G. PLASS [redacted]

HHU/ab

MPS/Int. SR/68-10
14. 10. 1968

COMPUTER ANALYSIS OF REFLECTED STEP PULSES AND OTHER TRANSIENTS

by

H. H. Umstätter

1. Summary

One method of testing feedback control systems and electrical networks such as amplifiers, filters, cables or delay line magnets is to send a step pulse or some other test pulse into the network and observe how it is transmitted or reflected. One obtains an output signal y(t) of a characteristic shape. The computer programme analyses this shape, compares it with the input pulse and produces plots of the frequency response, i.e. amplitude and phase of the network.

2. Principle and use of the programme

The programme in compiled version consists of 34 binary cards plus a small FORTRAN subroutine which contains the formula of the Laplace transform of the test input pulse. It can be run as an express job because it requires 16000 memory locations and not more than 10 sec central processor time on the CDC 6600 computer.

PS/6713

For example, we may have used a time domain reflectometer with an attached x-y-recorder which plots the reflected or transmitted pulse on paper, or we may have photographed a transient on an oscilloscope as shown in fig. 1. We take readings of the signal y(t) and the time t in $N \geq 4$ consecutive points and punch this number N into the first data card and the values t and y(t) into the following cards as shown in table 1. It is not necessary to take readings at equal time intervals but to take only the more significant points (e.g. maxima and minima) because the programme connects these points by a fairly smooth curve (cubic Lagrange interpolation) and calculates 600 equidistant sampling points situated in the middle of every time interval.

In the last data card we punch the maximum frequency up to which we want to plot the frequency response (see table 1, frequency unit = 1/time unit, e.g. sec and Hz or nanosec and GHz).

Now the programme carries out the Laplace transform from the time domain into the frequency domain replacing the integral by a sum: The signal y(t) is multiplied by $e^{-j\omega_0 t}$ in 600 sampling points. These products are multiplied by the time interval $t_o = (t_n - t_1)/600$ and summed up. This yields the complex amplitude at a frequency ω_0. The process is repeated for 500 frequencies ω_0, $2\omega_0$, $3\omega_0$, ... 500 ω_0 in order to obtain detailed plots of frequency response. Then the 500 complex amplitudes of the output pulse are divided by the amplitudes of the input pulse in order to evaluate the network function (e.g."gain" or transfer function of an amplifier). The amplitudes of the input pulse are evaluated in an exchangeable subprogramme COMPLEX FUNCTION SPECTR(F) containing the Laplace transform of the input pulse (e.g. step pulse). Finally, the magnitude and phase of the network function are evaluated, the 500 points are connected by linear interpolation and plotted on the CALCOMP plotter. We obtain two plots 10 x 14 inches of the amplitude and phase lag (modulo 360°) versus frequency as shown in fig. 2a, 2b.

The following difficulties have arisen by the fact that the signal is recorded only from $t = t_1$ to t_n, whereas the integral should extend from $-\infty$ to $+\infty$. If we want the integral to vanish for $t \leq t_1$ we have to shift the base line vertically from $y(t)$ to $y(t)-y(t_1)$, otherwise the programme would return the frequency response of a signal which jumps at $t = t_1$ from $y = 0$ to $y(t_1)$. But even if the base line is clamped to zero at $t = t_1$ a second difficulty arises at the end of the record in fig. 1, because the signal at t_n is not zero and one cannot extend the numerical integration with 600 sampling points to infinity. Therefore, the numerical integration is carried out only up to $t = t_n$ and an analytical expression is added for the integral from t_n to infinity using the stationary value $y = y(t_n)$. Much computer time has been saved by the use of the complex notation rather than a Fourier series with real terms, because the factor $e^{j\omega_0 t_0}$ is evaluated only once and the other $600 \cdot 500 = 300000$ terms follow from recurrent multiplication.

3. Examples of subroutines for various test pulses

a) Responses to a step pulse

Fig. 1 shows a step pulse which has been sent through a 300 MHz low-pass filter for the wide band pick-up station. The Laplace transform of an ideal step pulse is $1/p$ where $p = j\omega$ is the complex frequency. We write this Laplace transform of the input pulse into the following subprogramme for the analysis of step pulse responses:

```
CØMPLEX FUNCTIØN SPECTR(F)
CØMMØN PIBY2, PI2
ØMEGA = PI2*F
SPECTR=CMPLX(0.,-1./ØMEGA)
RETURN
END
```

Quantities PIBY2=$\pi/2$ and PI2=2π are 15 digit constants which were stored in the COMMON store for convenience and may be referenced by the subprogramme. It can be seen that the real part of SPECTR is zero and the imaginary part $- j/\omega$. If the bandwidth of the oscilloscope does not allow to show all details or if the rise time of the step pulse is not negligibly small, a correction can be made multiplying the spectrum by a bandwidth limiting factor

$$e^{-(f/f_o)^2} = e^{-(1.733\ t_r \cdot f)^2}$$

where f_o is the upper frequency limit where the oscilloscope attenuates by 1 neper = 8.686 db or alternatively t_r is the rise time from 10 % to 90 %. The Gaussian error distribution function

$$e^{-(f/f_o)^2}$$

has been assumed for the spectrum and the error integral for the rise of the test pulse, because a preamplifier with a linear phase response (no phase distortion) should have such an amplitude response. For example, the spectrum of a step pulse with 0.15 nanosec rise time which produced the response in fig. 1, is given by the following subroutine:

```
COMPLEX FUNCTIØN SPECTR(F)
CØMMØN PIBY2,PI2
RISETM=0.15
ØMEGA=PI2*F
SPECTR=CMPLX(0.,-EXP(-(1.733*RISETM*F)**2)/ØMEGA)
RETURN
END
```

The results are shown in fig. 2a and 2b. We assumed that the input pulse occurred at t_o. If we shift the origin of the time scale so that the output pulse is delayed by τ, this only adds a linear phase $\varphi = \omega\tau$ to the phase response. This may be useful when comparing input and output pulses on a long cable where the phase shifts would otherwise be impracticably large. The programme can also be used to analyse reflected pulses.

Fig. 3 shows how the step pulse of a time domain reflectometer is reflected on one of our BNC 75 Ω terminations which have a parasitic capacity. In this case one has to change the polarity of the signal before entering the data into the computer (in order to obtain the correct phase). The reflected signal $\rho(t)$ measured in percent of a completely reflected pulse was punched into the data cards, and the same subprogramme with 0.15 nsec rise time was used. The results are shown in fig. 4a, 4b. The reflection coefficient

$$\rho = \frac{Z - Z_o}{Z + Z_o}$$

starts with a phase of -90^o (parasitic capacity) and its magnitude increases with frequency up to 41 % at 3.8 GHz. It seems to be compensated at 4.6 GHz (resonance). It would be difficult to measure this frequency response directly with a bridge because of the presence of other reflections on the same cable.

b) Spectrum analysis and response to a short unit impulse

The simplest subprogramme is the following:

```
COMPLEX FUNCTION SPECTR(F)
SPECTR=1.
RETURN
END
```

If we use it to analyse the transient in fig. 3, we obtain the plots in fig. 5a and 5b that show the spectrum of the output pulse without any modification (division by 1.). However, SPECTR=1. can also be regarded as the Laplace transform of a short unit impulse (Dirac ρ-function) that could have been sent through some amplifier of limited bandwidth and therefore produces an output pulse which is much longer than the test pulse. In this case we would also use the above subprogramme and fig. 5a, 5b would be the frequency response of the amplifier.

c) Exponentially decaying pulse, square pulse

We can use any test pulse of which the Laplace transform is known, e.g. a pulse obtained by a condenser discharge with time constant RC = 14.8 msec and initial voltage 50 V. Then we have the subroutine

```
COMPLEX FUNCTION SPECTR(F)
COMMON PIBY2,PI2
COMPLEX P
P=CMPLX(0.,PI2*F)
RC=14.8
SPECTR=50.*RC/(1.+P*RC)
RETURN
END
```

$P = j2\pi f$ is the complex frequency. We can multiply the spectrum by a constant in order to increase the amplitude and we can delay a pulse by τ multiplying the Laplace transform by $e^{-p\tau}$. Example of a negative step pulse of 3 V which is delayed by 2.1 msec:

```
P=CMPLX(0.,PI2*F)
TAU=2.1
SPECTR=CEXP(P*TAU)*(-3./P)
```

If we have a sequence of pulses we add the Laplace transforms (linear superposition). A positive step plus a delayed negative step of the same height form a square pulse of length τ:

```
COMPLEX FUNCTION SPECTR(F)
COMMON PIBY2,PI2
COMPLEX P
P=CMPLX(0.,PI2*F)
TAU=2.1
SPECTR=3./P-CEXP(P*TAU)*3./P
RETURN
END
```

In all preceding cases a rise time correction can be made by multiplying the whole SPECTR by the factor $e^{-(f/f_0)^2}$.

4. Final remarks

This programme is called "SNOPSER" because it is the counterpart of another computer programme "RESPONS" which solves the inverse problem, i.e. to calculate the transient response of a network of which the frequency response has been measured or is given by a mathematical expression in terms of the circuit elements. It has been used to calculate for example how various signals are transmitted through long cables plus an amplifier or how a control system responds to a sudden perturbation. Both programmes accept the same subroutines "SPECTR(F)". With this pair of computer programmes we have a free choice between measurements of transient response or frequency response whichever is more practical.

H. H. Umstätter

Distribution: (open)
Scientific Staff MPS and SI

CERN DATA FORM

Data No. ______________ Date ______________

Programmer ______________ Page ____ of ______

Division ______ Tel. No. ______

1st card: I5 — 25 — this is the number of measuring points (t_n , y_n) which will follow beneath

up to 100 consecutive cards:

F10.4 time t	F10.4 signal y(t)	F10.4 time t	F10.4 signal y(t)	F10.4 time t	F10.4 signal y(t)
-0.6	.06	-0.3	.07	-0.2	.08
-0.1	.20	-0.05	.27	0.1	.20
0.25	.37	0.4	.64	0.6	1.0
1.0	2.05	2.0	5.0	2.23	5.4
2.6	5.64	3.2	5.4	3.74	5.0
4.1	4.91	4.4	5.0	5.	5.33
6.	5.70	7.	5.44	7.6	5.35
8.	5.39	9.	5.59	11.3	5.46
13.	5.57				

last card: F10.4 max. frequency — .700

TABLE 1

INPUT DATA

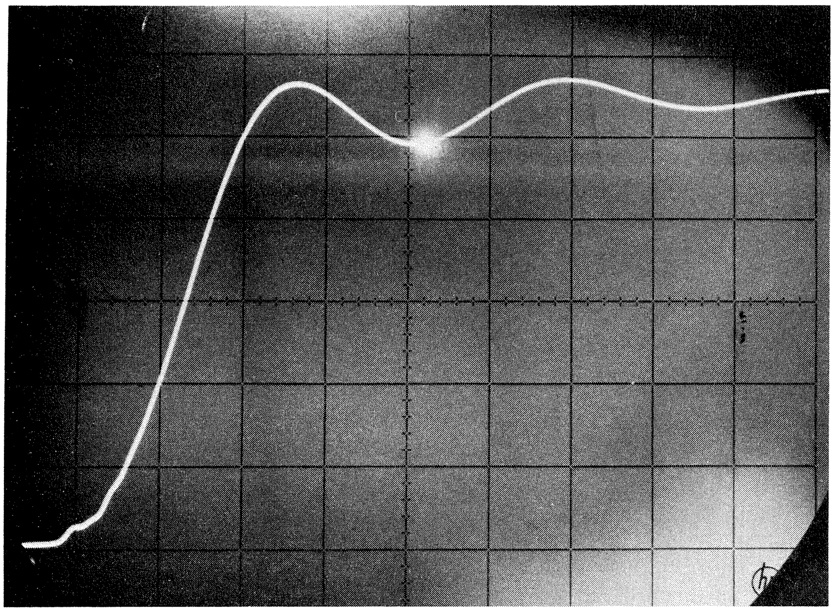


Fig. 1 1 ns/div.

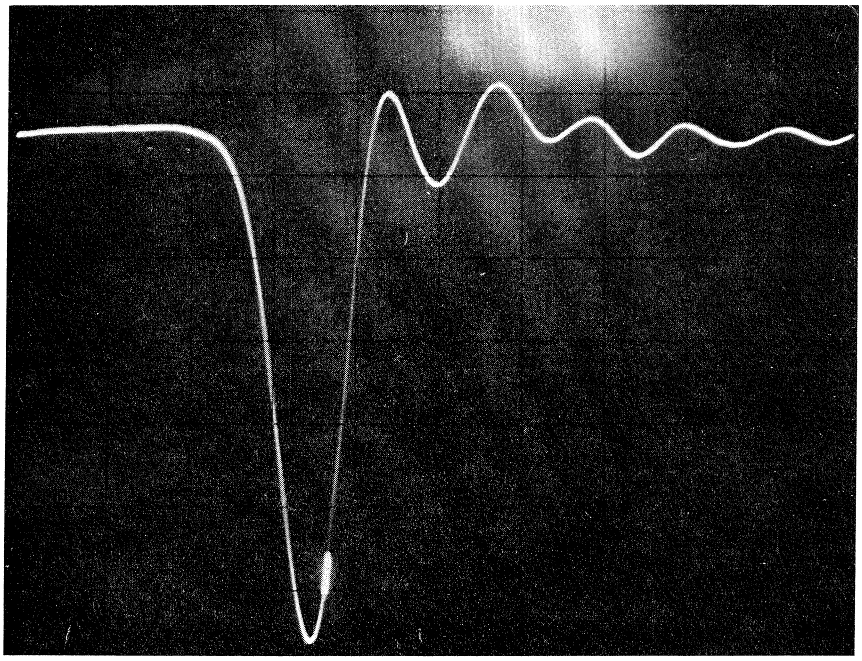


Fig. 3 time base: 0.2 ns/div., vertical: 2% /division.

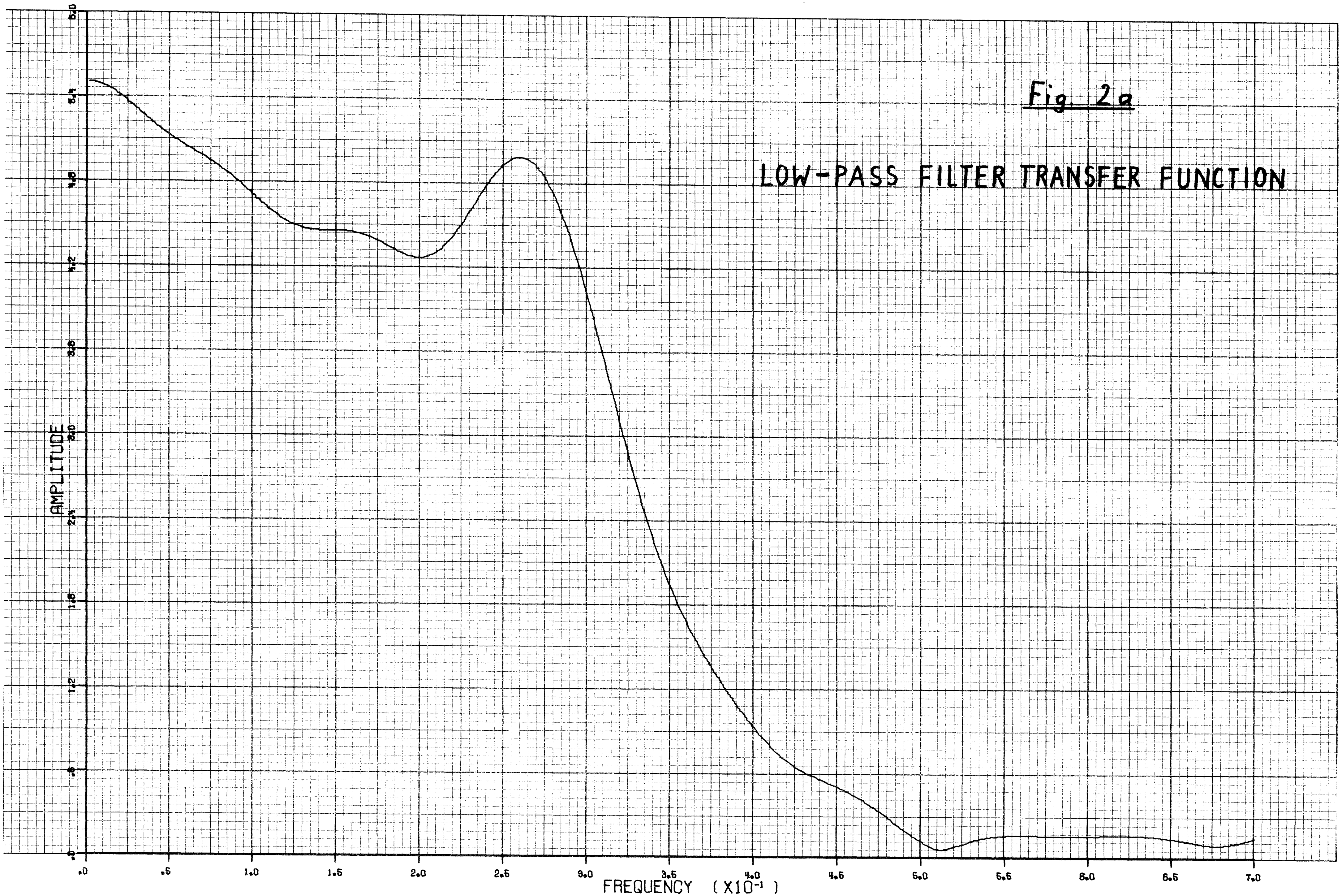
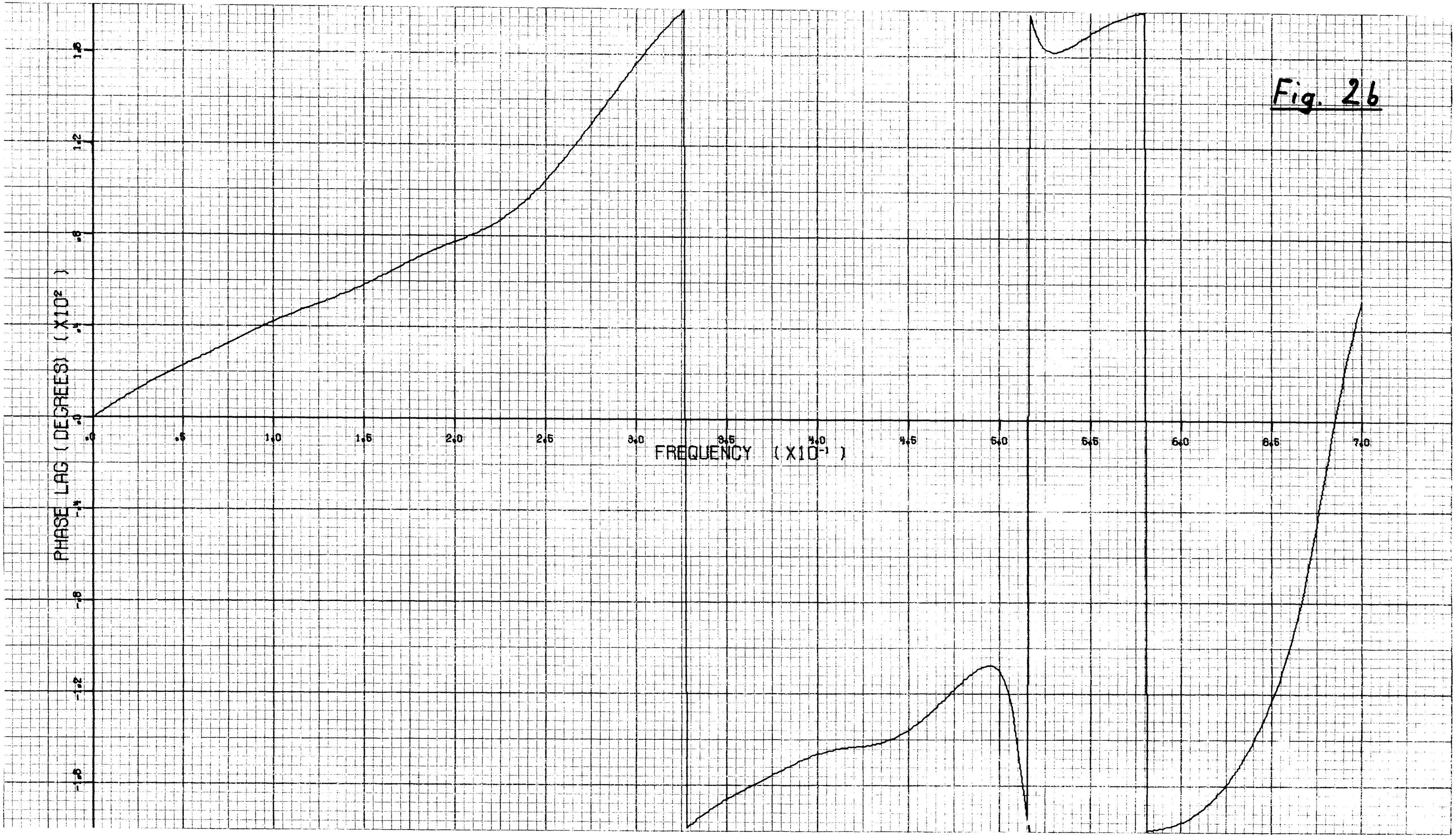
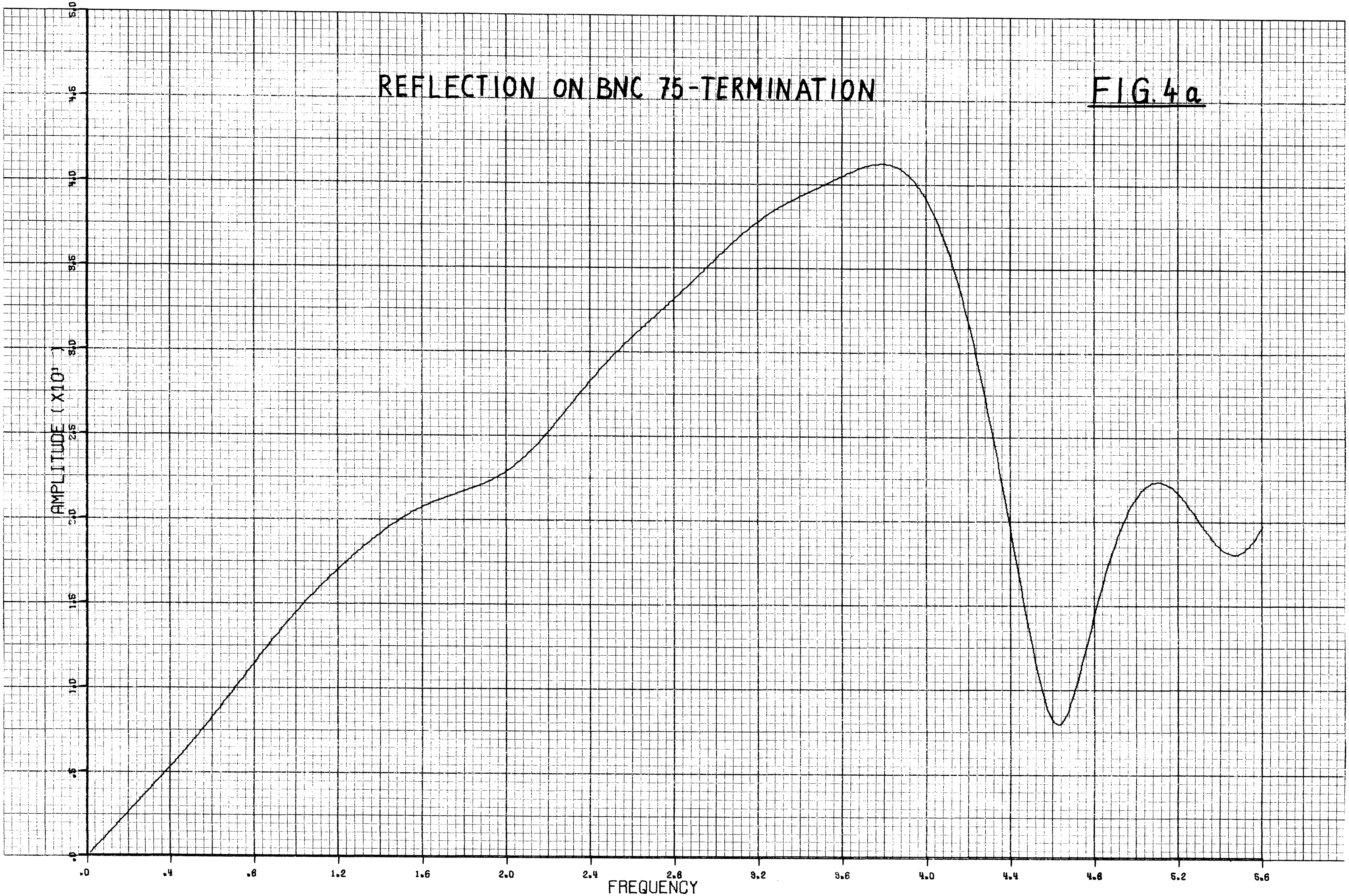


Fig. 2a

LOW-PASS FILTER TRANSFER FUNCTION

Fig. 2b


PHASE LAG (DEGREES) (X10²)
FREQUENCY (X10⁻¹)
.0
.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
1.6
1.2
.8
.4
-.4
-.8
-1.2
-1.6



REFLECTION ON BNC 75-TERMINATION
AMPLITUDE (X10¹)
FREQUENCY
.0
.4
.8
1.2
1.6
2.0
2.4
2.8
3.2
3.6
4.0
4.4
4.8
5.2
5.6
0.
.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0

FIG. 4a

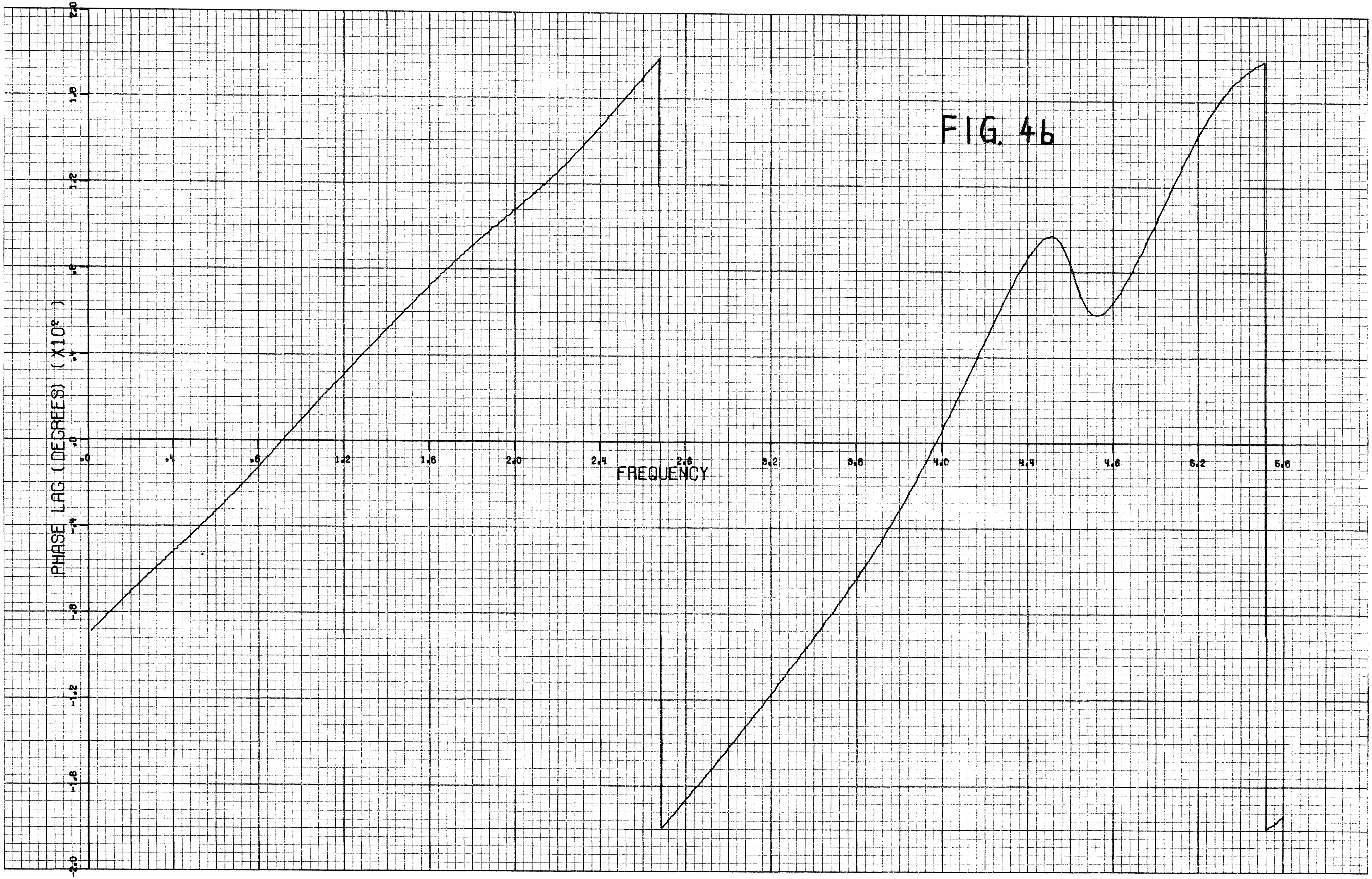

PHASE LAG (DEGREES) (X10²)
FREQUENCY
2.0
1.6
1.2
.8
.4
0
-.4
-.8
-1.2
-1.6
-2.0
.4
.8
1.2
1.6
2.0
2.4
2.8
3.2
3.6
4.0
4.4
4.8
5.2
5.6

FIG. 4b

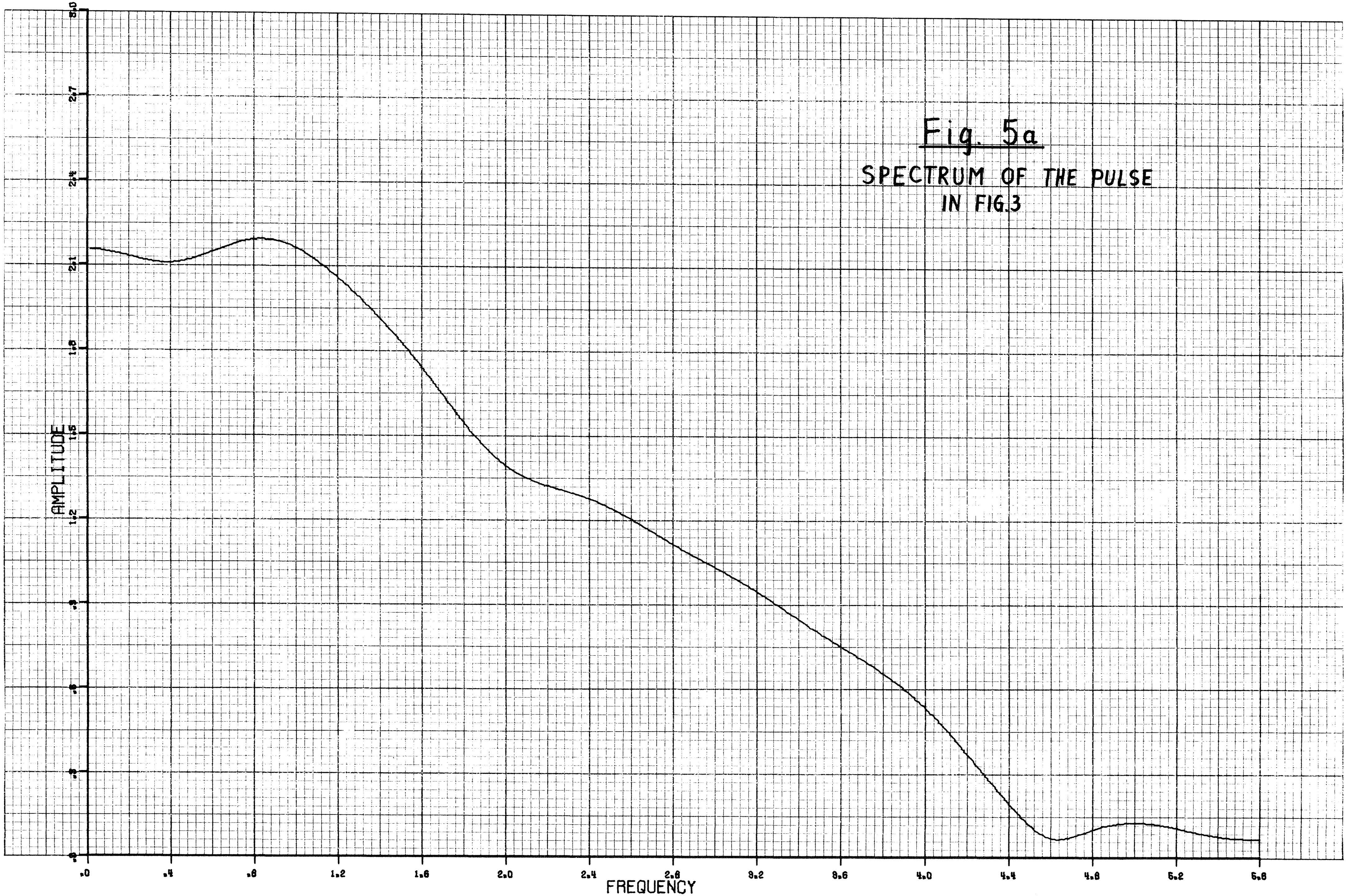


Fig. 5a

SPECTRUM OF THE PULSE IN FIG.3

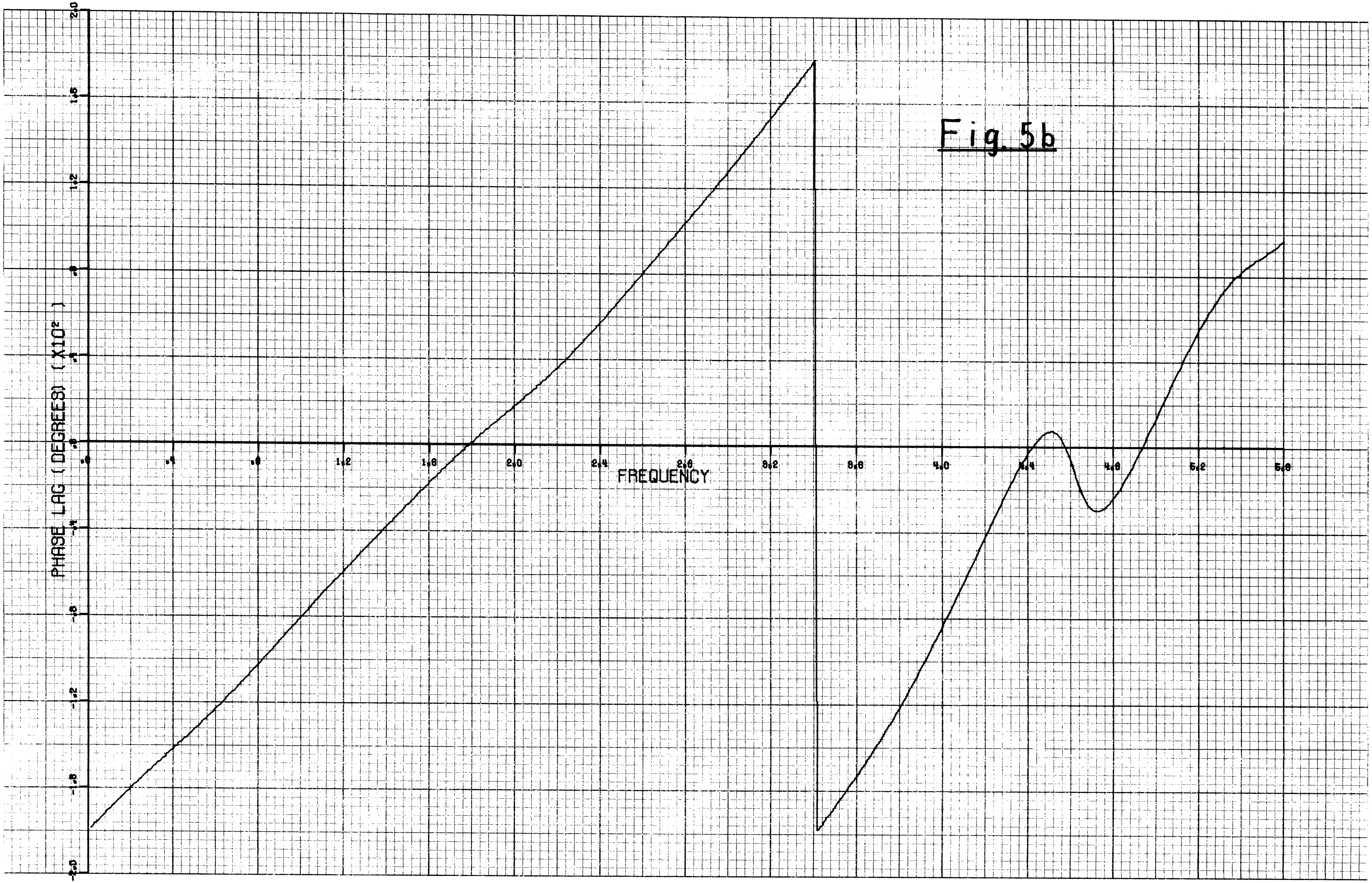


Fig. 5b