Title
DESCRIPTION, FIELD TEST AND DATA ANALYSIS OF A CONTROL-SOURCE EM SYSTEM (EM-60)

Author
Morrison, H. F.

Publication Date
1978-10-01

Presented at the Lawerence Berkeley
Laboratory, University of California
Exhibit in conjunction with the 48th
Annual Meeting of the Society of
Exploration Geophysicist, Richmond Field
Station, Lawerence Berkeley Laboratory,
Berkeley, California, November 3, 1978

DESCRIPTION, FIELD TEST AND DATA ANALYSIS OF A CONTROL-SOURCE EM SYSTEM (EM-60)

H. F. Morrison, N. E. Goldstein, M. Hoversten,
G. Oppliger, and C. Riveros

October 1978

Prepared for the U. S. Department of Energy
under Contract W-7405-ENG-48

DISCLAIMER

LBL-7088 DRAFT

DESCRIPTION, FIELD TEST AND DATA ANALYSIS OF A CONTROLLED-SOURCE EM SYSTEM (EM-60)

H.F. Morrison[1], N.E. Goldstein[2], M. Hoversten[1], G. Oppliger[1], and C. Riveros[2]

[1] Department of Materials and Mineral Sciences
Engineering Geosciences
University of California
Berkeley, California 94720

[2] Earth Sciences Division
Lawrence Berkeley Laboratory
Berkeley, California 94720

October 1978

Work performed for the U.S. Department of Energy
Division of Geothermal Energy, under Contract W-7405-ENG-48

TABLE OF CONTENTS

Table of Contents
(continued)

INTRODUCTION

Electrical and electromagnetic (EM) techniques have been shown to be useful for delineating either the gross geological structure or the reservoir region of some convective geothermal systems. This application is based on the relationship between the bulk resistivity of the reservoir region and a complex function involving, among other things, temperature, type and concentration of ionic species, presence of a gas phase, effects from conducting sulfide or clay minerals, and fracture permeability. It has been noted in the literature that liquid-dominated geothermal systems exhibit a lower resistivity than surrounding rocks for several possible reasons: (a) increased ion mobility; (b) a higher concentration of ions; and (c) increased permeability and/or porosity. However, there is evidence that some geothermal reservoirs exhibit higher bulk resistivity than surrounding rock because of a vapor phase (The Geysers) or a porosity loss caused by secondary minerals (Cerro Prieto).

Of the techniques available to determine subsurface resistivities, dc resistivity has been the most widely used, but the magnetotelluric (MT) method has also been used in both reconnaissance and detailed studies; and several controlled-source EM techniques have been tried as well (Keller and Rapolla, 1976; Harthill, 1976; Jackson and Keller, 1972; Ghosh and Hallof, 1973; and Keller, 1970).

In the LBL/U.C. Berkeley evaluation of geophysical techniques for geothermal exploration, a successful test was made in Grass Valley, Nevada, of a prototype frequency-domain EM system (Jain and Morrison, 1976; Jain, 1978). These experiments showed that the EM soundings gave interpreted results that compared well with those from dipole-dipole dc resistivity surveys. Based on the need for continued development and demonstration of a field-worthy system (Ward, 1978), and supported through the Department of Energy/Division of Geothermal Energy's Exploration Technology Program, LBL and U.C. Berkeley have developed the EM-60 system, the number related to the 60 kW output of the motor generator used.

The system, easily expandable to include time-domain measurements, is designed for use with a large moment, horizontal-coil transmitting antenna. This choice was based on the need to overcome a number of problems encountered in dc resistivity, MT and existing controlled-source EM systems:

(1) Because no ground contact is needed, the system is better suited to areas where the contact resistances are high, such as sand-covered desert regions or talus slopes on mountains.

(2) A magnetic field detector can be used, thus eliminating the need for long wires, other than the transmitter coil, to be laid out and retrieved.

(3) The transmitter can be installed at a convenient location, an especially helpful feature in terrain where access is limited, and a survey around the transmitter site is conducted by moving the receiver only.

(4) Vertical resistivity soundings are made by varying frequency, not transmitter-receiver separation as in dc resistivity, thus avoiding interpretational difficulties introduced by lateral inhomogeneities.

(5) By generating an EM field over a broad frequency range (10^3 Hz to as low as 10^{-3} Hz), the sounding curves provide both good resolution of the near-surface as well as depth penetration to basement.

(6) The system would not depend on natural field activity, and would therefore provide reliable data in bands where the absence of natural signal often leads to incomplete MT data.

Despite considerable interest in higher frequency EM techniques for mineral exploration throughout much of the world, and for low-frequency EM techniques for petroleum exploration in Russia (Vanyan, 1967; Smith, 1963), surprisingly little work on EM soundings have been done in western countries. Compared to the rapid technological advances in seismic reflection, for example, developments in EM techniques have been slow. The difficulty in interpreting EM results even for simple geological settings, problems in generating and measuring the low-frequency magnetic fields, and field problems associated with laying out and retrieving long heavy wires, have discouraged efforts to employ EM techniques, even in areas where seismic and other techniques are not useful.

This report, divided into three sections describing the transmitter, the receiver and data interpretations, should show that we have made significant technical advances toward the development of a large moment EM system employing a magnetic dipole source. Hopefully, the system will have practical application in geothermal and other surveys.

THE EM-60 TRANSMITTER

LIST OF FIGURES

This section gives a brief description of the EM-60 transmitter, its general design and the considerations involved in the selection of a practical coil size and weight for routine field operations. The transmitter was designed with several criteria in mind:

(a) The system should provide a large magnetic moment, greater than 10^6 MKS at low frequencies,

(b) The system must operate reliably under adverse field conditions with a small field crew,

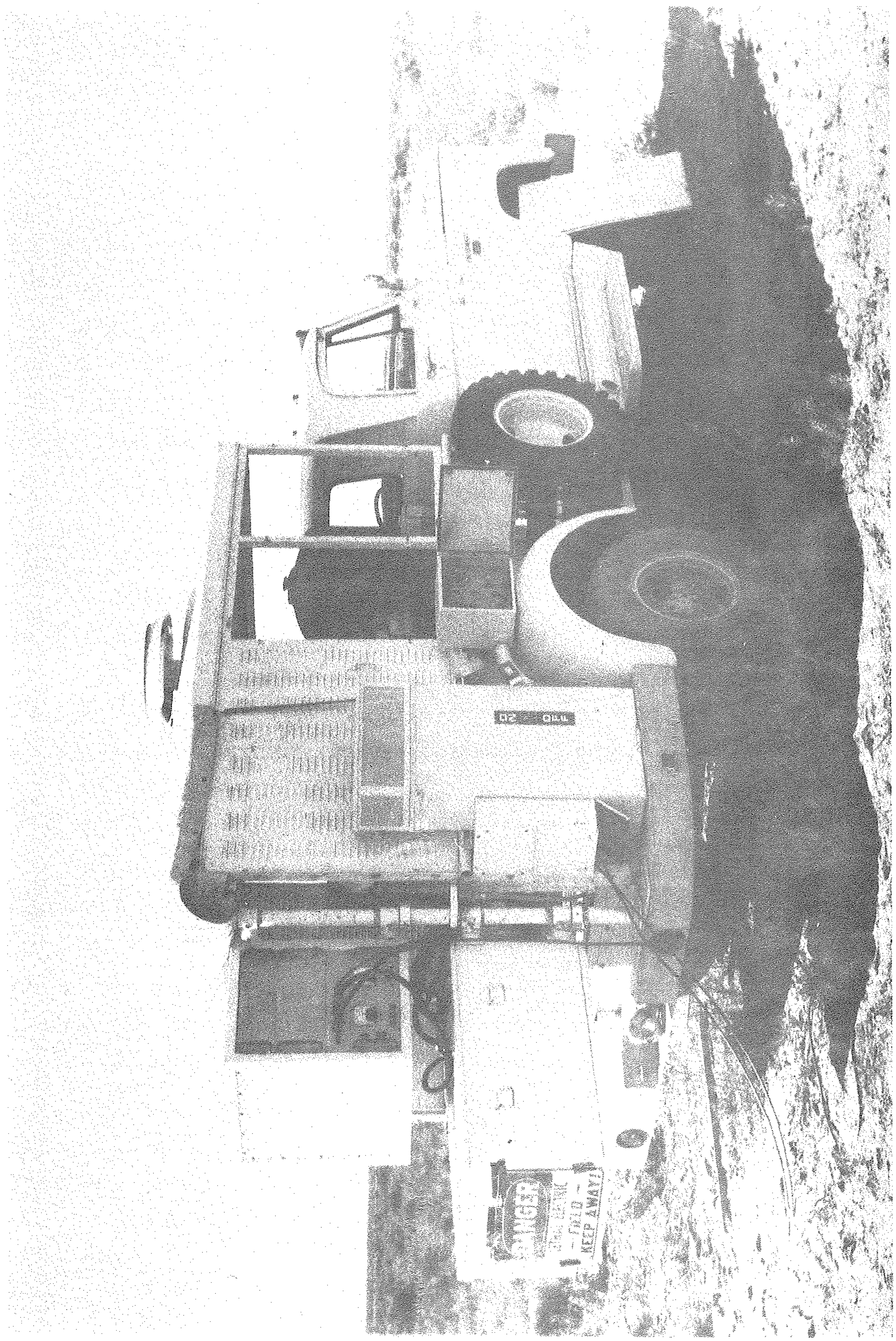
(c) The system must be both safe and easy to operate; and

(d) The system should be relatively inexpensive so that copies or similar systems may be replicated at a reasonable cost.

Except for the last point, for which we have no basis for judgment or comparison, all the criteria seem to have been met. The transmitter is operated by one man; however, laying out and retrieving the horizontal loop antenna requires a larger crew, the exact number of which would depend on loop weight, geometry and terrain, etc. Electronic schematics and mechanical drawings are not presented here, but are available. The key design feature of the transmitter is the transistorized switching arrays which permit rapid switching of large currents into the loop.

The Transmitter

The EM-60 system is powered by a Hercules gasoline engine linked to an aircraft 60kW, 400 Hz, 3ϕ alternator. These two components form the motor generator (MG) set, and are mounted in the back of a Dodge one-ton-chassis, four-wheel-drive truck (Figure 1). Truck and motor-generator set were selected, in part, on the basis of availability of these components at LBL. The output is full-wave rectified and capable



(XBL 789-12515)

Figure 1. The EM-60 in operation in Grass Valley, Nevada.

of providing ±150 volts, 400 amp to an external load which is a horizontal coil (Figure 2). The block diagram, Figure 2, shows that the direction of current flow through the coil is controlled by one of two transistors. These are actually parallel arrays of 6 to 60 transistors mounted modularly in a box called the "crate" (Figure 3). The crate also houses the full-wave rectifier.

Transistor modules are interchangeable, each consisting of a heat sink and fan to enhance heat dissipation. With 18 to 20 of the modules in place, up to 400 amp may be delivered to the coil. Above the crate is the electronics rack. This houses the amplifiers used to control the transistors in the crate (Figure 4). During travel and storage, crate and electronics boxes are carried internally, protected by a snug-fitting cover attached to the rear of the truck. During operations both are swung away for cooling and easier access.

Separate from the transmitter truck, but connected to it by cable, is the remote control box (Figure 5). This contains a crystal-controlled oscillator and dividers, so that a fundamental period of from 10^{-3} to 10^{3} can be selected. On the panel of the control unit are range and thumb-wheel switches for selecting the fundamental period, as well as controls and indicator lights for the transmitter. The remote box may be taken 100-150 feet from the transmitter truck where the motor-generator noise level is lower. It was found, however, that the noise level drops off rapidly away from the truck, even when the louvered side panels are removed.

The operating frequency, f_o, the inverse of the selected period, is amplified at the truck and used to turn the switching transistors on and off via the array driver chassis. Since isolation between the load voltages and the truck chassis is desirable, optical couplers link the array driver to the control signals. For the same reason, separate floating voltages are provided for the array driver. The crate controller links the truck to the remote box and houses the control electronics. These chassis are in the upper rack (Figure 4).

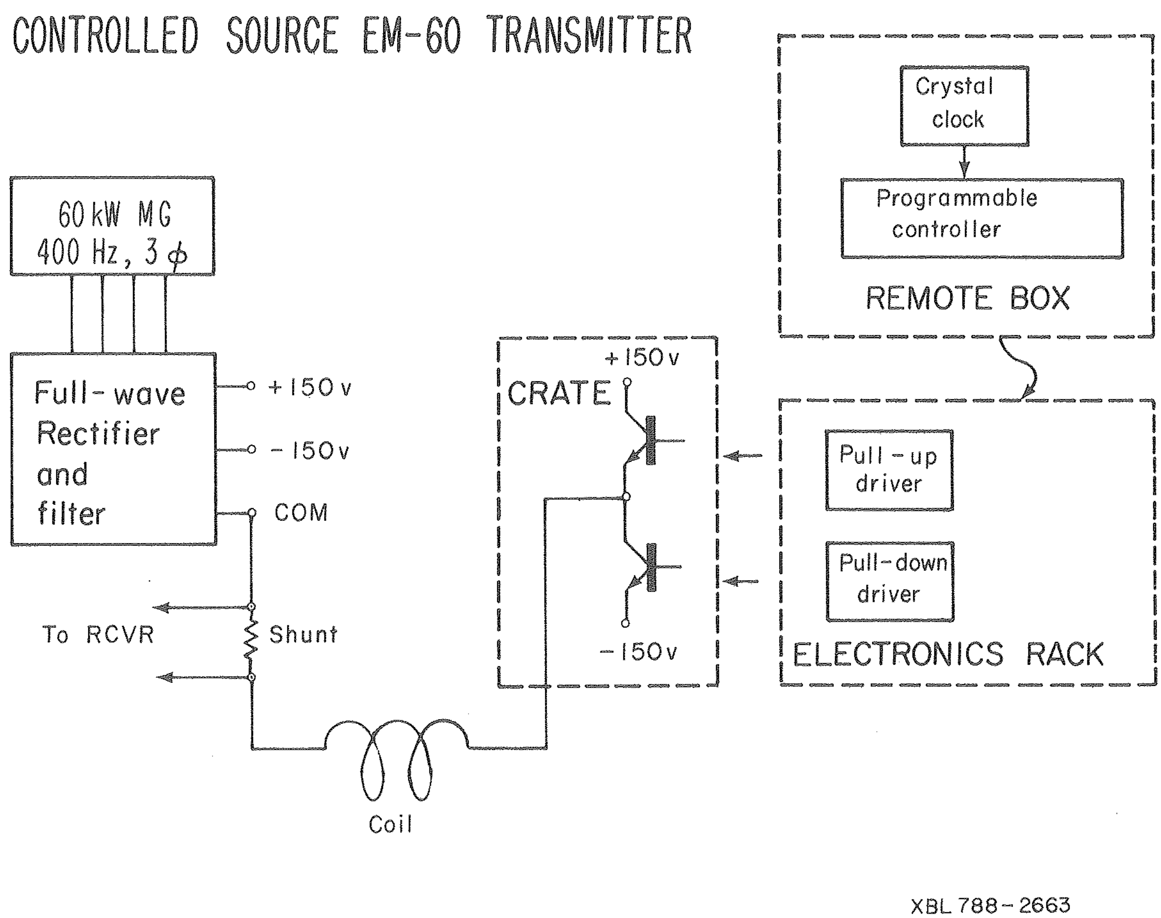
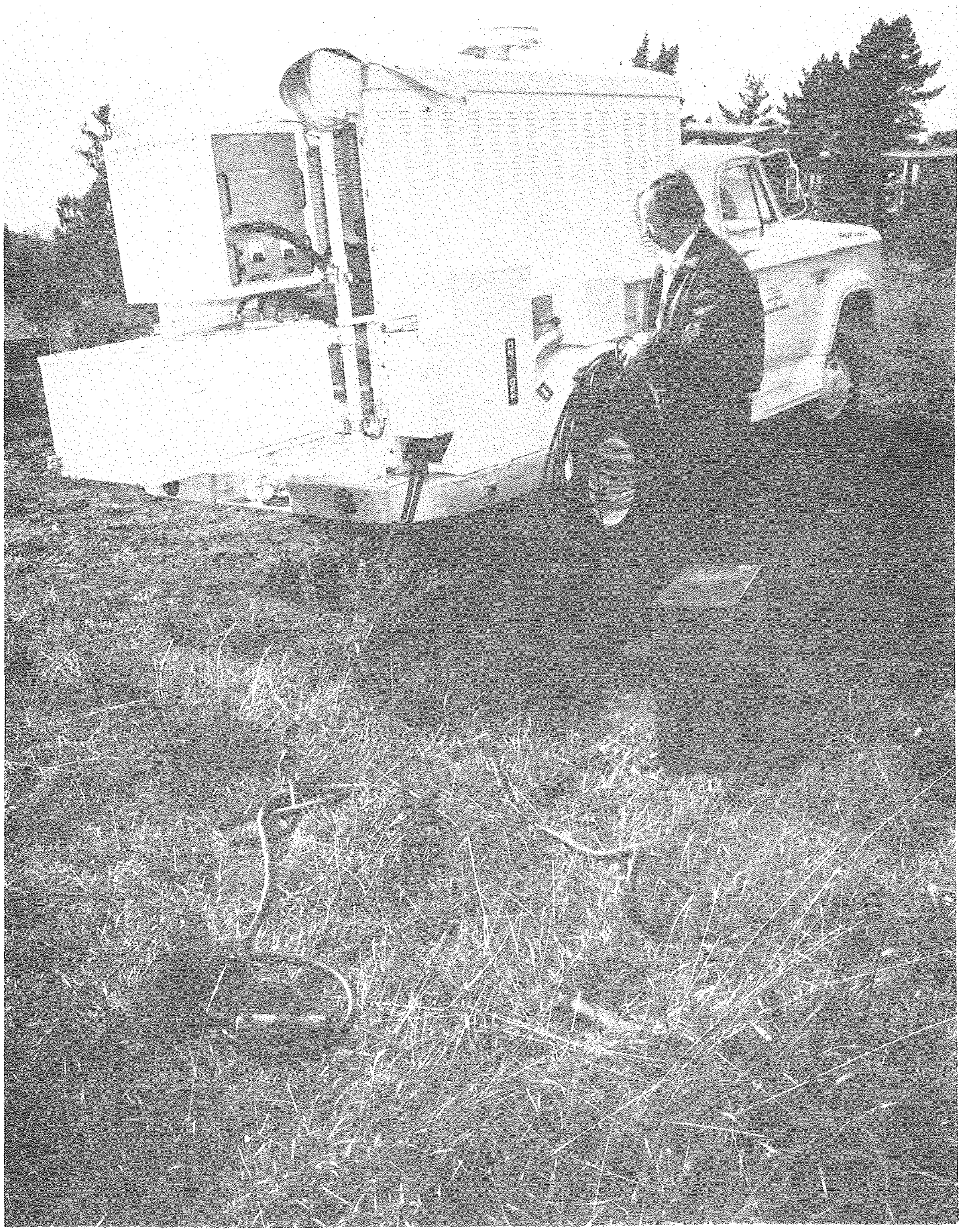


Figure 2. A general block diagram of the EM-60 transmitter section.

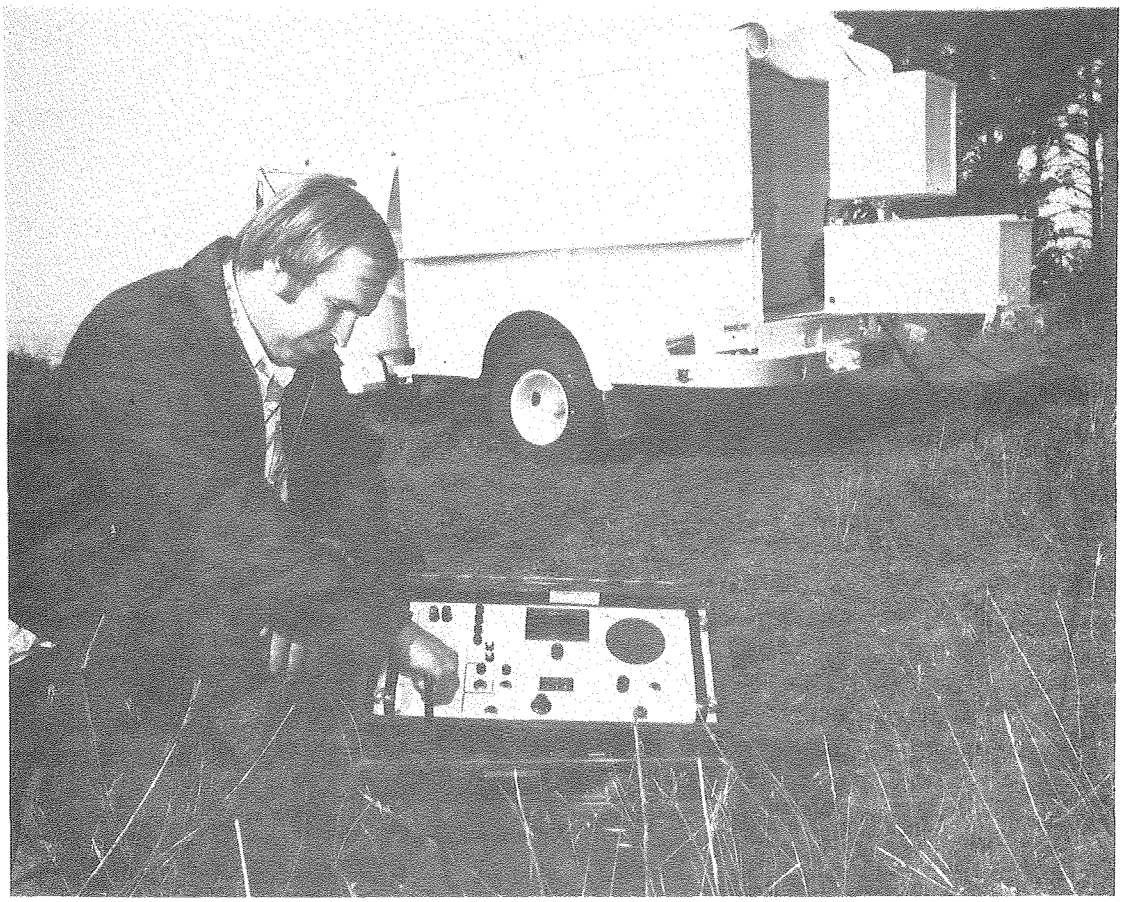
(CBB 788-8381)

Figure 3. The crate with its cover removed, showing the modular arrays of transistor switches and the full-wave rectifier at left.



(CBB 781-953)

Figure 4. Rear of transmitter truck showing the electronics box (top) and crate (bottom) swung out for operations. A 4/0 cable is being attached for tests.



(CBB 781-957)

Figure 5. Fundamental period is set at the remote control-box which also monitors transmitter operations.

Magnetic Dipole Moment

For electromagnetic surveys it is usual to desire the largest moment, M, practical or possible. By definition:

$$M = NIA \tag{1}$$

where

N = number of turns,
I = current in amperes, and
A = coil area in meters^2.

The current, I, defined by Ohm's Law is:

$$I = \frac{V}{Z}\ , \tag{2}$$

and depends on the voltage, V, from the rectifiers and the impedance, Z, of the loop.

The EM-60 is a square-wave voltage generator, switching between +150 and -150 volts. At low frequencies, the inductive nature of the coil can be ignored and the moment can be given as

$$M = \frac{NVA}{R}\ , \tag{3}$$

where V is ±150 volts, and R is the resistance of the coil.

Coil resistance is given as:

$$R = \rho \ell \tag{4}$$

where ρ is the resistance per unit length of wire, and ℓ is the total length of wire in the coil. For a circular coil the area, A, may be expressed in terms of the coil length as

$$A = \frac{\ell^2}{4\pi N^2}\ . \tag{5}$$

Substituting equations 4 and 5 into equation 3, the dipole moment is given in terms of the wire parameters useful in planning a field survey,

$$M = \frac{V\ell}{4\pi\rho N} \tag{6}$$

This shows that the maximum moment from a given length of wire is produced using only one turn. However, in many field situations, terrain, vegetation, and/or water may dictate the use of smaller area, multi-turn coils. Also important in surveys are the total weight of wire that can be brought into the field and the amount of current that may safely be carried through the wire. Higher currents than those recommended can sometimes be used so long as the heating effects do not pose a fire hazard or create other problems; e.g., a hot wire melting into ice would be difficult to retrieve.

For several wire sizes that have been used or considered for use with the EM-60, we list in Table 1 the minimum wire length considered safe. For these lengths heating is only slightly detectable by hand. Table 1 also shows the corresponding weights for the minimum lengths. Initially, we contemplated using a 4/0 welding cable to realize the 400 400 amp capability of the EM-60. This would require laying out and retrieving at least 4 km of cable weighing 4000 kg, not an insignificant task for men and machines. Because LBL does not have field equipment to handle cable of this length and weight, and because the parameters are antithetical to a cost-effective exploration method, field tests and surveys have been conducted with shorter lengths of the smaller #10 and #6 cables. Therefore, the EM-60 has been operated well below its full capability, delivering typically ±63 amperes to the coil.

Field Tests

The EM-60 was given its first full-scale field test in Grass Valley, Nevada during July 1978. The site was chosen because previous electrical and electromagnetic surveys along established geophysical lines had provided us with a subsurface electrical model against which the EM-60 results could be compared. The terrain is flat and open, making loop

TABLE 1

DESIGN CONSIDERATIONS FOR THE EM-60 TRANSMITTER COIL

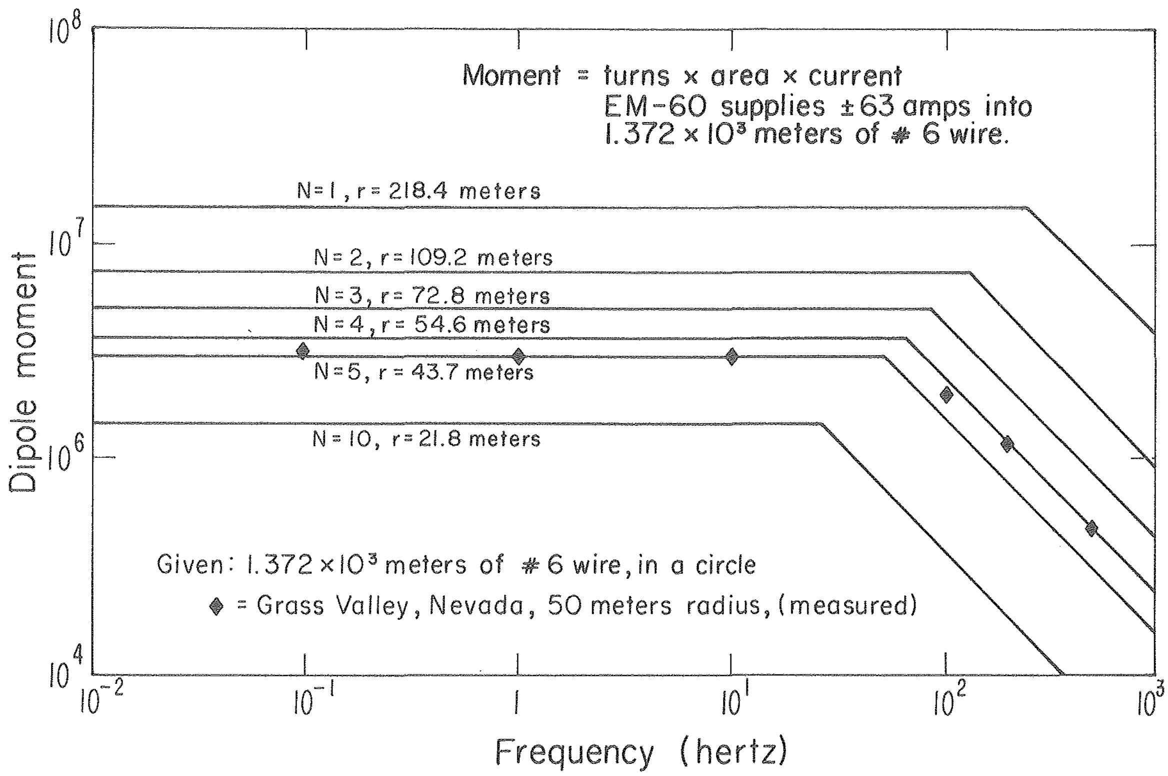
	WIRE SIZE			
PARAMETER	10	6	2	4/0
Wire resistance/km (/km)	3.28	1.30	.513	.160
Weight (kg/km)	49	118	299	955
Minimum coil length to prevent excessive heating (km)	1.8	2.3	3.3	4.2
Minimum coil weight(1) to prevent excessive heating (kg)	86	273	972	4000
Current carrying capacity of minimum length cable (amps)	25	50	90	225

(1) Weight does not include weight of insulation.

handling easy. The coil used consisted of four turns of #6 wire, 100m in diameter and 1372m in total length. The 115m of cable not used in the loop provided pigtails to the transmitter truck. Figure 6 shows a comparison of calculated dipole moments for various turn-area combinations and the measured moment for the coil used in Grass Valley. The dipole moments are calculated on the basis of 126 amp peak-to-peak delivered to the coils at low frequency. Depending on coil diameter and number of turns, a cut-off frequency exists above which the dipole moment declines because of the inductive reactance. In practice the measured dipole moment did not quite follow the theoretical curves above the cut-off frequency. This is because the load, due to its reactive nature, caused the motor-generator to labor less at higher frequencies, thus increasing the effective power input to the loop.

Current in the coil was monitored by means of a 0.01Ω, 0.1 percent shunt resistor. This shunt also provided the reference voltage carried to channel 1 of the receiver by means of a twisted pair of wires. The reference voltage served as the current amplitude and phase reference at the microprocessor-based receiver described in Section II of this report.

Except for refueling operations, the transmitter operated continuously and without failure during the five days of field operations. During this time the ambient temperature exceeded 42°C in the shade, and the longest continuous run was nine hours.



XBL 788-2664A

Figure 6. Theoretical and observed dipole moments over the 10^{-3} to 10^3 Hz frequency range for a circular loop of #6 cable.

A MICROCOMPUTER-BASED RECEIVER FOR THE EM SYSTEM

LIST OF FIGURES

LIST OF TABLES

A Receiver for an EM Prospecting System

This section describes a programmable, multichannel, multi-frequency, phase-sensitive receiver. The receiver was designed and built in the Engineering Geoscience Group, Department of Materials Science and Mineral Engineering, U.C. Berkeley, as part of a Lawrence Berkeley Laboratory project to develop for geothermal exploration, a large-moment electromagnetic prospecting system. The transmitter for this system, described in the previous section, consists of a 60-kw motor-generator, power-transister switching circuitry and a horizontal loop antenna. At a receiver station the magnetic fields are detected by means of a 3-component SQUID magnetometer. The signals are then conditioned by a set of amplifier-filters, and processed by the microcomputer-controlled frequency-domain receiver (Figures 1 and 2). The electric field components may also be detected and processed if so desired. Field tests at Grass Valley, Nevada, described in the following section, showed the system capable of obtaining well-defined sounding curves (amplitude and phase of magnetic fields) from 1 kHz down to 0.1 Hz. Transmitter-receiver separations of 1 to 2 km were used with transmitter moments of about 2×10^6 MKS.

Measurements at frequencies below 0.1 Hz were made, but the statistical error grows larger with decreasing frequency because of the rapidly worsening signal-to-noise ratio. The noise is geomagnetic fluctuations, the spectrum of which varies approximately as 1/f. Although the receiver is designed for frequencies to 10^{-3} Hz (1000 second period), obtaining useful information below 10^{-1} Hz would depend on longer averaging times and larger primary fields than used during tests. Low frequency information might also be obtained by means of a magnetic gradiometer detector to cancel common geomagnetic noise.

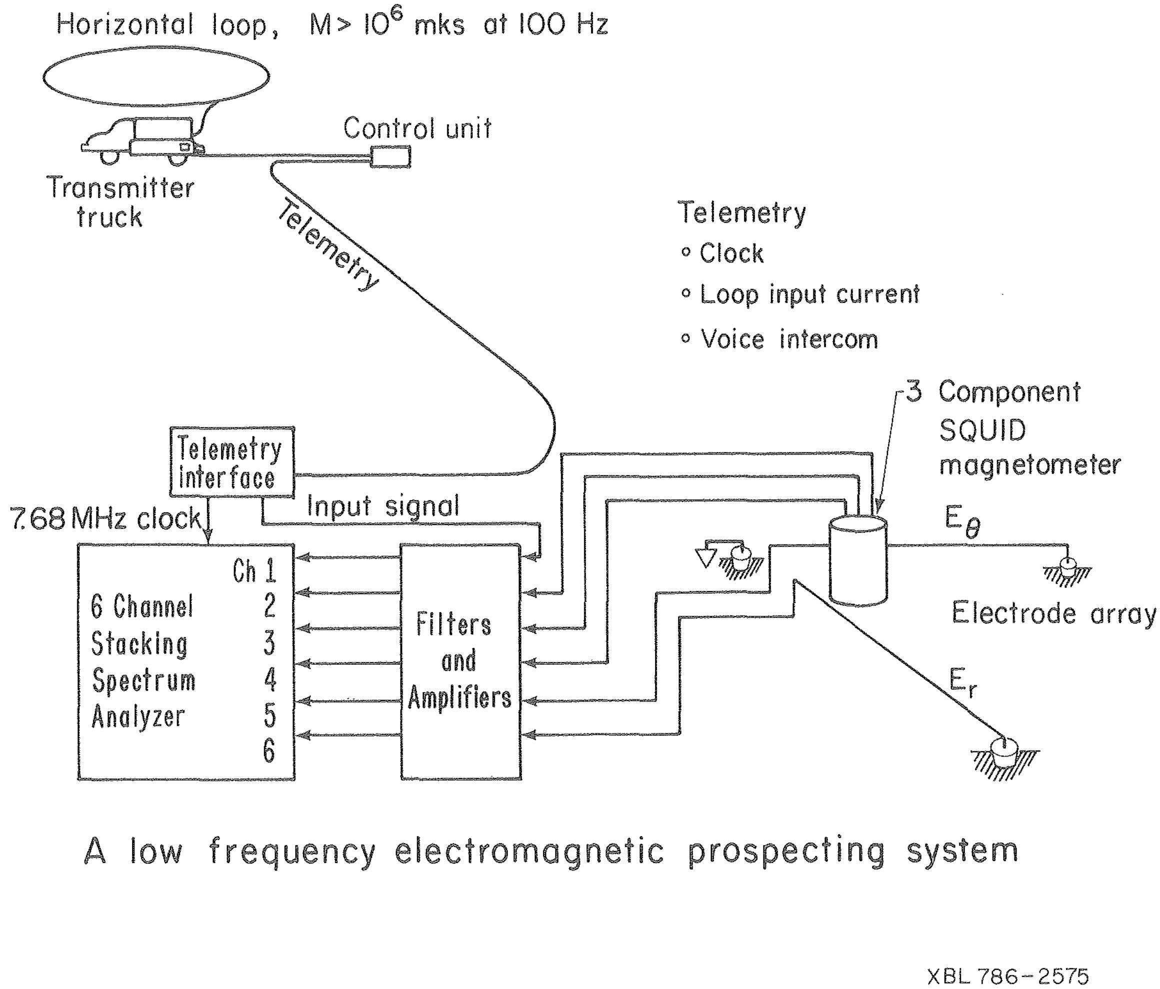
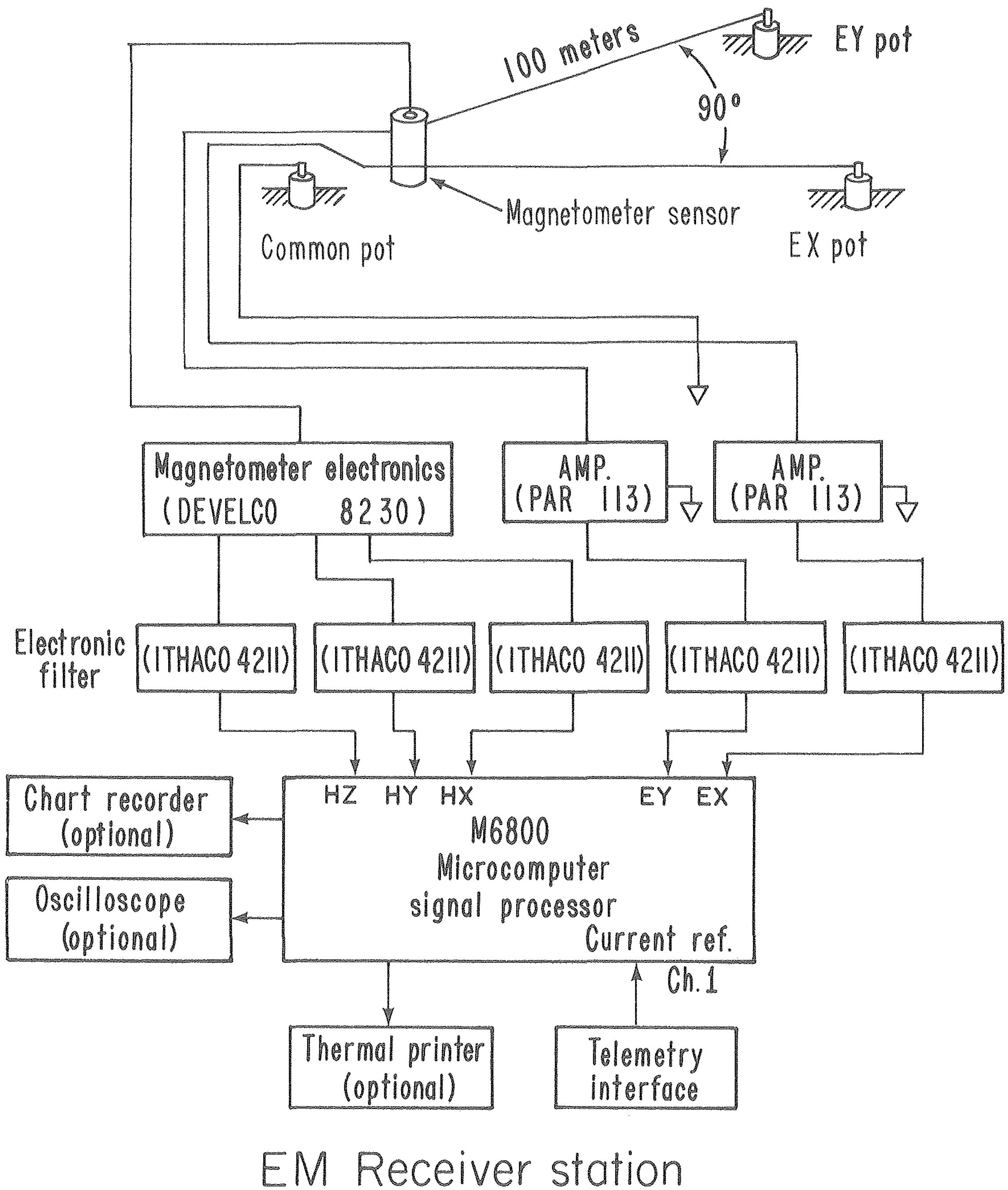


Figure 1. A low frequency electromagnetic prospecting system.



XBL 786-2576

Figure 2. EM receiver station.

The field tests showed it was practical to analyze the first 4 odd harmonics (1, 3, 5 and 7) in the transmitted square-wave at one time. This reduces the number of frequency changes at the transmitter to only one per decade for frequencies below 100 Hz. In tests, overlapping spectral estimates were obtained from closely-spaced fundamental and harmonic frequencies, and the comparisons were good.

Through the keypad the operator is able to set the parameters controlling the signal processing, such as:

a) Period of the fundamental current waveform.

b) The maximum number of odd harmonics of the waveform, up to 16, to be measured.

c) The number of cycles of the signal to be averaged prior to Fourier decomposition.

d) The number of input channels (up to 6; e.g., 3 magnetic and 2 telluric and a reference from the transmitter).

Amplitude and phase information at each harmonic can be displayed sequentially on the receiver's five-digit LCD (Figure 3). However, it is more efficient to record the data on the optional six-column thermal printer. Table I gives an example of the thermal printer output format. In addition, the operator may call routines which display the stored wave forms on an oscilloscope, or sequentially dump them to a chart recorder. Descriptions and examples of chart recorder and oscilloscope displays are given in Table 2.

Table 3 lists the receiver's basic specifications and special features.

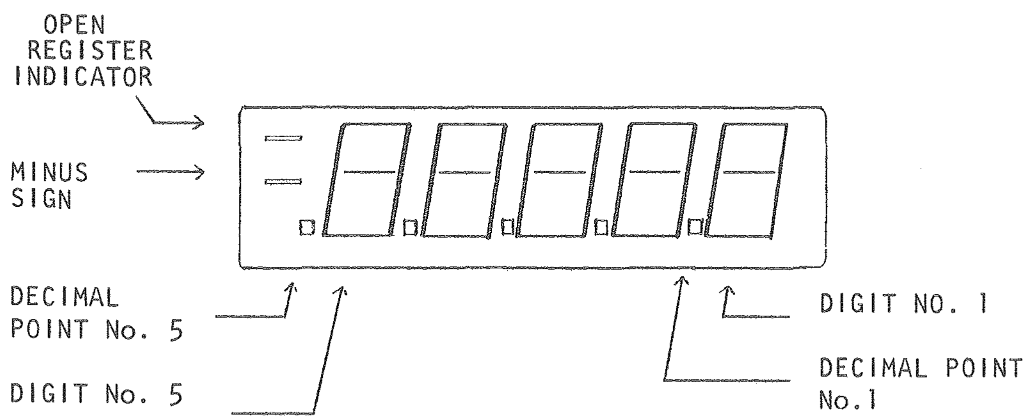


Figure 3. Liquid Crystal Display Format.

TABLE 1

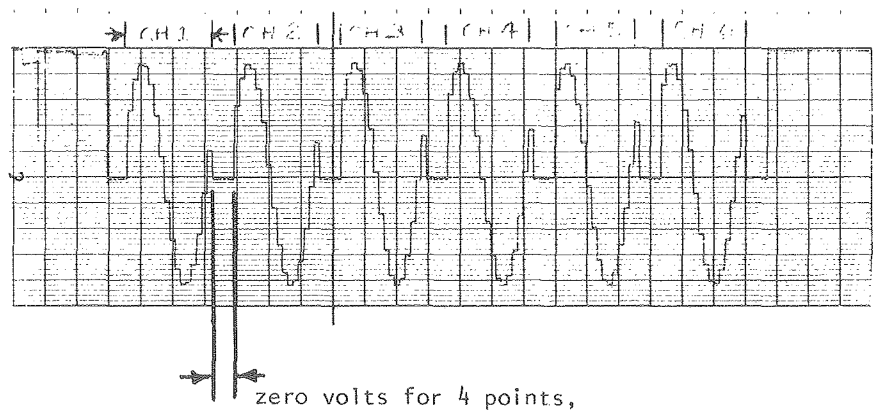
THERMAL PRINTER OUTPUT FORMAT

		00	STATION NO.
		00 17	RUN NO.
		10.	NO. OF CYCLES AVERAGED (EXPONENT OF 2)
		1.0 1	PERIOD IN MILLISECONDS
		01.	HARMONIC NO.
CH 1		3849.3	AMPLITUDE IN MILLIVOLTS OR REAL[1]
		2.9095	PHASE IN DEGREES OR IMAGINARY[1]
CH 2		3849.0	AMP or REAL
	--	.0010	PHASE or IMAG
CH 3		3849.0	AMP or REAL
	--	.0022	PHASE or IMAG
CH 4		3849.0	AMP or REAL
	--	.0033	PHASE or IMAG
CH 5		3849.3	AMP or REAL
	--	.0044	PHASE or IMAG
CH 6		3849.6	AMP or REAL
	--	.0056	PHASE or IMAG

1. When the rectangular mode is used, the in-phase or real part replaces amplitude and the quadrature or imaginary part replaces phase. Note that the real and imagniary parts are not scaled by the constant 1.19266 to obtain millivolts and are not phase corrected for the sampling skew. See Table 10 key no. 1.

TABLE 2

CHART RECORDER AND OSCILLOSCOPE DISPLAY



zero volts for 4 points,
functions to separate channels.

Example of a chart recorder dump of six channels of a digitally stored sinewave at 16 points per cycle. Paper moved to left.

CHART AND SCOPE DISPLAY

Output voltage range	+/- 5 volts
Digital to analog conversion resolution	8 bits (39.6mvolts/bit)

CHART RECORDER DISPLAY

Points per cycle	Time required for dump
64	14 seconds
16	4.2 seconds
4	1.8 seconds

OSCILLOSCOPE DISPLAY

Points per cycle	Refresh rate	Display time window
64	45 HZ	22 m sec
16	150 HZ	6.7 m sec
4	350 HZ	2.9 m sec

TABLE 3

M6800 MICROCOMPUTER SIGNAL PROCESSOR

PROGRAMMED AS A

SIX CHANNEL WAVEFORM STACKING, HARMONIC ANALYZER

FREQUENCY RANGE	1.01×10^{-3}Hz to 1.0 KHz (990 sec to 1.0 msec)
PHASE ACCURACY	Better than 0.05 degrees
NUMBER OF CYCLES AVERAGED	Up to 2^{15} cycles
NUMBER OF POINTS SAMPLED PER CYCLE PER CHANNEL	1.0 KHz to 101 Hz: 4 pts/cycle 100 Hz to 13 Hz: 16 pts/cycle 12.5 Hz to 0.00101 Hz: 64 pts/cycle
NUMBER OF HARMONICS	Up to 32, 8, or 2 harmonics for 64, 16, or 4 pts cycle, respectively
ANALOG INPUTS	
CONFIGURATION	Six single-ended or differential channels
INPUT VOLTAGE	$\pm$ 5V signal voltage
ANALOG TO DIGITAL CONVERSION RESOLUTION	12 bits binary
SYNCHRONIZATION SIGNAL	7.68 MHz, TTL internal or external, switch selectable
PHASE REFERENCE	Phases of harmonics in the channel 1 waveform serve as phase references for channels 2 to 6
DETECTION ALGORITHM	16 and 4 pts/cycle (8 bit data resolution) i. acquire 8 cycles of data ii. stack data, repeat i iii. sine and cosine transform stacked data 16 and 64 pts/cycle (12 bit data resolution) i. acquire and stack data continuously ii. sine and cosine and transform stacked data
QUANTITIES OUTPUT	Amplitudes, phases, number of cycles averaged, harmonic number, period of fundamental, station no., and run no.
DATA OUTPUT FORM	5 digit LCD and 6 column thermal printer
POWER CONSUMPTION	10-15 watts
INTERNAL BATTERY LIFE	8-10 hours continuous
SIZE AND WEIGHT	9 x 16 x 16 inches, 35 lbs.

An Adaptable Receiver Design

Although the instrument was designed as a specialized receiver for a particular EM system, it has a general structure adaptable to many signal processing tasks in geophysics. Perhaps the most important feature of this structure is programmability. This feature allows one to modify the function of the instrument through a programming change rather than by time-consuming hardware modifications.

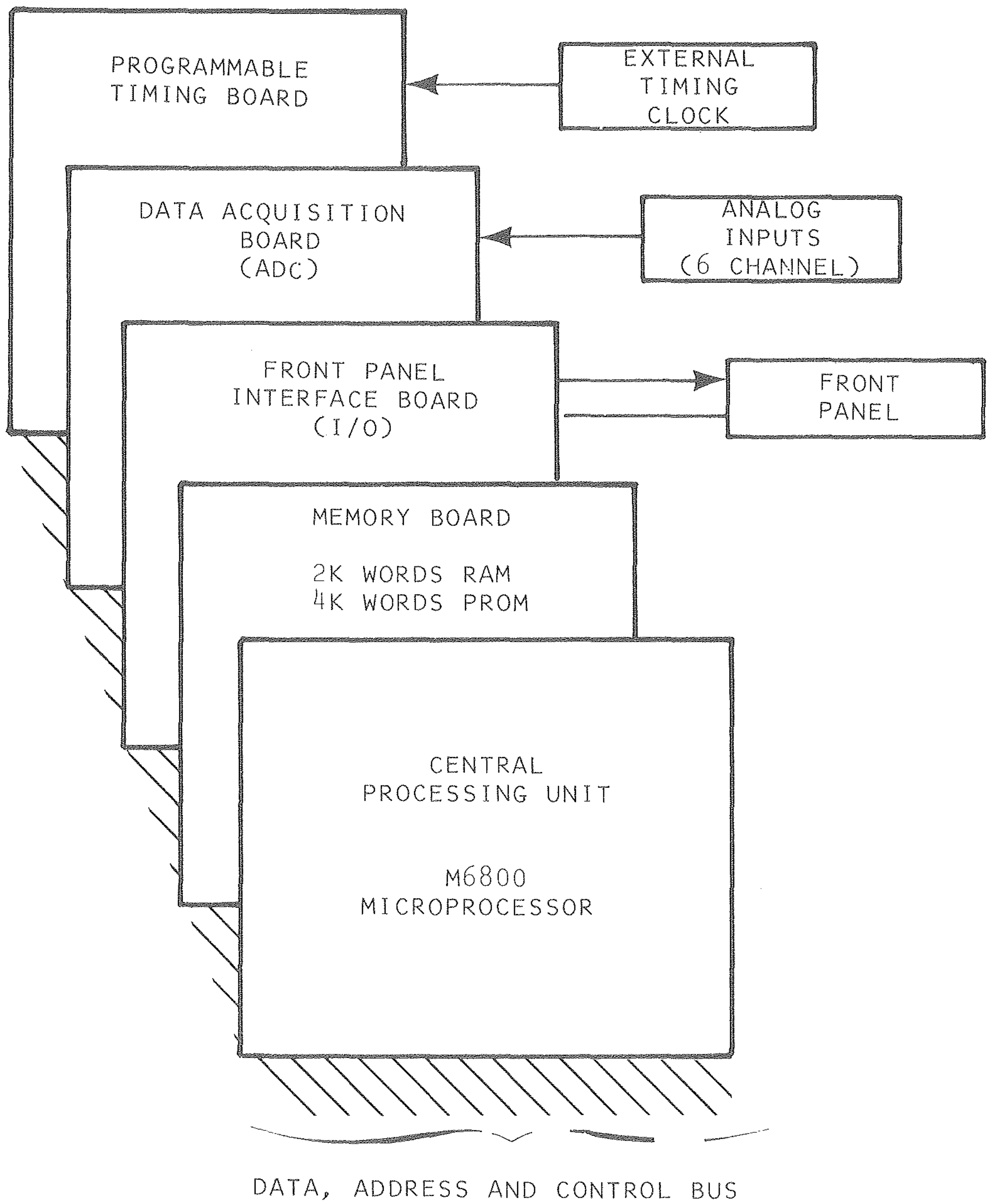
The receiver's hardware is also designed for flexibility. The hardware is organized around a backplane bus containing the address, data, and control lines for the microcomputer (Figure 4 and Table 4). The chassis has eight circuit-card slots connected to this bus. Five slots are used in the present system; the other three may be used for system expansion; e.g., additional memory, special control boards, or analog filter circuits.

These features, combined with the instrument's calculator-like operation and portability, make it extremely promising as a general purpose receiver for exploration geophysics.

At present, a time-domain EM program is under development for the receiver, and other program additions are being considered.

Simple Operation

The receiver is simple to operate. When power is turned on or the reset switch is activated, the receiver automatically performs self-test routines and initializes all control parameters to bring itself to an operational state (Table 5). The program automatically selects the optimal number of points per cycle and analog to digital conversion word sizes for data acquisition operations (Table 6). Although the user has the option to select several control switches and parameters to increase the efficiency of the signal processing operations, the operator is required only to set the period and the number of cycles of waveform to be averaged, and call a signal processing routine. Table 7 gives a list of the steps the operator follows. These procedures will be discussed in more detial in following sections.



XBL786-2578

Figure 4. System hardware structure for M6800 microcomputer signal processor.

TABLE 4

M6800 SIGNAL PROCESSOR BACKPLANE

PIN NO.	NAME	PIN NO.	NAME
1	-15^V	29	D4
2	-15^V	30	D5
3	$+15^V$	31	D6
4	$+15^V$	32	D7
5	0^V	33	
6	0^V	34	
7	$+5^V$	35	
8	$+5^V$	36	
9	AØ (LSB)	37	
10	A1	38	
11	A2	39	$\overline{\text{NMI}}$
12	A3	40	$\overline{\text{IRQ}}$
13	A4	41	Ø2
14	A5	42	VMA
15	A6	43	R/W
16	A7	44	Ø1
17	A8	45	$\overline{\text{RST}}$
18	A9	46	$\overline{\text{HALT}}$
19	A10	47	
20	A11	48	300 BAUD (x16)
21	A12	49	PAGE ØØ
22	A13	50	PAGE FF
23	A14	51	$+5^V$
24	A15	52	$+5^V$
25	DØ (LSB)	53	0^V
26	D1	54	0^V
27	D2	55	-9^V
28	D3	56	-9^V

TABLE 5

M6800 MICROCOMPUTER SIGNAL PROCESSOR

SPECIAL FEATURES

I. VERSATILE CONTROL ROUTINES ALLOW AUTOMATIC CALCULATION AND PRINTOUT OF SINGLE OR SELECTED GROUPS OF HARMONICS (E.G. ODD HARMONICS, 15TH THROUGH 1ST).

II. CHART RECORDER DUMP OF STORED SIGNALS.

III. OSCILLOSCOPE DISPLAY OF STORED SIGNALS.

IV. VOLT METER FUNCTION - DISPLAYS VOLTAGE ON SELECTED CHANNEL IN MILLIVOLTS. THIS ROUTINE IS USED TO SET GAIN LEVELS. THE SYSTEM SUPPLY VOLTAGE MAY BE CHECKED BY EXAMINING THE VOLTAGE ON CHANNEL 7.

V. MAXIMUM VALUE FUNCTION - FINDS MAXIMUM VALUE ON EACH OF THE STORED WAVEFORMS AND DISPLAYS VALUE IN MILLIVOLTS.

VI. AUDIO TRANSDUCER ALERTS OPERATOR TO COMPLETION OF LONG SIGNAL AVERAGING OPERATIONS.

VII. AUTOMATIC SYSTEM TEST ROUTINES:

A) TESTS 2K OF DATA STORAGE MEMORY (RAM) IDENTIFYING ANY DEFECTIVE MEMORY CHIP (16 OF THESE).

B) TESTS 4K OF PROGRAM STORAGE MEMORY (ROM) IDENTIFYING DEFECTIVE ROM CHIP (4 OF THESE).

C) TESTS FOR PRESENCE OF TIMING SIGNALS: CHECKS RATIOS OF SAMPLE TO CYCLE PULSES AT EACH OF 4, 16, AND 64 POINTS PER CYCLE: DISPLAYS ERROR-IDENTIFYING-CODES IF ERRORS ARE DETECTED.

D) WRITES TEST SQUARE WAVE INTO MEMORY FOR CHECK OUT OF TRANSFORM ROUTINES.

E) TESTS LIQUID CRYSTAL DISPLAY AND PRINTER.

TABLE 6

COMPUTER SELECTION OF NO. OF POINTS PER CYCLE AND ANALOG TO DIGITAL CONVERSION WORD SIZE.

POINTS PER CYCLE	PERIOD RANGE	
64	990 sec THROUGH 80 msec	
	(0.00101 HZ)	(12.5 HZ)
16	79 msec THROUGH 10 msec	
	(12.658 HZ)	(100 HZ)
4	9.0 msec THROUGH 1.0 msec	
	(111.11 HZ)	(1.0 KHZ)

ANALOG TO DIGITAL CONVERSION WORD SIZE	PERIOD RANGE	
12 BIT WORDS	990 sec THROUGH 20 sec	
	(0.00101 HZ)	(50 HZ)
8 BIT WORDS	19 msec THROUGH 1.0 msec	
	(52.63 HZ)	(1.0 KHZ)

TABLE 7

RECEIVER OPERATION PROCEDURE

Part 1 Set Control Switches and Connect Cables

- Select internal or external 12 volt power source.
- Turn receiver power on.
- Select internal or external SYNC (7.68 MHZ).
- Put run/load switch in run position.
- Place mode switches in selected positions.
- Press test switch on printer, before connecting printer.
- Connect printer, turn printer power on.
- Connect input and output cables.

Part 2 System Test

- Press reset - system test.
- Look for error codes and examine checksums.
  (see section on system test)

Part 3 Set Parameters

- Enter Period: Press (PER), (NO.), (NO.), (NO.), (LCK PER)
- Enter Harmonic No.: Press (HRM), (NO.), (NO.), (RTN)
- Enter No. of Cyc. Avg.: Press (NO.CYC), (NO.), (NO.), (RTN)
- Enter No. of Channels: Press (NO.CHL), (NO.), (NO.), (RTN)
- Enter Station No.: Press (MEM), (2), (NO.), (NO.), (RTN)

Part 4 Call System Programs

- Call volt meter routine for each channel and set gains.
  Press (RUN), (VLT), (NO.).
- Call one of the acquisition routines e. g. (RUN), (A1)
- Call scope display routine (RUN), (OSC), (1)
  or call chart display routine (RUN), (CHT)

High Accuracy Phase Measurements

The receiver was designed to make high accuracy phase and amplitude measurements under conditions of low signal-to-noise ratios. High accuracy measurements are particularly important for electromagnetic soundings at frequencies below about 10 Hz, where phase accuracies of 0.1 degree may be required to invert the soundings reliably.

The phase accuracy obtained from a given sinusoidal waveform, excluding aliasing, is a function of the signal or data resolution, the number of points per cycle, and the precision of the transform arithmetic. Figure 5 shows the maximum phase error that can be expected with a given resolution and number of points per cycle. If the signal-to-noise level of the measured signal is known, Figure 5 may be used to estimate the number of times the waveform must be stacked to obtain a given phase accuracy. One may assume $N^{-\frac{1}{2}}$ reduction of noise.

Under favorable signal-to-noise conditions, exceptionally accurate phase measurements may be made. For example, laboratory tests have shown the receiver capable of measuring relative phases with accuracies better than 0.002 degree below 12.5 Hz, 0.006 degree below 100 Hz, and 0.05 degree below 1000 Hz.

The periods of the harmonics in the stacked waveforms correspond exactly to those analyzed by the sine and cosine transform routine, by definition of the harmonic content of a periodic waveform. This precise matching of waveform periods eliminates spectral smearing resulting from the finite data lengths, and makes the high accuracy phase measurements possible.

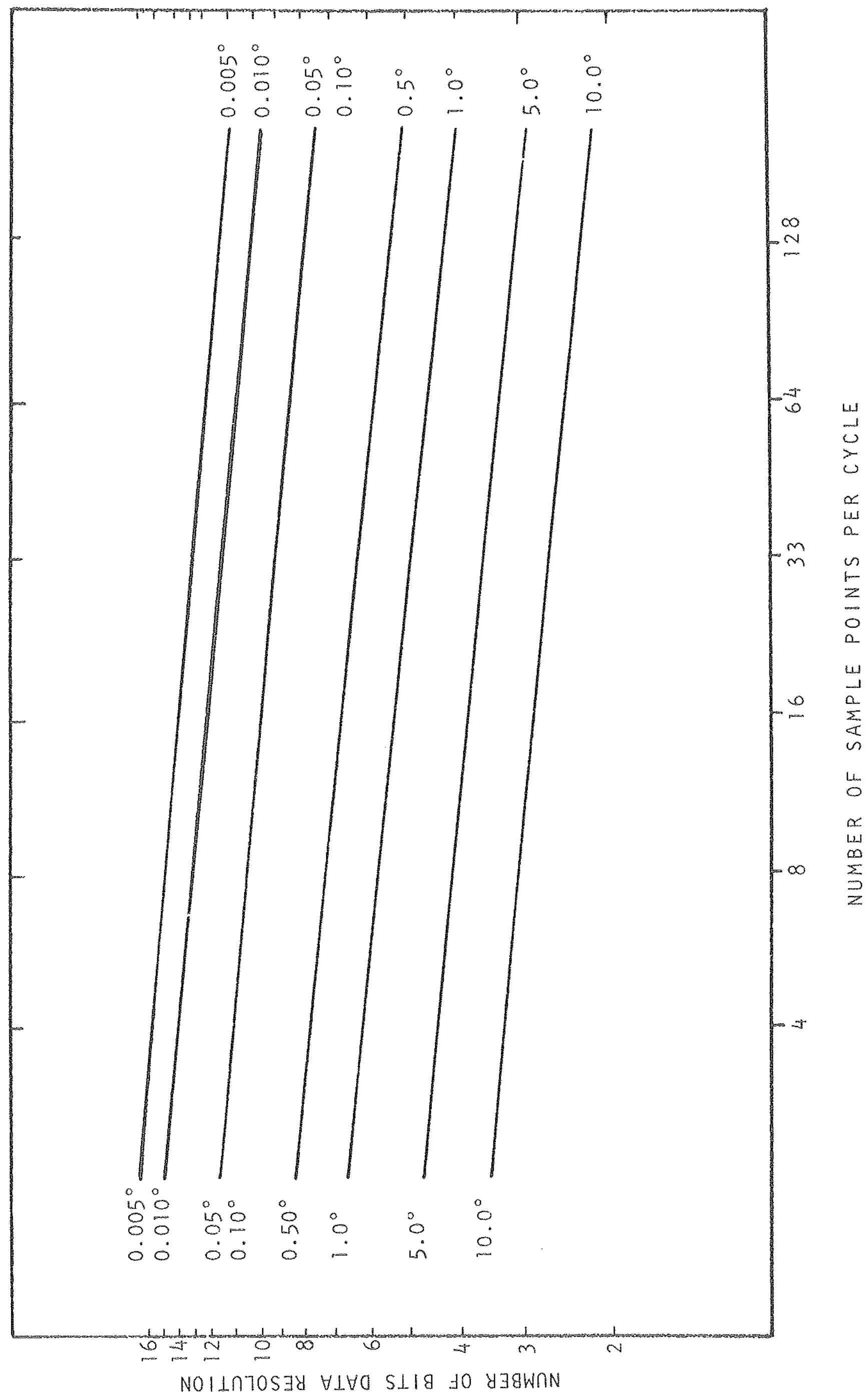


Figure 5. Maximum phase error using 16-bit fixed point constants. Fortran simulation of microcomputer arithmetic.

Relative Phase Measurements

The receiver operates as an independent unit, in the sense that it does not depend on control signals from the transmitter. The transmitter and receiver run asynchronously; each unit is driven by a separate crystal clock, having a frequency accuracy requirement of only 100 ppm. Phase measurements relative to the transmitter current are made by processing the transmitter current waveform (on channel 1) along with the magnetic field signals (on the other channels). The receiver then computes phase relative to the transmitter current by subtracting the calculated phase of the transmitter current from that of the other channels' phases. Any signal can be put on channel 1 to act as the phase reference for the other 5 channels.

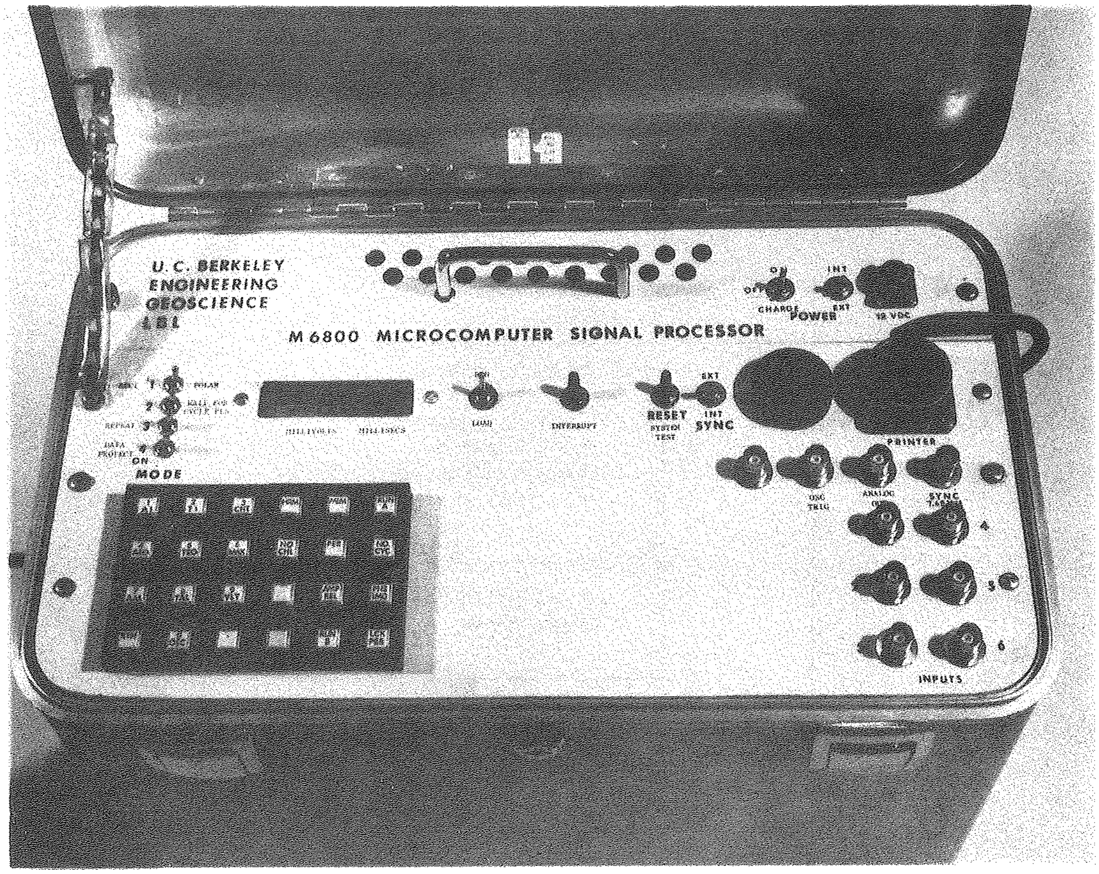
The receiver acts as a narrow-band digital filter. The accuracy of the filter's center frequency is related directly to the accuracy of the clock controlling the signal-sampling circuitry. The sharpness or selectivity of the filter increases with the number of cycles averaged. If the receiver and transmitter clocks are not locked together, the transmitter and receiver will be operating at slightly different frequencies. This difference in frequencies puts a restriction on how sharp the receiver's digital filter may be made before the transmitted signal begins to be filtered out. Using clocks of 10 ppm accuracy, several thousand cycles of transmitter signal may be averaged with no detrimental effects due to filter selectivity.

It is also possible to lock the transmitter and receiver clocks together through the external 7.68 MHz clock input on the receiver. This requires telemetering the transmitter clock signal to the receiver but allows unrestricted stacking of the waveforms.

Frequency domain EM soundings can be made with the receiver in two ways. One approach involves the normalization of the phase and amplitude spectra of the magnetic field by the spectra of the transmitter current. This method requires that the voltage across a shunt resistor in the transmitter loop be brought to channel 1 of the receiver via a twisted pair of wires. These wires are the only physical connection between the transmitter and receiver, and provide an absolute phase and amplitude reference for the system. The second approach eliminates the need for a current reference from the transmitter by analyzing phase and amplitude relations between the vertical and horizontal magnetic fields, which define a polarization ellipse. The essential information on earth conductivity structures is contained in EM soundings produced by either the transmitter current reference or polarization ellipse approaches.

SIGNAL PROCESSING

Table 3 lists the principal features of the receiver, which is built around the M6800 microprocessor (Figure 6). The simplified program structure is shown in Figure 7, and the hardware structure is shown in Figure 4. A multichannel, 12-bit analog-to-digital conversion module is used to sequentially sample six channels of electrical signals. The sampled waveforms from each of the channels are stacked in memory to improve the signal-to-noise ratio, then normalized by the number of cycles averaged. The discrete Fourier transform is then obtained using a table of 16-bit sine and cosine constants and a software multiply routine using 16-bit fixed point operands and producing 32-bit products. In-phase and quadrature results from the transform are converted to phase (in degrees) and amplitude (in millivolts) using a CORDIC rotation method. Next, the phase-shift errors introduced by the sequential sampling of the six channels are corrected, and the phase of channel 1 is subtracted from the phases of channels 2 through 6. Thus, phases for signals on channels 2 through 6 are all relative to the phase of the signal on channel 1. The binary results of the processing are converted to BCD and printed out on a thermal printer. Table 1 gives



(CBB 7810-13519)

Figure 6. M6800 microcomputer signal processor.

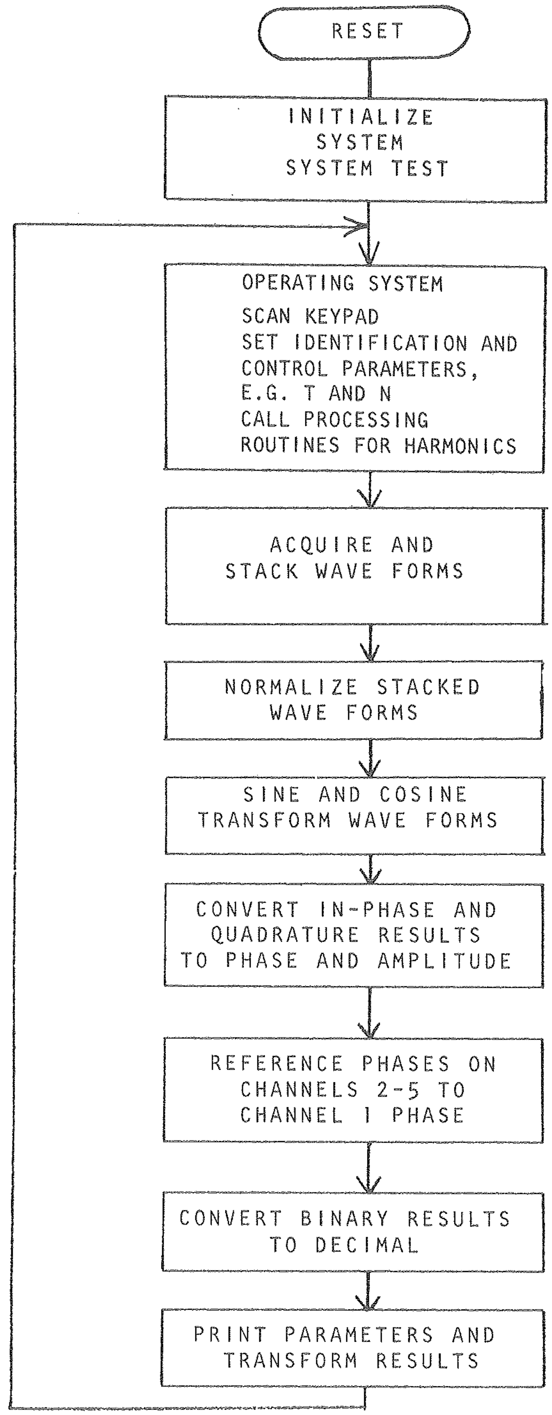


Figure 7. Simplified program structure for M6800 microcomputer signal processor.

an example of the output format from the thermal printer. In this case, a common signal was entered onto all six channels. Amplitudes agree to within ±0.3 millivolts, and the maximum phase error, on channel 6, is 0.0056 degrees (.0977 milliradians).

OPERATIONS

Table 5 lists special features of the receiver. In addition to the keypad accessible signal processing, waveform display and utility routines, the receiver contains a system test routine designed to test vital sections of system hardware. The system test routine is automatically called when the receiver is powered up and each time the reset switch is pressed. The automatic test programs are listed in Table 5, Sections VII A-E.

Systems Programs

The receiver has ten keypad accessible system programs which allow the operator to control the instrument's function. These programs are called by pressing the RUN key followed by the number key corresponding to the selected program. The key symbols and programs are defined in Table 8.

Stored Parameters

Three types of stored values may be accessed from the keypad: (1) operator-set control parameters, e.g., number of cycles averaged; (2) program-set parameters that may be examined by the operator, e.g., number of points per cycles; (3) signal processing results, e.g., phase and amplitude. The more frequently used parameters have been assigned separate control keys for faster access; less frequently used parameters are read by pressing MEM, then the number key associated with the particular parameter. Key symbols and parameter descriptions are given in Tables 9 and 10.

TABLE 8

SYSTEM PROGRAMS

Key Symbol	The RUN key followed by the program key number given on left calls[A] the following programs:
1 A1	Acquires a set of signals at the selected period and averages the selected number of cycles; then transforms and prints a single harmonic defined by HRM.
4 AEO	Acquires signals as the above program. Then transforms and prints every other harmonic beginning with HRM down through the first harmonic.
7 AAL	Acquires signals as above program. Then transforms and prints all harmonics beginning with HRM down through the first harmonic.
2 TI	Transforms and prints one harmonic defined by HRM.
5 TEO	Transforms and prints every other harmonic beginning with HRM down through the first harmonic.
8 TAL	Transforms and prints all harmonics beginning with HRM down through the first harmonic.
6 CHT	Dumps 6 channels of stored signals to chart recorder. See display format note. The keypad is not functional during this dump.
0 OSC	When followed by a number key (1 to 6), displays 6 channels of stored signals on oscilloscope. Scope trigger is positioned in front of channel's data corresponding to previously entered number. Trigger position may be changed any number of times. Exit by pressing RTN. (Refresh rate is 45Hz).
6 MAX	When followed by a number key (1 to 6), for channel number, displays voltage (in millivolts) present on selected channel. The sampling frequency is 6 times the number of points per cycle. Channel seven is internally connected to the +5V supply line. Exit this routine by pressing RTN.
RTN	No program is called. Returns control to operating system.

TABLE 9

NUMERICAL CONTROL PARAMETERS
AND
STORED TRANSFORM RESULTS.

Key Symbol

HRM — Harmonic Number

Range 01 through (PTS/CYCLE)/2. RTN closes location.

NO. CYC — Number of Cycles Averaged

Raises 2 to power entered.

Range: 2^0 through 2^{15} or 1 through 32,768

RTN displays in decimal the number of cycles averaged and closes the location.

PER — Period in milliseconds

e.g. 1.2 3 = 1.2×10^3 msec

No leading zeros; do not enter decimal point (range 9.9 5 through 1.0 0). Period set by following entry of 3 digits by $\frac{\text{LCK}}{\text{PER}}$. RTN results in no change in period and closes location.

NO. CHL — Number of Channels

Range: 1 through 6

Limits number of channels put through polar conversion and printing. RTN closes location.

AMP REL and PHS IMG — Amplitude and Phase or (real and imaginary)

when this key is followed by a number 1 through 6 (Channel No.) display shows value for that channel.

TABLE 10

OTHER STORED PARAMETERS

The MEM key followed by no. key accesses the following:

key no.

0 Points per cycle. 64, #16 or 4. set by program. Function of period.

1 Phase correction. Phase shift due to sampling time skew. Should be subtracted from channel N as (N -1) (PHASE COR) when using rectangular mode. Set by program. Function of harmonic and PTS/CYC.

2 Station number. Operator set. A two digit value with range of 00 through 99.

3 Run number. Operator set and program incremented each time a new set of data is acquired. A two digit value with range of 00 through 99.

4. Phase accuracy control for rectangular to polar conversion. Range 03 to 06. Parameter is initialized to 04.
04 produces 0.014 degree accuracy with a maximum calculation time of 2 sec/channel.
06 produces 0.0035 degree accuracy with a maximum calculation time of 8 sec/channel.

TABLE 11

CONTROL KEYS

Key Symbol	
RUN A	Key calls one of ten programs defined by number keys. (See Table V for the list of system programs.
RTN	Closes open parameter locations; and for system programs OSC, MAX, and VLT. It returns control to the operating system.
LCK PER	Sends entered period to programmable sample timing board.
RUN B	When followed by key no. Ø causes program to jump to next page of memory. This page is optional and user defined. System control or diagnostic programs may be placed on this page to extend the degree of specialization of this system, e.g., an IP program computing percent frequency effect.

TABLE 12

MODE SWITCHES

1 RECT or POLAR

Selects mode in which transformed values will be presented. (Note: Rectangular mode values are not phase corrected for sampling skew and are not scaled by the constant (1.192659) to obtain values in millivolts per root HZ. Polar values have all corrections applied.) This switch is read by program at the end of the SIN-COS transform routine.

2 WAIT FOR CYCLE PULSE

Causes acquisition routines to wait for the beginning of next cycle before starting data acquistion. It is useful when working with periods greater than about 2 seconds, in that when deactivated it eliminates the waiting period before the beginning of the next cycle. This switch is read by the program for periods greater than 20ms (50HZ) only. Acquisition routines with smaller periods always wait for the cycle pulse.

3 REPEAT

Causes any of the three data acquisition routines to repeat their processing and printing operations until the switch is turned off. Also causes the system test routine to be repeatedly called.

4 DATA PROJECT

Prevents accidental overwriting of data sets in memory. The acquisition routines read this switch before acquiring new data sets.

During operations, the numerical control parameters (Part 3, Table 7) must be entered correctly, but not necessarily in a particular order. Table 9 provides detailed descriptions and examples of how the variables must be entered. Table 10 lists other stored parameters. The number of points-per-cycle sampled and phase corrections are set up by the program, but these values may be examined by the operator through the keypad. The phase accuracy control determines the accuracy of the rectangular-to-polar conversion, and is preset automatically during system initialization. The only parameters that the operator may wish to change are the station number and the run number, which are used for data identification on the printer. The run number may be initialized to 0 each time transmission of a new fundamental period begins. After each averaging operation the run number will be automatically incremented.

Control Keys

There are four control keys. Two of these are used to call programs and the other two close memory locations after values have been entered. The control keys are described in Table 11.

Mode and Control Switches

There are four mode switches located in the upper left corner of the receiver front panel. These switches provide the following options; (1) rectangular or polar formats for the Fourier transform results; (2) waiting or not waiting for the beginning of the next waveform cycle before acquiring new data; (3) repeating the data collection and processing procedures or stopping after one operation; and, (4) protecting the waveforms stored in memory from being over-written or normal memory operation. These switches are described in Table 12.

In addition, there are four control switches in the center of the front panel (Table 13). Left to right the switches are used to: (1) select regular operation (RUN) or a program loading mode; (2) interrupt an executing program; (3) reset and test the system; and, (4) select internal or external synchronization clocks.

TABLE 13

CONTROL SWITCHES

RESET - SYSTEM TEST

Momentary contact causes instrument to begin the system test sequence, testing memory and timing and initializing all system parameters. (See system test routine description.)

INTERRUPT

Momentary contact will interrupt and terminate the execution of any program. Control is given back to the operating system, parameters are not effected.

RUN/LOAD

Selects one of two sets of RESET and INTERRUPT vectors. RUN is the standard set for system operation. The load set corresponds to the Motorola MIKBUG vector set. If a MIKBUG oriented TTY interface board is present, programs may be loaded into memory and examined when this switch is in the load position.
This switch has the potential to be used to select between two operating systems.

SYNC INT/EXT

Selects between internal and external 7.6800 MHz clocks for data sampling timing.

Diagnostic Warning Clocks

Diagnostic warning codes are provided to aid the operator in identifying incorrectly set parameters or system malfunctions. When an error is detected, the appropriate warning code is displayed on the two least-significant digits of the display; and in most cases, the program is then halted, disabling the keypad. When the displayed code corresponds to an improperly set parameter, the parameter may be re-entered after the interrupt switch is pressed. The warning codes are defined in Table 14.

Memory Test and Memory Error Codes

As part of the system test routine, the data and program memories are checked for errors. If a data memory (RAM) error is found, the program halts and a code identifying the defective chip is displayed. (See the RAM error codes in Table 15). The program memory (PROM) test routine sequentially calculates and displays a checksum for each of the 1K PROM chips. Comparison of the displayed checksums with the correct values given in Table 15 allows identification of defective chips. (See the Memory Board Layout drawing, Figure 17, for chip locations.)

PHASE POLARITY CONVENTIONS

The Transform

Signals are SINE and COSINE transformed using an $\ell^{-i\omega t}$ convention (i.e. $-\sin\omega t$ for sine transform and $\cos\omega t$ for cosine transform).

Phase Signs

If a wave crest arrives prior to the crest of another wave of zero phase, the former wave is defined to have a positive phase advance.

TABLE 14

WARNING CODES

H1	Illegal number of cycles averaged
H2	Illegal harmonic number
H3	Illegal period
H6	Waiting for cycle pulse
H7	Waiting for sample pulse group
H8	Incorrect ratio of sample to cycle pulses
HH	Incorrect use of AMP or PHS keys

TABLE 15

MEMORY ERROR CODES

RAM ERROR CODES

FORMAT: A0105

$\frac{1}{4}$K no. of MEM (1 to 8)

Bad bit no. 1 to 8)

$\frac{1}{4}$K no. 1 to 4 for 1st K

$\frac{1}{4}$K no. 5 to 8 for 2nd K

PROM CHECK SUMS

Each checksum is displayed for 1 sec.

1PP 6

2PP-P

3PP5L

4PP1P

Check sum symbols

1K no. of 1K PROM chips

Phase Relative to Channel 1

The computed phase on channels 2 through 6 are relative to channel 1 phase; that is, channel 1 phase is subtracted from the phases on the other channels.

The phase on channel 1 is measured relative to the beginning of the cycle pulse. This pulse has a precision period matching that of the transmitter, but the two are asynchronous.

DEVELOPMENT SYSTEM

Programs for this system were developed using a CDC COMPASS-based cross assembler, written by John Wood, Lawrence Berkeley Laboratory computer consultant. The source program was written and edited in the Geoscience Engineering Laboratory on a ADM-3 CRT Terminal using the NETED interactive editing program. After assembly the machine code was written onto a cassette tape and loaded into the development hardware for debugging. The development hardware consists of the EM receiver with an extra RAM memory board to simulate PROM and a teletype interface board with a MIKBUG operating system.

APPENDIX A

A SYSTEM PROGRAM FOR A 6-CHANNEL EM RECEIVER

```
         IDENT   CAR,BEGIN
         SST
         M6800

**********EM RECEIVER**********

** PURPOSE - COMPLETE SYSTEM PROGRAM FOR A 6 CHANNEL EM RECEIVER            **
**           STACKS 6 CHANNELS OF SIGNALS, SINE AND COSINE TRANSFORMS       **
**           ANY HARMONIC IN SIGNALS, OUTPUTS AMPLITUDES IN MILLIVOLTS      **
**           AND PHASES RELATIVE TO CHANNEL ONE IN DEGREES                  **
**  AUTHOR  -  GARY L OPPLIGER , ENGINEERING GEOSCIENCE GROUP                **
**             UNIV. OF CAL BERKELEY.                                        **
**  VERSION - 2.0   JULY 12, 1978                                            **

**********DEFINITIONS**********

BEGIN    EQU    0≡B000

** PIA ADRS DEFINITIONS **

PIAA1    EQU    0≡FFE4      * LCD AND PRINTER PIAS
CPIAA1   EQU    0≡FFE5      *
PIAB1    EQU    0≡FFE6      *
CPIAB1   EQU    0≡FFE7      *
PIAA2    EQU    0≡FFE8      *
CPIAA2   EQU    0≡FFE9      *
PIAB2    EQU    0≡FFEA      *
CPIAB2   EQU    0≡FFEB      *
PIAA3    EQU    0≡FFEC       * KEYPAD AND MODE
CPIAA3   EQU    0≡FFED       *
PIAB3    EQU    0≡FFEE       * SWITCH PIA
CPIAB3   EQU    0≡FFEF       *
PIAA4    EQU    0≡FF08      * TIMER PIA
CPIAA4   EQU    0≡FF09      *
PIAB4    EQU    0≡FF0A      *
CPIAB4   EQU    0≡FF0B      *
PIAA5    EQU    0≡FF0C       * ADC PIA
CPIAA5   EQU    0≡FF0D       *
PIAB5    EQU    0≡FF0E       *
CPIAB5   EQU    0≡FF0F       *

***PROGRAM STACK LOCATION*******

STACKP   EQU    0≡14F        BOTTOM OF PROGRAM STACK  (100 THROUGH 14F)

******VARIABLE DEFINITIONS******


FLAG12   EQU    0≡49         1 BYTE   EQ 1 IF 12 BIT ADC WORDS ARE USED
DATAB    EQU    0≡4A         2
PSHBOT   EQU    0≡4C         2
DTPTCH   EQU    0≡4E         1  THIS VALUE MUST LEAD PTCYCH
PTCYCH   EQU    0≡4F         1
VAR      EQU    0≡50         SETS LOCATION OF BLOCK OF VARIABLES IN RAM

CYCAVG   EQU    VAR          1 BYTE
NCYCLE   EQU    VAR+1        2 BYTES
PTSCYC   EQU    VAR+3        1
HRMNIC   EQU    VAR+4        1
CNTINC   EQU    VAR+5        1
CNLPPT   EQU    VAR+6        1
NCHPR    EQU    VAR+7        1
NOSHF    EQU    VAR+8        1
```

```
DATPT     EQU    VAR+9       2
STORAR    EQU    VAR+11      2
TABLEP    EQU    VAR+13      2
BCDOUT    EQU    STORAR      2 BYTES
BNIN      EQU    TABLEP      2
TABLEB    EQU    VAR+15      2
STRPT     EQU    VAR+17      2
CHCNT     EQU    VAR+19      1

Y         EQU    VAR+20      2
XX        EQU    VAR+22      2
U         EQU    VAR+24      4
FF        EQU    VAR+28      1

SRTCH1    EQU    VAR+29      2
SRTCH2    EQU    VAR+31      1
SRTCH3    EQU    VAR+32      1
EXPA      EQU    SRTCH2      1
COEB      EQU    SRTCH3      1
HARM10    EQU    VAR+33      1

VV        EQU    VAR+34

C3        EQU    VV          MS * SCRATCH AREA
C2        EQU    VV+1           * FOR BINBCD
C1        EQU    VV+2        LS *
D3        EQU    VV+3        MS  * UNSIGNED BINARY INPUT
D2        EQU    VV+4            * MAX. OF 23 BITS
D1        EQU    VV+5        LS  *
OUT4      EQU    VV+6        MS * DECIMAL OUTPUT
OUT3      EQU    VV+7           * FORMAT- 08 76 54 32
OUT2      EQU    VV+8           *
OUT1      EQU    VV+9        LS *
SUBT      EQU    VV+10       SCRATCH
PHS1      EQU    VV+11       1   MS BYTE  PHASE COR IN BINARY
PHS2      EQU    VV+12       2   2 LS BYTES
PHS10     EQU    0=E1        4 BYTES
PTCY10    EQU    0=E5        1 BYTE  PTS/CYC IN BCD
NTACQ     EQU    0=E6        2 BYTES NO. OF TIMES ACQUS1 IS CALLED
HSAVE     EQU    0=E8        1   HARMONIC NO. STORAGE LOCATION
NRUN      EQU    0=E9        1 BYTE
NSTN      EQU    0=EA        1
SRTCH4    EQU    0=EB        1
SRTCH5    EQU    0=EC        1
SRTCH6    EQU    0=ED        2
ROTACC    EQU    0=EF        1 VARIABLE FOR CONTROLLING ACCURACY OF ARCTAN FN.
REAL      EQU    0=F0        2 HOLDS ADRS OF REAL VALUE
IMAG      EQU    0=F2        2 HOLDS ADRS OF IMAG VALUE
DIVTEN    EQU    0=F4        1 DIVIDE BY TEN FLAG ( MOVE DECIMAL POINT LEFT )
CHNO      EQU    0=F5        1 CHANNEL NO.
QUAD      EQU    0=F6        1 QUADRANT NO.

***  DEFINITIONS OF RAM STORAGE AREAS   ***

**  UNUSED RAM AREA (150 THROUGH 1FF) **

RAMSTR    EQU    0=0000      START OF RAM
RAMEND    EQU    0=07FF      END OF RAM   2K REQUIRED

AMPSTR    EQU    0=98         24 BYTES
PHSSTR    EQU    0=80         24 BYTES
STARTC    EQU    0=B0         24 BYTES
STARTS    EQU    0=C8         24 BYTES
```

```
DATA      EQU    0=0200       1536 BYTES ENDS AT 07FF
** DATA IS BEGINNING OF DATA STACKING AREA. ONLY THE FIRST 96 BYTES ARE **
** USED WHEN 4 PTS/CYC ARE STACKED THE REMAINDER IS USED AS A DATA      **
** COLLECTION AREA. DPSHT AND DFSHB DEFINE COLLECTION AREA               **
DPSHT     EQU    DATA+384       TOP OF DATA STACK USED WITH 4,16 PTS/CYC

** PERIOD TO FTSCYC AND ACC WORD SIZE MAP **
** FOR MIN 930 KHZ CPU CLOCK              **
E0        EQU    0=60         *  6    LONG PERIOD LIMIT
CC0       EQU    0=10         * 1.0  (E.G. 1.0E6 MILLISEC )
E1        EQU    0=10          *  1      64 PTS/CYC
CC1       EQU    0=80          * 8.0   12.5HZ  OR 80MS
E2        EQU    0=10         *  1       16 PTS/CYC
CC2       EQU    0=10         *  1.0   100HZ
E3        EQU    0=00          *  0       4 PTS/CYC
CC3       EQU    0=10          * 1.0   SHORT PERIOD LIMIT
E4        EQU    0=10         * 1  12 TO 8 BIT ADC CUTOFF
CC4       EQU    0=20         * 2.0   50HZ  OR 20MS


          ORG    BEGIN

          JMP    MASTER       INTP   *
          JMP    MASTER       SWI    * RESET AND INTERRUPT VECTORS
          JMP    MASTER       NMI    *
          JMP    RESETS       RESET  *

** TABLE USED BY ROUTINE RUN    EACH ROUTINE CORRESPONDS TO A KEY NO. **
FUNTAB    JMP    OSCDIS       RUN0  DISPLAY DATA ON SCOPE
          JMP    ATPONE       RUN1  ACQUIRE,TRANS,PRINT ONE HARMONIC
          JMP    SINCOS       RUN2  SIN AND COS TRANSFORM ROUTINE
          JMP    CHARTD       RUN3  DUMP DATA ON CHART PAPER
          JMP    ATPEVO       RUN4  ACQUIRE,TRANS,PRINT EVERY OTHER HARMONIC
          JMP    PRTEVO       RUN5  PRINT EVERY OTHER HARMONIC
          JMP    MAXSGN       RUN6  FIND MAXIMUM SIGNAL
          JMP    ATPALL       RUN7  ACQUIRE,TRANS,PRINT ALL HARMONIC
          JMP    PRTALL       RUN8  PRINT ALL HARMONICS
          JMP    VOLTMT       RUN9  MAKES DEVICE A VOLT METER


RESETS    LDS    [STACKP      RESET STACK POINTER TO PROGRAM STACK LOCATION
          LDX    [0=FFF4      * SET TIMER PIA
          STX    PIAA4
          STX    PIAB4
          CLRB                B FLAG SET,RESULTS IN BRANCH TO SYSTST
          BRA    INZS1
INZSYS    LDAB   [01          B FLAG SET,PREVENTS BRANCH TO SYSTST
          CLR    CPIAA1
          CLR    CPIAB1
          CLR    CPIAA2
          CLR    CPIAB2
          CLR    CPIAA3
          CLR    CPIAB3
          CLR    CPIAA5
          CLR    CPIAB5
INZS1     LDX    [0=C0F4      * SET KEYPAD PIA  PA6-PA7 ARE OUTPUTS
          STX    PIAA3        * PA0-PA5 ARE INPUTS FOR COLUMNS
          LDX    [0=0FF4      * PB0-PB3 ARE OUTPUTS FOR ROWS
          STX    PIAB3        * PB5-PB7 ARE INPUTS FOR MODE SWITCHES
          STX    PIAA5         * SET ADC PIA
          LDX    [0=00F4       * PA0-PA4 ARE OUTPUTS
          STX    PIAB5         * PB0-PB7 ARE INPUTS
```

```
          LDX     [0=FFF4     * SET LDC PIAS
          STX     PIAA1       * CB2 IS SET HIGH
          STX     PIAB1       *
          STX     PIAA2       *
          LDX     [0=F7FC     * MAKE PB3 AN INPUT FOR PRINTER
          STX     PIAB2       * ALL OTHERS ARE OUTPUTS
          CLR     PIAB2       * ASSURES PRINTER POWER IS OFF.
          TSTB                * CHECK B FLAG
          BNE     INZS2       * IF EQU BRANCH TO SYSTEM TEST
          BRA     SYSTST      SYSTEM TEST
INZS2     JSR     CLRDIS      CLEAR DISPLAY
          RTS


** SYSTEM TEST    CHECKS  2K RAM AND 4K ROM, WRITES SQUAREWAVE INTO MEM **
SYSTST    BRA     RAMCK       BRANCH TO RAM CHECK
RAMRTN    JSR     TIMERT      DC TEST OF SAMPLE AND CYCLE PULSES
          JSR     TESTB3      * READ SWITCH 3
          BNE     SYSTST      * IF SET REPEAT RAM AND PULSE TESTS
          LDAA    [0=CC       * SET UP LCD AS  XPPXX TO INDICATE PROM
          STAA    PIAB1       * PLACE PP IN DIGITS 3 AND 4
          CLRB
          LDX     [BEGIN      BEGINNING CF ROM
          STX     DATPT        SAVE
PROMLP    JSR     PROMCK      CREATE CHECK SUM ON 1K OF PROM
          INCB                INC 1K CHIF COUNTER
          STAA    PIAA1       PLACE CHECK SUM IN LOWEST 2 DIGITS ON LCD
          STAB    PIAA2       PLACE CHIP NO. IN 5TH DIGIT POSITION ON LCD
          LDX     [0=FF02
          JSR     DELAY3      DELAY OF 900000 MACHINE CYCLES PER CALL
          CMPB    [04         IF CHIP NO. EQU 4 QUIT PROM CHECK
          BNE     PROMLP
          JSR     LDMEM       LOAD MEMORY WITH SQUAREWAVE FOR TESTS
          LDAA    [0=88       * LOAD -.8.8.8.8.8  INTO LCD
          STAA    PIAA1       * TO TEST ALL LCD SEGMENTS
          STAA    PIAB1       *
          LDAA    [0=F8       *
          STAA    PIAA2       *
          LDAA    [0=03       *
          STAA    PIAB2       *
          JSR     PWRON       PRINTER POWER ON
          JSR     PRNT        * PRINT -.8.8.8.8.8
          JSR     PRNT        * TWICE
          JSR     PWROFF      PRINTER POWER OFF
          JSR     INZVAR      INITIALIZE VARIABLES
          JMP     CKLOCP      TESTS COMPLETE GO TO MAIN PROGRAM


** ROTATING BIT RAM TEST    ROUTINE DOES NOT USE RAM STORAGE   **
** IF A BAD LOCATION IS FOUND PROGRAM HALTS AND LCD DISPLAYS   **
** A IN 5TH DIGIT TC INDICATE RAM, BIT NO.(1 - 8) IN 3RD DIG   **
** AND QUARTER K NO. IN 1ST DIG (1 - 8). FORMAT  A0502         **
RAMCK     LDX     [RAMSTR     LOCATION OF START OF CHECK
          LDAA    [0=DD       * DISPLAY  (AAAAA) ON LCD
          STAA    PIAA1       *
          STAA    PIAB1       *
          LDAA    [0=0D       *
          STAA    PIAA2       *
SHFTST    CLR     00,X        * WRITE ZEROS INTO MEM
          LDAA    00,X        *
          BNE     BADBIT      * IF NOT ZERO BAD BIT
          INCA                 SET ACCA TO 1
```

```
          STAA    00,X
          CMPA    00,X          COMPARE ACC A WITH MEMORY LOCATION
          BNE     BADBIT        IF NOT EQU BRANCH TO BAD BIT
RAMLP     ASLA                  *
          ASL     00,X          * SHIFT BITS
          CMPA    00,X          COMPARE ACC A WITH MEMORY LOCATION
          BNE     BADBIT        IF NOT EQU BRANCH TO BAD BIT
          TSTA                  * TEST FOR BIT IN 8 POSITION
          BPL     RAMLP         * IF FOUND INC INDEX,TEST NEXT BYTE
          INX
          CPX     [RAMEND+1     ADRS OF LAST BYTE TO BE TESTED +1
          BNE     SHFTST        IF NOT LAST BYTE, TEST NEXT BYTE
          JMP     RAMRTN        BRANCH SERVES FN SIMILAR TO RTS
BADBIT    CLRB                  *
TSTLP1    INCB                  * DETERMINE BIT POSITION OF BAD BIT
          LSRA                  *
          BNE     TSTLP1        *
          STX     PIAA1         STORE INDEX IN LCD, MS BYTE GOES IN DIG 1,2
          LDAA    [0=F4         * LS BYTE WENT INTO CONTROL REG
          STAA    CPIAA1        * RESTORE CONTROL REG
          INC     PIAA1         INC DIG 1 TO CREATE 1/4 K NO.
          STAB    PIAB1          * WRITE BIT NO. IN DIGITS 3 AND 4.
RAMSLF    BRA     RAMSLF        HALT THE COMPUTER


** PROM CHECK   CALCULATES A CHECKSUM FOR 1K OF ROM    **
PROMCK    CLR     SRTCH1
          CLR     SRTCH1+1
          CLRA
PRMLP     LDX     DATPT
          ADDA    00,X
          INX
          STX     DATPT
          LDX     SRTCH1
          INX
          STX     SRTCH1
          CPX     [0=0400       * CHECK FOR 1000TH TIME THROUGH LOOP
          BNE     PRMLP         *
          RTS


** TIMER SAMPLE AND CYCLE PULSE TEST     TESTS RATIO OF SAMPLE/CYCLE PULSES **
** FOR 64, 16, AND 4 PTS PER CYCLE   DISPLAYS H6 WHEN WAITING FOR CYCLE     **
** PULSE, DISPLAYS H7 WHEN WAITING FOR SET OF SAMPLE PULSES, DISPLAYS H8    **
** AND HALTS PROGRAM IF RATIO OF SAMPLE TO CYCLE PULSES IS IN ERROR         **
TIMERT    LDAA    [0=25         PERIOD COEF. 2.5
          CLRB                  PERIOD EXPONENT E0 MSEC
          BSR     STRPER        STORE PERIOD   SET COUNTERS
          LDX     [25           SET RATIO SAMP/CYC +1 FOR THIS PERIOD
          BSR     RATIOT        DO RATIO TEST
          LDAA    [0=40         PERIOD COEF.  4.0
          LDAB    [0=10         PERIOD EXP.   E1
          BSR     STRPER        STORE PERIOD   SET COUNTERS
          LDX     [97           SET RATIO  AS ABOVE
          BSR     RATIOT        DO RATIO TEST
          LDAA    [0=20         PERIOD COEF.  2.0
          TAB                   PERIOD EXP    E2   5HZ
          BSR     STRPER        STORE PERIOD
          LDX     [385          SET RATIO
          BSR     RATIOT        DO RATIO TEST
          RTS


** STORES PERIOD AND SET PRE AND POST COUNTERS  **
STRPER    STAA    PIAA4         SET PERIOD COEF.
```

```
         STAB    PIAB4        SET PERIOD EXP.
         JSR     PRDSET       SET PRE AND POST COUNTERS AND PTS/CYC
         RTS


** TEST OF RATIO OF SAMPLE TO CYCLE PULSES  **
RATIOT   STX     SRTCH6       STORE IDEAL RATIO NO.
         JSR     CLRDIS       CLEAR LCD DISPLAY
         LDX     [0000        SET RATIO COUNTER TO ZERO
         LDAB    PIAB4        CLEAR CYCLE PULSE BY READING PIA
         LDAB    PIAB5        CLEAR SAMPLE PULSE
         LDAB    [0=B6        * WAITING FOR CYCLE PULSE
         STAB    PIAA1        * PUT H6 ON LCD
TTLP1    LDAA    CPIAB4        * WAIT FOR CYCLE PULSE
         BPL     TTLP1         *
         LDAB    PIAB4        CLEAR CYCLE PULSE
         INC     PIAA1        PUT H7 ON LCD. WAITING FOR SET OF SAMPLE PULSES
TTLP2    LDAA    CPIAB5       WAIT FOR SAMPLE PULSE
         BPL     TTLP2
         LDAB    PIAB5        CLEAR SAMPLE PULSE
         INX                  INC RATIO COUNTER
         LDAA    CPIAB4       * LOOP BACK UNTIL CYCLE PULSE IS FOUND
         BPL     TTLP2        *
         CPX     SRTCH6       COMPARE RATIO COUNT WITH IDEAL VALUE
         BEQ     TTOK         IF EQUAL RETURN
         INC     PIAA1        PUT H8 ON LCD TO INDICATE RATIO ERROR
TTHALT   BRA     TTHALT
TTOK     JSR     CLRDIS       CLEAR LCD
         RTS



** LOAD MEMORY WITH +5,-5 VOLT SQUAREWAVE FOR TEST                **
** SINGLE CHNL PATTERN IS 2 HIGH,8 LOW,8 H,8 L,ETC FOR 64 PTS     **
LDMEM    LDX     [1536        NO. BYTES TO BE CLEARED
         STX     SRTCH1
         LDX     [DATA        START OF SECTION TO BE CLEARED
         JSR     CLRM2        CLEAR MEMORY
         LDX     [DATA        DEFINE STARTING ADRS
         LDAB    [12          FOR FIRST PARTIAL CYCLE
         BRA     LDM2         START PARTIAL CYCLE
LDM1     LDAB    [48          6CH*8PTS=48 SET COUNTER FOR 8 +5V POINTS
LDM2     LDAA    [0=7F         +5 VOLTS
         STAA    00,X         LOAD VALUE IN MEMORY
         CPX     [DATA+1532   COMPARE WITH LAST LOCATION
         BEQ     LDMOUT       IF EQU RTS
         JSR     INX4         SELECT NEXT LOCATION (INX 4 TIMES)
         DECB                 DEC COUNTER
         BNE     LDM2         IF NOT 0 STORE ONE MORE +5V VALUE
         LDAA    [0=80         -5V
         LDAB    [48          SET COUNTER FOR 8 -5V POINTS
LDM3     STAA    00,X         STORE -5V VALUE
         JSR     INX4         SELECT NEXT VALUE
         DECB                 DEC COUNTER
         BNE     LDM3         IF NOT 0 STORE ON MORE -5V VALUE
         BRA     LDM1      DO NEXT +5 CYCLE
LDMOUT   RTS


** INITIALIZE VARIABLES            **
INZVAR   LDAA    [0=9C        NO. OF BYTES TO BE CLEARED
         LDX     [0=0050      START OF SECTION TO BE CLEARED
```

```
         JSR     CLRMEM        CLEAR MEMORY
         INC     HARM10
         INC     HRMNIC
         INC     NCYCLE+1
         LDAA    [4
         STAA    ROTACC        SET ARCTAN FN ACCURACY
         JSR     PRDSET        SET PTS/CYC AND PRE AND POST COUNTERS
         LDAA    [0=06         *
         STAA    NCHPR         * SET NO. CF CHANNELS OPERATED ON
         RTS



** LOOK FOR ANY PRESSED KEY  IF FOUND ACC A IS A NONZERO VALUE **
KEYQ     CLRA                  2 (MACHINE CYCLES)
         STAA    PIAB3         5
         LDAA    PIAA3         4
         COMA                  2
         ANDA    [0=3F         2
         RTS                   5



** DELAY ROUTINE
** DELAY EQ -  NO. CYC.=62+(SRTCH3-1)*14+(SRTCH2-2)*3580   **
DELAY1   LDX     [0=1A62         * PRESET DELAY 87278 CYCLES
DELAY3   STX     SRTCH2
DELAY2   DEC     SRTCH2
         BNE     DLYLP1
         RTS
DLYLP1   DEC     SRTCH3
         BEQ     DELAY2
         BRA     DLYLP1
         RTS


** CHECKS FOR RELEASE OF PRESSEC KEY **
RELESE   CLRA
         STAA    PIAB3            ROW
RELLP1   LDAA    PIAA3          COLUMN
         COMA
         ANDA    [0=3F
         BNE     RELLP1
         BSR     DELAY1           ABOUT 90 MILLISEC DELAY FOR KEY DEBOUNCE
         RTS



** SOUNDS BEEPER FOR 8518 CYCLES  **
BEEP     LDAA    [0=FC         * BRING LINE CB2 HIGH FOR BEEPER
         STAA    CPIAB3        *
         LDX     [0=0462        *
         BSR     DELAY3         * DELAY
         LDAA    [0=F4         *
         STAA    CPIAB3        * BRING CB2 LOW FOR BEEPER OFF
         RTS
```

```
** ACTIVATE ONE ROW ON KEYPAD, TEST COLUMNS FOR RESPONSE  **
LDTST     STAA   PIAB3          ACTIVATE ROW
          LDAA   PIAA3          READ COLUMN
          INCB
          COMA
          ANDA   [0≡3F
          RTS



** INTERPRET PRESSED KEY  **
INTP      ASLB
          ASLB
          ASLB
          BSR    DELAY1
          DECB
          LDX    [0≡F901
          STX    SRTCH2
LOOP1     INC    SRTCH2
          BEQ    ERR
          INCB
          CMPA   SRTCH3
          BEQ    END
          ASL    SRTCH3
          BRA    LOOP1
ERR       NOP
          BRA    RWSLCT
END       BSR    BEEP
          TSTB
          RTS



** MASTER ROUTINE FOR SCANNING KEYPAD **
SCNKEY    BSR    RELESE
RWSLCT    LDAB   [0≡FF
          LDAA   [0≡0E
          BSR    LDTST
          BGT    INTP
          LDAA   [0≡0D
          BSR    LDTST
          BGT    INTP
          LDAA   [0≡0B
          BSR    LDTST
          BGT    INTP
          LDAA   [0≡07
          BSR    LDTST
          BGT    INTP
          BRA    RWSLCT
          RTS



** CLEAR LIQUID CRYSTAL DISPLAY **
CLRDIS    LDAA   [0≡FF
          STAA   PIAA1
          STAA   PIAB1
          LDAA   [0≡0F
```

```
          STAA    PIAA2
          LDAA    PIAB2        * SAVE PRINTER CONTROLS       MAY 14 78
          ANDA    [0≡FC        *
          STAA    PIAB2
          RTS



** DISPLAYS A NO. ON LSD OF LCD WITHOUT AFFECTING OTHER VALUES **
LCD1      LDAB    PIAA1
          ANDB    [0≡FC
          ANDA    [0≡0F
          ABA
          STAA    PIAA1
          ANDA    [0≡0F
          RTS



** CONVERT KEY CODE NO. TO DECIMAL VALUE 0-9     **
TSTCD     CMPB    00,X         COMPARE KEYCODE WITH TABLE CODE
          BNE     SCLP         * IF THEY ARE EQUAL LOAD COUNTER
          LDAA    SRTCH3       * INTO ACC A AS BCD VALUE
SCLP      INX                  POINT INDEX TO NEXT CODE
          DEC     SRTCH3       * DEC COUNTER
          RTS

** KEYPAD CODE CONVERSION TABLE    **
CONV1     CON     0≡0A         9
          CON     0≡09         8
          CON     0≡08         7
          CON     0≡12         6
          CON     0≡11         5
          CON     0≡10         4
          CON     0≡1A         3
          CON     0≡19         2
          CON     0≡18         1
          CON     0≡C1         0



** MASTER ROUTINE FOR KEY CODE TO DECIMAL CONVERSION    **
NCODE     LDAA    [0≡09
          STAA    SRTCH3
          LDAA    [0≡88
          LDX     [CONV1       KEYPAD CODE CONVERSION TABLE
NCLP      JSR     TSTCD
          BMI     ENDL
          BRA     NCLP
ENDL      RTS



** SELECTS 1 OF 10 SUBROUTINES THAT MAY BE CALLED   **
RUN       BSR     CLRDIS
          BSR     SCNKEY       * LOOK FOR NO. ENTERED ON KEYPAD
          BSR     NCODE        *
          CMPA    [0≡88        IF A NUMBER IS NOT FOUND RETURN
          BEQ     RNOUT
          BSR     LCD1           DISPLAY ON LCD
          LDX     [RUNTAB-3      RUN TABLE -3
```

```
RNLP       INX
           INX
           INX
           DECA                ACC A IS COUNTER SELECTOR
           BPL     RNLP
           JSR     00,X        CALL  SELECTED RUN ROUTINE
RNOUT      LDAB    [0≡F4        * TURN OFF OPEN REGISTER INDICATOR
           STAB    CPIAA2       *
           RTS



** CONTROL KEY JUMP TABLE **
CKTAB      JMP     RUNB        0 EQUIVALENT KEY NO.
           JMP     HRMSLT      1
           JMP     MEM         2
           JMP     RUN         3
           JMP     NCHNLS      4
           JMP     PER         5
           JMP     CYC         6
           JMP     CDUM        7
           JMP     AMP         8
           JMP     PHS         9

CDUM       RTS

** THIS IS THE MASTER CONTROL LOOP FOR THE PROGRAM   **
MASTER     LDS     [STACKP       SET STACK POINTER TO PROGRAM STACK LOCATION
CONKEY     JSR     INZSYS
CKLOOP     JSR     SCNKEY
           JSR     NCODE
           CMPA    [0≡BB
           BNE     CKLOOP
           SUBB    [0≡03
           JSR     NCODE
           CMPA    [0≡BB
           BEQ     CKLOOP
           LDAB    [0≡FC       * TURN ON OPEN REGISTER INDICATOR
           STAB    CPIAA2      *
           LDX     [CKTAB-3    CONTROL KEY JUMP TABLE -3
           BSR     RNLP
           BRA     CKLOOP




** DISPLAYS PERIOD  FORMAT - (B B 1.2 B 3) B = BLANK **
DISPER     LDAA    PIAA4
           STAA    PIAB1
           LDAA    PIAB4
           ANDA    [0≡70
           LSRA
           LSRA
           LSRA
           LSRA
           ADDA    [0≡F0
           STAA    PIAA1
           LDAA    [0≡4F
           STAA    PIAA2
           RTS
```

```
** SUBROUTINES STDIG1 AND STDIG2 - STORE DIGITS 1 AND 2 **
STDIG1    LDAB   [0≡0F
          ANDB   0,X
          BSR    SHFL4
          BRA    SAME
STDIG2    LDAB   [0≡F0
          ANDB   0,X
SAME      ABA
          STAA   0,X
          JSR    SCNKEY
          RTS



** LOAD PERIOD DISPLAYED ON LCD TO TIMER  **
LDPER     LDAA   PIAB1
          STAA   PIAA4
          LDAB   PIAB4
          ANDB   [0≡8F
          LDAA   PIAA1
          ANDA   [0≡07
          JSR    SHFL4
          ABA
          STAA   PIAB4
LDP1      BSR    DISPER
          RTS



** ALLOWS ENTEFING ANC EXAMINATION OF PERIOD  **
PER       JSR    CLFDIS     CLEAR LCD
          BSR    DISPER
          JSR    SCNKEY
          BEQ    TOCK
          JSR    NCODE
          LDX    [PIAB1
          JSR    STCIG1
          BEQ    TOCK
          JSR    NCODE
          LDX    [PIAB1
          JSR    STDIG2
          BEQ    TOCK
          JSR    NCODE
          LDX    [PIAA1
          JSR    ST[IG2
          BEQ    TOCK
          CMPB   [0≡05
          BNE    TOCK
          BSR    LDPER
          BSR    PRDSET     SET PTS/CYC AND PRE AND POST COUNTERS
TOCK      JSR    DISPER
          RTS


PRDSET    JSR    PTSSET     SELECT NO. PTS/CYC
          JSR    PTSCON      SET PTS/CYC CONTROL BITS ON TIMER BOAPD
          LDAA   PTSCYC     * CONVERT TO BCD
```

```
          JSR     BNBCD2       *
          STAA    PTCY10       *
          RTS



SHFL4     ASLA
          ASLA
          ASLA
          ASLA
          RTS


** DISPLAY ON LCD NO. OF CYCLES TO BE AVERAGED **
DISCYC    LDAB    CYCAVG
          JSR     DISDG2
          RTS



** MANAGES DISPLAY AND ENTRY OF NO. OF CYCLES TO BE AVERAGED **
CYC       BSR     DISCYC
          JSR     SCNKEY
          BEQ     LASTD1
STROV1    JSR     NCODE
          LDX     [PIAB1
          LDAB    [0≡FF
          STAB    0,X
          JSR     STDIG1
          BEQ     LASTD1
          JSR     NCODE
          LDX     [PIAB1
          JSR     STDIG2
          BEQ     LASTD1
          BRA     STROV1
LASTD1    LDAA    PIAB1
          STAA    CYCAVG
          BSR     DISCYC
          BSR     SETCNT
          RTS



** SET NO. OF CYCLES TO BE AVERAGED IN BINARY **
SETCNT    CLRB
          LDAA    CYCAVG
          CMPA    [0≡15
          BHI     ERROR1
          CMPA    [0≡0F
          BLE     SKIP
          LDAB    [0≡0A
          ANDA    [0≡0F
SKIP      ABA
          BSR     SETCN2
          JSR     CYCDEC       CONVERT NO. OF CYCLES TO BCD AND DISPLAY
          RTS
ERROR1    LDAB    [0≡B1
          STAB    PIAB1
          RTS
```

```
SETCN2    LDX     [0≡0001
          STX     NCYCLE
SLOOP     DECA
          BMI     RTN
          ASL     NCYCLE+1
          ROL     NCYCLE
          BRA     SLOOP
RTN       RTS



** DISPLAY AMPLITUDE AND PHASE  **
AMP       LDX     [AMPSTR
          STX     SRTCH1
          BRA     AP1
PHS       LDX     [PHSSTR
          STX     SRTCH1
AP1       JSR     CLRDIS
          JSR     SCNKEY
          JSR     NCODE
          CMPA    [0≡0B
          BNE     AP2
          STAA    PIAA1
          BRA     APOUT
AP2       DECA
          BMI     APOUT
          CMPA    [0≡05
          BGT     APOUT
          ASLA
          ASLA
          ADDA    SRTCH1+1
          STAA    SRTCH1+1
          BCC     AP3          * ADDED APRIL 22,78
          INC     SRTCH1       *
AP3       LDX     SRTCH1
          BSR     LCDDIS
APOUT     RTS

** LOAD LCD WITH 4 BYTES POINTED AT BY INDEX  **
LCDDIS    LDAA    3,X
          STAA    PIAA1
          LDAA    02,X
          STAA    PIAB1
          LDAA    01,X
          STAA    PIAA2
          LDAA    00,X
          ANDA    [0≡03        *  SAVE PRINTER CONTROLS          MAY 14 78
          LDAB    PIAB2        *
          ANDB    [0≡FC        *
          ABA                  *
          STAA    PIAB2
          RTS

** MANAGES PARAMETERS ACCESSED THROUGH MEM KEY **
MEM       JSR     CLRDIS
          JSR     SCNKEY       *
          JSR     NCODE        * DECODE KEY
          ANDA    [0≡0F
          CMPA    [0≡00
          BNE     MJ1
```

```
         JMP    DISPTC        DISPLAY PTS/CYC
MJ1      CMPA   [0=01
         BNE    MJ2
         JMP    DISPHC        DISPLAY PHASE CORRECTION
MJ2      CMPA   [0=02
         BNE    MJ3
         JMP    STATNO
MJ3      CMPA   [0=03
         BNE    MJ4
         JMP    RUNNC
MJ4      CMPA   [0=04
         BNE    MJ5
         JMP    CROTAC       ROUTINE TO SET ROTATION ACCURACY
MJ5      RTS


** DISPLAY POINTS PER CYCLE **
DISPTC   JSR    CLRDIS
         LDAA   PTCY10
         STAA   PIAA1        DISPLAY POINTS / CYCLE
         RTS


** DISPLAY STATION NO. AND RUN NO. FORMAT = ( XX XX)  **
DSTNRN   JSR    CLRDIS
         LDAB   NSTN         LOAD STATION NO.
         CLRA
         JSR    RORBA        CIRCULATE ACC B INTO ACC A  RIGHT ROTATION
         ORAA   [0=0F        BLANK DIGIT NO 3
         STAA   PIAB1         LOAD DIGITS 3 AND 4 INTO LCD
         STAB   PIAA2         LOAD DIGIT 5 AND DECIMAL PTS INTO LCD
         LDAA   NRUN         * DISPLAY RUN NO. ON DIGITS 1 AND 2
         STAA   PIAA1        *
         RTS


** DISPLAYS NO. OF CYCLES TO BE AVERAGED IN BCD FORM  **
CYCDEC   LDX    NCYCLE       * CONVERT NCYCLE TO 5 DIGIT BCD VALUE
         STX    D2            D1 IS LS BYTE
         CLR    D3            CLEAR MS BYTE
         JSR    BINBCD       CONVERT TO BCD
         JSR    CLRDIS       CLEAR DISPLAY
         LDAA   OUT1         LS 2 DIGITS
         STAA   PIAA1        TO LCD 2 LS DIGITS
         LDAA   OUT2         NEXT 2 DIGITS
         STAA   PIAB1        TO LCD
         LDAA   OUT3         MS DIGIT
         STAA   PIAA2        TO LCD
         RTS




*****   COSINE AND SINE CONSTANT TABLE   *****
* TABLE STRUCTURE - 80 2 BYTE CONSTANTS, WITH OVERLAPPING SETS OF 64 COS *
* AND 64 SIN 2 BYTE CONSTANTS.

COSTOP   CON    0=7F,0=FF,0=7F,0=61,0=7D,0=89,0=7A,0=7C
         CON    0=76,0=41,0=70,0=E2,0=6A,0=6D,0=62,0=F1
         CON    0=5A,0=82,0=51,0=33,0=47,0=1C,0=3C,0=56
         CON    0=30,0=F8,0=25,0=28,0=18,0=F9,0=0C,0=8C
SINTOP   CON    0=00,0=00,0=F3,0=74,0=E7,0=07,0=DA,0=D8
```

```
          CON     0≡CF,0≡05,0≡C3,0≡AA,0≡B8,0≡E4,0≡AE,0≡CD
          CON     0≡A5,0≡7E,0≡9D,0≡0F,0≡95,0≡93,0≡8F,0≡1E
          CON     0≡89,0≡BF,0≡85,0≡84,0≡82,0≡77,0≡80,0≡9F
          CON     0≡80,0≡01,0≡80,0≡9F,0≡82,0≡77,0≡85,0≡84
          CON     0≡89,0≡BF,0≡8F,0≡1E,0≡95,0≡93,0≡9D,0≡0F
          CON     0≡A5,0≡7E,0≡AE,0≡CD,0≡B8,0≡E4,0≡C3,0≡AA
          CON     0≡CF,0≡05,0≡DA,0≡D8,0≡E7,0≡07,0≡F3,0≡74
          CON     0≡00,0≡00,0≡0C,0≡8C,0≡18,0≡F9,0≡25,0≡28
          CON     0≡30,0≡FB,0≡3C,0≡56,0≡47,0≡1C,0≡51,0≡33
          CON     0≡5A,0≡82,0≡62,0≡F1,0≡6A,0≡6D,0≡70,0≡E2
          CON     0≡76,0≡41,0≡7A,0≡7C,0≡7D,0≡89
COSBOT    CON     0≡7F,0≡61
          CON     0≡7F,0≡FF,0≡7F,0≡61,0≡7D,0≡89,0≡7A,0≡7C
          CON     0≡76,0≡41,0≡70,0≡E2,0≡6A,0≡6D,0≡62,0≡F1
          CON     0≡5A,0≡82,0≡51,0≡33,0≡47,0≡1C,0≡3C,0≡56
          CON     0≡30,0≡FB,0≡25,0≡28,0≡18,0≡F9
SINBOT    CON     0≡0C,0≡8C


** MASTER ROUTINE FOR DOING SINE AND COSINE TRANSFORMS   **
SINCOS    LDX     [DATA       START OF STORAGE AREA FOR STACKED SIGNALS
          STX     DATPT       POINTER TO SIGNALS
          LDX     [STARTC     START OF MEM SECTION TO BE CLEARED
          LDAA    [0≡30       NO. OF BYTES TO BE CLEARED IS 48
          JSR     CLRMEM      CLEAR STORAGE AREA FOR TRANS RESULTS
          LDX     [STARTC     * DEF. STORAGE AREA FOR
          STX     STORAR      *  COS TRANS RESULTS
          LDX     [COSTOP       *
          STX     TABLEP        * DEFINE TOP AND BOTTOM
          LDX     [COSBOT       * OF TRANS CON TABLE
          STX     TABLEB        * FOR COS TRANS
          BSR     TRNSFM      DO COS TRANS
          LDX     [DATA
          STX     DATPT
          LDX     [STARTS     * DEF. STORAGE AREA FOR
          STX     STORAR      * SIN TRANS RESULTS
          LDX     [SINTOP       *
          STX     TABLEP        *     SAME
          LDX     [SINBOT       * FOR SIN TRANS
          STX     TABLEB        *
          BSR     TRNSFM      DO SIN TRANS
          JSR     PHSCOR      CALCULATE PHASE CORRECTION
          CLR     DIVTEN      SET DIVIDE BY 10 FLAG TO ZERO
          JSR     TESTB1      TEST SWITCH 1
          BNE     RECT        IF ON SKIP RECT TO POLAR CONVERSION
          JSR     POLAR       CONVERT TO POLAR COORDINATES
          INC     DIVTEN      SET DIVIDE BY TEN FLAG TO 1
RECT      LDX     [PHSSTP      *
          STX     BCDOUT       *
          LDX     [STARTS      *
          STX     BNIN         *
          JSR     CNVRT        * BINARY TO BCD FOR SINE
          CLR     DIVTEN      CLEAR DIVIDE BY 10 FLAG
          LDX     [AMPSTR     *  IDENTIFY OUTPUTS AND INPUTS
          STX     BCDOUT      *
          LDX     [STARTC     *
          STX     BNIN        *
          JSR     CNVRT       * BINARY TO BCD FOR COSINE
          JSR     MPRINT      *
NOPR      RTS
```

```
** PERFORMS SINE OR COSINE TRANSFORM **
TRNSFM     JSR     CCNTRL      SET COUNTER INCREMENT (CNTINC) AND SET CNLPPT
           BSR     LOAD        TABLEP TO ACC      *  BACK SET
           SUBB    CNTINC                         *  TABLE
           SBCA    [0=00                          *  POINTER
           BSR     STORE       ACC TO TABLEP      *
TRNLP      BSR     LOAD        TABLEP TO ACC   *     ADVANCE
           ADDB    CNTINC                      *     TABLE
           ADCA    [0=00                       *     POINTER
           BSR     STORE       ACC TO TABLEP   *
           LDAA    TABLEB                         *  RESET
           LDAB    TABLEB+1                       * CONSTANT
           SUBB    TABLEP+1                       * POINTER
           SBCA    TABLEP                         *   IF
           BCS     TRN1                           * GREATER
           BRA     TRN2                           * THAN
TRN1       BSR     LOAD             * RESET       * END OF
           SUBB    [0=80       L28  * TABLE       * TABLE
           SBCA    [0=00            * POINTER     *
           BSR     STORE            *             *
TRN2       JSR     SETMLT      SET UP AND MULTIPLY
           DEC     CNLPPT
           BNE     TRNLP
           JSR     NORM        NORMALIZE TRANS RESULTS FOR 64,16,4 PTS/CYC
           RTS




** SETS SIZE FOR COUNTER INCREMENT FOR USE IN SELECTING CONSTANTS **
** FROM SINE AND COSINE TABLE.                                    **
CCNTRL     LDAA    PTSCYC
           STAA    CNLPPT
           JSR     HRMCK
           LDAA    PTSCYC
           LDAB    [0=40
LP1        LSRA
           LSRB
           CMPA    [0=02
           BNE     LP1
           LDAA    HRMNIC
LP2        CMPB    [0=01
           BEQ     OUT
           ASLA
           LSRB
           BRA     LP2
OUT        STAA    CNTINC
           RTS




** LOAD POINTER INTO ACC A AND B    **
LOAD       LDAA    TABLEP
           LDAB    TABLEP+1
           RTS
```

```
** STORES ACC A AND B INTO POINTER **
STORE      STAA   TABLEP
           STAB   TABLEP+1
           RTS



** SETS UP AND MULTIPLIES DATA FOR 6 CHANNELS  **
SETMLT     LDX    STORAR      RESET POINTER TO TRANSFORM STACK TO
           STX    STRPT       POINT AT CH 1 SIN OR COS.
           LDX    TABLEP      *  LOAD ADRS OF CONSTANT
           LDX    00,X        * LOAD CONSTANT INTO MULTIPLICAND POSITION
           STX    XX          * (NOT DESTROYED AFTER MULTIPLY)
           LDAA   #$06         * SET CHANNEL COUNTER
           STAA   CHCNT        *
CHANLP     BSR    DATSET
           JSR    MULT16
           BSR    EXPSHF
           BSR    ADTFSK
           DEC    CHCNT       *
           BNE    CHANLP      * DEC CHANNEL COUNTER
           RTS



*** SUBROUTINE  ADD TO TRANSFORM STACK  ***
ADTFSK     LDX    STRPT       LOAD STORAGE AREA POINTER
           LDAA   U+3
           ADDA   3,X
           STAA   3,X
           LDAA   U+2
           ADCA   2,X
           STAA   2,X
           LDAA   U+1
           ADCA   1,X
           STAA   1,X
           LDAA   U
           ADCA   0,X
           STAA   0,X
           JSR    INX4        SET ADRS OF STORAGE AREA POINTER TO NEXT CH
           STX    STRPT
           RTS



** SET UP DATA VALUE FOR MULTIPLY **
DATSET     LDX    DATPT       LOAD ADRS OF STACKED DATA
           LDX    00,X        * PUT DATA FROM CH N (MS 16 BITS) INTO
           STX    Y           * MULTIPLIER POSITION FOR SUB MULT16
           LDX    DATPT       RELOAD ADRS OF STACKED DATA
           JSR    INX4        POINT INDEX AT DATA FOR NEXT CH (N+1)
           STX    DATPT       STORE ADRS OF NEXT DATA VALUE
           RTS
```

```
** EXPAND PRODUCT OF MULTIPLY BY SHIFTING TO RIGHT **
**  THIS MAKES ROOM FOR STACKING 64 VALUES          **
EXPSHF     LDAB   [0=05
SHFLP      ASR    U
           ROR    U+1
           ROR    U+2
           ROR    U+3
           DECB
           BNE    SHFLP
           RTS

** ROUTINE CLEARS UNLIMITED SECTIONS OF MEM **
** INDEX MUST CONTAIN STARTING ADRS .
** SRTCH1 MUST BE NO. OF BYTES TO BE CLEARED.
CLRM2      STX    DATPT
CLRMLP     LDX    DATPT
           CLR    00,X
           INX
           STX    DATPT
           LDX    SRTCH1
           DEX
           STX    SRTCH1
           BNE    CLRMLP
           RTS


** INDEX MUST CONTAIN STARTING ADRS / ACC A MUST CONTAIN NO. CF BYTES TO   *
**  BE CLEARED.
CLRMEM     CLR    00,X
           INX
           DECA                 DEC COUNTER
           BNE    CLFMEM          LOOP
           RTS


** SELECTS MAXIMUM POINTS PER CYCLE POSSIBLE **
PTSSET     LDAA   PIAA4         *
           CMPA   [0=0A         *
           BCS    PTSERR        * LEADING ZERO FOUND ERROR
           LDAA   [E0            *
           LDAB   [CC0           *
           BSR    COMPAR         *
           BMI    PTS64          *
           BRA    PTSERR         * PERIOD TOO LARGE
PTS64      LDAA   [64
           STAA   PTSCYC        *
           LDAA   [E1           *
           LDAB   [CC1          *
           BSR    COMPAR        *
           BMI    PTS16         *
           BRA    SETFL         * 64 PTS/CYC
PTS16      LDAA   [16            *
           STAA   PTSCYC         *
           LDAA   [E2            *
           LDAB   [CC2           *
           BSR    COMPAR         *
           BMI    PTS4           *
           BRA    STF12          *  16 PTS/CYC
PTS4       LDAA   [4            *
           STAA   PTSCYC        *
           LDAA   [E3           *
           LDAB   [CC3          *
```

```
          BSR     COMPAR        *
          BMI     PTSERR        * PERIOD TOO SMALL
** NEXT 6 LINES SET UP FOR 16 PTS/CYC. THESE VALUES WILL BE USED IF **
** FLAG12 IS CLEARED  IF 8 BIT ADC WORD SIZE IS USED                 **
STF12     LDX     [0=8060       128, 96 DEC  OR 80, 60 HEX
          STX     DTPTCH         DTPTCH, PTCYCH
          LDX     [DPSHT+768    *
          STX     PSHBCT        * DEFINE BOTTOM OF DATA COLLECTION AREA
          LDX     [DATA+380      *
          STX     DATAB           * BOTTOM OF STACKING AREA
          LDAA    [16
          CMPA    PTSCYC
          BEQ     SETFL         IF PTSCYC = 16 SKIP NEXT SECTION
* NEXT SIX LINES FOR 4 PTS/CYC WITH 8 BIT ADC WORDS *
          LDX     [0=2018        32, 24 DEC OR 20, 18 HEX
          STX     DTPTCH         DTPTCH, PTCYCH
          LDX     [DPSHT+192    *
          STX     PSHBCT        * DEFINE BOTTOM OF DATA COLLECTION AREA
          LDX     [DATA+92       *
          STX     DATAB          * BOTTOM OF STACKING AREA
SETFL     CLR     FLAG12        CLEAR FLAG FOR 8 BIT ADC WORDS
          LDAA    [E4            50HZ CHANGE OVER POINT
          LDAB    [CC4
          BSR     COMPAR
          BMI     ADC8
          INC     FLAG12        SET FLAG12 FOR 12 BIT ADC WORD SIZE
ADC8      RTS
PTSERR    LDAB    [0=83          DISPLAYS F3
          JSR     ERROR




** COMPARE PERIODS **
COMPAR    STAA    EXPA
          STAB    COEB
          LDAA    PIAB4         READ EXP
          ANDA    [0=70
          CMPA    EXPA
          BHI     GTHAN         ACC > M
          BCS     LTHAN         ACC < M
          LDAA    PIAA4          READ COEF
          CMPA    COEB
          BHI     GTHAN         ACC > M
          BCS     LTHAN         ACC < M
          CLRA                  * EQUAL
          RTS
LTHAN     LDAA    [0=80         * LESS THAN
          RTS
GTHAN     LDAA    [0=01         * GREATER THAN
          RTS


** NORMALIZE TRANSFORM RESULTS FOR 64, 16, OR 4 POINTS/CYCLE **
NORM      CLRB
          LDAA    PTSCYC        *
          CMPA    [0=40         *
          BEQ     NREND         * 64 PTS/CYC USED  NORMALIZATION NOT NEEDED
NORM0     ASLA                   *  DETERMINE NO. OF SHIFTS
          BMI     NORM1          *  NEEDED TO NORMALIZE
```

```
         INCB                    *  TRANSFCRMED VALUES,
         BRA      NORM0          *  STORE NO. IN NOSHF.
NORM1    STAB     NOSHF          *
         LDX      STORAR        LCAD ADRS OF STORAGE AREA FOR TRANSFORMED VALUES
         LDAA     [0=06         SET CH COUNTER
NLOOP1   LDAB     NOSHF         LCAD NO. OF SHIFT COUNTER
NLOOP2   BSR      SHIFTL         4 BYTE TIMES 2 MULT.
         DECB                   DEC NO. OF SHIFTS COUNTER
         BNE      NLCOP2
         JSR      INX4          SET INDEX TO NEXT VALUE IN DATA SET
         DECA                   DEC CHANNEL COUNTER
         BNE      NLCOP1
NREND    RTS


SHIFTL   ASL      3,X
         ROL      2,X
         ROL      1,X
         ROL      0,X
         RTS


** CONVERT BINARY RESULTS TC DISPLAYABLE FORM  **
CNVRT    LDAA     [0=06
         STAA     CHCNT
CNVLP    BSR      BCDCON
         TST      DIVTEN        TEST DIVID BY TEN FLAG
         BEQ      SKIPMV
         JSR      MVDEC         MCVE DECIMAL PT.
SKIPMV   JSR      INX4          POINT TO NEXT SET OF DECIMAL OUTPUT LOCATIONS
         STX      BCDOUT        *
         LDX      BNIN           * POINT TC NEXT SET OF BINARY
         JSR      INX4           * INPUT LOCATIONS
         STX      BNIN           *
         DEC      CHCNT
         BNE      CNVLP           LOOP 6 TIMES
         RTS


** MAIN ROUTINE PREPARES BINARY DATA FOR DISPLAY AS BCD VALUES **
BCDCON   BSR      BCDPT1        DO PART 1
         BSR      BCDPT2        DO PART 2
         RTS


** PREPARES BINARY VALUE FOR DISPLAY AS SIGNED 4 DIGIT BCD VALUE **
BCDCO2   BSR      BCDPT1        DO PART 1
         LDAB     [12           * SHIFT 4 BYTES 12 BITS RIGHT
         BSR      SHRLP         *
         LDAA     [0=0F          * BLANK LEADING ZERO
         STAA     01,X           * AND ALL DECIMALS
         LDAA     SRTCH2        *
         STAA     00,X          * LOAD SIGN
         LDAA     02,X           * BLANK SECOND ZERO IF PRESENT
         BITA     [0=F0          *
         BNE      BCOUT          *
         ORAA     [0=F0          *
         STAA     02,X           *
BCOUT    RTS                     FCRMAT (- XXXX)
```

```
** CONVERTS BINARY DATA TO BCD FOR DISPLAY (PART 1) **
BCDPT1    CLRB                NC MINUS SIGN
          LDX    BNIN         *
          LDAA   02,X         *  LOAD BINARY DATA, 4TH BYTE NOT USED
          STAA   D1           *LS BYTE
          LDAA   01,X         *
          STAA   D2           *
          LDAA   00,X         *
          STAA   D3           *MS BYTE
          BPL    POSIT          TEST FOR NEG BINARY VALUE
          COM    D3           * NEGATE NEG BINARY VALUE
          COM    D2           *
          COM    D1           *
          LDAB   [0=02          SET NEGATIVE SIGN
POSIT     STAB   SRTCH2         SAVE FOR LATER
          JSR    BINBCD       BINARY TO BCD ROUTINE
          LDX    BCDOUT
          LDAA   OUT1         * LSD
          STAA   03,X         *      STORE DECIMAL RESULTS
          LDAA   OUT2         *
          STAA   02,X         *
          LDAA   OUT3         *
          STAA   01,X         *
          LDAA   [0=40          SET 3RD DECIMAL (X X.X X X)
          LDAB   OUT4         *
          ANDB   [0=0F        *
          STAB   00,X         * MSD
          RTS

** CONVERTS BINARY DATA TO BCD FOR DISPLAY (PART 2) **
BCDPT2    BNE    BLARG          * NO LEADING ZERO
          LDAB   01,X         *
          ANDB   [0=F0        *
          BNE    BMED         * NEXT DIGIT IS NOT A ZERO  - BRANCH
          LDAB   01,X          *
          ANDB   [0=0F         *
          BNE    NOZERO        * TEST FOR LEADING ZERO
          ORAB   [0=0F        *  BLANK LEADING ZERO
          STAB   01,X         *
NOZERO    BRA    BSML              SMALLEST NO.
BLARG     BSR    SHR44         SHIFT RIGHT
          LSRA                 SHIFT DECIMAL
BMED      BSR    SHR44
          LSRA                 SHIFT DECIMAL
BSML      ORAA   01,X         *
          STAA   01,X         * ADD DECIMAL POINT
          LDAA   SRTCH2        *
          STAA   00,X          * LOAD SIGN
          RTS


SHR44     LDAB   [0=04        SHIFT 4 BYTES 4 PLACES RIGHT
SHRLP     LSR    00,X
          ROR    01,X
          ROR    02,X
          ROR    03,X
          DECB
          BNE    SHRLP
          RTS


** CALCULATES PHASE CORRECTION IN TENTHOUSANDTHS OF DEGREES **
```

```
** DUE TO TIME SAMPLING SKEW                                            **
PHSCOR   LDAB   [0=40
         LDAA   PTSCYC
PHLP     LSRB
         LSRA
         CMPA   [0=01
         BNE    PHLP
         LDAA   HRMNIC     *
         STAA   Y+1        *
         CLR    Y          * LOAD HARMONIC NO. INTO MULTIPLY POSITION
         STAB   XX+1
         CLR    XX
         JSR    MULT16
         LDX    U+2
         STX    Y
         LDX    [0=249F    * LOAD CONSTANT FOR MULTIPLY (0.9375 DEGREES)
         STX    XX         *
         JSR    MULT16
         LDX    U+2
         STX    PHS2
         LDAA   U+1
         STAA   PHS1
         LDX    [PHS1      * CONVERT BINARY PHASE TO BCD
         STX    BNIN       *
         LDX    [PHS10     *
         STX    BCDOUT     *
         JSR    BCDCON     *
         JSR    MVDEC        MOVE DECIMAL (1/10) RESULTS ARE IN DEGREES
         RTS

MVDEC    LDAA   01,X       ROUTINE MOVES DECIMAL PT TO LEFT
         ANDA   [0=F0
         ASLA              MOVE DEC PT LEFT
         LDAB   01,X       *
         ANDB   [0=0F      * RECOMBINE
         ABA               *
         STAA   01,X        RELOAD DECIMAL PT
         RTS

DISPHC   LDX    [PHS10      DISPLAY PHASE CORRECTION
         JSR    LCDDIS
         RTS



** HARMONIC SIZE CHECK **
HRMCK    LDAB   HRMNIC
         LDAA   PTSCYC
         LSRA
         CBA
         BCC    HRMOK1
         LDAB   [0=B2
         JSR    ERROR
HRMOK1   RTS
```

```
** DISPLAYS ERROR SYMBOL (ACCB) AND HALTS PROGRAM **
ERROR      JSR    CLRDIS
           STAB   PIAB1
ESELF      BRA    ESELF
           RTS



** ENTER HARMONIC NO.   **
HRMSLT     LDAB   HARM10
           JSR    DISDG2
           JSR    ENTDS2
           STAA   HARM10
           TAB
           JSR    DISDG2
           JSR    BCDBN2
           STAB   HRMNIC
           JSR    HRMCK
           JSR    PHSCOR        CALCULATE PHASE CORRECTION
           RTS



** DISPLAYS TWO DIGITS IN ACC B ON LCD FORMAT (B B 0 2.B B) B=BLANK **
DISDG2     JSR    CLRDIS
           STAB   PIAB1
           LDAA   [0=2F
           STAA   PIAA2
           RTS



** CONVERTS 2 DIGIT BCD VALUE IN ACCB TO 8 BIT BINARY VALUE IN B **
BCDBN2     TBA
           ANDB   [0=0F
           LSRA
           LSRA
           LSRA
           LSRA
           BEQ    BCDRTS
BCDLP      ADDB   [0=0A
           DECA
           BNE    BCDLP
BCDRTS     RTS



** ENTERS AND DISPLAYS TWO DIGITS   **
ENTDS2     JSR    SCNKEY
           BEQ    LASTDG
STROVR     JSR    NCCDE
           LDX    [PIAB1
           LDAB   [0=FF
           STAB   0,X
           JSR    STDIG1
           BEQ    LASTDG
           JSR    NCCDE
           LDX    [PIAB1
           JSR    STDIG2
           BEQ    LASTDG
```

```
         BRA     STROVR
LASTDG   LDAA    PIAB1
         RTS


* ROUTINE SETS THE PRE AND POST COUNTERS ON TIMING BOARD *
PTSCON   LDAB    [0≡00        64 PTS CONTROL BITS
         LDAA    [0≡10        = 16 PTS/CYC
         CMPA    PTSCYC
         BCS     PCSET         (64)
         LDAB    [0≡04         16 PTS  CONTROL BITS
         CMPA    PTSCYC
         BEQ     PCSET         (16)
         LDAB    [0≡08         4 PTS CONTROL BITS
PCSET    LDAA    PIAB4        *  SAVE OTHER BITS
         ANDA    [0≡F3        *  AND ADD PTS/CYC
         ABA                  *  CONTROL BITS
         STAA    PIAB4        * (CONTROL BITS ARE B2,B3 )
         RTS

**  DISPLAYS AND ALLOWS ENTERING OF ROTACC         **
** CONTROLS ARCTAN ACCURACY   TABLE FOLLOWS **
** ROTACC = 3 RESULTS IN .0 ;8 DEG ACCURACY, TIME 1 SEC CHAN. **
** ROTACC = 7 RESULTS IN .00175 DEG ACCURACY, TIME 16 SEC/CHAN **
CROTAC   LDX     [ROTACC
         BSR     DIS2
         LDAB    ROTACC
         CMPB    [0≡02
         BHI     CROT1
         LDAB    [0≡03
CROT1    CMPB    [0≡07
         BLS     CROT2
         LDAB    [0≡07
CROT2    STAB    ROTACC
         JSR     DISDG2
         RTS




* DISPLAYS AND ALLOWS ENTERING OF CHANNEL NO.   *
NCHNLS   LDX     [NCHPR       * LOAD ADRS OF NO. OF CH OPERATED ON AND PRINTED
         BSR     DIS2
         LDAB    NCHPR
         DECB                 * TEST FOR CH NO. OUTSIDE RANGE OF 1 TO 6
         CMPB    [05          *
         BHI     NCHLD6       *
         RTS                  OK WITHIN RANGE
NCHLD6   LDAB    [06          OUTSIDE RANGE LOAD IN 6
         STAB    NCHPR
         RTS

* DISPLAYS AND ALLOWS ENTERING OF STATION NO.   *
STATNO   LDX     [NSTN        * LOAD ADRS OF STATION NO.
         BSR     DIS2         DISPLAY AND ENTER VALUES
         RTS


* DISPLAY AND ALLOWS ENTERING OF RUN NO.         *
RUNNO    LDX     [NRUN
```

```
          BSR     DIS2
          RTS

** GENERAL ROUTINE FOR DISPLAYING AND ENTERING 2 DIGIT NO. **
DIS2      STX     TABLEB
          LDAB    00,X          LOAD VALUE TO BE DISPLAYED
          JSR     DISDG2         DISPLAY 2 DIGITS
          JSR     ENTDS2        ENTER 2 DIGITS
          LDX     TABLEB        *
          STAA    00,X          * STORE NEW VALUE
          TAB
          JSR     DISDG2
          RTS



** TURN PRINTER POWER ON  **
PWRON     LDAA    PIAB2         * TURN PRINTER POWER ON
          ORAA    [0=50
          STAA    PIAB2         *
          LDX     [0=3A02        *  DELAY 200 MS
          JSR     DELAY3         * ( AT 1000KHZ CLOCK )
          RTS


** CONTROLS PARAMETER PRINT SEQUENCE  **
MPRINT    BSR     PWRON         TURN PRINTER POWER ON
          BSR     BLANK
          BSR     DFTDIS        PRINT COS AND SIN
          LDAB    HARM10        *
          JSR     DISDG2        * PRINT HARMONIC NO.
          BSR     PRNT          *
          JSR     DISPER         * PRINT PERIOD
          BSR     PRNT           *
          JSR     DISCYC        * PRINT NO. CYCLES
          BSR     PRNT          *
          JSR     DSTNRN        * PRINT STATION AND RUN NO.
          BSR     PRNT          *
          BSR     BLANK
          BSR     PWROFF        TURN PRINTER POWER OFF
          RTS


** TURN PRINTER POWER OFF  **
PWROFF    LDAA    PIAB2         *
          ANDA    [0=BF         * TURN POWER OFF
          STAA    PIAB2         *
          RTS


** PRINTS VALUE DISPLAYED ON LCD **
PRNT      LDAA    PIAB2         *
          ORAA    [0=20         * PRINT LINE HIGH = PRINT
          STAA    PIAB2         *
          LDAA    PIAB2          *
          ANDA    [0=DF          * PRINT LINE LOW
          STAA    PIAB2          *
BUSY      LDAA    PIAB2         *
          ANDA    [0=08         * TEST BUSY LINE
          BNE     BUSY          *
          RTS


** ADVANCES PAPER ON PRINTER WITHOUT PRINTING **
```

```
BLANK     JSR    CLRDIS      *
          BSR    PRNT        * PRINT A BLANK
          RTS


** ROUTINE PRINTS VALUES POINTED AT BY INDEX REG. **
TPRINT    JSR    LCDDIS
          JSR    PRNT
          RTS


** ROUTINE PRINTS COS AND SIN TRANSFORM RESULTS **
DFTDIS    LDAA   NCHPR       *
          CMPA   [01         * SELECT NO OF CH. TO BE PRINTED
          BEQ    CHAN1       *
          CMPA   [02         *
          BEQ    CHAN2       *
          CMPA   [03         *
          BEQ    CHAN3       *
          CMPA   [04         *
          BEQ    CHAN4       *
          CMPA   [05         *
          BEQ    CHAN5       *
CHAN6     LDX    [PHSSTR+20
          BSR    TPRINT
          LDX    [AMPSTR+20
          BSR    TPRINT
CHAN5     LDX    [PHSSTR+16
          BSR    TPRINT
          LDX    [AMPSTR+16
          BSR    TPRINT
CHAN4     LDX    [PHSSTR+12
          BSR    TPRINT
          LDX    [AMPSTR+12
          BSR    TPRINT
CHAN3     LDX    [PHSSTR+8
          BSR    TPRINT
          LDX    [AMPSTR+8
          BSR    TPRINT
CHAN2     LDX    [PHSSTR+4
          BSR    TPRINT
          LDX    [AMPSTR+4
          BSR    TPRINT
CHAN1     LDX    [PHSSTR
          BSR    TPRINT
          LDX    [AMPSTR
          BSR    TPRINT
          RTS


** ROUTINE TRANSFORMS AND PRINTS ALL HARMONICS  **
**  BEGINNING AT HARMONIC NO. HRMNIC           **
PRTALL    BSR    TRSET
          BEQ    POUT1       IF ZERO RETURN
          BSR    HMSET
          BRA    PRTALL

** ROUTINE TRANSFORMS AND PRINTS EVERY OTHER HARMONIC  **
** BEGINNING AT HARMONIC NO. HRMNIC                    **
PRTEVO    BSR    TRSET
          DECA
          BLE    POUT1       IF -1 OR 0 RETURN
          BSR    HMSET
```

```
        BRA     PRTEVO

TPSET   JSR     HRMCK
        JSR     SINCOS
        LDAA    HRMNIC
        DECA
        RTS

HMSET   STAA    HRMNIC
        JSR     BNBCD2
        STAA    HARM10
POUT1   RTS


** CONVERTS 1 BYTE BINARY TO 2 DECIMAL  (MAX VALUE IS 99)
BNBCD2  CLRB
ADD10   ADDB    [0=10
        SUBA    [0=0A
        BCC     ADD10
        SUBB    [0=10
        ADDA    [0=0A
        ABA
        RTS


** 2S COMP 16 TIMES 16 BIT MULTIPLY WITH 32 BIT PRODUCT WITH 2 SIGN BITS **
**   XX,XX+1          IS     MULTIPLICAND        NOT DESTROYED
**   Y,Y+1            IS     MULTIPLIER
***   U,U+1,U+2,U+3   IS     PRODUCT
MULT16  LDX     [0=0005
        CLRA
MLP1    STAA    XX+1,X
        DEX
        BNE     MLP1
        LDX     [0=0010
MLP2    LDAA    Y+1
        ANDA    [01
        TAB
        EORA    FF
        BEQ     SHIFT
        TSTB
        BEQ     ADD
        LDAA    U+1
        LDAB    U
        SUBA    XX+1
        SBCB    XX
        STAA    U+1
        STAB    U
        BRA     SHIFT
ADD     LDAA    U+1
        LDAB    U
        ADDA    XX+1
        ADCB    XX
        STAA    U+1
        STAB    U
SHIFT   CLR     FF
        ROR     Y
        ROR     Y+1
        ROL     FF
        ASR     U
        ROR     U+1
        ROR     U+2
```

```
          ROR     U+3
          DEX
          BNE     MLP2
          RTS



** 3 BYTE BINARY TO 7 DIGIT BCD CONVERSION  INPUT D3,D2,D1          **
** OUTPUT OUT4,OUT3,OUT2,OUT1  UNSIGNED VALJES ONLY                 **
BINBCD    LDX     [C106         1000000
          BSR     BINCON        LCAD CONSTANT INTO C3
BIN1      BSR     BINSUB        SUETRACT
          BCC     BIN1
          BSR     BINADD        ADC
          STAA    OUT4
          LDX     [C105         100000
          BSR     BINCCN        LCAD CONSTANT INTO C3
BIN2      BSR     BINSUB
          BCC     BIN2
          BSR     BINSET
          STAA    OUT3
          LDX     [C104
          BSR     BINCON
BIN3      BSR     BINSUB
          BCC     BIN3
          BSR     BINADD
          ADDA    OUT3
          STAA    OUT3
          LDX     [C103         1000
          BSR     BINCCN
BIN4      BSR     BINSUB
          BCC     BIN4
          BSR     BINSET
          STAA    OUT2
          LDX     [C102         100
          BSR     BINCCN
BIN5      BSR     BINSUB
          BCC     BIN5
          BSR     BINADD
          ADDA    OUT2
          STAA    OUT2
          CLR     SUBT
          LDAA    [0=0A
          STAA    C1
BIN6      BSR     BINSUB
          BCC     BIN6
          BSR     BINSET
          ADDA    D1
          STAA    OUT1
          RTS

** COMPONENT OF BCD CONVERSION ROUTINE **
BINCON    CLR     SUBT
          LDAA    00,X
          STAA    C3            MS BYTE
          LDAA    01,X
          STAA    C2
          LDAA    02,X          LS BYTE
          STAA    C1
```

```
          RTS

** COMPONENT OF BCD CONVERSION FOUTINE **
BINSUB    INC      SUBT
          BSR      SUB3
          RTS

** COMPONENT OF BCD CONVERSION ROUTINE **
BINADD    BSR      ADD3
          LDAA     SUBT
          DECA
          RTS

** COMPONENT OF BCD CONVERSION ROUTINE **
BINSET    BSR      ADD3
          LDAA     SUBT
          DECA
          ASLA
          ASLA
          ASLA
          ASLA
          RTS

** CONSTANT TABLE FOR BINARY TO BCD CONVERSION **
C106      CON      0≡CF,0≡42,C≡40                    1000000
C105      CON      0≡01,0≡86,0≡A0                    100000
C104      CON      0≡C0,0≡27,0≡10                    1000C
C103      CON      0≡00,0≡03,0≡E8                    1000
C102      CON      0≡00,0≡00,0≡64                    100

** 3 BYTE ADD FOR BCD CON\ERSICN ROUTINE **
ADD3      LDAA     D1
          ADDA     C1
          STAA     D1
          LDAA     D2
          ADCA     C2
          STAA     D2
          LDAA     D3
          ADCA     C3
          STAA     D3
          RTS

** 3 BYTE SUBTRACT FOR BCD CONVERSION ROUTINE **
SUB3      LDAA     D1
          SUBA     C1
          STAA     D1
          LDAA     D2
          SBCA     C2
          STAA     D2
          LDAA     D3
          SBCA     C3
          STAA     D3
          RTS

* ACQUIRES. TRANSFORMS AND PRINTS ONE HARMONIC    *
```

```
* CALLED BY KEYS  RUN 1                                    *
ATPONE    BSR    TESTB4       * IF SWITCH 4 SET DO NOT ACQUIRE DATA
          BNE    KYOUT1       *
          BSR    ACQDAT        ACQUIRE A DATA SET
          JSR    SINCOS       TRANSFORM AND PRINT ONE HARMONIC
          BSR    TESTB3       * READ SWITCH 3
          BNE    ATPONE       * IF SET REPEAT SEQUENCE
KYOUT1    RTS


* ACQUIRES, TRANSFORMS AND PRINTS EVERY OTHER HARMONIC *
*   CALLED BY KEYS   RUN  4                            *
ATPEVO    BSR    TESTB4       * IF DATA PROTECT IS ON RTN
          BNE    KYOUT4       *
          BSR    ACQX          ACQUIRE A DATA SET
          JSR    PRTEVO       TRANSFORM AND PRINT RESULTS
          BSR    HRSET        * RESET HARMONIC NO. AND TEST
          BNE    ATPEVO       * REPEAT SWITCH ( 3 )
KYOUT4    RTS


* ACQUIRE, TRANSFORM AND PRINT ALL HARMONICS   *
*  CALLED BY KEYS  RUN  7                       *
ATPALL    BSR    TESTB4       * IF DATA PROTECT IS ON RTN
          BNE    KYOUT7       *
          BSR    ACQX          ACQUIRES A DATA SET
          JSR    PRTALL       TRANSFORM AND PRINT RESULTS
          BSR    HRSET        * RESET HARMONIC NO. AND TEST
          BNE    ATPALL       * REPEAT SWITCH ( 3 )
KYOUT7    RTS


ACQX      LDAA   HRMNIC       * SAVE HARMONIC NO.
          STAA   HSAVE        *
          BSR    ACQDAT        ACQUIRE A DATA SET
          RTS

HRSET     LDAA   HSAVE        * RESET HARMONIC NO.
          JSR    HMSET        *
          BSR    TESTB3        TEST REPEAT SWITCH
          RTS

* TEST SWITCH 1   *
TESTB1    LDAA   PIAB3        READ MODE SWITCH
          ANDA   [0=10        MASK OUT ALL BJT SWITCH NO.1
          RTS

* TEST SWITCH 4    *
TESTB4    LDAA   PIAB3        READ MODE SWITCH
          ANDA   [0=80         MASK OUT ALL BUT SWITCH NO. 4
          RTS

*  TEST SWITCH 3   *
TESTB3    LDAA   PIAB3        READ MODE SWITCH
          ANDA   [0=40         MASK OUT ALL BUT SWITCH NO. 3
          RTS


** CIRCULATE ACC B 4 BITS INTO ACC A    **
RORBA     RORB
          RORA
          RORB
          RORA
```

```
         RORB
         RORA
         RORB
         RORA
         RTS


** MASTER ACQUISITICN ROUTINE FCR 64 16 4 PTS/CYC  **
ACQDAT   JSR    CLFDIS       *
         LDAA   CYCAVG       *
         STAA   PIAA1          DISPLAY NO. OF CYCLES AVERAGEC
         LDAA   PIAB4        * READ PERIOD EXP
         ANDA   [0=70        *
         LDAB   PIAA4        * READ PERIOD COEFFCIENT
         BSR    RORBA       CIRCULATE E 4 BIT INTO A
         ORAB   [0=A0        * SET 2ND AND 4TH DECIMAL PT
         STAB   PIAA2
         STAA   PIAB1
         LDAB   FLAG12        EQ 1 FOR 12 BIT ADC WORDS
         BEQ    BRAC1         * SELECT ACQUSIT. ROUTINE AS FN OF ADC WORD SIZE
         JSR    DTAQ64        ROUTINE FCR 64 OR 16 PTS PER CYCLE
         BRA    BRAC2
BRAC1    JSR    DATAQ4        ROUTINE FCR 4 PTS/CYC
BRAC2    LDAB   [20          *
ADLP1    JSR    BEEP         * CALL BEEP 20 TIMES
         DECB
         BNE    ADLP1        *
         JSR    CLRDIS         CLEAR DISPLAY
         LDAA   NRUN         *
         ADDA   [01          * THIS ADD SETS HALF CARRY FLAG
         DAA                 *
         STAA   NRUN         * INCREMENT RUN NUMBER
NOACQ    RTS


** MASTER ROUTINE FOR 64 OR 16 PTS / CYCLE  **
** WITH 12 BIT ADC WORDS                    **
DTAQ64   LDX    [1536        * NO. BYTES TO BE CLEARED
         STX    SRTCH1       *
         LDX    [DATA         * LOCATION OF DATA TO BE CLEAREC
         JSR    CLRM2       CLEAR MEMORY SECTION
         LDX    NCYCLE       * SET NO. CF CYCLES TO BE AVERAGEC
         STX    SRTCH1       *
         BSR    AQ1664        ACQUIRE DATA
         LDX    [0384        *  =64*6  (64 PTS/CYC)
DT64     STX    SRTCH1      *NO. OF 4 BYTE WORDS TO BE NORMALIZED
         LDX    [DATA       *DEFINE LOCATION OF DATA
         STX    DATPT        *
         JSR    ADJUST         NORMALIZE
         RTS


*** ACQUISITION ROUTINE FOR 16 AND 64 POINTS PER CYCLE   ***
** WITH 12 BIT ADC DATA VALUES.                          ***
AQ1664   CLR    PIAA5       SELECT CH 0
         LDAA   PIAB3       READ MODE SWITCH
         LDAB   PIAB4       CLEAR CYCLE PULSE BY READING PIA
         LDAB   PIAB5       CLEAR SAMPLE PULSE
         ANDA   [0=20       MASK ALL BUT MODE 2 BIT
         BEQ    AQQ2        IF MODE 2 NOT SET LOOK FOR CYCLE PULSE
```

```
AQQ1      LDAA   CPIAB5     * WAIT FOR SAMPLE PULSE FOR CH 0 (DOWN GOING)
          BPL    AQQ1       *
          BRA    AQQ3       SKIP WAIT FOR CYCLE PULSE
AQQ2      LDAA   CPIAB4     * WAIT FOR CYCLE PULSE
          BPL    AQQ2       * (OCCURS SAME TIME AS SAMPLE PULSE CH 0 )
AQQ3      LDAA   PIAB5      CLEAR SAMPLE PULSE
          INC    PIAA5         SELECT CH 1
CYLOOP    LDX    SRTCH1     4  (MACHINE CYCLES) TEST CYCLE COUNTER
          BEQ    AQFIN      4   IF ZERO RTS
          DEX               4   * DEC AND STORE CYCLE COUNTER
          STX    SRTCH1     5   *
          LDX    [DATA      3   ADRS OF 1ST VALUE IN DATA STACKING AREA
          LDAB   PTSCYC     3   SET LOOP COUNTER B TO NO. PTS/CYC 16 OR 64
CHLOOP    BSR    ACQSTK     101 WAIT FOR SAMPLE PULSE CH 1,READ CH 0
          INC    PIAA5      6   SELECT CH 2
          BSR    ACQSTK     101 WAIT FOR SAMPLE PULSE CH 2, READ CH 1
          INC    PIAA5      6   SELECT CH 3
          BSR    ACQSTK     101 WAIT FOR SAMPLE PULSE CH 3, READ CH 2
          INC    PIAA5      6   SELECT CH 4
          BSR    ACQSTK     101 WAIT FOR SAMPLE PULSE CH 4, READ CH 3
          INC    PIAA5      6   SELECT CH 5
          BSR    ACQSTK     101 WAIT FOR SAMPLE PULSE CH 5, READ CH 4
          CLR    PIAA5      6   SELECT CH 0
          BSR    ACQSTK     101 WAIT FOR SAMPLE PULSE CH 0, READ CH 5
          INC    PIAA5      6   SELECT CH 1
          DECB              2   DEC PTS/CYC COUNTER (16 OR 64)
          BEQ    CYLOOP     4   IF ZERO START NEW CYCLE
          BRA    CHLOOP     4   IF NOT 0 GO THROUGHT CH SEQUENCE ONCE MORE
AQFIN     RTS


** ROUTINE ACQUISITION AND STACK - STACKS ONE 12 BIT WORD FROM ADC INTO **
** FOUR 8 BIT BYTES. REQUIRES 101 MACHINE CYCLES TO CALL AND RETURN.
ACQSTK    LDAA   CPIAB5     4  WAIT FOR SAMPLE PULSE FOR CH 1 (DOWN GOING)
          BPL    ACQSTK     4
          LDAA   PIAA5      4  READ LS BYTE FROM ADC
          ANDA   [0=F0      2  MASK OUT CH ADRS
          ADDA   03,X       5  * STACK IN MEM
          STAA   03,X       6  *
          LDAA   PIAB5      4  READ MS BYTE FROM ADC
          ADCA   02,X       5   STACK IN MEM
          STAA   02,X       6
          LDAA   01,X       5  ADD CARRY TO 3RD BYTE
          ADCA   [00        2
          STAA   01,X       6
          BCC    ACSK1      4  BRANCH IF CARRY CLEAR
          INC    00,X       7  OTHERWISE INC MS BYTE
ACSK1     INX               4  *
          INX               4  * POINT INDEX TO NEXT STACKING LOCATION
          INX               4  *
          INX               4  *
          RTS


** MASTER ROUTINE FOR 4 POINTS PER CYCLE **
DATAQ4    JSR    NOCALL     SET NO. OF CALLS TO ACQUS1
          LDX    [384       * NO OF BYTES TO BE CLEARED
          STX    SRTCH1     *
          LDX    [DATA       STARTING ADRS
          JSR    CLRM2      CLEAR STACKING AREA
DATLP     JSR    ACQUS1      ACQUIRIES 8 CYCLES OF DATA
```

```
         BSR     STACK         NEW STACK DATA
         LDX     NTACQ         * DEC COUNTER
         DEX                   *
         STX     NTACQ         *
         BNE     DATLP
         LDX     [0096         *
         STX     SRTCH1        * NO. OF 4 BYTE WORDS TO BE NORMALIZED
         LDX     [DATA          *
         STX     DATPT          * DEFINE LOCATION OF DATA
         JSR     ADJUST         ADJUST DATA FOR NO. OF PTS STACKED
         RTS


** ROUTINE FOR STACKING 8 CYCLES OF DATA AT 16 OR 4 PTS/CYC **
STACK    LDAB    [08           * SET LOOP1 COUNTER (8 CYCLES OF DATA)
         STAB    SRTCH2        *
         STS     STORAR         SAVE STACK POINTER
         LDS     [DPSHT         SET STACK POINTER TO TOP OF NONSTACKED DATA
STKLP1   LDX     DATAB         RESET POINTER TO DATA STACKING AREA
         LDAB    PTCYCH        RESET LOOP2 COUNTER (6 CH AT 4 OR 16 PTS/CYC)
STKLP2   PULA                   PULL DATA FROM STACK
         ADDA    02,X          * ADD 8 BIT BYTE TO FOUR BYTE WORD
         STAA    02,X          * LS BYTE (03,X) IS NOT USED
         LDAA    01,X          *.
         ADCA    [00           *
         STAA    01,X          *
         BCC     STKB          *
         INC     00,X          *
STKB     DEX                    * SET TO NEXT DATA STACKING WORD
         DEX                    *
         DEX                    *
         DEX                    *
         DECB                  *
         BNE     STKLP2        * LOOP2
         DEC     SRTCH2         *
         BNE     STKLP1         * LOOP1
         LDS     STORAR        RESTORE STACK POINTER
         RTS



** ROUTINE NORMALIZES DATA WITH NO. OF CYCLES STACKED **
ADJUST   LDAB    CYCAVG         EXP OF 2  BCD
         JSR     BCDBN2         CONVERT TO BINARY
         LDAA    [16           *
         SBA                   * SET NO. OF SHIFT NEED TO NORMALIZE
         STAA    NOSHF         *
ADJLP1   LDAB    NOSHF         RESET NO. OF SHIFT COUNTER
         LDX     DATPT         LOAD POINTER TO DATA
ADJLP2   JSR     SHIFTL        SHIFT 4 BYTES 1 BIT LEFT
         DECB                  DEC NOSHF COUNTER
         BNE     ADJLP2
         LDAA    00,X           * REMOVE DC OFFSET TO CREATE
         ADDA    [0=80          * 2S COMPLIMENT NUMBERS
         STAA    00,X           *
         JSR     INX4          SELECT NEXT VALUE TO BE SHIFTED
         STX     DATPT         *
         LDX     SRTCH1         * DEC WORD COUNTER
         DEX                    *
         STX     SRTCH1         *
         BNE     ADJLP1
         RTS
```

```
*** ROUTINE SETS NO. OF CALLS TC ACQUS1 BASED ON CYCAVG ***
*** THE MINIMUM NO. OF CYCLES AVERAGED IS 8            ***
NOCALL   LDAB   CYCAVG     CYCAVG IS EXP OF 2  IN BCD
         JSR    BCDBN2     CONVERT TO BINARY
         SUBB   I03        SUE OUT 3 FOR MIN NO. OF CYCLES
         BPL    NOCA1      IF RESULT IS POSITIVE CHANGE NOTHING
         LDAB   I03        * OTHERWISE RESET TO 3
         STAB   CYCAVG     *
         TBA
         JSR    SETCN2     RESET CYCLE COUNTER TO 2 EXP 3
         CLRB              * FOR N=3
NOCA1    LDX    I0001       *
         STX    NTACQ       * SETS COUNTER FOR NO. OF TIMES
NOLOOP   DECB               * ACQUS1 IS CALLED
         BMI    RETRN       *
         ASL    NTACQ+1     *
         ROL    NTACQ       *
         BRA    NOLOOP      *
RETRN    RTS
```

```
*** ROUTINE FOR ACQUIRING 8 BIT ADC DATA VALUES AT 16 OR 4 POINTS PER CYCLE ***
*** PUSHES DATA BYTES IN TO DATA STORAGE AREA       ***
ACQUS1   LDAB   DTFTCH     SET LOOP COUNTER
         STS    SRTCH1     STORE STACK POINTER
         LDS    PSHBOT     SET STACK POINTER TO START OF DATA STORAGE AREA
         CLR    PIAA5      SELECT CHANNEL 0
         LDAA   PIAB4      CLEAR CYCLE PULSE BY READING PIA
         NOP               A FUNCTIONAL NOP
AQLP0    LDAA   CPIAB4     WAIT FOR CYCLE PULSE. SAME AS SMPL PULSE CH 0
         BPL    AQLP0      NOTE TEST, SET PIA ACCORDINGLY
         LDAA   PIAB5      CLEAR SAMPLE PULSE
         INC    PIAA5      SELECT CH 1
AQLP1    LDAA   CPIAB5     * WAIT FOR SAMPLE PULSE (DOWN GOING)
         BPL    AQLP1      * FOR CH 1
         LDAA   PIAB5      * READ DATA FOR CH 0
         INC    PIAA5      * SELECT CH 2
         PSHA              * STORE DATA FOR CH 0.
AQLP2    LDAA   CPIAB5      * WAIT FOR SAMPLE PULSE
         BPL    AQLP2       * FOR CH 2
         LDAA   PIAB5       * READ DATA FOR CH 1
         INC    PIAA5       * SELECT CH 3
         PSHA               * STORE DATA FOR CH 1
AQLP3    LDAA   CPIAB5     * WAIT FOR SAMPLE PULSE  CH 3
         BPL    AQLP3      *
         LDAA   PIAB5      * READ CH 2
         INC    PIAA5      * SELECT CH 4
         PSHA              * STORE DATA CH 2
AQLP4    LDAA   CPIAB5      * WAIT FOR SAMPLE PULSE  CH 4
         BPL    AQLP4       *
         LDAA   PIAB5       * READ CH 3
         INC    PIAA5       * SELECT CH 5
         PSHA               * STORE DATA CH 3
AQLP5    LDAA   CPIAB5     * WAIT FOR SAMPLLE PULSE  CH 5
         BPL    AQLP5      *
         LDAA   PIAB5      * READ CH 4
         CLR    PIAA5      * SELECT CH 0
         PSHA              * STORE DATA CH 4
AQLP6    LDAA   CPIAB5      * WAIT FOR SAMPLE PULSE   CH 0
         BPL    AQLP6       *
```

```
         LDAA    PIAB5        * READ CH 5
         INC     PIAA5        * SELECT CH 1
         PSHA                 * STORE DATA CH 5
         DECB                  *
         BNE     AQLP1         * LOOP
         LDS     SRTCH1       RESTORE STACK POINTER
         RTS



** ROUTINE DISPLAYS VOLTAGE ON SELECTED CHANNEL IN MILLIVOLTS **
** RANGE +/- 5000 MV
VOLTMT   JSR     CLRDIS
         JSR     RELESE      WAIT FOR RELEASE OF KEY
KEYFND   JSR     RWSLCT      (SCNKEY AFTER RELESE)
         BEQ     VLTOUT       RTS IF RTN IS FOUND
         JSR     NCODE       *
         ANDA    [0=0F       *
         DECA                *
         STAA    PIAA5       * SELECT CHANNEL
         LDAA    PIAB5         CLEAR SAMPLE PULSE BY READING PIAB5
         JSR     RELESE      WAIT FOR RELEASE OF KEY
VLP1     JSR     KEYQ        LOOK FOR ANY PRESSED KEY
         BNE     KEYFND      IF FOUND DECODE KEY
         LDAA    CPIAB5      *WAIT FOR SAMPLE PULSE
         BPL     VLP1        *
         LDAA    PIAB5       MS 8 BITS
         LDAB    PIAA5       LS 4 BITS
         ANDB    [0=F0        MASK CH ADRS
         ADDA    [0=80        REMOVE OFFSET (INVERT SIGN BIT)
         STAA    Y           *
         STAB    Y+1         *
         BSR     MVDIS       CONVERT TO MILLIVOLTS THEN DISPLAY
         BRA     VLP1
VLTOUT   RTS


* CONVERTS Y TO MILLIVOLTS THEN DISPLAYS RESULT ON LCD  *
MVDIS    LDX     [0=4C55      * SETUP MULTIPLY  (19541)
         STX     XX           * FOR MV RESULT
         JSR     MULT16
         LDX     [U
         JSR     SHIFTL        * TIMES 2.
         STX     BNIN
         LDX     [Y            (USES Y,Y+1,XX,XX+1)
         STX     BCDOUT
         JSR     BCDCO2      CONVERT TO BCD
         JSR     LCDDIS      DISPLAY
         RTS


** SCANS PREVIOUSLY ACQUIRED DATA FOR ABSOLUTE MAX VALUE ON SELECTED CHAN *
MAXSGN   JSR     CLRDIS
MAXLP    JSR     SCNKEY      SCAN KEY PAD FOR PRESSED KEY
         BEQ     MXOUT       TEST FOR RTN KEY
         JSR     NCODE       *
         ANDA    [0=0F       * DETERMINE CHANNEL NO.
         STAA    CHCNT       STORE CH NO.
         BSR     FNDMAX       FIND MAXIMUM VALUE
         BSR     MVDIS       CONVERT TO MILLIVOLTS AND DISPLAY
         BRA     MAXLP        LOOP FOR NEXT KEY
MXOUT    JSR     CLRDIS
```

```
         RTS

** SCANS ONE CHANNELS DATA FOR ABSOLUTE MAX VALUE - CALLED BY MAXSGN  **
FNDMAX   CLR    Y
         CLR    Y+1
         CLR    SRTCH3      CLEAR SIGN FLAG 2
         LDAA   PTSCYC      *
         STAA   CNTINC      * SET COUNTER FOR NO OF DATA POINTS
         LDX    [DATA-4     BACK SET POINTER TO DATA
         LDAB   CHCNT       READ CHANNEL NO.
CHNLP    BSR    INX4        *
         DECB               * SET POINTER TO FIRST DATA VALUE OF SELECTED CH
         BNE    CHNLP       *
NXTV     CLR    SRTCH2      CLEAR SIGN FLAG 1
         LDAB   01,X         * LOAD IN SELECTED DATA VALUE
         LDAA   00,X         * MS BYTE
         BPL    PLUS
         INC    SRTCH2      SET SIGN FLAG
         COMA               * NEGATE VALUE SO THAT ALL VALUES USED
         COMB               * ARE POSITIVE
PLUS     STAA   U             SAVE ACC A
         CMPB   Y+1          *
         SBCA   Y            *
         LDAA   U            * COMPARE SELECTED VALUE WITH LAST
         BCS    LESS         * STORED LARGEST VALUE, STORE LARGEST VALUE
         STAA   Y            *
         STAB   Y+1          *
         LDAA   SRTCH2      * SAVE SIGN FLAG OF LARGEST VALUE
         STAA   SRTCH3      *
LESS     BSR    INX24        * SET POINTER TO NEXT DATA VALUE  24 BYTES DOWN
         DEC    CNTINC      DEC NO. DATA POINTS COUNTER
         BNE    NXTV        LOOP BACK FOR NEXT COMPARISON
         LDAA   SRTCH3      *
         BEQ    FDOUT       * RESTORE SIGN TO LARGEST VALUE
         COM    Y           *
         COM    Y+1         *
FDOUT    RTS


INX4     INX      INCREMENTS INDEX REG FOUR TIMES
         INX
         INX
         INX
         RTS

* INCREMENT INDEX REGISTER 24 TIMES *
INX24    LDAB   [24
INXLP    INX
         DECB
         BNE    INXLP
         RTS


**  DUMPS 6 CHANNELS OF DATA TO CHART PAPER                          **
** VALUES ARE SENT TO THE DAC AT A RATE OF 30 VALUES PER SECOND  **
** DATA RATE IS SET BY VARIABLE SRTCH1 AND ROUTINE DELAY4          **
CHARTD   LDX    [0=101A     * 4122 DECIMAL
         STX    SRTCH1      *  FOR DELAY OF 33,000 MACHINE CYCLES
         JSR    ZEROL         START WITH ZERO VOLT LEVEL
         BSR    SNDANA
         RTS
```

```
** DISPLAYS 6 CH DATA ON SCOPE WHILE SCANNING KEY PAD FOR RTN AND **
** NO. FOR POSITIONING TRIGGER PULSE IN ONE OF 6 CH POSITIONS     **
OSCDIS    LDX     [0=0000
          STX     SRTCH1      * SET DELAY FOR ROUTINE DELAY4
          JSR     RELESE       WAIT FOR RELEASE OF KEY
FNDKEY    JSR     RWSLCT      ( SCNKEY AFTER RELESE )
          BEQ     OSCOUT      RTS IF RETURN KEY FOUND
          JSR     NCODE
          ANDA    [0=0F
          STAA    SRTCH5      STORE POSITION OF SCOPE TRIGGER PULSE
          JSR     RELESE
OSCLP     BSR     SNDANA      SEND 6 CH OF DATA TO DAC
          JSR     KEYQ        29 * LOOK FOR ANY PRESSED KEY
          BNE     FNDKEY      4  IF KEY FOUND JUMP OUT OF LOOP AND DECODE
          BRA     OSCLP       4
OSCOUT    RTS


** SENDS 6 CHANNELS OF STORED DATA TO 8 BIT DIGITAL TO ANALOG CONVERTER **
** DATA RATE IS DETERMINED BY ROUTINE DELAY4 AND SRTCH1.                **
** SRTCH5 DETERMINES POSITION OF SCOPE TRIGGER PULSE                    **
** WITH SRTCH1=0  64 POINTS TAKES 3.5 MILLISECONDS, CRT REFRESH RATE    **
** WILL BE ABOUT 42 TIMES PER SECOND AT 1MHZ CPU CLOCK                  **

SNDANA    LDAA    [0=F0       2 *
          STAA    CPIAA3      5 * ACCESS DATA DIRECTION REG.
          LDX     [0=FFF4     3 * CHANGE TO OUTPUTS
          STX     PIAA3       6 * MAKE CA2 AN OUTPUT (LOW)
          CLR     CHCNT       6
NXTCH     BSR     ZEROL           MIN OF 148 CYCLES
          INC     CHCNT       6
          LDAA    CHCNT       3
          CMPA    [7          2 SET NEXT CHANNEL TO BE OUTPUT
          BEQ     SNRTS       4 IF CHCNT EQ 7 RTS
          CMPA    SRTCH5      3 * CHANNEL POSITION FOR SCOPE TRIGGER
          BNE     NOTRIG      4 *
          JSR     TRIGGR      27* SEND PULSE TO TRIGGER SCOPE
NOTRIG    LDX     [DATA-4     4  * ADD 4 TIMES CHANNEL NO.(CHCNT) TO
          STX     DATPT       5  * DATA-4 TO SET POSITION OF NEXT CH
          ASLA                2  * TO BE SENT TO DAC
          ASLA                2  *
          ADDA    DATPT+1     3  *
          STAA    DATPT+1     4  *
          LDAA    DATPT       3  *
          ADCA    [00         2  *
          STAA    DATPT       4  *
          LDAA    PTSCYC      3*
          STAA    CNTINC      4* SET NO. OF POINTS PER CHANNEL
NXTDAT    LDX     DATPT       4 *
          LDAA    00,X        5 * LOAD DATA VALUE INTO ACC A
          ADDA    [0=80       2 * CONVERT 2S COMP VALUE TO OFFSET BINARY
          STAA    PIAA3       5 * SEND VALUE TO DIGITAL TO ANALOG CONVERTER
          LDX     SRTCH1      5  *
          BEQ     SKIPDY      4  *
          BSR     DELAY4         *
SKIPDY    LDAA    DATPT+1     3*
          ADDA    [24         2* SELECT NEXT DATA VALUE ON THIS CHANNEL
          STAA    DATPT+1     4*  ( DATPT = DATPT+24 )
          LDAA    DATPT       3*
          ADCA    [00         2*
```

```
          STAA    DATPT         4*
          DEC     CNTINC        6 * DEC POINTS PER CHANNEL COUNTER
          BNE     NXTDAT        4 * IF NOT ZERO TAKE NEXT DATA POINT
          BRA     NXTCH         4  START ON NEXT CHANNELS DATA
SNRTS     LDAA    [0=F0         2* ACCESS DATA DIRECTION REG.
          STAA    CPIAA3        5*
          LDX     [0=00F4       3* MAKE LINES INPUTS AGAIN FOR KEY PAD USE
          STX     PIAA3         6* KEEP CA2 LOW
          RTS                   5


** DELAY FORMULA IS  30+(SRTCH1)*8 = NO. OF CYCLES.    **
** VALID FOR SRTCH1 = 0000 THROUGH FFFE .              **
DELAY4    LDX     SRTCH1        5
          INX                   4 INX MAKES SRTCH1=0 SHORTEST DELAY POSSIBLE
DYLP      DEX                   4
          BNE     DYLP          4
          RTS                   5


** ZERO VOLTS LINE.  USED AS CHANNEL SPACER. MIN OF 148 CYCLES **
ZEROL     LDAA    [0=7F         2 *
          STAA    PIAA3         3 * SET DAC TO ZERO VOLTS
          LDAA    [4            2
ZERLP     BSR     DELAY4          (MIN OF 26 CYCLES)
          DECA                  2
          BNE     ZERLP         4
          RTS                   5


** SCOPE TRIGGER PULSE   7 CPU MACHINE CYCLES WIDE  **
TRIGGR    LDAB    [0=FC         2
          STAB    CPIAA3        5 MAKE LINE CA2 HIGH
          LDAB    [0=F4         2
          STAB    CPIAA3        5 MAKE LINE CA2 LOW
          RTS


** MASTER ROUTINE FOR CONVERTING REAL AND IMAGINARY PARTS FROM **
** TRANSFORM TO POLAR COORDINATES                              **
** AMP AND PHASE ARE OBTAINED USING CORDIC ROTATIONS           **
POLAR     LDX     [STARTC       *
          STX     REAL          * DEFINE START OF REAL LOCATIONS
          LDX     [STARTS        *
          STX     IMAG           * DEFINE START OF IMAGINARY LOCATIONS
          CLR     CHNO          CLEAR CHANNEL NO.
PLARLP    INC     CHNO
          BSR     SETROT        CALCULATE AMP AND PHASE FOR CHANNEL CHNO
          LDX     IMAG
          JSR     INX4          SET POINTER TO NEXT IMAG VALUE
          STX     IMAG
          LDX     REAL
          JSR     INX4          SET POINTER TO NEXT REAL VALUE
          STX     REAL
          LDAA    NCHPR         *
          CMPA    CHNO          * IF CHNO EQU NO. OF CHANNELS USED, RTS
          BNE     PLARLP        *
          JSR     BEEP          SOUND BEEPER
          RTS

** NEGATE VALUE ( AN APPROXIMATION) **
```

```
NEGATE    COM     00,X
          COM     01,X
          COM     02,X
          COM     03,X
          RTS


** LOADS VALUE POINTED TO BY INDEX INTO U. THEN SHIFTS U ROTACC BITS  **
** TO RIGHT. THE NO OF SHIFTS CONTROLS THE ROTATION STEP SIZE          **
LSWORK    LDAA    00,X        5
          STAA    U           4  MACHINE CYCLES
          LDAA    01,X        5
          STAA    U+1         4
          LDAA    02,X        5
          STAA    U+2         4
MSHIFT    LDAB    ROTACC      3  LOAD NO SHIFTS COUNTER
SFLOOP    LSR     U           6
          ROR     U+1         6
          ROR     U+2         6
          DECB                2
          BNE     SFLOOP      4
          RTS

** ROUTINE SETROT -  MOVES VECTOR TO FIRST QUADRANT, USES CORDIC ROTATIONS *
** TO ROTATE THAT VECTOR TO 0 DEGREES, CALCULATES NO OF DEGREES ROTATED    *
** CORRECTS FOR ACTUAL QUADRANT, CORRECTS FOR PHASE SHIFT DO TO DATA      **
** SAMPLING TIME SKEW, FINAL PHASE ON CHANNELS 2 TO 6 IS RELATIVE TO CH 1 **
** PHASE, FINAL PHASE ON CH 1 IS ACTUAL PHASE, CALCULATES AMPLITUDES IN   **
** MILLIVOLTS PER ROOT HZ .                                               **
SETROT    LDX     [0=FFFF     *
          STX     SRTCH1      * SET NO OF ROTATIONS COUNTER TO  -1
** MOVE VECTOR TO 1ST QUADRANT   **
** MIXED IN WITH QUAD MOVE IS A TEST TO SEE IF BOTH REAL AND IMAG PARTS **
** ARE ZERO (ONLY MS 16 BITS ARE LOOKED AT) IF TRUE ROTATION CNT SET 0  **
TQUAD     CLRB
          LDX     IMAG
          TST     00,X
          BPL     PIMAG
          BSR     NEGATE      NEGATE IMAGINARY PART
          ADDB    [02          0010 = NEG. IMAG PART
PIMAG     CLRA                CLEAR ZERO FLAG
          LDX     00,X        LOAD 2 MSB TO INDX, IF ZERO SET FLAG
          BNE     NOTZRO      NOT ZERO
          INCA                IMAG PART IS ZERO, SET FLAG
NOTZRO    LDX     REAL
          TST     00,X
          BPL     PREAL
          BSR     NEGATE      NEGATE REAL PART
          INCB                 0001 = NEG. REAL PART
PREAL     STAB    QUAD        QUAD IS CODE FOR QUADRANT OF VECTOR
          LDX     00,X        LOAD 2 MSB OF REAL VALUE
          BNE     ROTATE      NOT ZERO CONTINUE
          DECA                DEC ZERO FLAG
          BNE     ROTATE      IF ZERO MEANS BOTH VALUES WERE ZERO
          STX     SRTCH1       ZERO IS IN INDEX SO IT IS USED TO SET ROT CNT
          BRA     CORECT      BOTH WERE 0 SO SKIP ROTATION ROUTINE
*  QUAD = 00, 01, 10, 11  EQU  QUADRANT  1, 2, 4, 3.       *
** NEXT SECTION ROTATES VECTOR (IN FIRST QUAD.) CLOCKWISE UNTIL IMAG **
** BECOMES NEGATIVE. NO. OF ROTATION STEPS IS IN SRTCH1              **
ROTATE    LDX     SRTCH1      4 *
          INX                 * 4
          STX     SRTCH1      5 * INC ROTATION COUNTER
```

```
          LDX     IMAG        4
          BSR     LSWORK            LOAD U AND SHIFT RIGHT
          LDX     REAL        4
          LDAA    03,X        5 NEXT 11 LINES DO COS = COS + SIN/2 EXP N
          ADDA    U+2         3
          STAA    03,X        6
          LDAA    02,X        5
          ADCA    U+1         3
          STAA    02,X        6
          LDAA    01,X        5
          ADCA    U           3
          STAA    01,X        6
          BCC     NOINC       4
          INC     00,X        7
NOINC     BSR     LSWORK
          LDX     IMAG        4
          LDAA    03,X        5 NEXT 11 LINES DO  SIN=SIN - COS/2 EXP N
          SUBA    U+2         3
          STAA    03,X        6
          LDAA    02,X        5
          SBCA    U+1         3
          STAA    02,X        6
          LDAA    01,X        5
          SBCA    U           3
          STAA    01,X        6
          LDAA    00,X        5
          SBCA    [00         2
          STAA    00,X        6
          BCC     ROTATE      4 IF SIN IS STILL POS. LOOP BACK
** NEXT SECTION CORRECTS ROTATION ANGLE TO ORIGINAL QUADRANT  **
CORECT    LDX     SRTCH1      * SCALE UP ROTATION STEP
          STX     XX          *
          LDX     [17905      *
          STX     Y           *
          JSR     MULT16      *
          LDAB    ROTACC       * SET NO. OF SHIFTS TO OBTAIN ANGLE IN  TEN
          ADDB    [03          * THOUSANDTHS OF DEGREES
          LDX     [U           *
          JSR     SHRLP        * SHIFT RIGHT  NOMINAL SHIFT IS 5
          LDAB    QUAD        LOAD QUADRANT CODE
          BEQ     COROUT       QUAD. 1  NO CORRECTION
          JSR     NEGATE      NEGATE ANGLE
          CMPB    [02         QUADRANT 4
          BEQ     COROUT
          LDAA    [0=40       * ADD 180.0000 DEGREES
          ADDA    U+3         *
          STAA    U+3         *
          LDAA    [0=77       * 1800000 DEC = 1B7740 HEX
          ADCA    U+2         *
          STAA    U+2         *
          LDAA    [0=1B       *
          ADCA    U+1         *
          STAA    U+1         *
          CMPB    [01          QUADRANT 2
          BEQ     COROUT
          JSR     NEGATE      NEGATE ANGLE
** NEXT SECTION CORRECTS FOR DATA SAMPLING TIME SKEW  **
COROUT    LDAB    CHNO        LOAD CHANNEL NO.
PCLOOP    DECB
          BEQ     NOPSHF      CHANNEL 1, NO PHASE CORRECTION NEEDED
          LDX     [PHS1       * SUBTRACT PHASE SHIFT FROM CHANNEL PHASE
          BSR     SUBFU3      *
```

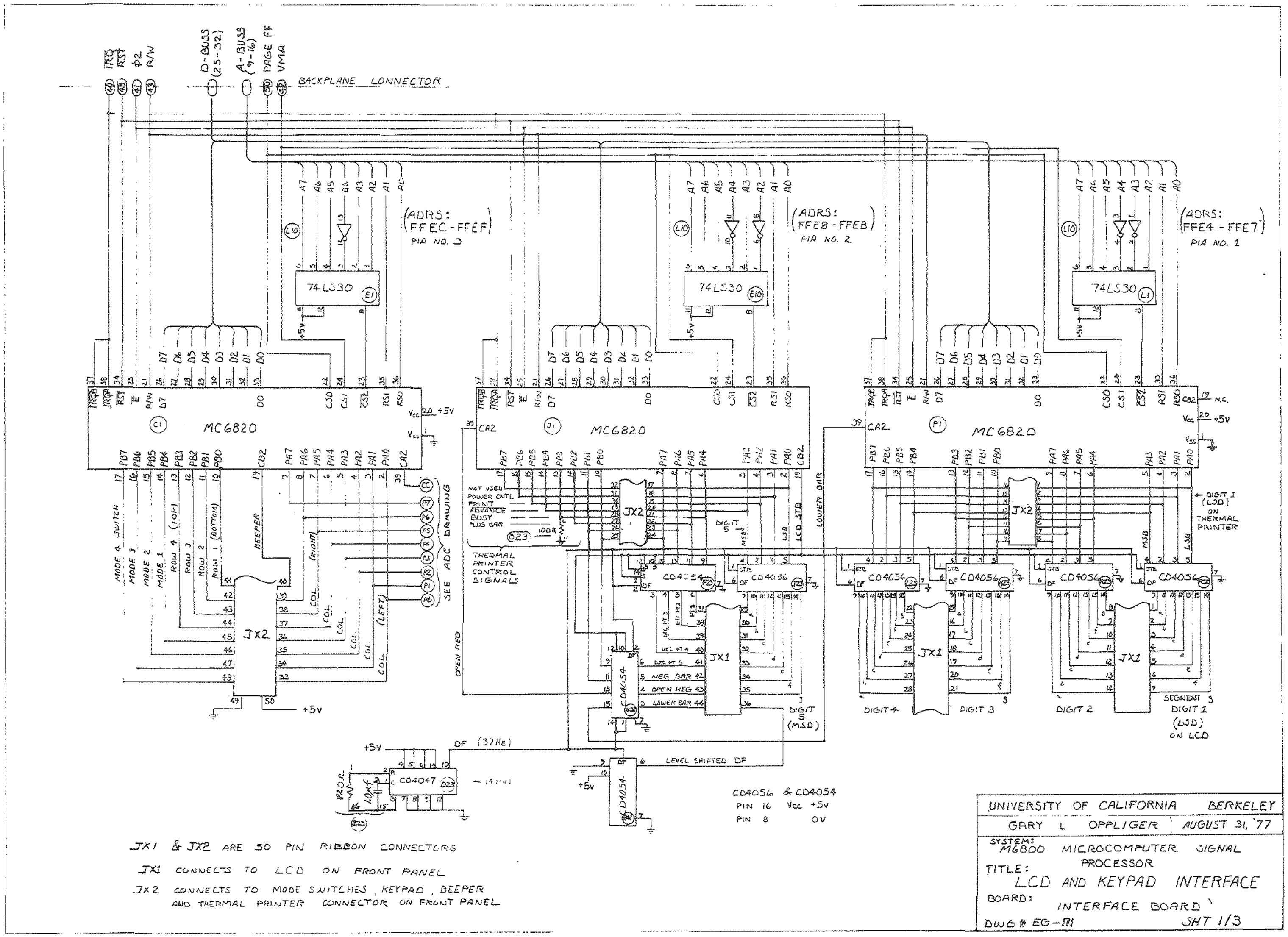
```
          BRA    PCLOOP      IF NEECED REPEAT PHASE CORRECTION
** NEXT  SECTION REFERENCES ALL PHASES TO CH 1 PHASE   **
NOPSHF    LDAB   CHNO        *
          DECB               *
          BEQ    SKREF       * CHECK CHANNEL NO IF CH 1, SKIP
REFCH     LDX    (STARTS     * SUBTRACT CH 1 PHASE FROM OTHER
          BSR    SUBFU3      * CHANNEL PHASES
SKREF     LDX    IMAG         * LOAD U INTO BINARY
          BSR    STRU3        * IMAGINARY LOCATION
** NEXT SECTION CONVERTS AMPLITUDE FOUNC BY ROTATION TO MILIVOLTS **
** IT IS EQUIVALENT TO MULTIPLYING AMPLITUDE BY 1.19265869        **
CNVTMV    LDX    REAL        * LOAD AMPLITUDE INTO MULT POSITION
          LDX    00,X        *
          STX    XX          *
          LDX    (0=4C54       19540 CONSTANT HAS ERROR OF 4 PPM
          STX    Y
          JSR    MULT16      MULTIPLY
          LDAB   (06         *
          JSR    SHFLP       * CORRECT MULTIPLY BY SHIFTING
          LDX    REAL         *
          BSR    STRU3        * STORE NEW AMPLITUDE
          RTS


** STORE 3 LS BYTES OF U IN 3 MS BYTES CF INDEXED LOCATION **
STRU3     LDAA   U+1
          STAA   00,X
          LDAA   U+2
          STAA   01,X
          LDAA   U+3
          STAA   02,X
          RTS


** SUBTRACT 3 MS BYTES CF INDEXED LOCATION FROM 3 LS BYTES OF U  **
SUBFU3    LDAA   U+3
          SUBA   02,X
          STAA   U+3
          LDAA   U+2
          SBCA   01,X
          STAA   U+2
          LDAA   U+1
          SBCA   00,X
          STAA   U+1
          RTS


** ROUTINE MAKES COMPUTER JUMP TO NEXT 4K MEMORY SECTION, IE BEGIN+0=1000 **
** IT IS CALLED BY PRESSING <RUNB> THEN <0>, IT MAY BE USED TO JUMP TO AN **
** AUXILLARY SET OF CONTROL OR TEST PROGRAMS                              **
RUNB      JSR    SCNKEY
          JSR    NCODE
          TSTA
          BNE    RBOUT
          JMP    BEGIN+0=1000
RBOUT     RTS




          STOP
          END
```

```
555555555.555555555.555555555.555555555.555555555.555555555.555555555.555555555.
555555555.555555555.555555555.555555555.555555555.555555555.555555555.555555555.
555555555.555555555.555555555.555555555.555555555.555555555.555555555.555555555.
555555555.555555555.555555555.555555555.555555555.555555555.555555555.555555555.
```

```
555555555.555555555.555555555.555555555.555555555.555555555.555555555.555555555.
555555555.555555555.555555555.555555555.555555555.555555555.555555555.555555555.
555555555.555555555.555555555.555555555.555555555.555555555.555555555.555555555.
        1         2         3         4         5         6         7         8

 EMRECOO                                                                     5
                                                                      96
 **BKY69K*C 02 OCT 78 18.13.15.

$ 0.63;     44 PAGES;  2,643 PRINT LINES;  PRINTER  12, EQ
B I L L B O A R D      WRITEUPS SUBSET BKYNEWS WAS LAST CHANGED  SEP 28
                       HANDBOOK SUBSET CHANGES WAS LAST CHANGED  SEPT 13


 OCT 1                         LBL NEWSLETTER
THE COMPUTER CENTERS MONTHLY NEWSLETTER (NOT BKYNEWS) IS FREE,
TO GET ON THE MAILING LIST, CALL X5529.
    LIBRARY        SUBSET       DATE
SEPT 30            THE MSS WRITER WILL BE SHUT OFF FOREVER OCT. 1, 1978

SEPT 29            PSS HARDWARE FAILURE
THE FOLLOWING PSS  SUBSETS WERE DESTROYED DUE TO HARDWARE FAILURE
    LIBRARY        SUBSET
    HIGHRNPP       FCCEFF, STFCF

SEPT 28            ACCOUNTING CHARGES TO CHANGE
THE CHANGES IN ACCOUNTING CHARGES DETAILED IN BKYNEWS WILL GO INTO EFFECT
OCTOBER 1, 1978.

SEP 21             UNUSED TAPES ON THE MOVE
TAPES NOT ACCESSED FOR OVER 14 MONTHS WILL MOVE INTO INACTIVE STORAGE
OCT. 10.  SEND A SIGNED TLIST INDICATING THOSE TAPES THAT YOU WANT
INACTIVE TO TAPE LIBRARIAN.  OTHERWISE, THEY WILL BE FORCE-MOVED AND
WILL DECREMENT YOUR GROUP TAPE ALLOCATION BY A LIKE NUMBER.

SEPT 14            NEW WRITEUP AVAILABLE      (WITH APOLOGIES)
ANNOUNCING THE BIRTH OF A NEW WRITEUP CALLED  NETWORK  WHICH DESCRIBES THE
USE OF THE COMMERCIAL NETWORK  TYMNET  AND THE ARPANET  AT BKY.
 TO GET A COPY . . .
   LIBCOPY,WRITEUPS,OUTPUT,NETWORK.
   DISPOSE,OUTPUT=PR,PA=1F.        (IF PRINTED AT BKY)

SEPT 11            INTERACTIVE TIMEOUT TO BE REDUCED TO 10 MINUTES
ON SEPT 13, THE INTERACTIVE TIMEOUT WILL BE REDUCED FROM 30 MINUTES TO
10 MINUTES. THE REASON FOR THIS CHANGE IS TO TRY AND REDUCE CLOGGED
LOGON QUEUES ON THE B AND C MACHINES.  WE HOPE THIS CHANGE WILL IMPROVE
SERVICE. ONCE AGAIN WE ASK YOUR COOPERATION. PLEASE RELEASE RESOURCES WHEN YOU
ARE NOT ACTIVELY INTERACTING WITH THE COMPUTER.

    TO CALL A CONSULTANT   415-843-1089 DIRECT DIAL
                               451-5981 FTS ONLY
```

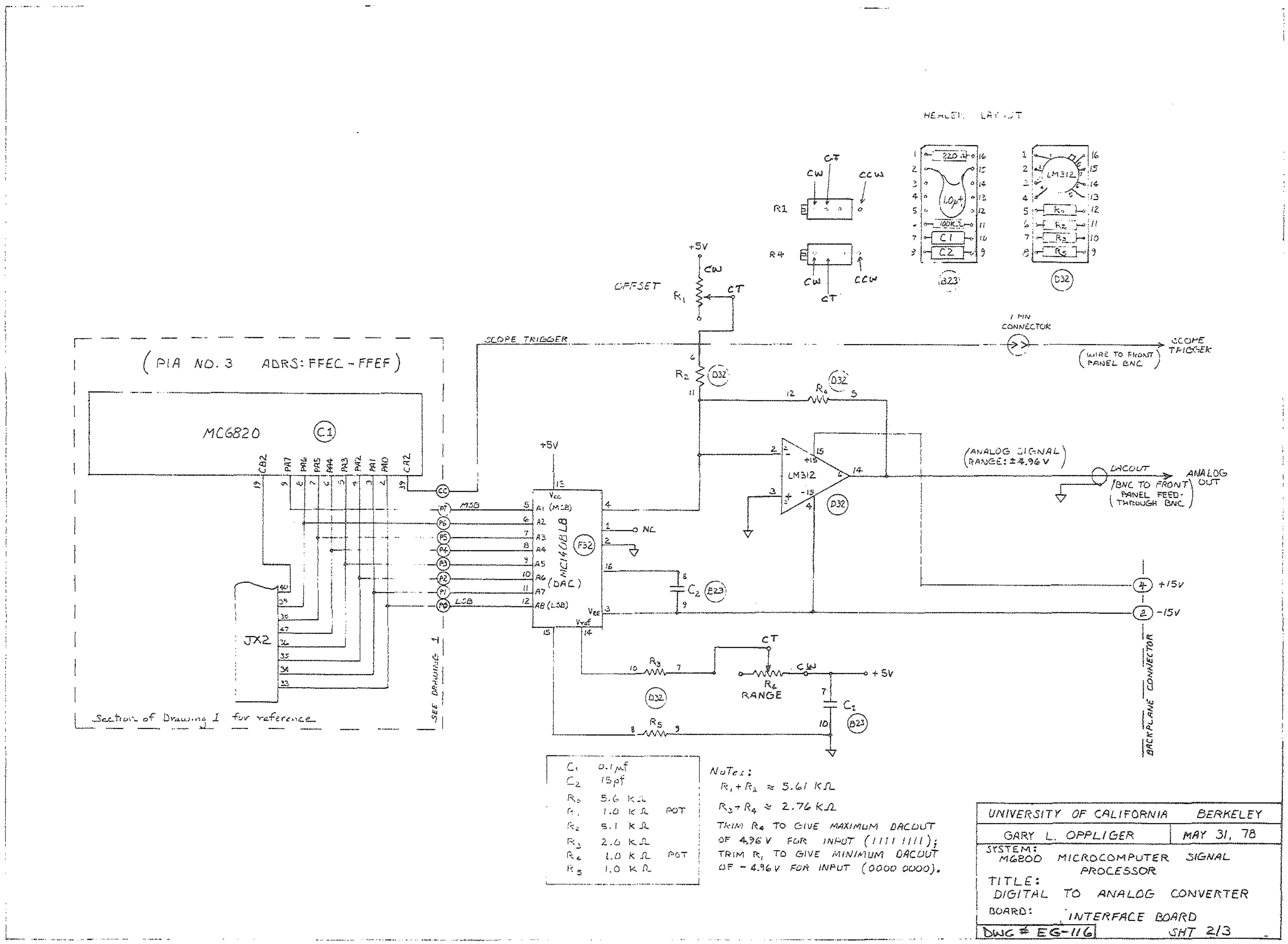
APPENDIX B

BOARD LAYOUTS FOR MICROCOMPUTER SIGNAL PROCESSOR



XBL 7810-11974

Figure 8. LCD and keypad interface.



XBL 7810-11975

Figure 9. Digital to analog converter.

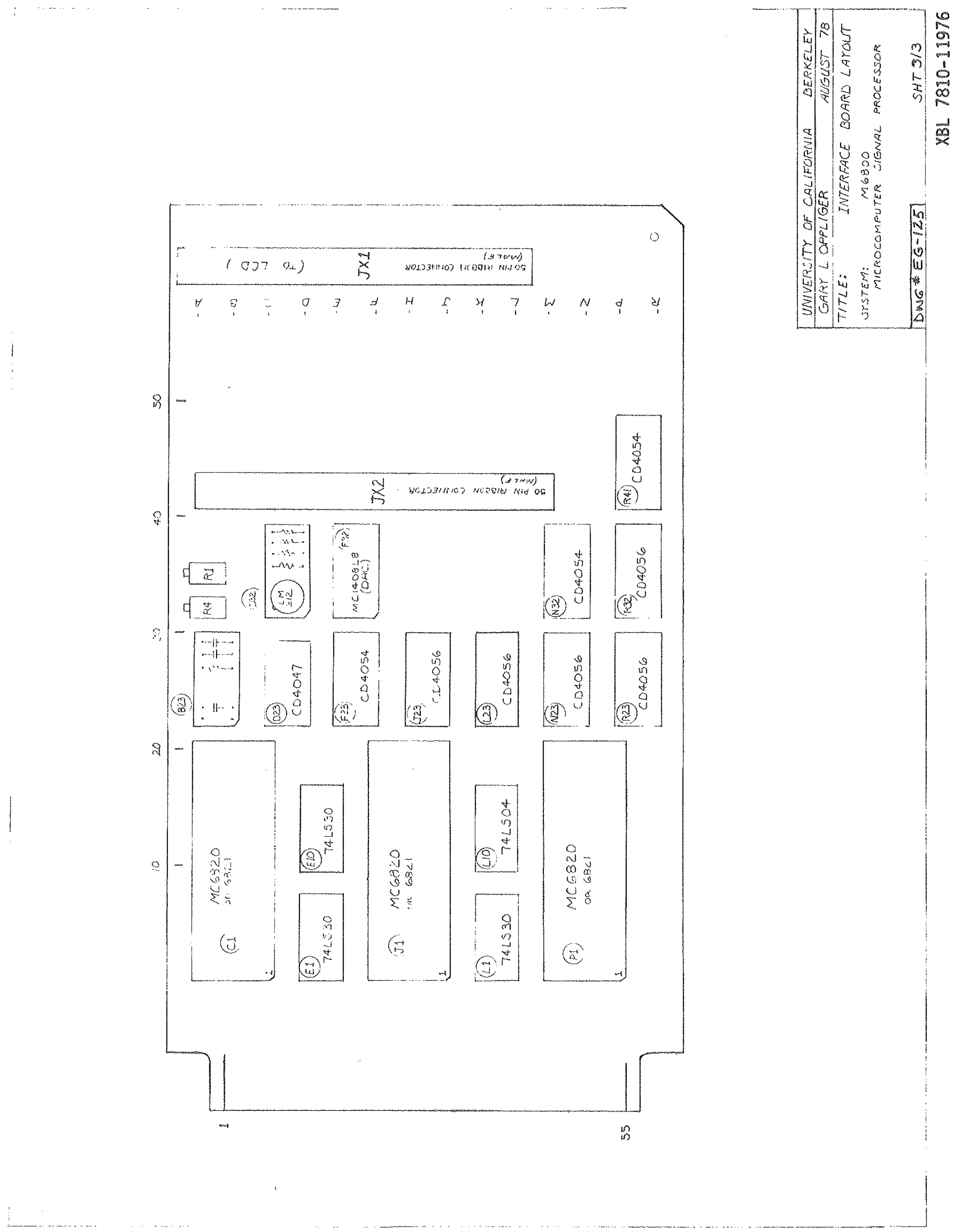
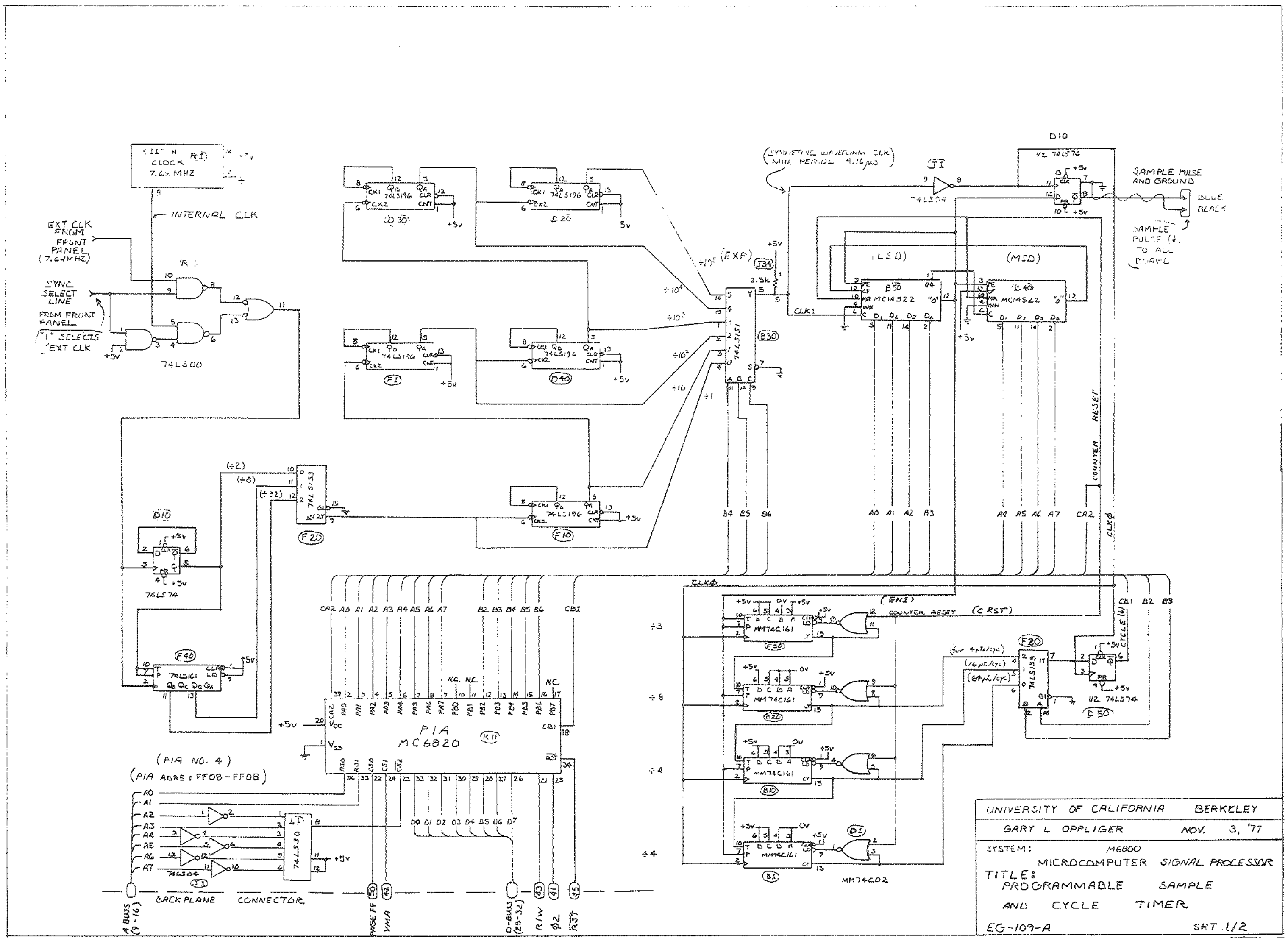
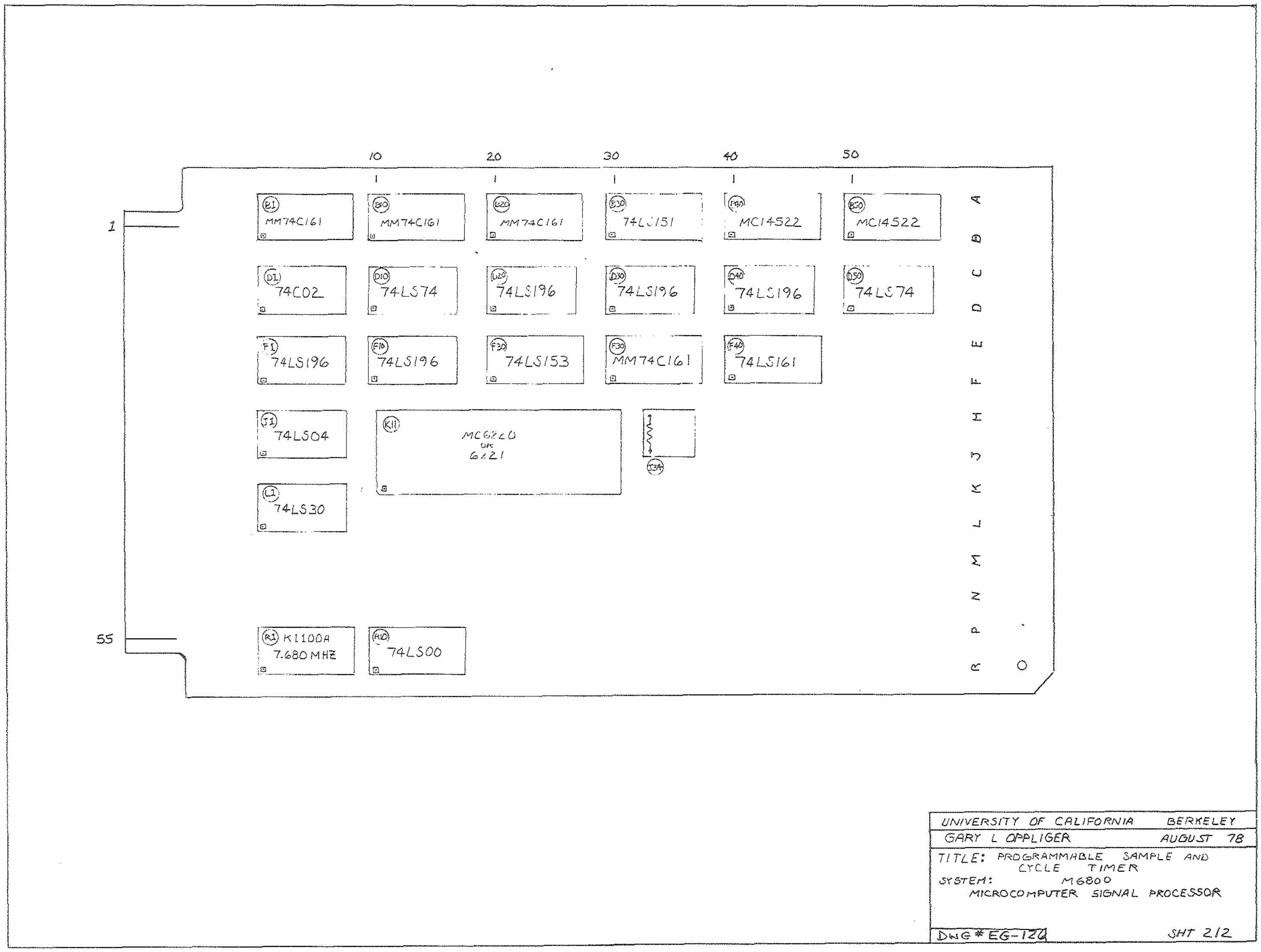


Figure 10. Interface board layout.



XBL 7810-11977

Figure 11. Programmable sample and cycle timer.



XBL 7810-11978

Figure 12. Programmable sample and cycle timer.

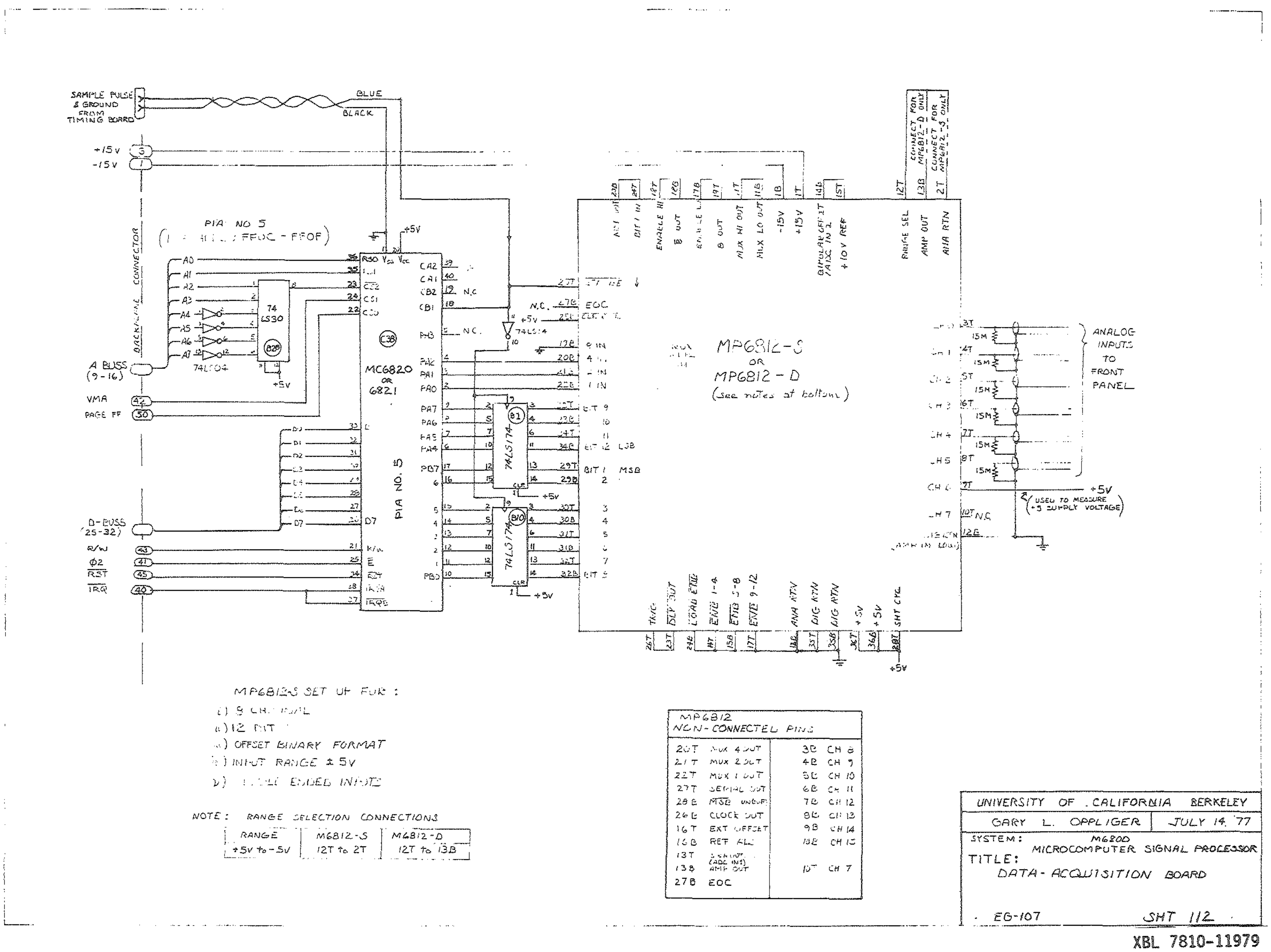
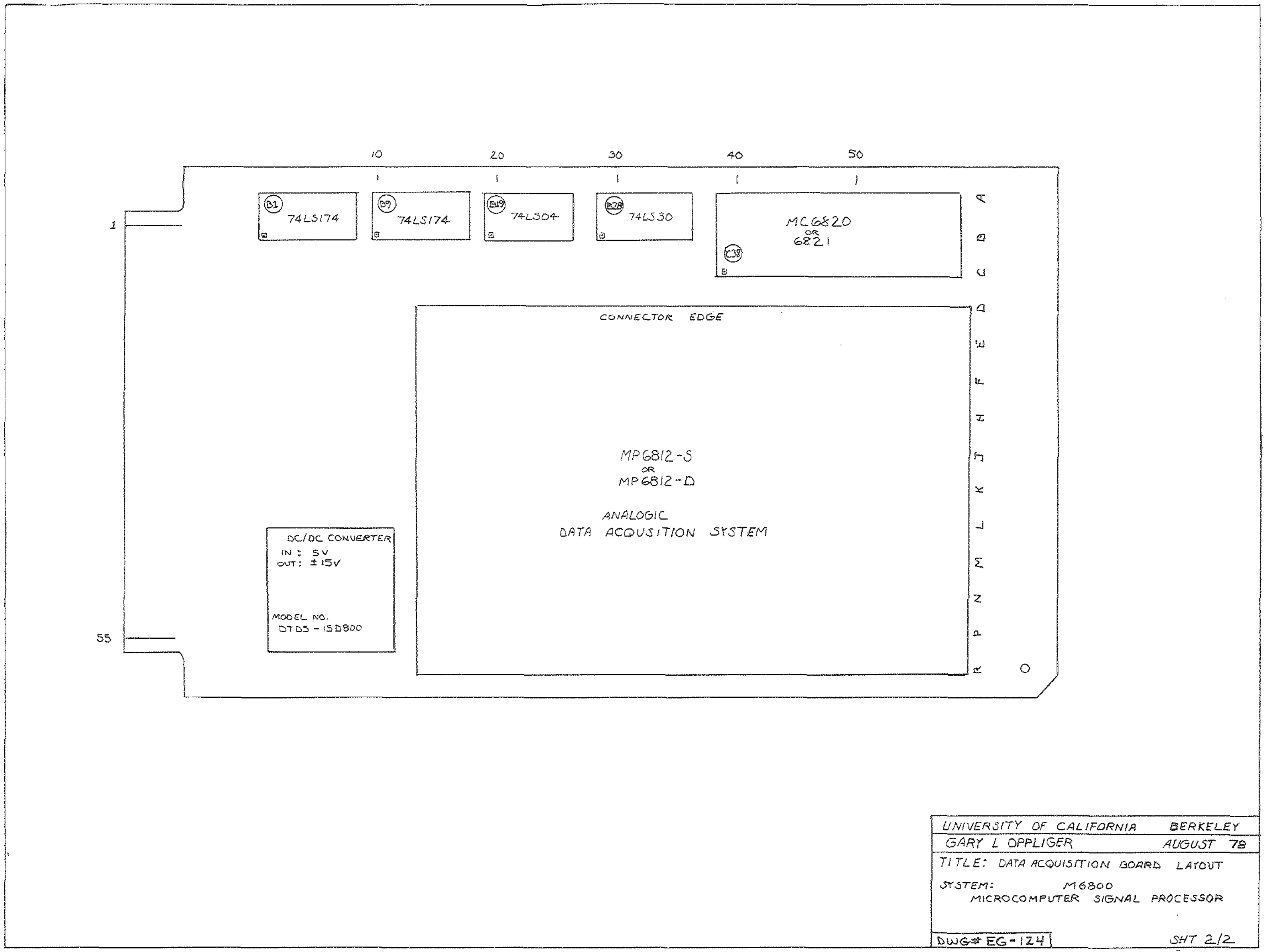
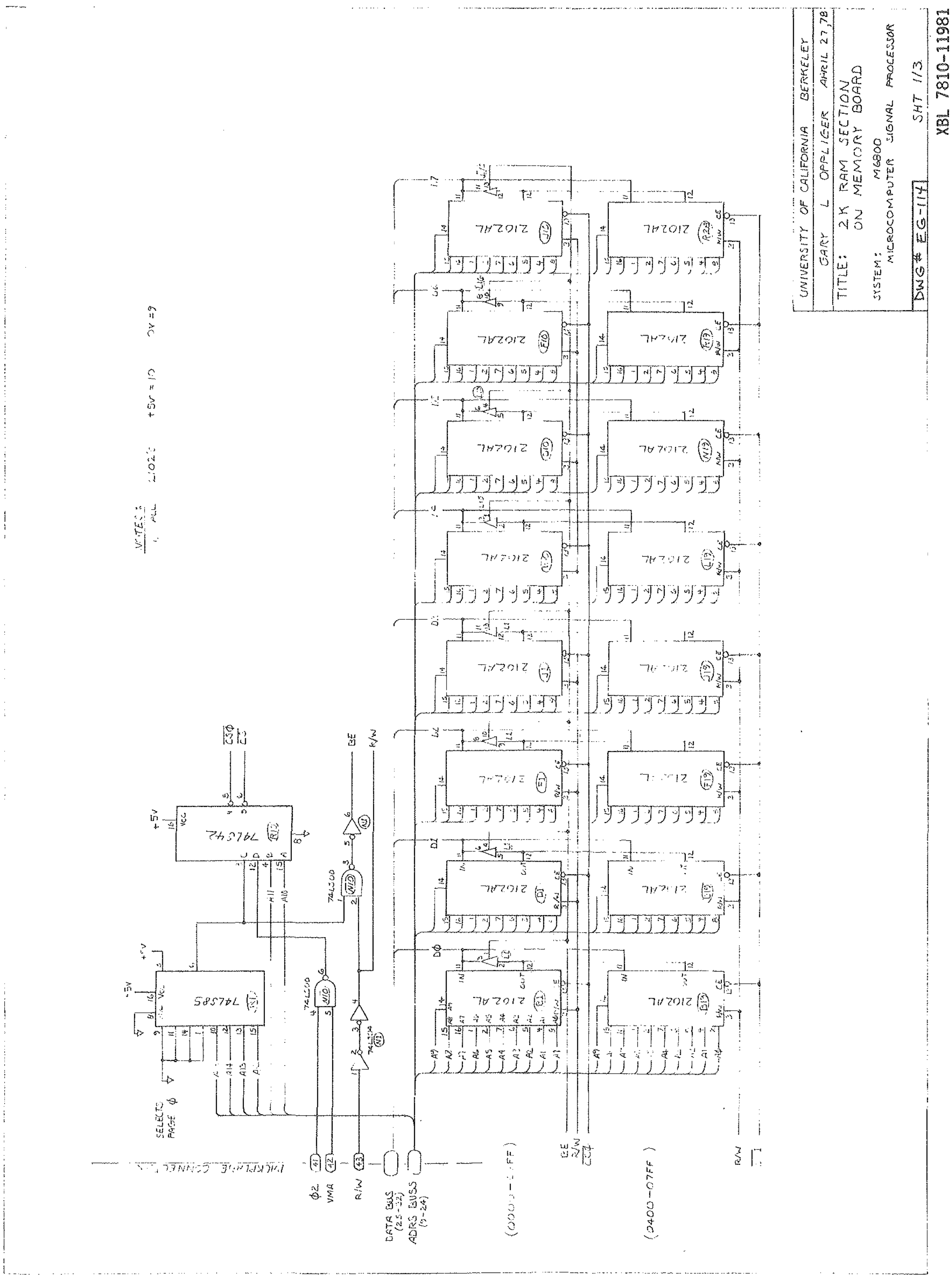


Figure 13. Data acquisition board.



XBL 7810-11980

Figure 14. Data acquisition board layout.



XBL 7810-11981

Figure 15. 2K RAM section on memory board.

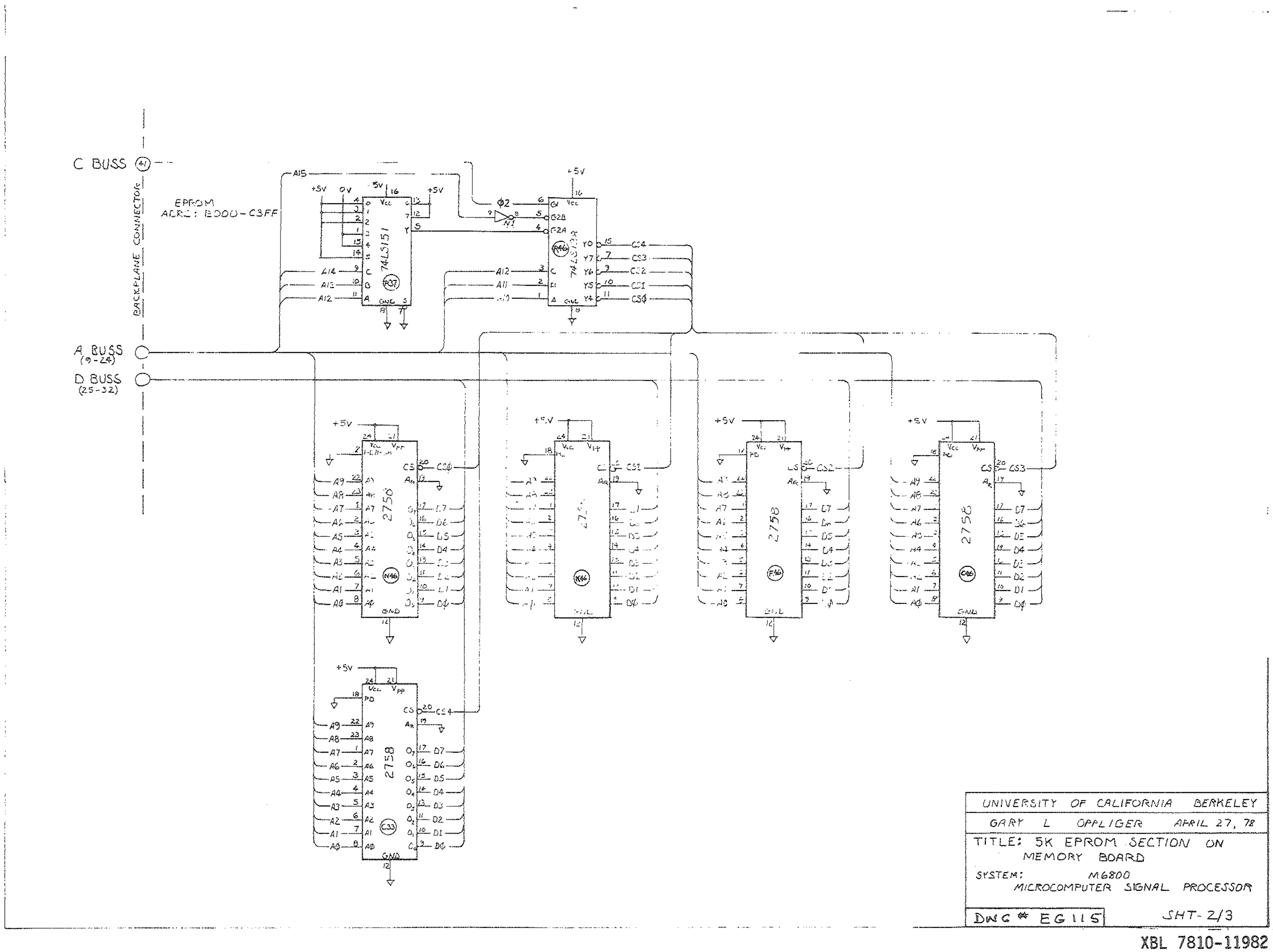
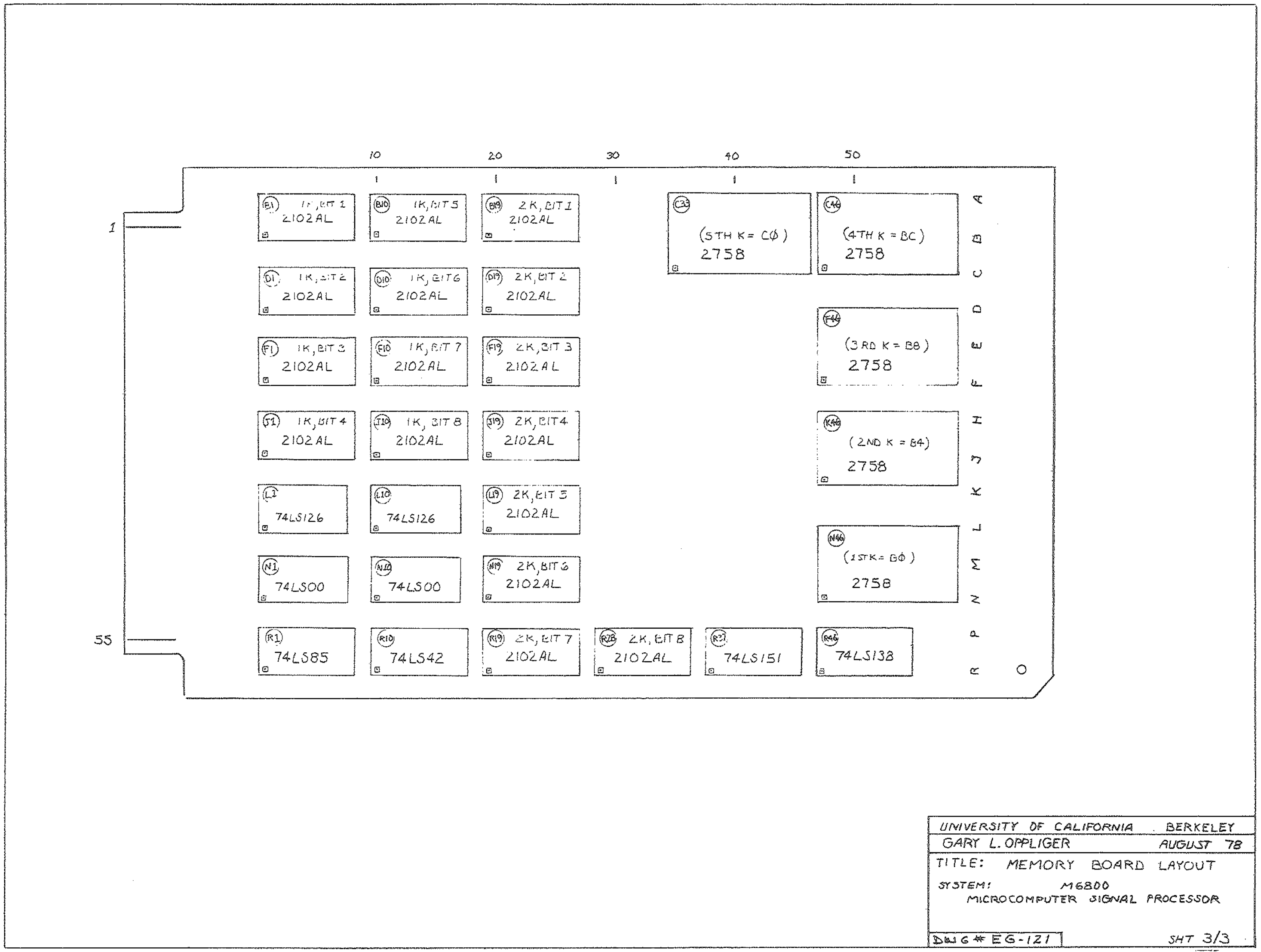
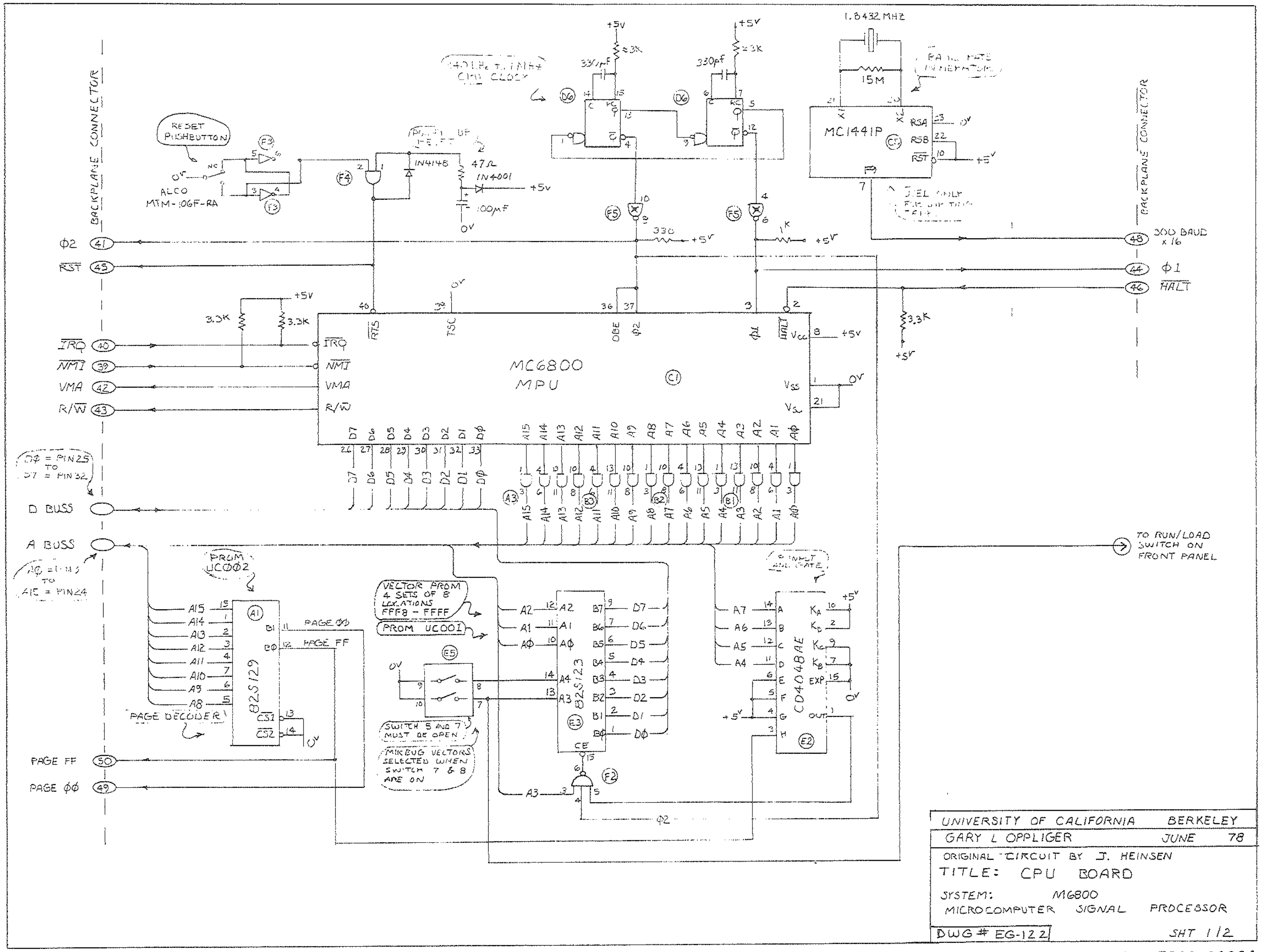


Figure 16. 5K EPROM section on memory board.



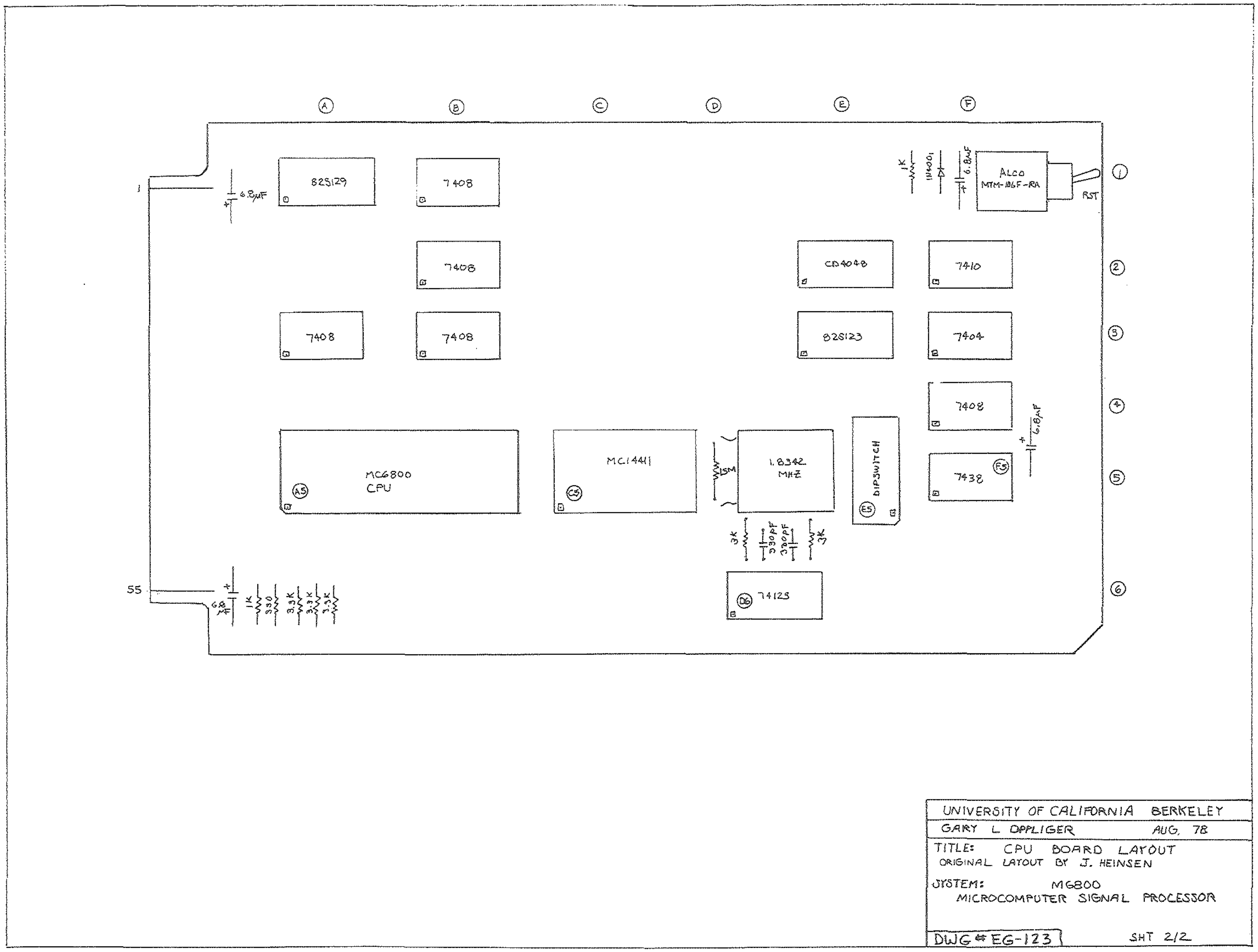
XBL 7810-11983

Figure 17. Memory board layout.



XBL 7810-11984

Figure 18. CPU board.



XBL 7810-11973

Figure 19. CPU board layout.

GRASS VALLEY FIELD TEST

LIST OF FIGURES

This section describes a field test of the EM-60, the data analysis and interpretation procedures, and a comparison between the survey results and the results obtained using other electrical techniques. The Leach Hot Springs area in Grass Valley, Pershing County, Nevada, was chosen for the first field site at which the entire system would be tested following local testing in Berkeley. The site, approximately 22 miles south of Winnemucca, lies within the regionally high heat flow area of northern Nevada (Figure 1) and has been surveyed in detail by means of various geophysical techniques (Beyer et al, 1976), including a low power prototype of the EM-60 (Jain, 1978).

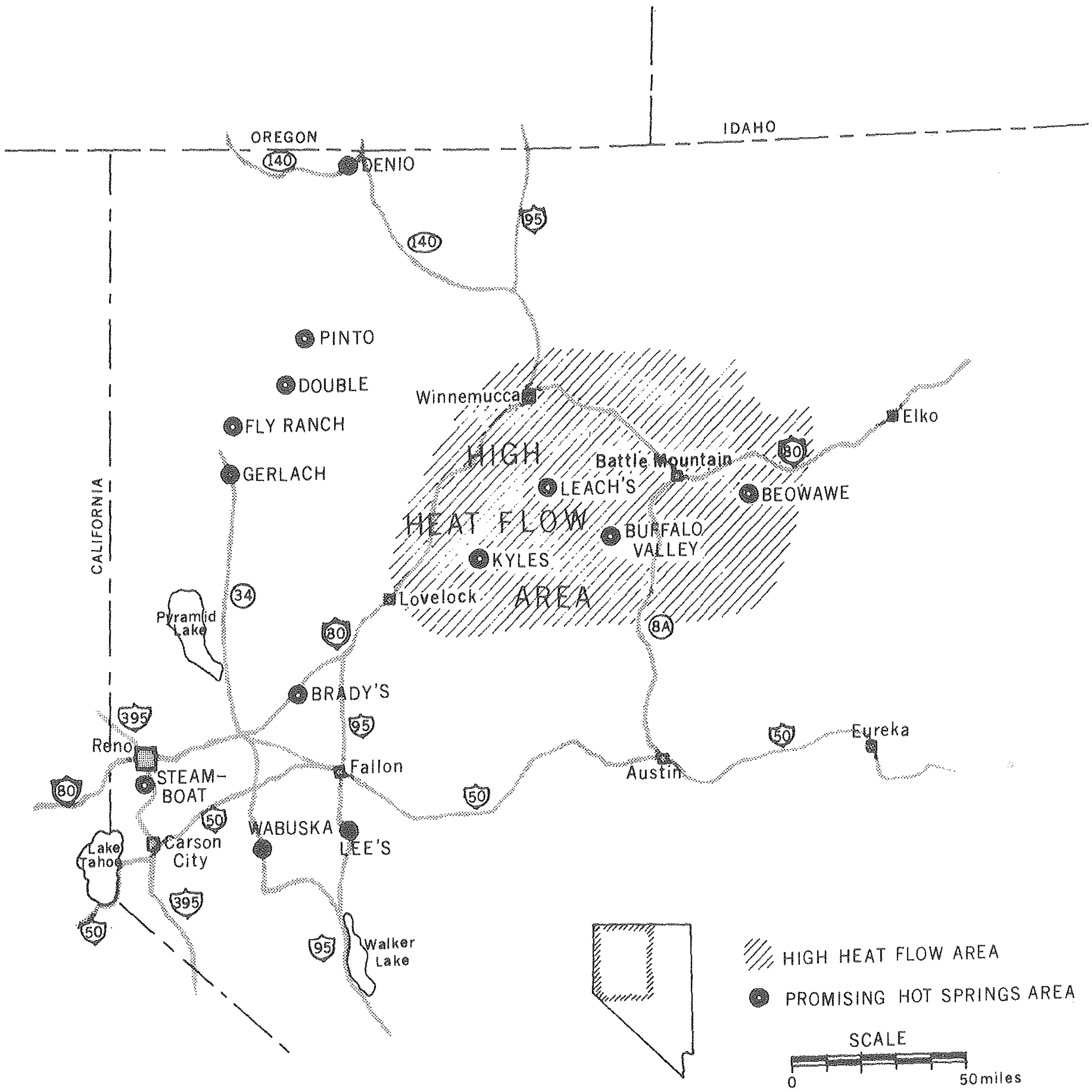
Survey Plan

The initial field test was conducted along established geophysical line E-E' (Figure 2) at stations previously occupied by Jain (1978). For direct comparisons of raw data and results with those of Jain, we followed his field procedures. The EM transmitter was placed at 3 West and the receiver was moved between sites 4 West, 5 West and 1 West. The unit separation between stations is 1 km.

Despite the dust and high temperatures during the work in July 1978, the survey proceeded quickly. The survey area is nearly flat (elevations of all stations are within 3m of each other) and the access is good. The principal instrumental problem encountered was the overheating of the electronics box for the Develco magnetometer caused by the high ambient temperature. This was easily solved by keeping it in an ice-filled tray.

Instrumentation and Procedures

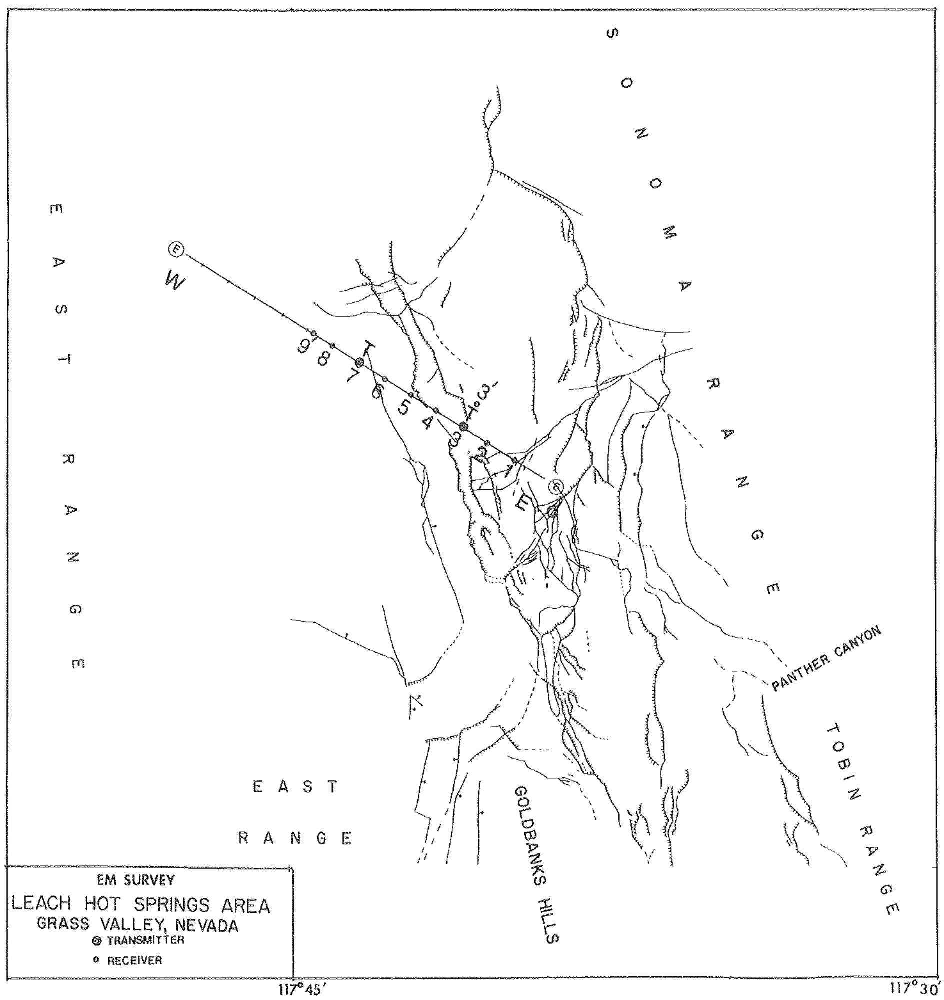
The transmitter loop consisted of a 4-turn, 50m radius horizontal loop of #6 AWG copper welding cable. Current was supplied in a square wave of positive and negative polarity at any desired period between 10^{-3} - 10^{3} sec by a 60 kW generator (see Section II on the transmitter). Peak-to-peak current carried by the coil ranged from $\sim$126 amp at 0.1 Hz to $\sim$15 amp at 10^{3} Hz.



Hot Springs in Northwestern Nevada

(XBL 735 676)

Figure 1. Location map, northwestern Nevada, showing prominent thermal springs within and outside of the Battle Mountain high heat flow area (after Sass et al, 1971).



XBL 784-8035

Figure 2. Electromagnetic transmitter and receiver locations.

The receiver system, described in detail in Section II, utilized a 3-axis Josephson-effect superconducting magnetometer as a sensor. A Develco model 8230 with sensitivity $10^{-5}\gamma/$ Hz was used. Each signal from the sensor was band-passed by means of a four-pole Butterworth filter and amplified. The pass band for a particular transmitter period is chosen according to the following considerations:

(1) The low-frequency cut-off is set just below the fundamental frequency to remove the geomagnetic and sferic noise. Natural geomagnetic noise is particularly bad at 20-30 sec period.

(2) The high frequency cut-off provides anti-alias control and it should be set below the Nyquist frequency. However, as it is desirable to reduce high-frequency natural noise, we set the high-cut frequency just above the highest odd harmonic we wished to extract from the signal. Although the system is designed for periods up to 1000 seconds, we learned that it is extremely difficult to obtain reliable results at periods 50 seconds and larger. This limitation is primarily due to lightning and sferic noise swamping the signal during the long times needed to average periods longer than 50 seconds. In Nevada it was almost impossible to find a 50 second or longer period without a lightning strike that would either throw the SQUID detectors out of lock or generate signals that exceeded the receiver's dynamic range. A further fundamental problem is that the natural noise spectrum rises roughly as 1/f below 0.1 Hz and so the averaging time to achieve a desired signal-to-noise ratio rapidly becomes impractical as the frequency decreases.

Experimentation was conducted to determine the number of harmonics that could be obtained accurately from a given period. Field tests show that the seventh harmonic can be obtained with no substantial errors for frequencies below 100 Hz. This allows us to obtain an entire decade of frequency measurements from a single transmitter period. However, above 100 Hz only, the fundamental frequency is transformed because the sampling rate is reduced to only four points-per-cycle. This does not significantly slow down the rate of data acquisition. For further details of the receiver capabilities see Section II on the digital signal averager.

After digitization each signal is averaged for the desired number of cycles, then Fourier transformed to yield spectral information on the odd harmonics of the signal. These values are printed on a thermal printer in one of two forms; either real and complex parts, or amplitude and phase.

The analog signals from all channels were also monitored continuously by means of two Gould paper-chart recorders. This enables the operator to interrupt and recommence the signal averaging, should interference from lightning or large spheric fluctuations degrade the data. A block schematic of the data acquisition system is shown in Section II.

Method of Interpretation

The interpretation of the electromagnetic sounding data (amplitude and phase) has been carried out using a direct one-dimensional (layered earth) inversion method. An initial estimate of the model parameters is made, and the inversion algorithm modifies these parameters until a best-fit, in the weighted-least-squares sense, is found between the observed data and the model predicted data. The application of direct inversion methods in electrical exploration has been described by Wu (1968), Parker (1970), Glenn (1973), Inman et al, (1973).

The inverse problem can be stated mathematically as

$$\phi = \sum_{i=1}^{n} w_i^{\,2}[y_i - f(b^\circ, x_i)]^2 \qquad (1)$$

where

N is the number of observed data

w_i is the weighting factor for the i^{th} data value

y_i is the ith observed data (i.e. amplitude or phase)

b^{o} is an initial estimate of the M model parameters (e.g., resistivity and layer thickness)

x_i is the known dependent variables (e.g., frequency and geometry) f is the non-linear function which relates the parameter b^o, x_i, to the observed quantities phase and amplitude.

Simply stated, the inverse problem is to find a set of model parameters, b_j which minimize ϕ. The values of b_j which minimize (1) are given by the solution to the set of equations:

$$\frac{\partial \phi}{\partial b_j} = 0 \qquad j = 1, M \tag{2}$$

Writing (1) in this form, we obtain

$$\sum_{i=1}^{n} w_i^2 f_i \frac{\partial f_i}{\partial b_j} = \sum_{i=1}^{n} w_i^2 y_i \frac{\partial f_i}{\partial b_j} , \qquad j = 1, M \tag{3}$$

where

$$f_i = f(\underset{\sim}{b}, x_i).$$

In general, the function $f(\underset{\sim}{b}, \underset{\sim}{x})$ is a non-linear function of b_j; thus making solution of (3) in closed form impossible. In practice ϕ is minimized by an interative technique.

Inversion Algorithm

The iterative weighted-least-squares algorithm used to interpret the EM-60 data follows a modified Marquardt approach. The model function $f(\underset{\sim}{b}^o, x)$ in equation (1) is expanded as a Taylor series about the current estimate, $\underset{\sim}{b}$, and only the first order terms are retained. This yields a linear estimate of the parameter changes, $\underset{\sim}{t}$, needed to reach the minimum of ϕ. The classic least squares statement of the problem would be

$$[A]\underset{\sim}{t} = \underset{\sim}{g} , \tag{4}$$

where

$$[A] = [P]^T [Q] [P] ,$$

and

$$\underset{\sim}{g} = [P]^T [Q] [y-f] .$$

P is the (NxM) matrix with elements $\left. \frac{\partial f_i}{\partial b_j} \right|_{b=b_{current}}$ and Q is the weight matrix.

The least squares estimate of t is given by

$$\underset{\sim}{t} = ([A]^T [A])^{-1} [A]^T \underset{\sim}{g} . \tag{5}$$

This linear estimate of the changes needed in the parameter vector can become unstable when $([A]^T[A])$ is nearly singular because the inverse blows up. (Instability means elements of t become so large they lie far outside a linear region about the present b and thus are invalid estimates.) To prevent this, a constant named a Ridge Regression estimate is added to the diagonal terms of $([A]^T[A])$. The so-called Ridge Regression estimate of t is:

$$\underset{\sim}{t}_{RR} = ([A]^T [A] + [I]K)^{-1} [A]^T \underset{\sim}{g} . \tag{6}$$

The benefit of (6) is that the inversion of $([A]^T[A] + [I]K)$ is stable. The value of K is varied throughout the inversion. At first the smallest value of K is found for which the estimate $\underset{\sim}{t}_{RR}$ yields a new model with a better fit to the data. As the interative process nears a minimum, the value of K is decreased so as to approach the classic least squares inverse.

The weighting matrix Q is a diagonal matrix with the diagonal terms equal to the inverse of the data variance. In this way, the residual for each data point is compared with its expected error.

$$Q = \frac{1}{\sigma^2}\begin{bmatrix} \frac{1}{\sigma_1{}^2} & & & 0 \\ & \frac{1}{\sigma_2{}^2} & & \\ & & \ddots & \\ & & & \frac{1}{\sigma_N{}^2} \end{bmatrix}$$

where

σ is a scalar factor called the problem standard deviation.

Statistical Evaluation of a Model

A set of model parameters, $\underset{\sim}{b}$, which minimize (1) is considered a good approximation with respect to the data if

$$(\chi_F^2)_{1-\alpha} \geq (\chi_F^2)_o \tag{7}$$

where $(\chi_F^2)_{1-\alpha}$ is the chi-square value at the $(1-\alpha)$ confidence level with $F = N-M$ degrees of freedom. The experimental value of the chi-square is given by

$$(\chi_F^2)_o = \frac{\hat{\sigma}^2}{\sigma^2} \quad ,$$

where

$$\hat{\sigma}^2 = \frac{[y-f]^T [Q]_{J=1} [y-f]}{N-M} = \frac{\phi MIN}{N-M} \quad .$$

$\hat{\sigma}$ is an estimate of the true problem standard deviation. Because data errors are expressed as percent of the actual data and used as the weights in Q, σ is assumed to be 1 (Jain, 1978).

The uncertainty in the estimated model parameters is given as (Bevington, 1969)

$$\sigma b_j^{\,2} = \sigma^2(\mathrm{cov}(P)_{jj}) \quad , \tag{8}$$

where the parameter covariance matrix, cov (P), is written as

$$\mathrm{cov}(P) = \left\{[P]^T[Q]_{\alpha=1}[P]\right\}^{-1} \quad . \tag{9}$$

Equation (8) gives the parameter variance for a linear solution only. In the case of a non-linear problem, as this one, (8) can be used as an approximation in conjunction with the parameter correlations. The parameter correlations are a measure of the linear dependence between parameters, and are given by

$$\mathrm{CORR}(b_{ij}) = \frac{\mathrm{cov}(P)_{ij}}{\mathrm{cov}(P)_{ii}\ \mathrm{cov}(P)_{jj}}$$

If the value of $\mathrm{CORR}(b_{ij})$ is near unity, then the parameters b_i and b_j are strongly correlated and nearly linearly dependent. In such a case the individual parameters are not well determined; rather, their ratio (if correlation coefficient is +1) or product (if correlation coefficient is -1) can be determined from the data.

If the correlations are small, then the standard deviations, given by the square roots of the diagonals of (8) , are a good measure of the uncertainty of each parameter. If, however, two parameters are highly correlated, CORR $b_{ij} \simeq \pm 1$, then the standard deviations will be larger than the actual uncertainties. Figure 3 illustrates this fact with a generalized slice of solution space. The two coordinate axes correspond to two parameters of the estimated layered earth model. The ellipse indicates a confidence region within which the residual sum of squares, ϕ, is expected to lie for a certain percent of the repeated experiments. This region also defines the values of the parameter ρ_2 (resistivity) and t_2 (thickness) which will give a residual sum of squares within the contour. The origin is defined by the parameter value at the final

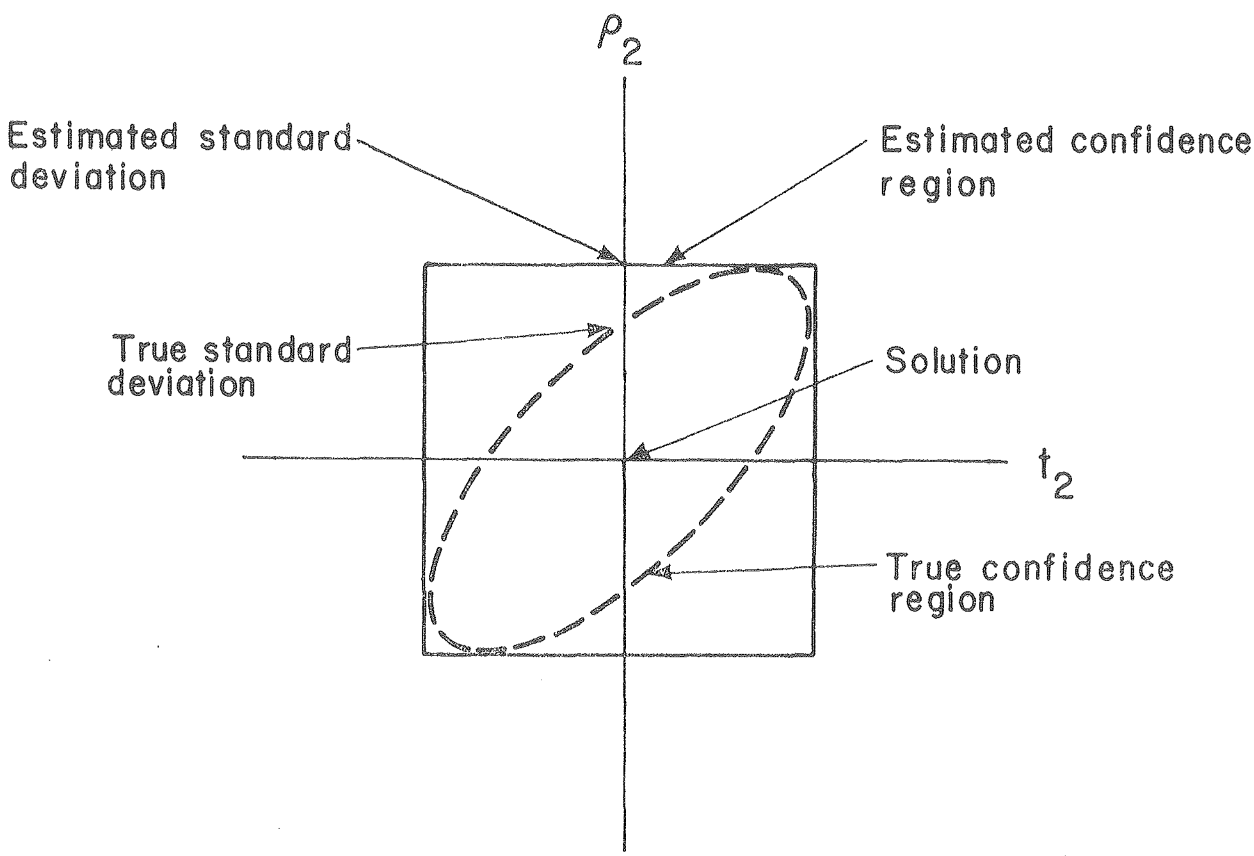


Figure 3. Generalized slice of solution space.

solution. The tilt of the axis of the ellipse is a measure of the degree of correlation between the two parameters. If the standard deviations from (8) are taken to be the true deviation estimates, then the ellipse is enclosed by a large box whose sides are defined by the standard deviation. The box, which ignores parameters correlation, represents a much larger confidence region than the ellipse. By using the standard deviation implied by the box, one obtains a very conservative estimate of the parameter confidence interval for correlated parameters. Therefore, by considering the standard deviations in conjunction with parameter correlations, a more realistic parameter standard deviation can be arrived at, which is always less or equal to the standard deviation computed from (8).

For a further description of the inversion method and procedure, see Jain (1978).

Combined Data Interpretation

During the EM sounding survey carried out in Nevada, three orthogonal components of magnetic field were measured for each transmitter-receiver location. This provided four sounding curves: the amplitude and phase at selected frequencies for both the vertical and radial components, $|H_r|$, $|H_z|$, H_r phase and H_z phase. The tangential magnetic field would be zero over a horizontal uniform medium. The amplitude of this component can thus be used as a qualitative measure of the inhomogeneity of the ground. If each sounding curve were inverted separately, four different earth models would result. These would then have to be averaged in some way to obtain a single model. A more objective approach is to find a single model which best fits all the data simultaneously. In this approach each data point is first weighted by its standard error (defined as the standard deviation divided by the square root of the number of samples) to set its relative importance and accuracy. All data sets are then inverted simultaneously.

Survey Results

The survey line E-E' crosses Grass Valley from southeast to northwest, passing approximately 1 km northeast of Leach Hot Springs (Figure 2). The orientation of the line is approximately 45° to the strike of the local geologic structure. Sounding data were taken with the transmitter as station 3 West and receiver locations at 1, 2, 4, and 5 West.

The observed field data and their standard errors for the four soundings are tabulated in Appendix A, and are illustrated in Figures (4) through (15). The transmitter-receiver locations are indicated on each figure (e.g. T3-R4 stands for transmitter at 3 West and receiver at 4 West). The standard errors listed in Appendix A of this section, are not plotted on Figures (4) through (15) since they would not show up on the scales used.

As previously discussed, the four sets of sounding data (amplitude and phase of H_z and H_r) were simultaneously inverted to obtain an overall best-fit model. The standard errors listed in Appendix A were derived from the diagonal weighting matrix [Q].

Ordinarily, considerable effort might be needed to originate a set of initial model parameters to begin the data inversions. All existing geological and geophysical data must be considered in making a first guess. However, since this had already been done by Jain (1978), the final models obtained from his work were used as starting models in the interpretation of our data. Previous data clearly indicated a basic three-layer structure with a fairly thick and conductive middle layer overlying and underlying more resistive layers. In cases where a three-layer model resulted in poor parameter resolution, a two-layer model was used to more accurately define the depth to and resistivity of the conductive layer. As previously noted by Jain (1978) for the three-layer case, the resistivity of the bottom layer is very poorly resolved. Whether $\rho_3 = 1$ or 100 Ωm makes little difference on the other parameters, and so we use the higher value as a constant.

Figures 4 and 5 present the data for sounding T3-R1. Here the three-layer model fits the data fairly well and the model parameters are well resolved. Figures 6 and 7 present the data for sounding T3-R2 with another three-layer model fit to the data. Figures 8 and 9 represent a two-layer fit to the same data. Note that the common parameters between the two models have virtually the same values. However, the resolution of the two-layer model parameters is much better, lending increased confidence in the depth to and resistivity of the conductive target.

Figures 10 through 13 presenting data for T3-R4 show the same situation as found for T3-R2. Figures 14 and 15 are for T3-R5 and yield a three-layer model with good parameter resolution.

In general, all the models are consistent with the seeming exception of the models for T3-R2. However, ρ_2 and h_2 for T3-R2 are highly correlated, correlation coefficient = 0.99. This means that the ratio h_2/ρ_2 is all that can be determined from the inversion. Therefore a thicker, more resistive middle layer (which would keep the ratio h_2/ρ_2 constant) would also fit the data. Thus, a general model of 8-10 Ωm, 500m thick top layer above a 100 Ωm basement is consistent with all the data obtained. The models interpreted from data previously taken by Jain (1978) are shown in Figure 16 for comparison.

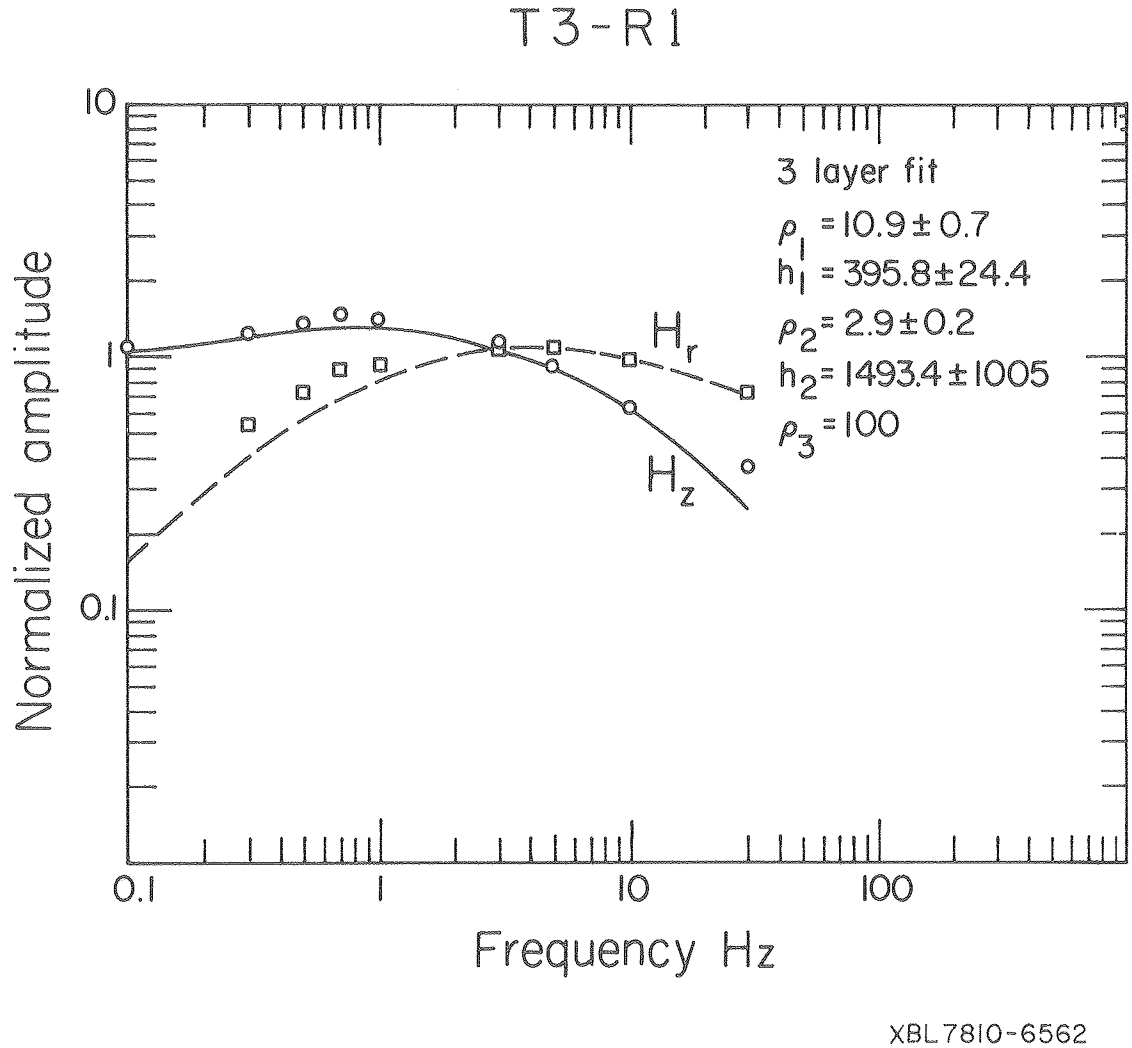
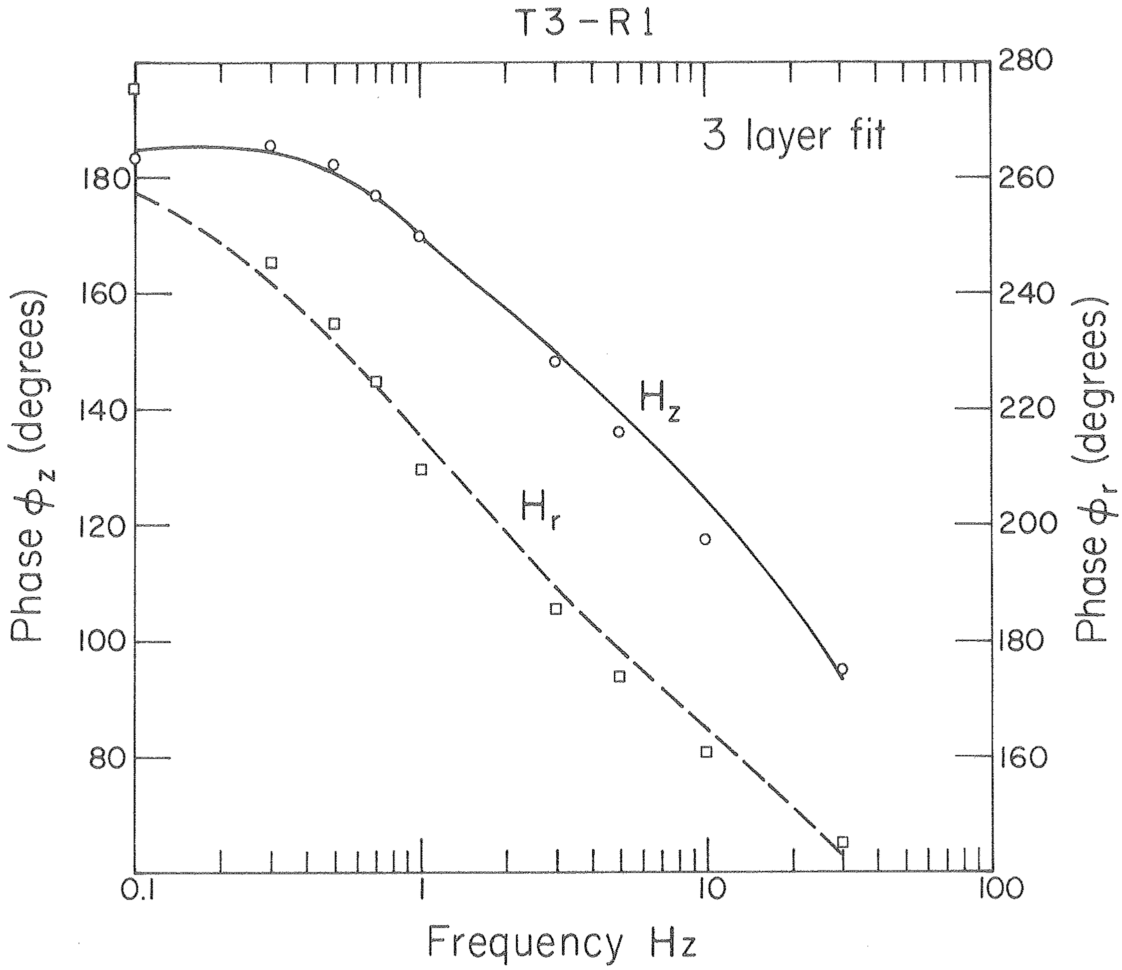
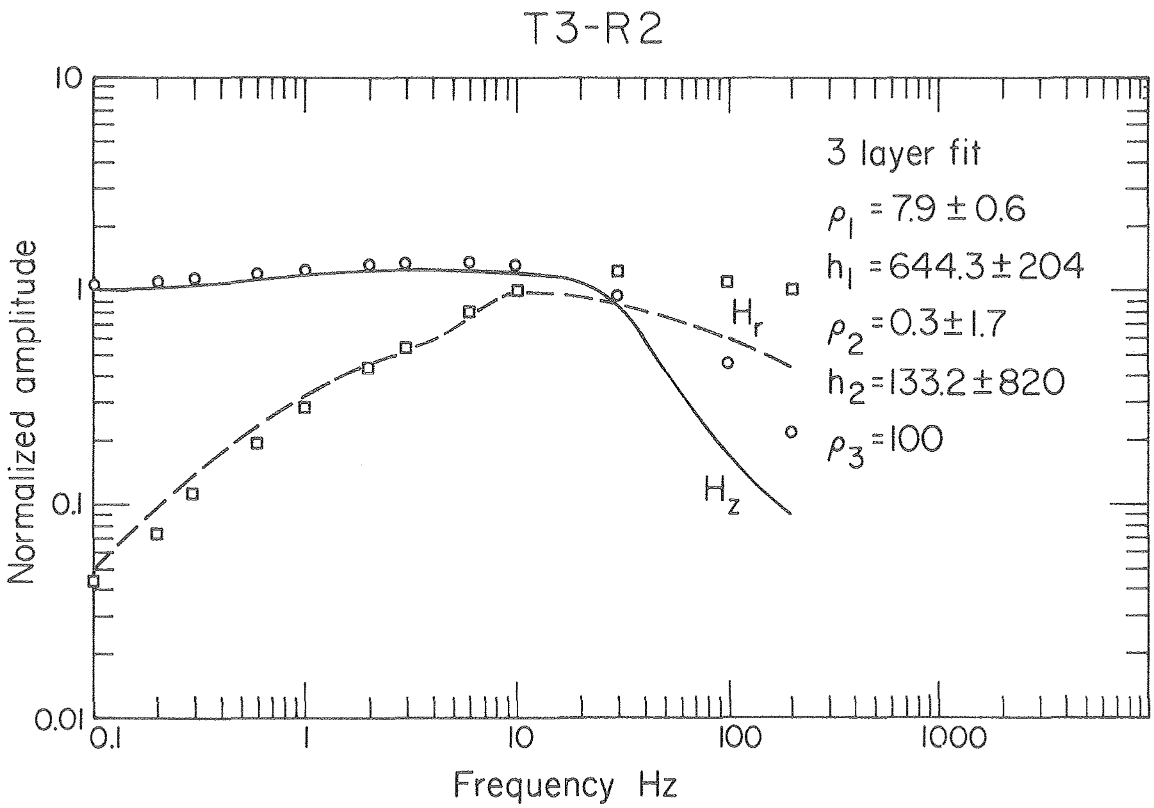


Figure 4. Amplitude spectra, T3-R1.



XBL 7810-6560

Figure 5. Phase spectra, T3-R1.



XBL 7810-6561

Figure 6. Amplitude spectra, T3-R2.

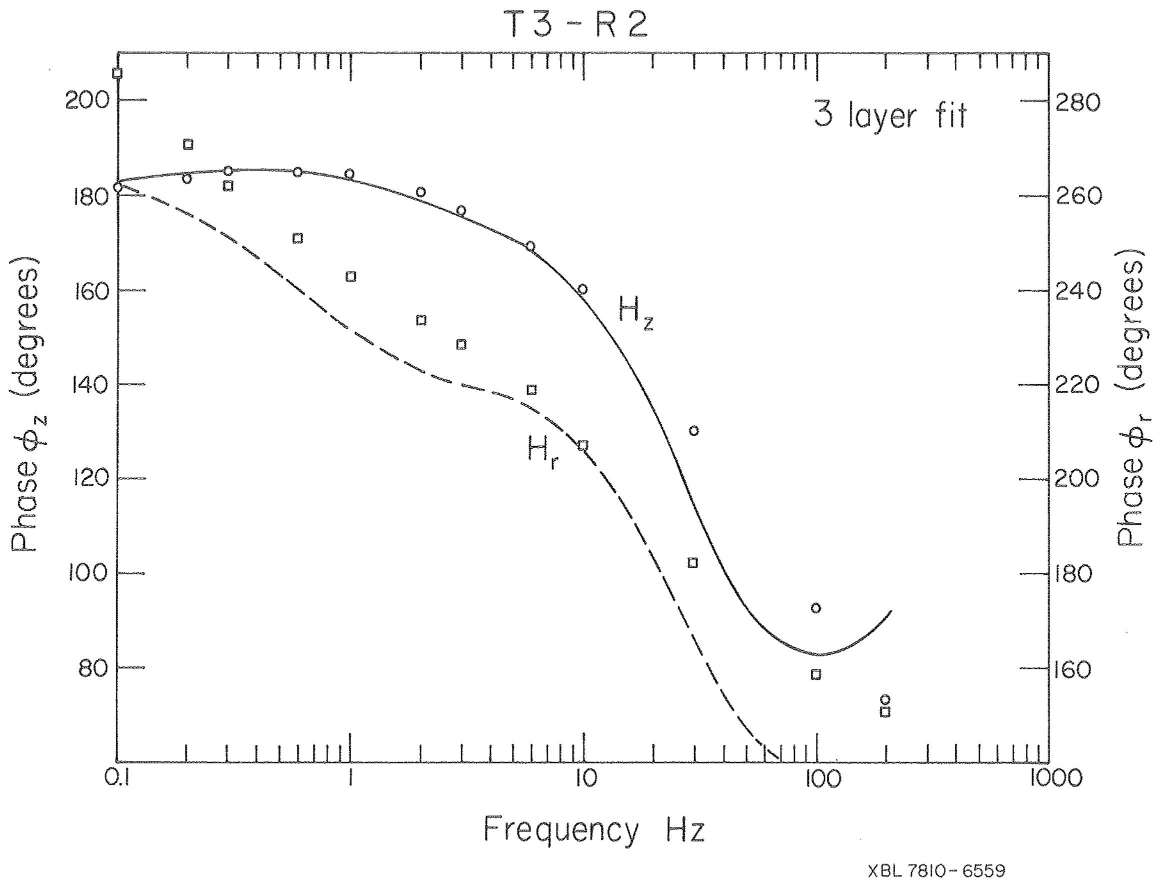


Figure 7. Phase spectra, T3-R2.

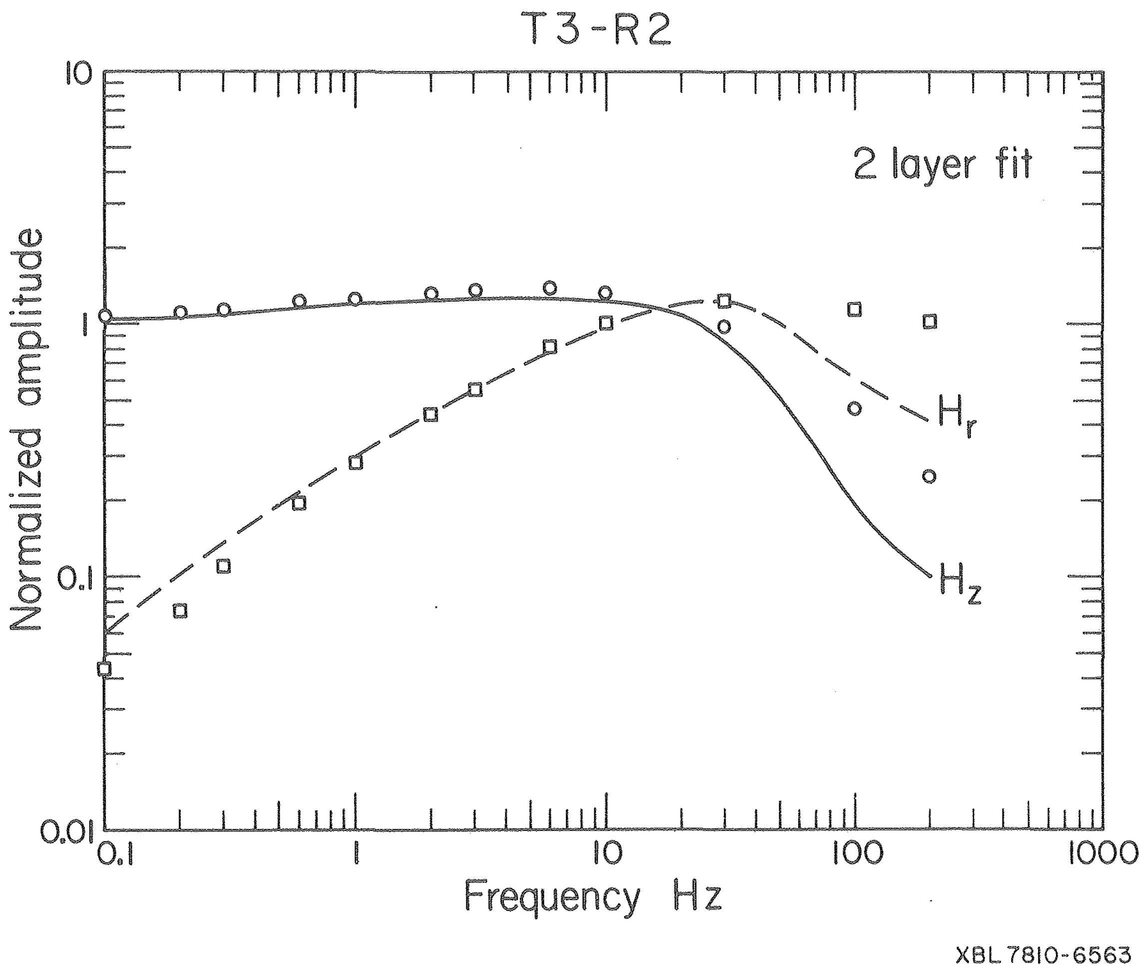


Figure 8. Amplitude spectra, T3-R2, two-layer fit.

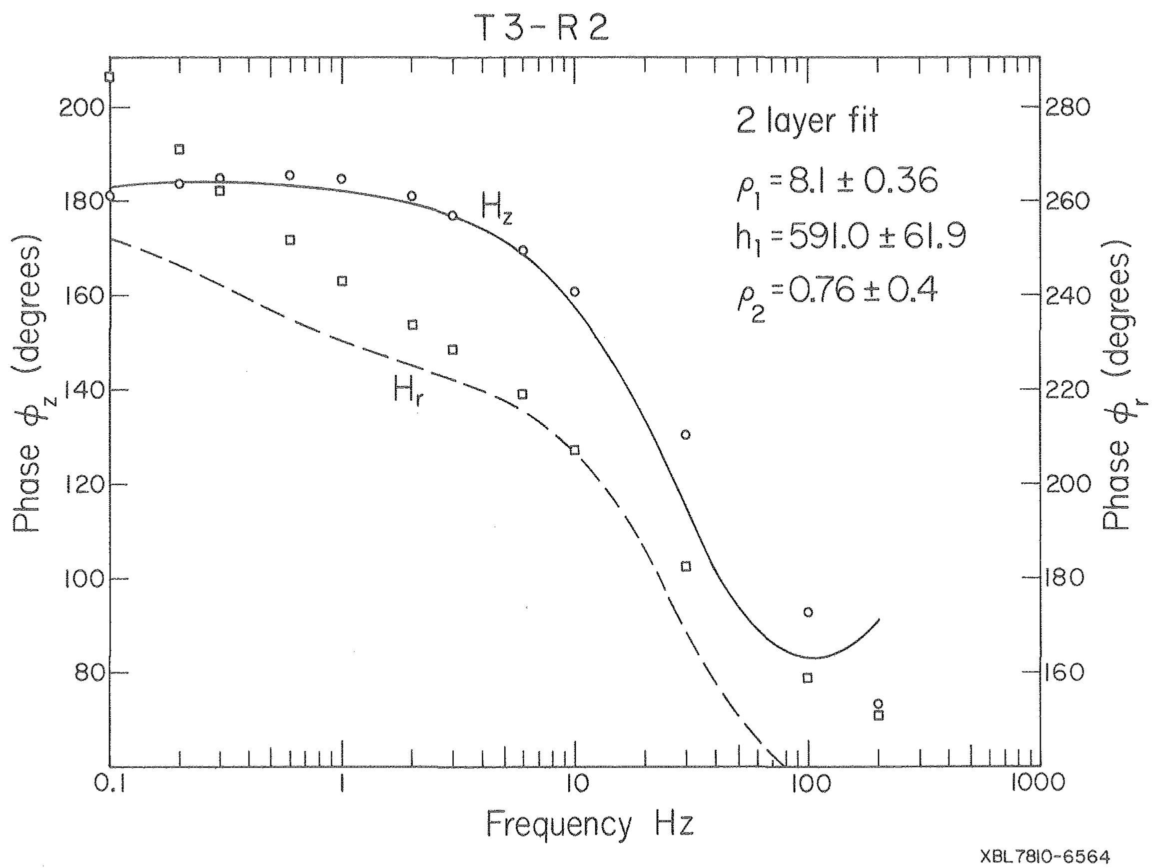


Figure 9. Phase spectra, T3-R2, two-layer fit.

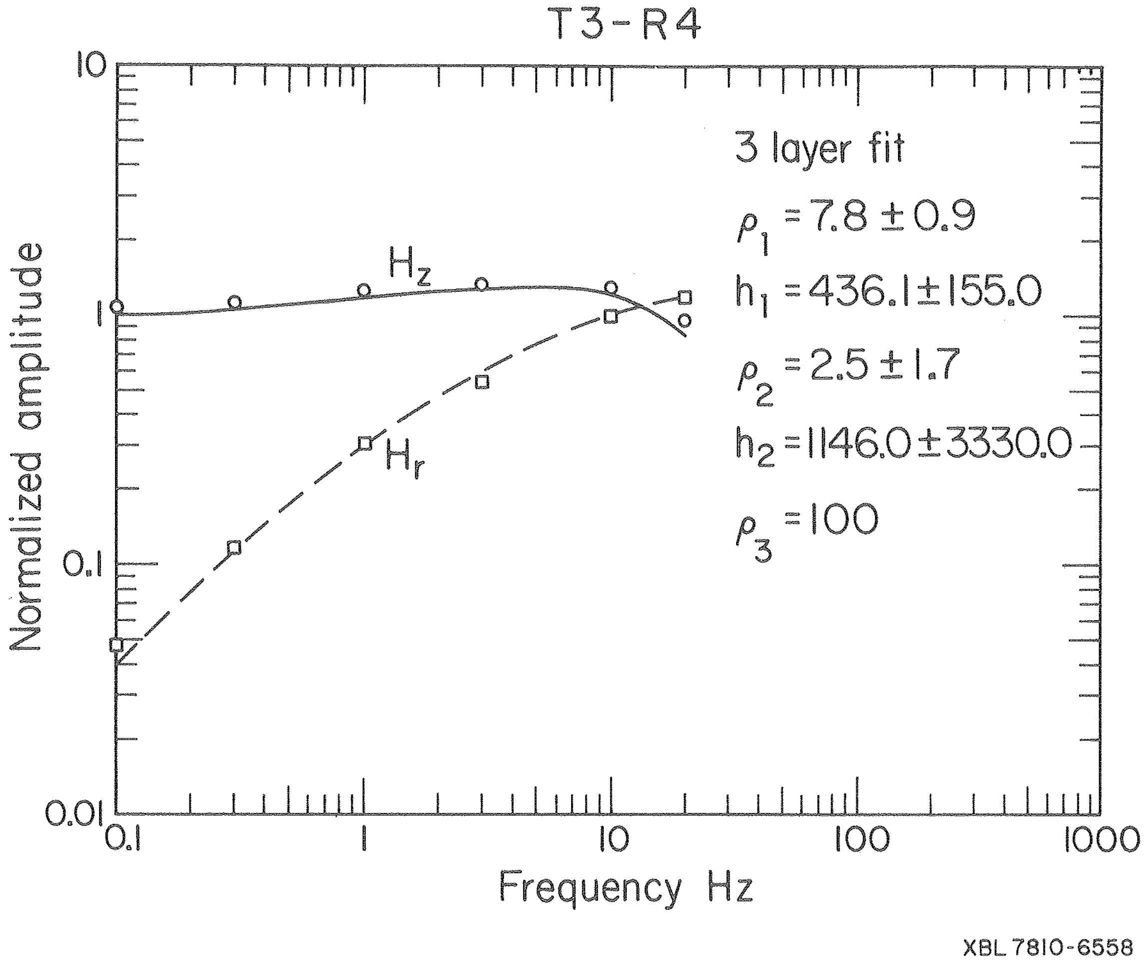


Figure 10. Amplitude spectra, T3-R4.

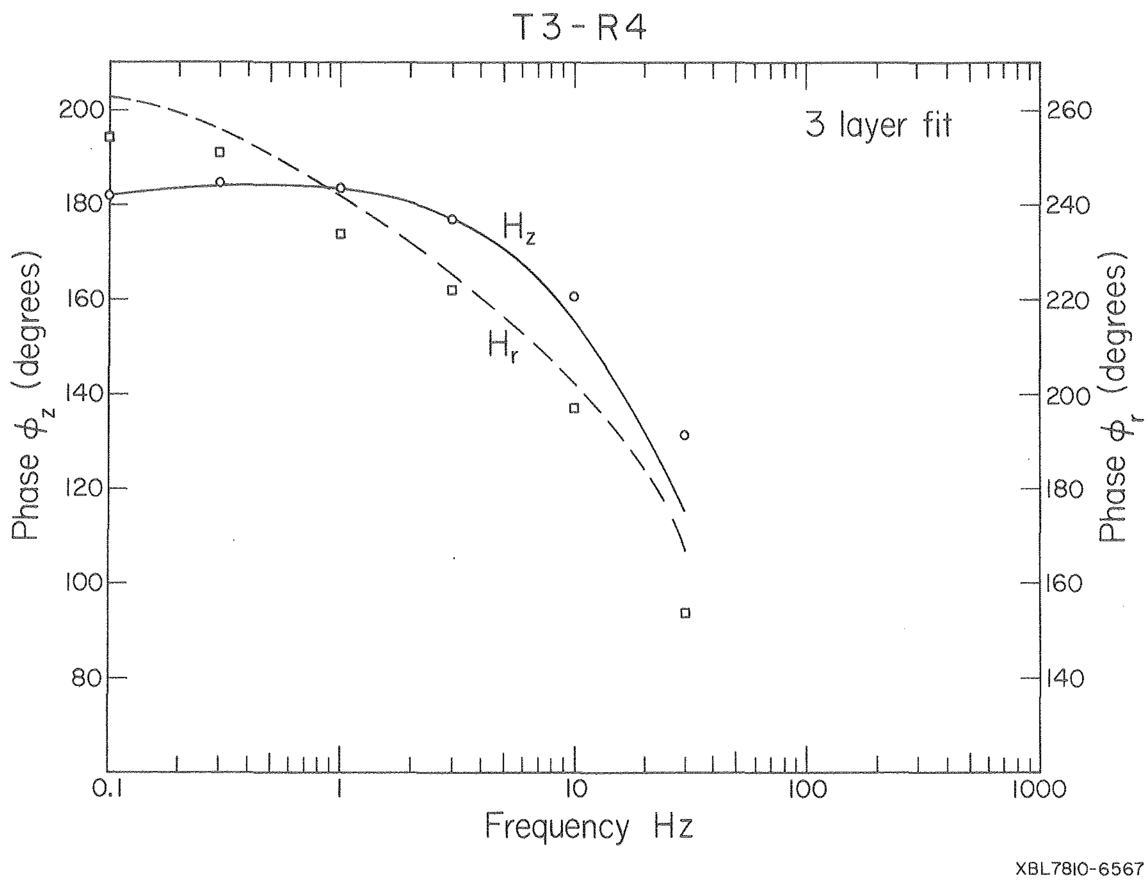


Figure 11. Phase spectra, T3-R4.

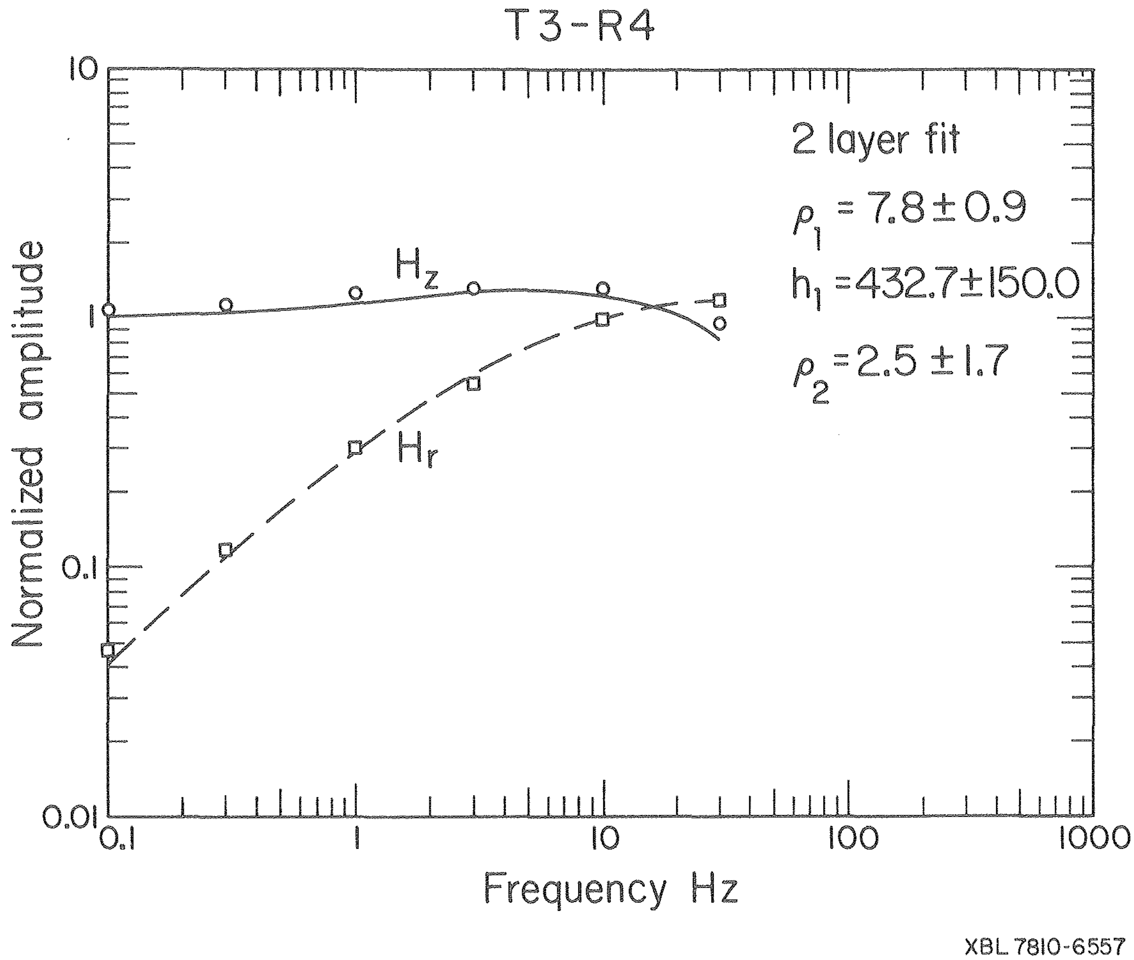


Figure 12. Amplitude spectra, T3-R4, two-layer fit.

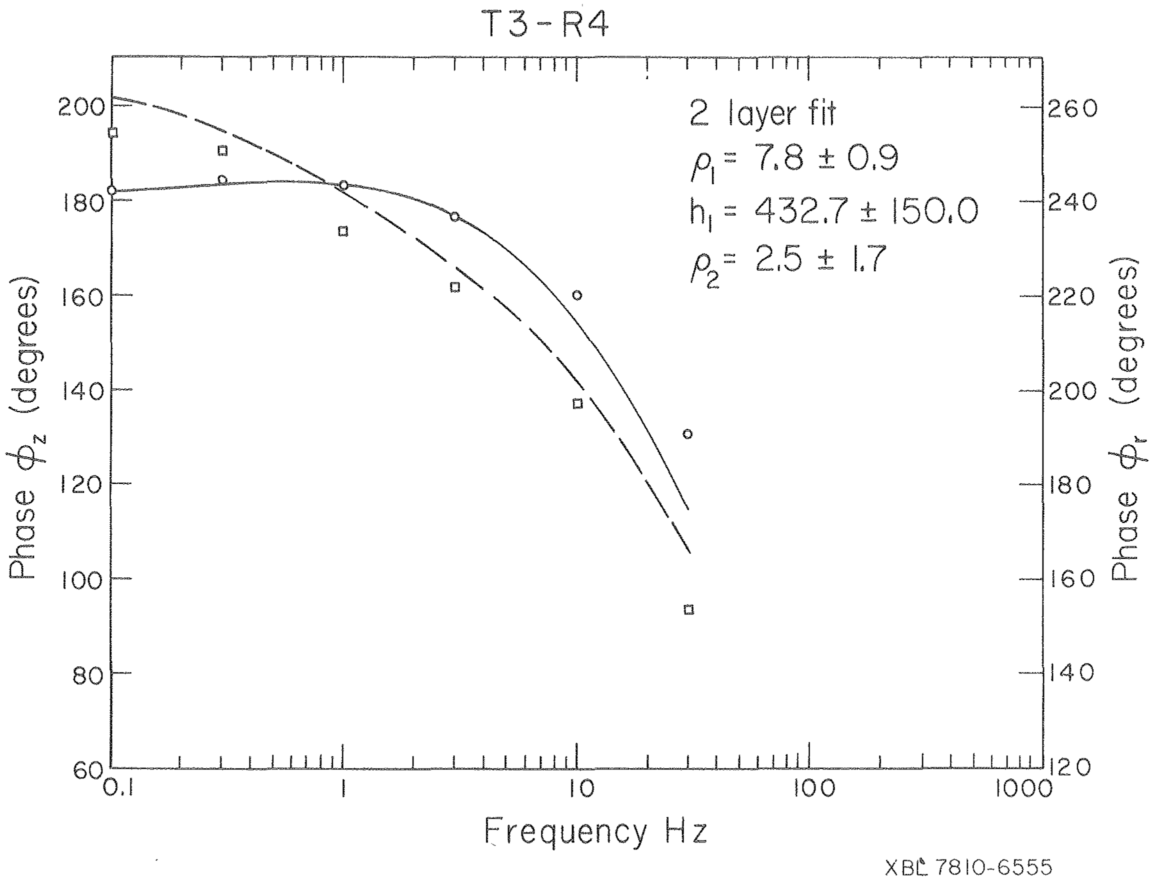


Figure 13. Phase spectra, T3-R4, two-layer fit.

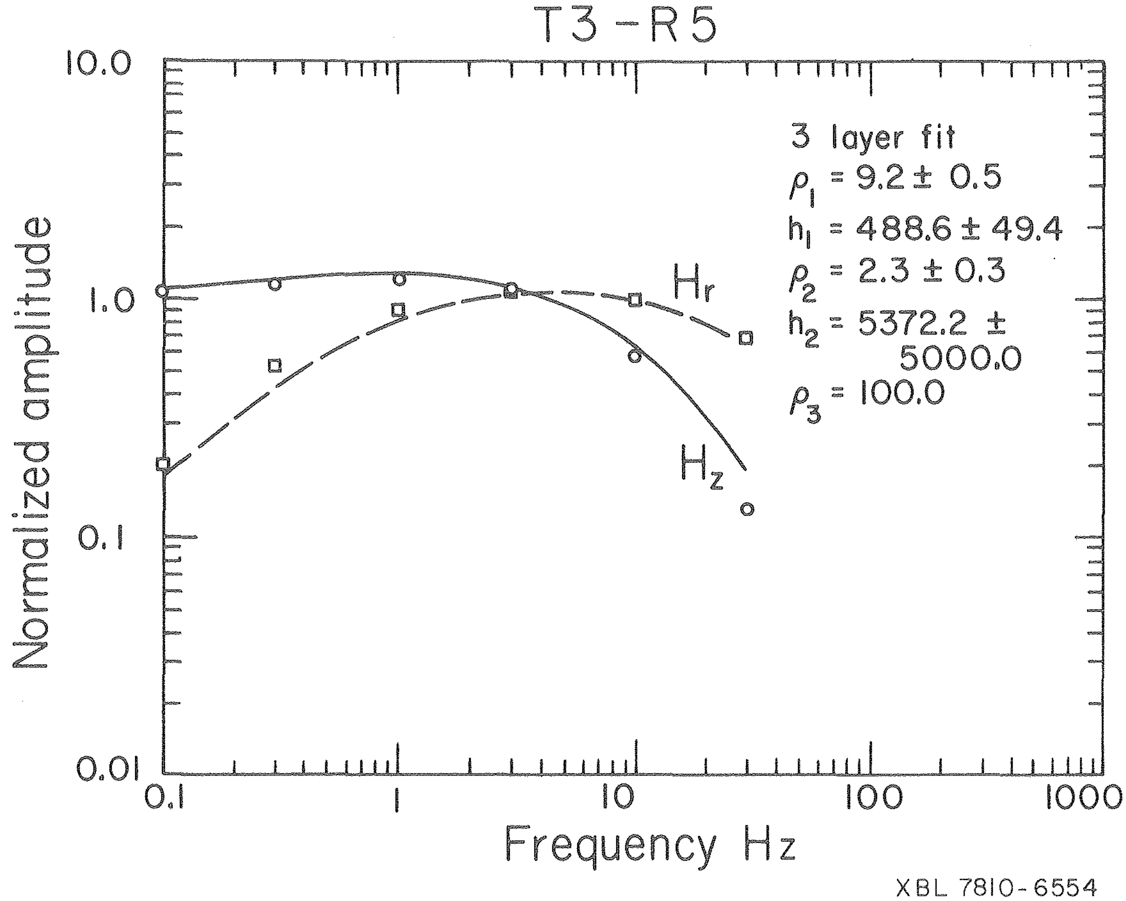


Figure 14. Amplitude spectra, T3-R5.

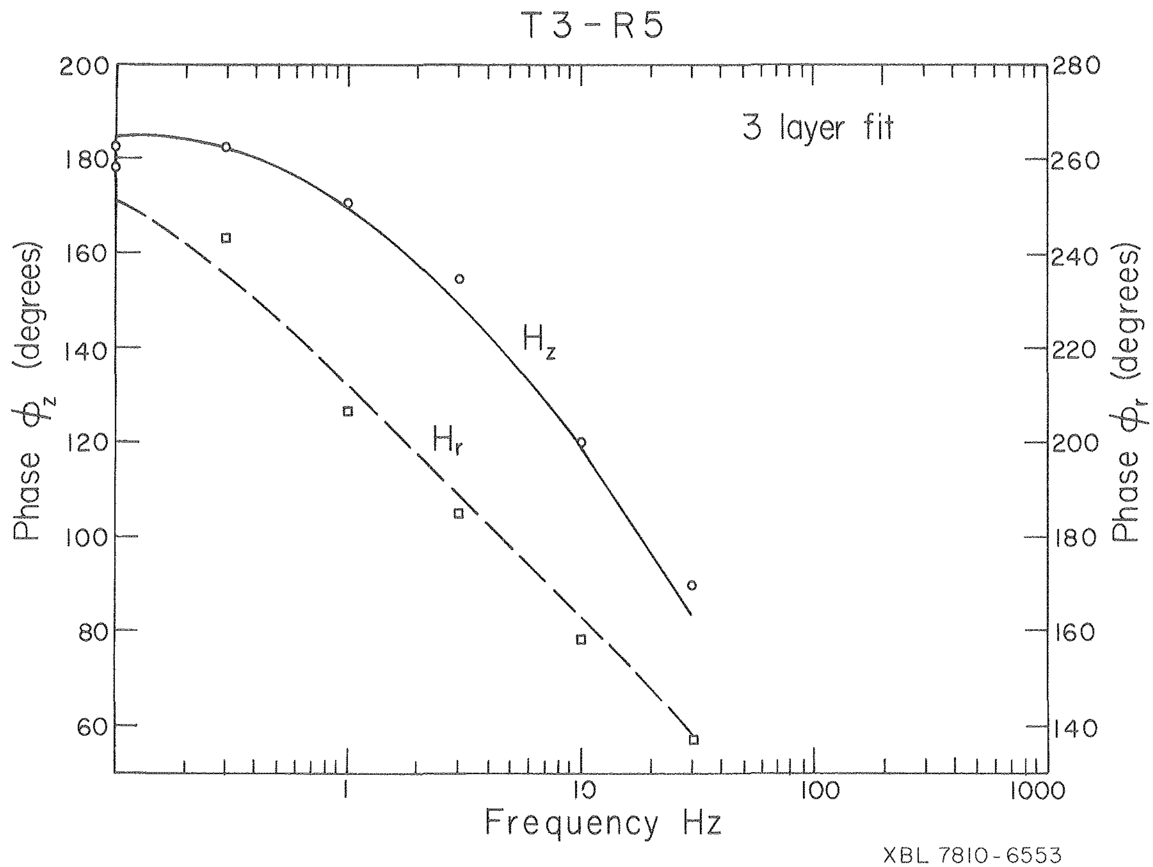
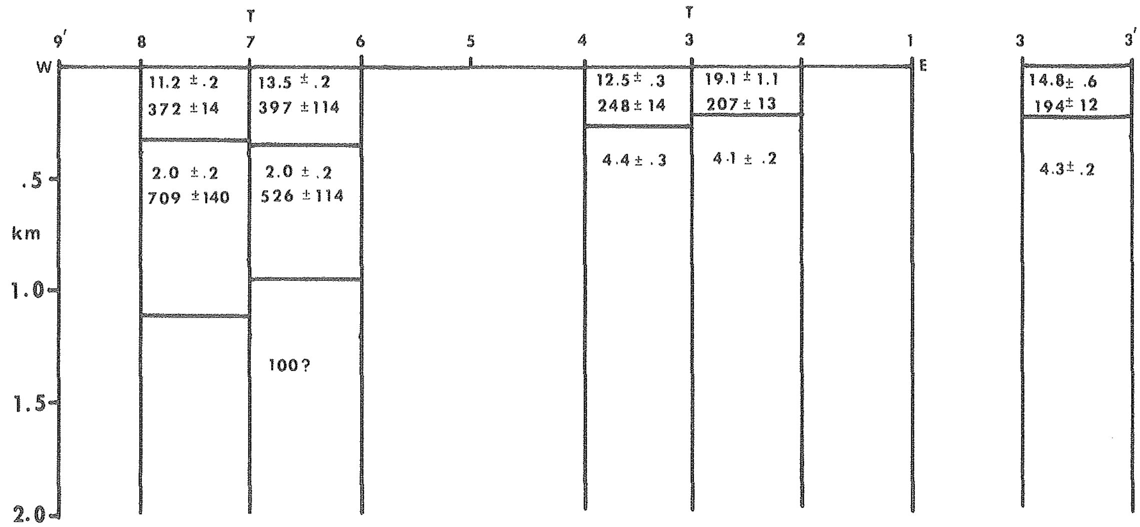
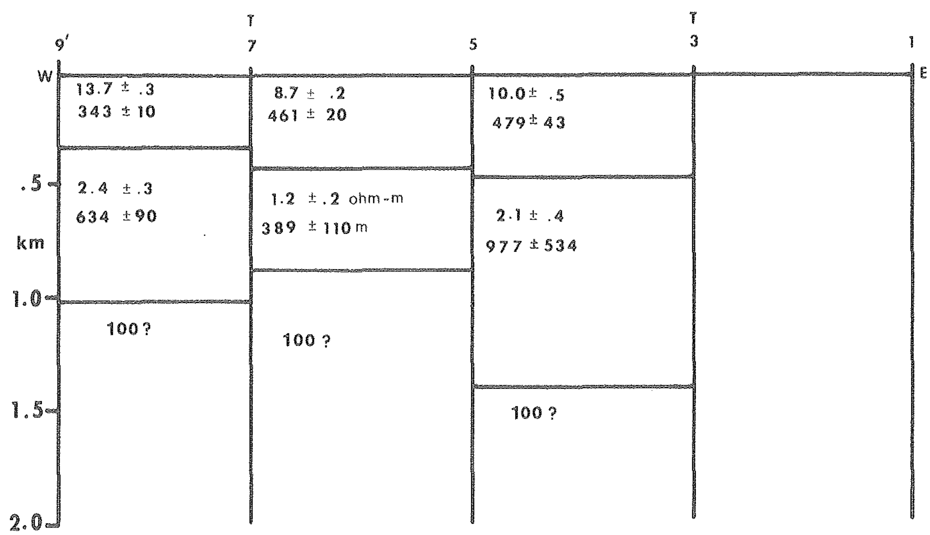


Figure 15. Phase spectra, T3-R5.





XBL 784-8029

Figure 16. Resistivity sections along Line E-E' obtained from interpretations of 1 km and 2 km sounding data (after Jain, 1978).

APPENDIX A

TABULATION OF RESULTS

Appendix A

T3-R1

Freq.	Normalized Field* H_z	H_r	Phase in Degrees** Φ_z	Φ_r
0.1	1.105±0.4	0.220±7.4	183.83±0.18	275.53±3.00
0.3	1.255±0.52	0.548±3.4	185.91±0.29	245.82±2.60
0.5	1.386±0.54	0.733±3.6	182.26±0.27	235,02±1.98
0.7	1.450±0.71	0.904±2.8	176.70±0.26	224.78±4.60
1.0	1.418±0.21	0.930±0.66	170.34±0.04	209.96±0.23
3.0	1.138±0.12	1.095±0.30	148.50±0.19	185.71±0.12
5.0	0.942±0.32	1.092±0.17	136.17±0.17	174.00±0.65
10.0	0.648±0.15	0.989±0.08	117.80±0.54	161.77±0.05
30.0	0.377±1.5	0.740±0.48	95.30±1.40	145.16±0.18

T3-R2

0.1	1.085±0.07	0.044±1.0	181.95±0.03	286.60±1.27
0.2	1.114±0.07	0.074±1.56	183.67±0.02	271.70±1.26
0.3	1.152±0.12	0.114±0.78	184.82±0.09	262.17±0.56
0.6	1.232±0.07	0.197±1.1	185.25±0.05	251.90±0.62
1.0	1.281±0.01	0.287±0.30	184.31±0.01	243.12±0.14
2.0	1.350±0.02	0.442±0.28	180.92±0.01	233.71±0.15
3.0	1.380±0.02	0.559±0.12	176.88±0.02	228.75±0.10
6.0	1.401±0.27	0.811±1.1	169.83±0.18	219.26±0.58
10.0	1.354±0.16	1.016±0.14	160.44±0.02	207.44±0.01
30.0	0.996±0.26	1.250±0.54	130.17±0.20	182.27±0.06
100.0	0.469±0.33	1.16 ±0.21	94.74±0.39	159.00±0.11
200.0	0.248±2.2	1.019±1.30	73.60±2.10	151.42±1.0

T3-R4

0.1	1.098±0.08	0.047±4.4	182.16±0.08	254.25±2.2
0.3	1.123±0.16	0.117±1.37	184.66±0.03	251.25±0.75
1.0	1.271±0.02	0.302±0.08	183.52±0.01	233.95±0.38
3.0	1.341±0.03	0.546±0.06	177.27±0.01	177.27±0.03
10.0	1.317±0.007	0.991±0.03	160.42±0.01	160.42±0.01
30.0	0.966±0.55	1.179±0.12	131.55±0.16	153.98±0.04
100.0	0.507±0.31	0.632±0.54	107.89±0.28	59.48±0.17

T3-R5

0.1	1.101±0.31	0.203±3.0	182.62±0.15	258.47±3.9
0.3	1.186±0.42	0.534±3.0	183.01±0.29	243.70±2.0
1.0	1.257±0.09	0.923±0.49	171.24±0.05	206.69±0.24
3.0	1.105±0.23	1.052±0.61	154.62±0.14	185.13±0.19
10.0	0.596±0.38	1.007±0.27	120.18±0.18	158.57±0.15
30.0	0.131±14.4	0.708±1.36	90.04±20.81	137.35±0.64

*Errors in percentages

**Errors in degrees

REFERENCES

REFERENCES

Bevington, P.R., 1969, Data reduction and error analysis for the physical sciences: McGraw Hill Book Company, Inc., New York.

Beyer, J.H., 1977, Telluric and d.c. resistivity techniques applied to the geophysical investigation of Basin and Range Geothermal Systems: Ph.D. thesis, University of California, Berkeley.

Beyer, J.H., Dey, A., Liaw, A., Majer, E., McEvilly, T.V., Morrison, H.F., and Wollenberg, H., 1976, Preliminary open file report: geological and geophysical studies in Grass Valley, Nevada, LBL-5262.

Beyer, J.H., Morrison, H.F., and Dey, A., 1975, Electrical exploration of geothermal systems in Basin and Range Valleys of Nevada; Proceedings: Second U.N. Symposium on the Development and Use of Geothermal Resources, May 1975, pp. 889-894.

Bhattacharyya, B.K., 19755, Propagation of transient electromagnetic waves in a conducting medium: Geophysics, v. 20, no. 4, pp. 959-961.

Dey, A., and Morrison, H.F., 1973, Electromagnetic coupling in frequency and time-domain induced polarization surveys over multilayered earth; Geophysics, v. 38, no. 2.

Frischknecht, F.C., 1967, Fields about an oscillating magnetic dipole over a two-layer earth, and application to ground and airborne electromagnetic surveys: Quarterly, Colorado School of Mines, v. 62, no. 1.

Ghosh, M.K., and Hallof, G., 1973, Loop-loop EM depth sounding method for delineating geothermal and permafrost zones: Abstract: Geophysics, c. 38, p. 1201.

Glenn, W.E., Ryu, J., Ward, S.H., Peeples, J.J., and Phillips, R.J., 1973, The inversion of vertical magnetic dipole sounding data: Geophysics, v. 38, pp. 1109-1129.

Glenn, W.E., and Ward, S.H., 1976, Statistical evaluation of electrical sounding methods. Part I: Experiment design: Geophysics, v. 41, pp. 1207-1221.

Harthill, N., 1976, Time domain electromagnetic soundings: IEEE Trans. on Geoscience Electronics, v. GE-14, pp. 256-260.

Inman, J.R., 1975, Resistivity inversion with Ridge regression: Geophysics, v. 40, pp. 798-817.

Inman, J.R., Ryu, J., and Ward, S.H., 1973, Resistivity inversion: Geophysics, v. 38, pp. 1088-1108.

REFERENCES
(continued)

Jackson, D.B., and Keller, G.V., 1972, An electromagnetic sounding survey of the summit of Kilauea Volcano, Hawaii: Journ. of Geophys. Res., v. 77, pp. 4957-4967.

Jain, B.K., 1978, A low frequency electromagnetic prospecting system: Ph.D. thesis, University of California, Berkeley, LBL-7042, 128 pp.

Jain, B.K., and Morrison, H.F., 1976, Progress report, inductive resistivity survey in Grass Valley, Nevada: Lawrence Berkeley Laboratory, UCID-3997, 46 pp.

Jupp, D.L.B., and Vozoff, K., 1975, Stable iterative methods for the inversion of geophysical data: Geophys. J.R. Astr. Soc., v. 42, pp. 957-976.

Keller, G.V., 1970, Induction method in prospecting for hot water: Geothermics - Special Issue 2.

Keller, G.V., and Rapolla, A., 1976, A comparison of two electromagnetic probing techniques: IEEE Trans. on Geoscience Electronics, v. GE-14, pp. 250-256.

Morrison, H.F., Hoversten, M., Riveros, C., and Oppliger, G., 1978, A low frequency electromagnetic prospecting system for geothermal exploration: Lawrence Berkeley Laboratory Report (in press).

Parker, R.L., 1970, The inverse problem of electrical conductivity in the mantle: Geophys. J.R. Astr. Soc., v. 22, pp. 121-138.

Ryu, J., Morrison, H.F., and Ward, S.H., 1970, Electromagnetic fields about a loop source of current: Geophysics, v. 35, pp. 862-896.

Smith, N.J., 1963, Geophysical activity in 1962: Geophysics, v. 28, no. 6, pp. 1048-1071.

Vanyan, L.L., 1967, Electromagnetic depth sounding: Consultants Bureau.

Wait, J.R., 1951, A conducting sphere in a time varying magnetic field: Geophysics, v. 16, no. 5, pp. 666-672.

Ward, S.H., 1977, A report on "Workshop on Electrical Methods in Geothermal Exploration": Geophysics, v. 42, pp. 664-666.

Ward, S.H., 1978, Program review: resource evaluation, reservoir confirmation, and exploration technology: University of Utah, Depart. of Geology and Geophysics, Technical Report 78-1701, b.5.1.

Wu, F.T., 1968, The inverse problem of magnetotelluric sounding: Geophyics, v. 35, pp. 972-979.

TECHNICAL INFORMATION DEPARTMENT
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720