### Sensitivity Analysis and Evaluation of the LOADRATE Load Zone Procedure for Implementation by Texas Department of Transportation

#### Abstract

A previous research study developed the LOADRATE program to replace current procedures that required three different computer programs for analyzing load-zoned roads in Texas. The Texas Department of Transportation was not satisfied with the results of the LOADRATE analyses of load-zoned roads when compared with analyses using current procedures. Differences between the two methods resulted in the need for an independent analysis of the LOADRATE program and procedures. The purpose of this study was to evaluate the code and analysis procedures of LOADRATE and make recommendations as to whether the Department should abandon the current analysis procedures and implement the LOADRATE program. This report summarizes the findings of a study of the LOADRATE program code and analysis procedures.

The study conducted parallel evaluations of several load-zoned roads using both the LOADRATE and current TxDOT procedures. The results verified previous TxDOT findings that indicated that LOADRATE was very liberal in lifting load restrictions on roads that current analyses showed should remain restricted in gross vehicle weight. This study reviewed the coding of the LOADRATE and LDATA programs and made many corrections and improvements to the existing code. The study also conducted sensitivity analyses of the variables used as input to the programs and compared the results of back-calculation of base and subgrade modulus with the Modulus computer program. The study specifically reviewed the method of predicting rutting using traffic and FWD deflections in the LOADRATE program.

The results of the study found significant differences in the results of analyses with LOADRATE versus current procedures that could not be corrected in this limited study. The rutting model unique to this program appears to be the primary cause for the vast difference in the analyses of load-zoned roads. There are significant differences in the calculation of base and subgrade modulus, but this error should have resulted in LOADRATE being more conservative in its results. This study recommended that LOADRATE procedures and computer program not be implemented by TxDOT for statewide analysis of load-zoned roads.

#### Key Words

Load-zoned roads, analyses, procedures, load restrictions, rutting, subgrade modulus, computer program, code, variables, sensitivity

#### Distribution Statement

No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.
SENSITIVITY ANALYSIS AND EVALUATION OF THE LOADRATE LOAD ZONE PROCEDURE FOR IMPLEMENTATION BY TEXAS DEPARTMENT OF TRANSPORTATION

by

Michael T. McNerney
Dennis S. Collier

Research Report 1944-1F

Research Project 3-18-92-1944

Improved Performance and Operation of the LOADRATE Load Zone Evaluation Procedure

conducted for the

Texas Department of Transportation

by the

CENTER FOR TRANSPORTATION RESEARCH
Bureau of Engineering Research
THE UNIVERSITY OF TEXAS AT AUSTIN

January 1993
IMPLEMENTATION STATEMENT

The purpose of this study is to ascertain whether the LOADRATE program is a reasonable method for determining the load-zoning of Farm-to-Market roads in Texas. The study provides a fair and rational basis for evaluating the usability of the LOADRATE program. Based on the results of this study, there are serious problems. The authors recommend that LOADRATE not be implemented.

Prepared in cooperation with the Texas Department of Transportation.

DISCLAIMERS

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Texas Department of Transportation. This report does not constitute a standard, specification, or regulation.

There was no invention or discovery conceived or first actually reduced to practice in the course of or under this contract, including any art, method, process, machine, manufacture, design or composition of matter, or any new and useful improvement thereof, or any variety of plant which is or may be patentable under the patent laws of the United States of America or any foreign country.

NOT INTENDED FOR CONSTRUCTION, PERMIT, OR BIDDING PURPOSES

Michael T. Mc Nerney, P. E. (Texas No. 70176)
Dennis S. Collier

Study Supervisors
PREFACE

The authors wish to thank the Texas Department of Transportation Pavement Management Section, who sponsored this research and provided data and assistance in the analysis of and procedures for load zoning techniques. The authors are also grateful for the assistance of Paul DiGiovanni, specifically for his work on the computer coding and for running the many computer simulations required for this analysis.
## METRIC (SI*) CONVERSION FACTORS

### APPROXIMATE CONVERSIONS TO SI UNITS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>When You Know</th>
<th>Multiply by</th>
<th>To Find</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LENGTH</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>in</td>
<td>inches</td>
<td>2.54</td>
<td>centimeters</td>
<td>cm</td>
</tr>
<tr>
<td>ft</td>
<td>feet</td>
<td>0.3048</td>
<td>meters</td>
<td>m</td>
</tr>
<tr>
<td>yd</td>
<td>yards</td>
<td>0.914</td>
<td>meters</td>
<td>m</td>
</tr>
<tr>
<td>mi</td>
<td>miles</td>
<td>1.61</td>
<td>kilometers</td>
<td>km</td>
</tr>
<tr>
<td><strong>AREA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>in²</td>
<td>square inches</td>
<td>645.2</td>
<td>millimeters squared</td>
<td>mm²</td>
</tr>
<tr>
<td>ft²</td>
<td>square feet</td>
<td>0.0929</td>
<td>meters squared</td>
<td>m²</td>
</tr>
<tr>
<td>yd²</td>
<td>square yards</td>
<td>0.836</td>
<td>meters squared</td>
<td>m²</td>
</tr>
<tr>
<td>mi²</td>
<td>square miles</td>
<td>2.59</td>
<td>kilometers squared</td>
<td>km²</td>
</tr>
<tr>
<td>ac</td>
<td>acres</td>
<td>0.395</td>
<td>hectares</td>
<td>ha</td>
</tr>
<tr>
<td><strong>MASS (weight)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>oz</td>
<td>ounces</td>
<td>28.35</td>
<td>grams</td>
<td>g</td>
</tr>
<tr>
<td>lb</td>
<td>pounds</td>
<td>0.454</td>
<td>kilograms</td>
<td>kg</td>
</tr>
<tr>
<td>T</td>
<td>short tons (2,000 lb)</td>
<td>0.907</td>
<td>megagrams</td>
<td>Mg</td>
</tr>
<tr>
<td><strong>VOLUME</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fl oz</td>
<td>fluid ounces</td>
<td>29.57</td>
<td>milliliters</td>
<td>mL</td>
</tr>
<tr>
<td>gal</td>
<td>gallons</td>
<td>3.785</td>
<td>liters</td>
<td>L</td>
</tr>
<tr>
<td>ft³</td>
<td>cubic feet</td>
<td>0.0328</td>
<td>meters cubed</td>
<td>m³</td>
</tr>
<tr>
<td>yd³</td>
<td>cubic yards</td>
<td>0.0765</td>
<td>meters cubed</td>
<td>m³</td>
</tr>
</tbody>
</table>

### APPROXIMATE CONVERSIONS FROM SI UNITS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>When You Know</th>
<th>Multiply by</th>
<th>To Find</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LENGTH</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mm</td>
<td>millimeters</td>
<td>0.039</td>
<td>inches</td>
<td>in</td>
</tr>
<tr>
<td>m</td>
<td>meters</td>
<td>3.28</td>
<td>feet</td>
<td>ft</td>
</tr>
<tr>
<td>m</td>
<td>meters</td>
<td>1.09</td>
<td>yards</td>
<td>yd</td>
</tr>
<tr>
<td>km</td>
<td>kilometers</td>
<td>0.621</td>
<td>miles</td>
<td>mi</td>
</tr>
<tr>
<td><strong>AREA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>in²</td>
<td>square inches</td>
<td>0.0016</td>
<td>square feet</td>
<td>ft²</td>
</tr>
<tr>
<td>m²</td>
<td>meters squared</td>
<td>10.764</td>
<td>square yards</td>
<td>yd²</td>
</tr>
<tr>
<td>km²</td>
<td>kilometers squared</td>
<td>0.39</td>
<td>square miles</td>
<td>mi²</td>
</tr>
<tr>
<td>ha</td>
<td>hectares (16,000 m²)</td>
<td>2.53</td>
<td></td>
<td>ha</td>
</tr>
<tr>
<td><strong>MASS (weight)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>g</td>
<td>grams</td>
<td>0.0353</td>
<td>ounces</td>
<td>oz</td>
</tr>
<tr>
<td>kg</td>
<td>kilograms</td>
<td>2.205</td>
<td>pounds</td>
<td>lb</td>
</tr>
<tr>
<td>Mg</td>
<td>megagrams (1,000 kg)</td>
<td>1.103</td>
<td></td>
<td>Mg</td>
</tr>
<tr>
<td><strong>VOLUME</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mL</td>
<td>milliliters</td>
<td>0.034</td>
<td>fluid ounces</td>
<td>fl oz</td>
</tr>
<tr>
<td>L</td>
<td>liters</td>
<td>6.264</td>
<td>gallons</td>
<td>gal</td>
</tr>
<tr>
<td>m³</td>
<td>meters cubed</td>
<td>35.315</td>
<td>cubic feet</td>
<td>ft³</td>
</tr>
<tr>
<td>m³</td>
<td>meters cubed</td>
<td>1.308</td>
<td>cubic yards</td>
<td>yd³</td>
</tr>
</tbody>
</table>

### TEMPERATURE (exact)

<table>
<thead>
<tr>
<th>°C</th>
<th>Celsius temperature</th>
<th>°F</th>
<th>Fahrenheit temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>5/9</td>
<td>(after subtracting 32)</td>
<td>9/5 (then add 32)</td>
<td>32</td>
</tr>
</tbody>
</table>

* SI is the symbol for the International System of Measurements.

NOTE: Volumes greater than 1,000 L shall be shown in m³.
TABLE OF CONTENTS

IMPLEMENTATION STATEMENT ................................................................. iii
DISCLAIMERS .................................................................................................. iii
PREFACE .......................................................................................................... iv
SUMMARY .......................................................................................................... ix

CHAPTER 1. INTRODUCTION
PROJECT SCOPE .............................................................................................. 1
RESEARCH APPROACH .................................................................................... 1

CHAPTER 2. BACKGROUND
THE CURRENT LOAD-ZONED ROAD ANALYSIS ............................................. 3
LOADRATE AND LDATA PROGRAMS .............................................................. 3

CHAPTER 3. CODING COMMENTS
LDATA .................................................................................................................. 5
PROGRAMMING DEFICIENCIES IN LOADRATE ............................................. 5
VARIABLES AND ARRAYS ............................................................................... 6
CODE .................................................................................................................. 6
DOCUMENTATION ............................................................................................ 8

CHAPTER 4. PROGRAM PERFORMANCE
COMPARISON OF LDATA AND MODULUS PROGRAMS ................................. 9
MOISTURE CORRECTION ............................................................................... 14
EVALUATION OF THE RUTTING MODEL ..................................................... 14

CHAPTER 5. CONCLUSIONS AND RECOMMENDATIONS
FACTORS AFFECTING PAVEMENT LIFE ....................................................... 19
PERFORMANCE OF LOADRATE ................................................................. 19
RECOMMENDATIONS ...................................................................................... 20

REFERENCES .................................................................................................. 21

APPENDIX A ..................................................................................................... 23
SUMMARY

The study conducted parallel analyses of several load-zoned roads using the LOADRATE program and current TxDOT procedures. The findings verified a recent paper by members of the TxDOT pavement management section that the majority of the roads released from load-zoning by LOADRATE would continue to be restricted if analyzed by current procedures. The findings of the study were that the LOADRATE program gave significantly lower modulus values with higher variability than the Modulus program currently used by the Department. Even with the lower modulus, the LOADRATE program permitted significantly greater vehicle loadings than would the current procedures. No major errors in coding were discovered, but the coding was poorly written and several improvements were made in the revised code. The difference in results between the LOADRATE program and the current procedures is primarily contained in the analysis portion of the program that uses a rutting prediction model. Validation of the rutting prediction model was not found to be adequate to justify using this new procedure. This study recommends that the LOADRATE program not be implemented by Texas Department of Transportation for evaluation of load-zoned roads. Implementation of the LOADRATE program without more detailed analysis and field testing of the rutting model could lead to premature failure of 75-90 percent of the 17,000 lane-miles of load-zoned roads.
CHAPTER 1. INTRODUCTION

The Texas Department of Transportation (TxDOT) is responsible for one of the nation's largest road systems with over 78,000 center-line miles of state-maintained highway network. Of this network, approximately 17,250 miles (22 percent) are light-duty pavement that are load-zoned for less than the normal legal limit truck traffic of 80,000 pounds gross vehicle weight and 20,000 pounds per single axle. Load-zoning is accomplished by limiting the maximum gross vehicle weight to 58,420 pounds, which was the weight limitation when many of these older light-duty pavements were first constructed.

The load-zoned roads are typically Farm-to-Market (FM) roads which have very thin-wearing surfaces of asphalt over a compacted base material of various thickness. Some roads may have stabilized based materials.

Frequently, the Pavement Management section of D-18 (now D-8) is requested to review the load-zoning of certain sections of the system to determine if the lifting of the load restrictions is feasible and appropriate. TxDOT sponsored research Project 473 with the Texas Transportation Institute (TTI) to develop a computer program and procedure for an improved method of analyzing load-zoned roadways using nondestructive testing (Ref 1). The results of that research project were the LDATA and LOADRATE computer programs.

When D-18 attempted to use the computer programs, thereby replacing the current procedures, the results were not as expected. In a test of 40 load-zoned road sections in Ellis County, the Department found that the computer programs predicted no rutting problems on 39 out of 40 road sections and recommended lifting load restrictions (Ref 2). In sharp contrast, the current method, though more time-consuming and cumbersome, predicted that the load-zoning restrictions could be lifted on only four of the 40 sections.

Clearly, since the results of these two procedures are at such extreme odds with one another, the Department could not blindly follow the LOADRATE computer program and evaluation procedure without additional information. The risk of premature failure on a majority of the load-zoned roads was too great a risk to take without further exploration of this computer program and the procedures used to evaluate it. The Department sponsored a small research effort with the Center for Transportation Research to evaluate and correct any errors in the LDATA and LOADRATE computer programs so that the Department might use them for evaluating roads for load-zoning.

PROJECT SCOPE

This study (Research Project 1944) was limited in time and budget and was not an effort to replace LOADRATE or write totally new procedures. The objectives of the project were to: (1) correct errors and omissions in the coding and execution of the LDATA and LOADRATE microcomputer programs, and (2) validate the operation of the programs to perform their intended functions as a reasonable tool for analyzing nondestructive test data for farm-to-market roads to permit evaluation of allowable loading.

RESEARCH APPROACH

To achieve the objectives of this study, the research effort was divided into three tasks. The first was to review the coding
and execution of the programs. LDATA and LOADRATE were searched for coding errors and extraneous and/or unnecessary code was eliminated. Each module was tested to see if it accomplished what was expected. In some instances, modules were broken out separately and run independently as separate programs to determine sensitivity to input variables and effects on the analyses.

The second task was to validate the operation of the programs as a tool for analyzing load-zoned roads. The LDATA program, which prepares the falling-weight deflectometer (FWD) data and traffic data for the LOADRATE program, was evaluated. It was compared to current mechanistic analysis method results and to the TTI Modulus program, which is more widely accepted for back-calculation of modulus of elasticity (Ref 3). It was believed that the LDATA program and LOADRATE programs would be sensitive to field conditions such as moisture, temperature, and seasonal variation of FWD data. The LOADRATE program was analyzed in terms of the way it calculates the allowable repetitions of legal load until rutting failure and the sensitivity of the model to the input variables.

If the LOADRATE program was found to be suitable for evaluating load-zoned roads, then the documentation of the program and procedures were to be improved as the third task. The Department found that the program was not very user-friendly nor well-documented. The documentation of the program was improved within the program code, but no additional manuals were developed. A report documenting the findings of the research results was also prepared.
CHAPTER 2. BACKGROUND

THE CURRENT LOAD-ZONED ROAD ANALYSIS

The original method for analysis of load-zoned roads was the Texas Triaxial Method. It is similar to the procedure by which many of the roads were designed. Over the years an empirical chart has been developed that relates pavement thickness to allowable design wheel load. The Texas Triaxial test is a laboratory test of saturated soil specimens that determines resistance to shearing forces (Ref 4). The soil is classified from 1 to 6 with 1 being very good base material. In most cases, the Texas Triaxial test values are known from laboratory tests taken during construction. The saturated soil condition is conservative and represents a worst-case scenario in the field (usually found only during the springtime rainy season or more frequently in East Texas near coastal areas). The method has been demonstrated to produce good results, and the data are available or can be estimated from soil samples or soil maps.

With the addition of falling-weight deflectometer equipment, a mechanistic analysis has replaced the Texas Triaxial test as the primary method of load-zone analysis by the Department. This method of testing uses the FWD data and the Modulus program to back-calculate the modulus of elasticity of the base and subgrade layers. The program uses the estimated thickness of the base layer and the deflections from the FWD data. A modulus of the base and subgrade are back-calculated using layered elastic theory, usually neglecting the thickness of the thin asphalt layer. The Modulus program was developed by TTI and is widely used for this back-calculation procedure. Both the LDATA program and the Modulus program back-calculate the modulus of the base and subgrade from the same FWD data.

In the current evaluation procedure, the Department takes the moduli generated from the FWD data using the Modulus program and imports the data into a Lotus 1-2-3™ spreadsheet. The data are then analyzed with the computer program BISAR, which uses layered elastic theory to determine the stresses and strains in the base and subgrade layers from 18-KIP single-wheel loads. The stresses and strains are then used with Asphalt Institute procedures to determine a limiting wheel load to achieve a ten-year life. If the allowable wheel load is below 20,000 pounds per axle (10,000 pounds per half-axle), the road remains restricted in gross weight. The several different programs that must be used separately to achieve the final answer make the procedure somewhat cumbersome, but the method has proved reliable in the past. Using this current procedure as well as the Texas Triaxial method, the Department obtains a markedly different result than that from the LOADRATE program.

LOADRATE AND LDATA PROGRAMS

The LDATA program is written in FORTRAN and basically takes the FWD data and back-calculates a modulus of elasticity of the base and subgrade based upon inputs of soil type and assumed base thickness. The program also calculates the number of ESALs for a 20-year period based upon input parameters. The LOADRATE program is written in BASIC and runs on a PC-compatible computer. The LOADRATE program takes the output file from LDATA and uses a rutting prediction model to esti-
mate the number of ESAL applications necessary to reach a specified failure criterion. The normal TxDOT criterion is 0.5 inches of rutting.
CHAPTER 3. CODING COMMENTS

This chapter cites examples of coding problems, particularly with the LOADRATE program. The deficiencies of both programs are discussed from a coding perspective. This chapter is not intended to be a criticism of each program's ability to provide and/or predict load-zone restrictions.

LDATA

One minor problem in following the instructions of the LDATA program was noticed. In the step that calculates the 20-year life of traffic, three possible method options are coded. One optional method asks for the current traffic year and a growth rate. If that method is used, the program will calculate only 19 years of traffic if the instructions are followed exactly. The reason this problem is not significant is the way the Department usually analyzes each roadway. The traffic division provides traffic for the first year and 20th year ESAL rate. Therefore the Department does not use that particular option.

PROGRAMMING DEFICIENCIES IN LOADRATE

Variables
- Using variables that are never changed
- Assigning variables/arrays twice in two consecutive lines

Unimplemented code

Code optimization
- Moving code into sequential order instead of using GOTO/GOSUB routines
- Removing unnecessary and blank REM statements
- Making multiple comparisons within an IF...THEN statement explicit
- Combining related/short lines of code
- Specifying variable for NEXT statement
- Removing "; GOTO" statements within IF...THEN statements
- Breaking long/complex lines of code
- Restructuring lines of code for simplicity/explicitness
- Removing parentheses
- Using decimal equivalents instead of executing a division
- Removing GOTO statement referring to next line
- Removing lines of code that have been REM-ed out

Documentation
- No listing of variable descriptions
- Dividing the program into distinct sections

Besides the problems presented by the rutting module, the LOADRATE program is flawed in many other ways. For example, variables and arrays are incorrectly assigned or used, some sections of the code are never executed, very little documentation is provided within the program itself, and the program is not very efficient. Even though some of the comments made below refer to lines of code that are syntactically correct and probably have no effect on the results generated by the program, the code itself may not be the original intent of the programmer, nor may it be understood in the same way by other programmers.

Unless otherwise indicated, all comments made below apply to the original program listed in Appendix A. Excerpts from the code are denoted by bold and italics lettering (i.e., Code).
Variables and arrays throughout the LOADRATE program are often misused, mis-assigned, or simply never changed. Here are a few examples:

1) Using variables/arrays that are never assigned.
The variable TBC is never assigned within the program nor by the user; however, it is called several times.

   Line 131  IF TBC = 1 THEN ....
   Line 177  IF TBC = 0 THEN ....

Other examples are lines 178, 205, and 341.

2) Using variables/arrays that are never changed.
The variable DESIGN is referred to only twice within the program.

   Line 186  IF DESIGN = 1 THEN : GOTO 4340
   Line 342  DESIGN = 0

The variable NH is assigned only once but is called several times.

   Line 343  NH = 0
   Line 131  IF TBC = 1 THEN : BA(I) = NH
   Line 137  IF NH = 0 THEN : PPR = 0
   Lines 308-315  IF (EQ = 1) AND (NH <> 0) THEN : PRINT #1, NS(I) ....

Similarly, other variables have been identified, such as RUT1 (assigned in line 940), SX (line 631), and BEND (line 1110).

3) Assigning a single variable different values in two consecutive lines.

The array DE(#,#)

   Line 200  IF EQ = 0 THEN : DE(I, 7)
              = 3.38075 * D(I, 5) ^
                639462
   Line 201  IF EQ = 0 THEN : DE(I, 7)
              = 4.5688721# * D(I, 5) ^
                  .578444

CODE

The code, besides being written in BASIC, is not very efficient. Here are a few reasons why.

1) Since the program is relatively small, there is no reason to place the introductory material near the end of the code as a subroutine. This is the case with lines 461 - 486, the introduction to LOADRATE.  

2) Even though LOADRATE is relatively small, unnecessary and blank REM statements simply require more memory and more CPU time. If a line can be left out, then it should be left out. Some REM statements that describe variables or arrays should be included in the introductory material describing the program to other programmers.

3) Multiple comparison within an IF ... THEN statement should be made explicit (for the programmer and the computer).

   Line 444  IF (CHO4 <> 1 AND CHO4 <> 0) THEN ....

should read as

   IF (CHO4 <> 1) AND (CHO4 <> 0) THEN ....

4) Short or related lines should be combined. This not only reduces the size of the program, but can reduce confusion and the possibility of errors within the program.
Line 551  \( X = X(II) : Y = D(I, II) \)
Line 552  \( X = \log(X) : Y = \log(Y) \)

should read as

\[
X = \log(X(II)) : Y = \log(D(I,II))
\]

In this example, the original code is syntactically correct. However, as shown with the variable \( DE(#, #) \) above, this may become a problem and may not be what the original programmer wanted.

5) With multiple \( FOR...NEXT \) loops, especially with nested loops, it becomes necessary to specify the variable intended by the \( NEXT \) statement. This also makes reading the program easier.

Line 74  \( NEXT I \)
Line 75  \( NEXT \)

Line 75 should specify the variable \( NTT \) (just as line 74 specified the variable \( I \)).

6) In BASIC, it is not necessary in \( IF...THEN \) statements to specify \( GOTO \). This reduces the code by one less command.

Line 113  \( IF ITE > 1 THEN : GOTO 3010 \)

should read as

\( IF ITE > 1 THEN \ 3010 \)

8) LOADRATE has few, if any, error checking routines that should be used whenever user input is accepted. For example, when inputting the unit weight of a material, the programmer can expect that the unit weight of a material material will be higher than water (\( UW = 64.2 \) pcf) and less than three times water (concrete weighs approximately 135 pcf). This helps prevent errors in user inputs.

Line 814  \( INPUT \ "UNIT WEIGHT \ OF \ MATERIAL \ (pcf) \ : \ " , \ UW \)

The next line should be

\( IF (UW < 100) \ OR (UW > 170) \ THEN \{ \text{input \ UW \ again} \} \)

9) Some sections of code are not even implemented.

Since, in line 591, there is a \( GOTO \) statement and since line 592 does not have a line number (meaning that there is no other way of getting to it), line 592 is never reached.

Line 591  \( GOTO 7800 \)
Line 592  \( END \)

10) Nested \( IF...THEN \) statements should not be used.

Line 35  \( IF SLECT1 = 3 \ THEN : \ IF RUT1 <> 1 \ THEN : INIOPT ... \)

should read as

\( IF (SLECT1 = 3) \ AND \ (RUT1 <> 1) \ THEN \ INIOPT ... \)

Also, note the removal of the colon after the second \( THEN \) statement.
11) Some lines of the code simply do not make mathematical sense. For example, in BASIC LOG refers to the base 10 of the logarithmic function. So, \[ LOG(10) \] equals 1.

Line 853 \[ THETA = 10^\left(\frac{\log(EI / K1)}{\log(10)} / K2\right) \]

Other times the mathematics performs no function at all. For example, the equation below first divides the negative of \( RHO \) by \( ZZ1 \). Then it raises the result to the first power (any number raised to the first power equals that number), and then it multiplies that result by a negative sign, which only negates the negative of \( RHO \). For example, consider the next line of code.

Line 509 \[ ER1 = ALPHA \times \exp\left(-\left(-\frac{RHO}{ZZ1}\right)^1\right) + 5.9 \]

should read as

\[ ER2 = ALPHA \times \exp\left(RHO / ZZ2\right) + 5.9 \]

**DOCUMENTATION**

Most programs written in BASIC are intended to be read by other programmers. Without documentation within the program, reading the program becomes difficult. There are two simple methods that every programmer should use to document a program. The first method is to give a list of variables used in the program and the purpose behind each variable. The second method is to indicate where the program performs different functions by breaking the program up into modules.
CHAPTER 4. PROGRAM PERFORMANCE

COMPARISON OF LDATA AND MODULUS PROGRAMS

In order to determine why there was such a difference in performance between the LOADRATE procedure and the current procedures, several of the individual modules were tested separately. The first module tested was the LDATA program.

This program takes the FWD data file and calculates the modulus of elasticity of the base and subgrade. Input parameters include the estimated thickness of the base layer and the type of subgrade soil. A sensitivity analysis was made of the effects of assumed base thickness on calculated modulus. Normally the thickness of the base layer is known from construction records, but if it is not, then this input parameter has an effect on the base and subgrade modulus calculated by the program. Figure 1 shows the effect of the LDATA program on calculation of base and subgrade modulus based upon an assumed thickness of the base layer. In this particular example, the actual base layer thickness was 11.5 inches. As was expected, the assumed base thickness has a moderate and nearly linear effect on the calculated subgrade modulus but a greater effect on the calculated base modulus. In this example the minimum value of the base modulus is near the actual base layer thickness.

Figure 2 shows the effect of assumed base thickness on the number of ESALs to achieve 0.5-inch rutting for all three different soil types used in the program. As Figure 2 shows, the effect of soil type can have a larger impact on the number of ESALs until rutting failure in the LOADRATE model than assumed base thickness.

The LDATA program was also compared to the Modulus program, which is the primary FWD data-reduction program used by the Department. Several FM roads in Ellis County were selected as comparison cases. Generally LDATA back-calculates a lower modulus for both the subgrade and base materials using the same FWD and input thickness and soil type. As a typical example, Figures 3 and 4 show that in the weaker sections of the roadway of FM 983, the moduli calculated by LOADRATE and Modulus can differ by a factor of two.

Using FM 2377 as another example, Figure 5 for base material and Figure 6 for subgrade material show that the LOADRATE program generally calculates a lower modulus than the Modulus program and has a higher variation as well. What is most surprising in the results of the FM 2377 data is that the LDATA program shows a large section of pavement that has a higher back-calculated modulus in the
subgrade layer than in the base layer (the shaded area in Figure 7).

The ratio of the modulus of the base layer to the subgrade layer can be very important in predicting pavement behavior and performance. For good performance of base materials, it is expected that the ratio of base to subgrade stiffness would be 4.
to 1. The comparison for FM 2377 of the stiffness ratio of base modulus to subgrade modulus for the two back-calculation programs is shown in Figure 8. Notice the significant area where the stiffness ratio is less than one. The means of the stiffness ratios
Figure 6. Comparison of Calculated Subgrade Moduli for FM 2377.

Figure 7. Moduli Calculated by LOADRATE for FM 2377.
are very nearly the same: 2.63 for the *Modulus* program and 2.24 for the LOADRATE program. It is the difference in variation that is most interesting. The standard deviation of the ratio for the *Modulus* program was 0.87, but it was 1.46 for the LOADRATE program.

The data of the computed stiffness ratios of FM 2377 shown in Figure 8 were statistically analyzed, and a frequency distribution analysis of those data is shown in Figure 9. From the histograms of the frequency distributions, one can see that the frequency distribution of the computed ratio of relative stiffness very nicely approximates a normal distribution for the *Modulus* program. However, the frequency distribution of the LOADRATE-computed ratio of relative stiffness is
skewed right and does not approximate a normal distribution pattern.

The distribution pattern of the LOADRATE data could be looked at as bimodal, but what is important is the significant difference in relative stiffness that these two back-calculation programs yield from analysis of the same FWD data. From the frequency distributions it is evident that the base-to-subgrade modulus ratio is better-defined and has less variation in the Modulus back-calculation than in the LDATA program. From a brief literature review, it appears that the LDATA program was tested to a limited extent with laboratory evaluation during its development, but it was not compared to the Modulus program (Ref 5).

The Modulus program has received much wider acceptance than LDATA and has less variation in the back-calculated modulus and modulus ratio. Since the LDATA program generally calculates a lower modulus of elasticity for both the base and subgrade, it would be logical to assume that the LOADRATE program would predict more rutting and more load-zone restrictions than the Department's method of evaluation using the Modulus program and layered elastic theory. However, as was demonstrated earlier, the reverse is true; LOADRATE predicts less rutting and nearly no load-zone restrictions than the Department's method of evaluation using the Modulus program and layered elastic theory. The differences in modulus calculated by LOADRATE and Modulus are sufficiently different as to cause a difference in analysis.

MOISTURE CORRECTION

The LDATA program as written appears to have input screens that allow the user to correct the moisture and temperature parameters in calculating a final modulus. However, there is no code that actually applies any corrections to the FWD data or modulus output values based upon either temperature or moisture. This is unfortunate, as both the seasonal variation of moisture and the temperature of the asphalt surface can have a significant impact upon rutting.

Recent testing by Dr. Thomas White at Purdue University has shown that surface rutting increases significantly in thin asphalt overlays at elevated temperatures (unpublished data). Analysis reported by Marshal Thompson at the University of Illinois shows that the rutting variation during spring, when moisture contents are greatest, is one of the most statistically significant variables in rutting prediction (Ref 6). It is possible that nearly all subgrade rutting of a pavement can occur during the Spring when moisture contents are high, and strength and stiffness of the subgrade can be one-half to one-third the normal summertime values.

EVALUATION OF THE RUTTING MODEL

The study had neither the time nor the budget to do a detailed analysis of the rutting model used in LOADRATE. However, the module was removed from the program and run with various base and subgrade moduli and three different soil types. Figures 10, 11, and 12 give the results of the sensitivity analysis.

The sensitivity analysis shows that there is little effect on predicting rutting by the base modulus. One would disagree with this premise unless all the rutting is expected to take place in the subgrade and not in the base or surface treatment. The three figures also show that any subgrade modulus above 10,000 psi will not result in a 0.5-inch rut for the 20-year traffic on FM 2377. Above 10,000 psi subgrade modulus, there is barely any difference in the type of soil. Only with subgrades at 5,000 psi and below is there a difference in soil types and excessive rutting in the 20-year traffic.

A look at the coding of the rutting model and a check of Research Report 473-1 reveals that the data used in the LOADRATE rutting model are computer-generated (Ref 5). The empirical data from
which the computer-generated data are derived may not be applicable to much of the real world scenarios found on load-zoned roads. More information is needed on the rutting model to determine its applicability to predicting performance of currently load-zoned roads. Using regression analysis of field conditions to predict rutting should be limited to very similar conditions of traffic loading, stress conditions, soil type, base material, surface material, and thickness and environmental conditions. There are so many variables that affect rutting that regression analysis of the aggregated variables will not provide predictive results.

Figure 10. LOADRATE Sensitivity of Rut Depth to Modulus of Elasticity for Silty Clay for FM 2377.
Figure 11. LOADRATE Sensitivity of Rut Depth to Modulus of Elasticity for Clayey Sand for FM 2377
Figure 12. LOADRATE Sensitivity of Rut Depth to Modulus of Elasticity for Heavy Clay for FM 2377
CHAPTER 5. CONCLUSIONS AND RECOMMENDATIONS

FACTORS AFFECTING PAVEMENT LIFE

There are many factors that influence the longevity of low-volume roads. The principal factors affecting the distress and ultimately the performance of asphalt-surfaced pavements are rutting, fatigue, and low-temperature cracking. Of these factors both fatigue and rutting are load-related. If one is to determine whether a particular section of roadway can withstand a certain level of traffic without restriction, several factors should be considered.

The Strategic Highway Research Program (SHRP) has proved that there was a significant amount of knowledge we simply did not possess about asphalt binders and their effects on asphalt concrete pavement performance. The variations in asphalt concrete performance with the current viscosity- or penetration-graded binder specifications is quite significant. However, on low-volume roads with very thin layers of asphalt as a wearing course, the asphalt characteristics may be minor except during high- or low-temperature conditions.

The properties of the aggregate are highly significant in the strength of the asphalt concrete. However, in the typical FM-type road construction, the strength of the asphalt is not the major factor if the asphalt meets minimum State specifications. The quality of the aggregate in the base material can be very significant and can lend considerable strength to the pavement system.

The effect of stress and strain on loading is very important. When traffic volumes are light, a few instances of heavy loads that produce high stress and strains could cause more damage than 20 years of normal loading. Truck loadings are highly variable. Typical weigh-in-motion (WIM) sites report large variations in truck loading with significant amounts of traffic above legal loadings. Some WIM sites have reported as high as 20-30 percent overloads in truck traffic. It is possible that the rutting or fatigue damage in light-duty pavements caused by occasional very heavy loads can be much greater than that caused by normal traffic.

The effects of moisture on the subgrade, base, and pavement surface cannot be over-emphasized. It is widely known that high moisture conditions in the springtime environment result in highly damaged pavements. It has been suggested that the wear to some pavements during the three months of spring, accumulated over the pavement lifetime, could account for a very high percentage of all the load damage to the pavement. Some states have a load-zoning policy that rates roads only during the three months of spring. The theory is that soil support and strength will be at least two or three times weaker during the springtime, so that the greatest damage is likely during this period.

PERFORMANCE OF LOADRATE

The LOADRATE program does not do an adequate job of evaluating roads for removal from load-zoned status. The data available for load-zone evaluation are FWD deflections, soil type and/or lab analysis, base-type and thickness. The calculation of modulus values by LOADRATE is significantly lower than that by the Modulus program. The lower modulus could be intentional since there is no correction for season, moisture, or temperature of the FWD deflections, but there is no documentation which might indicate that this is the case. The large variation in base-to-subgrade modulus ratio calculated by
LOADRATE versus *Modulus* leaves one uncomfortable about the results of the LOADRATE-calculated moduli.

The coding of the program is poorly done and not well documented. The code was not written for clarity nor written to be user-friendly. Even with corrections made to the code, the program is not very efficient or useful. The LDATA portion was written in FORTRAN and is case-sensitive requiring all capital letters. The program also has very limited error trapping, requiring a restart of the program if certain common errors are made in the input screens.

The rutting model portion of the program provides results that are extremely different from those using current procedures. If there is any hope of using this program, the rutting model must be studied for its applicability for predicting rutting on low-volume roads. The effect of using FWD data gathered in the spring season versus the summer could provide significantly different results. It may be possible that LOADRATE could predict results similar to those obtained with current procedures if all FWD data were taken during springtime or adjustments were made to account for seasonal variations.

However, the bottom line is that the method of calculation of predicted rutting and allowable loading is empirically based on regression models and compared to computer-generated data. There is not sufficient documentation provided to the authors that this rutting is a better predictor than the current procedures. LOADRATE does not provide results that agree with current procedures or with the more conservative Texas Triaxial method. The implementation of the LOADRATE program and procedures could lead to the premature failure of 75-90 percent of the 17,000 lane-miles of load-zoned roads.

**RECOMMENDATIONS**

1. LOADRATE should not be adopted as replacement for current TxDOT procedures for evaluating load-zoned roads.
2. LOADRATE should not be used as a replacement for *Modulus* for back-calculating modulus of the base and subgrade from FWD data.
3. The LOADRATE program should not be revised for the purpose of evaluating load-zoned roads.
4. If TxDOT requires integrated computer programs to evaluate load-zoned roads, new programs should be developed using the *Modulus* program as a kernel together with current procedures and the present method of evaluation.
5. Research and field-testing are required to validate other possible subgrade-rutting models. Surface-rutting models should also be incorporated into any program used for load-zone analysis.
6. Monitoring the performance of surface- and subgrade-rutting models as well as the surface-cracking models over the various geographical locations in Texas should be performed and documented in a database with a geographical interface for calibration of these models as well as management of the load-zoned network.
REFERENCES


APPENDIX A

The following code reflects both the corrected listing for LOADRATE and the original code for the Texas Transportation Institute program. The sequential line numbers are for reference only. They are not part of either program.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51

'***************************$$ PROGRAM - LOADRATE - $$$$$***************************
*  Version AUGUST 1989 *
*  Load Rating of Light Pavement Structures *
*  WRITTEN BY K. M. CHUA *
*  REVISRED BY *
*  B. LANKA SANTHA *
*  Pavement Systems, Texas Transportation Institute *
*  TTI Building, Texas A&M University, *
*  College Station, Texas 77843 *
*  (409) 845-5982 *
*  Update #1 20th August 1989 *
*  Update and Revision 29th January 1993 *
*  At the Center for Transportation Research *
*  The University of Texas at Austin *
*  Austin, TX 78705-2650 *
*  (512) 472-8875 *
*  Update and Revision 29th January 1993 *

'******************************************************

'VARIABLE DESCRIPTION

'CK$(#) Rut depth (in.)
'AC(#) Base thickness (in.)
'AM(#) Deflection (mil) [index, sensor #]
'BA(#) 'BBA(#) 'BC(#) 'C(#) 'D(#,#) Modulus change due to suction(psi);temp.correc.data
'DE(##) Modulus change due to temp(psi);temp. correc. data
'DYR(#) Cumulative % sections
'ES(#) Base elastic modulus (psi)
'E2(#) Subgrade elastic modulus (psi)
'FSUM(#) 'FW(#) 'HI(#) 'JSUM(#) 'K1(#) 'K1S(#) Wanted cond. modulus (psi);temp. correction data:new E
'NS(#) Section number
NX(#) Remaining life (years)
NY(#) Input modulus (psi); temp. correction data; old E
OE(#) Allowable wheel passes
passes(#) Number of passes
PFW(#) FWD load (lbs)
PRESS(#) Rut depth caused [in.]
PS(#) Total change in E due to temp. & suction
R(#) Single wheel/ESWL (lbs) [truck#, axle#]
RFINAL(#)  
SI(#) (Poly. Interpol) Parameter values used in the data base
SL(#) Base rutting potential input parameter for data base
SS(#) Base resilient modulus input parameter for data base
SSO(#) Subgrade resilient modulus input parameter for data base
TOTDE(#) Subgrade rutting potential input parameter for data base
W(#) Base thickness input parameter for data base
WL(#) (Poly. Interpol) Rut depths from data base corresponding to parameter
WLOAD(#) 
WOS(#) 
WP(#) 
WW(#) 
Xi(#) Forward sensor spacing (in.); radial distance
XI(#) (Poly. Interpol) Rut depths from data base corresponding to parameter
XJ(#) 
XL(#) 
XM(#) 
Y1(#) 
YJTEMP(#) 
YKTEMP(#) 
YLTEMP(#)

"a first cycle strain (ABA, ACH, ACL, ASM)"
"aa Coeff. (ABA, ACH, ACL, ASM)"
"ABA A coeff for ABA
"AACH A coeff for ASG for MSG=1
"AACL A coeff for ASG for MSG=2
"AASM A coeff for ASG for MSG=3
"ab Exponent (ABBA, BCH, ACL, ASM)"
"ABA A coeff for base for XII
"ABBA B coeff for ABA
"ABCH B coeff for ASG for MSG=1
"ABCL B coeff for ASG for MSG=2
"ABS M B coeff for ASG for MSG=3
"AL1 Allowable legal axle load single
"AL2 Allowable legal axle load tandem
"AL3 Allowable legal axle load triple
"AL # of years for GR
"ALP Alpha: Linear thermal expansion
"AV Ave. number of passes to cause specified rut depth
"b Residual deformation rate (BA, BCH, BCL, BSM)"
BBABase course material;
B B coeff for base for XII
BBABSubgrade material;
B B coeff for subgrade for XLL
CH01-4Menu selection
COLPrinter column number
DELVDelta Volume; Change in volume
DTEABS(DTEMP)
DELMDelta(RTEMP-TTEMP)
E Elastic modulus of soil grains
EBAElastic resilient modulus - base course (psi)
EiElastic modulus
EQEQ=0:Dynaflec
EQEQ=1:Falling Weight Deflectometer (FWO)
EQQ# of equivalent standard wheel (9000 lbs) passes
ESAL1First year traffic; E equiv. standard axle load
ESG Elastic resilient modulus - subgrade (psi)
GRAnnual traffic growth rate
HIInput suction (psi) condition; temp. correction data
HLWanted suction (psi) condition; temp. correction data
I "Index, counter"
II Remaining life (years)
IDDistrict
INIOPTIteration counter
ITEModulus coeff
K1K2 is the power of the eq. E=K1*Theta^K2
K01SIGR/SIGZ ratio for base course
K02SIGR/SIGZ ratio for subgrade
LGABAlog a Base course material
LGASGlog a Subgrade material
MINXStarting section number
MINYMinimum "Y" value
MAXXEnding section number
MAXYMaximum "Y" value
MSGSubgrade Material type
MSG = 1 Heavy clay (CH)
MSG = 2 Light/silty clay, clayey silt (CL-ML)
MSG = 3 Clayey/silty/uniform sand (SC-SM)
MrResilient modulus (ksi) (EBA, ESG)
MITYpe of base material
NNumber of wheel passes (cycles)
N2Porosity
NTTTruck number
NUMNumber of levels of the parameter
NVNew value; temporary value storage
OPP OP=1: soil classification
OP=2: Lab data
OOOO=1: indicates all corrections have been done
PATotal number of passes during ESAL1 period
PNumber of passes used for rut depth calculation
PILpi=3.1415926536
PLHardcopy O=no 1=yes 2=results only
PR Type of base material
PRO1Property of material = 3(1-(poissons ratio)^2)/4E
PRI-10Printer control characters
PSChange in pressure due to change in suction
Printer type and associated printer characters
PT=0: none specified
PT=1: Epson/Panasonic
PT=2: Mannesmann Tally

Change in Hydrostatic pressure due to a change in temp.

Allowable rut depth (in.)
Existing rut depth (in.)
Recorded rut (in.)

Wanted temp. condition; Temp. correction data: Reference temp

Allowable rut (in.); Rut depth used for remaining life

and number of passes calculation

Soil specific gravity
Calculation selection
Selected format
Optional output selection
SLO=1: t$=Elastic mod. of base course
SLO=2: t$=Elastic mod. of subgrade
SLO=3: t$=Number of passes for rut depth
SLO=4: t$=Remaining life
Thickness of base layer (in.)
Rut temperature correction (TEMRUT=0: not corrected)

Bulk stress; Sum of the principal stresses; E=K(1)\*THETA^K(2)

Input temp condition; Temp. correction data
Poisson's ratio
Unit weight of material (pcf)
Void ratio
Which line to edit data
Interpolated value
Value of the parameter
Base rutting potential
Base resilient modulus
Subgrade resilient modulus
Subgrade rutting potential
Thickness of base layer
Elastic mod. for base course depth
Elastic mod. for subgrade depth
Job title
Job title - rut depth subroutine
County
Date
Road
Null; "Press return to continue"
Input file name
Output file name

Dimensioning

DIM D(160, 7), S0(160), DE(160, 7), AC(160), PFW(160), K1(160), W(160), BC(160), K1S(160)"
DIM WL(160, 8), AM(160), PS(160), SS(160), CK$(160), NS(160), H(160), S1(160), BA(160), NX(160)"
DIM SS0(160), BBA(160), WW(160), WP(160), WOS(160), E1(160), E2(160).
DIM DSUM(10), JSUMC10), FSUM(10), DF(10), FW(10), R(10), PRESS(10), WLOAD(10), x(10), DRY(160)

DIM det(160), des(160), NE(160), OE(160), TOTDE(160), RFINAL(160), passes(160), NY(160)

DIM XM(160), XL(160), XK(160), XJ(160), XI(160), Y1(5), XI(5), YLTEMP(5), YKTEMP(5)

MN

ALLOWABLE

INITIALIZE SELECTED VARIABLES

PT = 0
X(1) = .01: X(2) = 12: X(3) = 24: X(4) = 36
X(5) = 48: X(6) = 60: X(7) = 72
SG = 2.67
PI# = 3.1415926536#

MAIN MENU OF SELECTIONS

CLS: KEY OFF
LOCATE 3, 10
PRINT "LOAD RATE - STRUCTURAL ANALYSIS OF LIGHT PAVEMENTS"
LOCATE 5, 10
PRINT "MAIN MENU"
LOCATE 7, 10
PRINT "SELECT THE OPERATION DESIRED"
LOCATE 9, 10
PRINT "(1) Structural analysis of pavements"
LOCATE 10, 10
PRINT "(2) Temperature correction of base modulus only"
LOCATE 11, 10
PRINT "(3) Rut depth calculation only"
LOCATE 12, 10
PRINT "(4) Exit program"
LOCATE 13, 10
PRINT "moisture effects on base material"
LOCATE 14, 10
PRINT "Temperature correction of base modulus only"
LOCATE 15, 10
PRINT "Rut depth calculation only"
LOCATE 16, 10
PRINT "Exit program"
LOCATE 17, 10
PRINT "ENTER SELECTION : ": SLECT1
IF (SLECT1 < 1) OR (SLECT1 > 4) THEN 730
IF SLECT1 = 2 THEN CH01 = 0: GOTO 10010
IF SLECT1 = 3 THEN INIOPT = 2: GOSUB 12400
IF SLECT1 = 4 THEN CLS: END

FOR I = 1 TO 160
NY(I) = 0
DRY(I) = 0
PS(I) = 0
PRINT : PRINT
1790 PRINT "INPUT FILE NAME:"'
INPUT x$
1791 'PRINT x$
PRINT "Please Wait......"
OPEN ""I"", #1, x$
INPUT #1, EQ
INPUT #1, ID
INPUT #1, C$
INPUT #1, F$
INPUT #1, D$
INPUT #1, N, NC, NT, RX, RM, AL, PA, GR, ESAL1
FOR NTT = 1 TO NT
INPUT #1, NX(NTT)
FOR I = 1 TO NX(NTT)
INPUT #1, WL(NTT, I)
NEXT I
NEXT NTT
FOR I = 1 TO NC
IF EQ = 0 THEN INPUT #1, NS(I), BA(I), D(I, 1), D(I, 2), D(I, 3), D(I, 4), D(I, 5)
IF EQ = 1 THEN INPUT #1, NS(I), BA(I), D(I, 1), D(I, 2), D(I, 3)
IF EQ = 1 THEN INPUT #1, D(I, 4), D(I, 5), D(I, 6), D(I, 7), PFW(I)
NEXT I
CLOSE
' To print input card images
OPEN ""SCRN:"" FOR OUTPUT AS #1
FOR i te = 1 TO NT
IF i te > 1 THEN PRINT #: GOTO 2440"
CLS : PRINT #: "TEXAS TRANSPORTATION INSTITUTE"
PRINT #: "LOAD RATING OF LIGHT PAVEMENT"
PRINT #: "JOB : "; a$; " (INPUT FILE ""; x$; "")"
PRINT #: "DISTRICT:"; ID; " COUNTY:"; C$; " ROAD:"; F$
PRINT #: "ALLOWABLE RUT(INS):"; RX; " RECORDER RUT(INS):"; RM"
PRINT #: "press RETURN to continue": nn$"
PRINT #: "TRUCK NO. "; NTT"
PRINT #: "AXLE NUMBER SINGLE WHEEL/ESWL(LBS)"
'To loop of the number of axles NX and number of trucks NTT
INPUT "press return to continue(line2441)"; nn$
FOR I = 1 TO NX(NTT)
PRINT #: TAB(5); I; TAB(20); WL(NTT, I)
NEXT I
INPUT "Press RETURN to continue(line2442)"; nn$
IF i te > 1 THEN 3010
PRINT #: "ANNUAL TRAFFIC GROWTH RATE:"; GR; " # OF YEARS:"; AL:
FIRST YEAR TRAFFIC: ESAL1
PRINT #1, "TOTAL NUMBER OF PASSES DURING ABOVE PERIOD:" PA
PRINT #1, "DATE:" D$: " DYNAFLECT"
PRINT #1, "FALLING WEIGHT DEFLECTOMETER"
PRINT #1, "Press RETURN to continue"; nn$
PRINT #1, "SECTION BASE DEFLECTION"
PRINT #1, "DATE:" D$
PRINT #1, "FALLING WEIGHT"
PRINT #1, "INPUT "Press RETURN to continue"; nn$
PRINT #1, "SECTION BASE DEFLECTION" If EO = 0 THEN PRINT #1, "NO. THICKNESS (MILS)"
PRINT #1, "NO. THICKNESS (MILS)"
PRINT #1, "INPUT "Press RETURN to continue"; nn$
PRINT #1, "SECTION BASE DEFLECTION" If EO = 0 THEN PRINT #1, "NO. THICKNESS (MILS)"
PRINT #1, "NO. THICKNESS (MILS)"
FOR I = 1 TO NC
PRINT 0
FOR I = 1 TO NC
PRINT "IF I = 15 THEN LOCATE 25, 1: INPUT "Press RETURN to continue""; nn$
PRINT "IF TBC = 1 THEN BA(I) = 0"
IF BA(I) <> 0 THEN 2890
WW = (DE(I, 1) - DE(I, 7)) / DE(I, 7)
BA(I) = ((SSO(I) / 1958 * WW) ^ 1.85
BA(I) = INT((BA(I) * 10 + 5) / 10)
IF PPR = 2 THEN 3000
2890 IF EO = 0 THEN PRINT #1, NS(I); TAB(12); BA(I); TAB(23);
IF EO = 1 THEN PRINT #1, USING "###.##"; D(I, 1); D(I, 2); D(I, 3); D(I, 4); D(I, 5)
PRINT #1, "PsW(I)"
3000 NEXT I
3010 CLOSE
3080 LOCATE 25, 1: PRINT SPC(132);
3570 IF ite <> 1 THEN 3640
IF TBC = 0 THEN LOCATE 25, 1: INPUT "DO YOU WANT HARDCOPY? O=NO I=YES RESULTS ONLY"; PR
IF TBC = 1 THEN LOCATE 25, 1: INPUT "DO YOU WANT HARDCOPY? O=NO I=YES RESULTS ONLY"; PR
LOCATE 25, 1: PRINT "line after 3670(LINE3071)"
3670 IF ite <> 1 THEN 3640
IF TBC = 0 THEN LOCATE 25, 1: INPUT "DO YOU WANT HARDCOPY? O=NO I=YES RESULTS ONLY"; PR
3671 'PRINT "line after 3670(LINE3071)"
IF EO = 0 THEN PL = 1000
IF EO = 1 THEN PL = 10956.3
FOR I = 1 TO NC
IF EO = 1 THEN SO(I) = PFW(I) / D(I, 1); DE(I, 1) = PL / SO(I)"
29
367 "’D(I, 7) may not be taken at 94.5°’ then need to change"
368 " IF EQ = 1 THEN GOSUB 7290: DE(I, 7) = D7(I) * PL / PFW(I)"
369 ’D7(I) IS FROM SUBROUTINE GEOMBASIN
370 ’To convert Dynaflect readings to DE
371 ’"4000 IF EQ = 0 THEN DE(I, 7) = 3.38075 * D(I, 5) ^ .639462"
372 ’"4001 IF EQ = 0 THEN DE(I, 7) = 4.5688721# * D(I, 5) ^ .578444: DE(I, 1) =
373 "7.24474 + (29.6906 * D(I, 1))"
374 " IF EQ = 0 THEN DE(I, 7) = 07(1) * PI(I)
375 ’T0 convert Dynaflect readings to DE
376 IF TBC = 0 THEN SS(O(I)) = SO(I)
377 IF 07(1) <= 83.46183 THEN SO(I) = 86.0122 * EXP(.00187211 * PL(I)
378 H(I) = BA(I)
379 NEXT I
380 ’PRINT ’”Please Wait ......””
381 FOR I = 1 TO NC
382 x = BA(I): Y = DE(I, 7)
383 IF x > 18 THEN x = 18
384 K1S(I) = (955.651 + 370.3322 * x) * Y ^ (-1.21665 + .016349 * x)
385 IF K1S(I) < 5450 THEN K1S(I) = 5450
386 IF K1S(I) > 62200 THEN K1S(I) = 62200
387 LL1 = 4342945 * LOG(K1S(I))
388 A11 = 10 ^ (10.56968 - 1.97986 * LL1)
389 A12 = 361.506 - 2131.05 / (K1S(I) * 10 ^ -3) - 18.305
390 AL2 = 391 A11 * BA(I) + A12 * BA(I) * BA(I)
392 B11 = .032285 - .109 / (K1S(I) * 10 ^ -4)
393 B12 = .123403 - .03883 * LL1
394 B13 = -.00426 + .001256 * LL1
395 B1 = B11 + B12 * BA(I) + B13 * BA(I) * BA(I)
396 K1(I) = (DE(I, 1) / A1) ^ (1 / B1)
397 IF K1(I) <= 3000 THEN K1(I) = 3000
398 IF K1(I) > 60000 THEN K1(I) = 60000
399 NEXT I
400 ’=============================================================================
401 4790 ’Elastic modulus calculations ECAL
402 ’=============================================================================
403 AV = 0
404 K01 = .6: K02 = .82
405 AREA = PI# * 5.9 * 5.9
406 FOR I = 1 TO NC
407 IF EQ = 0 THEN PFW(I) = 9000
408 SL(I) = PFW(I) / AREA
410 ZZ1 = 1 + BA(I) / 2: ZZ2 = 7 + BA(I)
411 ALS1 = 2.416966 * SL(I) ^ .477234: ALS2 = .6007171 * K1S(I) ^ .058992
412 ALH = 1.338562 * (1 + BA(I)) ^ -.145887
413 RHB = .5961413# * K1(I) ^ 5.593462E02
414 RHH = 3.0385722# * (1 + BA(I)) ^ .419624
415 RHS = 272.63462# * K1S(I) ^ -.58427
416 NSG = 1.01866 + 1.239625 / (K1S(I) * 10 ^ -3) + .000013 * (K1S(I) * 10 ^ -3) ^ 3 + .061115 * (K1S(I) * 10 ^ -3) - .00165 * (K1S(I) * 10 ^ -3) ^ 2
417 ALPHA = ALS1 * ALS2 * ALH
418 RHO = (1 + BA(I)) * LOG(RHH * RHO * RHB / ALPHA) / LOG(2.7182818#)
419 ER1 = ALPHA * EXP((-(-RHO / ZZ1) ^ 1)) + 5.9: ER2 = ALPHA * EXP((-(-RHO / ZZ2) ^ 1)) + 5.9
SIGZ1 = PFW(I) / (PI# * ER1 * ER1): SIGZ2 = PFW(I) / (PI# * ER2 * ER2)
SIGR1 = K01 * SIGZ1: SIGR2 = K02 * SIGZ2
E2(I) = KIS(I) * (ABS(SIGZ2 - SIGR2)) ^ ASG: E1(I) = K1(I) * (SIGZ1 + 2 * SIGR1) ^ .33

NEXT I

'------------------------------------------------------------------------
'To calculate the rut depth and number of passes
'------------------------------------------------------------------------
INOPT = 3: GOSUB 12400
FOR I = 1 TO NC
PS(I) = passes(I)
AM(I) = RFINAL(I)
IF GR < 0 THEN 5097
NY(I) = (LOG(PS(I) * GR + ESAL1) - LOG(ESAL1)) / LOG(1 + GR)
TOTYR = TOTYR + NY(I)
IF NY(I) > NYMAX THEN NYMAX = NY(I)
NEXT I
IF GR < 0 THEN 5100
FOR II = 1 TO NYMAX
FOR I = 1 TO NC
IF NY(I) >= II THEN DYR(I) = DYRCII) + 1
NEXT I
DYRCII) = (OYRCII I NC I 100
IF DYRCII) = 100 THEN 
NEXT II
AL1 = ((S100 / 10) ^ .2432) * 18
AL2 = ((S100 / 10) ^ .2467) * 33.5
AL3 = ((S100 / 10) ^ .2534) * 48.25
IF AL1 > 18 THEN AL1 = 18
IF AL2 > 34 THEN AL2 = 34
IF AL3 > 46 THEN AL3 = 46
MINX = NS(1)
MAXX = NS(NC)

"6100 IF PR = 1 THEN OPEN ""LPT1:"" FOR RANDOM AS #1: CLS : PRINT
""Printing .....""

IF PR = 0 THEN OPEN ""SCRN:"" FOR OUTPUT AS #1: CLS

NO. OF RUT"
IF GR < 0 THEN PRINT #1, "" "" SECTION LAYER PROPERTIES

ALLOWABLE DEPTH"
IF GR < 0 THEN PRINT #1, "" "" NO. ELASTIC MODULUS

SUBGR
E1-BASE E2-SUBGR
PASSES (IN)"
IF (EO = 1) AND (GR < 0) THEN PRINT #1, "" BASE/SUBB

SUBGRADE
PASSES (IN)"
IF (EO = 0) AND (GR < 0) THEN PRINT #1, "" REMAINING NO. OF RUT"

LIFE ALLOWABLE DEPTH"
IF (EO = 0) AND (GR >= 0) THEN PRINT #1, "" E1-BASE

E2-SUBGR (YEARS) PASSES (IN)"
IF (EO = 1) AND (GR >= 0) THEN PRINT #1, "" BASE/SUBB

SUBGRADE (YEARS) PASSES (IN)"
FOR I = 1 TO NC
IF (EO = 1) THEN PRINT #1, TAB(3): NS(I); TAB(13): USING ""####": E1(I); TAB(25):
IF (EO = 1) THEN PRINT USING ""####": E2(1):

31
467 " IF (EQ = 1) AND (GR >= 0) THEN PRINT #1, TAB(39); USING "###.#"; NY(1):"
468 " IF (EQ = 1) THEN PRINT #1, TAB(51); USING "#############"; PS(I):"
469 " IF (EQ = 1) THEN PRINT #1, USING "#####.#"; AM(I)"
470 NEXT I
471 " PRINT #1,"
472 "5101 'INPUT "'press return to continue(line5101)'"': nn$"
473 “ PRINT #1,”
474 “ IF GR < 0 THEN 5740
475 “ PRINT #1,”
476 "PRINT #1, ""
477 " PRINT #1, "" SINGLE TANDEM
478 " PRINT #1,"
479 " IF S100 >= 10 THEN PRINT #1, "" LEGAL LEGAL
480 " IF S100 < 10 THEN PRINT #1, USING ""
481 " # # " AL1: AL2: AL3"
482 " PRINT #1,"
483 “ PRINT #1,”
484 “ PRINT #1, "" AVERAGE NUMBER OF Passes TO CAUSE SPECIFIED Rut (""; RX; ";") ; AV"
485 “ PRINT #1,"
486 “ PRINT #1, "" NUMBER OF Passes USED FOR Rut Depth CALCULATION : "; PA"
487 “ PRINT #1,”
488 “ PRINT #1, "" Rut Depth USED FOR REMAINING LIFE AND NUMBER OF Passes 
CALCULATION ; ": RX"
489 “5741 'INPUT "'press return to continue(line5741)'"': nn$"
490 NEXT ite
491 PR = 0
492 PRINT
493 CLOSE
494 IF TBC = 0 THEN 7580
495 GOTO 2290
496 "-----------------------------------------------
497 6510 ' SUBROUTINE to open an output file OUTFILE
498 "-----------------------------------------------
499 500 CLS : KEY OFF
500 “ LOCATE 5, 1: PRINT "" TO CREATE OUTPUT FILE""
501 “ LOCATE 6, 1: INPUT "" ENTER OUTPUT FILE NAME AND <RET>: ""; XX$"
502 “ CLS : KEY OFF
503 “ IF TEMRUT = 1 THEN SLF = 1: GOTO 6536
504 “ LOCATE 7, 1: PRINT "" Select the format:""
505 “ LOCATE 9, 1: PRINT ""(1):"
506 “ IF GR < 0 THEN LOCATE 10, 1: PRINT "" Sec. Thick E1(Base) E2(Subgr) Passes Rut depth """
507 “ IF GR >= 0 THEN LOCATE 10, 1: PRINT "" Sec. Thick E1(Base) E2(Subgr) Passes Rut depth Remaining life""
508 “ LOCATE 11, 1: PRINT ""(2):"
509 “ IF (EQ = 1) AND (GR < 0) THEN LOCATE 12, 1: PRINT "" Sect. Thick D1 E7 FWD load E1(Base) E2(Subgr) Passes ""
510 “ IF (EQ = 0) AND (GR < 0) THEN LOCATE 12, 1: PRINT "" Sect. Thick D1 E7 FWD load E1(Base) E2(Subgr) Passes ""
511 “ IF (EQ = 0) AND (GR >= 0) THEN LOCATE 12, 1: PRINT "" Sect. Thick D1 E5 FWD load E1(Base) E2(Subgr) Passes Remaining life""
512 “ IF (EQ = 1) AND (GR >= 0) THEN LOCATE 12, 1: PRINT "" Sect. Thick D1 E7 FWD load E1(Base) E2(Subgr) Passes Remaining life""
513 “ IF (EQ = 0) AND (GR >= 0) THEN LOCATE 12, 1: PRINT "" Sect. Thick D1
D5  FWD load  E1(Base)  E2(Subgr)  Passes  Remaining life****
514  ""   LOCATE 20,1: INPUT ""ENTER SELECTION AND <RET>:"": SLF
515  IF (SLF < 1) OR (SLF > 2) THEN 6510
516  ""#536  OPEN ""#0"", #2, X$: PRINT #2, "" ""****
517  ""CLS: PRINT #2, ""TEXAS TRANSPORTATION INSTITUTE ""
518  ""PRINT #2, ""LOAD RATING OF LIGHT PAVEMENT""
519  ""PRINT #2, ""
520  ""PRINT #2, ""JOB : "": a$: ""(INPUT FILE "": x$: "")""
521  ""PRINT #2, ""
522  ""PRINT #2, ""DISTRICT:"": ID: "": COUNTY:"": C$: "": ROAD:"":
FS"
523  ""PRINT #2, ""ALLOWABLE RUT(INS):"": RX: "": (RECORDED RUT(INS):"
RM"
524  ""PRINT #2, ""
525  ""PRINT #2, ""TRUCK NO. 1""
526  ""PRINT #2, ""AXLE NUMBER SINGLE WHEEL/ESWL(LBS)"
527  ""PRINT #2, TAB(5): ""1": TAB(20): ""9600""
528  ""PRINT #2, ""
529  ""PRINT #2, ""ANNUAL TRAFFIC GROWTH RATE:"": GR: "": # OF YEARS:"": AL:
""FIRST YEAR TRAFFIC:"": ESL""
530  ""PRINT #2, ""TOTAL NUMBER OF PASSES DURING ABOVE PERIOD:"": PA"
531  ""PRINT #2, ""
532  ""IF EO = 0 THEN PRINT #2, ""DATE:"": D$: "": DYNAMAFLECT"
533  ""IF EO = 1 THEN PRINT #2, ""DATE:"": D$: "": FALLING WEIGHT
DEFLECTOMETER""
534  IF GR < 0 THEN 6562
535  ""IF TEMRUT = 1 THEN PRINT #2, "" TEMP. CORRECTED
* ""** ""**""
536  ""IF TEMRUT = 0 AND SLF = 1 THEN PRINT #2, ""**""**""**""**""
537  ""IF TEMRUT = 0 AND SLF = 1 THEN PRINT #2, """" SEC BASE E1-BASE
E2-SUBGR NO. OF RUT DEPTH REMAINING LIFE""
538  ""IF TEMRUT = 1 THEN PRINT #2, """" SEC BASE E1-BASE E2-SUBGR
NO. OF RUT DEPTH REMAINING LIFE"
539  ""IF TEMRUT = 0 AND SLF = 1 THEN PRINT #2, ""# (IN) (PSI)
(PSI) PASSES (IN) (YEARS)"
540  ""IF TEMRUT = 1 THEN PRINT #2, ""# (IN) (PSI) (PSI)
PASSES (IN) (YEARS)"
541  ""IF TEMRUT = 0 AND SLF = 1 THEN FOR I = 1 TO NC: PRINT #2, USING
""###.###.###.###.###.###.###.###.###.###.###.###.###.###.###.###.###
: NS(I); BA(I); E1(I); E2(I); PS(I); AM(I); NY(I): NEXT"
542  ""IF TEMRUT = 1 THEN FOR I = 1 TO NC: PRINT #2, USING ""###.###.###.###
#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####
: NS(I); BA(I); E1(I); E2(I); PS(I); AM(I); NY(I): NEXT"
543  ""IF SLF = 2 THEN PRINT #2, ""*
544  ""IF SLF = 2 AND EO = 1 THEN PRINT #2, """" SEC BASE D1 D7
FWD LOAD E1-BASE E2-SUBGR NO. OF REMAINING LIFE"
545  ""IF SLF = 2 AND EO = 0 THEN PRINT #2, """" SEC BASE D1 D5
FWD LOAD E1-BASE E2-SUBGR NO. OF REMAINING LIFE"
546  ""IF SLF = 2 THEN PRINT #2, ""# (IN) (MILS) (MILS) (LBS)
(PSI) (PSI) PASSES (YEARS)"
547  ""IF SLF = 2 AND EO = 1 THEN FOR I = 1 TO NC: PRINT #2, USING ""###.###
#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####
: NS(I); BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I); NY(I): NEXT"
548  ""IF SLF = 2 AND EO = 0 THEN FOR I = 1 TO NC: PRINT #2, USING ""###.###
#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####
: NS(I); BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I); NY(I): NEXT"
PRINT #2,
IF SLF = 1 THEN PRINT #2, "" * NUMBER OF PASSES AND REMAINING LIFE
FOR SPECIFIED RUT DEPTH OF ""; RX; ""in"")
IF SLF = 1 THEN PRINT #2, "" ** RUT DEPTH FOR SPECIFIED NUMBER OF
PASSES OF ""; PA; "" IN "; AL; ""YEARS"");
IF SLF = 2 THEN PRINT #2, "" * NUMBER OF PASSES AND REMAINING LIFE
FOR SPECIFIED RUT DEPTH OF ""; RX; ""in"");
PRINT #2,
PRINT #2. "" Remaining Life(yrs) Cumulative %
Sections"
FOR II = 1 TO NYMAX
PRINT #2, USING "" ## ###.##"";
II; Dyr(I)
NEXT II
PRINT #2.
PRINT #2.
IF S100 >= 10 THEN PRINT #2, "" LEGAL LEGAL
IF S100 = 1 THEN PRINT #2, USING "" ## ##""; AL1; AL2; AL3"
GOTO 6563

=================================================================
SAVE DATA WITHOUT REMAINING LIFE - GR<0
=================================================================
IF TEMRUT = 1 THEN PRINT #2, "" TEMP. CORRECTED
* **
IF TEMRUT = 0 AND SLF = 1 THEN PRINT #2, ""
* **
IF TEMRUT = 0 AND SLF = 1 THEN PRINT #2, "" SEC BASE E1-BASE
E2-SUBGR NO. OF RUT DEPTH ""
IF TEMRUT = 1 THEN PRINT #2, "" SEC BASE E1-BASE E2-SUBGR
NO. OF RUT DEPTH ""
IF TEMRUT = 0 AND SLF = 1 THEN PRINT #2, "" # (IN) (PSI)
(PSI) PASSES (IN) ""
IF TEMRUT = 1 THEN PRINT #2, "" # (IN) (PSI) (PSI)
PASSES (IN) ""
IF TEMRUT = 0 AND SLF = 1 THEN FOR I = 1 TO NC: PRINT #2, USING
###.### ###.### "" NS(I): BA(I); E1(I); E2(I); PS(I); AM(I); NEXT"
IF TEMRUT = 1 THEN FOR I = 1 TO NC: PRINT #2, USING ""###.### ###.### "" NS(I): BA(I); E1(I); E2(I); PS(I); AM(I); NEXT"
IF SLF = 2 THEN PRINT #2, ""
* **
IF SLF = 2 AND EQ = 1 THEN PRINT #2, "" SEC BASE D1 D7
FWD LOAD E1-BASE E2-SUBG NO. OF REMAINING LIFE"
IF SLF = 2 AND EQ = 0 THEN PRINT #2, "" SEC BASE D1 D5
FWD LOAD E1-BASE E2-SUBG NO. OF REMAINING LIFE"
IF SLF = 2 THEN PRINT #2, "" # (IN) (MILS) (MILS) (LBS)
(PSI) (PSI) PASSES (YEARS)"
IF SLF = 2 AND EQ = 1 THEN FOR I = 1 TO NC: PRINT #2, USING ""###.### ""
#### "" NS(I): BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I); NEXT"

34
583 "IF SLF = 2 AND EO = 0 THEN FOR I = 1 TO NC: PRINT #2, USING ""##.##"";
NS(I); BA(I); D(I, 1); D(I, 7); FE(I); E2(I); PS(I): NEXT"
584 "IF SLF = 1 THEN PRINT #2, ""* NUMBER OF PASSES FOR SPECIFIED RUT DEPTH OF "";
SD; RX: ""** NUMBER OF PASSES OF "";
585 IF SLF = 1 THEN PRINT #2, ""** NUMBER OF PASSES OF "";
586 IF SLF = 1 THEN PRINT #2, ""** NUMBER OF PASSES OF "";
587 IF SLF = 1 THEN PRINT #2, ""** NUMBER OF PASSES FOR SPECIFIED RUT DEPTH OF "";
588 IF SLF = 1 THEN PRINT #2, ""** NUMBER OF PASSES FOR SPECIFIED RUT DEPTH OF "";
589 IF SLF = 1 THEN PRINT #2, ""** NUMBER OF PASSES FOR SPECIFIED RUT DEPTH OF "";
590 IF TEC = 1 THEN PRINT #2, ""* DO YOU WANT TO SAVE TEMPERATURE AND MOISTURE CORRECTED DATA? 1=YES, 0=NO "";
591 IF TEC <> 1 THEN 6563
592 "PRINT #2, ""* INPUT CONDITIONS: WANTED CONDITIONS: "";
593 "PRINT #2, ""* INPUT CONDITIONS: WANTED CONDITIONS: "";
594 "PRINT #2, ""* INPUT CONDITIONS: WANTED CONDITIONS: "";
595 "PRINT #2, USING ""* TEMPERATURE = "";
596 "PRINT #2, USING ""* SUCION = "";
597 "PRINT #2,""* INPUT WANTED CONDITION MODULUS CHANGE TO SUCTION "";
598 "PRINT #2, ""* MODULUS MODULUS DUE TO TEMPER. DUE TO SUCTION "";
599 "PRINT #2, ""* MODULUS MODULUS DUE TO TEMPER. DUE TO SUCTION "";
600 "PRINT #2, ""* MODULUS MODULUS DUE TO TEMPER. DUE TO SUCTION "";
601 "PRINT #2,""* MODULUS MODULUS DUE TO TEMPER. DUE TO SUCTION "";
602 "FOR I = 1 TO KK1
603 "PRINT #2, USING ""* MODULUS MODULUS DUE TO TEMPER. DUE TO SUCTION "";
604 NEXT I:
605 6584 RETURN
606 '=====================================================================
607 6970 'SUBROUTINE MAINMENU
608 '=====================================================================
609 'KEY OFF: CLS
610 'Input for geometric regression GEOMINPUT
611 'for non-standard FWD sensor spacing
612 'GR = 1: CLS: 'To initialize and activate GEOMBASIN
613 'Standard FWD sensor spacing for Texas SDHPT is as follows
614 'LOCATE 5, 10: PRINT ""FWD Sensor #1 #2 #3 #4 #5 #6 #7"";
615 "LOCATE 5, 10: PRINT ""FWD Sensor #1 #2 #3 #4 #5 #6 #7"";
616 "LOCATE 6, 10: PRINT ""Dist. (ins) from load"";
617 "LOCATE 7, 10: PRINT ""Type in new spacing ......."";
618 "LOCATE 8, 10: INPUT ""Sensor #1 at (ins)""; x(1)
619 "LOCATE 9, 10: INPUT ""Sensor #2 at (ins)""; x(2)
620 "LOCATE 10, 10: INPUT ""Sensor #3 at (ins)""; x(3)
621 "LOCATE 11, 10: INPUT ""Sensor #4 at (ins)""; x(4)
622 "LOCATE 12, 10: INPUT ""Sensor #5 at (ins)""; x(5)
623 "LOCATE 13, 10: INPUT ""Sensor #6 at (ins)""; x(6)
624 "LOCATE 14, 10: INPUT ""Sensor #7 at (ins)""; x(7)
625 IF x(1) = 0 THEN x(1) = .01
SUBROUTINE geometric regression GEOMBASIN

J = 0; K = 0; L = 0; M = 0; R2 = 0
N = 4

IF D(I, 7) = 0 THEN N = 3
IF x(7) = 0 THEN N = 3
FOR II = 4 TO (3 + N)

x = LOG(x(II)); Y = LOG(O(I, II))
J = J + x; K = K + Y; L = L + x * x; M = M + Y * Y; R2 = R2 + x * Y

NEXT II

b = (N * R2 - K * J) / (N * L - J * J)
a = (K - b * J) / N
J = b * (R2 - J * K / N)
M = M - K * K / N
K = M - J
R2 = J / M

RETURN

SUBROUTINE OUTMENU

PRINT "press return to continue(line7581)"; INPUT nn$
CLS; KEY OFF
LOCATE 2, 10: PRINT "OUTPUT MENU"
LOCATE 4, 10: PRINT "Select Optional Output Format"
LOCATE 6, 10: PRINT "(1) Plot Base Course Elastic Modulus"
LOCATE 7, 10: PRINT "(2) Plot Subgrade Elastic Modulus"
LOCATE 8, 10: PRINT "(3) Plot Number of Vehicle Passes"
LOCATE 9, 10: PRINT "(4) Plot Remaining Life"
LOCATE 10, 10: PRINT "(5) Temperature Correction of Base Modulus"
LOCATE 11, 10: PRINT "(5.1) Rut depth, Passes and Remaining Life"
LOCATE 12, 10: PRINT "for corrected moduli"
LOCATE 13, 10: PRINT "(6) Save the Output as a file"
LOCATE 14, 10: PRINT "(7) Exit to Main Menu"
LOCATE 20, 10: INPUT "ENTER SELECTION AND <RET> : ": SLO
IF SLO = 7 OR SLO = 0 THEN CLS; SLO = 0; TEMRUT = 0; TEC = 0; GOTO 5080

IF SLO = 6 THEN GOSUB 6510; GOTO 7580: 'To OUTFILE and back
IF SLO = 5 THEN TEC = 1: GOSUB 10010
IF SLO = 5.1 THEN FOR I = 1 TO 100: NY(I) = 0: NEXT
IF SLO = 5.1 THEN FOR I = 1 TO 100: DYR(I) = 0: NEXT
IF SLO = 5.1 THEN FOR I = 1 TO 100: PS(I) = 0: NEXT
IF SLO = 5.1 THEN TEMRUT = 1: GOTO 5080
PRCHK = 0
CLS
PRINT : PRINT
PRINT
IF SLO = 1 THEN t$ = "ELASTIC MODULUS OF BASE COURSE"
IF SLO = 2 THEN t$ = "ELASTIC MODULUS OF SUBGRADE"
IF SLO = 3 THEN t$ = "NUMBER OF PASSES FOR RUT DEPTH"
IF SLO = 4 THEN t$ = "REMAINING LIFE"
MINY = 0
IF SLO = 1 THEN INPUT "Minimum E1 [Default=0]: "; MINY
IF SLO = 1 THEN INPUT "Maximum E1 [Default=100000]: "; MAXY
IF (SLO = 1) AND (MAXY = 0) THEN MAXY = 100000
IF SLO = 2 THEN INPUT "Minimum E2 [Default=0]: "; MINY
IF SLO = 2 THEN MAXY = 25000: INPUT "Maximum E2 [Default=25000]: "; MAXY
IF (SLO = 2) AND (MAXY = 0) THEN MAXY = 500000
IF SLO = 3 THEN INPUT "Minimum PASSES [Default=0]: "; MINY
IF SLO = 3 THEN INPUT "Maximum PASSES [Default=0.5E6]: "; MAXY
IF (SLO = 3) AND (MAXY = 0) THEN MAXY = 500000
IF SLO = 4 THEN INPUT "Minimum REMAINING LIFE [Default=0]: "; MINY
IF SLO = 4 THEN INPUT "Maximum REMAINING LIFE [Default=24]: "; MAXY
IF (SLO = 4) AND (MAXY = 0) THEN MAXY = 24
PR = 1
XSacle = 60 / (MAXX - MINX)
YScale = 20 / (MAXY - MINY)
YORG = 22 + MINY * YSCALE
YORG = 10 - MINX * XSCALE
KEY OFF
CLS
'INPUT "press return to continue (line8299)"; nn$: CLS'
IF PRCHK = 0 THEN PR1 = 196: PR2 = 179: PR3 = 218: PR4 = 191: PR5 = 192
IF PRCHK = 0 THEN PR6 = 217: PR7 = 193: PR8 = 195: PR9 = 180: PR10 = 194
FOR COL = 11 TO 69
  LOCATE 2, COL: PRINT CHR$(PR1)"
NEXT COL
FOR ROW = 3 TO 21
  LOCATE ROW, 10: PRINT CHR$(PR2)"
  LOCATE ROW, 70: PRINT CHR$(PR2)"
NEXT ROW
  LOCATE 2, 10: PRINT CHR$(PR3)"
  LOCATE 2, 70: PRINT CHR$(PR4)"
  LOCATE 22, 10: PRINT CHR$(PR5)"
  LOCATE 22, 70: PRINT CHR$(PR6)"
FOR I = 1 TO 9
  LOCATE 22, (10 + 6 * I): PRINT CHR$(PR7)"
  LOCATE (2 + 2 * I), 10: PRINT CHR$(PR8)"
  LOCATE (2 + 2 * I), 70: PRINT CHR$(PR9)"
  LOCATE 2, (10 + 6 * I): PRINT CHR$(PR10)"
NEXT I
IF PRCHK = 1 AND PT = 1 THEN GOSUB 9350
IF PRCHK = 1 AND PT = 2 THEN GOSUB 9210
LOCATE 1, 10: PRINT t$"
LOCATE 1, 55: PRINT x$"
XINT = (MAXX - MINX) / 10
YINT = (MAXY - MINY) / 10
LOCATE 23, 8: PRINT (MINX)"
LOCATE 23, 20: PRINT (MINX + 2 * XINT)"
LOCATE 23, 32: PRINT (MINX + 4 * XINT)"
LOCATE 23, 44: PRINT (MINX + 6 * XINT)"
LOCATE 23, 56: PRINT (MINX + 8 * XINT)"
LOCATE 23, 69: PRINT (MINX + 10 * XINT)"
LOCATE 2, 1: PRINT MAXY
LOCATE 6, 1: PRINT INT(MAXY - YINT * 2) * 100 / 100
LOCATE 10, 1: PRINT INT(MAXY - YINT * 4) * 100 / 100
LOCATE 14, 1: PRINT INT(MAXY - YINT * 6) * 100 / 100
LOCATE 18, 1: PRINT INT(MAXY - YINT * 8) * 100 / 100
LOCATE 22, 1: PRINT MINY

IF SLO = 1 THEN LOCATE 11, 2: PRINT ", E1"
IF SLO = 1 THEN LOCATE 12, 2: PRINT ", (psi)"
IF SLO = 2 THEN LOCATE 11, 2: PRINT ", E2"
IF SLO = 2 THEN LOCATE 12, 2: PRINT ", (psi)"
IF SLO = 3 THEN LOCATE 11, 2: PRINT ", No. of"
IF SLO = 3 THEN LOCATE 12, 2: PRINT ", Passes"
IF SLO = 4 THEN LOCATE 11, 2: PRINT ", Remain."
IF SLO = 4 THEN LOCATE 12, 2: PRINT ", Life"
IF SLO = 4 THEN LOCATE 13, 2: PRINT ", (years)"
IF PRCHK = 1 THEN LOCATE 25, 1: PRINT SPC(70);" GOTO 8300
IF PRCHK = 1 THEN LOCATE 25, 1: PRINT ", RE-PLOTTING ... ";
FOR I = 1 TO NC
x = NS(I)
IF SLO = 1 THEN Y = E1(I)
IF SLO = 1 AND Y > MAXY THEN Y = MAXY
IF SLO = 2 THEN Y = E2(I)
IF SLO = 2 AND Y > MAXY THEN Y = MAXY
IF SLO = 3 THEN Y = PS(I)
IF SLO = 3 AND Y > MAXY THEN Y = MAXY
IF SLO = 4 THEN Y = NY(I)
IF SLO = 4 AND Y > MAXY THEN Y = MAXY
SCY = YORG - INT(Y * YSCALE); SCX = XORG + INT(x * XSCALE)
IF (SCX < 10) OR (SCX > 70) OR (SCY < 2) OR (SCY > 22) THEN 9050
LOCATE SCY, SCX: PRINT CHR$(42)
LOCATE 25, 1: PRINT SPC(75);
" 9050 NEXT I
LOCATE 25, 1: PRINT SPC(75);
PRCHK = 1: PRINT ", PRESS ENTER TO CONTINUE (line9051)"; : INPUT ; nn$"
PRCHK = 1
GOTO 8300

=====================================================================
9210 ' SUBROUTINE MT180 to replot the graph for the mannesmann tally
=====================================================================
PR1 = 157: PR2 = 156: PR3 = 134: PR4 = 149: PR5 = 153
PR6 = 154: PR7 = 158: PR8 = 150: PR9 = 151: PR10 = 152
PRCHK = 1: RETURN

=====================================================================
9350 ' SUBROUTINE EPSON to replot the graph for the epson/panasonic
=====================================================================
PR1 = 46: PR2 = 124: PR3 = 46: PR4 = 46: PR5 = 46
PR6 = 46: PR7 = 43: PR8 = 43: PR9 = 43: PR10 = 43
PRCHK = 1: RETURN

=====================================================================
10010 ' TEMPERATURE CORRECTION OF BASE MODULUS ONLY
=====================================================================
THIS SUBROUTINE PREDICTS THE MODULUS OF GRANULAR MATERIALS
AT DIFFERENT TEMPERATURES AND MOISTURES BASED ON A
MICROMECHANICAL APPROACH

CLS

PRINT : PRINT : PRINT

PRINT TAB(25); "TYPE OF BASE MATERIAL:" : PRINT

PRINT TAB(25); " 1 = CRUSHED LIMESTONE "'

PRINT TAB(25); " 2 = LIME ORE GRAVEL"

PRINT TAB(25); " 3 = IRON ORE GRAVEL"

PRINT TAB(25); " 4 = CALICHE"

PRINT TAB(25); " 5 = RIVER GRAVEL"

PRINT TAB(25); " 6 = CALICHE GRAVEL"

PRINT TAB(25); " 7 = SANDSHELL"

PRINT : PRINT : PRINT

PRINT TAB(25); "PROPERTIES OF THE MATERIAL SELECTED:"

PRINT TAB(25); "1. LINEAR THERMAL EXPANSION ALP = .... ALP"

PRINT TAB(25); "2. ELASTIC MODULUS OF SOIL GRAINS E = .... E"

PRINT TAB(25); "3. POISSON'S RATIO U = .... U"

PRINT TAB(25); "4. K1 = .... K1"

PRINT TAB(25); "5. K2 = .... K2"

PRINT TAB(25); "DO YOU WANT TO CHANGE? O=NO 1=YES " : CHOICE

PRINT : PRINT : PRINT

PRINT TAB(25); "UNIT WEIGHT OF MATERIAL (pcf) : UW"

PRINT : PRINT : PRINT

PRINT TAB(25); "CONDITIONS AT WHICH THE MODULUS IS OBTAINED:"

PRINT TAB(25); "TEMPERATURE, Ti (40 deg.F < Ti < 110 deg.F) :

PRINT TAB(25); "SUCTION, Hi (-145 psi < Hi < 0 psi) :

PRINT

LOCATE 3, 5: INPUT "UNIT WEIGHT OF MATERIAL (pcf) : UW"

IF (UW < 100) OR (UW > 170) THEN 10460

IF SELECT1 = 2 THEN 10478

CH01 = 1

GOTO 10500

K1 = 1

LOCATE 5, 5: INPUT "MODULUS OF MATERIAL (psi) : EI"

IF (EI < 20000) OR (EI > 600000) THEN 10478

LOCATE 7, 5: PRINT "CONDITIONS AT WHICH THE MODULUS IS OBTAINED:

LOCATE 9, 5: INPUT "TEMPERATURE, Ti (40 deg.F < Ti < 110 deg.F) :

LOCATE 11, 5: INPUT "SUCTION, Hi (-145 psi < Hi < 0 psi) :

10460: K2 = .4

10478: U = .45

10500: K2 = .45

10540 LOCATE 11. 5: INPUT "SUCTION, Hi (-145 psi < Hi < 0 psi) :

10540 LOCATE 11. 5: INPUT "SUCTION, Hi (-145 psi < Hi < 0 psi) :
IF (HI < -145) OR (HI > 0) THEN 10540
10560 LOCATE 13, 5: PRINT """"INPUT WANTED CONDITIONS: """
10570 """"LOCATE 15, 5: INPUT """"TEMPERATURE, To (40 deg.F < To < 110 deg.F) :
"""
10580 """"RTEMP"
10590 IF (RTEMP < 40) OR (RTEMP > 110) THEN 10560
10600 10590 LOCATE 17, 5: INPUT """"SUCTION, Ho (-145 psi < Ho < 0 psi) :
"""
144 IF (HL < 145) OR (HL > 0) THEN 10590
145 IF CH01 = 1 THEN KK1 = NC
146 FOR I = 1 TO KK1
147 IF CH01 = 1 THEN EI = EI(I): K1 K1(I): K2 = .33
148 VOID = 1 / (UW / SG / 62.4) - 1
149 N2 = VOID / (1 + VOID)
150 XS = (.4764 - N2) / (.4764 - .2595)
151 IF XS < 0 THEN XS = 0
152 IF XS > 1 THEN XS = 1
153 PRO1 = .75 * (1 - U ^ 2) / E
154 THETA = 10 ^ (LOG(EI / K1) / LOG(10) / K2)
155 DTEMP = RTEMP - TTEMP
156 IF DTEMP = 0 THEN det(I) = 0: GOTO 10800
157 DT = ABS(DTEMP)
158 DELV = DT * ALP * 3
159 P = (XS / SQR(2) / PRO1 + (1 - XS) / 4 / PRO1) * (1 / 3 * DELV) ^ (1.5)
160 PT1 = (DTEMP / DT) * P
161 det(I) = K1 * K2 * THETA ^ (K2 - 1) * PT1
162 10800 PS = -(HL - HI) * .13
163 des(I) = K1 * K2 * THETA ^ (K2 - 1) * PS
164 TOTDE(I) = det(I) + des(I)
165 NE(I) = EI + TOTDE(I)
166 OE(I) = EI
167 NEXT I
168 CLS
169 PRINT : PRINT : PRINT TAB(10); PRINT """"INPUT CONDITIONS: WANTED CONDITIONS:"
170 PRINT 
171 """"PRINT USING """"TEMPERATURE = ###.# deg. F TEMPERATURE =
172 ###.# deg. F""""; TTEMP; RTEMP"
173 """"PRINT USING """"SUCTION = #######.## psi SUCTION =
174 #######.## psi """"; HI; HL"
175 PRINT
176 """"PRINT """"INPUT WANTED CONDITION MODULUS CHANGE MODULUS
177 CHANGE"
178 """"PRINT """"MODULUS MODULUS DUE TO TEMPER. DUE TO
179 SUCTION"
180 """"PRINT """"(psi) (psi) (psi)
181 """"PRINT """"(psi)"
182 PRINT 
183 FOR I = 1 TO KK1
184 """"PRINT USING """"####.########.########.####; OE(I); NE(I); det(I); des(I)"
185 NEXT I
186 PRINT : PRINT
187 """"INPUT """"PRESS RETURN TO CONTINUE............"""": PRESS"
188 CLS
189 IF SLECT1 <> 2 THEN 10988
190 """"PRINT : PRINT : PRINT """"1. ANOTHER TEMPERATURE CORRECTION
""""
PRINT: PRINT "" 2. EXIT TO MAIN MANUAL "";
CH02
IF CH02 = 1 THEN 10010
GOTO 730

10988 PRINT: PRINT: PRINT "" 1. EXIT TO MAIN MENU "";
PRINT: PRINT: PRINT "" 2. EXIT TO OUTPUT MENU "";
CH03
IF CH03 = 1 THEN 730
GOTO 7580

***********************************************************************
* RUT DEPTH CALCULATION ONLY
***********************************************************************
by K.A.S. Yapa
" Pavement Systems, Texas Transportation Institute,"
" TTI Building, Texas A & M University,"
" College Station, Texas 77843."
(409)-845-9910.
24th MARCH 1988.

FOR I = 1 TO 2
FOR J = 1 TO 3
FOR K = 1 TO 3
FOR L = 1 TO 3
READ RUT(I, J, K, L, 1), RUT(I, J, K, L, 2), RUT(I, J, K, L, 3)
NEXT L: NEXT K: NEXT J: NEXT I
GOSUB 14240
IF INIOPT = 2 THEN NC = 1
FOR INC = 1 TO NC
IF TEMRUT = 1 THEN E1(INC) = NE(INC)
GOSUB 14850
" GOTO INPUT1 IF ANY CORRECTIONS ARE NEEDED
IF CORR = 1 THEN GOSUB 14240
CLS
PRINT "" Calculating........... ""
SELECT PARAMETERS FOR INTERPOLATION
FOR I = 1 TO 2
FOR J = 1 TO 3
FOR K = 1 TO 3
FOR L = 1 TO 3
FOR M = 1 TO 3
Y1(M) = RUT(I, J, K, L, M); X1(M) = XM(M): NEXT M
NUM = 3: x = XMM: GOSUB 13700
YLTEMP(L) = Y: XMFLAG = XFLAG
NEXT L
FOR LL = 1 TO 3: X1(LL) = XL(LL): Y1(LL) = YLTEMP(LL): NEXT LL
NUM = 3: x = XLL: GOSUB 13700
YKTEMP(K) = Y: XLFLAG = XFLAG

PRINT PRINT INPUT  Select the Operation
IF CH02 = 1 THEN 10010
GOTO 730
PRINT PRINT INPUT  Select the Operation
IF CH03 = 1 THEN 730
GOTO 7580

***********************************************************************
*-loop to calculate rut depths for each fwd section
***********************************************************************
GOSUB 14240
12400
FOR KK = 1 TO 3: X1(KK) = XK(KK): Y1(KK) = YKTEMP(KK): NEXT KK
NUM = 3: x = XK: GOSUB 13700
YJTEMP(J) = Y: XJFLAG = XFLAG
NEXT J

FOR JJ = 1 TO 3: X1(JJ) = XJ(JJ): Y1(JJ) = YJTEMP(JJ): NEXT JJ
NUM = 3: x = XJ: GOSUB 13700
YITEMP(I) = Y: XJFLAG = XFLAG
NEXT I

FOR II = 1 TO 2: X1(II) = XI(II): Y1(II) = YITEMP(II): NEXT II
NUM = 2: x = XI: GOSUB 13700
RUTCAL = Y * CYL / 300000: XIFLAG = XFLAG

'GOTO SUBROUTINE OUTPUT
GOSUB 15680
NEXT INC
IF INIOPT <> 2 THEN RETURN
PRINT: PRINT: PRINT
"1. ANOTHER RUT DEPTH CALCULATION"
"2. EXIT TO MAIN MANUAL"
PRINT: INPUT "" Select the Operation

CH05"
IF CH05 = 1 THEN 12400
GOTO 730

'DATA BASE OF RUT DEPTHS
DATA 0.2609, 0.2804, 0.3191, 0.2216, 0.2223, 0.1831, 0.1502, 0.1253
DATA 0.2705, 0.3030, 0.3431, 0.2451, 0.2491, 0.2556, 0.2168, 0.1951, 0.1680
DATA 0.367, 0.4519, 0.5286, 0.3503, 0.4172, 0.4580, 0.3331, 0.3831, 0.3882
DATA 0.2702, 0.2912, 0.3019, 0.2249, 0.2185, 0.2002, 0.1797, 0.1458, 0.0991
DATA 0.2804, 0.3099, 0.3549, 0.2473, 0.2490, 0.2600, 0.2137, 0.1882, 0.1652
DATA 0.3900, 0.4610, 0.5322, 0.3688, 0.4231, 0.4561, 0.3475, 0.3842, 0.3798
DATA 0.2946, 0.3130, 0.3110, 0.2344, 0.2263, 0.2031, 0.1750, 0.1403, 0.0837
DATA 0.3013, 0.3356, 0.3755, 0.2548, 0.2600, 0.2687, 0.2087, 0.1842, 0.1618
DATA 0.3977, 0.4840, 0.5424, 0.3750, 0.4343, 0.4541, 0.3498, 0.3834, 0.3670
DATA 0.1613, 0.2376, 0.3056, 0.1258, 0.1727, 0.2086, 0.0934, 0.1082, 0.1119
DATA 0.1901, 0.2528, 0.3247, 0.1588, 0.1993, 0.2374, 0.1280, 0.1457, 0.1500
DATA 0.2607, 0.3681, 0.4915, 0.2453, 0.3355, 0.4217, 0.2300, 0.3038, 0.3523
DATA 0.1951, 0.2524, 0.2767, 0.1491, 0.1794, 0.1747, 0.1039, 0.1072, 0.0732
DATA 0.1957, 0.2643, 0.3400, 0.1628, 0.2037, 0.2453, 0.1296, 0.1433, 0.1505
DATA 0.2920, 0.4006, 0.5025, 0.2713, 0.3648, 0.4262, 0.2508, 0.3230,
DATA 0.2254, 0.2785, 0.2834, 0.1652, 0.1918, 0.1736, 0.1056, 0.1059.
DATA 0.2256, 0.2974, 0.3642, 0.1790, 0.2216, 0.2574, 0.1328, 0.1461.
DATA 0.3322, 0.4287, 0.5197, 0.3034, 0.3786, 0.4327, 0.2749, 0.3287.
DATA 0.3456.

INPUT PARAMETERS USED IN CREATING THE DATA BASE
XM(#), XL(#), XK(#), XJ(#), XI(#).
DATA 18, 0.0100, 25000, 100000, 0.0075.
DATA 12, 0.0060, 15000, 70000, 0.0025.
DATA 6, 0.0020, 5000, 40000, 0.0000.
END

============================================================
SUBROUTINE FOR POLYNOMIAL INTERPOLATION
NS = 1
DIF = ABS(x - XI(1))
FOR a = 1 TO NUM
DIFT = ABS(x - XI(a))
'SELECT THE BEST STARTING POINT
IF DIFT < DIF THEN NS = a: DIF = DIFT
C(a) = Y1(a); DD(a) = Y1(a)
NEXT a
XFLAG = 0
'ENFORCE LIMITS ON EXTRAPOLATION (MAXIMUM = 1.5 * DIFFERENCE BETWEEN TWO
'CONSECUTIVE PARAMETER LEVELS)
IF DIF <= 1.5 * (ABS(X1(1) - X1(2))) THEN 13980
DIF = 1.5 * (ABS(X1(1) - X1(2)))
IF NS = 1 THEN x = X1(1) + DIF
IF NS = NUM THEN x = X1(NUM) - DIF
XFLAG = x
Y = Y1(NS)
NS = NS - 1
AEND = NUM - 1
FOR a = 1 TO AEND
HO = X1(a) - x
HP = X1(a + 1) - x
W = C(a + 1) - DD(a)
DEN = W / (HO - HP)
DD(a) = HP * DEN
C(a) = HO * DEN
NEXT a
'PICK THE SHORTEST PATH TO MOVE
IF (2 * NS) < AEND THEN DY = C(NS + 1): GOTO 14210
DY = DD(NS): NS = NS - 1
Y = Y + DY
RETURN

============================================================
SUBROUTINE INPUT1
USE IF ONLY THE RUT LEVELS ARE NEEDED
IF INIOPT <> 2 GOTO 14400
CLS: INPUT "JOB DESCRIPTION: "'; AA$'
INPUT "Resilient Modulus - Base Course (psi)"'; EBA"
CLS: LOCATE 4, 10: PRINT "" Subgrade Material Type:"
LOCATE 7, 10: PRINT ""1) Heavy Clay CCH)"
LOCATE 9, 10: PRINT ""2) Light/Silty Clay, Clayey Silt CCL ML)"
LOCATE 11, 10: PRINT ""3) Clayey/Silty/Uniform Sand - (SC-SM)
LOCATE 15, 10: PRINT ""ENTER SELECTION & <RET>""; MSG
GOTO 14750

CLS: LOCATE 4, 10: PRINT ""Laboratory Data Input:""
LOCATE 6, 10: PRINT ""Log a - Intercept of the Straight Line Fit on a Log Log Plot of Accumulated Residual Strain""
LOCATE 7, 10: PRINT ""b - Slope of the Straight Line Fit"
LOCATE 8, 10: PRINT ""Subgrade Material ,,,,
LOCATE 10, 10: PRINT ""b - Subgrade Material\nLOCATE 11, 10: PRINT ""Subgrade CH - Clay"
LOCATE 13, 10: PRINT ""Subgrade CL-ML"
LOCATE 15, 10: PRINT ""Subgrade SC-SM"
LOCATE 16, 10: PRINT ""Base Course Material ,,,
LOCATE 18, 10: PRINT ""Subgrade Material ,,,
LOCATE 20, 10: PRINT ""Subgrade Material\nLOCATE 22, 10: INPUT ""DO YOU WANT ANY CORRECTIONS - O=NO, 1=YES""; CORR

IF CORR = 1 THEN 14405
IF CORR <> 0 GOTO 14714
GOTO 14714
CLS: BEEP: PRINT ""Input Data are Incompatible. Check & Re-enter"":"
INPUT ""Press RETURN to continue""; mn$
GOTO 14405

14750 SKIP IF ONLY RUT LEVELS ARE CALCULATED
14780 IF INIOPT = 2 GOTO 14840
14790 'OBTAIN THE TRUCK PASSES AND ALLOWABLE AND MEASURED RUT LEVELS FROM
14800 'THE MAIN PROGRAM
14810 EQPASS = PA
14820 RALLOW = RX
14830 REXIST = RM
14840 14840 RETURN

14850 'SUBROUTINE INPUT2
14880 'SKIP IF ONLY RUT LEVELS ARE CALCULATED
14900 IF INIOPT = 2 GOTO 14940
14910 'APPROXIMATE METHOD TO DETERMINE RUTTING BEHAVIOR
14930 IF OPP <> 1 GOTO 15280
14940 AABA = .0174: AACH = .0933: AACL = .001: AASM = .095
14950 ABBA = -.57: ABCH = -2.64: ABCL = -.73: ABSM = -1.95
14960 BBA = .125: BCH = .236: BCL = .162: BSM = .142
14970 ABA = AABA * (EBA / 1000) ^ ABBA
14980 IF ABA > .0035 THEN ABA = .0035
14990 IF MSG <> 1 THEN PRINT ''Subgrade CH - Clay''
15000 IF MSG = 2 THEN PRINT ''Subgrade CL-ML''
15010 IF MSG <> 3 THEN PRINT ''Subgrade SC-SM''
15020 GOTO 15510

15190 LOCATE 2, 10: PRINT ''JOB: ''; AA$
15210 LOCATE 4, 10: PRINT ''Resilient Modulus (psi) - Base = ''; EBA
15230 LOCATE 6, 10: PRINT '' - Subgrade = ''; ESG
15250 LOCATE 8, 10: PRINT ''Thickness of Base Layer = ''; TBA;
15270 LOCATE 10, 10: PRINT ''# of Equivalent Standard Wheel Passes ='';
15290 IF OPP <> 1 GOTO 15460
15310 LOCATE 12, 10: PRINT ''Allowable Rut Depth =''; RALLOW; '' in.''
15330 LOCATE 14, 10: PRINT ''Measured Rut Depth =''; REXIST; '' in.''
15350 IF OPP <> 1 GOTO 15460
15370 LOCATE 16, 10: IF MSG = 1 THEN PRINT ''Subgrade - CH - Clay''
15390 LOCATE 16, 10: IF MSG = 2 THEN PRINT ''Subgrade - CL-ML''
15410 LOCATE 16, 10: IF MSG = 3 THEN PRINT ''Subgrade - SC-SM''
15430 GOTO 15510
LOCATE 16, 10: PRINT """"Base Course Material - """
LOCATE 17, 10: PRINT "" Log a = "" ; LGABA; "", b = "" ;
BBA
LOCATE 19, 1G: PRINT """"Subgrade Material - """
LOCATE 20, 10: PRINT "" Log a = "" ; LGASG; "", b = "" ;
BS
LOCATE 22, 10: INPUT ""DO YOU WANT ANY CORRECTIONS - O=NO, 1=YES" ;
CORR
IF CORR = 1 THEN RETURN
IF CORR <> 0 GOTO 15510
ASSIGN VALUES TO INTERPOLATION PARAMETERS
CYL = 300000
XII = ABA * CYL ^ BBA
XLL = ASG * CYL ^ BSG
IF XII > .015 THEN XII = .015
IF XLL > .016 THEN XLL = .016
IF XLI < .001 AND TBA < 10 THEN XLL = .001
XMM = TBA: XJJ = EBA: XKK = ESG
IF TBA < 10 AND EBA < 60000 AND ESG > 30000 THEN XKK = 30000: IF TBA <
6 THEN XMM = 6: IF EBA < 30000 THEN XJJ = 30000
RETURN
ASSIGN VALUES TO INTERPOLATION PARAMETERS
CYL = 300000
XII = ABA * CYL ^ BBA
XLL = ASG * CYL ^ BSG
IF XII > .015 THEN XII = .015
IF XLL > .016 THEN XLL = .016
IF XLI < .001 AND TBA < 10 THEN XLL = .001
XMM = TBA: XJJ = EBA: XKK = ESG
IF TBA < 10 AND EBA < 60000 AND ESG > 30000 THEN XKK = 30000: IF TBA <
6 THEN XMM = 6: IF EBA < 30000 THEN XJJ = 30000
RETURN
ASSIGN VALUES TO INTERPOLATION PARAMETERS
CYL = 300000
XII = ABA * CYL ^ BBA
XLL = ASG * CYL ^ BSG
IF XII > .015 THEN XII = .015
IF XLL > .016 THEN XLL = .016
IF XLI < .001 AND TBA < 10 THEN XLL = .001
XMM = TBA: XJJ = EBA: XKK = ESG
IF TBA < 10 AND EBA < 60000 AND ESG > 30000 THEN XKK = 30000: IF TBA <
6 THEN XMM = 6: IF EBA < 30000 THEN XJJ = 30000
RETURN
ASSIGN VALUES TO INTERPOLATION PARAMETERS
CYL = 300000
XII = ABA * CYL ^ BBA
XLL = ASG * CYL ^ BSG
IF XII > .015 THEN XII = .015
IF XLL > .016 THEN XLL = .016
IF XLI < .001 AND TBA < 10 THEN XLL = .001
XMM = TBA: XJJ = EBA: XKK = ESG
IF TBA < 10 AND EBA < 60000 AND ESG > 30000 THEN XKK = 30000: IF TBA <
6 THEN XMM = 6: IF EBA < 30000 THEN XJJ = 30000
RETURN
ASSIGN VALUES TO INTERPOLATION PARAMETERS
CYL = 300000
XII = ABA * CYL ^ BBA
XLL = ASG * CYL ^ BSG
IF XII > .015 THEN XII = .015
IF XLL > .016 THEN XLL = .016
IF XLI < .001 AND TBA < 10 THEN XLL = .001
XMM = TBA: XJJ = EBA: XKK = ESG
IF TBA < 10 AND EBA < 60000 AND ESG > 30000 THEN XKK = 30000: IF TBA <
6 THEN XMM = 6: IF EBA < 30000 THEN XJJ = 30000
RETURN

Original Loadrate document
DIM D(160, 7), S0(160), DE(160, 7)
DIM AC(160), PFW(160), K1(160), W(160), BC(160), K1S(160)
DIM WL(160, 8), AM(160), PS(160): DIM SS(160), CK$(160)
DIM NS(160), H(160), SI(160), BA(160), NX(160)
DIM SS0(160), BBA(160), WW(160), WP(160), WOS(160)
DIM EI(160), E2(160), SL(160), Y(160), D7(160)
DIM DSUM(10), JSUM(10), FSUM(10), DF(10), FW(10), R(10), PRESS(10)
DIM WLOAD(10), X(10), OYR(160)
DIM DET(160), DES(160), NE(160), OE(160), TOTDE(160)
DIM RFINAL(160), passes(160), NY(160)
DIM XM(160), XL(160), XK(160), XJ(160), XI(160)
DIM Y(5), X(5), YLTEMP(5), YKTEMP(5)
DIM YJTEMP(5), YITEMP(5), C(5), DD(5)
REM To initialize FWD sensor spacing
X(1) = .01: X(2) = 12: X(3) = 24: X(4) = 36: X(5) = 48: X(6) = 60: X(7) = 72
GOSUB 6589
REM SELECT TEMP. CORRECTION OR STRUCTURAL ANALYSIS OR RUT DEPTH CALCULATION
IF SLECT1 = 2 THEN : CH01 = 0: GOSUB 10010
IF SLECT1 = 3 THEN : IF RUT1 <> 1 THEN : INIOPT = 2: GOSUB 12000
IF SLECT1 = 3 THEN : IF RUT1 = 1 THEN : INIOPT = 2: GOSUB 12400
IF SLECT1 = 4 THEN : CLS : GOTO 5980
GOSUB 6950: REM SELECTION TABLE — MAIN MENU
FOR I = 1 TO 160
    NY(I) = 0
    DYR(I) = 0
    PS(I) = 0
NEXT I
REM MAIN PROGRAM ————————————
'Initializing
CO = 0: 00 = 0
PRINT : PRINT
GOTO 1790
IF OP <> 1 AND 00 <> 1 THEN : GOTO 1740
IF 00 = 1 THEN : 00 = 0: GOTO 3570
'5
54
55 1740 IF 00 <> 0 THEN GOTO 3080
56 GOTO 2290
57
58 1790 'To read from file ————————————
59 'PRINT "" FILE NAME"
60 INPUT X$
61 'PRINT ""Please Wait ......""
62 OPEN ""I"", #1, X$
63 " INPUT #1. EQ"
64 " INPUT #1. ID"
65 " INPUT #1, C$"
66 " INPUT #1, FS$"
67 " INPUT #1, D$"
68 " INPUT #1, a$"
69 " INPUT #1, NC, NT, RX, RM, AL, PA, GR, ESAL1"
70 FOR NTT = 1 TO NT
71 " INPUT #1, NX(NTT)"
72 FOR I = 1 TO NX(NTT)
73 " INPUT #1, WL(NTT, I)"
74 NEXT I
75 NEXT
76 FOR I = 1 TO NC
77 " IF EO = 0 THEN : INPUT #1, NS(I), BA(I), D(I, 1), D(I, 2), D(I, 3), D(I, 4), D(I, 5)"
78 " IF EO = 1 THEN : INPUT #1, NS(I), BA(I), D(I, 1), D(I, 2), D(I, 3)"
79 " IF EO = 1 THEN : INPUT #1, D(I, 4), D(I, 5), D(I, 6), D(I, 7), PFW(I)"
80 NEXT
81 CLOSE
82 GOTO 2290
83
84
85 2290 REM To print input card images
86
87
88 "2320 OPEN ""SCRN:"" FOR OUTPUT AS #1"
89 REM ITERATE FOR THE NUMBER OF TRUCKS CONSIDERED
90 2340 FOR ITE = 1 TO NT
91 " IF ITE > 1 THEN : PRINT #1. : GOTO 2440"
92 "CLS : PRINT #1, ""TEXAS TRANSPORTATION INSTITUTE"
93 "PRINT #1, ""LOAD RATING OF LIGHT PAVEMENT"
94 "PRINT #1,"" JOB : ""; a$; "" ""INPUT FILE ""; X$; "")"
95 "PRINT #1,"" DISTRICT : ""; ID; "" COUNTY : ""; C$; "" ROAD : ""; F$"
96 "PRINT #1, ""ALLOWABLE RUT(INS):""; RX; "" RECORDED RUT(INS) : ""; RM"
97 "PRINT #1, ""ANNUAL TRAFFIC GROWTH RATE:""; GR; "" # OF YEARS : ""; AL; "" YEAR TRAFFIC : ""; ESAL1"
98 "PRINT #1, ""TOTAL NUMBER OF PASSES DURING ABOVE PERIOD:""; PA"
99 "PRINT #1, ""DATE : ""; D$; "" DYNAFLECT"
100 2440 NTT = ITE
101 "PRINT #1, ""TRUCK NO. : ""; NTT"
102 "PRINT #1, ""AXLE NUMBER SINGLE WHEEL/ESWL(LBS)"
103 "PRINT #1, ""To loop of the number of axles NX and number of trucks NTT"
104 FOR I = 1 TO NX(NTT)
105 "PRINT #1, TAB(5); I; TAB(20); WL(NTT, I)"
106 NEXT I
107 "PRINT #1, ""For more than one truck considered"
108 IF ITE > 1 THEN : GOTO 3010
109 "PRINT #1, ""ANNUAL TRAFFIC GROWTH RATE:""; GR; "" # OF YEARS : ""; AL; "" FIRST YEAR TRAFFIC : ""; ESAL1"
110 "PRINT #1, ""TOTAL NUMBER OF PASSES DURING ABOVE PERIOD:""; PA"
111 "IF EO = 0 THEN : PRINT #1, ""DATE : ""; D$; "" DYNAFLECT"
112 "IF EO = 1 THEN : PRINT #1, ""DATE : ""; D$; "" FALLING WEIGHT DEFLECTOMETER""
118       REM CONTINUE
119       "       PRINT #1," ""SECTION BASE
120       PRINT #1, ""NO. THICKNESS
121       IF EQ = 0 THEN : PRINT #1, ""(MILS)
122       IF EQ = 1 THEN : PRINT #1, ""(INS)
123       IF EQ = 0 THEN : PRINT #1, ""(MILS)
124       IF EQ = 1 THEN : PRINT #1, ""(INS)
125       IF EQ = 1 THEN : PRINT #1, ""(RADIAL DISTANCE)
126       IF EQ = 1 THEN : PRINT #1, ""(INS)
127       FOR I = 1 TO NC
128       "CALCULATION OF BASE THICKNESS
129       IF TBC = 1 THEN : BA(I) = NH
130       IF BA(I) <> 0 THEN : GOTO 2890
131       WW = (DE(I, 1) - DE(I, 7)) / DE(I, 7)
132       BA(I) = (SSO(I) / 1958 * WW)^ 1.85
133       BA(I) = INT((BA(I) * 10 + 5) / 10)
134       IF NH = 0 THEN : PPR = 0
135       IF PPR = 2 THEN : GOTO 3000
136       REM To print image for Dynafl ect
137       REM Check if hardcopy is needed
138       "       IF EQ = 0 THEN : PRINT #1. NS(I); TAB(12); BA(I); TAB(23);"
139       "       IF EQ = 0 THEN : PRINT #1. USING ""###.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.#####.##49
IF CO = 0 AND 00 = 0 THEN : GOTO 3570
169 IF CO = 0 AND 00 = 1 THEN : GOTO 3530
170
171 3530 IF OP <> 0 THEN : CLOSE : GOTO 1440
173
174 3570 'Check point for printing ________________
175
176 IF ITE <> 1 THEN : GOTO 3640
177 " IF TBC = 0 THEN : LOCATE 25, 1:
178 " INPUT ""DO YOU WANT HARDCOPY? O=NO 1=YES 2=RESULTS ONLY"": PR"
179 " IF TBC = 1 THEN : LOCATE 25, 1: INPUT ""DO YOU WANT HARDCOPY? O=NO 1=YES": PR"
180 " LOCATE 25, 1: PRINT """"
181 " IF PR = 2 THEN : PR = 1: PPR = 2: GOTO 3670
182 3640 IF PR = 1 THEN : CLOSE : OPEN ""LPT1:"" FOR RANDOM AS #1: GOTO 2340"
183 3650 REM Computation for Dynaflect readings  EQ is 0
184 3670 REM CONTINUE
185 IF ITE <> 1 THEN : GOTO 4790
186 IF DESIGN = 1 THEN : GOTO 4340
187 IF EQ = 0 THEN : PL = 1000
188 IF EQ = 1 THEN : PL = 10956.3
189 FOR I = 1 TO NC
190 " IF EQ = 1 THEN : SO(I) = PFW(I) / D(I, 1)"
191 " IF EQ = 1 THEN : DE(I, 1) = PL / SO(I)"
192 " REM D(I,7) may not be taken at 94.5" then need to change"
193 " IF (EQ = 1) THEN : GOSUB 7290: 'GEOMBASIN"
194 " IF (EQ = 1) THEN : DE(I, 7) = D7(I) * PL / PFW(I)"
195 " REM D7(I) IS FROM SUBROUTINE GEOMBASIN"
196 " REM To convert Dynaflect readings to DE ---
198 " REM To obtain stiffnesses
200 " IF EQ = 0 THEN : DE(I, 7) = 3.38075 * D(I, 5) ^ .639462"
201 " IF EQ = 0 THEN : DE(I, 7) = 4.5688721# * D(I, 5) ^ .578444"
202 " IF EQ = 0 THEN : DE(I, 1) = -7.24474 + (29.6906 * D(I, 1))"
203 " REM To convert readings to standard deflection DE ---
204 " IF EQ = 0 THEN : SO(I) = 86.0122 * EXP(.00187211# * PL / D(I, 1))"
205 " IF TBC = 0 THEN : SSOCI = SSOCI + SO(I)
206 " SI - is used only when low overall stiffness is encountered
208 'SI - is used only when low overall stiffness is encountered
209 SI(I) = -109.663 + 1.31393 * SO(I)
210 IF SO(I) <= 83.46183 THEN : SI(I) = SO(I)
211 H(I) = BA(I)
212 NEXT
213
216 REM To determine the type of subgrade ___________
217 " PRINT ""Please Wait ....... ""
218 " REM To convert readings to standard deflection DE ---
220 FOR I = 1 TO NC
REM To determine subgrade KIS-value
X = BA(I): Y = DE(I, 7): IF X > 18 THEN : X = 18
KIS(I) = (9556.651 + 370.3322 X) * DE(I, 7) ^ (-1.21665 + .016349 X)
IF KIS(I) < 5450 THEN : KIS(I) = 5450
IF KIS(I) > 62200 THEN : KIS(I) = 62200
LL1 = .4342945 * LOG(KIS(I))
A1 = 10 ^ (10.50698 - 1.97986 KIS(I))
A1 = A1 + A1 * BA(I) + A1 * BA(I) * BA(I)
K1(I) = (DE(I, 1) ^ A1) ^ (1 / A1)
IF K1(I) <= 3000 THEN : K1(I) = 3000
IF K1(I) > 60000 THEN : K1(I) = 60000
NEXT I

To determine material properties

FOR I = 1 TO NC
AV = 0
To calculate elastic modulus ECAL
FOR I = 1 TO NC
GOSUB 6600
NEXT I

To calculate the rut depth and number of passes
IF RUT1 = 1 THEN : INIOPT = 3: GOSUB 12400
IF RUT1 <> 1 THEN : INIOPT = 3: GOSUB 12000
FOR I = 1 TO NC
PS(I) = passes(I)
AM(I) = RFINAL(I)
IF GR < 0 THEN : GOTO 5097
REMAINING LIFE
NY(I) = (LOG(PS(I) * GR + ESAL1) - LOG(ESAL1)) / LOG(1 + GR)
TOTYR = TOTYR + NY(I)
IF NY(I) > NYMAX THEN : NYMAX = NY(I)
5097 AV = AV + PS(I)
NEXT I
IF GR < 0 THEN : GOTO 5100
FOR II = 1 TO NYMAX
FOR I = 1 TO NC
IF NY(I) >= II THEN : DYR(II) = DYR(II) + 1
NEXT I
DYR(II) = (DYR(II) / NC) * 100
IF DYR(II) = 100 THEN : S100 = II
CALCULATIONS OF ALLOWABLE AXLE LOADS

\[ AL1 = ((S100 / 10)^{.2432}) \times 18 \]
\[ AL2 = ((S100 / 10)^{.2467}) \times 33.5 \]
\[ AL3 = ((S100 / 10)^{.2534}) \times 48.25 \]

IF AL1 > 18 THEN: AL1 = 18
IF AL2 > 34 THEN: AL2 = 34
IF AL3 > 46 THEN: AL3 = 46

FOR PLOTS
\[ \min = \text{NS}(I) \]
\[ \max = \text{NS}(NC) \]

To print results
\[ \text{IF PR} = 1 \text{ THEN: OPEN ""LPT1:"": CLS: PRINT ""Printing ...""} \]
\[ \text{IF PR} = 0 \text{ THEN: OPEN ""SCRN:"": FOR OUTPUT AS #1: CLS} \]

IF GR < 0 THEN: PRINT #1. ""SECTION SUBGR ALLOWABLE DEPTH"" ""NO. OF RUT""

IF GR < 0 THEN: PRINT #1. ""SECTION SUBGR ALLOWABLE DEPTH"" ""NO. ELASTIC MODULUS E1-BASE E2-BASE/SUBB SUBGRADE PASSES (IN)""

IF GR >= 0 THEN: PRINT #1. ""SECTION SUBGR ALLOWABLE DEPTH"" ""NO. ELASTIC MODULUS E1-BASE E2-BASE/SUBB SUBGRADE PASSES (IN)""

FOR I = 1 TO NC

IF (EQ = 1) AND (NH <> 0) THEN: PRINT #1, NS(I); TAB(8); E1(I); E2(I); TAB(19); AM(I); TAB(74); ""DESIGN"";

IF (EQ = 1) AND (NH = 0) THEN: PRINT #1, NS(I); TAB(3); E1(I); TAB(13); USING ""#####""; E2(I);

IF (EQ = 1) AND (NH = 0) THEN: PRINT #1, TAB(25); USING ""###.#""; NY(I);

IF (EQ = 1) AND (NH = 0) AND (GR >= 0) THEN: PRINT #1, TAB(39); USING ""##.##""; PS(I);

IF (EQ = 1) AND (NH = 0) THEN: PRINT #1, USING ""#####.#""; AM(I)"
PRINT #1, ""Allowable Axle Load Limits (kips)"
PRINT #1, ""SINGLE TANDEM TRIPLE"
PRINT #1, 
IF GR < 0 THEN: GOTO 5740
PRINT 111.
祈算 Axle Load Limits (kips)
PRINT #1, SINGLE TANDEM TRIPLE'
PRINT #1.
IF S100 >= 10 THEN: PRINT #1, ""LEGAL LEGAL"
PRINT #1, ""ALLOWABLE AXIAL LOAD LIMITS:
PRINT #1, ""AVERAGE NUMBER OF PASSES TO CAUSE SPECIFIED RUT ("", RX;
PRINT #1, "")": AV
PRINT #1, ""NUMBER OF PASSES USED FOR RUT DEPTH CALCULATION : "", PA
PRINT #1, ""RUT DEPTH USED FOR REMAINING LIFE AND NUMBER OF PASSES
CALCULATION : "", RX
NEXT TIE
PRINT
CLOSE
PR = 0
PRINT
361 IF TBC = 0 THEN: GOTO 5960
DESIGN = 0
NH = 0
GOTO 2320
REM CONTINUE to get output menu OUTMENU
5960 GOSUB 7580: 'To OUTMENU
5980 END
CLS: KEY OFF
LOCATE 5, 1: PRINT ""TO CREATE OUTPUT FILE"
LOCATE 6, 1: INPUT ""ENTER OUTPUT FILE NAME AND <RET>:"", XX$'
CLS: KEY OFF
IF TEMRUT = 1 THEN: SLF = 1: GOTO 6536
LOCATE 7, 1: PRINT ""Select the format:"
LOCATE 9, 1: PRINT ""(1)"
LOCATE 10, 1: PRINT ""Sec. Thick E1(Base) E2(Subgr) Passes Rut depth"
LOCATE 10, 1: PRINT ""(2)"
LOCATE 11, 1: PRINT ""Sec. Thick E1(Base) E2(Subgr) Passes Rut depth Remaining life"
LOCATE 12, 1: PRINT ""(2)"
LOCATE 12, 1: PRINT ""Sec. Thick D1 D7 FWD load E1(Base) E2(Subgr) Passes"
LOCATE 12, 1: PRINT ""Sec. Thick D1 D5 FWD load E1(Base) E2(Subgr) Passes"
LOCATE 12, 1: PRINT ""Sec. Thick D1 D7 FWD load E1(Base) E2(Subgr) Passes Remaining life""
367 " IF (EQ = 0) AND (GR > 0) THEN: LOCATE 12, 1: PRINT ""Sect. Thick D1
D5 FWD load E1(Base) E2(Subgr) Passes Remaining life"
368 " LOCATE 20, 1: INPUT ""ENTER SELECTION AND <RET> :" ; SLF
369 IF (SLF < 1) OR (SLF > 2) THEN: GOTO 6612
370 "6536 OPEN "0'", #2, XX$: PRINT #2, ""
371 " CLS: PRINT #2, "" TEXAS TRANSPORTATION INSTITUTE"
372 " PRINT #2, ""LOAD RATING OF LIGHT PAVEMENT"
373 " PRINT #2, ""TRUCK NO. 1"
374 " PRINT #2, ""AXLE NUMBER SINGLE WHEEL/ESWL(LBS)"
375 " PRINT #2, TAB(5); ""9000"
376 " PRINT #2, ""ANNUAL TRAFFIC GROWTH RATE:"; RX; "" #OF YEARS:"; AL;
377 " IF RX = 0 THEN: PRINT #2, ""DATE:"; D$: "" FWD BASE E1-Base E2-Subgr
378 " IF GR < 1 THEN: GOTO 6562
379 " IF TEMRUT = 1 THEN: PRINT #2, ""SEC BASE E1-BASE E2-SUBGR NO. OF REMAINING LIFE"
380 " IF TEMRUT = 1 THEN: PRINT #2, ""SEC BASE E1-BASE E2-SUBGR NO. OF REMAINING LIFE"
381 " IF TEMRUT = 1 THEN: PRINT #2, ""ALLOWABLE RUT(INS):"; RX; "" RECORDED RUT(INS):"
382 " IF TEMRUT = 1 THEN: PRINT #2, ""TRUCK NO. 1"
383 " IF TEMRUT = 1 THEN: PRINT #2, ""AXLE NUMBER SINGLE WHEEL/ESWL(LBS)"
384 " IF TEMRUT = 1 THEN: PRINT #2, ""9000"
385 " IF TEMRUT = 0 AND SLF = 1 THEN: PRINT #2, ""FIRST YEAR TRAFFIC:"; ESAL1
386 " IF TEMRUT = 0 AND SLF = 1 THEN: PRINT #2, ""TOTAL NUMBER OF PASSES DURING ABOVE PERIOD:"; PA
387 " IF TEMRUT = 0 AND SLF = 1 THEN: PRINT #2, ""DATE:"; D$: "" DYNAPLECT"
388 " IF TEMRUT = 0 AND SLF = 1 THEN: PRINT #2, ""DATE:"; D$: "" FALLING WEIGHT DEFLECTOMETER"
389 " IF TEMRUT = 0 AND SLF = 1 THEN: PRINT #2, ""SEC BASE E1-Base E2-Subgr NO. OF REMAINING LIFE"
390 " REM CONTINUE
391 " IF GR < 0 THEN: GOTO 6662
392 " IF TEMRUT = 1 THEN: PRINT #2, "" TEMP. CORRECTED
393 " IF TEMRUT = 0 AND SLF = 1 THEN: PRINT #2, "" FIRST YEAR TRAFFIC:"; ESAL1
394 " IF TEMRUT = 0 AND SLF = 1 THEN: PRINT #2, "" SEC BASE E1-BASE E2-SUBGR NO. OF REMAINING LIFE"
395 " IF TEMRUT = 1 THEN: PRINT #2, "" SEC BASE E1-BASE E2-SUBGR NO. OF REMAINING LIFE"
396 " IF TEMRUT = 1 THEN: PRINT #2, "" # (IN) (PSI) Passes (IN) (YEARS)"
397 " IF TEMRUT = 1 THEN: PRINT #2, "" # (IN) (PSI) (PSI) Passes (IN) (YEARS)"
398 " IF TEMRUT = 1 THEN: FOR I = 1 TO NC: PRINT #2, USING ""1111.1111 1111.1111 1111.1111 ""
NSCI); BA(I); D1(I): DCI. 7); PFWCI); El(I); E2(I); PS(I); AM(I); NY(I): NEXT"
399 " IF TEMRUT = 1 THEN: FOR I = 1 TO NC: PRINT #2, USING ""1111.1111 1111.1111 1111.1111 ""
NSCI); BA(I); El(I); E2(I); PS(I); AM(I); NY(I): NEXT"
400 " IF SLF = 2 THEN: PRINT #2, "" FWD BASE E1-Base E2-Subgr NO. OF REMAINING LIFE"
401 " IF SLF = 2 AND EQ = 1 THEN: PRINT #2, "" SEC BASE D1 D7
FWD LOAD E1-BASE E2-SUBG NO. OF REMAINING LIFE"
402 " IF SLF = 2 AND EQ = 0 THEN: PRINT #2, "" SEC BASE D1 D5
FWD LOAD E1-BASE E2-SUBG NO. OF REMAINING LIFE"
403 " IF SLF = 2 THEN: PRINT #2, "" # (IN) (MILS) (MILS) (LBS) (PSI) (PSI) Passes (YEARS)"
404 " IF SLF = 2 AND EQ = 1 THEN: FOR I = 1 TO NC: PRINT #2, USING ""1111.1111 1111.1111 1111.1111 ""
NSCI); BA(I); D1(I); D1(7); PFW(I); El(I); E2(I); PS(I); AM(I); NY(I): NEXT"
405 " " IF SLF = 2 AND EQ = 0 THEN : FOR I = 1 TO NC: PRINT #2, USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I); NY(I): NEXT"
406 " " PRINT #2."n
407 " " IF SLF = 1 THEN : PRINT #2, "": * NUMBER OF PROPS AND REMAINING LIFE FOR SPECIFIED RUT DEPTH OF "": RX; ""in"":
408 " " IF SLF = 1 THEN : PRINT #2, "": ** RUT DEPTH FOR SPECIFIED NUMBER OF PASSES OF "": PA; "" IN"": AL; ""YEARS":
409 " " IF SLF = 2 THEN : PRINT #2, "": ** NUMBER OF PROPS AND REMAINING LIFE FOR SPECIFIED RUT DEPTH OF "": RX; ""in"":
410 " " PRINT #2."n
411 " " PRINT #2, "": Remaining Life(yrs) Cumulative % Sections"
412 FOR II = 1 TO NYMAX
413 " " PRINT #2. USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); E1(I); E2(I); PS(I); AM(I): NEXT"
414 " " PRINT #2."n
415 " " IF S100 >= 10 THEN : PRINT #2, "": LEGAL
416 " " IF S100 < 10 THEN : PRINT #2, USING ""##.## flfl.## flfNI flflllflffff #.##": AL1; AL2; AL3"
417 GOTO 6563
418 6562 "SAVE DATA WITHOUT REMAINING LIFE - GR<O"
419 " " IF TEMRUT = 1 THEN : PRINT #2, "": TEMP. CORRECTED
420 " " IF TEMRUT = 0 AND SLF = 1 THEN : PRINT #2, "":
421 " " IF TEMRUT = 0 AND SLF = 1 THEN : PRINT #2, "": SEC BASE E1-BASE E2-SUBGR NO. OF RUT DEPTH "":
422 " " IF TEMRUT = 1 THEN : PRINT #2, "": SEC BASE E1-BASE E2-SUBGR NO. OF RUT DEPTH "":
423 " " IF TEMRUT = 0 AND SLF = 1 THEN : PRINT #2, "": # (IN) (PSI) (PSI) PASSES (IN) (PSI) (PSI) PASSES (IN) "":
424 " " IF TEMRUT = 0 AND SLF = 1 THEN : FOR I = 1 TO NC: PRINT #2. USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); E1(I); E2(I); PS(I); AM(I): NEXT"
425 " " IF TEMRUT = 1 THEN : PRINT #2, "":
426 " " IF TEMRUT = 0 AND SLF = 1 THEN : PRINT #2, "": SEC BASE E1-BASE E2-SUBGR NO. OF REMAINING LIFE"":
427 " " IF TEMRUT = 1 THEN : PRINT #2, "": SEC BASE E1-BASE E2-SUBGR NO. OF REMAINING LIFE"":
428 " " IF TEMRUT = 0 AND SLF = 1 THEN : PRINT #2, "": # (IN) (MILS) (MILS) (LBS) (PSI) (PSI) PASSES (YEARS) "":
429 " " IF TEMRUT = 1 THEN : PRINT #2, "":
430 " " IF TEMRUT = 0 AND SLF = 1 THEN : FOR I = 1 TO NC: PRINT #2. USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); E1(I); E2(I); PS(I); AM(I): NEXT"
431 " " IF TEMRUT = 1 THEN : FOR I = 1 TO NC: PRINT #2. USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); E1(I); E2(I); PS(I); AM(I): NEXT"
432 " " IF S100 >= 10 THEN : PRINT #2, "": LEGAL IF S100 < 10 THEN : PRINT #2, USING ""##.## flfl.## flfNI flflllflffff #.##": AL1; AL2; AL3"
433 GOTO 6563
434 " " IF SLF = 2 AND EQ = 1 THEN : FOR I = 1 TO NC: PRINT #2, USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I): NEXT"
435 " " IF SLF = 2 AND EQ = 1 THEN : FOR I = 1 TO NC: PRINT #2, USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I): NEXT"
436 " " IF SLF = 2 AND EQ = 1 THEN : FOR I = 1 TO NC: PRINT #2, USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I): NEXT"
437 " " IF SLF = 2 AND EQ = 0 THEN : FOR I = 1 TO NC: PRINT #2, USING ""##.## flfl.## flfNI flflllflffff #.##": NS(I); BA(I); D(I, 1); D(I, 7); PFW(I); E1(I); E2(I); PS(I): NEXT"
PRINT #2, "TEMP. CORR. DATA"
PRINT #2, "IF TEC = 1 THEN: INPUT "DO YOU WANT TO SAVE TEMPERATURE AND MOISTURE CORRECTED DATA? 1=YES, 0=NO "; CH04
IF (CH04 <> 1 AND CH04 <> 0) THEN GOTO 6563
IF (CH04 = 0) GOTO 6584
PRINT #2, "INPUT CONDITIONS: WANTED CONDITIONS:", TEMPERATURE = ###.## deg. F, SUCTION = ###.## psi; HI: HL,
PRINT #2, "MODULUS CHANGE DUE TO TEMPER. DUE TO SUCTION"
PRINT #2, MODULUS (psi) (psi) (psi)
FOR I = 1 TO KKI
PRINT #2, USING "OECI); NE(I); DET(I); DES(I)"
NEXT I: CLOSE
6584 RETURN
6585 REM SUBROUTINE TO SELECT TEMP. CORRECTION OR STRUCTURAL ANALYSIS
6586 CLS : KEY OFF
LOCATE 3, 10
"LOADRATE - STRUCTURAL ANALYSIS OF LIGHT PAVEMENTS"
LOCATE 5, 10
PRINT "MAIN MENU"
LOCATE 7, 10
PRINT "SELECT THE OPERATION DESIRED:"
LOCATE 9, 10
PRINT "(1) Structural analysis of pavements"
LOCATE 10, 10
PRINT ( moduli, # of passes, rut depth, remaining life and allowable"
LOCATE 11, 10
PRINT "axle load calculations, and analysis of temperature and ""
LOCATE 12, 10
PRINT "moisture effects on base material )"
LOCATE 13, 10
PRINT "(2) Temperature correction of base modulus only"
LOCATE 14, 10
PRINT "(3) Rut depth calculation only"
LOCATE 15, 10
PRINT "(4) Exit program"
LOCATE 20, 10
" ENTER SELECTION AND <RET> : " SLECT1"
IF (SLECT1 < 1) OR (SLECT1 > 4) THEN GOTO 6586
RETURN
6600 'SUBROUTINE for elastic modulus calculations ECAL

'Initializing various values - all changes will be made here
K01 = .6: REM SIGR/SIGZ ratio for base course
K02 = .82: REM SIGR/SIGZ ratio for subgrade
'EMOD will be calculated at test load
AREA = 3.1415927# * 5.9 * 5.9

FOR I = 1 TO NC
   IF EO = 0 THEN PFW(I) = 9000: REM DYNAFLECT - CALCULATE FOR WHEEL LOAD

   SL(I) = PFW(I) / AREA
   ZZ1 = 1 + BA(I) / 2: REM Elastic modulus for base course depth
   ZZ2 = 7! + BA(I): REM Elastic Modulus for subgrade depth
   ALS1 = 2.416966 * SL(I) ^ .477234
   ALS2 = .6007171 * KIS(I) ^ .058992
   ALH = 1.338562 * (1 + BA(I))^ -.145887
   RHB = .59614133# * K1S(I) ^ .593462E-02
   RHH = 3.0385722# * (1 + BA(I)) ^ .419624
   RHS = 272.63462# * K1S(I) ^ -.58427
   NSG = -1.01866 + 1.239625 / (KIS(I) * 10 ^ -3) + .000013 * (KIS(I) * 10 ^ -3) ^ 3 + .061115 * (KIS(I) * 10 ^ -3) - .00165 * (KIS(I) * 10 ^ -3) ^ 2
   ALPHA = ALS1 * ALS2 * ALH
   RHO = (1 + BA(I)) * LOG(RHH * RSH / RHB / ALPHA) / LOG(2.7182818#)
   ER1 = ALPHA * EXP((-RHO / ZZ1) ^ 1) + 5.9
   ER2 = ALPHA * EXP((-RHO / ZZ2) ^ 1)) + 5.9
   SIGZ1 = PFW(I) / (3.1415927# * ER1 * ER1)
   SIGZ2 = PFW(I) / (3.1415927# * ER2 * ER2)
   SIGR1 = K01 * SIGZ1
   SIGR2 = K02 * SIGZ2
   E2(I) = KIS(I) * (ABS(SIGZ2 - SIGR2)) ^ NSG
   El(I) = K1I(I) * (SIGZ1 + 2 * SIGR1) ^ .33
NEXT
RETURN

REM SUBROUTINE MAINMENU

REM SUBROUTINE Input for geometric regression GEOMINPUT

' for non-standard FWD sensor spacing
GR = 1:CLS:REM To initialize and activate GEOMBASIN
'Standard FWD sensor spacing for Texas SDHPT is as follows
LOCATE 5, 10: PRINT "FWD Sensor #1 #2 #3 #4 #5 #6 #7"
LOCATE 6, 10: PRINT "Dist.(ins) from load": 0; 12; 24; 36; 48; 60; 72"

LOCATE 7, 10: PRINT "Type in new spacing ..."
LOCATE 8, 10: INPUT "Sensor #1 at (ins) " ; X(1)
LOCATE 9, 10: INPUT "Sensor #2 at (ins) " ; X(2)
LOCATE 10, 10: INPUT "Sensor #3 at (ins) " ; X(3)
LOCATE 11, 10: INPUT "Sensor #4 at (ins) " ; X(4)
LOCATE 12, 10: INPUT "Sensor #5 at (ins) " ; X(5)"
539 " LOCATE 13, 10: INPUT ""Sensor #6 at (ins) "": X(6)"
540 " LOCATE 14, 10: INPUT ""Sensor #7 at (ins) "": Y(7)"
541 IF X(1) = 0 THEN : X(1) = .01
542 RETURN
543
544 7290 REM SUBROUTINE geometric regression GEOMBASIN
545  " J = 0: K = 0: L = 0: M = 0: R2 = 0
546  " N = 4
547 IF D(I, 7) = 0 THEN : N = 3
548 IF X(7) = 0 THEN : N = 3
549 FOR II = 4 TO (3 + N)
550 " X = X(II): Y = D(I, II)"
551 " X = LOG(X): Y = LOG(Y)
552 J = J + X: K = K + Y: L = L + X * X: M = M + Y * Y
553 R2 = R2 + X * Y
554 NEXT II
555 B = (N * R2 - K * J) / (N * L - J * J)
556 a = (K - B * J) / N
557 J = B * (R2 - J * K / N)
558 M = M - K * J / N
559 K = M - J
560 R2 = J / M
561 " print ""D7@94.49 ="":exp(a)*94.49^b"
562 " D7(I) = EXP(a) * 94.49 ^ B
563 " RETURN
564
565 7580 REM SUBROUTINE OUTMENU
566 "CLS : KEY OFF
567 " LOCATE 2, 10: PRINT "" OUTPUT MENU ""
568 " LOCATE 4, 10: PRINT "" Select Optional Output Format"
569 " LOCATE 6, 10: PRINT "" (1) Plot Base Course Elastic Modulus"
570 " LOCATE 7, 10: PRINT "" (2) Plot Subgrade Elastic Modulus"
571 " LOCATE 8, 10: PRINT "" (3) Plot Number of Vehicle Passes"
572 " LOCATE 9, 10: PRINT "" (4) Plot Remaining Life"
573 " LOCATE 10, 10: PRINT "" (5) Temperature Correction of Base Modulus"
574 " LOCATE 11, 10: PRINT "" (5.1) Rut depth, Passes and Remaining Life"
575 " LOCATE 12, 10: PRINT "" for corrected moduli"
576 " LOCATE 13, 10: PRINT "" (6) Save the Output as a file"
577 " LOCATE 14, 10: PRINT "" (7) Exit to Main Menu"
578 IF SLO = 7 OR SLO = 0 THEN : CLS : SLO = 0: TEMRUT = 0: TEC = 0: GOTO 730
579 IF SLO = 6 THEN : GOSUB 6510: GOTO 7580: 'To OUTFILE and back
580 IF SLO = 5 THEN : TEC = 1: GOSUB 10010
581 IF SLO = 5.1 THEN : FOR I = 1 TO 100: NY(I) = 0: NEXT
582 IF SLO = 5.1 THEN : FOR I = 1 TO 100: DYR(I) = 0: NEXT
583 IF SLO = 5.1 THEN : FOR I = 1 TO 100: PS(I) = 0: NEXT
584 IF SLO = 5.1 THEN : TEMRUT = 1: GOSUB 5080
585 " GOTO 7800
586 END
587
588 7800 REM PROGRAM SBRPLOT
589"
PRCHK = 0
IF PT = 0 THEN : PT = 1: REM 1=Epson/Panasonic 2=Mannesmann Tally
REM ******** THIS PART OF THE PROGRAM SHOULD REMAIN UNTouched
*************
CLS
PRINT

REM ************* plot at low resolution graphics
PRINT
RD = 0
PRINT

" 'INPUT'' TITLE : '';T$"
" IF SLO = 1 THEN : t$ = ""ELASTIC MODULUS OF BASE COURSE""
" IF SLO = 2 THEN : t$ = ""ELASTIC MODULUS OF SUBGRADE""
" IF SLO = 3 THEN : t$ = ""NUMBER OF PASSES FOR RUT DEPTH""
" IF SLO = 4 THEN : t$ = ""REMAINING LIFE"
" PR "plotting : ";T$"
" 'INPUT "Starting Section Number: "; MINX"
" 'INPUT "Ending Section Number: "; MAXX"
PRINT

MINY = 0

" IF SLO = 1 THEN : INPUT "Minimum E1 [Default=0]: "; MINY"
" IF SLO = 2 THEN : INPUT "Maximum MINY [Default=100000]: "; MAXY"
" IF SLO = 1 AND MAXY = 0 THEN : MAXY = 100000!
" IF SLO = 2 THEN : INPUT "Minimum E2 [Default=0]: "; MINY"
" IF SLO = 2 THEN : MAXY = 25000: INPUT "Maximum E2 [Default=25000]: "; MAXY"
" IF SLO = 2 AND MAXY = 0 THEN : MAXY = 25000
" IF SLO = 3 THEN : INPUT "Minimum PASSES [Default=0]: "; MINY"
" IF SLO = 3 THEN : INPUT "Maximum PASSES [Default=0.5E6]: "; MAXY"
" IF SLO = 3 AND MAXY = 0 THEN : MAXY = 500000!
" IF SLO = 4 THEN : INPUT "Minimum REMAINING LIFE [Default=0]: "; MINY"
" IF SLO = 4 THEN : INPUT "Maximum REMAINING LIFE [Default=24]: "; MAXY"
" IF SLO = 4 AND MAXY = 0 THEN : MAXY = 24

PRINT
SX = 2: 'Number of alphabet for NC
IF SX = 0 THEN : SX = 1
PRINT

" 'INPUT "hardcopy? 1=yes 0=no ";PR"
PR = 1
XSCALE = 60 / (MAXX - MINX)
YSCALE = 20 / (MAXY - MINY)
YORG = 22 + MINY * YSCALE
XORG = 10 - MINX * XSCALE
KEY OFF
CLS

PRINT

8300 REM these are ASCII for the IBM
IF PRCHK = 0 THEN : PR1 = 196: PR2 = 179: PR3 = 218: PR4 = 191: PR5 = 192: PR6 = 217
FOR COL = 11 TO 69
   LOCATE 2, COL: PRINT CHR$(PR1)"
LOCATE 22. COL: PRINT CHR$(PR1)"
NEXT
FOR ROW = 3 TO 21
  "    LOCATE ROW, 10: PRINT CHR$(PR2)"
  "    LOCATE ROW, 70: PRINT CHR$(PR2)"
NEXT
"    LOCATE 2, 10: PRINT CHR$(PR3)"
"    LOCATE 2, 70: PRINT CHR$(PR4)"
"    LOCATE 22, 10: PRINT CHR$(PR5)"
"    LOCATE 22, 70: PRINT CHR$(PR6)"
IF PRCHK = 0 THEN: PR7 = 193: PR8 = 195
IF PRCHK = 0 THEN: PR9 = 180: PR10 = 194
FOR I = 1 TO 9
  "    LOCATE 22, (10 + 6 * I): PRINT CHR$(PR7)"
  "    LOCATE (2 + 2 * I), 10: PRINT CHR$(PR8)"
  "    LOCATE (2 + 2 * I), 70: PRINT CHR$(PR9)"
  "    LOCATE 2, (10 + 6 * I): PRINT CHR$(PR10)"
NEXT
IF PRCHK = 1 AND PT = 1 THEN: GOSUB 9350
IF PRCHK = 1 AND PT = 2 THEN: GOSUB 9210
LOCATE 1, 10: PRINT "t$"
LOCATE 1, 55: PRINT X$"
XINT = (MAXX - MINX) / 10
YINT = (MAXY - MINY) / 10
LOCATE 23, 8: PRINT (MINX)"
LOCATE 23, 20: PRINT (MINX + 2 * XINT)"
LOCATE 23, 32: PRINT (MINX + 4 * XINT)"
LOCATE 23, 44: PRINT (MINX + 6 * XINT)"
LOCATE 23, 56: PRINT (MINX + 8 * XINT)"
LOCATE 23, 69: PRINT (MINX + 10 * XINT)"
'LOCATE 24,30 :PRINT ""Section Numbers"
LOCATE 2, 1: PRINT MAXY"
LOCATE 6, 1: PRINT INT((MAXY - YINT * 2) * 100) / 100"
LOCATE 10, 1: PRINT INT((MAXY - YINT * 4) * 100) / 100"
LOCATE 14, 1: PRINT INT((MAXY - YINT * 6) * 100) / 100"
LOCATE 18, 1: PRINT INT((MAXY - YINT * 8) * 100) / 100"
LOCATE 22, 1: PRINT MINY"
IF SLO = 1 THEN: LOCATE 11, 2: PRINT ""E1""
IF SLO = 1 THEN: LOCATE 12, 2: PRINT ""(psi)"
IF SLO = 2 THEN: LOCATE 11, 2: PRINT ""E2"
IF SLO = 2 THEN: LOCATE 12, 2: PRINT ""(psi)"
IF SLO = 3 THEN: LOCATE 11, 2: PRINT ""No.of"
IF SLO = 3 THEN: LOCATE 12, 2: PRINT ""Passes"
IF SLO = 4 THEN: LOCATE 11, 2: PRINT ""Remain.""
IF SLO = 4 THEN: LOCATE 12, 2: PRINT ""Life"
IF SLO = 4 THEN: LOCATE 13, 2: PRINT ""(years)"
REM CONTINUE to exit loop
'get points
IF PRCHK = 1 THEN: LOCATE 25, 1: PRINT SPC(70);""
IF PRCHK = 1 THEN: LOCATE 25, 1: PRINT ""RE-Plotting . . . . . ;"
FOR I = 1 TO NC
  X = NS(I)
  IF SLO = 1 THEN: Y = E1(I)
  IF SLO = 1 AND Y > MAXY THEN: Y = MAXY
  IF SLO = 2 THEN: Y = E2(I)
IF SLO = 2 AND Y > MAXY THEN : Y = MAXY
IF SLO = 3 THEN : Y = PS(I)
IF SLO = 3 AND Y > MAXY THEN : Y = MAXY
IF SLO = 4 THEN : Y = NY(I)
IF SLO = 4 AND Y > MA;Y THEN : Y = MAXY

' display point and coord

SCY = YORG - INT(Y * YSCALE); SCX = XORG + INT(X * XSCALE)
IF SCX < 10 THEN GOTO 9050
IF SCX > 70 THEN GOTO 9050
IF SCY < 2 THEN GOTO 9050
IF SCY > 22 THEN GOTO 9050
LOCATE SCY, SCX: PRINT CHR$(42); GOTO 9050

IF SCX < 10 THEN GOTO 9050
IF SCX > 70 THEN GOTO 9050
IF SCY < 2 THEN GOTO 9050
IF SCY > 22 THEN GOTO 9050
LOCATE 25, 1: PRINT SPC(75);"

IF PRCHK = 1 THEN : LOCATE 25, 1: INPUT "PrtSc? <RET> if NO"; NN$
IF PRCHK = 1 THEN : GOTO 9150
LOCATE 25, 1: PRINT "PRESS ENTER TO CONTINUE ";
INPUT ; NN$
IF PRCHK = 1 THEN : GOTO 9150
IF PR = 1 AND PT = 1 THEN : GOSUB 9350: GOTO 8300
IF PR = 1 AND PT = 2 THEN : GOSUB 9210: GOTO 8300
IF PR = 1 AND PT = 3 THEN : GOTO 9150: 'No adjustment required
9150 REM CONTINUE
CLS
GOTO 7580: 'To return to OUTMENU

9210 REM SUBROUTINE MT180 to replot the graph for the mannesmann tally
PR1 = 157: REM FROM IBM 196
PR2 = 156: REM FROM IBM 179
PR3 = 134: REM FROM IBM 218
PR4 = 149: REM FROM IBM 191
PR5 = 153: REM FROM IBM 192
PR6 = 154: REM FROM IBM 217
PR7 = 158: REM FROM IBM 193
PR8 = 150: REM FROM IBM 195
PR9 = 151: REM FROM IBM 180
PR10 = 152: REM FROM IBM 194
PRCHK = 1
RETURN

9350 REM SUBROUTINE EPSON to replot the graph for the epson/panasonic
PR1 = 45: REM FROM IBM 196
PR2 = 124: REM FROM IBM 179
PR3 = 46: REM FROM IBM 218
PR4 = 46: REM FROM IBM 191
PR5 = 46: REM FROM IBM 192
PR6 = 46: REM FROM IBM 217
PR7 = 43: REM FROM IBM 193
PR8 = 43: REM FROM IBM 195
PR9 = 43: REM FROM IBM 180
PR10 = 43: REM FROM IBM 194
PRCHK = 1
RETURN
OPEN "0", #1, "B:FM421.RES": FOR II = 1 TO 26: PRINT #1, WS(II), BA(II), D(II, 1), D(II, 7), K1(II), K1S(II), E1(II), E2(II), SS(II), BC(II), PSCII), AM(II): NEXT:
CLOSE

OPEN "0", #1, "B:FM421.SAN": FOR II = 1 TO 26: PRINT #1, PSCII), AM(Il): NEXT:
CLOSE

10010 CLS
10130 REM SUBROUTINE for temperature correction TEM
P 'THIS SUBROUTINE PREDICTS THE MODULUS OF GRANULAR MATERIALS
P 'AT DIFFERENT TEMPERATURES AND MOISTURES BASED ON A
P 'MICROMECHANICAL APPROACH
P
P '=======INPUT========
P
PRINT: PRINT
PRINT TAB(25); "TYPE OF BASE MATERIAL:"": PRINT
PRINT TAB(25); " 1 = CRUSHED LIMESTONE"":"" P
PRINT TAB(25); " 1 = LIME ORE GRAVEL"":"" P
PRINT TAB(25); " 1 = IRON ORE GRAVEL"":"" P
PRINT TAB(25); " 2 = RIVER GRAVEL"":"" P
PRINT TAB(25); " 2 = CALICHE"":"" P
PRINT TAB(25); " 3 = CALICHE GRAVEL"":"" P
PRINT TAB(25); " 3 = SANDSHELL"":"" P
PRINT TAB(25): : INPUT "SELECT YOUR OPTION ": MTYPE
IF (MTYPE <= 0 OR MTYPE > 3) GOTO 10130
IF (MTYPE = 1) THEN ALP = 5 * 10 ^ (-6): E = 6.4 * 10 ^ 6: U = .17: K1 = 14000: K2 = .4
IF (MTYPE = 2) THEN ALP = 6.5 * 10 ^ (-6): E = 7.8 * 10 ^ 6: U = .2: K1 = 24000: K2 = .37
IF (MTYPE = 3) THEN ALP = 5 * 10 ^ (-6): E = 8.534001 * 10 ^ 6: U = .17: K1 = 7210: K2 = .45

10260 CLS : PRINT : PRINT

PRINT TAB(25); "PROPERTIES OF THE MATERIAL SELECTED:"": PRINT
PRINT TAB(25): : INPUT "DO YOU WANT TO CHANGE? O=NO 1=YES ": CHOICE
IF (CHOICE <> 1 AND CHOICE <> 0) THEN GOTO 10350
IF (CHOICE = 0) GOTO 10460

PRINT : PRINT TAB(20); "" WHICH LINE ": WL
PRINT TAB(20): : INPUT ""INPUT NEW VALUE ": NV
IF (WL = 1) THEN ALP = NV
IF (WL = 2) THEN E = NV
IF (WL = 3) THEN U = NV
IF (WL = 4) THEN K1 = NV
IF (WL = 5) THEN K2 = NV
GOTO 10260

10460 CLS : PRINT : PRINT : PRINT TAB(25):
INPUT ""UNIT WEIGHT OF MATERIAL (pcf) ": UW
CLS

62
IF SELECT1 = 2 THEN : GOTO 10478
CH01 = 1
GOTO 10500
10478  KK1 = 1
PRINT : PRINT TAB(25);
"INPUT ""MODULUS OF MATERIAL (psi) " : "', EI"
10500 PRINT : PRINT "10510 PRINT TAB(20); ""CONDITIONS AT WHICH THE MODULUS IS OBTAINED: ""
PRINT TAB(25); : INPUT ""TEMPERATURE, Ti (40 deg.F < Ti < 110 deg.F) : "', TTEMP"
10540 PRINT TAB(25); : INPUT ""SUCTION, Hi (-145 psi < Hi < 0 psi) :
10590 PRINT TAB(25); : INPUT ""TEMPERATURE, To (40 deg.F < To < 110 deg.F) :
IF (CH01 = 1) THEN KK1 = NC
FOR I = 1 TO KK1
IF CH01 = 1 THEN EI El(I)
IF CH01 = 1 THEN K1 K1(I)
IF CH01 = 1 THEN K2 = .33

'=======CALCULATION========
VOID = 1 / (UW / 2.67 / 62.4) - 1
N = VOID / (1 + VOID) 'POROSITY
X = (.4764 - N) / (.4764 - .2595)
IF (X < 0) THEN X = 0
IF (X > 1) THEN X = 1
PI = 3.1415927#
PRO1 = 3 / 4 * (1 - U^2) / E
'==K2 IS THE POWER IN THE EQUATION E=K1*TETHA^K2==
'K2 = 1 / 3
THETA = 10 ^ (LOG(EI / K1) / LOG(10) / K2)
DTEMP = RTEMP - TEMPERATURE REFERENCE
DT = ABS(DTEMP)
IF (DTEMP = 0) THEN NE = EI: GOTO 10800
DELV = DT * ALP * 3
PSC = X / 2 ^ (1 / 2) / PRO1 * (1 / 3 * DELV) ^ (3 / 2)
PFC = (1 - X) / 4 / PRO1 * (1 / 3 * DELV) ^ (3 / 2)
PT = (PSC + PFC) * DTEMP / DT
DET(I) = K1 * K2 * THETA ^ (K2 - 1) * PT
10800 PS = -(HL - HI) * .13
DES(I) = K1 * K2 * THETA ^ (K2 - 1) * PS
TOTDE(I) = DET(I) + DES(I)
NE(I) = EI + TOTDE(I)
OE(I) = EI
NEXT I

'=======OUTPUT========
CLS
PRINT: PRINT: PRINT TAB(10):
PRINT "" INPUT CONDITIONS: WANTED CONDITIONS:"
PRINT
PRINT USING "" TEMPERATURE = ###.## deg. F TEMPERATURE = ###.## deg. F""; TTEMP; RTTEMP
PRINT USING "" SUCTION = ####.## psi SUCTION = ####.## psi""; HI; HL
PRINT
PRINT INPUT WANTED CONDITIONS:
PRINT ""
PRINT USING "" TEMPERATURE IN""; 1
PRINT USING "" SUCTION ""; HI; HL
PRINT
FOR I = 1 TO KK1
PRINT USING ""; OE(I); NE(I); DET(I); DES(I)
NEXT I
PRINT: PRINT
INPUT "" PRESS RETURN TO CONTINUE........""; PRESS
CLS
IF SLECT1 <> 2 THEN: GOTO 10988
PRINT: PRINT: PRINT "" 1. ANOTHER TEMPERATURE CORRECTION """
PRINT: PRINT "" 2. EXIT TO MAIN MANUAL """
PRINT: PRINT INPUT "" Select the Operation Desired: ""; CH02
IF CH02 = 1 THEN: GOTO 10010
GOTO 730
10988 PRINT: PRINT: PRINT "" 1. EXIT TO MAIN MENU """
PRINT: PRINT "" 2. EXIT TO OUTPUT MENU """
PRINT: PRINT INPUT "" Select the Operation Desired: ""; CH03
IF CH03 = 1 THEN: GOTO 730
GOTO 7580
RETURN
12000 '****************************************************
12000 ' RUT DEPTH PREDICTION SUBROUTINE FOR LOW-VOLUME ROADS
12000 ' BY
12000 ' K. A. S. YAPA
12000 '********************************************************************************
12000 ' Pavement Systems, Texas Transportation Institute."
12000 ' " TTI Building, Texas A & M University."
12000 ' " College Station, Texas 77843."
12000 ' (409)-845-9910.
12000 ' 24th MARCH 1988.
12000 '
12000 'This subroutine predicts the rut depth of a low-volume road by using a data
12000 'base of rut depths calculated by the Mechano-lattice program. A
12000 'multi-dimensional polynomial interpolation routine is used to
12000 'interpolate among the input parameters. Required inputs are the
12000 'resilient modulus of both the base course and the subgrade
12000 'layers, the material classification of the subgrade layer
12000 'and the thickness of the base layer.
12000 'Optionally, laboratory data from a permanent deformation test for
12000 'each material layer can be input, in place of the material classification.
12000 'DIM rut(2, 3, 3, 3, 3)"
923 ' LOAD THE DATABASE INTO AN ARRAY
924 FOR I = 1 TO 2
925 FOR J = 1 TO 3
926 FOR K = 1 TO 3
927 FOR L = 1 TO 3
928 " READ rut(I, J, K, L, 1), rut(I, J, K, L, 2), rut(I, J, K, L, 3)"
929 NEXT L: NEXT K: NEXT J: NEXT I
930 ' READ THE ORIGINAL PARAMETER LEVELS OF THE DATABASE
931 FOR I = 1 TO 3
932 " READ XM(I), XL(I), XK(I), XJ(I), XI(I)"
933 NEXT I
934 ' GOTO SUBROUTINE INPUT1 ............................
935 RUT1 = 1
936
940 12400 GOSUB 14240
941 ' LOOP TO CALCULATE RUT DEPTHS FOR EACH FWD SECTION
942 ' IF INIOPT = 2 THEN NC = 1
943 ' FOR INC = 1 TO NC
944 IF TEMRUT = 1 THEN : E1(INC) = NE(INC)
945 REM GOTO SUBROUTINE INPUT2 ............................
946 GOSUB 14850
947 ' GOTO INPUT1 IF ANY CORRECTIONS ARE NEEDED
948 IF CORR = 1 THEN GOSUB 14240
949 CLS
950 ' PRINT "Calculating............."
951 ' SELECT PARAMETERS FOR INTERPOLATION
952 ' FOR I = 1 TO 2
953 FOR J = 1 TO 3
954 FOR K = 1 TO 3
955 FOR L = 1 TO 3
956 ' M = BASE THICKNESS
957 " Y1(M) = rut(I, J, K, L, M)"
958 X1(M) = XM(M): NEXT M
959 ' CALL THE INTERPOLATION ROUTINE
960 NUM = 3: X = XMM: GOSUB 13700
961 YLTEMP(L) = Y: XMFLAG = XFLAG
962 NEXT L
963 ' L = SUBGRADE RUTTING POTENTIAL
964 ' FOR LL = 1 TO 3: X1(LL) = XL(LL): Y1(LL) = YLTEMP(LL): NEXT LL
965 NUM = 3: X = XLL: GOSUB 13700
966 YKTEMP(K) = Y: XLFLAG = XFLAG
Desired: K - SUBGRADE RESILIENT MODULUS

FOR KK = 1 TO 3: XI(KK) = XK(KK): Y1(KK) = YKTEMP(KK): NEXT KK
NUM = 3: X = XK: GOSUB 13700
YJTEMP(J) = Y: XJFLAG = XFLAG
NEXT J

J - BASE RESILIENT MODULUS

FOR JJ = 1 TO 3: XI(JJ) = XJ(JJ): Y1(JJ) = YJTEMP(JJ): NEXT JJ
NUM = 3: X = XJ: GOSUB 13700
YITEMP(I) = Y: XJFLAG = XFLAG
NEXT I

I - BASE RUTTING POTENTIAL

FOR II = 1 TO 2: XI(II) = XI(II): Y1(II) = YITEMP(II): NEXT II
NUM = 2: X = XI: GOSUB 13700
RUTCAL = Y * CYL / 300000!: XFLAG = XFLAG

GOTO SUBROUTINE OUTPUT

GOSUB 15680
NEXT INC
IF INIOPT <> 2 THEN : GOTO 13685
PRINT : PRINT : PRINT
"1. ANOTHER RUT DEPTH CALCULATION """
"2. EXIT TO MAIN MANUAL """
PRINT : PRINT : INPUT : INPUT Select the Operation

Desired: CH05"

IF CH05 = 1 THEN : GOTO 12400
GOTO 730

DATA BASE OF RUT DEPTHS

```
1  2  3
DATA  0.2609  0.2804  0.3191
DATA  0.2216  0.2152  0.2223
DATA  0.1831  0.1502  0.1253
DATA  0.2705  0.3030  0.3431
DATA  0.2451  0.2491  0.2556
DATA  0.2168  0.1951  0.1680
DATA  0.3674  0.4519  0.5286
DATA  0.3503  0.4172  0.4580
DATA  0.3331  0.3831  0.3882
DATA  0.2702  0.2912  0.3019
DATA  0.2249  0.2185  0.2002
DATA  0.1797  0.1458  0.0991
DATA  0.2804  0.3099  0.3549
DATA  0.2473  0.2490  0.2600
DATA  0.2137  0.1882  0.1652
DATA  0.3900  0.4610  0.5322
DATA  0.3688  0.4231  0.4561
DATA  0.3475  0.3842  0.3798
DATA  0.2946  0.3130  0.3110
DATA  0.2344  0.2263  0.2031
DATA  0.1750  0.1403  0.0837
```
1038 " DATA 0.3013 , 0.3356 , 0.3755"
1039 " DATA 0.2548 , 0.2600 , 0.2687"
1040 " DATA 0.2087 , 0.1842 , 0.1618"
1041 " DATA 0.3977 , 0.4840 , 0.5424"
1042 " DATA 0.3750 , 0.4343 , 0.4541"
1043 " DATA 0.3498 , 0.3834 , 0.3670"
1044 " DATA 0.1613 , 0.2376 , 0.3056"
1045 " DATA 0.1258 , 0.1727 , 0.2086"
1046 " DATA 0.0934 , 0.1082 , 0.1119"
1047 " DATA 0.1901 , 0.2528 , 0.3247"
1048 " DATA 0.1588 , 0.1993 , 0.2374"
1049 " DATA 0.1280 , 0.1457 , 0.1500"
1050 " DATA 0.2607 , 0.3681 , 0.4915"
1051 " DATA 0.2453 , 0.3355 , 0.4217"
1052 " DATA 0.2300 , 0.3038 , 0.3523"
1053 " DATA 0.1951 , 0.2524 , 0.2767"
1054 " DATA 0.1491 , 0.1794 , 0.1747"
1055 " DATA 0.1039 , 0.1072 , 0.0732"
1056 " DATA 0.1957 , 0.2643 , 0.3400"
1057 " DATA 0.1628 , 0.2037 , 0.2453"
1058 " DATA 0.1296 , 0.1433 , 0.1505"
1059 " DATA 0.2920 , 0.4006 , 0.5025"
1060 " DATA 0.2713 , 0.3648 , 0.4262"
1061 " DATA 0.2508 , 0.3230 , 0.3502"
1062 " DATA 0.2254 , 0.2785 , 0.2834"
1063 " DATA 0.1652 , 0.1918 , 0.1736"
1064 " DATA 0.1056 , 0.1059 , 0.0643"
1065 " DATA 0.2256 , 0.2974 , 0.3642"
1066 " DATA 0.1790 , 0.2216 , 0.2574"
1067 " DATA 0.1328 , 0.1461 , 0.1506"
1068 " DATA 0.3322 , 0.4287 , 0.5197"
1069 " DATA 0.3034 , 0.3786 , 0.4327"
1070 " DATA 0.2749 , 0.3287 , 0.3456"
1071
1072 'INPUT PARAMETERS USED IN CREATING THE DATA
1073
1074 " DATA 18, 0.0100, 25000, 100000, 0.0075"
1075 " DATA 12, 0.0060, 15000, 70000, 0.0025"
1076 " DATA 6, 0.0020, 5000, 40000, 0.0000"
1077 END
1078 13685 RETURN
1079 '********************************************************************
1080 13700 'SUBROUTINE FOR POLYNOMIAL INTERPOLATION
1081 '********************************************************************
1082 ' X - VALUE OF THE PARAMETER
1083 ' X1(I) - PARAMETER VALUES USED IN THE DATA
1084 ' Y1(I) - RUT DEPTHS FROM DATA BASE CORRESPONDING TO PARAMETER
1085 ' Y - INTERPOLATED VALUE
1086 ' NUM - NUMBER OF LEVELS OF THE PARAMETER
1087 .
1088 NS = 1
1089 DIF = ABS(X - X1(1))
1090 FOR a = 1 TO NUM
1091 DIFT = ABS(X - X1(a))
1092 .
1093 .
1094 .
1095 IF DIFT < DIF THEN : NS = a: DIF = DIFT
C(a) = Y1(a); DD(a) = Y1(a)
NEXT a
XFLAG = 0

'ENFORCE LIMITS ON EXTRAPOLATION (MAXIMUM = 1.5 * DIFFERENCE BETWEEN TWO
'CONSECUTIVE PARAMETER LEVELS)

IF DIF > 1.5 * (ABS(X1(1) - X1(2))) THEN GOTO 13940 ELSE GOTO 13980
DIF = 1.5 * (ABS(X1(1) - X1(2)))
IF NS = 1 THEN : X = X1(1) + DIF
IF NS = NUM THEN : X = X1(NUM) - DIF
XFLAG = X

Y = Y1(NS)
NS = NS - 1
BEND = 1
FOR B = 1 TO BEND
AEND = NUM - B
FOR a = 1 TO AEND
HO = X1(a) - X
HP = X1(a + B) - X
W = C(a + 1) - DD(a)
DEN = HO - HP
DEN = W / DEN

' D - CORRECTION FROM THE LOWER LEVEL
'C - CORRECTION FROM THE UPPER LEVEL

DD(a) = HP * DEN
C(a) = HO * DEN
NEXT a

' PICK THE SHORTEST PATH TO MOVE

IF (2 * NS) < AEND THEN : DY = C(NS + 1): GOTO 14210:
ELSE GOTO 14200

DY = DD(NS); NS = NS - 1
NEXT B
RETURN

'************************************************************************************************
' SUBROUTINE INPUT
'************************************************************************************************

CLS
'USE IF ONLY THE RUT LEVELS ARE NEEDED

IF INIOPT <> 2 GOTO 14400
CLS : INPUT ""JOB DESCRIPTION :""; AA$
INPUT ""Resilient Modulus - Base Course (psi)""; EBA$
INPUT ""Resilient Modulus - Subgrade (psi)""; ESG$
INPUT ""Thickness of Base Layer (in)""; TBA$
INPUT ""# of Equivalent Standard Wheel (9000 lbs) Passes""; EQPASS$
INPUT ""Allowable Rut Depth (in.)""; RALLOW$
INPUT ""Existing Rut Depth (in.)""; REXIST$

LOCATE 4, 10: PRINT ""INPUT DATA OPTIONS FOR RUT DEPTH CALCULATION:"
1153 \[ \text{LOCATE 6, 10: PRINT ""} 1) \text{Require subgrade material""} \]
1154 \[ \text{LOCATE 7, 10: PRINT ""} \text{classification to determine approximate ""} \]
1155 \[ \text{LOCATE 8, 10: PRINT ""} \text{rutting potentials.""} \]
1156 \[ \text{LOCATE 10, 10: PRINT ""} 2) \text{Require laboratory data on residual deforma-} \]
1157 \[ \text{LOCATE 11, 10: PRINT ""} \text{tion behavior of base and subgrade.""} \]
1158 \[ \text{LOCATE 15, 10: INPUT ""OPTION: 1=SOIL CLASS, 2=LAB DATA ""; OPP"} \]
1159 \[ \text{IF OPP = 2 THEN GOTO 14590} \]
1160 \[ \text{IF OPP <> 1 THEN GOTO 14380} \]
1161 \[ \text{REM ALL BASE COURSE MATERIALS ARE CLASSIFIED AS ONE GROUP.} \]
1162 \[ \text{CLS : LOCATE 4, 10: PRINT ""Subgrade Material Type :""} \]
1163 \[ \text{LOCATE 7, 10: PRINT ""} 1) \text{Heavy Clay - (CH)""} \]
1164 \[ \text{LOCATE 9, 10: PRINT ""} 2) \text{Light/Silty Clay, Clayey Silt - (CL-ML)""} \]
1165 \[ \text{LOCATE 11, 10: PRINT ""} 3) \text{Clayey/Silty/Uniform Sand - (SC-SM)""} \]
1166 \[ \text{LOCATE 15, 10: INPUT ""ENTER SELECTION & \langle RET\rangle ""; MSG} \]
1167 \[ \text{GOTO 14750} \]
1168 \[ \text{CLS : LOCATE 4, 10: PRINT ""Laboratory Data Input:""} \]
1169 \[ \text{LOCATE 6, 10: PRINT ""Log - Intercept of the Straight Line Fit on a} \]
1170 \[ \text{LOCATE 7, 11: PRINT ""Log-Log Plot of Accumulated Residual} \]
1171 \[ \text{LOCATE 8, 10: PRINT ""} \text{Strain""} \]
1172 \[ \text{LOCATE 10, 10: PRINT ""} \text{b - Slope of the Straight Line Fit""} \]
1173 \[ \text{LOCATE 7, 10: PRINT ""} \text{Log-Log Plot of Accumulated Residual Strain""} \]
1174 \[ \text{LOCATE 8, 10: PRINT ""} \text{vs. Number of Load Repetitions""} \]
1175 \[ \text{LOCATE 10, 10: PRINT ""} \text{b - Slope of the Straight Line Fit""} \]
1176 \[ \text{PRINT : PRINT} \]
1177 \[ \text{INPUT ""Log a - Base Course Material""; LGABA} \]
1178 \[ \text{INPUT ""b - Base Course Material""; BBA} \]
1179 \[ \text{INPUT ""Log a - Subgrade Material""; LGASG} \]
1180 \[ \text{INPUT ""b - Subgrade Material""; BSG} \]
1181 \[ \text{IF INIOPT = 2 GOTO 14718} \]
1182 \[ \text{CLS} \]
1183 \[ \text{IF OPP <> 1 GOTO 14704} \]
1184 \[ \text{LOCATE 16, 10: IF MSG = 1 THEN PRINT ""Subgrade - CH - Clay""} \]
1185 \[ \text{LOCATE 16, 10: IF MSG = 2 THEN PRINT ""Subgrade - CL-ML""} \]
1186 \[ \text{LOCATE 16, 10: IF MSG = 3 THEN PRINT ""Subgrade - SC-SM""} \]
1187 \[ \text{GOTO 14714} \]
1188 \[ \text{CLS : LOCATE 16, 10: PRINT ""Base Course Material - ""} \]
1189 \[ \text{LOCATE 17, 10: PRINT ""Log a = ""; LGABA; "". b = ""} \]
1190 \[ \text{BBA} \]
1191 \[ \text{LOCATE 19, 10: PRINT ""Subgrade Material - ""} \]
1192 \[ \text{LOCATE 20, 10: PRINT ""Log a = ""; LGASG; "". b = ""} \]
1193 \[ \text{BSG} \]
1194 \[ \text{CLS : BEEP: PRINT ""DO YOU WANT ANY CORRECTIONS - 0=NO, 1=YES"";} \]
1195 \[ \text{CORR} \]
1196 \[ \text{IF CORR = 1 THEN GOTO 14380} \]
1197 \[ \text{IF CORR <> 0 GOTO 14714} \]
1198 \[ \text{14718 ABA = 10 \^ LGABA; ASG = 10 \^ LGASG} \]
1199 \[ \text{IF ((ABA * 50000! \^ BRA) < .015) AND ((ASG * 50000! \^ BSG) < .016) GOTO} \]
1200 \[ \text{14750} \]
1201 \[ \text{CLS : BEEP: PRINT ""Input Data are Incompatible. Check & Re-enter!"";} \]
1202 \[ \text{GOTO 14400} \]
IF INIOPT = 2 GOTO 14840
'OBTAIN THE TRUCK PASSES AND ALLOWABLE AND MEASURED RUT LEVELS FROM THE MAIN PROGRAM

EOPASS = PA
RALLOW = RX
REXIST = RM
RETURN

'************************************************************~*

'********************************************************************

'SUBROUTINE INPUT2

SKIP IF ONLY RUT LEVELS ARE CALCULATED
IF INIOPT = 2 GOTO 14940

EBA = E1(INC)
ESG = E2(INC)
TBA = 9A(INC)

RETURN

'CALCULATE "a" AND "b" ("a" IS KEPT WITHIN PRACTICAL LIMITS)

ABA = AABA * (EBA / 1000) ^ ABA
IF ABA > .0035 THEN ABA = .0035
IF MSG <> 1 GOTO 15190
BSG = BCH
ASG = AACH * (ESG / 1000) ^ ABCH
IF ASG > .004 THEN ASG = .004
IF MSG <> 2 GOTO 15230
BSG = BCL
ASG = AACL * (ESG / 1000) ^ ABCL
IF ASG > .001 THEN ASG = .001
IF MSG <> 3 GOTO 15270
BSG = BSM
ASG = AASM * (ESG / 1000) ^ ABSM
IF ASG > .005 THEN ASG = .005

REM USE IF ONLY RUT LEVELS ARE NEEDED

IF INIOPT <> 2 GOTO 15550

CLS : REM DISPLAY INPUT DATA
LOCATE 2, 10: PRINT """"; AA$
LOCATE 4, 10: PRINT "Resilient Modulus (psi) - Base = "; EBA"
LOCATE 6, 10: PRINT ""
LOCATE 8, 10: PRINT ""Thickness of Base Layer = " ; ESG
LOCATE 10, 10: PRINT ""# of Equivalent Standard Wheel Passes = " ; EQPASS
LOCATE 12, 10: PRINT ""Allowable Rut Depth = " ; RALLOW;
LOCATE 14, 10: PRINT ""Measured Rut Depth = " ; TBA;

IF OPP <> 1 GOTO 15460
LOCATE 16, 10: IF MSG = 1 THEN PRINT ""Subgrade - CH - Clay""
LOCATE 16, 10: IF MSG = 2 THEN PRINT ""Subgrade - CL-ML"
LOCATE 16, 10: IF MSG = 3 THEN PRINT ""Subgrade - SC-SM"
GOTO 15510

LOCATE 17, 10: PRINT ""Allowable Rut Depth = " ; RALLOW;
LOCATE 19, 10: PRINT ""Measured Rut Depth = " ; REXIST;
GOTO 15460

LOCATE 20, 10: INPUT ""DO YOU WANT ANY CORRECTIONS - 0=NO, 1=YES""
IF CORR = 1 THEN RETURN
IF CORR <> 0 GOTO 15510

15550 'ASSIGN VALUES TO INTERPOLATION PARAMETERS
CYL = 300000!
XII = ABA * CYL ^ BBA
XLL = ASG * CYL ^ BSG
IF XII > .015 THEN XII = .015
IF XLL > .016 THEN XLL = .016
IF XLL < .001 AND TBA < 10 THEN XLL = .001

XMM = TBA: XJJ = EBA: XKK = ESG
IF TBA < 10 AND ESG > 20000! AND EBA > 100000 THEN XJJ = 100000
IF TBA < 10 AND EBA > 60000 AND ESG > 30000 THEN XKK = 30000: IF TBA < 6 THEN XMM = 6: IF EBA < 30000 THEN XJJ = 30000
RETURN

15680 'SUBROUTINE OUTPUT

15760 IF INIOPT <> 2 GOTO 15760
CLS
IF EOPASS = 0 GOTO 15810
LOCATE 6, 10: PRINT ""EQUIVALENT STANDARD WHEEL PASSES = " ; EQPASS"
LOCATE 8, 10: PRINT ""RUT DEPTH CAUSED = " ; RFINAL(1); " in."
LOCATE 10, 10: PRINT ""ALLOWABLE RUT DEPTH = " ; RALLOW; " in."
LOCATE 12, 10: PRINT ""EXISTING RUT DEPTH = " ; REXIST; " in."
LOCATE 14, 10: PRINT "ALLOWABLE WHEEL PASSES = "; passes(1)

IF passes(1) = 0 THEN : LOCATE 16, 5: PRINT "INPUT DATA MAY BE INCOMPATIBLE. CHECK & RERUN!"

PRINT : PRINT : PRINT

15860 IF INIOPT = 2 THEN : INPUT "" ; PRESS RETURN TO CONTINUE ......"" ; PRESS

CLS

RETURN