AD-A274 586

WL-TR-93-4098

OPTICAL CHARACTERIZATION AND GUIDED WAVE LOSS MEASUREMENTS FOR END-CAPPED 6F-POLYBENZOXAZOLE

WRIGHT LABORATORY

ANGELA L. MCPHERSON
JEFFERY W. BAUR

SEPTEMBER 1993

FINAL REPORT FOR 11/01/90-06/01/93

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

DTIC
ELECTE
JAN 10 1994
S B D

MATERIALS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AFB OH 45433-7734

94-00875

94 1 7 315

NOTICE

WHEN GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA ARE USED FOR ANY PURPOSE OTHER THAN IN CONNECTION WITH A DEFINITELY GOVERNMENT-RELATED PROCUREMENT, THE UNITED STATES GOVERNMENT INCURS NO RESPONSIBILITY OR ANY OBLIGATION WHATSOEVER. THE FACT THAT THE GOVERNMENT MAY HAVE FORMULATED OR IN ANY WAY SUPPLIED THE SAID DRAWINGS, SPECIFICATIONS, OR OTHER DATA, IS NOT TO BE REGARDED BY IMPLICATION, OR OTHERWISE IN ANY MANNER CONSTRUED, AS LICENSING THE HOLDER, OR ANY OTHER PERSON OR CORPORATION; OR AS CONVEYING ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY IN ANY WAY BE RELATED THERETO.

THIS REPORT IS RELEASABLE TO THE NATIONAL TECHNICAL INFORMATION SERVICE (NTIS). AT NTIS, IT WILL BE AVAILABLE TO THE GENERAL PUBLIC, INCLUDING FOREIGN NATIONS.

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

Angela L. McPherson

Angela L. McPherson, Research Physicist
Electronic & Optical Mat'ls Branch
Electromagnetic Mat'ls & Surv. Div.

Robert L. Denison

ROBERT L. DENISON, Chief
Electronic & Optical Mat'ls Branch
Electromagnetic Mat'ls & Surv. Div.

William R Woody

WILLIAM R. WOODY, Chief
Electromagnetic Mat'ls & Surv. Div.
Materials Directorate

IF YOUR ADDRESS HAS CHANGED, IF YOU WISH TO BE REMOVED FROM OUR MAILING LIST, OR IF THE ADDRESSEE IS NO LONGER EMPLOYED BY YOUR ORGANIZATION PLEASE NOTIFY WL/MLPO, BLDG 651, 3005 P ST STE 6, WPAFB OH 45433-7707, TO HELP MAINTAIN A CURRENT MAILING LIST.

COPIES OF THIS REPORT SHOULD NOT BE RETURNED UNLESS RETURN IS REQUIRED BY SECURITY CONSIDERATIONS, CONTRACTUAL OBLIGATIONS, OR NOTICE ON A SPECIFIC DOCUMENT.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	SEP 1993	FINAL 11/01/90--06/01/93

4. TITLE AND SUBTITLE
OPTICAL CHARACTERIZATION AND GUIDED WAVE LOSS MEASUREMENTS FOR END-CAPPED 6F-POLYBENZOXAZOLE

5. FUNDING NUMBERS
PE 61102
PR 2305
TA FO
WU FJ

6. AUTHOR(S)
ANGELA L. MCPHERSON
JEFFERY W. BAUR

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
MATERIALS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT PATTERSON AFB OH 45433-7734

8. PERFORMING ORGANIZATION REPORT NUMBER
WL-TR-93-4098

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
MATERIALS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT PATTERSON AFB OH 45433-7734

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
WL-TR-93-4098

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

A Mathematica software package to model the behavior of light inside an optical waveguide is discussed. Examples of package features are presented and discussed. Possible new waveguide materials are investigated with this package as a baseline for material evaluation. The use of the polymer 6F-PBO-1E as a possible candidate for high temperature optical waveguide material based in its high Tg of 300C and its optical clarity is discussed. Theoretical studies are performed with the Mathematica package. Slab waveguide samples are fabricated by spin-coating on oxidiz05 silicon wafers and characteriz0d through measurements of the index of refraction and waveguide loss. Results show that the material exhibits its lowest loss of 4.5 dB/cm at a wavelength of 700nm. We conclude, based on experimental results, that this material, 6F-PBO-1E, while it may be useful for some optical applications, does not meet the Air Force specification of 1 dB/cm or less performance guideline for optical waveguide materials and therefore is not appropriate for Air Force applications in its present form.

14. SUBJECT TERMS
optical waveguide loss, planar waveguide, absorption, index of refraction

15. NUMBER OF PAGES
96

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

TABLE OF CONTENTS

TABLE OF CONTENTS (*Continued*)

TABLE OF CONTENTS (*Concluded*)

DTIC QUALITY INSPECTED 5

Accession For

Distribution/

Availability Codes

Dist

A-1

LIST OF FIGURES

LIST OF FIGURES (*Continued*)

LIST OF FIGURES *(Concluded)*

List of Tables

ACKNOWLEDGEMENTS

The authors wish to acknowledge S.J. Bai of the Materials Directorate for his help in verification of the index of refraction measurements and Dr. Uma Ramabadran of the University of Dayton Research Institute for her support and consultation. Mr. Ronald O. Neaffer is acknowledged for his assistance in the ellipsometric work and Mr. E. Thomas Kensky for his assistance with the waveguide stage and programming efforts. We also wish to acknowledge Daychem Inc, Centerville OH, for supplying the polymer material on which this study is based. Finally, we wish to thank Dr. F. Ken Hopkins and Dr. David Zelmon for their help and suggestions on the modeling section of this report.

1.0 INTRODUCTION

Optical waveguides are structures that carry optical signals by confining light in one or two dimensions and have been studied for some time[1]. These structures have been studied extensively during the past two decades because they are the central component of most integrated optical devices including couplers, modulators, and interferometers. Given the wide variety of materials and processes that may be used in waveguide fabrication, it is imperative that designers be able to predict the performance of this critical component. We have chosen to develop calculations for the four layer planar waveguide structure since it is generally adequate to model the behavior of light energy in various device-specific waveguide configurations such as channel or rib-type.

We have developed a program that may be used to calculate field distribution, propagation loss, and critical values for the four-layer planar optical waveguide. This problem has previously been solved using a variety of methods[2,3]. We discuss a simpler approach to solving the complex eigenvalue equation using the software package *Mathematica*.

2.0 THEORETICAL BACKGROUND

2.1 THE WAVEGUIDE PARAMETERS

A planar waveguide is a multilayer structure such as that shown in Figure 1. At an interface between two media of different indices of refraction, electromagnetic (EM) waves are split into a reflected and refracted beam. The reflected beam obeys the law of reflection, and the refracted beam obeys Snell's Law :

$$n_1 \sin \theta_1 = n_2 \sin \theta_2 \qquad (1)$$

where n_1, n_2 are the indices of refraction for regions 1 and 2 and θ_1, θ_2 are the angles of incidence and refraction, respectively, and are defined with respect to the normal to the

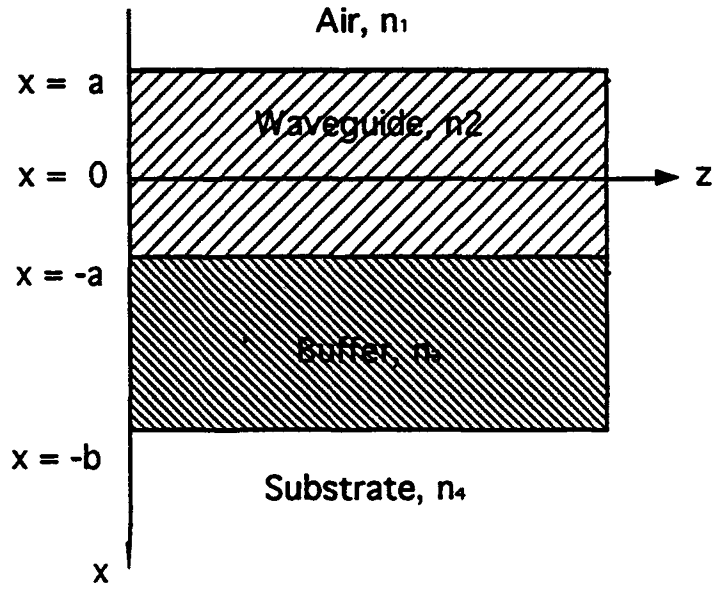


Figure 1 Diagram of the Four Layer Planar Waveguide.

interface between regions 1 and 2. In the case $n_1<n_2$ there is always a real angle, θ_2, that satisfies this equation and, therefore, there is transmittance from region 1 to region 2. However, if $n_1>n_2$ and $\theta_2>\theta_1$, then there is a maximum angle of incidence for which there is a real solution for θ_2. This occurs when θ_2 reaches its maximum value of 90°. The corresponding value of the incident beam angle is known as the critical angle and is given by:

$$\frac{n_2}{n_1}\sin\theta_c = 1 \tag{2}$$

Total internal reflection occurs at angles of incidence larger than the critical angle and, therefore, there is no refracted beam (θ_2=90°). If a material with refractive index n_2 is placed between two other materials with indices n_1 and n_3 so that $n_1,n_3<n_2$, the light can be confined to the center layer by total internal reflection. This is the situation in the planar waveguide.

The structure under consideration (see Figure 1) consists of four layers, each having a different index of refraction. Light is coupled into the waveguide from layer 1, usually air. Layer 2, called the guiding layer, is the high-index layer where the energy is to be confined. Layer 3 is the buffer layer and is used to prevent any possible coupling of the guided wave from layer 2 to layer 4. The entire arrangement is fabricated on a substrate, layer 4, which can have a very high index.

2.2 THE EIGENVALUE EQUATION

The behavior of light in a structure such as Figure 1 is calculated by solving the wave equation in each of the four layers and applying appropriate boundary conditions. In each layer of the guide, the behavior of the EM wave E(x,y,z,t) is described by the wave equation[4] :

$$\nabla^2\vec{E}(\vec{r}) + k_o^2 n^2 \vec{E}(\vec{r}) = 0 \tag{3}$$

where $k_o=2\pi/\lambda$, λ is the wavelength of the propagating light, and n is the refractive index of the material. We assume the waves are time harmonic, the dielectric is lossless and there are no free charges.

The waveguide problem may be solved by assuming that the optical fields propagate as inhomogeneous plane waves of the form:

$$\vec{E}_i(\vec{r},t)=\vec{E}_i(x,y)e^{+i(\omega t - \beta z)} \tag{4}$$

Assuming no y-direction variation, substituting this expression into the wave equation yields:

$$\left(\frac{\partial^2}{\partial x^2}\right)\vec{E}_i(x) + \left[k_o^2 n_i^2 - \beta^2\right]\vec{E}_i(x) = 0 \tag{5}$$

where n_i is the index of the i-th layer

We define:

$$\begin{aligned} \sqrt{\beta^2 - k_o^2 n_1^2} &\equiv q \\ \sqrt{k_o^2 n_2^2 - \beta^2} &\equiv h \\ \sqrt{\beta^2 - k_o^2 n_3^2} &\equiv p \\ \sqrt{k_o^2 n_4^2 - \beta^2} &\equiv s \end{aligned} \tag{6}$$

to be the propagation parameters of layers 1, 2, 3, and 4, respectively.

(Note: $\beta = k_o n_e$ where n_e is the unknown effective refractive index of the waveguide.)

The wave equation will have two sets of solutions corresponding to the two orthogonal polarizations: transverse electric (TE) and transverse magnetic (TM). Calculations for both

TE and TM modal solutions proceed in the same manner, so our discussion will be confined to the TE modes of propagation only.

For the four-layer guide, we assume the relation $n_2>n_3>n_1$. Remembering this and going back to equation (5), we can write down the solutions to the equation in each layer of the guide. The desired condition consists of energy confinement/propagation in the guiding layer, evanescent fields in the first and third layers and possible propagation in the fourth layer. We can solve the ordinary differential equations (ODE) and, realizing that all fields are finite and no fields originate in the substrate, the solutions are:

$$
\begin{aligned}
\vec{E_1}(\vec{r},t) &= A_1(x,y)\, e^{-q(x-a)} \\
\vec{E_2}(\vec{r},t) &= A_2(x,y)\sin(hx) + A_2'(x,y)\cos(hx) \\
\vec{E_3}(\vec{r},t) &= A_3(x,y)\, e^{p(x+a)} + A_3'(x,y)\, e^{-p(x+a)} \\
\vec{E_4}(\vec{r},t) &= A_4(x,y)\, e^{-i\,s(x+a+b)}
\end{aligned}
\tag{7}
$$

Now we must impose boundary conditions. At each interface, in the absence of any sources, both the tangential electric (**E**) and tangential magnetic (**H**) fields must be continuous. For source-free media, the **E** and **H** fields are related via Maxwell's equations:

$$
\begin{aligned}
\nabla\times\vec{E} &= -i\omega\mu\vec{H} \\
\nabla\times\vec{H} &= i\omega\epsilon\vec{E}
\end{aligned}
\tag{8}
$$

The corresponding H fields are:

$$
\begin{aligned}
\vec{H_1}(\vec{r},t) &= \frac{i}{\omega\mu}\left[-A_1\, q\, e^{-q(x-a)}\right] \\
\vec{H_2}(\vec{r},t) &= \frac{i}{\omega\mu}\left[A_2\, h\cos(hx) - A_2'\, h\sin(hx)\right] \\
\vec{H_3}(\vec{r},t) &= \frac{i}{\omega\mu}\left[A_3\, p\, e^{p(x+a)} - A_3'\, p\, e^{-p(x+a)}\right] \\
\vec{H_4}(\vec{r},t) &= \frac{1}{\omega\mu}\left[A_4\, s\, e^{-is(x+a+b)}\right]
\end{aligned}
\tag{9}
$$

Upon application of the boundary conditions, the following set of equations result (where a, b are defined in Figure 1).

$$
\begin{aligned}
&A_1 = A_2 \sin(ha) + A_2' \cos(ha)\\
&A_2 \sin(-ha) + A_2' \cos(-ha) = A_3 + A_3'\\
&A_3\, e^{-pb} + A_3'\, e^{pb} = A_4\\
&-A_1\, q = A_2\, h \cos(ha) + A_2'\, h \sin(ha)\\
&A_2\, h \cos(-ha) - A_2'\, h \sin(-ha) = A_3\, p - A_3'\, p\\
&i\left[A_3\, p\, e^{-pb} - A_3'\, p\, e^{pb}\right] = s\, A_4
\end{aligned}
\tag{10}
$$

This system of equations can be simplified to one equation with one unknown, β; this β may be complex, representing energy lost to the system via coupling into the substrate. The eigenvalue equation for β is:

$$
(p + is)\left[\left(h^2 - pq\right)\mathrm{Tan}(2ah) - h(p + q)\right] + e^{-2pt}(p - is)\left[\left(h^2 + pq\right)\mathrm{Tan}(2ah) + h(p - q)\right] = 0 \tag{11}
$$

where $t = (b-a)$,

$n_4 = n_{4real} + i\, n_{4imag}$,

$\beta = \beta_{real} + i\, \beta_{imag}$

2.3. LIMITS ON n_e

The limits on the complex propagation constant, β, must be examined before any modeling is presented. These limits form a baseline and are used extensively in arriving at the solution to the eigenvalue equation.

It has been shown that to enable guided mode solutions, the index of refraction of the guiding layer must be higher than that of the surrounding layers. The assumption is also made that $n_2>n_3>n_1$, which is always true if the first layer is air. Next, we recall the wave equation in the general form inside each layer as shown in equation (5). It has the clause:

$$\left[k_o^2 n_i^2 - \beta^2\right] \tag{12}$$

which will define the behavior of the equation. In the first case, take $\beta > k_o n_2$. Simplifying the equation above results in the following generalized ODE:

$$\frac{1}{E}\left(\frac{\partial^2 E}{\partial x^2}\right) > 0 \tag{13}$$

This can be rewritten as:

$$\left(\frac{\partial^2 E}{\partial x^2}\right) = \alpha^2 E \tag{14}$$

where α is some constant. The general solution to this ODE is of the form:

$$E = A\, e^{\alpha x} + B\, e^{-\alpha x} \tag{15}$$

This solution then corresponds to all layers in the structure based on the assumption that $n_2>n_3>n_1$. Reviewing the definition of propagation as outlined in Section 2.1, this solution does not correspond to a guided mode.

The second case investigated is $k_o n_3 < \beta < k_o n_2$. In this case, the ODE becomes:

$$\left(\frac{\partial^2 E}{\partial x^2}\right) = -\alpha^2 E \tag{16}$$

The two regions, $\beta < k_o n_2$ and $k_o n_1, k_o n_3 < \beta$ must now be examined. β now has two solutions depending on which region is examined. In layer 2,

$$E = A \sin(\alpha x) + B \cos(\alpha x) \tag{17}$$

while in layers 1 and 3 the same solution as in equation (14) exists. This solution corresponds to a guided mode, i.e., there is propagation in the guiding layer (layer 2), and evanescent fields in the surrounding layers (layers 1 and 3).

In the case where $k_o n_1 < \beta < k_o n_3$ there is exponential behavior in layer 1 and sinusoidal behavior in regions 2 and 3. Also, if $0 < \beta < k_o n_1$, there is sinusoidal behavior in all regions. Neither of these corresponds to a guided mode solution. Therefore, the limits on β for guided mode solutions is $k_o n_3 < \beta < k_o n_2$.

3.0. MODELING

The eigenvalue equation may be solved using the mathematics software package *Mathematica* to calculate the field distributions, propagation loss, and critical values for the four-layer planar waveguide. The basic algorithm calculates all possible roots of the eigenvalue equation, defines all confined modes and plots the waveform for each mode. The program requires only specification of the refractive indices of the four layers, the thickness of the guiding and buffer layers and the wavelength of the input light.

This algorithm can also determine other waveguide properties including waveguide attenuation (energy loss into the fourth layer) and the cutoff values (wavelength and guiding layer thickness below which no modes are confined) which are important considerations in fabrication.

Mathematica's built-in root-finding procedure, **FindRoot**, is used to solve the equation. The solution must lie between the values k n3 and k n2 (k is $2\pi/\lambda$, λ being the wavelength of the incident light) for the equation to hold true. We can use this stipulation to define an initial value for **FindRoot**, however, the root converges very slowly. A quicker method is to solve the problem by considering an infinitely thick buffer layer. This is equivalent to considering a three-layer asymmetric waveguide[1]. The four-layer eigenvalue equation reduces to (s corresponds to the fourth layer and is set to zero):

$$\mathrm{Tan}(2ah) = \frac{h(p+q)}{\left(h^2 - pq\right)} \qquad (18)$$

Since this equation has roots that converge relatively quickly, the value of β which is the solution to this eigenvalue equation is used as the starting value for the **FindRoot** routine in calculating the eigenvalues for the four-layer equation.

For a given set of parameters, the four-layer eigenvalue equation may have multiple roots corresponding to the different modes supported by the waveguide. It is easy to see from the three-layer eigenvalue equation that a simple **Do** loop changing the initial values for **FindRoot** will not solve for all roots. The equation blows up at certain initial values. For this reason, the combination of *Mathematica*'s **FindRoot, Check,** and **Continue** functions are indispensable. **Check** is a boolean variable which scans the output stream for error messages. If errors are not encountered, **Check** returns the value of its first argument. The value of the second argument, in this case **Continue**, is returned if errors are found. Combining these functions (along with checks for duplicate roots and assuming FourLayerEquation is the eigenvalue equation) gives the root-finding portion of the code:

```
Check[rule2 = FindRoot[FourLayerEquation == 0,
      {B,B/.rule1},AccuracyGoal -> 12],Continue[]];
```

Check and **Continue** allow for the evaluation at these points without crashing the code. The starting point 'B/.rule1' is the solution to the much simpler three-layer problem.

The source code shown above will completely solve the eigenvalue equation for all confined modes of the user-specified waveguide structure. Once the propagation constant, β, is known, it is used to calculate the amplitudes A_i in each layer of the waveguide.

The basic algorithm scans the range of possible roots of the eigenvalue equation. It then produces all roots, thereby, defining all confined modes, and plots the waveform for each mode. The program requires only specification of the refractive indices of the four layers, the thickness of the guiding and buffer layers, and the wavelength of the input light.

The relative simplicity of this algorithm allows for the determination of other waveguide properties. Also offered in this package are routines to calculate the waveguide attenuation (energy loss into the fourth layer and associated with the specified structure) and the cutoff values (wavelength and guiding layer thickness below which no modes are confined) which are important considerations in the fabrication of waveguiding structures.

3.1. PACKAGE FEATURES

The following subsections describe and give examples of the procedures featured in the *Mathematica* package PlanarWaveguides.m. For further details on the code, see Section 10.0.

3.1.1. PROCEDURE "CheckInput"

Note: This is an internal procedure.

Several checks are made on the input values to see if there are any errors in the input which would preclude the existence of any guided modes. These checks include cutoff thickness, cutoff wavelength, $n_1<Re[n_2]>n_3$, and other erroneous input values. If any of these errors are encountered, execution of the code will cease. A default of one iteration through the root-finding procedure is assumed for the case where the increment entered is zero.

The cutoff thickness and cutoff wavelength are calculated from the three-layer eigenvalue equation as the value of the effective index of refraction (proportional to the propagation constant) approaches the index of the third (cladding) layer. If the waveguide thickness is less than the cutoff thickness or the cutoff wavelength is shorter than the input wavelength, an appropriate message is printed and program execution stops.

3.1.2. PROCEDURE "PlotThis"

Note: This is an internal procedure.

As the number of guided modes may differ as the parameters are varied, the values of attenuation are separated by mode number. "PlotThis" accepts the list of roots found and marks the associated position in "modeFlags" as to the mode number. "ModeFlags" is then used to plot the attenuation versus the mode number. The Re[newList[[i,1]]] is used for the cases that involve the complex n_4 parameter.

A variation of n_4 (or n_2) may result in values such as 3.85 - 0.01 I and 3.85 - 0.02 I. The attenuation values are considered different for the number of modes, but the values will be plotted above/below each other as only the Re[n_4] (Re[n_2]) is used in plotting.

The parameters for this routine are the list of waveguide parameter values with their associated attenuation values, the "plotLabel," and "biggestListCount" which holds the maximum number of guided modes for any given value of the varied parameter.

3.1.3. PROCEDURE "FindRoots"

Note: This is an internal procedure.

"FindRoots" is the core of the code. The roots of the eigenvalue equation are actually found in this procedure.

Since the interest is only in the guided mode solutions (roots of the eigenvalue equation between the index of layer two and three times a constant k), the testing for roots scans the entire range of possible roots, with a step-size (increment2) dependent on the size of the range.

The parameters are the increment size for the test range (increment2), the present value of the varied parameter (variable), "threeLayerEquation," "fourLayerEquation," k, n_2, n_3, and flag which is simply used to test if the guiding layer thickness is the parameter being varied. This must be tested for because the equations and theory depend on the value "halfGuide" which is one-half the guiding layer thickness. To avoid confusing the user, the value of the total guiding layer thickness is printed out instead of "halfGuide."

To obtain starting values for the four-layer algorithm, the solutions for the three-layer waveguide are calculated. The simplicity of the three-layer eigenvalue equation allows less stringent requirements on the starting values for its solutions. These solutions can then in turn be used as initial conditions for the four-layer equation.

The "AccuracyGoal" and "MaxIterations" were chosen to allow quick turn-around-time, yet still achieve accurate solutions. To aid in shortening the calculation time, if the root found is greater than the test value, the test value will be set to the new root. "FindRoots" returns a list of β values which is added (Union) to the list of all solutions found for the varied parameter.

"Bold" is used to hold the value of the previous root for testing against the present -- if N[Bold,3] != N[B/.rule2,3] the root is considered different and added to the list. "Rule1" and "rule2" hold the results of "FindRoot" for the three-and four-layer eigenvalue equation, respectively. Test is the test value for the range of possible roots. List is the intermediate list of output values.

3.1.4. PROCEDURE "PropagationConstant"

"PropagationConstant" calculates and plots all confined modes for the input parameters. For each propagation constant found, the waveform is plotted showing the mode numbers explicitly. The field amplitude coefficients "ampTwoA," "ampTwoB," "ampThreeA," "ampThreeB," and "ampFour," are calculated by *Mathematica* using the eigenvalue equation and the "Solve" function; "ampOne" is set to 1. The function "Equation" is the four-layer equation, k is the propagation constant, increment is the step through the range of possible roots and list is the list of propagation constant values. The parameters q, h, p, and s are the same as those defined in equation (6).

PropagationConstant[1.,1.691,1.46,3.85 - 0.02 I,1,1.43,1]

10.0032 - 0.0000236124 I
10.47 - 1.13667 10^{-6} I

Begin plotting routine.....

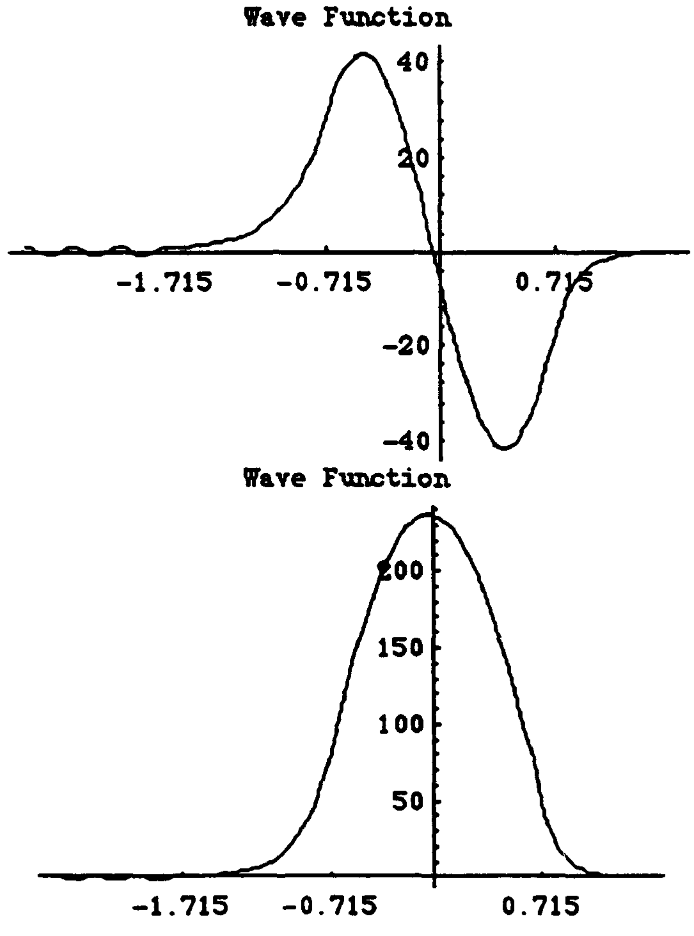


Figure 2 Example of "PropagationConstant" Procedure

3.1.5. Attenuation Calculation Procedures

"N1Attenuation," "N2Attenuation," "N3Attenuation," "N4Attenuation," "GuideAttenuation," "CladdingAttenuation," and "WavelengthAttenuation" all have the same basic outline. Each one calculates the loss due to attenuation versus a varied waveguide parameter. For a description of the procedure see "N1Attenuation."

3.1.5.1 PROCEDURE "N1Attenuation"

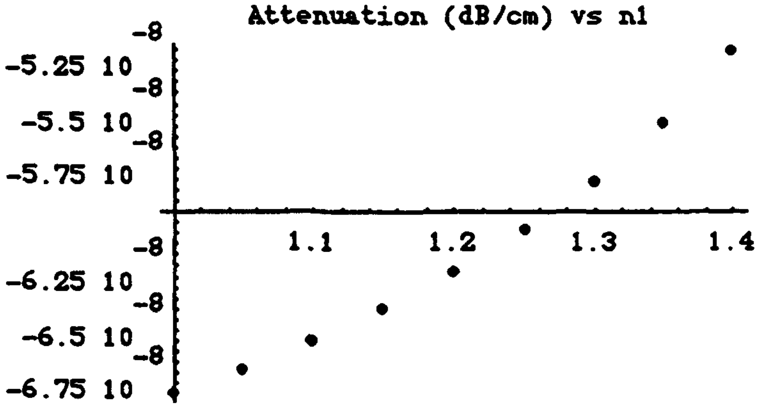
"N1Attenuation" accepts user input in the format:

"N1Attenuation"[{start,end,increment},n_2,n_3,n_4,cladding,guide,wavelength]

where {start,end,increment} replaces n_1 and, therefore, defines the range of values for n_1. All equations that involve n_1 are rewritten replacing n_1 with 'variable' where 'variable' ranges from start to end in steps of increment. The input parameters are checked, the roots searched for and the plot is made. All parameters and local variables have been defined previously. NumberQ[N[*variablename*]] is used to check the input before any processing has taken place to make sure the variables contain numeric values. Since a user enters these values, it is possible that the values are not numerical as they should be and any computation will crash.

N1Attenuation[{1.,1.4,0.05},1.67,1.46,3.85 - 0.02 I,0.5,0.5,0.6328]

{1., $-6.84822\ 10^{-8}$ }
{1.05, $-6.73274\ 10^{-8}$ }
{1.1, $-6.60235\ 10^{-8}$ }
{1.15, $-6.45425\ 10^{-8}$ }
{1.2, $-6.28474\ 10^{-8}$ }
{1.25, $-6.08884\ 10^{-8}$ }
{1.3, $-5.85966\ 10^{-8}$ }
{1.35, $-2.94539\ 10^{-6}$ }
{1.35, $-5.58731\ 10^{-8}$ }
{1.4, $-2.39124\ 10^{-6}$ }
{1.4, $-5.25691\ 10^{-8}$ }



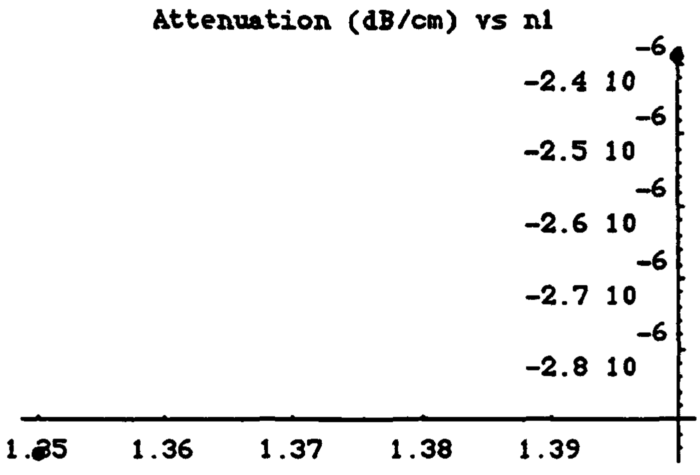


Figure 3 Example of "N1Attenuation" Procedure

3.1.5.2 PROCEDURE "N2Attenuation"

"N2Attenuation" is like the previous attenuation functions, except there are special cases for n_2 since it may be a complex number. The variables "imaginaryVar" and "realVar" are used for varying such parts and variable is the combination of the "Re" and "Im" part. A variation of n_2 may result in values such as 1.52 - 0.01 I and 1.55 - 0.02 I. The attenuation values are considered different for the number of modes, but the values will be plotted above/below each other as only the Re[n_2] is used in plotting.

N2Attenuation[1.,{1.6,1.68,0.02},1.46,3.85 - 0.02 I,0.5,0.5,0.6328]

{1.6, -2.91513 10^{-7} }
{1.62, -1.90045 10^{-7} }
{1.64, -1.25314 10^{-7} }
{1.66, -8.35437 10^{-8} }
{1.68, -5.6277 10^{-8} }

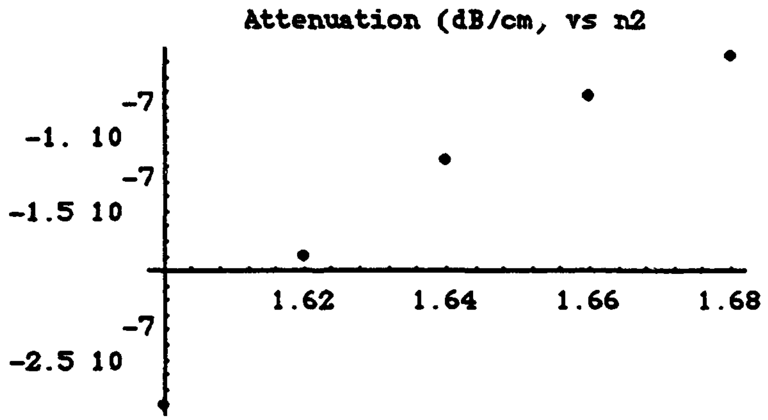
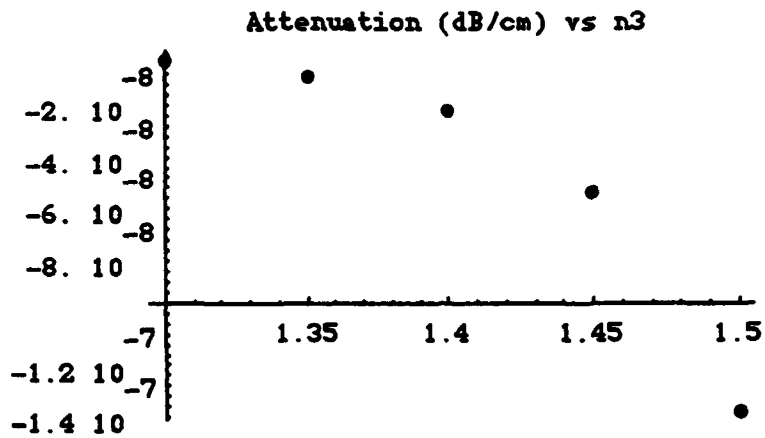


Figure 4 Example of "N2Attenuation" Procedure

3.1.5.3. PROCEDURE "N3Attenuation"

N3Attenuation[1.,1.67,{1.3,1.5,0.05},3.85 - 0.02 I,0.5,0.5,0.6328]

{1.3, $-6.59959\ 10^{-9}$ }
{1.35, $-9.75759\ 10^{-7}$ }
{1.35, $-1.26601\ 10^{-8}$ }
{1.4, $-2.06399\ 10^{-6}$ }
{1.4, $-2.59717\ 10^{-8}$ }
{1.45, $-5.77298\ 10^{-8}$ }
{1.5, $-1.41408\ 10^{-7}$ }



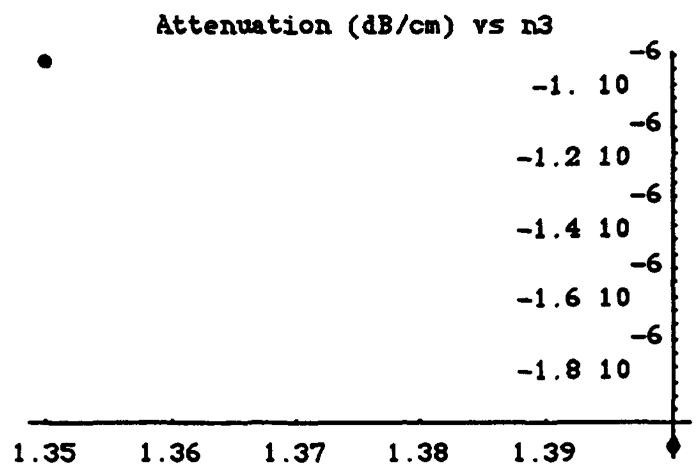


Figure 5 Example of "N3Attenuation" Procedure

3.1.5.4. PROCEDURE "N4Attenuation"

"N4Attenuation" is like the previous attenuation functions, except there are special cases for n_4 since it is a complex number. The variables "imaginaryVar" and "realVar" are used for varying such parts and variable is the combination of the "Re" and "Im" part. A variation of n_4 may result in values such as 3.80 - 0.01 I and 3.80 - 0.02 I. The attenuation values are considered different for the number of modes, but the values will be plotted above/below each other as only the Re[n_4] is used in plotting.

```
N4Attenuation[1.,1.67,1.46,{3.80 - 0.01 I,3.9 - 0.02 I,0.05 - 0.005
I},0.5,0.5,0.6328]
```

```
{3.8 - 0.01 I, -6.95785 10^-8 }
{3.85 - 0.01 I, -6.85647 10^-8 }
{3.9 - 0.01 I, -6.75815 10^-8 }
{3.8 - 0.015 I, -6.95353 10^-8 }
{3.85 - 0.015 I, -6.85236 10^-8 }
{3.9 - 0.015 I, -6.75423 10^-8 }
{3.8 - 0.02 I, -6.94919 10^-8 }
{3.85 - 0.02 I, -6.84822 10^-8 }
{3.9 - 0.02 I, -6.75028 10^-8 }
```

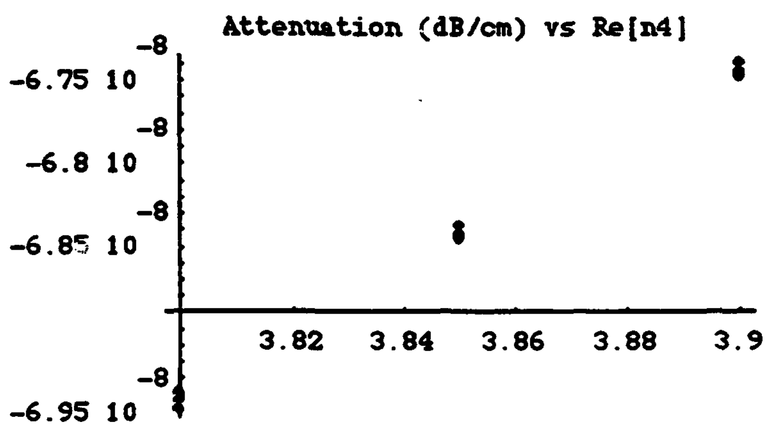


Figure 6 Example of "N4Attenuation" Procedure

3.1.5.5. PROCEDURE "GuideAttenuation"

GuideAttenuation[1.,1.67,1.46,3.85 - 0.02 I,1,{0.5,0.52,0.005},0.6328]

{0.5, $-8.26635\ 10^{-11}$ }
{0.505, $-7.82954\ 10^{-11}$ }
{0.51, $-7.42218\ 10^{-11}$ }
{0.515, $-7.0419\ 10^{-11}$ }
{0.52, $-6.43929\ 10^{-11}$ }

Attenuation (dB/cm)
vs
Guiding Layer Thickness (microns)

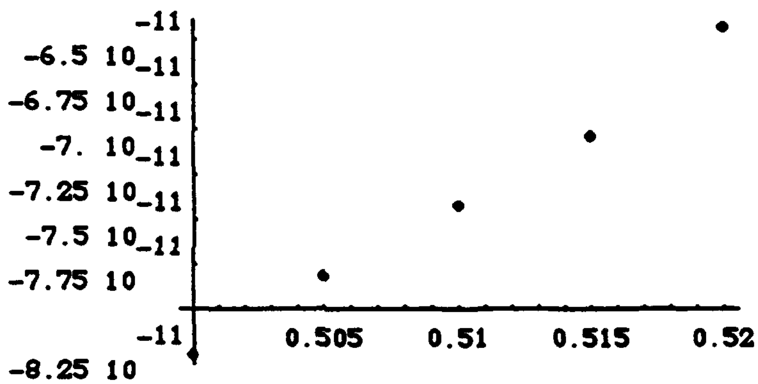


Figure 7 Example of "GuideAttenuation" Procedure

3.1.5.6. PROCEDURE "CladdingAttenuation"

CladdingAttenuation[1.,1.67,1.46,3.85 - 0.02 I,{0.5,1,0.1},0.5,0.6328]

{0.5, -6.84822 10^{-8} }
{0.6, -1.78486 10^{-8} }
{0.7, -4.65518 10^{-9} }
{0.8, -1.21442 10^{-9} }
{0.9, -3.16838 10^{-10} }

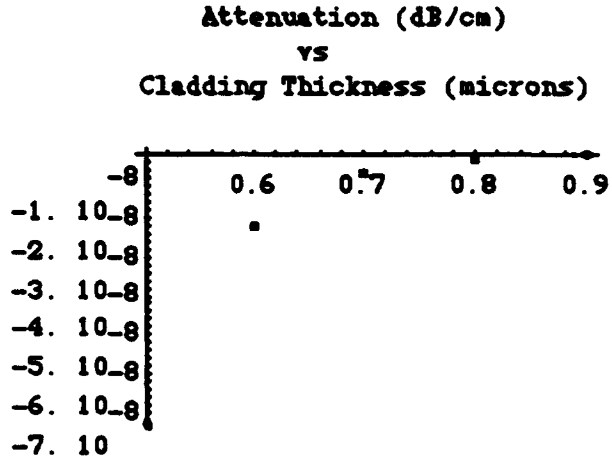


Figure 8 Example of "CladdingAttenuation" Procedure

3.1.5.7. PROCEDURE "WavelengthAttenuation"

WavelengthAttenuation[1.,1.67,1.46,3.85 - 0.02 I,0.5,0.5,{0.5,0.7,0.05}]

{0.5, $-5.09033\ 10^{-7}$ }
{0.55, $-1.70656\ 10^{-8}$ }
{0.6, $-4.13852\ 10^{-8}$ }
{0.65, $-8.73335\ 10^{-8}$ }
{0.7, $-1.64989\ 10^{-7}$ }

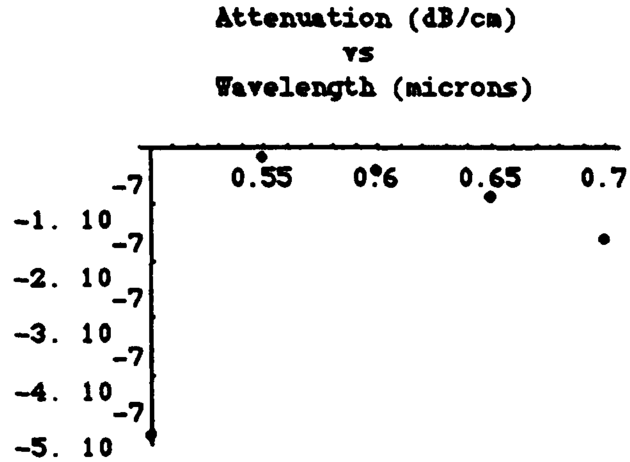


Figure 9 Example of "WavelengthAttenuation" Procedure

3.1.6. PROCEDURE "CutoffThickness"

The cutoff thickness ("cutoffGuide") is calculated from the three-layer eigenvalue equation as the value of the effective index of refraction (proportional to the propagation constant) approaches the index of the third (cladding) layer. Error messages and appropriate "Break" statements are implemented in cases of error. A plot of cutoff thickness versus n_2 is output along with the values.

```
CutoffThickness[{1.5,1.6,0.01},1.,1.46,0.6328]
1.5  0.368192
1.51  0.319696
1.52  0.284015
1.53  0.256386
1.54  0.234201
1.55  0.2159
1.56  0.200485
1.57  0.187282
1.58  0.175819
1.59  0.165753
1.6  0.156829
```

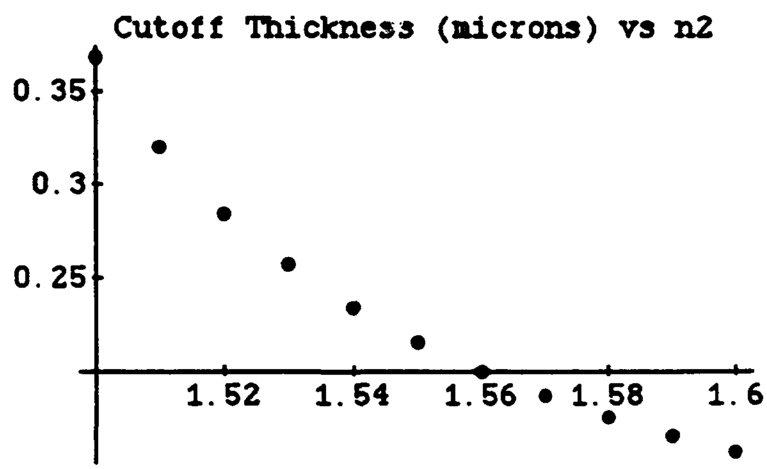


Figure 10 Example of "CutoffThickness" Procedure

3.1.7. PROCEDURE "CutoffWavelength"

The cutoff wavelength ("cutoffWave") is calculated from the three-layer eigenvalue equation as the value of the effective index of refraction approaches the index of the third (cladding) layer. Error messages and appropriate "Break" statements are implemented in cases of error. A plot of cutoff wavelength versus n_2 is output along with the values.

```
CutoffWavelength[{1.5,1.6,0.01},1.,1.46,0.5]
1.5  0.859333
1.51  0.989692
1.52  1.11402
1.53  1.23408
1.54  1.35098
1.55  1.46549
1.56  1.57817
1.57  1.68943
1.58  1.79958
1.59  1.90886
1.6  2.01748
```

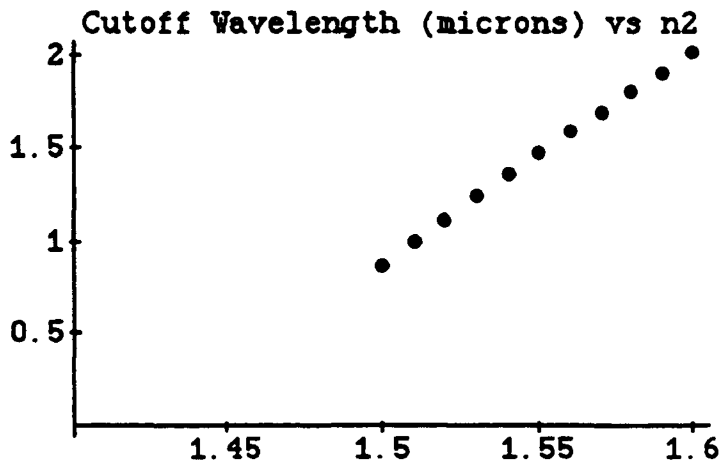


Figure 11 Example of "CutoffWavelength" Procedure

3.2. MODELING EXAMPLE

As an example, consider a structure consisting of 0.4 μm guiding layer of PMMA (n=1.489) on 0.65 μm SiO_2 (n=1.46) on a silicon substrate (n=3.85-0.02 I) operating at a wavelength of 0.6328 μm. It is useful to know, before actually attempting to waveguide, how many guided modes are supported by the structure. The parameters are entered into the code (see Section 9.1. for details) with the following result:

PropagationConstant[1.0,1.489,1.46,3.85 - 0.02 I,0.4,0.65,0.6328]

There are no guided modes for this four-layer guide.

The structure may, however, support modes at other wavelengths. These wavelengths can be located using this code as outlined in Section 3.1.7. The result is shown in Figure 12. It can be seen that 0.564μm is the cutoff wavelength for this structure. This means that if a shorter wavelength than 0.564μm is chosen, there is a guided mode for this structure. We can now calculate the modal behavior of the guide at 0.514μm. The propagation constant β and the fields in each layer are calculated and plotted as shown in Figure 13. Valuable experimentation time is saved by noting how ill-confined this particular mode is and fabricate another sample with thicker guiding and buffer layers for better confinement. As a final step, a sample with 0.5μm PMMA and 2μm SiO_2 at 0.514μm is examined. This calculation is shown in Figure 14. This structure has two guided modes, the lower order mode being fairly well confined.

CutoffWavelength[{1.479,1.501,0.002},1.0,1.46,0.04]

1.479	0.439212
1.481	0.465665
1.483	0.491297
1.485	0.516219
1.487	0.540522
1.489	0.564279
1.491	0.58755
1.493	0.610388
1.495	0.632833
1.497	0.654925
1.499	0.676694

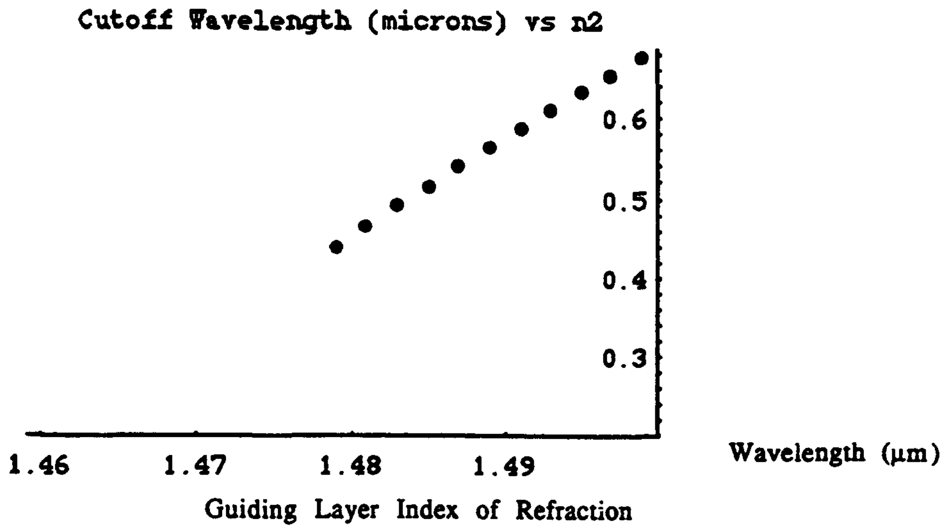


Figure 12 Theoretical Calculation of Cutoff Wavelength for 0.4µm PMMA on 0.65µm SiO_2 on Si

PropagationConstant[1.0,1.489,1.46,3.85 - 0.02 I,0.4,0.65,0.514]

17.9227 - 0.00375371 I

Begin plotting routine.....

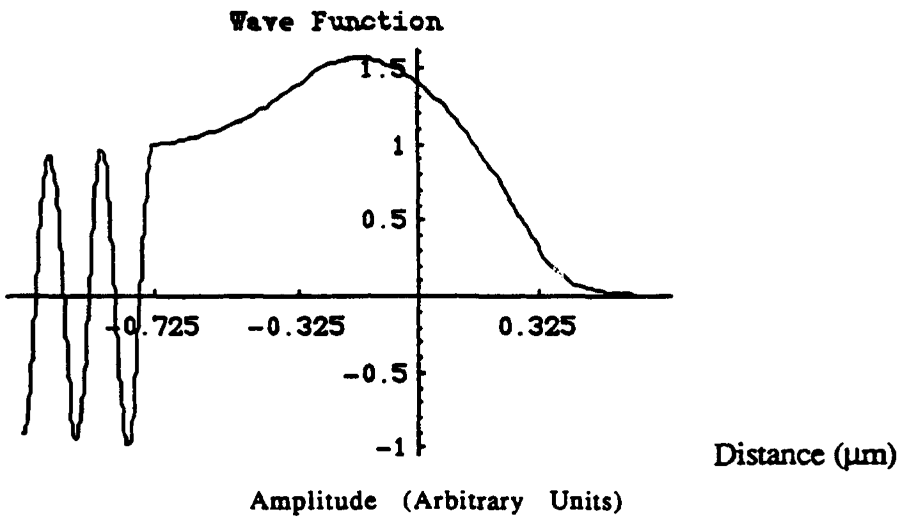
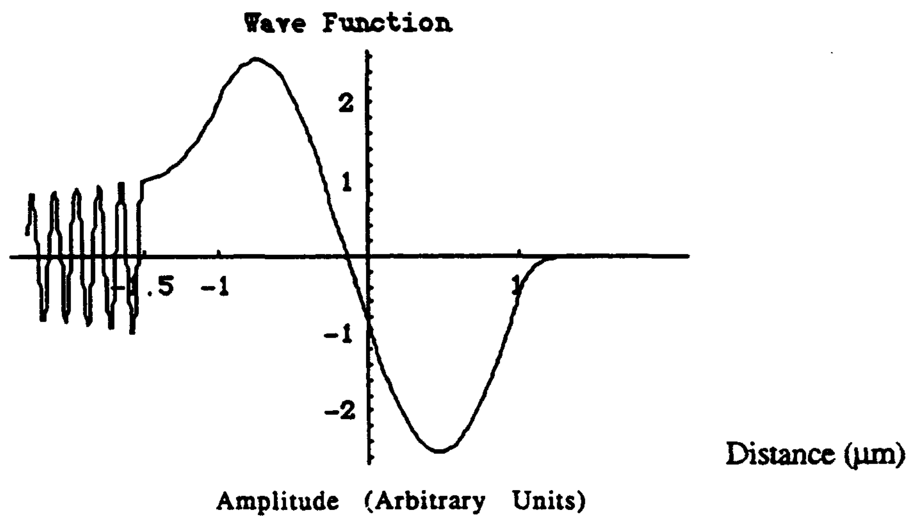


Figure 13 Theoretical Calculation of the Propagation Constant for 0.4μm PMMA on 0.65μm SiO_2 on Si at λ=0.514mm

PropagationConstant[1.0,1.489,1.46,3.85 - 0.02 I,0.5,2,0.514]

18.0052 - 0.000628331 I
18.1522 - 0.000114449 I

Begin plotting routine.....



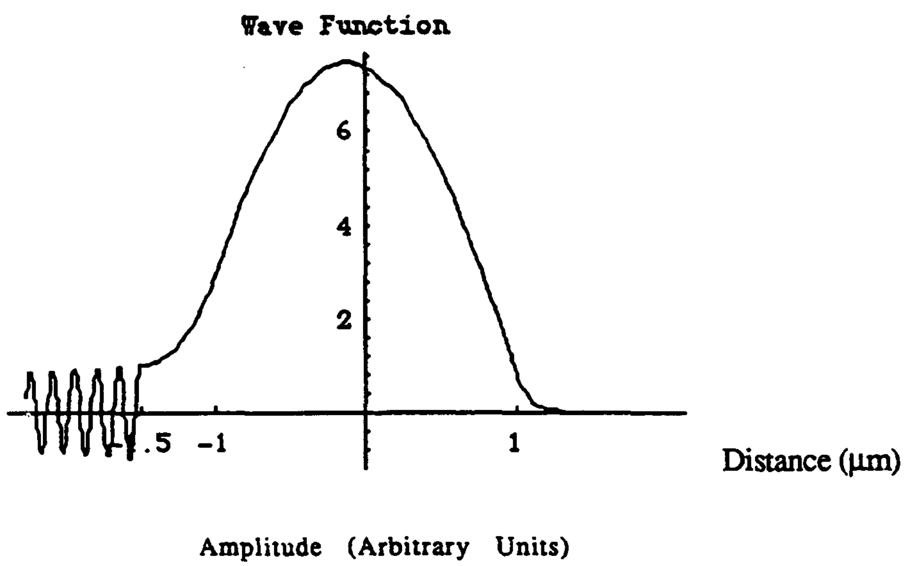


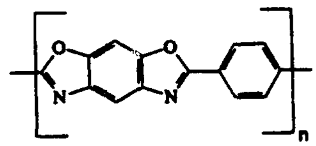
Figure 14 Theoretical Calculation of the Propagation Constant for 0.5μm PMMA on 2.0μm SiO_2 on Si at λ=0.514μm

4.0. WAVEGUIDE MATERIAL

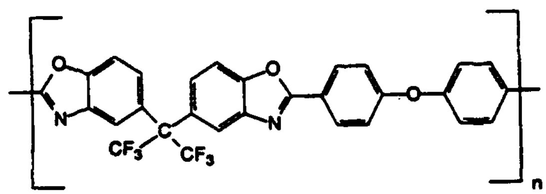
Recent work in optical waveguides has been in developing a material that has both good optical qualities as well as long-term environmental stability and durability. One area of materials interest is in polymers. Unfortunately, many of the well-known polymers that exhibit desirable optical characteristics have a low (less than 100°C) glass transition temperature (T_g) which can cause drastic deterioration of several key mechanical and physical properties as the polymer chains gain high mobility at moderate temperatures. Low T_g polymers are also more susceptible to thermal degradation with short, high temperature excursions experienced in commercial waveguide processing techniques. Having a high T_g and visual transparency, a hexafluoro-isopropylidene-polybenzoxazole (6F-PBO), became a primary candidate.

Having been synthesized by various methods approximately 25 years ago[5-9], polybenzoxazole polymers have been known for their high strength, high modulus, and high thermo-oxidative stability, but have also suffered from limited solubility and low processability which, overall, has severely hampered their material development. One manner in which these limitations have been overcome is through the incorporation of a 2,2-hexafluoroisopropylidene (6F) group into the polybenzoxazole backbone to produce a more transparent, soluble polymer which did not sacrifice thermo-oxidative stability[10-11]. Having high use temperature comparable to that of epoxy and polyimide systems, the 6F polybenzoxazoles (6F-PBO) have an advantage in that unlike the polyimides which are only soluble in the prepolymer form, it is soluble in common organic solvents in the closed polymer form. They also have the advantage that their Tg can be varied from 94°C to 382°C through structural modification of the aromatic spacer group in the backbone[12]. The version of 6F-PBO which was examined in this paper contained a paralinked diphenyl ether aromatic spacer group (Figure 15) which resulted in a Tg of 300°C. This polymer is denoted 6F-PBO-1E.

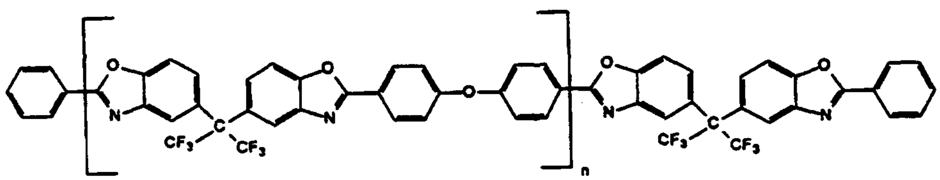
Having established 6F-PBO-1E as meeting the requirements on processability and T_g, the optical properties were then examined. This technical report describes our findings in the investigation of the optical waveguide loss measurements on the polymer material 6F-PBO-1E.



PBO (Poly-p-phenylene benzobisthiazole)
(rigid-rod polymer)



6F-PBO-1



6F-PBO-1E (End-capped)

Figure 15 6F-PBO and Variations

4.1. WAVEGUIDE FABRICATION

4.1.1. THE SUBSTRATE

The substrate material used for the optical waveguide fabrication is silicon. Wafers with <100> orientation were used. Due to the silicon high index of refraction, an oxide layer is necessary for guided wave propagation. The oxide layer was grown with steam at

1100°C for approximately 4 hours. Thickness calculations were accomplished with the growth modeling package TSupreme4 by Technical Modeling Associates. The 2- and 3-inch wafers were oxidized separately and their calculated oxide thicknesses are 1.4 μm and 1.5 μm, respectively.

The substrate material was measured for thickness and refractive index values using an ellipsometry analysis software package, DAFIBM, in conjunction with a fixed wavelength (λ = 632.8nm) Rudolph Research AutoEL II automated nulling ellipsometer. The results of this analysis are shown in Figures 16-19. The 3-inch wafers have considerable variation in both index and thickness, most likely due to a slight bowing of the wafer after oxidation. Based on the analysis of the two wafer types, the majority of the guided wave loss measurements were done on the 2-inch wafers.

Values for index of refraction and thickness for both the oxide layer and the polymer layer are averaged over the 2-inch sample values only. The oxide layer thickness was found to be 1.30μm and 1.43μm for the 2- and 3-inch wafers, respectively. The index of refraction of the oxide layer is taken to be 1.455. A common base clean procedure was used to prepare the wafers for spinning (see Section 8.0).

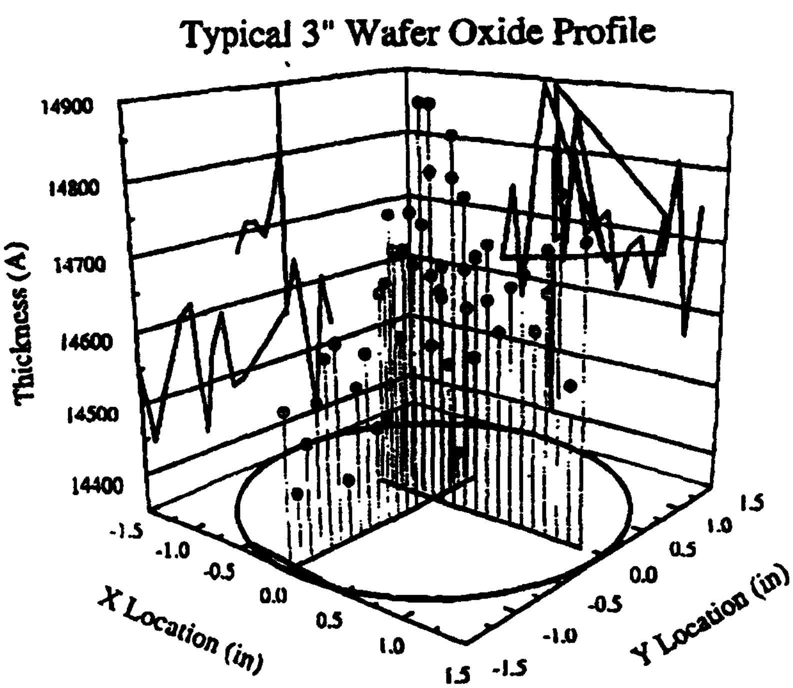


Figure 16 Oxide Thickness Profile taken on Profilometer for 3-Inch Oxidized Silicon Wafer

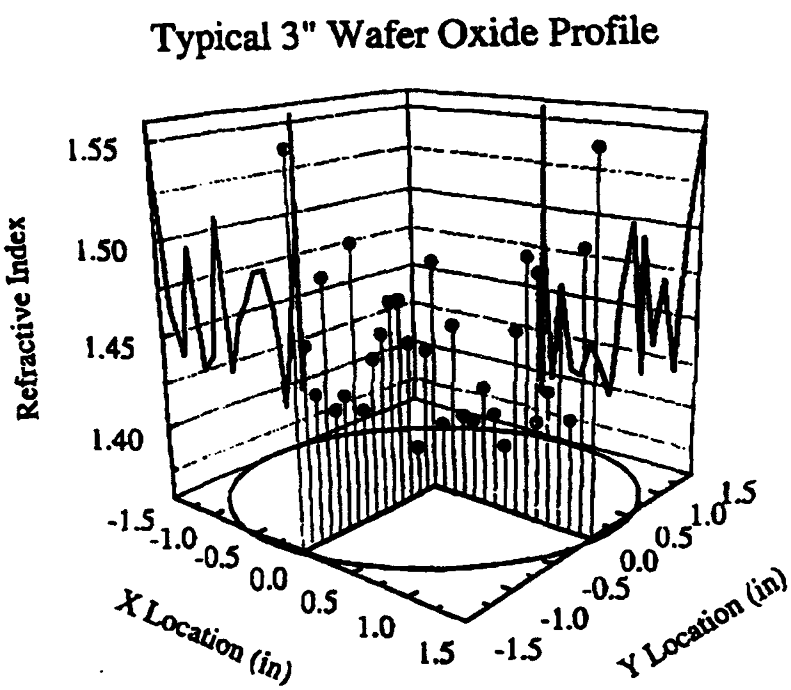


Figure 17 Refractive Index Profile of Oxide Layer taken on Ellipsometer for 3-Inch Oxidized Silicon Wafer

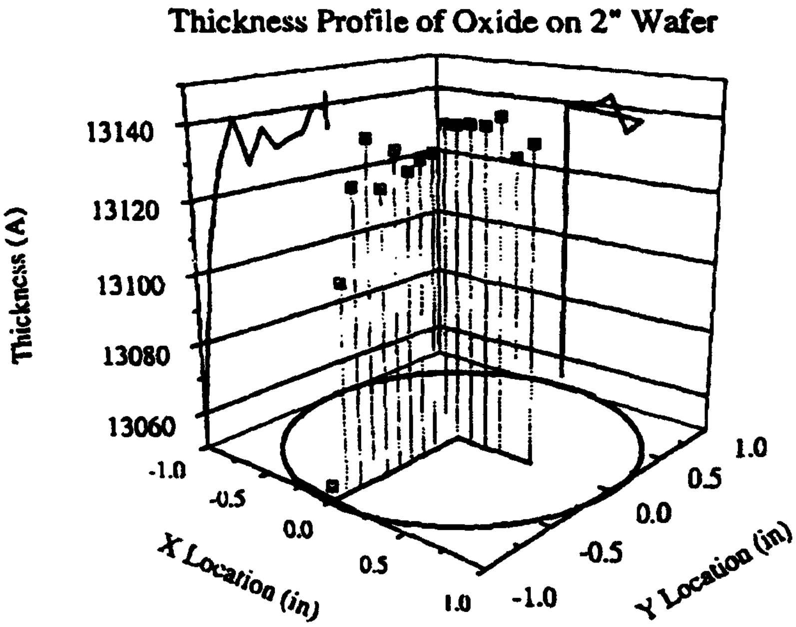


Figure 18 Oxide Thickness Profile Taken on Profilometry for 2-Inch Oxidized Silicon Wafer

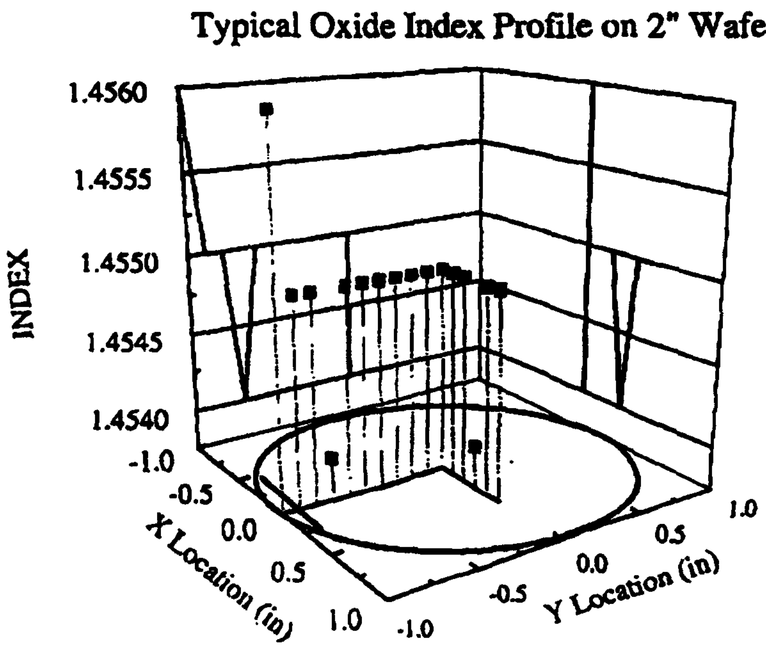


Figure 19 Refractive Index Profile of Oxide Layer taken on Ellipsometer for 2-Inch Oxidized Silicon Wafer

4.1.2. THE GUIDING MATERIAL

The polymer was prepared by Daychem, Inc. in a manner similar to that described by Reinhardt[12] with an additional post-polymerization end-capping of the polymer chains with benzoic acid[13]. The polymer was end-capped to inhibit reaction of the chain ends at elevated temperatures. Due to the polymer high temperature stability, high solubility, and transparency, it appeared to offer great promise as an optical waveguide material and possibly even as a host for a guest-host, second-order nonlinear optical film.

In an attempt to find the optimum organic solvent that would both dissolve the polymer and produce a high quality spin-coated film, the polymer was dissolved in several organic solvents. The solvents which caused the most solubility included chloroform, 1,1',2,2'-tetrachloroethane, 2-picoline, and 4-picoline (each at 5 wt.%). Each solution was pipetted onto 1" x 1" microscope glass slide and spun at 3000rpm for 60 seconds on the Solitec 5110-CT spin coater in a clean room environment. Optical inspection and Dektak II 2A profilometry results indicate that the film spun from the tetrachloroethane solution was the most transparent film with the most uniform thickness and least surface roughness.

Efforts were made to increase the concentration of the tetrachloroethane solution in order to produce a thicker film under the same spinning conditions. A thicker film would support more modes of propagation and more beam confinement, but a more concentrated and viscous solution would also be more difficult to syringe filter through a 0.5μm Teflon filter as is required for reduction of dust and other scattering centers. The same spin parameters were employed due to the initial success in film quality and thickness uniformity. A 7 wt% solution of the 6F-PBO-1E in tetrachloroethane was successfully filtered after many unsuccessful attempts at higher concentrations.

The 6F-PBO-1E from the 7 wt% solution in tetrachloroethane was spun onto the five oxidized silicon substrates at 3000 rpm for 60 seconds. Each wafer was baked out at 80°C for 12 hours following the spin to drive off the tetrachloroethane solvent.

4.2. SAMPLE CHARACTERIZATION

Visual inspection of the resulting baked samples revealed limited inclusions and, overall, relatively good film quality. These samples were then evaluated to determine index of refraction, thickness, and absorption before attempting to waveguide.

4.2.1. GUIDING LAYER THICKNESS

The thickness was measured on a Dektak IIA Profilometer. A radial scratch was induced in an unaltered region of the polymer film and the step-height of the scratch was measured. Figure 20 shows the results of this study. The average over all five samples was 1.00μm and this value was used in the theoretical evaluation. As is typical in spin coating procedures, there is a relatively uniform thin area in the center of the sample, and increasing thickness at the edges. Due to this fact, all measurements were made on the uniform portion of the samples. In addition, this flat uniform region is more prominent on the 2-inch wafers, and the information previously presented for the oxide layer and index of refraction data was only recorded on those samples.

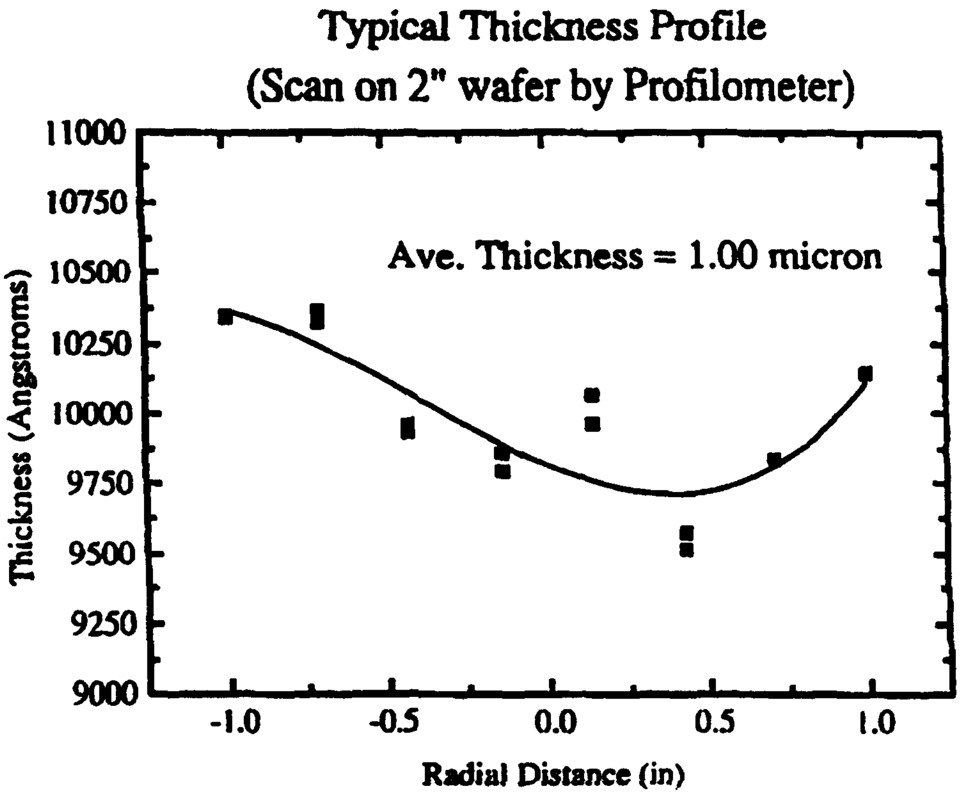


Figure 20 Polymer Thickness Profile taken on Profilometer for 2-Inch Wafer

4.2.2. INDEX OF REFRACTION

The index of refraction of the polymer film was determined using a four-layer (substrate - oxide - film - air) ellipsometry analysis software package, DAFIBM, in conjunction with a fixed wavelength (λ = 632.8nm) Rudolph Research AutoEL II automated nulling ellipsometer. If the index and thickness of the oxide layer and the thickness of the polymer layer is known, this software program can be used to calculate the index of the polymer layer.

The index of refraction was found by performing two perpendicular radial scans on both the 2- and3-inch wafers. Figure 21 shows the data taken on the 2-inch sample with the ellipsometer. Analysis of the data showed that a variation in index would generally consist of a level region near 1.69, increasing to indices close to 2.0 on the edges. The higher index outer regions are currently a source of further study.

To confirm the index measurements made using the ellipsometer, a more easily analyzed single layer of polymer on bare silicon was fabricated under the same processing conditions from the same solution as the waveguiding samples. In addition to analysis by ellipsometry, this new film was analyzed using a Metricon 2010 Prism Coupler where the index and thickness of the film were measured by recording the angles at which modes are coupled into and out of the film by a symmetric prism (technique described by Tien and Ulrich[14]). Index and thickness measurements were obtained in the inner 1-inch diameter of a 2-inch sample. Within this region, the average index as measured with the ellipsometer was found to be 1.668 (2.5% variation), while over five equally spaced regions on the wafer, the Metricon measured the index to be 1.6647 (0.01% variation) in the TE mode at 632.8nm.

The average polymer layer index of refraction, over the three procedures performed (see Table 1), is 1.691.

Refractive Index Profile in Uniform Region of 2" Wafer
(Scanned by Ellipsometer)

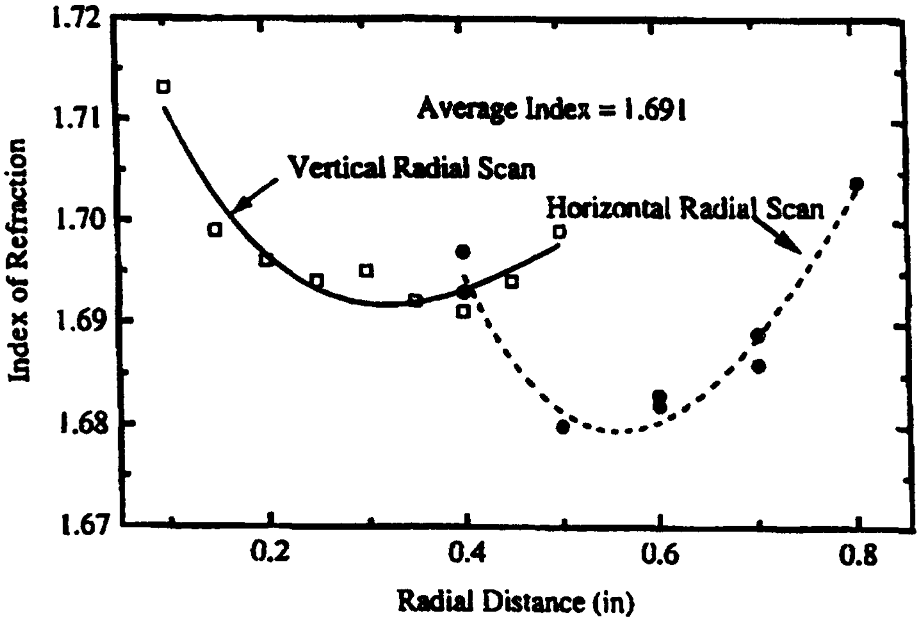


Figure 21 Refractive Index Profile of Polymer Layer taken on Ellipsometer for 2-Inch Wafer

Procedure	6F-PBO-EC on Oxidized Wafers		6F-PBO-EC on Unoxidized Wafers	
	Index	Thickness	Index	Thickness
Ellipsometer	1.692	0.98μm	1.668	1.06μm
Metricon	------	------	1.6647	1.06μm
Ellipsometer	------	1.00μm	------	------

TABLE 1 Comparison of Three Methods for Finding Index of Refraction and Thickness of 6F-PBO-1E

4.2.3. ABSORPTION

Material-inherent properties are not considered in the theoretical calculations so it is appropriate to investigate the effects the material itself will have on the waveguiding properties.

In order to evaluate the absorption losses for 6F-PBO-1E, a free-standing film was cast through slow evaporation of the solution in a Petri dish. Transmission studies of this film at normal incidence over the wavelength range 250 - 3200nm were then performed using a Hitachi U-4001 UV/VIS/NIR spectrophotometer. The results are shown in Figure 22. For an absorbing film at normal incidence, the following equations for Reflectivity, R, Transmissivity, T, and absorption coefficient, α, apply.[15-16]

$$T = \frac{(1 - R)^2 e^{-\alpha d}}{1 - R^2 e^{-2 \alpha d}} \quad \text{where d is the thickness of the film} \tag{19}$$

$$\alpha = \frac{1}{d} \ln\left(\frac{(1 - R)^2 + \sqrt{(1 - R)^4 + 4 T^2 R^2}}{2 T}\right) \tag{20}$$

Assuming the film to be non-absorbing ($\alpha = 0$) in the region where the transmission is maximum, the index of the film can be calculated using[17]:

$$T = \frac{(1 - R)^2}{1 - R^2} \tag{21}$$

where the reflectance for a free-standing, non-absorbing film is[4]:

$$R = \left(\frac{n - 1}{n + 1}\right)^2 \tag{22}$$

Transmission Spectra for Free-Standing Film

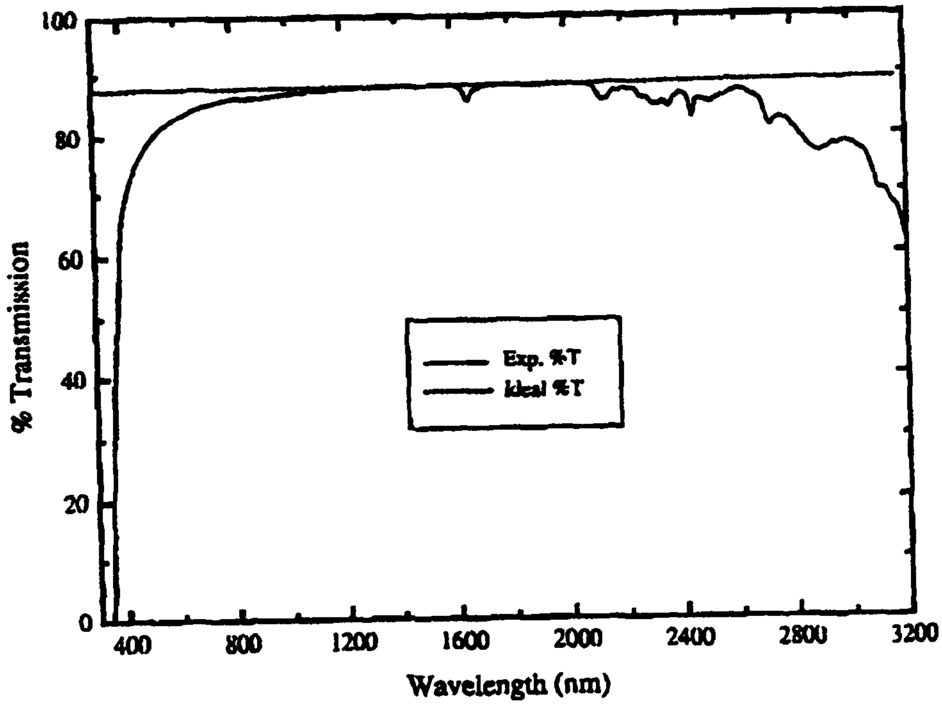


Figure 22 Transmission Data Taken on Free-Standing Film of 6F-PBO-1E with Spectrophotometer

Solving for the index of refraction gives:

$$n = \frac{2 + \sqrt{4 - 4\,T^2}}{2\,T} \tag{23}$$

The assumption is then made that dispersion is negligible so the index found at the maximum transmission wavelength is a constant in the calculation of the absorption coefficient over all wavelengths. The reflectivity is also assumed to be a constant. These assumptions allow only for a semi-qualitative analysis of the film, but this is sufficient to investigate the absorption trends.

The absorption coefficient α can then be calculated using:

$$\alpha = \frac{1}{d} \ln\left(\frac{(1 - R)^2 + \sqrt{(1 - R)^4 + 4\,T^2 R^2}}{2\,T}\right) \tag{24}$$

where R utilizes the formula for non-absorbing films, T is that measured experimentally with the spectrophotometer, and d is the thickness of the film as measured with the profilometer.

Figure 23 depicts the findings of this investigation. The loss coefficient, a, as calculated from equation (23) depicts strong absorption in the shorter wavelength region. This information was useful in evaluating the experimental waveguide loss data.

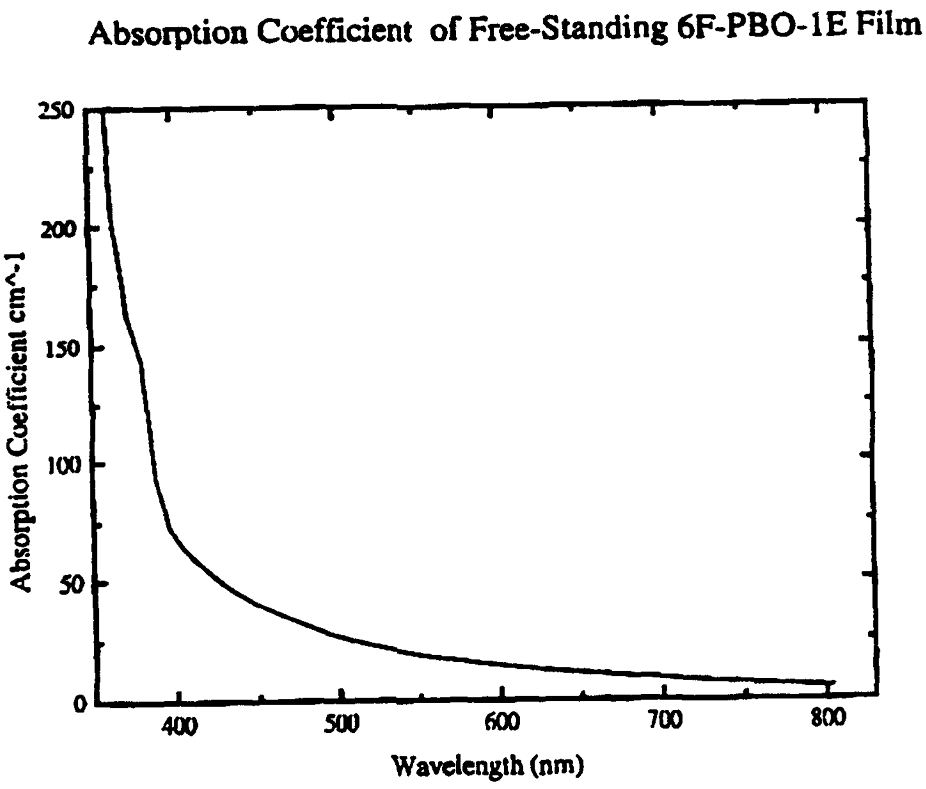


Figure 23 Absorption Coefficient for 6F-PBO-1E as Calculated from the Transmission Data

4.3. THEORETICAL BEHAVIOR

Once the index of refraction and thickness of both the oxide and guiding layers of the samples have been measured, the package presented in Section 3.0 was used to predict the behavior of the material as a planar waveguide.

The loss of each confined mode has been calculated at several wavelengths, and the combined results are shown in Figure 24. There is at least one confined mode for all wavelengths of interest, therefore, the samples fabricated theoretically fulfill the requirements for measuring the optical attenuation.

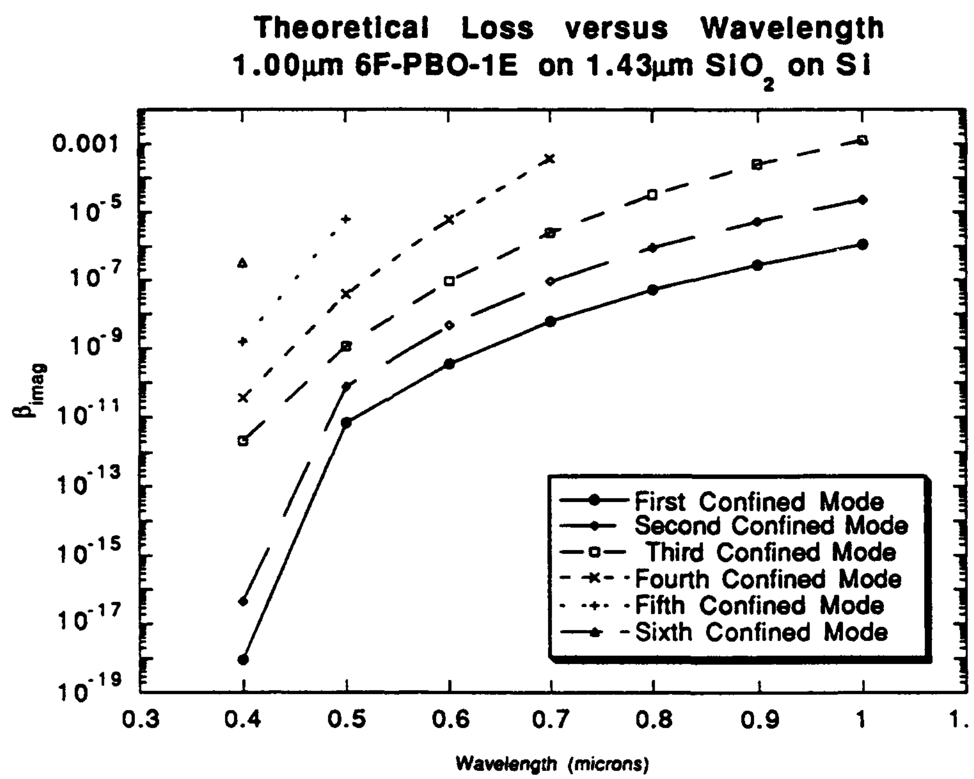


Figure 24 Theoretical Plot of Loss versus Wavelength Computed with *Mathematica*

5.0. EXPERIMENTAL MEASUREMENTS OF WAVEGUIDE ATTENUATION

5.1. Introduction

When studying a material for possible use in optical integrated circuits, one important consideration is attenuation in the waveguide. Waveguide attenuation may have many causes including absorption, scattering, and substrate coupling.

The energy loss in the waveguide may be modeled using Beer's Law:

$$I = I_0 \, e^{-\alpha z} \quad (25)$$

where I_o is the input intensity
α is the loss coefficient and
z is the distance of propagation

By definition:

$$1 \text{ dB} \equiv 10 \log\left(\frac{I}{I_0}\right) \quad (26)$$

Taking z = 1 cm and transforming logarithm bases yields:

$$\log\left(\frac{I}{I_0}\right) = -0.434 \, \alpha \text{ cm} \quad (27)$$

Transforming this expression into dB notation gives:

$$\frac{\text{dB}}{\text{cm}} = -4.34 \, \alpha \quad (28)$$

where α is the attenuation coefficient.

5.2. LOSS MEASUREMENT TECHNIQUES

The absorption coefficient, α, can be measured by any of three common loss measurement methods: three-prism method[18], two-prism method, or out-of-plane scattering method. This study uses all three and compares the results. A custom-designed waveguide-loss-measurement apparatus[19] is used to perform all three types of measurements.

5.2.1. THE THREE PRISM METHOD

In the three prism method[18], the input, coupling, and output prisms and their associated detectors are arranged as in Figure 25. Prisms 1 and 3 are clamped while 2 is moveable. Light is coupled into the guide with prism 1, called the input prism, and out of

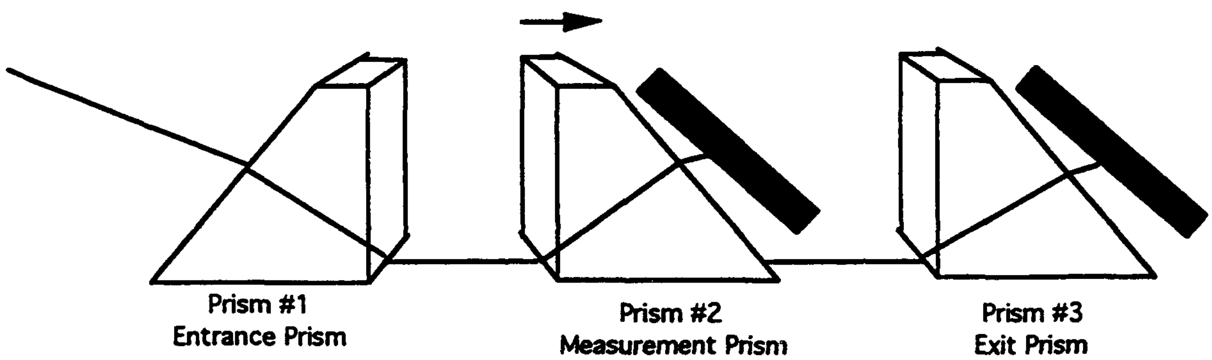


Figure 25 Experimental Setup for the Three-Prism Method to Measure Attenuation

the guide through prisms 2 and 3, referred to as the measurement and output prisms, respectively. Measurement of the light intensity from both prisms 2 and 3 eliminates the need to know the coupling coefficient for the moveable prism[18].

The setup works by first measuring the output at prism 3 without prism 2 clamped, then measuring the output of the 2nd and 3rd prisms as a function of distance as prism 2 moves along the guide. The intensity can then be calculated as a function of distance with the relation:

$$I(z) = \frac{P_2(z)\, P_3^0}{P_3^0 - P_3} \tag{29}$$

where $P_3{}^0$ is the output measured at prism 3 without prism 2 clamped, $P_2(z)$, and P_3 are the outputs of prisms 2 and 3, respectively.

A Schott SF-6 45°-45°-90° coupling prism and a Hamamatsu S1227-66BQ photodiode with a 33 mm^2 active area are used in each of the measurement positions. The active area of the detector is parallel to, and covers the out-coupling face of, the prism. The prism-detector assembly moves in unison to insist on uniformity. The procedure is computer controlled using a Hewlett-Packard 9000-330 computer and multiprogrammer. The waveguide is placed on the platform, and the input coupling prism is clamped in place. The operator adjusts the input beam for propagation through, and total internal reflection (TIR) inside, the coupling prism. TIR, by definition, creates evanescent fields outside the prism boundaries, in the air gap between the prism and sample. This evanescent field is coupled into the sample and becomes a guided mode if TIR is also satisfied inside the polymer layer. The measurement and output coupling prism assemblies are then positioned over the guide. The computer records the clamping pressure by means of load cells located in the prism arm assembly. The operator chooses a measurement distance and increment size over which to take measurements and then, subsequently, clamps and unclamps the measurement prism as the computer moves the measurement arm and records the data. Finally, measured intensity data are output to a computer file for plotting.

Once the intensity profile has been established, the data can be fitted to an exponential function (according to Beer's Law), and the attenuation coefficient can be determined.

5.2.2. THE TWO PRISM METHOD

The two-prism loss measurements are performed in much the same manner as the three-prism method, except of course, the elimination of the need for the third prism. In the two prism method, the load cells incorporated in the prism arm assembly of the custom apparatus[19] allow assurance of constant coupling efficiency, therefore, eliminating the need for the third prism. One disadvantage in both the prism methods, however, comes when measuring soft films. Each time the prism is clamped down on the sample, it has the possibility of damaging the film surface. For this reason, a non-contact method of loss measurement is primarily used in this study.

5.2.3. THE OUT-OF-PLANE SCATTERING METHOD

The non-contact method for determining the attenuation coefficient is the third common waveguide loss measurement technique, referred to as the out-of-plane scattering method. In this method, there is an input coupling prism and a moveable detector as shown in Figure 26. The detector is a Hamamatsu S1227-66BQ photodiode with a 33 mm^2 active area masked to a 1-mm-wide slit positioned perpendicular to the guiding streak. The detector is moved along the guiding streak, holding the position above the sample constant with the micrometer and arm assembly. The intensity of the light scattered perpendicular to the guiding streak is measured as a function of distance from the coupling spot. The intensity profile is fit to an exponential function, and the attenuation coefficient is determined from the slope of the curve. This method is much simpler than the three-prism method, and is not only used with delicate films, but with all types of samples.

The apparatus used for the out-of-plane scattering method is the same as that used for the prism methods. In this procedure, the first two arms are used. The measurement arm no longer has a prism attached but is fit with a detector which is parallel to the platform. As in the three-prism method, the operator adjusts the input beam for TIR. However, the computer now controls the movement of the detector along the streak, records all pertinent information, and outputs it to a file for inspection.

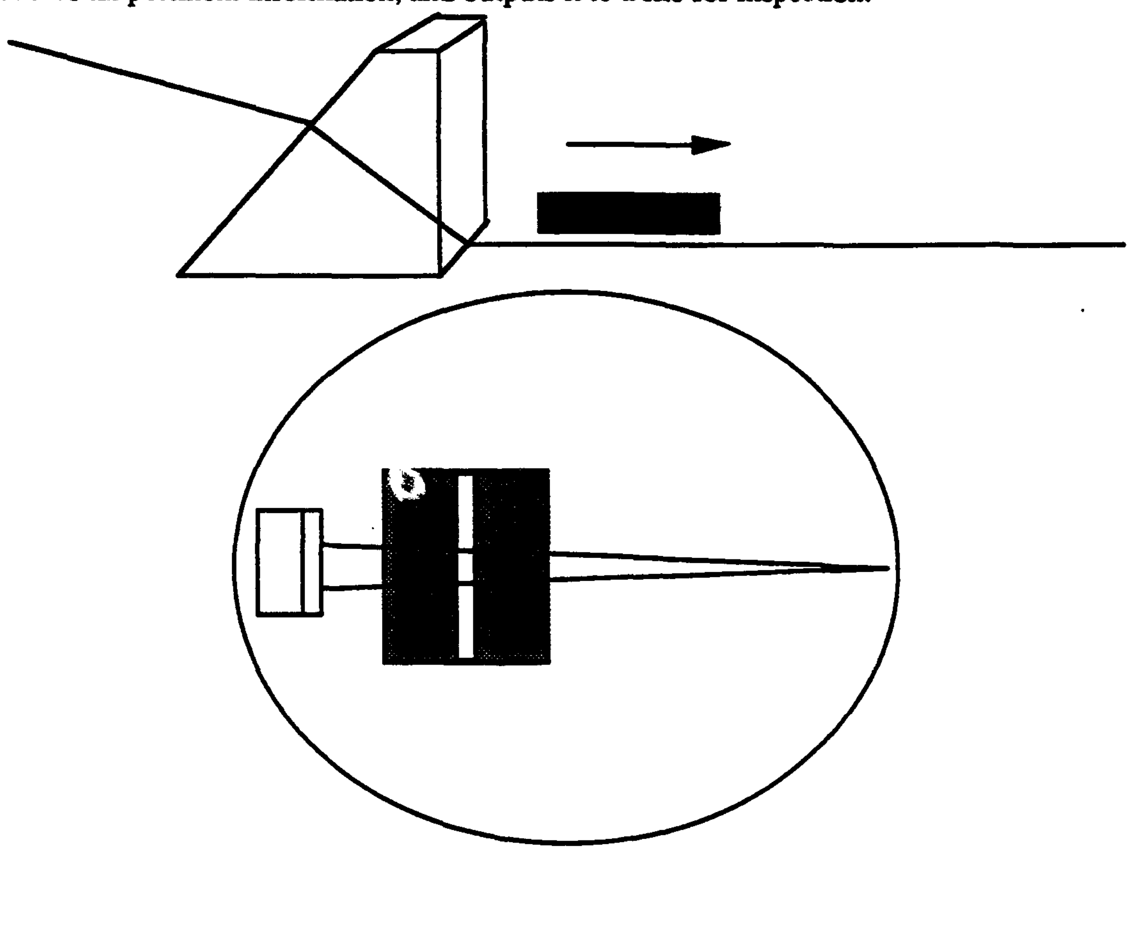


Figure 26 Experimental Setup for the Out-of-Plane Scattering Method, Showing Detector Positioning and Masking.

5.3. EXPERIMENTAL PARAMETERS

Loss measurements were conducted at wavelengths between 0.4579 and 1.0μm. The sources were a Spectra-Physics Argon Ion laser, a Coherent 599-01 Dye laser using Rhodamine 6G and a Spectra-Physics 3950 Ti:Sapphire laser. A Burleigh Model WA-10 Wavemeter WA-10 was used to measure the wavelength of the input light and a Newport SP 550 polarization rotator was used to ensure TE modes. Schott SF-6 glass (n=1.805) prisms were used to couple the light into the guide. The beam was weakly focused at the point of contact with the guide with a 1000 mm focal length lens. Loss measurements were performed on each of five samples over the wavelength range 0.46 - 0.975μm.

Five samples of end-capped 6F-PBO were fabricated on oxidized silicon wafers, two on 3-inch wafers and three on 2-inch wafers. Preliminary observations of guided streaks suggest this material has a relatively high loss. Therefore, most of the loss measurements were made using the out-of-plane scattering technique on the 2-inch wafers. However, some three-prism data were taken on the 3-inch wafers. The 2-inch wafers could not be measured with the three-prism method due to a size constraint. Some data were taken on these samples by the two-prism method where the coupling coefficient of the measurement prism was strictly monitored and kept constant throughout the experiment using the load cells accommodated in the arm of the stage. Due to the destructive nature of the prism out-coupling methods on polymer materials, very few data points could be collected on these samples and only at wavelengths where the material has low absorption.

5.4. RAW WAVEGUIDE DATA

Figures 27-32 are representative plots of the raw data taken on one of the 2-inch samples with the out-of-plane scattering method. The raw data consists of the relative positions of the measurement prism and the corresponding intensities. The data are plotted with the software package KaleidaGraph Version 2.1.3, and the standard exponential fit included in that package is calculated. The best fit exponential curve, along with the correlation value, R, is displayed in the bottom left corner of each plot. The sample used to obtain the data in Figures 27-32 consists of a 1.0μm film of 6F-PBO-1E on 1.4μm oxidized silicon on a 2-inch silicon wafer. The operating wavelengths are noted on the plots along with the measured loss in dB/cm.

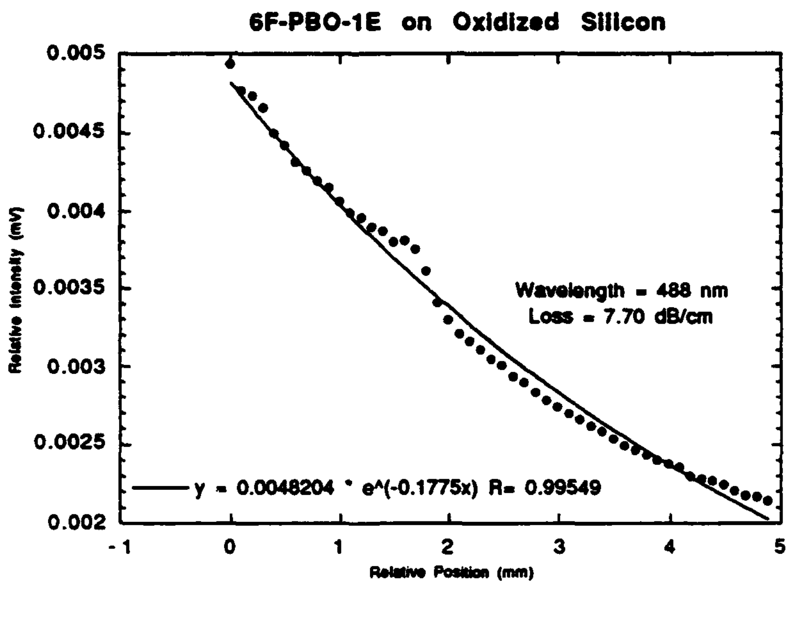


Figure 27 Raw Data for 1.0μm 6F-PBO-1E on 1.4μm SiO_2 on Si at 488nm

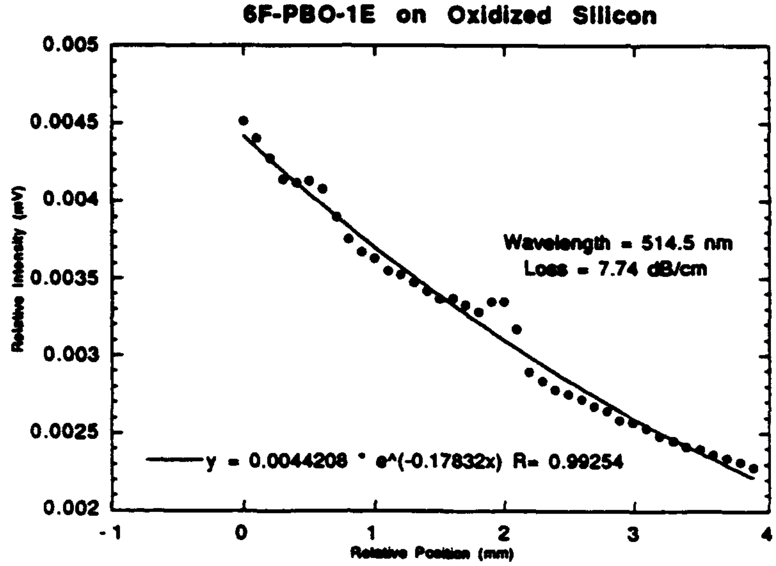


Figure 28 Raw Data for 1.0μm 6F-PBO-1E on 1.4μm SiO_2 on Si at 514.5nm

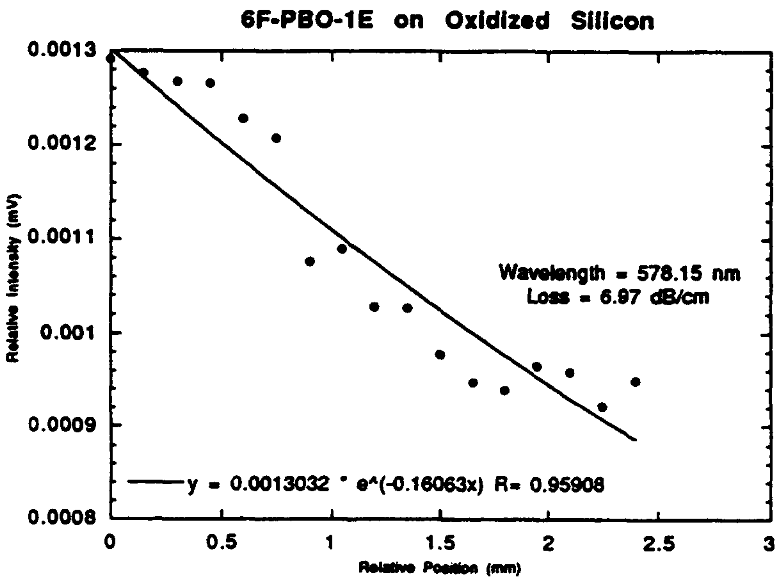


Figure 29 Raw Data for 1.0μm 6F-PBO-1E on 1.4μm SiO_2 on Si at 578.15nm

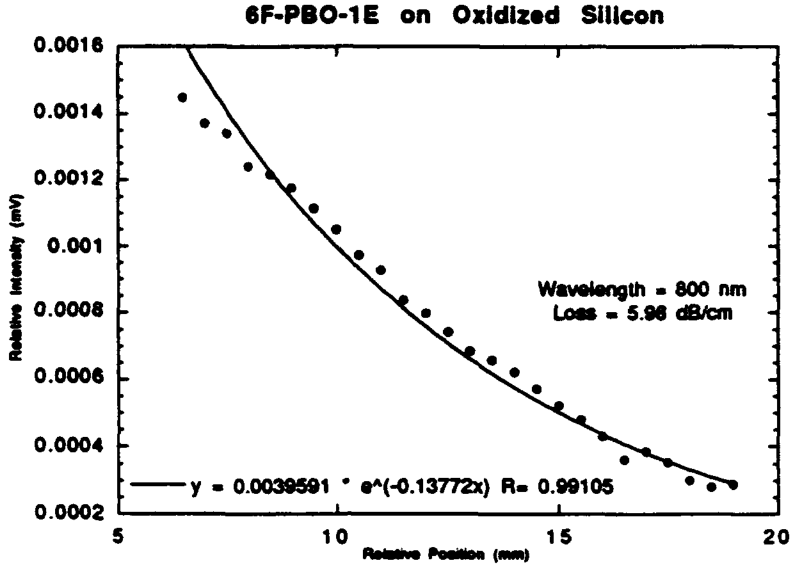


Figure 30 Raw Data for 1.0μm 6F-PBO-1E on 1.4μm SiO_2 on Si at 800nm

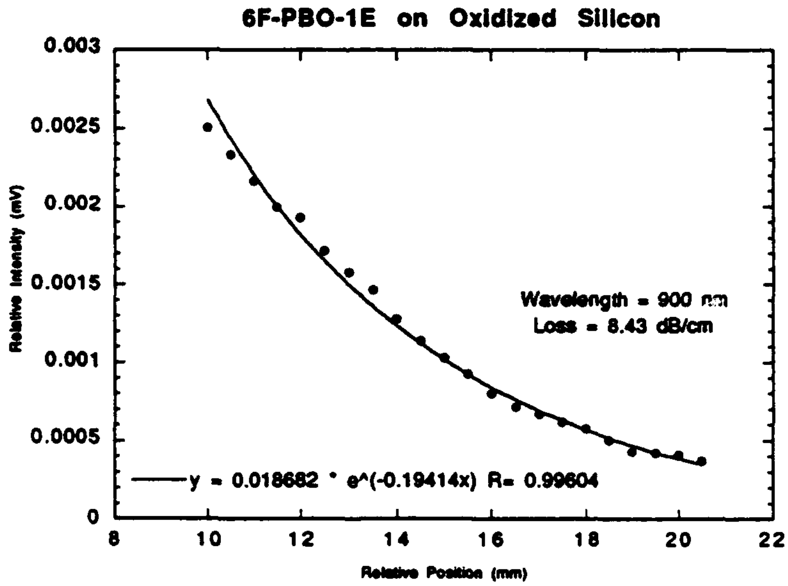


Figure 31 Raw Data for 1.0μm 6F-PBO-1E on 1.4μm SiO_2 on Si at 900nm

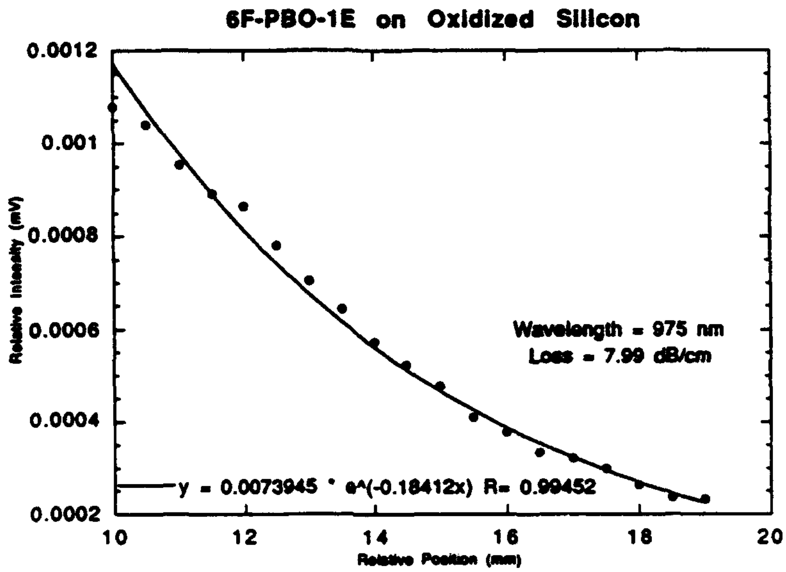


Figure 32 Raw Data for 1.0μm 6F-PBO-1E on 1.4μm SiO_2 on Si at 975nm

5.5. THREE METHODS COMPARISON

Figure 33 shows the comparison on the three methods of loss measurement on the 3-inch wafers. The sample used to obtain this data consists of a 1.0μm film 6F-PBO-1E on 1.5μm oxidized silicon on a 3-inch silicon wafer. The absorption coefficient of the 3-inch wafers are somewhat lower than that of the 2-inch wafers. This is true of all samples tested, probably due to the substrate irregularity as pointed out in Section 4.1.1.

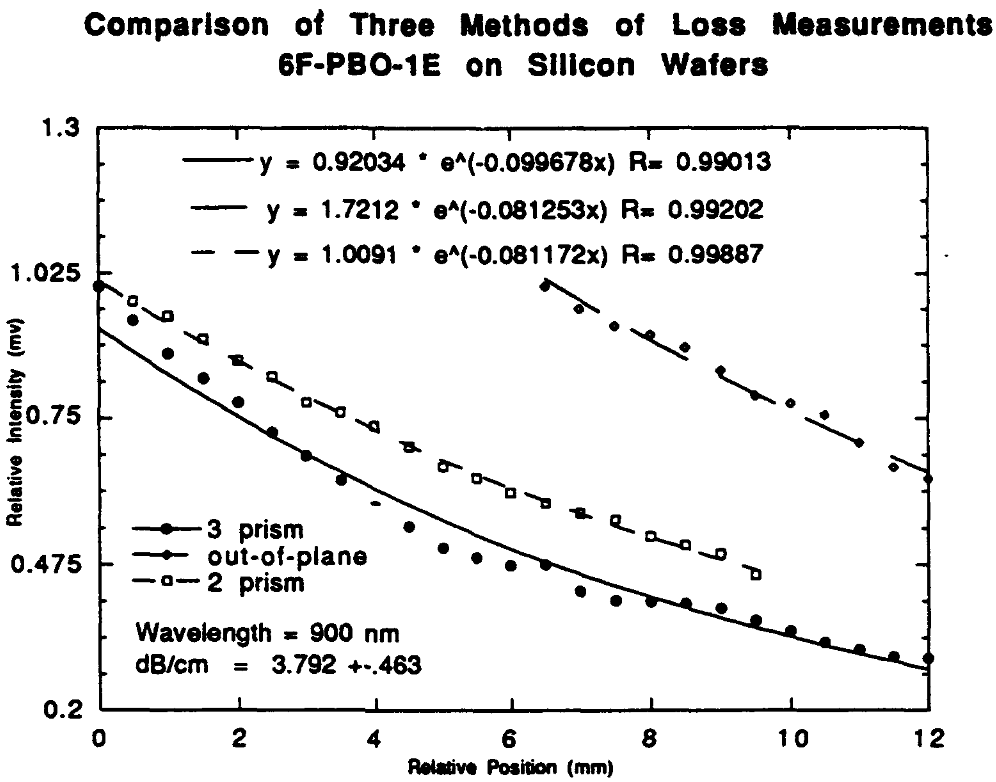


Figure 33 Comparison of Three Methods of Loss Measurement on 1.0μm 6F-PBO-1E on 1.4μm SiO_2 on 3" Si Wafer at 900nm

5.6. WAVEGUIDE RESULTS

The average values for optical waveguide attenuation in dB/cm along with their standard deviations are shown in Figure 34 and TABLE 2.

Upon inspection of the experimental data, it is observed that the loss increases at both the long and short wavelength ends of the tested spectrum. The increase at the long wavelength end agrees with what would be expected since, as the wavelength increases, the confinement decreases which results in more coupling into the buffer and substrate layers and more energy lost to the system. The high losses at shorter wavelengths may be due to scattering and a significant increase in absorption. Scattering losses are expected to increase as the wavelength shortens due to the fact that the wavelength is approaching the particle size. In addition, the material itself is more absorbing at the lower wavelengths. This agrees with what was determined experimentally.

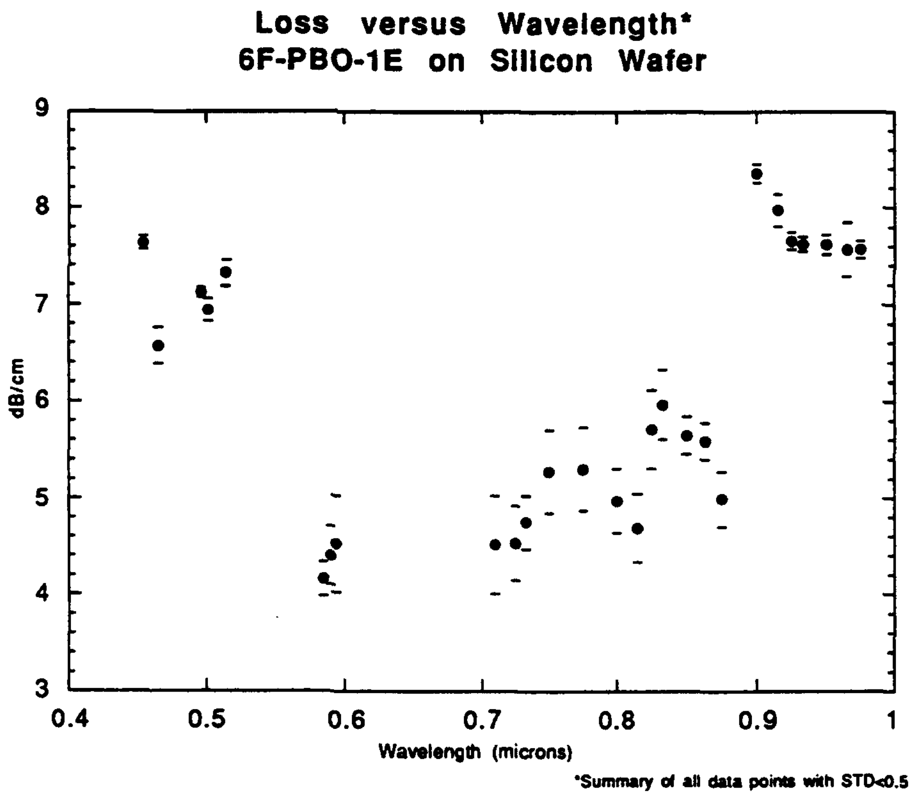


Figure 34 Summary of Loss Data for 6F-PBO-1E at 0.45 - 0.975μm Including Out-of-Plane, Three-Prism and Two-Prism Methods

	2" Samples. All methods		2" Samples 2 Prism Method		3" Samples. All methods		3" Samples 3 Prism Method	
Wavelength (nm)	Loss (dB/cm)	STD (dB/cm)	Loss (dB/cm)	STD (dB/cm)	Loss (dB/cm)	STD (dB/cm)	Loss (dB/cm)	STD (dB/cm)
457.9	0.61	0.634	—	—	1.13	0.466	—	—
488.0	5.40	2.205	—	—	0.84	—	—	—
501.7	6.94	0.113	—	—	—	—	—	—
514.5	7.32	0.136	—	—	1.27	—	—	—
725.0	4.19	1.137	—	—	6.89	7.265	—	—
750.0	4.60	1.476	—	—	2.17	—	—	—
775.0	5.30	0.430	—	—	—	—	—	—
800.0	4.97	0.331	—	—	—	—	—	—
825.0	5.71	0.404	—	—	—	—	—	—
850.0	5.71	2.092	5.53	0.079	6.62	2.298	—	—
875.0	7.66	0.345	4.86	0.121	6.54	1.588	—	—
900.0	6.24	2.456	4.12	0.399	5.47	1.646	4.38	0.439
925.0	7.57	0.091	—	—	5.18	1.634	4.84	0.101
950.0	7.62	0.098	—	—	5.35	1.460	4.80	0.252
975.0	7.57	0.298	—	—	6.30	2.187	3.79	0.851

TABLE 2 Summary of Average Loss Data for 6F-PBO-1E

6.0. CONCLUSION

A modeling package that can be used in conjunction with material fabrications to evaluate possible new optical waveguide materials has been discussed. In addition, the polymer, 6F-PBO-1E, has been investigated to ascertain its eligibility as a waveguiding material. The subsequent results of the characterization, modeling, and experimentation of 6F-PBO-1E after fabrication into waveguide form have been reported.

The polymer 6F-PBO-1E has been characterized to determine its index of refraction, absorption, and optical waveguide attenuation. The measured index of 1.691, along with its low absorption at longer wavelengths, makes 6F-PBO-1E attractive as an optical waveguide material. However, with losses ranging from 4 to 9 dB/cm over the visible wavelengths, while it may still be useful in other applications, this material does not meet the Air Force (e.g. WL/ELOT, RL/ERO) loss guideline of 1 dB/cm or less for optical applications. We conclude that this material, 6F-PBO-1E, in its present form, does not warrant further investigation as an Air Force applicable optical waveguide material.

Current work with the hexafluoro-isopropylidene-polybenzoxazole polymers, however, involves the investigation of variations on the 6F-PBO-1E, via changing the end-capping material, as a viable candidate for high temperature second-order nonlinear optical host material.

7.0. REFERENCES

(1) Yariv, Amnon. 1985. *Optical Electronics* (Third Edition). New York, CBS College Publishing.

(2) Borland, William C. 1989. "Modeling Optical Waveguide Dispersion: Four-Layer Planar, Channel, and Quantum Well Structures." Ph. D. Dissertation. University of Cincinnati Department of Electrical and Computer Engineering.

(3) Zelmon, David E., William C. Borland, Howard E. Jackson, Joseph T. Boyd. 1986. "Near Cutoff Propagation in Low Loss Optical Waveguides Formed On SiO_2/Si Substrate." in SPIE Vol 704 Integrated Optical Circuit Engineering IV.

(4) Born, Max and Emil Wolf. *Principles of Optics., First Edition* (Pergamon Press, New York, 1980).

(5) Kubata, T. and R. Nakanishi, J. Polymer Sci., Part B **2**, 655 (1964).

(6) Imai, Y., I. Taoka, and Y. Iwakura, Makromolekulare Chem. **83**, 167 (1965).

(7) Moyer, W. W., Jr., C. Cole, and T. Anyos, J Polymer Sci., Part A 3, 2107 (1965).

(8) Hergenrother, P. M., W. Wrasidlo, and H. H. Levine, Am. Chem. Soc., Div. of Polym. Chem. Preprints **5** (1), 153 (1964).

(9) Imai, Y., K. Uno, and Y. Iwakura, Makromolekulare Chem. **83,** 179 (1965).

(10) Imai, Y., M. Kakimoto, and Y. Maruyama, German Offen. DE3,707,125 (1987). Chem. Abstr. **108**, 5681 (1988).

(11) Maruyama, Y., Y. Oishi, M. Kakimoto, and Y Imai, Macromolecules. **21**, 2305 (1988).

(12) Reinhardt, B. A., Poly. Comm. **31**, 453 (1990).

(13) AF Contract F33615-92-C-5907.

(14) Tien, P.K., R. Ulrich, J. of Optical Society of America **60,** 1325 (1970).

(15) Moss, T.S. 1961. *Optical Properties of Semi-Conductors..* p. 11-14. London, Butterworths.

(16) Pankove, Jacques I. 1971. *Optical Processes in Semiconductors*. p. 90-94. Dover Publications, Inc. New York.

(17) Spry, R.J., S.J. Bai. Work in Progress.

(18) Won, Y.H., P.C. Jaussaud, G.H. Chartier. 1980. "Three Prism Loss Measurements of Optical Waveguides." in Appl. Phys. Lett. 37(3). p 269-271.

(19) Kensky, E.T., D.E. Zelmon. "A Fully Automated Stage for Loss Measurements of Optical Waveguides." in Review of Scientific Instruments. Vol 64. No 7 July 1993.

(20) Elliot, David J. 19XX. *Integrated Circuit Fabrication Technology* p 111.

APPENDIX A: BASE CLEAN PROCEDURE[20]

Materials:

- Beaker of the appropriate size
- Ammonium Hydroxide (NH_4OH)
- Deionized water
- Hydrogen Peroxide (30%)
- Thermometer
- Hotplate

1. Carefully mix a 4:1 H_2O:NH_4OH solution in the beaker. Add NH_4OH to the H_2O slowly.

2. Heat mixture to 75°C.

3. Add the same quantity of hydrogen peroxide as NH_4OH to the solution.

4. Slowly dip the wafer in the beaker and allow it to soak in the solution for 15 minutes, maintaining the temperature at 65-75°C.

5. At the end of 15 minutes, put the entire beaker under a running stream of deionized water for 5 minutes.

6. Remove the wafer and blow dry with dry Nitrogen.

APPENDIX B: RAW WAVEGUIDE DATA TABLES

Out-of-plane scattering method, first 3" wafer

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.10	2	457.9	0.0161	0.808
0.10	2	457.9	0.0374	0.980
0.10	2	457.9	0.0245	0.950
0.10	2	472.7	0.0286	0.905
0.10	2	472.7	0.0135	0.892
0.10	2	476.5	0.0112	0.887
0.10	2	488.0	0.0193	0.912
0.10	1	496.5	0.0365	0.883
0.10	1	496.5	0.0336	0.936
0.10	1	514.5	0.0292	0.942
0.10	1	528.7	0.0345	0.941
0.10	2	528.7	0.1912	0.917
0.10	2	702.5	0.1924	0.999
0.20	3	725.0	0.4264	0.912
0.20	3	725.0	0.2204	0.894
0.10	4	692.5	0.0157	0.918
0.10	2	692.5	0.0244	0.869
0.10	3	698.0	0.0428	0.983
0.10	4	698.0	0.0421	0.967
0.10	4	698.0	0.0464	0.962
0.10	4	696.0	0.0425	0.965
0.10	4	696.0	0.0557	0.841
0.10	4	696.0	0.0394	0.923
0.10	3	700.0	0.0883	0.922
0.10	3	700.0	0.0629	0.940
0.10	3	725.0	0.0415	0.948
0.10	3	725.0	0.0398	0.960
0.10	3	725.0	0.0654	0.980
0.10	3	750.0	0.0501	0.981

Out-of-plane scattering method, first 2" wafer

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.50	25	705.0	0.0217	0.993
0.50	25	705.0	0.0225	0.989
0.50	25	725.0	0.0260	0.993

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.25	15	750.0	0.0476	0.997
0.25	15	750.0	0.0409	0.993
0.20	10	718.8	0.0905	0.988
0.20	10	718.8	0.1121	0.992
0.20	10	710.0	0.1166	0.991
0.20	10	725.0	0.1118	0.992
0.20	10	725.0	0.1077	0.994
0.20	10	725.0	0.1111	0.993
0.20	10	733.0	0.1143	0.988
0.20	10	733.0	0.1172	0.987
0.20	10	733.0	0.1150	0.987
0.20	10	750.0	0.1283	0.991
0.25	10	750.0	0.1280	0.992
0.25	10	750.0	0.1376	0.994
0.25	10	765.0	0.1364	0.992
0.25	10	765.0	0.1438	0.987
0.25	10	775.0	0.1327	0.990
0.25	10	775.0	0.1371	0.992
0.25	10	785.0	0.1323	0.990
0.25	10	785.0	0.1296	0.989
0.25	10	800.0	0.1209	0.975
0.25	10	800.0	0.1229	0.969
0.25	10	710.0	0.0937	0.989
0.25	10	725.0	0.0913	0.989
0.25	10	733.0	0.1047	0.981
0.25	10	750.0	0.1215	0.970
0.25	10	765.0	0.1266	0.976
0.25	10	775.0	0.1257	0.979
0.25	10	785.0	0.0805	0.974
0.25	10	710.0	0.1271	0.971
0.20	10	710.0	0.1093	0.993
0.20	10	710.0	0.1053	0.983
0.20	10	710.0	0.1073	0.985
0.20	10	765.0	0.1341	0.992
0.20	10	775.0	0.1300	0.990
0.20	10	785.0	0.1267	0.986
0.20	10	800.0	0.1081	0.986
0.20	10	815.0	0.1070	0.987
0.20	10	815.0	0.1045	0.987
0.20	10	815.0	0.1188	0.962
0.10	8	825.0	0.1314	0.973
0.15	8	825.0	0.1334	0.974
0.15	7	825.0	0.1461	0.980
0.15	8	833.0	0.1444	0.978
0.15	8	833.0	0.1441	0.978
0.15	8	833.0	0.1421	0.975
0.15	8	850.0	0.1325	0.981
0.15	8	850.0	0.1316	0.979
0.15	8	850.0	0.1398	0.982

Out-of-plane scattering method, Second 2" wafer

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.25	10	710.0	0.0878	0.985
0.25	10	710.0	0.0914	0.982
0.15	8	725.0	0.1123	0.989
0.15	8	725.0	0.1150	0.988
0.15	8	733.0	0.1148	0.987
0.15	8	733.0	0.1121	0.986
0.25	10	750.0	0.1048	0.977
0.25	10	765.0	0.1006	0.972
0.25	10	765.0	0.1102	0.971
0.25	10	775.0	0.1062	0.970
0.25	10	775.0	0.1168	0.968
0.25	10	785.0	0.1079	0.972
0.25	10	785.0	0.1109	0.969
0.15	8	800.0	0.1221	0.965
0.15	8	815.0	0.1177	0.968
15.00	8	815.0	0.1162	0.964
0.15	8	800.0	0.1209	0.968
0.15	8	825.0	0.1231	0.965
0.15	8	825.0	0.1239	0.966
0.20	8	833.0	0.1301	0.969
0.25	8	833.0	0.1270	0.973
0.25	8	850.0	0.1374	0.973
0.25	8	850.0	0.1330	0.968
0.10	5	572.3	0.0489	0.892
0.10	5	572.3	0.0491	0.880
0.10	5	572.3	0.0738	0.946
0.10	5	578.2	0.1050	0.964
0.10	5	578.2	0.0951	0.938
0.15	7	578.2	0.0497	0.921
0.15	7	584.7	0.0999	0.963
0.15	7	584.7	0.9061	0.965
0.15	7	584.7	0.0917	0.964
0.15	7	590.0	0.0949	0.966
0.15	7	590.0	0.1006	0.949
0.10	5	590.0	0.1089	0.976
0.10	5	593.9	0.1045	0.942
0.10	5	593.9	0.0927	0.912
0.10	5	593.9	0.1158	0.949
0.10	4	528.7	0.1545	0.993
0.10	4	582.7	0.1456	0.991
0.10	4	514.5	0.1702	0.995
0.10	4	514.5	0.1708	0.995
0.10	4	514.5	0.1651	0.993
0.15	6	501.7	0.1572	0.992
0.15	6	501.7	0.1601	0.992
0.15	6	501.7	0.1624	0.992

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.15	6	496.5	0.1656	0.989
0.15	6	496.5	0.1637	0.988
0.15	6	496.5	0.1631	0.989
0.10	5	488.0	0.1775	0.995
0.10	5	488.0	0.1782	0.996
0.10	5	488.0	0.1794	0.996
0.10	5	476.5	0.1682	0.995
0.10	5	476.5	0.1701	0.996
0.10	5	476.5	0.1640	0.995
0.15	6	472.4	0.1784	0.992
0.15	6	472.4	0.1473	0.991
0.15	6	472.4	0.1612	0.993
0.15	6	472.4	0.1615	0.990
0.15	6	465.8	0.1517	0.990
0.15	6	465.8	0.1469	0.990
0.15	6	465.8	0.1556	0.992
0.10	4	457.9	0.1154	0.973
0.10	4	457.9	0.1086	0.956
0.10	4	457.9	0.0882	0.955
0.10	4	454.5	0.1776	0.992
0.10	4	454.5	0.1746	0.994
0.10	4	454.5	0.1752	0.994
0.10	4	472.7	0.0589	0.951
0.10	3	476.5	0.0629	0.976
0.10	3	476.5	0.0863	0.974
0.10	4	476.5	0.0749	0.952
0.10	4	488.0	0.0851	0.965
0.10	4	488.0	0.0758	0.967
0.10	4	488.0	0.0925	0.965
0.10	4	488.0	0.0817	0.952
0.10	4	528.7	0.0753	0.976
0.10	4	528.7	0.0770	0.989
0.50	20	700.0	0.0767	0.982
0.25	15	700.0	0.1063	0.972
0.25	10	710.0	0.1006	0.995
0.50	20	710.0	0.0962	0.998
0.25	20	710.0	0.1097	0.996
0.50	20	725.0	0.0953	0.995
0.50	20	725.0	0.0979	0.995
0.50	20	725.0	0.0977	0.994
0.50	20	733.0	0.1034	0.991
0.50	20	733.0	0.1014	0.992
0.50	20	733.0	0.1019	0.989
0.50	20	750.0	0.1201	0.968
0.50	20	750.0	0.1162	0.972
0.50	20	750.0	0.1159	0.973
0.50	20	765.0	0.1156	0.978
0.50	20	765.0	0.1140	0.980
0.50	20	765.0	0.1140	0.977

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.50	20	775.0	0.1187	0.974
0.50	20	775.0	0.1150	0.976
0.50	20	775.0	0.1174	0.973
0.50	20	785.0	0.1120	0.973
0.50	20	785.0	0.1144	0.972
0.50	20	785.0	0.1123	0.976
0.50	20	800.0	0.1060	0.968
0.50	20	800.0	0.1074	0.969
0.50	20	800.0	0.1088	0.968
0.50	20	815.0	0.1002	0.957
0.50	20	815.0	0.1001	0.956
0.50	20	815.0	0.1007	0.958
0.50	30	863.0	0.1310	0.997
0.50	30	863.0	0.1247	0.990
0.65	30	875.0	0.1292	0.996
0.50	25	875.0	0.1238	0.995
0.50	25	885.0	0.1189	0.999
0.50	25	885.0	0.1426	0.998
0.50	25	900.0	0.1941	0.996
0.50	25	900.0	0.1909	0.996
0.50	25	915.0	0.1864	0.995
0.50	25	915.0	0.1810	0.995
0.50	25	925.0	0.1778	0.996
0.50	25	925.0	0.1749	0.993
0.50	25	933.0	0.1768	0.995
0.50	25	933.0	0.1743	0.996
0.50	25	950.0	0.1739	0.994
0.50	25	950.0	0.1771	0.995
0.50	25	965.0	0.1698	0.994
0.50	25	965.0	0.1789	0.996
0.50	25	971.7	0.1623	0.994
0.50	25	971.7	0.1791	0.994
0.50	25	975.0	0.1696	0.997
0.50	25	975.0	0.1793	0.988

Out-of-plane scattering method, Second 3" wafer

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.50	45	900.0	0.1031	0.992
0.50	45	900.0	0.0759	0.989
0.50	40	915.0	0.1021	0.992
0.50	40	915.0	0.0716	0.985
0.50	40	915.0	0.0921	0.992
0.50	40	924.5	0.1246	0.994
0.50	40	935.0	0.1408	0.988

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.50	45	935.0	0.0810	0.991
0.50	40	945.0	0.1176	0.997
0.50	40	945.0	0.1017	0.991
0.50	40	945.0	0.1061	0.992
0.50	40	885.0	0.0727	0.995
0.50	40	885.0	0.1099	0.988
0.50	40	875.0	0.1083	0.991
0.50	40	875.0	0.1289	0.985
0.50	45	850.0	0.1832	0.986
0.50	45	850.0	0.1826	0.990
0.50	40	850.0	0.1950	0.990
0.50	40	850.0	0.1836	0.991
0.50	40	850.0	0.1512	0.997
0.50	40	850.0	0.1280	0.995
0.50	40	850.0	0.0448	0.992
0.50	40	875.0	0.1571	0.984
0.50	40	875.0	0.1844	0.997
0.50	40	875.0	0.1113	0.990
0.50	40	875.0	0.1606	0.987
0.50	40	875.0	0.2049	0.967
0.50	40	900.0	0.1815	0.997
0.50	40	900.0	0.1407	0.993
0.50	40	900.0	0.1688	0.986
0.50	40	900.0	0.1623	0.983
0.50	40	925.0	0.0993	0.985
0.50	40	925.0	0.1846	0.980
0.50	40	925.0	0.0897	0.993
0.50	40	950.0	0.1860	0.994
0.60	35	950.0	0.1521	0.997
0.60	35	950.0	0.0905	0.995
0.60	35	950.0	0.1031	0.989
0.60	35	975.0	0.1052	0.995
0.60	35	975.0	0.0863	0.997
0.60	35	975.0	0.1147	0.993
0.50	30	975.0	0.2155	0.995

Three-prism method, Second 3" wafer

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.50	20	975.0	0.0735	0.989
0.50	17	975.0	0.1012	0.998
0.50	17	950.0	0.1062	0.998
0.50	17	950.0	0.1082	0.999
0.50	17	950.0	0.1171	0.999
0.50	17	925.0	0.1099	0.997

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.50	17	925.0	0.1132	0.997
0.50	15	900.0	0.1116	0.994
0.50	15	900.0	0.0997	0.990
0.50	15	900.0	0.0915	0.998

Two-prism method, Third 2" wafer

Increment (mm)	Total Distance (mm)	Wavelength (nm)	-Alpha	R-value
0.50	12	850.0	0.1251	0.995
0.50	12	850.0	0.1285	0.988
0.50	12	850.0	0.1267	0.992
0.50	12	850.0	0.1291	0.989
0.50	12	875.0	0.1098	0.996
0.50	12	875.0	0.1153	0.995
0.50	12	875.0	0.1097	0.995
0.50	12	875.0	0.1135	0.991
0.50	12	900.0	0.0885	0.996
0.50	12	900.0	0.1015	0.999

APPENDIX C: PlanarWaveguide.m

DOCUMENTATION

To run the program, open up a *Mathematica* shell and type on the first line:

<<["PlanarWaveguides`"]

This will open the package PlanarWaveguides.m and evaluate the appropriate initialization cells. Then type in the commands necessary as outlined below. When done entering the specifications, go to the right-hand-side of the shell and click on the outline. When the subshell is outlined, hit the <enter> key (not the <return>). A box will probably appear asking you to locate the PlanarWaveguides.m package so you must locate the package and then it will run to completion.

The following are input guidlines for each subroutine:

PropagationConstant[n1,n2,n3,n4,cladding,guide,wavelength] finds all roots for the parameters entered and plots the waveform.

N1Attenuation[{start,end,increment},n2,n3,n4,cladding,guide,wavelength] calculates attenuation (in dB/cm) for the range n1=start, n1<end, n1+=increment and plots the results. The input must be entered in the outlined format.

N2Attenuation[n1_,{start,end,increment},n3,n4,cladding,guide,wavelength] calculates attenuation (in dB/cm) for the range n2=start, n2<end, n2+=increment and plots the results. The input must be entered in the outlined format.

N3Attenuation[n1,n2,{start,end,increment},n4,cladding,guide,wavelength] calculates attenuation (in dB/cm) for the range n3=start, n3<end, n3+=increment and plots the results. The input must be entered in the outlined format.

N4Attenuation[n1,n2,n3,{start,end,increment},cladding,guide,wavelength] calculates attenuation (in dB/cm) for the range n4=start, n4<end, n4+=increment and plots the results. The input must be entered in the outlined format.

CladdingAttenuation[n1,n2,n3,n4,{start,end,increment},guide,wavelength] calculates attenuation (in dB/cm) for the range cladding=start, cladding<end, cladding+=increment and plots the results. The input must be entered in the outlined format.

GuideAttenuation[n1,n2,n3,n4,cladding,{start,end,increment},wavelength] calculates attenuation (in dB/cm) for the range guide=start, guide<end, guide+=increment and plots the results. The input must be entered in the outlined format.

WavelengthAttenuation[n1,n2,n3,n4,cladding,guide,{start,end,increment}] calculates attenuation (in dB/cm) for the range wavelength=start, wavelength<end, wavelength+=increment and plots the results. The input must be entered in the outlined format.

CutoffThickness[{start,end,increment},n1,n3,wavelength] finds the cutoff thickness for the parameters entered. The input must be entered in the outlined format.

CutoffWavelength[{start,end,increment},n1,n3,guide] finds the cutoff wavelength for the parameters entered. The input must be entered in the outlined format.

SOURCE CODE

```
(*    :Title:   PlanarWaveguides  *)

(*    :Author:   Angela  McPherson  *)

(*    :Summary:   There  are  four  sections:
      1. PropagationConstant  generates  plots  of  the  waveform
      for  all  modes  found  for  the  given  parameters.
      2.  Attenuation
            a. N1Attenuation  generates  values  of  attenuation
            for  a  range  of  n1  values  and  plots  the  output.
            b.  N2Attenuation  (same  format  as  above)
            c.  N3Attenuation
            d.  N4Attenuation
            e.  CladdingAttenuation
            f.  GuideAttenuation
            g.  WavelengthAttenuation
      3. CutoffThickness  generates  the  cutoff  values  for  the
      guiding  layer  thickness  below  which  there  will  be  no
      confined  modes.
      4.  CutoffWavelength  (same  format  as  above).
*)

(*    :Context:   PlanarWaveguides`  *)

(*    :Package  Version:  2.1  *)

(*    :Keywords:   waveguide,attenuation,propagation   constant,
     mode *)

(*    :Source:
        Borland,  William  C.  1989.  "Modeling  Optical  Waveguide  Dispersion:
          Four-Layer  Planar,  Channel,  and  Quantum  Well  Structures."  Ph.  D.
          Dissertation.  Univeristy  of  Cincinnati  Department  of  Electrical  and
          Computer  Engineering.
        Yariv,  Amnon.  1985.  Optical  Electronics  (Third  Edition).  New  York,  CBS
          College  Publishing.
         Zelmon,  David  E.,  William  C.  Borland,  Howard  E.  Jackson,  Joseph  T.  Boyd.
          1986.  "Near  Cutoff  Propagation  in  Low  Loss  Optical  Waveguides  Formed
          On  SiO2/Si  Substrate."  in  SPIE  Vol  704  Integrated  Optical  Circuit
          Engineering  IV.
```

```
*)

(*   :Mathematica Version: 2.2 *)

(* :Limitation: The length of time involved depends on the
    number of iterations asked for and how quickly the solutions
    converge.  On average, the calculation takes several minutes.
*)

(*    :Discussion:  Before using this package, see the ::usage
     information for details on the input format.  In places
     where there is a range, that is the range of the parameter
     you wish to vary (as stated in the procedure name.)
*)


BeginPackage["PlanarWaveguides`"]


FindRoots::usage="\nFindRoots[increment2_,threeLayerEquation[variable]_,fo
urLayerEquation[variable]_,k_,n2_,n3_,listCount_,flag_,list_] is the core of the
code that scans the range of possible roots to find all confined mode
parameters"

PlotThis::usage="\nPlotThis[list_,plotLabel_,biggestListCount_] is the plotting
routine for the attenuation plots"

CheckInput::usage="\nCheckInput[n1_,n2_,n3_,n4_,halfGuide_,wavelength_,k
_,start_,end_,increment_] checks to see if the input values satisfy appropriate
restraints.  Start, end and increment are options."

PropagationConstant::usage="\nPropagationConstant[n1_,n2_,n3_,n4_,cladding
_,guide_,wavelength_] finds all roots for the parameters entered and plots the
waveform."

N1Attenuation::usage="\nN1Attenuation[{start_,end_,increment_},n2_,n3_,n4_,
cladding_,guide_,wavelength_] calculates attenuation (in dB/cm) for the
range n1=start, n1<end, n1+=increment and plots the results.  The input must
be entered in the outlined format."

N2Attenuation::usage="\nN2Attenuation[n1_,{start_,end_,increment_},n3_,n4_,
cladding_,guide_,wavelength_] calculates attenuation (in dB/cm) for the
range n2=start, n2<end, n2+=increment and plots the results.  The input must
be entered in the outlined format."

N3Attenuation::usage="\nN3Attenuation[n1_,n2_,{start_,end_,increment_},n4_,
cladding_,guide_,wavelength_] calculates attenuation (in dB/cm) for the
range n3=start, n3<end, n3+=increment and plots the results.  The input must
be entered in the outlined format."
```

```
N4Attenuation::usage="\nN4Attenuation[n1_,n2_,n3_,{start_,end_,increment_},
cladding_,guide_,wavelength_] calculates attenuation (in dB/cm) for the
range n4=start, n4<end, n4+=increment and plots the results. The input must
be entered in the outlined format."

CladdingAttenuation::usage="\nCladdingAttenuation[n1_,n2_,n3_,n4_,{start_,e
nd_,increment_},guide_,wavelength_] calculates attenuation (in dB/cm) for
the range cladding=start, cladding<end, cladding+=increment and plots the
results. The input must be entered in the outlined format."

GuideAttenuation::usage="\nGuideAttenuation[n1_,n2_,n3_,n4_,cladding_,{star
t_,end_,increment_},wavelength_] calculates attenuation (in dB/cm) for the
range guide=start, guide<end, guide+=increment and plots the results. The
input must be entered in the outlined format."

WavelengthAttenuation::usage="\nWavelengthAttenuation[n1_,n2_,n3_,n4_,cl
adding_,guide_,{start_,end_,increment_}] calculates attenuation (in dB/cm) for
the range wavelength=start, wavelength<end, wavelength+=increment and
plots the results. The input must be entered in the outlined format."

CutoffThickness::usage="\nCutoffThickness[{start_,end_,increment_},n1_,n3_,
wavelength__] finds the cutoff thickness for the parameters entered. The
input must be entered in the outlined format."

CutoffWavelength::usage="\nCutoffWavelength[{start_,end_,increment_},n1_,
n3_,guide_] finds the cutoff wavelength for the parameters entered. The input
must be entered in the outlined format."


Off[Greater::nord];
Off[Less::nord];
Off[General::spell];
Off[General::spell1];
Off[ReplaceAll::reps];


Begin["`private`"]


CheckInput[n1_,n2_,n3_,n4_,halfGuide_,wavelength_,k_,start_:0,end_:0,incre
ment_:0] :=
Block[{cutoffWave,cutoffGuide},
 cutoffWave = .;cutoffGuide = .;
  If[(n1<1)||(n2<1)||(n3<1)||(Re[n4]<1),
   Print["One of the indices of refraction entered is less than 1."];
   Break[];
 ];
  If[(((Re[start]>Re[end])&&(Re[increment]>=0))||
  ((Im[start]>Im[end])&&(Im[increment]>=0))),
   Print["The input range values were entered incorrectly"];
  Break[];
```

```
];
 If[((n1>=n2)||(n3>=n2)),
  Print["The basic requirement for waveguiding (n1<n2>n3) was not satisfied"];
  Break[];
 ];
 cutoffGuide = ArcTan[Sqrt[(n3^2 - n1^2)/(n2^2 - n3^2)]]/(2 k Sqrt[n2^2 -
n3^2]);
 If[(cutoffGuide > halfGuide),
  Print["The guiding thickness entered is below cutoff"];
  Break[];
 ];
 cutoffWave = 4 halfGuide N[Pi] Sqrt[n2^2 - n3^2]/ArcTan[Sqrt[n3^2-
n1^2]/Sqrt[n2^2-n3^2]];
 If[(cutoffWave < wavelength),
  Print["The wavelength entered is below the cutoff wavelength"];
  Break[];
 ];
];



FindRoots[increment2_,variable_,threeLayerEquation_,fourLayerEquation_,k_
,n2_,n3_,flag_] :=
Block[{rule1,rule2,test,newList},
 newList = Table[0,{x,0,1},{y,0,1}];
 For[test=k n3,test<k n2,test+=increment2,
 rule1 = .;rule2 = .;
  condition1 = threeLayerEquation/.B->test;
  condition2 = threeLayerEquation/.B->(test-increment2);
 If[(condition1>0 && condition2<0) ||
  (condition1<0 && condition2>0),
    rule1 = FindRoot[threeLayerEquation == 0,
    {B,test},AccuracyGoal -> 6];
    If[NumberQ[B/.rule1],
    Check[rule2 =
      FindRoot[fourLayerEquation == 0,
       {B,B/.rule1},MaxIterations -> 20,
       AccuracyGoal -> 6],Continue[]];
      If[(((Re[B/.rule1]>k n3)&&(Re[B/.rule2]>k n3))&&
       ((Re[B/.rule1]<k n2)&&(Re[B/.rule2]<k n2))&&
       (Im[B/.rule2]<0))||((Re[B/.rule2]>k n3)&&
       (Re[B/.rule2]<k n2)&&(Im[B/.rule2]<0)),
     If[flag==1,
         newList = Union[newList,{{2 variable,4.34 Im[B/.rule2] 10^(-4)}}],
         newList = Union[newList,{{variable,4.34 Im[B/.rule2] 10^(-4)}}];
    ];
   ];
  ];
 ];
   If[(Re[B/.rule2]>test),test = Re[B/.rule2]];
 ];
```

```
If[Length[Complement[newList,{{0,0}}]]==0,
 Print["There are no guided modes for the four layer guide at ",variable];
];
 newList = Complement[newList,{{0,0}}];


 For[loop=2,loop<=Length[newList],loop++,
  If[Abs[Re[newList[[loop,2]]-newList[[loop-1,2]]]]<=.001,
  newList = Delete[newList,loop];
 ];
];
 For[loop=1,loop<=Length[newList],loop++,
  Print[newList[[loop]]];
];
 Return[newList];
];



PlotThis[list_,plotLabel_,biggestListCount_] :=
 Block[{newList,modeFlags,i,j,list2},
  newList = Complement[list,{{0,0}}];
  modeFlags = Table[0,{x,0,Length[newList]}];
  For[j=1,j<=biggestListCount,j++,
 i=j;
   While[(i<=Length[newList]),
   While[(newList[[i,1]] == newList[[(i+1),1]]),
  i++;
 ];
   If[(modeFlags[[i]]!=0)&&((newList[[(i-j+1),1]]==
    newList[[(i-j+2),1]])||(newList[[(i-j+3),1]]==
    newList[[(i-j+2),1]]))&&(modeFlags[[i-j+1]]==0),
   modeFlags[[i-j+1]]=j;
 ];
   If[modeFlags[[i]]==0,modeFlags[[i]]=j];
 i++;
 ];
];
 For[j=1,j<=biggestListCount,j++,
 list2 = Table[0,{x,0,1},{y,0,1}];
  For[i=1,i<=Length[newList],i++,
  If[modeFlags[[i]]==j,
   list2 = Union[list2,{{Re[newList[[i,1]]],newList[[i,2]]}}];
 ];
 ];
 list2 = Complement[list2,{{0,0}}];
 Print[list2];
 ListPlot[list2,PlotStyle -> PointSize[.02],
 PlotRange -> All,PlotLabel -> plotLabel];
];
```

```
];

PropagationConstant[n1_,n2_,n3_,n4_,cladding_,guide_,wavelength_]  :=
Block[{q,h,p,s,equation3,equation4,k,B,increment,list},
halfGuide = .5 guide;
k = 2*N[Pi]/wavelength;
q = Sqrt[B^2 - k^2 n1^2];h = Sqrt[k^2 n2^2 - B^2];
p = Sqrt[B^2 - k^2 n3^2];s = Sqrt[k^2 n4^2 - B^2];
equation3 = Tan[2 halfGuide h] - h (p + q)/(h^2 - p q);
equation4 = ((p+I s) ((h^2-p q) Tan[2 halfGuide h]-
h (p+q))+Exp[-2 p cladding] (p-I s) ((h^2+p q)*
Tan[2 halfGuide h]+h (p-q)));
increment = (k n2 - k n3)/1000;
list = Table[0,{x,0,1}];
CheckInput[n1,n2,n3,n4,halfGuide,wavelength,k];
If[increment==0,increment = 1000];
For[test=k n3,test<k n2,test+=increment,
 rule1 = .;rule2 = .;
  condition1 = equation3/.B->test;
  condition2 = equation3/.B->(test-increment);
 If[(condition1>0 && condition2<0) ||
  (condition1<0 && condition2>0),
    rule1 = FindRoot[equation3 == 0,{B,test},
    AccuracyGoal -> 6];
     If[NumberQ[B/.rule1],
      Check[rule2 = FindRoot[equation4 == 0,{B,B/.rule1},
      MaxIterations -> 20, AccuracyGoal -> 10],Continue[]];
       If[(((Re[B/.rule1]>k n3)&&(Re[B/.rule2]>k n3))&&
        ((Re[B/.rule1]<k n2)&&(Re[B/.rule2]<k n2))&&
        (Im[B/.rule2]<0))||((Re[B/.rule2]>k n3)&&
        (Re[B/.rule2]<k n2)&&(Im[B/.rule2]<0)),
        list = Union[list,{B/.rule2}];
   ];
      If[(Re[B/.rule2]>test),test = Re[B/.rule2]];
   ];
  ];
];
list = Complement[list,{0,B}];
For[loop=2,loop<=Length[list],loop++,
 If[Abs[Re[list[[loop]]-list[[loop-1]]]]<=.001,
  list = Delete[list,loop]]];
For[loop=1,loop<=Length[list],loop++,Print[list[[loop]]]];
Print["Begin plotting routine....."];
For[i=1,i<=Length[list],i++,
 B = list[[i]];
 q = Sqrt[B^2 - k^2 n1^2];h = Sqrt[k^2 n2^2 - B^2];
 p = Sqrt[B^2 - k^2 n3^2];s = Sqrt[k^2 n4^2 - B^2];
 MM = 1;

  CC = MM*(-(E^(cladding*p)*(-Sin[-(h*halfGuide)]^(-1) -
```

```
(Cos[-(h*halfGuide)]*(-(p/(-((h*Cos[-(h*halfGuide)]^2)/
Sin[-(h*halfGuide)]) - h*Sin[-(h*halfGuide)])) +
(h*Cos[-(h*halfGuide)])/(Sin[-(h*halfGuide)]*
(-((h*Cos[-(h*halfGuide)]^2)/Sin[-(h*halfGuide)]) -
h*Sin[-(h*halfGuide)]))))/Sin[-(h*halfGuide)])) +
(-1/(2*E^(cladding*p)) + (I/2*s)/(E^(cladding*p)*p))*
(-Sin[-(h*halfGuide)]^(-1) -
(Cos[-(h*halfGuide)]*(p/(-((h*Cos[-(h*halfGuide)]^2)/
Sin[-(h*halfGuide)]) - h*Sin[-(h*halfGuide)]) +
(h*Cos[-(h*halfGuide)])/(Sin[-(h*halfGuide)]*
(-((h*Cos[-(h*halfGuide)]^2)/Sin[-(h*halfGuide)]) -
h*Sin[-(h*halfGuide)]))))/Sin[-(h*halfGuide)] -
E^(2*cladding*p)*(-Sin[-(h*halfGuide)]^(-1) -
(Cos[-(h*halfGuide)]*(-(p/(-((h*Cos[-(h*halfGuide)]^2)/
Sin[-(h*halfGuide)]) - h*Sin[-(h*halfGuide)])) +
(h*Cos[-(h*halfGuide)])/(Sin[-(h*halfGuide)]*
(-((h*Cos[-(h*halfGuide)]^2)/Sin[-(h*halfGuide)]) -
h*Sin[-(h*halfGuide)]))))/Sin[-(h*halfGuide)])));

DD = (-(E^(cladding*p)*(-(p/(-((h*Cos[-(h*halfGuide)]^2)/
Sin[-(h*halfGuide)]) - h*Sin[-(h*halfGuide)])) +
(h*Cos[-(h*halfGuide)])/(Sin[-(h*halfGuide)]*
(-((h*Cos[-(h*halfGuide)]^2)/Sin[-(h*halfGuide)]) -
h*Sin[-(h*halfGuide)])))) + (-1/(2*E^(cladding*p)) +
(I/2*s)/(E^(cladding*p)*p))*(p/
(-((h*Cos[-(h*halfGuide)]^2)/Sin[-(h*halfGuide)]) -
h*Sin[-(h*halfGuide)]) + (h*Cos[-(h*halfGuide)])/
(Sin[-(h*halfGuide)]*(-((h*Cos[-(h*halfGuide)]^2)/
Sin[-(h*halfGuide)]) - h*Sin[-(h*halfGuide)])) -
E^(2*cladding*p)*(-(p/(-((h*Cos[-(h*halfGuide)]^2)/
Sin[-(h*halfGuide)]) - h*Sin[-(h*halfGuide)])) +
(h*Cos[-(h*halfGuide)])/(Sin[-(h*halfGuide)]*
(-((h*Cos[-(h*halfGuide)]^2)/Sin[-(h*halfGuide)]) -
h*Sin[-(h*halfGuide)])))));

FF = MM*(E^(cladding*p) + E^(2*cladding*p)*
(-1/(2*E^(cladding*p)) + (I/2*s)/(E^(cladding*p)*p)));

GG = MM*(1/(2*E^(cladding*p)) +
(-I/2*s)/(E^(cladding*p)*p));

BB = CC Sin[h halfGuide] + DD Cos[h halfGuide];

plot1 = Plot[Re[BB Exp[-q (x - halfGuide)]],
{x,halfGuide,2 halfGuide},DisplayFunction -> Identity];

plot2 = Plot[Re[CC Sin[h x] + DD Cos[h x]],
{x, -halfGuide, halfGuide},DisplayFunction -> Identity];

plot3 = Plot[Re[FF Exp[p (x + halfGuide)] +
GG Exp[-p (x + halfGuide)]],{x, -(cladding + halfGuide),
```

```
-halfGuide},DisplayFunction -> Identity];

plot4 = Plot[Re[MM Exp[I s (x + cladding + halfGuide)]],
{x, -1.5 (cladding + halfGuide), -(cladding + halfGuide)},
DisplayFunction -> Identity];

Show[plot4,plot3,plot2,plot1, PlotRange -> All,
PlotLabel -> "Wave Function",
DisplayFunction -> $DisplayFunction,
Ticks -> {{halfGuide,-halfGuide,-(cladding + halfGuide)},
Automatic}];
];
] /;
NumberQ[N[n1]]&&NumberQ[N[n2]]&&NumberQ[N[n3]]&&NumberQ[N[n4]]&&Nu
mberQ[N[cladding]]&&NumberQ[N[guide]]&&NumberQ[N[wavelength]]


N1Attenuation[{start_,end_,increment_},n2_,n3_,n4_,cladding_,guide_,wavele
ngth_] :=
Block[{list,increment2,variable,listCount,biggestListCount,
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,
threeLayerEquation,argument},
 biggestListCount = 0;
 flag = 0;
 list = Table[0,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
 q[argument_] := Sqrt[B^2 - k^2 (argument)^2];
 h = Sqrt[k^2 n2^2 - B^2];
 p = Sqrt[B^2 - k^2 n3^2];
 s = Sqrt[k^2 n4^2 - B^2];
 k = 2 N[Pi]/wavelength;
 fourLayerEquation[argument_] := ((p + I s) ((h^2 - p q[argument]) Tan[2
halfGuide h] - h (p + q[argument])) + Exp[-2 p cladding] (p - I s) ((h^2 + p
q[argument]) Tan[2 halfGuide h] + h (p - q[argument])));
 threeLayerEquation[argument_] := Tan[2 halfGuide h] - h (p +
q[argument])/(h^2 - p q[argument]);
   CheckInput[start,n2,n3,n4,halfGuide,wavelength,k,start,end,increment];
   CheckInput[end,n2,n3,n4,halfGuide,wavelength,k,start,end,increment];
  If[increment==0,increment = 1000];
  plotLabel = "Attenuation (dB/cm) vs n1";
  increment2 = (k n2 - k n3)/10;
   For[variable=start,variable<=end,variable+=increment,
  list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
    If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
     biggestListCount=Length[Complement[list2,{{0,0}}]]];
 ];
 ];
  PlotThis[list,plotLabel,biggestListCount];
```

```
] /;
NumberQ[N[start]]&&NumberQ[N[end]]&&NumberQ[N[increment]]&&NumberQ[
N[n2]]&&NumberQ[N[n3]]&&NumberQ[N[n4]]&&NumberQ[N[cladding]]&&Numb
erQ[N[guide]]&&NumberQ[N[wavelength]]


N2Attenuation[n1_,{start_,end_,increment_},n3_,n4_,cladding_,guide_,wavele
ngth_] :=
Block[{list,increment2,variable,listCount,biggestListCount,
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,threeLayerEquation},
 biggestListCount = 0;
 flag = 0;
 list = Table[0,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
 q = Sqrt[B^2 - k^2 n1^2];
 h[argument_] := Sqrt[k^2 (argument)^2 - B^2];
 p = Sqrt[B^2 - k^2 n3^2];
 s = Sqrt[k^2 n4^2 - B^2];
 k = 2 N[Pi]/wavelength;
 fourLayerEquation[argument_] := ((p + I s) (((h[argument])^2 - p q) Tan[2
halfGuide h[argument]] - h[argument] (p + q)) + Exp[-2 p cladding] (p - I s)
(((h[argument])^2 + p q) Tan[2 halfGuide h[argument]] + h[argument] (p -
q)));
  threeLayerEquation[argument_] := Tan[2 halfGuide h[argument]] -
h[argument] (p + q)/((h[argument])^2 - p q);
   CheckInput[n1,start,n3,n4,halfGuide,wavelength,k,start,end,increment];
   CheckInput[n1,end,n3,n4,halfGuide,wavelength,k,start,end,increment];
  If[increment==0,increment = 1000];
  plotLabel = "Attenuation (dB/cm) vs n2";
   For[variable=start,variable<=end,variable+=increment,
   increment2 = (k variable - k n3)/10;
  list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,variable,n3,flag])];
    If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
     biggestListCount=Length[Complement[list2,{{0,0}}]];
 ];
 ];
  PlotThis[list,plotLabel,biggestListCount];
] /;
NumberQ[N[n1]]&&NumberQ[N[start]]&&NumberQ[N[end]]&&NumberQ[N[incre
ment]]&&NumberQ[N[n3]]&&NumberQ[N[n4]]&&NumberQ[N[cladding]]&&Numb
erQ[N[guide]]&&NumberQ[N[wavelength]]



N2ComplexAttenuation[n1_,{start_,end_,increment_},n3_,n4_,cladding_,guide_,
wavelength_] :=
Block[{list,increment2,variable,listCount,biggestListCount,
```

```
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,threeLayerEquation,imag
inaryVar,realVar},
 biggestListCount = 0;
 flag = 0;
 list = Table[0,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
 q = Sqrt[B^2 - k^2 n1^2];
 h[argument_] := Sqrt[k^2 (argument)^2 - B^2];
 p = Sqrt[B^2 - k^2 n3^2];
 s = Sqrt[k^2 n4^2 - B^2];
 k = 2 N[Pi]/wavelength;
 fourLayerEquation[argument_] := ((p + I s) (((h[argument])^2 - p q) Tan[2
halfGuide h[argument]] - h[argument] (p + q)) + Exp[-2 p cladding] (p - I s)
(((h[argument])^2 + p q) Tan[2 halfGuide h[argument]] + h[argument] (p -
q)));
 threeLayerEquation[argument_] := Tan[2 halfGuide h[argument]] -
h[argument] (p + q)/((h[argument])^2 - p q);
   CheckInput[n1,start,n3,n4,halfGuide,wavelength,k,start,end,increment];
   CheckInput[n1,end,n3,n4,halfGuide,wavelength,k,start,end,increment];
 If[increment==0,increment = 1000];
 plotLabel = "Attenuation (dB/cm) vs Re[n2]";


 increment2 = (k n2 - k n3)/10;


  If[Re[increment]==0,
   increment2 = (k n2 - k n3)/10;

For[imaginaryVar=Abs[Im[start]],imaginaryVar<=Abs[Im[end]],imaginaryVar+
=Abs[Im[increment]],
    variable = Re[start] - imaginaryVar I;
   list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
      If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
       biggestListCount=Length[Complement[list2,{{0,0}}]];
  ];
 ];
    PlotThis[list,plotLabel,biggestListCount];
 ];
  If[Im[increment]==0,

For[realVar=Abs[Re[start]],realVar<=Abs[Re[end]],realVar+=Abs[Re[increment]]
,
    variable = realVar - Abs[Im[start]] I;
    increment2 = (k realVar - k n3)/10;
   list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
      If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
```

```
        biggestListCount=Length[Complement[list2,{{0,0}}]];
   ];
  ];
    PlotThis[list,plotLabel,biggestListCount];
 ];
   If[(Im[increment]!=0)&&(Re[increment]!=0),
      If[(Re[start]-Re[end])/Re[increment]>=(Im[start]-Im[end])/Im[increment],
       For[realVar=Re[start],realVar<=Re[end],realVar+=Re[increment],

For[imaginaryVar=Abs[Im[start]],imaginaryVar<=Abs[Im[end]],imaginaryVar+
=Abs[Im[increment]],
       variable = realVar - imaginaryVar I;
       increment2 = (k realVar - k n3)/10;
    list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
         If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
          biggestListCount=Length[Complement[list2,{{0,0}}]];
     ];
    ];
   ];
  ];
      If[(Re[start]-Re[end])/Re[increment]<(Im[start]-Im[end])/Im[increment],

For[imaginaryVar=Abs[Im[start]],imaginaryVar<=Abs[Im[end]],imaginaryVar+
=Abs[Im[increment]],
         For[realVar=Re[start],realVar<=Re[end],realVar+=Re[increment],
       variable = realVar - imaginaryVar I;
       increment2 = (k realVar - k n3)/10;
    list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
         If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
          biggestListCount=Length[Complement[list2,{{0,0}}]];
     ];
    ];
   ];
  ];
    PlotThis[list,plotLabel,biggestListCount];
 ];
] /;
NumberQ[N[n1]]&&NumberQ[N[n2]]&&NumberQ[N[n3]]&&NumberQ[N[start]]&&
NumberQ[N[end]]&&NumberQ[N[increment]]&&NumberQ[N[cladding]]&&Numb
erQ[N[guide]]&&NumberQ[N[wavelength]]


N3Attenuation[n1_,n2_,{start_,end_,increment_},n4_,cladding_,guide_,wavele
ngth_] :=
Block[{list,increment2,variable,listCount,biggestListCount,
```

```
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,threeLayerEquation},
 biggestListCount = 0;
 flag = 0;
 list = Table[0,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
 q = Sqrt[B^2 - k^2 n1^2];h = Sqrt[k^2 n2^2 - B^2];
 p[argument_] := Sqrt[B^2 - k^2 (argument)^2];
 s = Sqrt[k^2 n4^2 - B^2];
 k = 2 N[Pi]/wavelength;
 fourLayerEquation[argument_] := ((p[argument] + I s) ((h^2 - p[argument] q)
Tan[2 halfGuide h] - h (p[argument] + q)) + Exp[-2 p[argument] cladding]
(p[argument] - I s) ((h^2 + p[argument] q) Tan[2 halfGuide h] + h (p[argument]
- q)));
 threeLayerEquation[argument_] := Tan[2 halfGuide h] - h (p[argument] +
q)/(h^2 - p[argument] q);
   CheckInput[n1,n2,start,n4,halfGuide,wavelength,k,start,end,increment];
   CheckInput[n1,n2,end,n4,halfGuide,wavelength,k,start,end,increment];
  If[increment==0,increment = 1000];
  plotLabel = "Attenuation (dB/cm) vs n3";
   For[variable=start,variable<=end,variable+=increment,
   increment2 = (k n2 - k variable)/10;
  list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,variable,flag])];
    If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
     biggestListCount=Length[Complement[list2,{{0,0}}]];
  ];
 ];
  PlotThis[list,plotLabel,biggestListCount];
] /;
NumberQ[N[n1]]&&NumberQ[N[n2]]&&NumberQ[N[start]]&&NumberQ[N[end]]&
&NumberQ[N[increment]]&&NumberQ[N[n4]]&&NumberQ[N[cladding]]&&Numb
erQ[N[guide]]&&NumberQ[N[wavelength]]


N4Attenuation[n1_,n2_,n3_,{start_,end_,increment_},cladding_,guide_,wavele
ngth_] :=
Block[{list,increment2,variable,listCount,biggestListCount,
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,threeLayerEquation,imag
inaryVar,realVar},
 biggestListCount = 0;
 flag = 0;
 list = Table[0,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
 q = Sqrt[B^2 - k^2 n1^2];h = Sqrt[k^2 n2^2 - B^2];
 p = Sqrt[B^2 - k^2 n3^2];
 s[argument_] := Sqrt[k^2 (argument)^2 - B^2];
 k = 2 N[Pi]/wavelength;
 fourLayerEquation[argument_] := ((p + I s[argument]) ((h^2 - p q) Tan[2
halfGuide h] - h (p + q)) + Exp[-2 p cladding] (p - I s[argument]) ((h^2 + p q)
Tan[2 halfGuide h] + h (p - q)));
```

```
threeLayerEquation[argument_] := Tan[2 halfGuide h] - h (p + q)/(h^2 - p q);
  CheckInput[n1,n2,n3,start,halfGuide,wavelength,k,start,end,increment];
  CheckInput[n1,n2,n3,end,halfGuide,wavelength,k,start,end,increment];
 If[increment==0,increment = 1000];
 plotLabel = "Attenuation (dB/cm) vs Re[n4]";
 increment2 = (k n2 - k n3)/10;
 If[Re[increment]==0,

For[imaginaryVar=Abs[Im[start]],imaginaryVar<=Abs[Im[end]],imaginaryVar+
=Abs[Im[increment]],
    variable = Re[start] - imaginaryVar I;
  list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
      If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
       biggestListCount=Length[Complement[list2,{{0,0}}]];
  ];
 ];
    PlotThis[list,plotLabel,biggestListCount];
];
 If[Im[increment]==0,

For[realVar=Abs[Re[start]],realVar<=Abs[Re[end]],realVar+=Abs[Re[increment]]
,
    variable = realVar - Abs[Im[start]] I;
  list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
      If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
       biggestListCount=Length[Complement[list2,{{0,0}}]];
  ];
 ];
    PlotThis[list,plotLabel,biggestListCount];
];
  If[(Im[increment]!=0)&&(Re[increment]!=0),
     If[(Re[start]-Re[end])/Re[increment]>=(Im[start]-Im[end])/Im[increment],
      For[realVar=Re[start],realVar<=Re[end],realVar+=Re[increment],

For[imaginaryVar=Abs[Im[start]],imaginaryVar<=Abs[Im[end]],imaginaryVar+
=Abs[Im[increment]],
      variable = realVar - imaginaryVar I;
   list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
        If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
         biggestListCount=Length[Complement[list2,{{0,0}}]];
   ];
  ];
 ];
];
     If[(Re[start]-Re[end])/Re[increment]<(Im[start]-Im[end])/Im[increment],
```

```
For[imaginaryVar=Abs[Im[start]],imaginaryVar<=Abs[Im[end]],imaginaryVar+
=Abs[Im[increment]],
     For[realVar=Re[start],realVar<=Re[end],realVar+=Re[increment],
    variable = realVar - imaginaryVar I;
  list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
     If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
      biggestListCount=Length[Complement[list2,{{0,0}}]];
  ];
  ];
 ];
 ];
  PlotThis[list,plotLabel,biggestListCount];
];
] /;
NumberQ[N[n1]]&&NumberQ[N[n2]]&&NumberQ[N[n3]]&&NumberQ[N[start]]&&
NumberQ[N[end]]&&NumberQ[N[increment]]&&NumberQ[N[cladding]]&&Numb
erQ[N[guide]]&&NumberQ[N[wavelength]]


CladdingAttenuation[n1_,n2_,n3_,n4_,{start_,end_,increment_},guide_,wavele
ngth_] :=
Block[{list,increment2,variable,listCount,biggestListCount,
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,threeLayerEquation},
 biggestListCount = 0;
 flag = 0;
 list = Table[0,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
 q = Sqrt[B^2 - k^2 n1^2];h = Sqrt[k^2 n2^2 - B^2];
 p = Sqrt[B^2 - k^2 n3^2];s = Sqrt[k^2 n4^2 - B^2];
 k = 2*N[Pi]/wavelength;
 fourLayerEquation[argument_] := ((p + I s) ((h^2 - p q) Tan[2 halfGuide h] - h
(p + q)) + Exp[-2 p argument] (p - I s) ((h^2 + p q) Tan[2 halfGuide h] + h (p -
q)));
 threeLayerEquation[argument_] := Tan[2 halfGuide h] - h (p + q)/(h^2 - p q);
  CheckInput[n1,n2,n3,n4,halfGuide,wavelength,k,start,end,increment];
 If[increment==0,increment = 1000];
 plotLabel = "     Attenuation (dB/cm)\n          vs\nCladding Thickness
(microns)";
 increment2 = (k n2 - k n3)/10;
  For[variable=start,variable<=end,variable+=increment,
 list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
   If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
    biggestListCount=Length[Complement[list2,{{0,0}}]];
 ];
 ];
  PlotThis[list,plotLabel,biggestListCount];
```

```
] /;
NumberQ[N[n1]]&&NumberQ[N[n2]]&&NumberQ[N[n3]]&&NumberQ[N[n4]]&&Nu
mberQ[N[start]]&&NumberQ[N[end]]&&NumberQ[N[increment]]&&NumberQ[N[g
uide]]&&NumberQ[N[wavelength]]


GuideAttenuation[n1_,n2_,n3_,n4_,cladding_,{start_,end_,increment_},wavele
ngth_] :=
Block[{list,increment2,variable,listCount,biggestListCount,
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,threeLayerEquation},
 biggestListCount = 0;
 flag = 1;
 list = Table[0,{x,0,1},{y,0,1}];
 newStart = .5 start;newEnd = .5 end;newIncrement = .5 increment;
 halfGuide = .5 guide;
 q = Sqrt[B^2 - k^2 n1^2];h = Sqrt[k^2 n2^2 - B^2];
 p = Sqrt[B^2 - k^2 n3^2];s = Sqrt[k^2 n4^2 - B^2];
 k = 2*N[Pi]/wavelength;
 fourLayerEquation[argument_] := ((p + I s) ((h^2 - p q) Tan[2 argument h] - h
(p + q)) + Exp[-2 p cladding] (p - I s) ((h^2 + p q) Tan[2 argument h] + h (p -
q)));
 threeLayerEquation[argument_] := Tan[2 argument h] - h (p + q)/(h^2 - p q);
  CheckInput[n1,n2,n3,n4,newStart,wavelength,k,start,end,increment];
  CheckInput[n1,n2,n3,n4,newEnd,wavelength,k,start,end,increment];
 If[increment==0,increment = 1000];
 plotLabel = "   Attenuation (dB/cm)\n            vs\nGuiding Thickness (microns)";
 increment2 = (k n2 - k n3)/10;
  For[variable=newStart,variable<=newEnd,variable+=newIncrement,
 list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k,n2,n3,flag])];
    If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
     biggestListCount=Length[Complement[list2,{{0,0}}]];
 ];
 ];
  PlotThis[list,plotLabel,biggestListCount];
] /;
NumberQ[N[n1]]&&NumberQ[N[n2]]&&NumberQ[N[n3]]&&NumberQ[N[n4]]&&Nu
mberQ[N[cladding]]&&NumberQ[N[start]]&&NumberQ[N[end]]&&NumberQ[N[inc
rement]]&&NumberQ[N[wavelength]]


WavelengthAttenuation[n1_,n2_,n3_,n4_,cladding_,guide_,{start_,end_,increm
ent_}] :=
Block[{list,increment2,variable,listCount,biggestListCount,
plotLabel,flag,halfGuide,q,h,p,s,k,fourLayerEquation,threeLayerEquation},
 biggestListCount = 0;
 flag = 0;
 list = Table[0,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
  k[argument_] := 2 N[Pi]/argument;
```

```
q[argument_] := Sqrt[B^2 - (k[argument])^2 n1^2];
h[argument_] := Sqrt[(k[argument])^2 n2^2 - B^2];
p[argument_] := Sqrt[B^2 - (k[argument])^2 n3^2];
s[argument_] := Sqrt[(k[argument])^2 n4^2 - B^2];
fourLayerEquation[argument_] := ((p[argument] + I s[argument])
(((h[argument])^2 - p[argument] q[argument]) Tan[2 halfGuide h[argument]] -
h[argument] (p[argument] + q[argument])) + Exp[-2 p[argument] cladding]
(p[argument] - I s[argument]) (((h[argument])^2 + p[argument] q[argument])
Tan[2 halfGuide h[argument]] + h[argument] (p[argument] - q[argument])));
threeLayerEquation[argument_] := Tan[2 halfGuide h[argument]] -
h[argument] (p[argument] + q[argument])/((h[argument])^2 - p[argument]
q[argument]);
  CheckInput[n1,n2,n3,n4,halfGuide,start,k[start],start,end,increment];
  CheckInput[n1,n2,n3,n4,halfGuide,end,k[end],start,end,increment];
 If[increment==0,increment = 1000];
 plotLabel = "Attenuation (dB/cm)\n          vs\nWavelength (microns)";
 For[variable=start,variable<=end,variable+=increment,
  increment2 = (k[variable] n2 - k[variable] n3)/10;
 list =
Union[list,(list2=FindRoots[increment2,variable,threeLayerEquation[variable],
fourLayerEquation[variable],k[variable],n2,n3,flag])];
   If[Length[Complement[list2,{{0,0}}]]>biggestListCount,
    biggestListCount=Length[Complement[list2,{{0,0}}]];
 ];
 ];
 PlotThis[list,plotLabel,biggestListCount];
] /;
NumberQ[N[n1]]&&NumberQ[N[n2]]&&NumberQ[N[n3]]&&NumberQ[N[n4]]&&Nu
mberQ[N[cladding]]&&NumberQ[N[guide]]&&NumberQ[N[start]]&&NumberQ[N[e
nd]]&&NumberQ[N[increment]]


CutoffThickness[{start_,end_,increment_},n1_,n3_,wavelength_] :=
Block[{k,list,n2,cutoffGuide},
 k = 2*N[Pi]/wavelength;
 list = Table[0 x,{x,0,1},{y,0,1}];
 If[n1==n3,
  Print["The parameters entered define a symmetric guide therefore there is
no cutoff"];
  Break[];
 ];
 If[((start>end)&&(increment>0)),
  Print["Input values were entered incorrectly"];
  Break[];
 ];
 If[((n1>=start)&&(start<=n3))||((n1>=end)&&(end<=n3)),
  Print["The parameters entered disobey the requirement for waveguiding
n1<n2>n3"];
  Break[];
 ];
 For[n2=start,n2<=end,n2+=increment,
```

```
cutoffGuide=ArcTan[Sqrt[(n3^2 - n1^2)/(n2^2 - n3^2)]]/(k Sqrt[n2^2 - n3^2]);
list=Union[list,{{n2,cutoffGuide}}];
Print[n2,"  ",cutoffGuide];
];
list = Complement[list,{{0,0}}];
ListPlot[list,PlotStyle -> PointSize[.02],
PlotLabel -> "Cutoff Thickness (microns) vs n2"];
] /;
NumberQ[N[start]]&&NumberQ[N[end]]&&NumberQ[N[increment]]&&NumberQ[
N[n1]]&&NumberQ[N[n3]]&&NumberQ[N[wavelength]]


CutoffWavelength[{start_,end_,increment_},n1_,n3_,guide_]  :=
Block[{n2,list,cutoffWave},
 list = Table[0 x,{x,0,1},{y,0,1}];
 halfGuide = .5 guide;
 If[n1==n3,Print["The parameters entered define a symmetric guide therefore
there is no cutoff"];Break[]];
  If[((start>end)&&(increment>0)),Print["Input values were entered
incorrectly"];Break[]];
 If[(n1>=start) || (start<=n3) || (n1>=end) || (end<=n3),
   Print["The parameters entered disobey the requirement for waveguiding
(n1<n2>n3)"];
  Break[];
 ];
  For[n2=start,n2<=end,n2+=increment,
   cutoffWave=4 halfGuide N[Pi] Sqrt[n2^2 - n3^2]/
    ArcTan[Sqrt[n3^2-n1^2]/Sqrt[n2^2-n3^2]];
   list=Union[list,{{n2,cutoffWave}}];
   Print[n2,"  ",cutoffWave];
 ];
 ListPlot[list,PlotStyle -> PointSize[.02],
 PlotLabel -> "Cutoff Wavelength (microns) vs n2"];
] /;
NumberQ[N[start]]&&NumberQ[N[end]]&&NumberQ[N[increment]]&&NumberQ[
N[n1]]&&NumberQ[N[n3]]&&NumberQ[N[guide]]


End[ (* "`private`" *) ]

Protect[PropagationConstant,N1Attenuation,N2Attenuation,
N3Attenuation,N4Attenuation,CladdingAttenuation,
GuideAttenuation,WavelengthAttenuation,FindRoots,CheckInput,
PlotThis,CutoffThickness,CutoffWavelength]

EndPackage[]
```


END
1-94