NTIA Report 87-216
NCS Technical Information Bulletin 87-22

A Program Description of FIBRAM: A Radiation Attenuation Model for Optical Fibers

W. J. Ingram

U.S. DEPARTMENT OF COMMERCE
Malcolm Baldrige, Secretary

Alfred C. Sikes, Assistant Secretary
for Communications and Information

NATIONAL COMMUNICATIONS SYSTEM

June 1987

PREFACE

This report represents part of a series of studies conducted for the National Communications System (NCS), Technology and Standards Branch, Office of the Manager, Washington, DC.

The views, opinions, and/or conclusions contained in this report are those of the author and should not be construed as an official NCS, U.S. Department of Commerce (DOC), or National Telecommunications and Information Administration (NTIA) policy or decision unless designated by other official documentation.

Certain commercial equipment, instruments, services, protocols, and materials are identified in this report to adequately specify the engineering issues. Such identification does not imply recommendation or endorsement by NCS, DOC, or NTIA, nor does it imply that the material, equipment, or service identified is necessarily the best available for the purpose.

TABLE OF CONTENTS

	Page
LIST OF FIGURES	vii
LIST OF TABLES	viii
ABSTRACT	1
1. INTRODUCTION	1
1.1 Description of the Program	1
1.2 The Scope of this Paper	2
1.3 System Description	2
2. BASIS FOR THE MODEL	2
2.1 Optical Fiber Links	2
2.2 Optical Fiber Sublinks	3
2.3 Optical Fiber Sublink Sections	5
3. HOW FIBRAM WORKS	9
3:1 Framework Subroutines	9
3.2 Utility Subroutines	20
3.3 Calculating Subroutines	29
4. APPLICATIONS OF FIBRAM	40
4.1 Modifying a Previously Entered Data File	40
4.2 Substituting Fiber Types	41
4.3 Using the RANGER Feature	41
4.4 Data Perturbations	44
5. REFERENCES	49
APPENDIX A. USING THE PROGRAM	51
A.1 Basic Program Usage	51
A.2 Advanced Features	54
APPENDIX B. MODIFYING FIBRAM	57
B.1 Updating the Program Disk	57
B.2 Specific Kinds of Modifications	60

TABLE OF CONTENTS (cont.)

Page

APPENDIX C. VARIABLES USED IN THIS REPORT 65

APPENDIX D. FILES ON THE FIBRAM PROGRAM DISK 67

APPENDIX E. SAMPLE RUNS OF FIBRAM 69

E.1 Modifying a Previously Entered Data File 69
E.2 Using the RANGER Feature 73

APPENDIX F. SOURCE CODE LISTINGS 79

F.1 Program FIBRAM 79
F.2 Framework Subroutines 82
F.3 Utility Subroutines 92
F.4 Calculating Subroutines 100
F.5 Informational Text Files 105

APPENDIX G. OBTAINING COPIES OF FIBRAM 109

G.1 Obtaining Copies of the Original FIBRAM Program 109
G.2 Obtaining Information on Updated Versions of FIBRAM . . . 109

LIST OF FIGURES

		Page
Figure 1.	Typical optical fiber link and sublink.	4
Figure 2.	Structured chart of FIBRAM and its subroutines. . .	10
Figure 3.	Flow chart for main program FIBRAM.	12
Figure 4.	Flow chart for subroutine MODATA.	15
Figure 5.	Flow chart for subroutine RUNTIME.	21
Figure 6.	Bit error ratio of an APD 11A photodiode as a function of the receive signal power and radiation dosage (after NCS, 1985).	24
Figure 7.	Linear approximation of Figure 6.	26
Figure 8.	Flow chart for subroutine DARKEN.	31
Figure 9.	Typical free-field nuclear radiation environment (after NCS, 1985).	33
Figure 10.	Induced loss of Sumitomo Standard optical fiber as a function of time and total dosage (Friebele, 1987).	35
Figure 11.	Different values of k used to calculate the "BER verses Time" graph.	37
Figure 12.	"BER verses Time" graph showing the change as a result of a change in input power level. . . .	42
Figure 13.	"BER verses Time" graph using radiation-hardened and unhardened fiber optic cable.	43
Figure 14.	"BER versus Time" graph for AT&T single-mode fiber operating at a wavelength of 1.3 microns in response to various levels of fallout radiation.	45
Figure 15.	"BER versus Time" graph for multimode cable operating at 1.3 microns.	46
Figure 16.	The flow of Figure 15 represented on the graph of the bit error ratio of APD 11E, 11F and 11G photodiodes as a function of the receive signal power and radiation dosage (after NCS, 1985). . . .	48
Figure A-1.	"BER versus Time" graph produced with Lotus 1-2-3 software.	55

LIST OF FIGURES (cont.)

		Page
Figure A-2.	"BER versus Time" graph produced with Chart-Master software.	56

LIST OF TABLES

Table 1.	Subroutines of FIBRAM Divided Into Subcategories	11
Table 2.	Parameters That Can be Modified by the User	16
Table 3.	Digitized Data File BER-11A.AT	27
Table 4.	Format Summary for Data Files to be Read by GRAPHR	28
Table 5.	Calculation Time for Different Values of k	38
Table 6.	Display of FIBRAM's Current Variables When Using Parameters File SM-13.DAT	40
Table 7.	Point-By-Point Analysis of the Flow of Figures 19 and 20	47

A PROGRAM DESCRIPTION OF FIBRAM: A RADIATION ATTENUATION MODEL FOR OPTICAL FIBERS

W. J. Ingram*

This report describes a fiber optics system model and its computer implementation. This implementation can calculate the bit error ratio (BER) versus time for optical fibers that have been exposed to gamma radiation. The program is designed so that the user may arbitrarily change any or all of the system input variables and produce separate output calculations. The primary output of the program is a table of the BER as a function of time. This table may be stored on magnetic media and later incorporated into computer graphics programs. The program was written in FORTRAN 77 for the IBM PC/AT/XT computers. Flow charts and program listings are included in this report.

Key Words: BER; bit error ratio; computer program; gamma radiation; optical fiber; optical fiber communications

1. INTRODUCTION

1.1 Description of the Program

The primary output of program FIBRAM (FIBer optic Radiation Attenuation Model) is a table of data showing the calculated bit error ratio (BER) versus time after exposure of an optical fiber system to gamma radiation from a homogeneous radiation source such as radioactive fallout. This model is partially based on the findings of a particular measurement study conducted for the National Communications System (NCS) (NCS, 1985).

User input will include specific data about the optical fiber link, gamma radiation dosage rates, and the time range of interest. The program will generate bit error ratios using theoretical radiation algorithms, manufacturers' specifications, and experimental graphic results. Each step in calculating the BER is executed in separate subroutines that can be easily modified by the user if more efficient or more accurate methods are needed.

*The author is with the Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U.S. Department of Commerce, Boulder, CO 80303-3328.

1.2 The Scope of this Paper

The purpose of this paper is twofold. First, an overall view of an optical fiber link and sublink is presented. This view includes such attributes as protection factor (PF), intrinsic loss, and loss due to gamma radiation. This information should provide the user with sufficient background to understand the capabilities and limitations of FIBRAM and its sub-programs. Second, the inputs and outputs of the program are described in a way that these subsections may be read and used independently of the remainder of the report. Sample runs presented in Section 4 describe examples of program inputs and the respective outputs.

1.3 System Description

The lightwave digital transmission system that is the basis for NCS TIB 85-11 and, subsequently, this model/program, is the FT3C Northeast Corridor lightwave project. The FT3C lightwave digital transmission system is a medium-to high-capacity short- to long-haul trunk transmission system. The FT3C lightwave system transmits digitally encoded voice and/or data information between terminal or maintenance offices in the form of light pulses at the rate of 90.524 Mb/s. These light pulses are carried over lightguide fibers, frequently referred to as optical waveguides or optical fibers. Up to 1344 voice circuits can be transmitted over a pair of optical fibers at the nominal 90-Mb/s rate (NCS, 1985).

2. BASIS FOR THE MODEL

2.1 Optical Fiber Links

The optical fiber links use direct digital detection of light pulses to convey information between endpoints (called nodes). At the transmitting node, data (usually in electrical form) are converted into pulses of light and sent into the optical fiber. At the receiving node, the received pulses of light are transformed back into a form (also usually electrical) that can be correctly interpreted by a receiving unit such as a computer, telephone, or video display.

A fiber link of any appreciable length will usually contain several electrical regenerating devices along its length. These regenerators amplify optical pulses that may have attenuated. In addition, some of these regenerator devices may also correct the data.

2.2 Optical Fiber Sublinks

The optical fiber path between two adjacent regenerator stations is known as a sublink and is the main focus of this program and report. Each sublink will consist of a transmitting light-emitting diode (LED) at the transmitting end of the fiber, the fiber cable, and a photodiode receiver at the receiving end. This is shown in Figure 1.

The BER of each sublink as a function of time, $BER_{sub}(t)$, is exponentially related to the power received at the photodiode, $P_r(t)$, and the radiation induced losses in the photodiode. This can be expressed as

$$BER_{sub}(t) = fl_{pd} \left[P_r(t),\ \gamma_{pd}(t),\ PF_{pd} \right] \tag{1}$$

where $\gamma_{pd}(t)$ is the gamma dosage rate at the photodiode as a function of time, PF_{pd} is the protection factor at the photodiode, and fl_{pd} is an exponential function (unique to a photodiode type) relating these four quantities.

The function, fl_{pd}, has been shown to be an equation relating the BER to the average optical energy, P, in photons per bit of the form

$$BER = .5\ e^{-2P}. \tag{2}$$

This is a fundamental limit on the bit error ratio and is commonly referred to as the "quantum limit" (Salz, 1985).

The receive signal power at the photodiode is determined by using the following power budget equation:

$$P_r(t) = P_t - L_{in} - L_\Gamma(t) \quad \text{dBm} \tag{3}$$

where

P_t is the power transmitted by the LED, in dBm,
L_{in} is the intrinsic loss, in dB,
$L_\Gamma(t)$ is the loss due to gamma radiation, in dB.

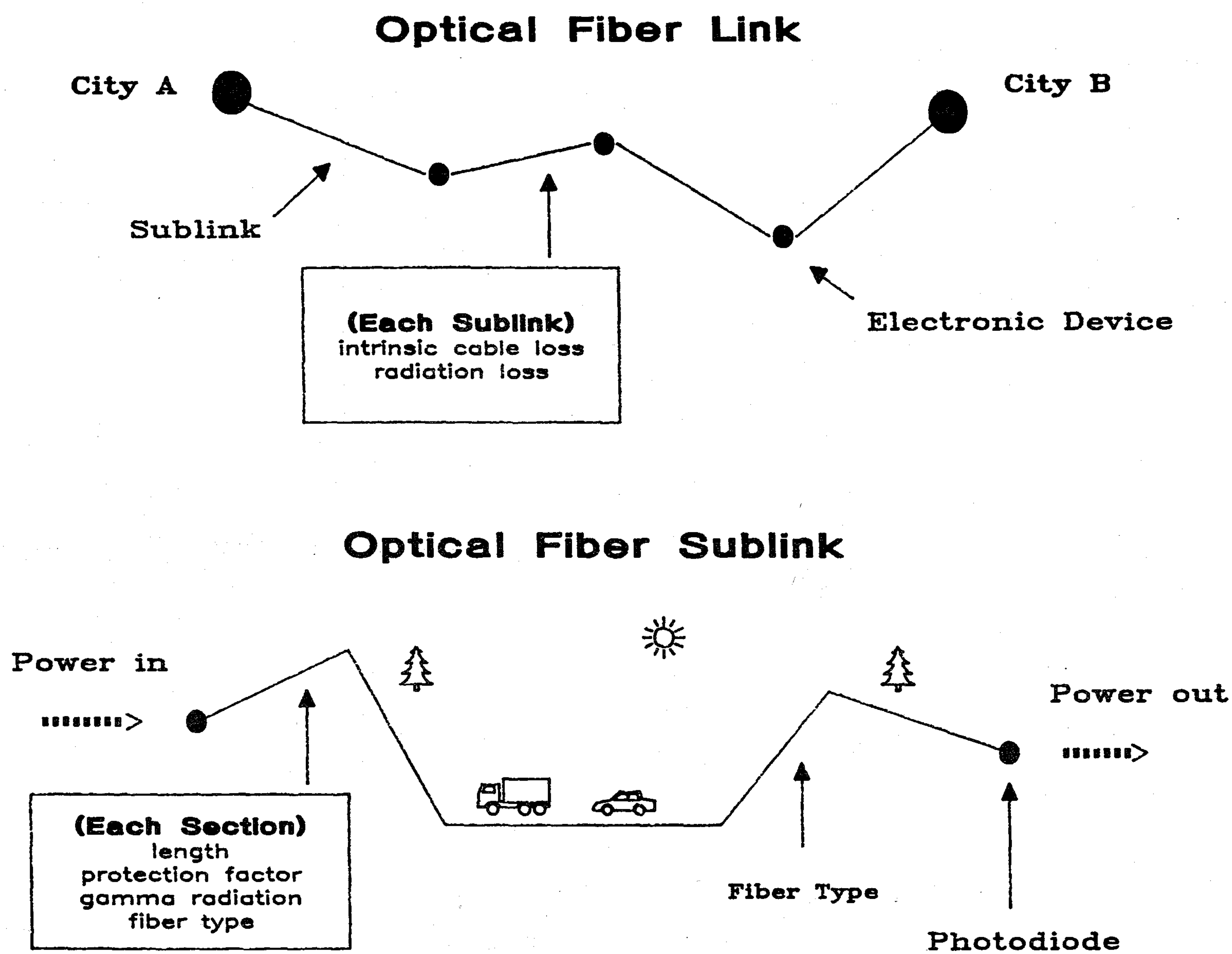

Figure 1. A typical optical fiber link and sublink.

2.3 Optical Fiber Sublink Sections

A fiber's power loss due to gamma radiation is dependent on how much radiation actually reaches the fiber, causing degradation of the signal. Radiation is absorbed by phosphors in the optical fiber, producing ionization. This ionization is composed of free electrons and holes (missing electrons) populating the fiber. The holes and electrons can become trapped at color centers and absorb light energy intended to transmit optical data. This effect is called darkening and is the main focus of this report.

Fiber that is closer to the source of the gamma rays can be expected to receive larger dosages of radiation than fiber that is farther away. When considering a large, homogeneous radiation source, the length of a typical optical sublink is often insignificant when compared to the distance over which radiation levels are reduced appreciably. This assumption allows a single radiation dosage to be distributed along the entire length of the sublink. FIBRAM does, however, allow the user to input different radiation doses for each sublink section if the above assumption is not applicable for specific situations.

Terrain also can have a great effect on the amount of gamma radiation that reaches an optical fiber cable. Cable that is not shielded and is laid on top of the ground with no protection will be exposed to much higher doses of gamma radiation than cable that is buried 3 feet (1 meter) underground. The exposed cable will have a much greater power loss due to radiation darkening than the buried cable will.

The protection factor (PF) of a length of optical fiber is defined to be the ratio of the radiation dosage for a fully exposed fiber $\Gamma_i(t)$, to the radiation dosage for a protected fiber, $\Gamma_i'(t)$. The PF is also defined to be the ratio of the radiation dosage rate for a fully exposed fiber $\gamma_i(t)$, to the radiation dosage rate for a protected fiber, $\gamma_i'(t)$. This is shown as

$$PF_i = \frac{\Gamma_i(t)}{\Gamma_i'(t)} = \frac{\gamma_i(t)}{\gamma_i'(t)}. \qquad (4)$$

Fiber that is not protected and is fully exposed to radiation (free field) is defined to have a PF of 1. Optical fiber buried in 3 feet of earth can have a PF of up to 7000, while fiber installed inside buildings can have PFs of between 2 and 80 (NCS, 1985). On the other hand, optical fiber in areas where radiation can be reflected or focused can have protection factors between 0 and 1.

For increased protection against natural and man-made damage, a typical optical fiber sublink will have most of its length buried at least 3 feet underground. At both ends of any sublink, the fiber is usually brought above ground at a regenerator station and connected to fiber cable that constitutes the next sublink. Along the sublink route itself, there may be several unavoidable situations such as river or creek fording. Each of these incongruities will lend itself to a different protection factor for these sections. In those areas where these incongruities adversely affect the protection factor, different types of optic cable may be used to off-set the radiation darkening effects. Such fibers are known as "radiation-hardened" optic cable. Each length of fiber with a unique protection factor and fiber type is known as a separate sublink section.

For this report, a sublink section is further defined to be the summation, along the entire sublink, of all lengths of fiber that are composed of the same type of optical fiber, have the same protection factor, and are exposed to the same dosage of gamma radiation.

Consider, for example, a optical fiber sublink 5000 meters long containing three separate bridge crossings of 600 meters each and one of 1000 meters. At each of these bridge crossings, radiation-hardened cable is used for extra protection. Also suppose that the entire sublink is exposed to the same maximum gamma radiation dosage of 500 rads. Sublink section #1 could then be considered to be "one" bridge crossing consisting of 2800 meters of "Radiation-Hardened Brand X" cable with a protection factor of 1 and a gamma dosage of 500 rads. Section #2 could be considered to be "one" buried "Regular Brand X" cable 2200 meters long with a protection factor of 7000 and a gamma dosage of 500 rads.

Each sublink section, i, will have four factors associated with it; length, l_i, protection factor, PF_i, fiber type, FT_i, and maximum gamma dosage, γ_{maxi}. Although the maximum gamma dosage can be uniquely specified for each sublink section, the remainder of this report will assume that a constant gamma dosage has been specified for the entire sublink.

2.3.1 Intrinsic Loss of the Fiber

The power loss inherent in the optical fiber cable before exposure to radiation is called the intrinsic loss. This intrinsic loss is dependent only upon the type of fiber being used and the length of each sublink section. By definition, it is unaffected by gamma radiation. A sublink section is assumed to be a length of optic cable with its own individual intrinsic factor. The intrinsic loss of the entire sublink can then be expressed as the summation of the intrinsic losses of each sublink section as follows:

$$L_{in} = \sum_{i=1}^{n} T_i \, l_i \tag{5}$$

where n is the number of sublink sections, T_i is the intrinsic loss factor for section i (decibels/kilometer), and l_i is the length of section i (kilometers). The intrinsic loss factor is unique for each type of fiber being used.

2.3.2 Darkening Loss of the Fiber

As optical cable is exposed to gamma radiation, the fiber will absorb some of the radiation, causing ionization to occur. The resulting free electrons and holes will act as defects, absorbing some of the light energy of the transmitted signal pulses. This is known as the darkening loss of the fiber. The darkening loss of the entire sublink, $L_\Gamma(t)$, is equal to the summation of the darkening losses of each of the sublink sections, $L_{\Gamma i}(t)$, and is shown as

$$L_\Gamma(t) = \sum_{i=1}^{n} L_{\Gamma i}(t). \tag{6}$$

The darkening loss at time, t, in each of the sublink sections, i, is related to the type of fiber being used, FT_i, the protection factor, PF_i, the length, l_i, the total accumulated gamma radiation at that section, $\Gamma_i(t)$, and the current gamma dosage rate, $\gamma_i(t)$. This is represented as

$$L_{gi}(t) = f2 \left[\gamma_i(t) \ , \ \Gamma_i(t) \ , \ l_i \ , \ PF_i \ , \ FT_i \right] \tag{7}$$

where f2 is a function relating these values, $L_{gi}(t)$ is the actual radiation darkening loss, and

$$\Gamma_i(t) = \int_0^t \gamma_i(\tau) \, d\tau . \tag{8}$$

By combining (7) and (4) we get

$$L_{gi}(t) = f3 \left[\gamma_i'(t) \ , \ \Gamma_i'(t) \ , \ l_i \ , \ FT_i \right] \tag{9}$$

where f3 is a function relating these values. The function f3 is an experimentally-determined function that will be discussed later in this report.

The free electrons and holes created during ionization of the fiber will, given enough time, recombine. This will cause a lessening of the darkening effect. This phenomenon is called the recovery factor, $R_i(t)$, and is a function of the time, the history of the accumulated radiation dosage, and the current exposed dosage rate. This variable is dimensionless, normalized to a value of 1 at time, 0, and is represented as

$$R_i(t) = f4 \left[t, \ \gamma_i'(t) \ , \ \Gamma_i'(t) \ , \ FT_i \right] \tag{10}$$

where f4 is a function relating these values. The function f4 is a pseudo-exponential function that is continually decreasing with time (NCS, 1985).

2.3.3 Total Darkening Losses

The darkening loss of each sublink section is expressed as

$$L_{\Gamma i}(t) = l_i \ L_{gi}(t) \ R_i(t) \tag{11}$$

where $L_{\Gamma i}(t)$ is the darkening loss of the sublink section in dB/meter. The factors $L_{gi}(t)$ and $R_i(t)$ are often difficult to determine independently and are usually graphically expressed as one combined variable, $L_{Gi}(t)$. For this report it is assumed that the user will provide loss data expressed as the single variable $L_{Gi}(t)$.

The total darkening loss of the sublink can then be expressed as

$$L_\Gamma(t) = \sum_{i=1}^{n} l_i \, L_{Gi}(t). \tag{12}$$

The results of (5) and (12) can then be substituted into (3) to yield the expected power at the receiving end of the sublink. The modified power budget equation is shown as

$$P_r(t) = P_t - \sum_{i=1}^{n} T_i \, l_i - \sum_{i=1}^{n} l_i \, L_{Gi}(t) \tag{13}$$

3. HOW FIBRAM WORKS

FIBRAM is composed of many different subroutines working together to produce the bit error ratio as a function of time. Figure 2 shows the program structure of the FIBRAM subroutines. Each of the subroutines is fully explained in this report and is separated into one of three descriptive categories; framework, utility, or calculating. This is shown in Table 1.

Each FIBRAM subroutine is based on current theories and data. If the user discovers a more accurate method of calculating one or more of these variables, new subroutines can easily be written and substituted into FIBRAM.

3.1 Framework Subroutines

The framework subroutines used by FIBRAM provide the necessary input/output support, but do very little calculating. They are designed to aid the user in entering, modifying, and using data.

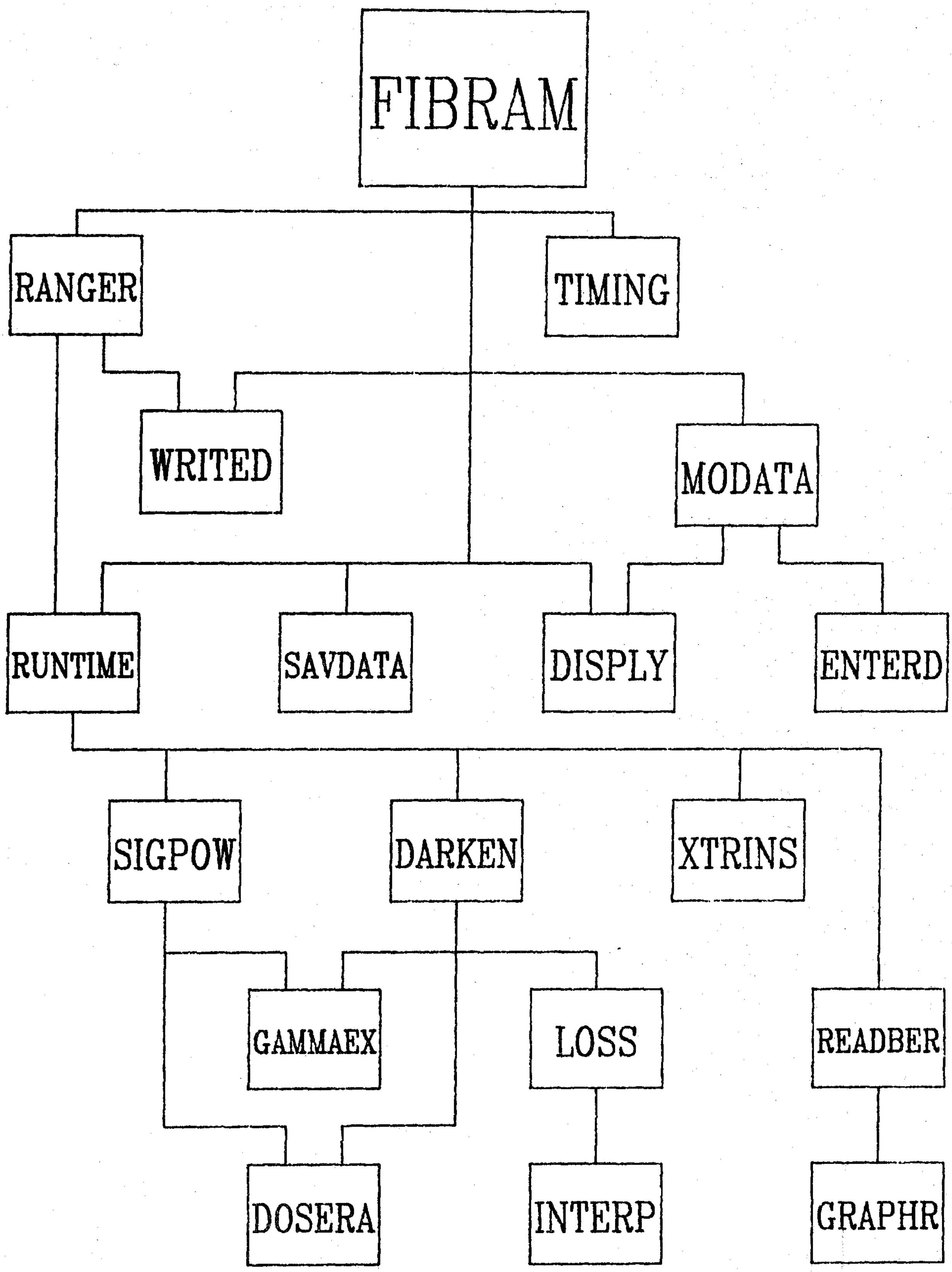

Figure 2. Structure chart of FIBRAM and its subroutines.

Table 1. Subroutines of FIBRAM Divided Into Subcategories

Framework Subroutines	Section	Appendix Listing
FIBRAM	3.1.1	F.1
TIMING	3.1.2	F.2.7
MODATA	3.1.3	F.2.3
ENTERD	3.1.4	F.2.1
SAVDATA	3.1.5	F.2.6
GETDATA	3.1.6	F.2.2
RANGER	3.1.7	F.2.4
RUNTIME	3.1.8	F.4.5
Utility Subroutines		
ENTERA	3.2.1	F.3.2
ENTERX	3.2.1	F.3.3
FREAD	3.2.2	F.3.4
DISPLY	3.2.3	F.3.1
PRDATA	3.2.4	F.3.5
LOGDATA	3.2.5	F.3.9
GRAPHR	3.2.6	F.3.7
FTYPES	3.2.7	F.3.5
GRAPHIN	3.2.8	F.3.6
INTERP	3.2.9	F.3.8
Calculating Subroutines		
DARKEN	3.3.1	F.4.1
XTRINS	3.3.2	F.4.7
DOSERA	3.3.3	F.4.2
GAMMAEX	3.3.4	F.4.3
LOSS	3.3.5	F.4.4
SIGPOW	3.3.6	F.4.6
READBER	3.3.7	F.4.5

3.1.1 Main Program FIBRAM

The main program segment called FIBRAM is the home base for all other FIBRAM operations. Its flowchart is shown in Figure 3. In addition to the main menu, FIBRAM contains the input routine for each menu item selection. When the user runs FIBRAM, the main menu is called up and the user is prompted to select an item. After any menu selection has been selected and executed, FIBRAM returns program control to the main menu and waits for the next menu command.

From main program FIBRAM, the user is able to access subroutines that can load, save, or modify system variables; run or exit the model; and send the model results to the video screen or a disk file.

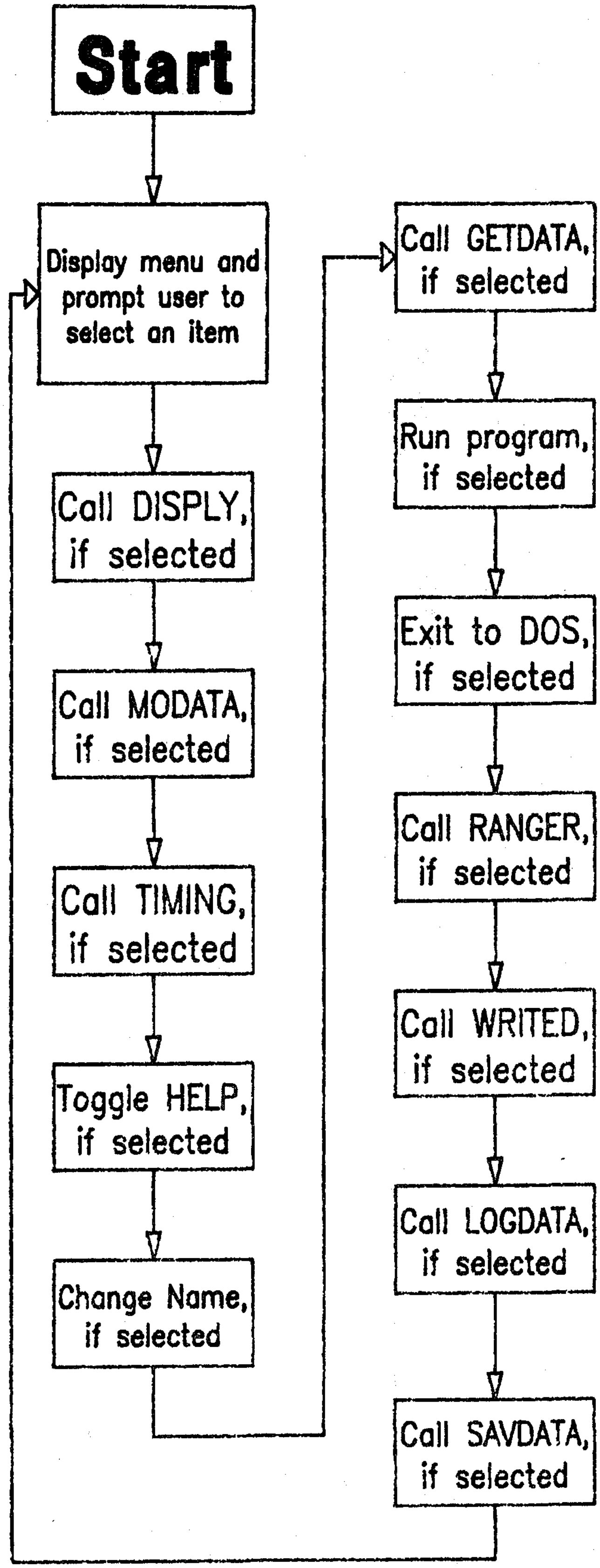

Figure 3. Flow chart for main program FIBRAM.

FIBRAM allows the user access to three bookkeeping aids. These are the HELP feature, the AUTODISPLAY feature, and the NAME feature.

The HELP feature may be toggled ON or OFF. When HELP is toggled ON, many parts of FIBRAM will call the subroutine FREAD, which will then convey to the video screen any information available useful to the subroutine currently in use. With the help function OFF, this additional information will be skipped.

The AUTODISPLAY feature may also be toggled ON or OFF. This feature allows the user to see the internal calculations of FIBRAM as it executes a set of variables. This can be useful in identifying parameter values that cause the model to go out of range. When AUTODISPLAY is toggled OFF, the model will run about 20 percent faster.

For the convenience of the user, the NAME feature is included so that the user can identify previously saved sets of variables. Naming each set of variables is not necessary for the correct operation of program FIBRAM. If the user has given a name to a set of variables, it is saved to the disk during execution of the SAVDATA subroutine.

3.1.2 Subroutine TIMING

Subroutine TIMING is the subroutine from which the user enters the timing information for the model. When this subroutine is accessed, the user is first asked for the starting time of the model (seconds). The user is then asked for the model ending time (seconds). The ending time must be larger than the starting time or FIBRAM will assume the user is confused and display the help feature associated with timing and prompt the user to reenter those values.

After the start and end times are entered, the user will then be asked for the offset time and the incremental time step factor. An offset time of 500 seconds, for example, would be used if the user wished to examine the model beginning at 500 seconds after initial irradiation rather than from the time of the irradiation itself. FIBRAM's output to the user would then begin at time 0, even though this point is actually 500 seconds after detonation.

The incremental time step factor is used to determine the times that FIBRAM will model the fiber optic system. Since the typical bit error ratio response of a fiber optic system exposed to radiation is logarithmic over time, a pseudo-logarithmic input time scale is utilized. If the user has chosen an incremental time step factor of 0, for example, the input time scale will begin at 10 to the power of 0 (i.e., 1) and progress as follows:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 40, 50, 60, etc.

until the end time, which the user has earlier specified, is surpassed.

If, for example, the user has chosen the following variables:

Start Time = 500 seconds
End Time = 550 seconds
Offset Time = 500 seconds
Incremental Time Step = 0 (Step Size = 1 second),

the system would be modeled at the actual times 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 520, 530, 540, and 550 seconds. The resultant output times would be offset by 500 seconds and would be listed as 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 40, and 50 seconds. This more clearly shown as

Actual Time BER was calculated (seconds)	Outputted Time (seconds) (Offset is 500 seconds)
501	1
502	2
503	3
504	4
505	5
506	6
507	7
508	8
509	9
510	10
520	20
530	30
540	40
550	50.

3.1.3 Subroutine MODATA

Subroutine MODATA provides the user with a method to change most of the system variables. Its flowchart is shown in Figure 4. The system variables

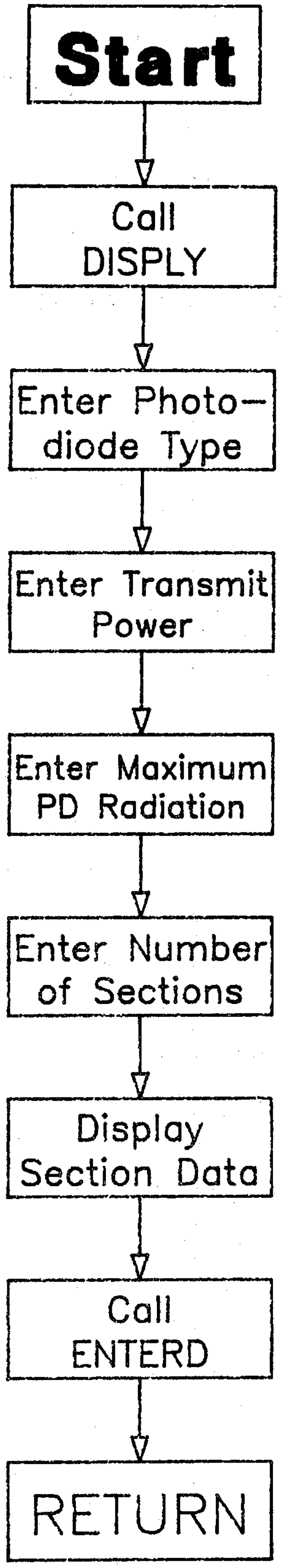

Figure 4. Flow chart for subroutine MODATA.

Table 2. Parameters That Can be Modified by the User

- The Name for the System Variables
- The Help Function (ON/OFF)
- The Autodisplay Function (ON/OFF)
- Photodiode Type
  - Maximum Gamma Radiation Exposure That the Photodiode Can Survive
- Timing Parameters
  - Starting Time
  - Ending Time
  - Offset Time
  - Incremental Time Step Factor
- Number of Model Sections
  - Length of Each Section
  - Protection Factor of Each Section
  - Type of Optical Fiber
  - Intrinsic Loss Factor of the Fiber Type
  - Maximum Gamma for Each Section

are listed in Table 2. When accessed, MODATA will display a listing of the current values of each of the system variables by calling subroutine DISPLY (Section 3.2.3). The user is then prompted to change each of the variables. The user should respond by typing in a value for each of the variables. Be sure to use a decimal point in all numeric inputs otherwise Fortran will incorrectly interpret your input. If, at any of these prompts, the user types <RETURN>, the current value, as shown, is retained.

Subroutine MODATA calls subroutine ENTERD (Section 3.1.4) to allow the user to modify the system variables pertaining to each individual sublink section. These would be the length, protection factor, fiber type, and the maximum gamma radiation to which each section will be exposed. The user should input these individual values in the same manner as in the rest of subroutine MODATA.

Subroutine MODATA is designed to handle many different kinds of fiber and photodiodes. The current list of optical fiber types and photodiode types are contained in text files FIBTYPES.TXT and PHOTYPES.TXT, respectively. If the user discovers the need for more types, FIBRAM's files may be modified to reflect these new types. The user may consult Appendix B for further details.

3.1.4 Subroutine ENTERD

Subroutine ENTERD is called by subroutine MODATA and allows the user to modify the individual variables for each sublink section. These variables are length, protection factor, type of fiber, intrinsic factor of that fiber type, and maximum exposed gamma dosage for the region.

ENTERD will first display on the screen the number of sections (including an extra section labeled "photodiode") and the current values of the variables associated with each section. The subroutine will then prompt the user to enter the number of the section whose variables are to be changed. If the user types a negative number, ENTERD will assume that all section variables are to be changed, beginning with the first one and proceeding sequentially through each sublink section.

If the user types <RETURN> at ENTERD's prompt to change a variable value, the current value will remain unchanged.

ENTERD makes the major assumption that each section along the entire sublink will be exposed to the same dosage of gamma radiation. When a dosage is entered by the user for one section, ENTERD assumes that radiation dosage value for any subsequent sections to be modified. In these subsequent sections, the user may affirm this dosage value by hitting <RETURN> at the prompt to enter a new value. The user can, of course, override this feature simply by manually entering a different amount of gamma radiation for each sublink section.

Consider that the user has typed '-1.' <RETURN> (Note the decimal point as part of the input) to indicate a desire to modify all variables of all sections. The user then entered 500.0 rads for the amount of gamma radiation that section #1 is exposed to. When the user is subsequently prompted to enter a value for the amount of radiation for section #2, 500 rads will be displayed as the current value, although no previous amount had yet been entered for section #2's dosage. If the user wishes to use 500 rads, the <RETURN> key is pressed and the current value (500 rads) will be accepted.

The gamma radiation dosage that each section is exposed to is assumed to remain constant over the relatively short distance of an optical fiber sublink, as compared to the greater distance which cause gamma radiation to attenuate appreciably.

3.1.5 Subroutine SAVDATA

Subroutine SAVDATA will save, on a disk file, the system variables that the user has specified. These variables are shown in Table 2. Once the user modifies the data and chooses 's' <RETURN> from the main menu, the user will be prompted to provide a file name in which to save these variables. Variables saved by this subroutine can be recalled by using subroutine GETDATA (below) and can be edited with any text-editing program. Standard notation for this file name is one that has an extension ".DAT" (as in the file name DEFAULT.DAT).

3.1.6 Subroutine GETDATA

Subroutine GETDATA is a subroutine that is parallel to subroutine SAVDATA with the exception that GETDATA loads the variables from disk where SAVDATA saves the variables to disk. To access this module from the main menu, the user should type 'l' (as in "load") <RETURN>. The variables are the same as those in SAVDATA (Section 3.1.5) and are shown in Table 2. Both subroutines use the same format when accessing the disk-based data.

3.1.7 Subroutine RANGER

The essence of FIBRAM is included within the subroutine RUNTIME (Section 3.1.8) and can be accessed either directly through the keyboard from the main menu or indirectly by using the subroutine RANGER.

Subroutine RANGER will allow the user to model an optical fiber sublink's response to the "ranging" of particular variables. This can be done by assigning this variable different values while keeping all the other parameters constant.

Subroutine RANGER is capable of "ranging" the following parameters:

Maximum Amount of Exposed Gamma Radiation

Intrinsic Cable Loss

Maximum Transmitted Power.

When the user activates subroutine RANGER (by typing '>' <RETURN> from the main menu), there are two steps involved. The first step is performed by the user and involves answering questions asked by subroutine RANGER. This information will tell RANGER which parameters file (saved earlier by the SAVDATA subroutine) to load, which variable to vary, and where to save any results.

RANGER first asks the user to enter the file name containing the parameter values that are not to be "ranged." Since a value for every parameter is included in this file, a value for the "ranged" variable will also be included. This value will be ignored at run time in deference to its "ranged" value. RANGER will then prompt the user to choose which variable is to be ranged and to type in its corresponding numeral. If the <RETURN> key alone is pressed, the user is returned to the main menu.

After the user selects a variable, RANGER asks the user to enter the number of different values that the chosen variable will assume. It will then ask the user to enter these values one at a time followed by the output file name for the results of that value. After all the above data have been entered, the user is then asked whether this arrangement of values and file names should be saved in a different file for later use. If so, the user is prompted for a new file name, which should have the extension ".RNG". This file can be subsequently modified by the user with any text editing package.

When the user decides to use previously stored values and file names (i.e., with the extension ".RNG"), the user should type '0.' (zero) <RETURN> when asked for the number of different value that the chosen variable will assume. This will cause RANGER to ask for the name of a previously saved file containing these values and file names.

In the second step, the computer will automatically run the base variable file once for each subsequent value of the specified variable. The results of each run are automatically saved using subroutine LOGDATA (Section 3.2.5) in the disk file specified earlier by the user in the first step.

This second step is performed completely by the computer and requires no further input from the user. This step can take anywhere from 2 minutes to several hours to perform, depending upon the chosen parameter values.

3.1.8 Subroutine RUNTIME

Subroutine RUNTIME is the centerpiece of FIBRAM. It calls, in the appropriate order, the other calculating subroutines to arrive at the correct bit error ratio for the current variables and specified time. Its flowchart is shown in Figure 5. RUNTIME is also responsible for incrementing the timing variable for the model.

RUNTIME can be called by the user directly from the main menu by typing 'r' <RETURN> or indirectly when the user invokes the RANGER option from the menu.

When RUNTIME is executed, it calculates the bit error ratio of the optical fiber sublink by determining the darkening factor and then "reading" a graph relating the signal power loss and radiation rate to the bit error ratio.

When it is first executed, RUNTIME sets up the main timing loop. Each time that FIBRAM returns program control to RUNTIME, the timing loop is incremented up to the end time (as specified earlier by the user).

During each pass through the timing loop, RUNTIME calls the subroutines DARKEN (Section 3.3.1), SIGPOW (Section 3.3.6), and READBER (Section 3.3.7). DARKEN calculates the power loss caused by darkening of the fiber as a result of gamma irradiation. SIGPOW calculates the overall receive signal power at the photodiode by subtracting each individual power loss (including the darkening loss previously calculated) in dB from the transmitted power in dBm. READBER uses linear approximation methods to determine the bit error ratio from the receive signal power and the radiation exposure rate at the photodiode.

Once the bit error ratio has been determined for a particular time, it and the time it was calculated (minus the offset time) are stored in an array. The timing loop is then incremented and a new BER is calculated. When the model is finished, this array of data can be written to an ASCII disk file, displayed on the video screen, or printed out on paper.

3.2 Utility Subroutines

Each of the following utility subroutines is generally called by other subroutines to perform one specific task and then to return to the main program flow. For example, ENTERA is used to accept one string of ASCII characters from the keyboard. It is used throughout FIBRAM whenever a file name needs to be entered by the user.

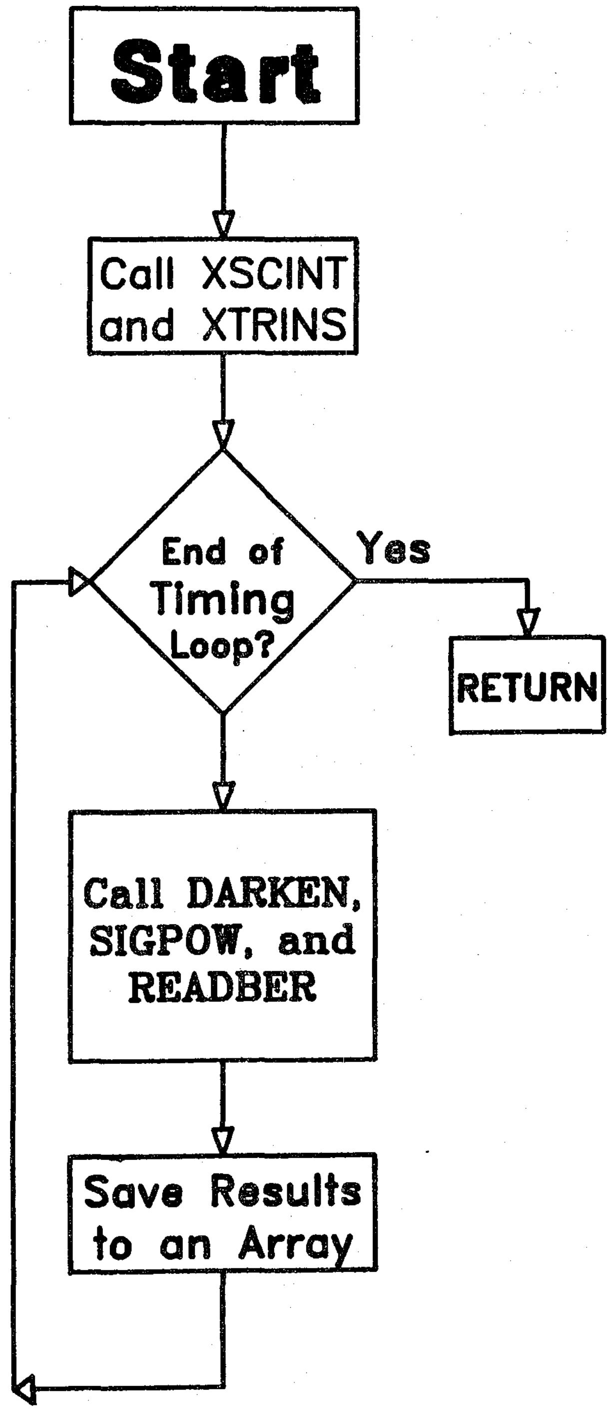

Figure 5. Flow chart for subroutine RUNTIME.

Standard Fortran functions used by FIBRAM, such as ALOG10 and SQRT, are not explicitly described in this manual. The user is advised to consult a standard Fortran manual for further details on the use of these functions (CDC, 1985).

3.2.1 Subroutines ENTERA and ENTERX

These simple subroutines each accept a single variable input from the keyboard. ENTERA accepts an ASCII character string while ENTERX accepts a numerical, real variable.

Each of these two subroutines is called many times from numerous places within FIBRAM.

3.2.2 Subroutine FREAD

Subroutine FREAD ("file read") is called primarily when the help function is toggled ON. It is used to display informational text about the currently activated subroutine. It will search the current disk directory for a file containing the specified information. Once it has found the file, it will transfer the contents, line by line, onto the screen for the user to read.

Files that can be read by FREAD have the extension ".TXT". The following files are included on this disk and in Appendix F.5:

FIBTYPES.TXT	INTRO.TXT	MAINMENU.TXT
NSECTION.TXT	PHOTYPES.TXT	RANGE.TXT
TIME.TXT	WORST.TXT.	

When the user finds a need to update one of these informational files or to add a completely new one, any word processing package or screen editor can be used to create/edit the file.

3.2.3 Subroutine DISPLY

Subroutine DISPLY will print to the screen the current values of the variables listed in Table 2. This subroutine can be called directly by typing 'd' at the main menu or indirectly by calling subroutine MODATA (Section 3.1.3).

3.2.4 Subroutine PRDATA

Once the user has run the model (by typing 'r' <RETURN> from the main menu), the results of that run are present in volatile RAM and will be lost when the program is exited or power is removed from the computer. These results can be viewed on the user's video screen by accessing subroutine PRDATA ("print data") from the main menu by typing 'p' <RETURN>. This will provide the user with a column of data showing the bit error ratio of the model versus the output time (note that this time value is the actual time minus the offset time).

3.2.5 Subroutine LOGDATA

Subroutine LOGDATA performs essentially the same function as subroutine PRDATA (Section 3.2.4) with the exception that the results are "printed" into an ASCII disk file rather than onto the video screen. The user is prompted for the name of a disk file on which to write these results. If the user provides a file name with an extension ".PRN", the file is known as a "print file," and can be easily incorporated into most commercially available graphical charting programs.

Subroutine LOGDATA does not "print" the actual values of the output time or the BER, rather it "prints" the logarithm (base 10) of those values. Since logarithms of nonpositive numbers are not real numbers, the user should be careful when specifying the "offset time" variable to make sure that all output times are positive.

This feature can be invoked from the main menu by typing 'g' <RETURN>.

3.2.6 Subroutine GRAPHR

Subroutine GRAPHR ("graph read") is a subroutine to allow FIBRAM to estimate values from data represented in graphical form. These data usually are very nonlinear and cannot be accurately represented by simple equations. These data used in FIBRAM are usually three dimensional and can be graphically presented as a family of curves on a two-dimensional graph. Figure 6 is an example of this kind of graph.

Most data points on such a graph will not lie exactly on one of the family of curves. This means that value must be approximated. The first step for properly approximating between these graphic values is to digitize the graph.

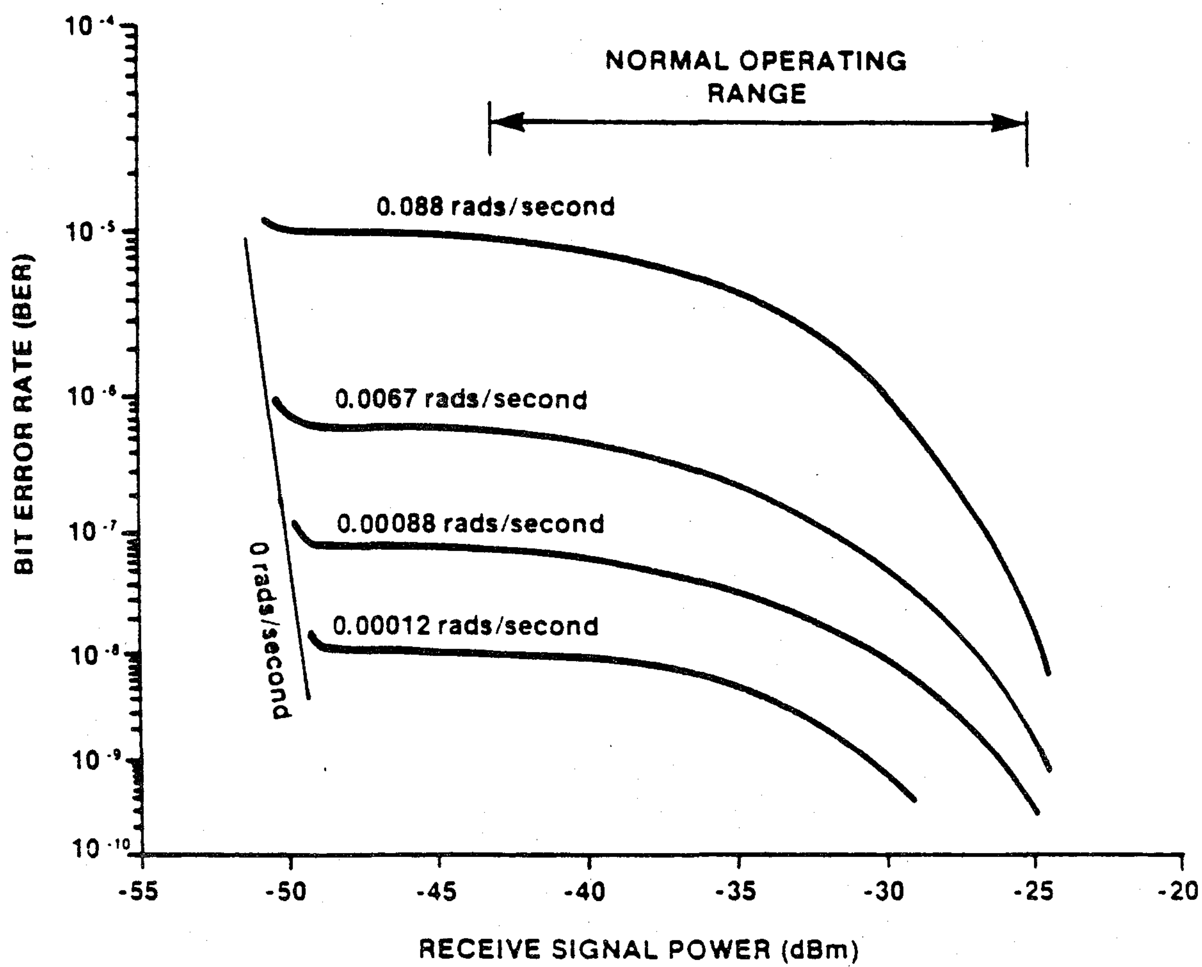

Figure 6. Bit error ratio of an APD 11A photodiode as a function of the receive signal power and radiation dosage (after NCS, 1985).

The digitized data must be in the format that GRAPHR understands. The format for doing this is described below.

GRAPHR can only be used to interpolate on three-dimensional graphs. Figure 6 shows a family of curves representing the bit error ratio (the "z" coordinate) of an APD 11A photodiode as a function of receive signal power ("x" coordinate) and radiation dosage rate ("y" coordinate).

Figure 7 shows a simple, straight line segment approximation of Figure 6. These straight line segments are then translated into data that can be properly interpreted by FIBRAM. Table 3 contains the actual digitized data file that GRAPHR will read to interpolate Figure 7, thus closely approximating Figure 6. More accurate digitization can be performed and entered by the user if it seems necessary.

The first line of the data file contains three data values and is shown as

```
   5 .100000E-20 .880000E-01                                        (14)
+----+----+----+----+----+----+
0    5    10        20        30
```

Columns 1 through 4 of (14) represent the number of curves along the "y" axis. In this example, this number is 5 and is entered as three spaces and the numeral "5". Columns 5 through 16 contain the value of the lowest curve in the "y" axis family. In this case, the value of the smallest curve would be zero. Since a value of absolute zero is unattainable for radiation dosage rate, a value of 1 x 10^{-19} is assumed. This is entered as " .100000E-20".

Columns 17 through 28 contain the value of the largest curve in the "y" axis family. In this case, the value of the largest curve would be .088 rads/second. This is entered as " .880000E-01".

The second line of the data file is shown as

```
 .100000E-20   3-.525000E+02-.200000E+02                            (15)
+----+----+----+----+----+----+----+----+
0    5    10        20        30        40
```

and contains data values pertaining to the first (smallest) curve in the "y" axis family. Columns 1 through 12 contain a real number identifying this first curve in the "y" axis family. In this case the number is 1 x 10^{-19}.

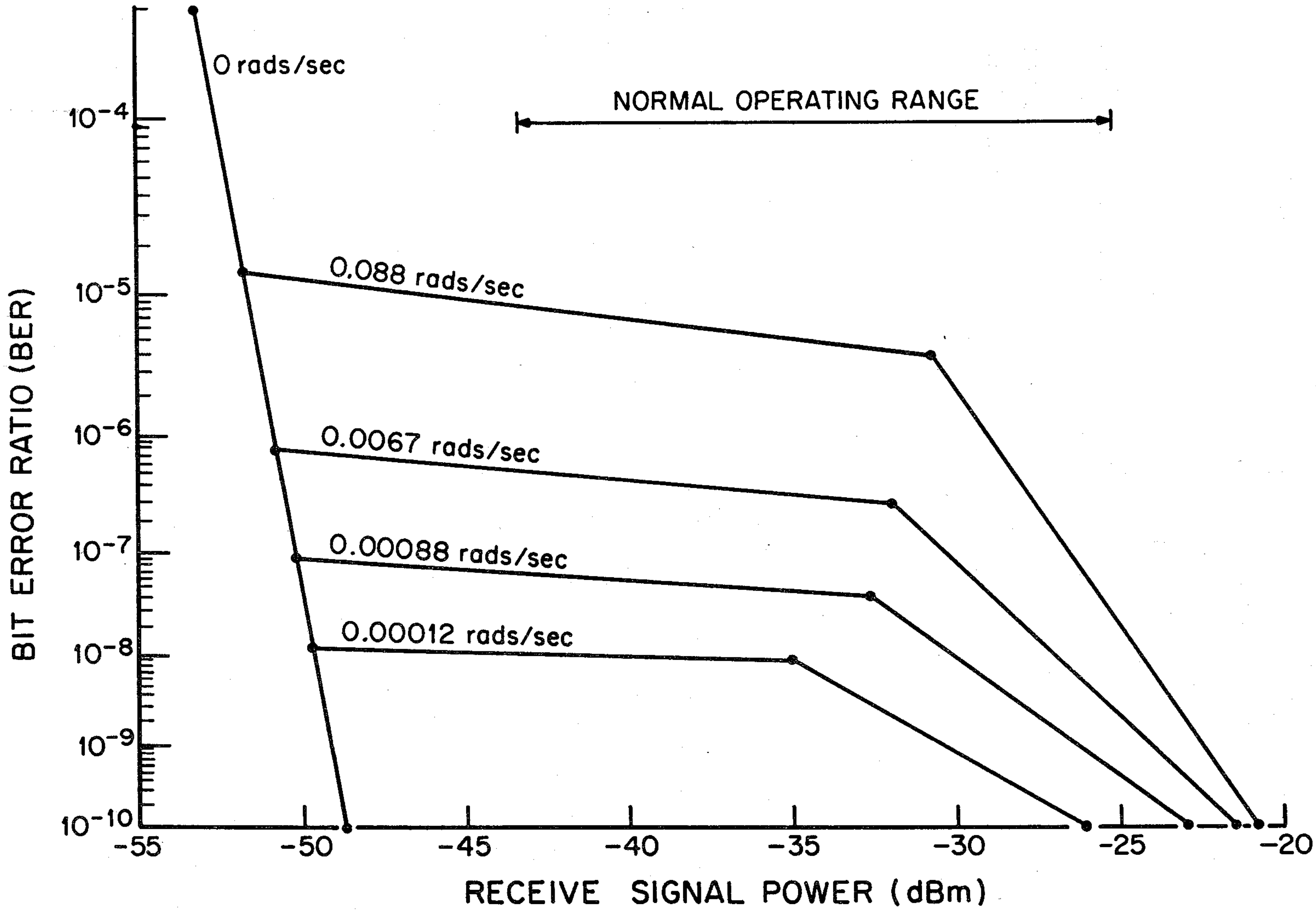

Figure 7. Linear approximation of Figure 6.

Table 3. Digitized Data File BER-11A.AT

```
  5 .100000E-20 .880000E-01
 .100000E-20    3-.525000E+02-.200000E+02
-.525000E+02 .200000E-03
-.485000E+02 .100000E-09
-.215000E+02 .100000E-09
 .120000E-03    5-.525000E+02-.215000E+02
-.525000E+02 .200000E-03
-.495000E+02 .140000E-07
-.350000E+02 .700000E-08
-.260000E+02 .100000E-09
-.215000E+02 .100000E-09
 .880000E-03    5-.525000E+02-.215000E+02
-.525000E+02 .200000E-03
-.500000E+02 .900000E-07
-.327000E+02 .400000E-07
-.230000E+02 .100000E-09
-.215000E+02 .100000E-09
 .670000E-02    5-.525000E+02-.215000E+02
-.525000E+02 .200000E-03
-.505000E+02 .800000E-06
-.320000E+02 .300000E-06
-.220000E+02 .100000E-09
-.215000E+02 .100000E-09
 .880000E-01    4-.525000E+02-.215000E+02
-.525000E+02 .200000E-03
-.515000E+02 .120000E-04
-.305000E+02 .400000E-05
-.215000E+02 .100000E-09
```

Columns 13 through 16 of (15) contain an integer representing the total number of points in the specific curve being modeled. In this case, the curve identified as 1×10^{-19} has three points in it. Columns 17 through 28 and 29 through 40, respectively, are real numbers identifying the minimum and maximum "X" values for this curve.

The third through fifth lines of the data files are shown as

```
 -.525000E+02 .200000E-03                                        (16)
 -.485000E+02 .100000E-09
 -.215000E+02 .100000E-09
+----+----+----+----+----+----+
0    5    10        20        30
```

and they identify the actual data points that constitute the linear approximation of the curve. Columns 1 through 12 of each of these lines represent the "x" coordinate of the point, and columns 13 through 24 represent the "z" coordinates. The points shown in this example correspond to (-52.5, 2 x 10^{-4}), (-48.5, 1 x 10^{-10}), and (-21.5, 1 x 10^{-10}).

The formatting technique used for lines 2 through 5 is repeated four times to include the rest of the family of "y" axis curves.

A summary of these unique formatting techniques is given in Table 4.

Table 4. Format Summary for Data Files to be Read by GRAPHR

Line No.	Columns	Type	Variable Description
1	1 - 4	Integer	Number of curves each with a discrete "y" value associated with it
1	5 - 16	Real	Minimum "y" value
1	17 - 28	Real	Maximum "y" value
2	1 - 12	Real	The "y" value for the first curve in the "y" axis family of curves
2	13 - 16	Integer	Number of points constituting the straight-line approximation of curve "y"
2	17 - 28	Real	Minimum "x" value along curve "y"
2	29 - 40	Real	Maximum "x" value along curve "y"
3-5 (The number of lines is equal to the number of points for curve "y" that was specified in line 2)	1 - 12	Real	The "x" coordinate of each point
3-5	13 - 24	Real	The "z" coordinate of each point
6-?	Repeat lines 2-5 for each new curve "y" until the maximum "y" value specified in line 1 is reached.		

3.2.7 Subroutine FTYPES

Subroutine FTYPES allows the user to easily enter data for new fiber types. The actual procedure is described further in Appendix B.2.1. Subroutine FTYPES calls subroutine GRAPHIN (Section 3.2.8) to enter each data value and subroutine INTERP (Section 3.2.9) to allow the user to test the newly-entered data.

3.2.8 Subroutine GRAPHIN

Subroutine GRAPHIN is called by subroutine FTYPES (Section 3.2.7). It will prompt the user to enter each new pair of data points. The data points must be entered in ascending order (as arranged by the "X" coordinate). Any "X" coordinate that is not in ascending order will indicate to GRAPHIN that the user is finished entering data points and is ready to store the file. The user must then enter a file name in which to store these data.

3.2.9 Subroutine INTERP

Subroutine INTERP is used to approximate a data point from a data file. The data in the file is stored as a two dimensional array representing an "X" coordinate and a "Y" coordinate. The inputs to subroutine INTERP would be the "X" coordinate and the file name of the data file. INTERP will use linear approximation to determine the corresponding "Y" coordinate. This value and the dose rate at which the data file was measured will be outputs of subroutine INTERP.

Subroutines LOSS (Section 3.3.5) and GRAPHIN (Section 3.2.8) both use INTERP to approximate data.

3.3 Calculating Subroutines

The following subroutines are the workhorses of FIBRAM. They perform the calculation of the variables involved with determining how well the optical fiber link will survive gamma radiation. These subroutines can be called hundreds or thousands of times during each model run.

Each calculating subroutine is specific to one particular variable of the model. Each subroutine is based on current theories and data. If the user discovers a more accurate method of calculating one or more of these variables, new subroutines can easily be written and substituted into FIBRAM. This is explained further in Appendix B.

3.3.1 Subroutine DARKEN

Subroutine DARKEN is called by RUNTIME in order to calculate the total darkening of the optical fiber cable in response to gamma radiation. Its flowchart is shown in Figure 8. This amount of darkening is dependant upon the length, loss factor, and recovery factor of each section of the sublink. This was shown earlier in (12).

To calculate the loss and recovery factors, DARKEN first needs to determine the current gamma radiation exposure rate and the total accumulated radiation dosage. This is done by calling subroutines DOSERA (Section 3.3.3) and GAMMAEX (Section 3.3.4) respectively. After these values are calculated, DARKEN determines the loss and recovery factors by calling subroutine LOSS (Section 3.3.5). As explained earlier, the loss and recovery factors are not individually calculated, but are determined as one composite variable.

DARKEN then uses (12) to calculate the total fiber darkening in decibels. DARKEN will print these intermediate values to the video screen if the AUTODISPLAY feature has been toggled ON. This printing of intermediate values can be helpful to the user when tracking down results that seem unusual.

Once the darkening factor has been calculated, program control is returned to RUNTIME to finish calculating the BER and to increment the timing loop.

3.3.2 Subroutine XTRINS

Subroutine XTRINS calculates the intrinsic power loss, in dB, of the cable before it is exposed to radiation. This loss occurs because the glass is not perfect and contains unavoidable irregularities which can attenuate the signal. The loss will appear as a constant value, unique to the type of fiber being used, regardless of the amount of radiation exposure to the fiber.

The intrinsic loss is simply calculated by multiplying the total length of each sublink section in kilometers by its intrinsic loss factor in decibels per kilometer. This was shown in (5).

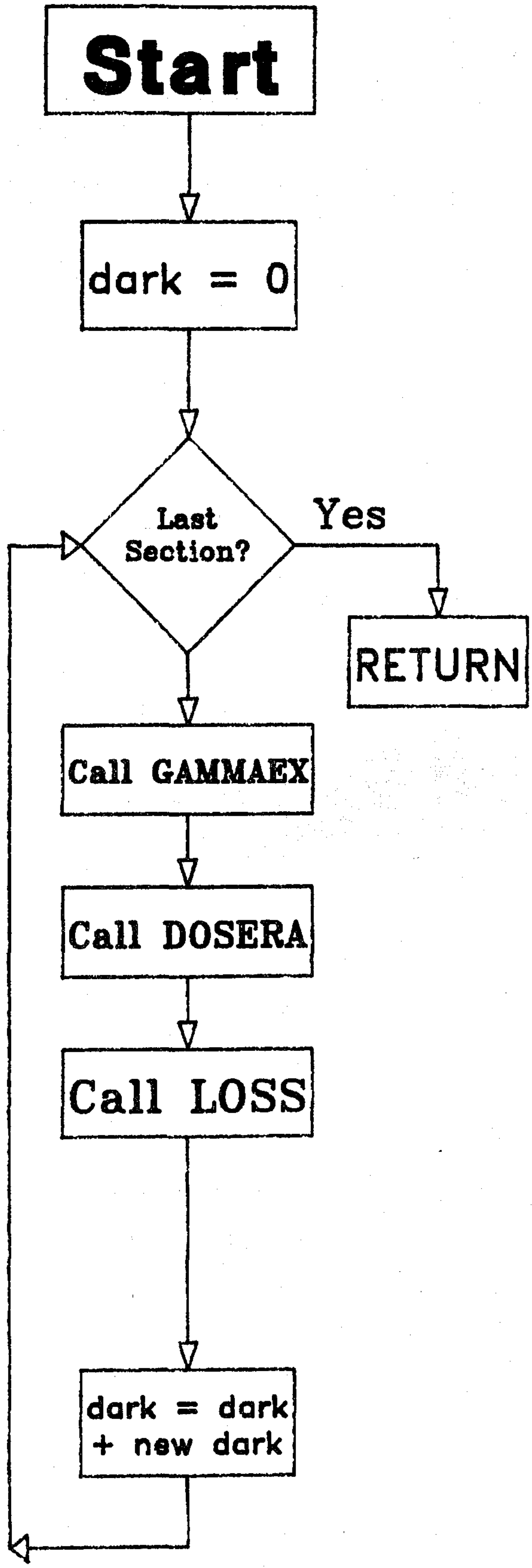

Figure 8. Flow chart for subroutine DARKEN.

3.3.3 Subroutine DOSERA

Subroutine DOSERA calculates the current dosage rate of fallout gamma radiation to which the fiber is being exposed. For an endoatmospheric nuclear detonation, the free-field gamma radiation dosage rate, as a function of time, can be approximated by

$$\gamma(t) = \begin{cases} 0 & t_0 > t \\ \gamma_{max}\,(t_0/t)^{1.2} & t_0 \le t \le 1.57 \times 10^7 \\ < \gamma_{max}\,(t_0/t)^{1.2} & t > 1.57 \times 10^7 \end{cases} \qquad (17)$$

where t represents the time after detonation, t_0 represents the time when the peak free-field dose rate, γ_{max}, occurs at the location under consideration. The variables t and t_0 are expressed in seconds and γ_{max} is in rads/second. Figure 9 shows a typical free-field nuclear fallout radiation environment.

For the practical range of weapon yields and heights of burst, a worst-case or lower bound for t_0 of about 500 seconds can be expected outside the blast/thermal destruction region (NCS, 1985). This value is assumed throughout this program and report. Substituting this value into (17) yields

$$\gamma(t) = \begin{cases} 0 & 500 > t \\ 1733\ t^{-1.2}\ \gamma_{max} & 500 \le t \le 1.57 \times 10^7 \\ < 1733\ t^{-1.2}\ \gamma_{max} & t > 1.57 \times 10^7. \end{cases} \qquad (18)$$

3.3.4 Subroutine GAMMAEX

Subroutine GAMMAEX calculates the total dosage of gamma radiation to which the fiber sublink section has been exposed. This dosage is calculated at the current time of the model and is assumed to be constant along the entire length of the sublink section. The protection factor of each section will affect the radiation dosage for that section. The equation used to calculate this total dosage can be determined by integrating (17) and evaluating at time t_0 = 500 to form

$$\Gamma(t) = \begin{cases} 0 & 500 > t \\ 8664\ \gamma_{max}\ (.289 - t^{-0.2}) & 500 \le t \le 1.57 \times 10^7. \end{cases} \qquad (19a)$$

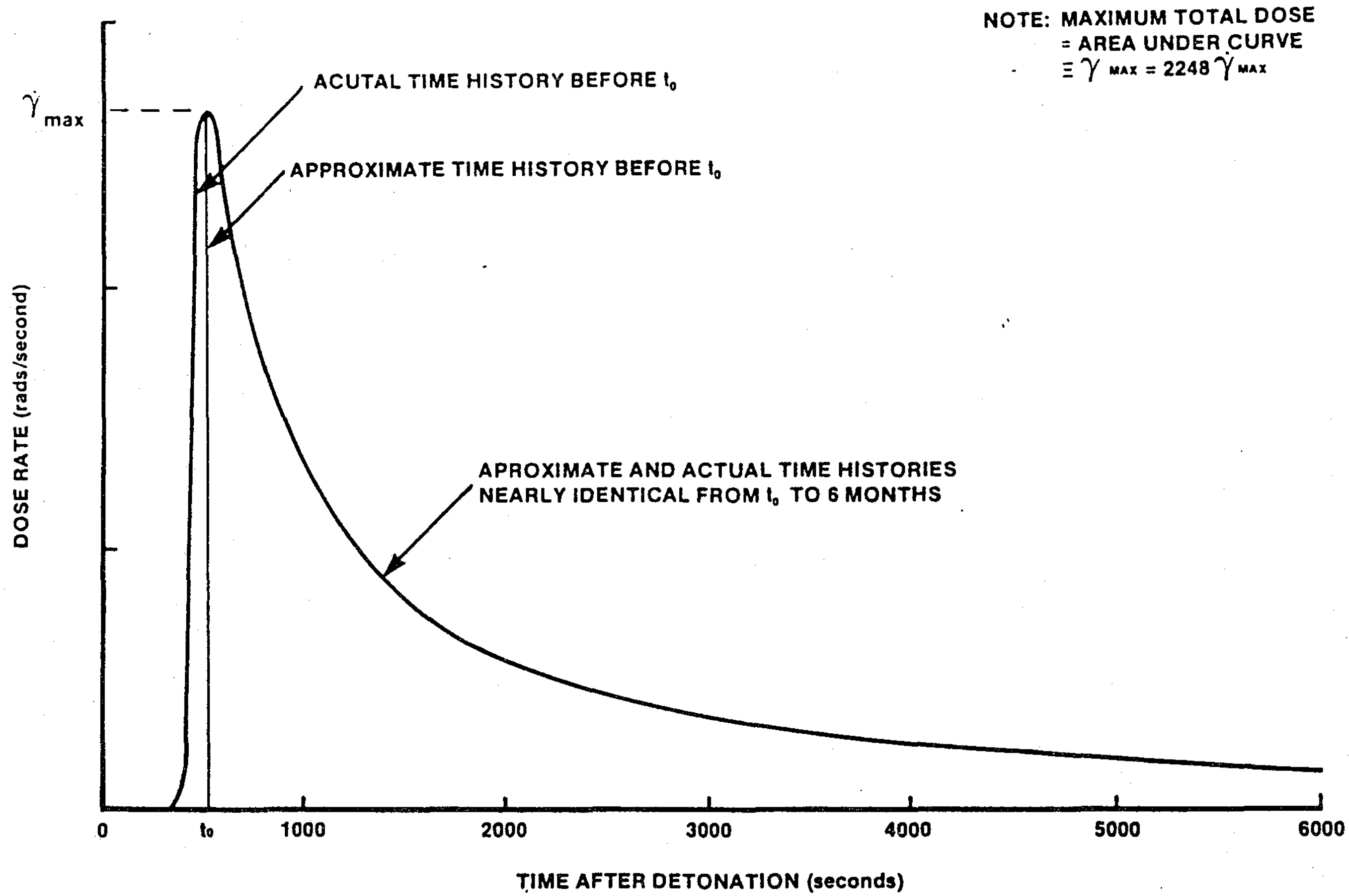

Figure 9. Typical free-field nuclear radiation environment (after NCS, 1985).

The additional dosage accumulated for the time period $1.57 \times 10^7 < t < \infty$ can be estimated by other means to be $0.0366\ \gamma_{max}\ t^{1.2}$ (NCS, 1985). The total dosage for $1.57 \times 10^7 < t < \infty$ is bounded by

$$5\ \gamma_{max}\ t_0^{1.2}\ (t_0^{-0.2} - 0.0364) < \Gamma(t) \leq 5\ \gamma_{max}\ t_0^{1.2}\ (t_0^{-0.2} - 0.0291) \equiv \Gamma_{max} \qquad (19b)$$

By evaluating at $t_0 = 500$ seconds, (19b) becomes

$$\Gamma_{max} = 2248\ \gamma_{max}. \qquad (20)$$

By substituting (20) into (18) and (19), the following equations are arrived at

$$\gamma\ (t) = \begin{cases} 0 & 500 > t \\ .771\ t^{-1.2}\ \Gamma_{max} & 500 \leq t \leq 1.57 \times 10^7 \\ < .771\ t^{-1.2}\ \Gamma_{max} & t > 1.57 \times 10^7 \end{cases} \qquad (21)$$

and

$$\Gamma\ (t) = \begin{cases} 0 & 500 > t \\ 3.85\ \Gamma_{max}\ \ (.289 - t^{-0.2}) & 500 \leq t \leq 1.57 \times 10^7 \end{cases} \qquad (22a)$$

with

$$0.972\ \Gamma_{max} < \Gamma\ (t) < \Gamma_{max} \qquad t > 1.57 \times 10^7. \qquad (22b)$$

3.3.5 Subroutine LOSS

The power loss due to the darkening of fiber exposed to fallout radiation in each sublink section is calculated in subroutine LOSS. The loss factor is a combination of the actual loss factor and the recovery factor. A typical graph of this value verses time is shown in Figure 10. This figure shows that the induced loss rate is not only a function of time after exposure, but also a function of the total exposed dosage.

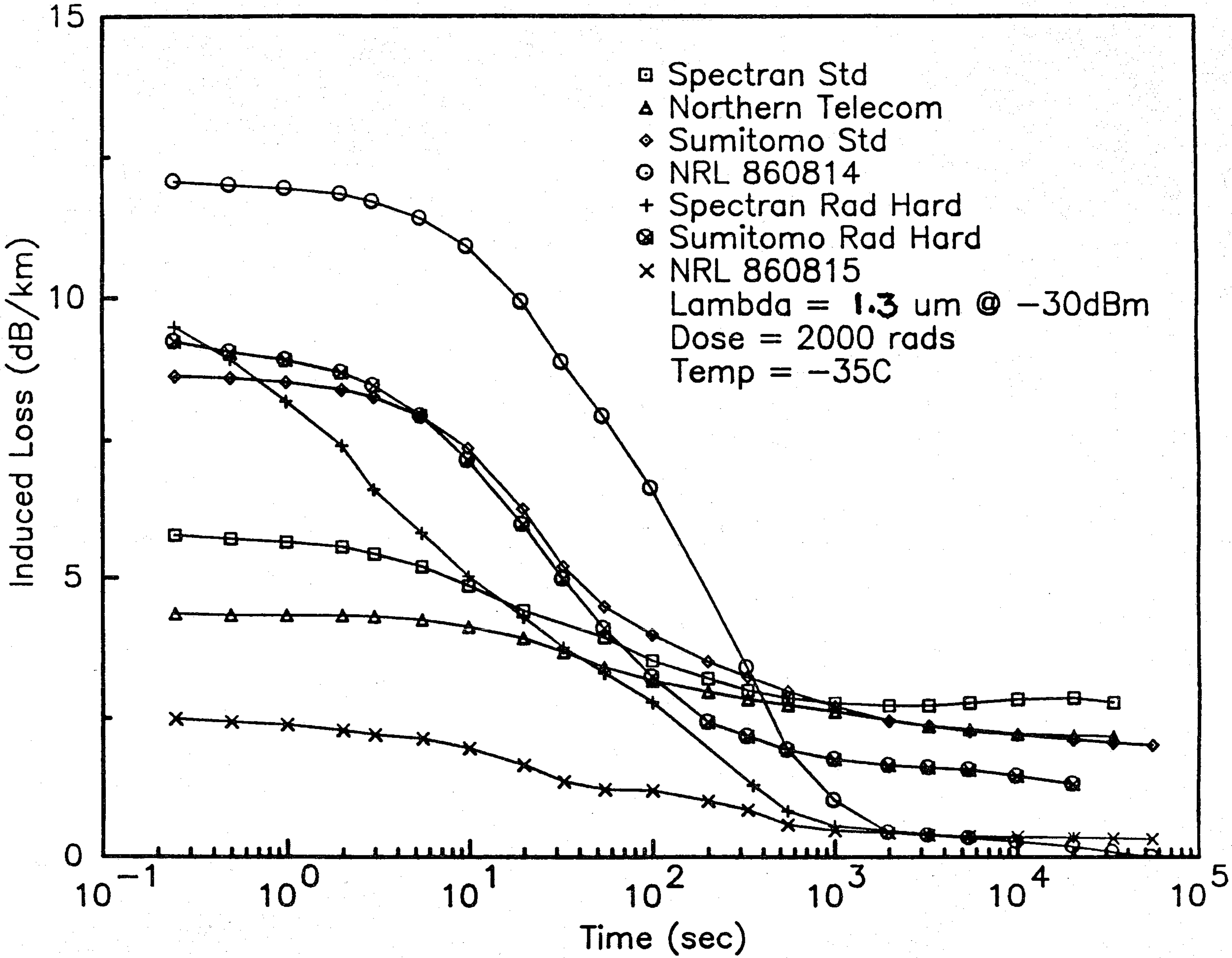

Figure 10. Induced loss of Sumitomo Standard optical fiber as a function of time and total dosage (Friebele, 1987).

Figure 10 shows the recovery of optical fiber that has been exposed to gamma radiation for a pre-determined amount of time to arrive at a specified total radiation dosage after which the radiation source was removed (Friebele, 1987). FIBRAM is designed to model a fiber's response to an exponentially decaying dosage rate. The ideal method of calculation would be to integrate and scale down Figure 10 over the exponential decay curve for the gamma dosage. This is shown as:

$$\text{Total Loss}(t) = \int_{\tau = 500}^{t} \frac{\text{Loss}(t - \tau)\ \text{Rate}(t - \tau)}{\Omega}\, d\tau \tag{23}$$

where Ω is the gamma dosage rate at which the induced loss curve was measured.

This method would not be practical since each type of optical fiber has a unique induced loss curve and one equation could not be pre-determined that would satisfy all fiber types. The method used in FIBRAM to solve this dilemma is to simulate such an integration by doing a piece-meal summation over the induced loss curve for the specified optical fiber. This can be expressed as:

$$\text{Total Loss}(t) = \sum_{i=1}^{k} \frac{\text{Loss}(i)\ \text{Rate}(i)}{\Omega}\ \frac{(t-500)}{k} \tag{24}$$

where k is the number of iterations. Theoretically, if k were set to infinity, (24) would yield the same results as (23). Since computers have finite limitations, however, this is not possible. The parameters file SM-13.DAT was executed three different times using values of k = 3, 10, and 100. Figure 11 and Table 5 show these results as output files OUTPUT0.PRN, OUTPUT1.PRN, and OUTPUT2.PRN, respectively. From the figure and table, the value of k = 10 seems to be a reasonable compromise between accuracy (very large k) and computing speed (small k). This value of k can be changed by user modification to subroutine LOSS.

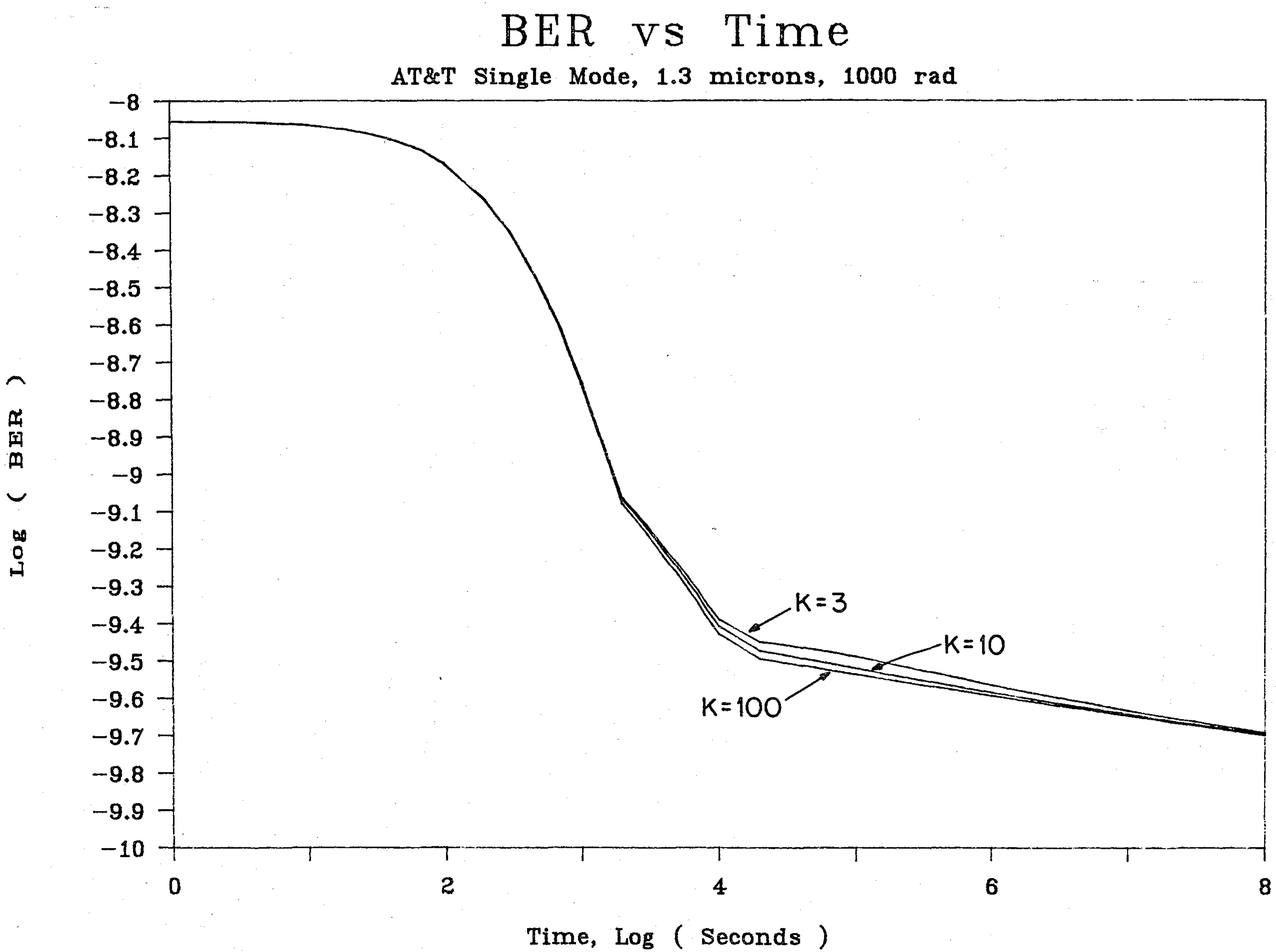

Figure 11. Different values of k used to calculate the "BER versus Time" graph.

Table 5. Calculation Time for Different Values of k

Number of increments (k)	Calculation time on an IBM AT (h:mm:ss)
3	0:03:56
10	0:07:13
100	1:03:02

The exposed gamma dosage at each iteration is determined by calling subroutine GAMMAEX (Section 3.3.4). Each induced loss value is determined by calling subroutine INTERP (Section 3.2.9) to interpolate the approximate value from the data files.

Photo-bleaching recovery techniques are not provided in this version of LOSS because the effects were found to be negligible when compared to the detrimental effects of darkening (NCS, 1985). As better and more accurate data is discovered, subroutine LOSS can be modified to include these data.

Natural recovery can, however, be a significant factor. After optical fiber cable has been darkened by exposure to radiation, it will slowly and continually recover, thus becoming less dark. Depending on the type of fiber and the radiation dosage, it can take hours, days, or even years to show significant recovery levels. Also, the recovery of the cable may be offset by the fact that the fiber is continually being exposed to more damaging gamma radiation while it is recovering from the effects of earlier radiation. It may take a long time, depending on the cable and total dosage, for the radiation levels to decrease to a point where the fiber recovery becomes apparent.

3.3.6 Subroutine SIGPOW

Subroutine SIGPOW calculates the receive signal power at the photodiode. This is done by using (3). The losses for attenuation and darkening, both in decibels, are subtracted from the transmitted power specified by the user.

This subroutine performs several other valuable functions. First, SIGPOW calls GAMMAEX (Section 3.3.4) to determine the amount of gamma radiation exposure to the photodiode itself. High enough levels of radiation will cause the photodiode to fail. If SIGPOW determines that the photodiode has failed, an error condition is signaled to RUNTIME by the receive signal power being set to -9999. Subroutine RUNTIME will realize the photodiode has failed and will set the bit error ratio to a failure value of 1/2.

If the photodiode has not failed, SIGPOW will then determine the radiation dosage rate at the photodiode by calling DOSERA (Section 3.3.3). This dosage rate will be used to determine the bit error ratio when RUNTIME calls subroutine READBER (Section 3.3.7).

3.3.7 Subroutine READBER

The graphs of bit error ratios for photodiodes provided in this report are valid only within specific power ranges. Subroutine READBER will determine whether the receive signal power for the current calculation is within the specified power range for the photodiode being modeled.

If the receive signal power is lower than the lowest acceptable power level, the bit error ratio is set to 1/2, indicating that no useful data can be transmitted along the fiber link. If the receive signal power is greater than the highest applicable power level, no signal degradation is assumed and the bit error ratio is set to 1×10^{-10}, which is assumed, in this report, to be the best bit error ratio possible.

Figure 7 showed an example of a linear approximation of the bit error ratio verses the receive signal power. In this example, the highest power level that can be modeled is -20 dBm and the lowest allowed power level is -52.5 dBm. Above -20 dBm, the bit error ratio is assumed to be 1×10^{-10} and below -52.5 dBm, the bit error ratio is assumed to be 1/2. Between -20 and -52.5 dBm, the bit error ratio must be linearly approximated from the graph using subroutine GRAPHR.

If the receive signal power is within the specified power range, subroutine GRAPHR (Section 3.2.6) will be called to approximate the expected bit error ratio from data files specific to the selected photodiode. Subroutine READBER determines which file contains the appropriate data for GRAPHR to interpolate. READBER then calls GRAPHR to calculate the bit error ratio.

4. APPLICATIONS OF FIBRAM

4.1 Modifying a Previously Entered Data File

Table 6 shows the variables stored in the disk-based data file named SM-13.DAT. This file is included on the FIBRAM disk and contains data pertaining to single-mode AT&T optical fiber operating at a wavelength of 1.3 microns. It is used several times in this report for generating system models.

Appendix E.1 shows a sample run of FIBRAM where the user has modified and run the variables stored in data file SM-13.DAT. The user's inputs are highlighted by underlining them.

In this example, the user has loaded data file SM-13.DAT by typing 'l' (as in "load") <RETURN> and then entering the file name. The user ran the model by typing 'r' <RETURN> and then saved the results to a disk file by typing 'g' <RETURN> 'OUTPUT1.PRN' <RETURN> (Note that OUTPUT1.PRN is the same output file as described earlier in Section 3.3.5).

The user then decided to lower the transmitted power of the model from 0.01047 milliwatts to 0.007 milliwatts. The user did this by typing 'm' <RETURN> from the main menu to get into the modification mode. At this point FIBRAM automatically printed out a listing of the current variables and then prompted the user to enter new values. Each time the user was prompted to change a variable that did not need to be changed, the <RETURN> key was simply pressed. When the user wished to change a particular value (in this case the

Table 6. Display of FIBRAM's Current Variables When Using Parameters File SM-13.DAT

```
These are the current values;
Help is off.  Autodisplay is on.
Total power transmitted is     .0104700 milliwatts
Maximum allowed gamma for a PD of type 2 is     1500.000 rads
The start and end times are    500.00 and     90000500.00 s
The initial spacing between time steps is     1.00 s
The output will be offset by     500.000 seconds

Sect    Length (m)       PF  Gamma (rad)  F-Type  Intrins
  1           30.0      4.0     1000.0       1     .380000
  2        39390.0   7000.0     1000.0       1     .380000
  3          580.0      1.0     1000.0       1     .380000
  4    photodiode      10.0     1000.0
```

transmit power), the new value was entered followed by the <RETURN> key.

When all modifications were entered the user could have chosen to save this new configuration to a disk file by typing 's' <RETURN>. Instead, the user chose to only run the model by typing 'r' <RETURN>. Note that the user has toggled the Autodisplay feature ON to view the model as it is being calculated. Several representative calculations are shown in the example.

When the model was finished running, the user typed 'g' <RETURN> to save the results in a disk file named OUTPUT3.PRN and then exited normally.

Figure 12 shows a graph of OUTPUT1.PRN and OUTPUT3.PRN. This shows the change in the bit error ratio versus time when the input power is changed from 0.01047 milliwatts to 0.007 milliwatts.

4.2 Substituting Fiber Types

A close examination of Table 6 will show that section 3 is 580 meters of cable with a protection factor of 1. Section 1 is 30 meters with a protection factor of 4. These values indicate that the cable has been installed in an area where there is no natural protection such as a bridge crossing. If the system is modeled using Spectran Standard Cable (Fiber Type 5, 1.3 microns, exposed at 2000 rads. See Figure 10.) for all three sections, the output curve shown in Figure 13 is produced. If this model was to be run again with type number 9 (Spectran Radiation-Hardened Cable, 1.3 microns, exposed at 2000 rads. See Figure 10.) used in place of type number 5 for sections 1 and 3, an improved bit error ratio would be achieved. This is also shown in Figure 13.

4.3 Using the RANGER Feature

Appendix E.2 shows a sample run of FIBRAM where the user has decided to model the system variables contained in data file SM-13.DAT for three different values of gamma radiation exposure. The user's inputs are highlighted by underlining them.

In this example, the user has typed '>' <RETURN> to invoke the RANGER feature to model an optical fiber system in response to the different levels of gamma radiation exposure. FIBRAM then asked the user to enter the name of the

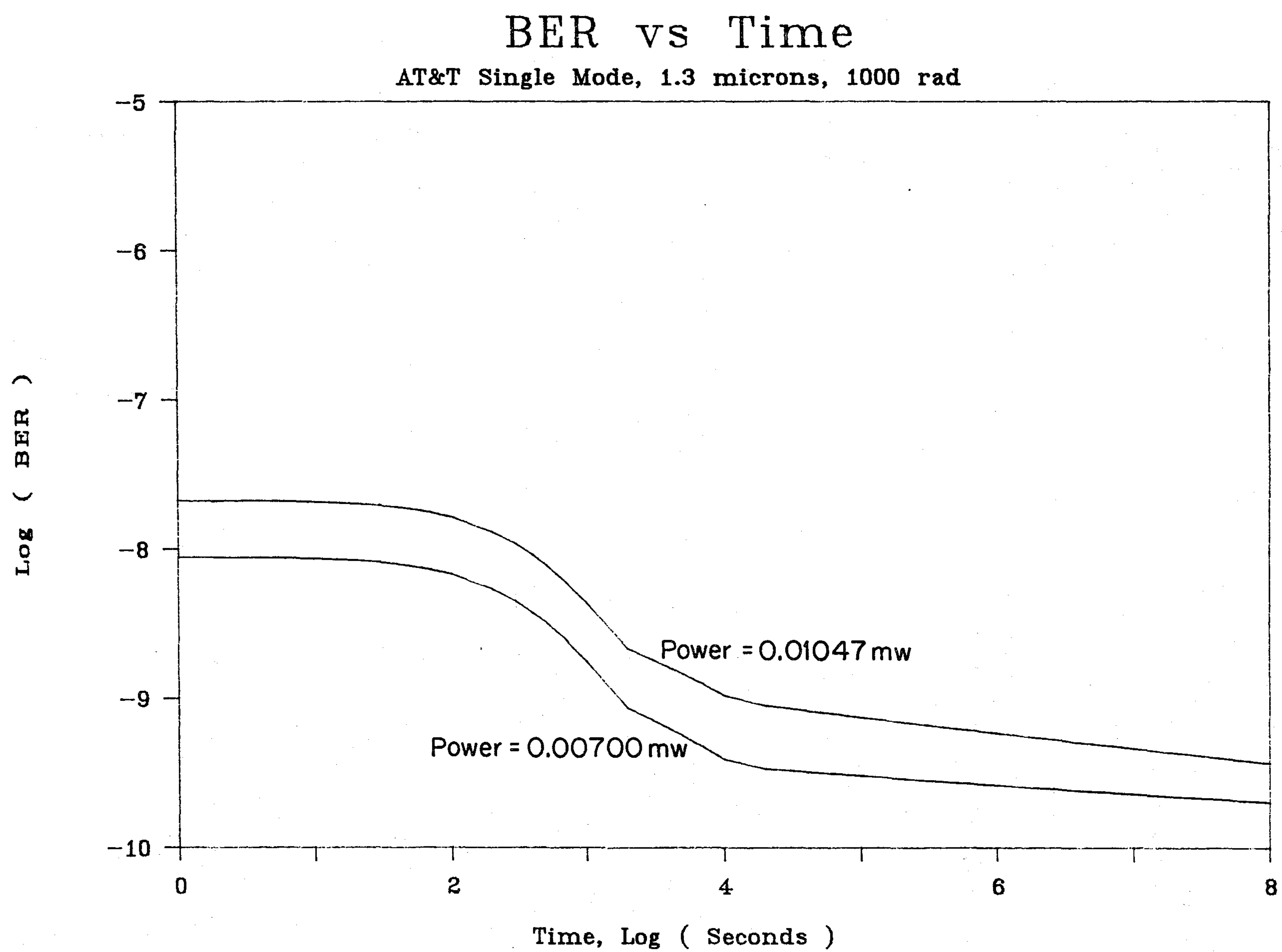

Figure 12. "BER verses Time" graph showing the change as a result of a change in input power level.

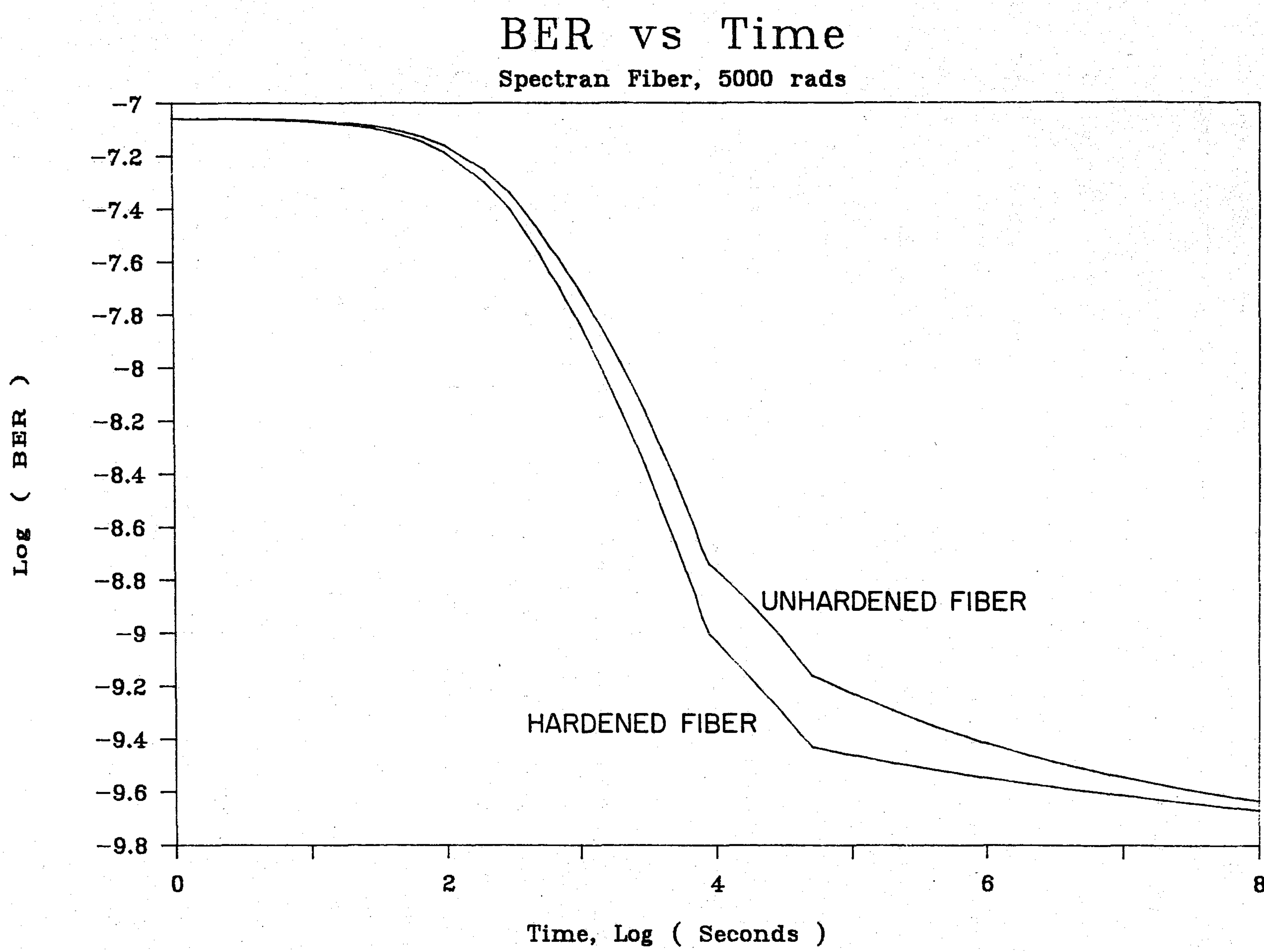

Figure 13. "BER verses Time" graph using radiation-hardened and unhardened fiber optic cable.

disk file containing the previously entered base parameters. In this case the user has chosen to use the variable file SM-13.DAT, as shown in Table 6. These current values were then displayed on the video screen for the user's benefit.

The user was presented with a table of variables and asked which of them the user wished to vary. The user has typed '2.' <RETURN>, which corresponds to the gamma radiation exposure level.

FIBRAM then asked the user to enter the number of different levels this variable will assume. This user has typed '0.' (zero) <RETURN>, indicating to FIBRAM that he/she wishes to use levels previously stored on a disk file. FIBRAM responded by asking the user to type in the name of that file, in this case G50-500.RNG. The data in this file told FIBRAM to model the system at gamma exposure levels of 500, 1000, and 5000 rads. The results of these runs were to be saved in the files OUTPUT4.PRN, OUTPUT1.PRN, and OUTPUT5.PRN, respectively (Note that OUTPUT1.PRN is the same output file as described in Sections 3.3.5 and 4.1). After the run was completed, the user exited normally.

Figure 14 shows the graph of the bit error ratio versus time using the print files created by this sample run and Lotus 1-2-3.

4.4 Data Perturbations

Figure 15 shows a sample run using multimode cable at 1.3 microns and a power input of 0.02 milliwatts. The user will notice that Figure 15 shows some unusual looking perturbations. At first glance, this graph would seem to be in error because many of the other graphs in this report are well-behaved functions that continually decrease with time. This is not the case, however, for Figure 15.

Figure 15 is the result of the photodiode voltage being reduced below the threshold voltage, causing the photodiode to enter the cut-off region and producing an abruptly worsened bit error ratio. Eventually, fiber recovery increases the photodiode voltage back above the threshold voltage and the bit error ratio improves. In order to be driven into the cut-off region, the power input to the photodiode (before irradiation and including intrinsic losses) must be far below the normal operating range specified for that photodiode. Fiber optics systems will not normally be expected to operate at such reduced power levels.

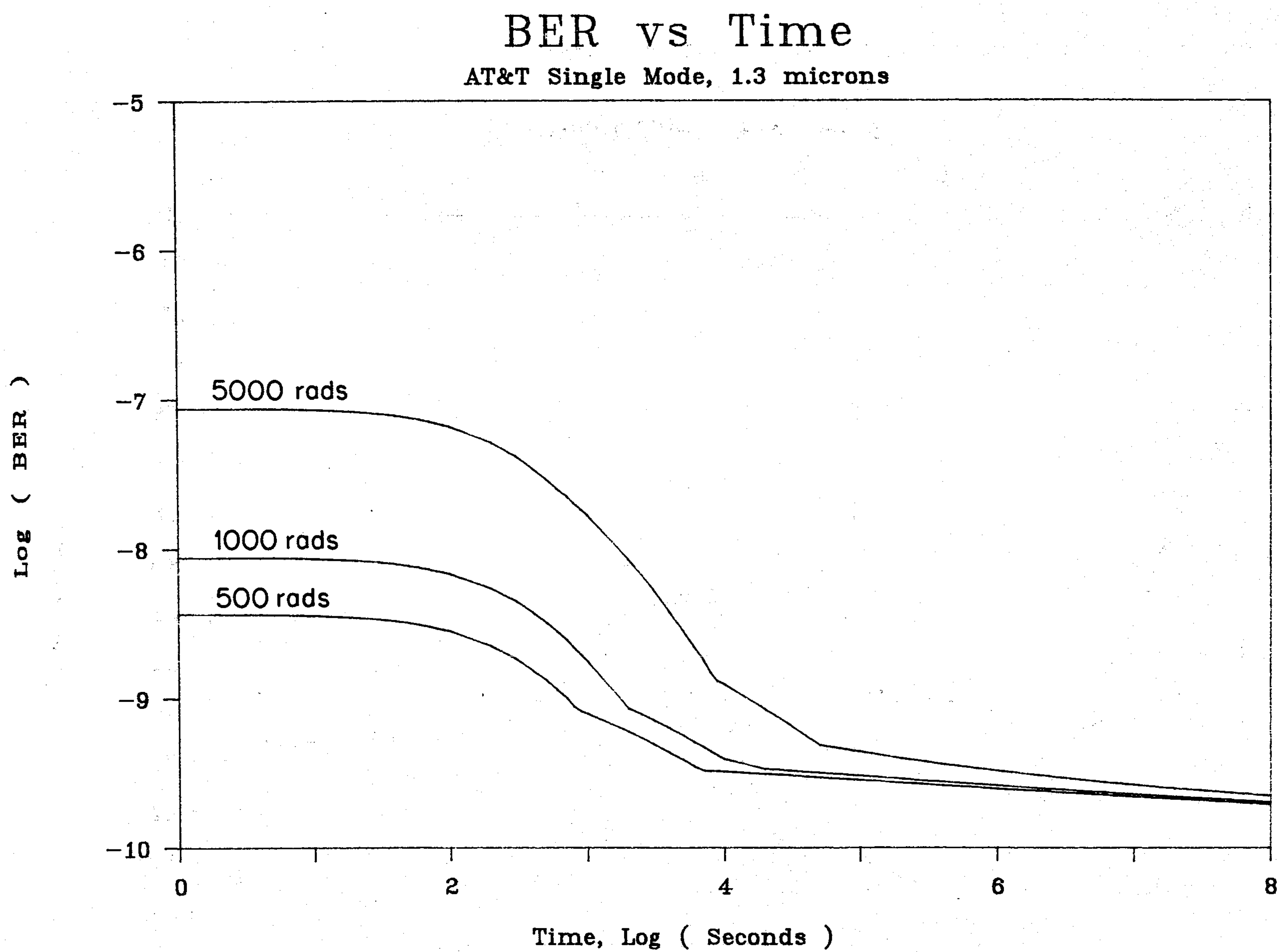

Figure 14. "BER versus Time" graph for AT&T single-mode fiber operating at a wavelength of 1.3 microns in response to various levels of fallout radiation.

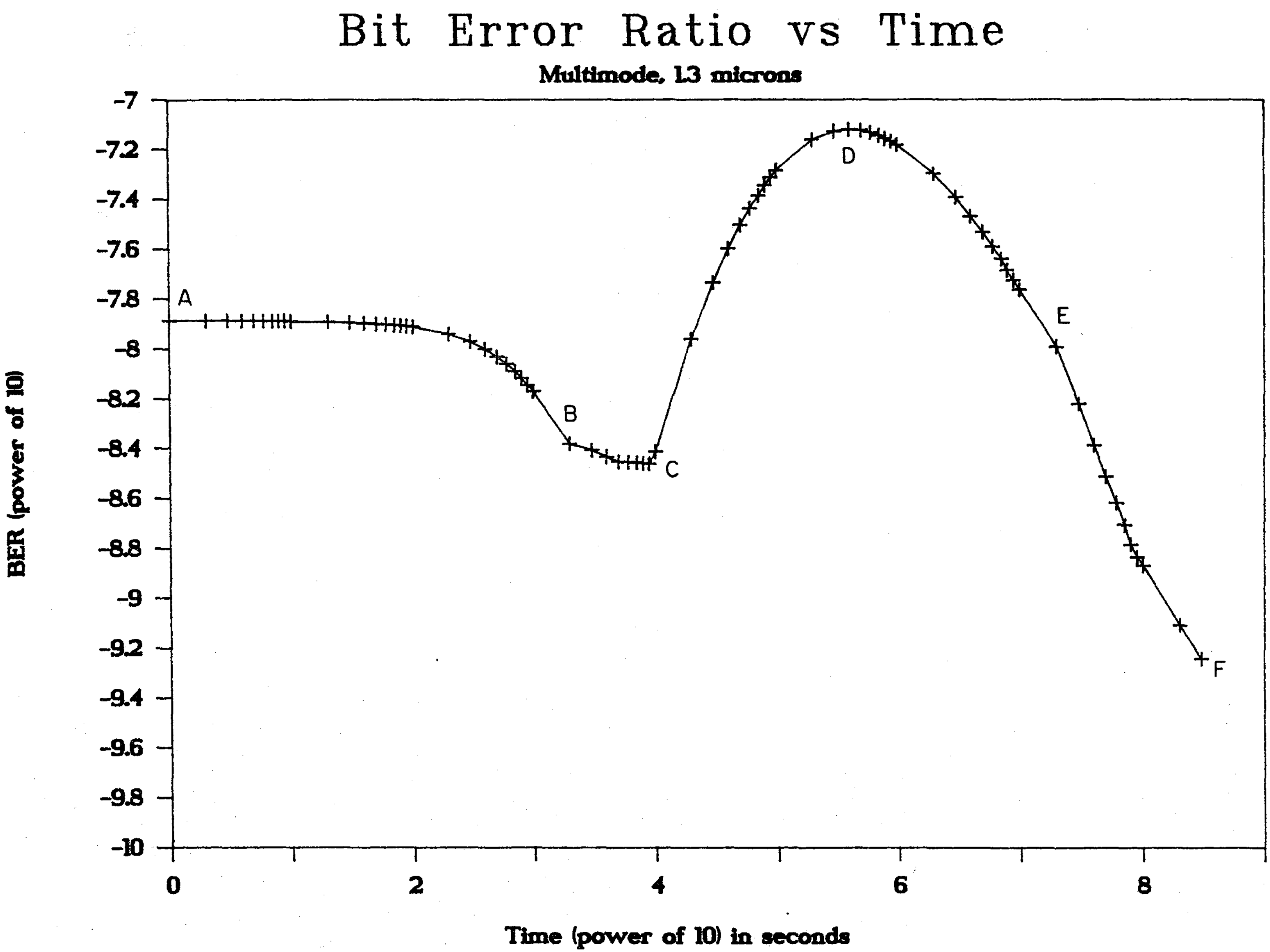

Figure 15. "BER versus Time" graph for multimode calbe operating at 1.3 microns.

Figure 16 shows the straight-line segment approximation of the photodiode curve used in calculating the bit error ratio of Figure 15. The line where the cut-off region begins is labeled "0 rads/sec". Section 3.2.4 describes the methodology for arriving at Figure 16.

Table 7 shows a point-by-point analysis of the optical fiber system as time ranges from 1 second to 300,000,000 seconds. Critical points of interest in Figures 15 and 16 are alphabetically labeled and listed in the table. By alphabetically following the labels in the table and on the two figures, we can follow the reasoning behind the unusual perturbations in the data.

The model begins at time equals 1 second and is labeled as point A. Figure 15 shows that the system model performs as expected to the point marked C. Figure 16 shows that, from point A to point C, the bit error ratio is still dependent upon both the receive signal power and the dosage rate. At point C, Figure 16 shows that the receive signal power is small enough to cause the photodiode to enter the cut-off region. The bit error ratio then becomes independent of the dosage rate and become dependent only upon the receive signal power. Points D and E on Figure 16 show the heavy dependency of the BER on only the receive signal power. As recovery takes place within the optical fiber cable, the receive signal power improves to the point where the photodiode voltage is once again above the threshold voltage. The bit error ratio then becomes dependent upon both the receive signal power and the radiation dosage rate. This is shown as point F on Figure 16.

Table 7 also shows that, while the dosage rate seems to decrease almost exponentially with time, the receive signal power has a relatively long time constant and will gradually fall to a minimum value where the recovery factor will cause it to increase again slightly.

Table 7. Point-By-Point Analysis of the Flow of Figures 15 and 16

Label	Time (seconds)	Dosage Rate (rads/sec)	Receive Signal Power (dBm)	BER
A	1	0.44 E-1	-35.78	0.13 E-7
B	1000	0.12 E-1	-37.71	0.68 E-8
C	9000	0.13 E-2	-38.99	0.35 E-8
D	400000	0.77 E-4	-40.19	0.75 E-7
E	10000000	0.30 E-6	-39.75	0.17 E-7
F	300000000	0.52 E-8	-38.48	0.57 E-9

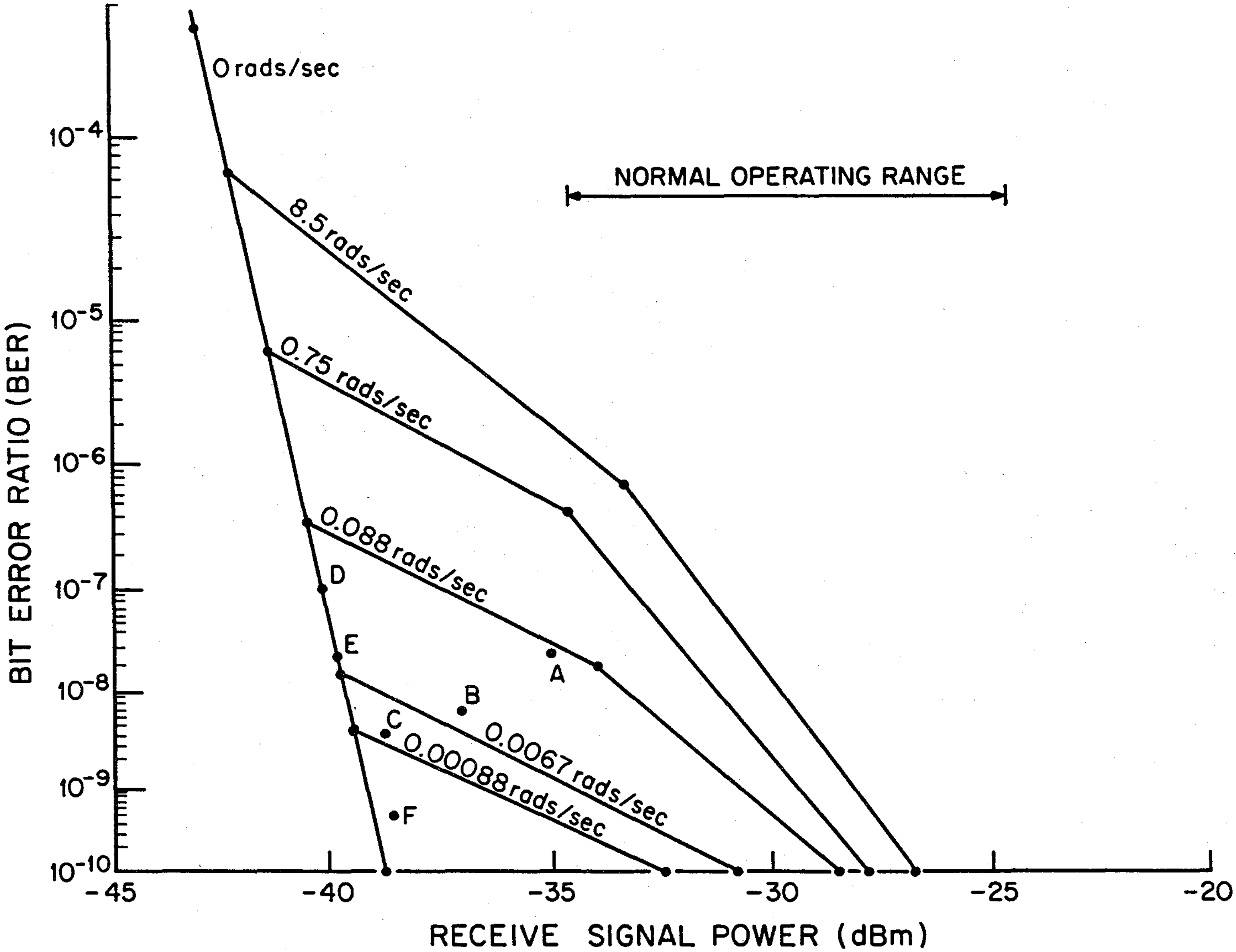

Figure 16. The flow of Figure 15 represented on the graph of the bit error ratio of APD 11E, 11F and 11G photodiodes as a function of the receive signal power and radiation dosage (after NCS, 1985).

5. REFERENCES

CDC (1985), Control Data Corporation Fortran Version 5 Reference Manual.

Friebele, E.J. (1987), Private communication with U.S. Department of the Navy, March 20.

NCS (1985), Assessment of Nuclear Fallout Radiation on FT3C Lightwave Digital Transmission System, National Communications System Technical Information Bulletin 85-11, November.

Salz, J. (1985), Coherent Lightwave Communications, AT&T Technical Journal 64, No. 10, December, pp 2153-2210.

APPENDIX A. USING THE PROGRAM

FIBRAM has been designed for the first-time user as well as the user who is already familiar with the program. It is designed to run on an IBM PC/XT/AT or some other computer utilizing MS-DOS.

In this report, the following conventions are used:

<RETURN> means to hit the return or enter key after text entry. If <RETURN> appears by itself, it indicates that no other text need be entered.

'Text', in single quotation marks, means to type the letters t, e, x, and t, in lower case, without the quotation marks.

In most cases, items in single quotation marks will be required to be followed with a <RETURN>.

When the user is prompted to enter a file name, a file name such as FILENAME.XXX should be used. In this example, FILENAME corresponds to any easily remembered sequence of letters and numbers and XXX corresponds to the file extension appropriate for that file. The appropriate file extension is given in the accompanying documentation for the subroutine where it is needed.

A.1 Basic Program Usage

A.1.1 Starting the Program

Before starting FIBRAM, the user should examine the disk or subdirectory where FIBRAM is stored. The following files should be included in the directory listing:

FIBRAM.EXE	FIBTYPES.TXT	INTRO.TXT
MAINMENU.TXT	NSECTION.TXT	PHOTYPES.TXT
RANGE.TXT	TIME.TXT	WORST.TXT
BER-11A.GPH	BER11DEF.GPH	*.FTY.

To begin using the program, the user should type 'FIBRAM' <RETURN> at the DOS prompt. At this point FIBRAM will perform several initialization routines and then display its main menu. The novice user may wish to type 'h' <RETURN> to toggle the automatic HELP function. The default position for the help function is OFF. In the OFF position, FIBRAM will display only an abbreviated main menu of options and will not offer specific explanations in preface to questions that it asks of the user. If, when running the model, the user types in values that are not allowed, FIBRAM will assume the user misunderstands the question and will activate the help function to provide additional information for the user.

When the help function is toggled to the ON position, FIBRAM will present the user with the expanded main menu explanation of each of the commands available. It will also provide specific explanations of questions it is about to ask the user during the running of the program.

A.1.2 Using the System Variables

To calculate useful results, FIBRAM requires the input of data specific to the optical fiber system that is to be modeled. These needed variables are shown in Table 2.

To display the variables listed in Table 2, the user should type 'd' <RETURN> (subroutine DISPLY) from the main menu. This will cause FIBRAM to display all of the current system variables.

To change any of these variables, the user should type 'm' <RETURN>. This will cause FIBRAM to display the values of the current system parameters and then prompt the user to modify them.

When changing any of the system variables, the user will be presented with the current value and a prompt to change that value. If the user just types <RETURN>, the current value is retained unchanged. If the user types in a new value and <RETURN>, this new value now becomes the current value.

If the user has the help function ON, a short explanation will be provided for many of the variables before the user is prompted to change them. Otherwise the user may consult Section 3 of this document for further explanation of these variables and the subroutines in which they are contained.

The timing variables of the model may be changed by the user, from the main menu, by typing 't' <RETURN>. These variables are the starting time, the ending time, the offset time, and the incremental size of the timing steps.

The optical fiber model that the user designs may be given a name. This can be done from the main menu by typing 'n' <RETURN>. This name will be displayed each time the main menu is accessed and does not affect the running of the model or the file name the user may be using for saving or loading variables or results. This name option is provided purely for the convenience of the user when organizing disk files.

Once the user has finished adjusting the system variables, they may be saved as a variables file for retrieval. This can be done by typing 's' <RETURN> from the main menu. FIBRAM will then respond by asking the user for a file name in which to save the variables. The extension of this file name should be ".DAT". This command will save ALL current variable values onto a disk file with that name.

This variable file can then be retrieved by typing 'l' (as in "load") <RETURN> from the main menu. FIBRAM will then ask the user for the file name previously used for saving the variables. The program will then load these previous variable values into volatile memory and return to the main menu.

A.1.3 Running the Model

Once the system variables have been entered or loaded into the computer, the model can be run by typing 'r' <RETURN> from the main menu.

As the user runs the model to calculate the bit error ratio, the following intermediate calculations are displayed:

- The current time, t, of the model
- The current total amount of radiation to which each section is exposed
- The current radiation dosage rate for each section
- The current loss factor for each section
- The total darkening for the entire sublink
- The current total amount of radiation to which the photodiode is exposed
- The current radiation dosage rate at the photodiode
- The current received signal power at the photodiode (in dBm).

A.1.4 The Results

When the model is finished, the bit error ratio, along with the time at which it was calculated (minus the offset time), is stored in random access memory (RAM) as a program array. These results can be either displayed on the screen, sent to a disk file, or printed on paper.

To print these values to the video screen, the user should type 'p' <RETURN> from the main menu after the model has been run. If the user wishes to have a "hard copy" of the results sent to a printer, the 'Ctrl-PrtSc' buttons should be pressed before typing 'p' <RETURN>.

To save these results to a disk file, the user should type 'g' <RETURN> from the main menu. The logarithms of the output time and the bit error ratio will be "printed" to the specified disk file.

When saving results in a disk file, the user will be asked to provide a file name in which to save these results. The user should provide a file name with an extension ".PRN". This kind of file is known as a "print file" by most commercially available graphics programs. Print files can usually be loaded into graphics programs and incorporated into graphical output. Figure A-1 shows a sample graphics output using Lotus 1-2-3. The print files used to create this graph and others in this report are labeled OUTPUTβ.PRN and are included on this disk for those users of graphics packages who may wish to have the raw output from these examples. Figure A-2 shows the same output file used to create Figure A-1 charted with Chart-Master, another graphical charting package. It should be noted at this point that each commercially available graphing package has its own unique characteristics and may have too many or too few features for the particular applications of each user.

A.2 Advanced Features

For the user that needs to view how their system model reacts to the variance of a particular input, the RANGER feature is available. Variables which can thus be ranged are the input power or amount of exposed radiation. RANGER can be invoked by typing '>' <RETURN> from the main menu. The user is then asked to enter a base variable file and then input the values of the variable to be ranged. When the last value is entered, FIBRAM will model the system at each of the values indicated and save the subsequent results to specified disk files.

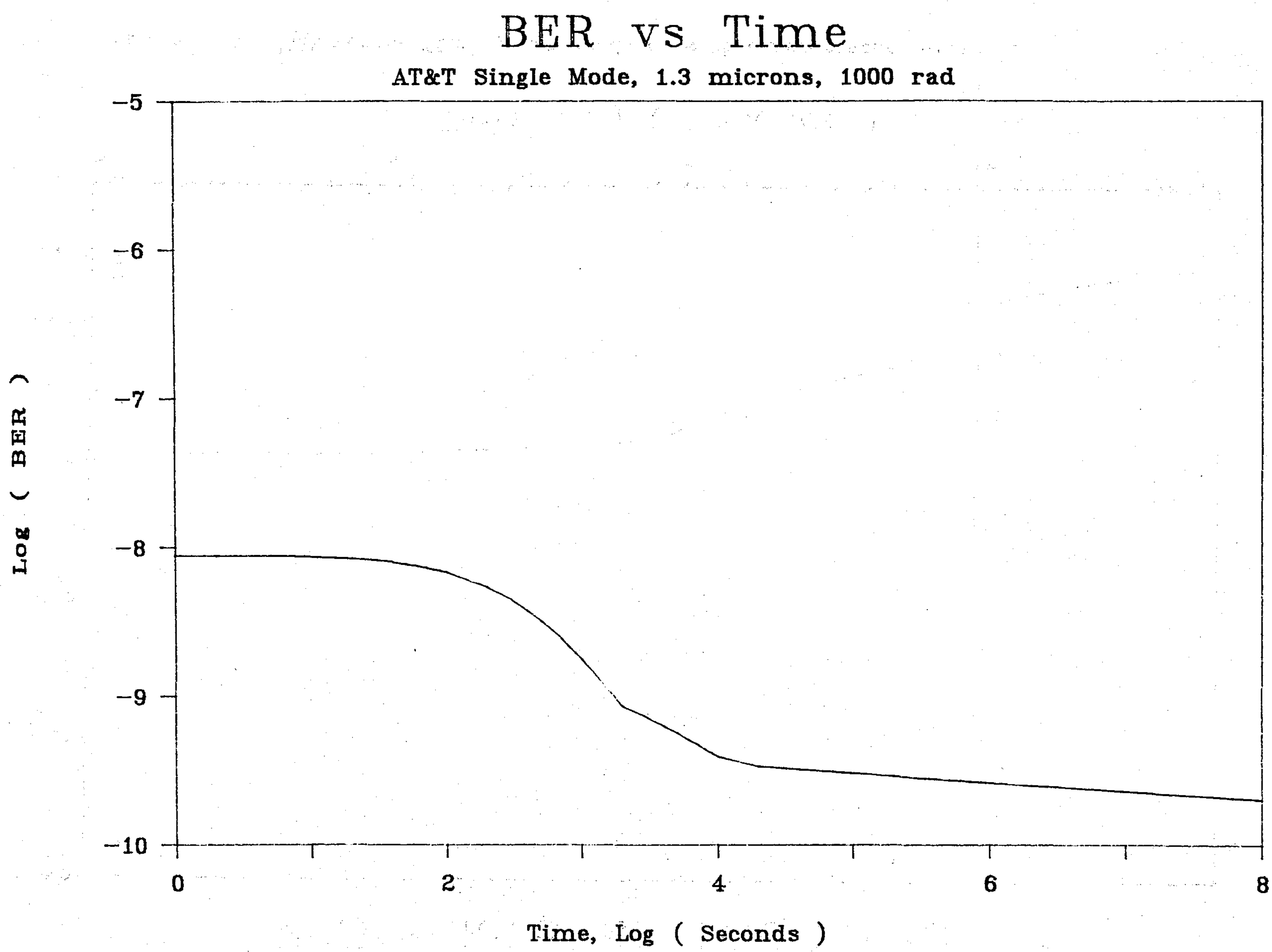

Figure A-1. "BER versus Time" graph produced with Lotus 1-2-3 software.

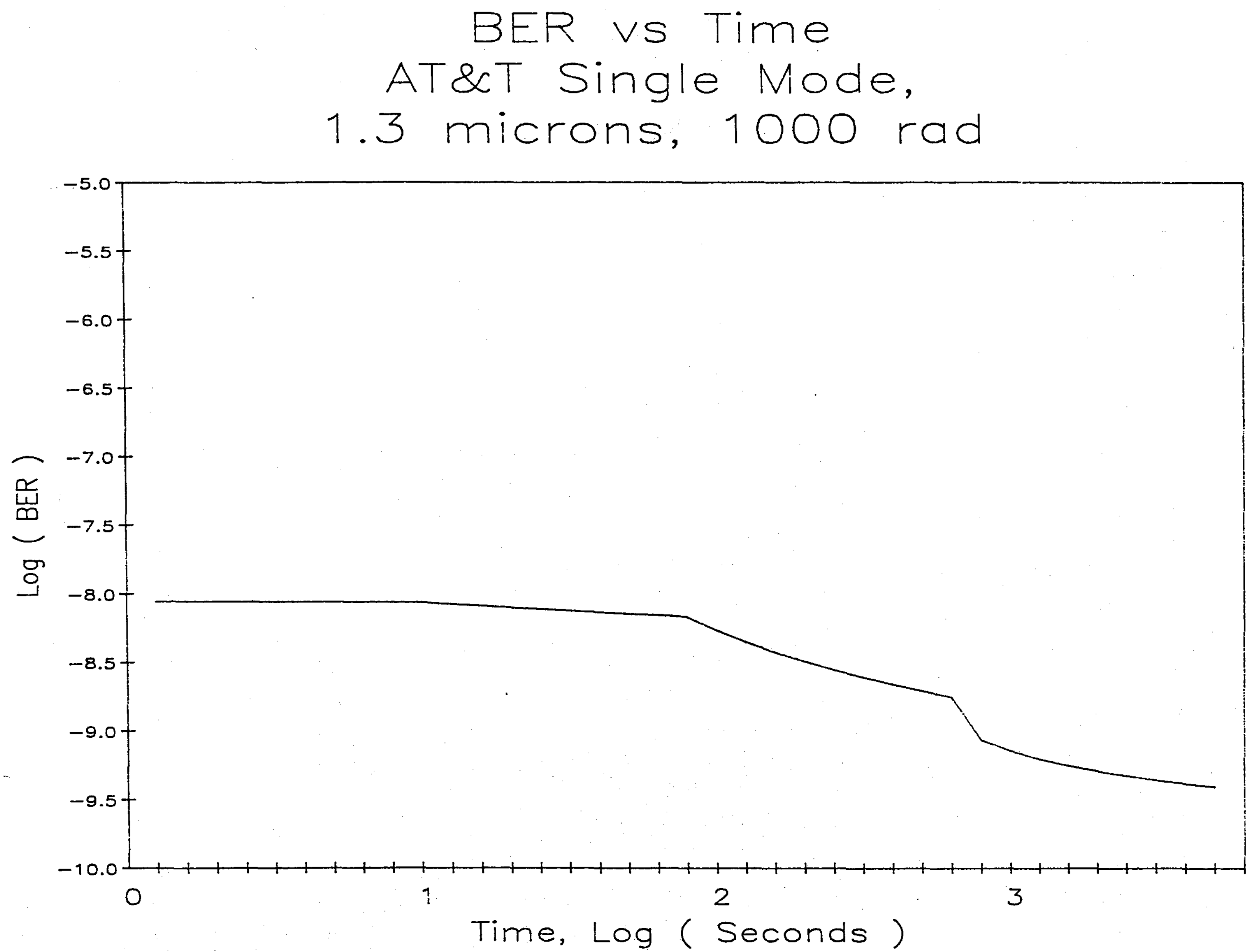

Figure A-2. "BER versus Time" graph produced with Chart-Master software.

APPENDIX B. MODIFYING FIBRAM

It will sometimes be useful for the user to be able to modify the subroutines that comprise FIBRAM. The user will need the following items to make modifications to FIBRAM:

- the original FIBRAM program diskette
- two blank, formatted diskettes (not included)
- a Fortran 77 compiling program (not included)
- a text editing program (not included)
- an IBM PC/AT compatible computer with two disk drives (not included)
- an MS-DOS disk (not included, usually supplied with computer).

To modify a subroutine, the user will need to perform the following six steps:

1. Make a backup copy of the program disk onto the first blank disk and put it in a safe place
2. Copy the Fortran and object files from the program disk to the second blank disk
3. Edit the Fortran programs on the second disk
4. Recompile and link FIBRAM, using the second disk
5. Copy the modified Fortran and object files from the second disk back onto the program disk
6. Send the author of FIBRAM a copy of the changes.

B.1 Updating the Program Disk

B.1.1 Making a Backup Disk

Insert the DOS disk into drive A (users of IBM AT/XTs may have slightly different procedures for copying and modifying files and should consult their user manual for their type of computer). At the DOS prompt, type

'diskcopy a: b:'<RETURN>.

The computer will tell the user to insert the FIBRAM program disk (source disk, disk A) into drive A and the backup disk (target disk, disk B) into drive B. The user should do so and then "press any key to begin copying." Remove the disks from the drives when the computer indicates the copying is done. Put this backup copy (disk B) in a safe place.

B.1.2 Copy the Files From FIBRAM

Insert the FIBRAM program disk into drive A and the second blank, formatted disk (disk C) into drive B. At the DOS prompt, type

'copy a:\fortran*.for b:'<RETURN>.

This will copy all the Fortran programs from disk A to disk C. When the computer indicates it is done copying, type

'copy a:\objects*.obj b:'<RETURN>.

This will copy all of the Fortran object files from disk A to disk C. Remove the FIBRAM program disk from drive A and place it in a safe place.

B.1.3 Editing the Fortran Programs

The user should now use any text editor or word processing program with text editing capabilities to modify as many of the Fortran programs as needed. These files will have the extension ".FOR". The user should be sure to replace these programs onto disk C when modifications are completed. Appendix B.2 contains more specific information on what changes need to be made for particular program modifications.

B.1.4 Compiling and Linking the Program

Once the user has completed modifications to the Fortran programs, these programs need to be compiled and linked to create a new executable copy of FIBRAM. Instructions on how to use the compiler can be found in the user manual for the specific brand of Fortran compiler the user has chosen. The user should note that only Fortran programs that have been modified in Appendix B.1.3 need to be recompiled.

Linking the programs, on the other hand, requires object files pertaining to all Fortran programs and subroutines used by FIBRAM, regardless of whether or not they have been presently modified. These object files have an extension of ".OBJ". When specifying which subroutines are to be linked by the linking program, FIBRAM.OBJ should be listed first. This should result in the executable program being named FIBRAM.EXE. Other, more specific, linking instructions can be found in the user manual for the user's brand of linking program.

B.1.5 Copy the Modified Files Onto the FIBRAM Diskette

Insert the FIBRAM program disk (disk A) into drive A and the second disk with the modified programs on it (disk C) into drive B. At the DOS prompt, type

'copy b:*.for a:\fortran\'<RETURN>.

This will copy the Fortran programs back onto disk A. When the computer indicates it has completed the copying, type

'copy b:*.obj a:\object\'<RETURN>.

This will copy the modified object files back onto disk A. When the computer indicates it has finished copying, type

'copy b:fibram.exe a:\'<RETURN>.

This will copy the executable FIBRAM program back onto disk A. Remove the FIBRAM program disk from drive A. This is now a working copy of FIBRAM with the new modifications added.

B.1.6 Send the Author a Copy of the Modified Program

Once the user has modified the program to accommodate new circumstances, a copy of the modified Fortran programs (preferably on floppy disk) should be sent along with a concise explanation of the changes to:

William Ingram
Mail Code: ITS.N1
U.S. Department of Commerce, NTIA
325 Broadway
Boulder, CO 80303-3328

These modifications are important to the author so that the best and most up-to-date versions of FIBRAM can be distributed to those users who need them.

B.2 Specific Kinds of Modifications

Each of the following sections contains some specific guidelines for the user who may wish to modify FIBRAM. It would be overwhelming to attempt to list and explain every modification that every user may have occasion to make. In that light, only some of the more obvious and useful changes are explained in this section.

B.2.1 Changing the Number of Fiber Types

When changing the number of fiber types or the data contained in one of the current types, the steps outlined above need not be followed. If the user needs only to change or add data to current fiber data files, any text editing program may be used, as long as the present spacing format is conserved.

If the user wishes to add a whole new type of optical fiber, the user must obtain a graph or other tabular data showing the induced loss of the new fiber (dB/km) as a function of the time after exposure (seconds). The time after exposure values must be the logarithm of the actual times. The user should then run the program on the FIBRAM disk called FTYPES by typing 'FTYPES' <RETURN> at the DOS prompt. The first value that FTYPES will ask for is the total gamma radiation dosage at which these loss measurements were made. After that, FTYPES will expect the user to enter the logarithm of the time as the "X" value and the induced loss as the "Y" value. When the user is done, a value of "X" which is lower than the most previously entered "X" value should be entered to indicate no more data is available. FTYPES will then ask for a file name in which to store these data. This name should have a file extension of ".FTY".

With any text editing program, the disk file FIBTYPES.TXT should be edited to reflect the name of the disk file containing the data for the new fiber type, making sure that the file format remains unchanged.

B.2.2 Change the Number of Photodiode Types

This section should be consulted when the user decides to change the parameters of an existing type of photodiode (e.g., name, bit-error-ratio vs. receive signal power curves, etc.) or add completely new parameters for a new type of photodiode. The specific Fortran files that may need to be changed are:

READBER, GRAPHR, and PHOTYPES.TXT.

Subroutine READBER will "read" a graph of the form shown earlier in Figures 6 and 15. It will estimate the bit error ratio when given the receive signal power and the current dosage rate. Subroutine GRAPHR does the actual reading of the graph, whereas READBER determines the variables to run GRAPHR correctly.

Currently READBER allows only two photodiode types. They are 1) a silicon Avalanche Photodiode (APD) 11A type and 2) an indium gallium arsenide PIN photodiode 11D, 11E, or 11F type. For each photodiode type, READBER contains statements similar to the following:

```
"       if(photype.eq.1)then"                                    (B-1)
"       ymax=-19."
"       ymin=-42.5"
"       fname='ber-11a.gph'"
"       end if".
```

These statements represent the assignment of the individual parameters for the photodiode (in this case photodiode type #1). Ymax and ymin, respectively, are the maximum and minimum allowed receive signal power levels and are described earlier in Section 3.3.7. Fname is the file name where the graphical data are stored. The user may modify these existing values if different or more accurate values are needed. The actual graphical data stored in fname can be modified by following the format and procedures outlined in Section 3.2.4.

If the user is adding data for new types of photodiodes, new program sections similar to (B-1) should be created and added to READBER for each new type to be added. Also, for each new type, the user must create a new data file as described in Section 3.2.4 and save it with the file name and extension (fname) used in the new program section.

Subroutine GRAPHR will probably not need to be modified. It uses standard interpolation methods to "read" the graphical data specified by READBER (fname). If the user decides to replace or modify these methods, the subroutine is documented with comment statements that would allow the user quickly to find the portions that need to be changed.

The textual file PHOTYPES.TXT should be modified by the user to reflect any changes or additions to the number, names, or characteristics of the available photodiode types. Most commercially available text editors and some word processing packages can help the user update these files quickly and easily. The present format of the text file must not be changed. The current text file is shown in Appendix F.5.5.

B.2.3 Adding New Variables

It may become desirable for the user to include several new variables not currently included. One of these variables could be temperature. Current theories suggest that temperature has only a slight effect on the accuracy of the results. These effects are small enough that they can be neglected for most studies and are therefore not included in this report (NCS, 1985). A methodology to add these new variables, however, is included here in case the user finds it necessary.

Adding a new variable is a complex prospect, but can be represented by the following four steps:

1. Determine which factors will be affected.
2. Determine how each factor is affected.
3. Change the subroutines.
4. Change the associated text files.

B.2.4 Changing the Methods of Calculating Losses and Radiation Rates

The user may find a simpler or more accurate method for calculating the intrinsic or darkening losses in the optical fiber cable or the user may find a better way to calculate the radiation dosage rates. The subroutines that would need to be modified in these cases are XTRINS, LOSS, DOSERA, and GAMMAEX, respectively.

Subroutines XTRINS, LOSS, DOSERA, and GAMMAEX return calculated variables indirectly through subroutine common statements.

If the user decides to write his/her own subroutine, the five "common" statements, included in the program listing for FIBRAM (Appendix F.1), should begin the subroutine. This is so that the user can take advantage of many variables previously calculated by FIBRAM or its subroutines. Before the user makes changes to any subroutine, he/she should become familiar with the present structure of the subroutine so that the proper variables may be calculated at the proper times.

B.2.5 Other Modifications

Other modifications to FIBRAM and its subroutines are possible. Program listings for all subroutines are included in Appendix F. Section 3 contains explanations of every subroutine included in FIBRAM.

APPENDIX C. VARIABLES USED BY FIBRAM

Variable	Variable Description
BER_{sub}	The bit error ratio of the sublink (dimensionless).
FT_i	The type of fiber that composes sublink section number i.
k	The number of iterations used to simulate the integral in (23)
l_i	The length of each individual sublink section (meters). The sum of l_i over the entire sublink is l.
$L_{gi}(t)$	The actual radiation darkening loss of each sublink section.
$L_{Gi}(t)$	The induced (measured) radiation darkening loss of each sublink section. It is equivalent to $L_{gi}(t) * R_i(t)$.
L_{in}	The intrinsic loss of the entire sublink (dB).
$L_{\Gamma}(t)$	The darkening loss of the entire sublink (dB).
$L_{\Gamma i}(t)$	The darkening loss of each sublink section (dB).
P	The average optical energy.
PF_i	The protection factor of each sublink section (dimensionless).
PF_{pd}	The protection factor of the photodiode (dimensionless).
$P_r(t)$	The receive signal power of the sublink (dBm).
P_t	The transmitted power of the sublink (dBm).
$R_i(t)$	The recovery factor of each sublink section (dimensionless).
t	Time (seconds)
T_i	The intrinsic loss factor for each sublink section.
$\gamma_i(t)$	The rate of radiation exposure for each sublink section (rad/second).
$\gamma_i'(t)$	The exposed rate of radiation exposure for each sublink section (rad/second). It is equivalent to $\gamma_i(t) / PF_i$.

γ_{maxi}	The maximum rate of radiation exposure for each sublink section (rad/second).
$\gamma_{pd}(t)$	The gamma radiation dosage rate at the photodiode.
$\Gamma_i(t)$	The accumulated total of exposed radiation for each sublink section (rad).
$\Gamma_i'(t)$	The exposed accumulated total of exposed radiation for each sublink section (rad). It is equivalent to $\Gamma_i(t)$ / PF_i.
Γ_{max}	The maximum accumulated radiation dosage for each sublink section (rad).
Ω	The gamma radiation dosage rate at which the induced loss curves were measured.

APPENDIX D. FILES ON THE FIBRAM PROGRAM DISK

The following is a list of each data file included on the FIBRAM program disk. Each item is accompanied by a short description.

DATA FILE NAME	FILE DESCRIPTION
FIBRAM.EXE	This is the executable FIBRAM file. It has been compiled form the Fortran files listed below.
*.FOR	These are the actual Fortran programs written in Fortran 77.
*.OBJ	These are the object files created from the Fortran programs above. These files are linked together with a linking program (not included) to form the executable version of FIBRAM.
*.TXT	These are the textual information files available to help the user decide what inputs are expected at each input. Each of these files is called internally by the subroutine FREAD.
SM-13.DAT	This file contains data specific for single-mode, 1.3 micron, optical fiber systems.
OUTPUTβ.PRN	This file contains the output results from the test run numbered β. There are five such output files and each is ready to be incorporated into most graphical charting packages.
G50-5000.RNG	This file contains the attributes used in Appendix E.2 to tell FIBRAM which variables are to be ranged, what values these ranged variables will assume, and where the results of those runs will be stored. In this case, the user has decided to range the gamma radiation exposure between 500 and 5,000 rads.

APPENDIX E. SAMPLE RUNS OF FIBRAM

E.1 Modifying a Previously Entered Data File

This appendix section is to be examined in conjunction with Section 4.1 of this manual.

A><u>fibram</u><RETURN>

Main Menu for

The help function is OFF
The Autodisplay function is OFF

m=Modify, t=Timing, h=Help, n=Name, s=Save
l=Load, r=Run, x=eXit, p=Print, d=Display
g=loGarithmic, >=Range

Please enter a choice from this menu
<u>l</u><RETURN>
Please enter a filename
<u>sm-13.dat</u><RETURN>

Main Menu for AT&T Single Mode - 1.3 mm

The help function is OFF
The Autodisplay function is ON

m=Modify, t=Timing, h=Help, n=Name, s=Save
l=Load, r=Run, x=eXit, p=Print, d=Display
g=loGarithmic, >=Range

Please enter a choice from this menu
<u>r</u><RETURN>

```
*** Now calculating at time       501.00
Calculating for section No.  1
Gamma ex= .5537E+00 Dose rate= .1110E+00 Loss= .1016E-06
Calculating for section No.  2
Gamma ex= .3164E-03 Dose rate= .6341E-04 Loss= .5807E-10
Calculating for section No.  3
Gamma ex= .2215E+01 Dose rate= .4439E+00 Loss= .4065E-06
 Total darkening is  .0002411 dBm
 Rec sig pow=  -35.00     Gamma=    .2215     Drate=    .4439E-01
  Bit error ratio is .880855E-08


*** Now calculating at time       502.00
Calculating for section No.  1
Gamma ex= .6644E+00 Dose rate= .1107E+00 Loss= .2031E-06
Calculating for section No.  2
Gamma ex= .3797E-03 Dose rate= .6326E-04 Loss= .1161E-09
```

```
Calculating for section No.  3
Gamma ex= .2658E+01 Dose rate= .4428E+00 Loss= .8124E-06
 Total darkening is  .0004818 dBm
 Rec sig pow=  -35.00     Gamma=    .2658     Drate=    .4428E-01
  Bit error ratio is .878301E-08

.
.
.

*** Now calculating at time  90000500.00
Calculating for section No.  1
Gamma ex= .2535E+03 Dose rate= .5494E-07 Loss= .1038E-04
Calculating for section No.  2
Gamma ex= .1448E+00 Dose rate= .3140E-10 Loss= .5932E-08
Calculating for section No.  3
Gamma ex= .1014E+04 Dose rate= .2198E-06 Loss= .4152E-04
 Total darkening is  .0246290 dBm
 Rec sig pow=  -35.02     Gamma=   101.4     Drate=    .2198E-07
  Bit error ratio is .202477E-09


*** Now calculating at time 100000500.00
Calculating for section No.  1
Gamma ex= .2540E+03 Dose rate= .4842E-07 Loss= .1009E-04
Calculating for section No.  2
Gamma ex= .1451E+00 Dose rate= .2767E-10 Loss= .5764E-08
Calculating for section No.  3
Gamma ex= .1016E+04 Dose rate= .1937E-06 Loss= .4035E-04
 Total darkening is  .0239330 dBm
 Rec sig pow=  -35.02     Gamma=   101.6     Drate=    .1937E-07
  Bit error ratio is .201364E-09


    Main Menu for AT&T Single Mode - 1.3 mm

    The help function is OFF
    The Autodisplay function is ON

m=Modify,t=Timing,h=Help,n=Name,s=Save
l=Load,r=Run,x=eXit,p=Print,d=Display
g=loGarithmic,>=Range

Please enter a choice from this menu
g<RETURN>
Please enter a file name to save results in
output1.prn<RETURN>
```

```
    Main Menu for AT&T Single Mode - 1.3 mm

    The help function is OFF
    The Autodisplay function is ON

m=Modify,t=Timing,h=Help,n=Name,s=Save
l=Load,r=Run,x=eXit,p=Print,d=Display
g=loGarithmic,>=Range

Please enter a choice from this menu
m<RETURN>

These are the current values;
  Help is off.  Autodisplay is  on.
Total power transmitted is    .0104700 milliwatts
Maximum allowed gamma for a PD of type 2 is    1500.000 rads
The start and end times are   500.00 and     90000500.00 s
The initial spacing between time steps is     1.00 s
The output will be offset by     500.000 seconds

Sect   Length (m)       PF  Gamma (rad)  F-Type  Intrins
  1         30.0       4.0    1000.0       1      .380000
  2      39390.0    7000.0    1000.0       1      .380000
  3        580.0       1.0    1000.0       1      .380000
  4   photodiode      10.0    1000.0

  HELP is off,  Autodisplay is  on.

Type "1." to have HELP on
<RETURN>
Type "1." to have Autodisplay on
1.<RETURN>

 1. 11-a
 2. 11-d, e, and f
 3.
 4.
 5.
 6.
 7.
 8.
 9.
10.

The current photodiode type is 2. Enter new type
<RETURN>
Transmit power is  .0104700 mwatts. Enter new power
.007<RETURN>
Max gamma for the PD is  1500.00. Enter new max gamma
<RETURN>
The current number is   3. Enter new number of sections
<RETURN>
```

```
Sect   Length (m)        PF  Gamma (rad)  F-Type  Intrins
  1          30.0       4.0     1000.0      1     .380000
  2       39390.0    7000.0     1000.0      1     .380000
  3         580.0       1.0     1000.0      1     .380000
  4    photodiode      10.0     1000.0

Which section No. to change (0=end, -1=all)?
<RETURN>


    Main Menu for AT&T Single Mode - 1.3 mm

    The help function is OFF
    The Autodisplay function is ON

m=Modify,t=Timing,h=Help,n=Name,s=Save
l=Load,r=Run,x=eXit,p=Print,d=Display
g=loGarithmic,>=Range

Please enter a choice from this menu
r<RETURN>

*** Now calculating at time       501.00
Calculating for section No.  1
Gamma ex= .5537E+00 Dose rate= .1110E+00 Loss= .1016E-06
Calculating for section No.  2
Gamma ex= .3164E-03 Dose rate= .6341E-04 Loss= .5807E-10
Calculating for section No.  3
Gamma ex= .2215E+01 Dose rate= .4439E+00 Loss= .4065E-06
 Total darkening is  .0002411 dBm
 Rec sig pow=  -36.75     Gamma=    .2215     Drate=   .4439E-01
  Bit error ratio is .211179E-07


*** Now calculating at time       502.00
Calculating for section No.  1
Gamma ex= .6644E+00 Dose rate= .1107E+00 Loss= .2031E-06
Calculating for section No.  2
Gamma ex= .3797E-03 Dose rate= .6326E-04 Loss= .1161E-09
Calculating for section No.  3
Gamma ex= .2658E+01 Dose rate= .4428E+00 Loss= .8124E-06
 Total darkening is  .0004818 dBm
 Rec sig pow=  -36.75     Gamma=    .2658     Drate=   .4428E-01
  Bit error ratio is .210577E-07

.
.
.

*** Now calculating at time 100000500.00
Calculating for section No.  1
Gamma ex= .2540E+03 Dose rate= .4842E-07 Loss= .1009E-04
```

```
Calculating for section No.  2
Gamma ex= .1451E+00 Dose rate= .2767E-10 Loss= .5764E-08
Calculating for section No.  3
Gamma ex= .1016E+04 Dose rate= .1937E-06 Loss= .4035E-04
 Total darkening is  .0239330 dBm
 Rec sig pow=  -36.77     Gamma=   101.6     Drate=   .1937E-07
  Bit error ratio is .368698E-09


    Main Menu for AT&T Single Mode - 1.3 mm

    The help function is OFF
    The Autodisplay function is ON

m=Modify,t=Timing,h=Help,n=Name,s=Save
l=Load,r=Run,x=eXit,p=Print,d=Display
g=loGarithmic,>=Range

Please enter a choice from this menu
g<RETURN>
Please enter a file name to save results in
output3.prn<RETURN>


    Main Menu for AT&T Single Mode - 1.3 mm

    The help function is OFF
    The Autodisplay function is ON

m=Modify,t=Timing,h=Help,n=Name,s=Save
l=Load,r=Run,x=eXit,p=Print,d=Display
g=loGarithmic,>=Range

Please enter a choice from this menu
x<RETURN>

A>
```

E.2 Using the RANGER Feature

This appendix section is to be examined in conjunction with Section 4.3 of this manual. It depicts a user's interaction with FIBRAM to model a fiber optic system over a range of different gamma radiation exposure levels.

```
A>fibram<RETURN>


    Main Menu for

    The help function is OFF
    The Autodisplay function is OFF

m=Modify,t=Timing,h=Help,n=Name,s=Save
l=Load,r=Run,x=eXit,p=Print,d=Display
g=loGarithmic,>=Range

Please enter a choice from this menu
><RETURN>
Enter the base variable file name
sm-13.dat<RETURN>

These are the current values;
  Help is off.  Autodisplay is  on.
Total power transmitted is    .0104700 milliwatts
Maximum allowed gamma for a PD of type 2 is    1500.000 rads
The start and end times are   500.00 and    90000500.00 s
The initial spacing between time steps is    1.00 s
The output will be offset by     500.000 seconds


Sect   Length (m)       PF  Gamma (rad)  F-Type  Intrins
  1          30.0      4.0     1000.0       1     .380000
  2       39390.0   7000.0     1000.0       1     .380000
  3         580.0      1.0     1000.0       1     .380000
  4   photodiode      10.0     1000.0


Choose a variable to specify a range for:

1. Transmitted power    2. Maximum gamma for region
3. Intrinsic loss

2.<RETURN>
Enter the number of variable files to process (0=      previous file)
0.<RETURN>
Enter the file name of previous "Range" variables:
g50-5000.rng<RETURN>

File No.       Value     Storage file
No. 1      500.0000000   output4.prn
No. 2     1000.0000000   output1.prn
No. 3     5000.0000000   output5.prn

Do you want to start over?
n<RETURN>
```

```
-----Maximum gamma (rads)=      500.00


*** Now calculating at time       501.00
Calculating for section No.  1
Gamma ex= .2769E+00 Dose rate= .5548E-01 Loss= .5081E-07
Calculating for section No.  2
Gamma ex= .1582E-03 Dose rate= .3170E-04 Loss= .2903E-10
Calculating for section No.  3
Gamma ex= .1107E+01 Dose rate= .2219E+00 Loss= .2032E-06
 Total darkening is  .0001205 dBm
 Rec sig pow=  -35.00     Gamma=    .1107     Drate=    .2219E-01
  Bit error ratio is .366879E-08


*** Now calculating at time       502.00
Calculating for section No.  1
Gamma ex= .3322E+00 Dose rate= .5535E-01 Loss= .1015E-06
Calculating for section No.  2
Gamma ex= .1898E-03 Dose rate= .3163E-04 Loss= .5803E-10
Calculating for section No.  3
Gamma ex= .1329E+01 Dose rate= .2214E+00 Loss= .4062E-06
 Total darkening is  .0002409 dBm
 Rec sig pow=  -35.00     Gamma=    .1329     Drate=    .2214E-01
  Bit error ratio is .365794E-08

.
.
.

*** Now calculating at time 100000500.00
Calculating for section No.  1
Gamma ex= .1270E+03 Dose rate= .2421E-07 Loss= .5044E-05
Calculating for section No.  2
Gamma ex= .7257E-01 Dose rate= .1383E-10 Loss= .2882E-08
Calculating for section No.  3
Gamma ex= .5080E+03 Dose rate= .9683E-07 Loss= .2018E-04
 Total darkening is  .0119665 dBm
 Rec sig pow=  -35.01     Gamma=   50.80     Drate=    .9683E-08
  Bit error ratio is .194850E-09

-----Maximum gamma (rads)=     1000.00


*** Now calculating at time       501.00
Calculating for section No.  1
Gamma ex= .5537E+00 Dose rate= .1110E+00 Loss= .1016E-06
Calculating for section No.  2
Gamma ex= .3164E-03 Dose rate= .6341E-04 Loss= .5807E-10
Calculating for section No.  3
Gamma ex= .2215E+01 Dose rate= .4439E+00 Loss= .4065E-06
```

```
 Total darkening is  .0002411 dBm
 Rec sig pow=  -35.00     Gamma=   .2215     Drate=   .4439E-01
  Bit error ratio is .880855E-08


*** Now calculating at time       502.00
Calculating for section No.  1
Gamma ex= .6644E+00 Dose rate= .1107E+00 Loss= .2031E-06
Calculating for section No.  2
Gamma ex= .3797E-03 Dose rate= .6326E-04 Loss= .1161E-09
Calculating for section No.  3
Gamma ex= .2658E+01 Dose rate= .4428E+00 Loss= .8124E-06
 Total darkening is  .0004818 dBm
 Rec sig pow=  -35.00     Gamma=   .2658     Drate=   .4428E-01
  Bit error ratio is .878301E-08

.
.
.

*** Now calculating at time 100000500.00
Calculating for section No.  1
Gamma ex= .2540E+03 Dose rate= .4842E-07 Loss= .1009E-04
Calculating for section No.  2
Gamma ex= .1451E+00 Dose rate= .2767E-10 Loss= .5764E-08
Calculating for section No.  3
Gamma ex= .1016E+04 Dose rate= .1937E-06 Loss= .4035E-04
 Total darkening is  .0239330 dBm
 Rec sig pow=  -35.02     Gamma=   101.6     Drate=   .1937E-07
  Bit error ratio is .201364E-09

-----Maximum gamma (rads)=     5000.00


*** Now calculating at time       501.00
Calculating for section No.  1
Gamma ex= .2769E+01 Dose rate= .5548E+00 Loss= .5081E-06
Calculating for section No.  2
Gamma ex= .1582E-02 Dose rate= .3170E-03 Loss= .2903E-09
Calculating for section No.  3
Gamma ex= .1107E+02 Dose rate= .2219E+01 Loss= .2032E-05
 Total darkening is  .0012055 dBm
 Rec sig pow=  -35.00     Gamma=   1.107     Drate=   .2219
  Bit error ratio is .872997E-07


*** Now calculating at time       502.00
Calculating for section No.  1
Gamma ex= .3322E+01 Dose rate= .5535E+00 Loss= .1015E-05
Calculating for section No.  2
Gamma ex= .1898E-02 Dose rate= .3163E-03 Loss= .5803E-09
Calculating for section No.  3
```

```
Gamma ex= .1329E+02 Dose rate= .2214E+01 Loss= .4062E-05
 Total darkening is  .0024092 dBm
 Rec sig pow=  -35.00      Gamma=    1.329      Drate=    .2214
  Bit error ratio is .870231E-07

.
.
.

*** Now calculating at time 100000500.00
Calculating for section No.  1
Gamma ex= .1270E+04 Dose rate= .2421E-06 Loss= .5044E-04
Calculating for section No.  2
Gamma ex= .7257E+00 Dose rate= .1383E-09 Loss= .2882E-07
Calculating for section No.  3
Gamma ex= .5080E+04 Dose rate= .9683E-06 Loss= .2018E-03
 Total darkening is  .1196650 dBm
 Rec sig pow=  -35.12      Gamma=    508.0      Drate=    .9683E-07
  Bit error ratio is .223303E-09


    Main Menu for AT&T Single Mode - 1.3 mm

    The help function is OFF
    The Autodisplay function is ON

m=Modify,t=Timing,h=Help,n=Name,s=Save
l=Load,r=Run,x=eXit,p=Print,d=Display
g=loGarithmic,>=Range

Please enter a choice from this menu
x<RETURN>

A>
```

APPENDIX F. SOURCE CODE LISTINGS

The following programs and subprograms are listed in Table 1 and explained in Section 3.

The informational text files are also included in this section. Each of these text files can be viewed while running FIBRAM by invoking the help feature.

F.1 Program FIBRAM

The following program is called FIBRAM and is the figurehead of the FIBRAM system. It sets up the "common" statements and calls MENU and RUNTIME to do the actual calculating.

```
-----------------------------------------------------------------
      program fibram

****** Optical fiber radiation attenuation model ******

********************************+********************************
*       ***********************************************+**      *
*    ***************+********      ********+*****************   *
*  *********+************************************************** *
*********                                               *********
*******          This program was written by:             ****+**
***+**                                                     ******
******                   William Ingram                    ******
******       Institute for Telecommunications Sciences     ******
******                Department of Commerce               ******
****+*                     Boulder, CO                     ******
******                                                     ******
**+***                     April, 1987                     *+****
******                                                     ******
******   It is based upon National Communications System   ******
****** Technical Information Bulletin 85-11, Assessment of ******
****** Nuclear Fallout Radiation on FT3C Lightwave Digital ******
******         Transmission System, November 1985.         ******
***+**                                                     ******
****** It is designed to allow a user to input variables   ****+*
******  associated with a fiber optic communications link  ******
******    and to produce an output showing the bit error   ******
*+****    ratio as a function of time after exposure to    ******
*******                gamma radiation.                   *****+*
****+****                                               *********
*  ******************+***************************************** *
*    +***************+*******      ****************+********    *
*       *******************************+******************      *
***********+*****************************************************
```

```
****** Introductions and pleasantries ******

      integer photype
      character fname*30,ans*1,time*10,spr*10,title*40,fn*30

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /d/tstart,tend,tstep,gstart,nsteps
      common /e/title

****** Main menu

  100 write (*,500)title
      if(help.eq.0.)write (*,525)
      if(help.eq.1.)write (*,530)
      if(displ.eq.0.)write (*,535)
      if(displ.eq.1.)write (*,540)

      if(help.eq.0.)then
      write (*,550)
      write (*,552)
      write (*,554)
      go to 199
      end if

      write (*,530)
      fn='mainmenu.txt'
      call fread(fn)

  199 write (*,600)
  200 call entera(choice)

****** Examine choice of user

      if(choice.eq.'r') call runtime
      if(choice.eq.'x') goto 9999

      if(choice.eq.'h')then
      help=1.-help
      if(help.eq.0.)go to 223
      fname='intro.txt'
      call fread(fname)
  223 go to 200
      end if

      if(choice.eq.'n')then
      write (*,510)
      call entera(title)
      end if
```

```
      if(choice.eq.'p')then
      call prdata
      go to 200
      end if
      if(choice.eq.'>')call ranger

      if(choice.eq.'g')then
      write (*,560)
      call entera(fname)
      call logdata(fname)
      end if

      if(choice.eq.'s')then
      write (*,520)
      call entera(fname)
      call savdata(fname)
      end if

      if(choice.eq.'l')then
      write (*,520)
      call entera(fname)
      call getdata(fname)
      end if

      if(choice.eq.'m')call modata
      if(choice.eq.'d')call disply
      if(choice.eq.'t')call timing
      go to 100

  500 format(///'      Main Menu for ',a40)
  600 format(/' Please enter a choice from this menu')
  510 format(/' Please enter a new title for these variables')
  520 format(/' Please enter a filename')
  525 format(/'     The help function is OFF')
  530 format(/'     The help function is ON')
  535 format('     The Autodisplay function is OFF'/)
  540 format('     The Autodisplay function is ON'/)
  550 format(' m=Modify,t=Timing,h=Help,n=Name,s=Save')
  552 format(' l=Load,r=Run,x=eXit,p=Print,d=Display')
  554 format(' g=loGarithmic,>=Range')
  560 format(/' Please enter a file name to save results in')

 9999 end
```

-=-

F.2 Framework Subroutines

-=-

F.2.1 Subroutine ENTERD

```
      subroutine enterd (idata,ipd)

****** Enter data for each section ***********

      character fname*30
      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)

      if(ipd.ne.1)write (*,503) idata
      if(ipd.eq.1)write (*,504)

****** Get length.

      if(ipd.eq.1)go to 100
      write (*,500) xlength(idata)
      call enterx(xleng)
      if(xleng.eq.0)go to 100
      xlength(idata)=xleng

****** Get protection factor.

  100 write (*,502) pf(idata)
      call enterx(xpf)
      if(xpf.eq.0)go to 102
      pf(idata)=xpf

****** Get maximum gamma

  102 write (*,501) gammamax(idata)
      call enterx(xg)
      if(xg.eq.0)go to 103
      gammamax(idata)=xg

****** Get The Fiber Type.

  103 if(ipd.eq.1)go to 108
      fname='fibtypes.txt'
      call fread(fname)

      write (*,510) nfibtype(idata)
      call enterx(xfib)
      if(xfib.eq.0)goto 108
      nfibtype(idata)=xfib
```

```
****** Get Intrinsic Fatcor.

  108 if(ipd.eq.1)go to 104
      write (*,520) trins(idata)
      call enterx(xtrin)
      if(xtrin.eq.0)go to 104
      trins(idata)=xtrin



  500 format(/' The length is ',f12.2,' meters.  Enter new length')
  501 format(/' The Maximum gamma is',f12.2,' rads.  Enter new gamma')
  502 format(/' The old PF was ',f12.2,'.  Enter new PF')
  503 format(//' Data for section ',i3)
  504 format(//' Data for the photodiode')
  510 format(/' The current Fiber Type is',i4,'.  Enter the new type')
  520 format(/' Intrinsic loss is ',f10.6,' db/km. Enter new loss.')

  104 return
      end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.2.2 Subroutine GETDATA

```
      subroutine getdat(fname)

****** Read in previous system variables **********

      character fname*30,title*40,buffer*40
      integer photype

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /d/tstart,tend,tstep,gstart,nsteps
      common /e/title

      open(8,file=fname,err=200)

      read(8,510)buffer
      read(8,515)powtransm,photype,displ,help
      read(8,510)buffer
      read(8,525)tstart,tend,tstep,gstart,nsteps
      read(8,510)buffer
      read(8,530)worstcase,title,gpmax
      read(8,510)buffer
      read(8,535)nsections

      do 103 i=1,nsections+1
  103 read(8,520) xlength(i),pf(i),gammamax(i),nfibtype(i),trins(i)
      close (8)
      return
```

```
  510 format(a40)
  515 format(f12.2,i5,2f12.2)
  520 format(3f12.2,i5,f12.8)
  525 format(4f12.2,i5)
  530 format(f18.9,a40,f12.2)
  535 format(i5)
  540 format(/' Oops, that file doesn't exist!')

****** Write default file if none exists

  200 write (*,540)
      return
      end
```

-=-

F.2.3 Subroutine MODATA

```
      subroutine modata

****** Modify/enter the system variables ************

      character fname*30
      integer photype,ptype

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase

****** Print out variables.

      call disply

*******  Change HELP and Autodisplay

      on=' on'
      off='off'

      xhelp = off
      if(help.eq.1.) xhelp = on
      xdisp = off
      if(displ.eq.1.) xdisp = on

      write (*,565) xhelp,xdisp
      write (*,566)
      call enterx(help)

  111 write (*,567)
      call enterx(displ)

****** Enter the type number of the photodiode.

  222 fname='photypes.txt'
      call fread(fname)
```

```
      write (*,630) photype
      call enterx(xptype)
      ptype=xptype
      if(ptype.le.0)goto 101

      photype=ptype

****** Enter the transmitted power.

  101 if(help.ne.1.)go to 114
      fname='worst.txt'
      call fread(fname)
  114 write (*,536) worstcase
      call enterx(aworst)
      if(aworst.eq.0.)go to 305
      powtransm=10*alog10(aworst)
      worstcase=aworst

****** Maximum gamma for a photodiode.

  305 write (*,531) gpmax
      call enterx(gmax)
      if(gmax.le.0.)go to 205
      gpmax=gmax

****** Enter the number of sections involved.

  205 if(help.ne.1.)go to 113
      fname='nsection.txt'
      call fread(fname)
  113 write (*,535) nsections
      call enterx(xnsect)
      nsect=xnsect
      if((nsect.lt.0).or.(nsect.gt.200))go to 101
      if(nsect.eq.0)go to 105
      nsections=nsect

****** Print out the data associated with each section.

  105 write (*,540)
      do 106 i=1,nsections

****** Change data if needed.

  106 write (*,545)i,xlength(i),pf(i),gammamax(i),nfibtype(i),trins(i)
      write (*,547)nsections+1,pf(nsections+1),gammamax(nsections+1)
      write (*,550)
      call enterx(xwhich)
      iwhich=xwhich
      if(iwhich.eq.0)go to 102
      if(iwhich.lt.0)then
      do 107 i=1,nsections
```

```
      call enterd(i,0)
  107 gammamax(i+1)=gammamax(i)
      nfibtype(i+1)=nfibtype(i)
      call enterd(nsections+1,1)
      go to 105
      end if
      if(iwhich.le.nsections)then
      call enterd(iwhich,0)
      go to 105
      end if
      call enterd(nsections+1,1)
      go to 105

  102 return

  531 format(/' Max gamma for the PD is ',f8.2,'. Enter new max gamma
  630 format(/' The current photodiode type is ',i1,'. Enter new type
  535 format(/' The current number is ',i3,'. Enter new number'
     *' of sections')
  536 format(/' Transmit power is ',f9.7,' mwatts. Enter new power')
  540 format(//' Sect   Length (m)      PF  Gamma (rad)  F-Type  Intr
     *')
  545 format(' ',i3,'    ',3f10.1,i7,f12.6)
  547 format(' ',i3,'    photodiode',2f10.1)
  550 format(/' Which section No. to change (0=end, -1=all)?')
  565 format(/'    HELP is ',a3,',  Autodisplay is ',a3,'.'/)
  566 format(' Type "1." to have HELP on')
  567 format(' Type "1." to have Autodisplay on')
      end
```
-=-

F.2.4 Subroutine RANGER

```
      subroutine ranger

      integer ftype,fibertype,photype,ptype,logr
      character choice*3,title*40,fname*30,save*30,base*30,ans*30

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase

      dimension val(30),save(30)

      if(help.ne.1.)go to 527
      fname='range.txt'
      call fread(fname)
```

```
****** Enter base variables file name

  527 write (*,550)
      call entera(base)
      call getdata(base)
      call disply

****** Enter the variable number to be changed

      write (*,605)
      write (*,610)
      write (*,615)
      call enterx(xans)
      ians=xans
      if(ians.eq.0)return

****** Enter Number of Files to Process

  201 write (*,500)
      call enterx(xnfiles)
      nfiles=xnfiles

****** If No. of files is 0, ask for name of previously saved file

      if(nfiles.eq.0)then
      write (*,515)
      call entera(fname)
      open(8,file=fname)
      read(8,520)nfiles,ians
      do 300 i=1,nfiles
  300 read(8,525) val(i),save(i)
      close (8)

****** Show current variable ranges and output file names

      write (*,531)
      do 301 i=1,nfiles
  301 write (*,535) i,val(i),save(i)
      write (*,530)
      call entera(ans)
      if(ans.eq.'y')go to 201

      go to 202
      end if
```

```
****** Enter variables and output file names manually

      do 200 i=1,nfiles
      write (*,505)
      call enterx(value)
      val(i)=value
      write (*,510)value
      call entera(fname)
  200 save(i)=fname

****** Ask to save newly entered data

      write (*,540)
      call entera(ans)
      if(ans.eq.'n')go to 202
      write (*,545)
      call entera(fname)
      open(8,file=fname,status='new')
      write (8,520)nfiles,ians
      do 400 i=1,nfiles
  400 write (8,525) val(i),save(i)

****** Run the ranging portion

  202 do 117 j=1,nfiles

      if(ians.eq.1)then
      if(displ.eq.1.) write (*,900)val(j)
      worstcase=val(j)
      powtransm=10*alog10(worstcase)
      go to 111
      end if

      if(ians.eq.2)then
      if(displ.eq.1.) write (*,700)val(j)
      do 30 i=1,nsections+1
   30 gammamax(i)=val(j)
      go to 111
      end if

      if(ians.eq.3)then
      if(displ.eq.1.) write (*,800)val(j)
      do 44 i=1,nsections
   44 trins(i)=val(j)
      go to 111
      end if

  111 call runtime

  112 fname=save(j)
      call logdata(fname)
```

```
  117 continue
      call getdata(base)
      return

  500 format(/' Enter the number of variable files to process (0=
     *previous file)')
  505 format(/' Enter the first/next variable value')
  510 format(/' Enter the file name to save the results of ',f15.7)
  515 format(/' Enter the file name of previous "Range" variables:')
  520 format(2i2)
  525 format(2a30)
  530 format(/' Do you want to start over?')
  531 format(/' File No.      Value     Storage file')
  535 format(' No.',i2,'  ',f15.7,'  ',a30)
  540 format(/' Do you wish to save these names for later use?')
  545 format(/' Well then, enter a file name for them')
  550 format(/' Enter the base variable file name')
  605 format(//' Choose a variable to specify a range for:')
  610 format(/' 1. Transmitted power   2. Maximum gamma for region')
  615 format(' 3. Intrinsic loss')
  900 format(/' -----Transmitted power (mwatts)=',f15.7)
  700 format(/' -----Maximum gamma (rads)=',f12.2)
  800 format(/' -----Intrinsic cable loss (dB/km)=',f15.7)
 9999 end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.2.5 Subroutine RUNTIME

```
      subroutine runtime

****** Subroutine that actually runs the model ******

****** Introductions and pleasantries ******

      integer photype
      character fname*30,ans*1,time*10,spr*10

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /d/tstart,tend,tstep,gstart,nsteps
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime

****** Calculate loss factors (not time-dependent) ******

  108 call xtrins

****** Timing Loop to Calculate BER Over Range ******

      ex=int(alog10(tstep)+.5)

      nsteps=1
      xtime=tstart
```

```
 21 xtime=xtime+10**ex
    if(nsteps.lt.10)go to 44
    x10=float(nsteps-1)
    y10=mod(x10,9.)
    if(y10.eq.0)ex=ex+1

 44 if(displ.eq.1) write (*,600)xtime
    call darken

    call sigpow

    call readber(errrate)

    if(displ.eq.1) write (*,574)recsig,gxposed,drate
    times(nsteps)=xtime-gstart
    if(displ.eq.1) write (*,777)errrate
    biter(nsteps)=errrate

    if(xtime.gt.tend)return
    nsteps=nsteps+1
    go to 21

574 format('  Rec sig pow=',g12.4,' Gamma=',g12.4,' Drate=',g12.4)
600 format(//' *** Now calculating at time ',f12.2)
777 format('   Bit error ratio is',e12.6)
9999 end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.2.6 Subroutine SAVDATA

```
      subroutine savdata(fname)

******* Read in previous system variables **********

      character fname*30,title*40,buffer*40
      integer photype

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /d/tstart,tend,tstep,gstart,nsteps
      common /e/title

      open(8,file=fname,status='new')

      write(8,510)
      write(8,515)powtransm,photype,displ,help
      write(8,511)
      write(8,525)tstart,tend,tstep,gstart,nsteps
      write(8,512)
      write(8,530)worstcase,title,gpmax
      write(8,513)
      write(8,535)nsections
```

```
      do 103 i=1,nsections+1
  103 write(8,520) xlength(i),pf(i),gammamax(i),nfibtype(i),trins(i)
      close (8)
      return

  510 format(' Power,PType,Autodisp,Help')
  511 format(' Timing Parameters')
  512 format(' Worst Case,Title,Max Photo Gamma')
  513 format(' No. of Sections and Parameters')
  515 format(f12.2,i5,2f12.2)
  520 format(3f12.2,i5,f12.8)
  525 format(4f12.2,i4)
  530 format(f18.9,a40,f12.2)
  535 format(i5)

      end
```

-=-

F.2.7 Subroutine TIMING

```
      subroutine timing

      character fname*30
      integer photype

      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /d/tstart,tend,tstep,gstart,nsteps

****** Set timing variables ******

      if(help.ne.1.)go to 115
  108 fname='time.txt'
      call fread(fname)

  115 write (*,560) tstart
      write (*,561)
      call enterx(tstart)

      write (*,565) tend
      write (*,566)
      call enterx(xx)
      if(xx.ne.0.)tend=xx
      if(tstart.ge.tend)go to 108

      write (*,568) gstart
      write (*,569)
      call enterx(gstart)

      write (*,575) tstep
      write (*,576)
      call enterx(xexp)
      tstep=10**xexp
      return
```

```
560 format(//' The current starting time is ',f12.2,' seconds.')
561 format(' Enter a new starting time (<return> means "0")')
565 format(//' The current ending time is ',f12.2,' seconds.')
566 format(' Enter a new ending time')
568 format(//' The current off-set time is ',f12.2,' seconds.')
569 format(' Enter a new off-set time (<return> means "0")')
575 format(//' The current timing step size is ',f12.2,' seconds.')
576 format(' Enter a power of 10 for the initial timing step size (<ret
   *urn> is "0")')

    return
    end
```

-=-

F.3 Utility Subroutines

-=-

F.3.1 Subroutine DISPLY

```
      subroutine disply (title)

      character fname*30,title*40,xhelp*3,xdisp*3,on*3,off*3
      integer photype

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /d/tstart,tend,tstep,gstart,nsteps

      on=' on'
      off='off'
      xhelp = off
      if(help.eq.1.) xhelp = on
      xdisp = off
      if(displ.eq.1.) xdisp = on

   77 write (*,560)
      write (*,565) xhelp,xdisp
      write (*,571) worstcase
      write (*,574) photype,gpmax
      write (*,575) tstart,tend
      write (*,579) tstep
      write (*,587) gstart

****** Print out the data associated with each section.

  105 write (*,540)
      do 106 i=1,nsections
  106 write (*,545)i,xlength(i),pf(i),gammamax(i),nfibtype(i),trins(i)
      write (*,547)nsections+1,pf(nsections+1),gammamax(nsections+1)
  102 return
```

```
  540 format(//' Sect   Length (m)      PF  Gamma (rad)  F-Type  Intrins
     *')
  545 format(' ',i3,'    ',3f10.1,i7,f12.6)
  547 format(' ',i3,'    photodiode',2f10.1)
  560 format(//' These are the current values;')
  565 format('   Help is ',a3,'.  Autodisplay is ',a3,'.')
  571 format(' Total power transmitted is ',f11.7,' milliwatts')
  574 format(' Maximum allowed gamma for a PD of type ',i1,' is',f12.3,
     *' rads')
  575 format(' The start and end times are',f9.2,' and',f16.2,' s')
  579 format(' The initial spacing between time steps is',f9.2,' s')
  587 format(' The output will be offset by',f12.3,' seconds')
      end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.3.2 Subroutine ENTERA

```
      subroutine entera(a)

************* Enter characters ***************
      character a*30
    2 format(a30)
    3 continue
      read(*,2,end=3)a
      return
      end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.3.3 Subroutine ENTERX

```
      subroutine enterx(x)

********** Enter Real Number ************
    2 format(f16.3)
    3 continue
      read(*,2,end=3)x
      return
      end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.3.4 Subroutine FREAD

```
      subroutine fread(fname)

************ Display Text File Onto Screen ************
      character fname*30, xline*90
      write (*,500)
    4 format(a90)
    2 format('+',a90)
      open(7,file=fname)
    5 read(7,4,end=3)xline
      write (*,2)xline
      go to 5
    3 continue
      close(7)
```

```
      return
  500 format(/)
      end
```

-=-

F.3.5 Subroutine FTYPES

```
      character name*30,choice*1

   30 write (*,500)
      write (*,502)
      write (*,503)
      write (*,504)
      write (*,600)
      call entera(choice)

      if(choice.eq.'e')call graphin

      if(choice.eq.'t')then
      write (*,710)
      call entera(name)
      do 40 i=1,3
      write (*,700)
      call enterx(x1)
      call interp(name,x1,y1,grate)
   40 write (*,705) y1
      end if

      if(choice.eq.'x')goto 60

      goto 30

  500 format(/'  Main Menu'/)
  502 format(' e=Enter New Fiber Data')
  503 format(' t=Test Old Fiber Data')
  504 format(' x=eXit Program'/)
  600 format(/' Please enter a choice from this menu')
  700 format(/' Enter an X value')
  705 format('  The corresponding Y value is 'f12.6)
  710 format(' Enter a filename')

   60 end
```

-=-

F.3.6 Subroutine GRAPHIN

```
      subroutine graphin

********** Subroutine to enter/modify graphic data ***********

      character filename*30
      dimension dgraph(2,200)
```

```
      n=0
      write (*,740)
      call enterx(grate)

      write (*,700)
      write (*,701)
      write (*,702)
      write (*,703)

**********  Enter the Data  **********

   20 n=n+1
      write (*,705)
      call enterx(x1)
      if ((n.gt.1).and.(x1.lt.dgraph(1,n-1))) goto 30
      write (*,710)
      call enterx(y1)
      dgraph(1,n)=x1
      dgraph(2,n)=y1
      goto 20

   30 write (*,715)
      write (*,716)
      write (*,717)
      write (*,718)
      call entera(filename)

**********  Save the data  ***********

      open (8,file=filename,status='new')

      write (8,730) n-1,grate
      do 40 i=1,n-1
   40 write (8,720) dgraph(1,i), dgraph(2,i)
      close(8)

  700 format(/' Enter each subsequent pair of points.  Each "X"')
  701 format('  value must be larger than the previously entered')
  702 format('  "X" value.  Enter an "X" value that is NOT larger')
  703 format('  to indicate that you are finished entering data.')

  705 format(/' Enter your next "X" value (Smaller number to quit)')
  710 format(/' Enter the corresponding "Y" value')

  715 format(/' Enter a file name in which to save this data.')
  716 format('  (Note: You can modify this file with any text-')
  717 format('  editing program to fix any points you may have')
  718 format('  entered incorrectly.)'/)

  720 format(2f20.10)
  730 format(i7,f14.4)
  740 format(/' At what dosage rate was this data measured?')
```

```
      return
      end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.3.7 Subroutine GRAPHR

```
      subroutine graphr(fname,x1in,x2in,output,ilog1,ilog2,ilog3)

****** Read values from digitized graphical data ******
*****************************************************************
****** Note:  This subroutine uses a linear approximation method
******  to determine the data when the data do not coincide
******  precisely with a fixed data point.
******
****** Note2:  In order for this subroutine to work properly,
******   The graphical data must be in a specified format
******   according to the following:
******
******  Line 1: (i4,2e12.6) No. of curves in plot, Minimum value
******   of first curve, Maximum value of last curve.
******
******  Line 2: (e12.6,i4,2e12.6) Curve value, No. of points in curve,
******   Minimum value of curve point, maximum value of curve point.
******
******  Lines 3-(3+No. of points in curve):  (2e12.6)  The 'x'
******   coordinate, the 'z' coordinate.
******
******  Next Lines:  Go back and repeat last two steps until
******   total number of curves is completed.
******
****** Note3:  The value of 'ilog' depends on which scales are
******  logarithmic:
******
******   ilog1 = 1 if 'x' is logarithmic, 0 if not.
******   ilog2 = 1 if 'y' is logarithmic, 0 if not.
******   ilog3 = 1 if 'z' is logarithmic, 0 if not.
*****************************************************************
      character fname*30
      dimension graph1(200),graph2(200),graph3(200),graph4(200)
      open (8,file=fname)

      x1input=x1in
      x2input=x2in
  104 if(ilog2.eq.0)go to 106

      x2input=alog10(x2input)

****** Read the number of curves and the min and max of
******  the curves.
```

```
  106 read (8,500) nx1,x1min,x1max

      if(ilog1.eq.0)go to 25
      if(x1input.le.0)x1input=x1min

      x1min=alog10(x1min)
      x1max=alog10(x1max)
      x1input=alog10(x1input)

   25 if(x1input.gt.x1max)go to 90

****** Read the variables for each curve.

      read (8,510) x1a,nx2a,x2mina,x2maxa

      if(ilog1.eq.0)go to 108
      x1a=alog10(x1a)

  108 if(ilog2.eq.0)go to 110
      x2mina=alog10(x2mina)
      x2maxa=alog10(x2maxa)

****** Check to see if this is the upper curve.

  110 if(x1a.gt.x1input)go to 30

****** If not, make this the lower curve and check the next one.

      x1=x1a
      nx2=nx2a
      x2min=x2mina
      x2max=x2maxa

****** Read in data for lower curve.

      do 20 i=1,nx2
      read (8,505) graph1(i),graph2(i)

      if(ilog2.eq.0)go to 120
      graph1(i)=alog10(graph1(i))

  120 if(ilog3.eq.0)go to 20
      graph2(i)=alog10(graph2(i))

   20 continue
      go to 25

****** Read in the data for the upper curve

   30 do 40 i=1,nx2a
      read (8,505) graph3(i),graph4(i)
```

```
      if(ilog2.eq.0)go to 140
      graph3(i)=alog10(graph3(i))

  140 if(ilog3.eq.0)go to 40
      graph4(i)=alog10(graph4(i))
   40 continue

****** Calculate interpolation factor between curves.

  150 x1p=(x1input-x1)/(x1a-x1)

****** Find interpolation position along lower curve.

      do 50 ix2=1,nx2
   50 if(x2input.lt.graph1(ix2))go to 60
      go to 90

****** Calculate lower curve interpolation factor.

   60 x2i=x2input
      g1=graph1(ix2)
      g1h=graph1(ix2-1)
      x2p=(x2i-g1h)/(g1-g1h)

******Find interpolation position along upper curve

      do 70 ix2a=1,nx2a
   70 if(x2input.lt.graph3(ix2a))go to 80
      go to 90

****** Calculate interpolation factor along upper curve.

   80 g3=graph3(ix2a)
      g3h=graph3(ix2a-1)
      x2ap=(x2i-g3h)/(g3-g3h)

****** Do interpolation along each curve, then between curves.

      g4=graph4(ix2a)
      g4h=graph4(ix2a-1)
      g2=graph2(ix2)
      g2h=graph2(ix2-1)

      out1=x2p*(g2-g2h)+g2h
      out2=x2ap*(g4-g4h)+g4h
      output=x1p*(out2-out1)+out1
      if(ilog1.eq.1)output=10**output
      go to 10
```

```
****** Out of range routine.

   90 output=-9999.
      go to 10
   10 close (8)
      return

  500 format(i4,2e12.6)
  505 format(2e12.6)
  510 format(e12.6,i4,2e12.6)
      end
```
-=-

F.3.8 Subroutine INTERP

```
      subroutine interp(filename,x,y,grate)

****** Subroutine to interpolate values from graphic values
******  entered with the subroutine GRAPHIN.  The value "X" is
******  inputted into this subroutine and the value "Y" is
******  returned to the calling program.

      character filename*30
      dimension dgraph(2,200)

      open(8,file=filename,err=200)

      read (8,730) n,grate
      do 40 i=1,n
      read (8,720) dgraph(1,i), dgraph(2,i)
      if ((i.eq.1).and.(dgraph(1,i).gt.x)) goto 50
      if (dgraph(1,i).lt.x) goto 40

      slope=(dgraph(2,i-1)-dgraph(2,i))/(dgraph(1,i-1)-dgraph(1,i))
      b=dgraph(2,i-1)-slope*dgraph(1,i-1)
      y=slope*x+b
      goto 9999

   40 continue

   50 write (*,710)
      goto 9999

  700 format(/' Oops, I can't find that file')
  710 format(/' Oops, that value is out of range'/)
  720 format(2f20.10)
  730 format(i7,f14.4)
  200 write (*,700)

 9999 close(8)
      return
      end
```

-=-

F.3.9 Subroutine LOGDATA

```
      subroutine logdata(filename)

********** Write the logarithm of the results to ASCII file ******
      integer fibertype,ftype

      common /b/nsections,trins(200),times(200),biter(200)
      common /d/tstart,tend,tstep,gstart,nsteps

      character filename*30
      open (8,file=filename,status='new')
      do 103 i=1,nsteps
      t=alog10(times(i))
      b=alog10(biter(i))
  103 write (8,520)t,b
      close(8)
      return
  520 format(f12.2,',',f18.14)
      end
```

-=-

F.3.10 Subroutine PRDATA

```
      subroutine prdata

********** Subroutine to save results **************
      integer fibertype

      common /b/nsections,trins(200),times(200),biter(200)
      common /d/tstart,tend,tstep,gstart,nsteps

      write (*,515)
      do 103 i=1,nsteps
  103 write (*,520) times(i),biter(i)
      return

  515 format('        Time,   Bit error ratio')
  520 format(f12.2,',    ',e12.6)
      end
```

-=-

F.4 Calculating Subroutines

-=-

F.4.1 Subroutine DARKEN

```
      subroutine darken

********** Calculate darkening factor ************

      integer photype
```

```
      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /d/tstart,tend,tstep,gstart,nsteps
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime

      darkn = 0
      do 10 i=1,nsections
      if(displ.eq.1.)write (*,500)i

****** Find gamma exposed.

      call gammaex(i)

****** Find the dosage rate.

      call dosera(i)

****** Find the loss factor.

      call loss(i)

****** Convert to meters

      xloss=xloss/1000.

      if(displ.eq.1.)write (*,530)gxposed,drate,xloss
   10 darkn = darkn + xlength(i) * xloss
      if(displ.eq.1.)write (*,525)darkn
      return

  530 format(' Gamma ex=',e10.4,' Dose rate=',e10.4,' Loss=',e10.4)
  500 format(' Calculating for section No.'i3)
  525 format('  Total darkening is ',f9.7,' dBm')
      end
```

-=-

F.4.2 Subroutine DOSERA

```
      subroutine dosera(i)

********** Calculate gamma dosage rate ************

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime

****** Note:  The actual dosage rate for time > 15700000 is an
******  equation unsuitable for computer calculations and is
******  therefore estimated by the equation for time < 15700000.
```

```
      if(xtime.lt.500.)then
      drate=0.
      go to 10
      end if

      drate=gammamax(i)*.771*xtime**(-1.2)/pf(i)
   10 return
      end
```

-=-

F.4.3 Subroutine GAMMAEX

```
      subroutine gammaex (i)

************ Calculate exposed gamma at time t **********

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime

****** Note:  The actual exposed gamma for times greater than
******  15700000 is an inexact equation not suitable for
******  translation to a computer and is therefore estimated.

      if (xtime.lt.500.)then
      gxposed=0.
      go to 10
      end if

      if(time.gt.15700000.)then
      gxposed=.986*gammamax(i)/pf(i)
      go to 10
      end if

      gxposed=3.85*gammamax(i)*(.289-1/(xtime**(.2)))/pf(i)
   10 return
      end
```

-=-

F.4.4 Subroutine LOSS

```
      subroutine loss(i)

********* Calculate the loss factor *************

      integer photype
      character fname*30,filename*30,buffer*40

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime
```

```
      if(xtime.le.500.)then
      xloss=0.
      return
      end if

      filename='fibtypes.txt'
      ntypes=0
      xloss=0.

      open(8,file=filename)
   40 read(8,505)buffer
      buff2=buffer(1:2)
      ntypes=ntypes+1
      if(ntypes.eq.nfibtypes(i))fname=buffer(5:)
      if(buff2.ne.'99')goto 40
      close(8)

      nincraments=10
      size=(xtime-500.)/nincraments

      inc=nincraments-1
      do 60 k=1,inc
      tau=size*k+500.

      ytime=alog10(xtime-tau)

      call interp(fname,ytime,xlossk,grate)

   60 xloss=xloss+xlossk*(.771*gammamax(i)*tau**(-1.2)*size)/pf(i)/grate

   20 return

  505 format(a40)

      end
```

-=-

F.4.5 Subroutine READBER

```
      subroutine readber(errrate)

****** Subroutine to read the BER from either a graph or
******  equations.

      integer photype
      character fname*30

      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime

      if(photype.eq.2)then
      ymax=-19.
      ymin=-42.5
```

```
      fname='berlldef.gph'
      end if

      if(photype.eq.1)then
      ymax=-21.
      ymin=-52.5
      fname='ber-lla.gph'
      end if

      if(recsig.lt.ymin)then
      errrate=.5
      return
      end if

      if(recsig.gt.ymax)then
      errrate=.1E-11
      return
      end if

      call graphr(fname,drate,recsig,errrate,1,0,1)
      return
      end
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

F.4.6 Subroutine SIGPOW

```
      subroutine sigpow

****** This subroutine calculates the Receive Signal Power
******  at the photodiode.  It will include losses due to
******  Gamma Radiation, Intrinsicness, and Scintillation.

      integer photype

      common /b/nsections,trins(200),times(200),biter(200)
      common /c/displ,help,powtransm,photype,gpmax,worstcase
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime

      recsig=powtransm-xlossi-darkn

****** Check to see if photodiode has been destroyed

      call gammaex (nsections+1)
      if(gxposed.gt.gpmax)then
      recsig=-9999.
      go to 10
      end if

****** If not, find the gamma rate at the photodiode

   10 call dosera (nsections+1)
      return
      end
```

-=-

F.4.7 Subroutine XTRINS

```
      subroutine xtrins

****** This subroutine will calculate the intrinsic loss
******  of the optical fiber sublink.

      integer fibertype,photype

      common /a/xlength(200),gammamax(200),pf(200),nfibtype(200)
      common /b/nsections,trins(200),times(200),biter(200)
      common /f/gxposed,darkn,xloss,drate,xlossi,recsig,xtime

      xlossi=0.
      do 100 i=1, nsections
  100 xlossi=xlossi+(xlength(i)*trins(i)/1000.)

      return
      end
```

-=-

F.5 Informational Text Files

-=-

F.5.1 Informational Text File FIBTYPES.TXT

```
 1. att13sm.fty
 2. att825mm.fty
 3. att13mm.fty
 4. nrl-814.fty
 5. spec-std.fty
 6. att652a.fty
 7. corn1524.fty
 8. nrl-815.fty
 9. spec-rhd.fty
10.
99. end
```

-=-

F.5.2 Informational Text File INTRO.TXT

FIBer optic Radiation Antenuation Model (FIBRAM)

This program is designed to analyze the Bit Error Ratio for an optical fiber sublink when exposed to gamma radiation fallout over a period of time.

As you run this program you will be able to change the following parameters:

>> The Name for the System Variables
>> The Help Function (ON/OFF)

>> The Autodisplay Function (ON/OFF)
>> Photodiode Type
- Maximum Gamma Radiation Exposure That the
- Photodiode Can Survive

>> Timing Parameters
- Starting Time
- Ending Time
- Offset Time
- Incremental Time Step Factor

>> Number of Model Sections
- Length of Each Section
- Protection Factor of Each Section
- Type of Optical Fiber
- Intrinsic Loss Factor of the Fiber Type
- Maximum Gamma for Each Section

Help files will be available for most of the above topics if you have the HELP feature toggled to the ON position. This can be done directly through the main menu by typing 'h' <RETURN>, or indirectly when modifying the system parameters with the 'm' <RETURN> command.

After you have entered new system parameter values for your optical fiber system, you may save these values in a file and call them up the next time you run the program. This is done by calling SAVE and LOAD, respectively, from the main menu.

As with most Fortran programs, FIBRAM is very fussy about the format of variables entered from the keyboard by the user. If the user does not type in numerals in the format FIBRAM is expecting, it will often put the decimal point in the wrong place. The user can override the formatting simply by entering REAL numbers (i.e, always use decimal points).

Many of the questions the user will be prompted to answer will have default answers. If a value does have a default value, FIBRAM will display this value before it prompts the user to enter a new one. If the user wishes to accept the default value, the <RETURN> key should be pressed.

-=-

F.5.3 Informational Text File MAINMENU.TXT

d = Display all of the current system parameters
m = Modify the overall parameters
t = Modify the timing parameters
h = Toggle the help function (ON/OFF)
n = Change the overall name of these modeling parameters
s = Save all parameters to disk
l = Load previous parameters from disk

r = Run the simulator
x = Exit to DOS

p = Print the output results to the screen

g = Print the output results to an ASCII file (log/log)

> = Automatically run a range of values

-=-

F.5.4 Informational Text File NSECTIONS.TXT

You will be asked how many different sections your optical fiber sublink has. FIBRAM defines a 'different section' as a part or parts of the fiber cable that have a protection factor, fiber type, and maximum gamma exposure significantly different than all other parts of the fiber. If the user is unsure if protection factors are significantly close, separate sublink sections should be specified, although the model will take longer to run.

-=-

F.5.5 Informational Text File PHOTYPES.TXT

1. 11-a
2. 11-d, e, and f
3.
4.
5.
6.
7.
8.
9.
10.

-=-

F.5.6 Informational Text File RANGE.TXT

This part of the program allows the user to run FIBRAM several different times, while substituting different values for one of the variables. For example, if the user wished to analyze the same model at gamma exposure levels of 500, 1000, and 5000 rads, this subroutine would be used.

The user will be asked to enter the number of values the selected variable is to assume. In the above case, this number would be 3 (500, 1000, and 5000). The user is then prompted to indicate which variable is to 'ranged' (gamma exposure level).

The user is then asked to supply FIBRAM with the 3 values (in this case) to be analyzed. Following each input, the user is prompted to enter a file name to save any output results to. These file names should be unique so that no data is lost by saving it to files that already exist.

Once the user has typed in all this information, the user is given the opportunity to save these values and file names to disk in case this series of values is to be used again in the future.

If the user has already saved such a file and wishes to use this former data, '0.' <RETURN> should be typed when asked for the number of values the selected variable is to assume. This will cause FIBRAM to ask for the file name that these previously saved variables are stored under.

-=-

F.5.7 Informational Text File TIME.TXT

You will now be asked to enter the timing information for your model. There are four pieces of information you must tell FIBRAM. They are:

- Starting Time
- Ending Time
- Offset Time
- Incremental Time Step Factor

The starting and ending times are self-explanatory. When entering the starting time, it should be noted that a <RETURN> with no value typed in will result in a value of 0 being assumed.

The offset time indicates the time that the user wants the model to consider to be time zero. For example, if the offset and starting times are 400 and 410 seconds respectively, the model will be modeled at 410 seconds after radiation exposure, but the outputted time will only be 10 (410 minus 400) seconds.

The incremental time step factor indicates the logarithm of the size of time step the user wants to use. If a step factor of 2 is specified, the model steps will begin with a size of 10^2, or 100. Every ten time steps this factor is automatically increased by 1.

-=-

F.5.8 Informational Text File WORST.TXT

You are to now enter the magnitude of the transmitted power signal being entered into the front end of the fiber optic sub-link you are now modeling. This value should be in milliwatts. FIBRAM will convert this value into decibel milliwatts for its internal use.

The user should choose a value such that when FIBRAM converts it to decibel milliwatts and then subtracts the intrinsic losses, the resulting decibel figure is within the optimum operating range for the chosen photodiode. If this is not done properly, the photodiode may not be operating in its linear region.

-=-

APPENDIX G. OBTAINING COPIES OF FIBRAM

G.1 Obtaining Copies of the Original FIBRAM Program

The user should contact the National Technical Information Service (NTIS) for information on ordering additional copies of FIBRAM. FIBRAM can be ordered in two different formats.

The first format is the report itself. This format does not contain a program disk. The user may utilize the programs by entering the subroutines manually from the report listings and then compiling them.

The second format contains the report and a floppy disk containing the program files. This disk is formatted for an IBM PC/AT/XT or compatible computer and contains several additional data files and sample runs whose listings are not listed in the body of the report itself.

When ordering either of these formats, ask for the NTIA report titled, "A Program Description of FIBRAM: A Radiation Attenuation Model for Optical Fibers." Be sure to specify whether you want a program disk included or not.

G.2 Obtaining Information on Updated Versions of FIBRAM

From time to time there may be updated and/or improved versions of FIBRAM available through the NTIS update service for a nominal fee. The NTIS will provide notification of program updates to those users who have purchased original FIBRAM reports through NTIS.