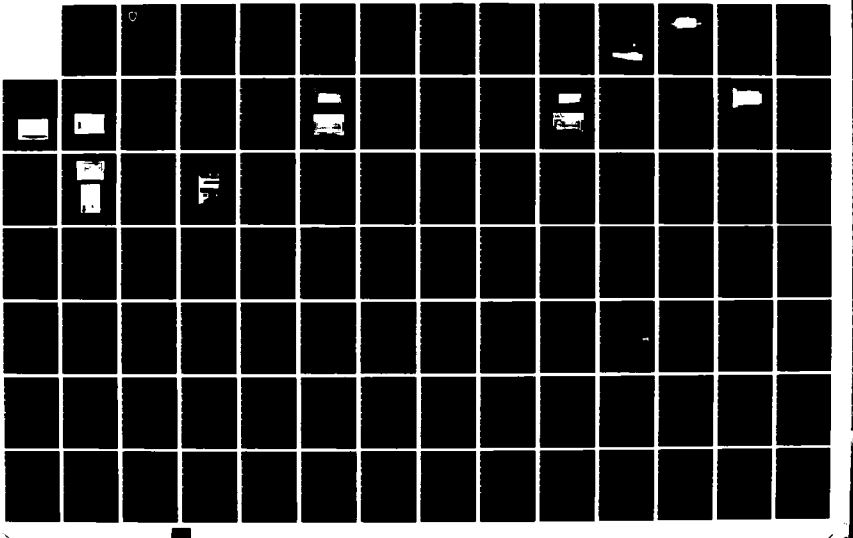
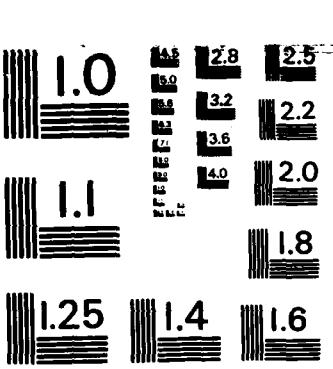
AD-A139 956 RDI TASK FINAL REPORT OF RESEARCH AND DEVELOPMENT OF SOFTWARE BALLISTIC T..(U) ABERDEEN PROVING GROUND MD MATERIEL TESTING DIRECTORATE C L FRANCIS JAN 84 1/4

UNCLASSIFIED APG-MT-5952 F/G 9/2 NL





AD NO._____
FUNDING PROJECT 1W665702D623
TECOM PROJECT NO. 5-CO-APO-DFW-203
APG REPORT NO. APG-MT-5952

RDI TASK FINAL REPORT

OF

RESEARCH AND DEVELOPMENT OF SOFTWARE,

BALLISTIC TEST SITE TERMINAL

C. L. FRANCIS

MATERIEL TESTING DIRECTORATE

US ARMY ABERDEEN PROVING GROUND
ABERDEEN PROVING GROUND, MD 21005

JANUARY 1984

Period Covered:
1 October 1979 to 31 December 1983

Prepared for:
US ARMY TEST AND EVALUATION COMMAND
ABERDEEN PROVING GROUND, MD 21005

US ARMY TEST AND EVALUATION COMMAND
ABERDEEN PROVING GROUND, MD 21005

DISTRIBUTION UNLIMITED.

DISPOSITION INSTRUCTIONS

Destroy this report when no longer needed. Do not return to the originator.

DISCLAIMER STATEMENT

The views, opinions, and/or findings in this report are those of the author(s) and should not be construed as an official Department of the Army position, unless so designated by other official documentation.

The use of trade names in this report does not constitute an official indorsement or approval of the use of such commercial hardware or software. This report may not be cited for purposes of advertisement.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER TECOM Project 5-CO-APO-DFW-203	2. GOVT ACCESSION NO. AD A139 986	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) RESEARCH AND DEVELOPMENT OF SOFTWARE, BALLISTIC TEST SITE TERMINAL		5. TYPE OF REPORT & PERIOD COVERED RDI Task, Final, 1 October 1979 to 31 December 1983
		6. PERFORMING ORG. REPORT NUMBER APG-MT-5952
7. AUTHOR(s) C. L. Francis		8. CONTRACT OR GRANT NUMBER(s) None
9. PERFORMING ORGANIZATION NAME AND ADDRESS Materiel Testing Directorate ATTN: STEAP-MT-G Aberdeen Proving Ground, MD 21005		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS None
11. CONTROLLING OFFICE NAME AND ADDRESS Commander, US Army Test and Evaluation Command ATTN: DRSTE-AD-I Aberdeen Proving Ground, MD 21005		12. REPORT DATE January 1984
		13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) None		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE None

16. DISTRIBUTION STATEMENT (of this Report)

Distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

None

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Ballistic data acquisition
Real time data analysis
Automated data acquisition
Automated data processing
Adaptive sampling rate digitizer
Programmable data acquisition equipment

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The Ballistic Test Site Terminal is a minicomputer-based digital data acquisition system developed by the Materiel Testing Directorate of Aberdeen Proving Ground. The Ballistic Test Site Terminal is an 8-, 16-, or 32-channel, voltage, charge or resistance input, 80 kHz bandwidth transient signal recording and processing system with 0.2% accuracy. The design goals, hardware configuration, and software enviroment are described. Documentation for the operational software, the analysis software, and the

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. data file format is provided. The design, development, and testing of this software were accomplished by Materiel Testing Directorate during the period of 1 October 1979 to 31 December 1983.

TABLE OF CONTENTS

A1

ABSTRACT

The Ballistic Test Site Terminal is a minicomputer-based digital data acquisition system developed by the Materiel Testing Directorate of Aberdeen Proving Ground. The Ballistic Test Site Terminal is an 8-, 16-, or 32-channel, voltage, charge or resistance input, 80 kHz bandwidth transient signal recording and processing system with 0.2% accuracy. The design goals, hardware configuration, and software environment are described. Documentation for the operational software, the analysis software, and the data file format is provided. The design, development, and testing of this software were accomplished by Materiel Testing Directorate during the period of 1 October 1979 to 31 December 1983.

FOREWORD

The Materiel Testing Directorate (MTD) of Aberdeen Proving Ground (APG) was responsible for the overall design of the Ballistic Test Site Terminal and for the development of the software. Mr. Sam Harley conceived the idea for the adaptive sampling rate digitizer (ASRD). Datacom, Inc., was responsible for the construction of the data vans and for the initial design and fabrication of the ASRD. Precision Filters, Inc., was responsible for the design and fabrication of the signal conditioner and filter/amplifier. Mr. Harry Cunningham, Mr. Sam Harley, Mr. Palmer Paules, Mrs. Genevieve Tighe, and Mr. Scott Walton all contributed to the software.

SECTION 1. BODY

1. BACKGROUND

The Ballistic Test Site Terminal (BTST) is part of the Automated Data Acquisition and Processing Technology (ADAPT) program of MTD, APG, MD. The BTST is an 8-, 16-, or 32-channel, voltage, charge, or resistance input, 80 kHz bandwidth transient signal recording and processing system with 0.2% accuracy. The reasons for establishing this program and its initial goals are contained in Reference 1. Descriptions of the initial hardware design considerations are in References 2 and 3.

The BTST consists of a unique combination of hardware and software designed to provide a comprehensive data acquisition system for ballistic events. This combination provides a very powerful instrumentation system since the ability to quickly modify the software allows flexibility in meeting changing day-to-day test requirements.

The first of three 16-channel semitrailer BTSTs was delivered to MTD in December 1979 (fig. 1-1). The BTST started supporting tests in July 1980 on a limited basis. Following receipt of signal conditioning in June 1981, the BTSTs were put into wider use on a variety of test programs. Based on field experience, changes in digitizer design to improve performance were initiated in June 1982 and completed in May 1983. An 8-channel motorized van BTST was fielded in November 1982 (fig. 1-2). An 8-channel fixed-site BTST was placed in operation in June 1983. An additional three 8-channel semitrailer BTSTs are on order to provide HAWK radar data acquisition.

Figure 1-1. Semitrailer BTST.

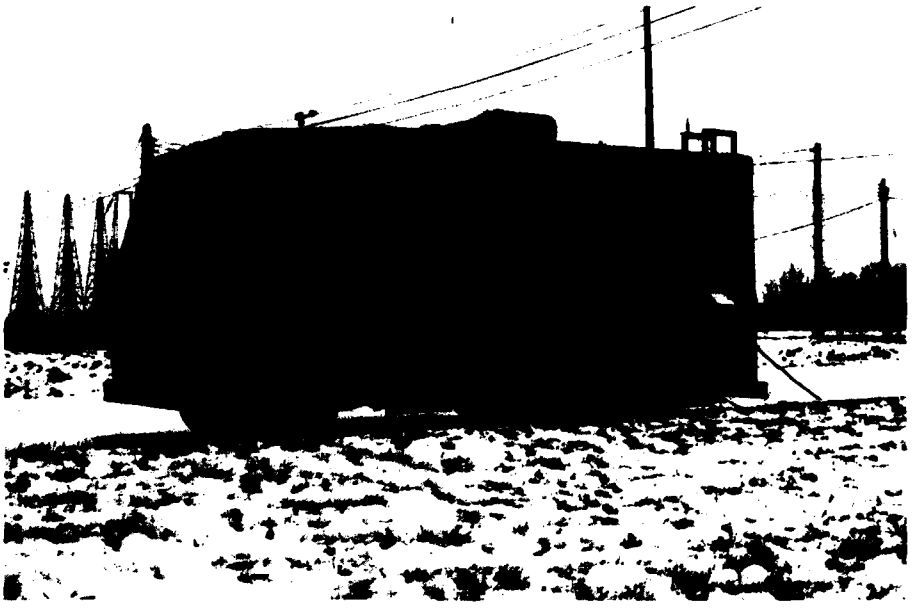


Figure 1-2. Motorized van BTST.

From 1979 to the present, the software used in the BTSTs has been continually evolving. A portion of the software was developed under the Research and Development of Instrumentation (RDI) program as a part of the research and development of software project. This report describes the current software and hardware configuration of the BTST.

2. OBJECTIVES

The objectives of this report are to:

a. Describe the hardware configuration of the BTST.

b. Document the operational software used in the BTST.

c. Document the analytical software used in the BTST.

d. Describe operational procedures for the BTST.

e. Provide information necessary to develop additional software for the BTST.

3. DETAILS OF TASK

3.1 System Design Goals

The design and/or selection of hardware components and the structure of the operational software were governed by the following goals:

a. All hardware in the system should be programmable from the computer to the maximum extent possible. This goal allows automated operation and relieves the operator of responsibility to properly set a large number of controls.

b. All software used should be menu-driven using simple operator commands. Use of complex computer operating system or editing functions should be avoided. This goal allows an operator to successfully function without the need for a great deal of sophisticated training.

c. The data file produced by the system should contain all of the available information required for subsequent processing. Separate log books or files containing additional information should not be required normally. This goal is required if truly automated data processing is desired.

3.2 Hardware Configuration

A block diagram of the BTST equipment is in Figure 3-1. The system can be divided into four basic subsystems:

a. Data acquisition equipment.

b. Built-in test equipment (BITE).

c. Computer equipment.

d. Communications equipment.

The first three subsystems will be described in detail since an understanding of these subsystems is necessary to understand the software. The communications equipment will not be described at all.

3.2 (Cont'd)

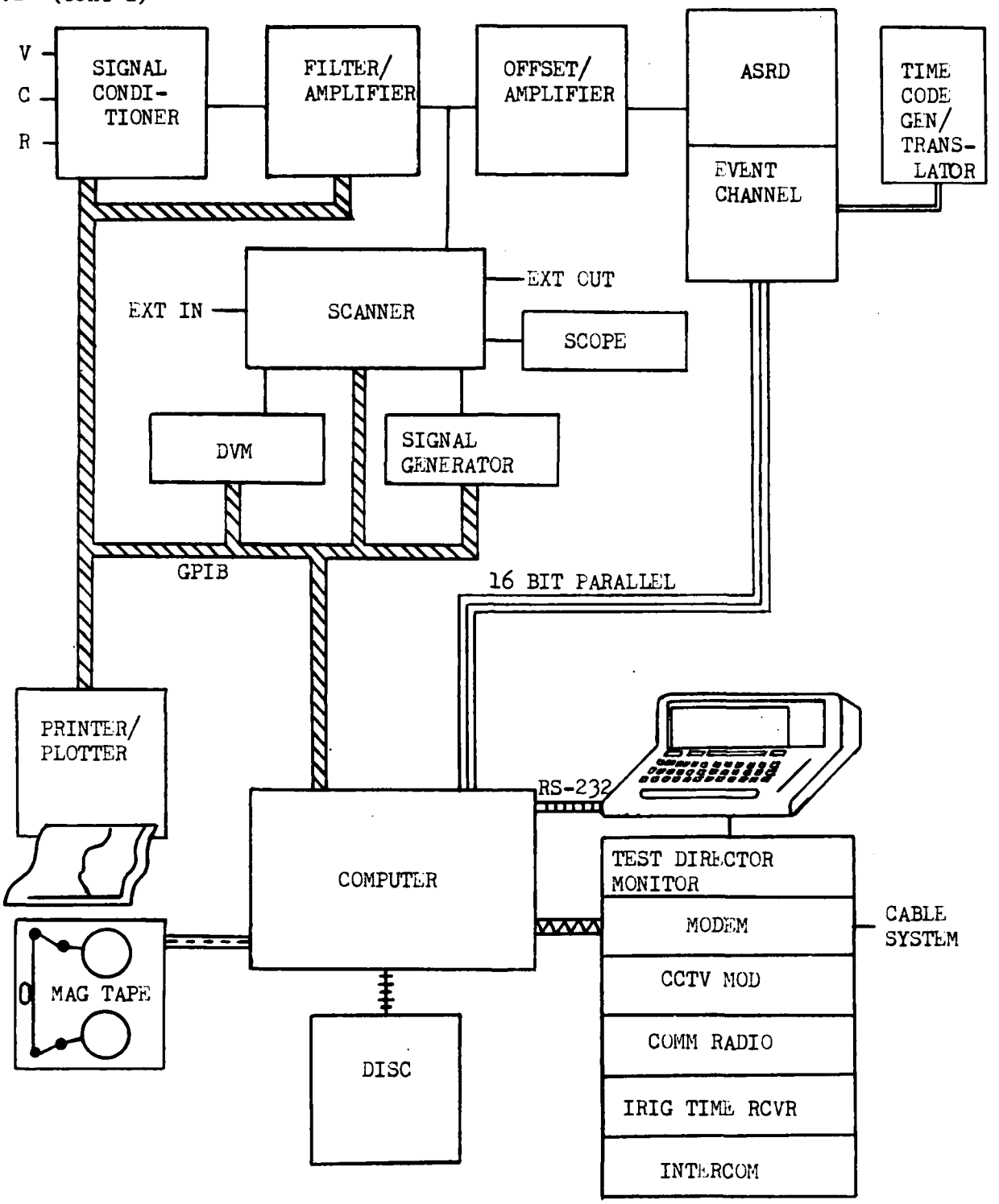


Figure 3-1. BTST block diagram.

3.2.1 Data Acquisition Equipment

The basic requirement of the BTST is to accurately record the transient signals generated by a ballistic event. The components of the data acquisition equipment perform this task.

3.2.1.1 Signal entrance panel. Signal cables from transducers enter the BTST at the signal entrance panel mounted on the exterior of the trailer. For each channel there are four inputs available:

a. A differential voltage connector.

b. A single-ended voltage connector.

c. A charge connector.

d. A multiconductor strain bridge connector.

Each input connector is wired to the appropriate point inside the BTST.

In addition to the analog signal connectors, communication and auxiliary signal connectors are also mounted on the signal entrance panel.

Figure 3-2 shows the layout of the signal entrance panel.

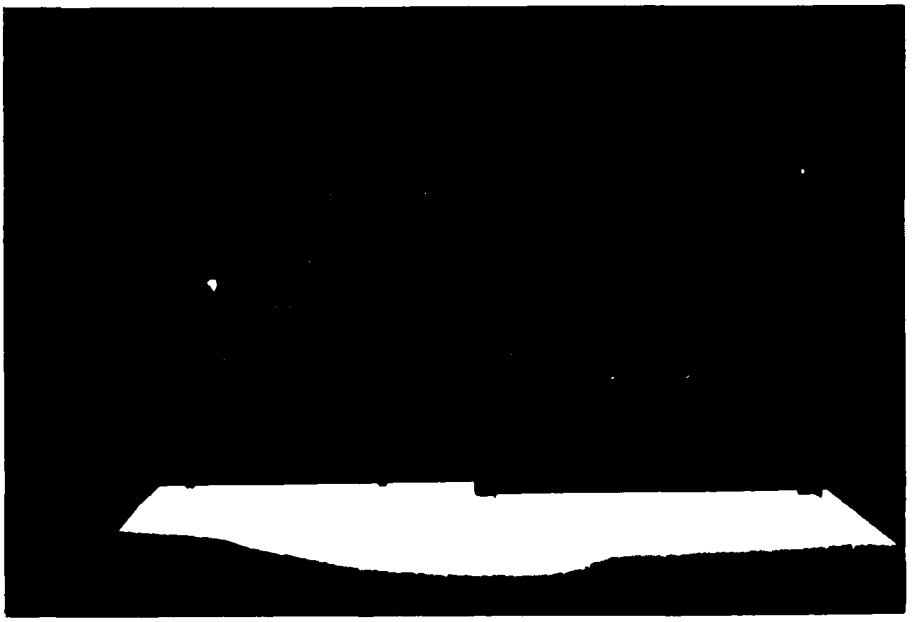


Figure 3-2. Signal entrance panel.

3.2.1.2 Voltage patch panel. A manual patch panel is provided for voltage transducers to perform the following functions:

a. Select differential or single-ended input.

b. Provide a means of inserting a cable voltage source for active transducers.

c. Provide a means to parallel the inputs of two or more channels.

d. Provide a convenient location to insert a test signal into the system.

Although a programmable patch panel would have been desirable for this function, a manual version was chosen due to its simplicity. Since this panel is not normally changed during a test, the lack of programmability at this point is not considered to compromise the goals of the system. A picture of the Trompeter model 1505-0606 panel used is in Figure 3-3. Figure 3-4 shows how the panel is configured in the system.

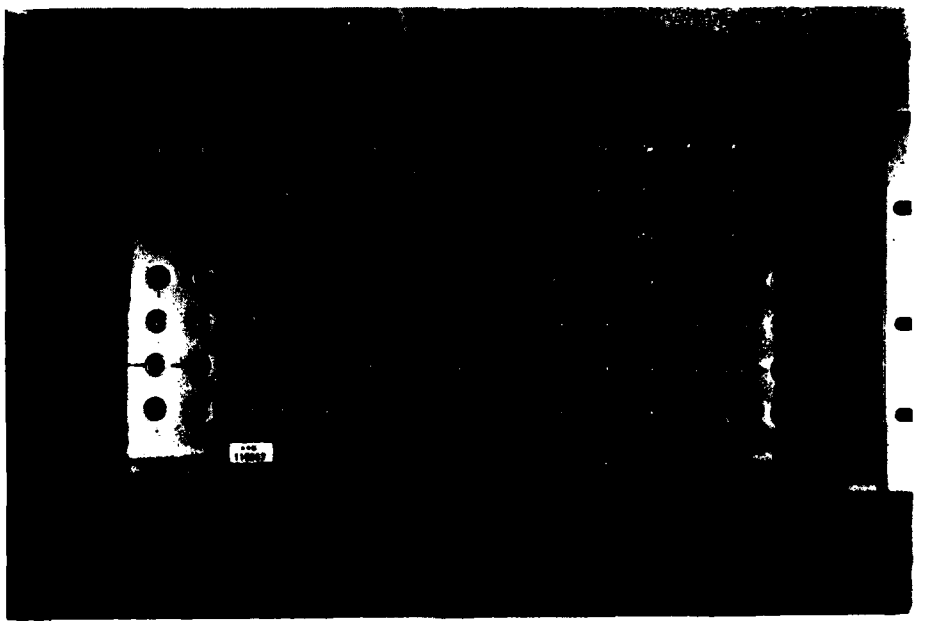


Figure 3-3. Voltage patch panel.

3.2.1.2 (Cont'd)

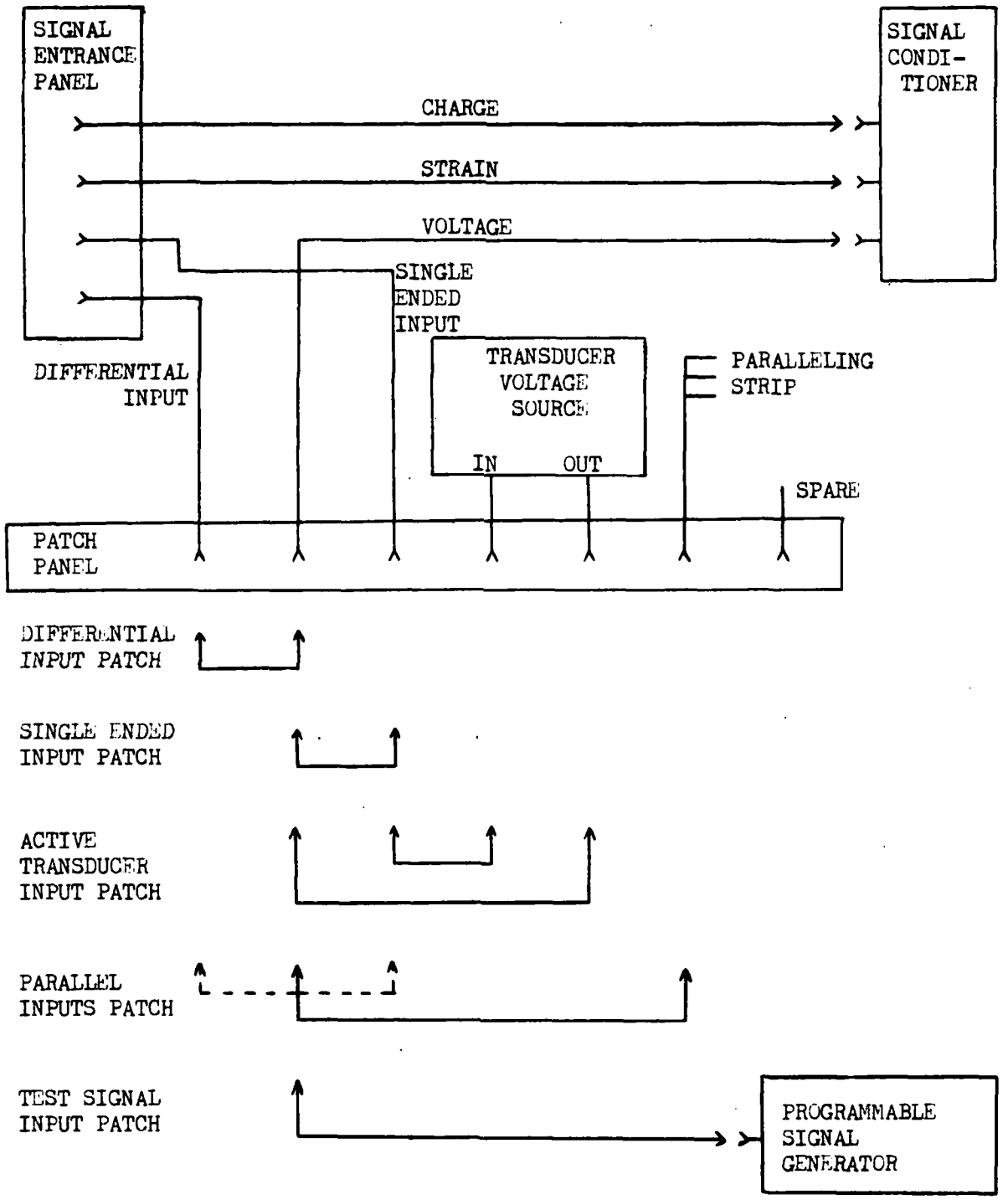


Figure 3-4. Patch panel interconnections.

3.2.1.3 Signal conditioner. A variety of transducers is required to perform ballistic measurements. These transducers fall into three categories: voltage, charge, and resistance. The signal conditioner converts a charge or resistance input to a voltage output. Although a number of commercial signal conditioning modules exists, they do not allow for programmable control. Therefore, APG initiated a signal conditioner design to meet its requirements. A block diagram of the final configuration is in Figure 3-5. A picture of a single channel's printed circuit card is in Figure 3-6, and a picture of the housing for the cards is in Figure 3-7. Important characteristics of the signal conditioner are in Table 3-1. A detailed specification is in Appendix A. The features which make the signal conditioner unique are:

a. Sixteen channels in a single unit.

b. Voltage, charge, and strain in a single unit.

c. Complete programmability from a remote controller.

d. Isolated power supplies for each channel's charge and strain electronics.

e. Manual or automatic strain bridge balance.

f. Removable jumpers for breaking ground loops.

The signal conditioner is available commercially as Precision Filters, Inc., model 316. A complete description is in Reference 4.

3.2.1.3 (Cont'd)

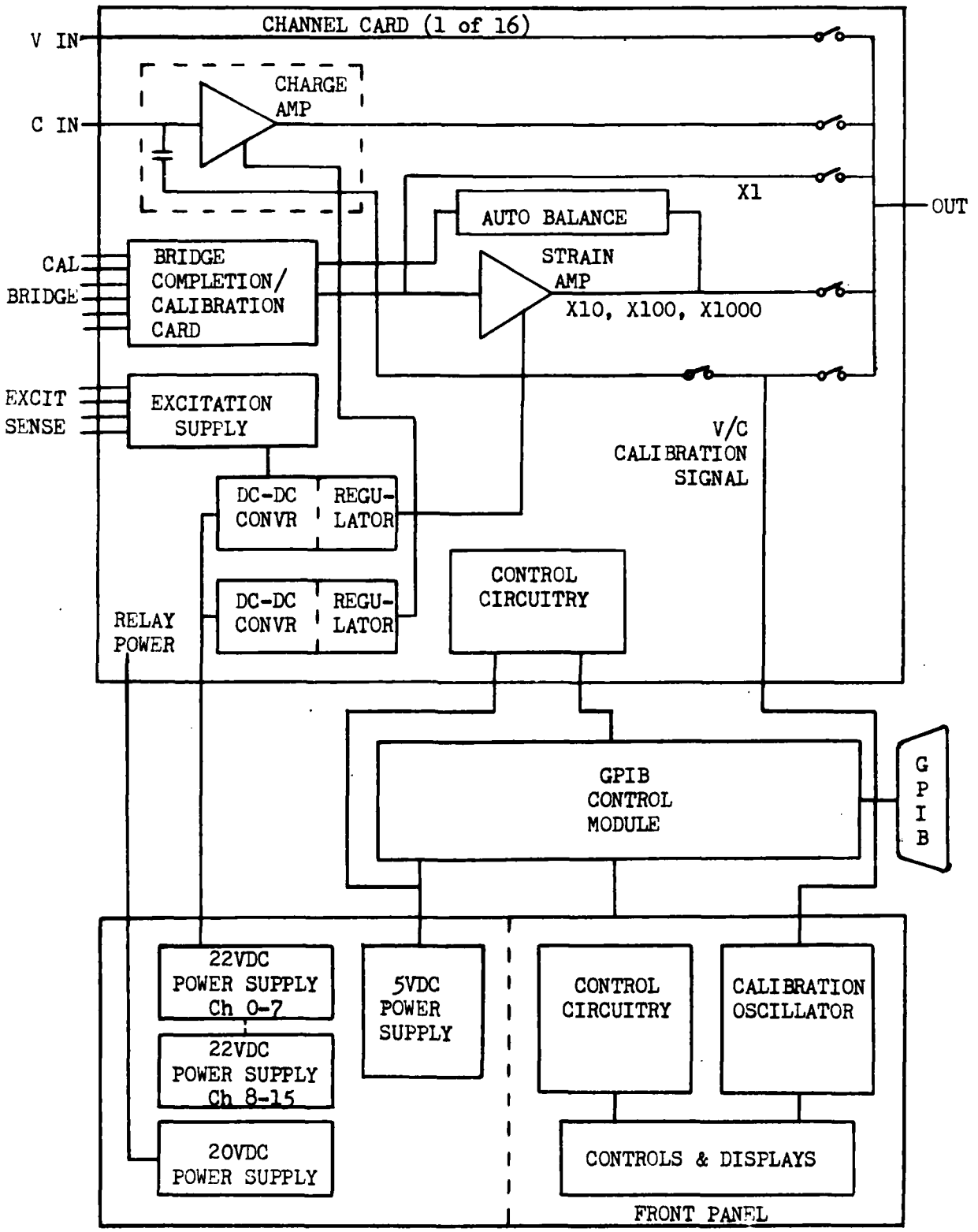


Figure 3-5. Block diagram of the signal conditioner.

3.2.1.3 (Cont'd)

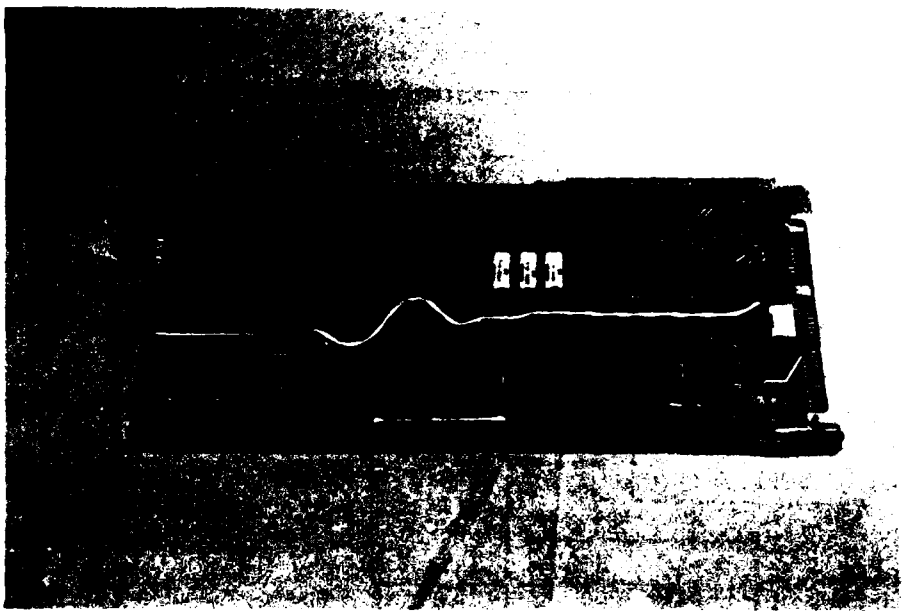


Figure 3-6. Signal conditioner printed circuit card.

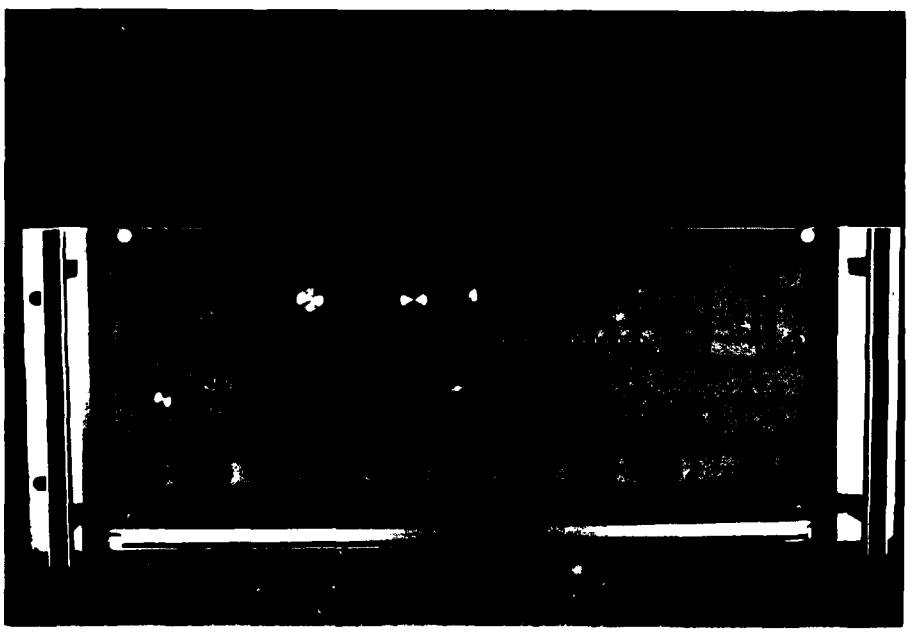


Figure 3-7. Signal conditioner main frame.

TABLE 3-1. SIGNAL CONDITIONER CHARACTERISTICS

Number of channels: 16

Programmability: GPIB or front panel

Modes:

Voltage - DC to 80 kHz

Charge - 0.1 mV/pC
1 Hz to 80 kHz
100,000 pC maximum input

Strain - X1 DC to 80 kHz
X10 DC to 25 kHz
X100 DC to 2.5 kHz
X1000 DC to 250 Hz
0.2 to 30 V excitation at 100 mA
1, 2, or 4 arm bridge
Manual or automatic balance

Power: Each channel isolated.

Calibrator: Voltage - 0 to 8 V in 7.8 mV steps.
Charge - 0 to 80,000 pC in 78 pC steps.
Strain - 7 steps shunt/128 steps series.

The signal conditioner has been found to be a very useful and flexible device. The primary shortcomings in the unit are:

a. The linearity of the digital to analog converter (DAC) used in the voltage/charge calibrator is not adequate to support a first or second order least squares fit with residuals less than 0.1%. Thus, use of the calibrator requires a table for each value.

b. The circuit card containing the bridge completion and calibration resistors is inconvenient to change in a field environment.

c. The charge calibration capacitor is sealed inside the charge amplifier making calibration of the capacitor impossible.

3.2.1.4 Filter/amplifier. Once the signal has passed through the signal conditioner, it may be necessary to amplify the signal to a usable level. Also, prior to digitizing, the signal must be low-pass filtered to prevent aliasing. A description of aliasing and filter considerations is contained in Appendix B. Although a number of commercial filters and amplifiers exist, they do not allow for programmable control. Therefore, APG initiated a filter/amplifier design combining both functions into a single package to meet its requirements. A block diagram of the final configuration is in Figure 3-8.

3.2.1.4 (Cont'd)

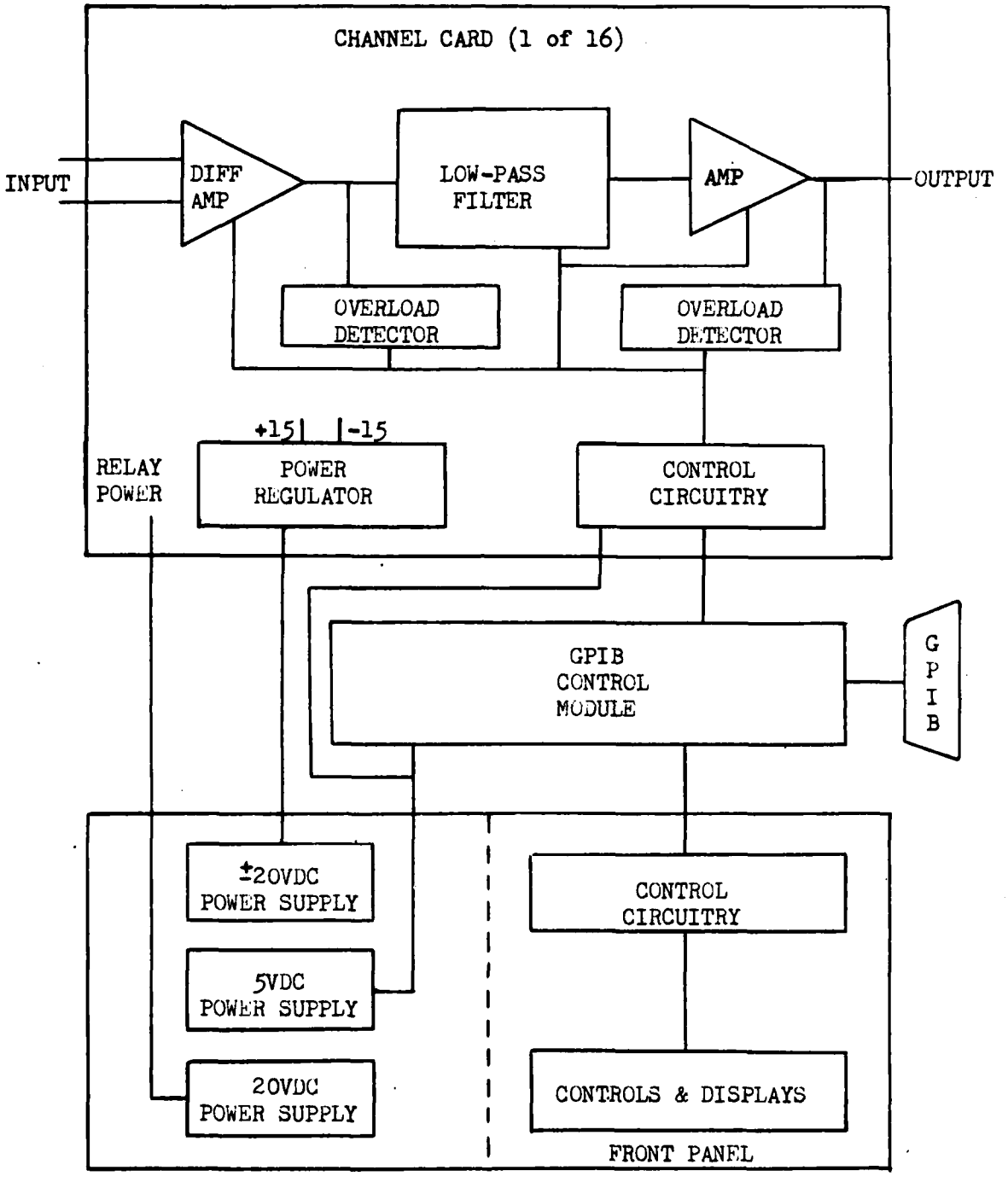


Figure 3-8. Block diagram of filter/amplifier.

3.2.1.4 (Cont'd)

A picture of a single channel's printed circuit card is in Figure 3-9, and a picture of the housing for the cards is in Figure 3-10. Important characteristics of the filter/amplifier are in Table 3-2. A detailed specification is in Appendix C. The features which make the filter/amplifier unique are:

a. Sixteen channels in a single unit.

b. Filter and amplifier in a single unit.

c. Complete programmability from a remote controller.

d. Overload detection capability.

e. Removable jumpers for breaking ground loops.

The filter/amplifier is available commercially as Precision Filters, Inc., model 416. A complete description is available in Reference 5.

The only shortcoming in the unit is that available gain steps are too coarse, causing a loss of resolution in the final data.

3.2.1.4 (Cont'd)

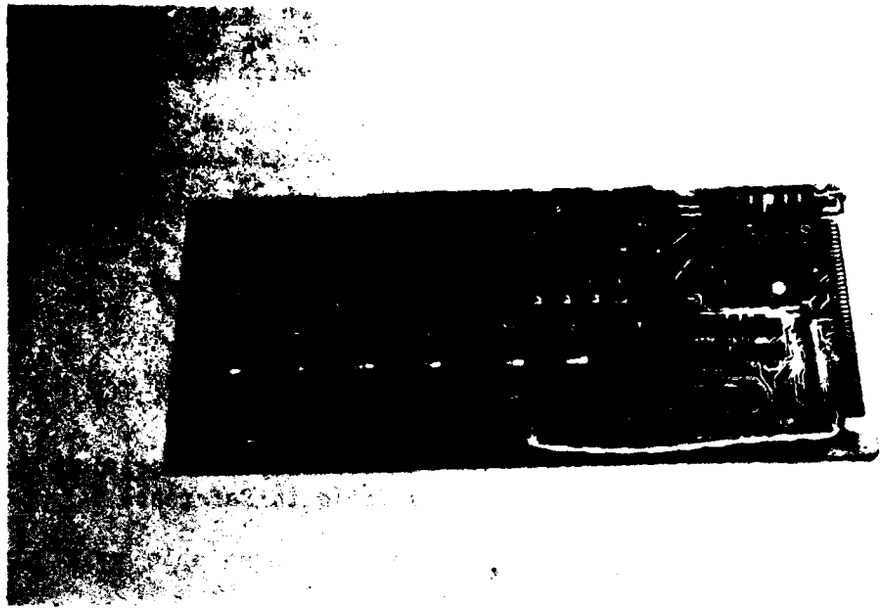


Figure 3-9. Filter/amplifier printed circuit card.

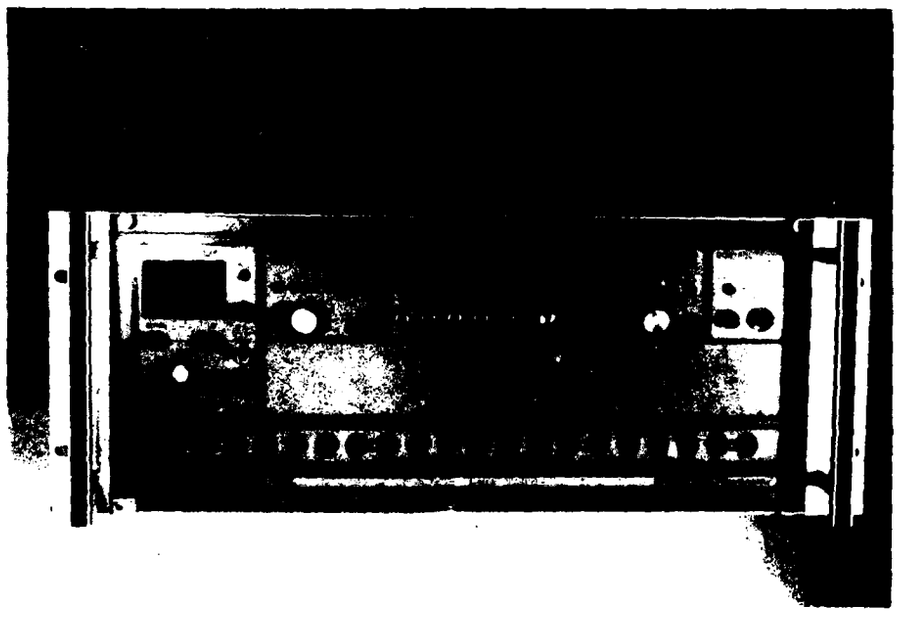


Figure 3-10. Filter/amplifier main frame.

TABLE 3-2. FILTER/AMPLIFIER CHARACTERISTICS

Number of channels: 16

Programmability: GPIB or front panel

Type input: Differential

Input impedance: 20 million ohms

Prefilter gains: 1, 2, 5, 10, 20, 50

Postfilter gains: 1, 2, 5, 10, 20

Total gain combinations: 15

Filter cutoff frequencies: 0.5 to 15.5 kHz in 0.5 kHz steps
5 to 155 kHz in 5 kHz steps

Overload detector threshold: ±10 V

Overload detector transient response: 2 microseconds

Type output: Single ended

Output impedance: 51 ohms

Three types of low-pass filters are available for use in the filter/amplifier. One is a six pole, six zero elliptic filter providing 80 dB per octave rolloff but poor step response. The other two are a six pole Bessell filter providing 32 dB per octave rolloff or a modified Bessell with six added zeroes providing 52 dB per octave rolloff with both having good step response. The choice of filter type is a tradeoff between rolloff and step response, which is a function of the type of test. A discussion of the tradeoffs involved is in Appendix B.

3.2.1.5 Offset/amplifier. The original design for the BTST did not include an offset/amplifier. However, as experience was gained with the system, the need to obtain additional resolution on many tests was encountered. This requirement arises from the fact that the majority of ballistic signals are unipolar. Since the digitizer is bipolar, one-half the resolution of the digitizer is immediately sacrificed. Secondly, the discrete gain combinations of the filter/amplifier may not allow a full scale deflection without risk of clipping when the normal round-to-round amplitude variations are taken into account. Thus, overall, for the average signal, only one-third of the available resolution of the system is actually utilized. The reduced resolution is of particular concern when the difference between two similar channels is taken.

One method which can be used to improve the resolution of unipolar signals is to offset the signal to the opposite polarity and then amplify the signal to obtain nearly full scale utilization. Unfortunately, no commercially

3.2.1.5 (Cont'd)

manufactured offset/amplifier meeting the requirements of the BTST could be located. Therefore, a custom design is currently being procured. The offset/amplifier will have the following characteristics:

a. Sixteen channels in a single unit.

b. Offset from -8 V to +8 V in 0.25 V steps.

c. Gain steps from 1 to 7.5 in 0.25 steps.

d. Overload detection capability.

e. Bypass mode when not required.

f. Complete programmability from a remote controller.

While waiting for delivery of this unit, an interim offset/amplifier has been designed and fabricated in-house. This unit does not possess the programmability and adjustability that the final design will possess. However, it allows higher resolution information to be acquired now.

The block diagram of the interim offset/amplifier is in Figure 3-11. A picture of the unit is in Figure 3-12. Important characteristics of the offset/amplifier are:

Bandwidth: 200 kHz.

Offset: -4.1 V.

Gain: 2.

Number of channels: 4.

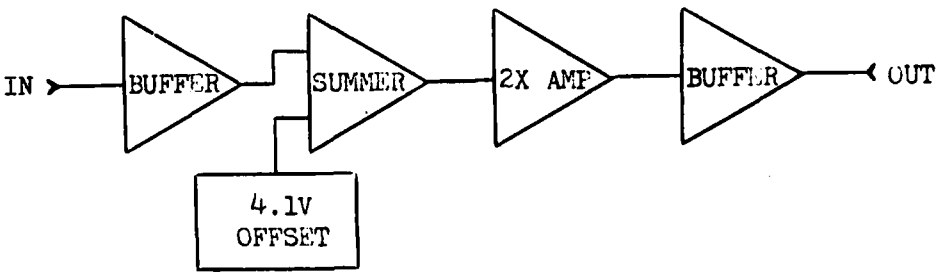


Figure 3-11. Block diagram of offset/amplifier.

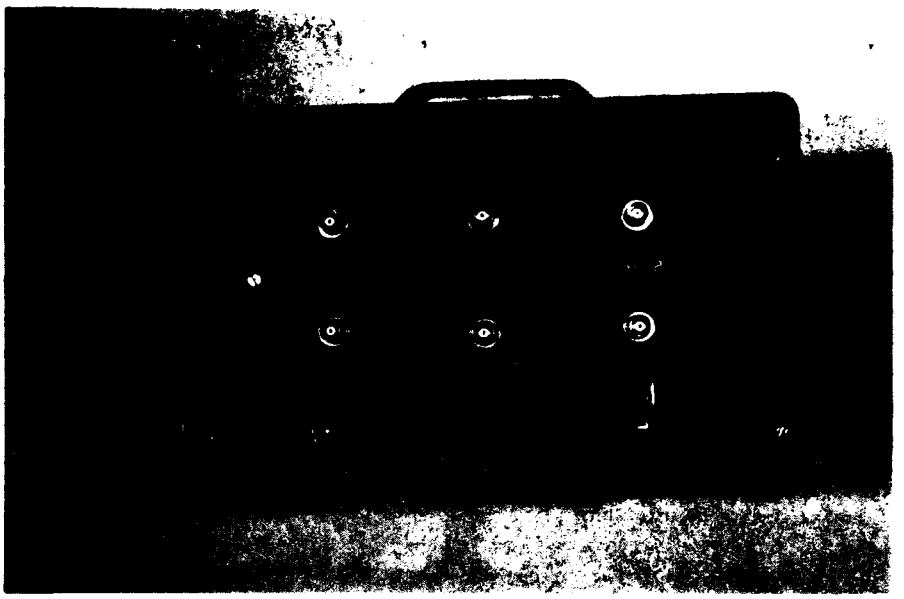


Figure 3-12. Offset/amplifier.

3.2.1.6 Adaptive sampling rate digitizer (ASRD). After the input signal has been conditioned, filtered, and amplified, it is ready to be stored for later processing. In previous data acquisition systems, this was accomplished using an analog, frequency modulation (FM) tape recorder. The tapes were digitized after the test and processed in a batch environment. Advances in digital technology made the development of an on-site multichannel digitizer feasible in the late 1970s. Table 3-3 is a comparison of tape recorder and digitizer characteristics.

TABLE 3-3. COMPARISON OF TAPE RECORDER AND DIGITIZER CHARACTERISTICS

Characteristic	Tape Recorder	Digitizer
Accuracy	1%	0.1% (10 bits) 0.025% (12 bits)
Longest record	200 seconds	1 second
Triggering	Not required - start recorder prior to event and stop after event. Do not need to know signal characteristics.	Required due to short record length. Must know signal characteristics.

3.2.1.6 (Cont'd)

TABLE 3-3 (CONT'D)

Characteristic	Tape Recorder	Digitizer
Quick look capability	Oscilloscope or oscillograph	Scaled plots and analysis routines.
Documentation	Voice channel on tape. Separate log books required.	Extensive capability. Sketches and pictures not usable.
Processing	Requires digitizing.	Ready to use.
Exchangability	IRIG standard.	Nonstandard format.

The primary strengths of the tape recorder are the large storage capacity and not needing to know the signal characteristics. Although a number of commercial digitizers were available, none allowed the disadvantages detailed above to be overcome. For this reason, APG initiated a digitizer design. A block diagram of the final design is in Figure 3-13. A picture of a single channel assembly is in Figure 3-14, and a picture of the events channel and ASRD housing is in Figure 3-15. Important characteristics of the digitizer are in Table 3-4. A detailed specification is in Appendix D. The features which make the digitizer unique are:

a. Sixteen channels in a single unit.

b. Each channel completely independent of all others.

c. Complete programmability from a remote controller.

d. Unique triggering modes.

e. All channels synchronized and time tagged.

f. Adaptive sample rate capability.

g. Separate event time of occurrence channel.

The ASRD was custom fabricated for APG by Datacom, Inc., and is not available as a commercial product. A description of the ASRD hardware, firmware, and software can be found in Reference 6.

3.2.1.6 (Cont'd)

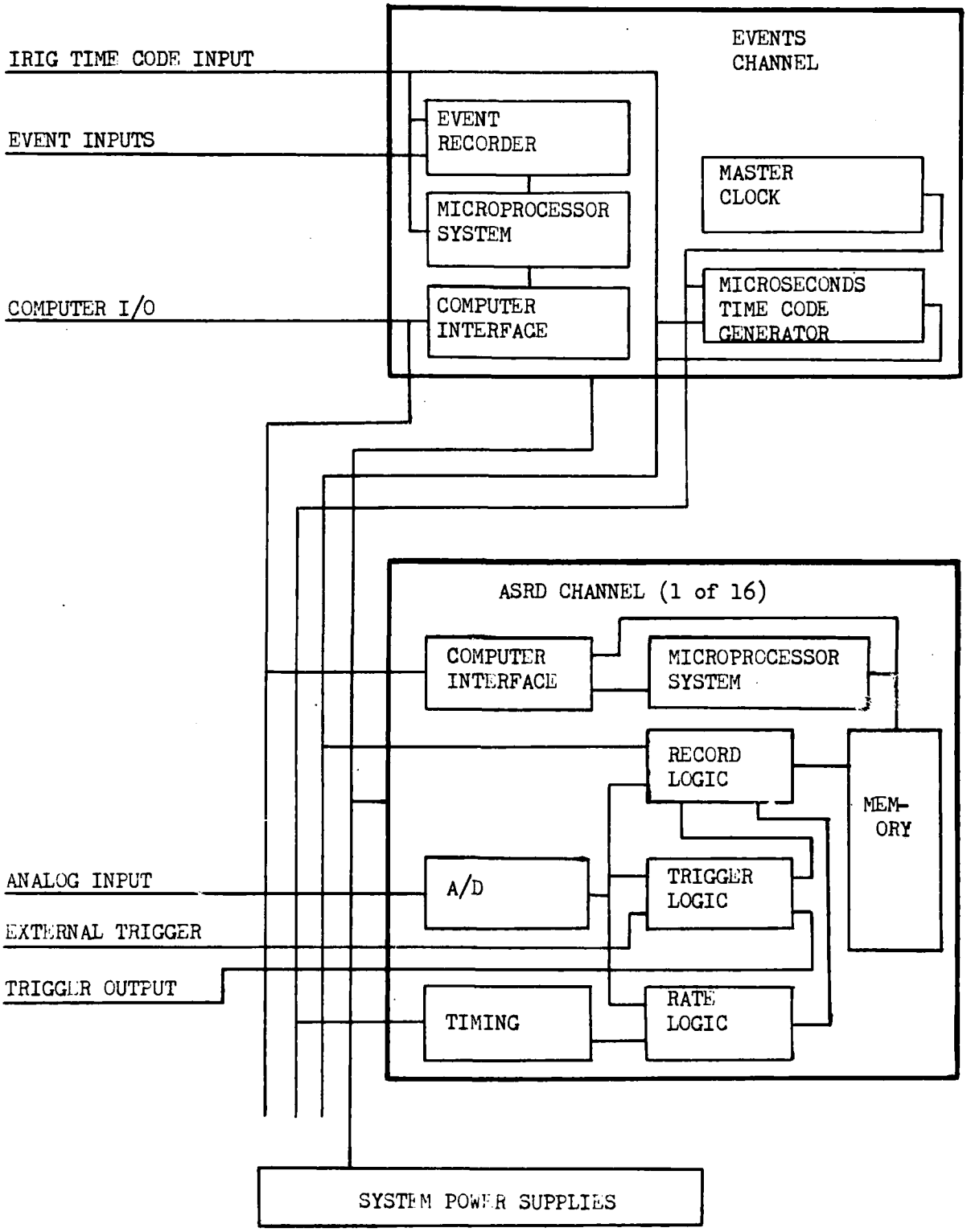


Figure 3-13. Block diagram of ASRD.

3.2.1.6 (Cont'd)

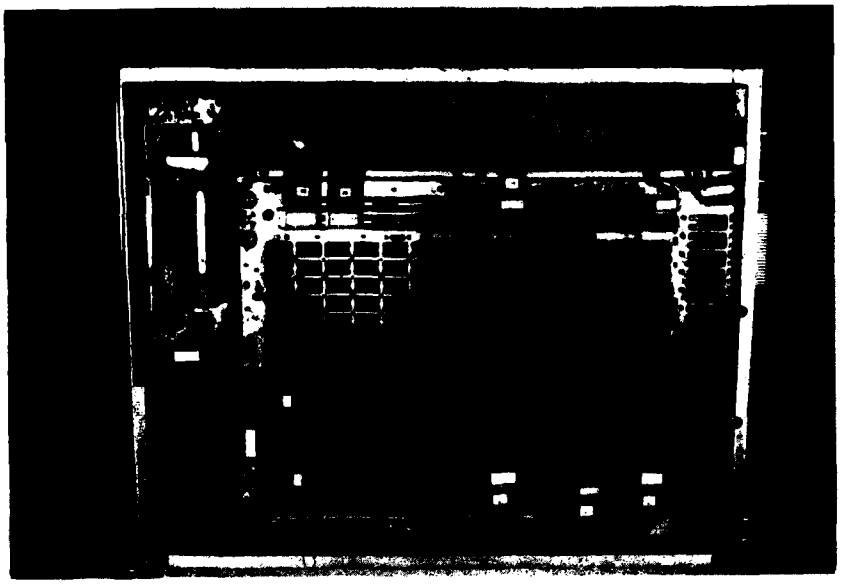


Figure 3-14. Picture of ASRD channel assembly.

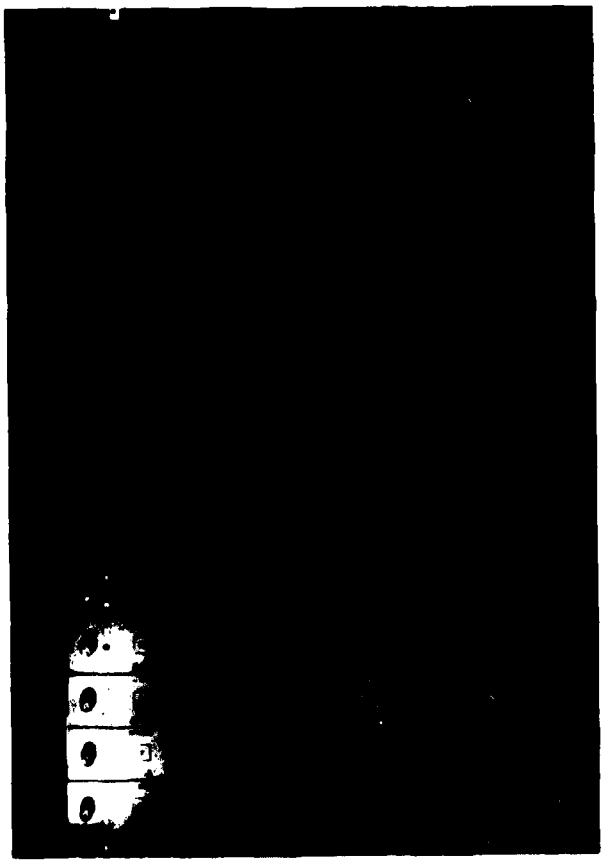


Figure 3-15. Picture of ASRD housing.

TABLE 3-4. ASRD CHARACTERISTICS

Number of channels per housing: 16

Number of channels per controller: 32

Programmability: 16 bit TTL interface

Memory size: 16 k to 128 k words (16 bits per word) in 16 k increments

Trigger modes: Internal - relative, absolute, normal
External
Master/slave
Run delay
Stop delay
Pretrigger memory

Rates: 97.7 Hz to 800 kHz in binary sequence (14 total)

Modes: Fixed rate or adaptive sample rate

Time resolution: 1 microsecond

3.2.1.7 Time code generator/translator. The time code generator/translator (TCG/T) provides day, hour, minute, second, and millisecond time information to the digitizer. This time information, when augmented with digitizer internal timing information, allows the time sequence of independent channels to be established. When placed in translator mode and fed with the central range time from a receiver, the time in all BTSTs can be synchronized. While the digitizer will function without the time code generator, the time information supplied by the TCG/T will be missing and the internally generated time information will be incorrect. The TCG/T is available commercially as Datum 9300-7049. Characteristics of the TCG/T are in Table 3-5.

TABLE 3-5 TIME CODE GENERATOR/TRANSLATOR CHARACTERISTICS

Translator input format: IRIG B

Internal generator: Crystal controlled
1 MHz
±5 parts per million stability

Parallel output: BCD format
44 bits - day, hour, minute, second, millisecond
TTL compatible

Display: 12 LED digits

Size: 4.4 cm H by 43.2 cm W by 45.7 cm D (1.75 by 17.0 by 8.0 in.)

Temperature: 0° to 50° C operating

Power: 120 VAC, 60 Hz, 35 W

3.2.2 Built-in Test Equipment

The capability to view input signals and to perform self-test, checkout, and calibration functions is provided by the BITE. The components of the BITE are not used directly in the data acquisition process; however, they provide essential support. The BITE available in each BTST consists of a scanner, oscilloscope, digital voltmeter, calibration capacitor, and signal generator. A counter/timer which is shared among all BTSTs is also required. A picture of the installed BITE is in Figure 3-16. Figure 3-17 shows how the BITE is interconnected. Manuals for each item of BITE are located in the BTST for reference. Detailed specifications for the BITE are in Appendix E.

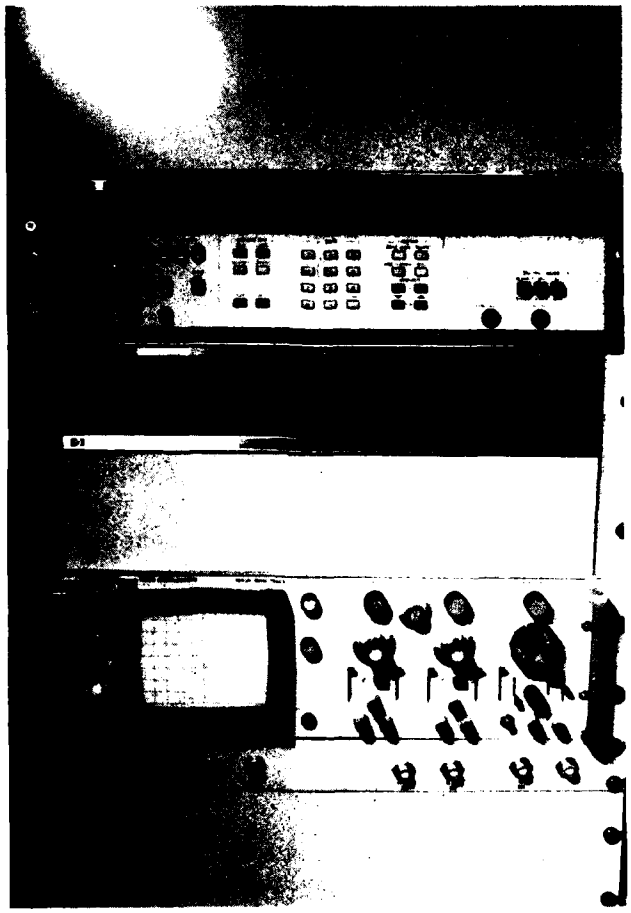


Figure 3-16. BITE.

3.2.2 (Cont'd)

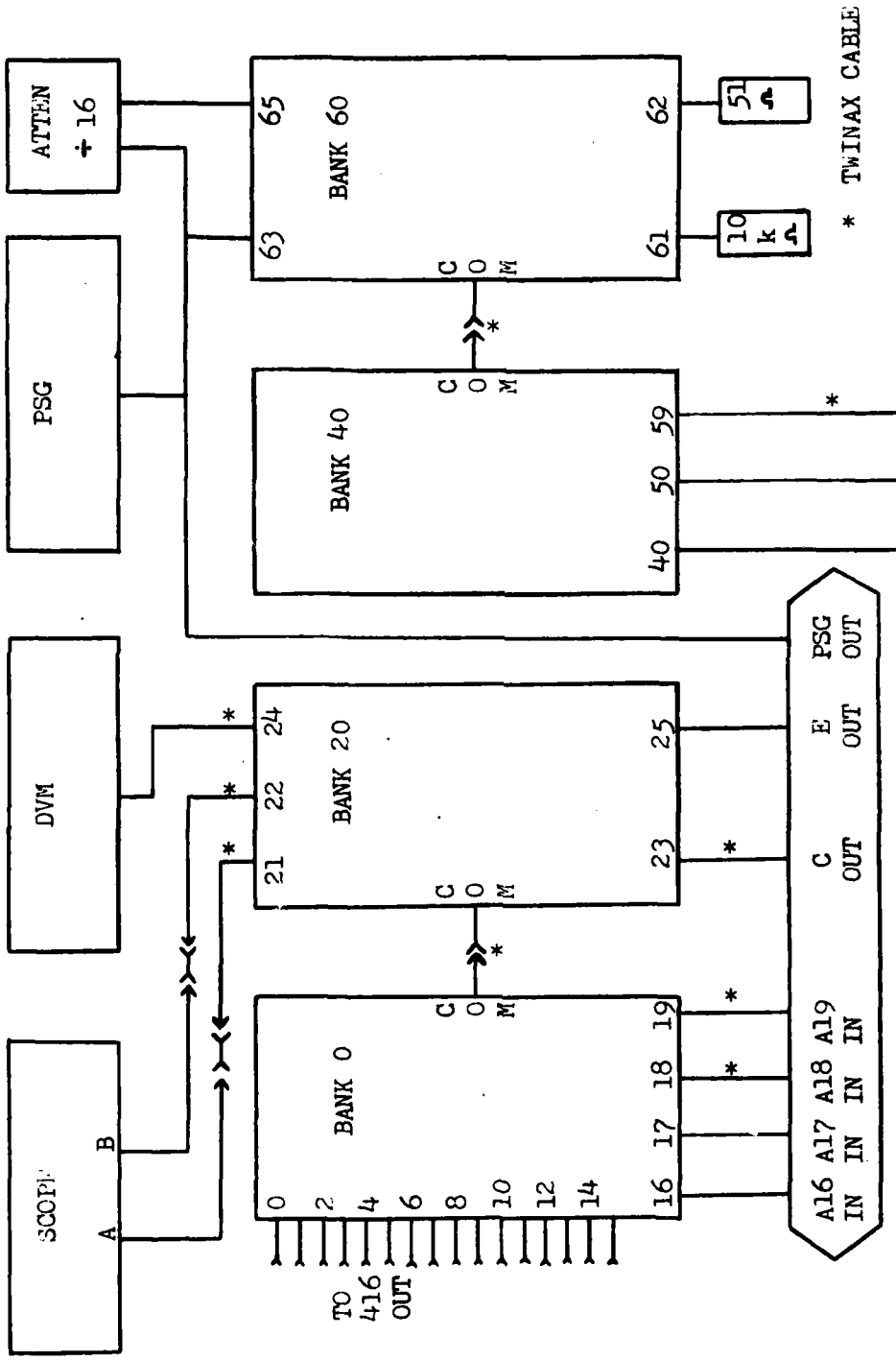


Figure 3-17. BITE interconnections.

3.2.2.1 Scanner. The scanner is a programmable relay actuator which provides a switchable interface between the data acquisition equipment and the BITE. The scanner has four banks each having 20 positions. The banks are independently programmable. A diagram of the scanner bank configuration is in Figure 3-18.

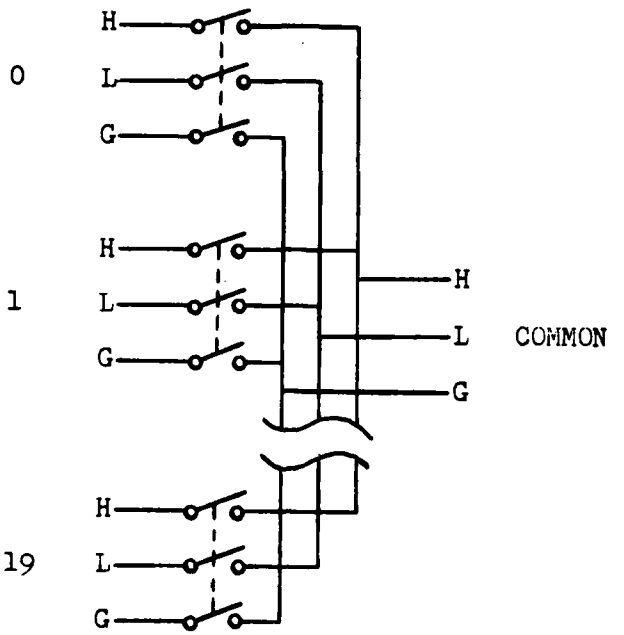


Figure 3-18. Scanner bank configuration.

The scanner does not have a front panel and can be programmed only from the computer. Important characteristics of the scanner are:

Number of independent banks: 4.

Number of positions per bank: 20.

Contacts per position: 3 (high, low, and shield).

Programmability: GPIB only.

Model: Hewlett-Packard 3495A.

3.2.2.2 Oscilloscope. The oscilloscope allows channels to be viewed or compared during test setup or checkout to determine that noise levels and ambient values are correct. The oscilloscope is the only piece of BITE which is not programmable. Important characteristics of the oscilloscope are:

Type of input: Differential.

Number of channels: 2.

Bandwidth: 500 kHz.

Sensitivity: 0.1 mV.

Model: Hewlett-Packard 1200B.

3.2.2.3 Digital voltmeter. The digital voltmeter (DVM) provides the capability to make high accuracy alternating current (AC), direct current (DC), and resistance measurements. This capability is primarily used for calibration of the data acquisition equipment. The DVM is periodically removed from the BTST to allow for laboratory certification of traceability to National Bureau of Standards. Important characteristics of the DVM are:

Function: AC V, DC V, and resistance.

Programmability: Front panel or GPIB.

Display: 5-1/2 digits.

Ranges: DC - 0.1 V, 1 V, 10 V, 100 V, 1000 V.
AC - 1 V, 10 V, 100 V, 1000 V.
Ohms - 0.1 kΩ , 1 kΩ , 10 kΩ , 100 kΩ , 1000 kΩ , 10,000 kΩ .

AC measurement: Root-mean-square (rms).

Model: Hewlett-Packard 3455A.

3.2.2.4 Programmable signal generator. The programmable signal generator (PSG) serves as a signal source for testing and calibrating the data acquisition equipment. A wide variety of test conditions can be simulated by the generator. Important characteristics of the PSG are:

Programmability: Front panel or GPIB.

Waveforms: Sine, square, or triangle.

Modes: Continuous or counted burst.

Frequency range: 0.001 Hz to 50 MHz.

Amplitude range: 10 mV to 10 V.

Offset range: ±10 mV to ±10 V.

Model: Hewlett-Packard 8165A.

3.2.2.5 Programmable counter/timer. A single programmable counter/timer (PCT) is rotated among the BTSTs to calibrate the ASRD time base. The PCT has laboratory certification of traceability to National Bureau of Standards. Important characteristics of the PCT are:

Functions: Time interval and frequency.

Time base: 10 MHz.

Display: 9 digits.

Programmability: Front panel or GPIB.

Programmable parameters: Function, range, trigger slope, coupling, and threshold.

Model: Racal-Dana 9514.

3.2.2.6 Calibration capacitor. Calibration of the charge amplifier in the signal conditioner requires the use of a calibrated capacitor. The input of the capacitor is fed by the PSG and the output goes to the charge amplifier. The capacitor is periodically removed from the BTST to allow for laboratory certification of traceability to National Bureau of Standards. Important characteristics of the calibration capacitor are:

Capacitance: 0.01 microfarad.

Temperature coefficient: -0.014% per °C.

Model: Kistler 5371A10000.

3.2.3 Computer Equipment

The capability to control the BTST and to store, process, print, plot, or record the data acquired by the system is provided by the computer equipment. The computer equipment is all commercially available from Hewlett-Packard. At each BTST there is a set of manuals describing the computer hardware. These manuals should be referenced for specific information on individual items of equipment.

3.2.3.1 Minicomputer. The general purpose minicomputer exercises control over the other computer equipment, the data acquisition equipment, and the BITE. Control is exercised through interface cards which plug into the input/output (I/O) section of the machine. Instructions for operating the computer and

locations of the I/O and memory cards are in Appendix F. Guidelines for use of the GPIB interface are in Appendix G. General characteristics of the computer are:

Word size: 16 bits.

Memory capacity: 512 k words.

Memory size used: 128 k words.

I/O card slots: 12.

I/O cards used: Time base generator.
Disc interface.
Terminal interface.
Magnetic tape interface (2 cards).
GPIB interface.
16 bit TTL I/O interface.
High speed serial interface.

Direct Memory Access (DMA) capability: 512 k words per second.
Two channels.
16 bit word size.
8192 word block size.

Interrupt capability: Multilevel vectored.

Memory error detection: Single parity bit.

Front panel display/switch register: 16 bit I/O.

Instruction times: ADD (direct) - 1 microsecond.
ADD (indirect) - 1.5 microseconds.

Size: 31.1 cm H by 48.3 cm W by 63.5 cm D
(12.25 by 19 by 25 in.).

Temperature: 5° to 37° C operating; -28° to 50° C storage.

Power: 120 VAC, 60 Hz, 900 W.

3.2.3.2 Disc. The storage of programs and data is accomplished by the disc memory subsystem. There are several different capacity discs used in the BTSTs; however, they all use the same controller so their operation is the same in all systems. General characteristics of the disc system are:

7906 capacity: 9.8 million words.

7920 capacity: 25 million words.

7925 capacity: 60 million words.

Access time: Track to track - 5 ms.
Average random - 25 ms.

Transfer rate: 468.8 k words per second.

Tilt: Up to ±20° about either axis.

Controller: Multiple access.

Temperature: 10° to 40° C operating; -40° to +65° C storage.

Power: 120 VAC, 60 Hz.

3.2.3.3 Terminal. The entry and display of information, as well as the plotting of graphics, is accomplished by the terminal. The terminal consists of a keyboard unit and a display module. The keyboard unit houses a standard typewriter-like set of keys plus a number of special function keys. The display module holds the cathode ray tube (CRT) display, dual cassette tape units, terminal interface, and all other electronics. The terminal serves as the I/O device to allow the operator to exercise control over all of the other equipment. General characteristics of the terminal are:

Display size: 13.3 cm H by 26.0 cm W (5.25 by 10.25 in.).

Graphics capability: 360 x 720 pixels.
Internal graphics generator.
Graphics cursor.

Graphics functions: Cursor on/off.
Pixel on/off.
Position cursor.
Line type.
Position origin.
Graphics text size and direction.

Alphanumeric capability: 80 characters by 24 lines.
128 ASC II character set.
9 x 15 character cell.
Full descenders and underline.

Alphanumeric functions: Video normal/inverse.
Cursor up/down/right/left.
Display roll up/down.
Clear memory.
Go to start of memory.
Display lock.
Margin set left/right.
Cursor position.

Keyboard: 56 alphanumeric keys (QWERTY).
11 graphic control keys.
12 display control keys.
8 special function keys.
18 special control keys.
Band rate, parity, and duplex switches.

Video output: 875 line composite video.

Size: Display - 34.5 cm H by 44.1 cm W by 46 cm D (13.6 by 17.4 by 18.1 in.)
Keyboard - 9.2 cm H by 44.4 cm W by 23 cm D (3.6 by 17.5 by 9.1 in.).

Power: 120 VAC, 60 Hz, 150 W.

Model: Hewlett-Packard 2648A with option 007 and 13254A video interface.

3.2.3.4 Printer/plotter. The printer/plotter is used to generate a printed copy of alphanumeric or graphic information. The printer/plotter has better resolution than the terminal, but it is much slower. It is not possible to copy directly from the terminal to the printer/plotter. General characteristics of the printer/plotter are:

Type: Thermal.

Plot size: 18.8 cm W by 500 cm L.

Text size: 88 characters x 90 lines.

Character set: 128 character ASC II.

Print speed: 30 characters per second.

Plot speed: 25.6 cm/sec each axis.

Plot accuracy: $\pm$0.2% of deflection $\pm$0.35 mm.

Size: 21.0 cm H by 44.2 cm W by 48.3 cm D (8.3 by 17.4 by 19.0 in.).

Temperature: 0° to 55° C operating.

Power: 120 VAC, 60 Hz, 300 W.

Model: Hewlett-Packard 7245A.

3.2.3.5 Magnetic tape drive. The magnetic tape drive provides the capability to back up data or programs stored on disc. The tape drive serves as the primary means of transferring data from the BTST to a processing facility. Updating of programs and documentation is also accomplished using the tape drive. General characteristics of the magnetic tape drive are:

Tracks: 9.

Density: 1600 bytes per inch (BPI).

Recording mode: Phase encoding.

Write speed: 45 ips.

Fast forward/rewind speed: 160 ips.

Size: 60.2 cm H by 48.3 cm W by 40.3 cm D (23.7 by 19 by 16 in.).

Temperature: 0° to 55° C operating.

Power: 120 VAC, 60 Hz, 400 W.

Model: Hewlett-Packard 7970E-151.

3.3 Software Configuration

There are four categories of software which the operator must be concerned with:

a. Computer operating system.

b. Operational software.

c. Analytical software.

d. Utility software.

Each is required to make a BTST function.

3.3.1 Computer Operating System

While the computer equipment described in the preceding paragraph contains all of the hardware elements necessary to operate the BTST, the computer operating system provides the programs required to make the pieces of equipment operate together as a coherent system. Although not strictly a part of the operating system, all of the programs supplied by the computer manufacturer, which are required to make a BTST function effectively, will be described in this paragraph.

3.3.1.1 Real time executive. The computer supervisory program, which normally controls all other computer programs, is the operating system. In the BTST the operating system is called a real time executive (RTE). Depending upon the age of the BTST, either an RTE-IVA, RTE-IVB, or RTE-6/VM operating system is used. Although there are some differences between the three systems, as far as BTST operation is concerned, they are equivalent. RTE has the following general capabilities:

a. Receives and processes operator commands or other program's requests (CALL EXEC).

b. Supervises equipment resources.

c. Supervises computer memory utilization.

d. Schedules program execution based on priority.

e. Supervises the computer interrupt system.

The computer can only execute one program at a time. However, the operating system will allow several programs to reside in memory at the same time with the program having the highest priority actually executing. If the executing program is forced to wait for a resource, then execution of the next highest priority program is begun or resumed. When the higher priority program is ready to continue, the lower priority program is halted and its results saved temporarily. In this manner, the computer appears to be handling several jobs at once. This technique is called multitasking.

Multitasking is useful for most of the operations required on a BTST. But, because there is overhead in keeping track of all the resources involved, any time-critical task cannot be carried out through the operating system. For this reason, data acquisition and critical control tasks are carried out by programs which turn off the interrupt system (and thereby the operating system). When the operating system is turned off, any computer equipment not specifically controlled by the program is inoperative.

To ease the task of addressing equipment attached to the computer, a hierarchy of addressing methods is made available by the operating system. At the lowest level is the I/O slot select code (SC), followed by the equipment number (EQT) with subchannel, and at the highest level the logical unit (LU). A description of each of these addressing methods is contained in Appendix H.

An extensive knowledge of the operating system is not necessary for operation of the BTST. Normally the operating system functions without any operator intervention at all. A summary of useful RTE operator commands is in Appendix H for those instances where operator input is required.

3.3.1.2 File manager. All of the source and relocatable code for programs, as well as documentation, are stored on discs in an organized collection of files. Each file is given a name to allow it to be accessed easily. The file manager is a supervisory program (called FMGR) which allows an operator to create, catalog, manipulate, and purge files in addition to some useful utility operations. A summary of useful FMGR commands, file naming specifics, and disc cartridge layouts are in Appendix H.

3.3.1.3 Text editor. The capability to enter or modify text in a file is provided by a program named EDITR. This program is used for FORTRAN and assembly language source code files and for documentation files. It is basically a line oriented editor which has a few global editting features. A list of EDITR commands is provided in Appendix H.

3.3.1.4 FORTRAN compiler. The majority of programs designed to operate the BTST or process its data is written in FORTRAN. To convert the FORTRAN program source code into a form which the computer can use, a FORTRAN compiler is required. The program name is FTN4 or FTN7X, depending on the age of the system. All programs are written in FTN4, which is compatible with FTN7X (the reverse is not true). The input for the compiler is a source file created by EDITR. The output from the compiler is a relocatable file stored on disc.

3.3.1.5 Assembler. Interface card driver routines and some special software are written in Hewlett-Packard assembly language. To convert the program source code into a form which the computer can use, an assembler is required. The program is called ASMB or MACRO, depending on the age of the system. The

input to the assembler is a source file created by EDITR. The output from the assembler is a relocatable file stored on disc.

3.3.1.6 Linker/loader. After all of the relocatable modules which make up a program have been created by FTN4 or ASMB, it is necessary to link them together, add system library modules, place the program code on disc, and create a program name segment for the operating system to recognize the existence of the program. This task is accomplished by program LOADR. If a program is loaded with the permanent load option, then it is not necessary to load it again unless a change in the program is made.

A discussion of the steps involved in loading a program and the steps involved in program development is in Appendix H.

3.3.1.7 System status. Occasionally it may be necessary to determine what the system is doing (or not doing). A system utility named WHZAT allows the operator to find out what programs are currently scheduled and what their current status is. If a piece of equipment is not ready, then this information will be available also. Through an option, the current memory configuration and status can be determined. Examples of the output provided by WHZAT are in Appendix H.

3.3.1.8 System disc track status. An area on disc is set aside for use by the operating system. This area is not supervised or used by the file manager program; only the operating system can write in this area of disc. The area is used for the following purposes:

a. The operating system program and tables.

b. System library routines.

c. Swap area for programs suspended and removed from memory.

d. Storage for programs created by LOADR.

e. Temporary storage for EDITR.

Program LGTAT allows an operator to display the contents of the system disc. This is necessary when the system area fills up, in order to make room for program loading. An example of the output provided by program LGTAT is in Appendix H.

3.3.1.9 WELCOM file. When the computer operating system is first read from disc, a file manager transfer file called WELCOM is automatically executed. This file allows the computer to be initialized to any condition desired by the system manager. This capability is used on the BTST. The functions performed by the BTST WELCOM file are:

a. Display the BTST trailer number on the terminal.

b. Display the system generation file name and date.

c. Schedule program PING.

d. Set the system list device to the terminal.

The WELCOM file resides on disc cartridge 11 on RTE-IVA systems and on disc cartridge 2 on RTE-IVB and RTE-6/VM systems. The operator must make certain that copies are not interchanged between trailers when software updates are performed. An example WELCOM file is in Appendix H.

3.3.1.10 System backup. Although the disc systems are highly reliable, the possibility that a disc can fail, destroying its information, does exist. To prevent the complete loss of all information, backup of the operating system and file manager disc areas is required. Backup of ballistic data is handled separately in the next section.

The system area is normally backed up separately from the file manager area. A new system backup needs to be performed only when a change in the operating system is made. System changes are not made very frequently, so this process will not be required often. Details on performing a system backup or restore are in Appendix H.

The file manager area needs to be updated periodically or after any major change in the file manager disc contents. Maintaining current file manager backups is essential to proper operation of the BTST. If an outdated backup is used to restore a disc, old program source or relocatable code may be introduced, giving unpredictable or undesirable results. Details on performing a file manager backup or restore are in Appendix H.

3.3.2 BTST Operational Software

Just as the operating system is required to make the computer function effectively, the BTST operational software is required to make the data acquisition hardware function effectively. The operational software is divided into three components:

a. Data acquisition. Setup and control of equipment, documentation of test information, and storage and retrieval of data files.

b. Calibration. Calibration of the data acquisition equipment.

c. Test. Verification and fault isolation of the data acquisition equipment.

Efficient operation of a BTST requires that an operator be able to perform all three tasks. While most of the software is designed to be simple to use, the operator must still have an overall understanding of the hardware-software relationships involved.

In order to make the software source code easy to understand and to make it modifiable and supportable in the long term, a set of programming standards have been applied to all software generated for the BTST. A copy of the software standards is in Appendix I.

3.3.2.1 Data acquisition programs

3.3.2.1.1 Program ADCHK. The primary data acquisition program for the BTST is ADCHK. ADCHK is a comprehensive supervisory program which allows the operator to control the data acquisition process. Operator instructions are entered as two-letter commands with additional qualifying parameters if needed. ADCHK checks all command inputs as closely as possible to determine that they are acceptable. Once an acceptable command is obtained, ADCHK either handles the command processing itself or schedules another program to process the command. The program is designed to handle a 32-channel system, but it will work effectively with 8- or 16-channel systems.

ADCHK is essentially self-contained in that once it is running it does not require use of either the EDITR or FMGR programs. To allow this autonomy, ADCHK makes use of up to three disc LUs to store parameters, documentation, and data (LUs 18, 19, and 20). These LUs should not be used for file storage by FMGR (they should be dismounted at all times). On these three data LUs, ADCHK maintains supervisory control. The layout of the data discs for use by ADCHK is given in Appendix J.

Besides the data disc areas, ADCHK also makes use of two files stored on file manager LUs. One of the files contains all of the calibration information required by ADCHK. This file is named @CALZZ and is the first file on LU14. The other file contains the ADCHK menu. This file is named @MENU and is the first file on LU10. A description of the contents of these two files is in Appendix J.

The ADCHK commands are divided into five categories:

a. Setup commands. Initialization or preparation of parameters and documentation information prior to data acquisition.

b. Equipment control commands. Operation of the data acquisition equipment and BITE.

c. Examine data file commands. Read and display data or parameters from a data file on disc.

d. Modify or annotate data file commands. Correct or update documentation information in a data file.

e. Utility commands. Housekeeping functions.

A complete description of the ADCHK commands and the support subroutines and programs utilized by ADCHK is in Appendix K.

After data has been acquired by the digitizer, it is transferred through the computer to disc. A special format was developed to allow all of the documentation information and data to be placed in one record. The format for the ADCHK data record is in Appendix L.

3.3.2.1.2 Calibration programs. In order to ensure that accurate measurements are obtained by the BTST, it is necessary to calibrate the equipment. Calibration of each item in the data acquisition chain is carried out, using the BITE. The philosophy used is that a calibration factor is determined for each mode or range on each piece of equipment. Thus, for each channel of the signal conditioner there are six factors, for the filter/amplifier there are 15, for the interim offset/amplifier there is one, and for the digitizer there is one. These values are stored in a disc file by the calibration programs. When data are acquired, the appropriate calibration factors and equipment serial numbers are read from the calibration file and saved in the data record.

The equipment should be calibrated every three months or whenever a change is made in an item. Therefore, calibration is not done on a round-by-round or a daily basis. Since stable, solid state components are involved in the system, there should not be any observable difference from day-to-day as is experienced when analog tape is used (unless an equipment failure occurs).

In the process of performing a calibration, many features of the equipment are used. However, the calibration programs are not a substitute for the verification programs described in the next paragraph. The calibration programs assume that the equipment being calibrated is working properly. The only criterion the calibration programs check against is if the calibration factor is within $\pm 4\%$ of the nominal value.

For a particular piece of equipment, each channel is calibrated individually. The normal input and output cables are removed from the channel and the calibration cables are attached. The program then performs the calibration, using the BITE. After checking the results, the new values are written to the calibration file.

The calibration programs are:

CA316 - Performs six level calibrations on the signal conditioner: voltage, charge, strain X1, strain X10, strain X100, strain X1000. Three input cables (one with the calibration capacitor) and an output cable are required.

CA416 - Performs 15 voltage calibrations on the filter/amplifier for each gain: 1, 2, 4, 5, 10, 20, 25, 40, 50, 100, 200, 250, 400, 500, 1000. One input cable and an output cable are required.

CADIG - Performs a single voltage calibration on the digitizer. One input cable is required.

CATIM - Performs a timing check on the digitizer. One input cable is required.

CALX - Allows calibration values to be read, modified, and swapped. This capability is useful when maintenance problems are encountered and it is necessary to move or replace cards in an item.

The format of the calibration factor storage file is in Appendix J. The calibration program operating instructions are in Appendix M.

3.3.2.1.3 Test programs. The equipment in the data acquisition chain is extremely complex. There are many variables to be set for each channel. Although the equipment is reliable, occasional failures are to be expected. Finding faults in ballistic firing data resulting from equipment malfunction is counterproductive. Therefore, a group of test or verification programs have been developed to aid the operator in assessing the performance level of the BTST.

The test programs should be run on any new piece of equipment, any item suspected of not functioning properly, and at periodic intervals to certify system performance. The programs are not guaranteed to find every conceivable failure mode. However, they exercise the vast majority of capabilities of each device. Unfortunately, the programs do not tell how to repair a malfunction or why a malfunction occurs. Instead they only verify whether the equipment works up to the expectations of the program.

Due to the complexity of the equipment involved, it is not always possible for the programs to perform the tests completely automatically. When operator intervention is required, the programs prompt the operator to perform a task and wait until it is completed. As a minimum, input and output cables may require attention on each channel.

The verification programs are:

CK316 - Checks the signal conditioner in voltage, charge, or strain mode. Measures gains, DC offsets, noise values, and calibrator outputs.

CK416 - Measures the filter/amplifier gains, bandwidths, DC offsets, and noise values. Checks the passband ripple and overload detectors.

CHKOU - Checks the digitizer for variable sample rate operation. Checks time words, change words, rates used, decrease delay, trigger threshold and slope, stop delay, and amplitude level. Error conditions are listed and all data are saved on disc in an ADCHK compatible file for additional analysis. The test can be performed at 80 kHz, 40 kHz, or 5 kHz.

TRGRT - Checks the digitizer for trigger threshold and slope, absolute value and relative trigger operation. A total of 12 tests can be performed to check all possible combinations of internal trigger logic. Error conditions are listed and all data are saved on disc in an ADCHK compatible format for additional analysis.

RMTST - Initiates the digitizer internal self-test routines for register and memory verification. Error conditions are listed.

MSTST - Checks the digitizer master-slave triggering relationship. Error conditions are listed with a limited trigger time printout optionally available.

PRETR - Checks the digitizer pretrigger memory operation. Error conditions are listed with a limited data printout optionally available.

RSDLY - Checks the digitizer run-stop delay operation. Error conditions are listed and all data are saved on disc in an ADCHK compatible format for additional analysis.

Instructions for operating the verification programs are in Appendix N.

3.3.2.1.4 Backing up and restoring data. Since the possibility of disc failure exists, a copy of the data on disc should be made to provide a backup. A backup copy should be made at the end of each day, at the end of a test, before the trailer is moved, or if the disc is full. There are two ways in which the operator can make a backup. If a copy of all the records in a data directory is to be made, then the BU command of ADCHK can be used. This is the normal way to perform the backup. However, if only a portion of the records are desired, then program DISCS should be used.

To restore data from tape to disc which was saved using the BU command, the RE command of ADCHK can be used. If the data was saved by program DISCS, then program TAPER must be used to restore the data.

Instructions for programs DISCS and TAPER are in Appendix O.

3.3.3 Analytical Software

In the old ballistic trailers after a round was fired, the main analysis technique available was to examine an oscillograph trace. With the presence of a computer in the BTST more powerful analytical capability is available. The lengths to which one can go are basically limited by the amount of time available. However, some important indicators of data quality can be made very quickly using the programs described below. While the programs provide a great deal of valuable information, the operator is ultimately responsible for determining their adequacy for the particular test. Operating instructions for the analytical programs are in Appendix O.

3.3.3.1 Program QPEAK. For many ballistic tests the parameter of prime importance is the peak of the recorded waveform. If the peak is within expected limits, then the channel is probably functioning properly. Program QPEAK quickly provides the operator with peak information for each channel. In particular, the following information is supplied:

Round number.

Time and date (if data date and header date do not agree, a warning message is given).

Header remarks.

Channel word count.

Channel gain.

Channel ambient count.

Channel maximum digitizer count.

Channel minimum digitizer count.

Channel positive peak (maximum - ambient) in engineering units.

Channel negative peak (minimum - ambient) in engineering units.

Channel units.

By scanning the QPEAK output, the operator can determine that the major header entries are correct, that the correct amount of data has been taken, what the channel gain setting is, that there is no unexpected offset or drift in the baseline (ambient), that the digitizer has not clipped nor is lacking resolution (positive and negative counts), and that the signal is within the expected range (positive and negative peak with units). Round-to-round changes in signal level can be determined quickly by comparing the QPEAK output for each round.

3.3.3.2 Plotting programs. Numerical indicators such as those provided by QPEAK are useful in quickly assessing data quality. However, a graphical presentation of the data is often the most valuable overall indicator of data quality. Noise, improper equipment setup, transducer or cable problems, and equipment malfunction can usually be identified graphically at levels where the numerical indicators will not show any problem. Two plotting programs are available to provide the operator with a graphical representation of the data. One of the programs will plot on either the graphics terminal or the printer/plotter while the other program will be a plot only on the printer/plotter. The graphics terminal has medium resolution, digital presentation, and high-plotting speed, whereas the printer/plotter has high resolution, analog presentation, and low-plotting speed.

The plotting programs are QPLOT and DPLOT. Table 3-6 compares the features of the two programs. QPLOT is usually used to create plots on the graphics terminal during testing. Report quality plots can be generated after testing with DPLOT.

TABLE 3-6. PLOT PROGRAM CAPABILITIES

Terminal	QPLOT	DPLOT
Printer/plotter	Yes	Yes
Graphics terminal	Yes	No
Vertical scaling units	Transducer or volts	Transducer
Vertical or horizontal scaling mode	Automatic	Automatic or operator (time/eng units)
Horizontal scale divisions	5 tic marks at 2.35 inch intervals	9 tic marks at 1 inch intervals
Vertical scale divisions	3 tic marks at 3.32 inch intervals	6 tic marks at 1 inch intervals
Plot entire data record	Yes	Yes
Method	Maximum point in interval	Every point

TABLE 3-6 (CONT'D)

Terminal	QPLOT	DPLOT
Offset from start	Yes	Yes
Type	Number of points	Time
Ambient subtracted	Yes (scaled mode)	Yes
Horizontal marker	No	Yes
Incremental plots	Yes	No
Vertical scale determined by	Whole record or each plot	-
Modification of input information	No	Labels, legend, ambient, scale factor
Multiple plots per page	No	Yes
Point plot	Yes	Yes
Unattended operation	No	Yes
Use	Quick-look	Report

3.3.3.3 Program CDIFF. With some ballistic data and primarily differential chamber pressure, the information of interest is really the difference between two channels. Program CDIFF provides the capability to take the difference between two channels in an existing ADCHK data record. The program is restricted to differencing channels which were recorded at the same fixed rate and which were triggered at the same time. CDIFF multiplies each channel's data by its appropriate scale factor, takes the difference between the two channels, scales back to integer data format, and stores the new data in an ADCHK compatible data file on disc. Since the file is ADCHK compatible it can be plotted, analyzed, and saved using the same techniques applicable to any other data file. The difference data are placed in the data record as channel 31 and the round number in the header is modified to flag the difference record.

3.3.3.4 Program BLAST. Analysis of blast overpressure records involves measuring the peak pressure and two time intervals called the A-duration and B-duration. Program BLAST generates these values in the manner prescribed by MIL-STD-1474B. With this program available on the BTST, it is very easy for test personnel to determine if the blast measurements are being made properly.

3.3.3.5 Program SHIFT. Occasionally, it is necessary to make static measurements on the BTST. This is typically encountered with strain gage measurements of structural components of weapon systems. Permanent deformation of the structure can be detected by comparing the recorded level from round-to-round. Program SHIFT allows these comparisons to be made.

3.3.4 Utility Software

A number of programs which do not fall into any of the previous categories are also available on the BTST. These programs perform a variety of utility functions which make operation of the system easier. Many of these programs will not be used in daily operation; however, the operator needs to be familiar

with their existence. Operating instructions for each of the utility programs can be found in Appendix O. The programs can be divided into the following categories:

a. Gage factor calculations: GAGEF.

b. ADCHK support: LIRT, MKMNU, PRMCL, TBLGN.

c. Data record change or manipulation: BLKTR, CHANG, CHFRM, DRCHG, FILTR, FUDGE, ODLST.

d. Equipment control: DGCHK, GPIB, MCCHK, OVRLD, PG316, PG416.

e. Data record processing: CHECK, FNDTM, RMS.

f. Miscellaneous: DISCS, FILOC, GETIT, LUMAP, OREAD, PING, TAPER.

g. Temperature measurement: CTEMP, TEMP, THERM.

3.3.4.1 Gage factor calculations. Program GAGEF. This program allows gage calibration factors to be converted from a nonstandard format to the format required by ADCHK.

3.3.4.2 ADCHK support. Program LIRT. This program allows a copy of a rate table to be made on a list device.

Program MKMNU. This program formats the type 3 file manager file MENU into type 1 file @MENU so that ADCHK can display the command menu.

Program PRMCL. This program sets the unused portion of the parameter tracks to zero.

Program TBLGN. This program generates the standard rate tables and allows them to be written to disc.

3.3.4.3 Data record change or manipulation. Program BLKTR. This program copies a sequential block of ADCHK data records from a source disc to a destination disc.

Program CHANG. This program allows a word from an ADCHK record to be read and, if desired, changed.

Programs CHF, CHFRM, and ODLST. There are three different formats of ADCHK data records: version I, the original format; version II, the current 10-bit format; and version III, the current 12-bit format. Program CHFRM converts version I format to version II. Program CHF converts version II format to version III.

Program ODLST. This program allows the version I directory to be read (it cannot be read with the version II or III ADCHK DL command).

Program DRCHG. This program allows an ADCHK data directory to be truncated or expanded.

Program FILTR. This program copies a single ADCHK data record from a source disc to a destination disc.

Program FUDGE. This program allows gage factors, gage factor units, and/or plot labels to be changed on a single record or on an entire disc LU.

3.3.4.4 Equipment control. Program DGCHK. This program allows the ASRD to be controlled from a terminal. Program DGCHK is scheduled by the ADCHK CK command.

Program GPIB. This program allows a number of GPIB commands to be executed from a terminal.

Program MCCHK. This program checks the operation of a microcircuit register interface card.

Program OVRLD. This program reads and displays overloads on the filter/amplifier.

Program PG316. This program reads the ADCHK disc parameters and programs the signal conditioner accordingly.

Program PG416. This program reads the ADCHK disc parameters and programs the filter/amplifier accordingly.

3.3.4.5 Data record processing. Program CHECK. This program reads a data record and then counts for each channel the number of time words, change words, data words, data words at each rate, missing change words, multiple change words, and time words not in correct groups.

Program FNDTM. This program reads a data record and then finds for each channel the trigger time, previous sample time, and trigger location.

Program RMS. This program reads a data record and then calculates for each channel the AC or DC RMS value, average value, minimum count, maximum count, and time interval.

3.3.4.6 Miscellaneous. Program DISCS. This program copies data from a disc LU to a tape for backup.

Program FILOC. This program lists the disc LU, track and sector location of a file manager file.

Program GETIT. This program reads and displays requested areas of system memory.

Program LUMAP. This program reads the system tables and displays all defined I/O devices LU, equipment number, subchannel, select code, driver, DMA capacity, buffering, status, and label.

Program OREAD. This program allows the contents of any disc LU to be read in integer, octal, real, or ASCII format.

Program PING. This program controls the computer front panel LEDs to provide a constantly moving display.

Program TAPER. This program restores data from a backup tape to a disc LU.

3.3.4.7 Temperature measurement. Program CTEMP. This program displays a temperature reading from a single channel of a scanner thermocouple reference bank on the terminal.

Program TEMP. This program reads a disc record created by program THERM, converts the readings to temperature, and then writes the results to disc.

Program THERM. This program reads voltage values from a scanner thermocouple reference bank and stores the readings on disc.

3.3.5 Subroutines and Functions

To meet the requirements of the BTST software, a large number of subroutines and functions have been written. Some of these are unique to the BTST operational environment while others are of a more general utility. An alphabetical listing of subprograms utilized in the BTST software is in Appendix P.

3.4 BTST Operation

At this point all of the software and hardware resources of the BTST have been described. It is now possible to put all of this information, plus some additional instructions, together to describe how to operate a BTST. As the operating procedures are presented, it will be necessary to refer to other sections of this document for detailed instructions. When the operator becomes familiar with the steps involved in carrying out the various procedures, reliance on the documentation will diminish. However, the operator must perform all of the steps involved if the integrity of the data acquisition process is to be maintained. The steps involved are:

a. Unpack and set up the trailer following the instructions in Appendix Q.

b. Turn on the power to the digitizer, signal conditioner, filter/amplifier, offset/amplifier, scanner, DVM, PSG, printer/plotter, and any other GPIB equipment attached. The equipment must be allowed to warm up and stabilize before data acquisition or calibration. A minimum of 30 minutes, and typically an hour, must be allowed. If large internal temperature changes are encountered, a longer period may be required.

c. When the digitizer has completed running its internal test routines (approximately 1 minute per 16 k of memory is required), the operator should visually inspect each channel's front panel data display. In bits 0 to 4 should be the address for the channel - from 0 to 15 or 16 to 31 from top to bottom in sequence. In bits 10 to 13 should be the memory size of the channel in 16 k increments. After the visual inspection, the ST command of ADCHK should be exercised to obtain an error status report. Any problems found must be corrected before continuing.

3.4 (Cont'd)

d. Use program CALX (app M) to obtain a listing of all channels' and devices' serial numbers and calibration dates. Check that all calibration dates are current (with 3 months). Check that the serial numbers on the listing match the serial numbers of the cards actually installed in the equipment. If any discrepancies are found, they should be corrected by performing calibrations or swapping as appropriate. Even when only a portion of the channels are required for a particular test, it is important that all the channels be checked to provide a known complement of backup channels.

e. If any changes were made to any channel, a new listing of serial numbers and calibration dates should be made.

f. The correct listing should be dated and placed in the trailer binder.

g. A copy of the calibration file should be made (if any changes were made or if a copy does not exist) on a cassette tape. The copy can be made from FMGR by typing:

ST,@CALZZ,4,BN.

If it is necessary later to restore the calibration file from the cassette tape, this can be done from FMGR by typing:

DU,4,@CALZZ::14,BN.

h. The operator should check that a current system backup and FMGR backup are stored in the tape cabinet. If not, the instructions in Appendix H should be followed.

i. The operator should check that the data stored on data LUs 18, 19, and 20 have been backed up or are no longer required. If backup is required, the BU command of ADCHK (or alternatively, program DISCS) should be used to make a tape copy of the data.

j. Unless the information on LU 19 is required for the test, the PU command of ADCHK should be used to clear the disc.

k. With a knowledge of the transducers to be used and the desired results, the operator should enter all of the parameters for each channel using the PA command of ADCHK. The entries should be checked for accuracy using the PD command. If a number of channels are involved, the operator may wish to make a copy of the parameters using the SP subcommand of the PM command of ADCHK.

Alternatively, if parameters have been saved from a previous test, they may be restored using the GP subcommand of the PM command of ADCHK.

3.4 (Cont'd)

Parameter files may be transferred from one trailer to another by cassette tape using the

:ST,*name,4,BN

to save the file on tape and

:ST,4,*name:::1,BN

to recover the file from tape.

l. Using an appropriate stimulus at the transducer, the proper operation of the system should be checked. The CH command of ADCHK should be used to allow the stimulus signal to be observed on the oscilloscope. If ground loop problems are observed, follow the procedure outlined in Appendix R to try to eliminate the problem. The LO and AR commands of ADCHK should be exercised to verify proper triggering of the channel. When all testing is completed, the scanner should be cleared using the F subcommand of the CH command of ADCHK.

m. The header information should be entered using the HE command of ADCHK.

n. The operator should check that the time code generator/translator is operating and is set correctly.

o. When ready for the record, the LO and AR commands of ADCHK should be exercised. The operator should watch for any warning messages reported during the load operation. Invalid parameters, nonresponding channels, and overloads are trapped by the software. Any problems must be corrected before continuing. The trigger LEDs on the digitizers should be visually monitored. If a channel triggers prior to the event of interest, the test director should be alerted immediately and the problem corrected.

p. After the event of interest has occurred, the TR command of ADCHK should be used to transfer the data from the digitizers to the disc. The operator should watch for any warning messages reported during the transfer (overloads or scale factor problems). Any problems must be corrected before continuing.

q. At this point, the operator must make use of the appropriate analytical programs described in paragraph 3.3.3. The choice of programs will depend upon the type of data, the amount of time between rounds, and previous experience. The first round requires a lot of analysis to make certain that all channels are functioning properly. Later rounds may not require this amount of scrutiny.

r. Notes should be added to the record header using the NO command of ADCHK and comments should be added to the channel documentation using the CO command of ADCHK if appropriate.

s. Any errors or omissions found in the header or labels should be corrected using the MH or ML command, respectively.

t. The header should be updated for the next round to be fired using the HE command of ADCHK.

u. Return to step o for the next round.

v. If the disc is filled, it is the end of the day, or the test is completed, then the data on the disc must be backed up using the BU command of ADCHK.

w. Equipment which is not fully functional should be tagged and removed from the BTST for repair. The following information should be placed on the tag (gummed labels should not be used):

(1) Model and serial numbers.

(2) Trailer number.

(3) Channel number.

(4) Symptoms.

(5) Person reporting problem.

(6) Date.

The same information should be entered in the trailer log.

x. Pack up the trailer following the instructions in Appendix Q.

3.5 Software Generation and Modification

A description of available programs and subprograms is provided in Appendices M through P. These descriptions should be scrutinized to determine if an existing module is available to satisfy a requirement. If it is necessary to modify existing software or generate new software, then the standards of Appendix I should be followed.

It is important to note that one of the biggest problems in software reliability is ensuring that only the most recent, accurate version of a program's source or relocatable code is maintained on a system. When multiple versions of code are allowed to exist under the same file name on a system there is a great potential for trouble. Unpredictable and/or incorrect results are likely tc occur. Other than comparing source code line-by-line to look for differences, there is no guaranteed way to determine the age of two files. The simplest (though not guaranteed) way to determine the version of a file is through the revision code entered on the program statement line. Anytime a change is made in a source file, no matter how trivial, the revision code and the documentation history must be updated. This simple action can save much aggravation in the long term.

The greatest danger to file integrity comes if a restore of a file manager cartridge is done. Unless the save operation is current, the possibility of introducing old versions of files is great. Care must be exercised to ensure that whenever updates are done a new file manager save is performed.

3.5.1 Operational Software Changes

Due to the highly structured nature of ADCHK and its supporting programs, changes or additions to the operational software must be carried out with great care. Since ADCHK itself is a very large program already, only relatively simple additions can be made to ADCHK. In general, a program scheduled by ADCHK or a stand-alone program must be used.

If a command is to be added to ADCHK, the following steps should be taken:

a. Add the command characters to the command decode section of ADCHK.

b. Add the command processing code in the appropriate position in ADCHK. All command processing code is in alphabetical order.

c. Update the NCMDS parameter for the menu in ADCHK.

d. Update the documentation section of ADCHK.

e. Using EDITR, add the command to file MENU.

f. Run program MKMNU to update the menu file @MENU.

g. Update the command documentation file xxCMND.

Appendices J (ADCHK data discs and file formats) and L (ADCHK data file format) should be consulted for the storage formats used in operating the BTST. Any changes in these formats have wide ranging implications on all of the software base used to support the BTSTs.

Modification of the calibration and verification software must be approached carefully to maintain a valid control sequence. The GPIB devices' operating manuals must be examined for programming command formats. Appropriate time delays must be inserted to allow the equipment time to respond.

3.5.2 Adding Analytical Software

Writing additional analytical routines is conceptually easier than for operational routines since the sequence of operations is straightforward and there are already subroutines written to handle many of the processing tasks. In particular, the following steps should be taken in order to process data:

a. Obtain the data location.

b. Use subroutine CHLOC to obtain the start locations and number of words for each channel.

c. Select the channel to process.

d. Use subroutine GSCTR to obtain the location of documentation information for the selected channel. The base rate for the channel is required if any time information is desired and the scale factor is required if any amplitude information is desired.

e. Use an EXEC call to read the documentation information.

f. Determine the start point in the data.

g. Use function JDATA to return the data points required for processing.

h. The data word must be tested to determine if it is a time word or a digitized word.

i. The rate must be separated from the A/D output and corrected to a full four bits, if necessary.

j. The data must be converted to 2's complement format if additional processing is to be done. The code IDATA = (IDATA*16)/16 does this efficiently.

4. CONCLUSIONS

It is concluded that the hardware and software elements making up the ballistic test site terminal have evolved into a sophisticated control, data acquisition, and data processing system.

5. RECOMMENDATIONS

It is recommended that funding for research and development of software be continued.

(Page 50 Blank)

SECTION 2. APPENDICES

APPENDIX A - SIGNAL CONDITIONER SPECIFICATIONS

The signal conditioner system is composed of a main frame which provides power, connectors, control and display functions, and houses up to 16 single channel signal conditioners and a GPIB control module.

MAIN FRAME

Number of channels: 16.

Input power: 120 VAC, 60 Hz, 300 W.

Size: 17.8 cm H x 48.3 cm W x 45.7 cm D (7 x 19 x 18 in.).

Temperature: 0° to 40° C operating; -20° to 70° C storage.

Displays:
- Channel address
- All channels selected
- Input mode
- Strain gain
- Strain balance mode
- Auto balance active
- Remote control
- Calibration polarity
- Calibration level
- Calibrator on.

Controls:
- Powers
- Channel step/scan
- Scan rate
- Channel up
- Channel down
- Select all channels
- Input mode
- Input mode load
- Strain gain
- Strain gain load
- Auto/manual balance
- Auto balance initiate
- Remote/local control
- Calibration polarity
- Calibration level
- Calibrator ON/OFF
- Calibration load.
- GPIB address

Connectors: Charge - Microdot 051-0049-0001
(mates with 032-002-001).

Voltage - Amphenol 31-2225
(mates with 31-2226).

Strain - Cannon DAC 15-P
(mates with DAC 15-S).

Output - Same as voltage.

GPIB - As per IEEE 488-1978.

Calibration: 1000 Hz sine wave.
0 to 7.9375 V in 0.0625 V steps (front panel control).
0 to 7.9922 V in 0.0078 V steps (GPIB control).
Linearity of 1% (7 bits).

Model: Precision filters 316-R-01.

SIGNAL CONDITIONER

CHARGE

Bandwidth: 1 Hz to 80 kHz (±0.1 dB).

Sensitivity: 0.1 mV/pC.

Maximum input: ± 100000pC.

Noise: 0.1pC + 0.01 pC/picofarad.

Built-in calibration capacitor: 0.01 microfard ±0.1%.

STRAIN

Bandwidth:

x1	DC to 80 kHz
x10	DC to 25 kHz
x100	DC to 2.5 kHz
x1000	DC to 250 kHz

Configuration: 1, 2, or 4 arm.

Excitation: 0.2 to 30 V at 100 mA
±5 mV resolution
150 microvolts peak-to-peak (0.1 Hz to 20 kHz) noise
0.1% or 500 microvolts per 8 hours stability
0.005%/°C or 100 microvolts/°C temperature stability
0.005% or 300 microvolts line regulation (for ±10% line variation)
0.01% or 300 microvolts load regulation (for no load to full load).

Automatic balance: Range - $(R_{BRIDGE})^2/2*R_{BALANCE}$
Resolution - ±0.1% of range
2 seconds maximum time required.

Calibration: 7 relays each line
Series (128 steps) or shunt (7 steps).

Completion board: Removable assembly using soldered-in resistors for:
Completion network
Balance insertion
Calibration.

VOLTAGE

Bandwidth: DC to 155 kHz.

Note: All charge, strain,and voltage measurements are made using a differential amplifier to remove common mode power supply noise generated by the signal conditioner card.

Model: Precision filters 316-SC-01.

GPIB CONTROL MODULE

GPIB capability: Listener
Remote enable
Go to local
Local lock out.

Programming commands:

ADOO	Address channel 0
.	
.	
.	
AD15	Address channel 15
ADA	Address all channels
IV	Voltage mode
IC	Charge mode
IS	Strain mode
G1	Strain gain x1
G2	Strain gain x10
G3	Strain gain x100
G4	Strain gain x1000
BM	Manual balance mode
BA	Automatic balance mode
ST	Initiate auto balance
CAL pxxx.zON	Turn calibrator on. p = + or -. xxx = 000 to 127. z = 0 to 7 (.z is optional).

CAL pxxx.zON — Turn calibrator on.
p = + or -.
xxx = 000 to 127.
z = 0 to 7 (.z is optional).

OF — Turn calibrator off.

Model: Precision filters 316-C-01.

STRAIN INPUT TO SIGNAL ENTRANCE PANEL CONNECTIONS

Signal Conditioner		Signal Entrance Panel
1	+ EXC	A
2	+ SENSE	B
3	BRIDGE 1	C
4	BRIDGE 2	D
5	BRIDGE 3	E
6	BRIDGE 4	F
7	- SENSE	H
8	- EXC	J
9	+ CAL	K
10	- CAL	L
11		
12	EXC GND	N
13	OUTPUT SHIELD	
14	OUTPUT HIGH	
15	OUTPUT LOW	

BRIDGE COMPLETION NETWORK

STANDARD CONFIGURATION

See Reference 4, drawings C1286, sheet 3, and C1304, sheet 1.

R1, R4: Jumper
R2, R3, R5-R8: Open
R9: 200 k
R10-R30: Open

R31: 60.4 k
R32: 1050
R33: 49.9
R43: 787 k
R44: 196 k
R45: 16.2 k

R34: 124 k	R46: 1.56 M
R35, R36: Jumper	R47: 392 k
	R48: 27.4 k
R37: 243 k	R49: 3.16 M
R38: 5.9 K	R50: 787 k
R39: Jumper	R51: 51.1 k
R40: 392 k	L1, L2: Open
R41: 88.7 k	
R42: 18.7 k	

With these values, the bridge completion network is setup for 4-arm external completion with shunt calibration. The calibration resistors provide changes of 0.25, 0.5, 1, 2, 4, 8, and 16 ohms when a 1000 ohm per arm external bridge is used.

(Page A-6 Blank)

APPENDIX B - SAMPLE RATE AND FILTER CONSIDERATIONS

When an analog tape recorder is used to capture an event, the input waveform is continuously sampled. When a digitizer is used to capture an event, the input waveform is sampled at discrete intervals. The ability to faithfully reproduce the input waveform from the digitized signal is a function of the sample rate versus the highest frequency component of the input waveform. Figure B-1 shows the effect of sampling at different rates on a sine wave input signal. It can be seen that the higher the ratio of sample rate to sine wave frequency, the better the reproduced waveform. For transient waveforms such as generated by ballistic events, ratios in the range of 10 to 20 are needed to allow reasonably accurate peak values to be measured. Ratios of less than 10 are not advisable in most cases. Ratios as high as 100 may need to be used if extremely accurate measurements are required.

An additional problem encountered in digitized signals is that the input signal is normally not synchronized with the digitizer sample clock. Thus, two identical signals digitized at different times will have different sets of sampled points. In particular, the peak signal points will differ. Again this effect can be reduced by sampling at adequate rates.

It can be seen in Figure B-1 that when the sampling ratio falls below 2, not only is the amplitude incorrect but the apparent frequency of the waveform is different. This effect is called aliasing. The apparent frequency is the difference between the sampling frequency and the waveform frequency.

Although it is possible to set the sampling rate to exceed the expected signal frequency, additional higher frequency components may be present in noise associated with the signal. The effect of the noise alias components cause distortion in the signal waveform. To prevent this from happening, an anti-aliasing low-pass filter is desirable in front of the digitizer to remove as much of the noise as possible. In the BTST the filter/amplifier performs this function.

Three types of filter characteristics are available in the filter/amplifier cards: LP1, TD2, and a modified TD2. The LP1 is an elliptic low-pass filter with six poles and six zeroes. The TD2 is a Bessel low-pass filter with six poles. The modified TD2 is a Bessel low-pass filter with additional zeroes. The modified Bessel is achieved by moving jumpers on the TD2 card. Figure B-2 shows how the jumpers are arranged for the two filter types.

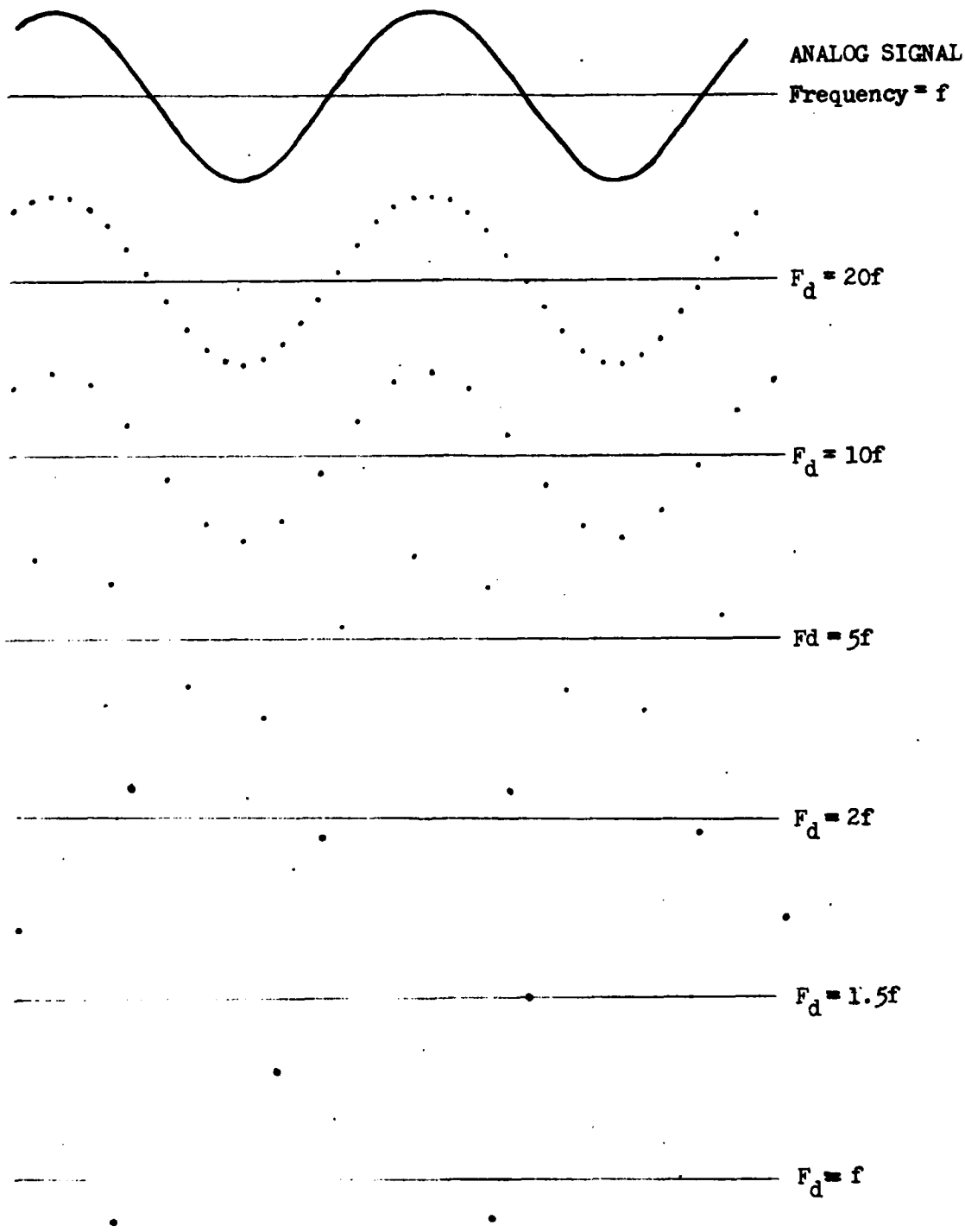
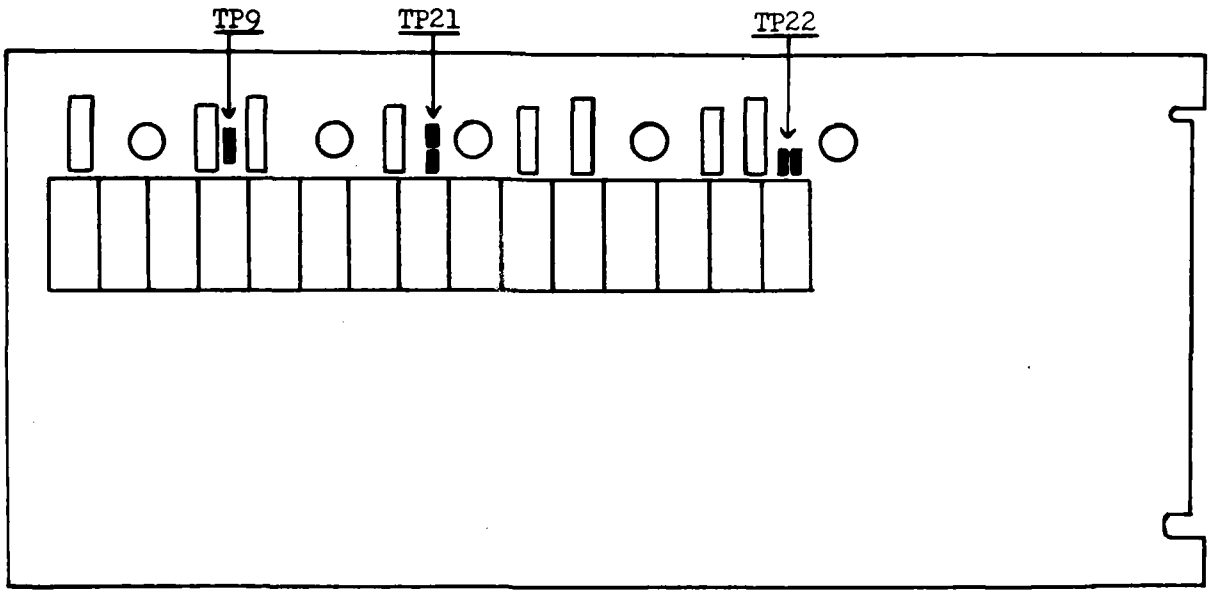
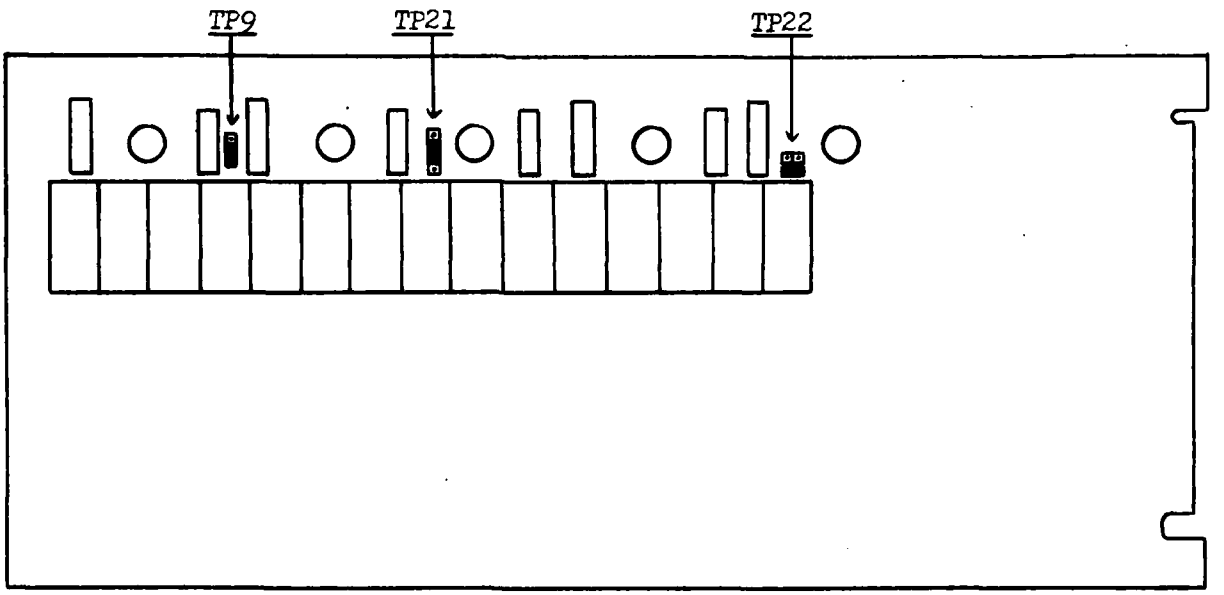


Figure B-1. Sine wave digitized at different sampling rates.



TD2 BESSEL JUMPER POSITIONS



TD2 MODIFIED BESSEL JUMPER POSITIONS

Figure B-2. Filter jumper positions for normal and modified response.

The transfer function of the three types of filters is in Figure B-3. The transfer function curves were generated using a Hewlett-Packard 5423A waveform analyzer.

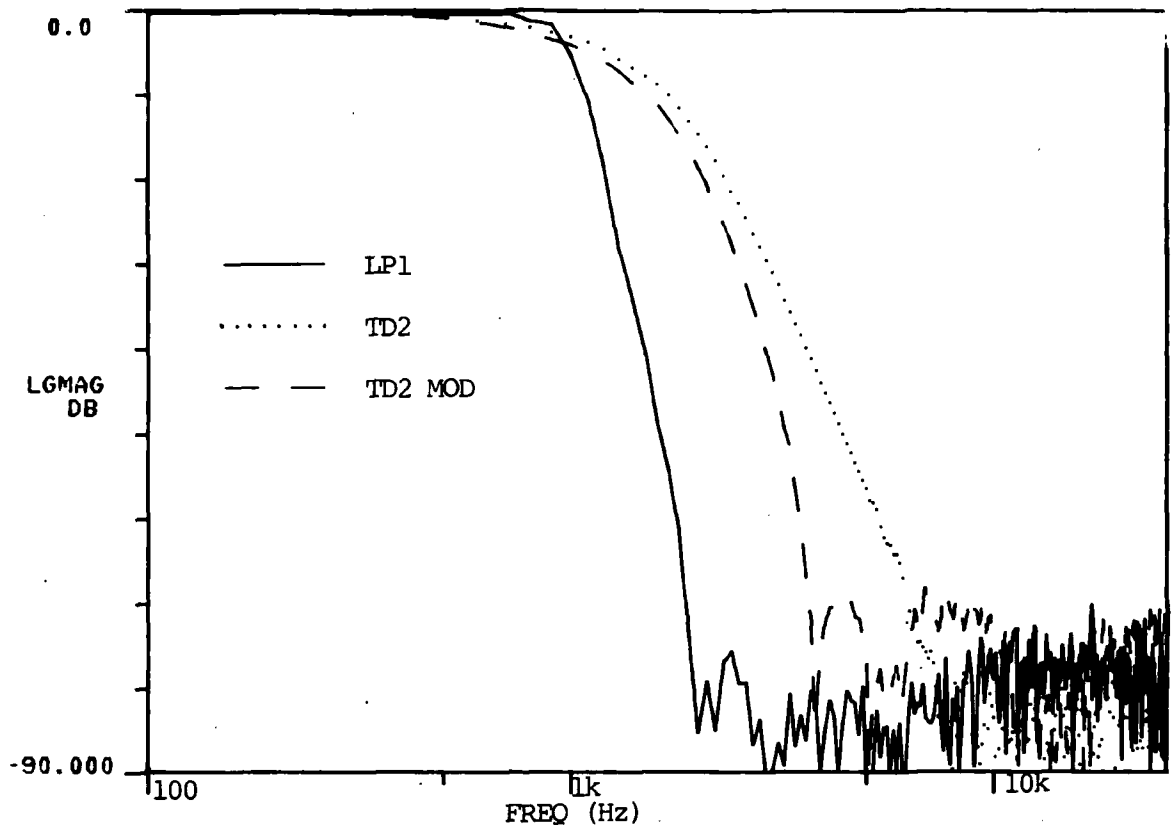
The LP1 provides the greatest alias protection, the modified TD2 the second best protection, while the TD2 provides the least amount of protection. For situations where the signal to noise ratio is low or when transducer resonances are present, the use of the best anti-aliasing filter is essential.

The step response of the three types of filters is in Figure B-4. The step response curves were generated using the programmable signal generator and automatic sampling rate digitizer. The TD2 and modified TD2 have nearly identical step response characteristics, whereas the LP1 has very poor step response characteristics. The vast majority of ballistic events can be categorized as transient in nature with the step response characteristics being of primary concern.

The cutoff frequency of the filter must be set from knowledge of either the bandwidth or the rise time of the signal from the transducer. The bandwidth and risetime are related by

$$\text{Bandwidth} \times \text{Risetime} = 0.35.$$

The final choice of filter type, filter frequency and digitizing rate must be made taking into account the transducer characteristics, signal duration, noise sources, available memory and accuracy requirements. In most cases tradeoffs will be required.



Figures B-3. Transfer function of filters used in BTST (cutoff frequency of 1 kHz).

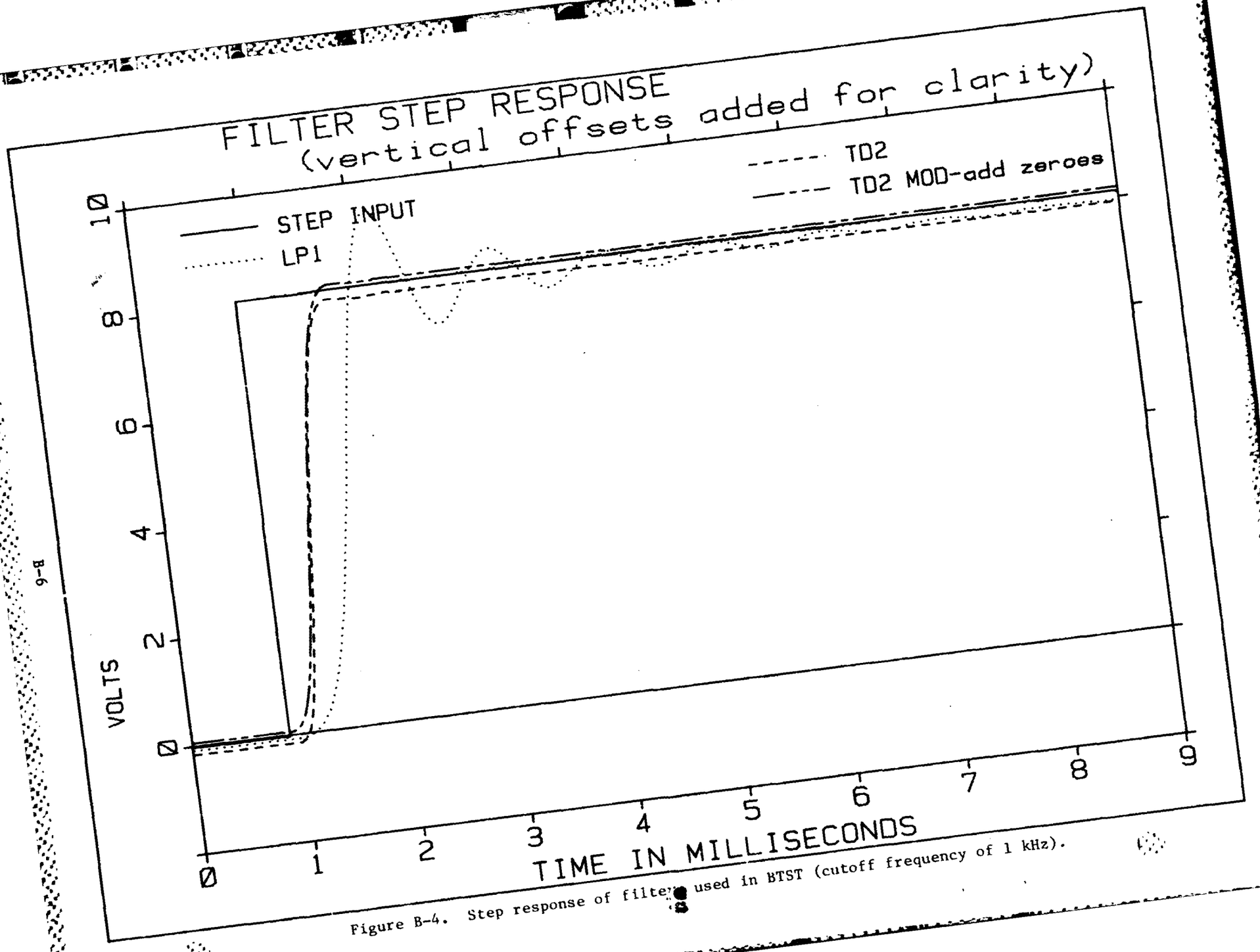


Figure B-4. Step response of filters used in BTST (cutoff frequency of 1 kHz).

APPENDIX C - FILTER/AMPLIFIER SPECIFICATIONS

The filter/amplifier system is composed of a main frame which provides power, connectors, control and display functions, and houses up to 16 single channel filter/amplifiers and a GPIB control module.

MAIN FRAME

Number of channels: 16.

Input power: 120 VAC, 60 Hz, 300 W.

Size: 17.8 cm H x 48.3 cm W x 45.7 cm D (7 x 19 x 18 in.).

Temperature: 0° to 40° C operating/-20° to 70° C storage.

Displays:
- Channel address
- All channels selected
- Prefilter gain
- Postfilter gain
- Cutoff frequency
- Low-pass card installed
- Time delay card installed
- Remote control
- Prefilter overload
- Postfilter overload
- System overload

Controls:
- Power
- Channel step/scan
- Scan rate
- Channel up
- Channel down
- Select all channels
- Prefilter gain
- Prefilter gain load
- Postfilter gain
- Postfilter gain load
- Cutoff frequency
- Cutoff frequency load
- Remote/local control

Connectors:
- Input - Amphenol 31-2225 (mates with 31-2226).
- Output - BNC.
- GPIB - As per IEEE 488-1978.

Model: Precision filters 416-R.

FILTER/AMPLIFIER

Input type: Differential.

Input impedance: 20 million ohm differential.
5 million ohm common mode.

Maximum input: ±10 V peak without clipping.
±40 V peak without damage.

Common mode rejection ratio: 100 dB (50 to 500 Hz)
>60 dB at 100 kHz.

Output type: Single-ended.

Output impedance: 51 ohm.

Output level: ±10 V peak (±15 milliamperers).

Noise: 300 microvolts at 150 kHz setting with
15 MHz measurement bandwidth.

Drift: 250 microvolts/°C at minimum gain.

Prefilter gains: 1, 2, 5, 10, 20, 50.

Postfilter gains: 1, 2, 5, 10, 20.

Distortion: -80 dB referenced to full scale.

Cross-talk: -100 dB referenced to full scale.

Hum: -100 dB referenced to full scale.

Overload detector level: ±10 V.

Overload detector accuracy: ±2%.

Minimum overload duration: 2 microseconds.

ELLIPTIC FILTER

Type: 6 pole, 6 zero low pass.

Pass-band ripple: 0.1 dB peak-to-peak.

Stop-band attenuation: 80 dB minimum.

Attenuation: 0.1 dB at F_c.
3 dB at 1.08 F_c.
45.5 dB at 1.69 F_c.
60 dB at 1.92 F_c.
80 dB at 2.13 F_c.

Settling time: $1.29/F_c$ to 10%.
$1.85/F_c$ to 5%.
$4.90/F_c$ to 1%.

Model: Precision filters 416-01-LP1-05-M

BESSEL FILTER

Type: 6 pole.

Pass-band ripple: 0.2 dB peak-to-peak.

Stop-band attenuation: 80 dB minimum.

Attenuation: 3 dB at F_c.
14 dB at $2F_c$.
31 dB at $3F_c$.
45 dB at $4F_c$.
80 dB at $7F_c$.

Settling time: $1/F_c$ to 1%.

Model: Precision filter 416-01-TD2-05.

MODIFIED BESSEL FILTER

Type: 6 pole, 6 zero.

Pass-band ripple: 0.2 dB peak-to peak.

Stop-band attenuation: 70 dB minimum.

Attenuation: 4 dB at F_c.
18 dB at $2F_c$.
43 dB at $3F_c$.
71 dB at $4F_c$.

Settling time: 1/Fc to 1%.

Model: Precision filters 416-01-TD2-05.

GPIB CONTROL MODULE

GPIB capability: Listener
Talker
Remote enable
Go to local
Local lockout
Serial poll.

Programming commands:

C00	Address channel 0.
.	
.	
.	
C15	Address channel 15.
CA	Address all channels.
Axx	Set prefilter gain, xx = 01, 02, 05, 10, 20, or 50.
Gxx	Set postfilter gain, xx = 01, 02, 05, 10, or 20.
FxxxMz	Set frequency, xxx = 005 to 155 in steps of 5, z = 0 (x0.1) or z = 1 (x1).
0	Return overload codes when placed in talker mode.
R	Return current setup information when placed in talker mode.

Output format:

Overloads: 0 - no overload.
1 - prefilter overload.
2 - postfilter overload.
3 - both overloaded.

Status: $\begin{matrix} L \\ T \end{matrix}$AxxGxxFxxMx

where L = elliptic and T = Bessel and A, G, F, and M are as defined above.

Model: Precision filters 416-C-01B (talker capability)
416-C-01 (nontalker capability).

APPENDIX D - ASRD CHARACTERISTICS

The ASRD consists of a housing which provides power, cooling, and interconnections for an event channel and for up to 16 ASRD channels.

HOUSING

ASRD channel capacity:	16.
Event channel capacity:	1.
Power input:	120 VAC, 60 Hz, 1000 W.
Power supplies:	+5 VDC at 150 A (logic). ±15 VDC at 2.4 A (A/D board). -12 VDC at 1.7 A (FIFO). +15 VDC at 15.8 A (memory board).
Cooling:	550 cfm.

EVENT CHANNEL

Number of inputs:	8.
Input type:	TTL.
Time resolution:	1 microsecond.
Trigger:	Any change in input lasting longer than 200 nsec.
Minimum time between changes:	2 microseconds.
Number of events:	40.
First event:	Time and event line status when ASRD is armed.
Master clock frequency:	8 MHz.
Display:	Arm enabled indicator.
Control:	Microprocessor master reset.
Commands supported:	1 - Read day from time code generator and transfer to computer. 2 - Read time and transfer to computer. 3 - Transfer number of events to computer. 4 - Transfer events data to computer.

Connectors: Computer interface - 3M 3433 (mates with 3425).

Time code input - 3M 3433.

ASRD interface - 3M 3433.

Power input/clock output - MS3122E22-55P (mates with MS3121F22-55S).

Events input - BNC.

Microprocessor system: 8085 8-bit microprocessor
(2) 8155 256 byte random access memory (RAM) - input/output (I/O) port
8185 1 k byte RAM
8755 2 k byte erasable, programmable read-only memory (EPROM) - I/O port.

Data formats:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	D200	D100	D80	D40	D20	D10	D8	D4	D2	D1
0	0	0	0	0	0	0	0	0	0	N32	N16	N8	N4	N2	N1
0	0	0	0	0	0	0	0	E7	E6	E5	E4	E3	E2	E1	E0
0	0	U512	U256	U128	U64	U32	U16	U8	U4	U2	U1	H20	H10	H8	H4
0	0	H2	H1	M40	M20	M10	M8	M4	M2	M1	S40	S20	S10	S8	S4
0	0	S2	S1	T800	T400	T200	T100	T80	T40	T20	T10	T8	T4	T2	T1

where D = days, N = number of events, E = event channel value, H = hours, M = minutes, S = seconds, T = milliseconds, and U = microseconds.

ASRD CHANNEL

Input level: ±10 V.

Input impedance: 10 k ohm (12 bit)/3 k ohm (10 bit).

Digitizing rates: 97.7 to 800,000 samples per second in binary sequence.

Resolution: 12 bits/10 bits (depends on A/D board).

Memory size: 16 k to 128 k in 16 k steps. Each memory word is 16 bits wide.

Digitizing modes:	Fixed sample rate (FSR) - all samples taken at same rate. Adaptive sample rate (ASR) - rate of change of signal determines sample rate.
Trigger modes:	Internal External Slave Master from internal Master from external Master from internal or slave Master from external or slave.
Internal trigger modes:	Normal - threshold and slope. Absolute - absolute value of threshold. Relative - DC offset at arm added to programmed threshold to give trigger threshold.
Trigger qualifiers:	Run delay - time from trigger to start of data recording. Stop delay - time from trigger lost to stop of data recording. Pretrigger memory - memory prior to trigger point.
Retrigger:	Allowed if memory is still available.
Clock source:	Event channel master clock.
Synchronization:	At arm time.
Time marking:	Hours, minutes, seconds, milliseconds, and microseconds from events channel inserted at trigger point.

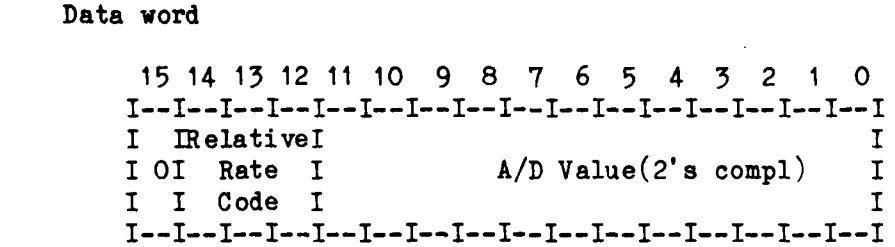
Programmable parameters:	Channel enabled Trigger specification Master Slave External/internal Normal Absolute Relative Threshold Slope Run delay Stop delay Pretrigger memory Memory size Digitizing mode FSR - rate ASR - rate table decrease delay averaging length scaling value.
Microprocessor system:	8085 8-bit microprocessor. 8155 256 byte RAM - I/O port. 8755 2 k byte EPROM - I/O port. 2716 2 k byte EPROM
Internal self-test:	Data memory. Programmable registers. Rate clock.
Connectors:	Analog input - BNC. External trigger input - BNC. Channel trigger output - BNC. Power/clock/address - MS3470L14-12P (mates with MS3476L14-12S). Command/time bus - printed circuit edge connector (mates with 3M 3415-0001).
Displays:	Current data/memory word or microprocessor data. Channel triggered.
Size:	5.1 cm H x 42.4 cm W x 52.4 cm D (2 x 16-11/16 x 20-5/8 in.).

Commands supported:

0	NOP.
1	Zero error status register.
2	Perform fast memory check.
3	Load parameters to registers.
4	Display ADC on front panel.
5	Perform complete memory test.
6	Enable channel to take data.
7	Disable channel to take data.
8	Perform hardware register test.
9	Resize memory.
10	Transfer trigger set indicator to host.
11	Specify decrease delay.
12	Specify trigger mode, slope and threshold.
13	Specify stop delay.
14	Specify start delay.
15	Specify calculation control register.
16	Specify memory size.
17	Specify pertrigger area.
18	Transfer current ADC value to computer.
19	Calculate and transfer average and noise level to computer.
20	Specify starting address for data transfer.
21	Specify number of words to transfer.
22	Transfer error words to computer.
23	Transfer starting address (current acq cycle).
24	Transfer number of words collected.
25	Transfer serial number and software revision code.
26	NOP.
27	NOP.
28	Transfer memory error location and contents to host.
29	Transfer register contents to host.
30	Specify rate table.
31	Transfer rate table to computer.
32	Transfer data to computer-DMA, auto adrs and nwds.
33	Transfer data to computer-DMA, specified adrs and nwds.
34	Transfer data to computer-PRGM, auto adrs and nwds.
35	Transfer data to computer-PRGM, specified adrs and nwds.
36	Transfer data from computer to memory-specified adrs and nwds.
37	NOP.
38	NOP.
39	NOP.

Data word format:

Data word



Rate code = relative rate code + base code.

Rate = 800,000 sps/$2^{(15 - \text{rate code})}$.

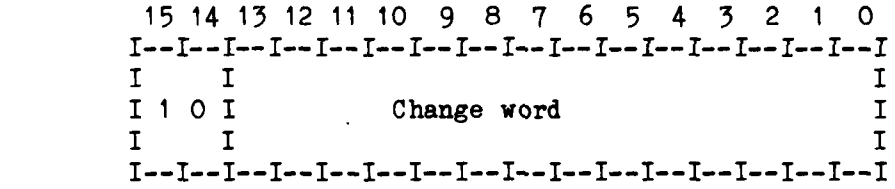
A/D values are:

+2047 = 0 1 1 1 1 1 1 1 1 1 1 1

+1 = 0 0 0 0 0 0 0 0 0 0 0 1
0 = 0 0 0 0 0 0 0 0 0 0 0 0
-1 = 1 1 1 1 1 1 1 1 1 1 1 1

-2048 = 1 0 0 0 0 0 0 0 0 0 0 0

Change word



Change words are inserted when the variable sampling rate mode is used.
Change words are inserted between the last sample taken at one rate and the first sample taken at a higher or lower rate.
The time interval between the two samples is given by:
t(2) - t(1) = Change word * 1.25 microseconds.

IRIG time words:

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	U512	U256	U128	U64	U32	U16	U8	U4	U2	U1	H20	H10	H8	H4
2	1	1	H2	H1	M40	M20	M10	M8	M4	M2	M1	S40	S20	S10	S8	S4
3	1	1	S2	S1	T800	T400	T200	T100	T80	T40	T20	T10	T8	T4	T2	T1

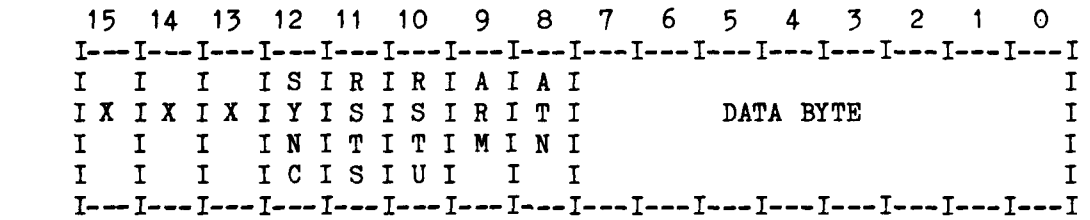
where H = hours, M = minutes, S = seconds, T = milliseconds, and U = microseconds.

Time words are inserted each time the system is triggered.
Word 4 provides the time between the sample saved in memory and the sample (taken at 800,000 sps) which occurred closest to trigger.
The time interval between the two samples is given by:

$$t(2) - t(1) = (\text{Relative trigger word} - 1) * 1.25 \text{ microseconds.}$$

If a change word occurs during the same master clock cycle, the change word shall replace the relative trigger word. In this case there will be only three IRIG time words.

EVENT AND ASRD COMMAND FORMAT



where: ATN = attention. With bit 8 set and bit 7 of the data byte set, bits 0-6 are the channel address.
ARM = ARM all selected channels to take data.
RSTU = RESET Microprocessor.
RSTS = RESET system.
SYNC = SYNC channels rate clocks.
X = Not used.

If bits 8 - 15 are not set, then the data byte is used to transfer the command code and associated data.

(Page D-8 Blank)

APPENDIX E - BUILT-IN TEST EQUIPMENT SPECIFICATIONS

E.1 SCANNER

Switching: Make before break.

Type contacts: Three pole, low thermal reed relays per channel (high, low, and guard).

Maximum contact ratings: 42 V peak at 40 mA.

Maximum input: 42 V peak channel to channel. 42 V peak channel to chassis.

Isolation: 10 million ohms.

Switching time: 1 ms.

Controls: Power.

Display: Channel closed.

Power: 120 VAC, 60 Hz, 100 W.

Temperature: 0 to 55° C operating, -40 to 75° C storage.

Size: 19.1 cm H x 42.9 cm W x 52.1 cm D (7.5 x 16.9 x 20.5 in.).

Jumpers: Internally jumped so each bank is independent of the others.

GPIB capability: Listener.

Programming commands:

xx	Close channel xx, open any channel in same bank currently closed, xx = 00 to 79.
C	Open all channels.
E	Execute command.

GPIB address: Set internally by solder jumpers (factory set to 9).

Model: Hewlett-Packard 3495A with four option 004 duodecade relay cards.

Initial setup: The internal switches on each bank must be set as shown in Figure E-1 to allow the scanner to work with the BTST software.

Precaution: Scanner rear panel door must be locked in place to provide reliable scanner operation.

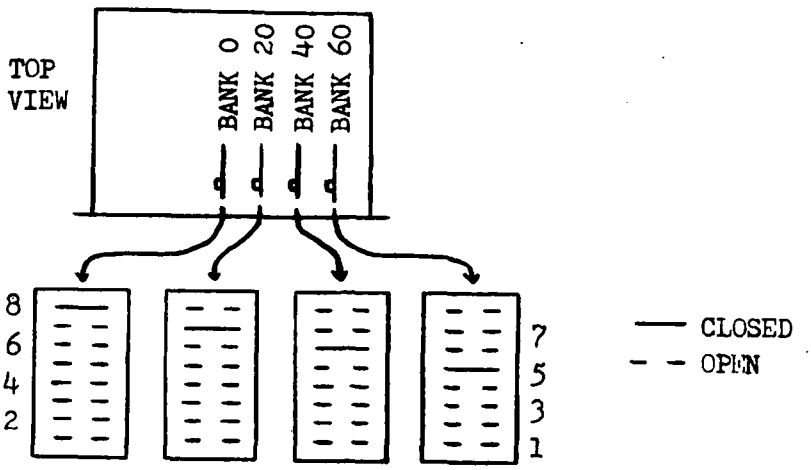


Figure E-1. Scanner bank address switch settings.

E.2 OSCILLOSCOPE

Bandwidth: DC-500 kHz.

Vertical sensitivity: 0.1 mV/division to 20 V/division.

Modes: Channel A, channel B, channels A and B (chopped or alternate).

Input type: Differential or single ended.

Input impedance: 1 million ohm.

Input coupling: DC, AC or OFF.

Common mode rejection: Minimum 30 dB over DC to 10 kHz.

Trigger source: Internal, external, or line.

Trigger mode: Free run, normal, or single sweep.

Power: 115 VAC, 60 Hz, 150 W.

Size: 13.3 cm H x 48.3 cm W x 46.6 cm D (5.2 x 19 x 18.4 in.).

Model: Hewlett-Packard 1200B.

E.3 DIGITAL VOLTMETER

DC VOLTS

Input impedance: 10^{10} ohms.

Maximum input voltage: ±1000 V peak high to low.
±500 V peak guard to chassis.
±200 V peak low to guard.

Ranges: 1, 10, 100, and 1000 V.

Accuracy (long term): 0.009% + 5 digits.

AC VOLTS

Input impedance: 2 Mohms.

Maximum input voltage: ±1414 V peak high to low.
±500 V peak guard to chassis.
±200 V peak low to guard.

Ranges: 1, 10, 100, and 1000 V rms.

Accuracy (long term): 0.06% + 60 digits (30 Hz to 20 kHz).
0.6% + 130 digits (20 kHz to 100 kHz).

Reading rate: 1.5 readings per second.

GENERAL

GPIB capability: Talker
Listener
Service request
Device clear
Device trigger.

Programmable commands:

F1	DC V
F2	AC V
F3	Fast AC V
F4	2-wire ohms
F5	4-wire ohms
F6	Test.
R1	0.1 range
R2	1 range
R3	10 range
R4	100 range
R5	1000 range
R6	10 k range
R7	Auto range.
T1	Internal trigger
T2	External trigger
T3	Hold/manual trigger.

M1 Scale math mode
M2 Error math mode
M3 Math mode OFF.

EY Enter Y parameter
EZ Enter Z parameter
SY Store Y parameter
SZ Store Z parameter.

A0 Auto calibrate OFF.
A1 Auto calibrate ON.

H0 High resolution mode OFF.
H1 High resolution mode ON.

D0 Data ready request OFF.
D1 Data ready request ON.

B Binary program mode.

Output format: ±D.DDDDDD±DD.

Power: 120 VAC, 60 Hz, 60 W.

Temperature: 0 to 50° C operating, -40 to 75° C storage.

Warmup time: One hour.

Size: 8.9 cm H x 42.6 cm W x 52.7 cm D (3-1/2 x 16-3/4 x 20-3/4 in.).

Model: Hewlett-Packard 3455A.

E.4 PROGRAMMABLE SIGNAL GENERATOR

Frequency range: 0.001 Hz to 50 MHz.

Frequency accuracy: ±0.001%.

Output range: 10 mV p-p to 10 V p-p.

Output accuracy: ±2%.

Offset range: 0 to ±5 V.

Offset accuracy: ±2%.

Waveforms: Sine, square, or triangle.

Modes: Normal, voltage controlled oscillator, triggered, gate and burst.

Programmability: Front panel or GPIB.

Power: 120 VAC, 60 Hz, 200 W.

Temperature: 0 to 50° C (operating); -20 to 70° C (storage).

Size: 14.5 cm H x 42.6 cm W x 45.0 cm D (5.7 x 16.8 x 17.7 in.).

GPIB capability: Talker
Listener
Service request
Device trigger
Remote/local.

Programmable commands:

Frequency (millihertz)	FRQxMZ	x = 1 to 9999
(hertz)	FRQxHZ	x = 0.001 to 9999
(kilohertz)	FRQxKHZ	x = 0.001 to 9999
(megahertz)	FRQxMHZ	x = 0.001 to 50.
Amplitude (millivolts)	AMPxMV	x = 10/20 to 999
(volts)	AMPxV	x = 0.01 to 10/20.
Offset (millivolts)	OFSxMV	x = ±10/20 to ±999
(volts)	OFSxV	x = ±0.01 to ±5/10.
Burst length	BURxBT	x = 1 to 9999.
Function	Fx	1 = sine, 2 = trngl, 3 = square.
Duty cycle	Dx	1 = 20%, 2 = 50%, 3 = 80%.
Trigger	Tx	1 = norm, 2 = VCO, 3 = ext, 4 = gate, 5 = burst.
Frequency modulation	FMx	0 = OFF, 1 = ON.
Output modulation	Ox	D = disable, E = enable.
Output polarity	Ox	I = inverted, N = normal.
Output impedance	Ox	1 = 1 k ohm, 5 = 50 ohm.
Store parameters	STOx	x = 0-9.
Recall parameters	RCLx	x = 0-9.

Status BYTE: OS00ECCC.

S = service request.

E = error flag.

CCC = error code, 0 = amplitude,
1 = offset, 2 = frequency,
3 = impedance, 4 = duty cycle,
7 = syntax.

Model: Hewlett-Packard 8165A.

E.5 PROGRAMMABLE COUNTER/TIMER

Time base: 10 MHz.

Aging rate: Less than one part in 1,000,000,000 per month.

Modes: Time interval and frequency.

Frequency range: 0 to 100 MHz.

Time interval range: 0.1 microsecond to 1,000,000,000 seconds.

Input coupling: AC or DC.

Trigger slope: + or -.

Trigger level: 3.00 V to -3.00 V in 0.01 V steps.

Input attenuator: X1, X10, or X100.

Input impedance: 1 Mohm.

Maximum input: 250 V rms.

Programmability: Front panel or GPIB.

GPIB capability:
Talker
Listener
Service request
Device trigger
Device clear
Remote/local.

Programmable commands:

Frequency - channel A	F0
Frequency - channel C	F1
Ratio B/A	F2
Ratio C/A	F3
Period - channel A	F4
Time interval	F5

Period average - channel A	F6
Time interval average	F7
Frequency - channel B	F8
Events on C during A to B	F9
Totalize	F:
Range	Rx x = 0 to 7
Positive slope	n0 n = A or B
Negative slope	n1
DC coupling	n2
AC coupling	n3
X1 attenuation	n4
X10 attenuation	n5
X100 attenuation	n6
Automatic trigger level	LnA
Programmed trigger level:	Ln±xxx xxx = -300 to +300.
Separate inputs	C0
Common input - channel A	C1
Test	C2
Single reading - no SRQ	S0
Single reading with SRQ	S1
Multiple reading - no SRQ	S2
Multiple reading with SRQ	S3
Software trigger - single reading	T
Reset	R

Output format: .XXXXXXXXE±XX.

Reading time: 3.3 ms.

Power: 120 VAC, 60 Hz, 80 W.

Size: 8.9 cm H x 48.3 cm W x 40.7 cm D (3.5 x 19 x 16 in.).

Temperature: 0 to 50° C (operating), -40° to 70° C (storage).

Model: Racal-Dana 9514 with options 01, 22, and 60.

(Page E-8 Blank)

APPENDIX F - COMPUTER EQUIPMENT OPERATION DETAILS

F.1 COMPUTER

F.1.1 Memory

The layout of the memory slots in the computer is shown in Figure F-1. The memory slots are accessible through the front. A minimum of 64 k words of memory is required with 128 k words typical. A variety of memory cards is used in the systems. Different boards may be interchanged as long as they are jumpered properly and they are the same speed as the controller.

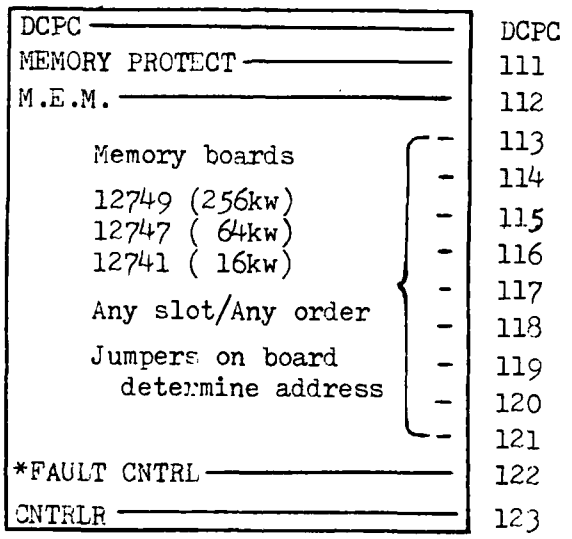


Figure F-1. Computer memory slots.

In the bottom slot is the memory controller card. If used, the next to the bottom slot may have a fault control board which detects and corrects single bit errors and detects multiple bit errors. In the top slot is the dual channel port controller (DCPC) card which provides direct memory access (DMA) capability. The second slot from the top houses the memory protect card. The memory protect card traps all attempts by a program to operate outside its allocated memory space. Third from the top is the memory expansion module (MEM) which provides extended addressing capability. All of these cards must be installed in the correct slot for proper operation.

F.1.2 I/O

The I/O slots are accessed from the rear of the computer. The order and type of I/O cards varies from system to system. The correct configuration can be found from the system generation listing. There can be no empty slots between installed cards. Note that slot numbering begins at 10 and is done in octal. The cards typically found in a system include:

Time base generator - Provides a crystal controlled time base for generating operating system interrupts.

Disc interface - Provides the interface between the computer and the disc controller.

Buffered asynchronous communications interface - Provides the interface between the computer and the terminal.

Mag tape interface - Provides the interface between the computer and the 9-track mag tape drive. Two consecutive card slots are required. Card one goes in the lower slot.

GPIB interface - Provides the interface between the computer and GPIB devices.

Required switch settings for this card are:

PP IO Code		Configuration	
1-8:	Open (OFF)	1-4:	Closed
		5-7:	Open
		8:	Closed

Note that the card uses one GPIB address itself (address 0).

Microcircuit register interface - Provides the interface between the computer and ASRD.

This card may be either a 12566B or 12566C card. The jumpers required for the cards are:

	12566B	12566C
W1	B	B
W2	B	B
W3	B	B
W4	B	B
W5	Open	In
W6	Open	In
W7	Open	In
W8	Open	In
W9	A	A
W10	NA	Don't care
W11	NA	OUT
W12	NA	IN
W13	NA	OUT

F.1.3 Firmware

Much of the capability to perform specialized tasks resides in firmware installed in the computer. Each operating system has unique features which are implemented in microcode resident in firmware. Additional features such as fast FORTRAN processing, distributed systems, etc., are implemented in a similar manner. As a consequence, if CPUs are swapped, the interchangeability is governed not only by physical considerations but also by the firmware installed.

Typically the model numbers of firmware options are listed inside the front panel of the computer. On older computers the ROMs are installed inside the computer, while on newer models the ROMs are installed on the firmware expansion module in slot 10 of the I/O slots.

F.1.4 Boot Loader ROMs

In order to initially load programs into computer memory, up to four boot loader ROMs may be installed on the CPU board. There are typically two boot loader ROMs installed for each computer: one for loading the operating system from disc and one for loading absolute programs (off-line backup and diagnostics) from cassette tape. The selection of loader ROM is made at boot time from bits 15 and 14 of the front panel switch register. The correct code is normally shown on the inside front panel of the computer.

F.1.5 Front Panel

The front panel of the computer contains a number of switches and displays. These are used in booting the system, displaying status on certain programs and can be used for controlling the computer for diagnostic purposes. Figure F-2 shows the front panel arrangement.

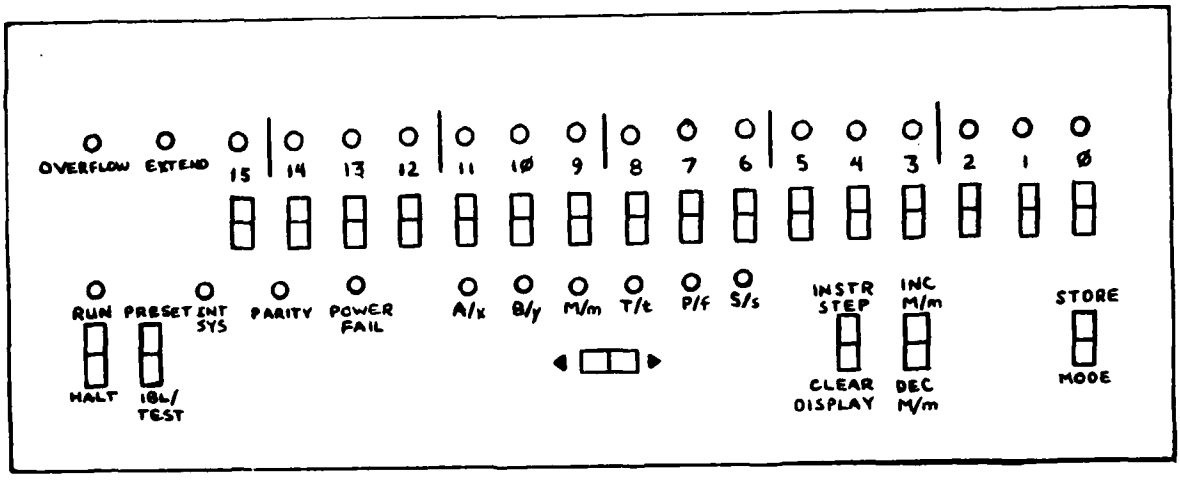


Figure F-2. Front panel of computer.

F.1.6 Boot Procedure

a. Press HALT.

b. Select the S register.

c. Press CLEAR DISPLAY.

d. Press the bits for the load desired:

15-14 Loader ROM code.

13 Not used.

12 Set to 1.

11-6 Select code of disc or terminal.

5 Set only for reconfigure boot.

4-0 Set for surface of disc (normally 0).

e. Press STORE.

f. Press PRESET.

g. Press IBL/TEST.

h. Press RUN.

Load should occur.

F.1.7 Computer Troubleshooting

a. Power computer down before removing or inserting any board.

b. If memory problems are encountered, reseat all boards in memory slots.

c. If computer will not boot, reseat all I/O and memory boards.

d. If computer runs slow, remove cable from serial interface card.

e. When the computer is powered up, it performs an internal self-test. If the test fails, then all bits on the register select and/or switch register will be on.

F.2 DISK DRIVE AND CONTROLLER

The controller used in the BTST is called a multiple access controller (MAC). The controller converts the computer commands into commands which the disc can handle. There are no controls or displays associated with the controller.

The disc drive has three switches which must be set properly:

Unit select - 0.

Format - yes.

Read only - no.

In addition, during operation the RUN/STOP switch must be in the RUN position.

The disc drive has a number of built-in sensors and test criteria. If any of the test criteria are out of tolerance a drive fault light will come on and the disc will cease to function. If due to transient effects, the drive fault can be cleared by powering the controller and disc drive down and then back up again. The 7920 and 7925 disc drives have additional indicators inside to specify the cause of the fault. If a permanent drive fault is encountered, the cause should be reported when the maintenance call is placed.

The controller has two diagnostic LEDs mounted on the right hand side of the microprocessor board. The front panel of the controller must be removed to view them. If the LEDs are on, then a problem exists in the controller. Both the disc drive and controller have internal fuses as well as rear panel mounted fuses.

If a high pitched buzzing or grinding noise is heard from the disc drive, it is likely that a head crash has occurred. Maintenance is required to correct a head crash. Never substitute a new disc pack in a drive if a head crash is suspected.

F.3 TERMINAL

The terminal has two modes of operation: local or remote. For all BTST operations the terminal is used in remote mode. There are a number of switches which affect the operation of the terminal. The required settings are:

REMOTE - down.

CAPS LOCK - down.

All other push button switches - up.

Duplex - full.

Parity - none.

Baud rate - 9600.

Certain programs activate special capabilities of the terminal (memory lock, display functions, etc.). If the program aborts, these functions will remain enabled which can cause problems. Pressing the appropriate function key will correct the situation.

If the terminal becomes locked up for some reason, it can be cleared by pressing the RESET key twice rapidly. This will clear memory, reset all of the enhanced functions and clear the special function keys.

The terminal has a built-in self-test function. When this key is pressed, the terminal checks its RAM, ROM, and graphics capability. The communications interface is not checked.

When the terminal is activated in the remote mode by the computer, the transmit LED should be on. Normally, no other LEDs should be on unless an enhanced function is being performed.

The rear door on the terminal should be locked in place to prevent the keyboard connector or the interface connector from pulling loose.

F.4 MAG TAPE DRIVE

Prior to use, a tape must be mounted on the drive. A tape loading diagram is mounted inside the drive. After mounting the tape, the LOAD and ON LINE buttons must be pressed. If a write operation is to be performed, a write ring must be installed in the tape.

The unit selector switch in the lower left corner must be set to zero. The drive will not work in any other position.

Periodically the heads and pivots on the drive must be cleaned to ensure error-free operation. Instructions are in the tape drive manual.

When the drive is moved, check that the front case of the drive is locked in place. Otherwise the case may swing open in transit.

If a tape write operation is started, but cannot be completed for some reason and the program is aborted, the tape drive buffer will still contain the original information. If the situation which caused the problem is corrected, the tape drive will write the buffer to tape. The extra buffer written on the tape can cause problems when the tape is read later. Do not abort a program when tape write problems are encountered. Correct the problem, then UP, 8.

F.5 PRINTER/PLOTTER

The printer/plotter has a number of switch settings on its rear panel. The correct settings are:

Address selector must point to address switch.

Address switch - 1.

Error beep - ON.

6LPI/8LPI - 8LPI (but 6LPI is okay).

SI/SO/8 bit - 8 bit.

ENG/METRIC - as appropriate for paper.

SCALED/9872 - 9872.

Note that the printer/plotter uses two bus addresses: 1 is the plotter and 2 is the printer. Address 2 must not be used on any other device.

An internal confidence test can be initiated by pressing the button on the rear panel. This will cause a test grid and lettering to be drawn. This test does not check the communications interface.

When the paper used by the printer runs low, a red line appears on the edge of the paper.

APPENDIX G - GPIB USAGE GUIDELINES

The GPIB provides an industry standard connector, wiring, handshake, and data transfer protocol for interfacing equipment to a computer. Since the GPIB allows several devices to be connected to the same computer interface card, it provides a very versatile and inexpensive control mechanism. There are several general rules which must be followed to ensure that proper operation is achieved.

1. Each device connected on the bus must have a unique address. Addresses are set using small switches normally on the back panel of the instrument. Occasionally, the switch may be inside the instrument or jumpers on a circuit card may be utilized. If two devices have the same address, unpredictable results may occur. Some devices may use two or more addresses. Even though only one address switch setting is made on the instrument, the other addresses are not available for use by other instruments.

2. The available address range of the bus is 0 to 31. However, only addresses 1 to 30 should be used on instruments. The HP 1000 uses address 0 as the system controller address.

3. Although 30 useable addresses are available, only 14 devices should be connected to the bus at any time. Each device places a load on the bus. If too many loads are connected, the bus will cease to function or function erratically. Unfortunately the number 14 is not a rigid limit. Depending on the type of devices, length and configuration of cables and controller card used, a lower or higher limit may be found.

4. The total length of cables used to connect all devices must not exceed 20 m. Again, this is not a hard and fast limit. Always use the shortest cables possible. In general, a branching cable connection setup will work better than a single linear connection.

5. The power on at least half of the devices connected must be on to ensure reliable operation. Highest reliability is obtained when power to all devices connected to the bus is on.

6. Never turn the power to a device on or off while bus transactions with other devices are taking place.

7. Some devices have talk only or listen only options on their address switch. These options should never be used on the same bus as addressed devices.

8. When a device is placed in the remote mode, local front panel control is normally disabled. A return to local pushbutton must be exercised to return control to the front panel.

9. Each device has its own set of control capabilities and programming codes. The individual device manuals must be consulted to determine each device's peculiarities. A summary of each devices programming capabilities is included in the appropriate hardware section of the manual.

10. If the bus hangs up and no transactions occur, power all devices connected to the bus off and then power them all up again.

The address assignments for the instruments on the bus are:

Address	LU	Device
1 and 2	7,6	7245 Plotter/printer
3	23	9514 Counter/timer
5	25	316 Signal conditioner 1
6	26	8165 Programmable signal generator 1
7	27	3455 Digital Voltmeter 1
8	28	416 Filter/amplifier 1
9	29	3495 Scanner 1
10	30	516 Offset/amplifier 1
11	31	316 Signal conditioner 2
13	33	3455 Digital Voltmeter 2
14	34	416 Filter/amplifier 2
15	35	3495 Scanner 2
16	36	516 Offset/amplifier 2

APPENDIX H - COMPUTER OPERATING SYSTEM DETAILS

H.1 SELECT CODES, EQUIPMENT NUMBERS, SUBCHANNELS AND LOGICAL UNITS

Each interface card installed in an I/O slot of the computer is given an address. The address is called the select code (SC) for the card. The select codes are normally expressed in octal format. Select codes 0-7 are used internally by the computer while 10-25 are available for I/O slots.

The operating system allows an equipment number (EQT) to be assigned to each interface card. The equipment number is not related to the select code. The system disc is always EQT 1 and the system terminal is always EQT 2. Other EQTs may be assigned sequentially to other I/O slots.

If an interface has more than one device it controls, then each device must have a subchannel number assigned. The EQT and the subchannel number allow each device to be uniquely addressed by the system.

To allow software to address different devices, an additional level of addressing called the logical unit (LU) is used. An LU is assigned to each unique address defined by the EQT and subchannel. The LU is used in all FORTRAN programs to address I/O devices. The operating system allows LUs to be reassigned to meet new requirements. However, the assignment of EQTs to select codes cannot be changed without reconfiguring the operating system.

H.2 RTE OPERATOR COMMANDS

AB - Aborts current FMGR operation

BR,prog name - Sets break flag for a program. If program checks this flag, can change execution path.

EQ,eqt - Returns current assignment: SC DV.xx D B 0 avail
0 0 1 down
2 busy
3 DMA wait

GO,prog name - reschedules a progam suspended previously

LU,lu - Returns current assignment: LU #xx = Exx Sxx

LU,lu,eqt,subch - Changes current lu assignment to new values

OF,prog name,1 - Terminates program immediately
8 - Same as 1 but also purges a temporary program

PR,prog name,priority - Changes a programs priority

RU,prog name,p1,...,p5 - Run a program

ST,prog name - Returns status of a program:
priority status next run time

SZ,prog name - Returns size of a program:

last octal address	no of pages	EMA pages	MSEG size

SZ,prog name,new size - Change size of program

TI - Returns system time year, day, hour, min, sec

TM,yyyy,ddd,hh,mm,ss - Set system time to new value

TO,eqt - Returns current time out for device (10msec counts)

TO,eqt,new value - Change time out for device (new value is in 10msec counts)

UP,eqt - Up a downed device

H.3 FILE MANAGER

H.3.1 NAMR Parameter

File manager commands which access a file or device use the namr parameter to identify the correct file or device. The format of the namr parameter is:

namr = File name:security:cartridge:type:file size:record size.
Logical unit number.

With the exception of file name, the remainder of the subparameters may be deleted and the program will supply a default value. The subparameters are:

File name - Up to six printable ASCII characters (plus, minus, colon, comma or blank excluded) where the first character is not a number.

Security - Integer or two ASCII characters where zero is unprotected (default), positive integer is write protected, negative integer is read and write protected.

Cartridge - Integer or two ASCII characters where zero is first available cartridge (default), positive integer is cartridge reference number, negative integer is cartridge logical unit.

File type - Positive integer (default depends on command).

0 nondisc file.

1 fixed length 128-word record.

2 Fixed length user defined length.

3 Variable length record, sequential access, automatic extents.

4 ASCII code and source programs (otherwise like 3).

5 Relocatable binary code (otherwise like 3).

6 Memory-image format (otherwise like 3).

7 Absolute binary (otherwise like 3).

>7 user defined.

File size - Number of 128 word blocks (1 to 16383), positive integer is use specified number, negative integer is allocate remainder of cartridge.

Record size - Number of words per record (1 to 32767) applies to type 2 files only.

H.3.2 FILE MANAGER COMMANDS

FILE MANIPULATION COMMANDS

CR,namr	CREATE FILE namr-FILE TYPE & SIZE MUST BE SPECIFIED.
PU,namr	PURGE FILE namr.
ST,namr1,namr2	STORE namr1 INTO namr2 AFTER CREATING IT.
DU,namr1,namr2	DUMP namr1 INTO EXISTING namr2.
LI,namr,format,L1,L2	LIST namr ON LIST LU. format: D = directory list, S = ASCII list, B = binary list. L1 = start line #. L2 = stop line #. WITH THE D FORMAT, THIS COMMAND MAY BE USED TO CLOSE AN OPEN FILE.
RN,namr1,namr2	RENAME namr1 to namr2. CHANGES NAME ONLY, OPTIONAL PARAMETERS ARE NOT AFFECTED.

UTILITY COMMANDS

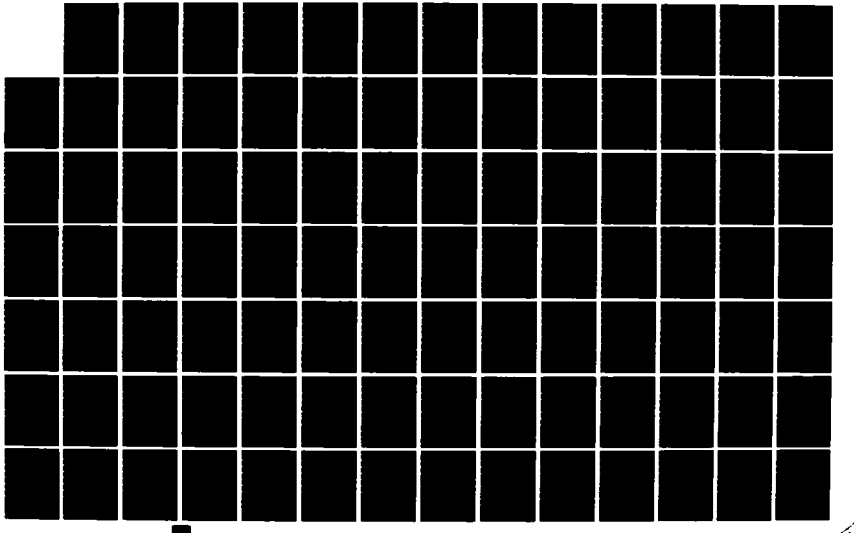
??	EXPLAINS IMMEDIATELY PRECEDING ERROR CODE.
??,#	EXPLAINS ERROR #.
EX	EXIT FMGR.
LL,lu	SET LIST DEVICE TO lu. OUTPUT OF LI, CL, & DL COMMANDS GOES TO THIS LU.
RU,name,p1,...,p5	RUN PROGRAM name AND PASS PARAMETERS p1 - p5.
SYxx...	PERFORMS RTE-IV OPERATING SYSTEM COMMAND xx... .
TR,namr,p1,...,p9	TRANSFER FMGR CONTROL TO FILE namr AND PASS PARAMETERS p1 - p9.

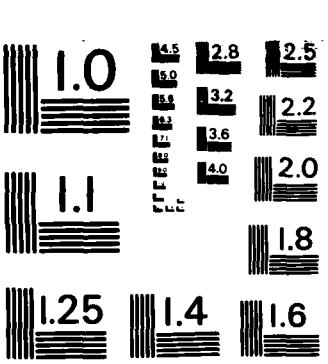
CARTRIDGE MANIPULATION COMMANDS

CL	LISTS ALL MOUNTED CARTIDGES.
DL	LISTS CONTENTS OF ALL CARTRIDGES.
DL,xx	LISTS CONTENTS OF CARTRIDGE xx.
DL,namr	LIST ALL FILES namr ON ANY CARTRIDGE. (USE MINUS SIGN FOR DON'T CARE ON ANY CHARACTER IN namr).
DC,cartridge	DISMOUNT CARTRIDGE; IF MINUS, LU; IF POSITIVE, CARTRIDGE REFERENCE.
MC,lu	MOUNT CARTRIDGE lu.

AD-A139 956 RDI TASK FINAL REPORT OF RESEARCH AND DEVELOPMENT OF SOFTWARE BALLISTIC T..(U) ABERDEEN PROVING GROUND MD MATERIEL TESTING DIRECTORATE C L FRANCIS JAN 84 2/4

UNCLASSIFIED APG-MT-5952 F/G 9/2 NL





H.3.3 FMGR DISK USAGE

LU	CR	LABEL	CONTENTS	COMMENTS
02	02	SYSTEM	SYSTEM	
10	10	CR0010	ADCHK FILES	
11	11	CR0011	MISC FILES	
12	12	CR0012	LIBRARY FILES	NA ON 7906
13	13	CR0013	SYS GEN FILES	NA ON 7906
14	14	CR0014	BTST FILES	
15	15	CR0015	SPARE/RADAR/NA	7920/7925/7906
16	16	CR0016	DOCUMENTATION	
17	17	CR0017	SPARE	
18	18	CR0018	DATA	
19	19	CR0019	DATA	
20	20	CR0020	DATA	
21	21	CR0021	DATA	7925 ONLY

H.4 PROGRAM EDITR COMMANDS

EDITR uses / as the prompt character. The first character following the prompt is used to enter an EDITR command.

DISPLAY COMMANDS

n	Go to line number n of file.
P	Display pending line.
Ln	List n lines starting at pending line.
/	List next line.
+n	Skip forward n lines.
^n	Skip backward n lines.
-n	Delete n lines starting with pending line.
N	Print the current line number.
H	Print the number of characters in the current line.
Xn	Change prompt character to n.
Tnc,...	Set tab character n with tab stops at column c,...

EDIT COMMANDS

text	Enter text as a new line following pending line.
Itext	Insert text before pending line.
Ptext	Edit pending line.
Rtext	Replace pending line with text.
Ctext	Edit pending line and go to next line.

UTILITY COMMANDS

Mnamr	Merge text from file namr following pending line.
F/field	From pending line find field or EOF.
F	From pending line find field previously defined or EOF.
D/field	Delete from pending line to field or EOF.
D	Delete from pending line to field previously defined or EOF.

CONTROL COMMANDS

CTRL I	Insert characters.
CTRL C	Delete characters.
CTRL T	Truncate line.

EXCHANGE COMMANDS

Gold/new	Character exchange on pending line.
Xold/new	Character exchange all lines.

TERMINATE COMMANDS

A	Abort EDITR.
ECnamr	Create file namr to hold text.
ER	Place editted text back in same file.

H.5 PROGRAM DEVELOPMENT STEPS

The steps for creating an entirely new program or modifying an existing program are basically the same. The required steps are:

a. Create or modify the source code for the program or subprogram using program EDITR. All source code file names should begin with an & (i.e., &ADCHK or &TRANS). Note that if an existing source file is modified, the revision code should be updated.

b. The source code must be compiled or assembled into a relocatable module. The format for doing this is:

```
        FTN4
RU,     FTN7X,  source file, list device, relocatable file
        ASMB
        MACRO
```

where the appropriate compiler or assembler program is selected. The list device may be Ø (bit bucket), the terminal LU or the line printer (LU6 is the default). The relocatable file name should begin with % and should normally be the same as the source file name otherwise. If a minus sign (-) is used for the relocatable file name, then the source file name will be used with the & replaced by a %. The compiled or assembled code must be error free (warnings are permissible) before continuing to the next step.

c. The main program module plus all subprogram modules must be linked together and loaded on to the system area of disc by program LOADR. There are three different modes in which LOADR will operate:

(1) Load file mode. A load file containing a list of directives controls LOADR operation. Table H-1 provides a list of LOADR commands and Figure H-1 is an example load file.

TABLE H-1. LOADR COMMANDS

```
OP          - Process opcodes which follow the comma.
              Valid opcodes are:

                RP - Program is a permanent replacement.
                LB - Large background program.
                EB - Extended background program (RTE-6/VM only).
                SS - Use subsystem global area.

EC          - Echo file commands.
SZ,xx       - Set size of program to xx.
AS,nn       - Assign program to partition nn.
SE,namr     - Search file namr for undefined externals.
SER,namr    - Search file namr for undefined externals multiple times.
LI,namr     - Set up file namr as a library file.
RE,namr     - Relocate file namr.
EN          - End of commands.
/A          - Abort LOADR.
```

```
OP,RPLB
RE,%CHKOU::14
RE,%PRHDR::14
RE,%CHLOC::14
RE,%DGIO  ::14
RE,%JDATA::14
RE,%SETUP::14
RE,%TRANS::14
RE,%OTDMA::14
RE,%IASRD::14
RE,%MCVRT::14
RE,%DELAY::14
RE,%NOCHL::14
RE,%WORD  ::14
RE,%RTMNS::14
RE,%TRTIM::14
RE,%GETLU::14
RE,%GSCTR::14
RE,%ERMSG::14
RE,%RBASE::14
EN
```

Figure H-1. Typical load file listing.

Load file names normally start with a $ sign and use the program name for the remainder of the file name. The load file is created using program EDITR. A number of system libraries use the $ as the first character and should not be confused with load files. The load file mode is the preferred method for loading programs. Load files exist for all programs using subprograms written for the BTST. The operator can load a program with this option by typing:

RU,LOADR,$PNAME

where PNAME is the program to be loaded.

(2) Interactive mode. The LOADR will prompt the operator for command entry. This mode is not recommended except for simple programs as it is easy to make errors. This option is exercised by typing

RU,LOADR.

(3) Single module mode. If there are no subprogram modules, a program may be loaded using

RU,LOADR,,relocatable file.

This results in a temporary program load. Simple programs which are used infrequently may be loaded using this method.

No matter which mode is used, LOADR produces an output list which identifies the modules loaded. An examination of this list shows the revision code and data for each module (assuming the programmer has entered this information in the source file). When unexpected problems are encountered in program execution, the LOADR output list should be examined to determine that the correct version of the modules are being relocated.

It should be noted that if option (3), the single module mode, or if the RP command is omitted for options (1) or (2), then the program load is temporary. That is, the program name is stored in memory, not on disc. If the system is rebooted or power is lost, then the program will no longer be available. All of the important BTST programs are permanently loaded.

There are two additional LOADR commands which the operator should be aware of: the list all loaded programs command LI and the purge permanently loaded program command PU. These commands may be activated through interactive mode (2) or by typing:

RU,LOADR,,,,$\begin{matrix}LI\\PU\end{matrix}$.

The LOADR does not delete the old version of a program until the new version successfully loads. This means that unless the operator watches the terminal to determine that the load was successful, there is no way of knowing if a program on disc is the old version or the new version. A second consequence of this LOADR characteristic is that the system area on disc can be broken up into small unuseable segments. When this happens the LOADR will halt with a waiting for disc space message. The PU command must be exercised to make room for the new load operation.

H.6 PROGRAM WHZAT

When program execution problems are encountered, WHZAT provides a capability to determine where the problem is. A careful examination of the WHZAT output listing allows the operator to find the current status of a program.

Figure H-2 is a listing of the output provided by program WHZAT in the program status mode. The top line and bottom line are the current system time when the line is processed in hour, minute, second, and millisecond format. The header abbreviations used are:

PT = Partition number.
SZ = Program size.
PRGRM = Program name.
T = Program type.
PRIOR = Program priority.
DRMT = Dormant.
SCHD = Scheduled.
I/O = I/O wait.
WAIT = Program wait.
MEMY = Memory wait.
DISC = Disc space wait.
OPER = Operator input wait.
NEXT TIME = If time scheduled, next time program is scheduled.

Following the heading is a list of all programs currently on the system schedule list. The program status falls into the category under where the first number appears after the priority entry. For example, in Figure H-2 programs WHZAT and PING are currently scheduled since they have a 1 in the SCHD column, programs FMGR and ADCHK are waiting on other programs to complete since they have a 3 in the WAIT column and program OVRLD is in an I/O wait on EQT 6 since it has 2 in the I/O column.

After the program listing is a listing of down LUs and down EQTs. Down devices may cause a program to go into an I/O wait until the problem with the device is cleared up.

```
*RU,WHZAT
 7:16:13:170
*****************************************************************************
PT SZ PRGRM,T ,PRIOR*DRMT*SCHD*I/O *WAIT*MEMY*DISC*OPER * NEXT TIME  *
*****************************************************************************
 6  3 WHZAT*3 *00001 ***** 1
 3 22 FMGR *3 *00099 ********************* 3,ADCHK
 2 27 ADCHK*4 *00099 ********************* 3,OVRLD
 5  5 OVRLD*3 *00099 ********** 2,  6(2[000000000])
 7  2 PING *3 *32767 ***** 1
*****************************************************************************
DOWN LU'S
*****************************************************************************
DOWN EQT'S
*****************************************************************************
 7:16:13:190
```

Figure H-2. Listing of program WHZAT program status output.

Occasionally it is necessary to determine the memory partition arrangement or the status of a partition. This can be accomplished using a WHZAT option. If any number other than zero is entered as the second parameter in the run string, then the partition status mode is exercised.

Figure H-3 is a listing of the output provided by program WHZAT in the memory status mode. The top line and bottom line are the current system time when the line is processed in hour, minute, second, and millisecond format. The header abbreviations used are:

PTN# = Partition number.
SIZE = Partition size in k words.
PAGES = Memory page location.
BG/RT = Background or real time partition.
PRGRM = Program currently assigned to partition.

```
*RU,WHZAT,,1
 7:17:50:400
*************************************************************************
PTN#   SIZE     PAGES    BG/RT PRGRM
*************************************************************************
 1M      90     38- 127 BG     <NONE>
 2S      29     38-  66 BG     ADCHK
 3S      22     67-  88 BG     FMGR
 4S      21     89- 109 BG     <NONE>
 5S       6    110- 115 BG     <NONE>
 6S       4    116- 119 BG     WHZAT
 7S       2    120- 121 BG     PING
 8S       2    122- 123 BG     <NONE>
 9S       2    124- 125 BG     <NONE>
10S       2    126- 127 BG     <NONE>
11   <UNDEFINED>
12   <UNDEFINED>
*************************************************************************
 7:17:50:430
```

Figure H-3. Listing of program WHZAT partition status output.

H.7 PROGRAM LGTAT

The operating system and programs LOADR and EDITR make use of tracks in the system area of LU2. In particular, LOADR stores all loaded programs in this area. Before loading a program, LOADR checks if a contiguous track block large enough to hold the program exists. If not, the LOADR goes into a wait for disc space. When this happens, it is necessary to purge an existing program to make room for the new program. If a replacement of an existing program is taking place, then the existing program can be purged. Otherwise, some other program must be purged. Program LGTAT allows an operator to find the size of the largest contiguous track block prior to loading a program.

Figure H-4 is the output listing for program LGTAT. Although there are 42 available tracks in the system area, there are only four contiguous tracks available. From one to six contiguous tracks are required depending on program size. In this case, it may be necessary to purge an existing program prior to executing LOADR.

```
:RU,LGTAT
TOTAL AVAILABLE TRACKS =     42
LARGEST CONTIGUOUS TRACK BLOCK =      4
```

Figure H-4. Listing of program LGTAT largest track block mode output.

The operating system makes use of the system track area to swap programs from memory. If more programs are scheduled than will fit in memory, then a lower priority program currently executing is copied to disc to make memory available for the higher priority program. When the higher priority program is finished, the lower priority program is copied back to memory to continue execution. To perform this swap, sufficient disc space must be available to copy the program from memory. If not, the operating system becomes bogged down and, from a practical standpoint, the computer is dead. The solution to this problem is the same as for program LOADR: purge a program to create room for the swap.

To obtain a map showing the tracks used by a program and the location of blank tracks, an LGTAT option may be used. If any number other than zero is entered as the second parameter in the run string, then the track map mode is exercised.

Figure H-5 is a listing of the output provided by program LGTAT in the track map mode. The following abbreviations are used:

FMP = File manager tracks.
D.RTR = File manager directory area.
& = Loaded program storage tracks.
^ = Swapped program storage tracks.
-- = Available track.

In this case tracks 0-49 are the system, library, and programs loaded at system generation time. Tracks 50-230 are filled with operator loaded programs. Tracks 231-234 are the largest contiguous track block. Tracks 235-255 are available for file manager use. To create a larger contiguous track block requires purging a program or several programs. The track map allows this to be done while taking advantage of available tracks. Occasionally, the best thing to do is to purge and then reload all programs. This will result in a largest contiguous track block of 42.

```
:RU,LGTAT,,1
TRACK ASSIGNMENT TABLE          & =PROG ^ =SWAP

TRACK   0       1       2       3       4       5       6       7       8       9
    0 SYSTEM  SYSTEM  SYSTEM  SYSTEM  SYSTEM  ,,,,,&  ,,,,,&  FTN4 &  FTN4 &  ASMB &
   10 ASMB &  XREF &  XREF &  WHZAT&  FMGR1&  FMGR3&  FMGR5&  RT4GN&  RT4GN&  RT4GN&
   20 RT4G1&  SWTCH&  SWTCH&  SWTCH&  JSAVE&  JSAVE&  JRSTR&  LOADR&  LOADR&  LOADR&
   30 REMAT&  QCLM &  EXECM&  RFAM &  LIBRY   LIBRY   LIBRY   LIBRY   LIBRY   LIBRY
   40 LIBRY   LIBRY   LIBRY   LIBRY   LIBRY   LIBRY   LIBRY    ENTS    ENTS   SYSTEM
   50 GAGEF&  GAGEF&    --    SOSI &  PING &  PING &  CDIFF&  CDIFF&  CDIFF&  EDITR
   60    --   CK316&  CK316&  BPLOT&  BPLOT&  BPLOT&  BPLOT&    --    APTST&  APTST&
   70 APTST&  APTST&    --      --    BLKTR&  BLKTR&  FILTR&  FILTR&  DISCS&  TAPER&
   80    --   RREAD&  OVRLD&  OVRLD&  BLAST&  BLAST&  FNDTM&  FNDTM&  CPLOT&  CPLOT&
   90 CPLOT&  CPLOT&  CPLOT&  FIFFT&  FIFFT&    --    APART&  APART&  APART&  APART&
  100 APART&  APPAT&  APPAT&    --      --    IBREM&    --      --    SINWV&  SINWV&
  110 SINWV&  STEST&  STEST&  CHF  &  CHF  &  PG316&  PG316&  PG416&  CATIM&  CATIM&
  120 CATIM&  IBABT&    --      --    FUDGE&  FUDGE&  TRIGR&  TRIGR&    --    FGETR&
  130 FGETR&    --    CK416&    --      --    ODLST&  LUMAP&    --      --    CALX &
  140 CHKOT&  CHKOT&  CHKOT&  CHKOT&  CHKOT&    --    CHFRM&  CHFRM&  CHFRM&  CHFRM&
  150 CA316&  CA316&    --      --    CLASS&  CLASS&  GENAR&  GENAR&  GENAR&  DRLST&
  160    --     --    PRMDS&  PRMDS&  QPEAK&  QPEAK&    --    PURGE&    --      --
  170 TESTO&    --      --    HEADR&  HEADR&  SHIFT&    --    SPARM&  SPARM&  SPARM&
  180 CADIG&  CADIG&  CADIG&  CA416&  CA416&  ADCHK&  ADCHK&  ADCHK&  ADCHK&  ADCHK&
  190 ADCHK&  TEST &  TEST &  DPLOT&  DPLOT&  DPLOT&    --      --    TE416&  TE416&
  200 CA516&    --    IBSTS&  TRGRT&  TRGRT&  TRGRT&    --      --      --    BDCHK&
  210 BDCHK&  BDCHK&  BDCHK&  BDCHK&  APLOT&  APLOT&  APLOT&  APLOT&  APLOT&  CHECK&
  220 RPARM&  RPARM&  RPARM&  MCCHK&  RMS  &    --      --    OREAD&  OREAD&  TRYEQ&
  230 TRYEQ&    --      --      --      --    FMP     FMP     FMP     FMP     FMP
  240  FMP     FMP     FMP     FMP     FMP     FMP     FMP     FMP     FMP     FMP
  250  FMP     FMP     FMP     FMP     FMP    D.RTR

THE LS TRACK(S) ARE UNDEFINED
TOTAL AVAILABLE TRACKS =     42
LARGEST CONTIGUOUS TRACK BLOCK =      4
```

Figure H-5. Listing of program LGTAT track map mode output.

H.8 WELCOM FILE

A wide variety of tasks can be accomplished with FMGR and operating system commands. The WELCOM file provides an opportunity to initialize the system immediately following boot to a desired configuration.

Figure H-6 is a listing of a typical WELCOM file. Lines 1 and 17 set the severity code to make the messages in lines 2-6 and 18-24 display properly. Lines 7 and 25 set the severity code to warn the operator if errors are encountered. Line 8 sets the logical list device to the terminal. Line 9 stores a set special function keys to the terminal. Line 10 runs program PING. Lines 11 and 12 reassign two LUs. Lines 13 - 15 create an additional copy of FMGR for a two operator system. Line 16 turns on the auxiliary terminal. Line 24 documents the system generation answer file and creation date.

```
01      :SV,4
02      :TE,**************************************************************
03      :TE,*****                                                    *****
04      :TE,***** NODE 900            NODE 900            NODE 900 *****
05      :TE,*****                                                    *****
06      :TE,**************************************************************
07      :SV,0
08      :LL,1
09      :ST,KEYS,1
10      :SYRU,PING
11      :SYLU,30,7,1
12      :SYLU,32,6,0
13      :RN,FMGR::2,FMG09
14      :RP,FMG09
15      :RN,FMG09::2,FMGR
16      :CN,9,20B
17      :SV,4
18      :TE,Δ**************************************************************Δ
19      :TE,Δ*****                                                    *****Δ
20      :TE,Δ***** NODE 900                                  NODE 900 *****Δ
21      :TE,*                  BALLISTIC TEST SITE TERMINAL                *
22      :TE,Δ*****                                                    *****Δ
23      :TE,Δ*****             ANSWER FILE BTS6AN 28JUL82             *****Δ
24      :TE,Δ**************************************************************Δ
25      :SV,0
26      :TR
```

Figure H-6. WELCOM file listing.

H.9 SYSTEM BACKUP

H.9.1 OFF LINE DISC BACKUP & RESTORE PROCEDURE (7906 & 7920)

```
INSERT !DSKUP CARTRIDGE IN LEFT CASSETTE DRIVE.  IF AN H-P LABELLED
TAPE FIND FILE 2.  IF AN APG LABELLED TAPE DO NOTHING. HALT COMPUTER.
   SELECT S REG.  SET BIT 14 & SELECT CODE OF TERMINAL IN BITS 6-11.
   STORE.  PRESET.  IBL/TEST.  RUN.  CASSETTE SHOULD BE READ.
   COMPUTER SHOULD HALT & DISPLAY 102077 (IF NOT START OVER).
   SELECT P REGISTER.  SET BIT 1.  STORE.
   SELECT S REGISTER.  SET TERMINAL SELECT CODE IN BITS 0-5.  STORE.
   PRESET.  DO NOT IBL/TEST.  RUN.  DSKUP SHOULD BE RUNNING NOW.

DISC BACKUP UTILITY
TASK?


   ******** UNIT SAVE OF SYSTEM DISC ********

        NOTE:  THIS PROCEDURE IS REQUIRED ONLY IF THE OPERATING
               SYSTEM IS CHANGED.

     [MOUNT & LOAD A TAPE ON DRIVE.  MUST HAVE A WRITE RING.]

TASK?
SA
MAG TAPE CHANNEL?
mag tape select code
SOURCE DISC CHANNEL?
disc select code
SOURCE DISC TYPE?
7920 or 7906
SOURCE DISC DRIVE #?
0
TYPE OF SAVE?
UN
RTE OR DOS?
RTE
WANT TRACK SPARING?
YE
ENTER TRACK MAP INFO FOR SOURCE DISC UNIT AS SHOWN BELOW
#TRKS, FIRST CYL #, HEAD #, # SURFACES
SUBCHNL 00?
enter track map from sys gen listing (typically 256,0,0,2)
SUBCHNL 01?
/E
6144 BUFFER DESIRED?
YE
VERIFY?
YE
FILE ID?
!dskup ddmmyy btst xxx system unit save track map initials
MT FILE#?
1



    [DSKUP WILL COPY FROM DISC TO TAPE. CURRENT TRACK WILL]
    [BE DISPLAYED ON SWITCH REGISTER. WHEN DONE REWINDS]
VERIFYING
    [COMPARES TAPE TO DISC TO CHECK THAT TAPE IS OK.]
    [WHEN DONE REWINDS]
TASK COMPLETED
TASK?
    [DISMOUNT TAPE & LABEL AS UNIT SAVE, DATE & TRLR #.]
    [REMOVE WRITE RING FROM TAPE & STORE TAPE.]
    [REBOOT COMPUTER.]
```

******** UNIT RESTORE OF SYSTEM DISC ********

NOTE: THIS PROCEDURE IS REQUIRED ONLY IF A DISC FAILURE OCCURS OR IF THE SYSTEM WONT BOOT.

[MOUNT & LOAD TAPE ON DRIVE]

```
TASK?
RE
MAG TAPE CHANNEL#?
enter select code of mag tape
MT FILE #?
1
   [DSKUP LOCATES FILE REQUESTED AND READS HEADER]
FILE ID:
<<<<<<<<< HEADER FROM TAPE FILE >>>>>>>>>
TAPE#:01
OK?
ye or no
   [IF NO, DSKUP THEN ASKS FOR NEW MT FILE #]
   [IF YE DSKUP CONTINUES]
DEST DISC CHANNEL#?
enter select code of disc
DEST DISC DRIVE#?
0
VERIFY?
YE
   [DSKUP COPIES TAPE TO DISC.  SWITCH REG DISPLAYS CURRENT TRACK.]
   [TAPE REWINDS.]
VERIFYING
   [DSKUP COMPARES TAPE TO DISC.  TAPE REWINDS.
TASK COMPLETED
TASK?
   [DISMOUNT TAPE & STORE.  REBOOT COMPUTER.]
```

H.9.2 OFF LINE DISC BACKUP & RESTORE PROCEDURE (7925)

******** UNIT SAVE OF SYSTEM DISC ********

NOTE: THIS PROCEDURE IS REQUIRED ONLY IF THE OPERATING SYSTEM IS CHANGED.

[MOUNT & LOAD A TAPE ON DRIVE. MUST HAVE A WRITE RING.]

[PSAVE IS AN ON-LINE UTILITY AND DOES NOT REQUIRE LOADING] [FROM CASSETTE TAPE.]

[SET SYSTEM TIME AS PSAVE RECORDS THIS ON TAPE.]

```
:SYTM,yyyy,dddd,hh,mm,ss
```

```
:RU,PSAVE,,2,8,1,UNVE,,40 character title
```

[PSAVE WILL COPY FROM DISC TO TAPE. WHEN DONE REWINDS] [AND COMPARES TAPE TO DISC TO CHECK THAT TAPE IS OK.] [WRITES HEADER TO TERMINAL.]

```
PSAVE NORMAL END OF JOB
```

[DISMOUNT TAPE & LABEL AS UNIT SAVE, DATE & TRLR #.] [REMOVE WRITE RING FROM TAPE & STORE TAPE.]

******** UNIT RESTORE OF SYSTEM DISC ********

NOTE: THIS PROCEDURE IS REQUIRED ONLY IF A DISC FAILURE OCCURS OR IF THE SYSTEM WONT BOOT.

[MOUNT & LOAD TAPE ON DRIVE]

[FOLLOW THE PROCEDURE IN THE UTILITY PROGRAMS MANUAL FOR] [LOADING THE OFF-LINE UTILITY PACKAGE(!BKOF1, !BKOF2, !BKOF3).] [OBSERVE THE FOLLOWING PRECAUTIONS:] [BE CAREFUL, MOST MISTAKES MEAN STARTING OVER.] [WHEN DOING THE RECONFIGURE BOOT, USE A DISC SELECT CODE OF] [ZERO.] [RECONFIGURE TBG, MAG TAPE(BOTH SELECT CODES) AND DISC(USE] [EQT 3 ENTRY-79XX(MAC) DISC).] [NOTE THAT DISC IS NOT NORMAL LU 2.]

```
TASK?
RE,,13,8,1,UNVE
```

[PRSTR COPIES TAPE TO DISC, REWINDS THE TAPE AND VERIFIES THE] [DISC COPY.]

```
PRSTR NORMAL END OF JOB

TASK?
```

[DISMOUNT TAPE & STORE. BOOT COMPUTER.]

H.10 FILE MANAGER BACKUP

H.10.1 JSAVE AND JRSTR PROCEDURES FOR RTE-IVA SYSTEMS

```
******** JSAVE COPIES A FMGR CARTRIDGE TO TAPE ********
   [MOUNT & LOAD A TAPE ON DRIVE - MAKE SURE IT HAS WRITE RING]
:RU,JSAVE
/JSAVE: REV 6-08-77
/JSAVE: MAG TAPE LU: mag tape lu
/JSAVE: DISC LU: <[,LAST TRACK]   LU= 0 =>END>:  disc lu
/JSAVE: MAG TAPE FILE: if new tape = 1 - if old tape = desired file
   [IF NOT FILE 1, JSAVE WILL FORWARD FILE TO CORRECT POINT]
/JSAVE: HEADER: jsave lu xx ddmmyy btst xxx file type initials
   [JSAVE WILL NOW COPY LU TO TAPE]
/JSAVE: DISC LU: enter next disc lu to be saved
      .
      .
/JSAVE: DISC LU: 0
   [JSAVE REWINDS TAPE]
/JSAVE: DONE!
   [DISMOUNT & LABEL TAPE WITH FILE NO & LU]

******** JRSTR RESTORES A FMGR CARTRIDGE FROM TAPE ********
   [MOUNT AND LOAD THE DESIRED TAPE ON DRIVE]
:RU,JRSTR
/JRSTR: REV 6/08/77
/JRSTR: MAG TAPE LU: mag tape lu
/JRSTR: MAG TAPE FILE: <0<=DIRECTORY, 0 = END>: file no of lu to restore
   [JRSTR FORWARD FILES TO CORRECT FILE IF NECESSARY]
   [JRSTR THEN DISPLAYS HEADER READ FROM THAT FILE FOLLOWED BY ?]
   [IF CORRECT FILE HEADER TYPE YE - IF NOT TYPE NO]
<<<<<<<<<<<HEADER FROM TAPE>>>>>>>>>>>     ? ye or no
   [IF NO, JRSTR ASKS WHICH FILE AGAIN]
   [IF YE, JRSTR CONTINUES]
/JRSTR: DISC LU: disc lu to be restored
   [JRSTR TRANSFERS FROM TAPE TO DISC]
/JRSTR: MAG TAPE FILE: enter next file to be read
      .
      .
/JRSTR: MAG TAPE FILE: 0
   [JRSTR REWINDS TAPE]
/JRSTR: DONE!
   [DISMOUNT AND STORE TAPE]
```

H.10.2 WRITT AND READT PROCEDURES FOR RTE-IVB & RTE-6/VM SYSTEMS

******** WRITT COPIES A FMGR CARTRIDGE TO TAPE ********

[MOUNT & LOAD A TAPE ON DRIVE - MAKE SURE IT HAS WRITE RING]

[WRITT DOES NOT INTERNALLY POSITION THE TAPE, THE OPERATOR]
[MUST DO THIS. IF THERE ARE OLD SAVES ON THE TAPE THAT ARE]
[TO BE SAVED, THEN FMGR :CN,8,FF COMMANDS MUST BE USED TO]
[LOCATE THE DESIRED POSITION ON TAPE FOR THE NEW SAVES.]

[SET SYSTEM TIME AS WRITT RECORDS THIS ON TAPE]

```
:SYTM,yyyy,ddd,hh,mm,ss
```

```
:RU,WRITT,cartridge number to save,,IH
```

[PRIOR TO WRITING TO TAPE WRITT READS THE TAPE. IF IT FINDS]
[DATA ON TAPE, IT DISPLAYS A PORTION AND ASKS THE OPERATOR IF]
[IT IS OK TO PROCEED. IF NOT, WRITT STOPS. IF YES, WRITT]
[COPIES CARTRIDGE TO TAPE.]

```
/WRITT: STOP
```

[IF ADDITIONAL CARTRIDGES ARE TO BE SAVED REPEAT PROCEDURE.]

[DISMOUNT & LABEL TAPE WITH FILE NO & LU]

******** READT RESTORES A FMGR CARTRIDGE FROM TAPE ********

[MOUNT AND LOAD THE DESIRED TAPE ON DRIVE]

[TAPE MUST BE POSITIONED BY OPERATOR TO DESIRED LOCATION BY]
[USE OF FMGR :CN,8,FF COMMANDS.]

```
:RU,READT,cartridge number to restore,,G,,IH
```

[READT THEN DISPLAYS AN OVERLAY WARNING MESSAGE. IF OK TO]
[PROCEED, TYPE YES AND READT WILL COPY TAPE TO DISC. IF]
[NOT OK, TYPE NO AND READT WILL STOP.]

```
/READT: STOP
```

[IF ADDITIONAL CARTRIDGES ARE TO BE RESTORED, REPEAT PROCEDURE.]

[DISMOUNT AND STORE TAPE.]

(Page H-20 Blank)

APPENDIX I - PROGRAMMING STANDARDS AND GUIDELINES

1. The flexibility inherent in many higher level languages provides a programmer with the means to construct programs, routines, and algorithms consisting of many styles and forms. Although these may satisfy the problem at hand they invariably reflect the personality of the designer, when such routines must be integrated into an existing software library for use and maintenence by personnel other than the author problems arise. Therefore the following policy, guidelines, and standards are set forth.

2. In order to facilitate algorithm design, to increase maintainability, and to reduce the development time, Top-Down-Structure should be implemented.

3. The following guidelines should be strictly adhered to:

    a. Prior to the creation of a new routine or algorithm, existing libraries should be searched for desired functions. This proceedure will reduce replication of effort.

    b. Routines should be designed in a modular fashion. In addition software should be designed in a cosmopolitan manner, ie thought should go beyond the specific application so as possibly to end up with a multi-use routine.

    c. In house programming standards must be utilized.

4. The following constitute the minimum programming standards for higher level languages:

    a. Programs and separately compiled routines, modules, or functions must each contain a header block which delineates revision level, date, author, pertinent history, and a brief description of function.

    b. Programs must provide a description and definition of input/ output variables.

    c. Separately compiled routines and functions must provide a definition of all parameters and/or common variables used.

    d. Variable declaration will be primarily ordered by byte size (descending) and further arranged in alphabetical order.

e. Names and/or structures of required files must be listed in a program or routine's header block.

f. Utilization of the "GO TO" statement should be restricted implementing the "IF....THEN ELSE" construction, wherever possible.

g. Use of the "arithmetic if" construction is not recommended.

h. Executable statement numbers should begin at 100 and increase monotonicly over the extent of a program or subroutine. A uniform increment of 20 is recommended. All statement numbers should be right justified to column 5.

i. I/O format numbers should conform to the following convention:

1. Input statement format numbers = 8XXX
2. Output statement format numbers = 9XXX

j. Judiciously placed comment statements should be included as means of built in documentation.

k. Data typing should be restricted to Fortran's default values whenever possible (I--N integer, everything else real)

l. Subroutines should not be appended to the driver routine or main program.

5. Assembly language standards are as follows:

a. Programs and separately assembled routines, modules, or functions must each contain a header block which delineates revision level, date, author, pertinent history, and a brief description of function.

b. Programs must provide a description and definition of input/ output variables.

c. Code must be arranged in columns, with one column each constituting the label field, the op-code field, the operand field, and the comment field. (No rule concerning the beginning position of these columns is established.) Comments making up a header block or separating blocks of code may deviate from this rule.

d. Nearly every statement will be accompanied by a comment. Exceptions such as STC CLEAR CARRY BIT CMC are allowable.

e. Sections of code which are logically separate because of function will be separated. Logically distinct blocks of code will be preceded by comments which describe the function performed.

f. Subroutines and macros will be proceeded by a header block which describes the function of the routine. Variables or registers used by or altered by the routine will be listed. Any requirements of or alteration to the stack or stack pointer will be indicated.

g. Labels which are descriptive in nature will be used.

6. Programs, routines, functions, or other modules not conforming to at a minimum 4.a above will be subject to deletion from the system.

7. These standards apply to all higher level languages and assembly language as appropriate, independent of the machine on which software will be utilized. These standards do not preclude the use of machine dependent features, but are intended to aid in the design of readable, structured routines and programs which are documented to the extent that they are capable of being shared by a community of users.

(Page I-4 Blank)

APPENDIX J - ADCHK DATA DISC AND FILE FORMATS

The BTST discs are organized such that operational parameters, directories, etc. are stored on the first two tracks of a data lu(tracks 0 and 1). Data records are stored starting at track 2 to the end of disc. The organization of the first two tracks on each data disc is shown below with the format of the various components following that.

TRACK	SECTOR	CONTENTS
0	0-2	Channel 0 setup parameters
	3-5	Channel 1 setup parameters
	.	
	.	
	.	
	93-95	Channel 31 setup parameters
1	0-3	Directory tracks
	4-5	Not used
	6	Header
	7	Not used
	9-12	Rate table storage
	13-95	Not used

CHANNEL PARAMETER FORMAT:

FILE WORD	PARAM#	DESCRIPTION	TYPE DATA	
1	1	SELECTED	A	
2	2	MASTER	A	
3	3	SLAVE	A	
4	4	XTRGR	A	
5	5	THRSHLD V	I	10*THRESHOLD
6	6	SLOPE	B	
7	7	RUN DLY	I	
8	8	STP DLY	I	
9	9	PRETRGR	I	
10	10	FXD RATE	A	
11	11	MEM SIZE	I	
12	12	TYPE DATA	A	
13	13	MODE	A	
14	14	BALANCE	A	
15	15	FREQ(KHZ)	I	10*FREQ
16	16	RATE T/C	I	
17-18	17	GAGE FCTR	R	
19-20	18	EXCT/INTR	R	
21-22	19	PEAK SGNL	R	
23-25	20	GF UNITS	B	6 characters
26-27	21	GAIN	R	
28-30	22	XDCR MFG	B	6 characters
31-33	23	XDCR MDL#	B	6 characters
34-36	24	XDCR SN	B	6 characters
37-53	25	PLOT LABEL	B	34 characters
54-68	26	Y-AXIS LBL	B	30 characters
69-102	27	REMARKS	B	68 characters
103-127		NOT USED		
128		CHANNEL NO	I	
129-192		NOT USED		

DATA TYPES ARE:
A = ONE ASCII CHARACTER RIGHT JUSTIFIED(MSB=0)
B = TWO ASCII CHARACTERS PER WORD
I = INTEGER
R = REAL

DIRECTORY FORMAT:

SECTOR	WORDS	USE
0	1	START TRACK OF LAST FILE IN DIRECTORY
	2	NO. OF DATA FILES
	3	START TRACK OF FILE 1
	4	NO. OF TRACKS FOR FILE 1
	.	
	.	
	.	
	63	START TRACK OF FILE 31
	64	NO. OF TRACKS FOR FILE 41
N	1	START TRACK OF FILE N*32
(N=1,2,3)	2	NO. OF TRACKS FOR FILE N*32
	.	
	.	
	.	
	63	START TRACK OF FILE N*32+31
	64	NO. OF TRACKS FOR FILE N*32+31

HEADER FORMAT:

WORDS	ITEM	DESCRIPTION
1-4		RESERVED FOR OTHER USES
5-6	1	ROUND NUMBER
7-11	2	TEST DIRECTOR
12-13	3	SCHEDULE NUMBER
14-16	4	WEAPON TYPE
17-19	5	TUBE NUMBER
20-21	6	TRAILER NUMBER
22-23	7	OPERATORS INITIALS
24-27	8	DATE(DDMMMYY)
28-64	9	REMARKS(74 ASCII CHARACTERS)

All items are stored in ASCII format.

RATE TABLE FORMAT:

SECTOR	WORDS	RATE TABLE
9	1-12	0
	13-24	1
	25-36	2
	37-48	3
	49-60	4
	61-64	Not used
10	1-12	5
	13-24	6
	25-36	7
	37-48	8
	49-60	9
	61-64	Not used
11	1-12	10
	13-24	11
	25-36	12
	37-48	13
	49-60	14
	61-64	Not used
12	1-12	15
	13-24	16
	25-36	17
	37-48	18
	49-60	19
	61-64	Not used

Each table has the following format:

WORD	CONTENTS
1	No of rates in table
2	Scalar
3	Length of variable length shift register
4	Decrease delay before rate is reduced
5	MSB: Rate 1 LSB: Start Address
.	.
.	.
.	.
12	MSB: Rate 8 LSB: Start Address

CALIBRATION FILE @CALZZ

@CALZZ is a fixed length file used to store the calibration factors of equipment used in the BTST. @CALZZ provides room for 32 channels of calibration information with each channel using 64 words of storage. @CALZZ is located starting at track 0, sector 0 of LU 14. Thus, the channel number is the sector address for a given channel. Serial numbers and calibration dates are stored in integer format in the file while real format is used for the calibration factors themselves. The organization of the sector is:

WORDS	CONTENTS
1	ASRD serial number
2	ASRD A/D calibration date
3-4	ASRD A/D calibration factor
5	316 serial number
6	316 calibration date
7-8	316 voltage calibration factor
9-10	316 charge calibration factor
11-12	316 strain calibration factor
13-14	316 strain x10 calibration factor
15-16	316 strain x100 calibration factor
17-18	316 strain x1000 calibration factor
19	416 serial number
20	416 calibration date
21-22	416 gain x1 calibration factor
23-24	416 gain x2 calibration factor
25-26	416 gain x4 calibration factor
27-28	416 gain x5 calibration factor
29-30	416 gain x10 calibration factor
31-32	416 gain x20 calibration factor
33-34	416 gain x25 calibration factor
35-36	416 gain x40 calibration factor
37-38	416 gain x50 calibration factor
39-40	416 gain x100 calibration factor
41-42	416 gain x200 calibration factor
43-44	416 gain x250 calibration factor
45-46	416 gain x400 calibration factor
47-48	416 gain x500 calibration factor
49-50	416 gain x1000 calibration factor
51	516 serial number
52	516 calibration date
53-54	516 calibration factor
55-62	Not used
63	ASRD serial number
64	ASRD time calibration date

MENU FILE @MENU

@MENU contains the ADCHK menu in a format compatible with terminal display. @MENU is a fixed length file with room for up to 40 commands. Each command may use up to 32 characters(16 words of storage). Program MKMNU is used to create the file from text file MENU. @MENU is located starting at track 0, sector 0 of LU 10. The format of the file is:

WORDS	CONTENTS
1-16	Command 1
17-32	Command 2
.	.
.	.
.	.
609-624	Command 39
625-640	Command 40

APPENDIX K - ADCHK OPERATORS MANUAL

ADCHK is an interactive command driven control program for operating a Ballistic Test Site Terminal(BTST). The program is composed of a number of subroutines and it also schedules a number of other programs to perform certain tasks. Two letter commands with optional delimiter parameters are used by the operator to initiate all tasks.

The ADCHK commands can be divided into five broad catagories:

1. Setup commands - these commands allow the operator to initialize or prepare equipment parameters and headers prior to data acquistion (HE, PA, PD, PM, RT, SE).
2. Data acquisition/equipment control commands - these commands allow the operator to control the signal conditioning, digitizer, and scanner (AR, CH, DI, HA, LO, OV, TR, VE).
3. Examine data file commands - these commands allow the operator to read various parameters, headers, events, and data from a data file on disc (EV, IN, RA, RC, RH, RN, RP).
4. Modify or annotate data file commands - these commands allow the operator to correct or update headers and labels on a data file on disc (CO, MH, ML, NO).
5. Utility commands - these commands allow the operator to perform a variety of housekeeping functions which help improve the overall efficiency of operation (BU, CK, DL, EX, LC, LI, LU, PU, RE, RU, ST).

ADCHK may be run from any terminal on the system. However, the terminal should be an H-P 2648 type terminal to display the ADCHK subroutines properly. In general, only one version of ADCHK should be running on the system at one time since the program is not designed to share resources.

When ADCHK is first run the following default values are set:
  Data disc lu = 19
  Last channel = value defined in subroutine NOCHL
These values can be examinined and changed using the appropriate commands.

When ADCHK is first run it responds by displaying a list of the available operator commands. The list is:

VALID ADCHK COMMANDS ARE:

AR	ARm ASRD	NO,u,k	NOtes
BU	BackUp data disc	OV,lu	OVerloads on 416
CH	CHannel scan	PA,c	PArameter changes
CK	CheCK ASRD	PD,lu	Parameter Display
CO,u,k,c	COmments	PM	Param. Maintenance
DI,sec,sub	DIsplay ASRD signal	PU	PUrge directory
DL,m,n,lu	Directory List(m-n)	RA,u,k,lu	Read All parameters
EV,u,k,lu	EVents display	RC,u,k,c,lu	Read Comments
EX	EXit ADCHK	RE	REstore from tape
HA	HAlt ASRD	RH,u,k,lu	Read Header
HE	HEader changes	RN,u,k,lu	Read Notes
IN,u,k,lu	INspect data	RP,u,k,lu	Read Parameters
LC	Last Channel changes	RT,n	Rate Table n changes
LI,lu	LIst ADCHK commands	RU,nmr,p1-5	RUn program nmr
LO,lu,skip	LOad ASRD & sig cond	SE	SElect channels
LU	change default LU	ST,lu	Self Test ASRD
MH,u,k	Modify Header	TR,u,k	TRansfer data
ML,u,k,c	Modify Labels	VE,c	VErify rate table

where: u,k = data lu (18-20) & track (2-1022)
lu = list device's logical unit # (1,3-6,8,9)
c = channel number. (0<c<15)

The commands are listed in alphabetical order for easy reference.

The sequence which is best for learning the commands is:

EX, LI, LU, LC, DL, PU, RU, HE, RH, MH, RT, PA, SE, RP,

PD, RA, PM, SP, GP, ML, VE, LO, AR, HA, TR, OV, CH, DI,

EV, IN, NO, RN, CO, RC, ST, CK, BU, RE.

A description of each command is stored in a file on the BTST. The file name is xxCMND::16 where xx is the two letter command.

ADCHK receives all operator commands, decodes the commands, sets parameter values, checks parameter values, and then implements the command. ADCHK implements the command either internally, by calling a subroutine, or by scheduling another program. The breakdown of how various commands are handled is described below.

INTERNAL COMMANDS

ADCHK handles some commands internally without reference to command specific subroutines or programs. These commands are:

EX HA LC LI LU RU

UTILITY & COMMAND SUBROUTINES

ADCHK calls on a number of subroutines to perform its tasks. Some of the subroutines are utility subroutines used by ADCHK and other subroutines. Some of the subroutines are specific to one command or group of related commands only. The ADCHK utility subroutines and the command specific subroutines are:

ARMOT	AR			
ATTNN				
BCKUP	BU			
BITS				
CHSCN	CH			
CKCHL				
CKLST				
DELAY				
DFLT				
DGIO				
DRLST	DL			
EDIT				
ERMSG				
GSCTR				
HEADR	HE	MH	RH	
IASRD				
INSPT	IN			
MCVRT				
MLBL	ML			
NOCHL				
NOTES	CO	NO	RC	RN
OTDMA				
PURGE	PU			
RATBL	RT	VE		
RBASE	TR			
REVNT	EV			
RSTOR	RE			
SCHDL				
SETUP	LO			
TMCVT				
TRANS	TR			

SCHEDULED PROGRAMS

Some ADCHK commands also schedule other programs to perform tasks. In these cases ADCHK schedules the program and then waits for the scheduled program to complete its operations before ADCHK resumes its own operations. The programs scheduled by ADCHK and the scheduling commands are:

DGCHK	CK					
DSPLY	DI					
OVRLD	LO	OV	TR			
PG316	LO					
PG416	LO					
PRMAN	PA	PD	PM	RA	RP	SE
RMTST	ST					

All of the scheduled programs must be loaded on the system, otherwise a schedule error will be returned and the task will not be done.

AR COMMAND

COMMAND PURPOSE: ARm the ASRD.

COMMAND FORMAT: AR

COMMAND DESCRIPTION:
The AR command performs the following tasks:

1. Pulses the ARM line to allow any channel set to relative trigger to measure the ambient value and adjust the trigger threshold.

2. Pulses the hardware reset line to synchronize all rate clocks, clear all variable sample rate accumulators and clear the event FIFOs.

3. Sets the arm line enabling channels to take data and causing the events channel to read and store the status of the eight event lines plus the current time in word zero of the events channel.

The AR command takes about one second to perform these tasks.

The AR command should be early enough before the trigger event so that the prememory shift register has had time to completely flush and the pretrigger memory has had time to fill. If the ASRD is triggered at arm time, unpredictable results may occur.

The AR command would normally follow in sequence an LO command. The arm signal remains active until an HA, LO, TR, or VE command is issued. If an accidental trigger on a channel occurs, the channel must be setup again(LO command) as the AR command does not reset the memory address register to zero.

There are no warning, error, or diagnostic messages associated with the AR command.

EXAMPLE:

```
>AR
>
```

BU COMMAND

COMMAND PURPOSE: BackUp a data disc to tape.

COMMAND FORMAT: BU

COMMAND DESCRIPTION:
The BU command allows a copy of disc data records to be made to tape for backup purposes. The BU command works on the current default data lu. The BU command will copy all data records listed in the data directory. A header indicating the data disc lu, trailer number, round numbers of the first and last rounds in the directory, date of the last round in the directory, test director and the track numbers copied is created. The operator is given an opportunity to add to the header. A copy of the disc data is made to tape, the tape is rewound and verified against the disc data. During verification the directory is reconstructed from tape entries and listed to the terminal to provide assurance to the operator that a valid copy of the disc has been made. A copy of the header is written to file TPRCRD to document the creation of a backup.

The BU command does not perform any tape positioning. Thus, if multiple backups are to be made on the same tape, the operator must manually perform tape positioning.

EXAMPLE:

```
>BU

     BACKUP OF LU 19      TRACKS 1 -  233

HEADER:
BCKUP LU 19 BTST  900 27NOV83 RNDS WRMA -  99* HOLZINGER  TRKS 1 -  233

OPERATOR HEADER(80 CHARACTERS):
HOT ROUNDS

OK TO PROCEED(YE OR NO)?YE

  [Disc is copied to tape, tape is rewound and verified.  During]
  [rewind a copy of directory is displayed to terminal.  Copy of]
  [directory is not displayed here to conserve space.           ]

FILE TPRCRD UPDATED -- BCKUP COMPLETED

>

-----------------------------------------------------------------

>BU

     LU 19 DISC DIRECTORY IS EMPTY.  NO COPY CAN BE MADE.

>
```

CH COMMAND

COMMAND PURPOSE: CHannel observation using scanner.

COMMAND FORMAT: CH

COMMAND DESCRIPTION:
Inorder to check transducers and cables, the H-P 3495A scanner may be used to connect the output of the filter/amplifier to the oscilloscope or digital volt meter. This can be accomplished using the CH command. The subcommands available within the CH command are:

c,d	Sets bank 0 to channel c and bank 20 to destination d. Channel c can have values of 0-19(16-32) where 0-15 (16-31) are the respective channel outputs of the 416 filter/amplifier and 16-19(32-35) are external inputs. The destinations can be A = scope ch A, B = scope ch B, C = external output C, D = digital voltmeter & E = external output E. A value of -1 can also be entered for c. This will cause banks 0 and 20 to be cleared without affecting banks 40 and 60. The choice of the left(0-19) or right(16-35) group of channels is made by the T command. Note that there is some overlap on channel numbers between the two groups. The operator must differentiate between external inputs 16-19 on the left group and channels 16-19 on the right group.
E	Exit subroutine CHSCN leaving the scanner programmed.
F	Exit subroutine CHSCN clearing the scanner(open all contacts).
I	Increment through the scan list set up by the R or S commands. A carriage return increments to the next channel in the list, while any other character will cause termination. If there are no entries in the scan list an error message will be sent to the operator.
M	List the menu of commands for CHSCN.
P,n	Program the set of contacts n where n can be 0-79. This command does not check for valid connections. It is intended primarily for controlling banks 40 and 60 which are not set by any other command in CHSCN.
R,d	Display the scan list and allow changes to be made in the entries. The R command uses three commands to control its operation: a,c change list element a to channel c. a,-1 change list element a to -1. The first -1 in the list repeats the scan. F finish R command processing. The scan list controls the source signal from bank 0. The destination of bank 20 is set to d. This command sets the list up and the I command performs the scan. If an invalid entry is made, an error message will be sent to the operator.
S,d,b,e	Set the scan list to perform a sequential scan on bank 0 from channel b through channel e with destination d. This command sets the list up and the I command performs the scan.
T	Display the current channel group(0-19 or 16-35). [Valid with a 32 channel system only]
T,x	Toggle from the current group setting to the other group setting where x may be any entry. [Valid with a 32 channel system only]

The R and the S command share the same program buffer. Therefore, if one type of scan is programmed, then the other type of scan is programmed, the original scan list is lost and must be entered again. Once a scan list is set up it remains available even if the CH command is exited or finished as long as ADCHK is not exited. There are separate buffers for the left(0-19) and right(16-35) groups used in a 32 channel system.
If an invalid command is entered, an error message is sent to the operator. The display on the scanner will show the value programmed in each bank.

EXAMPLE:

```
>CH
            CHANNEL SCANNING COMMANDS:
      c,d           Filter/Amplifier output
      E             Exit-Scanner stays programmed
      F             Finish-Opens all contacts
      I             Increment thru scan list
      M             Menu
      P,n           Program contacts n (n=0-79)
      R,d           Random scan list
      S,d,b,e       Sequential scan list from ch b to e
      T             Display current toggle group
      T,x           Toggle current channel groups(0-19 <> 16-35)
           where:   c is chan # 0-15: 416, 16-19:ext in, -1 clears
                      or chan #16-31: 416, 32-35:ext in, -1 clears
                    d is the dstntn: A-scope ch A     B-scope ch B
                       C-ext out        D-DVM         E-ext out

CH>8,A
       [Output of channel 8 is sent to scope.]

CH>P,59
       [Contacts 59 are closed.]

CH>I
     THE SCAN LIST HAS NO ENTRIES
       [Scan list must have entries before the I command    ]
       [can be executed.                                    ]

CH>S,A,5,9
       [Set up for sequential scan on scope channel A from  ]
       [channel 5 to channel 9.                             ]

CH>I
CR to increment -- F to finish
       [Scan list is now executed.  Each carriage return    ]
       [will advance to next channel.  F will terminate.    ]
F
```

```
CH>R,A
  SCAN LIST:  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
              5  6  7  8  9 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1

          Entry format: a,c  *where a is list element & c is ch#
                        a,-1 *ist -1 as list element repeates scan
                        F    *Finished
        [Scan list contains entries from previous sequential]
        [scan.  They may now be changed for desired random  ]
        [scan pattern.  When a new value is entered, the    ]
        [current display is erased and the updated values   ]
        [displayed.                                         ]
1,3
        [First scan element is changed.                     ]
2,9
        [Second scan element is changed.                    ]
3,155
INPUT ERROR-TRY AGAIN
        [Invalid entry results in error message.  Cursor    ]
        [moves to old entry and waits for new entry.        ]
3,15
        [Third scan element is changed.                     ]
4,-1
        [Fourth scan element is changed.                    ]
F
        [Scan list processing is finished.                  ]
        [For clarity the final list is repeated.            ]
  SCAN LIST:  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
              3  9 15 -1  9 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1

        [Note that entry 5 remains unchanged, but will not be]
        [read since entry 4 is -1.                          ]

CH>I
CR to increment -- F to finish
        [Scan list is now executed.  Each carriage return   ]
        [will advance to next channel.  F will terminate.   ]
F

CH>T
SCANNER CONTROLS CHANNELS  0-19
        [Scanner controls left group of channels.  No change]
        [is made in group controlled.                       ]

CH>T,1
SCANNER CONTROLS CHANNELS 16-35
        [Scanner control switched to right group of channels.]

CH>8,F
IMPROPER COMMAND FORMAT

CH>E
        [CH command is terminated.  Scanner remains programmed.]

>
```

CK COMMAND

COMMAND PURPOSE: CheCK the ASRD.

COMMAND FORMAT: CK

COMMAND DESCRIPTION:
The CK command schedules program DGCHK which gives the operator the capability to communicate directly with the ASRD channels. A wide variety of tasks can be accomplished with DGCHK to ensure correct functioning of the digitizer. Under normal circumstances the use of the CK command should not be required. However, when problems are encountered, this command provides a means of controlling the digitizer which can not be accomplished in any other way.

The commands supported by the CK command are:

AR	ARM ALL CHANNELS
AV	GET THE CURRENT AVERAGE AND NOISE VALUES
CC,val	SPECIFY THE CALCULATION CONTROL WORD
CH,chan	SPECIFY THE ADDRESSED CHANNEL
CV	GET THE CURRENT SIGNAL VALUE
DA	DMA DATA TRANSFER--AUTO PARAMETERS (*)
DD,val	SPECIFY THE DECREASE DELAY
DS	DMA DATA TRANSFER--SPECIFIED PARAMETERS (**)
DV	DISPLAY SIGNAL VALUE ON LED DISPLAY
EX	EXIT 'CK' MODE
FT	PERFORM FAST MEMORY TEST
GA	GET STARTING ADDRESS FOR AUTO TRANSFER (*)
GE	GET ERROR WORDS
GL	GET MEMORY FAIL LOCATION
GN	GET NUMBER OF WORDS FOR AUTO TRANSFER (*)
GR	GET REGISTER CONTENTS
GT	GET RATE TABLE
HA	HALT DATA ACQUISITION MODE ALL CHANNELS
LC,list	LIST COMMANDS TO LIST
LD	LOAD PARAMETERS TO CHANNEL REGISTERS
MF	FILL MEMORY WITH SEQUENTIAL NUMBERS (**)
MT	EXECUTE MEMORY TEST
OF	TURN CHANNEL OFF WRT TO DATA ACQUISITION
ON	TURN CHANNEL ON WRT TO DATA ACQUISITION
PA	PROG DATA TRANSFER--AUTO PARAMETERS (*)
PS	PROG DATA TRANSFER--SPECIFIED PARAMETERS (**)
PT,val	SET PRETRIGGER SIZE (1 K BLOCKS)
RE	RESET MICROPROCESSOR ALL CHANNELS
RD,val	SET RUN DELAY
RT	REGISTER TESTS
SA,v1,v2	SET STARTING ADDRESS (**)
SD,val	SET STOP DELAY
SN,v1,v2	SET NUMBER OF WORDS TO TRANSFER (**)
ST	SET RATE TABLE
SY	ISSUE SYNC TO ALL CHANNELS
SZ,val	SET MEMORY SIZE (1 K BLOCKS)
S#	GET CHANNEL SERIAL NUMBER AND SW REV CODE
TG	GET TRIGGER SET INDICATOR
TM,val	SET TRIGGER MODE AND THRESHOLD
TR,val	SET TRACE FLAG (0=OFF, 1=ON)
ZE	ZERO ERROR WORDS

******* EVENTS SUBSYSTEM ONLY COMMANDS **********

DA	READ DAY OF YEAR
EV	READ EVENTS
NE	READ NUMBER OF EVENTS
TI	READ CURRENT TIME

A complete description of program DGCHK is contained in the ASRD software manual.

EXAMPLE:

```
>CK

      TO GET A COMMAND LIST ENTER 'LC,list_lu'

    /CK (CHAN **):
```

CO COMMAND

COMMAND PURPOSE: COmments annotation for a data record channel.

COMMAND FORMAT: CO,data disc lu,record start track,channel number

data disc lu	- default = current default lu range = 18-20
record start track	- default = last record in directory range = 2-201 for 18 2-1022 for 19 & 20
channel number	- default = 0 range = 0 to last channel

COMMAND DESCRIPTION:
After a data record is created by the TR command, it may be necessary to annotate the record with additional documentation information concerning a particular channel. The CO command provides the capability to annotate any channel desired. Using the data disc lu and record start track parameters any data record on disc may be annotated. The initial channel desired can be set by the channel number parameter. If an invalid parameter is entered, the command is aborted and an error message is sent to the operator. There is storage space for 128 characters of information. For display purposes, the storage space is divided into two lines of 64 characters each. Entries can be made on either line. However, the normal procedure would be to fill up line one first, then to fill line two. The entries can be edited using the following commands:

/ - leaves existing character unchanged, used same as the EDITR /.

$ - truncates currently displayed characters from $ to end of line.

A | is displayed to mark the end of the available space for character input. If an operator command error is detected, an error message is displayed. If a channel which was not selected is requested, a warning message is displayed.

EXAMPLE:

```
>CO

 CHNL  0  COMMENTS:   FROM LU 19   TRACK  231

  1.
  2.

 ENTER: # to edit, Finish, Kill  >>1
   [Currently no annotation information is stored in the record.]
   [Line 1 is to be edited.  Current contents will be displayed ]
   [immediately below answer left justified for easy entry.     ]
   [New values may be added.                                    ]

CABLE SNAPPED DURING RECOIL.  FIRST 40 MSEC OF RECORD IS
   [New values for line 1 have been entered.  Current display  ]
   [will be erased and new values written in old values place. ]
   [Next command may be entered.                               ]
 ENTER: # to edit, Finish, Kill  >>22
INPUT ERROR - TRY AGAIN
   [Invalid entry results in error message.  Cursor moves to   ]
   [old entry and waits for new entry.                         ]
 ENTER: # to edit, Finish, Kill  >>2

OK.  IGNORE REST.
   [New values for line 2 have been entered.  Current display  ]
   [will be erased and new values written in old values place. ]
   [For clarity display is repeated here.                      ]
```

```
  1.  CABLE SNAPPED DURING RECOIL.  FIRST 40 MSEC OF RECORD IS
  2.  OK.  IGNORE REST.
ENTER: # to edit, Finish, Kill  >>F
  [Editing is finished.  Currently displayed values are saved ]
  [in data record.                                            ]

  ENTER - new chnl #      F<inish> :2
   [Opportunity to go to another channel or finish CO processing.]
  CHANNEL  2 NOT SELECTED

  ENTER - new chnl #      F<inish> :22
   [Invalid channel number will result in repeat of instructions.]

  ENTER - new chnl #      F<inish> :F

>CO,18,231
IMPROPER COMMAND FORMAT**TRY AGAIN    >
   [Invalid parameter entry.  Command aborted.]
```

DI COMMAND

COMMAND PURPOSE: DIsplay input signal.

COMMAND FORMAT: DI,display time,subset switch

display time - default = operator termination
range = 1 to 32767

subset switch - default = all selected channels
range = -32768 to 32767

COMMAND DESCRIPTION:
The DI command displays the ASRD input signals of selected channels on the graphics display. The display is in the form of a bar with the position of the bar representing the average value of the input signal and the height of the bar representing the noise component of the input signal. If the subset switch is not set, then all selected channels will be displayed. If the switch is set, then the operator can enter a subset of the selected channels to display. Since the process of reading the ASRD, formatting the data and displaying a bar is relatively slow, the display is not a substitute for a good scope. However, the DI command provides a convenient way to check the general status of all selected channels quickly. Because of the slow sampling and display, the DI command will not normally capture intermittent or transient type events. Use of the DI command must be terminated prior to arming the digitizer. The amount of time for the display to last(in seconds) before returning to ADCHK is determined by the display time parameter. If this parameter is defaulted or if it is necessary to stop a timed display in progress, then the operator can terminate the display by striking a key until an operating system prompt is obtained and then typing BR,DSPLY. The terminal screen will be restored at the end of a complete scan.

EXAMPLE:

```
>DI,60

   [All currently selected channels are displayed for 60 seconds.]

>DI,,1

ENTER NUMBER OF CHANNELS TO BE DISPLAYED: 2
ENTER CHANNELS TO BE DISPLAYED: 3,7

   [Only channels 3 and 7 will be displayed.  Display will continue]
   [until terminated by opertor.                                    ]

>
```

DL COMMAND

COMMAND PURPOSE: Directory List of the data record directory on a data lu.

COMMAND FORMAT: DL,start record,stop record,list lu

start record - default = 1
range = 1-127

stop record - default = end of directory
range = start record to 127

list lu - default = operator terminal
range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
The DL command reads the directory from the current default data lu and displays it on the list lu. A directory may contain a maximum of 127 entries. The start and stop record parameters may be used to obtain a list of only a portion of the records in a directory. The list may be sent to any valid output device on the system using the list lu parameter. If invalid parameters are entered, the command will be aborted and an error message sent to the operator.

EXAMPLE:

```
>DL
 ***DIRECTORY LIST-LU18, TRK   1
 FILE   STRT   # OF   RND   TEST          SCHD   WEAPON   TRLR   DATE
  NO     TRK   TRKS   NO    DIRECTOR      NO     TYPE     NO     DDMMMYY
   1       2     16   1     KURIATA              155mm    700    14NOV80
   2      18     24   1     CAMMARATA            M-60     700    28JAN82
   3      42     11   1     CHEATER              4.2"     1000   25MAR82
   4      53      1   TEST  EVENTS        TEST   BOX      900    25MAR82
   5      54      1   TEST  0                    PSG      900    7APR82
   6      55      1   TEST  0,8                  PSG      900    20AUG82
   7      56      1   TEST  0,8                  PSG      900    20AUG82

>
```

[Lu 18 is current data lu. Entire contents of directory are listed.]

```
>DL,51,62
 ***DIRECTORY LIST-LU19, TRK   1
 FILE   STRT   # OF   RND   TEST          SCHD   WEAPON   TRLR   DATE
  NO     TRK   TRKS   NO    DIRECTOR      NO     TYPE     NO     DDMMMYY
  51     105      1   TEST  EVENTS        TEST   BOX      900    25MAR82
  52     106      1   TEST  EVENTS        TEST   BOX      900    25MAR82
  53     107      1   TEST  EVENTS        TEST   BOX      900    25MAR82
  54     108      1   TEST  EVENTS        TEST   BOX      900    25MAR82
  55     109      1   TEST  EVENTS        TEST   BOX      900    25MAR82
  56     110      1   TEST  EVENTS        TEST   BOX      900    25MAR82
  57     111     24   1     CAMMARATA            M-60     700    28JAN82
  58     135     24   1     CAMMARATA            M-60     700    28JAN82
  59     159     10   1     CHEATER              4.2"     1000   25MAR82
  60     169      1   TEST  0                    PSG      900    7APR82
  61     170      1   TEST  0                    PSG      900    7APR82
  62     171      1   TEST  0                    PSG      900    7APR82

>
```

[Lu 19 is current data lu. Only requested records are listed.]

```
>DL
 LU 20 DIRECTORY EMPTY

>
```

[Lu 20 was current data lu. Directory is empty.]

```
>DL,99
 ***DIRECTORY LIST-LU19, TRK   1
 FILE   STRT   # OF   RND    TEST        SCHD   WEAPON   TRLR   DATE
  NO     TRK   TRKS   NO     DIRECTOR    NO     TYPE     NO     DDMMMYY

>
```

[There is no record 99 in the directory on lu 19.]

```
>DL,20,10
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

[Invalid parameter entry. Start record is larger than stop record.]

```
>DL,,220
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

[Invalid parameter entry. Stop record is out of range.]

```
>DL,,,10
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

[Invalid parameter entry. Lu 10 is not a valid list device.]

EV COMMAND

COMMAND PURPOSE: EVent channel display from a data record.

COMMAND FORMAT: EV,data disc lu,record start track,list lu

data disc lu	- default = current default lu range = 18-20
record start track	- default = last record in directory range = 2-201 for 18 2-1022 for 19 & 20
list lu	- default = operator terminal range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
When a data record is created, the events channel is automatically read and stored in the data record. The EV command provides the capabiltity to read the events channel data. Using the data disc lu and record start track parameters, the events channel of any data record may be read. The list lu parameter provides the capability to list the data on any valid lu. If an invalid parameter is entered, the command is aborted and an error message is sent to the operator. There should always be one entry in the events channel - the time and status at arming. There may be up to fourty entries. Only valid entries are listed, so the length of the display will be dependent upon how many entries occurred. The display consists of the data lu and track where the data was read, the event number, the status of the eight inputs and the time at which the change occurred.

EXAMPLE:

```
>EV,,72
    EVENT CHANNEL DATA - LU 20   TRACK   72    DAY 252

  WORD     CHANNEL NUMBERS          TIME
   NO      8 7 6 5 4 3 2 1     HR MN  SECONDS
  ----     - - - - - - - -     -- --  ---------
    1      1 1 1 1 1 1 1 1      7 23  22.210539

       [Arming is only valid event.]

>EV
    EVENT CHANNEL DATA - LU 20   TRACK   78    DAY 252

  WORD     CHANNEL NUMBERS          TIME
   NO      8 7 6 5 4 3 2 1     HR MN  SECONDS
  ----     - - - - - - - -     -- --  ---------
    1      0 0 0 0 0 0 0 0      7 25  25.336146
    2      0 0 0 0 0 0 0 1      7 25  35.203397
    3      0 0 0 0 0 0 1 0      7 25  35.205397
    4      0 0 0 0 0 1 0 0      7 25  35.207397
    5      0 0 0 0 1 0 0 0      7 25  35.209397
    6      0 0 0 1 0 0 0 0      7 25  35.211397
    7      0 0 1 0 0 0 0 0      7 25  35.213397
    8      0 1 0 0 0 0 0 0      7 25  35.215397
    9      1 0 1 0 1 0 1 0      7 25  35.217397
   10      1 0 0 0 0 0 0 0      7 25  35.219397
   11      1 1 0 1 0 1 0 1      7 25  35.221397
   12      1 0 0 0 0 0 0 0      7 25  35.223397
   13      1 1 1 1 0 0 0 0      7 25  35.225397

       [Arming plus 12 other events were recorded and are displayed.]

>EV,199
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

EX COMMAND

COMMAND PURPOSE: EXit from ADCHK.

COMMAND FORMAT: EX

COMMAND DESCRIPTION:
The EX command will terminate program ADCHK. There are no housekeeping or cleanup functions performed by this command. All disc stored parameters remain the same.

EXAMPLE:

```
>EX
:
```

GP COMMAND

COMMAND PURPOSE: Get Parameters saved in an FMGR file.

COMMAND FORMAT: GP

COMMAND DESCRIPTION:
After parameters have been saved in a FMGR file, they may be retrieved using the GP command. The GP command places the parameters stored in the file in the parameter area of the current default data lu. The command initially requests the file name to be retrieved. Once opened, all operations continue on this file until the command is terminated. After a valid file name is entered, the GP command menu is displayed and then operator input commands will be accepted. The allowed operator sub-commands are:

- D - Display the channel number and label saved in the parameter file.
- F - Finished - return to PRMAN.
- L - List the GP command menu.
- Mn - Display the values of channel n parameters currently saved on disc.
- On - Display the values of channel n parameters saved in the FMGR file.
- Rn - Replace ALL channel's parameters on disc with the values saved in channel n parameters in the FMGR file.
- S - Store the saved parameters for each channel in the parameter file to its corresponding channel parameter area on disc.
- Xn,m - Store the saved parameters for channel n in the parameter file to the channel m parameter area on disc.

If an invalid sub-command is entered, the menu is repeated. If a request for a channel not saved in the file is made, an error message is sent to the display and the sub-command is terminated. If any FMGR errors are encountered, an error message is sent indicating the FMGR error code and the GP command is terminated. Note that the GP command is executed as a subcommand of PRMAN and may not be executed directly from ADCHK.

EXAMPLE:

```
PRMAN>GP

   ENTER THE PARAMETER FILE NAME:*XYZ

   ERROR IN CALL TO OPEN.  IERR=   -6
     [File *XYZ was not found.]

    LU 19 CHANNELS SELECTED:   1   2   4

PRMAN>GP

   ENTER THE PARAMETER FILE NAME:*LEE

   MENU FOR MANIPULATION OF PARAMETER DATA:
     [Menu would normally be listed here.  It is not listed ]
     [to conserve space.                                    ]
GP>D

     FILE *LEE   :  TEST PARAMETERS
     CHANNELS SAVED:   6   7
GP>R6
     [Replace all disc parameters with saved channel 6 para-]
     [meters.                                               ]
GP>S
     [Channels 6 & 7 will have there disc parameters        ]
     [replaced by the saved values.                         ]
GP>X3,7

   INVALID CHANNEL REQUEST!
     [Channel 3 is not in saved parameter file.]
```

```
GP>7,3
     [Channel 3 will have its disc parameters replaced by    ]
     [saved values of channel 7.                             ]
GP>F

     LU 19 CHANNELS SELECTED:   0   1   2   3   4   5   6   7

PRMAN>
```

HA COMMAND

COMMAND PURPOSE: HAlt the ASRD.

COMMAND FORMAT: HA

COMMAND DESCRIPTION:
The HA command removes the arm signal, effectively halting the ASRD. No other registers or functions are affected. Any data already captured is retained, but no more data is taken. The HA command may be used when it is not desired to save the data taken with the TR command or if it is not desired to execute the TR command immediately. The ASRD can be rearmed using the AR command if desired. However, if any channel has triggered, a new setup should be performed first (LO command).

There are no error, warning or diagnostic messages associated with the HA command.

EXAMPLE:

```
>HA
>
```

HE COMMAND

COMMAND PURPOSE: HEader entry and modification.

COMMAND FORMAT: HE

COMMAND DESCRIPTION:
Each data file is annotated with a header to provide information about the round. The HE command provides a means of entering or changing the header information that will be stored on the data file. A working or scratch header is stored on sector 6 of each data lu. Thus, each data lu may have a different scratch header. When ASRD data is transfered to disc to form the data file, a copy of the scratch header is also stored with the data file. Therefore, the scratch header should normally be updated before each round is fired to insure its accuracy. Usually the only item that changes on a round by round basis is the round number. However, there may be other items which will require update. The items contained in the header are essentially the same as those made on STEAP-MT form 316 BALLISTIC MEASUREMENT DATA SHEET. The entries and the maximum characters for each are:

1. Round number - 4
2. Test director - 10
3. Schedule number - 4
4. Weapon type - 6
5. Tube number - 6
6. Trailer number - 4
7. Operator initials - 3
8. Date - 7
9. Remarks - 74

There is no checking for accuracy or quantity on any of the entries. Up to the maximum number of characters will be accepted. The header program supports three commands:

item#,new value - replaces the currently displayed value for item# with the new value.

F - finished, currently displayed values will be stored on disc.

K - kill, disc values are not changed.

For items 1-8 even if only one character needs to be changed, the entire entry must be typed in. However, for item 9, there are two editting commands to aid the operator. They are:

/ - leaves existing character unchanged, used the same as the EDITR /.

$ - truncates currently displayed characters from $ to end of line.

Note that the | displayed in each item marks the end of the available space for characters. If an operator command error is detected, an error message is displayed.

```
EXAMPLE:
  >HE
   ENTER:  Item#,new val * Finish * Kill * Use / to edit or $ to
                                                          truncate #9
  1.   1234|          RND NO
  2.   DOPPLERMAN|    TEST DIR
  3.   2109|          SCHED NO
  4.   155mm |        WEAPON
  5.   987654|        TUBE NO
  6.   1000|          TRAILER NO
  7.   CLF|           OPR INITIALS
  8.   27SEP82|       DATE(DDMMMYY)
  9.  TEST HEADER FOR DEMONSTRATION PURPOSES                          |

  #1,1235
             [Valid entry will result in item 1 value changing to]
             [new value.  Current header display is erased and    ]
             [new display is written.  # prompt signifies ready   ]
             [for new entry.                                      ]
  #F
             [Currently displayed values are saved on disc.       ]
  >

  ------------------------------------------------------------------

  >HE
   ENTER:  Item#,new val * Finish * Kill * Use / to edit or $ to
                                                          truncate #9
  1.   1235|          RND NO
  2.   DOPPLERMAN|    TEST DIR
  3.   2109|          SCHED NO
  4.   155mm |        WEAPON
  5.   987654|        TUBE NO
  6.   1000|          TRAILER NO
  7.   CLF|           OPR INITIALS
  8.   27SEP82|       DATE(DDMMMYY)
  9.  TEST HEADER FOR DEMONSTRATION PURPOSES                          |

  #99,ERROR
  INPUT ERROR-TRY AGAIN
             [Invalid entry causes error message.  Cursor moves up]
             [to entry and waits for new input.                   ]
  #K
             [No changes are made in values stored on disc.       ]

  >
```

IN COMMAND

COMMAND PURPOSE: INspect a data record.

COMMAND FORMAT: IN,data disc lu,record start track,list lu

data disc lu	- default = current default lu range = 18-20
record start track	- default = last record in directory range = 2-201 on 18 2-1022 on 19 & 20
list lu	- default = operator terminal range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
Although programs are available to plot and analyze the data saved on disc, occassionally it is necessary to look at the raw data. The IN command provides this capability. Using the data disc lu and record start track, any data record on disc may be examined. The results may be sent to a list device using the list lu parameter. If an invalid parameter is entered, the command is aborted and an error message is sent to the operator. The sub-commands which are available within the IN command are:

F	Finished - return to ADCHK.
Gm	Display data in groups of m words. Default value on entry to subroutine is 16. Range is 1 to 32767.
H	Displays a header consisting of each selected channel, the number of words in that channel, and its location in the data record.
Jn	Jump directly to the first word of channel n and display m words (set by G comand). If n is omitted, go to the next available channel.
Sbb	Search forward for code bits bb starting from the current location. Bits 15 & 14 of each word are tested. Valid values of bb are 00, 10, and 11. If a match is found, m words (see G command) are displayed starting with the word which matched. The value of bb is set to 11 when ADCHK is first run. If bb is omitted, the last value entered will be used.
+i	Move forward i samples relative to the bottom of the current display and display m words (see G command) starting at the new value.
-i	Move backward i samples relative to the top of the current display and display m words (see G command) starting at the new value.
n	Go directly to word number n of the record and display m words (see G command).
CR	If no commands are entered and the RETURN key is depressed, the next m words (see G command) will be displayed. If the header was the last thing displayed, then the display will begin on the first data word of the first selected channel.

Data values are interpreted as being data, change words or IRIG time words. However, header, channel location, parameters, events, or annotation entries are not interpreted by this command. The octal values will be displayed, but the values are not converted to meaningful results. If the display crosses the boundary between two channels, a warning label is inserted. However, if a jump from within one channel to within another channel is made, no label is provided. The operator must keep track of his location. If the end of the data record is encountered, a warning message is displayed and no additional words are displayed. The command first displays the data lu and track where the data was read. Then a header for annotating the data is listed. This information is locked on the display memory and will not change until the IN command is terminated. Initially a list of selected channels, number of words in the channel and location in the data record is displayed (see H command above).

EXAMPLE:

```
>IN
 DATA STORED ON LOGICAL UNIT 19 STARTING AT TRACK  227

    SAMPLE   1 111 11                    RATE   VALUE
    NUMBER   5 432 109 876 543 210       CODE   VOLTS
    ------   ----------------------      ----   -----
       [The header above is locked in memory and does not change.]
    1289      CHANNEL #  0             2047 WORDS
       0      CHANNEL #  5                0 WORDS
    3336      CHANNEL #  6             2047 WORDS
    5383      CHANNEL # 11             7133 WORDS
       [The display of sample number, channel number, and number]
       [of words in the channel is always given first by the IN ]
       [command.  The same display can be obtained later using  ]
       [the H sub-command.  Note that if a channel is selected  ]
       [but does not trigger, it will appear only in the header.]
ENTER:   #, +i, -i, H<eader>, G<roup>m, Sbb, Jn, F<inish> or 'CR'
J0
*************** BEGINNING OF CHANNEL   0 DATA ***************
    1289     0 000 000 000 000 100     8       .020
    1290     0 000 000 000 000 100     8       .020
    1291     0 000 000 000 000 101     8       .024
    1292     0 000 000 000 000 101     8       .024
    1293     0 000 000 000 000 110     8       .029
    1294     0 000 000 000 000 110     8       .029
    1295     0 000 000 000 000 110     8       .029
    1296     0 000 000 000 000 101     8       .024
    1297     0 000 000 000 000 011     8       .015
    1298     0 000 000 000 000 011     8       .015
    1299     1 000 000 000 001 100   CHGWD=    12
    1300     0 001 000 000 000 100     9       .020
    1301     0 001 000 000 000 100     9       .020
    1302     0 001 000 000 000 011     9       .015
    1303     0 001 000 000 000 100     9       .020
    1304     0 001 000 000 000 100     9       .020
       [Jump to channel 0.  First 16 words are displayed.     ]
       [Since start of channel, warning message is displayed.]
ENTER:   #, +i, -i, H<eader>, G<roup>m, Sbb, Jn, F<inish> or 'CR'
G6
       [Set grouping to 6.  When G command is entered, value ]
       [is erased and cursor moves to old position and waits ]
       [for new entry.                                       ]
S11
       [Search for IRIG time words is initiated.             ]
    1369     1 100 000 110 100 001      7H:23M:29.806026S
    1370     1 111 010 001 101 010
    1371     1 101 100 000 000 110
    1372     1 100 000 000 000 110
    1373     0 010 111 111 111 110     A    -0.005
    1374     0 010 111 111 111 101     A    -0.010
```

```
ENTER:   #, +i, -i, H<eader>, G<roup>m, Sbb, Jn, F<inish> or 'CR'
1711
        [Go directly to sample 1711.]
     1711     0 010 000 000 000 000     A      .000
     1712     0 010 000 000 000 000     A      .000
     1713     0 010 000 000 000 000     A      .000
     1714     0 010 000 000 000 000     A      .000
     1715     0 010 000 000 000 001     A      .005
     1716     0 010 000 000 000 001     A      .005
ENTER:   #, +i, -i, H<eader>, G<roup>m, Sbb, Jn, F<inish> or 'CR'
+100
        [Go forward 100 samples from bottom of last display.]
     1816     0 010 000 000 000 001     A      .005
     1817     0 010 000 000 000 001     A      .005
     1818     0 010 000 000 000 001     A      .005
     1819     0 010 000 000 000 001     A      .005
     1820     0 010 000 000 000 001     A      .005
     1821     0 010 000 000 000 001     A      .005
ENTER:   #, +i, -i, H<eader>, G<roup>m, Sbb, Jn, F<inish> or 'CR'
-16
        [Go backward 16 samples from top of last display.]
     1800     0 000 000 000 000 000     8      .000
     1801     0 000 000 000 000 001     8      .005
     1802     0 000 000 000 000 001     8      .005
     1803     0 000 000 000 000 001     8      .005
     1804     0 000 000 000 000 000     8      .000
     1805     0 000 000 000 000 001     8      .005
ENTER:   #, +i, -i, H<eader>, G<roup>m, Sbb, Jn, F<inish> or 'CR'
15000
INVALID ENTRY -- TRY AGAIN
        [Attempt to go directly to sample 15000 is aborted since]
        [there are not that many entries in record.  Cursor is  ]
        [moved back to old entry and waits for new entry.       ]
2568
   12515     0 001 000 000 000 001     9      .005

END OF RECORD
        [12515 is last entry in record.  Warning message is issued.]

ENTER:   #, +i, -i, H<eader>, G<roup>m, Sbb, Jn, F<inish> or 'CR'
F

>IN,119,227
IMPROPER COMMAND FORMAT**TRY AGAIN   >
```

LC COMMAND

COMMAND PURPOSE: Displays the cuurent Last Channel and allows the last channel to be changed.

COMMAND FORMAT: LC

COMMAND DESCRIPTION:
When ADCHK is first run, the last channel is established by subroutine NOCHL. This value should correspond to the number of channels actually in the trailer where the software resides. If it is necessary to examine data taken on a system with more or less channels, then the LC command may be used to change the value to correspond to the data being examined. The command displays the current value and then asks the operator if a change is desired. If no change is required, the value is not changed and the command terminates. If a change is requested, then the operator is given an opportunity to enter the new value. If an invalid entry is made, no change is made and the display is repeated until a valid answer is obtained.

EXAMPLE:

```
>LC
     CURRENT LAST CHANNEL IS 15 -- CHANGE(Y OR N):N

>
       [Current last channel is 15.  No change is made.]
```

```
>LC
     CURRENT LAST CHANNEL IS 15 -- CHANGE(Y OR N):Y

     ENTER NEW LAST CHANNEL:7

>
       [Last channel is changed from 15 to 7.]
```

```
>LC
     CURRENT LAST CHANNEL IS 15 -- CHANGE(Y OR N):M
     CURRENT LAST CHANNEL IS 15 -- CHANGE(Y OR N):N

>
       [Invalid entry results in repeat of question.]
```

```
>LC
     CURRENT LAST CHANNEL IS 15 -- CHANGE(Y OR N):Y

     ENTER NEW LAST CHANNEL:311
     CURRENT LAST CHANNEL IS 15 -- CHANGE(Y OR N):Y

     ENTER NEW LAST CHANNEL:31

>
       [Invalid entry results in repeat of question.]
```

LI COMMAND

COMMAND PURPOSE: LIst the ADCHK commands.

COMMAND FORMAT: LI,list lu

list lu - default = operators terminal
range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
The LI command writes the list of valid ADCHK commands to the specified list device. If an invalid list device is specified, the command will be aborted and an error message sent to the operator.

EXAMPLE:

```
>LI
                VALID ADCHK COMMANDS ARE:
AR              ARm ASRD                    NO,u,k        NOtes
BU              BackUp data disc            OV,lu         OVerloads on 416
CH              CHannel scan                PA,c          PArameter changes
CK              CheckK ASRD                 PD,lu         Parameter Display
CO,u,k,c        COmments                    PM            Param. Maintenance
DI,sec,sub      DIsplay ASRD signal         PU            PUrge directory
DL,m,n,lu       Directory List(m-n)         RA,u,k,lu     Read All parameters
EV,u,k,lu       EVents display              RC,u,k,c,lu   Read Comments
EX              EXit ADCHK                  RE            REstore from tape
HA              HAlt ASRD                   RH,u,k,lu     Read Header
HE              HEader changes              RN,u,k,lu     Read Notes
IN,u,k,lu       INspect data                RP,u,k,lu     Read Parameters
LC              Last Channel changes        RT,n          Rate Table n changes
LI,lu           LIst ADCHK commands         RU,nmr,p1-5   RUn program nmr
LO,lu,skip      LOad ASRD & sig cond        SE            SElect channels
LU              change default LU           ST,lu         Self Test ASRD
MH,u,k          Modify Header               TR,u,k        TRansfer data
ML,u,k,c        Modify Labels               VE,c          VErify rate table

     where: u,k = data lu (18-20) & track (2-1022)
            lu  = list device's logical unit # (1,3-6,8,9)
            c   = channel number. (0<c<15)

>
```

```
>LI,7
IMPROPER COMMAND FORMAT**TRY AGAIN    >

       [Invalid parameter entry.  Lu 7 is not a valid list]
       [device.                                           ]
```

LO COMMAND

COMMAND PURPOSE: LOad signal conditioner, filter/amplifier and ASRD with setup parameters.

COMMAND FORMAT: LO,skip switch

skip switch - default = no change
range = -32768 to 32767

COMMAND DESCRIPTION:
After all parameters have been entered using the PA command, the parameters may be loaded into the signal conditioner, filter/amplifier and ASRD using the LO command. The LO command performs its operations in the following sequence: ASRD, signal conditioner, filter/amplifier, clear overloads, and check overloads. The LO command uses the parameters from the current default lu. If it is desired to load only the ASRD and bypass the signal conditioner, filter/amplifier and overload check, then the skip switch may be set. Any non-zero entry for this parameter will set the switch. Once set the switch remains set until another non-zero entry is received. A warning message is sent to the operator each time the LO command is executed when the switch is set. This feature is useful when testing the ASRD. The program performs the following tasks:

ASRD - Clears all channels. Disables all channels for data acquisition. For each selected channel the following steps are taken: Zero error status registers. Send parameters to ASRD. Load parameters to registers. Check for load errors. Enable for data acquisition. Set to display mode.

316 - Sets calibrator voltage to zero, turns calibrator off, sets balance to manual, and sets strain gain to X1 on all channels. If channel is selected, checks mode for validity. If strain mode, checks gain and balance for validity. If mode is invalid, channel is bypassed. If balance is invalid, manual is set. If gain is invalid, X1 is set. Warning messages for invalid parameters are sent to the operator.

416 - If channel is selected, checks gain and frequency for validity. If gain is negative, warns operator. If gain is invalid, loads X1/X1. If frequency is invalid, loads 0.5kHz. If channel is not selected, sets gain to X1/X1 and frequency to 5kHz. Warning messages for invalid parameters are sent to the operator.

OVER LOADS - Clears all overloads that may have been generated by previous programming or transients. Reads overloads again and warns operator if any remain. Lists channel number followed by overload code: 1 = prefilter, 2 = post filter, 3 = both.

When all operations are complete a message is sent to the operator indicating which data lu parameters were used.

EXAMPLE:

```
>LO
 SETUP COMPLETE -- DATA LU 19

>LO,1
      PROGRAMMING OF 316 & 416 BYPASSED
 SETUP COMPLETE -- DATA LU 19
      [Skip switch was set.]

>LO
      PROGRAMMING OF 316 & 416 BYPASSED
 SETUP COMPLETE -- DATA LU 19
      [Skip switch remains set.]

>LO,9999
 SETUP COMPLETE -- DATA LU 19
      [Skip switch is reset.]
```

```
>LO
 SETUP COMPLETE -- DATA LU 19
       [Skip switch remains reset.]

>
```

```
>LO
 DIG DECR DELAY ERR-CHNLS:   5
 DIG TRIGR WORD ERR-CHNLS:  11  14
 DIG STOP DELAY ERR-CHNLS:   7
 DIG STRT DELAY ERR-CHNLS:   7
 DIG SCLR/LNGTH ERR-CHNLS:  12
 DIG RTTBL ADRS ERR-CHNLS:   0
 DIG RTTBL DATA ERR-CHNLS:   4   9  13
 DIG MEMRY SPEC ERR-CHNLS:   8  15
 DIG PRETRIGGER ERR-CHNLS:   3
 316 CHNLS WITH BAD MODE:   00
 316 CHNLS WITH BAD GAIN:   01
 316 CHNLS WITH BAD BLNC:   02
     CHNLS WITH NEG GAIN:   03  07  11
 416 CHNLS WITH BAD GAIN:   04
 416 CHNLS WITH BAD FREQ:   05
 OVERLOADS:    01/2   05/3   14/1   15/3

 SETUP COMPLETE -- DATA LU 20
       [Error messages which could be encountered during]
       [a load.                                         ]

>
```

LU COMMAND

COMMAND PURPOSE: Displays the current default data lu and allows the default data lu to be changed.

COMMAND FORMAT: LU

COMMAND DESCRIPTION:
When ADCHK is first run, the default data lu is set to 19. The default value is used by many of the other ADCHK commands. The LU command provides the capability to check what the current default data lu value is and to change that value if desired. The allowed data lus are 18, 19, & 20. If a value other than these three is entered, no change is made and the display will be repeated until a valid entry is obtained.

EXAMPLE:

```
>LU
 PRESENT DEFAULT LU=19      NEW DEFAULT LU=?20

>
         [Default lu has been changed from 19 to 20.]
```

```
>LU
 PRESENT DEFAULT LU=20      NEW DEFAULT LU=?18

>
         [Default lu has been changed from 20 to 18.]
```

```
>LU
 PRESENT DEFAULT LU=19      NEW DEFAULT LU=?21
 PRESENT DEFAULT LU=19      NEW DEFAULT LU=?20

>
         [Invalid entry results in repeat of question.]
```

MH COMMAND

COMMAND PURPOSE: Modify Header from a data record

COMMAND FORMAT: MH,data disc lu,record start track

data disc lu - default = current default lu
range = 18-20

record start track - default = last record in directory
range = 2-201 for 18
2-1022 for 19 & 20

COMMAND DESCRIPTION:
The header created by the HE command is automatically stored with the data record when it is created. If it is necessary to modify a data record header then the MH command may be used. Using the data disc lu and record start track parameters it is possible to modify the header from any data record stored on disc. The data disc lu and record start track used are displayed first. Then the nine items in the header are displayed. The commands and procedures for modifying the header are the same as those used to create the header using the HE command. If invalid parameters are entered, the command is aborted and an error message is sent to the operator. However, if a valid track is entered but it is not the start track for a record, the display may be filled with garbage.

EXAMPLE:

```
>MH
      DATA LU 19      TRACK 195
 ENTER:  Item#,new val * Finish * Kill * Use / to edit or $ to
                                                    truncate #9
1.   1      |        RND NO
2.   CHEATER    |   TEST DIR
3.          |       SCHED NO
4.   4.2"   |       WEAPON
5.          |       TUBE NO
6.   1000|          TRAILER NO
7.   DWC|            OPR INITIALS
8.   25MAR82|       DATE(DDMMMYY)
9.  329A1   INCREMENT 41

          [Header from default lu 19, track 195 is displayed. ]
#1,2
          [Valid entry will result in item 1 value changing to]
          [new value.  Current header display is erased and    ]
          [new display is written.  # prompt signifies ready   ]
          [for new entry.                                      ]
#F
          [Currently displayed values are saved in data record. ]
          [Header has been modified.                            ]

>
```

```
>MH,18,2
      DATA LU 18      TRACK    2
 ENTER:  Item#,new val * Finish * Kill * Use / to edit or $ to
                                                    truncate #9
1.   1    |          RND NO
2.   KURIATA    |    TEST DIR
3.        |          SCHED NO
4.   155mm |         WEAPON
5.        |          TUBE NO
6.   700 |           TRAILER NO
7.   WCN|            OPR INITIALS
8.   14NOV80|       DATE(DDMMMYY)
9. WARMER
```

[Header from specified lu & track is displayed.]

```
#33,2089
INPUT ERROR-TRY AGAIN
```

[Invalid entry causes error message. Cursor]
[moves up to entry and waits for new input.]

```
#K
```

[No changes are made in data record header.]

```
>
```

```
>MH,21
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

[Invalid parameter entry. Command aborted.]

ML COMMAND

COMMAND PURPOSE: Modify Labels in a data record.

COMMAND FORMAT: ML,data disc lu,record start track,channels number

data disc lu	- default = current default lu range = 18-20
record start track	- default = last record in directory range = 2-201 for 18 2-1022 for 19 & 20
channel number	- default = 0 range = 0 to last channel

COMMAND DESCRIPTION:

In general once the setup parameters are stored in a data record, it is not desirable to change them. However, the type data, plot labels and remarks entries do not affect hardware, so they may be modified some time after the data record has been created without destroying the integrity of the data. The ML command provides the means to do this. The command input parameters of data disc lu and record start track allow any data record stored on disc to be read. The command input parameter of channel number allows going directly to a channel of interest. If an invalid command input parameter is entered, the command is aborted and an error message is sent to the operator. The ML command supports four subcommands:

F - Finished - save currently displayed values.

K - Kill - do not save currently displayed values.

C,c - save currently displayed values and go to channel c.

n - edit label n. Edit commands are:

/ - leaves existing character unchanged, used same as EDITR /.

$ - truncates currently displayed characters from $ to end of line.

| - denotes end of label field.

If an invalid ML subcommand is entered, the command is not executed and an error message is sent to the operator.

EXAMPLE:

```
>ML

     LABELS FOR CHANNEL  0     DATA LU 19     TRACK  195

1. PLOT LABEL:CH 3 BLAST POS                    |
2. Y-AXIS LBL:kPA                    |
3. RMRKS:M319 ADAPTER IN USE                                    |
4. TYPE DATA:T|

ENTER:  Item # to edit  C,new chnl  Finish  Kill
#1
CH 3 BLAST POS                |
/////////////// NO 5
```

[Labels from last data record on current default lu are displayed. Operator has asked to edit label 1. Changes will cause current display to be erased and new values will be displayed. # prompt signifies ready for new entry.]

```
#9
INPUT ERROR - TRY AGAIN
      [Invalid command entry results in error message.      ]
      [Cursor moves up to prompt and waits for new entry.   ]
#F
      [Currently displayed values are saved in data record.  ]
      [Labels have been modified.                            ]
>
```

```
>ML,18,2

     LABELS FOR CHANNEL  0     DATA LU 18     TRACK   2

CHANNEL NOT SELECTED -- ENTER:  C,new chnl  Finish  Kill

#
      [Specified data disc lu and track have been read.     ]
      [However, channel zero was not selected.              ]
```

```
>ML,199,210,3
IMPROPER COMMAND FORMAT**TRY AGAIN    >

      [Invalid parameter entry.  Command aborted.]
```

NO COMMAND

COMMAND PURPOSE: NOtes annotation for a data record.

COMMAND FORMAT: NO,data disc lu,record start track

data disc lu - default = current default lu
range = 18-20

record start track - default = last record in directory
range = 2-201 for 18
2-1022 for 19 & 20

COMMAND DESCRIPTION:
After a data record is created by the TR command, it may be necessary to annotate the record with additional documentation information. The NO command provides the capability to do this. Using the data disc lu and record start track parameters any data record on disc may be annotated. If an invalid parameter is entered, the command is aborted and an error message is sent to the operator. There is storage space for 128 characters of information. For display purposes, the storage space is divided into two lines of 64 characters each. Entries can be made on either line. However, the normal procedure would be to fill up line one first, then to fill line two. The entries can be edited using the following commands:

/ - leaves existing character unchanged, used same as the EDITR /.

$ - truncates currently displayed characters from $ to end of line.

A | is displayed to mark the end of the available space for character input. If an operator command error is detected, an error message is displayed.

EXAMPLE:

```
>NO

                    NOTES:   FROM LU 19   TRACK  231

  1.
  2.

 ENTER: # to edit, Finish, Kill  >>1
   [Currently no annotation information is stored in the record. ]
   [Line 1 is to be edited.  Current contents will be displayed  ]
   [immediately below answer left justified for easy entry.      ]
   [New values may be added.                                     ]

GUN BRACKET BROKE TEARING ALL CABLES OFF.  ONLY FIRST 40
   [New values for line 1 have been entered.  Current display  ]
   [will be erased and new values written in old values place. ]
   [Next command may be entered.                               ]
 ENTER: # to edit, Finish, Kill  >>22
INPUT ERROR - TRY AGAIN
   [Invalid entry results in error message.  Cursor moves to   ]
   [old entry and waits for new entry.                         ]
 ENTER: # to edit, Finish, Kill  >>2

MSEC OF DATA IS VALID.
   [New values for line 2 have been entered.  Current display  ]
   [will be erased and new values written in old values place. ]
   [For clarity display is repeated here.                      ]

  1.  GUN BRACKET BROKE TEARING ALL CABLES OFF.  ONLY FIRST 40
  2.  MSEC OF DATA IS VALID.
 ENTER: # to edit, Finish, Kill  >>F
   [Editing is finished.  Currently displayed values are saved ]
   [in data record.                                            ]

>NO,18,231
IMPROPER COMMAND FORMAT**TRY AGAIN   >
   [Invalid parameter entry.  Command aborted.]
```

OV COMMAND

COMMAND PURPOSE: OVerload display.

COMMAND FORMAT: OV,list lu

list lu - default = operator terminal
range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
To check if any filter/amplifier channel is overloaded, the OV command may be used. The OV command reads all the channels on the 416, then displays the results on the list lu. The list will give the channel number plus the overload code. The codes are: 1-prefilter, 2-postfilter, and 3-both. Channels with no overloads are not displayed.

EXAMPLE:

```
>OV
 OVERLOADS:     01/2     09/1     13/3     15/1

>OV
 OVERLOADS:

>
```

PA COMMAND

COMMAND PURPOSE: PArameter entry and modification.

COMMAND FORMAT: PA,channel number

channel number - default = 0
range = 0 to last channel

COMMAND DESCRIPTION:

The PA command provides a means of entering the parameters each channel needs for setting up the signal conditioning, the filter/amplifier, and the ASRD for data acquisition. In addition, it provides a way to enter documentation information that is needed to later process the record. There are twenty-seven parameters which can be entered or modified by the PA command. Each channels setup parameters are stored on a data lu. The stored values are read by the LO command to setup the hardware. When the data file is created, a copy of these parameters is stored with the data file. Each data lu has its own set of parameters for the PA command to work on. The PA command always works on the default data lu. The PA command checks each entry as carefully as possible to determine that it is correct and that it is consistent with other parameters already entered. However, the operator must carefully check all entries to determine their appropriateness for the particular job at hand. The commands which are available within the PA command are:

Cn	Writes the displayed values of the current channel to disc, then reads and displays the values for channel "n".
F	Writes the displayed values of the current channel to disc and returns to ADCHK.
K	Return to ADCHK. The displayed values of the current channel are NOT written to disc.
i,v	Changes parameter number "i" (1<=i<=24) to the value specified by "v". NOTE: The new value will be displayed immediately, but the disk in not updated until the issuance of a "C", "F", or "FA" command.
i	Causes parameter number "i" (25<=i<=27) to be displayed at the bottom of the screen for editing. The available editing commands are: / Leaves the existing character unchanged. * Truncates the displayed line at the point inserted. | Indicates the end of the available field.

The parameters which this subroutine accesses are:

param#	parameter	range	description
1	SELECTED	Y,N	Y=SELECTED N=NOT SELECTED Determines if a channel will be used or not. If selected the signal conditioning, filter/amplifier and ASRD are setup using the parameters entered on disc by PARAM.
2	MASTER	Y,N	Y=MASTER N=NOT MASTER Determines if a channel will be a master to trigger slave channels. More than one channel may be a master.
3	SLAVE	Y,N	Y=SLAVE N=NOT SLAVE Determines if a channel will be a slave to trigger from a master channel. More than one channel may be a slave. A channel may be a master and a slave simultaneously.

4	XTRIGR	Y,N	Y=EXTERNAL TRIGGER N=INTERNAL TRIGGER Determines if a channel will trigger from an external TTL pulse on J4 or from the internal trigger circuitry.
5	THRSHLD V	+/-9.9	INTERNAL TRIGGER THRESHOLD SETTING The value in volts to be used to trigger the the channel.
6	SLOPE	R/+,-,A	INTERNAL TRIGGER SLOPE (+, -, A, R+ or R-) Determines the trigger qualifier to be used with the threshold setting. Allows three slope choices +, - or A where A is absolute value of the threshold of parameter 5. An additional modifier, the R, may be added to make the trigger relative to the value at arming time. There are five possible choices: +, -, A, R+ or R-.
7	RUN DLY	0-5240	RUN DELAY The time to delay in milliseconds from the point where a trigger is received to the point where data acquisition begins.
8	STP DLY	0-5240	STOP DELAY The time to delay in milliseconds from the point where a trigger is lost to the point where data acquisition stops. Note that if the stop delay is shorter than the run delay, it is possible that little or no data will be taken if the signal does not remain triggered long enough for the start delay to expire.
9	PRETRIGR	0-127	PRETRIGGER MEMORY SIZE IN kWORDS The number of k words of data to be saved before the trigger point. The PRETRIGR size must be less than or equal to the MEM SIZE value.
10	FIX RATE	Y,N	Y=FIXED RATE N=VARIABLE SAMPLING RATE Determines whether all data points will be uniformly spaced in t e(fixed rate) or whether the rate will be adjusted for the actual data bandwidth.
11	MEM SIZE	1-128	MEMORY SIZE IN kWORDS The memory size in terms of 1k blocks. This number can be smaller than the actual amount of memory installed, but it should never exceed the amount of memory actually installed. Also, the MEM SIZE must be greater than or equal to PRETRIGR value.
12	TYPE DATA	A-Z	TYPE DATA CODE A single character code which specifies the type of processing program to be used to process the data.
13	MODE	V,C,S	SIGNAL CONDITIONER MODE Determines what mode the Precision Filters 316 signal conditioning unit will operate in: V=VOLTAGE C=CHARGE S=STRAIN
14	BALANCE	A,M	STRAIN BRIDGE BALANCE METHOD Determines whether AUTOMATIC(A) or MANUAL(M) balance will be used on the strain bridge of the Precision Filters 316 signal conditioner.

15 FREQ(KHZ) .5-155 DATA BANDWIDTH
The frequency in kHz that will be used to set the filter in the Precision Filters 416 filter/amplifier. PARAM rounds all entries to the nearest 5kHz increment in the range 16-155kHz and to the nearest 0.5kHz in the range 0.5-15.5kHz. If strain mode is selected and gain is greater than 1000, the system bandwidth is reduced by the Precision Filters 316 signal conditioner. If the gain will result in a bandwidth below the value specified in this parameter, a minus sign will be added to the GAIN parameter value to warn the operator. The 316 bandwidths are 25kHz(X10), 2.5kHz(X100) and 0.25kHz(X1000).
If the bandwidth entered is positive, the software will automatically determine the appropriate rate table or rate code.
If the bandwidth entered is negative, the choice of rate table or rate code is left to the operator in parameter 16 (RATE T/C).

16 RATE T/C 0-19/ 2-F RATE TABLE/RATE CODE
Determines what rate table/rate code will be used when the automatic default is NOT selected by parameter 15 (FREQ).
If parameter 10 (FIX RATE) is N, this parameter will select a rate table 0-19.
If parameter 10 (FIX RATE) is Y, this parameter will select a rate code 2-F.

17 GAGE FCTR real# TRANSDUCER GAGE FACTOR
Sets the transducer gage factor so that all data for the channel can be scaled in engineering units. The units that should be used when entering the gage factor for each mode are:

Voltage - xyz/volt
Charge - xyz/picocoulomb
Strain - xyz/volt/volt

where xyz is whatever engineering units the final data is to have (i.e. PSI, kilopascals, etc.). The same units must be used in response to parameters 19 (PEAK SGNL), 20 (GF UNITS), and 25,26 (PLOT LABELS). If the units are not consistent, the gain will be calculated incorrectly. Many transducers come calibrated in terms of a sensitivity (picocoulombs/xyz). The sensitivity must be inverted to provide the gage factor.

18 EXCT/INTR real# EXCITATION VOLTAGE/TRANSDUCER INTERCEPT
If strain mode is being used the value of the bridge excitation voltage should be entered here. PARAM will not allow the mode to be changed to strain if parameter 18 is zero. So parameter 18 must be changed then the mode.
If voltage or charge mode are being used and the transducer has an intercept value in its calibration, it is entered here. If there is no intercept value, 0.0 must be entered.

19 PEAK SGNL real# EXPECTED PEAK SIGNAL
If automatic gain calculation is desired, the expected peak signal should be entered. The peak signal value should be in the same units (xyz) as the gage factor. A 15% safety factor is used in the calculations to prevent clipping, so realistic peak signals should be entered.

If manual calculation of the gain is desired, a value of 0.0 should be entered. The manually calculated gain can then be entered in parameter 21 (GAIN).

20	GF UNITS		GAGE FACTOR UNITS The units used on the gage factor entry in parameter 17 (GAGE FCTR).
21	GAIN	real#	GAIN If automatic gain calculation is selected, this entry will display the calculated gain converted to actual gain settings available on the Precision Filters 316 & 416. If the calculated gain is less than 1 (indicating that clipping will occur), a value of -1 is displayed to warn the operator. If the calculated gain exceeds the maximum available gain (1000 in voltage & charge or 1000000 in strain) a -1000 or -1000000 is displayed to warn the operator. A minus sign may also appear in strain mode if a bandwidth conflict will result as listed in parameter 15 (FREQ). If manual entry is selected, the operator can enter a gain from 1-1000 in voltage or charge mode and a gain of 1-1000000 in strain mode. The displayed gain is converted to actual available gain settings. A minus sign will appear in the gain only if a bandwidth conflict will result.
22	XDCR MFG	6char	TRANSDUCER MANUFACTURER Enter brand name of the transducer manufacturer.
23	XDCR MDL#	6char	TRANSDUCER MODEL NUMBER Enter the model number of the transducer.
24	XDCR SN	6char	TRANSDUCER SERIAL NUMBER Enter the transducer serial number.
25	PLOT LABEL		PLOT MAIN LABEL Enter up to 34 characters which will be used to label the plots. Round number and channel number are automatically written to the plot so they need not be entered here.
26	Y-AXIS LBL		PLOT Y-AXIS LABEL Enter up to 30 characters which will be used to label the y-axis of the plot.
27	RMRKS		REMARKS Enter up to 68 characters of additional information if needed.

As was pointed out in the list above, there is interaction between some of the parameters. For clarity, the parameters which interact are listed below:

param#	value	effect
1	N	Current channel is not selected. Only parameter 1 will be displayed and can be changed.
1	Y	Current channel is selected. Other parameters will be displayed and can be changed.
4	N	External trigger is NOT selected. Parameters 5 & 6 (THRSHLD & SLOPE) will be displayed and can be changed.
4	Y	External trigger is selected. Parameters 5 & 6 (THRSHLD & SLOPE) will NOT be displayed and can NOT be changed.

10	N	Parameter 16 (RATE T/C) display range will be 0-19.
10	Y	Parameter 16 (RATE T/C) display range will be 2-F.
13	S	Strain mode is selected. Parameter 14 (BALANCE) will be displayed and can be changed.
13	V,C	Voltage or charge mode is selected. Parameter 14 (BALANCE) will NOT be displayed and can NOT be changed.
15	neg#	Parameter 16 (RATE T/C) will be displayed and can be changed.
15	pos#	Parameter 16 (RATE T/C) will NOT be displayed and can NOT be changed.
18	0.0	Parameter 13 (MODE) may not be changed from V or C to S unless a non-zero value is entered for the EXCIT.
18	#0.0	Parameter 13 (MODE) may be changed from V or C to S.
19	0.0	Parameter 21 (GAIN) will be displayed and may be changed.
19	#0.0	Parameter 21 (GAIN) will be displayed but may NOT be changed. The displayed value is calculated from the peak signal and gage factor.

EXAMPLE:

```
  >PA
      [Valid command has been entered.  Display will be erased and]
      [parameter values will be displayed.                        ]
       PARAMETERS FOR CHANNEL   0      LU 19      TRACK     0
 #    value    parameter   range      #      value      parameter   range
---   -----    ---------   -----     ---   ----------   ---------   -----
 1.       Y *  SELECTED      Y,N      13.            V *  MODE        V,C,S
 2.       N    MASTER        Y,N      14.               BALANCE      A,M
 3.       N    SLAVE         Y,N      15.         80.0 *  FREQ(KHZ)   0.5-155
 4.       N *  XTRIGR        Y,N      16.               RATE T/C     0-19/2-F
 5.     3.5    THRSHLD V   +/-9.9     17.   .1019E+01   GAGE FCTR    real#
 6.       +    SLOPE      R/+,-,A     18.   .0000E+00   EXCT/INTR    real#
 7.       0    RUN DELAY   0-5240      19.   .4000E+05   PEAK SGNL    real#
 8.      13    STP DELAY   0-5240      20.        PSI/V   GF UNITS     6 chr
 9.       0    PRETRIGER   0-127       21.         400.   GAIN         1-iM
10.       Y    FIX RATE      Y,N       22.      KISTLR    XDCR MFG     6 chr
11.       2    MEM SIZE    1-128       23.       V500     XDCR MDL#    6 chr
12.       T    DATA TYPE     A-Z       24.      9B1201    XDCR SN      6 chr
25. PLOT LABEL: CHAMBER PRESSURE
26. Y-AXIS LBL: PSI
27. RMRKS:M399 ADAPTER INSTALLED

 ENTER:  <param#,newvalue> - C<newchannel#> - K<ill> - F<inish>

      [Current default Iu is 19.  Channel 0 is displayed as       ]
      [default parameter.  Parameters may be changed as desired.  ]
      [If an invalid entry is made the error message              ]
      [    INVALID ENTRY - TRY AGAIN!                             ]
      [will be displayed and cursor will move up to old entry and ]
      [wait for new input.                                        ]

  ------------------------------------------------------------

  >PA,99
  IMPROPER COMMAND FORMAT**TRY AGAIN    >

      [Invalid parameter entry.  Command aborted.]
```

PD COMMAND

COMMAND PURPOSE: Parameter Display

COMMAND FORMAT: PD,list lu

list lu - default = operator terminal
range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
Parameters are entered on a channel by channel basis using the the PA command. To obtain a display of all channels parameters at one time the PD command may be used. The PD command will display all selected channels from the current default lu. Only eighteen of the twenty-seven possible parameters are displayed. The nine parameters which are not displayed are primarily the labels and documentation parameters. The essential data acquisition parameters are all displayed. The PD command does not allow any modification of the parameters displayed. Parameters which are not active are not displayed. The display may be sent to any valid list device using the list lu input parameter. If an invalid list lu parameter is entered, the command is aborted and an error message is sent to the operator.

EXAMPLE:

```
>PD
        PARAMETERS FROM LU 19, TRACK     0

 CH  SEL MAS SLA XTR  THRS   SLP  RUN DLY  STP DLY  PRE TRG  FIX  MEM SIZ  TYP  MOD  BAL   FREQ KHZ  RATE   GAGE FACTOR      GAIN
 --  --- --- --- ---  ----   --  -------  -------  -------  ---  -------  ---  ---  ---  -------  ----  -----------  ---------
  0   Y   N   N   N    7.9    +        0       30        0   N        2    T    S    A      2.5          .1000E+01          10.
  1   Y   N   N   N    7.9    +        0       30        0   N        2    T    S    A      4.0          .1000E+01        2500.
  2   Y   N   N   N    7.9    +        0       30        0   N       16    B    C         -2.0   18     .7800E-01        1000.
  3   Y   Y   N   N   -3.6    -        4       18        0   Y        2    T    S    M      1.0          .1040E+02     -400000.
  4   Y   N   N   Y                    0       30        0   Y        2    T    S    A       .5          .1000E+01       50000.
  5   Y   N   N   N    5.0    A        0       30        0   N        2    T    S    A     80.0          .1000E+01      -10000.
  6   Y   N   N   N    7.9    +        0       30        0   N        2    T    S    A       .5          .1000E+01      100000.
  7   Y   N   N   N   -2.5   R+        0       30        0   N        2    T    S    A     35.0          .1000E+01    -1000000
  8   Y   N   Y   N   -2.5   R-        0       30        0   Y        8    T    S    A    -13.5    C     .1346E+02          10.
  9   Y   N   N   Y                    0       30        0   Y        2    T    S    M      4.0          .4670E+00        2500.
 10   Y   N   N   N    7.9    +        0       30        0   Y        2    T    V         4.0          .1000E+01           1.
 11   Y   N   N   N    7.9    +        0       30        0   Y        2    T    V        13.5          .1000E+01           1.
 12   Y   N   N   N    7.9    +        0       30        0   Y        2    T    V         4.0          .1000E+01           1.
 14   Y   N   N   N   -3.5    +        0       30        0   Y        2    T    V         4.0          .1000E+01           1.
 15   Y   N   N   N    5.0    -        0       13        0   Y        4    R    V      -155.0    F     .9500E+01           1.

  [Parameters from current default lu are displayed on operators]
  [terminal.                                                    ]

  ------------------------------------------------------------

  >PD,10
  IMPROPER COMMAND FORMAT**TRY AGAIN    >

   [Invalid parameter entry.  Command aborted.]
```

PM COMMAND

COMMAND PURPOSE: Parameter Maintenance.

COMMAND FORMAT: PM

COMMAND DESCRIPTION:
The PM command provides a number of utility functions for the maintenance of parameters on a BTST data disc. The PM command always works on the current default data lu. The PM command checks each entry as carefully as possible to determine that it is correct. However, the operator must exercise great care in use of some of the PM commands as they can cause difficult to correct changes in the data lu parameters. The commands which are available within the PM command are:

EX	- EXit PRMAN and return to ADCHK.
GP	- Get Parameters saved in an FMGR file previously by the SP command. This command has its own set of subcommands. Reference the GP command instruction sheet for use of this command.
LI,lu	- LIst PRMAN commands to list device lu.
MP,p,c	- Move Parameter number p of channel c to all other channels currently selected.
MS,c	- Move all parameters of channel c to all other channels currently Selected. This command replaces the FA subcommand of the PA command.
PA,c	- Parameter setup. This is the same as the ADCHK PA command. Reference the PA command instruction sheet for use of this command.
PD,lu	- Parameter Display of all currently selected channels. This is the same as the ADCHK PD command. Reference the PD command instruction sheet for use of this command.
SE	- SElected channels change. This is the same as the ADCHK SE command. Reference the SE command instruction sheet for use of this command.
SP	- Save Parameters of all currently selected channels in an FMGR file. Reference the SP command instruction sheet for use of this command.
SW,c1,c2	- SWap channel c1's and channel c2's parameters.

When the PM command is initiated, the command list is displayed along with a list of all currently selected channels. After executing each PM command, the list of currently selected channels is displayed again.

EXAMPLE:

```
>PM
          PRMAN COMMANDS

      EX        - EXit PRMAN
      GP        - Get Parameters from FMGR file
      LI,lu     - LIst PRMAN commands
      MP,p,c    - Move Parameter p of chnl c to selected chnls
      MS,c      - Move chnl c to Selected chnls
      PA,c      - Parameter setup
      PD,lu     - Parameter Display of selected chnls
      SE        - SElected chnls change
      SP        - Save Parameters of selected chnls in FMGR file
      SW,c1,c2  - SWap chnl 1 & chnl 2 parameters

    LU 19 CHANNELS SELECTED:  1  2  4

          [Current default lu is 19.  Channels 1, 2 & 4 are]
          [currently selected.                              ]

PRMAN>SW,1,4

          [Swap all parameters on channels 1 and 4.]

    LU 19 CHANNELS SELECTED:  1  2  4

PRMAN>MP,5,2

          [Parameter 5 (threshold) from channel 2 is copied to]
          [channels 1 and 4 also.                              ]

    LU 19 CHANNELS SELECTED:  1  2  4

PRMAN>MS,4

          [All parameters from channel 4 are copied to channels]
          [1 and 2.                                            ]

    LU 19 CHANNELS SELECTED:  1  2  4

PRMAN>EX
 >
```

PU COMMAND

COMMAND PURPOSE: To PUrge a data file directory of all entries.

COMMAND FORMAT: PU

COMMAND DESCRIPTION:
A data lu directory can hold only 127 entries. If a directory is full or if it is desired to delete all entries for the start of a new test, the PU command will accomplish this function. The PU command works on the current default data lu only. The command first displays the current default data lu, then asks if the purge is desired. If an answer other than Y or N is input, the display will be repeated until a valid answer is obtained.

EXAMPLES:

```
>PU
  PURGE DIRECTORY ON LU 19 (Y OR N)?N

>
```

[Current default data lu is 19. It was not purged.]

```
>PU
 PURGE DIRECTORY ON LU 20 (Y OR N)?Y
DIRECTORY PURGED

>
```

[Current default data lu is 20. It was purged.]

```
>PU
 PURGE DIRECTORY ON LU 19 (Y OR N)?U
 PURGE DIRECTORY ON LU 19 (Y OR N)?Y
DIRECTORY PURGED

>
```

[Invalid entry results in repeat of question.]

RA COMMAND

COMMAND PURPOSE: Read from a data record All channels parameters.

COMMAND FORMAT: RA,data disc lu,record start track,list lu

```
data disc lu          - default = current default lu
                        range = 18-20

record start track    - default = last record in
                                  directory
                        range = 2-201 for 18
                                2-1022 for 19 & 20

list lu               - default = operator terminal
                        range = 1, 3-6, 8, 9
```

COMMAND DESCRIPTION:
Parameters in a data record can be read on a channel to channel basis using the RP command. To display all of the parameters in a data record the RA command may be used. The command input parameters of data disc lu and record start track allow any record on disc to be displayed. The display may be sent to any valid list device using the command input parameter of list lu. Only eighteen of the possible twenty-seven parameters are displayed. The nine parameters not listed are labels or documentation parameters. The essential data acquisition parameters are displayed. The first items displayed by the command are the disc lu and track which have been read. No modification of the displayed parameters is supported by the RA command. Only channels selected to take data will appear in the parameter list. If an invalid input parameter is entered, the command is aborted and an error message is sent to the operator. If a valid track is entered, but it is not the data record start track, the display may be filled with garbage.

EXAMPLE:

>RA

PARAMETERS FROM LU 19, TRACK 195

CH	SEL	MAS	SLA	XTR	THRS	SLP	RUN DLY	STP DLY	PRE TRG	FIX	MEM SIZ	TYP	MOD	BAL	FREQ KHZ	RATE	GAGE FACTOR	GAIN
0	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.7031E+00	2.
1	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.1035E+03	10.
2	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.9980E+02	20.
3	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.1303E+03	50.
4	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.1347E+03	20.
5	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.9860E+02	40.
6	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.1051E+03	50.
7	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.1004E+03	20.
8	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.9810E+02	40.
9	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.1071E+03	50.
13	Y	N	N	N	6.0	+	0	200	0	N	16	B	X		40.0		.1114E+03	50.
14	Y	Y	N	N	6.0	+	0	50	0	N	16	C	X		40.0		.6873E+02	5.
15	Y	N	Y				0	50	0	N	16	S	X		40.0		.0000E+00	250.

[Parameters from last record of directory of current default lu are]
[displayed on operator terminal.]

>RA,17,131
IMPROPER COMMAND FORMAT**TRY AGAIN >

[Invalid parameter entry. Command aborted.]

RC COMMAND

COMMAND PURPOSE: Read Comments annotation for a data record channel.

COMMAND FORMAT: RC,data disc lu,record start track,channel number,
list lu

data disc lu	- default = current default lu range = 18-20
record start track	- default = last record in directory range = 2-201 for 18 2-1022 for 19 & 20
channel number	- default = 0 range = 0 to last channel
list lu	- default = operator terminal range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
The RC command allows the operator to read comments about a channel from a data record. No modifications of the comments are supported by the RC command. Using the data disc lu and record start track parameters any data record on disc may be annotated. The initial channel desired can be set by the channel number parameter. Any valid list device may be selected using the list lu parameter. If an invalid parameter is entered, the command is aborted and an error message is sent to the operator. All 128 possible characters are displayed even if they are not used. A | is displayed to mark the end of the available space for character input. If an operator command error is detected, an error message is displayed. If a channel which was not selected is requested, a warning message is displayed.

EXAMPLE:

```
>RC

 CHNL  0  COMMENTS:   FROM LU 19   TRACK  231

      CABLE SNAPPED DURING RECOIL.  FIRST 40 MSEC OF RECORD IS|
      OK.  IGNORE REST.

  ENTER - new chnl #      F<inish> :2
   [Opportunity to go to another channel or finish RC processing.]
  CHANNEL  2 NOT SELECTED

  ENTER - new chnl #      F<inish> :22
   [Invalid channel number will result in repeat of instructions.]

  ENTER - new chnl #      F<inish> :F

>RC,18,231
IMPROPER COMMAND FORMAT**TRY AGAIN   >
   [Invalid parameter entry.  Command aborted.]
```

RE COMMAND

COMMAND PURPOSE: REstore a data disc from a tape backup.

COMMAND FORMAT: RE

COMMAND DESCRIPTION:
The RE command allows a copy of disc data records to be restored from a backup tape. The RE command works on the current default data lu. The header written to tape by the BU command is displayed for operator inspection. The operator must then determine whether to perform the restore or not.

The RE command does not perform any tape positioning. Thus, if multiple restores are to be made on the same tape, the operator must manually perform tape positioning.

EXAMPLE:

```
>RE

     RESTORE OF LU 19     TRACKS 1 -  233

HEADER:
BCKUP LU 19 BTST  900 RNDS 27NOV83 WRMA -  99* HOLZINGER  TRKS 1 -  233
HOT ROUNDS

PERFORM RESTORE(YE OR NO)?YE

  [Tape is copied to disc.]

>
```

RH COMMAND

COMMAND PURPOSE: Read Header from a data record.

COMMAND FORMAT: RH,data disc lu,record start track,list lu

data disc lu	- default = current default lu range = 18-20
record start track	- default = last record in directory range = 2-201 for 18 2-1022 for 19 & 20
list lu	- default = operator terminal range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
The header created by the HE command is automatically stored with the data record when it is created. If it is necessary to read a data record header then the RH command may be used. Using the data disc lu and record start track parameters it is possible to read the header from any data record stored on disc. The data disc lu and record start track used are displayed first. Then the nine items in the header are displayed. If invalid parameters are entered, the command is aborted and an error message is sent to the operator. However, if a valid track is entered but it is not the start track for a record, the display may be filled with garbage.

EXAMPLE:

```
>RH
     DATA LU 19      TRACK 195
1.   1     |         RND NO
2.   CHEATER  |      TEST DIR
3.         |         SCHED NO
4.   4.2"   |        WEAPON
5.         |         TUBE NO
6.   1000|           TRAILER NO
7.   DWC|            OPR INITIALS
8.   25MAR82|       DATE(DDMMMYY)
9.  329A1   INCREMENT 41

>
```

[Header from default lu 19, track 195 is displayed.]

```
>RH,18,2
     DATA LU 18      TRACK    2
1.   1     |         RND NO
2.   KURIATA   |      TEST DIR
3.         |         SCHED NO
4.   155mm |         WEAPON
5.         |         TUBE NO
6.   700 |           TRAILER NO
7.   WCN|            OPR INITIALS
8.   14NOV80|       DATE(DDMMMYY)
9.  WARMER

>
```

[Header read from specified lu & track is displayed.]

```
>RH,21
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

[Invalid parameter entry. Command aborted.]

RN COMMAND

COMMAND PURPOSE: Read Notes from a data record.

COMMAND FORMAT: RN,data disc lu,record start track,list lu

data disc lu	- default = current default lu range = 18-20
record start track	- default = last record in directory range = 2-201 for 18 2-1022 for 19 & 20
list lu	- default = operator terminal range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
The RN command allows the operator to read notes from a data record. No modifications of the notes are supported by the RN command. Using the data disc lu and record start track parameters any data record on disc may be read. The results may be sent to any valid list device using the list lu parameter. If an invalid parameter is entered, the command is aborted and an error message is sent to the operator. All 128 possible characters are displayed even if they are not used. A | is displayed to mark the end of the available space for character input. There are no warning or error messages associated with the RN command.

EXAMPLE:

```
>RN

                    NOTES:   FROM LU 19   TRACK  231

      GUN BRACKET BROKE TEARING ALL CABLES OFF.  ONLY FIRST 40|
      MSEC OF DATA IS VALID.

>RN,18,231
IMPROPER COMMAND FORMAT**TRY AGAIN    >
   [Invalid parameter entry.  Command aborted.]
```

RP COMMAND

COMMAND PURPOSE: Read Parameters from a disc data record.

COMMAND FORMAT: RP,data disc lu,record start track,list lu

data disc lu - default = current default lu
range = 18-20

record start track - default = last record in directory
range = 2-201 for 18
2-1022 for 19 & 20

list lu - default = operator terminal
range = 1, 3-6, 8, 9

COMMAND DESCRIPTION:
A copy of the parameters used to setup the hardware is placed in the data record when it is created by the TR command. The RP command provides a means for reading these parameters from a data record. No modifications or changes of the parameters are supported by the RP command. Using the data disc lu and record start track parameters it is possible to read the parameters from any data record on disc. The channel number, data disc lu and record start track are displayed first. Then the twenty-seven parameters for that channel are displayed in the same format used by the PA command(see the PA command for the meanings of the various parameters). If an invalid data disc lu, record start track, or list device is supplied, the command is aborted and an error message is sent to the operator. However, if valid parameters are entered but they do not specify a true record start track, the list may be filled with garbage. The RP command supports two subcommands:

n - Displays the values for channel n.

EX - EXit RP command - return to ADCHK.

EXAMPLE:

```
>RP,19,272
RP>  ENTER - channel number  or  EXit:1

     [Valid command has been entered.  Valid channel number has  ]
     [been entered.  Parameters will be diplayed on terminal.    ]


     CHANNEL NO  1      LU 19      TRACK  272

  #    VALUE   PARAMETER       #       VALUE      PARAMETER
 ---   -----   ---------      ---   -----------   ---------
  1.       Y   SELECTED        13.             C   MODE
  2.       N   MASTER          14.                 BALANCE
  3.       N   SLAVE           15.         -20.0   FREQ(KHZ)
  4.       Y   XTRIG           16.             D   RATE T/C
  5.           THRSHLD V       17.    .8475E+01    GAGE FCTR
  6.           SLOPE           18.  1.0000E+01    EXCT/INTR
  7.       0   RUNDLY          19.   .8000E+05    PEAK SGNL
  8.     130   STPDLY          20.          PS/PCB GF UNITS
  9.       0   PRETRIGR        21.             5   GAIN
 10.       Y   FIX RATE        22.        KISTLE   XDCR MFG
 11.      16   MEM SIZE        23.          6211   XDCR MDL#
 12.       D   DATA TYPE       24.        168649   XDCR SN
 25. PLOT LABEL:CHAMBER PRESSURE; REAR GAGE
 26. Y-AXIS LBL:PRESSURE IN PSI
 27. RMRKS:KISTLER,6211,K=8.333,Y=1004,SENSITIVTY=0.1180

RP>  ENTER - channel number  or  EXit:99
INVALID CHANNEL ENTRY
```

```
RP>  ENTER - channel number  or  EXit:EX
>
```

[Invalid channel number was entered.]

```
>RP,18,1,1
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

[Invalid parameter entry. Command aborted.]

RT COMMAND

COMMAND PURPOSE: Rate Table changes.

COMMAND FORMAT: RT,rate table number

rate table number - default = 0
range = 0-19

COMMAND DESCRIPTION:

The RT command controls the display & update of rate tables. Rate tables are numbered 0 through 19. Each rate table contains the information required to set up the ASRD in variable sampling rate mode: decrease delay, variable length shift register length, scalar, and entries from which the rate table itself can be formed. The rate table stored in the ASRD contains 256 rate values. Rather than storing all of these in the operator accessible tables, only the start address and rate for that address are saved. The program checks that the rates and addresses increase but no restrictions on their spacing are implemented. Separate rate tables are stored on each data lu. Tables 0-15 are the standard rate tables for normal ballistic data acquisition. Usually it is not necessary to modify them; however it can be done if necessary. Tables 16-19 should normally be set up with non-standard rate tables if required for a test. All entries are checked for proper range. If errors are found the value is ignored and an error message is sent to the operator. If an invalid rate table parameter is entered, the command will be aborted and an error message is sent to the operator.

The rate table subroutine supports the following commands:

R,n	Replaces the old rate table values with the currently displayed values, then reads and displays rate table n.
F	Finished - replace the old rate table values with the currently displayed values.
K	Kill - disc values are not changed.
e,a,r	Update the currently displayed rate and address of entry "e" with new rate "r" and start address "a". If there is currently no entry "e", will add a new entry to table. If "e" is negative, will remove entry "e" and any higher entries. (range of e: 1 to n+1 where n = current # of table entries - maximum of 8 entries allowed, range of a: 0-255, range of r: 2-F)
D,delay	Update the currently displayed decrease delay with the new delay value(range =1-4094).
L,length	Update the currently displayed variable length shift register value with the new length(range = 1-64).
S,scalar	Update the currently displayed scalar value with the new scalar(range = 0-2).

EXAMPLE:

```
>RT

              RATE TABLE  0 on LU 19

     SCALAR 0         LENGTH 64        DELAY 1460

 ENTRY# ST ADRS RATE
 ------ ------- ----
      1       0    5
      2       4    6
      3       8    7
      4      16    8
      5      32    9
      6      64    A
      7     128    B
ENTER: Entry#,adrs,rate  D,dly  L,lngth  S,sclr  R,tbl  Finish  Kill
#K
       [Default to rate table 0.  This is the standard table for]
       [5kHz data.  No changes were made.                        ]
>
```

```
>RT,19

              RATE TABLE 19 on LU 19

     SCALAR 1         LENGTH 64        DELAY 2048

 ENTRY# ST ADRS RATE
 ------ ------- ----
      1       0    3
      2     100    4
      3     200    5
ENTER: Entry#,adrs,rate  D,dly  L,lngth  S,sclr  R,tbl  Finish  Kill
       [Specified table is displayed.]
#2,210,6
       [Valid entry will result in entry # 2 changing to new]
       [value.  Current rate table display is erased and new]
       [display is written. # prompt signifies ready for new]
       [entry.                                              ]
#D,3072
       [Valid entry will result in DELAY changing to new    ]
       [value.  Current rate table display is erased and new]
       [display is written. # prompt signifies ready for new]
       [entry.  For clarity this new display is listed below]
            RATE TABLE 19 on LU 19

     SCALAR 1         LENGTH 64        DELAY 3072

 ENTRY# ST ADRS RATE
 ------ ------- ----
      1       0    3
      2     210    6
      3     200    5
ENTER: Entry#,adrs,rate  D,dly  L,lngth  S,sclr  R,tbl  Finish  Kill
#F
INCONSISTANCIES FOUND - CORRECT RATE  ADRS
       [Attempt to finish is aborted due to inconsistancies.]
       [Errors must be corrected before table can be saved. ]
#2,100,4
       [Valid entry will result in entry # 2 changing to new]
       [value.  Current rate table display is erased and new]
       [display is written. # prompt signifies ready for new]
       [entry.                                              ]
#4,210,6
       [Valid entry will result in addition of entry 4 to   ]
       [table.  Current rate table display is erased and new]
       [display is written. # prompt signifies ready for new]
       [entry.  For clarity this new display is listed below]
```

```
                RATE TABLE 19 on LU 19

      SCALAR 1          LENGTH 64          DELAY 3072

  ENTRY# ST ADRS RATE
  ------ ------- ----
       1       0    3
       2     100    4
       3     200    5
       4     210    6
ENTER: Entry#,adrs,rate  D,dly  L,lngth  S,sclr  R,tbl  Finish  Kill
#F
        [Currently displayed table values are saved.  Rate    ]
        [table 19 has been modified.                          ]
>
```

```
>RT,19

                RATE TABLE 19 on LU 19

      SCALAR 1          LENGTH 64          DELAY 3072

  ENTRY# ST ADRS RATE
  ------ ------- ----
       1       0    3
       2     100    4
       3     200    5
       4     210    6
ENTER: Entry#,adrs,rate  D,dly  L,lngth  S,sclr  R,tbl  Finish  Kill
#-4
        [Valid entry will result in entry # 4 being deleted. ]
        [Current rate table display is erased and new display]
        [is written.  # prompt signifies ready for new entry.]
#G
           IMPROPER COMMAND -- TRY AGAIN!
        [Invalid entry causes error message.  Cursor moves up]
        [to entry and waits for new input.                    ]
#F
>
```

```
>RT,99
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

[Invalid parameter entry. Command aborted.]

NOTES: The RT command does not support a list lu. Therefore, a copy of a rate table can not be made using the RT command. However, program LIRT can be used to list a rate table to any valid output device. The run specification is:

```
RU,LIRT,intrctv lu(dflt=1),list lu(dflt=intrctv),
                disc lu(dflt=intrctv)
```

The output format is similar to that of the RT command.

If the standard rate tables are destroyed or if changes have been made to them, it is possible to restore them by using program TBLGN. This program generates all of the standard tables and will write them to disc if required. The run specification is:

```
RU,TBLGN,intrctv lu(dflt=1),list lu(dflt=intrctv lu)
```

The TBLGN output is given below for reference.

ASRD RATE TABLES

TBL NO	SCALAR	LENGTH	DECR DLY	ENTRY1 STADR/RATE	ENTRY2 STADR/RATE	ENTRY3 STADR/RATE	ENTRY4 STADR/RATE	ENTRY5 STADR/RATE	ENTRY6 STADR/RATE	ENTRY7 STADR/RATE
0	0	64	1460	0/5	4/6	8/7	16/8	32/9	64/A	128/B
1	0	32	720	0/6	4/7	8/8	16/9	32/A	64/B	128/C
2	0	21	400	0/7	5/8	10/9	21/A	42/B	84/C	168/D
3	0	16	400	0/7	4/8	8/9	16/A	32/B	64/C	128/D
4	1	25	240	0/8	6/9	12/A	25/B	50/C	100/D	200/E
5	1	21	240	0/8	5/9	10/A	21/B	42/C	84/D	168/E
6	0	9	240	0/8	4/9	9/A	18/B	36/C	72/D	144/E
7	0	8	240	0/8	4/9	8/A	16/B	32/C	64/D	128/E
8	0	7	160	0/9	7/A	14/B	28/C	56/D	112/E	224/F
9	1	13	160	0/9	6/A	13/B	26/C	52/D	104/E	208/F
10	0	6	160	0/9	6/A	12/B	24/C	48/D	96/E	192/F
11	1	11	160	0/9	5/A	11/B	22/C	44/D	88/E	176/F
12	0	5	160	0/9	5/A	10/B	20/C	40/D	80/E	160/F
13	1	9	160	0/9	4/A	9/B	18/C	36/D	72/E	144/F
14	0	4	160	0/9	4/A	8/B	16/C	32/D	64/E	128/F
15	0	4	160	0/9	4/A	8/B	16/C	32/D	64/E	128/F

RU COMMAND

COMMAND PURPOSE: RUn a program from ADCHK.

COMMAND FORMAT: RU,program name,p1,p2,p3,p4,p5

program name - any valid loaded program name

p1 - default = operators terminal
range = -32768 to 32767

p2-p5 - default = 0
range = -32768 to 32767

COMMAND DESCRIPTION:
The RU command allows most data analysis, utility, and system programs to be run from ADCHK without exitting. Programs which require ASCII parameters to be passed in can not be scheduled from ADCHK(ie, RU,FTN4,&DUMMY,0,-). When a program is scheduled by the RU command, ADCHK waits for the scheduled program to complete before it resumes its own execution. If invalid parameters are entered, the command will be aborted and an error message is sent to the operator. If the program is not loaded, then a scheduling error message is sent to the operator terminal.

EXAMPLE:

```
>RU,FILTR,,19,20,1
  FILE  1 FROM LU 19 TRANSFERED TO FILE  1 ON LU 20

>
```

[Program FILTR is scheduled. p1&p5 are defaulted.]
[p2-p4 are specified.]

```
>RU,FILTR
  ENTER SOURCE DISC LU, DEST DISC LU, & FILE# TO XFER:19,20,5
  FILE  5 FROM LU 19 TRANSFERED TO FILE 2 ON LU 20

>
```

[Program FILTR is scheduled. All parameters]
[are defaulted.]

```
>RU,FTN4,&CA316,0,-
IMPROPER COMMAND FORMAT**TRY AGAIN   >
```

[ASCII parameters can not be passed.]

```
>RU,DUMMY
PROGRAM DUMMY  SCHEDULING ERROR: SC05
```

[Program DUMMY is not loaded on the system, so a]
[scheduling error is returned.]

```
>
```

SE COMMAND

COMMAND PURPOSE: SElect/deselect channels in the parameter list.

COMMAND FORMAT: SE

COMMAND DESCRIPTION:
The SE command allows channels to be selected or deselected without accessing the individual channels using the PA command. The command operates on the current default data lu only. The operations supported by the SE command are:

DA	Deselect all channels in the parameter list.
EX	EXit from SE processing and return to ADCHK.
SA	Select all channels in the parameter list.
n	Select channel n in the parameter list.
-n	Deselect channel n in the parameter list.

If more than one channel is to be selected or deselected at a time, the channel numbers may be entered at one time by separating them by commas. The order of entry is not important. If a channel is entered more than once, the final entry determines the end result. A DA, EX or SA may not be entered at the same time as a channel number.

EXAMPLE:

```
>SE

    LU 19 CHANNELS SELECTED:  4  5  6  7

 ENTER: SA-sel all, DA-desel all, n-sel ch, -n-desel ch, EX-EXit
SE>0,1,2,3,-7

    LU 19 CHANNELS SELECTED:  0  1  2  3  4  5  6

 ENTER: SA-sel all, DA-desel all, n-sel ch, -n-desel ch, EX-EXit
SE>DA
  NO CHANNELS ARE CURRENTLY SELECTED ON LU 19

 ENTER: SA-sel all, DA-desel all, n-sel ch, -n-desel ch, EX-EXit
SE>SA

    LU 19 CHANNELS SELECTED:  0  1  2  3  4  5  6  7

 ENTER: SA-sel all, DA-desel all, n-sel ch, -n-desel ch, EX-EXit
SE>-0,-7,-5

    LU 19 CHANNELS SELECTED:  1  2  3  4  6

 ENTER: SA-sel all, DA-desel all, n-sel ch, -n-desel ch, EX-EXit
SE>EF
  INVALID ENTRY --- TRY AGAIN
SE>EX

>
```

SP COMMAND

COMMAND PURPOSE: Save Parameters in an FMGR file.

COMMAND FORMAT: SP

COMMAND DESCRIPTION:
Once all the parameters are entered onto the scratch parameter track, it may be desireable to save the parameters. The SP command allows the parameters to be saved in an independent file stored on an FMGR lu. Thus, they may be transfered to another trailer or used as a backup for later tests. The SP command works on the current default lu. The file name where the parameters are to be saved is requested first. The file name must begin with an *. The file created is a type one file which can NOT be modified on the editor. Next the file name and a list of the currently selected channels are displayed and the operator is given an opertunity to terminate if the name or channels are not correct. Next the operator is given an opertunity to enter a 64 character label for identification purposes. If any FMGR errors are encountered, an error message is sent indicating the FMGR error code and the command will be terminated. Note that the SP command is executed as a subcommand of PRMAN and may not be executed directly from ADCHK.

EXAMPLE:

```
PRMAN>SP

 ENTER NEW FILE NAME(FIRST CHARACTER MUST BE *):*BRKT1
  LU 19 CHANNELS TO BE SAVED IN FILE *BRKT1 ARE:
       1    2    4

 OK TO PROCEED(YE OR NO)?YE
 ENTER FILE LABEL:
BRACKET TEST 19NOV83 BTST 1000

    LU 19 CHANNELS SELECTED:    1    2    4

PRMAN>SP

 ENTER NEW FILE NAME(FIRST CHARACTER MUST BE *):*BRKT1
  LU 19 CHANNELS TO BE SAVED IN FILE *BRKT1 ARE:
       1    2    4

 OK TO PROCEED(YE OR NO)?YE

   ERROR IN CALL TO 'CREAT'.  IERR =    -2

     [FMGR error code -2 = duplicate file name.  Name *BRKT1 has]
     [already been used.                                        ]

    LU 19 CHANNELS SELECTED:    1    2    4

PRMAN>
```

ST COMMAND

COMMAND PURPOSE: Obtain Self Test results for the ASRD.

COMMAND FORMAT: ST,list lu

```
                    list lu - default = operators terminal
                              range = 1, 3-6, 8, 9
```

COMMAND DESCRIPTION:

When the ASRD is powered-up, it executes an internal self test of its registers, memory and clock. It also determines the amount of memory installed in the channel. The ST command allows this information to be read from the ASRD and examined by the operator. The operator should check the ST command listing for errors and determine that the listed memory size agrees with the installed size. Any channel which has errors or incorrect memory size should be retested and then repaired or replaced. Retesting can be accomplished using program RMTST. If an improper list lu is entered, the command will be aborted and an error message sent to the operator.

If an ADCHK LO command is used before the ST command is used, then the results will not be valid. The ST command should be the first thing done to the digitizer after power-up. Note that approximately one minute of test time is required by the ASRD for each 16k block of memory. So adequate time must be allowed after power-up before the ST command can be used. When all channels have a display of channel number and memory size it is safe to use the ST command.

EXAMPLE:

```
>ST
********** REGISTER/MEMORY TEST **********

CHANNELS TESTED:   0   1   2   3   4   5   6   7
*****  CHANNEL  0     MEMSIZ =  16K     # ERRORS =  0
*****  CHANNEL  1     MEMSIZ =  16K     # ERRORS =  0
*****  CHANNEL  2     MEMSIZ =  64K     # ERRORS =  0
*****  CHANNEL  3     MEMSIZ =  64K     # ERRORS =  0
*****  CHANNEL  4     MEMSIZ =  16K     # ERRORS =  0
*****  CHANNEL  5     MEMSIZ =   0K     # ERRORS = 18
            DECREASE DELAY REGISTER ERROR
            TRIGGER MODE AND THRESHOLD REGISTER ERROR
            STOP DELAY REGISTER ERROR
            START DELAY REGISTER ERROR
            CALCULATION CONTROL REGISTER ERROR
            RATE TABLE ADDRESS REGISTER ERROR
            RATE TABLE DATA ERROR
            MEMORY ADDRESS REGISTER (0-15) ERROR
            MEMORY ADDRESS REGISTER (16) ERROR
            PRETRIGGER REGISTER (0-15) ERROR
            PRETRIGGER REGISTER (16) ERROR
            END OF MEMORY REGISTER ERROR
            FAST MEMORY TEST ERROR
            MEMORY ERROR--ALL ZERO'S FAILURE
            MEMORY ERROR--NONSTABLE ZERO
            MEMORY ERROR--WALKING BIT TEST
            MEMORY ERROR--ALL ONE'S FAILURE
            SAMPLE RATE CLOCK ERROR
            FIRST FAILED MEMORY ADDRESS:   3243.   CONTENTS: 000020
*****  CHANNEL  6     MEMSIZ =  16K     # ERRORS =  0
*****  CHANNEL  7     MEMSIZ =  16K     # ERRORS =  0

    [All channels except 5 have correct memory size and no]
    [errors.  Channel 5 is shown with all possible error  ]
    [messages - a situation which could not really occur. ]

>
```

```
>ST,7
IMPROPER COMMAND FORMAT**TRY AGAIN    >

      [Invalid parameter entry.  Lu 7 is not a valid list]
      [device.                                           ]
```

TR COMMAND

COMMAND PURPOSE: TRansfer data from the ASRD to disc.

COMMAND FORMAT: TR,data disc lu,record start track

data disc lu - default = current default lu
range = 18-20

record start track - default = next directory track
range = 2-201 for 18
2-1022 for 19 & 20

COMMAND DESCRIPTION:
Once data has been acquired in the ASRD it can be saved on disc using the TR command. The data disc lu and record start track allow the data to placed at any location on the data discs. However, if the default is used then the created record is entered in the record directory making subsequent operations easier. If the default is not used the operator must keep his own log of where the data is located. If an invalid parameter is entered, the command is aborted and an error message is sent to the operator. If the parameters are defaulted, a check is made to determine that there is still room in the directory for another entry. If no room remains, a warning message is sent to the operator and the command is terminated. Next, a check is made of the available room on the data lu. If less than 43 tracks remain, the operator is told how many tracks are left and asked if he wants to continue the transfer. If the transfer continues, the program performs the following tasks:

1. Reads and displays any overloads on the 416.
2. Reads the scratch header.
3. Reads the scratch parameters for all selected channels.
4. Reads the appropriate calibration factors and other documentation information.
5. Reads the events channel.
6. Reads the ASRD data.
7. Zeroes any remaining space not used on a track.
8. Updates the directory.

If the process is not completed, then for all practical purposes the data is not available since the directory entry is not created. During step four, if a cal factor, gage factor or excitation value results in an overall scale factor of zero, a warning message is sent to the operator. If the transfer "dies", it will be necessary to reboot the computer.

EXAMPLE:

```
>TR
BEGIN TRANSFER.

   32671 WORDS TRANSFERRED TO DISK LU 19 TRACK    206
       6 TOTAL TRACKS UTILIZED

   [Normal transfer results in new data record creating a]
   [new directory entry.  No warnings!                   ]

>
```

```
>TR
DIRECTORY FULL -- TRANSFER NOT INITIATED

>
```

```
>TR
ONLY   23 TRACKS REMAIN--CONTINUE(Y OR N)?N

>
```

```
>TR
 OVERLOADS:    07/1     11/3     14/2

BEGIN TRANSFER.
      WARNING:  CHANNEL  3 HAS A SCALE FACTOR OF ZERO
      WARNING:  CHANNEL 12 HAS A SCALE FACTOR OF ZERO
  240367 WORDS TRANSFERRED TO DISK LU 19 TRACK    857
       40 TOTAL TRACKS UTILIZED

>
```

```
>TR,17,124
IMPROPER COMMAND FORMAT**TRY AGAIN    >
```

VE COMMAND

COMMAND PURPOSE: VErify rate tables loaded into ASRD.

COMMAND FORMAT: VE,channel number

channel number - default = 0
range = 0 to last channel

COMMAND DESCRIPTION:
The rate table loaded into the ASRD can be read back to determine what values were actually loaded. This may be done with the VE command. The command reads the 256 table entries from the ASRD. Then it scans through the entries making a table consisting of the start address and rate for each group of identical rates. Finally it displays the table on the operator terminal. The operator can then determine if the rate structure is correct. If the table is filled with many errors, then only the first 8 rates in the table will be displayed. Any channel can be checked by using the channel number parameter. If an invalid parameter entry is made, the command is aborted and an error message is sent to the operator. If the channel is not selected, the channel is not read and a warning message is sent to the operator. If the channel is set for fixed rate, then the channel is read and the first entry is displayed. For fixed rate, the first entry establishes the rate. A warning message is also sent to the operator.

EXAMPLE:

```
>VE,5
            RATE TABLE FOR CHANNEL  5

 ENTRY# ST ADRS RATE
 ------ ------- ----
    1        0    5
    2        4    6
    3        8    7
    4       16    8
    5       32    9
    6       64    A
    7      128    B

     [Examination by operator is required to determine]
     [if table is correct.                             ]

>VE,8
     CHANNEL NOT SELECTED

     [Channel is not selected, so no valid setup information]
     [exists in the ASRD.  The channel is not read.          ]

>VE
          RATE TABLE FOR CHANNEL  0
     CHANNEL IS FIXED RATE - ONLY FIRST ENTRY IS DISPLAYED

 ENTRY# ST ADRS RATE
 ------ ------- ----
    1        0    8

     [Channel is set up for fixed rate, so only first entry]
     [is displayed.                                         ]

>VE,55
IMPROPER COMMAND FORMAT**TRY AGAIN   >

     [Invalid parameter entry.  Command aborted.]
```

APPENDIX L - ADCHK DATA RECORD FORMAT

This document provides a description of the digital data record which is produced by a BTST operating with version III of ADCHK.

Since the number of channels and the number of words per channel can vary, a data record has no set number of words. In the BTST a record is stored on disc with the first word of the record being the first word of the track. No use is made of the space from the end of a record to the the end of the track(however, these words are all set to zero). If a channel is not selected by the operator it will not appear anywhere in the data record. If a channel is selected, but no data is taken by that channel, then the channel header will appear in the data record but there will be no data entry for the channel.

The general sequence of the record is as follows:

```
ROUND HEADER
CHANNEL POINTERS
FILE NOTES
SPARE
DOCUMENTATION INFORMATION FOR FIRST SELECTED CHANNEL
      :
      :
DOCUMENTATION INFORMATION FOR LAST SELECTED CHANNEL
EVENTS CHANNEL DATA
DATA WORDS FOR FIRST SELECTED CHANNEL
      :
      :
DATA WORDS FOR LAST SELECTED CHANNEL
REMAINDER OF TRACK IS FILLED WITH ZEROES
```

ROUND HEADER FORMAT - SECTOR 0 OF DATA FILE:

Word	Contents
1 & 2	This is a floating point number which is a count of the number of words in the record including round header, channel pointers, channel doumentation and channel data.
3	Integer word which contains the location of the EVENTS channel data relative to the start of the data record.
4	Not used at this time. Currently set to zero.
5 thru 64	ROUND HEADER information in ASCII format taken from sector 6 of track 1 of the data lu. See below for specific usage of these words. This information can be read using the RH command of ADCHK.

CHANNEL POINTERS FORMAT - SECTOR 1 OF DATA FILE:

word	channel	
1,2	0	The pointer contains a real word which is the location of the start of the channel's data relative to the start of the data record. If a channel is not selected its value is -1.0. If the channel is selected but no data was taken, its value is 0.0.
3,4	1	
:	:	
63,64	31	

FILE NOTES - SECTOR 2 OF DATA FILE

This sector is filled with blanks when the data record is created. Later notes concerning this round may be entered in the record using the NO command of ADCHK. The notes may be read with the RN command of ADCHK.

SPARE - SECTOR 3 OF DATA FILE

This sector is filled with zeroes when the data record is created. This sector is set aside for future expansion if necessary.

CHANNEL DOCUMENTATION FORMAT - 4 SECTORS FOR EACH SELECTED CHANNEL:

sector	words	usage	
1,2	1-128	Setup parameters taken directly from track 0. The meanings and locations of the parameters are given below. The parameters may be read using the RP command of ADCHK. The plot labels and remarks may be modified using the ML command of ADCHK.	
3	1	Channel No.	
	2	416 GAIN read from device.	NCI
	3	416 FREQ read from device.	NCI
	4	416 OVERLOAD code read from device.	
	5	416 FILTER type read from device.	NCI
	6	316 MODE read from device.	NCI
	7	316 GAIN read from device.	NCI
	8	516 GAIN read from device.	NCI
	9	516 OFST read from device.	NCI
	10	516 OVERLOAD code read from device.	NCI
	11-22	RATE TABLE if variable sampling rate.	
	23	BASE RATE for reconstructing rate codes.	
	24-26	Not used = 0.	
	27,28	316 calibration factor.	
	29,30	416 calibration factor.	
	31,32	516 calibration factor.	
	33,34	ASRD calibration factor.	
	35,36	Scale factor.	
	37	Baseline count (set to -4096 by TRANS).	
	38	Maximum count (set to -4096 by TRANS).	
	39	Minimum count (set to -4096 by TRANS).	
	40	Not used = 0.	
	41,42	Stop word number (set to 0 by TRANS).	
	43-60	Not used = 0.	
	61	316 SN	
	62	416 SN	
	63	516 SN	
	64	ASRD SN	
		NCI=NOT CURRENTLY IMPLEMENTED	
4	1-64	Post event comments. This area is set to ASCII blanks when the data record is created. Later comments on the channel may be added using the CO command of ADCHK. The comments may be read using the RC command of ADCHK.	

EVENTS CHANNEL FORMAT:

The events channel data starts on a sector boundary. There are 40 possible events with 4 words required for each event. The first event is the time at which the ASRD was armed. It is always present. If the events channel is not used then this will be the only entry. The location of the events data can be found from word 3 of sector 0 of the first track of the data. The events data can be read by the EV command of ADCHK. The format of the events words is given below.

words	useage
1	Number of events saved
2	Days segment of time word(three BCD digits)
3,4	Not used = 0.
5-8	First event - time at which ASRD was armed.
*	Additional words will be present only if valid events occurred.

CHANNEL DATA WORDS FORMAT:

The first selected channels data words follow the last valid event's word. The location is obtained from the channel pointers in sector 1 of the first track of the data record. The number of words that a channel can take varies from 0-131070. Only data words actually taken will be saved. There are two two codes used for labelling data:

bit 15	useage
0	Data word
1	Change word or time word

Change words and time words may be differentiated:

bit 14	useage
0	Change word
1	Time word

The data words may be examined using the IN command of ADCHK. The formats for data words, change words, and IRIG time words are given below.

HEADER FORMAT:

WORDS	DESCRIPTION
5-6	ROUND NUMBER
7-11	TEST DIRECTOR
12-13	SCHEDULE NUMBER
14-16	WEAPON TYPE
17-19	TUBE NUMBER
20-21	TRAILER NUMBER
22-23	OPERATORS INITIALS
24-27	DATE(DDMMYY)
28-64	REMARKS(74 ASCII CHARACTERS)

All header items are stored in ASCII format.

CHANNEL PARAMETER FORMAT:

The channel setup parameters are stored in the following locations in the sectors:

FILE WORD	PARAM#	DESCRIPTION	TYPE DATA	
1	1	SELECTED	A	
2	2	MASTER	A	
3	3	SLAVE	A	
4	4	XTRGR	A	
5	5	THRSHLD V	I	10*THRESHOLD
6	6	SLOPE	B	
7	7	RUN DLY	I	
8	8	STP DLY	I	
9	9	PRETRGR	I	
10	10	FXD RATE	A	
11	11	MEM SIZE	I	
12	12	TYPE DATA	A	
13	13	MODE	A	
14	14	BALANCE	A	
15	15	FREQ(KHZ)	I	10*FREQ
16	16	RATE T/C	I	
17-18	17	GAGE FCTR	R	
19-20	18	EXCT/INTR	R	
21-22	19	PEAK SGNL	R	
23-25	20	GF UNITS	B	6 characters
26-27	21	GAIN	R	
28-30	22	XDCR MFG	B	6 characters
31-33	23	XDCR MDL#	B	6 characters
34-36	24	XDCR SN	B	6 characters
37-53	25	PLOT LABEL	B	34 characters
54-68	26	Y-AXIS LBL	B	30 characters
69-102	27	RMRKS	B	68 characters
103-127		NOT USED		
128		CHANNEL NO		

DATA TYPES ARE:

- A ONE ASCII CHARACTER RIGHT JUSTIFIED(MSB=0)
- B TWO ASCII CHARACTERS PER WORD
- I INTEGER
- R REAL

RATE TABLE FORMAT:

Each table uses 12 words of storage which consists of:

Word #	Contents
1	No of rates in table
2	Scalar
3	Length of variable length shift register
4	Decrease delay before rate is reduced
5	MSB: Rate 1 LSB: Start Address
.	.
.	.
12	MSB: Rate 8 LSB: Start Address

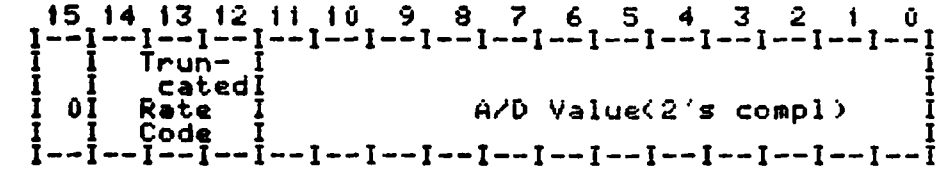
EVENT WORDS FORMAT:

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1.	0	0	0	0	0	0	0	0	E7	E6	E5	E4	E3	E2	E1	E0
2.	0	0	U512	U256	U128	U64	U32	U16	U8	U4	U2	U1	H20	H10	H8	H4
3.	0	0	H2	H1	M40	M20	M10	M8	M4	M2	M1	S40	S20	S10	S8	S4
4.	0	0	S2	S1	T800	T400	T200	T100	T80	T40	T20	T10	T8	T4	T2	T1

WHERE E=EVENT#, H=HOURS, M=MINUTES,
S=SECONDS, T=MILLISECONDS & U=MICROSECONDS

The first event is the state of the event lines at the time the system is armed(made ready to take data). Additional events are recorded any time there is a change in one of the event lines. The resolution of the time code is one microsecond, this is the resolution of the events time. Events are listed in the sequence in which they occur.

DATA WORD FORMAT IS:



Rate Codes are:

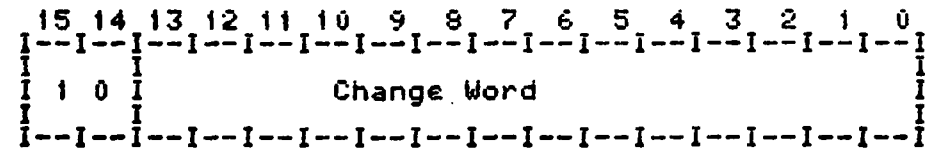
				RATE(Thou smpls/sec)	INTERVAL (microsec)
0	0	0	0	Not valid	Not valid
0	0	0	1	Not valid	Not valid
0	0	1	0	0.0977	10240.
0	0	1	1	0.1953	5120.
0	1	0	0	0.3906	2560.
0	1	0	1	0.7813	1280.
0	1	1	0	1.5625	640.
0	1	1	1	3.125	320.
1	0	0	0	6.25	160.
1	0	0	1	12.5	80.
1	0	1	0	25.	40.
1	0	1	1	50.	20.
1	1	0	0	100.	10.
1	1	0	1	200.	5.
1	1	1	0	400.	2.5
1	1	1	1	800.	1.25

While four bits are required to specify the fourteen rate codes, only three bits are used in the data word. The three bits are the three LSBs of the four required. The actual rate code must be reconstructed from the truncated rate code using the knowledge as to what rate/rates were used. For this reconstruction to work only eight consecutive codes may be used in a table.

A/D Values are:

+2047=	0	1	1	1	1	1	1	1	1	1	1	1	
+1=	0	0	0	0	0	0	0	0	0	0	0	1	
0=	0	0	0	0	0	0	0	0	0	0	0	0	
-1=	1	1	1	1	1	1	1	1	1	1	1	1	
-2048=	1	0	0	0	0	0	0	0	0	0	0	0	

CHANGE WORD FORMAT IS:



Change words are found in the data record when the variable sampling rate mode is used. Their function is to indicate the time between samples when the rate changes. Change words are inserted between the last sample taken at one rate and the first sample taken at a higher or lower rate. The time interval between the two samples is given by:

t(2)-t(1) = Change Word * 1.25 microseconds

The rate code for the sample following a change word should be ignored and the change word used to calculate the sample spacing.

TIME WORDS FORMAT:

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1.	1	1	U512	U256	U128	U64	U32	U16	U8	U4	U2	U1	H20	H10	H8	H4
2.	1	1	H2	H1	M40	M20	M10	M8	M4	M2	M1	S40	S20	S10	S8	S4
3.	1	1	S2	S1	T800	T400	T200	T100	T80	T40	T20	T10	T8	T4	T2	T1
4.	1	1	Relative trigger word													

WHERE H=HOURS, M=MINUTES, S=SECONDS, T=MILLISECONDS
& U=MICROSECONDS

Time words are inserted each time the system is triggered. Although the time words give the time at which the trigger occurred, they do not appear in the data record at the sample which represents the trigger point. The time words are inserted 100 microseconds prior to the trigger point sample. At 800k sps the time words occur 80 samples before the trigger point sample; at 400k sps, 40 samples before; etc.

Word 4 provides the time between the sample saved in memory and the sample (taken at 800000sps) which occured closest to trigger. The time interval between the two samples shall be given by:

t(2)-t(1)=(Relative trigger word - 1) * 1.25microseconds

If a change word occurs during the same master clock cycle, the change word ID bit shall replace the relative trigger word ID bit. In this case the change word is also the relative trigger word.

Using the time words it is possible to align two independently triggered, variable sample rate channels to within +/- 2.5 microseconds.

APPENDIX M - CALIBRATION PROGRAMS' OPERATING INSTRUCTIONS

SIGNAL CONDITIONER CALIBRATION

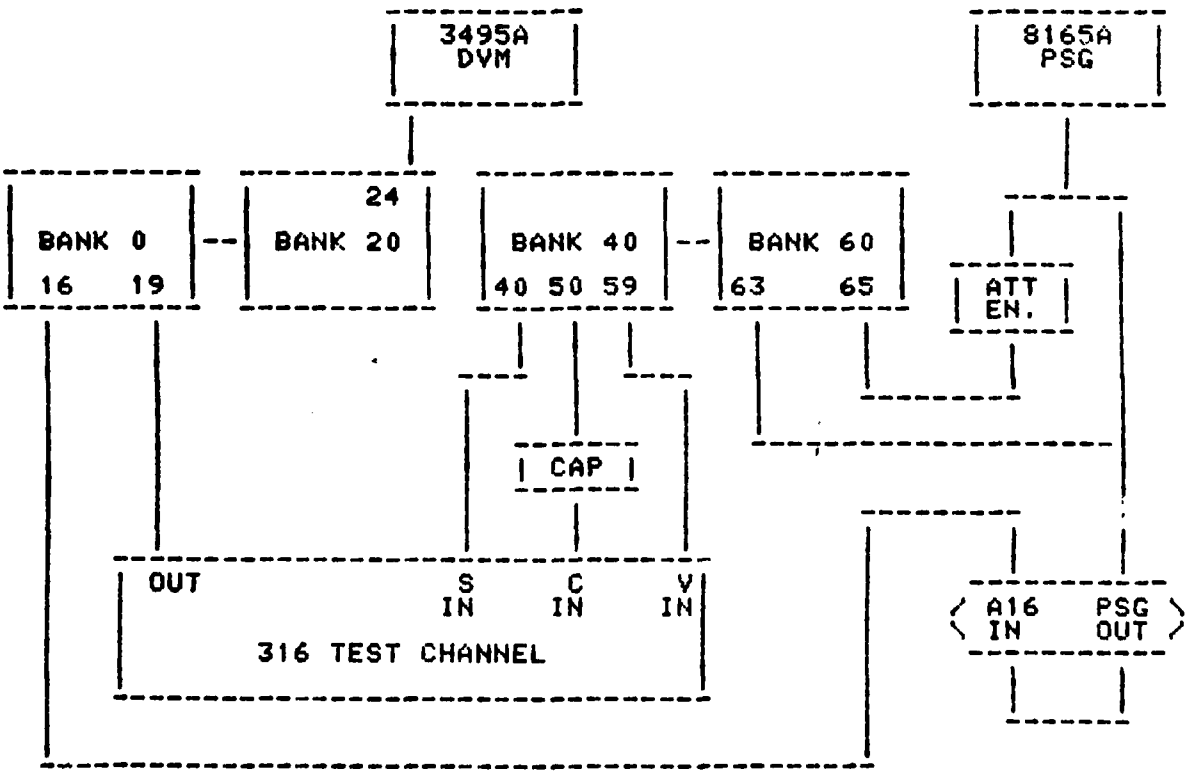
Amplitude calibration of the BTST signal conditioner is accomplished by applying a 1kHz sine wave to the signal conditioner and measuring the input and output amplitudes with a DVM. Due to the wide gain range of the signal conditioner (1-1000), an attenuator is required at the higher gains to allow an accurate measurement to be made. The DVM must be calibrated with traceability to National Bureau of Standards. For the charge mode calibration a 0.01microfarad calibration capacitor is required. The capacitor must have traceability to National Bureau of Standards. The input amplitude for the strain amplifier must be decreased as the gain is increased to prevent clipping in the signal conditioner. The approximate levels used in the calibration are:

MODE	GAIN	PSG	SC IN	SC OUT
V		9.6	9.6	9.6
C		9.6	9.6	9.6
S	1	9.6	9.6	9.6
S	10	0.96	0.96	9.6
S	100	1.54	0.096	9.6
S	1000	0.15	0.0096	9.6

After the PSG and signal conditioner have been programmed, the DVM measures the input voltage and then the output voltage. The ratio of the two readings is calculated and compared to the expected value. If the measured gain is not within 5% of the expected gain, the new value can not be written to disc.

Only one channel at a time may be calibrated. The normal BITE equipment installed in a BTST is used to perform the calibration.

The equipment connections required for performing the calibration are:



Normally the only cable connection changes required are:

1. Connect a short cable from the PSG OUT to the A16 IN jacks on the front patch panel.
2. Connect scanner cables 40, 50 and 59 to the strain, charge capacitor and voltage inputs respectively.
3. Connect scanner cable 19 to the signal conditioner output.

CA316 is an interactive command oriented program. The operator is prompted to check the current time at the start. The operator must enter the calibration factor for the charge calibration capacitor. The operator must either calibrate the attenuator or enter a calibration factor for the attenuator. The program will prompt the operator with the steps to calibrate the attenuator (if needed). The operator is prompted to change cables prior to starting each channels calibration. Although the program checks for suitability of the calibration factor as mentioned above, the operator has final control over accepting or rejecting the value.

The run string for the program is:

```
RU,CA316,intrctv lu(dflt=1)
```

The commands supported by CA316 are:

AT,x	ATtenuator value entry or change(x for bank 2 only).
CA,lu	CAlibrate channel with list to lu.
CP	CaPacitor value entry or change.
DA	DAte check or change.
DI,lu	DIsplay the dates and serial numbers of channels which have been calibrated with list to lu.
EX	EXit from CA316.
LI	LIst the CA316 commands.
PR,lu	PRint the previous CA results again with list to lu.
RE,lu	REad the calibration results with list to lu.
RU,namr,p1-5	RUn program namr with parameters p1-p5.
SC,n,x	SCanner contacts n closed(x for bank 2 only).

FILTER/AMPLIFIER CALIBRATION

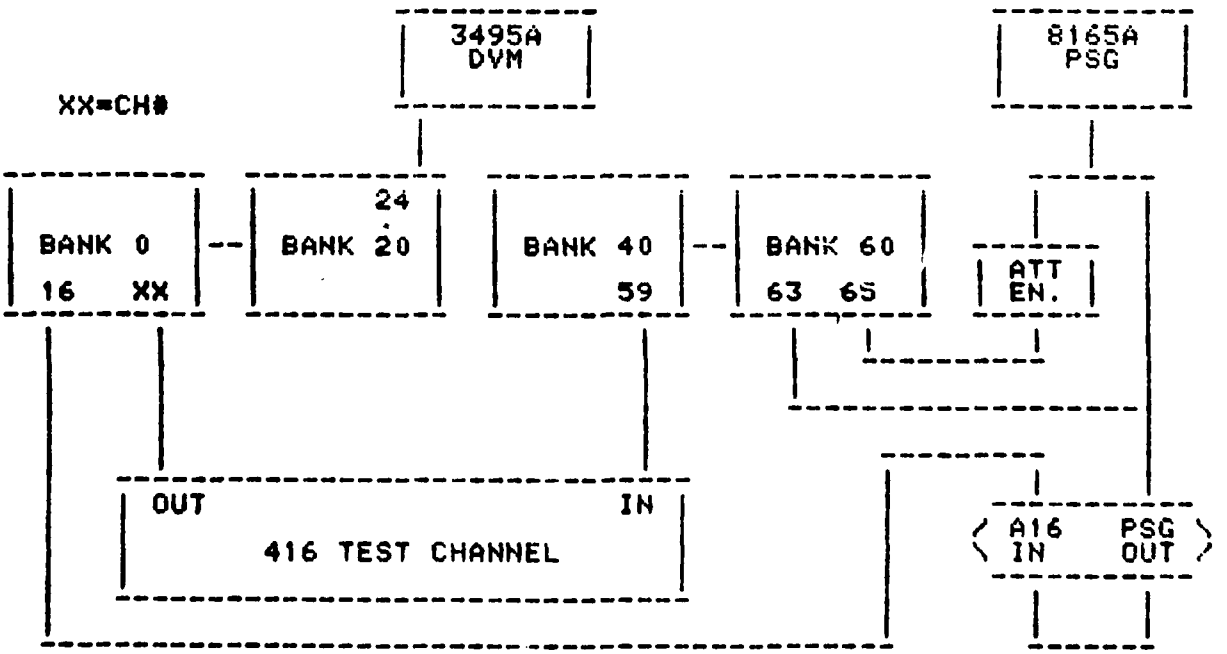
Amplitude calibration of the BTST filter/amplifier is accomplished by applying a 1kHz sine wave to the filter/amplifier and measuring the input and output amplitudes with a DVM. Due to the wide gain range of the filter/amplifier(1-1000), an attenuator is required at the higher gains to allow an accurate measurement to be made. The DVM must be calibrated with traceability to National Bureau of Standards. The input amplitude must be decreased as the gain is increased to prevent clipping in the filter/amplifier. The approximate levels used in the calbration are:

GAIN	PSG	F/A IN	F/A OUT
1	9.6	9.6	9.6
2	4.8	4.8	9.6
4	2.4	2.4	9.6
5	1.92	1.92	9.5
10	0.96	0.96	9.6
20	7.68	0.48	9.6
25	6.14	0.38	9.5
40	3.84	0.24	9.6
50	3.07	0.19	9.5
100	1.54	0.096	9.6
200	0.77	0.048	9.6
250	0.61	0.038	9.5
400	0.38	0.024	9.6
500	0.31	0.019	9.5
1000	0.15	0.0096	9.6

After the PSG and filter/amplifier have been programmed, the DVM measures the input voltage and then the output voltage. The ratio of the two readings is calculated and compared to the expected value. If the measured gain is not within 5% of the expected gain, the new value can not be written to disc.

Only one channel at a time may be calibrated. The normal BITE equipment installed in a BTST is used to perform the calibration.

The equipment connections required for performing the calibration are:



Normally the only cable connection changes required are:

1. Connect a short cable from the PSG OUT to the A16 IN jacks on the front patch panel.
2. Connect scanner cable 59 to the channel to be tested.

CA416 is an interactive command oriented program. The operator is prompted to check the current time at the start. The operator must either calibrate the attenuator or enter a calibration factor for the attenuator. The program will prompt the operator with the steps to calibrate the attenuator(if needed). The operator is prompted to change cables prior to starting each channels calibration. Although the program checks for suitability of the calibration factor as mentioned above, the operator has final control over accepting or rejecting the value.

The run string for the program is:

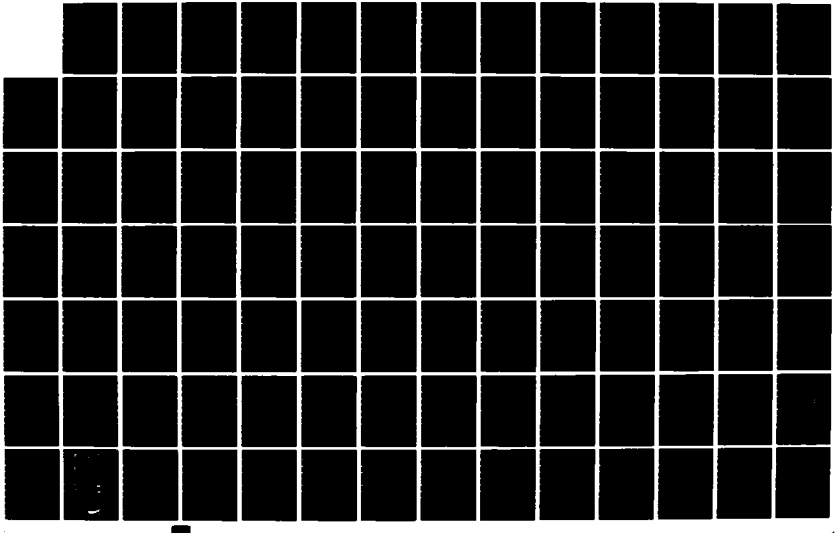
RU,CA416,intrctv lu(dflt=1)

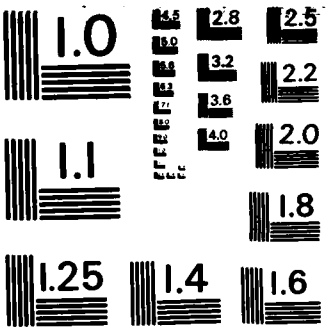
The commands supported by CA416 are:

AT,x	ATtenuator value entry or change(x for bank 2 only).
CA,lu	CAlibrate channel with list to lu.
DA	DAte check or change.
DI,lu	DIsplay the dates and serial numbers of channels which have been calibrated with list to lu.
EX	EXit from CA416.
LI	LIst the CA416 commands.
PR,lu	PRint the previous CA results again with list to lu.
RE,lu	REad the calibration results with list to lu.
RU,namr,p1-5	RUn program namr with parameters p1-p5.
SC,n,x	SCanner contacts n closed(x for bank 2 only).

AD-A139 956 RDI TASK FINAL REPORT OF RESEARCH AND DEVELOPMENT OF SOFTWARE BALLISTIC T..(U) ABERDEEN PROVING GROUND MD MATERIEL TESTING DIRECTORATE C L FRANCIS JAN 84 3/4

UNCLASSIFIED APG-MT-5952 F/G 9/2 NL



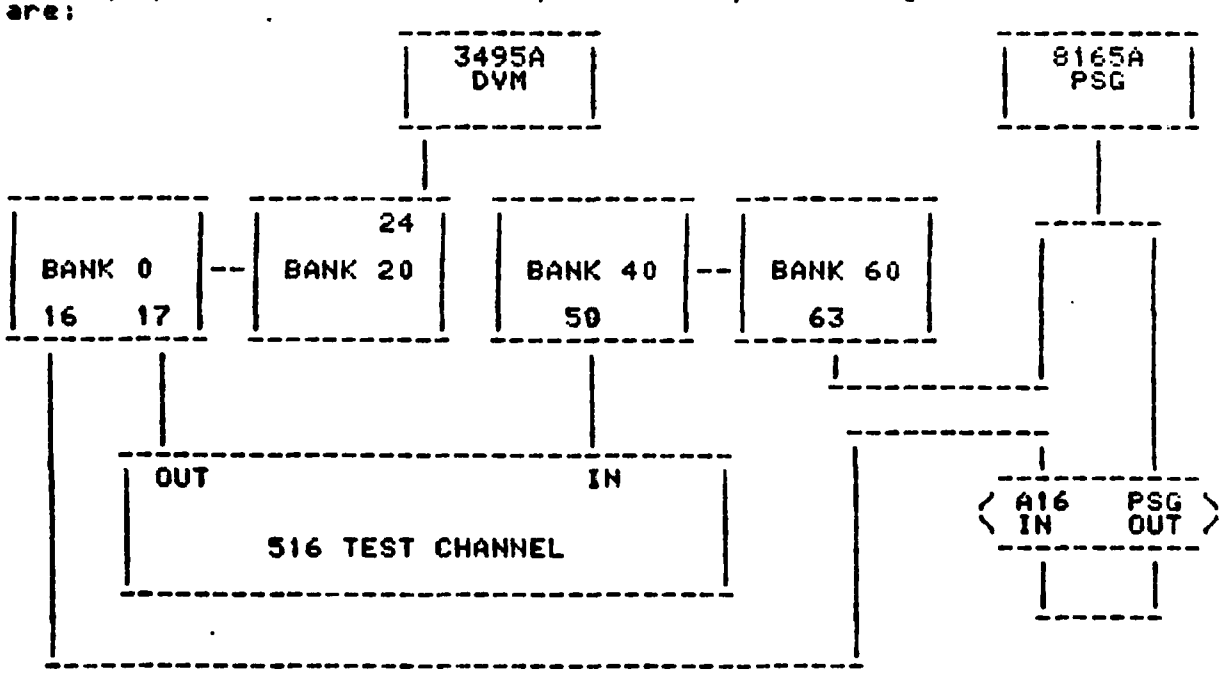


OFFSET/AMPLIFIER CALIBRATION

Amplitude calibration of the BTST offset/amplifier is accomplished by applying a 1kHz sine wave to the offset/amplifier and measuring the input and output amplitudes with a DVM. The DVM must be calibrated with traceablity to National Bureau of Standards. The input wave form has an amplitude of 8.4V p-p with an offset of 4.2V. The offset/amplifier output is then about 16.8V p-p. After the PSG and offset/amplifier have been programmed, the DVM measures the input voltage and then the output voltage. The ratio of the two readings is calculated and compared to the expected value. If the measured gain is not within 5% of the expected gain, the new value can not be written to disc.

Only one channel at a time may be calibrated. The normal BITE equipment installed in a BTST is used to perform the calibration.

The equipment connections required for performing the calibration are:



Normally the only cable connection changes required are:

1. Connect a short cable from the PSG OUT to the A16 IN jacks on the front patch panel.
2. Connect scanner cable 59 to the offset/amplifier input.
3. Connect scanner cable 17 to the offset/amplifier output.

CA516 is an interactive command oriented program. The operator is prompted to check the current time at the start. The operator is prompted to change cables prior to starting each channels calibration. Although the program checks for suitability of the calibration factor as mentioned above, the operator has final control over accepting or rejecting the value.

The run string for the program is:

```
RU,CA516,intrctv lu(dflt=1)
```

The commands supported by CA516 are:

CA,lu	CAlibrate channel with list to lu.
DA	DAte check or change.
DI,lu	DIsplay the dates and serial numbers of channels which have been calibrated with list to lu.
EX	EXit from CA516.
LI	LIst the CA516 commands.
PR,lu	PRint the previous CA results again with list to lu.
RE,lu	REad the calibration results with list to lu.
RU,namr,p1-5	RUn program namr with parameters p1-p5.
SC,n,x	SCanner contacts n closed(x for bank 2 only).

ASRD AMPLITUDE CALIBRATION

Amplitude calibration of the BTST digitizer is accomplished by applying a 1kHz sine wave simultaneously to the ASRD and a DVM. The DVM must be calibrated with traceability to National Bureau of Standards. An input amplitude of 18.8V peak-to-peak is used for the calibration signal. When the ASRD is triggered to take data, the DVM is also triggered to make a voltage measurement. Due to slightly different measuring processes, the two readings are not truly identical. However, the readings do overlap in time. The ASRD is set to record data at a 50kHz rate which gives 50 samples for each cycle of input waveform.

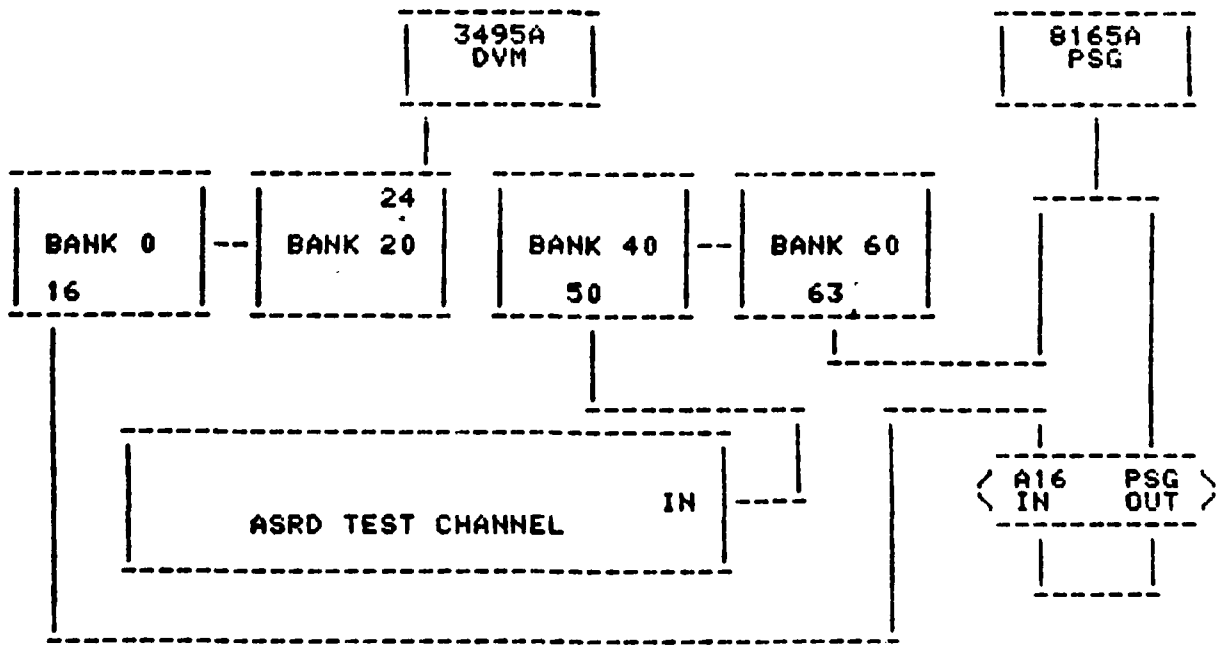
The DVM measurement is a true RMS measurement of the input waveform. A calculation of the true RMS value of the data points taken by the ASRD is performed. Approximately 6000 samples are taken by the ASRD. The RMS is calculated starting at the first positive point of a cycle at least 100 samples from the beginning. The end point is set at the first positive point of a cycle at least 5000 points from the start point. In this manner at least 100 complete cycles of input waveform are included in the RMS calculation. The voltage value read from the DVM is divided by the RMS count value of the ASRD to arrive at a calibration factor for the channel.

To prevent an improperly functioning channel from being "calibrated", several checks are made to validate the measurement. If any of the checks is failed, then the new calibration value can not be written to disc to replace the old value. The channel fails if:

1. The calibration value varies by more than 5% from the expected value of 0.00488V/count.
2. The input waveform clipped as evidenced by a data sample of -2048 or +2047.
3. The time calculated for 100 cycles does not fall in the range 0.9890 to 0.1011 seconds.

Only one channel at a time may be calibrated. The normal BITE equipment installed in a BTST is used to perform the calibration.

The equipment connections required for performing the calibration are:



Normally the only cable connection changes required are:

1. Connect a short cable from the PSG OUT to the A16 IN jacks on the front patch panel.
2. Connect scanner cable 50 to the channel to be tested.

CADIG is an interactive command oriented program. The operator is prompted to check the current time at the start. The operator is prompted to change cables prior to starting each channels calibration. Although the program checks for suitablility of the calibration factor as mentioned above, the operator has final control over accepting or rejecting the value.

The run string for the program is:

```
RU,CADIG,intrctv lu(dflt=1)
```

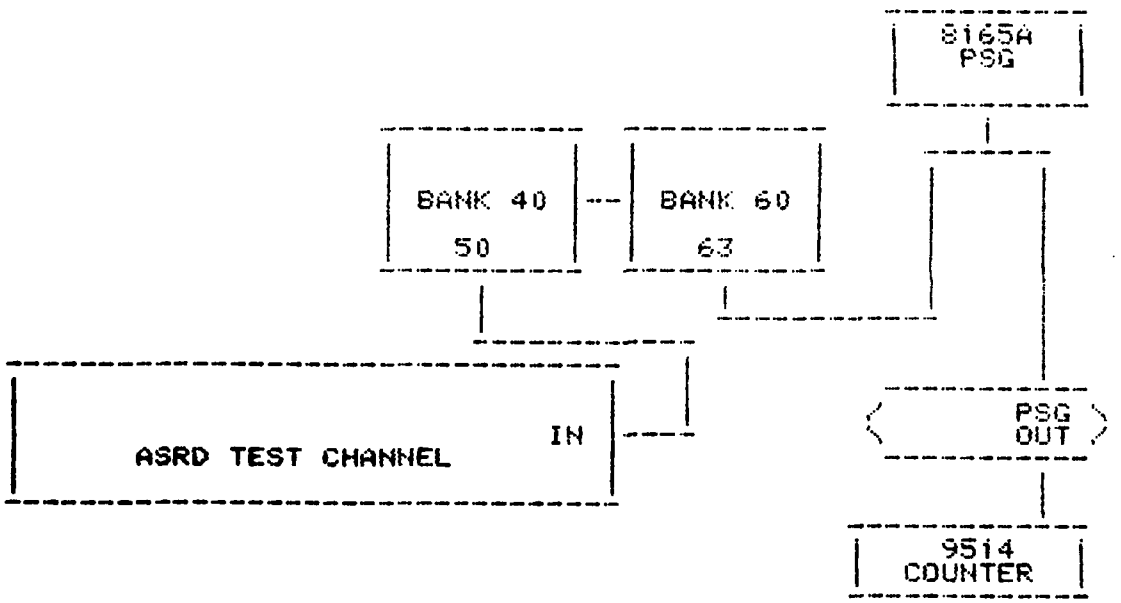
The commands supported by CADIG are:

CA,lu	CAlibrate channel with list to lu.
DA	DAte check or change.
DI,lu	DIsplay the dates and serial numbers of channels which have been calibrated with list to lu.
EX	EXit from CADIG.
LI	LIst the CADIG commands.
PR,lu	PRint the previous CA results again with list to lu.
RE,lu	REad the calibration results with list to lu.
RU,namr,p1-5	RUn program namr with parameters p1-p5.
SC,n,x	SCanner contacts n closed(x for bank 2 only).

ASRD TIME CALIBRATION

Time calibration of the BTST digitizer is accomplished by applying an 0.5 second pulse from the PSG to the digitizer and measuring the pulse width on a counter/timer. The counter/time must be calibrated with traceability to National Bureau of Standards. The PSG output pulse goes from 0V to 9V and remains there for half a second. Then the pulse returns to 0V. The counter/timer is programmed to start at 4.5V with positive slope and to stop at 4.5V with a negative slope. The digitizer is programmed for variable sample rate table 15 and will trigger on the pulse the same as the counter/timer. The program calculates the time from the transition through the 4.5V level on the rising edge to the falling edge of the pulse. This calculated time is compared to the counter/timer measured time. If the two are in agreement within +/-5microseconds then the channel is considered calibrated. If the two readings are not in agreement then the new calibration can not be written to disc. Only one channel at a time may be calibrated.

The equipment connections required for performing the calibration are:



CATIM is an interactive command oriented program. The operator is prompted to check the current time at the start. The operator is prompted to change cables prior to starting each channels calibration. Although the program checks the suitability of the calibration as mentioned above, the operator has final control over accepting or rejecting the value.

The run string for the program is:

```
RU,CATIM,intrctv lu(dflt=1)
```

The commands supported by CATIM are:

CA,lu	CAlibrate channel with list to lu.
DA	DAte check or change.
DI,lu	DIsplay the dates and serial numbers of channels which have been calibrated with list to lu.
EX	EXit from CATIM.
LI	LIst the CATIM commands.
PR,lu	PRint the previous CA results again with list to lu.
RE,lu	REad the calibration results with list to lu.
RU,namr,p1-5	RUn program namr with parameters p1-p5.
SC,n,x	SCanner contacts n closed(x for bank 2 only).

CALIBRATION FILE DISPLAY AND MODIFICATION

Program CALX provides the capability to display and modify the entries in the calibration file. The serial numbers and calibration dates of all channels may be listed or the serial numbers, calibration dates and calibration factors may be listed. The operator may enter changes by hand for any entry in the calibration file or may swap the calibration information for two channels. When operator entries are made, the program checks that the calibration factor is within 5% of the nominal value.

The run string for the program is:

```
RU,CALX,intrctv lu(dflt=1)

  intrctv lu:  interactive operator terminal(2648)
```

CALX is an interactive command oriented program. The commands supported by CALX are:

```
DA,lu           Display serial numbers, dates and cal factors for
                  All channels on list lu.
DI,lu           DIsplay serial numbers and dates for all channels
                  on list lu.
EN,c,d          ENter new values for device d in channel c.
EX              EXit CALX.
LI              LIst CALX commands.
RE,c,lu         REad & display channel c serial numbers, dates
                  and cal factors on list lu.
RU,name,p1-p5   RUn program name with parameters p1,...,p5.
SW,c1,c2,d      SWap all entries for device d on channels c1 & c2.

  device d:  SC=316, FA=416, OA=516, DI=asrd, TI=time & AL=all
```

The list format for the DA and RE commands is:

```
CH DV   SN    DATE    CAL FACTORS
-- -- ----- ---- ------------------------------------
 0 DI    17 3249  .004892
   SC  6658 3249 V:1.0001 C:1.0028 S1:1.0037 S2: 9.991 S3: 99.39 S4: 989.6
   FA  2121 3249  .9819 1.9629 3.9378 4.9166  9.824 19.630 24.486 39.244
               49.011  98.27 195.91 244.87 392.39 489.12  978.2
   OA  1001 3249 2.0150
   TI    17 3249
```

The list format for the DI command is:

```
        316 SC        416 FA        516 OA       DIGITIZER       TIME
     ***********   ***********   ***********   ***********   ***********
CH     SN    DATE    SN    DATE    SN    DATE    SN    DATE    SN    DATE
--   -----  ----  -----  ----  -----  ----  -----  ----  -----  ----
 0    6658  3249   2121  3249   1001  3249     17  3249     17  3249
 1    6924  3249    892  3249   1002  3249     60  3249     60  3249
```

APPENDIX N - VERIFICATION PROGRAMS' OPERATION INSTRUCTIONS

PROGRAM CK316

Program CK316 checks the performance of the signal conditioner. The program performs a series of measurements for each of the operating modes of the signal conditioner. For voltage and charge gain at four different frequencies, noise, offset, polarity and calibration steps are all measured. For strain mode auto balance, manual balance, excitation, gain at each strain gain using the calibration steps are all measured. Each programmable function of the signal conditioner is exercised by the program. The program results must be screened by an operator to look for problems. In many cases there is no pass/fail criteria, but rather only looking for symptoms of a problem. The program controls the PSG, the scanner, the DVM and the signal conditioner. The operator must move the cables from channel to channel as only one channel may be checked at a time.

VOLTAGE & CHARGE GAIN TEST: The PSG is set to provide a sine wave at an amplitude of 9.8V peak. Four frequencies are used 80kHz, 8kHz, 800Hz and 80Hz. The gain is established by measuring the input voltage and then the output voltage. The ratio of the two readings is calculated and listed.

VOLTAGE & CHARGE NOISE TEST: A termination resistor is connected to the input and the AC noise is measured. 51 ohm and 10k ohm resistors are used in the voltage mode while only 51 ohms is used in the charge mode.

VOLTAGE & CHARGE POLARITY TEST: The PSG is programmed to provide a +5VDC signal. The output signal is measured to determine its polarity. Voltage mode should be positive and charge mode should be negative.

VOLTAGE & CHARGE OFFSET TEST: A termination resistor is connected to the input and the DC offset is measured. 10k ohm resistance is used for voltage mode while 51 ohm is used for charge mode.

VOLTAGE & CHARGE CALIBRATOR TEST: The internal calibrator has 128 programmable voltage levels(0-127). Ten of these levels are measured(0, 1, 2, 4, 8, 16, 32, 64, 96, 127) and a first order least squares fit is performed on voltage versus step. The slope, offset and largest residual from the fit are listed.

VOLTAGE & CHARGE SCOPE TEST: With a termination resistor connected to the input, the output of the channel is connected to the oscilloscope to allow a check for noise spikes which are not detectable on the DVM noise measurement. A 10k ohm resistor is used for voltage mode while a 51 ohm resistor is used for charge mode. Typically any noise spikes should be below 1mV peak.

STRAIN EXCITATION TEST: The output of the channel is connected to the DVM. The operator is instructed to adjust the excitation control until a value of 10.00V is obtained. This adjustment checks that the excitation supply is working and provides a known value of excitation for the following tests.

AUTO BALANCE TEST: The auto balance circuit is activated and then the output level is measured. The resulting DC level is a measure of how well the auto balance circuitry works.

STRAIN GAIN & CALIBRATION TEST: The internal calibrator has sixteen independent steps(0, 1, -1, 2, -2, 4, -4, 8, -8, 16, -16, 32, -32, 64, -64). A calibration at these sixteen steps is carried out at each strain gain value(1, 10, 100, 1000).

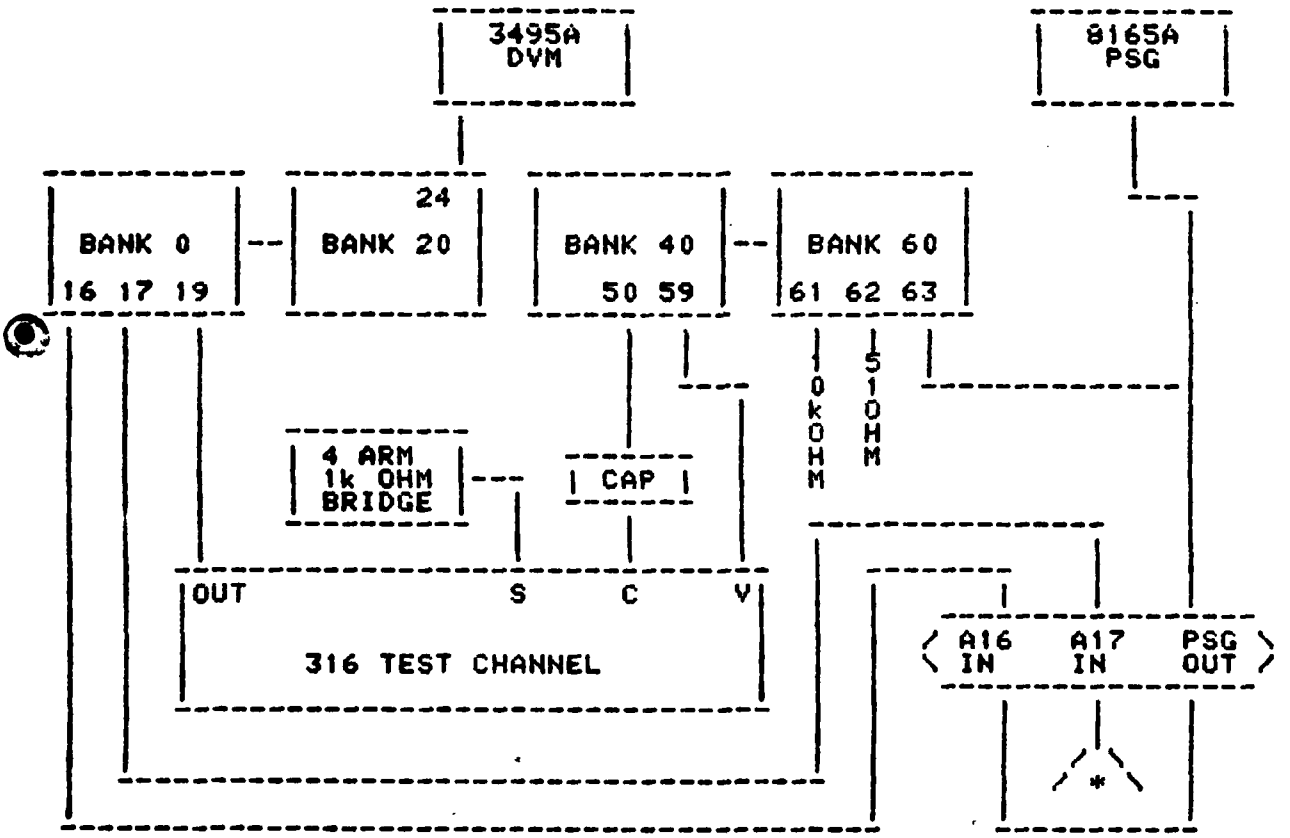
A first order least squares fit of the output voltage measured versus step is performed. The slope, offset and largest residual are listed. Note that at a gain of 1000 the last four steps can not be used as they generate an output larger than +/-10V.

STRAIN MANUAL BALANCE TEST: The output of the channel is connected to the DVM. The operator is instructed to turn the manual balance control while observing the DVM for changes.

STRAIN SCOPE TEST: The output of the channel is connected to the oscilloscope to allow a check for noise spikes. Typically any noise spikes should be below 1mV peak.

Only one channel at a time may be verified. The normal BITE equipment installed in a BTST is used to perform the verification.

The equipment connections required for performing the verification are:



* = clip leads to test excitation voltage

Normally the only cable connection changes required are:

1. Connect a short cable from the PSG OUT to the A16 IN jacks on the front patch panel.
2. Connect scanner cables 50 and 59 to the charge capacitor and voltage inputs respectively.
3. Connect the 4 arm 1k bridge to the strain input.
4. Connect scanner cable 19 to the signal conditioner output.

CK316 is an interactive command oriented program. The operator is prompted to change cables prior to starting each channels verification.

The run string for the program is:

```
RU,CK316,intrctv lu(dflt=1)
```

The commands supported by CK316 are:

```
AL              Test ALL modes.
CH              Test CHarge mode.
EX              EXit from CK316.
LI              LIst the CK316 commands.
PR,lu           PRint the previous test results again with list
                to lu.
RU,namr,p1-5    RUn program namr with parameters p1-p5.
SC,n,x          SCanner contacts n closed(x for bank 2 only).
ST              Test STrain bridge mode.
VO              Test VOltage mode.
```

The test results provided by the program are in the following format:

```
CHANNEL  0      VOLTAGE MODE      SN6658
  GAIN WITH 9.8V IN-1.004@80KHZ 1.002@8KHZ 1.000@800KHZ 1.000@80HZ
  NOISE-- .0019V@50ohm        .0041V@10kohm
  POLARITY OK? YES        4.98V MEASURED
  OFFSET =    .0000V
  CALIBRATOR SLOPE= .0443  OFFSET= .0001  MAX RES@STEP 127 = -.0018V
  SCOPE CHECK: OK .8MVP-P

CHANNEL  0      CHARGE  MODE      SN6658
  GAIN WITH 9.8V IN-1.001@80KHZ 1.000@8KHZ 1.003@800KHZ 1.003@80HZ
  NOISE-- .0022V@50ohm
  POLARITY OK? YES         -4.54V
  OFFSET =  -.0026V
  CALIBRATOR-SLOPE= .0443  OFFSET= .0001  MAX RES@STEP 127 = -.0018V
  SCOPE CHECK: OK  1.4MV P-P

CHANNEL  0      STRAIN MODE       SN6658
  AUTOBALANCE:     .000702V
  CAL-X    1GAIN:SLOPE= .000621 OFST= .000606 MAX RES@STEP 64= .000042V
  CAL-X   10GAIN:SLOPE= .006215 OFST=-.000212 MAX RES@STEP-32= .000504V
  CAL-X  100GAIN:SLOPE= .062188 OFST=-.001998 MAX RES@STEP-32= .005127V
  CAL-X1000GAIN:SLOPE= .617859 OFST=-.015612 MAX RES@STEP 16= .002091V
  SCOPE CHECK: OK  0.7MVP-P
```

PROGRAM CK416

Program CK416 checks the performance of the filter/amplifier. The program performs a series of measurements to check the gain, filter response, overload detection, noise and offset characteristics. Only a few of the many possible programmable parameter combinations are exercised, however each bit and each function is exercised by the the program. The program results must be screened by an operator to look for problems. In many cases there is no pass/fail criteria, but rather only looking for symptoms of a problem. The program controls the PSG, the scanner, the DVM and the filter/amplifier. The operator must move the cables from channel to channel as only one channel may be checked at a time.

GAIN TEST: The PSG is set to provide a 1kHz sine wave at an amplitude that will produce an output signal near 9.9V peak. The filter/amplifier is set to a frequency of 80kHz. A series of gain measurements are made for each prefilter gain setting(postfilter set to 1) and then each postfilter gain setting(prefilter set to 1). The gain is established by measuring the input voltage and then the output voltage. The ratio of the two readings is calculated and listed.

FILTER TEST: The filter/amplifier is set to a gain of 1 and to the desired filter cutoff frequency. The PSG is programmed to provide a 1.6V peak sine wave signal. As a baseline the ratio of input to output is measured at one-tenth the cutoff frequency. Then the PSG is moved to the cutoff frequency minus 3kHz where the frequency is incremented until the 3dB point is found. Checks are also made at 1/4, 1/16 and 1/256 of the cutoff frequency to check the passband. This procedure is carried out at 0.5, 5, 10, 20, 40 and 80kHz.

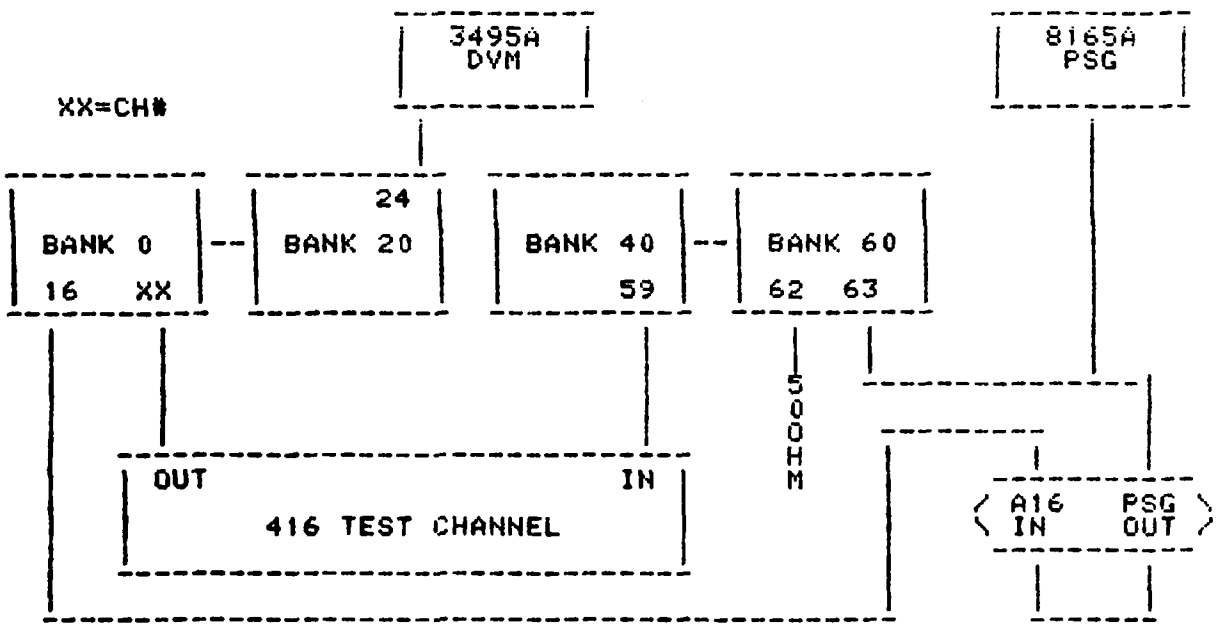
POLARITY TEST: The PSG is programmed to provide a +5VDC signal. The filter/amplifier is programmed for a gain of 1 and a cutoff frequency of 80kHz. The output signal is checked to determine that it is +5VDC.

OVERLOAD TEST: The PSG is programmed to provide a single cycle of a sine wave at 10kHz. The amplitude of the waveform and the gain and frequency of the filter/amplifier are adjusted to generate prefilter, postfilter and both prefilter and postfilter overloads. In each case the filter/amplifier overload circuitry is read and checked to see if it is working properly.

OFFSET AND NOISE TEST: A 50 ohm termination is placed across the channel. The DC offset and the AC noise of the channel are measured at gains of 1, 10, 100 and 1000.

Only one channel at a time may be verified. The normal BITE equipment installed in a BTST is used to perform the verification.

The equipment connections required for performing the verification are:



Normally the only cable connection changes required are:

1. Connect a short cable from the PSG OUT to the A16 IN jacks on the front patch panel.
2. Connect scannar cable 59 to the channel to be tested.

CK416 is an interactive command oriented program. The operator is prompted to change cables prior to starting each channels verification.

The run string for the program is:

```
RU,CK416,intrctv lu(dflt=1)
```

The commands supported by CK416 are:

```
EX              EXit from CK416.
LI              LIst the CK416 commands.
PR,lu           PRint the previous TE results again with list
                to lu.
RU,namr,p1-5    RUn program namr with parameters p1-p5.
SC,n,x          SCanner contacts n closed(x for bank 2 only).
TE              TEst channel.
```

The list results provided by the program are in the following format:

```
CH  0           SN9034
      GAIN    [80kHz BW  VOUT=9.9VPK @ 1KHZ]
        SET     1         2         5         10        20        50
        PRE     1.000     2.003     5.007     10.019    20.031    50.089
        POST    1.000     2.003     5.012     10.028    20.043
      BANDWIDTH    [VIN=1.60VPK      GAIN=1]
        FREQ:       .5      5.0      10.0      20.0      40.0      80.0
        3DB F:      .49     4.95      9.91     19.89     39.98     80.15
        PB OK:
      POLARITY OK? YE
      OVERLOADS:  0-OK  1-OK  2-OK  3-OK
      OFST & NOISE     [50OHM TERM, 80KHZ BW]
        GAIN:          1        10       100      1000
        OFST:       .001     -.000      .014      .186
        NOISE       .000      .000      .000      .008
```

PROGRAM CHKOU

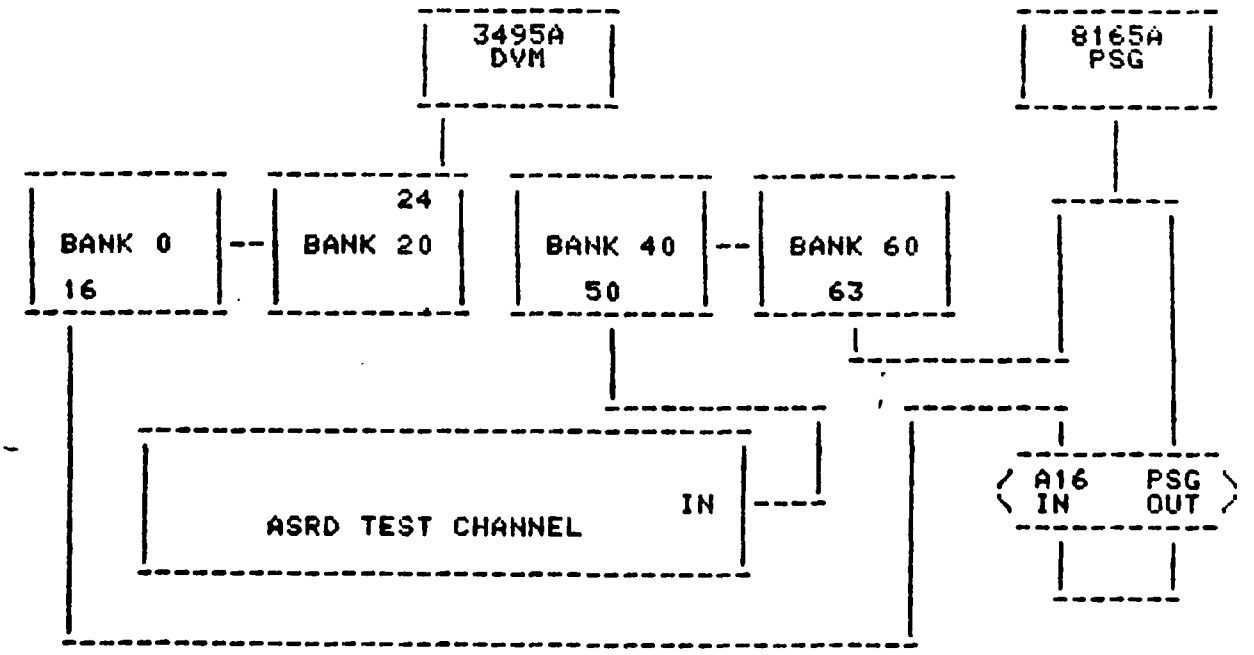
Program CHKOU evaluates the general performance of an ASRD channel. The program controls the PSG, the scanner, the DVM and the digitizer. The channel data is stored on disc in an ADCHK compatible data file so that plots or further analysis may be carried out. There are two modes of operation for the program: one which performs the tests and the other which allows data on disc to be processed again. A total of 3 different tests are possible with the program to test the channel at high, medium and low rates.

The PSG is programmed to provide a burst of sine waves of 19V p-p amplitude with frequency changed for the particular test to be performed. The digitizer is programmed for variable sample rate appropriate to the input frequency, with internal trigger, no run delay and appropriate stop delay and memory size to allow the entire burst to be recorded. A number of conditions are checked to determine if the channel is working correctly.

The parameters which change with test number are:

Parameter	Test 1	Test 2	Test 3
PSG frequency	79kHz	34kHz	5kHz
PSG burst length	250	400	350
ASRD stop delay	6	13	100
ASRD memory size	4k	6k	5k
ASRD rate table	15	6	0
Minimum number of data points	2500	4700	3500
Maximum number of data points	2900	5150	4850

The equipment connections required for performing the verification are:



Normally the only cable connection changes required are:

1. Connect a short cable from the PSG OUT to the A16 IN jacks on the front patch panel.
2. Connect scanner cable 50 to the channel(s) to be tested.

```
**********************************************************
*                                                        *
*  NOTE:  This program writes new parameters to the data *
*         disc parameters area.  Save the current parameters *
*         in a file using the ADCHK SP command if necessary. *
*                                                        *
**********************************************************
```

The run string for the program is:

```
RU,CHKOU,intrctv lu(dflt=1),list lu(dflt=intrctv lu),
        test no,data disc lu(dflt=stop),
        data start track(dflt=directory)
```

intrctv lu: operator interactive terminal

list lu: results list device(terminal, printer or tape)

test no = 1-3: test all channels selected on disc - use ADCHK SE command to select channels prior to running CHKOU

test no # 1-3: reprocess data already on disc

data disc lu: disc lu where parameters are located and where test data is to be stored

data start track: location where data generated by this test should be stored

There are a number of different error conditions which may prevent the channel from testing successfully. Some of the conditions and the associated error messages printed are:

DIG rgstr/mem ERR-CHNLS: nn mm .....
During the load process if any errors are encountered this message will appear. The channel may not perform properly if load errors are present.

CHANNELS NOT TRIGGERED ARE: nn mm .....
The channel did not tirgger. Check cabling from PSG to scanner and scanner to channel. Check that PSG and scanner are programmed.

CHANNELS WITH INSUFFICIENT DATA ARE: nn mm .....
The channel triggered but did not take the minimum number of data points.

NUMBER OF DATA POINTS TAKEN(xxxxxx)EXCEEDS EXPECTED NUMBER-nnnn
The channel took more data points than it should have. Stop delay or decrease delay may not be working.

THE NUMBER OF IRIG TIME WORDS (xxxxxx) IS NOT CORRECT
Should have three or four time words. More or less is an error.

THERE ARE xxxxxx MISSING CHANGE WORDS
When the rate changes there should be a change word inserted between the old and new rates.

THERE WERE ONLY x CHANGE WORDS
Should have at least two change words in the record.

MAX SEL RATE OF xx DOES NOT HAVE MOST DATA POINTS--RATE yy DOES
With a burst input the rate should go to the maximum allowed value in the table.

DATA POINTS AT RATES NOT SELECTED WERE FOUND
If a rate is not in the table then it should not be found in the data.

THERE WERE xxxxxx DOUBLE CHANGE WORDS
There should not be two adjacent change words in the record.

THERE WERE xxxxxx TIME WORDS NOT IN GROUPS OF 3 OR 4
Time words should occur only in groups of 3 or 4. Any other size group is an error.

NO OF DECREASE DELAY WORDS xxxx IS OUT OF RANGE (75-90)
There should be 75-90 data words from the point at which the signal reaches baseline and the point at which the rate decreases.

DATA EXHAUSTED BEFORE TERMINAL BASELINE WAS FOUND
Sine wave still present at end of data record.

CHANGE OR TIME WORD FOUND WHILE LOOKING FOR TERMINAL BASELINE
There should be no change or time words in the end of the sine wave.

RATE n FOR TERMINAL BASELINE IS NOT EXPECTED RATE m
The end of the sine wave should still be at the maximum rate.

TERMINAL BASELINE FOUND BUT END OF DATA FOUND PRIOR TO DECREASE IN RATE
There was no decrease in rate even though the burst input is gone.

RMS VALUE OF xx.xxx IS OUT OF RANGE
Calculated value of RMS is not within +/-5% of measured value.

AVERAGE VALUE OF xx.xxx IS OUT OF RANGE
Calculated value of average is not within +/-0.25V of measured value.

SIGNAL IS CLIPPED
Input signal should not exceed 9.98 or go below -9.98

TIME WORDS ANALYSIS BYPASSED DUE TO MISSING TIME WORDS
If less than three time words are present, the time words analysis is bypassed.

IRIG TIME LOCATION OF xxxxxx IS NOT CORRECT
The time words group does not appear at words 79-83 as required.

TRIGGER POINT IS AT xxxxxx NOT AT EXPECTED VALUE OF yyyyyy
The transition through the threshold value with the proper slope was found outside the allowed region following the trigger time and not at the expected position.

TRIGGER POINT NOT FOUND WITHIN EXPECTED INTERVAL
The transition through the threshold value with the proper slope was not found within 100 words following the trigger time word.

CHANNELS WITH WRONG IRIG TIME ARE: nn mm .....
If a channels trigger time word is not in agreement with any other channels time word (assuming at least two channels agree), then it is listed as a wrong time word.

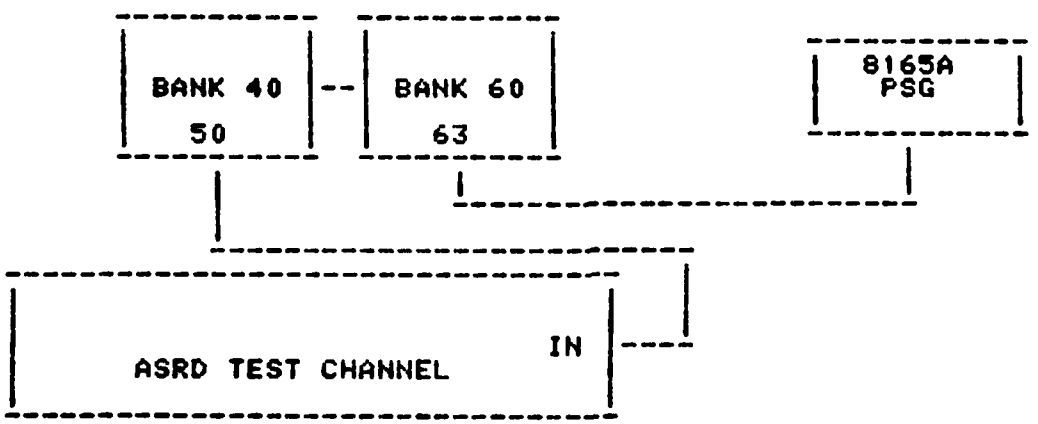
CHANNELS WITH QUESTIONABLE IRIG TIME ARE: nn mm .....
If two channels trigger time words agree with each other but not with other channels which agree, then they are listed as having a questionable time word.

PROGRAM MSTST

Program MSTST checks the master/slave trigger circuitry of an ASRD channel. The program sets one channel as master and others as slaves and then checks that all channels trigger at the same time. The program controls the PSG, the scanner and the digitizer. All data analysis is carried out in memory and no disc data file is created.

The PSG is programmed to provide a single cycle of a triangle wave of 19Vp-p amplitude and 500Hz frequency. The master is programmed with a threshold of 5V and positive slope. The slave channels can trigger only from the master. The first 100 words from each channel are transfered to memory and the trigger time word is found. The trigger time for each slave is compared to the master channel. If the time difference is less than 1 microsecond then the master/slave trigger circuitry is considered to be working correctly.

The equipment connections required for performing the verification are:



Normally the only cable connection change required is to connect scanner cable 50 to the channel(s) to be tested.

The run string for the program is:

RU,MSTST,intrctv lu(dflt=1),list lu(dlft=intrctv lu),
chnl no,data disk lu(dflt=intrctv),print control

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

chnl no: master channel number - does not have to be selected on disc

data disc lu: disc lu where parameters are located

print control = 0: do not print trigger time buffer
print control # 0: print trigger time buffer

There are a number of different error conditions which may prevent the channel from testing successfully. Some of the conditions and the associated error messages printed are:

MSTST - ICH = nn - ERROR IN PARAMETER LOAD
An error was encountered in trying to load parameters to the channel. Run RMTST to try to locate the problem.

MSTST ICH = nn - READ MAR ERROR
An error was encountered in trying to read the memory address register of the channel. Run RMTST to try to locate the problem.

MSTST ICH = nn - NO TRIGGER
The channel did not trigger. Check cabling from PSG to scanner and scanner to channel. Check that PSG and scanner are programmed. Run TRGRT to test trigger circuity.

MSTST ICH = nn - WORD CNT(nnnnnn) BELOW REQUIRED 100 POINTS
The channel triggered but did not take the required amount of data.

MSTST ICH = nn - NO IRIG TIME FOUND
The trigger time word was not found. Verify the time word insertion circuitry with TRGRT.

MSTST ICH = nn FAILED DIFF =xxxxx.xxxxxx
The slave channel trigger time differed from the master time by more than 1 microseconds. The difference is listed.

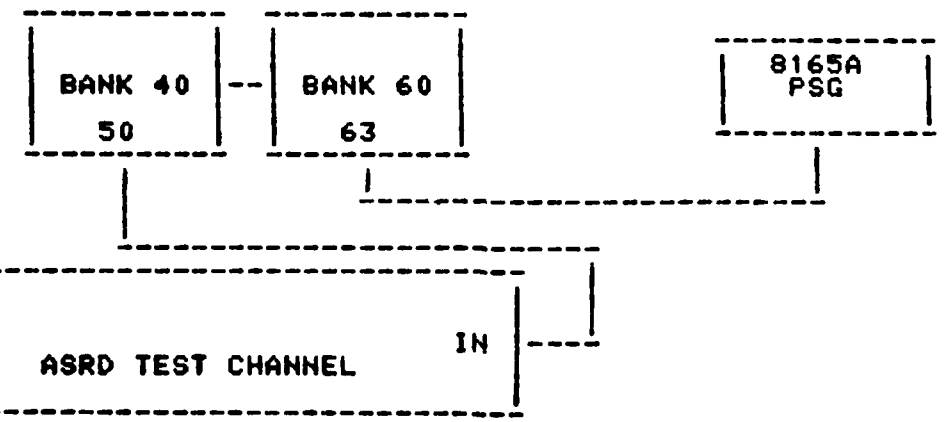
MSTST FINISHED - nn CHANNELS FAILED
A test summary message is provided.

PROGRAM PRETR

Program PRETR checks the pretrigger circuitry of an ASRD channel. The program performs a test for each value of pretrigger from 1k to memory size-1k in 1k steps. The program controls the PSG, the scanner and the digitizer. All data analysis is carried out in memory and no disc data file is created. If five errors are accumulated on a channel, then testing is terminated for that channel. There are two test modes possible with the program: one which tests only a single channel and the other which tests all channels currently selected on the data disc.

The PSG is programmed to provide a single cycle of a triangle wave of 19Vp-p amplitude and 500Hz frequency. The digitizer is programmed with a threshold of 5V and positive slope along with the test value of pretrigger. In a properly working channel the transition through the 5V trigger level should occur between samples 85 and 86 following the programmed pretrigger value. After the test run is in the digitizer memory, the program transfers 100 words starting at the end of pretrigger memory. If the trigger transition is found from 5 to 7 words after the trigger time word, then the channel is considered to be working correctly.

The equipment connections required for performing the verification are:



Normally the only cable connection change required is to connect scanner cable 50 to the channel(s) to be tested.

The run string for the program is:

```
RU,PRETR,intrctv lu(dflt=1),list lu(dlft=intrctv lu),
        chnl no,data disk lu(dflt=intrctv),print control
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

chnl no = 0-31: test only the specified channel
chnl no # 0-31: test all channels selected on disc - use ADCHK SE command to select channels prior to running PRETR

data disc lu: disc lu where parameters are located

print control = 0: do not print last 24 words in data buffer
print control # 0: print last 24 words in data buffer

There are a number of different error conditions which may prevent the channel from testing successfully. Some of the conditions and the associated error messages printed are:

PRETR = xxxK - ERROR IN PARAMETER LOAD
An error was encountered in trying to load parameters to the channel. Run RMTST to try to locate the problem.

PRETR = xxxK - NO TRIGGER
The channel did not trigger. Check cabling from PSG to scanner and scanner to channel. Check that PSG and scanner are programmed. Run TRGRT to test trigger circuity.

PRETR = xxxK - WORD CNT(nnnnnn) BELOW EXPECTED #(mmmmmm)
The channel triggered but did not take the required amount of data. The channel should take pretrigger + 1k worth of data.

PRETR = xxxK - NO IRIG TIME FOUND
The trigger time word was not found. Verify the time word insertion circuitry with TRGRT.

PRETR = xxxK - THRESHOLD NOT FOUND
The transition from less than 5V to greater than 5V was not found in the buffer.

PRETR = xxxK = THRESHOLD AT nn NOT VALID
The transition was found, but not at the correct location.

PRETR TERMINATED FOR ICH= nn ON FIFTH ERROR
Five errors have been encountered on a channel. Processing is terminated.

^^^^^^^^ CHANNEL nn WITH MEMSIZ = xxxK HAD m ERRORS ^^^^^^^^
A test summary message for each channel is provided.

PROGRAM RMTST

Program RMTST tests the internal registers, memory and clock of an ASRD channel. The internal microprocessor carries out the test and reports errors back to RMTST. No test equipment is needed for this test. All program operations are carried out in memory and no disc data file is created.

The program reads the memory size of a channel, zeros the error register, starts the register test, waits until it is done, starts the memory test and then goes on to the next channel. The amount of time required for the test is determined by the size of the largest memory board. Allow about 1.3 minutes for each 16k of memory size. If any error codes are returned by the channel, an error message will result.

The run string for the program is:

```
RU,RMTST,intrctv lu(dflt=1),list lu(dflt=intrctv lu),
        chan no,data disk lu(dflt=intrctv)
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

chan no = 0-31: test only the specified channel
chan no # 0-31: test all channels selected on disc - use ADCHK SE command to select channels prior to running RMTST

data disc lu: disc lu where parameters are located

The results are printed in the following format:

***** CHANNEL nn MEMSIZ = xxxK # ERRORS = yy

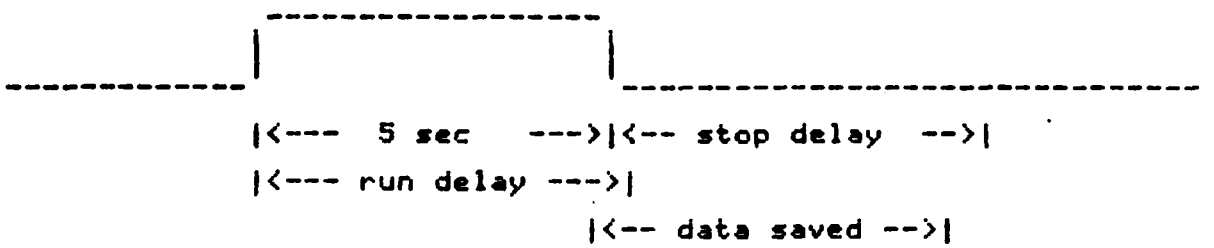
For a working channel yy will be zero. If yy is positive, then a description of the error(s) found will be printed. If yy = -1 then a communication error with the channel occurred and the test was not run. In this case an additional error message will be printed with a description of the error.

PROGRAM RSDLY

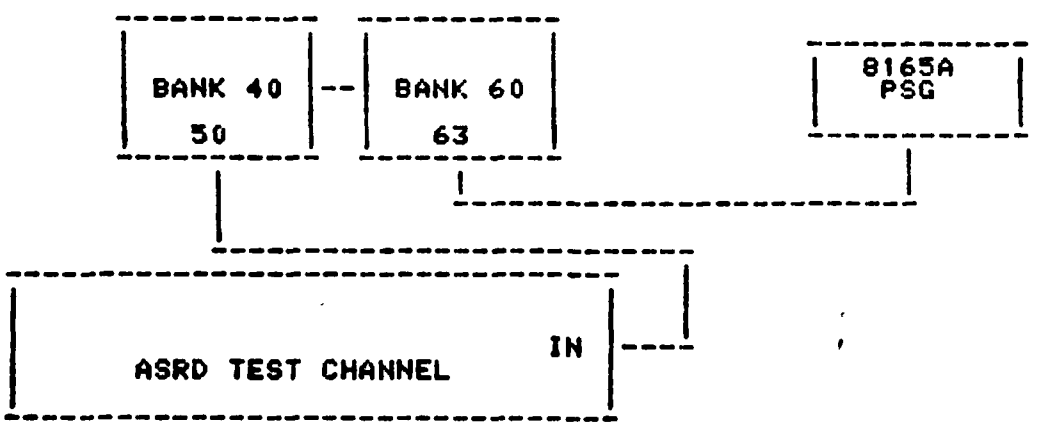
Program RSDLY performs a test of the run and stop delay circuitry of an ASRD channel. The program controls the PSG, the scanner and the digitizer. The channel data is stored on disc in an ADCHK compatible data file so that plots or further analysis may be carried out. There are two modes of operation for the program: one which performs the tests and the other which allows data on disc to be processed again.

The PSG is programmed to provide a single pulse from 0V to +9V in amplitude and 5 seconds in duration. The digitizer is programmed with a threshold of 5V and positive slope, run and stop delays of 5.1 seconds each and fixed rate 4(390.6sps). With this combination the run delay should cause the digitizer to start taking data 0.1 second after the pulse returns to 0V. However, due to the 80 word pretrigger memory, there will be 0.2 seconds worth of data prior to the run signal. Thus, the transition from +9V to 0V should occur around word 42 of the record. The stop delay should cause approximately 1953 words of data to be saved in memory. If these two conditions are met then the run and stop delay circuitry is considered to be working correctly.

The timing sequence is:



The equipment connections required for performing the verification are:



Normally the only cable connection change required is to connect scanner cable 50 to the channel(s) to be tested.

```
**************************************************************
*  NOTE:  This program writes new parameters to the data     *
*         disc parameters area.  Save the current parameters *
*         in a file using the ADCHK SP command if necessary. *
**************************************************************
```

The run string for the program is:

```
RU,RSDLY,intrctv lu(dflt=1),list lu(dflt=intrctv lu),
         test no,data disc lu(dflt=stop),
         data start track(dflt=directory)
```

intrctv lu: operator interactive terminal

list lu: results list device(terminal, printer or tape)

test no = 1: test all channels selected on disc - use ADCHK SE command to select channels prior to running RSDLY

test no # 1: reprocess data already on disc

data disc lu: disc lu where parameters are located and where test data is to be stored

data start track: location where data generated by this test should be stored

There are a number of different error conditions which may prevent the channel from testing successfully. Some of the conditions and the associated error messages printed are:

DIG rgstr/mem ERR-CHNLS: nn mm .....
During the load process if any errors are encountered this message will appear. The channel may not perform properly if load errors are present.

CHANNELS NOT TRIGGERED ARE: nn mm .....
The channel did not tirgger. Check cabling from PSG to scanner and scanner to channel. Check that PSG and scanner are programmed. Run TRGRT to test trigger circuitry.

CHANNELS WITH INSUFFICIENT DATA ARE: nn mm .....
The channel triggered but did not take 100 or more points.

THE NUMBER OF IRIG TIME WORDS (xxxxxx) IS NOT CORRECT
Should have four time words. More or less is an error.

THERE WERE xxx CHANGE WORDS - SHOULD HAVE NONE
Fixed rate data should have no change words.

DATA POINTS AT RATES NOT SELECTED WERE FOUND
All data should be at rate = 4.

TIME WORDS ANALYSIS BYPASSED DUE TO MISSING TIME WORDS
If less than four time words are present, the time words analysis is bypassed.

IRIG TIME LOCATION OF xxxxxx IS NOT CORRECT
The time words group does not appear at words 79-83 as required.

RUN DELAY: LEVEL AT SMPL 32 IS NOT CORRECT
The 9V pulse should be present at word 32 if the run delay is working.

RUN DELAY: TRANSITION NOT IN EXPECTED INTERVAL(32-52)
The transition from 9V to 0V was not found between words 32 52 as would be expected if the run delay is working.

STOP DELAY: WRDCNT(xxxxxx) NOT IN EXPECTED RANGE(1940-1975)
The number of data words in memory is not the amount expected if the stop delay is working.

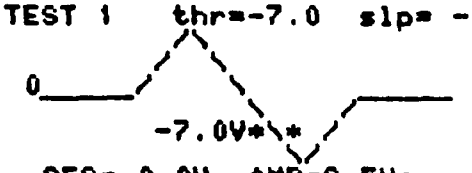
PROGRAM TRGRT

Program TRGRT performs a test of the internal trigger circuitry of an ASRD channel. The program controls the PSG, the scanner and the digitizer. The channel data is stored on disc in an ADCHK compatible data file so that plots or further analysis may be carried out. There are two modes of operation for the program: one which performs the tests and the other which allows data on disc to be processed again. A total of 12 different tests are possible with the program to test the various combinations of internal trigger parameters.

The PSG is programmed to provide a single cycle of a triangle wave with amplitude and offset changed for the particular test to be performed. The digitizer is programmed for rate B(50ksps) with threshold and slope determined by the particular test to be performed. In a properly working channel the transition through the threshold should occur between the fifth and sixth samples following the trigger time word. If the correct transition point and slope are found, then the channel internal trigger circuity is considered to be working properly.

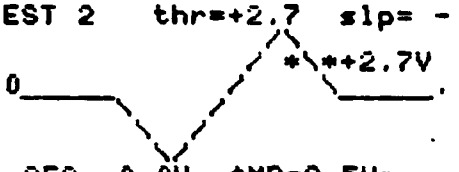
The trigger threshold and slope and the PSG waveform used for each test are:

TEST 1 thr=-7.0 slp= -



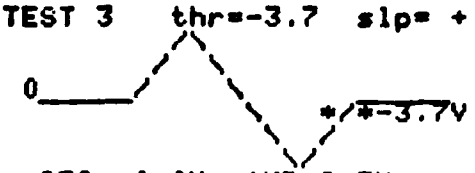
OFS= 0.0V AMP=9.5Vp

TEST 2 thr=+2.7 slp= -



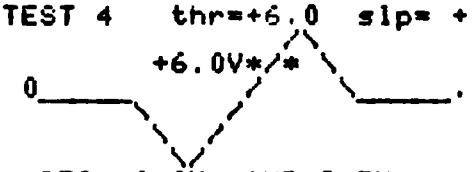
OFS= 0.0V AMP=9.5Vp

TEST 3 thr=-3.7 slp= +



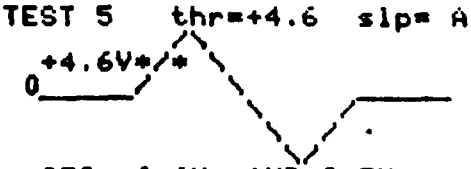
OFS= 0.0V AMP=9.5Vp

TEST 4 thr=+6.0 slp= +



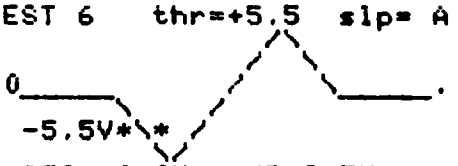
OFS= 0.0V AMP=9.5Vp

TEST 5 thr=+4.6 slp= A



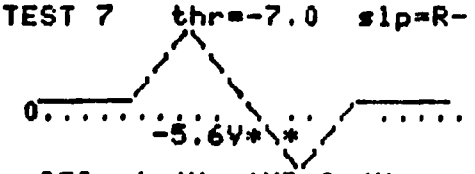
OFS= 0.0V AMP=9.5Vp

TEST 6 thr=+5.5 slp= A



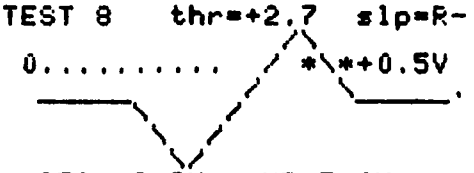
OFS= 0.0V AMP=9.5Vp

TEST 7 thr=-7.0 slp=R-

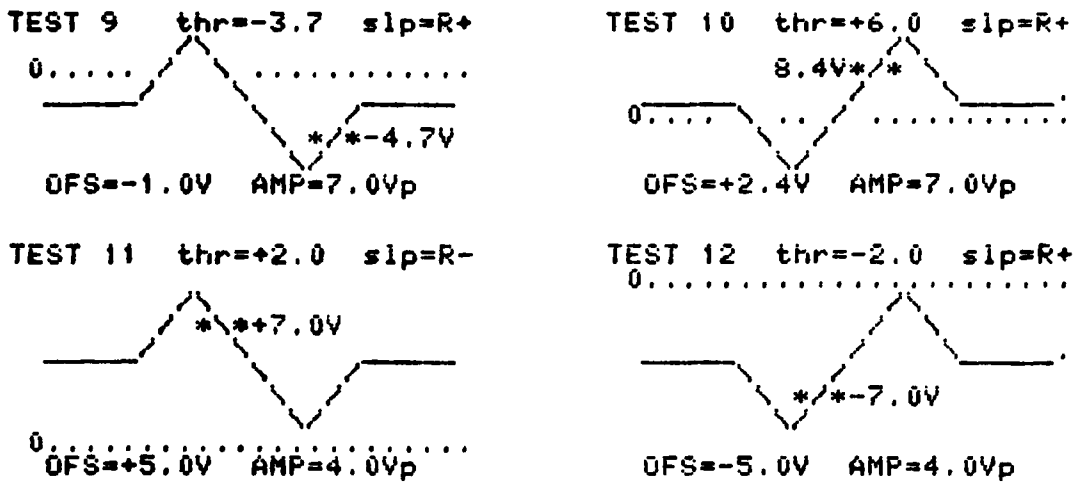


OFS=+1.4V AMP=8.4Vp

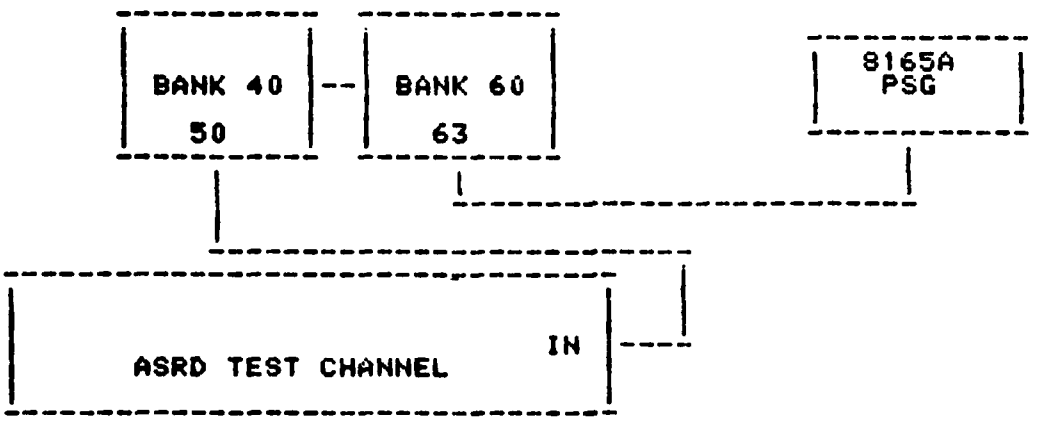
TEST 8 thr=+2.7 slp=R-



OFS=-2.2V AMP=7.0Vp



The equipment connections required for performing the verification are:



Normally the only cable connection change required is to connect scanner cable 50 to the channel(s) to be tested.

```
*****************************************************************
*                                                               *
*  NOTE:  This program writes new parameters to the data        *
*         disc parameters area.  Save the current parameters    *
*         in a file using the ADCHK SP command if necessary.    *
*                                                               *
*****************************************************************
```

The run string for the program is:

```
RU,TRGRT,intrctv lu(dflt=1),list lu(dflt=intrctv lu),
         test no,data disc lu(dflt=stop),
         data start track(dflt=directory)
```

intrctv lu: operator interactive terminal

list lu: results list device(terminal, printer or tape)

test no = 1-12: test all channels selected on disc - use ADCHK SE command to select channels prior to running TRGRT

test no # 1-12: reprocess data already on disc

data disc lu: disc lu where parameters are located and where test data is to be stored

data start track: location where data generated by this test should be stored

There are a number of different error conditions which may prevent the channel from testing successfully. Some of the conditions and the associated error messages printed are:

DIG rgstr/mem ERR-CHNLS: nn mm .....
During the load process if any errors are encountered this message will appear. The channel may not perform properly if load errors are present.

CHANNELS NOT TRIGGERED ARE: nn mm .....
The channel did not tirgger. Check cabling from PSG to scanner and scanner to channel. Check that PSG and scanner are programmed.

CHANNELS WITH INSUFFICIENT DATA ARE: nn mm .....
The channel triggered but did not take 100 or more points.

THE NUMBER OF IRIG TIME WORDS (xxxxxx) IS NOT CORRECT
Should have four time words. More or less is an error.

THERE WERE xxx CHANGE WORDS - SHOULD HAVE NONE
Fixed rate data should have no change words.

DATA POINTS AT RATES NOT SELECTED WERE FOUND
All data should be at rate = B.

TIME WORDS ANALYSIS BYPASSED DUE TO MISSING TIME WORDS
If less than four time words are present, the time words analysis is bypassed.

IRIG TIME LOCATION OF xxxxxx IS NOT CORRECT
The time words group does not appear at words 79-83 as required.

TRIGGER POINT IS AT xxxxxx NOT AT EXPECTED VALUE
The transition through the threshold value with the proper slope was found at words 3, 4 or 8-10 following the trigger time but not at the expected value of 5-7.

TRIGGER POINT NOT FOUND WITHIN EXPECTED INTERVAL
The transition through the threshold value with the proper slope was not found at words 3-10 following the trigger time.

APPENDIX O - ANALYTICAL AND UTILITY PROGRAMS' OPERATION INSTRUCTIONS

BLAST	LIRT
BLKTR	LUMAP
CDIFF	MCCHK
CHANG	MKMNU
CHECK	ODLST
CHF	OREAD
CHFRM	OVRLD
CTEMP	PG316
DGCHK	PG416
DISCS	PING
DPLOT	PRMCL
DRCHG	QPEAK
FILOC	QPLOT
FILTR	RMS
FNDTM	SHIFT
FUDGE	TAPER
GAGEF	TBLGN
GETIT	TEMP
GPIB	THERM

A description of each program is stored in a file on the BTST. The name of the file is Pxxxxx::16 where xxxxx is the program name.

PROGRAM BLAST

Program BLAST processes blast overpressure data to provide the ambient value for points 2-21, the positive peak pressure, the A-duration, the impulse of the initial positive portion of the waveform, and the B-duration. An option is available to change the start and stop point of the ambient calculation.

The run string for BLAST is:

```
RU,BLAST,intrctv lu(dflt=1),list lu(dflt=intrctv lu),
         disc data lu(dflt=intrctv),
         data start track(dflt=intrctv),
         change ambient limits(dflt=no)
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

disc data lu: disc lu where data is located

data start track: disc track where data to be processed starts

change ambient limits = -99: request operator to enter new start and stop limits

change ambient limits # -99: use program limits

The list format used is:

```
RND NO. D28

M72-750 LAW -- HOT -- +140 DEGREE F

CH  AMBIENT  SD AMBNT  A-DUR   B-DUR    IMPULSE    PEAK    UNITS
NO  (CNTS)    (CNTS)  (MSEC)  (MSEC)  (MS-UNITS)
 1     -3       .000     .94    8.03       9.17   41.36   KPA
 2     -2       .000    3.77   16.92    2152.70 1028.15   PASCALS
```

The following abbreviations are used in the printout:

CH	= channel number
AMBIENT	= ambient(average) value of data from words 2-21(counts)
SD AMBNT	= standard deviation in ambient(counts)
A-DUR	= A-duration time in milliseconds
B-DUR	= B-duration time in milliseconds
IMPULSE	= integral of positive portion of waveform in units of milliseconds*pressure
PEAK	= positive peak: (IMAX-IAMBT)*scale factor
UNITS	= vertical axis plot label

PROGRAM BLKTR

Program BLKTR allows a block of records on a BTST data lu to be transfered to another data lu. The source data disc lu, the destination data disc lu, the start record number and the stop record number are entered interactively. Then program FILTR is repeatedly scheduled to accomplish the task.

The run string for BLKTR is:

```
RU,BLKTR,intrctv lu(dflt=1)
```

intrctv lu: interactive operator terminal

PROGRAM CDIFF

Program CDIFF calculates the point by point difference between two channels in the same data record and saves the difference in a new ADCHK compatible data record. In order for the difference to be correct the two channels must be fixed rate with the same rate and must have triggered at the same time. The program checks for these conditions prior to starting. The program uses the space above the last directory entry for the temporary storage of intermediate results. The program calculates the amount of space it requires for the new data record and the temporary storage prior to starting and checks that adequate disc space is available. Normally every available data point is differenced, however an option exists to set an offset and number of points to change the difference limits. Another option is available to write the new record to a different disc lu than the original data came from.

The data record created by CDIFF is identical to a record created by ADCHK except that there are no event words. The parameters used are those of the first channel except that the scale factor is changed, the plot label is changed and the remarks entry is blanked. Any blanks in the round number header are converted to *s.

The run string for CDIFF is:

```
RU,CDIFF,intrctv lu(dflt=1, -lu=change difference limits),
        data disc lu(dflt=intrctv, -lu=change destination lu),
        data start track(dflt=intrctv, -1=last directory entry),
        channel 1,channel 2(if both channels are defaulted,
          then channel entry is intrctv)
```

intrctv lu: interactive operator lu

disc data lu: disc lu where data is located(if positive also location of new record)

data start track: disc track where data to be processed starts

channel 1: minuend channel

channel 2: subtrahend channel

PROGRAM CHANG

Program CHANG allows words in an ADCHK data record to be changed on an individual basis. If a glitch or bad data words are present in a record they may cause considerable problems. Using CHANG the problems may be removed or modified. Use of CHANG should be restricted. If it is used, a comment should be entered in the comment section of the channel's documentation.

Since CHANG is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%CHANG.

The run string for CHANG is:

```
RU,CHANG,intrctv lu(dflt=1)
```

intrctv lu: interactive operator terminal

PROGRAM CHECK

Program CHECK provides a count of the number of data, change and trigger time words and a breakdown of data words at each rate for a set of points from a BTST data record. In addition, the program determines if there are missing change words, double change words or trigger time words not in groups four. The check may be done for a single channel or for all channels in a record. An offset and number of words entry allow either all or part of the data record to be processed.

The run string for CHECK is:

```
RU,CHECK,intrctv lu(dflt=1),list lu(dflt=intrctv lu)
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

The list format used is:

```
DISC LU=  19 TRACK=    2 OFFSET=      0 POINTS=  1000

    CH  2 DATA=    997  CHG=      0  TIME=     4
           MCW=      0  DCW=      0  TGRP=     0
RATES 0-7:       0       0       0       0       0       0       0       0
RATES 8-F:       0       0       0       0       0     997       0       0
```

The following abbreviations are used in the printout:

- CH = channel number
- DATA = number of data words
- CHG = number of change words
- TIME = number of trigger time words
- MCW = number of missing change words
- DCW = number of double (or more) change words
- TGRP = number of trigger time words not in groups of four

PROGRAM CHF

Program CHF converts a version II data record into a version III data record. All of the information in the version II record is saved in the version III record. However, since there is information in the version III record which does not exist in the version II record some of the entries in the version III format must be used carefully. The new record created is entered in the directory of the destination data disc.

The run string for CHF is:

```
RU,CHF,intrctv lu(dflt=1),source data lu(dflt=intrctv),
          source start track(dflt=intrctv),
          destination data lu(dflt=intrctv),'
```

intrctv lu: interactive operator terminal

source data lu: disc lu where record to be converted is located

source start track: disc track where data record to be converted starts

destination data lu: disc lu where new record is to be created

PROGRAM CHFRM

Program CHFRM converts a version I data record into a version II data record. All of the information in the version I record is saved in the version II record. However, since there is information in the version II record which does not exist in the version I record some of the entries in the version II format must be used carefully. An FMGR file named [PLOTS is read by the program to get the labels used on the plots. If the original [PLOTS for the data is not available a new version will have to be created to make CHFRM run. An opportunity to read the [PLOTS record before starting the conversion process is provided. The new record created is entered in the directory of the destination data disc.

The run string for CHFRM is:

```
RU,CHFRM,intrctv lu(dflt=1),source data lu(dflt=intrctv),
        source start track(dflt=intrctv),
        destination data lu(dflt=intrctv),
        check [PLOTS file(dflt=yes)
```

intrctv lu: interactive operator terminal

source data lu: disc lu where record to be converted is located

source start track: disc track where data record to be converted starts

destination data lu: disc lu where new record is to be created

check [PLOTS file = 0: display [PLOTS file and wait for operator response

check [PLOTS file # 0: use [PLOTS file without checking

PROGRAM CTEMP

Program CTEMP provides temperature display from a single thermocouple channel. The thermocouple must be attached to a type 005 thermocouple reference junction bank installed in a 3495A scanner. The BTST DVM is used to make the voltage measurements. The program then calculates the temperature and displays it on the terminal. The program may be stopped by getting an operating system prompt and typing BR,CTEMP.

The run string for CTEMP is:

```
RU,CTEMP,intrctv lu(dflt=1)
```

intrctv lu: interactive operator terminal

PROGRAM DGCHK

Program DGCHK allows all of the internal commands of the digitizer to be executed from a terminal. Troubleshooting of ASRD problems can be carried out more easily with this program than with the standard data acquisition programs.

The run string for DGCHK is:

```
RU,DGCHK,intrctv lu(dflt=1)

  intrctv lu:  interactive operator terminal
```

The commands which may be used on data channels are:

```
AR              ARM ALL CHANNELS
AV              GET THE CURRENT AVERAGE AND NOISE VALUES
CC,val          SPECIFY THE CALCULATION CONTROL WORD
CH,chan         SPECIFY THE ADDRESSED CHANNEL
CV              GET THE CURRENT SIGNAL VALUE
DA              DMA DATA TRANSFER--AUTO PARAMETERS (*)
DD,val          SPECIFY THE DECREASE DELAY
DS              DMA DATA TRANSFER--SPECIFIED PARAMETERS (**)
DV              DISPLAY SIGNAL VALUE ON LED DISPLAY
EX              EXIT 'CK' MODE
FT              PERFORM FAST MEMORY TEST
GA              GET STARTING ADDRESS FOR AUTO TRANSFER (*)
GE              GET ERROR WORDS
GL              GET MEMORY FAIL LOCATION
GN              GET NUMBER OF WORDS FOR AUTO TRANSFER (*)
GR              GET REGISTER CONTENTS
GT              GET RATE TABLE
HA              HALT DATA ACQUISITION MODE ALL CHANNELS
LC,list         LIST COMMANDS TO LIST
LD              LOAD PARAMETERS TO CHANNEL REGISTERS
MF              FILL MEMORY WITH SEQUENTIAL NUMBERS (**)
MT              EXECUTE MEMORY TEST
OF              TURN CHANNEL OFF WRT TO DATA ACQUISITION
ON              TURN CHANNEL ON WRT TO DATA ACQUISITION
PA              PROG DATA TRANSFER--AUTO PARAMETERS (*)
PS              PROG DATA TRANSFER--SPECIFIED PARAMETERS (**)
PT,val          SET PRETRIGGER SIZE (1 K BLOCKS)
RE              RESET MICROPROCESSOR ALL CHANNELS
RD,val          SET RUN DELAY
RT              REGISTER TESTS
SA,v1,v2        SET STARTING ADDRESS (**)
SD,val          SET STOP DELAY
SN,v1,v2        SET NUMBER OF WORDS TO TRANSFER (**)
ST              SET RATE TABLE
SY              ISSUE SYNC TO ALL CHANNELS
SZ,val          SET MEMORY SIZE (1 K BLOCKS)
S#              GET CHANNEL SERIAL NUMBER AND SW REV CODE
TG              GET TRIGGER SET INDICATOR
TM,val          SET TRIGGER MODE AND THRESHOLD
TR,val          SET TRACE FLAG (0=OFF, 1=ON)
ZE              ZERO ERROR WORDS
```

The commands which may be used on the events channel are:

```
DA              READ DAY OF YEAR
EV              READ EVENTS
NE              READ NUMBER OF EVENTS
TI              READ CURRENT TIME
```

PROGRAM DISCS

Program DISCS makes a copy of data stored on disc to a nine-track magnetic tape for backup purposes. A header may be placed on the tape for identification purposes. More than one save may be done on a tape, but the operator is responsible for tape positioning.

The run string for DISCS in the interactive mode is:

RU,DISCS,intctv lu(dflt=1), mag tape lu(dflt=8)

intrctv lu: interactive operator terminal

mag tape lu: nine-track magnetic tape drive lu

The run string for DISCS in the non-interactive mode is:

RU,DISCS,data disc lu,mag tape lu,start track(dflt=trouble),
number of tracks,file number

data disc lu: disc lu where data is located

mag tape lu: nine-track magnetic tape drive lu

start track: first data disc track to be saved - a value of zero is not allowed

number of tracks: number of tracks to be saved

file number: identifying number to be written on tape record

PROGRAM DPLOT

Program DPLOT provides formal plots of BTST data records. The program has a number of options which allow different display formats for the data. The program can draw plots on the 7245 printer/plotter or optionally on a H-P 9872 or 7225 four pen plotter. Every data point in the plot interval is plotted. This technique requires considerable time but produces an accurate plot. The plots are generated with horizontal and vertical tic mark graduations spaced one inch apart so that an engineer's scale may be used. The program has three modes of operation: non-interactive, partially interactive and fully interactive. In the non-interactive mode, the data disc lu, data start track, plot duration and channel number are specified in the run string. In the partially interactive mode, this information plus the plot device lu are entered interactively. In the fully interactive mode, the above information plus start time, new maximum and minimum scale values, new labels, new scale factor, new ambient and line type may be entered. Also in this mode, multiple plots may be made on the same piece of paper. It should be noted that when new maximum and minimum values or time durations are used that the program will expand the values to obtain convenient integer tic marks. Also peak values should be greater than three and less than 6000 otherwise a scaling factor of 0.001 or 1000 will be used. All time entries should be in milliseconds. Three types of time coordination are possible for multiple plots: match first word, match trigger word and operator supplied offset. When operator supplied offset is used a positive offset value moves the record to the right. Also, if a start time other than zero was used on the first trace, then the offset of that trace equals the negative of the start time used.

The run string for DPLOT is:

```
RU,DPLOT,intrctv lu(dflt=1),plot control,
         disc data lu(dflt=intrctv),
         data start track(dflt=intrctv),
         channel number
```

intrctv lu: interactive operator terminal

plot control > 0: non-interactive mode - entry is used as plot duration
plot control = 0: partially interactive mode
plot control < 0: fully interactive mode

disc data lu: disc lu where data is located

data start track: disc track where data to be plotted starts

channel number: channel to be plotted

The available line types are:

NO	TYPE
1	solid line
2	dots
3	dashes
4	alternate short & long dashes
5	point plot

PROGRAM DRCHG

Program DRCHG allows an ADCHK data directory to be truncated or restored by an operator. The program prints the current number of files and last file start track. Then the operator is given an opportunity to change the entries to new values. The program does not check or validate the new entries in any way.

Since DRCHG is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%DRCHG.

The run string for DRCHG is:

RU,DRCHG,intrctv lu(dflt=1)

intrctv lu: interactive operator terminal

PROGRAM FILOC

Program FILOC finds and displays the disc lu, track, sector, file type and file size of an FMGR file. Note that the file size is in sectors and that the information listed applies to the body of the file and not to any extents associated with the file.

Since FILOC is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%FILOC.

The run string for FILOC is:

RU,FILOC,intrctv lu(dflt=1)

intrctv lu: interactive operator terminal

PROGRAM FILTR

Program FILTR transfers a BTST record from one data disc lu to another data disc lu. The record number is used to located the data on the source lu and the program creates a directory entry for the new record on the destination lu. The record on the source lu is not modified in any way.

The run string for FILTR is:

RU,FILTR,intrctv lu(dflt=1),source data lu(dflt=intrctv),
destination data lu,record number

intrctv lu: interactive operator terminal

source data lu: disc lu where record to be transfered is located

destination data lu: disc lu where record is to be transfered

record number: the number of the record to be transfered in the in the source lu directory

PROGRAM FNDTM

Program FNDTM locates the first occurance of the trigger time words for each channel in a BTST data record. The program then provides the following information: the trigger time, the offset word, the absolute time at the sample preceding the trigger time words, the word number of the first trigger time word, and the elapsed time from the first data sample in the record to the word preceding the trigger time words.

The run string for FNDTM is:

```
RU,FNDTM,intrctv lu(dflt=1),list lu(dflt=intrctv lu),
        data disc lu(dflt=intrctv),
        data start track(dflt=intrctv, -1=last directory entry)
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

data disc lu: disc lu where data is located

data start track: disc track where data record to be processed starts

The list format used is:

CH	HR	MN	SECONDS	OFST	PREV SMPL	WORD NO	TIME(MILLISEC)
0	7	16	34.769415	5	34.769409	81	1.600
1	7	16	34.769421	10	34.769408	81	1.600
3	7	16	34.769416	6	34.769408	81	1.600
7	7	16	34.769417	7	34.769408	81	1.600

The following abbreviations are used:

CH - channel number
HR - hours of trigger time word
MN - minutes of trigger time word
SECONDS - seconds of trigger time word
OFST - offset of trigger time word
PREV SMPL - absolute time at sample previous to first trigger time word. Note that hours and minutes are not listed. Depending on size of offset they may be different from trigger word hours and minutes.
WORD NO - word number of first trigger word in record.
TIME - elapsed time from beginning of record to sample previous to first trigger time word.

PROGRAM FUDGE

Program FUDGE allows the scale factor, plot label and y-axis label in a BTST data record to be permanently changed. The old values may be reviewed prior to making a change. If desired all records on a data lu may be changed to the new values at the same time. If this is done, the old value on each record to be changed may be compared to the initial value read from the first record read to guarantee that the data to be changed is constant. Each record changed has the word FUDGE written over the gage factor units to document the change. It is recommended that any data changed by this program be backed up prior to initiating any changes as there is no guarantee that the data record can be restored to its original configuration.

The run string for FUDGE is:

RU,FUDGE,intrctv lu(dflt=1),bypass check

intrctv lu: interactive operator terminal

bypass check = -99: bypass check of old values before changing

bypass check # -99: perform check of old values before changing

PROGRAM GAGEF

Program GAGEF provides a means of converting from a non-standard BTST gage factor to the standard format required by the PA command of ADCHK. The program provides an opportunity to obtain a printed record of the conversion for documentation. A number of possible conversions can be done with the program. The available unit options are:

Pressure in PSI or kPa
Force in pounds or newtons
Acceleration in g's
Strain(single arm)
User defined

The available conversion options are:

millivolts/xyz	to	xyz/volt	PCB gage
pCb/xyz	to	xyz/pCb	Kistler gage
millivolts/volt/xyz	to	xyz/volt/volt	CEC or Endevco gage
xyz/ohm	to	xyz/volt/volt	Rod Pull or Hat gage
K & Y	to	xyz/pCb	Tourmaline gage
BSLZ fit	to	xyz/...	Rod Pull or CEC gage
Microphone cal	to	xyz/...	B & K gage

The run string for GAGEF is:

RU,GAGEF,intrctv lu(dflt=1)

intrctv lu: interactive operator terminal

PROGRAM GETIT

Program GETIT allows system table areas to be read and displayed.

Since GETIT is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%GETIT.

The run string for GETIT is:

RU,GETIT,list lu(dflt=1),start address(dflt=stop),
number of words(dflt=1)

list lu: results list device(terminal, printer or tape)

start address: memory address to start reading tables(enter in octal by using a B, ie 1756B)

number of words: number of words to be read

NOTE: all parameter entry is by run string.

PROGRAM GPIB

Program GPIB provides a number of control functions for devices connected on the general purpose interface bus(GPIB). The program is not required for normal operation of the BTST, but is useful for trouble shooting problems of equipment connected to the bus.

The run string for GPIB is:

RU,GPIB,intrctv lu(dflt=1)

intrctv lu: interactive operator terminal

The commands supported by GPIB are:

```
DC,lu   - Device Clear
EN,blu  - ENter bus controller lu
EX      - EXit GPIB
FI,dlu  - FInd bus control lu from device dlu
GL,lu   - Go to Local
GT,lu   - Group execute Trigger
IC      - Interface Clear
LI,llu  - List GPIB commands on list lu
LL      - Local Lockout
LC      - Lockout Clear
RE,lu   - Remote Enable
ST,dlu  - STatus byte read from device dlu
UC      - Universal Clear
UU      - Untalk/Unlisten
WR,dlu  - WRite message to device dlu
```

where lu = device lu or bus controller lu

PROGRAM LIRT

Program LIRT allows a rate table stored on an ADCHK data disc to be output to a list device. The RT command of ADCHK does not allow listing to a printer. The decrease delay, length, scalar and rate table entries are all listed.

Since LIRT is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%LIRT.

The run string for LIRT is:

```
RU,LIRT,intrctv lu(dflt=1),list lu(dflt=intrctv),
        disc lu(dflt=intrctv)
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

disc lu: data disc lu where rate tables are stored

PROGRAM LUMAP

Program LUMAP provides a configuration list for the computer hardware. All currently defined LUs, EQTs, subchannels, select codes, driver types, DMA and buffering capabilites and status are provided. The list is valid for equipment generated into the system but not for devices using off-line drivers. Thus, the ASRD will not appear in the list generated by LUMAP.

The run string for LUMAP is:

```
RU,LUMAP,list lu(dflt=1)
```

list lu: results list device(terminal, printer or tape)

The list format used is:

LU	EQT	SUBCHNL	SELECTCODE	DVR	DMA	BUF	STATUS	DEVICE
1	2	0	16	05	0	1	2	SYS TERM
2	1	0	11	32	1	0	0	SYS DISC
3	3	0	21	00	0	1	0	TERMINAL
4	2	1	16	05	0	1	2	L. CAS

The following abbreviations are used:

LU - logical unit
EQT - equipment number
SUBCHNL - equipment subchannel number
SELECT CODE - computer I/O card number
DVR - software driver name
DMA - does device have direct memory access capability? (0 = no, 1 = yes)
BUF - does device use an output buffer capability? (0 = no, 1= yes)
STATUS - what is current device status? (0 = available, 1 = down, 2 = busy, 3= no DMA channel)
DEVICE - device description

PROGRAM MCCHK

Program CHCHK allows a simple check of the H-P 12566 microcircuit interface card to be made. A diagnostic test hood must be installed on the card or on the cable prior to starting the test. The diagnostic hood should have all 16 input lines connected to the respective output lines. The program outputs 32 words to the card and compares the input word to the output word. The first 16 words are a one bit moving across a field of zeroes while the second 16 words are a 0 moving across a field of ones. The test does not check the interrupt circuitry or timing of the card.

The run string for MCCHK is:

```
RU,MCCHK,intrctv lu(dflt=1)
```

intrctv lu: interactive operator terminal

PROGRAM MKMNU

Program MKMNU takes type 3 file MENU created by program EDITR and converts it into a type 1 file @MENU. ADCHK commands are entered alphabetically in file MENU and reformated for double column presentation in file @MENU.

Since MKMNU is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%MKMNU.

The run string for MKMNU is:

```
RU,MKMNU,intrctv lu(dflt=1)
```

intrctv lu: interactive operator terminal

PROGRAM ODLST

Program ODLST provides a directory list of data records saved in version I format. The version I directory format is different from versions II and III(which are identical). To use program CHFRM the start tracks of data records must be known. This program provides a means of reading the record's location.

The run string for ODLST is:

```
RU,ODLST,intrctv lu(dflt=1),stop record number(dflt=95),
        list lu(dflt=intrctv lu),data disc lu(dflt=intrctv),
        data start track
```

intrctv lu: interactive operator terminal

stop record number: stop directory list when this record is reached

list lu: results list device(terminal, printer or tape)

data disc lu: disc lu where data is located

data start track: disc track where data record starts

PROGRAM OREAD

Program OREAD provides the capability to read data from a disc lu and display it. The display mode may be set to integer, octal or real format. In addition, the ASCII equivalent of the formatted data is always displayed on the right side of the listing. The program can be used on disc lus with either 96 or 128 sectors per track, but the operator must make the correct choice as the program does not change automatically.

The run string for OREAD is:

RU,OREAD,intrctv lu(dflt=1),disc lu(dflt=intrctv),disc track,
disc sector,display mode(dflt=octal)

intrctv lu: interactive operator terminal

disc lu: lu of disc to be read

disc track: disc track where data is to be read

disc sector: disc sector where data is to be read

display mode: ASCII word - "O ", "I " or "R " to set initial display mode

The commands supported by OREAD are:

EX	- EXit OREAD.
LI	- LIst the OREAD commands and the current number of sectors per track setting.
LL,lu	- Set List Lu to lu.
LU,l,t,s	- Set disc LU to l, track to t and sector to s. If no entry is made for t and s they will remain the same.
MO,x	- Set MOde to x where x = I, O, R (Integer, Octal or Real).
NS	- Switch the Number of Sectors per track to the alternate value. Initially NS is set to 96. It may be switched to 128.
SE,s	- Set disc SEctor to s.
TR,t,s	- Set disc TRack to t and sector to s. If no entry is made for s it will remain the same.
n	- Increment the sector by n relative to the bottom of the display.
-n	- Decrement the sector by -n relative to the top of the display.
cr	- A carriage return(cr) will cause the display of the next two sectors in sequence. Track boundaries are crossed automatically.

PROGRAM OVRLD

Program OVRLD allows overloads on the Precision Filters model 416 filter/amplifier to be read, read and displayed, or read, displayed and passed back to the calling program. The 416 has four overload codes: 0 = no overload, 1 = prefilter overload, 2 = postfilter overload and 3 = both pre- and postfilter overload.

The run string for OVRLD is:

RU,OVRLD,list lu(dflt=1),display mode

list lu: results list device(terminal or printer)

display mode = 0: output the overload display even if no channels are overloaded
display mode = 1: no display even if channels are overloaded (used to clear overloads)
display mode = 2: output the overload display only if a channel is overloaded
display mode = 3: same as 2 except also return overload codes to calling program -- returns four words (two bits per channel)

PROGRAM PG316

Program PG316 programs the Precision Filters model 316 signal conditioner. Initially all channels are addressed and sent the following commands: turn calibrator off, set cal step to zero, set strain gain to 1 and set manual balance. Parameters are read from the parameter track of the data lu. The parameters are checked to determine that they are allowed values. The program then sends valid parameters to the channel. If the mode is not valid, no programming is done. If the gain is not valid, gain is set to 1. If the balance is not valid, the balance is set to manual. A warning message is sent to the operator if invalid parameters are found.

The run string for PG316 is:

RU,PG316,intrctv lu(dflt=1),data disc lu(dflt=intrctv)

intrctv lu: interactive operator terminal

data disc lu: data disc where parameters are located

PROGRAM PG416

Program PG416 programs the Precision Filters model 416 filter/amplifier. Parameters are read from the parameter track of the data lu. The parameters are checked to determine that they are allowed values. The program then sends the prefilter gain, the postfilter gain and the cutoff frequency to the 416 over the GPIB. If a channel is not selected or does not have valid parameters then the gain is set to 1 and the frequency is set to 5kHz. A warning message is sent to the operator if invalid parameters are found.

The run string for PG416 is:

```
RU,PG416,intrctv lu(dflt=1),data disc lu(dflt=intrctv)
```

intrctv lu: interactive operator terminal

data disc lu: data disc where parameters are located

PROGRAM PING

Program PING provides a changing display pattern on the computer front panel LEDs. The display serves no function other than an activity indicator for the computer. The priority of PING is set to 32767 (the lowest priority) so that its operation does not interfere with the functioning of any other programs. Normally PING is started running from the WELCOM file.

The run string for PING is:

```
RU,PING          from the operating system or
SYRU,PING        from FMGR
```

PROGRAM PRMCL

Program PRMCL zeroes the unused portions of the parameter track on an ADCHK data disc lu. Although this is not necessary for normal operation, it allows a set of known values to be placed in these locations. The used parameter locations are not disturbed.

Since PRMCL is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%PRMCL.

The run string for PRMCL is:

```
RU,PRMCL,intrctv lu(dflt=1)
```

intrctv lu: interactive operator terminal

PROGRAM QPEAK

Program QPEAK provides a list of a number of pieces of information which are useful for determining overall data record quality. For each channel in a data record, QPEAK determines the number of words recorded, the gain used, the average value at points 2-21, the minimum data value, the maximum data value, the scaled largest positive excursion from ambient, the scaled largest negative excursion from ambient, and the units used. This information allows an operator to quickly evaluate that a channel took the correct amount of data, did not clip, and has the correct peak values. In addition, the program reads the round number from the header, finds the trigger time from the first channel, compares it to the day and month from the header, and flags the list if they are different. The maximum and minimum data value and the ambient value calculated by QPEAK are stored in the data record for future use(if not already written).

The run string for QPEAK is:

```
RU,QPEAK,intrctv lu(dflt=1),list lu(dflt=6),
         disc data lu(dflt=intrctv),
         data start track(dflt=intrctv, -1=last directory entry)
         recalculate max & min(dflt=no)
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

disc data lu: disc lu where data is located

data start track: disc track where data to be processed starts

recalculate max & min = -99: recalculate maximum & minimum and write to data record

recalculate max & min # -99: use existing maximum & minimum values if possible

The list format used is:

```
                ROUND NO.    CHK1
                FIRED AT    7: 7 ON  9- 14- 83
           ***** IRIG DATE IS  9 SEP
                STORED ON LU 20 TRACK    2

CH   WRDCNT      GAIN   IAMBT    IMIN    IMAX    PEAK+     PEAK-   UNITS
 0    2770         1      62   -1674    1770    33.91    -34.46   VOLTS
 1    2787         1      86   -1650    1794    33.84    -34.39   VOLTS
 2    2770         1      62   -1670    1770    33.90    -34.38   VOLTS
```

The following abbreviations are used in the printout:

- ***** = header date does not agree with trigger time date
- CH = channel number
- WRDCNT = number of data words
- GAIN = total gain used
- IAMBT = ambient(average) value of data from words 2-21 in counts
- IMAX = largest integer count
- IMIN = smallest integer count
- PEAK+ = positive peak: (IMAX-IAMBT)*scale factor
- PEAK- = negative peak: (IMIN-IAMBT)*scale factor
- UNITS = vertical axis plot label

PROGRAM QPLOT

Program QPLOT provides quick plots of BTST data records. The program has a number of options which allow different display formats for the data. The program can draw plots on either the 2648 graphics terminal or the 7245 printer/plotter. To provide quick plot response the program does not necessarily plot all of the points in a channel's data record. Instead only a fraction of the points in a record are plotted to reduce the total plot time. The data record to be displayed is broken up into a number of equal sized bins. The number of bins is determined by dividing the record size by the resolution of the display device(600 for the 2648 and 1000 for the 7245). Within a bin the program scans through all of the points to find the minimum and maximum in the bin. The point with the largest absolute value is plotted as the bin representative. Using this scheme provides a reasonable representation for most BTST data. However, it can be misleading at times. For example, a sine wave record may end up being plotted as a straight line since the peak of the sine wave may end up as the largest point in every bin. Basically the processing technique operates very much like a low pass filter: any information shorter than a bin width will be lost on the plot. The operator can control the initial offset into the data record and the number of record data points to be plotted on each plot through an interactive response. Once entered these values are used for each channel to be plotted. Normally the plot vertical axis is scaled based on the record scale factor and the largest and smallest points in the data record. However, options exist to make each plot self scaling and to display data in volts rather than scaled units.

If the entire record is plotted, the maximum, minimum and ambient values stored in the data record will be used if available. If not available, the program will calculate them and store them in the data file for later use. The time required to draw a plot varies considerably depending on the number of points in the record and on whether the maximum, minimum and ambient have been previously calculated.

The run string for QPLOT is:

```
RU,QPLOT,intrctv lu(dflt=1, -lu=self scale),plot device,
         line type(dflt=solid, -line=volts scaling),
         disc data lu(dflt=intrctv),
         data start track(dflt=intrctv, -1=last directory entry)
```

intrctv lu: interactive operator terminal

plot device = 1: 7245 printer/plotter(lu 7)
plot device # 1: 2648 graphics terminal

line type: line type to use on plot

disc data lu: disc lu where data is located

data start track: disc track where data to be plotted starts

The available line types are:

NO	2648	7245
1	solid line	solid line
2		solid line
3		solid line
4	alternate short & long dashes with big space	alternate short & long dashes with big space
5	long dash	long dash
6	short dash	short dash
7	dots	dots
8	alternate short & long dashes with short space	alternate dot & dash
9	3 short dashes then big space	solid line
10	long & 2 short dashes	long & 2 short dashes
11	point plot	point plot

PROGRAM RMS

Program RMS calculates the rms (root-mean-square) value, the largest value, the smallest value, the average value and the time interval for a set of data words in a BTST data record. The calculation may be done for a single channel or for all channels in a record. An offset and number of words entry allow either all or part of the data record to be processed.

The run string for RMS is:

RU,RMS,intrctv lu(dflt=1),list lu(dflt=intrctv lu),

ac or dc rms(dflt=ac)

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

ac or dc rms: <0 perform a dc rms calculation, otherwise perform an ac rms calculation

The list format used is:

DISC LU= 19 TRACK= 2 OFFSET= 1000 #POINTS= 2000

CH	AC RMS	YMAX	YMIN	TSUM	AVERAGE
1	3.054	.976	-8.652	10000.	-5.820

The following abbreviations are used in the printout:

CH = channel number
RMS = root-mean-square value of data set with AC or DC indicating type of calculation
YMAX = largest value in data set
YMIN = smallest value in data set
TSUM = time interval of data set
AVERAGE = average value of data set

PROGRAM SHIFT

Program SHIFT calculates the average value of points 2-21 of each channel in a data record, repeats the process for another data record and then differences the two records. The program lists the average for the first record's channels, the average for the second record's channels, the difference between the records in counts, the scaled difference between the channels and the units. An option to change the start and stop points for performing the average is available.

The run string for SHIFT is:

```
RU,SHIFT,intrctv lu(dflt=1),list lu(dflt=6),
        change average limits(dflt=no)
```

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

change average limits = -99: request operator to enter new start and stop limits

change average limits # -99: use program limits

The list format used is:

```
ZERO LEVEL FROM ROUND NO.   TR01
                FIRED AT    7:16 ON 14SEP83
                STORED ON LU 20 TRACK   36

SHIFTED DATA FROM ROUND NO.   TR02
                FIRED AT    7:21 ON 14SEP83
                STORED ON LU 20 TRACK   38

CH   IZERO    ILEVEL     ISHIFT      SHIFT    UNITS
 0     114        63        -51      -1.01    VOLTS
 1      43        87         44        .87    VOLTS
```

PROGRAM TAPER

Program TAPER restores a copy of data saved on a nine-track magnetic tape to disc for backup purposes. The header must be manually checked for verification purposes. More than one save may be stored on a tape. If so, the operator is responsible for tape positioning.

The run string for TAPER is:

```
RU,TAPER,intctv lu(dflt=1), mag tape lu(dflt=8)
```

intrctv lu: interactive operator terminal

mag tape lu: nine-track magnetic tape drive lu

PROGRAM TBLGN

Program TBLGN is used to create standard rate tables 0-15 on the ADCAK data disc lu. Normally this needs to be done when a new disc is installed or the tables on an old disc have been destroyed. When the program is run it generates a list of all of the tables. Then the operator has an opportunity to write the tables to the data lus if desired.

Since TBLGN is not used very often, it is not loaded permanently on disc. To load the program: RU,LOADR,,%TBLGN.

The run string for TBLGN is:

RU,TBLGN,intrctv lu(dflt=1),list lu(dflt=intrctv lu)

intrctv lu: interactive operator terminal

list lu: results list device(terminal, printer or tape)

PROGRAM TEMP

Program TEMP converts thermocouple voltage measurements made by program THERM to temperature readings. The program reads the raw data from disc, converts to temperature and then writes the temperature to a new record on disc.

The run string for TEMP is:

RU,TEMP,intrctv lu(dflt=1)

intrctv lu: interactive operator terminal

PROGRAM THERM

Program THERM provides temperature measurement with up to 19 thermocouple channels. The thermocouples must be attached to a type 005 thermocouple reference junction bank installed in a 3495A scanner. The BTST DVM is used to make the voltage measurements. The program sequentially scans the selected channels and saves the results on disc. The program uses software timing so all other system activity must be terminated and no interruptions in the program can be allowed. The program may be stopped by getting an operating system prompt and typing BR,THERM. Data is stored on disc in a non-multiplexed format. The number of channels determines the number of measurements made on a channel. The data buffer provides 6112 measurements for one channel. This is reduced to 320 measurements on each channel if all 19 channels are used. The raw measurements are stored on disc not temperature readings. Program TEMP must be used to convert the raw readings to temperatures. The system records at a rate of 0.18 seconds per reading.

The run string for THERM is:

RU,THERM,intrctv lu(dflt=1)

intrctv lu: interactive operator terminal

APPENDIX P - SUBROUTINE DESCRIPTIONS

FORTRAN callable subprograms written in FORTRAN.

ANSWR	DELAY	LABL	RPTER
ARMOT	DFLT	LOOKS	RTMNB
ATTNN	DGIO	LSQFT	RTMNS
AXIS	DRAW	MCVRT	SCALE
BELL	EDIT	MLBL	SCHDL
BITS	ERMSG	NOCHL	SETUP
CALPR	EQTIM	NOTES	TCHNG
CHADD	GETLU	OTDMA	TMCVT
CHCNT	GRAF2	PARAM	TRANS
CHLOC	GSCTR	PRHDR	TRTIM
CHSCN	GTLTC	PRMOT	VIEW
CKCHL	GTSCL	RAMBT	VIO
CKDEV	HMRKR	RATBL	VLTIN
CKLST	INSPT	RBASE	WORD
CKOVL	IYT2	RDMAR	WSCNR
CLEAR	JBCD	RDVM	ZERO
CNTRL	JDATA	REVNT	ZSPLN

FORTRAN callable subprograms written in assembler

IASRD	IOPSY	MCKT

FORTRAN CALLABLE SUBPROGRAMS WRITTEN IN FORTRAN

SUBROUTINE ANSWR(LU,KDEV,ICODE,IENTRY,RVAL,IVAL), REV*C 04DEC83
$ CLF

THIS SUBROUTINE IS USED TO PROCESS ANSWERS ENTERED FOR THE EN COMMAND OF CALX. IT PROMPTS THE OPERATOR, READS THE REPLY, THEN CHECKS TO SEE IF THE REPLY IS VALID. IF THE REPLY IS NOT VALID, THEN THE OPERATOR IS GIVEN A ERROR MESSAGE AND READS THE NEW REPLY. ENTRIES ARE CHECKED TO DETERMINE OF THEY ARE WITHIN 5% OF THE EXPECTED VALUE.

SUBROUTINE ARGUMENTS:

LU - INPUT TO SUBROUTINE SPECIFYING GRAPHICS TERMINAL LU

KDEV - INPUT TO SUBROUTINE INDICATING WHICH DEVICE TO PROCESS:
1=DIGITIZER 2=SGNL COND 3=FLTR/AMP 4=OFST/AMP 5=ASRD TIME

ICODE - OUTPUT OF SUBROUTINE INDICATING WHAT ACTION SHOULD BE TAKEN BY CALLING PROGRAM:
0=NEW ENTRY RETURNED 1=FINISHED(SAVE) 2=KILL(DONT SAVE)

THE FOLLOWING ARGUMENTS ARE VALID ONLY WHEN ICODE = 0 (NEW ENTRY)

IENTRY - OUTPUT OF SUBROUTINE INDICATING WHICH ITEM IS TO BE UPDATED

RVAL - OUTPUT OF SUBROUTINE WHICH CONTAINS THE NEW REAL VALUE IF ITEM IS 3 OR LARGER

IVAL - OUTPUT OF SUBROUTINE WHICH CONTAINS THE NEW INTEGER VALUE

SUBROUTINE ARMOT, REV C 11DEC83 CLF

SUBROUTINE TO ISSUE ARM SEQUENCE TO ASRD.

SUBROUTINE ATTNN(LU,ICHAN,IERR), REV C 11DEC83 CLF

SUBROUTINE TO ADDRESS AN ASRD CHANNEL.

SUBROUTINE ARGUMENTS:

LU - ERROR LOGGING LU(INPUT)

ICHAN - CHANNEL TO BE ADDRESSED(INPUT)

IERR - ERROR CODE(OUTPUT)

```
  SUBROUTINE AXIS(LUPLT,START,TLOFST,
 $IYMIN,IXTIC,IYTIC,YMULT,XMULT) ,  REV B  15FEB83  WSW

THIS SUBROUTINE DRAWS THE STANDARD AXES AND PLOT
WINDOW FOR USE BY DPLOT.
DATA SCALING AND PLOTTER INITIALIZATION HAVE ALREADY

    VARIABLES USED ARE:

          LUPLT= LU OF PLOTTING DEVICE, EXECEPT THAT A NEGATIVE
                 NUMBER IS USED IF THE PLOTTER IS NOT A 7245

          START= TIME IN MILLISECONDS TO BEGIN PLOT
          TLOFST=NUMBER OF MILLISECONDS TO SUBTRACT
                 FROM TIME AXIS LABELS

          IYMIN=VALUE(IN ENGR. UNITS) OF BOTTOM Y TIC MARK
          IXTIC=NUMBER OF MILLISECONDS(OR MICROSECONDS IF XMULT=.001)
                 BETWEEN TIC MARKS ON TIME AXIS
          IYTIC=NUMBER OF ENGINEERING UNITS BETWEEN Y-AXIS TIC
                 MARKS MULTIPLIED BY YMULT

          YMULT=VALUE OF .001, 1.0,OR 1000. BY WHICH ALL AMPLITUDE
                 DATA ARE MULTIPLIED
          XMULT=VALUE OF .001, 1.0, OR 1000. BY WHICH ALL TIME
                 DATA ARE MULTIPLIED


        9 TIC MARKS 1" APART ON X AXIS

1000,6796                                          10144,6796
   |----|----|----|----|----|----|----|----|----|
   |                                             |
   |         1" = 1016 PLATTENS                  |
   |                                             |
   |                                             |
   |                                             |
   |  6 TIC MARKS 1"                             |
   |  APART ON Y AXIS                            |
   |                                             |
   |                                             |
   |                                             |
   |----|----|----|----|----|----|----|----|----|
1000,700                                           10144,700
```

```
   SUBROUTINE BELL(LU,N),  REV B  30MAR82  CLF

THIS PROGRAM CAUSES THE BELL ON A TERMINAL TO SOUND.

SUBROUTINE ARGUMENTS:

  LU      - LOGICAL UNIT OF TERMINAL (INPUT)
  N       - NUMBER OF TIMES BELL IS TO SOUND (INPUT)
```

SUBROUTINE BITS(IVAL,IBITS), REV B 1JUN82 CLF

THIS SUBROUTINE TAKES THE 16 BIT WORD IN IVAL AND BREAKS IT DOWN INTO THE 16 BITS WHICH ARE RETURNED IN IBITS.

SUBROUTINE ARGUMENTS:

IVAL - WORD TO BE BROKEN DOWN(INPUT)

IBITS - ARRAY CONTAINING 16 BITS OF IVAL(OUTPUT)
EACH WORD OF IBITS WILL BE AN INTEGER 0 OR 1
IBITS(1) = VALUE OF IVAL BIT 15
IBITS(2) = VALUE OF IVAL BIT 14
.
.
.
IBITS(16) = VALUE OF IVAL BIT 0

SUBROUTINE CALPR(ICH,LULST,IHDR), REV B 04DEC83 CLF

THIS SUBROUTINE DISPLAYS THE CALIBRATION PARAMETERS FOR A CHANNEL ON A LIST DEVICE.

SUBROUTINE ARGUMENTS:

ICH - CHANNEL NUMBER TO DISPLAY(INPUT)

LULST - LIST DEVICE LU(INPUT)

IHDR - HEADER DISPLAY SWITCH(INPUT)
IHDR = 1 - DISPLAY HEADER
IHDR # 1 - DO NOT DISPLAY HEADER

SUBROUTINE CHADD(LUDK,ISTRK,IADTBL), REV G 27MAY80

SUBROUTINE TO ACCESS THE DISC AND DETERMINE THE TRACK, SECTOR, AND WORD ADDRESSES FOR EACH CHANNEL IN THE APG BALLISTIC DATA ACQUISITION SYSTEM.

ARGUMENTS ARE DEFINED AS FOLLOWS:

LUDK	DISC LU(INPUT)
ISTRK	STARTING TRACK NUMBER(INPUT)
IADTBL	ARRAY CONTAINING THE RESULTS(OUTPUT):
	IADTBL(INDEX*4+1)=STARTING TRACK FOR CHAN INDEX
	IADTBL(INDEX*4+2)=STARTING SECTOR FOR CHAN INDEX
	IADTBL(INDEX*4+3)=STARTING WORD FOR CHAN INDEX
	IADTBL(INDEX*4+4)=NO OF WORDS COLLECTED FOR CHAN INDEX

SUBROUTINE CHCNT(IBUF,ISTRT,ISTOP,ICNT), REV A 29MAR82 CLF

THIS SUBROUTINE SCANS AN ASCII ARRAY TO FIND THE NUMBER OF CHARACTERS PRECEDING A STRING OF TRUNCATING ASCII SPACES. THE CHARACTER COUNT IS RETURNED.

SUBROUTINE ARGUMENTS:

IBUF - ASCII ARRAY TO BE SEARCHED(INPUT)
ISTRT - START CHARACTER IN ARRAY(INPUT)
ISTOP - STOP CHARACTER IN ARRAY(INPUT)
ICNT - CHARACTER COUNT

SUBROUTINE CHLOC, REV A 21APR82 CLF

THIS SUBROUTINE IS USED TO FIND THE POINTERS(WORD NUMBER IN THE DATA FILE) AND THE NUMBER OF WORDS FOR EACH CHANNEL IN A BTST DATA FILE.

ALL VARIABLES ARE PASSED THROUGH COMMON. THE VARIABLES ARE:

JBUF	- NOT USED
ICHAN	- NOT USED
LUDK	- DISC LU WHERE DATA IS LOCATED
ISTRK	- DISC START TRACK WHERE DATA IS LOCATED
LU	- LU FOR ERROR MESSAGES
PNTRS	- WORD NUMBER IN DATA FILE WHERE EACH CHANNELS DATA STARTS
WRDCNT	- NUMBER OF WORDS OF DATA IN FILE FOR CHANNEL

SUBROUTINE CHSCN(LU), REV*F 3NOV83 CLF

THIS SUBROUTINE PROVIDES CONTROL OF THE HP 3495A SCANNER FOR USE WITH THE BALLISTIC TEST SITE TERMINAL. THE SCANNER IS SET UP WITH FOUR DUODECADE SCANNING CARDS. ONE CARD HANDLES THE OUTPUTS OF THE FILTER-AMPLIFIER UNIT (16 BNC)AND FOUR EXTERNAL INPUTS(TWO BNC & TWO TWINAX). THE SECOND CARD DIRECTS THE OUTPUT OF THE FIRST CARD TO THE SCOPE, DVM & EXT OUTPUTS. THE THIRD AND FOURTH CARDS ARE USED FOR TEST FUNCTIONS AND ARE NOT DIRECTLY CONTROLLED BY THIS SUBROUTINE. PROVISION IS INCLUDED TO CONTROL TWO SEPARATE SCANNERS FOR A 32 CHANNEL TRAILER. IN THIS CASE, THE PROGRAM CAN CONTROL ONLY ONE SCANNER AT A TIME.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE LU(INPUT)

THE SCANNER JUMPERS HAVE BEEN SET TO ALLOW EACH DUODECADE TO FUNCTION INDEPENDENTLY.

SUBROUTINE CHSEL(LU,IDLU,LSTCH,NSELCH,ISELCT,IPRINT), REV A 10NO $V83 CLF

THIS SUBROUTINE DETERMINES THE CHANNELS WHICH ARE SELECTED ON A BTST PARAMETER LU.

SUBROUTINE ARGUMENTS:

LU	- INTERACTIVE LU FOR ERROR MESSAGES(INPUT)
IDLU	- DISK LU WHERE PARAMETERS ARE STORED(INPUT)
LSTCH	- LAST CHANNEL IN SYSTEM(INPUT)
NSELCH	- NUMBER OF SELECTED CHANNELS(OUTPUT)
ISELCT	- ARRAY CONTAINING LIST OF SELECTED CHANNELS(OUTPUT)
IPRINT	- PRINT CONTROL(INPUT) IPRINT = 0: DONT PRINT RESULTS IPRINT = 1: PRINT ONLY IF NO CHANNELS ARE SELECTED IPRINT = 2: PRINT ALL RESULTS

SUBROUTINE CKCHL(ICH,LSTCH), REV A 16JUL82 CLF

THIS SUBROUTINE CHECKS THE CHANNEL PARAMETER TO DETERMINE IF IT IS WITH IN THE ALLOWED RANGE. IF IT IS NOT, A VALUE OF -1 IS RETURNED. THE ALLOWED RANGE IS 0 TO LSTCH.

SUBROUTINE ARGUMENTS:

ICH - CHANNEL PARAMETER TO BE CHECKED(INPUT)
UNCHANGED IF VALID, -1 OTHERWISE(OUTPUT)

LSTCH - LAST CHANNEL ALLOWED(INPUT)

SUBROUTINE CKDEV(IDEV), REV B 04DEC83 CLF

THIS SUBROUTINE CHECKS THE IDEV INPUT TO DETERMINE IF IT IS A VALID ENTRY. VALID ENTRIES ARE: AL, DI, FA, OA, SC, TI.

IDEV - DEVICE PARAMETER TO BE CHECKED(INPUT)
UNCHANGED IF VALID, -1 OTHERWISE(OUTPUT)

SUBROUTINE CKLST(LIST), REV A 16JUL82 CLF

THIS SUBROUTINE CHECKS THE LIST LU TO DETERMINE THAT IT FALLS WITHIN THE ALLOWABLE RANGE OF DEVICES.

SUBROUTINE ARGUMENT:

LIST - LIST LU TO BE CHECKED(INPUT)
UNCHANGED IF VALID, -1 OTHERWISE(OUTPUT)

THE CURRENT ALLOWED DEVICES ARE:

LU 1 - SYSTEM TERMINAL
LU 3 - TERMINAL
LU 4 - LEFT CASSETTE
LU 5 - RIGHT CASSETTE
LU 6 - PRINTER
LU 8 - MAG TAPE
LU 9 - TERMINAL

SUBROUTINE CKOVL(LU416,ICHAN,IOVRLD), REV B 6FEB82 CLF

THIS SUBROUTINE READS AN OVERLOAD FROM ONE CHANNEL OF A PRECISION FILTERS 416/90067 FILTER AMPLIFIER.

SUBROUTINE ARGUMENTS:

LU416 - LU OF 416 TO BE READ(INPUT)

ICHAN - ASCII CHANNEL NUMBER TO BE READ(INPUT)

IOVRLD - LEFT HAND BYTE IS OVERLOAD CODE FOR CHANNEL (OUTPUT)
RIGHT HAND BYTE IS NULL

```
   SUBROUTINE CLEAR(INBUF,ISTART,ISTOP)

THIS SUBROUTINE PLACES ASCII BLANKS IN BOTH BYTES OF A BUFFER
STARTING AT WORD ISTART THROUGH WORD ISTOP.

SUBROUTINE ARGUMENTS:

 INBUF  - BUFFER WITH UNKNOWN CONTENTS(INPUT)
          BUFFER FILLED WITH ASCII BLANKS(OUTPUT)

 ISTART - START WORD IN INBUF TO BE CLEARED(INPUT)

 ISTOP  - STOP WORD IN INBUF TO BE CLEARED(INPUT)
```

```
   SUBROUTINE CNTRL(LU,LUDVC,LUBUS),  REV B  13APR83  CLF

THIS SUBROUTINE GOES TO THE SYSTEM TABLES TO FIND THE LU FOR THE
HP-IB CONTROLLER CARD WHEN GIVEN THE LU OF ANY DEVICE ON THE BUS.

SUBROUTINE ARGUMENTS:

  LU    - LU FOR LISTING ERROR MESSAGES(INPUT)

  LUDVC - LU OF A DEVICE ON THE BUS(INPUT)

  LUBUS - LU OF THE BUS CONTROLLER(OUTPUT)
```

```
   SUBROUTINE DELAY(N),  REV B  10OCT81  CLF

THIS SUBROUTINE PRODUCES A NOMINAL TIME DELAY OF N*0.01 SECONDS.
THE SUBROUTINE USES THE REAL TIME CLOCK OF THE COMPUTER WHICH HAS
RESOLUTION OF ONE CENTISECOND ASSUMING THE SUBROUTINE
IS NOT SWAPPED OUT OR THAT THE INTERUPT SYSTEM IS NOT SHUT DOWN.
SINCE THE PROGRAM IS NOT SYNCHRONIZED WITH THE CLOCK THE DELAY
TIME FOR INPUT VALUE N WILL BE IN THE RANGE:
     (N-1)*0.01SEC + E   TO   N*0.01SEC + E
WHERE E IS THE EXECUTION TIME FOR THE PROGRAM.

 N - NUMBER OF CENTISECONDS OF DELAY(INPUT)
       VALUES OF 0 OR LESS PRODUCE AN IMMEDIATE RETURN
       MAXIMUM ALLOWED VALUE OF 32767 PROVIDES 5.5 MINUTES
       N IS NOT MODIFIED BY THE CALL
```

```
   SUBROUTINE DFLT(LU,IDLU,IDTRK),  REV A  11JUN82  CLF

THIS SUBROUTINE FINDS THE START TRACK OF THE LAST ENTRY IN THE
DIRECTORY WHEN A DEFAULT TO THE DIRECTORY IS PERFORMED.

SUBROUTINE ARGUMENTS:

  LU    - LIST LU(INPUT)

  IDLU  - DISC LU WHERE DIRECTORY IS LOCATED(INPUT)

  IDTRK - START TRACK OF LAST ENTRY IN DIRECTORY(OUTPUT)
          RETURNS -1 IF THERE ARE NOT ENTRIES
```

```
  SUBROUTINE DGIO(LU,ICHAN,IDIR,ICODE,IERR,IBUF,LEN),  REV C  11DEC8
$3  CLF

SUBROUTINE TO COMMUNICATE WITH ASRD AND PROVIDE ERROR REPORTING.

   ARGUMENTS ARE DEFINED AS:

     LU    - ERROR DISPLAY DEVICE(INPUT)

     ICHAN - CURRENTLY ADDRESSED CHANNEL(INPUT)

     IDIR  - DIRECTION OF TRANSACTION     1=> INPUT   (INPUT)
                                          2=> OUTPUT
                                          3=> CONTROL

     ICODE - COMMAND CODE(INPUT)

     IERR  - ERROR RETURN(OUTPUT)

     IBUF  - DATA BUFFER(USE DEPENDS ON COMMAND CODE)

     LEN   - NUMBER OF WORDS INVOLVED(INPUT)
```

```
  SUBROUTINE DRAW(LUPLT,START,STOP,OFFSET,IYMIN,IXTIC,IYTIC,
$YSF,IAMBNT,ILINE,YMULT,XMULT),  REV B  15FEB83  WSW

     THIS SUBROUTINE DRAWS THE DATA RECORD ON A STANDARD
     AXIS AS PROVIDED BY SUBROUTINE "AXIS"

     THIS SUBROUTINE USES JDATA AND GETS THE DATA LU, START TRACK,
     AND CHANNEL THROUGH COMMON.  THE SUBROUTINE ASSUMES THAT
     CHLOC HAS BEEN RUN AND THE VALUES IN PNTRS AND WRDCNT ARE VALID.

     VARIABLES USED ARE:

          LUPLT= LU OF PLOTTING DEVICE, BUT A NEGATIVE NUMBER
                 IS USED IF THE PLOTTING DEVICE IS NOT A 7245

          START= TIME(IN MILLISECONDS) TO BEGIN PLOT
          STOP = TIME TO STOP PLOT
          OFFSET= NUMBER OF MILLISECONDS THAT THE RECORD IS MOVED FWD

          IYMIN=VALUE IN ENGINEERING UNITS OF BOTTOM Y-AXIS TIC MARK
          IXTIC=NUMBER OF MILLISECONDS(OR MICROSECONDS IF YMULT=.001)
                BETWEEN TIME AXIS TIC MARKS
          IYTIC=NUMBER OF ENGINEERING UNITS BETWEEN Y-AXIS TIC MARKS

          YSF= SCALE FACTOR FOR Y AXIS (ENGR. UNITS/COUNT)

          IAMBNT=AMBIENT VALUE( ZERO LINE ) IN COUNTS

          ILINE=LINE TYPE (SEE BELOW)

          YMULT=VALUE OF .001, 1.0, OR 1000. BY WHICH ALL AMPLITUDE
                DATA ARE MULTIPLIED
          XMULT=VALUE OF .001, 1.0, OR 1000. BY WHICH ALL TIME
                DATA ARE MULTIPLIED
```

SUBROUTINE DRLST(LU,LIST,IDLU,ISTART,ISTOP), REV G 19DEC83 CLF

THIS PROGRAM ALLOWS THE ASRD DATA FILE DIRECTORY TO BE LISTED ON A DEVICE. THE DIRECTORY ENTRIES ARE AUTOMATICALLY CREATED BY THE TRANSFER(TR) COMMAND.

SUBROUTINE ARGUMENTS:

LU - INTRCTV LU(INPUT)

LIST - LIST DEVICE LU(INPUT)

IDLU - THE DISC LU WHERE THE DIRECTORY IS LOCATED(INPUT)

ISTART - THE FILE NO AT WHICH THE LISTING STARTS(INPUT)

ISTOP - THE FILE NO AT WHICH THE LISTING STOPS(INPUT)

COMMON IS USED BY THIS SUBROUTINE FOR INTERNAL BUFFERS

SUBROUTINE EDIT(LBUF,MBUF,NWRDS), REV A 4JUN82 CLF

THIS SUBROUTINE IS USED TO PERFORM LIMITED EDITING ON ASCII COMMENTS USED IN THE REMARKS, NOTES AND COMMENT SECTIONS OF THE BTST SOFTWARE. THE SUBROUTINE SUPPORTS THE FOLLOWING COMMANDS:

/ - LEAVES THE CHARACTER IN THE EXISTING TEXT UNCHANGED
$ - TRUNCATES THE EXISTING TEXT AT THE LOCATION OF THE $

SUBROUTINE ARGUMENTS:

LBUF - BUFFER CONTAINING THE EDIT COMMANDS/TEXT(INPUT)

MBUF - BUFFER CONTAINING THE OLD TEXT TO BE EDITTED(INPUT)
BUFFER CONTAINING THE EDITTED TEXT(OUTPUT)

NWRDS - THE NUMBER OF WORDS IN BUFFERS LBUF OR MBUF(INPUT)

SUBROUTINE ERMSG, REV A 1NOV83 CLF

THIS SUBROUTINE SORTS THROUGH THE ERROR LIST CONTAINED IN ARRAY KERR PASSED THROUGH COMMON. FOR EACH ERROR CODE IT FINDS WHICH CHANNELS HAVE THAT ERROR. AFTER SEARCHING FOR ALL ERRORS A CONSOLIDATED ERROR LIST IS SENT TO THE TERMINAL.

INPUT ARGUMENTS(ALL PASSED THROUGH COMMON):

KERR - ARRAY CONTAINING ERROR STATUS WORDS FOR EACH CHANNEL.

LU - LU WHERE ERROR MESSAGE SHOULD BE PRINTED.

LSTCH - LAST CHANNEL TO BE PROCESSED

SUBROUTINE EQTIM(OFSET,IPNTS,IRATE,IEQBUF,FINAL,MODE,NSPLN,
$LASTY,OLD,ZDERV,SDIAG,DIAG), REV*C 29AUG83 GZT

THIS SUBROUTINE READS A BTST DATA RECORD TAKEN AT VARIABLE SAMPLE RATE AND FILLS A BUFFER WITH EQUALLY SPACED DATA POINTS. POINTS ARE DISCARDED IF THE BUFFER RATE IS LOWER THAN THE DATA RATE AND POINTS ARE ADDED IF THE BUFFER RATE IS HIGHER THAN THE DATA RATE.

SUBROUTINE ARGUMENTS:

OFSET - OFFSET INTO THE DATA FILE(INPUT)

IPNTS - NUMBER OF POINTS IN THE EQUALLY SPACED BUFFER(INPUT)

IRATE - DESIRED RATE FOR THE EQUALLY SPACED BUFFER(INPUT)

IEQBUF - BUFFER RETURNED WITH POINTS EQUALLY SPACED IN TIME(OUTPUT)

FINAL - POSITON OF THE LAST ACTUAL DATA POINT(OUTPUT)

MODE - LINEAR OR CUBIC SPLINE INTERPOLATION - L OR C(INPUT)

NSPLN - NUMBER OF DATA POINTS TO BE USED BY SUBROUTINE ZSPLN(INPUT)
IT SHOULD BE CHOSEN AS LARGE AS POSSIBLE CONSIDERING IPNTS,WORDCOUNT MINUS OFFSET, AND AVAILABLE MEMORY. SINCE SDIAG AND DIAG CAN BE EQUIVALENCED WITH IEQBUF, THE ABSOLUTE MAXIMUM SHOULD BE IPNTS/4. IF NSPLN IS LESS THAN FOUR, THE MODE SHOULD BE CHANGED EXPLICITLY TO LINEAR SINCE THE PROCESS WOULD DEFAULT TO LINEAR.

LASTY - LAST DATA ITEM (ACTUAL OR ADDED) FROM PREVIOUS CALL(INPUT)

OLD - THE TIME ELAPSED BETWEEN LASTY AND NEXT DATA ITEM(INPUT)

ZDERV - ARRAY OF SECOND DERIVATIVES TO BE PRODUCED BY ZSPLN(INPUT)

SDIAG - ARRAY NEEDED INTERNALLY BY ZSPLN-DIMENSION NSPLN(INPUT)

DIAG - ARRAY NEEDED INTERNALLY BY ZSPLN-DIMENSION NSPLN(INPUT)

PARAMETERS INPUT THROUGH COMMON:

ICH - CHANNEL NUMBER TO BE PROCESSED

LUDK - DISC LU WHERE DATA IS LOCATED

ISTRK - START TRACK WHERE DATA IS LOCATED

LU - LIST DEVICE FOR ERROR MESSAGES

WRDCNT - NUMBER OF DATA POINTS AVAILABLE IN RECORD

SUBROUTINE GETLU(LU,LUDK,ITRK), REV A 21OCT81 CLF

THIS SUBROUTINE OBTAINS THE DATA DISC LU AND TRACK INTERACTIVELY FROM AN INPUT DEVICE. IT CHECKS THE TWO VALUES TO MAKE SURE THAT THEY ARE VALID ENTRIES. THIS SUBROUTINE IS INTENDED FOR USE ON THE BALLISTIC TST DATA FILES LOCATED ON LU'S 18,19,&20.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE LU(INPUT)

LUDK - DATA DISC LU(OUTPUT)

ITRK - DATA START TRACK(OUTPUT)

```
 SUBROUTINE GRAF2(LU,N,IX,IY,LABEL,NTOT,NC,XSCAL,YSCAL,
$L,IP,ID,IOUT,XOFF,YOFF,IYMAX,IYMIN),  REV E  21OCT82  CLF


THIS SUBROUTINE IS A MODIFICATION OF SR GRAF.  IT ELIMINATES
  SELF SCALING IN AMPLITUDE.  EXTERNAL SCALING PARAMETERS ARE
  ADDED IN THE CALL (IYMAX,IYMIN).

SUBROUTINE ARGUMENTS(ALL INPUTS):

  LU      - LOGICAL UNIT NUMBER TO PLOT ON
  N       - NUMBER OF DATA POINTS TO BE PLOTTED
            <1025 FOR 7245, <3072 FOR 2648
            (DETERMINED BY DIM OF IOUT: FOR 7245
            N=DIM(IOUT)/6  FOR 2648 N=DIM(IOUT)/2)
  IX      - X-AXIS DATA ARRAY (INTEGER) DIM=N
  IY      - Y-AXIS DATA ARRAY (INTEGER) DIM=N
  LABEL   - ALPHANUMERIC LABELS ARRAY (STRING)
            TITLE IN FIRST 50 CHARACTERS (25 WORDS)
            X-AXIS LABEL IN NEXT 30 CHARACTERS (15 WORDS)
            Y-AXIS LABEL IN NEXT 30 CHARACTERS (15 WORDS)
             NOTE: THE Y AXIS LABEL SHOULD BE REDUCED TO
                   28, 18, OR 12 CHARACTERS FOR VALUES OF
                   NTOT=2, 3, OR 4 RESP ON THE 2648.
                   VALUES OF 26 &20 SHOULD BE USED WITH
                   THE 7245 FOR NTOT=3 & 2 RESP.
  NTOT    - NUMBER OF GRAPHS PER SCREEN
  NC      - WHICH GRAPH CHANNEL TO BE PLOTTED WITH THIS CALL
  XSCAL   - REAL NUMBER REPRESENTING SCALE FACTOR
  YSCAL   - REAL NUMBER          "          "
  L       - LINE TYPE (1-11)
             1 - SOLID LINE
             2 - 2648-USER DEFINED  7245-SOLID LINE
             3 - SAME AS 2
             4 - ALTERNATE SHORT & LONG DASH WITH BIG SPACE
             5 - LONG DASH
             6 - SHORT DASH
             7 - DOTS
             8 - 2648-ALTERNATE SMALL & LONG DASH WITH SMALL
                 SPACE.  7245-ALTERNATE DASH & DOT
             9 - 2648-THREE SMALL DASHES THEN LARGE SPACE
                 7245-SOLID LINE
            10 - LARGE DASH AND TWO SMALL DASHES
            11 - POINT PLOT
  IP      - ARRAY OF TRIGGER POINT SPECIFERS
            IP(1) = NUMBER OF TRIGGER POINTS
            IP(2),... CONTAIN THE TRIGGER POINTS.  THE NUMBER
            IS LIMITED BY THE DIMENSION OF IP IN THE MAIN PRGM.
            THE TRIGGER POINT IS MARKED BY A FULL SCALE DEFLECTION
  IOUT    - OUTPUT ARRAY
  YOFF    - Y OFFSET FOR DATA
  XOFF    - X OFFSET FOR DATA
  ID      - 0 FOR 2648, 1 FOR 7245
  IYMAX   - LARGEST VALUE IN Y ARRAY (INPUT)
  IYMIN   - SMALLEST VALUE IN Y ARRAY (INPUT)
             IF IYMIN=IYMAX=0, THEN FIND MIN & MAX INTERNALLY.  OTHERWISE
             IYMIN AND IYMAX ARE USED AS PASSED INTO SUBROUTINE.

                NOTE:  VALUES OF IX, IY, XSCAL & YSCAL ARE
                       MODIFIED BY A CALL TO THIS SUBROUTINE
```

SUBROUTINE GSCTR(ICH,IDLU,ISTRK,IDTRK,ISCTR), REV B 12JUL82 CLF

THIS SUBROUTINE LOCATES THE STARTING SECTOR OF A CHANNELS DOCUMENTATION SECTION IN A BTST DATA FILE.

SUBROUTINE ARGUMENTS:

ICH - CHANNEL NO(INPUT)

IDLU - DISC LU WHERE DATA FILE IS LOCATED(INPUT)

ISTRK - START TRACK OF DATA FILE(INPUT)

IDTRK - START TRACK OF CHANNEL DOCUMENTATION SECTION(OUTPUT)

ISCTR - START SECTOR OF DOCUMENTATION SECTION(OUTPUT)
RETURN -1 IF ICH WAS NOT SELECTED

SUBROUTINE GTLTC(LU,LUDK,ITRK,ICH), REV A 01JUL83 GZT

THIS SUBROUTINE OBTAINS THE DATA DISK LU, TRACK, AND CHANNEL INTERACTIVELY FROM AN INPUT DEVICE. IT CHECKS THE VALUES TO MAKE SURE THEY ARE VALID ENTRIES. THIS SUBROUTINE IS INTENDED FOR USE ON THE BALLISTIC TST DATA FILES ON LU'S 18,19,&20.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE LU(INPUT)

LUDK - DATA DISK LU(OUTPUT)

ITRK - DATA START TRACK(OUTPUT)

ICH - CHANNEL NUMBER(OUTPUT)

SUBROUTINE GTSCL(XMIN,XMAX,NG,XMING,XMAXG,DELX), REV B 29OCT82 $CLF

THIS SUBROUTINE IS USED TO GENERATE THE MAXIMUM & MINIMUM VALUES USED ON A PLOT AXIS.

SUBROUTINE ARGUMENTS:

XMIN - MINIMUM VALUE OF DATA ARRAY IN ENGINEERING UNITS(INPUT)

XMAX - MAXIMUM VALUE OF DATA ARRAY IN ENGINEERING UNITS(INPUT)

NG - NUMBER OF TIC MARKS ON AXIS(INPUT)

XMING - MINIMUM VALUE GENERATED FOR AXIS(OUTPUT)

XMAXG - MAXIMUM VALUE GENERATED FOR AXIS(OUTPUT)

DELX - DIFFERENCE BETWEEN TIC MARKS IN ENGINEERING UNITS(OUTPUT)

```
   SUBROUTINE HEADR(LU,IPLU,IPTRK,IFUNC),  REV I  19DEC83  CLF

THIS SUBROUTINE ALLOWS THE SCRATCH HEADER TO BE READ AND MODIFIED,
ALLOWS A DATA RECORD HEADER TO BE READ, OR ALLOWS A DATA RECORD
HEADER TO BE READ AND MODIFIED.

SUBROUTINE ARGUMENTS:

   LU     - INTERACTIVE LU - IFUNC = 1 OR 3  (INPUT)
            LIST LU        - IFUNC = 2

   IPLU   - DISC LU WHERE HEADER IS LOCATED(INPUT)

   IPTRK  - DISC TRACK(INPUT)
              FOR IFUNC = 1 MUST = 1
              FOR IFUNC = 2 OR 3:  IF -1 DEFAULT TO DIRECTORY
                                   IF >0 GO TO SPECIFIED TRACK

   IFUNC  - FUNCTION TO PERFORM(INPUT)
              =1: DISPLAY & MODIFICATION OF A "SCRATCH" HEADER LOCATED
                  ON TRACK 1, SECTOR 6 OF THE CURRENT DATA DISC.  THE
                  "SCRATCH" HEADER IS AUTOMATICALLY STORED WITH THE DATA
                  WHEN A TRANSFER(TR) IS PERFORMED. [ADCHK HE COMMAND]
              =2: DISPLAY OF THE HEADER ON A STORED DATA FILE(LOCATED ON
                  THE DATA FILES FIRST TRACK, SECTOR 4).  ALLOWS LISTING
                  TO ANY LU.  [ADCHK RH COMMAND]
              =3: DISPLAY AND MODIFICATION OF THE HEADER ON A STORED
                  DATA FILE.  [ADCHK MH COMMAND]

COMMON IS USED BY THIS SUBROUTINE FOR INTERNAL BUFFERS.
```

```
   SUBROUTINE HMRKR(LU,ID,IP,IXT,IOFX,IOFY,RMIY,YSC),  REV B  29OCT82
  *  CLF

THIS SUBROUTINE PLOTS HORIZONTAL MARKERS ON PLOTS.  IT
HANDLES BOTH THE 7245 AND 2648.

SUBROUTINE ARGUMENTS(ALL ARE INPUT):

  LU     - PLOT LU

  ID     - DEVICE ID CODE: 0=2648   1=7245

  IP     - ARRAY CONTAINING MARKER POINTS.  IP(1)=NO OF MARKERS

  IXT    - MAXIMUM X VALUE ALLOWED ON PLOT (PLOTTER UNITS)

  IOFX   - START OF X AXIS (PLOTTER UNITS)

  IOFY   - START OF Y AXIS (PLOTTER UNITS)

  RMIY   - MINIMUM Y VALUE (PLOTTER UNITS)

  YSC    - Y SCALE FACTOR

  !!!!! WARNING !!!!!   DOES NOT WORK FOR MULTIPLE PLOTS
```

SUBROUTINE INSPT, REV*I 03FEB83 CLF

THIS SUBROUTINE ALLOWS ASRD DATA STORED ON DISC TO BE INSPECTED. RAW DATA IS FORMATTED IN BIT IMAGE AND HUMAN UNITS TO ALLOW EASY INTERPRETATION BY THE OPERATOR.

SUBROUTINE ARGUMENTS PASSED THROUGH COMMON:

LU - INTERACTIVE LU(INPUT)

LIST - LIST DEVICE LU(INPUT)

IDLU - DISC LU WHERE DATA IS LOCATED(INPUT)

IDTRK - START TRACK OF DATA RECORD ON DISC(INPUT)

LSTCH - LAST CHANNEL IN SYSTEM(INPUT)

SUBROUTINE IYT2(IY,IT,NPTS,OFST,AVAIL,TSF,INDX), REV*G 02JUN83 *GZT

SUBROUTINE TO GENERATE INTEGER Y AND T ARRAYS FOR PLOTTING.

SUBROUTINE ARGUMENTS:

IY - INTEGER Y ARRAY(OUTPUT)

IT - INTEGER TIME ARRAY(OUTPUT)

NPTS - NUMBER OF POINTS DESIRED IN IY AND IT ARRAYS(INPUT)
ACTUAL NUMBER OF POINTS IN IY AND IT ARRAYS(OUTPUT)

OFST - OFFSET INTO DATA ARRAY TO START SEARCH(INPUT)
NUMBER OF DATA POINTS READ FROM ARRAY(OUTPUT)

AVAIL - NUMBER OF POINTS ASSOCIATED WITH THE CHANNEL(INPUT)

TSF - FACTOR BY WHICH IT POINTS MUST BE MULTIPLIED(OUTPUT) TO GET TIME IN MILLISECONDS

INDX - PARAMETER TO SKIP POINTS FOR PLOTTING(OUTPUT)
AS ONLY 600 POINTS OF RESOLUTION ARE AVAILABLE, TO PLOT 16400 POINTS ON GRAPHICS TERMINAL, 'INDX' SHOULD = 28)

FUNCTION JBCD(IWORD), REV A 15SEP82 CLF

THIS FUNCTION TAKES A FOUR DIGIT BINARY CODED DECIMAL(BCD) INPUT AND RETURNS THE INTEGER VALUE OF THE FOUR BCD DIGITS.

INPUT ARGUMENT: IWORD - 16 BIT INTEGER WORD CONTAINING FOUR BCD DIGITS. IWORD IS NOT CHANGED. THE BCD FORMAT IS:

BITS 15 0

BIT WEIGHT

8 4 2 1
0 0 0 0 8 4 2 1
0 0 0 0 0 0 0 0 8 4 2 1
0 0 0 0 0 0 0 0 0 0 0 0 8 4 2 1

```
   FUNCTION JDATA(SMPL),  REV A  21APR82  CLF

THIS FUNCTION RETURNS A DATA WORD READ FROM A BTST DATA FILE
FOR A GIVEN CHANNEL.  THE FUNCTION USES A 6144 WORD BUFFER IN
COMMON TO HOLD A PORTION OF THE DATA FILE.  IF THE SAMPLE
REQUESTED IS IN THE BUFFER A DISC READ IS NOT MADE.  IF THE CHANNEL
NUMBER, DISC LU OR TRACK CHANGE, OR THE REQUESTED SAMPLE IS NOT IN
THE BUFFER, THEN A DISC READ IS MADE TO FILL THE BUFFER WITH ANOTHER
TRACK OF DATA.

INPUT ARGUMENTS:

  SMPL   - THE INDEX OF THE SAMPLE VALUE TO BE RETURNED
           IF LESS THAN 0 OR GREATER THAN NUMBER OF WORDS IN CHANNEL
           WILL RETURN AN ERROR MESSAGE AND STOP.
           IF 0 WILL RESET THE FUNCTION SO THAT A SUBSEQUENT CALL
           WILL FORCE A DISC READ.  THIS ALLOWS THE JBUF COMMON
           BUFFER TO BE USED WITH OTHER SUBROUTINES IF NEEDED.

INPUT ARGUMENTS PASSED THROUGH COMMON:

  JBUF   - BUFFER USED TO HOLD DATA
  ICH    - THE CHANNEL FROM WHICH THE DATA IS TO BE READ
  LUDK   - THE DISC LU WHERE THE DATA IS LOCATED
  ISTRK  - THE DISC TRACK WHERE THE DATA STARTS
  LU     - LOGICAL UNIT FOR RECEIVING ERROR MESSAGES
  PNTRS  - THE CHANNEL ADDRESS POINTERS
  WRDCNT - THE NUMBER OF WORDS IN THE CHANNEL

   PNTRS AND WRDCNT ARE OBTAINED BY CALLING SUBROUTINE CHLOC PRIOR TO
   USING THIS FUNCTION.
```

```
   SUBROUTINE LABL(RMIN,RMAX,IDIR,IOFX,IOFY,LU,ITIC),  REV B  28OCT82
  &  CLF

THIS SUBROUTINE IS CALLED BY GRAF TO WRITE THE NUMBERS USED TO LABEL
THE X & Y AXIS OF A PLOT ON THE 2648 GRAPHICS TERMINAL.

SUBROUTINE ARGUMENTS(ALL ARE INPUTS):

  RMIN   - MINIMUM VALUE USED ON AXIS IN ENGINEERING UNITS(REAL)

  RMAX   - MAXIMUM VALUE USED ON AXIS IN ENGINEERING UNITS(REAL)

  IDIR   - DIRECTION OF AXIS TO BE LABELED X=6,Y=7

  IOFX   - OFFSET IN X DIRECTION FOR FIRST LABEL

  IOFY   - OFFSET IN Y DIRECTION FOR FIRST LABEL

  LU     - GRAPHICS TERMINAL LU

  ITIC   - ARRAY CONTAINING LOCATIONS OF TIC MARKS
```

```
   SUBROUTINE LOOKS(LABEL,ISTART,ISTOP),  REV B  29OCT82  CLF

  THIS SUBROUTINE LOOKS THROUGH THE ARRAY LABEL
  FOR SPACES, FROM ISTOP (THE LAST WORD OF SEARCH STRING)
  TILL ISTART (THE FIRST WORD IN THE SEARCH STRING).
  UPON FINDING A NON-SPACE CHARCTER, IT RETURNS ISTART WITH
  THE NUMBER OF WORDS IN THE SPECIFIED STRING WITH NON-TRAILING
  SPACES TO THE RIGHT
```

SUBROUTINE LSQFT(NPTS,X,Y,SLOPE,OFFSET,ERR), REV A 7JAN81 CLF

SUBROUTINE TO PERFORM A FIRST ORDER LINEAR LEAST SQUARES FIT TO A SET OF DATA POINTS.

SUBROUTINE ARGUMENTS:

NPTS	- NUMBER OF DATA PAIRS IN X AND Y(INPUT)
X	- ARRAY OF X DATA POINTS(INPUT)
Y	- ARRAY OF Y DATA POINTS(INPUT)
SLOPE	- SLOPE CALCULATED BY THE LEAST SQUARES FIT(OUTPUT)
OFST	- OFFSET CALCULATED BY THE LEAST SQUARES FIT(OUTPUT)
ERR	- ARRAY CONTAINING THE DEVIATIONS OF THE POINTS FROM THE FIT(OUTPUT)

SUBROUTINE MCVRT(ID), REV A 3MAR80 CLF

THIS SUBROUTINE TAKES A TWO DIGIT INTEGER INPUT THROUGH ID AND CONVERTS IT INTO THE CORRESPONDING ASCII DIGITS FOR OUTPUT THROUGH ID.

SUBROUTINE MLBL, REV C 26FEB83 CLF

THIS SUBROUTINE IS USED TO MODIFY THE TYPE DATA, PLOT LABELS OR REMARKS ENTRIES SAVED IN THE DOCUMENTATION OF A BTST DATA FILE.

SUBROUTINE ARGUMENTS INPUT THROUGH COMMON:

LU	- INTERACTIVE LU
IDLU	- DISC LU WHERE DATA IS LOCATED
IDTRK	- START TRACK OF DATA RECORD ON DISC
ICH	- CHANNEL TO MODIFY

SUBROUTINE NOCHL(LSTCH), REV A 16JUL82 CLF

THIS SUBROUTINE RETURNS THE NUMBER OF THE LAST CHANNEL IN A BTST.

SUBROUTINE ARGUMENT:

LSTCH	- LAST CHANNEL IN SYSTEM(OUTPUT)

SUBROUTINE NOTES(MODE), REV B 16JUL82 CLF

THIS SUBROUTINE IS CALLED BY ADCHK TO PROVIDE DOCUMENTATION NOTES & COMMENTS. IT PERFORMS DIFFERENT FUNCTIONS DEPENDENT ON THE INPUT MODE:

1. READS NOTES FROM A DATA FILE AND LISTS THEM ON AN OUTPUT DEVICE.
2. ALLOWS NOTES TO BE ENTERED OR MODIFIED ON A DATA FILE.
3. READS COMMENTS FROM A CHANNELS DOCUMENTATION FILE AND LISTS THEM ON AN OUTPUT DEVICE.
4. ALLOWS COMMENTS TO BE ENTERED OR MODIFIED TO A CHANNELS DOCUMENTATION FILE.

SUBROUTINE ARGUMENT:

MODE - MODE OF OPERATION(INPUT)

PARAMETERS INPUT THROUGH COMMON:

LU - INTERACTIVE TERMINAL

LIST - LIST DEVICE LU

IDLU - DISC LU WHERE DATA IS LOCATED

IDTRK - START TRACK OF DATA RECORD

SUBROUTINE OTDMA(N), REV C 1JUN82 CLF

THIS SUBROUTINE IS USED TO WRITE DATA TO DISC FROM A 6144 WORD BUFFER. THE BUFFER IS WRITTEN TO DISC EITHER WHEN IT IS FULL OR IF ONLY PARTIALLY FILLED IT WILL PAD THE BUFFER WITH ZEROS TO FILL IT.

N - N=0 INITIALIZES SUBROUTINE(DONE BEFORE FIRST DATA WORD IS PUT IN BUFFER)
N>0 INCREMENTS INDEX COUNTER BY N. IF BUFFER IS FULL WRITES IT TO DISC
N<0 TERMINATES SUBROUTINE(WILL TAKE CURRENT CONTENTS OF BUFFER UP TO INDEX-1 THEN FILL REMAINDER OF BUFFER WITH ZEROES AND WRITE TO DISC)

THE FOLLOWING PARAMETERS ARE PASSED TO THE SUBROUTINE IN COMMON:

IDLU - DATA DISC LU
IDTRK - DATA DISC TRACK
MAXDMA - BUFFER SIZE
MBUF - DATA BUFFER

SUBROUTINE PARAM(LU,ICHAN,IDLU,LSTCH), REV*H 16DEC83 CLF

THIS SUBROUTINE ALLOWS ADCHK PARAMETERS TO BE DISPLAYED AND CHANGED.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE OPERATOR TERMINAL[2648](INPUT)

ICHAN - CHANNEL TO BE DISPLAYED AND CHANGED(INPUT)

IDLU - DISC LU WHERE PARAMETERS ARE LOCATED(INPUT)

LSTCH - LAST CHANNEL ON SYSTEM(INPUT)

SUBROUTINE PRHDR(ICH,LU,IHDRSW), REV A 24JAN83 CLF

THIS SUBROUTINE PRINTS A CHANNEL HEADER FOR LABELING DIAGNOSTIC OR ANALYSIS PROGRAM OUTPUT FOR CASES WHERE NO HEADER IS DESIRED UNTIL AN ERROR IS FOUND. ONCE PRINTED THE HEADER WILL NOT BE REPEATED AGAIN.

INPUT ARGUMENTS:

ICH - CHANNEL NUMBER TO USE IN HEADER

LU - LIST LU WHERE HEADER IS TO BE PRINTED

IHDRSW - SWITCH TO DETERMINE IF HEADER IS TO BE PRINTED OR NOT
IHDRSW = 0 WILL CAUSE HEADER TO BE PRINTED. A VALUE OF 1 WILL BE RETURNED BY THE SUBROUTINE
IHDRSW # 0 WILL RESULT IN NO HEADER BEING PRINTED. THE VALUE OF IHDRSW IS NOT CHANGED.

SUBROUTINE PRMDS(LU,LUDK,ITRK,MODE,LSTCH), REV*E 10NOV83 CLF

THIS SUBROUTINE PRODUCES A DISPLAY OF 18 OF THE ASRD SETUP PARAMETERS FOR ALL CHANNELS. IF THE CHANNEL IS NOT SELECTED, NONE OF THE PARAMETERS ARE DISPLAYED. IF EXTERNAL TRIGGER IS SELECTED, THRESHOLD AND SLOPE ARE NOT DISPLAYED. IF STRAIN MODE IS SELECTED, BALANCE IS DISPLAYED. IF THE FREQUENCY IS NEGATIVE, THE RATE TABLE/CODE IS DISPLAYED.

SUBROUTINE ARGUMENTS

LU - LIST LU FOR PARAMETER DISPLAY(INPUT)

LUDK - DISC LU WHERE PARAMETERS ARE TO BE READ(INPUT)

ITRK - DISC TRACK WHERE PARAMETERS ARE TO BE READ(INPUT)

MODE - OPERATIONAL MODE(INPUT)
MODE=1:PARAMETERS FROM TRACK 0 OF DATA LU(PD COMMAND)
MODE=2:PARAMETERS FROM DATA FILE(RA COMMAND)

LSTCH - LAST CHANNEL IN SYSTEM(INPUT)

SUBROUTINE PRMFR(IPARAM,ITHRSH), REV A 09DEC83 CLF

THIS SUBROUTINE FORMATS ADCHK PARAMETERS FOR DISPLAY.

SUBROUTINE ARGUMENTS:

IPARAM - ARRAY CONTAINING PARAMETERS IN DISC IMAGE FORMAT(INPUT)
RETURNS WITH FORMATTED PARAMETERS(OUTPUT)

ITHRSH - ARRAY CONTAINING FORMATTED THRESHOLD(OUTPUT)

SUBROUTINE PRMLS(LU,IPARAM), REV A 08DEC83 CLF

THIS SUBROUTINE LISTS ADCHK PARAMETERS FROM A BUFFER TO A LIST DEVICE.

SUBROUTINE ARGUMENTS:

LU - LIST DEVICE(INPUT)

IPARAM - PARAMETER BUFFER(INPUT)

SUBROUTINE PRMOT(LU,ICH,IPARAM,IERR), REV*D 9MAY83 CLF

THIS SUBROUTINE LOADS THE ASRD WITH A SET OF PARAMETERS INPUT VIA A BUFFER. IT PERFORMS NO INTERNAL CHECKS ON THE PARAMETERS.

SUBROUTINES DGIO AND IASRD ARE USED TO HANDLE THE INTERFACE WITH THE DIGITIZER.

IT IS ASSUMED THAT ALL CHANNELS HAVE BEEN CLEARED AND THE DESIRED CHANNEL HAS BEEN ADDRESSED PRIOR TO ENTRY INTO THIS SUBROUTINE.

SUBROUTINE ARGUMENTS:

LU - LIST DEVICE FOR ERROR MESSAGES(INPUT)

ICH - CHANNEL TO BE PROGRAMMED(INPUT)

IPARAM - ARRAY CONTAINING THE SETUP PARAMETERS(INPUT)
(1) - NOT USED
(2) - MASTER 1=YES 0=NO
(3) - SLAVE 1=YES 0=NO
(4) - XTRIGR 1=YES 0=NO
(5) - THRESHOLD 10*DESIRED VALUE IN VOLTS
(6) - SLOPE (" +" " -" " A" "R+" "R-")
(7) - RUN DELAY IN MILLISECONDS
(8) - STOP DELAY IN MILLISECONDS
(9) - PRETRIGGER SIZE IN K
(10) - NOT USED
(11) - MEMORY SIZE IN K
(12) - SCALAR
(13) - LENGTH
(14) - DECREASE DELAY
(15) - NUMBER OF ENTRIES IN RATE TABLE
(16) - FIRST RATE
(17) - FIRST START ADDRESS
.
.
(30) - EIGTH RATE
(31) - EIGTH START ADDRESS

IERR - ERROR CODE(OUTPUT)
ZERO IF NO ERROR

SUBROUTINE PURGE(LU,IPLU), REV F 19DEC83 CLF

THIS SUBROUTINE IS TO CLEAR THE DIRECTORY. IT SETS THE FIRST WORD OF THE DIRECTORY(START TRK OF LAST FILE) TO 2 AND THE SECOND WORD(NUMBER OF DATA FILES) TO 0 ON THE CURRENT DEFAULT LU. NONE OF THE OTHER DIRECTORY ENTRIES ARE ACTUALLY WRITTEN OVER.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE LU(INPUT)

IPLU - DISC LU WHERE DIRECTORY IS LOCATED(INPUT)

FUNCTION RAMBT(ISTART,ISTOP), REV*B 10AUG83 GZT

THIS FUNCTION RETURNS THE AMBIENT(THE TIME WEIGHTED AVERAGE) FOR A SET OF BTST DATA POINTS.

FUNCTION ARGUMENTS:

ISTART - THE DATA POINT LOCATION TO START AVERAGING(INPUT)
USE 2 TO COMPUTE THE STANDARD AMBIENT

ISTOP - THE DATA POINT LOCATION TO STOP AVERAGING(INPUT)
USE 21 TO COMPUTE THE STANDARD AMBIENT.

INPUT PARAMETERS PASSED THROUGH COMMON:

ICH - CHANNEL NUMBER TO PROCESS

LUDK - DISC LU WHERE DATA IS LOCATED

ISTRK - START TRACK WHERE DATA IS LOCATED

WRDCNT - NO OF DATA POINTS IN THIS CHANNEL

SUBROUTINE RATBL, REV*H 14DEC83 CLF

THIS SUBROUTINE ALLOWS BTST RATE TABLES TO BE DISPLAYED AND MODIFIED.

PARAMETERS INPUT THROUGH COMMON:

LU - INTERACTIVE LU(2648)

NTAB - TABLE NUMBER

IDLU - DISC LU WHERE TABLES ARE STORED

MODE - MODE OF OPERATION
MODE = 0 DISPLAY OR MODIFY TABLE ON DISC
MODE = -1 DISPLAY TABLE READ BY VE COMMAND

SUBROUTINE RBASE(MBUF,IDLU,IRATE,ITABLE), REV*A 6MAY83 CLF

THIS SUBROUTINE FINDS THE BASE RATE USED TO ADJUST THE ASRD RATES STORED IN A DATA FILE.

ARGUMENTS:

MBUF - INPUT: CONTAINS THE PARAMETERS USED TO SETUP A CHANNEL
OUTPUT: IF FIXED RATE, UNCHANGED.
IF VARIABLE RATE, CONTAINS RATE TABLE.

IDLU - INPUT: DISC LU WHERE RATE TABLE IS LOCATED.

IRATE - OUTPUT: BASE RATE USED TO ADJUST RATE CODES.

ITABLE - OUTPUT: -1 = FIXED RATE, >=0 = VARIABLE RATE TABLE

SUBROUTINE RDMAR(LU,ICH,MODE,STADRS,WRDCNT,IERR), REV A 25MAR83 $ CLF

THIS SUBROUTINE PERFORMS TWO FUNCTIONS:
IT WILL READ THE START ADDRESS AND WORD COUNT OF AN ASRD CHANNEL
OR
IT WILL WRITE A NEW START ADDRESS AND WORD COUNT TO AN ASRD CHANNEL

NOTE: THIS SUBROUTINE ASSUMES THAT A VALID CHANNEL ADDRESS COMMAND (CALL ATTNN) HAS ALREADY BEEN PERFORMED.

SUBROUTINE ARGUMENTS:

LU - ERROR LIST DEVICE LU(INPUT)

ICH - ASRD CHANNEL TO BE READ(INPUT)

MODE - MODE OF OPERATION(INPUT)
MODE = 0 READ START ADDRESS AND WORD COUNT
MODE = 1 WRITE START ADDRESS AND WORD COUNT

STADRS - START ADDRESS OF ASRD MEMORY ADDRESS COUNTER
OUTPUT IF MODE = 0 -- INPUT IF MODE = 1

WRDCNT - REAL COUNT OF NUMBER OF WORDS TAKEN BY CHANNEL
OUTPUT IF MODE = 0 -- INPUT IF MODE = 1

IERR - ERROR CODE(OUTPUT)
ZERO IF NO ERRORS

SUBROUTINE RDVM(LUDVM,VALUE), REV A 160CT81 CLF

THIS SUBROUTINE TRIGGERS THE H-P 3455A DIGITAL VOLTMETER TO TAKE A MEASUREMENT AND THEN READS THE VALUE. THE DVM MUST HAVE BEEN PREVIOUSLY SET UP TO TAKE THE READING.

SUBROUTINE ARGUMENTS:

LUDVM - THE LU OF THE DVM(INPUT)

VALUE - THE VALUE READ FROM THE DVM(OUTPUT)

SUBROUTINE REVNT, REV*F 1JUN83 CLF

THIS SUBROUTINE IS USED TO DISPLAY THE EVENT CHANNELS FROM A DATA FILE STORED ON DISC. THE EVENTS ARE READ THEN PROCESSED BY SUBROUTINES BITS AND TMCVT BEFORE BEING OUTPUT.

PARAMETERS INPUT THROUGH COMMON:

LU - INTERACTIVE TERMINAL

LIST - LIST DEVICE

IDLU - DISC LU WHERE DATA IS STORED

IDTRK - START TRACK OF DATA RECORD TO BE PROCESSED

SUBROUTINE RPARM(LU,IDLU,LSTCH), REV*D 12NOV83 CLF

THIS SUBROUTINE RESTORES THE PARAMETERS SAVED IN AN FMGR FILE TO A DATA LU.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE OPERATOR LU(INPUT)

IDLU - DATA DISC LU WHERE PARAMETERS ARE STORED(INPUT)

LSTCH - LAST CHANNEL IN SYSTEM(INPUT)

SUBROUTINE RPTER(LU,IERR1,IERR2), REV B 31MAY83 CLF

THIS SUBROUTINE DECODES THE ERROR STATUS WORDS RETURNED BY THE BTST DIGITIZER AND PROVIDES AN ERROR MESSAGE FOR EACH ERROR BIT SET.

SUBROUTINE ARGUMENTS:

LU - LIST LU WHERE ERROR MESSAGE IS TO BE PRINTED(INPUT)

IERR1 - ERROR STATUS REGISTER WORD 1(INPUT)

IERR2 - ERROR STATUS REGISTER WORD 2(INPUT)

```
      SUBROUTINE RTMNB(IOFST,ICOUNT,RMS,YMAX,YMIN,SF,OF,
     $  TSUM,AVG,IBUF,IBRATE),  REV A  12DEC83  CLF
```

SUBROUTINE TO CALCULATE THE RMS, AVG, MAX, AND MIN OF AN ARRAY CONTAINING DATA FROM THE APG BALLISTIC DATA ACQUISITION SYSTEM.

ARGUMENTS ARE DEFINED AS FOLLOWS:

IOFST - OFFSET INTO THE DATA FILE(INPUT)

ICOUNT - NUMBER OF POINTS INCLUDED IN RMS CALCULATION(INPUT)

NOTE THAT ONLY AN INTEGER NUMBER OF POINTS CAN BE HANDLED BY THIS ROUTINE AS COMAPARED TO REAL VALUES USED IN RTMNS.

RMS - ON INPUT-RMS .GE. 0 CALCULATES AC RMS
RMS .LT. 0 CALCULATES DC RMS
ON RETURN-CONTAINS CALCULATED RMS VALUE

YMAX - MAX VALUE IN ARRAY-PHYSICAL UNITS(OUTPUT)

YMIN - MIN VALUE IN ARRAY-PHYSICAL UNITS(OUTPUT)

SF - SCALE FACTOR ASSOCIATED WITH DATA ARRAY(INPUT)

OF - OFFSET ASSOCIATED WITH DATA ARRAY(INPUT)

TSUM - TIME INTERVAL OVER WHICH CALCULATIONS ARE PERFORMED(OUTPUT)

AVG - AVERAGE VALUE OF DATA POINTS(OUTPUT)

IBUF - ARRAY CONTAINING DATA POINTS(INPUT)

IBRATE - BASE RATE FOR RECONSTRUCTING CORRECT SAMPLE RATE(INPUT)

```
      SUBROUTINE RTMNS(OFST,COUNT,RMS,YMAX,YMIN,SF,OF,
     $  TSUM,AVG),  REV*I  10MAR83  CLF
```

SUBROUTINE TO CALCULATE THE RMS, AVG, MAX, AND MIN OF A SET OF BTST DATA POINTS STORED ON DISC.

SUBROUTINE ARGUMENTS:

OFST - OFFSET INTO THE DATA FILE(INPUT)

COUNT - NUMBER OF POINTS INCLUDED IN RMS CALCULATION(INPUT)

RMS - ON INPUT-RMS .GE. 0 CALCULATES AC RMS
RMS .LT. 0 CALCULATES DC RMS
ON RETURN-CONTAINS CALCULATED RMS VALUE

YMAX - MAX VALUE IN ARRAY-PHYSICAL UNITS(OUTPUT)

YMIN - MIN VALUE IN ARRAY-PHYSICAL UNITS(OUTPUT)

SF - SCALE FACTOR ASSOCIATED WITH DATA ARRAY(INPUT)

OF - OFFSET ASSOCIATED WITH DATA ARRAY(INPUT)

TSUM - TIME INTERVAL OVER WHICH CALCULATIONS ARE PERFORMED(OUTPUT)

AVG - AVERAGE VALUE OF DATA POINTS(OUTPUT)

```
  SUBROUTINE SCALE(LU,LUPLT,START,STOP,OFFSET,TLOFST,YMAX,YMIN,
 $IYMIN,IXTIC,IYTIC,YMULT,XMULT) ,  REV A  15FEB83  WSW

THIS SUBROUTINE FINDS THE APPROPRIATE SCALE SIZE
FOR USE BY DPLOT.  IT ALSO INITIALIZES THE PLOTTER.
TIME SCALING IS ESTABLISHED USING THE START TIME,STOP
TIME AND TIME LABEL OFFSET VALUES SUPLIED BY THE MAIN
PROGRAM. AMPLITUDE SCALING IS ACCOMPLISHED USING
THE YMAX AND YMIN VALUES SUPPLIED BY THE MAIN PROGRAM.

THE SUBROUTINE CALCULATES XTIC,YTIC, & YMIN AND
PROVIDES THES VALUES BACK TO THE MAIN PROGRAM.
TIC INTERVALS ARE CHOSEN TO BE INTERVALS OF
1,2,4,5,10,20,40 **** ETC.

IF THE TIME OR AMPLITUDE VALUES HAVE TO BE SCALED
BY A FACTOR OF 1000 TO OBTAIN REASONABLE TIC LABELS
A MULTIPLIER OF .001 OR 1000 CAN BE USED.  THE VALUES OF
YMULT AND XMULT PROVIDE THIS INFORMATION TO THE
MAIN PROGRAM.

    VARIABLES USED ARE:

        LU=INTERACTIVE LU
        LUPLT= LU OF PLOTTING DEVICE, EXECEPT THAT A NEGATIVE
               NUMBER IS USED IF THE PLOTTER IS NOT A 7245

        START= TIME IN MILLISECONDS TO BEGIN PLOT
        STOP = TIME IN MILLISECONDS TO STOP PLOT
        OFFSET=TIME IN MILLISECONDS TO MOVE RECORD FORWARD
        TLOFST=NUMBER OF MILLISECONDS TO SUBTRACT
               FROM TIME AXIS LABELS

        YMAX=MAXIMUM VALUE(IN ENGINEERING UNITS) IN RECORD
        YMIN=MAXIMUM VALUE(IN ENGINEERING UNITS) IN RECORD
        IYMIN=VALUE(IN ENGR. UNITS) OF BOTTOM Y TIC MARK
        IXTIC=NUMBER OF MILLISECONDS(OR MICROSECONDS IF XMULT=.001)
              BETWEEN TIC MARKS ON TIME AXIS
        IYTIC=NUMBER OF ENGINEERING UNITS BETWEEN Y-AXIS TIC
              MARKS MULTIPLIED BY YMULT

        YMULT=VALUE OF .001, 1.0,OR 1000. BY WHICH ALL AMPLITUDE
              DATA ARE MULTIPLIED
        XMULT=VALUE OF .001, 1.0, OR 1000. BY WHICH ALL TIME
              DATA ARE MULTIPLIED

        9 TIC MARKS 1" APART ON X AXIS

1000,6796                                          10144,6796
   |----|----|----|----|----|----|----|----|----|
   |                                             |
   =          1" = 1016 PLATTENS                 =
   |                                             |
   =                                             =
   |                                             |
   =                                             =
   |                                             |
   =  6 TIC MARKS 1"                             =
   |  APART ON Y AXIS                            |
   =                                             =
   |                                             |
   =                                             =
   |                                             |
   |----|----|----|----|----|----|----|----|----|
1000,700                                           10144,700
```

SUBROUTINE SCHDL(LU,ITYPE,NAME,IP1,IP2,IP3,IP4,IP5,IBUFR,IBUFL),
$REV A 3NOV83 CLF

THIS SUBROUTINE ALLOWS A PROGRAM TO SCHEDULE ANOTHER PROGRAM WITH THE NO ABORT BIT SET TO PREVENT THE SCHEDULING PROGRAM FROM BEING ABORTED IF A SCHEDULING PROBLEM IS ENCOUNTERED.

SUBROUTINE ARGUMENTS(ALL ARE INPUTS):

LU - LU FOR ERROR MESSAGES

ITYPE - TYPE OF SCHEDULE TO BE DONE(8,9,10,23,24)

IP1 to IP5 - PARAMETERS TO PASS TO SCHEDULED PROGRAM

IBUFR - BUFFER TO PASS TO SCHEDULED PROGRAM

IBUFL - LENGTH OF IBUFR

SUBROUTINE SELCH(LU,LUDK,LSTCH), REV B 10NOV83 CLF

THIS PROGRAM SELECTS AND DESELECTS CHANNELS IN THE PARAMETER FILE FOR THE BTST.

LU - INTERACTIVE LU(INPUT)

LUDK - DISC LU WHERE PARAMETERS ARE LOCATED(INPUT)

LSTCH - LAST CHANNEL IN SYSTEM(INPUT)

SUBROUTINE SETUP, REV*H 1NOV83 CLF

THIS SUBROUTINE READS THE BTST SETUP PARAMETERS FROM DISC, INTERPRETS THE ENTRIES, AND PROGRAMS THE ASRD, THE SIGNAL CONDITIONER AND THE FILTER AMPLIFIER FOR DATA ACQUISITION.

INPUT PARAMETERS PASSED THROUGH COMMON:

ISKIP - SKIP 316 & 416 PROGRAMMING IF NOT ZERO

LU - LIST DEVICE LU FOR MESSAGES

IDLU - DISC LU WHERE PARAMETERS ARE LOCATED

LSTCH - LAST CHANNEL IN SYSTEM

SUBROUTINE SPARM(LU,IDLU,NSELCH,ISELCT,LSTCH), REV C 12NOV83 CL
$F

THIS PROGRAM SAVES THE CURRENTLY SELECTED PARAMETERS FROM A BTST DATA LU IN A USER NAMED FMGR FILE. THE FILE MUST BEGIN WITH A '*'. THE FILE CREATED IS A TYPE 1. THE LENGTH OF THE FILE IS (1 + # OF CHNLS SAVED) BLOCKS. THE FILE IS A DIRECT IMAGE OF THE PARAMETERS AS STORED ON DISK. THE FIRST RECORD OF THE FILE CONTAINS THE TOTAL NUMBER OF CHANNELS SAVED IN THE FILE, FOLLOWED BY THE CHANNEL NUMBERS WHICH WERE SAVED.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE LU (INPUT)

IDLU - DISC DATA LU (INPUT)

NSELCH - NUMBER OF SELECTED CHANNELS (INPUT)

ISELCT - ARRAY CONTAINING THE LIST OF SELECTED CHANNELS (INPUT)

LSTCH - LAST CHANNEL IN SYSTEM(INPUT)

SUBROUTINE TCHNG(IOFST,ICOUNT,ITRNS,TIME,IBUF,IBRATE), REV A 14D
$EC83 CLF

SUBROUTINE TO DETERMINE AT WHAT TIME BTST DATA PASSES THROUGH A GIVEN VALUE. THE TRANSITION MAY BE EITHER POSITIVE OR NEGATIVE GOING. UP TO 85 TRANSITION TIMES ARE RETURNED - AT THAT POINT PROCESSING IS STOPPED.

SUBROUTINE ARGUMENTS:

IOFST - (INPUT)OFFSET INTO THE DATA ARRAY
(RETURN) -1 IF ALL DATA POINTS REQUESTED WERE READ
LAST SMPL READ IF TIME ARRAY WAS FILLED AND DATA REMAINS TO BE PROCESSED

ICOUNT - NUMBER OF POINTS TO BE PROCESSED(INPUT)

ITRNS - TRANSITION VALUE(INPUT)

TIME - ARRAY CONTAINING THE TIMES OF THE FIRST 85 TRANSITIONS FOUND IN THE DATA FILE(OUTPUT). RETURNS WITH -1 IN EACH VALUE FOR WHICH NO TRANSITIONS WERE FOUND.

IBUF - ARRAY CONTAINING THE DATA TO BE PROCESSED(INPUT)

IBRATE - BASE RATE USED TO RECONSTRUCT DATA SAMPLE RATE(INPUT)

SUBROUTINE TMCVT(MBUF,IP,SEC), REV*C 07MAY83 CLF

THIS SUBROUTINE UNSCRAMBLES THE TIME CODE INFORMATION FROM A DATA OR EVENT CHANNEL.

SUBROUTINE ARGUMENTS ARE:

MBUF - SCRAMBLED TIME BUFFER-THREE WORDS(INPUT)

IP - UNSCRAMBLED TIME ARRAY(OUTPUT)
IP(1) INTEGER HOURS
IP(2) INTEGER MINUTES
IP(3) INTEGER SECONDS
IP(4) INTEGER MILLISECONDS
IP(5) INTEGER MICROSECONDS

SEC - DOUBLE PRECISION SECONDS+MILLISECONDS+MICROSECONDS(OUTPUT)

SUBROUTINE TRANS, REV*N 1NOV83 CLF

FOR EACH SELECTED CHANNEL, THIS SUBROUTINE GENERATES THE DOCUMENTATION SECTORS AND TRANSFERS DATA FROM THE ASRD TO DISC. A NEW DIRECTORY ENTRY IS CREATED FOR THE NEW DATA RECORD.

INPUT PARAMETERS PASSED THROUGH COMMON:

LU - LIST DEVICE FOR MESSAGES

IDLU - DISC LU WHERE PARAMETERS, HEADER & DIRECTORY ARE SAVED AND WHERE DATA RECORD IS TO BE STORED

IDTRK - START TRACK WHERE NEW DATA RECORD IS TO BE STORED IF -1 USE DIRECTORY TO DETERMINE START TRACK

ISKIP - SKIP OVERLOAD CHECK ON 416 IF NOT ZERO

LSTCH - LAST CHANNEL IN SYSTEM

SUBROUTINE TRTIM(TTIME,NBUF,COUNT), REV*D 4NOV83 CLF

THIS SUBROUTINE FINDS THE TIME (IN MILLISECONDS) FROM WORD 1 OF A GIVEN BALLISTIC TST DATA RECORD TO THE TIME AT WHICH THE FIRST IRIG TRIGGER TIME WORD IS INSERTED IN THE DATA.

SUBROUTINE ARGUMENTS:

TTIME - TIME IN MILLISECONDS FROM BEGINNING OF RECORD TO TIME WORDS(OUTPUT)

NBUF - ARRAY CONTAINING FOUR TIME WORDS(OUTPUT)

COUNT - LOCATION OF TIME WORDS IN RECORD(OUTPUT)

INPUT PARAMETERS PASSED IN COMMON:

ICH - CHANNEL TO PROCESS

LUDK - DISC LU WHERE DATA IS LOCATED

ITRK - START TRACK OF DATA RECORD TO BE PROCESSED

WRDCNT - ARRAY CONTAINING WORD COUNTS OF ALL CHANNELS

SUBROUTINE VIEW(LU,LULST,LUSCNR,KSCOPE), REV A 11AUG81 CLF

THIS SUBROUTINE IS USED TO CHECK THE OUTPUT OF A CARD UNDER TEST VISUALLY ON THE SCOPE AND ALLOW AN OPERATOR RESPONSE TO BE ENTERED.

SUBROUTINE ARGUMENTS:

LU - INTERACTIVE LU(INPUT)

LULST - LIST LU(INPUT)

LUSCNR - SCANNER LU(INPUT)

KSCOPE - SCANNER ADDRESS FOR SCOPE CHANNEL(INPUT)

SUBROUTINE VIO(LUSCNR,ICH1,ICH2,LUDVM,V1,V2,RATIO), REV B 10MAR82 CLF

THIS SUBROUTINE USES THE H-P 3495A SCANNER AND H-P 3455A DVM TO TAKE TWO MEASUREMENTS AND COMPUTE THEIR RATIO. THE SUBROUTINE ASSUMES THAT THE DVM HAS BEEN SET UP FOR THE MEASUREMENT TO BE MADE.

SUBROUTINE ARGUMENTS:

LUSCNR - LU OF THE SCANNER(INPUT)

ICH1 - ASCII CHANNEL NUMBER OF THE FIRST CHANNEL(INPUT)

ICH2 - ASCII CHANNEL NUMBER OF THE SECOND CHANNEL(INPUT)

LUDVM - LU OF THE DVM(INPUT)

V1 - MEASUREMENT OF CHANNEL 1(OUTPUT)

V2 - MEASUREMENT OF CHANNEL 2(OUTPUT)

RATIO - RATIO OF V2/V1(OUTPUT)

SUBROUTINE VLTIN(ICHAN,IBNK,IFC,ISCNR,MTR,RSLT),REV B 060CT80 CLF

THIS SUBROUTINE CONNECTS A CHANNEL THRU THE SCANNER TO THE DIGITAL VOLTMETER, TRIGGERS THE VOLTMETER, AND READS THE RESULT.

SUBROUTINE ARGUMENTS:

ICHAN - CHANNEL NUMBER 0 THRU 19(INPUT)

IBNK - BANK 1 THRU 4(INPUT)

IFC - FUNCTION(INPUT)
1=DC VOLTS
2=AC VOLTS
3=FAST AC VOLTS
4=2 WIRE K OHMS

ISCNR - SCANNER LU(INPUT)

MTR - DVM LU(INPUT)

RSLT - MEASUREMENT RESULT(OUTPUT)

```
   SUBROUTINE WORD(OFST,PNTS,TOTALS,RATE),  REV*D  09MAR83  CLF

THIS SUBROUTINE COUNTS THE NUMBER OF DATA, IRIG TIME,
CHANGE WORDS & MISSING AND WRONG CHANGE WORDS IN A CHANNEL.
IT ALSO COUNTS THE NUMBER OF TIMES EACH RATE IS FOUND.

SUBROUTINE ARGUMENTS:

  OFST   - OFFSET INTO THE DATA FILE(INPUT)

  PNTS   - NUMBER OF POINTS BEYOND THE OFFSET TO BE EXAMINED(INPUT)

  TOTALS - ARRAY CONTAINING RESULTS(OUTPUT)
             TOTALS(1)=NO OF DATA WORDS
             TOTALS(2)=NO OF CHANGE WORDS
             TOTALS(3)=NO OF TIME WORDS
             TOTALS(4)=NO OF MISSING CHANGE WORDS
             TOTALS(5)=NO OF DOUBLE CHANGE WORDS (DOES NOT COUNT
                       DOUBLE CHANGE WORD FOLLOWING TIME WORDS)
             TOTALS(6)=NO OF TIME WORDS NOT IN GROUPS OF THREE OR FOUR

  RATE   - ARRAY CONTAINING THE NUMBER OF OCCURRANCES OF EACH RATE(OUTPUT
             RATE(1)=NUMBER OF DATA WORDS WITH RATE 0
             RATE(2)=NUMBER OF DATA WORDS WITH RATE 1
                      .
                      .
                      .
             RATE(16)=NUMBER OF DATA WORDS WITH RATE F

INPUT PARAMETERS PASSED THROUGH COMMON:

  ICH    - CHANNEL NUMBER TO BE PROCESSED

  LUDK   - DISC LU WHERE DATA IS STORED

  ITRK   - START TRACK OF DATA RECORD TO BE PROCESSED

-----------------------------------------------------------------

   SUBROUTINE WSCNR(LUSCNR,ICHAN),  REV A  160CT81  CLF

THIS SUBROUTINE WRITES A CHANNEL ADDRESS TO THE H-P 3495A SCANNER.
THE ALLOWED VALUES ARE ASCII 00 THRU 79.  OTHER VALUES WILL CAUSE NO
ACTION BY THE SCANNER.  AN INPUT OF ASCII "C " WILL CLEAR THE SCANNER.

SUBROUTINE ARGUMENTS:

  LUSCNR - LU OF SCANNER(INPUT)

  ICHAN  - ASCII CHANNEL NO(INPUT)

-----------------------------------------------------------------

   SUBROUTINE ZERO(NASC),  REV A  04NOV80 CLF

THIS SUBROUTINE INSERTS ZEROS IN PLACE OF BLANKS IN A SIX
CHARACTER ASCII ARRAY.

-----------------------------------------------------------------
```

```
      SUBROUTINE ZSPLN(N,C,D,Z,OFSET,LASTY,AGAIN,OLD,IBRATE),   REV*B  29
     $AUG83  GZT

THIS ROUTINE READS  N  DATA ITEMS USING JDATA AND COMPUTES
SECOND DERIVATIVE VALUES NEEDED FOR CUBIC SPLINE INTERPOLATION.
   NOTE: LINEAR ENDPOINTS ARE ASSUMED.

SUBROUTINE ARGUMENTS:

  N       - NUMBER OF DATA WORDS USED FOR THE CUBIC SPLINE CURVE(INPUT)

  C       - ARRAY OF SUPER-DIAGONAL MATRIX ENTRIES(OUTPUT)

  D       - ARRAY OF DIAGONAL MATRIX ENTRIES(OUTPUT)

  Z       - ARRAY OF SECOND DERIVATIVE VALUES(OUTPUT)

  OFSET   - OFFSET INTO THE DATA FILE(INPUT)

  LASTY   - LAST DATA VALUE(REAL OR ADDED) FROM PREVIOUS CALL(INPUT)

  AGAIN   - TRUE IF SUBROUTINE HAS BEEN CALLED PREVIOUSLY
            NEEDED BECAUSE NEITHER OLD OR FINAL CAN BE CHANGED

  OLD     - SLICE OF TIME BETWEEN LASTY AND NEXT REAL DATA ITEM(INPUT)

  IBRATE  - BASE RATE USED TO RECONSTRUCT DATA SAMPLE INTERVAL(INPUT)

PARAMETERS INPUT THROUGH COMMON:

  ICH     - CHANNEL NUMBER

  WRDCNT  - NUMBER OF WORDS IN CHANNEL RECORD
```

FORTRAN CALLABLE SUBPROGRAMS WRITTEN IN ASSEMBLY LANGUAGE

```
NAM IASRD,6    REV F 10 APR 83


ROUTINE TO COMMUNICATE WITH THE IMPROVED ASRD
CALLING SEQUENCE IS:
     CALL IASRD(ITYP,ICMND,IERR,IBUF,NWDS)

 ITYP    TRANSACTION TYPE
         = 1  INPUT TO COMPUTER
         = 2  OUTPUT TO IASRD
         = 3  CONTROL
         = 4  1ST SEGMENT OF A CHAINED INPUT
         = 5  1ST SEGMENT OF A CHAINED OUTPUT
 ICMND   COMMAND CODE
         = CHANNEL NUMBER FOR CHANNEL DESIGNATION (TYPE 3)
         = 1000B FOR GENERAL ARM (TYPE 3)
         = 2000B FOR MICROPROCESSOR RESET (TYPE 3)
         = 4000B FOR GENERAL RESET (TYPE 3)
         =10000B FOR SYNC (TYPE 3)
         <0 FOR CLEAR
         = DEVICE COMMANDS FOR TYPE 1 OR 2
 IERR    ERROR CODE
         =  0 NO ERROR
         = -1 TIME OUT
         = -2 ILLEGAL TRANSACTION CODE
         =  1 ILLEGAL COMMAND CODE
         =  2 CHANNEL BUSY
 IBUF    DATA BUFFER
```

```
NAM IOPSY,6   REV A  28JAN83  CLF

THIS ROUTINE GETS THE OPERATING SYSTEM ID CODE USING SYSTEM
SUBROUTINE .OPSY AND RETURNS THE CODE TO THE CALLING ROUTINE

CALL IOPSY(ID)

  ID = OPERATING SYSTEM ID CODE:  -9 = RTE IV
                                 -17 = RTE 6/VM
                                  -1 = RTE III
```

```
NAM MCKT

FORTRAN CALLABLE SUBROUTINE MCKT FOR INPUT AND OUTPUT THROUGH
A MICROCIRCUIT INTERFACE CARD.  CALLING SEQUENCE IS:
       CALL MCKT(IOPER,MCSC,IBUF,IBUFL,IOPTN),   WHERE:
                IOPER = OPERATION
                      = 1 FOR INPUT TO COMPUTER,
                      = 2 FOR OUTPUT TO DEVICE
                MCSC  = MICROCIRCUIT INTERFACE SELECT CODE
                IBUF  = INPUT/OUTPUT BUFFER
                IBUFL = BUFFER LENGTH
                IOPTN = OPTIONS AVAILABLE
                          BIT 0 = 1 FOR WAIT ON DEVICE FLAG
                                = 0 FOR NO WAIT ON FLAG
                          BIT 1 = 1 TO ISSUE CONTROL TO DEVICE
                                = 0 NO CONTROL BIT TO DEVICE
```


(Page P-32 Blank)

APPENDIX Q - TRAILER SETUP AND PACKUP CHECKLISTS

SET UP CHECKLIST

Set up platform, stairs and guard rails.
Install power cable.
Engage door safety lock if stairs are not installed.
Check that A/C Selector Switch is off.
Check power panel for phase/voltage. Correct if necessary.
Remove Graphics Terminal tie down kit.
Power up.
Turn on A/C as appropriate.
Install disc pack (if necessary).
Boot up computer.
Remove plotter and chair tie downs.
Turn on emergency light power if necessary.
Install exterior light(s) if night operation is anticipated.
Activate memory back up power if required.
Install IRIG and commo radio antennas.
Unlock storage cabinets and unpack needed equipment.

PACK UP CHECKLIST

Back up disc data on mag tape.
Remove and store disc pack if a long move.
Remove cables from Turkey hole and seal hole.
Remove cables from signal entrance panel and secure panel doors.
Remove outside lights and store inside van if necessary.
Pack equipment and lock storage cabinets.
Remove IRIG and commo radio antennas.
Turn off memory back up power if installed.
Tie down plotter and chairs.
Turn A/C Selector Switch to off.
Power down.
Install graphics terminal tie down kit.
Disconnect and store power cable.
Check that rear doors are closed and latched.
Check that all equipment and supplies are secure.
Lock van doors.
Disassemble and store guard rails, platform and stairs.

APPENDIX R - ELIMINATING GROUND LOOPS

One of the biggest problems encountered in data acquisition is eliminating the undesireable effects of ground loops. Ground loops are produced when two or more points in the measurement system, that are nominally at ground potential, are connected by a conducting path. If the conducting path is a signal cable and if the potential of the two points is different, then current will flow in the signal return line generating an undesireable noise signal. Ground loops may appear in ballistic testing since safety requirements mandate grounding the weapon where transducers are located and the trailer where recording instruments are located. Within a rack of equipment, ground loops can be generated by the bonding of metal chassis and the safety ground wire in a three wire electrical cord.

A number of features have been built into the BTST data acquisition equipment to help eliminate the effects of ground loops. The operator must be aware of these features and take advantage of them when possible. The features are:

a. Each channel in the signal conditioner is totally independent of the others. There are no common grounds between channels. Each channel has its own isolated power supply.

b. Jumpers are available on each signal conditioner card which allow ground paths to be broken if necessary. Figure R-1 shows where these jumpers are electrically located while Figure R-2 shows where they are physically located. Often the removal of a single channel's jumper is all that is required.

c. The high input impedance differential amplifier in the filter/amplifier provides a very high impedance return path on the signal leads.

d. The input shield on the filter/amplifier may be disconnected from signal ground if required. Figure R-3 shows the electrical location of this jumper while Figure R-4 shows where it is physically located.

e. The signal ground of the filter/amplifier may be connected to or separated (by 100 ohms) from chassis ground by means of a switch on the back of the filter/amplifier. The electrical location of this switch is shown in Figure R-3.

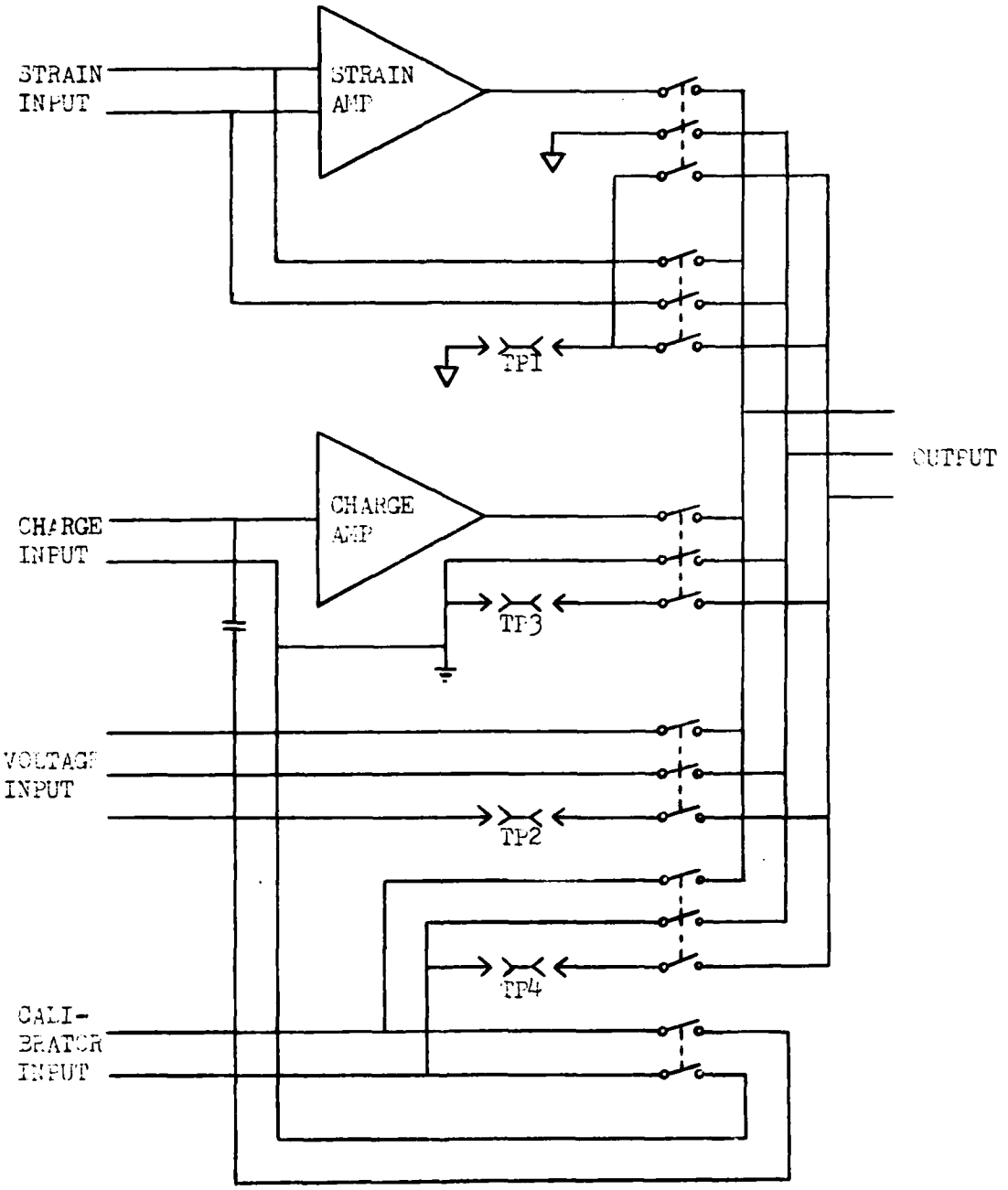


Figure R-1. Electrical locations of jumpers for breaking ground loops on signal conditioner.

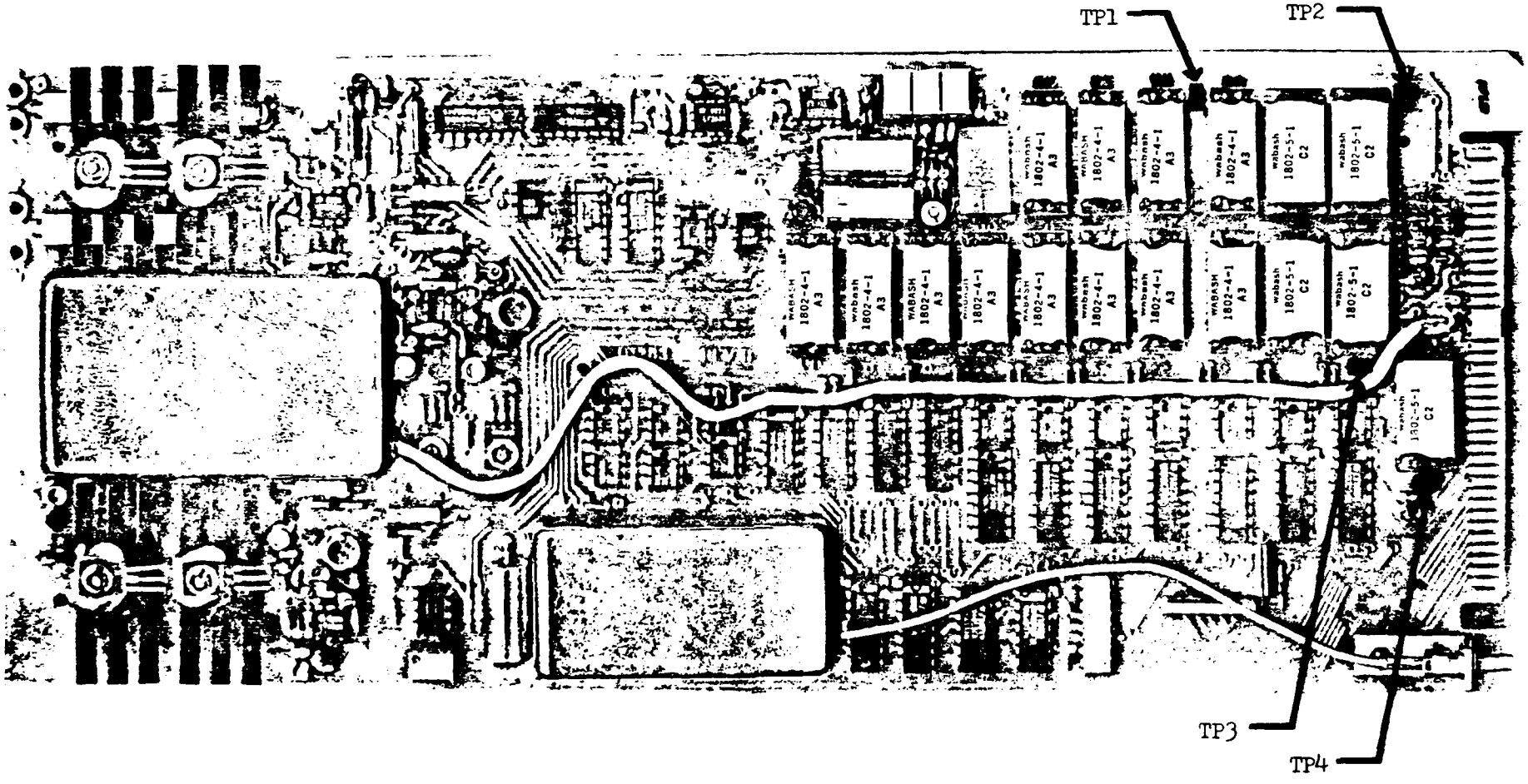


Figure R-2. Physical locations of jumpers for breaking ground loops on signal conditioner card.

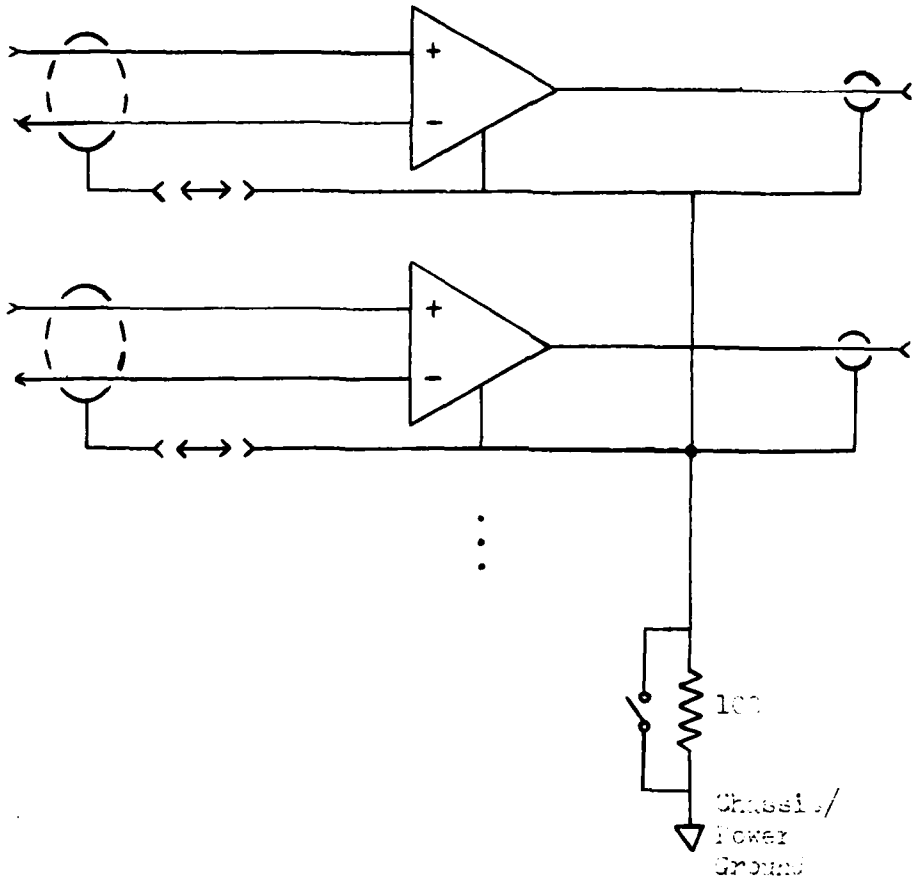


Figure R-3. Electrical location of jumpers and switch for breaking ground loops on filter/amplifier.

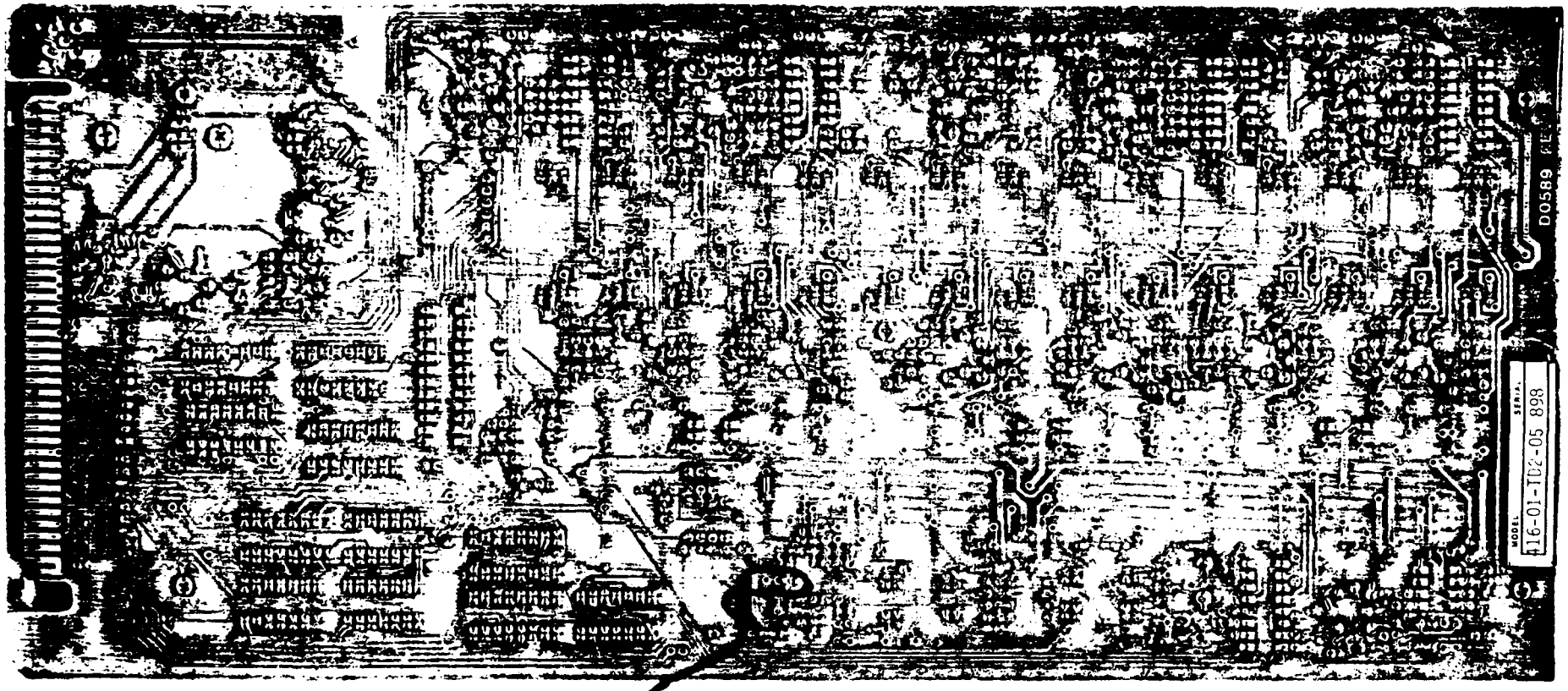


Figure R-4. Filter/amplifier shield grounding point location.

In most cases there is no magic cure for ground loop problems. However, by making use of the features outlined above, most ground loop problems should be correctable. The operator should follow these additional precautions:

a. Check that all cables have good outer insulation (no bare shields) and that connectors are not allowed to touch each other or metal chassis parts.

b. When jumpers are removed or added, label the channel involved. Return all channels to their original configuration at the end of the test.

c. Be especially careful if connecting additional equipment into the BTST. Most commercial test equipment uses single ended inputs tied to chassis common. This arrangement can quickly defeat all of the precautions built into the system.

APPENDIX S - INTERPRETING AN ASRD DATA RECORD

Figure S-1 is a simplified block diagram of the data recording logic of the ASRD. The microprocessor subsystem, register control, and channel interface are not shown on this diagram. The analog signal is converted to a 12 bit digital signal by the sample and hold and A/D converter. Note that these two modules are continuously clocked at 800 kHz. The output of the A/D is then applied simultaneously to an 80 word shift register, the internal trigger circuitry, and the variable rate circuitry.

If internal trigger is selected, the eight most significant bits of the signal are compared to the programmed threshold. With eight bits, the resolution of the threshold circuitry is 0.08 V. When the threshold, slope, and run delay conditions are met, the record logic will start the memory word counter to save data in memory.

If variable sample rate is selected, the absolute value of the difference between the current and previous samples is calculated. Only the ten most significant bits of the signal are used. This word is applied simultaneously to the summing input of an accumlator and to a programmable variable length shift register. The output of the variable length shift register is applied to the difference input of the accumulator. The result of these operations is to form a running sum of the absolute value of the sample to sample differences. This value is then operated on by a programmable scalar to produce an eight bit output. This value represents the average rate of change of the signal. These eight bits are used as an address for the programmable rate table. If the rate increases, this information is passed to the record logic immediately. However, if the rate decreases the programmable decrease delay circuitry holds off the decrease in rate until the delay has expired.

The data is passed through an 80 word shift register to provide time for the triggering and rate determining circuitry to perform their functions. The output of the shift register feeds a tri-state bus which can also be driven by the change word counter or the trigger time register. This allows all three types of words to be inserted at any time since each uses a different time slice of the master 8 MHz clock. A 40 word first-in-first-out (FIFO) register provides buffering between the potentially high burst rate (8 MHz) possible if all three types of data must be inserted in one master cycle and the 1 MHz maximum input rate of the memory. Although the input of the FIFO has the 800 ksps data available, samples are clocked into the FIFO at a rate determined by the record logic. Thus, it is at this point that rates lower than 800,000 sps are obtained by discarding samples. Change and trigger time words are always clocked into the FIFO by the record logic. Following the FIFO is another 80 word shift register which provides a minimum of 80 pretrigger samples. The output of this shift register goes to the data memory board.

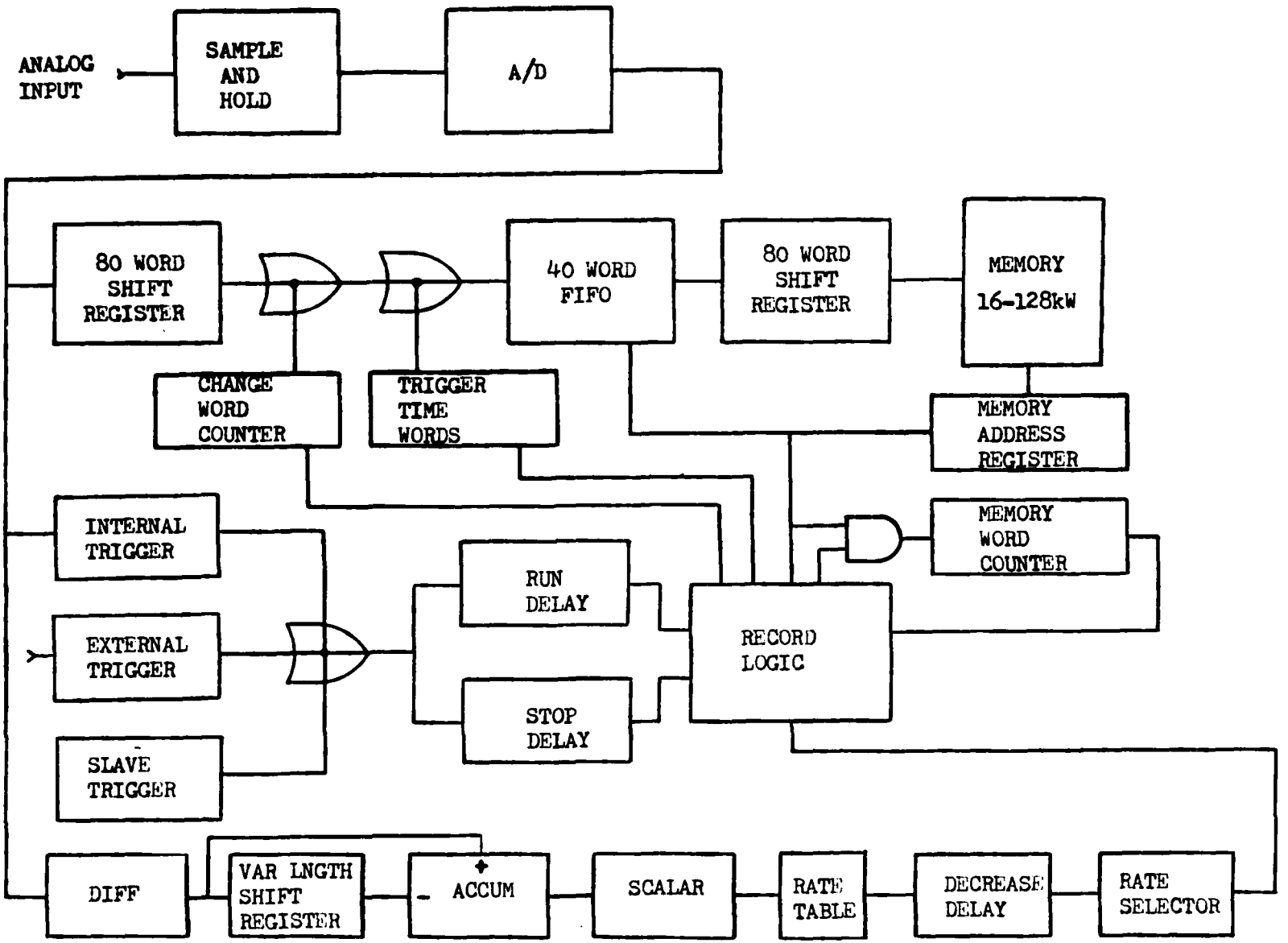


Figure S-1. Simplified block diagram of ASRD data recording circuitry.

The presence of the second 80-word shift register results in the necess to guarantee that sufficient stop delay is available to pump the contents of the shift register into memory. For example, at rate 9 (12,500 sps), 6.4 ms is required to move a sample through the shift register. A second consideration to keep in mind is that if the ASRD triggers immediately when armed, then the 80 words in the shift register which represent old data will be pumped into memory ahead of the new data points. If this condition is anticipated, an amount of time greater than 80/(lowest possible rate) should be programmed for the run delay. With most BTST operations the shift register is flushed and a run delay is not required.

The trigger words are inserted in the data stream at the time the trigger is generated. But recall that the sample which generated the trigger is at the input to the first 80-word shift register while the trigger time is being inserted after the shift register output. The current recording rate determines how many of the 80 samples stored in the first shift register will appear in the data record. For rate codes 2-8 the sample causing the trigger will immediately follow the trigger words, while for rate codes 9 through F the sample causing the trigger will be positioned

$$\frac{80}{2^{15-\text{RATE CODE}}}$$

samples after the trigger words (assuming the rate doesn't change). Regardless of the number of samples, the time from trigger sample to trigger words is 100 microseconds (80 samples/800,000 sps).

The trigger time words will normally start at word 80 in the data record due to the second 80 word shift register. At higher rates this may shift to a slightly higher value. If a pretrigger memory value is programmed into the ASRD, then the trigger words will be positioned in the data record at sample

$$80 + 1024 * \text{PRETRIGGER VALUE}.$$

In the preceding discussion, it was assumed that zero run delay was used. If a nonzero run delay is used, then the trigger time words are inserted and positioned exactly as described above. However, the position of the sample producing the trigger cannot be specified exactly as was done above. In fact, if the run delay is large enough, the sample causing the trigger may not even be recorded in the data record. In addition, since the 781.25 Hz clock controlling the run (and also the stop) delay is not synchronized with the trigger event, the value of run delay specified is potentially off by 1.28 ms. For example, a specified value of 1 could fall between 0 and 1.28 ms, while a value of 2 could fall between 1.28 and 2.56 ms.

In order to allow channel-to-channel time coordination, the first three trigger words provide an absolute time in hours, minutes, and seconds with a resolution to 1 microsecond. Since the sample which caused the trigger may be discarded if the recording rate is less than rate code F, the fourth trigger word provides a count of rate F samples from the previous recorded sample. With this information the absolute time of any data sample can be calculated. It is possible that if the rate changes on the same master clock cycle that the trigger words are inserted, then the fourth time word will not be present but a change word will be inserted. In this case, the change word should be interpreted as a relative trigger word also.

When rate changes occur, a change word is inserted between the last sample at the old rate and the first sample at the new rate to indicate the number of rate code F samples that actually elapsed between the two recorded samples. The change word count can go from 1 to 800,000/lowest possible rate used. Figure S-2 is a timing diagram of some of the rate clocks used in the ASRD. When an increase or decrease in rate is called for by the rate determining circuitry, an immediate increase or decrease may not be possible if it is not the appropriate rate clocks turn to insert a data word. In Figure S-2, if the current rate is 9, the current sample is represented by the left most rate 9 mark and that sample causes an increase in rate to B, then the change word will be 16 as the ASRD must wait 16 master clock cycles before the next rate B clock is allowed. As a second example, if the current rate is C, the current sample is the third sample to the right of the left most rate C mark and the rate is to decrease to A, then the change word will be 29.

It is possible for two samples having identical rates to have a change word inserted between them. This will occur, for example, if after the last recorded sample, the rate determining circuitry made a decision to go to a lower rate; but before the first sample at the lower rate is recorded, the rate determining circuitry made a decision to go back to the original rate. In this case, the change word must be used since the time between the two samples may not be the normal amount.

The sample causing an increase in rate will be present at the input to the first 80 word shift register at the time the change word is being inserted after the shift register output. Thus, the location of the change word relative to the word causing the increase will be the same as for trigger time words.

This is not the case for a decrease in rate. When a decrease in rate occurs, the decrease delay must be taken into account. Normally the size of the decrease delay will exceed 80, so the sample causing the decrease will precede the change word by an amount giving by

$$\frac{\text{DECREASE DELAY - 80.}}{2^{15}\text{-RATE CODE}}$$

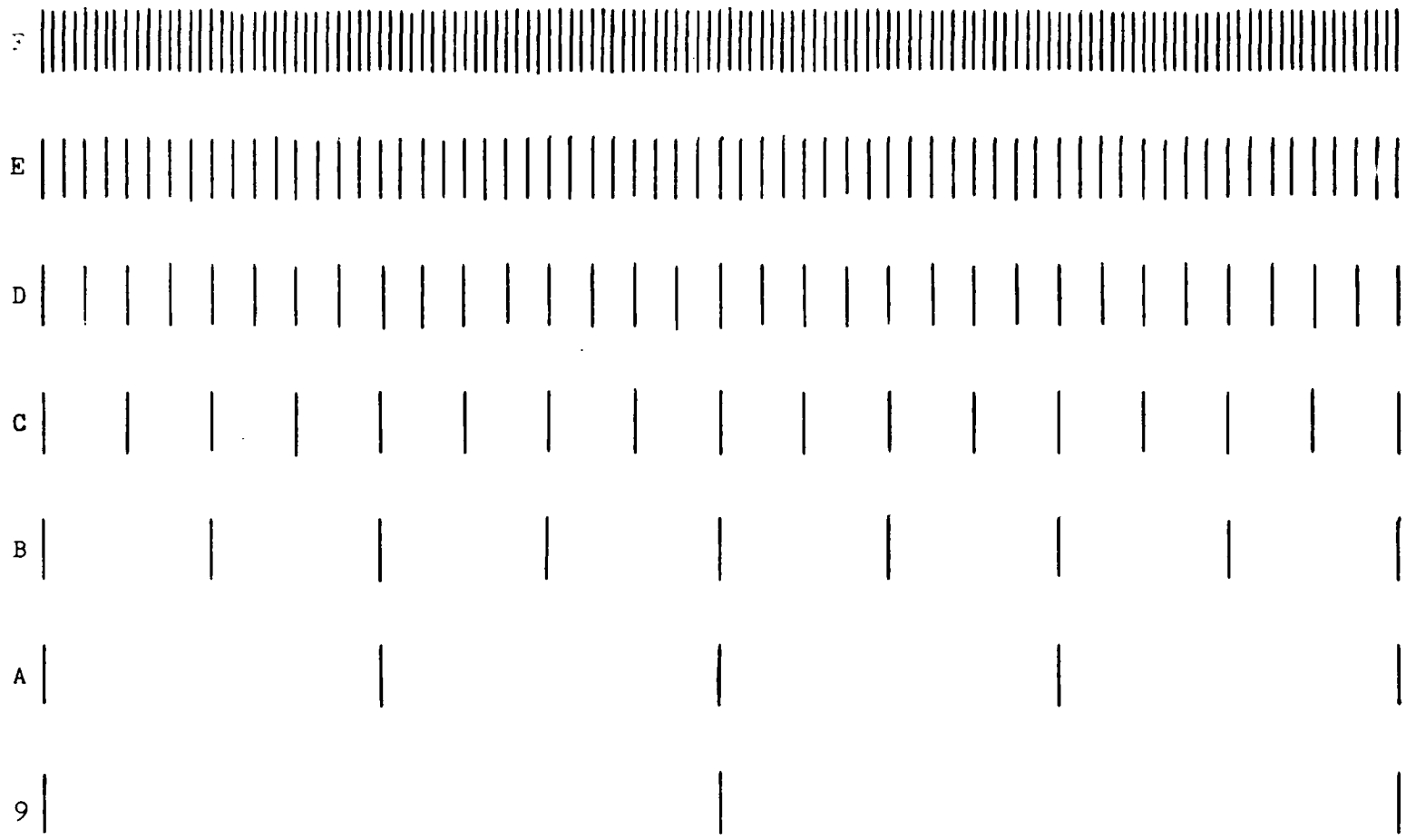


Figure S-2. ASRD rate code timing sequence.

Figure S-3 is a plot of a single cycle of a sawtooth wave recorded on the ASRD. The plot is done in point mode so that the changes in rate can be seen. Qualitatively, examination of Figure S-3 shows that initially a low rate is used. The rate increases to capture the rising edge of the sawtooth wave, decreases on the less rapidly changing falling edge, then increases again for the second rising edge and finally decreases back to the original lower rate after the input signal terminates. The setup conditions used for generating Figure S-3 are in Table S-1.

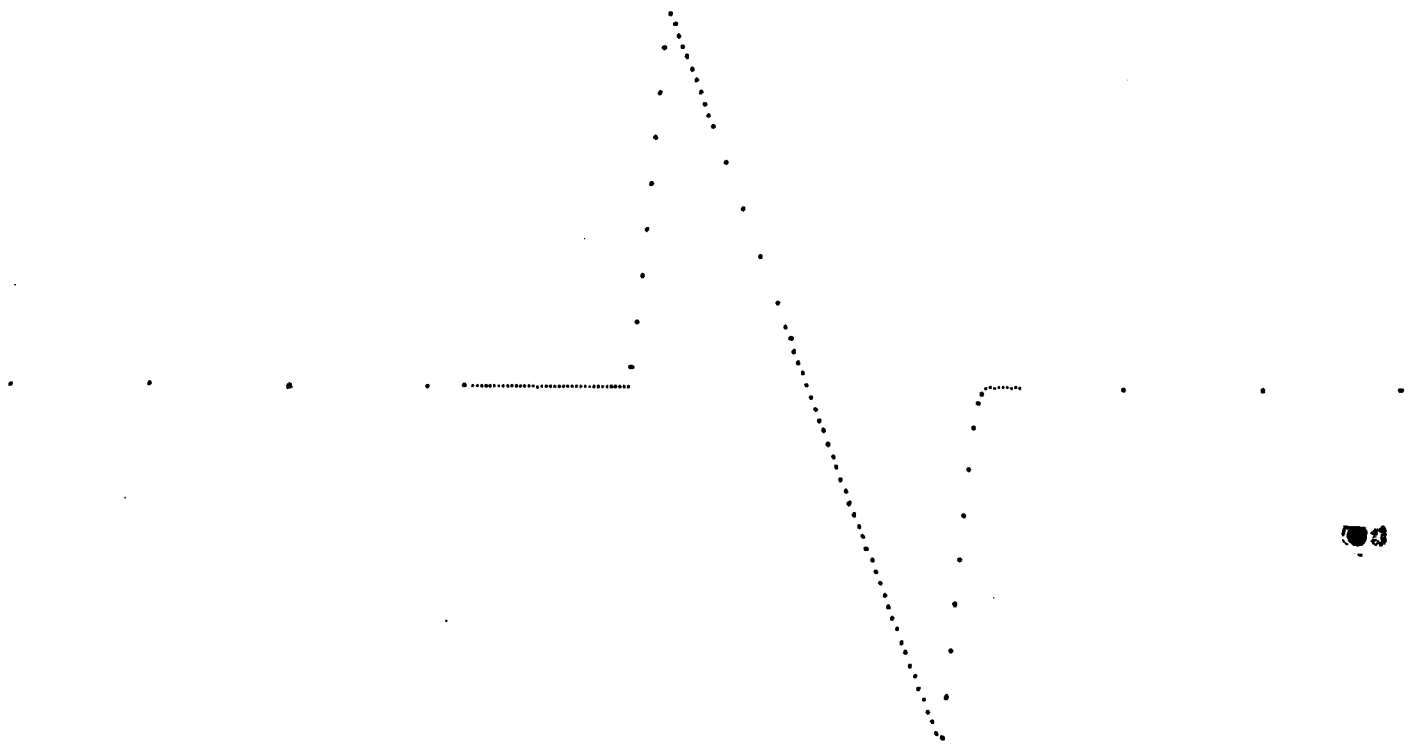


Figure S-3. Point plot of sawtooth waveform recorded on ASRD.

TABLE S-1. SETUP CONDITIONS FOR SAWTOOTH WAVEFORM EXAMPLE

INPUT SIGNAL:	12 V peak-to-peak amplitude 5 kHz frequency Triangle wave with 20% duty cycle.
ASRD PARAMETERS:	Internal trigger +5 V threshold + slope 0 run delay 100 ms stop delay Rate table 7 Summing length 8 Scalar 0 Decrease delay 100.

RATE TABLE 7:

SUM	RATE CODE
0 - 7	9
8 - 15	A
16 - 31	B
32 - 63	C
64 - 127	D
128 - 255	E

Table S-2 lists the data points that were used to generate Figure S-3. These points are only a portion of the data record taken by the ASRD. Samples 1 - 67 and 209 - 1023 were all at rate 9 and are of no interest for this discussion. Table S-2 shows the sample numbers and the data bits for samples 68 - 208. If the word is a data sample then the data bits column is divided from left to right into identification bit, 3 rate bits and 12 A/D bits. For data samples the rate bits are interpreted and the rate code is listed, the /D bits are interpreted as both a 12 bit count and a 10 bit count. The absolute value of the sample-to-sample difference in the 10 bit count is listed in the DIFF column. Finally, the running sum of eight (in this case) master clock sample differences is listed in the SUM column. The DIFF and SUM columns are an attempt to reconstruct the internal workings of the ASRD rate determining circuitry. If the word is a change word then the data bits column is divided into two identification bits and 14 change word bits. The designation CW is given with the interpreted value of the change word. If the words are trigger time words, then the data bits column is divided into 2 identification bits and 14 time bits. The designation TTW is given for the first three trigger time words, while the designation RT is given with the interpreted relative time for the fourth trigger time word. On the right hand side of Table S-2 letters have been added to designate important events which will be discussed below.

TABLE S-2. LIST OF SAMPLES RECORDED FOR SAWTOOTH WAVEFORM

SAMPLE	DATA BITS	RATE	12 BIT CNT	10 BIT CNT	DIFF	SUM	
68	0 001 0000000010000	9	16	4	0	0	
69	0 001 0000000001110	9	14	3	-1	1	
70	0 001 0000000001111	9	15	3	0	0	
71	0 001 0000000001110	9	14	3	0	0	
72	0 001 0000000001111	9	15	3	0	0	
73	10 0000000000010000	CW =	16				
74	0 100 0000000001111	C	15	3	0	0	A
75	10 0000000000000100	CW =	4				
76	0 101 0000000001110	D	14	3	0	0	B
77	10 0000000000000010	CW =	2				
78	0 110 0000000001110	E	14	3	0	0	C
79	0 110 0000000001111	E	15	3	0	0	
80	0 110 0000000001111	E	15	3	0	0	
81	0 110 0000000001110	E	14	3	0	0	
82	11 0111001111001101	TTW					D
83	11 0100101010100001	TTW					
84	11 1010001100001111	TTW					
85	11 0000000000000001	RT =	1				
86	0 110 0000000001111	E	15	3	0	0	
87	0 110 0000000001111	E	15	3	0	0	
88	0 110 0000000001110	E	14	3	0	0	
89	0 110 0000000001111	E	15	3	0	0	
90	0 110 0000000010000	E	16	4	1	1	
91	0 110 0000000001110	E	14	3	-1	0	
92	0 110 0000000001110	E	14	3	0	0	
93	0 110 0000000001110	E	14	3	0	0	
94	0 110 0000000001111	E	15	3	0	0	
95	0 110 0000000001111	E	15	3	0	0	
96	0 110 0000000001100	E	12	3	0	0	
97	0 110 0000000001110	E	14	3	0	0	
98	0 110 0000000001111	E	15	3	0	0	
99	0 110 0000000010000	E	16	4	1	1	
100	0 110 0000000001101	E	13	3	-1	0	
101	0 110 0000000001111	E	15	3	0	0	
102	0 110 0000000001111	E	15	3	0	0	
103	0 110 0000000001111	E	15	3	0	0	
104	0 110 0000000001110	E	14	3	0	0	
105	0 110 0000000001110	E	14	3	0	0	
106	0 110 0000000001110	E	14	3	0	0	
107	0 110 0000000001110	E	14	3	0	0	
108	0 110 0000000001101	E	13	3	-1	0	
109	0 110 0000000001111	E	15	3	0	0	
110	0 110 0000000010000	E	16	4	1	1	
111	0 110 0000000001101	E	13	3	-1	0	
112	0 110 0000000001111	E	15	3	0	0	
113	0 110 0000000001111	E	15	3	0	0	
114	0 110 0000000001110	E	14	3	0	0	
115	0 110 0000000001110	E	14	3	0	0	
116	0 110 0000000001101	E	13	3	-1	0	
117	0 110 0000000001110	E	14	3	0	0	
118	0 110 0000010100001	E	81	0	-17	-17	
119	0 110 0000111100111	E	231	7	7	-10	a
120	0 110 0001011111111	E	383	7	7	-3	
121	0 110 0010000011000	E	536	134	99	96	c
122	0 110 0010101011111	E	687	171	37	133	
123	0 110 0011010001000	E	840	210	77	210	
124	0 110 0011111011110	E	990	247	37	247	
125	0 110 0100011110011	E	1139	284	37	284	d
126	0 110 0100011100000	E	1248	312	28	312	
127	0 110 0100011000010	E	1218	304	-8	304	e
128	0 110 0100100111110	E	1182	295	-9	295	
129	0 110 0100011111001	E	1145	286	-9	286	
130	0 110 0100001010101	E	1109	277	-9	277	
131	0 110 0100001011101	E	1069	267	-10	267	
132	0 110 0100000001000	E	1032	258	-9	258	
133	0 110 0011111011111	E	991	247	-11	247	

TABLE S-2 (CONT'D)

EVENT A. The first event of interest is the increase in rate occurring at sample 74. A SUM value of 32 or more is required to increase the rate to C. The first recorded sample where this occurs is 119 which has been marked as a. Interpolation between samples 118 and 119 gives a SUM value of 35. So, in reality the increase in rate was probably related to a sample which was not saved. A change word of 16 was inserted at sample 73.

It should be noted that at sample 118 SUM was adequate to initiate a change to rate A. However, due to the clock sequence, the ASRD did not have a chance to record a rate A sample before the rate determining circuitry went to rate C.

EVENT B. The next event of interest is the increase in rate occurring at sample 76. A SUM value of 64 or more is required to increase the rate to D. The first recorded sample where this occurs is 120 which has been marked as b. Again the sample producing the change was probably the previous unrecorded sample. A change word of 4 was inserted at sample 75.

EVENT C. At sample 78 the rate increases to E, the highest value allowed in the table. A SUM value of 128 or more is required to increase the rate to E. The sample which produced this change is 121 which has been marked as c. A change word of 2 was inserted at sample 77.

EVENT D. The trigger time words were inserted at samples 82 - 85. The first recorded sample where the threshold exceeds 5 V (count of 1024) is 125 which has been marked as d. The first three words convert to a time of hours, minutes, and seconds. The relative time word count of 1 indicates that this time applies to sample 81. Interpolation between samples 126 and 127 gives a count value of 1065 for the unrecorded sample. So the trigger was probably due to a sample which was not recorded.

EVENT E. At sample 138 the rate decreases to c. Looking backward in the record, the SUM drops below 132 at sample 127. This should result in a decrease to rate D and start the decrease delay circuitry. However, at sample 129 the SUM drops below 64 which should cause a decrease to rate C. If the decrease delay from the first decrease transition has not timed out when a second decrease transition is received, then the decrease delay is <u>not</u> reset and the recorded samples will show only the last calculated rate.

EVENTS F and G. These two increases in rate are similar to events B and C, respectively.

EVENT H. The decrease to rate 9 is similar to event E except the SUM eventually drops below 8 to allow a return to the lowest rate in the table.

APPENDIX T - REFERENCES

1. Drabo, Fasig, and MacDonald, Final Report, Advanced Data Collection Techniques, TECOM Project No. 9-CO-001-000-133. US Army Aberdeen Proving Ground, Report APG-MT-4493, November 1974. (Distribution controlled by TECOM, ATTN: AMSTE-ME. AD B004 526L.)

2. Abramovitz and Harley, RDI Task, Final Report of Automated Data Acquisition and Processing Technology (ADAPT), TECOM Project No. 5-CO-APO-ADP-601. US Army Aberdeen Proving Ground, Report APG-MT-5096, April 1978. (Distribution controlled by TECOM, ATTN: DRSTE-AD-I. AD B028 166L.)

3. Cunningham, Harley, and Paules, RDI Task, Final Report of Automated Data Acquisition and Processing Technology (ADAPT), TECOM Project No. 5-CO-APO-ADP-601. US Army Aberdeen Proving Ground, Report APG-MT-5292, November 1979. (Distribution controlled be TECOM, ATTN: DRSTE-AD-I. AD B044 520L.)

4. System 316 Instruction Manual, Precision Filters, Inc., Ithaca, NY, January 1982.

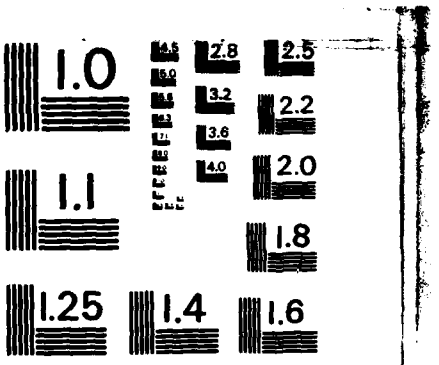
5. System 416 Instruction Manual, Precision Filters, Inc., Ithaca, NY, July 1979.

6. Harley, RDI Task, Final Report of Research and Development of Software, Adaptive Sample Rate Digitizer, TECOM Project 5-CO-APO-DFW-203. US Army Aberdeen Proving Ground, Report APG-MT-5956, January 1984. (In process.)

(Page T-2 Blank)

AD-A139 956 RDI TASK FINAL REPORT OF RESEARCH AND DEVELOPMENT OF SOFTWARE BALLISTIC T..(U) ABERDEEN PROVING GROUND MD MATERIEL TESTING DIRECTORATE C L FRANCIS JAN 84 4/4

UNCLASSIFIED APG-MT-5952 F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

APPENDIX U - ABBREVIATIONS

A = ampere
AC = alternating current
ADAPT = automated data acquisition and processing technology
ASCII = American Standard Code for Information Interchange
ASRD = adaptive sampling rate digitizer
BCD = binary coded decimal
BITE = built-in test equipment
BPI = bytes per inch
CRT = cathode ray tube
DAC = digital to analog converter
dB = decibels
DC = direct current
DCPC = dual channel port controller
DMA = direct memory access
DVM = digital voltmeter
EPROM = erasable, programmable read-only memory
EQT = equipment number
FIFO = first-in first-out
FM = frequency modulation
GPIB = general purpose interface bus
IRIG = inter-range instrumentation group
I/O = input/output
k = 1024
kHz = kiloHertz
LU = logical unit
mA = milliamperes
MEM = memory expansion module
mV = milliVolts
pC = picocoulombs
PCT = programmable counter/timer
PSG = programmable signal generator
RAM = random access memory
RMS = root mean square
RTE = real time executive
TCG/T = time code generator/translator
TTL = transistor transistor logic
SC = select code

(Page U-2 Blank)

APPENDIX V - DISTRIBUTION LIST

TECOM Project No. 5-CO-APO-DFW-203

Addressee	No. of Copies
Commander	
US Army Test and Evaluation Command	
ATTN: DRSTE-AD-I	5
DRSTE-AD-M	1
DRSTE-MS	1
DRSTE-TO-F	1
DRSTE-CM-A	1
Aberdeen Proving Ground, MD 21005	
Commander	
US Army Cold Regions Test Center	
ATTN: STEAC-TD-MI	1
STECR-TD	1
APO Seattle 98733	
Commander	
White Sands Missile Range	
ATTN: STEWS-TE-PL	5
STEWS-ID	1
STEWS-PL/ID	1
White Sands, NM 88002	
Commander	
US Army Yuma Proving Ground	
ATTN: STEYP-MDP	1
STEYP-MNI	1
STEYP-MMI	1
Yuma, AZ 85364	
Commander	
US Army Dugway Proving Ground	
ATTN: STEDP-MT-T	1
STEDP-PP	1
Dugway, UT 84022	
Commander	
US Army Electronic Proving Ground	
ATTN: STEEP-MT-T	2
Fort Huachuca, AZ 85613	
Commander	
US Army Tropic Test Center	
ATTN: STETC-TD	1
STETC-TD-M	1
APO Miami 34004	

Addressee	No. of Copies
Commander US Army Training and Doctrine Command ATTN: ATCD-TC-I Fort Monroe, VA 23651	1
Commander US Army Jefferson Proving Ground ATTN: STEJP-TD STEJP-TD-D Madison, IN 46250	1 20
Commander US Army Aircraft Development Test Activity ATTN: STEBG-DS Fort Rucker, AL 36362	1
Commander Armament Research and Development Center US Army Armament, Munitions and Chemical Command ATTN: DRSMC-TSI(D) Dover, NJ 07801	1
Director US Army Ballistic Research Laboratory/ARDC ATTN: DRSMC-BLA-S(A) DRSMC-BLL(A) Aberdeen Proving Ground, MD 21005	2 1
Commander US Army Aberdeen Proving Ground ATTN: STEAP-MT-D STEAP-MT-G, Mr. Fasig STEAP-MT-GI, Dr. Francis STEAP-MT-A STEAP-MT-T STEAP-MT-I STEAP-MT-U STEAP-MT-MM, Mr. Giroux STEAP-MT-GB-F Aberdeen Proving Ground, MD 21005	1 2 20 1 1 5 1 2 40
Administrator Defense Technical Information Center ATTN: DDA Cameron Station Alexandria, VA 22314	2

Secondary distribution is unlimited.

END

FILMED

5-84

DTIC