NOTICE

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

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

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

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

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C., 1970, c. C-30. Please read the authorization forms which accompany this thesis.

THÈSES CANADIENNES

AVIS

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

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

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

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

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d’auteur. SRC 1970, c. C-30. Veuillez prendre connaissance des formules d’autorisation qui accompagnent cette thèse.

CANADIAN THESES

THIS DISSERTATION HAS BEEN MICROFILMED EXACTLY AS RECEIVED

LA THÈSE A ÉTÉ MICROFILMÉE TELLE QUE NOUS L’AVONS RÉCEVE
PERMISSION TO MICROFILM — AUTORISATION DE MICROFILMER

Please print or type — Écrire en lettres moulées ou dactylographier

Full Name of Author — Nom complet de l’auteur

Irena Streibl

Date of Birth — Date de naissance
10-10-55

Country of Birth — Lieu de naissance
CZECHOSLOVAKIA

Permanent Address — Adresse fixe
3410 Albion Rd.
Ottawa, Ont. K1V 5W3

Title of Thesis — Titre de la thèse
SAW Diffraction Compensation for LiNbO3

University — Université
CARLETON

Degree for which thesis was presented — Grade pour lequel cette thèse fut présentée
M. ENG.

Year this degree conferred — Année d’obtention de ce grade
1984

Name of Supervisor — Nom du directeur de thèse
B.A. SYRETT

Permission is hereby granted to the NATIONAL LIBRARY OF CANADA to microfilm this thesis and to lend or sell copies of the film.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author’s written permission

Date
APRIL 3, 1984

Signature
Irena Streibl

NL-01 (r. 82/11)
SAW DIFFRACTION COMPENSATION FOR LiNbO₃

by

Irena Streibl, B.A.Sc.

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfilment of the requirements for the degree of Master of Engineering

Department of Electronics

Carleton University
Ottawa, Ontario
January 9, 1984
LICENCE TO CARLETON UNIVERSITY

In the interests of facilitating research by others at this institution and elsewhere, I hereby grant a licence to:

CARLETON UNIVERSITY

to make copies of my thesis:

SAW DIFFRACTION COMPENSATION FOR INDO

or substantial parts thereof, the copyright which is invested in me, provided that the licence is subject to the following conditions:

Only single copies shall be made or authorized to be made at any one time, and only in response to a written request from the library of any university or similar institution or its own behalf or on behalf of one of its users.

This licence shall continue for the full term of the copyright, or for so long as may be legally permitted.

The Universal Copyright Notice shall appear on the title page of all copies made under the authority of this licence.

This licence does not permit the sale of authorized copies at a profit, but does permit the collection by the institution or institutions concerned of charges covering actual costs.

All copies under the authority of this licence shall bear a statement to the effect that the copy in question "is being made available in this form by the authority of the copyright owner solely for the purpose of private study and research and may not be copied or reproduced except as permitted by the copyright laws without written authority from the copyright owner."

The foregoing shall in no way preclude the granting by the author of a licence to the National Library of Canada to reproduce the thesis and to lend or to sell copies of the same. For this purpose it shall also be permissible for Carleton University to submit the abovementioned thesis to the National Library of Canada.

Signature of Witness

Signature of Student

Date

Degree

Department of Student
The undersigned recommend to the Faculty of Graduate Studies and Research, the acceptance of the thesis:

"SAW Diffraction Compensation for LiNbO₃"

submitted by Irena Streibi, B.A. Sc., in partial fulfillment of the requirements for the degree of Master of Engineering.

[Signature]
Thesis Supervisor

[Signature]
Chairman, Department of Electronics

Carleton University
March 1984
ABSTRACT

The work in this thesis presents an experimental and analytical study of the effects of diffraction on the frequency response of surface acoustic wave filters.

An algorithm which models and compensates for surface acoustic wave diffraction on Y-cut Z-propagating lithium niobate substrates has been developed. The approach to the diffraction model is based on a physical understanding of the problem and uses the concept of Fresnel integrals with a novel approach to modelling the anisotropic velocity behaviour of the crystal. Based on the diffraction model a diffraction compensation algorithm is developed via time domain considerations.

The approach taken in this thesis is accurate enough to give a 9 dB improvement in side lobe levels for YZ-LiNbO₃ substrates.
ACKNOWLEDGEMENT

I would like to express my deep appreciation for the support and guidance to my thesis' supervisor Dr. Mark S. Sutners of Bell-Northern Research, Ottawa. Without his expertise, continued encouragement and involvement, the completion of this work would not be possible.

The author recognizes the privileges extended to her by Bell-Northern Research, Ottawa, namely Dr. Dave R. Corn of department 1C52 who suggested the thesis topic and provided her with access to the laboratory at the Transmission Development Division 1C00 of Bell-Northern Research, and shared an enthusiastic interest in this work.

Finally, the author is most grateful for the constant guidance and encouragement provided, throughout the entire program, by her thesis supervisor from Carleton University, Professor Barry A. Syrett.
# TABLE OF CONTENTS

## CHAPTER I: INTRODUCTION

1.1 Introduction to the Thesis Topic ........................................... 1
1.2 Thesis Objective ............................................................. 7
1.3 Thesis Outline ............................................................... 10

## CHAPTER II: PHYSICAL THEORY OF SAW DEVICES

2.1 Principle of Operation ....................................................... 12
2.2 Design Considerations ..................................................... 18
   2.2.2 Temperature Coefficient of Delay .............................. 20
   2.2.3 Loss Mechanisms .................................................. 20
   2.2.4 Electromagnetic Feedthrough .................................. 21
   2.2.5 Triple Transit Echo ................................................ 22
   2.2.6 Edge Reflections ................................................... 24
   2.2.7 Bulk Wave Coupling .............................................. 24
2.3 Principles of Diffraction Theory ........................................ 26

## CHAPTER III: DESIGN ALGORITHM

3.1 Design Analysis ............................................................. 31
3.2 Design Procedure .......................................................... 33
3.3 Selection of the Window Function ..................................... 37
   3.3.1 Rectangular Window .............................................. 37
   3.3.2 Hamming Window ................................................. 39
3.4 Computer Design Algorithm .............................................. 42
### TABLE OF CONTENTS (Continued)

<table>
<thead>
<tr>
<th>CHAPTER IV: DIFFRACTION MODELLING</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Introduction</td>
<td>44</td>
</tr>
<tr>
<td>4.2 Diffraction Integral</td>
<td>46</td>
</tr>
<tr>
<td>4.3 Anisotropy of LiNbO$_3$</td>
<td>50</td>
</tr>
<tr>
<td>4.4 Diffraction Integral for LiNbO$_3$</td>
<td>55</td>
</tr>
<tr>
<td>4.5 Modelling of Diffraction</td>
<td>58</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER V: COMPENSATION ALGORITHM</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Introduction</td>
<td>63</td>
</tr>
<tr>
<td>5.2 Development of the Diffraction Compensation Algorithm</td>
<td>65</td>
</tr>
<tr>
<td>5.3 Computer Algorithm</td>
<td>69</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER VI: TESTING AND EXPERIMENTAL RESULTS</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Time Domain Measurements</td>
<td>73</td>
</tr>
<tr>
<td>6.2 Practical Implementation</td>
<td>76</td>
</tr>
<tr>
<td>6.3 Experimental Results</td>
<td>78</td>
</tr>
<tr>
<td>6.3.1 Diffraction Un-compensated Filter</td>
<td>79</td>
</tr>
<tr>
<td>6.3.2 Diffraction Compensated Filter</td>
<td>82</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER VII: CONCLUSIONS AND FUTURE WORK</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 Conclusions</td>
<td>87</td>
</tr>
<tr>
<td>7.2 Future Work</td>
<td>89</td>
</tr>
</tbody>
</table>
TABLE OF CONTENTS (Continued)

<table>
<thead>
<tr>
<th>REFERENCES</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPENDIX I: FRESNEL INTEGRAL</td>
<td>94</td>
</tr>
<tr>
<td>APPENDIX II: COMPUTATION OF FRESNEL INTEGRALS</td>
<td>98</td>
</tr>
<tr>
<td>APPENDIX III: FRESNEL INTEGRAL WITH THE ANISOTROPIC APPROXIMATION FOR YZ-LnInO$_3$</td>
<td>100</td>
</tr>
<tr>
<td>APPENDIX IV: FREQUENCY DIVIDER CIRCUIT</td>
<td>104</td>
</tr>
<tr>
<td>APPENDIX V: FABRICATION SUMMARY</td>
<td>105</td>
</tr>
<tr>
<td>APPENDIX VI: COMPUTER PROGRAM LISTINGS</td>
<td>108</td>
</tr>
</tbody>
</table>
LIST OF SYMBOLS

$A_m$ amplitude of the $m$-th sample

$B_a$ reactive component of the acoustic admittance

$C_T$ static capacitance

DFT Discrete Fourier Transform

FFT Fast Fourier Transform

$e$ base $e = 2.718281828$

$f$ frequency

$f_0$ center frequency

$f_e$ lower passband frequency

$f_s$ sampling frequency

$f_u$ upper passband frequency

$f_{se}$ lower stopband frequency

$f_{su}$ upper stopband frequency

$G_a$ radiation conductance

$h(t)$ impulse response

$H(f)$ frequency response

IDT interdigital transducer

IL insertion loss

$I(x)$ current distribution on the radiating element
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>k</td>
<td>acoustic wave propagation constant</td>
</tr>
<tr>
<td>$k_e$</td>
<td>acoustic energy propagation constant</td>
</tr>
<tr>
<td>$k^2$</td>
<td>coupling coefficient</td>
</tr>
<tr>
<td>L</td>
<td>length of the radiating element scaled to the acoustic wavelength</td>
</tr>
<tr>
<td>$+L/2,-L/2$</td>
<td>integration limits for the radiating element</td>
</tr>
<tr>
<td>LiNbO$_3$</td>
<td>lithium niobate</td>
</tr>
<tr>
<td>N</td>
<td>number of transducer fingers</td>
</tr>
<tr>
<td>NP</td>
<td>number of transducer finger pairs</td>
</tr>
<tr>
<td>$N_s$</td>
<td>number of samples required</td>
</tr>
<tr>
<td>P</td>
<td>field point</td>
</tr>
<tr>
<td>$S_n$</td>
<td>sign of the n-th sample</td>
</tr>
<tr>
<td>t</td>
<td>time</td>
</tr>
<tr>
<td>$t_n$</td>
<td>time of the n-th sample</td>
</tr>
<tr>
<td>$\Delta t$</td>
<td>time increment from finger to finger</td>
</tr>
<tr>
<td>$t_+,t_-$</td>
<td>limits of the Fresnel integral</td>
</tr>
<tr>
<td>TTE</td>
<td>triple transit echo</td>
</tr>
<tr>
<td>$v,v_p$</td>
<td>acoustic phase velocity</td>
</tr>
<tr>
<td>$v_e$</td>
<td>acoustic energy velocity</td>
</tr>
<tr>
<td>$v_f$</td>
<td>unmetallized surface acoustic velocity</td>
</tr>
<tr>
<td>$v_g$</td>
<td>group velocity</td>
</tr>
</tbody>
</table>
LIST OF SYMBOLS (Continued)

\( v_s \)  
metallized surface acoustic velocity

\( v_o \)  
acoustic velocity at the pure mode axis

\( v_+^2, v_-^2 \)  
limits of the Fresnel integral

\( X \)  
distance from the center of the radiating element to the field point along the x-axis, scaled by the acoustic wavelength

\( y_s \)  
acoustic admittance

\( Z \)  
distance from the radiating element of the field point along the pure mode axis, scaled by the acoustic wavelength.

\( \sigma(t + \tau_m) \)  
Dirac delta function

\( \phi \)  
angle of propagation

\( \lambda \)  
free space wavelength

\( \lambda_a \)  
acoustic wavelength

\( \lambda_e \)  
energy wavelength

\( \lambda_g \)  
group wavelength

\( \rho \)  
distance from the center of the radiating element to the field point

\( \rho_o \)  
distance from the radiating element to the field point along the pure mode axis

\( \tau_D \)  
delay between two transducers
LIST OF SYMBOLS (Continued)

\( r_i \) delay between the center of the input transducer and the edge of the substrate

\( r_o \) delay between the center of the output transducer and the edge of the substrate

\( w \) angular frequency

\( w_s \) sampling angular frequency
CHAPTER I

INTRODUCTION

1.1 Introduction to the Thesis Topic

Surface acoustic waves have developed as an important new technology since the publication of the first delay line by White [1] in 1965. There are many advantages to using surface acoustic wave (SAW) devices in place of conventional components. Low cost, small size, uniformity of performance and ease of manufacture have contributed to making this a successful technology. To illustrate these points a SAW IF band pass filter as shown in Figure 1.1 fits into a TO-8 header. In large quantity production runs (>10,000/yr.) it would cost about $5-$20 per unit to manufacture.

The techniques used in the fabrication of SAW devices are compatible with the planar technology used for integrated circuits. Since the substrate is a chemically pure single crystal and since the device pattern is highly reproducible, the performance from device to device is virtually identical. This results in devices which do not need to be tuned once manufactured. A simple visual screening of the devices ensures a yield approaching 100%.

A few of the limitations of SAW devices will be discussed below.

A conventional interdigital transducer (IDT)\(^1\) has a simplified equivalent electrical circuit consisting of the acoustic admittance \(Y_a = G_a + jB_a\) and the static capacitance \(C_a\) (Figure 1.2).

\(^1\) The IDT is described in Chapter 2, Section 2.1.
FIGURE 1.1: 70 MHz filter IMS-102 mounted in a TO-8 header
FIGURE 1.2: An equivalent circuit of an interdigital transducer (IDT). $Y_{in}$ is the input admittance of an IDT.

The real component of the acoustic admittance, $G_a$, is the radiation conductance and it represents the electrical to mechanical energy transformation. The imaginary component $B_a$ is the reactive component of the acoustic admittance. Both $G_a$ and $B_a$ are frequency dependent terms. $C_T$ is the static capacitance which is frequency independent. $C_T$ depends on the number of fingers, the finger overlap, the spacing between fingers and the dielectric constant of the substrate.

When the input transducer is matched at the electrical port, half of the acoustic power is transmitted towards the output transducer and the other half is dissipated in an acoustic load (Figure 1.3). By reciprocity, the same applies to the output transducer. Hence there

FIGURE 1.3: Signal transmission with bi-directional IDT's
is a minimum built-in loss of 3 dB (\( \frac{1}{2} \) power) for each transducer. This inherent loss is usually compensated for by amplification.

Since it is very difficult to obtain a perfect acoustic match, part of the signal travelling to the left of the input IDT will be reflected back to the receiving IDT from the substrate edge. This unwanted delayed image will cause ripple in the filter’s passband. Cutting the substrate edge at an angle helps to disperse these reflections. Applying an acoustic absorber at the edges also significantly attenuates these reflections. The problem of reflections applies to the output transducer as well.

There exists another reflection inherent to bi-directional transducers, the triple transit echo (TTE) which can be suppressed but never completely eliminated. The transmitted signal of Figure 1.3 reaches the output IDT and part of it is reflected back towards the input IDT where it gets re-reflected to the output. This delayed signal is typically 40 to 50 dB down from the main signal and causes a passband ripple in the order of ±0.05 dB [2].

The metallization of the IDT is another source of reflections, however, these may be suppressed by the use of split fingers. Fingers spaced at a half wavelength separation will cause the reflections to build up in phase. By selecting the split fingers design, the reflections from successive fingers will cancel each other [2].

Diffraction is a source of wavefront modification related to propagation in any medium. In SAW devices, the acoustic beam spreads out as it propagates away from the transmitting IDT. As a result, the
receiving transducer will intercept only a fraction of the transmitted power. This problem occurs predominantly in apodized\(^1\) transducers where there exist regions of very small overlap. To overcome this problem, the finger overlap must be adjusted. This is the subject of Chapter 5.

The distortion due to diffraction is most noticeable in the filter's stopband as an increase in the level of the sidelobes (Figure 1.4). Also some degradation of the passband ripple and slope is attributed to diffraction.

In the next section the thesis objective is defined and the sequence of work presented is outlined.

\(^1\) In an apodized IDT the fingers are not of uniform length, that is their length varies from finger to finger.
FIGURE 1.4: Predicted frequency response of an ideal transversal filter compared to the experimental result.
1.2 Thesis Objective

Many authors have addressed the topic of diffraction modelling and diffraction compensation for SAW devices in recent years [3,4,5,6,7,8]. However, no universally successful solution has been proposed. The investigations reported have been unique to specific substrates and the results have not been generalized.

Lithium Niobate (LiNbO₃) is a substrate of great interest for fabrication of surface acoustic wave filters. It has a high electromechanical coupling coefficient $k^2$ which permits high transducer conversion efficiency for wideband filters. However, modelling the effects of diffraction on this material has proven to be difficult because of the highly anisotropic nature of the velocity versus the direction of propagation.

It is the objective of this thesis to isolate the effects of diffraction in the response of a SAW IF filter built on a LiNbO₃ substrate and to develop a computer model for this effect. Based on the computer model a new filter is designed, fabricated and tested. This new filter has correction for diffraction incorporated into its design. The frequency response of the latter device is compared to the desired response and the model and the compensation technique are discussed and evaluated. The errors in the design algorithms are quantified and suggestions for refinement of the numerical techniques are presented.

Figure 1.5 outlines the work that has been done to meet the thesis objective. As the first step a design without diffraction considerations was completed. A filter based on those specifications was fabricated. Using this first device as a benchmark, the effects of diffraction were measured and diffraction profiles, showing the
FIGURE 1.5: Flowchart of work presented in this thesis
spreading of the SAW wave energy as the SAW wave travelled from the input IDT to the output IDT, were predicted on the computer. The theoretical model, based on the diffraction profiles, was developed and used to compare this design to the experimental frequency response. Based on this model a new diffraction-compensated filter design was completed. Another filter was fabricated from the compensated design and its predicted frequency response was compared to the experimental result. Finally the effectiveness of the model and compensation algorithm were evaluated.
1.3 Thesis Outline

Chapter 2 outlines the basic operation of surface acoustic waves and presents terminology specifically related to SAW devices. The physical requirements for a surface wave to propagate are stated and wave propagation in an isotropic medium is compared to that in an anisotropic medium. Further, the design considerations in SAW filters are discussed as well as the radiation from a slit according to classical optics [9].

Chapter 3 describes the procedure used to design a SAW filter without any consideration of diffraction effects. A flowchart of the computer design algorithm is included.

Chapter 4 starts with a discussion of the literature covering the topic of diffraction modelling and diffraction compensation. The theory developed in this thesis for modelling diffraction on LiNbO$_3$ is summarized. The diffraction integral is derived for an isotropic medium and modifications to the integral are presented in order to include the anisotropy on LiNbO$_3$. Computer modelling for both cases is discussed.

In Chapter 5 the diffraction compensation algorithm is derived. The computer implementation of the compensation algorithm is discussed and the predicted diffraction frequency response for both the isotropic and the anisotropic model are compared.

The experimental results of all the devices fabricated are presented and discussed in Chapter 6. A description of the experimental test set-up used to isolate the effects of diffraction is also described.
Finally, in Chapter 7, conclusions are drawn from the experimental results and the modelling and compensation techniques are reviewed. Recommendations for future work are discussed as a completion of this work.
CHAPTER II

PHYSICAL THEORY OF SAW DEVICES

2.1 Principle of Operation

Surface acoustic waves are generated by an interdigital transducer (IDT) deposited on a piezoelectric substrate (Figure 2.1).

![Diagram of an interdigital transducer on a piezoelectric substrate]

FIGURE 2.1: Prototype of a SAW IDT on YZ-LINbO₃

The term piezoelectricity means an interdependence of elastic and electric properties in certain materials. A material is piezoelectric if a mechanical deformation induces an electric polarization (direct effect) or conversely if it experiences a mechanical deformation (strain) when an electric field is applied (inverse effect). The discovery of the direct piezoelectric effect was made by Pierre and Jacques Curie in 1880 [10].

The interdigital transducer (IDT) is composed of two metal comb-shaped electrodes. The voltage applied to these electrodes or fingers, produces an electric field which creates a stress-strain relationship in the vicinity of the surface giving rise to various types of elastic waves. A surface wave is, in principle, any deforma-
tion that when propagating affects only a small depth of material near the surface. The elastic wave of interest is the Rayleigh wave (Figure 2.2) which is generated in the direction orthogonal to the comb fingers, i.e., the Z-direction of Figure 2.1.

Rayleigh waves consist of a longitudinal displacement and a π/2 phase shifted vertical displacement (Figure 2.3). The extremity of the polarization vector describes an ellipse and the particle displacement within the sagittal plane\(^1\) vanishes at a depth of two wavelengths (Figure 2.2). Since Rayleigh waves are confined to the surface of the crystal, successive wave crests may be tapped or generated by an interdigital transducer which is deposited on the surface plane.

As far as the Rayleigh waves are concerned, the transducer can be modelled as a sequence of ultrasonic line sources (Figure 2.4). For an applied sinusoidal voltage, all vibrations interfere constructively only if the distance \(d\) (Figure 2.1) between two neighbouring fingers is close to half the elastic wavelength. Any deformation created at time \(t\) by a finger pair, for a given polarity of the applied voltage, travels at the Rayleigh velocity \(V_R\) along a distance \(\lambda/2\) during the half period \(T/2\). At time \(t + T/2\) this deformation has reached the next pair of fingers precisely when the voltage, which has changed sign, generates an in-phase deformation; the elastic excitation due to the second pair re-enforces that of the first pair. The frequency \(f_0 = V_R/2d\) that corresponds to this cumulative effect is called the synchronous frequency or the resonant frequency of the transducer. If the frequency departs from this value, the interference between the elastic signals due to various finger pairs is less constructive and the overall vibration is weaker. Thus the bandwidth of a transducer is dependent on the number of fingers.

---

\(^1\) The sagittal plane is perpendicular to the plane of propagation in the direction of propagation.
FIGURE 2.2: The propagation of a Rayleigh wave in a crystal [10]

FIGURE 2.3: Amplitude of displacement components as a function of depth. Displacement components are in phase quadrature [2].
FIGURE 2.4: Method of discrete sources. The transducer is regarded as a discrete sequence of sources, each one located midway between two fingers, and of amplitude proportional to the overlap length.

In the discrete source approach each finger is considered as an infinitely narrow ultrasonic source or receiver located midway between two fingers, as shown in Figure 2.4. The amplitude \( A \) assigned to each source is proportional to the overlap length of the two fingers, with positive or negative sign depending on the direction of the electric field.

With these assumptions the impulse response \( h(t) \) of a transducer is a series of Dirac functions:

\[
h(t) = \sum_n \frac{A_n}{\Delta t} \delta(t - t_n)
\]  

(2.1)
The sign $S_n = (-1)^n$ implies that the electric field is reversed from one interdigital interval to the next.

The frequency response is obtained as:

$$H(f) = \sum_n S_n A_n e^{-j 2\pi f_n}$$  \hspace{1cm} (2.2)

which is the Fourier transform of the impulse response equation (2.1):

$$H(f) = \int_{-\infty}^{\infty} h(t) e^{-j 2\pi f t} \, dt$$  \hspace{1cm} (2.3)

A sequence of delta functions represents not only the sine wave at the center frequency $f_o$, but also all the odd harmonic sine waves at frequencies $(2m + 1) f_o$, $m = 0, 1, 2 \ldots n$ (Figure 2.5).

![Fundamental and 3rd Harmonic](image)

**FIGURE 2.5:** A sequence of delta functions. Only odd harmonics are present.

Due to the half wavelength spacing between the fingers of the IDT, all of the even harmonic components cancel completely.
The electric field associated with the elastic wave, when passing under the electrodes, induces a time varying voltage. The shape of this signal depends on the number of receiver fingers (Figure 2.6). It follows from Figure 2.6 that the overall filter's impulse response is the convolution of the impulse responses of the input IDT and of the receiving IDT. The frequency response is the Fourier transform of the filter's impulse response.

![Figure 2.6: Impulse response of a delay line consisting of two interdigital transducers [10].](image)

(a) The receiver has only one finger pair
(b) The receiver is identical to the emitter.
2.2 Design Considerations

There are several important parameters and corruptions affecting the performance of a SAW device.

2.2.1 The Coupling Coefficient

It is customary to define a coupling constant for surface waves [10] by

$$k^2 = 2 \left( \frac{f}{v} - \frac{v_s}{v} \right)$$

(2.4)

where $v_f$ = the free unmetallized surface velocity
$v_s$ = the metallized surface velocity.

This electromechanical coupling constant $k^2$ measures the relative effect of a metal film on the propagation velocity $v$. It characterizes the material's ability to transform an electric signal into an elastic surface wave through a transducer consisting of metal fingers deposited on the surface.

For Y-cut lithium niobate (Figure 2.7) the maximum difference between the two velocities occurs when the propagation is along the Z-axis: $v_f = 3488 \text{ m/s}$ and $v_s = 3404 \text{ m/s}$. The resulting value of $k^2 \equiv 4.94\%$ is quite high compared to other piezoelectric substrates [10].
FIGURE 2.7: Rayleigh wave in lithium niobate (Y-cut). Velocities $v_n$ on metallized surface and $v_f$ on unmetallized surface vs. propagation direction in the XZ plane [10].
2.2.2 Temperature-Coefficient of Delay

The temperature coefficient of delay is another important consideration in SAW devices. A change in the temperature causes a change in the surface wave velocity and the transducer's center frequency will change correspondingly. The time delay between the input and the output transducer will also vary. The value of this coefficient is -90 ppm/°C for Y-cut Z-propagating LiNbO₃.

2.2.3 Loss Mechanisms

Acoustic attenuation is a fundamental loss mechanism where the attenuation constant is approximately proportional to the square of the frequency. As the surface wave propagates, part of its energy is scattered by the solid and dissipates as heat in the substrate. This loss is of importance at frequencies above 500 MHz; at standard IF frequencies it is negligible. The same holds true for the ohmic losses in the transducer electrodes. There is little loss due to the coupling of surface waves to bulk waves for the crystal cuts normally used, since bulk waves travel faster than surface waves thus binding the energy to the surface. The attenuation of a surface-wave by coupling to ultrasonic waves in the air can be ignored for frequencies below 1 GHz.
2.2.4 Electromagnetic Feedthrough

There is always a certain amount of electromagnetic (EM) feedthrough. This means that not all of the electrical signal follows the acoustic path between the input and output IDT. Some of it is transmitted by electromagnetic waves which travel at the speed of light.

The feedthrough signal occurs "instantly" when the input voltage is applied while the desired signal is delayed by travelling at the acoustic velocity along the surface of the crystal (Figure 2.8) [11]. Beating of these two signals will cause ripples in the frequency response. By careful packaging of the device the EM feedthrough noise floor can be brought down to be better than -60 dB.

![Diagram of signal delays](image)

**FIGURE 2.8:** The delay between signals in time domain.

The EM feedthrough arrives at the receiver "instantly" followed by the longitudinal bulkwave. The main SAW signal coincides with the shear bulkwave. Reflections from the edges arrive later and finally the triple transit echo.
2.2.5 Triple Transit Echo

The triple transit echo (TTE) causes a further interference for bidirectional transducers (Figure 2.9). When a voltage is applied to the electrodes of the input IDT half of the signal is transmitted in the desired propagation direction towards the output IDT. As the main signal reaches the output IDT part of it is reflected back towards the input IDT where it is re-reflected again to the output IDT. This delayed and attenuated image will beat with the main signal thus causing ripples in the magnitude and phase of the frequency response. The frequency of this ripple is determined by the inverse of twice the delay time $\tau_D$ between the two transducers [11].

$$f_{\text{TTE}} = \frac{1}{2\tau_D}$$  \hspace{1cm} (2.5)

The level of the TTE interference relates to the level of the main signal by the following expression [12]:

$$\frac{\text{TTE power}}{\text{main signal power}} \text{ (dB)} = 6\text{(dB)} + 2\text{IL (dB)}$$  \hspace{1cm} (2.6)

where $\text{IL}$ is the insertion loss of the transducer.

The triple transit echo can be eliminated by building unidirectional transducers. However these designs are much more involved and are more costly to fabricate than the bidirectional transducers. The system application of the filter will dictate if such an approach should be used.
FIGURE 2.9: SAW delay line with two bidirectional interdigital transducers
2.2.6 Edge Reflections

The reflections from the substrate edges are also a source of passband ripple (Figure 2.10). The frequency of the ripple due to the input IDT is equal to the inverse of twice the delay time between the center of the IDT and the substrate edge:

\[ f_i = \frac{1}{2\tau_i} \]  

(2.7)

and similarly for the output IDT

\[ f_o = \frac{1}{2\tau_o} \]  

(2.8)

However, these reflections can be quite easily eliminated by applying an acoustic absorber at the edges and by cutting the substrate at an angle with respect to the transducer fingers. In the latter case the signal will be reflected away from the transducer and the possibility of performance degradation due to reflections is eliminated.

2.2.7 Bulk Wave Coupling

For YZ-LiNbO₃ crystals, the bulk wave coupling is another source of performance degradation. The transducer excites not only Rayleigh waves but also different modes of bulk waves [13,14,15]. As the frequency increases the bulk waves propagate steeper into the substrate. But the slow shear bulk wave travels at a velocity only 3% higher than the Rayleigh wave and it can propagate near the surface [13]. Hence at the high frequency side of the passband the coupling of the bulk wave significantly increases the sidelobe levels.
FIGURE 2.10(a): Passband response showing ripples due to end of substrate reflections.

FIGURE 2.10(b): Passband response without ripples from end of substrate reflections.
2.3 Principles of Diffraction Theory

Diffraction is a phenomenon inherent to wave propagation regardless of the propagation medium.

Diffraction is the spreading of the acoustic beam as shown in Figure 2.10.

![Diagram of beam spreading due to diffraction](image)

**FIGURE 2.11**: Beam spreading due to diffraction.

It is apparent from the above Figure 2.10 that not all of the radiated energy is intercepted by the output IDT. This introduces an additional source of performance degradation in SAW devices.

Diffraction in SAW devices is analogous to the diffraction encountered in optics when a plane wave illuminates a slit (Figure 2.11) [9,16,17]. In contrast to the classical optical case, the propagation medium of SAW devices is highly anisotropic and the resulting variation of phase velocity with the angle of propagation must be taken into account in diffraction calculations [8].

In the case of isotropic diffraction the phase velocity remains constant regardless of the direction of propagation. Figure 2.12(a) illustrates how the isotropic diffraction causes the beam profile to
FIGURE 2.12: Plane wave incident on a slit in a screen. The edge
diffraction leads to spreading of the radiation from the
slit [16].

quickly lose its uniform amplitude and phase as it travels away from
an input transducer. The profile undergoes significant fluctuations
in shape and it eventually broadens to an infinite width. For the
anisotropic case (Figure 2.12(b)) the direction of phase and group
velocity vectors coincide only for specific values of θ defined as the
pure mode axes. Thus the acoustic beam is shown propagating off at an
angle $\phi$, the power flow angle. For an anisotropic substrate the
diffraction patterns are unique to the particular anisotropy and do
not occur in the isotropic manner.

Now consider the situation shown in Figure 2.11 in which an
infinite plane wave is incident on an infinite flat sheet. The sheet
has a slot of width a and of infinite length in the direction normal
to the page. The field distribution across the slot may be assumed to
be uniform and the field everywhere to the right of the sheet is the
result of the section of the wave that passes through the slot.

By Huygens' principle [17] the field everywhere to the right of
the sheet is the same as though each point in the plane of the slot is
the source of a new spherical wave (Figure 2.11). Each of these point
sources is of equal magnitude and phase. Thus by Huygens' principle
the slotted sheet with a uniform field across the opening can be
replaced by a continuous array of point sources which just fill the
opening. The field pattern in the X-Z plane is then calculated in the
same way as for a continuous linear array of point sources.
FIGURE 2.13: (a) Schematic representation of launching and propagation of acoustic surface wave on an isotropic substrate. Note diffraction effects. $\theta$ defines direction of propagation with respect to reference crystalline axis.

(b) Schematic representation of launching and propagation of acoustic surface wave on anisotropic substrate. Note diffraction effects and how the phase velocity and power flow directions are separated by angle $\phi$. $\phi$ defines the direction of propagation while $\theta$ is power flow angle [8].
The diffraction patterns can be computed by applying Huygens' principle to the linear radiator (Figure 2.13):

![Linear radiator configuration](image)

**FIGURE 2.14: Linear radiator configuration**

The field at point P is expressed as

\[
E(x_1, R) = K \int_{-L/2}^{+L/2} I(x) \frac{e^{-jkr}}{\rho} \, dx (2.9)
\]

where
- \( k = \frac{2\pi}{\lambda} \) is the wave number
- \( K \) = constant along x-axis
- \( I(x) \) = current distribution on the element
- \( \rho \) = distance from the radiating element to the field point P.

The solution in the region near the radiating element is given by the Fresnel integral and the patterns are called the Fresnel patterns (Figures 2.14(a) and (b)). The far field pattern takes on the \( \sin x/x \) shape (Figure 2.14(c)) and is called the Fraunhofer diffraction pattern [18]. It is shown in polar coordinates in Figure 2.14(d).
FIGURE 2.15: Fresnel and Fraunhofer patterns of a slot of width $a$
[17].

Now that the principles of operation of SAW devices as well as
some of the physical design considerations have been reviewed, one can
proceed with the development of the design algorithm in Chapter 3.
CHAPTER III

DESIGN ALGORITHM

3.1 Design Analysis

Given the desired frequency specification of a filter (Figure 3.1) a design procedure must be determined which will satisfy the required criteria.

The filter's impulse response is the Inverse Fourier Transform [19] of the desired frequency response (Figure 3.2). The infinite impulse response is then truncated by applying the appropriate window function [20].

![Graph of the impulse response and frequency response]

**FIGURE 3.1:** Design Specification of a 70 MHz Filter

The width of the window in the time domain is determined from the transition width in the frequency domain, i.e., the convolving function of Figure 3.1. The multiplication of the impulse response by the specified window function yields the filter's finite impulse response. Taking the Discrete Fourier Transform (DFT) [20] of the finite impulse response results in the final frequency response. The resulting frequency response can be alternatively obtained by convolving the desired frequency response with the convolving function, i.e., the transition width in the frequency domain (Figure 3.1).
FIGURE 3.2: Generalized design process [11]
3.2 Design Procedure

As the first step in the filter design the number of fingers of the input non-apodized transducer will be determined. The finger overlaps are of constant width and hence carry uniform weighting of ±1 depending on the polarity of the connecting rail (ref. Figure 2.1). The term "apodized" means a variation in the length of the finger overlaps which will be the case for the output receiving transducer.

The number of fingers required for the input IDT is given by the position of the first null of the filter's frequency response. Figure 3.1 specifies the first nulls to be at 63 and 77 MHz. Since the uniform non-apodized transducer has a rectangular profile in the time domain, its frequency response will be the \( \sin x/x \) function. The position of the nulls of the sinc function (Figure 3.3) is related to the number of finger pairs by the following expression:

\[
(1 \pm \frac{n}{NP}) f_o = f_n ; \quad n = \pm 1, \pm 2, \ldots
\]  

(3.1)

![Figure 3.3: Position of nulls of \( \sin x/x \) dependence on the number of finger pairs NP.](image)
From (3.1) follows the choice of 9.5 finger pairs or 19 fingers with the first nulls positioned at 62.6 MHz and 77.4 MHz.

The specification given by Figure 3.1 requires the roll-off to be less than 1 dB at 66.5 MHz and 73.5 MHz. However, with the present design it is -3.6 dB at those frequencies. This discrepancy will be compensated for in the design of the output apodized IDT.

Figure 3.4(b) shows the desired rectangular filter response while Figure 3.4(a) shows the sin x/x frequency response of the input IDT. Dividing the position of the sin x/x bounded by f₀ ± Δf into the ideal response yields the required frequency response of the apodized IDT Figure 3.4(c). The "horns" of this response compensate for the roll-off due to the sin x/x response of the non-apodized IDT. Since the filter's frequency response is obtained by multiplying the frequency response of the input IDT with that of the output IDT, one can see that multiplying the waveforms of Figures 3.4(a) and 3.4(c) yields the required filter response shown in 3.4(b).

The next step is to determine the output IDT's impulse response by using the Inverse Fast Fourier Transform [21]. This yields the infinite impulse response which is truncated by multiplication with the window function (Figures 3.4(d) and (e)). The apodization weighting of the output IDT is specified by the sample weights of the truncated impulse response.

The convolution of the input IDT's impulse response, Figure 3.5(a), with that of the output IDT, Figure 3.5(b), yields the filter's finite impulse response (FIR) as indicated in Figure 3.5(c). The FFT of the finite impulse response results in the final frequency response of the filter, Figure 3.5(d).

The design sequence outlined above may have to be repeated 2 or 3 times to achieve the required frequency specification. The width Δf of Figure 3.4 has to be adjusted by trial and error to compensate for the effect of the window function at the "horns" of the apodized IDT's frequency response Figure 3.4(c).
FIGURE 3.4: Design process of a SAW filter

\[ f_0 = \text{center frequency} \]

\[ 2\Delta f = \text{passband width} \]

\[ \tau_D = \text{delay between the input and output transducers} \]
FIGURE 38: Basic steps in filter design
3.3 Selection of the Window Function

The selection of the right window function is an important consideration in the filter design. The procedure used here is based on that developed by Harris [22].

It is evident from Figure 3.1 that the transition width of the filter is specified by the position of the first null of the window function.

3.3.1 Rectangular Window

For a rectangular window the resolution between the first nulls is equal to 2 bins (Figure 3.6), where a bin is the fundamental frequency resolution.

\[ 1 \text{ bin} = \frac{\omega_s}{N} \]  

(3.2)

where \( \omega_s \) = sampling frequency

\( N \) = number of samples, one sample corresponds to one finger
FIGURE 3.6: Fourier transform of rectangular window is \( \sin x/x \). The first side lobe level is -13.4 dB. A bin is the fundamental frequency resolution.

The sampling frequency is equal to \( 2f_o \) since the analysis assumes two fingers per wavelength.

Figure 3.1 indicates that 1 bin = 3.5 MHz and one can calculate the necessary number of fingers using (3.2) as:

\[
N = \frac{\omega}{1 \text{ bin}}
\]

Substituting the above values into (3.3) and letting \( f_o = 70 \) MHz yields:

\[
N = \frac{2\pi \left( 2f_o \right)}{2\pi \left( 1 \text{ bin} \right)} = \frac{2 \left( 70 \right)}{3.5} = 40 \text{ fingers}
\]
Hence the transition width of 3.5 MHz corresponds to 40 fingers. As a safeguard the transition width should be reduced by 20% corresponding to 700 kHz. The new transition width is 2.8 MHz and corresponds to 50 fingers.

The truncation of the impulse response by a rectangular window causes significant ripple in the filter's passband as indicated in Figure 3.2. Figure 3.1 specifies the sidelobe level to be below -40 dB. Here again the rectangular window is unsuitable since it has the first sidelobe at -13.4 dB. Another window function must be considered then to meet the filter's criteria.

3.3.2 Hamming Window

The Hamming window belongs to the \( \cos \alpha(x) \) family of windows. The parameter \( \alpha = 0.54 \) and the first side lobe level is -43 dB.

The general form of the Hamming window is [22]:

\[
\omega(n) = \begin{cases} 
0.54 + 0.46 \cos \left( \frac{2\pi}{N} n \right); & n = -\frac{N}{2}, \ldots, -1, 0, 1, \ldots, \frac{N}{2} \\
0.54 - 0.46 \cos \left( \frac{2\pi}{N} n \right); & n = 0, 1, 2, \ldots, N-1
\end{cases}
\]  (3.4)

where \( N \) = total number of samples (fingers)
\( n \) = \( n \)-th sample (finger)

As in the case of the rectangular window the first zero occurs at 3.5 MHz and it is defined by:

\[
\omega = 2.6 \left( \frac{2\pi}{N} \right) f_s
\]  (3.5)

where \( \omega \) = position of the first null
\( f_s \) = sampling frequency
\( N_s \) = number of samples (fingers) required
From equation (3.5) the number of fingers required follows as:

\[ N = 2.6 \left( \frac{2\pi f_s}{\omega} \right) \frac{1}{\omega} \quad (3.6) \]

Substituting for \( f_s = 140 \) MHz and \( \omega = 3.5 \) MHz \((2\pi)\) yields:

\[ N = 2.6 \left( 2\pi \right) \left( 140 \right) \frac{1}{2\pi \cdot 3.5} \]

\[ N = 104 \text{ fingers} \]

Applying a 20% guardband increases the number of fingers required to 125.

Figure 3.7 compares the truncation of the infinite impulse response as done by the rectangular window and by the Hamming window. The Hamming window, a cosine function on a pedestal, reduces the width of the main lobe as well as the level of the adjacent lobes. This translates into the frequency domain as reduced ripple in the passband and sidelobe levels below -43 dB.

The conclusion can be drawn that the Hamming window is the most suitable one to use to meet the required filter specification indicated by Figure 3.1.
FIGURE 3.7: The truncation of the infinite impulse response by multiplication with the rectangular and Hamming windows.
3.4 Computer Design Algorithm

The flowchart of the computer design routines is indicated in Figure 3.8.

The vectors representing the input and output IDT consist of 512 samples each. The spacing between samples corresponds to half a wavelength which can be converted to $\Delta t = \frac{1}{2f_0}$ in the time domain. The total length of 512 samples is based on the amount of work space required to perform numerical convolution and the FFT. For an easier application of the FFT it is recommended that the vector be selected as a power of two [21], i.e., $512 = (2)^9$.

In step 1, the 512 sample vector representing the input IDT is generated and it consists of a sequence of samples with weighting $+1$ or $-1$ and zeros. The initial sequence of $+1$ represents the 19 fingers of the non-apodized IDT and the remaining sample values are set to zero.

In the next step the frequency response of the non-apodized IDT is computed using the FFT algorithm. The output IDT's frequency response is predistorted in order to compensate for the input IDT's frequency response.

Next the Inverse Fourier Transform is applied to the output IDT's frequency response and its time domain response is evaluated.

Step 4 applies the Hamming window function to truncate the infinite impulse response of the output IDT and the 125 sample output IDT is positioned at the correct delay with respect to the input IDT.

Finally, the numerical convolution of the two IDT's is performed which yields the finite impulse response of the filter. The last check in the design sequence is the frequency response obtained by applying the FFT to the filter's impulse response.
FIGURE 3.8: Flowchart of design sequence of a SAW filter
CHAPTER IV

DIFFRACTION MODELLING

4.1 Introduction

Diffraction modelling in SAW devices is an area of great interest. Because of the highly anisotropic behaviour of the materials used for the fabrication of these devices it is by no means a trivial task to model the diffraction effects. Since the anisotropy, the change in velocity with the propagation direction, is unique to each substrate, there exists no universal solution to this problem.

Szabo and Slobodnik [8] used the parabolic velocity surface theory to model diffraction on quartz and the angular spectrum of waves theory to model diffraction on YZ-LINbO₃ [36]. They reported good agreement between the predicted and experimental diffraction profiles for quartz but not for YZ-LINbO₃.

An efficient numerical technique for simulating the effect of diffraction in SAW devices was developed by Penuri [7]. It was tested for parabolic diffraction and the experimental and theoretical frequency responses were typically within 2 dB on ST-quartz.

Savage and Mattheer [5] modelled diffraction using the Fresnel and parabolic anisotropy approximations for ST-quartz substrate. They published good improvement in the filter's frequency response when diffraction has been compensated for.

Mader [6] developed a compensation algorithm which computes corrections to the tap weights from the deviation of the measured frequency response from the desired one. They achieved excellent agreement between theory and experiment since their method compensates for other second-order effects as well.
Fast computation of diffraction in general anisotropic media by use of the Geometrical Theory of Diffraction (GTD) was presented by Radasky and Matthaei [4]. Their technique provides a means for computing diffraction patterns on a medium like YZ-LiNbO$_3$, where the parabolic anisotropic approximation is not valid. Comparisons between the diffraction profiles obtained by using the angular spectrum calculations and the GTD on YZ-LiNbO$_3$ substrate indicate that the GTD technique is very good, however the angular spectrum of waves solution did not give good results for YZ-LiNbO$_3$ when used by Szabo and Siobodnik [8].

The technique for diffraction compensation on YZ-LiNbO$_3$ developed in this thesis is based on approximating the velocity versus the angle of propagation curve (Figure 2.7) by an eighth-order polynomial [3]. A correction for the velocity change is then incorporated into the diffraction integral. The approach uses the concept of Fresnel integrals which are used to calculate diffraction patterns of subelements of a finger in an apodized transducer. The complete diffraction pattern is obtained by shifting and integrating this pattern over the output aperture. This approach minimizes the error introduced by moving the $\exp(-jko_0)$ term outside of the integral.
4.2 Diffraction Integral

The diffraction integral evaluates the complex field strength at a certain point \( P \) away from the radiating element, i.e., finger of a transducer (Figure 4.1).

**FIGURE 4.1:** The complex field strength at point \( P \) is evaluated by shifting the radiating element of length \( L \) along the \( x \)-axis from start to end.
The complex field at point P is expressed as [18]:

\[ E(x_1, \rho_0) = K \int_{\text{line}} A(x) \frac{e^{-jk\rho}}{\rho} \, dx \]  \tag{4.1}

where \( K \) = a constant along the \( x \)-axis, equal to unity
\( A(x) \) = acoustic distribution on the element,
\( A(x) = 1 \) for all \( x \)
\( k \) = the wave number = \( 2\pi/\lambda_0 \)

The distance from the incremental radiating element \( dx \) to the field point P is:

\[ \rho = \sqrt{\rho_0^2 + (x - x_1)^2} \] \tag{4.2}

From Figure 4.1 follows that \( x_1 = 0 \), hence:

\[ \rho = \sqrt{\rho_0^2 + x^2} \]

\[ = \rho_0 \sqrt{1 + \frac{x^2}{\rho_0^2}} \] \tag{4.3}

In general \( \rho_0 \) is very much greater than the array length, i.e., the finger length. Consequently, the denominator of (4.1) varies only slightly over the integral and may be replaced by \( \rho_0 \). The effect of the variation of \( \rho \) in the exponent is quite different, however, since \( \exp(-jk\rho) \) rotates 360° when the distance changes by one wavelength.

Expanding the expression under the square root in (4.3) in a binomial series and preserving only the dominant terms yields:

\[ [1 + \frac{x^2}{\rho_0^2}]^{\frac{1}{2}} = 1 + \frac{1}{2} \frac{x^2}{\rho_0^2} \] \tag{4.4}
Substituting (4.4) into (4.3) gives:

\[ \rho = \rho_0 + \frac{2}{\rho_0} \frac{x}{2\rho_0} \quad (4.5) \]

Hence (4.1) can be expressed as:

\[ E(x_1, \rho_0) = \int_{\text{line } \rho_0} \frac{1}{\rho_0} \exp \left( -jk \left( \rho_0 + \frac{2}{2\rho_0} \right) \right) dx \quad (4.6) \]

The limits of the integral are \( x - L/2 \) and \( x + L/2 \) from Figure 4.1.

Hence,

\[ E(x_1, \rho_0) = \frac{\exp \left( -jk \frac{x}{2\rho_0} \right) x + L/2}{\rho_0} \int_{x - L/2}^{x + L/2} \exp \left( -jk \frac{x}{2\rho_0} \right) dx \quad (4.7) \]

The term \( \exp \left( -jk \frac{x}{2\rho_0} \right) \) pertains to the near field radiation pattern, the Fresnel zone of diffraction theory, hence the expression the Fresnel integral. The Fresnel integral can be shown to asymptotically approach the far-field distribution [9]. Using the substitution \( t = \pi V^2 / 2 \), (4.7) becomes

\[ E(x_1, z) = \frac{\pi}{2} \frac{V_+^2}{\sqrt{2x}} \int \exp \left( -\frac{t}{2\pi} \right) dt \quad (4.8) \]

and the integration limits are

\[ v_+ = \frac{x + L/2}{\sqrt{2}/2} \quad v_- = \frac{x - L/2}{\sqrt{2}/2} \quad (4.9) \]
where the wavelength scaled parameters are adopted:

\[ \hat{L} = \frac{L}{\lambda_a} \quad \hat{Z} = \frac{z}{\lambda_a} \quad \hat{X} = \frac{X}{\lambda_a} \]

and \( \lambda_a \) is the acoustic wavelength at center frequency.

For each computation the following form of integral (4.8) is adopted:

\[
E(x_1, z) = C \left[ \int_0^{\frac{\pi}{2}} \frac{v^2}{2\pi t} \exp(-jt) \ dt - \int_0^{\frac{\pi}{2}} \frac{v^2}{2\pi t} \exp(-jt) \ dt \right] \quad (4.10)
\]

where \( C = \frac{\exp(-j2\frac{\pi}{2})}{\sqrt{2\pi}} \)

The derivation of the Fresnel integrals (4.8) and (4.10) is shown in Appendix I.

The above form of Fresnel integral is evaluated by the series expansion by J. Boersma [23]:

\[
f(x) = \int_0^x \frac{e^{-jt}}{\sqrt{2\pi t}} \ dt = C(x) - jS(x) \quad (4.11)
\]

For \( 0 < x < 4 \)

\[
f(x) = e^{-jx} \sqrt{\frac{x}{4}} \sum_{n=0}^{11} (a_n + jb_n) \left( \frac{x}{4} \right)^n \quad (4.11a)
\]

For \( x > 4 \)

\[
f(x) = \frac{1-j}{2} e^{-jx} \sqrt{\frac{4}{x}} \sum_{n=0}^{11} (c_n + jd_n) \left( \frac{4}{x} \right)^n \quad (4.11b)
\]

The values of coefficients \( a_n, b_n, c_n \) and \( d_n \) are listed in Appendix II.
4.3 Anisotropy of LiNbO₃

Now consider the anisotropy of the LiNbO₃ crystal (Figure 2.7). The velocity versus the angle of propagation curve is symmetrical about the pure mode axis 0°. It will be shown that this curve can be approximated by an eighth-order polynomial.

The LiNbO₃ crystal exhibits yet another kind of anisotropy [22]. The direction and velocity of the energy propagation do not coincide with the direction and velocity of the phase propagation (Figure 4.2).

![Acoustic beam trajectory in an anisotropic solid](image-url)
The wave phase fronts travel along the direction of \( \hat{k} \) which is normal to the transmitting transducer, but the wave packet energy travels in the direction \( \hat{v}_e \) which is at an angle to \( \hat{k} \). This means that the receiving transducer must be offset in order to intercept the acoustic pulse [37]. The angle between the two vectors has to be determined in order to get an accurate model for diffraction effects on lithium niobate crystals.

\[ \hat{v}_e \cdot d\hat{L} = 0. \quad [10] \]

The energy velocity vector is normal to the slowness surface (Figure 4.3(a)), the reciprocal of the phase velocity as a function of the angle of propagation. For an isotropic substrate the wave vector \( \hat{k} \) and the energy velocity \( \hat{v}_e \) point in the same direction.
FIGURE 4.4: Geometrical configuration for the power flow angle $\phi$

Referring to Figure 4.4 consider $\vec{k}(\Theta)$ for an angle $\Theta$ with respect to the z-axis, the pure mode axis of propagation for y-cut LiNbO$_3$. Now consider a small increment in $\Theta$ to $\Theta + \Delta \Theta$ and create a triangle between the tangent to the isotropic curve AB and the anisotropic curve CD along with $\vec{k}(\Theta + \Delta \Theta)$. In the limit as $\Delta \Theta \to 0$ the angle between $\vec{k}(\Theta + \Delta \Theta)$ and the isotropic tangent AB approaches 90°. The angle $\phi$ in the limit is defined by [28]:

$$\lim_{\Delta \Theta \to 0} \tan \phi = \frac{\Delta k(\Theta)}{\Delta \Theta \cdot k(\Theta)}$$  \hspace{1cm} (4.12)$$

thus $$\tan \phi = \frac{dk(\Theta)}{k(\Theta) \, d\Theta}$$  \hspace{1cm} (4.13)$$

By simple geometry it can be shown that the angle between $\vec{k}(\Theta)$ and $\vec{P}(\Theta)$ is also $\phi$, the power flow angle. The approximation for the phase velocity versus the angle of propagation is [3]:

$$v(\Theta) = v_o \left(1 + a\Theta^2 + b\Theta^4 + c\Theta^6 + d\Theta^8\right)$$  \hspace{1cm} (4.14)$$
where
\[ a = -1.26 \times 10^{-4} \]
\[ b = 3.6892 \times 10^{-7} \]
\[ c = -5.0887 \times 10^{-10} \]
\[ d = 2.7557 \times 10^{-13} \]

These values of a, b, c and d coefficients give the best fit to the phase velocities between -30° to +30° reported by Szabo and Slobodnik [22].

Now consider
\[ k(0) = \frac{\omega}{v(0)} \]  \hspace{1cm} (4.15)

Going back to (4.14) and using (4.13) and (4.15)
\[ \frac{dk(0)}{d\theta} = \frac{d}{d\theta} \left[ \frac{\omega}{v_0 (1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8)} \right] \]
\[ = \frac{\omega}{v_0} \left[ \frac{1}{(1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8)} \right]^2 \cdot \]
\[ - (2a\theta + 4b\theta^3 + 6c\theta^5 + 8d\theta^7) \]  \hspace{1cm} (4.16)

Thus substituting (4.16) as well as (4.14) and (4.13) into (4.14) yields:
\[ \tan \phi = \frac{1}{k(0)} \frac{dk(0)}{d\theta} \]
\[ = \frac{v(0)}{\omega} \frac{dk(0)}{d\theta} \]
\[ = \frac{v_0 (1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8)}{\omega} \cdot \]
\[ \frac{\omega}{v_0} \left[ \frac{1}{(1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8)} \right]^2 \cdot \]
\[ - (2a\theta + 4b\theta^3 + 6c\theta^5 + 8d\theta^7) \]
\[ \tan \phi = \frac{-(2a\theta + 4b\theta^3 + 6c\theta^5 + 8d\theta^7)}{(1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8)} \]  \hspace{1cm} (4.17)
Result (4.17) agrees with the expression by Crabb [25], hence:

\[ \phi = \tan^{-1} \left( \frac{1}{\nu(\theta)} \frac{dv(\theta)}{d\theta} \right) \]  \hspace{0.5cm} (4.18)

where \( \phi \) is the "power flow angle", the angle between the phase vector \( \vec{k}(\theta) \) and the energy flow vector \( \vec{F}(\theta) \) of Figure 4.4.

\[ \text{FIGURE 4.5: Propagation velocities on the surface of YZ-LiNbO}_3 \text{ crystal} \]

From Figure 4.5 it can be seen that energy propagates at an angle \( (\theta - \phi) \) on LiNbO\(_3\) substrates. The energy velocity vector can be resolved into two components \( \vec{v}_p \) and \( \vec{v}_l \), where \( \vec{v}_p \) is equal to the plane wave phase velocity. Since the phase fronts for an infinitely wide wave front would remain constant, one can conclude from Figure 4.5 that:

\[ v_g(\theta) = v_e(\theta) = \frac{v_p(\theta)}{\cos \phi} \]  \hspace{0.5cm} (4.19)

or \[ v_p(\theta) = v_e(\theta) \cos \phi \]  \hspace{0.5cm} (4.19a)

and \[ \lambda_p = \lambda_g \cos \phi \]  \hspace{0.5cm} (4.19b)
4.4 Diffraction Integral for LiNbO$_3$

The approach outlined here is based on the analysis of Cohen [26].

Since it is the energy that has to be intercepted by the output transducer, the velocity and direction of propagation of the energy vector $\mathbf{v}_e$ must be considered.

The dependence of the phase velocity $v_p$ on the angle of propagation $\Phi$ was stated in (4.14) as:

$$v_p = v_0 (1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8)$$

now $v_e = \frac{v_p}{\cos\Phi}$

$$= \frac{v_0}{\cos\Phi} (1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8) \quad (4.20)$$

and $k_e = \frac{\omega}{v_e}$

$$= \frac{\omega}{v_0} \cdot \frac{\cos\Phi}{1 + a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8} \quad (4.21)$$

where $\frac{\omega}{v_0} = k_0$.

For small values of $a\theta^2$, $b\theta^4$, $c\theta^6$, $d\theta^8$ the small number approximation can be made as follows:

$$k_e \equiv k_0 \cos\Phi (1 - a\theta^2 - b\theta^4 - c\theta^6 - d\theta^8) \quad (4.22)$$

Let $(- a\theta^2 - b\theta^4 - c\theta^6 - d\theta^8) = \gamma$

hence $k_e \equiv k_0 \cos\Phi (1 + \gamma) \quad (4.23)$
Now one must modify the Fresnel integral to accommodate the energy velocity parameters.

![Diagram showing geometrical configuration for evaluating the effect of power flow angle \( \phi \) on the complex field strength at point P.]

**FIGURE 4.6:** Geometrical configuration for evaluating the effect of power flow angle \( \phi \) on the complex field strength at point P.

From Figure 4.6 it is apparent that

\[
\tan \left( \theta - \phi \right) = \frac{X_1}{\rho_0} \quad (4.24)
\]

where \( \phi \) is a negative angle

\[
hence \quad X_1 = \rho_0 \tan \left( \theta - \phi \right) \quad (4.24a)
\]

\( X_1 \) now has to appear in the limit of the integral as \( X_1 \to L/2 \).

Hence substituting the energy velocity parameters into (4.7) yields:

\[
E(x, \rho_0) = \frac{\exp \left( -jk \frac{x}{\rho_0} \right)}{\rho_0} \int_{\frac{X_1-L/2}{X_1+L/2}} \exp \left[ -jk \frac{x^2}{2\rho_0} \right] dx \quad (4.25)
\]
Transforming the above Fresnel integral into its classical form where \( z = z_0 \) and using acoustic wavelength scaled parameters:

\[
E(x, z) = \exp\left(-4z^2 \cos^2(\lambda + \gamma)\right) \frac{\pi}{2} \frac{v^2}{\sqrt{2 \pi \cos^2(1 + \gamma)}} \int_{-\infty}^{\infty} \exp\left(-\frac{t^2}{2}\right) dt
\]

(4.26)

where the limits \( v_- \) and \( v_+ \) are:

\[
v_+ = \frac{x_1 + \frac{L}{2}}{\sqrt{\frac{Z}{2}}} \left[\cos^2(1 + \gamma)\right]^\frac{1}{n}
\]

(4.26a)

\[
v_- = \frac{x_1 - \frac{L}{2}}{\sqrt{\frac{Z}{2}}} \left[\cos^2(1 + \gamma)\right]^\frac{1}{n}
\]

(4.26b)

The step by step derivation of these results is given in Appendix 3.
4.5 Modelling of Diffraction [3]

The effects of diffraction of the output apodized transducer are the dominant ones. The diffraction due to the input non-apodized transducer is of lesser importance. The approach taken here assumes that the diffraction pattern due to the fingers of the apodized IDT will vary only a little from finger to finger of the input non-apodized IDT. Hence the diffraction pattern of each finger of the output IDT is computed at the center of the non-apodized IDT (Figure 4.7).

![Diagram of IDT](image)

**FIGURE 4.7:** The diffraction pattern of each finger of the output transducer is evaluated at the center of the input transducer.

Each complex field point of the diffraction pattern is computed using the Fresnel integral, i.e., if 100 points are required across the output aperture then 100 Fresnel integrals must be solved. In order to get the final complex finger weighting the values of the field points across the whole aperture have to be totalled, i.e., integrated. This yields one complex value for each finger of the output IDT. However, the phase of each finger has to be corrected for by subtracting the phase shift between each finger and the center of the input IDT. Hence, the complex weights as computed at the center of the input IDT are placed back at their original position in the output apodized IDT.
The next step is the complex numerical convolution between the input and output IDT to obtain the filter's complex impulse response. Applying the FFT for complex numbers to the impulse response yields the filter's frequency response with the effects of diffraction included.

For the anisotropic case the modelling becomes more complicated due to the introduction of the variation in phase velocity and power flow angle with the direction of propagation.

The term outside the integral of (4.26) is the phase delay term. There is an error in this delay time for long transmitting fingers. It is assumed that this delay is approximately constant for the limits of integration, i.e., between the center of the finger and the extreme ends. For long transmitting fingers this assumption is no longer true and a significant phase error is encountered. To avoid this problem the long finger is subdivided into segments no longer than 2 wavelengths each (Figure 4.8). For distances of \( z \approx 100 \lambda_0 \), the width of \( L = 2\lambda_0 \) gives a maximum phase error of approximately \( \pm 20^\circ \) on YZ-LiNbO$_3$ [28].
FIGURE 4.8: A composite diffraction profile is obtained by adding the contribution of the diffraction profiles of the individual subelements.

From Figure 4.8, it follows that the delays D1 to D4 are equal. Thus adding up the contributions of the individual segments P1 to P4 over the output aperture width, \(-L/2\) to \(+L/2\), will yield the correct finger weight after diffraction. The rest of the process is the same as for the isotropic case: 1) shift the diffracted weights back to their original position; 2) convolve the two IDT’s; and 3) take the FFT of the impulse response to get the desired frequency response.
Finally, compare the two models with the experimental results (Figure 4.9, 4.10). There is a discrepancy in both cases between the model and experiment on the high frequency side. This is due to the coupling of the surface wave with the shear bulk wave [13]. This coupling peaks at about 77 MHz which explains the loss of the first null of the \( \sin \pi x / x \) at 77.3 MHz.

The anisotropic model appears to be the better one since the individual nulls follow the experimental results more closely.
FIGURE 4.9: The isotropic model versus experimental results of 70 FIL02

FIGURE 4.10: The anisotropic model versus experimental results of 70 FIL102
CHAPTER V

COMPENSATION ALGORITHM

5.1 Introduction

Compensation for the effects of diffraction can be accomplished in the time domain or in the frequency domain. In this thesis the problem has been approached via time domain considerations [28]. The diffraction compensation algorithm has been based on the work of Mitchell and Parker [29] and [30, 31, 32, 33, 34].

A single transmitting finger can be decomposed into two split fingers which are separated spatially by a quarter wavelength as shown in Figure 5.1. This arrangement offers an additional degree of freedom in the filter design.

Having calculated the diffraction weights for these split fingers and comparing them to the original transmitted weight, a phase, and an amplitude error is noted. The amplitude correction is achieved by adjusting the finger overlaps of both split fingers simultaneously until the diffracted weights take on the desired value. The compensation for the phase error is accomplished by adjusting the relative amplitudes of the split fingers.
FIGURE 5.1: (a) Single electrode structure IDT  
(b) Double electrode (split finger) structure IDT

The successive fingers in the single-electrode IDT are separated spatially by a half wavelength while the successive fingers in the double electrode IDT are separated spatially by a quarter wavelength.
5.2 Development of the Diffraction Compensation Algorithm [28]

The transmitted single vector sample TW (Figure 5.2) is decomposed into the left split finger TWL and the right split finger TWR. The two split fingers are 90 degrees out of phase with respect to each other. The left split finger TWL is leading TW by 45 degrees while the right split finger TWR is lagging TW by 45 degrees. At a certain observation point, i.e., the center of the receiving transducer, the change in the transmitted sample's weighting due to diffraction has to be evaluated (Figure 5.3).

From Figure 5.3 it follows that both samples TWL and TWR travel the same distance \( R_0 \), when their change in weighting due to diffraction is evaluated. The new sample weights are DWL for the left split finger and DWR for the right one. To obtain the single weight DWSN, which represents TW after diffraction, the contributions of DWL and DWR have to be added together. Due to diffraction there is a phase offset \( \phi_k \) between TW and DWSN as well as a difference in their respective magnitudes. In order to compensate for these discrepancies, the phasor CWSN is transmitted in place of the phasor TW. The phasor CWSN is the complex conjugate of DWSN such that after diffraction, approximately TW is received. CWSN is decomposed into the left and right component CWL and CWR respectively. These two phasors represent the new weighting of the split fingers TWL and TWR, originally designed to transmit TW.

From the trigonometry of Figure 5.2 the relationships between all the phasors under consideration are derived.

\[
\begin{align*}
\text{DWL} &= |\text{DWL}| \left(45^\circ + \phi_k\right) \quad (5.1a) \\
\text{DWR} &= |\text{DWR}| \left(\phi_k - 45^\circ\right) \quad (5.1b)
\end{align*}
\]
FIGURE 5.2: Phasor diagram of transmitted and diffracted weights.
TW - original transmitted sample
DWSN - received sample after diffraction
CWSN - transmitted sample with diffraction correction included
FIGURE 5.3: Evaluation of the diffracted weights DWL and DWR of the split transmitting fingers TWL and TWR.

Since the magnitudes of DWL, and DWR are equal, a common term can be used.

\[ |DWL| = |DWR| = |DW| \]

The magnitude of \(|DW|\) relates to the magnitude of DWSN in the following way:

\[ |DW| = |DWSN| / \sqrt{2} \tag{5.2} \]

To generate the single phasor DWSN, the contributions of DWL and DWR have to be summed:

\[ DWSN = DWL + DWR \]

\[ = |DW| 45^\circ + \phi_k + |DW| \phi_k - 45^\circ \tag{5.3} \]
Performing this summation in complex form yields:

\[
\overline{\text{DWSN}} = |D_W| \left[ \cos (45^\circ + \phi_k) + j\sin (45^\circ + \phi_k) \\
+ \cos (\phi_k - 45^\circ) + j\sin (\phi_k - 45^\circ) \right] \tag{5.4} \\
= |D_W| [2\cos 45^\circ \cos \phi_k + j2\cos 45^\circ \sin \phi_k]
\]

but \( |D_W| = \overline{|\text{DWSN}|} / \sqrt{2} \) from equation (5.2)

hence:

\[
\overline{\text{DWSN}} = \overline{|\text{DWSN}|} [\cos \phi_k + j\sin \phi_k] \tag{5.5}
\]

The result (5.5) agrees with Figure 5.2.

Next, derive the expression for \( \overline{\text{CW}} \) and \( \overline{\text{CW}} \). Their sum must be the complex conjugate of \( \overline{\text{DWSN}} \).

\[
\overline{\text{CWSN}} = \overline{\text{CW}} + \overline{\text{CW}} \\
= |\text{CW}| +45^\circ + |\text{CW}| -45^\circ
\]

Find the expression for \( |\text{CW}| \) and \( |\text{CW}| \) from Figure 5.2.

\[
|\text{TW}| = |\text{CW}| \cos 45^\circ \tag{5.7a}
\]

and \( |\text{TW}| = |\text{CW}| \cos (45^\circ + \phi_k) \) \( \tag{5.7b} \)

equating (5.7a) and (5.7b) yields:

\[
|\text{CW}| = |\text{DW}| \frac{\cos (45^\circ + \phi_k)}{\cos 45^\circ} \tag{5.7c}
\]

Substituting for \( |\text{DW}| \) expression (5.2)

\[
|\text{CW}| = |\text{DWSN}| \cos (45^\circ + \phi_k) \tag{5.8}
\]
Similarly for $|CWR|$:

$$TW_2 = |CWR| \cos (-45^\circ) \tag{5.9a}$$

$$TW_2 = |DWR| \cos (\phi_k - 45^\circ) \tag{5.9b}$$

hence

$$|CWR| = |DWR| \frac{\cos (\phi_k - 45^\circ)}{\cos (-45^\circ)} \tag{5.9c}$$

and

$$|CWR| = |DWSN| \cos (\phi_k - 45^\circ) \tag{5.10}$$

Substituting (5.8) and (5.10) into (5.6)

$$\overline{CWSN} = |DWSN| \left[ \cos (45^\circ + \phi_k) (\cos 45^\circ + j\sin 45^\circ) 
+ \cos (\phi_k - 45^\circ) (\cos 45^\circ - j\sin 45^\circ) \right]$$

$$= |DWSN| (\cos \phi_k - j\sin \phi_k) \tag{5.11}$$

Equation (5.11) is the complex conjugate of equation (5.5) and hence

$$\overline{CWSN} = DWSN^*.$$ 

The phase correction for diffraction is implemented by using expressions (5.8) and (5.10) which adjust the relative length of the left and right split fingers. The magnitude correction is done by simple subtraction of the desired and received sample magnitude:

$$|CWL| = |CWL| \left[ 1 + |TW| - |DWSN| \right] \tag{5.12}$$

5.3 Computer algorithm

The flow graph of the computer compensation algorithm is shown in Figure 5.4. It outlines the steps in the sequence they were taken, however, not all the details are included.
FIGURE 5.4: Flow graph of diffraction compensation algorithm.
First, the maximum number of iterations ITMAX is established to avoid excessive computing time. The magnitude relaxation factor RLXM and the phase relaxation factor RLXP are also initiated as well as the minimum acceptable amplitude error. All the transmitting fingers TW are generated and then each one of them is decomposed into the left and right split fingers TWL and TWR respectively (Figure 5.2). Next, the diffracted weights DWL and DWR of the split fingers TWL and TWR are computed at the center of the output transducer.

The diffracted weight sum normalized DWSN is then obtained as the phasor addition of DWL and DWR and the amplitude and phase of DWSN are compared to those of TW. The error in magnitude ERRLIN and the error in phase ERRPHS are hence established for each iteration. Once the amplitude and phase error reach a sufficiently low value, e.g., the amplitude error should be less than 10^{-6} and the phase error one degree maximum, or the number of allowable iterations is exceeded, the impulse response and the frequency response of the compensated design are generated.

The predicted diffraction compensated frequency response based on the isotropic model is shown in Figure 5.5 and the one based on the anisotropic model is shown in Figure 5.6. The isotropic compensation yields sidelobe levels of -53 dB while the anisotropic compensation indicates -46 dB. The anisotropic compensation algorithm diverged after the fifteenth iteration and the frequency response shown in Figure 5.6 is the best one obtained before divergence. Further refinement of the numerical techniques is necessary to improve the compensated frequency response based on the anisotropic model.

However, the evaluation of the two compensated designs must be based on experimental results given in Chapter VI.
FIGURE 5.5: The predicted compensated frequency response based on the isotropic model compared to the Dirac delta model.

FIGURE 5.6: The predicted compensated frequency response based on the anisotropic model compared to the Dirac delta model.
CHAPTER VI

TESTING AND EXPERIMENTAL RESULTS

6.1 Time Domain Measurements

A measurement set-up is described here which enables time domain testing of a SAW device's pulse response [35] to be performed. The same set-up isolates spurious time domain signals from the swept frequency amplitude and phase measurements, hence, the effects of diffraction on the frequency response are isolated.

The test set consists of a network analyzer and two time gates, one at the input and one at the output of the SAW filter as shown in Figure 6.1. The time gates each contain a pulse generator and a balanced mixer which are used to pulse modulate the input and the output signals of the device under test.

![Block diagram of the test set for pulse response measurements](image)

*FIGURE 6.1: Block diagram of the test set for pulse response measurements*
The step-by-step description of the time domain processor follows.

The RF output of the tracking generator generates a 70 MHz sinewave (Figure 6.2(a)). Time gate 1 sends out a narrow pulse (Figure 6.2(b)), approximately 12 RF cycles wide, which multiplies with the RF signal and the final result is an RF pulse (Figure 6.2(c)) which excites the SAW filter's impulse response. The pulse of time gate 2 is delayed with respect to that of time gate 1 as can be seen in Figure 6.2(e). The amount of delay corresponds directly to the time response of the SAW filter (Figures 6.2(d) and (f)). The position of the second pulse is such as to isolate the desired SAW impulse response, the TTE or reflections. The output of time gate 2 is connected to the RF input of the spectrum analyzer and the frequency domain response is observed.
FIGURE 6.2: Time gated frequency response measurement

a) RF output from tracking generator
b) Time gate 1
c) Input signal to the test sample
d) Output signal from the test sample
e) Time gate 2
f) RF input to the spectrum analyzer
g) Signal evaluated in the spectrum analyzer
6.2 Practical Implementation

In order to realize the block diagram of Figure 6.1 some practical aspects of the set-up had to be considered. Pulse generator 1 has to be triggered externally from the tracking generator so that the pulse generator is synchronized. Pulse generator 2 is also triggered externally but from pulse generator 1. Hence the source and the two pulse generators are in synchronism. The pulse rate of generator 1 has to allow an adequate amount of time for pulse 2 to window on to the desired SAW response (Figure 6.2(f)). If the pulse rate were 70 MHz then the separation of pulses would be 0.014 µs. But the TTE is 6.26 µs delayed from the instant the time domain response is excited. If the pulse rate were to remain 70 MHz the window of pulse 2 could not be applied. Hence the 70 MHz RF signal has to be divided down to allow for the full SAW time domain response including the TTE to be observed between pulses. A division by 1000 satisfies this requirement. The pulse rate is then 70 kHz which corresponds to 14.3 µs between pulses. The block diagram is now like that of Figure 6.3. There is a 3 dB power splitter at the output of the tracking generator which adjusts the power level to the frequency divider consisting of the Emitter Coupled Logic (ECL) gates. Each gate is a MC10137 decade counter. Three of these will be connected sequentially in order to achieve the desired division count (Appendix IV). The mixers used are HP10514A and HP10534A balanced mixers and for better isolation between the local and RF ports two mixers are connected in series. There is a 20 dB C-COR 3580 amplifier both at the input and at the output of the SAW filter. Both the input pulse and the output of the SAW filter are displayed on the oscilloscope. By moving the window of pulse 2 and adjusting its width (Figure 6.4) the SAW impulse response is isolated from the secondary reflections and its frequency response is observed on the spectrum analyzer (Figure 6.7).
FIGURE 6.3: Block diagram of RF pulse measurement technique to isolate main signal from spurious coupling in SAW filter.

FIGURE 6.4: Delay of the different signals of a 70 MHz SAW filter.
6.3 Experimental Results

In this section the experimental results are presented. The mask of one of the devices fabricated is shown in Figure 6.5. The bars around the outside are for alignment of the mask on the wafer. They also serve as a guideline for the sawing of the wafer into the individual dice. The two diagonal bars between the two transducers are electrically grounded in order to reduce the electromagnetic feed-through.

FIGURE 6.5: Mask of a 70 MHz SAW filter
6.3.1 Diffraction Un-compensated Filter

Figure 6.6(a) shows the frequency response of the 70 MHz filter. The vertical scale is 10 dB/div and the sweep is 70 ± 25 MHz. All of the spurious signals are included. Figure 6.6(b) depicts the significant ripples due to the reflections from the end of substrate in the passband of the device. The vertical scale is 2 dB/div and the sweep about \( f_0 \) = 70 MHz is 10 MHz.

Figures 6.7(a) and (b) present the frequency response of the 70 MHz filter without the spurious interference. The window of pulse 2 is positioned exactly as indicated in Figure 6.4, hence, the response of the main SAW signal has been isolated from the delayed reflections and the TTE. Diffraction raises the level of the side-lobes (see Figure 1.4) and reduces the depth of the nulls of the sinc function due to the non-apodized transducer.

There is a significant slope in the passband of the 70 MHz filter fabricated on the YZ-LiNbO\(_3\) substrate (Figure 6.7(b)). Figure 6.7(a) shows a noticeable difference between the level of the sidelobes at the high frequency side of the passband and the low frequency one. The sidelobes at the high frequency side are quite a bit higher than those at the low frequency side.

The cause of this asymmetry was established to be due to the strong coupling with the surface shear bulk wave [13,14,15]. This shear bulk wave travels at a velocity only 3 percent higher than the center frequency of the filter. The increase in the coupling level causes the slope in the passband to increase with frequency and the high frequency sidelobe level to increase. The bulk wave signal arrives at almost the same delay as the main SAW signal and therefore cannot be therefore isolated by the window of pulse 2. The only way to avoid it is to use a different crystal cut of LiNbO\(_3\), for example the 128° rotated Y-cut.
FIGURE 6.6(a): The measured frequency response of the 70 FIL102 device with spurious signal interference

FIGURE 6.6(b): The measured passband response of the 70 FIL102 device showing ripples due to end of substrate reflections
FIGURE 6.7(a): The measured frequency response of the 70 FIL102 device with spurious interference removed.

FIGURE 6.7(b): The measured passband response of the 70 FIL102 device showing response without ripples from end of substrate reflections.
Another diffraction un-compensated filter 70 FIL102 was fabricated (Figure 6.8), but this time the substrate bottom was roughened in order to disperse the bulk wave. Note the improvement in the shape of the sidelobes as compared to Figure 6.6(a) and the depth of the nulls due to the un-apodized transducer. However, the bulk wave coupling is still degrading the frequency response of 70 FIL102 in the passband and at the high frequency side. The intensity of the bulk wave is shown in Figure 6.9. The surface of the substrate has been coated with viscous material which absorbs the Rayleigh wave, hence only the bulk wave response is observed on the network analyzer. The upper frequency cutoff is due to the backside roughening. The bulk wave frequency response peaks at about 77 MHz where it is only 40 dB below the main SAW signal.

This strong bulk wave coupling masks the first upper null of 70 FIL102 which occurs at 77.3 MHz. The second null is at 84.6 MHz where the bulk wave coupling is 60 dB below the main SAW response, hence there is a well-pronounced deep null at that frequency.

6.3.2 Diffraction Compensated Filter

Three types of diffraction compensated filters were fabricated. Two are based on the anisotropic diffraction model (Figures 6.11 and 6.12) and one is based on the isotropic diffraction model (Figure 6.10). All three filters show an improvement in the passband and in the sidelobe suppression when compared to the un-compensated design (Figure 6.8). The filters of Figures 6.10 and 6.11 have a 30 degree incline in the apodized IDT. The filter in Figure 6.12 has no incline in the apodization. All three filters show the loss of the first upper null due to bulk wave coupling. The filter 70 FIL201 which was designed according to the isotropic model shows an 8 dB improvement in the first sidelobe suppression. 70 FIL202, the diffraction compensated filter based on the anisotropic model indicates a 5 dB improvement in the first sidelobe level. 70 FIL203
FIGURE 6.8: The measured frequency response of the filter 70 FIL102 with the substrate bottom roughened to disperse the bulk wave.

FIGURE 6.9: The measured bulk wave coupling affecting the frequency response of filter 70 FIL102.


FIGURE 6.10: The measured frequency response of the diffraction compensated filter based on the isotropic diffraction model (70 FIL201).

FIGURE 6.11: The measured frequency response of the diffraction compensated filter based on the anisotropic diffraction model (70 FIL202).
FIGURE 6.12: The measured frequency response of the diffraction compensated filter based on the anisotropic diffraction model (70 FIL203). There was no incline in the apodized IDT.

has the best frequency response of all the diffraction compensated filters. The first sidelobes of 70 FIL203 are 9 dB below the level of the un-compensated 70 FIL102.

The shape of the passband of 70 FIL203 appears to be slightly different from that of 70 FIL201 and 70 FIL202. However, the nulls are at exactly the same frequencies. The change in the passband characteristic is attributed to the fact there is no incline in the design of the apodized transducer. Due to lack of time there was no diffraction un-compensated filter with no incline fabricated for comparison.

The diffraction compensated filter 70 FIL202, which is based on the anisotropic model, had an error in the mask layout. There is a slight discrepancy in the magnitude of the finger overlaps along the
apodization incline and there is a good chance of an improved sidelobe suppression if this error is corrected.
CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

This thesis contains a basic treatment of the diffraction effects which are common to surface acoustic wave devices. The study has been carried out at both the theoretical and experimental levels, with good agreement.

The experimental results, however, do not yield a clear indication as to which diffraction compensation algorithm is more effective – the isotropic one or the anisotropic one. The anisotropic diffraction model is thought to be far superior to the isotropic one but the performance of the compensated filter based on the anisotropic model was anticipated to be better than what the experimental results indicate.

At present, an interesting problem arises when using the anisotropic model in the diffraction compensation algorithm, namely, the algorithm diverges after the fifteenth iteration. This causes the predicted first sidelobe level to be at only -46 dB and the experimental results indicate -41 dB for 70 FIL 203. Compared to this predicted frequency response the experiment is in good agreement.

The predicted frequency response for the isotropic diffraction correction indicates sidelobe levels of -53 dB but the experiment shows the first sidelobes at -40 dB. From this point of view the anisotropic compensated design is more successful and further refinement of the numerical techniques may bring significant improvements in the experimental results based on the anisotropic model.
A diffraction model based on the understanding of the physics of the SAW propagation behaviour on a highly anisotropic crystal such as YZ-LiNbO₃ has been developed. The novel approach taken to modelling the velocity versus the angle of propagation characteristic of YZ-LiNbO₃ with an 8th-order polynomial is considered successful. The experimental results indicate a 9 dB improvement in sidelobe levels for YZ-LiNbO₃.
7.2 Future Work

Future work in the area of this thesis may include some, or all of the following points.

1. A comprehensive analysis of the velocity profile of the YZ-LiNbO_3 substrate is a worthwhile task to be undertaken. The velocities used are based on those published by Szabo and Slobodnik in 1973 [24]. It is felt that with today's technology more accurate velocities for YZ-LiNbO_3 may be obtained.

2. An improvement of the diffraction model, by including the effect of the velocity change as the wavefront propagates across the metallized and unmetallized parts of the transducer, is recommended.

3. To further improve the diffraction model, the diffraction profiles may be generated at each receiving finger rather than just at the center of the receiving transducer. A more complex convolutional process will have to be implemented but the procedure will remain analogous to that outlined in Chapter IV.

4. To establish the diffraction model and the diffraction compensation algorithm for YZ-LiNbO_3, developed in this thesis, as a solid benchmark further experimental results must be obtained for a variety of distances between transducers and for a number of different aperture widths as well as transducer lengths.

5. There is a great deal of room left for improving the compensation time for the diffraction model. This can be achieved by using advanced numerical techniques and structured
programming. In the diffraction compensation algorithm there can be a significant saving in computation time achieved by developing a more efficient algorithm which will reduce the number of iterations before convergence.

6. A major contribution towards improving the filter's frequency response can be made by developing a design algorithm to counteract the effects of shear bulk wave coupling on the high frequency side of the passband. This may be accomplished by modifying the design algorithm presented in Chapter III.

7. Finally, it is felt that the techniques developed in this thesis for diffraction compensation on YZ-LiNbO$_3$ may be extended to other cuts of LiNbO$_3$ or to other substrate materials. It may be an interesting task to investigate the validity of this suggestion both theoretically and experimentally for a number of different substrate materials.
References


APPENDIX I

FRESNEL INTEGRAL

Consider the Fresnel integral,

\[ E(x_1, \rho_o) = \frac{\exp(-jk\rho_o)}{\rho_o} \int_{x-L/2}^{X+L/2} \exp\left(-j\frac{x}{2\rho_o}\right) dx \]  

where \( k = \frac{2n}{\lambda} \). Substituting for \( k \) into (1) yields:

\[ E(x_1, \rho_o) = \frac{\exp(-jk\rho_o)}{\rho_o} \int_{x-L/2}^{X+L/2} \exp\left(-j\frac{2n}{\lambda} \cdot \frac{x^2}{2\rho_o}\right) dx \]

\[ = \frac{\exp(-jk\rho_o)}{\rho_o} \int_{x-L/2}^{X+L/2} \exp\left(-j\frac{x^2}{\lambda \rho_o}\right) dx \]  

(2)

Let \( \frac{\sqrt{2}}{2} = \frac{x^2}{\lambda \rho_o} \)

or \( x = \sqrt{\frac{\lambda \rho_o}{2}} \) \( v \) (note \( \rho_o \equiv z \))  

(3)

Then \( \frac{2vdv}{2} = \frac{2xdx}{\lambda \rho_o} \)

\[ dx = \frac{\lambda \rho_o}{2} \frac{vdv}{x} \]

Using (3): \( dx = \frac{\lambda \rho_o}{2} \frac{vdv}{\sqrt{2}} \) \( \frac{\sqrt{\lambda \rho_o}}{2} \) \( dv \)

(4)
Substituting (4) into (2) gives:

\[
E(x_1, \rho_o) = \frac{\exp(-jk \rho_o)}{\rho_o} \int_\text{line} \exp(-j\frac{v^2}{2}) \, dv \sqrt{\frac{\lambda a \rho_o}{2}}
\]

\[
= \frac{\exp(-jk \rho_o)}{\rho_o} \sqrt{\frac{\lambda a \rho_o}{2}} \int_\text{line} \exp(-j\frac{v^2}{2}) \, dv
\]

(5)

Now transform the limits using (3)

When \( x = x + L/2 \)

\[
\sqrt{\frac{\lambda a \rho_o}{2}} v = x + L/2
\]

or \( v_+ = \frac{x + L/2}{\sqrt{\lambda a \rho_o}} = \frac{x + L/2}{\sqrt{\lambda a z/2}} (\rho_o \equiv z) \)

Similarly, the lower limit is derived as

\[
v_- = \frac{x - L/2}{\sqrt{\lambda a z/2}}
\]

and integral (5) can be put in the form:

\[
E(x_1, z) = \exp(-jkr) \sqrt{\frac{\lambda a x}{2}} \int_{v_-}^{v_+} \exp(-j\frac{v^2}{2}) \, dv
\]

(6)
For convenience acoustic wavelength scaled parameters are adopted:

\[ \hat{L} = L/\lambda_a \quad \hat{Z} = z/\lambda_a \quad \hat{X} = x/\lambda_a \]  

(7)

So that the integration limits of (6) become

\[ v_+ = \frac{\hat{X} + \hat{L}/2}{\sqrt{2}/2} \quad v_- = \frac{\hat{X} - \hat{L}/2}{\sqrt{2}/2} \]  

(8)

and the integral (6) becomes

\[ E(x_1, z) = \exp \left( -j2\pi \hat{Z} \right) \sqrt{\frac{\hat{X}}{2\hat{Z}}} \int_{v_-}^{v_+} \exp \left( -jx_1 \frac{v^2}{2} \right) dv \]  

(9)

One further transformation is needed to put (9) into the form required for the numerical evaluation of the Fresnel integral (ref. Appendix II).

From (9), let

\[ C = \frac{\exp \left( -j2\pi \hat{Z} \right)}{\sqrt{\frac{\hat{X}}{2\hat{Z}}}} \]

and (9) is rewritten in the following form:

\[ E(x_1, z) = C \int_{0}^{v_+} \exp \left( -jx_1 \frac{v^2}{2} \right) dv - C \int_{0}^{v_-} \exp \left( -jx_1 \frac{v^2}{2} \right) dv \]  

(10)

Now apply the transformation

\[ t = \frac{v^2}{2} \]

or

\[ v = \sqrt{\frac{2t}{x_1}} = \sqrt{\frac{2}{x_1}} \cdot \hat{t} \]

for which

\[ dv = \frac{dt}{\sqrt{2\pi t}} \]
The limits are transformed according to:

\[ v = v_+ + t_+ = \frac{v_+^2}{2} \]

when \( v = v_+ + t_+ \),

\[ v = v_- + t_- = \frac{v_-^2}{2} \]

and \( v = v_- + t_- \),

hence (10) becomes

\[
E(x_1, z) = C \int_0^{t_+} \frac{\exp(-jt)}{\sqrt{2\pi t}} \, dt - C \int_0^{t_-} \frac{\exp(-jt)}{\sqrt{2\pi t}} \, dt
\]

(11)

The above form of Fresnel integral (11) can be evaluated by the series expansion listed in Appendix II.
APPENDIX II

COMPUTATION OF FRESNEL INTEGRALS [23]

Two approximations, one valid for \( x \) less than 4 and the other valid for \( x \) larger than 4, are listed here for the Fresnel integrals defined in the form:

\[
f(x) = \frac{x}{\sqrt{2\pi t}} \int_{0}^{x} e^{-jt} dt = C(x) - jS(x)
\]

These approximations are the following:

(1) For \( 0 < x < 4 \)

\[
f(x) = e^{-\frac{x}{2}} \int_{\frac{x}{4}}^{1} \sum_{n=0}^{11} \left( a_n + jb_n \right) \left( \frac{x}{4} \right)^n
\]

(2) For \( x > 4 \)

\[
f(x) = \frac{1-\frac{x}{2} + e^{-\frac{x}{2}} \int_{\frac{x}{4}}^{1} \sum_{n=0}^{11} \left( c_n + jd_n \right) \left( \frac{x}{4} \right)^n}
\]

The numerical values of the coefficients \( a_n, b_n, c_n \) and \( d_n \) are given by:

<table>
<thead>
<tr>
<th>( a_0 )</th>
<th>( b_0 )</th>
<th>( c_0 )</th>
<th>( d_0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.595769140</td>
<td>-0.000000033</td>
<td>0.0</td>
<td>0.199471140</td>
</tr>
<tr>
<td>-0.000001702</td>
<td>4.25387524</td>
<td>-0.024933975</td>
<td>-0.00000023</td>
</tr>
<tr>
<td>-0.80856854</td>
<td>-0.000092810</td>
<td>-0.0003936</td>
<td>-0.009351341</td>
</tr>
<tr>
<td>-0.000576361</td>
<td>-7.780020400</td>
<td>-0.003770956</td>
<td>-0.00023006</td>
</tr>
<tr>
<td>6.920691902</td>
<td>-0.009520895</td>
<td>0.00689892</td>
<td>0.004851466</td>
</tr>
<tr>
<td>-0.016898657</td>
<td>-5.075161298</td>
<td>-0.009497136</td>
<td>-0.001903218</td>
</tr>
<tr>
<td>-3.050485660</td>
<td>-0.138341947</td>
<td>-0.011948809</td>
<td>-0.017122914</td>
</tr>
<tr>
<td>-0.075752419</td>
<td>-1.363729124</td>
<td>-0.006748873</td>
<td>-0.029064067</td>
</tr>
<tr>
<td>0.850663781</td>
<td>-0.403349276</td>
<td>-0.000246420</td>
<td>-0.027928955</td>
</tr>
<tr>
<td>-0.025639041</td>
<td>-0.702222016</td>
<td>0.002102967</td>
<td>0.016497308</td>
</tr>
<tr>
<td>-0.150239060</td>
<td>-0.216195929</td>
<td>0.001217930</td>
<td>-0.005578515</td>
</tr>
<tr>
<td>0.034404779</td>
<td>-0.019547031</td>
<td>-0.000233939</td>
<td>0.000838386</td>
</tr>
</tbody>
</table>
The maximum error is $1.6 \times 10^{-9}$ for the first approximation and $0.5 \times 10^{-9}$ for the second approximation.
APPENDIX III

FRESNEL INTEGRAL WITH THE ANISOTROPIC APPROXIMATION FOR YZ-LINBO₃

Consider the Fresnel Integral

\[ E(x_1, \rho_0) = \exp\left(-\frac{jk_e \rho_0}{\rho_0}\right) \int_{x_1-L/2}^{x_1+L/2} \exp\left(-\frac{jk_e x^2}{2\rho_0}\right) \, dx \tag{1} \]

where \( k_e \) is the energy propagation vector equal to \([3]::\\)

\[ k_e = k_0 \cos \phi \left(1 - a\theta^2 - b\theta^4 - c\theta^6 - d\theta^8\right) \tag{2} \]

Let \( \gamma = -(a\theta^2 + b\theta^4 + c\theta^6 + d\theta^8) \)

hence \( k_e = k_0 \cos \phi (1 + \gamma) \tag{3} \)

where \( \phi \) is the power flow angle and \( k_0 \) is the wave number along the pure mode axis. Next, the Fresnel integral (1) is modified to include the energy propagation parameters:

\[ E(x_1, \rho_0) = \frac{\exp\left(-j k_0 \cos \phi (1+\gamma) \rho_0\right)}{\rho_0} \int_{x_1-L/2}^{x_1+L/2} \exp\left(-j k_0 \cos \phi (1+\gamma)\right) \frac{x^2}{2\rho_0} \, dx \tag{4} \]

where \( k_0 = \frac{2\pi}{\lambda} \).

Now, to transform (4) into Fresnel integral classic form:

let \( \frac{2\pi}{\lambda} = \frac{x^2}{\lambda \rho_0} \cos \phi (1+\gamma) \) \text{ note: } \rho_0 = z
or \( x = \sqrt{\frac{\lambda \rho_0}{2}} \frac{v}{\sqrt{\cos^\gamma (1+\gamma)}} \)  \( (5) \)

Then, \( dx = \sqrt{\frac{\lambda \rho_0}{2}} \frac{dv}{\sqrt{\cos^\gamma (1+\gamma)}} \) \( (6) \)

Substituting (5) into (1) yields:

\[
E(x_1, \rho_0) = \frac{\exp \left(-\frac{jx}{\lambda \rho_0} \cos^\gamma (1+\gamma)\right)}{\rho_0 \sqrt{\cos^\gamma (1+\gamma)}} \sqrt{\frac{\lambda \rho_0}{2}} \int_{v_-}^{v_+} \exp \left(-\frac{jv^2}{2}\right) dv \]  \( (7) \)

Now, transforming the limits using (5):

When \( x = x_1 + \frac{L}{2} \)

\[
\sqrt{\frac{\lambda \rho_0}{2}} \frac{v}{\sqrt{\cos^\gamma (1+\gamma)}} = x_1 + \frac{L}{2}
\]

or \( v_+ = \frac{x_1 + \frac{L}{2}}{\sqrt{\frac{\lambda \rho_0}{2}}} \sqrt{\cos^\gamma (1+\gamma)} \)

Similarly, the lower limit is derived as

or \( v_- = \frac{x_1 - \frac{L}{2}}{\sqrt{\frac{\lambda \rho_0}{2}}} \sqrt{\cos^\gamma (1+\gamma)} \)

where \( \rho_0 \equiv z \)
For convenience acoustic wavelength scaled parameters are adopted:

\[ \hat{L} = L/\lambda_a \quad \hat{Z} = z/\lambda_a \quad \hat{X}_1 = X_1/\lambda_a \]

The integration limits become:

\[ v_+ = \frac{\hat{X}_1 - \hat{L}/2}{\sqrt{2/2}} [\cos \theta (1+\gamma)]^\frac{1}{2} \]  

\[ v_- = \frac{\hat{X}_1 - \hat{L}/2}{\sqrt{2/2}} [\cos \theta (1-\gamma)]^\frac{1}{2} \]  

Using (8) and (9), integral (7) becomes:

\[ E(x_1, z) = \frac{\exp \left(-j2\hat{Z}\cos \theta (1+\gamma)\right)}{\sqrt{2Z\cos \theta (1+\gamma)}} \int_{v_-}^{v_+} \exp \left(-j\frac{v^2}{2}\right) dv \]  

Integral (10) has to be put into the form required for the numerical evaluation, as outlined in Appendix II.

From (10), let

\[ C = \frac{\exp \left(-j2\hat{Z}\cos \theta (1+\gamma)\right)}{\sqrt{2Z\cos \theta (1+\gamma)}} \]

and (10) is rewritten in the following form:

\[ E(x_1, z) = c \int_{v_-}^{v_+} \exp \left(-j\frac{v^2}{2}\right) dv - c \int_{0}^{v_-} \exp \left(-j\frac{v^2}{2}\right) dv \]

Now apply the transformation

\[ t = \frac{v^2}{2} \]
or \( v = \sqrt{\frac{2t}{x}} \)

for which \( dv = \frac{dt}{\sqrt{2xt}} \).

The limits are transformed according to:

when \( v = v_+ + t_+ = \frac{x}{2} v_+^2 \)

\[ v = v_- + t_- = \frac{x}{2} v_-^2 \]

hence (11) takes on the following form:

\[
E(x_1, z) = c \int_0^{t_+} \exp\left(-\frac{zt}{2x}ight) dt - c \int_0^{t_-} \frac{\exp\left(-\frac{zt}{2x}\right)}{\sqrt{2}} dt
\]  \hspace{1cm} (12)

Integral (12) is evaluated by the series expansion of Appendix II.
APPENDIX IV

FREQUENCY DIVIDER CIRCUIT

The 70 MHz input signal is first amplified and then de-coupled to the pulse shaping network consisting of two Schottky diodes. The ECL switching threshold \( V_{BB} \) is provided by the line driver. The synchronous counter divides the 70 MHz input into a 70 kHz output pulse which is low pass filtered and triggers pulse generator 1 of Figure 6.3.
APPENDIX V

FABRICATION SUMMARY

1) Cleaning of Wafers

Wafers are placed in a plastic holder and lowered into the ultrasonic bath for about 5 minutes. The ultrasonic movement will shake off any dust particles attached to the surface of the wafers. Next, the wafers are lowered into a beaker with propanol to rinse off the ultrasonic bath residue. Then the wafers are microcleaned in a liquid soap and de-ionized water bath for about 10 minutes, followed by rinsing in a cascaded tank with de-ionized water only, for 5 minutes. To clean away any residual soap, the wafers are then rinsed in a beaker with chromic acid, followed by a thorough rinse in de-ionized water in the cascaded tank for 15 minutes.

2) Drying Process

Wafers are dried in the reflux tank for 3 to 4 minutes.

3) Metal Deposition

Metal may be deposited on the wafer surface either by evaporating or by sputtering. Sputtering is used for depositions thicker than 10 microns and the metallized surface is not as smooth as that achieved by using an evaporator.

For the fabrication of 70FIL devices the electron beam evaporator was used and a very smooth and even layer of aluminum was obtained. To measure the thickness of deposition, a quartz crystal, which changes its resonant frequency as the deposition layer increases, is used. The deposition rate was approximately 9A/sec for aluminum and the total layer deposited was 2000A.
4) Coating with photoresist and developing

The wafer is placed on a spinning table where it is held down by a vacuum. A layer of adhesive material is applied by spinning the table. This first layer ensures good adhesion of the photoresist itself since the aluminum surface is very smooth and shiny and the photoresist does not adhere uniformly otherwise. Next, photoresist is applied and spun to yield about a 1 micron layer. The thinner the layer, the better for the developing stage.

Positive photoresist is used to coat 7OFIL devices. The positive photoresist defines lines as narrow as 1 micron while the negative photoresist defines line width starting at 3 microns.

After the application of the photoresist, the wafer is baked in the oven at 85 - 90°C for approximately 25 minutes. Since LiNbO₃ is very fragile with sudden temperature changes, there must be a gradual increase and decrease in the oven temperature. Baking of the photoresist improves the success of the developing process.

The mask is then mounted on to the mask aligner and the flat edge of the wafer is lined up with the mask boundary. When the mask touches the substrate it is exposed to light. The exposure time is about 3 seconds and the light intensity approximately 7 mW/cm². Commercial developers are used, the developing time is about 45 to 60 seconds.

After developing, the wafers are baked again for approximately 20 minutes, this ensures good adhesion of the photoresist which will then be easier to etch.
5) **Etching**

The photoresist is etched away in an etching bath which is at a temperature of about 40°C. The etching should be uniform so that some areas do not overetch while others may be underetched.

The etched wafers are rinsed in de-ionized water and then in propanol to remove any remaining photoresist.

Lastly the wafers are rinsed in ultrasonic bath and then inspected under a microscope. There are approximately 12 dice per wafer of 70 MHz filter "70FIL102".

The wafers are diced with a standard IC saw with a special diamond tip for LiNbO$_3$. The individual dice are mounted in TO-8 packages where the wire bonding is done with thermal compression.
APPENDIX VI

COMPUTER PROGRAM LISTINGS

FILE FILTR FORTRAN A BELL-NORTHERN RESEARCH

C THIS PROGRAM PERFORMS NUMERICAL CONVOLUTION BETWEEN THE INPUT
C AND OUTPUT IDT'S TO GENERATE THE FILTER'S IMPULSE RESPONSE.
C THE FAST FOURIER TRANSFORM OF THE IMPULSE RESPONSE IS COMPUTED
C TO YIELD THE FILTER'S FREQUENCY RESPONSE.
C
C IMPLICIT REAL*(A-H,O-Z)
DIMENSION FR(512),X(9),TD(512),WK(1),Z(512),WK(512)
DIMENSION X(512),T(512),YW(512)
COMPLEX*16 FFT(257),FR(512),FT2(512),Y(512)
COMMON FT2,Y,X,T,YW
COMMON /DATA/ FO,F1,FSL,FSU,FP,FL,PI,N,NS,NFI,NFO,NDATA
C
C DEFINE CONSTANTS /DATA/ OF COMMON
C
FO = 700.000
F1 = 3.50000
FSL = 630.000
FSU = 770.000
FU = 73.50000
FL = 66.30000
PI = 3.141592653589793
N = 1000
NS = 128
NFI = 19
NFO = 125
NDATAS = 512
C
C DEFINE PARAMETERS
C
ND = NDATAS/2+1
ND2 = NDATAS/2
NFI = (NFI+1)/2
NF = NFI+1
N = NFO+1
C
OMEGAS = 2.00000*FO
DOMEGA = OMEGAS/NDATA
DTIME = 1/OMEGAS
ETIME = DTIME/2.00000
TLENG = DFLAT(NFO)+DTIME
C
C SUBROUTINE 'INPUT' GENERATES THE INPUT NON-APODIZED IDT
C CALL INPUT
C
C SUBROUTINE 'TRUNC' GENERATES THE OUTPUT IDT'S IMPULSE RESPONSE
C CALL TRUNC
C
DO 5 I=1,NDATAS
C WRITE(9,907) FT2(I),Y(I),X(I),T(I),YW(I)
C 5 CONTINUE
SUBROUTINE TRUNC

THIS PROGRAM USES THE WINDOW FUNCTION TO TRUNCATE THE IMPULSE
RESPONSE OF THE OUTPUT IDT. THE TRUNCATED RESPONSE IS THEN
POSITIONED ACCORDING TO DELAY.

IMPLICIT REAL*(A-H,O-Z)
DIMENSION TW(512),W(512),X(512),Y(512),YW(512)
COMPLEX*16 FT(512),Y(512)
COMMON FT2,Y,X,T,YW
COMMON /DATA/ F0,F1,FSL,FSU,FU,FI,N,F1,NF1,NFO,NDATA

DEFINE PARAMETERS

ND2 = NDATA/2
NF1 = (NFI+1)/2
NF = NFI+1

OMEGAS = 2.000*FO
DOMEGA = OMEGAS/NDATA
DTIME = 1/OMEGAS
ETIME = DTIME/2.000
TLONG = DFLOAT(NFO)+DTIME

KDATA = NDATA/2+1
LDATA = NDATA/2-1
MDATA = NDATA+1

KFO = (NFO-1)/2
MFO = (NFO+1)/2

NFO1 = KDATA-MFO
NFO2 = NFO+1
NFO3 = KDATA+KFO
NFO4 = NFO+1

TDD = DFLOAT(N)+DTIME
TDE = DFLOAT(N)+DTIME+ETIME

NFDD = (NFI+1)/2+N
NFDE = NFI/2+1+N

NFOE1 = KDATA-NFO/2-1
NNE1 = NFDE-NFO/2-1
NNE2 = NFDE-NFO/2

NN1 = NFDE-MFO
NN2 = NN1+NFO+1

NDF = NFDE-MFO
NDF2 = KDATA-MFO
NDF3 = KDATA+MFO
FILE TRUNC FORTRAN A BELL-NORTHERN RESEARCH

NE1 = NFDO-NFD/2
NE2 = ND2-NFD/2
NE3 = ND2-NFD/2+1

C SUBROUTINE INPUT GENERATES THE INPUT AND THE TIME VECTORS
CALL INPUT

C SUBROUTINE 'APOD' GENERATES THE OUTPUT IDT IMPULSE RESPONSE
CALL APOD

C PRINT HEADINGS
WRITE(9,906)

C CALCULATE TIME OF WINDOW
DO 100 I=1,NDATA
C CHECK FOR EVEN NFDO
I=1,NFDO
CMULT=DFLOAT(J)
TIME=CMULT+DTIME
TW(I)=TIME

C GO TO 100
C CALCULATION FOR EVEN NFDO

110 J=1,ND2
CMULT=DFLOAT(J)
TIME=CMULT+DTIME-ETIME
TW(I)=TIME

C GO TO 100
C COMPUTE THE AMPLITUDE OF THE HAMMING WINDOW FUNCTION

DO 200 I=1,NF01
W(I)=0.000
200 CONTINUE

DO 210 I=NF02,NF03
PAR=TW(I)
W(I)=5.40-0.51*4.6D-01*DCOS(2.0D0*PI*(PAR/TLENG))
210 CONTINUE

DO 220 I=1,NDATA
W(I)=0.000
220 CONTINUE
FILE TRUNC FORTRAN A BELL-NORTHERN RESEARCH

C
C TRUNCATE IMPULSE RESPONSE WITH HAMMING WINDOW
C DO 400 I=1,NDATA
C CHECK FOR EVEN NFO
C IF((NFO/2)*2.EQ.NFO) GO TO 410
C ODD NFO
C YW(I)=DREAL(Y(I))*W(I)
C GO TO 400
C EVEN NFO
C 410 IF(I.EQ.NDATA) GO TO 420
C YW(I)*((DREAL(Y(I)))*DREAL(Y(I+1)))/2)*W(I)
C GO TO 400
C 420 YW(I)*((DREAL(Y(I)))*DREAL(Y(I+1)))/2)*W(I)
C 400 CONTINUE
C CHECK FOR ODD OR EVEN INPUT OR OUTPUT IDT
C IF((NFI/2)*2.EQ.NFI AND (NFO/2)*2.EQ.NFO) GO TO 590
C IF((NFI/2)*2.EQ.NFI) GO TO 690
C IF((NFO/2)*2.EQ.NFO) GO TO 790
C POSITION ODD - ODD IDT'S
C DO 500 I=1,NFO
J=1,NFI
K=1,NFO
YW(J)+YW(K)
500 CONTINUE
C DO 510 I=1,NFI
YW(I)=0.000
510 CONTINUE
C DO 520 I=1,NDATA
YW(I)=0.000
520 CONTINUE
C GO TO 900
C POSITION EVEN - EVEN IDT'S

9
FILE TRUNC FORTRAN A BELL-NORTHERN RESEARCH

C 590 DO 600 I=1,NFO
     J=I+NNE1
     K=I+NFO1
     YW(J)=YW(K)
600 CONTINUE

C 610 DO 610 I=1,NNE1
     YW(I)=0.0DD
610 CONTINUE

C 620 DO 620 I=NNE2,NDATA
     YW(I)=0.0DD
620 CONTINUE

C 690 DO 700 I=1,NFO
     J=I+MD1
     K=I+MD2
     YW(J)=YW(K)
700 CONTINUE

C 710 DO 710 I=1,MD1
     YW(I)=0.0DD
710 CONTINUE

C 720 DO 720 I=1,MD3
     YW(I)=0.0DD
720 CONTINUE

C 900 GO TO 900

C 790 DO 800 I=1,NFO
     J=I+NNE1
     K=I+NME2
     YW(J)=YW(K)
800 CONTINUE

C 810 DO 810 I=1,NME2
     YW(I)=0.0DD
810 CONTINUE

C 820 DO 820 I=NME3,NDATA
     YW(I)=0.0DD
820 CONTINUE

C 900 CHECK FOR OVERLAP OF INPUT AND OUTPUT IDT'S

C 910 IF (PAR.LT.-1.0D-12.AND.PAR.GT.1.0D-12) GO TO 920
     GO TO 910
C 920 WRITE(6,907)  
STOP  
C 910 CONTINUE  
C AMAX = DABS(YW(I))  
DO 950 I = 1, NDATA  
IF(DABS(YW(I)).GT. AMAX) AMAX = DABS(YW(I))  
950 CONTINUE  
C PRINT RESULTS  
C DO 1000 I = 1, NDATA  
YW(I) = YW(I)/AMAX  
C WRITE(9,905) I, T(I), X(I), YW(I)  
WRITE(11,1009) YW(I)  
1000 CONTINUE  
C  
C 905 FORMAT(110, 1P020.8)  
906 FORMAT(9X,'I', 8X, 'TIME', 16X, 'INPUT IDT', 11X, 'OUTPUT IDT')  
907 FORMAT(10X, 'DELAY ERROR: INCREASE THE DELAY')  
1009 FORMAT(1P020.10)  
C  
RETURN  
C STOP  
END
FILE, APOD FORTRAN A BELL-NORTHERN RESEARCH

SUBROUTINE APOD

THIS PROGRAM EVALUATES THE OUTPUT IDT APODIZATION
BY PERFORMING THE INVERSE FOURIER TRANSFORM OF THE
FREQUENCY RESPONSE.

IMPLICIT REAL*8(A-H.O-Z)
DIMENSION IWK(10),X(512),T(512),YW(512)
COMPLEX*16 FT2(512),Y(512)
COMMON FT2,Y,X,T,YW
COMMON /DATA/ FO,F1,FSL,FSU,FL,PI,N,NS,NF1,NFO,NDATA

DEFINE PARAMETERS

M=9
KDATA+DATA/2+1
LDATA+DATA/2-1
NDATA+DATA+1

SUBROUTINE 'INPUT' GENERATES THE INPUT NON-APODIZED IDT
AND THE TIME VECTOR

CALL INPUT

SUBROUTINE 'FREQ' GENERATES THE FREQUENCY
RESPONSE OF THE OUTPUT IDT

CALL FREQ

PRINT HEADINGS I, TIME, Y
WRITE(*,9,908)

IMSL SUBROUTINE FFT2C COMPUTES
THE INVERSE FOURIER TRANSFORM

CALL FFT2C(FT2,M,IWK)

REARRANGE FFT2C OUTPUT VECTOR FOR SYMMETRY

DO 10 I=1,KDATA
J=I-1
K=KDATA-J
Y(K)=FT2(I)
10 CONTINUE

DO 20 I=1,LDATA
J=DATA-I
K=DATA+I
Y(K)=FT2(J)
20 CONTINUE

PRINT RESULTS

DO 50 I=1,NDATA
FILE APOD FORTRAN A BELL-NORTHERN RESEARCH

C WRITE(*,905) I,T(I),X(I),Y(I)
C50 CONTINUE
C
905 FORMAT(13,3X,1PD10.3,1P4D15.4)
906 FORMAT(2X,'1',4X,'TIME',11X,'X(I)',11X,'Y(I)-REAL',5X,'Y(I)-IMAG')
C L
C
RETURN
C STOP
END
SUBROUTINE FREO
C THIS PROGRAM COMPUTES THE FAST FOURIER TRANSFORM (FFT)
C OF THE INPUT UNIFORM TRANSDUCER.
C THE DESIRED FREQUENCY RESPONSE OF THE OUTPUT IDT IS
C GENERATED.
C IMPLICIT REAL*8(A-H,O-Z)
DIMENSION K(9),WK(1),FR(512),X(512),T(512),YW(512)
COMPLEX*K(16,FT(512),FFT(512),FT2(512),Y(512)
COMMON FT2,Y,X,T,YW
COMMON /DATA/ FO,F1,FSL,FSU,FL,PI,N,NS,NF1,NF0,NDATA
C DEFINE PARAMETERS
ND2=NDATA/2
OMEGAS=2.000*FO
DOMEGA=OMEGAS/NDATA
NS1 = ND2-NS
NS2 = NS1+1
NS3 = NS2+2*NS
NS4 = NS3+1
C SUBROUTINE 'INPUT' GENERATES THE INPUT NON-APODIZED IDT
CALL INPUT
C C IMGL SUBROUTINE FFTRC COMPUTES THE FAST FOURIER TRANSFORM
C OF THE INPUT NON-APODIZED TRANSDUCER.
CALL FFTRC(X,NDATA,FT1,K,WK)
C PRINT HEADING
WRITE(9,906)
C CALCULATE THE REMAINING VALUES OF FT1
DO 30 I=2,ND2
J=NDATA+2-I
FT1(J)=DCONJG(FT1(I))
30 CONTINUE
C CALCULATE THE MAX VALUE OF FT1 TO NORMALIZE FT1
AMAX=CDABS(FT1(1))
DO 40 I=1,NDATA
IF(CDABS(FT1(I)).GT.AMAX) AMAX=CDABS(FT1(I))
40 CONTINUE
C DO 60 I=1,NDATA
C NORMALIZE FT1
FT1(I)=FT1(I)/AMAX
FILE: FREQ  FORTRAN A  BELL-NORTHERN RESEARCH

60 CONTINUE
C  COMPUTE MAG OF FT1 AND SET PHASE = 0.0
C  DO 61 I=1,NDATA
     FMAG=CDABS(FT1(I))
     FT1(I)=DCMPLX(FMAG,0.000)
61 CONTINUE
C  CALCULATE THE DESIRED OUTPUT IDT RESPONSE
C  DO 65 I=1,NS1
     FT2(I)=0.000
65 CONTINUE
C  DO 66 I=NS2,NS3
     FT2(I)=1.000/FT1(I)
C  DO 66 I=1,NS4
     FT2(I)=1.000
66 CONTINUE
C  COMPUTE THE MAX VALUE OF FT2 TO NORMALIZE FT2
C     BMAX=CDABS(FT2(I))
C  DO 68 I=1,NDATA
     IF(CDABS(FT2(I)).GT.BMAX) BMAX=CDABS(FT2(I))
68 CONTINUE
C  NORMALIZE FT2
C  DO 69 I=1,NDATA
     FT2(I)=FT2(I)/BMAX
69 CONTINUE
C  DO 80 I=1,NDATA
C  COMPUTE THE MAGNITUDE OF FT1 IN DB'S
C     ABSVAL=CDABS(FT1(I))
C     IF(ABSVAL.LE.0.0) ABSVAL=10.0-12
C     AMAG=20.000+DLOG10(ABSVAL)
C  COMPUTE THE PHASE OF FT1 IN DEGREES
C     APHASE=0.000
C  COMPUTE THE MAGNITUDE OF FT2 IN DB'S
C     BBSVAL=CDABS(FT2(I))
C     IF(BBSVAL.LE.0.0) BBSVAL=10.0-12
C     BMAG=20.000+DLOG10(BBSVAL)
C  COMPUTE THE PHASE OF FT2 IN DEGREES
C     BPHASE=0.000
FILE: FREQ FORTRAN A BELL-NORTHERN RESEARCH

C COMPUTE FREQUENCIES CORRESPONDING TO SAMPLE NUMBERS
C
J=I-1
CMULT=DFLOAT(J)
FREQ=CMULT*DFMEGA
FR(I)=FREQ
C
WRITE(9,905) I,FR(I),AMAG,APHASE,BMAG,BPHASE
C
80 CONTINUE
C
905 FORMAT( I3,X,1PD9.3,X,1P4E11.3)
906 FORMAT(2X,'I',X,'FREQ(HZ)',4X,'AMAG(DB)',3X,'APHASE(DEG)' )
907 FORMAT( I3,X,1P4E11.3)
C
C STOP
RETURN
END
FILE: INPUT FORTRAN A BELL-NORTHERN RESEARCH

SUBROUTINE INPUT

C THIS PROGRAM GENERATES COMPLETE INPUT IDT DATA FILE
C AND THE CORRESPONDING TIME VECTOR.
C
C IMPLICIT REAL *(A-H,O-Z)
C DIMENSION X(512),T(512),YW(512)
C COMPLEX=F2(512),Y(512)
C COMMON FT2,Y,X,T,YW
C COMMON /DATA/ FO,F1,FSL,FSU,FL,PI,N,NS,NFI,NFD,NDATA
C
C ABBREVIATIONS
C
FO = CENTRE FREQUENCY
F1 = BANDWIDTH TO ROLLOFF (2+F1)
NFI = NUMBER OF FINGERS OF INPUT TRANSUDER
NFD = NUMBER OF FINGERS OF OUTPUT TRANSUSCER
CTIME= STEP IN TIME FROM FINGER TO FINGER
TLEN= TOTAL LENGTH OF TRANSUDER IN TIME
DELAY= SEPARATION OF INPUT AND OUTPUT IDT.CENTER TO CENTER

DEFINE PARAMETERS

KFI=NFI+1
NFI=(NFI+1)/2

CTIME=1/(2.0DO+F0)
CTIME=CTIME/2.0DO

PRINT HEADINGS I,TIME,XIN
WRITE(9,900)

GENERATE INPUT IDT DATA

DO 101 I=1,NFI
J=(-1)**(I+1)
CMULT=DFLOAT(J)
X(I)=CMULT
101 CONTINUE

DO 102 I=KFI,NDATA
X(I)=0.0DO
102 CONTINUE

GENERATE TIME FILE

DO 103 I=1,NDATA

TEST FOR EVEN NFI
IF((NFI/2)+2.EQ.NFI) GO TO 10

CALCULATION FOR ODD NFI
FILE INPUT FORTRAN A BELL-NORTHERN RESEARCH

J=I-MFI
CMULT=DFLOAT(J)
TIME=CMULT+DTIME
T(I)=TIME
GO TO 103

C CALCULATION FOR EVEN NFI

10 J=I-MFI/2
CMULT=DFLOAT(J)
TIME=CMULT+DTIME+ETIME
T(I)=TIME
GO TO 103

CONTINUE

C PRINT RESULTS

DO 104 I=1,NDATA
WRITE(9,905) I,T(I),X(I)
CONTINUE

905 FORMAT(I10,1P3D20.6)
906 FORMAT(9X,'I',10X,'TIME(I)',13X,'XIN(I)')
RETURN

C STOP END
THIS ROUTINE COMPUTES A NEW DIFFRACTION COMPENSATED DESIGN OF SURFACE ACOUSTIC WAVE FILTER

IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION YW(125),ACOUNT(125),SUBL(125),ALENG(125),XOLD(125,410)
DIMENSION XLIN(512),XPHS(512),TWLIN(512),TWPHS(512),XNEW(500)
DIMENSION XX(500),XNEW(500),XXMAG(512),XXPHS(512),YYMAG(500),YYPHS(500)
DIMENSION XLEN(512),XMAG(512),XMAG(512),YMAG(512),XNEW(500),XSUB(19)
DIMENSION FFTLIM(512),FFTTL(512),FFTPH(512),XLEN(512),XSUB(19)
DIMENSION YYNN(500),YYMAG(500),YYPH(500),YNEW(500),YNEW(500)
DIMENSION YNEW(500),YYNEW(500),YLEN(512),YNEW(500),YNEW(500)
COMPLEX*16 X(S12),XW(S12),SIN(125),YD(125),YOLD(125,410),TX(S12)
COMPLEX*16 YNEW(500),YYNEW(500),YNEW(500),YNEW(500)
COMPLEX*16 O(S12),FFT(S12),XSIM(19),XDIFF(S12)
COMMON /DATA/ PI,FO,FLAMBD,DELAY,AP1,AP2,NFO,NFI,NDATA

DEFINE CONSTANTS /DATA/ OF COMMON
PI = 3.14159265358979 / 50.00000000
FO = 70.00000000
FLAMBD = 49.20000000
DELAY = 5419.52000000
AP1 = 2481.48000000
AP2 = 45.00000000
NFO = 125
NFI = 19
NDATA = 512

DEFINE PARAMETERS
KPO = -(NFO-1)/2
KFI = -(NFI-1)/2
DTR = PI/180.00000000
RTD = 180.00000000/PI
THETA = 45.00000000*DTR
ITMAX = 8
ITMIN = 7
IT = 1
NN = 57
RLXM = 0.75000000
RLXP = 0.00000000

IS YOUR SUBSTRATE ISOTROPIC??? (1 = Y, 0 = N)
WRITE(8,1000)
READ(6,+)* ISO
IF(ISOT.NE.1.AND.ISOT.NE.0) GO TO 1
IF(ISOT.EQ.0) GO TO 2
WRITE(10,1000)
WRITE(9,1000)
WRITE(7,1000)
FILE ANISOT FORTRAN A BELL-NORTHERN RESEARCH

AR = 0.000
BR = 0.000
CR = 0.000
dR = 0.000
GO TO 3

C
DO POLYNOMIAL CURVEFIT OF VELOCITY VS. ANGLE OF PROPAGATION

ON LINGO3 SUBSTRATES (BASED ON SLOBODNIN'S DATA)
A, B, C, D ARE COEFFICIENTS OF 8TH ORDER POLYNOMIAL

2 WRITE(10,1006)
WRITE(9,1006)
WRITE(7,1006)
A = +1.28000D-04
B = 3.64920D-07
C = -5.08870D-10
D = 2.75510D-13

C
ANIO0560
ANIO0570
ANIO0580
ANIO0590
ANIO0800
ANIO0810
ANIO0820
ANIO0830
ANIO0840
ANIO0850
ANIO0860
ANIO0870
ANIO0880
ANIO0890
ANIO0900
ANIO0910
ANIO0920
ANIO0930
ANIO0940
ANIO0950
ANIO0960
ANIO0970
ANIO0980
ANIO0990
ANIO1000
ANIO1010
ANIO1020
ANIO1030
ANIO1040
ANIO1050
ANIO1060
ANIO1070
ANIO1080
ANIO1090
ANIO1100

C
WRITE(6,1001) A
WRITE(9,1001) A
WRITE(8,1002) B
WRITE(10,1002) B
WRITE(9,1002) B

C
WRITE(6,1003) C
WRITE(10,1003) C
WRITE(9,1003) C
WRITE(6,1004) D
WRITE(10,1004) D
WRITE(9,1004) D

C
DO YOU WANT TO HAVE AN INCLINE IN THE APODIZED IDT? (Y=1, 0=NO)

3 WRITE(6,1007)
READ(6,*) INCLN
IF(INCLN.NE.1.AND.INCLN.NE.0) GO TO 3
IF(INCLN.EQ.1) GO TO 4
IF(INCLN.EQ.0) BETA = 0.000
GO TO 5

C
HOW MANY DEGREES???

4 WRITE(8,1006)
READ(8,1006) BETA
WRITE(8,1009) BETA
WRITE(10,1009) BETA
WRITE(9,1009) BETA
BETA = DTR* BETA
FILE: ANISOT FORTRAN A BELL-NORTHERN RESEARCH

C GENERATE THE INPUT UNAPODIZED VECTOR 'X'
5 CALL INPIDT(X, XLIN, XPAR)
C APODIZATION SIGN VECTOR OF THE INPUT UNAPODIZED IDT
C
DO 30 I = 1, NFI
XLENG(I) = AP2*XLIN(I)
CMULT = DSIGN(1.0, DREAL(X(I)))
XSINE(I) = DCMPLX(CMULT, 0.0, 0.0)
30 CONTINUE

C SUBLGTH OF THE TRANSMITTING FINGERS OF THE INPUT UNAPODIZED IDT
C
N = 25
DO 31 I = 1, NFI
C
DO 32 J = 1, N
XSUB(I) = DABS(XLENG(I))/DFLOAT(J)
IF(XSUB(I).LE.2.0) GO TO 33
32 CONTINUE
31 CONTINUE
C
33 NXCNT(I) = J
C WRITE(10, 2004) I, NXCNT(I), XLENG(I), XSUB(I)
C
C 31 CONTINUE
C
C COMPUTE DIFFRACTED WEIGHTS OF THE INPUT IDT
C GAMA = 0.0
C NDAT = NFI
C CALL DIF[NDAT, KFI, AR, BR, CR, DR, GAMA, XSUB, XOLD, YOLD, NBR]
C
C DO 900 I = 1, NFI
C DO 901 J = 1, NBR
C
WRITE(10, 2004) I, J, XOLD(I, J), YOLD(I, J)
C
C 901 CONTINUE
C 900 CONTINUE
C
C GO THRU SORTING, SLIDING AND INTEGRATION
C
DO 51 I = 1, NFI
SUB = XSUB(I)
NVAL = NXCNT(I)
C
DO 52 J = 1, NBR
XX(J) = XOLD(I, J)
YY(J) = YOLD(I, J)
52 CONTINUE
C CALL SORT(NBR, SUB, XX, YY, XNEW, YNEW, NINT)
C CALL SLIDE(NVAL, INT, XNEW, YNEW, XPROF, YPROF, NTOT)
C CALL INTEGR(NTOT, XPROF, YPROF, SUM)
C
XDIFF(I) = SUM+XSINE(I)
FILE ANISOT FORTRAN A BELL-NORTHERN RESEARCH

51 CONTINUE
C REMAINING VALUES OF X-VECTOR(512) ARE ZERO
C NFF = NFI+1
DO 53 I = NFF,NDATA
XDIF(I) = DCMPLX(T0,O.DO,0.DO)
53 CONTINUE
C DO 54 I = 1,NFI
XDIF(I) = XDIF(I)/XDIF(NFI)
54 CONTINUE
C CALL MAGPHS(XDIF,XDOLN,XMDAG,XDPHNS)
C DO 55 I = 1,NDATA
C WRITE(10,2002) I,XDOLN(I),XDPHNS(I)
C CONTINUE
C READ ORIGINAL APODIZATION WEIGHTING OF OUTPUT IDT
DO 56 I = 1,NFI
READ(I,1)YW(I)
ALENG(I) = AP2+YW(I)
56 CONTINUE
C OUTPUT IDT 125 SAMPLE COMPLEX VECTOR YO(125)
CALL OUTIDT(YV,YQ)
C CREATE APODIZATION SIGN VECTOR OF THE OUTPUT IDT
DO 57 I = 1,NFI
CMULT = DSIGN(1.DO,DREAL(YQ(I)))
SINE(I) = DCMPLX(CMULT,0.DO)
57 CONTINUE
C POSITION THE OUTPUT IDT AT THE CORRECT DELAY AND GET TW(512)
CALL POSITN(YO,TW,TWLN,TWPHS)
C N = 25
C COMPUTE SUBLNGTH OF THE TRANSMITTING FINGER
DO 60 I = 1,NFO
C DO 65 J = 1,N
C SUBL(J) = DABS(ALENG(I))/DFLOAT(J)
C IF(SUBL(J),LE,2.000) GO TO 6
C CONTINUE
C 6 NCOUNT(N) = J
C WRITE(10,2004) I,NCOUNT(I),ALENG(I),SUBL(I)
C CONTINUE
FILE ANISOT FORTRAN A BELL-NORTHERN RESEARCH

NDAT = NFD
CALL DIF(NDAT, KFO, AR, BR, CR, DR, BETA, SUBL, XOLD, YOLD, NUMBER)
C
C GO THRU SORTING, SLIDING & INTEGRATION
C
DO 100 I = 1, NFD
SUB = SUBL(I)
NVAL = NCOUNT(I)
C
DO 200 J = 1, NUMBER
XX(J) = XOLD(I, J)
C
IF(NVAL.EQ.1) GO TO 11
GO TO 13
C
11 DUMMY = DABS(XX(J))
IF(DUMMY.GT.22.5D00) GO TO 12
GO TO 13
C
12 YY(J) = DCMPLX(0.0D0, 0.0D0)
GO TO 200
C
13 YY(J) = YOLD(I, J)
C
200 CONTINUE
C
IF(NVAL.EQ.1) GO TO 10
GO TO 15
C
10 CALL INTEGR(NUMBER, XX, YY, SUM)
GO TO 20
C
15 CALL MPGHS(NUMBER, YY, YYLIN, YYMAG, YPHS)
C
CALL SORT(NUMBER, SUB, XX, YY, XNEW, YNEW, INT)
C
CALL MPGHS(INT, YNEW, XNEW, YNEW, YNEWP)
C
CALL COMPLX(INT, YNEWL, YNEWP, YNEW)
C
CALL SLIDE(NVAL, INT, XNEW, YNEW, XPROF, YPROF, NTOT)
C
CALL MPGHS(NTOT, YPROF, YLIN, YMAG, YPHS)
C
DO 240 K = 1, INT
WRITE(10, 2002) K, XX(K), YYLIN(K), YPHS(K), XNEW(K), YNEWL(K), YNEWP(K)
C
& XPROF(K), YLIN(K), YPHS(K)
C240 CONTINUE
C
DO 241 L = 1, NTOT
WRITE(10, 2002) L, XPROF(L), YLIN(L), YPHS(L)
C241 CONTINUE
C
WRITE(*, *) NTOT


FILE ANISOT FORTRAN A BELL-NORTHERN RESEARCH

C CALL INTEG(INTOT,XPROF,YPROF,SUM)
C
20 TOTL(I) = SUM*SINE(I)
C
C WRITE(10,2002) I,TOTL(I)
C
100 CONTINUE
C
C CALL POSITN(TOTL,TX,TXMAG,TXPHS)
C
C CALL NORM(TX,TXM)
C
C CALL MAGPHS(TXN,TXNLIN,TXMAG,TXMPHS)
C
C CALL CONVOL(TXM,XDIF,Q,OLIN,QMAG,QPHS)
C
DO 242 I = 1,NDATA
WRITE(10,2002) I,XDLIN(I),XDPHS(I),TXNLIN(I),TXMPHS(I),OLIN(I)
& QPHS(I)
242 CONTINUE
C
C FFT2C FOR EXP(-JBD)
DO 243 I = 1,NDATA
Q(I) = DCOMAG(Q(I))
243 CONTINUE
C
C CALL FFT2C(Q,MM,IN)
C
C DO 244 I = 1,NDATA
Q(I) = DCOMAG(Q(I))/NDATA
244 CONTINUE
C
C CALL FNORM(Q,FFT)
C
C CALL MAGPHS(FFT,FXTLIN,FXTMAG,FFTPHS)
C
C CALL FRVECT(FR)
C
'DO 250 I = 1,NDATA
WRITE(9,2002) I,FR(I),FXTMAG(I),FFTPHS(I)
250 CONTINUE
C
C DO 500 I = 184,330
FR(I) = FR(I)*1.0D+08
WRITE(3,2001) FR(I),FXTMAG(I)
500 CONTINUE
C
1000 FORMAT(/,5X,'IS YOUR SUBSTRATE ISOTROPIC??? (1 = Y, O = N)')
1001 FORMAT(5X,'A = ',1PD16.7)
1002 FORMAT(5X,'B = ',1PD16.7)
1003 FORMAT(5X,'C = ',1PD16.7)
1004 FORMAT(5X,'D = ',1PD16.7)
1005 FORMAT(5X,'THIS SUBSTRATE IS ISOTROPIC!!'),/
1006 FORMAT(5X,'THIS SUBSTRATE IS AN-ISOTROPIC!!'),/
1007 FORMAT(5X,'THIS SUBSTRATE IS AN-ISOTROPIC!!*/,/)
1008
FILE ANISOT FORTRAN A BELL-NORTHERN RESEARCH

1007 FORMAT(/,5X,'DO YOU WISH TO HAVE AN INCLINE IN THE APODIZED IDT???ANIO3310
& (1=YES.0=NO)') ANIO3320
1008 FORMAT(5X,'HOW MANY DEGREES??') ANIO3330
1009 FORMAT(5X,'THE APODIZED IDT HAS AN INCLINE OF BETA(DEG) = ', ANIO3340
& 1PD10.4.///) ANIO3350

C ANIO3360
2001 FORMAT(1P3D20.10) ANIO3370
2002 FORMAT(1B,1P10D12.4) ANIO3380
2003 FORMAT(1H1) ANIO3390
2004 FORMAT(21S,1P10D12.4) ANIO3400

C ANIO3410
C ANIO3420
STOP ANIO3430
END ANIO3440
FILE COMPEN FORTRAN A BELL-NORTHERN RESEARCH

RLXM = 0.0000
RLXP = 0.100
IT = 1
N1 = 57
KFO = -(NF0-1)/2
KFI = -(NF1-1)/2

C IS YOUR SUBSTRATE ISOTROPIC??? (1 = Y, 0 = N)
1 WRITE(6,1000)
READ(6,*) ISOT
IF(ISOT NE 1 AND ISOT NE 0) GO TO 1
IF(ISOT EQ 0) GO TO 2

C WRITE(10,1021)
WRITE(9,1021)
WRITE(7,1021)
AR = 0.000
BR = 0.000
CR = 0.000
DR = 0.000
GO TO 3

C DO POLYNOMIAL CURVEFIT OF VELOCITY VS ANGLE OF PROPAGATION
C ON LINE03 SUBSTRATES (BASED ON SLOBODNIK'S DATA)
C A, B, C, D ARE COEFFICIENTS OF 8TH ORDER POLYNOMIAL

2 WRITE(10,1022)
WRITE(9,1022)
WRITE(7,1022)
A = -126000-0.04
B = 3.68920-0.07
C = -5.08870-0.10
D = 2.75510-0.13

C AR + A*(RTD+2)
BR + B*(RTD+4)
CR + C*(RTD+6)
DR + D*(RTD+8)

C WRITE(6,1029) A
WRITE(7,1029) A
WRITE(9,1029) A

C WRITE(6,1010) B
WRITE(7,1010) B
WRITE(9,1010) B

C WRITE(6,1027) C
WRITE(7,1027) C
WRITE(9,1027) C

C WRITE(6,1028) D
WRITE(7,1028) D
WRITE(9,1028) D
FILE COMPEN FORTRAN A BELL-NORTHERN RESEARCH

C DO YOU WISH TO HAVE AN INCLINE IN THE APODIZED IDT? (1= y, 0= n)  
3 WRITE(8,1023)
   1023   I=1,NDATA
   1030   IF(INCLN NE.1.AND.INCLN NE.0) GO TO 3
   1040   IF(INCLN.EQ.1) GO TO 4
   1050   IF(INCLN.EQ.0) BETA = 0.OOO
   1060   GO TO 5
   1070 C HOW MANY DEGREES??
   1080 WRITE(6,1024)
   1090 READ(6,*) BETA
   1100 WRITE(6,1025) BETA
   1110 WRITE(7,1025) BETA
   1120 WRITE(9,1025) BETA
   1130 BETA = DTR*BETA
   1140 C GENERATE THE INPUT UNAPODIZED VECTOR 'X'
   1150 CALL INPIDT(X,XLIN,XPHS)
   1160 C APODIZATION SIGN VECTOR OF THE INPUT UNAPODIZED IDT
   1170 APODIZATION SIGN VECTOR OF THE INPUT UNAPODIZED IDT
   1180 DO 30 I = 1,NDATA
   1190 XLENG(I) = APX*XLIN(I)
   1200 CMULT = DSIGN(1.ODD,DREAL(X(I)))
   1210 XSINE(I) = DCMPLX(CMULT,0.ODO)
   1220 CONTINUE
   1230 C COMPUTE DIFFRACTED WEIGHTS OF THE INPUT IDT
   1240 GAMMA = 0.ODO
   1250 NDAT = NFI
   1260 CALL DIFCOR(NDAT,KFI,AR,CR,CR,GAMA,X,XDIF)
   1270 C NORMALIZE X-VECTOR WRt. LARGEST WEIGHT
   1280 DO 54 I = 1,NFI
   1290 XDIF(I) = (XDIF(I)*XDIF(NFI))/XSINE(I)
   1300 CONTINUE
   1310 C CALL MAGPHS(XDIF,XDLIN,XDMAG,XDPHS)
   1320 C READ ORIGINAL APODIZATION WEIGHTING OF OUTPUT IDT
   1330 DD 50 I=1,NFO
   1340 READ(1,*) YW(I)
   1350 ALENG(I) = DABS(YW(I))*AP2
   1360 CONTINUE
   1370 C OUTPUT IDT 125 SAMPLE COMPLEX VECTOR YO(125)
   1380 CALL OUTIDT(YW,YO)
   1390 C POSITION THE OUTPUT IDT AT THE CORRECT DELAY AND GET TW(512)
   1400 CALL POSITN(YO,TWX,TWLLN,TWPHSS)
   1410 C CREATE APODIZATION SIGN VECTOR
   1420 DO 55 I=1,NDATA
   1430 TW(I) = TWX(I)
   1440 CMULT = DSIGN(1.ODO,DREAL(TW(I)))
   1450 SINE(I) = DCMPLX(CMULT,0.ODO)
   1460 CONTINUE
FILE COMPEN FORTRAN A BELL-NORTHERN RESEARCH

55 CONTINUE

C

C COMPUTE TRANSMITTING SPLIT FINGERS

C DO 60 I = 1,NDATA

C TWL(I) = (TW(I)/DSORT(2.000))*DCMPLX(DCOS(THTA),DSIN(THTA))

C TWR(I) = (TW(I)/DSORT(2.000))*DCMPLX(DCOS(THTA),DSIN(-THTA))

C CONTINUE

60 CONTINUE

C

C READ IN LAST SET OF RELAXED VALUES

C DO 200 I = 1,NFD

C READ(11,2001) TWLL(I),TWR(I)

C TXL(I) = DABS(TWLL(I))*SINE(I)*DCMPLX(DCOS(THTA),DSIN(THTA))

C TXR(I) = DABS(TWR(I))*SINE(I)*DCMPLX(DCOS(THTA),DSIN(-THTA))

C CONTINUE

200 CONTINUE

C

C CALL POSITN(TXL,TWL,TWLL,TWL)

C CALL POSITN(TXR,TWR,TWRL,TWR)

C

C CALL NORML(TWL,TWLN)

C CALL NORML(TWR,TWRN)

C

C DIFFRACTED WEIGHTS OF VECTOR TV(512)-DW(512)

C NDAT = NFD

C DO 10 CALL DIFCOR(NDAT, KFD, AR, BR, CR, DR, BETA, TWLN, DWL)

C CALL DIFCOR(NDAT, KFD, AR, BR, CR, DR, BETA, TWRN, DWR)

C

C DO 66 I = 1,NDATA

C DWL(I) = DWL(I)*SINE(I)

C DWR(I) = DWR(I)*SINE(I)

C TWW(I,1) = TW(I)

C TWW2(I,1) = TWLN(I)

C TWW2(I,1) = TWRN(I)

C CONTINUE

C

C CALL NORML(DWL,DWL)

C CALL NORML(DWR,DWR)

C

C CALL MAGPHS(DWL,DWL,TDLM,TDLP)

C CALL MAGPHS(DWR,DWR,TDLM,TDLP)

C

C CALL MAGPHS(TWL,TWL,TDLM,TDLP)

C CALL MAGPHS(TWR,TWR,TDLM,TDLP)

C

C DO 65 I = 1,NDATA

C DUMLL(I) = TWW(I)

C DUMLP(I) = TWW(I)

C DUMRL(I) = TWW(I)

C DUMRP(I) = TWW(I)

C CONTINUE

C

C CALL MAGPHS(TWL,TWLM,TWMAG,TPHWS)

C

C PUT DWL IN THE CORRECT QUADRANT

C DO 68 J = 1,NFD

C I = J+NN1

C CONTINUE

C

C
FILE COMPEN FORTRAN A BELL-NORTHERN RESEARCH

DWLP(I) = DWLP(I)+90.000
IF(DWLP(I) GT 180.000) DWLP(I) = DWLP(I)-360.000
DW(I) = DWLP(I)+DCMPLX(DCOS(DWLP(I)*DTR),DSIN(DWLP(I)*DTR))
98 CONTINUE
C
C NORMALIZE DWL WRT. ABSOLUTE VALUE OF CENTER WEIGHT
CALL NORM(DWL,DWLN)
C
C WRITE(10,2003)
C
C ADD DWL & DWR TO GET A SINGLE COMBINED WEIGHT
DO 70 I = 1,NDATA
DWS(I) = DWL(I)+DWR(I)
70 CONTINUE
C
C NORMALIZE DWS(512) WRT CENTER WEIGHT = 1
CALL NORM(DWS,DWSN)
C
C COMPUTE MAGNITUDE AND PHASE OF DWN(512)
CALL MAGPHS(DWSN,DWSNL,DWSNM,DWSNP)
C
C ERRMIN = 1.00-02
C
C CALL ERROR(TWLLIN,TWPHS5,DWSNL,DWSNP,ERRLIN,ERRPHS,ERRMAX)
C
C12 TW(I) = TWW(I,1)
C
C COMPUTE COMPLEX WEIGHTS OF NEW LEFT AND RIGHT SPLIT FINGERS
DO 100 I=1,NDATA
C
C MAGNITUDE RELAXATION FACTOR 'RLXM'
RLXM = 0.2500
C
C PHASE RELAXATION FACTOR 'RLXP'
RLXP = 0.0000
C
C IF(TWLNN(I) EQ.0.000) GO TO 14
C
C RATIO = (TWLLIN(I)-DWSNL(I))/TWLLIN(I)
C IF(DABS(RATIO).LE.1.00-03) ERRLIN(I) = 0.000
C CMULT = DSIGN(1.000,ERRPHS(I))
C IF(DABS(ERRPHS(I)).GE.5.000) ERRPHS(I) = 5.000*CMULT
C IF(DABS(ERRPHS(I)).GE.5.000) RLXP = 1.000
C IF(DABS(ERRPHS(I)).EQ.1.000) ERRPHS(I) = 0.000
C
C PHIK = DTR*ERRPHS(I)+RLXP
ABSL = DCABS(TWL(I)+RLXP+ERRLIN(I))
ABSR = DCABS(TWBI(I)+RLXP+ERRLIN(I))
TWL(I)=ABS*DCOS(THTA+PHIK)*SINE(I)*DCMPLX(DCOS(THTA),DSIN(THTA))
TWBI(I)=ABSR*DCOS(PHIK-THTA)*SINE(I)+DCMPLX(DCOS(THTA),DSIN(-THTA))
C
100 CONTINUE
C
C CALL NORM(TWL,TWL)
CALL NORM(TWBI,TWBI)
FILE COMPEN FORTRAN A BELL-NORTHERN RESEARCH

CALL MAGPHS(TWLN,TWLL,TWLM,TWLP)
CALL MAGPHS(TWNL,TWRL,TWRM,TWRP)

C DO 52 IT=1,NDATA
C TWLN(2,1) = TWLN(I)
C TWRM(2,1) = TWRM(I)
C TWNN = TWLN(I)+TWRM(I)
C TWW(2,1) = TWNN(I)
52 CONTINUE

CALL NORM(TWNN,TWNN)
CALL MAGPHS(TWNN,TWNNL,TWNNM,TWNNP)

C WRITE(10,2003)
C DO 67 I = 58,182
C RATIO = (TWNN(I)-DWSNL(I))/TWNN(I)
C WRITE(10,2002) I,TWNN(I),DUMRL(I),DUMRL(I),DULL(I),DULL(I),DWLP(I),
C &DWRP(I),DWSNL(I),DWSMM(I),RATIO,ERRPHS(I)
67 CONTINUE

C DO 69 I = 1,NDATA
C TW(I) = TWNN(I)
69 CONTINUE

C IT = IT+1
C WRITE(6,1003) IT,ERRMAX,ERRMIN
C IF(ERRMAX.LT.ERRMIN) GO TO 20
C IF(IT.EQ.ITMAX) GO TO 20
C IF(IT.EQ.ITMIN) GO TO 15
C
C WRITE(6,1003) IT,ERRMAX
C GO TO 10
C
C15 RLX = 0.0000
C RLXP = 1.0000
C WRITE(6,2001) RLXP,RLX
C GO TO 12
C
C PERFORM NUMERICAL CONVOLUTION BETWEEN THE INPUT AND OUTPUT IDT'S
C
C20 CALL CONVOL(DWSN,XDIF,H,MILN,MMAG,HPHS)

C COMPUTE THE FAST FOURIER TRANSFORM OF IMPULSE RESPONSE
C TO OBTAIN THE FREQUENCY RESPONSE
C CALL FFT2C(H,MM,IMK)
C
C NORMALIZE 'FFT' WRT. CENTER VALUE
C CALL FNORM(H,FFTN)
C
C COMPUTE MAGNITUDE LINEAR & DB AND PHASE
C
C COMPUTE MAGNITUDE LINEAR & DB AND PHASE
CALL MAGPHS(FFT1N,FFT1N,FFT1N,FFT1N,FFT1N)
C
GENERATE FREQUENCIES CORRESPONDING TO ALL 512 SAMPLES
CALL FRVXET(FR)
C
GENERATE TIME DELAY VECTOR 'T' CORRESP. TO ALL 512 SAMPLES
CALL TIME(T)
C
WRITE(10,2003)
C
PRINT APODIZATION DATA
C
DO 11 J = 1,NFO
   I = J+NFO
   TWWL(I) = DWLLL(I)*SINE(I)
   TWRD(I) = DWMRL(I)+SINE(I)
   RATIO = (TWWL(I)-DWSNL(I))/TWWL(I)
   WRITE(I,11) TWWL(I),TWRD(I)
11 CONTINUE
C
PRINT RESULTS OF FREQUENCY RESPONSE
C
DO 500 I=1,NDATA
   WRITE(9,2002) I,FR(I),FFTMAG(I),FTTPHS(I),MLIN(I),HPHS(I)
500 CONTINUE
C
GENERATE PLOTTING FILE
C
DO 550 I = 184,330
   FR(I) = FR(I)/1.00+06
   WRITE(3,2001) FR(I),FFTMAG(I)
550 CONTINUE
C
2001 FORMAT(1P20.10)
2002 FORMAT(1B.1P10D11.3)
2003 FORMAT(1H1)
C
1000 FORMAT(/,5X,'IS YOUR SUBSTRATE ISOTROPIC??? (1 * Y, 0 * N)')
1001 FORMAT(10,15.2016,T.1P20.6)
1002 FORMAT(9X,'4X,'0.4X,'F-RE',10X,'F-IM',10X,'Z=LIMIT',10X,'THTA')
   $)
1003 FORMAT(1S,1P914.6)
1004 FORMAT(1P3D20.10)
1005 FORMAT(/,9X,'4X,'0.4X,'F-RE',10X,'F-IM',10X,'Z=LIMIT',10X,'THTA')
   &, 9X, 'THTA(DEC)', 1)
1006 FORMAT(2X,130(' '))
1007 FORMAT(7X,105(' '))
1008 FORMAT(/,9X,'4X,'0.4X,'F-RE',10X,'F-IM',10X,'Z=LIMIT',10X,'THTA')
   &, 9X, 'THTA(DEC)', 1)
1009 FORMAT(5X,'A = ',1PD16.7)
1010 FORMAT(5X,'B = ',1PD16.7)
1011 FORMAT(/,5X,'I AM NOW COMPUTING DIFFRACTION EFFECT OF THE APODIZED'
   & DD 'FR (AS Y)')
1012 FORMAT(/,1P10D12.4)
1013 FORMAT(9X,'1.4X,'0.4X,'F-RE',10X,'ZL',10X,'F-REAL',10X,'BBSVAL')
FILE COMPEN  FORTRAN A  BELL-NORTHERN RESEARCH

1014 FORMAT(4X,'1.' ,4X,'THETA(\text{DEG})' ,5X,'PHI(\text{DEG})' ,6X,'VEL (M/S)' ,5X,'XX') COM03860
     & 12X,'XX1', 12X,'K-NBR', 8X,'GAM', 11X,'ABSVAL', 8X,'PHS(\text{DEG})') COM03870
1015 FORMAT(9X,'1.' ,4X,'FREQ(MHZ)' ,3X,'MAG(DB)' ,5X,'PHS(\text{DEG})' ,4X,
     & '\text{PHASE(\text{DEG})', 4X,'WT-WITH DIF', 8X,'\text{WT-NO DIF}') COM03880
1016 FORMAT(5X,'INPUT VECTOR (S12)') COM03890
1017 FORMAT(5X,'TIME VECTOR T(S12)') COM03900
1018 FORMAT(5X,'OUTPUT IDT VM(S12)') COM03910
1019 FORMAT(5X,'IMPULSE RESPONSE W(S12)') COM03920
1020 FORMAT(5X,'FAST FOURIER TRANSFORM FFT(S12)') COM03930
1021 FORMAT(5X,'THIS SUBSTRATE IS ISOTROPIC!!!') COM03940
1022 FORMAT(5X,'THIS SUBSTRATE IS AN-ISOTROPIC!!!') COM03950
1023 FORMAT(7X,'DO YOU WISH TO HAVE AN INCLINE IN THE APDIZED IDT??') COM03960
     & (1=YES, 0=NO)') COM03970
1024 FORMAT(5X,'HOW MANY DEGREES??') COM03980
1025 FORMAT(5X,'\text{THE APDIZED IDT HAS AN INCLINE OF } \beta(\text{DEG}) = ',
     & (1=IDTO, 4=//) COM04000
1026 FORMAT(10X,'BE PATIENT, PLEASE!!!') COM04010
1027 FORMAT(5X,'C = ', (1PO 16.7) COM04020
1028 FORMAT(5X,'D = ', (1PO 16.7) COM04030
1029 FORMAT(5X,'A = ', (1PO 16.7) COM04040
C
C
STOP
END

COM04050
COM04060
COM04070
COM04080
COM04090
SUBROUTINE CONVOL (VM, XI, W, WLIN, WMAG, WPHS)

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION WSVAL(512), WMAG(512), WPHS(512), WLIN(512)
COMPLEX*16 VM(512), SUM, W(512), XI(512)
COMMON /DATA/ PI, FO, FLAMB, DELAY, AP1, AP2, NFO, NFI, NDATA

CK2 = 180.000/PI

PERFORM NUMERICAL CONVOLUTION OF THE REAL INPUT VECTOR 'XX'
AND THE COMPLEX OUTPUT VECTOR 'AMAG'

SUM = DCMPLX(0.000, 0.000)

DO 100 I = 1, NDATA
   DO 200 J = 1, NDATA

   JJ = J-I+1
   IF (JJ LT 1) JJ = JJ+NDATA

   SUM = SUM+XI(JJ)*VM(J)

200 CONTINUE

   W(I) = SUM

SUM = DCMPLX(0.000, 0.000)

100 CONTINUE

   AMAX = CDABS(W(I))
   DO 300 I = 1, NDATA
       F = CDABS(W(I)).GT. AMAX
       AMAX = F .AND. CDABS(W(I))
300 CONTINUE

   DO 400 I = 1, NDATA

   W(I) = W(I)/AMAX

   MAGNITUDE LINEAR
   WLIN(I) = CDABS(W(I))

   MAGNITUDE (DB)
   WSVAL(I) = CDABS(W(I))
   IF (WSVAL(I).LE.0.000) WSVAL(I) = 1.00-12
   WMAG(I) = 20.000+LOG10(WSVAL(I))

   PHASE (DEG)
   IF (DIMAG(W(I)).EQ.0.000) GO TO 10
   WPHS(I) = CK2*DATAN2(DIMAG(W(I)), DREAL(W(I)))
   GO TO 400
10   IF (DREAL(W(I)).GE.0.000) WPHS(I) = 0.000
   IF (DREAL(W(I)).LT.0.000) WPHS(I) = 180.000

400 CONTINUE
FILE CONVOL FORTRAN A BELL-NORTHERN RESEARCH

C

RETURN

END

CON00560
CON00570
CON00580
SUBROUTINE DIF(NNN, KF, AR, BR, CR, DR, BETA, XOLD, YOLD, NUMBER)

C THIS PROGRAM MODELS DIFFRACTION ON LINBO3 SUBSTRATES IT CAN BE
C USED IN DESIGNING SURFACE ACOUSTIC WAVE FILTERS
C
C IMPLICIT REAL*(8A-H.O-Z)
C REAL*4 XLOW
C DIMENSION X(401), Y(401), XDLN(512), XDNG(512), XDPh(512), XR(512)
C DIMENSION ALEN(125), RO(125), BMAG(512), BMH(512), CBVA(1512)
C DIMENSION TK(101), H(7), YW(125)
C DIMENSION XOLD(125, 410), YOLD(125)
C COMPLEX*16 FM(125), FT(512), XI(512), WDI(25), WD(512), DW(512)
C COMPLEX*16 FFM(512), FTM(512), XO(512), XD(512), AN(401), YOLD(125, 410)
C COMMON /DATA/ PI, FO, FLAMBE, DELAY, AP1, AP2, NF0, NF1, NDATA
C
C DEFINE PARAMETERS
C
C KDATA = NDATA/2 + 1
C KF = (NF0 + 1)/2
C DELL = (AP1 - AP2)/2.000
C R0D = DELAY + FLOAT(KF) * 500
C APR = 45.000
C NN1 = 57
C CK1 = PI / 180.000
C CK2 = 180.000 / PI
C CK3 = PI / 2.000
C
C WRITE (6, 10011) NNN, KF, AR, BR, CR, DR, BETA
C
C INITIALIZE THE PLOTTING ROUTINE
C
C CALL SPLTVI ("PROFILES")
C CALL SPRLT
C CALL SDATE
C CALL SBOX
C CALL SPRN
C CALL SSAXIS ("WAVELENGTH (LAMBDA)", 125 , 25 , 0.10)
C CALL SYAXIS ("MAG.", 3, 2.0, 2.10)
C
C EXPLANATION OF SYMBOLS
C
C ALEN = TOTAL LENGTH OF RADIATING ELEMENT (FINGER OF IDT)
C RO = DISTANCE ALONG DIRECTION OF PROPAGATION (Z-AXIS)
C FROM POINT X=O,2*K TO POINT P WHERE THE COMPLEX
C FIELD STRENGTH IS CALCULATED
C RO-Q AT X=O,2*K
C XX = DISTANCE ALONG THE X-AXIS FROM X=O TO THE CENTER OF
C THE RADIATING ELEMENT, FINGER OF IDT, DETERMINES THE
C OFFSET OF THE FINGER POSITION WRT. X=O,2*K
C XXMAX, XXMIN = EXTREMES OF DISTANCE XX
C DELXX = INCREMENT OF XX WHICH DEFINES A NEW POSITION OF RADIATING
C FINGER WRT X=O,2*K FOR THE COMPUTATION OF COMPLEX
C FIELD STRENGTH AT POINT P
FILE DIFF FORTRAN A BELL-NORTHERN RESEARCH

Z = LIMIT OF INTEGRATION
F = COMPLEX FIELD STRENGTH,RESULT OF FRENSNEL'S INTEGRAL
F(X)=CL*EXP(-I*X)

XLLIM = -50 000
XLX = 50 000
DELXY = 0 2500

I AM NOW COMPUTING DIFFRACTION EFFECT OF THE APOIZED IDT
WRITE(6,10)**1
WRITE(6,1028)

PRINT HEADINGS TO PRINT OUT,PROFILES FI 701SH 'I'-PROF OUT
WRITE(7,1041)
WRITE(7,1006)

DD 100 I = 1,NNN
II = I-1
IF(KF GT 0) II = -(-1)**I
IF(KF LT 0) KF = -ABS(KF)

INITIALIZE PLOTTER COUNT
N = *
W = *

SET PLOT PAGE COUNT
NPLOT = 1

COMPUTE RADIATING FINGER'S LENGTH
ALENG(I) = SUBL(I)
BLENG = ALENG(I)*500

ABC = DFLOAT(KF+II)
XD = 500+DFLOAT(KF+II)*DTAN(BETA)

R0(I) = RO+DFLOAT(I)*500
XXLOW = XLLIM+XD*500
INTG = IFX(XXLOW)
XX = DFLOAT(INTG)
IF(BETA EQ 0 000) XX=XXLIM

AXSTOP = XX+XD
AXMAX = APR+500+XD
AXMIN = APR+500+XD

COMPUTE ANGLE OF PROPAGATION 'THTAO'
THTAO = DATAN(XX/RO(I))
THTAO = CM2*THTAO
THTAOA = DABS(THTAO)

WRITE(6,1001) I,N,THTAO,THTAOA

THTAO2 = THTAO+THTAO
THTAO3 = THTAO2+THTAO
THTAOA2 = THTAO2+THTAO2
THTAOA3 = THTAOA2+THTAOA2
THTAOA4 = THTAOA2+THTAOA2
THTAOA5 = THTAOA2+THTAOA2


DIF0110  THTAO6 = THTAO4+THTAO2
DIF0112  THTAO7 = THTAO6+THTAO
DIF0113  THTAO8 = THTAO6+THTAO2
DIF0114  C
DIF0115  GAMMA = -(1/R+THTAO2+B+THTAO4+C+THTAO8+D+THTAO8)
DIF0116  GAMMA = 1+GAMMA
DIF0117  C
DIF0118  COMPUTE THE PHASE VELOCITY OF PROPAGATION
DIF0119  VEL  = 3487 7600*(1.0000-GAMMA)
DIF0120  C
DIF0121  DVEL = 2*0.0+V+THTAO4+4*0.0+B+THTAO3
DIF0122  DVEL = 3487 7600*(DVEL+6.0+C+THTAO8+8.0+D+THTAO8)
DIF0123  C
DIF0124  COMPUTE THE POWER FLOW ANGLE 'PHEE'
DIF0125  PHEE = DATAN(DVEL/VEL)
DIF0126  C
DIF0127  PHE = CK2+PHEE
DIF0128  PH1 = DCOS(PHEE)
DIF0129  GAMMA = 1+GAMMA
DIF0130  C
DIF0131  COMPUTE ANGLE 'ALPHA' OF POWER FLOW PROPAGATION
DIF0132  ALPHA = THTAO+PHEE
DIF0133  ALPHA = CK2+ALPHA
DIF0134  C
DIF0135  COMPUTE DISTANCE XX1 WHERE POWER FLOW ARRIVES
DIF0136  XX1 = R0(I)+DATAN(ALPHA)
DIF0137  C
DIF0138  IF(XX1 .LT. XXL(IN)) GO TO 10,
DIF0139  IF(XX1 .GT. XXSTOP) GO TO 10
DIF0140  GO TO 15
DIF0141  C
DIF0142  10 F = DCMPLEX(0.000,0.000)
DIF0143  GO TO 16
DIF0144  C
DIF0145  UPPER LIMIT 'Z'
DIF0146  C
DIF0147  15 VU = (((XX1+BLENG)/DSORT(R0(I)+0.5D0))+DSORT(GAM))
DIF0148  ZU = CK3+VU+VU
DIF0149  C
DIF0150  COMPUTE FRENSNEL INTEGRAL 'F1' WITH UPPER LIMIT
DIF0151  CALL FRENS(IZ1,F1)
DIF0152  C
DIF0153  LOWER LIMIT 'Z'
DIF0154  C
DIF0155  VL = (((XX1-BLENG)/DSORT(R0(I)+0.5D0))+DSORT(GAM))
DIF0156  ZL = CK3+VL+VL
DIF0157  C
DIF0158  COMPUTE FRENSNEL INTEGRAL 'F2' WITH LOWER LIMIT
DIF0159  CALL FRENS(IZL,F2)
DIF0160  C
DIF0161  CHECK FOR SIGN OF F1 AND F2
DIF0162  IF((XX1+BLENG).LE.0.000) F1 = F1
DIF0163  IF((XX1-BLENG).LE.0.000) F2 = F2
DIF0164  C
DIF0165  SUBTRACT F2 FROM F1 TO OBTAIN FRENSNEL INTEGRAL F
DIF0166  F = F1-F2
ABSOLUTE VALUE OF FRESNEL INTEGRAL F = F1 - F2
BBSVAL = CDABS(F)

MULTIPLY F BY CONSTANTS IN FRONT OF INTEGRAL SIGN
C1 = 2 OD0*PI*ROI1*GAMA*PHI
C2 = DCMPLE(0.0, 0.0, C1)
F = F*(CDEM(C2)/DSQRT2.0*ROI1*GAMA*PHI))

MULTIPLY 'F' BY COUPLING COEFFICIENT 'K SQUARED'
CK0 = 1.0 OD0/(1.0 OD0/CK31+DABS(THTAO1))
F = F*CK0

DEFINE 'Y' VARIABLE FOR INTEGRATION
AN(N) = F

COMPUTE MAGNITUDE LINEAR
ABSVAL = CDABS(F)

COMPUTE PHASE IN DEGREES
IF(DIMAG(F) EQ 0.0 OD0) GO TO 8
APHS = CK2+DATAN2(DIMAG(F),DREAL(F))
GO TO 9

IF(DREAL(F) GE 0.0 OD0) APHS=0.0 OD0
IF(DIMAG(F) LT 0.0 OD0) APHS=180.0 OD0

COMPUTE INTENSITY OF DIFFRACTION PROFILE AT POINT P
VAL1 = ABSVAL-ABSVAL

ESTABLISH X AND Y AXIS OF PLOTTER
X(N) = XX1
Y(N) = ABSVAL

XOLD(I,N1) = XX1
YOLD(I,N1) = F

WRITE(7,1003) I,THTA,PHE,VEL,XX,XX1,GAMA,GAM,ABSV,APH
WRITE(7,1003) I,THTA,PHE,ALPH,XX1,ROI1,GAMA1,GAMA2,GAM

ESTABLISH COUNT FOR PLOTTER
N = N+1

INCREMENT DISTANCE XX TO COMPUTE NEW COMPLEX FIELD STRENGTH AT POINT P
XX = XX+DELLX

CHECK IF ALL COMPLEX FIELDS ALONG DISTANCE XX HAVE BEEN COMPUTED, REFER TO XX=XX+DELLX
IF(XX.LE.XXSTOP) GO TO 5

TOTAL NUMBER OF POINTS TO BE PLOTTED
N = N-1

NUMBER = N

FINISH
FILE DIF FORTRAN A BELL-NORTHERN RESEARCH

C WRITE(*,*) N. NUMBER
DIF02210
C DIF02220
C DIF02230
C CALL ON PLOTTING ROUTINES(SUPPLIED BY BNR'S SYSTEM)
DIF02240
C CALL $LINE(X,Y,N,O,O)
DIF02250
C CALL $TITLE(DIFRN PROFILE NO '21.0 15)
DIF02260
C CALL $SYNUMF(FLOAT(NPLD),0)
DIF02270
C CALL $NLTV
DIF02280
C DIF02290
C CONTINUE
DIF02300
C DIF02310
C DIF02320
C DIF02330
C DIF02340
C DIF02350
C 100 FORMAT(215,1P0D12.4)
DIF02360
C 101 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02370
DIF02380
C 102 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02390
DIF02400
C 103 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02410
DIF02420
C 104 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02430
DIF02440
C 105 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02450
DIF02460
C 106 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02470
DIF02480
C 107 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02490
DIF02500
C 108 FORMAT(9X,'I',4X,'U',8X,'F-RE',10X,'F-IM',10X,'Z-LIMIT',10X,'THTA'
DIF02510
DIF02520
C 109 FORMAT(5X,'A = ',1P1D,6.7)
DIF02530
C 10A FORMAT(5X,'B = ',1P1D,6.7)
DIF02540
C 110 FORMAT(/,5X,'I AM NOW COMPUTING DIFFRACTION EFFECT OF THE APODIZED IDT'
DIF02550
DIF02560
C 111 FORMAT(/,5X,'I AM NOW COMPUTING DIFFRACTION EFFECT OF THE APODIZED IDT'
DIF02570
DIF02580
C 112 FORMAT(10,1P1D012.4)
DIF02590
C 113 FORMAT(9X,'I',5X,'XX',10X,'RD',10X,'ZU',10X,'ZL',10X,'F-REAL'
DIF02600
DIF02610
C 114 FORMAT(4X,'I',4X,'THTA(DEG)',5X,'PHE(DEG)',6X,'VEL (M/S)',5X,'XX'
DIF02620
DIF02630
FILE DIFCOR FORTRAN A BELL-NORTHERN RESEARCH

SUBROUTINE DIFCOR(NDAT,KF,AR,BR,CR,DR,BETA,TW,DW)

THIS ROUTINE COMPUTES A NEW DIFFRACTION COMPENSATED DESIGN OF
SURFACE ACOUSTIC WAVE FILTER

IMPLICIT REAL*8 (A-H,O-Z)

DIMENSION NCDUNT(125),SUBL(125),ALEN(125),XOLD(125,410)
DIMENSION XX(500),XPROF(500),TXMAG(512),TXPHS(512),YPHS(500)
DIMENSION TXLIN(512),TXNAG(512),TXPHS(512),YLIN(500),YNAG(500)
DIMENSION YPHS(500),YNEW(500),YNEW(500),YNEW(500),YNEW(500)

COMMON /DATA/ PI,FO,FLAMB,DELAY,AP1,AP2,NFO,NFI,NDATA

DEFINE PARAMETERS

DTR = PI/180.000
RTD = 180.000/PI

NN1 = 57

DO 60 I = 1,NDAT

II = I

IF(NDAT EQ NFI) II = II+NN1

ALEN(I) = AP2*CDABS(TW(II))

DO 61 J = 1,N

SUBL(I) = ALEN(I)/DFLOAT(J)

IF(SUBL(I) LE 2.000) GO TO 6

61 CONTINUE

6 NCDUNT(I) = J

WRITE(10,2004) I,NCDUNT(I),ALEN(I),SUBL(I)

60 CONTINUE

CALL DIFCOR(NDAT,KF,AR,BR,CR,DR,BETA,SUBL,XOLD,YOLD,NUMBER)

GO THRU SORTING,SLIDING & INTEGRATION

DO 100 I = 1,NDAT

SUB = SUBL(I)

NVAL = NCDUNT(I)

100 CONTINUE

DO 200 J = 1,NUMBER

200 CONTINUE

DIF0010
DIF0020
DIF0030
DIF0040
DIF0050
DIF0060
DIF0070
DIF0080
DIF0090
DIF0100
DIF0110
DIF0120
DIF0130
DIF0140
DIF0150
DIF0160
DIF0170
DIF0180
DIF0190
DIF0200
DIF0210
DIF0220
DIF0230
DIF0240
DIF0250
DIF0260
DIF0270
DIF0280
DIF0290
DIF0300
DIF0310
DIF0320
DIF0330
DIF0340
DIF0350
DIF0360
DIF0370
DIF0380
DIF0390
DIF0400
DIF0410
DIF0420
DIF0430
DIF0440
DIF0450
DIF0460
DIF0470
DIF0480
DIF0490
DIF0500
DIF0510
DIF0520
DIF0530
DIF0540
DIF0550
FILE DIFCOR FORTRAN A BELL-NORTHERN RESEARCH

XX(J) = XOLD(I,J)
C
IF(INVAL EQ 1) GO TO 11
GO TO 13
C
11 DUMMY = DABS(XX(J))
IF(DUMMY GT 22 5000) GO TO 12
GO TO 13
C
12 YY(J) = DCMPLX(0 000,0 000)
GO TO 200
C
13 YY(J) = YOLD(I,J)
C
200 CONTINUE
C
IF(INVAL EQ 11) GO TO 10
GO TO 15
C
10 CALL INTEGR(NUMBER,XX,YY,SUM)
GO TO 20
C
15 CALL MGPHS(NUMBER,YY,YYLIN,YYMG,YPHSL)
C
CALL SORT(NUMBER,SUB,XX,YY,XNEW,YNEW,INT)
C
CALL MGPHS(INT,YNEW,YNEWL,YNEW,YNEWP)
C
CALL SLIDE(INVAL,INT,XNEW,YNEW,XPROF,YPROF,NTOT)
C
CALL MGPHS(NTOT,YPROF,YLIN,YMG,YPHS)
C
DO 240 K = 1,INT
WRITE(10,2002) K,XX(K),YYLIN(K),YYMG(K),YPHSL(K)
C &amp;YPROF(K),YLIN(K),YPHS(K)
C
240 CONTINUE
C
DO 241 L = 1,NTOT
WRITE(10,2002) L,XPROF(L),YLIN(L),YPHSL(L)
C
241 CONTINUE
C
WRITE(6,*) NTOT
C
CALL INTEGR(NTOT,XPROF,YPROF,SUM)
C
20 TOTL(I) = SUM
C
WRITE(10,2002) I,TOTL(I)
C
100 CONTINUE
C
IF(NDAT.EQ.NFI) GO TO 7
GO TO 8
C
7 CALL XPOS(NDAT,TOTL,DW)
GO TO 9
C
CALL POSITN(TOTL,DW,DWLM,DWPHS)

1000 FORMAT(/,5X,'IS YOUR SUBSTRATE ISOTROPIC?? (1 = Y, O = N)')
1001 FORMAT(5X,'A = ',1PD16.7)
1002 FORMAT(5X,'B = ',1PD16.7)
1003 FORMAT(5X,'C = ',1PD16.7)
1004 FORMAT(5X,'D = ',1PD16.7)
1005 FORMAT(5X,'THIS SUBSTRATE IS ISOTROPIC!!!')
1006 FORMAT(5X,'THIS SUBSTRATE IS AN-ISOTROPIC!!!')
1007 FORMAT(/,5X,'DO YOU WISH TO HAVE AN INCLINE IN THE APODIZED IDT??')
1008 FORMAT(5X,'HOW MANY DEGREES??')
1009 FORMAT(5X,'THE APODIZED IDT HAS AN INCLINE OF BETA(DEG) = ')

RETURN
END
SUBROUTINE ERROR(TWLINE, TWPHT, CWNSLP, CWSNPM, ERRLIN, ERRPHS, ERRMAX) ERR00010
ERR00020
ERR00030
ERR00040
ERR00050
ERR00060
ERR00070
ERR00080
ERR00090
ERR00100
ERR00110
ERR00120
ERR00130
ERR00140
ERR00150
ERR00160
ERR00170
ERR00180
ERR00190
ERR00200
ERR00210
ERR00220
ERR00230
ERR00240
ERR00250
ERR00260
ERR00270
ERR00280
ERR00290
ERR00300
ERR00310
ERR00320

ERRM = 0.000
DO 100 I = 1, NDATA
ERRLIN(I) = TWLINE(I) - CWNSLP(I)
ERRPHS(I) = TWPHT(I) - CWSNPM(I)
IF(CWNSP(I) LE -90.000 AND CWSNP(I) G E -180.000) ERRPHS(I) = 0.000
IF(CWNSP(I) GE 90.000 AND CWSNP(I) LE 180.000) ERRPHS(I) = 0.000
IF(DABS(ERRPHS(I)) GT ERRM) ERRM = DABS(ERRPHS(I))

CONTINUE

ERRMAX = DABS(ERRM)
RETURN
END
FILE FORTRAN A BELL-NORTHERN RESEARCH

SUBROUTINE FNORM(FFT,FFTN)

THIS ROUTINE NORMALIZES THE FAST FOURIER TRANSFORM(FFT) WRT ITS CENTER VALUE

IMPLCIT REAL*8(A-H,O-Z)
COMPLEX*16 FFT(512),FFTN(512),FMAX
COMMON /DATA/ PI,FO,FLAMBO,DELAY,AP1,AP2,NP2,NF1,NDATA

C

KDATA = NDATA/2+1

C

FMAX = FFT(KDATA)
DO 100 I=1,NDATA
  FFTN(I) = FFT(I)/FMAX
100 CONTINUE

C

RETURN
END
SUBROUTINE FRES(Z,F)

THIS SUBROUTINE COMPUTES THE FRESNEL INTEGRAL
BY USING A SERIES EXPANSION APPROXIMATION

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION A(12), B(12), C(12), D(12)
COMPLEX*16 P(12), Q(12), F, FIND, G, SUM, DUMMY
COMMON /DATA/, PI, FO, FLAMBD, DELAY, AP1, AP2, NF, NFI, NDATA

THE NUMERICAL VALUES OF THE FRESNEL COEFFICIENTS

A(1) = 1 595769140000
A(2) = 1 70200006
A(3) = -6 8085868540000
A(4) = -5 763610004
A(5) = 6 920691920000
A(6) = 1 689665700002
A(7) = -3 050485660000
A(8) = -7 575241900002
A(9) = 8 506637810000
A(10) = -2 563904100002
A(11) = -1 502309600001
A(12) = 3 440477900002

B(1) = 3.30008
B(2) = 4 2553875240000
B(3) = -9 28100005
B(4) = 7 78002040000
B(5) = 5 208950003
B(6) = 5 075612980000
B(7) = 1 383149700001
B(8) = -1 3637291240000
B(9) = -4 033492760001
B(10) = 7 022220160001
B(11) = 2 161959290001
B(12) = 1 954703100002

C(1) = 0 000
C(2) = 2 49339750002
C(3) = 3 9380006
C(4) = 5 770996003
C(5) = 6 89892004
C(6) = 9 497139603
C(7) = 1 948809000
C(8) = -6 784873003
C(9) = 2 4642004
C(10) = 2 102967003
C(11) = -1 21793003
C(12) = 2 33939004

D(1) = 1 99471140001
D(2) = 2 30008
D(3) = -9 351341003
FILE FRES FORTRAN A BELL-NORTHERN RESEARCH

D(4) = 2 300000D-05
D(5) = 4 85146600D-03
D(6) = 1 90321800D-03
D(7) = 1 717291400D-02
D(8) = 2 998408700D-02
D(9) = -2 792895500D-02
D(10) = 1 649730800D-02
D(11) = 5 59851500D-03
D(12) = 8 38386600D-04

N = 12
G = DCMLX(1,0DOO,:1,0DOO)
Z = DABS(Z)
LIMIT 'Z' FOR OVERFLOW ERROR 'IMD 208I'
IF(Z LE 1 D-12) Z=1 D-12
IF(Z GE 1 D+12) Z=1 D+12
FIND = DCMLX(0,0DOO,:Z)

CHECK IF LIMIT 'Z' IS GREATER THAN 4 0
IF(Z GE 4 0000) GO TO 10

PERFORM SERIES EXPANSION FOR FRESNEL INTEGRAL
SUM = DCMLX(0 0DOO,0 0DOO)
Z4 = 1 0DOO
DO 100 I=1,N
P(I) = (DCMLX(A(I),B(I)))+Z4
WRITE(3,1000) I,Z,Z4,P(I)
Z4 = (Z/4 0000)+Z4
SUM = SUM+P(I)
IF(Z4 LE 1 D-20) GO TO 110

100 CONTINUE

110 F = SUM+(DSQR(Z/4 0000))+(DCMLX(DCOS(Z),DSIN(Z))
DUMMY = CDAR(FIND)
WRITE(6,1000)
GO TO 20

FRESNEL INTEGRAL FOR Z>4 0
SUM = DCMLX(0 0DOO,0 0DOO)
Z4 = 1 0DOO
DO 200 J=1,N
P(J) = (DCMLX(C(J),D(J)))+Z4
FILE FRES FORTRAN A BELL-NORTHERN RESEARCH

C WRITE(3,1000) J,Z,Z4,P(J)
  Z4=14.000/Z+Z4
C    SUM=SUM+P(J)
C IF(Z4 LE 1.0D-20) GO TO 210
C 200 CONTINUE
C
C 210 F=SUM+(DSQR(D4,000/Z))+(DCMPLX(DCOS(Z),-DSIN(Z)))*G/2.000
C     DUMMY = CEXP(FIND)
C WRITE(6,1000) FIND,DUMMY
C FILE FORMAT(10,1P4D20,1D1)
C 20 RETURN
END
FILE FRVECT FORTRAN 4 BELL-NORTHERN RESEARCH

SUBROUTINE FRVECT(FR)

THIS ROUTINE GENERATES THE FREQUENCIES CORRESPONDING TO ALL
512 SAMPLES OF 700MHZ FILTER DESIGN

IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION FR(512)
COMMON /DATA/PI,FO,FLAMBDA,DELAY,AP1,AP2,NFO,NFI,NDATA

OMEGAS = 2.000*FO
DOMEGA = OMEGAS/NDATA

DO 100 I=1,NDATA
  J = I-1
  CMULT = DFLDAT(J)
  FREQ = CMULT*DOMEGA
  FR(I) = FREQ

100 CONTINUE

RETURN
END
FILE: INPIDT  FORTRAN  A  BELL-NORTHERN RESEARCH

SUBROUTINE INPIDT(AI XLIN, XPHS)

THIS PROGRAM GENERATES COMPLETE INPUT IDT DATA FILE

IMPLICIT REAL*8(A-H, O-Z)

DIMENSION XLIN(512), XPHS(512)

COMPLEX*16 Z(512)

COMMON /DATA PI, FO, FLAMB, DELAY AP1, AP2, NFG, NF, NDATA

ABBREVIATIONS

FF * CENTRE FREQUENCY

FR * BANDWIDTH TO ROLLOFF (2*F1)

NFI * NUMBER OF FINGERS OF INPUT TRANSDUCER

DELAT * SEPARATION OF INPUT AND OUTPUT IDT CENTER TO CENTER

DEFINE PARAMETERS

CK2 = 180.000 / (F1)

KF1 = NFI * FF

GENERATE INPUT IDT DATA

DO 101 I = 1, NF1

CMULT = DCMPLX(Z(I))

XI(I) = DCMPLX(cmult * 0.000)

101 CONTINUE

DO 102 I = KF1, NDATA

XI(I) = DCMPLX(0.000 - CMULT * 0.000)

102 CONTINUE

CM RADIUS (LINEAR)

XLI[I] = CDABS(XI[I])

CM PHASE (DEG)

IF (REAL(XII(I)) GE 0.000) XPHS(I) = 0.000

IF (REAL(XII(I)) LT 0.000) XPHS(I) = 180.000

103 CONTINUE

RETURN

END
SUBROUTINE INTEGRATE(N, Y, TOTAL)

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION X(500), Y(500), SUM, TOTAL
COMMON /DATA/ PI, FO, FLAMBDA, DELAY, AP1, AP2, NFO, NDATA

LIM = (N+1)/2
SUM = DCMPXY(0, 0, 0, 0)
DO 100 I = 1, N
J = I + 1
SUM = SUM + 5000*(Y(I) + Y(J-1) + Y(J-2) + Y(J-3) + Y(J-4) + Y(J-5))
J = J + 1
100 CONTINUE
TOTAL = SUM
RETURN
END
SUBROUTINE MAGPHS(IN LIN AMAG AHPS)

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION AMAG(5,2), AHPS(5,2), ALIN(5,2)
COMPLEX*16 AS(1:2)
COMMON DATA PK, PC, AMBC, DELA, AP, AP2, NFO, NF, NDATA

DO 30 BC = 1, NDATA

COMPUTE THE MAGNITUDE IN LINEAR
ABSVAL = CCABS(AS(1,1))
ALIN = ABSVAL

MAGNITUDE IN DB
IF(ABSVAL <= 0.0) ABSVAL = 0.1
AMAG(:) = 20.0 * LOG10(ABSVAL)

COMPUTE THE PHASE OF FOR IN DEGREES
IF(REAL(AMAG) <= 0.0) GO TO 50
APHS(:,:) = ACOS(AMAG(:,1)) 
GO TO 80

50 IF(REAL(AMAG) >= 0.0) AHS(:,:) = -BC
IF(REAL(AMAG) <= 0.0) AHS(:,:) = -BC
GO TO 80

80 CONTINUE
RETURN
END
FILE MGPHS FORTRAN A BELL-NORTHERN RESEARCH

SUBROUTINE MGPHS(NN,X,ALIN,AMAG,APHS)
C
IMPLICIT REAL*A-H,Z
DIMENSION AMAG(500),APHS(500),ALIN(500)
COMPLEX*16 X(500)
COMMON /DATA/ PI,FO,FLAMBD,DELAY,AP1,AP2,NF0,NFI,NDATA
C
DO 80 I=1,NN
C
COMPUTE THE MAGNITUDE IN LINEAR
ABSVAL = CDABS(X(I))
ALIN(I) = ABSVAL
C
MAGNITUDE IN DB'S
IF(ABSVAL LE 0.0D-12) AMAG(I) = 20.0D0+DLOG10(ABSVAL)
C
COMPUTE THE PHASE OF FTR IN DEGREES
C
IF(DIMAG(X(I)) EQ 0.0D0) GO TO 50
APHS(I) = (180.0D0/PI)*DAN2(DIMAG(X(I)),DREAL(X(I)))
GO TO 80
C
50 IF(DREAL(X(I)) GE 0.0D0) APHS(I) = 0.0D0
IF(DREAL(X(I)) LT 0.0D0) APHS(I) = 180.0D0
C
80 CONTINUE
C
RETURN
END
FILE MODPLT FORTRAN A BELL-NORTHERN RESEARCH

C THIS IS A PLOTTING ROUTINE USED WITH THE T4013 GRAPHICS TERMINAL
C DISPLAY 'PLOTFILE' T4013 TEXT
C TO GET A HARD COPY FROM THE PLOTTER DOWNSTAIRS, TYPE
C MERPLT 'PLOTFILE' ( PEN 80
C ![PLTOUT REAL *(A-H-O-Z)
C REAL+4 X(500),Y(500),FREQ(10)
C CALL $PLOTH('MODPLOT')
C CALL $DATE
C CALL $GRD
C CALL $AXIS('FREQ (MHZ)', 11.90, .5, .20)
C CALL $AXIS('MAG (DB)', 9.5, -105, .11)
C NPLT+1
C
C WRITE(6,1)
1 FORMAT('OENTER THE NUMBER OF LINES(10 MAX)'
READ(1,1)INF
DO 7 I=1,INF
WRITE(6,2)
2 FORMAT('OWHAT IS LINE NO',1X,'.1X','IN MHZ?'
READ(3,2,FREC(I))
WRITE(6,3)FREC(I)
3 FORMAT('OHOW MANY POINTS AT',1X,FREC,2,1X,'MHZ?'
READ(3,2,INF)
WRITE(6,4)
4 FORMAT('OENTER DATA POINTS(FREQ(MHZ), MAG(DB))'
DO 6 J=1,INF
READ(3,2,FREC(J),INF)
WRITE(6,5)FREC(J),INF
5 CONTINUE
6 IF(100 .LT. NF) GO TO 8
CALL $SLINE(FREC,INF,0.01)
7 CONTINUE
8 GO TO 9
9 CALL $TITLE('170MHZ FILTER ISO CPEN VS IDEAL\ PLOT NO
A.45.0.15)
CALL $SYNUM(FLOAT(NPLT),.0)
30 CALL $PLTD
C C900 FORMAT(1P2010,2)
C STOP
END
SUBROUTINE NORM(A,N)
C
C THIS SUBROUTINE NORMALIZES A COMPLEX VECTOR WRT ITS LARGEST VALUE.
C
IMPLICIT REAL*8 (A-H,O-Z)
COMPLEX*16 A(N),AMAX
COMMON /DATA/ PI,FO,FLAMB,DELAY,AP1,AP2,NFO,NFI,NDATA
C
INTDEL = 110
NFIC = (NFI+5)/2
NFOC = NFIC+INTDEL
C
AMAX = A(NFOC)
DO 200 I=1,NDATA
AN(I) = A(I)/AMAX
200 CONTINUE
C
RETURN
END
SUBROUTINE NORM(L(A,AN))

C THIS SUBROUTINE NORMALIZES A COMPLEX VECTOR WRT. ITS LARGEST VALUE

C

IMPLICIT REAL*8 (A-H,O-Z)
COMPLEX*16 A(S12),AN(S12)
COMMON /DATA/ PI,FO,FLAMBO,DELAY,AP1,AP2,NF0,NF1,NDATA

C

INTDEL = 110
NFIC = (NF1+1)/2
NFOC = NFIC+INTDEL

C

AMAX = CDABS(A(NFOC))
DO 200 I=1,NDATA
AN(I) = A(I)/AMAX
200 CONTINUE

C

RETURN
END
FILE OUTIDT FORTRAN A BELL-NORTHERN RESEARCH

SUBROUTINE OUTIDT(YW,WD)

THIS PROGRAM GENERATES COMPLETE OUTPUT IDT DATA FILE

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION YW(125)
COMPLEX*16 WD(125)
COMMON /DATA/ PI,FO,FLAMB,DELAY,AP1,AP2,NFO,NFI,NDATA

ABBREVIATIONS

FO = CENTRE FREQUENCY
F1 = BANDWIDTH TO ROLLOFF (2*F1)
NFI = NUMBER OF FINGERS OF INPUT TRANSDUCER
DELAY = SEPARATION OF INPUT AND OUTPUT IDT,CENTER TO CENTER

DEFINE PARAMETERS
KFO=NFO+1

GENERATE OUTPUT IDT DATA

DO 101 I=1,NFO
WD(I) = DCMPXYW(I),0.000)
101 CONTINUE

DO 102 I=KFO,NDATA
WD(I)=DCMPX(0.000,0.000)
102 CONTINUE

RETURN
END
SUBROUTINE POSITN(AMAG, VMAG, VPHS)

THIS PROGRAM POSITIONS THE OUTPUT IDT AT THE SPECIFIED DELAY
WRT. THE INPUT IDT

IMPLICIT REAL*(8)(A-H,O-Z)
REAL*4 DELAYS
DIMENSION VMAG(512), VPHS(512)
COMPLEX*16 AMAG(125), VM(512)
COMMON /DATA/P1, F0, FLAMBO, DELAY, AP1, AP2, NFO, NFI, NDATA

DEFINE PARAMETERS

CK2 = 180.0000D/PI
DELAGS = SNGL(DELAY)
INTDEL = IFIX(DELAYS)
IFC = (NFI+1)/2
KFC = (NFO+1)/2
NCO = IFC*INTDEL
N=1 = NCO*KFC
N=2 = N1+NFO+1

DO 100 I=1,NFO
J=I+NN1
VM(J) = AMAG(I)
100 CONTINUE

DO 200 I=1,NN2
VM(I) = DCMLAX(0.0000D,0.0000D)
200 CONTINUE

DO 300 I=NN2,NDATA
VM(I) = DCMLAX(0.0000D,0.0000D)
300 CONTINUE

AMAX = CDABS(VM(I))
DO 400 I=1,NDATA
IF(CDABS(VM(I)).GT.AMAK) AMAX=CDABS(VM(I))
400 CONTINUE

DO 500 I=1,NDATA
VM(I) = VM(I)/AMAX
50 CONTINUE

MAGNITUDE(LINEAR)
VMAG(I) = CDABS(VM(I))

PHASE(DEG)
IF(DIMAG(VM(I)).EQ.0.0000D) GO TO 50
VPHS(I) = CK2*DATAN2(DIMAG(VM(I)),DREAL(VM(I)))
GO TO 500

50 IF(DREAL(VM(I)).GE.0.0000D) VPHS(I)=0.0000D
IF(DREAL(VM(I)).LT.0.0000D) VPHS(I)=180.0000D

END
FILE POSITN FORTRAN A BELL-NORTHERN RESEARCH

500 CONTINUE
C
  RETURN
END
SUBROUTINE SLIDE(NCOUNT, INT, XOLD, YOLD, XPROF, YPROF, NTOT)

C THIS ROUTINE LINES UP AND ADDS Y-AXIS VALUES OF INDIVIDUAL PROFILES IN ORDER TO OBTAIN THE FINAL DIFFRACTION PROFILE

C

IMPLICIT REAL*(A-H,O-Z)
DIMENSION XOLD(500), XNEW(500), YPROF(500)
COMPLEX*16 YOLD(500), Y(500), YNEW(500), YPROF(500)
COMMON /DATA/ PI, FO, FLAMBDA, DELAY, AP1, AP2, NFD, NF1, NDATA

C

NCENTR = (INT+1)/2
NINC = 8
NHALF = NINC/2
N = NINC
M = NINC

C

INITIALIZE THE VALUES OF Y(500)
DO 50 I = 1, INT
    Y(I) = DCMPLX(0.0, 0.0)
50 CONTINUE

C

DO 100 I = 1, NCOUNT
    IM1 = I - 1
    NSTEP = NHALF - NCOUNT - NHALF - IM1 - NINC
    DO 200 J = 1, INT
        XX = DABS(XOLD(J))
        IF(XX GT 22.5) GO TO 200
        JJ = J + NSTEP
        XNEW(N) = XOLD(J)
        YNEW(N) = YOLD(JJ) + Y(N)
        Y(N) = YNEW(N)
        N = N + 1
200 CONTINUE

C

NTOT = N - 1
N = 1

C

DO 300 I = 1, NTO T
    XPROF(I) = XNEW(I)
300 CONTINUE
FILE SLIDE FORTRAN A BELL-NORTHERN RESEARCH

100 CONTINUE
C WRITE(10,2002)
C IF((NCOUNT/2)+2 EQ NCOUNT) GO TO 49
C DO 400 I = 1,NTOT
C IM1 = I-1
C J = 159+8*IM1
C K = 159+1*IM1
C L = 159-8*IM1
C WRITE(10,2001) I,J,K,L,XPROF(I),YOLD(J),YOLD(K),YOLD(L),XPROF(I)
C CONTINUE
C GO TO 48
C 49 DO 500 I = 1,NTOT
C IM1 = I-1
C J = 127+4*IM1
C K = 127-4*IM1
C WRITE(10,2001) I,J,K,XPROF(I),YOLD(J),YOLD(K),XPROF(I)
C CONTINUE
C 500 CONTINUE
C C2001 FORMAT(45,1P9,12.4)
C 2002 FORMAT(81I1)
C RETURN
END
FILE SORT FORTRAN A BELL-NORTHERN RESEARCH

SUBROUTINE EDIT(NBR, SUB, XOLD, YOLD, XNEW, YNEW, INTG)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION XOLD(500), XNEW(500), YLIN(500), YPHS(500)
      COMPLEX*16 YOLD(500), YNEW(500)
      COMMON /DATA/ PI, FO, FLAMB, DELAY, AP1, AP2, NFO, NFI, NDATA

      XSTOP = 50.000

      GENERATE NEW X-AXIS GRID FOR FINGERS LONGER THAN TWO WAVELENGTHS
      DELXO = SUB/8.000
      INT = XSTOP/DELXO+0.5000
      INTM1 = INT-1
      INTEG = INT+INTM1

      DO 100 J = 1, INT
         JT1 = J-1
         I = J+INTM1
         K = INT-JT1
         DUMMY = DFLOAT(JM1)+DELXO
         IF(DUMMY.GT.XSTOP) GO TO 100
         XNEW(I) = DUMMY
         XNEW(K) = DUMMY

      100 CONTINUE

      PERFORM LINEAR INTERPOLATION OF Y-AXIS VALUES
      DO 200 I = 1, INTEG
         IP1 = I+1
         XDUM = XNEW(I)
         DO 200 J = 1, NBR
         JP1 = J+1
         YO = DREAL(YOLD(J))
         Y1 = DREAL(YOLD(JP1))
         YOO = DIMAG(YOLD(J))
         Y11 = DIMAG(YOLD(JP1))
         XO = XOLD(J)
         X1 = XOLD(JP1)
         DELX = DABS(X1-XO)
         IF(XDUM.GE.XO.AND.XDUM.LE.X1) GO TO 200

      200 CONTINUE
C 20 R = DABS(XDUM-XO)/DELX
   DELY1 = Y1 - Y0
   DELY2 = Y11 - Y00
   YNEWL = Y0 + R*DELY1
   YNEWP = Y00 + R*DELY2
   GO TO 15
C 200 CONTINUE
C 15 YNEW() = DCMPLX(YNEWL,YNEWP)
C 150 CONTINUE
C RETURN
END
SUBROUTINE TIME(T)

THIS PROGRAM GENERATES COMPLETE TIME VECTOR CORRESPONDING
TO ALL 512 SAMPLES

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION T(512)
COMMON DATA PT, FO, FLAMBO, DELAY, AP1, AP2, NF1, NF2, NOCTA

ABBREVIATIONS
FO = CENTRE FREQUENCY
F = BANDWIDTH TO ROLL-OFF (2*F1)
NF = NUMBER OF FINGERS OF INPUT TRANSDUCER
NF1 = NUMBER OF FINGERS OF OUTPUT TRANSDUCER
D TIME = STEP IN TIME FROM FINGER TO FINGER
LEN = TOTAL LENGTH OF TRANSDUCER IN TIME
DELAY = SEPARATION OF INPUT AND OUTPUT IDT, CENTER TO CENTER

DEFINE PARAMETERS
NF = INT(NF1)+1/2
D TIME = 1/(2.0D0*FO)
E TIME = D TIME/2.0D0

GENERATE TIME VECTOR
DO 103 I=1,NOCTA

TEST FOR EVEN NF1
IF(NF1.2+2 EQ NF1) GO TO 10

CALCULATION FOR ODD NF1
J=1-NF1
C MULT = D F FLOAT(J)
TIME = C MULT + D TIME
TIME = TIME + TIME
GO TO 103

CALCULATION FOR EVEN NF1
J=1-NF1,2
C MULT = D F FLOAT(J)
TIME = C MULT + D TIME - E TIME
TIME = TIME + TIME

103 CONTINUE
RETURN
END
FILE XPOS FORTRAN & BELL-NORTHERN RESEARCH

SUBROUTINE XPOS(N,X,XN)
C
C THIS ROUTINE WILL POSITION THE INPUT IDT AT THE FIRST N SAMPLES
C OF THE 512 SAMPLE VECTOR XN(512)
C
C IMPLICIT REAL*8 (A-H,O-Z)
C COMPLEX*16 X(125),XN(512)
C COMMON /DATA/ PI,FO,FLAMBO,DELAY,AP1,AP2,NFO,NFI,NDATA
C
C DO 100 I = 1,N
C XN(I) = X(I)
100 CONTINUE
C
C NP1 = N+1
C
C DO 200 I = NP1,NDATA
C XN(I) = DCMPLX(0.000,0.000)
200 CONTINUE
C
C RETURN
C
END
END
12·06·86
FIN