IC497
IC498
IC499
IC500
IC501
IC502
IC503
IC504
IC505
IC506
IC507
IC508
IC509
IC510
IC511
IC512
IC513
IC514
IC515
IC516
IC517
IC518
IC519
IC520
IC521
IC522
IC523
IC524
IC525
IC526
IC527
IC528
IC529
IC530
IC531
IC532
IC533
IC534
IC535
IC536
IC537
IC538
IC539
IC540
IC541
IC542
IC543
IC544
IC545
IC546
IC547
IC548
IC549
IC550
IC551
IC552
IC553
IC554
IC555
IC556
IC557
IC558
IC559
IC560
IC561
IC562
IC563
IC564
IC565
IC566
IC567
IC568
IC569
IC570
IC571
IC572
IC573
IC574
IC575
IC576
IC577
IC578
IC579
IC580
IC581
IC582
IC583
IC584
IC585
IC586
IC587
IC588
IC589
IC590
IC591
IC592
IC593
IC594
IC595
IC596
IC597
IC598
IC599
IC600
IC601
IC602
IC603
IC604
IC605
IC606
IC607
IC608
IC609
IC610
IC611
IC612
IC613
IC614
IC615
IC616
IC617
IC618
IC619
IC620
IC621
IC622
IC623
IC624
IC625
IC626
IC627
IC628
IC629
IC630
IC631
IC632
IC633
IC634
IC635
IC636
IC637
IC638
IC639
IC640
IC641
IC642
IC643
IC644
IC645
IC646
IC647
IC648
IC649
IC650
IC651
IC652
IC653
IC654
IC655
IC656
IC657
IC658
IC659
IC660
IC661
IC662
IC663
IC664
IC665
IC666
IC667
IC668
IC669
IC670
IC671
IC672
IC673
IC674
IC675
IC676
IC677
IC678
IC679
IC680
IC681
IC682
IC683
IC684
IC685
IC686
IC687
IC688
IC689
IC690
IC691
IC692
IC693
IC694
IC695
IC696
IC697
IC698
IC699
IC700
IC701
IC702
IC703
IC704
IC705
IC706
IC707
IC708
IC709
IC710
IC711
IC712
IC713
IC714
IC715
IC716
IC717
IC718
IC719
IC720
IC721
IC722
IC723
IC724
IC725
IC726
IC727
IC728
IC729
IC730
IC731
IC732
IC733
IC734
IC735
IC736
IC737
IC738
IC739
IC740
IC741
IC742
IC743
IC744
IC745
IC746
IC747
IC748
IC749
IC750
IC751
IC752
IC753
IC754
IC755
IC756
IC757
IC758
IC759
IC760
IC761
IC762
IC763
IC764
IC765
IC766
IC767
IC768
IC769
IC770
IC771
IC772
IC773
IC774
IC775
IC776
IC777
IC778
IC779
IC780
IC781
IC782
IC783
IC784
IC785
IC786
IC787
IC788
IC789
IC790
IC791
IC792
IC793
IC794
IC795
IC796
IC797
IC798
IC799
IC800
IC801
IC802
IC803
IC804
IC805
IC806
IC807
IC808
IC809
IC810
IC811
IC812
IC813
IC814
IC815
IC816
IC817
IC818
IC819
IC820
IC821
IC822
IC823
IC824
IC825
IC826
IC827
IC828
IC829
IC830
IC831
IC832
IC833
IC834
IC835
IC836
IC837
IC838
IC839
IC840
IC841
IC842
IC843
IC844
IC845
IC846
IC847
IC848
IC849
IC850
IC851
IC852
IC853
IC854
IC855
IC856
IC857
IC858
IC859
IC860
IC861
IC862
IC863
IC864
IC865
IC866
IC867
IC868
IC869
IC870
IC871
IC872
IC873
IC874
IC875
IC876
IC877
IC878
IC879
IC880
IC881
IC882
IC883
IC884
IC885
IC886
IC887
IC888
IC889
IC890
IC891
IC892
IC893
IC894
IC895
IC896
IC897
IC898
IC899
IC900
IC901
IC902
IC903
IC904
IC905
IC906
IC907
IC908
IC909
IC910
IC911
IC912
IC913
IC914
IC915
IC916
IC917
IC918
IC919
IC920
IC921
IC922
IC923
IC924
IC925
IC926
IC927
IC928
IC929
IC930
IC931
IC932
IC933
IC934
IC935
IC936
IC937
IC938
IC939
IC940
IC941
IC942
IC943
IC944
IC945
IC946
IC947
IC948
IC949
IC950
IC951
IC952
IC953
IC954
IC955
IC956
IC957
IC958
IC959
IC960
IC961
IC962
IC963
IC964
IC965
IC966
IC967
IC968
IC969
IC970
IC971
IC972
IC973
IC974
IC975
IC976
IC977
IC978
IC979
IC980
IC981
IC982
IC983
IC984
IC985
IC986
IC987
IC988
IC989
IC990
IC991
IC992
IC993
IC994
IC995
IC996
IC997
IC998
IC999
IC1000

THIRD ANGLE PROJECTION

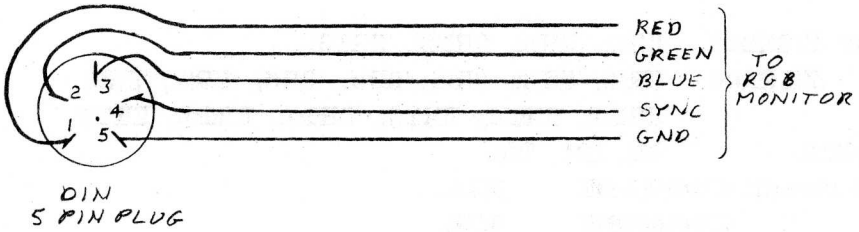
MANUFACTURE TO BE IN ACCORDANCE WITH FEEDBACK WORKSHOP / ASSEMBLY / TEST CODE OF PRACTICE



LEAD A



LEAD B



LEAD C

ISS	DATE	SIG	BNo	CH	APP

EXCEPT WHERE STATED ALL DIMENSIONS IN mm TOLERANCES
FROM 0 TO 10 ± 0,1
OVER 10 TO 50 ± 0,2
OVER 50 TO 250 ± 0,3
ANGLES ±

MATL

FINISH

FEEDBACK LIMITED

TITLE POWERTRAN
CORTEX RGB INTERFACE

DR	DATE	SCALE
CH	4	
APP		

CORTEX R.G.B. INTERFACE.

PCB COMPONENTS.

Resistors: All resistors are 0.125W, 5%.

22 ohms	R20, R45.
82 "	R27, R41, R52.
150 "	R7, R8.
220 "	R18, R33, R43.
330 "	R22, R36, R46.
390 "	R25, R28, R39, R42, R44, R50, R53.
470 "	R34.
560 "	R16, R17, R47.
680 "	R10, R19, R21.
1K "	R4, R9, R13, R26, R35, R40, R51.
2.2K "	R29, R30, R31, R32.
2.7K "	R23, R37, R48.
8.2K "	R2, R6, R24, R38, R49.
10K "	R5.
18K "	R12.
27K "	R14.
39K "	R11.
47K "	R3.
100K "	R15.
470K "	R1.

Capacitors: All voltage ratings greater than 12V.

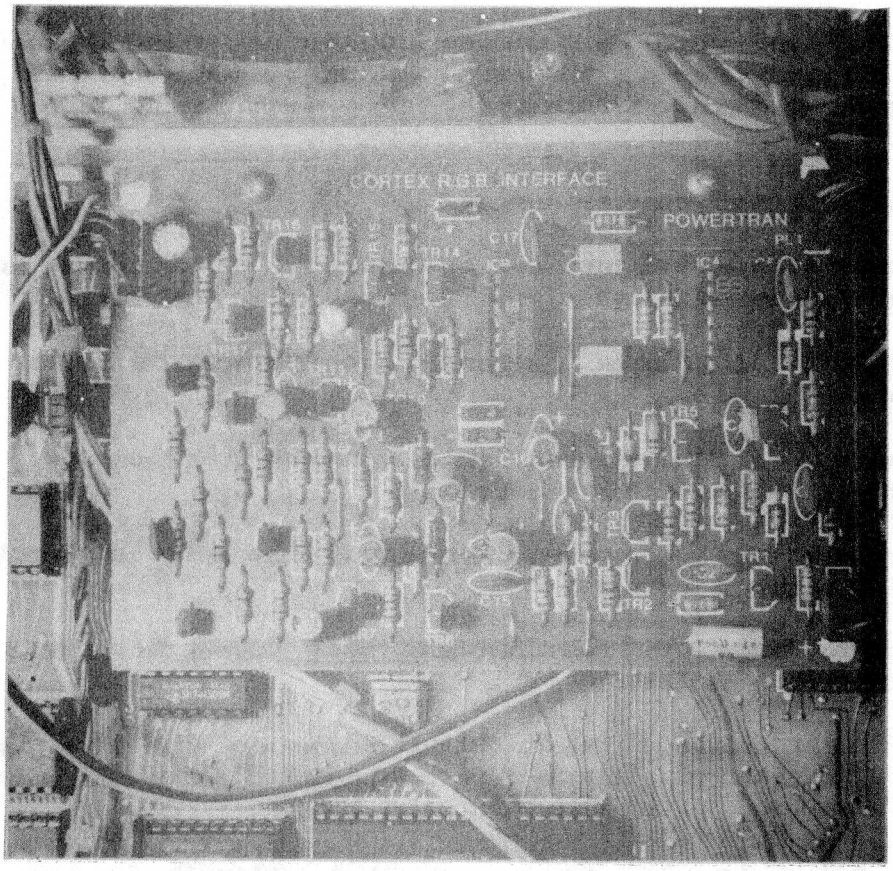
47nF Ceramic disc.	C1, C15, C17, C18, C20.
100pF " "	C2.
47pF " "	C3.
220pF " "	C4, C5.
470nF Polyester.	C6, C7.
10uF Electrolytic.	C8, C9, C10, C11, C12, C13, C14.
47uF Electrolytic.	C16, C19.

Semiconductors:

Transistor ZTX213	TR1, TR8, TR12, TR16.
" ZTX108	TR2, TR3, TR4, TR5, TR6, TR7, TR9, TR10, TR11, TR13, TR14, TR15, TR17.
Diode BAX13	D1, D2, D3.
Integrated circuit CD4011BE	ICA.
" " CD4066BE	ICB.
Inductor 47uH, 0.35A.	L1.
Connector, Male, 4 way.	PL1.
" " 5 way.	PL2.
Terminal pins, 1mm.	2 off.
Printed circuit board.	1 off.

OTHER COMPONENTS.

DIN Plug, 5 pin, 240°.	1 off.
DIN Socket, 5 pin, 240°.	1 off.
Connector, Male, 5 way.	1 off.
Connector shell, 5 way.	2 off.
Connector shell, 4 way.	1 off.
Crimp terminal, Female.	14 off.
Board spacer.	3 off.



Disc Drive

1. Carefully remove the main board.

2. Solder on the 34 way right angled socket.

3. Connect pin 32 on the 5" connector to pin 14 on the 8" connector. You need only do this if you intend to use double sided drives.

4. Solder the IC sockets onto the main board. Insert the relevant IC's ensuring correct position.

5. Fasten the socket of the three way mains connector, through the hole at the rear of the cover. Tighten. Connect this socket to the 3 pin plug on the Power Supply board. This is to power the disc drives.

6. Install the main board and fasten cover.

7. Switch on Cortex and check for normal operation. If it does not power up check for solder splashes and incorrectly inserted IC's.

8. Make up the power cable as shown. Connect to computer and test for correct voltages on pins.

9. Take the two piece metal cabinet and stick on the six 'sticky feet'. This avoids the disc drive warping.

10. Screw the ribbon cable clamp in position under the edge connector, using two M3 X 12 pan head screws.

11. Screw the disc drives into position on the base using four M3 X 8 pan head screws for each drive.

12. Clip the 34 way ribbon cable to each drive ensuring that the red conductor goes to terminal 1. Push the cable down onto the clamp base and clip the top on, thus clamping the cable.

13. Connect the power cable to each drive and exit through the half round notch with the tie inside the case.

14. To configure the drives for i and 2:

    a. If only one drive is used the terminator block (14 pin DIL package marked 150) should be left in place. The jumper link should be in position marked 0 or DS0. The jumper position are next to terminal 1 on the ribbon cable connector.

    b. For two drives the left hand drive (as you look down) is number 0 so the jumper link should be installed in this position and the terminator block removed.

       The right hand drive should have the link in position marked 1 and the terminator block should be left in position.

15. Screw down the top cover of the disc drive enclosure.

16. Place the assembly on top of the Cortex cabinet. Plug the 34 way and power cables into the Cortex.

17. Switch on. The normal power up screen should appear. If not check for correct connection of power supply and 34 way cable.

18. Follow instructions accompanying software.

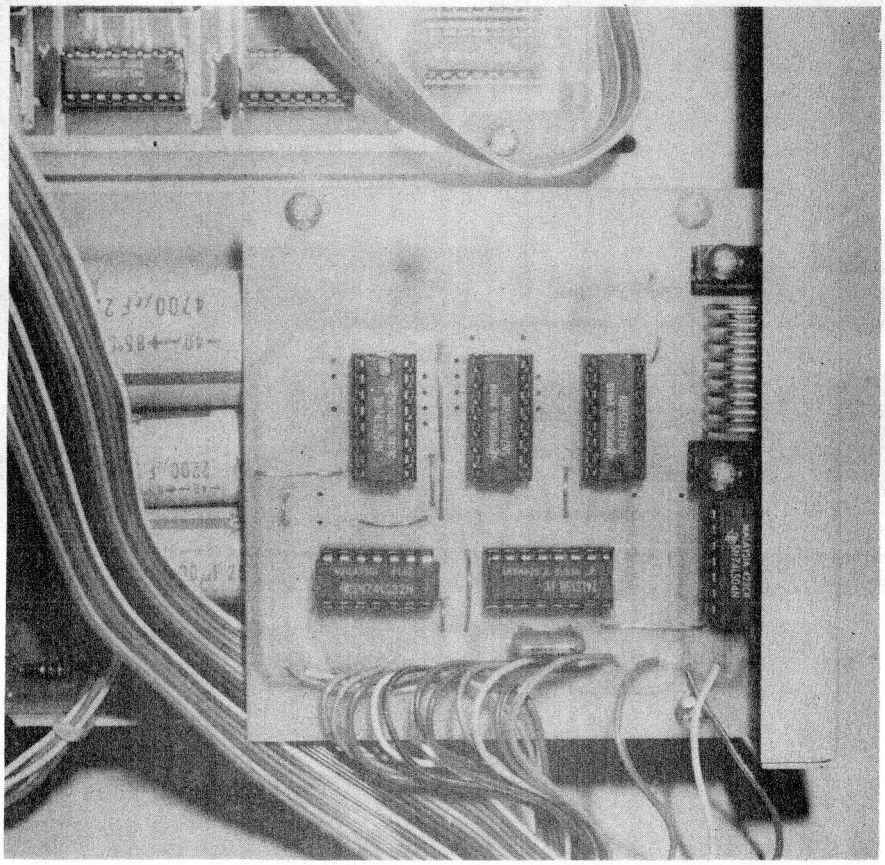
Centronics Construction and Installation

Construct the Centronics board as per the magazine article.

The board is designed to mount over the power supply P.C.B on the CORTEX II using the Hex pillars and screws provided. Remove three of the fixing nuts on the power P.C.B. Replace these with the hex spacers and tighten. The Centronics board mounts over these pillars and is screwed into place using the pan head screws.

If you have an old type Cortex it is suggested that the board is fixed, upside down to the top cover of the case between the disc drives, using double sided sticky pads provided and ensuring that the solder side of the board does not touch the case! Connection in this case must be made between the printer and board by flying lead. Ensuring that the flying lead cannot pull the P.C.B away from its fixing.

The Centronics P.C.B is connected to the main board using the 17-way ribbon cable provided. Care must be taken when soldering to the main board that no two lines are shorted together.



FAULT FINDING

1, 2, 3, 4 - Main Processor

5 - Video

6 - Keyboard

7 - Cassette Interface

8 - Disc Drives

If you are reading this section your Cortex has failed to work. At this stage it is assumed that the **TIME** LED is not flashing. If it is go on to section 5. It is well worth checking the values and orientation of all components at this point. Also check both sides of the board for solder splashes and clippings from components. A common source of error is an IC leg outside the holder! Use a bright light to check the board. Use a recommended flux solvent to thoroughly clean the board.

If you still cannot find any construction errors then proceed as follows: Obtain a LOGIC PROBE (Approximately £20). Preferably TTL. If you already have an oscilliscope you will not need this instrument. Normally a LOGIC PROBE has two LEDs which indicate the status of a particular pin when the probe tip is applied to the pin in operation. The probe should give 4 logic states.

LEDS	H	L
HIGH	1	0
LOW	0	1
NOT CONNECTED	0	0
PULSING	1	1

The probe is extremely useful for any digital work.

The four LEDs on the front cover give a good idea of the likely problem. These LEDs should go on in the following sequence.

	RUN	IDLE	**MAP**	**TIME**	
A. about 0.25SEC	0	0	0	1	
B. about 1 SEC	1	0	0	1	
C.	0	1	0	1	FLASHING APPROX. 1Hz

1. If RUN does not come on at all check RESET signal (Pin 1 on IC20) should be HIGH. Press the reset button should see signal go LOW for as long as the button is depressed. Recheck this on pin 22 of IC11. Should have same results.

2. If RUN LED stays on for approximately 1sec and goes out and IDLE does not come on (and possibly a continuous BEEP) check the EPROMS (IC45, 46, 47) for a leg outside the holder or a bad connection.

3. If RUN LED comes on and stays on. Then remove IC32 and IC44.

   a) Check READY (Pin 23 on IC11) should be high. If low then check EXT CRU (Pin 6 IC28) should be 0. If not check for a short in data lines (See Sec G.)

   b) Check address lines on EPROMS (ICs 45, 46, 47) all should by cycling immediately after RESET. If one line is not then check for short.

   c) Check DATA lines on EPROMS, these should all be cycling (Pins 11 - 19). If one is not then check for short.

   d) Chip select 2 (CS2 Pin 27) on each EPROM should pulse at least once after RESET.

   e) Check each EPROM (Pin 22) this should be initially low after RESET. After 1 sec should go high if it does then go to section 4.

   f) If remaining low remove power from board and using a Multi-meter on 1K Ω range check for 0 Ω between address lines. Note: to confirm short circuit reverse meter probes and recheck.

   g) Repeat for data lines. Using low Ω range. If your Multi-Meter has a high voltage battery (>15v) then for safety remove data ICs i.e. 11, 15, 28, 45, 46, 47 and 48.

4. Turn power off. Remove IC36 - 43. Check for short circuit between Memory Address lines (MA) (Pins 13, 14, 19, 21, 24, 27, 32, 33 with Multimeter) Check for short circuit between Pin 2 on each RAM chip.

   If the above check out then suspect a component fault.

   If state C on LEDs is reached then majority of LOGIC is working.

5. Connect TV to channel 36.
   Should show:-

   CORTEX BASIC Rev 1.1 © 1982
   * READY

   If you get a blank screen and no sound, adjust R37 (preset). If still no picture then put logic probe on SYNC (Top of R25) If pulses exist then check all component values, polarity and soldering. If no pulses remove D5 and recheck. If pulses occur replace D5 in the correct orientation.

   If a black and white picture exists rotate CV1 through 360° slowly and check for colour.

   If still no colour change R22 from 120K to 220-330K.

   If still no colour replace R38 with 220K preset and adjust as necessary.

6. If graphical characters appear rather than alpha-numeric then check presence of Q2 and that pin 10 of DIL connector is low.

   On power-up if the keyboard does not respond then it is possible that there is a short in the keyboard array. This is in effect like a button being continuously depressed.

   To find the short simply press the REPEAT key. This should tell you which tracks are shorted.

   With no keys depressed then Pin 8 - 15 of IC4 should be cycling. When a key is depressed these pins should freeze. If not cycling check key array for short.

7. Ensure that pin 3 of IC70 is taken to 5v and a 1nf capacitor is soldered between pins 1 and 2. (IC70) on the underside of the board. The following program will test the hardware. Type in the following program:-

```
1Ø  BASE Ø18ØH
2Ø  CRB (16) = 1
3Ø  CRB (17) = 1
4Ø  WAIT 5Ø
5Ø  CRB (17) = Ø
6Ø  WAIT 5Ø
7Ø  GOTO 3Ø
```

   Plug in cassette and start recording

   Type RUN

   After 30 seconds press ESCAPE.

   Rewind cassette and playback.

   Look with a logic probe on Pin 3 of IC67. This should be toggling between 1 and 0 every 0.5 sec.

   Adjust volume to give a clean toggle with no jitter.

   If no success ensure that the cassette has correctly recorded the test signal generated in the program, listen to the cassette. The recorded sound should be alternating high and then low every 0.5 sec. If not check that whilst recording pin 3 of IC67 is toggling. If the pin is low then the cassette recorder is not connected. To check the working of the cassette interface place the probe on pin 2 of IC67 whilst the test program is running. This should be toggling.

   If no toggle replace R44 with 100K preset adjust as necessary.

   If still no success try using a computer compatible recorder. Avoid Dolby.

8. Assuming that 5¼" Disc Drives are being used as supplied. It has been known for some domestic TVs to affect the drives. Move TV away. READY jumper diodes should not be connected.

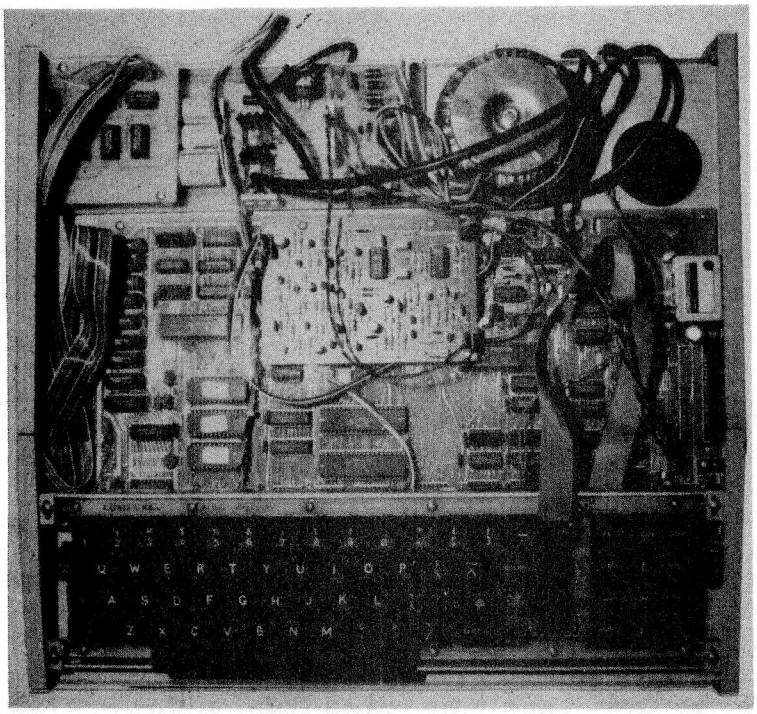
   Obtain a 10MHz oscilliscope.

   Put in BOOT disc.

Type BOOT. At this point the drives try and read data from track zero. If it fails then the read/write head moves in 3 tracks and then out 3 tracks and tries again. You should hear the head oscillating when you type BOOT on the disc drive with its LED illuminated. Monitor signal on Pin 5 of IC70. This should be 2 µ sec wide 50% duty cycle. Adjust R70 to being within 5%. Note: a blank or corrupted disc will not give 50% duty cycle. If a very small duty cycle check that C29 is connected to Pin 7 of IC70.

Check RKM (Pin 26 of IC76) signal is phase locked with above signal (there is some jitter).

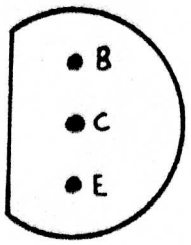
Check that pin 5 of IC76 goes low periodically. This shows track zero has been reached.



CORTEX II POINTS ARISING 23.1.1985

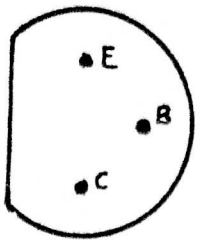
Please note:

The transistors supplied in some kits for the main board are BC182L and BC212L instead of BC182 and BC212. They are electrically identical but the pin-outs are different as shown below. The leads require bending to ensure they go into the correct holes in the P.C.B.



BC182L
BC212L

Pin View



BC182
BC212