



Fracture Technology Associates



**Automated Fatigue Crack Growth Testing and Analysis – Series 2001
Version 3.12**

Users' Variable Amplitude Manual

April 2012

Keith Donald
Amy Blair
Fracture Technology Associates
2412 Emrick Boulevard
Bethlehem, PA 18020
610-868-3054
www.FractureTech.com

Table of Contents

1.0	Introduction to Fracture Technology Associates' Variable Amplitude Software....	5
1.1	System Concept	5
1.2	About This Manual	5
2.0	Using the FTA Variable Amplitude Testing Software.....	7
3.0	Variable Amplitude Testing Software: Description of Menus and Tabs	10
3.1	Splash Screen	10
3.2	Data Input	11
3.2.1	Initialize.....	11
3.2.2	Stress Parameters	12
3.3	Control: Variable Amplitude	14
3.3.1	Full Function Generator	14
3.3.2	Spectrum File.....	17
3.3.2.1	End Point File.....	17
3.3.2.2	Endpoint File Filter	22
3.3.2.3	Validation Log Filter	23
3.3.2.4	Block File	25
3.3.2.5	Segment File.....	28
3.3.3	Correction File	30
3.4	Display	31
3.4.1	Status: variable amplitude.....	31
3.4.2	Load – Displacement Display	34
3.4.3	Load Scan Display.....	35
4.0	Variable Amplitude Analysis Software	39

1.0 Introduction to Fracture Technology Associates' Variable Amplitude Software

1.1 System Concept

Fracture Technology Associates' Variable Amplitude (VA) software provides additional functionality to the standard Fatigue Crack Growth Rate package by allowing the use of user-generated waveforms far more complex than those permitted by the basic application.

The software is characterized by five unique features:

- A waveform output having a constant loading rate with upper and lower frequency limits.
- A command correction compensation based on endpoint errors: each endpoint error is associated with the unique combination of the previous, current and next endpoint.
- A real time damage parameter used to estimate the success in attaining peak loads in a spectrum test.
- The storage of command correction factors unique to each spectrum and test conditions so that endpoint errors can be minimized at the beginning of duplicate tests.
- A validation log for cataloguing 'missed endpoints' beyond a user -specified error threshold.

The software supports a wide variety of file types, including spectra characterized by load peaks and valleys (end-point files); repetitive load peak and valley test sequences (block files); and customized waveform shapes that allow computation of crack size within one cycle and more flexibility than the custom waveform feature of the steady-state application (segment files). A fourth category, flight mix files, is not recommended for most users (contact FTA for details, if required). In-depth discussion of these files (including instructions for their construction and issues arising) begins on page 17 of this manual.

1.2 About This Manual

This document is intended to provide support for the use of FTA's VA software only, particularly as it differs from the standard FCGR package. Full familiarity with the standard system (such as software/hardware setup and the use of sine and other simple waveforms) is assumed. Readers seeking such background information are directed to FTA's steady state reference manual.

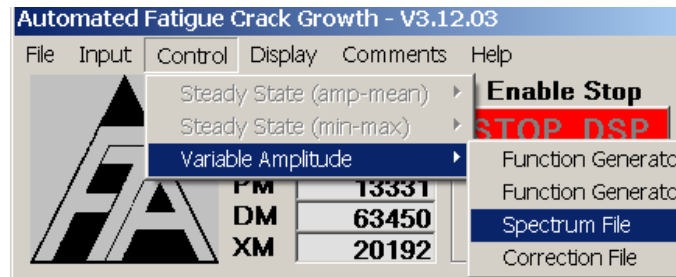
The following section of this guide features a sample test procedure, which illustrates the use of the VA application and highlights the ways in which it differs from the standard software. Next may be found a discussion of the menus and displays unique to the program, including detailed consideration of waveform files and their construction. A final section covering the use of the VA analysis program is under construction: please contact FTA with any questions concerning VA analysis.

2.0 Using the FTA Variable Amplitude Testing Software

The following test procedure provides instructions for running a variable amplitude test using an end-point spectrum file. The procedure is intended only to highlight differences between the VA software and the steady-state version, with particular attention paid to unique features of the function generator. Readers seeking a more basic understanding of FTA's testing applications (including details of file types; setup of compliance and DC potential drop tests; and basic menus and displays) are directed to Section 3 of the standard steady state manual.

1. Ensure the Adwin is powered on and the green light is blinking, indicating that communication with the computer has been established. Reboot as necessary.
2. Open the FCGR testing program. On the initial **Splash Screen**, chose the crack monitor type (**Compliance** or **DCPD**) and the **Variable Amplitude** application. While the default ADwin memory size of 16M is appropriate for most applications, choose the 32M option if the spectrum file contains more than 2,500,000 endpoints (files may contain up to 10,000,000 endpoints). Note that the 32M option requires an ADwin with an integrated USB port.
3. Start a **New Test**.
4. Choose the **Unit Type (English or Metric)** and select **Input Data**. Start the test as usual by entering the test ID; loading a parameter file; and/or choosing the appropriate K-calibration type. The **Initialize** check boxes are identical to those on the steady state menu, with the addition of the **Clear Correction** option. Enabling **Clear Correction** and **Send[ing] to DSP** initializes the creation of a new correction file and clears the correction factors created during the previous test. See Step 13, below, and page 30 of this manual for more information about correction files.
5. Address the **Coefficients** and **Dimensions** tabs. These function in the same way as those in the standard software. Input specimen dimensions and load cell and other calibrations as appropriate.
6. Proceed to the **Stress Parameters** tab. It is important to note that although this tab looks similar to the steady state version, it operates somewhat differently (see page 12) in the VA software. Although three modes of control are available, in most cases **Constant Load** is the appropriate choice. In the **Maximum Load** field, input the value (in engineering units) that corresponds to the full-scale (reference) load level specified in the spectrum file (see page 17 for details about the terms used and spectrum files themselves). *Note that **Stress Ratio** remains inactive and the **Constant Kmax** control mode is unavailable in the VA software.*
7. The **Data Acquisition** and **Data Storage** tabs operate identically to those in the steady state software. **Send input to DSP** and **Print to Log** as usual.
8. Address the **Options** and **General Information** forms as usual.
9. Under **Control (Variable Amplitude)**, open the **Full and Mini Function Generators**. FTA recommends that for VA testing, **Command Control** be set to **Firm**. This corrects the command signal by one-half the endpoint error and accelerates the generation of command compensation parameters necessary to minimize endpoint errors. **Specify Loading Rate** and **Frequency** (see page 14 of this manual for details).

- Under **Control (Variable Amplitude)** select **Spectrum File** and choose the appropriate file type from the dropdown menu – in this case, an end-point file (for full descriptions of spectrum files recognized by the software, see Section 3.3.2 of this manual).



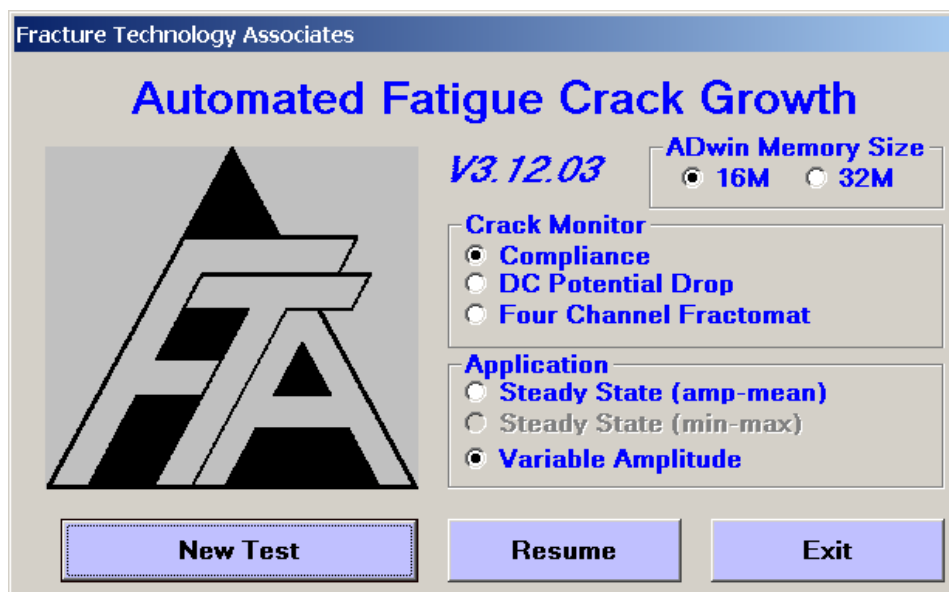
- When the end-point file popup menu appears, press **Read from File** and navigate to the appropriate .dat file. When the file has been loaded into the program, its name and file path will appear at the bottom of the form. **Send to DSP**. The individual lines of the file may be examined by navigating **Back/Forward One** or the file may be edited in the same manner as user-defined waveforms in the steady state software (see page 17 for full details).
- Start/Enable Validation** if a record of target and actual loads (validation, or .vld, file) is required.
- It may be appropriate at this point to return to **Control (Variable Amplitude)** and select a **Correction File** (.cor) from a previous duplicate test. *Correction arrays (see page 30 for a full description) are created automatically during VA tests as a record of the tuning parameters required to minimize load target-feedback errors. By loading the resulting correction factors into a new test, initial rapid optimization of the command signal can be assured. These files should be used with caution, however, since they are sensitive to any changes in test machine dynamics or specimen compliance. FTA recommends that correction files only be used on duplicate tests on individual load frames. Correction files should never be used with K-control VA tests.* It is important to remember that although correction arrays are created automatically, they are not saved unless stored manually by the user at the end of the test (**Correction File** menu).
- Open the **Status, Load-Scan, dadN/dK**, and other displays as is customary.
- Load the specimen and instrument it as usual. Zero the DCPD conditioner or clip gage.
- Enable external control and observe the load on the **Status** screen. The load will adjust to the spectrum file's first endpoint.
- Here follows a key difference between the steady-state and VA software both in the order of operations and the utility of the **Command Control** function. On the **FNG**, ensure **FNG Shutdown** and **DCPD/Slope Analysis** *only* are ticked. Press **Run** to start cycling *without* ticking **Command Control**. Do not activate **Command Control** without first verifying the system's correct operation through observation of the **End Point Error Status** (bottom left of the **FNG**) and the **Damage [Parameter]** (lower left of **Status** screen) as follows:

- **End Point Error Status:** this value displays the error between the target and actual loads at each endpoint. This offset may be minimized at the start of the test by gradually adjusting the controller's set point until the end-point error is close to zero. It may also be necessary to adjust the controller's proportional gain.
 - **Damage [Parameter]:** this value (Γ) estimates the program's success in attaining load target accuracy (see Section 3.3.1 for more details). A value of 1.000 ± 1.001 is desired and may be attained initially (before **Command Control** is activated) through judicious adjustment of the controller's proportional gain. After **Command Control** has been activated, the **Damage [Parameter]** will, over time, approach unity as the command signal is corrected automatically by the program.
18. When satisfied that endpoint errors are minimal and the **Damage [Parameter]** is acceptable, tick **Command Control**. This will enable further automatic adjustment of the command signal to minimize target-feedback errors.
 19. For DCPD tests, enter **DCPD active/reference** values in the **Initial PD** field on the **Dimensions** tab to establish an initial crack length.
 20. Observe the **Load Scan** display to ensure good agreement between target loads and feedback signals from the controller. The command compensation may also be examined on this screen (page 35).
 21. Turn on **Data Storage**.
 22. Verify correct ramp rate and adjust as necessary.
 23. The test will finish when final limits are reached. Save the analysis file, the correction file (if required), and perform final housekeeping as with a steady state test.

3.0 Variable Amplitude Testing Software: Description of Menus and Tabs

The following section offers a complete description of the displays and menus for variable amplitude testing only, with differences highlighted between these and their steady state counterparts. All menus and displays common to VA and steady state testing are omitted here. Users may refer to Section 4 of the steady state reference manual for information regarding these displays.

3.1 Splash Screen



When the FCGR testing program is first opened, the **Splash Screen** is displayed. The application title and version number are identified at the top of this form. All standard applications use the default 16M ADwin memory size option. The 32M option is used only with the variable amplitude application, when endpoint files contain more than 2,500,000 endpoints (up 10,000,000 endpoints with 32M option). The 32M option requires ADwins with the integrated USB port.

From this form the **Crack Monitor** type (**Compliance**, **DC Potential Drop**, or **Four Channel Fractomat**) is selected. If a different crack monitor type is required for subsequent tests, the program must be shut down completely and restarted. Select the **Variable Amplitude** application.

3.2 Data Input

3.2.1 INITIALIZE

The screenshot shows the 'Data Input' software window with the 'Initialize' tab selected. The window title is 'Data Input' and it has standard Windows window controls. The 'Initialize' tab is active, and the other tabs are 'Coefficients', 'Dimensions.....', 'Stress Parameters', 'Data Acquisition', and 'Data Storage'. The 'Test ID' field contains the text 'test'. Below this are three buttons: 'Load Test Parameters', 'Save Test Parameters', and 'Start Log File'. The 'Test parameter path and filename' field contains the path 'I:\Software Development\Current Software Development\FCGR'. The 'Data storage path' field also contains 'I:\Software Development\Current Software Development\FCGR'. On the right side, there is a section for 'K Calibration Type' with an unchecked 'Enable Cal Type' checkbox and a list of radio button options: 'Polynomial (Type 1)', 'Polynomial (Type 2)', 'Center Crack' (which is selected), 'SEN (pin loaded)', 'SEN (rigid)', 'Bend', 'Extended Compact', 'Surface Crack', 'Corner Crack', and 'Brazilian Disk'. In the bottom left, there is an 'Initialize' section with four checked checkboxes: 'Initialize', 'Clear Data Storage Buffer', 'Clear Cycle Count', and 'Clear Elapse Timer'. Below this is another checked checkbox labeled 'Clear Correction'. At the bottom of the window are three buttons: 'Receive from DSP', 'Send to DSP', and 'Print to Log'.

The VA **Initialize** tab is identical to the steady state version with the exception of the **Clear Correction** check box in the bottom left corner. When initializing a new test, ticking this box and **Send[ing] to DSP** clears all the correction factors from the previous test (see page 30 for more information on correction files).

3.2.2 STRESS PARAMETERS

The screenshot shows the 'Data Input' window with the 'Stress Parameters' tab selected. The window contains the following fields and controls:

- Maximum Load (lbs):** 5000.0
- Stress Ratio (R):** 0.00
- Control Type:**
 - Constant Load
 - K Control - (Constant C)
 - K Control - (Look-up Table)
- K Control Mode:**
 - Fixed Stress Ratio
 - Constant Kmax
- K Control Parameters:**
 - Initial a (in): 0.00
 - Initial Kmax (ksi sqrt[in]): 0.0
 - K-Gradient (1/in): 0.00
 - Initial Delta K (ksi sqrt[in]): 0.0
 - Final a (in): 0.00
 - Final Kmax (ksi sqrt[in]): 0.0

Buttons at the bottom: **Receive from DSP**, **Send to DSP**, and **Print to Log**.

Although superficially identical to the steady state **Stress Parameters** tab, the VA version functions somewhat differently.

Maximum Load designates the load associated with the full-scale reference value specified in the end-point file (see page 17 for more details on endpoint file creation). For example, if the full-scale value in the file is 100 and the **Maximum Load** entered on the **Stress Parameters** tab is 5,000 lbs, then endpoints of 100 will prompt a command signal equivalent to 5,000 lbs of force. Note that this force may be exceeded if the file contains endpoints with values higher than the full-scale value of 100.

Three modes of K control are possible:

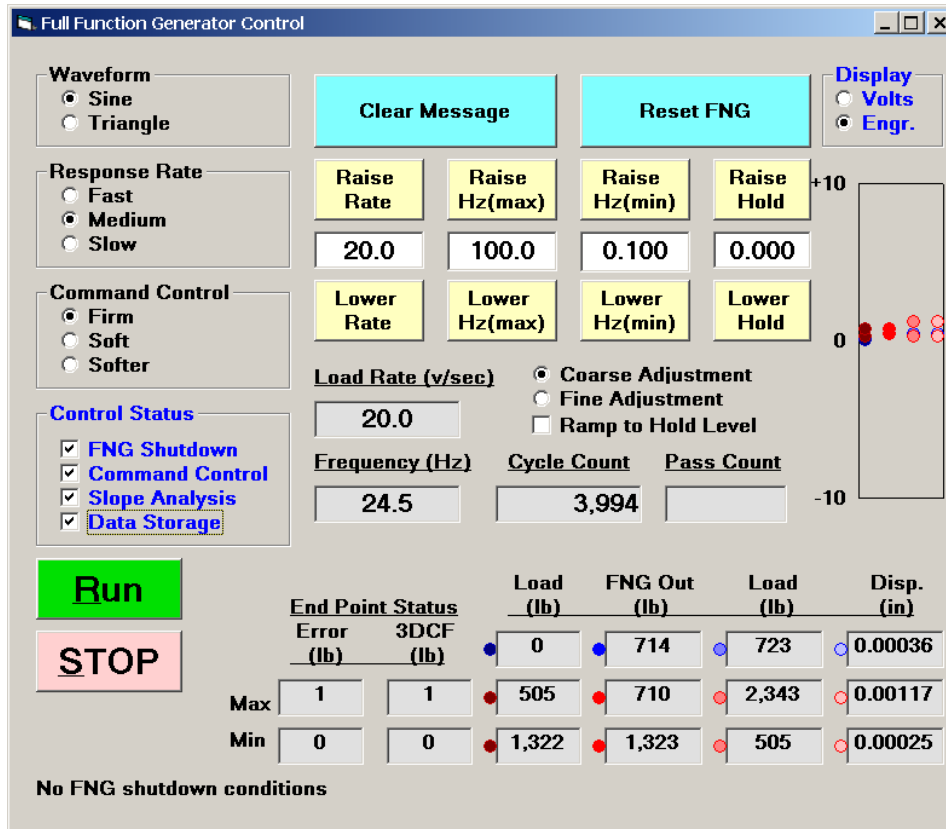
- **Constant load:** enables the user to specify the force (in engineering units) associated with a full-scale (reference) load level, which will remain constant throughout the test. Thus, all endpoints are associated with designated loads that do not change as the test progresses.
- **K Control (Constant C):** scales the spectrum according to an **Initial Kmax** and **K-gradient** in a manner similar to the steady state software. For this option, the **Initial Kmax** is associated with the full-scale load level specified in the endpoint file. The spectrum is subsequently scaled proportionally to the change in **Kmax** and the **Maximum Load** field is updated by the software when the user **Receive[s] from DSP**.

- **K Control (Look-up Table)**: scales the spectrum through means of a K-profile provided by the user in a look-up table.

In the VA software, **Stress Ratio** is inactive and the **Constant Kmax** control mode is unavailable.

3.3 Control: Variable Amplitude

3.3.1 FULL FUNCTION GENERATOR



It is important to remember that **Command Control** on the VA function generator (**FNG**) operates somewhat differently from its steady state version. In common with the steady-state software, when **Command Control** is selected the function generator will adjust the command signal to minimize errors between target load and actual load.

Unlike the steady-state software, however, the VA **FNG** outputs the full command signal required by the spectrum waveform *even if **Command Control** is not enabled*. Activation of **Command Control** initiates correction of the command signal based on endpoint errors and takes into account the unique sequence of the previous, current and next endpoint as it acquires the appropriate command corrections. Extra caution is therefore required when initiating **Command Control** due to this characteristic. See the test procedure in Section 2 of this manual for FTA's recommended order of operations.

The **Firm** setting is recommended for variable amplitude testing to optimize the speed at which command-correction parameters are obtained.

A variety of control choices unique to the VA application are available on the **FNG**. Key among these are *load rate* and *frequency*, which are specified in text boxes associated

with the yellow buttons labeled **Raise/Lower [Ramp] Rate** and **Raise/Lower Hz (min/max)**.

- **Raise/Lower [Ramp] Rate:** specifies a constant loading rate in terms of volts per second. A constant loading rate is the default control for all VA tests, although this may be overridden through use of the **Hz (min-max)** options described below.

Calculate the volts/second rate as follows:

$$\text{Volts / second} = (\text{desired force / second}) / (\text{load cell calibration})$$

- **Raise/Lower Hz (max):** specifies an upper frequency limit in cycles/second. This allows the user to ensure that the ramp rate never exceeds a chosen frequency. Should particular load peaks or valleys in the waveform require the ramp rate to surpass the maximum frequency specified, the ramp rate is slowed accordingly.
- **Raise/Lower Hz (min):** allows the user to specify lower frequency limit in cycles per second. See **Raise/Lower Hz (max)**, above.

The user is encouraged to experiment with load rate and frequency combinations in simulation to gain a complete understanding of this capability and/or evaluate the command signal prior to testing. Note that consistent units (English or Metric) between the desired force and load cell calibration factor must be used.

An additional text box labeled **Raise/Lower Hold** allows the user to specify a particular hold level when the test is stopped. This value may be changed if the **Ramp to Hold Level** checkbox has been activated. The default hold level (where **FNG Hold** is set to zero) is that of the next endpoint in the file.

A variety of status displays appear at the bottom of the **FNG**:

- **Cycle Count:** this count is half the number of endpoints or reversals in loading.
- **Pass Count:** the number of complete passes through the spectrum file.
- **End-point Status:**
 - **Error:** the difference between target load and actual load for each endpoint. A negative value indicates that the actual load is higher than the target load for that endpoint.
 - **3DCF** (Three-Dimensional Correction Factor). This is the correction that is applied to the command signal to minimize endpoint error. It is retrieved from a three-dimensional array unique to the combination of a previous, current, and next endpoint. The correction factor is based on historical

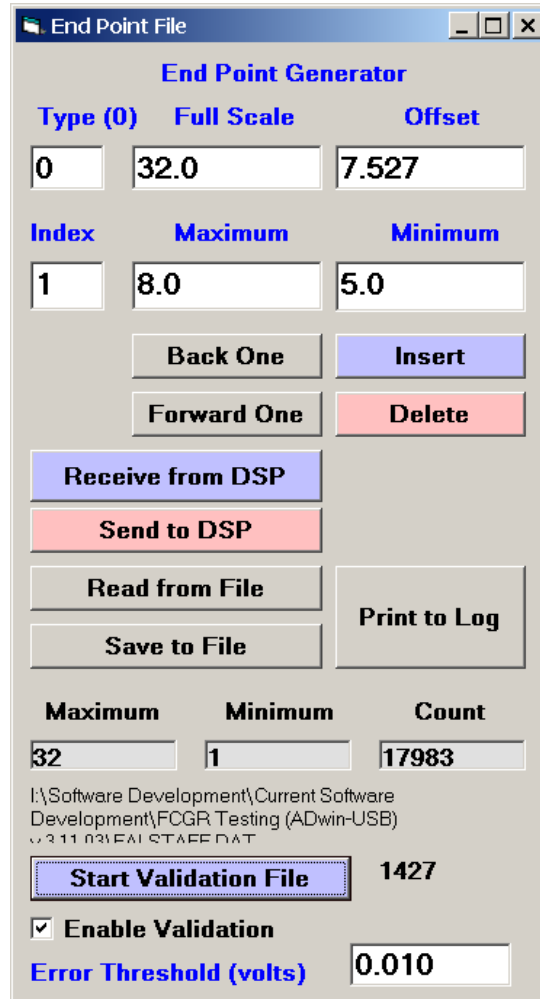
performance and is not updated until the three-endpoint combination is repeated.

- **[Target] Load:** the load specified in the endpoint file.
- **FNG Out:** the load (command) signal from the ADwin.
- **[Actual] Load:** the load (feedback) signal from the controller.

3.3.2 SPECTRUM FILE

Spectrum files are loaded/edited from the **Main Menu** (Control → Variable Amplitude → Spectrum File → File Type). Four types of spectrum files are recognized by the VA testing program: endpoint files, block files, flight mix files, and segment files. Each of these is discussed more fully, below, except for flight mix files, which are highly specialized and not appropriate for most users.

3.3.2.1 End Point File



An **End-Point File** is characterized by a list of load peaks and valleys. The actual loads, ramp rate, and other control functions are determined on the **Stress Parameters** tab and **Function Generator**.

From the **End-point File** menu, an existing spectrum file (.dat extension) may be loaded into the program by pressing the **Read from File** button. Once this is done, it can be examined by scrolling through the file by using the **Back One** and **Forward One** buttons. A spectrum can also be created from within the program. Details about the composition of end-point files; notes on their construction; and further information about the **End-point File** menu appear below.

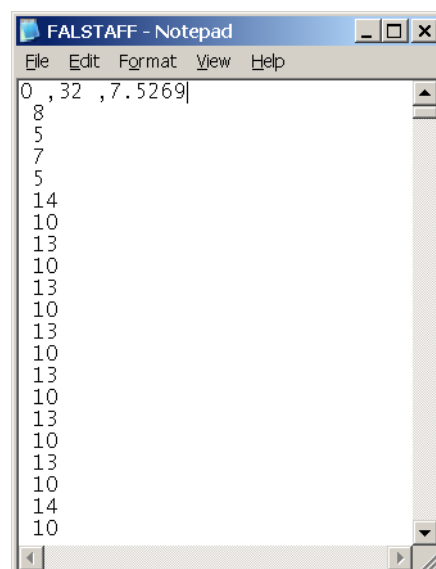
Note: It is important to remove non-turning endpoints (that is, endpoints that do not represent a cycle's maximum or minimum amplitude) from a spectrum file before loading it into the testing software. Removal of non-turning endpoints can be accomplished with the filter utility available in the VA analysis program. See the next section for details.

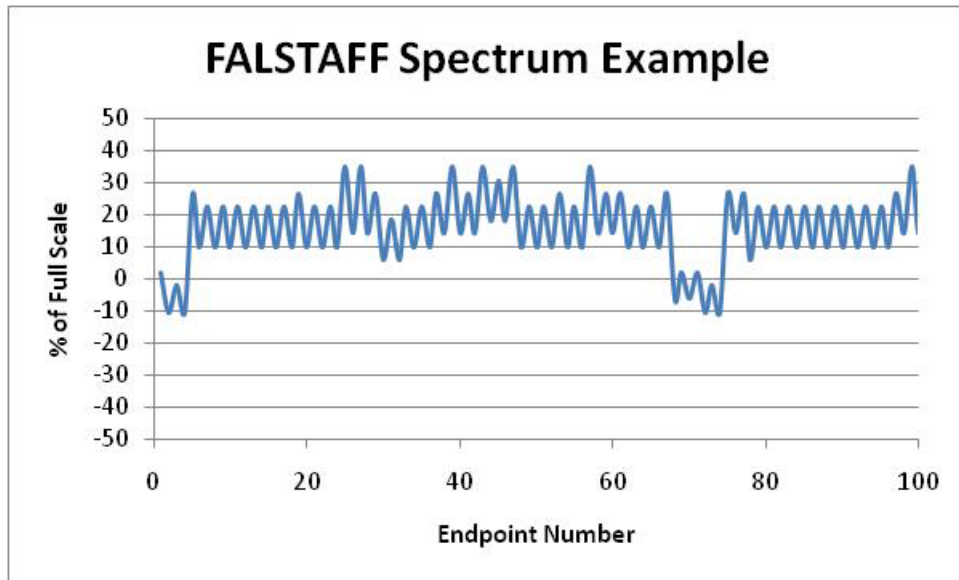
An end-point file consists of a header containing three values (**Type**, **Full Scale** and **Offset**), which may be viewed or edited in the text boxes at the top of the menu. Each additional line in the file contains a value corresponding to a target peak and valley endpoint (The actual load associated with the full-scale value for any given test is designated on the **Stress Parameters** tab of the **Data Input** form). Each index is associated with two endpoints, or one cycle. **Insert** and **Delete** buttons allow creation and deletion of indices. The file can then be saved using the **Save to File** button.

Most spectrum files are characterized by a full-scale value of 100 and an offset value of 0, which corresponds to zero load. Endpoints usually fall on or between +100 and -100 and do not have to be integers (see the example below). Although not recommended, endpoints may fall outside this range as long as they are not outside the range of the load cell (beyond -10 to +10 volts). However, the errors for these endpoints may not be properly corrected because the three-dimensional array of the command correction factor assumes all endpoints fall on or within the full-scale range.

The example in the above menu is from the FALSTAFF spectrum and is unique in its character because the endpoints are integers from 1 to 32 and zero load occurs between endpoints 7 and 8 (7.5269), a reminder that computer resources were limited when this spectrum was developed.

An example of a FALSTAFF spectrum file showing header information and the first 20 endpoints combined with a graph showing the first 100 endpoints is shown below:





The first three variables in the file contain header information:

- **Type:** an end-point file is always designated by a value of zero in the **Type** field.
- **Full Scale:** designates the value of the endpoint around which the spectrum loads are scaled. In the example above, endpoints with the value of 32 in the spectrum file correspond to the **Maximum Load**, which is set to 5000 lbs. on the **Stress Parameters** tab, below. If the offset value were 0, then an endpoint of 16 would correspond to a 16/32 or 50% of maximum load (2,500 lbs). However, in this example, an offset of 7.5269 means that an endpoint of 16 would be $(16 - 7.5269) / (32 - 7.5269)$ or 34.62% of maximum load.

Data Input

Initialize | Coefficients | Dimensions.... | **Stress Parameters** | Data Acquisition | Data Storage

Maximum Load (lbs)

Stress Ratio (R)

Control Type

Constant Load

K Control - (Constant C)

K Control - (Look-up Table)

K Control Mode

Fixed Stress Ratio

Constant Kmax

K Control Parameters

Initial a (in)

Initial Kmax (ksi sqrt[in])

K-Gradient (1/in)

Initial Delta K (ksi sqrt[in])

Final a (in)

Final Kmax (ksi sqrt[in])

- **Offset:** designates the waveform's zero offset, below which loads are negative. A value of zero scales each endpoint proportionally to the full scale value. A non-zero value *offsets* and scales each endpoint according to the example shown above.

Three text boxes allow for the input of lines in the endpoint file:

- **Index:** designates the sequence of the cycle in the file. A cycle is defined as two load reversals or endpoints. The first index value is one. The index is only displayed in the **End Point File** form, not in the file itself.
- **Maximum:** the maximum endpoint load level of the cycle. Note that the maximum value could exceed the full-scale value, although this is not recommended.
- **Minimum:** the minimum endpoint load level of the cycle. Note that the minimum value could exceed the negative full-scale value, although this is not recommended.

Note: An end-point file can start with either a maximum or minimum value. This is not critical to the proper function of the software as long as there are no non-turning endpoints (that is, endpoints that do not occur at the maximum or minimum amplitude of a waveform's cycle) and the file has an even number of endpoints.

Several navigation buttons allow examination and editing of the endpoint file:

- **Back One:** scrolling feature that decreases the index and endpoint pair by 1
- **Forward One:** scrolling feature that increases the index and endpoint pair by 1
- **Insert:** allows the insertion of an endpoint pair or cycle in the spectrum sequence
- **Delete:** allows the deletion of an endpoint pair or cycle in the spectrum sequence
- **Receive from DSP:** recalls the spectrum file that already resides in the ADwin (after a computer crash, for example).
- **Send to DSP:** sends the spectrum file to the ADwin
- **Read from File:** allows the user to select and load into the program a previously-created spectrum file
- **Save to File:** saves a spectrum file created or edited within the VA program.

- **Print to Log:** prints the spectrum file to the .log file

Three display boxes offer at-a-glance details of the endpoint file:

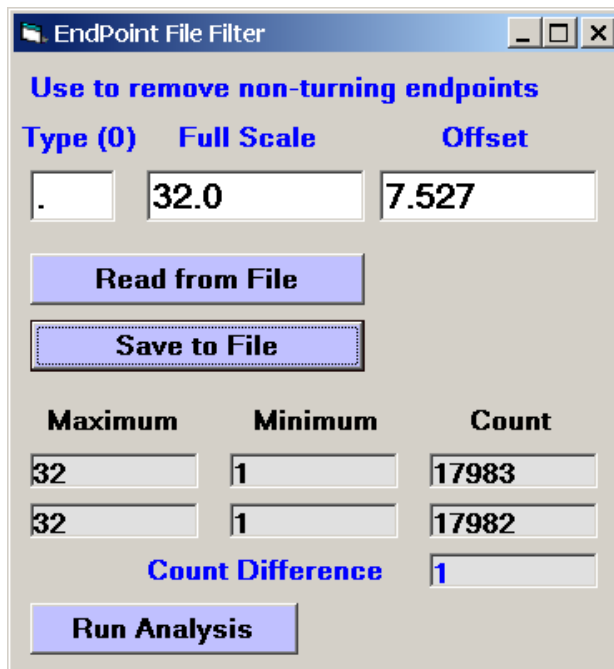
- **Maximum:** the endpoint[s] in the file with the highest value
- **Minimum:** the endpoint[s] in the file with the lowest value
- **Count:** the total number of cycles in the endpoint file (half the number of endpoints)

The **Endpoint File** menu also allows for the creation of a validation file (.vld extension), which records target and actual loads during the test. Press **Start Validation File** to create the file and **Enable Validation** to transfer data to the file at 1000-cycle intervals. Validation files contain five columns as follows:

1. Cycle count
2. Maximum target load (volts)
3. Minimum target load (volts)
4. Maximum actual load (volts)
5. Minimum actual load (volts)

Since an unfiltered validation file can become very large, the **Error Threshold (volts)** text box allows selecting an error threshold between target and actual load. Errors below the threshold will not be stored in the validation file, substantially reducing the file size. Typically values are 0.010 to 0.020 volts.

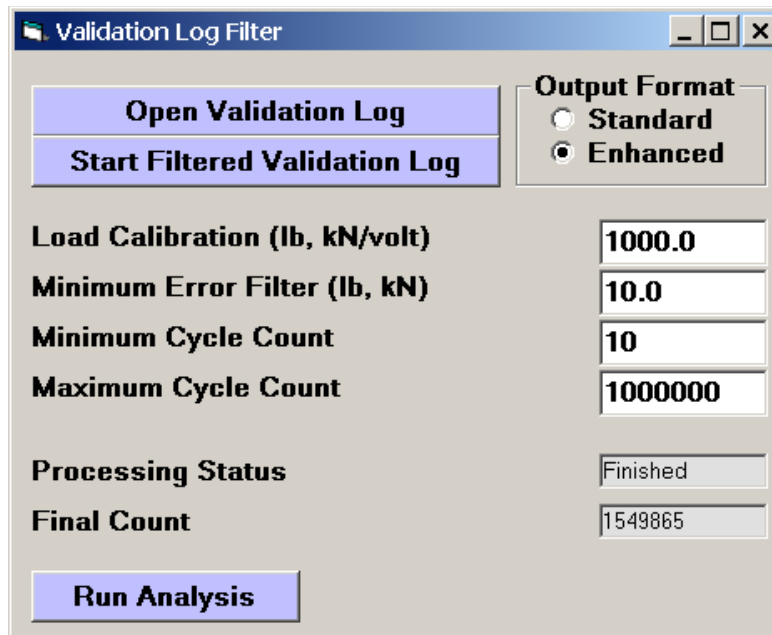
3.3.2.2 Endpoint File Filter



The **End Point File Filter**, which may be found in FTA's analysis program, is a utility for removing non-turning endpoints from end-point files. It is important to avoid non-turning endpoints in all spectrum files (a non-turning endpoint is one that does not occur at the maximum or minimum amplitude of a waveform's cycle). The routine allows the user to scan a spectrum file for non-turning endpoints and remove them automatically, creating in the process a new spectrum file. Use of this routine requires that the user **Read from File**, and **Run Analysis**, followed by **Save to File**. It is a good idea to save the resulting file under a new name, to avoid losing the original file.

NB Spectrum files such as block files and segment files cannot be edited with the **End Point File Filter**.

3.3.2.3 Validation Log Filter



The **Validation Log Filter**, which may be found in FTA's analysis program, is a utility for enhancement of the raw validation log file (.vld). This routine allows further filtering of validation logs according a minimum error determined by the user as well as a minimum and maximum cycle count.

In the **Standard Output Format**, the output file (.vl2) generates the same five columns of data as the input file (.vld).

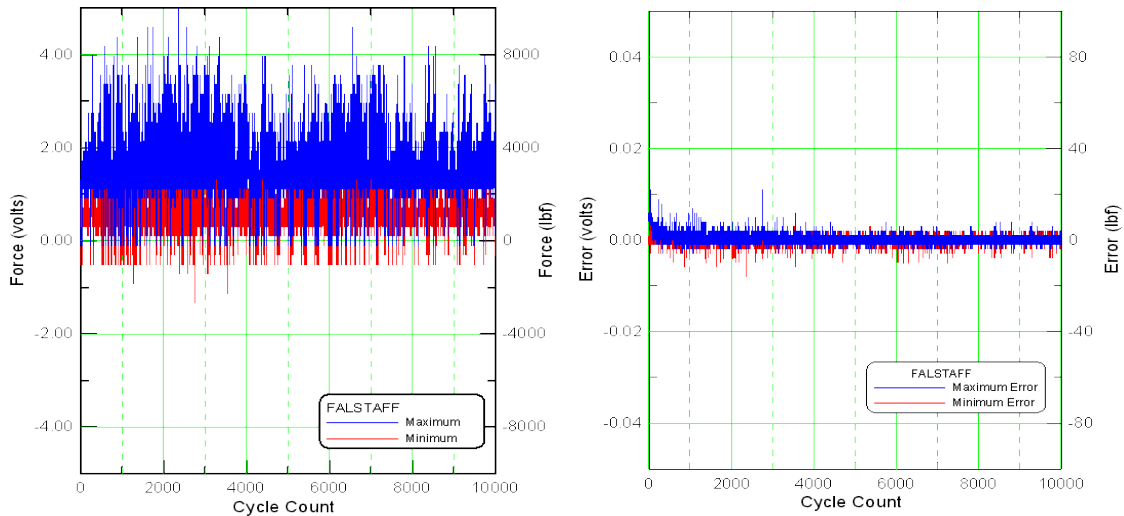
In the **Enhanced Output Format**, engineering units are assigned to the raw voltage and two extra columns of maximum error and minimum error are added to the (.vl2) as follows:

1. Cycle count
2. Maximum target load (lbs, kN)
3. Minimum target load (lbs, kN)
4. Maximum actual load (lbs, kN)
5. Minimum actual load (lbs, kN)
6. Maximum error load (lbs, kN)
7. Minimum error load (lbs, kN)

To use this routine, open the existing .vld file by pressing **Open Validation Log**. Specify the test's **Load Calibration**, the **Minimum Error** for filtering purposes, the **Minimum Cycle Count**, and the **Maximum Cycle Count**. Data are only processed between the minimum and maximum cycle count. Activate the utility by pressing **Start Filtered Validation Log** followed by **Run Analysis**. The .vl2 file is saved automatically.

The following figure illustrates the results of the validation log. On the left is a plot of maximum in minimum endpoint values for the first 10,000 cycles of a FALSTAFF spectrum. On the right are the associated endpoint errors for the maximum and minimum endpoint values. Note that typical loading errors are less than 0.1% of full range (Highest load is 10,000 lbs, typical error is less than 10 lbs).

Validation Log Results



Actual Endpoints

Endpoint Errors

Typical errors less than 0.1% of full range

3.3.2.4 Block File

Block File

Block Generator

Block Advance

Insert

Delete

Back One

Forward One

Receive from DSP

Send to DSP

Read from File

Save to File

Type (should be 1) 1

Full Scale 100.0

Offset 0.0

Index 1

Hold

Break Point

Minimum 10.0

Maximum 50.0

Cycle Count 5

Frequency (Hz) 1.0

Hold Time (sec) 0.0

Crack Length 0.0

Minimum 5 Maximum 50 Block Count 3

Print to Log

I:\Software Development\Current Software Development\FCGR Testing (ADwin-USB)

The software's **Block File** function is a useful feature for automating test sequences where constant amplitude test conditions are required to change as a function of crack length and/or cycle count. A single set of conditions is referred to as a *block* and appears as one line in the block file. Hold periods may also be programmed into the sequence.

Block files are loaded in exactly the same manner as end-point files. The file navigation/editing buttons also operate similarly. Pressing **Block Advance** skips the user to the next block, or line, in the file.

As on the **End Point File** menu, three text boxes contain the header information **Type**, **Full Scale**, and **Offset**. For block files, always set Type to 1 and Offset to 0.

- **Hold:** when activated, the load is held at the block's maximum load level (see the previous section of this manual for a discussion of terms) and cycle count becomes inactive. The length of the hold level is defined in the **Hold Time (sec)** text box. A hold may be programmed as a block anywhere in the sequence.
- **Break Point:** when activated, the block will automatically advance when a target crack length is achieved. The **Cycle Count** must be set to zero to avoid advancing to the next block after the accumulated number of cycles.

Six text boxes allow for editing/examination of the block file:

- **Index:** designates the sequence of the block in the file.
- **Minimum:** the block's minimum endpoint level
- **Maximum:** the block's maximum endpoint level
- **Cycle Count:** the number of cycles in the block. If this value is zero, then cycling will continue in this block until a target crack length (break point) is achieved or the **Block Advance** button is pressed.
- **Frequency (Hz):** the cyclic frequency for the block. This feature allows each block to have a unique cyclic frequency regardless of the settings on the **Full Function Generator**. If this value is zero, then the frequency is defined by the loading rate and frequency parameters specified on the **Full Function Generator**.
- **Crack Length:** this parameters specifies the crack length for advancing to the next block if the **Break Point** mode is enabled.

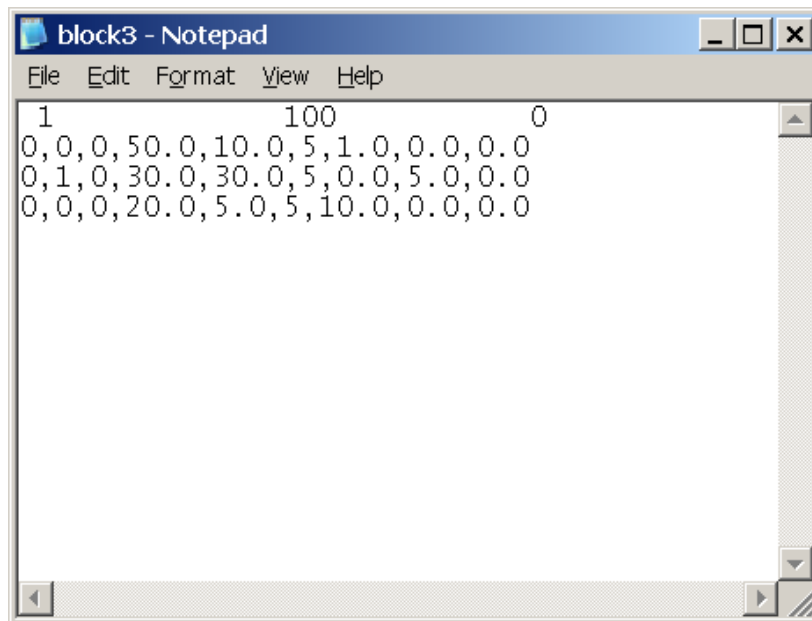
Three display boxes offer at-a-glance information about the block file:

- **Minimum:** maximum load level of the block file
- **Maximum:** minimum load of the block file
- **Block Count:** total number of blocks in the file

Below the 3-item header, the file consists of nine columns, which correspond to the text boxes and input options described above:

1. Index (always 0 in file)
2. Hold (1/on, 0/off)
3. Break Point (1/on, 0/off)
4. Maximum
5. Minimum
6. Number of cycles for block
7. Freq (Hz)
8. Hold time (sec)
9. Crack length

The following file is an example of three blocks. Since full scale is 100, each maximum and minimum level represents the % of full scale. The first block is five cycles between levels of 5% and 50% of full scale at a cyclic frequency of 1.0 Hz; the second block is a five second hold at a level of 30% of full scale; the third block is five cycles between 5% and 20% of full scale at a cyclic frequency of 10.0 Hz.



```
1          100          0
0,0,0,50.0,10.0,5,1.0,0.0,0.0
0,1,0,30.0,30.0,5,0.0,5.0,0.0
0,0,0,20.0,5.0,5,10.0,0.0,0.0
```

Note: As with end-point files, blocks should be constructed to avoid non-turning endpoints. With the block file, there is no utility to filter non-turning endpoints. The correct operation of the block file should be verified in simulation mode.

3.3.2.5 Segment File

Segment File

Segment Generator

Insert
Delete
Reset
Back One
Forward One
Receive from DSP
Send to DSP
Read from File
Save to File

Type (should be 3) 3

Full Scale 100.0

Offset 0.0

Inhibit Count

Carry Insert

Index 1

Endpoint 90.0

Frequency (Hz) 0.3333

Time (sec) 1.5

Total Time 11.50045

Minimum 5

Maximum 90

Segment Count 16

Print to Log

I:\Software Development\Current Software Development\FCGR Testing (ADwin-USB)

The **Segment File** generator allows greater customization of waveform shape than is possible with the **Custom Waveform** option in the **Steady State** application. This is useful if crack length determination is desired during one complete cycle or pass of the spectrum. It is also useful if the shape of the waveform requires a long hold period with a relatively short period of cyclic loading.

Each segment is entered as an endpoint value followed by a frequency (Hz) or time (seconds) to reach the endpoint from the previous endpoint. The frequency would be for a complete cycle of two endpoints. Therefore, the relation between frequency and time (period) is:

$$\text{Frequency} = 1 / (2 * \text{Time})$$

Either the frequency or the time may be entered for each endpoint. Two endpoints (segments) equal one cycle. Segments are converted to cycle count unless the **Inhibit Count** checkbox is checked for a particular segment. This feature allows cycle count to occur once every pass in the segment file if all but two segments are inhibited. In any event, DCPD data acquisition and current switching is synchronized according to the segment count allowing a crack length calculation as often as every four segments.

Endpoints are connected by a straight line or haversine depending on whether a **Sine** or **Triangle** waveform is selected on the **Full Function Generator**.

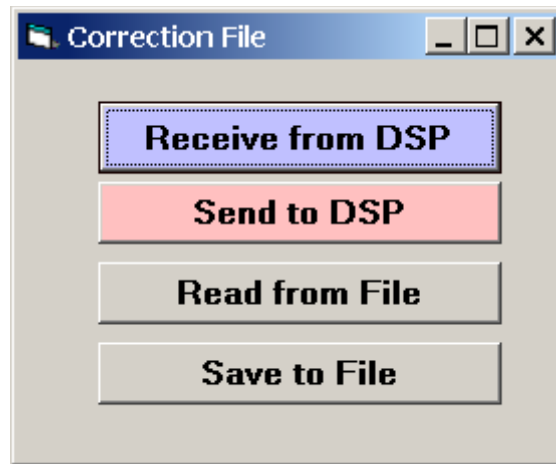
The **Carry Insert** checkbox allows inserting repetitive segments by activating the checkbox and pressing the **Insert** button.

Several text boxes offer at-a-glance information about the segment file. **Total Time** displays the elapse time of the segment file. **Minimum**, **Maximum**, and **Segment Count** refer to the minimum, maximum and total number of segments in the file.

Pressing the **Reset** button sets the total time to zero and goes back to Index 1. The **Reset** button does not alter the segment file but operates the same as the pressing **Back One** button until the beginning of the segment file has been reached.

An even number of segments must be entered in the file. If the file contains an odd number of segments, the last segment will be ignored since the algorithm does not adequately correct for segments that do not follow a peak/valley order. Therefore, the command correction function is automatically disabled when using the segment generator. However, the software still reads the signals and compares the command to the feedback and displays the error on the **FNG**. This can be useful for manually tuning the system. A damage parameter is also calculated.

3.3.3 CORRECTION FILE



When a variable amplitude test is running, the program produces an array of correction parameters unique to each endpoint. The contents of this array may be stored as a **Correction File** and used during subsequent, duplicate tests to enable initial command correction without waiting for the signal to optimize automatically over time, as it does when **Command Control** is activated.

The contents of array maybe stored in a correction file at any time during a test by pressing the **Receive from DSP** and **Save to File** buttons. The file can then be recalled for subsequent tests using the **Read from File** button and the pressing the **Send to DSP** button.

During normal operation, the user will notice that the **Damage Parameter** (on the **Status** display) will approach unity as the test progresses if either **Command Control** has been activated or manual corrections are being made. If a correction file has been created, however, it can be used to circumvent this gradual process during subsequent tests. It is important to note that these corrections are *extremely* sensitive to a wide variety of factors including test frequency, specimen compliance, and machine dynamics. If any of these parameters have changed, the correction file should be cleared on the **Initialize** tab on the **Data Input** form prior to starting a new test. In this case, a new correction file will be generated as the test progresses.

3.4 Display

3.4.1 STATUS: VARIABLE AMPLITUDE

Status - Compliance									
Test ID:		test		I:\Software Development\Current Software					
	Target Load		Command Signal		Actual Load		Clip Gage		
	lb	%	lb	%	lb	%	in	%	
Max	505.3	5.1	505.3	5.1	1322.3	13.2	0.00066	13.2	
Min	1526.8	15.3	1322.5	13.2	709.4	7.1	0.00036	7.1	
a/W	a	Last a	Next a	Count	Time	Date			
0.5101	1.02022	1.02000	1.03000	1.868	4:54:32	3/18/2012			
					0.09	0.00			
Slope #	Slope#	Corr	Kmax	Delta K	da/dN	Damage			
50	49	1.00000	10.73	10.73	5.261E-7	1.0003			
						1.0003			
EvB/P	Count	a	EvB/P	Count	a				
1.2505	1.770	1.02022	1.2506	823	1.02028				

The VA **Status** display is, for the most part, identical to the steady-state version. It is important to note, however, that the values for **Kmax** and **Delta K** both correspond to the stress intensity at maximum load and are thus identical.

An important feature of the VA display is the **Damage [Parameter] (Γ)** at the lower right of the screen. This function estimates the program's success in attaining load peaks, which is important since loading errors have an impact on life prediction. The effect of such errors on crack growth rates should take into account the Paris relationship between the stress intensity range ΔK and the fatigue crack growth rate da/dN as follows:

$$\frac{da}{dN} = C\Delta K^m$$

where

da = crack length increment

dN = cycle count increment

C = intercept

ΔK = cyclic stress intensity

m = power law exponent

Values of m are generally between 2.5 and 4.0. For simplicity, an integer value of 3 has been chosen for the real-time calculation of this parameter. Since for a given crack size, ΔK is proportional to ΔP , we can rationalize that the effect of loading errors on fatigue crack growth rates is also proportional to the ratio of the target and actual ΔP raised to the power of m . The accumulated effect is given as follows [¹]:

$$\Gamma = \frac{\sum (\Delta P_{Actual})^m}{\sum (\Delta P_{Target})^m}$$

Where:

Γ = *damage parameter*

ΔP_{ACTUAL} = actual cyclic load

ΔP_{TARGET} = target cyclic load

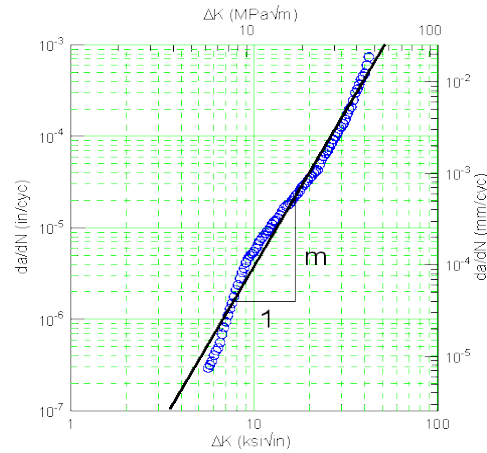
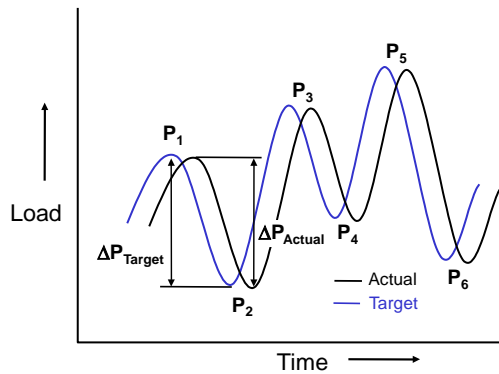
¹ McKeighan, P.C., Fess II, F.E., Petit, M., and Campbell, F.S., 'Quantifying the Magnitude and Effect of Loading Errors during Fatigue Crack Growth Testing Under Constant and Variable Amplitude Loading,' *Applications of Automation Technology in Fatigue and Fracture Testing and Analysis: Fourth Volume, ASTM STP 1411*, A.A. Braun, P.C. McKeighan, A.M. Nicolson and R.D. Lohr, Eds., American Society for Testing and Materials, West Conshohocken, PA, 2000.

The figure below illustrates the calculation of the damage parameter:

Damage parameter (McKeighan, et al) ...

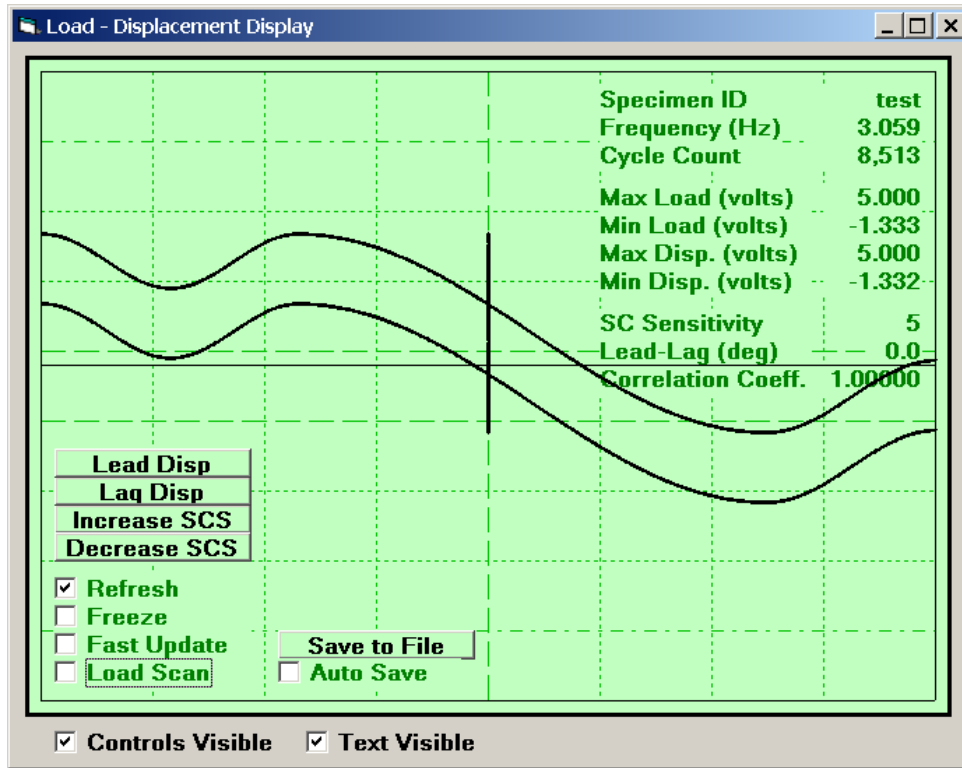
$$\Gamma = \frac{\sum (\Delta P_{Actual})^m}{\sum (\Delta P_{Target})^m}$$

$$\frac{da}{dN} = C \cdot \Delta K^m$$



A damage parameter of 1.000 ± 0.001 is desired. Note that the value displayed in black is updated with crack length while the blue value is updated with each storage interval.

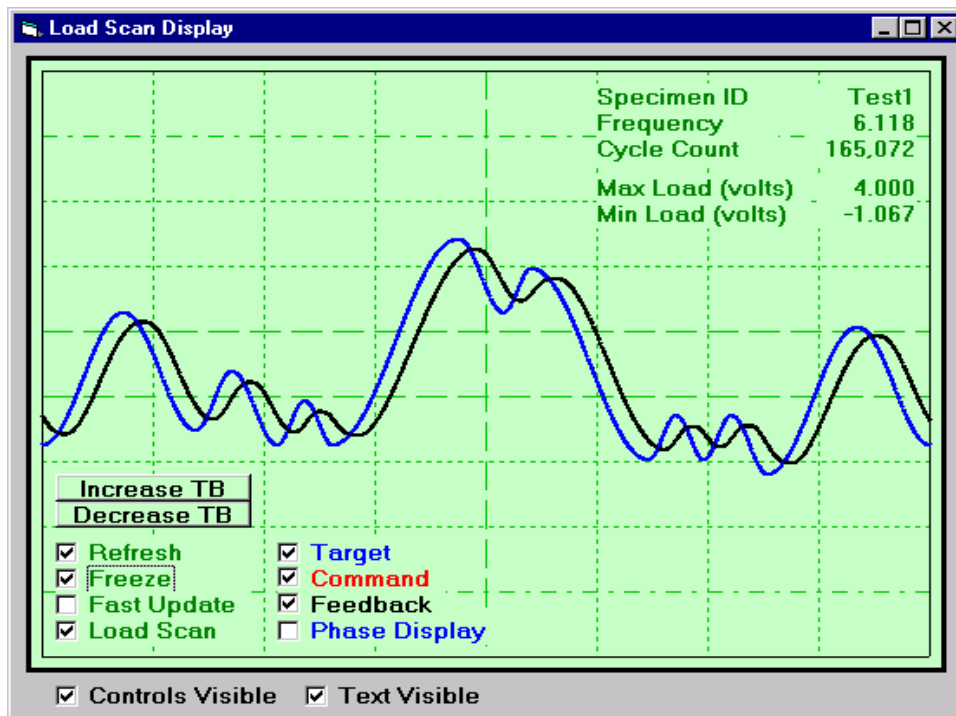
3.4.2 LOAD – DISPLACEMENT DISPLAY



The **Load Displacement** display plots, for two cycles, the load (top curve) and displacement (bottom curve) versus time. Each signal is automatically scaled to the same relative amplitude. The full range voltage of each signal is displayed in the upper right hand corner. The vertical trace in the middle indicates non-linearity between load and displacement. It is important to monitor this non-linearity and to verify that the thin horizontal line displayed is above the non-linearity associated with crack closure. The level of this horizontal line is adjusted by changing the **Minimum Slope Level (%)** on the **Data Acquisition** tab of the **Data Input** form.

3.4.3 LOAD SCAN DISPLAY

A closed-loop servo-hydraulic system has dynamic performance limitations depending on such factors as load capacity, hydraulic flow, and servo-valve response. Under dynamic conditions, a significant lag may occur between the command (target load) and the actual loads (feedback from the load cell). The following figure shows an example of significant lag and under-compensation due to poor PID tuning (for this example, the controller was de-tuned to accentuate the desired effect).

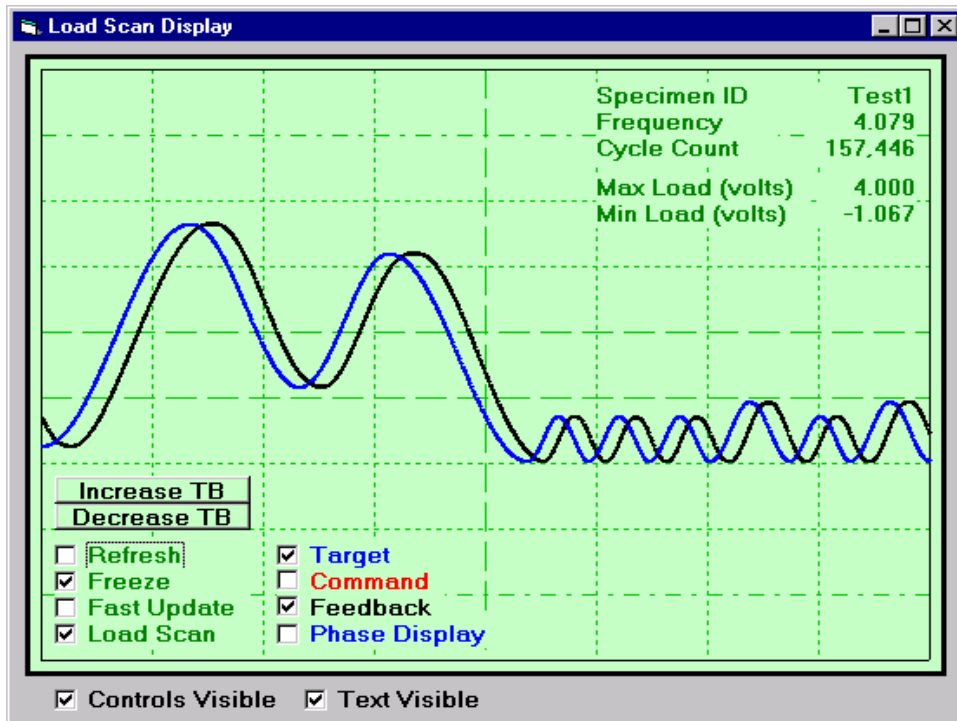


The miss-match between the target (blue) and actual (black) loads can usually be minimized through proper tuning of the control loop. However, under variable amplitude loading it can be very difficult to use simple PID tuning parameters to handle all combinations of endpoints. This is especially true as the cyclic frequency is increased.

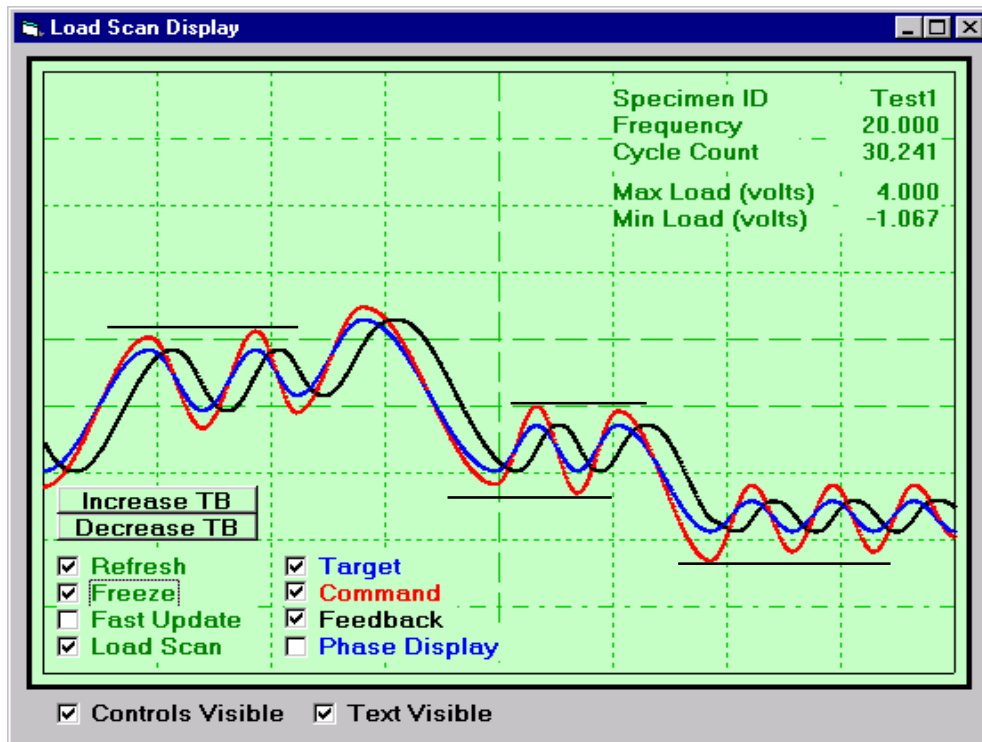
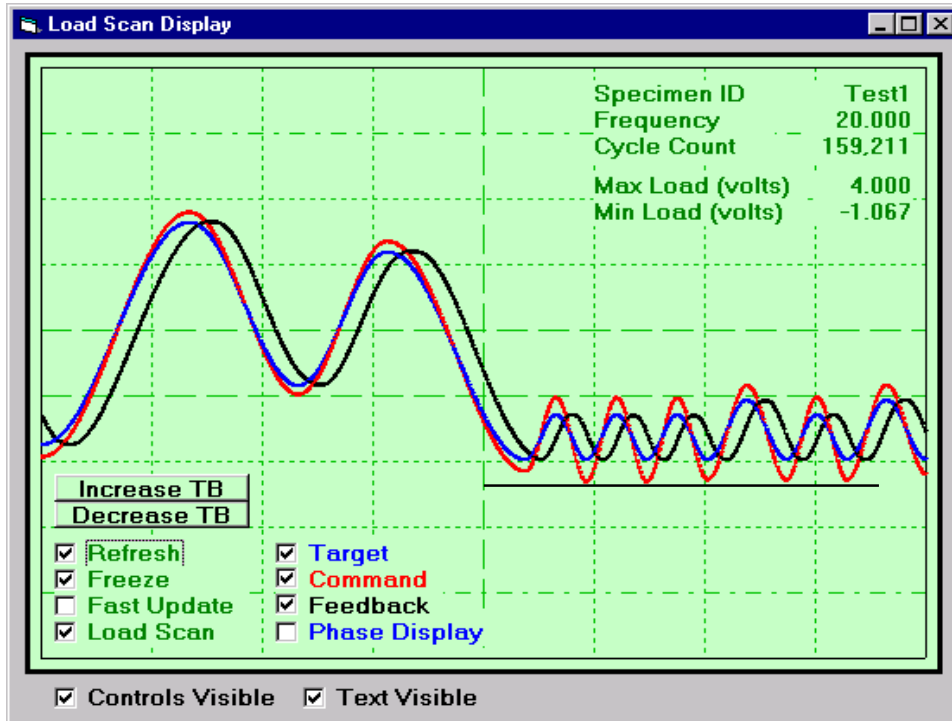
Modification of the command signal based on past performance results in greater load precision than possible with simple PID tuning. In most cases, the target-feedback errors can be reduced to less than 0.1% of range even if the phase lag between command and feedback is quite large (45-90°). The successful implementation of this approach requires that the past performance of the command-feedback relationship be based on the level of the previous, current and next endpoint. Such an approach provides a signature unique to each endpoint. The command-feedback algorithm scans an array to determine if this unique combination of endpoints has occurred in the past. If so, then correction is applied to the next occurrence of this endpoint signature such that the magnitude of this correction is equal to one-half the target-feedback error (if the

Firm Command Control option is selected). The correction of one-half the error may require several passes of a given sequence to reduce the error to less than 0.1% of range. However, a more aggressive correction scheme could lead to oscillations and instability under certain tuning conditions.

The following figure illustrates the use of this command-feedback compensation routine. Note that the target and feedback are out of phase but at the same amplitude.



The following figures show the addition of the command signal (in red) as well as the previously-displayed target and actual load trace. Note that the level of command signal for the same target level differs as a function of the previous and next endpoint.



4.0 Variable Amplitude Analysis Software

This section of the manual is still under construction. Please contact FTA for further information about variable amplitude analysis.

Acknowledgement
Thanks to Pete Blair for the ADwin art