

AD-A042 631

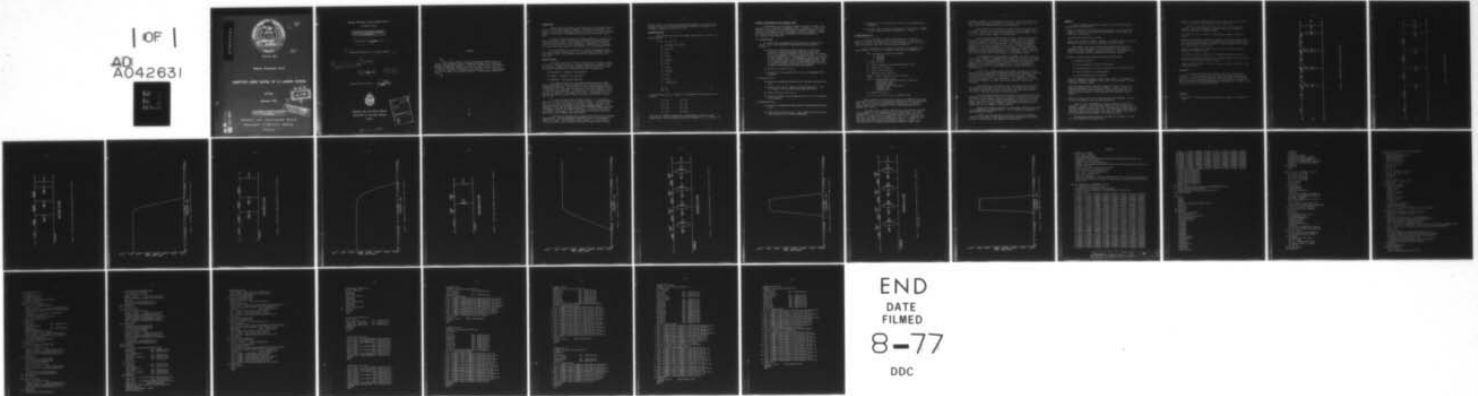
DEFENCE RESEARCH ESTABLISHMENT PACIFIC VICTORIA (BRIT--ETC F/G 9/5  
COMPUTER-AIDED DESIGN OF LC LADDER FILTERS.(U)

UNCLASSIFIED

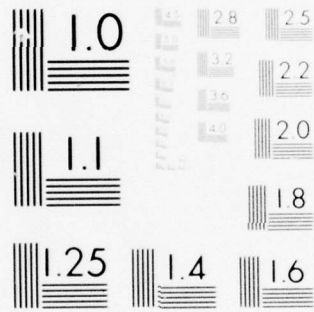
DEC 76 R C WEIR  
DREP-TM-76-10

NL

| OF |  
40  
A042631



END  
DATE  
FILMED  
8-77  
DDC



MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

ADA 042631



3

*[Handwritten signature]*

Victoria, B.C.

Technical Memorandum 76-10

# COMPUTER-AIDED DESIGN OF LC LADDER FILTERS

R.C. Weir

December 1976

DDC  
RECEIVED  
AUG 10 1977  
C

*This information is furnished with the express understanding that proprietary and patent rights will be protected.*

**DISTRIBUTION STATEMENT A**  
Approved for public release;  
Distribution Unlimited

ADJ NO. —  
DDC FILE COPY

Research and Development Branch  
Department of National Defence  
Canada

DEFENCE RESEARCH ESTABLISHMENT PACIFIC  
VICTORIA, B.C.

This information is furnished with the express understanding that proprietary and patent rights will be protected.

9 Technical Memorandum 76-10

6 COMPUTER-AIDED DESIGN OF LC LADDER FILTERS

14 DREP-TM-76-10  
10 R.C. Weir

11 December 1976

12 34p.

Approved for distribution

*J. Flynn*  
CDREP



RESEARCH AND DEVELOPMENT BRANCH  
DEPARTMENT OF NATIONAL DEFENCE  
CANADA

ACCESSION for	
NTIS	White Section <input checked="" type="checkbox"/>
DDC	Buff Section <input type="checkbox"/>
UNANNOUNCED	
JUSTIFICATION	
BY DISTRIBUTION/AVAILABILITY CODES	
Dist.	SP. CVAL
A	

403 246

13

## ABSTRACT

↙ This report contains a Fortran program that enables the user to produce, from published tables of normalized element values, the circuit diagram and amplitude response of a desired LC ladder filter. Butterworth, Thomson and Chebyshev characteristics are included, and the filter may be low-pass, high-pass or bandpass of any order from 1 to 8. Some commonly used sets of tables are included internally. Several design examples are given. ↗

INTRODUCTION

Though active RC networks are now popular in filter design, there is still a place for the passive LC variety. The latter is invaluable, for example, in situations requiring a minimum use of power, such as an installation at a remote site or an underwater location with a long cable link to shore.

Tables have been published by Weinberg (Reference 1) that give the normalized element values for Butterworth, Chebyshev and Thomson (Bessel) filters of various orders. This information is sufficient to enable the user to design filters with little difficulty. However, use of a computer can further reduce the amount of work involved; to a bare minimum in fact.

In the appendix is the listing of a Fortran IV computer program that yields circuit diagram and plot of the amplitude-vs-frequency response of a desired filter. The required input data are entered on one or two DATA cards.

GENERAL REMARKS

The general form of the filters considered in this report is shown in Figure 1. This network may be a low-pass, high-pass or bandpass filter. It can have any number of reactive elements (order of filter) up to eight. The response characteristic is selected from the following group:

- (a) Butterworth - maximally flat magnitude.
- (b) Thomson - maximally flat time delay.
- (c) Chebyshev - equal-ripple magnitude.

The Weinberg tables list the element values (in units of ohms, henrys and farads) for low-pass filters that are normalized with respect to frequency and impedance. For each type of filter, these normalized values are grouped in sets, each under the heading of  $r$ , which represents the ratio of input and load resistances. In each set are ten rows of element values, one for each value of  $n$ , the order of the filter. The maximum value of  $n$  used by the program is eight, though it would not be difficult to extend this to ten.

The basic low-pass filter is shown in Figure 2. In normalized form,  $R_B = 1.0$  ohm;  $R_A$  may be 0.0 ohm (short circuit) or a value determined by  $r$ . If  $n$  is even,  $r = R_A/R_B$ ; if  $n$  is odd,  $r = R_B/R_A$ . The filter configuration may be as shown ( $n=8$ ) or have fewer reactive elements. It is important to note that the labelling order of the element values taken from the Weinberg tables is reversed in this report. In a fifth-order filter, for example, L5 becomes L1. This should cause no difficulty, especially after two or three examples have been studied.

A few words of explanation are required with regard to the frequency normalization. Both Butterworth and Chebyshev filters are normalized to make  $\omega_c = 1.0$  rps. For the Thomson filter, however,  $\omega_c = 1/t_0$ , where  $t_0$  is the time delay at zero frequency (different for each order of filter).

As will be seen, the computer program arranges matters so that all of the designed filters have a specified amplitude response at the design cutoff frequency. Usually this level is -3.0 dB.

### TRANSFER FUNCTION

The transfer function of the general network shown in Figure 1 is easily obtained:

$$\text{Let } E_2 = 1.0, R_B = 1.0$$

$$I_2 = (Y_8 + 1/R_B) E_2 = Y_8 + 1.0$$

$$V_c = Z_7 I_2 + 1.0$$

$$I_h = Y_6 V_c$$

$$I_g = I_h + I_2$$

$$V_b = Z_5 I_g + V_c$$

$$I_f = Y_4 V_b$$

$$I_e = I_f + I_g$$

$$V_a = Z_3 I_e + V_b$$

$$I_d = Y_2 V_a$$

$$I_1 = I_d + I_e$$

$$E_1 = (R_A + Z_1) I_1 + V_a$$

$$\frac{E_2}{E_1} = \frac{1}{E_1} \tag{1}$$

For the low-pass filter of Figure 2, the impedances and admittances are as follows:

$$\begin{array}{ll} Z_1 = L_1 s & Y_2 = C_2 s \\ Z_3 = L_3 s & Y_4 = C_4 s \\ Z_5 = L_5 s & Y_6 = C_6 s \\ Z_7 = L_7 s & Y_8 = C_8 s \end{array}$$

Note that the transfer function (1) is applicable to filters of lesser orders than 8. For a sixth-order filter, for example,  $Z_7 = 0.0$  and  $Y_8 = 0.0$ .

## FREQUENCY TRANSFORMATION AND FREQUENCY SHIFT

Transformation of the frequency variable is used to convert a low-pass filter into a high-pass or bandpass filter. Obviously, no such transformation is necessary if a low-pass design is required; however, in some cases an extra frequency shift is essential, as will be explained. Though the actual transformations are not discussed here (see Reference 1) the consequent steps are given.

### (1) Low-Pass Filter

To convert from the normalized filter to one with a cutoff frequency  $\omega_c$  and a load resistance of the required value, proceed as follows:

- (a) Multiply all normalized element values by  $\omega_k$ , the radian frequency at which the amplitude response is down 3 dB. (This step is necessary because of the way in which the Thomson filter is normalized. For the Butterworth filter,  $\omega_k = 1.0$  rps. For the Chebyshev filter, the amplitude at the corner frequency is down an amount equal to the declared passband ripple. Thus,  $\omega_k$  is slightly greater than 1.0 rps for ripples less than 3 dB. A little forethought is required in this connection, as will be explained in the section describing the program.)
- (b) Divide all L's and C's by  $\omega_c$ .
- (c) Multiply all L's and divide all C's by K, the impedance-level parameter.

### (2) High-pass Filter.

- (a) Obtain the normalized low-pass filter, including multiplication by  $\omega_k$ .
- (b) Replace each L with a C equal to  $1/L$  and replace each C with an L equal to  $1/C$ , the units being henrys and farads.
- (c) Divide every new L and C by  $\omega_c$ .
- (d) Multiply all new L's and divide all new C's by K, the impedance-level parameter.

### (3) Band-pass Filter.

- (a) Obtain the normalized low-pass filter, including multiplication by  $\omega_k$ .
- (b) Divide all L's and C's by  $\omega_c = \omega_2 - \omega_1$ , where  $\omega_1$  and  $\omega_2$  are the edge frequencies of the desired pass band.

- (c) Multiply all L's and divide all C's by K, the impedance-level parameter.
- (d) To each L add in series a capacitance  $C = 1/L\omega_0^2$ , where  $\omega_0 = \sqrt{\omega_1\omega_2}$ ; to each C add in parallel an inductance  $L = 1/C\omega_0^2$ .

#### PROGRAM DESCRIPTION

It is not intended to explain the program line by line, but rather to indicate the purpose of the various sections. It will help to know, for instance, that almost half of the main program and all of the sub-routines are used to draw the circuit diagram and label it.

Since a knowledge of the input data variables is important to an understanding of the rest of the program, these are listed immediately.

PASS - determines the kind of pass band.

- 1 - low-pass
- 2 - high-pass
- 3 - bandpass

N - order of filter

FC - cutoff frequency of low-pass or high-pass filter

F1 - lower corner frequency

F2 - higher corner frequency

K - impedance-level multiplier

DEBCOR - decibel response at cutoff frequency

SELECT - chooses element values

- 0 external list entered by means of DATA cards
- 1 Butterworth, RA=0
- 2 Butterworth, RA=1.0
- 3 Chebyshev, RA=0 (1-dB ripple)
- 4 Thomson, RA=0
- 5 Thomson, RA=1.0

R - resistance ratio, r.

$L_1, C_2, L_3, C_4, L_5, C_6, L_7, C_8$  - element values.

These variables, corresponding to those in the NAME LIST statement, are entered by means of DATA cards placed immediately before the END OF DATA card. The list is largely self-explanatory, but further clarification will be given during the remainder of the program description and the examples that follow.

The large block of numbers appearing after the NAME LIST statement in the program is a three-dimensional array called FILTER. This array contains eight rows (n=1 to 8) of normalized element values for each of five sets of low-pass filters, making forty rows in all. These sets have been put in the array for the convenience of the user. Included are the five cases given above opposite the variable SELECT. If SELECT = 0, the entire block is skipped and the external normalized element values

selected; if SELECT = 2 (for example), selection is made from FILTER of the row of normalized element values that will produce a Butterworth filter of order N with input resistance equal to load resistance.

There should be no difficulty in knowing which row has been chosen. N is easily picked out, for it equals the number of non-zero values given in the row. The sets are listed in the order shown opposite SELECT. As a check, the selected values appear in the print-out.

These normalized element values are used as a basis for the design of all the filters dealt with in this report, whether they are low-pass, high-pass or bandpass in nature. The choice of this option is indicated by setting PASS = 1, 2 or 3, respectively. Other required information includes FC, the cutoff frequency in the case of a low-pass or high-pass design, or F1 and F2 if the design is to be bandpass. The impedance-level parameter K is also necessary, as well as the resistance ratio r, which is associated with the chosen element values. If SELECT  $\neq$  0, r is set internally and is not required; nor is it necessary to assign element values.

The remaining piece of information is DEBCOR, the decibel response at the cutoff frequency. Although this is usually set equal to -3 dB, there is no reason - with one precaution - why it should not be some other value. The precaution is concerned with the undulating nature of the amplitude response in the pass band of a Chebyshev filter. If the magnitude of DEBCOR does not exceed the ripple magnitude, an erroneous result will be obtained for  $\omega_k$ .

Given a set of element values, the computer proceeds by a step-by-step calculation to determine the amplitude level at each frequency. When the level drops below DEBCOR the calculations cease and the radian frequency at that point is designated  $\omega_k$ . This result can be negated, if desired, by inserting after  $\omega_k = \omega$  a card reading  $\omega_k = 1.0$ . There may be occasions when it is desirable to do this, but for most designs it will not be necessary.

Having found  $\omega_k$  and knowing  $\omega_c$  and K, the program then calculates the component values (R's, L's, C's) for the actual filter in the manner described in the previous section of the report. The amplitude response of the filter is now calculated and the results plotted.

The considerable length of that section of the program dealing with drawing the circuit diagram is due to the number of possibilities that exist. The network may be low-pass, high-pass or bandpass; it may be of any order from 1 to 8, odd or even; the input resistance may be present or not; the element values may be given in henrys or millihenrys, microfarads or picofarads, with any of a number of decimal places.

Despite the length of this part of the program, inspection will show it to be segmented and straightforward. Once the pattern is recognized, understanding comes easily.

Finally, all one need know to obtain one of the filter designs covered in this report is what information must be supplied to the program on one or two DATA cards. Some examples are given below.

### EXAMPLES

Several examples are given to clarify the use of the program and to demonstrate its versatility.

1. Design a sixth-order, Butterworth low-pass filter to have a cutoff frequency of 9,500 Hz, with input and load resistances of 1,500 ohms and 6,000 ohms, respectively.

The DATA cards should read as follows:

PASS=1, N=6, FC=9500.0, K=6000.0, DEBCOR=-3.0, SELECT=0, R=0.25,  $L_1=0.0675$ , C2=3.1601, L3=0.3130, C4=6.4673, L5=0.4567, C6=6.3425, \*END\*.

The element values are obtained from the Butterworth tables in Reference 1 under the heading  $r = 1/4 (=1,500/6,000)$ . Note that the labelling has been reversed. Figure 3 gives the circuit diagram of the filter and Figure 4 shows the amplitude-vs-frequency response.

2. Design a low-pass filter with the following specifications:

- (a) minimum overshoot;
- (b) cutoff frequency of 250 Hz (-3 dB point);
- (c) amplitude fall-off to reach a maximum rate of 30 dB per octave.
- (d) input resistance to be 20,000 ohms;
- (e) load resistance to be 5,000 ohms.

Enter the following parameters on DATA cards: PASS=1, N=5, FC=250.0, K=5000.0, DEBCOR=-3.0, SELECT=0, R=0.25,  $L_1=0.2731$ , C2=0.0498, L3=1.2690, C4=0.1084, L5=2.8247, \*END\*

Note that the specification "minimum overshoot" dictates the use of the tables for maximally flat time delay (Thomson). Since  $n$  is odd,  $R = 5,000/20,000$  instead of the reciprocal. The circuit diagram for this design is given in Figure 5, while the amplitude response is shown in Figure 6.

3. Design a low-pass filter to the specifications given in Example 2 except for the interchange of input and load resistances.

Since  $n$  is odd and therefore  $r = R_p/R_L$ , use must be made of the reciprocity theorem. This means simply that the design proceeds as in Example 2, assuming the input and load resistances are as given there, and then the network is turned end for end. Lift the bottom of the 5,000-ohm resistance, making this the input, and ground the (former) input side of the 20,000-ohms resistance making it the load resistance.

The amplitude response should be the same as in Example 2, except for a much-reduced flat insertion loss.

4. Design a third-order, Butterworth high-pass filter with cutoff frequency of 500 Hz, input and load resistances to be 12,000 ohms.

Since  $r=1$ , the internal element values can be used by letting  $SELECT=2$ . The DATA card reads as follows:  $PASS=2$ ,  $N=3$ ,  $FC=500.0$ ,  $K=12000.0$ ,  $DEBCOR=-3.0$ ,  $SELECT=2$ ,  $*END*$

The network is given in Figure 7; the amplitude response is shown in Figure 8.

5. Design an eighth-order, Butterworth bandpass filter with corner frequencies at 450 Hz and 1,500 Hz. To minimize insertion loss, the input resistance should be omitted ( $RA=0$ , assuming a source impedance of zero ohms). The load resistance is 8,200 ohms.

Input the following data:  $PASS=3$ ,  $N=8$ ,  $F1=450.0$ ,  $F2=1500.0$ ,  $K=8200.0$ ,  $DEBCOR=-3.0$ ,  $SELECT=1$ ,  $*END*$

The network is shown in Figure 9; the amplitude response in Figure 10.

6. Change the previous design to that of a Chebyshev filter with 1-dB ripple. All other specifications remain the same.

Simply change to  $SELECT=3$ . Figure 11 shows the circuit and Figure 12 the amplitude response.

#### CONCLUSION

A Fortran program has been provided as an aid to the design of LC filters. Its use in conjunction with tables of element values given in Reference 1 facilitates quick designs of Butterworth, Thomson and Chebyshev filters of various orders and pass bands. For each design, a circuit diagram is drawn and the corresponding amplitude response curve is plotted. The entire process can be completed in minutes.

#### REFERENCE

1. Weinberg, L., "Network Analysis and Synthesis", Chapter 13, McGraw-Hill, 1962.

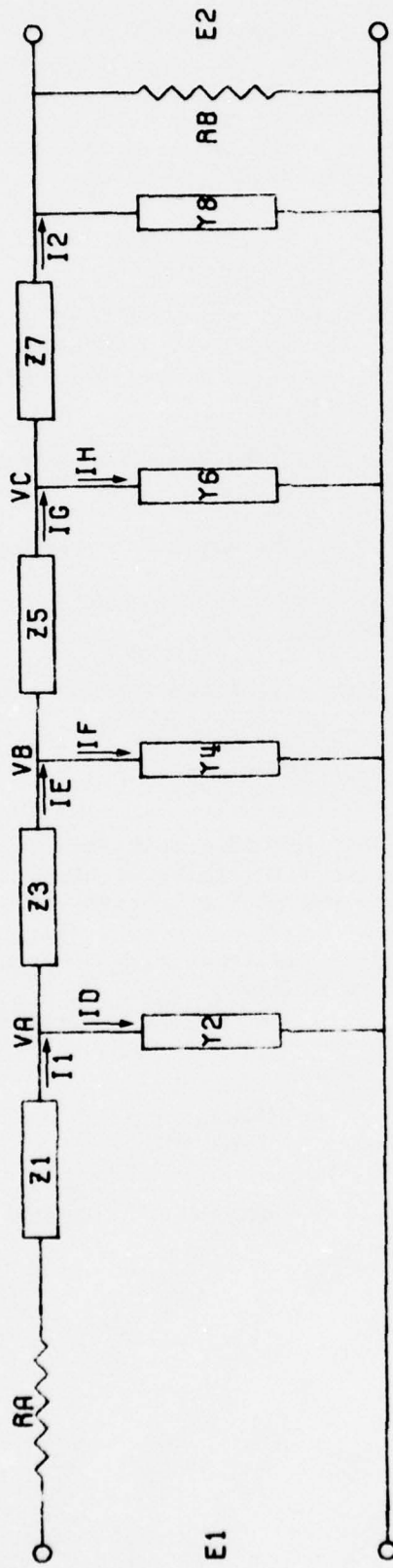


Figure 1. General form of ladder filter network.

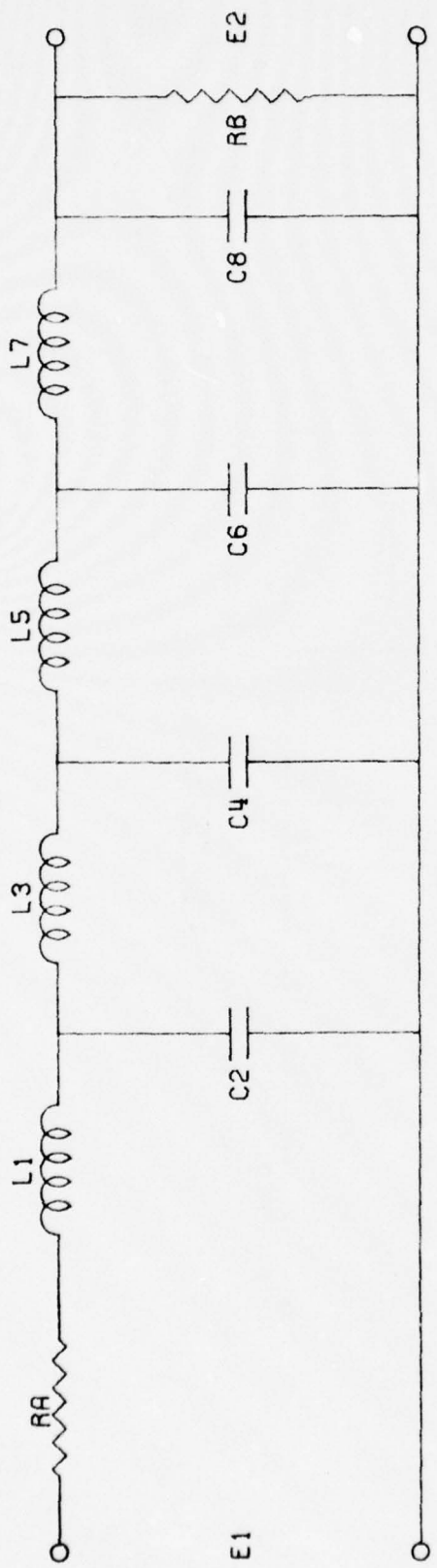
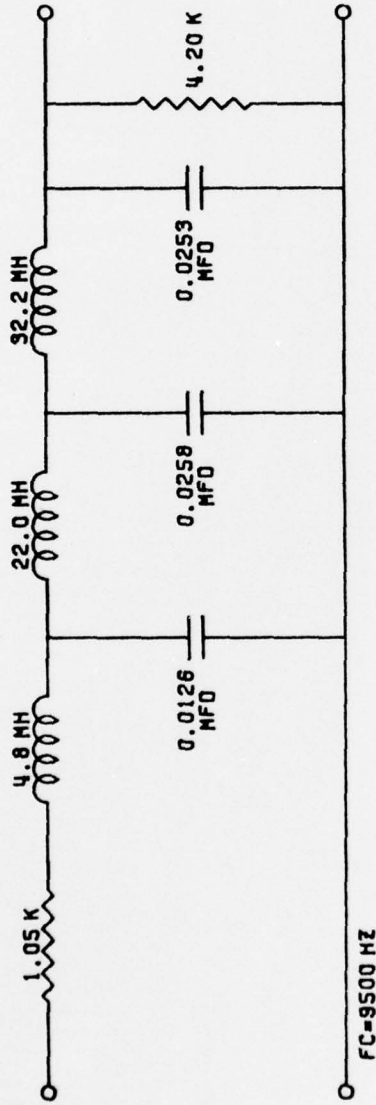


Figure 2. Basic low-pass filter.



LOW-PASS FILTER

Figure 3. Sixth-order, Butterworth low-pass filter ( $f_c = 9500$  Hz).

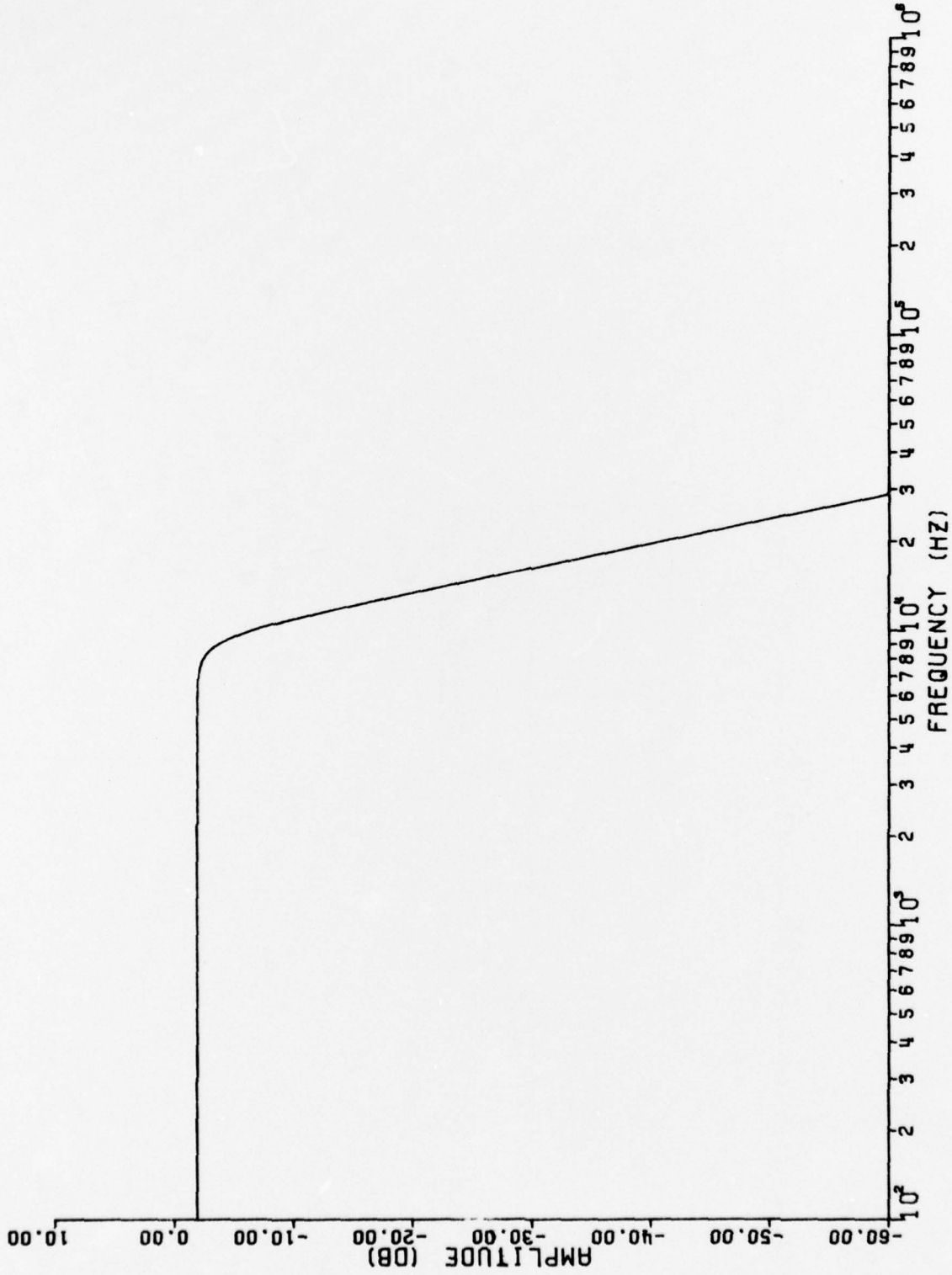
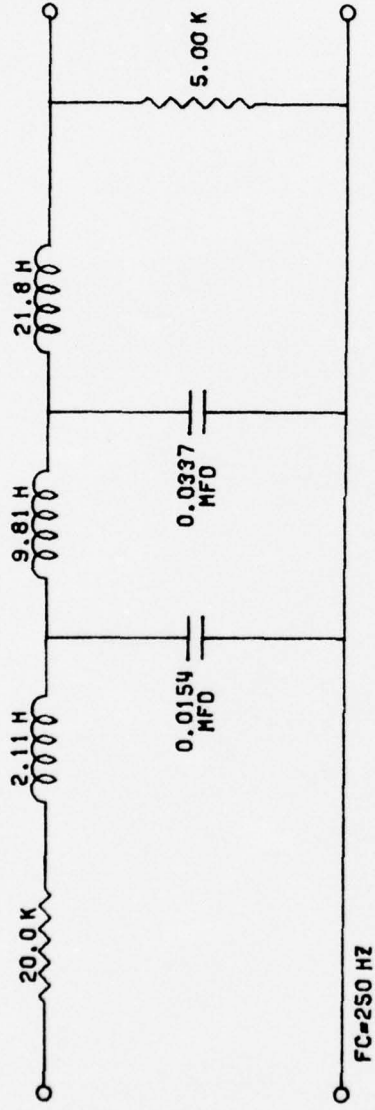


Figure 4. Amplitude response of filter shown in Figure 3.



LOW-PASS FILTER

Figure 5. Fifth-order, Thomson low-pass filter ( $f_c = 250$  Hz).

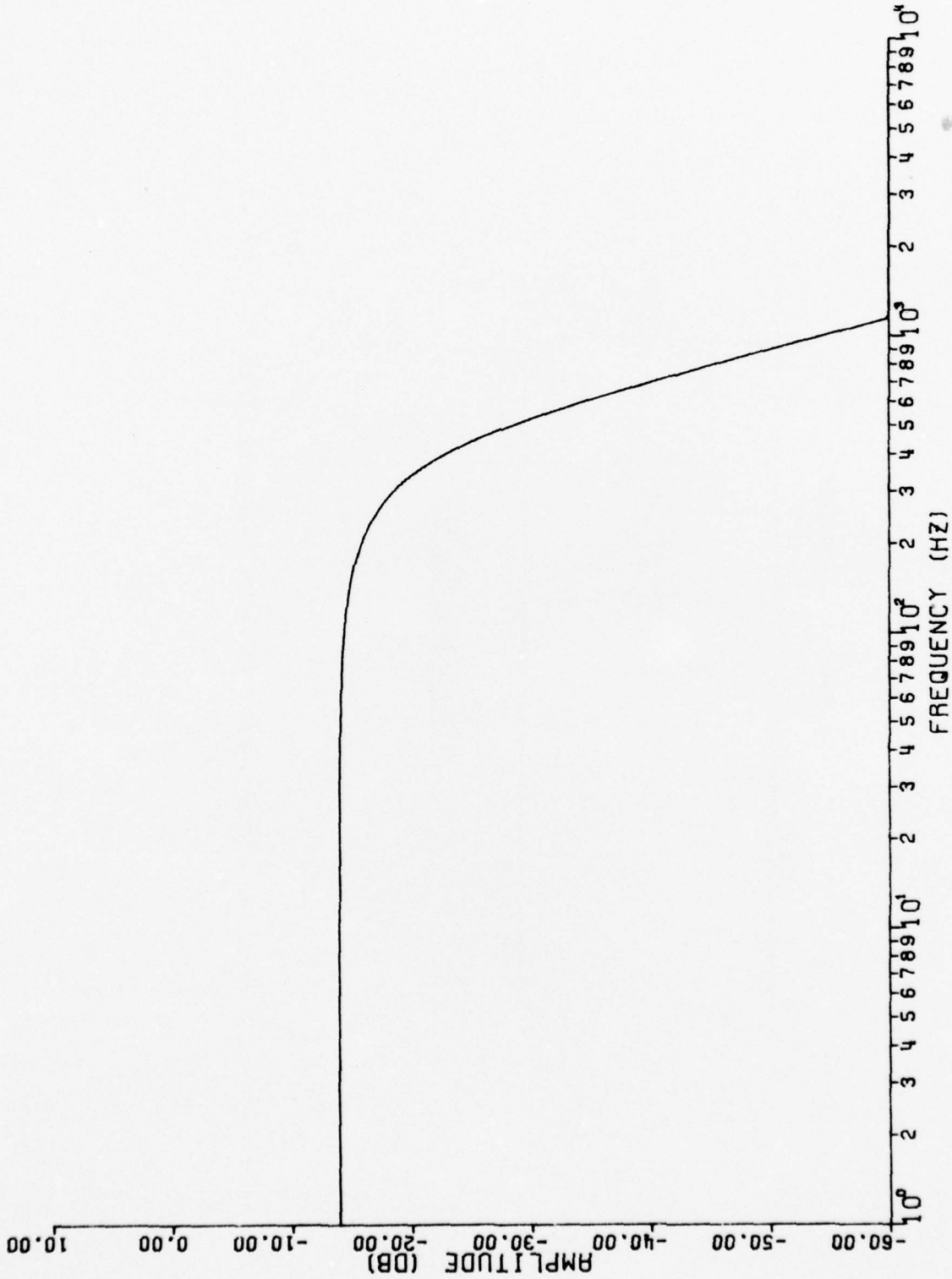


Figure 6. Amplitude response of filter shown in Figure 5.

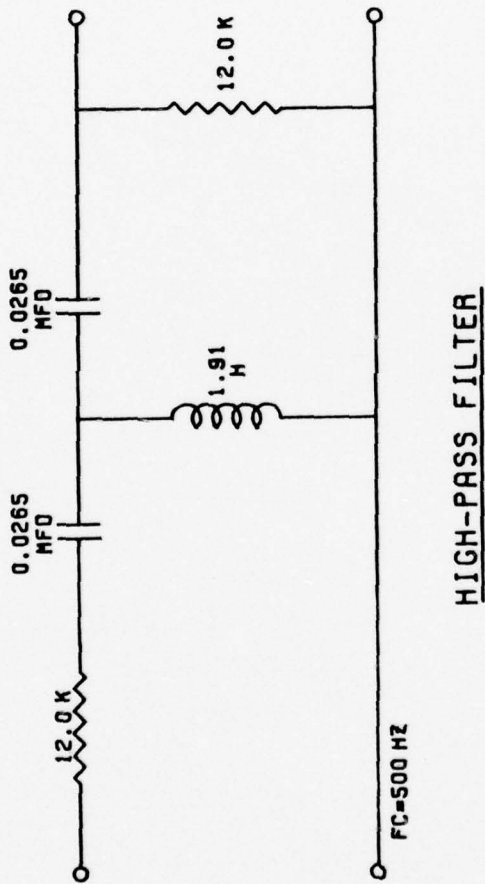


Figure 7. Third-order, Butterworth high-pass filter ( $f_c = 500$  Hz).

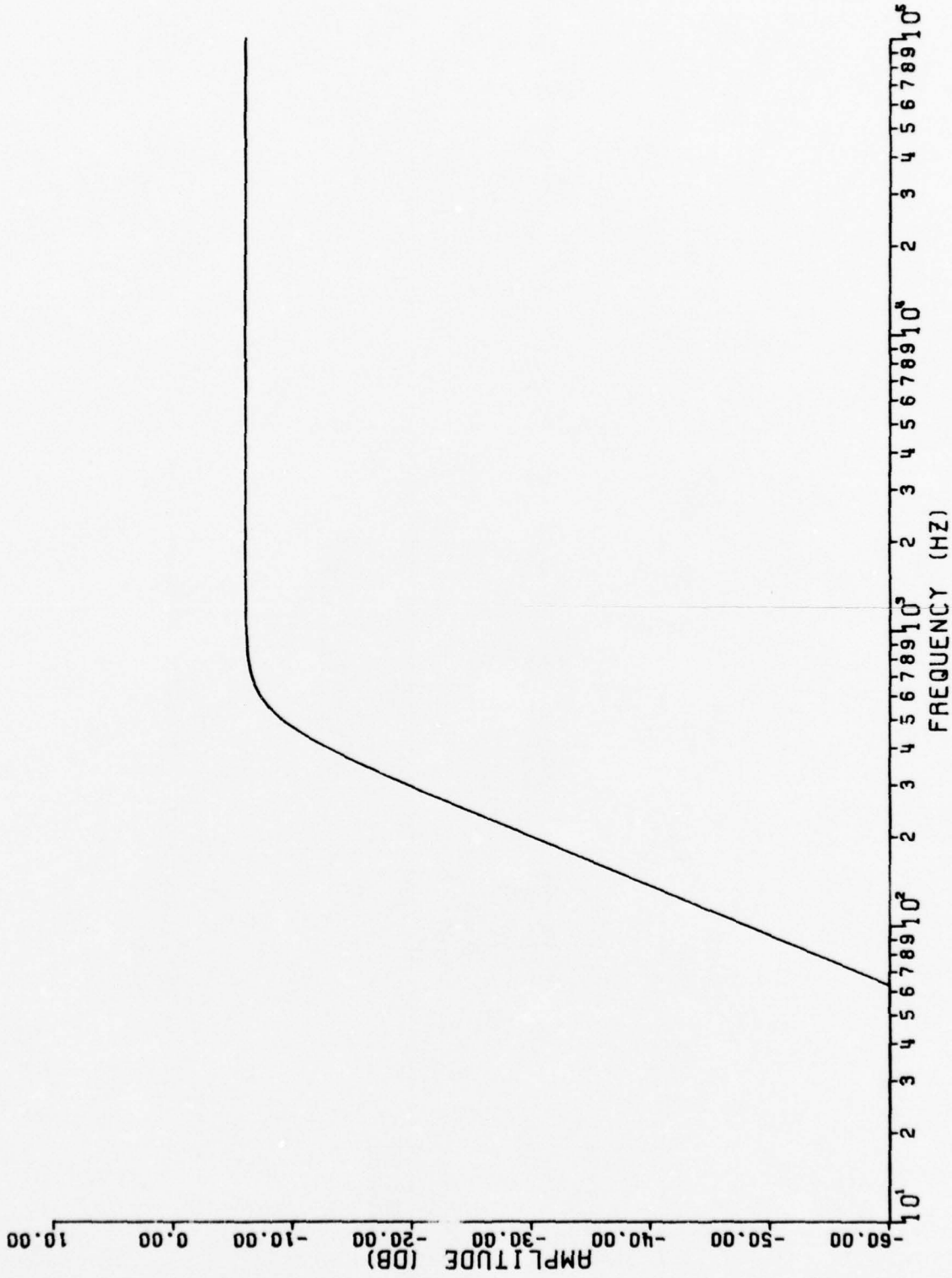


Figure 8. Amplitude response of filter shown in Figure 7.

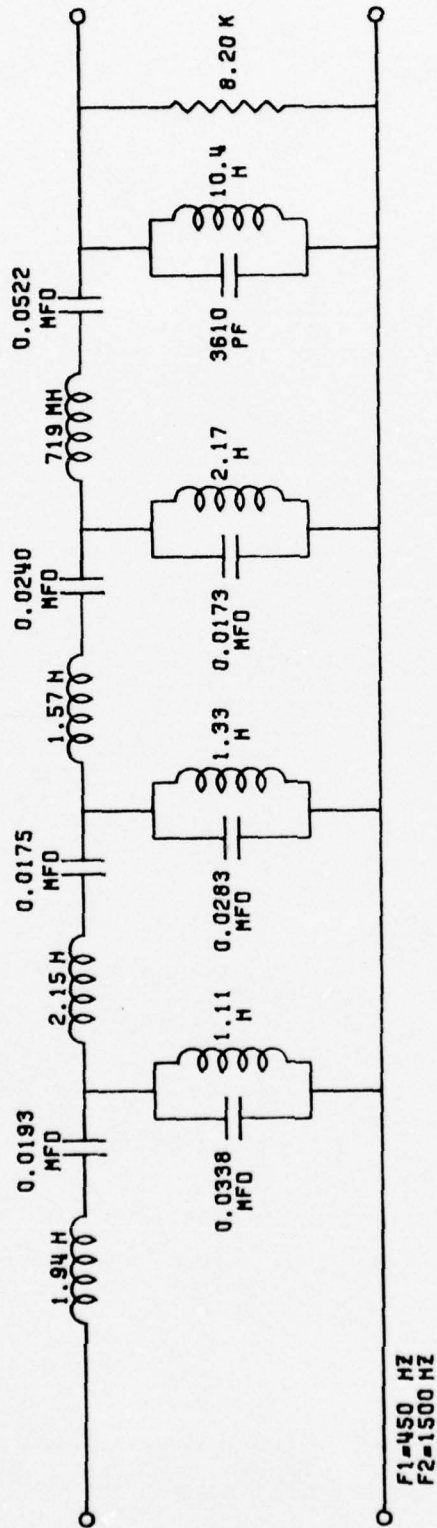


Figure 9. Eighth-order, Butterworth bandpass filter ( $f_1=450 \text{ Hz}$ ,  $f_2=1500 \text{ Hz}$ ).

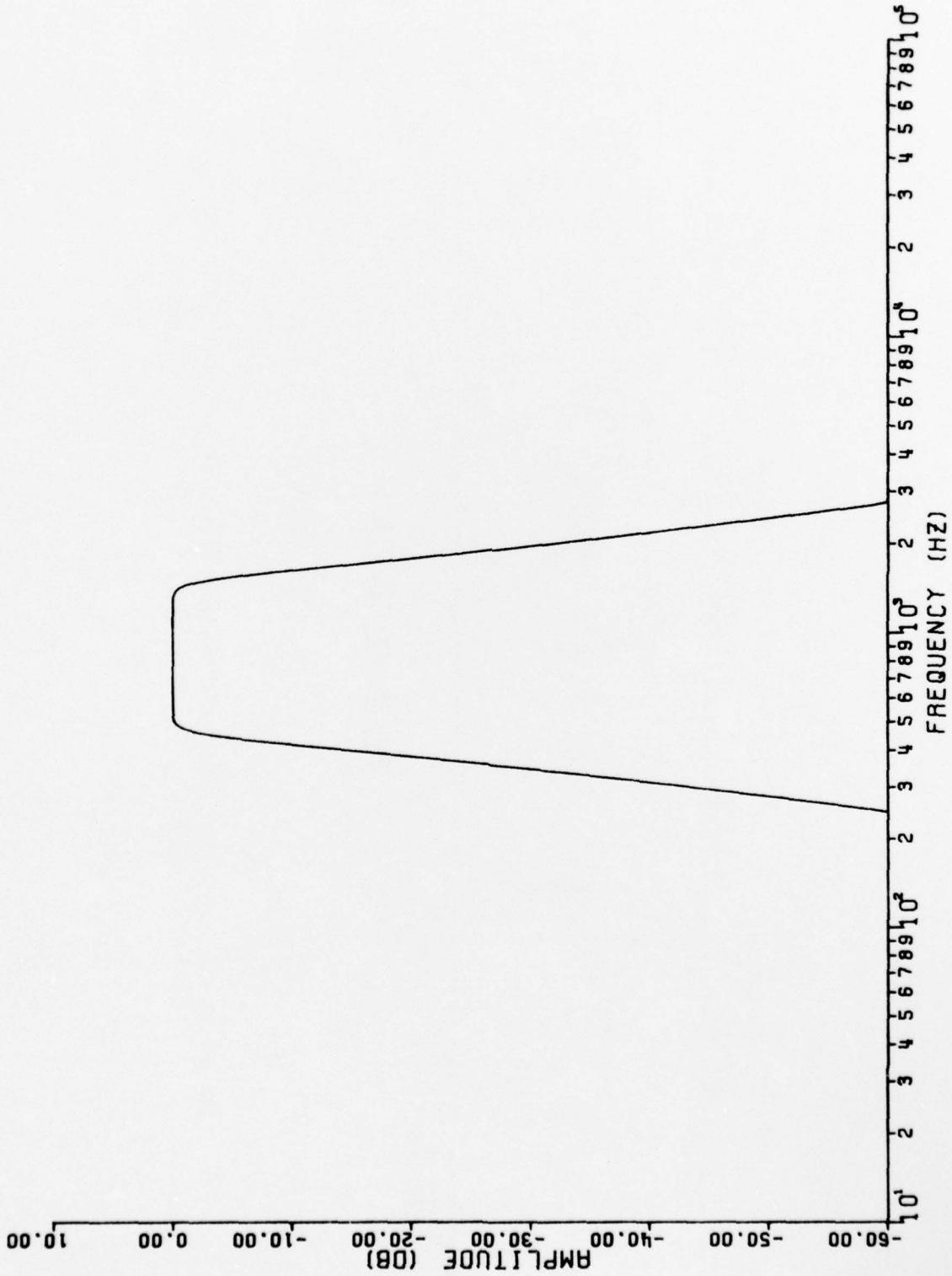


Figure 10. Amplitude response of filter shown in Figure 9.

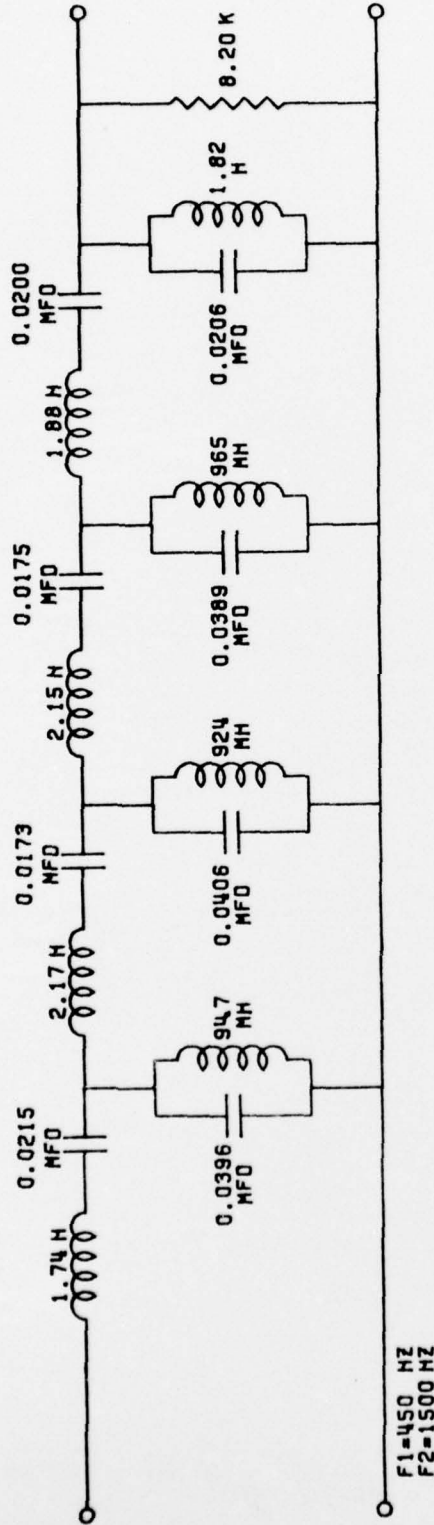


Figure 11. Eighth-order, Chebyshev bandpass filter ( $f_1=450$  Hz,  $f_2=1500$  Hz, 1-dB ripple).

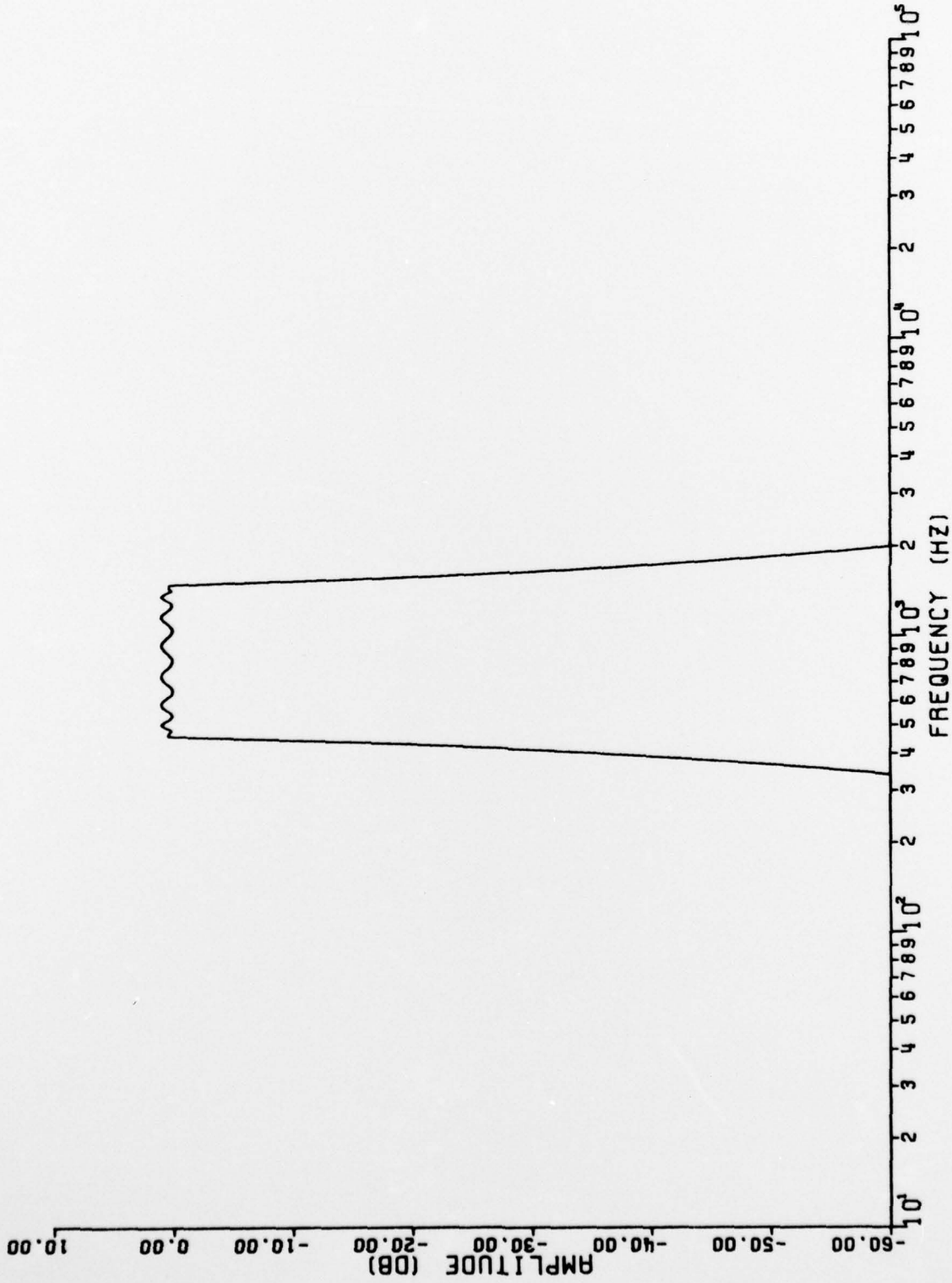


Figure 12. Amplitude response of filter shown in Figure 11.

APPENDIX

LC PASSIVE FILTERS

```

DATA NAME /IWETRI/
CALL PLTNAME(NAME,4)
DIMENSION FREQLG(1603),FREQ(1603),G(1603),DEREF(1603),EL(16)
DIMENSION FILTER(8,8,5)
COMMON X,Y,M,L,Q,T,J,ELEM(16)
COMPLEX S,Z1,Y2,Z3,Y4,Z5,Y6,Z7,Y8,I2,VC,IH,IG,VR,IF,IE,VA,ID,I1,E1
COMPLEX ZEL(16)
REAL L1,L2,L3,L4,L5,L6,L7,I8,LEN,K
INTEGER SELECT,U,P,PASS,Q,T,V
CALL LIMIT (0.0,40.0,0.0,10.5)
CALL PLOT (2.0,2.0,-3)
CALL FACTOR(0.8)
NAMELIST PASS,N,FC,F1,F2,K,DEBCOR,SELECT,R,L1,C2,L3,C4,L5,C6,L7,C8
FC=0.,F1=0.,F2=0.,L1=0.,C2=0.,L3=0.,C4=0.,I5=0.,C6=0.,L7=0.,C8=0.
INPUT(105)

```

SELECT NORMALIZED COMPONENT VALUES

```

IF (SELECT.EQ.0) GO TO 100
IF (SELECT.EQ.2.OR.SELECT.EQ.5)R=1.0
IF (SELECT.EQ.1.OR.SELECT.EQ.3.OR.SELECT.EQ.4)R=0.0
DATA FILTER/

```

>1.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.4142,	0.7071,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.5000,	1.3333,	0.5000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.5307,	1.5772,	1.0824,	0.3827,	0.0000,	0.0000,	0.0000,	0.0000,
>1.5451,	1.6944,	1.3820,	0.8944,	0.3090,	0.0000,	0.0000,	0.0000,
>1.5529,	1.7593,	1.5529,	1.2016,	0.7579,	0.2588,	0.0000,	0.0000,
>1.5576,	1.7988,	1.6588,	1.3972,	1.0550,	0.6560,	0.2225,	0.0000,
>1.5607,	1.8246,	1.7287,	1.5283,	1.2588,	0.9370,	0.5776,	0.1951,
>2.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.4142,	1.4142,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.0000,	2.0000,	1.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>0.7654,	1.8478,	1.8478,	0.7654,	0.0000,	0.0000,	0.0000,	0.0000,
>0.6180,	1.6180,	2.0000,	1.6180,	0.6180,	0.0000,	0.0000,	0.0000,
>0.5176,	1.4142,	1.9319,	1.9319,	1.4142,	0.5176,	0.0000,	0.0000,
>0.4450,	1.2470,	1.8019,	2.0000,	1.8019,	1.2470,	0.4450,	0.0000,
>0.3902,	1.1111,	1.6629,	1.9616,	1.9616,	1.6629,	1.1111,	0.3902,
>0.5088,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>0.9957,	0.9110,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.5088,	1.3332,	1.0118,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.2817,	1.9093,	1.4126,	1.0495,	0.0000,	0.0000,	0.0000,	0.0000,
>1.6652,	1.5908,	1.9938,	1.4441,	1.0674,	0.0000,	0.0000,	0.0000,
>1.3457,	2.0491,	1.6507,	2.0270,	1.4601,	1.0773,	0.0000,	0.0000,
>1.7118,	1.6485,	2.1192,	1.6736,	2.0437,	1.4694,	1.0832,	0.0000,
>1.3691,	2.0922,	1.7021,	2.1453,	1.6850,	2.0537,	1.4751,	1.0872,
>1.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>1.0000,	0.3333,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>0.8333,	0.4800,	1.6667,	0.0000,	0.0000,	0.0000,	0.0000,	0.0000,
>0.7101,	0.4627,	0.2899,	0.1000,	0.0000,	0.0000,	0.0000,	0.0000,
>0.6231,	0.4215,	0.3103,	0.1948,	0.0667,	0.0000,	0.0000,	0.0000,
>0.5595,	0.3821,	0.3005,	0.2246,	0.1400,	0.0476,	0.0000,	0.0000,

```
>0.5111, 0.3487, 0.2827, 0.2288, 0.1704, 0.1055, 0.0357, 0.0000,  
>0.4732, 0.3212, 0.2639, 0.2227, 0.1806, 0.1338, 0.0823, 0.0278,  
>2.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000,  
>0.4226, 1.5574, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000,  
>0.1922, 0.5528, 1.2550, 0.0000, 0.0000, 0.0000, 0.0000, 0.0000,  
>0.1104, 0.3181, 0.5116, 1.0598, 0.0000, 0.0000, 0.0000, 0.0000,  
>0.0718, 0.2090, 0.3312, 0.4577, 0.9303, 0.0000, 0.0000, 0.0000,  
>0.0505, 0.1480, 0.2364, 0.3158, 0.4116, 0.8377, 0.0000, 0.0000,  
>0.0375, 0.1104, 0.1778, 0.2378, 0.2944, 0.3774, 0.7677, 0.0000,  
>0.0289, 0.0855, 0.1387, 0.1867, 0.2297, 0.2735, 0.3446, 0.7125,
```

```
L1=FILTER(1,N,SELECT)  
C2=FILTER(2,N,SELECT)  
L3=FILTER(3,N,SELECT)  
C4=FILTER(4,N,SELECT)  
L5=FILTER(5,N,SELECT)  
C6=FILTER(6,N,SELECT)  
L7=FILTER(7,N,SELECT)  
C8=FILTER(8,N,SELECT)
```

100 CONTINUE

```
WRITE(108,91)PASS,N,FC,F1,F2,K,DEBCOR,SELECT,R
```

91 FORMAT(2I6,5F10.2,I6,F7.3)

```
WRITE(108,92)L1,C2,L3,C4,L5,C6,L7,C8
```

92 FORMAT(8F8.4)

FIND WK

```
RR=1.0
```

```
P=N/2
```

```
E=2*P
```

```
IF(E.EQ.N.OR.R.EQ.0.0)GO TO 16
```

```
RA=RB/R
```

```
GO TO 29
```

16 RA=RB\*R

29 CONTINUE

```
FREQLG(1)=-2.79818
```

```
DA 4 I=1,1601
```

```
FREQ(I)=10.0**FREQLG(I)
```

```
W=6.2832*FREQ(I)
```

```
S=CMPLX(0.0,W)
```

```
Z1=L1*S
```

```
Z2=C2*S
```

```
Z3=L3*S
```

```
Z4=C4*S
```

```
Z5=L5*S
```

```
Z6=C6*S
```

```
Z7=L7*S
```

```
Z8=C8*S
```

```
I2=Y8+i.0/RR
```

```
VC=Z7*I2+1.0
```

```
IH=Y6*VC
```

```
IG=IH+I2
```

```
VR=Z5*IG+VC
```

```
IF=Y4*VR
```

```
IF=IF+IG
```

```
VA=Z3*IF+VR
```

```
ID=Y2*VA
I1=ID+IE
F1=(RA+Z1)*I1+VA
G(I)=CABS((RA+RB)/(RB*E1))
DFBEE(I)=20.0*ALOG10(G(I))
IF(DFBEE(I).LE.DERCOR) GO TO 5
FREQLG(I+1)=FREQLG(I)+0.0025
4 CONTINUE
5 CONTINUE
WK=W
```

ATTAIN ACTUAL COMPONENT VALUES

```
EL(1)=L1; EL(2)=C2
EL(3)=L3; EL(4)=C4
EL(5)=L5; EL(6)=C6
EL(7)=L7; EL(8)=C8
DO 6 J=1,16
ELEM(J)=0.0
6 CONTINUE
WC=6.2832*FC
V1=ALOG10(FC)
V2=AINT(ALOG10(FC))
V3=V1-V2
IF(V3.LE.0.5)FREQLG(1)=V2-2.0
IF(V3.GT.0.5)FREQLG(1)=V2-1.0
RA=RA*K; RB=RB*K
IF(.NOT.(PASS.EQ.2)) GO TO 9
DO 7 J=1,4
IF(EL(2*J-1).EQ.0.0)GO TO 8
EL(2*J-1)=1.0/(EL(2*J-1)*WK*WC*K)
8 IF(EL(2*J).EQ.0.0)GO TO 7
EL(2*J)=K/(EL(2*J)*WK*WC)
7 CONTINUE
GO TO 14
9 IF(PASS.EQ.1) GO TO 10
WC=6.2832*(F2-F1)
WOSQ=39.4784*F1*F2
FO=SQRT(F1*F2)
V1=ALOG10(FO)
V2=AINT(ALOG10(FO))
V3=V1-V2
IF(V3.LE.0.5)FREQLG(1)=V2-2.0
IF(V3.GT.0.5)FREQLG(1)=V2-1.0
10 DO 11 J=1,4
EL(2*J-1)=EL(2*J-1)*K*WK/WC
EL(2*J)=EL(2*J)*WK/(K*WC)
11 CONTINUE
IF(PASS.EQ.1) GO TO 14
DO 12 J=1,8
IF(EL(J).EQ.0.0)GO TO 13
EL(J+8)=1.0/(EL(J)*WOSQ)
GO TO 12
13 EL(J+8)=0.0
12 CONTINUE
```

CALCULATE AND PLOT AMPLITUDE RESPONSE

```
14 DO 15 I=1,401
    FREQ(I)=10.0**FREQLG(I)
    W=6.2832*FREQ(I)
    S=CMPLX(0.0,W)
    GO TO (1,2,3)PASS
  1 DO 51 J=1,8
    ZFL(J)=EL(J)*S
51 CONTINUE
    GO TO 19
  2 DO 17 J=1,8
    IF(EL(J).EQ.0.0)GO TO 17
    ZFL(J)=1.0/(EL(J)*S)
17 CONTINUE
    GO TO 19
  3 DO 18 J=1,8
    ZFL(J)=EL(J)*(S+WOSQ/S)
18 CONTINUE
19 I2=ZEL(8)+1.0/RB
    VC=ZEL(7)*I2+1.0
    IH=ZEL(6)*VC
    IG=IH+I2
    VR=ZEL(5)*IG+VC
    IF=ZEL(4)*VR
    IF=IF+IG
    VA=ZEL(3)*IF+VR
    ID=ZEL(2)*VA
    I1=ID+IE
    E1=(RA+ZEL(1))*I1+VA
    G(I)=CABS(1.0/F1)
    DEBEE(I)=20.0*ALOG10(G(I))
    WRITE(108,20)FREQLG(I),FREQ(I),G(I),DEBEE(I)
20 FORMAT(4F16.6)
    IF(DEBEE(I).LE.-60.0)DEBEE(I)=-60.0
    FREQLG(I+1)=FREQLG(I)+0.01
15 CONTINUE
    FREQLG(402)=FREQLG(1); FREQLG(403)=0.4
    FREQ(402)=10.0**FREQLG(402); FREQ(403)=0.4
    DEBEE(402)=-60.0; DEBEE(403)=10.0
    CALL LINE(FREQLG,DEBEE,401,1,0,0)
    CALL AXIS(0.,0.,'AMPLITUDE (DB)',14,7.,90.,DEBEE(402),DEBEE(403))
    CALL LGAXS(0.,0.,'FREQUENCY (HZ)',14,10.0,0.0,FREQ(402),
>FREQ(403))
    DO 21 J=1,4
    IF(.NOT.(PASS.EQ.2))ELEM(2*J-1)=EL(2*J-1)
    IF(.NOT.(PASS.EQ.2))ELEM(2*J)=EL(2*J)*10.0**6.0
    IF(PASS.EQ.2)ELEM(2*J-1)=EI(2*J-1)*10.0**6.0
    IF(PASS.EQ.2)ELEM(2*J)=EL(2*J)
    IF(PASS.EQ.3)ELEM(2*J+7)=EI(2*J+7)*10.0**6.0
    IF(PASS.EQ.3)ELEM(2*J+8)=EI(2*J+8)
21 CONTINUE
    WRITE(108,22)RA
22 FORMAT(F10.1)
    DO 23 J=1,8
    WRITE(108,24)ELEM(J),ELEM(.1+8)
```

```
24 FORMAT(2F12.6)
23 CONTINUE
  WRITE(108,25)RR
25 FORMAT(F10.1)
  WRITE(108,26)WK,FC,F1,F2,F0
26 FORMAT(5F10.2)
DRAW CIRCUIT DIAGRAM
  CALL PLBT(14.0,4.0,-3)
  X=0.0; Y=0.0;          CALL PLBT(X,Y,3)
  CALL CIRCL(X,Y,0.0,360.0,0.05,0.05,0.0)
  IF(R.GT.0)GO TO 27
  X=X+1.0;          CALL PLBT(X,Y,2)
  IF(PASS.EQ.3)A=-1.0
  IF(.NOT.(PASS.EQ.3))A=-0.6
  CALL PLBT(A,0.0,-3)
  GO TO 28
27 CONTINUE
  X=X+0.6;          CALL PLBT(X,Y,2)
  CALL SRESH(X,Y)
  X=X+0.2;          CALL PLBT(X,Y,2)
  RAK=RA/1000.0
  V=AINT(ALOG10(RA));U=5-V
  CALL NUMBER(.75,0.07,0.10,RAK,0.0,U)
  CALL SYMBOL(1.2,0.07,0.10,K,0.0,1)
  IF(PASS.EQ.3)R=-0.4
  IF(.NOT.(PASS.EQ.3))B=0.0
  CALL PLBT(B,0.0,-3)
28 CONTINUE
  GO TO (1000,2000,3000)PASS
1000 DO 1001 J=1,P
  CALL PLBT(1.6,0.0,-3)
  X=0.0; Y=0.0;          CALL PLBT(X,Y,3)
  M=AINT(ALOG10(10000.*ELEM(2*J-1)))
  T=AINT(ALOG10(10000000.*ELFM(2*J)))
  CALL SLC
1001 CONTINUE
  IF(R.EQ.0.0)LEN=2.2+1.6*P
  IF(R.GT.0.0)LEN=2.8+1.6*P
  IF(E.EQ.N)GO TO 1200
  CALL PLBT(1.6,0.0,-3)
  X=0.0; Y=0.0;          CALL PLBT(X,Y,3)
  M=AINT(ALOG10(10000.*ELEM(2*P+1)))
  CALL SL
  IF(R.EQ.0.0)LEN=2.2+1.6*(P+1)
  IF(R.GT.0.0)LEN=2.8+1.6*(P+1)
1200 CONTINUE
  GO TO 4000
2000 DO 2001 J=1,P
  CALL PLBT(1.6,0.0,-3)
  X=0.0; Y=0.0;          CALL PLBT(X,Y,3)
  L=AINT(ALOG10(10000000.*ELFM(2*J-1)))
  Q=AINT(ALOG10(10000.*ELEM(2*J)))
  CALL SCL
2001 CONTINUE
  IF(R.EQ.0.0)LEN=2.2+1.6*P
```

```
IF(R.GT.0.0)LEN=2.8+1.6*P
IF(E.EQ.N)GA TB 2200
CALL PLBT(1.6,0.0,-3)
X=0.0; Y=0.0; CALL PLBT(X,Y,3)
M=AINT(ALBG10(10000000.*ELFM(2*P+1)))
CALL SC
IF(R.EQ.0.0)LEN=2.2+1.6*(P+1)
IF(R.GT.0.0)LEN=2.8+1.6*(P+1)
2200 CONTINUE
GA TB 4000
3000 DA 3001 J=1,P
CALL PLBT(2.0,0.0,-3)
X=0.0; Y=0.0; CALL PLBT(X,Y,3)
M=AINT(ALBG10(10000.*ELEM(2*J-1)))
L=AINT(ALBG10(10000000.*ELFM(2*J+7)))
Q=AINT(ALBG10(10000.*ELEM(2*J+8)))
T=AINT(ALBG10(10000000.*ELFM(2*J)))
CALL SLCB1
3001 CONTINUE
IF(R.EQ.0.0)LEN=2.6+2.0*P
IF(R.GT.0.0)LEN=3.2+2.0*P
IF(E.EQ.N)GA TB 3200
CALL PLBT(2.0,0.0,-3)
X=0.0; Y=0.0; CALL PLBT(X,Y,3)
M=AINT(ALBG10(10000.*ELEM(2*P+1)))
L=AINT(ALBG10(10000000.*ELFM(2*P+9)))
CALL SLCB2
IF(R.EQ.0.0)LEN=2.2+2.0*(P+1)
IF(R.GT.0.0)LEN=2.8+2.0*(P+1)
3200 CONTINUE
4000 CONTINUE
IF(PASS.EQ.3.AND.E.EQ.N)GA TB 30
X=X+0.6; CALL PLBT(X,Y,3)
Y=Y+0.65; CALL PLBT(X,Y,2)
CALL SRESV(X,Y)
Y=Y+0.65; CALL PLBT(X,Y,2)
X=X+0.6; Y=Y+2.1; CALL PLBT(X,Y,3)
X=X+1.2; CALL PLBT(X,Y,2)
GA TB 35
30 CONTINUE
X=X+1.0; CALL PLBT(X,Y,3)
Y=Y+0.65; CALL PLBT(X,Y,2)
CALL SRESV(X,Y)
Y=Y+0.65; CALL PLBT(X,Y,2)
X=X+1.0; Y=Y+2.1; CALL PLBT(X,Y,3)
X=X+1.6; CALL PLBT(X,Y,2)
35 CONTINUE
CALL CIRCL (X,Y,180.0,-180.0,0.05,0.05,0.0)
Y=Y+2.1; CALL PLBT(X,Y,3)
CALL CIRCL (X,Y,180.0,-180.0,0.05,0.05,0.0)
X=X+LEN; CALL PLBT(X,Y,2)
CALL CIRCL (X,Y,0.0,360.0,0.05,0.05,0.0)
RBK=RB/1000.0
V=AINT(ALBG10(RB)); U=5-V
RAX1=X+LEN=0.47
```

```
RAX2=RAX1+0.45
CALL NUMBER (RAX1,-1.1,0.1,RBK,0.0,U)
CALL SYMBOL (RAX2,-1.1,0.1,'K',0.0,1)
GO TO (31,32,33)PASS
31 D=1.0+AINT(ALOG10(FC))
CALL PLOT(X,-2.8,-3)
X=0.0; Y=0.0; CALL PLOT(X,Y,3)
X=(LEN/2.0)-1.05
CALL SYMBOL (X,0.0,.14,'LOW-PASS FILTER',0.0,15)
X=(LEN/2.0)-1.05;Y=-0.05; CALL PLOT(X,Y,3)
X=X+2.1; CALL PLOT(X,Y,2)
CALL SYMBOL(0.2,0.5,0.1,'FC=',0.0,3)
CALL NUMBER (0.5,0.5,0.1,FC,0.0,-1)
CALL SYMBOL (.55+0.1*D,0.5,0.1,'HZ',0.0,2)
GO TO 34
32 D=1.0+AINT(ALOG10(FC))
CALL PLOT(X,-2.8,-3)
X=0.0; Y=0.0; CALL PLOT(X,Y,3)
X=(LEN/2.0)-1.12
CALL SYMBOL (X,0.0,.14,'HIGH-PASS FILTER',0.0,16)
X=(LEN/2.0)-1.12;Y=-0.05; CALL PLOT(X,Y,3)
X=X+2.24; CALL PLOT(X,Y,2)
CALL SYMBOL(0.2,0.5,0.1,'FC=',0.0,3)
CALL NUMBER (0.5,0.5,0.1,FC,0.0,-1)
CALL SYMBOL (.55+0.1*D,0.5,0.1,'HZ',0.0,2)
GO TO 34
33 D=1.0+AINT(ALOG10(F1))
CALL PLOT(X,-2.8,-3)
X=0.0; Y=0.0; CALL PLOT(X,Y,3)
D=1.0+AINT(ALOG10(F2))
X=(LEN/2.0)-1.05
CALL SYMBOL (X,0.0,.14,'BANDPASS FILTER',0.0,15)
X=(LEN/2.0)-1.05;Y=-0.05; CALL PLOT(X,Y,3)
X=X+2.1; CALL PLOT(X,Y,2)
CALL SYMBOL (0.2,0.5,0.1,'F1=',0.0,3)
CALL NUMBER (0.5,0.5,0.1,F1,0.0,-1)
CALL SYMBOL (.55+0.1*D,0.5,0.1,'HZ',0.0,2)
CALL SYMBOL (0.2,0.35,0.1,'F2=',0.0,3)
CALL NUMBER (0.5,.35,0.1,F2,0.0,-1)
CALL SYMBOL (.55+0.1*D,0.35,0.1,'HZ',0.0,2)
34 CONTINUE
CALL NWPL0T
STOP
END
```

SUBROUTINE SRESH(X,Y)

DB 901 MOP=1,5

X=X+0.04

Y=Y+0.04

CALL PLBT(X,Y,2)

X=X+0.08

Y=Y-0.08

CALL PLBT(X,Y,2)

X=X+0.04

Y=Y+0.04

CALL PLBT(X,Y,2)

901

CONTINUE

RETURN

END

SUBROUTINE SRESV(X,Y)

DB 1 I=1,5

X=X+0.04; Y=Y-0.04; CALL PLBT(X,Y,2)

X=X-0.08; Y=Y-0.08; CALL PLBT(X,Y,2)

X=X+0.04; Y=Y-0.04; CALL PLBT(X,Y,2)

1 CONTINUE

RETURN

END

SUBROUTINE SLH(X,Y)

CALL CIRCL (X,Y,180.0,-45.0,0.1,0.1,0.0)

X=X+0.1707; Y=Y-0.0707; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,225.0,-45.0,0.1,0.1,0.0)

X=X+0.1414; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,225.0,-45.0,0.1,0.1,0.0)

X=X+0.1414; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,225.0,-45.0,0.1,0.1,0.0)

X=X+0.1414; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,225.0,0.0,0.1,0.1,0.0)

X=X+0.1707; Y=Y+0.0707; CALL PLBT(X,Y,3)

RETURN

END

SUBROUTINE SLV(X,Y)

CALL CIRCL (X,Y,90.0,-135.0,0.1,0.1,0.0)

X=X-0.0707; Y=Y-0.1707; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,135.0,-135.0,0.1,0.1,0.0)

Y=Y-0.1414; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,135.0,-135.0,0.1,0.1,0.0)

Y=Y-0.1414; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,135.0,-135.0,0.1,0.1,0.0)

Y=Y-0.1414; CALL PLBT(X,Y,3)

CALL CIRCL (X,Y,135.0,-90.0,0.1,0.1,0.0)

X=X+0.0707; Y=Y-0.1707; CALL PLBT(X,Y,3)

RETURN

END

```
SUBROUTINE SL
COMMON X,Y,M,L,Q,T,J,ELEM(16)
X=X+0.4172;          CALL PLOT(X,Y,2)
CALL SLH(X,Y)
X=X+0.4172;          CALL PLOT(X,Y,2)
GB TB (1,2,3,4,5)M
1 CALL NUMBER(0.55,0.14,0.10,1000.*ELEM(2*I-1),0.0,1)
  CALL SYMBOL(0.90,0.14,0.10,'MH',0.0,2); GB TB 6
2 CALL NUMBER(0.50,0.14,0.10,1000.*ELEM(2*I-1),0.0,1)
  CALL SYMBOL(0.95,0.14,0.10,'MH',0.0,2); GB TB 6
3 CALL NUMBER(0.55,0.14,0.10,1000.*ELEM(2*I-1),0.0,-1)
  CALL SYMBOL(0.90,0.14,0.10,'MH',0.0,2); GB TB 6
4 CALL NUMBER(0.55,0.14,0.10,ELEM(2*I-1),0.0,2)
  CALL SYMBOL(1.00,0.14,0.10,'H',0.0,1); GB TB 6
5 CALL NUMBER(0.55,0.14,0.10,ELEM(2*I-1),0.0,1)
  CALL SYMBOL(1.00,0.14,0.10,'H',0.0,1)
6 CONTINUE
X=1.6; Y=0.0;      CALL PLOT(X,Y,3)
RETURN
END
```

```
SUBROUTINE SLC
COMMON X,Y,M,L,Q,T,J,ELEM(16)
INTEGER T
CALL SL
Y=Y-1.0;           CALL PLOT(X,Y,2)
X=X-0.15;          CALL PLOT(X,Y,3)
X=X+0.3;           CALL PLOT(X,Y,2)
Y=Y-0.1;           CALL PLOT(X,Y,3)
X=X-0.3;           CALL PLOT(X,Y,2)
X=X+0.15;          CALL PLOT(X,Y,3)
Y=Y-1.0;           CALL PLOT(X,Y,2)
Y=Y+2.1;           CALL PLOT(X,Y,3)
IF(T.LE.4)ELEM(2*I)=ELEM(2*I)*10.0**6.0
GB TB (1,2,3,4,5,6,7,8)T
1 CALL NUMBER(1.25,-1.025,0.1,ELEM(2*I),0.,-1)
  CALL SYMBOL(1.20,-1.175,0.1,'PF',0.,2); GB TB 9
2 CALL NUMBER(1.20,-1.025,0.1,ELEM(2*I),0.,-1)
  CALL SYMBOL(1.20,-1.175,0.1,'PF',0.,2); GB TB 9
3 CALL NUMBER(1.10,-1.025,0.1,ELEM(2*I),0.,-1)
  CALL SYMBOL(1.15,-1.175,0.1,'PF',0.,2); GB TB 9
4 CALL NUMBER(1.00,-1.025,0.1,ELEM(2*I),0.,-1)
  CALL SYMBOL(1.10,-1.175,0.1,'PF',0.,2); GB TB 9
5 CALL NUMBER(0.80,-1.025,0.1,ELEM(2*I),0.,4)
  CALL SYMBOL(0.95,-1.175,0.1,'MFD',0.,3); GB TB 9
6 CALL NUMBER(0.90,-1.025,0.1,ELEM(2*I),0.,3)
  CALL SYMBOL(1.00,-1.175,0.1,'MFD',0.,3); GB TB 9
7 CALL NUMBER(1.00,-1.025,0.1,ELEM(2*I),0.,2)
  CALL SYMBOL(1.05,-1.175,0.1,'MFD',0.,3); GB TB 9
8 CALL NUMBER(1.00,-1.025,0.1,ELEM(2*I),0.,1)
  CALL SYMBOL(1.05,-1.175,0.1,'MFD',0.,3)
9 CONTINUE
RETURN
END
```

```
SUBROUTINE SC
COMMON X,Y,M,L,Q,T,J,ELEM(16)
X=X+0.75;          CALL PL0T(X,Y,2)
Y=Y+0.15;         CALL PL0T(X,Y,3)
Y=Y+0.3;          CALL PL0T(X,Y,2)
X=X+0.1;          CALL PL0T(X,Y,3)
Y=Y+0.3;          CALL PL0T(X,Y,2)
Y=Y+0.15;         CALL PL0T(X,Y,3)
X=X+0.75;         CALL PL0T(X,Y,2)
IF(L.LE.4)ELEM(2*J+7)=ELEM(2*J+7)*10.0**6.0
GB TB (1,2,3,4,5,6,7,8)L
1 CALL NUMBER(0.75,0.37,0.1,FLEM(2*J-1),0.,-1)
  CALL SYMB0L(0.70,0.20,0.1,'PF',0.,2); GB TB 9
2 CALL NUMBER(0.70,0.37,0.1,ELEM(2*J-1),0.,-1)
  CALL SYMB0L(0.70,0.20,0.1,'PF',0.,2); GB TB 9
3 CALL NUMBER(0.65,0.37,0.1,FLEM(2*J-1),0.,-1)
  CALL SYMB0L(0.70,0.20,0.1,'PF',0.,2); GB TB 9
4 CALL NUMBER(0.60,0.37,0.1,FLEM(2*J-1),0.,-1)
  CALL SYMB0L(0.70,0.20,0.1,'PF',0.,2); GB TB 9
5 CALL NUMBER(0.50,0.37,0.1,FLEM(2*J-1),0.,4)
  CALL SYMB0L(0.65,0.20,0.1,'MFD',0.,3); GB TB 9
6 CALL NUMBER(0.55,0.37,0.1,FLEM(2*J-1),0.,3)
  CALL SYMB0L(0.65,0.20,0.1,'MFD',0.,3); GB TB 9
7 CALL NUMBER(0.60,0.37,0.1,FLEM(2*J-1),0.,2)
  CALL SYMB0L(0.65,0.20,0.1,'MFD',0.,3); GB TB 9
8 CALL NUMBER(0.60,0.37,0.1,FLEM(2*J-1),0.,1)
  CALL SYMB0L(0.65,0.20,0.1,'MFD',0.,3); GB TB 9
9 CONTINUE
X=1.6; Y=0.0;     CALL PL0T(X,Y,3)
RETURN
END

SUBROUTINE SCL
COMMON X,Y,M,L,Q,T,J,ELEM(16)
INTEGER Q
CALL SC
Y=Y+0.6672;       CALL PL0T(X,Y,2)
CALL SLV(X,Y)
Y=Y+0.6672;       CALL PL0T(X,Y,2)
Y=Y+2.1;          CALL PL0T(X,Y,3)
GB TB (1,2,3,4,5)Q
1 CALL NUMBER(1.76,-1.025,0.1,ELEM(2*J)*1000.,0.,1)
  CALL SYMB0L(1.81,-1.175,0.1,'MH',0.,2); GB TB 6
2 CALL NUMBER(1.76,-1.025,0.1,ELEM(2*J)*1000.,0.,1)
  CALL SYMB0L(1.86,-1.175,0.1,'MH',0.,2); GB TB 6
3 CALL NUMBER(1.76,-1.025,0.1,ELEM(2*J)*1000.,0.,-1)
  CALL SYMB0L(1.81,-1.175,0.1,'MH',0.,2); GB TB 6
4 CALL NUMBER(1.76,-1.025,0.1,ELEM(2*J),0.,2)
  CALL SYMB0L(1.91,-1.175,0.1,'H',0.,1); GB TB 6
5 CALL NUMBER(1.76,-1.025,0.1,ELEM(2*J),0.,1)
  CALL SYMB0L(1.91,-1.175,0.1,'H',0.,1)
6 CONTINUE
RETURN
END
```

```
SUBROUTINE SLCR1
COMMON X,Y,M,L,Q,T,J,ELEM(16)
INTEGER Q,T
CALL SLCB2
Y=Y-0.5;          CALL PLOT(X,Y,2)
X=X+0.2;          CALL PLOT(X,Y,2)
Y=Y-0.1672;      CALL PLOT(X,Y,2)
CALL SLV(X,Y)
Y=Y-0.1672;      CALL PLOT(X,Y,2)
X=X-0.4;          CALL PLOT(X,Y,2)
Y=Y+0.5;          CALL PLOT(X,Y,2)
X=X-0.15;         CALL PLOT(X,Y,3)
X=X+0.3;          CALL PLOT(X,Y,2)
Y=Y+0.1;          CALL PLOT(X,Y,3)
X=X-0.3;          CALL PLOT(X,Y,2)
X=X+0.15;         CALL PLOT(X,Y,3)
Y=Y+0.5;          CALL PLOT(X,Y,2)
X=X+0.2;          CALL PLOT(X,Y,2)
Y=Y-1.1;          CALL PLOT(X,Y,3)
Y=Y-0.5;          CALL PLOT(X,Y,2)
Y=Y+2.1;          CALL PLOT(X,Y,3)
GA TB (1,2,3,4,5)0
1 CALL NUMBER(2.36,-1.025,0.1,ELEM(2*J+8)*1000,0,1)
  CALL SYMBOL(2.41,-1.175,0.1,'MH',0,2),GA TB 6
2 CALL NUMBER(2.36,-1.025,0.1,ELEM(2*J+8)*1000,0,1)
  CALL SYMBOL(2.46,-1.175,0.1,'MH',0,2),GA TB 6
3 CALL NUMBER(2.36,-1.025,0.1,ELEM(2*J+8)*1000,0,-1)
  CALL SYMBOL(2.41,-1.175,0.1,'MH',0,2),GA TB 6
4 CALL NUMBER(2.36,-1.025,0.1,ELEM(2*J+8),0,2)
  CALL SYMBOL(2.51,-1.175,0.1,'H',0,1),GA TB 6
5 CALL NUMBER(2.36,-1.025,0.1,ELEM(2*J+8),0,1)
  CALL SYMBOL(2.51,-1.175,0.1,'H',0,1)
6 IF(T.LE.4)ELEM(2*J)=ELEM(2*J)*10.0**6.0
  GA TB (7,8,9,10,11,12,13,14)T
7 CALL NUMBER(1.45,-1.025,0.1,ELEM(2*J),0,-1)
  CALL SYMBOL(1.40,-1.175,0.1,'PF',0,2),GA TB 15
8 CALL NUMBER(1.40,-1.025,0.1,ELEM(2*J),0,-1)
  CALL SYMBOL(1.40,-1.175,0.1,'PF',0,2),GA TB 15
9 CALL NUMBER(1.30,-1.025,0.1,ELEM(2*J),0,-1)
  CALL SYMBOL(1.35,-1.175,0.1,'PF',0,2),GA TB 15
10 CALL NUMBER(1.20,-1.025,0.1,ELEM(2*J),0,-1)
  CALL SYMBOL(1.30,-1.175,0.1,'PF',0,2),GA TB 15
11 CALL NUMBER(1.00,-1.025,0.1,ELEM(2*J),0,4)
  CALL SYMBOL(1.15,-1.175,0.1,'MFD',0,3),GA TB 15
12 CALL NUMBER(1.10,-1.025,0.1,ELEM(2*J),0,3)
  CALL SYMBOL(1.20,-1.175,0.1,'MFD',0,3),GA TB 15
13 CALL NUMBER(1.20,-1.025,0.1,ELEM(2*J),0,2)
  CALL SYMBOL(1.25,-1.175,0.1,'MFD',0,3),GA TB 15
14 CALL NUMBER(1.20,-1.025,0.1,ELEM(2*J),0,1)
  CALL SYMBOL(1.25,-1.175,0.1,'MFD',0,3)
15 CONTINUE
  X=2.0; Y=0.0;   CALL PLOT(X,Y,3)
  RETURN
END
```

```
SUBROUTINE SLCR2
COMMON X,Y,M,L,Q,T,J,ELEM(16)
INTEGER Q,T
X=X+0.3472;          CALL PLOT(X,Y,2)
CALL SLH(X,Y)
X=X+0.4372;          CALL PLOT(X,Y,2)
Y=Y+0.15;           CALL PLOT(X,Y,3)
Y=Y+0.3;            CALL PLOT(X,Y,2)
X=X+0.1;            CALL PLOT(X,Y,3)
Y=Y+0.3;            CALL PLOT(X,Y,2)
Y=Y+0.15;           CALL PLOT(X,Y,3)
X=X+0.35;           CALL PLOT(X,Y,2)
GB TB (1,2,3,4,5)M
1 CALL NUMBER(0.48,0.14,0.1,FLEM(2*J-1)*1000.,0.,1)
  CALL SYMB0L(0.83,0.14,0.1,'MH',0.,2); GB TB 6
2 CALL NUMBER(0.43,0.14,0.1,FLEM(2*J-1)*1000.,0.,1)
  CALL SYMB0L(0.88,0.14,0.1,'MH',0.,2); GB TB 6
3 CALL NUMBER(0.48,0.14,0.1,ELEM(2*J-1)*1000.,0.,1)
  CALL SYMB0L(0.83,0.14,0.1,'MH',0.,2); GB TB 6
4 CALL NUMBER(0.48,0.14,0.1,ELEM(2*J-1),0.,2)
  CALL SYMB0L(0.93,0.14,0.1,'H',0.,1); GB TB 6
5 CALL NUMBER(0.48,0.14,0.1,FLEM(2*J-1),0.,1)
  CALL SYMB0L(0.93,0.14,0.1,'H',0.,1)
6 IF(L.LE.4)ELEM(2*J+7)=ELEM(2*J+7)*10.0**6.0
  GB TB (7,8,9,10,11,12,13,14)L
7 CALL NUMBER(1.55,0.37,0.1,FLEM(2*J+7),0.,-1)
  CALL SYMB0L(1.50,0.18,0.1,'PF',0.,2); GB TB 15
8 CALL NUMBER(1.50,0.37,0.1,FLEM(2*J+7),0.,-1)
  CALL SYMB0L(1.50,0.18,0.1,'PF',0.,2); GB TB 15
9 CALL NUMBER(1.45,0.37,0.1,FLEM(2*J+7),0.,-1)
  CALL SYMB0L(1.50,0.18,0.1,'PF',0.,2); GB TB 15
10 CALL NUMBER(1.47,0.37,0.1,FLEM(2*J+7),0.,-1)
  CALL SYMB0L(1.50,0.18,0.1,'PF',0.,2); GB TB 15
11 CALL NUMBER(1.30,0.37,0.1,FLEM(2*J+7),0.,4)
  CALL SYMB0L(1.45,0.18,0.1,'MFD',0.,3); GB TB 15
12 CALL NUMBER(1.35,0.37,0.1,FLEM(2*J+7),0.,3)
  CALL SYMB0L(1.45,0.18,0.1,'MFD',0.,3); GB TB 15
13 CALL NUMBER(1.40,0.37,0.1,FLEM(2*J+7),0.,2)
  CALL SYMB0L(1.45,0.18,0.1,'MFD',0.,3); GB TB 15
14 CALL NUMBER(1.40,0.37,0.1,FLEM(2*J+7),0.,1)
  CALL SYMB0L(1.45,0.18,0.1,'MFD',0.,3); GB TB 15
15 CONTINUE
X=2.0; Y=0.0;      CALL PLOT(X,Y,3)
RETURN
END
```

