A digital low frequency spectrum analyzer, using a programmable pocket calculator

A DIGITAL LOW FREQUENCY SPECTRUM ANALYZER, USING A PROGRAMMABLE POCKET CALCULATOR

by

W. P. Spruit

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering

Eindhoven The Netherlands

A DIGITAL LOW FREQUENCY SPECTRUM ANALYZER, USING A PROGRAMMABLE POCKET CALCULATOR

by

W.P. Spruit

TH-Report 78-E-85

ISBN 90-6144-085-8

Eindhoven
June 1978

June 1978

A DIGITAL LOW FREQUENCY SPECTRUM ANALYZER, USING A PROGRAMMABLE POCKET CALCULATOR

by

W.P. Spruit

TH-Report 78-E-85

ISBN 90-6144-085-8

A DIGITAL LOW FREQUENCY SPECTRUM ANALYZER, USING A PROGRAMMABLE POCKET CALCULATOR

Abstract:

A measuring instrument utilising the Texas Instruments' appliances SR 52 or SR 56 is described.
An application as a digital noise measuring system is discussed in detail.
The same instrument can, however, perform other functions.

Results concerning sine-wave and noisy input signals are presented.

The instrument makes use of an input-output processor for the SR 52/56, developed by Verkroost [1], for which the author designed and tested print lay-outs.

INTRODUCTION

The tremendous popularity of the pocket calculator has made it a really versatile piece of computing apparatus, and compared to other arithmetical hardware, the price is very low.
The use is, however, restricted, because it is designed to be man-operated. It could be used as a micro-computing element in intelligent measuring equipment, if it were possible to feed it directly with digital signals, and read out the results directly, too.
This problem has been solved largely by Verkroost [1]. He designed an input-output processor for the SR 52 and SR 56 programmables. The author made prints for this processor and, with the help of these, built an automatic measuring instrument (see fig. 1).

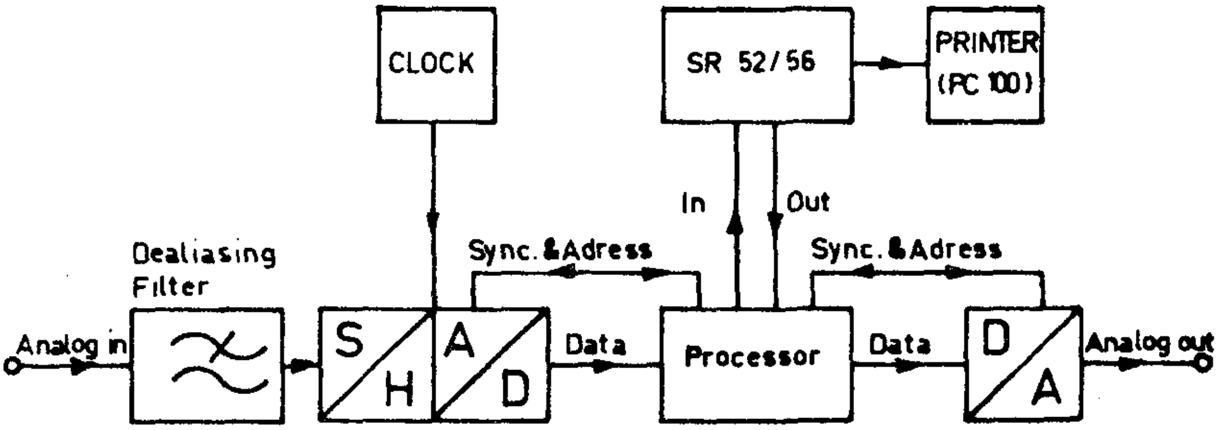


Fig. 1: Block diagram of the measuring system

We use this instrument as a programmable digital filter, which computes the mean square of the filtered signal. The filter frequency ranges from virtually zero to approx. 0.1 Hz (the lower frequency limit being only a function of the experimenter's patience).
This use offers advantages because analog filtering becomes difficult at very low frequencies, and other techniques (like Fast Fourier transform) are very expensive owing to the large computer memories needed. For detailed information ón the central part of the system (the processor) see ref. [1].

1. THEORY OF OPERATION

A digital filter can be represented by fig. 2 [2,3,4].

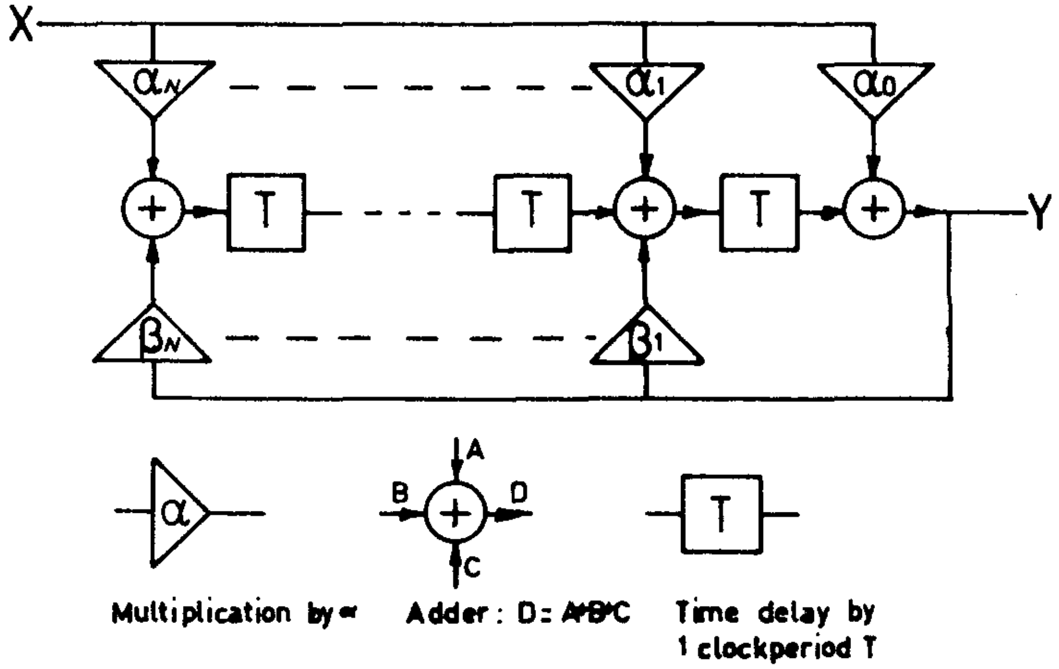


Fig. 2: A digital filter

The transfer function of such a system is:

$$H(z) = \frac{Y}{X} = \frac{\sum_{o}^{N} \alpha_n z^{-n}}{1 - \sum_{1}^{N} \beta_n \; z^{-n}} \tag{1}$$

where z^{-n} means an n clock-period time delay. On the other hand, the output of any digital system can be written as:

$$h'(t) = \sum_{K=0}^{\infty} h(KT)\,\delta(t-KT) \tag{2}$$

where h(KT) is the sampled value of the signal at time KT, and h'(t) the sample-and-hold output, which changes at time T, 2T, ..., KT.

The Laplace-transform of this function is:

$$\mathcal{L}\{h'(t)\} = \sum_{K=0}^{\infty} h(KT)e^{-pKT} \tag{3}$$

with $p = \alpha + j\omega$

Further we have

$$z = e^{pT} \tag{4}$$

This gives:

$$\{h'(t)\} = \sum_{K=0}^{\infty} h(KT) z^{-K} \tag{5}$$

This relation makes it possible to find the digital equivalent of an analog filter by means of digitizing. However, we do not use this straightforward method, because digital filters have transfer functions which recur with a period of 1/2 ω_s (ω_s = the sample frequency of the system = $\frac{2\pi}{T}$) in the frequency domain. This would need complicated analog filter to be converted.

Therefore, we use a frequency-domain transformation, which makes it possible to convert all normal analog filters into digital ones, without the properties (in the frequency domain) changing appreciably.

This transform (called bi-linear z-transform) transforms ω of the analog filter into ν of the digital one:

$$W = \frac{z-1}{z+1} \tag{6}$$

where $W = u + j\nu$.
For $W = j\nu$ and $z = e^{pT}$ (4) this gives:

$$\nu = \tan \frac{\omega T}{2} = \tan \frac{\omega}{\omega_s} \pi \tag{7}$$

Relation (7) implies that all ν will be represented by ω's lying between $-1/2 \leq \omega_s \leq 1/2$.
This eliminates folding of ν, but introduces a nonlinear frequency distortion known as "frequency warping".

The effect of frequency warping upon the properties of the system will be shown in section 3. If one takes this effect into account the bi-linear z-transform makes it possible to use all analog filter design methods for digital filters, too.

A source of errors is formed by the fact that the signal is "sample-and-hold" before it can be digitized (see fig. 1).

The transfer function of a first-order sample-and-hold circuit is [3]:

$$X(j\omega) = \frac{\sin\frac{\omega T}{2}}{\frac{\omega T}{2}} \exp - \frac{j\omega T}{2}$$

The modulus of this function is pictured in fig. 3. It is obvious that it influences the total system response.

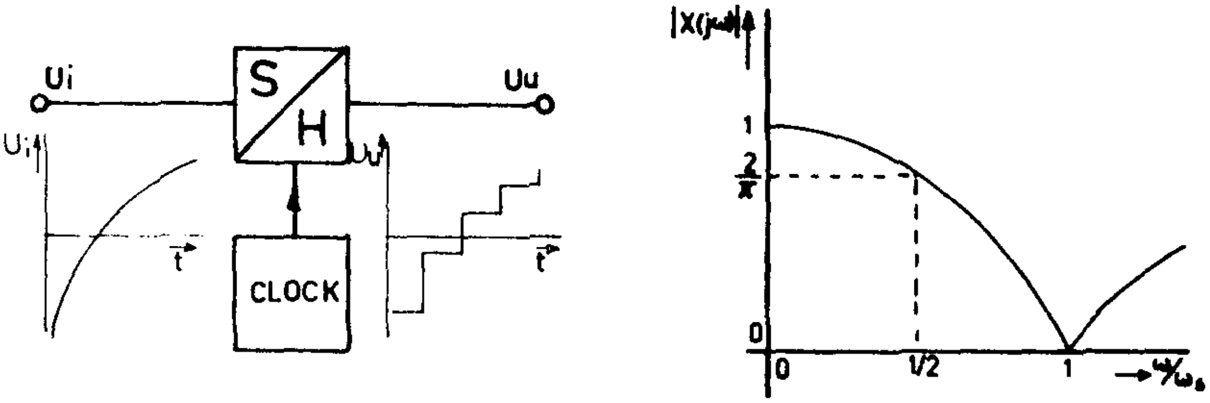


Fig. 3: First-order sample-and-hold with its transfer function

The realization of the function pictured in fig. 2 by means of a pocket calculator is quite simple. The quantities α_o-α_n and β_1-β_n can be stored into the various registers. The delay functions can also be performed by the registers (one for each unit).

Since the SR 52 contains 20 registers, a 6th order filter would be possible.

2. THE MEASURING EQUIPMENT

2.1. The processor

For convenience, we shall give a summary of the working principles of the processor (cf. ref. [1]).

The processor consists of three parts:

- the encoder (print 2),
- the decoding and memory part (print 1, I.C.'s, 3 A-D, 4 A-D and 5 A-D),
- the controller (print 1, I.C.'s 1 A-C and 2 A-E).

The encoder

The keyboard of the calculator itself consists of a switching matrix (see app. 10). The columns of the matrix are fed with the "digit pulses". (16 time-shifted pulses, which are also used to multiplex the display; see fig. 4).

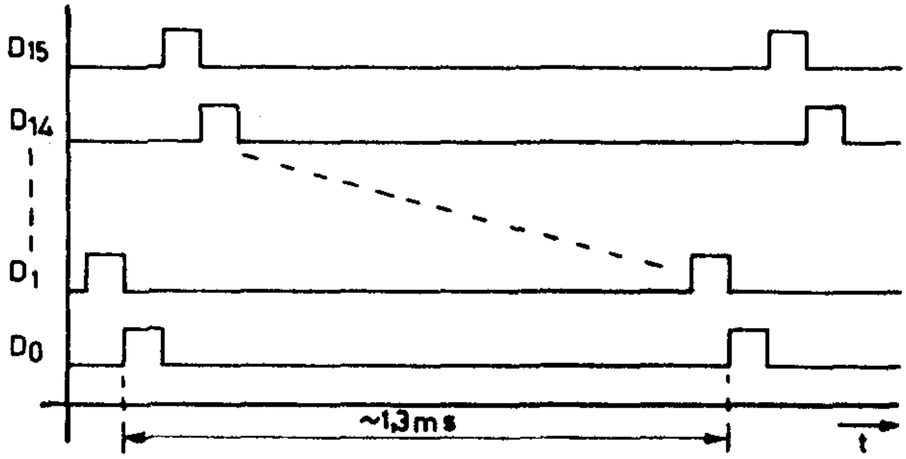


Fig. 4: Digit pulses

The rows of this matrix are connected to the "key inputs" of the arithmetical logic.

The calculator clock (ϕ_1), D_{15} and 4 of the 5 K inputs are carried outside the calculator to the processor. With the aid of ϕ_1 and D_{15} all 16 digit pulses are re-formed in the processor. (I.C. 9, 13 and 16). Encoding of incoming data is now possible by connecting the appropriate digit pulse to the appropriate K input (through I.C. 15). The keys simulated are: 0-9, +/-, . , EE, R/S resp. A (SR 56 and SR 52 respectively).

The decoder and memory

To decode the multiplexed data of the display, it is not only necessary to decode the 7-segment code into a BCD code, but also to know the timing schedule of the multiplexer. This is rather complicated.
The display itself has the format:

- sign of mantissa,
- 10-digit mantissa (with digital point anywhere between sign of the mantissa and sign of the exponent),
- sign of the exponent,
- 2-digit exponent.

The digits are displayed from left to right in time slots of the digit pulses (There are 8 such time slots P_1-P_8 in a digit pulse. P_1-P_8 are generated by I.C. 11) (see fig. 5).

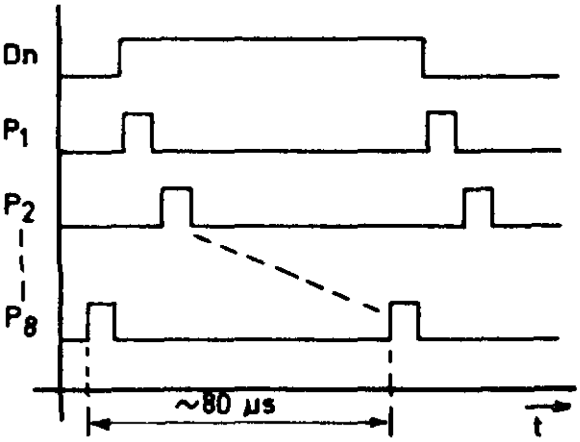


Fig. 5:

The timing is:

- The 10 digits of the mantissa at $D_{12}P_2$ ...... D_3P_2
  with the most significant digit (m.s.d.) at $D_{12}P_2$.
- The digital point at one of $D_{13}P_3$ ...... D_3P_3
  according to the place of the point.
- Information on the EE mode at D_2P_1
  ("1" means exp-mode).
- The exponent at D_2P_2 and D_1P_2
- The sign of the mantissa at D_2P_5
- The sign of the exponent at D_1P_5

According to this schedule a write-enable pulse train is formed (I.C. 1,2,6,7, 8, and 12).

These pulses enable the memory I.C. (I.C. 5A) to store the decoded data in the sequence of table 2.

The decoding itself is performed by I.C. 3B-D, 4B-D and 5B-D.

The coding of the BCD code is given in table 1 (Note that more codes are used than in the normal BCD-code).

TABLE 1

DATA	D C B A
0	0 0 0 0
1	0 0 0 1
⋮	⋮
9	1 0 0 1
-	1 0 1 0
-	1 0 1 1
Dig. point	1 1 0 0
change sign	1 1 0 1
EE	1 1 1 0
No Display*	1 1 1 1

* Leading zero's are omitted

TABLE 2

DATA/ADDRESS	D C B A
MSD'Dig. Point*	0 0 0 0
Dig. 2/Dig.Point*	0 0 0 1
⋮	⋮
Dig. 10/Dig. Point*	1 0 0 1
Dig. Point*	1 0 1 0
Sign Mantissa	1 0 1 1
EE/No Exp.	1 1 0 0
MSD Exp	1 1 0 1
Sign Exp	1 1 1 0
LSD Exp	1 1 1 1

* if displayed

The controller

The controller takes care of the proper timing of reading data into the calculator, storing the result and transmitting this to a D/A converter or to the next processor/calculator combination.

Note that more of these units can be cascaded in order to make higher-order filters. We used only one.

For use with A/D and D/A converters it is necessary to describe the input and output signals of the controller (see fig. 6).

The cyclus begins at the clockpulse, which starts the A/D conversion. The A/D converter generates the signal G(0,1) when conversion is completed. The processor answers with clockpulses called "Read $G_p(0,1)$" which clock the data - in order of table 3 - in the calculator, until all data are clocked in.

Hereupon the A/D converter sinks G(0,1) and generates set $\overline{C_1}$, which sets the calculator to execute the program (During $\overline{C_1}$ the program is performed, and the final display information stored).

When C_1 rises again, the processor generates $C_1(1,2)$, which enables the D/A converter to clock-in data in the order of table 2.

We made no use of $\overline{\text{set } C_2}$ and $\overline{C_2}$, because the D/A converter needs virtually no time for conversion.

The schematic drawings of the processor and the print lay-outs, together with component placing are given in app. 1-6.

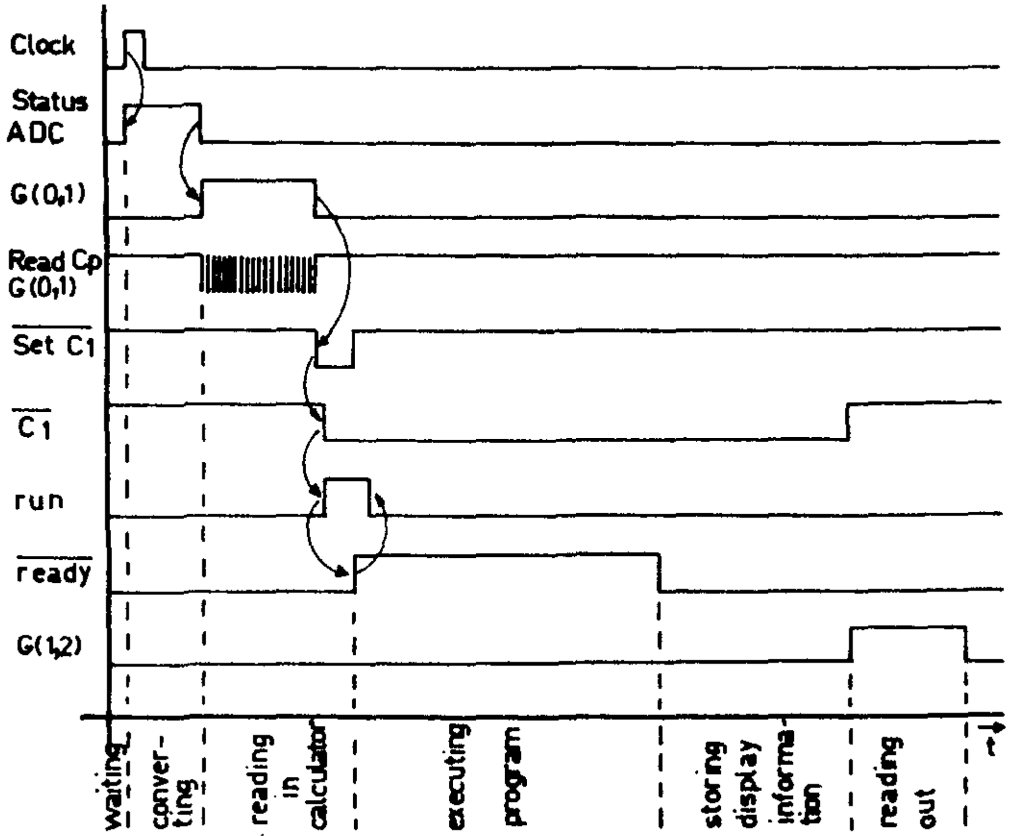


Fig. 6: Timing of the controller signals

2.2. The A/D converter

The A/D converter is rather simple in design (see app. 7). We used a dual slope converter with 3½ digit precision, which needs a minimum of external components, not even a sample-and-hold circuit, because the input-frequency is low compared to the maximum conversion frequency.

The three 8-to-1 multiplexers (-151) multiplex the output data of the A/D converter a they are addressed by the 8-counter (-193), which in its turn gets its clockpulses $G_p(0,1)$ out of the processor.
As long as the outputs of this counter are not all 0, G(0,1) leaves 1 and clocking continues. When G(0,1) sinks, $\overline{\text{set } C_1}$ will give a pulse to indicate the end of sending.

In this way the data are clocked in the order of table 3 (as mentioned earlier).

TABLE 3

Clockpulse no./ DATA	D C B A	
1	0 0 0 0	0
2	1 1 1 1	-
3	1 1 1 X	Sign
4	X X X X	Digit 1 (MSD)
5	X X X X	Digit 2
6	X X X X	Digit 3
7	X X X X	Digit 4
8	1 1 1 1	-

A device has been incorporated giving the warning "sampling too fast". This signal is set when clock * C_1 = 1 and reset when clock * $\overline{C_1}$ = 1. This means, that it operates when the A/D converter starts before $\overline{C_1}$ is lowered.
Some special attention should be given to the overload protection of the A/D converter. This type of converter shows all its outputs "0" in overload condition. This means, that a noisy signal at the edge of overload will be seen as if it were changing from "1999" to "0000", which will cause excessive overload of the filter. Effective clipping of the input signal is, therefore, essential. It is true that clipping disturbs the spectrum, but this is less bad than the effect mentioned above. For technical information on the A/D converter see ref. [5].

2.3. The D/A converter (app. 8)

The calculator displays a 10-digit mantissa and a 2-digit exponent plus signs. This makes it impossible to convert the whole displayed number, because D/A converters have at best a dynamic range of 10^5.
Because the output of the converter is used to feed a recorder, a 3-digit BCD converter is accurate enough. At this point, the need arises to fix a certain format. This fixes the place of the significant digits on the display, and therefore in the memory register of the processor, too. We chose a format of 5 fixed-point digits, and no exponent, whereas the number that is displayed is always less than 1.00000.

Form the 5 digits that follow the point 3 are converted. Usually the 3 leading digits are converted, but if the first one or two of them are zero, the middle respectively last, 3 digits can be converterd. This gives a higher dynamic range to the converter.
As there were no D/A converters available with BCD input and sign, we used a 3-digit converter with a separate inverting amplifier. The output of the system is connected to the output of this amplifier or the output of the D/A converter by means of a reed-switch, which acts upon the sign. This is possible because of the very low speed of the whole system.

For the selection of the digits to be converted a 16-counter (-193), which counts the Read C_p pulses, and a 1-of-16 decoder (-154) are used. The five last digits of the mantissa will thus arrive at the input of the D/A converter in conjunction with the C_p pulses 5-9.
(The signals of pins 7-11 of the -154 will be low at their respective times) (see also table 2).

Three of the output pulses of the 1-of-16 decoder are selected by means of a switch to trigger 3 quad latches (-75) which latch the data for the converter.
This switch acts as a multiplier of the output signal, (i.e. when the first one or two of the digits after the point are zero, one can convert the digits 2, 3 and 4, respectively 3, 4 and 5 instead of 1, 2 and 3. This means a multiplication of the output by a factor 10 and 100 respectively).

An overflow indication is incorporated. The 4th -75 detects whether the 2 digits preceding the point are "no display" and 0. When this is not the case, the output is connected to 10.00 V by means of a reed relay R_1, and an output which is meant to drive an L.E.D. ("format") is switched on.
The sign is converted (as mentioned) by R_2 and the 5th -75, which is triggered upon the 15th Read C_p pulse.

The outputs 1-5 of the 1-of-16 decoder (-154), which stand for the digital point and the 4 digits preceding it, are carried outside. This makes it possible to convert also the 3 digits preceding the point (the fourth digit and the point are used for overflow detection). The use of this second D/A converter is explained in section 3. For technical information on the D/A converter see ref. 5.

2.4. The de-aliasing filter

The schematic diagram of the aliasing filter is given in app. 9.
It is a 6th-order Chebyshew low-pass filter with 0.5 dB ripple [4,6]. This means a ripple of 6%, which is sufficiently small for our purpose (Chebyshew filters have better attenuation slopes than maximum flat filters).
The highest roll-off frequency is chosen 0.095 Hz which gives an attenuation of less than 0.2 dB at 0.09 Hz and 55 dB at 0.20 Hz.
This forms a good de-aliasing filter for a sample-frequency of 0.20 Hz, which is chosen, because the calculator needs approximately 3s to perform its program. For more complicated programs, with more than 5s performing time, the roll-off frequency of the de-aliasing filter can be divided by 2 by means of reed switches.
See further refs. [4] and [6].

3. THE SOFTWARE

To demonstrate the usefulness of the system, the following noise measuring set-up will be demonstrated (see fig. 7):

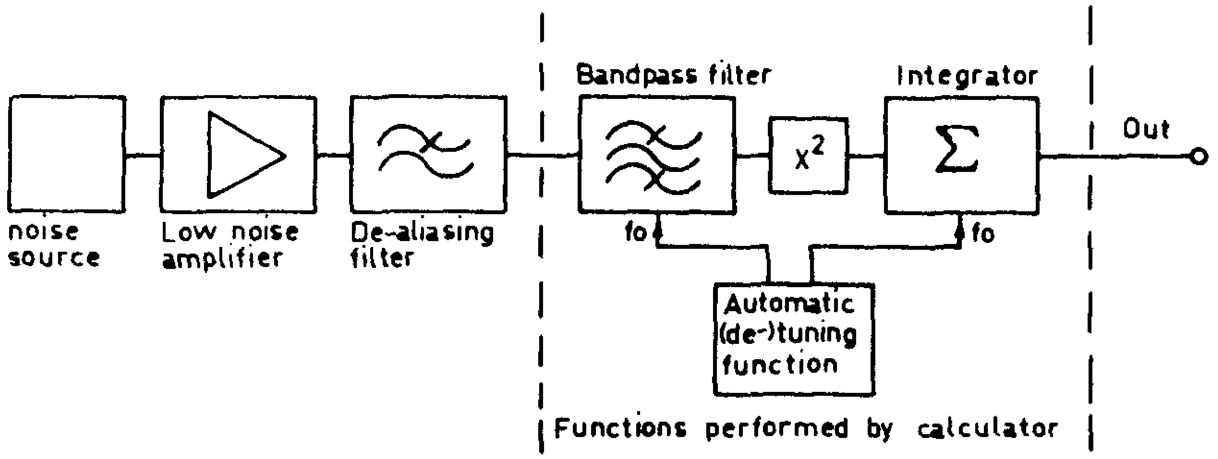


Fig. 7: Noise measuring set-up

A flow diagram of this program is given in fig. 8; the program itself is listed in app. 10.

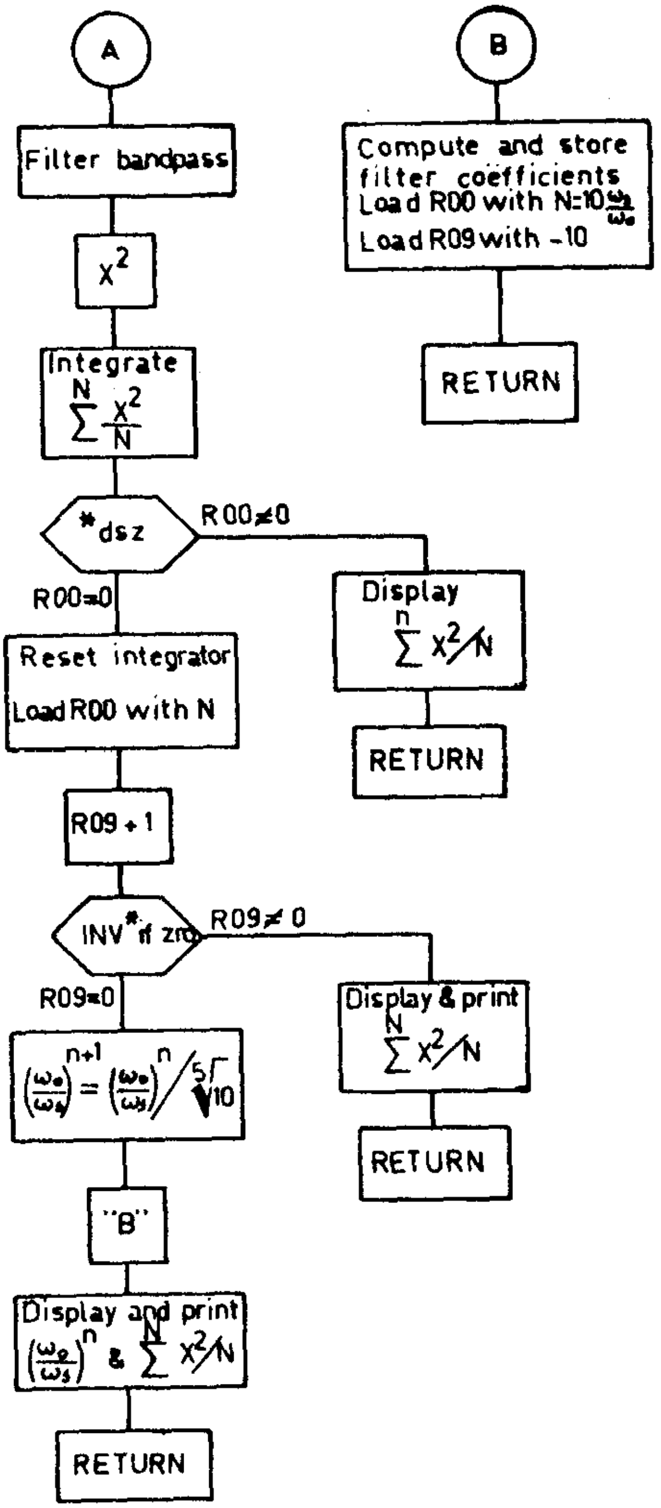


Fig. 8: Flow diagram of the program

The program calculates the filter coefficients $\alpha_{1,2}$ and $\beta_{1,2}$ for a given starting frequency ω_o/ω_s. It calculates the output of the bandpass filter, and the mean square of this filtered signal by means of integration. The integration time is 10 periods of ω_o.

At the end of the integration, the output signal is printed out, and the integrator is reset.

This is done 10 times in order to be able to detect spikes in the filtered signal.

One may average by hand over the significant outputs afterwards.

When the instrument has integrated 10 times for the same frequency, it calculates a new filter frequency:

$$(\omega_o)^{n+1} = \frac{(\omega_o)^n}{\sqrt[5]{10}} \quad .$$

It then calculates the adjoining new filter coefficients and starts filtering again. This gives a spectrum with 5 frequencies per decade. A print command is built in, which prints ω_o and $\overset{N}{\Sigma} X^2/N$ separated by the digital point at the end of each frequency.

This quantity is displayed also. The second D/A converter becomes useful here, because the spectrum can be written on an X-Y recorder, with the aid of the second converter.

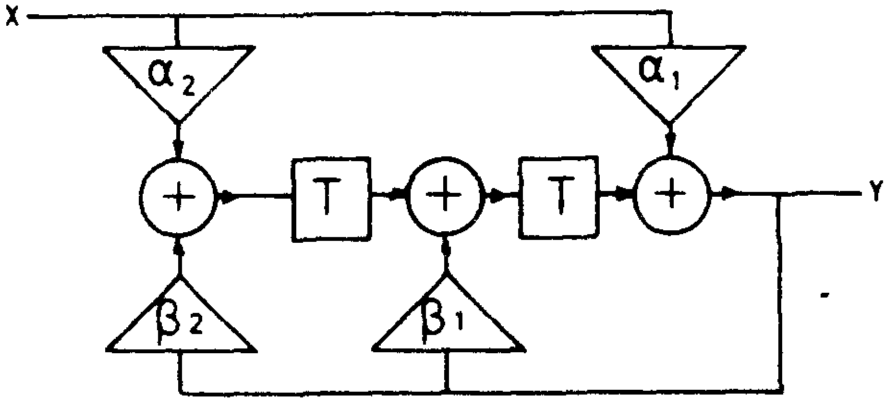


Fig. 9: Bandpass filter

The bandpass filter is a second-order filter with the transfer function:

$$H_{(p)} = \frac{\omega_o}{Q} \; \frac{p}{\omega_o^2 + \frac{\omega_o}{Q} p + p^2}$$

which is a rather common transfer function for analog selective amplifiers.

Applying the bi-linear Z-transform we have:

$$H(z) = \frac{\nu_o}{Q} \frac{1 - Z^2}{(1 + \frac{\nu_o}{Q} + \nu_o^2) + (2\nu_o^2 - 2)Z^{-1} + (1 + \frac{\nu_o}{Q} + \nu_o^2)Z^{-2}}$$

with

$$\nu_o = \tan\left(\frac{\omega_o}{\omega_s} \cdot \pi\right)$$

This transfer function is realized in the block diagram presented in fig. 9 with:

$$\alpha_1 = -\alpha_2 = \frac{\nu_o}{Q} \frac{1}{1 + \frac{\nu_o}{Q} + \nu_o^2}$$

$$\beta_2 = \frac{1 - \frac{\nu_o}{Q} + \nu_o^2}{1 + \frac{\nu_o}{Q} + \nu_o^2}$$

$$\beta_1 = \frac{2\nu_o^2 - 2}{1 + \frac{\nu_o}{Q} + \nu_o^2}$$

The quantities $\alpha_{1,2}$ and $\beta_{1,2}$ which are calculated in the program steps 144-215 are stored in the registers R_{15}, R_{18} and R_{19} respectively. The time-delays T_1 and T_2 make use of R_{02} and R_{03} respectively. In this program Q has been taken 10, but owing to frequency-warping, Q will increase for $\omega_o \to \frac{1}{2}\omega_s$. Fig. 10 illustrates this increase of Q. The output of the total system should be multiplied by this function (for noise measurements). The integrator is quite simple:

$$Y = \sum_{i=0}^{N} X_i \cdot \frac{1}{N}$$

where $N = 10\ \omega_o/\omega_s$ ($\hat{=}$ 10 periodes of ω_o).

The above filter system is an example: fixed higher-order (to 6th order) filters, or even quite different systems, computing correlation coefficients or probability distributions are possible.

As filters, linear and logarithmic outputs for Y and ω can be used.

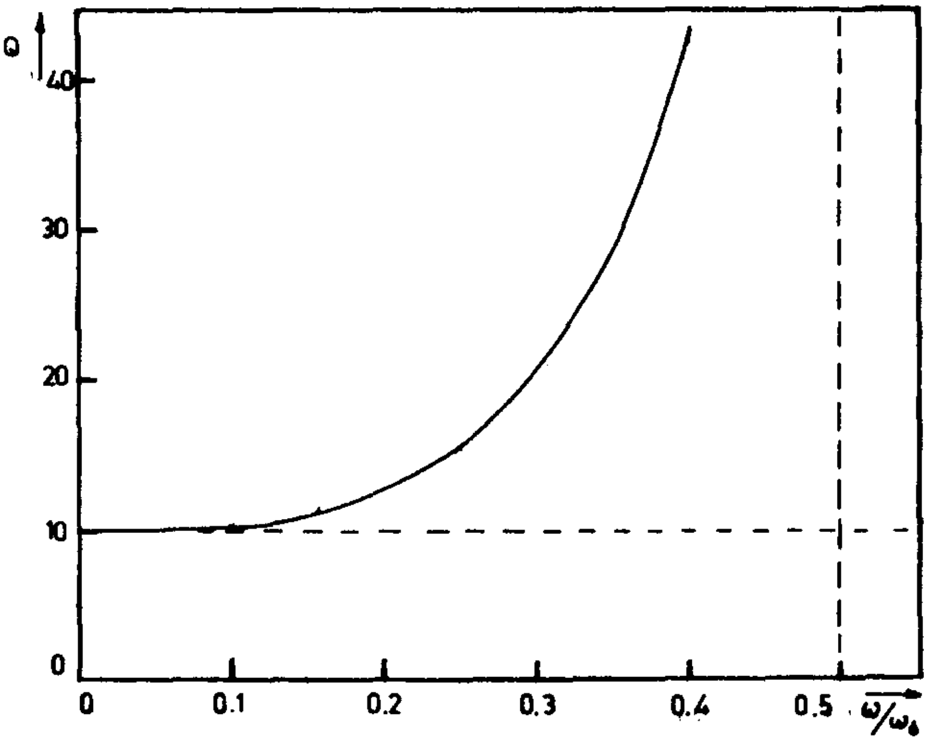


Fig. 10:

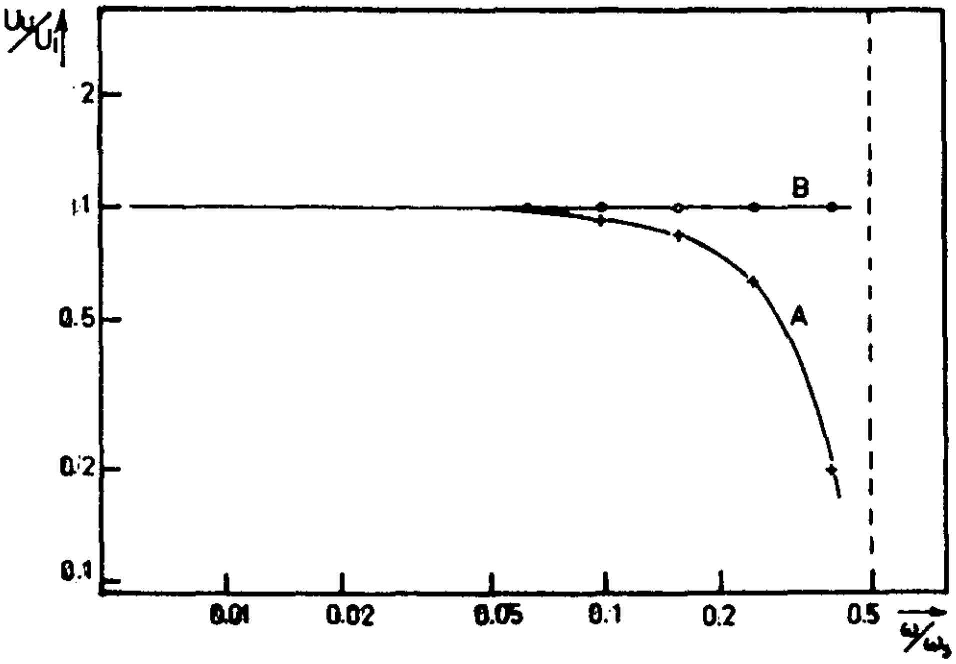


Fig. 11: Frequency response, A: uncorrected; B: corrected

4. RESULTS

Fig. 11 gives the output of the described filter for a sine wave input, with constant amplitude and the bandfilter tuned to this frequency .

A - corrected,
B- corrected for frequency warping, ripple of the aliasing filter and the transfer function of the sample-and-hold (see also figs. 3 and 10). This correction is carried out by hand.
In fig. 12 is represented the output of the system for a sine wave input, with fixed ω_o vs frequency. The effect of frequency warping is here clearly visible.

Fig. 13 shows the output for noise with a 1/f spectrum corrected for the above mentioned effects.

The deviation from the 1/f line is within 20%, as is to be expected for averaging over about 100 periods (we assume here that the statistical error in the spectrum is inversely proportional to $\sqrt{\Delta f . T} = \sqrt{\frac{N}{Q}}$, where N is the number of periods, over which integration is carried out.

In the result of fig. 13 the existent "spikes" in the output of the filter have been ignored. The spikes are caused by R.F. interference from electrical apparatus, and the like. Because the system is rather sensitive to such interference, and because the calculator has a virtually infinite dynamic range (about 10^{200}), the system should be shielded well and provided with a good mains filter.

CONCLUSIONS

The automatic measuring instrument here described, is a convenient and inexpensive low-frequency noise measuring set-up.
Operating the system is rather easy. The use of prerecorded magnetic cards eliminates the need of programming experience of the operator.
The system shows no significant differences in properties with respect to the analog filters normally used for audio frequencies.
The frequency range is, however, restricted to about 0.1 Hz (upper limit).
If one takes into account that the experimenter's time is restricted, the frequency range is about 10^{-4} to 10^{-1} Hz. At 10^{-4} Hz one point of the spectrum takes about 12 days if one integrates over 100 periods. But this is a general restriction to noise measurements at low frequencies.
Other uses than those described are possible, but this a matter for further investigation.

The author wishes to thank Ir. G. Verkroost and Ing. A.C.P. van Meer who developed the input-output processor for the SR 52/56 and who gave a great deal of assistance in building the instrument and applying the digital filter techniques.

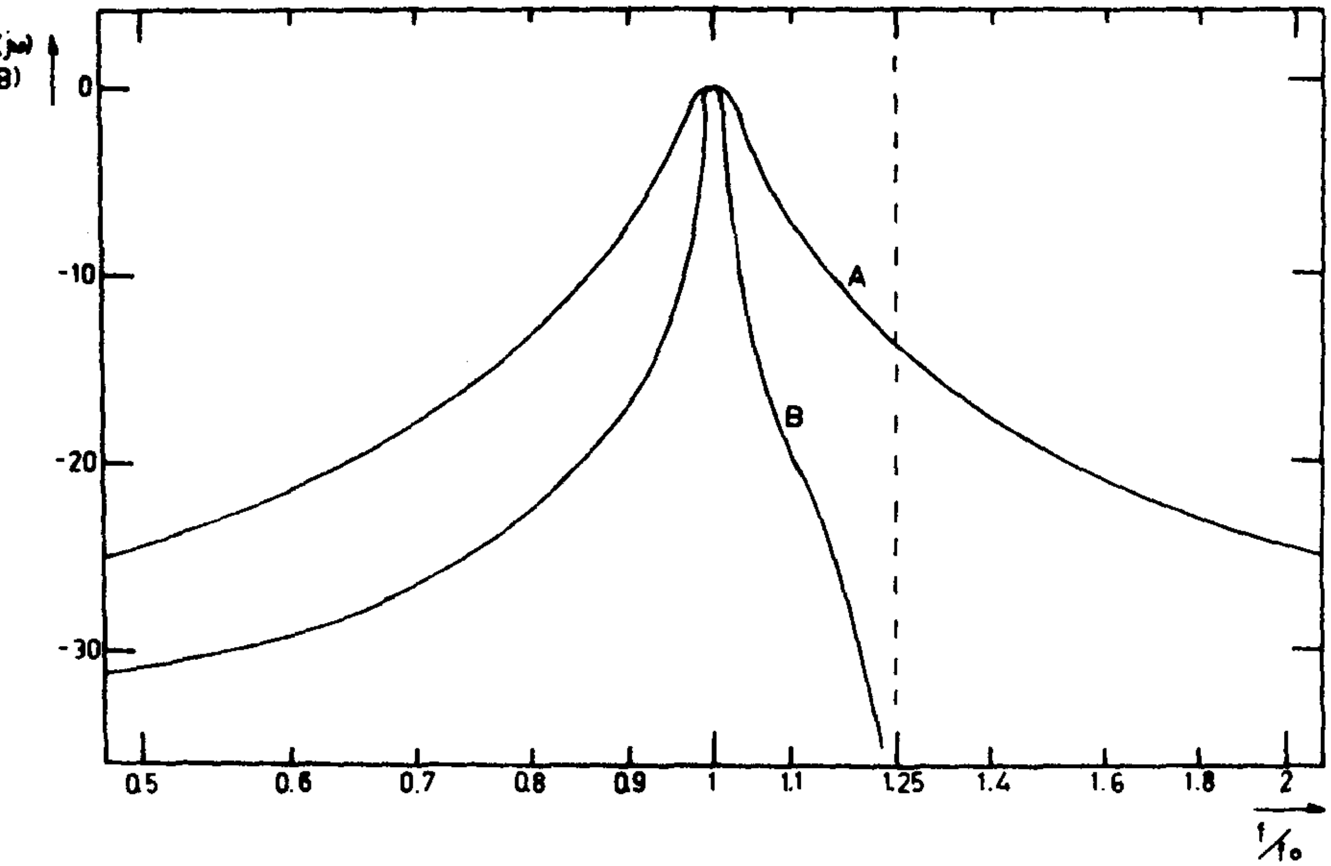


Fig. 12: A: $\omega_o/\omega_s = 0.1$; B: $\omega_o/\omega_s = 0.4$

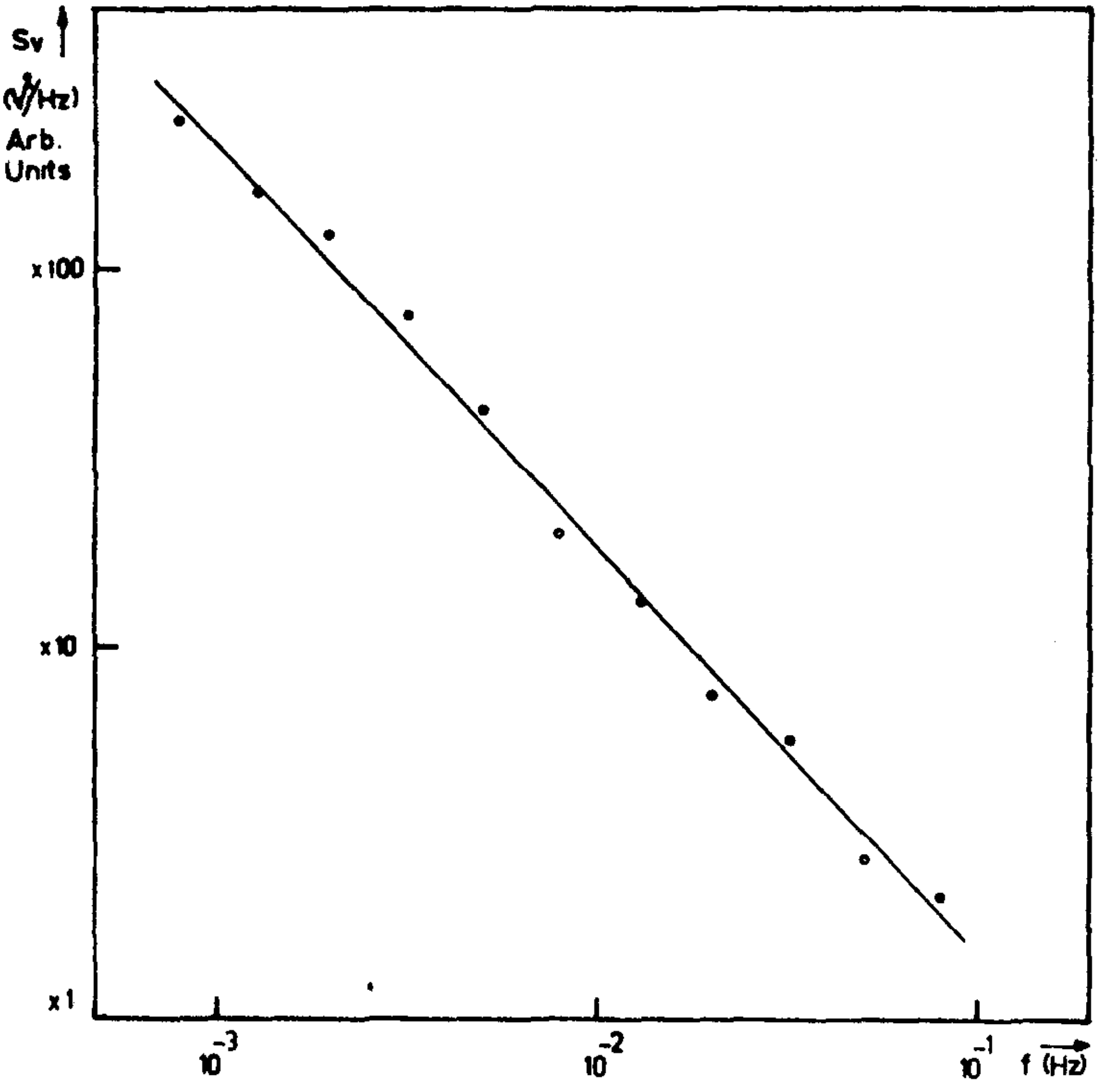
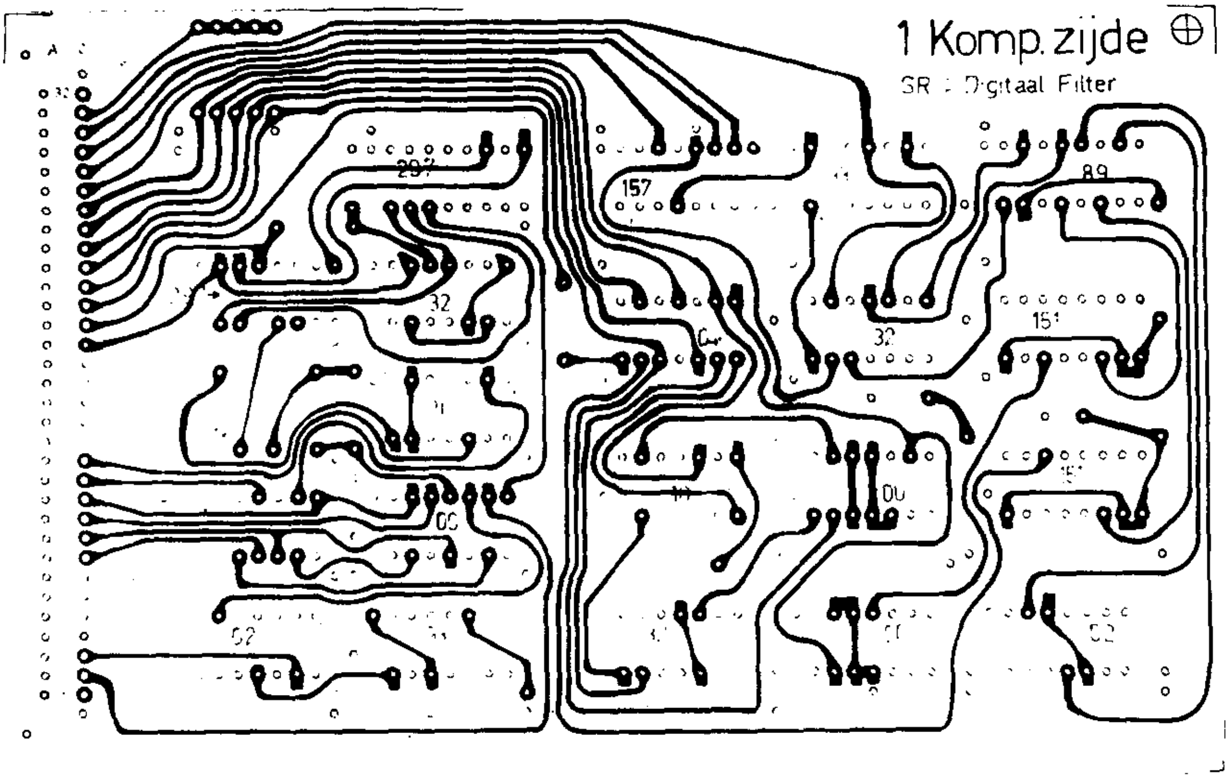
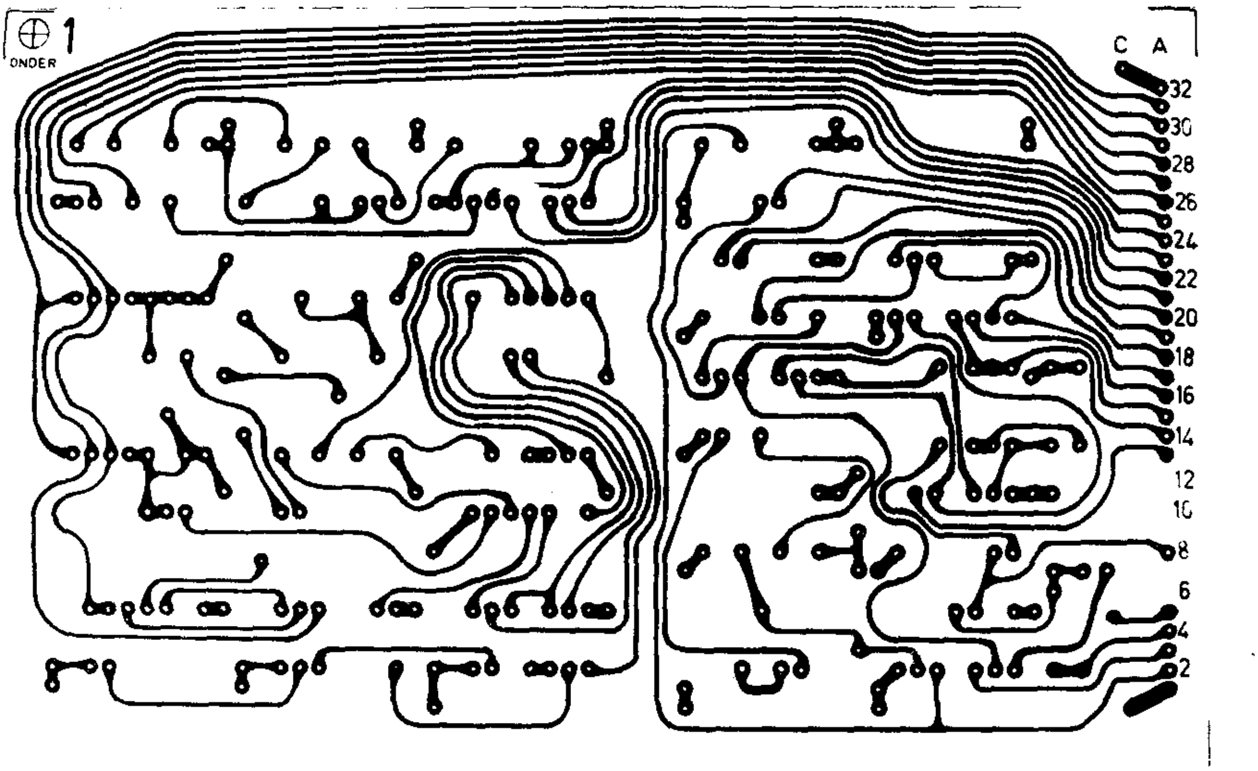
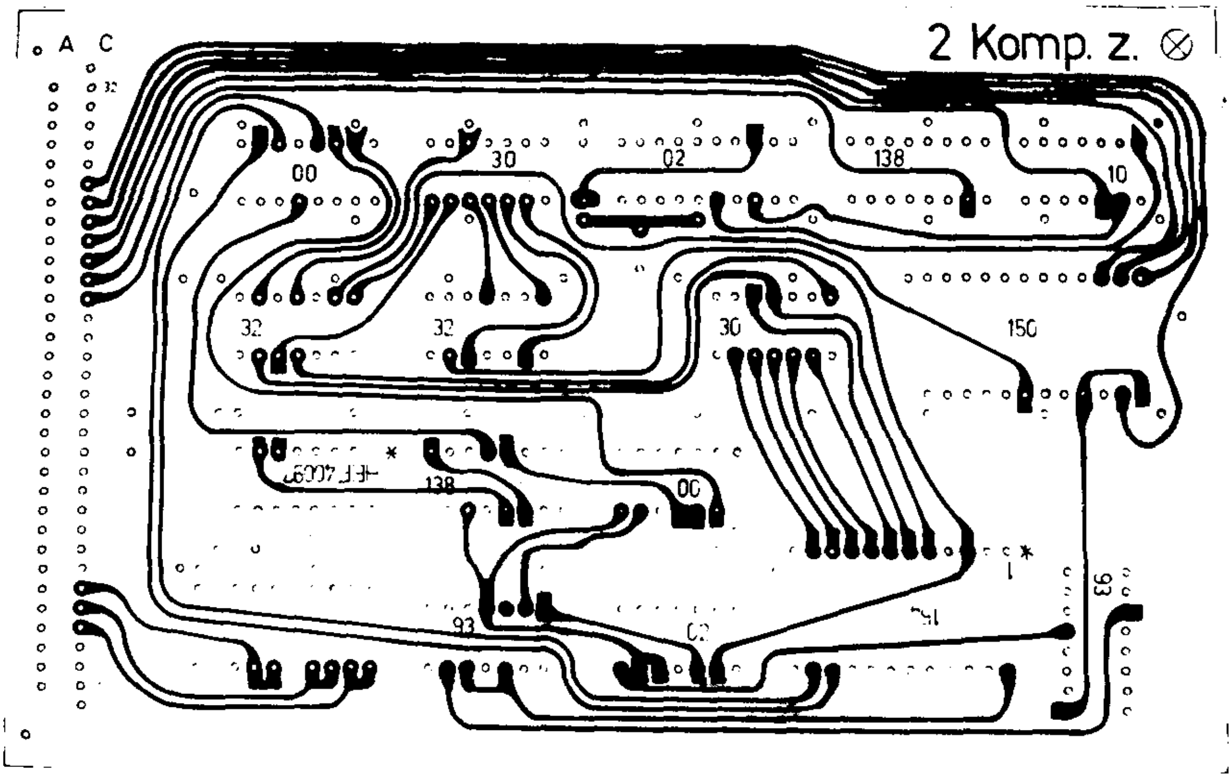


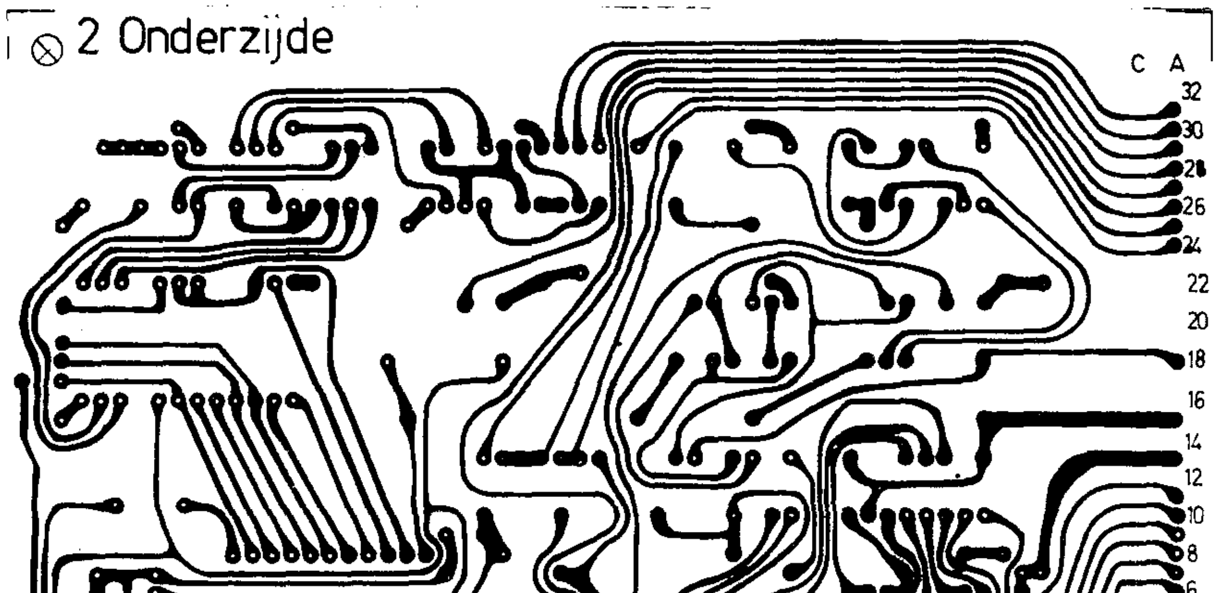
Fig. 13:





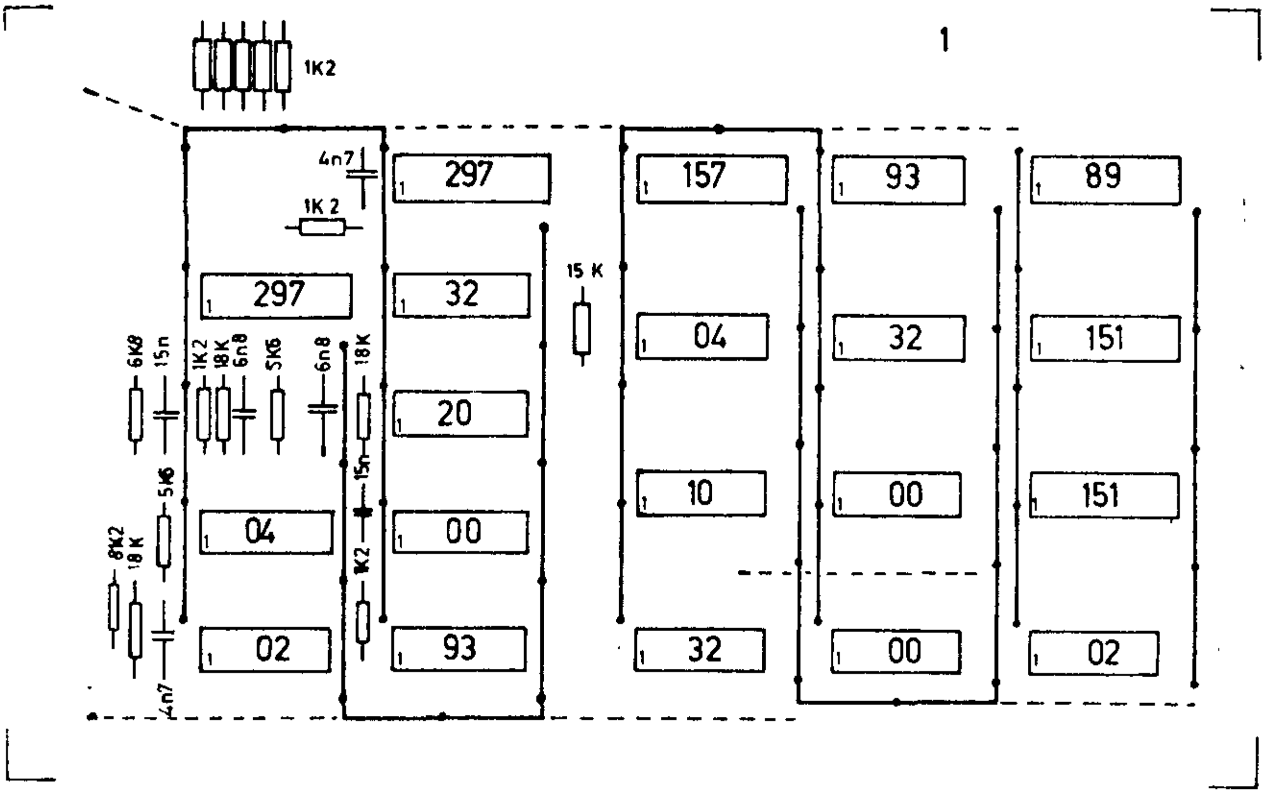
APPENDIX 1
Print 1

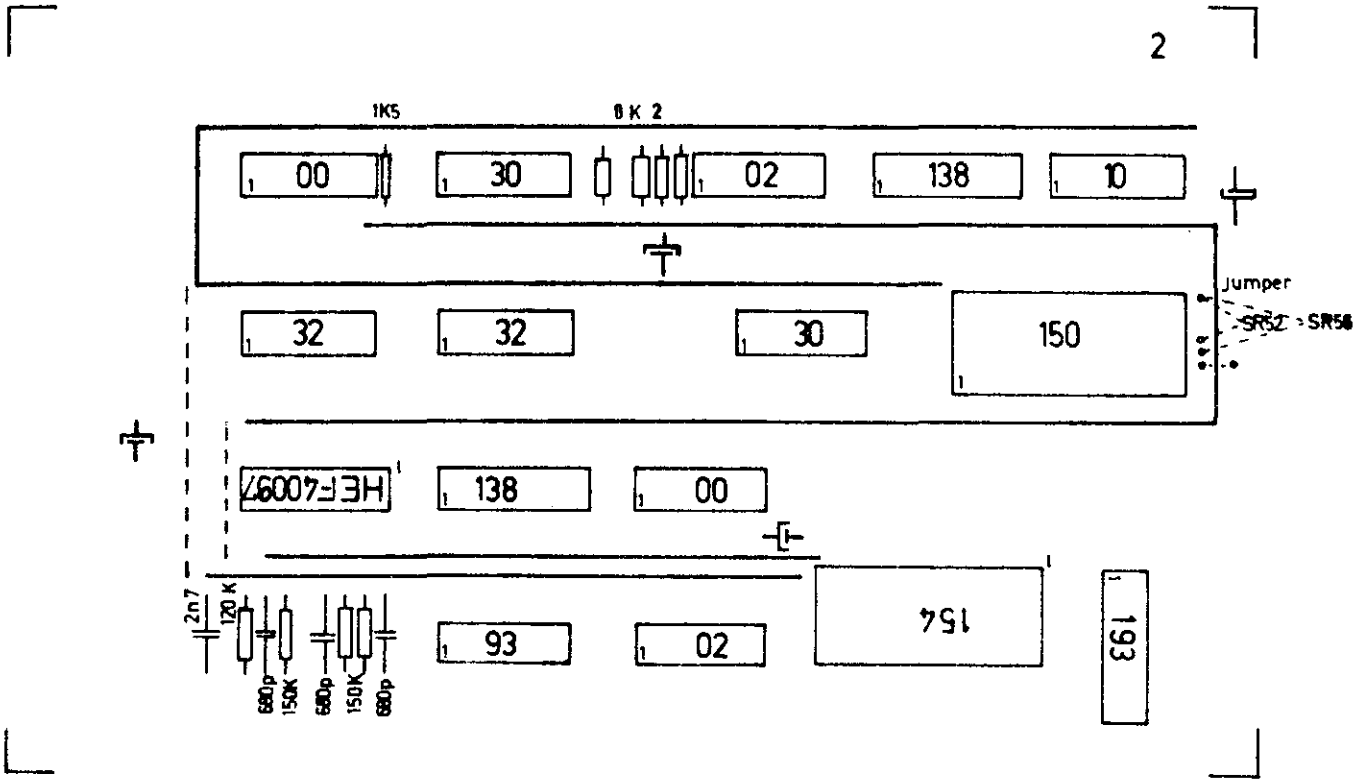




APPENDIX 2

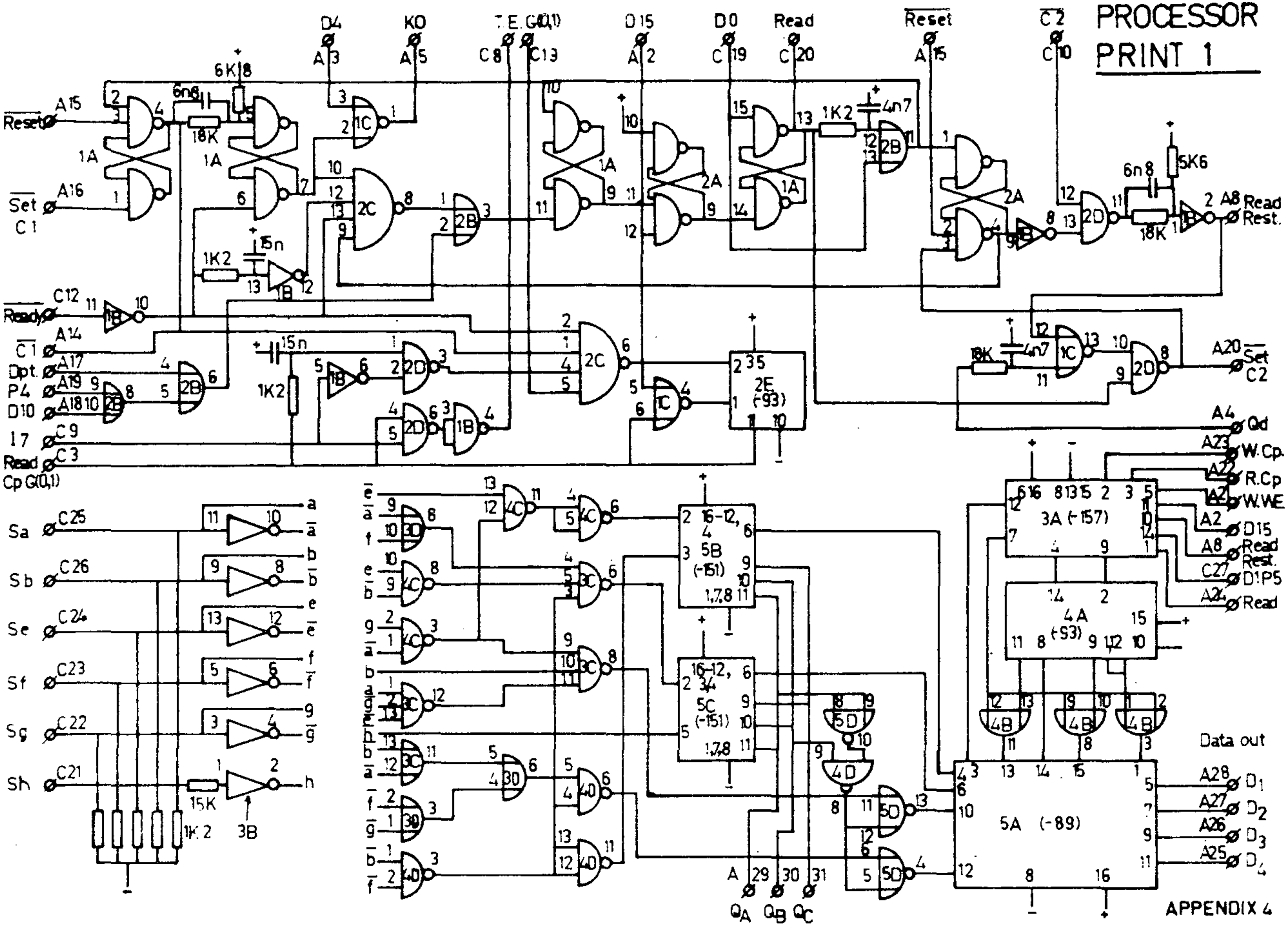
Print 2



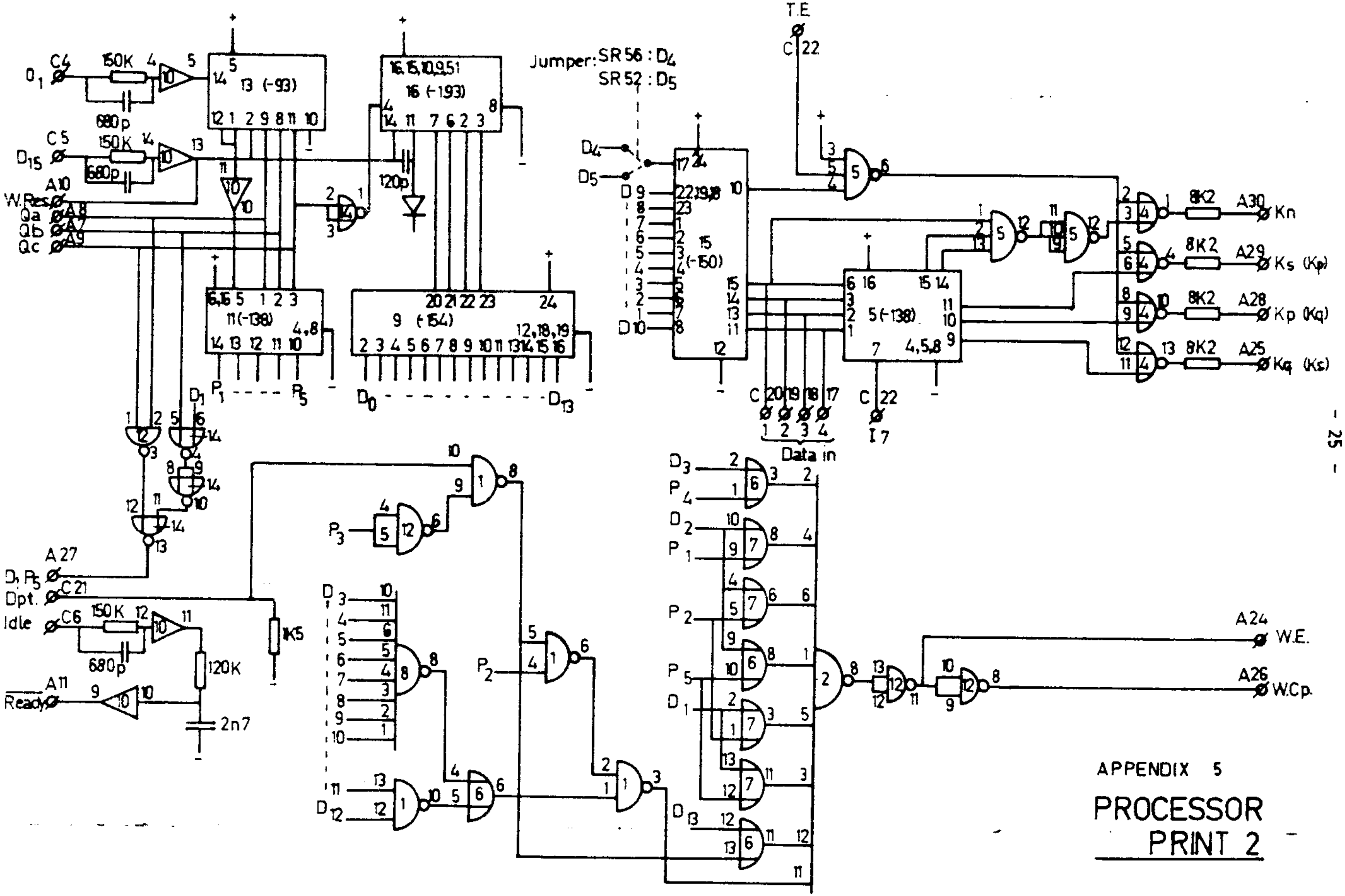


APENDIX 3
Component placing

PROCESSOR PRINT 1



APPENDIX 4



APPENDIX 5

PROCESSOR PRINT 2

Cable-list

Print 1/Connector 1

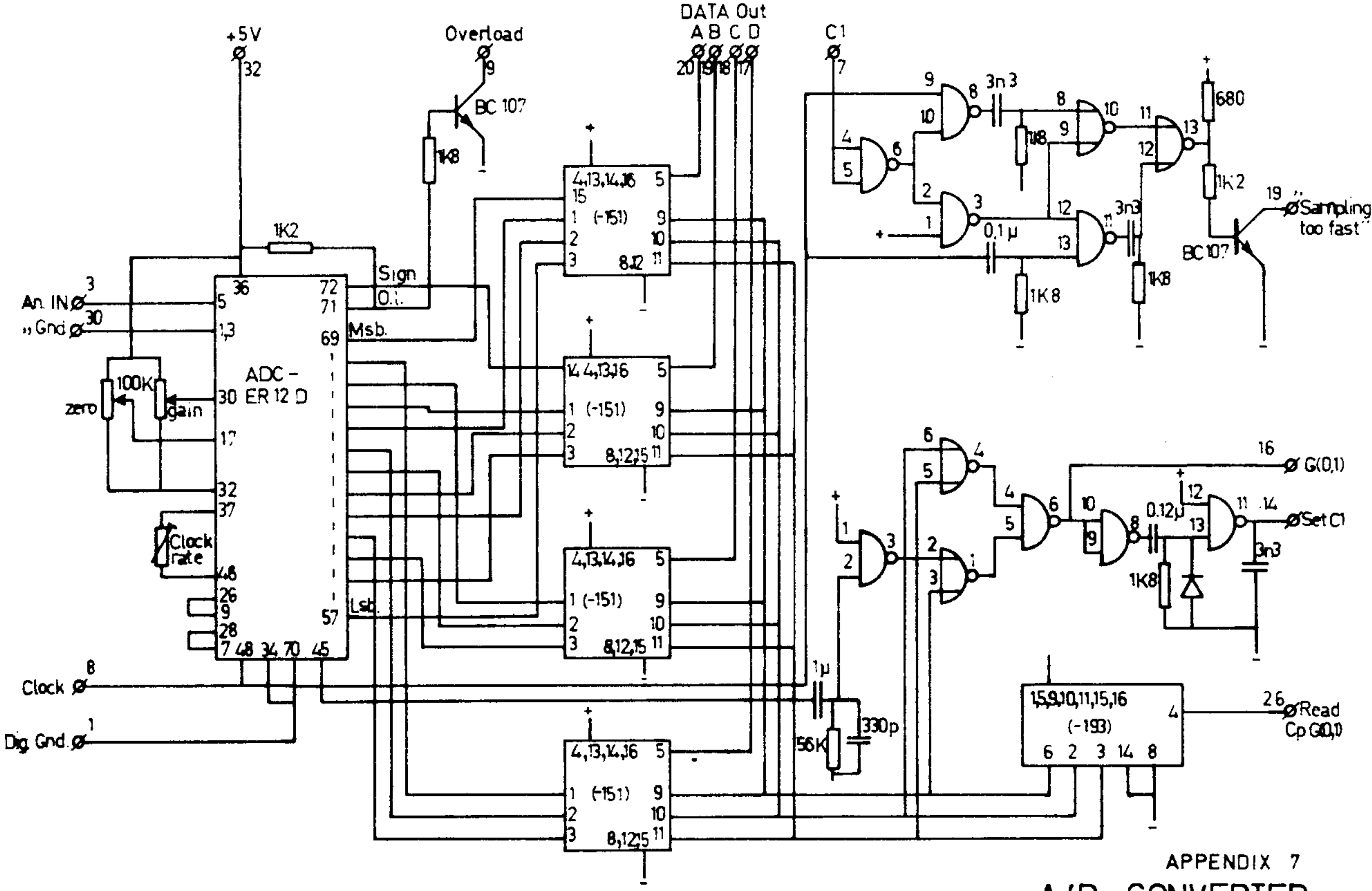
	A		C	
No	Name	→	Name	→
1	V^-	-5V		
2	D_{15}	P_1 C_{28}	Read	P_1 A_{24}
3	$\overline{D_4}$ $(\overline{D_1})$	P_2 A_{28} $(P_2$ $A_4)$	Read C_p G(0,1)	
4	Q_D	P_1 C_{30}		
5	K_o	calc		
6				
7				
8	Read Reset Teller	P_1 C_{29}	TE	P_2 C_{23}
9			I_7	P_2 C_{22}
10			$\overline{C_2}$	
11			G(1,2)	
12			Ready	P_2 A_{11}
13	$\overline{G(1,2)}$		G(0,1)	
14	$\overline{C_1}$			
15	$\overline{\text{Reset}}$			
16	$\overline{\text{set } C_1}$			
17	DPT	P_2 C_{21}		
18	$\overline{D_{10}}$	P_2 A_6		
19	$\overline{P_{4a}}$	P_2 A_{17}	$\overline{D_o}$	P_2 A_5
20	$\overline{\text{set } C_2}$		Read	P_1 A_{24}
21	Write WE	P_2 A_{24}	h 7 segm code	calc
22	Read C_p G(1,2)		g	
23	Write C_p	P_2 A_{26}	f	
24	Read	P_1 C_2/P_1 C_{20}	e	
25	D_4 DATA OUT	D/A	a	
26	D_3		b	
27	D_2		D_1 P_5	P_2 A_{27}
28	D_1		D_{15}	P_2 A_{10}/P_1 A_2
29	A	P_2 A_8	Read Reset	P_1 A_8
30	B	P_2 A_7	Q_D	P_1 A_4
31	C	P_2 A_9	V^-	-5V
32	V^+	+5V		

Cable-List

Print 2/Connector 2

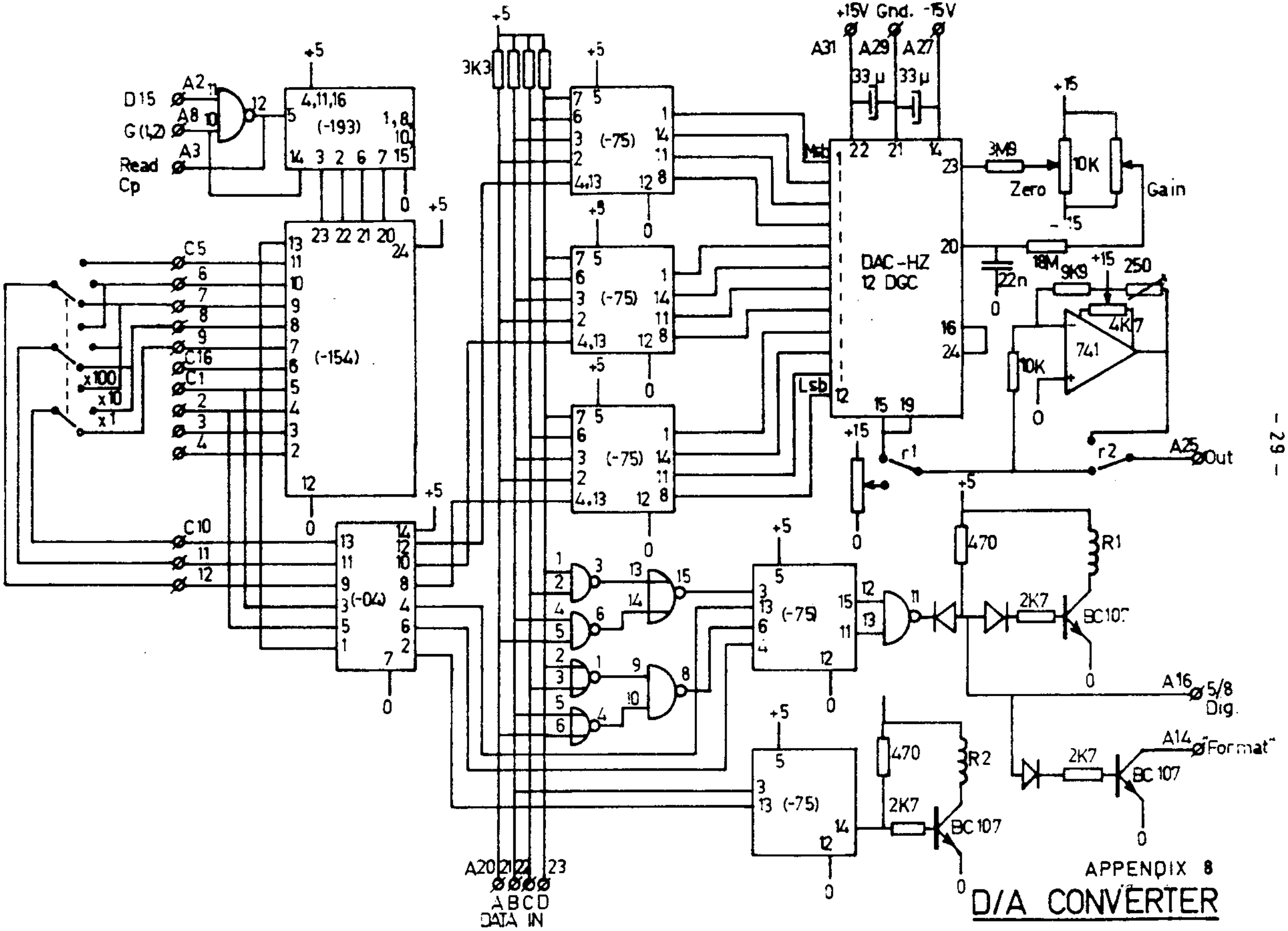
	A		C	
No	Name	→	Name	→
1				
2				
3				
4	$\overline{D_1}$	$(P_1\ A_3)$	1	calc
5	$\overline{D_o}$	$P_1\ C_{19}$	D_{15}	calc
6	$\overline{D_{10}}$	$P_1\ A_{18}$	$\overline{\text{IDLE}}$	calc
7	Q_B	$P_1\ A_{31}$		
8	Q_A	$P_1\ A_{30}$		
9	Q_C	$P_1\ A_{29}$		
10	Write Reset	$P_1\ C_{28}$		
11	Ready	$P_1\ C_{13}$		
12				
13	V^+	+5V	V^+	-
14				
15	V^-	-	V^-	-5V
16				
17				
18	$\overline{P_{4a}}$	$P_1\ A_{19}$		
19				
20				
21			DPT	$P_1\ A_{17}$/calc
22			I_7	$P_1\ C_{10}$
23			T.E.	$P_1\ C_9$
24	Write WE	$P_1\ A_{21}$	D_1 DATA IN	A/D
25	KQ(KS)	calc	D_2	
26	Write Cp	$P_1\ A_{23}$	D_3	
27	$D_1\ P_5$	$P_1\ C_{27}$	D_4	
28	$\overline{D_4}$	$P_1\ A_3$		
29	KP(KQ)	calc		
30	KS(KP)	calc		
31	KN	calc		
32				

APPENDIX 6



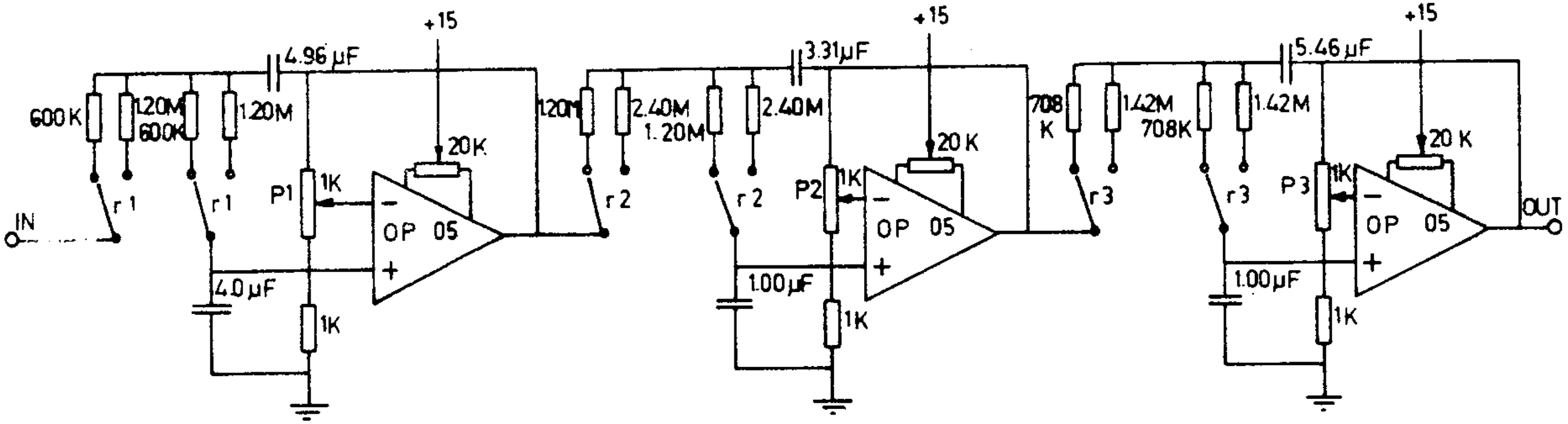
APPENDIX 7

A/D CONVERTER



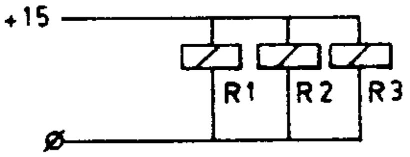
APPENDIX 8

D/A CONVERTER



TUNE TO :

	fo		Hm/Ho
Stage 1	0.03764	Hz	0 dB
Stage 2	0.07297		5.50
Stage 3	0.09609		16.30



APPENDIX 9

DEALIASING FILTER

USER INSTRUCTIONS

Step	Procedure	Enter	Press	Display
1	Enter program	Card (Side A)	2ND READ	
		Card (Side B)	2ND READ	
	Enter start frequency	ω_o/ω_s	B	N_1.00000
	"Run/Reset" on "Run"			

REGISTERS

R_{00}	*dsz	R_{01}	Vin	R_{02}	T_1	R_{03}	T_2	R_{04}	V_o (hp)
R_{05}	Σx^2	R_{06}	ω_o/ω_s $+\Sigma x^2$	R_{07}	ω_o/ω_s	R_{08}	N_2	R_{09}	N_1
R_{10}		R_{11}	ν_o	R_{12}		R_{13}	used	R_{14}	
R_{15}	α_2	R_{16}		R_{17}		R_{18}	α_3	R_{19}	α_4

LABELS

A	Vin	B	ω_o/ω_s	C		D		E	

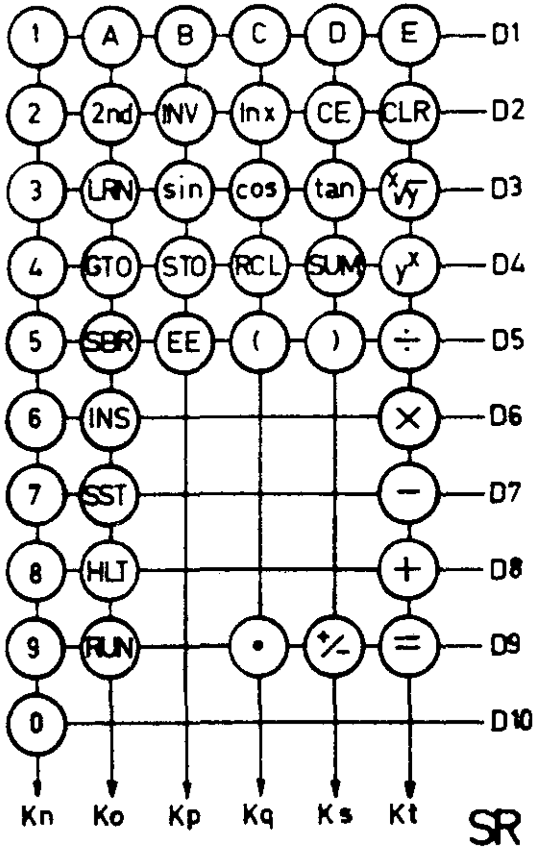
PROGRAM LISTING

Display		Key	Display		Key	Display		Key	Display		Key	Display		Key
000	46	*LBL	015	00	0	030	01	1	045	00	0	060	01	1
001	11	A	016	04	4	031	08	8	046	04	4	061	01	1
002	65	X	017	65	X	032	65	X	047	55	÷	062	09	9
003	43	RCL	018	43	RCL	033	43	RCL	048	00	0	063	00	0
004	01	1	019	01	1	034	00	0	049	02	2	064	48	*EXC
005	05	5	020	09	9	035	04	4	050	00	0	065	00	0
006	85	+	021	85	+	036	75	-	051	00	0	066	05	5
007	42	STO	022	43	RCL	037	43	RCL	052	95	=	067	55	÷
008	00	0	023	00	0	038	00	0	053	57	*fix	068	43	RCL
009	01	1	024	02	2	039	01	1	054	05	5	069	00	0
010	43	RCL	025	95	=	040	95	=	055	40	* x^2	070	08	8
011	00	0	026	42	STO	041	42	STO	056	44	SUM	071	42	STO
012	03	3	027	00	0	042	00	0	057	00	0	072	00	0
013	95	=	028	03	3	043	02	2	058	05	5	073	00	0
014	42	STO	029	43	RCL	044	43	RCL	059	59	*dsz	074	95	=

Display		Key	Display		Key	Display		Key	Display		Key	Display		Key
075	42	STO	105	55	÷	135	95	=	165	01	1	195	95	=
076	00	0	106	93	· o	136	34	tan	166	05	5	196	42	STO
077	06	6	107	02	2	137	42	STO	167	01	1	197	01	1
078	01	1	108	22	INV	138	01	1	168	75	−	198	08	8
079	44	SUM	109	28	*log	139	01	1	169	43	RCL	199	01	1
080	00	0	110	95	=	140	55	÷	170	01	1	200	00	0
081	09	9	111	12	B	141	01	1	171	01	1	201	55	÷
082	43	RCL	112	43	RCL	142	00	0	192	40	*x^2	202	43	RCL
083	00	0	113	00	0	143	85	+	173	95	=	203	00	0
084	09	9	114	06	6	144	01	1	174	55	÷	204	07	7
085	22	INV	115	57	*fix	145	85	+	175	43	RCL	205	95	=
086	90	*ifzro	116	05	5	146	43	RCL	176	01	1	206	57	*fix
087	01	1	117	98	*prt	147	01	1	177	03	3	207	00	0
088	01	1	118	56	*rtn	148	01	1	178	65	X	208	52	EE
089	02	2	119	43	RCL	149	40	*x^2	179	02	2	209	22	INV
090	43	RCL	120	00	0	150	95	=	180	95	=	210	52	EE
091	00	0	121	05	5	151	42	STO	181	42	STO	211	42	STO
092	07	7	122	55	÷	152	01	1	182	01	1	212	00	0
093	57	*fix	123	43	RCL	153	03	3	183	09	9	213	08	8
094	03	3	124	00	0	154	20	*1/x	184	43	RCL	214	42	STO
095	52	EE	125	08	8	155	65	X	185	01	1	215	00	0
096	03	3	126	95	=	156	43	RCL	186	01	1	216	00	0
097	22	INV	127	56	*rtn	157	01	1	187	55	÷	217	01	1
098	52	EE	128	46	*LBL	158	01	1	188	05	5	218	00	0
099	44	SUM	129	12	B	159	55	÷	189	55	÷	219	94	+/−
100	00	0	130	42	STO	160	01	1	190	43	RCL	220	42	STO
101	06	6	131	00	0	161	00	0	191	01	1	221	00	0
102	43	RCL	132	07	7	162	94	+/−	192	03	3	222	09	9
103	00	0	133	65	X	163	95	=	193	75	−	223	56	*rtn
104	07	7	134	59	*π	164	42	STO	194	01	1			

* denotes 2nd function key.

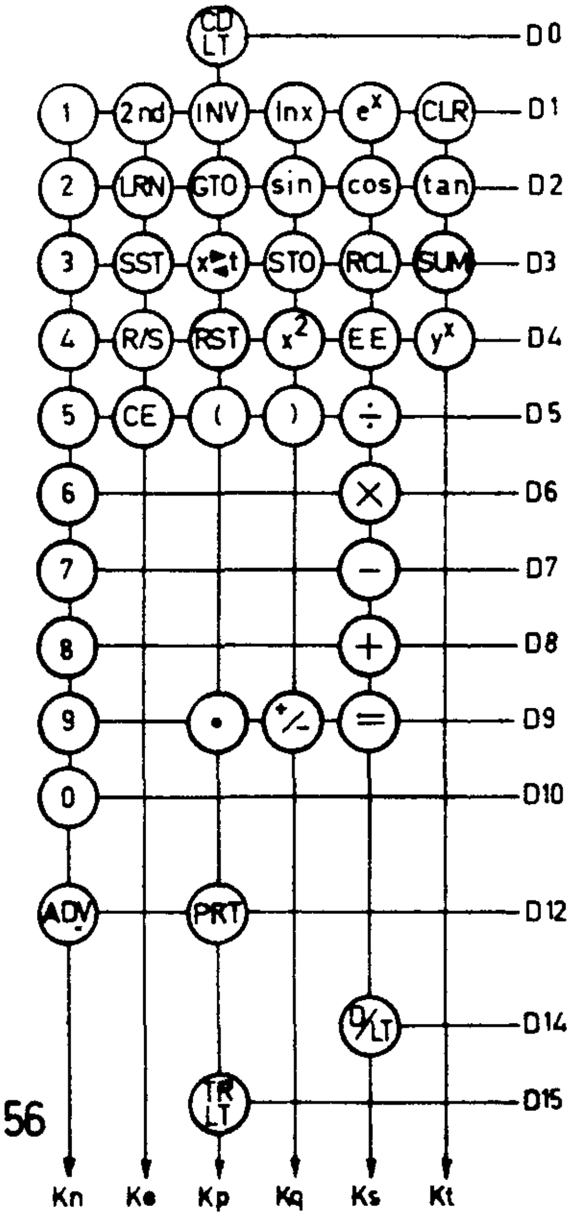
APPENDIX 10



	Kn	Ko	Kp	Kq	Ks	Kt
D1	1	A	B	C	D	E
D2	2	2nd	INV	lnx	CE	CLR
D3	3	LRN	sin	cos	tan	$\sqrt[x]{y}$
D4	4	GTO	STO	RCL	SUM	y^x
D5	5	SBR	EE	(	)	÷
D6	6	INS				×
D7	7	SST				−
D8	8	HLT				+
D9	9	RUN		•	+/−	=
D10	0					

SR 52

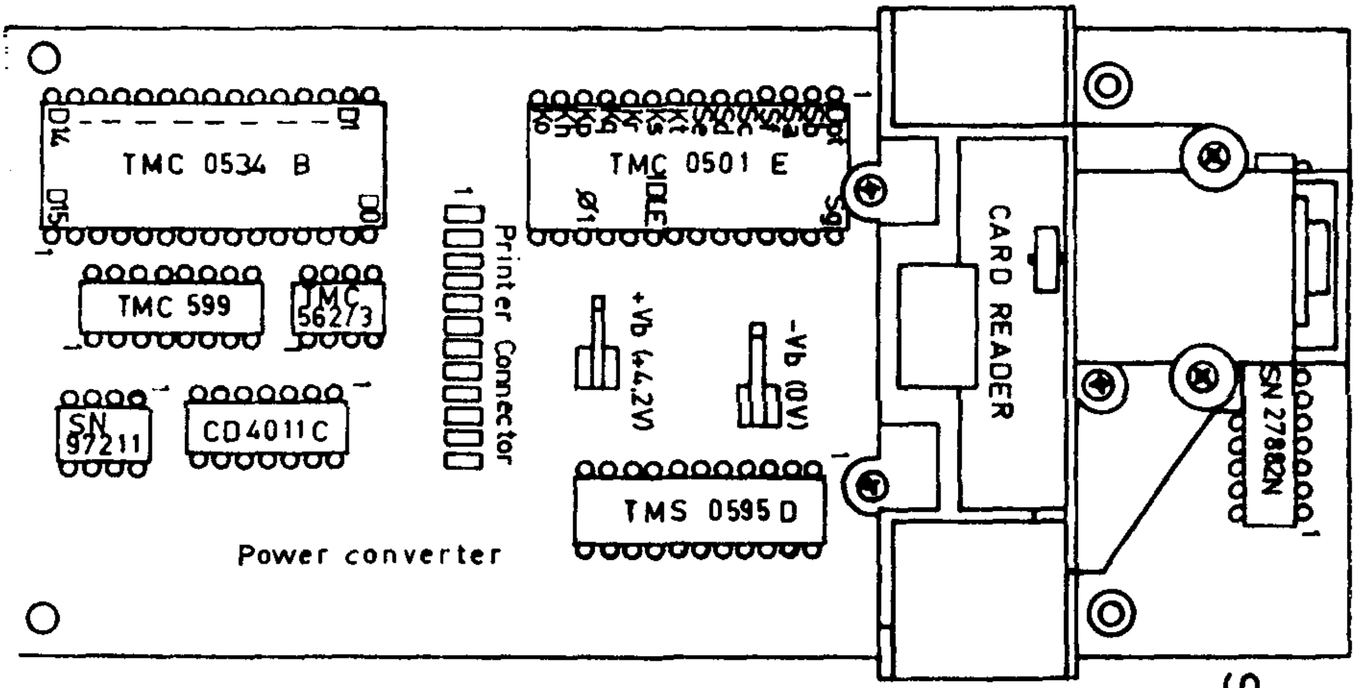
Keyboard Connections



	Kn	Ko	Kp	Kq	Ks	Kt
D0			CD LT			
D1	1	2nd	INV	lnx	e^x	CLR
D2	2	LRN	GTO	sin	cos	tan
D3	3	SST	x⇌t	STO	RCL	SUM
D4	4	R/S	RST	x^2	EE	y^x
D5	5	CE	(	)	÷	
D6	6				×	
D7	7				−	
D8	8				+	
D9	9		•	+/−	=	
D10	0					
D12	ADV		PRT			
D14					D/LT	
D15			TR LT			

SR 56

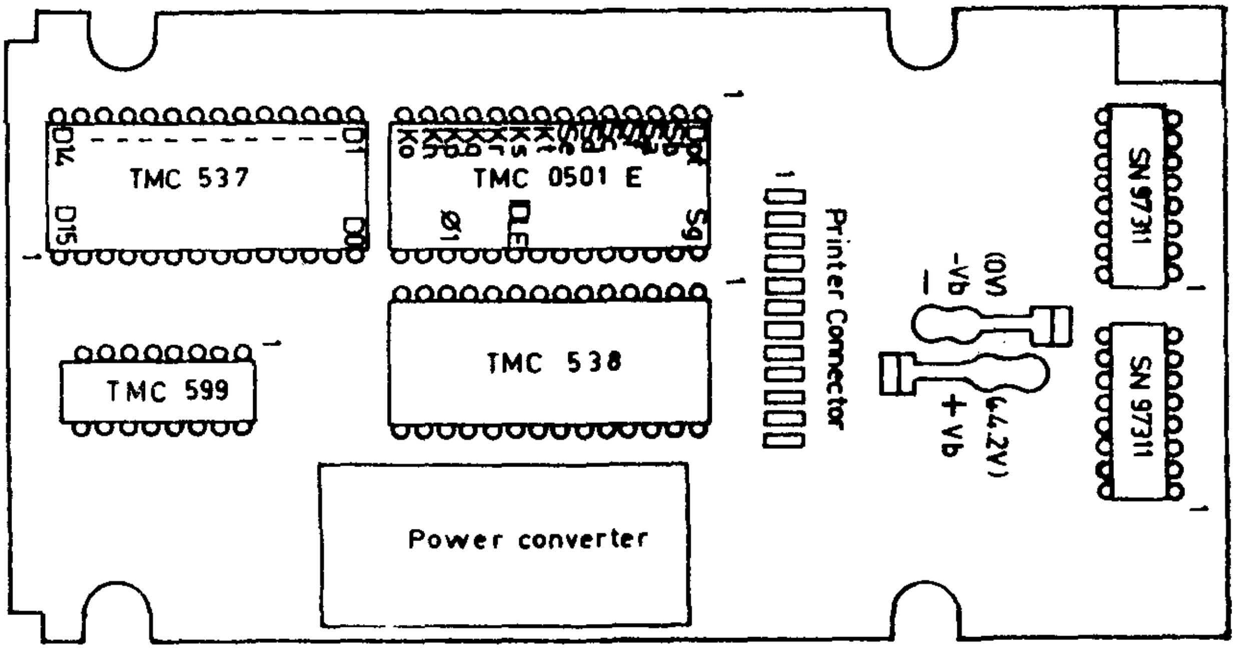
APPENDIX 11



SR52

PIN ASSIGNMENS

SR 56



REFERENCES

1 G. Verkroost, to be published.

2 H.W. Schüssler, Digitale Systeme zur Signalverarbeitung, Springer 1973.

3 W.D. Stanley, Digital Signal Processing, Prentice Hall 1975.

4 L.P. Huelsman (Ed.) Active Filters: Lumped, Distributed, Integrated, Digital and Parametric, McGraw Hill 1970.

5 Datel Systems Inc. Engineering Product Handbook, Gold Book 1976.

6 Howard W. Sams & Co Inc., Reference Data for Radio Engineers, 5th ed. , ITT 1972.