A pulse-echo, immersion method for ultrasonic evaluation of a material which accounts for and eliminates nonlevelness in the equipment set-up and sample thickness variation effects employs a single transducer and automatic scanning and digital imaging to obtain an image of a property of the material, such as pore fraction. The nonlevelness and thickness variation effects are accounted for by pre-scan adjustments of the time window to ensure that the echoes received at each scan point are gated in the center of the window. This information is input into the scan file so that, during the automatic scanning for the material evaluation, each received echo is centered in its time window. A cross-correlation function calculates the velocity at each scan point, which is then proportionalized to a color or grey scale and displayed on a video screen.

15 Claims, 5 Drawing Sheets

(1 of 5 Drawings in Color)
FIG. 1
FIG. 3a

FIG. 3b
FIG. 4
1

PULSE-ECHO ULTRASONIC IMAGING METHOD FOR ELIMINATING SAMPLE THICKNESS VARIATION EFFECTS

ORIGIN OF THE INVENTION

The invention described herein was made by an employee of the United States Government and may be manufactured and used by or for the Government for governmental purposes without payment of any royalties thereon or therefor.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates to ultrasonic evaluation of material properties. More particularly, the invention relates to non-destructive ultrasonic evaluation of materials by measuring velocity using a single transducer pulse-echo immersion system, automatic scanning and digital imaging, which provides a video image of the sample in color or grey scale which is a map of a material property such as porosity fraction.

2. Background Of the Disclosure

Nondestructive evaluation applicable to evaluating properties of materials such as ceramics, metals, plastics and various composites are known to those skilled in the art and include x-radiography, ultrasound or ultrasonic evaluation, and thermal methods. These methods provide an efficient, quasi-quantitative measure of material homogeneity, but often lack the precision necessary for microstructure evaluation of high-performance materials, such as high temperature oxidation resistant ceramics and the like. The development and use of materials for high-performance applications requires detailed, quantitative knowledge of microstructural and compositional variability for defining acceptable levels of variability and for rejecting those materials and processes that yield sample-to-sample and within-sample variations likely to result in unacceptable property (e.g., strength, thermal conductivity, oxidation resistance, resistance to spalling, etc.) variations. Such variability must be precisely characterized either directly in terms of property measurement or indirectly through microstructural characterization where microstructure-property relations have been previously established.

Repeated, uniformly spaced ultrasonic contact measurements have been successful for quantifying and mapping inhomogeneity in various ceramics (e.g., SiC, Al₂O₃, YBa₂Cu₃O₇ and Si₃N₄) and metals in terms of ultrasonic material properties such as reflection coefficient, velocity and attenuation coefficient as mentioned, for example, by Roth, et. al. in Quantitative Mapping of Pore Fraction Variations in Silicon Nitride Using an Ultrasonic Contact Scan Technique, NASA TP 3377 (1993). This publication describes quantitatively characterizing material (e.g., Si₃N₄) microstructure in terms of actual ultrasonic wave parameters. The wave variables include reflection coefficient, attenuation coefficient and velocity. A post-scan interactive data display system is used for comparing ultrasonic properties at different locations within samples and viewing the resultant ultrasonic images. Further refinement of this process is disclosed by Roth, et. al. in PSIDD: A Post-Scan to Interactive Data Display System for Ultrasonic Scans, NASA TM-4545 (1993). This process relates to contact scans and does not disclose how to account for thickness variations in the sample being measured. Piche discloses a single transducer immersion method for evaluating plastic using a technique in which 16 scan points are pulsed for the sample and the results evaluated using regression analysis [L. Piche, Ultrasonic Velocity Measurement for the Determination of Density in Polyethylene, Polymer Eng. & Sci., v. 24, n.17, p. 1538-58 (Dec. 1984). This method does not relate to forming an image of the sample property, nor does it provide an experimental technique that automatically accounts for nonlevelness and thickness variation during a scan procedure required to form an image. Consequently, a need still exists for a method which will permit ultrasonic material evaluation that will account for nonlevelness and thickness variations in the material, require only a single transducer, eliminate problems associated with physical contact between the transducer and sample or buffer rod, and display, on a video screen in grey scale or color, an image of the scanned material which is a map of an internal structural property of the material, such as porosity fraction.

SUMMARY OF THE INVENTION

The invention relates to a method for nondestructive ultrasonic evaluation of materials by measuring velocity using a pulse-echo immersion system with automatic scanning, echo cross-correlation and digital imaging to obtain a grey scale or color image of the sample. The velocity values obtained for each scan point are scaled on a grey or color scale and displayed on a video screen which shows a material property, such as porosity fraction, on the images. Automatic scanning eliminates the need for a separate scanning file. The automatic scanning allows for quantitative and diagnostic measurements of nonlevelness and sample thickness variation fields are accounted for and eliminated by insuring that the echoes at each scan point are first gated and input into a scan parameter file in a computer, so that during the subsequent automatic scanning each received echo is centered in the time window set for it. While it is possible, but not practical to do a manual prescan at each and every scan point needed for a two dimensional video image of the material property being evaluated, many sample thickness variations are in the form of a uniform thickness variation from one edge to another. In this case, the preliminary scans are performed along a single line in the x- and y-directions of the sample to provide slant correction factors. The slant correction factors are input into the scan parameter file so that any wedge-shape variations are taken into account during the automatic scanning for the material evaluation, to insure that each echo received during the automatic scanning is centered within the time window. A single transducer is used in a preferred embodiment of the invention.

In the immersion method of the invention, the material to be evaluated is surrounded by a liquid and positioned on an acoustic reflector which is also immersed in the liquid. An ultrasonic wave of a known frequency is transmitted through the liquid and four separate echoes are recorded and evaluated at each scan point. Each echo is received as an analog waveform which is digitized and stored in a computer. The echoes received, digitized and stored during the sample evaluation scans are the first two successive echoes reflected off the back surface of the sample, the first echo reflected off the front surface of the acoustic reflector in which the received wave has passed through the sample, and the forth of the first echo reflected off the front surface of the reflector with the sample not present, so that the received wave does not pass through the sample. This means that at least two separate scans must be made, with and without the sample present between the transducer and reflector. However, a practical matter it is difficult from both a hardware and software perspective to accomplish this in just two scans and obtain maximum time resolution and thus maximum accuracy. Consequently three or four separate scans are
performed, with three being faster and four being more accurate. The choice is left to the discretion of the practitioner. In the embodiment in which four separate scans are performed during the sample evaluation, the ultrasonic wave goes through both the liquid and the material during the first three scans. For the forth scan the material sample is removed so that the transmitted wave is reflected off the front surface of the reflector without going through the material. Although the order is not important, it is convenient to receive the first echo reflected off the back surface of the material during the first scan and the second successive echo reflected off the back surface of the sample during the second scan. During the third scan in which the transmitted wave goes through both the immersion liquid and the material sample, the first echo reflected off the front surface of the reflector is received. The first echo reflected off the front reflector surface is received during the fourth scan when the sample is not present. This process is repeated at a plurality of scan points sufficient to produce a video image of a material with a property, such as porosity, of the material.

After the scanning is completed, the digitized waveforms are retrieved from the computer and the time delay between the first two successive echoes received from the back surface of the material at each scan point is determined. The time delay between the two different reflections or echoes received off the reflector (with and without the transmitted wave going through the sample) is also determined for each scan point. The wave velocity at each scan point is then calculated from the time delays and the speed of the transmitted wave in the liquid. The velocity values for all of the scan points are scaled to corresponding proportional color or grey scale values which are then displayed on a video screen or cathode ray tube (CRT). Thus, in this embodiment of the invention, four separate scans are made at each scan point to separately receive, as analog waveforms, the first two successive ultrasonic echoes off the back surface of the sample and the first echo off the front surface of the reflector both with and without going through said sample; digitizing and storing the waveforms; retrieving the digitized waveforms; determining the received time delay between the first two successive sample back surface echoes and between the two reflector front surface echoes; calculating the wave velocity at each scan point; scaling the calculated velocities to corresponding proportional color or grey scale values, and displaying the resulting image. The wave velocity at each scan point is calculated from

\[
v = \frac{c}{2t} + 1
\]

wherein \(t\) is the received time delay between the two successive sample back surface echoes, wherein \(At\) is the time delay between the two different echoes received from the reflector with and without going through the sample, and wherein \(c\) is the speed of the transmitted ultrasound wave in the liquid. The embodiment in which three separate scans are made is similar to that in which four separate scans are made, with the difference being that during the first scan the first two successive reflections or echoes off the back surface of the sample are received, digitized and stored. It will be noted that above equation does not include the sample thickness value. This means that the thickness of the sample need not be measured or known.

As set forth above, prior to the two, three or four scans during which the sample is evaluated, nonlevelness and sample thickness variations are accounted for and eliminated by pre-scans to insure that the received reflections or echoes are within their set time windows to provide a complete waveform for evaluation and cross-correlation to accurately obtain the time delay data used in calculating the velocity values. In the case of a sample having a thickness variation in the form of a uniform thickness variation from one edge to another, preliminary scans are performed along a single line in both the x- and y-directions of the sample to provide slant correction factors. The slant correction factors are input into the computer scan parameter file so these variations are taken into account during the automatic scanning for the material evaluation.

**BRIEF DESCRIPTION OF THE DRAWINGS**

The file of this patent contains at least one drawing executed in color. Copies of this patent with color drawing(s) will be provided by the Patent and Trademark Office upon request and payment of the necessary fee.

FIG. 1 schematically illustrates the spatial relationship between the transducer, liquid, material sample, reflector plate and the transmitted and reflected ultrasonic echo waves in the practice of the invention. FIGS. 2(a) and 2(b) graphically illustrate the amplitude and time delay of the received analog ultrasonic echoes reflected off the material sample and reflector. FIGS. 3(a) and 3(b) graphically illustrate respective first and second sample back surface echoes in which the second is inverted with respect to the first. FIG. 4 is a block diagram schematically illustrating the instrumentation used in the invention. FIGS. 5(a) and 5(b) are photographs of video grey scale displays of a thickness based ultrasonic velocity image of a ceramic according to the prior art method and the method of the invention, respectively.

**DETAILED DESCRIPTION**

Referring to FIG. 1, transducer 10 is schematically shown as partially immersed in a liquid 12 which is the immersion fluid. Material sample 14 is shown positioned in the fluid 12 between the transducer 10 and reflector plate 16 in container or tank 18 located on top of x- and y-direction motorized stages 19 and 20. Supports 15 and 15' maintain sample 14 above reflector 16. The transducer 10 and motorized x- and y-direction stages 19 and 20 are electrically connected by means not shown to a pulser-receiver (not shown) and to means (not shown) for moving the x- and y-stages in their respective directions. Similarly, liquid 12 is connected to means not shown for maintaining the temperature of the liquid preferably within at least about ±1.0°F. of the temperature at which the scans are to be run. It is possible to maintain the liquid temperature within ±0.1°F. The better the temperature control, the more accurate the results will be. For example, if the temperature of the immersion liquid is ±1°F. and the liquid is water, ±1.5% in velocity is possible. If the porosity fraction or other property of the material at a particular point is such as to result in a velocity value difference in the sample of 2%, only a 0.5% micro-structural velocity difference might be detected if a 1+1°F. temperature variation is present during the scan. At each scan point an ultrasonic wave 22 of a known frequency is transmitted from transducer 10 through liquid 12 and into material sample 14. Entering material 14 causes part of wave 22 to be reflected (not shown) off the top surface of the sample, with the rest of the wave passing through the material as 23. Part of wave 23 continues through the material and to the top surface of acoustic reflector 16 as 24, is reflected back off the top surface of reflector 16 as 25,
passes back through the sample 14 and returns to the transducer 10 as wave 26. A portion of wave 23 is reflected off the back surface of the sample and returns to the transducer as 27. Part of the wave 23 reflected off the back surface of the material is reflected off the top surface, returns to the back surface, is again reflected back to the top surface and exits as wave 28. Waves 27 and 28 are the first two successive back surface reflected waves used in the method of the invention at each scan point. Not shown is the wave transmitted through the liquid and reflected back to the transducer without going through the material. This wave which is not shown and wave 26 are the two reflector front surface echoes used in the method of the invention. Motorized stages 19 and 20 form part of an automated scanning system which incrementally moves in both the x- and y-directions to obtain an ordered array of points across the entire surface of the material sample. A 20 MHz, broadband transducer was used in the practice of the invention. Broadband transducers emit a broadband frequency content dominated by a center frequency. That is, they are made to emit at a nominal frequency proximate that of the design frequency (e.g., 20 MHz), with a Gaussian fall-off on either side of the nominal center frequency. Thus, a 20 MHz broadband transducer will also emit frequencies slightly above and below the nominal center frequency of 20 MHz.

In the Piche article referred to above, although the two different reflector front surface echoes are captured and recorded, the first front surface echo and the first back surface echo are captured and recorded. This is different from the method of the invention which captures and records the first two successive sample back surface echoes and not the first front surface echo. Further, Piche does not use automatic sample scanning or digital imaging. FIGS. 2(a) and 2(b) graphically illustrate the reflected waveforms received and displayed on the CRT of an oscilloscope as time domain analog waveforms. Turning to FIGS. 2(a) and 2(b), the intensity or strength of the received waveform is displayed as voltage amplitude, which is the ordinate of the graph, and the received time delay as the abscissa. In this representation, B1, B2 and M refer to waves 27, 28 and 26 of FIG. 1, respectively, with M" representing the wave transmitted through the liquid and reflected off the front surface of the reflector without passing through the material sample. The time delay between the first two successive echoes reflected from the back or bottom surface of the material back to the transducer, B1 and B2, is readily obtained, as is the time delay between the two reflections received from the front surface of the reflector, M' and M".

Since the velocity of the ultrasonic wave is faster in denser media than in less dense media, voids, delaminations, porosity and other density variables within the material are obtained as a function of the speed of the wave, which is determined by the time delay between the two successive echoes received which have been reflected off the back of the material, and the time delay between the two different reflections from the front surface of the reflector. As set forth above under SUMMARY, the speed or velocity of the transmitted wave traveling through the material sample is determined according to the simple equation:

\[ v = c \left( \frac{\Delta t}{2\Delta t} + 1 \right) \]

wherein \( \Delta t \) is the received time delay between the two successive material sample back surface echoes, wherein \( \Delta t \) is the time delay between the two different echoes received from the reflector with and without going through said sample, and wherein \( c \) is the speed of the transmitted wave in the liquid. This equation is accurate for a single point measurement. Prior acoustic coupling between the transducer, material and reflector plate. Since the x-, y-direction scans made across the sample surface in the method of the invention can take a significant amount of time compared to that for a single point measurement and since the speed of sound in a liquid is also a function of temperature, the water is maintained at a constant temperature during the scanning. This is readily accomplished simply by using a constant temperature regulating means, such as a constant temperature water circulator, for maintaining the desired temperature constant during the ultrasonic scanning. It is convenient to keep the temperature of the water at about ambient or 68° F. ±1° F. during the scan, although other temperatures may be used if desired, as long as the temperature is maintained within no more than ±1° F. In the case of distilled, deionized water, the wave velocity may be obtained from published tables. However, tap water may be used as long as the velocity in the water is actually measured. The reflector is placed on the bottom of the tank. Other immersion liquids may be used if desired, such as Dow Corning 300 or 700 viscosity fluid.

The reflector is a solid plate of material having an acoustic impedance significantly different from that of the liquid or water. A flat plate of tungsten (e.g., 0.0625 to 0.125" thick) is preferably used, because tungsten has an acoustic impedance almost two orders of magnitude higher than water in units of g/cm²-see. The use of a tungsten plate results in the highest possible reflection amplitude of any solid material for the echoes reflecting off the front surface of the reflector plate. This large difference in acoustic impedance is important when attempting to obtain ultrasonic echoes that have to travel into and through immersion liquid and the sample, bounce off the reflector plate, and travel back through the liquid and sample to the transducer for reception. High frequency ultrasound provides greater time resolution than lower frequencies and is therefore more desirable for greater accuracy of the velocity of the ultrasound through the sample and corresponding velocity image. The higher the frequency, the greater the velocity accuracy. By having the highest reflection amplitude possible, it is possible to use the method of the invention (a) at higher frequencies where attenuation through the sample is greater than if using lower frequencies and (b) with materials that significantly attenuate ultrasound, such as composite materials. By high frequency ultrasound is meant from 1–100 MHz, typically 3–50 MHz and more typically 10–30 MHz.

The material sample is easily positioned over the reflector plate by using spacers on top of the plate and placing the material on top of the spacers. It is important that the spacers have the same height or thickness so that the material is as level as possible. Lucite is available as sheets which are very uniformly thick and it is convenient to use 0.5" cubes of this plastic as spacers. The material sample, such as a plate of silicon nitride ceramic, is placed on the plastic spacers over the tungsten reflector plate prior to scanning.

FIG. 4 schematically illustrates, in block diagram fashion, the basic system and instrumentation used for the scanning
and ultrasonic imaging according to an embodiment used in the practice of the invention. Thus, referring to FIG. 4, the basic instrumentation includes a transducer 59, a pulser-receiver 52, and a programmable waveform digitizer 54 having associated with it a vertical voltage amplifier 56, programmable time base 58, and analog and digital monitors 60 and 62, respectively. Also included are a time delay or synthesizer 64, an image processor 66, an X, Y, Z controller 68, computer or central processing unit (CPU) 70, and video display 72. In the embodiment shown, the computer 70 is connected with the vertical voltage amplifier 56 and time base 58 also serve as respective analog and digital oscilloscopes. The time synthesizer, time base, voltage amplifier and waveform digitizer are all general purpose interface (IEEE-488)bus (hereinafter “GPIOB”) programmed time and distance instrumentation in the example below is a Tektronix 7A16 P Programmable Vertical Voltage Amplifier (voltage base) and a Tektronix 7B90 P Programmable Time Base. The time delay for each scan is determined a priori and input to the computer or central selectable over the range 50 mv-lV/div, is automatically adjusted by the digitizer so that the entire received echo waveform is displayed with the oscilloscope time window. The digitizer digitizes each waveform received into 512 point arrays (at a sampling rate ranging from 0.512–1.024 GHz depending on the time base division setting). Each waveform is acquired 64 times and averaged to obtain a smoother waveform with averaged noise level using a Fortran algorithm included in the scanning program in the Appendix which is also found in the NASA Technical Memorandum 4106 referred to above. The X and Y positional and Z intensity values are accounted for and eliminated by pre-scans to insure that the received reflected or echoes are within their set time windows to provide a complete waveform for evaluation and cross-correlation to accurately obtain the time delay data used in calculating the velocity values. That is, during the nonlevelness and material thickness variation from one edge to another, preliminary scans, the operator notes if the time delay of each echo received at each scan point is such that it is no longer centered within the oscilloscope time window. If a received echo is not centered within the time window on the scope, this is noted and the time window changed for each such echo received until the received echo time domain waveform is completely within the new time window set for it to insure that the complete time domain waveform is captured or gated correctly within the new window. This time delay information at each scan point is inputted into the scan parameter file and recalled during the actual scanning during the material evaluation, to automatically adjust the time delay for the received echoes at each scan point so that each echo received during the scanning is centered within the time window set for it. This is very time consuming to do for each scan point. However, having a thickness variation in the form of a uniform thickness variation from one edge to another, preliminary scans are performed along a single line in both the x- and y-directions of the sample to provide slant correction factors. The slant correction factors are input into the computer scan parameter file so these variations are taken into account during the automatic scanning for the material evaluation. It is important that the echo at each scan point is centered in its time window, because the whole pulse or echo time domain waveform is needed to give the precise time delay between echoes for the cross-correlation which provides the velocity value. In doing this for a wedge shaped sample, prior to the automatic scan, the transducer scans along two straight lines over the sample, once in the x-direction and once in the y-direction to determine the sample thickness. The transducer is very time consuming to do for each scan point. However, having a thickness variation in the form of a uniform thickness variation from one edge to another, preliminary scans are performed along a single line in both the x- and y-directions of the sample to provide slant correction factors.
oscilloscope time window set for it so that the received waveform is displayed with the maximum possible amplitude on the CRT and still have the complete waveform. This permits the maximum time resolution of individual echoes without losing any part of the time domain waveform which appears on the CRT screen as a function of voltage (amplitude) and time, wherein time is the x-axis and voltage or amplitude is the y-axis. It is important and forms an aspect of the invention that the complete waveform or pulse echo be captured or "gated" on the CRT screen in order to perform an accurate cross-correlation later on in the procedure of the process of the invention. The cross-correlation of echoes provides the precise time delay between received echoes or pulses which is required to calculate the velocity or speed of the ultrasound in the material evaluated which, in turn, provides the information to gray or color scale the velocity data into a digitized map of the material density. This slant correction procedure also allows an accurate evaluation to be made without the need for specialized leveling equipment. These x- and y-direction time window corrections are called slant correction factors and they are inputted into the scan parameter file in units of "nsec/cm" where (a) the number of nsec is the time extent from sample end-to-end that is required to keep the specific echo centered and is determined using the (a) time synthesizer to reposition echoes in time and (b) the number of cm is the distance traveled by the transducer for which this slant factor is determined. By way of an illustrative, but nonlimiting example, the first scan point (0,0) along the x-direction may have a B1 echo centered at a time=6.77 usec, while the last scan point (40,0), may have B1 centered at time=7.14 usec. If the x-direction scan line length is 40 mm, the x-direction slant correction factor is obtained from \((7.14 - 6.77)/40 \mu\text{s/cm/mm}\). It should be noted that slant correction factors can be negative as well as positive numbers. The location of the time window during scanning for the material evaluation is automatically adjusted via computer control by using the formula:

\[ W_{DT} = T_{FP}[(X_{SC}(X_{SN})(X_{SD}) + Y_{SN}(Y_{SD})Y_{SD})] \]

wherein \(W_{DT}\) is the correct delay time window at a particular scan location, \(T_{FP}\) is the time delay at the initial scan location. \(X_{SC}\) and \(Y_{SC}\) are the x- and y-direction slant correction factors, \(X_{SN}\) and \(Y_{SN}\) are the scan point numbers in the x- and y-directions, and \(X_{SD}\) and \(Y_{SD}\) are the x- and y-direction scan increments. With many samples it has been found that the slant correction factors turn out to be the same for the B1, B2 echoes and the slant correction factors for the M' are the same. However, for some samples (e.g., thick samples), they may not be the same. In such cases a first x- and y-direction scan is made for the B1 echoes and a second x- and y-direction scan made for the B2 echoes. The same holds for the M' and M'' echoes for which two separate scans are made in the x- and y-directions.

A scan parameter file is input into a computer which contains all of the information necessary to automatically scan the material sample being evaluated. This information includes a predefined and ordered array of scan points over which to run the scan. By way of an illustrative, but nonlimiting example, in an example of the method of the invention in which the material being evaluated was a monolithic ceramic wedge, the scan consisted of a 41 (x-direction) by 81 (y-direction) grid of measurements for a total of 3,200 scan points, with each measurement or scan point separated by 1 mm (x-) and y- scan increment). Information input into the scan parameter file (NOTHICK_ALLSHAPE1.DAT) includes the following:

C **TITLE** NOTHICK ALLSHAPE1.DAT
C **SCAN INCREMENT (uM) IN X-DIRECTION IS: 1000.
C **SCAN INCREMENT (uM) IN Y-DIRECTION IS: 1000.
C **SCAN LENGTH (uM) IN X-DIRECTION IS: 40000.
C **SCAN LENGTH (uM) IN Y-DIRECTION IS: 10000.
C **X-DIRECTION SLANT CORRECTION FACTOR (nsec/cm) FOR B1 & B2 ECHOES IS: -0.0035
C **Y-DIRECTION SLANT CORRECTION FACTOR (nsec/cm) FOR REFLECTOR ECHOES IS: -0.0035
C **Y-DIRECTION SLANT CORRECTION FACTOR (nsec/cm) FOR B1 & B2 ECHOES IS: -0.0075
C **Y-DIRECTION SLANT CORRECTION FACTOR (nsec/cm) FOR REFLECTOR ECHOES IS: 0.0
C **TIME LOCATION (uSEC) OF B1 ECHO AT SCAN ORIGIN IS: 52.83
C **TIME LOCATION (uSEC) OF B2 ECHO AT SCAN ORIGIN IS: 52.31
C **TIME LOCATION (uSEC) OF REFLECTOR ECHO W/SAMPLE PRESENT AT SCAN LOCATION IS: 69.46
C **TIME LOCATION (uSEC) OF REFLECTOR ECHO W/O SAMPLE PRESENT AT SCAN LOCATION IS: 72.48
C **IMMERSION FLUID VELOCITY (cm/ uSEC) IS: 65.0
C **B2 PHASE-INVERTED WRT B1 (Y/N): N
C **M' PHASE INVERTED WRT M (Y/N): N

As set forth above, in the method of the invention, the transducer is activated so that the first front surface echo off the sample, the first two successive back surface echoes, and the first echo off the reflector plate are all seen in the oscilloscope display at the same time by adjusting the time base to the appropriate time per division setting and
adjusting the time synthesizer delay time. Viewing the first front surface echo off the sample enables the operator to know if the back surface echoes are also on the CRT screen. The unfocused transducer is positioned above the sample at a distance determined initially by the natural focal distance. When using an unfocused transducer a good initial starting height is approximately one to two inches above the sample. The reflector plate front surface echo may be low in amplitude compared to the sample back surface echoes, so that the pulse-receiver gain/attenuation or vertical amplifier gain settings may have to be increased to see this echo. It is important not to confuse the echoes off the front surface of the reflector plate with the second set of echoes originating from the front and back surfaces of the sample. The second set of echoes originating from the front and back surface of the sample will always occur at twice the delay time where the first set of these echoes appears. For example, if the first set of echoes begins at 50 msec on the digital oscilloscope, the second set will begin at 100 msec. If using, for example, a three millimeter thick sample placed on 0.5" thick plastic supports on the reflector plate, the first reflector echo will occur at about 20 msec after the time where the first set of echoes originates and thus the reflector echo will be seen at about 70 msec in this illustration. Another way to note the reflector plate echo is to raise and lower the sample while noting the location of the stationary echo corresponding to the stationary reflector plate. It is essential to have reflector plate echoes that will not interfere with the second set of echoes originating from the front and back sample surfaces. Attention is next focused on the first back surface echo from the sample, B1. The echo is centered in the oscilloscope time window to obtaining maximum time resolution by adjusting the time base time per division and the time synthesizer delay. The synthesizer time is recorded and inputted into the scan parameter computer file. This procedure is repeated for the second back surface sample echo, B2, the first front surface echo off the reflector plate with the sample present, M and the first echo off the reflector plate with the sample removed, M'. The next step is to account for and eliminate any nonlevelness in the set-up and also sample thickness variations. This done at each scan point, except that in the case of uniform thickness variations in the sample, the slant correction factors outlined above are outlined and input into the scan parameter file in the computer.

The scanning is then automatically performed through the remainder of the scanning points previously inputted into the scan parameter file in the computer using a program written in Fortran and IEX GPIB to perform the scanning and also to obtain maximum vertical voltage resolution of the received ultrasonic waveforms. Scanning is accomplished through the use of computer controlled x- and y- microcanning tables used to reposition the sample in the x- or y-direction in a 1 mm increment (other increments may be used at the convenience and discretion of the practitioner) for the next measurement. The ultrasonic waveforms received are then digitized (512x512 pixel resolution) at each scan location and stored successively in the scan data file in the computer. Four separate ultrasonic scans are performed at each scan location. As set forth above, the echoes are B1 (first echo off sample back surface, obtained in first scan), B2 (second echo off sample back surface, obtained in second scan), M' (first echo off front surface of reflector plate with the sample present, obtained in third scan), and M (first echo off front surface of reflector plate without the sample present, obtained in fourth scan). Each of the four echoes is obtained in a separate scan to obtain the maximum time resolution for each echo by setting, before each scan during the nonlevelness and material thickness variation procedure, the optimum time per division setting on the oscilloscope time base that allows maximum time resolution. The minimum number of scans for this thickness elimination procedure is two, but the time per division setting for only two scans cannot be obtained in this case as the time per division setting would be fixed for all three echoes obtained in the first of the scans using this scan procedure.

The following is an algorithm of a scanning program which accounts for and eliminates the nonlevelness of the set-up and uniform thickness variation effects of the sample in the resulting ultrasonic image displayed on the video, the code for which is included in the Appendix.

1) Determine the scan lengths and scan increments in the x- and y-directions, time positions of echoes at scan origin, slant correction factors, and immersion fluid velocity.

2) Edit NOTHICK_ALLSHAPE_FILE, which is the scan parameter file, and input information from 1) above.

3) Start scanner fortran program on computer which automatically does the following:

A) Initialize all GPIB instrumentation, which includes the time synthesizer, digitizer, time base, voltage amplifier, Klinger X, Y stages.

B) Perform scan to digitize B1 echoes and store in file
C) Digitize B1 at scan origin
D) Adjust voltage base for echo with maximum amplitude in video/oscilloscope window
Move Klinger tables under transducer in x-direction specified x-direction increment
Time synthesizer moves to delay time position determined by B1, B2, slant correction factors.
This result in echo in video being centered in the Tektronics analog video/oscilloscope display and subsequently digitized and stored.

Time position = T0 + (Sx(Nx/Dx)+Sy(Ny/Dy))

where T0 = correct delay time window at a particular scan location, Sx = x-direction slant correction factor (nsec/μm)
Sy = y-direction slant correction factor (nsec/μm)
Nx = scan point number in x-direction
Ny = scan point number in y-direction
Ix = x-direction scan increment (μm)
Ly = y-direction scan increment (μm)

II) Repeat 1) until one scan line in x-direction is completed.

III) Increment transducer in y-direction specified y-direction increment and repeat I)–II).

IV) Repeat I–III) until y-scan length is traversed and scan is completed.

V) Return Klinger tables to scan origin.

C) Perform scan to digitize B2 echoes and store in file by repeating steps B(I–V)
D) Perform scan to digitize reflector echoes with sample present and store in file by repeating steps B(I–V), but using reflector echo slant correction factor
E) Remove sample. Perform scan to digitize reflector echoes without sample present and store in file by repeating steps B(I–V), but using reflector echo slant correction factor

4) Start velocity calculation Fortran program on computer to produce a file of velocities at each scan location by performing the cross-correlation algorithm.
5,629,865

5) Start image formation Fortran program on computer which results in a file of values between 0 and 255 which scale directly with the velocity values.
6) Start image display program which brings grey scale level image up on video.

Before initiating the scanning procedure, the temperature of the water or other immersion fluid is measured. If the fluid is water, published tables or graphs of temperature and velocity can be used to determine the velocity of the ultrasound in the constant temperature water bath. If the immersion liquid is a liquid other than water, or if a more precise temperature than that available in published graphs and tables is desired, the velocity of the ultrasound in the liquid is determined by recording the times (Tp) where ultrasonic pulses occur for two different vertical positions (Z1 and Z2) of the transducer above the reflector plate. The velocity, V, is then determined from

\[ V = \frac{Z1 - Z2}{T1 - T2} \]

The phase relationships of (a) B1 compared to B2 and also (2) the reflector front surface echo with the sample present (M') compared to that without the sample present (M") are examined. These phase relationships are important for the computation of the velocity image of the scanned sample. The quantity 2\(t\) is obtained by cross-correlating echoes B1 and B2 which is defined as the precise time delay between the B1 and B2 echoes. If B1 and B2 are phase inverted with respect to each other, the time occurrence of the minimum in the cross-correlation function is used to obtain 2\(t\). If M' and M" are phase inverted with respect to each other, the time occurrence of the minimum in the cross-correlation function is used to obtain \(\Delta t\). Otherwise, at the time occurrence of the maximum in the cross-correlation function is used.

FIGS. 3(a) and 3(b) graphically illustrate the case in which B2 is phase inverted with respect to B1. The same holds for M' and M". Phase relationships generally remain the same throughout the scan, unless significant discrete microstructural defects are encountered by the ultrasonic wave.

After the scan has been completed and all the received echoes have been digitized and stored in the scan data file in the computer, they are recalled from the data file to perform the velocity image calculation for each scan location. In performing this cross-correlation, an overlap method is used by the computer based on a cross-correlation program using Fast Fourier transforms published in pages 415 and 416 (Correlation and Autocorrelation Using the FFT) in the book Numerical Recipes—The Art of Scientific Computing, by Press, et. al., 1986 Edition, Cambridge Univ. Press.. The Fortran program used is in the Appendix. Echoes M' and M" are also cross-correlated to obtain \(\Delta t\) where where M' is the echo reflected off the reflector plate front surface with the sample present, M" is the echo reflected off the reflector plate front surface without the sample present, and \(\Delta t\) is the time delay between them. If M' and M" are phase inverted with respect to each other, the time occurrence of the minimum in the cross-correlation function is used to obtain \(\Delta t\). Otherwise the time occurrence of the maximum in the cross-correlation function is used. The velocity, V, at each scan location is then calculated from the equation referred to above. The velocity value for each scan location is sequentially stored in the computer. After the scan is completed the velocity values are scaled on a gray or color scale with a value directly proportional to the velocity values, with the highest and lowest scale values corresponding to the highest and lowest velocity values.

The invention will be further understood with reference to the example below.

EXAMPLE

In this example, a sample of silicon nitride ceramic was evaluated using the thickness based velocity image method disclosed in the NASA Technical Memorandum TP 3377 referred to under Background. This method is based on a velocity, cross-correlation ultrasonic imaging method without the pre-scan to account for and eliminate nonlevelness in the set-up and sample thickness variations. In this method, only the first two sample back surface echoes are captured and evaluated. The silicon nitride ceramic was 3.5 mm thick with a uniform 300 micron thickness gradient. Very coarse time scaling was used so that the B1 and B2 echoes stayed in the time window while the sample thickness changed as the scan proceeded.

The same silicon nitride ceramic sample was also scanned and velocity imaged on a grey scale according to the method of the invention which included the prescans to eliminate set-up and sample thickness variations and which also captured and cross-correlated both the first two successive sample back surface echoes and the two different reflector front surface echoes.

In both cases, the 20 MHz broad band transducer was used, the immersion liquid was water and the back plate was tungsten as set forth above.

FIGS. 5(a) and 5(b) are photographs of video grey scale displays of the thickness based ultrasonic velocity image of a ceramic according to the prior art method, and an image according to the method of the invention which included the prescans, respectively. Referring to FIG. 5(a), it is seen that the top defect is masked due to that part of the sample being thicker than the bottom part. Also, the defect near the bottom is not too discernable and the lower portion is very light due to it being thinner. In marked contrast and as shown in FIG. 5(b), the method of the invention clearly and correctly illustrates the defect areas, including resolution of the upper defect and an overall porosity gradient in the sample. It is believed that this demonstrates the efficacy and improvement to the art of the invention.

It is understood that various other embodiments and modifications in the practice of the invention will be apparent to, and can be readily made by, those skilled in the art without departing from the scope and spirit of the invention described above. Accordingly, it is not intended that the scope of the claims appended hereto be limited to the exact description set forth above, but rather that the claims be construed as encompassing all of the features of patentable novelty which reside in the present invention, including all the features and embodiments which would be treated as equivalents thereof by those skilled in the art to which the invention pertains.
APPENDIX
not thick_SCAN.FOR --- 4 SCAN METHOD

Scan and data acquisition for the modified ultrasonic Immersion scanner

by Don J. Roth

PLEASE USE CLNEW.COM TO COMPILE & LINK

Latest Update 20-sep-94

integer*4 SCANSTEP,INDXSTEP
integer*4 SCANAXIS, INDXAXIS, ZSTEP

integer*4 117
character VOLANS*,1,TScheme*,1,ULANS*
CHARACTER 1,LOOPANS*,1,SHAPEANS*,1,ZIGANS*,1,FLAG*
character BUF*,1,SCHEME*,1
real DELAY(0:4),VOLTSET(0:4),MAXVLF,MAXVAT, VOL (4)
real slant1,slant2,slant3,slant4

integer*2 NSCAN, NINDX,PLACE,ZIGCOUNT,XXX, SCANKNT
integer*2 A(512),NAVE,CHECKP,WWFLAG

INTEGER*2 I15,116,126,127,MAXFS1P,MAXFS1T
INTEGER*2 I36,137,146,147
INTEGER*2 ILOPTHRU_11,LOOPTHRU_S,LLL,IT,IIJ,II
BYTE WRKO(80)

COMMON /BLK/ WRK0,BUFFER
COMMON /BLK/ A, DELAY, VOLTSET, MAI

format( A )
Data CHECKP;0,ZIGCOUNT;1/
Data 16/1/11/1611/17/1/PLACE;/

10010 format( A )
10020 format( I )
10050 format( F )
OPEN(unit=8, file="TXA1", status="NEW")
CALL ENTERPARAM( SCANSTEP,INDXSTEP, NSCAN, NINDX, NAVE, VOLTAGE,
1 , VOLTAGEU, VOLTAGE_MAX, SHAPEANS, LOOPANS, LOOPTHRU_11, LOOPTHRU_S,
1 , ZIGANS, SCHEME, TScheme, UZEROO, FILENAME, I15, SCANMODE, SLANT1, SLANT2
1 , SLANT3, SLANT4)

4328 SCANAXIS + 1
INDXAXIS = 2
ZSTEP = 10

CALL STRTGPB
DO (1800 IADDR = 5,1,-1)
1800 CALL INITINSTR(IADDR )

Setup of 7912: begin with V/D = 5, then automatically find
the best intensity and Digitize Defects

NOTE: MAY NEED TO CHANGE THIS V/D TO START

2000 CALL TEKGL
C************************************************************************
10 FORMAT (F)
READ (10,65403) DELAY1
READ (10,65403) DELAY2
READ (10,65403) DELAY3
READ (10,65403) DELAY4
READ (10,65402) VOLANS
READ (10,65403) VOL(1)
READ (10,65403) VOL(2)
READ (10,65403) VOL(3)
READ (10,65402) VOLANS
READ (10,65403) VOL(4)
65402 FORMAT (A)
65403 FORMAT (F)
S1 FORMAT(A32)
S2 FORMAT(A2)
S3 FORMAT(A4)
S4 FORMAT(A2)
TYPE *,'DELAY TIMES (B1,B2,R5,RNS)'
TYPE *,'DELAY1= DELAY1
TYPE *,'DELAY2= DELAY2
TYPE *,'DELAY3= DELAY3
TYPE *,'DELAY4= DELAY4
TYPE **,'CLOSE (10)
DELAY(1)=DELAY1/(1.10.**6.)
DELAY(2)=DELAY2/(1.10.**6.)
DELAY(3)=DELAY3/(1.10.**6.)
DELAY(4)=DELAY4/(1.10.**6.)
TYPE **,'TYPE *,'DELAY
TYPE **,'**
C***************************************************************
CALL INITINSTR(0)
CALL SETVOLTDIV(0.5)
CALL TEKGTL
CALL GETMAI(MA1)
CALL TEKGTL
CALL PUTTIME(DELAY(1))
CALL TEKGTL
CALL GETBESTMAI(MA1)
CALL DIGDEF
C
C !Preliminary Digitalizations to get Watch level and Timeset (Time/Div)
C
CALL TEKGTL
CALL AUTOSETVOLTS(MA1, VOL.TSET(1),NUMINT)
C
TYPE **,'AUTOSET VOL.TS'
IF( VOL.TSET(1).EQ. 999.) THEN
  TYPE *,DIGITIZER IS SCREWED UP
  GOTO 2000
ENDIF
CALL GETGRID( TIMESET, X )
CALL GETSA( NAVE, MAI, A )
CALL TEKGTL

TYPE *, 'Timeset=', TIMESET, ' Delay=', DELAY(I),
  'Volts=', VOLTSET(1)

WRITE(16,rec=116,fmt=53) TIMESET
116=416+1

C
C Get WATCH - standard ground level
C
  TYPE *, ' 
  TYPE *, ' 
  TYPE *, ' 
  TYPE *, ' 
  W = A(I)/NAVE
  WATCH = W - 100.

WRITE(8,12030) WATCH
  TYPE *, ' 
  TYPE *, ' 
  TYPE *, ' 
  TYPE *, ' 

12030 format( 'Minimum acceptable ground level =', F )

C SCANSTEP = 1000*SCANSTEP
C INDEXSTEP = 1000*INDEXSTEP
CALL SETXYZ( SCANSTEP, INDEXSTEP, ZSTEP )

WRITE(5,12140)

12140 format( '### SCANNING ###' )

IDIR=1

WWFLAG=0

do 2600 notht.ee.2)then
  !nothickness mod
if (notht.ee.2)then
  CALL MOVEORGXY( ZIGANS, SCANAXIS, NSCAN, INDXAXIS, NINDX)
  REWIND(12) ! DATHS FILE
  IF (ZIGANS.EQ.'Y') ZIGCOUNT=1
ENDIF

IF (NOTH_LEQ.4) THEN
  type *, ' 
  type *, ' 
  type *, ' 
  type *, ' Finished w/ Sample Scan **********
  type *, ' *** Remove Sample from tray & Hit <RET> to perform water scan
  type *, ' 
  type *, ' 
  Read (5,10020) IO
ENDIF
DO 2400 I=1,NINDX ! Outer loop over index
DO 2200 J=I,NSCAN ! Inner loop over scan
IF (NOTH_1.ge.2)THEN
    READ(12)JJ,II,FLAG,SCANCNT 'READ WHETHER 'H' OR 'S'
    SCAN
    type **,**
    type **,**
    type **,**
    type **,**second time around for jj,ii
    type *,FLAG = ,FLAG
    type *
    type **,**
ENDIF

IF ((SCANMODE.EQ.'P'.OR.SCANMODE.EQ.'L')AND.I.GT.1)GOTO 2400 !MOD FOR LINE POINT SCAN
IF (NOTH_1.EQ.1)THEN
    TYPE *,'I16= ',I16,' I17= ',I17
    ELSEIF (NOTH_1.EQ.1)THEN
        TYPE *,'I46=',I46,' I47= ',I47
        ENDIF
        IF( I.EQ.NSCAN )GOTO 2200 !DON'T PICK UP XDUCER EACH TIME
        IF (LOOPANS.EQ.'N')GOTO 2200
CONTINUE
3429 CONTINUE
IF (NOTH_1.LE.1 AND IT.LT.4)THEN
    TYPE *,I16=' ',I16,' I17=' ,I17
    ELSEIF (NOTH_1.LE.1 AND IT.EQ.4)THEN
        TYPE *,I16=' ',I16,' I17=' ,I17
    ENDIF
    IF (I.LE.NSCAN )GOTO 2200
    IF (LOOPANS.EQ.'N')GOTO 2200 !DON'T PICK UP XDUCER EACH TIME
    IF (NOTH_1.LE.1)THEN
        TYPE *,IT=' ',IT
        IF (NOTH_1.LE.1)THEN
            TYPE *,
IF (NOTH_I.EQ.1) TYPE *, *
IF (NOTH_I.EQ.1) TYPE *, 'VOLANS (WRT "ON SAMPLE")=', VOLANS
IF (NOTH_I.EQ.1) TYPE *, 'VOLANS (WRT "NOT ON SAMPLE")=', VOLANS
IF (NOTH_I.EQ.1) TYPE *, *

C*************** MOD FOR ZIGZAG SCAN ************************************

22001 IF (ZIGANS.EQ.'Y' AND ZIGCOUNT.EQ.0) THEN  ! ZIGZAG SCAN
   IF (SCANMODE.EQ.'P') GOTO 12306  ! FOR POINT MEASURE, DON'T MOVE X-AXIS
   CALL MOVXYZ(SCANAXIS,-1)  ! MOVE X-AXIS BACKWARD
   ELSEIF (ZIGANS.EQ.'Y' AND ZIGCOUNT.EQ.1) THEN
   IF (SCANMODE.EQ.'P') GOTO 12306  ! FOR POINT MEASURE, DON'T MOVE X-AXIS
   CALL MOVXYZ(SCANAXIS,1)  ! MOVE X-AXIS FORWARD
   ENDIF

12306 CONTINUE
   CALL WAIT2(1000)  ! ADD DELAY SO THAT DATA IS NOT TAKEN BEFORE
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   IF (ZIGANS.EQ.'Y') GOTO 12305

C***********************************************************************

24002 CONTINUE
   IF (LOOPANS.EQ.'N') GOTO 2200

12305 CONTINUE  ! NOTTHICK

2200 CONTINUE
   IF (I.EQ.NINDX) GOTO 2400  !
C FOR LINE SCAN OR POINT MEASURE, DON'T MOVE Y-AXIS
   IF (SCANMODE.EQ.'P' OR SCANMODE.EQ.'I') GOTO 2401
   CALL MOVXYZ(INDXAXIS, 1)  ! Move

   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
   CALL WAIT2(1000)
CALL WAIT2(1000)
CALL WAIT2(1000)
CALL WAIT2(1000)
CALL WAIT2(1000)
CALL WAIT2(1000)
CALL WAIT2(1000)
CALL WAIT2(1000)
CALL WAIT2(1000)

NTGO = NINDX-1

IF (ZIGANS.EQ.'Y') THEN
  NORMAL SCAN
  CALL MOVORG(SCANAXIS,NSCAN,VOLTAGE,U.

ELSEIF (ZIGCOUNT.EQ.1) THEN
  ZIGCOUNT=0
  GOTO 15021
ELSEIF(ZIGCOUNT.EQ.0)THEN
  ZIGCOUNT=1
ENDIF

C***********MOD FOR ZIGZAG SCAN**************************C
ELSEIF (ZIGANS.EQ.'Y') THEN
  ZIGCOUNT=0
  GOTO 15021
ELSEIF(ZIGCOUNT.EQ.1)THEN
  ZIGCOUNT=1
ENDIF

C***********MOD FOR ZIGZAG SCAN**************************C
CONTINUE ! NOTHICK

SUBROUTINE TAKEDATA_O(FILENAME,VOLANS,UOLANS,VOL,SLANTS,SLANT2,SLANT4,SCANSTEP,INDXSTEP,ZIGANS,NSCAN)

INTEGER*2 A(5,12), WFLAG,I,LOCX,LOCY,SLANT1,SLANT2,SLANT4,SCANSTEP,INDXSTEP,ZIGANS,NSCAN
INTEGER*4 I,J,K,L,M,N,O,P,Q,R
REAL*8 X,Y,Z,DDV,DDP,DDQ,DDR,DDU,DDV,DDW

C ENDIF
STOP

1300 FORMAT (15,REC=I15,FMT=52)JXXX
13100 FORMAT (13,13,13)
end
real  DELAY(0:4), VOLTSET(0:4), VOL.TAGE(4), VOL(4), SLANT(4)
REAL  SLANT2, SLANT4
character DAY(9), TIM(9), VOLANS(9), VOLANS(1), C(80), FILENAME(94), ZIGANS(9)
common /SBLK/ A, DELAY, VOLTSET, MA!

C DATA I140/
C
if (noth_l.eq.1)then
   OPEN (unit=14, file='[ROTH.MENU]DONSCAN_INTERP.LOG',
       status='NEW', ACCESS='SEQUENTIAL',
       FORM='UNFORMATTED')
   endif
C
51 FORMAT(A22)
52 FORMAT(A22)
53 FORMAT(A4)
54 FORMAT(A2)
WFLAG = 0
10600 format('(*115)

   if (noth_l.eq.1)then
      IX=1
      IY=1
   elseif (noth_l.eq.2) THEN
      IX=2
      IY=2
   elseif (noth_l.eq.3) THEN
      IX=3
      IY=3
   elseif (noth_l.eq.4) THEN
      IX=4
      IY=4
   endif
C
   DO 600 IT = IX, IY
   IF (IT.EQ.1 OR IT.EQ.2) THEN
      SLANTX=SLANT1
      SLANTY=SLANT3
   ELSEIF (IT.EQ.3 OR IT.EQ.4) THEN
      SLANTX=SLANT2
      SLANTY=SLANT4
   ENDIF
C
   **** MODIFICATION FOR ZIGZAG SCAN ******************************************
   IF (ZIGANS.EQ.'Y') THEN
      if (IT.LE.2 OR IT.LE.4 OR IT.LE.6 OR IT.LE.8 OR IT.LE.10
         OR IT.LE.12 OR IT.LE.14 OR IT.LE.16 OR IT.LE.18 OR IT.
         LE.20 OR IT.LE.22 OR IT.LE.24 OR IT.LE.26 OR IT.LE.28 OR.
         IT.LE.30 OR IT.LE.32 OR IT.LE.34 OR IT.LE.36 OR IT.LE.38.
         OR IT.LE.40 OR IT.LE.42 OR IT.LE.44 OR IT.LE.46 OR IT.LE.48.
         OR IT.LE.50 OR IT.LE.52 OR IT.LE.54 OR IT.LE.56 OR.
         IT.LE.58 OR IT.LE.60 OR IT.LE.62 OR IT.LE.64 OR IT.LE.66.
         OR IT.LE.68 OR IT.LE.70 OR IT.LE.72 OR IT.LE.74 OR IT.
         LE.76 OR IT.LE.78 OR IT.LE.80 OR IT.LE.82 OR IT.LE.84 OR.
   ELSE
      if (IT.LE.2 OR IT.LE.4 OR IT.LE.6 OR IT.LE.8 OR IT.
         LE.10 OR IT.LE.12 OR IT.LE.14 OR IT.LE.16 OR IT.LE.18 OR
         IT.LE.20 OR IT.LE.22 OR IT.LE.24 OR IT.LE.26 OR IT.LE.
         28 OR IT.LE.30 OR IT.LE.32 OR IT.LE.34 OR IT.LE.36 OR IT.
         LE.38 OR IT.LE.40 OR IT.LE.42 OR IT.LE.44 OR IT.LE.46 OR IT.
         LE.48 OR IT.LE.50 OR IT.LE.52 OR IT.LE.54 OR IT.LE.56 OR.
         IT.LE.58 OR IT.LE.60 OR IT.LE.62 OR IT.LE.64 OR IT.
         LE.66 OR IT.LE.68 OR IT.LE.70 OR IT.LE.72 OR IT.
         LE.74 OR IT.LE.76 OR IT.LE.78 OR IT.LE.80 OR IT.
         LE.82 OR IT.LE.84 OR.
   ENDIF
C
   **** MODIFICATION FOR ZIGZAG SCAN ******************************************
IF (IT.EQ.1 OR IT.EQ.2) THEN !CORRECT DELAY WINDOW FOR NONLEVELNESS
  DELAY_CORR = DELAY(IT) + (SLANTX * ILOCX * SCANSTEP) + (SLANTYR * ILOCY * INDXSTEP)
ELSEIF (IT.EQ.3 OR IT.EQ.4) THEN
  DELAY_CORR = DELAY(IT) + (SLANTX * ILOCX * SCANSTEP) + (SLANTYR * ILOCY * INDXSTEP)
ENDIF

TYPE *, J = J, DELAY(CORR) = DELAY(CORR)
TYPE *, I
TYPE *, WATCH
CALL PUTTIME(DELAY_CORR) ! Set delay

CONTINUE

IF ((IT.NE.4 AND VOLANS.EQ.'IJ') OR (IT.EQ.-1 AND VOLANS.EQ.'U')) THEN
  !!!!! USER-DEFINED VOLTAGE SETTINGS
  VOLTS = VOL(IT)
  CALL SETVOLTDIV(VOLTS)
  GOTO 1132
ENDIF

CALL AUTOSETVOLTS(MAVOLTS, NUMINT) ! Set V/D

IF ((VOLTS.GT.1.0) OR (VOLTS.LT.0.01)) THEN
  WRITE(5,1131) VOLTS
ENDIF

CALL TEKRESET(MAI)
GOTO 100
ENDIF

CALL GETSA(NAVE,MAI,A)

WRITE(5,1133) WATCH, WATCH

WWRITE(5,1133) WATCH, WATCH

FORMAT(4, 'CURRENT WATCH LEVEL IS: ', F10.5, 'MIN =: ', F10.5)
TYPE *, I

1100 CONTINUE
TYPE *,

IF (IT.NE.4.AND.VOLANS.EQ.'U') OR (IT.EQ.4.AND.VOLANS.EQ.'U') GOTO 56732

C

C IF (VOLANS.EQ.'U') GOTO 56732 ! SKIPI WATCH

IF (WATCH.LT.WATCH) THEN
    WWFLAG = WWFLAG+1
    IF (WWFLAG.EQ.2) THEN
        OPEN(unit=88, file='[ROTH.MF,NU]DONSCAN.LOG',
        status='NEW')
        ENDIF
        WRITE(5,10200)
        CALL TIME(T1M)
        CALL DATE(DAY)
C
        TYPE *, ' ' 
        TYPE *, ' ' 
        TYPE *, ' ' 
        TYPE *, ' ' 
        TYPE *, ' ' 
        TYPE *, ' ' 
        WRITE(88,10201)
        I16=116+1
        WRITE(88,10201)
        format('A.X=', A.X, 'Y=', A.Y, 'MAY LEAD TO BAD PROPERTY VALUE')
        C
        GOTO 100
        ENDIF

C**********************************************************************

56732 CONTINUE

C IF (noth_1.eq.1) then
500 WRITE(16,REC=I16,FMT=53) DELAY_CORR
      I16=I16+1
      WRITE(16,REC=I16,FMT=53) VOLTS
      I16=I16+1
C
    DO 54321 IJ=1,512
       I17SUB=((I17-1)*512)+IJ
       I17=I17+1
    54321 WRITE(17,REC=I17SUB,FMT=54) A(IJ)
       I17=I17+1
  elseif (noth_1.eq.2) then
5500 WRITE(26,REC=I26,FMT=53) DELAY_CORR
      I26=I26+1
      WRITE(26,REC=I26,FMT=53) VOLTS
      I26=I26+1
C
    DO 54329 IJ=1,512
       I2SUB=((I27-1)*512)+IJ
       I27=I27+1
    54329 WRITE(27,REC=I2SUB,FMT=54) A(IJ)
ELSEIF (NOTH_EQ.3) THEN
  WRITE(36,REC=I36,FMT=53) DELAY_CORR
  I36 = I36 + 1
  WRITE(36,REC=I36,FMT=53) VOLTS
  I36 = I36 + 1
  DO 58329 IJ = 1, 512
  I37SUB = ((I37 - 1) * 512) + IJ
  WRITE(37,REC=I37SUB,FMT=54) A(IJ)
  I37 = I37 + 1
  ELSEIF (NOTH_EQ.4) THEN
    WRITE(46,REC=I46,FMT=53) DELAY_CORR
    I46 = I46 + 1
    WRITE(46,REC=I46,FMT=53) VOLTS
    I46 = I46 + 1
    DO 59329 IJ = 1, 512
    I47SUB = ((I47 - 1) * 512) + IJ
    WRITE(47,REC=I47SUB,FMT=54) A(IJ)
    I47 = I47 + 1
  ENDIF
C***********************************************************************C
IF (I.EQ.0.AND.J.EQ.0) GOTO 20000 ! SKIP NEXT STEP (NOISE MEASURE)
C CHECK FOR ERROR IN FS2 VOLTSET
C
C IF (VOLTAGE(1).EQ.01) THEN
  WRITE (14) I,J,' VOLT FOR FS2=',VOLTAGE(1),
  ' #INT.PO.= ',NUMINT
  GOTO 100
  ENDIF
C CHECK FOR ERROR IN B1 VOLTSET
C
C IF (VOLTAGE(1).LT.VOLTAGE(2)) THEN
  WRITE (14) I,J,' VOLT FOR B1=',VOLTAGE(1),
  ' #INT.PO.= ',NUMINT
  GOTO 100
  ENDIF
  FORMAT (A4)
C***********************************************************************C
20000 CONTINUE
RETURN

C

5,629,865
SUBROUTINE TAKEDATA(PLACE, SCHEME, VOLANS, VOL, IZERO0, FILENAME, 
UOLANS, D6, D7, NOTH, L, L36, L37, L46, L47, 
SLANT1, SLANT2, SLANT3, SLANT4, SCANTSTEP, INDXSTEP, ZIANS, NSCAN, 
RCMAX, MAXFSIP, MAXVALF, RF1, RF2, MAXFSIP1, MAXVALT, RT1, RT2, J16, 
H7, NAVE, WATCH, J1, J15, IT)

INTEGER*2 A1(512), D6, D7, ILOCX, NSCAN, WFLAG, J16, MAXFSIP, PLACE 
INTEGER*2 WATCH_COUNT, J17, ILOCY, D6, I46 
INTEGER*2 MAXFSIP, NAVE, A(512), IT, SCANKNT 
INTEGER*4 I17SUB, I17STAT, D7SUB, D7STAT, J17, J147 
REAL*4 DELAY(0:4), VOIL, VOLSET(0:4), VOLAGE(4), SLANT, SLANT1, SLANT2, F1(512) 
COMPLEX Cf1(1024), CSPEC(1024) 
character SCHEME*, FLAG*, FILENAME*, UOLANS*, ZICANS* 
CHARACTER VOLANS*, TSCHEME* 
COMMON /SBLK/, A, DELAY, VOL, VOLSET, MAI

C DATA 1/4,0,1/4F1/49,
C IF (NOTH, L, E.4, 1) THEN 
& OPEN( un4, tile=FILENAME//'I IS'. status=NEW, ACCESS=SEQUENTIAL,
+ FORM=UNFORMATTED ) 
& ENDF
C FORMAT(A32)
51 FORMAT(A2)
52 FORMAT(A2)
53 FORMAT(A4)
54 FORMAT(A2)
C WATCH_COUNT=0
10600 format (*,115 )

IF (NOTH, L, E.4, 1 AND PLACE, E.4, 1) THEN 
& OPEN( unit=12, tile=FILENAME//'HIS', status=NEW,
+ FORM=UNFORMATTED, ORGANIZATION=SEQUENTIAL ) 
& SCANKNT=0 
& ENDF
C IF (noth, L, E.4, 1) THEN 
& IX=1 
& IY=1 
C ELSEIF (noth, L, E.4, 2) THEN 
& IX=2 
& IY=2 
C ELSEIF (noth, L, E.4, 3) THEN 
& IX=3 
& IY=3 
C ELSEIF (noth, L, E.4, 4) THEN 
& IX=4 
& IY=4 
& ENDF

12
C ****** MODIFICATION FOR ZIGZAG SCAN ********************

IF (ZIGANS.EQ.'Y') THEN
      .OR.I.EQ.40.OR.I.EQ.42.OR.I.EQ.44.OR.I.EQ.46.OR.I.EQ.48.
      .OR.I.EQ.50.OR.I.EQ.52.OR.I.EQ.54.OR.I.EQ.56.OR.
      .OR.I.EQ.58.OR.I.EQ.60.OR.I.EQ.62.OR.I.EQ.64.OR.I.EQ.66.
      .OR.I.EQ.68.OR.I.EQ.70.OR.I.EQ.72.OR.I.EQ.74.OR.I.
      .OR.I.EQ.76.OR.I.EQ.78.OR.I.EQ.80.OR.I.EQ.82.OR.I.EQ.84.OR.
      .OR.I.EQ.86.OR.I.EQ.88.OR.I.EQ.90.OR.I.EQ.92.OR.I.
      .OR.I.EQ.94.OR.I.EQ.96.OR.I.EQ.98.OR.I.EQ.100) THEN
    ILOCX=NSCAN-I FOR ZIGZAG SCAN, REVERSE BACK FOR DISPLAY
    ELSE
      ILOCX=J-1
    ENDIF
  ELSEIF (ZIGANS.EQ.'N') THEN
    !George Wood addition
    ILOCX=J-1
    ELSE
      ILOCX=J-1
ENDIF

IF (IT.EQ.1.OR.IT.EQ.2) THEN
  SLANTX=SLANT1
  SLANTY=SLANT3
ELSEIF (IT.EQ.3.OR.IT.EQ.4) THEN
  SLANTXR=SLANT2
  SLANTYR=SLANT4
ENDIF

IF (NOTH-I.EQ.1)'I'YPE*:
  IF (NOTH-I.EQ.1) TYPE*,IT=IT
  IF (NOTH-I.EQ.1) TYPE*,PLACE=PLACE
  IF (NOTH-I.EQ.1) TYPE*,SCHEME=SCHEME
  IF (NOTH-I.EQ.1) TYPE*:
  IF (IT.EQ.1.OR.IT.EQ.2) THEN 
    TYPE*:
    TYPE*:
    TYPE*:
    TYPE*:
    TYPE*:
ENDIF
IF (PLACE.GE.1.AND.SCHEME.EQ.'B'.AND.NOTH-I.EQ.1) THEN
   HIT=2 ! CORRECT DELAY WINDOW FOR NONLEVELNESS
   DELAY_CORR=DELAY(2)+(SLANTX*ILOCX*SCANSTEP)+(SLANTY*ILOCY*INDXSTEP)
   CALL PUTTIME(DELAY_CORR) ! Set delay FOR B2

!!! USER-DEFINE VOLTAGE SETTINGS?

IF (VOLANS.EQ.'U') THEN
   VOLTS=VOL(I)
   CALL SETVOLTDIV(VOLTS)
ENDIF
IF (VOLANS.EQ.'A') CALL AUTOSETVOLTS( MAI,VOLTS,NUMINT ) Set V/D
GOTO 1132
ENDIF

C KEEP TRACK OF POSITION WHERE DATA IS ACTUALLY TAKEN WITH A COUNTER
IF (IT.EQ.1.AND.PLACE.GE.1.AND.SCHEME.EQ.'X') SCANKNT=SCANKNT-I
IF (NOTH-I.EQ.1) TYPE*,'x=',J,' y=',I,' SCANKNT=',SCANKNT
   CALL PUTTIME( DELAY-CORK ) ! Set delay AS USUAL
CONTINUE

IF ((VOLANS.EQ.'U'.AND.(IT.IF.IQ.I OR IT.EQ.2 OR IT.EQ.3)).OR.
   (VOLANS.EQ.'U'.AND.IT.EQ.4))) THEN !!! USER-DEFINE VOLTAGE SETTINGS
   VOLTS=VOL(IT)
   CALL SETVOLTDIV(VOLTS)
   GOTO 1132
ENDIF

CALL AUTOSETVOLTS( MAI,VOLTS,NUMINT ) ! Set V/D

IF (VOLTS.GT.10.0).OR.(VOLTS.LT.-10.0)) THEN
   WRITE(5,1131)VOLTS
   CALL TEKRESET(MAI)
   GOTO 100
ENDIF

CALL GETSA@JAVE,MAI.A) ! Get waveform
   WATCH1=A/(NAVE) ! Make sure its acceptable
   TYPE*,' '
   TYPE*,' '
   TYPE*,' '
   WRITE(5,1133)WATCH1,WATCH

IF (VOLANS.EQ.U.OR.VOLANS.EQ.2) GOTO 56732 !! SKIP WATCH
IF (WATCH.LT.WATCH ) THEN
  WFLAG = 1
  WRITE( 5,10200 )
  FLAG='H' !H = SAMPLE HOLDER
  TYPE *,FLAG=FLAG
ENDIF
C
C COUNTER FOR BAD WATCH LEVEL, IE. ON WHICH WAVEFORM (B1 OR B2) DOES IT OCCUR
C
C IF (SCHEME.EQ.'X' AND IT.EQ.1) WATCH_COUNT = 1
C IF (SCHEME.EQ.'X' AND IT.EQ.2) WATCH_COUNT = 2
C TYPE *,WATCH_COUNT=WATCH_COUNT
GOTO 20000
10201 format( 'WAVEFORM BELOW "WATCH", GOING TO NEXT SCAN POINT')
10200 format('WAVEFORM BELOW "WATCH", GOING TO NEXT SCAN POINT')
C GOTO 100
ENDIF
C
C*******************************************************************

55732 CONTINUE
C*******************************************************************

IF (NOTH-I.EQ.1) TYPE *, '
IF (NOTH-I.EQ.1) TYPE *, PLACE '-', PLACE
IF (NOTH-I.EQ.1) TYPE *, SCHEME='SCHEME
IF (NOTH-I.EQ.1) TYPE *, TSCHEME='TSCHEME
IF (NOTH-I.EQ.1) TYPE *, ZEROO='ZEROO
IF (NOTH-I.EQ.1) TYPE *, IIIT='IIIT
IF (NOTH-I.EQ.1) TYPE *
IF (PLACE.GE.1 .AND. SCHEME.EQ.'R' .AND. IIIT.EQ.2 ) THEN
  AA = 0.
  A1(I)=A(I)
  A1(I)=A(I) - AAVE
  AA=AA + A(I)
  ! REAL WAVE = Waveform in volts
  ZERO = AA/512.
  IF (NOTH-I.EQ.1) TYPE *,ZEROO='ZEROO
  FF=FF
  DO 599 III=1,512
  FI(III)= (REAL(A1(I))-ZERO)*10./512.
  ZEROO=FF/512.
  IF (NOTH-I.EQ.1) TYPE *, ZEROO='ZEROO
  IF (NOTH-I.EQ.1) TYPE *,
C*******************************************************************

599 FF=FF
  ZEROO=FF/512.
  IF (NOTH-I.EQ.1) TYPE *,
  IF (NOTH-I.EQ.1) TYPE *
C*******************************************************************

C SET UZEROO = 2*(FIRST "NOISE" LEVEL DETECTED <> ASSUME WE ARE STARTING
C ON SAMPLE HOLDER)
C IF (TSCHEME.EQ.'A' AND PLACE.EQ.1) THEN
  UZEROO=2*ZEROO
  IF (NOTH-I.EQ.1) TYPE *
  IF (NOTH-I.EQ.1) TYPE *
  IF (NOTH-I.EQ.1) TYPE *
  AUTO THRESHOLD VOLTAGE NOISE LEVEL = UZEROO
  IF (NOTH-I.EQ.1) TYPE *,

15
ELSEIF (TSCHEME.EQ.'M' .AND. PLACE.EQ.1) THEN
IF (NOTH.LE.1) TYPE*' - 
IF (NOTH.EQ.1) TYPE '*' MANUAL设定 THRESHOLD VOLTAGE NOISE LEVEL = UZERO0
IF (NOTH.EQ.1) TYPE '*' ENDIF

C *********************************************************
C IF (ZEROO.LT.UZERO0) THEN !!! < FX. 0.02 VOLTS - ON SAMPLE HOLDER
C 56120 FLAG='H' HI = SAMPLE HOLDER
C IF (NOTH.LE.1) TYPE '*' FLAG = FLAG
GOTO 20000
ELSEIF (ZEROO.GE.UZERO0) THEN !!!> EX. 0.02 VOLTS - ON SAMPLE
C 56121 FLAG='S' IS = SAMPLE
C IF (NOTH.LE.1) TYPE '*' FLAG = FLAG
SCHEME='X'
GOTO 10386 ! START TAKEDATA LOOP AGAIN
ENDIF
ENDIF
IF (NOTH.LE.1) THEN
500 WRITE(16,REC=116,FMT=53) DELAY_CORR
116=116+1
WRITE(16,REC=116,FMT=53) VOLTS
116=116+1
C
DO 54321 IJI=1,512
117SUB=((I17-1)*512)+1
54321 WRITE(17,REC=117SUB,FMT=54) A(IJI)
117=117+1
ELSEIF (NOTH.LE.2) THEN
5500 WRITE(26,REC=126,FMT=53) DELAY_CORR
126=126+1
WRITE(26,REC=126,FMT=53) VOLTS
126=126+1
C
DO 54329 IJI=1,512
127SUB=((127-1)*512)+1
54329 WRITE(27,REC=127SUB,FMT=54) A(IJI)
127=127+1
ELSEIF (NOTH.LE.3) THEN
5500 WRITE(36,REC=136,FMT=53) DELAY_CORR
136=136+1
WRITE(36,REC=136,FMT=53) VOLTS
136=136+1
C
DO 58329 IJI=1,512
137SUB=((137-1)*512)+1
58329 WRITE(37,REC=137SUB,FMT=54) A(IJI)
137=137+1
ELSEIF (NOTH.LE.4) THEN


5900 WRITE(46,REC=146,FMT='33)DELAY_CORR
      146=146+1
WRITE(46,REC=146,FMT='33)VOLTS
      146=146+1
C
      DO 59029 JJ=1,512
      14751B=(147-1)*512+1
5929 WRITE(47,REC=147SUB,FMT='54) A(JJ)
      147=147+1

endif
C
C WRITE( 8,10600 )A(200)
C**********************************************************************
C IF (IT.EQ.1)VOLTAGE(1)=VOLTS
C IF (IT.EQ.2)VOLTAGE(2)=VOLTS
C IF (IT.EQ.3)VOLTAGE(3)=VOLTS
C IF (IT.EQ.4)VOLTAGE(4)=VOLTS
C**********************************************************************
C CONTINUE
C**********************************************************************
C FORMAT (A4)
C**********************************************************************
C 20000 IF (WFLAG.EQ.1) THEN
      CLOSE(8)
      OPEN ( unit=8, file="T/A1", status='OLD' )
      ENDMF
      IF (SCHEME.EQ.'X') SCHEME='B'
      !RESET BACK TO 'W' FOR NEXT SCAN POINT
C
C IF (LNE.6.AND.J.NE.0) THEN
C
C IRESET COUNTERS TO EXCLUDE DATA GATHERED ON POINT
C WHERE BAD WATCH LEVEL WAS FOUND
C
C IF (WATCH_COUNT.EQ.1) THEN !B1 HAD BAD WATCH LEVEL
C SCANKNT=SCANKNT-1
C 116=116-2
C 117=117-1
C ELSEIF (WATCH_COUNT.EQ.2) THEN !B2 HAD BAD WATCH LEVEL.
C SCANKNT=SCANKNT-1
C 116=116-4
C 117=117-2
C ENDIF
C
if (notl-l.eq.1) WRITE(12)J,I,FI,AG,SCANKNT !NOTE LOCATION WITH FLAG & LEAVE SUBR.
IF (NOTH-1.EQ.1)TYPE *
IF (NOTH-1.EQ.1)TYPE *
IF (NOTH-1.EQ.1)TYPE *
if (notl-l.eq.1) WRITE(I2)J,I,FLAG,SCANKNT !NOTE LOCATION WITH FLAG & LEAVE SUBR.
IF (NOTH.LEQ.1) TYPE *,' ' IF (NOTH.LEQ.1) TYPE *,' ' IF (NOTH.LEQ.1) TYPE *,' ' C ENDIF

C WATCH_COUNT=0 RETURN

cend

C
SUBROUTINE PRESSURE(VOLTAGEL, VOLTAGEU)

C
C**** PRESSURE=20*VOLTA GE **** (PLOWER.LE.VOLTA GE.GE.PUPPER)
C
C Adjust Z axis to get good pressure (PSI)
C <<< 11 = DOWN >>>
C
C PUPPER = .85 ! Upper Pressure = 17
C PLOWER = .75 ! Lower Pressure = 15
C PUPPER = .3 ! UPPER PRESSURE = 6
C PLOWER = .2 ! LOWER PRESSURE = 4
C PUPPER = .5 ! UPPER PRESSURE = 10
C PLOWER = .4 ! LOWER PRESSURE = 8
C PUPPER = .6 ! UPPER PRESSURE = 12
C PLOWER = .7 ! LOWER PRESSURE = 14
C PUPPER = .8 ! UPPER PRESSURE = 16
C PLOWER = .6 ! LOWER PRESSURE = 12
C PUPPER = .61 ! UPPER PRESSURE = 12.2
C PLOWER = VOLTAGEL
C PUPPER = VOLTAGEU

200 CALL GETFLUKE(P)
   IF (P.GE.PLOWER .AND. P.LE.PUPPER) GOTO 900
   IF (P.LT.PLOWER) CALL MOVXYZ(3, I)
   IF (P.GT.PUPPER) CALL MOVXYZ(3, -I)
   GOTO 200

900 RETURN
   end
SUBROUTINE PRESSURE_MAX(VOLTAGE_MAX, LLL, FILENAME,J,I,I15, CHECKP)

C CHARACTER FILENAME*34
INTEGER*2 CHECKP,LLL,LINCR,J,JXXX,I15,ITTT

52 FORMAT(A2)
P_MAX=VOLTAGE_MAX

200 CALL GETFLUKE(P)
IF (P.GT.P_MAX) THEN
CHECKP=CHECKP+1
IF (CHECKP.EQ.1)LINCR=L1,3
IF (CHECKP.EQ.2)LINCR=L1,2
IF (CHECKP.EQ.3)LINCR=1000
DO 2900 ITTT=1,LINCR

2900 CALL MOVXYZ(3,-1) 13 IS Z-AXIS, MOVE Z-AXIS UP
IF (CHECKP.EQ.3) THEN
JXXX=J+1 11 IS Y-LOC; GO BACK TO PREVIOUS ROW
WRITE(I5,REC=-15,FMT=52)JXXX
115-1(I5+1)
STOP
ENDIF
ENDDO
ENDDO

900 RETURN
end
C
SUBROUTINE MOVORG( SCANAXIS, NSTEPS, VOLTAGEL, VOLTAGEU, LOOPTHRU_H)
C
C Run the thing back to the SCANAXIS origin
C
  integer*4 SCANAXIS
  integer*2 NSTEPS,LOPTHRU_H
C
DO 1000 K=1,LOPTHRU_H
C 1000 CALL MOVXYZ(3,-1) ! Move up
DO 1200 K=1,NSTEPS-I
1200 CALL MOVXYZ( SCANAXIS, -1 ) ! Move back
C
DO 1400 K=1,LOPTHRU_H
C 1400 CALL MOVXYZ(3,1) ! Move down
C
CALL PRESSURE(VOLTAGEL, VOLTAGEU) ! np
C CALL MOVXYZ( SCANAXIS, 1 ) ! Move back & forth
C
RETURN
end

SUBROUTINE MOVORGXY( ZIGANS, SCANAXIS, NSCAN, INDXAXIS, NINDX)
C
C Run the thing back to the SCAN origin
C
CHARACTER ZIGANS*1
integer*4 SCANAXIS, INDXAXIS
integer*2 NSCAN, NINDX, 1

IF (ZIGANS.EQ.'Y') THEN

I=NINDX

IF (I.EQ.1 .OR. I.EQ.3 .OR. I.EQ.5 .OR. I.EQ.7 .OR. I.EQ.9
1 .OR. I.EQ.49 .OR. I.EQ.51 .OR. I.EQ.53 .OR. I.EQ.55 .OR.
1 .EQ.75 .OR. I.EQ.77 .OR. I.EQ.79 .OR. I.EQ.81 .OR. I.EQ.83 .OR.
1 .EQ.85 .OR. I.EQ.87 .OR. I.EQ.89 .OR. I.EQ.91 .OR.
1 .EQ.93 .OR. I.EQ.95 .OR. I.EQ.97 .OR. I.EQ.99) THEN

DO 1200 K=1,NSCAN-I
1200 CALL MOVXYZ( SCANAXIS, -1 ) ! Move back X
ENDIF
ENDIF

DO 1400 K=1,NINDX-I
1400 CALL MOVXYZ(INDXAXIS, -1) ! Move back Y
RETURN

end

c

59

5,629,865

60
SUBROUTINE ENTERPARAM( SCANSTEP,INDXSTEP,NSCAN,NINDEX,NAVE,VOLTAGE,U,S, \
ZIGANS,SCHEME,TSCHEN,IEROOG,FILENAME,SCANMODE,SLANT1,SLANT2, \
FILE,$2,SCANMODE,SLANT)
  integer*4 SCANSTEP,INDXSTEP \
  real SCANDIST,INDXDIST,PU,PL,SLANT,SLANT1,SLANT2  \
  real SLANT3,SLANT4,RSLANT1,RSLANT2  \
  integer*2 LOOPTHRU_H,LOOPTHRU_S,ISLANT1,ISLANT2  \
  integer*2 NSCAN,NINDEX,INDXDIST,16,115  \
  integer*2 FREQ,NAVE,ISLANT3,ISLANT4  \
  character FILENAME*34,FILEEXT*23,/header*32  \
CHARACTER SHAPEANS*I,ZIGANS*1  \
CHARACTER SCANMODE*1,SCHEME,*1  
DATA 16/115=1  
10010 format(A)  
10012 format(A2)  
10030 format(I)  
10020 format(I)  
C Initialization and entry of parameters  
C  
C  
C TYPE *;  
C TYPE *; ***** DONSCAN.FOR......AUTO SCAN'  
C TYPE *;  
OPEN (UNIT=10,FILE='NOTHICK_ALLSHAPE1.DAT',STATUS='OLD', 
FORM='FORMATTED')  
READ (10,93764) SCANMODE  
READ (10,93765) XSCANSTEP !(X IND)  
IF (SCANMODE.EQ.'P')SCANSTEP=2000.  
READ (10,93765) YINDXSTEP !(Y IND)  
IF (SCANMODE.EQ.'P')YINDXSTEP=2000.  
SCANSTEP=JNINT(XSCANSTEP)  
INDXSTEP=JNINT(YINDXSTEP)  
READ (10,93765) SCANDIST !(X DIST)  
IF (SCANMODE.EQ.'P')SCANDIST=2000.  
READ (10,93765) INDXDIST !(Y DIST)  
IF (SCANMODE.EQ.'P')INDXDIST=2000.  
IF (SCANMODE.EQ.'L')THEN  
YINDXSTEP=20000.  
INDXSTEP=JNINT(YINDXSTEP)  
INDXDIST=20000.  
ENDIF  
READ (10,93764) ZIGANS  
READ (10,93765) PL
"SLANT" is the number of nanoseconds / microns that the echoes move due to nonlevelness of the experiment or thickness variation. It must be incorporated into the program as an additive or subtractive value to all the delay times by first multiplying "SLANT" by the cumulative indexing distance in the X-direction (N*SCANSTEP) as the scan is being performed and then adding / subtracting this value to each delay. "SLANT" must be determined before the scan is run by moving the X and Y stages over the scan length and noting the shift time of an echo, say B1. This shift will be consistent for B1 and B2 echoes in most cases but another slant factor will have to be defined for the echo off the reflector plate. So we have SLANT1 and SLANT2.

`READ (10,93765) SLANT1`
`SLANT1=SLANT1/10.**9.`
`RSLANT1=INT(SLANT1)`

`READ (10,93765) SLANT2`
`SLANT2=SLANT2/10.**9.`
`RSLANT2=INT(SLANT2)`

`READ (10,93765) SLANT3`
`SLANT3=SLANT3/10.**9.`
`RSLANT3=INT(SLANT3)`

`READ (10,93765) SLANT4`
`SLANT4=SLANT4/10.**9.`
`RSLANT4=INT(SLANT4)`

READ (10,93765) FREQUENCY
READ (10,93764) FILEEXT
K=0
K=K+1
IF (FILEEXT(K:K).NE."') GOTO 126
FILEEXT(K+1:K+4)=.DAT
! UNIT 6 = raw data
READ (10,93764) CHEADER
93764  FORMAT (A)
93765  FORMAT (F)
93766  FORMAT (I)
C  VOLTAGE = PL/20.
C  VOLTAGEU = PU/20.
C  VOLTAGE_MAX = PM/20.
C  TYPE *,RUN TEMPDIRSCAN.COM - DIRSCAN SMALLTRANS_DATA.DAT*
C  TYPE *,DATA FOR ULTRASONIC SCAN (ALL FLOATING PT. EXCEPT INDEXES)*
C  TYPE *,SCAN DIMENSIONS/INDEXING DATA (MICRONS)*
C  TYPE *,X INDEX - INTEGER EXPRESSION*
C  TYPE *,SCANTYPE
C  TYPE *,Y INDEX - INTEGER EXPRESSION*
C  TYPE *,INDXSTEP
C  TYPE *,SCANSTEP
C  TYPE *,X SCAN DISTANCE (MICRONS) - FLOATING POINT*
C  TYPE *,SCANDIST
C  TYPE *,X SCAN DISTANCE (MICRONS) - FLOATING POINT*
C  TYPE *,INDXDIST
C  TYPE *,LOWER & UPPER PRESSURES FOR CONTACT SCAN*
C  TYPE *,PL,PU
C  TYPE *,THICKNESS (MM)*
C  TYPE *,THICKN
C  TYPE *,SLANT1 (X-LEVELNESS TIME CORRECTION FACTOR for b1,b2) (nsec / um)*
C  TYPE *,SLANT1
C  TYPE *,SLANT2 (X-LEVELNESS TIME CORRECTION FACTOR for reflector echoes) (nsec / um)*
C  TYPE *,SLANT2
C  TYPE *,SLANT3 (Y-LEVELNESS TIME CORRECTION FACTOR for b1,b2) (nsec / um)*
C  TYPE *,SLANT3
C  TYPE *,SLANT4 (Y-LEVELNESS TIME CORRECTION FACTOR for reflector echoes) (nsec / um)*
C  TYPE *,SLANT4
C  TYPE *,TRANSUDER CENTER FREQUENCY (FLOATING POINT)*
C  TYPE *,FREQUENCY
C  TYPE *,FILENAME FOR RAW DATA*
C  TYPE *,FILEEXT
C  TYPE *,CHHEADER INFO (UP TP 32 CHARACTERS)*
C  TYPE *,CHHEADER
C  TYPE *,

1000 NSCAN = (SCANDIST/SCANTYPE) + 1
NINDX = (INDXDIST/INDXSTEP) - 1
SCANDIST = SCANTYPE*(NSCAN-1) ! SCANDIST = negative
INDXDIST = INDXSTEP*(NINDX-1)
IF (SCANMODE.EQ. 'S') THEN
WRITE (5,11100) SCANDIST, NSCAN, SCANTYPE,
+ INDXDIST, NINDX, INDXSTEP
ELSEIF (SCANMODE.EQ. 'L') THEN
WRITE (5,11120) SCANDIST, NSCAN, SCANTYPE
ELSEIF (SCANMODE.EQ. 'P') THEN
WRITE (5,11130)
ENDIF
11100 FORMAT (I, + '/ X scan to ',F7.0,' ',',I3,' steps of ',I5, + '/ Y index to ',F7.0,' ',',I3,' steps of ',I5)
11200 FORMAT(' ',
   - /* Line scan to 15 steps of 15 */
11300 FORMAT(' ', /* Point Measure repeated 2 times */
   $SCHEME='B$
   $THICK= INT(THICKN*1000.)$
   $FREQ= FREQUENCY$
   $NAVE= 64$
   $FILENAME = [ROTH.DATA]//FILEEXT$
WRITE (5,11690)
11690 FORMAT(' $NUMBER OF AVERAGES SET AT 64' )
   $NUMBER OF AVERAGES SET AT 64$

   C
   DIRECT ACCESS STORAGE
C
   OPEN( unit=6, file=FILENAME//CII, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=32, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=32, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=15, file=FILENAME//I2, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=16, file=FILENAME//I4, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=17, file=FILENAME//WAV, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=26, file=FILENAME//R4, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=36, file=FILENAME//SR4, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=37, file=FILENAME//S4W, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=46, file=FILENAME//TR4, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   OPEN( unit=47, file=FILENAME//T4W, status='NEW',
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')
   $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=2, +
   + $ACCESS='DIRECT,RECORDTYPE='FIXED,RECL=4, +
      form='FORMATTED,ORGANIZATION='SEQUENTIAL')

51 FORMAT(A32)
52 FORMAT(A2)
53 FORMAT(A4)
54 FORMAT(A2)
C
C Write header : X, Y dist, X, Y scan points, dummies
C
WRITE(6,REC=16,FMT=51)CH HEADER
   $WRITE(6,REC=16,FMT=51)$HEADER
   $WRITE(6,REC=16,FMT=51)$HEADER
   $WRITE(6,REC=16,FMT=51)$HEADER
5,629,865

69

WRITE(6,REC=16,FMT=51)ZIGANS
l6=16+1
(IDIST1 = - HI1X(SCANDIST/1000.)
IDIST2 = IFIX(INDXDIST/1000.)
IF (IDIST2.LT.1)IDIST2=1 'TRICK TO ALLOW DISPLAY IMAGE ON PSIDD
WRITE(15,REC=115,FMT=52)IDIST1
115=115+1
WRITE(15,REC=115,FMT=52)IDIST2
115=115+1
WRITE(15,REC=115,FMT=52)NSCAN
115=115+1
IF (SCANMODE.EQ.'L'.OR.SCANMODE.EQ.'P')NINDX=1 'MOD FOR LINE/POINT SCAN
WRITE(15,REC=115,FMT=52)NINDX
115=115+1
WRITE(15,REC=115,FMT=52)ISLAN1
115=115+1
WRITE(15,REC=115,FMT=52)ISLANT2
115=115+1
WRITE(15,REC=115,FMT=52)ISLANT3
115=115+1
WRITE(15,REC=115,FMT=52)ISLANT4
115=115+1
WRITE(15,REC=115,FMT=52)DENS
115=115+1
WRITE(15,REC=115,FMT=52)FREQ
115=115+1
WRITE(15,REC=115,FMT=52)NAVE
115=115+1
RETURN
end
SUBROUTINE XPARAM( SCANSTEP,INDXSTEP,NSCAN,NINDX )
integer*4 SCANSTEP,INDXSTEP
integer*2 NSCAN,NINDX
SCANSTEP = 50
INDXSTEP = 50
NSCAN = 50
NINDX = 50
THICK = 1
OPEN(unit=6, file='XXX.DAT', status='OLD',
form='UNFORMATTED'
SCANSTEP = 50
RETURN
end
SUBROUTINE M0VKLING (AXIS,DIST)
MOVE KLINGER A SPECIFIED DISTANCE IN SPECIFIED DIRECTION
C
C
byte IXYZ
character*1 XYZ
equivalence( XYZ, IXYZ )
common WR0,BUFFER
CALL STRTPIB
CALL INITINSTR(2)
CALL INITINSTR(3)
CALL INITINSTR(4)

10010 format(A)
10020 format(I)
10100 format("SDistance?"

100 WRITE (5,10000)
10000 format("X, Y, or Z?")
READ (5,10010)XYZ
IF (IXYZ.GT.90. OR .IXYZ.LT.88 )GOTO 900
IPORN=I
WRITE (5,10100)
READ (5,10020)IREQ
IF (IREQ.LT.0) THEN
IPORN=-I
IREQ=ABS(IREQ)
ENDIF
CALL SETXYZ (IREQ,IREQ,IPORN)
CALL MOVXYZ (IAXIS,IPORN)
GOTO 100
900 STOP
RETURN
end

INCLUDE 'ROTH.MENU' BASEO-2SEC_M.FOR
CRUNCH program for Nothickness velocity scan

Don Roth 20-sep-1994

integer*2 SCANDIST,INDXDIST,NSCAN,NINDX,JJ,SCANKNT
integer*2 DENS,TRFREQ,AVES,NPOINT,16,115,116,118,126
integer*2 ARRAYSIZ,KEICKLEN,SELFREQ,ENDFREQ,385,336,646
INTEGER*2 islant1,islant2,islant3,islant4,NACT.POINT,JXXX
integer*2 rawwaveb1(512),rawwaveb2(512),rawwavers(512),rawwaverns(512)
integer*4 117,17,REC,127,1SUB,137,147,127SUB,137SUB,147SUB,117SUB
real*4 _VEI_U_VEL,A_VEL
real*4 ASPECB1(1024), ASPECB2(1024)
real*4 BIASPEC(256), B2ASPEC(256)
real*4 ASPECRS(1024), ASPECRNS(1024)
real*4 RSASPEC(256), RNASASPEC(256)
REAL*4 B1PHASE(256), PHASEB1(1024)
REAL*4 B2PHASE(256), PHASEB2(1024)
REAL*4 RSASPEC(256), PHASERS(1024)
REAL*4 RSSPHASE(256), PHASERSNS(1024)
REAL*4 BB1(512), BB2(512), RRS(512), RNRS(512)
REAL*4 VEL(5000), TRDIAM, BUFSIZE
character FILENAME*52, FILEEXT*26, CHHEADER*32.
character FSIEXT*I6, CALNAME*32, FLAG*I
character DIFFANS*I, PHASE*I, PHASENS*I
character SHAPFREQANS*1
character SHAPFREQANS*1, ZIGANS*1, DIB*5
character CRUNCH-CODE*2, CFILTER*1
character actual_selfreqans*I
byte FILEFS1IN
equivalence ( BIASPEC, ASPECB1 ) ! -----------!
equivalence ( B2ASPEC, ASPECB2 ) ! *ASPEC 256 !
equivalence ( RSASPEC, ASPECRS ) ! -----------!
equivalence ( RNASASPEC, ASPECRNS ) ! *ASPEC 256 !
equivalence ( B1PHASE, PHASEB1 ) ! -----------!
equivalence ( B2PHASE, PHASEB2 ) ! -----------!
equivalence ( RSSPHASE, PHASERS ) ! -----------!
equivalence ( RSSPHASE, PHASERSNS )
equivalence ( FSIEXT, FILEFS1IN )

DATA 16/I/115/116/1/117/1/17/1/118/1/
DATA 126/I/127/1/185/1/146/1/147/1/
NPI=0

10010 format(A)
10014 format(I,A )
10020 format(I)
10030 format(I)
C
OPEN (UNIT=14,FILE="NOTHICK_SELFREQ1.DAT", Status='OLD', Form='FORMATTED')
READ (14,98652) FILEEXT
READ (14,98652) DRIVE
READ (14,98652) PHASE
READ (14,98652) PHASE1
READ (14,98652) DIFFANS
READ (14,10030) BUFLENGTH
READ (14,10030) BUFVEL
READ (14,10030) TRDIAM
CLOSE (14)
98652 FORMAT (A)
C ********* OPEN FILTER FILE FOR UPPER & LOWER LIMITS ********* C
OPEN (UNIT=24,FILE="NOTHICK_LIMITS1.DAT", Status='OLD', Form='FORMATTED')
READ (24,98652) CFILTER
READ (24,10030) L_VEL
READ (24,10030) U_VEL
U_VEL = U_VEL
AV_VEL = (L_VEL+U_VEL)/2.
C
K=0
K=K+1
IF (FILEEXT(K:K).NE. "") GOTO 126
FILENAME("ROTH.DAT")/FILEEXT
UNIT 6 = raw data
IF (DRIVE.EQ.'A') DIR='DUC2:'
IF (DRIVE.EQ.'C') DIR='DUC6:'

C ******************** DIRECT ACCESS STORAGE ******************** C
OPEN (unit=76, file=DIR/FILENAME//CH, status='NEW',
+ ACCESS='DIRECT', RECORDTYPE='FIXED', RECL=4,
+ form='FORMATTED', ORGANIZATION='SEQUENTIAL')
OPEN (unit=66, file=FILENAME//CH, status='OLD',
+ ACCESS='DIRECT', RECORDTYPE='FIXED', RECL=32,
+ form='FORMATTED', ORGANIZATION='SEQUENTIAL')
OPEN (unit=13, file=FILENAME//TE, status='OLD',
+ ACCESS='DIRECT', RECORDTYPE='FIXED', RECL=2,
+ form='FORMATTED', ORGANIZATION='SEQUENTIAL')
OPEN (unit=16, file=FILENAME//RA, status='OLD',
+ ACCESS='DIRECT', RECORDTYPE='FIXED', RECL=4,
+ form='FORMATTED', ORGANIZATION='SEQUENTIAL')
OPEN (unit=17, file=FILENAME//WAV, status='OLD',
+ ACCESS='DIRECT', RECORDTYPE='FIXED', RECL=2,
C
51 FORMAT(A2)
52 FORMAT(A2)
53 FORMAT(A4)
153 FORMAT(1X,I4,1X,I4,1X,I4,1X,E12.5,1X,E12.5,1X,E12.5,1X,E12.5,1X,E12.5,1X,E12.5,1X,E12.5,1X,E12.5,1X,  
        E12.5)
54 FORMAT(A2)

K=0
120 K=K+1
IF( FILEEXT(K:K).NE.'.' )GOTO 120
    FILEEXT(K:K+3)='CAL'
CALNAME = 'ROTH.DATA'//FILEEXT

C***************UNIT=7, SCP FILE FOR SPECTRA, vel, FOR DIRECT ACCESS************C
C
OPEN( unit=7, file=DIR//FILENAME. status='NEW', ACCESS='DIRECT,  
+ RECORDTYPE='FIXED, RECL=4, FORM='FORMATTED,  
+ ORGANIZATION='SEQUENTIAL' )
WRITE( 6,10200 )FILENAME
10200 format( 'Filename of analyzed data is \"A\" ' )
FILEEXT(K:K+3)='CAL'
CALNAME = 'ROTH.DATA'//FILEEXT
OPEN( unit=8, file=CALNAME, status='NEW', ACCESS='DIRECT,  
+ RECORDTYPE='FIXED, RECL=4, FORM='FORMATTED,  
+ ORGANIZATION='SEQUENTIAL' )
89651 DO 76065 IOP1=1,1024
    PHASES1(IOP1)=0.
    PHASES2(IOP1)=0.
    PHASES3(IOP1)=0.
    PHASES3N(IOP1)=0.
    ASPECB1(IOP1)=0.
    ASPECB2(IOP1)=0.
    ASPECB3(IOP1)=0.
    ASPECB3N(IOP1)=0.
76065 ASPECB3N(IOP1)=0.
KCOUNT=1

C       # # # Read preliminary scan info # # #
C

79193  READ(66,REC=16,FMT=5)CHEOADER
       16=16+1
READ(66,REC=16,FMT=5)SHAPEANS
       16=16+1
READ(66,REC=16,FMT=5)ZIGANS
       16=16+1

IF (SHAPEANS NE 'Y') GOTO 89714
OPEN(UNIT=12,FILE='ROTH.DA/a',FILEEXT(1)='DATISH',
     STATUS='OLD',FORM='UNFORMATTED') INFO ON WHETHER WE ARE ON HOLDER/SAMPLE

89714  READ(15,REC=115,FMT=52)SCANDIST
       115=115+1
READ(15,REC=115,FMT=52)INDEXDIST
       115=115+1
READ(15,REC=115,FMT=52)NSCAN
       115=115+1
READ(15,REC=115,FMT=52)INDEX
       115=115+1
READ(15,REC=115,FMT=52)ISLAN1
       115=115+1
READ(15,REC=115,FMT=52)ISLAN2
       115=115+1
READ(15,REC=115,FMT=52)ISLAN3
       115=115+1
READ(15,REC=115,FMT=52)ISLAN4
       115=115+1
READ(15,REC=115,FMT=52)DEN
       115=115+1
READ(15,REC=115,FMT=52)TRFRQ
       115=115+1
READ(15,REC=115,FMT=52)AVES
       115=115+1
READ(15,REC=115,FMT=52)XXX
       115=115+1
ARRAYSIZE=(NSCAN)*(JXXX) IF SCAN FINISHED PROPERLY, JXXX=INDEX
READ(16,REC=16,FMT=53)TIMESET
       116=116+1

ENDFREQ=TRFRQ*2.5
CALL POINT_FROM_FREQ(ENDFREQ,TIMESET,ACTPOINT)

DELTAF = 1/(TIMESET*20.)
NPOIN F=NSCAN*XXX IF SCAN FINISHED PROPERLY, XXX=INDEX
NWAVES = 3*NPOINT

C      !!! Loop thru scan data
ITEM_COUNT=0
DO 1890 NIP=1,NPOINT !NPOINT IS TOTAL OF ALL POINTS
SCANKNT=NIP
ITEM_COUNT=ITEM_COUNT+1
IF (SHAPEANS.EQ.'N.' AND NIP.GE.2) ITEM_COUNT=2
TYPE * * *
TYPE * * ITEM_COUNT=ITEM_COUNT
TYPE * * CRUNCH_CODE=CRUNCH_CODE
TYPE * *
CCVACCUM=0.
CONTINUE
NIP=((NIP-1)/NSCAN)+1 !Y LOCATION
NIP=NIP+1 !X LOCATION
IF (NIPLEQ.(NSCAN+1)) NIP=1 !RESET X LOCATION

C ***** MODIFICATION FOR ZIGZAG SCAN ******************************
IF (ZIGANS.EQ.'Y') THEN
   IF (NIPLEQ.2.OR.NIPLEQ.4.OR.NIPLEQ.6.OR.NIPLEQ.8.OR.NIPLEQ.10
      1 OR.NIPLEQ.12.OR.NIPLEQ.14.OR.NIPLEQ.16.OR.NIPLEQ.18.OR.NIPLEQ.
      1 EQ.20.OR.NIPLEQ.22.OR.NIPLEQ.24.OR.NIPLEQ.26.OR.NIPLEQ.28.OR.
      1 NIPJ.EQ.30 OR.NIPLEQ.J=52 OR.NIPLEQ.EQ.34 OR.NIPLEQ.EQ.36 OR.
      1 OR.NIPLEQ.EQ.40 OR.NIPLEQ.EQ.42 OR.NIPLEQ.EQ.44 OR.
      1 OR.NIPLEQ.EQ.46 OR.NIPLEQ.EQ.48 OR.NIPLEQ.EQ.50 OR.
      1 OR.NIPLEQ.EQ.52 OR.NIPLEQ.EQ.54 OR.NIPLEQ.EQ.56 OR.
      1 NIPJ.EQ.58 OR.NIPLEQ.EQ.60 OR.NIPLEQ.EQ.62 OR.NIPLEQ.EQ.64 OR.
      1 OR.NIPLEQ.EQ.66 OR.NIPLEQ.EQ.68 OR.NIPLEQ.EQ.70 OR.
      1 OR.NIPLEQ.EQ.72 OR.NIPLEQ.EQ.74 OR.NIPLEQ.EQ.76 OR.
      1 OR.NIPLEQ.EQ.78 OR.NIPLEQ.EQ.80 OR.NIPLEQ.EQ.82 OR.
      1 OR.NIPLEQ.EQ.84 OR.NIPLEQ.EQ.86 OR.NIPLEQ.EQ.88 OR.
      1 NIPJ.EQ.90 OR.NIPLEQ.EQ.92 OR.NIPLEQ.EQ.94 OR.
      1 OR.NIPLEQ.EQ.96 OR.NIPLEQ.EQ.98 OR.NIPLEQ.EQ.100 THEN
      NIPG=NSCAN-NIP+1 !FOR ZIGZAG SCAN, REVERSE BACK FOR DISPLAY
   ELSE
      NIPG=NIP
   ENDIF
   IF (zigan.EQ.'N') THEN !George Wood addition
      NIPG=NIP !George Wood addition
   ENDIF
   TYPE * * *
   TYPE * *
   IF (SHAPEANS.EQ.'N') GOTO 89715
READ(12)JJ,I,FLAG,SCANKNT !READ WHETHER 'H' OR 'S'
   TYPE * * X=,NIPG; Y=,NIPJ; FLAG=,FLAG
END

IF (FLAG.EQ.'S') THEN
   READ(16,REC=16,FMT=53)DELAYB1
   116=116+1
   READ(16,REC=16,FMT=53)VOLTSETB1
   116=116+1
   READ(26,REC=26,FMT=53)DELAYB2
   126=126+1
   READ(26,REC=26,FMT=53)VOLTSETB2
   126=126+1

END
READ(36,REC=136,FMT=S3)DELYR3
I36=I36+1
READ(36,REC=136,FMT=S3)VOLTSETRS
I36=I36+1
READ(46,REC=146,FMT=S3)DELYR3NS
I46=I46+1
READ(46,REC=146,FMT=S3)VOLTSETRNS
I46=I46+1
DO 54322 III=1,512
54322 READ(I7,REC=((I7-1)*512)+III,FMT=54)RAWWAVEB1(III)
I7=I7+1
DO 54323 III=1,512
54323 READ(I7,REC=((I7-1)*512)+III,FMT=54)RAWWAVEB2(III)
I7=I7+1
DO 54324 III=1,512
54324 READ(I7,REC=((I7-1)*512)+III,FMT=54)RAWWAVERS(III)
I7=I7+1
DO 54325 III=1,512
54325 READ(47,REC=((I47-1)*512)+III,FMT=54)RAWWAVERS(III)
I47=I47+1
ELSEIF (FLAG.EQ.'H')THEN
GOTO 1890
ENDIF
II=II+1
A = 0.
DO 1400 I=1,512
RAWWAVEB(I)=RAWWAVEB(I)/AVES
1400 A=A + RAWWAVEB(I)
ZERO = A/512.
DO 1560 I=1,512
1560 BB1(I) = ( REAL(RAWWAVEB(I)) - ZERO )*( VOLTSETB1 )*10./512.
II=II+1
A = 0.
DO 1401 I=1,512
RAWWAVERS(I)=RAWWAVERS(I)/AVES
1401 A=A + RAWWAVERS(I)
ZERO = A/512.
DO 1570 I=1,512
1570 BB2(I) = ( REAL(RAWWAVERS(I)) - ZERO )*( VOLTSETB2 )*10./512.
II=II+1
A = 0.
DO 1402 I=1,512
RAWWAVERS(I)=RAWWAVERS(I)/AVES
1402 A=A + RAWWAVERS(I)
ZERO = A/512.
DO 1580 I=1,512
1580 RR3(I) = ( REAL(RAWWAVERS(I)) - ZERO )*( VOLTSETRS )*10./512.
II=II+1
A = 0.
DO 1403 I = 1,512
RAWWAVERNS(I) = RAWWAVERNS(I)/AVES
1403 A = RAWWAVERNS(I)
ZERO = A/512.
DO 1590 I = 1,512
RAWWAVERNS(I) = ZERO
1590 VOLTSETRNS = 10./512.

IF (PHASE.EQ.'N') THEN
CALL CORR(BB1,BB2,TWOTAU_DELAY) ! for time delay in sample
ELSEIF (PHASE.EQ.'Y') THEN ! PHASE INVERSION FOR BETWEEN BB1 & BB2
CALL MCORR(BB1,BB2,TWOTAU_DELAY) ! for time delay in sample
ENDIF

IF (PHASE.EQ.'N') THEN
CALL CORR(RRNS,RRNS,DELTAT_DELAY) ! for delay between reflector peaks
ELSEIF (PHASE.EQ.'Y') THEN ! PHASE INVERSION FOR BETWEEN RRNS & RNS
CALL MCORR(RRNS,RRNS,DELTAT_DELAY) ! for delay between reflector peaks
ENDIF

CALL OBTAIN_Magnitude_Spectra(BB1,ASPECB1)
CALL OBTAIN_Magnitude_Spectra(BB2,ASPECB2)
CALL OBTAIN_Magnitude_Spectra(RRNS,ASPECRNS)
CALL OBTAIN_Magnitude_Spectra(RRNS,ASPECRNS)

ADELAYS = DELAYB2 - DELAYB1
ADELAYR = DELAYRNS - DELAYRS

TWOTAU = ADELAYS + (TIMES1/512)*TWOTAU_DELAY ! for time delay in sample
DELTAT = DELAYR + (TIMES1/512)*DELTAT_DELAY ! for delay between reflector peaks (sample vs. no sample)

CALL NOThick_Vel_Calc(DELTAT,TWOTAU,VEL_NOTHICK)

VEL(SCANKNT) = VEL_NOTHICK
VELPREV = VEL_NOTHICK

CALL STORE_Spectra(ASPECB1,ACTPOINT,17,17SUB) ! STORE SPECTRAS AT EACH SCAN POINT
CALL STORE_Spectra(ASPECB2,ACTPOINT,17,17SUB)
CALL STORE_Spectra(ASPECRS,ACTPOINT,17,17SUB)
CALL STORE_Spectra(ASPECRNS,ACTPOINT,17,17SUB)
CONTINUE

IF (SHAPEANS.EQ.'N') SCANKNT=NPOINT
CALL STORE_VEL(I7SUB,185,SCANKNT,ACTPOINT,VEL) 1(STORE VELOCITIES AT END OF .SPC FILE

SUBROUTINE STORE_VEL(I7SUB,185,NPOINT,ACTPOINT,VEL)
!! STORE VELOCITIES AND FIND MAX.MIN & STORE IN CAL FILE
INTEGER*2 NPOINT,185,ACTPOINT
INTEGER*4 I7SUB
REAL*4 VEL(5000)

VUP=0.0
VLO=10.**8.
VACCUM=0.
I7SUB=I7SUB+1
DO 94326 IJI=I,NPOINT
VMAX=AMAX1(VUP,VEL(IJI))
VUP=VMAX
VMIN=AMIN1(VLO,VEL(IJI))
VLO=VMIN
VACCUM=VACCUM+VEL(IJI)

WRITE(7,REC=I7SUB,FMT=53)VEL(IJI)

94326 I7SUB=I7SUB+1
53 FORMAT(A4)
VAVE=VACCUM/NPOINT

WRITE(8,REC=185,FMT=53)VUP
185=185+1
WRITE(8,REC=185,FMT=53)VLO
185=185+1
WRITE(8,REC=185,FMT=53)VAVE

RETURN
END

SUBROUTINE POINT_FROM_FREQ(FREQ,TIMEPERDIV,ACTPOINT)
DETERMINE SPECTRA POINT FROM SPECIFIC FREQUENCY IN SPECTRA
SEE SUBROUTINE CENTERFREQ FOR SIMILAR PROCESSING EXPLANATION
INTEGER*4 DUMMYFREQ
INTEGER*2 FREQ,ACTPOINT
REAL*4 TIMEPERDIV,DEL_FREQ,TEMPPOINT

C
C
DUMMY_FREQ=FREQ*1E+06
DEL_FREQ=(1./2.*(10.*TIMEPERDIV)))
TEMPPOINT=DUMMY_FREQ/DEL_FREQ
ACTPOINT=INT(TEMPPOINT)

C
RETURN
END

SUBROUTINE OBTAIN_MAGNITUDE_SPECTRA(TWAVE,DUMMY_ASPEC)
COMPLEX C1(1024),CSPEC(1024)
INTEGER*4 ISTAT
REAL*4 DUMMY_ASPEC(1024),PHASE(1024),TWAVE(512)

DO 500 I=1,512
C1(I)=CMPLX(TWAVE(I))
CALL LSPSFFT_COMPLEX(C1,CSPEC,1024,ISTAT)
CALL LSPSPHASE_ANGLE(CSPEC,PHASE,DUMMY_ASPEC,1024)
RETURN
END

SUBROUTINE STORE_SPECTRA(DUMMY_ASPEC,ACTPOINT,17,17SUB)
INTEGER*2 ACTPOINT
INTEGER*4 REC,17,17SUB
REAL*4 DUMMY_ASPEC(1024)
DO 94326 IJ=1,ACTPOINT
17SUB=((17- I)*ACTPOINT)+IJ
WRITE(7,REC-17SUB,FMT=53)DUMMY-ASPEC(IJ)
17=17~
FORMAT(A4)
RETURN
END

SUBROUTINE FnTER(VEL_NOTHICK,VELPREV.CFILTER.SHAPEANS,L_VEL,U_VEL,
                  AV_VEL,NIPG.NIPJ.SCANKNT:I)
CHARACTER CFILTER*,CRUNCH_CODE*2,S~HAPEANS*
INTEGER*2 NIPG,NIPJ,11,SCANKNT
REAL*4 L_VEL,U_VEL,AV_VEL
C ******** TEST FOR VELOCITY OUTSIDE FILTER LIMITS ***** C
IF (SCANKNT.EQ.1)VELPREV=AV_VEL
IF (CFILTER.EQ.'Y.AND.(VEL_notthick.LT.L_VEL.OR.VEL_notthick.GT.U_VEL)))THEN
VEL_NOTHICK=VELPREV
CRUNCH_CODE='BC
END
ICRUNCH_CODE_COUNT=1
TYPE *,'BAD POINT AT',NIPG,NIPJ
ENDIF

IF (ICRUNCH_CODE_COUNT.EQ.0)CRUNCH_CODE='G'
WRITE(66,REC=118,FMT=53)CRUNCH_CODE
I18=I18+1
WRITE(66,REC=118,FMT=53)NIPG
I18=I18+1
WRITE(66,REC=118,FMT=53)NIPJ
I18=I18+1

53 FORMAT(A4)
RETURN
END

SUBROUTINE NOTHICK_VEL_CALC(DELTAT,TWOTAU,VEL_NOTHICK)
VELWATER=0.149 !CM/USEC
VEL_NOTHICK=VELWATER*((DELTAT/TWOTAU)+1)
RETURN
END

SUBROUTINE CORR(B1,B2,CXX)
COMPLEX*B NCl 1(5 12),NCl 2(5 12),NC13(5 12)
COMPLEX*B NCl 4(5 12),NC15(5 12),NC22(5 12)
REAL*4 NC16(5 12),B1(5 12),B2(5 12),CXX
INTEGER*4 STATUS
DO 333 I=1,512
NC11(I)=CMPLX(B1(I))
NC22(I)=CMPLX(B2(I))
333 CONTINUE
CALL LSP$FFT_COMPLEX(NC11,NC12,5 12,0,STATUS)
CALL LSP$FFT_COMPLEX(NC22,NC13,5 12,0,STATUS)
DO 777 I0=1,512
NC14(IO)=NC12(IO)*CONJG(NC13(IO))
777 CONTINUE
CALL LSP$FFT_COMPLEX(NC14,NC15,5 12,0,STATUS)
IF (.NOT. STATUS) CALL LIB$STGNAI(444,VAL(STATUS))
AMAX=0.0
CXX=0.0
DO 765 I=1,256
NC16(I+256)=REAL(NC15(I))
NC16(I)=REAL(NC15(I+256))
765 CONTINUE
SUBROUTINE MCORR(B1,B2,CXX)

!! modified by using absolute value
!! of minimum of correlation function
!! to take into account phase inversion
!! of B2 w/ respect to B1 such as what
!! happens w/ PMCs

COMPLEX*8 NC11(512),NC12(512),NC13(512)
COMPLEX*8 NC14(512),NC15(512),NC22(512)
REAL*4 NC16(512),B1(512),B2(512),CXX
INTEGER*4 STATUS

DO 333 I=1,512
NC1I(I)=CMPLX(B1(I))
NC12(I)=CMPLX(B2(I))
CONTINUE

CALL LSP$FFT-COMPLEX(NC11,NC12,512,0,STATUS)
CALL LSP$FFT-COMPLEX(NC22,NC13,512,0,STATUS)
DO 777 IO=1,512
NC14(IO)=NC12(IO)*CONJC1(NC13(IO))
CALL LSP$FFT-COMPLEX(NC14,NC15,512,0,STATUS)
IF (.NOT. STATUS) CALL LIB$SIGNAL($WAL,STATUS))

AMAX=0.0
AMIN=1.0E+6
CXX=0.0
DXX=0.0

DO 765 I=1,256
NC16(I+256)=REAL(NC15(I))
NC16(I)=REAL(NC15(I-256))
CONTINUE
DO 881 I=1,512

IF(NC16(I).GT.AMAX)THEN
AMAX=NC16(I)
CXX=I
ENDIF

CONTINUE
RETURN

END
to take into account phase inversion of B2 w/ respect to B1 such as what happens w/ PMCs

IF(NC16(I).LT.AMIN) THEN
  AMIN=NC16(I)
  DXX=I
ENDIF

CONTINUE

CXX=DXX
CXX=CXX-257.
RETURN
END
5,629,865

CC     PROGRAM: NOTHICK_INDEXMAKER.FOR
         Read Data file written by CRUNCH and place data into
         "Grinnell-ready" files
         Dimension of scan (SCAN DIRECITION, INDEX DIRECTION)
         Number of points (SCAN, INDEX)

         INTEGER*2 ACTPOINT
         BYTE       BFILEEXT(23)
         CHARACTER  FILENAME*44,FILEEXT*23,DRIVE*1
         CHARACTER  CALNAME*44,SELFREQANS*1,SHAPEANS*1,DISK*5
         CHARACTER  HEAD*23,DIR*16
         EQUIVALENCE (FILEEXT, BFILEEXT)
         INTEGER JMARK=0
         INTEGER IBS=1

OPEN (UNIT=10,FILE='NOTHICK_DADQ1.DAT',STATUS='OLD',
FORM='FORMATTED')
READ (10,34572) FILEEXT
READ (10,34572) DRIVE
34572 FORMAT (A)
4321 CONTINUE

HEAD=FILEEXT

IF (DRIVE.EQ.'A')DIR='DUC2:[ROTH.TMAG]' IF (DRIVE.EQ.'C')DIR='DUCO:[ROTH.IMAG]' IF (DRIVE.EQ.'C')DISHK='DUCO:

ICHA=0
DO I=1,LEN(FILEEXT)
IF (ICHA.NE.0) GOTO 98564
IF (FILEEXT(I).NE.' ') THEN
   ICHA=I
   FILEXT(CHA:ICHA+4)='.SPC'
ENDIF
98564 END DO

FILENAME = '[ROTH.DAT]/FILENAME
K=0
20 IF (FILEXT(K).NE.' ') GOTO 20
   FILEXT(K+1:K+13) = 'CAL'
CALNAME = '[ROTH.DAT]/FILENAME
CALL DODISP1S (ICHA,FILENAME,CALNAME,HEAD,DIR,DISHK,
SHAPEANS,SELFREQANS,ACTPOINT)
SUBROUTINE DONDISPIS (ICHA,FILENAME,CALNAME,HEAD,DIR,DISKT,
SHAPEANS,SELFREQANS,ACTPOINT)

CLOSE(10)
END

READ Data file written by CRUNCH and write values to Grinnell
-UNCORRECTED & CORRECTED DATA IS DISPLAYED---

Header: X3, Y3 Dimension of scan (SCAN, INDEX)
N1, N2 Number of points (SCAN, INDEX)

INTEGER*4 REC,SCANPOS,XXX
INTEGER*2 IXLENGTH,A,IXLENGTH,H,ACTPOINT
INTEGER*2 X3,Y3,Y87,N1,N2,NNNN2,NNNNN1
INTEGER*2 DENS,NFREQ,AVES,X4,Y4,J87
INTEGER*2 COUNT,NCOUNT,NSCAN,NIDX
INTEGER*2 C(1024),D(460)
INTEGER*2 SCANDIST,INDEXDIST,TRFREQ,ENDFREQ,116
INTEGER*2 D SHAPE(460),189,6,115
INTEGER*2 ISLANT1,ISLANT2,ISLANT3,ISLANT4,INDEX,INDEX,INDEX
INTEGER*2 PSEUDOGRNL(460,1024),SELFREQ
REAL FREQ,MEAN
REAL RNUP(7),RNLO(7),CBUFFER(81)
BYTE BSCALEMARK(4),CHFREQ(3)
BYTE BSUFFREQ
BYTE BFILNAME(44),BNAMELAB(20),BDATA-STATUS(40)
CHARACTER FILENAME(44),FILEEXT*23,SELFREQANS*,FFLAG*1
CHARACTER CALNAME*44,CHHEADER*32,SHAPEANS*1,ZIGANS*1
CHARACTER CORNAME*31,CHFREQ_C*3,PHASE*1,PHASE*1
CHARACTER SCALEMARK*4,SELFREQ*4,HEAD*23,DIR*16,DISTK*5

EQUIVALENCE (CHFREQ,CHFREQ,C)
EQUIVALENCE (BNAMELAB,NAMELAB)
EQUIVALENCE (BDATA_STATUS,DATA_STATUS)
EQUIVALENCE (BSCALEMARK,SCALEMARK)
EQUIVALENCE (BSUFFREQ,SUFFREQ)

SCALEMARK='1 MM'
SUFFREQ='MHZ'

TAKE CHARACTER "FILENAME" AND DECODE TO BYTE "BFILNAME"

DECODE (44,533,FILENAME) BFILNAME

533 FORMAT(44A1)
DETERMINE SHORTENED (-.SIT) FILENAME POSITION

I6=1
I15=1
I16=1

KMARK=0
DO 4321 P=12,44
IF (KMARK.EQ.1) GOTO 4321
IF (BFFILENAME(IPR).EQ.d6) THEN
IEND=IPR
KMARK=1
ENDIF
CONTINUE

34572 FORMAT (A)
34573 FORMAT (I)
51 FORMAT(A32)
52 FORMAT(A2)
53 FORMAT(A4)

OPEN( unit=15, file=FILENAME(1:IEND-1)/:dat2, status='OLD',
+ ACCESS='DIRECT',RECORDTYPE='FIXED',RECL=2,
+ form='FORMATTED',ORGANIZATION='SEQUENTIAL' )

OPEN( unit=16, file=FILENAME(1:IEND-1)/:dat4, status='OLD',
+ ACCESS='DIRECT',RECORDTYPE='FIXED',RECL=4,
+ form='FORMATTED',ORGANIZATION='SEQUENTIAL' )

C OPEN( unit=26, file=FILENAME(1:IEND-1)/:dat8, status='OLD',
C + ACCESS='DIRECT',RECORDTYPE='FIXED',RECL=8,
C + form='FORMATTED',ORGANIZATION='SEQUENTIAL' )

C **************UNIT=7, SPC FILE FOR SPECTRA, vol, FOR DIRECT ACCESS**********
C
C OPEN( unit=7, file=FILENAME, status='OLD',ACCESS='DIRECT',
C RECORDTYPE='FIXED',RECL=4,FORM='FORMATTED',
C ORGANIZATION='SEQUENTIAL')

OPEN (UNIT=14,FILE='THICK_SELFREQ1.DAT',
1 STATUS='OLD',FORM='FORMATTED')

79193 READ(66,REC=16,FMT=51)HEADER
16=16+1
READ(66,REC=16,FMT=51)SHAPEANS
16=16+1
READ(66,REC=16,FMT=51)ZIGANS
16=16+1

READ (14,98652) FILEEXT
READ (14,98652) DRIVE
READ (14,98652) PHASE
READ (14,98652) PHASE1
READ (14,98652) DIFFANS
READ (14,10030) BUFLENGTH
READ (14,10030) BUFVEL
READ (14,10030) TRDIAM
CLOSE (14)
98652 FORMAT (A)
10020 FORMAT (I)
10030 FORMAT (F)
89714 READ(15,REC=I15,FMT=32)SCANDIST
115=115+1
READ(15,REC=I15,FMT=32)INDEXDIST
115=115+1
READ(15,REC=I15,FMT=J2)NSCAN
115=115+1
READ(15,REC=I15,FMT=32)NINDX
115=115+1
READ(15,REC=I15,FMT=32)ISLANT1
115=115+1
READ(15,REC=I15,FMT=32)ISLANT2
115=115+1
READ(15,REC=I15,FMT=32)ISLANT3
115=115+1
READ(15,REC=I15,FMT=32)ISLANT4
115=115+1
READ(15,REC=I15,FMT=32)DENS
115=115+1
READ(15,REC=I15,FMT=32)TRFREQ
115=115+1
READ(15,REC=I15,FMT=32)AVES
115=115+1
READ(15,REC=I15,FMT=32)JXXX
115=115+1
C ARRAYSIZE=(NSCAN)*(JXXX) IF SCAN FINISHED PROPERLY,JXXX=NINDX

ENCODE(3,19878,CHFREQ) TRFREQ (ANALYSIS FREQUENCY
19878 FORMAT (I))
READ(16,REC=116,FMT=73)TIMESET
116=116+1
ENDFREQ=2.5*TRFREQ
CALL POINT_FROM_FREQ(ENDFREQ,TIMESET,ACTPOINT)
DO 9999 ICHOICE=1,1
  I87=1
  ACCUM=0.
  NBAD=0
  IF (ICHOICE.EQ.1) THEN
    NAMELAB='VELOCITY CM/US'
    CORNAME='VEL_C//CHFREQ_C'
    NEWHEADER='CORRECTED DATA FOR VELOCITY'
    NEWU=25
    NEWU1=45
    NEWU2=49
  ENDIF

  OPEN (UNIT=NEWU, FILE=DISKT//FILBNAME(1:END-1)//CORNAME//
  + 'CORHEADER', STATUS='NEW', FORM='UNFORMATTED')

12345 IF (JMARK.EQ.1) GOTO 90 !FILES ARE ALREADY OPEN
  CONTINUE
  90 CONTINUE
  IF (JMARK.EQ.1) GOTO 89306
  CONTINUE

  IF (SHAPEANS.NE.'Y') GOTO 89317
  OPEN(UNIT=52, FILE=FILENAME(1:CHA+10)/'.DATS',
  + 'OLD', STATUS='OLD', FORM='UNFORMATTED') INFO ON WHETHER WE ARE ON HOLDER/SAMPLE

89317 SCANDIST=ABS(SCANDIST)
  AD=NSCAN
  XREAL = REAL(ABS(SCANDIST))
  YREAL = REAL(INXDIST)
  NPTS = NSCAN*JXXX !!!IF SCAN COMPLETED PROPERLY, JXXX=INDEX
  IF (SHAPEANS.EQ.'Y') THEN
    DO 5697 I=1,NPTS
    ENDIF

5697 READ(UNIT=52, END=89318)ABC,JABC,FFLAG,SCANKNT

89318 PTS = REAL(NSCAN)*REAL(JXXX)
  X3=ABS(SCANDIST)
  Y3=INDXDIST
  N1=NSCAN
  N2=JXXX
  IF (I.EQ.1) THEN
    NNNNN1=N1
    NNNNN2=N2
  ENDIF
XREAL = REAL(ABS(X3))
YREAL = REAL(Y3)
FREQ = REAL(NFREQ)
PTS = REAL(N1)*REAL(N2)

C Proportion GRINNELL window similar to scan dimensions
C

200 IF (Y87.GE.X3) THEN
    R1 = FLOAT(X3)/FLOAT(Y3)
    X4 = 400
    Y4 = ININT(400/R1)
    IF (Y4.GE.480) THEN
        X4 = 300
        Y4 = ININT(300/R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 200
        Y4 = ININT(200/R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 100
        Y4 = ININT(100/R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 50
        Y4 = ININT(50/R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 25
        Y4 = ININT(25/R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 10
        Y4 = ININT(10/R1)
    ENDIF
ELSEIF (Y87.LT.X3) THEN
    R1 = FLOAT(Y3)/FLOAT(X3)
    Y4 = IFIX(400*R1)
    X4 = 400
    IF (Y4.GE.480) THEN
        X4 = 300
        Y4 = ININT(300*R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 200
        Y4 = ININT(200*R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 100
        Y4 = ININT(100*R1)
    ENDIF
ELSEIF (Y87.LT.X3) THEN
    R1 = FLOAT(X3)/FLOAT(Y3)
    X4 = 400
    IF (Y4.GE.480) THEN
        X4 = 300
        Y4 = ININT(300*R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 200
        Y4 = ININT(200*R1)
    ENDIF
    IF (Y4.GE.480) THEN
        X4 = 100
        Y4 = ININT(100*R1)
    ENDIF
ENDIF
X4 = 50
Y4 = INT(50*R1)
ENDIF
IF (Y4 .GE. 480) THEN
X4 = 25
Y4 = INT(25*R1)
ENDIF
IF (Y4 .GE. 480) THEN
X4 = 10
Y4 = INT(10*R1)
ENDIF

XD2G = REAL(N1)/REAL(X4)
YD2G = REAL(N2)/REAL(Y4)
XD2X = REAL(X4)/REAL(N1)
YD2Y = REAL(Y4)/REAL(N2)
I_INT = INT(XD2X) - 1
J_INT = INT(YD2Y) + 1

89306 CONTINUE
10300 FORMAT (/Enter choice: /)
54322 FORMAT (A)
4319 FORMAT (*.VALUE=, F)
TYPE *.
TYPE *,'Working...'
TYPE *.
43191 FORMAT ('BAD PTS/TOT. PTS:<15, T14,
+ CURRENT VALUE=", F11.2)
7493 FORMAT (F11.2)
7494 FORMAT (I)

C
C RETURN MAX/MIN OF DATA FOR GRINNEL DISPLAY & PRODUCE ILINE
C OF GRINNEL DATA (BUFFER)

CALL RETMAXMIN_FILTER(FILENAME, CORNAME, CALNAME, NAMELAB, DISKT, IEND, JXXX,
1 NSCAN, SCANNT, SHAPEANS, ZIGANS, ICHOICE, TIMESET, I89, COUNT, ACTPOINT, NEWU, GRBOT,
1 GRTOP, MEAN)

7487 COUNT = NCOUNT
RMIN = GRBOT
RMAX = GRTOP
187 = 1
RNLO(CHOICE)=RMN
RNUP(CHOICE)=RMA

DO 500 KI=1,JXXX
DO 49721 JI=1,NSCAN
NIPI=JI
NIPI=KI
SCANPOS=((NIPI-1)*NSCAN)+NIPI
I87=SCANPOS
READ (NEWW,REC=187,FMT=53) CBUFFER(JI)
49721
187=I87+1
462 FORMAT(F11.2)
CALL SCGREAL (CBUFFER, C, NI, GRBOT, GRTOP, MEAN) ! GREY SCALED
DATA_STATUS=’CORRECTED DATA’//NAMELAB
C
C INTERPOLATE DATA GRID (EX,81X81) FOR X4 x Y4 GRINNELL DISPLAY
C
CALL INTR (C,D,NSCAN,X4)

C IF (K1.EQ.20) TYPE *,’D=’,D
C
C ********* MOD TO GET RID OF FALSE BORDER ***********
C IF (SHAPEANS.EQ.’Y’) THEN
IXCOUNT=0
DO 555 IOP=1,X4 1 FORWARD X-DIRECTION
IF (D(IOP).EQ.1) GOTO 555
IXCOUNT=IXCOUNT+1
IF (IXCOUNT.EQ.1) THEN
DO 655 IQ=1,1_INT
555 D(IOP-IQ+1)=1
ENDIF
655 CONTINUE
C IF (K1.EQ.20) TYPE *,’D=’,D
ENDIF

C ******** MOD TO GET RID OF FALSE BORDER ************
DO 480 J=1, X4
480 PSEUDOKNL (J, NIPI) = D (J)
500 CONTINUE
OPEN (UNIT=NEWU2, FILE=DIRHEAD/CORNAME/.'PD1', STATUS='NEW',
FORM='UNFORMATTED')

WRITE(NEWU2) X4,Y4,N1,N2,ICHOICE
WRITE(NEWU2) SCANDIST,TRFREQ
WRITE(NEWU2) RMIN,RMAX,MEAN

TYPE *,X4,Y4,N1,N2,ICHOICE
TYPE *,SCANDIST,TRFREQ
TYPE *,RMIN,RMAX,MEAN

IXLENGTH=0
605 DO 800 I=1,X4
   IX = I-1
   DO 540 J= 1, N2
540   C(J) = PSEUDOGRNL(IJ,J)
   CALL INTR(C,D,JXXX,Y4)
   C IF (I,J.EQ.200)TYPE *,D=,D
   IF (SHAPEANS.EQ.'Y') THEN
       IXCOUNT=0
       DO 1555 IOP=1,Y4 !FORWARD Y-DIRECTION
           IF (D(IOP).EQ.I) GOTO 1555
           IXCOUNT=IXCOUNT+1
           IF (IXCOUNT.EQ.1) THEN
               DO 1655 I/J=I,J-INT
                   D(IOP-I/J+1)=1
               CONTINUE
               IXCOUNT=0
               IF (D(IOP).EQ.1) GOTO 1755
               IXCOUNT=IXCOUNT+1
               DO 1855 I/J=J-I,INT
                   D(IOP+I/J-1)=1
               CONTINUE
               1555 CONTINUE
       1655 CONTINUE
   ENDIF
1755 CONTINUE
   C IF (I,J.EQ.200)TYPE *,D=,D
   ENDIF

C ********** MOD TO GET RID OF FALSE BORDER ********** C
   C IF (SHAPEANS.EQ.'Y') THEN
       IXCOUNT=0
       DO 1755 IOP=1,Y4 !BACKWARD Y-DIRECTION
           IF (D(IOP).EQ.I) GOTO 1755
           IXCOUNT=IXCOUNT+1
           IF (IXCOUNT.EQ.1) THEN
               DO 1855 I/J=J-I,INT
                   D(IOP-J/J+1)=1
               CONTINUE
               IXCOUNT=0
               IF (D(IOP).EQ.1) GOTO 1955
               IXCOUNT=IXCOUNT+1
               DO 1955 I/J=J-I,INT
                   D(IOP+J/J-1)=1
               CONTINUE
               1755 CONTINUE
   ENDIF
1955 CONTINUE
   C IF (I,J.EQ.200)TYPE *,D=,D
   ENDIF
C ********** MOD TO GET RID OF FALSE BORDER ********** C
C**********************************************************
MODIFICATION FOR COMPLEX SHAPE *******
**********

IF (SHAPEANS.NE.'Y')GOTO 675 ! NOT A COMPLEX SHAPE
IF (SHAPEANS.EQ.'Y')THEN

XXX=1
IXSTART=I
IYLENGTH=0
NKEEP=0
DO 8970 J=1,Y4
IF (D(J).LT.2)GOTO 8970

IF (D(J).GE.Z)THEN !SAMPLE DATA
IF (NK!ZP.EQ.O)IYSTART=J
IYLENGTH=IYLENGTH+1
C**** FURTHER MOD FOR GEORGE'S DISCONTINUITY ON YBCO HEX SAMPLE ******
ENDIF
IYLENGTH=IYLENGTH+I
D-SHAPE(IYLENGTH)=D(J)
ENDIF
IF (D(J-XXX).LT.?)GO TO 8875
C**** FURTHER MOD FOR GEORGE'S DISCONTINUITY ON YBCO HEX SAMPLE ******
8875 CONTINUE
C TYPE *,'IXSTART='.IXSTART,'IYSTART='.IYSTART,'IYLENGTH='.IYLENGTH
!ALL LUCITE IN THIS COLUMN IF (IYLENGTH.EQ.O)GOTO 800
IXLENCTH=IXLENCTH+I
WRITE(NEWU2)IXSTART,IYSTART,IYLENGTH
DO 8979 IJK=I,IYLENGTH
GOTO 800
ENDIF

8979 WRITE(NEWU2)D-SHAPE(IJK)

C**********************************************************

675 WRITE(NEWU2) D

800 CONTINUE

TYPE *','HIGHEST',NAMELAB,' IS ',RMAX
TYPE *','LOWEST',NAMELAB,' IS ',RMIN
TYPE *','
WRITE (NEWU1)NEU'HEADER
WRITE (NEWU1)RMIN,RMAX
WRITE (NEWU1)MEAN
WRITE (NEWU1)ISCALEF

50
SUBROUTINE SCGREAL( CC, DD, N, BOT, TOP, MEAN )

C Convert values in CC to 1 thru 255 (same as SCGR, with CC REAL)

INTEGER*2 N
INTEGER*2 DD(N)
REAL CC(N), MEAN
RAN = TOP - BOT
IF (RAN.EQ.0.) RAN = 1.
DO 2000 I = 1, N
CI = CC(I)
Q = (CI - BOT) / RAN
IF (Q.LE.0.) Q = 0.
IF (Q.GT.1.) Q = 1.

end
Q = Q*251.
C DD(J) = NINT(Q) + 1
C FOR MY MODIFICATION ******************C
IF (CO.EQ.0.) THEN
  DD(J) = 1
  GOTO 2000
ELSEIF (CO.NE.0.) THEN
  DD(J) = NINT(Q) + 2
ENDIF
C 2000 CONTINUE
RETURN
END

SUBROUTINE INTR
CC Interpolate values in C(N2) to "fit" into D(NN)
CC
C SUBROUTINE INTR ( C,CQD2,NN )
C
C INTEGER*2 N2,NN
C INTEGER*2 CC(N2), DD(NN)
C
C RN = FLOAT(N2) / FLOAT(NN)
C DO 3000 K = 1,NN
C R = FLOAT(K) * RN
C IR = INT( R )
C IRP = IR + 1
C RR = R - FLOAT( I )
C IF ( IRP.LT.1 .OR. IRP.GT.W ) THEN
C IR = NN-1
C IRP = NN
C ENDIF
C CD = FLOAT( CC(IR) )
C CI = FLOAT( CC(IRP) )
C D0 = ( 1 - RR ) * CD + RR * CI
C 3000 DD(K) = IFIX(D0)
C 3000 DD(K) = IINT(D0)
C
C RETURN
C END
INTEGER*2 185,187,SCANKNT
INTEGER*2 1ABC,JABC,JXXX,189
INTEGER*2 NCOUNT
INTEGER*2 ZIGCOUNT
INTEGER*2 NBAD,COUNT,LCOUNTER
REAL  DATATHING(1024,9)
REAL  MEAN,VEL(5000)
REAL  CBUFFER(81)
CHARACTER FILENAME*44,CORNAME*15,NAMELAB*20,FFLAG*1
CHARACTER SHAPEANS*,ZIGANS*,DISKT*5,CALNAME*4
INTEGER*4 SCANPOSI

TYPE *,NAMELAB
LCOUNTER=1
ZIGCOUNT=1
ACCUM=0.
SCANPOS=0
SCANPOS1=0
NBAD=COUNT
187=1
XMAX=0.
XMIN=10000000.
PTS = REAL(NSCAN)*REAL(JXXX)
NPTS=NSCAN*JXXX

CAL FILE WITH MAX/MIN/MEAN

C OPEN(unit=8, file=FILENAME, status='OLD', ACCESS='DIRECT',
+ RECORDYPE='FIXED',RECL=4,FORM='FORMATTED',
+ ORGANIZATION='SEQUENTIAL')
185=1

read(8,REC=185,FMT=53)VUP
185=185+1
read(8,REC=185,FMT=53)VLO
185=185+1
read(8,REC=185,FMT=53)VAVE

IF (SHAPEANS.EQ.'Y') THEN
NPTS=SCANKNT
REWIND(32)
ENDIF

17=NPTS*4*ACTPOINT-1  ISTART OF VELOCITY READS IN_SPC_FILE

C SPC FILE (UNIT=7) CONTAINS ALL VELOCITY & SPECTRA DATA

OPEN (UNIT=7, FILE=DISKT//FILENAME, STATUS='OLD', ACCESS='DIRECT',
+ RECORDYPE='FIXED',RECL=4,FORM='FORMATTED',
+ ORGANIZATION='SEQUENTIAL')

OPEN (UNIT=NEWU,FILE=DISKT//FILENAME(1:END-1)//CORNAME//.COR',

53
DO 500 NIPJ=1,JXXX
DO 450 NIP1=1,NSCAN
C***** GET INFO FROM HOLDER.DAT (FFLAG='S' -> SAMPLE) (FFLAG='H' -> HOLDER)
KCON=KCON+1
J=NIPJ
SCANPOS=((NIPJ-1)*NSCAN)+NIP1 ;TRUE SCAN POSITION
IF (SHAPEANS.NE.'Y')GOTO 87123
READ(32)IABC,JABC,FFLAG,SCANNT
C
TYPE *,XH=JABC, YH=JABC, FFLAG=FFLAG
IF (FFLAG.NE.'S')GOTO 8702
!TRANSDUCER IS ON HOLDER GOTO NEXT PT.
SCANPOS=SCANPOS+1
CC
SCANPOS I=I=((NIPJ-I)*NSCAN)+NIP1 !TRUE SCAN POSITION FOR FILE RETRIEVAL FOR CIRCLE DISK SCAN
READ(7,REC=I7,FMT='53)VEL(SCANPOS)
17=17+1
DATATHING(J,5)=VEL(SCANPOS)
NPTS=(NIPJ-I)*NSCAN+NIP1
IF (NPTS.NE.NSCAN*JXXX)NPTS=NSCAN*JXXX
C
IF (ICHOICE.EQ.1)THEN
IF (SCANPOS.EQ.1)DATATHINGPREVIOUS=VAVE
IF (DATATHING(1,5).LT.(VLO+VLO*0.01))OR.
C
1 DATATHING(1,5).GT.(VUP-VUP*0.01))DATATHING(1,5)=DATATHINGPREVIOUS
C ENDF
IF (SHAPEANS.NE.'Y')GOTO 87651
8702 IF (FFLAG.NE.'S')THEN ISET TO 0 // TRANSDUCER IS ON HOLDER GOTO NEXT PT.
CBUFFER(J)=0.0
GOTO 450
ENDIF
87651 CBUFFER(J)=DATATHING(1,5)
390 CONTINUE
C***** KEEP TRACK OF MAX & MIN LIMITS OF CORRECTED DATA********C
C**** FOR FFLAG.NE.'S' -- WE SKIP THIS STEP SO THAT '0' IS NOT CONSIDERED
RMAX=AMAX((CBUFFER(J),XMAX)
RMN=AMIN((CBUFFER(J),XMIN)
XMAX=RMAX
XMIN=RMN
450 CONTINUE

IF (ZIGANS.EQ.'N') THEN ! NOT A ZIGZAG SCAN  - A NORMAL SCAN
  DO 49720 JI=1,NSCAN
    WRITE (NEWU,REC=187,FMT=53) CBUFFER(JI)
    ACCUM=ACCUM + CBUFFER(JI)
  49720 CONTINUE
ELSEIF (ZIGANS.EQ.'Y'.AND.ZIGCOUNT.EQ.1) THEN ! ZIGZAG SCAN - A NORMAL SCAN
  DO 49721 JI=I,NSCAN
    WRITE (NEWU,REC=187,FMT=53) CBUFFER(JI)
    ACCUM=ACCUM + CBUFFER(JI)
  49721 CONTINUE
ELSEIF (ZIGANS.EQ.'Y'.AND.ZIGCOUNT.EQ.0) THEN ! ZIGZAG SCAN
  DO 49722 JI=NSCAN,I,-1
    WRITE (NEWU,REC=187,FMT=53) CBUFFER(JI)
    ACCUM=ACCUM + CBUFFER(JI)
  49722 CONTINUE
ENDIF
C***** FOR ZIGZAG SCANS  - TO ALTER ROW STORAGE ***********C

IF (ZIGANS.EQ.'Y'.AND.ZIGCOUNT.EQ.1) THEN
  ZIGCOUNT=0
  GOTO 500
ELSEIF (ZIGANS.EQ.'Y'.AND.ZIGCOUNT.EQ.0) THEN
  ZIGCOUNT=1
ENDIF
C500 TYPE *,'Y=',JABC,';',CBUFFER=';',CBUFFER
500 CONTINUE
5001 GRBOT = RMIN
      GRTOP = RMAX
      TYPE *,'RMIN=',GRBOT
      TYPE *,'RMAX=',GRTOP
      MEAN = ACCUM/SCANPOS
      TYPE *,'MEAN=',MEAN
      IF (SHAPENSL.EQ.'Y') GOTO 88888
      REWIND(52)
88888 CONTINUE
53 FORMAT (A4)
43191 FORMAT (*','BAD PTS =',J5,' AT LOCATION (SCANPOS) =',I5)
RETURN
END
DETERMINE SPECTRA POINT FROM SPECIFIC FREQUENCY IN SPECTRA

SEE SUBROUTINE CENTERFREQ FOR SIMILAR PROCESSING EXPLANATION

INTEGER*4 DUMMYFREQ
INTEGER*2 SELFREQ, ACTPOINT
REAL*4 TIMEPERDIV, DELFREQ, TEMPPOINT

DUMMYFREQ = SELFREQ * 1E+06
DELFREQ = (1/(2.*10.*TIMEPERDIV)))
TEMPPOINT = DUMMYFREQ / DELFREQ
ACTPOINT = JNINT(TEMPPOINT)

RETURN
END

SUBROUTINE CENTERFREQ(DUMMYP, TIMEPERDIV, ACTFREQ)

DETERMINE FREQUENCY FROM POINT IN B1(F) SPECTRA

INTEGER*2 DUMMYP, ACTFREQ
REAL*4 TIMEPERDIV, DELFREQ, TEMPFREQ

WE KNOW THAT DELFREQ = (1/(2*N*DELTIME))
= (1/(2*N*(10*TIMEPERDIV)))
= (1/(2*(10*TIMEPERDIV)))

WHERE N = 512 POINTS/ACQUISITION (SCREEN)
TIMEPERDIV = TIME SETTING PER DIVISION (EX. 50 NSEC)
2 = WE EXTENDED 512 ARRAY TO 1024 ARRAY FOR PROCESSING
10 = THE NUMBER OF DIVISIONS/SCREEN

FOR EXAMPLE:
WE CAN DETERMINE THE FREQUENCY FOR WHICH THE POINT "MAXBIP"
CORRESPONDS TO BY JUST MULTIPLYING (DELFREQ) * (MAXBIP)

DELFREQ = (1/(2.*10.*TIMEPERDIV)))
TEMPFREQ = DELFREQ * DUMMYP
ACTFREQ = TEMPFREQ / 1E+06

RETURN
END

SUBROUTINE INTR (F1, F2, N, M)
IMPLICIT NONE

INTEGER*2 I, M, N
INTEGER*2 FI(N), F2(M)
REAL*4 X, XA(512), Y, YA(512)

DO I = 1, N
   XA(I) = FLOAT(I)
END DO

DO I = 1, M
   YA(I) = FLOAT(FI(I))
END DO

DO I = 1, M
   X = FLOAT(I-1) * FLOAT(N-I) / FLOAT(M-I) + 1.0
   CALL LINEAR_INTERPOLATION (XA, YA, N, X, Y)
   F2(I) = ININT(Y)
END DO

RETURN
END

SUBROUTINE LINEAR_INTERPOLATION (XA, YA, N, X, Y)

LINEAR_INTERPOLATION computes the value of the piecewise linear spline interpolant of the points in the arrays XA and YA at the point X. This routine is a modified version of the FORTRAN routine SPLINT found in the book "Numerical Recipes: The Art of Scientific Computing" by Press, Flannery, Teukolsky, and Vetterling published by the Cambridge University Press, 1986.

Variables:

NMAX : A constant which determines the array sizes for U, X, and Y.
K : The midpoint between KHI and KLO, used to find the two indices into the XA table such that XA(KLO) < X < XA(KHI).
KHI : The greater of the two indices into the XA array such that XA(KLO) < X < XA(KHI).
KLO : The lesser of the two indices into the XA array such that XA(KLO) < X < XA(KHI).
N : The index of the arrays and Y that contain interpolation points.
A : Weight given to YA(KLO).
B : Weight given to YA(KHI).
H : The difference of XA(KHI) and XA(KLO).
X : The value of X at which to compute the value of the interpolation function.
XA : The table of X coordinates of the interpolation points.
YA : The table of Y coordinates of the interpolation points.

SUBROUTINE LINEAR_INTERPOLATION (XA, YA, N, X, Y)
IMPLICIT NONE

INTEGER NMAX
PARAMETER (NMAX=512)

INTEGER*2 K, KHI, KLO, N
REAL*4 A, B, H
REAL*4 X, XA(NMAX), Y, YA(NMAX)

KLO = 1
KHI = N
DO WHILE (KHI-KLO.GT.1)
   K = (KHI + KLO) / 2
   IF (XA(K).GT.X) THEN
      KHI = K
   ELSE
      KLO = K
   END IF
   END DO

H = XA(KHI) - XA(KLO)
IF (H.NE.0.0) THEN
   A = (XA(KHI) - X) / H
   B = (X - XA(KLO)) / H
   Y = A * YA(KLO) + B * YA(KHI)
END IF

RETURN
END
C ******** NOTHICK_DRAW.FOR *****************C

INTEGER*2 TRFREQ,XX,IXLENGTH
INTEGER*2 SCANDIST,INDXDIST,NSCAN,NIDX,DENS
INTEGER*2 SELRFREQ,AVES,16,15,1,ISLANT1,ISLANT7,ISLANT3,ISLANT4
CHARACTER RESPI1,RESPI2,SELFREQANS1,SELFREQANS2
CHARACTER NEWOLD3,SHAPEANS1,ZIGANS1,CHFREQ_C1,CHDFR1
BYTE B, CHFREQ(3), BSUFFREQ(4)
BYTE BSCALEMARK(4)
BYTE BDATA_STATIJS(40), BLABEL(30), BFILEEXT(23)
CHARACTER LABEL*30, FILEEXT*23, CHEADER*32, TEBAR*
CHARACTER KESPl*, SELFREQANS*, RESP2*
CHARACTER NEWOLD*, SELFREQANS*, SHAPEANS1, ZIGANS1, CHFREQ_C1, CHDFR1

EQUIVALENCE (BFILEEXT, FILEEXT)
EQUIVALENCE (CHFREQ, CHFREQ_C)
EQUIVALENCE (L, B)
EQUIVALENCE (BSCALEMARK, SCALEMARK)
EQUIVALENCE (BSUFFREQ, SUFFREQ)
EQUIVALENCE (BDATA_STATUS, DATA_STATUS)
EQUIVALENCE (BLABEL, LABEL)

NEWOLD='OLD'

OPEN (UNIT=10, FILE='NOTHICK_DADQ1.DAT', STATUS=NEWOLD, 
FORM='FORMATTED')
READ (10, 511) FILEEXT
READ (10, 511) DRIVE
511 FORMAT (A)
51 FORMAT (A32)
52 FORMAT (A2)
53 FORMAT (A4)
677 FORMAT (I)

OPEN( unit=66, file=[ROTI.I.DATA]/[FILEEXT]/.datCH, status='OLD', 
+ ACCESS='DIRECT', RECORDTYPE='FIXED', RECL=32, 
+ FORM='FORMATTED', ORGANIZATION='SEQUENTIAL')

OPEN (UNIT=14, FILE='NOTHICK_SELRFREQ1.DAT', 
1 STATUS='OLD', FORM='FORMATTED')
OPEN (unit=15, file=[ROTH.DATA]/[FILEEXT]/.dat12, status='OLD', 
1 ACCESS='DIRECT', RECORDTYPE='FIXED', RECL=2, 
+ FORM='FORMATTED', ORGANIZATION='SEQUENTIAL')

16=1

79193 READ(66,REC=16,FMT=51)CHEADER
16=16+1
READ(66,REC=16,FMT=51)SHAPEANS
16=16+1
READ(66,REC=16,FMT=51)ZIGANS
16=16+1

READ (14,98652) FILEEXT
READ (14,98652) DRIVE
READ (14,98652) PHASE
READ (14,98652) PHASE1
READ (14,98652) DIFFANS
READ (14,10030) BUFSIZE
READ (14,10030) BUFSIZE
CLOSE (14)
98652 FORMAT (A)
10000 FORMAT (I)
10000 FORMAT (F)

115=1
89714 READ(15,REC=I15,FMT=52)SCANDIST
115=115+1
READ(15,REC=I15,FMT=52)INDXDIST
115=115+1
READ(15,REC=I15,FMT=52)NSCAN
115=115+1
READ(15,REC=I15,FMT=52)NINDEX
115=115+1
READ(15,REC=I15,FMT=52)JSAMPLE1
115=115+1
READ(15,REC=I15,FMT=52)JSAMPLE2
115=115+1
READ(15,REC=I15,FMT=52)JSAMPLE3
115=115+1
READ(15,REC=I15,FMT=52)JSAMPLE4
115=115+1
READ(15,REC=I15,FMT=52)DENS
115=115+1
READ(15,REC=I15,FMT=52)TRFREQ
115=115+1
READ(15,REC=I15,FMT=52)AVES
115=115+1
READ(15,REC=I15,FMT=52)JXXX
115=115+1
READ(15,REC=I15,FMT=52)XLENGTH
C ARRAYSIZE=(NSCAN)*(JXXX) IF SCAN FINISHED PROPERLY,JXXX=NINDEX

ENCODE (3,19878,CIIFREQ) TRFREQ ANALYSIS FREQUENCY
19878 FORMAT (F)

C DETERMINE SHORTENED (-.DAT) FILEEXT POSITION
C KMARK=0
C DO 4321 IPR=1,34
C IF (KMARK.EQ.1) GOTO 4321

GO
C IF (BFILEEXT(IPR),EQ,46) THEN !PERIOD/DOT IS ASCII 46
C IEND=IPR
C KMARK=1
C ENDIF
C4321 CONTINUE
C FILEEXT=FILEEXT(1:IEND-1)

TYPE *,'CURRENT IMAGE FILE IS ',FILEEXT
TYPE *,'
TYPE *,'IF YOU WANT A DIFFERENT IMAGE FILE, YOU MUST
TYPE *,'RECALL AN OLDER "NOTHICK_DADQ1.DAT" FILE'
TYPE *,'

IF (DRIVE,EQ,'A')DIR='DUC2':[ROTH.IMAG]
IF (DRIVE,EQ,'C')DIR='DUC0:[ROTH.IMAG]

5321 TYPE *,'CURRENT DRIVE TO READ IMAGES IS ',DIR(1:5)
IF (DIR(1:5),EQ,'DUC0:')THEN
TYPE *,'DO YOU WANT TO CHANGE TO DUC2:(Y/N)
READ (5,51)CHDR1
IF (CHDR1,NE,'Y',AND,CHDR1,NE,'N')GOTO 5321
ELSEIF (DIR(1:5),EQ,'DUC2:')THEN
TYPE *,'DO YOU WANT TO CHANGE TO DUC0:(Y/N)
READ (5,51)CHDR1
IF (CHDR1,NE,'Y',AND,CHDR1,NE,'N')GOTO 5321
ENDIF

211 TYPE *,'
TYPE *,'DO YOU WANT TO SEE TEXT AND COLORBAR (Y/N)?'
READ (5,51)TEBAR
IF (TEBAR,NE,'Y',AND,TEBAR,NE,'N')GOTO 211

FILENAME='[ROTH.DATA]'IIFILEEXT
CALL GRINIT
CALL GRSRST

TYPE *,'
TYPE *,'IMAGE FREQUENCY (MHz) = ',TRFREQ
ENCODE (3,9878,CHFREQ) TRFREQ
234 TYPE *,' ***** NOTHICKNESS VELOCITY IMAGE *****
TYPE *,'
WRITE (5,394)
9878 FORMAT (I3)
394 FORMAT ('%',COLOR (C) OR B&W (R) IMAGE?')
READ (5,2222) RESP1
IF (RESP1.EQ.'C') THEN
    TYPE '*'; I
    WRITE (5,499)
    FORMAT (@',')
    READ (5,2222) RESP2
ENDIF
500
CALL IMAGE(NEWOLD,FILEEXT,CHFREQ_C,RESP1,RESP2,SHAPEANS,IXLENGTH,
            FILENAME,DIR,TEBAR)
444 CONTINUE
STOP
END

INTEGER*2 D(460),CHANNEL,INPUT,IXLENGTH
INTEGER*2 X4,Y4,N1,N2,TRFREQ,SCANDIST
CHARACTER*1 RESPl *1,SHAPEANS*1,RESP2*1,TEBAR
CHARACTER CHFREQ_C*3
REAL RMIN, RMAX, MEAN
BYTE B, CHFREQ(3), BSUFFREQ(4), CZERO(2)
BYTE BFILENAME(34), BSCALEMARK(4)
BYTE BDATA_STATUS(40), BLABEL(30)
BYTE CHMEAN(5),CHRMAX(5),CHRMIN(5)
CHARACTER FILNAM*48, LABEL*30, FILEEXT*23
CHARACTER L*48, DIR*16, CZERO*2
CHARACTER SUFFREQ*4, FILENAME*34, SCALEMARK*4, DATA_STATUS*40
EQUIVALENCE (L,B)
EQUIVALENCE (BSCALEMARK, SCALEMARK)
EQUIVALENCE (BSUFFREQ, SUFFREQ)
EQUIVALENCE (BDATA_STATUS, DATA_STATUS)
EQUIVALENCE (BLABEL, LABEL)
EQUIVALENCE (CZERO, BZERO)
CZERO='0'
LLLL=0
IMAGE=1
212 CONTINUE
IF (IMAGE.EQ.1) FILNAM='VEL_C//CHFREQ_C//PDK'
501 CONTINUE
CHANNEL=0
IF (CHANNEL.NE.0.AND.CHANNEL.NE.2.AND.CHANNEL.NE.3) GOTO 501
IF (CHANNEL.EQ.0) THEN
    INPUT=0
499 FORMAT ('COLORRED (I<) OR NORMAL COLOR (C) SCHEME?')
MASK=I
ELSEIF (CHANNEL.EQ.2) THEN
INPUT=I
ENDIF

CALL ERASE

OPEN (UNIT=6,FILE=DIR$FILEEXT$FILNAM,STATUS='OLD,'FORM='UNFORMATTED')
READ(6) X4,Y4,N1,N2,NVALUE
READ(6) SCANDIST,TRFREQ
READ(6) RMIN,RMAX,MEAN

IF (NVALUE.EQ.1) LABEL='VELOCITY CM/US'

ICOLORBAR=0
SUFFREQ=' MI'
SCALEMARK=' MM'
DATA_STATUS=LABEL

CALL GRNIN(1,INPUT,INPUT,INPUT)
CALL GRNBY(1,1,1)
CALL GRSBFD

C **************************************************MODIFICATION FOR COMPLEX SHAPES *****

IF (SHAPEANS.NE.'Y')IX4=X4

DO 100 I=I,IX4

IF (SHAPEANS.NE.'Y')THEN
IX4=IXLENGTH
ENDIF

DO 100 I=I,IX4

READ(6) D
GOTO 789
ELSEIF (SHAPEANS.EQ.'Y')THEN

IF (I.NE.1.AND.IYLENGTH.EQ.1)GOTO 100
READ(6,END=100)IXSTART,IYSTART,IYLENGTH
DO 7897 UX=1,IYLENGTH

7897 READ(6,END=7897)

789 CONTINUE

CALL GRWA(D,IXSTART,IYSTART,1,IYLENGTH,0,0,0,1,1)

C **************************************************MODIFICATION FOR COMPLEX SHAPES *****

CALL GRSBFD

100 CONTINUE
IF (RESP1.EQ.'B')THEN
IF (TEBAR.EQ.'N')GOTO 213
CALL PLACEBAR(0)
GOTO 8999
ELSEIF (RESP1.NE.'B')THEN
IF (RESP2.EQ.'C')THEN
CALL COLORS (IVALUE,ICOLORBAR)
ELSEIF (RESP2.EQ.'R')THEN
CALL COLORRED (IVALUE,ICOLORBAR)
ENDIF
ENDIF
IF (TEBAR.EQ.W)GOTO 213
DECODE (34,8989,FILENAME) BFILENAME
FORMAT (34A1)
ENCODE (3,909,CHFREQ) TRFREQ
FORMAT (13)
CALL GRFCD (I,255,0,O,BFILENAME,,1,441,6,0,34,0,0,0)
CALL GRFCD (I,255,0,O,CHFREQ,200,441,6,0,3,0,0,0)
CALL GRFCD (1,255,0,O,SSUFFREQ,215,441,6,0,4,0,0,0)
CC PRINT LABEL TO GRINNELL
CC
NCHAR=LEN(DATA-STATUS)
DO 765 IKY=I,NCHAR
IKX=400-(IKY-1)*13
CALL GRFCD (1,255,0,O,BDATA STATW(IKY),505,IKX,0,O,1,0,0,0)
CALL GRSHFD
765 CONTINUE
CC
CC CALCULATE MAX, MIN, AND MEAN
CC
IF (IVALUE.EQ.1.OR.IVALUE.EQ.2.OR.IVALUE.EQ.6.OR.IVALUE.EQ.7) THEN
IF (IVALUE.EQ.1) THEN
IF (RMAX.EQ.1.0) RMAX=.9999
ENDIF
RRMAX=RMAX*10.**4.
RRMIN=RRMIN*10.**4.
RMEAN=RMEAN*10.**4.
ELSEIF (IVALUE.EQ.3.OR.IVALUE.EQ.4) THEN
RRMAX=RMAX/100.
ELSEIF (IVALUE.EQ.5) THEN
  IF (RMAX.GE.100.) RMAX=99.9
  IF (RMIN.GE.100.) RMIN=99.9
  IF (MEAN.GE.100.) MEAN=99.9
  IF (RMAX.LT.10.) THEN
    RRMAX=RMAX*10.**4.
    RRMIN=RMIN*10.**4.
    RMEAN=MEAN*10.**4.
  ELSE
    RRMAX=RMAX*10.**3.
    RRMIN=RMIN*10.**3.
    RMEAN=MEAN*10.**3.
  ENDIF
ENDIF

ELSEIF (IVALUE.EQ.2.OR.IVALUE.EQ.6.OR.IVALUE.EQ.7) THEN
  JKMN=465
ELSEIF (IVALUE.EQ.1.OR.IVALUE.EQ.3.OR.IVALUE.EQ.4) THEN
  JKMN=463
ELSEIF (IVALUE.EQ.5) THEN
  JKMN=470
ENDIF

CC DRAW DECIMAL POINT FOR MIN AND MAX TO GRINNELL

DO 9879 IU=0,350,350
  CALL GRFAR (1,255,0,0,JKMN,IU,1.1)
  CALL GRSBFD
9879  CONTINUE

IMEAN=JNINT(RMEAN)
IRMAX=JNINT(RRMAX)
IRMIN=JNINT(RRMIN)

TYPE *, 'CORRECTED:
TYPE *, IMEAN=IMEAN
TYPE *, IRMAX=IRMAX
TYPE *, IRMIN=IRMIN

IF (IMEAN.GE.1.0*10.**5) GOTO 73920
ENCODE (5,9878,CHMEAN) IMEAN
73920 IF (IRMAX.GE.1.0*10.**5) GOTO 73921
ENCODE (5,9878,CHRMAX) IRMAX
73921 IF (IRMIN.GE.1.0*10.**5) GOTO 73922
ENCODE (5,9878,CHRMIN) IRMIN
73922 CONTINUE
9878  FORMAT (15)
IF (MEAN.GE.1.0*10.**6.) GOTO 93920
ZZZZZ=RMAX-RMIN
IF (ZZZZZ.EQ.0.) ZZZZZZ=1.
SCALEF=((RMEAN-RMIN)+(ZZZZZ))*550. !SC. MEAN TO CLR BAR
SCALEF=ININT(SCALEF)
93920 CONTINUE

IF (MEAN.GE.1.0*10.**6.) GOTO 83920
CALL GRFCDS (1,255,CHMEAN,458,ISCALEF,6,0,4)
83920 IF (RMAX.GE.1.0*10.**6.) GOTO 83921
CALL GRFCDS (1,255,CHRMAX,345,6,0,4)
83921 IF (RMIN.GE.1.0*10.**6.) GOTO 83922
CALL GRFCDS (1,255,CHRMIN,458,0,6,0,4)
CALL GRSBFD
83922 CONTINUE
9877 CONTINUE

IF (IVALUE.EQ.2) THEN
JKMN=465
ELSEIF (IVALUE.EQ.1.OR.IVALUE.EQ.3.OR.IVALUE.EQ.4 OR.IVALUE.EQ.6
1 OR.IVALUE.EQ.7) THEN
JKMN=463
ELSEIF (IVALUE.EQ.5) THEN
JKMN=470
ENDIF
CALL GRFAR (1,255,0.0,JKMN,ISCALEF,1,1)
CALL GRSBFD
95631 CONTINUE

CC
CC DRAW SCALE ONTO IMAGE
CC
RSCANDIST=FLOAT(SCANDIST)
PIXELS=(X4*1.)/RSCANDIST
[PIXELS]=ININT(PIXELS)
CALL GRFVCS (1,255,340,420,340+1PIXELS,420)
CALL GRFCDS (1,255,BSCALEMARK,345,410,6,0,4)
CALL GRSBFD

C1) FOR DRAWING INITIAL ZEROS FOR ATTENUATION COEFF. < 0.1
IF (IVALUE.EQ.2.AND.RRMIN.LT.1000.) THEN
CALL GRFCDS (1,255,BZERO,JKMN,0,6,0,2)
CALL GRSBFD
ENDIF
213 CONTINUE
201 CLOSE(6)

1 FORMAT(*', WHICH IMAGE TO VIEW [*:PDF IN ROTH.IMAG]?')
2 FORMAT(A)
3 FORMAT(*', Channel (0,2,3) ?')
4 FORMAT('$Description ?? ')
5 FORMAT(6)
6 FORMAT('A')
7 FORMAT(' Range = .A, .A )
8 FORMAT(' Mean = .A15, .A )
9 FORMAT(1)

FORMAT(' Description ?? ')

CALL GRSEND

CONTINUE

RETURN

END

SUBROUTINE COLORSUB (IVALUE,ICOLORBAR)

INTEGER NR(256),NB(256),NG(256),IARRAY(18)

DATA IARRAY/0,2,1,0,1,2,1,0,2,1,2,0,1,2,1,0/

IF (IVALUE.GT.10) GOTO 9876

CALL GRINIT_OLD

CALL GRSKST

CALL GRZFC(0,0)

CALL GRZCL(0,255,255)

CALL GRZON(0,1)

CALL GRSBFD

B=4

A=2

L=256

IL=L/2

JL=4*IL

DO 100 I=IL+1,JL

NB(I)=255

NG(I)=B*I

IF(NG(I).GT.255)THEN

NG(I)=255

ENDIF

100 CONTINUE

DO 200 I=IL+1,JL

BB=((B/(1-A))*L)/(A/(A-1))

NG(I)=255

NR(I)=0

NB(I)=BB

IF (NB(I).GT.255) THEN

NR(I)=255

ENDIF

200 CONTINUE
DO 300 I=JL+1,JL+IL
RR=(B*I)-(A*L)
NR(I)=255
NR(I)=RR
IF (NR(I).GT.255) THEN
NR(I)=255
ENDIF
300 CONTINUE
DO 400 T=JL+IL+1,L
GG=((-B)*I)+(B*L)
NR(I)=ZER
NB(I)=O
NG(J)=G
IF(NG(I).GT.2SS) THEN
NG(I)=255
ENDIF
400 CONTINUE
800 KC=O
NR(I)=O
NG(I)=O
NB(I)=O
NR(256)=255
NG(256)=255
NB(256)=255
889 CALL GRNWR(1,NB,IARRAY(1+KC),O:256,0)
CALL GRNWR(1,NG,IARRAY(2+KC),O,256,0)
CALL GRNWR(1,NR,IARRAY(3+KC),0,256,0)
CALL GRSBFD
CALL GRNBND
CALL GRNSBF
CALL GRNIN(0,0,0,0,0)
IF (ICOLORBAR.EQ.1) THEN
CALL GRNBND(0,0,0,0,0)
ENDIF
CALL GRBND
C IF (IVALUE.EQ.1.OR.IVALU.EQ.3.OR.IVALU.EQ.5.OR.IVALU.EQ.7) THEN
DO 1111 I=1,300
DO 1111 J=1,900
1111 CONTINUE
GOTO 950
950 KC=O
9876 RETURN
END

SUBROUTINE PLACEBAR (ICOLORBAR)
CC ----Draw colorbar at right side of screen----
INTEGER*2 NA(45)
IF (ICOLORBAR.EQ.1) GOTO 1010
CONTINUE
DO 1000 I=0,350
DO 2000 J=1,19
G=I*(254.0/350.0)
NA(J)=G
CONTINUE
CALL GRWILW(NA,410,1,10,0,0,0,1,1)
CALL GRSBFD
CONTINUE
I
CONTINUE
1010 CONTINUE
RETURN
END

SUBROUTINE SAVEBAR
INTEGER*2 BAR(6)
TYPE *, 'Creating and saving the color bar...'
TYPE *,
OPEN(UNIT=34, NAME='BARFILE', STATUS='NEW',
FORM='UNFORMATTED', BLOCKSIZE=1024)
CALL GRININ(1,0,0,0)
CALL GRNB(Y(1,1,1,1))
CALL GRSBFD
DO 2060 IBBEL=0,459,3
CALL GRRLW (410, IBBEL, 6, 1, BAR)
WRITE(34) BAR
CONTINUE
CALL GRSBFD
CLOSE(34)
RETURN
END

SUBROUTINE ERASE
M=4095
N=4095
CALL GRFER (M, N, 0)
CALL GRSBFD
END

SUBROUTINE COLORS (IVALUE, ICOLORBAR)
INTEGER*2 NR(256),NB(256),NG(256),IARRAY(18)
DATA IARRAY/0,2,1,0,1,2,1,0,2,1,2,1,0/;
INTEGER*2 NA(45)

IF (IVALUE.GT.10) GOTO 9876
CALL GRINIT_OLD
CALL GRSRST
CALL GRZFTC(0,0)
CALL GRZC(0,255,255)
CALL GRZON(0,1)
CALL GRZBF'D

B=4
A=2
L=256
IL=I/B
JL=A*IL

DO 100 I=1,JL
NB(I)=255
NR(I)=0
NG(I)=B*I
IF (NG(I).GT.255) THEN
NG(I)=255
ENDIF
100 CONTINUE

DO 200 I=IL,JL
BB=((B/(I-A))*I)+((A/(A-I))*L)
NG(I)=255
NR(I)=0
NB(I)=BB
IF (NR(I).GT.255) THEN
NR(I)=255
ENDIF
200 CONTINUE

DO 300 I=JL+IL+1,JL+II,
NR(I)=255
NG(I)=255
RR=((B*I)-(A*L))
IF (RR(I).GT.255) THEN
RR(I)=255
ENDIF
300 CONTINUE

RR=(B*I)-(A*L)

DO 400 I=1,JL+1,JL+II,
NR(I)=255
NG(I)=255
70
NB(I)=0
NG(I)=GG
IF(NG(I).GT.255)THEN
NG(I)=255
ENDIF
400 CONTINUE
800 KC=0
NR(I)=0
NG(I)=0
NB(I)=0
NR(256)=255
NG(256)=255
NB(256)=255
889 CALL GRNWR(1,NB,IARRAY(1+KC),0,256,0)
CALL GRNWR(1,NG,IARRAY(2+KC),0,256,0)
CALL GRNWR(1,NR,IARRAY(3+KC),0,256,0)
CALL GRSBFD
CALL GRNIN(0,0,0,0)
IF (1COLORBAR.EQ.1) THEN
CALL GRNBY (0,0,0,0)
ENDIF
CALL GRSBFD
DO 11 11
1=1,300
DO 11 11 J=1,900
GOTO 950
1111 CONTINUE
GOTO 950
950 KC=0
CC -------Draw colorbar at right side of screen-------
9876 IF (1COLORBAR.EQ.1) GOTO 1010
CONTINUE
DO 1000 I=1,0,350
DO 2000 J=1,19
G=I*(254.0/350.0)
NA(J)=G
2000 CONTINUE
CALL GRWLW(NA,410,1,19,0,0,1,1)
CALL GRSBFD
1000 CONTINUE
1010 CONTINUE
RETURN
END

Subroutine colorred(VALUE,ICOLORBAR)
INTEGER*2 NR(256),NB(256),NG(256),IARRAY(18)
DATA IARRAY/0,1,2,0,2,1,1,0,2,1,2,0,0,1,2,1,0/
INTEGER*2 NA(45)
10010 FORMAT(A)
10020 FORMAT(I)
NM256 = 0

IF (IVALUE.GT.10) GOTO 9876
CALL GRINIT_OLD
CALL GRSRST
CALL GRSBFD
L=256
IL=I/3
JL=2*IL
NM256=255
DO 100 I=1,IL
NB(I)=0
NR(I)=I*3
NG(I)=0
IF (NR(I).GT.253 ) NR(I)=253
100 CONTINUE
DO 200 I=IL+JL,JL
NR(I)=253
NB(I)=O
IF (NG(I).GT.253 ) NG(I)=253
200 CONTINUE
DO 300 I=JL+1,255
NG(I)=3*I-IL
NR(I)=253
NB(I)=3*I-IL
IF (NR(I).GT.253 ) NR(I)=253
300 CONTINUE
800 KC=6
NR(I)=0
NG(I)=O
NB(I)=0
NR(256)=NM256
NG(256)=NM256
NB(256)=NM256

840 CALL GRNWR(1,NB,IARRAY(1+KC),0,256,0)
CALL GRNWR(1,NG,IARRAY(2+KC),0,256,0)
CALL GRNWR(1,NR,IARRAY(3+KC),0,256,0)
CALL GRSBFD
CALL GRNIN(0,0,0,0)
CALL GRSBFD
CALL GRNBY(0,0,0,0)
CALL GRSBFD
C WRITE( 5,10860 )
C10860 FORMAT( ' $<CR>$ for next color scheme, Q to quit:' )
C READ( 5,10010 )ANS

72
C IF (ANS.EQ.'Q' .OR. ANS.EQ.'q') GOTO 9876
GOTO 9876
KC = KC + 3
IF (KC.GE.18) KC = 0
GOTO 840

CC ------- Draw colorbar at right side of screen -------

9876 IF (ICOLORBAR.EQ.1) GOTO 1010
CONTINUE
DO 1000 I = 0, 350
DO 2000 J = 1, 19
G = (254.0/350.0)
NA(J) = G
2000 CONTINUE
CALL GRWLW(NA, 410, 1, 19.0, 0.1, 1)
CALL GRSBFD
1000 CONTINUE
1010 CONTINUE

C 900 CALL GRSEND
return
END
What is claimed is:

1. A pulse-echo, immersion method for ultrasonic evaluation of a material, employing automatic scanning and digital imaging to obtain an image of a property of said material, wherein said material is held in a holding apparatus and material thickness are accounted for and eliminated, said method comprising:
   (i) ultrasonically scanning said material at a plurality of scan points and receiving the first and second echoes, each of which is a complete waveform, reflected off the back surface of said material and the first echo reflected off the front surface of said reflector both with and without the presence of said material;
   (ii) automatically scanning said material at said scan points to receive said first and second back surface echoes and said two front surface echoes using the information obtained in (i) above, so that each echo received from each scan point during said automatic scanning is centered within its respective time window;
   (iii) digitizing each echo received during said automatic scanning and determining the time delay between said two different reflector front surface echoes received at each scan point during said automatic scanning and calculating the wave velocity, using a cross correlation function, at each said scan point from
   \[
   v = c \left( \frac{\Delta t}{2} + 1 \right)
   \]
   where \(c\) is the speed of the ultrasonic wave transmitted in said liquid, and scaling the velocity values obtained in (iv) to corresponding proportional color or grey scale values and displaying the resulting image.

2. A method according to claim 1 wherein a single transducer is used.

3. A method according to claim 2 wherein said transducer is a high frequency transducer which emits a frequency between 1–100 MHz.

4. A pulse-echo, immersion method for ultrasonic evaluation of a material employing a single transducer, automatic scanning and digital imaging to obtain an image of a property of said material, wherein said material has a uniform thickness variation and is positioned in an immersion liquid between said transducer and an acoustic reflector, said method comprising:
   (i) performing a preliminary scan along both the x- and y-directions of the material to provide slant correction factors which are input into a computer to account for said nonlevelness and thickness variation during the subsequent automatic scanning for said material evaluation in (ii) below;
   (b) adjusting the time delay during said preliminary scan for any received echoes which are not centered in the scan time window, so that each received echo is centered in its time window;
   (ii) automatically scanning said material at a plurality of scan points in both the x- and y-directions to receive the first and second back surface echoes and front surface echoes with and without the presence of said material between said transducer and reflector using the information obtained in (i) above, so that each echo received from each scan point during said automatic scanning is centered within its time window;
   (iii) digitizing each echo received during said automatic scanning and determining the time delay between said first two successive sample back surface echoes, \(x\), and the time delay, \(\Delta t\), between the two different reflector front surface echoes received at each scan point during said automatic scanning and calculating the wave velocity at each said scan point from
   \[
   v = \frac{c}{2t} \left( \frac{\Delta t}{2} + 1 \right)
   \]
   where \(c\) is the speed of the ultrasonic wave transmitted in said liquid, and scaling the velocity values obtained in (iii) to corresponding proportional color or grey scale values and displaying the resulting image.

5. A method according to claim 4 wherein said back and front surface echoes received from the first and last scan points in both the x- and y-directions during said preliminary scan determine said time base adjustments needed for each echo to be centered within the time frame for it.

6. A method according to claim 5 wherein the location of the time window during said automatic scanning for said material evaluation is automatically adjusted via computer control by using the formula:

\[
W_{ij} = T_{ij}(X_{ij}Y_{ij}) = (X_{ij}Y_{ij})^{2}
\]

wherein \(W_{ij}\) is the correct delay time window at a particular scan location, \(T_{ij}\) is the time delay at the initial scan location, \(X_{ij}\) and \(Y_{ij}\) are the x- and y-directions scan correction factors, \(X_{ij}\) and \(Y_{ij}\) are the scan point numbers in the x- and y-directions, and \(X_{ij}\) and \(Y_{ij}\) are the x- and y-direction scan increments.

7. A method according to claim 6 wherein two scans are automatically made to obtain said first back surface echoes and said two different reflector front surface echoes.

8. A method according to claim 7 wherein said first two back surface echoes and said reflector echo with said material present are made in said first scan.

9. A method according to claim 6 wherein three scans are automatically made to obtain said first two back surface echoes and said two different reflector front surface echoes.

10. A method according to claim 9 wherein said first two back surface echoes are received in one scan, wherein said reflector echo with said material present is made in another scan, and wherein said reflector echo without said material present is received in yet another scan.

11. A method according to claim 9 wherein said first back surface echo is received in said first scan, wherein said second back surface echo is received in said second scan, wherein said reflector echo with said material present is made in said third scan, and wherein said reflector echo without said material present is received in said fourth scan.

12. A method according to claim 6 wherein four scans are automatically made to obtain said first two back surface echoes and said two different reflector front surface echoes.

13. An ultrasonic, pulse-echo, immersion method employing automatic scanning and digital imaging to obtain an
image of a microstructural property of a material positioned in an immersion liquid between a transducer and an acoustic reflector, said method comprising:

(i) automatically scanning said material at least three times at a plurality of scan points in both the x- and y-directions to receive the first and second back surface echoes and front surface echoes with and without the presence of said material between said transducer and reflector, each of said echoes received being a complete waveform and gated within a time window;

(ii) digitizing each echo received during said automatic scanning and determining the time delay between said first two successive sample back surface echoes, \(2\tau\) and the time delay, \(\Delta t\), between the two different reflector front surface echoes received at each scan point during said automatic scanning and calculating the wave velocity at each said scan point from

\[
v = c \left( \frac{\Delta t}{2\tau} + 1 \right)
\]

where \(c\) is the speed of the ultrasonic wave transmitted in said liquid, and

(iii) scaling the velocity values obtained in (ii) to corresponding proportional color or grey scale values and displaying the resulting image.

14. A method according to claim 13 wherein four scans are automatically made to obtain said first two back surface echoes and said two different reflector front surface echoes.

15. A method according to claim 14 wherein said first back surface echo is received in said first scan, wherein said second back surface echo is received in said second scan, wherein said reflector echo with said material present is made in said third scan, and wherein said reflector echo without said material present is received in said fourth scan.