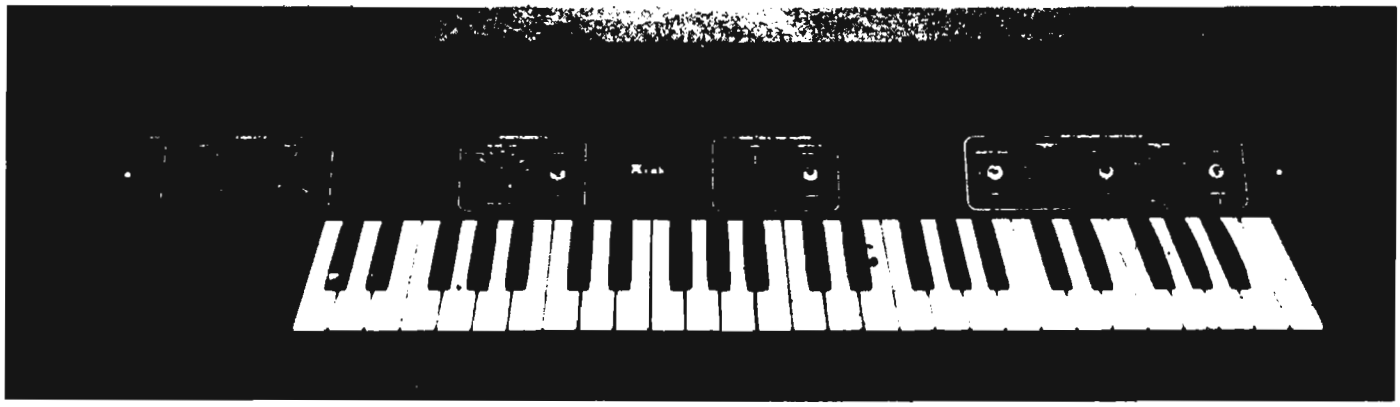
POLYPHONIC KEYBOARD CONTROLLER

Tired of playing one note at a time on a boring old monophonic synthesizer? In this design Tim Orr describes how you can build a four octave polyphonic keyboard controller incorporating first note priority.



THE MUSIC synthesizer is probably the most powerful musical instrument of today, and it will most probably form the basis of the next generation of keyboard instruments. However, the synthesizer suffers from one major drawback due to its unique structure. The disadvantage is that it is a monophonic instrument as opposed to traditional keyboard instruments, such as organs and piano's which are polyphonic, or multi-voiced. A brief resumé of synthesizer structure should clarify the reasons behind this.

To start with, the synthesizer is composed of a set of modules or independent circuit packages whose parameters in most cases are voltage controllable. For instance, a voltage controlled oscillator (VCO) has an output frequency (pitch) which is dependant on the magnitude of the input control voltage. These modules can be split up into three distinct groups. Firstly there are Sources, such as:

1. Noise
2. Voltage controlled oscillators

Secondly there are Modifiers which form by far the largest group:

1. Voltage controlled filters (VCF's)
2. Voltage controllled amplifiers (VCA's)
3. Ring modulators
4. Filter banks or graphic equalisers
5. Phase shifters
6. Reverbration

Thirdly there are control voltage sources:

1. Sample and holds
2. Sequencers
3. Transient generators
4. Trigger delays
5. Keyboard controllers

Getting Your Priorities Right

First note priority was adopted for this design, i.e. first note pressed to channel 1, second to channel 2, and so on. If more notes are pressed then the system can cope with, these are locked out. The reason for this, as opposed to last note priority, is that first note priority stops the note jumping that can occur when, momentarily, more notes are pressed than the system caters for.

Binary Notation

When the code (note code) driving the decoder energises a contact which is closed, the output of the OR gate goes high, showing a unique code on the input representing the particular note being pressed. This code, the note code, is arranged such that the lowest note is binary zero, the next note up binary one, the next

two and so on up to N.
The scanning can also be achieved using a multiplexer.

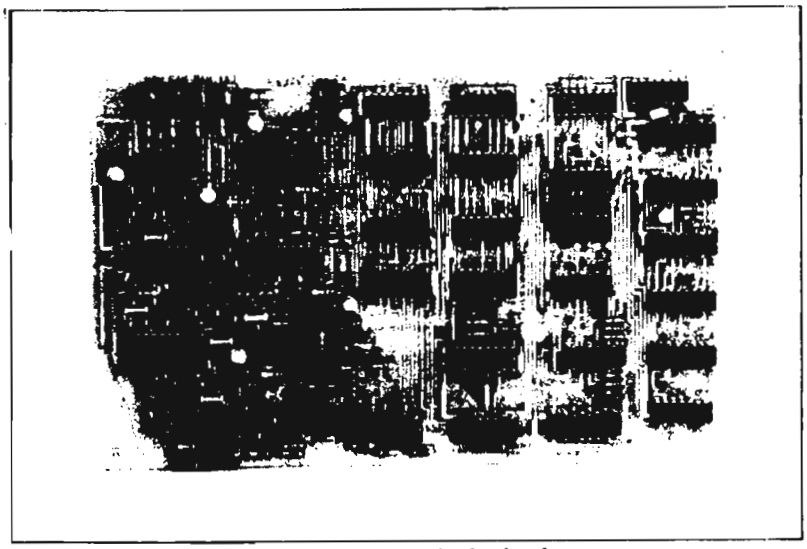
The size of keyboard decided on was a 4 octave one having 49 notes. Hence this makes the value of N 49 and therefore the size of the note code will be 6 bits (64 possibilities). In fact this is useful in that a 6 bit code will be just big enough to scan a 5 octave keyboard (61 notes) if required. In the case of this design it will simply be a mattter of adding 12 extra diodes since the decoder already had a total of 64 outputs. Incidentally, the scanner will have another output not yet mentioned. This is called 63rd note, (the 63rd output on the decoder) which simply provides a pulse to the decision logic to say that a scan has been completed. The multiplexer method would require decoding of the note code to do this.
The scanner simply gives each note a binary code, but how can this be extracted as a set of control voltages with associated gate signals?

Pumping Caps

The note code is changed to an analogue voltage using a D-A converter, the output of which is switched onto the correct analogue channel and held using a set of sample and holds. The gate signals are dealt with in a similar way using CR circuits. The counter for the note code causes the scanner to increment from the lowest note upwards. If three notes are depressed the scanner reaches the lowest note first and causes the output of the D-A to be stored by channel 1 sample and hold, and channel 1 gate capacitor to be pumped up. On moving on the channel counter is incremented, prepairing the output channels for channel 2 data. When the scanner reaches the second note up the process occurs again only using channel 2 and again for channel 3, with the third note. When the scan has been completed the channel counter is reset and made ready for the next scan.

Dying Charge

If on the next scan the notes are still depressed, the gate capacitor will again be pumped up maintaining the gate output high. When a note is released the time constant is such that the gate capacitor's charge dies away in about one and a half scan



The largest of the four PCB's, carrying the logic circuitry.

times, thus removing the gate signal. By experiment it was found that the scan time needs to be about 4 mS. Even when a key is pressed and released very quickly, it will have been scanned about ten times or more. The NAND gate should be mentioned because it allows two adjacent notes to be played. This is because if two notes right next to each other are depressed, the output of the scanner remains high for the duration of both notes and so only one note would be detected. By NANDing the scanner output with the clock the output is broken up allowing adjacent notes to be detected.

Note Jumping

Although this circuit will work, it is far from satisfactory. When notes other than the top note are released, the channels on which the remaining notes appear, above the released note, all jump down one place. This makes the instrument very difficult to play as it must be remembered to release the keys from the upper one downwards, to get a chord that dies away nicely without the note jumping effect.

Special Decision

This means that the simple logic must be replaced by some special decision logic, incorporating a memory of notes already activated in previous scans.

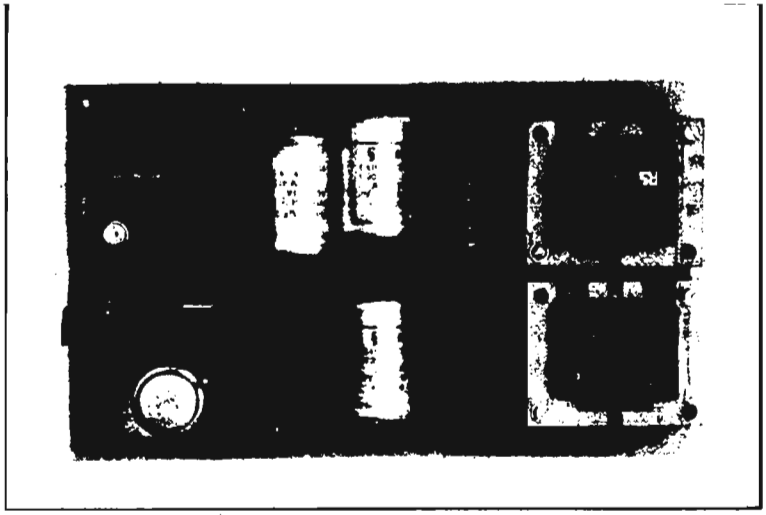
The scheme here is that note codes are gathered into the memory as the scanner sweeps up the keyboard. When the 63rd note is reached, the entire memory is dumped onto the output channels by sequencing the peripheral address lines. It is also necessary to reset all of the gate data bits in preparation for the next scan. This means that while a particular key remains pressed, the gate for that channel will be refreshed on every scan. When the key is released, the gate for that channel will go low when data is again output.

Logical Channels

The effect of the decision logic from the musicians point of view, is that upon playing a chord, say C, E and G the first one depressed normally comes out on channel 1, the second on channel 2 and the third on channel 3 (the difference in time between depresssions need only be milliseconds). There is, however, an exception to this when a note is depressed that is already stored in memory. For instance, if the three note chord described above were depressed such that C was first E second and G third, then it would be expected that C would come out on channel 1, E on channel 2 and G on channel 3. But if a previous chord had been played using the same C which had emerged on channel 2 then the decision logic would cause it to remain on channel 2 and so the E would be placed onto channel 1 and G onto channel 3.

Key Question

Construction of this project will depend almost entirely upon the keyboard it is built around. If you ▶



Power for the keyboard controller comes from this twin transformer board.

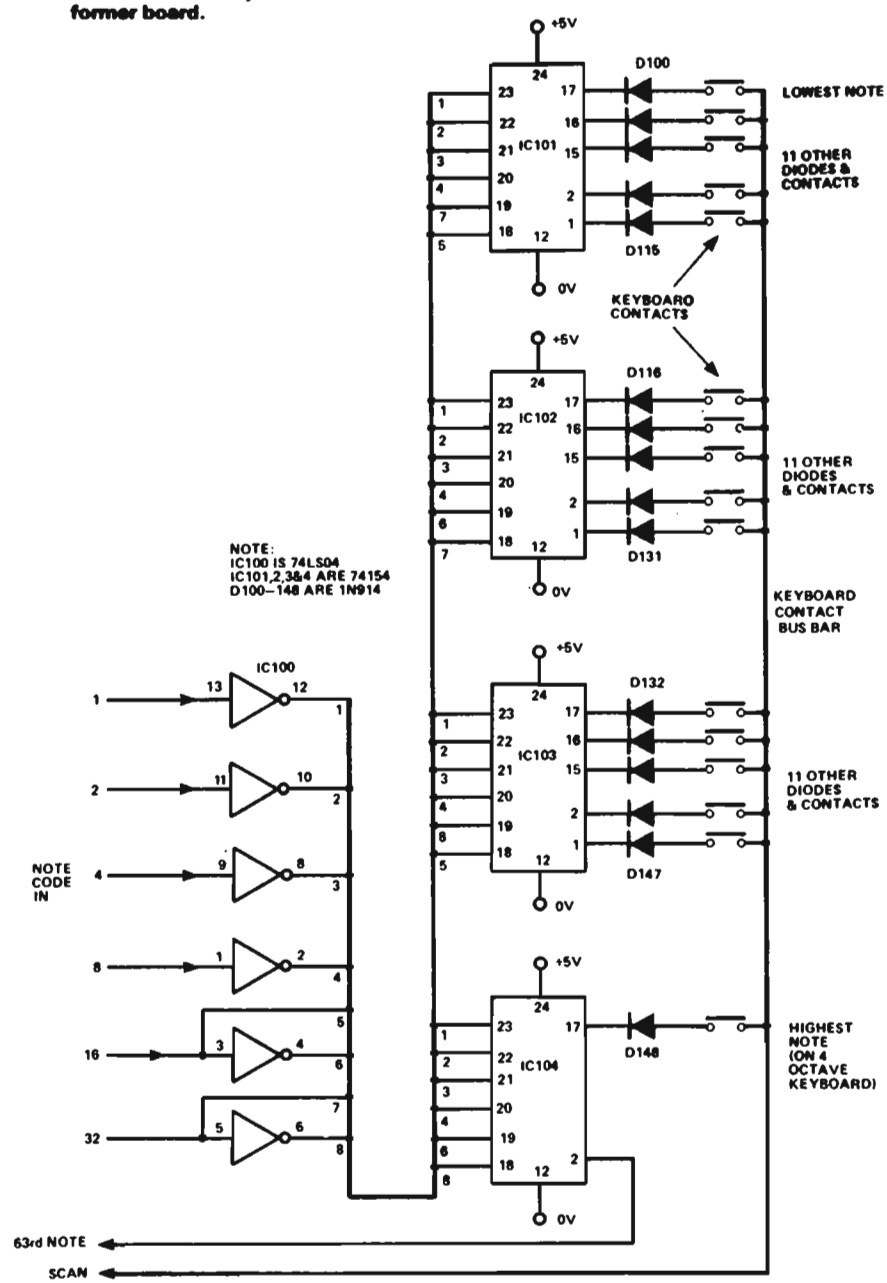


Fig. 1. Circuit diagram of the scanner. The four 74154's are used as a one out of 64 line decoder.

HOW IT WORKS

The Scanner: The IC's used for the scanner itself are 74154, which are one out of 16 line decoders. They are arranged such that one output goes low with the rest remaining high, dependent on the four bit code on the input. These IC's also have a pair of enable inputs both of which must be low. These allow four 74152's to be used as a one out of 64 line decoder, simply by the inclusion of the two inverters on inputs 16 and 32. The 63rd note output is obtained from the 63rd output of the decoder, and the scanner output is taken from the keyboard contact bus bar, having been ORed using the diodes.

Logic: The reaction of the circuit to a new note that has not been picked up by the scanner before is as follows: the note code counter increments the scanner by one note on alternate falling edges of the clock, until it reaches this particular note. The output of the scanner goes high registering that the contact is closed. This triggers the monostable IC12 pin 2 causing its $\overline{Q}$ output to go low long enough to set the decision cycle flip flop IC15 input pin 9 The output of this flip flop pin 11, then inhibits further pulses from clocking the note coder by taking pin 1 of IC15 low. At the same time it initiates the first decision cycle by allowing the counters IC9 (address counter) and IC13 (decision counter) to run by taking their $\overline{clear}$ inputs high. When the output of the decision counter is zero the memory address counter is clocked round, so that the logic can check if the note is already in the memory.

The memory address counter is incremented on the low going edge of the K pulse, which is simply the clock divided by two. Since the decision counter is only 2 bits it was convenient to derive K using the spare single stage counter in the 7493. Note that K is only active during the decision cycles and data block since the counter is cleared down when the scanner is scanning.

When the address counter reaches the number set on the Phonics switch, it is reset, and the decision counter incremented by one via the NAND gates IC18. This starts the second decision cycle where the logic is looking for a spare location to insert the new note. It has been assumed that channel 1 is in use and that the first available channel is channel 2. The circuit stops the data being entered in channel 1 by observing the state of the gate data output from the gate RAM pin 5. If this output is a logical '0' the channel is in use and must not be corrupted, and so the address counter is incremented so that the next channel can be tested.

In the case of this example the decision logic succeeds in entering its data in channel 2, but if the decision counter is incremented a second time before an empty channel is found, simply because all channels are in use, the decision cycle is ended and the scan continued. This third condition of the decision counter is decoded by the NAND gate IC16 and the invertor IC19, and reset is achieved via the three input NAND gate IC22 and invertor IC20, which reset the decision cycle flip flop restarting the note code counter and clearing down the memory address and decision counters.

The second and subsequent times that the scanner is stopped by the note that was loaded in channel 2, the decision logic will only get as far as its first test 'Is the note already in the RAM', so when the memory

address counter reaches 2 the comparator output goes high acknowledging that the note is already entered. This causes the gate bit to be refreshed (since it is reset during data block) along with the data being re-entered into the note memory, (re-entering the data in the note memory is not necessary but occurs due to circuit architecture) after which the decision flip flop IC15 is again reset and the scanner restarted.

All the time a note remains depressed the decision logic will refresh the gate bit associated with the channel in which the note has been placed. At the end of a scan the gate bits are reset immediately after they have been placed on the output channels meaning that if the note is not still depressed on the next scan the gate signal on the output channel will go low in the next data block period.

During a scan the data valid signal is high, it only toggles in data block. Simply enabling the gate RAM during the decision cycle loads it with a '1', since data valid is the input. Note that loading these Ram's with a '1' results in the output going to a 'O' as they invert. This is the reason for the invertors on the outputs of the note RAM's, which are also tri-state for the computer interface.

The clock for the system is an NE555 timer wired in the astable mode.

The Output Channels

There are two outputs per channel which are multiplexed out by the data block period. These are the gate outputs and the control voltage outputs. The gates are obtained from the CD4099 addressable latch (note that these outputs may need buffering depending on the impedance they are driving as the CD4099 is CMOS). The address lines of the latch are attached to the memory address counter and the input is connected to the gate data line (IC10 pin 2). The enable input of the latch is connected to the data strobe line so that as the data is output from the memory the correct gate state (1 or 0) is stored on the relevant channel.

The data sample pulses are for loading the sample and holds on the analogue channels. They are derived from the 1 of 8 decoder and the clock. To interface between the TTL logic and the analogue switches comparators are used so that the analogue signals can be between —3 volts and +12 volts. All the comparator outputs are disabled when the clock is high by using the two resistors R65 and R53 to feed the reference input to the comparators, the clock signal being attached to R65. The binary codes representing the notes are converted into analogue voltages using the D-A convertor IC14.

As the memory address counter is incremented in data block the data in the note memory is converted into an analogue voltage and passed onto the correct analogue channel by the comparator and analogue switch. The D-A convertor has a current output such that when the resistor R82 is added to convert it into a voltage, the output goes more negative with increasing binary codes. The op-amp IC29 (pins 12, 13 and 14) corrects this by inverting the output of the D-A. It also allows the scaling or volts per octave of the keyboard to be adjusted, by varying the resistor in the feedback loop. Another function that the op-amp allows is the summing of voltages that have to appear on all the output channels at once.

There are three sources of voltage that are summed at this point, the tune voltage, the vibrato voltage, and the pitch bender voltage. The tune voltage is derived from a potentiometer which draws its current from the voltage reference circuit. The vibrato voltage is generated by a standard triangle wave generator comprising a regenerative comparator IC29 pins 8, 9 and 10 and an integrator IC29 pins 5, 6 and 7. The output is coupled to the summing amplifier via a pair of back to back electrolytics to remove any DC offset and a pair of resistors that allow their centre point to be connected to earth via an external vibrato depth potentiometer.

Offsets around the circuit are trimmed out using the trimmer RV1 which obtains its reference from the diode D1. Since the offsets are predominantly in one direction due to Q2 the offset control only works in the negative voltage direction.

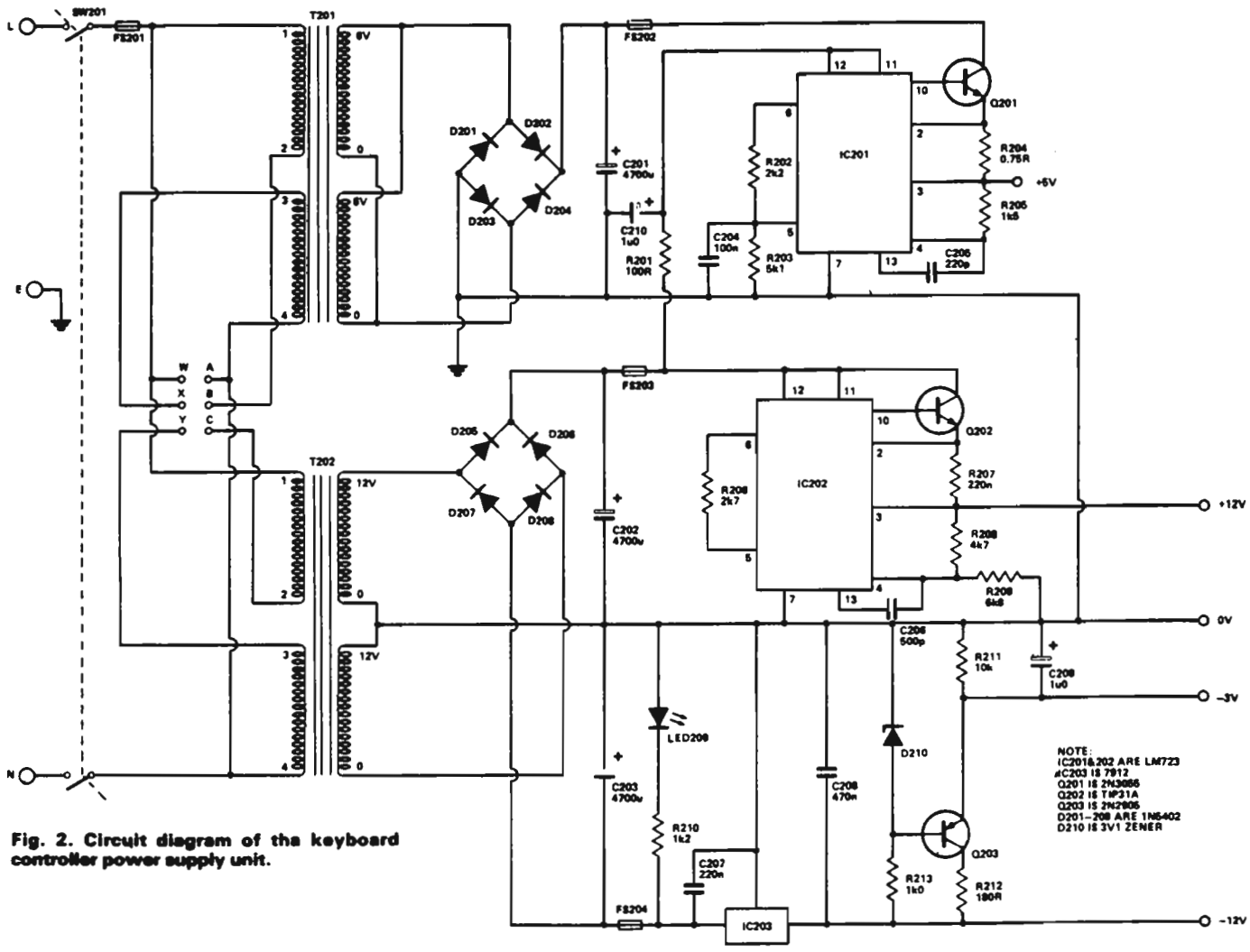


Fig. 2. Circuit diagram of the keyboard controller power supply unit.

employ the ARAK kit, no problems should arise at all. The PCBs are designed to fit their keys and comprehensive instructions are included with the kit.

We have not attempted to go into any detail with any other unit, simply because there is such a great diversity available on the market.

Setting Up

Once the components are all mounted on their boards, each section has to be set up. Let's start with the

PSU

Before the mains is connected to the PSU it should be thoroughly checked for shorts. The three low voltage fuses FS202, FS203 and FS204 should then be removed and the mains turned on. Now check the voltages across the smoothing capacitors C201, C202 and C203 which should be around +8V, +17V and —17V respectively. If this is the case the +12V regulator can be tested by replacing FS203. If this works the +5V regulator can be tested by replacing FS202. As the +5V regulator is supplied from the +12V supply via R201 they must be tested in this order. Finally the —12V and —3V supplies can be tested by inserting FS204. It should be noted that the fuse holders may need bending to give correct contact to the fuses as they are very simple pressed steel pieces for PC mounting.

The Logic Board

Check the logic board thoroughly for shorts on supplies. It is also wise to 'buzz out' every connection on the board to test for continuity which may well save a lot of fault finding time, but note that it will not guarantee correct operation as it does not test for shorts.

When these preliminary tests have been carried out and the power supply unit is functioning correctly power can be applied to the logic board. Firstly only apply the +5V supply until the TTL is known to be working correctly.

And a Log

Once the logic is working the analogue section can be tested. This time some setting up can also be done:

First check the positive reference is sitting at about 6V2 above earth. This level can be increased using the trimmer RV7 if a higher reference is required for any reason.

If the touch circuit is not to be used R63 should be removed as it will probably cause the output of IC29 pin 14 to saturate against one of the supplies as the output of the touch circuit is indeterminate.

R19 sets the maximum glide rate. The smaller it is the longer the maximum glide rate will be. However, it is unwise to make it any smaller as the maximum range is set by the V_{CE} on SAT of the switching transistor, this only creating an offset when it is turned on and not when it is turned off. It may be necessary to increase the value of R19 although problems will probably occur on one channel only and will most likely be remedied by replacing the switching transistor for one with a lower V_{CE} SAT.

PARTS LIST

ETI ▶

BOARD 2

R201	100R
R202	2k2
R203	5k1
R204, R207	0R75 1W
R208	4k7
R209	6k8
R210	1k2
R211	10k
R212	180R 1W
R213	1k
CAPACITORS	
C201-203	4700u 25V electrolytic
C204	100n polyesti
C205	220p
C206	500p
C207	220n polyester
C208	470n polyester
C209, C210	1u0 35v electrolytic
SEMICONDUCTORS	
Q201	2N3055
Q202	TIP31A
Q203	2N2905
IC201, IC202	LM723
IC203	7912
D201-208	1N5402
D209	LED
D210	3V1 Zener
MISCELLANEOUS	
TX201	RS 207-683
TX202	RS 207-699

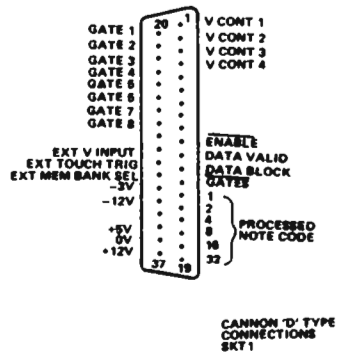
1A, 1A5, A (2 off) fuses and holders, DPST rocker switch.

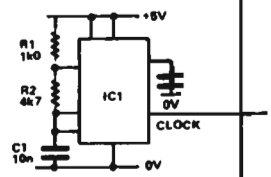
BOARDS 3 AND 4

SEMICONDUCTORS

IC100	74LS04
IC101-104	74154
D100-148	1N914

2 off of these components are required, as board 4 is identical to board 3.





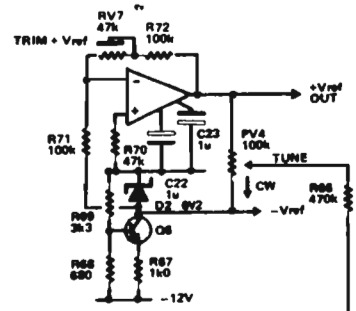
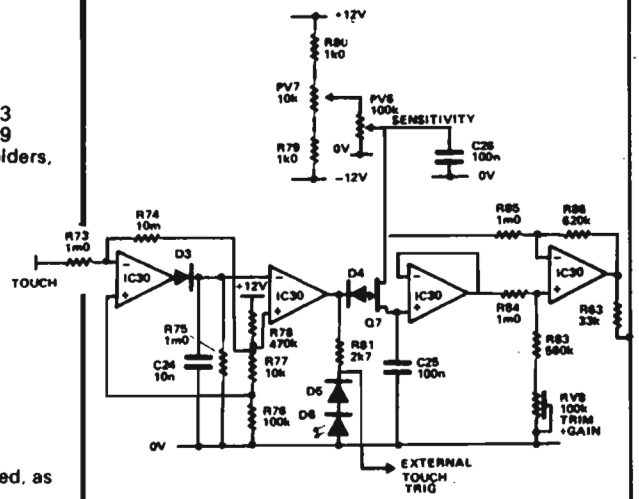
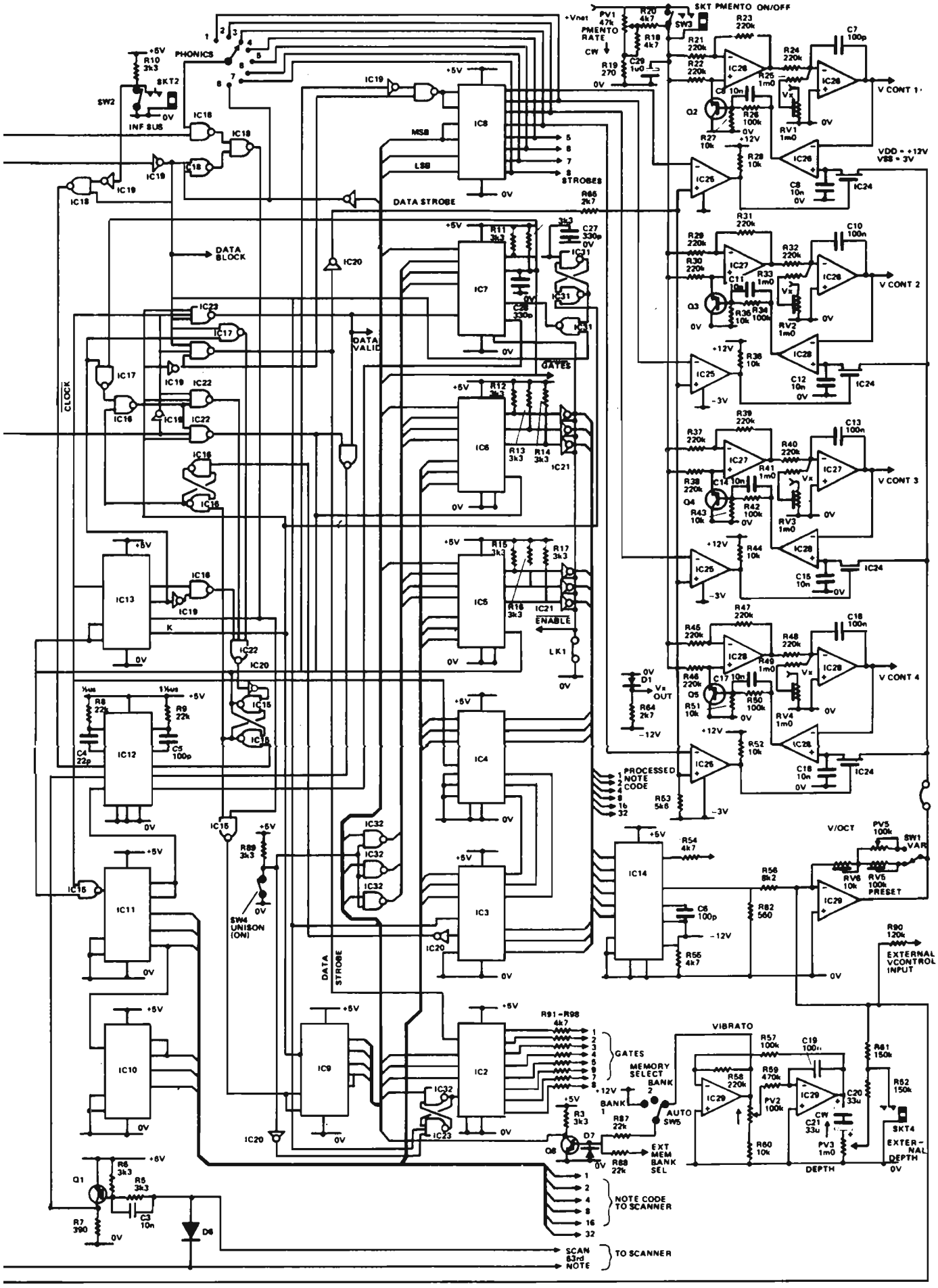
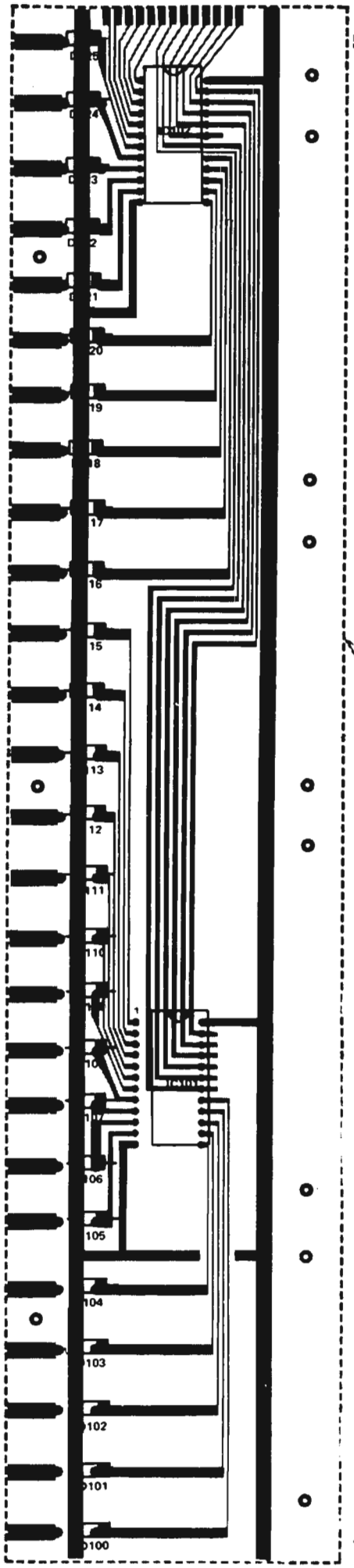


Fig. 3. Circuit diagram of the logic board.




SKT PMENTO ON/OFF
PMENTO RATE
PHONICS
SKT2
INF SUS
DATA STROBE
MSB
LSB
STROBES
DATA BLOCK
DATA VALID
GATES
ENABLE
CLOCK
PROCESSED NOTE CODE
V CONT 1
V CONT 2
V CONT 3
V CONT 4
VDD = +12V
VSS = -3V
V/OCT
EXTERNAL VCONTROL INPUT
VIBRATO
DEPTH
EXTERNAL DEPTH
GATES
MEMORY SELECT BANK
AUTO
EXT MEM BANK SEL
NOTE CODE TO SCANNER
SCAN 63rd NOTE
TO SCANNER
SW4 UNISON (ON)
R91-R98 4k7
IC1
IC2
IC3
IC4
IC5
IC6
IC7
IC8
IC9
IC10
IC11
IC12
IC13
IC14
IC15
IC16
IC17
IC18
IC19
IC20
IC21
IC22
IC23
IC24
IC25
IC26
IC27
IC28
IC29
IC31
IC32



PARTS LIST

RESISTORS		R61, R62	150k
R1, R67, R79, R80	1k0	R63	33k
R2, R18, R20, R54, R55, R91-R98	4k7	R65, R81	2k7
R3-R6, R10-R17, R69, R89	3k3	R68	680R
R7	390R	R70	47k
R8, R9, R87, R88	22k	R74	10M
R19	270k	R82	560R
R21-R24, R29-R32, R37-R40, R45-R48, R58	220k	R83	560k
R25, R33, R41, R49, R73, R75, R84, R85	1MO	R86	620k
R26, R34, R42, R50, R57, R71, R72, R75	100k	R90	120k
R27, R28, R35, R36, R43, R44, R51, R52, R60, R64, R77	10k	**POTENTIOMETERS**	
R53	5k6	PV1	47k
R56	8k2	PV2, PV4, PV5, PV6	100k
R59, R66, R78	470k	PV3	1MO Log
		PV7	10k
		RV1-4	1MO
		RV5, RV8	100k
		RV6	10k
		RV7	47k

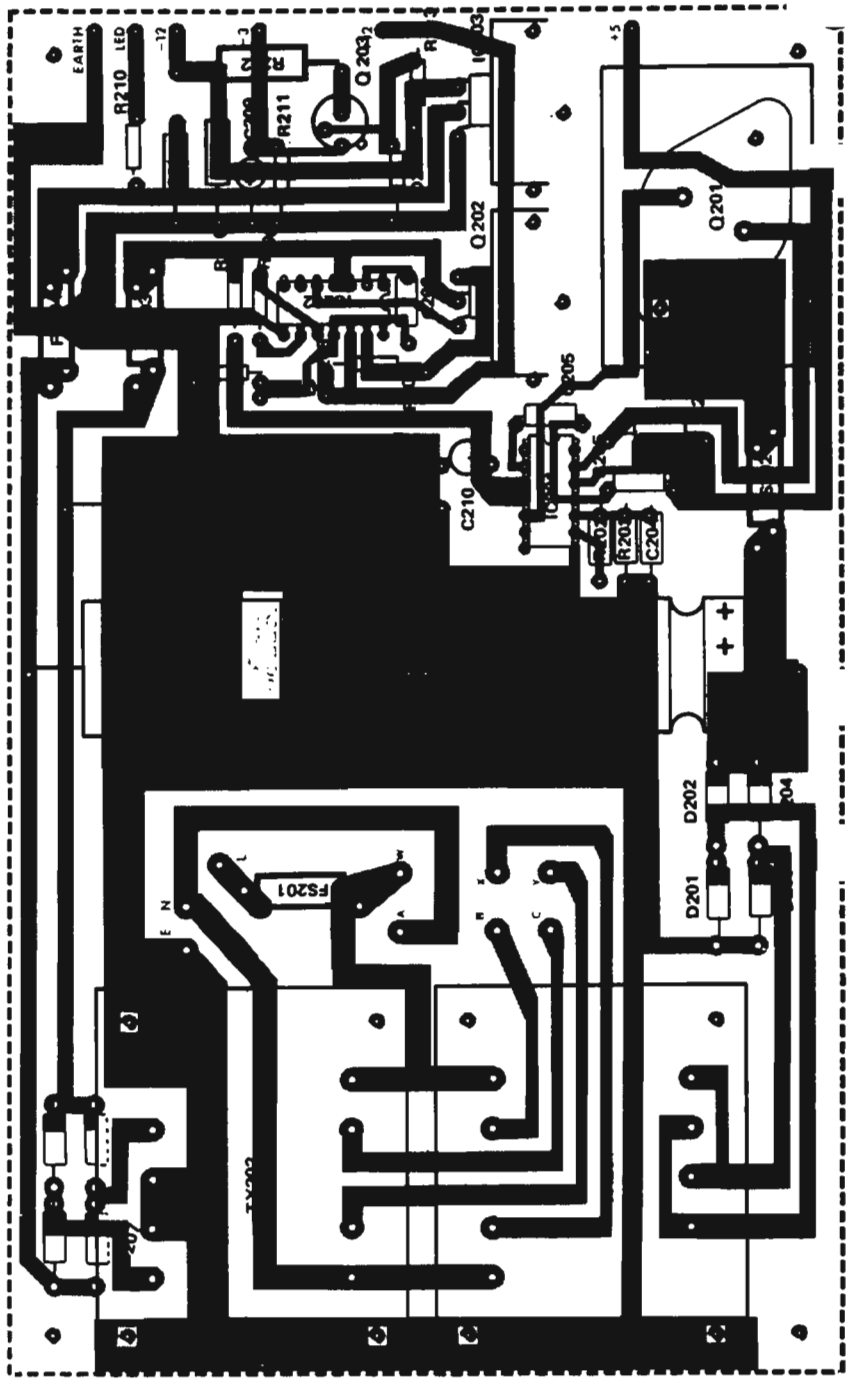


Fig. 4. (below) Component overlay of power supply board.
Fig. 5 (above) One of the two keyboard PCBs, designed to fit Araks keys.

CAPACITORS		IC9-11, IC13	74LS93
C1-3, C8, C9,		IC12	74LS123
C11, C12, C14, C15		IC14	MC1408L-8
C17, C18, C24	10n polyester	IC15-18,	
C4	22p	IC31, IC32	74LS00
C5, C6	100p	IC19, IC20	74LS04
C7, C10, C13, C16,		IC21	74LS366
C19, C25, C26	100n polyester		(or 74LS368)
C20, C21	33u	IC22, IC23	74LS10
C22, C23, C29	1u0 35V electrolytic	IC24	CD4066
C27, C28	330p	IC25	LM339
SEMICONDUCTORS		IC26, IC27,	
Q1	BCY72	IC28, IC30	TL084
Q2-6, Q8	BC107	IC29	LM4741
Q7	2N5163	D1, D3-7	1N914
IC1	NE555	D2	6V2 Zener
IC2	CD4099	D8	LED
IC3, IC4	74LS85	**MISCELLANEOUS**	
IC5-7	7489	37 way 'D' skt, stereo jack (3 off), SPDT switch (4 off), SPDT centre off.	
IC8	74LS155		

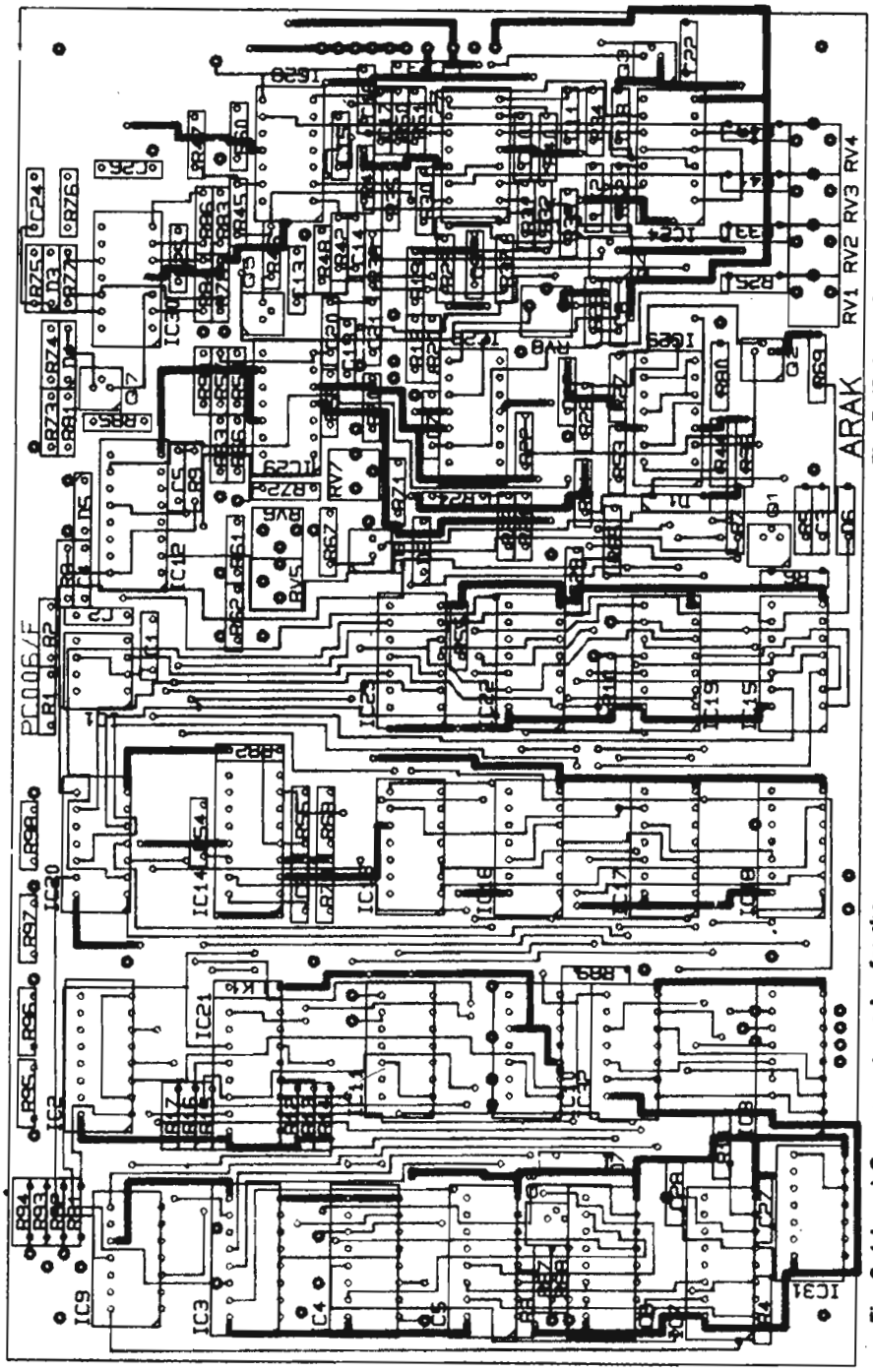


Fig. 6. (above) Component overlay for the logic board.

Fig. 7. (Below). One of the keyboard PCBs made to fit Arak's keys.