A computer controlled time domain reflectometry multiplexer
by Rodney David Carlson

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering
Montana State University

Abstract:
This paper discusses the design, implementation, and experimental verification of a one to thirty-two Time Domain Reflectometry (TDR) switch. The results of the investigation can be summarized. First, the device required a PC as a controller. Two separate programs were written for device drivers of the multiplexer. A C program permitted a TSR (Terminate Stay Resident) mode, while a Basic program allowed portable code with existing conductivity programs already written in that language.

Second, the multiplexer had to be designed with a large bandwidth and flat frequency response to prevent skewing caused by harmonic distortion. This allowed resolution of closely spaced reflections. All of these were resolved by carefully constructing good matches with the circuit board and high frequency relays. Nevertheless, the device still demonstrated fairly large power losses when taking into account several reflections, resulting in error for conductivity measurements made with the Giese and Tiemann analysis. A more complete formula was derived to account for the multiplexer in the TDR sytem, thus the multiplexer reflection and loss were modeled into the Giese and Tiemann theory. Experimentation verified the new formulas for calculating conductivity with the multiplexer present.

A COMPUTER CONTROLLED TIME DOMAIN REFLECTOMETRY MULTIPLEXER

by

Rodney David Carlson

A thesis submitted in partial fulfillment
of the requirements for the degree

of

Master of Science

in

Electrical Engineering

MONTANA STATE UNIVERSITY
Bozeman, Montana

May 1995

APPROVAL

of a thesis submitted by

Rodney David Carlson

This thesis has been read by each member of the thesis committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the College of Graduate Studies.

5-16-95
Date

Bruce R. McLeod
Chairperson, Graduate Committee

Approved for the Major Department

5-16-95
Date

Head, Major Department

Approved for the College of Graduate Studies

5/29/95
Date

Graduate Dean

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Montana State University-Bozeman, I agree that the Library shall make it available to borrowers under rules of the Library.

If I have indicated my intention to copyright this thesis by including a copyright notice page, copying is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for permission for extended quotation from or reproduction of this thesis in whole or in parts may be granted only by the copyright holder.

Signature Rod D. Carlson

Date 5-17-95

ACKNOWLEDGMENTS

The author would like to thank the entire Electrical Engineering department, my employers and fellow workers at Tektronix, and the taxpayers for making all this possible. The author would especially like to express his appreciation for Dr. Bruce R. McLeod. Most of all the author would like to dedicate this achievement to his mother for her endless emotional and physical investment, his father for years of encouragement and financial support, his older brother for being such an inspiration, and his maker for this life and the opportunity accompaning it.

TABLE OF CONTENTS

TABLE OF CONTENTS--CONTINUED

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES --CONTINUED

ABSTRACT

This paper discusses the design, implementation, and experimental verification of a one to thirty-two Time Domain Reflectometry (TDR) switch. The results of the investigation can be summarized. First, the device required a PC as a controller. Two separate programs were written for device drivers of the multiplexer. A C program permitted a TSR (Terminate Stay Resident) mode, while a Basic program allowed portable code with existing conductivity programs already written in that language.

Second, the multiplexer had to be designed with a large bandwidth and flat frequency response to prevent skewing caused by harmonic distortion. This allowed resolution of closely spaced reflections. All of these were resolved by carefully constructing good matches with the circuit board and high frequency relays. Nevertheless, the device still demonstrated fairly large power losses when taking into account several reflections, resulting in error for conductivity measurements made with the Giese and Tiemann analysis. A more complete formula was derived to account for the multiplexer in the TDR sytem, thus the multiplexer reflection and loss were modeled into the Giese and Tiemann theory. Experimentation verified the new formulas for calculating conductivity with the multiplexer present.

CHAPTER 1

INTRODUCTION

The following project originated as a joint venture between the Electrical Engineering and the Plant and Soil Departments at Montana State University, consisting of the development of a computer controlled Time Domain Reflectometry (TDR) multiplexer. Specifically, measuring soil conductivity with TDR at several locations required the operator to manually connect the probe of interest into the instrument. A high frequency multiplexer controlled by a PC and discussed in this paper reduced the amount of manual switching and thus the measuring times. In addition, the multiplexer increased the capacity of a single TDR instrument by the number of multiplexing channels for a fraction of the cost of buying more TDR instruments.

The high frequency performance of the multiplexer was critical. Insertion loss and mismatch between the instrument and multiplexer needed to be as low as possible. These losses were minimized by keeping the transmission lines as short as possible and making a fifty ohm microstrip match between the TDR, multiplexer, and soil probes. In addition, when a multiplexer had small reflections coupled with losses, the effect of reflection and loss drastically changed conductivity measurements. Existing multiplexer technology had to be studied to improve the insertion loss and bandwidth of this product.

A TDR multiplexer, from Cambell Scientific, consisted of four output channels. Its design employed high frequency switching relays embedded in a 50 ohm microstrip matched circuit board. Although the switcher worked for automated measurements, it was limited to only four probes. According to Dr. Jon Wraith (a soil scientist at MSU), more than twenty soil probes should be measured at numerous locations so that an accurate statistical sample can be calculated for conductivity in the field. Likewise, if

an environmentalist wanted to check for leakage from nuclear containment facilities, he or she would sample a grid of points surrounding the container. The existing one to four multiplexer design wasn't sufficient to meet either test case so a new multiplexer was designed to multiplex the TDR signal from one input to one of 32 output probes.

A serious design problem resulted from the BNC contact fatigue, caused from connecting probes to the connectors on the circuit board. The fatigue resulted in loose BNC connectors, thus epoxy was added around the base of the BNC connectors to distribute the connecting forces across the board surface. Furthermore, since the multiplexer was to be used in the field, prolonged battery power consumption had to be avoided. A car battery was an acceptable field power supply. But if the continuous current consumption was too large, even a heavy duty car battery would become dead within a few hours of use.

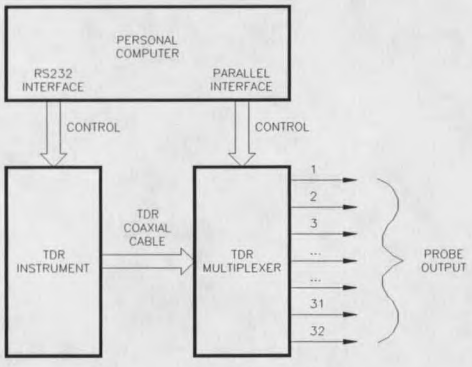


Figure 1. Block Diagram of PC Control

Figure one displays the block diagram for the operation of the entire TDR system. Notice that the multiplexer was designed to operate from the parallel port (25 pin D-subminiature) of the PC. This allowed word serial communication with the Tektronix 1502C TDR instrument through the RS232 port. The programs to support switching the TDR consisted of Basic function calls, since most of the

conductivity algorithms were written in that language. However, C device controlling programs were used to implement a DOS and Windows visual toggling interface.

To summarize, the predominant issues of need in this TDR multiplexer design included: minimizing loss and reflections, PC controlling software, resistance to contact fatigue, and portability with respect to power consumption.

CHAPTER 2

BACKGROUND INFORMATION ON THE TDR MULTIPLEXER

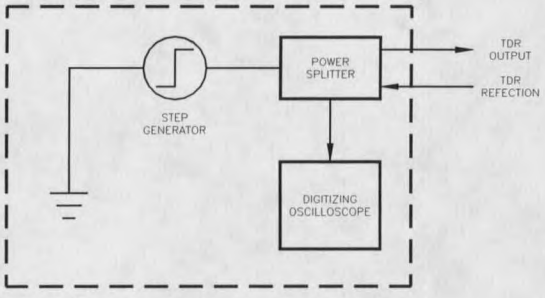
Theory of Time Domain Reflectometry

This section begins by discussing the basics of reflection theory employing TDR. Most of the theory traces back to Tektronix laboratories, in the early 1960's. Allen Zimmerman, Gordon Long, George Frye, and James A. Strickland were considered some of the fathers of the TDR pulse technique [1]. Initially, TDR was developed for testing the impedances of coaxial transmission lines. But, like most other technologies, it has since migrated into several fields. It's common to find TDR in fiber optics, power electronics, and even soil science applications.

Time Domain Reflectometry displays amplitude of reflection versus time of propagation on the y and x axes respectively. A simple model of TDR would be something like a "radar" detector. An initial electromagnetic pulse is emitted and the time required for the wave to travel to, reflect off a distant object, and return is recorded. Likewise, TDR instruments generate voltage pulses and the amplitude and reflection times are recorded and graphed.

The genius behind pulsed TDR involves splitting a repetitive square wave into two separate wave propagations. Part of the initial square wave amplitude is delivered directly to the oscilloscope. This establishes an amplitude and time reference to which all reflection events are compared. The other, propagating wave, continues to travel down the unknown transmission line, and when it encounters a change in impedance, a reflected wave returns to the oscilloscope. The initial and reflected voltages superimpose at some time $t_o + 2 \cdot \Delta T$, where ΔT represents the total time of the pulse wave to travel to the

discontinuity in impedance. The variable t_o denotes the time when the pulse achieves its 90% rise time for a matched load. A typical TDR device consists of a pulser, a power divider, and an oscilloscope as depicted in the Figure 2 below.



.Figure 2. The TDR Instrument

A conventional DC resistance circuit model can help one to derive reflection expressions for TDR [1]. Suppose that a battery source of 1 volt has an internal resistance R_G and is connected to a transmission line of resistance R_{LINE}, as portrayed by Figure 3. Furthermore, applying matched conditions for maximum power transfer to the transmission line requires R_G and R_{LINE} to be equal. Since the battery is 1 volt, the voltmeter inserted across R_{LINE} will indicate a half a volt when the switch is closed.

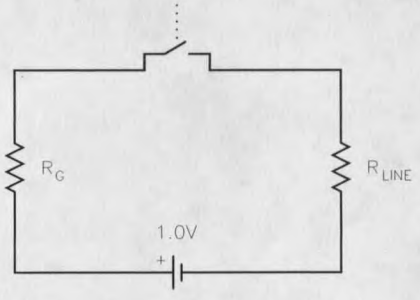


Figure 3. TDR Resistance Model

The time dimension can now be added to the existing DC circuit by addition of zero resistance wires. Replacing the battery source by a stepped generator gives a DC circuit the appearance of TDR (Figure 4).

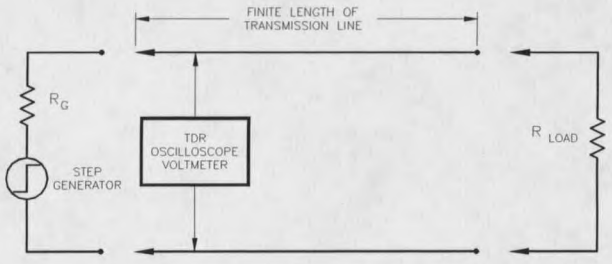


Figure 4. Model with Extension of Transmission Wires

Suppose that the resistance R_{LOAD} at the end of the zero resistance wires is infinity. The pulse generator then steps positive by one volt, but since the forward propagating wave hasn't reached the open at the end of the circuit, the voltmeter reads a half a volt. The forward moving wave reaches the open and the reflection will have the same magnitude and sign as the forward propagating wave [1]. At some time $t_o + 2 \cdot \Delta T$ the voltmeter will measure one volt, the sum of both the reflected and initial waves.

The actual experiment realistically approaches TDR by replacing the zero resistance wires in Figure 4 above with a transmission line having a constant impedance such as (50 ohm coaxial cable). The source must possess a 50 ohm resistance. A somewhat more complete method of handling signal reflections can then be explained. At the start of the pulse, the voltage would see 50 ohms from both the source and the coaxial line. The source supplies 1 volt but is split by the voltage divider according to the equation below:

$$V_o = V_G \cdot \frac{R_{LINE}}{R_G + R_{LINE}} = (1 \text{ Volt}) \cdot \frac{50}{50+50} = \frac{1}{2} \text{ Volt}. \qquad (1)$$

Time proceeds and R_{LINE} changes from 50 ohms to infinity (R_{LOAD}), the resistance of the open at the end of the coaxial cable. Taking the limit as R_{LOAD} approaches infinity, the voltage divider equation shows the output voltage converging to one volt. Thus, the numerical result agrees with the wave propagation model developed earlier.

$$V_o = \lim_{R_{LOAD} \to \infty} \left[V_G \cdot \frac{R_{LOAD}}{R_G + R_{LOAD}} \right] = 1 \text{ Volt} \tag{2}$$

Another extreme in TDR would be to terminate the coaxial cable with a short. From a wave model, the introductory pulse height would again be a half a volt. Then, the forward moving wave encounters the short at the end of the cable. It is returned with the same magnitude but opposite sign, canceling the initial voltage at the meter. The resistance divider once again confirms the reasoning; as the R_{LOAD} approaches zero so does the measured voltage.

$$V_o = \lim_{R_{load} \to 0} \cdot \left[V_G \cdot \frac{R_{LOAD}}{R_G + R_{LOAD}} \right] = 0 \text{ Volt} \tag{3}$$

The extreme cases of an open and a short prove that any other load should have a superposition voltage between one and zero volts as illustrated in Figure 5.

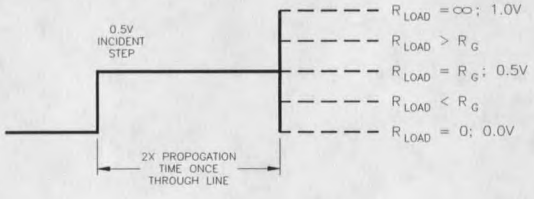


Figure 5. TDR Display for Various Load Resistances

A more convenient method of handling signal reflections is to define a reflection coefficient as the ratio of reflected to incident voltages

$$\rho = \frac{V_{REFLECTED}}{V_{INCIDENT}} = \frac{V_{OUT@LOAD} - V_{OUT@LINE}}{V_{OUT@LINE}} \tag{4}$$

The voltages $V_{OUT@LINE}$ and $V_{OUT@LOAD}$ are calculated from the divider equations listed below, before and after reflection from the load. Notice that these equations are derived from (1) and (2) assuming matched conditions between the source and line impedances ($R_G = R_{LINE}$).

$$V_{OUT@LINE} = \frac{1}{2} \cdot V_G \tag{5}$$

$$V_{OUT@LOAD} = V_G \cdot \frac{R_{LOAD}}{R_{LINE} + R_{LOAD}} \tag{6}$$

Substituting these equations into (4) and reducing produces (7) which is nothing more than the reflection coefficient common to microwave theory.

$$\rho = \frac{R_{LOAD} - R_{LINE}}{R_{LOAD} + R_{LINE}} \tag{7}$$

We therefore have the complete theory for a TDR system. As a final comment, the reflection coefficient (ρ) is usually expressed in impedances instead of resistances to account for inductance and capacitance

(8). In order to stay consistent with notation used later in this paper, the subscripts load and line will be replaced by s and μ in the general impedance equation.

$$\rho = \frac{Z_S - Z_\mu}{Z_S + Z_\mu} \tag{8}$$

With the basic theory explained, it is now appropriate to discuss the mathematics behind measuring conductivity with a TDR instrument provided by Giese and Tiemann [2], Dalton [3], and Topp [4]. The reader will then be familiar with the concepts and the key design parameters for a TDR multiplexer. Then, this chapter will continue with the high frequency matching techniques required. Functional design of software, hardware, and schematics is presented at the end of this section.

Theory for Measuring Conductivity and Water Content of Soil Samples

There are several methods of determining conductivity from TDR measurements, but the Giese and Tiemann analysis is by far the most prevalent in soil science [2]. It accounts for multiple reflections and loss inside the probe. In fact, due to its popularity, the other methods will only be mentioned in this article, not derived. After deriving the Giese and Tiemann conductivity equation, it will become obvious that only the initial and final voltages are necessary to determine the conductivity of any sample.

At this point, it would be wise to present a typical measuring configuration shown in Figure 6. As previously discussed, the TDR instrument consists of a spliter, pulser, and oscilloscope. The TDR is connected to the probe via a coaxial cable, and the probe can be one of two types. The first would be two parallel wire conductors, resembling a set of monopole antennas. One of these wires would connect to the cable's ground and the other to its center conductor (Figure 7). The second type of probe consists of an air core coaxial transmission line (Figure 8) filled with the sample to be tested.

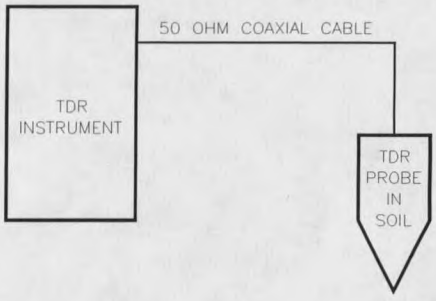


Figure 6. Typical Measuring Configuration

Although the set of monopoles are the easiest to deploy into the ground, it should be noted that the Giese and Tiemann equation is derived for coaxial probes only [2]. Using the set of wire antennas as a probe, and then calculating conductivity with the Giese and Tiemann equation introduces error.

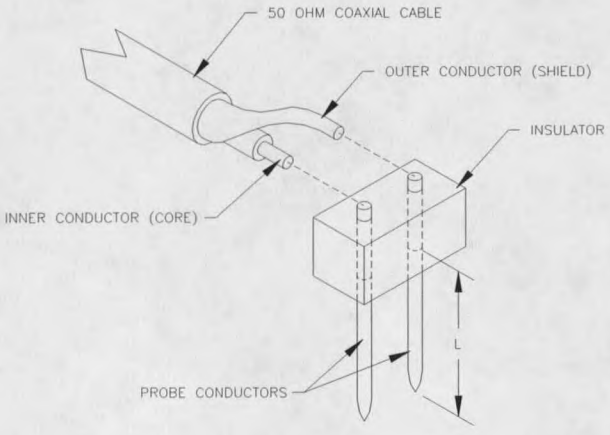


Figure 7. Parallel Conductor Probe

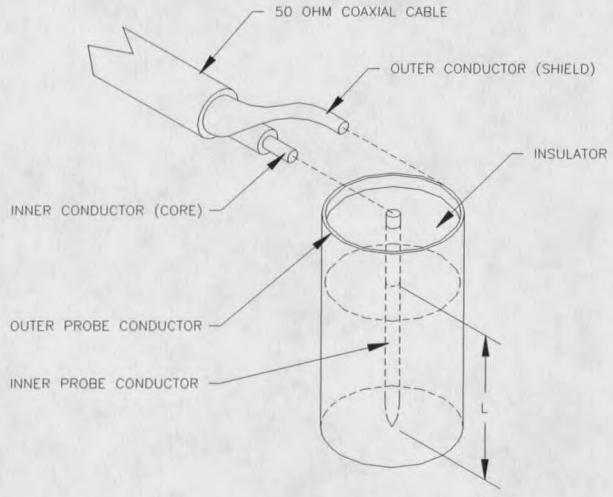


Figure 8. Coaxial Probe

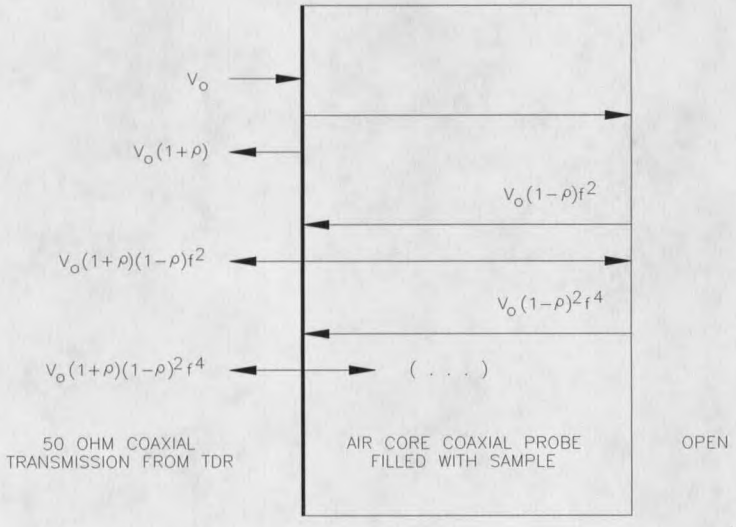


Figure 9. Cable and Probe Reflecting Interface

Suppose that we have a TDR instrument connected through cable to the "coaxial" style soil probe. If the cable impedance equals the source impedance of the TDR instrument, no reflections will be observed at the source and cable interface. On the other hand, the cable and probe interface will exhibit an impedance discontinuity, the boundary reflection depending on the dielectric and conductivity properties of the soil as well as the impedance of the cable. This boundary between the cable and the probe is viewed in Figure 9. Notice that multiple reflections occur inside the soil probe. A TDR response is shown along with the reflection diagram (Figure 10).

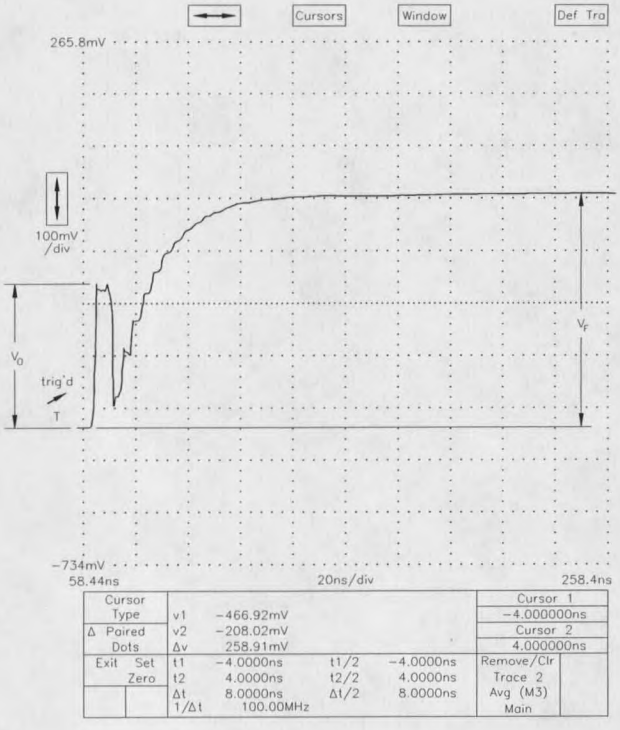


Figure 10. Typical TDR Response Curve

The diagram (Figure 9) is explained in this paragraph. A pulse is inserted with voltage V_0 and with time, the signal at the oscilloscope converges to a final voltage V_F. Summing reflected voltages to

the left of the probe in Figure 9 yields the final voltage V_F detected by the oscilloscope. Observe that the signal experiences a "gain" f in the soil, (the terminology is deceptive because "f" has a value less than one and thus accounts for losses caused by conduction).

$$V_F = V_o(1+\rho) + V_o(1+\rho)(1-\rho)f^2 + V_o(1+\rho)(1-\rho)^2 f^4 + V_o(1+\rho)(1-\rho)^3 f^6 + V_o(1+\rho)(1-\rho)^4 f^8 + \ldots \quad (9)$$

Reducing by factoring

$$V_F = V_o(1+\rho)\left[1 + (1-\rho)f^2 + (1-\rho)^2 f^4 + (1-\rho)^3 f^6 + (1-\rho)^4 f^8 + \ldots\right] . \quad (10)$$

By identifying the terms within the bracket as having recursive form, the infinite expansion can be simplified with the geometric series

$$\frac{1}{1-\gamma} = 1 + \gamma + \gamma^2 + \gamma^3 + \ldots \quad (11)$$

and

$$\gamma = f^2(1-\rho) . \quad (12)$$

The geometric series presumes the term γ is less than one, otherwise the equality cannot be applied to equation (11). The gain (f) always helps in meeting this condition, since f is less than or equal to one. However, the reflection coefficient ρ can be negative $(-1 \le \rho \le +1)$, and thus γ could exceed positive

one. The reader should understand the implications of this statement. Valid data can only be extracted for positive reflection coefficients, and from equation (8) this also means that probe impedance must be greater than the impedance for the coaxial transmission line. Considering these limitations, equation (10) is reduced to a finite form from the geometric series.

$$V_F = V_o(1+\rho)\left[\frac{1}{1-f^2(1-\rho)}\right]. \tag{13}$$

The reflection coefficient ρ has already been established, but is restated for convenience

$$\rho = \frac{Z_S - Z_\mu}{Z_S + Z_\mu}. \tag{14}$$

Replacing the reflection coefficient in equation (13) with equation (14) and then reducing produces (15)

$$\frac{V_F}{V_o} = \frac{Z_S}{\frac{1}{2}(Z_S + Z_\mu) + f^2 Z_\mu}. \tag{15}$$

Now, the second step in deriving the Giese and Tiemann formula is presented. The gain f can be expressed as a Taylor Series, neglecting everything except the first two terms. Because the truncated expansion requires small lengths (αL) for validity, the derivation is only applicable to thin sample analysis.

$$f^2 = e^{-2\alpha L} \approx 1 - 2\alpha L \tag{16}$$

The attenuation α equates to the following expression. The proof is provided in Appendix A.

$$\alpha = \frac{\sigma_T}{2\varepsilon_o c (K')^{\frac{1}{2}}} \tag{17}$$

Replacing α in equation (16) by equation (17), the following equation is obtained

$$f^2 = 1 - \frac{\sigma_T L}{\varepsilon_o c (K')^{\frac{1}{2}}}. \tag{18}$$

Dividing the characteristic impedance of an air core probe by the square root of the soil's dielectric constant $(K')^{0.5}$, gives the characteristic impedance of the same dimensioned probe filled with that soil instead of air as

$$Z_S = \frac{Z_o}{(K')^{0.5}}. \tag{19}$$

Solving for the ratio $\frac{Z_o}{Z\mu}$, by substituting equations (18) and (19) into equation (15) yields

$$\frac{Z_o}{Z_\mu} = \frac{\left(2\frac{\sigma_T L}{\varepsilon_o c} - (K')^{0.5}\right)}{\left(2\frac{V_o}{V_F} - 1\right)}. \tag{20}$$

Until now, this development has placed no restrictions on the type of probe. This changes when the relative dielectric constant K' is deduced for a coaxial probe (Appendix B).

$$(K')^{0.5} = \frac{\sigma_T L}{\varepsilon_o c} \tag{21}$$

By substituting the expression for the soil dielectric constant K' into the equation (20) and solving for the conductivity σ_T, the Giese and Tiemann proof is completed [2]. Thus

$$\sigma_T = \frac{\varepsilon_o c}{L} \cdot \frac{Z_o}{Z_\mu} \cdot \left(2\frac{V_o}{V_F} - 1\right) \tag{22}$$

where:

ε_o = Permitivity of free space= $\frac{1}{36 \cdot \pi} x10^{-12}$ F / m

c = Speed of light in free space = $3x10^8$ m/s

L = Depth of probe into the soil

= (Thin sample) m

Z_o = Probe impedance with air as dielectric

= (usually ≈ 50) ohms

Z_μ = Probe impedance with a dielectric before soil sample

= (usually = Z_o ≈ 50) ohms.

Table 1 states the requirements for the Giese and Tiemann analysis.

Condition	Reason
Positive Reflections at Cable and Probe Interface	Geometric Series Doesn't Converge to a Finite Solution for Negative Reflections.
Probe of Short Length	Error for Longer Lengths Caused By Truncating the Taylor Series Expansion
"Coaxial" Style Probe	Proof Depends on this Assumption

Table 1. Conditional Requirements of the Giese and Tiemann Analysis

Dalton and others proposed measuring the dielectric constant of the soil from TDR and correlating this result with the conductivity to evaluate the sample's water content [3]. To find the apparent dielectric constant K' from the travel time [4], Topp et al used the equation (23).

$$v = c / K'^{0.5} \tag{23}$$

The variable v denotes the velocity and it can be determined (Figure 11) from the elapsed time t_{AB} , the time between multiple reflections in the probe [5].

$$v = \frac{2L}{t_{AB}}. \tag{24}$$

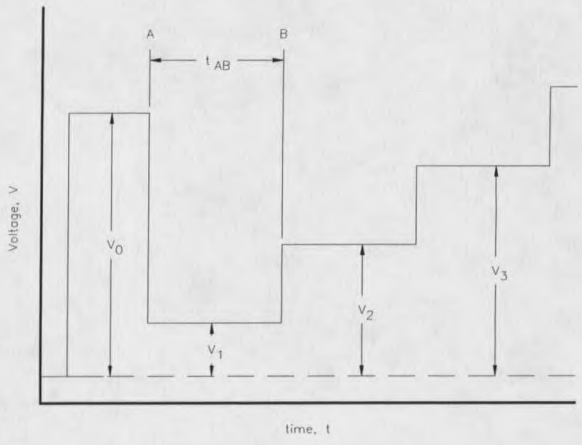


Figure 11. Measuring the Propagation Time between Multiple Probe Reflections

As a final note, the other methods for calculating conductivity introduce larger error than the Giese and Tiemann analysis [2], because they rely on a finite number of reflections in the sample. This causes more inaccuracy in taking the measurements. The resolution of the TDR instrument is the limiting factor of a measurement made with a finite number of reflections. This is an important consideration in ignoring these types of measurements with multiplexed signals, even though they may eliminate dispersion caused by the switch matrix.

Device Specifications

Utilizing the foregoing problem statement and application theory, the next step was to outline the preliminary design specifications for the multiplexer. These specifications resulted from considering both the conductivity and the dielectric constant (water content) requirements.

1. Both conductivity measurements with and without the TDR multiplexer had to agree within a 1:0.95 linear slope fit.

2. Insertion power losses optimally could not exceed more than 3 dB, for the entire 1.2 GHz (10^9 Hz) bandwidth.

3. Absolute maximum multiplexer reflection had to be below a 10% margin.

4. The BNC connectors and board material needed to be resistant to fatigue caused from connecting and removing probes.

5. For portability reasons, the external current consumption could not exceed 2 amps for a relay switching period of no more than one second.

6. The parallel port was used for communication with the multiplexer, and controlling software was written in basic as well as C programming languages.

Now that the preliminary design specifications have been outlined, the various design selection criteria and alternatives will be presented.

Minimizing Losses and Reflections in the Design

The first step in the design process was to separate the project into strategic areas. The most important issues were determined from the parameters outlined in design specifications. From the list of six specifications, the first three parameters were emphasized, addressing the insertion and reflection losses. Without meeting these conditions the entire multiplexer would be a failure.

Next, the type of switch was selected. Early in the design, the relay switching mechanism was chosen over solid state electronics, because solid state electronics possess unpredictable stray inductances and capacitances that reduce the TDR bandwidth. Tektronix manufactures transistor and diode TDR electronics in a proprietary hybrid circuit, thus reducing these ceramic packaging capacitances and inductances. This technology could not be implemented into the multiplexer without making the product cost prohibitive. However, experience gained in building and testing the TDR switch has convinced the author that the entire 1.2 GHz bandwidth isn't necessary to make accurate measurements. For this reason, solid state switching mechanisms could be considered in alternative designs.

Considering the simplicity of relay switching, several different layouts were investigated. Board constraints prevented the assembly from extending beyond two layers. The prototype cost increased from seventy dollars for two layers, to nearly four hundred dollars for three layers. A mistake in the three or more layered design would be expensive.

After selecting the two sided board, one of the two conduction planes had to be used as a ground plane to match the 50 ohm microstrip between relays. This caused another problem, which was how to deliver power traces to the individual relays. Since the ground plane had to remain intact under the TDR carrying traces, it could not be intersected by vias carrying power to the relays. Hence, these traces had to be worked between the TDR microstrip on top of the board.

Up to this point, the relay has been considered generic. The board layout required the type of relay to be specified. Many different types of relays could have been chosen for the design. However, the Aromat RK1E-L2-12V (Appendix C) exhibited a large (1.2 GHz) bandwidth in a small (0.45 x 0.8 x 0.375 inch) rectangular pack. In addition, it was a mechanical latching, solenoid switch. Therefore, the total power consumption to maintain a given state was zero which was necessary for battery operation.

With a two sided board, one plane used as ground and using Aromat relays, a circular topology was recognized as being the most efficient. In fact, no other array was found that could meet all the

conditions. The layout allowed short microstrip lines to all relays, lowering insertion losses. Total length of the TDR microstrip was 3.82 inches for each of the thirty-two paths. Figures 12 and 13 show the final configuration for the relays, power traces, and ground plane.

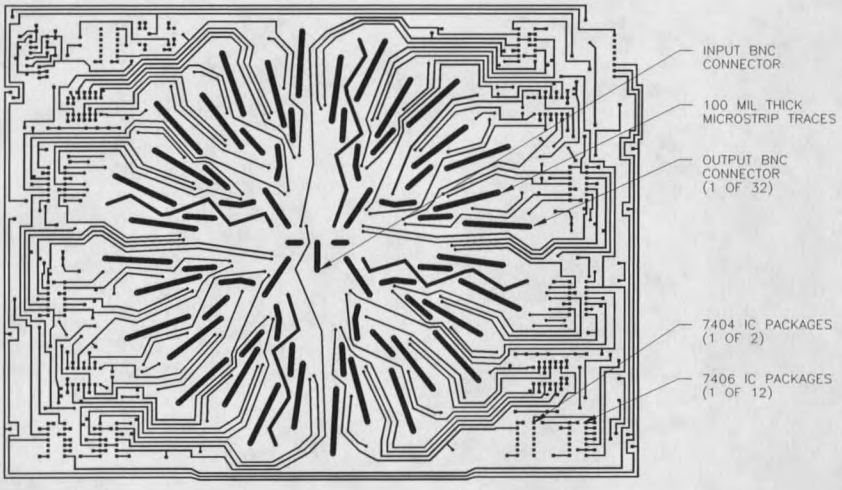


Figure 12. The Circuit Board Layout for the TDR Conduction Plane

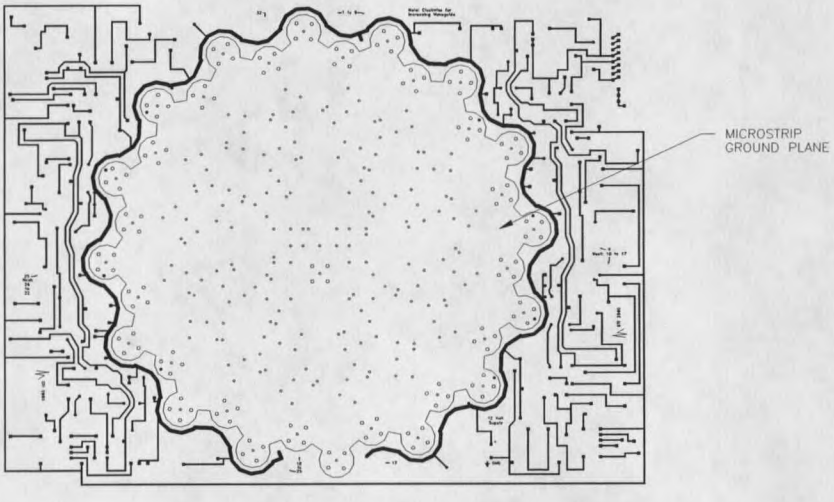


Figure 13. The Circuit Board Layout for the Ground Plane

This arrangement distributed the TDR signal from the input BNC connector in the center of the board, to one of the 32 output BNC connectors. The configuration is spacially compact, the physical size of the circuit board being 8.5 by 12 inches (about the size of a sheet of 8.5 by 11 paper). Relay power traces were delivered directly to the 740X series TTL controlling logic located on the outer perimeters of the circuit board.

Microstrip matching the relays to the BNC connectors was difficult. The board thickness and its dielectric constant had tolerances that contributed to the mismatch. The goal was to keep the multiplexer reflections lower than ten percent, even with variations in board and relay properties.

Four Teflon circuit board samples were donated by Tektronix for this project. Teflon ("Duroid" a registered trademark of Rogers Corp.) board had an extremely low loss tangent and strict dielectric and thickness tolerances. Unfortunately, there were several limitations on the Duroid board in this application. First, the Teflon board demonstrated flexing and deforming when subjected to light forces. The application of normal connecting forces to the BNCs would flex the board and produce loose solder joints, which was definitely unacceptable for this project. Two of these samples were 10 mils thick and deformed too easily.

In addition, there was a restriction dictating that the microstrip trace width could not exceed 100 mils. This constraint arose from the physical size of the BNC connectors. If the trace width exceeded 100 mils, then the TDR trace could intersect with the BNC grounding legs. As will be shown later, the trace width for a fifty ohm match was inversely proportional to the circuit board dielectric constant and directly proportional to the thickness of the board. The thicker Teflon boards had to be discarded since their 50 ohm microstrip matched width was one hundred thirty mils, more than 30 mils over the assumed maximum width.

The commonly used FR-4 board obtained from Merix resisted the deformation from normal forces associated with connecting the probes to the BNCs. A second advantage of the fiberglass composite was its relative availability compared to Teflon circuit boards. This meant lower cost from the board manufacturers. Also, the etching and plating processes for FR-4 were well known. In light of these details, FR-4 composite was selected for this multiplexer design.

Although FR-4 diminished the total board cost and stiffened the support to BNC connectors, there were several disadvantages in its use for this design. First, the board exhibited significant high frequency power attenuation, due to its large dielectric loss tangent. This lowered the resolution of the TDR instrument for small amplitude measurements. With multiple reflections, the loss accumulated and skewed the Giese and Tiemann (GT) calculation as discussed later in experimental results. The author had to derive an equation similar to GT but accounting for multiplexer reflections and loss. Another problem with FR-4 circuit boards was the variation in thickness and dielectric constant when it came from different manufacturers. In general, the nominal thickness will be 68 mils, with a +/- 10 mil tolerance. Likewise, the dielectric value for FR-4 deviates from an average of 6.2 (ε_R) to 5.0 and 7.0.

The relaxed dimensions and dielectric constant specifications permitted the reflection coefficient to depart from a perfect match with zero reflection. The microstrip was designed around nominal parameters, with the hope that statistically caused offsets would be small and possibly even cancel each other.

Microstrip Matching Technique for Constructing the Circuit Board

A useful set of equations for calculating the characteristic impedance for microstrip were provided by Gonzalez [6]. Figure 14 shows the variables W (microstrip width) and h (dielectric thickness) for the circuit board.

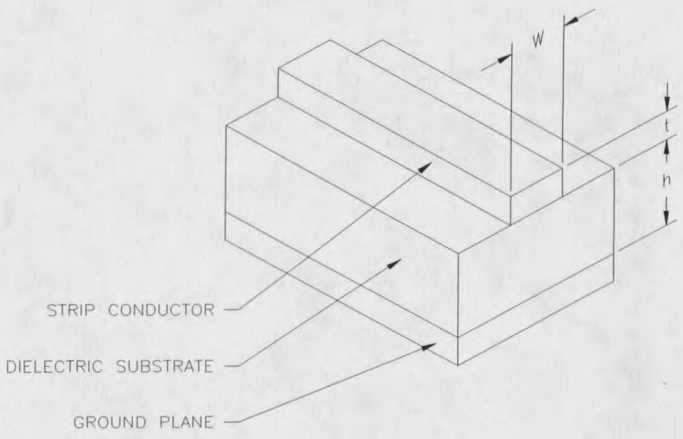


Figure 14. Physical Dimensions of the Microstrip Match

For $W/h \leq 1$:

$$Z_o = \frac{60}{\sqrt{\varepsilon_{ff}}} \ln\left(8\frac{h}{W} + 0.25\frac{W}{h}\right) \tag{25}$$

where

$$\varepsilon_{ff} = \frac{\varepsilon_R + 1}{2} + \frac{\varepsilon_R - 1}{2}\left[\left(1 + 12\frac{h}{W}\right)^{-\frac{1}{2}} + 0.04\left(1 - \frac{W}{h}\right)^2\right]. \tag{26}$$

For $W/h \geq 1$:

$$Z_o = \frac{120\pi / \sqrt{\varepsilon_{ff}}}{W/h + 1.393 + 0.667 \ln(W/h + 1.444)} \tag{27}$$

where

$$\varepsilon_{ff} = \frac{\varepsilon_R + 1}{2} + \frac{\varepsilon_R - 1}{2}\left(1 + 12\frac{h}{W}\right)^{-\frac{1}{2}}. \qquad (28)$$

Merix circuit board division claimed that their FR-4 board had a dielectric constant of 6.2 (ε_R). Based on this information, using equations (27) and (28), and assuming a board thickness of 68 mils with fifty ohm characteristic impedance, the microstrip width of 98 mils was obtained. Merix asked for the CAD drawings to be laid out with TDR widths of 100 mils to account for undercutting in etching the board. These calculations were compared with PUFF, a RF matching program. The output from PUFF agreed with the 98 mil width computed from equations (27) and (28).

The design consisted of a circular array of relays that distributed the TDR signal from one input to one of thirty-two BNC connectors at the output. FR-4 was chosen as the circuit board material, because of its competitive cost and structural strength. Finally, a microstrip width of 98 mils allowed the circuit board to be matched to the TDR and its probes.

Overall Design and Hardware Configuration

The principles of hardware control were briefly explained in the previous section. It was noted that the power traces to the relays had to be worked between the TDR microstrip on the top of the board. These power traces were controlled by NTE 7404 and 7406 IC packages. Hardware and software configuration for the multiplexer will now be analyzed.

Complexity of the TDR multiplexer originates from the layout, not from the electronics circuitry. Figure 15 below shows a schematic for a single relay.

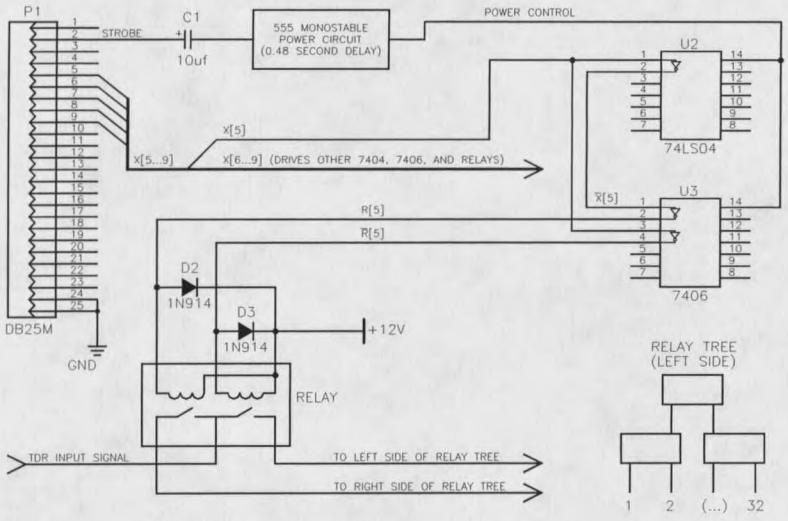


Figure 15. Schematic of a Single Relay

The control signal originated from parallel port of the PC. It was then input to the 7406 inverting buffer IC, and the output applied to the reset pin of the Aromat relay. The 7406 performed two functions. It reduced the fan out load on the PC parallel port, and the 7406 had an open collector output which could be switched to ground. Each 7406 inverter could sink 40 mA, while each relay only required 32 mA of latching current.

The Aromat relay was fabricated such that there were two coils for setting and resetting the mechanical latch. The relay function can be compared to a flip-flop circuit. Once latched in a reset state, only the noninverting signal can set the switch back to its original state. To construct the noninverting signal from the parallel port, it was passed through a 7404 inverting buffer. Then, the output of the 7404 connected to the input of the 7406 inverter. Finally, the 7406 output linked into the set pin of the relay.

After switching the latch position, a 555 timer circuit disconnected voltage supplies from the 7404 and 7406 ICs, eliminating continuous current pull from the controlling logic and relay solenoids.

During switching, the current measured as much as 1.25 amps. If there wasn't a timer to shut off this power, the device would be useless in the field. Even a heavy car battery would be discharged in about an hour. The timer circuit successfully shut down the power with only a residual 10 mA continuously flowing in the 555 timer circuit.

To turn the power back on, the computer must send a strobe on the trigger line of the parallel port. This causes the 555 monostable circuit to return five volt power to 7404 and 7406 ICs. The computer waits for the chips to stabilize (0.2 seconds) and sends the word to change the selected TDR path. Within 0.28 seconds, the monostable timer turns the power supply back off. The car battery will thus have a lifetime of 2-4 weeks (no charging) with the timer employed.

The hardware schematics were provided in Appendix D for troubleshooting circuitry. It was appropriate to discuss service issues for the device. In order of expected problems they were: the strobe electrolytic capacitor had a two year shelf life, the relays have a minimum expected life of five million cycles or one catastrophic 100G impulse drop, and last, any of the various ICs or the transistor switch may fail. Also, wearing of the BNC connectors should be mentioned at this point, unfortunately none of the BNC connectors can be replaced. This will be explained later when discussing experimental results.

Separating the electrolytic capacitor failure from other problems should be straight forward. A bad capacitor will not let the user change any channels; whereas a bad relay, solder connection, or 740X logic chip would allow partial control over some of the channels.

Several things should be noted before repairing a faulty circuit. First of all, when replacing the electrolytic capacitor, the polarity has to be observed before removing the old part and making the replacement. If an electrolytic capacitor is biased improperly, its life will be foreshortened and the component won't function properly in the circuit. Removing relays and transistor switches is fairly straight forward. But care must be used when removing them because the soldering iron can melt neighboring relays and destroy their hermetic seal. Finally, bad 7404 and 7406 chips are easily replaced

from their sockets by lightly prying upward on each side of the IC (once again one must observe and record the direction of the part before removing and replacing).

Software became relevant since the PC acted as the brains for controlling the relay circuitry. The design utilized both C and Basic programming languages. A C program was written to allow TSR (terminate stay resident) interaction with other programs as well as a DOS command line entry. The resident program statement was added to the autoexec.bat file consisting of the statement "W m". Then, activation of a TSR switching menu over running programs was accomplished by pressing the two hot keys <ctrl> and <alt> simultaneously. The same program was used to switch the multiplexer channels from the DOS command line prompt. For example, by typing "C:\ W 2" the program will switch to the second TDR channel. The source code W.C is shown in Appendix E for inspection. By modifying the TSR program (WWIN.C) and giving it an icon representation, the program was ported to a Microsoft Windows environment.

Since tests were already designed around a one to four relay multiplexer (Designed by Cambell Scientific in the early 80's and acquired by Tektronix at a later date), a Basic subroutine was constructed which passed identical parameters for control of the 32 channels. This allowed existing programs to be used without modification.

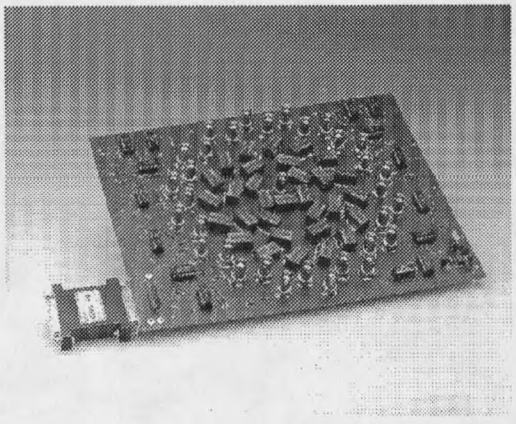


Fig 16. Picture of the Final Design

To finish this discussion of the TDR multiplexer design, a picture of the final product is shown on the previous page in Figure 16 and a sales brochure is displayed in Appendix F. Some of the device features are summarized. Hardware items such as the 50 ohm microstrip match, the circular array of relays, and the power shutdown timer help the device meet the preliminary specifications. In addition, several software packages allow a TSR mode, a DOS command line mode, a Windows icon program, and a BASIC subroutine for switching. With both the hardware and software completed, the next section will proceed with experimental procedures and methods employed in testing the design against the outlined specifications.

CHAPTER 3

TESTING THE PERFORMANCE OF THE MULTIPLEXER

Since, the multiplexer can introduce aberrations in both amplitude and time measurements, several experiments were performed in evaluating the device. The major areas of testing included bandwidth, insertion loss, and conductivity measurements. Experimental procedures will now be explained in more detail.

Utilization of the Network Analyzer

The bandwidth of the TDR switcher was easily obtained from a scalar network analyzer. The scalar network analyzer displayed all four S parameters for a two port network, with the phase neglected. For this design, the half power bandwidth was significant because it was directly related to the system rise time (where the rise time is the limiting parameter for resolving two closely spaced discontinuities in impedance).

The HP 8756A scalar network analyzer with a 8350B sweep oscillator was selected to examine the frequency response. The analyzer was calibrated with a short, an open, and a 50 ohm matched termination with channel one off. Channel two was set to measure the forward power (S_{21}^2) in decibels. The 8350 sweep range was set from 10 MHz to 2 GHz.

In order to confirm measurements, three known attenuators were connected between the 8756A and the 8530 sweep oscillator. The first attenuator had an identifying label of 2x50, which means a

reduction of signal by one half in a fifty ohm system. The marking implied that the voltage should be down 6.02 dB as computed in (29)

$$20\log\left(\frac{1}{2}\right) = -6.02\ \text{dB}. \tag{29}$$

The network analyzer displayed a 6 dB loss through the attenuator, up to a frequency of about 1.8 GHz. After reaching this critical frequency the power loss began to fluctuate (Figure 17) and no accurate measurement could be completed. The fluctuations were a result of TM mode propagation inside the coaxial cable and these energies escaped detection by the diode sampler inside the network analyzer. A similar effect was observed when making measurements with the TDR switch, but the critical frequency decreased to 1.5 GHz since the physical dimensions of the switch were larger than coaxial cable.

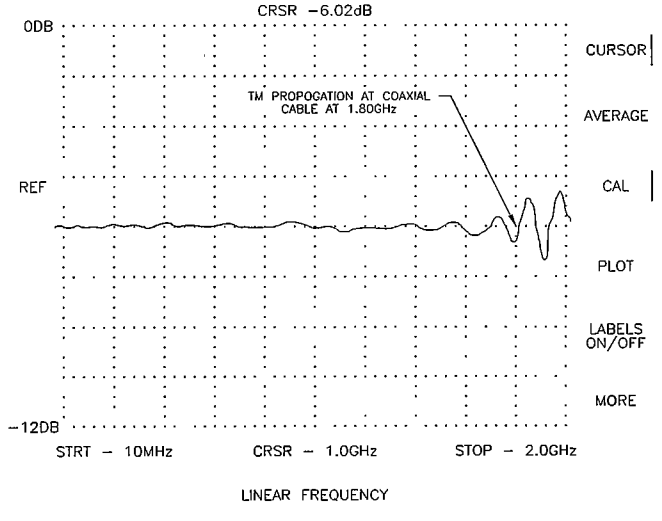


Figure 17. Network Analyzer Display for 6 dB Attenuator

With the fluctuations understood, the 2x50 attenuator was removed. It was replaced by a 2.5x50 and then a 5x50 attenuator, the network analyzer correctly displayed -7.958 dB and -13.9 dB; respectively.

Then the TDR multiplexer was connected between the network analyzer and sweep generator, the half power bandwidth (i.e. the frequency where the power is down by 3 dB) was recorded. This bandwidth measurement was used to determine the rise time measurements made in the next section.

Utilization of the TDR

Most of the multiplexer's tests incorporated the TDR instrument. The TDR was used to determine the system rise time, the multiplexer reflections, and the insertion loss.

The experimental procedure for the rise time was as described below. The fast 45 ps pulser was connected into the multiplexer, and its output was taken directly into a digitizing Tektronix 11802 oscilloscope. By auto triggering on the edge of the pulsed square wave, the rising edge of the waveform was observed. The oscilloscope cursor allowed acquisition at 10% and 90% of peak value. The difference in these times was the rise time. Without the multiplexer, one should expect the rise time to measure 45 ps, the same as the pulsed step response. However, since the multiplexer reduces the bandwidth, this period increased. Table 2 below restates the steps for measuring rise time.

1. Connect a fast 45 ps pulser into the multiplexer.
2. Take the multiplexer output into the digitizing oscilloscope.
3. Acquire on leading edge of the pulsed square wave.
4. Record the times at 10% and 90% of the "on" voltage.
Their difference is system rise time.

Table 2. Procedure for Measuring the Rise Time

It is obvious that the multiplexer reflections were easily measured with a TDR instrument. In fact, the whole purpose of TDR is to measure reflections. What may not be so obvious is that TDR can diagnose loose solder connections, bad relays, mismatch and inconsistency between each of the thirty two channels. Therefore, the device reflections were the most important measurements.

When a TDR instrument was connected to the multiplexer, a length of coaxial cable extended between the two devices. It was the intent of the reflection test to distinguish the location of input and output BNC connectors, relative to the entire time axis. First, the coaxial cable was disconnected from the (Figure 18) multiplexer, so the TDR showed an open at the position in time where the cable was left open. The location was then marked with a cursor.

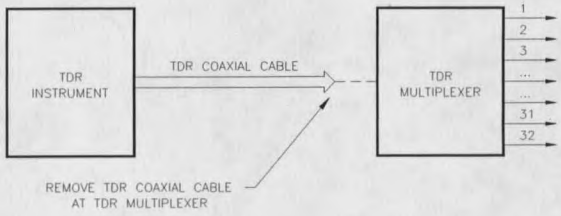


Fig 18. Disconnecting the Coaxial Cable from the TDR Multiplexer

Then the coaxial cable was reconnected to the multiplexer (the multiplexer termination is hard to find (Figure 19), even with the output BNC left open).

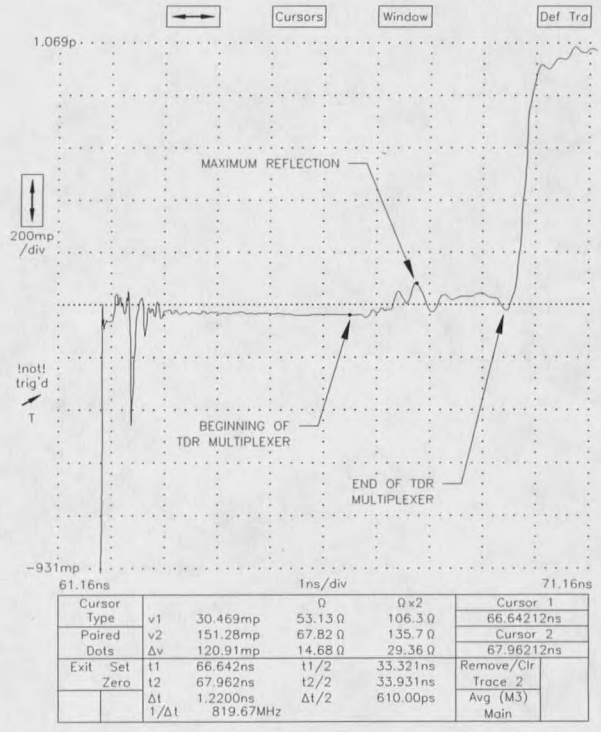


Fig 19. TDR Reflection for the Multiplexer with an Open at the BNC Output

To contrast the output BNC connector, a 50 ohm coaxial cable was added to the end of the TDR switch (the multiplexer reflections remain unchanged). The second cursor was used to mark the location of the multiplexer and cable interface. The device reflections were positioned between the two cursors (Figure 20), then the cursor was used to measure the maximum deviation in reflection.

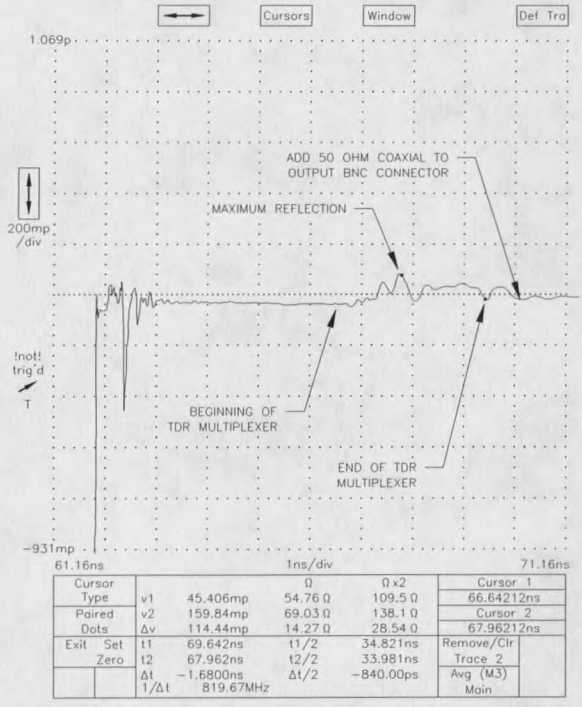


Figure 20. TDR Reflection for the Multiplexer with 50 ohm Cable at BNC Output

Notice that the measurement yielded a maximum reflection of 15 %, or a 17 ohm variation from the assumed 50 ohm coaxial cable (Fig 20). Although this appeared large, a standard BNC connector showed the same magnitude of reflection. Nevertheless, one of the design specifications had been exceeded by five percent and, when coupled with insertion loss, this reflection skewed the conductivity analysis. For convenience, Table 3 is presented to recap the testing procedure for multiplexer reflections.

1. Find the input BNC connector on the TDR screen.
2. Find the output BNC connector on the TDR screen.
3. Measure the Maximum deviation, between the input and output connectors.

Table 3. Procedure for Measuring the Maximum Reflection

At this point, it would be wise to mention how to employ TDR in measuring the multiplexer insertion loss. Suppose that the multiplexer is connected to the TDR and the output is shorted to ground. The pulse generator then steps positive by V_o, but since the forward propagating wave hasn't reached the short at the end of the multiplexer, the oscilloscope will read one half of V_o as shown in equation (30). The forward moving wave travels through the multiplexer and experiences a gain "f_{MUX}", although the terminology is deceptive because "f_{MUX}" has a value less than one and thus accounts for losses caused by conduction

$$V_{I@OSC} = \frac{V_o}{2} . \tag{30}$$

After reflecting from the short, the sign will change. The returned signal will have a value given by equation (31). Remember that the signal is attenuated in both directions of travel, so the loss f_{MUX} is squared.

$$V_{REF} = -\frac{V_o}{2} \cdot f_{MUX}^2 \tag{31}$$

At some time $t_o + 2 \cdot \Delta T$, both the initial and reflected voltages will superimpose, displaying the final voltage at the oscilloscope

$$V_{F@OSC} = V_{REF} + V_{I@OSC} = V_{I@OSC} \cdot (1 - f_{MUX}^2) . \tag{32}$$

Using algebra, to solve for the insertion power loss f^2_{MUX} gives the result

$$f^2_{MUX} = 1 - \frac{V_{F@OSC}}{V_{I@OSC}}. \tag{33}$$

1. Connect the multiplexer input to the TDR. 2. Short the multiplexer output. 3. Measure the initial and final voltages, then compute the insertion loss coefficient from equation (33).

Table 4. Procedure for Measuring the Insertion Loss

Correlating TDR and Network Analyzer Results

As previously mentioned, a reduction in system bandwidth results in longer rise times. The theory for explaining this relation will use the Laplace method for solving a representative one pole system. Then the method will be altered to determine the resolution time. Some comments regarding the impact of rise time on conductivity and water content measurements are stated at the end of this section.

A multiplexer cannot really be described as a one pole system. It is, however, a reasonable first order approximation. The focus of this derivation is not to obtain a 100% accurate expression, but to derive a rational approximate relation of bandwidth to rise time. In a similar fashion, the time resolution will be associated with the rise time of the instrument.

Assuming that the multiplexer has only one pole, the transfer function follows as equation (34). The variable (K) denotes the multiplexer gain (less than one), (s) is the sum of real and complex

frequencies, and (a) is the pole position on the real axis. Parameter (a) is also the radian 3 dB frequency, or in this case, the half power bandwidth. The 3 dB bandwidth in hertz is given by equation (35)

$$T(s) = \frac{V_o(s)}{Vin(s)} = \frac{a \cdot K}{s + a}. \tag{34}$$

and

$$\beta = f_{3dB} = \frac{a}{2\pi}. \tag{35}$$

Ideally, the pulse generator would have an infinite bandwidth, its signal having unit step impulse function. In the frequency domain, the unit step can be represented as the reciprocal of s

$$Vin(s) = \text{ unit step impulse} = u(s) = \frac{1}{s}. \tag{36}$$

Solving equation (34) for $V_o(s)$ by replacing Vin(s) with the unit step response (equation 36) we have

$$V_o(s) = \frac{1}{s} \cdot \frac{a \cdot K}{s + a}. \tag{37}$$

By partial fraction expansion, the product of equation (37) can be broken into two terms, where each component will then have an inverse Laplace transform given by the standard table (Appendix G)

$$V_o(s) = \frac{K}{s} - \frac{K}{(s+a)}. \tag{38}$$

Equation (38) can be inverse-transformed by inspection to be

$$V_o(t) = K \cdot u(t) - Ke^{-at}. \tag{39}$$

Recall that the function u(t) is the ideal unit pulse, applied to the input of the multiplexer. The actual signal exiting the TDR switch is an exponential function, depicted by Figure 21. The waveform approaches the input unit step function when the bandwidth is large and K is unity, the latter implying low insertion loss.

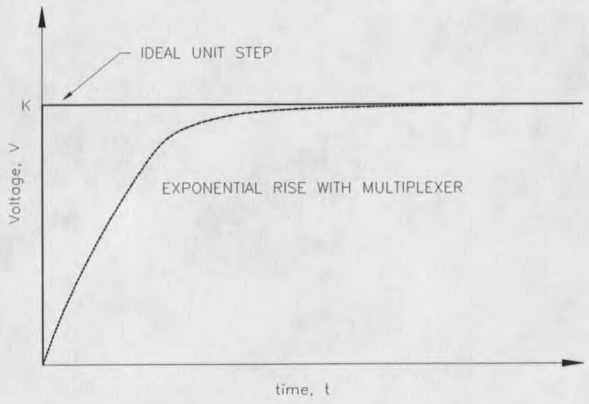


Figure 21. The Ideal and Actual Signals Exiting the Multiplexer

If the waveform approaches a unit step input with increasing bandwidth, then the rise time should decrease because the exponential slope is increasing. At the furthest extreme, an infinite bandwidth would permit a unit step output with a ramp of infinity. The initial rate of increase is found by evaluating the derivative of equation (39) at zero time thus

$$\frac{d[V_o(0)]}{dt} = K \cdot a. \tag{40}$$

This effectively demonstrates that the initial slope is directly proportional to the 3 dB frequency in radians.

Continuing with the pursuit of rise time, it was previously defined as the period required for the pulse to increase from 10% to 90% of its final value. First, the final voltage is (by letting the time approach infinity in equation (39))

$$V_F = \lim_{t\to\infty}[V_o(t)] = \lim_{t\to\infty}[K \cdot u(t) - Ke^{-at}] = K. \tag{41}$$

When the voltages at 10% and 90% of the final voltage are computed, they are 0.1K and 0.9K; respectively. By substituting these voltages into $V_o(t)$ of equation (39), the two times t_{10} and t_{90} are obtained

$$t_{10} = -\frac{\ln(0.9)}{a} \tag{42}$$

and

$$t_{90} = -\frac{\ln(0.1)}{a}. \tag{43}$$

The rise time is calculated by subtracting t_{10} from t_{90} thus

$$t_{RISE} = t_{90} - t_{10} = \frac{\ln(0.9) - \ln(0.1)}{a} = \frac{\ln(9)}{a} = \frac{2.19}{a} = \frac{2.19}{2\pi f_{3dB}} = \frac{0.35}{f_{3dB}}. \quad (44)$$

This time can also be expressed in terms of the multiplexer bandwidth, since f_{3dB} is the same as the half power bandwidth β. That is

$$t_{RISE} = \frac{0.35}{\beta}. \quad (45)$$

Now that the major objective has been met (determining rise time from bandwidth) a quick comment can clarify what has been learned. The most obvious point is that the rise time is inversely proportional to the multiplexer bandwidth. Any reduction in the system bandwidth causes the rise time to increase.

The reader may be questioning why so much effort has been devoted into studying rise time in the first place. The answer is very simple, if rise time increases, the overall TDR time resolution decreases. According to James A Strickland the time resolution can be defined as, "the ability of a TDR system to distinguish between two point discontinuities that are located very close together" [1]. The equivalent mathematical description involves the minimum time spacing of two equal point discontinuities, which give rise to a 50% valley between the two displayed reflections. Thus, a poorly resolved TDR system gives the appearance of three discontinuities, instead of two.

A model will illustrate time resolution as a function of the rise time. The ideal case (Fig 22) of a perfect unit impulse response would allow the discrimination of discontinuities, even at incrementally small distances. However, when the rise time is finite (Fig 23), there is a critical spacing in time between discontinuities that results in the 50% valley.

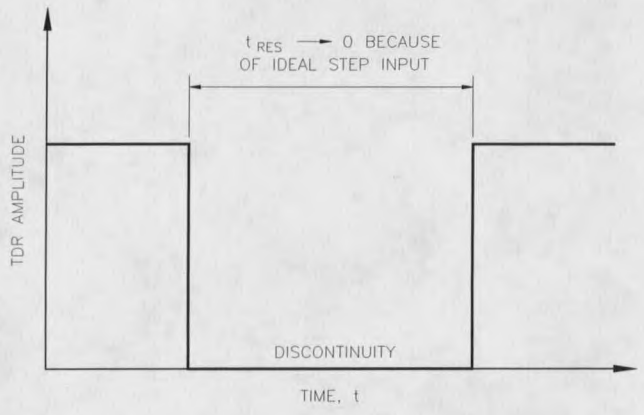


Figure 22. Ideal TDR Response for Two Discontinuities

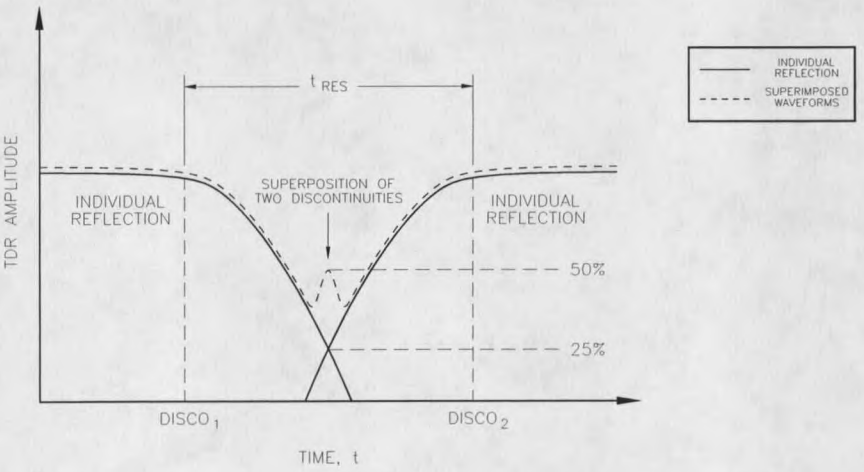


Figure 23. Actual TDR Response for Two Discontinuities

A theoretical relation of rise time with time resolution can be deduced by representing the multiplexer as a one pole filter response (equation (34)). Because the reflected waves are in the image of the waveform exiting the multiplexer, the minimum spacing will (Figure 23) be given by the time required for the superposition of both rise time waveforms to reach a 50% value. Therefore, the resolution time is double the 25% to 90% rise time (refer to equations (46) and (47)).

$$t_{25\%-90\%} = \frac{\ln(0.75) - \ln(0.1)}{a} = \frac{\ln(7.5)}{a} = \frac{\ln(7.5)}{2\pi f_{3dB}} = \frac{0.32}{f_{3dB}} = \frac{0.32}{\beta} \tag{46}$$

$$t_{RES} = 2 \cdot t_{25\%-90\%} = \frac{0.64}{\beta}. \tag{47}$$

As a final remark, notice that the resolution time will increase for smaller multiplexer bandwidths which is definitely undesirable. It should be pointed out, that the resolution issue of TDR in the time dimension may very well be an exaggeration since the GT analysis requires only the initial and final voltages. This implies that conductivity measurements are nearly independent of the pulse response. On the other hand, the dielectric constant (water content) measurement is dependent on an accurate measurement of elapsed time t_{AB} between multiple reflections in the probe. It seems reasonable that increasing resolution time could have adverse effects on the dielectric measurements. The significance of these points will be considered again, when explaining experimental results, and at the end of this paper where an alternate solid state design is proposed.

Experimental Setup for Measuring the Salinity of NaCl Solutions

Various properties of the multiplexer have been tested. Experimental procedures for the bandwidth, rise time, reflection, and insertion losses were furnished in detail. The experimental procedure for measuring conductivity of saline solutions can now be introduced.

Measuring conductivity answered the most important question, "does the multiplexer function properly with the TDR instrument?". The equipment for measuring conductivity (Figure 24) consisted of a Tektronix 11802 SD-24 TDR, a 50 ohm air core coaxial probe (Figure 8), and a 60Hz AC resistance bridge. Figure 25 shows the AC resistance bridge in detail, where the resistance R_{POT} was measured with a Fluke Data Precision 3500 ohm meter (four decimal places of accuracy).

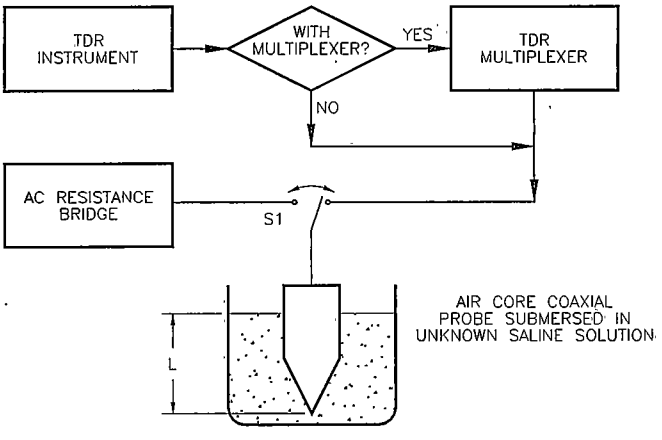


Figure 24. Experimental Setup for Measuring Conductivity

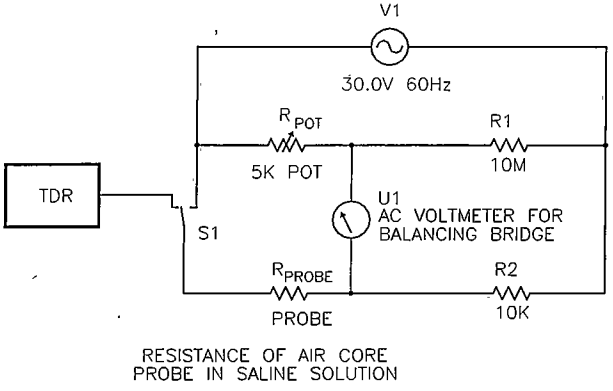


Figure 25. Conductivity Measurement with the AC Resistance Bridge

First, a water solution of sodium chloride was mixed until the salt crystals were totally dissolved. Then, two tests were performed in measuring conductivity through the coaxial probe, the Giese and Tiemann (GT) analysis and a resistance measurement. The AC resistance bridge provided a reference to compare the GT conductivity measurements (both with and without the multiplexer present). Switch S1 toggled the conductivity tests between the TDR and AC resistance bridge measurement devices.

Since theory for conductivity using TDR was thoroughly analyzed, the theory behind the AC resistance bridge was the primary objective of this section. The experimental procedure involved switching in the probe (S1 in Figures 24 and 25) to the bridge, varying the potentiometer resistance until the voltmeter measured zero volts, removing R_{POT}, and measuring its resistance with the precision ohm meter.

In order for the Wheatstone bridge to balance (zeroed voltmeter), the ratio of resistances of any two adjacent arms must equal the ratio of resistances of the remaining arms [7], so

$$R_{PROBE} = \frac{R_{POT} \cdot R_2}{R_1}. \tag{48}$$

The bridge reduces the ohm meter error by the ratio of R_2 to R_1 (48), thus increasing the accuracy of the resistance measurement to seven decimal places.

Now that the method for calculating the probe resistance has been explained, equation (49) was used to compute the conductivity (refer to Appendix H for the equation derivation),

$$\sigma_{BRIDGE} = \frac{\ln\left(\frac{b}{a}\right)}{2\pi \cdot R_{PROBE} \cdot L} \tag{49}$$

where L was the depth of the coaxial probe into the saline solution, R_{PROBE} was obtained from equation (48), b and a are the outer and inner radius of the coaxial probe respectively.

As a final note, the resistance measurement provided a reference in which the GT results could be compared. Several different concentrations of salt solutions were mixed and tested for their conductivity, using the method outlined in this procedure. The experimental results will be now be presented.

CHAPTER 4

EXPERIMENTAL RESULTS

Results of the Bandwidth and Rise Time Test

The network analyzer was used to determine the system bandwidth, with the multiplexer connected between the TDR and probe. The frequency variation of the multiplexer forward power (S_{21}^2) in decibels was as depicted on the network analyzer screen was displayed in Figure 26. At low frequencies the power was down zero decibels. As the frequency increased, through power decreased by 0.4 decibels every 100 MHz. The half power bandwidth (where the signal was down 3 decibels) was at 798 MHz, as shown by the display cursor in Figure 26. Notice that the waveform fluctuated with frequencies beyond 1.5 GHz, as previously explained.

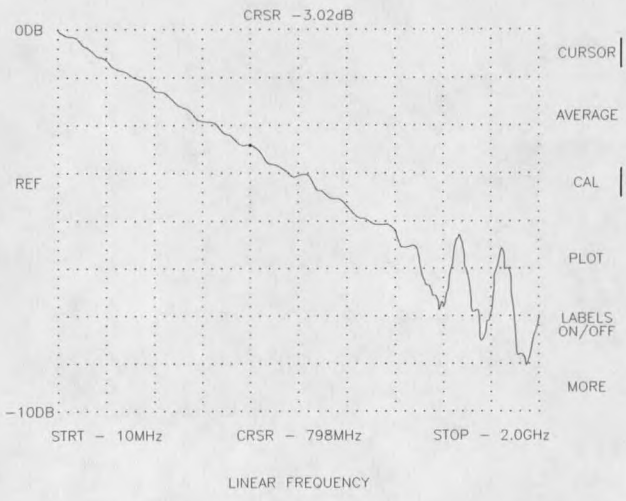


Figure 26. Network Analyzer Display for the Multiplexer

Equation (45) was derived to relate bandwidth to rise time, it is restated for convenience

$$t_{RISE} = \frac{0.35}{\beta}. \tag{50}$$

If the bandwidth was 798 MHz, then from equation (50) the 10% to 90% rise time calculated to be 438 ps. This result can be compared to the actual TDR rise time measurement. Figure 27 shows the TDR pulse after passing through the multiplexer, observe that the measured rise time was 361 ps. Therefore, the first order approximation of representing the multiplexer as a one pole system, had proven to be a fairly accurate model.

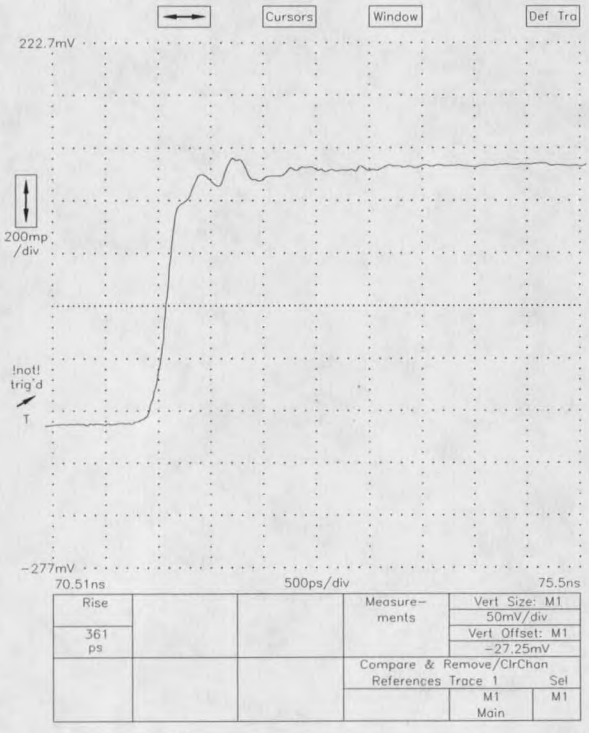


Figure 27. Multiplexer Rise Time Measurement

The TDR system (with multiplexer) resolution can now be calculated with the one pole model from equation (47) as

$$t_{RES} = 2 \cdot t_{25\%-90\%} = \frac{0.64}{\beta}. \tag{51}$$

By substituting in a bandwidth of 798 MHz into equation (51), the resolution time of 802 ps is obtained. The equivalent resolution in length is obtained by multiplying the resolution time t_{RES} by the velocity calculated from equation (23). The worst case resolution in length, that is a wave propagating at the speed of light ($3x10^8$m/s), would be 24 cm (9.45 inches). This means that two discontinuities with separation less than 24 cm cannot be resolved. Since, standard probes are usually 12.7 cm (5 inches) long, and the dielectric measurement from equation (24) requires a resolution of twice the probe length (25.4 cm or 10 inches), the dielectric constant is just barely resolved in the worst case scenario with a five inch probe. Most likely, the propagation velocity will be lower than the speed of light, this gives enough margin to make an accurate measurement of the dielectric constant (water content). Also, the probe can be made two inches longer for more distance (time) resolution, without violating the GT conditions of short probe lengths.

The conclusions from experimental procedures and the results of this section, proved that the TDR system with the multiplexer barely had enough resolution for making dielectric measurements. Conductivity measurements, on the other hand, were independent of time resolution since the bandwidth and rise time have no significance in conductivity analysis. Experimental results for the reflection and insertion measurements follow.

Results of the Reflection and Insertion Loss Experiments

The results of the reflection test were previously given. The maximum reflection of 15% was found in the TDR multiplexer, which, when combined with insertion losses, was enough to skew the GT conductivity analysis.

The reflection tests also demonstrated that two of the multiplexer channels were nearly open as shown in Figure 28. The two markers gave the location of the multiplexer relative to the entire time axis.

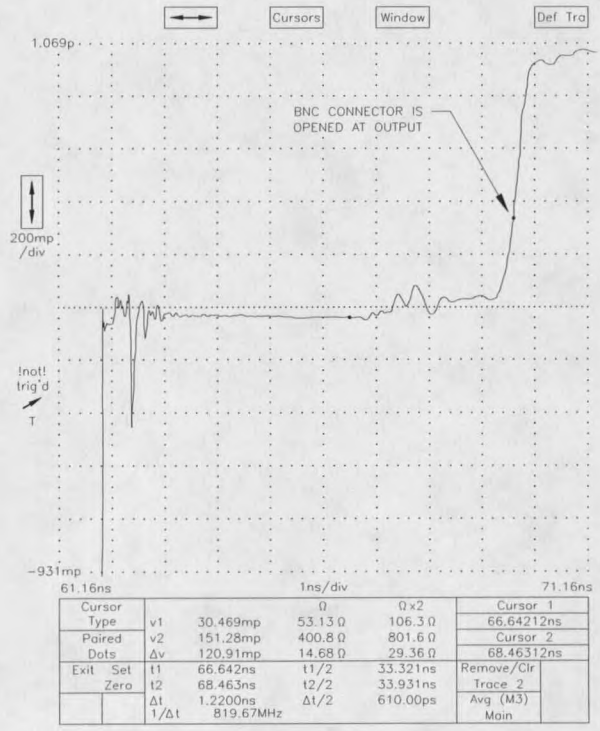


Figure 28. TDR Reflection Showing Loose BNC Connector

In this case, channel twelve showed an effective open at the output BNC connector, even after the addition of a fifty ohm match. The impedance mismatch disappeared when a force was applied to the side of the BNC connector, indicating that the BNC solder contact changed in characteristic impedance with applied lateral forces. To remedy the problem, the BNC connectors were set with the proper lateral force to output a 50 ohm impedance, then they were epoxied down to the board. This could be a problem since the BNC connectors cannot be replaced from the circuit board. With the reflections tests completed, the insertion power measurements were completed.

The experimental procedure for measuring the insertion power loss was quite simple. A short was placed on the output of the tested channel. Then the loss was given by equation (33) (and repeated below)

$$f_{MUX}^2 = 1 - \frac{V_{F@OSC}}{V_{I@OSC}}. \tag{52}$$

This was the insertion power loss for the multiplexer. Figure 29 displays the TDR screen with the multiplexer shorted. The initial voltage $V_{I@OSC}$ equaled 12.453 mV while the final voltage $V_{F@OSC}$ equaled 255.17 mV. Using equation (52), the insertion power loss coefficient f_{MUX}^2 was 0.951. This meant that up to 95 percent of the TDR power made it through the multiplexer, a fairly good power transfer.

The end result of the reflection and insertion power losses will become more obvious later in this chapter, when the author derives a correctional theory for the GT analysis. The results of GT and resistance bridge conductivity tests will now be discussed.

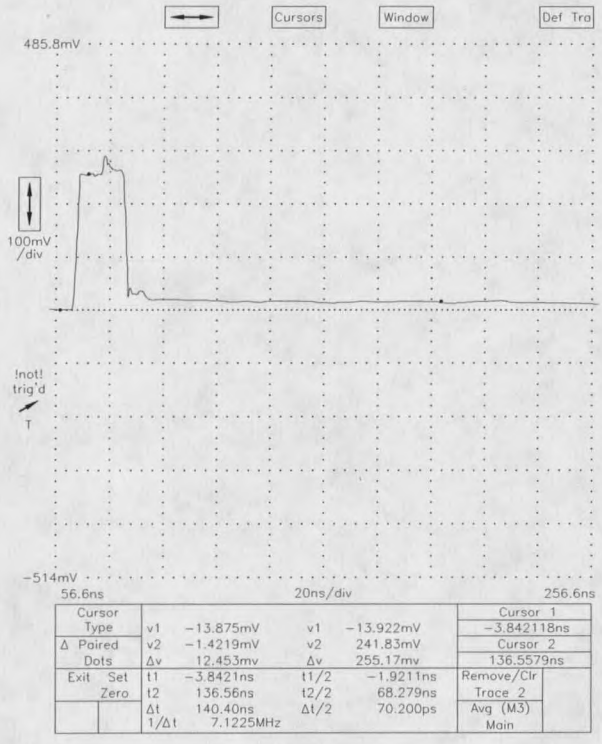


Figure 29. TDR Display for Measuring Multiplexer Insertion Loss

Conductivity Results of the NaCl Salt Solutions

The conductivity of several sodium chloride salt solutions were measured, using TDR as well as a resistance bridge. The results are summarized in Table 5 on the next page.

σ_{BRIDGE} (S/m)	σ_{GT} (S/m)	σ_{GT} (S/m)
(no multiplexer)	(no multiplexer)	(with multiplexer)
0.004	0.007	0.007
0.091	0.103	0.101
0.137	0.156	0.143
0.547	0.635	0.488
0.638	0.712	0.564
0.814	0.786	0.668
1.085	1.095	0.758

Table 5. Experimental Results for the Conductivity Measurements

The GT conductivities are graphed against the bridge conductivity in Figure 30. Bridge conductivity and the GT analysis (without the multiplexer) have a 1:0.9985 linear slope fit, an expected result since both measurements should agree (an ideal slope of 1:1). Unfortunately, the conductivity measurements with the multiplexer were nonlinear and a slope fit could not be performed on its conductivity data. Since the TDR multiplexer interfered with the conductivity measurements, a compensation theory will be derived in the next section.

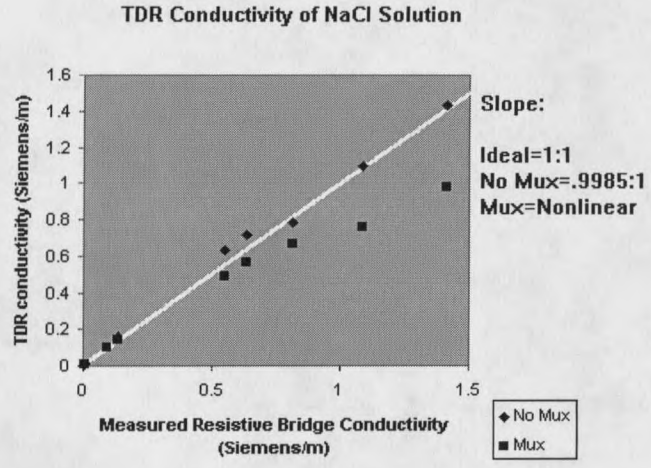


Figure 30. Graph of GT versus Bridge Conductivities

Compensating Theory for the Multiplexer

The TDR switch skews the GT analysis, it creates a second boundary that can reflect and absorb energy. A more complete theory is derived to account for the multiplexer in a TDR system.

Looking at Figure 9 of the GT analysis, the initial pulsed voltage V_o, undergoes several reflections (with losses f) between the coaxial cable and the probe. The final voltage will be the sum of all returned voltages. Thus, a person can think of the voltage ratio V_F to V_o as an equivalent reflection coefficient Γ_{NET}, assuming that the new analysis is only interested in final and initial voltages. This assumption is true for the compensating theory and so the ratio V_F to V_o (Γ_{NET}) is algebraically extracted from equation (22) as

$$\Gamma_{NET} = \frac{V_F}{V_o} = \frac{2}{\left(\frac{\sigma_T L}{\varepsilon_o c} \cdot \frac{Z_u}{Z_o} + 1\right)}. \qquad (53)$$

The equivalent reflection diagram is shown in Figure 31, with the voltage V_o applied from the pulse generator.

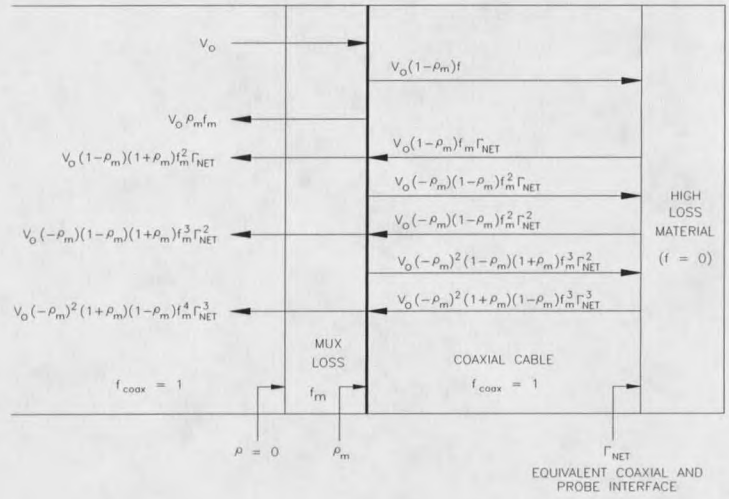


Figure 31. Equivalent Reflection Diagram

The pulse power is reflected several times and absorbed by the multiplexer and the conductivity probe. One can sum the voltage to the left of the multiplexer (Figure 31) and equate this to the new final voltage V_F' detected by the TDR instrument

$$\begin{aligned} V_F' = {} & V_o\rho_m f_m + V_o(1+\rho_m)(1-\rho_m)f_m^2\Gamma_{NET} + \\ & V_o(-\rho_m)(1+\rho_m)(1-\rho_m)f_m^3\Gamma_{NET}^2 + \\ & V_o(-\rho_m)^2(1+\rho_m)(1-\rho_m)f_m^4\Gamma_{NET}^3 + \ldots \end{aligned} \qquad (54)$$

The equation (54) is reduced by factoring

$$V_F' = V_o\rho_m f_m + V_o(1+\rho_m)(1-\rho_m)f_m^2\Gamma_{NET}\cdot \left[1+(-\rho_m)f_m\Gamma_{NET}+(-\rho_m)^2 f_m{}^2\Gamma_{NET}^2+\ldots\right. \tag{55}$$

where the geometric series is defined as

$$\frac{1}{1-\gamma} = 1+\gamma+\gamma^2+\gamma^3+\ldots \tag{56}$$

and

$$\gamma = (-\rho_m)f_m\Gamma_{NET}\cdot \tag{57}$$

Before continuing, notice that coefficient γ will always have a magnitude less than one. Therefore the geometric series of equation (56) doesn't require any other restrictions than already discussed in the GT analysis.

Now, dividing equation (55) by the stepped voltage V_o and using equations (56) and (57) to express the equation in finite form, the final to initial voltages are derived with the multiplexer present

$$\frac{V_F'}{V_o} = \rho_m f_m + (1+\rho_m)(1-\rho_m)f_m^2\Gamma_{NET}\cdot\left[\frac{1}{1+\rho_m f_m\Gamma_{NET}}\right]. \tag{58}$$

Replacing Γ_{NET} in equation (58) by equation (53) one has

$$\frac{V_F'}{V_o} = \rho_m f_m + 2(1+\rho_m)(1-\rho_m)f_m^2 \cdot \left[\frac{1}{\frac{\sigma_T L}{\varepsilon_o c} \cdot \frac{Z_u}{Z_o} + 1 + 2\rho_m f_m} \right]. \quad (59)$$

The compensated GT equation (60) is completed by solving for the conductivity σ_T from equation (59).

$$\sigma comp_T = \frac{\varepsilon_o c}{L} \cdot \frac{Z_o}{Z_\mu} \cdot \left[\frac{2(1+\rho_m)(1-\rho_m)f_m^2}{\left(\frac{V_F'}{V_o} - \rho_m f_m \right)} - (1+2\rho_m f_m) \right] \quad (60)$$

where:

ρ_m = The equivalent single boundary multiplexer reflection

f_m = The multiplexer loss

ε_o = Permittivity of free space= $\frac{1}{36 \cdot \pi} x10^{-12}$ F / m

c = Speed of light in free space = $3x10^8$ m/s

L = Depth of probe into the soil

= (Thin sample) m

Z_o = Probe impedance with air as dielectric

= (usually ≈ 50) ohms

Z_μ = Probe impedance with a dielectric before soil sample

= (usually = Z_o ≈ 50) ohms.

If the multiplexer reflection ρ_m is zero and the multiplexer loss f_m is one (i.e. the multiplexer has no loss), then equation (60) reduces to the familiar GT equation (22). For this design, the values of ρ_m and f_m were determined by solving for the coefficients that gave a best fit curve of equation (60) to the experimental data in Table X. Based on the curve fit analysis (Appendix I), the best fit value of ρ_m was 2.42%, and the best fit value of f_m was 0.9913

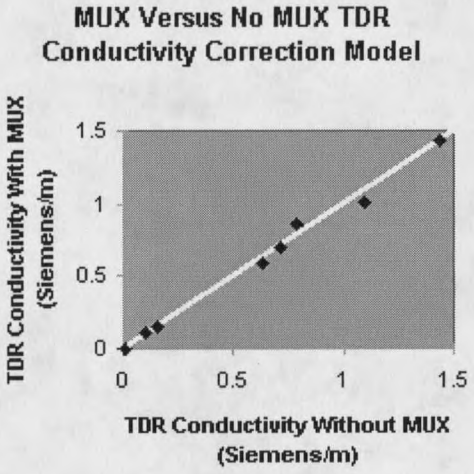


Figure 32. Graph of Corrected Conductivities

CHAPTER 5

CONCLUSIONS

This paper considered the design, implementation, and experimental verification of a one to thirty-two TDR switch. Some of the important design parameters were the reflections, the insertion losses, the accuracy of conductivity measurements, and the resolution for measuring dielectric content of soil samples (water content). All of these measured results met or exceeded the specifications outlined in the preliminary design, except for the TDR system resolution which was marginally adequate for making measurements.

The maximum reflection for the TDR multiplexer was 15%, five percent more than the specification. The extra reflection was mismatch caused by tolerances, such as the variations in the circuit board dielectric constant and thickness properties. The test also revealed that two of the output BNC connectors had loose contacts causing an open at the output when applying a lateral force. The solution was to epoxy the BNC connectors down to the board.

Insertion power losses were less than 5%. However, this was deceptive, because most of the loss occurred at higher frequency. System three decibel bandwidth decreased to 798 MHz, after adding the multiplexer, or equivalently the rise time increased from 45 psec (the rise time of the pulse generator) to 802 ps with the TDR multiplexer. The time resolution increased as a consequence of the high frequency components being attenuated, thus reducing the ability to make accurate dielectric measurements. An even larger problem existed with the large reflections and insertion loss, in that the Giese and Tiemann (GT) analysis wasn't adequate to determine conductivity.

Conductivity measurements using GT were incorrect because the multiplexer created an additional reflection boundary between the soil probe and the TDR instrument. In order to get valid conductivity results these reflections and losses had to be taken into account by another derivation, the compensation theory.

Utilizing the modified GT equation (60) resulted in more accurate measurements of conductivity with the TDR multiplexer (1:0.9995 linear fit between the TDR with and without the switch). Therefore, the multiplexer as designed was functional in both conductivity and dielectric constant measurements.

The author feels that the existing design could be improved substantially by mounting the circuit board into a unexposed mechanical box. Coaxial cable would connect the circuit board inside the box to external BNC connectors fixed to the front panel. This would alleviate the fatigue problems associated with connecting the BNC connectors to the board. Thus, Teflon could be used as the circuit board material since flexing would no longer be an issue.

An alternative solid state device could also be built to multiplex the TDR signal. The largest advantage of solid state electronics versus an electro-mechanical design is that it would allow channels to be sampled continuously, whereas relays have long (in seconds) switching times, thus preventing such sampling. Applications of TDR in measuring several points in the soil surrounding a nuclear containment facility, for instance, need a quick sampling rate to continuously monitor all channels. An improved design would consist of two broad band amplifiers in the forward and return paths of a directional coupler with the channel being turned on and off by controlling power to the amplifiers. The solid state switch has one major disadvantage, it will have a smaller bandwidth than an electro-mechanical device.

In summary, this thesis presented a functional design for switching TDR signals. Although the device demonstrated fairly large insertion losses and reflections, the overall design was a success. Several

issues were investigated and the analysis of the limiting mechanisms serves as a tool for the layout and implementation of successive designs.

REFERENCES CITED

1. Strickland J.A., Time Domain Reflectometry Measurements, Tektronix, Inc., Beaverton, Oregon, 1970

2. Giese K., and Tiemann R., "Determination of the Complex Permittivity From Thin sample Time Domain Reflectivity", Improved analysis of step response waveform, Vol. 7, 1975, pp. 45-59.

3. Dalton F.N., and Genuchten M., "The Time Domain Reflectometry Method for Measuring Soil and Water Content Salinity", Geoderma, 38, 1986, pp. 237-250

4. Topp. G.C., Davis J.L., and Annan A.P., "Electromagnetic Determination of soil water Content: Measurement in Coaxial Transmission Lines, Water Resources Research, Vol 16, 1980, pp. 574-582

5. Topp. G.C., Yanuka M., and Zegelen S., "Determination of Electrical Conductivity Using Time Domain Reflectometry: Soil and Water Experiments in Coaxial Lines, Water Resources Research, Vol 24, Num. 7, 1988, pp. 945-952

6. Gonzalez, Guillermo., Microwave Transistor Amplifiers, Prentice Hall Inc., 1984

7. Beckwith T., and Marangoni R., Electrical Measurements, 4th edition, Addison Wesley, Massachusetts, 1990

APPENDICES

APPENDIX A

THE DERIVATION FOR ATTENUATION

From Maxwell's equation we have

$$\nabla \times H = \frac{\partial D}{\partial t} + \sigma E$$

$$\nabla \times E = -\frac{\partial B}{\partial t}$$

Assuming a sinusoidal form and suppressing $e^{j\varpi t}$ we have

$$\nabla \times H = j\omega\varepsilon E + \sigma E$$

$$\nabla \times E = -j\omega\mu H$$

Take the cross product of $\nabla \times H$

$$\nabla \times \nabla \times H = (j\omega\varepsilon + \sigma) \cdot (\nabla \times E)$$

Use $\nabla \times H$ in the previous expression

$$\nabla \times \nabla \times H = (j\omega\varepsilon + \sigma) \cdot (-j\omega\mu H)$$

From vector math

$$\nabla \times \nabla \times H = \nabla(\nabla \cdot H) - \nabla^2 H$$

But from Maxwell's second equation

$$\nabla \cdot B = 0$$

so

$$\nabla \cdot \mu H = 0$$

$$\nabla \cdot H = 0$$

thus

$$\nabla \times \nabla \times H = -\nabla^2 H$$

and

$$-\nabla^2 H = (j\omega\varepsilon + \sigma)\cdot(-j\omega\mu H)$$

then assuming one direction propogation

$$\frac{-\partial^2 H}{\partial x^2} = (j\omega\varepsilon + \sigma)\cdot(-j\omega\mu H)$$

$$\frac{\partial^2 H}{\partial x^2} = -\omega^2\mu\varepsilon H + j\sigma\omega\mu H$$

Solving for the characteristic root

$$D^2 = -\omega^2\mu\varepsilon + j\sigma\omega\mu$$

and

$$D = \pm(-\omega^2\mu\varepsilon + j\sigma\omega\mu)^{\frac{1}{2}}$$

or

$$D = \pm j\omega\sqrt{\mu\varepsilon}\left(1 - \frac{j\sigma}{\omega\varepsilon}\right)^{\frac{1}{2}}$$

By Newton expansion for $(1+x)^{1/2}$ is

$$(1+x)^{\frac{1}{2}} = 1 + \frac{1}{2}x - \frac{1}{8}x^2 + \ldots$$

neglect everything except the two terms

$$\left(1 - \frac{j\sigma}{\omega\varepsilon}\right)^{\frac{1}{2}} \approx \left(1 - \frac{j\sigma}{2\omega\varepsilon}\right)$$

Then making the substitution

$$D \approx \pm j\omega\sqrt{\mu\varepsilon}\left(1-\frac{j\sigma}{2\omega\varepsilon}\right)$$

and distributing the complex and real part, the roots are

$$D = \pm j\omega\sqrt{\mu\varepsilon} \pm \frac{\sigma\sqrt{\mu\varepsilon}}{2\varepsilon}$$

Then using the characteristic root for solving the differential equation of H

$$H(x,t) = H^{+}e^{j\omega t}e^{-\left(j\omega\sqrt{\mu\varepsilon}+\frac{\sigma\sqrt{\mu\varepsilon}}{2\varepsilon}\right)\cdot x} + H^{-}e^{j\omega t}e^{\left(j\omega\sqrt{\mu\varepsilon}+\frac{\sigma\sqrt{\mu\varepsilon}}{2\varepsilon}\right)\cdot x}$$

The Eulers equation and algebra let the function H(x,t) (magnetic field intensity) to be put in terms of a sinusoidal form

$$H(x,t) = H^{+}e^{-\frac{\sigma}{2\omega\varepsilon}\cdot x}\cos\left(\omega t - \omega\sqrt{\mu\varepsilon}\right) + H^{-}e^{\frac{\sigma}{2\omega\varepsilon}\cdot x}\cos\left(\omega t + \omega\sqrt{\mu\varepsilon}\right)$$

The result is a cosine propogation being attenuated by $e^{-\frac{\sigma}{2\mu\varepsilon}\cdot x}$ with distance x, which is the attenuation of amplitude through the soil. Because the electric field (E) is derived by multiplying the magnetic field (H) by the characteristic impedance Z_0, the electric field (E)has the same attenuation as (H). The total distance of propagation is 2L (twice the length of the probe) so the attenuation probe is:

$$f^2 = e^{-2\alpha L}$$

where

$$\alpha = \frac{\sigma\sqrt{\mu\varepsilon}}{2\varepsilon}$$

Now the attenuation α can be placed into the familiar form of equation (17) by the equation (23).

$$V = \frac{c}{(K')^{\frac{1}{2}}}$$

where

$$V = \frac{1}{\sqrt{\mu\varepsilon}}$$

so:

$$\sqrt{\mu\varepsilon} = \frac{(K')^{\frac{1}{2}}}{c}$$

Then

$$\alpha = \frac{\sigma(K')^{\frac{1}{2}}}{2\varepsilon c}$$

but $\varepsilon=\varepsilon_0 K'$, substitute into the above equation and reduce, the result is

$$\alpha = \frac{\sigma}{2\varepsilon_0 c(K')^{\frac{1}{2}}}$$

APPENDIX B

RELATIVE DIELECTRIC CONSTANT K DEDUCED FOR COAXIAL PROBE

The inductance for an air core coaxial line is (ignoring external inductance):

$$L = \frac{\mu_0}{2\pi} \ln\left(\frac{b}{a}\right)$$

and the capacitance for the air core coaxial line is:

$$C = \frac{2\pi L \varepsilon_0}{\ln\left(\frac{b}{a}\right)}$$

The coaxial air core characteristic impedance is:

$$Z_0 = \sqrt{\frac{L}{C}} = \frac{1}{2\pi}\sqrt{\frac{\mu_0}{\varepsilon_0}} \ln\left(\frac{b}{a}\right)$$

Also, the dielectric constant was defined as

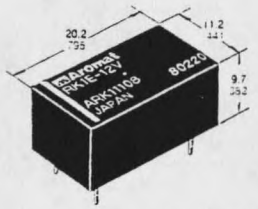
$$K^{\frac{1}{2}} = \frac{Z_0}{Zs} = \frac{Z_0}{R_{PROBE}} = \sigma L \sqrt{\frac{\mu_0}{\varepsilon_0}} = \frac{\sigma L}{\varepsilon_0 C}$$

$$R_{PROBE} = \frac{\ln\left(\frac{b}{a}\right)}{2\pi\sigma L}$$

APPENDIX C

AROMAT DATA SHEETS

NAiS | HIGH FREQUENCY RELAY | RK-RELAYS



mm inch

- Excellent high frequency characteristics
  Isolation: 60 dB or more (at 1.5 GHz)
  Insertion loss: 0.3 dB or less (at 900 MHz)
- High sensitivity in small size
  Size: 20.2×11.2×9.7 mm .795×.441×.382 inch
  Nominal power consumption: 200 mW (single side stable type)
- Sealed construction for automatic cleaning
- Latching types are also available

SPECIFICATIONS

Contact

Arrangement		1 Form C
Contact material		Gold-clad silver
Initial contact resistance max. (By HP4328A)		100 mΩ
Rating		
Max. switching power		10 W
Max. switching voltage		30 V DC
Max. switching current		0.01 A
Nominal switching capacity		0.01 A 24 V DC (10 W at 1.2 GHz, Zo = 50 Ω system)
High frequency characteristics (50 Ω system)		
	Isolation loss	Min. 60 dB (at 1.5 GHz)
	Insertion loss	Max. 0.3 dB (at 900 MHz)
	V.S.W.R.	Max. 1.5 (at 900 MHz)
Expected life (min. operations)		
Mechanical		5×10^6
Electrical	0.01 A 24 V DC	3×10^5
	10 W 1.2 GHz	10^5

Coil (at 25°C, 68°F)

	Nominal operating power
Single side stable	200 mW
1 coil latching	200 mW
2 coil latching	400 mW

Characteristics

Initial insulation resistance, min.		100 MΩ at 500 V DC
Initial breakdown voltage		
Between open contacts		500 Vrms
Between contact and coil		1,000 Vrms
Between contact and earth terminal		500 Vrms
Operate time* (at nominal voltage)		Approx. 6 msec.
Release time* (at nominal voltage)		Approx. 3 msec.
Temperature rise (max.)		60°C with nominal coil voltage across coil and rated contact current
Shock resistance	Functional	Min. 196 m/s² (20 G)
	Destructive	Min. 980 m/s² (100 G)
Vibration resistance	Functional	10 to 55 Hz at double amplitude of 3 mm
	Destructive	10 to 55 Hz at double amplitude of 5 mm
Ambient temperature		-40°C to 60°C -40°F to 140°F
Unit weight		Approx. 4.4 g .155 oz

* Excluding contact bounce time

TYPICAL APPLICATIONS

- Audio visual equipment
  broadcast satellite tuners
  VCRs, CATVs, TVs
- Communication equipment
  automobile telephones
  maritime telephones
- Instrumentation
  test equipment
  measuring equipment

ORDERING INFORMATION

Ex. RK [1] E — [L2] — [24V]

Contact arrangement	Operating function	Coil voltage, DC
1: Standard type 1R: R type (See Schematic on next page.)	Nil: Single side stable L: 1 coil latching L2: 2 coil latching	3, 5, 6, 9, 12, 24 V

TYPES AND COIL DATA (at 20°C 68°F)

• Single side stable type

Part No.		Nominal voltage, V DC	Pick-up voltage, max. V DC	Drop-out voltage, min. V DC	Coil resistance, Ω (±10%)	Nominal operating current, mA	Nominal operating power, mW	Maximum allowable voltage, V DC (at 60°C)
RK1E-3V	RK1RE-3V	3	2.25	0.3	45	67	200	3.3
RK1E-5V	RK1RE-5V	5	3.75	0.5	125	40	200	5.5
RK1E-6V	RK1RE-6V	6	4.5	0.6	180	34	200	6.6
RK1E-9V	RK1RE-9V	9	6.75	0.9	405	23	200	9.9
RK1E-12V	RK1RE-12V	12	9	1.2	720	17	200	13.2
RK1E-24V	RK1RE-24V	24	18	2.4	2,880	9	200	26.4

• 1 coil latching type

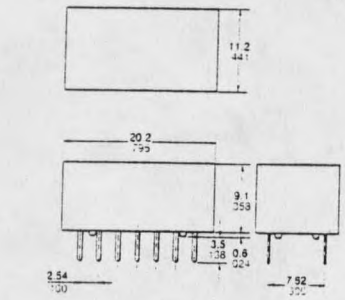
Part No.		Nominal voltage, V DC	Set voltage, max. V DC	Reset voltage, max. V DC	Coil resistance, Ω (±10%)	Nominal operating current, mA	Nominal operating power, mW	Maximum allowable voltage, V DC (at 60°C)
RK1E-L-3V	RK1RE-L-3V	3	2.25	2.25	45	67	200	3.3
RK1E-L-5V	RK1RE-L-5V	5	3.75	3.75	125	40	200	5.5
RK1E-L-6V	RK1RE-L-6V	6	4.5	4.5	180	34	200	6.6
RK1E-L-9V	RK1RE-L-9V	9	6.75	6.75	405	23	200	9.9
RK1E-L-12V	RK1RE-L-12V	12	9	9	720	17	200	13.2
RK1E-L-24V	RK1RE-L-24V	24	18	18	2,880	9	200	26.4

• 2 coil latching type

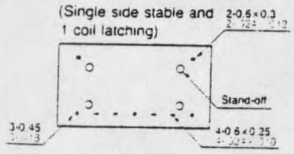
Part No.		Nominal voltage, V DC	Set voltage, max. V DC	Reset voltage, max. V DC	Coil resistance, Ω (±10%)	Nominal operating current, mA	Nominal operating power, mW	Maximum allowable voltage, V DC (at 60°C)
RK1E-L2-3V	RK1RE-L2-3V	3	2.25	2.25	22.5	133	400	3.3
RK1E-L2-5V	RK1RE-L2-5V	5	3.75	3.75	62.5	80	400	5.5
RK1E-L2-6V	RK1RE-L2-6V	6	4.5	4.5	90	67	400	6.6
RK1E-L2-9V	RK1RE-L2-9V	9	6.75	6.75	202.5	45	400	9.9
RK1E-L2-12V	RK1RE-L2-12V	12	9	9	360	34	400	13.2
RK1E-L2-24V	RK1RE-L2-24V	24	18	18	1440	17	400	26.4

DIMENSIONS

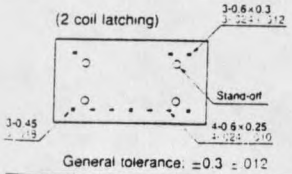
mm inch



(Single side stable and 1 coil latching)



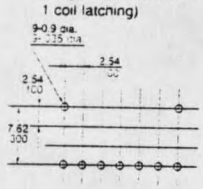
(2 coil latching)



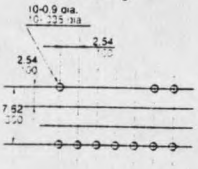
General tolerance: ±0.3 ±.012

PC board pattern (Copper-side view)

(Single side stable and 1 coil latching)



(2 coil latching)

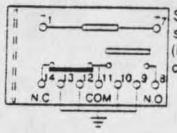


Tolerance: ±0.1 ±.003

Schematic (Bottom view)

Standard contact arrangement

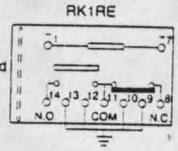
RK1E



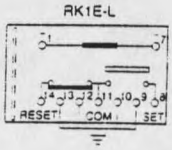
Single side stable (Deenergized condition)

R type contact arrangement

RK1RE

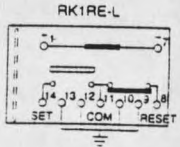


RK1E-L

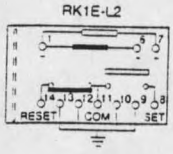


1 coil latching (Reset condition)

RK1RE-L

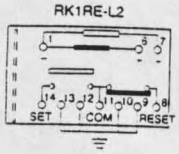


RK1E-L2



2 coil latching (Reset condition)

RK1RE-L2

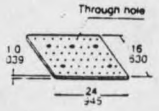


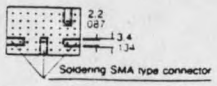
RK

DATA

mm inch

1. High frequency characteristics
Sample: RK1E-12V
No. of samples: n = 10 (10×2 contacts)

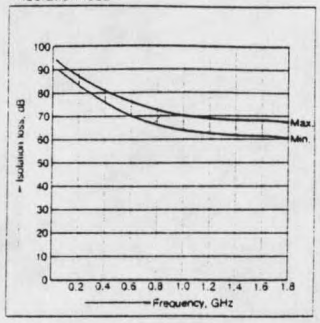




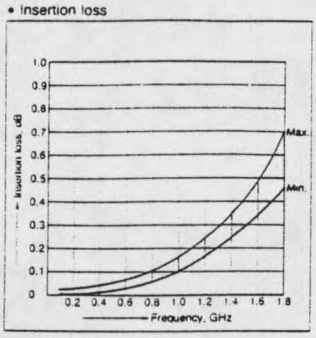
(Wiring diagram)

PC board
- Double-sided through hole
- Material: Glass-epoxy resin

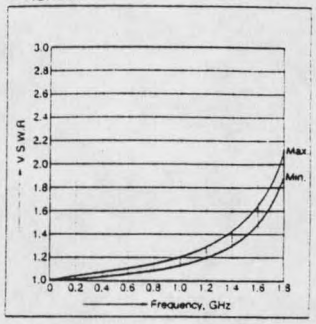
• Isolation loss



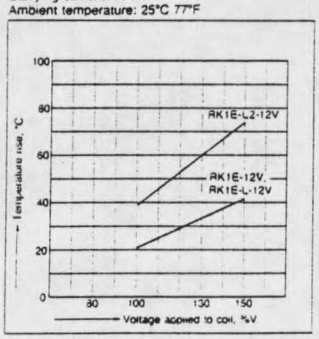
• Insertion loss



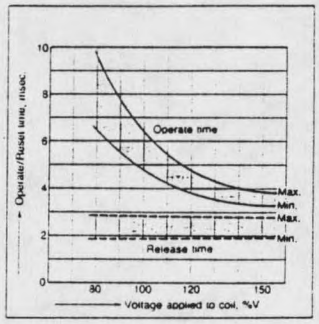
• V.S.W.R.



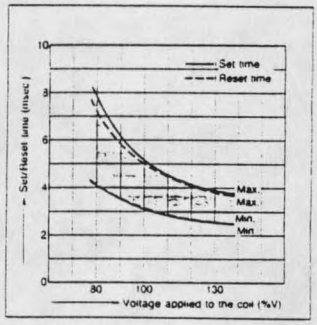
2. Coil temperature rise
Sample: RK1E-12V, RK1E-L-12V, RK1E-L2-12V
No. of samples: n = 6
Carrying current: 10 mA
Ambient temperature: 25°C 77°F



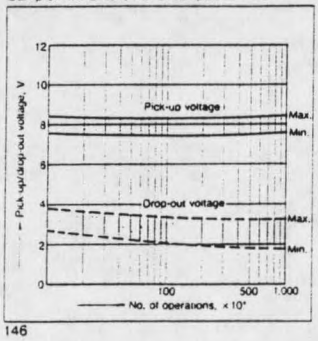
3.-(1) Operate/Release time
(Single side stable)
Sample: RK1E-12V; No. of samples: n = 6



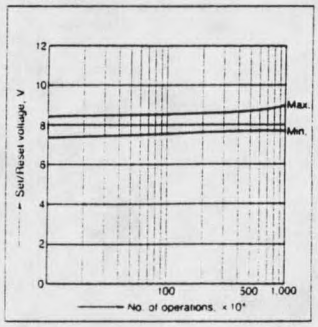
3.-(2) Set/Reset time (Latching)
Sample: RK1E-L-12V, RK1E-L2-12V
No. of samples: n = 12



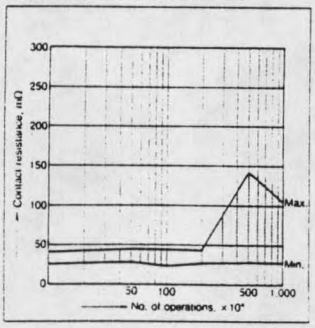
4.-(1) Mechanical life test
(Single side stable)
Sample: RK1E-12V; No. of samples: n = 12



4.-(2) Mechanical life test (Latching)
Sample: RK1E-L2-12V
No. of samples: n = 12

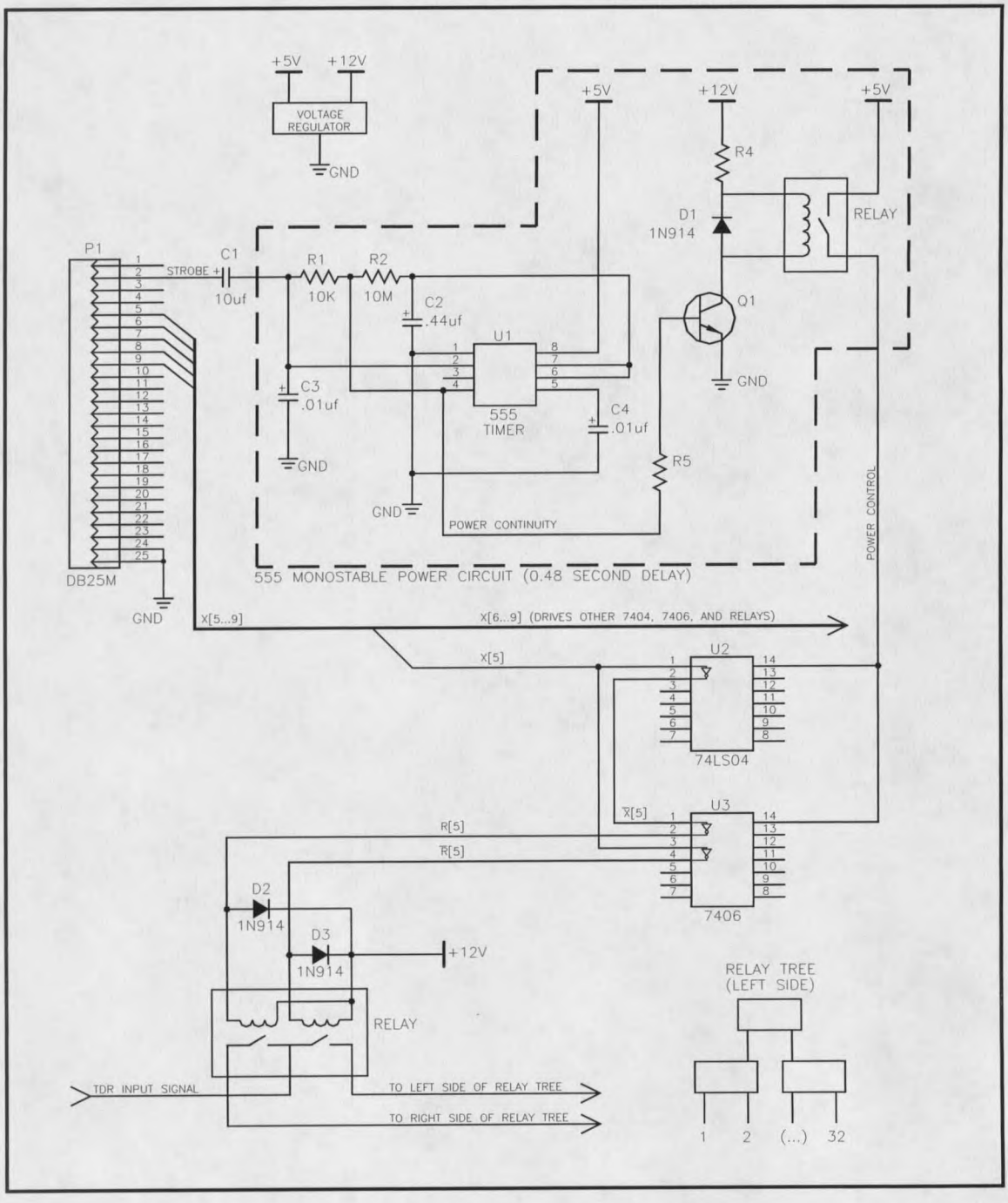


4.-(3) Mechanical life test
Sample: RK1E-12V
No. of samples: n = 20 (20×2 contacts)



APPENDIX D

SCHEMATIC OF TDR MULTIPLEXER


+5V
+12V
VOLTAGE REGULATOR
GND
+5V
+12V
+5V
R4
D1
1N914
RELAY
Q1
GND
P1
STROBE
C1
10uf
R1
10K
R2
10M
C2
.44uf
U1
555
TIMER
C3
.01uf
C4
.01uf
R5
GND
GND
POWER CONTINUITY
POWER CONTROL
DB25M
GND
555 MONOSTABLE POWER CIRCUIT (0.48 SECOND DELAY)
X[5...9]
X[6...9] (DRIVES OTHER 7404, 7406, AND RELAYS)
X[5]
U2
74LS04
X[5]
U3
R[5]
R[5]
7406
D2
1N914
D3
1N914
+12V
RELAY
TDR INPUT SIGNAL
TO LEFT SIDE OF RELAY TREE
TO RIGHT SIDE OF RELAY TREE
RELAY TREE
(LEFT SIDE)
1
2
(...)
32

APPENDIX E

C SOURCE CODE FOR W.C AND WWIN.C

```
/*
Programmer:  Rod D. Carlson
Date of production:  3-94
Utility objective:  TDR Multiplexer
Source file:  W.C
Object file:  W.OBJ
Executable file: W.EXE
Compiled in Microsoft QuickC Ver 2.50/Small model

Copyright (C) 1994, All rights reserved

Information in this file is provided to others on acceptance that said
contents will not be distributed without prior permission.

Inquiries can be directed to:

Rod Carlson
2219 W College St. #87
Bozeman, MT  59715
(406)586-2161
*/

#include <stdio.h>
#include <dos.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include "tsr.h"
#include "scr.h"
#include "kbd.h"
#include "mouse.h"
#include "wset.h"

#pragma check_stack( off )
#pragma check_pointer( off )

static void wave(int keystroke);
static void activate_wave(void);
/*static void waveguide_engine(char pathname[]); */
static void waveguide_engine(int);


/*
Function definitions:
wave:  function handler for screen and keyboard manipulation. Gets
the waveguide from user, processes information and calls
waveguide_engine.
```

activate_wave: wave determines whether it is to collect menu information from the user or to get it from DOS if the user typed it from the command line. Since the TSR handler stores a function pointer in the form void function(void), activate_wave acts as a buffer to the function wave which is in the form void wave (determination) when called by the TSR handler.

waveguide_engine: waveguide_engine takes the information passed to it by its parent and determines what byte to output to the parrallel port. It then charges up the triggering capacitor and sets the waveguide through the function guide_set. In addition, waveguide_engine sends out another byte to control an optional LED indicator. The current waveguide configuration is stored and the function then returns to its parent.

```
FILE *filelist,*tempfile,*actfile; /*handles to temporary-data files*/
unsigned long time_out; /*used in delay loops*/
unsigned cursor_char_1,cursor_char_2; /*stores current cursor characteristics*/

main(int argc, char **argv)
{
        int flag,strokes,operation; /*maintains what field user is typing data,
        how many keystrokes have been typed in field.*/
    /* char pathname[_MAX_PATH];*/ /*destination variable of path file information*/
        char pathname[_MAX_PATH];
        char password[15]; /*destination of user's password*/
        if (argc < 2)
        {
                printf("\n\nWAVEGUIDE can be utilized two ways. The user can elect to make it
memory");
                printf("\nresident and activate it with a special hotkey combination at anytime");
                printf("\nor the program can run as an application from the DOS command line. If
you");
                printf("\nelect to use the program in its memory resident status, do not activate it");
                printf("\nwhile in applications that are currently in a graphics mode. If you are");
                printf("\nunsure whether a particular application uses graphic modes, load the");
                printf("\napplication and activate WAVEGUIDE with the hotkey. If the keyboard or");
                printf("\nor the video screen hang, it is not safe to activate WAVEGUIDE within");
                printf("\nthis application.");
                printf("\n\nTo install WAVEGUIDE as memory resident:");
                printf("\nType at the DOS command line 'W M'");
                printf("\n\nTo run WAVEGUIDE at the DOS command line:");
                printf("\nType at the DOS command line 'W #' # indicates the waveguide number");
                printf("\n\n\CONTROL + LEFTSHIFT activates the memory resident
WAVEGUIDE\n\n\n");
                printf("\n\nCopyright (C) 1994 by Rod Carlson");
                exit(0);
        }
```

```
        if (*argv[1] == 'm' || *argv[1] == 'M')
        {
                printf("\n\n\nWAVEGUIDE TSR has been installed");
                printf("\nPress 'CONTROL + LEFT SHIFT' to activate");
                printf("\n\nCopyright (C) 1994 by Rod Carlson");
                printf("\nAll rights reserved\n");
                printf("\n\n\Do not activate WAVEGUIDE when graphics is active");
                switch (tsrinstall(activate_wave,CONTROL | LEFTSHIFT,0x01011001))
                {
                        case WRONGDOS:
                                scrclear(ALL);
                                setcursor(0,0);
                                printf("Cannot install TSR -<DOS 2.0");
                                exit(1);
                        case INSTALLED:
                                scrclear(ALL);
                                setcursor(0,0);
                                printf("TSR already installed");
                                exit(1);
                        case NOINT:
                                scrclear(ALL);
                                setcursor(0,0);
                                printf("Cannot find free interrupts - TSR not installed");
                                exit(1);
                        case ERROR:
                        default:
                                scrclear(ALL);
                                setcursor(0,0);
                                printf("Unknown error during installation - TSR not installed");
                                exit(1);
                }
        }
        else
                waveguide_engine(atoi(argv[1]));

}

static void activate_wave(void)
{
        wave(0);
        return;
}

static void wave(int keystroke)
{

        unsigned strokes = 0; /*same action as in main*/
        int keycode,control = 0,flag = 0; /*variable that BIOS keyboard
```

```
service routine dumps keystroke to, variable where the attributes
of a function key are stored, and what field user is typing information
to*/
unsigned int x, y, button; /*mouse x-y coordinates and button select*/
unsigned int i, j, spacex=24, spacey=8, blocksizx=80, blocksizy=16;
char pathname[_MAX_PATH]; /*same as in main*/
char buffer[80]; /*temporary work buffer*/
scrgetstyle(&cursor_char_1,&cursor_char_2);
scrsetstyle(33,33);
if (keystroke)
        return;
scrpush();
scrputs("                                                              ",0x1f,0,0);
scrputs("                                                              ",0x1f,1,0);
scrputs("                                                              ",0x1f,2,0);
scrputs("                                                              ",0x1f,3,0);
scrputs("    CH. 1     CH. 2     CH. 3     CH. 4     CH. 5     CH. 6 ",0x1f,4,0);
scrputs("                                                              ",0x1f,5,0);
scrputs("                                                              ",0x1f,6,0);
scrputs("    CH. 7     CH. 8     CH. 9     CH. 10    CH. 11    CH. 12",0x1f,7,0);
scrputs("                                                              ",0x1f,8,0);
scrputs("                                                              ",0x1f,9,0);
scrputs("    CH. 13    CH. 14    CH. 15    CH. 16    CH. 17    CH. 18",0x1f,10,0);
scrputs("                                                              ",0x1f,11,0);
scrputs("                                                              ",0x1f,12,0);
scrputs("    CH. 19    CH. 20    CH. 21    CH. 22    CH. 23    CH. 24 ",0x1f,13,0);
scrputs("                                                              ",0x1f,14,0);
scrputs("                                                              ",0x1f,15,0);
scrputs("    CH. 25    CH. 26    CH. 27    CH. 28    CH. 29    CH. 30",0x1f,16,0);
scrputs("                                                              ",0x1f,17,0);
scrputs("                                                              ",0x1f,18,0);
scrputs("    CH. 31    CH. 32    GTI TDR M2980                         ",0x1f,19,0);
scrputs("                        Controller Software                   ",0x1f,20,0);
scrputs("  Keyboard Input  Last   (C) Copyright 1994    OK     Cancel   ",0x1f,21,0);
scrputs("                                                              ",0x1f,22,0);
scrputs("                                                              ",0x1f,23,0);
scrputs("                                                              ",0x1f,24,0);
scrputs("                                                              ",0x1f,25,0);
mouseon();
while (keystroke != 27 && !flag)
{
        scrputc(221,0x1f | 0x80,22,3 + strokes);
        keycode = kbdgetcgo();
        control = (keycode >> 8) & 0x00ff;
        keystroke = keycode & 0x00ff;
        if (keystroke)
        {
                switch(keystroke)
```

```
        {
                case 8: /*Backspace*/
                        if (strokes)
                        {
                                scrputc(' ',0x1f,22,3 + strokes);
                                strokes--;
                                pathname[strokes] = '\0';
                        }
                        break;
                default: /*Any other key*/
                        if (strokes < 15 && isgraph(keystroke))
                        {
                                scrputc(keystroke,0x1f,22,3 + strokes);
                                pathname[strokes] = keystroke;
                                strokes++;
                                pathname[strokes] = '\0';
                        }
                        if ((strokes) && keystroke == 13)  /*Return*/
                        {       waveguide_engine(atoi(pathname));
                                flag=1;
                        }
                   /*  break;  */
        }

}
mousexy(&x, &y, &button);
if ( (x >= 512 && x <= 600) && (y >= 160 && y <= 176) )
{
        scrputs(" Cancel ", 0x78,21,66);
        mouseoff();
}
else
{
        scrputs(" Cancel ", 0x1f,21,66);
        mouseon();
}
if ( (x >= 432 && x <= 488) && (y >= 160 && y <= 176) )
{
        scrputs(" OK ", 0x78,21,56);
        mouseoff();
}
else
{
        scrputs(" OK ", 0x1f,21,56);
        mouseon();
}
if (button)
{
```

```
                    for (i=1,j=1; (j <= 6); j++)  /*Mouse in CH 1-32*/
                    {
                            for (i=1; (i <= 6 && i+6*(j-1) <= 32) ; i++)
                            {

                                    if (x >= i*spacex+(i-1)*blocksizx+16 && x <=
i*(spacex+blocksizx)+16
                                       && y >= j*spacey+(j-1)*blocksizy+24 && y <=
j*(spacey+blocksizy)+24)
                                    {
                                            itoa(i+6*(j-1),pathname,10);
                                            scrputs("          ", 0x1f,22,3);
                                            scrputs(pathname, 0x1f,22,3);
                                            strokes=strlen(pathname);
                                            scrputc(221,0x1f | 0x80,22,3 + strokes);

                                    }
                            }
                    }
                    if ( (x >= 512 && x <= 600) && (y >= 160 && y <= 176) )
                    {
                            flag=1;
                            break;
                    }
                    if ( (x >= 432 && x <= 488) && (y >= 160 && y <= 176) )
                    {
                            waveguide_engine(atoi(pathname));
                            flag=1;
                            break;
                    }
            }
        }
        scrpop(1);
```

W.C

```
        scrsetstyle(cursor_char_1,cursor_char_2);
        mousexy(&x, &y, &button);
        mouseoff();
}

static void waveguide_engine(int path)
{
        int bin, waveguide;
        waveguide=path;
        printf("%d",waveguide);
        cap_charge(waveguide);
```

```
        printf("%s","We got here1");
        time_delay(.7);
        printf("%s","We got here2");
        guide_set(&bin, waveguide);
        time_delay(.6);
        /* bin_out(bin); */
        wav_save(waveguide);
        printf("%s","We got here3");
        return;
}
```

WWIN.C

```
/*
Programmer:  Rod D. Carlson
Date of production:  3-94
Utility objective:  TDR Multiplexer
Source file:  W.C
Object file:  W.OBJ
Executable file: W.EXE
Compiled in Microsoft QuickC Ver 2.50/Small model
```


Inquiries can be directed to:

Rod Carlson
2219 W College St. #87
Bozeman, MT 59715
(406)586-2161

```
*/

#include <stdio.h>
#include <dos.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include "tsr.h"
#include "scr.h"
#include "kbd.h"
#include "mouse.h"
#include "wset.h"
#include <conio.h>
#include <stdio.h>
#include <process.h>

#define PORT956 0x3bc
#define PORT888 0x378

#pragma check_stack( off )
#pragma check_pointer( off )

static void wave(int keystroke);
static void activate_wave(void);
/*static void waveguide_engine(char pathname[]); */
```

```
static void waveguide_engine(int);

void guide_set(int *, int);
void cap_charge(int);
void bin_out(int);
void time_delay(float);
void wav_save(int);


/*
Function definitions:
wave:  function handler for screen and keyboard manipulation. Gets
the waveguide from user, processes information and calls
waveguide_engine.

waveguide_engine:  waveguide_engine takes the information passed to it by
its parent and determines what byte to output to the parrallel port.  It then
charges up the triggering capacitor and sets the waveguide through the
function guide_set.  In addition, waveguide_engine sends out another byte to
control an optional LED indicator.  The current waveguide configuration is
stored and the function then returns to its parent.


FILE *filelist,*tempfile,*actfile; /*handles to temporary-data files*/
unsigned long time_out; /*used in delay loops*/
unsigned cursor_char_1,cursor_char_2; /*stores current cursor characteristics*/

main()
{
        int operation,keystroke=0; /*maintains what field user is typing data,
        how many keystrokes have been typed in field.*/
    /* char pathname[_MAX_PATH];*/ /*destination variable of path file information*/
        char pathname[_MAX_PATH];
        char password[15]; /*destination of user's password*/
        unsigned strokes = 0; /*same action as in main*/
        int keycode,control = 0,flag = 0; /*variable that BIOS keyboard
        service routine dumps keystroke to, variable where the attributes
        of a function key are stored, and what field user is typing information
        to*/
        unsigned int x, y, button; /*mouse x-y coordinates and button select*/
        unsigned int i, j, spacex=24, spacey=8, blocksizx=80, blocksizy=16;

        char buffer[80]; /*temporary work buffer*/
        scrgetstyle(&cursor_char_1,&cursor_char_2);
        scrsetstyle(33,33);
        scrpush();
        scrputs("                                                    ",0x1f,0,0);
        scrputs("                                                    ",0x1f,1,0);
```

```
scrputs("                                                                    ",0x1f,2,0);
scrputs("                                                                    ",0x1f,3,0);
scrputs(" ³ CH. 1 ³ ³ CH. 2 ³ ³ CH. 3 ³ ³ CH. 4 ³ ³ CH. 5 ³ ³ CH. 6 ³ ",0x1f,4,0);
scrputs("                                                                    ",0x1f,5,0);
scrputs("                                                                    ",0x1f,6,0);
scrputs(" ³ CH. 7 ³ ³ CH. 8 ³ ³ CH. 9 ³ ³ CH. 10 ³ ³ CH. 11 ³ ³ CH. 12 ³ ",0x1f,7,0);
scrputs("                                                                    ",0x1f,8,0);
scrputs("                                                                    ",0x1f,9,0);
scrputs(" ³ CH. 13 ³ ³ CH. 14 ³ ³ CH. 15 ³ ³ CH. 16 ³ ³ CH. 17 ³ ³ CH. 18 ³ ",0x1f,10,0);
scrputs("                                                                    ",0x1f,11,0);
scrputs("                                                                    ",0x1f,12,0);
scrputs(" ³ CH. 19 ³ ³ CH. 20 ³ ³ CH. 21 ³ ³ CH. 22 ³ ³ CH. 23 ³ ³ CH. 24 ³ ",0x1f,13,0);
scrputs("                                                                    ",0x1f,14,0);
scrputs("                                                                    ",0x1f,15,0);
scrputs(" ³ CH. 25 ³ ³ CH. 26 ³ ³ CH. 27 ³ ³ CH. 28 ³ ³ CH. 29 ³ ³ CH. 30 ³ ",0x1f,16,0);
scrputs("                                                                    ",0x1f,17,0);
scrputs("                                                                    ",0x1f,18,0);
scrputs(" ³ CH. 31 ³ ³ CH. 32 ³              GTI TDR M2980                   ",0x1f,19,0);
scrputs("                              Controller Software                   ",0x1f,20,0);
scrputs("  Keyboard Input    (C) Copyright 1994    OK      Cancel            ",0x1f,21,0);
scrputs("                                                                    ",0x1f,22,0);
scrputs("                                                                    ",0x1f,23,0);
scrputs("                                                                    ",0x1f,24,0);
scrputs("                                                                    ",0x1f,25,0);
mouseon();
while (keystroke != 27 && !flag)
{
        scrputc(221,0x1f | 0x80,22,3 + strokes);
        keycode = kbdgetcgo();
        control = (keycode >> 8) & 0x00ff;
        keystroke = keycode & 0x00ff;
        if (keystroke)
        {
                switch(keystroke)
                {
                        case 8:  /*Backspace*/
                                if (strokes)
                                {
                                        scrputc(' ',0x1f,22,3 + strokes);
                                        strokes--;
                                        pathname[strokes] = '\0';
                                }
                                break;
                        default: /*Any other key*/
                                if (strokes < 15 && isgraph(keystroke))
                                {
                                        scrputc(keystroke,0x1f,22,3 + strokes);
                                        pathname[strokes] = keystroke;
```

```
                                        strokes++;
                                        pathname[strokes] = '\0';
                                }
                                if ((strokes) && keystroke == 13)  /*Return*/
                                {       waveguide_engine(atoi(pathname));
                                        flag=1;
                                }
                        /*  break;   */
                }

        }
        mousexy(&x, &y, &button);
        if ( (x >= 512 && x <= 600) && (y >= 160 && y <= 176) )
        {
                scrputs(" Cancel ", 0x78,21,66);
                mouseoff();
        }
        else
        {
                scrputs(" Cancel ", 0x1f,21,66);
                mouseon();
        }
        if ( (x >= 432 && x <= 488) && (y >= 160 && y <= 176) )
        {
                scrputs(" OK ", 0x78,21,56);
                mouseoff();
        }
        else
        {
                scrputs(" OK ", 0x1f,21,56);
                mouseon();
        }
        if (button)
        {

                for (i=1,j=1; (j <= 6); j++)  /*Mouse in CH 1-32*/
                {
                        for (i=1; (i <= 6 && i+6*(j-1) <= 32) ; i++)
                        {

                                if (x >= i*spacex+(i-1)*blocksizx+16 && x <=
i*(spacex+blocksizx)+16
                                   && y >= j*spacey+(j-1)*blocksizy+24 && y <=
j*(spacey+blocksizy)+24)
                                {
                                        itoa(i+6*(j-1),pathname,10);
                                        scrputs("          ", 0x1f,22,3);
                                        scrputs(pathname, 0x1f,22,3);
```

```
                                        strokes=strlen(pathname);
                                        scrputc(221,0x1f | 0x80,22,3 + strokes);

                                }
                        }
                }
                if ( (x >= 512 && x <= 600) && (y >= 160 && y <= 176) )
                {
                        flag=1;
                        break;
                }
                if ( (x >= 432 && x <= 488) && (y >= 160 && y <= 176) )
                {
                        waveguide_engine(atoi(pathname));
                        flag=1;
                        break;
                }
        }
    }
    scrpop(1);
    scrsetstyle(cursor_char_1,cursor_char_2);
    mousexy(&x, &y, &button);
    mouseoff();
}

static void waveguide_engine(int path)
{
 int bin, waveguide;
 waveguide=path;
 cap_charge(waveguide);
 time_delay(.7);
 guide_set(&bin, waveguide);
 time_delay(.6);
/* bin_out(bin); */
 wav_save(waveguide);
}

void cap_charge(int waveguide)
{
 if (waveguide <= 32 && waveguide >= 1)
   {
    outp(PORT888, 0x1);
    outp(PORT956, 0x1);
   }
 return;
}

void bin_out(int bin)
```

```
{
 outp(PORT888, bin);
 outp(PORT956, bin);
 return;
}

void guide_set(int *pbin, int waveguide)
{

 switch (waveguide) {
 case 1:
   outp(PORT888, 0x4e);
   outp(PORT956, 0x4e);
   *pbin=2;
   break;
 case 2:
   outp(PORT888, 0xb6);
   outp(PORT956, 0xb6);
   *pbin=4;
   break;
 case 3:
   outp(PORT888, 0xbe);
   outp(PORT956, 0xbe);
   *pbin=6;
   break;
 case 4:
   outp(PORT888, 0xa6);
   outp(PORT956, 0xa6);
   *pbin=8;
   break;
 case 5:
   outp(PORT888, 0xae);
   outp(PORT956, 0xae);
   *pbin=10;
   break;
 case 6:
   outp(PORT888, 0x96);
   outp(PORT956, 0x96);
   *pbin=12;
   break;
 case 7:
   outp(PORT888, 0x9e);
   outp(PORT956, 0x9e);
   *pbin=14;
   break;
 case 8:
   outp(PORT888, 0x86);
   outp(PORT956, 0x86);
```

```
    *pbin=16;
    break;
  case 9:
    outp(PORT888, 0x8e);
    outp(PORT956, 0x8e);
    *pbin=18;
    break;
  case 10:
    outp(PORT888, 0xf6);
    outp(PORT956, 0xf6);
    *pbin=160;
    break;
  case 11:
    outp(PORT888, 0xfe);
    outp(PORT956, 0xfe);
    *pbin=162;
    break;
  case 12:
    outp(PORT888, 0xe6);
    outp(PORT956, 0xe6);
    *pbin=164;
    break;
  case 13:
    outp(PORT888, 0xee);
    outp(PORT956, 0xee);
    *pbin=166;
    break;
  case 14:
    outp(PORT888, 0xd6);
    outp(PORT956, 0xd6);
    *pbin=168;
    break;
  case 15:
    outp(PORT888, 0xde);
    outp(PORT956, 0xde);
    *pbin=170;
    break;
  case 16:
    outp(PORT888, 0xc6);
    outp(PORT956, 0xc6);
    *pbin=172;
    break;
  case 17:
    outp(PORT888, 0xce);
    outp(PORT956, 0xce);
    *pbin=174;
    break;
  case 18:
```

```
    outp(PORT888, 0x36);
    outp(PORT956, 0x36);
    *pbin=176;
    break;
  case 19:
    outp(PORT888, 0x3e);
    outp(PORT956, 0x3e);
    *pbin=178;
    break;
  case 20:
    outp(PORT888, 0x26);
    outp(PORT956, 0x26);
    *pbin=192;
    break;
  case 21:
    outp(PORT888, 0x2e);
    outp(PORT956, 0x2e);
    *pbin=194;
    break;
  case 22:
    outp(PORT888, 0x16);
    outp(PORT956, 0x16);
    *pbin=196;
    break;
  case 23:
    outp(PORT888, 0x1e);
    outp(PORT956, 0x1e);
    *pbin=198;
    break;
  case 24:
    outp(PORT888, 0x6);
    outp(PORT956, 0x6);
    *pbin=200;
    break;
  case 25:
    outp(PORT888, 0xe);
    outp(PORT956, 0xe);
    *pbin=202;
    break;
  case 26:
    outp(PORT888, 0x76);
    outp(PORT956, 0x76);
    *pbin=204;
    break;
  case 27:
    outp(PORT888, 0x7e);
    outp(PORT956, 0x7e);
    *pbin=206;
```

```
    break;
  case 28:
    outp(PORT888, 0x66);
    outp(PORT956, 0x66);
    *pbin=208;
    break;
  case 29:
    outp(PORT888, 0x6e);
    outp(PORT956, 0x6e);
    *pbin=210;
    break;
  case 30:
    outp(PORT888, 0x56);
    outp(PORT956, 0x56);
    *pbin=224;
    break;
  case 31:
    outp(PORT888, 0x5e);
    outp(PORT956, 0x5e);
    *pbin=226;
    break;
  case 32:
    outp(PORT888, 0x46);
    outp(PORT956, 0x46);
    *pbin=228;
    break;
  default:
    printf("Waveguide does not exist or is disconnected!");
    abort;
    break;
}
 return;
}

void time_delay(float t)
{
    struct dostime_t timed;
    float initial, value, elapsed;
    _dos_gettime(&timed);
    initial=(float)timed.hsecond/100+(float)timed.second
            +(float)timed.minute*60+(float)timed.hour*3600;
    elapsed=0;
    while (elapsed < t)
            {
            _dos_gettime(&timed);
            value=(float)timed.hsecond/100+(float)timed.second
            +(float)timed.minute*60+(float)timed.hour*3600;
            elapsed=value-initial;
```

```
        }

    return;
}

void wav_save(int waveguide)
{
    FILE *pfoutfile;
    pfoutfile=fopen("what.wav", "wb");
    putw(waveguide, pfoutfile);
    fclose(pfoutfile);
    return;
}
```

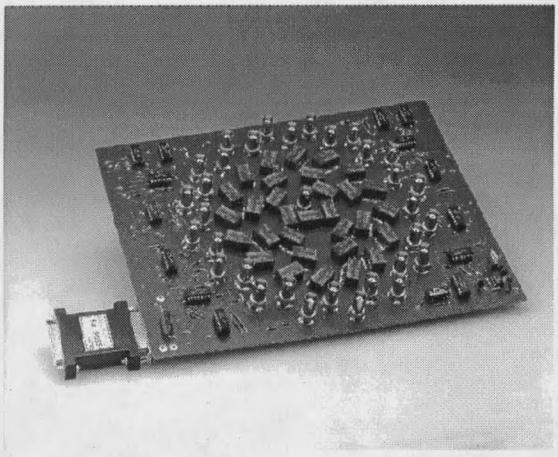
APPENDIX F

SALES BROCHURE

General Technologies, Inc.™

P.O. Box XXX, Portland, OR 97929
Phone (XXX) XXX-XXXX FAX (XXX) XXX-XXXX

M2980 TDR MULTIPLEXER



The TDR (Time Domain Reflectometer) Multiplexer is a computer controlled switch between your TDR instrument and one of 32 BNC coaxial outputs. This instrument was originally designed to accommodate soil dielectric measurements.

The software included with the TDR Multiplexer is similar to other TSR programs. Once installed, the program can be activated by pressing two "hotkeys". Program supports both mouse and keyboard entry. Additional software available in "BASIC" subroutines.

Switching TDR output is made simple so activation of a channel is made with ease.

Compatible with the Tektronix/Campbell Scientific Multiplexer software.

Utilization of a batch file with a list of selected instrument probes, time delays, and instrument triggers allows the user to make unattended automated measurements.

FEATURES:

- Input to any one of 32 outputs
- Reflection < 1%
- Match = 5 ohm ±1 ohm
- Insertion Power Losses < 1dB
- Bandwidth - 1.2 GHz
- Dimensions: 12" x 8.5" x 1"
- Connection: 25 pin D-subminiature
- Software Included
- One Year Warranty

APPLICATIONS

- Utilized with the Tektronix 1502C/F503C Metallic TDR.
- Assist in measurement of electrical conductivity.
- Determining the moisture content of soil.

COST:

- $1800 for M2980 TDR MUX and Software

FOR ORDERING INFORMATION CALL:

1-800-555-1234

APPENDIX G

LAPLACE TRANSFORM TABLES

f(t)	F(s)
$\delta(t)$	1
$u(t)$	$\frac{1}{s}$
$tu(t)$	$\frac{1}{s^2}$
$e^{-\alpha t}u(t)$	$\frac{1}{s+\alpha}$
$te^{-\alpha t}ut$	$\frac{1}{(s+\alpha)^2}$
$\sin\omega u(t)$	$\frac{\omega}{s^2+\omega^2}$
$\cos\omega tu(t)$	$\frac{s}{s^2+\omega^2}$
$e^{-\alpha t}\sin\omega tu(t)$	$\frac{\omega}{(s+\alpha)^2+\omega^2}$
$e^{-\alpha t}\cos\omega tu(t)$	$\frac{s+\alpha}{(s+\alpha)^2+\omega^2}$

Table 6. Laplace Transform Table

APPENDIX H

DERIVATION FOR PROBE CONDUCTANCE

$J = \sigma E$; where J=current density, σ = conductance, and E the applied electric field.

where J also equals

$$J = \sigma E_\rho a_\rho + \sigma E_\phi a_\phi + \sigma E_z a_z$$

but the coaxial probe current only flows in the radial direction so;

$$J = \sigma E_\rho a_\rho$$

The current flow can be found from the current density (J) by integrating over the area.

$$ds_\rho = \rho \cdot d\phi \cdot dz$$

$$I = \iint J \cdot ds = \int_0^{2\pi} \int_0^L \sigma E_\rho \rho d\phi dz = \sigma E_\rho \rho \cdot 2\pi \cdot L$$

solving for the radial electric field

$$E_\rho = \frac{I}{\sigma\rho \cdot 2\pi L}$$

$$Vz\int_a^b E_\rho \cdot dL = \int_a^b \left(\frac{I}{\sigma\rho \cdot 2\pi L}\right) \cdot d\rho = \frac{I}{\sigma 2\pi L}\int_a^b \frac{d\rho}{\rho}$$

$$V = \frac{I}{\sigma 2\pi L} \ln\left(\frac{b}{a}\right)$$

Then from the definition of resistance

$$R_{PROBE} = \frac{V}{I} = \frac{\ln\left(\frac{b}{a}\right)}{\sigma 2\pi L}$$

The probe conductance can be measured from the resistance by

$$\sigma = \frac{\ln\left(\frac{b}{a}\right)}{2\pi L R_{PROBE}}$$

APPENDIX I

CURVE FIT ANALYSIS

From equation (60) the corrected conductivity was derived to be

$$\sigma comp_t = \frac{\varepsilon_0 c}{L} \cdot \frac{Z_0}{Z_\mu} \left[\frac{2(1+\rho_m)(1-\rho_m)f_m^2}{\left(\frac{V_F'}{V_0} - \rho_m f_m\right)} - (1+2\rho_m f_m) \right]$$

The best curve fit is given by [7]

$$error = \sum_{i=1}^{N} (\sigma comp_t - \sigma_i)$$

To find the best fit curve for ρ_m and f_m

$$0 = \frac{\partial(error)}{\partial \rho_m} = \frac{\partial(error)}{\partial \sigma comp_t} \cdot \frac{\partial(\sigma comp_t)}{\partial(\rho_m)} = 0$$

$$0 = \frac{\partial(error)}{\partial f_m} = \frac{\partial(error)}{\partial \sigma comp_t} \cdot \frac{\partial(\sigma comp_t)}{\partial(f_m)} = 0$$

so

$$0 = \sum_{i=1}^{N} \left[\frac{\varepsilon_0 c}{L} \cdot \frac{Z_0}{Z_\mu} \left(\frac{2(1+\rho_m)(1-\rho_m)f_m^2}{\left(\frac{V_F'}{V_0}\right)_i - \rho_m f_m} - (1+2\rho_m f_m) \right) - \sigma_i \right] \cdot \left[-\frac{4\rho_m f_m^2}{\left(\frac{V_F'}{V_0}\right)_i - \rho_m f_m} + \frac{4(1+\rho_m)(1-\rho_m)f_m^3}{\left(\left(\frac{V_F'}{V_0}\right)_i - \rho_m f_m\right)^2} - 2f_m \right]$$

and

$$0=\sum_{i=1}^{N}\left[\frac{\varepsilon_0 c}{L}\cdot\frac{Z_0}{Z_\mu}\left(\frac{2(1+\rho_m)(1-\rho_m)f_m^2}{\left(\frac{V_F'}{V_0}\right)_i-\rho_m f_m}-(1+2\rho_m f_m)\right)-\sigma_i\right]\cdot\left[-\frac{4(1+\rho_m)(1-\rho_m)f_m}{\left(\frac{V_F'}{V_0}\right)_i-\rho_m f_m}+\frac{2(1+\rho_m)(1-\rho_m)f_m^2\rho_m}{\left(\left(\frac{V_F'}{V_0}\right)_i-\rho_m f_m\right)^2}-2\rho_m\right]$$

Notice that the two formulas expand out by the summation of (*i*) the measured case values. The previous two equations were placed in an equation solver (Chico Solver) and ρ_m and f_m were solved as the best fit from the experimental values.