CONTRACTOR REPORT ARCCB-CR-86011

ULTRASONIC IMAGING AND
AUTOMATED FLAW DETECTION SYSTEM

L. JONES
J. F. MC DONALD
G. P. CAPSIMALIS

RENSSELAER POLYTECHNIC INSTITUTE
TROY, NEW YORK 12180-3590

MARCH 1986

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT CENTER
CLOSE COMBAT ARMAMENTS CENTER
BENET WEAPONS LABORATORY
WATERVLIET, N.Y. 12189-4050

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED
DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The use of trade name(s) and/or manufacturer(s) does not constitute an official indorsement or approval.

DESTRUCTION NOTICE

For classified documents, follow the procedures in DoD 5200.22-M, Industrial Security Manual, Section II-19 or DoD 5200.1-R, Information Security Program Regulation, Chapter IX.

For unclassified, limited documents, destroy by any method that will prevent disclosure of contents or reconstruction of the document.

For unclassified, unlimited documents, destroy when the report is no longer needed. Do not return it to the originator.
An automated inspection system is described for detecting, imaging, and classifying flaws in cannon billets. The first phase of the project has produced a flaw detection system capable of triggering rapidly and reliably on flaws down to one-fourth inch in cross-section in billets up to 18 feet in length and 22 inches in diameter. This portion of the system has been optimized for low cost, high speed, simplicity, and maintainability utilizing...
7. AUTHORS (CONT'D)

L. Jones and J. F. McDonald
Rensselaer Polytechnic Institute
Center for Integrated Electronics
Troy, NY 12180-3590

G. P. Capsimalis
Armament Research and Development Center
Close Combat Armaments Center
Benet Weapons Laboratory
Watervliet, NY 12189-4050

20. ABSTRACT (CONT'D)

many commercially available components with a minimum use of one-of-a-kind electronic interface boards. It is a versatile, programmable stand-alone unit capable of scanning one billet every half hour. The second phase system is designed to considerably enhance the images of flaws which have been detected and located using the first phase system by providing a much larger ultrasonic array aperture and digitizing capacity. The resulting hardware is capable of both quasi-optic lens processing and digital holography.
# TABLE OF CONTENTS

1. Introduction  
   2

2. Description of Hardware  
   2.1 Overview of Hardware  
   2  
   2.2 Searle Imaging System  
   3  
   2.3 LSI-11 Microcomputer  
   7  
   2.4 Stepper Motor Driver  
   16

3. System Software  
   3.1 Software Motivation  
   19  
   3.2 Flaw Detection Algorithm  
   21  
   3.3 Thresholds  
   23  
   3.4 Memory Allocation  
   27

4. Mechanical Assembly  
   4.1 Transducer Cart Assembly  
   30  
   4.2 Linear Track Assembly  
   34  
   4.3 Stepper Motor Subsystem  
   37

5. System Operation  
   5.1 Mechanical Set-up and Adjustments  
   43  
   5.2 Computer Operation and Thresholds  
   51  
   5.3 Typical Scan of a Billet  
   66

6. Performance Testing and Results  
   6.1 Flaw Identification  
   71  
   6.2 Detecting the Shadow  
   72

7. Mathematical Study of Resolution  
   7.1 Aperture Limitations  
   78
7.2 Phase Two - Digital Holography 84

8. Image Enhancement by Deconvolution 87

8.1 Deconvolution 87

8.2 Hardware 93

8.3 Discussion of Deconvolution 98

9. Conclusions 99

10. Literature Cited 101

APPENDIX I ------- Software Listing------------- 102
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>System Block Diagram</td>
<td>4</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Photograph of System Cabinet</td>
<td>5</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Controller Register Definitions</td>
<td>9</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Input Module Block Diagram</td>
<td>11</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Display Module Block Diagram</td>
<td>13</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Photographs of Custom Modules</td>
<td>15</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Photographs of DEC Modules</td>
<td>18</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Flow Chart of Flaw Detection Algorithm</td>
<td>24</td>
</tr>
<tr>
<td>Figure 9</td>
<td>Memory Allocation</td>
<td>28</td>
</tr>
<tr>
<td>Figure 10</td>
<td>Transducer Cart</td>
<td>31</td>
</tr>
<tr>
<td>Figure 11</td>
<td>Mechanical Drawing of Cart</td>
<td>32</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Mechanical Assembly Overview</td>
<td>35</td>
</tr>
<tr>
<td>Figure 13</td>
<td>Pair of Jactuators</td>
<td>38</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Support of Stepper Motor</td>
<td>40</td>
</tr>
<tr>
<td>Figure 15</td>
<td>Ball Bearing Screw and Transducer Cart Assembly</td>
<td>42</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Accutrol 100</td>
<td>47</td>
</tr>
<tr>
<td>Figure 17</td>
<td>Placement of Rotational Switch</td>
<td>50</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Set Parameters Menu</td>
<td>55</td>
</tr>
<tr>
<td>Figure 19</td>
<td>Backwall Illumination Function</td>
<td>68</td>
</tr>
<tr>
<td>Figure 20</td>
<td>Photo: Resolution of Imaging System</td>
<td>74</td>
</tr>
<tr>
<td>Figure 21</td>
<td>Photo: Identification of Increasingly Smaller Flaws</td>
<td>75</td>
</tr>
<tr>
<td>Figure 22</td>
<td>Photo: Identification of Increasingly Smaller Flaws</td>
<td>76</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>23</td>
<td>Linear Phased Array Geometry</td>
<td>79</td>
</tr>
<tr>
<td>24</td>
<td>Deconvolution Architecture</td>
<td>95</td>
</tr>
</tbody>
</table>
PART 1

INTRODUCTION

An automated flaw detection system has been designed under contract with the Benet Weapons Lab of the U.S. Army. It is a fully automated system for use in detecting internal flaws in steel cannon billets. The goal is to locate and classify potentially hazardous flaws.

An image is created by sending a wave of ultrasonic signals into the steel and timing the arrival of acoustic reflections. The signals are sent and received through a phased array crystal transducer. The information thus gathered is digitized and stored in computer memory. An image can be derived from the collected data.

The system is designed around a commercially produced ultrasonic imager sold by Searle Ultrasound. An LSI-11 microcomputer is interfaced to the imager with custom designed modules. Ultrasonic image data is loaded into memories which can be examined by the LSI-11. The LSI-11 can then make decisions about the existence of flaws in the material.
PART 2

DESCRIPTION OF HARDWARE

2.1 OVERVIEW OF HARDWARE

The hardware for the first phase automated ultrasonic flaw detection system is centered around a SEARLE linear phased array ultrasonic imager, an LSI-11 microcomputer, and an assortment of custom-designed electronic modules. There is also a CRT display terminal for controlling the computer, and a stepper motor with associated drive electronics to position the imager's array. Ultrasound signals from the Searle unit's array are sent to an analog to digital converter, and the digital image information is stored into one of two framestore memories. The LSI-11 microcomputer can then enhance and analyze the image numerically. Details of this part of the process are explained in the software section of this report. The image data in memory, and any annotations added by the flaw detection program, are sent to a digital to analog converter and television timing generator. The result is a standard video signal which can be fed into a video monitor such as the monitor internal to the Searle unit. The resulting image is of better quality than the images generated by the Searle unit's internal imaging electronics, with the added advantage that the
image has also been analyzed by the computer for automated flaw detection.

A block diagram of the system appears in figure 1, and a photograph of the equipment is given in figure 2. The computer cabinet in the photograph consists of the Searle Ultrasonic Imager, a CRT display terminal, an LSI-11 microcomputer, and the Sigma Instruments stepping motor driver. Through the CRT terminal the operator can maintain control over each of these subassemblies to quickly and accurately detect flaws present in cannon billets.

2.2 THE SEARLE IMAGING SYSTEM

The Searle unit is a complete, self-contained, and easily maintainable linear phased array ultrasonic imaging system intended for medical applications. Its principal function is to handle the ultrasonic signal. The imager creates the signal and then sends it out to an array of piezoelectric crystals, also called the transducer array. It then switches modes and after a programmed delay will start to receive the ultrasonic reflection. This information is then used to produce an image for the video monitor on the front of the Searle unit.
SYSTEM BLOCK DIAGRAM

Figure 1
Figure 2. Photograph of System Cabinet.
The Searle unit has the capability of conditioning both the incident and the reflected signals which have been sent out and received by the transducer array. The parameters which control this conditioning could originally be set by means of several knobs on the front of the unit itself. To automate this process these knobs were disabled and the parameters were placed under computer control. In this way, once the optimum settings were experimentally determined they could be automatically programmed on power up, while still allowing for adjustments by the operator.

The Searle unit was designed to send out an ultrasonic pulse, receive its reflection, and produce an image for its video monitor without storing the image. This image can be seen when the switch on the left side of the case is in the internal video position. But when the unit is used directly on steel, the image produced is of very poor quality. The faster speed of sound in steel causes the image to be compressed in the vertical dimension, making good imaging impossible. For this reason, the ultrasonic signals from the unit are extracted and fed down to a custom designed analog to digital converter module (A/D) in the LSI-11. In digital form the image can be stored, enhanced, and analyzed for flaws. The LSI-11 then converts this information back into analog form.
and then sends it back up to the Searle unit for display. This analog signal is connected to the VIDEO IN terminal on the back of the Searle unit and can be seen when the switch on the left side of the case is in the external video position. An interface module has been constructed and installed in the Searle unit. It serves to buffer the analog and digital signals coming to and from the unit, as well as providing optional computer control of the front panel functions. The interface also controls the internal video monitor.

2.3 LSI-11 MICROCOMPUTER

I. CUSTOM DESIGNED MODULES

The LSI-11 is responsible for many tasks in the Flaw Detection system. These include controlling the Searle Imaging System, receiving, storing, and analyzing the data presented to it by the imaging unit, and creating a video signal which is sent back to the Searle unit's display. The computer also controls positioning the array by means of the stepper motor driver and several switches. These switches determine the limits to which the cart is allowed to travel.

To accomplish all of these tasks several custom designed modules had to be produced that would operate in conjunction with the necessary DEC original
equipment modules. There are four custom designed LSI-11 interface boards plus two others which interface to the Searle unit's internal bus, and the Sigma Instruments stepper motor driver. The four LSI-11 interface boards are described as follows:

1) CONTROLLER MODULE

This is a custom designed interface to the LSI-11 microcomputer which allows the software to have complete control of the various parts of the system. By accessing registers on this module, the program can control the acquisition of images, the gain parameters of the Searle unit, the mode of the monitor display, the handling of the framestore memories, and, in general, controlling the high speed internal data bus between the input, output, and framestore modules. A diagram of the registers and associated bit definitions is given in figure 3. Because much of the timing, amplification, display, and transmitter modules of the Searle unit are still used by the detection system, the servicing of the system is enhanced. For example, by disconnecting the interface cable from the LSI-11, the Searle unit behaves as if it was completely unmodified and could be shipped intact for repair.
CONTROLLER REGISTER DEFINITIONS

Figure 3
2) INPUT MODULE

The input module accepts the analog ultrasound image signals from the interface board in the Searle unit. An analog to digital converter on this module converts incoming image signals to digital form. Once enabled by the controller module, the input module sends the digital data out over the high speed data bus to the framestore module. Simultaneously, it generates the appropriate addresses and control signals for the memory in the framestore. Sampling rate and input signal attenuation are adjusted by software via the controller module in the LSI-11. A block diagram of the input module appears in figure 4.

3) DISPLAY MODULE

The display module is another custom designed circuit board whose function is to take the digital image data stored in the framestore and convert it into a form acceptable to a standard video monitor. Whenever the input module is not using the high speed data bus, the display module begins generating appropriate address and control signals for the framestore. Digital data read from the framestore is sent over the data bus to the display module where a digital to analog converter (D/A) creates an analog signal. Standard television synchronization is added.
Figure 4
to the signal and the result is a video signal compatible with any television monitor. This signal is fed to the Searle unit's internal video monitor where a picture of the material under test appears. Figure 5 contains a block diagram of this module.

4) FRAMESTORE MODULES

There are two framestore modules in the system which are always in opposite modes. One framestore is enabled onto the high speed data bus (here after called D-BUS) in order to accept data from the input module or supply data to the display module. The other framestore is enabled onto the LSI-11 processor bus (Q-BUS) so that the software programs may analyze the images and look for flaws. The framestores can be alternated between the Q-BUS and the D-BUS under program control. Both framestores are always active simultaneously on different buses. In this way, image acquisition and image analysis can be overlapped in order to make the most efficient use of time. In order to implement this sharing of memory, the framestores must have access to both buses.

Memory refresh is handled by the processor on the Q-BUS while refresh cycles are not necessary on the D-BUS. The D-BUS is always active, either with data acquisition in which the input module writes into the
DISPLAY MODULE

Figure 5
framestore, or with display operation, where read cycles continuously occur. This eliminates the need for refresh cycles while a module is selected onto the D-BUS.

Photographs of the custom designed boards and the stepping motor interface board appear in figure 6.

II. STANDARD O.E.M. MODULES

In addition to the four aforementioned custom digital cards, the LSI-11 incorporates the following four highly maintainable standard modules manufactured by Digital Equipment Corporation (DEC):

1) CPU MODULE

This module is a conventional LSI-11/2 processor card supplied by DEC. The processor executes the software which controls the entire system.

2) SERIAL LINE UNIT

This module is a model DLV-11J also supplied by DEC. It is a four port serial interface card with one port connected to the CRT terminal, another connected to the stepper motor controller, and the third is wired to a standard RS-232 connector on the
Figure 6. Photographs of Custom Modules.
Figure 6a. Framestore Memories.
back panel of the system rack. Any other computer or system can communicate with the system through this port on the back panel. The fourth port is not used on this card. The communications standard is RS-232. The baud rate can be changed by moving jumpers on the board.

3) RAM MODULE

This is a DEC standard memory card. It is used for storage of the various variables and constants used by the software, as well as a work space for computation of various parameters of the image.

4) PROM MODULE

This is also a DEC standard memory card. It is used for storage of the software. The contents of this memory are retained, even when power is lost. This enables the system to be ready to run whenever it is turned on.

Figure 7 contains photographs of the DEC boards.

2.4 STEPPER MOTOR DRIVER

In order to position the imaging array over various parts of the sample, a Sigma Instruments stepper motor driver is used. The driver is controlled by the LSI-11 via a serial line. A custom designed
interface is installed within the Sigma controller, to handle commands in the form of a direction and a stepping count. The interface then supplies a series of pulses to the Sigma stepper driver to move the motor a specified amount. When the motion is complete, the interface sets a ready flag to notify the LSI-11.
Figure 7. Photographs of DEC Modules.
PART 3

SYSTEM SOFTWARE

3.1 SOFTWARE MOTIVATION

The software for the system is permanently stored on programmable read-only memory chips mounted on the DEC PROM module. These programs are available to the user when the front panel switch is in the START position on power-up.

This is very important. If the front panel switch is not in the START position when power is turned on, the software will not be readily available to the operator.

The bulk of the software is the flaw detection algorithm. This algorithm has as its primary goal the rapid detection of any flaw within the frame. A flaw is represented by a high intensity reflection within the image of the steel. This reflection is caused by the impedance mismatch associated with a flaw or fracture. The change in density at the boundary causes part of the ultrasound wave to be passed through it and the rest of the wave to be reflected back from it. Because the change in density between steel and air is so great, most of the wave will be reflected back from the boundary.
These reflections, however, cannot be used as the primary feature on which the detection rests. Time is a limitation here. There are over 32,000 pixels in each image. Real time operations require the search to proceed at 1/30 second per frame. A real time search through this large array would involve much more time than is available.

Also, the detection of these high intensity reflections depends upon the orientation and characteristics of the flaw. If the flaw is at an angle with respect to the transducer array, the reflections may be scattered instead of reflected directly back at the imager. In this case the flaw is practically invisible to the imager.

Any flaw, without regard to its orientation or shape, will deflect some of the energy sent at it. This is used to advantage in the flaw detection algorithm.

All billets possess a smooth bore at the center. Because of this, in each image there will be reflections from the outer and inner walls of the billet. When an internal flaw or fracture dissipates some of the energy directed at it, there is a corresponding decrease in the amount of energy returned from the inner bore surface of the billet. Thus a flaw
casts a shadow in ultrasound on the inner bore surface. By inspecting the reflections from this surface, the presence or absence of a flaw can be determined by detecting these shadows. Confirmation of the flaw's existence can be made if its surface features are so oriented as to reflect directly back into the array aperture.

3.2 FLAW DETECTION ALGORITHM

The initial task of the program is to find the inner wall. The image of the wall is blurred by the long depth of focus and the resolution of the aperture in both length and depth directions. The task of finding the wall is accomplished by a fast scan at each recorded depth. The wall will be indicated by a line of high intensity reflection oriented horizontally. The program must have centered the image on the framestore through the internally programmable delay. If the inner and outer surfaces cannot be identified, due to misalignment or insufficient reflection, the program traps back to command mode.

Once the program has identified the inner surface, the testing for regions of diminution, or shadows, begins. The image of the backwall is blurred in depth and length. An averaged vector is created from the sampled backwall. At each sample point, the program
sums over the inner bore surface in the depth dimension. This gives a picture of the total energy reflected from the wall.

The blurring of the image in depth and length causes a corresponding blurring of the intensities as seen by the program scanning along the image of the wall. To reduce this effect, the program looks at groups of pixels at once. Averaging in this way does tend to filter the true data. However, if the image is clear enough in the length dimension, this method will preserve the shadows while reducing the inherent blurriness of the image.

These intensities thus extracted from the image are compared to a sample of the backwall with no flaws present. Two criteria are used in judging a flaw: a diminution in reflected intensity below the established threshold, and a minimum width for the shadow. Regions satisfying these criteria have their boundaries logged in an array.

The areas of the image above the shadow are then searched for possible secondary confirmation from the reflections from the flaw or fracture itself. The result of the search for each frame is reported out, and any flaws discovered are marked for display. Occasionally such confirmation is not possible due to
the orientation of the flaw. Distinction of two close		
targets may not be possible, but the flawed region		
containing both targets can be identified.

Figure 8 contains a flow chart of this algorithm.

3.3 THRESHOLDS

An important aspect of the algorithm in this		
system is the thresholds used in the various tests.
The values used are affected by several factors,
including the transmission gain, the size and		
composition of the sample, and the focusing of the		
array. These thresholds must be linked to the values		
obtained by the image, but must also reflect the		
choices of the operator. The best approach in this		
algorithm is to allow a greater possibility for false		
identification than for false rejection with respect to		
each pixel. This is optimal because the minimum width	
criterion can then be used to lower the false		
indentification error.

The best approach in a real time system for		
setting accurate thresholds involves sampling the		
unbiased image at the beginning of each run. The		
threshold for shadow detection can then be derived from		
the intensity function along the unbiased inner bore		
surface.
Figure 8b
3.4 MEMORY ALLOCATION

A diagram which describes how memory is allocated in the LSI-11 microcomputer can be found in figure 9. The system software which has already been mentioned is mapped into memory locations 0 through 17777. All addresses are in octal. These are the 4k of addresses the DEC PROM module has been set up to recognize.

The DEC RAM module is configured to recognize memory locations 20000 through 57777. Memory locations 20000 through 27777 are assigned as the program stack. Locations 30000 through 40000 are used for storing the values of variables and other partial results created by the software. The first task the software has when power is turned on to the computer is to copy all of the codes from the Read Only Memory chips into the Random Access Memory chips. These read-write memory chips recognize addresses 40000 through 60000 and it is from these addresses that the processor fetches instructions for execution.

The framestore which is connected to the LSI-11 processor bus recognizes addresses 60000 through 157777. Which framestore is actually connected to the Q-Bus is determined by one bit in memory location 160000. This register which is addressed as 160000 and three others which have addresses through 160006 are
MEMORY ALLOCATION MAP

FIGURE 9
physically located on the controller board and have been discussed earlier. There are four more sets of registers for each port of the DEC Serial Line Unit. Each port has a transmit and receive data buffer, plus a transmit and receive status register. These four device registers for any port are assigned to contiguous memory locations. The stepper motor device registers are at memory locations 176500-176506 and the CRT terminal device registers are at 177560-177566.
4.1 TRANSDUCER CART ASSEMBLY

The mechanical assembly is essentially a linear tracking system. The structure consists of two main subassemblies: the transducer cart assembly and the main support assembly.

The main structure of the cart is built around a pair of inverted A-frames. These frames, in addition to all the metal structures of the mechanical assembly, are constructed out of aluminum alloy. This was chosen because of aluminum's high strength to weight characteristic and its relative ease in machining. The device that holds the transducer in the cart consists of three plates: the clamping plate, the holding plate, and the nose plate. A photograph of the Transducer Cart Assembly appears in figure 10, and a mechanical drawing appears in figure 11. The nose plate is positioned at the bottom of the cart assembly. Besides being a main structural support for the overall A-frame, the neoprene cushioned slot in the center of the plate keeps the transducer vertically aligned. The holding and clamping plates are responsible for
constraining the transducer inside the cart. The holding plate has a tapered neoprene lined slot into which the transducer is pressure-clamped by the clamping plate. The pitch of the transducer can be controlled by varying the pressure applied by the wing nuts that hold the clamping plate in place on top of the cart. The holding plate is designed as a floating assembly within the cart. This design provides adequate control over both the pitch and yaw of the transducer, and also provides a good shock absorption system in the occurrence of a substantial momentum transfer to the transducer. This floating system is achieved by suspending the holding plate between two sets of compression springs. When the wing nuts are properly tightened, the nose of the transducer array will protrude from the bottom of the cart. The exact position of the transducer can be altered by increasing the amount of compression force that is applied to the top springs. This will also increase the damping force that is applied to the holding plate.

In order to obtain proper impedance matching as the ultrasonic signal passes from the transducer into the billet and back, the nose of the transducer is surrounded by a water bath. The water bath is contained within a clear Plexiglas shell. The Plexiglas shell is attached to the bottom of the nose.
plate with a metal flange. When the cart is lowered to the surface of the billet a rubber gasket clamped to the bottom of the water column provides a tight seal with the cannon billet.

The interface between the cart assembly and the linear tracking assembly is accomplished by the use of four open type pillow blocks. The pillow blocks are linear bearings that ride upon the shafts of the linear tracking assembly. Pillow blocks were chosen over conventional linear bearing assemblies because they will allow for a three degree deviation from the normal axial alignment without restricting linear movement. Open type pillow blocks are used so that support rails can be used for a more even load distribution in supporting the steel shafts along which the cart translates.

4.2 LINEAR TRACK ASSEMBLY

The linear track assembly is essentially the main support for the entire mechanical system. A drawing of the mechanical assembly appears in figure 12. The main elements of this structure are two 4 x 6 inch aluminum wide-flange beams. The beam structure is 25 feet in length and is not permanently attached to the testing table upon which the mechanical structure stands. The lack of rigid attachment is due to the necessity for
the beam structure to be removed while a billet is being loaded onto the table. This prevents the transducer from accidentally coming into contact with the billet while the billet is being loaded onto the testing table. This lack of rigid support necessitates that all bending moments that the structure is subjected to have to be eliminated or compensated for in the beam structure itself. With the cart in the middle of the beams the maximum deflection of the beam (this includes the deflection due to both the total load upon the beams and the bending moments that the beams are subject to because of the loading) will be a distance of .98mm. This stringent a tolerance is set in order not to cause any sacrifices in the data acquisition due to the transducer’s orientation as it translates down the billet.

Attached to the wide-flange beams are the structures that support the drive system for the cart. These include four A-frames and the additional bearings and mounting hardware associated with them. One of these A-frames supports the stepper motor while the other three house flange mounted bearings. These bearings provide added support to the ball screw which will only be supported at the ends and by the reactive force from its contact with the ball nut on the cart. These A-frame and bearing systems are bolted to the
wide-flange beams.

The remaining components of the mechanical assembly are the devices that hold the beam assembly above the test table on which the billet will be placed. Throughout the ultrasonic scanner's test life it will be used to test for flaws in billets ranging in size from 8 inches to 24 inches. To accomplish this, four machine screw jactuators are used. These jactuators are raised and lowered by turning a hand crank. These units are connected in pairs at either end of the test table. A photograph of two of the four units appears in figure 13. The jactuators allow easy control over the vertical positioning of the entire assembly with respect to the billet. The jactuators will be bolted directly to the test table provided by the Benet Weapons Lab. The use of these jactuators makes the job of correctly leveling the entire assembly significantly easier than if threaded rod and nut assemblies were to be used.

4.3 STEPPER MOTOR SUBSYSTEM

Of the many subsystems that make up the ultrasonic imager, there are two which serve to interface the two environments in which the system must function. The first of these systems, and of prime importance, is the ultrasonic array which has been dealt with earlier in
Figure 13. Pair of Jactuators.
this technical report. The stepper motor and its associated controls are the second main interface between the electrical and mechanical systems of the ultrasonic imager. The stepper motor provides an accurate method of positioning the cart/array over the billet.

Electrically the stepper motor system features two switches that indicate right- and left-hand travel limits and a third input which indicates the direction of movement of the cart/array over the billet. Mechanically the stepper motor is mated to an A-frame which is part of the cart/array support system. A photograph of this A-frame appears in figure 14. The stepper motor shaft is joined to a ball screw via a flexible disc coupling; this serves three purposes. Initially, the flexible coupling is used to smooth the torque conversion between the stepper motor and the ball screw. In this fashion the cart moves smoothly and maintains itself correctly aligned with the billet. The coupling also serves to correct potential misalignment with quick starting and stopping of the cart.

A ball bearing screw is used as the drive for the cart because it replaces the sliding friction associated with conventional power screws with the lower rolling friction of ball bearings. A photograph
of the ball bearing screw and the transducer cart assembly can be seen in figure 15. The ball screw transfers the rotational motion associated with the stepper motor into translational motion of the cart. The lower amount of friction due to the ball bearing provides a more efficient method of powering the cart. This system will move the cart so that the billet can be easily scanned along its total length.
Figure 15. Ball Bearing Screw and Transducer Cart Assembly.
PART 5

SYSTEM OPERATION

5.1 MECHANICAL SET-UP AND ADJUSTMENTS

The mechanical set-up for the Ultrasonic Imaging and Automated Flaw Detection System can be described in three major steps, each of which has several adjustments and cautions which must be considered while the set-up is taking place.

A. Placement of Billet

It has been discovered that the cannon billets do not possess a perfectly round outer surface so it is safe to assume that the level of water in the bath surrounding the transducer will be changing as the billet rotates. To allow for this a catch basin must be placed below the entire section of the billet which is to be scanned. Adjust the rollers to safely support the cannon billet while leaving enough room below the billet for a basin to catch any water which might leak out of or overflow from the water bath for the transducer array. There is a crank on each roller assembly to allow for the operator to move the rollers to the desired position. Once the rollers are in the appropriate position, place the cannon billet on the
rollers and mount the triangular stop at the south end of the billet. The south end of the billet will be considered the end which points towards the band saws at the Watervliet Arsenal, and the north end will be the end which points towards the forge.

B. Mounting of the Jactuators

Four machine screw jactuators have been chosen for the purpose of supporting the Linear Tracking Assembly (LTA) and keeping it in a fixed position. As mentioned earlier in this report, the four jactuators are connected in pairs by means of a flexible disc coupling. To assemble these main supports the coupling must first be slid back out of the way. Place the jactuators into the holes which have been cut out of the table, and line up the crank shafts on each pair of jactuators. Now slide together the flexible disc coupling and secure each half to its crank shaft. Each jactuator must be securely bolted to the table to ensure the LTA remains rigid.

C. Mounting the Linear Tracking Assembly

Proper positioning of the array relies only upon the assumption that the LTA be level. The same assumption was made about the top surface of the cannon billet as it sits on the rollers. Adjusting the
assembly in the length direction is not a problem as each pair of jactuators can always be raised or lowered in tandem. But each jactuator must first be adjusted so it is level with its mate before the LTA can be attached.

First, raise all four jactuators to a height that will not allow the transducer cart to come in contact with the billet when the LTA is lowered. Adjust each pair of jacks by placing a level across the top of both jacks and then raise or lower them individually as required. Each jack can be adjusted separately by holding the flexible disc coupling while rotating the top plate of the jactuator. Once each pair is level, the Linear Tracking Assembly can be lowered onto the jactuators. Be careful to align the pins on the bottom of the LTA with the large holes on the top of the jactuators. The LTA should be oriented with the stepper motor at the north end of the table.

D. Accutrol 100

During installation of the system, it was discovered that the rollers which are mounted on the table supplied by Benet Weapons Laboratory turned the cannon billet faster than had been expected. In order to slow down the AC induction motor which powers these rollers, the Accutrol 100 motor control was purchased
from Westinghouse Corporation. A photograph of this unit appears in figure 16.

The Accutrol 100 Adjustable Frequency Control uses solid state technology to adjust both the frequency and the voltage applied to the terminals of 3-phase AC motors. Since the speed of an AC motor is proportional to the applied frequency, this allows the speed of the motor to be changed either above or below its rated nameplate speed.

While the Accutrol 100 is capable of substantial speed adjustment the operator must be aware that the AC motor which is being controlled is a self-cooling motor. Therefore, at lower speeds the shaft mounted fan provides reduced cooling. Care must be taken not to overheat the motor. It was determined that an initial speed setting of 3 provided a slow enough speed to accurately detect flaws. Higher speeds will provide less accurate flaw detection while lower speeds will cause the motor to heat up quicker.

E. Adjustment of Jactuators

The second adjustment of the jactuators can now be made. Lower the LTA until the rubber gasket at the bottom of the transducer cart comes in contact with the cannon billet. Place the submersible water pump in the
basin of water below the billet and place the outlet hose into the hole provided at the south end of the cart assembly. Now adjust the jacks until the LTA is level in the length direction.

The final adjustment which must be made is the water level in the bath created by the transducer cart. This will only be a problem if the outer surface of the cannon billet is not perfectly round. If the billet is round then the water bath need only be filled once and no water will leak out. If it is not perfectly round then the height of the LTA, and the water pressure from the pump must be adjusted to maintain an adequate column of water.

The LTA should be lowered until it creates a good seal for most of the billet's revolution. But care must be taken not to lower the transducer cart to the point where friction from the billet will damage the clear Plexiglas shell at the bottom of the transducer cart. The flow of water should now be increased until the nose of the transducer is always surrounded by at least a half inch of water. If the water pressure is too great for the portion of the revolution when there is a good seal between the cart and the billet, the LTA may have to be lowered and the water pressure cut back. But once again, care must be taken not to lower the cart to the point where damage may result to the lower
portion of the cart.

F. Electrical Interface

The final step involved in setting up the mechanical system is to connect it to the rest of the Flaw Detection System. Connect the large orange cables to the electrical outlet mounted on the north end of the Aluminum I-Beams. Note the proper orientation of the cables. If they are reversed damage to the stepper motor may result.

There are three switches which must be attached to the mechanical system. Two of the switches will sense when the transducer cart has reached either end of the path which it is allowed to travel, and the third senses when a revolution of the billet has been completed. The two limit switches should be placed at the ends of the path you wish to scan and the third switch should be attached at the north end of the billet as seen in figure 17. The two limit switches should be placed with the arm of the switch pointing away from the transducer cart and the third switch should be placed with its arm pointing in the direction the billet rotates. A small bolt must be attached to the end of the billet for proper movement of this switch.
Finally, the transducer array must be placed in the transducer cart. The transducer array is very fragile and must be handled with extreme care. Remove the clamping plate from the top of the transducer cart assembly. The transducer array should be placed into the cart from the top with the cable pointing towards the north and then looping around and coming out of the top of the cart assembly. Press the transducer into the neoprene lined slot until the nose protrudes through the bottom. Replace the clamping plate and tighten the wing nuts securely. The orientation of the array may now be adjusted as necessary.

5.2 COMPUTER OPERATION AND THRESHOLDS

To begin using the Flaw Detection System, the switch on the front of the Computer Cabinet must be placed in the START position. If this is not done, the software will not start up when the power is turned on. With the switch on the front of the stand in the START position, the power switch on the back of the stand may now be turned on. You will notice 3 small red indicators on the back of the stand light up. This merely shows that power has been applied to all portions of the system. As you walk around to the front of the stand and the CRT terminal warms up, you
will see operator instructions and information printed on the screen.

INSTRUCTIONS FOR SYSTEM POWER UP
1 - TURN ON SWITCH ON FRONT OF STAND
2 - CONNECT TV AND STEPPER CABLES
3 - POWER UP TV AND STEPPER MOTOR

The switch on the front of the stand should now be placed in the ON position (down). The stepper cables are the large orange cables which connect the stepper motor driver to the stepper motor. If the entire mechanical system was just set up these cables were already connected as were the limit switches and probe. In any event, make sure these cables are connected before the stepper motor driver and the Searle unit are turned on. To continue with the instructions press any key on the CRT terminal.

The computer will then begin to initialize itself. For instance, the transducer cart will automatically move to the home position at the right limit switch. The computer will then ask the operator a series of questions. The first question printed on the screen will be:

ENTER DISPLACEMENT IN INCHES (0 TO 216)

This displacement provides the operator with the option of beginning the scan from a particular distance away from the home position. For instance, if earlier one day you saw some flaws one foot from the right limit
switch and you want to look at them again you can enter a displacement of 12 inches and the transducer cart will automatically go to that position. From then on this position will be considered the home position. If you desire to leave the home position at the default position of the north switch then a <CR> or 0 may be entered. The next message that will be printed on the screen will be:

ENTER BILLET LENGTH IN FEET (1 TO 18)
NOTE: LENGTH + DISPLACEMENT CAN NOT BE OVER 18 FT.

Now enter the length of the billet you wish to scan. Movement of the array is limited to 18 feet so the sum of the displacement and the length of the billet can not exceed this distance.

Now that you have provided the computer with information about the billet which is to be scanned, it will begin to time five rotations of the billet. It will average these five numbers and use this time as a reference to calculate the current angle of the billet. When five full revolutions of the billet are complete, the computer will go into command mode. The operator now has complete control over the imaging system to exercise whatever portion he desires. The options available to the operator are:
FIRST MENU COMMANDS:
(S) - SET PARAMETERS
(F) - ALIGN PROBE
(C) - CLEAR FRAMESTORE
(D) - CHANGE DISPLAYED FRAMESTORE
(F) - FIND FLAWS
(Q) - SYSTEM SHUT OFF
(M) - EXTRA ROUTINES

ENTER COMMAND:

Select one of these options by striking the associated letter.

(S) - SET PARAMETERS

Select this option to examine or alter one of the gains or thresholds used by the system. The command choices will be replaced by a list of parameters which is shown in figure 18. The number in parentheses which appears after each parameter is the current value.

A parameter is selected for change by entering the letter listed before the parameter. The parameter and its current value will appear at the top of the screen. If you do not wish to change this parameter, return without entering any numbers. To change a parameter, enter the new value, then return. The new value will appear in parentheses next to the parameter name.

Exit this program option by striking RETURN. The program will then revert to command mode and present the command choices again.
PARAMETER CHANGE ROUTINE

CHOOSE THE PARAMETER YOU WISH TO OPEN
OR EXIT THIS PROGRAM BY STRIKING <CR>

PARAMETERS:
(A) SAMPLING RATE (002)
(B) TRANSMIT GAIN (020)
(C) RECEIVER DELAY (020)
(D) NEAR RAMP GAIN (014)
(E) RAMP SLOPE (114)
(F) RECEIVER ATTENUATION (001)
(G) SYSTEM NOISE THRESHOLD (002)
(H) SHADOW DETECTION THRESHOLD (000)

Figure 18. Set Parameters Menu.
The parameters and their functions are summarized below:

**SAMPLING RATE** - controls the rate at which the ultrasound is sampled. In terms of its affect on the image, it determines the size of the image field. The range of values is 1, 2, or 3. One will give the slowest sampling rate, hence the largest field; three will give the fastest sampling rate, hence the smallest field of vision.

**TRANSMIT GAIN** - corresponds to the transmit gain setting on the front of the Searle unit. The gain is measured in dB down from the maximum, so the lowest number yields the strongest transmitted signal, and therefore the strongest reflected signal. The range of values is from 1 to 30.

**RECEIVER DELAY** - is the wait time between the transmission of the ultrasound pulse and the beginning of sampling. Increasing the delay will cause the image to move up on the TV screen. Adjust the value so that the billet is centered on the screen, with both the front and back walls visible.

**NEAR RAMP GAIN** - The amount of amplification the receiver provides varies as a function of time. This
function is in the form of a ramp which starts out at some low value in the near field (the top of the TV screen) and then ramps up to the maximum as time increases. The reason for this is because the intensity of the reflected signal tends to decrease from farther into the sample. To allow for this, and to provide for a better image, the amount of amplification is increased for signals coming from farther into the sample. The near ramp gain is the amount of gain which will be applied to signals near the surface of the sample.

RAMP SLOPE - is the rate at which the receiver gain will increase with time.

RECEIVER ATTENUATION - allows a tenfold reduction in the returned image intensity. This should be left as initialized by the program.

SYSTEM NOISE THRESHOLD - is used to compensate for the ordinary noise present in the image. This value should fall between 1 and 4 under normal conditions.

SHADOW DETECTION THRESHOLD - is used to allow flexibility in the setting of a threshold for flaw detection. Increasing the value will make detection easier and increase the chance of false alarms.
Decreasing the value will reduce the false alarm error rate, but at a cost of possible flaws missed. The proper setting of the flaw detection threshold with respect to the sampled image is most important to the actual implementation of the system in the manufacturing environment.

(P) - ALIGN PROBE

The probe must be adjusted until it is parallel with the billet. A graph will appear on the TV screen showing the backwall illumination function. In addition, a message will be displayed indicating the row numbers where the backwall image appears. The illumination function should be adjusted to be as flat as possible. The row numbers correspond to the location of the backwall as seen using the sampling program. There are 256 rows on the TV screen.

Return to the command mode by striking RETURN.

(C) - CLEAR FRAMESTORE

This option will clear the framestore which is currently displayed on the screen. The program returns to command mode automatically.
(D) - CHANGE DISPLAYED FRAMESTORE

This is used to look at one of the framestores, or to bring one onto the screen so that it can be cleared. This may be necessary if an unusual event causes an unwanted image to appear on the screen. A prompt appears when this option is entered. Return to the command mode by striking RETURN.

(F) - FIND FLAWS

This option is the most powerful of all, and comprises the bulk of the software. When this section of the software is entered the system will first align itself with the zero angle mark on the billet and then begin to scan the entire section of the billet specified on power up. It will continuously take samples and check them for flaws. When a flaw is found, its displacement from the home position and angle will be logged and the total number of flaws found will be displayed upon completion. The exit from this routine will place the operator in the second command mode. The second command level has the capability to go back and examine a portion of the billet where a shadow has been cast on the backwall. It then allows the operator to move the array by small amounts to get a better image of the flaw. The second command level also allows the operator to look at the list of all the
flaws that were located and recorded.

(Q) - SYSTEM SHUT OFF

When you desire to shut off the system, these instructions should be followed:

(M) - EXTRA ROUTINES

Selecting this option will make another command level available with routines which the operator may find useful. Here are the names of the routines which will be printed on the screen:

EXTRA MENU COMMANDS:
(C) - SINGLE SHOT FLAW DETECTION
(D) - REPEATED FLAW DETECTION
(U) - UPLOAD FRAME STORE
(Q) - SYSTEM SHUT OFF
(F) - EXIT TO FIRST MENU
(S) - EXIT TO SECOND MENU
(H) - MOVE MOTOR TO THE HOME POSITION
ENTER COMMAND:

These commands are not imperative for the proper functioning of the system, but merely conveniences which the operator may find useful. None of these command choices are extremely complicated and the functions are fairly obvious from their names:

(C) - SINGLE SHOT FLAW DETECTION
This option will take a single image from the Searle Unit and analyze it for flaws. The sample will be taken at a random position, whatever position the billet happens to be in when the option is selected. If any flaws are found in this image the flaws will be marked on the TV screen but not placed in the flaw log.

(D) - REPEATED FLAW DETECTION

This option will provide the operator with a continuous scan of the billet. It will take a sample at random, analyze it, report the results, and then repeat the process over and over until the operator presses RETURN. There is no movement of the array under this option, so the scan will encompass only the portion of the billet which happens to pass under the array.

(U) - UPLOAD FRAMESTORE

The system also has the capability to send a copy of the image in digital form to another system. This could be useful for more detailed analysis, to store an image permanently, or perhaps to display an image on a more sophisticated video terminal.

(Q) - SYSTEM SHUT OFF
This is the same option described earlier under the first menu commands.

(F) - EXIT TO FIRST MENU

This command is self-explanatory; it will send you to the first command level which has already been explained.

(S) - EXIT TO SECOND MENU

This command will send the operator into the command level which allows him to view a specific flaw. If the find flaws routine has not yet been run, then no flaws will have been logged and exercising the second command level will not yield any information. The second command level is described in more detail later in this report.

(H) - MOVE MOTOR TO THE HOME POSITION

Selecting this option will move the transducer cart assembly to the home position selected on power up. The same command choices will then be presented to the operator again.
The second menu commands give the operator the capability to go back and look at the flaws that were located under the find flaws routine. The operator can then assure himself whether or not the image contains a flaw or not. The second menu also allows the operator to look at a list of all the locations where shadows were found on the backwall. The operator can then use this list to determine which specific flaws he wants to look at. Here are all the options which will be presented to the operator under the second menu:

SECOND MENU COMMANDS:
(S) - SHOW FLAW LOG
(L) - LOOK AT FLAWS
(C) - CLEAR FRAMESTORE
(D) - CHANGE DISPLAYED FRAMESTORE
(Q) - SYSTEM SHUT OFF
(M) - EXTRA ROUTINES

ENTER COMMAND:

(S) - SHOW FLAW LOG

The flaw log is the list of locations where shadows were found on the backwall of an image. The format of the flaw log gives the operator a flaw number and the position of the flaw in terms of a distance from the home position and an angle away from the reference position.

(C) - CLEAR FRAMESTORE
(D) - CHANGE DISPLAYED FRAMESTORE
(Q) - SYSTEM SHUT OFF

63
EXTRA ROUTINES

These four options have been discussed earlier in this report.

LOOK AT FLAWS

Once the number of a particular flaw has been determined by looking in the flaw log it can then be displayed using this option. When this command is selected, the operator will be prompted for the number of the flaw:

ENTER FLAW NUMBER OR <RETURN> TO EXIT

When a valid flaw number is entered, the system will move the transducer to the correct position along the length of the billet and then align itself with the zero angle position of the billet. When the system finds that it is at the correct angle around the billet, it will take a sample and once again analyze it for flaws. When a shadow is found and a flaw located, the flaw will be highlighted with a box on the TV screen. While the operator is looking at the flaw he also has the capability to reposition the array by small amounts to get an even better picture of the flaw. Here are the options the operator will have when he is viewing a specific flaw:
LOCATE MOVEMENT INSTRUCTIONS:
(S) DO ANOTHER OR EXIT
(H) TO MOVE LEFT
(L) TO MOVE RIGHT
(K) TO MOVE UP
(J) TO MOVE DOWN
(P) DISPLAY PRESENT LOCATION

(S) DO ANOTHER OR EXIT

To exit this mode of operation or to look at another flaw in the flaw log this option should be selected.

(H) TO MOVE LEFT

This option will move the transducer cart assembly one-quarter inch to the left.

(L) TO MOVE RIGHT

This option will move the transducer cart assembly one-quarter inch to the right.

(K) TO MOVE UP

This option will increase the angle at which a sample is taken by a fraction of a degree. This fraction is determined after the rotation of the billet is timed on power up.

(J) TO MOVE DOWN
This option will decrease the angle at which a sample is taken by a fraction of a degree.

(P) DISPLAY PRESENT LOCATION

Select this option to display the location at which the system is currently taking samples. If the operator moves the position of the array from the location listed in the flaw log it will be reflected here. This will not however change the location which has been entered in the flaw log.

5.3 THE TYPICAL SCAN OF A BILLET

At this point we have discussed setting up the entire mechanical system and the operation of all the computer software in great detail. Because the system was described in such a detailed manner, we now give a simplified description of how the system can be used.

First, the mechanical system must be in place. If it is not or you wish to mount a new billet then the directions described in section 5.1 must be followed.

Once the mechanical system is set up you can proceed to the second part of the system's operation -- the LSI-11 microcomputer. With the switch on the front of the stand in the START position turn on the power switch at the rear of the stand. Instructions will
appear on the CRT terminal as it warms up. Follow these instructions and make sure to never connect the large orange cables while the power switch on the Sigma stepper motor driver is turned on.

After you have completed the power up instructions, you will have to describe what portion of the billet you wish to scan. You will be asked for a displacement (where you want the scan to begin) and how much of the billet you wish to scan. The system will then initialize itself. It will clear both framestores and move the transducer cart to the home position.

Now the complete system will be available to the operator. The first adjustment which needs to be made is to align the probe. When the align probe routine is run, a graph will appear on the TV screen. This curve is a measure of how well each part of the backwall is illuminated. One such curve appears in figure 19. Adjust the probe until a reasonably flat curve is displayed on the TV screen. This curve will never be perfectly flat due to variations in the gain of individual crystal elements, but it is still a good measure of the adjustment of the probe. A second method to check the alignment of the probe is to run the repeated flaw detection routine in the menu of extra routines. You will want the top of the billet and the inner bore surface to be parallel to the face
Figure 19. Backwall Illumination Function.
of the array. The picture on the TV screen is in reference to the probe so you can check this alignment by making sure these reflections are as horizontal as possible on the TV screen. Another feature you will want to check is how often the software detects flaws. If it constantly reports that shadows are present in the backwall, then further adjustment is necessary. Perhaps you will want to raise the transmit gain to get a stronger reflection with less shadows, or you could also increase the receiver ramp gain to provide you with more amplification in the receiver.

Once you have centered this image on the TV screen and you are satisfied that the system is accurately reporting when flaws are present you are ready to go back to the first menu and scan the billet. When the find flaws routine finishes it will report back how many possible flaws it has found and then send you into the second menu of control commands to examine the flaws.

Now you can choose which portions of the billet you wish to reexamine by looking into the flaw log. You can go to any of these locations by using the look at flaws routine. The operator can adjust the position of the array in order to convince himself whether or not a flaw is actually present.
This would conclude a typical scan of a billet. The power down instructions should be printed on the screen and followed to ensure that the next time the system is powered up the software will start automatically.
6.1 FLAW IDENTIFICATION

The system was tested on a section of an actual billet. Several screw holes were drilled in the billet. One hole was drilled into the outer surface, two in from the edge at different depths, and two up from the inner surface.

Early testing gave unconvincing results. The identification process broke down in the upper half of the image due to interference from the interface. Modification of the program to test the background resulted in improved detection. One of the flaws which was at a depth of about four inches was now detected, but the top two inches were still opaque. This occurred because the ringdown from the interface was full-scale; it completely masked the flaw image.

It was determined that the ringdown was primarily because of the interface between the probe and the couplant. The probe was then suspended two inches above the billet. This separated the image from the ringdown. Adjustments of the transmit gain were made in an attempt to offset the attenuation of the two-inch gap. This attenuation was found to be severe because a
water couplant was used. Later it was determined that glycerine, which has an acoustic impedance much closer to that of the steel, made a superior couplant.

With the software refined and the probe separated from the billet by the two-inch gap of glycerin, the system identified 1/4" screw holes at all depths in the billet.

6.2 DETECTING THE SHADOW

The width and relative intensity of the shadows were found to vary with depth as well as size of flaw. This effect has been attributed to the beam spreading inherent in the imager which acts like a fixed focal depth lens. The focusing is at its best near the outer surface of the billet. The shadow cast by a flaw near the outer surface is narrow and sharply defined. The intensity of each pixel in this shadow is one-third or less of the average along the trailing wall. A flaw located near the back wall is less well defined. The shadow from this flaw is spread across the image from the inner wall. The intensity drop is less pronounced; still it was found to be sufficient for reliable detection with appropriate threshold settings.
Extensive experimentation with the thresholds and sampling window along the inner wall produced a program that identified flaws at all depths with a minimum of false hits.

The program was also used to test the resolution of the imager and the software. Despite the smearing effects of the beam spread near the inner wall, good separation between the backwall and a flaw can be achieved as shown in figure 20.

Another experiment was set up using an actual billet with progressively smaller holes drilled into it. Figures 21 and 22 show slightly different views of this section of the billet. The bright streaks in these pictures are due to the fact that when the ultrasound signal hits the top of the flaw the entire signal is not reflected back to the array. Some of the energy continues into the flaw and bounces around between its two boundaries before returning to the array. These reflections will of course arrive at the array later in time, hence lower on the screen. Figures 20 and 21 show a 3/16 inch hole to the left, a 1/8 inch hole to the right, and a 1/4 inch hole in the center. The smallest hole detected had a thread diameter of 3/16 inch.
Figure 20. Photo: Resolution of Imaging System.
Figure 21. Photo: Identification of Increasingly Smaller Flaws.
Figure 22. Photo: Identification of Increasingly Smaller Flaws.
Theoretically, the pdf of the samples along the inner wall is Gaussian-distributed in the absence of a flaw. In practice, the unbiased intensity function along the inner bore surface will not follow a Gaussian distribution. There will be some systematic error due to variations in the crystal array elements and their associated circuitry. This effect can be seen in the Backwall Intensity Function of figure 18.

When a flaw is present, a second independent distribution is produced. The superposition of these two distributions can result in a region of uncertainty where they overlap. This region is largest for flaws located near the center bore.

This uncertainty can be minimized in several ways. Optimal setting of the threshold between the image and its flaw will minimize error, but any practical threshold selector will only approximate this boundary. The focusing can also be optimized, but there are fundamental limits on the spreading of the beam. A convolver operating on the inner wall image can also cause great improvement in the sharpness of the boundaries. In software, the search can be made using groups of pixels at once. Although this acts as a low pass filter which tends to blur the image of the shadow, it improves the chances of finding a group of low-value pixels located directly under the flaw.
PART 7

MATHEMATICAL STUDY OF RESOLUTION

7.1 APERTURE LIMITATIONS

Compensation for the beam spread function by deconvolution as discussed in part 8 is a useful augmentation for the phase one flaw detection system. Nevertheless, problems associated with system noise, analog part nonlinearities and dynamic range limit the usefulness of the deconvolution approach for aperture compensation. Therefore, phase two hardware is designed to address the limitations imposed by the size of the phased array, primarily by increasing the length of the array and the number of array elements.

Consider the geometry of the linear phased array shown in figure 23. Assume that the excitation in the rectangular aperture is focused only in the x direction towards the origin 0. The field at a point x, y, z near the focus 0 can be written as follows using the Fresnel approximation [3,4]:

\[
E(x,y,z) = \frac{IA}{L^2(L+Z)} e^{-ikz} \int_{-W}^{W} \int_{-D}^{D} \exp \left[ \frac{ik}{2L} \left( x'^2 + y'^2 \right) \right] \\
. \exp \left[ \frac{-ik}{2(L+Z)} \left( \left( x-x' \right)^2 + \left( y-y' \right)^2 \right) \right] \, dx' \, dy'
\]
LINEAR PHASED ARRAY GEOMETRY

Figure 23
where $A$ is the source strength and $k = \frac{2\pi}{\lambda}$ is the wave number. Restricting attention to the plane $y = 0$, we can explore the behavior near the focal spot due to the x-axis focusing.

Furthermore, let us first restrict $x$ to 0 to study the behavior of the focusing as a function of $z$, the perpendicular distance from the focal plane to the observation point. Then we find that

$$E(0,0,z) = \frac{IA}{\lambda L(1+z)} \sqrt{\frac{2L(1+z)}{kz}} e^{-ikz} \left[ \frac{1}{D} \left( \sqrt{\frac{kz}{2L(1+z)}} \right) \right]$$

where $\{ \frac{1}{D}(.) \}$ is one of the basic Fresnel integrals.

Similarly, by taking $z = 0$ we can explore the behavior of the focusing as a function of the lateral direction, $x$. Using similar mathematical manipulations we find:

$$E(x,0,0) = \exp \left\{ \frac{iKx^2}{2L} \right\} W \text{sinc} \left( \frac{Kxw}{L} \right)$$
where \( \text{sinc}(x) = \sin(x)/x \). The lateral width, \( s \), of the focal spot is roughly

\[
S = \frac{\lambda \pi}{W_k} = \frac{\lambda}{2} \frac{L}{D}
\]

as can be seen from the argument of the sinc function. The length of the focal spot is given by

\[
l = \frac{\lambda}{\pi} \alpha^2 \left( \frac{L}{D} \right)^2
\]

Clearly, the larger the numerical aperture, \( D/L \), the smaller the spot size will be in both length and width. Small spot width will be good for quasioptical signal processing such as is employed in the pulse-echo phased array of the phase one system, provided the spot length is not too small compared with the volume to be scanned since that system does not possess a dynamic focal depth capability. Depth information in the phase one system is determined by time of flight since the system is pulsed.
In the unfocused or $y$ direction, the spot confinement is also of interest. In general, the $y$ diffraction integral does not simplify mathematically to sufficiently give accurate beam width in the near (Fresnel) field. It can best be understood by examining the Cornu spiral. In basic terms, however, with no focusing along the $y$ axis the extent, $e$, of the spot in that direction will be at least as large as the width of the aperture's geometric shadow plus an amount due to diffraction fringing. Hence, we have approximately a width equal to the aperture size, $2W$, plus an estimated width taken from Fraunhofer considerations:

$$e = 2W + 2\frac{\lambda L}{W}$$

in the focal plane.

Now consider the following operating parameters:

$$\lambda = 0.6 \text{ mm}$$

$$W = 1.0 \text{ cm}$$

$$D = 1.0 \text{ cm}$$
These values are chosen to represent 4.0 MHz operation in steel, using only 16 elements at a time (as is the case in the phase one system). We find that the phase one beam has the following parameters:

\[ s = 6.0 \text{ mm}, \quad l = 19.2 \text{ cm}, \quad e = 4.4 \text{ cm}. \]

This shows that the focal spot will be very large regardless of how the beam is formed simply due to aperture considerations. Nevertheless, the large focal spot is not a serious problem in a system designed primarily to detect flaws, but not to image or distinguish flaws. The high impedance mismatch present at the flaw interface assures that energy will be returned into the aperture with sufficient signal to noise ratio. The advantages of high maintainability, rapid scanning capability, and low cost for the phase one system are therefore not compromised.

However, the poor lateral resolution of such a small aperture can only be handled by greatly increasing the array length. During phase two the entire set of 256 electrodes in the array aperture will be sampled simultaneously. Hence, in the phase two system the array aperture, \( D \), will increase to 14 cm. Then the corresponding beam parameters will become

\[ s = 0.43 \text{ mm}, \quad l = 0.98 \text{ mm}, \quad e = 4.4 \text{ cm}. \]
at the same depth previously considered. This greatly increased lateral resolution can be exploited to form high quality images using either simulated lens pulse-echo phased array processing or digital holography in two or three dimensions. However, the short depth of focus or focal spot length will demand a dynamic focal depth adjustment capability in the phased array signal processing, since a pulse will remain confined only while passing through the focal spot. For holographic processing, or quasioptic lens simulation, however, the phase two beam parameters will be practically ideal for this application.

7.2 PHASE TWO - DIGITAL HOLOGRAPHY

During phase two not only will the aperture length be dramatically increased, but the nature of the signal processing will be made fundamentally more flexible. The aperture will be discretely sampled at 256 uniformly spaced positions and the signals obtained at these array elements will be digitized. This will permit a wide variety of signal processing and imaging techniques to be employed. In theory, any manipulation of the data consistent with the signal to noise ratio, linearity, dynamic range and quantization accuracy of the acquired signals can be employed once the data is digitized. This includes simulated phased array lens
processing with dynamic focusing, weakly focused pulse-echo array processing, and digital holography in two or three dimensions.

The three-dimensional form of digital holography [5, 6, 7] can be implemented if the three-dimensional amplitude and phase field \( f(x,y,z) \) of a wave satisfying the CW Helmholtz equation is spatially sampled in a planar aperture (say \( z = 0 \)). Given that the field is entirely due to a backscattered wave, the two-dimensional Fourier transform

\[
F(u,v,z) = \int \int U(x,y,z) \, e^{i\phi(x,y,z)}
\]

\[
e^{-2\pi i(ux + uy)} \, dx \, dy
\]

for the field at a given depth, \( z \), can be obtained for any value of \( z \) from the value of \( F(u,v,0) \):

\[
f(x,y,z) = F(u,v,z) \, e^{2\pi i(ux + uy)} \, dudv
\]
This is a direct result of the fact that $f(x,y,z)$ satisfies the Helmholtz equation [6]. By substituting any value of $z$ into this equation and inverting the transform of $F(u,v,z)$, one can obtain $f(x,y,z)$ for that same value of $z$. Two-dimensional versions of this processing exist for samples obtained in a linear aperture. The result would be a trace of the field along a line at a given depth, $z$, from the array.

In order to implement this technique, the spatial sampling of the aperture must be made with array elements of very small dimensions in order that all of the backscattered plane (or line wave) components scattered by the flaw can be "seen" by each array element throughout the array aperture. Faithful reproduction of images can be expected only in regions where this "point" receiver approximation is met in practice, but the demands imposed by this criterion are no more severe than required by phased array focused lens simulation or any other imaging method. Development of an array technology which meets this objective is one of the ongoing objectives of this research program.
It has been observed that the Searle unit's phased array probe has an objectionable lack of resolution in both the depth and width dimensions. This is because the phased array utilizes only 16 of the possible 256 array transducer elements during any given pulse-echo data acquisition. Groups of 16 elements are selected in progression along the width of the probe. In this manner, 118 distinct groups of elements are selected to acquire an entire image comprised of 118 vertical samplings of the steel. This simplified imaging strategy limits the optical aperture of the array to approximately one centimeter (the width of 16 elements), which in turn limits the resolution. A certain degree of synthetic aperture enhancement is feasible using deconvolution.

A deconvolution board has been proposed for the purpose of image enhancement and improvement of flaw detection reliability in the phase one system. This board will be interfaced to the LSI-11 Q-bus, will have direct memory access (DMA) capability, and a high speed (10 MHz) arithmetic capability for simple array
Deconvolution has been proposed as a tool for enhancing the image sharpness. The cause of the impaired image sharpness has been traced to a resolution problem in the phased array probe, due to an effect known as the beam spreading function. We have observed that the beam spreading function is predictable and is limited by the aperture size of the array, as previously described. Since the phased array probe and the steel exhibit the properties of a linear system, the effect of this decrease in resolution can be studied using linear discrete systems theory such as Fourier and convolutional analysis [2].

The general effect of the beam spreading function is a blurring of the image and consequently an increase in the apparent size of any flaw or back wall shadow. Images which should have sharp (high contrast) edges are instead smeared into images with gradual intensity gradients. This effect occurs in both dimensions of the image, however the horizontal dimension (parallel to the back wall) is the most critical since most intensity comparisons are done along the back wall. Both the small aperture size of the probe and the limited number of transducer elements involved in the data acquisition have been shown to be causes for this spreading which can smear 0.25 inch features over an
Actual images are currently being analyzed with the VAX 11/780 computer at R.P.I. These images were obtained by uploading the contents of the framestore to the VAX. Samples of intensities from the back wall (corresponding to a row in the framestore memory) are considered as numerical sequences, or vectors, that can be transformed and convolved.

In general systems theory, time is considered to be the independent variable but, in this case, we are dealing with spatial distributions and spatial frequencies, and hence displacement along the back wall is the independent variable. Since the back wall samples are discrete in both amplitude and displacement, we will be dealing with the Discrete Fourier Transform (DFT) and discrete convolution.

To correct the beam spreading effect, a digital filter must be designed which will counteract the filter characteristic of the spreading function. Hence, we must first find the spreading filter characteristic by applying an impulse to the system and observing the response. The DFT of this response will be the desired spreading filter $H(k)$, by definition. Symbolically we note that if $X(k)$ denotes the DFT of the input and $Y(k)$ denotes the DFT of the output then $Y(k)=H(k)X(k)$. Note
that if the input is an impulse, then $X(k) = 1$ (by definition) and hence $Y(k)$ will equal $H(k)$, the desired spreading filter characteristic.

In our application we wish to determine the actual input to the system (the back wall or a flaw) when we have observed the output of the spreading filter $Y(k)$. Consequently we must find a filter characteristic which, when multiplied by $Y(k)$, will yield $X(k)$. This filter is the inverse of the spreading filter $H(k)$ which we will denote by $I(k) = 1/H(k)$. In practice, $I(k)$ will not be precisely the required filter for several reasons.

The spreading function is essentially low-pass in nature but the higher frequency components are required for good resolution. Merely boosting the high frequency components will not prove to be effective because of a hitherto unmentioned factor: noise. After the ultrasound is converted to electrical energy by the array elements, electrical noise (which has high frequency components) is superimposed on the signal that is eventually digitized. Hence, a compromise must be made between boosting the high frequencies and not allowing the noise to be increased beyond reasonable proportions. Due to this noise, the optimum image enhancement can not be obtained. The general attributes of the desired filter are known, however,
and provide a starting point for experimentation.

The desired inverse filter characteristic should essentially have unity gain at the low frequencies which will allow a uniform portion of the back wall to be virtually unaffected. Over a certain range of medium frequencies, the filter response should be greater than unity which will act to increase the edge sharpness of the image. Experimentation will determine what frequencies should be boosted and to what degree. Above some higher cut-off frequency (also experimentally determined), the filter response must decrease to zero gain. This acts to remove the high frequency noise from the enhanced image.

At this point it is appropriate to mention that since this analysis is being done in the spatial frequency domain, it is necessary to obtain the DFT of the input (the samples of the back wall). A multiplication is then required, followed by an inverse DFT transformation. Although this is readily done by computer simulation on the VAX, it is less easily accomplished in hardware. Consequently an alternate solution, known as deconvolution, is more feasible from the hardware point of view.
If we look at a linear discrete system in the time domain (or, in our case, the spatial distribution domain), the output of the system \( y(n) \) is calculated from a convolution of the input samples \( x(n) \) and the system weighting sequence \( h(n) \). This weighting sequence \( h(n) \) is actually the inverse DFT of the system filter characteristic \( H(k) \). Just as an experimental inverse filter \( I(k) \) exists, an inverse weighting sequence \( i(n) \) exists. The symbolic notation for this convolution is \( y(n)=h(n)\ast x(n) \) where \( n \) is an index meaning the \( n \)th sample of the numerical sequence or vector. The discrete convolution operator \( (\ast) \) is defined below:

\[
f(n) \ast g(n) = \sum_{k=-\infty}^{\infty} f(k) g(n-k)
\]

This sum of products formula can be implemented quite easily in hardware. Consequently, using convolution (or "deconvolution" since the inverse weighting sequence is used) we can calculate the output sequence (literally the enhanced back wall itself)
directly. Using this method we need only specify the inverse weighting sequence \( i(n) \) and the original back wall samples \( x(n) \).

8.2 HARDWARE

The proposed deconvolution board is an extremely powerful peripheral device to the LSI-11 Q-Bus. The primary function of the deconvolution board will be to perform the required multiplications and summations in a fraction of the time required by the LSI-11 for equivalent operations. Direct Memory Access (DMA) capability has been included on the board. This feature allows it to have random access to any memory on the LSI-11's Q-Bus without processor intervention. Consequently, with the proper microcode, the board can take control of the Q-Bus and execute flaw detection and framestore searching operations as well as deconvolution. In fact, if the board is programmed to its full capability, it can conceivably operate on the framestore more effectively than the LSI-11. The board has the capability of fetching its own instructions from memory, much like the master processor.

The primary architectural advantage of the hardware is that its control sequencing and arithmetic operations are based on Advanced Micro Devices' 2900 series family of bit-slice integrated circuits. AMD's
2900 series bit-slice architecture permits the design of a very versatile and powerful machine because its sequencing, decision-making, and arithmetic and control operations can be programmed with a design-dependent language called microcode. This microcode is stored in the board's writable control store memory. Microcode forms a "program" that determines the board's internal control and sequencing signals. Each instruction that is given to the board has an associated set of microinstructions that are executed.

As mentioned before, the board is very versatile. New modes of operation can be implemented by merely changing the microcode. Because the microcode is stored in read/write random access memory instead of read only memory, it can be modified from the system console or the flaw detection program. This allows the flaw analysis program to adaptively delegate varying responsibilities to the deconvolution board.

A diagram of the architecture is provided in figure 24. There are several subsystems on the board:

1) The AMD 2910 SEQUENCER controls all of the decision-making processes (condition code testing) and microprogram branching. It is this chip that generates the address to the microprogram memory and therefore determines the microinstruction to be executed next.
2) The WRITABLE CONTROL STORE is the microprogram memory. It is a 55 nanosecond access, 4K word by 72 bit static random access memory.

3) The WRITABLE CONTROL STORE CONTROL section controls the loading of the microprogram memory from the Q-Bus.

4) The AMD 2901 ALU is the arithmetic and logic unit for the board. It generates the 16-bit memory addresses for DMA and performs all the numerical and logical operations on the data.

5) The TRW MPY8-HJ is an 8-bit by 8-bit multiplier chip that multiplies the weighting sequence $h(k)$ by the input sequence $x(k)$. The multiplication can be done in only one board clock cycle which is approximately 80 times faster than the same operation on the LSI-11.

6) The H(K), Y(K) and X(K) RAM chips provide the on-board high speed data cache for the storage of the three sequences. The memories are large enough to allow four sequences to be stored in each.

7) The Q-BUS INTERFACE section contains the Q-BUS drivers and receivers, address decoders, and bus interface logic for software driven data transfers to and from the board. This interface section allows the programmer to have read/write access to the four
registers on the board as well as the three random access memory vectors.

8) The INTERRUPT CONTROL section is responsible for generating the Q-BUS signals that initiate a processor interrupt. The interrupt vector is fully programmable and is usually specified as one of the command string operands. Hence, one of several different interrupt service subroutines can be pointed to, depending on the particular function being performed by the board.

9) The DMA CONTROL section controls the LSI-11 Q-bus signals that are involved in requesting the bus and asserting bus mastership. Once bus mastership is attained, this section controls the Q-BUS timing for DMA data transfers to and from the board. Direct Memory Access allows blocks of data to be transferred from the framestore to the on-board cache at much higher rates than LSI-11 software loops would allow. DMA capability also allows the board to read or write data on its own initiative. This allows it to effectively fetch its own instructions.
8.3 DISCUSSION OF DECONVOLUTION

The only drawback to the deconvolution subsystem is the fact that the effects on the image will be severely limited in the presence of large amounts of noise. This system is inherently noisy and it is for this reason that the deconvolution board which has been discussed in this section could not be implemented.
CONCLUSIONS

Implementation of a flaw detection system consisting of a commercially available medical ultrasonic system interfaced to an LSI11/03 has proved feasible. The advantages of this approach are the high degree of maintainability of the bulk of the system components, the low cost of the system due to limited requirements for new engineering, and the high processing throughput rate obtainable. However, due to the small aperture employed in typical commercial systems (usually only 16 elements are activated at any time out of the 256 in the array), and the constrained method of beam formation, the quality of flaw imaging obtainable is inadequate for precise flaw characterization or close flaw resolution. To correct this situation a second phase system specialized for image formation has been proposed. This system would not at first offer the high throughput rate possible with the simpler phase one detection system, but would offer a much larger aperture (simultaneously accessing all 256 elements), and a greater flexibility in signal processing. This second phase system would operate in tandem with the phase one system's high speed flaw detector, concentrating on regions where the phase one
system locates a flaw. The phase two system will provide for a variety of imaging methods including focused simulated lens phase array techniques with dynamic focusing, pulse-echo weakly focused processing, and digital holography.


APPENDIX I

; FLAW DETECTION AND IDENTIFICATION PROGRAM
; BENET WEAPONS LAB PROJECT - R.P.I. MARCH 1982
; PROGRAMMED BY TOM CAVILLEER
; THIS PROGRAM IS WRITTEN IN MACRO11-RSX FOR USE
; ON THE PDP-11 WHICH PROVIDES THE CENTRAL PROCESSING
; CAPABILITY OF THE SYSTEM. THE PROGRAM SERVES
; SEVERAL PURPOSES. THE PROGRAM PROVIDES A MEANS OF
; OPERATOR CONTROL OF THE SYSTEM FUNCTIONS. THE USER
; CAN ALTER VARIOUS GAINS AND PARAMETERS OF THE SEARLE
; IMAGING SYSTEM. SUBROUTINES CONTROL THE DISPLAY OF
; INFORMATION ABOUT THE SYSTEM ON THE CRT TERMINAL AND
; THE SEARLE UNIT'S TELEVISION SCREEN. FINALLY, THE
; ACTUAL WORK OF FLAW DETECTION WITHIN THE IMAGE IS
; DONE BY THE SYSTEM SOFTWARE.
; THIS PROGRAM IS INTENDED AS THE CORE OF A
; LARGER SOFTWARE PACKAGE THAT WILL ACCOMMODATE CONTROL
; OF THE MECHANICAL MOVEMENTS OF THE SYSTEM AND THE
; STORAGE AND RETRIEVAL OF INFORMATION REGARDING THE
; LOCATION OF FLAWS. THESE ADDITIONAL FUNCTIONS WILL
; BE INTEGRATED INTO THE SOFTWARE AS THE MECHANICAL
; HARDWARE INVOLVED IS COMPLETED.
; ARRAY AND FLAW LOCATION PROGRAM
; PROGRAMMED BY RON BURGIN DECEMBER 1982
; THIS PROGRAM USES THE SOFTWARE DEVELOPED BY
; TOM CAVILLEER. THE LINE CLOCK INTERRUPT IS USED
; TO KEEP TRACK OF THE ROTATIONAL POSITION OF THE
; BILLET. ON POWER UP THE ARRAY IS MOVED COMPLETELY
; TO THE RIGHT. THE OPERATOR INPUTS THE DISPLACEMENT
; THAT HE WANTS TO START OFF AT AND THE LENGTH
; OF THE BILLET. AFTER RECEIVING LEGAL
; RESPONSES, THE BILLET IS TIMED FOR FIVE ROTATIONS
; TO CALCULATE THE AMOUNT TO ADD PER LINE CLOCK
; INTERRUPT. THE OPERATOR CAN THEN ALIGN THE ARRAY
; AND START THE SCAN.
; WHEN THE SCAN IS COMPLETE THE PROGRAM REPORTS
; THE NUMBER OF FLAWS FOUND. THE OPERATOR CAN THEN
; LOOK AT THE FLAW LOG TO GET AN IDEA OF WHERE THE
; FLAWS ARE. TO SEE THE FLAWS, THE OPERATOR ENTERS
; THE FLAW NUMBER (GIVEN BY THE DISPLAY FLAW ROUTINE)
; AND THE FLAW WILL BE DISPLAYED. BY PRESSING CERTAIN
; KEYS THE OPERATOR CAN MOVE THE ARRAY AND CHANGE THE
; ANGLE DESIRED, TO GET A BETTER VIEW OF THE FLAW.
; THE SOFTWARE RESIDES ON EIGHT INTEL 2708 PROMS
; LOCATED ON A STANDARD DIGITAL BOARD. THE PROGRAM
; CAN BE ADAPTED BY CHANGING A SINGLE LINE OF CODE INTO
; EITHER OF TWO CONFIGURATIONS. IF THE PROGRAM BEGINS
; WITH THE LINE: PROMS=0 ; THEN IT WILL BE ASSEMBLED
; AS CODE INTENDED FOR DOWNLOADING DIRECTLY TO THE
; RAM IN THE PDP-11. THIS MODE IS USED FOR PROGRAM
; DEVELOPMENT, MODIFICATION, AND TESTING. A PROGRAM
; READY FOR MORE PERMANENT STORAGE BEGINS WITH THE

102
; LINE: PROMS=1 ; THIS WILL BE ASSEMBLED FOR THE PROMS WHICH ARE LOCATED AT LOCATION 0 IN THE COMPUTER'S MEMORY.
; WHEN THE MACHINE IS POWERED UP, THE SOFTWARE BEGINS EXECUTION FROM THE PROMS. THE FIRST OPERATION OF THE PROGRAM IS TO COPY ITSELF TO RAM. THIS IS DONE TO INCREASE PROGRAM SPEED; SINCE THE INSTRUCTION FETCH TIME IS MUCH QUICKER FROM RAM THAN PROM. THE PROGRAM THEN PASSES CONTROL TO THE APPROPRIATE POINT IN RAM AND BEGINS EXECUTION.
; THIS PROGRAM WILL ASSEMBLE FOR STORAGE IN PROGRAMMABLE READ-ONLY MEMORY.

PROMS=1

; SECTION 0. INTERRUPT VECTORS
; THIS SECTION SETS UP THE INTERRUPT VECTORS AT THEIR PROPER LOCATIONS IN PROM. THE VECTORS AND THEIR LOCATIONS ARE AS FOLLOWS:
; STRTPC=24 ;POWER UP PC
; STRTPS=26 ;POWER UP PS
; PRSTHV=200 ; INTERRUPT FROM PROBE RESET
; STEPRV=300 ; INTERRUPT FROM STEPPER MOTOR
; TERMRV=60 ; RECEIVE INTERRUPT FROM TERMINAL
; TERMTV=64 ; TRANSMIT INTERRUPT TO TERMINAL
; CLKINT=100 ; INTERRUPT FROM LINE CLOCK

; .IF NE PROMS
; .ASECT

FDS4:
; .0

; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0

; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0

VSRTPC: ; .WORD 376
VSRTPS: ; .WORD 0

; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0
; .WORD 0

VTERMR: ; .WORD 420
; .WORD 0
VTERMT: .WORD 424
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0

CLKINT: .WORD CLKTIC ;LINE CLOCK
        INTERRUPT
CLKPRI: .WORD 340 ;LINE CLOCK
        HAS PRI0=7
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0

VPRSTH: .WORD 430
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
                .WORD 0
VSTEPR: .WORD 434
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD RXINT
        .WORD 200
        .WORD TXINT
        .WORD 200
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0
        .WORD 0

; SECTION 0-1. PROGRAM COPIES ITSELF TO RAM
COPY:
        MOV 37760, RO ;LOCATION IN RAM

105
MOV 442,R1 ;LOCATION IN PROM
MOV 10000,R5 ;WORD COUNT

S111:
MOV (R1)+,(R0)+ ;MOVE WORD TO RAM
SOB R5,S111 ;CONTINUE UNTIL COUNT=0
JMP 40000 ;TRANSFER CONTROL TO RAM

; THESE JUMPS STEER THE PROGRAM TO THE PROPER INTERRUPT
; SERVICE SUBROUTINE.
JMP 37760
JMP 37762
JMP 37764
JMP 37766
. ENDC

; SECTION 0-2. DOWNLOADED PROGRAM
; IF THE PROGRAM IS INTENDED FOR DOWNLOADING, THIS
; SECTION WILL ASSEMBLE IT FOR THE PROPER LOCATION.
. IF EQ PROMS
. ASECTION
FDS5: . =37760
. ENDC
JMP IVECT1
JMP IVECT2
JMP IVECT3
JMP IVECT4

; SECTION 1. INITIALIZATIONS
; THIS SECTION OF THE CODE INITIALIZES THE STACK,
; SYSTEM CONTROL PARAMETERS, DISPLAYS POWER UP
; VIA SUBROUTINE. IT ALSO CALLS A SUBROUTINE
; WHICH CLEARS BOTH SYSTEM FRAMESTORES.
INIT: MOV PSTACK,SP ;PROGRAM STACK
. IF EQ PROMS
MOV 37760,@ TERMRV ;CRT READ INTERRUPT
MOV 37762,@ TERMTV ;CRT WRITE INTERRUPT
MOV 37764,@ PRSTHV ;PROBE RESET HIGH
MOV 37766,@ STEPRV ;STEPPER MOTOR END
MOV CLKTIC,@ 100 ;CLOCK INTERRUPT
MOV 340,@ 102
. ENDC
JSR PC,POWERU ;POWER UP

INSTRUCTIONS
MOV 377,@ MTXBUF ;MOVE MOTOR A LITTLE
CLR @ FRAMEA ;FRAMEO ON QBUS
JSR PC,FSCLRPF ;CLEAR FRAMESTORE 0
INC @ FRAMEA ;FRAME1 ON QBUS
JSR PC,FSCLRFP ;CLEAR FRAMESTORE 1
MOV 1043,@ SRDLY ;INIT. SR, DELAY
MOV 2,@ NOISE2 ;INIT. BACKG. NOISE
MOV 60,@ NOISE ;INIT. NOISE LEVEL
MOV 2,@ GAINS ;INIT. GAINS
MOV 370,@ SLOPE ;INIT. SLOPE
MOV 605,@ GBUF ;INIT. RXATN, TXGAIN
MOV 0,@ SHADOW ;INIT. SHADOW THRESHOLD
CLR @ QBUS ;FRAMESTORE 0 ON QBUS
MOV    1, @ DBUS          ; FRAMESTORE 1 ON DBUS
JSR    PC, HOME           ; MOVE ARRAY ALL THE WAY LEFT
JSR    PC, GETLAR         ; GET DispMENT AND LENGTH
MOV    1, @ CFLAG         ; SET CFLAG FOR COMPLETE FLAW DECTIION
CLR    @ KFLAG            ; SET CFLAG FOR SENSITIVE FLAG
CLR    @ ROW2             ; THIS IS THE POINTER TO THE BOTTOM
 ; OF THE BACKWALL. IF ON THE FIRST ATTEMPT AT FLAW DETECTION THE BACKWALL IS NOT FOUND AND A LARGE NUMBER IS LEFT IN THIS LOCATION; THE SUBROUTINE CLEAN WILL TRY TO ACCESS NON-EXISTENT MEMORY. THIS WILL KILL THE SOFTWARE, AND THEREFORE MUST BE CLEARED. ; SECTION 2. MAIN MENU ROUTINES ; THIS SECTION HANDLES THE THREE MAIN MENUS. FOR EACH MENU THE CHOICES ARE DISPLAYED AND THEN WAITS FOR THE OPERATOR TO RESPOND. AN ILLEGAL RESPONSE IS IGNORED. ; FIRST MENU COMMNI:
    MOV DINIT1, RO          ; LOC. OF MESSAGE->RO
    JSR PC, OUTPUT          ; MESSAGE TO CRT
    MOVB @ DBUS, RO          ; FRAMESTORE ---> RO
    JSR PC, ONUM2           ; SHOW FRAMESTORE
    MOV DINIT2, RO           ; LOC. OF MESSAGE->RO
    JSR PC, OUTPUT           ; MESSAGE TO CRT
 ; PLACE COMMAND CHOICES ON THE SCREEN
    MOV CMESSI, RO           ; LOCATION OF MESSAGE
    JSR PC, OUTPUT           ; MESSAGE TO TERMINAL
 ; TAKE COMMAND FROM USER
S2L2: TSTB @ TRDCSR        ; TEST FOR KEYSTROKE
    BPL S2L2                ; WAIT FOR KEYSTROKE
INCHR: MOVB @ TRDBUF, R1   ; ACCEPT USER INPUT
 ; TEST CHARACTER AND BRANCH TO APPROPRIATE ROUTINE
TESTC: BIC 177600, R1      ; CLEAR NON ASCII BITS
    CMP R1, 123             ; S Pressed ?
    BEQ PARAMJ              ; SET FRAMES
    CMP R1, 120             ; P Pressed ?
    BEQ PROBEJ              ; PROBE
    CMP R1, 103             ; C Pressed?
    BEQ FSCLJ1              ; FRAMESTORE CLEAR
    CMP R1, 104             ; D Pressed?
    BEQ DFJ1                ; SWITCH FRAME STORES
    CMP R1, 115             ; M Pressed?
    BEQ COMMN3              ; EXTRA ROUTINES
    CMP R1, 106             ; F Pressed?
    BEQ SCANJ               ; FIND FLAWS
    CMP R1, 121             ; Q Pressed?
107
BEQ POWRDJ ;POWER DOWN
JMP COMMN1

PARAMJ: JMP PARAMS
PROBEJ: JMP PROBE
FSCLJ1: JSR PC,FSCLR
JMP COMMN1
DFJ1: JSR PC,DF
JMP COMMN1
SCANJ: JMP SCAN
POWRDJ: JSR PC,POWERD
JMP COMMN1

;THE SECOND MENU

COMMN2: MOV MENU2,R0 ;DISPLAY THE MENU
JSR PC,OUTPUT

COMW2: TSTB @ TRDCSR ;WAIT FOR A RESPONSE
BPL COMW2
MOV @ TRDBUF,R1 ;CLEAR RESPONSE
BIC 177600,R1 ;CLEAR UPPER BYTE
CMP R1, 123 ;S PRESSED?
BEQ DLOGJ ;DISPLAY LOG
CMP R1, 103 ;C PRESSED?
BEQ FSCLJ2 ;FRAMESTORE CLEAR
CMP R1, 104 ;D PRESSED?
BEQ DFJ2 ;SWITCH FRAME STORES
CMP R1, 115 ;M PRESSED?
BEQ COMMN3 ;EXTRA ROUTINES
CMP R1, 121 ;Q PRESSED?
BEQ POWRJ2 ;POWER DOWN
CMP R1, 114 ;L PRESSED?
BEQ LOCJ ;LOCATE FLAWS
JMP COMMN2 ;IGNORE RESPONSE

DLOGJ: JMP DLOG
LOCJ: JMP LOCATE
DFJ2: JSR PC,DF
JMP COMMN2
POWRJ2: JSR PC,POWERD
JMP COMMN2
FSCLJ2: JSR PC,FSCLR
JMP COMMN2

;COMMN3 EXTRA ROUTINES
;THIS IS THE THIRD MENU
;IT INCLUDES ROUTINES WHICH MAY BE USEFUL TO THE OPERATOR
;BUT AREN'T NECESSARY TO DO THE BILLET SCANNING. I.E.
;REPEATED FLAW DETECTION COULD AID IN SETTING THE PARAMETERS

COMMN3: MOV MENU3,R0 ;DISPLAY THE MENU
JSR PC,OUTPUT

COMW3: TSTB @ TRDCSR ;WAIT FOR A RESPONSE
BPL COMW3
MOV @ TRDBUF, R1 ; GET RESPONSE
BIC 177600, R1 ; CLEAR UPPER BYTE
CMP R1, 103 ; C PRESSED?
BEQ SSFJ ; SINGLE SHOT FLAW DETC
CMP R1, 104 ; D PRESSED?
BEQ RFDJ ; REPEATED FLAW DETC
CMP R1, 106 ; F PRESSED?
BNE COM3 ; NO
JMP COMMN1 ; EXIT TO FIRST MENU
COM3: CMP R1, 123 ; S PRESSED?
BNE COM3B ; NO
JMP COMMN2 ; EXIT TO SECOND MENU
COM3B: CMP RI, 121 ; Q PRESSED?
BEQ POWRJ3 ; POWER DOWN
CMP R1, 125 ; U PRESSED?
BEQ UPJ ; Upload FRAME STORE
CMP R1, 110 ; H PRESSED
BNE COMMN3 ; NO
JSR PC, HOMEA ; HOME AGAIN
BR COMMN3
SSFJ: JMP GO
RFDJ: JMP REPEAT
UPJ: JMP UPLOAD
POWRJ3: JSR PC, POWERD
JMP COMMN3

SECTION 2-1. COMMAND ROUTINES
IN DEVELOPING THIS PROGRAM, FLEXIBILITY WAS A MAJOR CONSIDERATION. BECAUSE I WANTED TO HAVE THE ABILITY TO ADD NEW SECTIONS AS THEY BECAME AVAILABLE, I DECIDED TO USE COMMAND ROUTINES TO CONTROL THE EXECUTION OF SUBROUTINES. THUS A NEW SUBROUTINE CAN BE ADDED WITHOUT CHANGING EXISTING WORKING ROUTINES. THIS SECTION CONSISTS OF THESE CONTROL ROUTINES. ALSO, THE SAME ROUTINES CAN BE USED IN SINGLE-SHOT OR REPEATED EXECUTION MODE

PARAMS CONTROLS THE SETTING OF PARAMETERS USING THE TERMINAL.
PARAMS:
JSR PC, PCHANG ; SET PARAMETERS SUBROUTINE
JMP COMMN1 ; RETURN
GO TAKES A SINGLE FRAME AND ANALYZES IT FOR FLAWS.
GO:
MOV 1, @ CFLAG
MOV @ QBUS, @ FRAMEA ; FILL FRAMESTORE
MOV @ DBUS, @ FRAMEB ; ANALYZE FRAMESTORE
JSR PC, SAMPLE ; TAKE SAMPLE
JSR PC, FLAWDP ; SEARCH FOR FLAWS
JSR PC, CLEAN ; CLEAN UP PICTURE
BELOW BACKWALL
MOV @ QBUS, @ FRAMEB ; SHOW RESULT
JSR PC, DISPLAY ; DISPLAY ON TV SCREEN
JMP COMMN3 ;RETURN

; PROBE PROVIDES A TV GRAPH OF THE BACKWALL
; ILLUMINATION FUNCTION.

PROBE:
    CLR @ FLAG2

PROBE1: MOV @ QBUS,@ FRAMEA ;FRAME ON QBUS
    MOV @ DBUS,@ FRAMEB
    JSR PC,SAMPLE ;TAKE SAMPLE
    JSR PC,ILLUM ;ILLUMINATION SUBROUTINE

; FLIP THE FRAMESTORE ON THE DBUS TO THE QBUS; AND THE
; FRAMESTORE ON THE QBUS TO THE DBUS.

FLIP: TST @ QBUS
    BEQ S3P1
    CLR @ QBUS
    INC @ DBUS
    BR S3P2

S3P1: INC @ QBUS
    CLR @ DBUS

; CONTINUE EXECUTING THESE ROUTINES UNTIL NEW COMMAND
; IS ENTERED.

S3P2: TSTB @ TRDCSR ;TEST FOR KEYSTROKE
    BPL PROBE1 ;REPEAT UNTIL KEYSTRK
    MOVB @ TRDBUF,@ CMDCHR ;DISPOSE OF CHAR.
    JMP INCHR ;ON COMMAND, RETURN

; REPEAT PROVIDES FOR CONTINUOUS FLAW DETECTION;
; IT IS TERMINATED BY A NEW COMMAND FROM THE USER.

REPEAT:
    MOV 1,@ CFLAG
    MOV @ QBUS,@ FRAMEA
    MOV @ DBUS,@ FRAMEB
    JSR PC,SAMPLE
    JSR PC,FLAWDP
    JSR PC,CLEAN

FLIP4: TST @ QBUS
    BEQ S3P41
    CLR @ QBUS
    INC @ DBUS
    BR S3P42

S3P41: INC @ QBUS
    CLR @ DBUS

S3P42: TSTB @ TRDCSR ;TEST FOR KEYSTROKE
    BPL REPEAT ;REPEAT UNTIL KEYSTROKE
    JMP COMMN3 ;RETURN

; FSCLR CONTROLS THE CLEARING OF EITHER OF THE FRAME-
; STORES.

FSCLR:
    MOV @ DBUS,@ FRAMEA ;CLEAR FRAME
    MOV @ QBUS,@ FRAMEB ;DISPLAY FRAME
    JSR PC,FSCLRP ;CLEAR FRAMESTORE
    RTS PC

; DF CONTROLS THE DISPLAY OF EITHER FRAMESTORE ON THE
; TV SCREEN AT THE USER'S COMMAND.

DF:
JSR PC,DFRAME
RTS PC

; SECTION 4. SAMPLE ROUTINE / DISPLAY ROUTINE
; THE SUBROUTINE SAMPLE IS USED TO TAKE AN IMAGE
; WITH THE SEARLE IMAGER. THE INPUT PARAMETER FRAMEA
; TELLS WHICH FRAMESTORE TO PLACE ON THE QBUS DURING
; THE READ. THE INPUT PARAMETER FRAMEB TELLS THE
; SUBROUTINE WHICH FRAMESTORE TO PLACE ON THE QBUS
; AFTER THE IMAGE IS READ.
; FOR EXAMPLE IF FRAMEA=0 AND FRAMEB=1
; THEN ON COMPLETION FRAMESTORE 0 WILL HAVE
; THE NEW DATA AND FRAMESTORE 1 WILL BE DISPLAYED.
; THE SUBSECTION OF SUBROUTINE BEGINNING WITH
; DISPLY: IS USED AS AN INDEPENDENT SUBROUTINE TO
; DISPLAY ONE OF THE FRAMESTORES AT USER COMMAND.
SAMPLE:

MOV @ SRDLY,@ ISRDLY ;LOAD IMAGER REGISTERS
MOV @ RAMPS,@ :RAMPS
MOV @ GBUF,@ IGAINS

S4L0: TSTB @ ICSR ;CHECK FOR PRSTH LOW
BMI S4L0

S4L1: TSTB @ ICSR ;WAIT FOR PRSTH
BPL S4L1

INTS1: MOVB @ FRAMEA,@ ICSR ;CONNECT BUSSES
BIS 2,@ ICSR ;FRAMESTORE WRITE ENABLED

DISPLY:

TSTB @ ICSR ;WAIT FOR PRSTH LOW
BMI DISPLY

S4L2: TSTB @ ICSR ;WAIT FOR PRSTH
BPL S4L2

INTS2: MOVB @ FRAMEB,@ ICSR ;CONNECT BUSSES
BIC 400,@ ICSR ;DISABLE INTERRUPT
RTS PC ;RETURN

; SECTION 5. ILLUMINATION FUNCTION DISPLAY ROUTINE
; THIS SUBROUTINE DISPLAYS A GRAPH OF THE BACKWALL
; ILLUMINATION FUNCTION ON THE TV SCREEN. THE SUB-
; ROUTINE SEARCH IS USED TO FIND THE BACKWALL. THE
; SUBROUTINE DISPLAYS THE ROW NUMBERS WHERE THE
; BACKWALL HAS BEEN FOUND ON THE CRT. IF THE WALL
; IS NOT FOUND, A FLAG IS SET AND THE ROUTINE RETURNS
; CONTROL TO THE COMMAND ROUTINE WHICH CALLED IT.

ILLUM:

JSR PC,SEARCH ;LOCATE BACKWALL
TSTB @ FLAG1 ;DID SEARCH FIND BACKWALL?
BEQ PATCH ;IF NOT, RETURN
RTS PC

; THIS SECTION OF CODE UPDATES THE ROW NUMBER DISPLAY
; ON THE CRT. ON THE FIRST PASS, HOWEVER, THE CODE
; ALSO CREATES THE MESSAGE TELLING THE USER WHERE THE
; FRAMESTORE HAS BEEN FOUND. BY ONLY WRITING THIS TEXT
; ONCE, FASTER OPERATION IN REPEATED MODE IS ATTAINED.

PATCH:

TSTB @ FLAG2 ;FIRST PASS?
BEQ S5P20 ;IF SO, WRITE TEXT

; ON MOST PASSES, JUST UPDATE THE ROW NUMBERS
MOV BMESS3,RO ;POSITION CURSOR
JSR PC,OUTPUT
MOVB @ ROW1,RO ;LOAD TOP OF WALL ROW
JSR PC,ONUM2 ;SHOW ROW NUMBER
MOV BMESS4,RO ;POSITION CURSOR
JSR PC,OUTPUT
MOVB @ ROW2,RO ;LOAD BOTTOM OF WALL ROW
JSR PC,ONUM2 ;SHOW ROW NUMBER
JMP S5P21 ;DRAW GRAPH

; ON FIRST PASS, CLEAR SCREEN AND WRITE TEXT
S5P20:  MOV WIFE,RO ;CLEAR MESSAGE AREA
JSR PC,OUTPUT
MOVB BMESS1,RO ;LOAD LOC. OF MESSAGE
JSR PC,OUTPUT ;OUTPUT TEXT
MOVB @ ROW1,RO ;LOAD TOP OF WALL ROW
JSR PC,ONUM2 ;SHOW ROW NUMBER
MOV BMESS2,RO ;LOAD LOC. OF MESSAGE
JSR PC,OUTPUT ;OUTPUT TEXT
MOVB @ ROW2,RO ;LOAD TOP OF WALL ROW
JSR PC,ONUM2 ;SHOW ROW NUMBER
MOVB 1,@ FLAG2 ;SET FIRST PASS FLAG

; PUT GRAPH ON SCREEN
S5P21:  JSR PC,SUPLIER ;SUM OVER BACKWALL IN COLUMN
JSR PC,SMOOTH ;AVERAGE ACROSS COLUMNS
MOV 60000,RO ;RO=COLUMN POINTER
MOV 166,R5 ;RS--COLUMN COUNTER
CLR R2 ;BACKWALL VECTOR INDEX
S5L6:
MOV R0,R1 ;LOAD UP POINTER
MOV 200,R4 ;NUMBER OF ROWS IN COLUMN
S5L8:
CLR (R1)+ ;CLEAR COLUMN
SOB R4,S5L8
MOV SMTHBW(R2),R3 ;TAKE DATA FROM VECTOR
ASR R3 ;SCALE R3 TO TV OUTPUT
ASR R3
NEG R3
ADD 300,R3
ADD R0,R3 ;MOVE TO CORRECT COLUMN
MOVB 77,(R3) ;PLACE DOT AT CORRECT POINT
ADD 400,R0 ;MOVE TO NEXT COLUMN
ADD 2,R2 ;INCREMENT VECTOR INDEX
SOB R5,S5L6 ;REPEAT FOR OTHER ROWS
RTS PC ;RETURN

; SECTION 6. CLEAR FRAMESTORE ROUTINE
; THIS SUBROUTINE CLEARS ONE OF THE FRAMESTORES.
; THE FRAME CLEARED WILL BE THE COMPLEMENT OF FRAMEA
; FRAMEB WILL BE THE DISPLAYED FRAME.
FSCLRP: MOV 60000,RO ;INITIALIZE POINTER
MOV 40000,R5 ;NUMBER OF WORDS IN FSTORE

S6LO: TSTB @ ICSR ;CHECK FOR PRSTH LOW
BMI S6LO

S6L2: TSTB @ ICSR ;WAIT FOR PRSTH
BPL S6L2

MOV @ FRAMEA,@ ICSR ;PUT OTHER FRAME ON

D-BUS

S6L1:
CLR (R0)+ ;CLEAR EACH WORD
SOB R5,S6L1 ;LOOP UNTIL COUNT=0

S6L3:
TSTB @ ICSR ;CHECK FOR PRSTH LOW
BMI S6L3

S6L4: TSTB @ ICSR ;WAIT FOR PRSTH
BPL S6L4

MOV @ FRAMEB,@ ICSR ;DISPLAY FRAMESTORE
RTS PC ;RETURN

; SECTION 7. PARAMETER CHANGE ROUTINE
; THIS SUBROUTINE ALLOWS THE USER TO ALTER PROGRAM
; THRESHOLDS.
; THIS SECTION OF PCHANG PRINTS THE USER PROMPT. THE
; CURRENT VALUES OF THE PARAMETERS ARE LISTED NEXT TO
; THE PARAMETER NAME.

PCHANG:

MOV CMESS2,RO ;LOAD MESSAGE LOCATION
JSR PC,OUTPUT ;PROMPT USER
MOV @ SRATE,RO ;CURRENT SAMPLING RATE
JSR PC,OUTNUM
MOV CMESS3,RO ;DISPLAY TRANSMIT GAIN
JSR PC,OUTPUT
MOV @ TXGAIN,RO
CLC
ROL RO
JSR PC,OUTNUM
MOV CMESS4,RO ;DISPLAY DELAY
JSR PC,OUTPUT
MOV @ DELAY,RO
JSR PC,OUTNUM
MOV CMESS5,RO ;DISPLAY NEAR GAIN
JSR PC,OUTPUT
MOV @ GAINS,RO
JSR PC,OUTNUM
MOV CMESS6,RO ;DISPLAY SLOPE GAIN
JSR PC,OUTPUT
MOV @ SLOPE,RO
BIC 200,RO ;CLEAR ENABLE BIT
JSR PC,OUTNUM
MOV CMESS7,RO ;DISPLAY ATTENUATION
JSR PC,OUTPUT
MOV @ RXATTN,RO
JSR PC,OUTNUM
MOV CMESS8,RO ;DISPLAY NOISE THRESHOLD
JSR PC,OUTPUT
MOV
  @ NOISE,RO
J
  JR PC,OUTNUM
MOV
  CMESSO,RO ;DISPLAY SHADOW THRESHOLD
J
  JR PC,OUTPUT
MOV
  @ SHADOW,RO
J
  JR PC,OUTNUM
; NOW WE WAIT FOR THE USER TO ENTER A NUMBER OR RETURN.
; IF HE RETURNS, CONTROL IS TRANSFERRED TO THE COMMAND
; ROUTINE. IF A NUMBER IS ENTERED, IT IS USED TO CALL
; THE CORRECT PARAMETER FOR CHANGE.
S7L2:  TSTB @ TRDCSR ;TEST FOR KEYSTROKE
BPL  S7L2 ;WAIT FOR KEYSTROKE
MOV
  @ TRDBUF,@ CMDCHR ;ACCEPT USER INPUT
BIC 177600,@ CMDCHR ;TRIM TO 7-BIT ASCII
CMPB
  @ CMDCHR, CR ;TEST FOR <CR>
BNE  S7P1 ;ON OTHER INPUT, CONTINUE
RTS PC ;ON <CR>, RETURN
S7P1:
CMPB
  @ CMDCHR, 111 ;TEST FOR ILLEGAL CHAR.
BPL  S7ERR1 ;PRINT ERROR IF ILLEGAL
CMPB
  @ CMDCHR, 101 ;TEST FOR ILLEGAL CHAR.
BMI  S7ERR1 ;PRINT ERROR IF ILLEGAL
BIC 177740,@ CMDCHR ;REDUCE CHARACTER
MOV
  @ CMDCHR,RO ;LOCATION OF COMMAND LIST
CLC
ROL
RO ;ADJUST TO WORD LENGTH
ADD @ S7LIST,RO ;RO NOW POINTER TO LIST
MOV
  (RO),R3
JMP (R1) ;BRANCH TO ROUTINE SELECTED
S7ERR1:  MOV
  WIPE,RO ;CLEAR MESSAGE AREA
JSR PC,OUTPUT
MOV
  EMESS2,RO ;LOC. OF ERROR MESSAGE
JSR PC,OUTPUT ;PRINT MESSAGE
JMP
  S7L2
; THIS SECTION CONTAINS THE SUBROUTINES WHICH ALTER THE
; VARIOUS PARAMETERS. EACH IS HANDLED IN AN
; INDEPENDENT ROUTINE SO THAT DIFFERENT MANIPULATIONS
; CAN BE USED TO PUT THE DECIMAL VALUE INTO THE
; REQUIRED FORMAT FOR USE BY THE PROGRAM.
S7AAAA:
MOV
  WIPE,RO ;CLEAR SCREEN
JSR PC,OUTPUT
MOV
  PMESSA,RO ;LOCATION OF MESSAGE
JSR PC,OUTPUT ;PROMPT USER
MOV
  @ SRATE,RO ;CURRENT SAMPLING RATE
MOV
  RO, -(SP) ;PUSH SAMPLING RATE
JSR PC,OUTNUM ;DISPLAY CURRENT VALUE
MOV
  (SP) +,R3 ;R3 = CURRENT VALUE
JSR PC,INNUM ;RETRIEVE USER CHANGE
MOV
  R3, @ SRATE ;STORE CHANGE
JMP
  PCHANG ;RETURN
S7BBBB:
MOV WIPE,RO ; CLEAR SCREEN
JSR PC,OUTPUT
MOV PMESSB,RO ; LOCATION OF MESSAGE
JSR PC,OUTPUT ; PROMPT USER
MOVB @ TXGAIN,RO ; CURRENT TRANSMIT GAIN
CLC
ROL R0
MOV R0, -(SP) ; PUSH TRANSMIT GAIN
JSR PC,OUTNUM ; DISPLAY CURRENT VALUE
MOV (SP)+,R3 ; R3 = CURRENT VALUE
JSR PC,INNUM ; RETRIEVE USER CHANGE
ASR R3 ; DIVIDE BY TWO TO SCALE
MOVB R3, @ TXGAIN ; STORE CHANGE
BEQ S7BBB1 ; IF INPUT=0 THEN DON'T SET ENABLE BIT
BISB 200, @ TXGAIN ; SET THE ENABLE BIT
S7BBB1: JMP PCHANG ; RETURN
S7CCCC: MOV WIPE,RO ; CLEAR SCREEN
JSR PC,OUTPUT
MOV PMESSC,RO ; LOCATION OF MESSAGE
JSR PC,OUTPUT ; PROMPT USER
MOVB @ DELAY,RO ; CURRENT DELAY
MOV RO, -(SP) ; PUSH CURRENT DELAY
JSR PC,OUTNUM ; DISPLAY CURRENT VALUE
MOV (SP)+,R3 ; R3 = CURRENT VALUE
JSR PC,INNUM ; RETRIEVE USER CHANGE
MOVB R3, @ DELAY ; STORE CHANGE
JMP PCHANG ; RETURN
S7DDDD: MOV WIPE,RO ; CLEAR SCREEN
JSR PC,OUTPUT
MOV PMESSD,RO ; LOCATION OF MESSAGE
JSR PC,OUTPUT ; PROMPT USER
MOVB @ GAINS,RO ; CURRENT VALUE OF GAINS
MOV RO, -(SP) ; PUSH CURRENT GAINS
JSR PC,OUTNUM ; DISPLAY CURRENT VALUE
MOV (SP)+,R3 ; R3 = CURRENT VALUE
JSR PC,INNUM ; RETRIEVE USER CHANGE
MOVB R3, @ GAINS ; STORE NEW GAIN VALUE
JMP PCHANG ; RETURN
S7EEEE: MOV WIPE,RO ; CLEAR SCREEN
JSR PC,OUTPUT
MOV PMESSF,RO ; LOCATION OF MESSAGE
JSR PC,OUTPUT ; PROMPT USER
MOVB @ SLOPE,RO ; CURRENT RAMP SLOPE
BICB 200, R0 ; CLEAR ENABLE BIT TO GET CORRECT NUMBER
MOV RO, -(SP) ; PUSH CURRENT SLOPE
JSR PC,OUTNUM ; DISPLAY CURRENT VALUE
MOV (SP)+,R3 ; R3 = CURRENT VALUE
JSR PC,INNUM ; RETRIEVE USER CHANGE

115
MOVB R3,@ SLOPE ;STORE NEW SLOPE VALUE
BEQ S7EEE1 ;IF ZERO THEN DON'T SET

ENABLE BIT

BISB 200,@ SLOPE ;SET ENABLE BIT
S7EEE1: JMP PCHANG ;RETURN
S7FFFF:

MOV WIPE,RO ;CLEAR SCREEN
JSR PC,OUTPUT
MOV PMESSG,RO ;LOCATION OF MESSAGE
JSR PC,OUTPUT ;PROMPT USER
MOVB @ RXATTN,RO ;CURRENT ATTENUATION
MOV R0,-(SP) ;PUSH CURRENT RXATTN
JSR PC,OUTNUM ;DISPLAY CURRENT VALUE
MOV (SP)+,R3 ;R3 = CURRENT VALUE
JSR PC,INNUM ;RETRIEVE USER CHANGE
TST R3
BEQ S7FFF1
MOVB 1,@ RXATTN ;SET TO 1
JMP PCHANG ;RETURN
S7FFF1: CLRB @ RXATTN ;SET TO 0
JMP PCHANG

S7GGGG:

MOV WIPE,RO ;CLEAR SCREEN
JSR PC,OUTPUT
MOV PMESSH,RO ;LOCATION OF MESSAGE
JSR PC,OUTPUT ;PROMPT USER
MOVB @ NOISE,RO ;CURRENT NOISE SETTING
MOV R0,-(SP) ;PUSH CURRENT NOISE

SETTING

JSR PC,OUTNUM ;DISPLAY CURRENT VALUE
MOV (SP)+,R3 ;R3 = CURRENT VALUE
JSR PC,INNUM ;RETRIEVE USER CHANGE
MOVB R3,@ NOISE ;STORE CHANGE
JMP PCHANG ;RETURN

S7HHHH:

MOV WIPE,RO ;CLEAR SCREEN
JSR PC,OUTPUT
MOV PMESSI,RO ;LOCATION OF MESSAGE
JSR PC,OUTPUT ;PROMPT USER
MOVB @ SHADOW,RO ;CURRENT SHADOW

THRESHOLD

MOV R0,-(SP) ;PUSH CURRENT THRESHOLD
JSR PC,OUTNUM ;DISPLAY CURRENT VALUE
MOV (SP)+,R3 ;R3 = CURRENT VALUE
JSR PC,INNUM ;RETRIEVE USER CHANGE
MOVB R3,@ SHADOW ;STORE CHANGE
JMP PCHANG ;RETURN

; SECTION 8. I/O SUBROUTINES
; SECTION 8-1. SUBROUTINE OUTNUM
; PRINTS ONE BYTE WITHIN PARENTHESES ON THE TERMINAL
OUTNUM:

BIC 177400,RO ;TAKE CARE OF HIGH BYTE
MOV NUMBUF,R1 ;OUTPUT BUFFER LOCATION

116
MOVB \(\text{LEFTP,}(R1)+\); LEFT PARENTHESIS
CLR R3 ; WORKING REGISTER
CLR R4 ; WORKING REGISTER
MOV RO, R2 ; NUMBER \(-\to R2\

; PERFORM OCTAL TO DECIMAL CONVERSION.
S7L3: SUB 144, R2
BMI S7P4
INC R3
MOV R2, RO
BR S7L3
S7P4: MOV RO, R2
S7L4: SUB 12, R2
BMI S7P5
INC R4
MOV R2, RO
BR S7L4

; CONVERT EACH DIGIT TO AN ASCII CHARACTER.
S7P5: ADD 260, R3
MOVB R3, (R1)+ ; LOAD INTO OUTPUT BUFFER
ADD 260, R4
MOVB R4, (R1)+ ; LOAD INTO OUTPUT BUFFER
ADD 260, RO
MOV R0, (R1)+ ; LOAD INTO OUTPUT BUFFER
MOV SPACE, (R1)+ ; RIGHT PARENTHESIS
MOVB RIGHTP, (R1)+ ; SPACE
CLRB (R1) ; TRAILING ZERO
MOV NUMBUF, RO ; LOCATION OF BUFFER
JSR PC, OUTPUT ; NUMBER TO TERMINAL
RTS PC ; RETURN

; SECTION 8-2. SUBROUTINE INNUM
; TAKES USER NUMBER FROM THE TERMINAL INTO A BYTE.
; IT CHECKS FOR A LEGAL RESPONSE.
INNUM:
CLR R1
CLR R2
CLR @ MANUAL

; TAKE FIRST INPUT CHARACTER
; CHAR1: TSTB @ TRDCSR
BPL CHAR1
MOVB @ TRDBUF, R1 ; CHAR. ENTERED BY USER
MOVB R1, @ TTXBUF ; ECHO
BIC 177600, R1
CMPB R1, 15 ; USER RETURNS TO IGNORE
BEQ GOBACK
CMPB R1, 115 ; MANUAL SETTING
BNE S7P101
MOV 1, @ MANUAL
RTS PC
S7P101:
CLR R3 ; OUTPUT REGISTER

117
; PROCESS EACH CHARACTER

PROCESS: SUB 60, R1
       BMI ILLNUM
       CMP 11, R1
       BLT ILLNUM
       CLC
       ROL R2 ;MULT. PREVIOUS RESULT BY 2
       ROL R3
       ROL R3 ;MULT. PREVIOUS RESULT BY 8
       ROL R3
       ADD R2, R3 ;PREVIOUS RESULT TIMES 10
       ADD R1, R3 ;ADD NEW LOW ORDER DIGIT
       MOV R3, R2 ;R2, R3 = PREVIOUS RESULT

; INPUT SUCCEEDING CHARACTERS

CHARN: TSTB @ TRDCSR ;WAIT FOR USER INPUT
       BPL CHARN
       MOVB @ TRDBUF, R1 ;CHAR. ENTERED BY USER
       MOVB R1, @ TTXBUF ;ECHO
       BIC 177600, R1 ;LIMIT TO 7 BITS
       CMPB R1, 15 ;TEST FOR RETURN
       BNE PRCESS

GOBACK: RTS PC

ILLNUM: MOV ILLNUM, R0
       JSR PC, OUTPUT
       JMP ILLNUM

S7LIST: .WORD S7LIST
       .WORD S7AAAA
       .WORD S7BBBB
       .WORD S7CCCC
       .WORD S7DDDD
       .WORD S7EEEE
       .WORD S7FFFF
       .WORD S7GGGG

; SECTION 8-3.

ONUM2 - OUTPUTS ONE BYTE WITHOUT THE PARENTHESES.
ONUM3 - OUTPUTS UP TO 999 WITHOUT BLANKS AND NO
       LEADING ZERO IN HUNDREDS PLACE.
       NFLAG TELLS WHICH MODE TO USE =0 THEN ONUM3 ELSE

ONUM2

ONUM3: CLR @ NFLAG ;SELECT MODE
       MOV NUMBUF, R1 ;SET UP BUFFER
       BR ONUM3A

ONUM2: MOV 1, @ NFLAG ;SELECT MODE
       BIC 177400, R0 ;TAKE CARE OF HIGH BYTE
       MOV NUMBUF, R1 ;OUTPUT BUFFER LOCATION
       MOVB SPACE, (R1)+ ;SPACE ON SCREEN

ONUM3A: CLR R3 ;WORKING REGISTER
       CLR R4 ;WORKING REGISTER
       MOV R0, R2 ;NUMBER -- R2

118
; PERFORM OCTAL TO DECIMAL CONVERSION.
S7L23:  SUB   144,R2
        BMI   S7P24
        INC   R3
        MOV   R2,R0
        BR   S7L23
S7P24:  MOV   R0,R2
S7L24:  SUB   12,R2
        BMI   S7P25
        INC   R4
        MOV   R2,R0
        BR   S7L24

; CONVERT EACH DIGIT TO AN ASCII CHARACTER.
S7P25:  TST   @ NFLAG
        BEQ   ON2
ON4:    ADD   260,R3
        MOVB  R3,(R1)+ ;LOAD INTO OUTPUT BUFFER
ON3:    ADD   260,R4
        MOVB  R4,(R1)+ ;LOAD INTO OUTPUT BUFFER
        ADD   260,R0
        MOVB  R0,(R1)+ ;LOAD INTO OUTPUT BUFFER
        TST   @ NFLAG
        BEQ   ON5
        MOVB  SPACE,(R1)+ ;SPACE
        MOVB  SPACE,(R1)+ ;SPACE
ON5:    CLR   (R1) ;TRAILING ZERO
        MOV   NUMBUF,R0 ;LOCATION OF BUFFER
        JSR   PC,OUTPUT ;NUMBER TO TERMINAL
        RTS   PC ;RETURN
ON2:    TST   R3
        BEQ   ON3
        BR   ON4

; SECTION 8-4. OUTPUT MESSAGE TO CRT TERMINAL
OUTPUT:  TSTB  @ TTXCSR ;OUTPUT MESSAGE
        BPL   OUTPUT
        MOVB  (R0)+,@ TTXBUF
        TSTB  (R0)
        BNE   OUTPUT
        RTS   PC

; SECTION 9. SUBROUTINE TO PERFORM SEARCH FOR BACKWALL
; IN ARRAY. THE SUBROUTINE PASSES THE ROW NUMBERS OF
; THE TOP AND BOTTOM OF THE BACKWALL TO THE CALLING
; ROUTINE THROUGH LOCATIONS ROW1 AND ROW2. A FLAG
; IS SET IF A SUITABLE IMAGE OF THE BACKWALL IS NOT
; FOUND.
SEARCH:
        CLR   @ FLAG1
        MOV   70000,R0 ;STARTING POINT
        MOV   377,R4 ;NUMBER OF ROWS
        MOVB  @ NOISE,R2 ;R2 = NOISE
        CLR   R3 ;NUMBER OF COLUMNS > NOISE
S5L1:   MOV   6,R5 ;NUMBER OF SAMPLE COLUMNS
CMP R3,R5 ;ALL COLUMNS > NOISE?
BEQ FRNTW1 ;IF YES, JUMP TO FRONT WALL

FOUND

MOV RO,RI ;RESET COLUMN POINTER
CLR R3 ;RESET NUMBER OF COLUMNS > NOISE

S5L2:
CMPB R2,(R1) ;COMPARE TO NOISE LEVEL
BPL S5P1 ;SKIP IF PIXEL <= NOISE
INC R3 ;PIXEL > NOISE

S5P1:
ADD 10000,R1 ;MOVE TO NEW COLUMN
SOB R5,S5L2 ;REPEAT FOR ALL SAMPLE COLUMNS
INC RO ;MOVE TO NEXT ROW
SOB R4,S5L1 ;TEST FOR LAST ROW
JMP S5ERR1 ;BRANCH TO ERROR1

FRNTW1:
MOV 6,R5 ;NUMBER OF SAMPLE POINTS
CMPB R3, 2 ;2 COLUMNS > NOISE?
BMI FRNTW2 ;IF NOT, BRANCH TO FRNTW2
MOV RO,RI ;RESET COLUMN POINTER
CLR R3 ;RESET PIXEL COUNT

S5L3:
CMPB R2,(R1) ;COMPARE TO NOISE LEVEL
BPL S5P3 ;SKIP IF PIXEL <= NOISE
INC R3 ;COUNT PIXELS > NOISE

S5P3:
ADD 10000,R1 ;MOVE TO NEW COLUMN
SOB R5,S5L3 ;REPEAT FOR ALL SAMPLE POINTS
INC RO ;MOVE TO NEXT ROW
SOB R4,FRNTW1 ;TEST FOR LAST ROW
JMP S5ERR1 ;BRANCH TO ERROR1

FRNTW2:
MOV R1,@ ROW3 ;STORE BOTTOM OF FRONTWALL

CLR @ ROW3+1

S5L90:
MOV 6,R5 ;NUMBER OF SAMPLE POINTS
CMPB R3,R5 ;ALL PIXELS > NOISE
BEQ BACKW1 ;IF YES, BRANCH TO BACKW1
MOV RO,R1 ;RESET COLUMN POINTER
CLR R3 ;RESET PIXEL COUNTER

S5L4:
CMPB R2,(R1) ;COMPARE TO NOISE LEVEL
BPL S5P4 ;SKIP IF PIXEL <= NOISE
INC R3 ;COUNT PIXELS > NOISE

S5P4:
ADD 10000,R1 ;MOVE TO NEW COLUMN
SOB R5,S5L4 ;REPEAT FOR ALL SAMPLE POINTS
INC RO ;MOVE TO NEXT ROW
SOB R4,S5L90 ;TEST FOR LAST ROW
JMP S5ERR1 ;BRANCH TO ERROR1

BACKW1:
DEC R1
MOV R1,@ ROW1 ;STORE TOP OF BACKWALL

CLR @ ROW1+1

S5L41:
MOV 6,R5 ;NUMBER OF SAMPLE POINTS
CMPB R3, 2 ;2 SAMPLES > NOISE?
BMI BACKW2 ;IF NOT, BRANCH TO BACKW2
MOV RO,R1 ;RESET COLUMN POINTER IN NEW ROW

CLR R3 ;CLEAR SAMPLE COUNTER

S5L5:
CMPB R2,(R1) ;COMPARE TO NOISE LEVEL
BPL S5P5 ;SKIP IF PIXEL <= NOISE

120
INC R3 ;COUNT PIXELS > NOISE
S5P5: ADD 10000,R1 ;MOVE TO NEW COLUMN
SOB R5,S5L5 ;REPEAT FOR ALL SAMPLE POINTS
INC R0 ;MOVE TO NEXT ROW
SOB R4,S5L41 ;TEST FOR LAST ROW
JMP S5ERR1 ;ON LAST ROW, BRANCH TO ERROR1
BACKW2: DEC R1
MOV R1,@ ROW2 ;STORE BOTTOM OF BACKWALL
CLR @ ROW2+1
RTS PC
; ERROR -- BACKWALL NOT FOUND
S5ERR1: MOV 1,@ FLAG1
MOV 0,@ FLAG2
MOV WIPE,R0 ;CLEAR SCREEN
JSR PC,OUTPUT
MOV EMESS1,R0 ;LOCATION OF ERROR MESSAGE
JSR PC,OUTPUT ;PRINT ERROR MESSAGE
RTS PC ;RETURN
; SECTION 11. PUTS A FRAMESTORE ON D-BUS
; THIS SUBROUTINE ALLOWS THE USER TO PLACE ONE OF
; THE FRAMESTORES ON DISPLAY.
DFRAME:
MOV WIPE,R0 ;CLEAR SCREEN
JSR PC,OUTPUT
MOV DMESS1,R0 ;LOCATION OF MESSAGE
JSR PC,OUTPUT
MOVB @ DBUS,R0 ;WHO'S ON D-BUS
JSR PC,ONUM2 ;TELL USER WHAT BUS HE'S ON
MOV DMESS2,R0
JSR PC,OUTPUT
MOVB @ DBUS,R3
JSR PC,INNUM ;RETRIEVE NUMBER
MOVB R3,@ DBUS ;STORE CHANGE
TST @ DBUS
BEQ S11P1
CLR @ QBUS
BR S11P2
S11P1: MOV 1,@ QBUS
S11P2: MOV @ QBUS,@ FRAMES
JSR PC,DISPLY ;SWITCH FRAME ON SCREEN
RTS PC
; SECTION 12. SUM OVER BACKWALL IN COLUMN
; SUBROUTINE SUMMER IS USED TO SUM VALUES OF
; THE BACKWALL IN A COLUMN. THIS GIVES US A TOTAL
; ENERGY REPRESENTATION OF THE BACKWALL. THE RESULT
; IS STORED IN A VECTOR IN MEMORY WHICH CAN THEN BE
; ACCESSED BY THE SMOOTHING AND BACKWALL DISPLAY
; SUBROUTINES.
; REGISTER USES:
; R0=PIXEL BUFFER
; R1=CUMULATIVE ROW SUM
; R2=CURRENT COLUMN

121
R3=BACKWALL START ADDRESS  
R4=COUNTER (WIDTH)  
R5=PIXEL POINTER

SUMMER:
CLR R2 ;FIRST ARRAY ELEMENT
MOV 60000,R3 ;FIRST COLUMN
ADD @ ROW1,R3 ;STARTING POINT
MOV @ ROW2,R4 ;COMPUTE WIDTH OF BACKWALL
SUB @ ROW1,R4
INC R4
MOV R4,@ WIDTH ;STORE WIDTH OF BACKWALL

; THE OUTER LOOP COMPUTES THE SUM FOR EACH OF THE
; COLUMNS IN THE FRAME.
S12L1:
CLR R1 ;SUM=0
MOV @ WIDTH,R4 ;LOAD COUNTER FOR INNER LOOP
MOV R3,R5 ;SET UP PIXEL POINTER

; THE INNER LOOP ACTUALLY PERFORMS THE SUMMING
; OPERATION FOR A PARTICULAR COLUMN.
S12L2:
MOVB (R5)+,R0 ;PIXEL TO BUFFER
ADD R0,R1 ;SUM OVER BACKWALL
S0B R4,S12L2 ;COUNT UNTIL DONE

; HERE WE CONTROL THE OUTER LOOP COUNT AND STORE THE
; SUM OBTAINED FROM THE INNER LOOP.
MOV R1,BACKWL(R2) ;MOVE SUM INTO VECTOR
ADD 400,R3 ;INCREMENT ROW POINTER
ADD 2,R2 ;INCREMENT VECTOR SUBSCRIPT
CMP 354,R2 ;TEST FOR END OF WALL
BNE S12L1 ;LOOP UNTIL DONE

; BACKWL THRU BACKWL+352 CONTAIN THE SUM OVER WALL AT
; EACH COLUMN.
S12P2:
RTS PC ;RETURN

; SECTION 14. FLAW DETECTION PROGRAM
; THIS SUBROUTINE DOES THE ACTUAL DETECTION. THE
; ROUTINE USES SUBROUTINE SEARCH TO FIND THE BACKWALL.
; IT THEN EXAMINES THE WALL IMAGE FOR SHADOWS FROM THE
; FLAWS.
; IF SUCH SHADOWS ARE FOUND AND CFLAG=1 THE PROGRAM
; ATTEMPTS TO LOCATE THE ACTUAL IMAGE OF THE FLAW AND
; HIGHLIGHT IT WITH A BOX FOR THE USER TO SEE.
; IF CFLAG=0 THEN THE FLAW PICTURE LOCATION IS LOGGED
FLAWDP:
JSR PC,SEARCH ;LOCATE BACKWALL
TSTB @ FLAG1 ;LOCATION OK?
BEQ S14P1 ;IF OK, CONTINUE
TST @ CFLAG ;IF CFLAG =0 THEN A FLAW
BNE FRET ;MAYBE PRESENT AND IS

LOGGED
JMP END ; AS A PRECAUTION
FRET:
RTS PC ;OTHERWISE, RETURN
S14P1:
JSR PC, SUMMER ; SUM OVER INNER SURFACE
JSR PC, SMOOTH ; CREATE SMOOTHED VECTOR

THIS SUBSECTION COMPUTES THE AVERAGE AND THE MINIMUM FOR THE BACKWALL. A COMPARISON OF THESE TWO NUMBERS IS THEN MADE. IF NO POINT IN THE WALL FALLS BELOW .4 OF THE AVERAGE FOR THE WALL; IT IS SAFE TO ASSUME THAT NO FLAW IS PRESENT. WE USE THIS CRITERION BECAUSE THE SHADOW THRESHOLD WILL BE SET AS APPROXIMATELY .4 OF THE OVERALL AVERAGE AND WE DON'T WANT TO WASTE TIME WITH A MORE DETAILED SEARCH IF NO VALUE WILL SATISFY THE CRITERION FOR A SHADOW.

REGISTER USES IN THIS SUBSECTION:
R0=LOCAL AVERAGE OF ARRAY
R1=MINCIMUM FROM ARRAY
R2=INDEX TO ARRAY
R3=AVERAGE OF ARRAY
R5=COUNTER

THE BACKWALL IS DIVIDED INTO 13 SECTIONS OF 8 VALUES EACH WHEN COMPUTING THE AVERAGE. THIS IS DONE SO THAT OVERFLOW OF REGISTERS DOES NOT BECOME A PROBLEM.

S14P2:
MOV SMTHBW, R2 ; INITIALIZE INDEX
MOV 15, R5 ; INITIALIZE COUNTER
CLR R3 ; INITIALIZE SUM
MOV (R2), R1 ; INITIALIZE MINIMUM

S14L1:
MOV 10, R4 ; SET UP INNER LOOP
CLR R0 ; CLEAR SUM

THIS LOOP FINDS THE MINIMUM AND AVERAGE FOR A GROUP OF 8 VALUES.

S14L2:
S14P3: CMP R1, (R2) ; LOOK FOR MINIMUM
BMI S14P4 ; STORE NEW MIN. IF NESS.
MOV (R2), R1

S14P4:
ADD (R2)+, R0 ; SUM ACROSS VALUES
SOB R4, S14L2 ; LOOP UNTIL DONE
ASR R0 ; COMPUTE LOCAL AVERAGE
ASR R0
ASR R0
ADD R0, R3 ; ADD TO OVERALL SUM
SOB R5, S14L1 ; LOOP FOR 13 SECTIONS

NOW MAKE THE COMPARISON TO SEE IF ANY MINIMUM VALUES FALL BELOW .4 OF THE OVERALL AVERAGE.

MOV R3, @ AVE ; STORE 13 TIMES AVERAGE
MOV R1, @ MIN ; STORE MINIMUM VALUE
ASL R1
ASL R1
ASL R1
ASL R1
ASL R1
CMP R1, R3 ; COMPARE AS TEST
BMI TESTFL ; CONTINUE IF R1 < R3
;
; NO FLAW FOUND IN CURRENT PICTURE.
NOFLN: TST @ CFLAG ; FULL TEST?
BEQ NOTH1 ; NO - DON'T DISPLAY A
MESSAGE
MOV FMESS1, RO ; PRINT 'NO FLAWS DETECTED'

NOTH1: RTS PC
;
; IF THE FIRST TEST INDICATES THAT A SHADOW MAY BE
; FOUND, WE COMPUTE THE THRESHOLD AS A FUNCTION OF
; THE LOCAL AVERAGE. THIS FUNCTION IS .4 TIMES THE
; AVERAGE PLUS A SHADOW NOISE FACTOR. THIS VALUE
; FOR THE THRESHOLD IS WHY WE RAN THE FIRST TEST TO
; SEE IF ANY POINT WOULD SATISFY THIS CRITERION.
TESTFL:

ASR R3 ; DIVIDE BY 32
ASR R3
ASR R3
ASR R3
ASR R3
ADD @ SHADOW, R3 ; ADD A SHADOW NOISE FACTOR
MOV SMTHBW, RO ; RO WILL POINT TO SMOOTHED
ARRAY THAT WAS FORMED BY SMOOTH ROUTINE.
MOV 150, R5 ; NUMBER OF POINTS IN ARRAY
MOV BORDRS, R1 ; BORDERS OF SHADOWS GO HERE
CLR R2 ; CLEAR COUNT OF BORDERS
;
; TEST FOR HIGH TO LOW THRESHOLD TRANSITION
S14L10:

CMP (RO)+, R3 ; COMPARE TO THRESHOLD
BPL S14P10 ; LOOP IF HIGH
MOV R0, (R1) ; LOG LOCATION OF TRANSITION
SUB 2, (R1)+
INC R2 ; INCREMENT COUNT
JMP S14P11 ; NOW TEST FOR LOW TO HIGH
;
; TEST FOR LOW TO HIGH THRESHOLD TRANSITION
S14L11:

CMP (RO)+, R3 ; COMPARE TO THRESHOLD
BMI S14P11 ; SKIP IF LOW
MOV R0, (R1) ; LOG LOCATION OF TRANSITION
SUB 2, (R1)+
INC R2 ; INCREMENT COUNT
JMP S14P10 ; REJOIN HIGH LOOP
S14P11: SOB R5, S14L11 ; CONTINUE LOW LOOP UNTIL DONE
; IT MAY BE THAT THE LAST SHADOW EXTENDS OFF THE SCREEN
; IN THIS CASE WE ADD A FINAL BORDER TO EVEN THINGS UP.
MOV R0, (R1) ; ADD FINAL BORDER IF NESS.
SUB 2, (R1)+
INC R2 ; ADD LAST BORDER TO COUNT
JMP S14P20 ; MOVE ON
S14P10: SOB R5, S14L10 ; CONTINUE HIGH LOOP UNTIL END
; AFTER ALL 150 ELEMENTS OF THE SMOOTHED ARRAY HAVE
; BEEN TESTED, AN ARRAY OF LOCATIONS WHERE BORDERS
; OCCURRED HAS BEEN CREATED. THERE IS AN EVEN NUMBER
; OF BORDERS. IF THE COUNT OF BORDERS = 0, THEN
; THERE IS NO FLAW AND WE RETURN TO THE COMMAND
; ROUTINE.
S14P20:
TST    R2
BEQ    NOTHIN

; FLAW(S) HAVE BEEN FOUND CHECK IF MORE PROCESSING
; WANTED.
; CFLAG=0 NO-LOG  CFLAG=1 YES - DRAW BOXES ECT.

TST    @ CFLAG
BNE    ARRAY
JMP    END ;NO MORE PROCESSING

; THE BORDERS ARRAY CONSISTS OF LOCATIONS IN THE
; SMOOTHED ARRAY VECTOR WHICH WE GENERATED FROM
; THE DATA IN THE FRAME. THIS SUBSECTION TRANSLATES
; THESE LOCATIONS INTO LOCATIONS IN THE FRAMESTORE.
ARRAY:
MOV    R2,R5 ;INIT. COUNT
MOV    BORDRS,RO ;LOC. OF BORDERS ARRAY
S14L30:
SUB    SMTHBW,(RO) ;SUBTRACT OFFSET OF ARRAY
MOV    7,R4 ;SHIFT INDEX TO HIGH BYTE
S14L31:
ASL    (RO)
SOB    R4,S14L31
ADD    60000,(RO) ;ADD OFFSET OF FRAMESTORE
ADD    @ ROW1,(RO) ;ADD ROW OF TOP OF WALL
SUB    4,(RO)+ ;ADD A BREATHING SPACE
SOB    R5,S14L30 ;DO FOR ALL BORDERS

; WE NOW SEARCH THE FRAMESTORE IN ORDER TO TRY TO DRAW
; BOXES AROUND THE FLAW.

; REGISTERS IN THIS SUBSECTION:
; RO=POINTER TO BORDERS ARRAY
; R1=POINTER TO FRAMESTORE
; R2=HORIZONTAL SUM IN AREA
; R3=HORIZONTAL COUNTER
; R4=VERTICAL COUNTER FOR SEARCH AREA
; R5=NUMBER OF SEARCH AREAS
; FIRST WE COMPUTE THE VERTICAL RANGE THAT WILL BE
; SEARCHED FOR FLAWS. IN DOING THIS WE WANT TO
; RESTRICT THE POINTER TO THE AREA BETWEEN THE TOP OF
; THE BACKWALL AND THE BOTTOM OF THE FRONTWALL.
BOXES:
MOV    BORDRS,RO ;BEGINNING OF ARRAY
CLR    @ PIXEL ;CLEAR PIXEL BUFFER
MOV    R2,R5 ;COMPUTE NUMBER OF SEARCH AREAS
ASR    R5
MOV    @ ROW1,R4 ;COMPUTE VERTICAL RANGE

125
SUB @ ROW3, R4
SUB 10, R4 ; MARGIN
MOV R4, @ VRANGE ; STORE VERTICAL RANGE

; NOW THE SEARCH BEGINS. FOR EACH SEARCH AREA (REG5)
; WE LOOK FOR REGIONS OF HIGH REFLECTED INTENSITY.
; THESE REGIONS ARE HIGHLIGHTED BY A BOX.
; COMPUTE THE WIDTH OF THIS SEARCH AREA FROM THE
; LOCATIONS OF ITS BORDERS NEAR THE BACKWALL.

S14L40:

MOV (R0)+, @ RSIDE ; LEFT SIDE OF AREA
MOV (R0), @ LSIDE ; RIGHT SIDE OF AREA
MOV (R0)+, R3 ; COMPUTE HORIZONTAL AREA
SUB @ RSIDE, R3
MOV 10, R1 ; DIVIDE

S14L99: ASR R3
SOB RI, SI4L99
INC R3
MOV R3, @ HRANGE ; STORE HORIZONTAL RANGE

; LIMIT THE POINTER TO THE VERTICAL RANGE. SET UP A
; FLAG WHICH WILL INDICATE IF SOMETHING WAS FOUND.

MOV @ VRANGE, R4 ; LOAD VERTICAL COUNTER
CLR @ FLAG4 ; CLEAR FLAG

; WE NOW SEARCH EACH ROW IN THE SEARCH AREA. THE
; PIXELS IN THAT ROW ARE SUMMED AND COMPARED TO THE
; SIZE OF THE ROW. IF ENOUGH INTENSITY IS FOUND, WE
; CONCLUDE THAT THIS IS PART OF A FLAW. IF NOT, THE
; SEARCH MOVES TO THE NEXT ROW.

S14L41:

MOV @ RSIDE, R1 ; SET UP POINTER
MOV @ HRANGE, R3 ; INIT. HORIZ. COUNT
CLR R2 ; INIT. SUM

S14L42:

MOVB (R1), @ PIXEL ; TAKE PIXEL
ADD @ PIXEL, R2 ; SUM OVER AREA
ADD 400, R1 ; MOVE POINTER
SOB R3, S14L42 ; COVER ROW IN AREA

; IF A FLAW HAS BEEN FOUND WE'LL BRANCH AT THIS POINT
; TO A SECTION OF THE PROGRAM THAT LOOKS FOR THE TOP
; OF THE FLAW. IF NONE HAS BEEN FOUND YET, WE CONTINUE
; WITH THE NEXT SUBSECTION; THIS LOOKS FOR THE BOTTOM
; OF THE FLAW.

TST @ FLAG4
BNE SI4P43

; HERE WE COMPARE TO SEE IF A FLAW HAS BEEN FOUND. IF
; SO, WE BEGIN TO DRAW THE VERTICAL LINES WHICH MAKE
; UP TWO SIDES OF THE BOX AROUND THE FLAW.

S14P44:

CMP R2, @ NOISE2
BMI SI4P45
MOV @ RSIDE, @ HOLD1 ; STORE BOTTOM OF FLAW
INC @ RSIDE ; POSITION CURSOR
INC @ RSIDE

126
INC @ LSIDE
INC @ LSIDE
DEC @ RSIDE
DEC @ LSIDE

; DRAW VERTICAL LINES
MOV RO,-(SP)
MOV @ RSIDE,RO
MOVB 77,(RO)
MOV @ LSIDE,RO
MOVB 77,(RO)
MOV (SP)+,RO

; POSITION CURSOR
DEC @ RSIDE
DEC @ LSIDE

; DRAW VERTICAL LINES
MOV RO,-(SP)
MOV @ RSIDE,RO
MOVB 77,(RO)
MOV @ LSIDE,RO
MOVB 77,(RO)
MOV (SP)+,RO

; PLACE LOCATION OF BOTTOM OF BOX IN LOCATION HOLD1.
; SET THE FLAG WHICH INDICATES THAT WE ARE NOW LOOKING
; FOR THE TOP OF THE FLAW.
ADD 2,@ HOLD1
MOV @ HRANGE,R3

S14P45: DEC @ RSIDE ;MOVE TO NEXT ROW UP FRAME
DEC @ LSIDE
DEC R4 ;TEST VERTICAL LIMIT
TST R4
BEQ KLUGE1 ;IF AT TOP OF RANGE, EXIT
JMP S14P55 ;IF NOT, CONTINUE SEARCH

KLUGE1: JMP S14P60 ;JUMP TO EXIT SECTION

; IN THIS SUBSECTION WE ARE LOOKING FOR THE TOP OF A
; FLAW. THE COMPARISON IS MADE FOR INTENSITY AGAIN.
; IF WE STILL HAVE ENOUGH INTENSITY TO INDICATE THE
; FLAW, WE EXTEND THE VERTICAL EDGES OF THE BOX. WHEN
; THE TOP OF THE FLAW IS FOUND, THE TOP AND BOTTOM OF
; THE BOX ARE DRAWN AND WE MOVE TO THE NEXT SEARCH AREA

S14P43:
CMP R2,@ NOISE2 ;COMPARISON
BPL S14P55 ;IF TOP FOUND, JUMP
MOV @ RSIDE,R1 ;EXTEND VERTICAL LINES
DEC @ RSIDE
DEC @ LSIDE

; DRAW VERTICAL SECTIONS
MOV RO,-(SP)
MOV @ RSIDE,RO
MOVB 77,(RO)
MOV @ LSIDE,RO
MOVB 77,(RO)
MOV (SP)+,RO

127
; POSITION CURSORS
DEC @ RSIDE
DEC @ LSIDE

; DRAW VERTICAL SECTIONS
MOV R0, -(SP)
MOV @ RSIDE, RO
MOVB 77, (RO)
MOV @ LSIDE, RO
MOVB 77, (RO)
MOV (SP)+, RO

; HERE WE DRAW THE TOP AND BOTTOM OF THE BOX.
SUB 2, R1 ; POSITION TOP OF BOX
MOV @ HRANGE, R3 ; LENGTH OF TOP LINE
S14L56:
    MOV 77, (R1) ; DRAW LINE
    ADD 400, R1 ; MOVE TO NEXT COLUMN
    SOB R3, S14L56 ; DRAW WHOLE LINE
    MOV @ HOLD1, R1 ; POSITION BOTTOM OF BOX
    MOV @ HRANGE, R3 ; LENGTH OF BOTTOM LINE
S14L96:
    MOV 77, (R1) ; DRAW LINE
    ADD 400, R1 ; MOVE TO NEXT COLUMN
    SOB R3, S14L96 ; DRAW WHOLE LINE
    CLR @ FLAG4 ; CLEAR FLAG
    JMP S14P60 ; GO BACK TO START

; FINISH VERTICAL SIDES OF BOX.
S14P55:
    MOV R0, -(SP)
    MOV @ RSIDE, RO
    MOVB 77, (RO)
    MOV @ LSIDE, RO
    MOVB 77, (RO)
    MOV (SP)+, RO
    DEC @ RSIDE
    DEC @ LSIDE
    DEC R4
    TST R4
    BEQ S14P60
    JMP S14L41

; TEST FOR END OF SEARCH AREAS. WHEN THE END IS FOUND,
; WE BRANCH TO THE CLEANUP SUBSECTION. OTHERWISE, WE
; MOVE BACK TO A NEW SEARCH AREA
S14P60:
    DEC R5 ; TEST FOR END
    TST R5
    BEQ END
    CLR @ FLAG4 ; RESET FLAG
    JMP S14L40 ; CONTINUE SEARCHES

; THIS SUBSECTION CLEANS UP, LOGS THE FLAW IF CFLAG=0
; OR CLAG=1 PRINTS A MESSAGE TO INDICATE THAT FLAW(S)
; WERE
; FOUND

;
END: 
TST @ CFLAG
BNE END1
MOV @ NFLAWS,RO ; LOG THE FLAW
MOV FLOG,R1 ; MAKE SURE THAT THE
ADD RO,R1 ; NUMBER OF FLAWS

WON'T
CMP R1, 37760 ; OVER WRITE OTHER
INFO
BGE MNYFLW ; TOO MANY FLAWS
MOV @ XLOC,FLOG(RO) ; PUT IN XLOC
INC RO
INC RO
MOV @ CANGLE,FLOG(RO) ; PUT IN ANGLE
INC RO
INC RO
MOV RO,@ NFLAWS ; STORE OFFSET FOR NEXT

IN NFLAWS
; NOTE - NFLAWS HAS FOUR TIMES THE NUMBER OF FLAWS
RTS PC ; RETURN
END1: MOV FMESS2,RO ; 'FLAW(S) DETECTED'
JSR PC,OUTPUT
RTS PC
MNYFLW: MOV FMESS3,RO ; 'TOO MANY FLAWS DETECTED'
JSR PC,OUTPUT
RTS PC

; SECTION 15. SMOOTHING SUBROUTINE
; THIS SUBROUTINE TAKES THE VECTOR BACKWL AND
; DOES AN AVERAGING OPERATION TO SMOOTH THE CURVE.
; THE RESULTING VECTOR IS STORED IN SMTHBW.
; REGISTER USES:
; R1 - TRAILING POINTER
; R2 - LEADING POINTER
; R3 - SUM
; R4 - INDEX TO SMOOOTHed BackWall vector (SMTHBW)
; R5 - COUNTER
; RO - POINTER
SMOOTH:
CLR R3 ; CLEAR SUM
MOV 10,R5 ; SET UP COUNT
MOV BACKWL,R1 ; INIT. TRAILING POINTER
MOV BACKWL+20,R2 ; INIT. LEADING POINTER
MOV R1,RO ; INIT. POINTER

; COMPUTE SUM OF FIRST EIGHT PIXELS.
S15L1:
ADD (RO)+,R3 ; SUM OVER EIGHT PIXELS
S0B R5,S15L1
; HERE WE COMPUTE THE AVERAGE, AND STORE IT IN THE
; FIRST FOUR LOCATIONS OF THE OUTPUT ARRAY.
S15P1:
MOV R3,RO
ASR RO
ASR RO

129
ASR RO
MOV RO,@ SMTHBW
MOV RO,@ SMTHBW+2
MOV RO,@ SMTHBW+4
MOV RO,@ SMTHBW+6
; SET UP LOOP FOR REMAINING VALUES. THE REST OF
; THE SUMS ARE COMPUTED IN AN OPTIMUM MANNER BY
; ONLY DEALING WITH TWO OF THE EIGHT VALUES IN
; THE SUM. THE LEADING AND TRAILING POINTERS ARE
; USED TO CHANGE THE SUM.
S15P2:
    MOV SMTHBW+10,R4
    MOV 156,R5
S15L2:
    ADD (R2)+,R3 ;ADD NEW VALUE
    SUB (R1)+,R3 ;SUBTRACT OLDEST VALUE
    MOV R3,R0 ;COMPUTE AVERAGE
    ASR R0
    ASR R0
    ASR R0
    MOV R0,(R4)+ ;STORE IN OUTPUT ARRAY
    SOB R5,S15L2 ;LOOP UNTIL END
; FILL LAST FOUR VALUES IN OUTPUT ARRAY WITH LAST VALUE
S15P3:
    MOV R0,(R4)+
    MOV R0,(R4)+
    MOV R0,(R4)+
    MOV R0,(R4)+
    RTS PC ;RETURN
; SECTION 16. INTERRUPT SERVICE
; SPACE IS ALLOCATED FOR FOUR INTERRUPT SERVICE
; SUBROUTINES.
IVECT1: JMP COMMN1
IVECT2: JMP COMMN1
IVECT3: JMP COMMN1
IVECT4: JMP COMMN1
; SECTION 17. UPLOAD FRAME STORE SUBROUTINE
; SUBROUTINE UPLODES A SINGLE IMAGE FROM
; FRAMESTORE TO THE VAX/lI SYSTEM (7/12/82)
; THE LINECLOCK INTERRUPT SWITCH MUST BE OFF
UPLOAD:
    MOV 20776,SP ;LOAD INTERRUPT VECTOR
    MOV 100,@ 176510
    MOV 400,R5 ;NUMBER OF ROWS
    MOV 157400,RO
LOOP:
    MOV RO,R1 ;OUTPUT ON ROW
    JSR PC,OTPT
    INC RO
    SOB R5,LOOP
; INFORM ME THAT UPLOAD IS DONE, THEN QUIT
MOVB 330, @ 177566
MOVB 32, @ 176516
JMP COMMN3
;RETURN
;SUBROUTINE OTPT
OTPT:
MOV 200, R4
;NUMBER OF COLUMNS
LOOP1:
CMP 100, R4
BNE DOIT
MOV 15, R2
;LOAD <CR>
MOV 100, @ 176514
;ENABLE INTERRUPT
WAIT
;WAIT FOR INTERRUPT
DOIT:
MOVB (R1), R2
;MOVE PIXEL TO BUFFER
MOV R2, R3
BIC 177770, R3
;LOW ORDER DIGIT IN REG3
BIS 60, R3
;ASCII
BIC 177707, R2
;HIGH ORDER DIGIT IN REG2
CLC
ROR R2
ROR R2
ROR R2
BIS 60, R2
;ASCII
MOV 100, @ 176514
;ENABLE INTERRUPT
WAIT
MOV R3, R2
MOV 100, @ 176514
;ENABLE INTERRUPT
WAIT
;WAIT FOR INTERRUPT
;UPDATE POINTER, COUNT, AND LOOP BACK
SUB 400, R1
SOB R4, LOOP1
;WHEN ROW IS DONE, OUTPUT <CR>, UPDATE POINTER AND
COUNT, THEN LOOP BACK
MOV 15, R2
;LOAD <CR>
MOV 100, @ 176514
;ENABLE INTERRUPT
WAIT
;WAIT FOR INTERRUPT
RTS PC
;INTERRUPT SERVICE SUBROUTINE TXINT
TXINT:
MOVB R2, @ 176516
;OUTPUT CHARACTER
BIC 100, @ 176514
;DISABLE INTERRUPT
RTI
RXINT:
MOVB @ 176512, @ 21000
;READ CHARACTER
BIC @ 177600, @ 21000
;MASK
CMP @ 21000, 23
;S ?
BNE RETI
TSTB @ 176510
BPL L1
MOVB @ 176512, @ 21000
;READ CHARACTER
BIC 177600, @ 21000
;MASK
THIS SECTION HAS MOST OF THE ROUTINES WHICH INVOLVE
POSITIONING THE ARRAY AND DISPLAYING THE FLAW LOG.
THIS COMPOSES MOST OF THE CODE WRITTEN BY RONALD
BURGIN.
LOCATE - THIS SECTION LOCATES AND DISPLAYS FLAWS
GIVEN THE FLAW LOG NUMBER. IT ALLOWS
THE USER TO ADJUST THE POSITION THAT
WILL BE PROCESSED. PICT FLAGS WHEN TO TAKE
A PICTURE (CAN TAKE WHEN PICT=0)

LOCATE: ADD 31, @ MAXX ; ADD TWO INCHES TO
MAX LENGTH
TST @ KFLAG ; WAS <S> PRESSED
WHILE IN JIGGLE?
BEQ LOCAT1 ; NO, CONTINUE
MOV (SP)+, RO ; YES POP RETURN
ADDRESS
CLR @ KFLAG ; AND RESET KEYBOARD
FLAG
LOCAT1: MOV 1, @ PICT ; DISABLE PICTURE
MOV LMI, RO ; 'ENTER FLAW'
JSR PC, OUTPUT
MOV 177777, R3 ; THIS SETS UP TEST
FOR
JSR PC, INNUM ; GET THE RESPONSE
CMP R3, 177777 ; A RETURN WITH NO
NUMBER
BEQ LOCD ; PERSON PRESSED
<RETURN>
DEC R3 ; FLAW 1 IS AT
OFFSET 0 IN LOG
ASL R3 ; MULTIPLY RESPONSE
BY 4
ASL R3 ; TO CALCULATE
DISPLACEMENT IN FLAW LOG
TST @ NFLAWS ; CHECK TO SEE IF
THERE ARE ANY FLAWS
BEQ LOCA9 ; NO THERE ISN'T
CMP R3, @ NFLAWS ; LEGAL ENTRY?
BLE GPOS ; LEGAL POSITION
ENTERED
LOCA9: JMP ILPOS ; BAD FLAW NUMBER
ENTERED
GPOS: MOV LM3, RO ; PRINT INSTRUCTIONS
JSR PC, OUTPUT
JSR PC, JIGGLE ; RESET CURRENT ANGLE
MOV FLOG(R3), @ TPOSX ; GET DESIRED
POSITION OFF LOG
MOV  1,@ MFLAG ;SET MOTOR SENSITIVE
CMP  @ TPOSX,@ DISP ;CHECK TO SEE IF AT
FAR RIGHT END
BNE  LOCNT ;NO - WE EXPECT MOTOR
TO MOVE
CLR  @ MFLAG ;NOT MOTOR SENSITIVE
EXPECTING TROUBLE
LOCNT: INC  R3 ;CALCULATE LOCATION
OF THE ANGLE IN
INC  R3 ;THE FLAW LOG.
MOV  FLOG(R3),@ TANGLE ;GET THE ANGLE
LOCLP: MOV  @ TPOSX,R2
CMP  R2, @ XLOC ;DECIDE WHICH WAY TO
GO
BEQ  LOCA2 ;ALREADY AT THE X
POSITION
BLO  MOVRT ;HAVE TO MOVE RIGHT
MOV  1, @ DIR ;SET UP TO MOVE
ARRAY LEFT
SUB  @ XLOC,R2 ;CALCULATE THE
DIFFERENCE IN COUNTS
BR  LOCMOV
MOVRT: MOV  0, @ DIR ;SET UP FOR MOVE TO
THE RIGHT
MOV  @ XLOC,R2 ;CALCULATE THE
DIFFERENCE IN COUNTS
LOCMOV: MOV  R2,R1
JSR  PC,MOTOR ;SEND THE COUNT
BR  LOCA2
;LOCD: SUB  31, @ MAXX ;SUBTRACT OFF 2 INCHES
JMP  COMMN2 ;LOCATE DONE- RETURN
TO COMMAND MODE
;NOW WAIT UNTIL ANGLE DESIRED PASSES BY
;AND WHILE WAITING CHECK FOR KEY PRESSES
;PICT CONTROLS WHEN THE PROGRAM CAN TAKE A PICTURE
;LOCA2: TST  @ KFLAG ;WAS KEY PRESSED WHILE
IN JIGGLE?
BEQ  LOCA2B ;IF SO RETURN THERE
JMP  J3
LOCA2B: INC  @ MFLAG ;SET MOTOR SENSITIVE
TST  @ PICT ;CAN TRY TO TAKE
PICTURE ?
BEQ  LOCA2C ;YES
CMP  @ CANGLE, @ TANGLE ;SEE IF CURR ANGLE
IS LESS TARGET ANGLE
BGT  LOCA2A ;NO CHECK KEYBOARD
CLR  @ PICT ;PICT CLEAR CAN TAKE
PICT WHEN POSSIBLE

133
LOCA2C: CMP @ CANGLE, @ TANGLE ; ANGLE PASSED?
BLT LOCA2A ; NO CHECK KEY BOARD
INC @ PICT ; PICT = 1 PICTURE

TAKEN
JMP LOCA3 ; GO TAKE PICTURE
LOCA2A: TSTB @ TRDCSR ; CHECK KEYBOARD
BPL LOCA2 ; NO KEY PRESSED
LOCA21: MOVB @ TRDBUF, R1 ; READ THE KEY
BIC 177600, R1
CMP 113, R1 ; UP ARROW KEY (K)
PRESSED?
BEQ LOCUP ; YES ADD 1/2 DANGLE
CMP 112, R1 ; DOWN ARROW KEY (J)
PRESSED?
BEQ LOCDW ; YES SUB 1/2 DANGLE
CMP 114, R1 ; RIGHT ARROW KEY (L)
PRESSED?
BEQ LOCRT ; YES MOVE ARRAY .24
CMP 110, R1 ; LEFT ARROW KEY (H)
INCHES RIGHT
CMP 123, R1 ; "S" PRESSED
BEQ LOCATI ; YES GO SEE WHAT USER
WANTS
CMP 120, R1 ; P PRESSED?
BEQ DISPL ; PRINT CURRENT
TARGET POSITION
BR LOCA2 ; IGNORE RESPONSE
LOCUP: MOV @ DANGLE, R1 ; CHANGE TARGET ANGLE
ASR R1 ; BY PLUS ONE HALF
DANGLE
ADD R1, @ TANGLE
CMP @ TANGLE, 23040. ; MAKE SURE ANGLE IS
VALID
BMI LOCA2
SUB 23040., @ TANGLE
BR LOCA2
LOCATI
LOCDW: MOV @ DANGLE, R1 ; CHANGE TARGET ANGLE
ASR R1 ; BY MINUS ONE HALF
DANGLE
SUB R1, @ TANGLE
BPL LOCA2 ; MAKE SURE ANGLE IS
VALID
ADD 23040., @ TANGLE
BR LOCA2
LOCATI: MOV 3, R1 ; MOVE MOTOR .24 INCH
RIGHT
CLR @ DIR
JSR PC, MOTOR
BR LOCA2
LOCLFT: MOV 3, R1 ; MOVE MOTOR .24 INCH
LEFT
MOV 1, @ DIR
JSR PC, MOTOR
BR LOCA2

DISPL: MOV POSPRT, RO ; PRINT CURRENT TARGET POSITION
JSR PC, OUTPUT
MOV @ XLOC, R1
MOV @ TANGLE, R2
JSR PC, DPOSA
JMP LOCA2

ILPOS: MOV LM2, R0 ; 'ILLEGAL FLAW NUMBER'
JSR PC, OUTPUT
JMP LOCAT1

; TAKE THE PICTURE

; LOCA3: INC @ QBUS, @ FRAMEA ; SET UP FRAMES
MOV @ QBUS, @ FRAMEA
MOV @ DBUS, @ FRAMEB
JSR PC, Sample ; TAKE SAMPLE
MOV 1, @ CFLAG ; SET FURTHER TESTING

FLAG
JSR PC, FLAWDP ; FIND FLAWS
JSR PC, CLEAN ; CLEAN UP PICTURE

BELOW BACKWALL
MOV @ QBUS, @ FRAMEA ; CLEAR THE
MOV @ QBUS, @ FRAMEB ; OTHER
JSR PC, FSCLRPF ; FRAME FOR NEXT PICTURE

PICTURE
TST @ QBUS ; SWAP WHICH FRAME WILL HAVE
BEQ LOCA4 ; NEXT PICTURE
CLR @ QBUS
INC @ DBUS
BR LOCA5 ; GO BACK AND WAIT

FOR NEXT PASS
LOCA4: CLR @ DBUS
INC @ QBUS

LOCA5: MOV 1, @ KFLAG ; SET KEYBOARD FLAG
JSR PC, JIGGLE
CLR @ KFLAG ; CLR KEYBOARD FLAG
JMP LOCA2

; SCAN - THIS ROUTINE CONTROLS THE SCANNING OF THE BILLET.
; BILLET POSITION IS HANDLED BY THE LINE CLOCK INTERRUPT ROUTINE. CFLAG IS CLEARED SO THAT THE FLAW FINDING ROUTINE WILL LOG THE FLAWS.
; A FLAW IN THE FLAW LOG IS TWO WORDS WIDE.
; THE FIRST WORD HAS THE NUMBER OF COUNTS FROM THE RIGHT OF THE FLAW. THE SECOND WORD HAS THE
ANGLE THAT THE FLAW WAS FOUND AT. NOTE THE
ANGLE IS REPRESENTED AS THE ANGLE * 64. THIS
WAS DONE SINCE INCREASED ACCURACY WAS DESIRED AND
FLOATING POINT NUMBERS AREN'T SUPPORTED.
NUMTIK HAS THE NUMBER OF LINE CLOCK INTERRUPTS
THAT WILL OCCUR IN ONE ROTATION. THIS IS USED
TO DECIDE WHEN TO MOVE THE ARRAY.

SCAN:   MOV   1,@ MFLAG   ;MOTOR SENSITIVE
        CLR   @ NFLAWS
        CLR   @ CFLAG   ;RESET FURTHER
TESTING FLAG
AOLOOP: JSR   PC,JIGGLE   ;RESET ANGLE TO 0
        CLR   @ NTICKS
AILOOP: MOV   @ QBUS,@ FRAMEA   ;SET UP FRAMES
        MOV   @ DBUS,@ FRAMEB
        JSR   PC,SAMPLE   ;TAKE A PICTURE
        JSR   PC,FLAWDP   ;CHECK FOR FLAWS
        JSR   PC,CLEAN   ;CLEAN UP PICTURE
BELOW BACKWALL
TST   @ QBUS   ;SWAP FRAMES
BEQ   SCAN99
CLR   @ QBUS
INC   @ DBUS
BR   SCAN98
SCAN99: CLR   @ DBUS
INC   @ QBUS
SCAN98: TSTB   @ TRDCSR   ;KEY PRESSED?
        BMI   SCAND
        CMP   @ NTICKS,@ NUMTIK   ;COMPLETED A
        BLT   AILOOP   ;NO
        MOV   @ XLOC,R1   ;CHECK TO SEE IF
        ADD   62,R1
        CMP   R1,@ MAXX   ;REACHED LEFT SIDE
        BGT   SCAND   ;SCAN COMPLETE
        MOV   62,R1   ;MOVE ARRAY FOUR
INCHES
        MOV   1,@ DIR
        JSR   PC,MOTOR
        CLR   @ NTICKS
SCN100: CMP   454,@ NTICKS   ;KILL 5 SECONDS TO
ALLOW ARRAY TO
        BGT   SCN100   ;STOP MOVING.
        BR   AOLOOP

;  DISPLAY THE NUMBER OF FLAWS FOUND.
;  NFLAWS WILL HAVE 4 TIMES THE NUMBER FOUND.
;  SCAND:   MOV   @ TRDBUF,RO   ;READ DUMMY KEY FOR
INCOMPLETE SCAN

136
MOV  MSCANI,RO       ;"SCAN COMPLETE"
JSR  PC,OUTPUT
MOV  @ NFAWS,RO    ;NFAWS HAS 4 TIMES
THE NUMBER OF FLAWS
ASR  RO
ACTUAL NUMBER OF FLAWS
ASR  RO
JSR  PC,ONUM3      ;CALCULATE THE
OF FLAWS
MOV  MSCAN2,RO    ;" FLAWS FOUND"
JSR  PC,OUTPUT

SCAND1: TSTB  @ TRDCSR ;WAIT FOR A KEY TO
BE PRESSED
BPL  SCAND1
JMP  COMMN2

;DISPLAY - THIS ROUTINE IS USED TO DISPLAY A FLAWS
POSITION.
;THE ROUTINE USES .08 INCHES PER COUNT. THE
ACTUAL
;COUNT IS .079 INCHES.
;R1 HAS X POS. R2 HAS ANGLE

DPOSA: MOV  R2,@ SAVE2    ;SAVES THE ANGLE
        SUB  @ DISP,R1    ;SUBTRACT OFF THE
        ASL  R1
        ASL  R1
        ASL  R1
        MOV  144,RO       ;DIVIDE BY 100 TO
CALCULATE INCHES
        JSR  PC,DIV
        MOV  R0,@ FRACT    ;HAS THE FRACTIONAL PART
        MOV  14,RO
        JSR  PC,DIV        ;DIVIDE BY 12 TO FIND
FEET AND INCHES
        MOV  R0,@ INCH     ;OUTPUT THE NUMBER OF
        MOV  R1,RO
        JSR  PC,ONUM3      ;' FT. '
        MOV  NFEET,RO
        JSR  PC,OUTPUT
        MOV  @ INCH,RO     ;OUTPUT THE NUMBER OF
        INCHES
        JSR  PC,ONUM3
        MOV  DPT,RO
        JSR  PC,OUTPUT
        MOV  @ FRACT,RO    ;OUTPUT THE FRACTIONAL
PART
        JSR  PC,ONUM3
        MOV  NINCH,RO     ;' INCHES, ANGLE='
        JSR  PC,OUTPUT
        MOV  @ SAVE2,R1    ;CALCULATE THE ANGLE
MOV 100,RO ;DIVIDE BY 64
JSR PC,DIV
MOV R1,@ ANG ;HAVE INTEGER PART IN
ANG
MOV 144,R1 ;CALCULATE FRACTIONAL
PART OF ANGLE
JSR PC,MUL ;MULTIPLY BY 100
MOV 100,RO ;DIVIDE BY 64
JSR PC,DIV
MOV R1,@ FRACT ;HAVE FRACTIONAL PART
MOV @ ANG,RO ;OUTPUT INTEGER PART
JSR PC,ONUM3
MOV DPT,RO
JSR PC,OUTPUT
MOV @ FRACT,RO ;OUTPUT FRACTIONAL PART
JSR PC,ONUM3
RTS PC
;
;CLEAN -THIS SUBROUTINE WILL CLEAR THE MEMORY
LOCATIONS
; WHICH CORRESPOND TO THE PORTION OF THE DISPLAY
; BELOW THE BACKWALL. IT IS NOT NECESSARY TO
DISPLAY
; THIS INFORMATION BECAUSE IT CORRESPONDS TO THE
; CENTER BORE OF THE BILLET, AND BESIDES IT
MAKES
; THE DISPLAY LOOK MESSY. THIS IS HOW THE
REGISTERS
; ARE USED
RO: PIXEL POINTER
R1: FIRST ADDRESS TO BE CLEARED
R2: POINTER TO TOP OF NEXT COLUMN
R3: COLUMN COUNTER
ROW2: BOTTOM OF BACKWALL
CLEAN: MOV 200,R3 ;COLUMN COUNTER
MOV 60000,RO ;START ADDRESS
MOV R0,R2
ADD 400,R2 ;COMPUTE TOP OF NEXT COLUMN
ADD @ ROW2,RO ;COMPUTE FIRST ADD TO BE
CLEARED
TST (R0)+ ;INC BY 2
BIC 1,RO ;MAKE ADDRESS A WORD BNDRY
MOV R0,R1
CLEAN: CLR (R1)+
CMP R1,R2 ;TOP OF NEXT COLUMN?
BNE CLEANA
ADD 400,R2 ;NEXT COLUMN
ADD 400,RO
MOV R0,R1
SOB R3,CLEANA
RTS PC
;
;MUL -THIS SECTION CALCULATES: RO*R1->RO,R1
; MULTIPLIES TWO SIXTEEN BIT NUMBERS IN RO AND R1,
AND
; FORMS A THIRTY-TWO BIT RESULT. RO HAS THE UPPER
WORD.
;
MUL:
  MOV  RO, @ TEMPM
  CLR  RO
  MOV  R2, -(SP)
  MOV  21, R2

MULTL:
  CLC
  ROR  RO
  ROR  R1
  BCC  MULT1
  ADD  @ TEMPM, RO

MULT1:
  SOB  R2, MULTL
  MOV  (SP)+, R2
  RTS  PC

; DIV - THIS ROUTINE CALCULATES R1/RO
; SIXTEEN BIT DIVISION.
; R1 HAS THE QUOTIENT
; RO THE REMAINDER.
;
DIV:
  MOV  RO, @ TEMPM
  CLR  RO
  MOV  R2, -(SP)
  MOV  20, R2

DIVL:
  CLC
  ROL  R1
  ROL  RO
  CMP  @ TEMPM, RO
  BHI  DIV1
  BIS  1, R1
  SUB  @ TEMPM, RO

DIV1:
  SOB  R2, DIVL
  MOV  (SP)+, R2
  RTS  PC

; HOME - THIS SECTION MOVES ARRAY ALL THE WAY TO THE
RIGHT
; SO THAT A FULL SCAN OF THE BILLET CAN BE DONE.
; HOMEA - USED AS A SUBSECTION TO MOVE ARRAY ALL THE
; WAY RIGHT WITHOUT CHANGING PARAMETERS.
;
HOME:
  MOV  7000, @ XLOC ; SET XLOC SO MOTOR
WILL MOVE
  CLR  @ NFLAWS ; NUMBER OF FLAWS = 0
  CLR  @ MFLAG ; NOT MOTOR SENSITIVE
  CLR  @ DISP ; CLEAR DISPLACEMENT

HOMEA:
  MOV  @ MRDBUF, RO ; RIGHT SWITCH HIT?
  BIC  @ MRIGHT, RO
  BNE  HOMED ; YES RETURN
  MOV  0, @ DIR ; SET DIRECTION RIGHT
  CMP  @ XLOC, @ DISP
BEQ HOMED
MOV 10000,R1 ; KILL TIME

HOME2: MOV @ MRDBUF,R0 ; AND LOOK FOR A
BIC @ MRIGHT,R0 ; RIGHT SWITCH TO
BE PRESSED
BNE HOMED
DEC R1
BNE HOME2 ; MOVE ARRAY AGAIN
MOV 1,R1
JSR PC,MOTOR
BR HOMEA

HOMED: RTS PC

; GETLAR - THIS ROUTINE GETS THE DISPLACEMENT AND LENGTH
; OF THE BILLET.
; IT ALSO CALCULATES THE D ANGLE PER CLOCK PULSE
; AND THE NUMBER OF CLOCK PULSES/REV.
;
GETLAR: MOV 77777,@ MAXX ; CLEAR PART OF SCREEN
MOV WIRE,RO
JSR PC,OUTPUT ; 'ENTER DISPLACEMENT'
MOV OFFSET,RO
JSR PC,OUTPUT
CLR R3 ; SET UP R3 FOR POSSIBLE
JSR PC, INNUM ; GET RESPONSE
TST R3 ; CHECK FOR A RETURN OR
BEQ ILLEN ; CHECK FOR LEGAL
CMP R3, 216. ; DISPLACEMENT
BGT GETLAR
INC @ DIR
MOV R3,RO
;
GETMOV: CMP R2, 100 ; NUMBER OF DIFFERENCE
BGT GMOV1 ; YES
MOV R2,R1 ; NO SEND THE REMAINING
GMOV1: MOV 100, R1 ; MOVE BY 64 COUNTS
JSR PC,MOTOR
SUB 100,R2 ; DEC DIFF COUNT BY 64

PULSES = 12.7 * INCHES
MOV 177,R1
JSR PC,MUL
MOV 12, R0
JSR PC,DIV
MOV R1,@ DISP
MOV R1,R2

COUNTS > 64

BGT GMOV1
MOV R2,R1

DIFFERENCE
JSR PC,MOTOR
CLR R2

CONDITION
BR GMOV2

GMOV2: MOV 100, R1
JSR PC,MOTOR
SUB 100,R2 ; DEC DIFF COUNT BY 64
GMOV2: TST R2 ; AT X POSITION?
       BNE GETMOV ; NO

ILLEN: MOV LENMES,RO ; 'ENTER BILLET LENGTH'
       JSR PC,OUTPUT ; GET RESPONSE
       CLR R3 ; GET RESPONSE
       JSR PC,INNUM
       TST R3
       BEQ ILLEN ; LENGTH <= 18
       CMP R3, 22 ; NO PRINT MESSAGE
       BGT ILLEN

MOV R3,RO ; CALCULATE MAX COUNT
MOV 230,R1
JSR PC,MUL
ADD @ DISP,R1
SUB 62, R1 ; SUBTRACT OFF 4

INCHES
CMP R1, 2692. ; TOTAL MUST BE 18 FT.

GETI: JSR PC,JIGGLE ; ALIGN NTICKS WITH CAM
SW
CLR @ NTICKS

GET1: JSR PC,JIGGLE
       SOB R3,GET1
       MOV @ NTICKS,R1 ; CALCULATE NUMTIK
       MOV 5,RO
       JSR PC,DIV
       MOV R1,@ NUMTIK
       MOV R1,RO ; CALCULATE DANGLE
       MOV 23040.,R1 ; 360/NUMTIK

; NOTE ANGLES ARE REPRESENTED 64 TIMES GREATER.

GETD: RTS PC

; CLKTIC - THIS IS THE INTERRUPT HANDLER FOR THE LINE C LOCK
; IT KEEPS TRACK OF THE BILLETS CURRENT POSITION
; AND IS USED TO TIME THE BILLET'S ROTATION.
; NOTE: BILLET'S ANGLE IS SIXTY-FOUR TIMES GREATER
THAN IT ACTUALLY IS. THIS IS TO ALLOW
FOR FINER ACCURACY WITHOUT FLOATING POINT
NUMBERS.

CLKTIC: INC @ NTICKS ;ADD ONE TO THE
NUMBER OF TICKS
ADD @ DANGLE, @ CANGLE ;UPDATE THE BILLETS
ANGLE
CMP 23040., @ CANGLE ; BILLETS ANGLE
BGT CLK1 ; NO
SUB 23040., @ CANGLE ; SUBTRACT OFF 360

CLK1: RTI

DLOG - THIS SECTION DISPLAYS THE FLAW LOG.
FLAWS ARE DISPLAYED 22 AT A TIME. AFTER
THE CURRENT 22 HAVE BEEN DISPLAYED, THE
PROGRAM WAITS FOR THE OPERATOR TO PRESS
A KEY. HITTING 'S' OR WHEN THE PROGRAM HAS
DISPLAYED ALL THE FLAWS, WILL RESULT IN
'THE FLAW LOG DISPLAYED' MESSAGE AND THE
PROGRAM WILL WAIT AGAIN FOR A KEY PRESS
BEFORE RETURNING TO THE MAIN PROGRAM.

DLOG: TSTB @ TTXCSR ; CLEAR THE SCREEN
BPL DLOG
MOV CLRSCN, @ TTXBUF
MOV @ NFLAWS, @ TEMP ; CALCULATE THE NUMBER
OF FLAWS
INCB @ TEMP ; NFLAWS HAS 4 TIMES
THE NUMBER
ASR @ TEMP
ASR @ TEMP
MOV 1, @ LCOUNT ; THIS IS FLAW NUMBER
CLR R5 ; FLAW LOG POINTER
DLOG1: MOV 26, @ COUNT ; PRINT OUT 22 LINES
PER SCREEN

DLOG2: TST @ TEMP
BEQ DLOGD
MOV LFCR, RO ; SEND LINE FEED AND CR
JSR PC, OUTPUT
MOV @ LCOUNT, R0 ; OUTPUT THE FLAW NUMBER
INC @ LCOUNT
JSR PC, ONUM3
MOV DFTB, RO ; '. '
JSR PC, OUTPUT
MOV FLOG(R5), R1
INC R5
INC R5
MOV FLOG(R5), R2
INC R5
INC R5
JSR PC, DPOSA ; DISPLAY POSITION
DEC @ TEMP ; DEC COUNTER
BNE DLOG2 ; NO GO DO ANOTHER ONE
MOV DLM,RO ; Press <S> TO STOP
JSR PC,OUTPUT ; OR ANY KEY TO CONTINUE

DLOG3: TSTB @ TRDCSR ; WAIT FOR A KEY STROKE
BPL DLOG3 ; GET RESPONSE
MOVB @ TRDBUF,R1
BIC 17600,R1 ; CLEAR UPPER BYTE
CMPB R1,123 ; 'S' PRESSED?
BNE DLOG1 ; NO CONTINUE

DISPLAYING LOG
DLOGD: MOV DLOGM,RO ; 'LF, CR LOG DISPLAYED
PRESS KEY TO EXIT'
JSR PC,OUTPUT

DLOG4: TSTB @ TRDCSR ; WAIT FOR KEY PRESS
BPL DLOG4
JMP COMM2

; JIGGLE- THIS SECTION JIGGLES MOTOR LEFT AND THEN
RIGHT
; SO THAT DETECTION OF THE CAM SWITCH IS
POSSIBLE
; THIS SECTION RETURNS WHEN THE CAM SWITCH HAS
TOGGLED. THIS IS USED TO TIME THE BILLET AND
ALSO TO RESET THE BILLET.

JIGGLE: CLR R5 ; R5 IS USED AS SWITCH

FLAG
MOV 177,@ MTXBUF ; MOVE LEFT AND RESET CAM
MOV @ MRDBUF,RO ; GET OLD DONE BIT STATUS
BIC @ MDONEB,RO
MOV @ MRDBUF,R1 ; WAIT FOR DONE TO GO 0
BIC @ MDONEB,R1
CMP R0,R1
BEQ J1A
BEQ J1B
GOTO 1
BIC @ MDONEB,R1
BEQ J1B
MOV 177777,RO ; KILL SOME TIME
JO: DEC R0
BNE JO
MOV 376,@ MTXBUF ; MOVE MOTOR BACK RIGHT

ENABLE CAM SWITCH
MOV @ MRDBUF,RO ; GET OLD DONE BIT STATUS
BIC @ MDONEB,RO
MOV @ MRDBUF,R1 ; WAIT FOR DONE TO GO 0
BIC @ MDONEB,R1
CMP R0,R1

143
**BEQ** J3A

**J3:**

```
BEQ J3A
MOV @ MRDBUF,RO ;GET STATUS BITS
MOV R0,R1
BIC @ MCAM,R1 ;CAM SET?
BEQ J4 ;NO
MOV 1,R5 ;YES SET FLAG
```

```
J4:
BIC @ MDONEB,RO ;MOTOR DONE?
BEQ J3 ;NO CONTINUE TO CHECK
TST R5 ;CAM SWITCH HIT
BEQ KEY ;NO WAIT FOR SET
CLR @ CANGLE ;RESET CURRENT ANGLE
RTS PC ;YES RETURN
```

**KEY:**

```
TST @ KFLAG
BEQ J3
TSTB @ TRDCSR
BPL J3
JMP LOCA21
```

; MOTOR - THIS ROUTINE HANDLES MOST OF THE INTERACTION BETWEEN THE MOTOR AND THE REST OF THE PROGRAMS;

R1 HAS THE NUMBER OF PULSES

***NOTE: THE ACTUAL NUMBER OF PULSES SENT IS R1*128

THE DIFFERENCE IS DUE TO THE HALF STEP IS USED ON THE STEPPER MOTOR.

DIR HAS THE DIRECTION TO MOVE THE ARRAY ( 0 -RIGHT 1 -LEFT)

MFLAG INDICATES WHETHER THE MOVE REQUEST IS SENSITIVE TO THE PROGRAM. (0 -NO 1 -YES).

IF YES THEN A MESSAGE WILL BE PRINTED IF

A SWITCH OR BOUNDARY WAS REACHED. IF NO THEN

THE MOVE REQUEST IS IGNORED.

REVISION AUG 09, 1983 BY RICHARD L. UNDERWOOD

DUE TO THE CHANGES IN THE INTERFACE BOARD TO THE STEPPER MOTOR CONTROLLER, THE ROUTINE MOTOR WAS REVISED AS FOLLOWS:

1) THE ORIGINAL ROUTINE SENT ONE PULSE TO THE INTERFACE BOARD WHICH REPRESENTED 0.079 INCHES ALONG THE X AXIS. THE HARDWARE NOW SENT ONLY ONE PULSE WHICH IS ONLY 0.0005 INCH ALONG THE X AXIS. TO COMPENSATE FOR THIS THE
COUNTS ARE MULTIPLIED
; BY 158.
;
; 2) IT IS POSSIBLE THAT THE NUMBER OF COUNTS WHEN
MULTIPLIED BY
; 158 WILL EXCEED THE STORAGE CAPABILITY OF A 16
BIT WORD.
; THEREFORE, WE MOVE IN MAXIMUM GROUPS OF 200 *
158.
; THE SECTION "CYCLE" WILL CHECK TO SEE THE NUMBER
OF REMAINING
; COUNTS AND TAKE APPROPRIATE ACTION.
;
; 3) THE SECTION THAT WAITS FOR THE BOARD TO ACCEPT
THE COUNTS
; WAS DELETED BECAUSE WE SEND ONLY ONE COUNT AT A
TIME.
;
; 4) THE EXIT POINT OF "MOTOR" WAS MOVED.
;
; 5) THE RAMP WAS ADJUSTED SO AS TO WORK PROPERLY.
;
MOTOR:  MOV  R1,TCOUNT ; STORE COUNTS 
TST  @ DIR ; WHICH DIRECTION? 
BEQ  MOTRT ; RIGHT 
MOV  @ MRDBUF,RO ; LEFT, CHECK LEFT SWITCH 
BIC  @ MLEFT,RO 
BNE  BNDRY ; ARRAY REACHED LEFT SIDE 
MOV  @ XLOC,RO ; CHECK TO SEE IF MOTOR
ADD  R1,RO ; WOULD MOVE PASSED END 
CMP  RO,@ MAXX 
BGT  BNDRY ; YES DONE MOVING MOTOR 
MOV  RO,@ XLOC 
BR  CYCLE 
MOTRT:  MOV  @ MRDBUF,RO ; MOVE ARRAY RIGHT - CHECK 
RIGHT SWITCH 
BIC  @ MRIGHT,RO 
BNE  BNDRY ; ARRAY REACHED RIGHT SIDE 
MOV  @ XLOC,RO ; CHECK TO SEE IF ARRAY
WOULD 
SUB  R1,RO ; BE MOVED OUT OF BOUNDS 
CMP  RO,@ DISP 
BLT  BNDRY ; YES DON'T MOVE MOTOR 
MOV  RO,@ XLOC 
CYCLE:  MOV  TCOUNT,R1 ; GET COUNTS TO MOVE 
CLR  FFLAG ; SET FLAG FOR ONLY ONE MOVE 
CMP  R1, 310 ; FFLAG IS 0 IF R1 <= 200 
BLE  MOT1 
DEC  FFLAG ; FFLAG IS -1 IF R1 > 200 
MOV  310,R1 ; SET UP FOR 200 COUNT MOVE 
SUB  R1,TCOUNT ; SUBT 200 FROM TCOUNT 
MOT1:  MOV  @ SPEED,@ TEMP
MOV 407, RO
JSR PC, MUL ;MULTIPLY BY 263 TO MOVE

.079 IN
MOV R1, @ TEMPC
MOV 1, @ SLINC
MOV @ NSTEPS, @ TEMPM ;NUMBER OF REAL HALF COUNTS IN RAMP UP
SUB @ NSTEPS, @ TEMPC ;SUBTRACT OFF COUNTS IN RAMP UP FROM TOTAL
JSR PC, MRAMPL

;************ NOW GO OVER ************

MOV @ TEMPC, R1
SUB @ NSTEPS, R1 ;SUBTRACT OFF COUNTS FOR RAMP DOWN

FOR RAMP DOWN
MOV R1, @ TEMPM
MOV @ FSPEED, @ TEMP
JSR PC, MRAMPL

;************ NOW RAMP DOWN ************

CLR @ SLINC
MOV @ NSTEPS, @ TEMPM
JSR PC, MRAMPL
TST FFLAG
BNE CYCLE ;-1, DO ANOTHER CYCLE

MOTDUN: RTS PC ;0, RETURN FROM MOTOR HERE

;************ THIS SECTION DOES THE ACTUAL MOVE
************

MRAMPL: MOV 176, R1
ADD @ DIR, R1 ;PUT IN DIRECTION BIT
MOVB R1, @ MTXBUF ;SEND THE PATTERN

; KILL SOME TIME TO FORM RAMP
MOV @ TEMP, RO

PAUSE:
DEC RO
BNE PAUSE
MOV 4, RO
TST @ SLINC ; IF SLINC =1 THEN
BEQ MOT7 ; RAMP DELAY =DELAY - 4
SUB RO, @ TEMP ; OR FSPEED WHICH EVER IS LARGER

CMP @ TEMP, @ FSPEED
BHIS MOT6
MOV @ FSPEED, @ TEMP

MOT6:
DEC @ TEMPM
BNE MRAMPL
RTS PC

MOT7:
CMP RO, @ SPEED
BHIS MOT8
ADD RO, @ TEMP
BR MOT6

MOT8:
MOV @ SPEED, @ TEMP
BR MOT6

146
; MOTOR MOVEMENT CONSTANTS
FSPEED: .WORD 60 ; THIS CONTROLS THE HIGHEST RATE
TO SEND PULSES
SPEED: .WORD 460 ; THIS CONTROLS THE SLOWEST RATE
TO SEND PULSES
NSTEPS: .WORD 100 ; THIS IS THE NUMBER OF STEPS IN
THE RAMP UP/DOWN
FFLAG: .WORD ; FLAG FOR LAST CYCLE ZERO = LAST
CYCLE
TCOUNT: .WORD ; STORAGE FOR COUNTS TO MOVE
;
; A SWITCH OR BOUNDARY REACHED IF MFLAG = 1 THEN THE
CONDITION
; WAS UNEXPECTED AND A MESSAGE IS TO BE PRINTED. THE
PROGRAM
; RETURNS CONTROL TO THE CALLING ROUTINE AFTER A SWITCH
IS PRESSED.
; IF MFLAG=0 THEN THE CONDITION WASN'T UNEXPECTED, I.E.
HOME
; ROUTINE WANTS THE LEFT SWITCH HIT, SO ROUTINE IGNORES
REQUEST.
;
BNDRY: TST @ MFLAG ; MOTOR SENSITIVE?
BEQ MOTDUN ; NO RETURN
MOV MMB,RO ; CANT MOVE MESSAGE
JSR PC,OUTPUT ; SEND THE MESSAGE
MWAIT1: TSTB @ TRDCSR ; WAIT FOR A KEY TO BE
PRESSED
BPL MWAIT1
MOV @ TRDBUF,RO ; READ DUMMY KEY
MOV MMC,RO ; CLEAR MESSAGE
JMP OUTPUT ; GOTO TO OUTPUT THERE RTS
;
; POWERU - PRINTS INSTRUCTIONS FOR POWER UP
; AND THEN WAITS FOR A KEY TO BE PRESSED.
;
POWERU: MOV PMESSU,RO
JSR PC,OUTPUT
POWL: TSTB @ TRDCSR
BPL POWL
MOV @ TRDBUF,R1
RTS PC
;
; POWERD - PRINTS INSTRUCTIONS FOR POWER DOWN
; AND THEN WAITS FOR A KEY TO BE PRESSED.
POWERD: MOV PMSSD,RO
JSR PC,OUTPUT
BR POWL
;
; SECTION XO. VARIABLES FOR MOVEMENT AND LOGGING
ROUTINES

147
CFLAG=32402 ;TELLS WHETHER TO DO A COMPLETE FLAW CHECK
CANGLE=32404 ;CURRENT ANGLE OF BILLET
NUMTIK=32406 ;NUMBER OF LINE CLOCK PULSES IN ONE REV.
DANGLE=32410 ;CHANGE IN ANGLE PER LINE CLOCK PULSE
KFLAG=32412 ;SHOULD KEYBOARD BE SENSITIVE TO ARRAY MOVE

XLOC=32416 ;CURRENT X OF ARRAY LOCATION IN COUNTS
MAXX=32420 ;MAX X POSITION THE ARRAY CAN MOVE

MCAM: .WORD 177757 ;THESE ARE THE BIT MASKS FOR SWITCHES
MDONEB: .WORD 177773
MLEFT: .WORD 177767
NFLAWS=32422 ;NUMBER OF FLAWS FOUND*4
NTICKS=32424 ;NUMBER OF TICKS
COUNT=32426 ;COUNTERS AND TEMPORARY LOCATIONS USED FOR CALCULATIONS
LCOUNT=32430
TEMP=32432
TEMPM=32434
INCH=32436
FRACT=32440
ANG=32442

TPOSX=32444 ;TARGET X POSITION FOR LOCATE
TANGLE=32446 ;TARGET ANGLE FOR LOCATE
DIR=32450 ;DIRECTION FLAG FOR MOTOR =1 LEFT 0 RIGHT
MFLAG=32452 ;MOTOR SENSITIVE FLAG
NFLAG=32454 ;NUMBER OUTPUT FLAG
DISP=32460 ;DISPLACEMENT
PICT=32462 ;PICTURE FLAG
SAVE2=32464 ;USED TO SAVE PARAMETERS
SLINC=32466 ;SLOPE FOR RAMP UP AND DOWN
TEMPC=32500 ;TEMP COUNT
FLOG=33000 ;THIS IS THE FLAW LOG DON'T PUT ANYTHING 33000->37760

;FLAWS ARE LOGGED BY LOCATION IN COUNTS AND ANGLE

; SECTION XI. VARIABLE NAMES/ MEMORY ALLOCATIONS
PSTACK=20776 ;PROGRAM STACK
CSRBUF=30000 ;BUFFER FOR ICSR
BUSENB=30000 ;WRITE ENB(1); Q-BUS/D-BUS(0)
INTENB=30001 ;DISPLAY ENB(15); INTERRUPT ENB(8)
SRDLY=30002 ;SAMPLING RATE AND DELAY
SRATE=30003 ;SAMPLING RATE
RAMPS=30004 ;RAMP GAINS: NEAR,FAR,SLOPE
GAINS=30004 ;NEAR,FAR RAMP GAINS
SLOPE=30005 ;SLOPE OF RAMP GAINS
GBUF=30006 ;TXGAIN AND RXATTN
TXGAIN=30006 ;TRANSMIT GAIN

148
RXATTN=30007 ;RECEIVER ATTENUATION
CMDCHR=30010 ;USER COMMAND CHARACTER
PIXEL=30012 ;SINGLE PIXEL BUFFER
TEMP1=30014 ;TEMPORARY RESTING PLACE
TEMP2=30016 ;TEMPORARY RESTING PLACE
FLAG1=30020 ;FLAW DETECTED?
FLAG2=30022
FRAMEA=30024 ;CONTROLS Q-BUS/D-BUS CONNECTION OF FRAMES:
             ; 0 -- FRAME0 ON Q-BUS;
             ; 1 -- FRAME0 ON D-BUS;
FRAME1 ON Q-BUS
FRAMEB=30026 ;SAME FUNCTION AS FRAMEA
NOISE=30030 ;NOISE SETTING
SHADOW=30032 ;SHADOW DETECTION SETTING
FLAG4=30034
NOISE2=30036
ROW1=30040 ;PLACE TO STORE ROW NUMBERS
ROW2=30042
ROW3=30044
WIDTH=30046
NUMBUF=30050 ;DON'T PUT ANYTHING FROM 30050 TO 30057:!!
ALPHA=30060 ;DON'T PUT ANYTHING FROM 30060 TO 30160!!
QBUS=30200
DBUS=30202
AVE=30210 ;USED IN FLAW DETECTION
MIN=30212 ;USED IN FLAW DETECTION
HOLD1=30214
VRANGE=30220
HRANGE=30222
RSIDE=30224
LSIDE=30226
MANUAL=30230 ;MANUAL CONTROL FLAG
BACKWL=31000 ;DON'T PUT ANYTHING FROM 31000 TO 31377:!!
SMTHBW=31400 ;DON'T PUT ANYTHING FROM 31400 TO 31777:!!
BORDRS=32000 ;DON'T PUT ANYTHING FROM 32000 TO 32400:!!
; SECTION X2. INTERFACE REGISTERS
ICSR=160000 ;IMAGER INTERFACE
ISRDLY=160002
IRAMPS=160004
IGAINS=160006
MRDCSR=176500 ;MOTOR INTERFACE
MRDBUF=176502
MTXCSR=176504
MTXBUF=176506
TRDCSR=177560 ;TERMINAL INTERFACE
TRDBUF=177562
SECTION X3. INTERRUPT VECTORS

STRTPC=24 ;POWER UP PC
STRTPS=26 ;POWER UP PS
PRSTHV=200 ;INTERRUPT FROM PROBE RESET
STEPRV=300 ;INTERRUPT FROM STEPPER MOTOR
TERMVR=60 ;RECEIVE INTERRUPT FROM TERMINAL
TERMVT=64 ;TRANSMIT INTERRUPT TO TERMINAL

SECTION X4. CONSTANTS

BEEP=7
CR=15
LF=12
CLRSCN=32
SPACE=40
LEFTP=50
RIGHTP=51
ESCAPE=33
EQUALS=75
BACKSP=10

SECTION X5. SYSTEM MESSAGES

WIPE: .BYTE ESCAPE,EQUALS
.ASCII '!'

.ASCII CR,LF
.ASCII '

.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
.ASCII CR,LF
ASCII 'ENTER COMMAND: '

CMESS1: .BYTE CLRSCN,ESCAPE,EQUALS
.ASCII '$'
.ASCII 'FIRST MENU COMMANDS:'
.ASCII CR,LF
.ASCII '(S) SET PARAMETERS'
.ASCII CR,LF
.ASCII '(P) ALIGN PROBE'

ASCII 'EXTRA MENU COMMANDS:'
.ASCII '(C) SINGLE SHOT FLAW DETECTION'
.ASCII '(D) REPEATED FLAW DETECTION'
.ASCII '(U) UPLOAD FRAME STORE'
.ASCII '(Q) SYSTEM SHUT OFF'
.ASCII '(F) EXIT TO FIRST MENU'
.ASCII '(S) EXIT TO SECOND MENU'
.ASCII '(H) MOVE MOTOR TO THE HOME POSITION'

ASCII 'FIRST MENU COMMANDS:'
.ASCII '(S) SET PARAMETERS'
.ASCII CR,LF
.ASCII '(P) ALIGN PROBE'

ASCII 'EXTRA MENU COMMANDS:'
.ASCII '(C) SINGLE SHOT FLAW DETECTION'
.ASCII '(D) REPEATED FLAW DETECTION'
.ASCII '(U) UPLOAD FRAME STORE'
.ASCII '(Q) SYSTEM SHUT OFF'
.ASCII '(F) EXIT TO FIRST MENU'
.ASCII '(S) EXIT TO SECOND MENU'
.ASCII '(H) MOVE MOTOR TO THE HOME POSITION'

ASCII 'FIRST MENU COMMANDS:'
.ASCII '(S) SET PARAMETERS'
.ASCII CR,LF
.ASCII '(P) ALIGN PROBE'
.BYTE CR,LF
.ASCII '(C) CLEAR FRAMESTORE'
.BYTE CR,LF
.ASCII '(D) CHANGE DISPLAYED FRAMESTORE'
.BYTE CR,LF
.ASCII '(F) FIND FLAWS'
.BYTE CR,LF
.ASCII '(Q) SYSTEM SHUT OFF'
.BYTE CR,LF
.ASCII '(M) EXTRA ROUTINES'
.BYTE CR,LF
.ASCII 'ENTER COMMAND:

MENU2: .BYTE CLRSCN,ESCAPE,EQUALS
.ASCII '$'
.ASCII 'SECOND MENU COMMANDS:
.BYTE CR,LF
.ASCII '(S) SHOW FLAW LOG'
.BYTE CR,LF
.ASCII '(L) LOOK AT FLAWS'
.BYTE CR,LF
.ASCII '(C) CLEAR FRAMESTORE'
.BYTE CR,LF
.ASCII '(D) CHANGE DISPLAYED FRAMESTORE'
.BYTE CR,LF
.ASCII '(Q) SYSTEM SHUT OFF'
.BYTE CR,LF
.ASCII '(M) EXTRA ROUTINES'
.BYTE CR,LF
.ASCII 'ENTER COMMAND:

EMESS1: .BYTE ESCAPE,EQUALS
.ASCII '!'
.BYTE BEEP
.ASCII 'ERROR -- BACKWALL NOT FOUND

CMESS2: .BYTE CLRSCN,ESCAPE,EQUALS
.ASCII '!'
.ASCII 'PARAMETER CHANGE ROUTINE'
.BYTE ESCAPE,EQUALS
.ASCII '$'
.ASCII 'CHOOSE THE PARAMETER YOU WISH TO OPEN'
.BYTE CR,LF
.ASCII 'OR EXIT THIS PROGRAM BY STRIKING <CR>'

<CR>': .BYTE CR,LF,LF
.ASCII 'PARAMETERS:
.BYTE CR,LF
.ASCII '(A) SAMPLING RATE'
.CMESS3: .BYTE CR,LF
.ASCII '(B) TRANSMIT GAIN'
.CMESS4: .BYTE CR,LF
.ASCII '(C) RECEIVER DELAY'
.CMESS5: .BYTE CR,LF

151
.ASCIZ '(D) NEAR RAMP GAIN'
CMESS7: .BYTE CR,LF
.ASCII '
CMESS8: .BYTE CR,LF
.ASCII '(E) RAMP SLOPE'
CMESS9: .BYTE CR,LF
.ASCII '(F) RECEIVER ATTENUATION'
CMESSO: .BYTE CR,LF
.ASCII '(G) SYSTEM NOISE THRESHOLD'
PMESSA: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER SAMPLING RATE (1,2,3): '
PMESSB: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER TRANSMIT GAIN (-db FROM 0 TO 30): '
PMESSC: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER DESIRED DELAY (0 THRU 63): '
PMESSD: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER NEAR RAMP GAIN: '
PMESSF: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER RAMP SLOPE: '
PMESSG: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER ATTENUATION (0 OR 1): '
PMESSH: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER BACKGROUND NOISE THRESHOLD (0 TO 20): '
PMESSI: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'ENTER SHADOW DETECTION THRESHOLD: '
EMESS2: .BYTE ESCAPE,EQUALS
.ASCII '!
.BYTE BEEP
.ASCII 'ERROR -- ILLEGAL ENTRY -- REENTER
COMMAND'
BMESS1: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'THE BACKWALL EXTENDS FROM ROW'
BMESS2: .BYTE CR,LF
.ASCII 'DOWN TO ROW'
DMESS1: .BYTE ESCAPE,EQUALS
.ASCII '!
.ASCII 'FRAMESTORE'
DMESS2: .ASCII ' IS CURRENTLY DISPLAYED'
.BYTE CR,LF
.ASCII 'ENTER CHOICE FOR DISPLAY (0 OR 1): '
BMESS3: .BYTE ESCAPE,EQUALS
.ASCII '!='
BMES4: .BYTE ESCAPE, EQUALS
       .BYTE 42, 75, 0
DINIT1: .BYTE CLRSCN, ESCAPE, EQUALS
       .ASCIZ '3 FRAMESTORE'
DINIT2: .ASCIZ 'IS CURRENTLY DISPLAYED'
DMESSO: .BYTE ESCAPE, EQUALS
       .ASCIZ '3'
FMES1: .BYTE ESCAPE, EQUALS
       .ASCIZ 'NO FLAWS DETECTED'
FMES2: .BYTE ESCAPE, EQUALS
       .ASCIZ 'FLAW(S) DETECTED'
LENMES: .BYTE CLRSCN, CR, LF
       .ASCII 'ENTER BILLET LENGTH IN FEET (1 TO 18)
       .BYTE CR, LF
       .ASCIZ 'NOTE: LENGTH + DISPLACEMENT CAN NOT BE
OVER 18 FT.'
MSCANI: .BYTE CR, LF
       .ASCIZ 'SCAN COMPLETE'
MSCAN2: .ASCIZ 'FLAWS FOUND. PRESS A KEY TO EXIT'
LFCR: .BYTE CR, LF
       .ASCIZ '
POSPRT: .BYTE ESCAPE, EQUALS
       .ASCIZ '!
       .BYTE CR, LF
       .ASCIZ 'CURRENT POSITION IS '
DPT: .ASCIZ '
DPTB: .ASCIZ '
NEFET: .ASCIZ 'FT.'
NINCH: .ASCIZ 'INCHES, ANGLE='
DLOGM: .BYTE CR, LF
       .BYTE BEEP
       .ASCIZ 'FLAW LOG DISPLAYED PRESS KEY TO EXIT'
LM1: .BYTE CR, LF
       .ASCIZ 'ENTER FLAW NUMBER OR <RETURN> TO EXIT'
LM2: .BYTE CR, LF
       .BYTE BEEP
       .ASCIZ 'ILLEGAL FLAW NUMBER ENTERED'
MMB: .BYTE BEEP, ESCAPE, EQUALS
       .ASCIZ '3 ARRAY REACHED A BOUNDARY - MOVE
IGNORED. PRESS A KEY'
MMC: .BYTE BEEP, ESCAPE, EQUALS
       .ASCIZ '3'
DLM: .BYTE CR, LF
       .ASCII 'PRESS <S> TO STOP'
       .BYTE CR, LF
       .ASCIZ 'OR ANY OTHER KEY TO CONTINUE'
FMESS3: .BYTE CR, LF
       .BYTE BEEP
       .ASCIZ 'TOO MANY FLAWS DETECTED'
       .BYTE CR, LF
       .ASCIZ 'FLAW LOG FULL'

153
LM3: .BYTE CLRSCN, CR, LF, LF, LF
.ASCII 'LOCATION MOVEMENT INSTRUCTIONS:
.BYTE CR, LF
.ASCII '(S) DO ANOTHER OR EXIT'
.BYTE CR, LF
.ASCII '(H) TO MOVE LEFT'
.BYTE CR, LF
.ASCII '(L) TO MOVE RIGHT'
.BYTE CR, LF
.ASCII '(K) TO MOVE UP'
.BYTE CR, LF
.ASCII '(J) TO MOVE DOWN'
.BYTE CR, LF
.ASCII '(P) DISPLAY PRESENT LOCATION'
MESS1: .BYTE CLRSCN, CR, LF
.ASCII 'FIVE ROTATIONS OF THE BILLET ARE BEING TIMED'
PMESSU: .BYTE CLRSCN, CR, LF
.ASCII 'INSTRUCTIONS FOR SYSTEM POWER UP'
.BYTE CR, LF
.ASCII '1 -CONNECT TV AND STEPPER CABLES'
.BYTE CR, LF
.ASCII '2 -POWER UP TV AND STEPPER MOTOR'
.BYTE CR, LF
.ASCII '3 -TURN ON SWITCH ON FRONT OF STAND'
.BYTE CR, LF
.ASCII 'PRESS A KEY WHEN COMPLETE'
PMSSD: .BYTE CLRSCN, CR, LF
.ASCII 'INSTRUCTIONS FOR SYSTEM POWER DOWN'
.BYTE CR, LF
.ASCII '1 -TURN OFF TV UNIT AND STEPPER MOTOR'
.BYTE CR, LF
.ASCII '2 -TURN OFF SWITCH ON THE BACK OF THE STAND'
.BYTE CR, LF
.ASCII '3 -DISCONNECT STEPPER MOTOR AND TV CABLES'
.BYTE CR, LF
.ASCII '4 -TURN OFF SWITCH ON FRONT OF THE STAND'
.ASCII 'PRESS ANY KEY TO EXIT'
OFFSET: .BYTE CLRSCN, CR, LF
.ASCII 'ENTER DISPLACEMENT IN INCHES (0 TO 216)'
ILNUM: .BYTE BEEP, CR, LF
.ASCII 'ILLEGAL DIGIT OR CHARACTER ENTERED.
TRY AGAIN'
TERM: .BYTE 0, 0, 0
.END
## TECHNICAL REPORT INTERNAL DISTRIBUTION LIST

<table>
<thead>
<tr>
<th>Role</th>
<th>ATTN:</th>
<th>Copies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chief, Development Engineering Branch</td>
<td>SMCAR-CCB-D</td>
<td>1</td>
</tr>
<tr>
<td>-DA</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-DP</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-DR</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-DS (Systems)</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-DS (ICAS Group)</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-DC</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-DM</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Chief, Engineering Support Branch</td>
<td>SMCAR-CCB-S</td>
<td>1</td>
</tr>
<tr>
<td>-SE</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Chief, Research Branch</td>
<td>SMCAR-CCB-R</td>
<td>2</td>
</tr>
<tr>
<td>-R (Ellen Fogarty)</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-RA</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-RM</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-RP</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>-RT</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Technical Library</td>
<td>SMCAR-CCB-TL</td>
<td>5</td>
</tr>
<tr>
<td>Technical Publications &amp; Editing Unit</td>
<td>SMCAR-CCB-TL</td>
<td>2</td>
</tr>
<tr>
<td>Director, Operations Directorate</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Director, Procurement Directorate</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Director, Product Assurance Directorate</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

**Note:** Please notify Director, Benet Weapons Laboratory, ATTN: SMCAR-CCB-TL, of any address changes.
# TECHNICAL REPORT EXTERNAL DISTRIBUTION LIST

<table>
<thead>
<tr>
<th>ASST SEC OF THE ARMY</th>
<th>NO. OF COPIES</th>
<th>NO. OF COPIES</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESEARCH &amp; DEVELOPMENT</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ATTN: DEP FOR SCI &amp; TECH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>THE PENTAGON</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WASHINGTON, D.C. 20315</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| COMMANDER | |
|-----------| |
| DEFENSE TECHNICAL INFO CENTER | 12 |
| ATTN: DTIC-DDA | | |
| CAMERON STATION | | |
| ALEXANDRIA, VA 22314 | | |

| COMMANDER | |
|-----------| |
| US ARMY MAT DEV & READ COMD | 1 |
| ATTN: DRXIB-SG | | |
| 5001 EISENHOWER AVE | | |
| ALEXANDRIA, VA 22333 | | |

| COMMANDER | |
|-----------| |
| ARMAMENT RES & DEV CTR | 1 |
| US ARMY AMCCOM | | |
| ATTN: SMCAR-FS | | |
| SMCAR-FSA | 1 |
| SMCAR-FSM | 1 |
| SMCAR-FSS | 1 |
| SMCAR-AEE | 1 |
| SMCAR-AES | 1 |
| SMCAR-AET-O (PLASTECH) | 1 |
| SMCAR-MSI (STINFO) | 2 |
| DOVER, NJ 07801 | | |

| DIRECTOR | |
|-----------| |
| BALLISTICS RESEARCH LABORATORY | 1 |
| ATTN: AMXR-TSB-S (STINFO) | | |
| ABERDEEN PROVING GROUND, MD 21005 | | |

| COMMANDER | |
|-----------| |
| MATERIEL SYSTEMS ANALYSIS ACTV | 1 |
| ATTN: DRXST-MP | | |
| ABERDEEN PROVING GROUND, MD 21005 | | |


| COMMANDER | |
|-----------| |
| COMMANDER | 1 |
| US ARMY AMCCOM | | |
| ATTN: SMCAR-ESP-L | | |
| ROCK ISLAND, IL 61299 | | |

| COMMANDER | |
|-----------| |
| COMMANDER | 1 |
| ROCK ISLAND ARSENAL | | |
| ATTN: SMCR-EM (MAT SCI DIV) | | |
| ROCK ISLAND, IL 61299 | | |

| DIRECTOR | |
|-----------| |
| DIRECTOR | 1 |
| US ARMY INDUSTRIAL BASE ENG ACTV | | |
| ATTN: DRXIB-M | | |
| ROCK ISLAND, IL 61299 | | |

| COMMANDER | |
|-----------| |
| COMMANDER | 1 |
| US ARMY TANK-AUTMV R&D COMD | | |
| ATTN: TECH LIB - DRSTA-TSL | | |
| WARREN, MI 48090 | | |

| COMMANDER | |
|-----------| |
| COMMANDER | 1 |
| US ARMY TANK-AUTMV COMD | | |
| ATTN: DRSTA-RC | | |
| WARREN, MI 48090 | | |

| COMMANDER | |
|-----------| |
| COMMANDER | 1 |
| US MILITARY ACADEMY | | |
| ATTN: CHMN, MECH ENGR DEPT | | |
| WEST POINT, NY 10996 | | |

| US ARMY MISSILE COMD | |
|---------------------| |
| REDSTONE SCIENTIFIC INFO CTR | 2 |
| ATTN: DOCUMENTS SECT, BLDG. 4484 | | |
| REDSTONE ARSENAL, AL 35898 | | |

<p>| COMMANDER | |
|-----------| |
| COMMANDER | 1 |
| US ARMY FGN SCIENCE &amp; TECH CTR | | |
| ATTN: DRXST-SD | | |
| 220 7TH STREET, N.E. | | |
| CHARLOTTESVILLE, VA 22901 | | |</p>
<table>
<thead>
<tr>
<th>NO.</th>
<th>NO.</th>
<th>COPIES</th>
<th>NO.</th>
<th>NO.</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMANDER</td>
<td>US ARMY LABCOM</td>
<td>MATERIALS TECHNOLOGY LAB</td>
<td>ATTN: SLCMT-IML</td>
<td>WATERTOWN, MA 01272</td>
</tr>
<tr>
<td>COMMANDER</td>
<td>US ARMY RESEARCH OFFICE</td>
<td>ATTN: CHIEF, IPO</td>
<td>P.O. BOX 12211</td>
<td>RESEARCH TRIANGLE PARK, NC 27709</td>
</tr>
<tr>
<td>COMMANDER</td>
<td>US ARMY HARRY DIAMOND LAB</td>
<td>ATTN: TECH LIB</td>
<td>2800 POWDER MILL ROAD</td>
<td>ADELPHIA, MD 20783</td>
</tr>
<tr>
<td>COMMANDER</td>
<td>NAVAL SURFACE WEAPONS CTR</td>
<td>ATTN: TECHNICAL LIBRARY</td>
<td>CODE X212</td>
<td>DAHLGREN, VA 22448</td>
</tr>
<tr>
<td>DIRECTOR</td>
<td>US NAVAL RESEARCH LAB</td>
<td>ATTN: DIR, MECH DIV</td>
<td>CODE 26-27, (DOC LIB)</td>
<td>WASHINGTON, D.C. 20375</td>
</tr>
<tr>
<td>COMMANDER</td>
<td>AIR FORCE ARMAMENT LABORATORY</td>
<td>ATTN: AFATL/DLJ</td>
<td>AFATL/DLJG</td>
<td>EGLIN AFB, FL 32542</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>METALS &amp; CERAMICS INFO CTR</td>
<td>BATTELLE COLUMBUS LAB</td>
</tr>
</tbody>
</table>

**NOTE:** PLEASE NOTIFY COMMANDER, ARMAMENT RESEARCH AND DEVELOPMENT CENTER, US ARMY AMCOM, ATTN: BENET WEAPONS LABORATORY, SHCAR-CCB-TL, WATERVERLIE, NY 12189, OF ANY ADDRESS CHANGES.