DATA ACQUISITION SYSTEM FOR FATIGUE LIFE ESTIMATION OF BRIDGE COMPONENTS

By

JOHN W. KELLY

Bachelor of Science

Oklahoma State University

Stillwater, Oklahoma

1994

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
MASTER OF SCIENCE
December, 1995

DATA ACQUISITION SYSTEM FOR FATIGUE LIFE ESTIMATION OF BRIDGE COMPONENTS

Thesis Approved:

Thesis Adviser

John G. Lloyd

Timothy D. Hague

Thomas C. Collins
Dean of the Graduate College

ACKNOWLEDGMENTS

This research was sponsored by the Oklahoma Department of Transportation (ODOT) and the Federal Highway Administration (FHA); financial support provided by ODOT and the FHA is gratefully acknowledged. In addition, the author wishes to thank the technical support group at Campbell Scientific for their assistance with the Datalogger programming and operations.

The opinions expressed in this report are those of the author's and do not necessarily represent the views of the project sponsors.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

A highway bridge on U.S. Highway 69 crossing the South Canadian River near Eufaula, Oklahoma, was widened to accommodate a larger volume of traffic. In so doing, there were construction difficulties which resulted in improper field welds. A total of 136 welds which were designed as category C failed to meet welding specifications due to poor welding techniques. As a result, these welds were no longer category C, but were some inferior classification with an unknown fatigue life. Using the calculated stresses, it was determined that these welds would fail in fatigue if the classification was lower than category C. If, however, the actual stresses could be measured, some or all of the welds may have an infinite life.

To investigate this possibility, it was necessary to determine the properties of the bridge and the welds. The Texas Department of Transportation (TxDOT) had developed a system to measure the in-service stresses in existing bridges. This system was developed in 1988 at the University of Texas at Austin [Post et al., 1988] to predict the fatigue life of bridge components. This system was used to determine the effective stress range so that a computer model of the bridge could be made, and so that the welds could be tested in a laboratory.

The poor welding eliminates these welds from any AASHTO classification, and the fatigue life could not be predicted based on previous data experiments. It was necessary to conduct tests on similar welds which were fabricated to emulate the poor

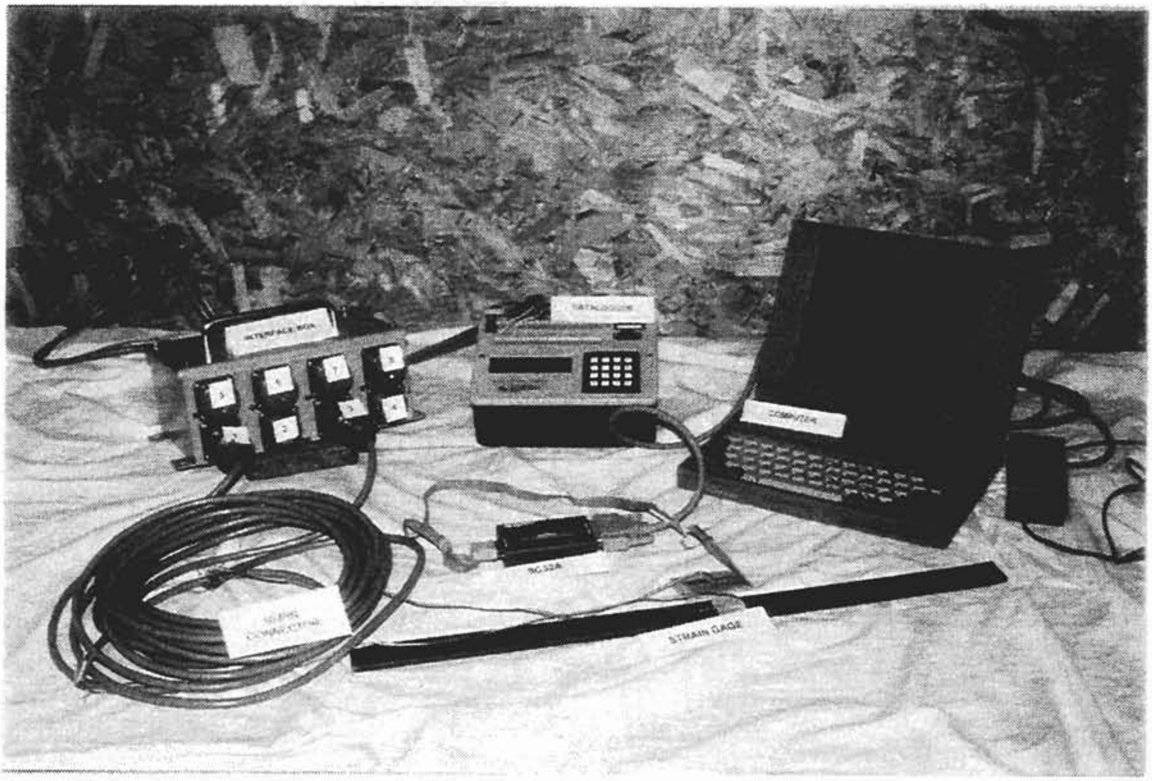


Figure 1. Data Acquisition System

range by conducting a rainflow histogram routine, and it is capable of recording continuous data as a truck is driven across the bridge deck, analogous to influence line data. The system at the time of this report will initiate a test, conduct the test, retrieve the data of the test, and produce tabulated data and raw data for additional graphic processing.

The system is driven by a computer program that is capable of operating all of the necessary steps in executing a test, collecting the data from the test and preparing the data for subsequent processing. The program is menu driven so that only a small amount of training should be necessary to equip persons who will be using the system.

1.3 Scope of Report

A detailed overview of the Data Acquisition System (DAS) is provided to present circuitry and layout of the system, and the purpose of each component. This will be helpful for future modifications, reproductions, or repairs. A portion of this report is also devoted to system programming, and is intended to be used as an operation manual. A printed copy of the program code is provided in Appendices C and D, and has been documented to assist in future program modifications. Finally, test data as well as an example problem are provided for operation assistance.

This system differs from the TxDOT system in that the user input is simplified, and the output is generated into two different forms: one data file for graphical processing and another for tabulated printouts. Additionally, the scope of the system has been expanded to increase usability. This includes the operation of the input channels, the employment of gage factors, and the direct output of actual stress values in units of ksi.

CHAPTER 2

LITERATURE SEARCH

2.1 Fatigue In Steel Bridges

Bridges undergo numerous cycles per day from the loading and unloading caused by traffic. This cycling produces stress changes in the structural members which can lead to fatigue failure. The danger of fatigue is that failures occur at stresses below yield. A cross frame diaphragm may be able to easily withstand the stresses from a single large truck crossing over the bridge deck without yielding. Yet if the same truck were to drive over the bridge millions of times, the diaphragm which had no threat of yielding may suddenly fail from a fatigue crack.

The cycling of load aggravates a material flaw or an irregularity to produce a crack. As the cycling increases the crack propagates and, if ignored, will lead to failure. A microcrack can occur from numerous types of material flaws. Some flaws are a result of discontinuities in the microstructure. Other flaws are a result of handling, erecting or fabricating the steel member. The most obvious type of flaw of this nature is a weld. When two pieces of steel are welded together, the pair becomes a single, continuous material. However, the weld introduces a great potential for fatigue failure by disrupting the natural makeup of the steel fibers. The weld is analogous to a paper perforation. If the paper is pulled or torn, it is most likely to tear along the perforation. When a welded member is cycled millions of times, the material is most likely to fatigue along the weld. The perforation and the weld are weak links of the respective systems; each provides a foothold for the failure mechanism to develop.

A welded member subjected to a cycling load will eventually fail from fatigue fracture. The number of cycles required to fail the member is dependent on the quality of the weld. Welding is an imperfection by nature in that it disrupts the continuity of the material and introduces residual stresses, stress concentrations and flaws. Even the smallest imperfection of a weldment can eventually produce a fracture. Since crack propagation increases exponentially with crack length, the fatigue life is closely dependent on the size of the largest or worst flaw [Barsom and Rolfe, 1987]. Similarly, poor weld quality can greatly decrease the number of cycles required to fracture.

To reduce the threat of fatigue failure, numerous tests have been conducted to determine the most effective way to make welded connections that minimize the consequences of fatigue. Technical organizations such as the American Association of State Highway and Transportation Officials (AASHTO), the American Institute of Steel Construction (AISC), the American Society for Testing and Materials (ASTM), and the American Welding Society (AWS) have written specifications for construction welding based on these tests [Barsom and Rolfe, 1987].

AASHTO has established different weld categories based on the type of connection, the type of weld and the location of weld. Each category has a stress range dependent on the number of cycles and the redundancy status. Category descriptions as well as the allowable stress ranges can be found in the AASHTO LRFD Bridge Design Specification, 1994.

2.2 Fatigue Life Estimation

When a bridge is designed, the fatigue life of the bridge is specified by the owner (usually the Department of Transportation). All bridge components are designed to meet or exceed the design life of the entire structure. Numerous studies are conducted to determine the types of loads that the bridge is expected to encounter as well as the number of cycles. These estimations are made conservatively to encompass the unknown and to develop a margin of safety.

Once the structure is constructed and in service, the actual stresses may be (and should be) less than the stresses from the design against the estimated loads. Using either calculated stresses or stresses based on measurements, the remaining fatigue life may be calculated using the following equation:

$$Y_f = \frac{f K \times 10^6}{T_a C (R_s S_r)^b} - a \tag{2.1}$$

where

Y_f = remaining fatigue life in years

S_r = effective stress range

R_s = reliability factor

C = stress cycles per truck passage

K, b, f = fatigue curve constants

T_a = estimated lifetime average daily truck volume

a = present age of the bridge in years.

This project focuses only on the effective stress range, S_r, and the estimated lifetime average daily truck volume of Equation 2.1.

Generally, S_r is estimated from design models and statistics. Although this is a practical and accepted method, it demands a conservative approach given the degree of uncertainty. Tests performed with the DAS will measure the actual in-service stress ranges, allowing a more accurate estimation to be calculated. There is still a degree of uncertainty in this method and is in no way intended to convey an exact result. The effective stress range is just an estimate based on working stresses rather than design model stresses.

2.3 Calculation of the Effective Stress Range

The DAS uses a Rainflow Histogram algorithm to count stress cycles. The effective stress range is calculated from this data with the equation:

$$S_r = \left(\Sigma p_i S_{ri}^3\right)^{1/3} \tag{2.2}$$

where p_i is the fraction of stress ranges within an interval, and S_{ri} is the interval mean. This summation calculates a weighted average of the stress cycles to represent an entire interval of stresses with a single number. For example, a 7-day test with 24-hour intervals will have a wide range of stresses throughout a single day. The Rainflow Histogram tabulates the number of cycles that occur at each histogram increment. Equation 2.2 is used to compile the entire interval output into a weighted sum. Assume that on one of the days, there were 999 cycles ranging from 2 to 3 ksi and 1 cycle of 18 ksi produced by a heavy truck. For this interval,

$$p_{1-999}S^3_{r(1-999)} = \left(\frac{999}{1000}\right)\left(\frac{2+3}{2}\right)^3 = 15.609 \qquad p_2 S^3_{r1000} = \left(\frac{1}{1000}\right)(18)^3 = 5.832$$

and

$$\underline{\underline{S_r = (15.609 + 5.832)^{1/3} = 2.778\text{ksi}}}\,.$$

One truck had very little effect on the effective stress range for this case; but if the number of trucks was increased from 0.1 to 2%, the results change to

$$p_{1-980}S^3_{r(1-980)} = \left(\frac{980}{1000}\right)\left(\frac{2+3}{2}\right)^3 = 15.313 \qquad p_{981\text{-}1000}S^3_{r(981\text{-}1000)} = \left(\frac{20}{1000}\right)(18)^3 = 116.64$$

and

$$\underline{\underline{S_r = (15.313 + 169.64)^{1/3} = 5.09\text{ksi}}}\,.$$

With just a small change in the number of cycles at the 18 ksi range, the effective stress range nearly doubles. Since the interval mean value in the equation is cubed, the sum is weighted toward the higher stresses, but extremely infrequent high stresses have very little overall effect.

2.4 Lifetime Average Daily Truck Volume

The output from a Rainflow histogram conveniently provides the number of cycles per interval. For a 1-week test with 24-hour intervals, the number of cycles per week can be calculated. Statistical methods have been developed to predict the average daily truck volume over the entire life of the bridge from the current traffic volume. Although data obtained from the DAS will be used for this estimation, the actual calculations are beyond the scope of this report. Information pertaining to this topic may be found in “Fatigue Evaluation Procedures for Steel Bridges” [Moses et al., 1987].

CHAPTER 3

DATA ACQUISITION SYSTEM AND EQUIPMENT

3.1 Datalogger

A datalogger (DL) is a programmable device that is capable of performing numerous electrical measurements, such as strain gage readings, and storing the data in memory for further processing. The datalogger device used for this project is the Campbell Scientific 21X model. The 21X, shown in Figure 2, has four excitation channels capable of supplying +/- 5 volts DC and 8 differential Input channels. It may be powered by either 12 D cell batteries or a 12V marine battery, and has a final storage capacity of 19,296 locations.

There are two ways to program the DL: (1.) entering coded instructions through the keypad located on the DL's control panel, or (2) downloading a coded instruction program via computer through the serial I/O. Entering a program with the keypad is quite tedious. Programs for the DL are written in a terse code comprised of numbers and letters which correspond to commands that are listed in the Campbell operation manual. Coded programs can be terribly long and difficult to debug for even simple tests such as the ones used for this project. Although this is a problem for keypad input, the repetitive nature of the programs is optimal for programming by computer.

The software written for this project allows the user to create DL tests without encountering the terse code. The main program, discussed in the following sections, generates the downloading files once the user has provided the appropriate parameters. This software was written with the user in mind, intent on minimizing user knowledge of a strict protocol of any kind. Although training will be necessary for testing crews, reference to the operation manual should be limited.

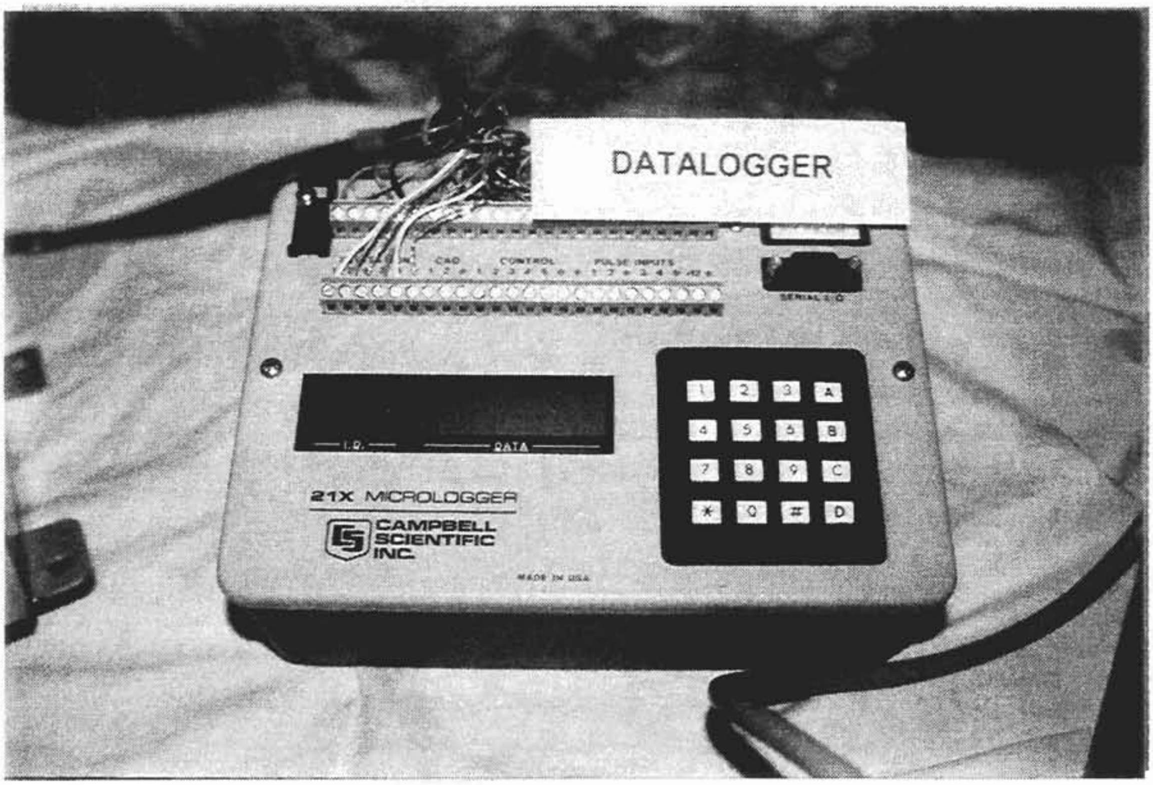


Figure 2. Datalogger

3.2 Interface Box

The Interface Box is a circuitry junction box for all the DL Input Channels. Figures 3 and 4 provide a photograph and a sketch of the front and back faces of the box. The Input Channels are the 8 receptacles on the front face, and will be referred to as Inputs for the remainder of this report. The Inputs may be used in any order and any combination. Each Input is equipped to process either a strain gage or a transducer, but only one device per Input. There are no restrictions on the mixing of these two types of devices within a test. Refer to Figure 5 for an example of Input configuration.

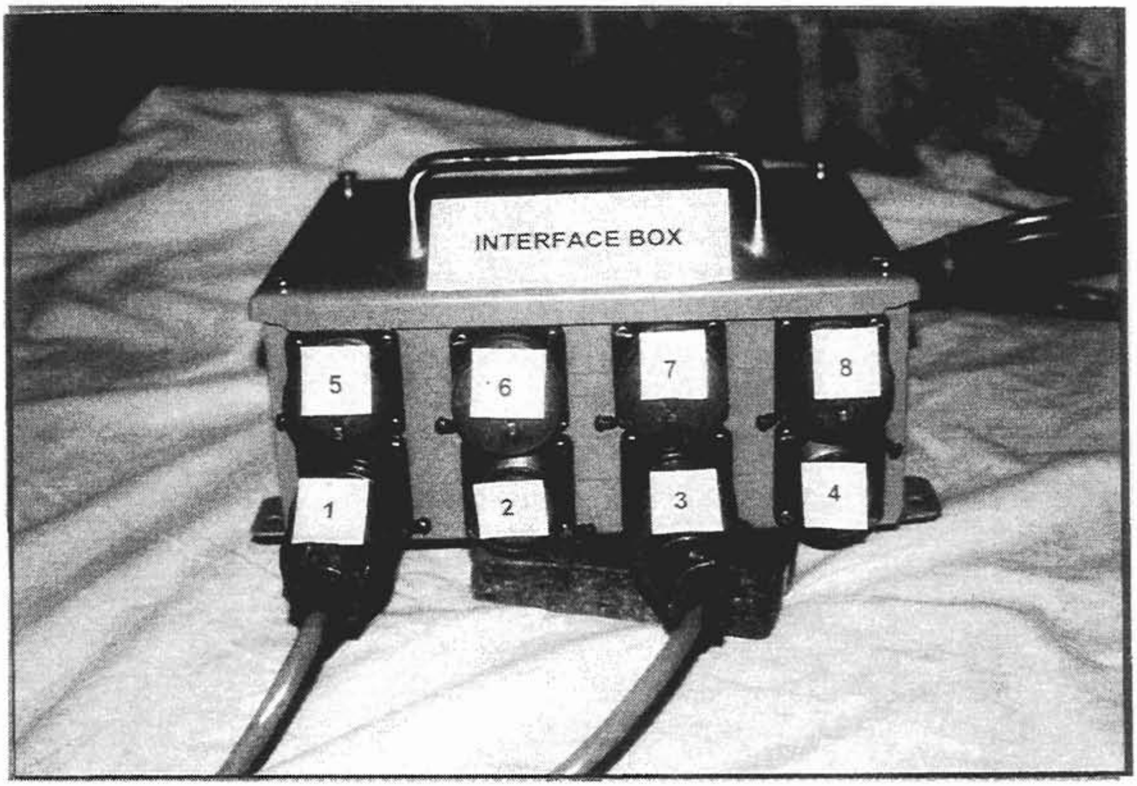
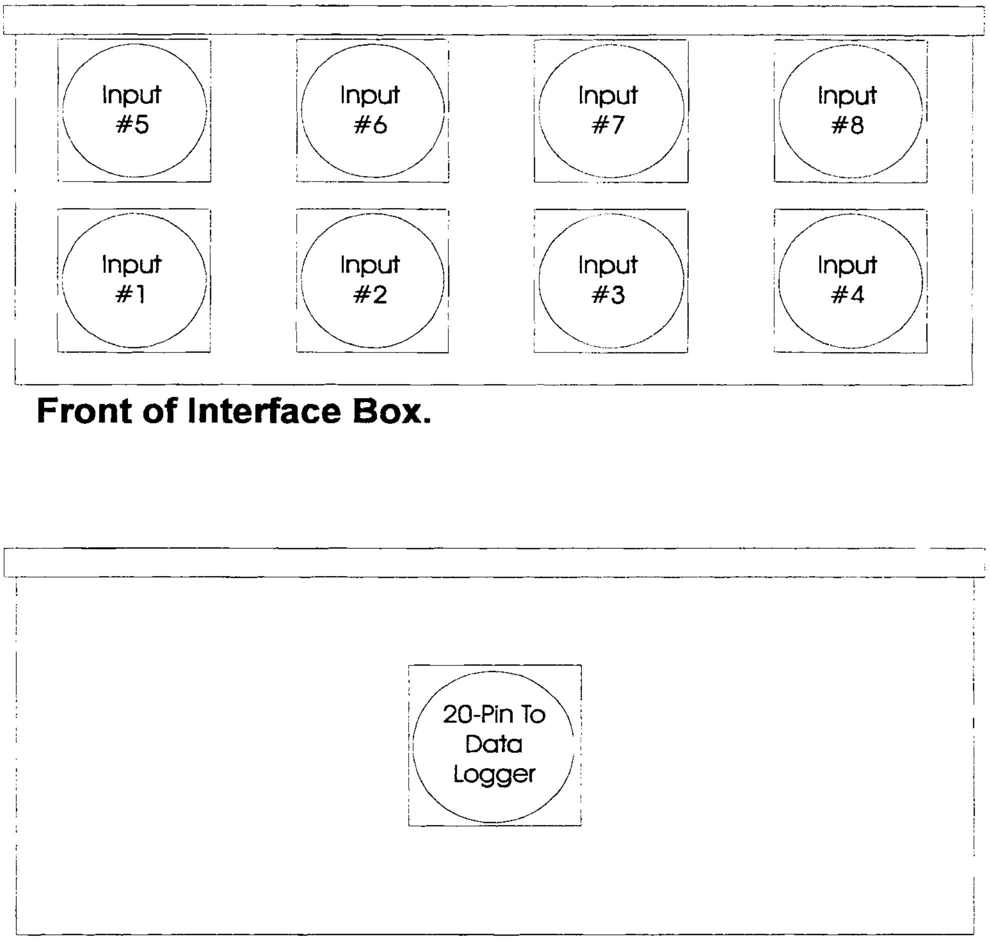


Figure 3. Interface Box



Front of Interface Box.

Back of Interface Box.

Figure 4. Sketch of Interface Box. The 8 Input channels are 16-pin "cannon plugs." All circuitry from the inputs are routed through the 20-pin "cannon plug" to the data logger (DL).

Input #5 Transducer	**Input #6** Vacant	**Input #7** Vacant	**Input #8** Strain Gage
Input #1 Vacant	**Input #2** Strain Gage	**Input #3** Vacant	**Input #4** Transducer

Figure 5. Input Example. Here, the inputs are not used in sequential order, and there is a mix of strain gages and transducers.

Since the DL has only 4 excitation channels, a means had to be devised to supply excitation to all 8 inputs. This was achieved by allotting two Input channels to every excitation channel. This system will be referred to as an Excitation Array. The designation of each array is presented in Figure 6. Note that the boxes are intended to represent the location and configuration of the Input plugs on the Interface Box front.

Excitation #1	Excitation #2	Excitation #3	Exciation #4
Input #5	Input #6	Input #7	Input #8
Input #1	Input #2	Input #3	Input #4

Figure 6. Excitation Arrays. Each excitation channel services two Input channels to make up an Excitation Array.

Figure 7 provides a layout of the inside of the Interface Box and an outline for the schematic of the circuitry. Note the 8 Inputs on the front face and the 8 Bridge Completion Module (BCM) chips, one per Input, located on the inside panel of the box. Symmetry was implemented as much as possible to simplify the wiring. The two terminal strips are divided into two sections each. The left sides of each terminal strip service the four Inputs on the left side of the box face (Inputs #1, #2, #5, and #6), and the right sides service the four on the right. The terminal strip closest to the BCM chips services the four channels located on the center of the box face (Inputs #2, #3, #6, and #7) and the other strip services the other four Inputs. The Common for the excitation circuits passes through the middle pair of poles of one of the terminal strips connecting the P- poles of each excitation channel to the Interface Box casing. Figures 8 through 10 may be useful in visualizing the circuitry layout.

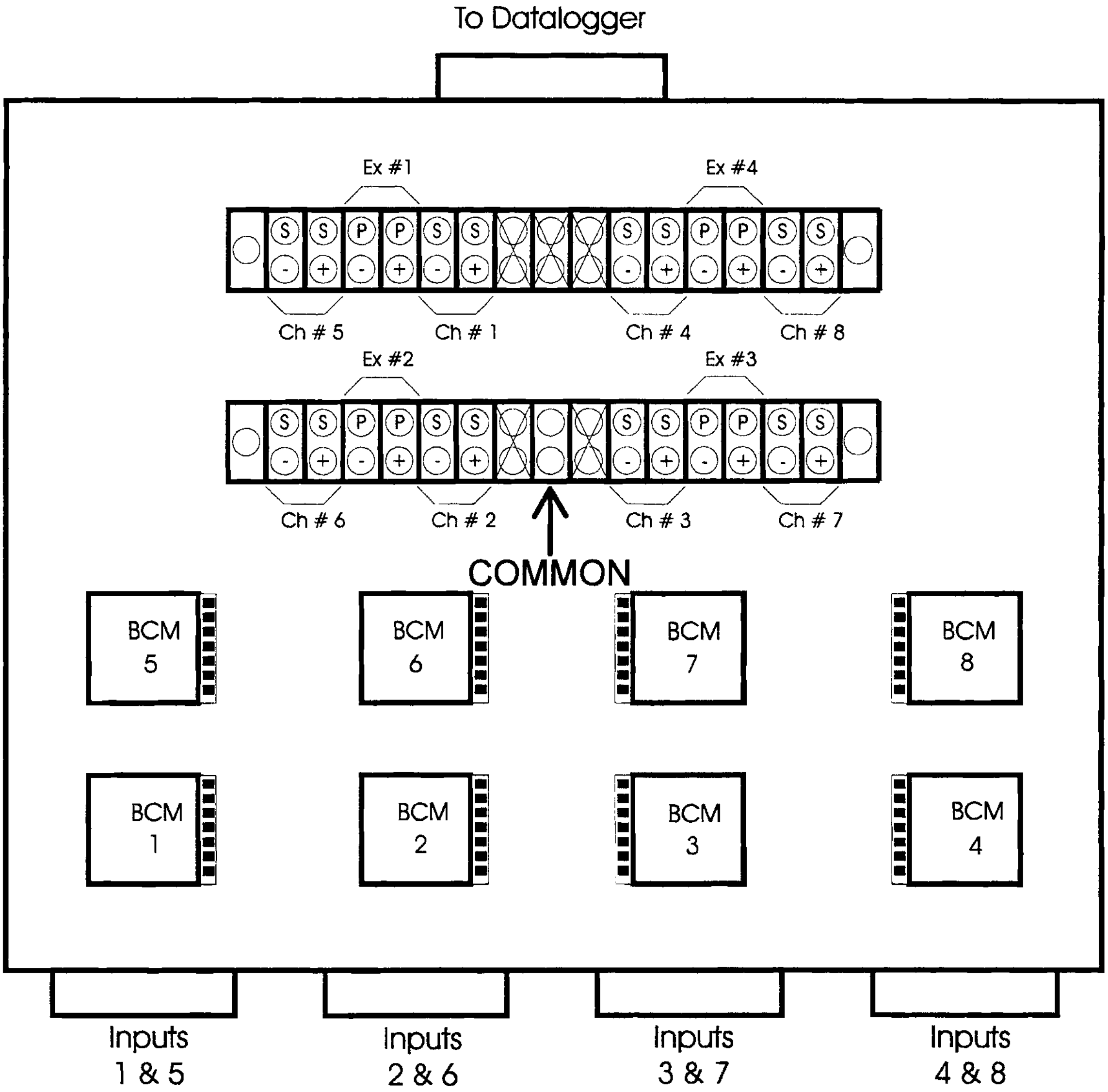


Figure 7. Plan View of the Interface Box. The terminal strips are partitioned by brackets to designate the divisions of the excitation channels and the Input channels. Refer to Figures 8 and 9 for a more detailed sketch of a typical excitation channel branch.

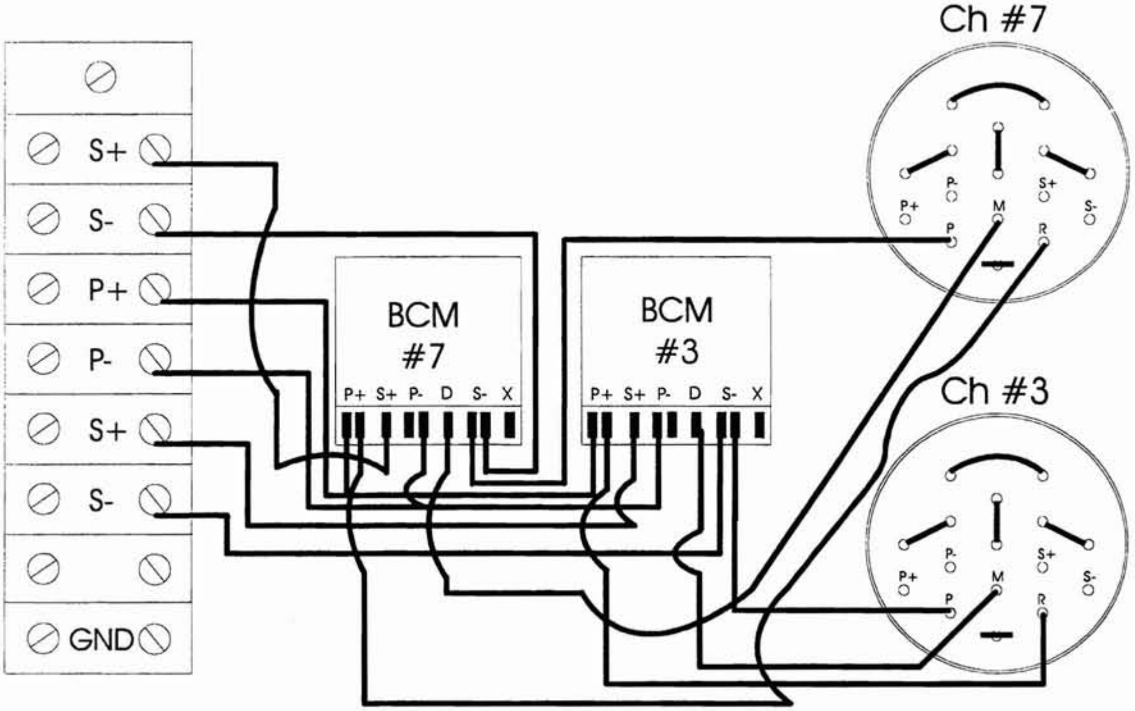


Figure 8. Strain Gage Circuit. Single strain gage circuit diagram for Excitation Channel #3 and associated Input Channels #3 and #7. Only pins P, M, and R of the channel plugs are used for the single strain gage. The 8 unlabeled pins that are joined in pairs are for the transducer circuit, as are the four pins label P-, P+, S-, and S+. The bottom pin with the dash through it is unused.

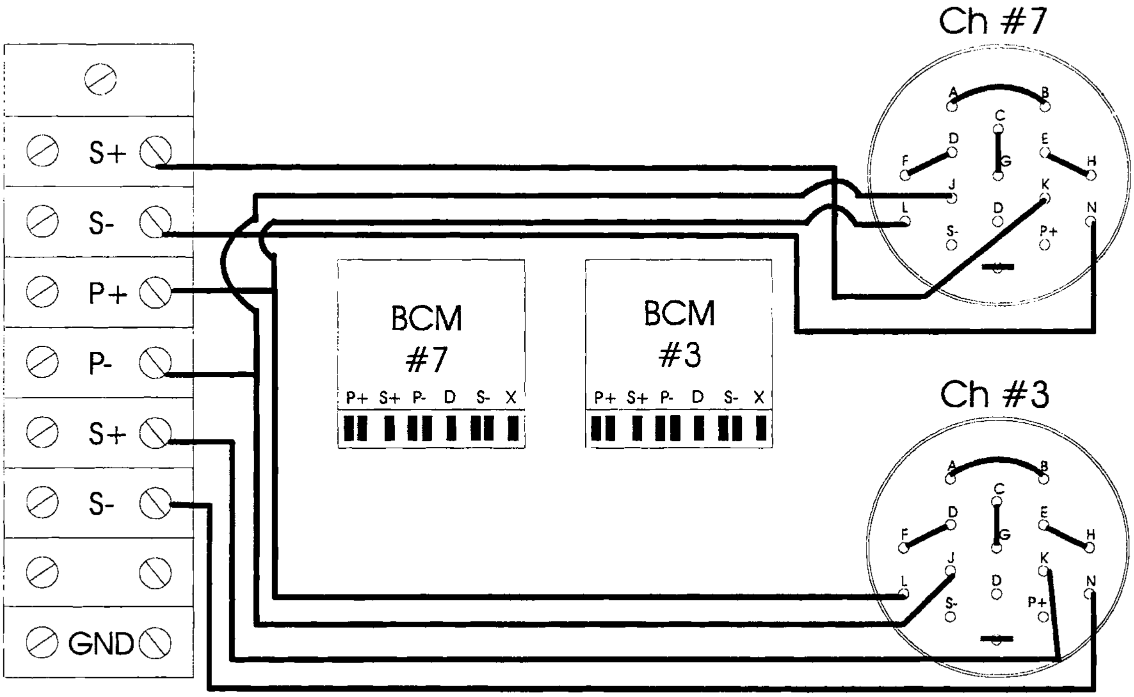


Figure 9. Transducer Circuit. Transducer circuit diagram for Excitation Channel #3 and associated Input Channels #3 and #7. All lines for single strain gage are omitted to increase legibility. Pins A through H adjoin the four strain gages into the bridge configuration. Pins L, J, K, & N are the excitation poles and measurement poles, respectively. The bottom pin with the dash through it is unused.

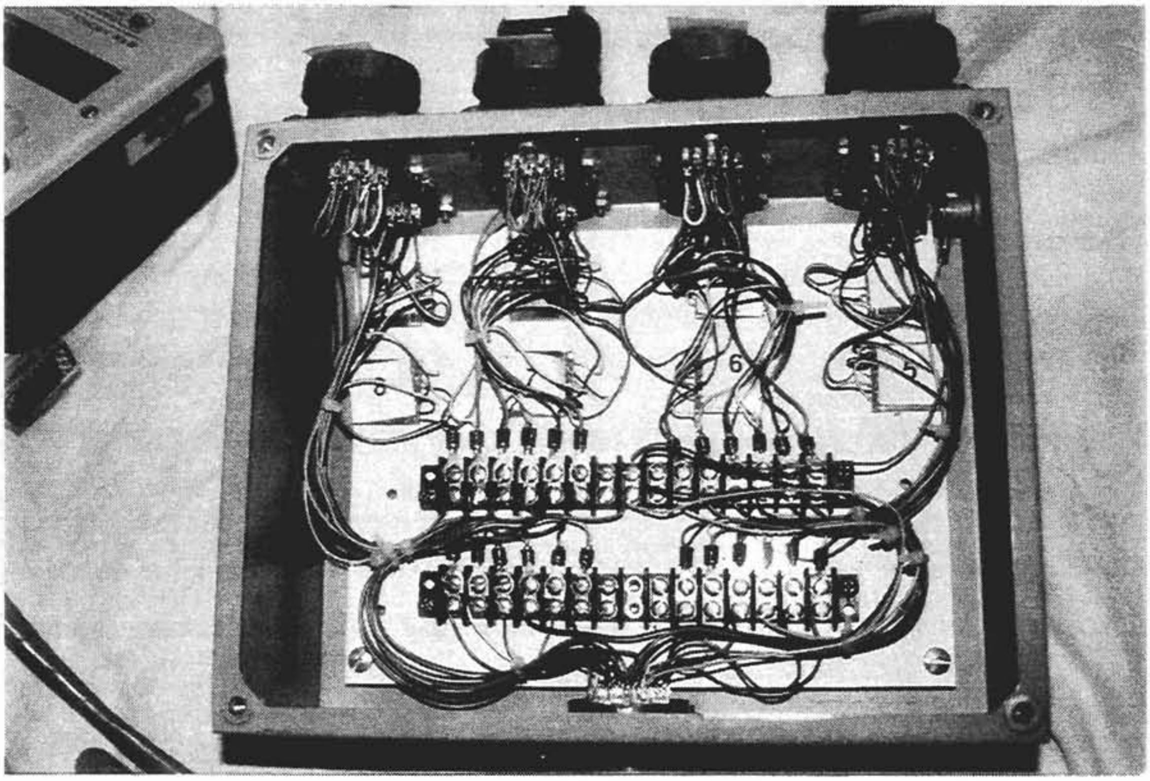


Figure 10. Interface Box Interior

3.2.1 Excitation. There are only four excitation channels to supply 8 Inputs. Therefore, each excitation channel is connected to two Inputs in parallel, and actually excites both simultaneously. The channel groupings for Excitation Arrays are as follows:

The ordering of the Input channels in Figure 11 is to convey the orientation of the Inputs on the front face of the Interface Box. Note that Inputs #1 through #4 make up the bottom row on the Interface Box and Inputs #5 through #8 make up the top row.

Excitation #1	Excitation #2	Excitation #3	Exciation #4
Input #5 Input #1	Input #6 Input #2	Input #7 Input #3	Input #8 Input #4

Figure 11. Excitation Arrays. Each excitation channel services two Input channels to make up an Excitation Array.

Each channel has a circuit for both a transducer and a strain gage. These circuits are shown in Figure 12. The P+ and P- are the excitation poles and the S+ and S- are the measurement poles. Although all Inputs share an excitation channel with another Input, each must have a separate set of measurement poles.

The excitation and measurement poles (P+, P-, S+, S-) make up an ensemble of leads for a Wheatstone Bridge, or a *full bridge*. Refer to Figure 13 for full bridge schematics of measurement devices. A full bridge is a configuration of four resistors of equal resistance. In the case of a transducer, all four resistors are 350Ω strain gages. If an individual strain gage is the measurement device, the full bridge is composed of one 350Ω strain gage and three 350Ω resistors, provided by the Bridge Completion Modules (BCM).

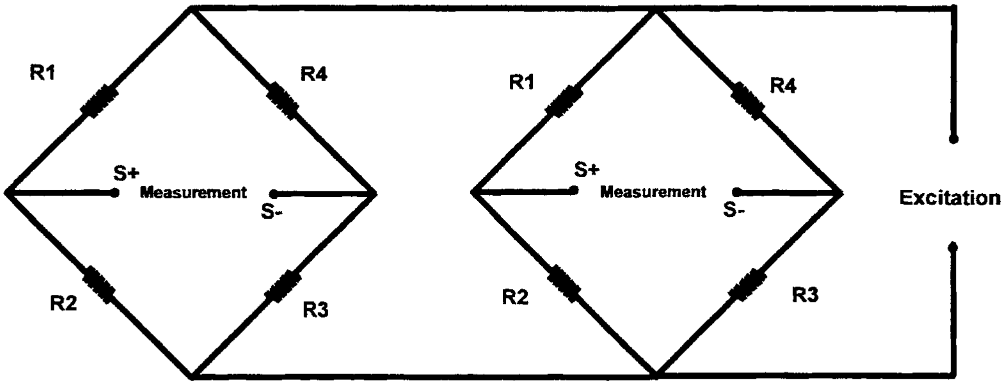


Figure 12. Schematic of a Single Excitation Array. Each bridge represents one Input which could be a strain gage circuit or a transducer. Since both bridges are parallel with each other and the excitation, the same voltage will be applied to either one for any scenario. This will also be true if one of the channels is vacant.

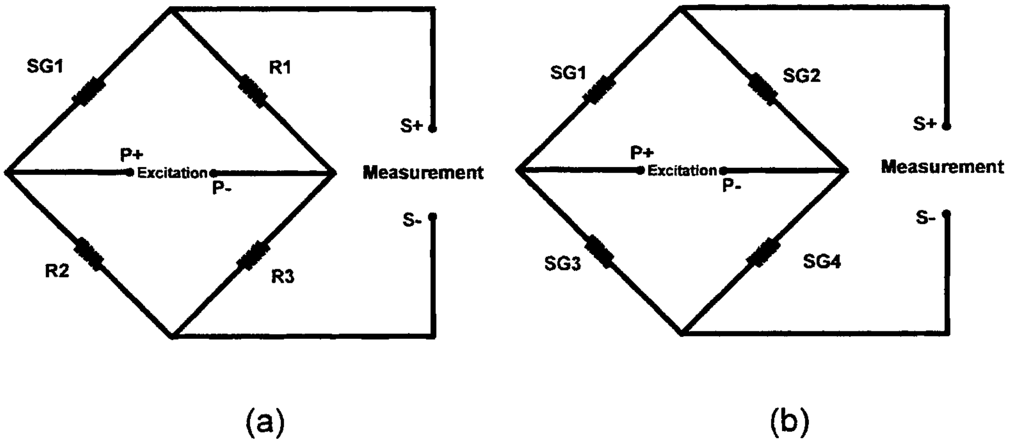


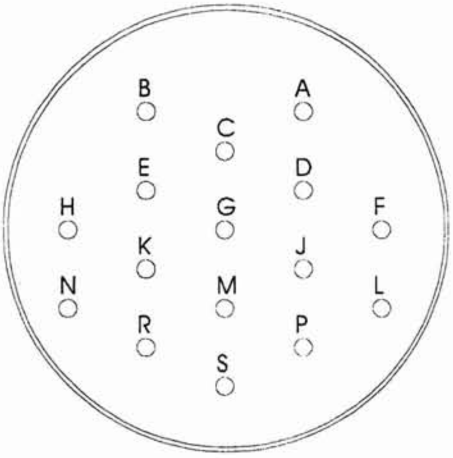
Figure 13. Full Bridge Circuits. Configurations of (a) a single strain gage (SG1) and a Bridge Completion Module (R1, R2, and R3), and (b) a transducer with 4 strain gages. Both circuits are equivalent

Since both devices are full bridges, the DL operates individual strain gage circuits and transducer circuits identically. There is no need for "switching" mechanisms or program logic to select the appropriate circuit. Instead, the circuitry is mechanically controlled by the Input connection. When an Input connector is plugged into the socket, the correct circuit will be activated while the other circuit remains open. Connections were fabricated so that individual strain gages and transducer circuits would not share any common pins. If a transducer is installed, leads from the four strain gages configure a full bridge inside the box. If a strain gage is installed, the leads from the gage are connected to the BCM.

Each BCM may be configured for a 120Ω gage or a 350Ω gage. The distinction is made within the Interface Box wiring. Refer to Figure 8 and note the solder tab labels on the BCM sketches. The tab with the label D is the wire leg for 350Ω gages and the tab with the label X is the wire leg for 120Ω gages. As shown in the schematic of Figure 8, the DAS is currently restricted to 350Ω gages. In order to expand the system to accept 120Ω gages, a wire connection must be made between the X tab on each BCM and the respective unused pin (pin S) of each Input connector (the dashed pin in Figure 8). Once this has been done, 120Ω gages may be employed only if the strain gage wire connection is modified. Instead of connecting the strain gage wires to pins P, M, and R, the wires should be connected to pins P, "the dashed pin," and R. Further detail on strain gage connections are discussed in section 3.3.

3.2.2 Measurement Readings. The measurement circuits are much simpler than the excitation circuits. Each Input is equipped with a set of measurement poles notated as S+ and S-. Figure 7 shows the terminal locations for each of the Input measurement poles. Figures 8 and 9 provide circuit diagrams for the transducer and strain gage of an Excitation Array; the S+ and S- poles can be traced out in this diagram. Only one-half of the circuit will ever be completed at one time (either the transducer or the strain gage). The Input connector mechanically completes the proper circuit.

3.2.3 16-Pin Connector. Input channels are attached to the Interface Box with 16-pin cannon plugs. The sketch in Figure 14 is a view of the solder cups where the cable is attached. This sketch along with Table 1 should be used as a map and legend to construct Input cables. A different colored wire is soldered to each of the pins. The same colors are used for each cable-connector system in order to make them all uniform.



Pin	Color	Function
A	Gray	Gage 1R
B	Yellow	Gage 3W
C	Purple	Gage 1W
D	Pink	Gage 2W
E	Brown	Gage 3R
F	Black	Gage 4R
G	Green	Gage 2R
H	Blue	Gage 4W
J	Red	Gage 1B (P+)
K	White	Gage 3B (S-)
L	Tan	Gage 4B (P-)
M	Red/Yellow	Strain Gage D
N	Orange	Gage 2B (S+)
P	Red/Green	Strain Gage S-
R	Red/Black	Strain Gage P+
S	Vacant	N/A

Figure 14. 16-Pin Connector. The display above is shown from an inside view of a connector plug where the solder cups are located. This is the same view that would be needed to construct a new cable.

Table 1. Wire Color Legend for 16-Pin. The table on the right is a legend for pin assignments. The colors are those of the wires in the 15-conductor cable. The highlighted lines are for strain gages; all other pins (except pin S) are used for the transducer. The letters R, W and B are the colors red, white and black, respectively. These are the colors of strain gage lead wire.

3.2.4 20-Pin Connector. A double cable fixed to a 20-pin cannon plug runs from the interface box to the DL. Two 15-conductor cables are used to service the pins. A view of the 20-pin plug is sketched in Figure 15. Table 2 lists each pin letter, the respective wire color, and a description of its function. All four excitation channels have a common pole (P-). The Common for the circuit is labeled on the terminal strip shown in Figure 7. From this terminal, the Common is carried to the casing of the Interface Box through a grounding screw which is mounted on the inside. The circuit is completed through the 20-Pin plug casing and a wire which links the casing to the DL. The 20 pins of the plug service the 8 measurement poles (16 pins) and the 4 excitation P+ poles (4 pins). Since there are not 20 unique wire colors, the same 10 colors were used from each cable. Although this could be confusing, it is not likely to cause problems since complete disassembly of the cable should not be necessary. If there is ever a need to repair the cable, a simple system of wire labeling prior to any wire disconnecting should prevent problems. In Table 2, annotations of 1 and 2 are employed to differentiate the wires into the two groups from which they originate.

3.3 Measurement Devices

Stress cannot be directly measured by any practical means. Strain, however, can be measured quite easily. Strain gages and transducers operate very similarly to measure the strain of the surface with which they are in contact. When a member, say a steel beam, is subjected to a load, the length of the member undergoes a displacement. The measurement device attached to the member must also undergo this displacement, allowing the DL to determine the strain. The stress is calculated from the strain measurement by Hooke's Law (Stress = Strain x Modulus of Elasticity). The Modulus of Elasticity, often referred to as E, is a material property, and is equal to 29,000 ksi for structural steel. All calculations for obtaining the stress values are performed by the computer and the DL during the test.

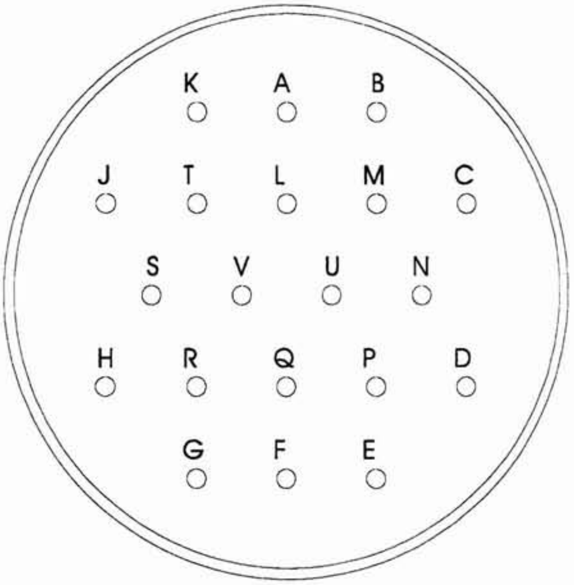


Figure 15. 20-Pin Connector. The schematic above represents the inside of the cable connector that is attached to the interface box. This is the view that would be needed to repair or reconstruct a cable.

Pin	Color	Function
A	Blue 1	7 S+
B	Orange 1	2 S+
C	Red/Black 1	5 S+
D	Red/Yellow 1	5 S-
E	Purple 1	1 S-
F	Tan 2	6 S-
G	Red/Yellow 2	4 S-
H	Red/Black 2	8 S-
J	Brown 2	8 S+
K	Purple 2	3 S+
L	Yellow 1	7 S-
M	Brown 1	2 S-
N	White 1	1 P+
P	Tan 1	1 S+
Q	Blue 2	6 S+
R	Green 2	4 S+
S	White 2	4 P+
T	Yellow 2	3 S-
U	Green 1	2 P+
V	Orange 2	3 P+
Case	Bare 1 & 2	Common

Table 2. Wire Color Legend for 20-Pin. Each pin of the connector services a colored wire. The numbers 1 or 2 after each color indicates from which cable the wire originates. All of the wires for cable 2 are shaded. The numbers in the Function column are associated with either the Excitation or Measurement Channels.

3.3.1 Single Strain Gage. A single strain gage can be quite useful for taking measurements in space limiting locations on the bridge. The disadvantage of the strain gage is that more time is required for proper bonding to the testing surface. A picture of a gage bonded to a steel bar is shown in Figure 16.

Compensation for lead wires should not be necessary. An 80-foot cable was measure to have a resistance of about 2.0Ω. The impact of this increased resistance for

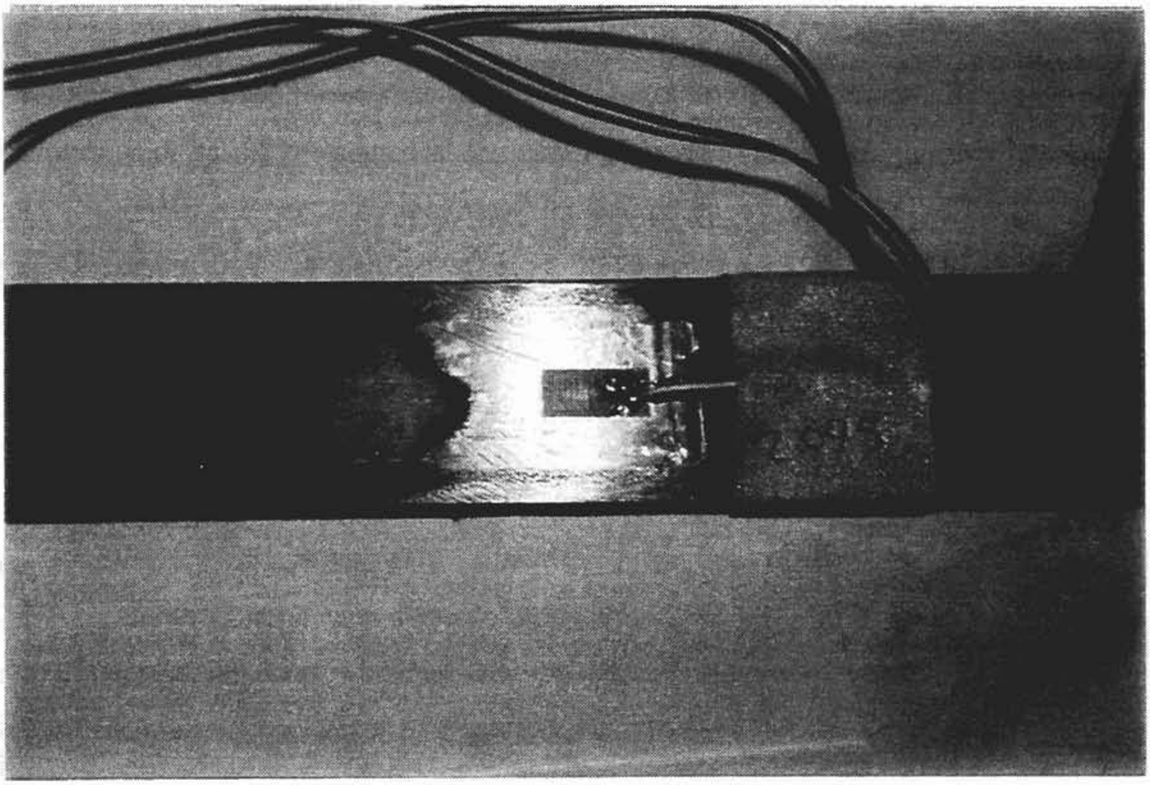


Figure 16. Strain Gage

a 350Ω gage is shown in the following equation for the adjusted gage factor [Measurements, 1993]:

$$GF' = GF \times \frac{350\Omega}{350\Omega + 2 \times 2\Omega} = 98.9\%\,GF$$

which will result in increased strain reading of approximately 1.1%. For most applications, this will be an acceptable range of error, but if lead wire compensation is desired, the correction to the gage factor may be made using the above equation. The resistance for the lead wire may be estimated from the linear relationship of 1Ω per 40 feet of cable; the multiplier of 2 is in the equation to account for the fact that each strain gage has two lead wires that run the length of the cable.

If the Interface Box is expanded to accept 120Ω gages, it is possible that lead wire resistance could become an important contributor of error. Examination of the equation will show that as the resistance of the strain gage decreases, the more significant the lead wire resistance becomes in effecting the altered gage factor.

All gages shall be bonded to steel surfaces according to the manufacturer's specifications. Some additional suggestions and requirements regarding gages are:

- Use only 350Ω gages--120Ω gages are not compatible with the BCMs.
- Properly align gage in the direction of measurement. Great care should be taken to avoid skewing which will cause inaccuracy.
- Protect all gages with polysulfide protective coating system. Refer to manufacturer's instructions for application.
- Use strain relief in lead wires and sufficient tape to prevent gage damage.
- Keep documentation on gage factors (GF) for each strain gage. Many gages with different GF's look exactly alike. For example, tape a label of the GF next to the bonded strain gage.

Any strain gage of any gage factor (GF) may be used provided that the resistance of the gage is 350Ω. The system is compatible with 350Ω gages only.

3.3.2 Transducer. There is a great advantage to the employment of transducers. No adhesive bonding is required--transducers are clamped directly to a beam flange or plate. Additionally, transducers amplify the strain which can enhance small strain readings. Unfortunately, this can also be a disadvantage; transducers must be calibrated, so the stress is only as accurate as the calibration. After extended use, transducers require re-calibration to ensure proper readings. Transducers that are not properly or frequently calibrated will produce bad results. Other disadvantages of transducers include:

- Recalibration. Transducers must be periodically recalibrated.
- Limited use. Only on flat surfaces large enough for a transducer to be clamped flush, such as beam flanges.

- Proper care and handling. Strain gages must be protected against damage, and bars must be protected against disfigurement and yielding.

The connection of the strain gages to the transducer is very critical. The transducer will appear to operate properly even if the gages are improperly sequenced. The color chart in Table 2 should be followed very carefully to ensure that the colored wires are correctly coordinated with the pin letters.

Figure 17 shows how the transducer strain gages are to be connected to one another and to the connector. Two of the gages will always measure the same sign, i.e. two gages will be in tension when the other pair is in compression. These pairs are indicated in the sketch with T1, T2, C1, and C2. This does not mean that T1 and T2 will always be in tension; it is used as a labeling system to show how the gages are to be oriented with one another. Depending on the type of transducer used, all four gages may measure the same magnitude of strain, while for other transducers, one pair may measure the primary strain of the transducer and the other pair will measure the Poisson strain. For the latter case, the primary pair will be labeled as the "tension" gages and the Poisson pair will be labeled the "compression" gages.

Transducers are very simple to use in a testing situation, but great care must be taken to ensure that no errors are made during construction. Calibration will be the true test of proper wiring of the transducer. Refer to the transducer manufacturer for the best calibration method.

3.4 Portable Computer System

All software necessary to conduct tests and operate the DL is installed on the portable computer system (PCS). The computer is a Gateway Colorbook 486 with a microprocessor speed of 66 MHz and a memory capacity of 250 Megabytes of hard disk space and 4 Megabytes of RAM. The power cord and power transforming unit for battery recharging are pictured with the computer in Figure 18. All system disks and

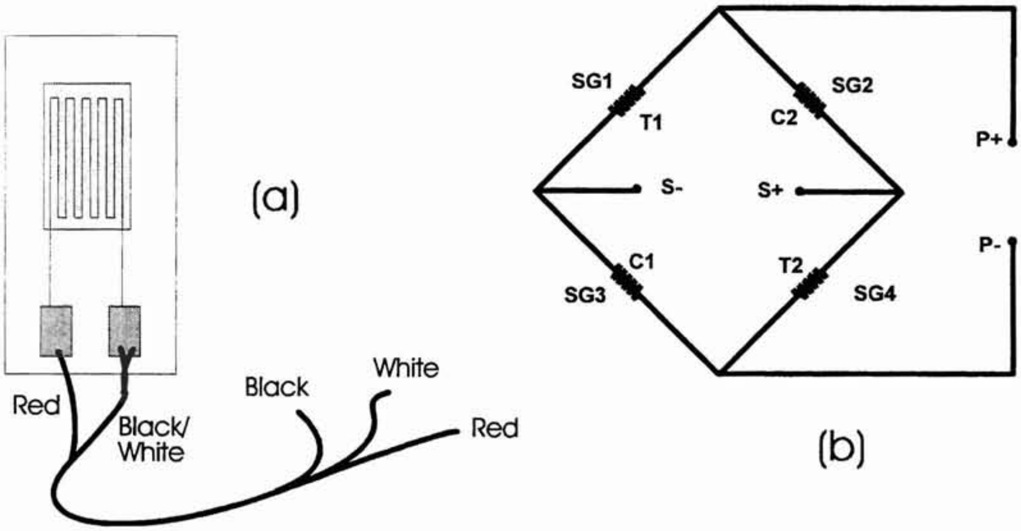


Figure 17. Strain Gage Wiring. (a) Strain gages have lead wires composed of three conductors. Lead wire assembly is identical for individual strain gages and for gages of transducers. (b) The full bridge schematic for transducers is shown once again. Reference the SG#'s with the Table 1 and Figure 14 for cable construction. The labels C1, C2, T1 and T2 are given to indicate tension and compression.

operation manuals are delivered with the computer. Information regarding the care and use of the PCS may be found in the manufacturer's operation manuals.

The battery will only last about 2 hours, so it is a good idea to always use the power cord and transforming unit when possible (i.e. inside the office). Since the PCS will be powered by the battery during testing, it should be fully charged prior to field excursions. The battery will charge only when the PCS is turned off, and can take up to 2 hours to become fully charged.

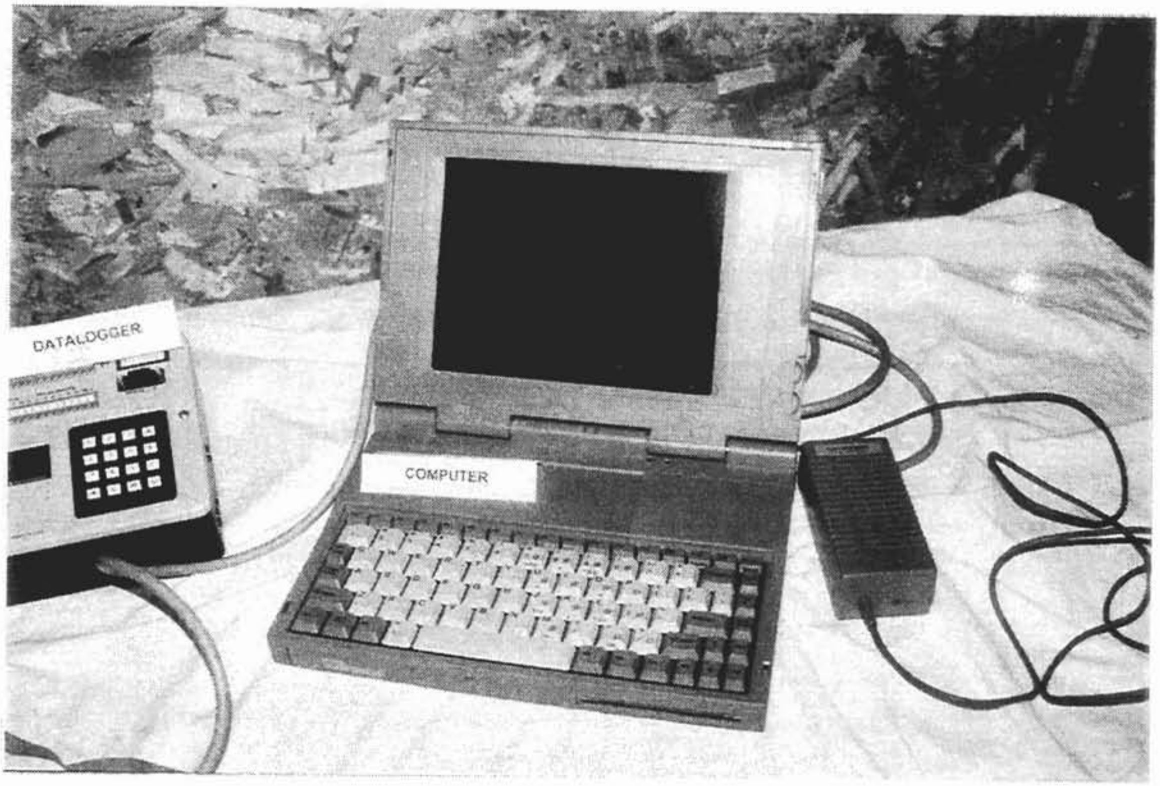


Figure 18. Portable Computer System (PCS)

3.5 SC32A

The SC32A (Figure 19) is provided by Campbell Scientific, and is required for direct link communications between the DL and the PCS. The SC32A manages file transfers from the computer to the DL and converts 21X logic to logic compatible with the computer [Campbell Scientific, Inc., 1992]. Refer to the Campbell Scientific manual for more information regarding the SC-32A unit.

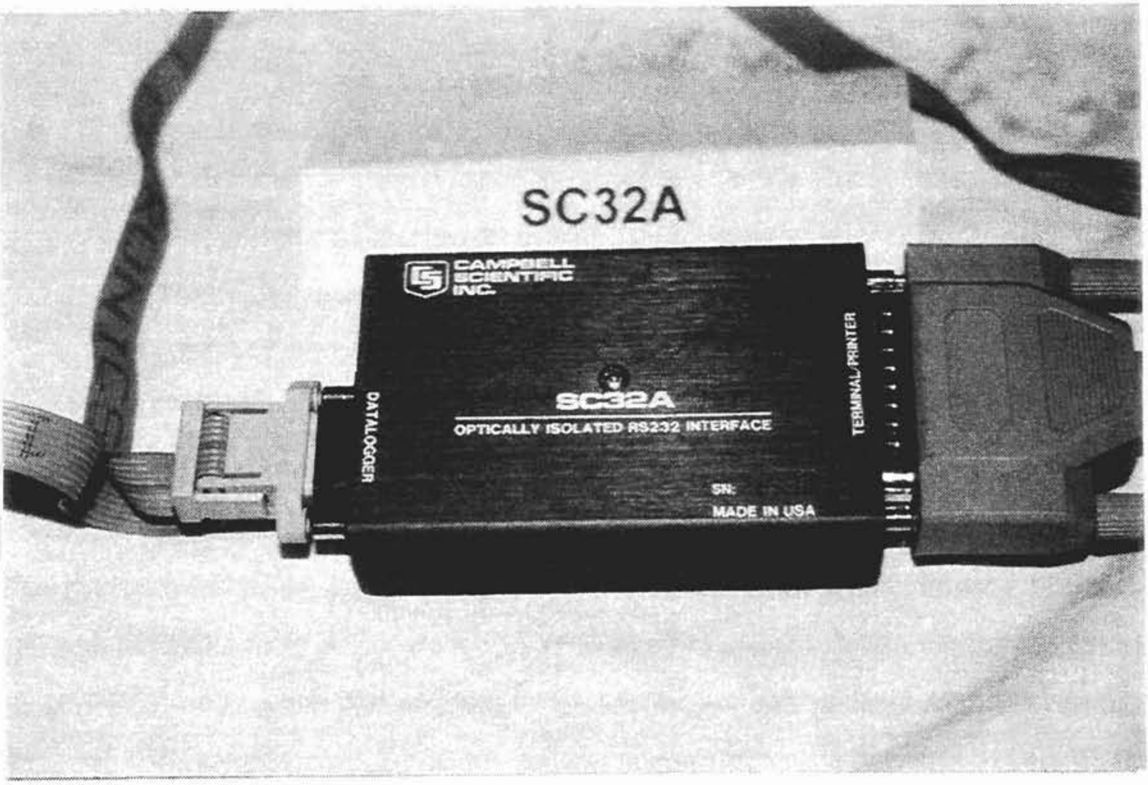


Figure 19. SC32A

CHAPTER 4

COMPUTER PROGRAMS

The testing software includes 7 program modules:

1. FRAME.BAT. A batch file that manages and executes all programs.
2. PROG.EXE. The main program for user interaction and test management.
3. GT.EXE. Campbell Scientific file called Graphterm. GT.EXE is used to download files to the DL, send the computer time, and monitor inputs.
4. TELCOM.EXE. Campbell Scientific file which is used for collecting data from the DL.
5. COLLECT.BAT. A short batch file used to execute TELCOM.EXE. It is also used as a Boolean expression within FRAME.BAT.
6. RETURN.T_F. A Boolean file used within FRAME.BAT.
7. RESET.BAT. A batch file that restores program variables.

FRAME.BAT. All programs are executed from a batch file program entitled FRAME.BAT (refer to flowchart in Figure 20). The decision to use a batch file program was made to facilitate a software conflict that occurred when collecting data. In early versions of PROG.EXE, data collection was initiated by a DOS SHELL command to call up GT.EXE, which allows the two files to be opened simultaneously. Conducting data collection this way produced a problem during the connection link between the computer and the DL.

This problem does not occur when TELCOM.EXE is executed from a batch file. Managing the software this way required more programming which in turn spawned an added benefit. The batch file required a loop to get back to PROG.EXE after a data

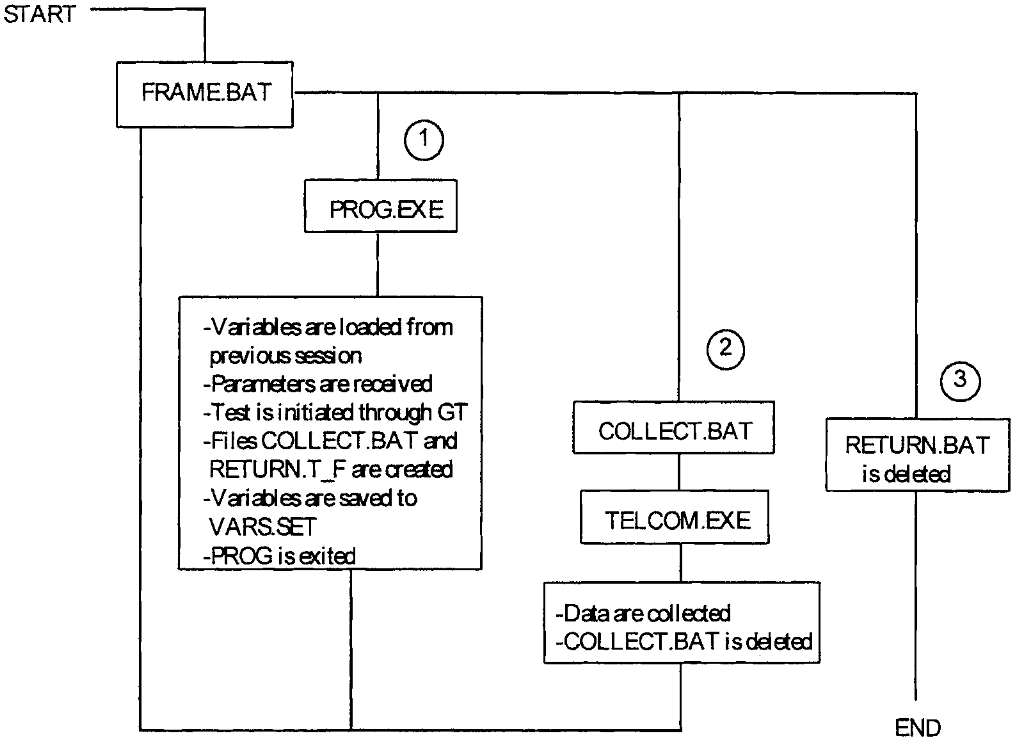


Figure 20. Flowchart for FRAME.BAT. This flowchart outlines a typical session. When FRAME is first started, it follows Path 1. If a test collection is executed, the next loop will follow Path 2. Paths 1 and 2 may be repeated until the user exits the program from the Main Menu, which will follow Path 3.

collection. When FRAME.BAT is initially run, PROG.EXE is executed. When the collection option is selected, PROG.EXE is terminated and the collection is carried through. A condition statement is needed to loop back to the line in the batch file which executes PROG.EXE. The condition is met with the Boolean file RETURN.T_F. This dummy file is created by PROG.EXE when the program is initiated and is not deleted until the user exits the program with the Quit option. This provides the conditional parameter to continue the loop. If the file RETURN.T_F exists, the batch file loops back to the beginning; if the file does not exist, the batch file is terminated.

Another problem occurs when PROG.EXE is terminated to collect data: the variables. When the program is initially executed, all variables are reset to some initial value. To solve this problem, a variable file entitled VARS.SET is used to store the values and settings of all program variables. When the program is restarted, all variables are restored from this file through a subroutine. This feature was extended to encompass every startup. Since variables remain unchanged, the user is able to implement automatic default settings. The parameters which remain the same from test to test will be saved and restored automatically.

The program module FRAME.BAT can be found in Appendix B.

PROG.EXE--Main Program. The main program, PROG.EXE, is a compiled stand alone QBASIC program which translates user Input parameters into datalogger downloadable files (extension DLD). A series of menus presents options and requests parameter inputs. On-screen instructions are provided for most menu functions and options, but some minor features are mentioned only in this text. Chapter 5 is devoted to fully explaining the operation of the main program.

PROG has automatic default settings. Every time the program is terminated, the current values of all global variables are saved to the data file VARS.SET. The next time the program is activated, these same settings are restored. For example, a particular set of Rainflow settings may be adopted as a standard for all tests. Once these settings are entered for a test, the values will remain as defaults for subsequent test Setups.

The source code for PROG.EXE may be found in the file PROG.BAS on the computer hard drive. PROG.EXE is the compiled, executable file of the source code PROG.BAS. A printed copy of the code is also available in Appendix C.

Graph Term (GT.EXE). Graph Term is one of two programs provided by Campbell Scientific to access the DL. GT is used to download files, and, unlike TELCOM.EXE, it is executed from PROG with a DOS SHELL command. There were no connection problems from downloading programs in this manner.

GT requires a station file with interfacing parameters. This station file, entitled OSU.STN, is already installed on the hard drive of the computer. Users should not encounter the station file unless the file is accidentally erased or new parameters are required. Instructions for creating or editing the station file can be found in the Campbell Scientific Manual. The parameters for this file are listed in Table 3.

Telecommunication Parameters For Station File	
Station Title:	OSU
Datalogger Type:	21X
Comm. Port:	COM1
Baud Rate:	9600
File Format:	Comma Delineated ASCII
Final Storage Area:	N/A
Interface Device:	SC32A

Table 3. Station Parameters for File OSU.STN

There are only three functions with which GT is employed:

1. Sending the computer time to the datalogger.
2. Downloading a datalogger program.
3. Monitoring Input locations.

The program GT.EXE is always run in the background of PROG.EXE. The user will only see the screen switch to different menus and displays. Knowledge of Graph Term operations are not necessary except for an occasional use of the ESC key to return to PROG.EXE.

TELCOM.EXE. TELCOM.EXE is another program included in the software provided by Campbell Scientific, and it is the collection counterpart to Graph Term.

Although GT will perform data collections, a representative from Campbell Scientific recommended the employment of TELCOM for this particular application.

TELCOM is used exclusively for data collection. A station file is required to provide for the computer the necessary information to connect to the DL. The same file, OSU.STN, is used by TELCOM and GT. When a collection is completed, all data are saved to a file with the extension DAT. The name for this file is derived from a the Maintitle parameter which is provided by the user in PROG.

COLLECT.BAT and RETURN.T_F. COLLECT.BAT and RETURN.T_F are both created by the main program. COLLECT contains the collect command line which reads:

```
TELCOM OSU /C /F [name]
```

where

- TELCOM is the program that makes the collection.
- OSU is the name of the station file.
- /C initiates the collection.
- /F assigns a name to the file of collected data.
- [name] is the assigned name which is provided by the user.

RETURN is a dummy file that is created to continue a DO LOOP in FRAME. RETURN is deleted upon program termination by the user, and COLLECT is deleted after each collection. Refer to Section 4.1 for information on how these files operate within FRAME.

RESET.BAT. If FRAME is terminated manually (CTRL^C or CTRL-ALT-DEL), the program variables may not be properly saved. An attempt to reboot FRAME may result in an error loop that can only be exited by resetting the PCS. RESET.BAT is a short program that restores a version of VARS.SET to one that will disengage the loop.

CHAPTER 5

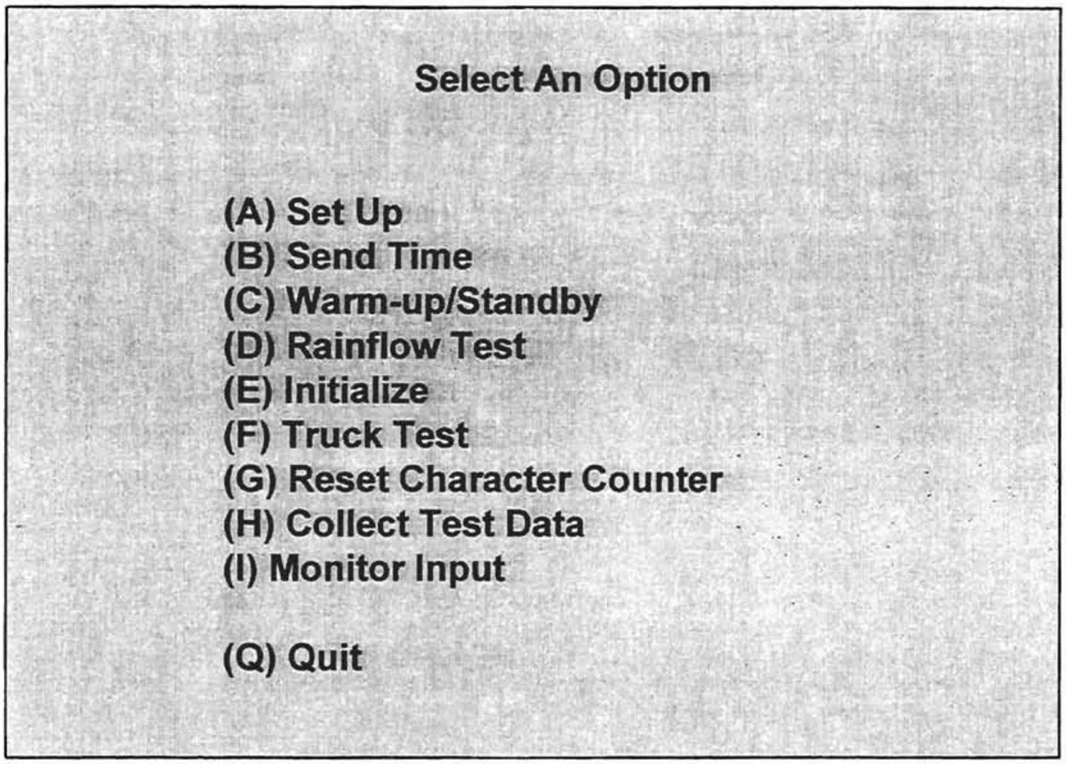
PROGRAM OPERATIONS

The DAS described in this report is a tool used in predicting the fatigue life of a bridge. It is up to the user to determine the areas of the bridge which are to be tested and the parameters of the test(s). The following sections describe each menu and each selection option from the menus.

The first menu available to the user is the main menu which is presented in Figure 21. The menu items are listed in the order of sequential execution. This does not imply, however, that all tests must be performed in this order or that all tests must include each of the steps A through I; it was designed to guide the user through the typical sequence of steps that most tests will follow. Menu selections are made by typing the letter which appears in parentheses.

5.1 Option A—Setup

To begin a new test or a new series of tests, Setup must be run. All tests require information from the user pertaining to the configuration of the measurement devices, gage factors, and Rainflow Settings. This information is entered through two menu screens. The first screen is the "Select An Option" Screen (Figure 21) which requests the title and the channel status for all 8 inputs. The second screen is the "Rainflow Settings" Screen. This menu (Figure 22), allows the user to set parameters for the Rainflow Histogram. These settings are based on command 81 from the 21X Datalogger user's manual. If the Setup is for a Truck Test, the Rainflow Settings are irrelevant and may be bypassed by typing ENTER for each prompt.



Select An Option

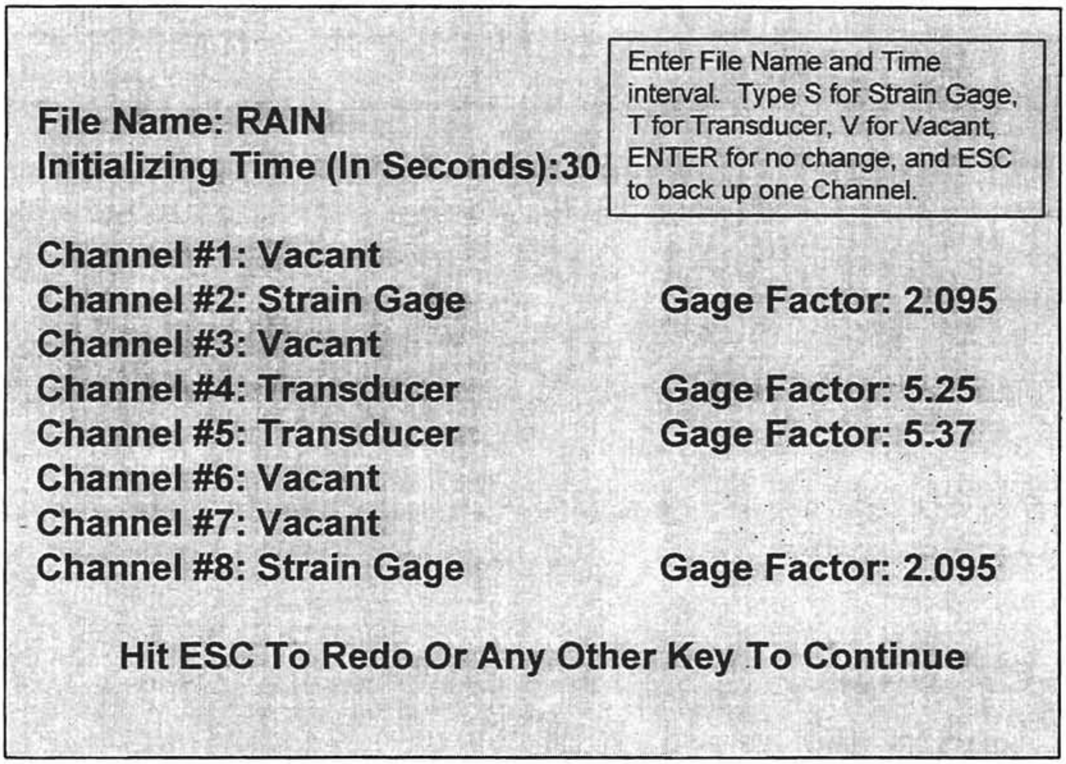
(A) Set Up
(B) Send Time
(C) Warm-up/Standby
(D) Rainflow Test
(E) Initialize
(F) Truck Test
(G) Reset Character Counter
(H) Collect Test Data
(I) Monitor Input

(Q) Quit

Figure 21. Main Menu Screen. User enters the letter corresponding to the desired function. Typing Q or the ESC key will exit the program to DOS.

Once all information is entered, it is saved to a file to be accessed each time the main program (PROG) is reactivated. Note that these previous entries appear as defaults (see menu in Figure 22). This is helpful if the same or nearly the same set of information is required for consecutive tests. The user may keep any or all of the default settings by typing the ENTER key with no entry change at each prompt.

5.1.1 Title and Initializing Time. Every test requires a title for file assignments. Any 7-character name may be used, provided that it conforms to the normal DOS file name rules (see DOS manual). Useful titles contain some significant information related to the structure and/or the test. A counting character, which is appended to all output



```
                                         Enter File Name and Time
                                         interval. Type S for Strain Gage,
File Name: RAIN                          T for Transducer, V for Vacant,
Initializing Time (In Seconds):30        ENTER for no change, and ESC
                                         to back up one Channel.

Channel #1: Vacant
Channel #2: Strain Gage                  Gage Factor: 2.095
Channel #3: Vacant
Channel #4: Transducer                   Gage Factor: 5.25
Channel #5: Transducer                   Gage Factor: 5.37
Channel #6: Vacant
Channel #7: Vacant
Channel #8: Strain Gage                  Gage Factor: 2.095

     Hit ESC To Redo Or Any Other Key To Continue
```

Figure 22. Setup Menu. This menu is used for both Rainflow and Truck Tests.

files, provides a labeling system for output files of multiple tests with the same title (see Section 5.7).

The initializing (INIT) time is the duration of the initializing or zeroing test. The data collected for this test will be used to balance the measurement readings for future Truck Tests only; the INIT time does not apply to Rainflow Tests. For more information regarding initialization, refer to Section 5.5.

5.1.2 Channel Assignments. Channel Assignments allow the user to designate an Input channel as a strain gage ("S"), transducer ("T"), or vacant ("V"). Additional options are the ENTER and ESC keys. ENTER will accept the default status and ESC will back the prompt up to the previous Input for editing.

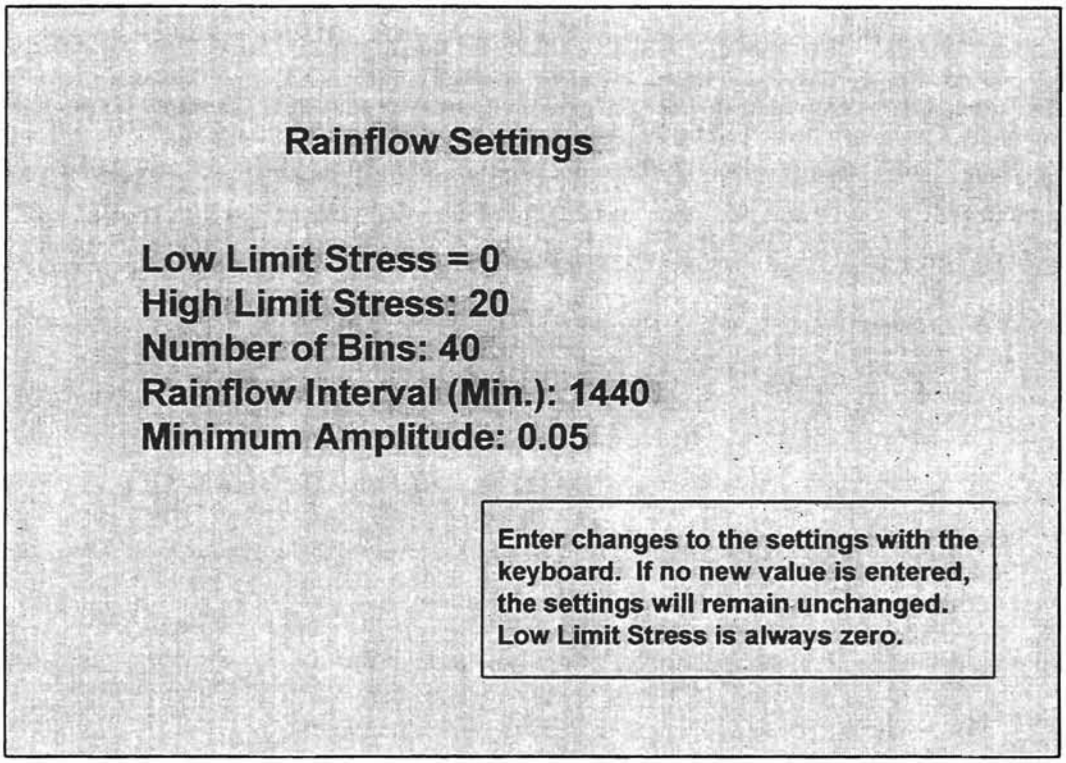
If "S" or "T" are typed, a prompt for the gage factor (GF) will be given. The GF for strain gages are provided by the manufacturer. For transducers, the GF is a calibration factor that must be obtained by a calibration test. Additional information on measurement devices and GF's may be found in Section 3.3.

5.1.3 Low Limit Stress. There is no prompt for the Low Limit value; the default is restricted to zero. Low Limit is used as a boundary for calculating the number of divisions (bins) in the Rainflow histogram. This entry has been left in the settings for future modifications. Should there be a need to permit changes to the Low Limit, the program code may be altered with only a few line amendments.

5.1.4 Upper Limit Stress. This parameter determines the size of the histogram. The Upper Limit should be greater than the largest stress value expected. The effectiveness of the histogram may be optimized by choosing a value close to the actual maximum stress. Results from a Truck Test can be helpful in this selection.

5.1.5 Number of Bins. A "counting matrix" is established by the datalogger based on the upper limit, lower limit, and number of amplitude bins. The user should be selective in choosing each of these parameters to obtain optimum output form. An amplitude range is the value of the upper limit, and the number of amplitude bins (maximum of 50) is the number of histogram divisions the DL will use to count cycles. The size of each bin is calculated by dividing the amplitude range by the number of amplitude bins. Cycles which have an amplitude greater than the Upper Limit are counted in the last bin.

Adopting a standard bin width can simplify these calculations. Once the Upper Limit is determined, the Number of Bins will be the quotient of the Upper Limit and the Bin Width.



Rainflow Settings

Low Limit Stress = 0
High Limit Stress: 20
Number of Bins: 40
Rainflow Interval (Min.): 1440
Minimum Amplitude: 0.05

Enter changes to the settings with the keyboard. If no new value is entered, the settings will remain unchanged. Low Limit Stress is always zero.

Figure 23. Rainflow Settings Menu

5.1.6 Rainflow Interval. Rainflow Tests are divided into test intervals. Data points are maintained in intermediate storage during the interval. At the end of the interval, the data are transferred to final storage until data are collected from the DL. It is recommended that multi-day tests have a test interval of 24 hours (1440 minutes). This will encompass the majority of Rainflow Tests.

5.1.7 Minimum Amplitude. Very small cycle amplitudes due to vibrations are of little consequence to fatigue life prediction, and may be viewed as noise. Counting such cycles will do nothing more than occupy memory in the DL final storage. The Minimum Amplitude parameter is used to eliminate these small cycles which will distort the effective stress range calculation and quickly consume the DL memory during testing.

This setting is a way of informing the DL of the triviality threshold. A quick view of the Monitor Input while no traffic is on the bridge is a good way to determine the order of magnitude of the minimum peak-valley distance. Values must be less than the bin width (see calculation example in Figure 24).

Lower Limit = 0 ksi

Upper Limit = 20 ksi

Amp Bins = 40

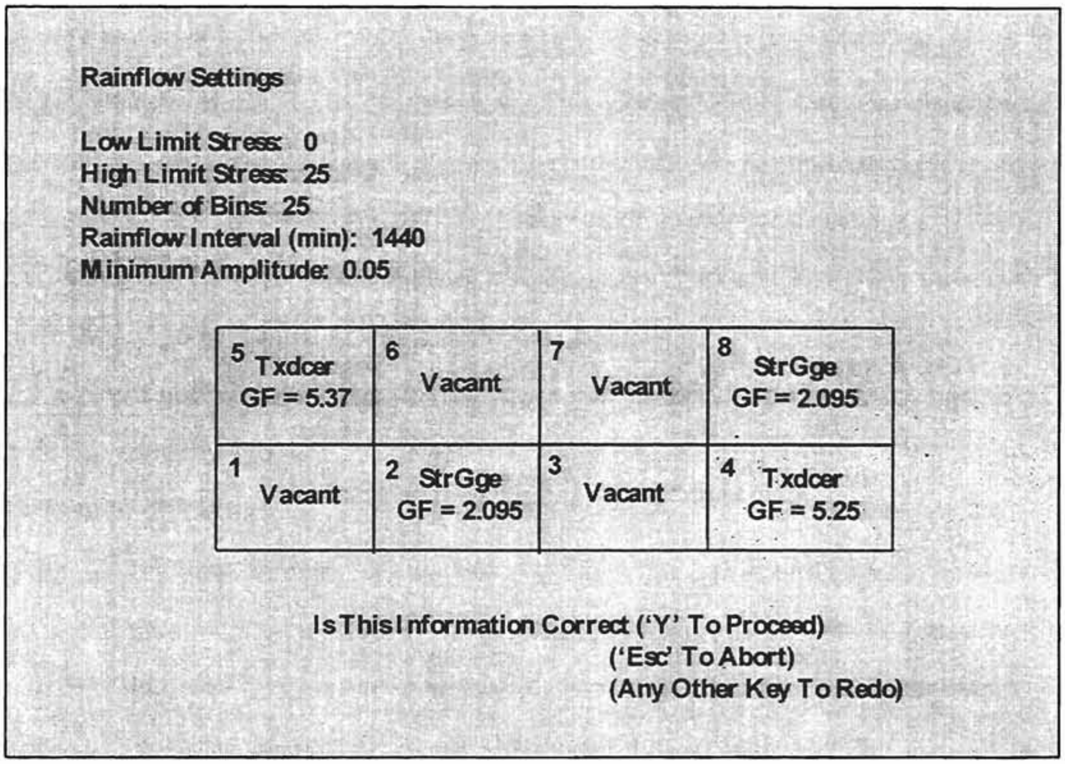
Bin width = (20 - 0) / 40 = 0.5 ksi

Figure 24. Bin Calculation Example. Each bin will have a width of 0.5 ksi. A cycle with an amplitude of 16.8 ksi will fall into bin number 34 which counts amplitude cycles between 16.5 and 17.0. A cycle with an amplitude of 22.1 will fall into bin number 40 which counts all amplitudes greater than 19.5 ksi.

5.1.8 Input Summary. The last screen of Setup displays the Input information. The graphic, which may be seen in Figure 25, resembles the front of the interface box with Input 1 in the lower left corner and Input 8 in the upper right corner. Typing Y or ENTER will accept the test parameters, ESC will abort Setup, and any other key will allow the user to redo Setup.

5.2 Option B—Send Time

The second option from the Main Menu (refer to Figure 21) is the Send Time. The internal clock of the DL may be synchronized with the computer clock with this command. Before the time is sent to the DL, the user is given the option to change the time and date of the computer clock. Once the time is sent, an internal flag is set high.



Rainflow Settings

Low Limit Stress: 0
High Limit Stress: 25
Number of Bins: 25
Rainflow Interval (min): 1440
Minimum Amplitude: 0.05

5 Txdcer GF = 5.37	6 Vacant	7 Vacant	8 StrGge GF = 2.095
1 Vacant	2 StrGge GF = 2.095	3 Vacant	4 Txdcer GF = 5.25

Is This Information Correct ('Y' To Proceed)
('Esc' To Abort)
(Any Other Key To Redo)

Figure 25. Input Summary Screen

Both the Rainflow and Truck Test subprograms include an algorithm to determine whether the time has been sent. If the flag is low, the computer will automatically download the time with the test program. Because of this feature, the Send Time option must only be selected if the DL is powered off and on during a test session. It is particularly important that this be done before Rainflow tests since the DL clock controls the interval starting and stopping times.

5.3 Option C—Gage Warm-up/Standby

Strain gages (and transducers) will report *drifting* values for about the first 20 minutes of excitation. To avoid this drift, it is recommended that gages be activated for a period of time to allow the circuitry to warm up to a steady-state position where readings will be more reliable. The amount of time required for this warm-up period may vary, but is usually about 20 to 30 minutes. This option allows the user to select the time interval and immediately begin the warm-up period. A dummy program is sent to the DL to activate the strain gages with excitation voltage only; no readings are stored in memory, which will help conserve battery life. The computer will notify the user with an on-screen message when the warm-up time has elapsed.

The amount of drift decreases as the gages warm up. If time does not permit a full 20 minutes to properly prepare the gages, a shorter period of time (5 to 10 minutes) can still help reduce drift errors.

Option C can also be used to maintain a current through the gages between tests, allowing the gages to remain active. The Standby option is similar to the Warm-up option, except that the time interval is indefinite. Pressing ESC will return the user to the Main Menu. The gages will remain in Standby mode until a test is initiated.

5.4 Option D—Rainflow Test

In order to calculate fatigue life, an effective stress value must be measured. This stress value is best determined from a histogram of stress cycles over a period of time, usually in days. The DL has a Rainflow Histogram algorithm that will collect data to calculate the effective stress value. As with the Truck Test, locations of the structure which are to be measured must be determined by the test supervisor. Once these locations are determined and the appropriate devices are properly in place, the Rainflow Test may begin.

To perform this test, the user must first run Setup. Selecting D from the Main Menu will begin the Rainflow Test. An Input Summary Diagram will be displayed to allow verification of the test parameters. If no corrections are necessary, the test will begin following the continue option.

It is recommended that tests extend for a duration of several days with 24-hour (1440-minute) intervals. The maximum test duration is dependent on number of bins, number of channels, interval lapse, and battery life. The rate of the DL memory consumption is increased with number of bins, number of channels, and interval lapse; increasing each of these will increase the number of data locations that must be saved in final storage. If all of the memory locations are filled, the DL will overwrite the data beginning of the first set. For a rainflow interval of 24 hours, test duration will be restricted by the battery life which is approximately 2 weeks. To calculate the amount of memory consumed by a test, the following formula derived from the memory capacity of the DL and the memory used by a Rainflow Test may be used:

$$16926-\left[4+\left(2\times\text{no. of bins}\right)\times\text{no. of channels}\times\text{no. of intervals}\right] \qquad (5.1)$$

The value of 16,926 is the final storage capacity; the remaining numbers and variables are the amount of memory used. The integer 4 accounts for a labeling value and the time the data record was made. Information on DL memory capacity may be found in the Campbell Scientific, Inc. 21X Operator's Manual.

Once the test parameters have been sent to the DL and the Main Menu returns to the screen, the Monitor Inputs option may be selected to verify that all of the channels are functioning properly. If any entry registers ±9999, then there is an open circuit on that Input. All connections should be checked until the short is located and mended. Once the test is satisfactorily in progress, the program may be exited and the computer powered down.

Although the DL is reading measurement devices and recording data, the first interval will not begin until 12:00 midnight. The data collected between the time the test is initiated and midnight is a partial interval, and all data collected during this time will be dismissed.

The data that are collected are processed into two forms of output. The tabulated data include the calculated value of the effective stress level for each channel interval. The DAT files contain graph plotting data and values for the effective stress level.

5.5 Option E—Initialize

Truck Tests require that all inputs be initialized to zero. All traffic should be blocked from the structure during this process. The initialization creates a reference which is the basis of all measurements. When an Input reports a stress of 3.5 ksi, this is interpreted as a stress change of 3.5 ksi from the initialized state. After traffic is stopped and Setup has been completed, the initialization (INIT) may begin. INIT time, which is specified in the Setup, can range from 10 seconds to 5 minutes. Factors to consider when determining the INIT time are bridge component sensitivity, wind effects, and previous experience. Initialization is essentially a Truck Test with no loads applied. The data values collected over the INIT time are averaged and incorporated into the Truck Test as offsets. A longer period of time is required if the INIT measurements fluctuate widely. Monitoring the inputs immediately after initialization is one way to evaluate measurement stability.

Collection is automatically executed after the initialization time has elapsed. Once the Main Menu reappears on the computer screen, the Truck Test may proceed. Initialization is required for Truck Tests Only; running the Initialization prior to a Rainflow Test will have no effect on the test.

5.6 Option F—Truck Test

An important tool in bridge design is the influence line diagram. The Truck Test will collect data to adequately construct an influence line for bridge locations which have been equipped with measurement devices. The information gathered from such a test is useful in constructing a computer model of the bridge under investigation. The model may be adjusted and fine tuned to reproduce the same results as the real bridge.

The Truck Test option of the DAS is a feature that provides the data necessary to fine tune the model. A Load Truck of known gross weight shall be driven over the bridge on a path specified by the test supervisor. As the truck is driven across the bridge, the DL will collect data at a maximum rate of 0.025 Hz. These data will be processed into two files. The first is a table that may be printed out with a word processor; the second is a data file which may be used for graphical output.

Before the actual test begins, the measurement devices must be initialized to a zero value (see Section 5.5). Once this activity is completed, the Main Menu screen will reappear.

When option F is selected, an Input Summary diagram will be displayed (refer to figure 25). A prompt will be given to allow the user to abort the test if corrections to the Inputs need to be made. If none are required, the test may begin by pressing any key. The computer will then download the program to the DL and return the user to the Main Menu. At this time, the Load Truck should be driven over the bridge at a constant speed. When the test is completed, the Collect Data option (Option H) should be selected immediately. Delay may cause the first blocks of data to be overwritten by new data. A Truck Test consisting of 8 Input devices operating at maximum DL frequency rate (3 measurements per second, default) can last 3 minutes, 55 seconds. This may be linearly adjusted to tests with fewer devices, i.e. a test with 4 devices may run 7 minutes, 50 seconds.

5.7 Option G—Reset Character Counter

Character counters are used for data collection files. These files are named after the test Maintitle with extensions TAB and DAT. The character counter provides distinction between different tests on the same bridge without requiring a new title for each test. Option G will reset the counter to the character A which will increment alphabetically until it is reset again, or until the letter Z is reached.

5.8 Option H—Collect Test Data

Once a test has been activated, it is no longer necessary for the computer to be connected to the DL. Any time the program is terminated, all parameters of the program are saved to a file. This is analogous to a bookmark in a novel; when the program is reactivated, it returns to the Main Menu as if it had never been terminated. This allows the user to run a long test (usually a Rainflow Test) without the computer present. When a test is completed, the computer may be returned to the DL ready to collect the test data.

To do this, execute the FRAME program and select the Collect Data option, H. Both Rainflow and Truck Tests generate two files when the data are collected. Tabular files are given the extension TAB and raw data files are given the extension DAT. The titles for these files will be the same as the Main Title provided by the user in the Setup mode. Since more than one collection may be necessary on a project, a counter character is added to each pair of collection files. For example, if the Main Title were "US177", the first collection files might be "US177A.TAB" and "US177A.DAT". The letter A is the counter character and will automatically increment alphabetically with subsequent collections. The user may reset this value to A with the Reset Counter Character option. This is only a bookkeeping parameter to aid in data file organization, and it is up to the user to decide when and if the counter should be reset. If the counter reaches the letter Z, it will roll over to the letter A on the next collection.

5.9 Option I—Monitor Input

Once a test has been initiated, the Monitor Input option is a good way to verify that all of the equipment is properly connected. Monitor Input is a function available through the GT program. When this option is selected, the user may be asked to enter a file name that contains the labels. Press the ENTER key to accept the default name. The Monitor screen will display 11 channels with values to the right. The Input numbers which are being used by the test do not necessarily directly correspond to the channel numbers on the screen. For example, if Inputs #2, #4, #5 and #8 are being used, channels 1 through 4, respectively, will show the stress readings, and channels 5 through 11 will be blank. An open circuit will display a ±9999.

To return to the Main Menu, type the ESC key followed by Q.

5.10 Option Q—Quit

Typing Q or ESC will terminate the program. All variable settings are saved to a data file for variables. When the program is reactivated for the next session, these settings will be restored as though the program had never been terminated.

CHAPTER 6

EXAMPLE INPUT

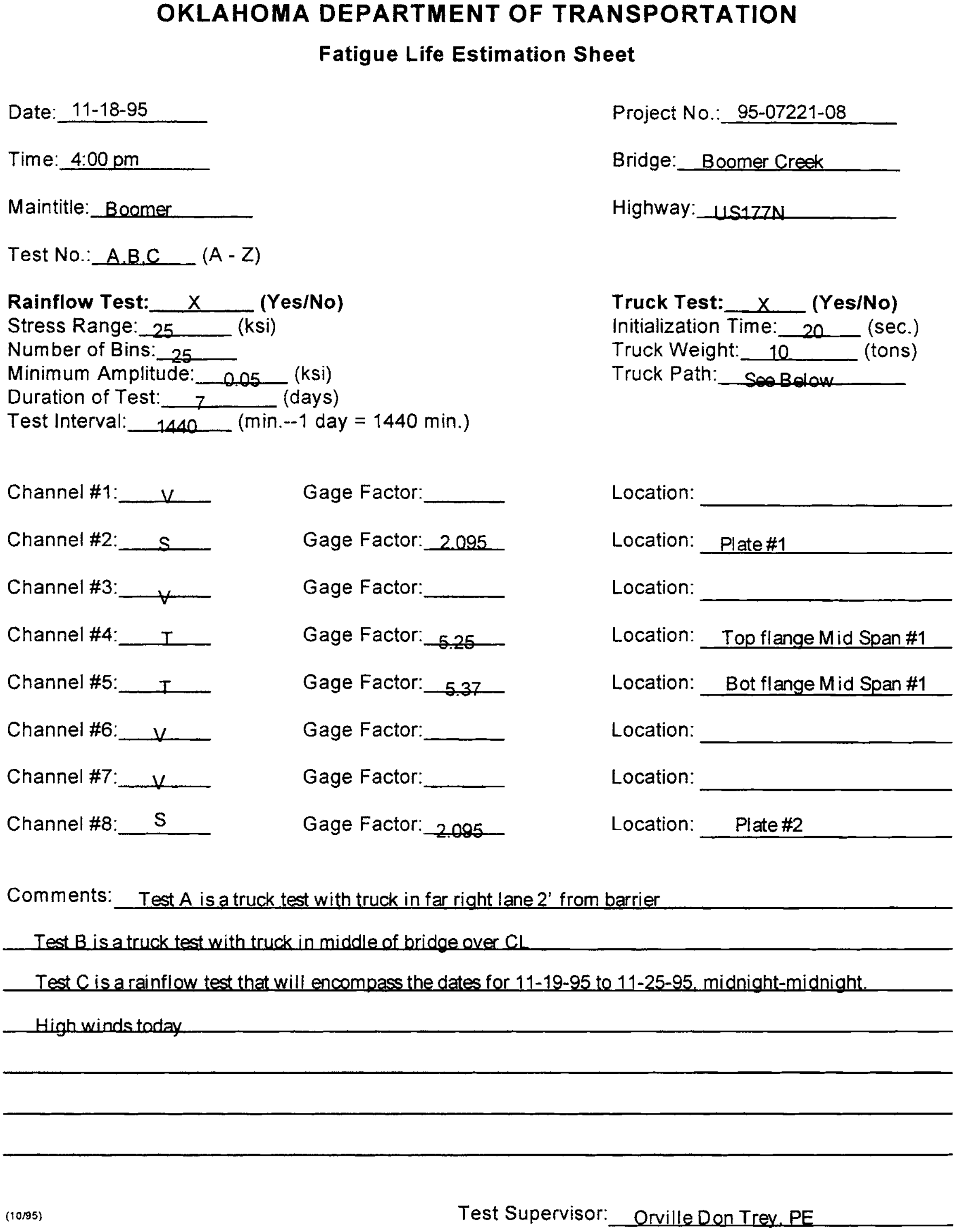
This chapter is devoted to presenting a test example. A total of four measuring devices will be used--two Transducers and two Strain Gages. The example will include both a Rainflow Test and a Truck Test. All commands and instructions are set in bold type to aid the reader in following the key strokes of the example.

6.1 Test Form

Before a test or a series of tests, a Fatigue Life Estimation Sheet (FLES) should be filled out. In some instances, the entire sheet may be completed before leaving for the bridge site. In other cases, it may be appropriate to complete the sheet at the site. Regardless of how or when the sheet is filled out, it should be completed in order to maintain a record of the test. This sheet is intended to be used as a bookkeeping tool. Comments at the bottom of the FLES will be beneficial when test results are analyzed and may prevent trips back to the bridge for re-tests. Most items on the sheet should be self-explanatory. Figure 26 shows a FLES that has been filled out for the example test.

6.2 Test Setup

Power up the PCS and **type FRAME** at the C:> prompt. Follow the instructions to the Main Menu. At some point before the test begins, the time must be downloaded to the DL. If the Setup is completed ahead of time in the office, there is no need to send the time. This example, however, will assume that the test is conducted in the field immediately following the Setup routine.



OKLAHOMA DEPARTMENT OF TRANSPORTATION

Fatigue Life Estimation Sheet

Date: 11-18-95

Project No.: 95-07221-08

Time: 4:00 pm

Bridge: Boomer Creek

Maintitle: Boomer

Highway: US177N

Test No.: A,B,C (A - Z)

Rainflow Test: X **(Yes/No)**
Stress Range: 25 (ksi)
Number of Bins: 25
Minimum Amplitude: 0.05 (ksi)
Duration of Test: 7 (days)
Test Interval: 1440 (min.--1 day = 1440 min.)

Truck Test: X **(Yes/No)**
Initialization Time: 20 (sec.)
Truck Weight: 10 (tons)
Truck Path: See Below

Channel #1: V	Gage Factor:	Location:
Channel #2: S	Gage Factor: 2.095	Location: Plate #1
Channel #3: V	Gage Factor:	Location:
Channel #4: T	Gage Factor: 5.25	Location: Top flange Mid Span #1
Channel #5: T	Gage Factor: 5.37	Location: Bot flange Mid Span #1
Channel #6: V	Gage Factor:	Location:
Channel #7: V	Gage Factor:	Location:
Channel #8: S	Gage Factor: 2.095	Location: Plate #2

Comments: Test A is a truck test with truck in far right lane 2' from barrier

Test B is a truck test with truck in middle of bridge over CL

Test C is a rainflow test that will encompass the dates for 11-19-95 to 11-25-95, midnight-midnight.

High winds today

(10/95)

Test Supervisor: Orville Don Trey, PE

Figure 26. Fatigue Life Estimation Sheet (FLES)

Type B to send the time. The next screen will display the computer time and date. If corrections are necessary, **type T** and enter the time in the same format as seen on the screen (only hour and minute required). Follow with the **ENTER** key and enter the correct date. If the ENTER key is depressed with no entry at all for time or date, no changes will be made.

The screen should quickly change to the GT program. The time send is completely automatic, but the user should watch to verify the connection. If time does not send, a message will be displayed stating so. Check all connections to ensure the DL is properly connected.

Refer to the FLES form and note that the letter **A** is written in the Test No. blank. This is an indication the reset must be selected. Once the screen has returned to the Main Menu, **type G** to reset the character counter. This will ensure the first data collection will be titled HWY177SA.

To begin the Setup, **Type A**. The Setup screen will appear and await the Maintitle entry. **Type HWY177S** in this blank. This title was chosen since the bridge under testing is located on U.S. Highway 177 South. The next prompt is for the Initialization Time. This test will use a time period of 1 minute, so **enter 60** seconds.

The next 8 prompts are for the Inputs. The FLES shows that Inputs #2, #4, #5, and #8 will be used. These were chosen randomly to show the Inputs do not necessarily need to be used in any order. The prompt will move down the screen after each Input. **Type V** for each of the vacant inputs, **type S** for Inputs #2 and #8 followed by GF **2.095**, and **type T** for Inputs #4 and #5 followed by GF **5.25** and **5.37**. If any errors are made in the parameter Input, the **ESC** key will move the prompt back up the screen. For example, if the prompt is at Input #8 and there is a mistake on Input #2, type the ESC key 6 times until the prompt reaches Input #2 and make the necessary corrections. Type ENTER key 6 times to move the prompt back to Input #8. Notice that the parameters for Inputs #3 through #7 remain unchanged when the ENTER key is used.

Experiment with the ESC key, ENTER key and other commands for the Setup menu to learn the operation of each edit command. After all information has been correctly entered, move on to the Rainflow Settings Menu. The High Limit Stress is **20** ksi, and the number of bins is **40**. This will create a histogram with a bin width of 0.5 ksi. The Minimum Peak-Valley Amplitude is 0.05 ksi which is 10% of the bin width. This is a good estimate, but it should be checked with Monitor Input prior to the test. These settings apply only to Rainflow Tests and will not affect the Truck Test in any way. The user may wish to do the Setup for both tests at one time (as shown above) or to do a separate Setup for the Rainflow Test; either way is acceptable.

Once the Rainflow settings have been entered, the display screen will reproduce the Setup parameters. If all information is correct, **type Y or the ENTER key** to continue or type any other key to redo the Setup.

6.3 Truck Test

The comments line of the FLES states that the first Truck Test (A) will be a 25-ton truck driven along the left lane 1 foot from the parapet wall. Once the truck is in place above the bridge abutment and traffic is closed off from the bridge, initialization may begin. From the Main Menu, **type E**. The initialization program will be downloaded and executed automatically by the computer program. A timer will be displayed to count off initializing time.

When this time has elapsed, the Main Menu will return to the screen. **Type I** to Monitor the Inputs. If a prompt appears to Input a file name for labels, **type the ENTER key** and proceed. Verify that all 4 inputs are reading stresses properly. The numbers should not necessarily read zero values since they have yet to be zeroed, nor should the readings necessarily be equal to each other. A properly functioning measurement device will display some value and fluctuate within a reasonable stress range. If a measurement device is malfunctioning, a value of ±99999 will be shown. This is an indication of an open circuit, and the test should be aborted to repair the device. To exit the Monitor, **type the ESC key followed by Q**. The Main Menu should appear on the screen.

should be a list of the repeated expression "512F" in the left column of the screen. After a successful collection, this process will be quite evident. If the collection is not successful, select the collection option from the Main Menu again to retry the attempt, and make note of the incident on the FLES comment line along with the new character counter associated with the collection. Although this is quite infrequent, it does occur due to communication misses between the PCS and the DL.

An unsuccessful collection may be indicated by an error message during the DL connection routine. Either the CTRL^C key combination or a warm boot (CTRL-ALT-DEL) will end the collection routine. To prevent a repeated collection error, it will be necessary to run the RESET file before restarting FRAME. RESET will remove the collection parameters from the variables file, VARS.SET. When this step has been completed, a second attempt to collect the data may be carried out.

CHAPTER 7

SUMMARY AND CONCLUSION

In order to make better evaluations regarding the remaining fatigue life of existing bridges, ODOT has funded the research and development of a system capable of measuring in-service bridge stresses. This Data Acquisition System (DAS) has been constructed and produces output information than can be used to estimate the remaining fatigue life of bridges.

The DAS uses the 21 X Datalogger by Campbell Scientific, Inc. to make strain measurements, convert these measurements to stress values, and store the data values into local memory until collected by a computer system. A computer program was written to operate the datalogger and to process the collected data into output files in tabular form for referencing and in raw data form for graphing.

A stress measurement accuracy calibration was conducted for this system. The results from the test reveal a nominal error of 0.25%. Measurement accuracy of this magnitude signifies that the accuracy of the system is dependent on the quality and accuracy of the measurement devices and how well they are installed. Quality results are possible to obtain with the DAS, provided that care is taken in field implementation.

The two basic functions of the DAS are effective stress range estimation and in-service stress influence line generation. These functions are performed through the Rainflow Test and Truck Test options, respectively. The effective stress range is used to estimate the remaining fatigue life of the bridge. The results from such a test are incorporated into an equation which is comprised of other bridge characteristics. The second function, the in-service stress influence line, is useful for bridge computer models. The data reported from the Truck Tests can be used to fine-tune the bridge model to become a more realistic facsimile of the real bridge in its response to loading. This is useful for comparison of the stresses which the bridge is designed to resist and the stresses which the bridge actually experiences.

The next step for the DAS is field implementation and “Beta” testing. Beginning in 1996, ODOT will use the DAS and evaluate the system. Although thorough laboratory testing has been conducted to eliminate all programming errors, there may be minor modifications required to enhance the ability of this system to meet the needs of ODOT with optimum performance. Revised versions of the software program will be available through the OSU School of Civil and Environmental Engineering.

REFERENCES

American Association of State Highways and Transportation Officials. *Standard Specifications for Highway Bridges.* Washington, DC, 1995.

Barsom, J. M., and S. T. Rolfe. *Fracture & Fatigue Control In Structures.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1987.

Campbell Scientific, Inc. *21x Micrologger Operator's Manual.* Campbell Scientific, Inc., 1992.

Doebelin, Ernest O. *Measurement Systems: Application and Design.* McGraw-Hill Publishing Company, 1990.

Measurements Group, Inc. *120 Ohms and Holding. Experimental Stress Analysis Notebook.* Raleigh, NC, April 1993.

Moses, F., C. G. Schilling, and K. S. Raju. *Fatigue Evaluation Procedures for Steel Bridges.* National Cooperative Highway Research Program Report No. 299. Transportation Research Board. Washington, DC. 1987.

Post, G. J., K. H. Frank, and B. A. Tahmassebi. *Estimating Residual Fatigue Life of Bridges.* Research Report 464-1F. Center for Transportation Research Bureau of Engineering Research, The University of Texas At Austin, Austin, TX, 1988.

APPENDIX A

DAS ACCURACY TEST

The DAS was tested with a strain calibrator for measurement accuracy. The calibrator is a device that produces known strain levels in increments of 50$\mu\epsilon$. This instrument was installed to one of the DAS Inputs to determine the ability of the system to accurately measure stress. The DAS measures strain values and converts them to stress values by a multiplier of 29,000ksi (Modulus of Elasticity, E). The calibrator strains were converted to stresses by the same method for purposes of comparison with the DAS.

Figure 27 is a graph of the DAS measurements and the calibrator input. Note that both plots are virtually collinear. Figure 28 is a bar graph showing the differences between the two measurements. Note that both instruments are nearly equal for all measurements. The average difference of each measurement was 0.25%. Table 4 provides the numerical input of the test.

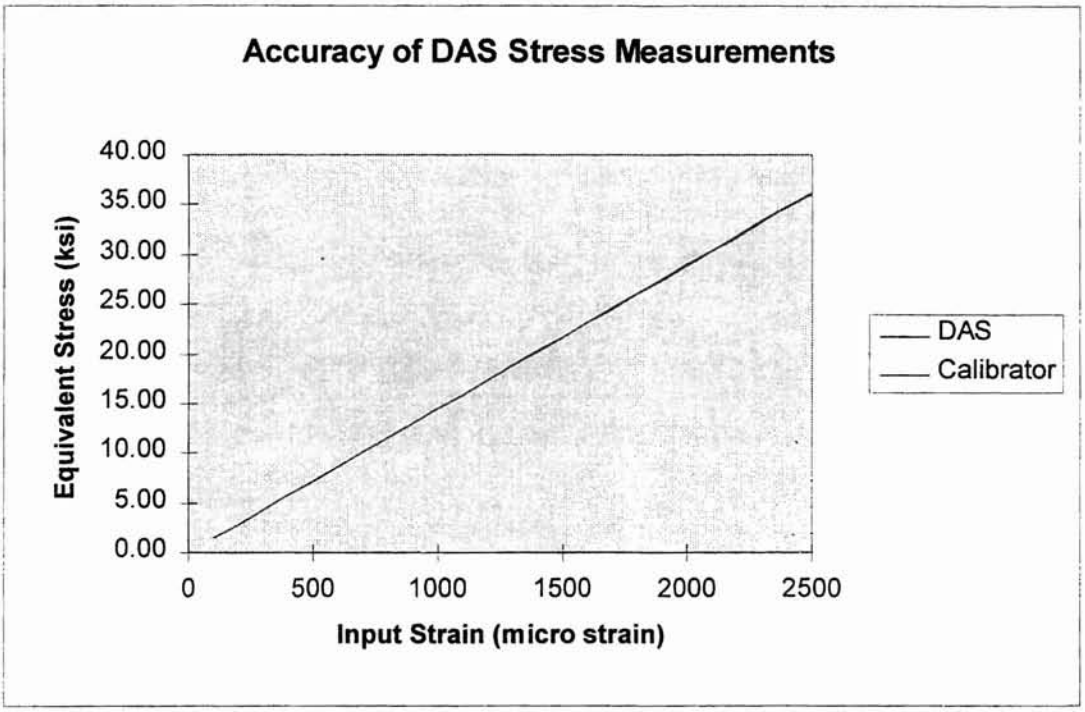


Figure 27. Linear Plot of DAS Measurements. Note that lines are collinear.

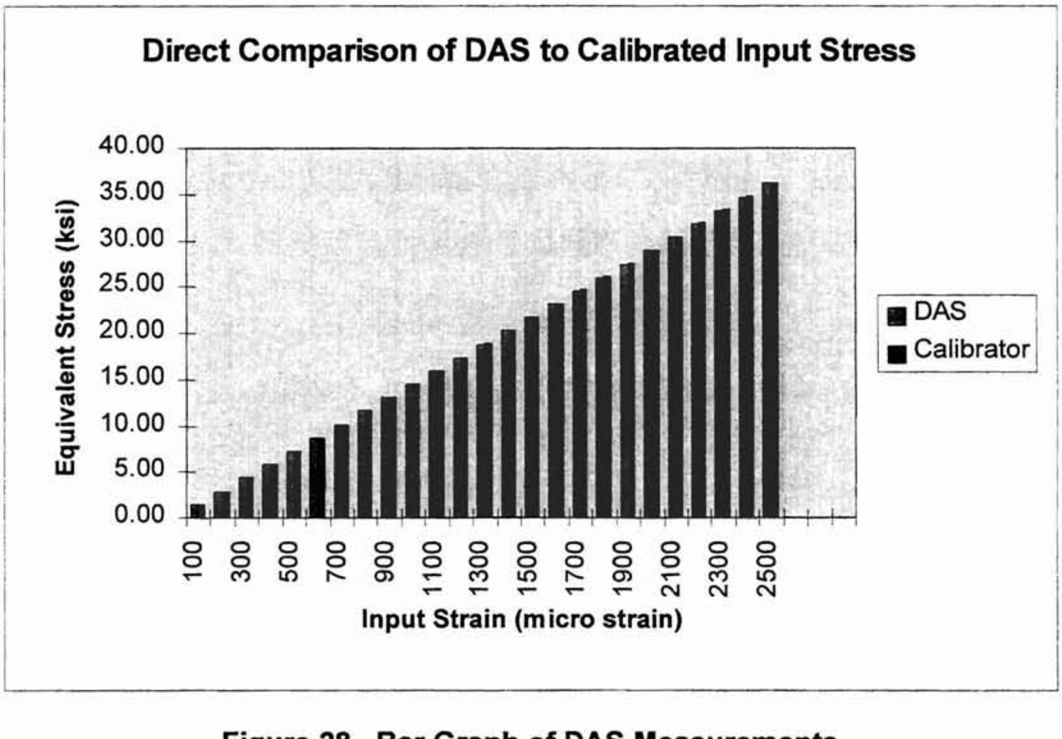


Figure 28. Bar Graph of DAS Measurements

Input	Calibrator	DAS
(micro-strain)	(ksi)	(ksi)
100	1.45	1.45
200	2.90	2.89
300	4.35	4.34
400	5.80	5.78
500	7.25	7.24
600	8.70	8.68
700	10.15	10.13
800	11.60	11.58
900	13.05	13.02
1000	14.50	14.47
1100	15.95	15.91
1200	17.40	17.36
1300	18.85	18.81
1400	20.30	20.25
1500	21.75	21.70
1600	23.20	23.15
1700	24.65	24.59
1800	26.10	26.04
1900	27.55	27.48
2000	29.00	28.92
2100	30.45	30.38
2200	31.90	31.81
2300	33.35	33.25
2400	34.80	34.70
2500	36.25	36.15

Table 4. Input Data for Calibration Test

APPENDIX B

SAMPLE DATA OUTPUT

A lab test, entitled "ODOT," was conducted to produce sample output. A 1.25" x 0.25" steel bar with a strain gage bonded on one face was subjected to several cycles of axial load. Both Truck and Rainflow Tests were simulated by applying load reversals. Tables 5 and 6 present the load history for each test.

The following sections present the output as it would appear in the files generated by the DAS. Each test generates two files: a raw data file with the extension DAT and a tabulated file with the extension TAB. The following example files are presented in partial form due to the great amount of data. A sample graphical generation of the Truck Test data is presented in Figure 29.

Nominal Load	Measured Stress
(lbs)	(ksi)
0	0.00
1000	4.00
1500	5.87
1000	4.00
2000	7.64
1000	4.00
1500	5.87
3000	11.40
2500	9.65
1000	4.00
4000	14.90
3500	13.20
2000	7.71
0	0.00

Table 5. Load History for Truck Test. As the specimen was subjected to the loads, stresses were measured by the DL.

Nominal Load	Amplitude	Nominal Stress
(lbs)	(lbs)	(ksi)
0	-	-
1000	1000	3.20
200	-800	-2.56
3000	2800	8.96
500	-2500	-8.00
2000	1500	4.80
700	-1300	-4.16
1700	1000	3.20
800	-900	-2.88
2800	2000	6.40
1200	-1600	-5.12
3300	2100	6.72
200	-3100	-9.92
4000	3800	12.16
1200	-2800	-8.96
1800	600	1.92
1000	-800	-2.56
3700	2700	8.64
400	-3300	-10.56
2000	1600	5.12
400	-1600	-5.12
3100	2700	8.64
1100	-2000	-6.40
4500	3400	10.88
0	-4500	-14.40
2000	2000	6.40
900	-1100	-3.52
2600	1700	5.44
1100	-1500	-4.80
2400	1300	4.16
0	-2400	-7.68

Table 6. Load History for Rainflow Test. As the specimen was subjected to the loads, stresses were measured by the DL. The Nominal Stresses listed in the table are obtained by dividing the load value by the cross-sectional area (P/A).

Truck Test Data Form File ODOTF.TAB

```
11-09-1995
22:53:50
ODOT
ODOTF.TAB




Time        Ch 1      Ch 2      Ch 3      Ch 4      Ch 5      Ch 6      Ch 7      Ch 8
------------------------------------------------------------------------------------------
   3.300                       14.92
   3.600                       14.93
   3.900                       14.91
   4.200                       14.92
   4.500                       14.92
   4.800                       14.92
   5.100                       14.91
   5.400                       14.92
   5.700                       14.91
   6.000                       14.91
   6.300                       14.91
   6.600                       14.90
   6.900                       14.90
   7.200                       14.90
   7.500                       14.90
   7.800                       14.90
   8.100                       14.89
   8.400                       14.89
   8.700                       14.89
   9.000                       14.88
   9.300                       14.88
   9.600                       14.89
   9.900                       14.89
  10.200                       14.88
  10.500                       14.87
  10.800                       14.88
```

Truck Test Data From File ODOTF.DAT

```
ODOT.DAT
ODOTF.DAT
Number of Channels:  1

  Time     No. 3
   0.00   -0.231
   0.30   -0.225
   0.60   -0.242
   0.90   -0.240
   1.20   -0.232
   1.50   -0.238
   1.80   -0.246
   2.10   -0.238
   2.40   -0.245
   2.70   -0.244
   3.00   -0.246
   3.30   -0.252
   3.60   -0.253
   3.90   -0.258
   4.20   -0.253
   4.50   -0.259
   4.80   -0.263
   5.10   -0.264
   5.40   -0.270
   5.70   -0.272
   6.00   -0.274
   6.30   -0.270
   6.60   -0.264
   6.90   -0.279
   7.20   -0.282
   7.50   -0.273
   7.80   -0.285
   8.10   -0.283
   8.40   -0.286
   8.70   -0.293
   9.00   -0.288
   9.30   -0.301
   9.60   -0.303
   9.90   -0.294
  10.20   -0.299
  10.50   -0.299
  10.80   -0.294
  11.10   -0.308
  11.40   -0.304
  11.70   -0.307
  12.00   -0.313
  12.30   -0.316
  12.60   -0.312
  12.90   -0.318
  13.20   -0.317
  13.50   -0.323
  13.80   -0.327
  14.10   -0.322
  14.40   -0.323
```

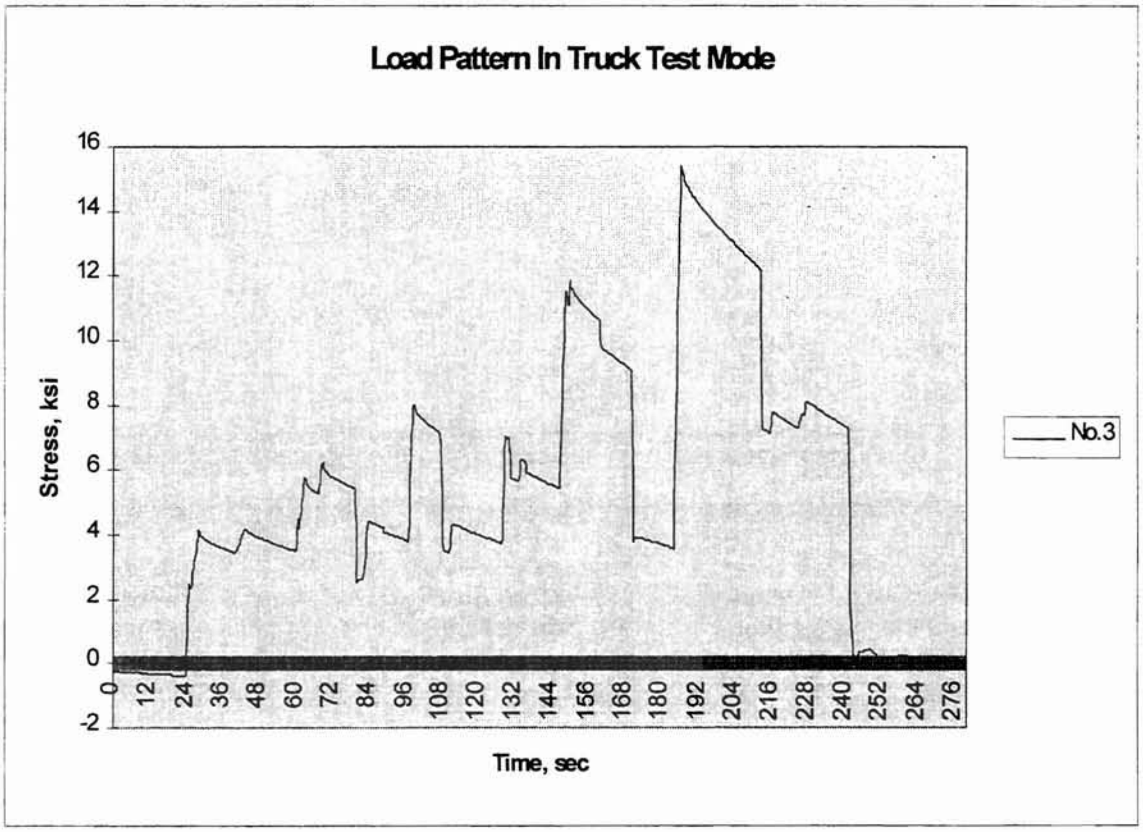


Figure 29. Truck Test Simulation Plot. A steel bar loaded in a testing machine was cycled with an axial load. The above data was collected by the DL to simulate Truck Test output.

Rainflow Data From File ODOTA.TAB

Interval = 1
Channel = 1

BIN NUMBER	COUNTS	STRESS RANGE	PERCENT	DAMAGE FACTOR
1	2	0.50	0.67	0.08
2	0	1.50	0.00	0.00
3	0	2.50	0.00	0.00
4	0	3.50	0.00	0.00
5	1	4.50	0.33	30.07
6	0	5.50	0.00	0.00
7	0	6.50	0.00	0.00
8	0	7.50	0.00	0.00
9	0	8.50	0.00	0.00
10	0	9.50	0.00	0.00
11	0	10.50	0.00	0.00
12	0	11.50	0.00	0.00
13	0	12.50	0.00	0.00
14	0	13.50	0.00	0.00
15	0	14.50	0.00	0.00
16	0	15.50	0.00	0.00
17	0	16.50	0.00	0.00
18	0	17.50	0.00	0.00
19	0	18.50	0.00	0.00
20	0	19.50	0.00	0.00
21	0	20.50	0.00	0.00
22	0	21.50	0.00	0.00
23	0	22.50	0.00	0.00
24	0	23.50	0.00	0.00
25	0	24.50	0.00	0.00

Cycles per Hour (CPH) = 180
Effective Stress Rate = 3.11257

Interval = 2
Channel = 1

BIN NUMBER	COUNTS	STRESS RANGE	PERCENT	DAMAGE FACTOR
1	4	0.50	0.57	0.00
2	0	1.50	0.00	0.00
3	0	2.50	0.00	0.00
4	0	3.50	0.00	0.00
5	1	4.50	0.14	12.75
6	1	5.50	0.14	23.29
7	0	6.50	0.00	0.00
8	0	7.50	0.00	0.00
9	0	8.50	0.00	0.00
10	0	9.50	0.00	0.00
11	0	10.50	0.00	0.00
12	1	11.50	0.14	212.92
13	0	12.50	0.00	0.00
14	0	13.50	0.00	0.00
15	0	14.50	0.00	0.00
16	0	15.50	0.00	0.00
17	0	16.50	0.00	0.00
18	0	17.50	0.00	0.00
19	0	18.50	0.00	0.00
20	0	19.50	0.00	0.00
21	0	20.50	0.00	0.00
22	0	21.50	0.00	0.00
23	0	22.50	0.00	0.00
24	0	23.50	0.00	0.00
25	0	24.50	0.00	0.00

Cycles per Hour (CPH) = 420
Effective Stress Range = 6.29096

Interval = 3
Channel = 1

BIN NUMBER	COUNTS	STRESS RANGE	PERCENT	DAMAGE FACTOR
1	2	0.50	0.40	0.05
2	0	1.50	0.00	0.00
3	0	2.50	0.00	0.00
4	0	3.50	0.00	0.00
5	0	4.50	0.00	0.00
6	0	5.50	0.00	0.00
7	1	6.50	0.20	54.93
8	0	7.50	0.00	0.00
9	0	8.50	0.00	0.00
10	1	9.50	0.20	171.48
11	0	10.50	0.00	0.00
12	0	11.50	0.00	0.00
13	0	12.50	0.00	0.00
14	0	13.50	0.00	0.00
15	1	14.50	0.20	609.73
16	0	15.50	0.00	0.00
17	0	16.50	0.00	0.00
18	0	17.50	0.00	0.00
19	0	18.50	0.00	0.00
20	0	19.50	0.00	0.00
21	0	20.50	0.00	0.00
22	0	21.50	0.00	0.00
23	0	22.50	0.00	0.00
24	0	23.50	0.00	0.00
25	0	24.50	0.00	0.00

Cycles per Hour (CPH) = 300
Effective Stress Range = 9.42104

Interval = 4
Channel = 1

BIN NUMBER	COUNTS	STRESS RANGE	PERCENT	DAMAGE FACTOR
1	1	0.50	0.17	0.02
2	0	1.50	0.00	0.00
3	2	2.50	0.33	5.16
4	0	3.50	0.00	0.00
5	0	4.50	0.00	0.00
6	0	5.50	0.00	0.00
7	0	6.50	0.00	0.00
8	1	7.50	0.17	71.72
9	0	8.50	0.00	0.00
10	1	9.50	0.17	145.75
11	0	10.50	0.00	0.00
12	0	11.50	0.00	0.00
13	0	12.50	0.00	0.00
14	0	13.50	0.00	0.00
15	0	14.50	0.00	0.00
16	1	15.50	0.17	633.06
17	0	16.50	0.00	0.00
18	0	17.50	0.00	0.00
19	0	18.50	0.00	0.00
20	0	19.50	0.00	0.00
21	0	20.50	0.00	0.00
22	0	21.50	0.00	0.00
23	0	22.50	0.00	0.00
24	0	23.50	0.00	0.00
25	0	24.50	0.00	0.00

Cycles per Hour (CPH) = 360
Effective Stress Range = 9.49384

```
Interval = 5
Channel  = 1

--------------------------------------------------------------------------
     BIN                         STRESS                        DAMAGE
   NUMBER        COUNTS          RANGE         PERCENT         FACTOR
--------------------------------------------------------------------------
      1            1              0.50           0.25            0.03
      2            0              1.50           0.00            0.00
      3            0              2.50           0.00            0.00
      4            0              3.50           0.00            0.00
      5            1              4.50           0.25           22.78
      6            1              5.50           0.25           41.59
      7            0              6.50           0.00            0.00
      8            0              7.50           0.00            0.00
      9            0              8.50           0.00            0.00
     10            1              9.50           0.25          214.34
     11            0             10.50           0.00            0.00
     12            0             11.50           0.00            0.00
     13            0             12.50           0.00            0.00
     14            0             13.50           0.00            0.00
     15            0             14.50           0.00            0.00
     16            0             15.50           0.00            0.00
     17            0             16.50           0.00            0.00
     18            0             17.50           0.00            0.00
     19            0             18.50           0.00            0.00
     20            0             19.50           0.00            0.00
     21            0             20.50           0.00            0.00
     22            0             21.50           0.00            0.00
     23            0             22.50           0.00            0.00
     24            0             23.50           0.00            0.00
     25            0             24.50           0.00            0.00
--------------------------------------------------------------------------
Cycles per Hour (CPH) =  240
Effective Stress Range =  6.53238


Summary Output

Interval     Ch#     No. Cycles     Eff.Sr.
--------    -----   ------------   ---------
   1          3          3           3.11
   2          3          7           6.29
   3          3          5           9.42
   4          3          6           9.49
   5          3          4           6.53
                     --------
            Total:      25
```

APPENDIX C

EQUIPMENT SPECIFICATIONS

Datalogger

21 X MICROLOGGER

by Campbell Scientific, Inc.

815 W. 1800 N.
Logan, Utah 84321

4 Excitation Channels

8 Measurement Inputs

8-Character LED Display

16-Button Keypad

INput/Output Serial Port

Memory Capacity:

PROM: 24k

Program Memory: 978k

System Memory: 1022k

Input Storage: 112k (28 locatoions)*

Intermediate Storage: 256k (64 locations)*

Final Storage: 38,592k (19,296 locations)*

*Default allocation on power-up. Total storage of 38,960k may be divided differently [Campbell Scientific, Inc., 1992].

Portable Computer

Gateway 2000 Color Book

486 System

Total Memory: 2,080k

Total Expanded Memory: 3,392k

Hard Drive: 250M

Processing Speed: 66MHz

Power Converter

Power Cord

Soft Carrying Case

APPENDIX D

PROGRAM SOURCE CODE FOR FRAME.BAT

```
@ECHO OFF
REM
REM
REM  PC208 SOFTWARE IS NOT DESIGNED TO WORK SIMULTANEOUSLY WITH
REM  OTHER PROGRAMS SUCH AS QBASIC PROGRAMS.  THE RESULT IS INTERMITTENT
REM  DATA RETRIEVALS--SOMETIMES DATA WILL COLLECT, AND OTHER TIMES
REM  IT WILL NOT.
REM  THIS BATCH FILE RUNS THE QBASIC PROGRAM AND THE PC208 SOFTWARE
REM  TO AVOID THE COLLECTION PROBLEM.  THE QBASIC PROGRAM WILL CREATE
REM  DELETE CERTAIN BATCH FILES.  THESE ARE USED AS CONDITIONAL STATEMENTS
REM  TO DETERMINE IF DATA IS TO BE COLLECTED OR IF THE USER IS ENDING
REM  THE SESSION.
REM
REM
REM  AGAIN IS A LABEL FOR THE PROGRAM LOOP
:AGAIN
CALL PROG.EXE
REM
REM  IF A COLLECTION IS TO BE MADE, THE QBASIC PROGRAM WILL CREATE THE
REM  COLLECT AND RETURN FILES.  IF COLLECT.BAT EXISTS, DATA WILL BE
REM  COLLECTED, IF RETURN.T_F EXISTS, THIS FRAME.BAT WILL LOOP TO :AGAIN.
REM
IF EXIST COLLECT.BAT CALL COLLECT.BAT
DEL COLLECT.BAT
IF EXIST RETURN.T_F GOTO AGAIN
REM
REM  IF THE USER EXITS THE QBASIC PROGRAM TO END THE SESSION, THE
REM  COLLECTION AND RETURN BATCH FILES ARE NOT CREATED.  THE ABSENCE
REM  OF THESE FILES WILL RESULT IN THE TERMINATION OF FRAME.BAT
REM
```

APPENDIX E

PROGRAM SOURCE CODE FOR PROG.EXE

The following program is located in the file PROG.BAS which may be found on the hard drive of the PCS. PROG.EXE is the compiled and executible file of PROG.BAS. To edit the source code, use the QBASIC editer (type QB PROG.BAS).

```
DECLARE SUB BeginTest (Alarm$, Day$, MainTitle$)
DECLARE SUB Collect (Tipe$, Title$)
DECLARE SUB Download (MainTitle$)
DECLARE SUB Rainflow (Channel$(), MainTitle$, Factor#(), Excit%, Delay#,
LowLim#, HighLim#, RnTime#, PvDist$, AmpBins%, Rate#, Hue%(), Ch%, Scrap%)
DECLARE SUB RainPrint (MainTitle$, Factor#(), LowLim#, HighLim#, RnTime#,
PvDist$, Rate#, Excit%, AmpBins%)
DECLARE SUB ReadInit (Factor#(), INIT#())
DECLARE SUB ReadRain (MainTitle$, Runner1$, Ch%, HighLim#, LowLim#, RnTime#,
AmpBins%)
DECLARE SUB ReadTruck (MainTitle$, Runner1$, Runner2$, Counter#, Factor#(),
INIT#(), Ch%)
DECLARE SUB ResetChar (Count%)
DECLARE SUB Rturn (Null1$)
DECLARE SUB SendInit (Delay#)
DECLARE SUB SendTime (TimeSet%)
DECLARE SUB SendWarm (Rate#, MainTitle$, Factor#(), Scrap%, Channel$(), Hue%(),
Ch%, Delay#)
DECLARE SUB SetAlarm (Alarm$, Day$)
DECLARE SUB Setup (Channel$(), MainTitle$, Factor#(), Excit%, Delay#, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Rate#, Hue%(), Ch%, Scrap%)
DECLARE SUB TruckPrint (Rate#, MainTitle$, Factor#(), INIT#(), Excit%)
DECLARE SUB TruckTest (Rate#, MainTitle$, Factor#(), Scrap%, Channel$(), Hue%(),
Ch%, Delay#)
DECLARE SUB VarrayRead (TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)
DECLARE SUB VarraySave (TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)
DECLARE SUB WarmUp (Null1$)

'************************************************
'*      PROGRAM SECTION ALPHA-1                 *
'*      THIS SECTION ALLOWS THE USER TO SELECT  *
'*      THE TYPE OF TEST THAT IS TO BE RUN.     *
'************************************************
DIM INIT#(8), Factor#(8), Channel$(8), Ch%(8), Hue%(8)
```

```
CALL Rturn(Nulll$)
CALL VarrayRead(TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)

'Scrap = 0
'Delay# = 0
'MainTitle$ = ""
Excit% = 5


IF Status$ = "New" THEN
   CLS

   '*************************************************************
   '*      THIS MAKES A TITLE BOX                               *
   '*************************************************************
   LOCATE 3, 12: PRINT CHR$(218); STRING$(53, 196); CHR$(191);
   LOCATE 13, 12: PRINT CHR$(192); STRING$(53, 196); CHR$(217);
      FOR Row = 4 TO 12
         LOCATE Row, 12: PRINT CHR$(179);
         LOCATE Row, 66: PRINT CHR$(179);
      NEXT Row

   LOCATE 5, 21
   COLOR 4: PRINT "OKLAHOMA DEPARTMENT OF TRANSPORTATION"
   LOCATE 8, 18
   PRINT "MAGIC FATIGUE TESTING MACHINE AND SOFTWARE"
   LOCATE 11, 16
   COLOR 5: PRINT "By Oklahoma State University Civil Engineering"
   LOCATE 20, 30
   PRINT "Hit Return To Begin"
   P$ = INPUT$(1)


   '**************************************************************
   '*                                                            *
   '*  This is a sieve for the collection type that has just been  *
   '*  obtained from the batch file. The 'Tipe$' variable will     *
   '*  dictate which collection was made: Init, Truck, or Rain.    *
   '*  If Init collection was made, program will automatically     *
   '*  process the data and begin the required test.  If Truck or *
   '*  Rain collection was made, data processing will begin.       *
   '*                                                            *
   '**************************************************************

ELSEIF Status$ = "Collect" AND Tipe$ = "Init" THEN

   CALL ReadInit(Factor#(), INIT#())
   CLS
   LOCATE 15, 20: PRINT "Initialization Complete.  Truck Test Ready."
   LOCATE 16, 22: PRINT "(Perss Any Key To Contiue To Main Menu)"
   Q$ = INPUT$(1)
   MainTitle$ = OtherTitle$
     CALL VarraySave(TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)
```

```
ELSEIF Status$ = "Collect" AND Tipe$ = "Truck" THEN

   Runner1$ = MainTitle$ + CHR$(65 + Count%) + ".TAB"
   Runner2$ = MainTitle$ + CHR$(65 + Count%) + ".DAT"
   Count% = Count% + 1
   IF Count% = 26 THEN Count% = 0
    CALL ReadTruck(MainTitle$, Runner1$, Runner2$, Counter#, Factor#(), INIT#(),
Ch%)
   CLS
   LOCATE 10, 15: PRINT "Your Data Files Are Titled:"
   LOCATE 12, 20: PRINT Runner1$
   LOCATE 13, 20: PRINT Runner2$
   LOCATE 15, 10: PRINT "Please Make Note Of These Assignements!"
   LOCATE 16, 10: PRINT "(Press ENTER To Continue)"
   DO WHILE Q$ <> CHR$(13)
   Q$ = INPUT$(1)
   LOOP


ELSEIF Status$ = "Collect" AND Tipe$ = "Rain" THEN

   Runner1$ = MainTitle$ + CHR$(65 + Count%) + ".TAB"
   Runner1$ = MainTitle$ + CHR$(65 + Count%) + ".DAT"
   Count% = Count% + 1
   IF Count% = 26 THEN Count% = 0
      CALL  ReadRain(MainTitle$,  Runner1$,  Ch%,  HighLim#,  LowLim#,  RnTime#,
AmpBins%)
   CLS
   LOCATE 10, 15: PRINT "Your Data Files Are Titled:"
   LOCATE 12, 20: PRINT Runner1$
   LOCATE 13, 20: PRINT Runner2$
   LOCATE 15, 10: PRINT "Please Make Note Of These Assignements!"
   LOCATE 16, 10: PRINT "(Press ENTER To Continue)"
   DO WHILE Q$ <> CHR$(13)
   Q$ = INPUT$(1)
   LOOP

END IF

WHILE Q$ <> "Q"
CLS

'***************************************************************
'*      THIS MAKES A LIST OF OPTIONS                           *
'***************************************************************

LOCATE 10, 25
COLOR 2: PRINT "Select An Option"
LOCATE 14, 10: PRINT "(A)   Setup"
LOCATE 15, 10: PRINT "(B)   Send Time"
LOCATE 16, 10: PRINT "(C)   Warm-up/Standby"
LOCATE 17, 10: PRINT "(D)   Rainflow Test"
LOCATE 18, 10: PRINT "(E)   Initialize"
LOCATE 19, 10: PRINT "(F)   Truck Test"
LOCATE 20, 10: PRINT "(G)   Reset Character Counter"
LOCATE 21, 10: PRINT "(H)   Collect Test Data"
LOCATE 22, 10: PRINT "(I)   Monitor Input"
LOCATE 24, 10: PRINT "(Q)   Quit"
COLOR 14
LOCATE 14, 11: PRINT "A"
LOCATE 15, 11: PRINT "B"
LOCATE 16, 11: PRINT "C"
LOCATE 17, 11: PRINT "D"
LOCATE 18, 11: PRINT "E"
LOCATE 19, 11: PRINT "F"
LOCATE 20, 11: PRINT "G"
LOCATE 21, 11: PRINT "H"
LOCATE 22, 11: PRINT "H"
LOCATE 24, 11: PRINT "Q"

DEFINT A-Z
   Q$ = " "
   WHILE INSTR("ABCDEFGHQ" + CHR$(27), Q$) = 0
       Q$ = UCASE$(INPUT$(1))
   WEND

'****************************************************************
'*      OPTION A ALLOWS USER TO INPUT TEST PARAMETERS.  THIS    *
'*      MODULE WILL PERFORM SETUP FOR BOTH RAINFLOW AND TRUCK   *
'*      TESTS.                                                  *
'****************************************************************

  IF Q$ = "A" THEN
      CALL Setup(Channel$(), MainTitle$, Factor#(), Excit%, Delay#, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Rate#, Hue%(), Ch%, Scrap%)
'    CALL BeginTest(Alarm$, Day$, MainTitle$)

'****************************************************************
'*      OPTION B SENDS COMPUTER TIME TO DATA LOGGER             *
'*      TO SYCRONIZE COMOPUTER AND DATA LOGGER                  *
'*         USER IS GIVEN OPTION TO ADJSUT COMPUTER TIME IF      *
'*         NECESSARY                                            *
'****************************************************************

  ELSEIF Q$ = "B" THEN

    CALL SendTime(TimeSet%)

'****************************************************************
'*      OPTION D INTITIATES A RAINFLOW TEST                     *
'*      RAINFLOW TEST WILL COLLECT MANY DATA POINTS OVER A LONG *
'*      PERIOD OF TIME (1 TO 14 DAYS) WHILE AMBIENT TRAFFIC     *
'*      CROSSES BRIDGE.                                         *
'****************************************************************

  ELSEIF Q$ = "C" THEN

   CALL SendWarm(Rate#, MainTitle$, Factor#(), Scrap%, Channel$(), Hue%(), Ch%,
Delay#)
  CALL WarmUp(Nulll$)

  ELSEIF Q$ = "D" THEN

    Tipe$ = "Rain"
    Scrap% = 0
     CALL Rainflow(Channel$(), MainTitle$, Factor#(), Excit%, Delay#, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Rate#, Hue%(), Ch%, Scrap%)

    IF Scrap% = 0 THEN
      IF TimeSet% <> 1 THEN
```

```
       CALL SendTime(TimeSet%)
      END IF

      CALL RainPrint(MainTitle$, Factor#(), LowLim#, HighLim#, RnTime#, PvDist$,
Rate#, Excit%, AmpBins%)
      CALL Download(MainTitle$)
    END IF

'***************************************************************
'*      OPTION E INTITIALIZES THE INPUTS FOR A TRUCK TEST.  THE *
'*      SET UP MUST BE RUN BEFORE INITIALIZING.                 *
'***************************************************************

  ELSEIF Q$ = "E" THEN

    FOR I = 1 TO 8
      INIT#(I) = 0
    NEXT I

    SHELL "DEL INIT.DAT"

    Tipe$ = "Init"
    OtherTitle$ = MainTitle$
    MainTitle$ = "INIT"
    CALL TruckPrint(Rate#, MainTitle$, Factor#(), INIT#(), Excit%)
    CALL Download(MainTitle$)
    CALL SendInit(Delay#)
    Title$ = MainTitle$ + ".DAT"
    CALL Collect(Tipe$, Title$)
    Status$ = "Collect"
      CALL VarraySave(TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#,  RnTime#,  PvDist$,  AmpBins%,  Status$,  Channel$(),  Ch%,  Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)
    END

'***************************************************************
'*      OPTION F INTITIATES A TRUCK TEST                        *
'*      TRUCK TEST WILL COLLECT MANY DATA POINTS OVER A SHORT   *
'*      PERIOD OF TIME (1 TO 5 MIN) WHILE A TRUCK IS DRIVEN     *
'*      OVER THE BRIDGE                                         *
'***************************************************************

  ELSEIF Q$ = "F" THEN


    Scrap% = 0
      CALL TruckTest(Rate#, MainTitle$, Factor#(), Scrap%, Channel$(), Hue%(),
Ch%, Delay#)

IF Scrap% = 0 THEN


    CALL TruckPrint(Rate#, MainTitle$, Factor#(), INIT#(), Excit%)
    CALL Download(MainTitle$)
    Counter# = Delay# / Rate#
    Title$ = MainTitle$ + ".DAT"
    Tipe$ = "Truck"
'    SHELL "GT OSU M"
'    PRINT "GT OSU M"
'    Q$ = INPUT$(1)
```

```
END IF

  ELSEIF Q$ = "G" THEN

    CALL ResetChar(Count%)

    Count% = 0 '***RESTS COUNTING PARAMETER FOR COLLECTION FILES.

'*****************************************************************
'*      OPTION H COLLECTS THE DATA FOR THE APPROPRIATE TEST.     *
'*      COLLECTION IS MADE BY USER INPUT, BUT MODIFICATIONS      *
'*      COULD EASILY BE MADE TO EXECUTE AN AUTOMATIC TIMED       *
'*      COLLECTION.  THE PROGRAM DETERMINES WHICH TYPE OF COL-    *
'*      LECTION IS TO BE MADE WITHOUT USER PROMPT.               *
'*****************************************************************


  ELSEIF Q$ = "H" THEN

    Title$ = MainTitle$ + ".DAT"
    SHELL "DEL " + Title$
    CALL Collect(Tipe$, Title$)
    Status$ = "Collect"
      CALL VarraySave(TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)
    END

  ELSEIF Q$ = "I" THEN

    SHELL "GT OSU M" '*** COMMAND FOR MONITOR INPUT IN GT.EXE

  ELSEIF Q$ = CHR$(27) THEN

    Q$ = "Q"

  END IF

WEND

IF Status$ = "Collect" THEN
  SHELL "DEL COLLECT.BAT"
'  PRINT "DEL COLLECT.BAT"
'  Q$ = INPUT$(1)

END IF

TimeSet% = 0
SHELL "DEL RETURN.T_F"
'PRINT "DEL RETURN.T_F"
'Q$ = INPUT$(1)
Status$ = "New"
CALL VarraySave(TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)

SHELL "COPY VARS.SET VARS.BAK"

CLS
```

```
END

DEFSNG A-Z
SUB BeginTest (Alarm$, Day$, MainTitle$)
'****************************************************************
'*      This sub will begin a test at a specific time.  The     *
'*      starting time is set with the SetAlarm Sub.             *
'*      The Download Sub will begin a test immediately.         *
'****************************************************************
CLS

'****************************************************************
'*      If no alarm time has been set, the test will begin      *
'*      immediately.                                            *
'*      If alarm time is the actual time (which is highly       *
'*      unlikely), the test will begin immediately.             *
'*      If the time is not the same as the alarm time, the      *
'*      display clock will countdown the time until the test    *
'*      begins.                                                 *
'****************************************************************

IF Alarm$ = "" THEN
  continue = 1
ELSEIF TIME$ <> Alarm$ THEN
  continue = 0
ELSE
  continue = 2

END IF

DO WHILE continue = 0

  LOCATE 10, 25: COLOR 14: PRINT "Test Will Begin Momentarily"
  LOCATE 12, 10: COLOR 4: PRINT "Current Time"
  LOCATE 12, 50: COLOR 13: PRINT "Alarm Time"
  LOCATE 13, 12: COLOR 15: PRINT TIME$
  LOCATE 13, 51: PRINT Alarm$

  IF TIME$ = Alarm$ THEN
    continue = 2
    Alarm$ = ""
  END IF

LOOP

IF Alarm$ = "" AND continue = 1 THEN
DO WHILE continue = 1

  CLS
   LOCATE 10, 25: COLOR 12: PRINT "No Alarm Has Been Set.  Do You Wish To Set
It?"
  LOCATE 12, 25: PRINT "(Please Type 'Y' To Set The Alarm Or 'N' To Begin Test
Immediately)."
  INPUT Cont$
  IF UCASE$(Cont$) = "Y" THEN
'     CALL SetAlarm(Alarm$, Day$, MainTitle$)
    continue = 3
  ELSEIF Cont$ = "N" OR Cont$ = "n" THEN
    continue = 2
  END IF
```

```
LOOP
END IF

IF Cont = 2 THEN

  'SendLine$ = "c:\etc\logger\gt osu k y d " + MainTitle$
  SendLine$ = "GT OSU K Y D " + MainTitle$

  PRINT SendLine$
END IF
IF Cont = 3 THEN
  Q$ = "E"
END IF

  COLOR 5: LOCATE 23, 23: PRINT "Hit Return To Continue ....."
  Q$ = INPUT$(1)

Alarm$ = ""
END SUB

DEFINT A-Z
SUB Collect (Tipe$, Title$)

OPEN "COLLECT.BAT" FOR OUTPUT AS #1
PRINT #1, "TELCOM OSU/C/F " + Title$
CLOSE #1

END SUB

DEFSNG A-Z
SUB Download (MainTitle$)
'*****************************************************************
'*      This sub will send the download file to the Datalogger  *
'*****************************************************************
CLS

Phrase$ = "GT OSU D " + MainTitle$ + " Q"
SHELL Phrase$

END SUB

DEFINT A-Z
SUB  Rainflow  (Channel$(),  MainTitle$,  Factor#(),  Excit%,  Delay#,  LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Rate#, Hue%(), Ch%, Scrap%)

I = 3
LOCATE I, 10: PRINT "Main Title: "; MainTitle$
LOCATE I + 1, 10: PRINT "Interval Rate: "; Rate#
COLOR 2: LOCATE 5, 10: PRINT "Low Limit Stress:  "; LowLim#
COLOR 2: LOCATE 6, 10: PRINT "High Limit Stress: "; HighLim#
COLOR 2: LOCATE 7, 10: PRINT "Number of Bins: "; AmpBins%
COLOR 2: LOCATE 8, 10: PRINT "Rainflow Interval: "; RnTime#
COLOR 2: LOCATE 9, 10: PRINT "Minimum Amplitude:  "; PvDist$

COLOR 1
LOCATE 13, 14: PRINT CHR$(218); STRING$(51, 196); CHR$(191);
LOCATE 21, 14: PRINT CHR$(192); STRING$(51, 196); CHR$(217);
   FOR Row = 14 TO 20
      LOCATE Row, 14: PRINT CHR$(179)
      LOCATE Row, 27: PRINT CHR$(179)
      LOCATE Row, 40: PRINT CHR$(179)
      LOCATE Row, 53: PRINT CHR$(179)
      LOCATE Row, 66: PRINT CHR$(179)
   NEXT Row
   LOCATE 17, 14: PRINT STRING$(53, 196)
      LOCATE 13, 27: PRINT CHR$(194)
      LOCATE 13, 40: PRINT CHR$(194)
      LOCATE 13, 53: PRINT CHR$(194)
      LOCATE 17, 14: PRINT CHR$(195)
      LOCATE 17, 27: PRINT CHR$(197)
      LOCATE 17, 40: PRINT CHR$(197)
      LOCATE 17, 53: PRINT CHR$(197)
      LOCATE 17, 66: PRINT CHR$(180)
      LOCATE 21, 27: PRINT CHR$(193)
      LOCATE 21, 40: PRINT CHR$(193)
      LOCATE 21, 53: PRINT CHR$(193)

FOR I = 1 TO 4
   COLOR 8: LOCATE 18, (2 + 13 * I): PRINT I
   COLOR 8: LOCATE 14, (2 + 13 * I): PRINT I + 4
NEXT I

FOR I = 1 TO 4
   COLOR Hue%(I)
   LOCATE 19, (5 + I * 13): PRINT Channel$(I)
   COLOR Hue%(I + 4)
   LOCATE 15, (5 + I * 13): PRINT Channel$(I + 4)
   COLOR 14

   IF Factor#(I) <> 0 THEN
      LOCATE 20, 3 + 13 * I: PRINT "GF ="; Factor#(I)
   END IF

   IF Factor#(I + 4) <> 0 THEN
     LOCATE 16, 3 + 13 * I: PRINT "GF ="; Factor#(I + 4)
   END IF
NEXT I
LOCATE 23, 18: COLOR 15: PRINT "Is This Information Correct ('Y' To Proceed)"
LOCATE 24, 18: PRINT "  ('Esc' To Abort -- Redo Input In Set Up)"
WHILE INSTR(CHR$(13) + CHR$(27) + "Y", Conf$) = 0
   Conf$ = UCASE$(INPUT$(1))
WEND

IF Conf$ = "Y" OR Conf$ = CHR$(13) THEN
   Scrap% = 0
ELSEIF Conf$ = CHR$(27) THEN
   Scrap% = 1
END IF

Ch% = 0
FOR n = 1 TO 8
   IF Factor#(n) <> 0 THEN Ch% = Ch% + 1
NEXT n

END SUB

SUB RainPrint (MainTitle$, Factor#(), LowLim#, HighLim#, RnTime#, PvDist$,
Rate#, Excit%, AmpBins%)

Dumb = 0
```

```
Title$ = MainTitle$ + ".DLD"

      OPEN Title$ FOR OUTPUT AS #1
      PRINT #1, "};21X"
      PRINT #1, ";"; Title$
      PRINT #1,
      PRINT #1, "MODE 1"
      PRINT #1, "SCAN RATE"; Rate#
      PRINT #1,


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

K = 0
I = 0
Count = 1
FOR n = 1 TO 8
  IF Factor#(n) <> 0 THEN
  I = I + 1
     PRINT #1, Count; ":P6"
  Count = Count + 1
     PRINT #1, "1:1"
     PRINT #1, "2:1"
     PRINT #1, "3:"; CHR$(n + 48)
     PRINT #1, "4:"; CHR$(n \ 5 + (n MOD 5) + 48)
     PRINT #1, "5:"; Excit% * 1000
     PRINT #1, "6:"; CHR$(I + 48)
     PRINT #1, "7:"; 29 * (4 / Factor#(n))
     PRINT #1, "8:0"
     PRINT #1,
  END IF

NEXT n

FOR n = 1 TO I
     PRINT #1, Count; ":P92"
  Count = Count + 1
     PRINT #1, "1:0"
     PRINT #1, "2:"; RnTime#
     PRINT #1, "3:10"
     PRINT #1,
'     PRINT #1, Count; ":P77"
'  Count = Count + 1
'     PRINT #1, "1:0111"
'     PRINT #1,

     PRINT #1, Count; ":P81"
  Count = Count + 1
     PRINT #1, "1:1"
     PRINT #1, "2:"; n
     PRINT #1, "3:1"
     PRINT #1, "4:2"
     PRINT #1, "5:"; AmpBins%
     PRINT #1, "6:"; LowLim#
     PRINT #1, "7:"; HighLim#
     PRINT #1, "8:"; PvDist$
     PRINT #1, "9:11"
     PRINT #1, "10:0"
     PRINT #1,
NEXT n

     PRINT #1, "MODE 2"
     PRINT #1, "SCAN RATE 0.0000"
     PRINT #1,
     PRINT #1, "MODE 3"
     PRINT #1,
     PRINT #1, "MODE 4"
     PRINT #1, "1:00"
     PRINT #1, "2:00"
     PRINT #1,
     PRINT #1, "MODE 10"
     PRINT #1, "1:28"
'     PRINT #1, "2:64"
     PRINT #1, "2:2434"
     PRINT #1,
     PRINT #1, "MODE 12"
     PRINT #1, "1:00"
     PRINT #1, "2:0000"
     PRINT #1, "  ÿ"

CLOSE #1

END SUB

SUB ReadInit (Factor#(), INIT#())
DIM Dumb#(8)

Title$ = "INIT.DAT"
OPEN Title$ FOR INPUT AS #1
Sum = 0
INPUT #1, Dumb#

DO

Sum = Sum + 1
INPUT #1, Dumb#
INPUT #1, Dumb#

FOR I = 1 TO 8
  IF Factor#(I) <> 0 THEN
    INPUT #1, Dumb#
    INIT#(I) = INIT#(I) + Dumb#
'    PRINT Sum, Dumb#, INIT#(I)
'    Q$ = INPUT$(1)
  END IF
NEXT I

LOOP UNTIL EOF(1)

FOR I = 1 TO 8
  INIT#(I) = (-1) * INIT#(I) / Sum
'  PRINT INIT#(I),
NEXT I
```

```
'Q$ = INPUT$(1)

CLOSE #1
END SUB

SUB ReadRain (MainTitle$, Runner1$, Ch%, HighLim#, LowLim#, RnTime#, AmpBins%)

DIM Sum%(20, 8), DamSum#(20, 8), CPH#(20, 8), SREff#(20, 8), Pnt%(8, 50)
Intrvl% = 0

Title$ = MainTitle$ + ".DAT"
OPEN Title$ FOR INPUT AS #1
OPEN Runner1$ FOR OUTPUT AS #2
INPUT #1, Dummy$

DO UNTIL EOF(1)
  Intrvl% = Intrvl% + 1

  FOR I = 1 TO Ch%
    INPUT #1, Dumb%
'     INPUT #1, Day%
'     INPUT #1, TymeHM%
'     INPUT #1, TymeSS%
    FOR J = 1 TO 2
      FOR K = 1 TO AmpBins%
        INPUT #1, Dumb%
        Pnt%(I, K) = Pnt%(I, K) + Dumb%
        IF J = 2 THEN Sum%(Intrvl%, I) = Sum%(Intrvl%, I) + Pnt%(I, K)
      NEXT K
    NEXT J
  NEXT I

FOR J = 1 TO Ch%
PRINT #2, "Interval ="; Intrvl%
PRINT #2, "Channel  ="; J
PRINT                                                                  #2,
"_________________________________________________________________________
_"
PRINT  #2,  "             BIN                                       STRESS
DAMAGE"
PRINT  #2,  "        NUMBER          COUNTS          RANGE          PERCENT
FACTOR"
PRINT                                                                  #2,
"_________________________________________________________________________
_"

FOR K = 1 TO AmpBins%
  PRINT #2, USING "#########"; K;
  PRINT #2, USING "###############  "; Pnt%(J, K);
  SR# = LowLim# + K * (HighLim# - LowLim#) / AmpBins%
  IF Sum%(Intrvl%, J) = 0 THEN
    Pct# = 0
    DamFac# = 0
  ELSE
    Pct# = Pnt%(J, K) / Sum%(Intrvl%, J)
    DamFac# = Pct# * SR# ^ 3
  END IF
  DamSum#(Intrvl%, J) = DamSum#(Intrvl%, J) + DamFac#
  PRINT #2, USING "#########.##  "; SR#; Pct#; DamFac#
  Pnt%(J, K) = 0
```

```
NEXT K

SREff#(Intrvl%, J) = DamSum#(Intrvl%, J) ^ (1 / 3)
CPH#(Intrvl%, J) = Sum%(Intrvl%, J) / (RnTime# / 60)
PRINT                                                                  #2,
"
________________________________________________________________________________
 "
_
PRINT #2, "Cycles per Hour (CPH) = "; CPH#(Intrvl%, J)
PRINT #2, "Effective Stress Rate = "; SREff#(Intrvl%, J)
PRINT #2,
PRINT #2,
NEXT J

LOOP
CLOSE #1
CLOSE #2
END SUB

SUB  ReadTruck  (MainTitle$,  Runner1$,  Runner2$,  Counter#,  Factor#(),  INIT#(),
Ch%)
DIM Dumb#(8)

Title$ = MainTitle$ + ".DAT"
   OPEN Title$ FOR INPUT AS #1
   OPEN Runner1$ FOR OUTPUT AS #2

   PRINT #2, DATE$
   PRINT #2, TIME$
   PRINT #2, MainTitle$
   PRINT #2, Runner1$
   PRINT #2, Comment1$
   PRINT #2, Comment2$
   PRINT #2, Comment3$
   PRINT #2,
   PRINT #2,
   PRINT #2,
    PRINT #2, "Time        Ch 1       Ch 2       Ch 3       Ch 4       Ch 5       Ch 6
Ch 7      Ch 8   "
   PRINT #2, "----------------------------------------------------------------
--------------"
            ' 1523:10.3  223.44

   '** Reads the blank line at the top of the file.

   INPUT #1, Dummy


DO UNTIL EOF(1)
   INPUT #1, Dummy
'   INPUT #1, Day#
'   INPUT #1, Min#
   INPUT #1, Sec#

   PRINT #2, USING "####.###"; Sec#;
   FOR J = 1 TO 8
    IF Factor#(J) <> 0 THEN
     INPUT #1, Pt#
     Pt# = Pt# - INIT#(J)
     IF J = 8 THEN
       PRINT #2, USING " ###.##  "; Pt#
     ELSE
       PRINT #2, USING " ###.##"; Pt#;
     END IF
    END IF
    IF Factor#(J) = 0 THEN
     IF J = 8 THEN
      PRINT #2,
     ELSE
      PRINT #2, "         ";
     END IF
    END IF
   NEXT J
'PRINT Dummy; Day#; Hour#; Min#; Sec#; Pt#
'Q$ = INPUT$(1)

LOOP
CLOSE #1
CLOSE #2

OPEN Title$ FOR INPUT AS #1
OPEN Runner2$ FOR OUTPUT AS #2

PRINT #2, Title$
PRINT #2, Runner2$
PRINT #2, "Number of Channels: "; Ch%
PRINT #2,
PRINT #2, "  Time ";

FOR I = 1 TO 8
  IF Factor#(I) <> 0 THEN
    PRINT #2, "    No. "; CHR$(48 + I);
  END IF
NEXT I
PRINT #2,


INPUT #1, Dumb#
INPUT #1, Dumb#
INPUT #1, Dumb#

Sec# = 0

PRINT #2, USING "####.##  "; Sec#;

FOR I = 1 TO Ch%
  INPUT #1, Stress#
  PRINT #2, USING "###.###  "; Stress#;
NEXT I
PRINT #2,

INPUT #1, Dumb2#
INPUT #1, Dumb2#
Sec# = Dumb2# - Dumb#
PRINT #2, USING "####.##  "; Sec#;

FOR I = 1 TO Ch%
  INPUT #1, Stress#
  PRINT #2, USING "###.###  "; Stress#;
NEXT I
PRINT #2,
J = 1

DO UNTIL EOF(1)
  J = J + 1
  INPUT #1, Dumb#
  INPUT #1, Dumb#
  PRINT #2, USING "####.##  "; Sec# * J;

  FOR I = 1 TO Ch%
    INPUT #1, Stress#
    PRINT #2, USING "###.###  "; Stress#;
  NEXT I
  PRINT #2,
LOOP

CLOSE #1
CLOSE #2

END SUB

SUB ResetChar (Count%)
CLS
continue = 0

DO WHILE continue = 0

LOCATE 15, 5: PRINT "With What Character Would You Like To Begin The Next
Collection?"
LOCATE 17, 15: INPUT "Character Counter = "; Dumb$

FOR I = 0 TO 25

IF UCASE$(Dumb$) = CHR$(65 + I) THEN
  Count% = I
  continue = 1
END IF

NEXT I

LOOP

END SUB

SUB Rturn (Nulll$)
OPEN "RETURN.T_F" FOR OUTPUT AS #1
PRINT #1, "This is a dummy file."
CLOSE #1
END SUB

SUB SendInit (Delay#)
'****************************************************************
'*      This sub sends the Initializing program to the data-    *
'*      logger and starts a delay timer while channels are      *
'*      initializing.                                           *
'****************************************************************

TurnOn# = TIMER
CLS
```

```
Q$ = ""

DO
Esc$ = INKEY$
COLOR 8
LOCATE 12, 15: PRINT "Channels Are Currently Being"; Initialized; ""
COLOR 14: LOCATE 15, 15: PRINT "Timer:    ";
PRINT USING "###.#"; TIMER - TurnOn#
LOCATE 17, 15: PRINT "End Time: "; Delay#
LOCATE 20, 17: COLOR 4: PRINT "(Hit Esc Key To Cancel)"
LOOP UNTIL Esc$ = CHR$(27) OR TIMER >= TurnOn# + Delay#

END SUB

DEFSNG A-Z
SUB SendTime (TimeSet%)
'*****************************************************************
'*      This sub syncornizes the datalogger time with the        *
'*      computer's time.                                         *
'*****************************************************************
CLS
LOCATE 9, 23: PRINT "The Internal Chronometer Is Set To:"
LOCATE 10, 23: PRINT "------------------------------------"
LOCATE 11, 27: PRINT TIME$, DATE$
LOCATE 15, 23: PRINT "Would You Like To Adjust The Computer Time"
LOCATE 16, 23: PRINT "Before Initializing The Datalogger Clock?"
LOCATE 18, 20: COLOR 15: PRINT "('T' To Adjust Clock, Any Other Key To Proceed)"
Q$ = INPUT$(1)
IF UCASE$(Q$) = "T" THEN
DO
  LOCATE 19, 23: PRINT "Enter The Correct Time"
  LOCATE 20, 30: INPUT Timer$
  LOCATE 21, 23: PRINT "Enter The Correct Date"
  LOCATE 22, 30: INPUT Dater$
  IF Timer$ <> "" THEN
    TIME$ = Timer$
  END IF
  IF Dater$ <> "" THEN
    DATE$ = Dater$
  END IF

  LOCATE 23, 23: PRINT "Press 'Y' To Continue or Any Other Key To Correct"
  T$ = INPUT$(1)
LOOP WHILE UCASE$(T$) <> "Y"

END IF
CLS
LOCATE 18, 23: COLOR 4: PRINT "Time Is Transmitting"

SHELL "GT OSU K Y Q"
'PRINT "GT OSU K Y Q"
'Q$ = INPUT$(1)

TimeSet% = 1
END SUB

DEFINT A-Z
SUB SendWarm (Rate#, MainTitle$, Factor#(), Scrap%, Channel$(), Hue%(), Ch%,
Delay#)
```

```
      OPEN "WARMER.DLD" FOR OUTPUT AS #1
      PRINT #1, "};21X"
      PRINT #1, ";WARMER.DLD"
      PRINT #1,
      PRINT #1, "MODE 1"
      PRINT #1, "SCAN RATE"; Rate#
      PRINT #1,


'****************************************************************
'*      QBasic will not print "1:P6" to a file correctly.  The  *
'*      Output appears as '1 : P6'.  It is for this reason that *
'*      the code for this sequence seems odd.                   *
'*      The n\ 5 + (n MOD 5) is a clever way of selecting the   *
'*      appropriate excitation channel for the corresponding    *
'*      input channe.                                           *
'****************************************************************

I = 0
FOR n = 1 TO 8
  IF Factor#(n) <> 0 THEN
  I = I + 1

      PRINT #1, CHR$(I + 48); ":P6" '** Command No. 6
      PRINT #1, "1:1"  '** Repetition
      PRINT #1, "2:11"  '** Resolution
      PRINT #1, "3:"; CHR$(n + 48)'** Input channel
      PRINT #1, "4:"; CHR$(n \ 5 + (n MOD 5) + 48)' "** Excitation channel"
      PRINT #1, "5:"; Excit% * 1000 '** Excitation voltage
      PRINT #1, "6:"; CHR$(I + 48)'** Storage location
      PRINT #1, "7:"; 29 * (4 / Factor#(n))'** Multiplicity factor
      PRINT #1, "8:0";
      PRINT #1,

END IF
NEXT n

      PRINT #1, "MODE 2"
      PRINT #1, "SCAN RATE 0.0000"
      PRINT #1,
      PRINT #1, "MODE 3"
      PRINT #1,
      PRINT #1, "MODE 4"
      PRINT #1, "1:00"
      PRINT #1, "2:00"
      PRINT #1,
      PRINT #1, "MODE 10"
      PRINT #1, "1:28"
      PRINT #1, "2:64"
      PRINT #1,
      PRINT #1, "MODE 12"
      PRINT #1, "1:00"
      PRINT #1, "2:0000"
      PRINT #1, "  ÿ"

CLOSE #1

Phrase$ = "GT OSU D WARMER Q"
SHELL Phrase$
```

```
END SUB

DEFSNG A-Z
SUB SetAlarm (Alarm$, Day$)
'****************************************************************
'*      This sub allows the user to set an alarm so that a      *
'*      test may begin automaticall at certain time.            *
'****************************************************************

CLS

COLOR 1
LOCATE 3, 12: PRINT CHR$(218); STRING$(53, 196); CHR$(191);
LOCATE 14, 12: PRINT CHR$(192); STRING$(53, 196); CHR$(217);
   FOR Row = 4 TO 13
      LOCATE Row, 12: PRINT CHR$(179);
      LOCATE Row, 66: PRINT CHR$(179);
   NEXT Row

COLOR 15: LOCATE 4, 26: PRINT "The Current Time Is: "; TIME$
COLOR 4: LOCATE 6, 16: PRINT "Please Enter The Time That You Would"
LOCATE 7, 16: PRINT "Like The Next Test To Start."
COLOR 2: LOCATE 17, 20: INPUT "Test Alarm Time   "; Alarm$
COLOR 15: LOCATE 9, 26: PRINT "The Current Date Is: "; DATE$
COLOR 4: LOCATE 11, 16: PRINT "Please Enter The Date That You Would"
LOCATE 12, 16: PRINT "Like This Test To Begin."
COLOR 2: LOCATE 19, 39: PRINT DATE$
LOCATE 19, 20: INPUT "Alarm Date Set    "; Day$
IF Day$ = "" THEN
  Day$ = DATE$
END IF
'Delta$ = Alarm$ - TIME$
LOCATE 21, 13: COLOR 3: PRINT "Datalogger Will Take Initial Readings At "
LOCATE 21, 54: COLOR 15: PRINT Alarm$
COLOR 5: LOCATE 23, 23: PRINT "Hit Return To Continue ....."
Q$ = INPUT$(1)
continue = 0

END SUB

DEFINT A-Z
SUB Setup (Channel$(), MainTitle$, Factor#(), Excit%, Delay#, LowLim#, HighLim#,
RnTime#, PvDist$, AmpBins%, Rate#, Hue%(), Ch%, Scrap%)

'Continue = 0
'Redo = 0
'RnTime# = "0"
'AmpBins% = "0"
'LowLim# = "0"
'HighLim# = "0"
'PvDist$ = "0"

DO WHILE continue = 0
Redo = 0
continue = 0
COLOR 13
CLS

'MainTitle$ = "RAIN1"
```

```
FOR I = 1 TO 8
  IF Channel$(I) = "Vacant" THEN Hue%(I) = 4
  IF Channel$(I) = "StrGge" THEN Hue%(I) = 2
  IF Channel$(I) = "Txdcer" THEN Hue%(I) = 1
NEXT I
COLOR 1
LOCATE 3, 3: PRINT "File Name:"; MainTitle$
LOCATE 4, 3: PRINT "Interval Rate:"; Rate#
LOCATE 5, 3: COLOR 1
PRINT "Initializing Time (In Seconds)  ";
COLOR 2: PRINT Delay#

FOR I = 1 TO 8
  IF Channel$(I) = "" THEN Channel$(I) = "Vacant"
  COLOR 1
  LOCATE I + 7, 3: PRINT "Channel #"; I; ":";
  COLOR Hue%(I): LOCATE I + 7, 16: PRINT Channel$(I)
  IF Channel$(I) <> "Vacant" THEN
    LOCATE I + 7, 30: COLOR 1: PRINT "Gage Factor: "
    LOCATE I + 7, 43: COLOR Hue%(I): PRINT Factor#(I)
  END IF

NEXT I
COLOR 15: LOCATE 3, 3: PRINT "File Name: "; MainTitle$
COLOR 15: LOCATE 3, 12: INPUT Dumb$
IF Dumb$ = "" THEN
  MainTitle$ = MainTitle$
ELSE
  MainTitle$ = Dumb$
END IF
COLOR 1: LOCATE 3, 3: PRINT "File Name: ";
COLOR 6: PRINT MainTitle$



'COLOR 15: LOCATE 4, 3: PRINT "Interval Rate:"; Rate#
'COLOR 15: LOCATE 4, 16: INPUT Dumb#
'IF Dumb# <> 0 THEN Rate# = Dumb#
COLOR 1: LOCATE 4, 3: PRINT "Interval Rate:";
COLOR 6: PRINT Rate#
LOCATE 5, 3: COLOR 15
PRINT "Initializing Time (In Seconds) ";
COLOR 13: INPUT Dumb#
IF Dumb# <> 0 THEN
  Delay# = Dumb#
END IF
LOCATE 5, 3: COLOR 1
PRINT "Initializing Time (In Seconds)  ";
COLOR 6: PRINT Delay#

FOR I = 1 TO 8

  LOCATE 7 + I, 3: COLOR 15: PRINT "Channel #"; I; ":"; Channel$(I)
  IF Factor#(I) <> 0 THEN
    LOCATE 7 + I, 30: PRINT "Gage Factor: "; Factor#(I)
  END IF

Dumb$ = INPUT$(1)
    IF UCASE$(Dumb$) = "V" THEN
      Channel$(I) = "Vacant"
  Factor#(I) = 0
  LOCATE 7 + I, 3: COLOR 1: PRINT "Channel #"; I; ":"
  LOCATE 7 + I, 16: COLOR 4: PRINT Channel$(I)
  LOCATE 7 + I, 30: PRINT "                    "
  Hue%(I) = 4

ELSEIF UCASE$(Dumb$) = "T" THEN
  Channel$(I) = "Txdcer "
  Hue%(I) = 1
  LOCATE 7 + I, 3: COLOR 1: PRINT "Channel #"; I; ":"
  LOCATE 7 + I, 16: COLOR 2: PRINT Channel$(I)

  DO WHILE Cont = 0
    LOCATE 7 + I, 30: INPUT "Gage Factor: "; Factor#(I)
    IF Factor#(I) > 0 THEN
      Cont = 1
    END IF
  LOOP
  Cont = 0
  LOCATE 7 + I, 30: COLOR 1: PRINT "Gage Factor: "

ELSEIF UCASE$(Dumb$) = "G" THEN
  Channel$(I) = "StrGge"
  Hue%(I) = 2
  LOCATE 7 + I, 3: COLOR 1: PRINT "Channel #"; I; ":"
  LOCATE 7 + I, 16: COLOR 2: PRINT Channel$(I)

  DO WHILE Cont = 0
    LOCATE 7 + I, 30: INPUT "Gage Factor: "; Factor#(I)
    IF Factor#(I) > 0 THEN
      Cont = 1
    END IF
  LOOP
  Cont = 0
  LOCATE 7 + I, 30: COLOR 1: PRINT "Gage Factor: "

ELSEIF Dumb$ = CHR$(13) THEN
  Channel$(I) = Channel$(I)
  Factor#(I) = Factor#(I)
  COLOR 1: LOCATE I + 7, 3: PRINT "Channel #"; I; ":"
  COLOR Hue%(I): LOCATE I + 7, 16: PRINT Channel$(I)
  IF Factor#(I) <> 0 THEN
    LOCATE I + 7, 43: PRINT Factor#(I)
    COLOR 1: LOCATE I + 7, 30: PRINT "Gage Factor: "
  END IF
ELSEIF Dumb$ = CHR$(27) THEN
  LOCATE I + 7, 3: COLOR 1: PRINT "Channel #"; I; ":"; Channel$(I)
  COLOR Hue%(I): LOCATE I + 7, 16: PRINT Channel$(I)
  COLOR 1: LOCATE I + 7, 30:
  IF Factor#(I) <> 0 THEN
    PRINT "Gage Factor:"
    COLOR Hue%(I): LOCATE I + 7, 43: PRINT Factor#(I)
  ELSE
    PRINT "                         "
  END IF
  I = I - 2

ELSE
  I = I - 1
```

```
    END IF

    IF I = -1 THEN
      I = 8
      Redo = 1
    END IF
NEXT I


LOCATE 17, 17: COLOR 7: PRINT "Hit ESC to redo or any other key to continue."

Q$ = INPUT$(1)
  IF Q$ = CHR$(27) THEN Redo = 1

IF Redo <> 1 THEN
CLS
COLOR 8: LOCATE 3, 10: PRINT "Rainflow Settings"
   COLOR 2: LOCATE 5, 10: PRINT "Low Limit Stress=  "; LowLim#
   COLOR 2: LOCATE 6, 10: PRINT "High Limit Stress: "; HighLim#
   COLOR 2: LOCATE 7, 10: PRINT "Number of Bins: "; AmpBins%
   COLOR 2: LOCATE 8, 10: PRINT "Rainflow Interval: "; RnTime#
   COLOR 2: LOCATE 9, 10: PRINT "Minimum Amplitude:  "; PvDist$

   COLOR 10: LOCATE 5, 10: PRINT "Low Limit Stress:"; LowLim#
'   IF Dumb# <> 0 THEN LowLim# = Dumb#
   COLOR 2: LOCATE 5, 10: PRINT "Low Limit Stress:  "; LowLim#

   COLOR 10: LOCATE 6, 10: INPUT "High Limit Stress:"; Dumb#
   IF Dumb# <> 0 THEN HighLim# = Dumb#
   COLOR 2: LOCATE 6, 10: PRINT "High Limit Stres:  "; HighLim#

   COLOR 10: LOCATE 7, 10: INPUT "Number of Bins:"; Dumb%
   IF Dumb% <> 0 THEN AmpBins% = Dumb%
   COLOR 2: LOCATE 7, 10: PRINT "Number of Bins: "; AmpBins%

   COLOR 10: LOCATE 8, 10: INPUT "Rainflow Interval:"; Dumb#
   IF Dumb# <> 0 THEN RnTime# = Dumb#
   COLOR 2: LOCATE 8, 10: PRINT "Rainflow Interval: "; RnTime#

   COLOR 10: LOCATE 9, 10: INPUT "Minimum Amplitude:"; Dumb$
   IF Dumb$ <> "" THEN PvDist$ = Dumb$
   COLOR 2: LOCATE 9, 10: PRINT "Minimum Amplitude:  "; PvDist$

END IF

IF Redo <> 1 THEN

COLOR 1

LOCATE 13, 14: PRINT CHR$(218); STRING$(51, 196); CHR$(191);
LOCATE 21, 14: PRINT CHR$(192); STRING$(51, 196); CHR$(217);
   FOR Row = 14 TO 20
      LOCATE Row, 14: PRINT CHR$(179)
      LOCATE Row, 27: PRINT CHR$(179)
      LOCATE Row, 40: PRINT CHR$(179)
      LOCATE Row, 53: PRINT CHR$(179)
      LOCATE Row, 66: PRINT CHR$(179)
   NEXT Row
   LOCATE 17, 14: PRINT STRING$(53, 196)
      LOCATE 13, 27: PRINT CHR$(194)
      LOCATE 13, 40: PRINT CHR$(194)
      LOCATE 13, 53: PRINT CHR$(194)
      LOCATE 17, 14: PRINT CHR$(195)
      LOCATE 17, 27: PRINT CHR$(197)
      LOCATE 17, 40: PRINT CHR$(197)
      LOCATE 17, 53: PRINT CHR$(197)
      LOCATE 17, 66: PRINT CHR$(180)
      LOCATE 21, 27: PRINT CHR$(193)
      LOCATE 21, 40: PRINT CHR$(193)
      LOCATE 21, 53: PRINT CHR$(193)

FOR I = 1 TO 4
   COLOR 8: LOCATE 18, (2 + 13 * I): PRINT I
   COLOR 8: LOCATE 14, (2 + 13 * I): PRINT I + 4
NEXT I

FOR I = 1 TO 4
   COLOR Hue%(I)
   LOCATE 19, (5 + I * 13): PRINT Channel$(I)
   COLOR Hue%(I + 4)
   LOCATE 15, (5 + I * 13): PRINT Channel$(I + 4)
   COLOR 14

   IF Factor#(I) <> 0 THEN
      LOCATE 20, 3 + 13 * I: PRINT "GF ="; Factor#(I)
   END IF

   IF Factor#(I + 4) <> 0 THEN
     LOCATE 16, 3 + 13 * I: PRINT "GF ="; Factor#(I + 4)
   END IF
NEXT I
LOCATE 23, 18: COLOR 15: PRINT "Is This Information Correct ('Y' To Proceed)"
LOCATE 24, 46: PRINT "('Esc' To Abort)"
LOCATE 24, 46: PRINT "(Any Other Key To Redo)"
Conf$ = INPUT$(1)
   IF UCASE$(Conf$) = "Y" OR UCASE$(Conf$) = CHR$(13) THEN
      continue = 1
   ELSEIF UCASE$(Conf$) = CHR$(27) THEN
     continue = 1
     Scrap% = 1
   ELSE
      continue = 0
END IF
END IF
LOOP

Ch% = 0
FOR n = 1 TO 8
   IF Factor#(n) <> 0 THEN Ch% = Ch% + 1
NEXT n

END SUB

SUB TruckPrint (Rate#, MainTitle$, Factor#(), INIT#(), Excit%)

'*****************************************************************
'*      This sub creates the download program for the data-      *
'*      logger.  The program is saved to a file under the        *
'*      variable Title$.  The extention (.dld) is added below.   *
'*      Initialized values for individual channels are imported *
'*      via the Init#() variable.  All other information is     *
'*      user input through the TurckTest Sub.                   *
'****************************************************************

'****************************************************************
'*      Empty channels employ the Dumb variable to flag an      *
'*      excitation value of 0.                                  *
'****************************************************************

Dumb = 0
Title$ = MainTitle$ + ".dld"

      OPEN Title$ FOR OUTPUT AS #1
      PRINT #1, "};21X"
      PRINT #1, ";"; Title$
      PRINT #1,
      PRINT #1, "MODE 1"
      PRINT #1, "SCAN RATE"; Rate#
      PRINT #1,


'****************************************************************
'*      QBasic will not print "1:P6" to a file correctly.  The  *
'*      Output appears as '1 : P6'.  It is for this reason that *
'*      the code for this sequence seems odd.                   *
'*      The n\ 5 + (n MOD 5) is a clever way of selecting the   *
'*      appropriate excitation channel for the corresponding    *
'*      input channe.                                           *
'****************************************************************

I = 0
FOR n = 1 TO 8
  IF Factor#(n) <> 0 THEN
  I = I + 1

      PRINT #1, CHR$(I + 48); ":P6" '** Command No. 6
      PRINT #1, "1:1"  '** Repetition
      PRINT #1, "2:11"  '** Resolution
      PRINT #1, "3:"; CHR$(n + 48)'** Input channel
      PRINT #1, "4:"; CHR$(n \ 5 + (n MOD 5) + 48)' "** Excitation channel"
      PRINT #1, "5:"; Excit% * 1000 '** Excitation voltage
      PRINT #1, "6:"; CHR$(I + 48)'** Storage location
      PRINT #1, "7:"; 29 * (4 / Factor#(n))'** Multiplicity factor
      PRINT #1, "8:"; INIT#(n)'** Initialized offset
      PRINT #1,

END IF
NEXT n

' Set Output Flag High At Beginning To record Every Data Point

      PRINT #1, I + 5; ":P86"'** Do Command
      PRINT #1, "1:10"  '** Output flag high
      PRINT #1,

      PRINT #1, I + 6; ":P80"'** Store value
      PRINT #1, "1:1" '** Final storage
      PRINT #1, "2:1"
      PRINT #1,
      PRINT #1, I + 7; ":P77"
      PRINT #1, "1:0001"
      PRINT #1,

      PRINT #1, I + 8; ":P70"'** Record Value
      PRINT #1, "1:"; I
      PRINT #1, "2:1"

      PRINT #1,
      PRINT #1, "MODE 2"
      PRINT #1, "SCAN RATE 0.0000"
      PRINT #1,
      PRINT #1, "MODE 3"
      PRINT #1,
      PRINT #1, "MODE 4"
      PRINT #1, "1:00"
      PRINT #1, "2:00"
      PRINT #1,
      PRINT #1, "MODE 10"
      PRINT #1, "1:28"
      PRINT #1, "2:64"
      PRINT #1,
      PRINT #1, "MODE 12"
      PRINT #1, "1:00"
      PRINT #1, "2:0000"
      PRINT #1, "  ÿ"

CLOSE #1

END SUB

SUB TruckTest (Rate#, MainTitle$, Factor#(), Scrap%, Channel$(), Hue%(), Ch%,
Delay#)

CLS
I = 8
LOCATE I, 15: PRINT "Main Title: "; MainTitle$
LOCATE I + 1, 15: PRINT "Interval Rate: "; Rate#
LOCATE I + 2, 15: PRINT "Intializing Time: "; Delay#

COLOR 1
LOCATE 13, 14: PRINT CHR$(218); STRING$(51, 196); CHR$(191);
LOCATE 21, 14: PRINT CHR$(192); STRING$(51, 196); CHR$(217);
   FOR Row = 14 TO 20
      LOCATE Row, 14: PRINT CHR$(179)
      LOCATE Row, 27: PRINT CHR$(179)
      LOCATE Row, 40: PRINT CHR$(179)
      LOCATE Row, 53: PRINT CHR$(179)
      LOCATE Row, 66: PRINT CHR$(179)
   NEXT Row
   LOCATE 17, 14: PRINT STRING$(53, 196)
      LOCATE 13, 27: PRINT CHR$(194)
      LOCATE 13, 40: PRINT CHR$(194)
      LOCATE 13, 53: PRINT CHR$(194)
      LOCATE 17, 14: PRINT CHR$(195)
      LOCATE 17, 27: PRINT CHR$(197)
      LOCATE 17, 40: PRINT CHR$(197)
      LOCATE 17, 53: PRINT CHR$(197)
      LOCATE 17, 66: PRINT CHR$(180)
      LOCATE 21, 27: PRINT CHR$(193)
      LOCATE 21, 40: PRINT CHR$(193)
      LOCATE 21, 53: PRINT CHR$(193)

FOR I = 1 TO 4
   COLOR 8: LOCATE 18, (2 + 13 * I): PRINT I
   COLOR 8: LOCATE 14, (2 + 13 * I): PRINT I + 4
NEXT I

FOR I = 1 TO 4
   COLOR Hue%(I)
   LOCATE 19, (5 + I * 13): PRINT Channel$(I)
   COLOR Hue%(I + 4)
   LOCATE 15, (5 + I * 13): PRINT Channel$(I + 4)
   COLOR 14

   IF Factor#(I) <> 0 THEN
      LOCATE 20, 3 + 13 * I: PRINT "GF ="; Factor#(I)
   END IF

   IF Factor#(I + 4) <> 0 THEN
     LOCATE 16, 3 + 13 * I: PRINT "GF ="; Factor#(I + 4)
   END IF
NEXT I
LOCATE 23, 18: COLOR 15: PRINT "Is This Information Correct ('Y' To Proceed)"
LOCATE 24, 18: PRINT "  ('Esc' To Abort -- Redo Input In Set Up)"

Conf$ = " "
WHILE INSTR(CHR$(13) + CHR$(27) + "Y", Conf$) = 0
   Conf$ = UCASE$(INPUT$(1))
WEND

IF Conf$ = "Y" OR Conf$ = CHR$(13) THEN
   Scrap% = 0
ELSEIF Conf$ = CHR$(27) THEN
   Scrap% = 1
END IF

Ch% = 0
FOR n = 1 TO 8
   IF Factor#(n) <> 0 THEN Ch% = Ch% + 1
NEXT n

END SUB

SUB VarrayRead (TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)

OPEN "VARS.SET" FOR INPUT AS #1

INPUT #1, AmpBins%

INPUT #1, Ch%

FOR I = 1 TO 8
   INPUT #1, Channel$(I)
NEXT I

INPUT #1, Count%
INPUT #1, Counter#
INPUT #1, Delay#
INPUT #1, Excit%

FOR n = 1 TO 8
   INPUT #1, Factor#(n)
NEXT n

INPUT #1, HighLim#

FOR n = 1 TO 8
   INPUT #1, INIT#(n)
NEXT n

INPUT #1, LowLim#
INPUT #1, MainTitle$
INPUT #1, OtherTitle$
INPUT #1, PvDist$
INPUT #1, Rate#
INPUT #1, RnTime#
INPUT #1, Status$
INPUT #1, TimeSet%
INPUT #1, Tipe$

CLOSE #1

END SUB

SUB VarraySave (TimeSet%, Rate#, MainTitle$, Factor#(), Excit%, LowLim#,
HighLim#, RnTime#, PvDist$, AmpBins%, Status$, Channel$(), Ch%, Count%,
Counter#, Delay#, INIT#(), OtherTitle$, Tipe$)

OPEN "VARS.SET" FOR OUTPUT AS #1

PRINT #1, AmpBins%
PRINT #1, Ch%

FOR I = 1 TO 8
   PRINT #1, Channel$(I)
NEXT I

PRINT #1, Count%
PRINT #1, Counter#
PRINT #1, Delay#
PRINT #1, Excit%

FOR n = 1 TO 8
   PRINT #1, Factor#(n)
NEXT n

PRINT #1, HighLim#

FOR n = 1 TO 8
   PRINT #1, INIT#(n)
NEXT n

PRINT #1, LowLim#
PRINT #1, MainTitle$
PRINT #1, OtherTitle$
PRINT #1, PvDist$
PRINT #1, Rate#
PRINT #1, RnTime#
PRINT #1, Status$
PRINT #1, TimeSet%
PRINT #1, Tipe$

CLOSE #1


END SUB

SUB WarmUp (Nulll$)
'*****************************************************************
'*      This sub sends the dummy Warm-up program to the data-    *
'*      logger and starts a delay timer while channels are      *
'*      initializing.                                           *
'*****************************************************************

TurnOn# = TIMER
CLS
Q$ = ""

Esc$ = INKEY$
COLOR 8
LOCATE 12, 15: PRINT "Channels Will Warm Up As Soon As A Time Is Entered."
LOCATE 16, 15: PRINT "(Enter 0 For The Delay Option.)"
LOCATE 15, 15: INPUT "Enter The Warm-Up Time Period In Minutes."; Delay#
COLOR 14: LOCATE 15, 15: PRINT "Timer:    ";
IF Delay# > 0 THEN
  DO
    PRINT USING "###.#"; TIMER - TurnOn#
    LOCATE 17, 15: PRINT "End Time: "; Delay#
    LOCATE 20, 17: COLOR 4: PRINT "(Hit ESC Key To Cancel)"
  LOOP UNTIL Esc$ = CHR$(27) OR TIMER >= TurnOn# + Delay#
ELSE
  DO
      LOCATE 15, 15: PRINT "Gages Are In Standby Mode!"
      LOCATE 20, 17: PRINT "Hit ESC Key To Return To Main Menu!)"
  LOOP UNTIL Esc$ = CHR$(27)
END IF

END SUB
```

VITA

John Welton Kelly

Candidate for the Degree of

Master of Science

Thesis: DATA ACQUISITION SYSTEM FOR FATIGUE LIFE ESTIMATION OF BRIDGE COMPONENTS

Major Field: Civil Engineering

Biographical:

Personal Data: Born in Fort Dix, New Jersey, March 17, 1971, the son of John W. and R. June Kelly. Married to Kristen Marie Myers on November 18, 1995.

Education: Graduated from James W. Robinson Secondary School, Fairfax, Virginia, in June, 1989; received the Bachelor of Science degree in Civil Engineering from Oklahoma State University in May, 1994; completed requirements for the Master of Science degree at Oklahoma State University in December, 1995.

Professional Experience: Research Assistant, School of Civil and Environmental Engineering, Oklahoma State University, May, 1994, to December, 1995.