CANADIAN THESES

THÈSES CANADIENNES

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30.

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30.

THIS DISSERTATION HAS BEEN MICROFILMED EXACTLY AS RECEIVED

LA THÈSE A ÉTÉ MICROFILMÉE TELLE QUE NOUS L'AVONS REÇUE

THE UNIVERSITY OF ALBERTA

TIME DOMAIN MEASUREMENT OF THE FREQUENCY RESPONSE OF GRADED-INDEX OPTICAL FIBERS

by

DOMINIQUE JODOIN

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

EDMONTON, ALBERTA

SPRING 1986

ISBN 0-315-30219-4

THE UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR DOMINIQUE JODOIN

TITLE OF THESIS TIME DOMAIN MEASUREMENT OF THE FREQUENCY RESPONSE OF GRADED-INDEX OPTICAL FIBERS

DEGREE FOR WHICH THESIS WAS PRESENTED MASTER OF SCIENCE

YEAR THIS DEGREE GRANTED SPRING 1986

(SIGNED) ....

PERMANENT ADDRESS:

561 Boulevard Youville

Chateauguay, Quebec

Canada J6J 4Z2

DATED October 25 1985

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled TIME DOMAIN MEASUREMENT OF THE FREQUENCY RESPONSE OF GRADED-INDEX OPTICAL FIBERS submitted by DOMINIQUE JODOIN in partial fulfilment of the requirements for the degree of MASTER OF SCIENCE.

....

Supervisor

....

....

Date...October 25, 1985............

ABSTRACT

The object of this work was to determine the frequency response, in situ, of existing Edmonton Telephone multimode graded-index optical fiber links. A pulse measurement technique was devised and then used to determine the upper frequency roll-off characteristics of several links.

The work involved the realization of an ultra-short duration, optical impulse generator, a wideband detection circuit, data recording at the input and the output of the fiber and development of a practical fast Fourier transform (FFT) technique for converting time domain measurements to frequency domain characteristics.

ACKNOWLEDGMENTS

I would like to acknowledge valuable assistance provided by the following persons:

1. Edmonton Telephone company employees: Laith Zalzalah, Harvey Meier, Stan Lucky and Ron Sollanych.
2. Erik Valk, M.Sc. student.
3. Dr. Graham Walker, from U.of A.
4. Greg Burley of BNR Edmonton.
5. Darryl Barabash of BNR Edmonton.
6. Randy Giles of BNR Edmonton.
7. Dr. Jan Conradi of BNR Edmonton.

Also, the encouragement and assistance provided by my co-workers in the optical communications research laboratory are gratefully acknowledged, in particular the contributions of Tai Ly, Scott Wakefield, Jann Binder, Rod Anderson, Graham McKinnon and especially Kees den Hartigh, the technician, were greatly appreciated. I would also like to thank especially my supervisor, Dr. George Cormack, for giving me the opportunity to work in a fiber optics area.

Finally, this work was made possible through the financial assistance provided by an Edmonton Telephone company contract and financial support from both BNR and NSERC.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

The maximum distance between repeaters that can be bridged by optical communication fibers carrying digital signals is principally determined by the loss and dispersion characteristics of the fiber. It is thus clear that dispersion and loss are two of the most important fiber parameters that must be carefully measured and controlled. The object of this thesis will be restricted to the measurement of the frequency-domain equivalent to dispersion and loss, specifically the optical fiber bandwidth, or as a more general result, the optical fiber frequency response.

There are two methods available to measure the frequency response of an optical fiber: one can take measurements in the frequency domain or in the time domain. The first method requires use and operation of a network analyser , which is extremely expensive ($50,000.00 to $500,000.00). Because of this fact we decided to concentrate our efforts on the time domain method.

To perform a time domain frequency response measurement, the basic system shown in Fig.1 can be used. A short optical pulse, which contains spectral (baseband) components at least up to the maximum frequency of interest, is injected into the fiber. Such a pulse may be generated, for example, by a solid state diode or a mode-locked laser. For this project we opted for a 1.3 micron (μm) solid-state

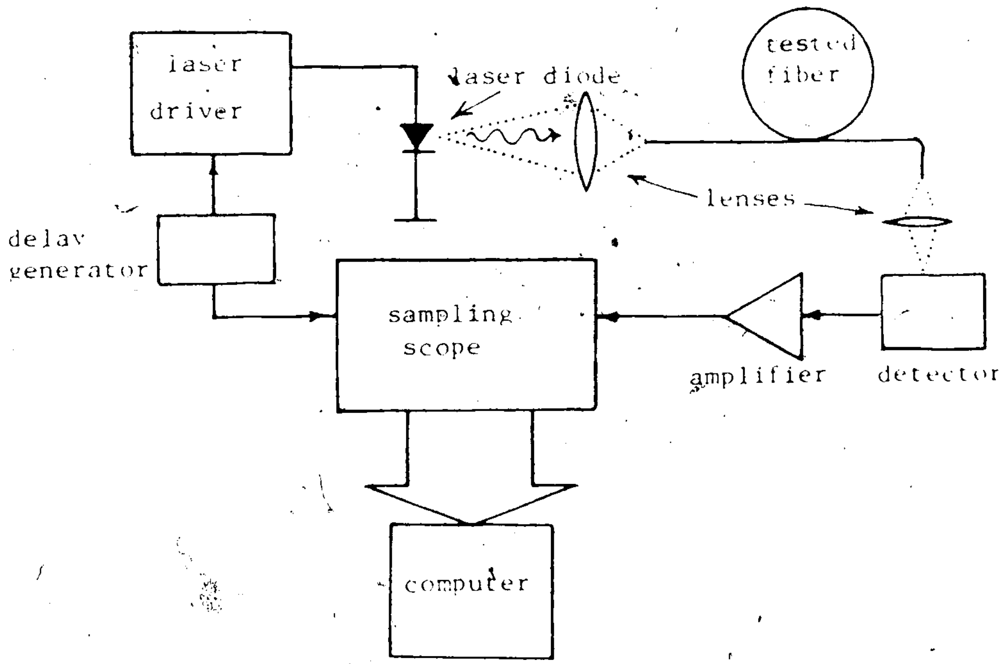


Fig.1 Basic time domain frequency measurement system.

InGaAsP laser which was similar in emission characteristics to the ones that would likely be used in the future on these optical lines, should higher bit rate transmission be desired.

In the measurement system illustrated in Fig.1, the output signal from the fiber is detected with a fast avalanche photodiode and displayed on a sampling oscilloscope. The displayed waveform is then averaged to remove the random noise, converted into digital form, and fed into a computer for signal processing.

Due to dispersion in the fiber, the output pulse is broadened and distorted. The pulse deformation is partly due to the measurement apparatus. Moreover, the input pulse has its own shape, which is certainly not a perfect impulse. To account for these two factors, the input pulse shape must also be recorded and its effect included in the subsequent analysis. In the case of in situ measurements, due to the distance between the two ends of the fiber (4.5 to 8.1 km), the input and output measurements had to be done at two very different times; therefore the reproducibility of the optical pulses had to be excellent, which was confirmed prior to the in situ field measurements.

Also, the pulse repetition rate had to be sufficiently slow to not have any overlap from adjacent pulses. With this proviso, the periodicity of the signal was effectively ignored and hence the response of the fiber to only individual pulses was considered.

The measurement procedure consisted of:

1. record the input and output signals on the fiber;
2. transfer this sampled and digitized data to the computer;
3. execute a fast Fourier transform on both recorded signals to obtain the frequency spectrum of each;
4. obtain the frequency response of the fiber by dividing the Fourier transform of the output signal by that of the input signal.

The second chapter will cover the design and testing of the optical pulse generator. The third chapter will describe the special devices necessary to observe and record the weak received signals at the far end of the fiber such as the detector, the amplifier, the sampling scope, etc. The fourth chapter will be devoted to the data processing techniques such as the signal averaging, the data transfer from the sampling scope to the computer, the windowing, the FFT program, etc. The fifth chapter will present the results obtained in the field and in the lab, and finally the sixth and last chapter will be devoted to a discussion of the results.

2. OPTICAL PULSE GENERATOR

The optical pulse generator that was built consisted of a 1.3 μm InGaAsP semiconductor laser diode driven by an avalanche transistor. Fig.2 shows a schematic of the circuit; it is an adaptation of the classical mercury-switch-charged transmission line pulse generator and was first described in reference 1.

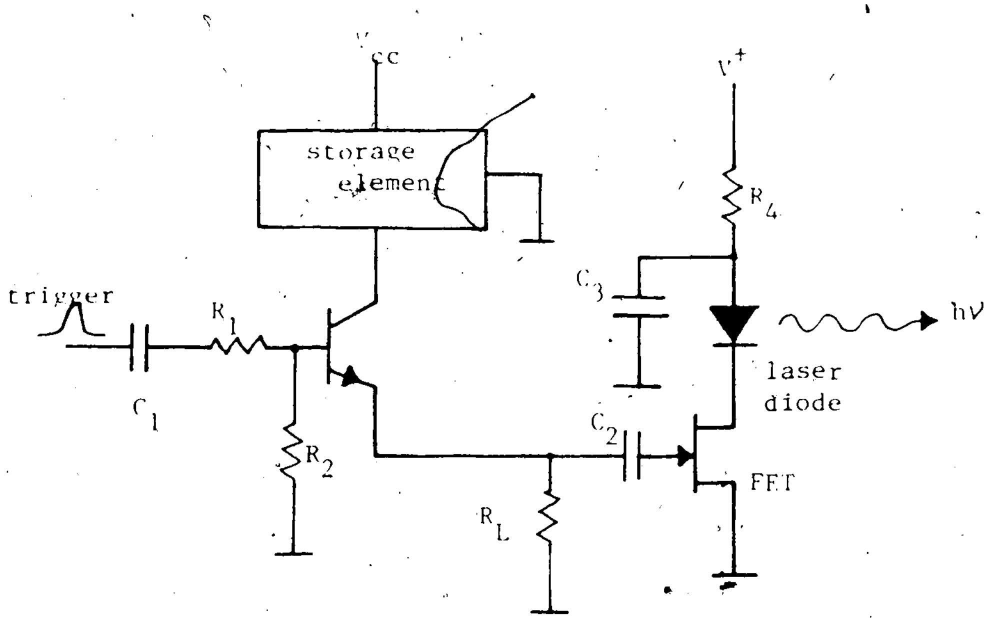


Fig.2 Optical impulse generator.

The first part of this chapter will cover the design of the electrical impulse generator and an evaluation of its performance. In the second part, the optical impulse generator as a whole will be discussed.

2.1 Electrical Impulse Generator

The electrical impulse generator circuit that was designed was originally intended to be used with a variety of lasers having differing characteristics. For example, the first tests were realized on RCA's SG-2001 laser diodes, which have a threshold current in the vicinity of 4 amp. Subsequent tests were done with the Northern Telecom's QLS-3A 1.3 μm InGaAsP lasers having the much lower threshold current of 110 mA. Thus, since very different magnitudes of current were required for driving different laser diodes, an electrical impulse generator was built that could be readily altered to change the output current.

There exist various devices to generate narrow electrical pulses. Examples are tunnel diodes, step recovery diodes (SRD), and avalanche transistors. The commonest form of solid-state pulse generator with subnanosecond risetime uses a transistor operating in the avalanche region. Huang [2] discussed the breakdown conditions for an n-p-n transistor under various base circuit configurations, and in particular showed how the onset of avalanche breakdown can be controlled by choice of an external base-emitter resistance R_B. Furthermore, it is possible to trigger the onset of avalanche breakdown externally. Using a transistor to discharge a charged transmission line, Andrews [3] produced a 40-V pulse with a risetime of 400 picoseconds (1 psec = 10^{-12} sec.), and Pfeiffer [4] obtained a 10-V pulse with a 120 psec risetime. While avalanche voltages can range as high as several hundred volts, the rate of rise of the voltage pulse generated usually remains constant at around 0.03 to 0.2 V/psec, resulting in poor risetimes for high voltage pulses applications [5].

The fastest solid-state device commercially available is the tunnel diode pulser [6]-[8], producing risetimes of around 20 to 25 psec. However, amplitudes are only 0.25 to 0.4 V, and the rate of rise is only about 0.01 to 0.02 V/psec. For these reasons, this device was discarded. Larger rates of rise with solid-state devices can be obtained using SRD's or "snap-off" diodes [9], usually operated in shunt with a transmission line to increase the rate of rise of a slower voltage step. For example, Schwartze [10] reported a circuit containing avalanche transistors and SRD's which would produce a 16 V , 80 psec risetime pulse.

Consequently, based on the very promising features of this last circuit, the design of a pulse generator using an avalanche transistor followed by successive pulse-sharpening stages using SRD's was considered. It should be recalled, at this point, that our goal was to generate short *optical* impulses having a frequency roll-off in the vicinity of 1.5 gigahertz (GHz), or, in the time domain, a duration full-width-at-half-maximum (FWHM) of ~300 psec. The avalanche portion of the proposed avalanche transistor-SRD generator was designed and 500 psec pulses were obtained. Prior to the design of a pulse-sharpening portion, the avalanche transistor was used to pulse a 1.3 μm laser diode.

Q-switched optical pulses considerably narrower (120 psec) than originally expected were obtained. Consequently, the design of the pulse-sharpening portion was dropped; (an excellent article from Tielert [11] describes a pulse circuit based on two SRD step-sharpening stages.)

In the next subsection, a review of the avalanche breakdown theory will be presented, followed by a discussion of criteria for choosing the transistor and the charge storage element. The experimental results obtained with this electrical impulse generator will then be given.

2.1.1 Principle of Avalanche Breakdown

The principle is described in the literature (see [12]-[15]) and will therefore be treated only briefly here. The maximum reverse-bias voltage which may be applied before breakdown between the collector and base terminals of a transistor, under the condition that the emitter lead be open-circuited, is represented by the symbol BV_{CEO} This breakdown voltage is a characteristic of the transistor alone. Breakdown occurs because of avalanche multiplication of the current I_{CO} that crosses the collector junction. As a result of this multiplication, the current becomes MI_{CO}, in which M is the factor by which the original current I_{CO} is multiplied by the avalanche effect. (We neglect the leakage current, which does not flow through the junction and is therefore not subject to avalanche multiplication). At a large value of collector to base voltage, namely BV_{CBO} (the

BV notation means breakdown voltage), the multiplication factor M becomes nominally infinite and the region of breakdown is then attained as shown in Fig.3 (taken from [12], p.194). Here the current rises abruptly, and large changes in current accompany small changes in applied voltages.

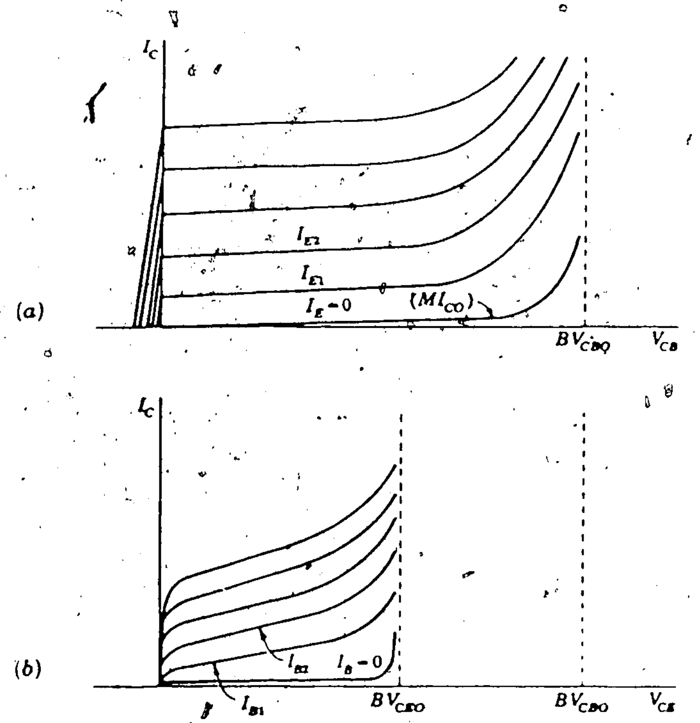


Fig.3 (a) Common-base characteristics extended into breakdown region. (b) Idealized common-emitter characteristics extended into breakdown region.

Fig.3(a) shows the common-base characteristics of a typical n-p-n junction transistor extended into the breakdown region. The curve for $I_E=0$ is a plot of MI_{CO} as a function of the voltage between the collector and base

(V_{CB}). The abrupt growth in I_C as BV_{CBO} is approached is evident in this graph.

It can be shown [12] that the collector-to-emitter breakdown voltage with open-circuited base, designated BV_{CEO}, is given by

$$BV_{CEO} = BV_{CBO} \sqrt[n]{1/h_{FE}} . \quad (1)$$

For example, with a germanium n-p-n transistor (n=6) with h_{FE}=200, equation 1 yields to

$$BV_{CEO} = 0.41 BV_{CBO} . \quad (2)$$

Idealized common-emitter characteristics extended into the breakdown region are presented in Fig.3(b).

Now if the base is connected to the emitter through a resistor R_B, as shown in Fig.4 (taken from [12]), the breakdown voltage, designated by BV_{CER}, will lie between BV_{CEO} and BV_{CBO}.

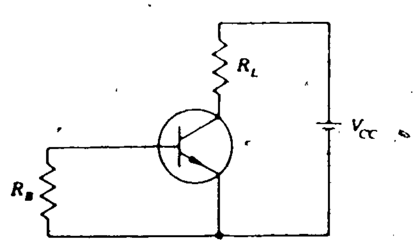


Fig.4 Common-emitter circuit with base resistor R_B.

To estimate BV_{CER}, some simplifying assumptions can be made concerning the emitter-junction diode. The semiconductor diode (in general) exhibits a threshold voltage V_γ in the forward direction. That is, until the forward voltage is about 0.2 V in germanium or 0.6 V in silicon, the forward current is very small. It will be assumed that until this threshold voltage of V_γ has been reached,the collector current will flow entirely to the base and hence through R_B.

It will also be assumed that once the threshold voltage is exceeded, nearly all the additional collector current will flow through the emitter junction, and the corresponding breakdown voltage is BV_{CEO}. Therefore, when the collector-to-emitter voltage is larger than BV_{CEO} and the threshold voltage of the emitter junction is reached, breakdown will occur. On this basis, breakdown will occur when the collector current MI_{CO} satisfies

$$M \cdot I_{CO} \cdot R_B \approx V_\gamma . \tag{3}$$

BV_{CER} will then be given by

$$BV_{CER} \approx BV_{CEO} \sqrt[n]{1-(I_{CO} R_B / V_\gamma)} . \tag{4}$$

The value of BV_{CER} for $R_B=0$ (when the base is short-circuited to the emitter) is denoted by the symbol BV_{CES}. Equation (4) suggests that $BV_{CES} = BV_{CEO}$. However, the presence of the base-spreading resistance R_b must be remembered, and R_B should be properly replaced by R_B+R_b. Accordingly, even when $R_B=0$, BV_{CES} is lower in magnitude than BV_{CEO}.

Equation 4 [12] was derived using the assumption that before breakdown, the current through R_B was very large in comparison with the emitter current. If R_B is made so large that this condition is not satisfied, then equation (4) is not applicable. Finally, it should be noted that, after breakdown has occured, the collector and the emitter currents will become very large in comparison with the base current. Therefore, at large currents the presence of R_B makes no difference, and the voltage across the transistor will drop from BV_{CER} to BV_{CEO} . Fig.5 was taken from [12] and shows plots of the collector current against the collector to emitter voltage extending into breakdown region.

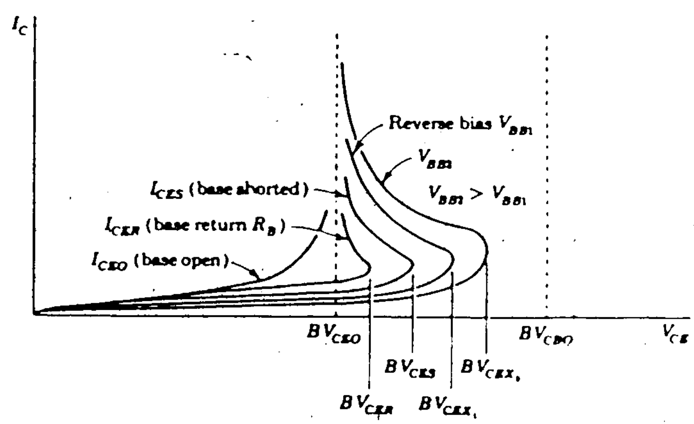


Fig.5 Plot, extended into the breakdown region, of collector current against V_{CE} for various connections to the base.

The breakdown voltage may also be increased by returning the resistor R_B to a voltage V_{BB} , as shown in Fig.6, which provides some back bias for the emitter-junction. In this case, the condition which

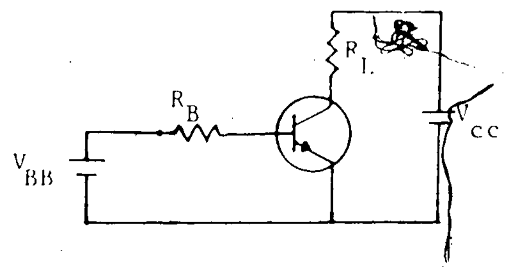


Fig.6 Common emitter transistor with a bias V_{BB} applied to the base.

determines the onset of breakdown is

$$MI_{CO}(R_B+R_b)=V_\gamma+V_{BB} \tag{5}$$

And the breakdown voltage, now represented by the symbol BV_{CEX} is given approximately by

$$BV_{CEX} = BV_{CBO}\sqrt[n]{1-(I_{CO}(R_B+R_b)/(V_\gamma+V_{BB}))} \tag{6}$$

2.1.2 Avalanche-Mode Transistor Circuits

The typical transistor volt-ampere characteristics of Fig.5 are reproduced in Fig.7(a), and a basic circuit using a capacitor as the charge storage element is indicated in Fig.7(b)[12].

For the avalanche operation of the circuit of Fig.7(b), the load line should be selected to yield a single stable point in the low-current region (see Fig.7(a)). The supply voltage charges the capacitor through R_C to a voltage V_{cc}.

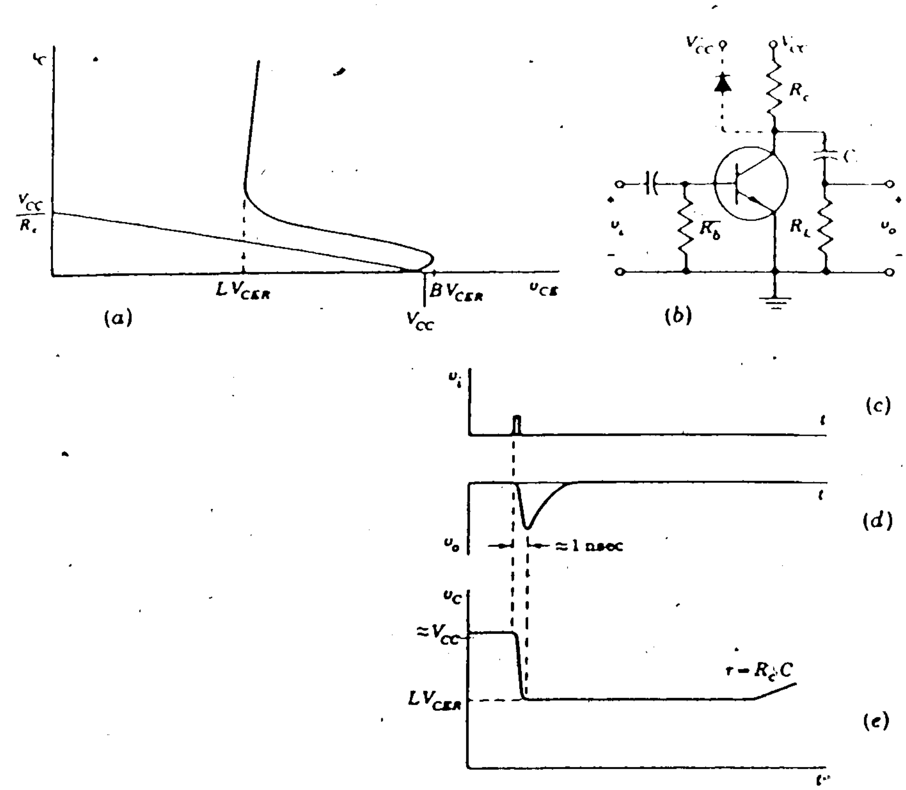


Fig.7 (a) Volt-ampere characteristics of avalanche-mode transistor. (b) Circuit of a pulse generator($R_1 < R_c$). (c) Triggering pulse. (d) Output waveform.(e) Waveform at the collector.

slightly less than the breakdown voltage BV_{CER} . A pulse shown in Fig.7(c), or some positive-going signal,applied to the base lowers the breakdown voltage of the transistor, and the capacitor discharges rapidly through the transistor and the resistance R_L. The voltage across R_L shown in Fig.7(d) increases rapidly, while the collector voltage drops rapidly. The speed with which these voltages change is determined by how quickly the transistor makes the transition from its low-current state to the state in which an avalanche discharge is established. This transition time is of the order of nanosecond and is the characteristic of the avalanche breakdown switching mode that is extremely useful in fast circuits.

Having reached a peak value, the output voltage then decays to zero as the capacitor discharges. The collector voltage, as shown in Fig.7(e), starts close to V_{CC} and drops at the same high speed to the latching voltage LV_{CEP}, which, as the symbol indicates, is a function of the base resistance and is very close to BV_{CEO}. Even after the capacitor has discharged, the collector voltage remains for a time at the latching voltage since an interval of time is required to allow the transistor to recover and return to its initial state. During this interval, a small transistor current flowing through R_C maintains the collector at the lower voltage. Finally, as the transistor approaches complete recovery, the collector voltage returns again to V_{cc} at a rate determined by the time constant $R_C C$.

The pulse shape obtained with the circuit shown in Fig.7(b) is a damped sinusoid. The duration of the main pulse depends on the values of R_L, C, and the parasitic inductance of the discharge capacitor. The rise-time and amplitude of the pulse depend on these same parameters in addition to the voltage-current characteristics of the switching transistor. For example, it is very difficult to specify the pulse amplitude across the load resistance R_L during switching, since the collector to emitter voltage varies during the switching transient, i.e., there is no constant internal resistance R_b. Thus the capacitor voltage simultaneously starts to decrease from BV_{CER} because of discharging. Only if the capacitance is very large, and hence the voltage can remain constant as the transistor is being switched on, can the maximum pulse amplitude of $(BV_{CER} - BV_{CEO})/R_L$ be reached. However under these conditions, the rise-time of the pulse is very long. For all the preceeding reasons, the capacitor was rejected as a storage element in the discharge circuit.

2.1.3 Transmission-Line Impulse Generator

Different conditions exist when a transmission line is used as the charge storage element in place of the capacitor. An avalanche transistor pulse generator whose pulse amplitude and width are separately controllable is shown in Fig.8(a). Here the capacitor has been replaced by an open-circuited delay line of characteristic impedance Z_0. An equivalent circuit useable for calculating the waveforms is shown in Fig.8(b), where avalanche breakdown is represented by the closing of the switch S.

When the positive trigger pulse is coupled through the capacitor to the base of the transistor, avalanche breakdown is initiated. The emitter-junction becomes forward biased and injects carriers into the base. Avalanche regeneration occurs and the collector voltage drops from BV_{CER}, which is practically equal to V_{CC}, to BV_{CEO} producing a negative

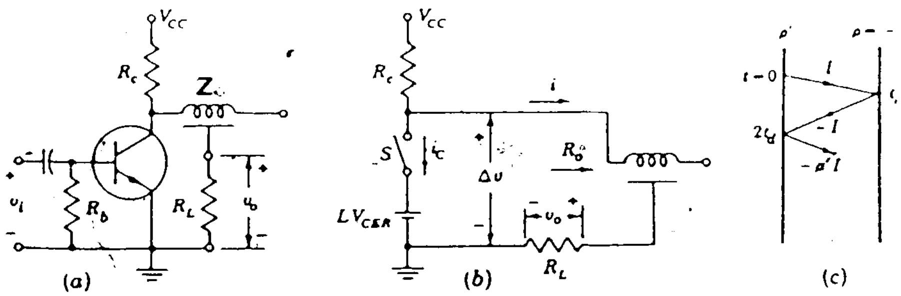


Fig.8 (a) Transmission line pulse generator. (b) Equivalent circuit after the transistor has been triggered. (c) Reflection chart for the line current.

voltage step $\Delta V=-V$ across the open-circuited line. This step has an amplitude of $V=-(V_{cc}-BV_{CEO})$, and since the initial line current is zero, then a current step

$$i(0+)=-V/(Z_o+R_L)=I \tag{7}$$

starts down the line at $t=0+$. A constant current through the collector is thus maintained while the step is travelling to the open end of the line. When this current wave reaches the end of the line at $t=t_d$, the one-way transit time for the line, it is reflected as a current step $-I$ (the reflection coefficient ρ for an open circuit termination is -1). At $t=2t_d$, this negative current step $-I$ reaches the transistor end of the line and, as indicated in the reflection chart of Fig.8(c), is again reflected as $(-I)\rho'$, where ρ' is the reflection coefficient at the collector. Hence, the total line current at $t=2t_d+$ is

$i(2t_d+)=I-I-I\rho'=-I\rho'$

$$=(-V/(Z_0+R_L))\ ((R_L/Z_0)-1)/\ ((R_L/Z_0)+1) \qquad (8)$$

If a single pulse is desired, i.e., $i(2t_d+)=0$, then the load resistance R_L must be equal to the delay line characteristic impedance Z_0. In this case, when the current pulse reaches the collector, no reflection occurs and the magnitude of the current pulse drops to zero. The line then slowly begins again to charge towards BV_{CER} and the cycle is repeated.

Altogether, the circuit of Fig.8(a) would develop across R_L a pulse whose duration $2t_d$ is controllable by adjusting the transmission line delay time, that is, by changing its length, and whose amplitude V_0 is adjustable by changing R_L, Z_0 and V in accordance with the relationship

$$V_0=+i(0+)\ R_L=+VR_L/(Z_0+R_L). \qquad (9)$$

For the case of single pulse generation, i.e., with $R_L=Z_0$,

$$V_0=+i(0+)R_L=\ +V/2=(V_{cc}-BV_{CEO})/2. \qquad (10)$$

A practical way to decrease the characteristic impedance Z_0 is to connect a few 50 ohm coax cable lines in parallel. For example, 10 lines were connected in parallel in order to obtain a 4 amp current pulse for the operation of the SG-2001 laser diode. Thus, ten 50 ohm lines in parallel gives a line with characteristic impedance of Z_0 = 5 ohm. For this case, using (7) and the experimental values of Z_0=5 ohm, BV_{CEO}=50 V and V_{cc} = 160 V yields

$$I=-V/(Z_0+R_L)=\ (V_{cc}-BV_{CEO})/(Z_0+R_L), \qquad (11)$$

or I=(160-50)/(5+5)=11 amp.

For the 1.3 μm laser diode, the threshold current was only 110 mA so a single 50 ohm coaxial cable was used. Thus, using (11) yields

I=(160-50)/(50+50)=1.1 amp.

Attenuators were used to further reduce the current pulse amplitude to the desired level.

2.1.4 Transistor Selection

The published values of BV_{CEO} and BV_{CBO} for specific transistors are hardly reliable as criteria for choosing a transistor, because they often represent only guaranteed values. Measured values could be up to three times as large as the minimum values specified [16]. The values obtained from data sheets on the maximum collector current are similarly unreliable. Depending on the pulse width in the nanosecond region, some transistors can switch up to 100 times their normal collector current. Another difficulty arises with the published transit frequency, f_t, in that this linear mode parameter is not simply related to the switching speed in the avalanche mode. However, according to Zuhlke [17], only fast switching transistors are also fast avalanche transistors.

Based on the above remarks and preliminary laboratory testing, the Motorola MMT 3904 was chosen for this project. It is a transistor designed for general purpose switching and amplifier applications. Other reasons that dictated this choice were the very small dimensions of the device, its ability to sustain high enough collector-to-base voltage without avalanching (160 volts), its relatively short rise (13 nsec) and fall (11 nsec) times, and its availability. The data sheets for the MMT 3904 are reproduced in Appendix 1.

2.1.5 Pulse Width

The velocity of propagation of electromagnetic waves in the coaxial cable used (RG 58 A/U) is equal to 66% of the speed of light in vacuum. Also, as it was shown in the preceeding subsection, the current output pulse, generated by the avalanche circuit has a duration equal to the time a wave takes to propagate from the transistor to the end of the discharge line and back. For example, in order to generate an impulse having a duration of one nanosecond, the length l of the discharge line has to be

$$l=(1/2)\times(1\times10^{-9}/(0.66\times3\times10^{8}))\ \text{meters}=10\ \text{cm}.$$

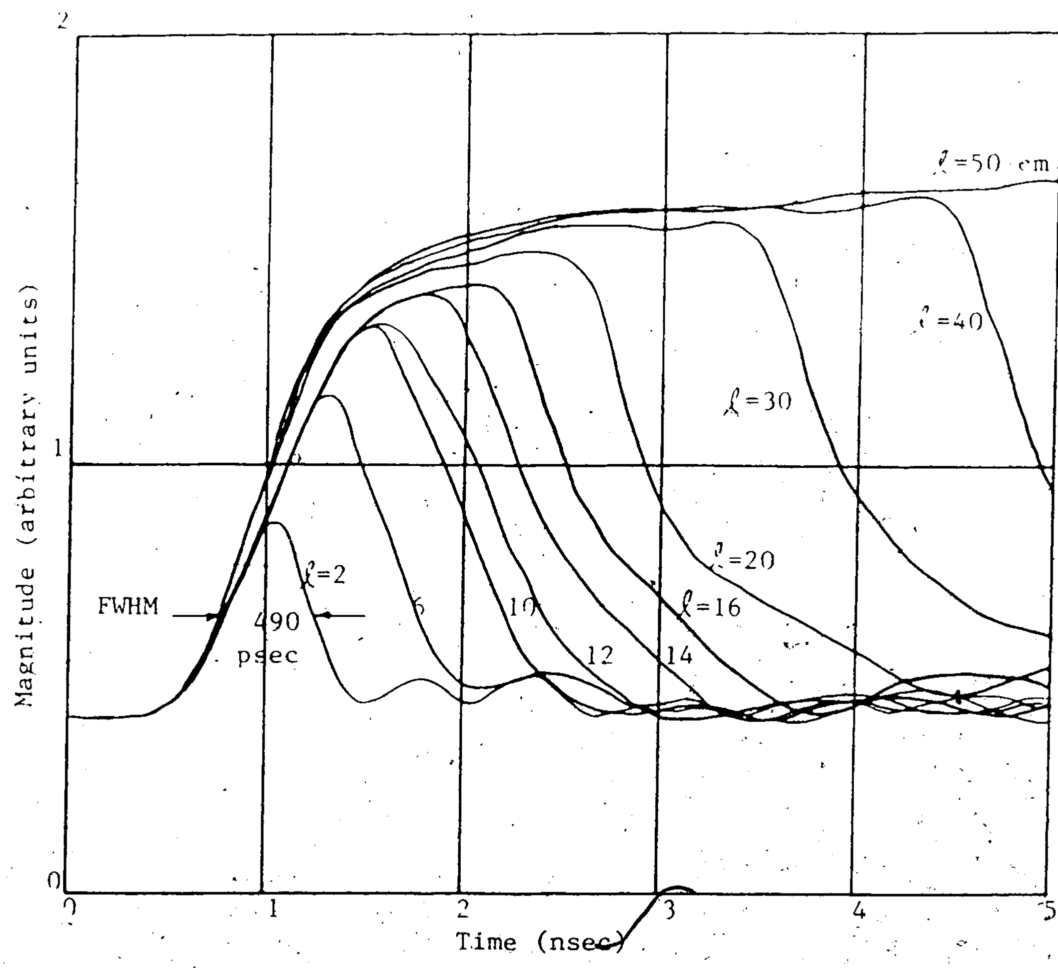


Fig. 9 Output current pulses from the avalanche transistor when the discharge line length l is varied.

Fig.9 shows the change in the waveshape of the current pulse when the discharge line is gradually shortened from 50 cm to 2 cm. The slope of the rising edge, for all curves, is approximately the same and is caused mainly by parasitic inductance in the circuit. For example, for a 50 cm discharge line, the rise time of the pulse (defined as the interval of time required for the pulse to rise from 10% to 90% of its plateau value) is about 1.4 nsec. But, in theory, if a 50 ohm delay line of 2 cm is used, the pulse width across the load resistance should be

$$\Delta\tau=2/(0.66\text{x}3\text{x}10^{-8})\text{sec}=200 \text{ psec}.$$

Seemingly, the avalanche transistor would be turned off by the reflected pulse before the output current pulse has reached its full swing! In fact, for any line with a delay time shorter than one half of the observed transistor rise time, i.e., 700 picoseconds, this feature is observed. It is clear from the results illustrated in Fig.9 that for any length shorter than ~16 cm, the output current pulse does not reach its full amplitude. For a discharge line of 2 cm, the output current pulse amplitude is down to one third of the one obtained with a 50 cm line, and for any lengths shorter than 2 cm, the reflections following the main pulse were of unacceptably large levels (up to one half of the peak amplitude). For this reason, it was decided that a length of 2 cm was optimal for the actual circuit layout, bearing in mind that one of the objectives was to obtain a minimum pulse duration.

Although it was possible to operate the circuit in the avalanche mode for V_{cc} values between 100 and 230 volts, V_{cc} was fixed at the middle of this range (i.e. 160 volts) to realize both reliable operation and a minimum number of external attenuators on the output line (each external attenuator unavoidably slightly degrades the pulse rise-time).

Thus, the final version of the electrical impulse generator was delivering a signal having an amplitude of 18 volts and a duration of 490 psec into a 50 ohm load. A typical output impulse and its frequency spectrum are presented in Figs. 10 and 11, respectively. The frequency data was obtained with the FFT program described in chapter 4.

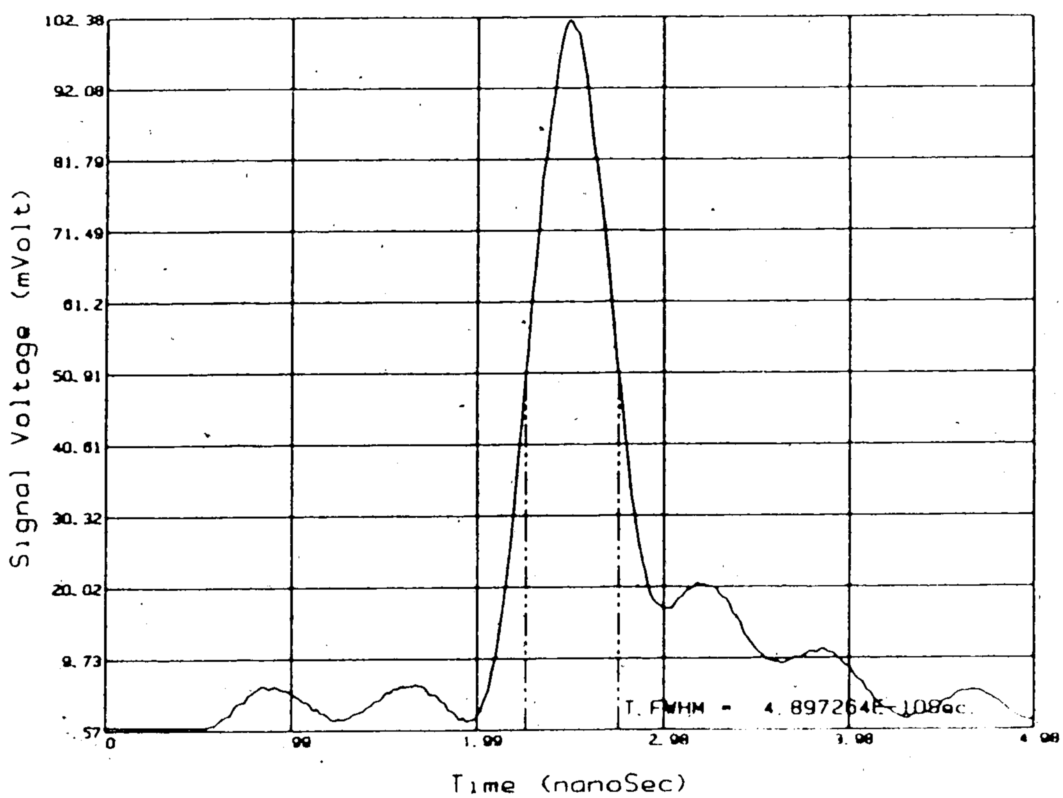


Fig.10 Typical output signal from the avalanche transistor pulse generator.

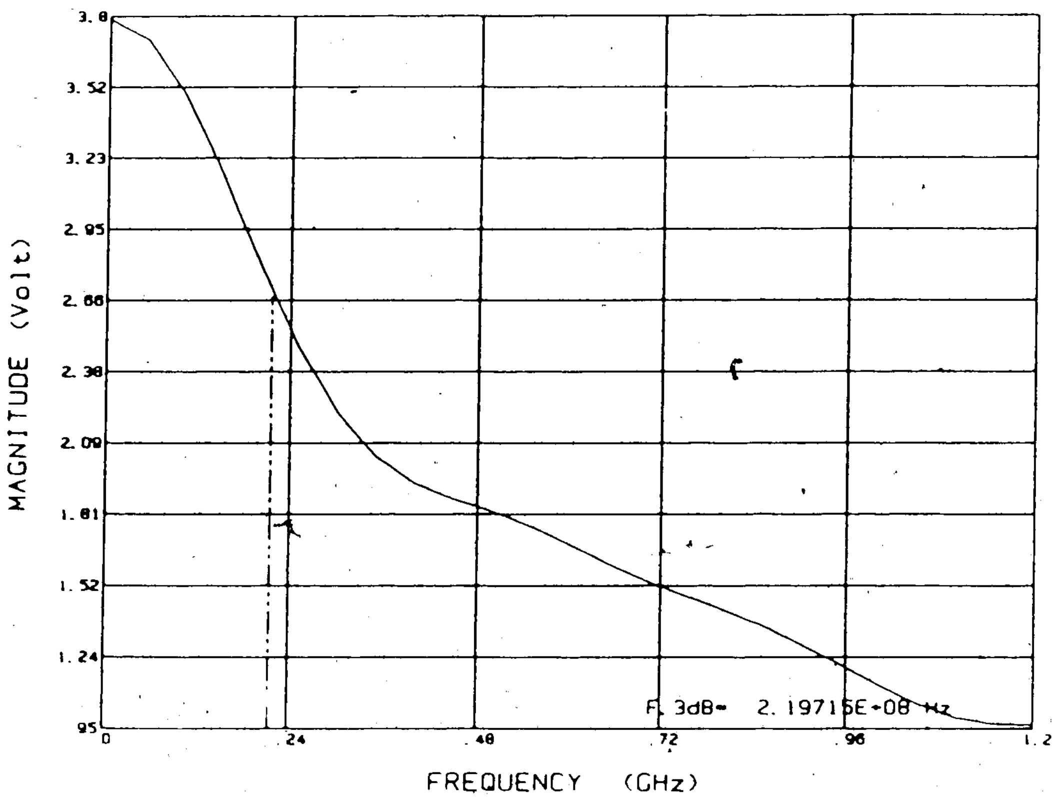


Fig.11 Frequency content of a typical generated electrical pulse.

2.1.6 Further Pulsewidth Reduction

The rising edge of the current pulse could be sharpened by reducing the physical dimensions of the circuit using microstrip design procedures. Also, in the waveform labelled l=20 cm in Fig.9, it is noticeable that the trailing edge is not very sharp. The reason for this is that the transistor internal impedance changes during switching [16]. This substantial distortion and the fact that the duration of the output pulse is fixed by the length of the line are in fact the main disadvantages of this pulse circuit. Nevertheless, it is possible to lessen these undesirable effects. Two methods will be mentioned here, since a very similiar version of the circuit illustrated in Fig.8.(a) was finally used.

One way to decrease the fall-time is to have the avalanche transistor not turned off by the reflected pulse but by an additional pulse generated by a second avalanche transistor at the end of the delay line [18](see Fig.12(a)).

Another way to lessen the fall-time is to use a second transistor to generate a second pulse approximatly 350 psec after the original pulse, inverted through a pulse transformer, and added to the original pulse (see Fig.12(b)). A typical output pulse from this circuit is presented in Fig.13. This pulse has a damped-sinusoid shape but the positive portion is shorter than the original pulse (410 psec compared to 490 psec originally).

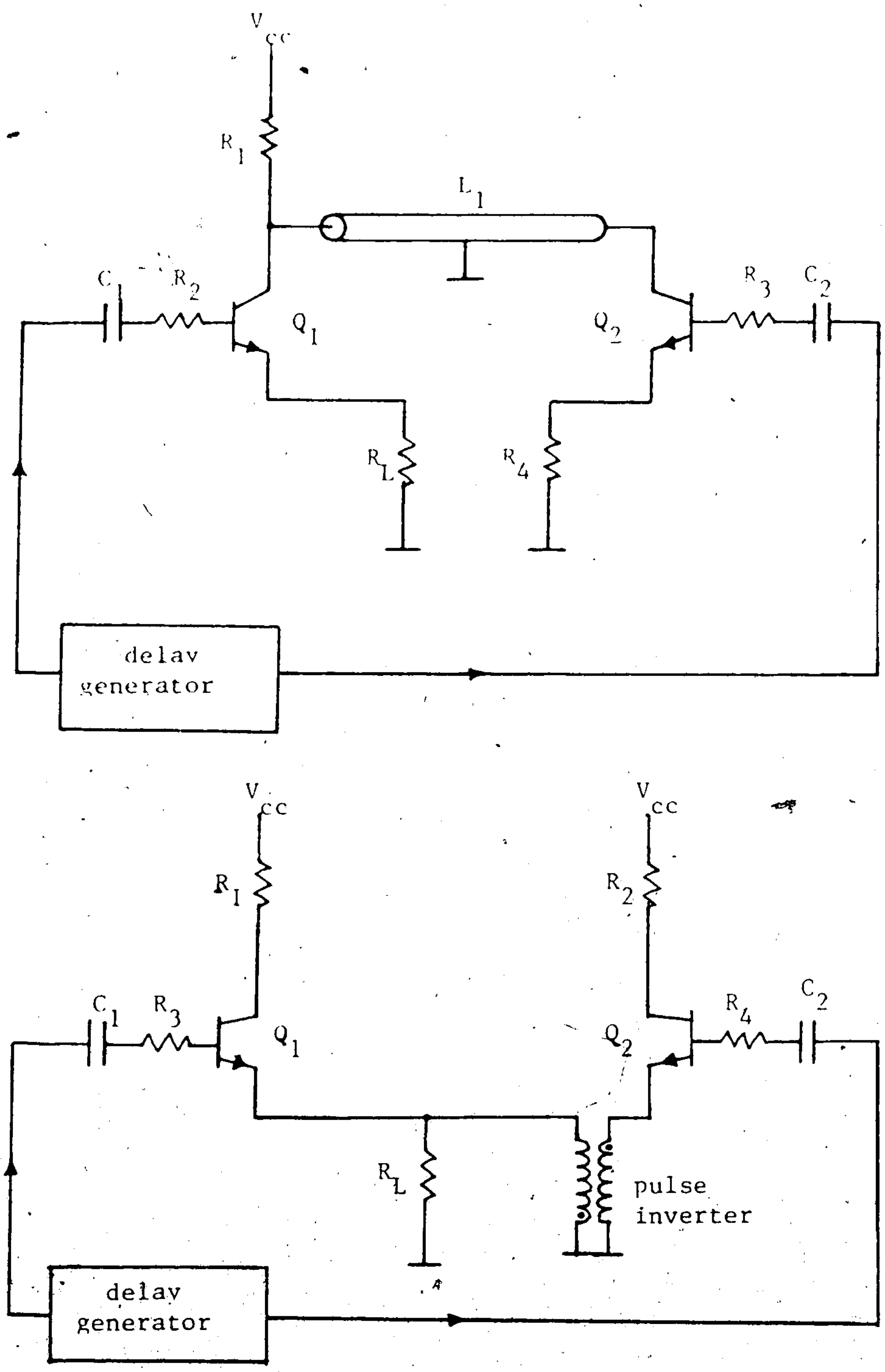


Fig.12 Electrical pulse generators using two transistors (a) at each end of the transmission line; and (b) in parallel.

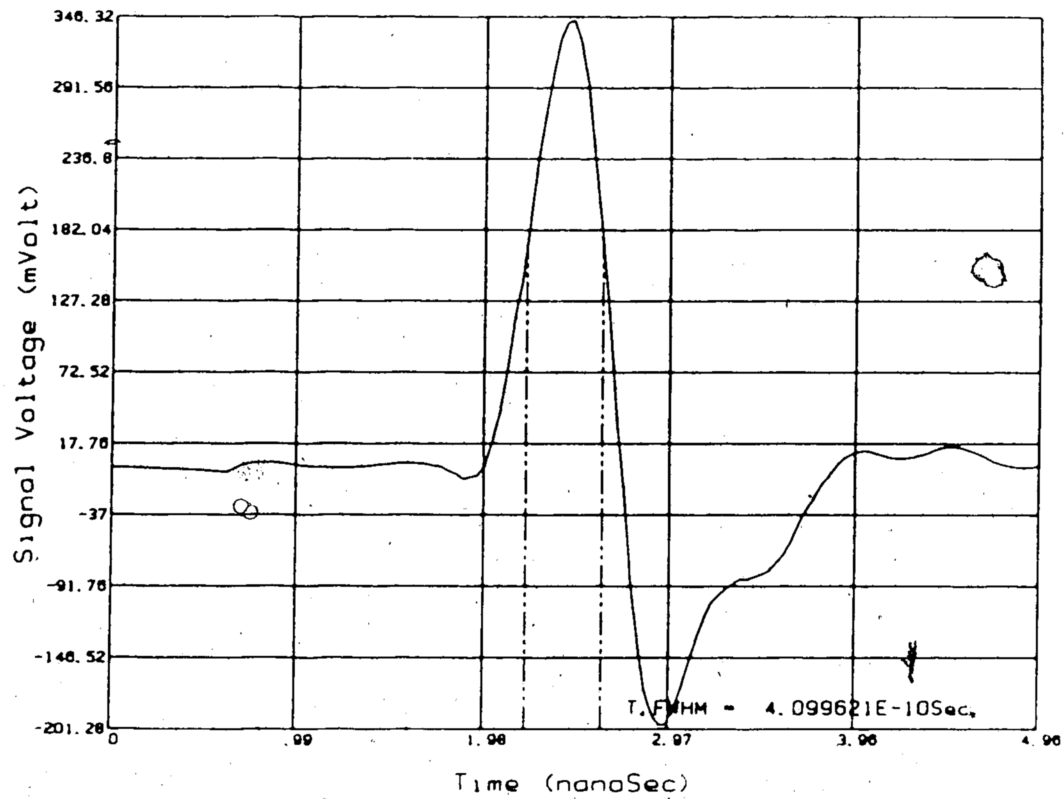


Fig.13 Output of a pulse generator using two transistors in parallel.

The negative portion (overshoot in Fig.13) of this new pulse would not have been of real concern if this circuit had been used to pulse the laser, since it would have been possible to simply add this pulse to a d.c. current source connected to the laser (a necessary complication to satisfy the requirement that a laser diode must never be subjected to a large reverse voltage and the d.c. source provides the laser threshold current). In this biasing configuration; the optical output power is limited to very small variations during the negative input pulse excursion.

2.2 Laser Diode Pulsing

The direct modulation performance of solid state heterostructure lasers has received considerable attention in recent years. To eliminate the well known lasing delay (for example, see [19]), the devices are usually operated with a continuous quiescent current near the threshold and the modulation is produced by a relatively small pulsed current superimposed on it. One of the principal features to have received both experimental [20-21] and theoretical [22-23] attention is the appearance of damped oscillations in the optical output when the laser is subjected to a step increase in the drive current. Ripper and Dyment [24] were the first, in 1968, to obtain internally Q-switched light pulses from a junction laser. Using specially fabricated diodes, they detected narrow bursts of light immediatly *after* the termination of the injection current pulse. These authors later reported [25] that at lower currents in the Q-switching region, a single light spike, whose width was about 300 psec, was observed. The optical impulse generator described herein uses this Q-switched laser mode.

The 1.3 μm laser diode pulsing circuit will be described first in this section. A review of semiconductor laser Q-switching phenomena will then be presented followed by a presentation of experimental results. Finally, a discussion on the stability of the generated optical pulses and on the design of a double optical pulse generator will terminate this section.

2.2.1 Laser Diode Pulsing Circuit

The laser diode quiescent optical output power was controlled through a very low frequency optical feedback loop. However, this part of the circuitry will not be discussed here. It will only be mentioned, with reference to Fig.14, that this circuit controls the laser quiescent current I (which is equal to the FET drain current I_d) by controlling the voltage V_1.

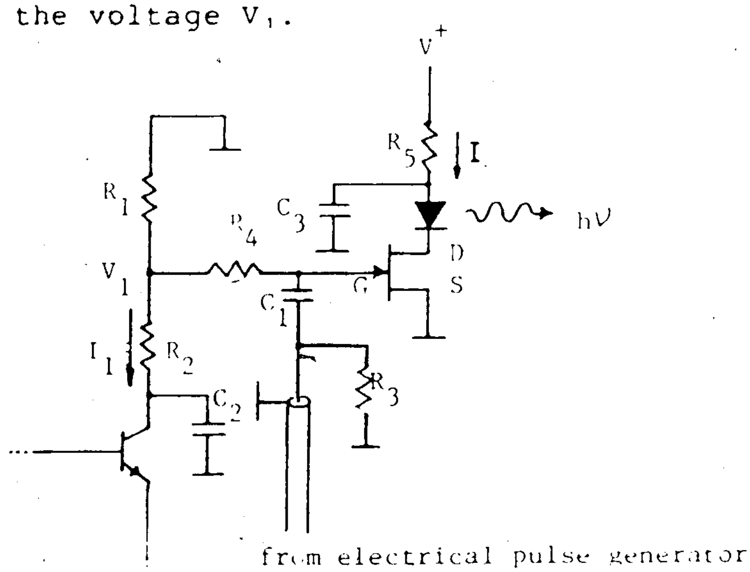


Fig.14 Laser diode pulsing circuit.

Because the gate-to-source leakage current, designated I_{GSS}, has a typical value of 10^{-10} to 10^{-12} amp, the input impedance at the gate electrode is extremely high and the voltage drop across R_4 is negligible. Thus the voltage V_1 is equal to the voltage between the gate and the source (V_{GS}). The laser quiescent current is controlled by varying the current I_1 flowing through R_1, which is equivalent to controlling the V_{GS} voltage:

$$I_d = I_{dss}\,(1 - V_{GS}/V_p)^2, \qquad (12)$$

where I_{dss} is the drain saturation current and V_p is the pinch-off voltage.

As mentioned earlier, the laser diode used for this project had a threshold current of ~110 mA. A NEC NE9002 microwave GaAs power FET was used to control the laser current. This n-channel device was chosen because of its high value of I_{dss}, namely 450 mA typical, for its very wide bandwidth (this FET is specified for amplifier and oscillator applications up to 20 GHz), and for its very small size. The data sheets for this transistor are reproduced in Appendix 1.

The non-d.c. part of the pulsing circuit shown in Fig.14 will now be examined. The short electrical pulses generated by the avalanche circuit are coupled through R_3 and C_1 directly to the gate terminal of the FET. C_1 is a 2 nanofarads (nF) coupling capacitor. The value of the resistor R_3 was experimentally chosen to provide a 50 ohm input impedance at the gate electrode, in order to prevent any signal reflection back to the avalanche circuit. Its value was 68 ohm. A typical characteristic curve for a n-channel FET is shown in Fig.15(a); Fig.15(b) represents an experimental power-current curve obtained with the 1.3 μm laser diode.

The specified values of I_{dss} and V_p for the NE9002 are 450 mA and -3.5 V, respectively. Thus in order to obtain a

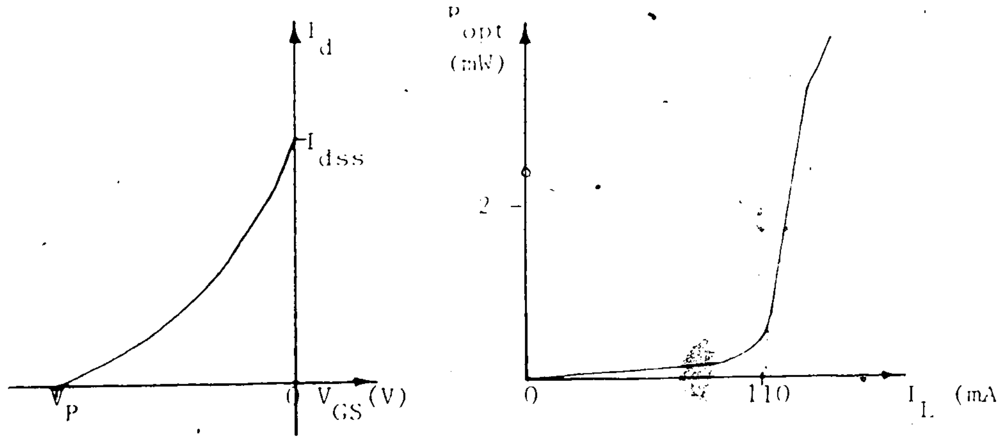


Fig.15 (a) Characteristic voltage-current curve of a typical n-channel FET. (b) Optical output power versus current for a 1.3 μm laser diode.

laser current of 110 mA, the required value of V_{GS} is obtained from (12):

$$I_d = I_{dss}\ (1 - V_{GS}/V_P)^2$$

$$V_{GS} = (1 - \sqrt{I_d/I_{dss}})V_P$$

$$= (1 - \sqrt{110mA/450mA})(-3.5)$$

$$= -1.77\ V$$

The capacitor C_3 is used to supply the necessary current pulse to the laser during the transient. Its value was chosen large enough (100 nF), to not have any noticeable effect on the shape of the drive pulse.

The dimensions of the pulsing circuit were minimum in order to limit parasitic inductances. For example, a 1 cm length of wire (size AWG 26) has an inductance of ~9 nanohenries (nH). Therefore, in a 50 ohm system, this relatively short element would have a time constant of

$\tau = L/R = 9 \times 10^{-9}/50$ sec

$= 188$ psec.

Thus the time required for the current through, or voltage across, this element to reach 100% of its final value in a 50 ohm system would be approximatly 5τ, or ~1 nsec. Clearly, the dimensions in the critical locations of the circuit had to be kept in the millimeter range. Examples are the connections between R_3, C_2 and the gate, between the source and the ground, between the laser diode and the drain, and also between the laser diode and C_3.

2.2.2 Q-Switching Laser Diodes

Dyment and Ripper [27] have proposed that the narrow pulses occuring after the termination of the injection current, i.e., the Q-switched light pulses, were controlled by trapping centres that behave like double acceptors. Since a detailed discussion and mathematical development of this trapping model are presented elsewhere [28]-[29] and are beyond the scope of this work, only the essential features of the model will be given.

In their singly ionized states, the semiconductor traps are optically absorbing energy and introduce a large loss into the cavity. Within certain ranges of temperature and current defined by region II of Fig.16 (for the L-137-19 diode), this loss is large enough to prevent any lasing throughout the entire duration of the current pulse.

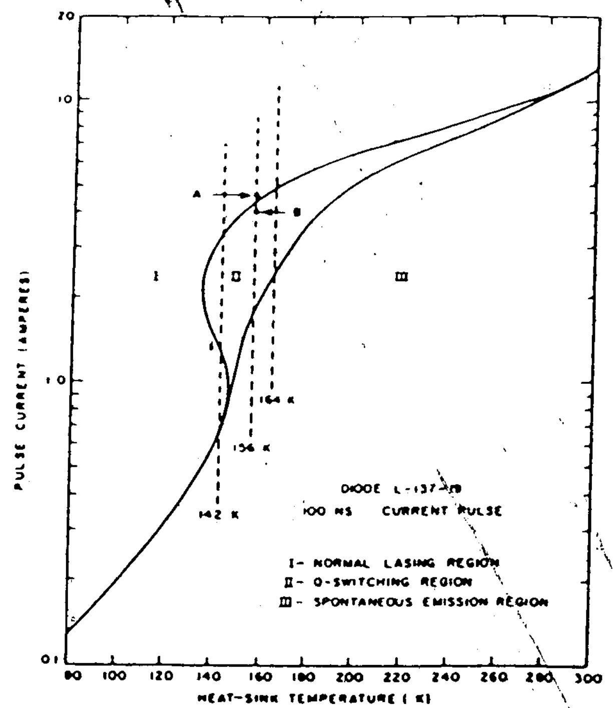


Fig.16 Quiescent current vs temperature for the L-137-19 laser diode.

Upon termination of the current pulse, most traps are rapidly transferred from their absorbing states (singly-ionized) to their nonabsorbing states (doubly-ionized). The resulting reduction in loss allows those injected carriers that have not yet spontaneously recombined to produceanarrow stimulated Q-switched light pulse. In Fig.16, this behavior is distinguished from the behavior in the normal region I (where lasing occurs during the injection current pulse) and the spontaneous emission region III (where no lasing occurs during or after the injection pulse).

Dyment, Ripper and Roldan reported in 1969 [25] that their measurements on various laser diodes showed that the single light spike behavior reported earlier [24] only occurs for the lower currents in the Q-switching region. At higher currents, additional light spikes were appearing, whose widths and spacings both decreased as the current was increased. Fig.17 [25] shows typical data.

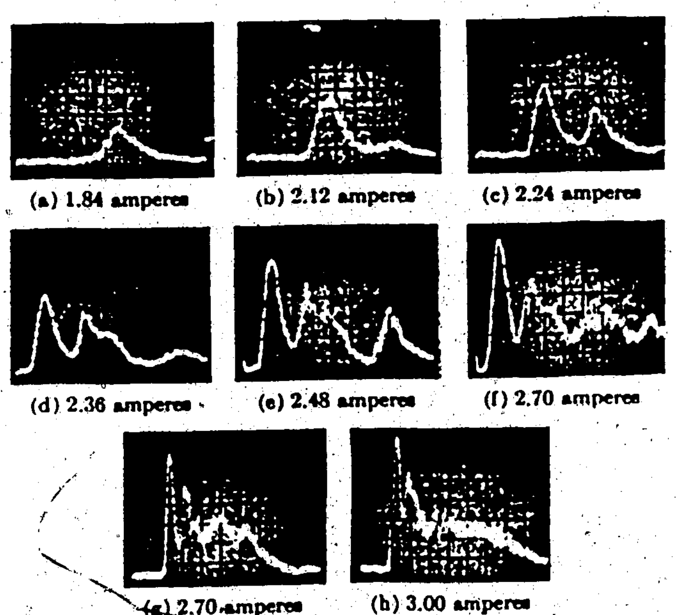


Fig.17 Q-Switched optical spikes.

The laser diode temperature was held fixed while the current amplitude was increased from the threshold for Q-switching to much larger currents. Near the threshold for

Q-switching, a single light spike, which was ~300 psec wide, was initially observed. As the current was increased, the amplitude of the initial spike increased and its width decreased. At sufficiently high currents, second and third spikes were observed by these authors.

2.2.3 Experimental Results

Dyment *et al*'s experimental procedure was adopted. The quiescent current of the laser was gradually increased from a value (~60 mA) much lower than the threshold current, while 220 mV, 480 psec pulses were applied to the gate of the FET. These pulses were applied at the low repetition rate of 3 KHz to limit the laser power dissipation.

When the laser quiescent current was maintained at 100 mA, the control voltage V_{GS} was equal to -1.87 V. The small-signal transconductance at this operating point can be calculated [26]

$$g_m = dI_d/dV_{GS} = -2\cdot(1-V_{GS}/V_P)\cdot(I_{ds}/V_P) \quad (13)$$
$$= -2(1-(-1.87)/(-3.5))\cdot(450/3.5)$$
$$= 120 \text{ mA/V}.$$

Thus, a pulse of 220 mV applied at the gate, when V_{GS}=-1.87 V, would generate a current pulse through the laser having an amplitude of 120 mA/V x 220 mV = 26 mA on top of the quiescent current of 100 mA. Thus, when the laser quiescent current is gradually increased from 60 mA to values approaching the threshold value of 110 mA, it is clear that lasing action starts at a quiescent current of

110 - 26 = 84 mA. Furthermore, at a quiescent current of 100 mA, the laser will experience a peak value of 100 mA + 26 mA = 126 mA. It was found experimentally that the onset of Q-switched light spikes was at a laser quiescent current of ~ 90 mA (for an input electrical voltage pulse of 220 mV). At this value of current , single Q-switched pulses were observed. When the laser current was further increased, multiple spikes appeared at the output of the laser. Results are presented in Figs.19 to 24 for the case where the input current pulse had a duration of 1.3 nsec. The shape of this current pulse is shown in Fig.25 and the experimental set-up used to record these waveforms if sketched in Fig.18.

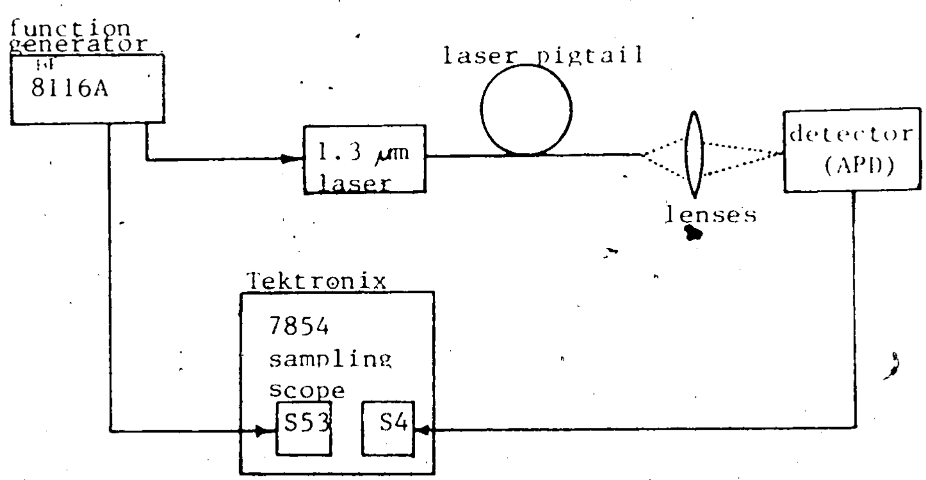


Fig.18 Q-Switched pulses measuring set-up.

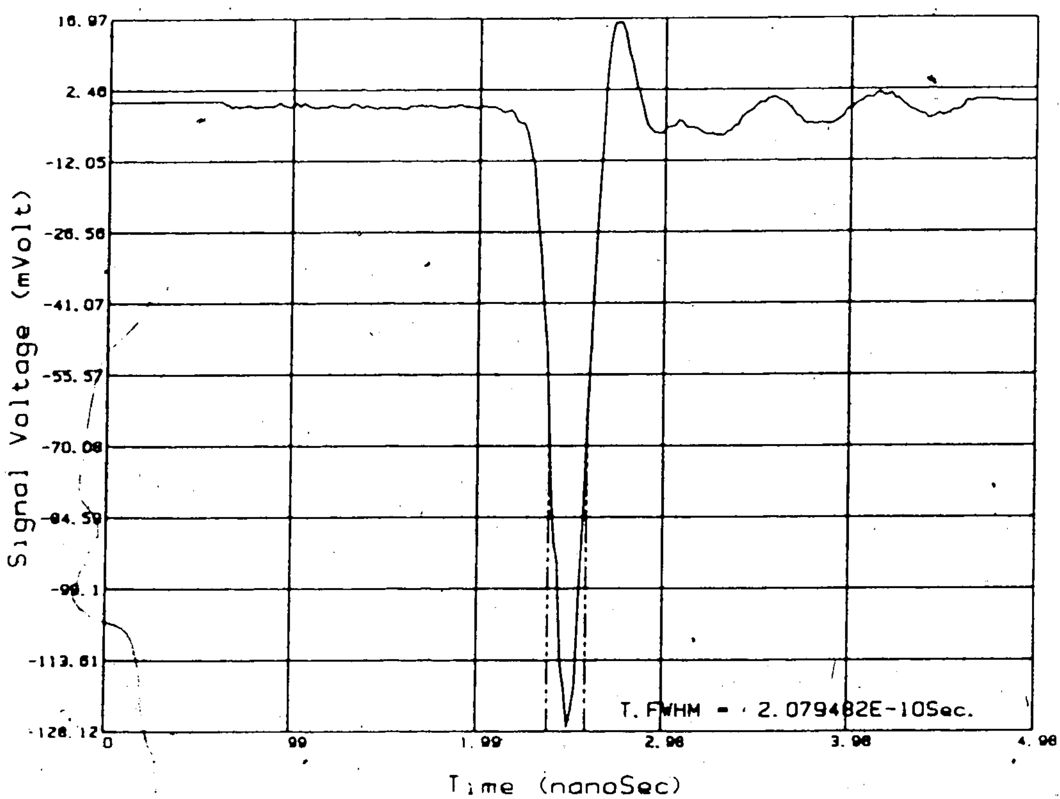


Fig.19 Laser output for a quiescent current of 90 mA.

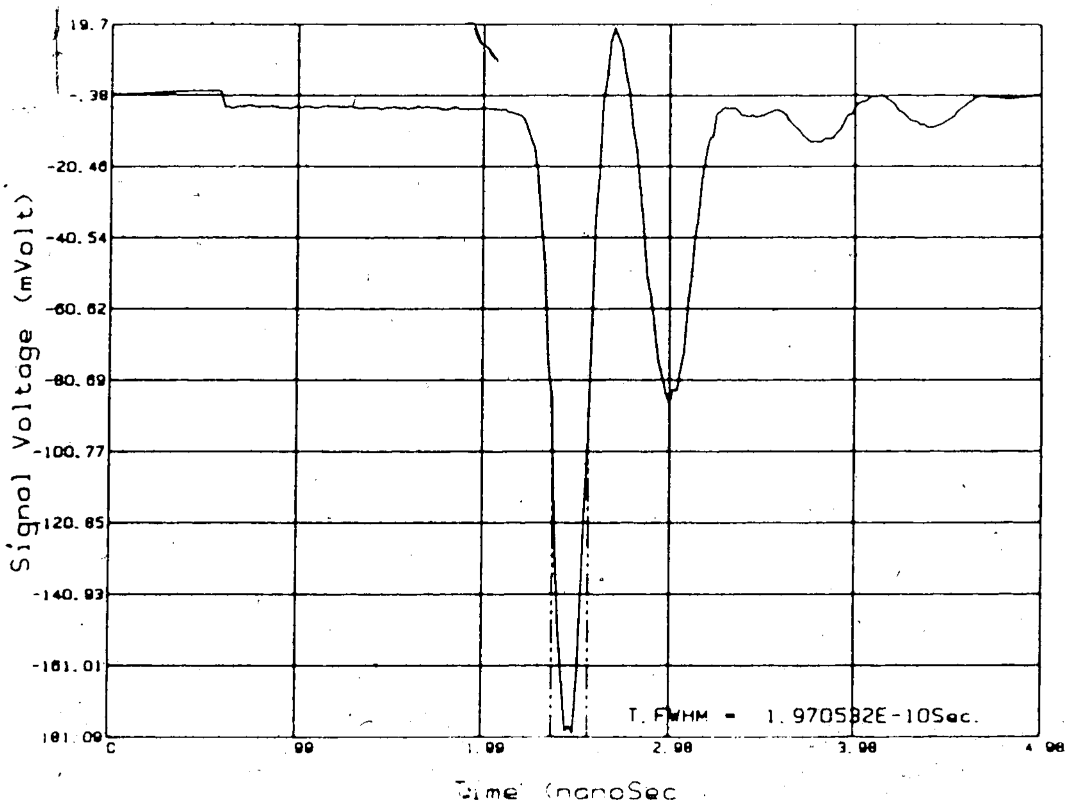


Fig.20 Laser output for a quiescent current of 91 mA.

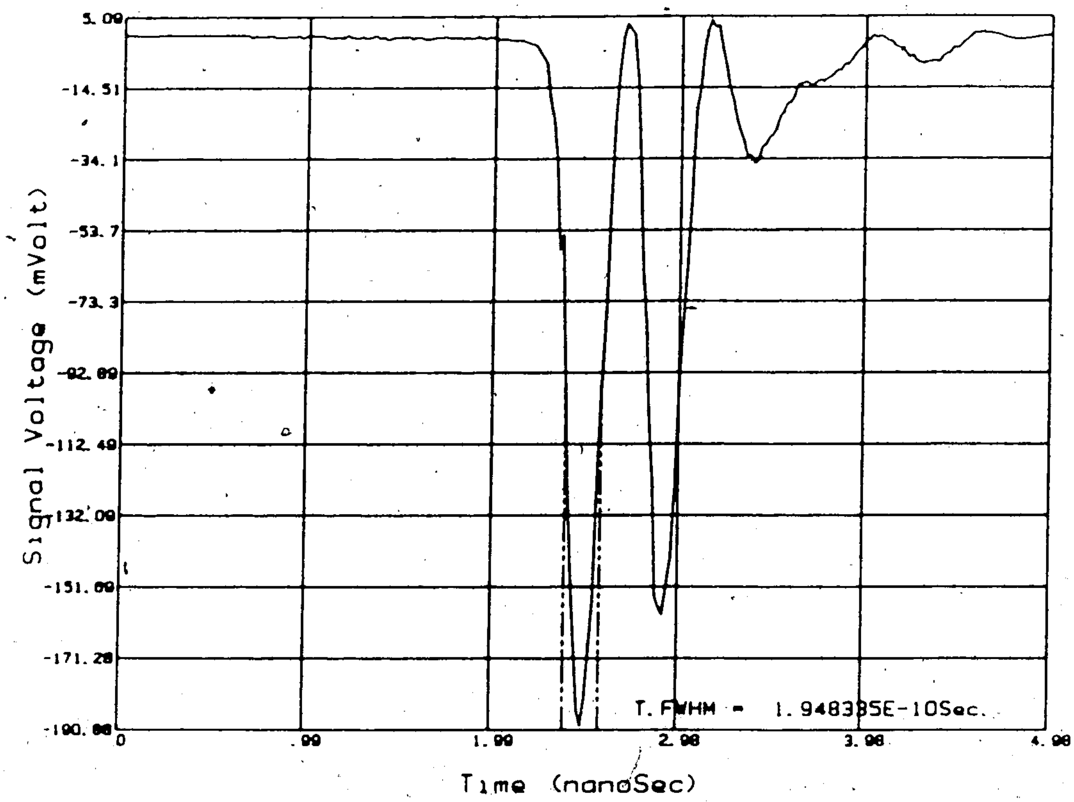


Fig.21 Laser output for a quiescent current of 92 mA.

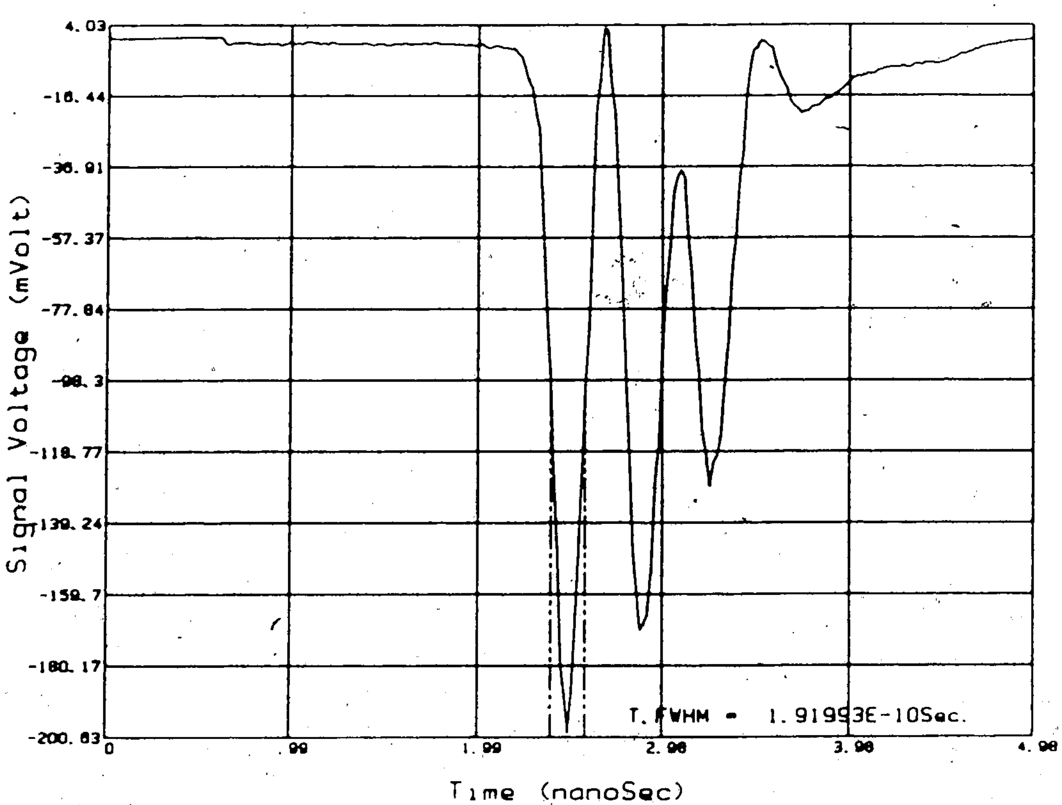


Fig.22 Laser output for a quiescent current of 93 mA.

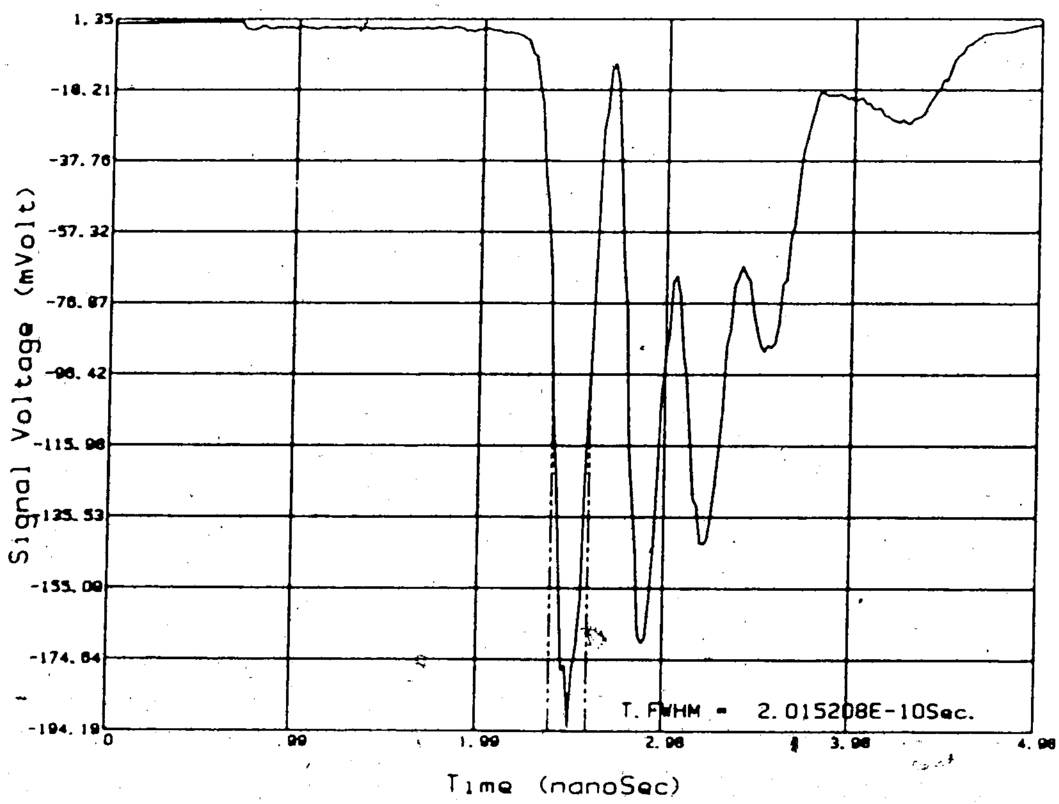


Fig.23 Laser output for a quiescent current of 94 mA.

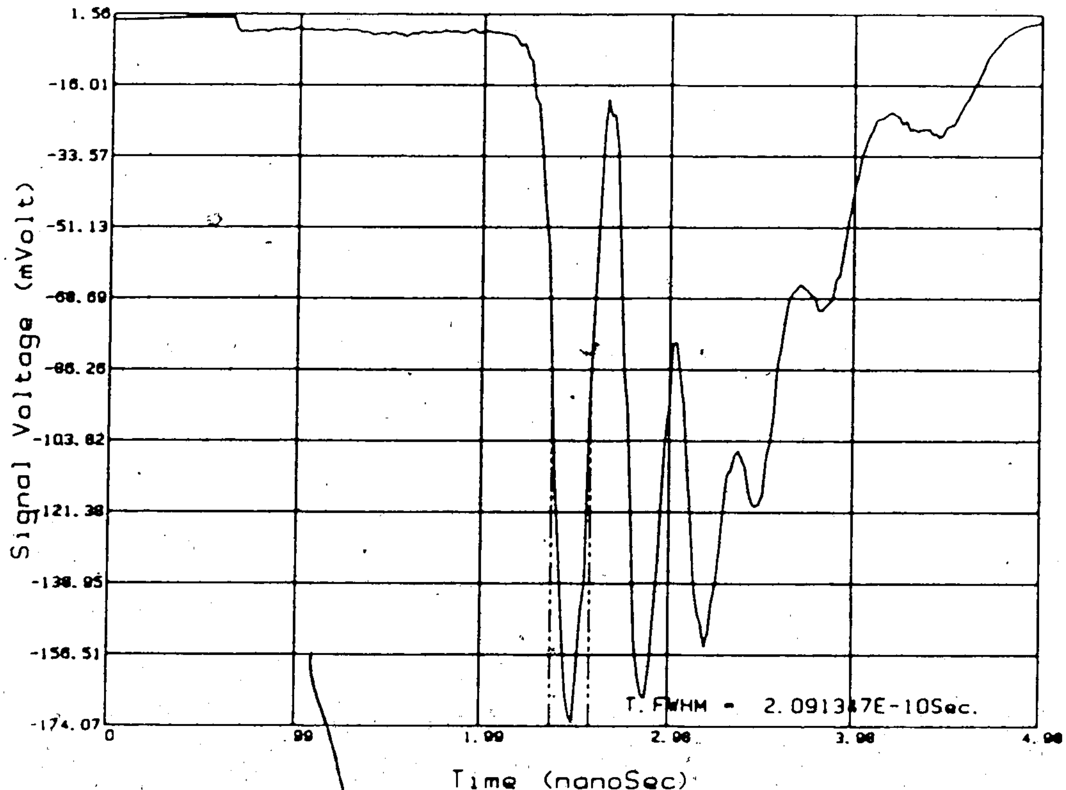


Fig.24 Laser output for a quiescent current of 95 mA.

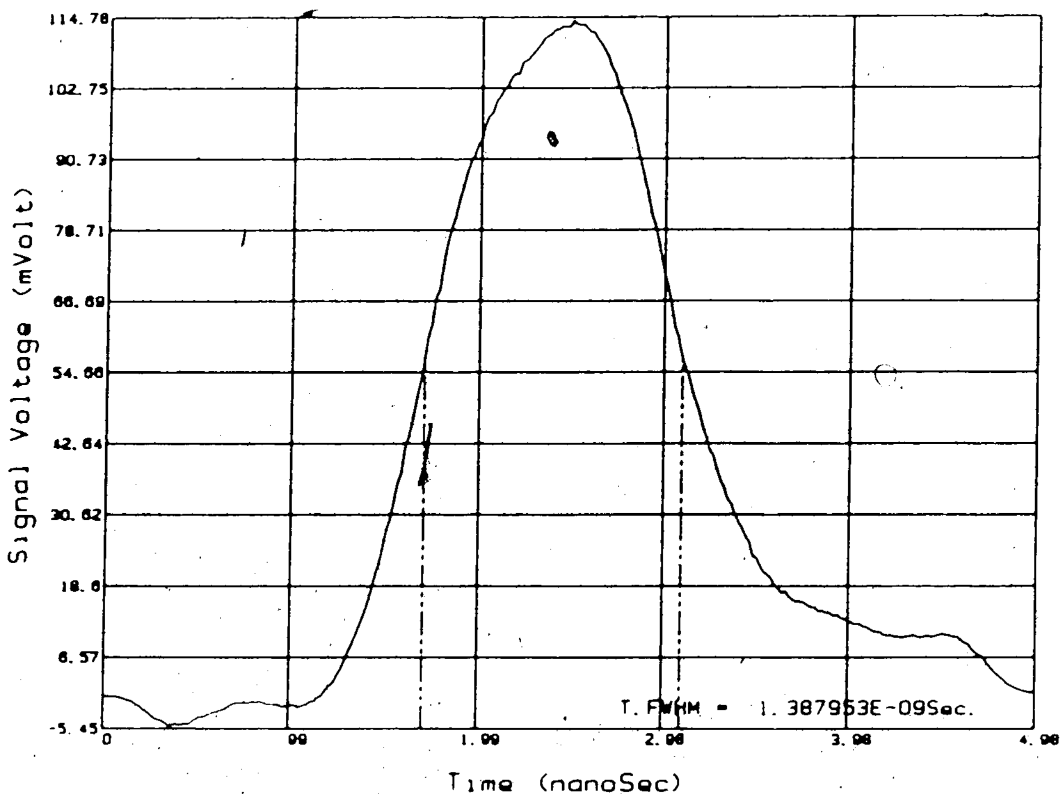


Fig.25 Input electrical pulse.

Care was taken throughout these experiments to ensure that the spikes observed were not caused by ringing in either the driving or the detection circuit. First, the current pulse shape (see Fig.25) was found to be unrelated to the observed optical pulses. Secondly, reflected signals due to impedance mismatch were ruled out as a causative effect by varying the cable length to the 50 ohm laser diode termination, and to the sampling scope. Variation of these lengths did not alter the optical pulse shapes. We finally considered the possibility that the variations in the separation between the optical pulses were due to some kind of saturation of the detector. This possibility was eliminated by reducing the light intensity with neutral density filters. The optical pulse shape remained unchanged. It is safe to conclude that the shape changes evident in Figs.19 to 24 are caused by the characteristics of the laser diode itself.

The risetime of the detector used (an APD available from Antel Corp. having the part no. AR-G15) for the above measurements was 90 psec, and its pulse broadening was quoted by the supplier as 200 psec (the Full Width Half Maximum (FWHM) value, represented by the letter τ). Thus, this detector could be reliably used for duration measurements on pulse of this order of magnitude or greater using the relation

$$\tau_{act} = \sqrt{(\tau_{obs})^2 - (200\ \text{psec})^2}\ . \tag{14}$$

The first pulse in Fig.19 is close to 200 psec in duration and a more accurate rise-time was found using a faster detector (Antel's Pin photodiode AR-G10) which has a risetime of 50 psec and a pulse response broadening of 75 psec. A single optical pulse of 127 psec duration at FWHM was recorded and is presented in Fig.26 followed by its computed frequency spectrum in Fig.27.

The optical fiber links that were to be tested in this project were expected to have a bandwidth-distance product of ~ 1.2 GHz km, or approximatly 150 MHz for a length of 8 km , or approximatly 260 MHz for a length of 4.5 km. Consequently, the optical impulse generator illustrated in Fig.14 was adequate for this project because the pulse generated contained a large spectral content out to about 1.24 GHz (see Fig.27).

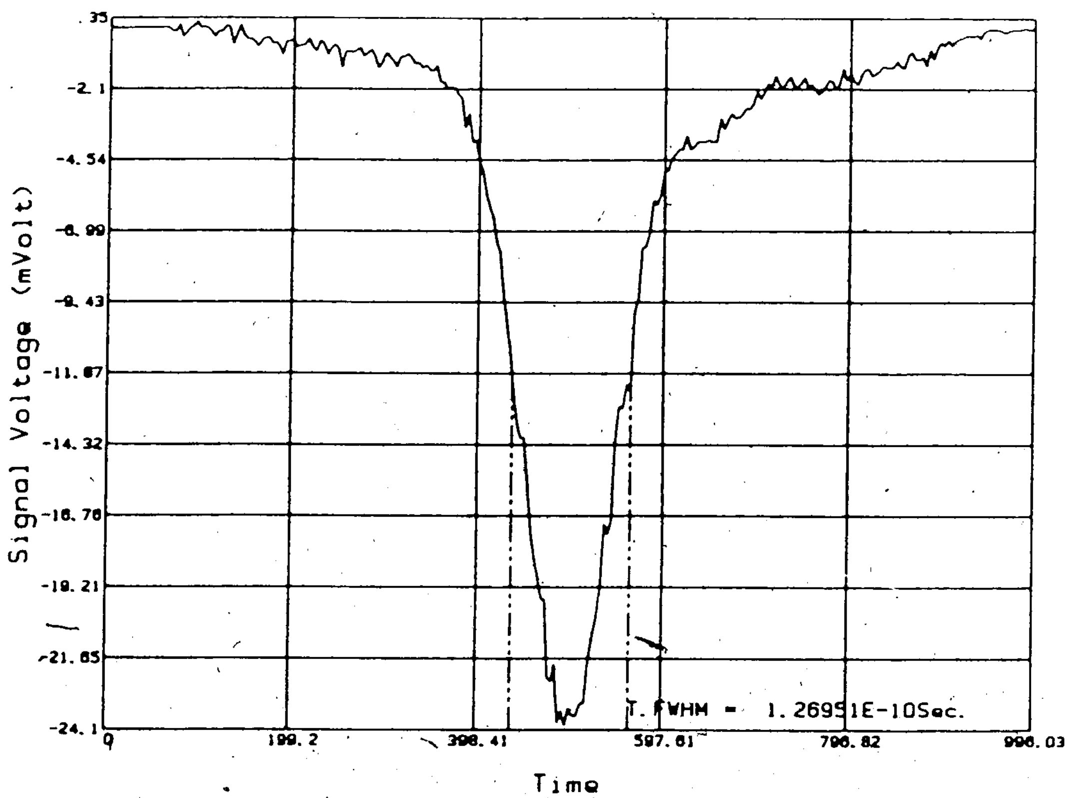


Fig.26 Single optical spike detected with a faster detector.

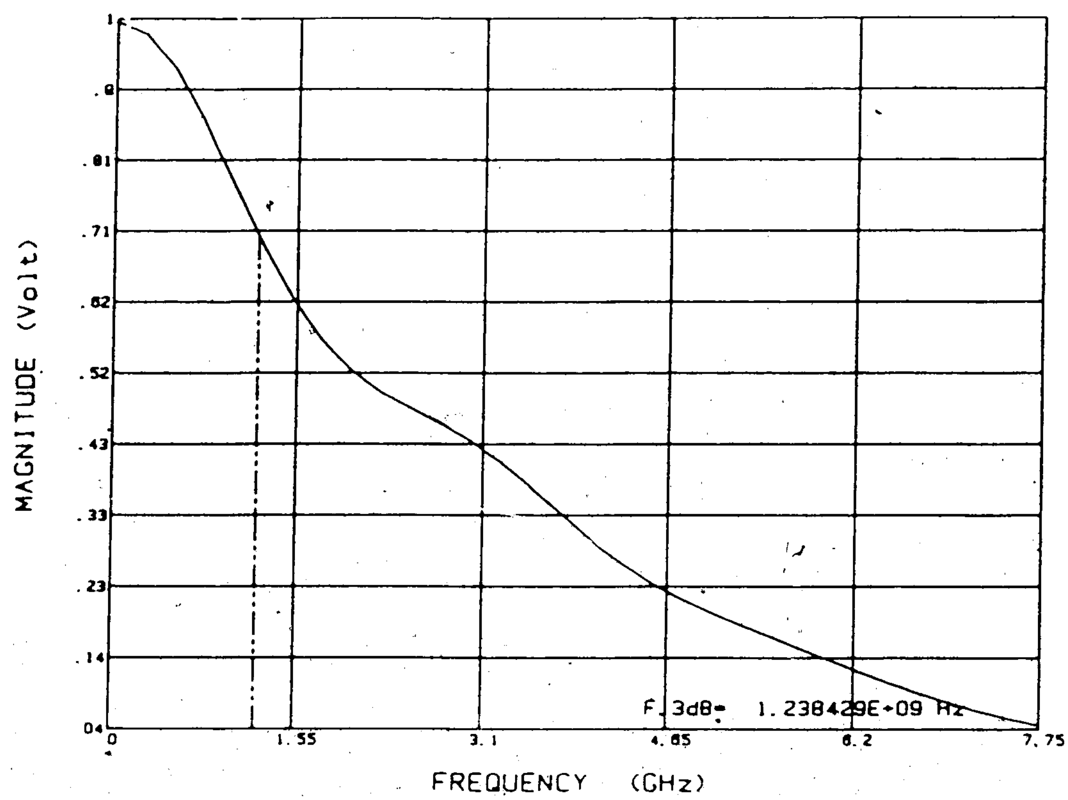


Fig.27 Computed frequency spectrum of a single optical spike detected with a faster detector.

In order to verify the long term stability of the optical pulse generator, it was turned on and left running for approximately 30 min. Then, a waveform was recorded, and its frequency spectrum was computed (see Figs. 28 and 29, respectively). The pulse generator was then left running for another hour when a second pulse was recorded. This hour is the approximate time it would take in the field to drive from one end of a fiber link to the other. This second waveform as well as its computed frequency content are shown in Figs. 30 and 31, respectively. The two recorded signals were identical in the time domain as well as in the frequency domain and the stability of the optical pulse generator was thus confirmed.

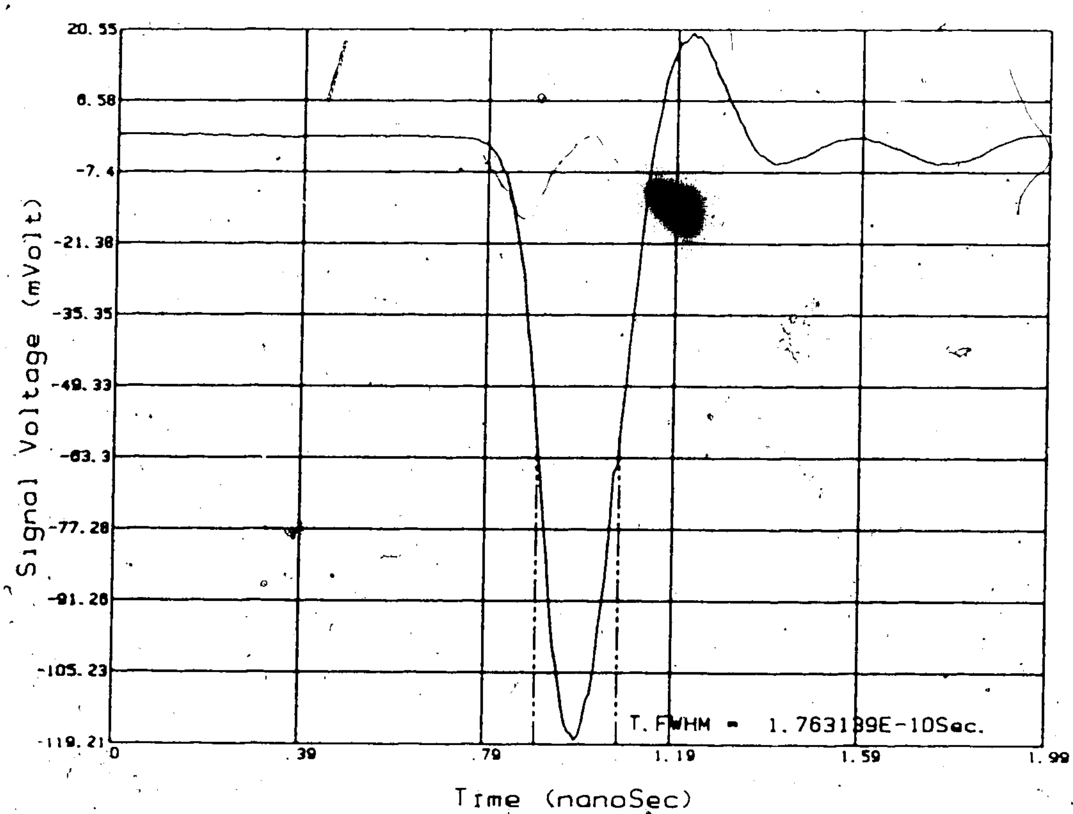


Fig.28 Time-domain output of the optical pulse generator at t=0 sec.

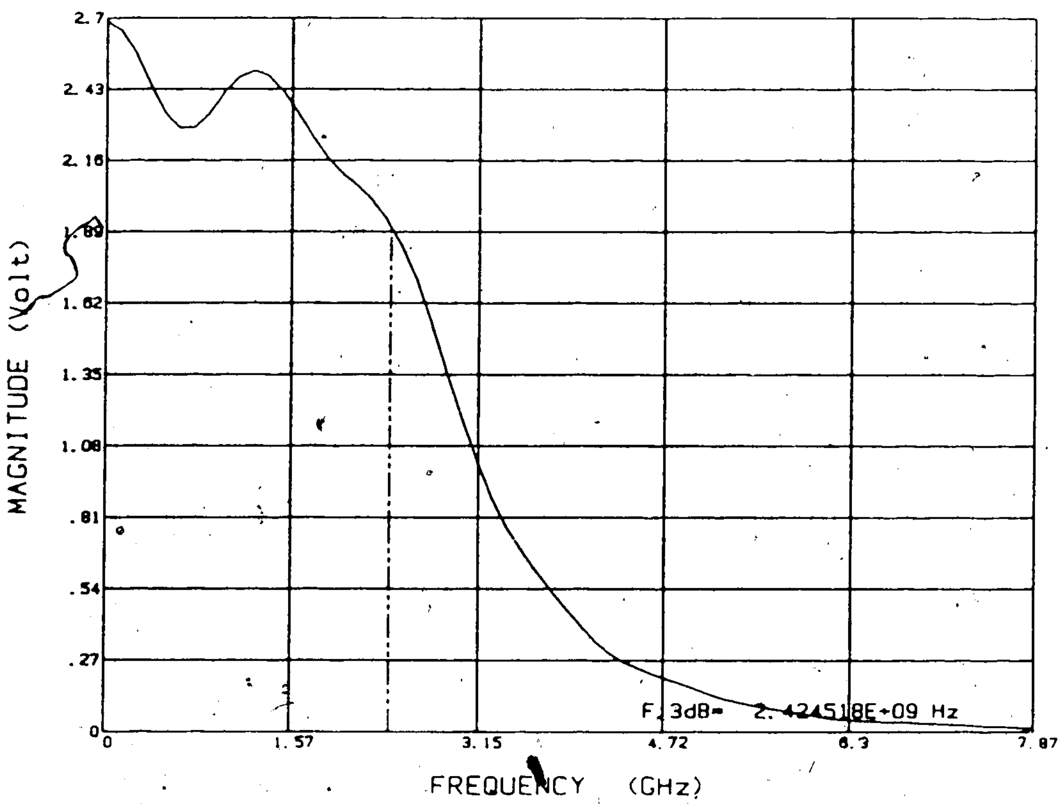


Fig.29 Frequency-domain output of the optical pulse generator at t=0 sec.

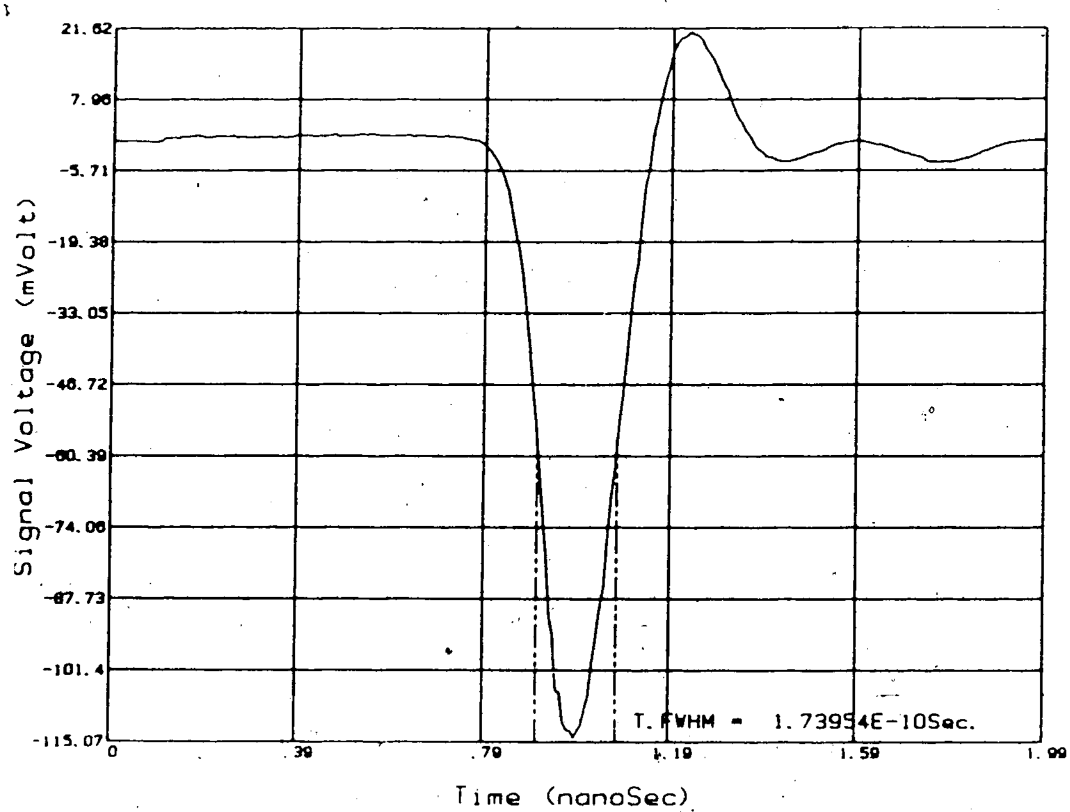


Fig.30 Time-domain output of the optical pulse generator at t=1 hour.

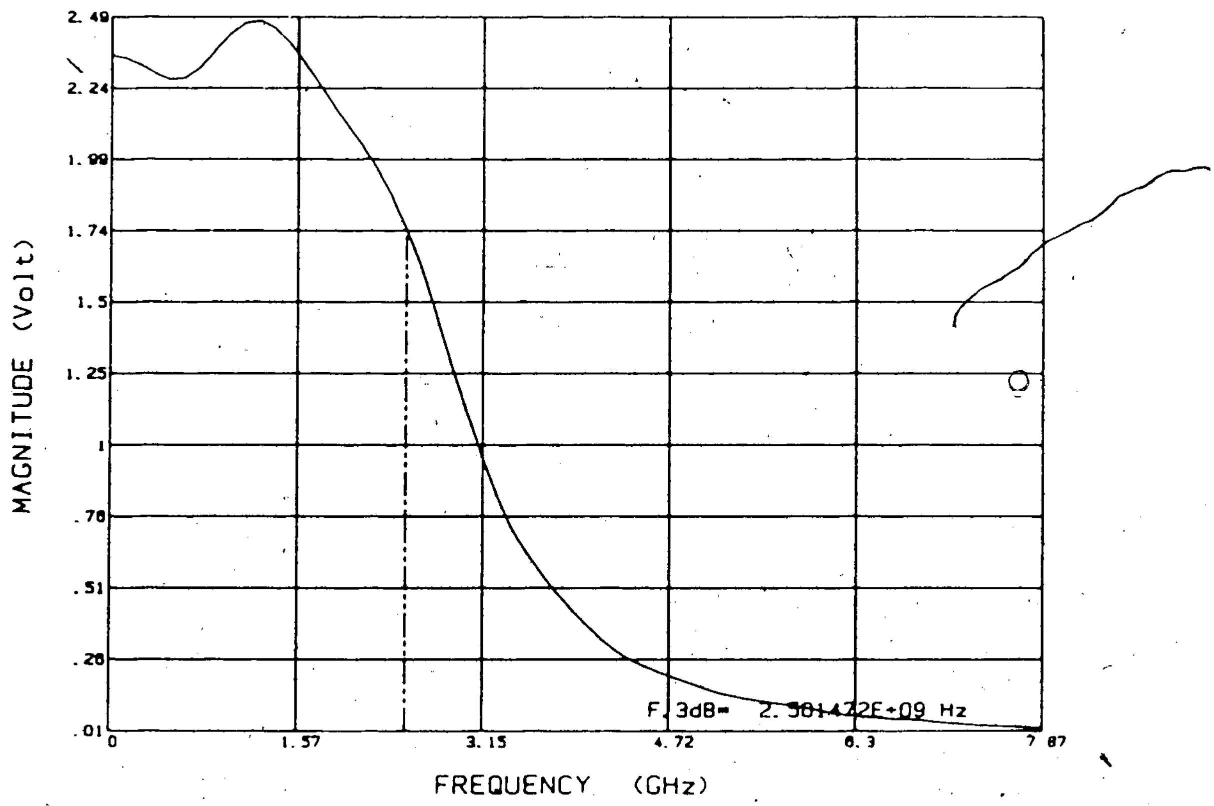


Fig.31 Frequency-domain output of the optical pulse generator at t=1 hour.

2.2.4 Double Pulse Generator

For practical reasons (triggering of the oscilloscope at the far end of the fiber) a double pulse generator was needed. This unit was designed around two independent avalanche transistor circuits enclosed in the same box. Each pulser had its own discharge line and trigger circuit and could be used as separate pulse generators. One of the discharge lines had a fixed length of 2 cm and was used to generate optimal narrow electrical pulses. The second was designed to produce a pulse of variable duration by plugging in the desired length of 50 ohm coax cable, as shown in Fig.32.

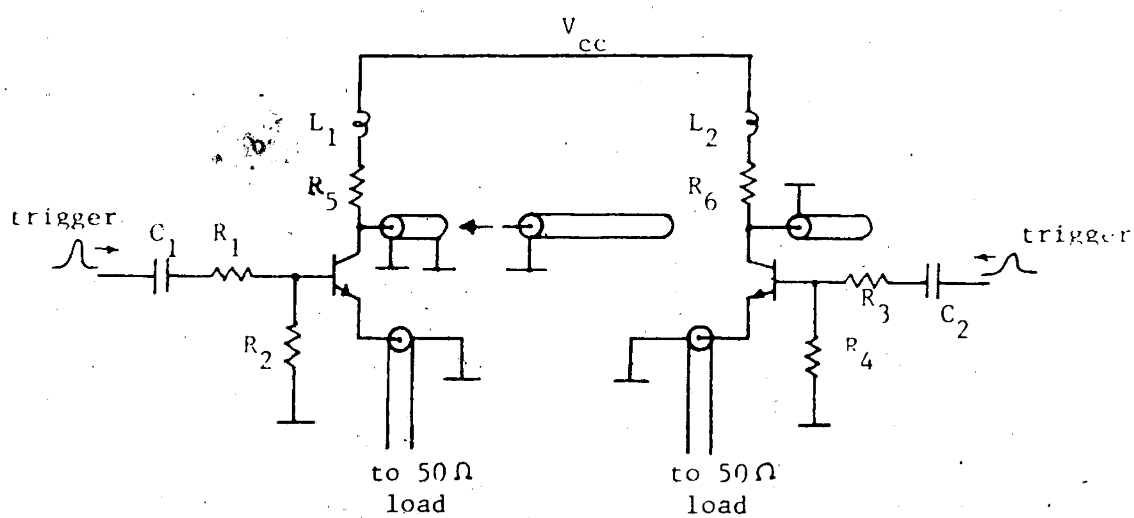


Fig.32 Double pulse generator circuit.

3. Optical Signal Detection

In this chapter, the experimental set-up used to detect the very weak optical signals at the output of the fiber link will be described. The different equipment used will be described in the following order: the V-groove connector, lenses, micropositioners, avalanche photodiode (APD), wideband amplifier, sampling scope and the matching network. The chapter will conclude with a description of the triggering scheme. Fig.33 shows the long-distance measurement set-up used in the field and will be referred to in the following discussion.

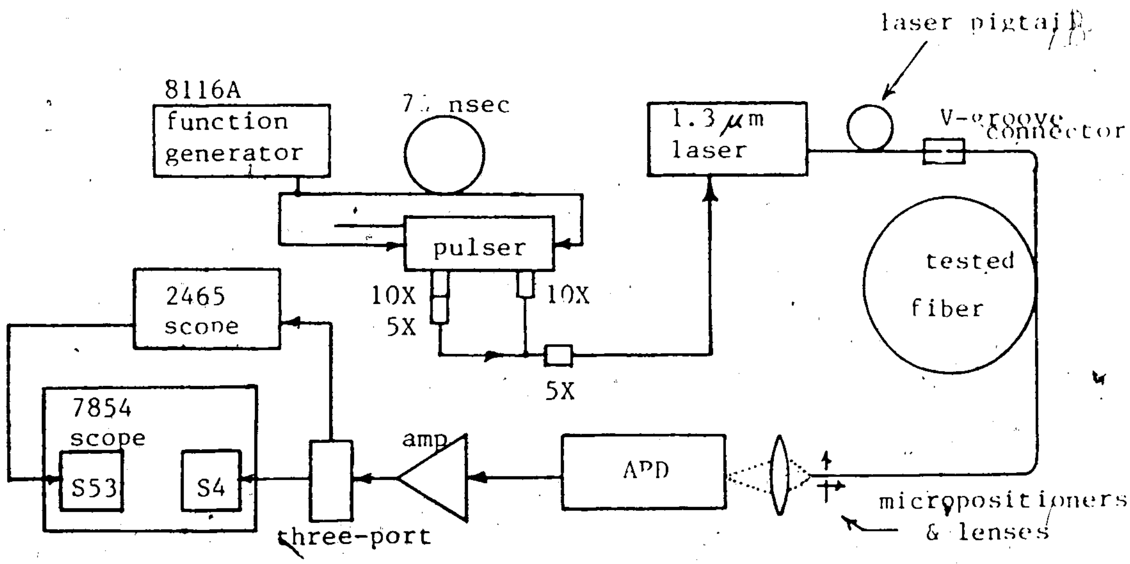


Fig.33 Long-distance experimental set-up.

3.1 Optical Equipment

The 1.3 μm laser diode module used for this project (a Northern Telecom QLS-3A module) was equipped with a 50 μm core multimode fiber pigtail. The optical power available from the output of this pigtail, at a quiescent current of 125 mA, is approximatly 0.75 milliWatt (mW). In order to couple this power efficiently into the fiber, a V-groove connector was used to butt-join the two ends. The space surrounding the two ends was filled with an index-matching liquid (glycerine) to maximize the coupling efficiency. Measurements using an optical time-domain reflectometer (OTDR) (a Tektronix OF150 fiber optics TDR) showed that coupling losses in the order of 1 to 1.5 dB were generally attained. Thus, in the worst case, the amount of power that would be coupled into the fiber (at a laser quiescent current of 125 mA, for example) is

$$-1.5 \text{ dB} = 10 \text{ LOG}(P_{in}/P_{pigtail}) = 10 \text{ LOG}(P_{in}/0.75\text{mW}).$$

Thus, $P_{in} = 0.53$ mW.

At the output of the fiber, lenses had to be used to recover the optical signal instead of a V-groove connector, because the APD was not equipped with a pigtail. Fig.34 shows a photo of the lenses, the APD, and the micropositioners. Fig.35 shows a schematic of the detection set-up.

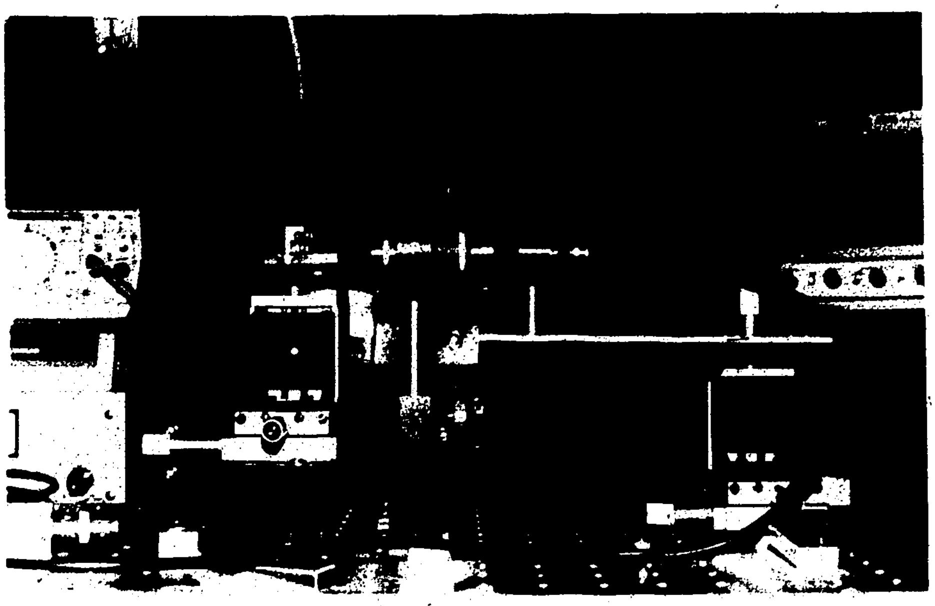


Fig.34 Lenses, APD and micropositionners.

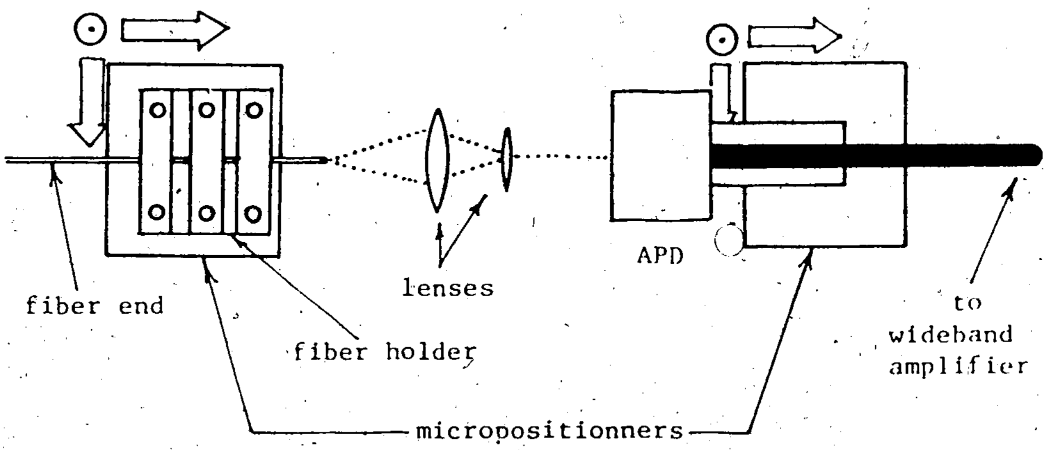


Fig.35 Detection set-up schematic.

The APD mounting had a flat window in front of the detector area, and the gain of the detector could be varied from 1 to 30 by adjusting the bias. The rise-time of this device (an Antel Corp. model AR-G15) was 90 psec and its pulse response τ was 200 psec. It is shown in chapter 4 that in the case of a purely Gaussian signal, the duration (FWHM) and the roll-off frequency, symbolized by τ and f_{3dB} respectively, can be related by the relation $(f_{3dB})(\tau)=0.3120$. Therefore, assuming that the generated optical pulses have a shape very close to a pure Gaussian, the f_{3dB} frequency of the detector is

$$f_{3dB} = (0.3120/200 \text{ psec}) = 1.6 \text{ GHz}.$$

The two micropositioners were used to precisely align the far end of the fiber on the active area of the detector in order to collect the maximum output optical power.

3.2 Wideband Amplifier

The wideband amplifier was designed to operate at frequencies much higher than the roll-off frequencies of the shortest fibers to be tested. These shortest fibers had an approximate length of 4500 m, and a roll-off frequency of ~130 MHz. Hence, an amplifier having a passband of at least twice this frequency was designed.

The amplifier was built around two integrated circuits cascaded, namely the NEC's MM766 and MM765 amplifiers, as shown in Fig.36. The frequency spectrum of the amplifier was measured using a wideband noise source (a Hewlett-Packard

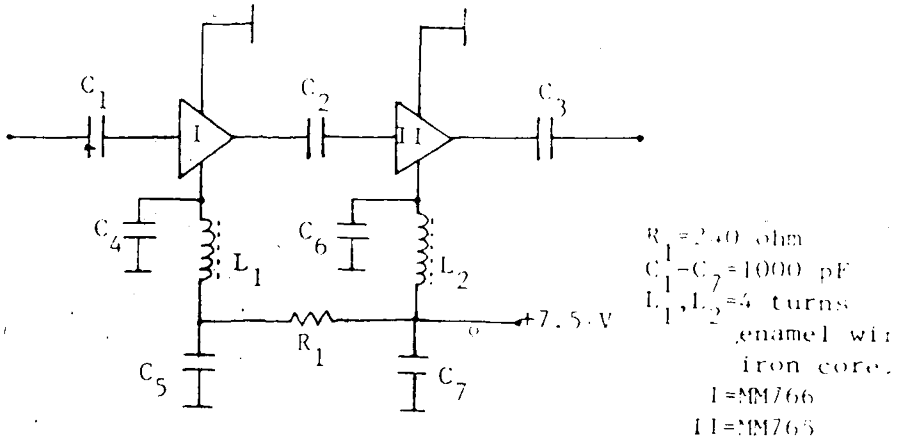


Fig.36 Block diagram of the wideband amplifier.

model no.HP346B) in tandem with a noise figure meter (a HP8970A). The gain of the amplifier was measured from 60 MHz to 1500 MHz, in increments of 20 MHz.

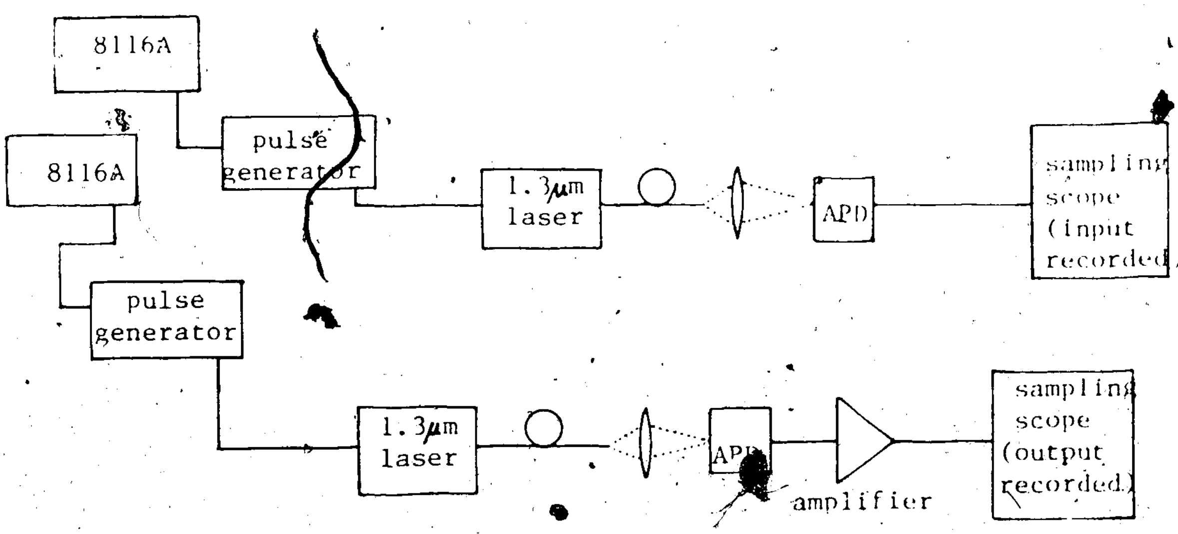


Fig.37 Amplifier gain measurement set-up

The results of these measurements are shown in Fig.38 and yield a roll-off frequency of ~650 MHz. The frequency spectrum of the amplifier was also measured using the time-domain technique that will be described in the next chapter, and the equipment set-up for this measurement is shown in Fig.37. The results, shown in Fig.39, yield also to the same roll-off frequency.

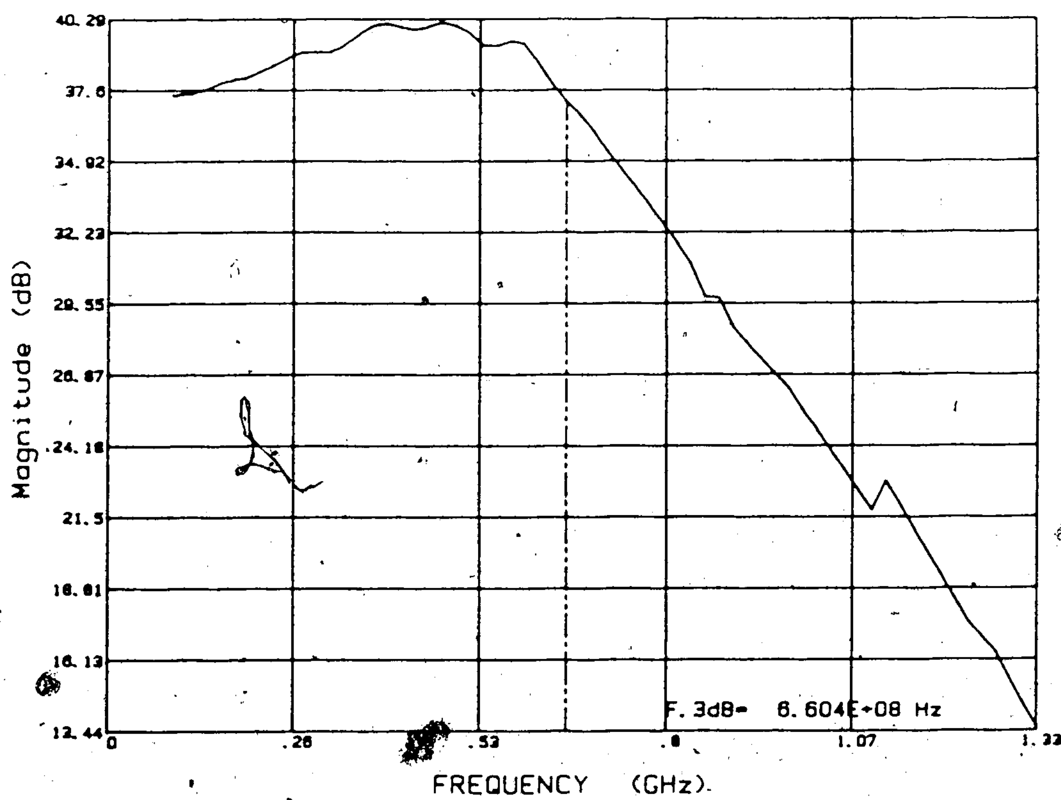


Fig.38 Wideband amplifier measured frequency spectrum.

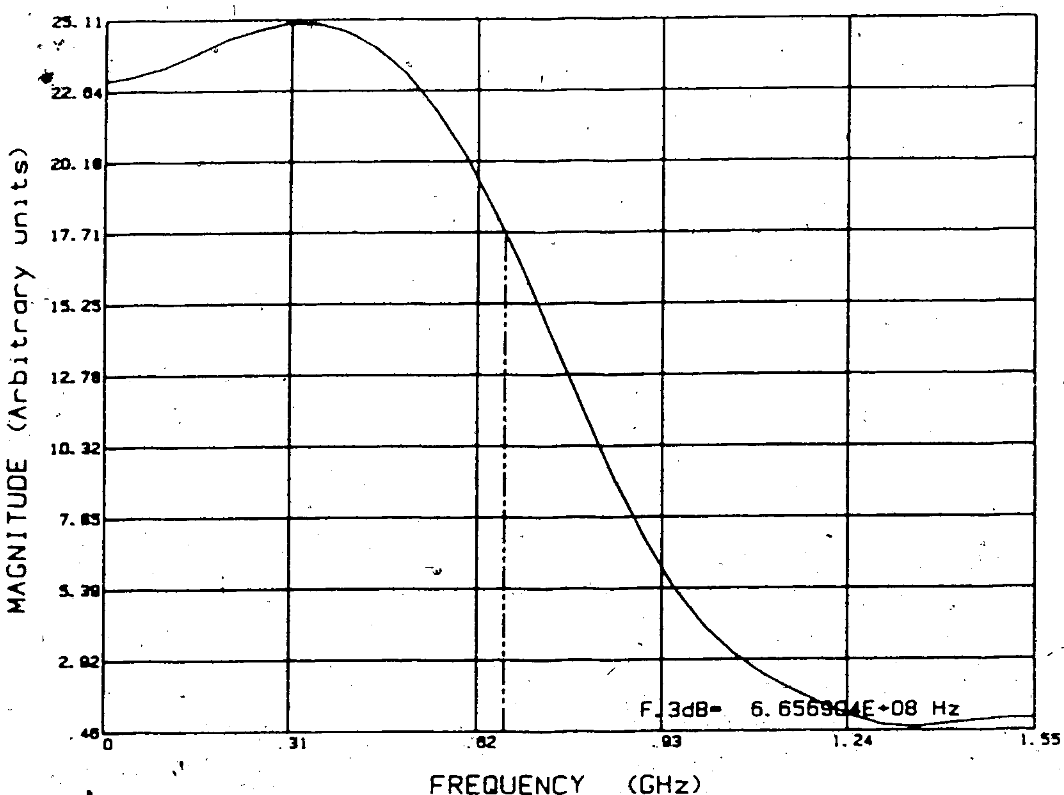


Fig.39 Wideband amplifier computed frequency spectrum.

3.3 Sampling Scope

In order to detect very narrow optical pulses, a Tektronix 7854 Sampling scope was used, together with the following plug-in units: a 7S12 TDR-Sampler, a S-4 Sampling Head and a S-53 Trigger Recognizer Head. With this configuration, the 7854 scope has a risetime of 25 psec. Fig.40 shows a photo of the sampling scope and the plug-in units.

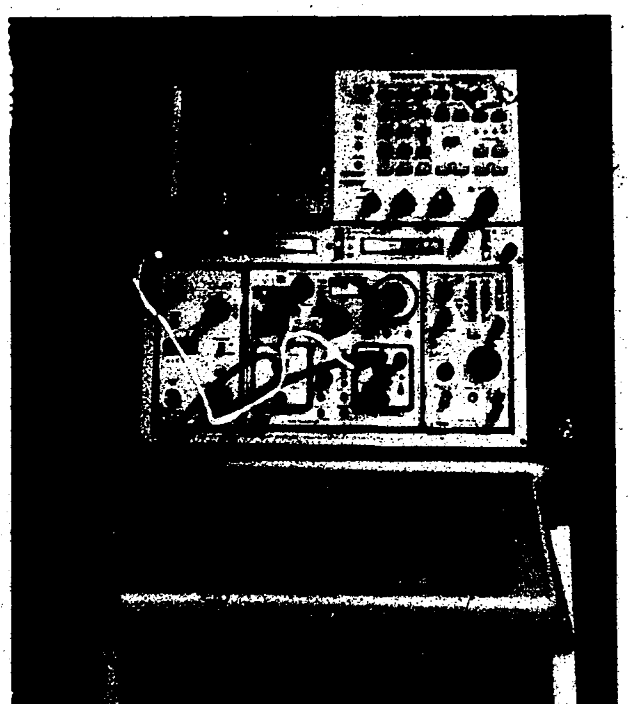


Fig. 40 Sampling scope and plug-in units.

The 7854 sampling scope was used to sample a displayed waveform and to store the digitized data in its internal memory. To remove the random noise which may be present on the real-time waveform, and to increase the accuracy of the digitized waveform, repetitive samples of the displayed signal were taken, and then averaged by the scope. The processor on the 7854 repeatedly re-acquires the real-time waveforms, digitizes and averages them then accumulates this data as a stored waveform. The result is that at the end of the averaging, the stored waveform is equal to the algebraic mean value of typically 200 individual waveforms.

The time increment between the sampled points of the averaged waveforms has to satisfy the Nyquist sampling theorem. This topic will be discussed in more detail in chapter 4, but for now it will only be mentioned that the number of samples per waveform was 256.

The S-4 sampling head has a risetime of 25 psec. Thus even for the fastest signals measured for this project, i.e. pulses having risetimes of ~100 psec, the effect of the sampling scope on the detected optical pulse risetime was negligible (see eqn.14).

The S-53 trigger recognizer head was used together with the S-4 to provide a trigger for the latter. A delay of ~70 nsec had to be provided between the arrival of the trigger pulse to the S-53 and the arrival of the signal to be displayed to the S-4. One way to realise this delay is to generate pulses at a rate of 1/70 nsec = 14 MHz, each pulse serving as a trigger for the following one. Another way is to generate double pulses separated by 70 nsec, at a much lower rate if desired. In order to limit the duty factor of the laser and thus to minimize its power dissipation, the latter method was chosen and a double optical pulse generator was designed.

The schematic of the double optical pulse generator is shown in Fig.41 and is based on the double electrical pulse generator discussed in chapter 2 and illustrated in Fig.32.

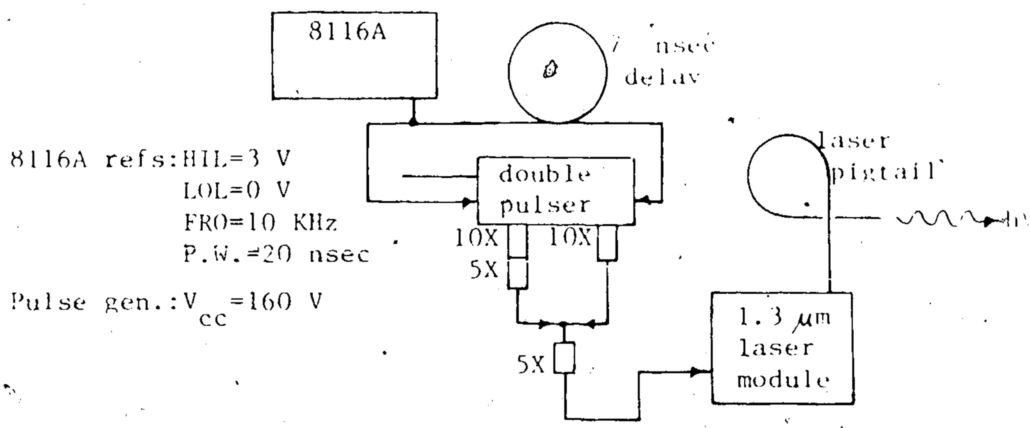


Fig.41 Double optical pulse generator schematic.

The trigger signal was leading the very short pulse by 70 nsec. This was done by using a section of 14 m of coax cable to delay the avalanche trigger, as shown in Fig.41.

3.4 Matching Network

A three-port 50 ohm matching network was designed to function as a signal splitter (or signal combiner). This unit was intended to be used for coupling the outputs of the double electrical pulse generator, and for splitting the output of the wideband amplifier in order to provide a trigger signal to the S-53. Single-sided PC board, copper tape and three 16.6 ohm chip resistors were used in the circuit shown in Fig.42.

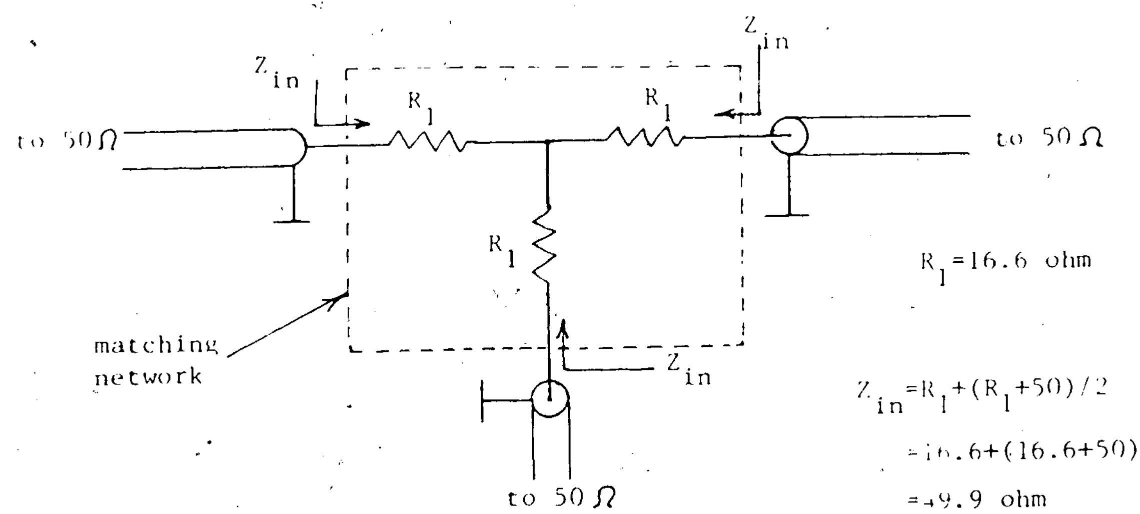


Fig.42 Three-port matching network circuit.

This matching network may also be called a 6 dB divider, since the current comming into one of the three branches is split equally between each of the other branches. Thus the power loss between the input branch and either of the two output branches is given by

$$10\ \log(P_{out}/P_{in})=10\ \log(\ R(I_{in}/2)^2/RI_{in}^2)\ =\ -6\ \mathrm{dB}.$$

In a situation where substantial power loss can be tolerated, an attenuator can be used to improve the matching of a line to an unusual load impedance, as shown in Fig.43. For example, a 10X attenuator terminated in a 25 ohm resistor has an input resistance of

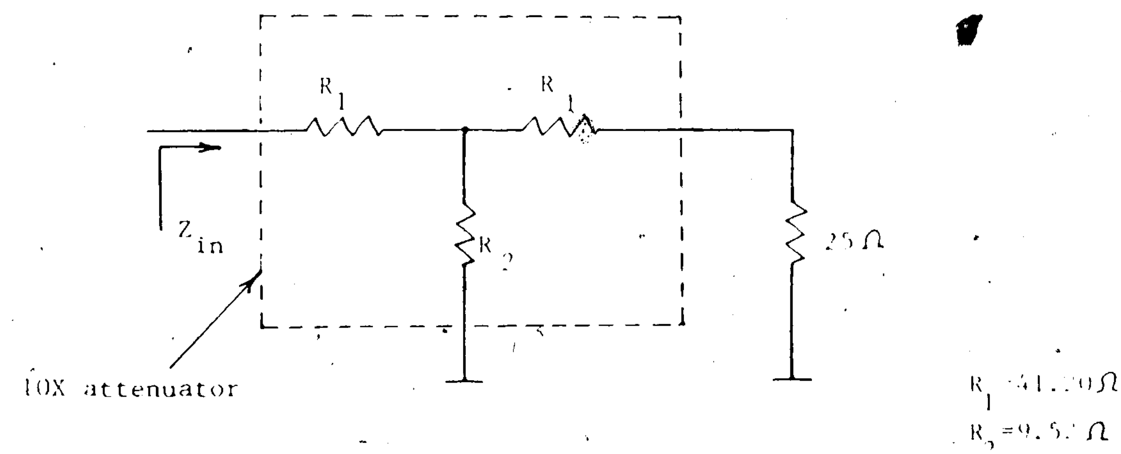


Fig.43 Attenuator used a matching network.

$$R_{in}=R_1+R_2(R_1+25)/(R_1+R_2+25)$$
$$=41.28+9.52(41.28+25)/(41.28+9.52+25)=49.6 \text{ ohm},$$

which is very close to a perfect match. Thus, attenuators were used in the double optical pulse generator circuit to improve the matching between the two avalanche generator outputs. The two attenuated electrical outputs were added together with a BNC adaptor, as shown in Fig.44.

3.5 Triggering Procedures

Calculations were required to determine the expected optical power levels at the far end of the fiber optic links to be tested. For a starting point, the input power was taken as the value of coupled optical power calculated at the beginning of this chapter, i.e. 0.53 mW. A typical length of 8 km of multimode graded-index fiber was assumed

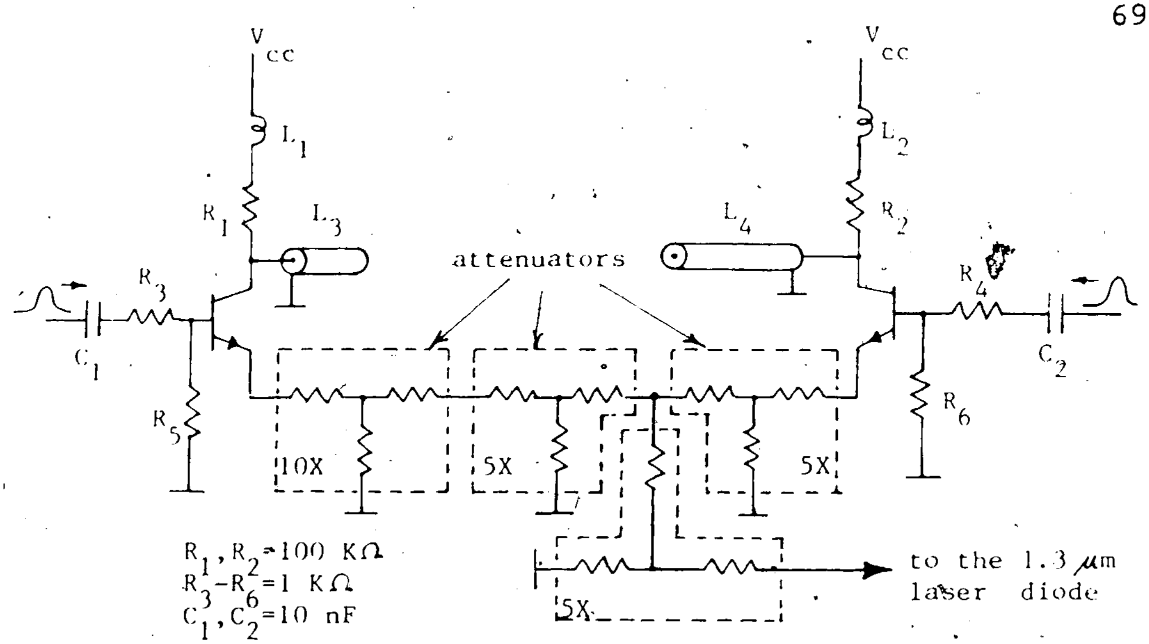


Fig. 44 Combining circuit for the double pulse generator.

to have an attenuation of ~ 3 dB/Km at 1.3 μm. The optical output power after propagation through this fiber would be approximately given by:

$$P_{out} = P_{in} - (\text{fiber loss})$$
$$= -2.76 \text{ dBm} - (8 \times 3.0 \text{ dB}) = -26.76 \text{ dBm}$$
$$= 2 \ \mu W.$$

The responsivity of the APD is approximatly 0.16 A/W at 1.3 μm. So, even if half of the power would be recovered at the output of the fiber (which is unlikely because of the losses through the glass-to-air interface, the lenses and the detector window), and if the APD would be biased to provide a gain of 15, the output voltage at the detector would only be

$$(2 \times 10^{-6})(0.16)(50)(15) = 0.24 \text{ mV}.$$

If a wideband amplifier having a gain of 40 dB would be used at the output of the detector, the voltage available at the output of the amplifier would be ~24 mV. If this signal was reduced even further after passing through a matching network (6 dB), the signal received by the sampling head (S-4) would be very weak. Preliminary tests in the lab showed that signals having an amplitude of 10 mV or less were very hard to display on the sampling scope. For this reason, a different triggering procedure was devised.

A 300 MHz real-time oscilloscope was available in the lab. This unit (a Tektronix 2465) had excellent trigger circuitry that allowed the display of very weak signals (~ 2 mV), and had also the capability of generating an output signal proportional to one of its input channels. So, to ease the problem of detection of very weak optical signals, the two oscilloscopes mentioned above were used in tandem as shown in Fig.33. The output of the 2465 oscilloscope was providing the trigger for the S-53, and the matching network discussed above was used to provide inputs to the two oscilloscopes.

4. SIGNAL PROCESSING

Successful completion of this project necessitated the development of a data acquisition system consisting of the Tektronix 7854 sampling oscilloscope, an IBM XT personal computer and an HP 7090A plotter. Digitized waveforms were transmitted from the sampling scope to the IBM XT which was then used to compute the spectral content and to control the graphical plotter display.

This chapter will cover the signal processing aspects of this project. First, a description will be given of the program that transfers the data from the sampling scope to the computer. Then, the technique of converting time-domain response into frequency-domain characteristics (the Fast Fourier Transform (FFT) operation) will be analysed. The discussion will cover the windowing problem and the control of the leakage and aliasing errors. Finally, the other programs used for the calculation and the display of the frequency spectrum of the fiber will be discussed.

4.1 Data Transfer Program

A program was written to transfer the digitized waveforms from the sampling scope to the computer (an IBM model XT). The hardware was based on a PC-Mate IEEE-488 interface board, and the program software was written in Advanced Basic and included the IEEE488 Software Package (from TECMAR Inc.). The PC-Mate 488 board allowed the computer to be the system controller on the General Purpose

Interface Bus (GPIB). Access to the GPIB permitted transfer of data to the IBM RAMs, to the disks, or to any peripheral devices controlled by the IBM such as the 7854 sampling scope, or the Hewlett-Packard HP7090A Plotter.

Other than the transfer of data from the sampling scope to the computer, the program that was written allowed: (i) data transfer from the computer *to* the sampling scope, (ii) storage (or recovery) of waveforms to (or from) the 10 Mbits hard disk, (iii) display of waveforms stored in the RAMs. The final version of this transfer program, called *scope*, is reproduced in Appendix 2.

In order to speed up the execution of the data transfer, a Basic Compiler Program was used. With this substantial modification, a typical transfer of 256 samples from the scope to the computer took approximately 15 sec.

4.2 Fast Fourier Transform

The techniques of spectral analysis employing Fourier transforms and Fourier series have long represented an important area of application in continuous-time signal processing. The development in 1965 of the Cooley-Tukey algorithm [30] for the rapid computation of the approximate spectrum paved the way for new and varied applications of spectral analysis. With this approach, the spectrum of a signal containing many thousands of sampled points can be obtained in a matter of milliseconds on a very fast computer.

The Fast Fourier Transform (FFT) technique is well documented in the literature (see [31]-[34]) and won't be analysed in detail here. The Cooley-Tukey algorithm will be briefly described in the next section along with the precautions that were taken to minimize the aliasing and leakage errors.

4.2.1 Cooley-Tukey FFT Algorithm

The FFT algorithm that was used is the standard version that Cooley and Tukey introduced in 1965. It exploits the various symmetries inherent in the definition of the Fourier transform, in order to speed-up the calculations. Due to the nature of the FFT algorithm, either the data fed to the program, or obtained from it, must be interlaced in order to obtain the proper ordering of the output data. In the FFT algorithm shown in Fig.45, the input data is interlaced prior to the execution of the FFT.

```
1500 N=2^M:NV2=N/2:NM1=N-1:J=1
1510 '
1520 FOR I=1 TO NM1
1530 IF I>=J THEN GOTO 1570
1540 T.REAL=X.REAL(J):T.IMAG=X.IMAG(J)
1550 X.REAL(J)=X.REAL(I):X.IMAG(J)=X.IMAG(I)
1560 X.REAL(I)=T.REAL:X.IMAG(I)=T.IMAG
1570 K=NV2
1580 IF K>=J THEN GOTO 1620
1590 J=J-K
1600 K=K/2
1610 GOTO 1580
1620 J=J+K
1630 NEXT I
1640 '
1650 PI=3.141592653589793#
1660 CLS:M=INT(M+.5)
1670 FOR L=1 TO M
1690 LE=2^L:LE1=LE/2
1700 U.REAL=1:U.IMAG=9.999999D-21
1710 IF ISIGN=1 THEN GOTO 1750
1720 W.REAL=COS(PI/LE1):W.IMAG=SIN(PI/LE1)
1740 GOTO 1770
1750 W.REAL=COS(PI/LE1):W.IMAG=-SIN(PI/LE1)
1760 '
1770 FOR J=1 TO LE1
1780 FOR I=J TO N STEP LE
1790 IP=I+LE1
1880 T.REAL=X.REAL(IP)*U.REAL - X.IMAG(IP)*U.IMAG
1890 T.IMAG=X.REAL(IP)*U.IMAG + X.IMAG(IP)*U.REAL
1900 X.REAL(IP)=X.REAL(I)-T.REAL
1910 X.IMAG(IP)=X.IMAG(I)-T.IMAG
1920 X.REAL(I)=X.REAL(I)+T.REAL
1930 X.IMAG(I)=X.IMAG(I)+T.IMAG
1940 NEXT I
2000 T.REAL=U.REAL*W.REAL - U.IMAG*W.IMAG
2010 T.IMAG=U.REAL*W.IMAG + U.IMAG*W.REAL
2014 U.REAL=T.REAL:U.IMAG=T.IMAG
2020 NEXT J
2030 NEXT L
```

Fig.45 Fast Fourier transform (FFT) BASIC program.

Turning a Fourier transform into the sampled data version FFT nearly always introduces error. The error known as leakage and aliasing almost invariably occurs when continuous time-domain waveforms are subjected to finite-time windowing and sampling (both of these operations are fundamental to the FFT). Let us first recall the definition of the Fourier transform

$$X(f)=\int_{-\infty}^{+\infty}x(t)e^{-j2\pi ft}dt, \tag{15}$$

where x(t) is a continuous time-domain function and X(f) is the corresponding frequency-domain function for which the integral is to be evaluated. To transform x(t) digitally, the Fourier transform must be restated as the discrete Fourier transform (DFT)

$$X_d(k\Delta f)=\Delta t\cdot\sum_{n=0}^{N-1}x(n\cdot\Delta t)e^{-j2\pi k\Delta f n\Delta t} \tag{16}$$

or, letting $\Delta f=1/N\Delta T$,

$$X_d(k\Delta f)=\Delta t\cdot\sum_{n=0}^{N-1}x(n\Delta t)e^{-j2\pi kn/N} \tag{17}$$

where k and n=0,1,2,...,N-1, Δt is the time-domain sampling interval, and N is the number of samples taken over the interval of time (N-1)·Δt.

Now, the FFT program shown in Fig.45 is nothing more than a computer algorithm that efficiently evaluates the DFT, so its mathematical properties are completely analoguous to the DFT's[35]. Similarly, the errors associated with the FFT derive from the DFT. Leakage error arises from the fact that the waveform is studied over only a short period (or window) of time, aliasing error arises if the waveform is sampled at too slow a rate and picket-fence error arises due to the discrete nature of the DFT. These errors are discussed in turn in the following three sections.

4.2.2 Leakage Errors (Decreased by Windowing)

In the integral transform, time extends from $-\infty$ to $+\infty$. In the discrete transform, only the time interval covering the N discrete samples is considered. In Fig.46, a continuous function of time, a sine wave, is assumed to exist over the time from $-\infty$ to $+\infty$. When this sine wave is transformed into the frequency domain by an FFT algorithm, a data window (Fig.46(b)) must be defined, and a segment of the waveform is viewed through this window. Thus, all knowledge of the waveform's behavior before and after the window is lost.

In effect, the window of Fig.46(b) is a unity amplitude pulse. The sine wave is "viewed" through the window when the two are multiplied together. The result of this time-domain multiplication of Figs.46(a) and (b) is shown in (c). Obviously, the act of windowing in the time domain must also effect the signal in the frequency domain. Figs.46(d), (e), and (f) are the magnitudes of Fourier transforms of (a), (b), and (c), respectively. Since multiplication in the time domain corresponds to convolution in the frequency domain, Fig.46(f) is produced by convolving the magnitude of plots (d) and (e).

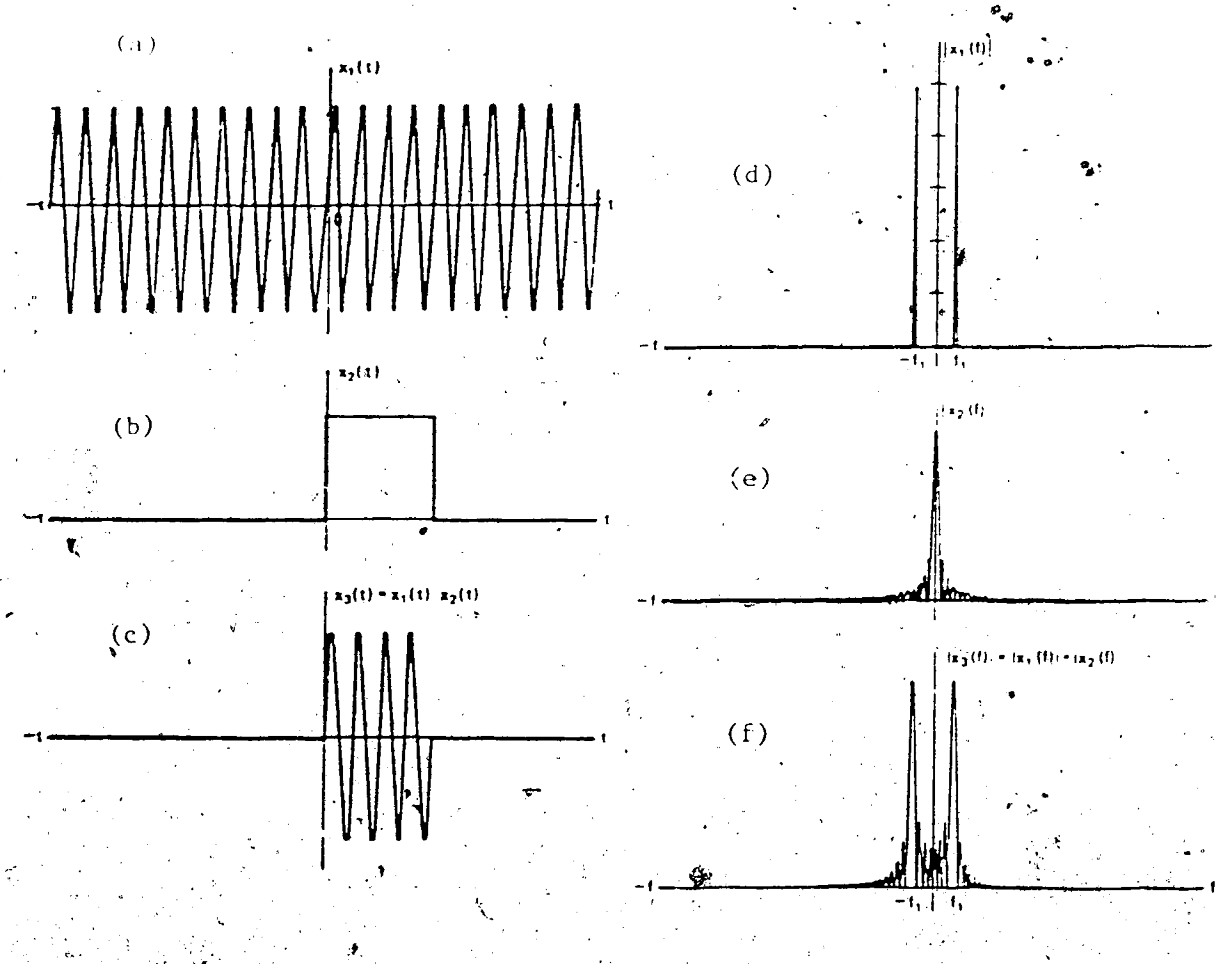


Fig.46 Effects of windowing on a sinusoid.

Fig.46 clearly shows the effect of windowing in the frequency domain. The original concentration of energy in the two impulses of Fig.46(d) has been smeared or "leaked" into the major lobes and sidelobes that appear in Fig.46(f). The same amount of energy is present in both cases, but it has been redistributed in such a way as to decrease peak magnitude.

Leakage generally occurs and just has to be lived with, although it can be diminished if the skirts of the window fall less drastically than do the ones of a rectangular window. Now, the windowing function is, in essence, a time-domain pulse of fixed energy, and any change in that pulse's shape must be reflected in a redistribution of its energy in the frequency domain. It follows that if the shape of a windowing function is changed to reduce side-lobe size, the energy normally associated with those side-lobes must go elsewhere. In general, the energy is forced into and widens the major lobe.

Besides the rectangular window, there are many other functions used for preconditioning acquired signals. Also called weighting functions, or convolution Kernels, only a few of them will be mentioned: the triangular, the cosine, the Hamming and the cosine squared, cubed and fourth order windows [35]. However in general, the spectra of these window functions consist of a main lobe and various side-lobes (see Fig.46). It is desireable that the window function satisfy the two criteria: (a) the main lobe should be as narrow as possible, and (b) the maximum side-lobe level should be as small as possible relative to the main lobe.

Both of these criteria cannot be simultaneously optimized, so that most usable window functions represent a compromise between the two factors. For this project, one of the main objectives was to measure the cut-off frequency of different fibers. Thus, we were concerned mostly with the selectivity (the ability to differentiate between unequal amplitude spectral components at adjacent frequencies) of the window rather than its resolution (the ability to distinguish between adjacent frequency components of equal amplitude). Because high selectivity was needed, a window function having very small side-lobes was used. The Hamming window was initially considered; it is defined by

$$w(t)=\begin{cases} 0.54+0.46\cos(2\pi t/\tau) & \text{for } |t|\le\tau/2 \\ 0 & \text{elsewhere,} \end{cases} \qquad (18)$$

where τ is defined as the width of the time-domain sampled waveform, i.e., $\tau=(N-1)T$.

The Hamming window function has a width of τ and is an even function of t. Fig.47(a)[34] shows half of the function, corresponding to the range $0\le t/\tau\le 0.5$. The function w(t) represents the continuous-time form of the window function as given by equation 18. The discrete-time window function is determined by replacing t with nT and evaluating the function for integer values of n. If W(f) represents the

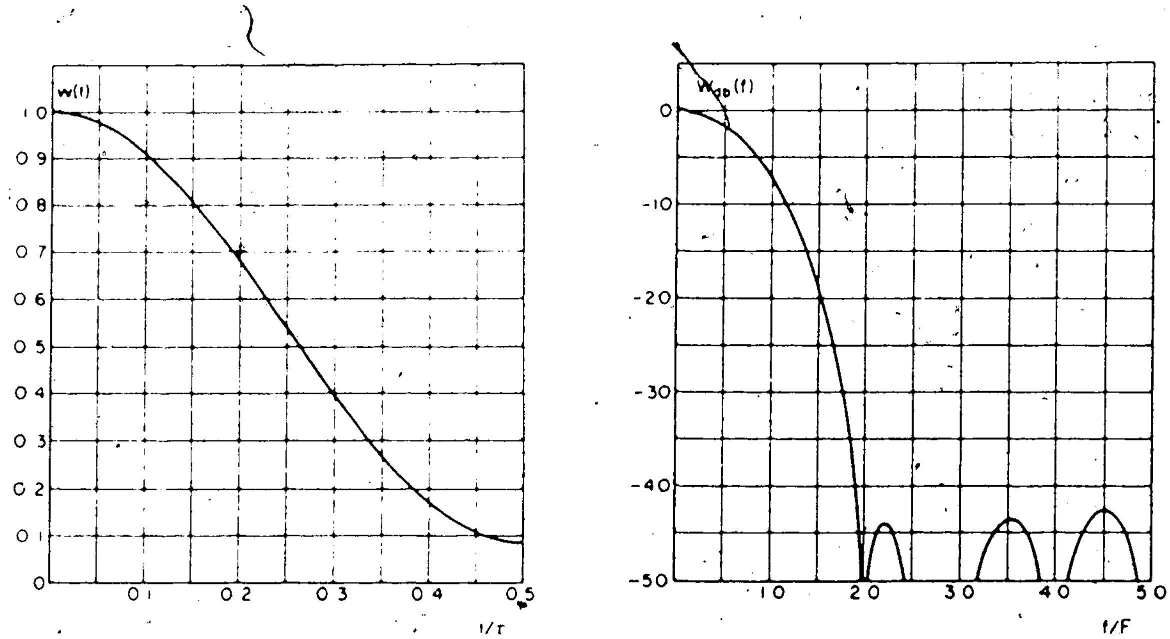


Fig.47 Hamming window function (a) in the time-domain and (b) in the frequency domain.

Fourier transform of w(t), and if W(0) represents the d.c. value, the amplitude response for the Hamming window function in decibels is defined by

W(f) in dB = 20 log W(f)/W(0). (19)

The amplitude response curves given in Fig.47(a) and (b) are a function of f/F, where F is defined as F=1/τ.

Fig.48 shows typical data obtained from a field measurement. It consists of 256 digitized samples of a waveform. The difference in level between the start and the end of this waveform was caused by the wideband amplifier and had to be removed to prevent a severe leakage error in the frequency domain results. Fig.49 shows the same data after a Hamming window had been applied to it. Clearly, the windowing in this case has removed this difference in levels, but this removal was at the expense of causing undesirable distortion in the useful part of the signal, e.g. in the center. In fact, as a result of the windowing operation, the duration τ of the waveform was reduced artificially by

$$(3.19 \text{ nsec} - 3.03 \text{ nsec})/3.19 \text{ nsec} = 5\%$$

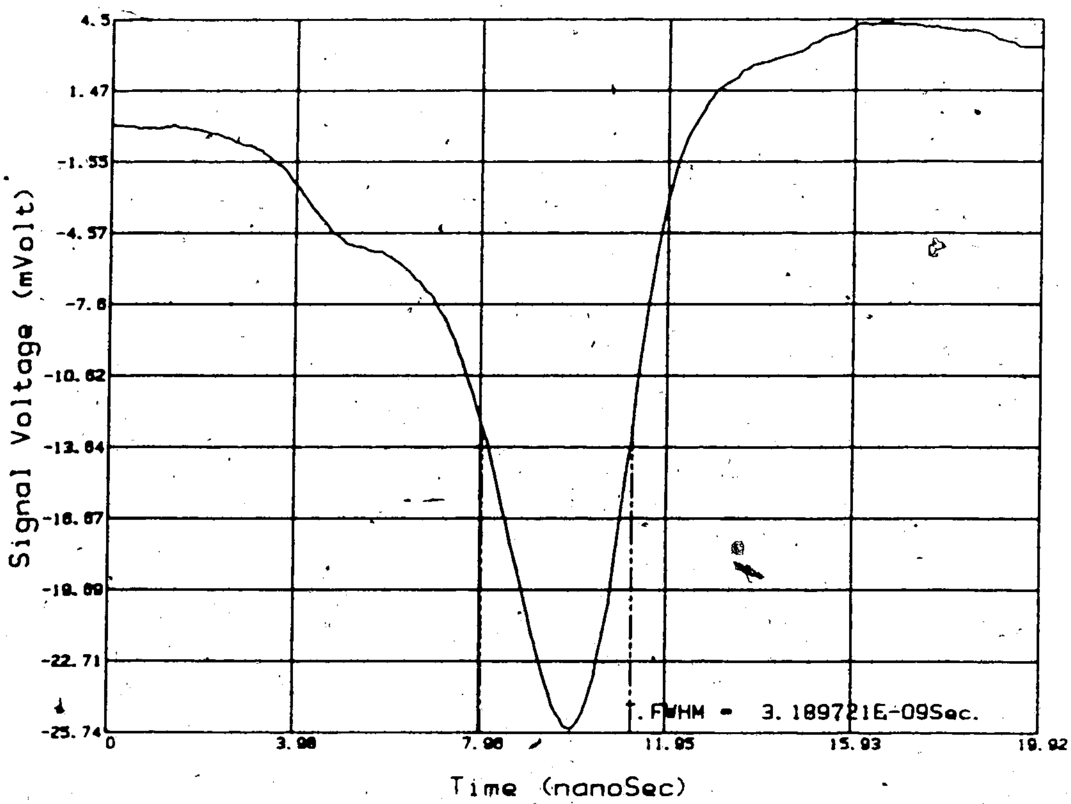


Fig.48 Typical data obtained from a field measurement.

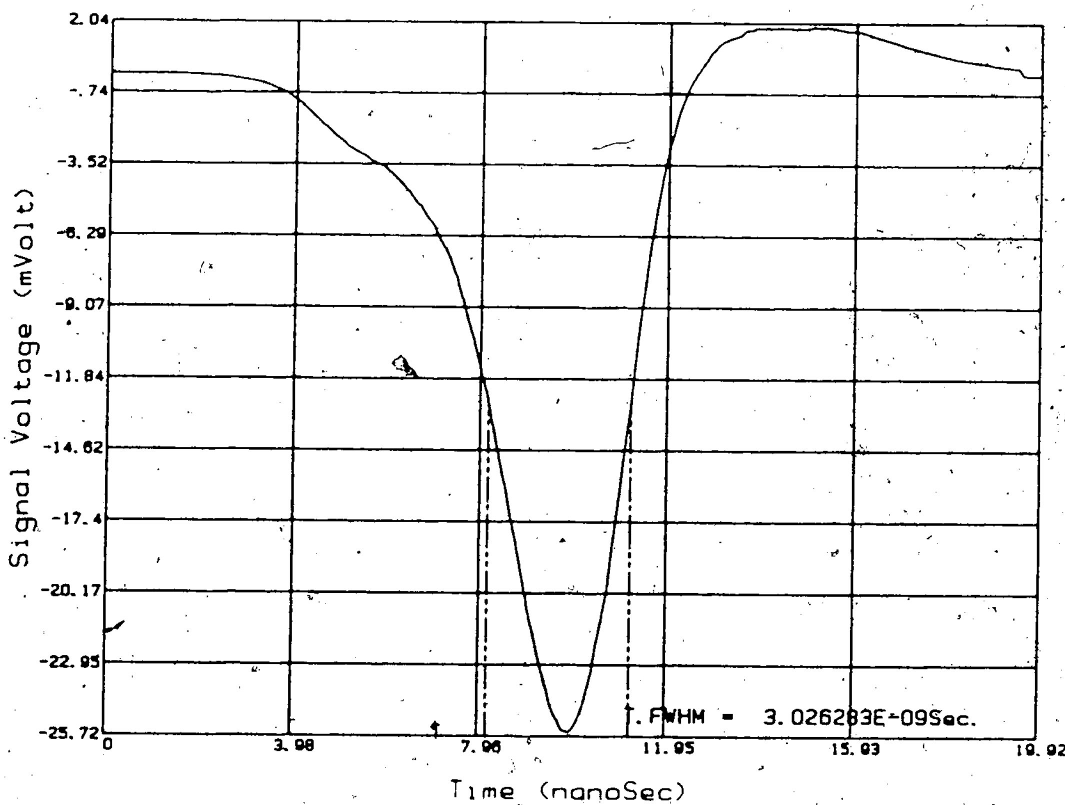


Fig.49 Typical data after a Hamming window has been applied on it.

Fig.50 shows the computed frequency spectrum for this data. From Fig.47(b), the first zero-crossing of the Hamming window function frequency spectrum has a value of 2x1/20nsec=100 MHz. This means that the actual frequency spectrum of the data has been convolved with a very wide function (~ 100 MHz), and that the accuracy of any fiber 3dB cutoff frequency computation would be very poor.

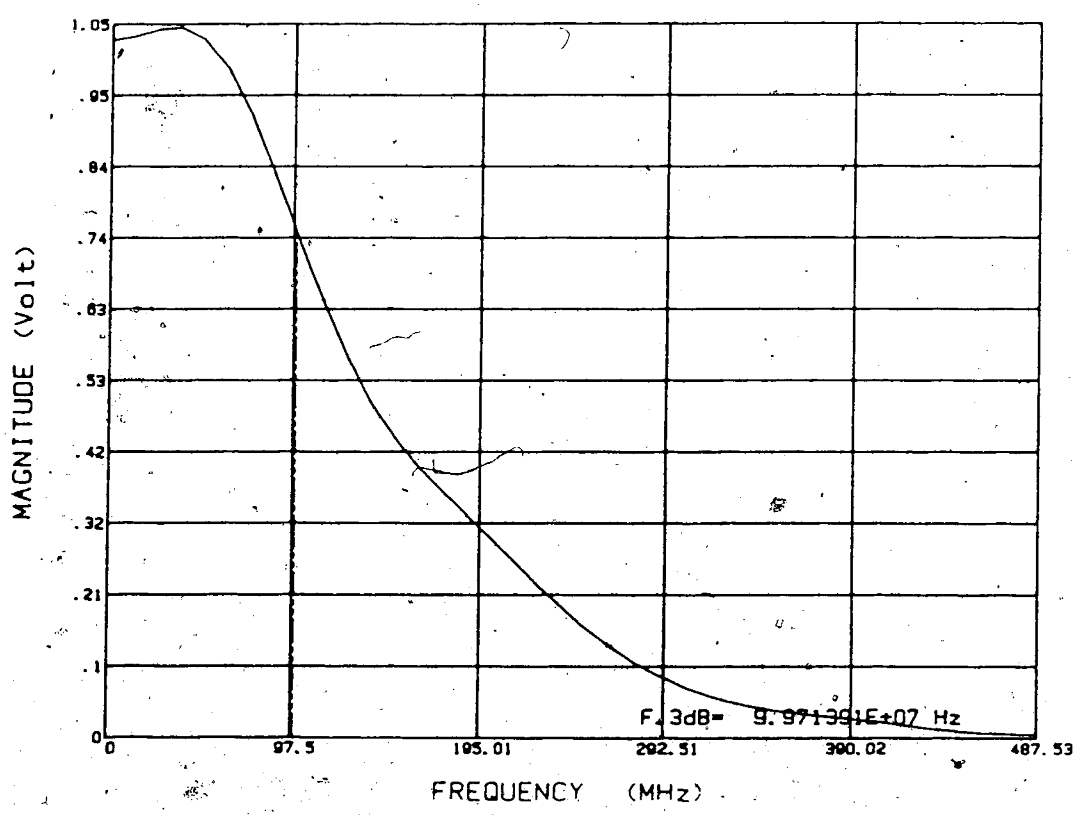


Fig.50 Frequency spectrum of the Hamming windowed data.

In order to reduce this distortion of the signal introduced by the Hamming window (or by any other simple window such as the Hanning, or the triangular), a hybrid one, based on the rectangular and the Hamming, was used instead.

The window had a main lobe narrower than that of the Hamming window, but increased sidelobe levels. The objective to be realized in the time domain with the window was to minimally smooth the waveform skirts without appreciably altering the duration of its main pulse. The window used is shown in Fig.51 in the time domain, and its spectral response is shown in Fig.52. In the time domain, this window was only affecting the first and last 10% of the original waveform, and in the frequency domain the width of the window's main lobe was lowered to ~50 MHz. Also, the first, second and third sidelobes had values of -14, -19 and -24 dB compared to -12.5, -19 and -21 dB for a rectangular window. Thus, the modified window caused less ripple in the frequency domain than the rectangular window, and less signal distortion in the time domain than the Hamming window.

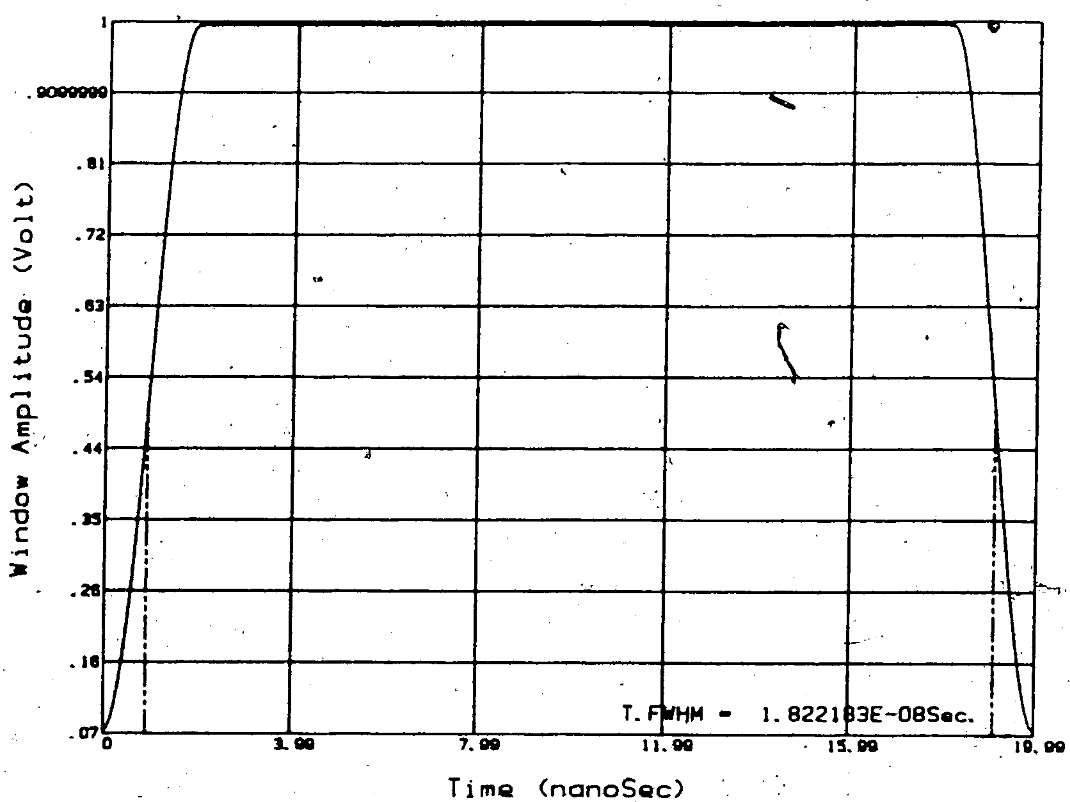


Fig.51 Modified Hamming window function.

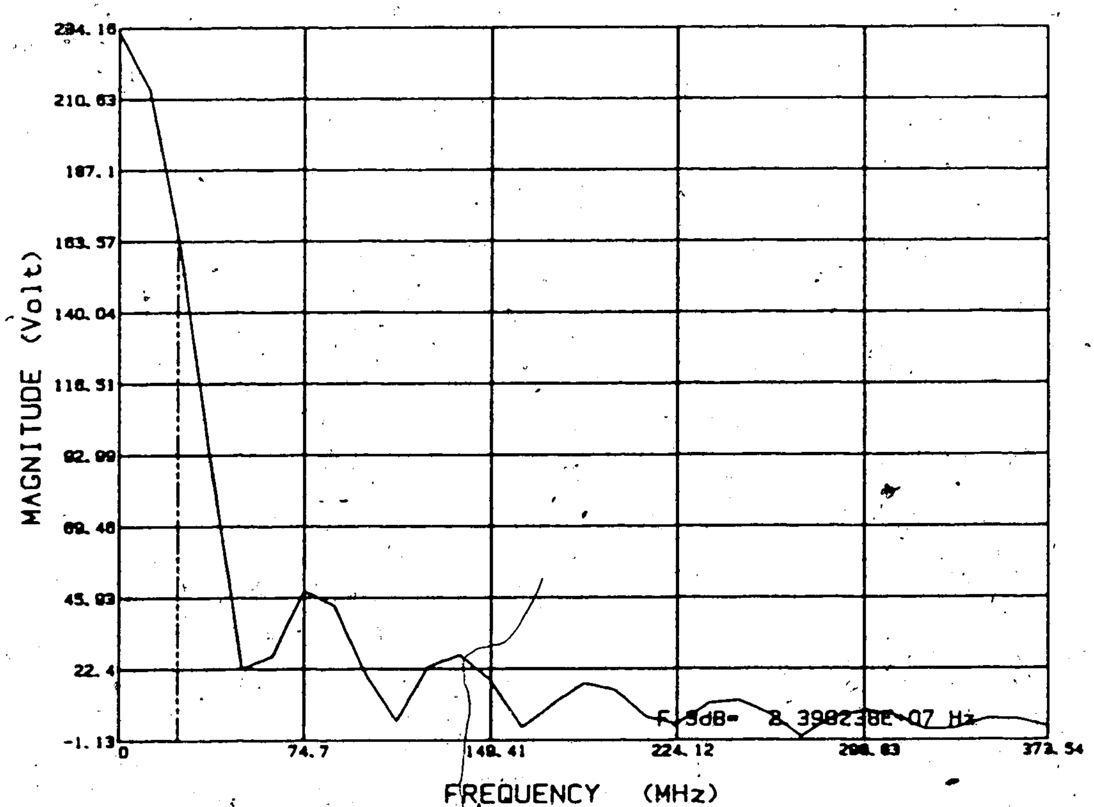


Fig.52 Frequency spectrum of the modified Hamming window function.

Fig.53 shows the input data of Fig.49 after the *modified window* (as this window will be called from here on) had been applied to it. Fig.54 shows the frequency spectrum of this windowed data. In the following chapter we will discuss how the precision of the computed frequency spectrum is affected by this windowing.

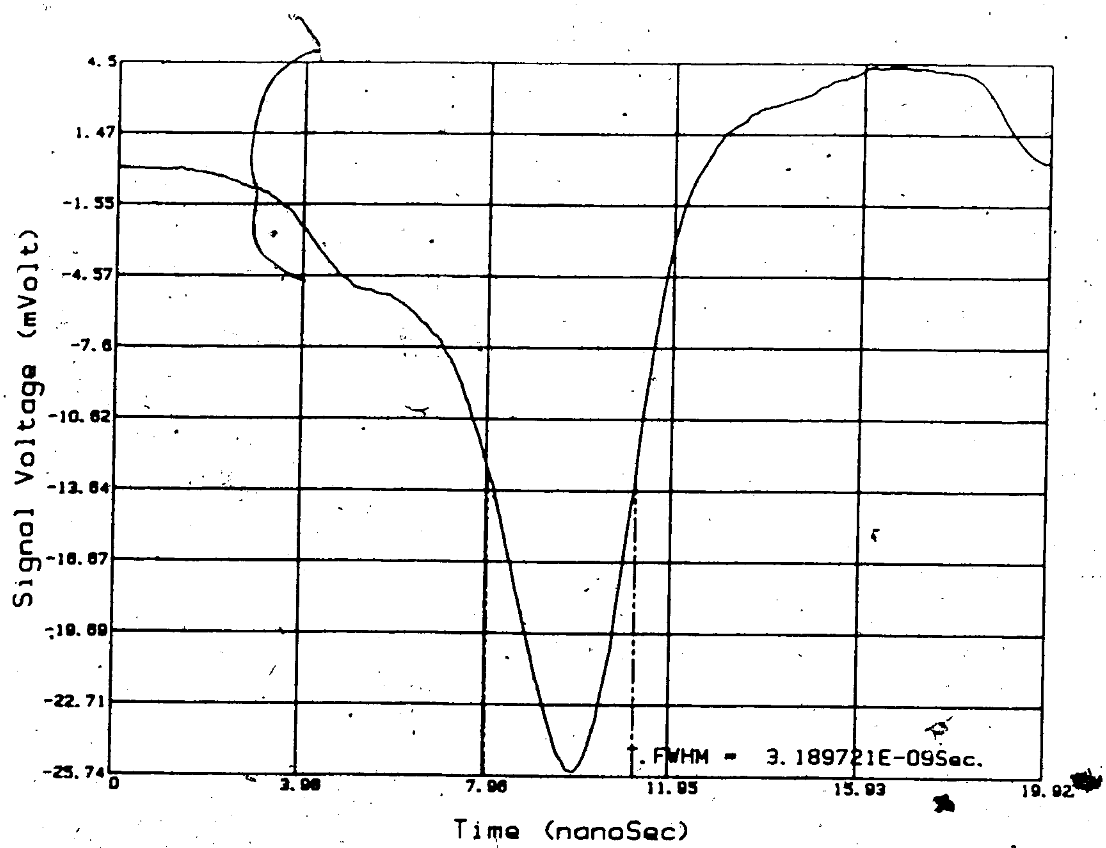


Fig.53 Data after the modified window has been applied.

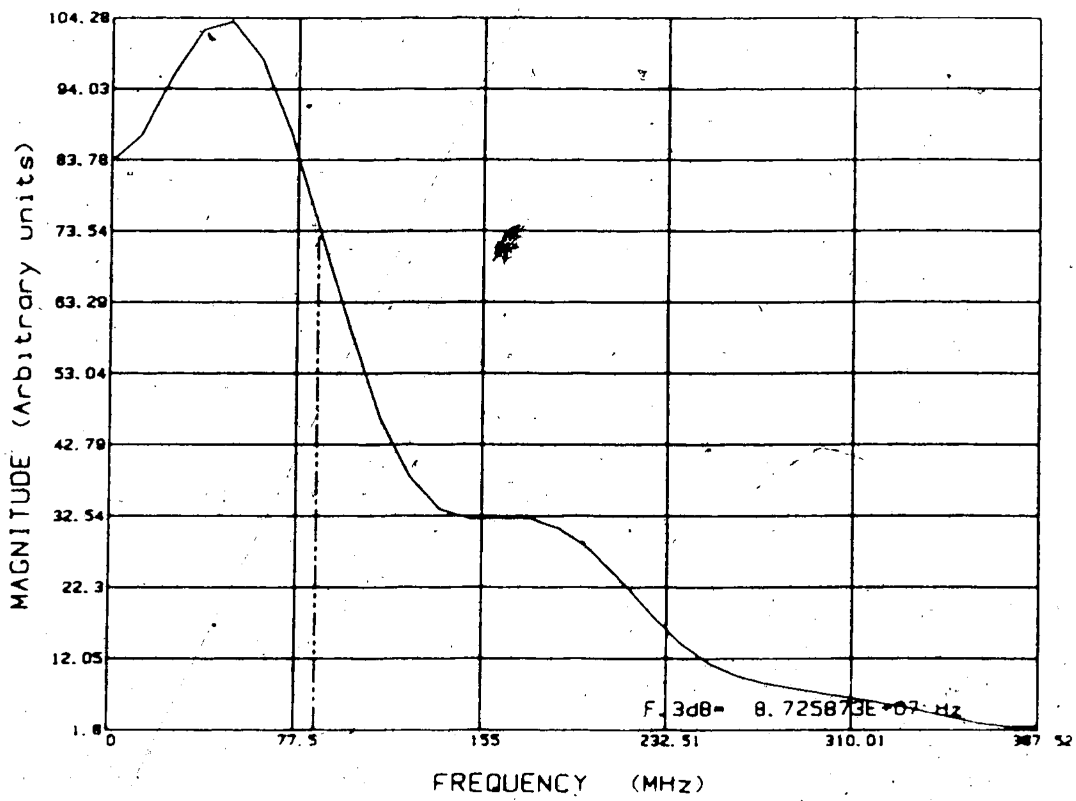


Fig.54 Frequency spectrum of the modified windowed data.

4.2.3 Aliasing Error (Decreased by High Sampling Rate)

To evaluate the FFT, the data modified by the window function must also be digitized, and this was realized with the sampling scope. However, the obtaining of discrete samples of the time-domain waveform may cause aliasing, or fold-over, errors.

Digitizing an analog waveform requires that the waveform's amplitude be sampled often enough to define it completely. The number of times that any waveform is sampled in a fixed period is referred to as the *sampling rate*. The well established sampling theorem (Nyquist criterion) states that the sampling rate must be at least twice the highest frequency present in the waveform for it to be defined completely. Failure to use a sufficient high sampling rate is the source of aliasing errors similiar to the ones shown in Fig.55 [35].

For this project, an extremely high sampling rate capability was available using the 7854 sampling scope, so that aliasing errors were reduced to insignificant levels. For example, with the waveform shown on Fig.30, the sampling period was 2 nsec/(256-1) = 7.84 psec. Thus, the equivalent sampling rate was equal to 1/7.84 psec = 127.5 GHz and the fold-over frequency (defined as half of the sampling rate) had a value of 63.75 GHz. The highest frequency content of any pulse measured for this project was ~3 GHz. Therefore, the aliasing effects were almost totally absent.

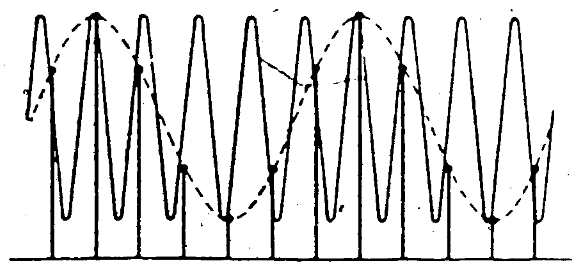


Fig.55 Aliasing errors.

4.2.4 Picket-Fence Effect Error (Decreased by Zero-Filling)

The picket-fence effect is of greatest concern when the frequency spectrum of a waveform contains discrete components (e.g. a sinusoid has a spectrum consisting of two impulses). The time domain values being fed to the FFT program are discrete and the frequency domain values obtained from the FFT are also discrete. In fact, the increments between samples in the time and frequency domains are related by

$$\Delta f = 1/N\Delta T. \quad (20)$$

Therefore, the frequency spectrum of a waveform is known only at discrete frequencies, separated by Δf hertz, and it is as if the real waveform's frequency spectrum is observed through a *picket-fence*! Clearly, increasing the number of samples within the same time frame is not a solution, since the frequency increment depends on the total frame width, not on the number of samples. Also, in some cases, increasing the duration of the time window is not a practical solution. For example, when a single pulse has to be analysed and adjacent pulses are very close, the trailing edges of the neighbouring pulses might cause inter-pulse interference.

A very effective way to remedy this problem is to extend both sides of the digitized frame with zeros. Fig.56 shows how the increment in the frequency-domain can be reduced by zero-filling in the time-domain.

For this project, the zero-filling technique was not utilized primarily to remedy the picket-fence effect, because most of the optical waveforms analysed had basically Gaussian shapes and thus did not exhibit inter-pulse interference. Rather, this technique was adopted for the following reasons,

The 3 dB frequency is defined as the frequency at which the energy-density spectrum has decreased 3 dB from its plateau value. For Gaussian-like signals, there is no plateau since the energy-density spectrum (defined as 10 log $|G(f)|^2$) starts decreasing right from the d.c. value. F_{3dB} was then defined as the frequency at which the energy spectrum had an amplitude 3 dB below its maximum value. If the f_{3dB} value of the analysed waveform's energy spectrum lies between two discrete frequencies, as shown in Fig.57,

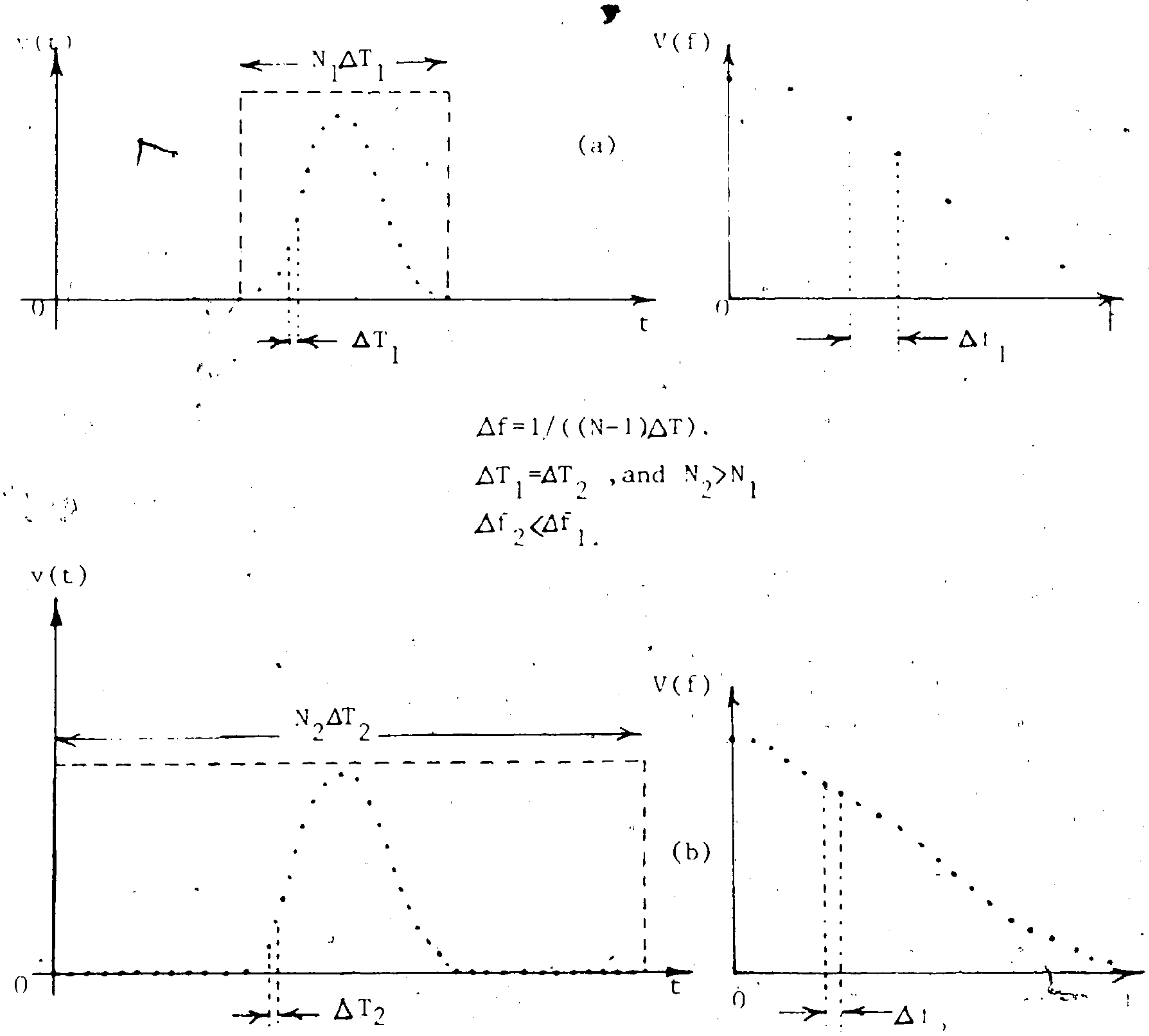


Fig.56 Time and frequency domain increments (a) without and (b) with zero-filling.

interpolation has to be done between the two adjacent frequencies. It was decided, for reasons of simplicity and also because the frequency responses analysed were generally decreasing monotonically, that a linear interpolation would be sufficient. (In fact, the precision of this interpolation was quantatively measured, and will be discussed in the next

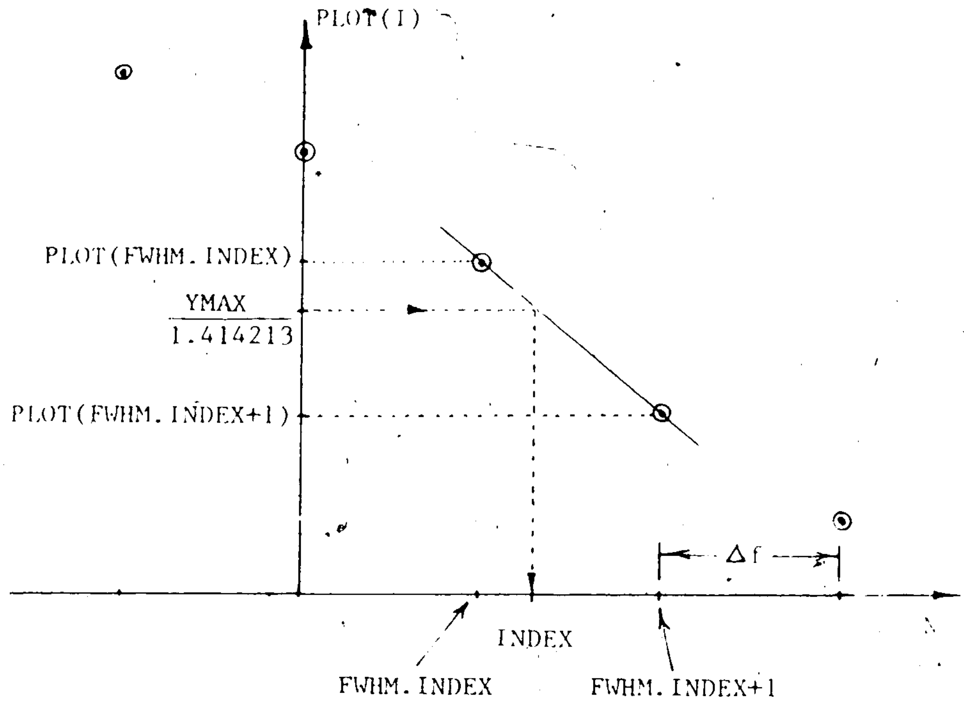


Fig.57 Graphical determination of the f_{3dB} frequency.

chapter.) It is clear, from Fig.57, that the time-domain zero-filling is reducing the frequency separation between adjacent values, and therefore increasing the precision of the linear interpolation.

Thus, the final version of the frequency spectrum computation program used the following algorithm:

1. obtain 256 samples of voltage from the sampling scope, together with the time increment defined as ΔT;
2. remove the d.c. content from each sample by zero-shifting the entire set of samples;
3. multiply the 256 samples with a 256 discrete point modified Hamming window function;

4. fill the center portion of a 1024 points array with the 256 windowed samples, and fill the remaining 768 points with zeros;
5. execute a 1024 point FFT on the array;
6. calculate the f_{3dB} frequency.

4.3 Other Programs

This chapter will discuss the different programs written to obtain the roll-off frequencies and the FWHM durations and to deconvolve the input signal and measuring device responses from the observed sampled waveforms.

4.3.1 F_{3dB} Calculation Program

The 3 dB roll-off frequency, symbolized by f_{3dB}, is computed using the FFT results. Once the FFT operation is completed, a 1024 points array is obtained; the first point corresponding to the frequency 0 Hz, the second point to a frequency of Δf, the third point to a frequency of $2\Delta f$, and so forth for the remaining 1021 points. The original data was 256 samples having a time increment ΔT between samples. This data was expanded to 1024 points by the zero-filling technique, so that the total width of the data processed was $1023\Delta T$ sec. Consequently, in the frequency domain, the increment between the discrete values obtain from the FFT is equal to

$$1/1024\Delta T = \Delta f. \tag{21}$$

With reference to Fig.57 and to the program listed in Appendix 3, the f_{3dB} frequency is calculated by: (i) finding which point of the FFT output array has the maximum amplitude (YMAX), this value being called MAXI. (ii) finding which point of the array has an amplitude the nearest to the maximum value minus 3 dB ($YMAX/\sqrt{2}=YMAX/1.414213$) , this value being called FWHM.INDEX ; (iii) executing a linear interpolation to obtain the exact location of the -3 dB frequency , this value being called INDEX. From the equations of the slope

$$\frac{PLOT(FWHM.INDEX)-PLOT(FWHM.INDEX+1)}{FWHM.INDEX-(FWHM.INDEX+1)} \qquad (22)$$

and

$$\frac{PLOT(FWHM.INDEX)-YMAX/\sqrt{2}}{FWHM.INDEX-INDEX}$$

, the value of INDEX is found from

$$INDEX=FWHM.INDEX + \frac{PLOT(FWHM.INDEX)-YMAX/\sqrt{2}}{PLOT(FWHM.INDEX)-PLOT(FWHM.INDEX+1)} \qquad (23)$$

, and the value of f_{3dB} is found by multiplying the value of the frequency increment Δf by the number INDEX-1. Thus,

$$f_{3dB}=(INDEX-1)\Delta f. \qquad (24)$$

4.3.2 τ Calculation Program

The duration of the Full Width Half Maximum (FWHM) pulse width , τ, was calculated in a similar way to that used for the f_{3dB} frequency. First, the d.c. content is removed from the waveform since we are strictly measuring impulses of light (any d.c. value included in the data obtained from the sampling oscilloscope is artificially added by it to the real data to modify its display on the screen). Second, the array containing the time domain waveform is scanned to determine the value at which the amplitude is a maximum; this point and this maximum value being called respectively MAXI and MAXARRAY. Then, the points at which the discrete array values have an amplitude nearest to MAXARRAY/2 are found. Linear interpolation similar to that mentioned above for the calculation of the f_{3dB} frequency is used to find precisely the two time coordinates that are located on either side of MAXI, as shown in Fig.59, and which are called DNI and UPI.

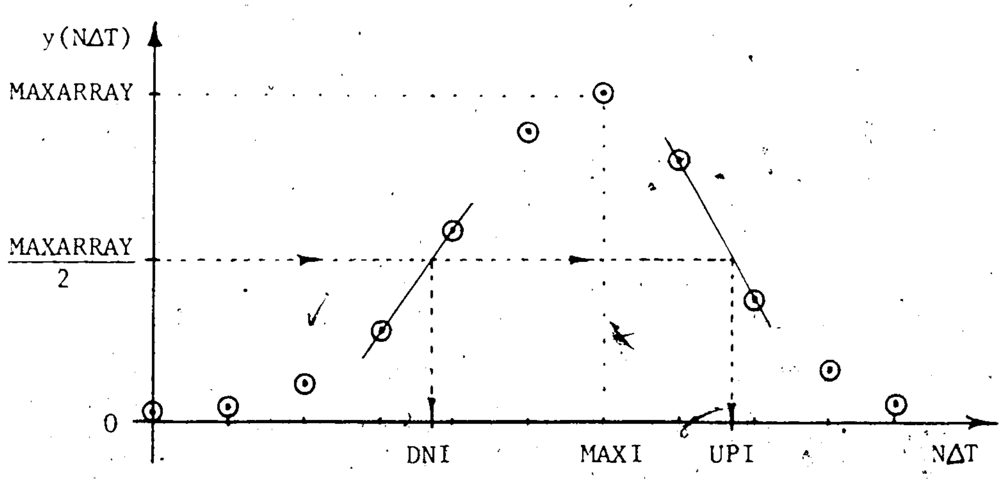


Fig.58 τ determination.

The value of τ is finally calculated by finding how many time increments separate these points UPI and DNI, and by multiplying the result with the time increment ΔT. Thus,

$$\tau=(UPI-DNI)\Delta T. \tag{25}$$

4.3.3 Deconvolution Program

In theory, both the input and the output optical signals on the fiber have to be considered in order to obtain a precise measurement of the frequency spectrum of the fiber. With reference to Fig.59, the following operations have to be done: (i) record the signal $v_{out_1}(t)$ at the far end of the fiber, (ii) record the signal $v_{out_2}(t)$ at the output of the fiber pigtail and (iii) deconvolve the effects of the measuring instruments and of the input pulse from the output $v_{out_1}(t)$, in order to obtain the optical fiber frequency spectrum. In fact, those measured signals mentioned above are a result of a convolution between the impulse responses of the input pulse, the fiber, the detector, the wideband amplifier and the sampling head represented respectively by $h_{in}(t)$, $h_1(t)$, $h_2(t)$, $h_3(t)$ and $h_4(t)$. Also, any other time-dispersal effects not specifically mentioned above can be grouped together as $h_5(t)$. Thus, we have

$$v_{out_1}(t)=v_{in}(t)*h_1(t)*h_2(t)*h_3(t)*h_4(t)*h_5(t), \tag{26}$$

where the symbol "*" means convolution. This is equivalent to multiplying the frequency spectra in the frequency domain as follows:

$$V_{out_1}(f)=V_{in}(f)\ H_1(f)\ H_2(f)\ H_3(f)\ H_4(f)\ H_5(f). \tag{27}$$

Similarily, in Fig.59(b), when the optical fiber is omitted,

$$v_{out_2}(t)=v_{in}(t)*h_2(t)*h_3(t)*h_4(t)*h_5(t), \tag{28}$$

or, in the frequency domain,

$$V_{out_2}(f)=V_{in}(f)\ H_2(f)\ H_3(f)\ H_4(f)\ H_5(f). \tag{29}$$

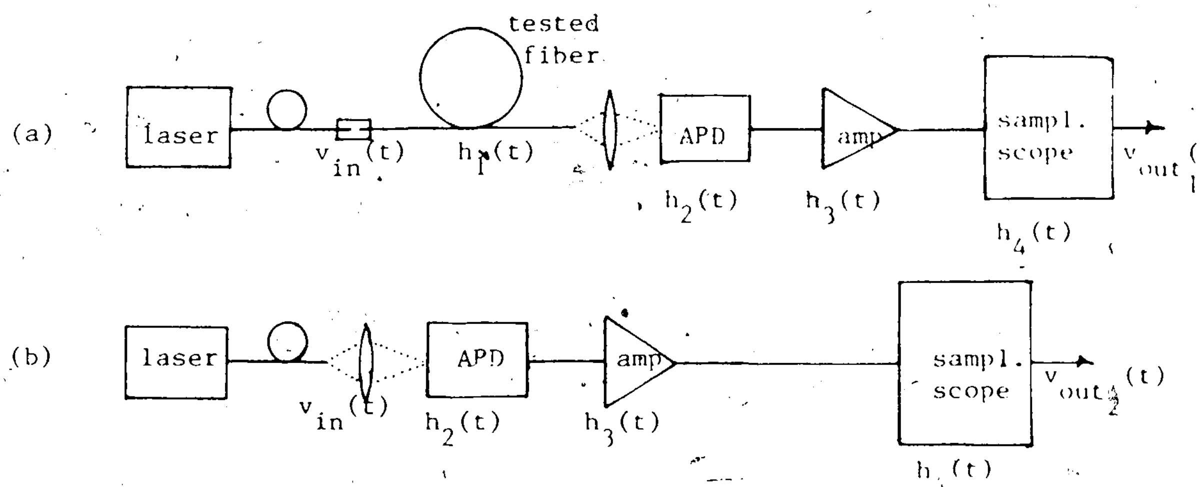


Fig.59 Convolution of impulse responses.

If $V_{out_1}(f)$ is divided by $V_{out_2}(f)$, the optical fiber frequency spectrum can be obtained:

$$\frac{V_{out_1}(f)}{V_{out_2}(f)} = \frac{V_{in}(f)\ H_1(f)\ H_2(f)\ H_3(f)\ H_4(f)\ H_5(f)}{V_{in}(f)\ H_2(f)\ H_3(f)\ H_4(f)\ H_5(f)} = H_1(f). \quad (30)$$

A program was written to do this division, and a listing is provided in Appendix 4. It performed a point-by-point division on two arrays, namely the discrete frequency spectra of the signals recorded at the input and at the output of the fiber. These waveforms were generally recorded on two different time domain scales, due to the large difference in their FWHM values (e.g. 120 psec at the input, and 2-3 nsec at the output). In this situation, the program was written to do a linear interpolation between the discrete values of the frequency domain arrays. Fig.60

illustrates this operation.

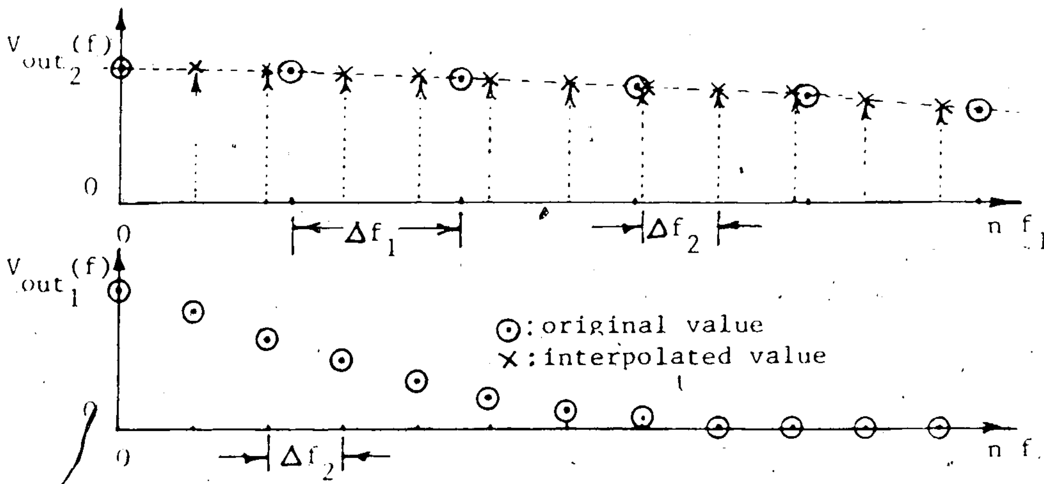


Fig.60 Deconvolution calculation.

4.4 Precision

A waveform of known characteristics, namely a Gaussian, was used to evaluate the precision of the FFT, the τ and the f_{3dB} calculation programs. This Gaussian, called $v_s(t)$, was given by

$$v_s(t)=\exp(-\alpha t^2). \tag{31}$$

The exact Fourier transform of this Gaussian waveform is given by

$$V_s(f)=A_0\exp(-\pi^2 f^2/\alpha) \tag{32}$$

, where the value of A_0 is equal to $(\alpha/\pi)^{-0.5}$.

It is possible in the case of a Gaussian waveform, to easily relate the pulse width τ to the 3 dB roll-off frequency f_{3dB} as follows:

$$0.5(\alpha/\pi)^{0.5}=\exp(-\alpha(\tau/2)^2$$

$$\therefore \quad \alpha= -\ln(0.5)/(\tau/2)^2$$

$$=2.7726/\tau^2 \qquad (33)$$

, and in the frequency domain

$$1/\sqrt{2}=\exp(-\pi^2 f_{3dB}^2/\ \alpha)$$

$$\therefore \quad \alpha=-(\pi f_{3dB})^2/\ln(1/\sqrt{2})$$

$$=28.4776\ f_{3dB}^2 \qquad (34)$$

Then, using (22) and (23),

$$2.7726/\tau^2=28.4776\ f_{3dB}^2$$

$$\therefore \quad (f_{3dB})(\tau)=0.3120 \qquad (35)$$

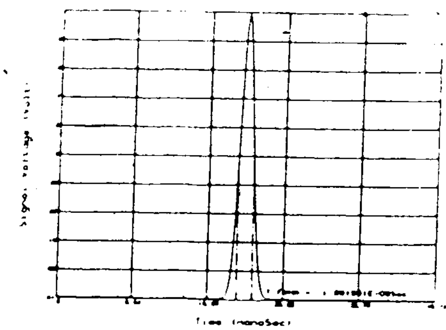
An FFT was computed on two 256 points Gaussian waveforms of the same width, but looked at through two different time windows. Thus, the two Gaussians had different time increments, and the results are shown in Figs.61 and 62. In both cases, the pulse width τ was approximatly equal to 1.991 nsec. Then, according to equation 24, the theoretical f_{3dB} should have a value of

(0.3120/1.991 nsec)=156.7 MHz.

Comparing this 156.7 MHz value to the 3 dB frequencies computed using the program yields an error of less than 1% in both cases. Also, from equation 20, the theoretical τ is obtained from

$$0.5=\exp(-\alpha(\tau/2)^2). \qquad (36)$$

Thus, in this case, α had a value of 7×10^{17}. Thus,



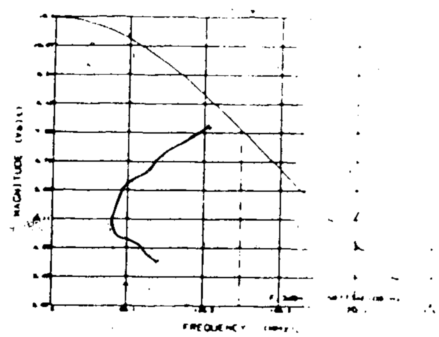


Fig.61 FFT on the first Gaussian waveform.

$$\ln(0.5)=(-7\times10^{17})(\tau/2)^2 \qquad (37)$$

$\therefore \quad \tau=1.9902$ nsec.

Thus, the error in the two values of τ was less than 1%. Finally, from equation 21, the theoretical value of f_{3dB} is obtained from

$$1/\sqrt{2}=\exp(-\pi^2(f_{3dB}^2/\alpha)). \qquad (38)$$

$\therefore f_{3dB}=156.78$ MHz.

In both cases, the computed value of f_{3dB} had an error of less than 1%.

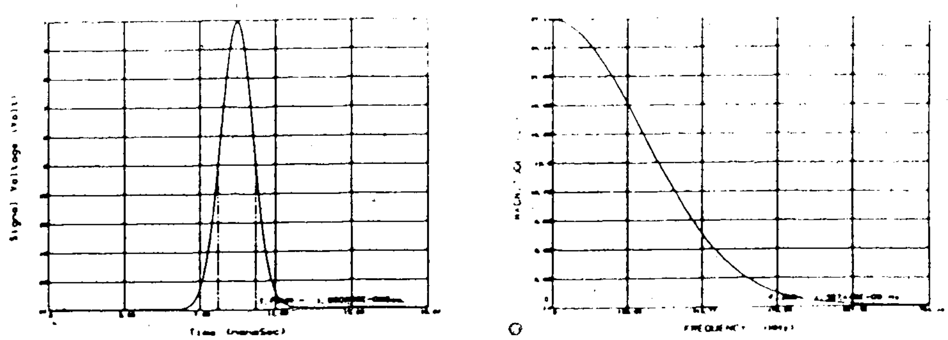


Fig.62 FFT on the second Gaussian waveform.

The performance of the computational methods was also tested by calculating the frequency spectrum of the wideband amplifier. A first waveform was recorded immediately at the output of the APD ; this waveform is shown in Fig.63, and its frequency content is shown in Fig.64.

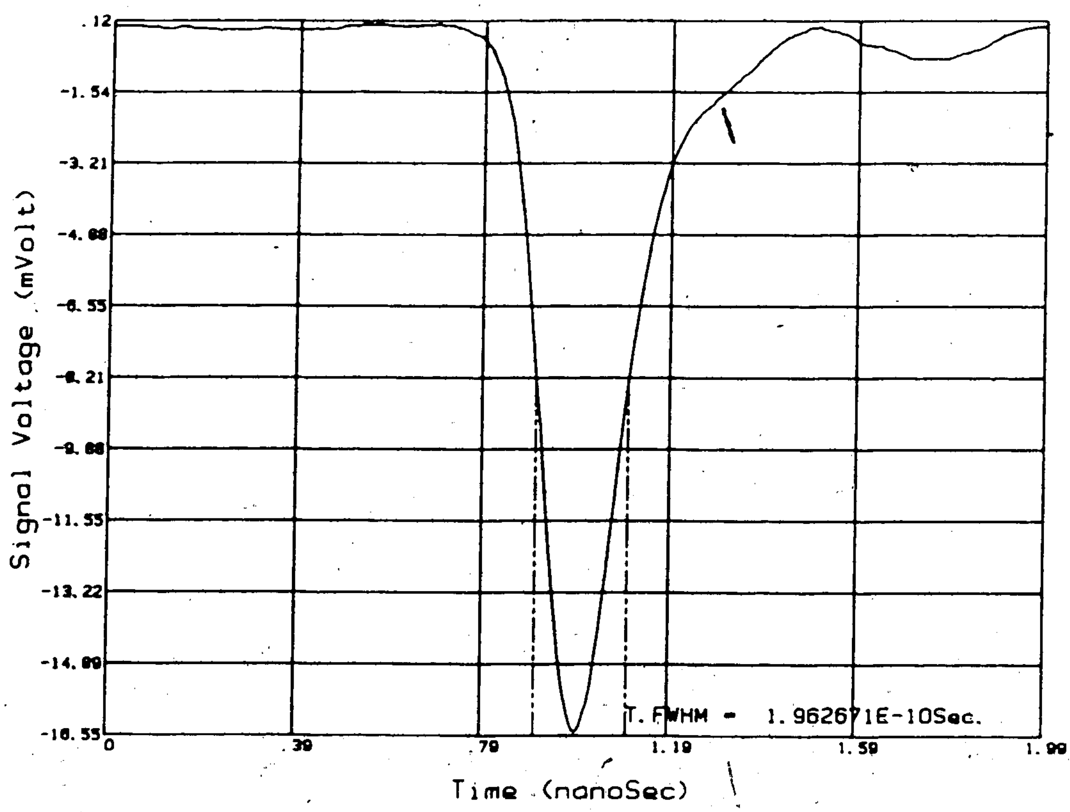


Fig.63 Amplifier input signal.

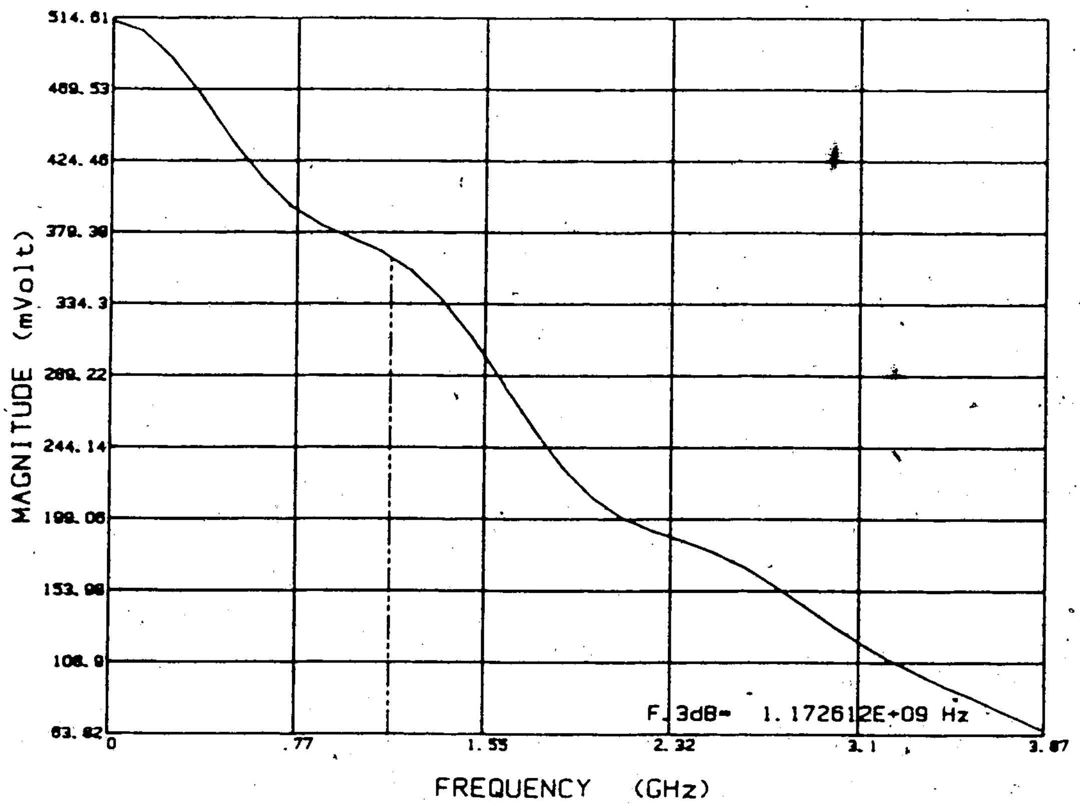


Fig.64 Frequency spectrum of the amplifier input signal.

Then, the wideband amplifier was inserted between the APD and the sampling scope (neutral density filters were used in front of the detector to attenuate the optical signal in order to prevent saturation of the amplifier), and a second waveform was recorded; this waveform and its computed frequency spectrum are shown in figs.65 and 66, respectively.

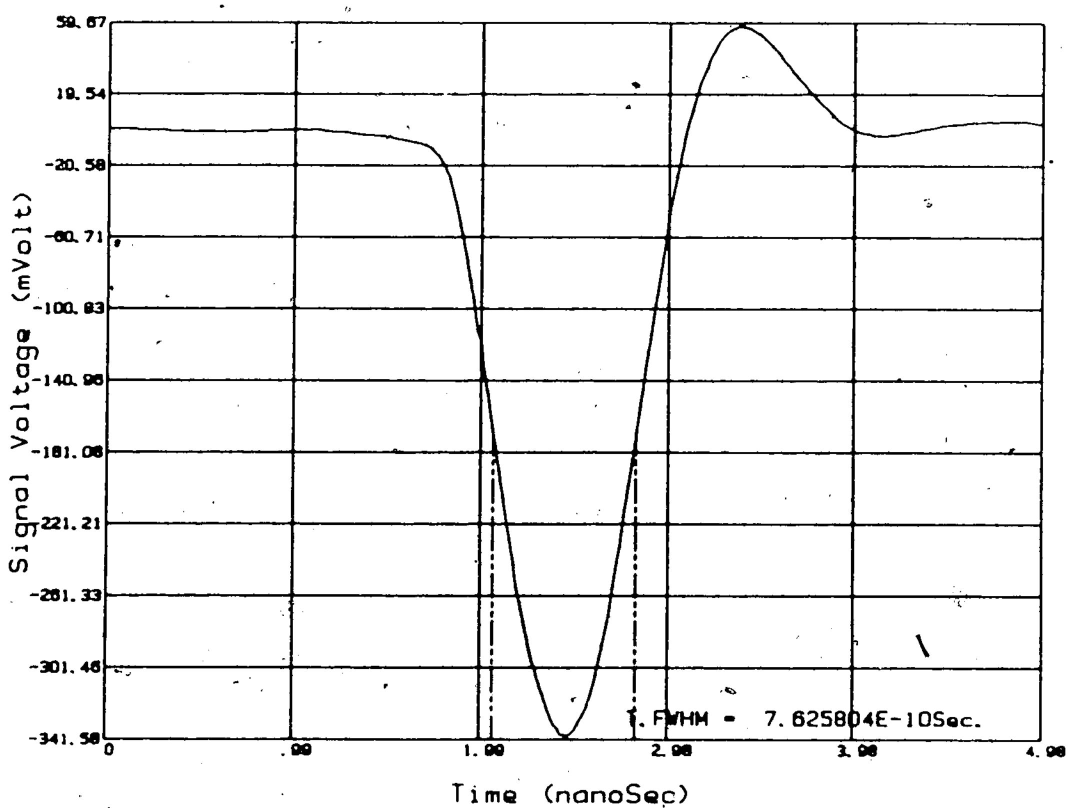


Fig.65 Amplifier output signal.

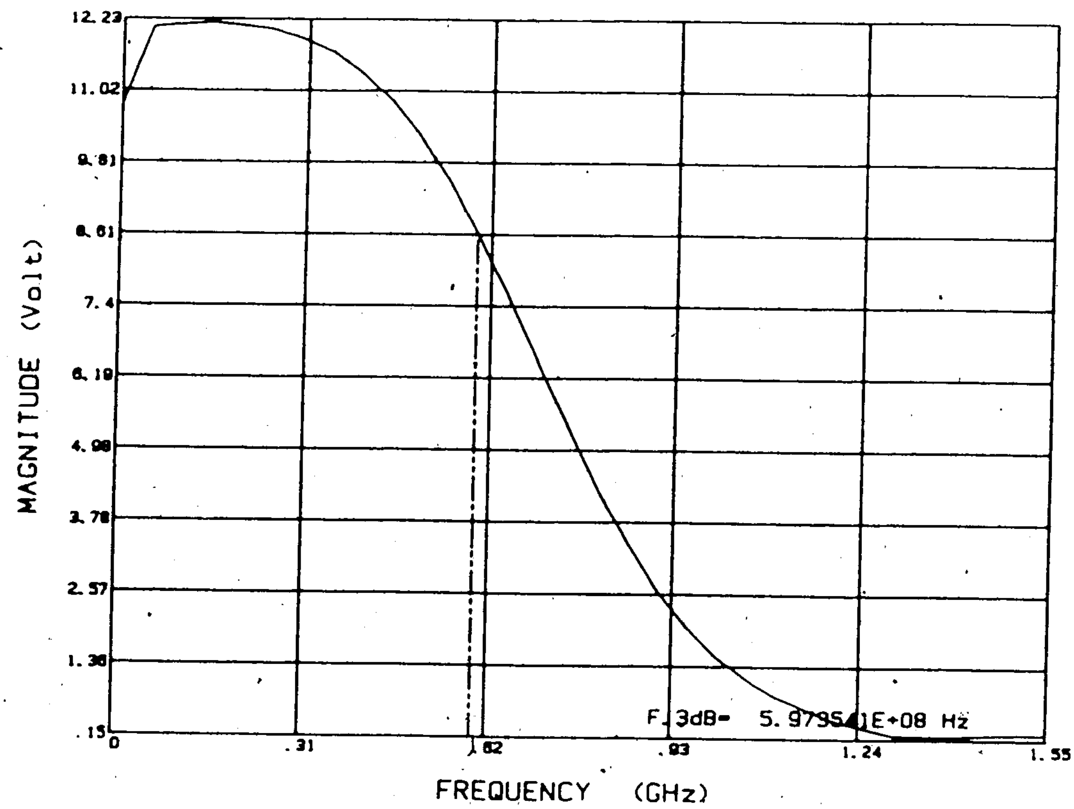


Fig.66 Frequency spectrum of the amplifier output signal.

Finally, using the deconvolution program discussed above, the second waveform frequency spectrum was divided point by point by the first waveform frequency spectrum in order to obtain the frequency response for the wideband amplifier. The results are shown in Fig.67 and comparison with the results obtained earlier (section 4.2) using noise source measurements yields an error of less than 1%. This precision was sufficient for this project and the programs were judged adequate.

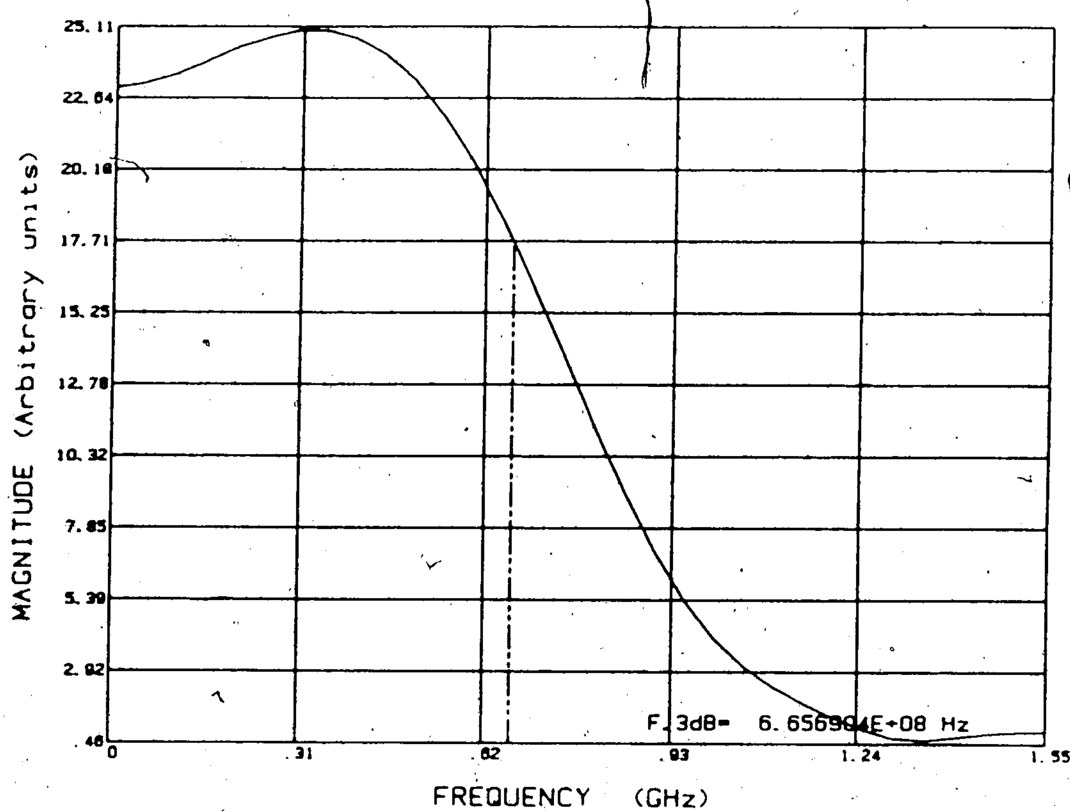


Fig.67 Wideband amplifier computed frequency spectrum.

5. EXPERIMENTAL RESULTS

This chapter will present a discussion of the measurements made on actual multimode optical fibers. The first part will deal with the preliminary results obtained in the lab on various spooled lengths of fiber, and the second part will report on the measurements made in the field on an assortment of fibers used to interconnect Edmonton Telephone Co. central offices.

5.1 Laboratory Results

In order to evaluate the measurement technique discussed in the preceeding chapters for the testing of relatively long optical fiber links, tests on spooled fiber were made in the lab prior to the field measurements.

Four fibers of different lengths were butt-joined together one-by-one in order to observe the changes in the frequency spectrum as the fiber length was increased. Fig.68 illustrates the procedure. First, the optical pulse right at the output of the laser pigtail was recorded. This input pulse is shown in Fig.69 and its frequency spectrum is shown in Fig.70. Then, a section of 3 km of fiber was butt-joined to the laser pigtail and the signal at the output of the fiber was recorded. The operation was repeated for butt-joining additional sections of 1.8, 1.6 and 3 km, one after each other, in order to measure the progressive degradation of frequency response as the total length was increased.

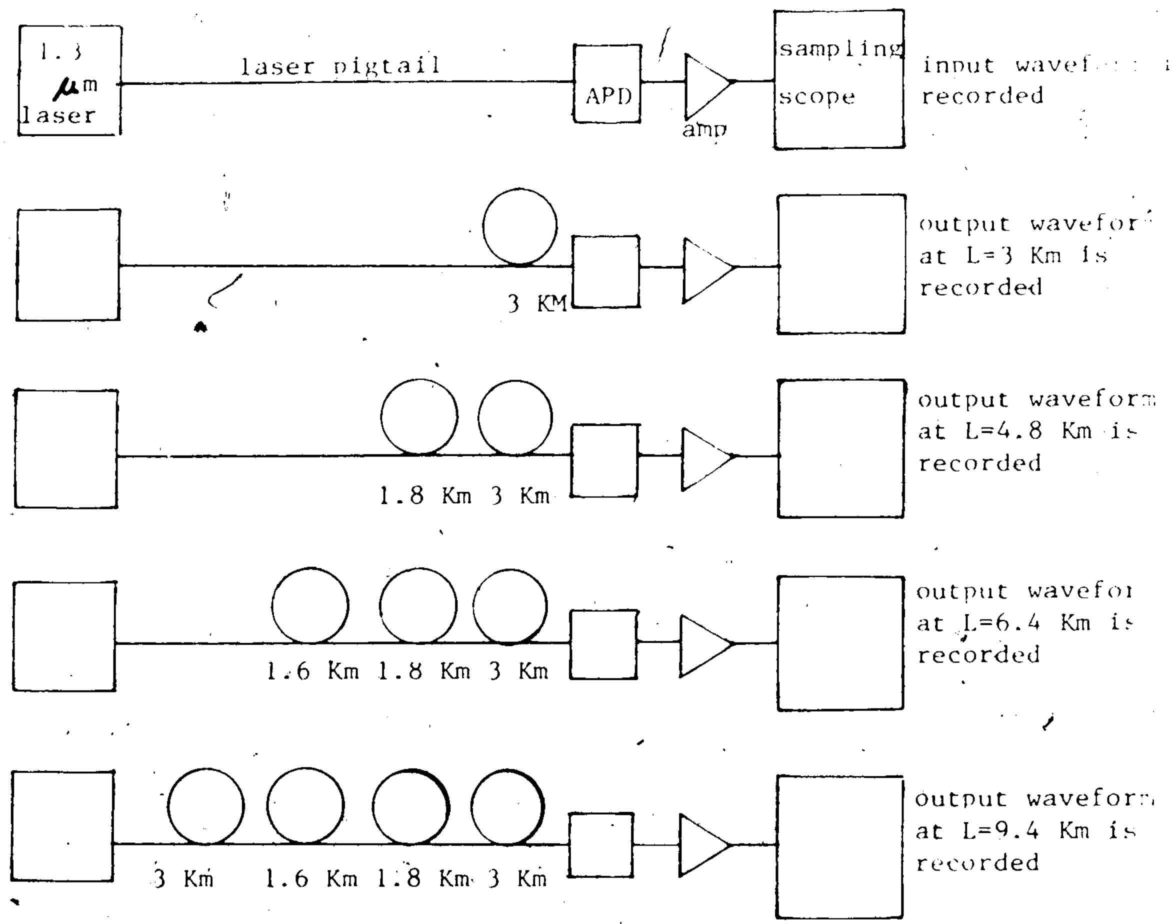


Fig.68 Laboratory measurement procedures.

The signal shown in Fig.69 is the optical signal at the input end of the fiber observed with the APD and amplifier connected to the oscilloscope. This signal has obviously been broadened to some extent by the frequency limitations of the bandpass of these two units. All waveforms given in Figs.69, 71, 73, 75, 77, 79, 80, 82, 84, 86, 88, 90, 92 and 94 contain these broadening components. It must be stressed that the deconvolution technique, eq.30, removes these broadening effects. Thus, the broadening components that affect the spectrum shown in Fig.70 are totally removed by this deconvolution procedure when it is applied to obtain the fiber frequency response. An example is Figs.71 and 72. Fig.72 is the net result of the entire FFT+deconvolution procedure and therefore contains no frequency degradation due to any of the measurement equipment. Of course this same technique was also applied for Figs.74, 76, 78, 81, 83, 85, 87, 89, 91, 93 and 95.

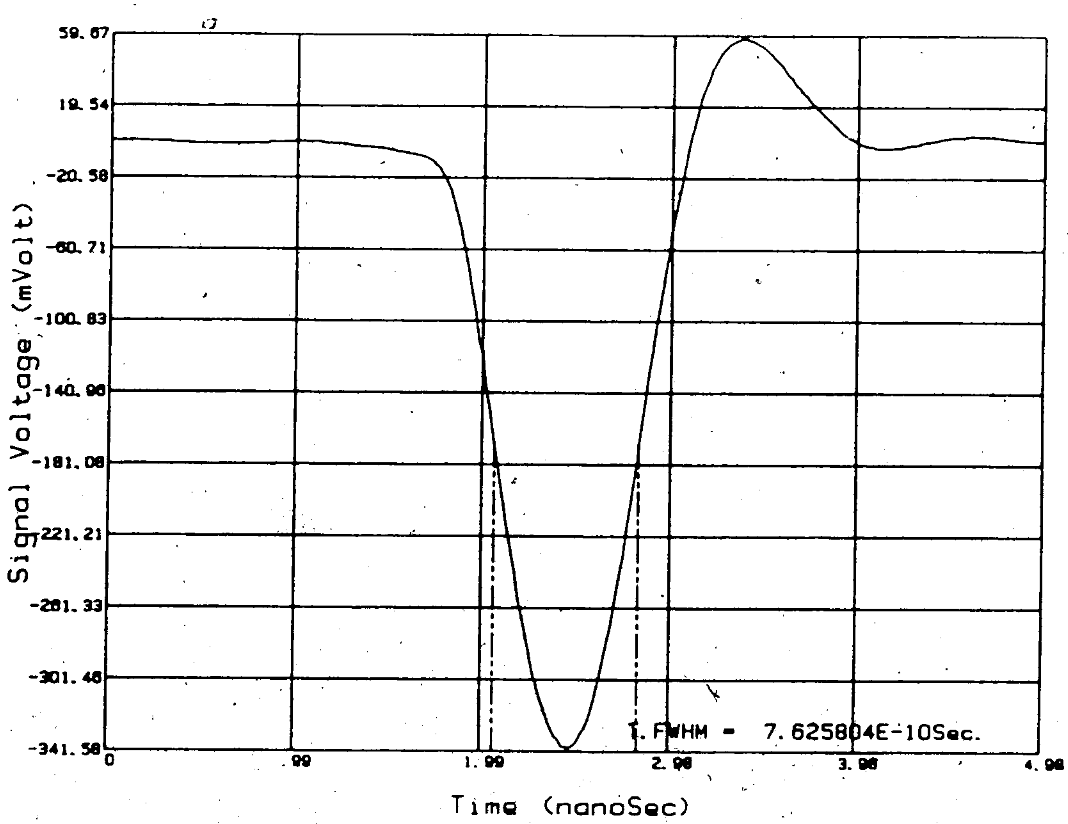


Fig.69 Optical pulse at the input to the fiber.

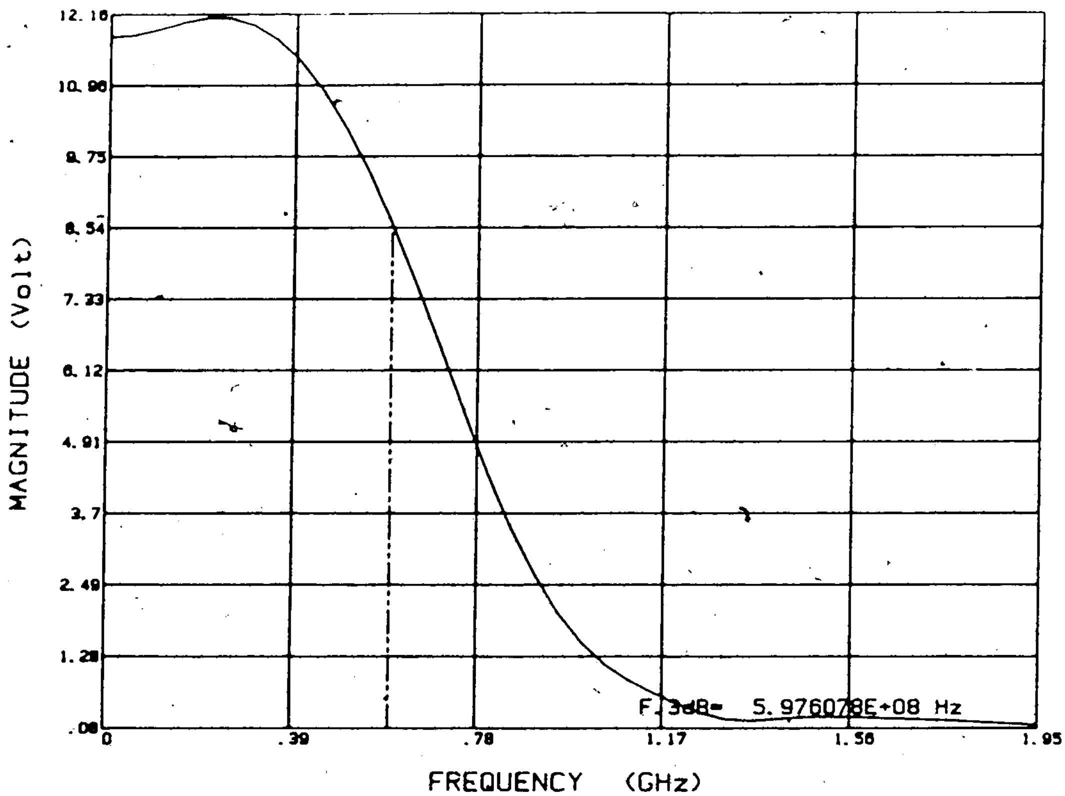


Fig.70 Frequency spectrum of the optical pulse at the input to the fiber computed using an FFT. This spectrum contains broadening components due to the finite risetime of the APD and amplifier.

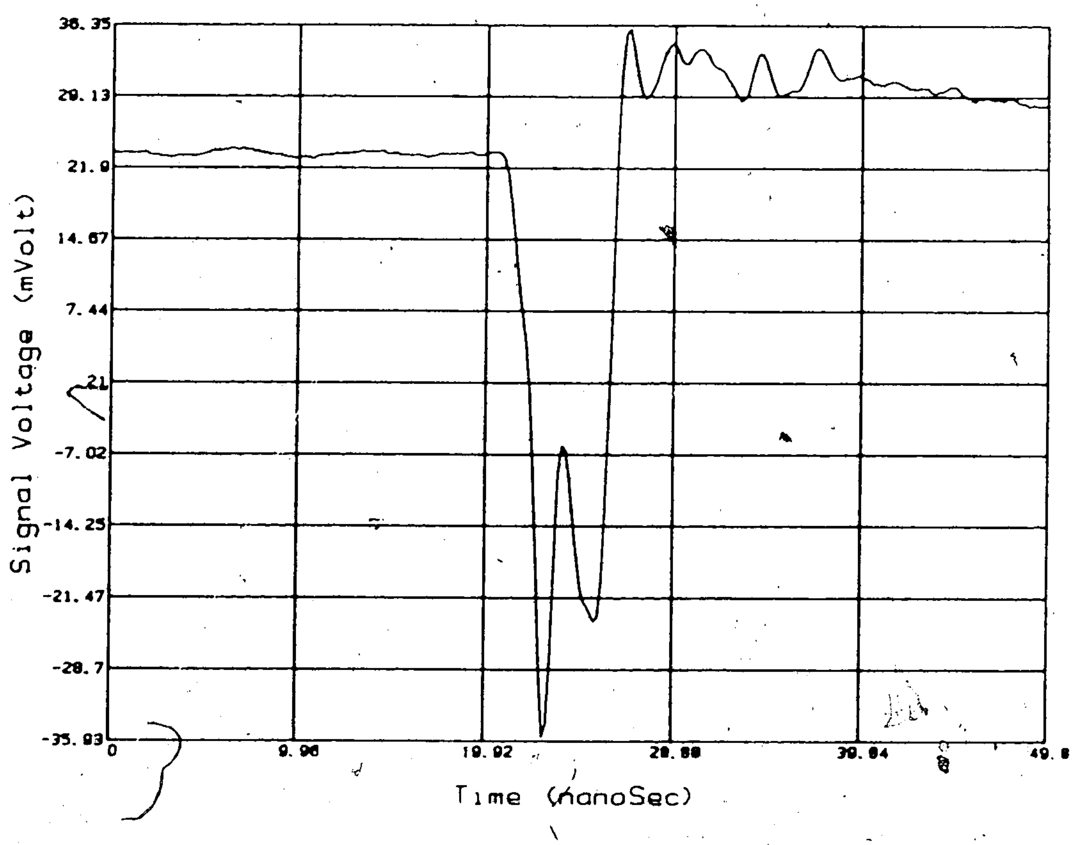


Fig.71 Optical time domain pulse at l=3 Km.

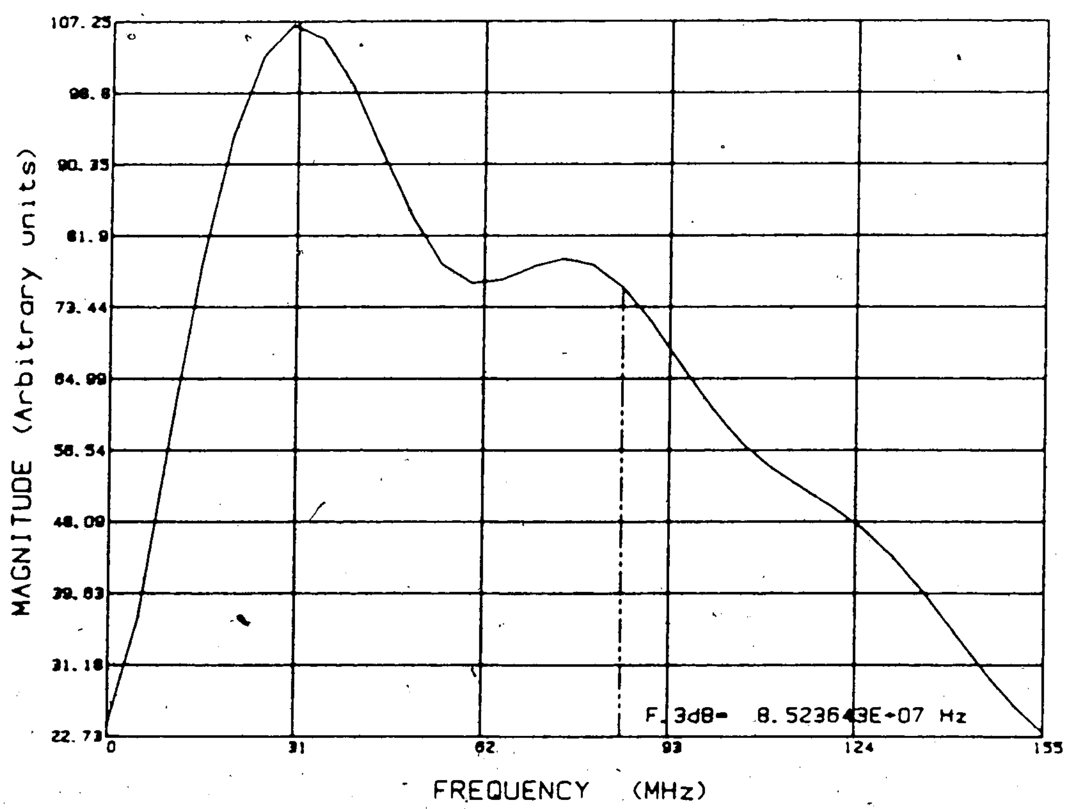


Fig.72 Computed fiber frequency response for l=3 Km.

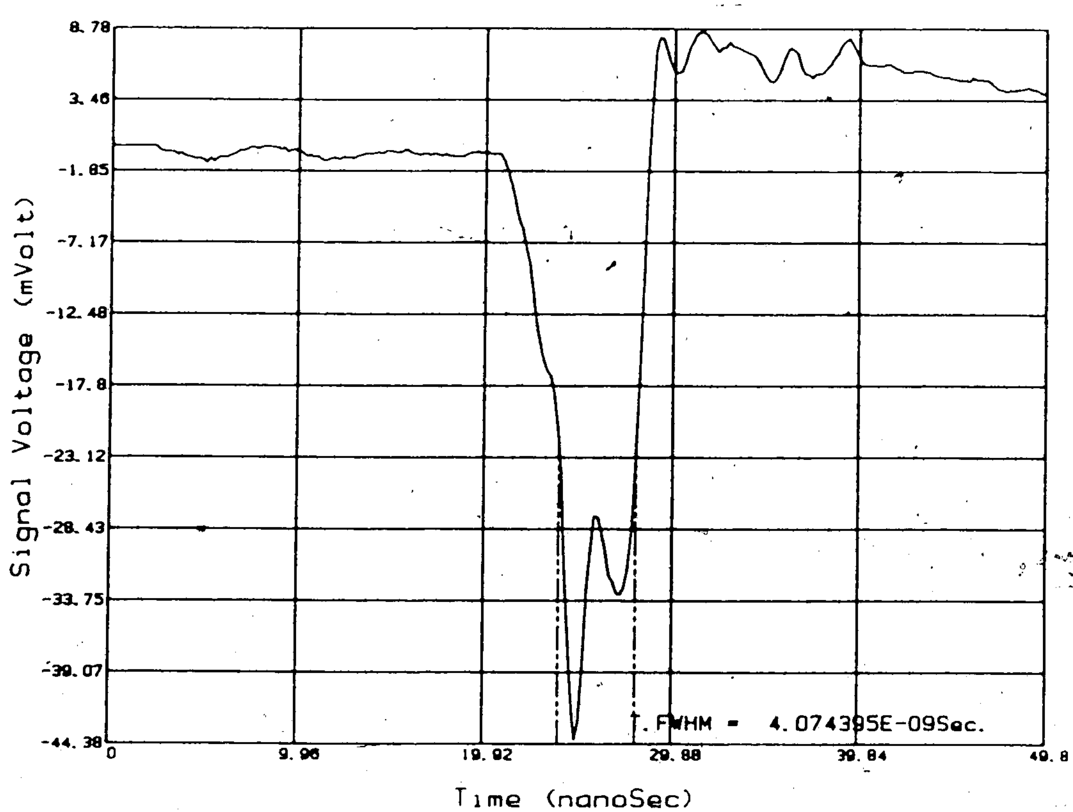


Fig.73 Optical time domain pulse at l=4.8 Km.

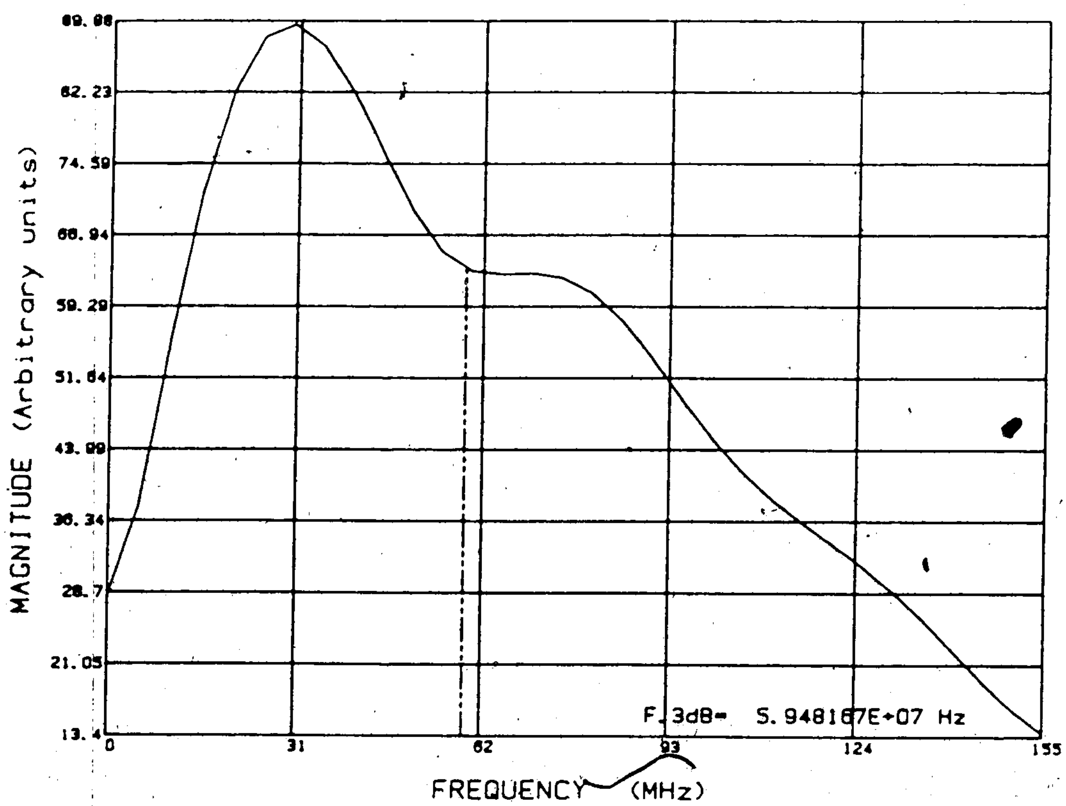


Fig.74 Computed fiber frequency response for l=4.8 Km.

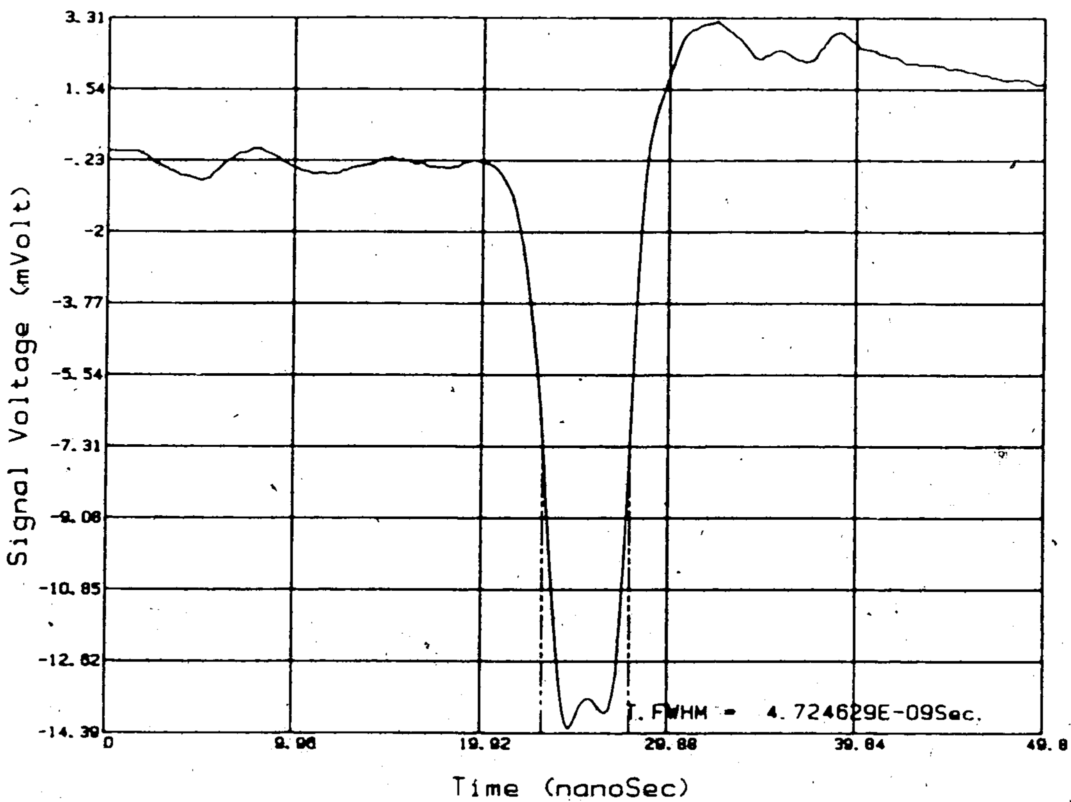


Fig.75 Optical time domain pulse at l=6.4 Km.

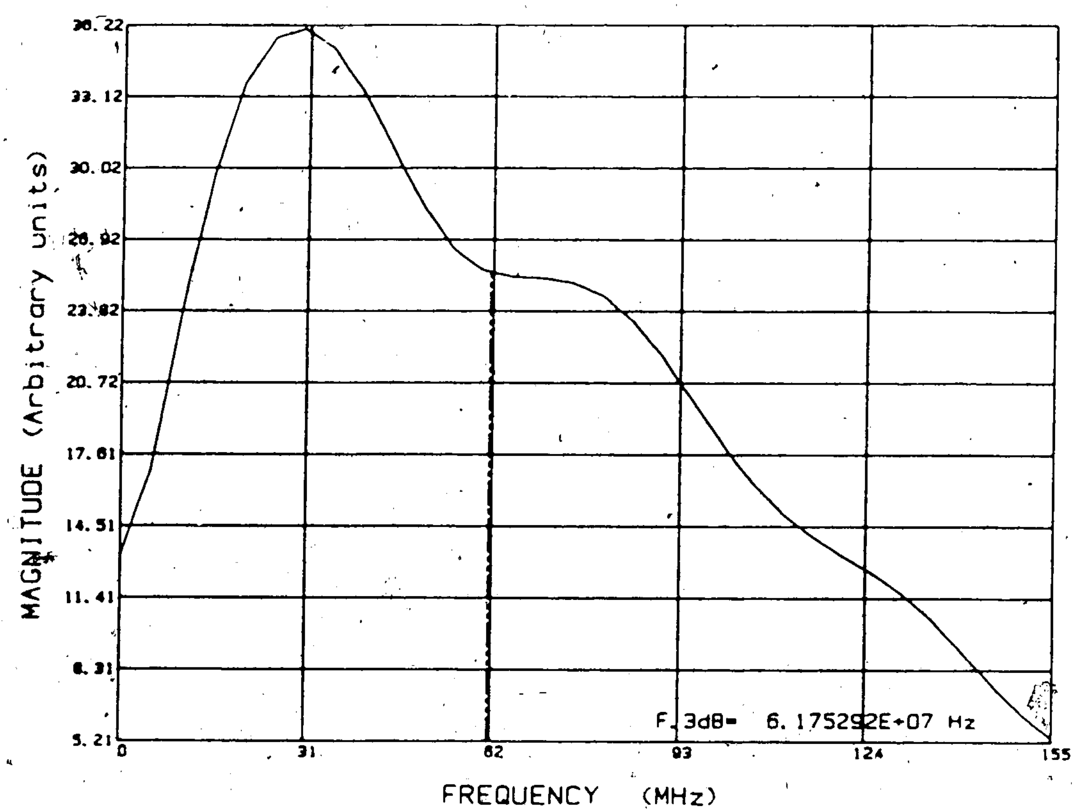


Fig.76 Computed fiber frequency response for l=6.4 Km.

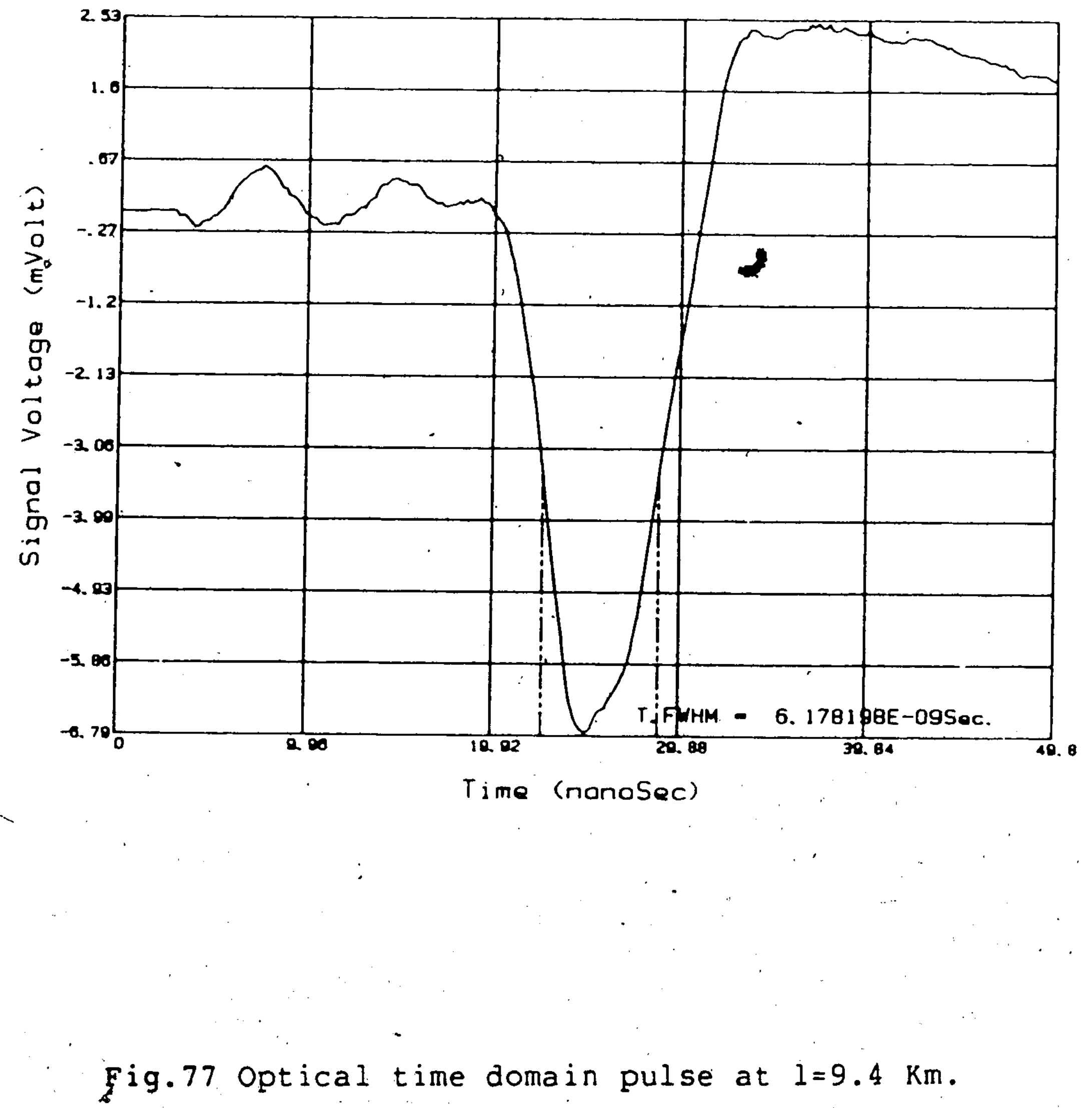


Fig.77 Optical time domain pulse at l=9.4 Km.

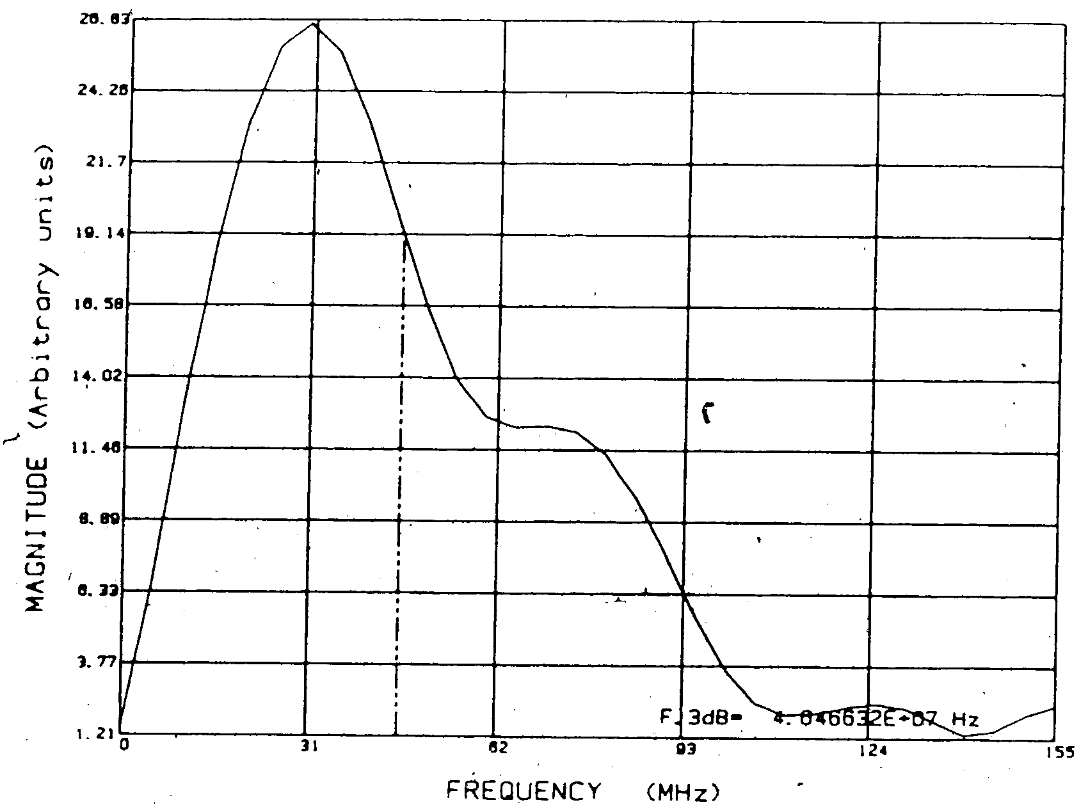


Fig.78 Computed fiber frequency response for l=9.4 Km.

The time domain waveforms at the far end of the fiber seem to consist of the superposition of two dispersed pulses. Care was taken throughout these experiments to ensure that this double-pulse effect was not caused by reflections. Rather, it was probably caused by an unexpected dispersion behavior for the initial 3 km section. Fig.79 shows two waveforms superimposed: the one having the larger "after-pulse" in its trailing edge is similar to the waveform recorded in Fig.71, and the second waveform is the optical signal measured at the same location in the optical fiber system but with a serpentine optical filter inserted in front of the detector to remove the higher order optical modes. The amplitude scale for these two waveforms was the same and it is clear that the amplitude of the main pulse has been less affected than the amplitude of the after-pulses. Consequently, the appearance of after-pulses is explained by a severe modal dispersion in the fiber.

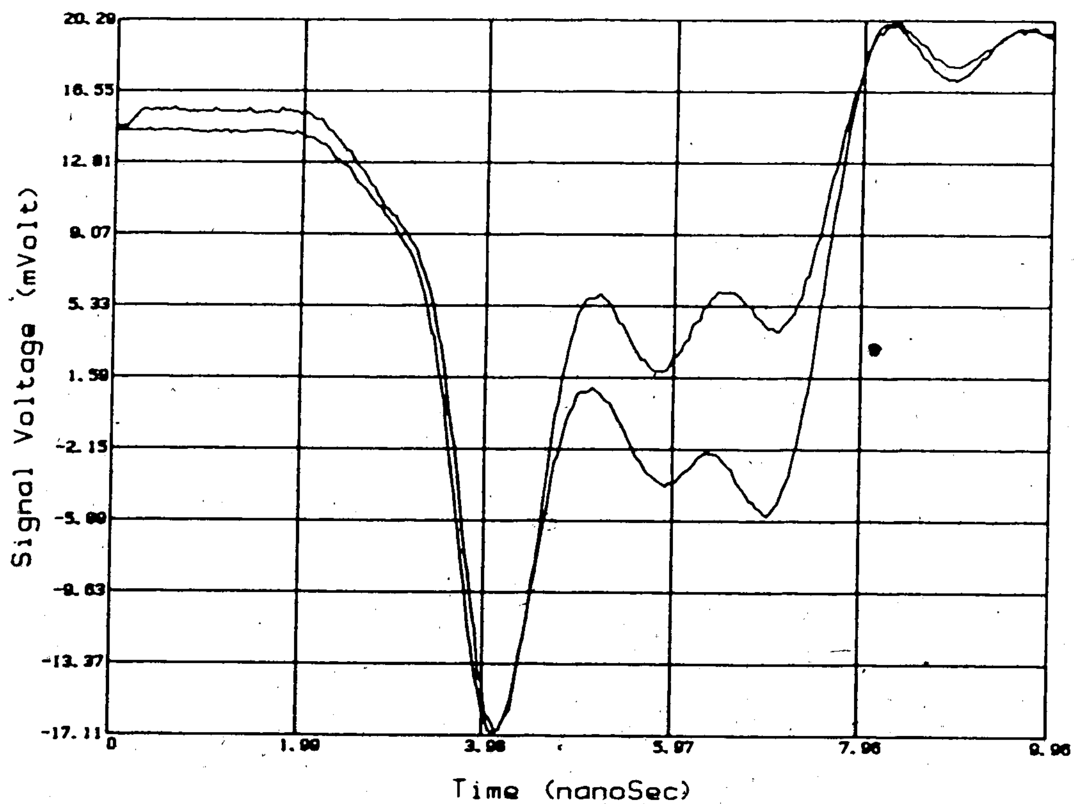


Fig.79 Effects of dispersion.

5.2 Field Results

The field measurements were realized using the same equipment set-up as for the lab measurements. Measurements were made on two trunks in the Edmonton Telephone Co. local area network: first on a section of 4432 m between the offices of LENDRUM and STRATHCONA ; and second on a section of 7826 m joining NORWOOD and STRATHCONA.

Figs.80 to 95 show the signals recorded at the far end of the fiber links as well as their computed deconvolved frequency spectra.

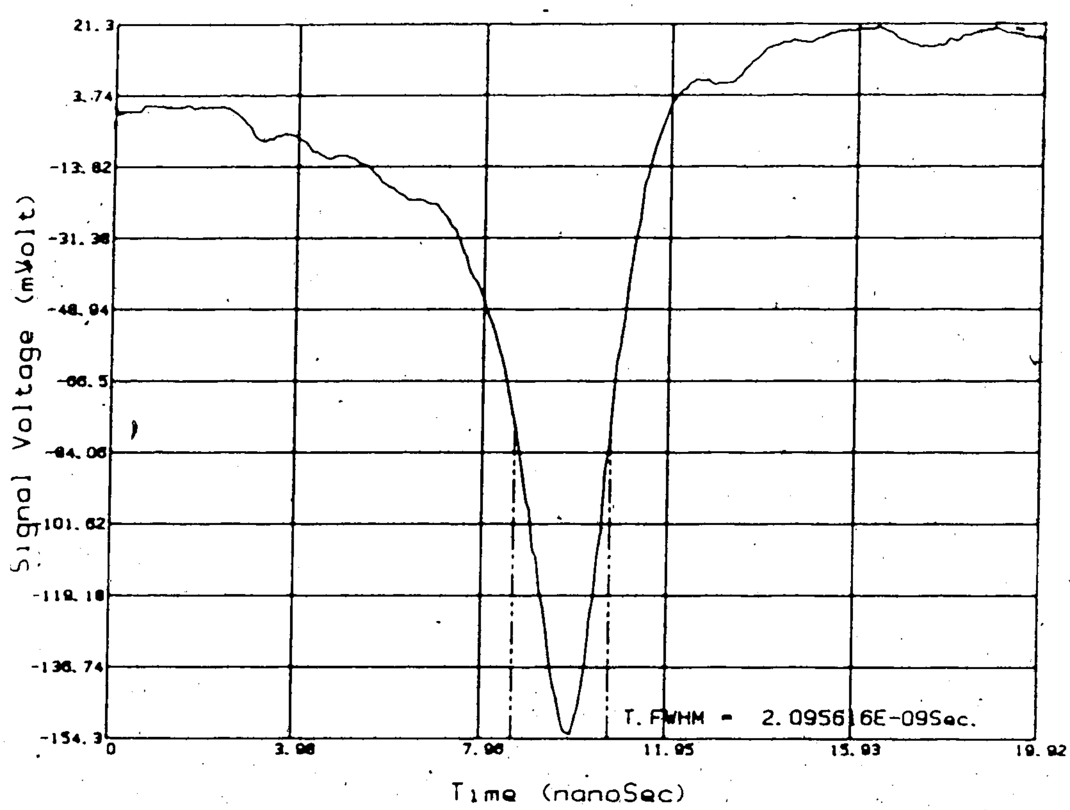


Fig.80 Optical time domain pulse at l=4432 m (line #7).

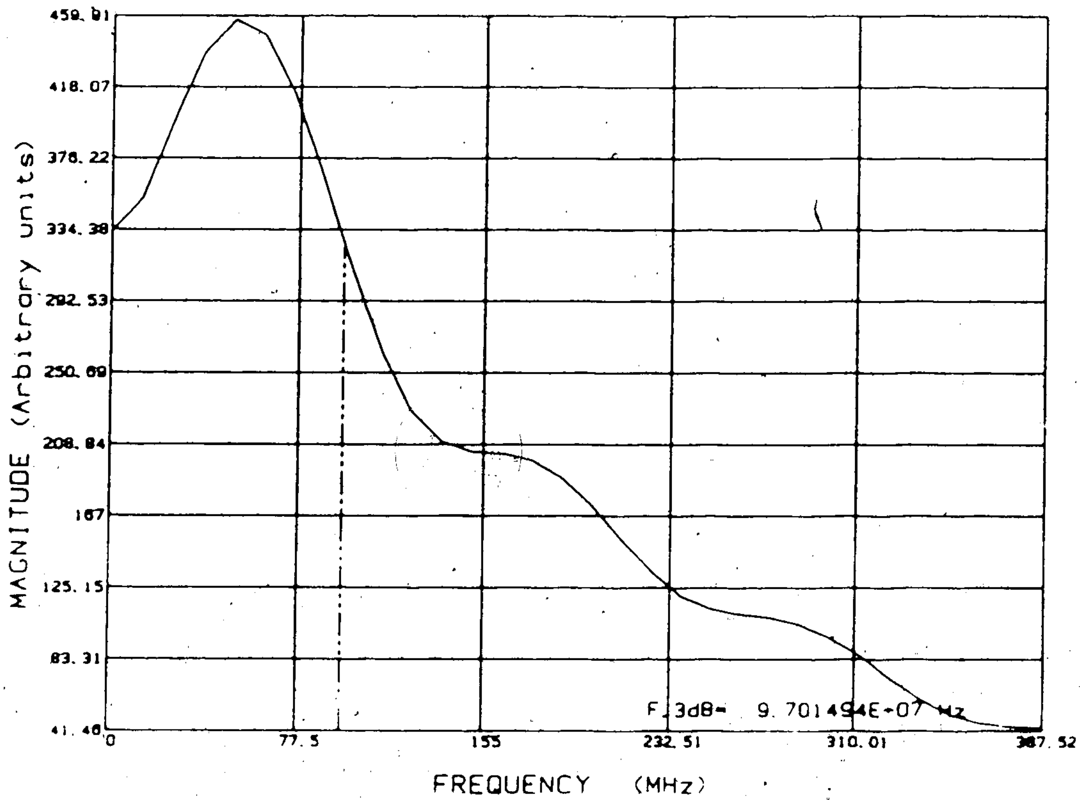


Fig.81 Computed fiber frequency response for l=4432 m (line #7).

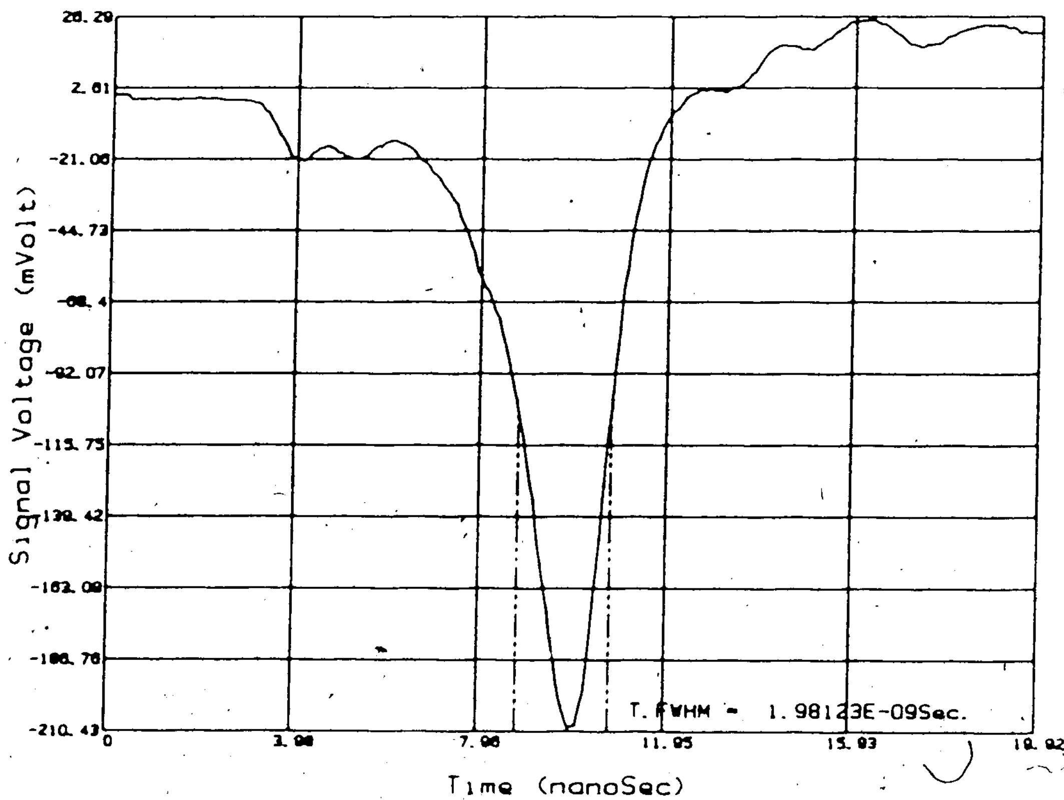


Fig.82 Optical time domain pulse at l=4432 m (line #8).

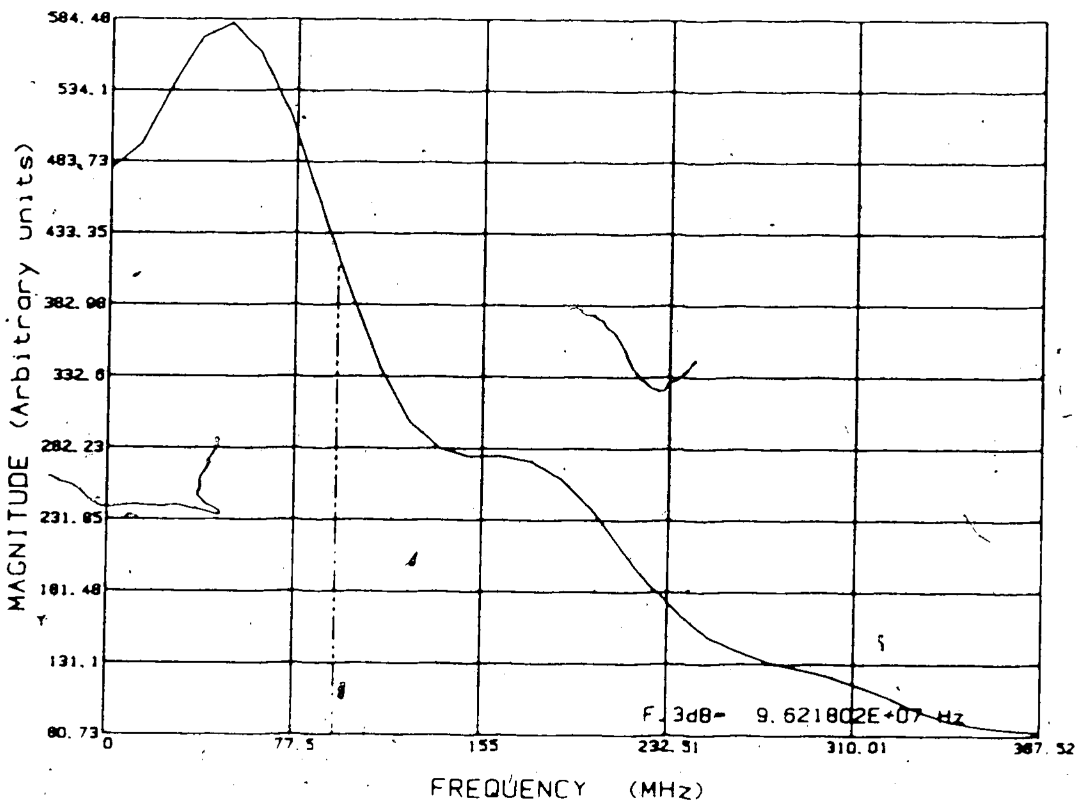


Fig.83 Computed fiber frequency response for l=4432 m (line #8).

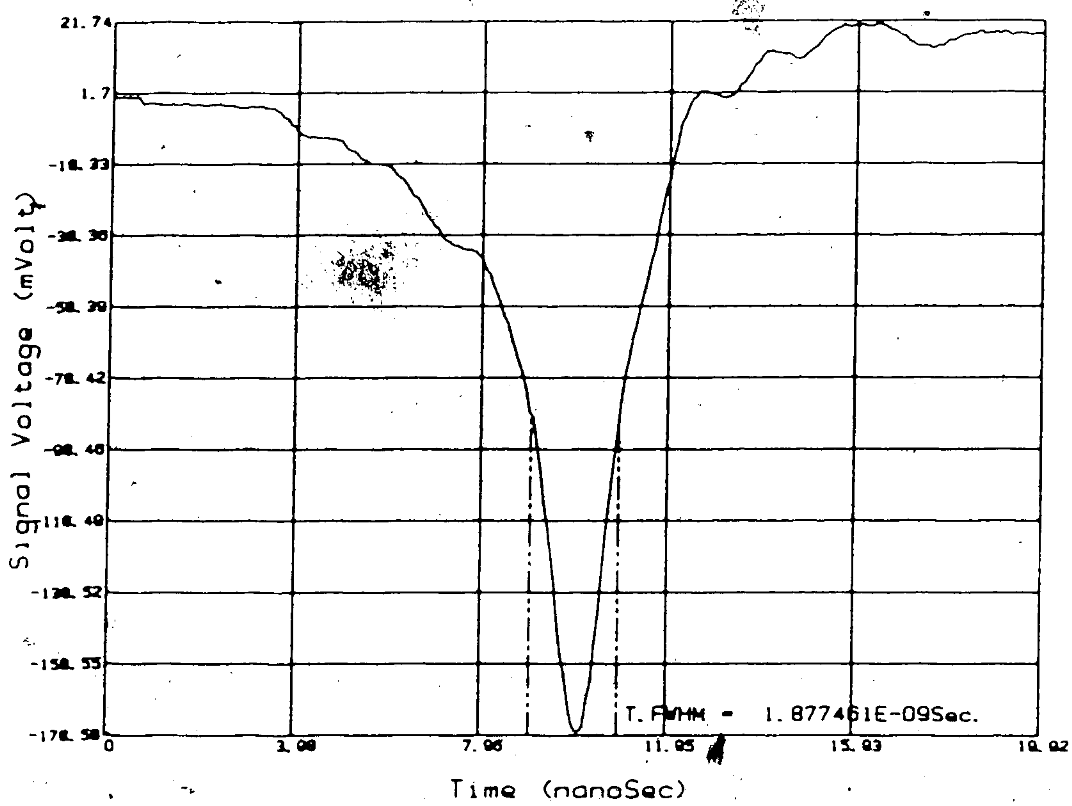


Fig.84 Optical time domain pulse at l=4432 m (line #9).

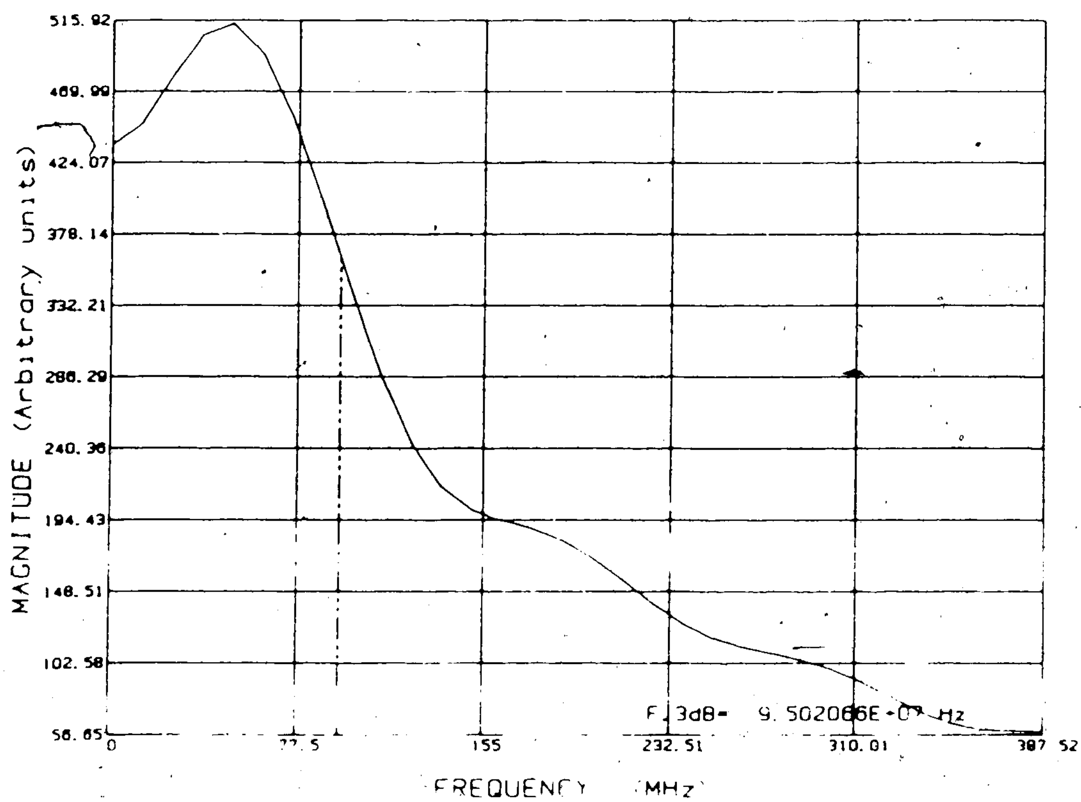


Fig.85 Computed fiber frequency response for l=4432 m (line #9).

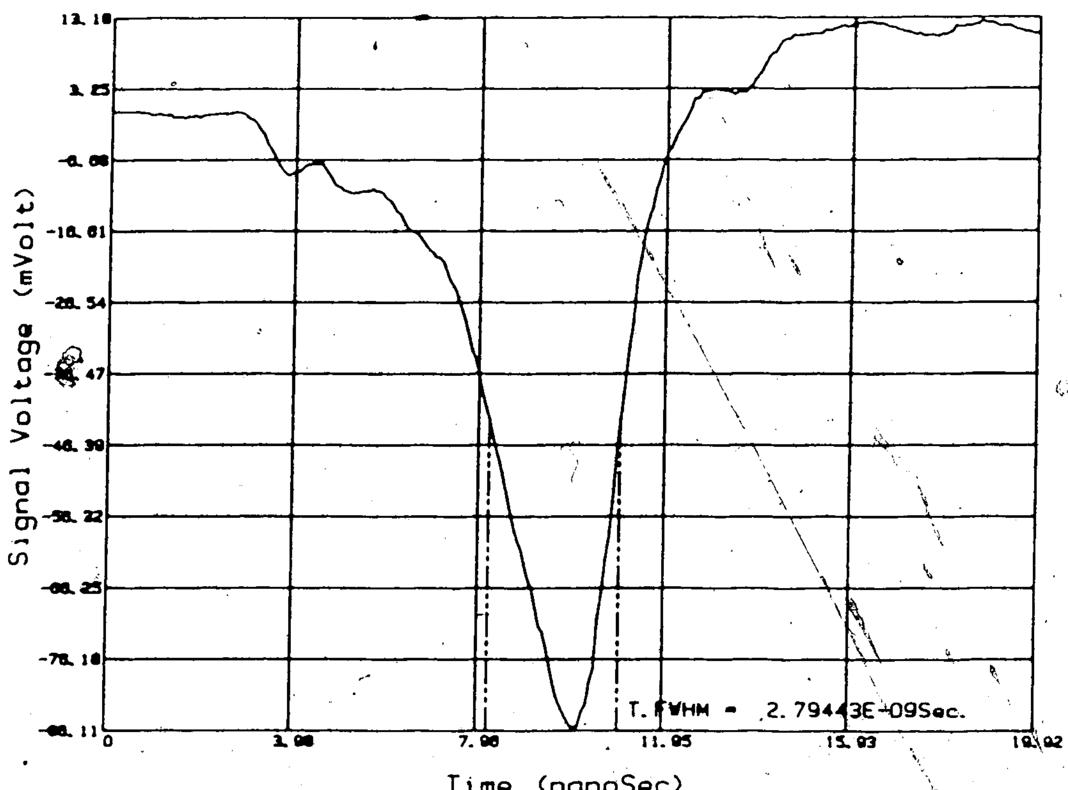


Fig.86 Optical time domain pulse at l=4432 m (line #12).

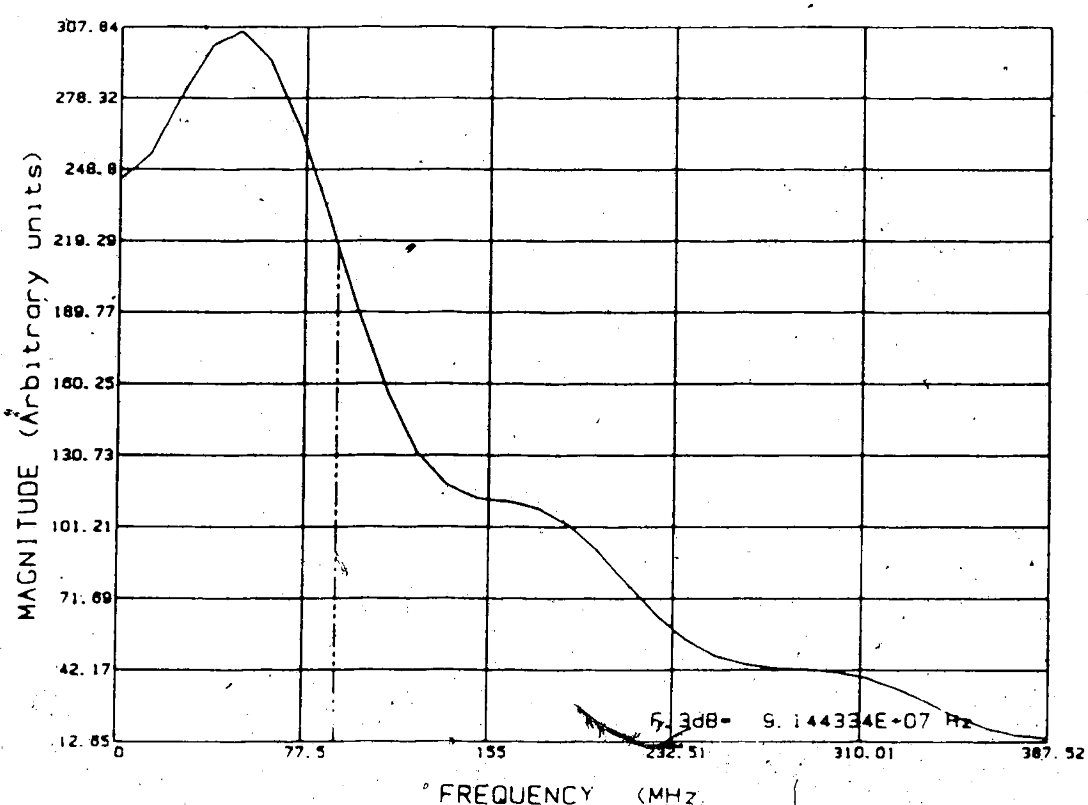


Fig.87 Computed fiber frequency response for l=4432 m (line #12).

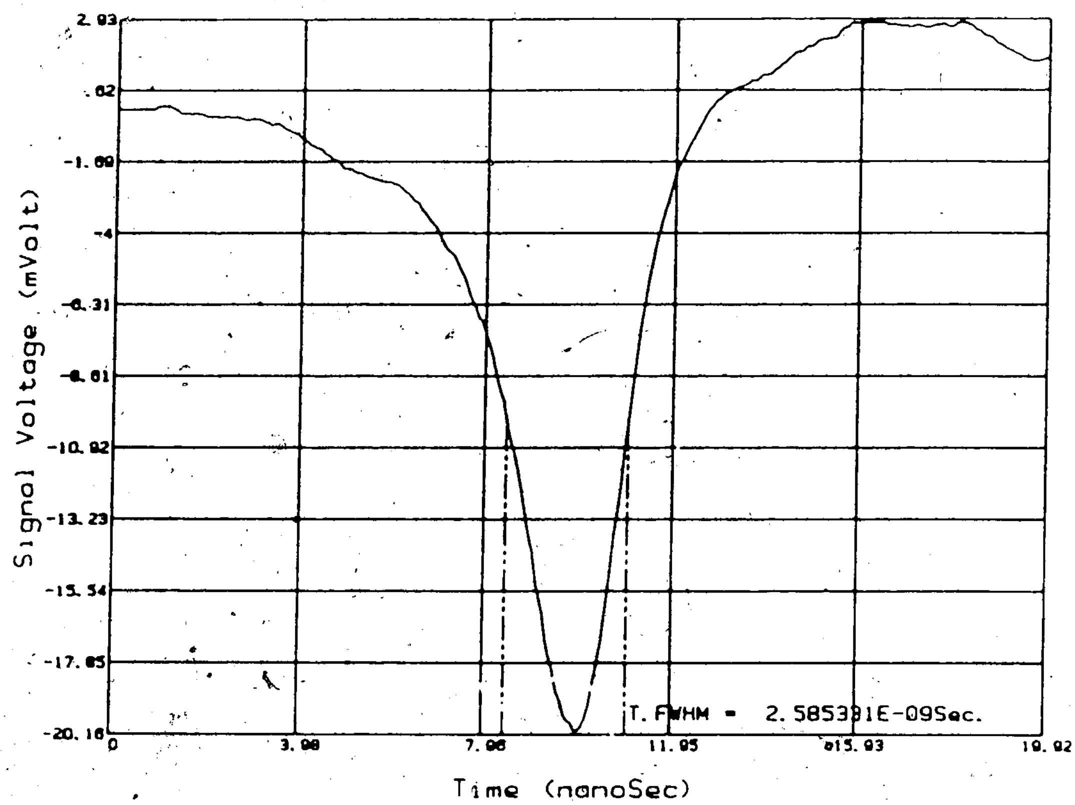


Fig.88 Optical time domain pulse at l=7826 m (line #1).

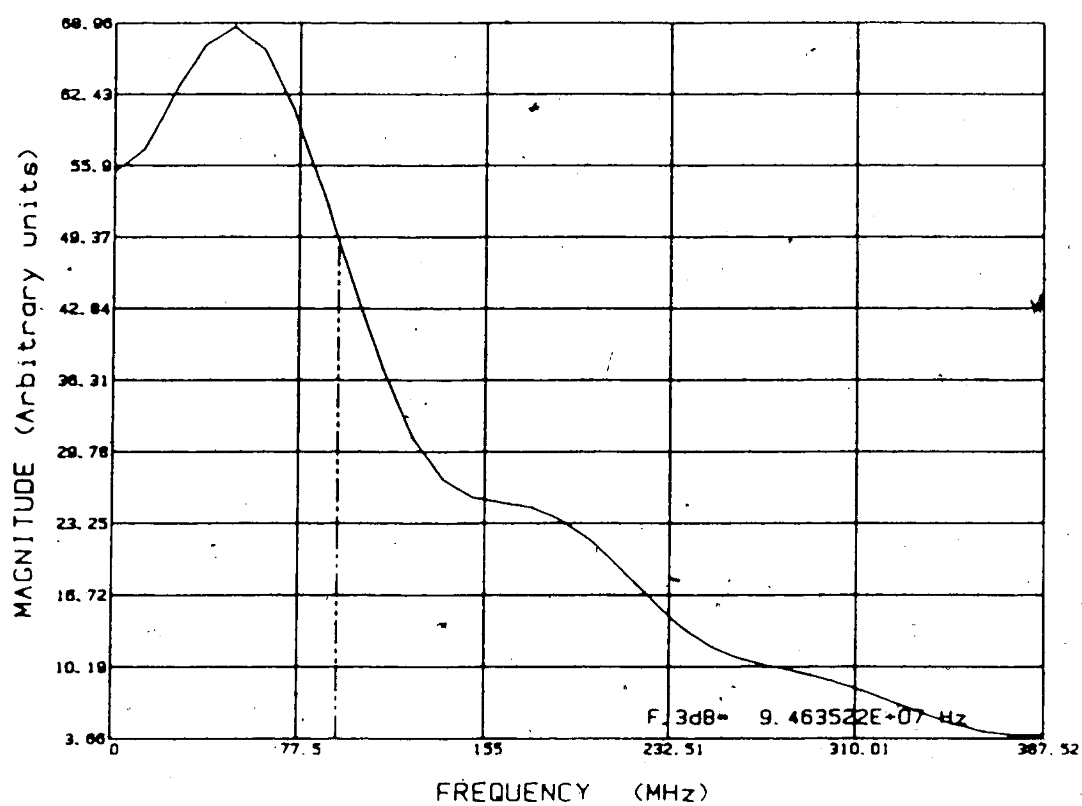


Fig.89 Computed fiber frequency response for l=7826 m (line #1).

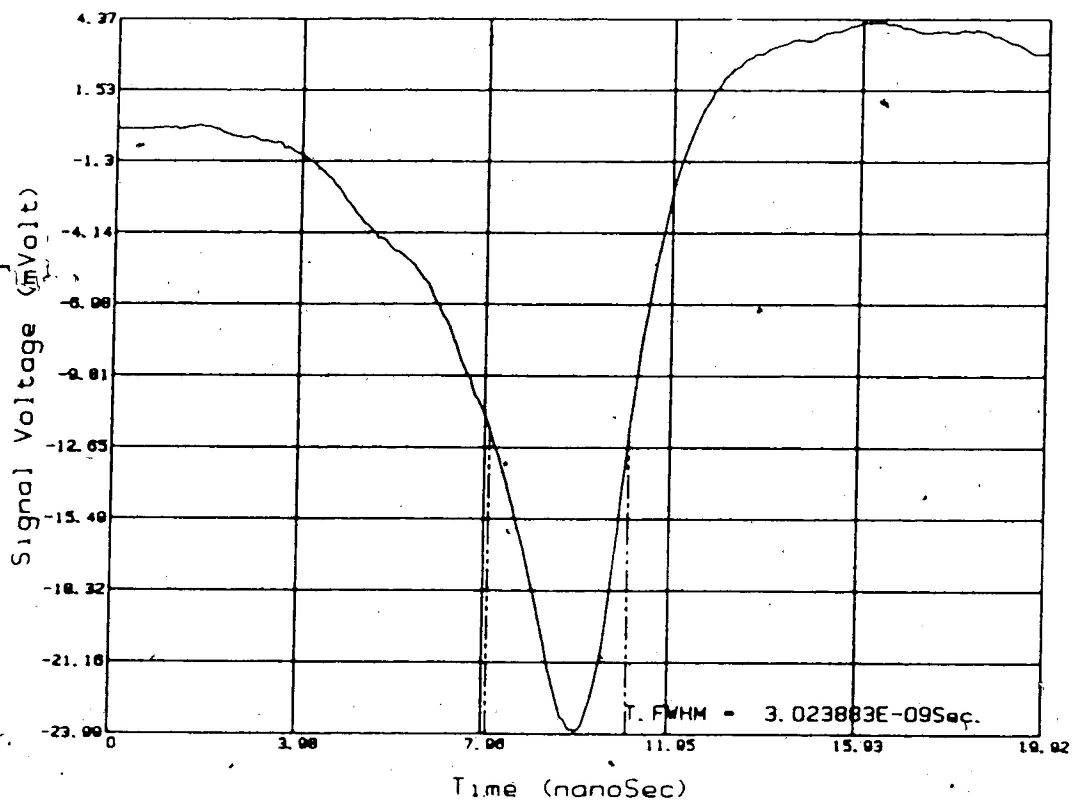


Fig.90 Optical time domain pulse at l=7826 m (line #2).

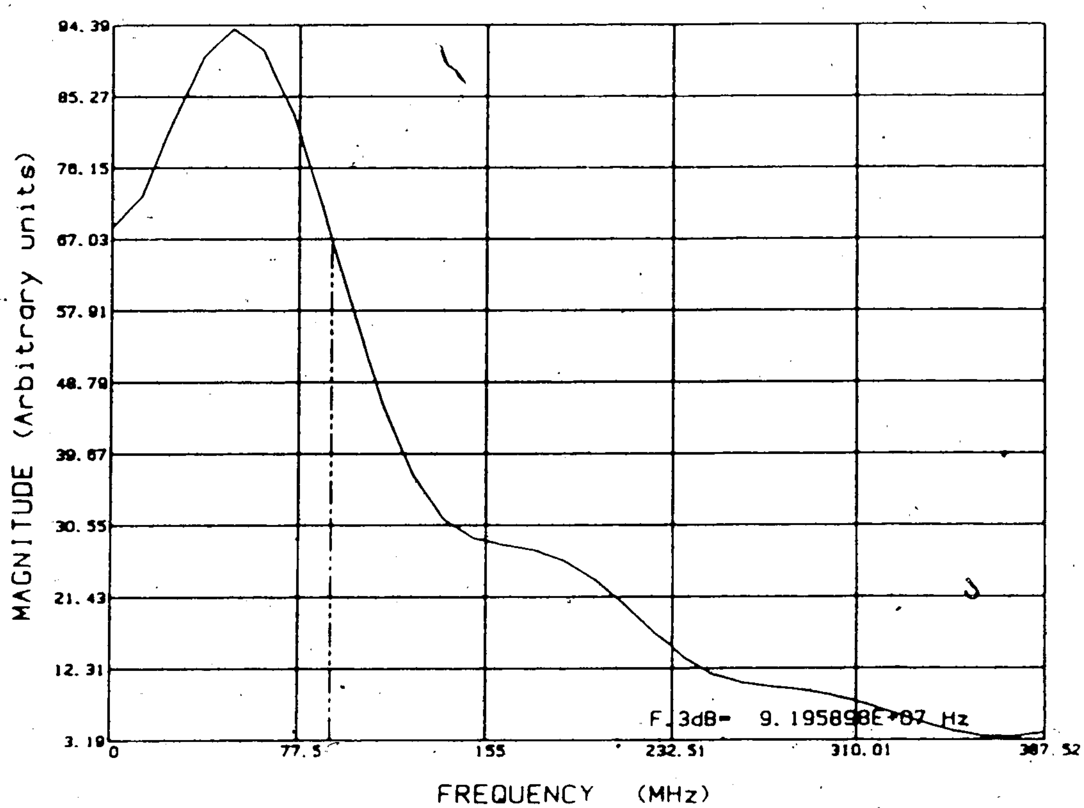


Fig.91 Computed fiber frequency response for l=7826 m (line #2).

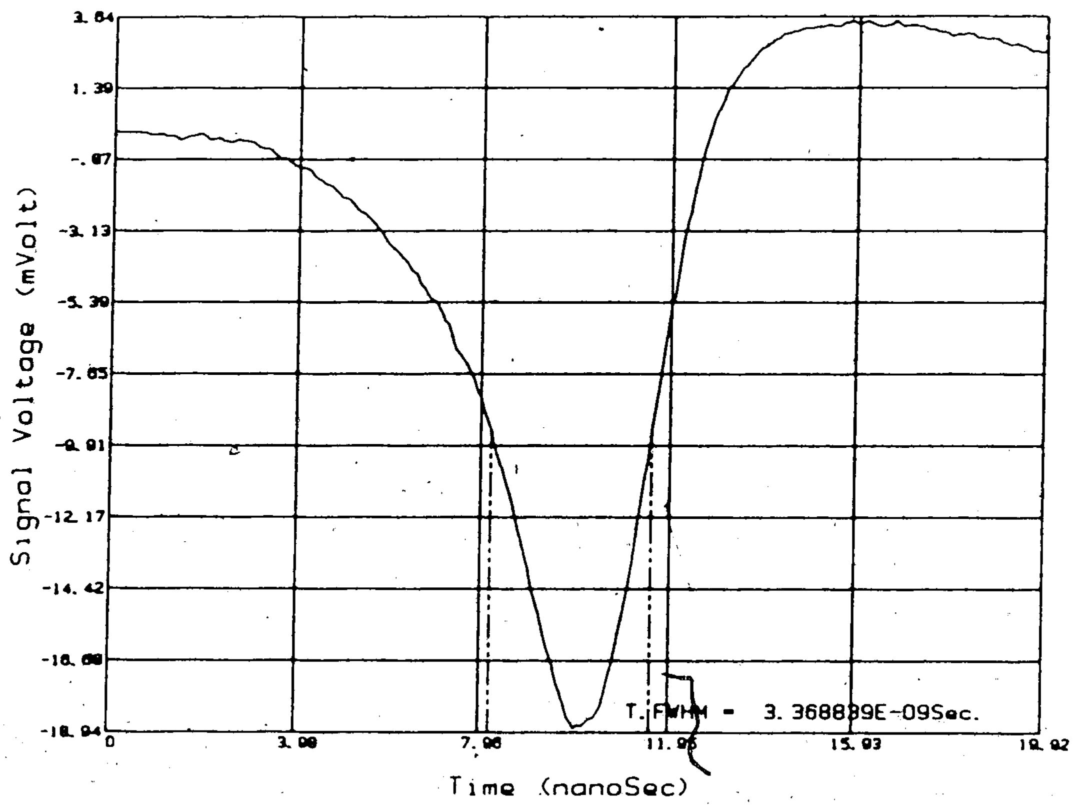


Fig.92 Optical time domain pulse at l=7826 m (line #3).

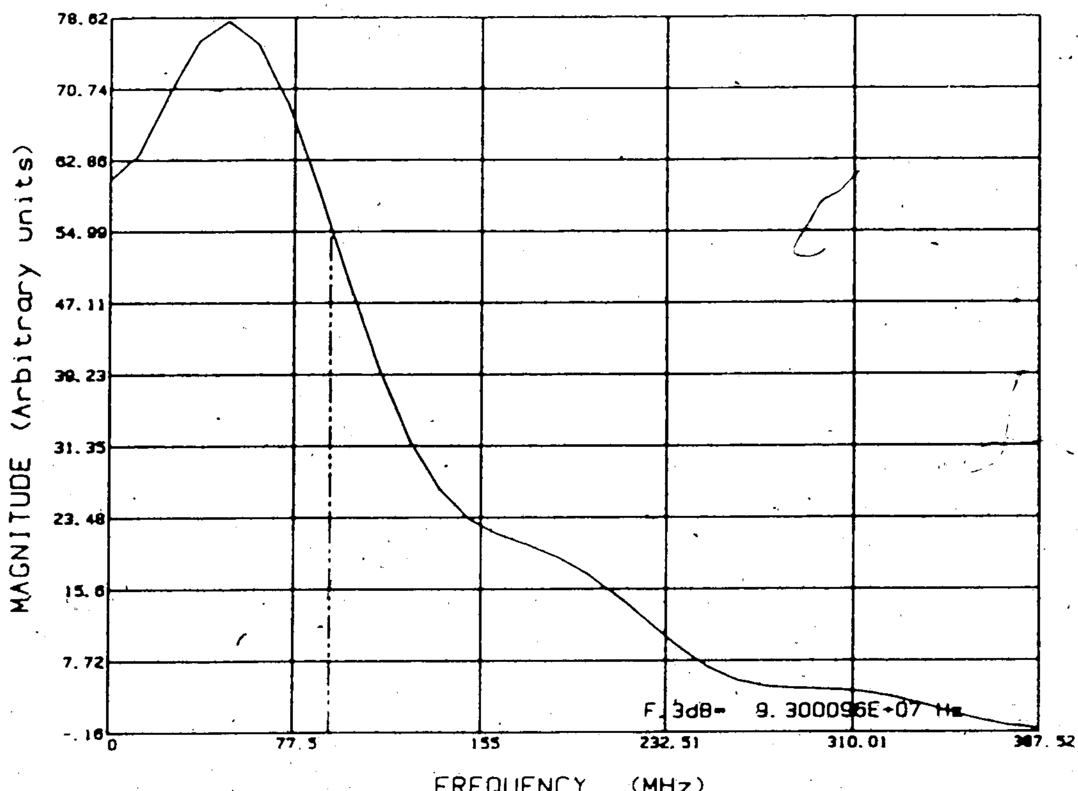


Fig.93 Computed fiber frequency response for l=7826 m (line #3).

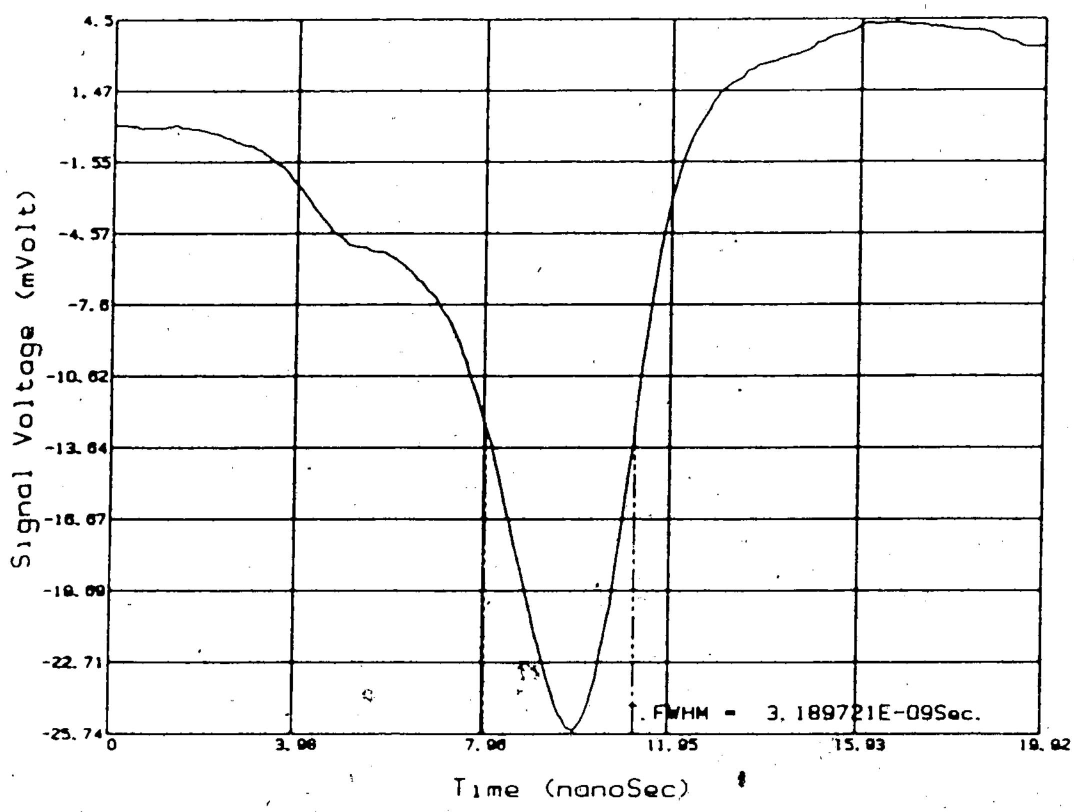


Fig.94 Optical time domain pulse at l=7826 m (line #4).

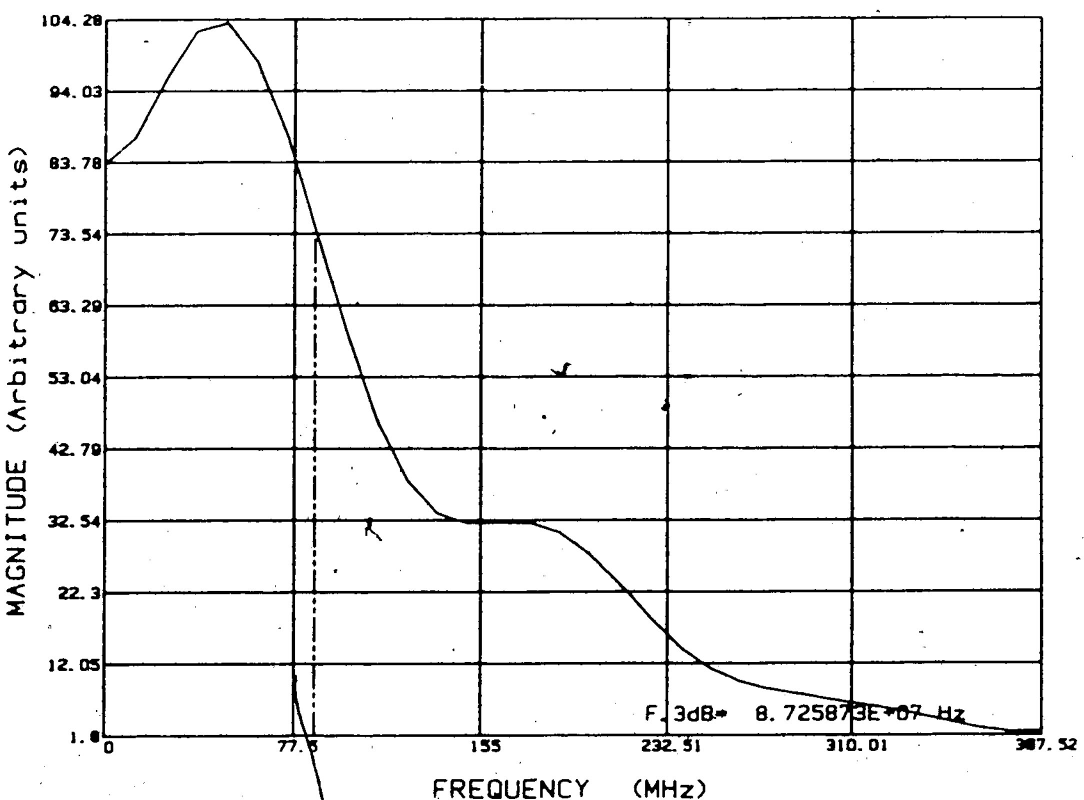


Fig.95 Computed fiber frequency response for l=7826 m (line #4).

6. DISCUSSION AND CONCLUSIONS

The results of the laboratory and field measurements are summarized in Tables 1 and 2, respectively.

Fiber length (m)	τ (nsec)	$f_{3\,dB}$ (MHz)
3000	4.23	85
4800	4.07	59
6400	4.72	62
9400	6.18	46

Table.1 Summary of the laboratory results.

Fiber length (m)	τ (nsec)	$f_{3\,dB}$ (MHz)
4432(#7)	2.10	97
4432(#8)	1.98	96
4432(#9)	1.88	93
4432(#12)	2.79	91
7826(#1)	2.58	95
7826(#2)	3.02	92
7826(#3)	3.37	93
7826(#4)	3.19	87

Table.2 Summary of the field results.

The laboratory results demonstrate the degradation of the roll-off frequency as the length of the fiber increases.

The calculated values of τ and f_{3dB} summarize the time- and frequency-domain waveform characteristics. For example, in the case of a pure Gaussian waveshape, it was demonstrated in section 4.4 that these two values can be related by the equation $(\tau)(f_{3dB})=0.3120$. When the signal deviates from a Gaussian, equations like this are not valid, and the values τ and f_{3dB} can lead to erroneous conclusions if they are taken as the sole parameters for the waveform analysed. For example, if the laboratory results for lengths of 4800 and 6400 m are considered, and if only the calculated values of τ and f_{3dB} are considered as the important parameters for these fibers, one could be led to think that the longer fiber has a better frequency response than the shorter one, since the latter has a lower value of f_{3dB}.

By looking at figs.74 and 76, the explanation for this discrepancy becomes evident. Depending on where the mathematical value of the -3 dB point happens to be (relative to the first sidelobe in the frequency spectrum) the f_{3dB} value can vary substantially with only slight amplitude changes. In fact, if these two frequency spectra are superimposed, they have a very similar shape.

Let us now consider the effects of these sidelobes. If the frequency spectrum shown in Fig.81 is examined, the frequency spacing between the main lobe and the sidelobes is approximately equal to 100 MHz. This spacing can be explained by considering that the windowed version of the waveform of Fig.80, sketched in Fig.96(a), can be approximated by the graphical summation of a pure Gaussian and a time-delayed rectangular pulse, as shown in Fig.96(b).

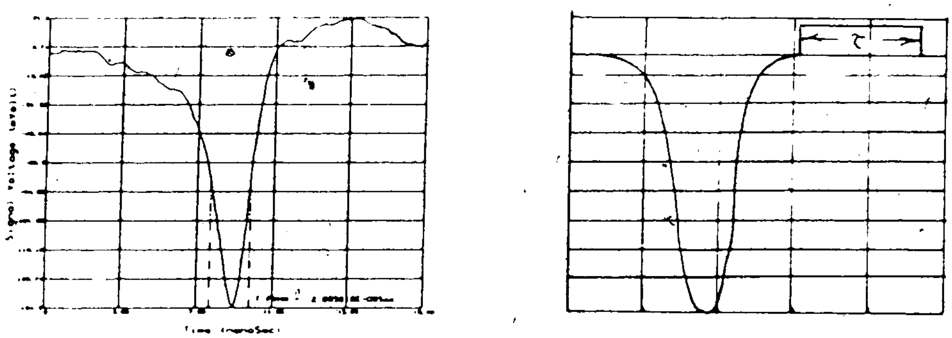


Fig.96 (a) Windowed version of Fig.80. (b) Pure Gaussian pulse added to a rectangular pulse.

The amplitude of the Fourier transform of the waveform shown in Fig.96(b) would be equal to the sum of the transforms of a Gaussian and of a rectangular pulse. The Gaussian frequency spectrum would cause the sum to decrease monotonically, and the frequency spectrum of the rectangular pulse would cause the sum to have sidelobes that would be separated by $1/2\tau$, τ being the width of the waveform. Thus, if τ has a value of 10 nsec as in this case, the frequency difference between sidelobes would be 1/2(10 nsec) = 100 MHz.

This sidelobe phenomenon is caused essentially by the presence of the step in the trailing edge of the time domain data. If the precision of the time domain frequency response method described in this thesis needs to be increased in the future, a better wideband amplifier would have to be designed in order to eliminate the perturbation in the trailing edge of the time domain pulses.

To conclude, a computer-controlled time-domain frequency response measurement system was developed that appears to have an accuracy of approximately 20% - the error being caused principally by the existing wideband amplifier. The method is useable for measurements on multimode optical fibers and may possibly be modified to operate with single mode fibers, since the f_{3dB} of the optical pulses is in excess of 1 GHz. In any case, the system has been used for the evaluation of the high frequency roll-off of many of the existing installed multimode optical fiber links in use by Edmonton Telephone Co. to carry 45 Mbit/sec signals. The measurements show that the fibers can nominally carry a higher bit rate but not much higher because the typical upper 3 dB frequency is about 90 MHz.

The author hopes that this work will stimulate others to continue research on this measurement technique because it does offer the attractive features of low cost, portable, fairly accurate measurements of multimode optical fiber frequency response.

LIST OF REFERENCES

[1] Andrews, J.R., "*Inexpensive Laser Diode Pulse Generator for Optical Waveguide Studies*", Rev. Sci. Instrum., Vol.45, pp.22-24. (1974).

[2] Huang, J.S.T., "*Study of Transistor Switching Circuit Stability in the Avalanche Region*", IEEE J. of Solid State, Vol.SC-2, pp.10-21. (1967).

[3] Andrews, J.R., "*Deflection Theory of Travelling Wave Oscilloscopes*", Ph.D. Dissertation, Electr. Eng. Dept., Univ. of Kansas, Lawrence. (1970).

[4] Pfeiffer, W., "*Ein einfacher impulsgenerator fuer reflexionsfaktor und sprung uebertragungsmessungen*", Inter. Elek. Rundschau, Vol.25, pp.268-272. (1972).

[5] Nicolson, A.M., Cronson, H.M., and Mitchell, P.G., "*Subnanosecond Risetime Pulse Generators*", IEEE Trans. on Instr. and Meas., Vol.IM-25, pp.104-107. (1976).

[6] Ortel, W.C.O., "*Monostable Tunnel Diode Trigger Circuit*", Proc. IEEE, Vol.54, pp.936-946. (1966).

[7] Cummings,A.J.,"*Tunnel Diode Pulse Generator for Liquid Helium Operation*", Rev. of Sci. Instr., Vol.36, pp.1503-1504. (1965)

[8] Andrews,J.R.,"*Improved Bias Supply for Tunnel Diode Picosecond Pulse Generator*", IEEE Trans. on Instr. and Meas., Vol.IM-19, pp.171-175. (1970).

[9] "*Coaxial Step Recovery Diode Modules (Impulse Train Generators)*", *Hewlett-Packard Components Note for Parts 33002A/B to 33005C/D.(1975)*

[10] *Schwarte,R.,"New Results of an Experimental Sampling System for Recording Fast Single Events"*, Electron. Lett., Vol.8, pp.94-96. (1972).

[11] Tielert,R.,"*Subnanosecond-Pulse Generator Employing 2-Stage Pulse Sharpener*",Electron. Lett.,Vol.12, pp.84-85. (1976).

[12] Millman,J.,and Taub,H.,*Pulse, Digital and Switching Waveforms*. New York: McGraw Hill, pp. 198-203, 508-513, 800-808. (1965).

[13] Hamilton,D.J.,Griffith,P.G.,and Shaver,F.H., "*Avalanche Transistor Circuits for Generating Rectangular Pulses*", Electron. Eng., Vol.34, pp.808-812.

(1962).

[14] Silver,S.L.,"*Avalanche Transistor Circuits*", Electron. World, Vol.77/78, pp.30-32. (1967).

[15] Magnuson,W.G.,"*Variable-Width Pulse Generation Using Avalanche Transistors*", IEEE Trans. on Instr. and Meas., Vol.IM-12, pp.56-64. (1963).

[16] Herden,W.B.,"*Application of Avalanche Transistors to Circuits With a Long Mean Time to Failure*", IEEE Trans. on Instr. and Meas., Vol.IM-25, pp.152-160. (1976).

[17] Zuhlke,R.,"*Erster und zweiter durchbruch bei transistoren*", Ph.D. Dissertation, Universitat Stuttgart, Germany. (1972).

[18] Rein,H.-M.,"*Subnanosecond-Pulse Generator With Variable Pulsewidth Using Avalanche Transistors*", Electron. Lett., Vol.11, pp.21-23. (1975).

[19] For a review, see for example, Paoli,T.L.,and Ripper,J.E., "*Direct Modulation of Semiconductor Lasers*", Proc. IEEE, Vol.58, pp.1457-1465. (1970).

[20] Roldan,R.,"*Spikes in the Light Output of Room Temperature GaAs Junction Lasers*", Appl. Phys. Lett., Vol.11, pp.346-348. (1967).

[21] Drozhbin,YU,A.,et al.,"*Generation of Ultra-Short Light Pulses With a GaAs Semiconductor Laser*", Sov.Phys.-JETP Lett., Vol.5, p.143. (1967).

[22] Ikegami,T.,Kobayashi,K.,and Suematsu,Y.,"*Transient Behaviour of Semiconductor Injection Lasers*", Electron. & Commun. Japan, Vol.53B. pp.82-89. (1972).

[23] Adams,M.J.,"*Rate Equations and Transient Phenomena in Semiconductor Lasers*", Opto-Electron., Vol.5, pp.201-215. (1973).

[24] Ripper,J.E.,and Dyment,J.C.,"*Internal Q-Switching in GaAs Junction Lasers*", Appl. Phys. Lett., Vol.12, pp.365-367. (1968).

[25] Dyment,J.C.,Ripper,J.E.,and Roldan,P.H.R.," *Spiking in Light Pulses from GaAs Q-Switched Junction Lasers*", IEEE J. of Quant. Electron., Vol.QE-5, pp.415-419. (1969).

[26] Gray,P.R.,and Meyer,R.G.,*Analysis and Design of Analog*

Integrated Circuits. New York: John Wiley & Sons, pp.46-59. (1984).

[27] Dyment,J.C.,and Ripper,J.E.,"*Temperature Behaviour of Stimulated Emission Delays in GaAs Diodes and a Proposed Trapping Model*", IEEE J. of Quant. Electron., Vol.QE-4, pp.155-160. (1968).

[28] Ripper,J.E.,"*Time Delays and Q-Switching in Junction Lasers: I - Theory*", IEEE J. of Quant. Electron., Vol.QE-5, pp.341-295. (1969).

[29] Ripper,J.E.,"*Time Delays and Q-Switching in Junction Lasers: II - Computer Calculations and Comparison With Experiments*", IEEE J. of Quant. Electron., Vol.QE-5, pp.396-403. (1969).

[30] Cooley,J.W.,and Tukey,J.W.,"*An Algorithm for the Machine Computation of Complex Fourier Series*", Math. of Comput., Vol.19, pp.297-301. (1965).

[31] Brigham,E.O.,and Morrow,R,E.,"*The Fast Fourier Transform*", IEEE Spectrum, Vol.4, pp.63-70. (1967).

[32] Cochran,W.T.,et al.,"*What is the Fast Fourier Transform ?*", IEEE Trans. on Audio and Electroacoust.,

Vol.AU-15, pp.45-55. (1967).

[33] Cooley,J.W.,Lewis,P.A.W.,and Welch,P.D.,"*The Fast Fourier Transform Algorithm and Its Applications*", IBM Research Paper RC-1743. (1967).

[34] Stanley,W.D.,Dougherty,G.R.,and Dougherty,R., *Digital Signal Processing*. Reston, Virginia: Reston Publishing Company, p.11, pp.262-280. (1984).

[35] Ramirez,R.W.,"*The Fast Fourier Transform's Errors are Predictable, Therefore Manageable*", Electronics, pp.96-102. (June 13, 1974).

APPENDIX 1

Data sheets for the MMT3904 and NE9002 transistors.

NEC

MICROWAVE GaAs POWER FET

NE9000, 1, 2 SERIES

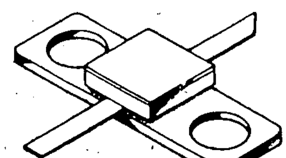
PRELIMINARY DATA SHEET

Ku-Band GaAs Power FET

FEATURES

- CLASS A OPERATION
- HIGH OUTPUT POWER
  P_{out} = 26.5dBm
  G_{1dB} = 7dB
- HIGH POWER ADDED EFFICIENCY

76 PACKAGE



Also available in 75 package. See page 3.

DESCRIPTION AND APPLICATIONS

The NE9000, 1, 2 is a 0.5 micron recessed gate GaAs power FET for commercial, military and space amplifier and oscillator applications to 20GHz. The series incorporates silicon nitride passivation for surface stabilization, and silicon dioxide glassivation for superior scratch resistance and mechanical protection. Four chip configurations are available. The NE900000 is a one cell die of 400μm gate width, the NE900100 is one cell of 750μm gate width, the NE900200 is a two cell die of 1500μm gate width and the NE900400 is a four cell of 3000μm gate width. The NE9004 and NE9008 (2 x 9004 chips) are covered in a separate data sheet. The series is available in chip form or a variety of hermetic ceramic packages. The NE900000 is a standard die the NE900100 and NE900200 are available as standard die or with wraparound source metallization. The NE900400 incorporates a plated heat sink and via hole source grounding for superior RF and thermal performance. The series conforms to MIL-S-19500 and is space qualified.

PERFORMANCE SPECIFICATIONS (T_a=25°C)

NE CHIP PART NUMBER / NE PACKAGE PART NUMBER / PACKAGE CODE[1]			NE900000 NE900075 NE900076 00(Chip),75,76		NE900100 NE900175 NE900176 00(Chip),75,76		NE900200 NE900275 NE900276 00(Chip),75,76	
SYMBOLS	PARAMETERS AND CONDITIONS	UNITS	MIN	TYP	MIN	TYP	MIN	TYP
P_{TEST}	Output Power at Test Point:							
	P_{in} = 12dBm, V_{DS} = 8V, I_D = 50mA, f = 14.5GHz	dBm	19.5	20.5				
	P_{in} = 15dBm, V_{DS} = 8V, I_D = 90mA, f = 14.5GHz	dBm			22	23		
	P_{in} = 19dBm, V_{DS} = 8V, I_D = 180mA, f = 14.5GHz	dBm					25.5	26.5
P_{1dB}	Output Power at 1dB Compression Point							
	V_{DS} = 8V, I_D = 50mA, f = 14.5GHz	dBm		20				
	V_{DS} = 8V, I_D = 90mA, f = 14.5GHz	dBm				23		
	V_{DS} = 8V, I_D = 180mA, f = 14.5GHz	dBm						25
G_{1dB}	Gain at 1dB Compression Point							
	V_{DS} = 8V, I_D = 50mA, f = 14.5GHz	dB		8				
	V_{DS} = 8V, I_D = 90mA, f = 14.5GHz	dB				7		
	V_{DS} = 8V, I_D = 180mA, f = 14.5GHz	dB						7
η_{add}	Power Added Efficiency V_{DS} = 8V, at P_{1dB} Conditions	%		27		27		26

SEE NOTES ON BACK PAGE

NEC Corporation

NE 9000 1, 2 SERIES Ku BAND GaAs POWER FET

ELECTRICAL CHARACTERISTICS (T_a = 25 C)

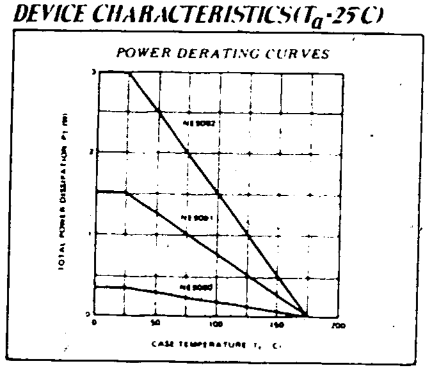
	NE CHIP PART NUMBER / NE PACKAGE PART NUMBER / PACKAGE CODE		NE900000 / NE900075 / NE900076 / 00 (Chip) 75 76			NE900100 / NE900175 / NE900176 / 00 (Chip) 75 76			NE900200 / NE900275 / NE900276 / 00 (Chip) 75 76		
SYMBOLS	PARAMETERS AND CONDITIONS	UNITS	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX
I_{DSS}	Saturated Drain Current at V_{DS} = 2.5V, V_{GS} = 0V	mA	80	120	[illegible]	150	225	300	300	450	600
V_P	Pinch off Voltage at V_{DS} = 2.5V, I_{DS} = 2.5mA	V	1.5	3.5	5						
	I_{DS} = 5.0mA	V				2.0	3.5	5			
	I_{DS} = 10.0mA	V							2.0	3.5	5
g_m	Transconductance at V_{DS} = 2.5V, I_{DS} = 50mA	m℧		25							
	I_{DS} = 90mA	m℧					50				
	I_{DS} = 180mA	m℧								100	
$R_{th(c-c)}$	Thermal Resistance	°C/W			180			100			50
P_T	Total Power Dissipation	W			0.8			1.5			3

SEE NOTES ON BACK PAGE

ABSOLUTE MAXIMUM RATINGS (T_a = 25 C)

SYMBOLS	PARAMETERS	UNITS	RATINGS
V_{DS}	Drain to Source Voltage	V	20
V_{GS}	Gate to Source Voltage	V	-9
I_D	Drain Current		
	NE900000, NE900075/76	mA	150
	NE900100, NE900175/76	mA	300
	NE900200, NE900275/76	mA	600
I_G	Gate Current		
	NE900000, NE900075/76	mA	1.3
	NE900100, NE900175/76	mA	2.6
	NE900200, NE900275/76	mA	5.0

DEVICE CHARACTERISTICS (T_a = 25 C)

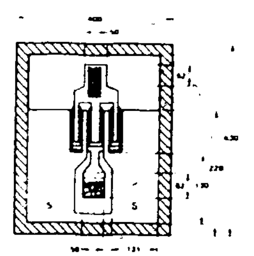


RELIABILITY SCREENING (HES 32752-03)

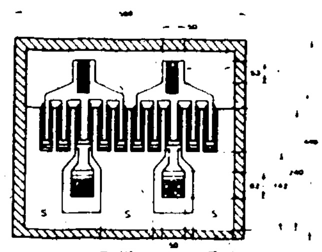
TEST	GRADE C Military Avionics	GRADE CX Military General	GRADE D Industrial
Precap Visual Inspection	100%	100%	
Vacuum Bake	100%	-	
High Temperature Storage	100%	100%	100%
Temperature Cycling	100%	100%	
Thermal Shock	100%		
Mechanical Shock (Y only)	100%		
Acceleration	100%	100%	
Gross Leak Test	100%	100%	100%
Fine Leak Test	100%	100%	100%
Area of Safe Operation (power only)	100%	100%	100%
High Temperature Reverse Bias (HTRB)	Optional	Optional	
Particle Impact Noise Detection (PIND)	Optional	Optional	
Electrical (DC) Tests	100%		
Power Burn in (168 hrs)	100%	100%	
Delta Calculation	Optional		
Group A Screening	100%	100*	100*
Group A Data	Optional		
External Visual	100%	100%	100%

PHYSICAL DIMENSIONS

NE900000 (Chip)
(Units in μm)



NE900100 NE900200 (Chip)
(Units in μm)



Die Thickness 110 - 160μm

Recommended Bonding Area

Plated Wraparound (optional)

Glassivated Area

DIE ATTACHMENT

Die attach can be accomplished with a Au-Ge (390-10°C) preforms in a forming gas environment. Epoxy die attach is not recommended.

BONDING

Gate and drain bonding wires should be minimum length, semi-hard gold wire (3-8% elongation) 30 microns or less in diameter.

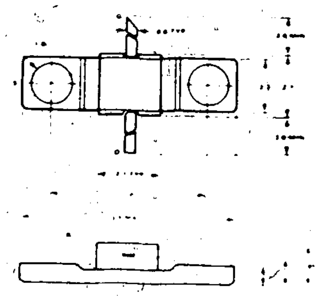
Bonding should be performed with a wedge tip that has a taper of approximately 15%. Die attach and bonding time should be kept to a minimum. As a general rule, the bonding operation should be kept within a 300°C - 10 minute curve. If longer periods are required, the temperature should be lowered.

PRECAUTIONS

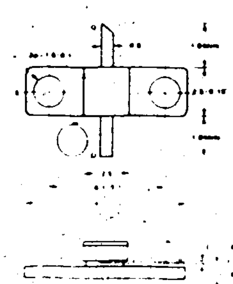
The user must operate in a clean, dry environment. The chip channel is glassivated for mechanical protection only and does not preclude the necessity of a clean environment.

The bonding equipment should be periodically checked for sources of surge voltage and should be properly grounded at all times. In fact, all test and handling equipment should be grounded to minimize the possibilities of static discharge.

PACKAGE CODE 75
(Units in mm)

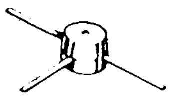


PACKAGE CODE 76
(Units in mm)



SEE NOTES ON BACK PAGE

MMT3903 (SILICON)
MMT3904



CASE 28(1)

NPN silicon micro-miniature annular transistors designed for general purpose switching and amplifier applications and for complementary circuitry with type MMT3905 and MMT3906 where high density packaging is required.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Collector-Emitter Voltage	V_{CEO}	40	Vdc
Collector-Base Voltage	V_{CB}	60	Vdc
Emitter-Base Voltage	V_{EB}	6.0	Vdc
Collector-Current — Continuous	I_C	200	mAdc
Total Device Dissipation @ T_A = 25°C Derate above 25°C	P_D	225 2.05	mW mW/°C
Operating & Storage Junction Temperature Range	T_J, T_{stg}	-55 to +135	°C

THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Ambient	θ_{JA}	0.490	°C/mW

FIGURE 1 – DELAY AND RISE TIME EQUIVALENT TEST CIRCUIT

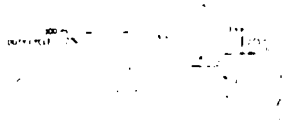
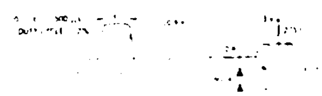


FIGURE 2 – STORAGE AND FALL TIME EQUIVALENT TEST CIRCUIT



ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
OFF CHARACTERISTICS					
Collector-Emitter Breakdown Voltage (1) (I_C = 1.0 mAdc, I_B = 0)	BV_{CEO}	40	-	-	Vdc
Collector-Base Breakdown Voltage (I_C = 10 µAdc, I_E = 0)	BV_{CBO}	60	-	-	Vdc
Emitter-Base Breakdown Voltage (I_E = 10 µAdc, I_C = 0)	BV_{EBO}	6.0	-	-	Vdc
Collector Cutoff Current (V_{CB} = 40 Vdc, I_E = 0)	I_{CBO}	-	-	50	nAdc
Emitter Cutoff Current ($V_{EB(off)}$ = 4.0 Vdc, I_C = 0)	I_{EBO}	-	-	50	nAdc
ON CHARACTERISTICS (1)					
DC Current Gain (I_C = 100 µAdc, V_{CE} = 1.0 Vdc) MMT3903 / MMT3904	h_{FE}	20 40	- -	- -	
(I_C = 1.0 mAdc, V_{CE} = 1.0 Vdc) MMT3903 / MMT3904		35 70	- -	- -	
(I_C = 10 mAdc, V_{CE} = 1.0 Vdc) MMT3903 / MMT3904		50 100	- -	150 300	
Collector-Emitter Saturation Voltage (I_C = 10 mAdc, I_B = 1.0 mAdc)	$V_{CE(sat)}$	-	-	0.2	Vdc
Base-Emitter Saturation Voltage (I_C = 10 mAdc, I_B = 1.0 mAdc)	$V_{BE(sat)}$	-	-	0.85	Vdc
SMALL-SIGNAL CHARACTERISTICS					
Current-Gain—Bandwidth Product (I_C = 10 mAdc, V_{CE} = 20 Vdc, f = 100 MHz) MMT3903 / MMT3904	f_T	250 300	- -	- -	MHz
Output Capacitance (V_{CB} = 5.0 Vdc, I_E = 0, f = 100 kHz)	C_{ob}	-	-	4.0	pF
Input Capacitance (V_{BE} = 0.5 Vdc, I_C = 0, f = 100 kHz)	C_{ib}	-	-	8.0	pF
Input Impedance (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	h_{ie}	-	1.0	-	k ohm
Voltage Feedback Ratio (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	h_{re}	-	2.0	-	X 10^{-4}
Small-Signal Current Gain (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) MMT3903 / MMT3904	h_{fe}	- -	100 200	- -	-
Output Admittance (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	h_{oe}	-	10	-	µmhos
Noise Figure (I_C = 100 µAdc, V_{CE} = 5.0 Vdc, R_S = 1.0 k ohms, Noise Bandwidth = f = 10 Hz to 15.7 kHz)	NF	-	3.0	-	dB
SWITCHING CHARACTERISTICS					
Delay Time (V_{CC} = 3.0 Vdc, $V_{BE(off)}$ = 0.5 Vdc,	t_d	-	24	-	ns
Rise Time I_C = 10 mAdc, I_{B1} = 1.0 mAdc)	t_r	-	13	-	ns
Storage Time (V_{CC} = 3.0 Vdc, I_C = 10 mAdc,	t_s	-	125	-	ns
Fall Time I_{B1} = I_{B2} = 1.0 mAdc)	t_f	-	11	-	ns

(1) Pulse Test: Pulse Width = 300 µs, Duty Cycle = 2.0%.
For characteristic curves, see 2N3903, 2N3904 Data.

APPENDIX 2

Listing of the data transfer program *scope*.

Listings of the SCOPE program

```
0 REM
1 GOSUB 1900 : REM Initialize arrays
2 GOSUB 100  : REM Initialize the GPIB bus
3 GOSUB 7000 : REM LOAD STRINGS
10 GOTO 3700 : REM run menu
99 END
100 REM Initialize the GPIB
110 MY.ADDR% = 0
120 BD.ADDR% = 784
130 OSC.ADDR% = 1 : OSC$ = "1" : REM 7854 GPIB address
160 PARAM$ = "INIT/0/784/P/" : REM P indicates polling
170 GO[illegible]60
175 PARAM$ = "SDR/1/" : GOSUB 9160 : REM Set up oscilloscope as a remote devic
190 RETURN
200 REM execute scope commands remotely
210 PRINT : INPUT "Yes m'Lord ";DATA.STRING$
220 WHILE (DATA.STRING$ <> "")
230 PARAM$ = "WR.STR/1//EOI/" : GOSUB 9160
240 PRINT : INPUT "Yes m'Lord ";DATA.STRING$
250 WEND
290 RETURN
1500 REM Load a waveform from the scope
1505 PRINT : INPUT "Load WFM # ";WFM$
1507 PRINT : PRINT "Store as waveform # ( 0 -";MAX.WAVE%;") "; : INPUT WAVE%
1510 PARAM$ = "WR.STR/1//EOI/" : DATA.STRING$ = WFM$ + " WFM SEND%" : GOSUB 9160

1520 PARAM$ = "UNLISTEN/" : GOSUB 9160
1530 PARAM$ = "TALK/1/" : GOSUB 9160
1540 PARAM$ = "MLA/" : GOSUB 9160
1550 PARAM$ = "GTSB/" : GOSUB 9160
1560 D$ = ""
1562 WHILE ( (RIGHT$(D$,5)) <> "CURVE" )
1570 GOSUB 1693 : REM get a letter
1580 D$ = D$ + L$
1590 WEND
1595 GOSUB 2200 : REM decipher waveform preamble
1600 REM start reading in numeric data
1605 I% = 0 : REM i% is pointer to current datum
1610 GOSUB 1693
1619 REM
1620 WHILE ( (NOT(NUMERIC%)) AND (LAST% <> TRUE%) ) : GOSUB 1693 : WEND
1622 IF (LAST% = TRUE%) THEN GOTO 1655
1625 N$ = "" : REM n$ is string representation of number
1630 WHILE ( NUMERIC% AND (LAST% <>TRUE%) ) : N$ = N$ + L$ : GOSUB 1693 : WEND
1635 IF (LAST% = TRUE%) THEN N$ = N$ + L$
1640 D(WAVE%,I%) = (VAL(N$)) * YMULT + YZERO
1645 I% = I% + 1
1650 IF (LAST% <> TRUE%) THEN GOTO 1619
1655 REM
1657 NR.PT%(WAVE%) = I% - 1 : REM nr.pt% is the number of points of data less 1
1660 PARAM$ = "TCSY/" : GOSUB 9160
1685 REM
1687 PRINT : PRINT BELL$ + "DONE" : FOR P = 1 TO 200 : NEXT P
1690 RETURN
1692 END
1693 REM Get a letter and return in l$ .
1694 PARAM$ = "RD.BYTE/" : GOSUB 9160
1695 IF ( (LAST% <> TRUE%) [illegible] DAT% <> 256) ) THEN L$ = CHR$(DAT%) ELSE L$ = ""
1696 NUMERIC% = ( ((L$ <= "9") AND (L$ >= "0")) OR (L$ = ".") OR (L$ = "-") OR (
L$ = "E") )
1697 RETURN
1700 REM Send a mountain of data to scope
1705 PRINT : PRINT "Send waveform # ( 0 -";MAX.WAVE%;") "; : INPUT WAVE%
1707 PRINT : INPUT "Store on scope as WFM # ";WFM$
1708 GOSUB 3190 : REM GET VALUES FOR YMULT, YZERO
1709 DATA.STRING$ = "0 WFM 1 >WFM" : PARAM$ = "WR.STR/1//EOI/" : GOSUB 9160
1710 DATA.STRING$ = "READ%" : PARAM$ = "WR.STR/1//EOI/" : GOSUB 9160
1720 PARAM$ = "UNLISTEN/" : GOSUB 9160
1730 PARAM$ = "LISTEN/1/" : GOSUB 9160
1740 PARAM$ = "MTA/" : GOSUB 9160
1750 PARAM$ = "GTSB/" : GOSUB 9160
1760 D$ = "WFMPRE ENCDG:ASC,NR.PT:" + STR$(NR.PT%(WAVE%) + 1) + ",PT.FMT:Y,XZERO
0,XINCR:" + STR$(XINCR(WAVE%)) + ",XUNIT:S,YZERO:" + STR$(YZERO) + ",YMULT:" +
STR$(YMULT) + ",YUNIT:V;" + CHR$(13) + "CURVE "
1765 FOR I% = 1 TO LEN(D$) : DAT% = ASC(MID$(D$,I%,1)) : PARAM$ = "WR.BYTE/" : G
OSUB 9160 : NEXT I%
1770 FOR I% = 0 TO NR.PT%(WAVE%)
1775 D.SEND = (D(WAVE%,I%) - YZERO) / YMULT
1777 D.SEND = ( INT( D.SEND * 10000 ) ) / 10000
1780 N$ = STR$(D.SEND) : J% = LEN(N$) : IF ( (LEFT$(N$,1)) = " " ) THEN N$ = RIG
HT$(N$,(J%-1))
1790 J% = LEN(N$) : IF ( (RIGHT$(N$,1)) = " " ) THEN N$ = LEFT$(N$,(J%-1))
1795 IF (I% <> 0) THEN N$ = "," + N$
1800 FOR J% = 1 TO LEN(N$) : DAT% = ASC(MID$(N$,J%,1)) : PARAM$ = "WR.BYTE/" : G
OSUB 9160 : NEXT J%
1810 NEXT I%
1820 LAST% = TRUE% : PARAM$ = "SET.EOI/" : GOSUB 9160
1830 DAT% = 13 : PARAM$ = "WR.BYTE/" : GOSUB 9160
1880 PARAM$ = "TCSY/" : GOSUB 9160
1885 DATA.STRING$ = "0 WFM "+WFM$+" >WFM.1 WFM 0 >WFM "+WFM$+" WFM" : PARAM$ = "
WR.STR/1//EOI/" : GOSUB 9160
1887 PRINT : PRINT BELL$ + "DONE" : FOR P = 1 TO 200 : NEXT
1890 RETURN
1900 REM initialize arrays
1901 SCREEN 0
1902 RESCUE% = 0 : REM rescue% indicates when an error has been trapped
1905 ON ERROR GOTO 4000 : REM set up selected error trapping
1910 MAX.WAVE% = 3 : REM total number of waves that can be stored at once
1920 MAX.PT% = 300 : REM maximum number of points per wave
1930 DIM D(MAX.WAVE%,MAX.PT%)
1940 DIM NR.PT%(MAX.WAVE%)
1950 DIM XINCR(MAX.WAVE%)
1955 DIM Y(MAX.PT%)
1960 WAVE% = 0
1965 BELL$ = CHR$(7)
1970 DIM CH.INDEX%(26) : DIM CH$(13) : REM for drawing chars
1990 RETURN
2000 REM wait for a keystroke and return it in A$
2010 A$ = INKEY$
2020 WHILE A$ = ""
2030 A$ = INKEY$
2040 WEND
2050 YES% = ( (A$ = "Y") OR (A$ = "y") )
2090 RETURN
2100 REM Send back a number in n pointed at in d$ by i%
2110 J% = I% + 1 : L$ = MID$(D$,J%,1)
2120 NUMERIC% = ( ((L$ <= "9") AND (L$ >= "0")) OR (L$ = ".") OR (L$ = "-") OR (
L$ = "E") )
2130 WHILE NUMERIC% : J% = J% + 1 : L$ = MID$(D$,J%,1)
2140 NUMERIC% = ( ((L$ <= "9") AND (L$ >= "0")) OR (L$ = ".") OR (L$ = "-") OR (
L$ = "E") )
2150 WEND
2160 N = VAL(MID$(D$,I%,(J%-I%)))
2190 RETURN
```

```
2200 REM decipher the waveform preamble
2210 I% = 1
2220 WHILE ( (MID$(D$,I%,6)) <> "XINCR:" ) : I% = I% + 1 : WEND
2225 I% = I% + 6 : GOSUB 2100 : REM n = XINCR
2230 XINCR(WAVE%) = N
2240 WHILE ( (MID$(D$,I%,6)) <> "YZERO:" ) : I% = I% + 1 : WEND
2250 I% = I% + 6 : GOSUB 2100 : YZERO = N
2260 WHILE ( (MID$(D$,I%,6)) <> "YMULT:" ) : I% = I% + 1 : WEND
2265 I% = I% + 6 : GOSUB 2100 : YMULT = N
2290 RETURN
2300 REM Execute an FFT on a waveform
2310 PRINT : PRINT "Execute FFT on waveform # ( 0 -";MAX.WAVE%;") "; : INPUT WAV
E%
2320 PRINT "Do you need to type the values of waveform # ";WAVE%;" (Y/N) ?"
2330 A$=INKEY$:IF A$="" GOTO 2330
2340 IF (A$="Y") OR (A$="N") THEN GOTO 2360
2350 PRINT"CAN YOU TYPE 'Y' OR 'N' PLEASE ?":GOTO 2330
2360 IF A$="Y" GOTO 2420
2370 FOR I% = 0 TO NR.PT%(WAVE%)
2380 Y(I%+1)=D(WAVE%,I%)
2390 Y.FACTOR=1
2400 NEXT I%
2410 GOTO 2540
2420 CLS
2430 PRINT : PRINT "  Number of points: ";NR.PT%(WAVE%) + 1
2440 PRINT : PRINT "  XINCR: ";XINCR(WAVE%)
2450 PRINT : PRINT "Numeric data:"
2460 FOR I% = 0 TO NR.PT%(WAVE%)
2470 Y(I%+1)=D(WAVE%,I%)
2480 PRINT "Y(";I%+1;")=";Y(I%+1),
2490 IF (CSRLIN < 22) THEN GOTO 2520
2500 PRINT : PRINT "Hit any key to continue"; : GOSUB 2000
2510 CLS
2520 Y.FACTOR=1
2530 NEXT I%
2540 NRPT=NR.PT%(WAVE%)+1:DIM PLOT(NRPT+4):DC=0:CLS:PRINT"PROGRAM IN ACTION...BE
 PATIENT ''''
2550 FOR I=1 TO 20
2560 DC=DC+Y(I)
2570 NEXT I
2580 DC=DC/20
2590 FOR I=1 TO NRPT
2600 Y(I)=Y(I)-DC
2610 NEXT I
2620 INC=XINCR(WAVE%)
2630 CHAIN"WINDOW.BLJ",,ALL
2660 PRINT"DO YOU NEED TO PLOT THE ";NRPT;" POINTS WINDOWED WAVEFORM (Y/N) ?"
2670 A$=INKEY$:IF A$="" GOTO 2670
2680 IF (A$="Y") OR (A$="N") THEN GOTO 2700
2690 PRINT"CAN YOU TYPE 'Y' OR 'N' PLEASE ?:GOTO 2362
2700 RETURN.NO=6
2710 IF A$="N" GOTO 2760
2720 FOR I=1 TO NRPT
2730 PLOT(I)=Y(I)
2740 NEXT I
2750 CHAIN"PLOT.BLJ",,ALL
2760 COMMON Y(),INC,CH,INDEX%(),CH$(),NRPT,PLOT(),Y.FACTOR
2770 CHAIN"ZOOMFFT.BLJ"
3000 REM plot the waveform
3010 PRINT : PRINT "Plot waveform # ( 0 -";MAX.WAVE%;") "; : INPUT WAVE%
3020 XMINP = 0 : XMAXP = 639 : YMINP = 0 : YMAXP = 199 : MARGIN = 10
3030 GOSUB 3190 : YZ = YZERO : REM YMIN = -4 : YMAX = 4 : YZ = 0
3040 PRINT : PRINT "Plot waveform # ( 0 -";MAX.WAVE%;") "; : INPUT WAVE%
3050 XSC = (XMAXP - XMINP - (2 * MARGIN)) / NR.PT%(WAVE%)
3060 YOFF = ( (YMAX - YZ) / YRANGE ) * ( YMAXP - YMINP - (2 * MARGIN) ) + MARGIN
3070 YSC = -1 * ( YMAXP - YMINP - (2 * MARGIN) ) / YRANGE
3080 REM plot
3090 REM put on lines
3100 SCREEN 2 : KEY OFF
3110 CLS
3120 LINE (XOFF,YMINP)-(XOFF,YMAXP) : LINE (XMINP,YOFF)-(XMAXP,YOFF)
3130 REM plot the waveform
3140 XP = XOFF : YP = ( D(WAVE%,0) - YZ ) * YSC + YOFF : PSET (XP,YP)
3150 FOR I% = 1 TO NR.PT%(WAVE%) : XP = I% * XSC + XOFF : YP = ( D(WAVE%,I%) - Y
Z ) * YSC + YOFF : LINE -(XP,YP) : NEXT I%
3160 GOSUB 2000 : REM wait for keypress
3170 KEY ON : SCREEN 0
3180 RETURN
3190 REM determine YMULT & YZERO
3200 YMAX = -1.701412E+38 : YMIN = 1.701412E+38
3210 FOR I% =0 TO NR.PT%(WAVE%)
3220 IF (D(WAVE%,I%) < YMIN) THEN YMIN = D(WAVE%,I%)
3230 IF (D(WAVE%,I%) > YMAX) THEN YMAX = D(WAVE%,I%)
3240 NEXT I%
3250 YRANGE = YMAX - YMIN
3260 YMULT = YRANGE / 7
3270 YZERO = YMAX - (YRANGE / 2)
3280 RETURN
3300 REM Save waveform on disk
3310 PRINT : PRINT "Save waveform # ( 0 -";MAX.WAVE%;") "; : INPUT WAVE%
3320 PRINT : INPUT "Please enter name of file of waveform to save ";NF$
3330 IF ( LEN(NF$) > 8 ) THEN PRINT : PRINT BELL$;"Too many characters (8 maximu
m)" : GOTO 3320
3340 NF$ = NF$ + ".WFM"
3350 OPEN NF$ FOR OUTPUT AS #1
3360 IF RESCUE% THEN RESCUE% = 0 : PRINT : PRINT BELL$;"File does not exist" : P
RINT : PRINT "Hit any key to continue" : GOSUB 2000 : GOTO 3420
3370 PRINT #1,NR.PT%(WAVE%) : PRINT #1,XINCR(WAVE%)
3380 FOR I% = 0 TO NR.PT%(WAVE%)
3390 PRINT #1,D(WAVE%,I%)
3400 NEXT I%
3410 CLOSE #1
3420 REM
3430 RETURN
3500 REM Retrieve waveform from disk
3510 PRINT : PRINT "Do you wish to see the catalog of waveforms ? "; : GOSUB 200
0 : REM get a keypress
3520 IF YES% THEN PRINT : PRINT : FILES "*.wfm"
3530 PRINT : INPUT "Please enter name of file of waveform to retrieve ";NF$
3540 IF (NF$ = "") THEN GOTO 3650
3550 IF ( ( (RIGHT$(NF$,4)) <> ".WFM" ) AND ( (RIGHT$(NF$,4)) <> ".wfm" ) ) THEN
 NF$ = NF$ + ".WFM"
3560 PRINT : PRINT "Store waveform as # ( 0 -";MAX.WAVE%;") "; : INPUT WAVE%
3562 IF WAVE%=1 THEN NAME1$=NF$
3564 IF WAVE%=2 THEN NAME2$=NF$
3566 IF WAVE%=3 THEN NAME3$=NF$
3568 IF WAVE%=0 THEN NAME0$=NF$
3570 OPEN NF$ FOR INPUT AS #1
3580 IF RESCUE% THEN RESCUE% = 0 : I% = LEN(NF$) : NF$ = LEFT$(NF$,(I%-4)) : OPE
N NF$ FOR INPUT AS #1
3590 IF RESCUE% THEN RESCUE% = 0 : PRINT : PRINT BELL$;"File does not exist" : P
RINT : PRINT "Hit any key to continue" : GOSUB 2000 : GOTO 3650
3600 INPUT #1,NR.PT%(WAVE%) : INPUT #1,XINCR(WAVE%)
3610 FOR I% = 0 TO NR.PT%(WAVE%)
3620 INPUT #1,D(WAVE%,I%)
3630 NEXT I%
3640 CLOSE #1
3650 REM
3660 RETURN
3700 REM Menu
3710 CLS
```

```
3720 MAXCHOICE%=7
3730 PRINT "FUNCTIONS:" : PRINT
3740 PRINT " 0.  Quit"
3750 PRINT " 1.  Load a waveform from the scope"
3760 PRINT " 2.  Dump a waveform to the scope"
3770 PRINT " 3.  Execute functions on scope"
3780 PRINT " 4.  Execute an FFT on a waveform"
3790 PRINT " 5.  Plot the waveform on the screen"
3800 PRINT " 6.  Save waveform on disk"
3810 PRINT " 7.  Retrieve waveform from disk"
3820 PRINT : PRINT "Please enter a number (0 -";MAXCHOICE%;") "; : INPUT CHOICE

3830 CLS
3840 IF ( (CHOICE% < 0) OR (CHOICE% > MAXCHOICE%) ) THEN GOTO 3700
3850 ON (CHOICE% + 1) GOSUB 3870,1500,1700,200,2300,5000,3300,3500
3860 GOTO 3700
3870 END
4000 REM error subroutine
4010 RESCUE% = ( (ERR = 53) OR (ERR = 52) )
4020 IF NOT(RESCUE%) THEN SCREEN 0 : PRINT BELL$;"Line";ERL; : ERROR ERR : END
4030 RESUME NEXT
5000 REM plot the wave
5010 NPIX = 639 : NPYX = 190 : TMX = 5 : BMX = 5 : LMX = 70 : RMX = 70
5020 PRINT : PRINT "Plot waveform # ( 0 -";MAX.WAVE%;") "; : INPUT WAVE%
5030 GOSUB 3190 : REM determine ymax, ymin
5040 B = (LOG(YRANGE)) / (LOG(10))
5050 TFAC = 10 ^ (INT(B))
5060 YMINP = (INT(YMIN/TFAC)) * TFAC
5070 YMAXP = (INT(YMAX/TFAC) + 1) * TFAC
5080 XRANGE = XINCR(WAVE%) * NR.PT%(WAVE%)
5090 B = (LOG(XRANGE)) / (LOG(10))
5100 XTFAC = 10 ^ (INT(B))
5110 XMAXP = (INT(XRANGE/XTFAC) + 1) * XTFAC
5120 YSC = (NPYX - BMX - TMX) / (YMAXP - YMINP) * (-1)
5130 YOFFX = NPYX - BMX - (YMINP * YSC)
5140 XOFFX = LMX
5150 XSC = (NPIX - LMX - RMX) / XMAXP
5160 REM plot box
5170 SCREEN 2 : KEY OFF : CLS : DRAW "C1"
5180 YP1 = YMAXP * YSC + YOFFX : YP2 = YMINP * YSC + YOFFX
5190 FOR X = 0 TO (XMAXP * 1.01) STEP XTFAC
5200 XP = X * XSC + XOFFX
5210 LINE(XP,YP1)-(XP,YP2)
5220 XPRINT = (INT(X * 1.0001 / XTFAC)) * XTFAC
5230 P$ = STR$(XPRINT) : GOSUB 6000 : REM get string ready to print
5240 PRESET(XP,(YP2 + 2))
5250 GOSUB 8000 : REM print string
5260 NEXT X
5270 XP1 = XOFFX : XP2 = XMAXP * XSC + XOFFX
5280 FOR Y = YMINP TO (YMAXP + (.01 * YRANGE)) STEP TFAC
5290 YP = Y * YSC + YOFFX
5300 LINE(XP1,YP)-(XP2,YP)
5310 YPRINT = (INT( (Y + .001 * TFAC) / TFAC)) * TFAC
5320 P$ = STR$(YPRINT) : GOSUB 6000
5330 I% = LEN(P$) : PRESET((XP1-2-(6*I%)),YP)
5340 GOSUB 8000 : REM print string
5350 NEXT Y
5360 REM start actual plot
5370 XP = XOFFX : YP = D(WAVE%,0) * YSC + YOFFX : PSET(XP,YP)
5380 FOR I% = 0 TO NR.PT%(WAVE%)
5390 XP = I% * XINCR(WAVE%) * XSC + XOFFX
5400 YP = D(WAVE%,I%) * YSC + YOFFX
5410 LINE-(XP,YP)
5420 NEXT I%
5422 LOCATE 2,12:IF WAVE%=1 THEN R$=NAME1$
5426 IF WAVE%=3 THEN R$=NAME3$
5427 IF WAVE%=0 THEN R$=NAME0$
5428 PRINT"Waveform name : ";R$
5430 GOSUB 2000
5440 SCREEN 0 : CLS
5450 RETURN
5460 REM TEST
5470 GOSUB 1900
5480 FOR I% = 0 TO 127 : D(0,I%) = RND(1)*RND(1)*4-2 : NEXT I%
5490 NR.PT%(0) = 127
5500 XINCR(0) = 1
5510 GOTO 2
6000 REM string shortening routine
6010 MAX.DIG% = 10
6020 WHILE ( (LEFT$(P$,1)) = " " )
6030 I% = LEN(P$)
6040 P$ = RIGHT$(P$,(I%-1))
6050 WEND
6060 I% = LEN(P$)
6070 IF (I% <= MAX.DIG%) THEN GOTO 6110
6080 REM else
6090 IF ( (LEFT$(P$,1)) = "-" ) THEN L% = MAX.DIG% ELSE L% = MAX.DIG% - 1
6100 IF ( (MID$(P$,(I%-3),1)) = "E") THEN P$ = LEFT$(P$,(L%-4)) + RIGHT$(P$,4
LSE P$ = LEFT$(P$,L%)
6110 REM
6120 RETURN
7000 REM data for next subrout
7010 DATA 0,13,2,11,3,12,5,0,6,1,7,2,8,3,9,4,10,5,11,6,12,7,13,8,14,9,26,10
7020 DATA C1BRR2FD4GL2HU4BR6BU,C1BFRUD6LR2BR3BU6,C1BDER2FD8D4R4BR2BU6
7030 DATA C1BDER2FDGL2R2FDGL2HBR6BU5,C1BR4G4R4D2U6BR2,C1R4L4D3R3FDGL3BE6
7040 DATA C1BRR2FGL4D4FR2EUHL2BR5BU3,C1R4D6BU3BE6,C1BRR2FDGL2HUED3DFR2EUBR2BU4
7050 DATA C1BRR2FD4GL2BU3R2L2HUBR6BU,C1R4L4D3R2L2D3R4BR2BU6,C1BD3R4BR2BU3
7060 DATA C1BD6BR2R2COLC1BR3BU6,C1BR2BD4U2L2R4BR2BU3
7070 FOR I% = 0 TO 13 : READ J%,K% : CH.INDEX%(J%) = K% : NEXT I%
7080 FOR I% = 0 TO 13 : READ CH$(I%) : NEXT I%
7090 RETURN
8000 REM draw numbers
8010 FOR I% = 1 TO (LEN(P$))
8020 DRAW CH$(CH.INDEX%(ASC(MID$(P$,I%,1))-43))
8030 NEXT I%
8040 RETURN
9050 :
8060 :
8070 :
8080 :
8090 :
8100 :
8110 :
8120 :
9130 :
8140 :
8150 :
3160 :
8170 :
9000 REM            IEEE-488 INTEFACE FOR THE IBM PC V4.10
9010 REM                   WRITTEN IN ADVANCED BASIC
9020 REM AND INCORPORATING ASSEMBLY LANGUAGE ROUTINES TO IMPLEMENT
9030 REM                DMA - DRIVEN GPIB TRANSACTIONS
9040 REM THE ASSEMBLY LANGUAGE ROUTINES MUST BE LOADED PRIOR TO ENTERING
9050 REM BASICA BY TYPING "SUBLIB".  THEN TYPE "BASICA", LOAD IEEE488.BAS,
9060 REM AND CALL SUBROUTINES AS DESCRIBED IN THE MANUAL.
9070 REM
9080 REM                   WRITTEN FOR TECMAR, INC.
9090 REM                             BY
9100 REM                       SCOTT C. JOHNSON
9110 REM
9120 REM (C) Copyright Tecmar, Inc. 1982,1983
9130 REM
9140 REM ********************* START OF SUBROUTINE ***************************
```

APPENDIX 3

Listing of the τ, the f_{3dB} and FFT programs.

```
1000 DELTA: REM $INCLUDE: COMDEF
5 DIM A(530):DIM CH.INDEX%(30):DIM CH$(30),T(512),FLD$(10),SAR$(20),PLOT(530),DT
(512)
10 '**********************************************************************
11 '*
12 '*                  "NOWIND.BAS" P R O G R A M                       *
13 '*                                                                    *
14 '**********************************************************************
15 '
16 '
17 '  This program centers an input array Y(I), removes the DC contents,
18 '  and multiplies it with a modified Hamming window function. The results
19 '  are "chained" to the program "PPPROG2.BAS" where an FFT is computed.
25 '
100 CLS:SCREEN 0:PRINT:PRINT"Do you wish to see the catalog of waveforms (Y/N) ?
"
110 A$=INKEY$:IF A$="" GOTO 110
120 IF (A$="Y") OR (A$="y")THEN PRINT:FILES "*.WFM"
130 PRINT:INPUT"please enter name of file of waveform to retrieve";WF$
140 IF(((RIGHT$(WF$,4))<>".WFM")AND((RIGHT$(WF$,4))<>".wfm"))THEN WF$=WF$+".WFM"

150 OPEN WF$ FOR INPUT AS #1
160 INPUT #1,NRPT:NRPT=NRPT+1
170 CLS:PRINT"WAVEFORM NAME:";WF$
180 PRINT"NRPT=";NRPT
190 INPUT #1,INC
200 PRINT"INC=";INC
210 FOR IX=1 TO NRPT
220 INPUT #1,Y(IX)
230 NEXT IX
240 CLOSE #1
250 DC=0
260 FOR IX=1 TO 10
270 DC=DC+Y(IX)
280 NEXT IX
290 DC=DC/10
300 FOR IX=1 TO NRPT
310 Y(IX)=Y(IX)-DC
320 NEXT IX
500 '
580 '  ************************THE Y(I) ARRAY IS CENTERED************************
590 '
600 MAXI=0:MAXARRAY=-1.5E-30
610 FOR I=1 TO NRPT
620 IF ABS(Y(I))<=ABS(MAXARRAY) THEN GOTO 640
630 MAXARRAY=Y(I):MAXI=I
640 NEXT I
650 IF (MAXI>=NRPT/2-2) AND (MAXI<=NRPT/2+2) THEN GOTO 880
660 IF MAXI>NRPT/2 THEN GOTO 780
670 SHIFT=NRPT/2-MAXI
680 FOR I=SHIFT+1 TO NRPT
690 A(I)=Y(I-SHIFT)
700 NEXT I
710 FOR I=1 TO SHIFT
720 A(I)=A(SHIFT+1)
730 NEXT I
740 FOR I=1 TO NRPT
750 Y(I)=A(I)
760 NEXT I
770 GOTO 880
780 SHIFT=MAXI-NRPT/2
790 FOR I=1 TO NRPT-SHIFT
800 A(I)=Y(I+SHIFT)
810 NEXT I
820 FOR I=NRPT-SHIFT TO NRPT
830 A(I)=A(NRPT-SHIFT-1)
840 NEXT I
850 FOR I=1 TO NRPT
860 Y(I)=A(I)
870 NEXT I
880 '  *********************THE HAMMING WINDOW IS APPLIED**********************
881 '
890 FOR IX=1 TO 25
900 W(IX)=.54+.46*COS((3.141592654*(INC*(IX-1))/(24*INC)))
910 NEXT IX
920 FOR IX=1 TO 25
930 Y(IX)=Y(IX)*W(26-IX)
940 Y(IX+231)=Y(IX+231)*W(IX)
950 NEXT IX
2000 '
2010 '  ************************THE t.FWHM IS CALCULATED************************
2020 '
2030 MAXI=0:MAXARRAY=-1.5E-30
2040 FOR I=1 TO NRPT
```

```
2050 IF ABS(Y(I))<=ABS(MAXARRAY) THEN GOTO 2070
2060 MAXARRAY=Y(I):MAXI=I
2070 NEXT I
2080 I=MAXI
2090 WHILE ABS(Y(I))>ABS(MAXARRAY/2)
2100 I=I+1
2110 WEND
2120 I=I-1
2130 PRINT"MAXARRAY=";MAXARRAY
2140 DELTA=(((ABS(MAXARRAY/2))-ABS(Y(I+1)))/(ABS(Y(I+1))-ABS(Y(I))))+1
2150 UPI=I+DELTA
2160 I=MAXI
2170 WHILE ABS(Y(I))>ABS(MAXARRAY/2)
2180 I=I-1
2190 WEND
2200 I=I+1
2210 DELTA=1-(((ABS(MAXARRAY/2))-ABS(Y(I-1)))/(ABS(Y(I))-ABS(Y(I-1))))
2220 DNI=I-DELTA
2230 DIFF=UPI-DNI
2240 T.FWHM=DIFF*INC
2250 '**********************THE RESULTS CAN BE PLOTTED***********************
2260 '
2270 CLS:KEY OFF
2280 PRINT"Do you want to plot the array Y(I) (Y/N) ?"
2290 A$=INKEY$:IF A$="" GOTO 2290
2300 IF (A$="Y")OR(A$="y")OR(A$="n") OR (A$="N") THEN GOTO 2320
2310 PRINT"CAN YOU TYPE 'Y' OR 'N' PLEASE ?":GOTO 2290
2320 IF (A$="N")OR(A$="n")THEN GOTO 3320
2330 FOR I=1 TO NRPT
2340 PLOT(I)=Y(I)
2350 NEXT I
2360 '***********************************************************************
2370 '*                                                                     *
2380 '*                  "PLOT.DLJ"   S U B P R O G R A M                   *
2390 '*                                                                     *
2400 '***********************************************************************
2410 '
2420 '
2430 '               This program is used to plot an array PLOT(I).
2440 '
2450 '
2460 KEY OFF
2470 SCREEN 2:CLS:PRINT"ENTERING THE PLOTTING PROGRAM...IT'S COMMING '''..."
2480 NPX%=639:NPY%=190:TM%=5:BM%=5:LM%=70:RM%=70
2490 GOSUB 3360
2500 YMAX=-1.7E+35:YMIN=1.7E+35
2510 FOR I%=1 TO NRPT
2520 IF PLOT(I%)<YMIN THEN YMIN=PLOT(I%)
2530 IF PLOT(I%)>YMAX THEN YMAX=PLOT(I%)
2540 NEXT I%
2550 FOR I%=1 TO NRPT/2
2560 IF PLOT(I%)=YMAX THEN MAX.I=I%
2570 NEXT I%
2580 YRANGE=YMAX-YMIN
2590 YMULT=YRANGE/7
2600 YZERO=YMAX-(YRANGE/2)
2610 B=(LOG(YRANGE))/(LOG(10))
2620 TFAC=10^(INT(B))
2630 YMINP=(INT(YMIN/TFAC))*TFAC
2640 YMAXP=(INT(YMAX/TFAC)+1)*TFAC
2650 XRANGE=INC*(NRPT-1)
2660 B=(LOG(XRANGE))/(LOG(10))
2670 XTFAC=10^(INT(B))
2680 XMAXP=(INT(XRANGE/XTFAC)+1)*XTFAC
2690 'LPRINT "XRANGE=";XRANGE;" YMINP=";YMINP;" YMAXP=";YMAXP
2700 'LPRINT "XTFAC=";XTFAC;" XMAXP=";XMAXP
2710 YSC=(NPY%-BM%-TM%)/(YMAXP-YMINP)*(-1)
2720 'LPRINT"YSC=";YSC;" NPY%=";NPY%;" TM%=";TM%;" XMAXP=",XMAXP
2730 'LPRINT"BM%=";BM%
2740  YOFF%=NPY%-BM%-(YMINP*YSC)
2750 XOFF%=LM%
2760 XSC=(NPX%-LM%-RM%)/XMAXP
2770 KEY OFF:CLS:DRAW "C1"
2780 YP1=YMAXP*YSC+YOFF%:YP2=YMINP*YSC+YOFF%
2790 FOR X=0 TO (XMAXP*1.01)STEP XTFAC
2800 XP=X*XSC+XOFF%
2810 LINE (XP,YP1)-(XP,YP2)
2820 XPRINT=(INT(X*1.0001/XTFAC))*XTFAC
2830 P$=STR$(XPRINT):GOSUB 3150
2840 PRESET(XP,(YP2+2))
2850 GOSUB 3290
2860 NEXT X
2870 XP1=XOFF%:XP2=XMAXP*XSC+XOFF%
2880 FOR Y=YMINP TO (YMAXP+(.01*YRANGE))STEP TFAC
2890 YP=Y*YSC+YOFF%
2900 LINE (XP1,YP)-(XP2,YP)
2910 YPRINT=(INT((Y+.001*TFAC)/TFAC))*TFAC
2920 ' LPRINT "YPRINT=";YPRINT
2930 P$=STR$(YPRINT):GOSUB 3150
2940 I%=LEN(P$):PRESET((XP1-2-(6*I%)),YP)
2950 GOSUB 3290
2960 NEXT Y
2970 XP=INC*XSC+XOFF%:YP=PLOT(1)*YSC+YOFF%:PSET(XP,YP)
2980 FOR I%=1 TO NRPT
2990 XP=I%*INC*XSC + XOFF%
3000 YP=PLOT(I%)*YSC+YOFF%
3010 LINE -(XP,YP)
3020 NEXT I%
3030 X2=DNI*INC*XSC+XOFF%:X1=UPI*INC*XSC+XOFF%
3040 DELTAY=((MAXARRAY/2)*YSC-YMINP*YSC)/20
3050 FOR I%=1 TO 20
3060 Y1=(MAXARRAY/2)*YSC+YOFF%-I%*DELTAY
3070 PSET(X1,Y1):PSET(X2,Y1)
3080 NEXT I%
3090 LOCATE 22,45
3100 PRINT"t.FWHM=";T.FWHM;" sec"
3110 LOCATE 2,12:PRINT"Waveform : ";WF$
3120 A$=INKEY$:IF A$="" GOTO 3120
3130 GOTO 3500
3140 '
3150 '
3160 '***************STRING SHORTENING SUBROUTINE***************************
3170 '
3180 MAX.DIG%=10
3190 WHILE ((LEFT$(P$,1))=" ")
3200 I%=LEN(P$)
3210 P$=RIGHT$(P$,(I%-1))
3220 WEND
3230 I%=LEN(P$)
3240 IF (I%<=MAX.DIG%)THEN GOTO 3270
3250 IF ((LEFT$(P$,1))="-")THEN L%=MAX.DIG% ELSE L%=MAX.DIG%-1
3260 IF ((MID$(P$,(I%-3),1))="E")THEN P$=LEFT$(P$,(L%-4))+RIGHT$(P$,4)ELSE P$=LE
FT$(P$,L%)
3270 REM
3280 RETURN
3290 '
3300 '***************NUMBER PLOTTING SUBROUTINE*****************************
3310 '
3320 FOR I%=1 TO (LEN(P$))
3330 DRAW CH$(CH.INDEX%(ASC(MID$(P$,I%,1))-43))
3340 NEXT I%
3350 RETURN
```

```
3360
3370   ***************DATA FOR NUMBER PLOTTING SUBROUTINE*****************
3380
3390 DATA 0,13,2,11,3,12,5,0,6,1,7,2,8,3,9,4,10,5,11,6,12,7,13,8,14,9,26,10
3400 DATA C1BRR2FD4GL2HU4BR6BD,C1BFRUD6LR2BR3BU6,C1BDER2FDG4R4BR2BU6
3410 DATA C1BDER2FDGL2R2FDGL2HBR6BU5,C1BR4G4R4D2U6BR2,C1R4L4D3R3FDGL3BE6
3420 DATA C1BRR2FBL4D4FR2EUHL2BR5BU3,C1R4DBD63BE6,C1BRR2FDGL2HUBD3DFR2EUBR2BU4
3430 DATA C1BRR2FD4GL2BU3R2L2HU3R6BU,C1R4L4D3R2L2D3R4BR2BU6,C1BD3R4BR2BU3
3440 DATA C1BD6BR2R2COLC1BR3BU6,C1BR2DD4U2L2R4BR2BU3
3450 FOR I% = 0 TO 13 : READ J%,K% : CH.INDEX%(J%) = K% : NEXT I%
3460 FOR I%=0 TO 13:READ CH$(I%):NEXT I%
3470 RETURN
3480 DATA &H55,&HCD,&H05,&H5D,&HC8
3490
3500
3510
3520
3530
3540 INPUT " DO YOU WANT A PLOT ON THE 7079A PLOTTER ";A$.
3550 IF A$="Y" OR A$="y" THEN GOSUB 3630
3560 SCREEN 0:PRINT"Type 'Q' to exit..."
3570 PRINT"..or, type any other key to proceed with the FFT computation..."
3580 A$=INKEY$:IF A$="" THEN GOTO 3580
3590 IF (A$="Q")OR (A$="q")THEN GOTO 3610
3600 GOTO 3620
3610 END
3620 CHAIN"PROG2"
3630 REM INITIALIZE PLOTTER AND PARAMETERS
```

```
1 REM $INCLUDE: COMDEF
5 DIM I.NORM(1024):DIM I.ANGLE(1024)
6 DIM PLOT(1024):DIM CH$(30):DIM CH.INDEX%(30),DT(1024),T(1024),SAR$(20),FLD$(10
)
50   ****************************************************************************
51   *                                                                          *
52   *                         "PPPROG2.BAS " P R O G R A M                     *
53   *                                                                          *
54   ****************************************************************************
55
56       This program is used together with FFT4.BAS or GAUSSIAN.BAS programs.
57   An array Y(I) is received from either of them these sampled points
58   are used to fill the center portion of a 1024 points array , i.e. I.REAL.
59   Another array ,I.IMAG(I),is filled with zeros.Both arrays are passed to
60   the subprogram FFT where a Fast Fourier Transform is computed.
100
101   *******************THE CONSTANTS ARE DEFINED*****************************
102
118 PSTART=(1024-NRPT)/2:PEND=(1024-PSTART)+1:INIT=NRPT
119 NRPT=1024:DELTAF=1/(1024*INC)
200   *******************THE ARRAY I IS DEFINED*********************************
201
205 KEY OFF:CLS:N=LOG(NRPT)/LOG(2):WIND=1/(NRPT-1)*INC
880 FOR J%=1 TO PSTART
890 I.REAL(J%)=0
895 PLOT(J%)=I.REAL(J%)
900 I.IMAG(J%)=0
910 NEXT J%
920 FOR I%=PSTART+1 TO PEND-1
930 I.REAL(I%)=Y(I%-PSTART)
935 PLOT(I%)=I.REAL(I%)
940 I.IMAG(I%)=0
950 NEXT I%
960 FOR I%=PEND TO NRPT
970 I.REAL(I%)=0
975 PLOT(I%)=I.REAL(I%)
980 I.IMAG(I%)=0
990 NEXT I%
991 RET.NO=1
1000   ****************************************************************************
1010   *                                                                          *
1020   *                       " FFT " S U B P R O G R A M                        *
1030   *                                                                          *
1040   ****************************************************************************
1050
1060          This program is used to compute a Fast Fourier Transform on
1070   an input array I.The FFT results are plotted.
1090
1340   *******************ISIGN IS DEFINED****************************************
1350                                    note: ISIGN=1 for Forward Transform
1360                                          ISIGN=0 for Inverse Transform
1370
1380 KEY OFF:CLS:PRINT"WHAT TYPE OF FFT DO YOU NEED ?"
1390 PRINT"":PRINT"TYPE 'F' FOR FORWARD TRANSFORM ,"
1400 PRINT" OR 'I' FOR INVERSE :"
1410 A$=INKEY$:IF A$="" GOTO 1410
```

```
1420 IF A$="F"OR A$="f" OR A$="I" OR A$="i" GOTO 1440
1430 PRINT"CAN YOU TYPE 'F' OR 'I' PLEASE ?":GOTO 1410
1440 IF A$="F" OR A$="f" THEN ISIGN=1
1450 IF A$="I" OR A$="i" THEN ISIGN=0
1460 CLS:PRINT"PROGRAM IN ACTION...BE PATIENT ''''"
1470
1480   ********************BEGINNING OF THE FFT************************************
1490
1500 N=2^M:NV2=N/2:NM1=N-1:J=1
1510
1520 FOR I=1 TO NM1
1530 IF I>=J THEN GOTO 1570
1540 T.REAL=X.REAL(J):T.IMAG=X.IMAG(J)
1550 X.REAL(J)=X.REAL(I):X.IMAG(J)=X.IMAG(I)
1560 X.REAL(I)=T.REAL:X.IMAG(I)=T.IMAG
1570 K=NV2
1580 IF K>=J THEN GOTO 1620
1590 J=J-K
1600 K=K/2
1610 GOTO 1580
1620 J=J+K
1630 NEXT I
1640
1650 PI=3.141592653589798
1660 CLS:M=INT(M+.5)
1665 PRINT"WAVEFORM NAME :"WF$
1670 FOR L=1 TO M
1680 PRINT TAB(5*L);M-L+1;"/";M;" TO GO..."
1690 LE=2^L:LE1=LE/2
1700 U.REAL=1:U.IMAG=9.999999D-21
1710 IF ISIGN=1 THEN GOTO 1750
1720 W.REAL=COS(PI/LE1):W.IMAG=SIN(PI/LE1)
1730  LPRINT "W.REAL=";W.REAL;"   W.IMAG=";W.IMAG
1740 GOTO 1770
1750 W.REAL=COS(PI/LE1):W.IMAG=-SIN(PI/LE1)
1760
1770 FOR J=1 TO LE1
1780 FOR I=J TO N STEP LE
1790 IP=I+LE1
1880 T.REAL=X.REAL(IP)*U.REAL - X.IMAG(IP)*U.IMAG
1890 T.IMAG=X.REAL(IP)*U.IMAG + X.IMAG(IP)*U.REAL
1895 S$="FFFFF"
1900 X.REAL(IP)=X.REAL(I)-T.REAL
1910 X.IMAG(IP)=X.IMAG(I)-T.IMAG
1920 X.REAL(I)=X.REAL(I)+T.REAL
1930 X.IMAG(I)=X.IMAG(I)+T.IMAG
1940 NEXT I
2000 T.REAL=U.REAL*W.REAL - U.IMAG*W.IMAG
2010 T.IMAG=U.REAL*W.IMAG + U.IMAG*W.REAL
2014 U.REAL=T.REAL:U.IMAG=T.IMAG
2020 NEXT J
2030 NEXT L
2040 MARY$="GFE"
2050 PLAY MARY$
2060 IF CHECK=1 GOTO 2470
2070 PRINT"DO YOU NEED AN INVERSE FFT OF THESE RESULTS (Y/N) ?"
2080 A$=INKEY$:IF A$="" GOTO 2080
2090 IF (A$="Y") OR (A$="N") THEN GOTO 2110
2100 PRINT"CAN YOU TYPE 'Y' OR 'N' PLEASE ?":GOTO 2080
2110 IF A$="N" GOTO 2190
2120 CHECK=1
2130 ISIGN=0
2140 GOTO 1460
2190 FOR I=1 TO NRPT
2200 X.NORM(I)=SQR(X.REAL(I)^2+X.IMAG(I)^2)
2210 NEXT I
```

```
2335 NRPT=64
2340 INC=DELTAF
2350 PRINT"DO YOU NEED TO STORE X.NORM(I) ARRAY IN A FILE (Y/N)? "
2360 A$=INKEY$:IF A$="" GOTO 2360
2370 IF (A$="Y") OR (A$="N") THEN GOTO 2390
2380 PRINT"CAN YOU TYPE 'Y OR 'N' PLEASE ?":GOTO 2360
2390 IF A$="N" GOTO 2470
2400 PRINT:INPUT"Please enter the name of the file to be saved :  ";B$
2410 IF(((RIGHT$(B$,4))<>".FFT")AND((RIGHT$(B$,4))<>".fft"))THEN B$=B$+".FFT"
2420 OPEN B$ FOR OUTPUT AS #1
2422 PRINT #1,512
2424 PRINT #1,INC
2430 FOR I=1 TO 512
2440 PRINT #1,X.NORM(I)
2450 NEXT I
2460 CLOSE
2470  ********************THE RESULTS ARE PLOTTED********************************
2480
2490
2500 FOR I=1 TO 1024
2570 PLOT(I)=X.NORM(I)
2580 NEXT I
2585 BLOB=0
2590 FULL=NRPT*INC
2700  ************************************************************************
2710   *                                                                      *
2720   *                 "PLOT.GLJ"   S U B P R O G R A M                     *
2730   *                                                                      *
2740  ************************************************************************
2750
2760
2770         This program is used to plot the results of the FFT.
2780
2790
2800 KEY OFF
2810 SCREEN 2:CLS:PRINT"ENTERING THE PLOTTING PROGRAM...IT'S COMMING ''''..."
2820 NPIX=639:NPYX=190:TMX=5:BMX=5:LMX=70:RMX=70
2830 IF RET.NO=1 THEN GOSUB 4670
2835 IF BLOB=5 THEN GOTO 3150
2840 YMAX=-1.7E+35:YMIN=1.7E+35
2850 FOR IX=1 TO 512
2860 IF PLOT(IX)<YMIN THEN YMIN=PLOT(IX)
2870 IF PLOT(IX)>YMAX THEN YMAX=PLOT(IX)
2880 NEXT IX
2890 FOR IX=1 TO 512
2900 IF PLOT(IX)=YMAX THEN MAX.I=IX
2910 NEXT IX
3000  ********************THE 3 dB FREQUENCY IS CALCULATED**********************
3001
3010
3020 DELTA=1.35E+20
3030 FOR I=MAX.I TO NRPT
3040 TESTDELTA=ABS(PLOT(I)-YMAX/1.414213)
3050 IF PLOT(I)<YMAX/1.414213 THEN GOTO 3090
3060 IF TESTDELTA>=DELTA THEN GOTO 3090
3070 'LPRINT "DELTA=";DELTA
3080 FWHM.INDEX=I:DELTA=TESTDELTA
3090
3100
3110 NEXT I
3120 INDEX=FWHM.INDEX+(PLOT(FWHM.INDEX)-YMAX/1.414213)/(PLOT(FWHM.INDEX)-PLOT(FW
HM.INDEX+1))
3130 LOCATE 5,45
3140 FWHM=(INDEX-1)*DELTAF
3150 CLS:PRINT"RANGE=";FULL;" Hz"
```

```
3170 PRINT"":PRINT"The range is ";FULL/FWHM;" times the F.3dB ; is it OK [illegible]"
3180 A$=INKEY$:IF A$="" GOTO 3180
3182 IF(A$="Y")OR(A$="y")OR(A$="N")OR(A$="n")THEN GOTO 3190
3184 PRINT"CAN YOU TYPE 'Y' OR 'N' PLEASE ?":GOTO 3180
3190 IF(A$="N")OR(A$="n")THEN GOTO 3200
3195 GOTO 4000
3200 PRINT"":PRINT"NRPT=";NRPT;" points."
3205 INPUT"What new value of NRPT do you want";NRPT
3210 BLOB=5:GOTO 2590
4000 YRANGE=YMAX-YMIN
4010 YMULT=YRANGE/7
4020 YZERO=YMAX-(YRANGE/2)
4030 B=(LOG(YRANGE))/(LOG(10))
4040 TFAC=10^(INT(B))
4050 YMINP=(INT(YMIN/TFAC))*TFAC
4060 YMAXP=(INT(YMAX/TFAC)+1)*TFAC
4070 XRANGE=INC*(NRPT-1)
4080 B=(LOG(XRANGE))/(LOG(2))
4090 XTFAC=2^(INT(B))
4100 XMAXP=(INT(XRANGE/XTFAC)+1)*XTFAC
4110  LPRINT "XRANGE=";XRANGE;" YMINP=";YMINP;" YMAXP=";YMAXP
4120  LPRINT "XTFAC=";XTFAC;" XMAXP=";XMAXP
4130 YSC=(NPYX-BMX-TMX)/(YMAXP-YMINP)*(-1)
4140  LPRINT"YSC=";YSC;" NPYX=";NPYX;" TMX=";TMX;" XMAXP=";XMAXP
4150 'LPRINT"BMX=";BMX
4160  YOFFX=NPYX-BMX-(YMINP*YSC)
4170 XOFFX=LMX
4180 XSC=(NPXX-LMX-RMX)/XMAXP
4190 KEY OFF:CLS:DRAW "C1"
4200 YP1=YMAXP*YSC+YOFFX:YP2=YMINP*YSC+YOFFX
4210 FOR X=0 TO (XMAXP*1.01)STEP XTFAC
4220 XP=X*XSC+XOFFX
4230 LINE (XP,YP1)-(XP,YP2)
4240 XPRINT=(INT(X*1.0001/XTFAC))*XTFAC
4250 P$=STR$(XPRINT):GOSUB 4460
4260 PRESET(XP,(YP2+2))
4270 GOSUB 4600
4280 NEXT X
4290 XP1=XOFFX:XP2=XMAXP*XSC+XOFFX
4300 FOR Y=YMINP TO (YMAXP+(.01*YRANGE))STEP TFAC
4310 YP=Y*YSC+YOFFX
4320 LINE (XP1,YP)-(XP2,YP)
4330 YPRINT=(INT((Y+.001*TFAC)/TFAC))*TFAC
4340 ' LPRINT "YPRINT=";YPRINT
4350 P$=STR$(YPRINT):GOSUB 4460
4360 IX=LEN(P$):PRESET((XP1-2-(6*IX)),YP)
4370 GOSUB 4600
4380 NEXT Y
4390 XP=XOFFX:YP=PLOT(1)*YSC+YOFFX:PSET(XP,YP)
4400 FOR IX=1 TO NRPT
4410 XP=(IX-1)*INC*XSC + XOFFX
4420 YP=PLOT(IX)*YSC+YOFFX
4430 LINE -(XP,YP)
4440 NEXT IX
4450 GOTO 4800
4460
4470  ****************STRING SHORTENING SUBROUTINE****************************
4480
4490 MAX.DIGX=10
4500 WHILE ((LEFT$(P$,1))=" ")
4510 IX=LEN(P$)
4520 P$=RIGHT$(P$,(IX-1))
4530 WEND
4540 IX=LEN(P$)
4550 IF IX<=MAX.DIGX THEN GOTO 4580
4560 IF [illegible]LEFT$(P$,1)="-")THEN LX=MAX.DIGX ELSE LX=MAX.DIGX
4570 IF [illegible]MID$(P$,(IX-3),1)="E")THEN P$=LEFT$(P$,(LX-4))+RIGHT$(P$,4)ELSE P$=LE
FT$(P$,LX)
4580 REM
4590 RETURN
4600
4610  ****************NUMBER PLOTTING SUBROUTINE****************************
4620
4630 FOR IX=1 TO (LEN(P$))
4640 DRAW CH$(CH.INDEXX(ASC(MID$(P$,IX,1))-43))
4650 NEXT IX
4660 RETURN
4670
4675 RET.NO=0
4680  ****************DATA FOR NUMBER PLOTTING SUBROUTINE******************
4690
4700 DATA 0,13,2,11,3,12,5,0,6,1,7,2,8,3,9,4,10,5,11,6,12,7,13,8,14,9,26,10
4710 DATA C1BRR2FD4GL2HU4BR6BU,C1BFRUD6LR2BR3BU6,C1BDER2FDG4R4BR2BU6
4720 DATA C1BDER2FDGL2R2FDGL2HBR6BU5,C1BR4G4R4D2U6BR2,C1R4L4D3R3FDGL3BE6
4730 DATA C1BRR2FBL4D4FR2EUHL2BR5BU3,C1R4D8DG3BE6,C1BRR2FDGL2HUBD3DFR2EUBR2BU4
4740 DATA C1BRR2FD4GL2BU3R2L2HUBR6BU,C1R4L4D3R2L2D3R4BR2BU6,C1BD3R4BR2BU3
4750 DATA C1BD6BR2R2COLC1BR3BU6,C1BR2DD4U2L2R4BR2BU3
4760 FOR IX = 0 TO 13 : READ JX,KX : CH.INDEXX(JX) = KX : NEXT IX
4770 FOR IX=0 TO 13:READ CH$(IX):NEXT IX
4780 RETURN
4790 DATA &H55,&HCD,&H05,&H5D,&HCB
4800
4810
10010 LOCATE 5,45:PRINT"F.3dB=";FWHM;"Hz"
10020 DELTAY=((YMAX/1.414213)*YSC-YMINP*YSC)/20
10030 XX=(INDEX-1)*INC*XSC+XOFFX
10040 FOR IX=1 TO 20
10050 Y1=(YMAX/1.414213)*YSC+YOFFX-IX*DELTAY
10060 PSET(XX,Y1)
10070 NEXT IX
10080 LOCATE 2,12:PRINT"Waveform name \ ";NF$
10090 A$=INKEY$:IF A$="" GOTO 10090
10100 INPUT "DO YOU NEED A PLOT ON 7090A PLOTTER " ; A$
10110 IF A$="Y" OR A$="y" THEN GOSUB 10170
10120 CLS:PRINT"Do you need to execute an FFT on another waveform (Y/N) ?"
10130 A$=INKEY$:IF A$="" THEN GOTO 10130
10140 IF (A$="n") OR (A$="N")THEN GOTO 10160
10150 CHAIN"FFT"
10160 END
10170 REM INITIALIZE PLOTTER AND PARAMETERS
```

APPENDIX 4

Listing of the deconvolution program.

```
  DIM ...(512), ...(2),SAM$(20),FLD$(10)
  DIM ...$(30),CH.INDEX%(30),PLOT(512)
  DIM NUM(512),INTER(512),RES(512)
  DIM DEN(512)
  CHECK=0
50   ************************************************************************
51   *                                                                      *
52   *                     "DOUT.BAS" P R O G R A M                         *
53   *                                                                      *
54   ************************************************************************
55
56
57                     This program is used to divide the array  NUM(I%) by
58     DEN(I%),and to store the results in another array RES(I%).The three
59     arrays are plotted.

     ************************THE ARRAYS ARE DEFINED***************************

106 CLS:SCREEN 0:PRINT"Do you wich to see the catalog of waveforms (Y/N) ?"
108 A$=INKEY$:IF A$="" GOTO 108
110 IF (A$="Y") OR (A$="y") THEN PRINT:FILES "*.FFT"
120 PRINT:INPUT"please enter name of input waveform file    ",LF$
122 IF(((RIGHT$(LF$,4))<>".FFT")AND((RIGHT$(LF$,4))<>".fft"))THEN LF$=LF$+".FF

130 PRINT:INPUT"please enter name of output waveform file   ",KF$
132 IF(((RIGHT$(KF$,4))<>".FFT")AND((RIGHT$(KF$,4))<>".fft"))THEN KF$=KF$+".FFT"

190 CLS
199
200   ***************************DEN(I) VALUES ARE READ**************************
201
210 OPEN LF$ FOR INPUT AS #1
212 INPUT #1,NRPT1
214 INPUT #1,INC1
216 INC=INC1:NRPT=NRPT1:DELTAF=INC
217   PRINT"NUMERATOR :":PRINT:PRINT"Waveform Name : ";LF$
218   PRINT"Number of points : ";NRPT:PRINT"Frequency increment : ";INC;" Hz"
220 FOR I%=1 TO NRPT
230 INPUT #1,Y
232 DEN(I%)=Y
235 PLOT(I%)=DEN(I%)
240 NEXT I%
250 CLOSE
260 NF$=LF$:RETURN.NO=1
270 GOTO 2000
299
300   ***************************NUM(I) VALUES ARE READ**************************
301
330 OPEN KF$ FOR INPUT AS #2
332 INPUT #2,NRPT2
334 INPUT #2,INC2
336 INC=INC2:NRPT=NRPT2:DELTAF=INC
338   PRINT"NUMERATOR :":PRINT:PRINT"Waveform Name : ";KF$
339   PRINT"Number of points : ";NRPT:PRINT"Frequency increment : ";INC;" Hz"
340 FOR I%=1 TO NRPT
350 INPUT #2,Y
360 NUM(I%)=Y
370 PLOT(I%)=NUM(I%)
380 NEXT I%
390 CLOSE
392 NF$=KF$:RETURN.NO=2
399
400   ***********************INTER(I) VALUES ARE CALCULATED*********************
401
410 IF(INC1)=INC2)THEN GOTO 450
420 PRINT"ERROR!!!..INC1<INC2"
430 END
450 IF(INC1=INC2)THEN GOTO 560
460 PLUS=INC2/INC1
470 INTER(1)=DEN(1)
480 J%=1
490 FOR I%=2 TO 256
500 IS=1+(I%-1)*PLUS
510 IF(IS<J%+1)THEN GOTO 530
520 J%=J%+1
530 INTER(I%)=DEN(J%+1)-(DEN(J%+1)-DEN(J%))*(J%+1-IS)
540 NEXT I%
550 GOTO 590
560 FOR I%=1 TO 256
570 INTER(I%)=DEN(I%)
580 NEXT I%
590 NRPT=256:NF$=LF$:RETURN.NO=3:INC=INC2
592 FOR I%=1 TO 256
594 PLOT(I%)=INTER(I%)
596 NEXT I%
600 GOTO 2000
699
700   ***************************RES(I) VALUES ARE CALCULATED*******************
701
710 FOR I%=1 TO 256
720 RES(I%)=NUM(I%)/INTER(I%)
730 PLOT(I%)=RES(I%)
740 NEXT I%
750 INC=INC2:NRPT=256
760 NF$=KF$+"/"+LF$
770 RETURN.NO=4:NRPT=256
780 GOTO 2000
1030
1040
1050
1060
2000   ************************************************************************
2010   *                                                                      *
2020   *                    "PLOT"   SUB P R O G R A M                        *
2030   *                                                                      *
2040   ************************************************************************
2050
2060
2070                    This program is used to plot an array PLOT(I).
2080
2090
2100 KEY OFF:BLOB=0
2110 SCREEN 2:CLS:FULL=NRPT*INC
2120 NPIX=639:NPYX=190:TMX=5:BMX=5:LMX=70:RMX=70
2130 IF CHECK=0 THEN GOSUB 4270
2135 IF BLOB=5 THEN GOTO 2400
2140 YMAX=-1.7E+20:YMIN=1.7E+20
2150 FOR I%=1 TO 32
2160 IF PLOT(I%)<YMIN THEN YMIN=PLOT(I%)
2170 IF PLOT(I%)>YMAX THEN YMAX=PLOT(I%)
2180 NEXT I%
2190 FOR I%=1 TO 32
2200 IF PLOT(I%)=YMAX THEN MAX.I=I%
2210 NEXT I%
2215 IF RETURN.NO=3 THEN GOTO 3000
2220
```

```
2240
2260 DELTA=1.35E+20
2270 FOR I=MAX.1 TO 32
2280 TESTDELTA=ABS(PLOT(I)-YMAX/1.414213)
2290 IF PLOT(I)<YMAX/1.414213 THEN GOTO 2330
2300 IF TESTDELTA>=DELTA THEN GOTO 2330
2310 FWHM.INDEX=I
2320 DELTA=TESTDELTA
2330 NEXT I
2360 INDEX=FWHM.INDEX+(PLOT(FWHM.INDEX)-YMAX/1.414213)/(PLOT(FWHM.INDEX)-PLOT(FWHM.INDEX+1))
2380 FWHM=(INDEX-1)*DELTAF
2400 CLS:PRINT"RANGE=";FULL;" Hz"
2410 PRINT"F.3dB=";FWHM;"Hz"
2420 PRINT "":PRINT"The range is ";FULL/FWHM;" times the F.3dB : is it OK (Y/N)"
2430 A$=INKEY$:IF A$="" THEN GOTO 2430
2440 IF (A$="Y")OR(A$="y")OR(A$="N")OR(A$="n")THEN GOTO 2460
2450 PRINT"can you type 'Y' or 'N' PLEASE ?":GOTO 2430
2460 IF(A$="N")OR(A$="n")THEN GOTO 2480
2470 GOTO 3000
2480 PRINT "":PRINT"NRPT=";NRPT;" points."
2490 INPUT"What new value of NRPT do you want ";NRPT
2500 BLOB=5:GOTO 2110
3000 YRANGE=YMAX-YMIN
3010 YMULT=YRANGE/7
3020 YZERO=YMAX-(YRANGE/2)
3030 B=(LOG(YRANGE))/(LOG(10))
3040 TFAC=10^(INT(B))
3050 YMINP=(INT(YMIN/TFAC))*TFAC
3060 YMAXP=(INT(YMAX/TFAC)+1)*TFAC
3070 IRANGE=INC*(NRPT-1)
3080 B=(LOG(IRANGE))/(LOG(10))
3090 ITFAC=10^(INT(B))
3100 IMAXP=(INT(IRANGE/ITFAC)+1)*ITFAC
3110 'LPRINT "IRANGE=";IRANGE;" YMINP=";YMINP;" YMAXP=";YMAXP
3120 'LPRINT "ITFAC=";ITFAC;" IMAXP=";IMAXP
3130 YSC=(NPYX-BMX-TMX)/(YMAXP-YMINP)*(-1)
3140 'LPRINT"YSC=";YSC;" NPYX=";NPYX;" TMX=";TMX;" IMAXP=";IMAXP
3150 'LPRINT"BMX=";BMX
3160 YOFFX=NPYX-BMX-(YMINP*YSC)
3170 IOFFX=LMX
3180 ISC=(NPIX-LMX-RMX)/IMAXP
3190 KEY OFF:CLS:DRAW "[illegible]"
3200 YP1=YMAXP*YSC+YOFFX:YP2=YMINP*YSC+YOFFX
3210 FOR X=0 TO (IMAXP*1.01)STEP ITFAC
3220 IP=X*ISC+IOFFX
3230 LINE (IP,YP1)-(IP,YP2)
3240 IPRINT=(INT(X*1.0001/ITFAC))*ITFAC
3250 P$=STR$(IPRINT):GOSUB 4060
3260 PRESET(IP,(YP2+2))
3270 GOSUB 4200
3280 NEXT X
3290 IP1=IOFFX:IP2=IMAXP*ISC+IOFFX
3300 FOR Y=YMINP TO (YMAXP+(.01*YRANGE))STEP TFAC
3310 YP=Y*YSC+YOFFX
3320 LINE (IP1,YP)-(IP2,YP)
3330 YPRINT=(INT((Y+.001*TFAC)/TFAC))*TFAC
3340 ' LPRINT"YPRINT=";YPRINT
3350 P$=STR$(YPRINT):GOSUB 4060
3360 IX=LEN(P$):PRESET((IP1-2-(6*IX)),YP)
3370 GOSUB 4200
3380 NEXT Y
3390 IP=IOFFX:YP=PLOT(1)*YSC+YOFFX:PSET(IP,YP)
[illegible]
3410 IP=(IX-1)*INC*ISC + IOFFX
3420 YP=PLOT(IX)*YSC+YOFFX
3430 LINE -(IP,YP)
3440 NEXT IX
3450 IF RETURN.NO=1 THEN A$="(denominator)"
3460 IF RETURN.NO=2 THEN A$="(numerator)"
3470 IF RETURN.NO=3 THEN A$="(Interpolated)"
3475 IF RETURN.NO=4 THEN A$=" "
3480 LOCATE 2,12:PRINT"Waveform Name : ";WF$+A$
3481 IF RETURN.NO=3 THEN GOTO 4000
3482 LOCATE 5,45:PRINT"F.3dB=";FWHM;" Hz"
3490 DELTAY=((YMAX/1.414213)*YSC-YMINP*YSC)/20
3500 X1=(INDEX-1)*INC*ISC+IOFFX
3510 FOR IX=1 TO 20
3520 Y1=((YMAX/1.414213)*YSC)+YOFFX-(IX*DELTAY)
3530 PSET(X1,Y1)
3540 NEXT IX
4000 A$=INKEY$:IF A$="" THEN GOTO 4000
4005 IF (A$<>"N")AND(A$<>"n") THEN GOTO 4010
4007 BLOB=5:GOTO 2110
4010 CLS:SCREEN 0
4020 IF RETURN.NO=1 THEN GOTO 300
4030 IF RETURN.NO=2 THEN GOTO 400
4040 IF RETURN.NO=3 THEN GOTO 700
4050 GOTO 10000
4060
4070 ***************STRING SHORTENING SUBROUTINE*************************
4080
4090 MAX.DIGX=10
4100 WHILE ((LEFT$(P$,1))=" ")
4110 IX=LEN(P$)
4120 P$=RIGHT$(P$,(IX-1))
4130 WEND
4140 IX=LEN(P$)
4150 IF (IX<=MAX.DIGX)THEN GOTO 4180
4160 IF ((LEFT$(P$,1))="-")THEN LX=MAX.DIGX ELSE LX=MAX.DIGX-1
4170 IF ((MID$(P$,(IX-3),1))="E")THEN P$=LEFT$(P$,(LX-4))+RIGHT$(P$,4)ELSE P$=LEFT$(P$,LX)
4180 REM
4190 RETURN
4200
4210 ***************NUMBER PLOTTING SUBROUTINE**************************
4220
4230 FOR IX=1 TO (LEN(P$))
4240 DRAW CH$(CH.INDEX%(ASC(MID$(P$,IX,1))-43))
4250 NEXT IX
4260 RETURN
4270
4280 ***************DATA FOR NUMBER PLOTTING SUBROUTINE*****************
4290
4300 DATA 0,13,2,11,3,12,5,0,6,1,7,2,8,3,9,4,10,5,11,6,12,7,13,8,14,9,26,10
4310 DATA C1BRR2FD4GL2HU4BR6U,C1BFRUD6LR2BR3BU6,C1BDER2FD6G4R4BR2BU6
4320 DATA C1BDER2FDGL2R2FDGL2HBR6BU5,C1BR4D4R4D2U6BR2,C1R4L4D3R3FDGL3BE6
4330 DATA C1BRR2FBL4D4FR2EUHL2BR5BU3,C1R4DGDG3BE6,C1BRR2FDGL2HUBD3DFR2EUBR2BU4
4340 DATA C1BRR2FD4GL2BU3R2L2HUBR6BU,C1R4L4D3R2L2D3R4BR2BU6,C1BD3R4BR2BU3
4350 DATA C1BD6BR2R2C0LC1BR3BU6,C1BR2BD4U2L2R4BR2BU3
4360 FOR IX = 0 TO 13 : READ JX,KX : CH.INDEX%(JX) = KX : NEXT IX
4370 FOR IX=0 TO 13:READ CH$(IX):NEXT IX
4380 CHECK=10
4390 RETURN
4400 DATA &H55,&HCD,&H05,&H5D,&HCB
4410 A$="D"
4420 PRINT A$
10000 INPUT "DO YOU NEED A PLOT ON 7090A PLOTTER " ; A$
10010 IF A$="Y" OR A$="y" THEN GOSUB 10070
[illegible]
10030 A$=INKEY$:IF A$="" THEN GOTO 10030
10040 IF (A$="n") OR (A$="N")THEN GOTO 10060
10050 GOTO 100
10060 END
10070 REM INITIALIZE PLOTTER AND PARAMETERS
```