

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.



UNIVERSITY OF ILLINOIS
URBANA

AERONOMY REPORT NO. 106

NEW ADVANCES IN THE PARTIAL-REFLECTION-DRIFTS EXPERIMENT USING MICROPROCESSORS

by
R. L. Ruggerio
S. A. Bowhill

December 1, 1982



Library of Congress ISSN 0568-0581

{NASA-CR-169891} NEW ADVANCES IN THE PARTIAL-REFLECTION-DRIFTS EXPERIMENT USING MICROPROCESSORS (Illinois Univ.) 90 p HC A05/MF A01 CSCL 17F 83-17771 G3/32 Unclass 08708

Supported by
National Aeronautics and Space Administration

Aeronomy Laboratory
Department of Electrical Engineering
University of Illinois
Urbana, Illinois

A E R O N O M Y R E P O R T

N O. 106

**NEW ADVANCES IN THE PARTIAL-REFLECTION-DRIFTS
EXPERIMENT USING MICROPROCESSORS**

by

**R. L. Ruggerio
S. A. Bowhill**

December 1, 1982

**Supported by
National Aeronautics and
Space Administration
Grant NSG 7506**

**Aeronomy Laboratory
Department of Electrical Engineering
University of Illinois
Urbana, Illinois**

ABSTRACT

Improvements to the partial-reflection-drifts experiment at Urbana, Illinois are completed. The results of the improvements include real-time processing and simultaneous measurements of the D region with coherent scatter. Preliminary results indicate a positive correlation between drift velocities calculated by both methods during a two-day interval.

The possibility now exists for extended observations between partial reflection and coherent scatter. In addition, preliminary measurements could be performed between partial reflection and meteor radar to complete a comparison of methods used to determine velocities in the D region.

PRECEDING PAGE BLANK NOT FILMED

TABLE OF CONTENTS

| | Page |
|---|------|
| ABSTRACT | iii |
| TABLE OF CONTENTS | iv |
| LIST OF FIGURES | vi |
| 1. INTRODUCTION AND STATEMENT OF THE PROBLEM | 1 |
| 2. THEORY OF PARTIAL-REFLECTION DRIFTS | 3 |
| 3. SYSTEM DESCRIPTION | 11 |
| 3.1 FORMER SYSTEM | 11 |
| 3.2 SYSTEM MODIFICATIONS | 15 |
| 3.2.1 Computer system | 17 |
| 3.2.2 A/D interface | 19 |
| 3.2.3 Pulser | 21 |
| 4. DATA COLLECTION AND ANALYSIS SYSTEM | 24 |
| 4.1 OBJECTIVES OF THE ANALYSIS | 24 |
| 4.2 COLLECTION SOFTWARE | 24 |
| 4.3 POSTPROCESSING SOFTWARE | 34 |
| 5. COMPARISON OF PARTIAL-REFLECTION AND COHERENT-SCATTER WIND DATA | 48 |
| 6. CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK | 56 |
| APPENDIX I COLLECTION PROGRAM | 58 |
| I.1 MEMORY USAGE FOR COLLECTION SOFTWARE | 65 |
| I.2 MEMORY MAP | 67 |
| APPENDIX II POSTPROCESSING PROGRAMS | 68 |
| II.1 3-ANTENNAS | 68 |
| II.2 4-ANTENNAS | 71 |
| APPENDIX III CYBER PLOTTING PROGRAMS | 74 |

| | |
|--|----|
| APPENDIX IV OPERATION MANUAL FOR PARTIAL-REFLECTION SYSTEM | 79 |
| REFERENCES | 82 |

LIST OF FIGURES

| Figure | Page |
|--|------|
| 2.1 Arrangement of receiving antennas | 5 |
| 2.2 Auto- and cross correlation functions | 7 |
| 2.3 Geometry of full-correlation analysis | 10 |
| 3.1 Block diagram of the former partial-reflection system. | 12 |
| 3.2 Partial-reflection antenna arrays at the Aeronomy Laboratory Field Station. | 13 |
| 3.3 Block diagram of the new partial-reflection system | 16 |
| 3.4 Block diagram of new computer system | 18 |
| 3.5 A/D interface circuit diagram | 20 |
| 3.6 Timing diagram for new pulser | 22 |
| 3.7 Circuit diagram for new pulser | 23 |
| 4.1a Flowchart for interrupt service routine (ISR) collection state. | 26 |
| 4.1b Flowchart for (ISR) wait state | 27 |
| 4.1c Flowchart for (ISR) stall state | 28 |
| 4.2 Flowchart for main code DCORR. | 30 |
| 4.3 Method used to calculate correlations functions | 35 |
| 4.4a Flowchart for 4 antenna postprocessing software | 36 |
| 4.4b Flowchart for computation of apparent and true velocity | 39 |
| 4.4c Flowchart for computation of ellipse | 40 |
| 4.4d Flowchart for curve fitting of cross correlations | 41 |
| 5.1 Scatter plot at 75 km. | 49 |
| 5.2 Scatter plot at 79.5 km. | 50 |
| 5.3 Scatter plot at 84 km. | 51 |

| Figure | | Page |
|--------|---|------|
| 5.4 | Scatter plot at 93 km. | 52 |
| 5.5 | Scatter plot at 97.5 km. | 53 |
| 5.6 | Drift velocity as a function of velocity perturbation factor | 55 |

1. INTRODUCTION AND STATEMENT OF THE PROBLEM

The purpose of this thesis is to describe how the D region of the atmosphere is remotely probed by a ground-based radar system used at the Aeronomy Field Station near Urbana. The method used is referred to as the partial-reflection-drifts experiment. Other experiments which use different methods at this installation include coherent scatter and meteor radar.

The partial-reflection-drifts experiment is used to determine the horizontal-drift velocity of ionized irregularities in the ionosphere. If a point radio source is used, the horizontally stratified irregularities produce a diffraction pattern over the ground. By sensing this diffraction pattern with a minimum of three antennas the horizontal-drift velocity can be computed.

The former system at the Aeronomy Field Station lacks the ability to produce reasonable amounts of data so that the dynamics of the upper atmosphere can be studied. The amount of computer time needed to process data is one of the limiting factors of the former system. Since the computer used for the drifts experiment also is used by the other experiments, simultaneous measurements for comparing techniques are difficult. The system also lacks a suitable storage medium for the results so that advanced studies could be performed.

The new system solves the problem by modifying the way data are collected, processed, and stored. By using small inexpensive microcomputers to distribute the computation load, simultaneous measurements become possible. The postprocessing computation time can be reduced by performing some calculations during the collection of data. The ability to store results on a floppy disk makes the storage medium ideal for postanalysis of results.

The major objective of these enhancements is to optimize the usefulness of the partial-reflection-drifts technique as a valid method for studying the dynamics of the upper atmosphere.

2. THEORY OF PARTIAL-REFLECTION DRIFTS

To determine the horizontal-drift velocity of the ionosphere it is necessary to illuminate the ionosphere with a single radio-wave point source. When this is done a diffraction pattern is formed from the ionized irregularities in the D region. According to Briggs (1977) the ions in the region of interest move with the neutral air, the drift velocity of the ions can be measured by applying the method developed by Briggs et al., (1950) known as the full-correlation analysis. The radio waves are directed at vertical incidence, it is therefore necessary to have a system of spaced antennas to track the motion of the irregularities. With three spaced antennas to sample the amplitude of the diffraction pattern observed on the ground, spatial properties and the movement of the pattern can be deduced.

The diffraction pattern is sampled at all antennas at equally spaced time intervals. The resulting records of amplitude represent the fluctuation of the diffraction pattern as it moves across the ground. The method of analysis applied to the recorded amplitudes requires the autocorrelation for each observation point and cross correlation for every pair of observation points. It is assumed that the information contained in these correlation functions can completely determine the velocity, direction of drift, the size and the orientation of the irregularity which is inferred from contours of constant correlation which are in the form of ellipses (Briggs 1968).

The signal amplitude of the diffraction pattern can be represented by a function $R(x,y,k)$, where x and y represent the space coordinates and k the time coordinate. With the amplitudes of the diffraction pattern defined by a function of three variables, two of them being spatial variables, the corresponding correlation function is also a function of space and time.

For the practical analysis the spatial variables are removed and the distances and direction of the baselines between antennas are recorded for the latter parts of the analysis. By numbering the antenna system as shown in Figure 2.1, the discrete correlation functions are functions of time only and have the general forms;

$$P_{ii}(K) = \frac{\frac{1}{N-T} \sum_{K=1}^{N-T} A_i(K) * A_i(K+1) - \left(\frac{1}{N-T} \sum_{K=1}^{N-T} A_i(K) \right)^2}{\left[\frac{1}{N} \sum_{K=1}^N A_i^2(K) - \left(\frac{1}{N} \sum_{K=1}^N A_i(K) \right)^2 \right]}$$

for the autocorrelations, and

$$P_{ij}(K) = \frac{\frac{1}{N-T} \sum_{K=1}^{N-T} A_i(K) * A_j(K+T) - \frac{1}{N-T} \left(\sum_{K=1}^{N-T} A_i(K) \right) \left(\sum_{n=1}^{N-T} A_j(K) \right)}{\left[\frac{1}{N} \sum_{K=1}^N A_i^2(K) - \left(\frac{1}{N} \sum_{K=1}^N A_i(K) \right)^2 \right] \left[\frac{1}{N} \sum_{n=1}^N A_j^2(K) - \left(\frac{1}{N} \sum_{n=1}^N A_j(K) \right)^2 \right]}$$

for the cross correlations.

The computed autocorrelations from each antenna should be identical, but due to statistical variations they are not in practice and the mean autocorrelation is used for the analysis. The cross correlation function has a maximum value displaced from the origin by some amount t' . The greater the velocity of the irregularities the smaller t' is. It is important, therefore, to separate the antennas by a sufficient distance so that t' is measurable, but not to space them so far apart that the cross correlation is zero (Briggs 1976). For the frequency used for our antenna system the optimum separation for the antennas is about 160 meters for D-region measurements. The antenna separation for the Urbana drifts experiment is 169 meters for the shorter sides of the triangle and 240 meters for the hypotenuse side.

The typical shape of the resulting auto- and cross correlation func-

ORIGINAL PAGE IS
OF POOR QUALITY

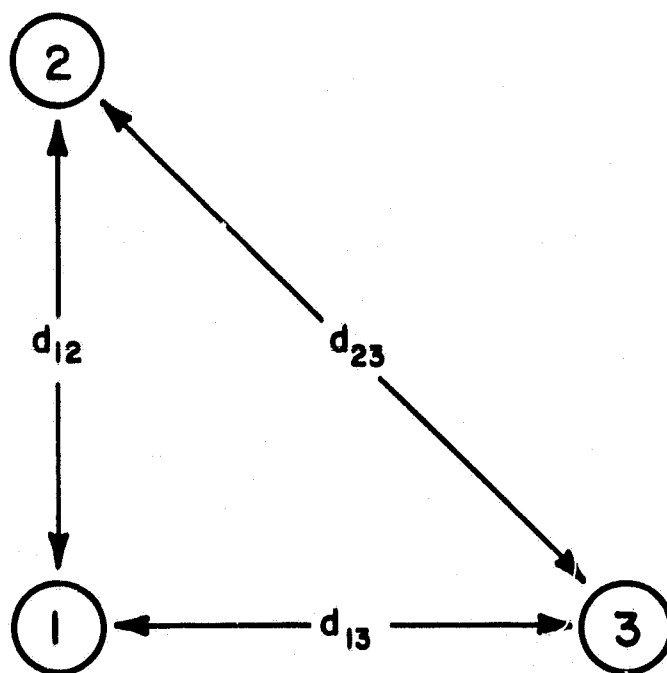


Figure 2.1 Arrangement of receiving antennas.

tions resembles those in Figure 2.2.

The information that is extracted from the correlation function for the full correlation analysis is;

- 1) The maximum value of the cross correlation $(p_m)_{ij}$ and the associated time displacement $(t')_{ij}$ from zero lag.
- 2) The values of time displacement $(t_m)_{ij}$ from the autocorrelation that correspond to the maximum values $(p_m)_{ij}$ found for the cross correlation.

The method that was first used for analyzing drifts records was known as the method of similar fades (Mitra 1949). This was a simple approach to the problem of determining the drift velocity. The major disadvantage of using this method is that it assumed the irregularity did not change in shape as it moved across the observation points. The full correlation analysis corrects this assumption by computing the correlation contour that represents the fading of the drifting pattern with no random changes. It is assumed that this contour is in the shape of an ellipse but there is no physical reason why this should be so, and it is possible that the contour might be of any shape whatever (Briggs et al., 1950). To calculate the ellipse the values of $(t_m)_{ij}$ and $(t')_{ij}$ are used in the formula;

$$(t_{ij})^2 = (t')_{ij}^2 - (t_m)_{ij}^2$$

where (t_{ij}) corresponds to the time shift for the autocorrelation to take on the value of the cross correlation at zero lag. By using the formula above the values of t_{ij} are less affected by statistical variation because it is computed at a higher level on the correlation functions (Briggs 1957).

Three velocity vectors defined by

$$\text{MAG}|V_{ij}| = d_{ij}/t_{ij}$$

$$\text{ANGLE}|V_{ij}| = \theta_{ij}$$

ORIGINAL PAGE IS
OF POOR QUALITY

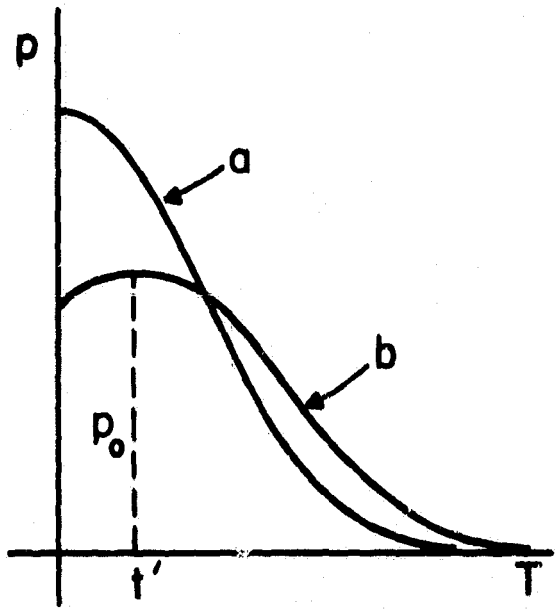


Figure 2.2 (a) Autocorrelation function.
(b) Cross correlation function.

are computed. The d_{ij} 's are the baseline distances in meters and the θ_{ij} 's are the angles measured from receiver "i" to receiver "j" with respect to the north. The ellipse that defines this contour of constant correlation is referred to as the characteristic ellipse.

The characteristic ellipse has the dimensions of velocity. If the ellipse is multiplied by the half time of the autocorrelation, the resulting ellipse will have dimensions of meters and gives the relative size of the irregularity. The ellipse is rotated through some angle θ_0 and has semimajor axis a and semiminor axis b . The axial ratio a/b gives an indication of the elongation of the irregularity and the angle θ_0 gives the direction of elongation.

The next calculation necessary is to determine the apparent velocity components between the antenna pairs. The three vectors

$$V'_{12} = d_{12}/\tau'_{12} \quad \theta_{12}$$

$$V'_{13} = d_{13}/\tau'_{13} \quad \theta_{13}$$

$$V'_{23} = d_{23}/\tau'_{23} \quad \theta_{23}$$

are computed and the end points will lie on a straight line referred to as the V' -line. If a vector perpendicular to this line is drawn from the origin it will define the magnitude and direction of the apparent velocity. The important information that is used from this is the slope of the V' -line defined by these vectors. The point of tangency of this line to the ellipse gives the true direction of drift ϕ measured clockwise from north. The vector from the origin to the tangency point is known as the $(V'_c)_v$ vector. The true velocity is then given by;

$$V_{\text{drift}} = \frac{(V'_c)_v}{V'}$$

This velocity must be divided by a factor of two to yield the correct

results due to the point source effect. The east and north components of the drift are given by;

$$V_E = V_{\text{drift}} \sin\phi \qquad V_N = V_{\text{drift}} \cos\phi$$

The geometry of the full-correlation analysis is shown in Figure 2.3.

The full-correlation analysis has been used for several years. During that time many investigators in the field have done extensive studies to determine the validity of the analysis, by producing models of the ionosphere and performing computer simulations (Pitteway et al., 1971). The best way to determine the validity of this analysis is to compare results from simultaneous measurements. This is done in Chapter 5.

ORIGINAL PAGE IS
OF POOR QUALITY

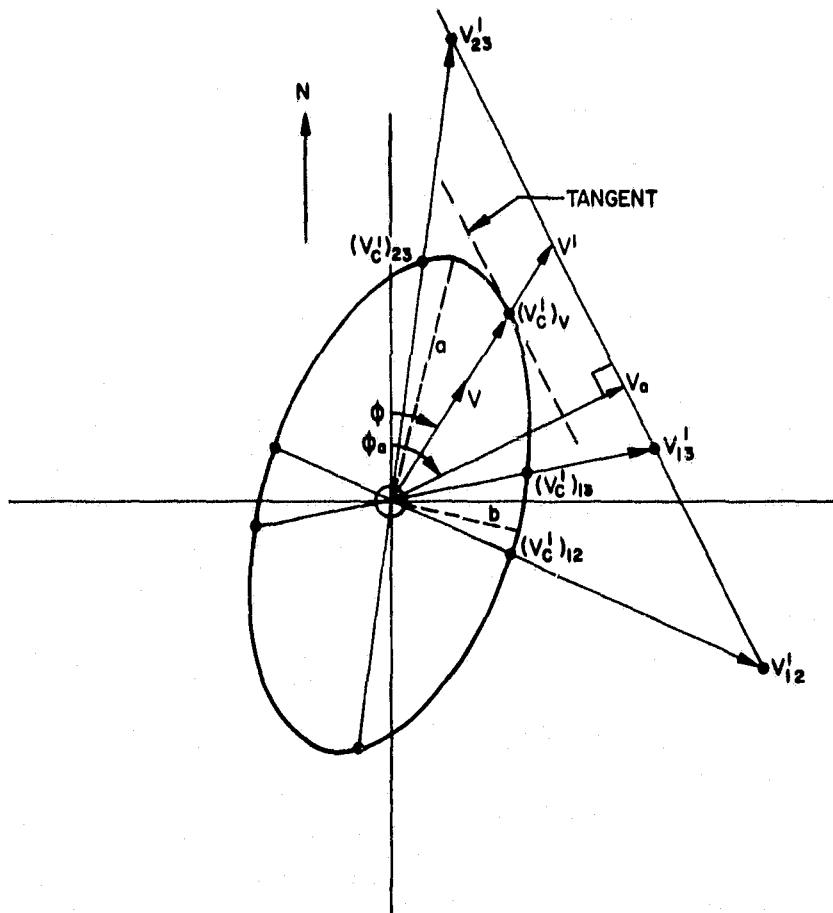


Figure 2.3 Geometry of full-correlation analysis.

3. SYSTEM DESCRIPTION

3.1 FORMER SYSTEM

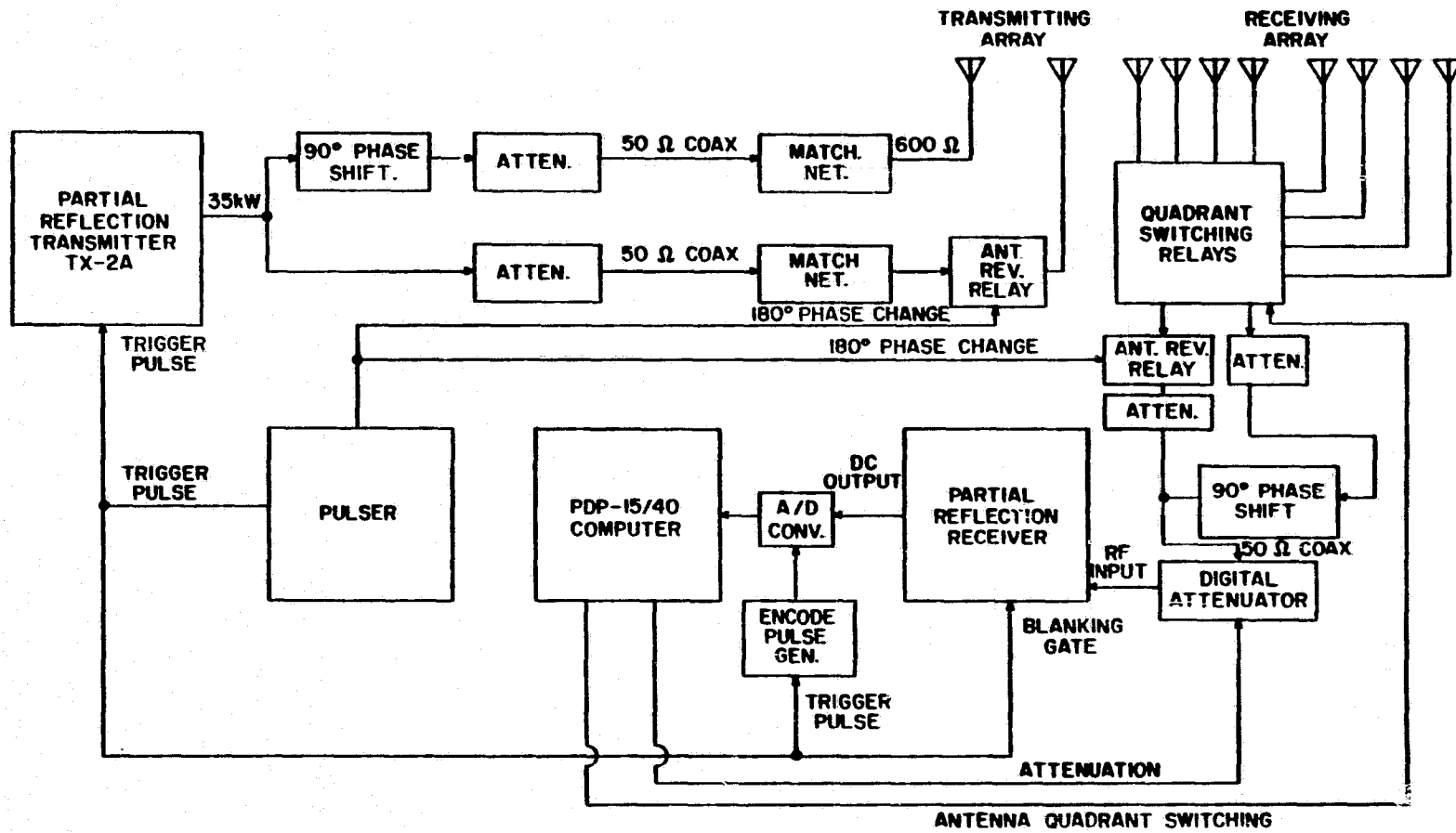
The partial-reflection system supports two experiments that are useful for D-region studies. The experiment that is being enhanced is the drifts experiment, the other is the differential-absorption experiment which is used to measure electron-density concentrations. A block diagram of the former Urbana partial-reflection system is shown in Figure 3.1. The major elements of this system are the transmitter, transmitting and receiving antennas, receiver, data-acquisition system, and radar controller.

The transmitter is a multistage tube type and is fully described by Henry (1966) and by Pirnat and Bowhill (1968). The final output is split between two 50 ohm coaxial cables. This is done to insert a phase shift of 90 degrees so that when the two outputs are fed into an orthogonal-dipole array the transmitted wave will be circularly polarized. Attenuators are used after the phase shift to ensure circular polarization. The characters of this transmitter are:

| | |
|-----------------------|--------------------|
| Peak Power | 35 kW |
| Frequency | 2.66 MHz |
| Pulse Width | 23 μ sec |
| Output Impedance | 50 ohm, unbalanced |
| Pulse Repetition Rate | 10 Hz |

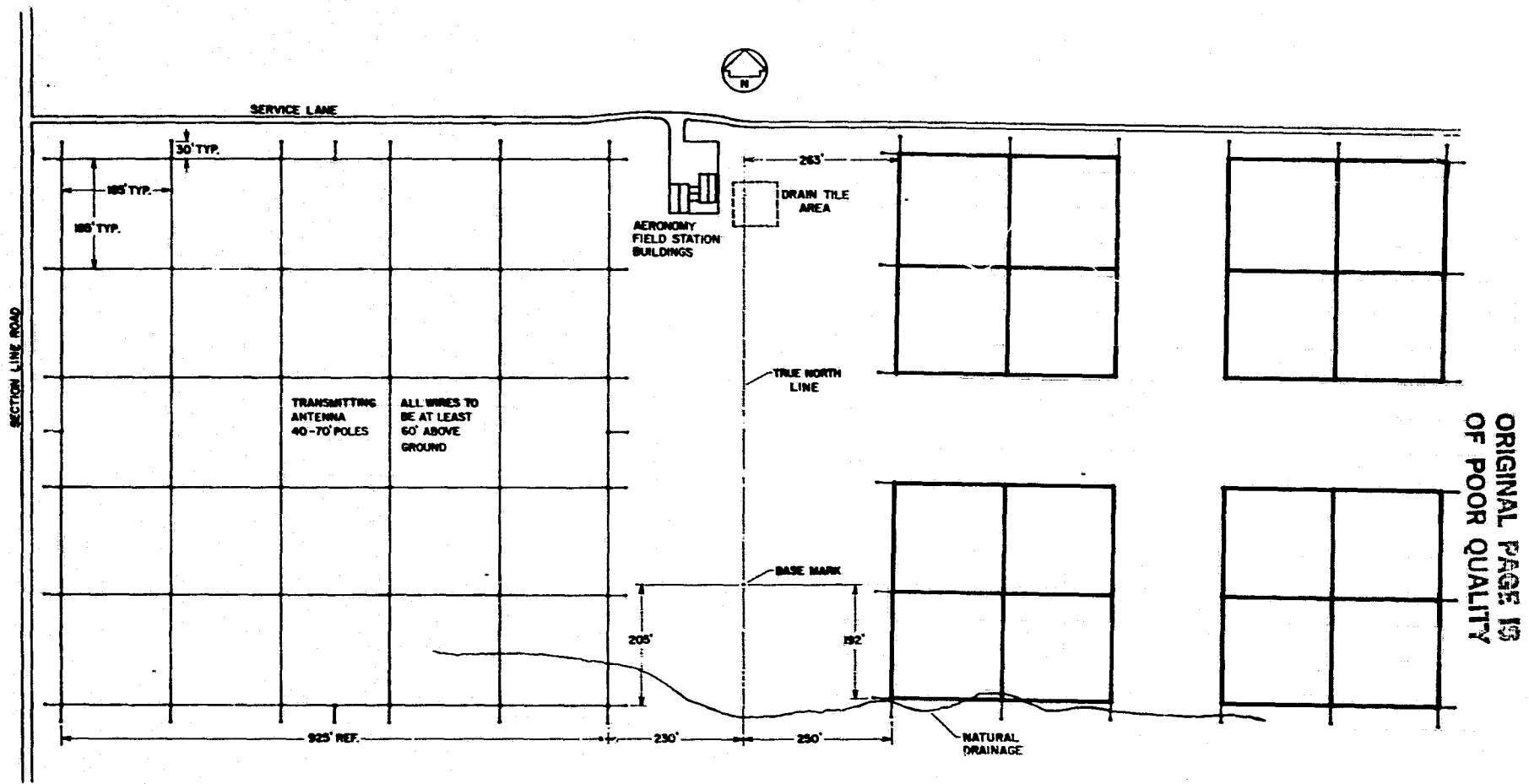
The layout of the antenna system is shown in Figure 3.2 (Weiland and Bowhill 1981). The array west of the Field Station Building is the transmitting array and the array to the east is the receiving array. Both antennas were identical when they were first constructed. The receiving array has been divided into four quadrants so that the drifts experiment can be implemented.

The transmitting array consists of 60 half-wave dipoles. The dipoles in the N-S and E-W direction are coupled together to form an orthogonal set.



ORIGINAL PAGE IS
OF POOR QUALITY

Figure 3.1 Block diagram of the former partial-reflection system.



ORIGINAL PAGE IS
OF POOR QUALITY

Figure 3.2 Partial-reflection antenna arrays at the Aeronomy Laboratory Field Station.

This is done to achieve circularly-polarized-transmitted waves. By inserting a relay network that inserts a 180 deg phase shift, the ordinary or extraordinary pulse can be transmitted. For the drifts experiment only one polarization is necessary and the ordinary is preferred over the extraordinary because the ordinary mode is attenuated less in the D region.

The receiving array consists of four smaller arrays called quadrants, that contain three full wavelength dipoles in both the N-S and E-W direction. The N-S and E-W dipoles are separately matched to one 50 ohm coaxial cable that is fed into the Field Station Building. The selection of any quadrant is accomplished by a relay network mounted on the inside east wall of the Field Station Building.

The receiver used for partial reflection is extremely linear. This is a requirement because the signal amplitudes must be accurately measured over a 50 dB range (Weiland and Bowhill 1978). The characteristics of the receivers are listed below:

| | |
|--------------------|--|
| Center Frequency | 2.66 MHz |
| Noise Figure | 3 dB |
| Bandwidth | 40 kHz |
| Recovery Time | 200 μ sec after removal of .1V RMS input |
| Gain Variation | 3 dB max |
| Rf Input Impedance | 50 ohm, unbalanced |
| Output Impedance | 10,000 ohms |
| Linearity | 55 dB for 1 dB deviation |

The data-acquisition system for the former system consisted of a Digital Equipment Corporation PDP 15/40 computer for processing of data and a Hewlett-Packard 5610A analog-to-digital converter for digitizing data from the receiver. The output of the A/D was a 10-bit word in two's complement form. The entire range of the A/D was not used however since the input range for the A/D was 1 volt and the output of the receiver is zero to +1 volt. The conversion rate is 100 kHz and corresponds to a height resolution

of 1.5 km.

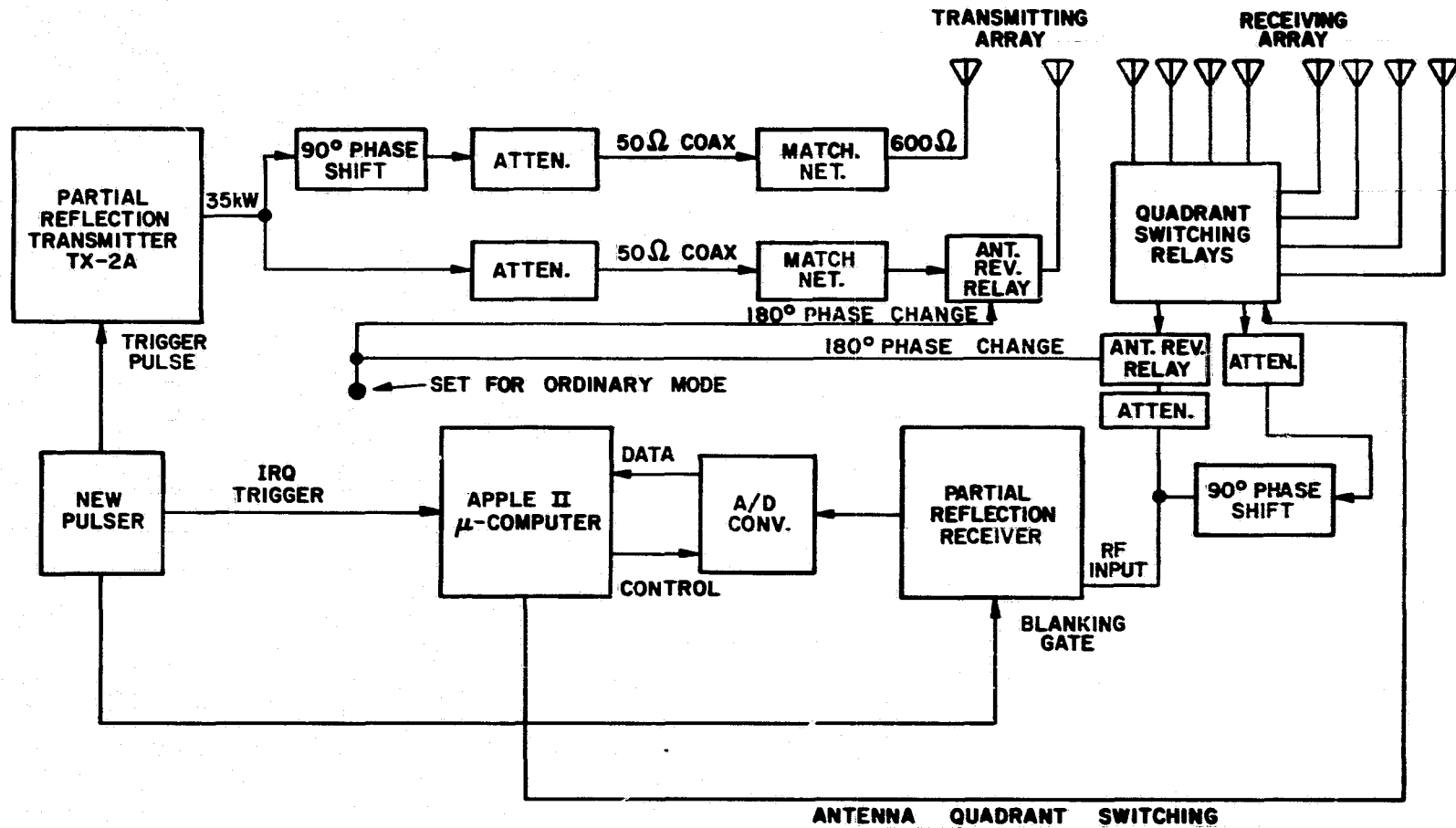
The radar controller referred to as the pulser generates the necessary control signals to operate the system. The pulser generated a 25-microsecond pulse to turn on the transmitter, a 100-microsecond long pulse to blank the receiver, a trigger pulse to activate the A/D and a logic signal to switch between the ordinary and extraordinary modes.

3.2 SYSTEM MODIFICATIONS

The purpose of modifying the system was to improve its performance and adapt the system to work with a microcomputer. The goals of these improvements are to:

- 1) Increase amount of data that can be taken daily.
- 2) Decrease postprocessing time.
- 3) Enable simultaneous measurements between the drifts experiment and the coherent-scatter or meteor-radar experiment.
- 4) Form a data base of results for advance analysis.

A block diagram of the new system is shown in Figure 3.3. The components of the former system that were replaced include the computer, pulser, and the A/D converter. The new system does not require both polarizations for collecting drift data. The new system therefore does not control the phase shifting networks for the extraordinary mode. The ordinary mode is preferred. The new pulser increases the pulse repetition frequency (PRF) from 10 Hz to 200 Hz. It was considered that by increasing the PRF to this rate the signal could be oversampled and would improve the signal-to-noise ratio. The Apple II computer is being used to gain independence from other experiments that use the PDP-15 computer. A new A/D converter is also necessary for compatibility with the Apple computer. With partial-reflection drifts independent from other experiments, simultaneous measurements can be per-



ORIGINAL, PAGE 18
OF POOR QUALITY

Figure 3.3 Block diagram of new partial-reflection system.

formed. The organization of the computer system will accomplish the rest of the desired goals.

3.2.1 Computer system. A block diagram for this computer system is shown in Figure 3.4. The computer system used for the drifts experiment consists of an Apple II Plus for the collection of data, an Apple II Plus for processing of data, a 10 megabyte Winchester disk drive for data storage, and various peripherals for the Apple II computer. The peripherals consist of a real time clock, a parallel interface, a modem, and an arithmetic processor. The function of each of the peripherals is summarized below.

1) Real-Time Clock. This peripheral generates the necessary information to record the date and time with each record of collected data.

2) Parallel Interface. This peripheral forms the interface to the A/D converter and the relay network for quadrant switching. It contains four parallel ports that can be used for input or output. It also has the ability to generate square wave signals that are used to synchronize the collection program with the A/D converter. The main component on this interface card is the 6522 versatile-interface adapter. It is an all purpose parallel-interface adapter for the 6502 eight bit μ processor chip which is the Apple's CPU.

3) Arithmetic Processor. This powerful peripheral enhances the computation power of the Apple. It is used to perform the necessary multiplications for the correlation functions. The processor card uses the AMD 9511 arithmetic processor chip. This integrated circuit can perform a variety of operations including transcendental functions. The time required to perform a 32-bit by 32-bit multiplication is 105 microseconds.

4) Modem. This peripheral along with the necessary software enables data transfers between computers. This modem is a direct-connect type with

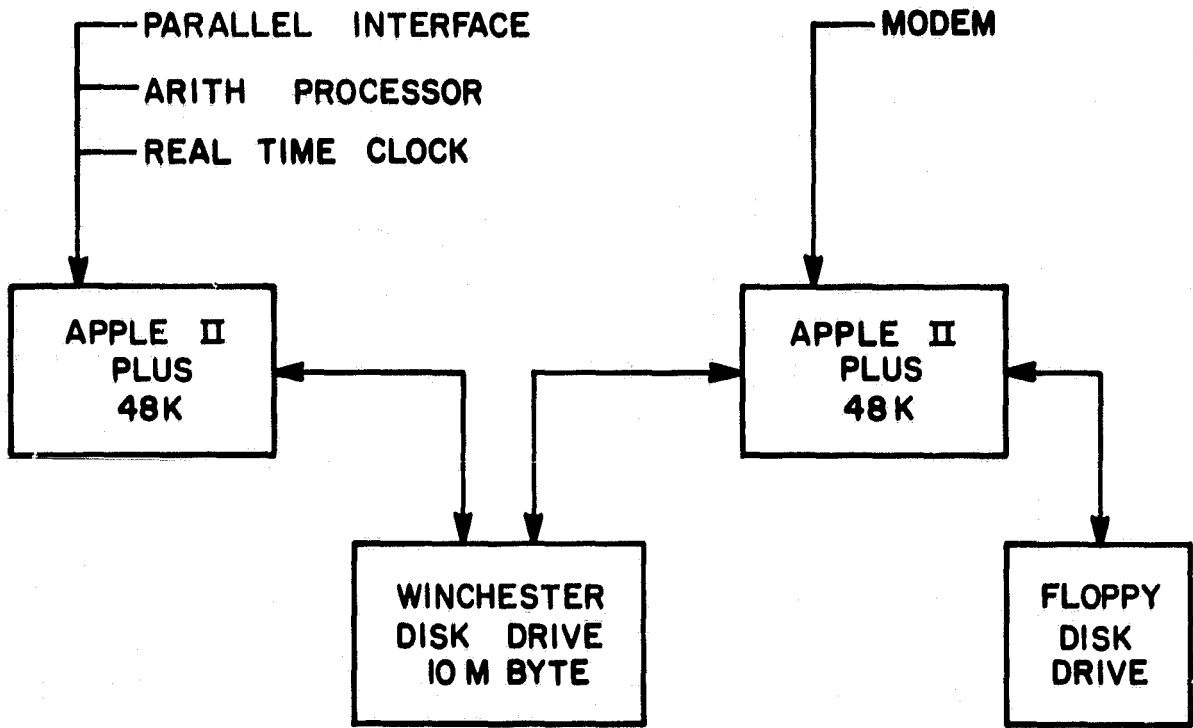


Figure 3.4 Block diagram of new computer system.

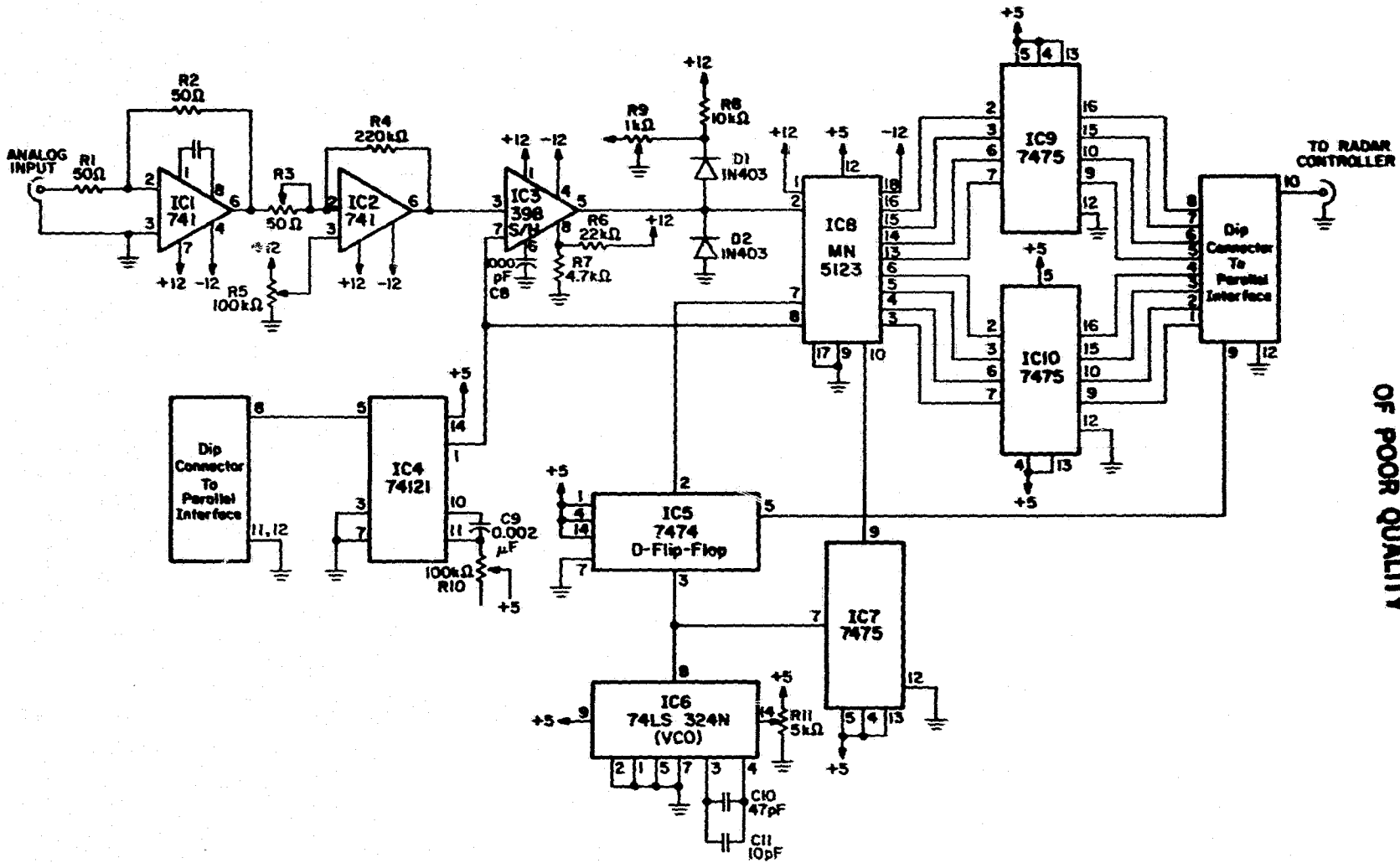
a transfer rate of 300 baud.

3.2.2 A/D interface. This interface performs the necessary analog-to-digital conversion for the drifts experiment. The interface converts the amplitude detected signal from the receiver to an eight-bit unsigned binary number. This binary number is transferred into the computer by means of a parallel interface. The circuit diagram for the interface is shown in Figure 3.5.

The A/D performs 10^5 conversions per second. This corresponds to a range resolution of 1.5 km. The main component of the interface is IC8. This integrated circuit package is a successive approximation type A/D with a conversion speed of six microseconds. For a sampling rate of 100 kHz that leaves four microseconds to sample the signal and latch the data to the parallel interface.

Only one control signal is needed to drive the interface and is generated by the parallel interface by means of an internal timer. This signal is a symmetric-square wave with a frequency of 100 kHz. The purpose of IC4 is to produce an asymmetric-square wave since the sampling and conversion time cannot be equal for the A/D used.

The operational amplifiers IC1 and IC2 are used to buffer the input, amplify, and level shift of the signal if necessary to accommodate the analog-to-digital converter. For the A/D used the level shifting ability is not used but the signal must be amplified by a factor of ten to take advantage of the range of the A/D. Because damage may occur if the input to IC8 is greater than +10 volts or less than zero, D1 and D2 are used to limit the signal to this range. The free running voltage-controlled oscillator (VCO) chip IC6 is used as the clock for the A/D and is buffered by IC7. The clock frequency for a six-microsecond conversion time is 1.33 MHz. The digital outputs of



ORIGINAL PAGE IS
OF POOR QUALITY

Figure 3.5 A/D-interface circuit diagram.

the A/D are buffered by IC9, IC10 and IC5.

3.2.3 Pulser. The pulser is the main controlling unit of the drifts experiment. This unit produces the necessary control signals for the transmitter, receiver blanker, and computer. The timing diagram that shows the relationship between these signals is shown in Figure 3.6, and the circuit diagram is shown in Figure 3.7.

All the control signals of the pulser are derived from IC1 which is a VCO. The resulting signal from this chip is fed through divide-by-two and divide-by-ten counters to produce five separate PRF rates. The desired PRF rate is used to drive four monostable circuits which produce the control signals.

The ability to choose the PRF rate allows some flexibility for future work with the drifts or the electron density experiment. The pulse width of the transmitted pulse can easily be varied by adjusting the potentiometer on IC4. When adjusting the PRF and pulse width the maximum duty cycle of the final stage amplifier of the transmitter must not be exceeded.

The blanker signal is set high before, during, and after the transmitter pulse is activated. The interrupt request control signal for the computer is adjustable by means of a panel mount potentiometer. The adjustment of this signal determines the approximate starting height where data are collected by the computer.

Due to the problems observed when both transmitters were operating for partial reflection and coherent scatter, the pulser was modified. The PRF of the coherent-scatter transmitter is 400 Hz and the TTL level signal that was generated by the radar controller for coherent scatter was used in place of IC1. The circuits that generate various PRF rates were disconnected until the need for them arises.

ORIGINAL PAGE IS
OF POOR QUALITY

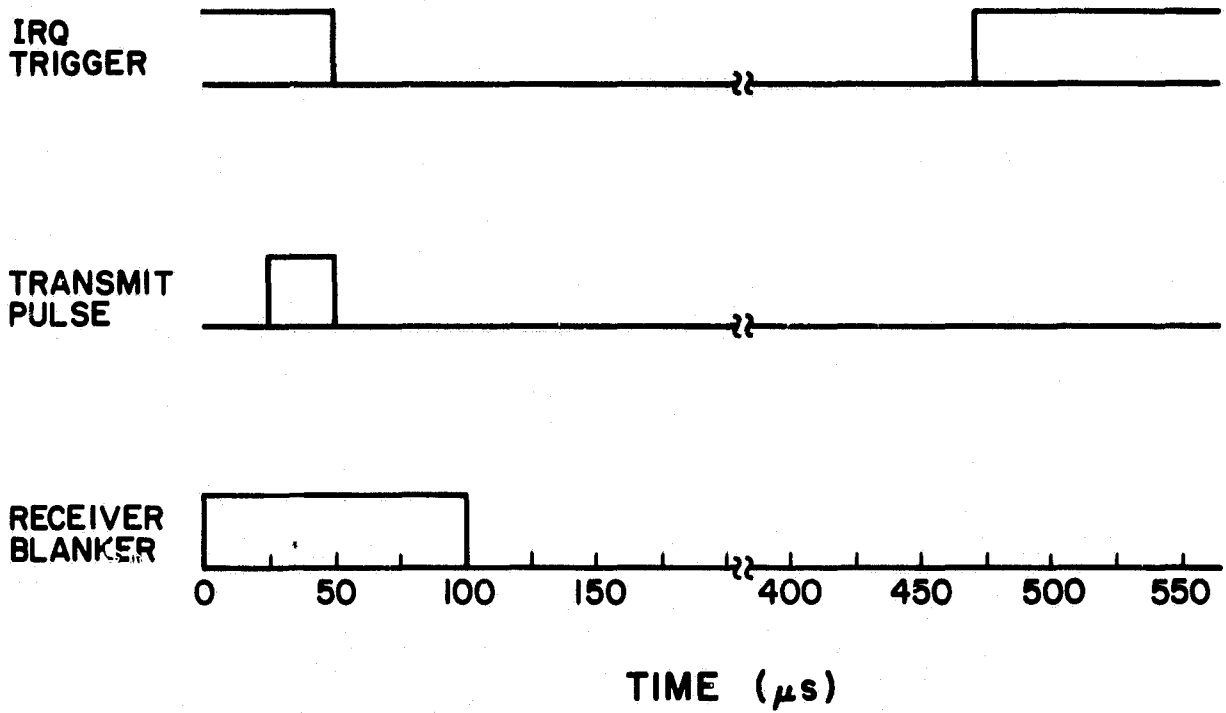


Figure 3.6 Timing diagram for new pulser.

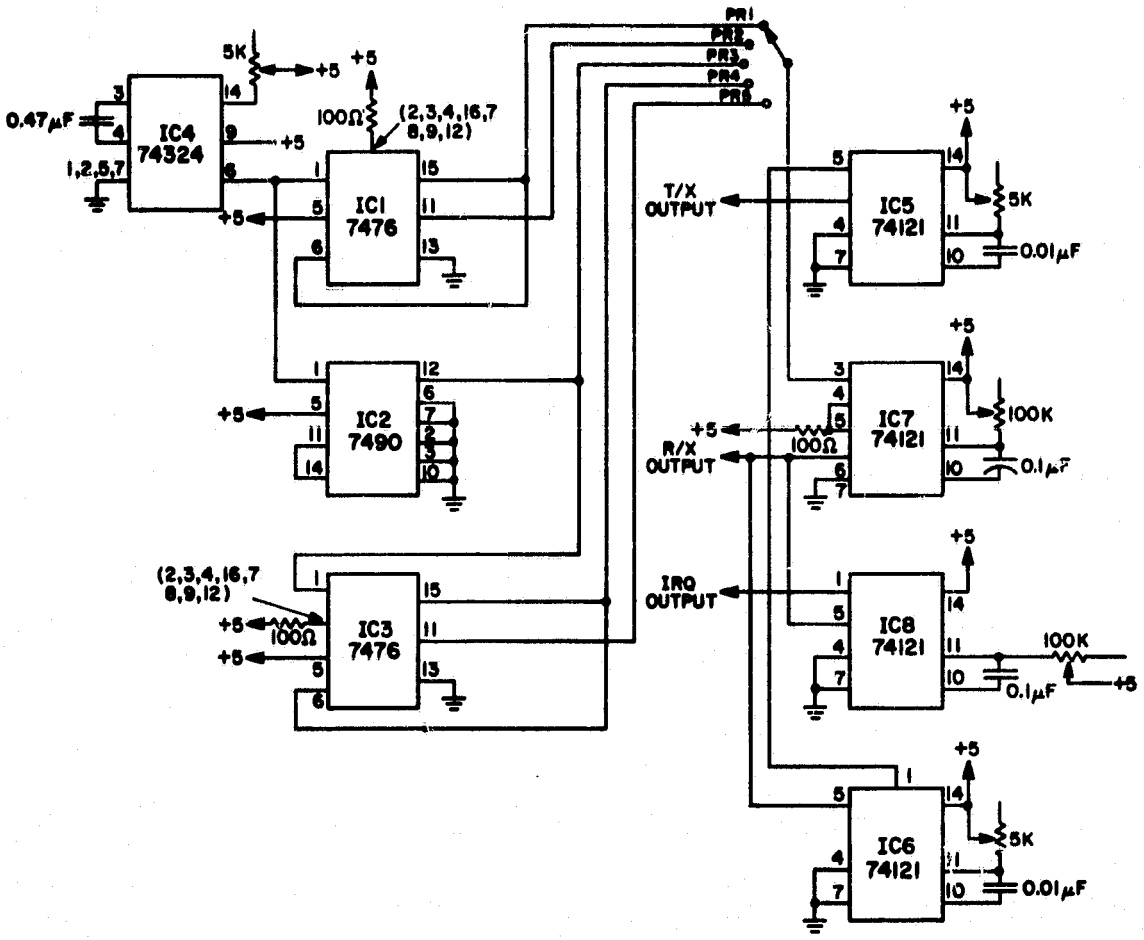


Figure 3.7 Circuit diagram for new pulser.

4. DATA COLLECTION AND ANALYSIS SYSTEM

4.1 OBJECTIVES OF THE ANALYSIS

The major objective of the analysis is to efficiently compute the drift velocity. The analysis can be broken down into two phases. The first phase is the collection of data and the second is the postprocessing of those data.

Since the actual collection of data requires very little time, it is advantageous to use the idle time of the computer to start processing data as they become available. This eliminates two problems in the procedure. The first is the amount of storage required to store the actual amplitudes, and the second is the time required for postprocessing. By computing the correlation functions at the time of the collection, the amount of memory required is reduced by a factor of two. The amount of postprocessing time is dramatically reduced since multiplication is a very time consuming operation.

The first phase now accomplishes more while the second phase is required to do less. This will optimize the procedure of the analysis in the sense that all usable time in the collection phase is used to reduce memory constraints and postprocessing time.

4.2 COLLECTION SOFTWARE

The collection software for the drifts experiment is written primarily in machine code. This is done to optimize the performance of the program so that it will execute at the highest speed possible on the computer. The outer interpreter language used to link the machine code to the computers operating system is FORTH. This higher-level language is extremely flexible and fast.

The collection program performs two major functions. One function is to collect data from the A/D and control the quadrant switching network. The

other is to compute the auto- and cross correlations. During each transmit pulse the radar controller interrupts the processor to collect data. The routine that is executed when this occurs is called the interrupt service routine (ISR). While the processor is waiting for the next interrupt signal from the radar controller it computes the correlation functions. This is the second function of the software.

The flow charts for the ISR are shown in Figures 4.1a, 4.1b, and 4.1c. This section of the program is executed after every transmit pulse. The routine can be in one of three states during execution.

The first state is called the stall state (Figure 4.1c). Because switching between quadrants is done with relays, sufficient time must be given to allow the relays to switch. This segment of the program counts a specified number of interrupts for this delay. A stall counter and a stall flag control the flow of this state. If the flag is set to "1" the program counts the interrupts until the maximum count is reached. For the present version of the collection program the maximum count is ten. This gives the relays 50 millisecond to switch for a PRF of 200 Hz.

The second state is called the wait state (Figure 4.1b). During this time the stall flag is set and the next antenna is switched into the receiver. One of the output ports on the parallel interface controls the quad select. By writing the appropriate number to the J2 port one of the four antennas is sampled; the following list shows the number and the quadrant it selects.

| QUADRANT | VALUE IN HEX TO SELECT |
|----------|------------------------|
| NE | 0E |
| NW | 0D |
| SW | 0B |
| SE | 07 |

After all quadrants are sampled the pointers in the program are updated. The sample ready flag is set for the processing part of the program. The

ORIGINAL PAGE IS
OF POOR QUALITY

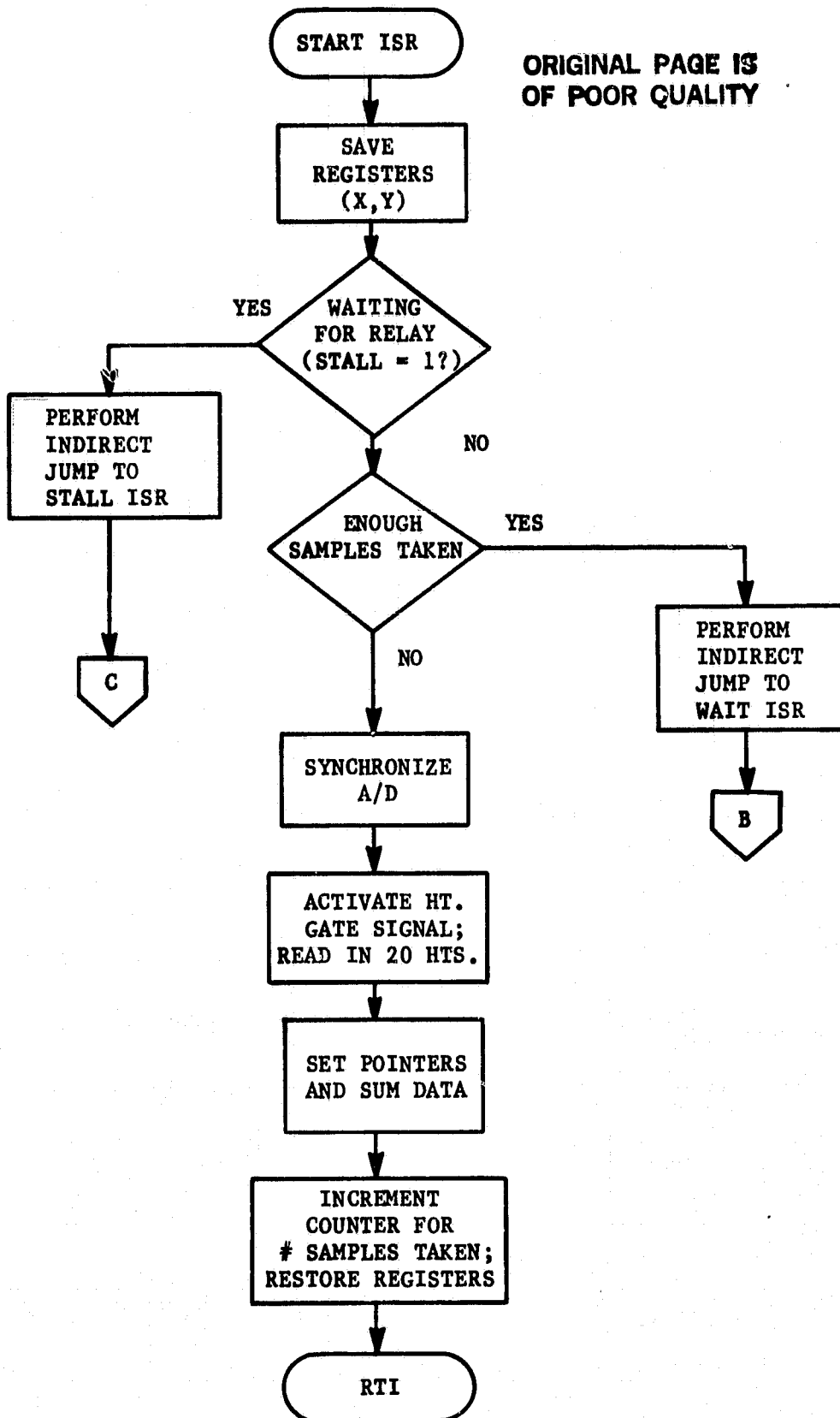


Figure 4.1a Flowchart for interrupt service routine (ISR) collection state.

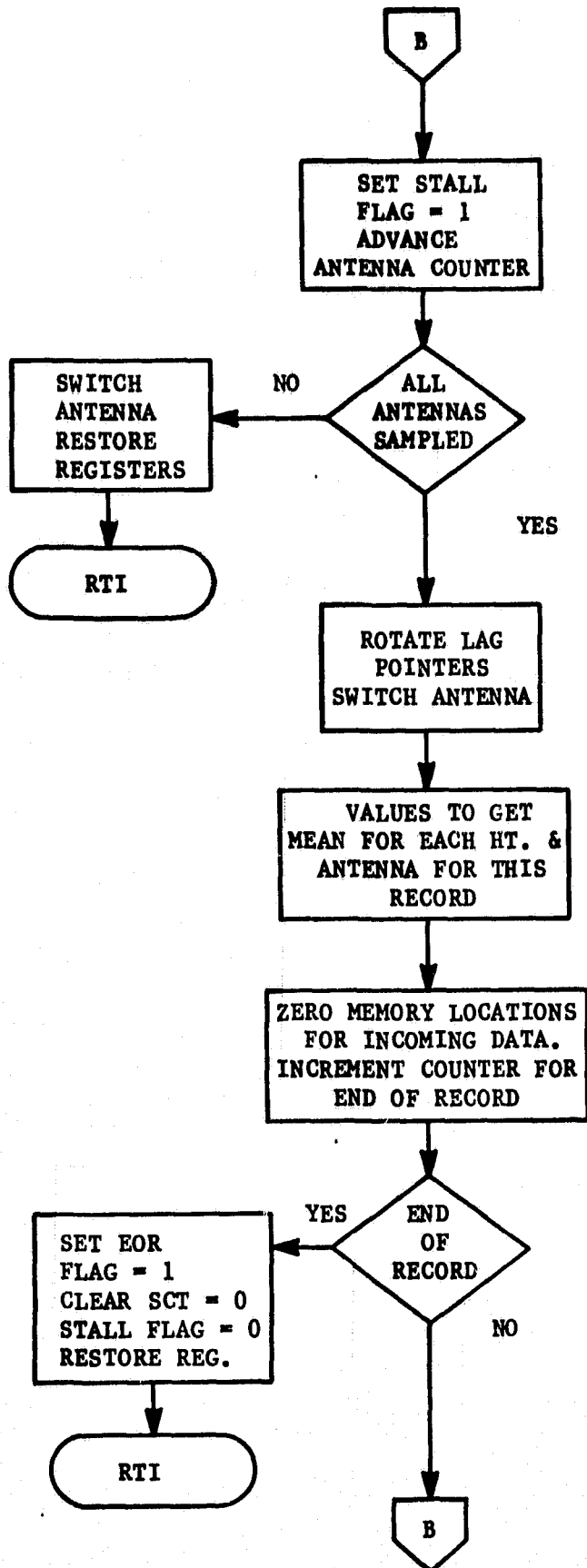


Figure 4.1b Flowchart for (ISR) wait state.

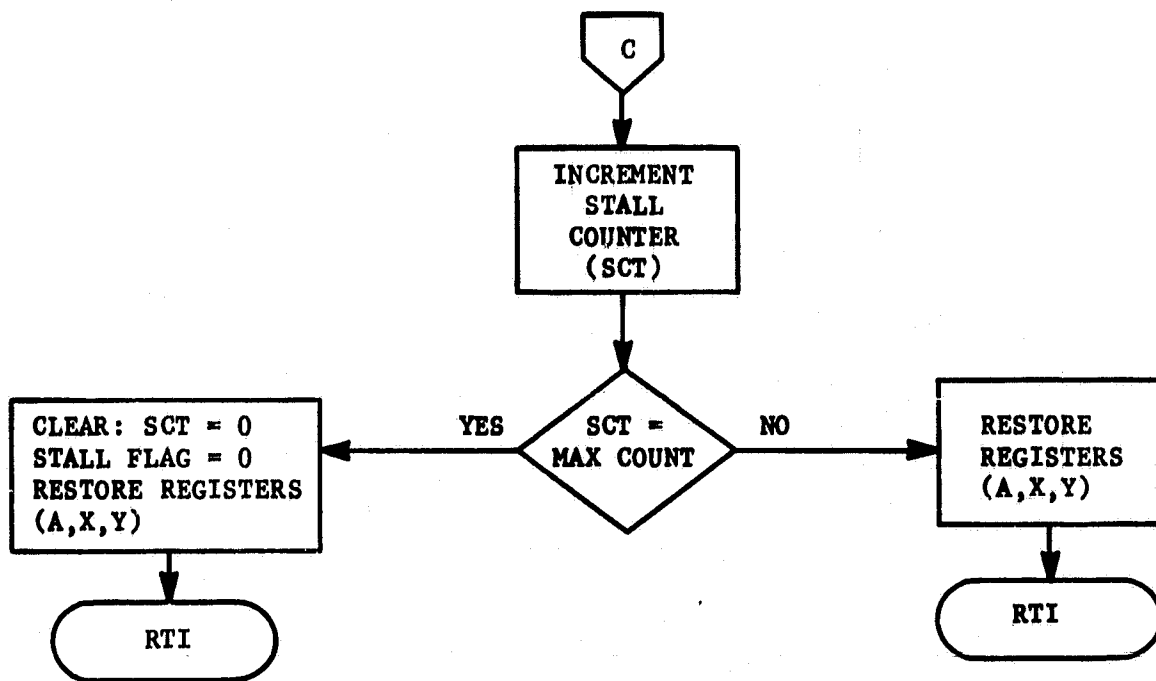


Figure 4.1c Flowchart for (ISR) stall state.

values for the new sample are summed to calculate the mean signal level for each antenna at each altitude. If enough samples have been collected for one minute of data, the end-of-record flag is set, and all other flags are cleared. If not, the program checks to see if the stall count is set for "1" and either increments the stall counter or clears the stall flag.

The last state of the ISR collects the data from the A/D (Figure 4.1a). The first step during this phase is to synchronize the A/D start convert signal with the program. Timer 1 is used on the versatile interface adapter (VIA) for this control signal. It generates a square wave with a frequency equal to $10^6 / (2N + 4)$ where N is the value in the timer 1 latch. The output port will toggle PB7, which is the MSB of the port, each time the timer 1 counts down the value in the latch. By writing to the output port, PB7 is reset and the countdown restarts. Once the A/D is synchronized a dummy read is done to enable latching on the input port.

The program also activates one of the annunciators on the Apple game I/O port by writing and reading a memory location. This gives an indication when the program is reading data. This signal which is generated on pin #15 of the game port is used to set the altitude range gate. The program then reads 20 samples into memory and sums eight of those samples into the input buffer. This summation is performed ten times for each antenna and altitude. This oversampling improves the quantization of the signal and is recommended when the A/D has a small word length (Hagen et al., 1973).

When the computer is not busy with the ISR it is computing the correlation functions. The routine that performs this operation is called DCORR. The flowcharts for this routine are shown in Figure 4.2. The routine DCORR computes the auto- and cross correlations in this manner it is necessary to save the past "L" samples, where "L" corresponds to the number of lags that must be

ORIGINAL PAGE IS
OF POOR QUALITY.

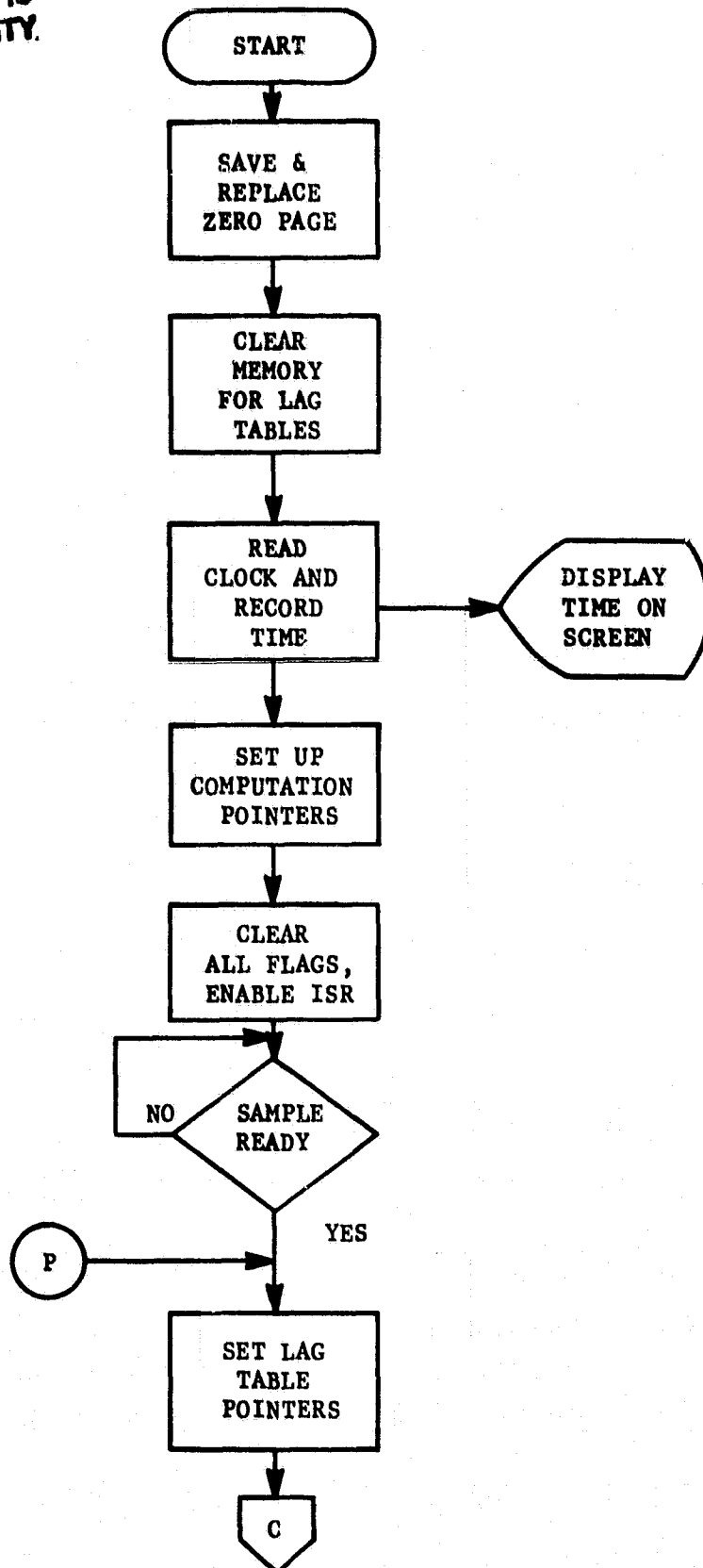


Figure 4.2 Flowchart for main code DCORR.

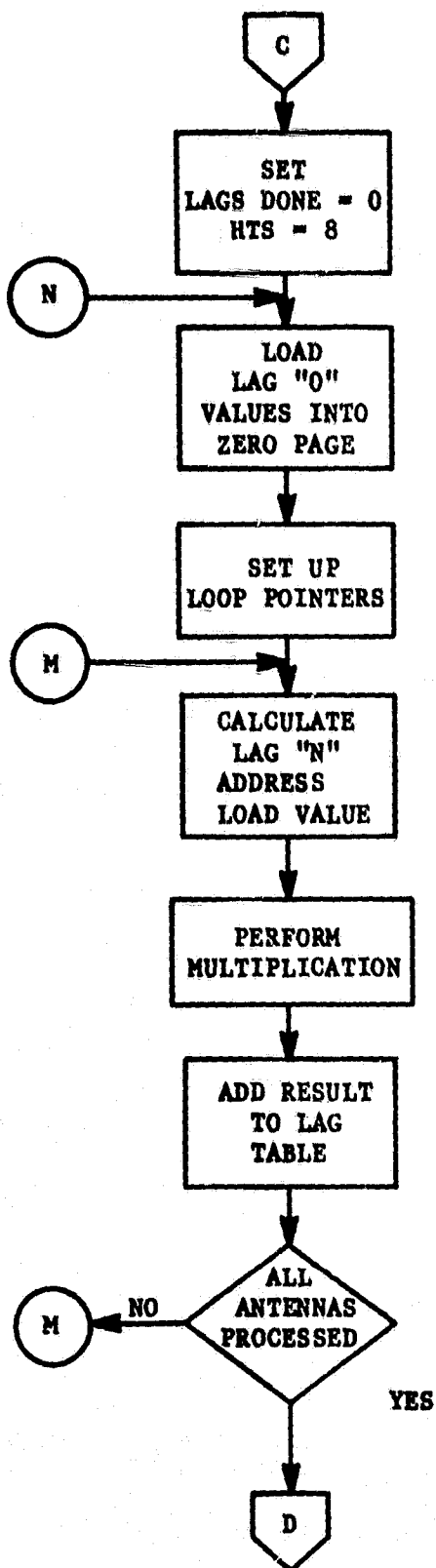
ORIGINAL PAGE IS
OF POOR QUALITY

Figure 4.2 (cont.)

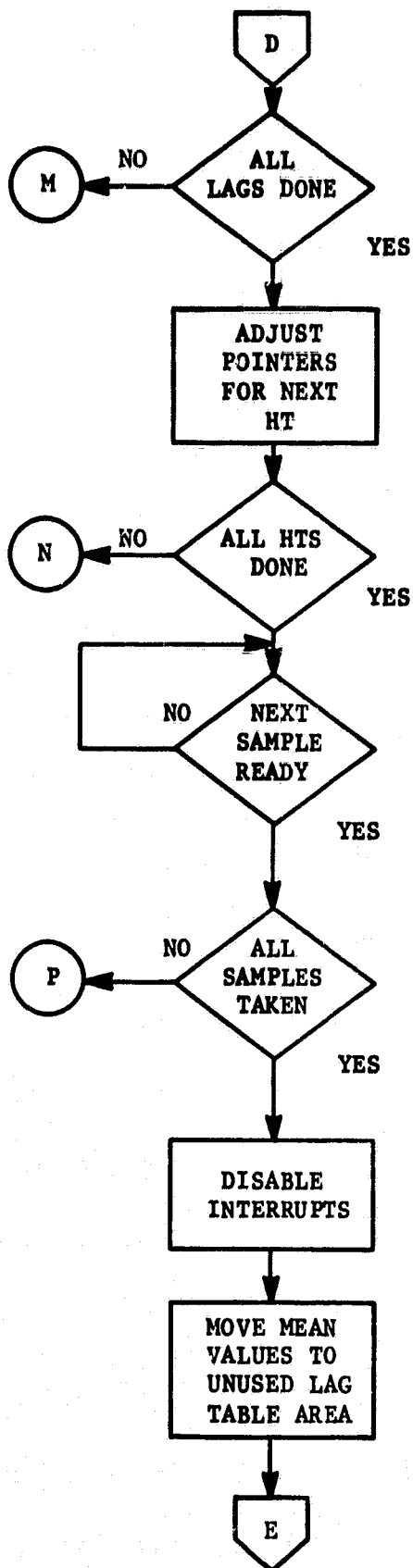
ORIGINAL PAGE IS
OF POOR QUALITY

Figure 4.2 (cont.)

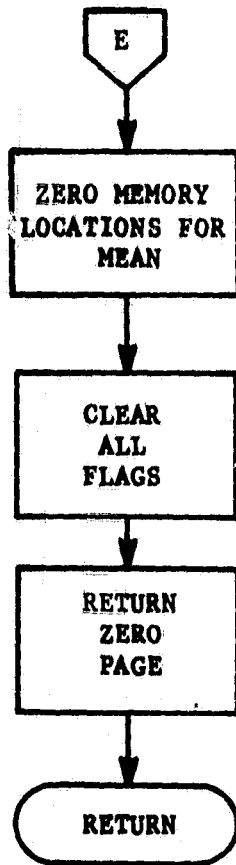


Figure 4.2 (cont.)

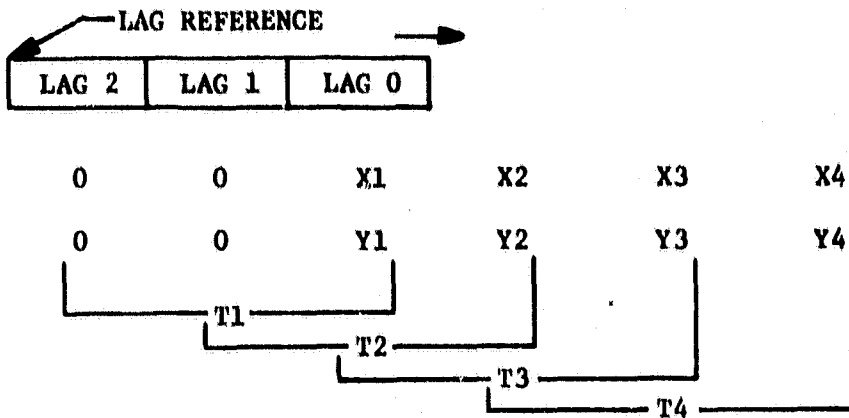
computed. By noting that for a cross correlation $R_{AB}(T)$ where A and B are the two different series, $R_{BA}(T)$ forms the negative lags for $R_{AB}(T)$. For the four-antenna system, four autocorrelations and six cross correlations must be computed. The collection program computes these by computing sixteen correlations with five positive lags and the zero lag. This simplifies the program and reduces the execution time to compute the correlation functions.

After each sample becomes available for processing the program adds the products that have been computed with the past L samples to the lag table. This operation can easily be understood if the products added to the lag table are displayed in the format shown in Figure 4.3. In this example the first two lags plus zero lag are computed. At a time T1 the first sample is available for processing. The products formed at this time interval are added to the corresponding lag bins. At the next time interval the calculation is repeated with the lag reference advanced by one. By repeating this process the values in the lag bins will contain the desired results at the end of the sampling interval. For the drifts-collection program the process is repeated 128 times, this corresponds to a 51-second sampling interval.

4.3 POSTPROCESSING SOFTWARE

The postprocessing software performs the full correlation analysis. The input to this program is the correlation functions computed by the collection software. The output generated includes various parameters of the analysis that are of interest.

The flowcharts for the postprocessing program are shown in Figures 4.4a, 4.4b, 4.4c, and 4.4d. Two different versions of this program have been written. The first version, which follows the flowcharts, calculates the drifts velocity four times with different combinations of three antennas. Because of an unexpected failure in the relay network for the NE quadrant, a



$$\begin{aligned}
 R_{XY}(0) &= \overbrace{X_1 Y_1}^{T1} + \overbrace{X_2 Y_2}^{T2} + \overbrace{X_3 Y_3}^{T3} + \overbrace{X_4 Y_4}^{T4} + \dots \\
 R_{XY}(1) &= 0 + X_2 Y_1 + X_3 Y_2 + X_4 Y_3 + \dots \\
 R_{XY}(2) &= 0 + 0 + X_3 Y_1 + X_4 Y_2 + \dots
 \end{aligned}$$

LAG BINS FOR LAGS 0, +1, +2

Figure 4.3 Method used to calculate correlation functions (refer to text).

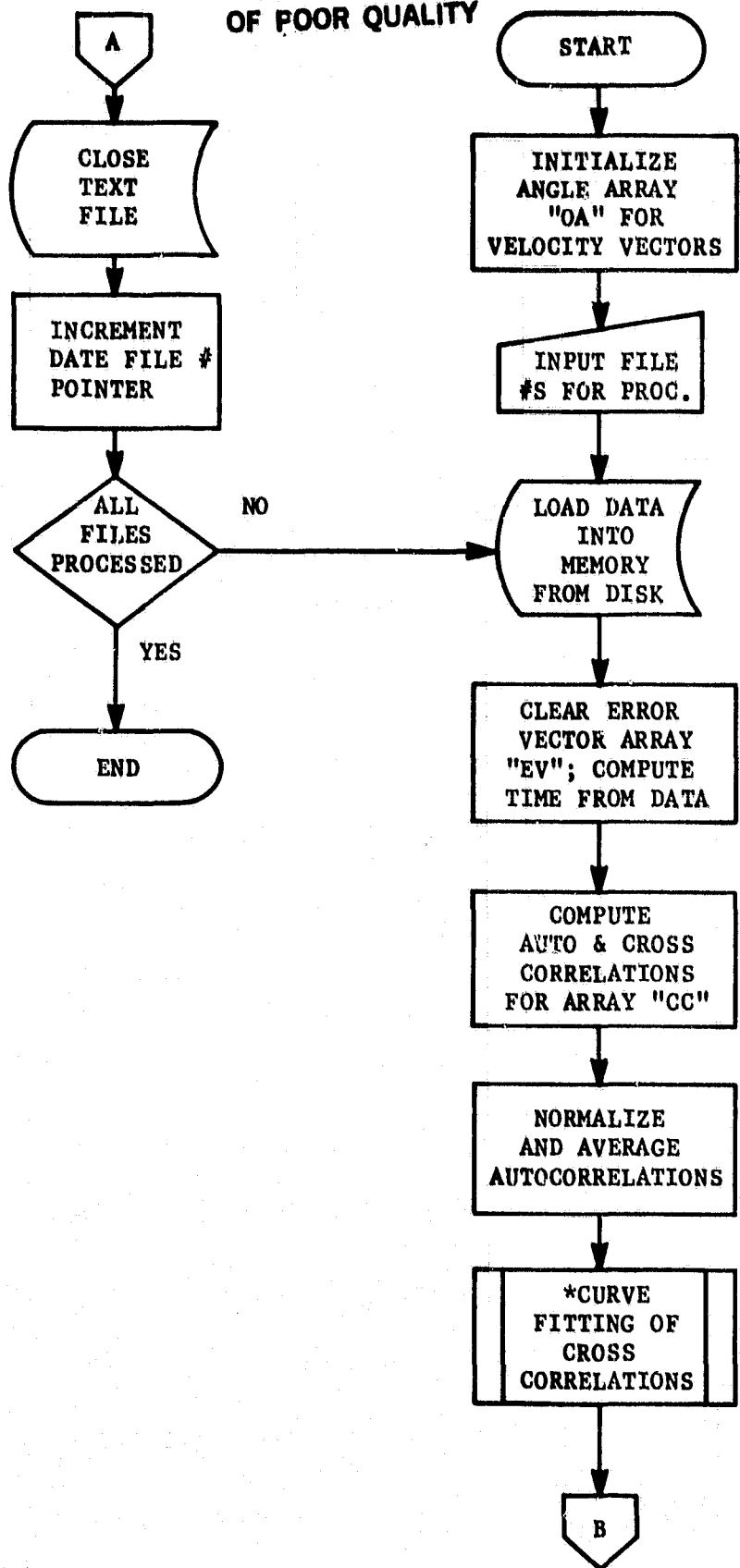


Figure 4.4a Flowchart for 4 antenna postprocessing software.

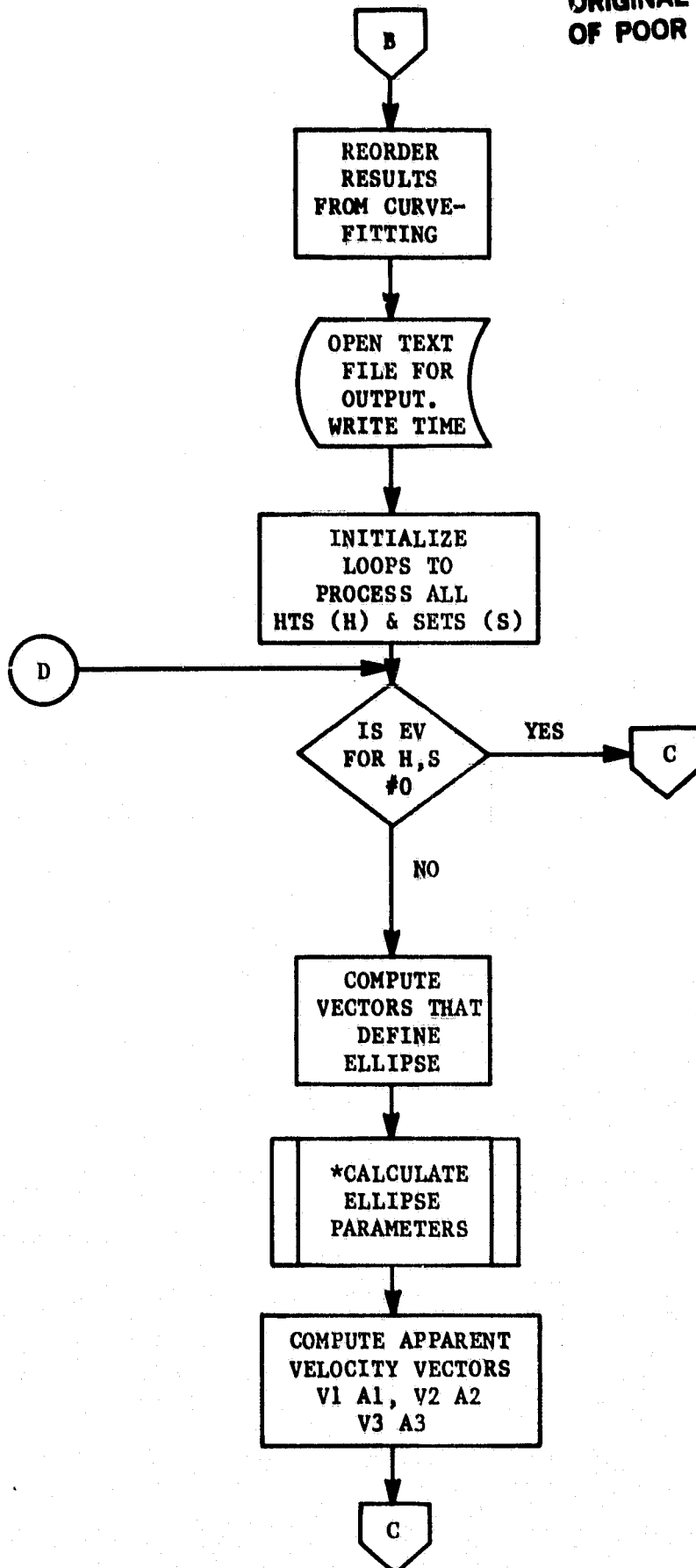


Figure 4.4a (cont.)

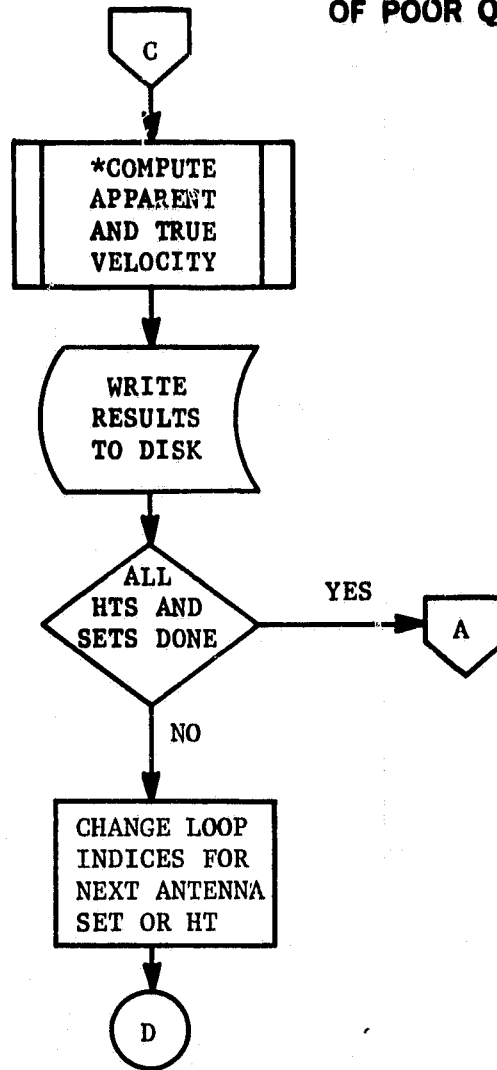
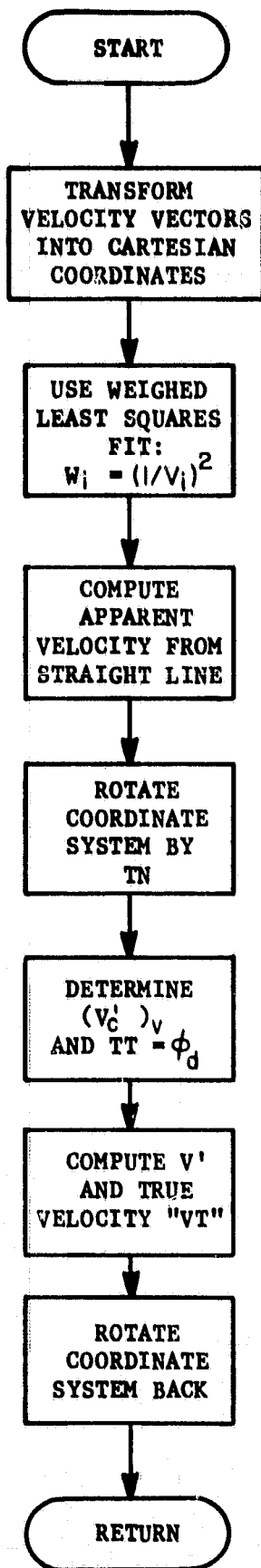


Figure 4.4a (cont.)



ORIGINAL PAGE IS
OF POOR QUALITY

Figure 4.4b Flowchart for computation of apparent and true velocity.

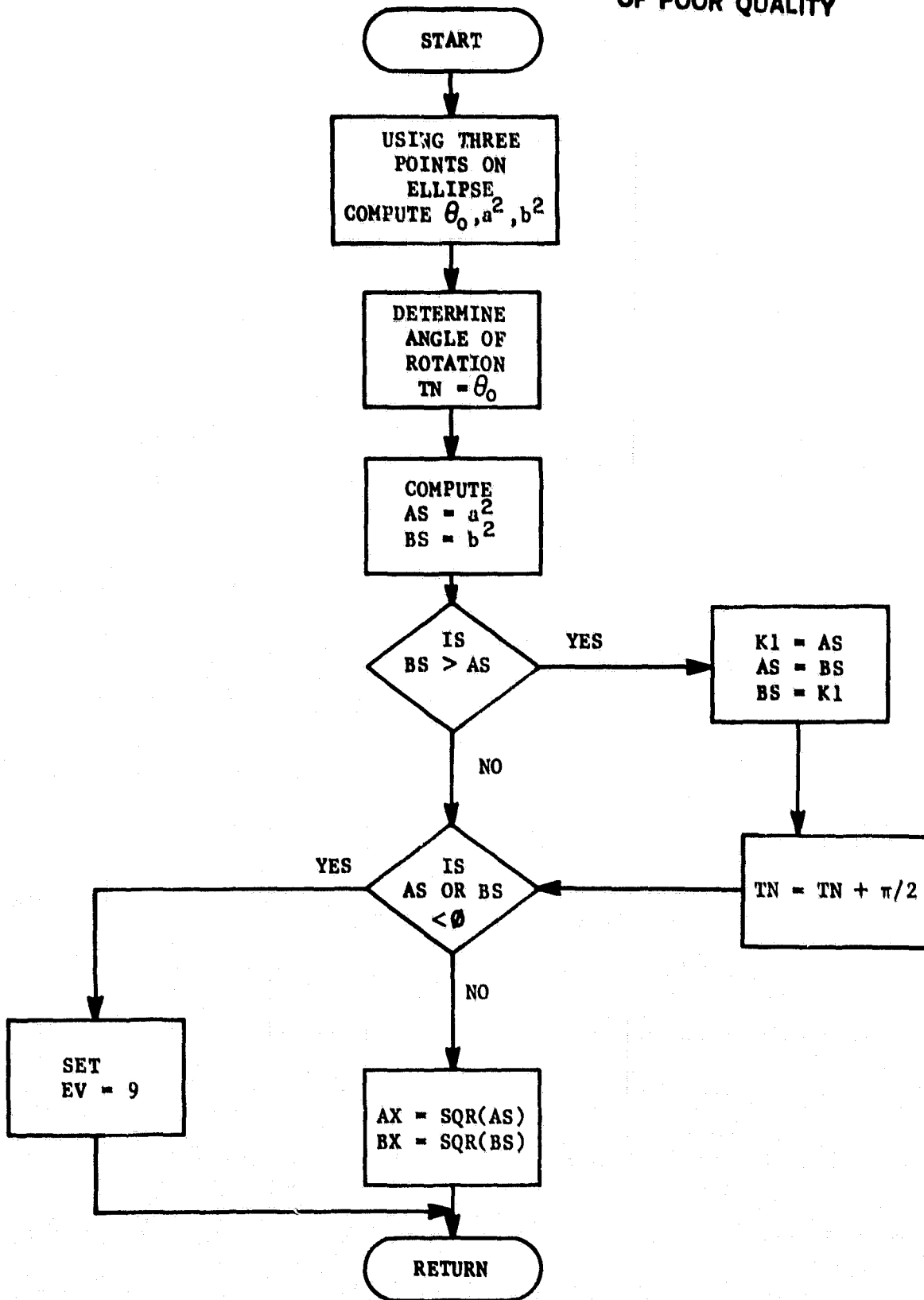


Figure 4.4c Flowchart for computation of ellipse.

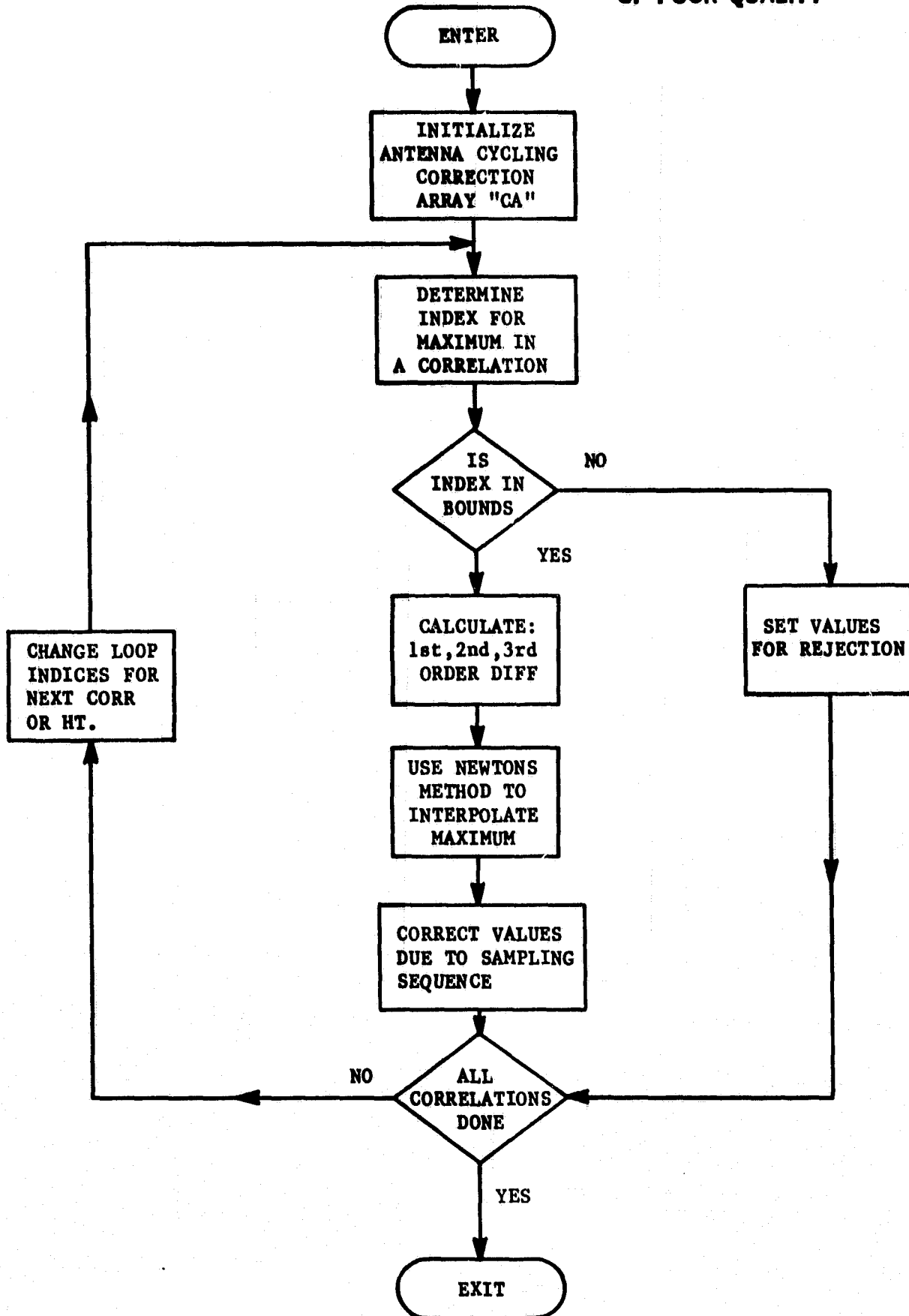


Figure 4.4d Flowchart for curve fitting of cross correlations.

second version of the processing program was developed. The second version is similar to the first except that it is optimized to calculate the drift velocity once with one combination of three antennas.

The major computations for the postprocessing program are:

- 1) Complete the calculation of the auto- and cross correlations.
- 2) Interpolate the cross correlation to determine the true maximum.
- 3) Compute the ellipse and its parameters.
- 4) Compute the apparent and true velocities.

The collection software does not have time to complete the calculation of the correlation functions. It stores all the necessary information so that the postprocessing program can complete the computation. The actual data that the processing program has to work with are the sum of the products and the mean values. The sums of the products, which are more commonly referred to as the autocorrelation and cross correlation, are four-byte integer values. The autocovariance is computed by subtracting the square of the mean and the cross covariance is computed by subtracting the product of the means. After the cross covariances are normalized using the zero lag of the autocovariance the autocovariance is normalized. The use of the terms autocorrelation and cross correlation throughout the rest of the text refer to the normalized autocovariance and the cross covariance.

Once the correlation functions are computed the next calculation is to interpolate the maximum of the cross correlations. For this interpolation a polynomial of the fourth degree is used. Using Stirling's Formula the first, second, third and fourth order differences are computed. If the maximum value is denoted by y_0 the required differences are

$$\Delta y_{-1} = y_0 - y_{-1}$$

$$\Delta y_0 = y_1 - y_0$$

$$\Delta^2 y_{-1} = y_1 - 2y_0 + y_{-1}$$

$$\Delta^3 y_{-2} = y_1 - 3y_0 + 3y_{-1} - y_{-2}$$

$$\Delta^3 y_{-1} = y_2 - 3y_0 - y_{-1}$$

$$\Delta^4 y_{-2} = y_2 - 4y_1 + 6y_0 - 4y_{-1} + y_{-2}$$

The form of the polynomial is:

$$y = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4$$

The values of the "a" constants for this equation are:

$$\begin{aligned} a_0 &= y_0 & a_3 &= 1/12(\Delta^3y_{-2} + \Delta^3y_{-1}) \\ a_1 &= 1/2(\Delta y_{-1} + \Delta y_0) - 1/12(\Delta^3y_{-2} + \Delta^3y_{-1}) & a_4 &= 1/24\Delta^4y_{-2} \\ a_2 &= 1/2(\Delta^2y_{-1}) - 1/24\Delta^4y_{-2} \end{aligned}$$

To determine where the maximum occurs it is necessary to set the derivative equal to zero.

$$\frac{dy}{dx} = a_1x + 2a_2x^2 + 3a_3x^3 + 4a_4x^4 = 0$$

The roots to this equation can be found by applying Newton's method. This requires the next derivative:

$$\frac{d^2y}{dx^2} = a_1 + 2a_2x + 6a_3x^2 + 12a_4x^3$$

The first approximation to the root is zero since the equation has been formed with respect to the maximum value y_0 located at $x = 0$. By iteration the interpolated value of x is found by using

$$x_N = x_{N-1} - \frac{dy}{dx} / \frac{d^2y}{dx^2}$$

"N" times until the root converges to a specified tolerance. The value of N for the processing program is 100. Most of the time only a few iterations are necessary for convergence.

Once the value of x is determined the maximum value is computed by evaluating the polynomial. The time delay in seconds is computed from

$$TD = [(KM - 6) + x] * .4 + CA$$

The value of KM is the index where the maximum occurred. For the zero lag the index KM would have the value of six. For positive lags take on the values 7-11, for negative lags the values 5-1. The values of "CA" is an antenna cycling correction value since all antennas are not sampled at the same time.

The approach taken to interpolating the maximum is a great improvement over the former processing program. That program was an adaption of the analysis program used at the University of Adelaide (Australia) for drifts analysis (Welland in Edwards 1978). The former program did not have an iteration limit for the determination of the maximum. It also used a different method for solving the maximum that wasted time, and at times lacked accuracy.

The second version of the processing program also did a parabolic interpolation if the iteration limit was exceeded or if the maximum of the cross correlation was outside the range of plus or minus the third lag. Because the fourth order fit requires two values on either side of the maximum, and only the ± 5 lags are computed, the parabolic interpolation allows a maximum to be computed if it occurs on the ± 4 th lag. The fourth-order fit may not converge rapidly if the peak is broad. In this case a parabolic fit is the best approximation that can be made. The calculated value of x for this fit is

$$x = 1/2 \left(\frac{y_{-1} - y_{+1}}{y_{-1} - 2y_0 + y_{+1}} \right)$$

and the maximum correlation value is

$$y_{\max} = y_0 - \frac{(y_{-1} - y_{+1})}{2} x + \frac{(y_{-1} - 2y_0 + y_{+1})}{2} x^2$$

The next step in the calculation is the determination of the ellipse. From the analysis the ellipse is defined by the endpoints of the vectors; $(V_c')_{12}$, $(V_c')_{13}$, $(V_c')_{23}$. The general form of an ellipse in polar coordinates that has been rotated through some angle θ_0 is

$$\frac{\cos^2(\theta - \theta_0)}{a^2} + \frac{\sin^2(\theta - \theta_0)}{b^2} = \frac{1}{r^2}$$

The three endpoints with polar coordinates (r_1, θ_1) , (r_2, θ_2) , (r_3, θ_3) are

sufficient to determine the characteristics of the ellipse (Fooks 1965).

By defining the two variables:

$$D_{12} = \frac{1}{r_1^2} - \frac{1}{r_2^2}$$

$$D_{13} = \frac{1}{r_1^2} - \frac{1}{r_3^2}$$

the value of θ_0 can be determined from the formula

$$\tan 2\theta_0 = \frac{D_{12}(\cos 2\theta_1 - \cos 2\theta_3) - D_{13}(\cos 2\theta_1 - \cos 2\theta_2)}{D_{13}(\sin 2\theta_1 - \sin 2\theta_2) - D_{12}(\sin 2\theta_1 - \sin 2\theta_3)}$$

The values of a^2 and b^2 are:

$$a^2 = \frac{(K_3 K_2 - K_1 K_4)}{(K_2/r_2^2 - K_4/r_1^2)}, \quad b^2 = \frac{(K_3 K_2 - K_1 K_4)}{(K_3/r_1^2 - K_1/r_2^2)}$$

where

$$K_1 = \frac{1 + \cos 2(\theta_1 - \theta_0)}{2}, \quad K_2 = \frac{1 - \cos 2(\theta_1 - \theta_0)}{2}$$

$$K_3 = \frac{1 + \cos 2(\theta_2 - \theta_0)}{2}, \quad K_4 = \frac{1 - \cos 2(\theta_2 - \theta_0)}{2}$$

Because the correlation contours can have any shape whatsoever (Briggs 1950), the values of a^2 and b^2 may be negative. When this occurs the analysis is aborted.

The last computation will calculate the apparent and true velocities. To calculate the apparent velocity the vectors, V_{12}' , V_{13}' , V_{23}' , are computed. The magnitudes are

$$v'_{ij} = \frac{d_{ij}}{t_{ij}}$$

where d_{ij} is the baseline distance between the antennas and t_{ij} is the time displacement from the cross correlation. The angle for each vector is the angle measured clockwise from north from "i" to "j". The endpoints of these vectors define the V' -line. To compute the best fit, the method of weighted

least squares is used. The value of the W_1 's is $(r_1)^{-2}$. By transforming the polar coordinates of these endpoints to cartesian coordinates (X,Y) the Y-intercept (YI) and slope of the line (M) are given by

$$YI = \frac{\sum W Y \sum W X^2 - \sum W X \sum W X Y}{\sum W \sum W X^2 - (\sum W X)^2}$$

$$M = \frac{\sum W \sum W X Y - \sum W Y \sum W X}{\sum W \sum W X^2 - (\sum W X)^2}$$

Once the slope and intercept are calculated the apparent velocity can be computed. The apparent velocity is defined by the distance and direction of the line that joins the origin to the point (X,Y) perpendicular to the V'-line.

The values of X and Y are:

$$X = \frac{-m}{(m + \frac{1}{m})}$$

$$Y = \frac{a}{(m^2 + 1)}$$

and the apparent velocity in polar coordinates is

$$V_a = \sqrt{x^2 + y^2}$$

$$\phi_a = \tan^{-1}(y/x)$$

The next step is to calculate the point of tangency between the ellipse and the V'-line. To calculate this point it is convenient to rotate the coordinate system so that cartesian coordinates can be used. The slope at any point (X_0, Y_0) on the ellipse becomes:

$$m = \frac{-b^2 X_0}{a^2 Y_0}$$

The slope m that is equated to this is

$$m' = \frac{-1}{\tan(\phi_a - \theta_0)} = -\frac{b^2}{a^2} \left(\frac{X_0}{Y_0}\right)$$

The possible solutions to the tangency point are

$$Y_0 = \pm \frac{b^2}{(b^2 + a^2(m')^2)^{1/2}}$$

$$X_0 = \pm \frac{a^2 m'}{(b^2 + a^2(m')^2)^{1/2}}$$

The correct values of x_0 and y_0 are chosen based on the fact that the true velocity must be within 90 degrees of the apparent velocity (Fooks and Jones 1961). The vector defined to the point (x_0, y_0) is known as the $(V_c')_v$ vector. The vector that lies in the same direction as the $(V_c')_v$ vector but extends to the V' -line is referred to as the V' vector. To calculate this vector the y -intercept calculated for the V' -line must be rotated to the new coordinated system. The transformation is

$$YI' = V_a [\sin(\phi_a - \theta_0) - m' \cos(\phi_a - \theta_0)]$$

If the slope of the V' vector is m_t . The endpoint coordinates of the V' vector are:

$$x = \frac{YI'}{m_t - m'} \qquad y = \frac{m_t YI'}{m_t - m'}$$

The magnitude of the true velocity is then

$$|V_{\text{drift}}| = \frac{(V_c')_v^2}{V'} = \frac{(X_0^2 + Y_0^2)}{\sqrt{X^2 + Y^2}}$$

The direction of drift, θ_t , must be modified by θ_0 because the value calculated was in the rotated coordinate system. Therefore $\theta_t = \theta'_t + \theta_0$.

5. COMPARISON OF PARTIAL-REFLECTION AND COHERENT-SCATTER WIND DATA

The new partial-reflection drifts experiment was used to make simultaneous measurements with coherent scatter on April 24 and 25 of 1982. These data are unique because during the time of the observations the upper atmosphere was exposed to energetic electrons from a solar flare. According to Briggs (1981), these two techniques used to determine atmospheric winds are basically the same, therefore the mean horizontal winds for the two days were compared between the experiments to determine if similarities existed.

The horizontal-wind velocity computed by coherent scatter is a projection of the N-S and E-W components onto the line-of-sight direction for their antenna system. The antenna system is pointed in the south-east direction at an azimuth angle of 126 deg from the north, 1.5 deg off-vertical. The wind velocity components calculated by the partial-reflection-drifts experiment are projected onto the same azimuth for comparisons.

Because of the small angle from vertical that is used to determine the horizontal velocity for coherent scatter, hourly means are used for the comparison. The altitudes that overlap between the experiments are 70.5, 75, 84, 88.5, 93, and 97.5 kilometers.

The comparison of the results shows a positive correlation for five of the six altitudes and are shown in Figures 5.1 - 5.5. The points on the scatter plot should form a straight line with the slope equal to unity. The comparison at 79.5 km yields a correlation coefficient of .54 and a confidence level of 96.3%. The results at the next altitude had a correlation coefficient of .357 with a confidence level of 86.7%. These altitudes indicate a definite similarity between the two experiments. The comparison at 75 and 93 kilometers had a weak correlation. The correlation at 97.5 km was .524 with a confidence level of 95.8%. The slope of the scatter plot

ORIGINAL PAGE IS
OF POOR QUALITY

DATE: 4/24/82
ALT = 75 km
PHO = 0.079
CONF LEVEL = 59.79%

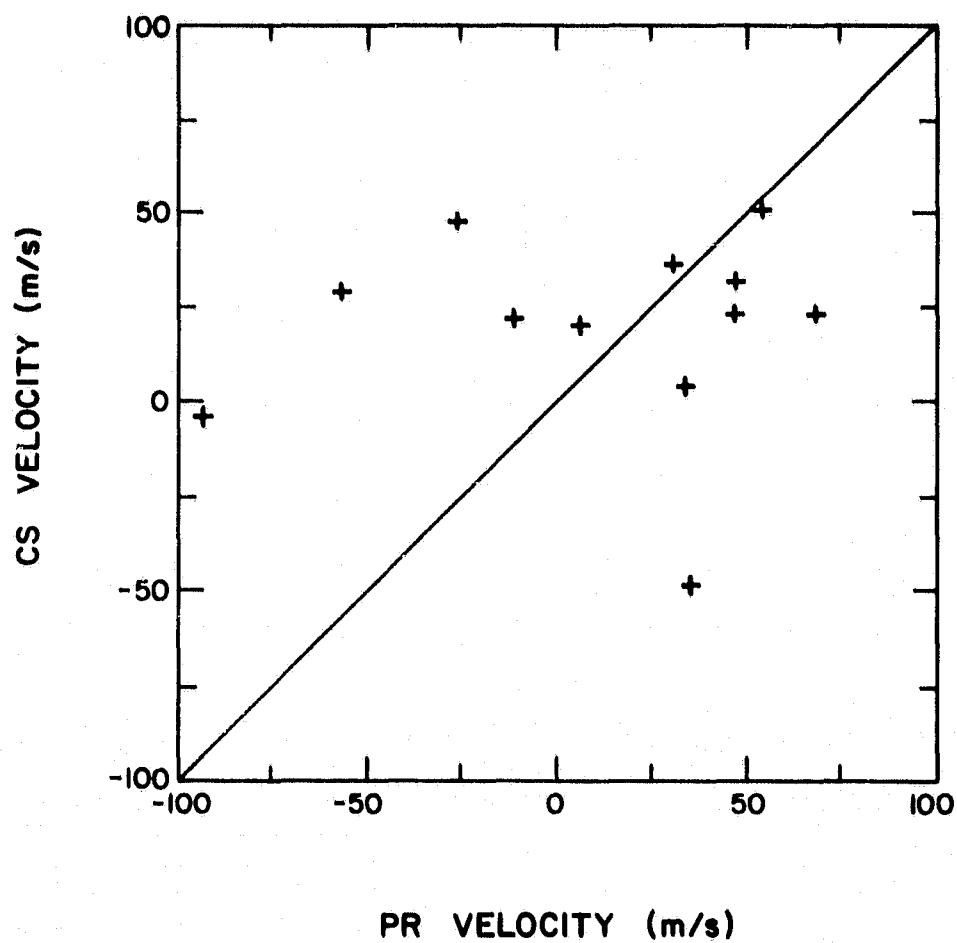


Figure 5.1 Scatter plot at 75 km.

ORIGINAL PAGE IS
OF POOR QUALITY

DATE: 4/24/82
ALT = 79.5 km
PHO = 0.540
CONF LEVEL = 96.3%

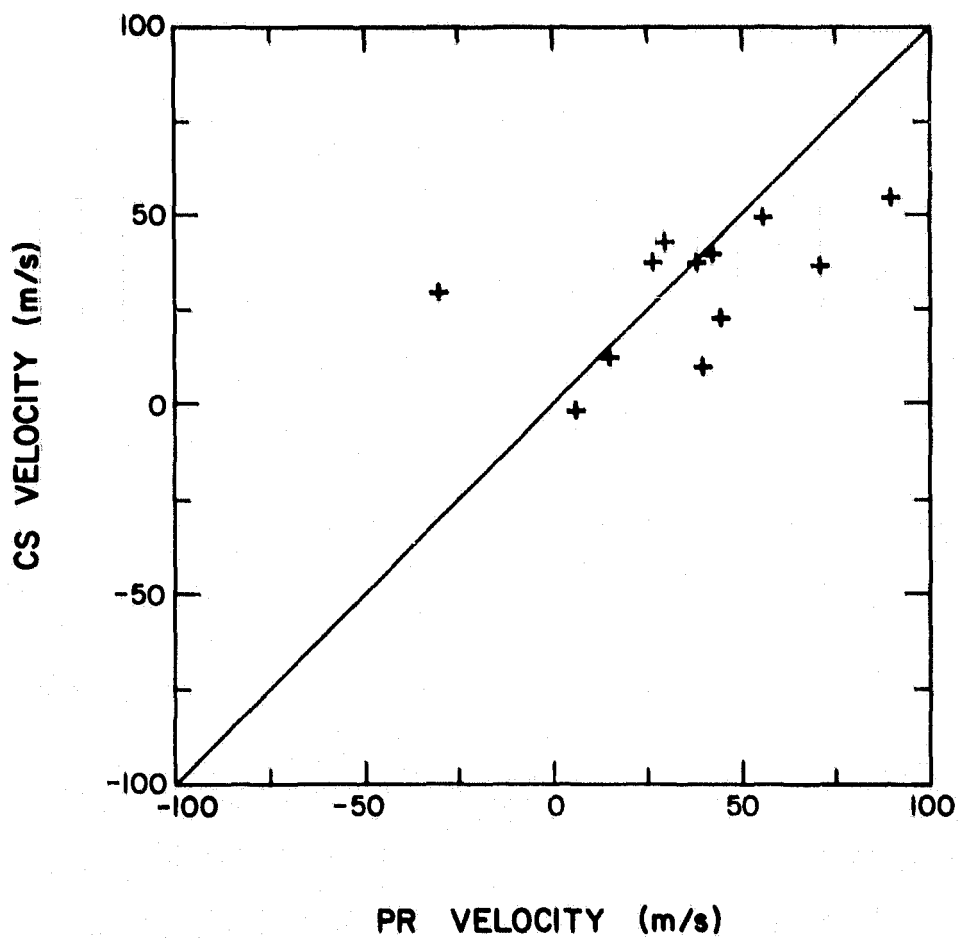


Figure 5.2 Scatter plot at 79.5 km.

DATE: 4/24/82
ALT=84.0 km
PHO=0.357
CONF LEVEL=86.7%

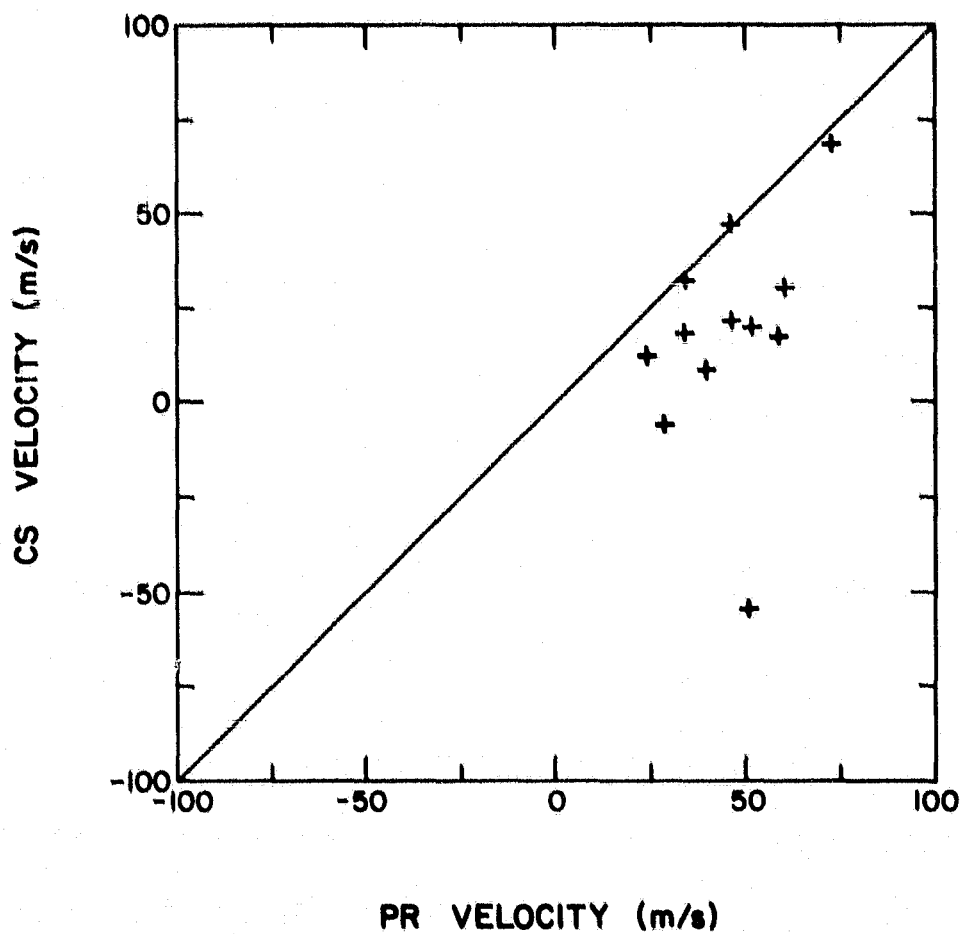


Figure 5.3 Scatter plot at 84 km.

ORIGINAL PAGE IS
OF POOR QUALITY

DATE: 4/24/82
ALT=93.0 km
PHO=0.135
CONF LEVEL = 66.03%

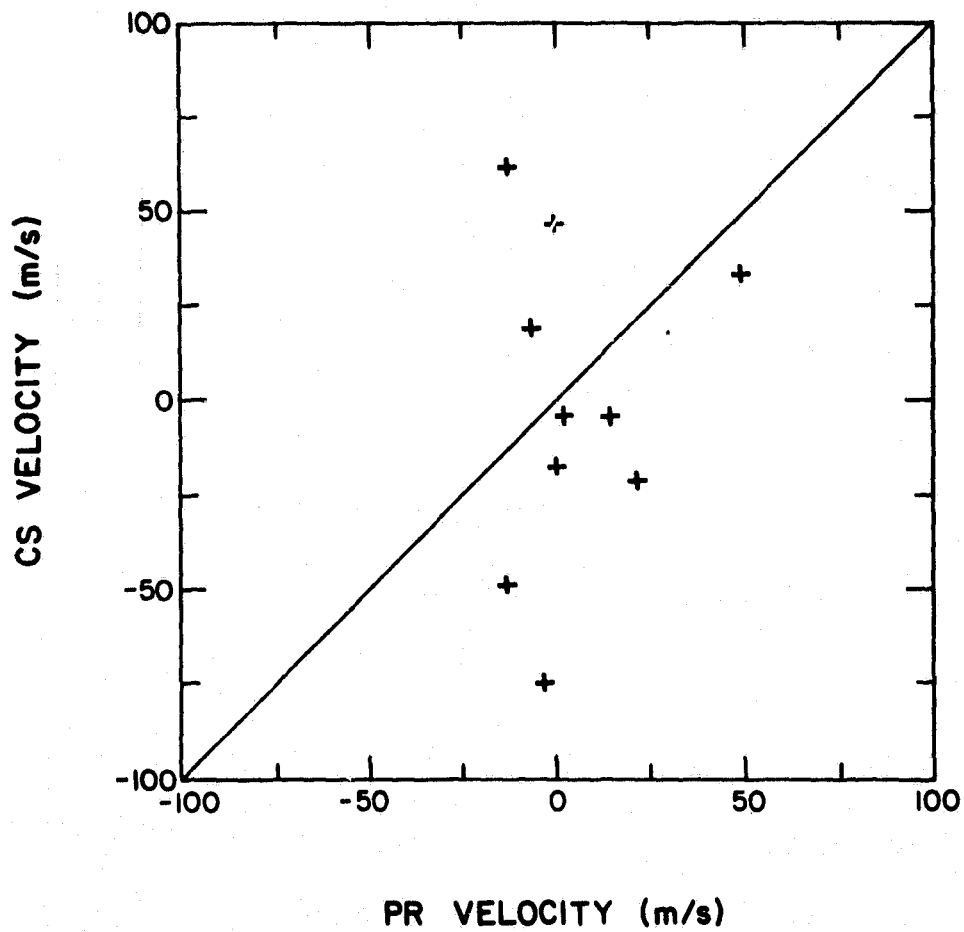


Figure 5.4 Scatter plot at 93 km.

ORIGINAL PAGE IS
OF POOR QUALITY

DATE: 4/24/82
ALT=97.5 km
PHO=0.524
CONF LEVEL=95.8%

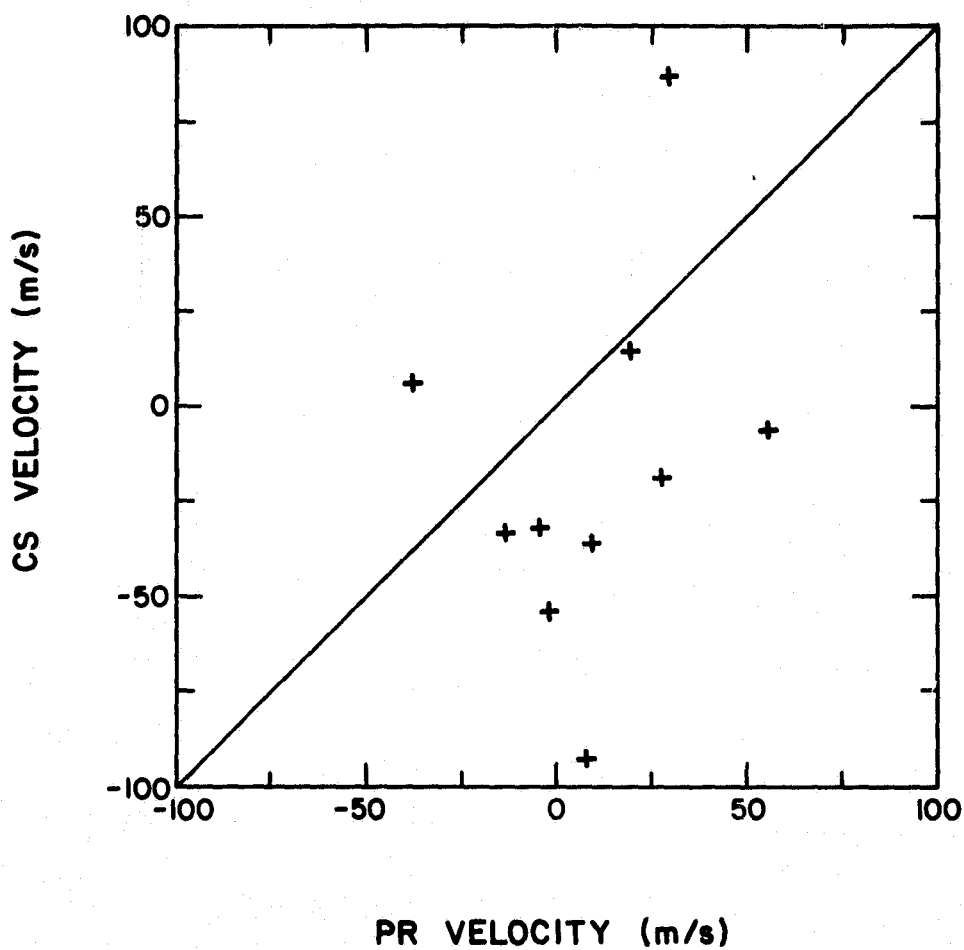


Figure 5.5 Scatter plot at 97.5 km.

for this altitude does not match the expected slope for a one-to-one agreement.

For some altitudes the agreement between coherent scatter and partial reflection is quite good. One observation about the altitudes that shows the greatest agreement is that they also correspond to regions where the scattered power is large, from 75 to 88.5 km altitude. The smaller correlations at 84 km and 75 km are somewhat unexpected if large scattered power improves the comparison.

The computer simulation of the drifts experiments done by Pitteway et al. (1971) shows an interesting result. In the simulation the calculated drift velocity is plotted versus noise factor (Figure 5.6). The noise factor is a velocity perturbation, introduced to account for random changes in the ionosphere. If the calculated drift velocity tends to be larger than the actual drift, with an increasing noise factor, the scatter plots at 84 km and 75 km may indicate this. By reducing the magnitude of the velocity calculated by the drifts experiment, better correlations result. The disagreement that occurs at 97.5 km may be due to receiver saturation. At this altitude the receiver saturates a small percentage of the time. During the collection of these data the objective of keeping the gain set for a range that is acceptable for all altitudes is hard to accomplish because of the large variations in the signal strength. The preliminary results from this simultaneous experiment indicate the need for future comparisons.

ORIGINAL PAGE IS
OF POOR QUALITY

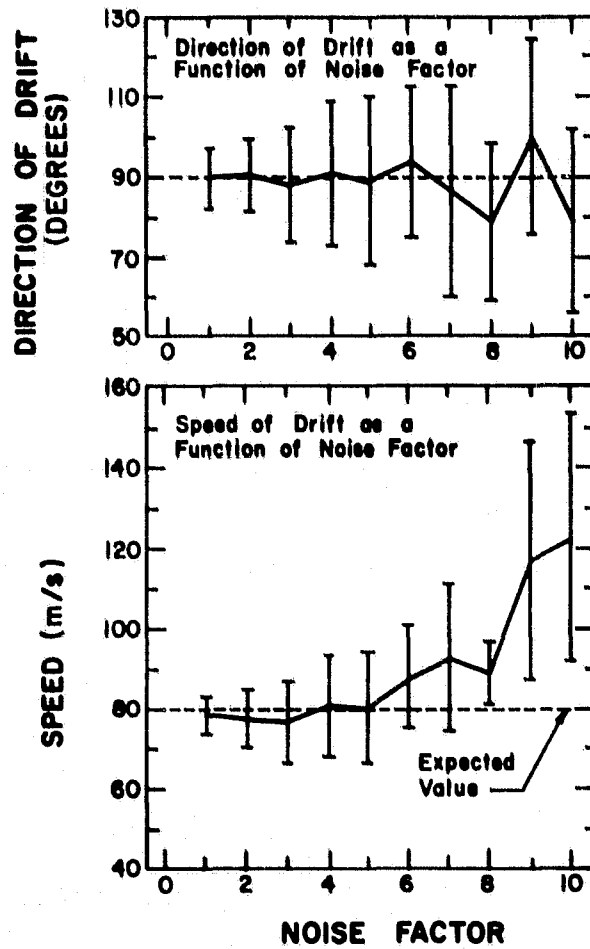


Figure 5.6 Drift velocity as a function of velocity perturbation factor.

6. CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

The work has dealt with the enhancements of the partial-reflection drifts experiment and the preliminary results of simultaneous measurements.

The modifications to the system:

- 1) Allow simultaneous measurements with coherent scatter;
- 2) Enable the drifts experiment to operate continuously so that large amounts of data could be collected;
- 3) Decrease the time required for postprocessing of data which has a direct relationship to the amount of data that can be collected daily; and
- 4) Permit easy access to processed results by storing them on floppy disk.

The results of the preliminary simultaneous measurements have indicated an agreement between the drifts method and coherent-scatter's method for determining horizontal wind velocity. This set of results indicates the need to continue simultaneous measurements under careful control of the receiver gain, until enough data have been collected to assure statistical significance.

Future improvements to the system could solve the remaining problems.

- 1) Signal-to-Noise-Ratio. By adding phase detectors so that coherent detection can be used the SNR would improve since the signal then can be integrated.
- 2) Transmitter. If the transmitter power could be increased the lower limit of the altitude range could be reduced.
- 3) Quadrant-Switching Network. This relay network should be replaced with electronic switching. The present design is not reliable for extended use.

4) Computer. If coherent detection is used, greater computing power is necessary to handle the complex correlation functions.

5) Receiver Gain. The large variance in signal amplitude dictates that a programmable attenuator is necessary to attenuate signals at selected altitude range gates.

APPENDIX I
COLLECTION PROGRAM.

```
1 FORGET DRIFT
2 (
3 *****
4 * COLLECTION SOFTWARE FOR PARTIAL *
5 * REFLECTION DRIFTS EXPERIMENT *
6 *
7 * COMPUTES AUTO AND CROSS CORRELATIONS *
8 * FOR THE FOUR ANTENNA SYSTEM USING A *
9 * HARDWARE MULTIPLIER *
10 *
11 * LIMITS: *
12 * 1. EIGHT ALTITUDES MAXIMUM *
13 * 2. SIX LAGS INCLUDING ZERO LAG *
14 * 3. PLUS AND MINUS 5 LAGS FOR CROSS *
15 * CORRELATIONS *
16 * 4. MAXIMUM PRF WITH ABOVE LIMITS IS *
17 * 200 HZ *
18 *
19 *
20 * WRITTEN BY *
21 * RAYMOND L. RUGGERIO *
22 * 1982 *
23 *****
24 )
25
26 ; DRIFT ;
27 0001 VARIABLE MINUTES
28 0000 VARIABLE WAIT
29 0000 VARIABLE LW
30 0000 VARIABLE HGATE
31 0000 VARIABLE PADDR2
32 0000 VARIABLE PADDR
33 ; INITI/O
34 HEX
35 FF C782 CI ( SET DDRB FOR OUTPUT )
36 C0 C78B CI ( SET ACR )
37 00 C78E CI ( DISABLE INTERRUPTS ON VIA )
38 03 C784 CI ( TIMER1 SET FOR 100KHZ )
39 00 C785 CI
40 04 C70C CI ( PCR SET FOR POS EDGE INTERRUPTS )
41 03 C70B CI ( ACR SET TO LATCH PORTS )
42 00 C703 CI ( DDRA SET FOR INPUT )
43 FF C702 CI ( DDRB SET FOR OUTPUT )
44 00 C70E CI ( CLEAR IER )
45 ;
46 CODE DCORR ( MAIN CODE FOR COLLECTION )
47 FF # LDX
48 BEGIN, ( SAVE ZERO PAGE AND REPLACE )
49 00 # LDA, ( WITH ZERO PAGE NEEDED FOR PROGRAM )
50 3F00 #X STA,
51 3E00 #X LDA,
52 00 #X STA,
53 # DEX,
54 ENDNE,
55 0E # LDX, ( CLEAR MEMORY FOR LAG TABLE COMPUTATIONS )
56 00 # LDY,
57 40 # LDA,
58 E1 # STA,
59 00 # LDA,
60 E0 # STA,
61 BEGIN,
62 BEGIN,
63 E0 #Y STA,
64 # INY,
65 ENDNE,
66 E1 # INC,
67 # DEX,
68 ENDNE,
69 04 # LDX, ( CLEAR ADDITIONAL MEMORY FOR LSB OF LAG TABLE )
70 00 # LDY,
71 70 # LDA,
72 E0 # STY,
73 E1 # STA,
74 00 # LDA,
75 BEGIN,
76 BEGIN,
77 E0 #Y STA,
78 # INY,
79 ENDNE,
80 E1 # INC,
81 # DEX,
82 ENDNE,
83 39 # LDA, ( READ CLOCK AND WAIT FOR MINUTE TO FLIP )
84 # PHA,
85 1B # LDA,
86 38 # STA,
87 C2 # LDA,
88 39 # STA,
```

ORIGINAL PAGE IS
OF POOR QUALITY

```

89 C200 JSR, ( JUMP TO SUBROUTINE TO READ CLOCK )
90 0289 LDA,
91 A2 STA,
92 BEGIN,
93 C219 LDA,
94 0289 STA,
95 C200 JSR, ( CONTINUE TO READ CLOCK UNTILL MINUTE FLIPS )
96 U289 LDA,
97 A0 CMP,
98 ENDDO,
99 0293 LDA, ( READ TIME INTO MEMORY LOCATIONS 4076-407F )
100 4076 STA,
101 0293 LDA,
102 4077 STA,
103 0293 LDA,
104 4078 STA,
105 0293 LDA,
106 4079 STA,
107 0293 LDA,
108 4080 STA,
109 0293 LDA,
110 4081 STA,
111 0293 LDA,
112 4082 STA,
113 0293 LDA,
114 4083 STA,
115 0293 LDA,
116 4084 STA,
117 0293 LDA,
118 4085 STA,
119 0293 LDA,
120 4086 STA,
121 39 PLA,
122 4D * LDA, ( SET INTEGRATION TABLE POINTERS )
123 AC * STA, ( FOR INCOMING DATA )
124 4C * LDA,
125 AE * STA,
126 B7 * LDA,
127 AB * STA, ( THESE ARE POINTERS FOR THE ISR )
128 AD * STA,
129 81 * LDA,
130 C70E * STA, ( ENABLE INTERRUPTS ON VIA )
131 C78E * STA,
132 * CLI, ( CLEAR INTERRUPT STATUS FLAG )
133 * LDA,
134 C701 * LDA,
135 BEGIN,
136 C1 * LDA, ( WAIT TILL SAMPLE READY FOR PROCESSING )
137 01 * CMP,
138 ENDRN,
139 00 * LDA, ( CLEAR FLAGS FOR SAMPLE READY AND END OF MINUTE )
140 C1 * STA,
141 BE * STA,
142 40 * LDA, ( SET LAG TABLE POINTERS FOR COMPUTATION )
143 8D * STA,
144 8B * STA,
145 00 * LDA,
146 8C * STA,
147 88 * STA,
148 86 * STA,
149 70 * LDA,
150 87 * STA,
151 80 * LDA,
152 8A * STA,
153 86 * LDA,
154 89 * STA,
155 4C * LDA,
156 A4 * STA,
157 4D * LDA,
158 A6 * STA, ( FIND ZERO LAG VALUES )
159 B1 * LDA,
160 A3 * STA, ( SET COMPUTATION POINTERS )
161 A5 * STA, ( SET HTS=8 )
162 08 * LDA,
163 A3 * STA,
164 00 * LDY,
165 A7 * STY, ( SET LAGS DONE = 0 )
166 96 * STY,
167 A3 * Y LDA, ( LOAD LAG '0' VALUES INTO ZERO PAGE )
168 91 * STA, ( FOR COMPUTATION )
169 A5 * Y LDA, ( N-W ANTENNA QUAD 1 )
170 96 * STA,
171 A3 * Y LDA, ( N-E ANTENNA QUAD 2 )
172 91 * STA,
173 97 * LDA,
174 97 * STA,
175 97 * STA,
176 A3 * Y LDA, ( S-W ANTENNA QUAD 3 )
177 93 * STA,
178 A5 * Y LDA,
179 98 * STA,
180 97 * STA,
181 A3 * Y LDA, ( S-E ANTENNA QUAD 4 )
182 94 * STA,
183 A5 * Y LDA,
184 99 * STA,
185 00 * LDY, ( RESET Y REG FOR LAG TABLE INDEX )
186 04 * LDA,
187 E0 * STA, ( RESET OPERATION POINTER )
188 A7 * LDY, ( OPERATION POINTER POINTS TO ANTENNA FROM LAG'N' )
189 B1 * X LDA, ( CALCULATE ADDRESS FOR NEXT LAG VARIABLE )
190 A2 * LDY,

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

192  DEX,
193  CLC,
194  E0 ,X ADC,
195  E0 ,X ADC,
196  TAX,
197  DEX,
198  4C00 ,X LDA, ( MOVE TO ZERO PAGE FOR PROCESSING )
199  80 ,X STA, ( ANTENNAS ARE PROCESSED FOR LAG 'N' IN THE ORDER )
200  4000 ,X LDA, ( QUAD 1,2,3,4 FOR LAG '0' THEY ARE PROCESSED IN THE )
201  04 ,X STA, ( IN ORDER: QUAD 4,3,2,1 )
202  04 ,X LDA,
203  BEGIN,
204  80 LDA,
205  COD0 STA,
206  81 LDA,
207  IFMI, ( DETERMINE IF VALUE NEG OR POS AND )
208  COD0 STA, ( LOAD ARITH PROCESSOR CARD FOR MULTIPLICATION )
209  FF ,X LDA,
210  COD0 STA,
211  COD0 STA,
212  ELSE,
213  COD0 STA,
214  00 ,X LDA,
215  COD0 STA,
216  COD0 STA,
217  THEN, ( LOAD VALUE FROM LAG ZERO INTO ARITH PROCESSOR CARD )
218  90 ,X LDA,
219  COD0 STA,
220  95 ,X LDA,
221  IFM,
222  COD0 STA,
223  FF ,X LDA,
224  COD0 STA,
225  COD0 STA,
226  ELSE,
227  COD0 STA,
228  00 ,X LDA,
229  COD0 STA,
230  COD0 STA,
231  THEN, ( LOAD CMD INTO PROCESSOR FOR 32-BIT MULT )
232  AE ,X LDA,
233  COD1 STA,
234  COD1 LDA,
235  COD1 LDA,
236  BEGIN, ( WAIT TILL MULTIPLICATION COMPLETE )
237  COD1 LDA,
238  COD0 LDA, ( PULL OFF RESULT )
239  PHA,
240  COD0 LDA,
241  PHA,
242  COD0 LDA,
243  PHA,
244  COD1 LDA,
245  CLC,
246  86 }Y ADC, ( ADD TO APPROPRIATE MEMORY LOCATION )
247  86 }Y STA, ( IN LAG TABLE )
248  PLA,
249  88 }Y ADC,
250  88 }Y STA,
251  PLA,
252  8A }Y ADC,
253  8A }Y STA,
254  PLA,
255  8C }Y ADC,
256  8C }Y STA,
257  PLA,
258  INI,
259  DEX, ( GO AFTER NEXT ANTENNA TO BE PROCESSED FROM ZERO LAG )
260  ENDNE,
261  E0 DEC,
262  IFNE,
263  PADDR2 ( ) JMP, ( GO AFTER NEXT ANTENNA FROM LAG 'N' )
264  THEN,
265  04 ,X LDA,
266  E0 STA,
267  A7 INC,
268  A7 LDA,
269  06 ,X CMP,
270  IFNE,
271  PADDR2 ( ) JMP, ( GO AFTER NEXT LAG )
272  THEN,
273  8D INC, ( IF ALL LAGS AND OPERATIONS DONE INC )
274  8B INC, ( POINTERS AND GO FOR NEXT HT )
275  CLC,
276  80 ,X LDA,
277  88 ADC,
278  STA,
279  00 ,X LDA,
280  89 ADC,
281  STA,
282  80 ,X LDA,
283  86 ADC,
284  STA,
285  00 ,X LDA,
286  87 ADC,
287  STA,
288  04 ,X LDA,
289  87 CLC,
290  A3 ,X LDA,

```



```

391 IFNE, ( COMPARE FOR ALL ANTENNAS SAMPLED ? )
392 BC STA,
393 STA,
394 B8 LDA, ( IF NOT JUST SWITCH ANTENNA )
395 C700 X STA,
396 00 LDA, ( CLEAR INTEGRATION COUNTER )
397 AA STA,
398 C0 INC, ( INC STALL COUNTER FOR RELAY SWITCHING )
399 INC,
400 TAX,
401 PLA, ( RETURN FROM INTERRUPT )
402 TAX,
403 C701 LDA,
404 45 LDA,
405 RTI,
406 ELSE,
407 B7 LDA, ( IF ALL ANTENNAS SAMPLED THEN ROTATE POINTERS )
408 PHA, ( THESE POINTERS ARE LOOK-UP VECTORS FOR POINTING )
409 B6 LDA, ( AT LAGS 0-5 PLUS A 6TH SLOT WHICH TAKES ON THE )
410 B7 STA, ( THE NEXT SAMPLE )
411 B5 LDA,
412 B6 STA,
413 B4 LDA,
414 B5 STA,
415 B3 LDA,
416 B4 STA,
417 B2 LDA,
418 B3 STA,
419 B1 LDA,
420 B2 STA,
421 B1 PLA,
422 B1 STA,
423 B1 STA, ( FIX POINTERS FOR ZERO LAG )
424 B3 STA,
425 B3 LDA,
426 B0 STA,
427 BC LDA,
428 B8 TAX,
429 C700 X LDA, ( SWITCH RELAY TO QUAD1 FOR INPUT )
430 00 # LDA,
431 AA STA,
432 1F LDY,
433 BEGIN, ( SUM UP VALUES FOR THIS SAMPLE TO COMPUTE )
434 CLC, ( MEAN OF EACH ANTENNA AND EACH HT )
435 A3 Y LDA,
436 4E64 Y ADC,
437 4E64 Y STA,
438 A5 Y LDA,
439 4E32 Y ADC,
440 4E32 Y STA,
441 IFCS,
442 00 LDA,
443 4E00 Y ADC,
444 4E00 Y STA,
445 ELSE,
446 THEN,
447 DEY,
448 ENDPL,
449 B7 LDA, ( CLEAR NEXT LAG AREA FOR INCOMING VALUES )
450 AD STA,
451 AB STA,
452 1F LDY,
453 00 LDA,
454 BEGIN,
455 AD Y STA,
456 AB Y STA,
457 DEY,
458 ENDPL,
459 01 LDA, ( SET FLAG FOR NEXT SAMPLE READY )
460 C1 STA,
461 BD INC,
462 B1 LDA,
463 BD CMP, ( CHECK TO SEE IF ALL SAMPLES COLLECTED )
464 IFNE, ( B1 HEX = 51 SEC COLLECTION TIME )
465 C0 INC, ( ELSE INC STALL COUNTER AND CHECK FOR END OF STALL PERIOD )
466 0A LDA,
467 C0 CMP,
468 IFNE,
469 PLA,
470 TAX,
471 PLA, ( RETURN FROM INTERRUPT )
472 TAX,
473 C701 LDA,
474 45 LDA,
475 RTI,
476 ELSE,
477 00 LDA, ( IF STALL COUNT = MAX COUNT CLEAR FLAGS )
478 E4 STA,
479 C0 STA,
480 PLA, ( RETURN FROM INTERRUPT )
481 TAX,
482 TAX,
483 PLA,
484 C701 LDA,
485 45 LDA,
486 RTI,
487 THEN,
488 ELSE,
489 01 LDA,
490 BE STA, ( IF ALL SAMPLES TAKEN SET FLAG FOR END OF MINUTE )

```

```

401 00 # LDA,
402 EA STA,
403 C0 STA,
404 PLA,
405 TAY,
406 PLA,
407 TAX,
408 LDA,
409 LDA,
410 RTI,
411
412 C701
413 45
414
415 THEN,
416 THEN,
417 ENDCODE ( END OF CODE FOR WAITISR )
418 ( STALL ISR IS PART OF THE ISR. IT IS USED TO BYPASS READING
419 DATA IN THE ISR UNTILL ADEQUATE TIMEIS GIVEN FOR THE ANTENNA )
420 RELAY TO CHANGE QUADRANTS.
421 CODE STALLISR ( START OF CODE FOR STALL ISR )
422 CO INC, ( INC COUNTER FOR STALL COUNT )
423 OA # LDA,
424 CO CMP, ( COMPARE FOR MAXIMUM COUNT )
425 IFNE, ( IF <> THEN RETURN FROM INTERRUPT )
426 PLA,
427 TAY,
428 PLA,
429 TAX,
430 LDA,
431 LDA,
432 RTI,
433
434 C701
435 45
436
437 ELSE,
438 00 # LDA, ( OTHERWISE CLEAR COUNT AND STALL FLAG )
439 EA STA,
440 C0 STA,
441 PLA, ( RETURN FROM INTERRUPT )
442 TAY,
443 PLA,
444 TAX,
445 LDA,
446 LDA,
447 RTI,
448
449 C701
450 45
451
452 THEN,
453 ENDCODE ( END OF CODE STALLISR )
454 : ISRCR ( INTERRUPT SERVICE ROUTINE "ISR" )
455 8A C, 9848, 48 C,
456 01A9, E4C5, 03D0, 6C C, ( CHECK FOR STALL AND INTEGRATION COUNT FLAGS )
457 LW, AA45, 0AC9, 03D0,
458 6C C, WAIT, 00A0,
459 80A9, 8D C,
460 C780, 8D C, ( RESET PB7 )
461 C058, 8EB1,
462 99 C, 4E90,
463 4EA5, 4E91 DO ( READ IN DATA )
464 8EB1, 99 C, I, LOOP
465 8D C, C059, 06A2,
466 18 C, B1B5, BCG5,
467 AD8 C, AB85, AE C,
468 HGATE,
469 8 0 DC,
470 1 400 * A0 + ( ADD DATA TO BUFFER )
471 18 C, ADB1, 7D C,
472 4E90, AD91, ABB1,
473 0069, AB91, E8E8, E8 C,
474 LOOP,
475 AA66, A868, AA68, AD C, C701, ( RESTORE REG AND RETURN FROM ISR )
476 45A5, 40 C,
477 : ISRCR, -5 DTOP +1
478 : ISRCR FLUSH DESTROY
479 : PADDRC WAITISR WAIT ! ( COMPUTE ADDRESS IN VERBS WAITISR, )
480 : STALLISR LW ! DCORR C1 + PADDR ! ( STALLISR, AND DCORR )
481 : DCORR 11E + PADDR2 !,
482 : ZPRINT HEX ( INT ZERO PAGE LOCATIONS FOR PROGRAM )
483 3F00 3E00 DO 0 I C1 LOOP ( ALL VALUES WILL SWAP WITH ZERO PAGE )
484 4D00 3EA5 ( LOHB )
485 4D00 3EAB ( LCIHB )
486 4C00 3EA3 ( LCILB )
487 4C00 3EAD ( LOLB )
488 C701 3EBE ( PORT A )
489 00 3EAA C1 ( CICT )
490 00 3E84 C1 ( STALL )
491 00 3E8C C1 ( ANT )
492 00 3E8D C1 ( FFC )
493 00 3E8E C1 ( DFLAG )
494 00 3E8F C1 ( SCT )
495 00 3E90 C1 ( DFG )
496 00 3E91 C1 ( NLAG )
497 00 3E92 C1
498 00 3E93 C1
499 00 3E94 C1
500 00 3E95 C1
501 00 3E96 C1
502 00 3E97 C1
503 00 3E98 C1
504 00 3E99 C1 ( ASW )
505 00 3E9A C1
506 00 3E9B C1
507 00 3E9C C1 ( NLHT )
508 00 3E9D C1
509 00 3E9E C1
510 00 3E9F C1
511 00 3E90 C1
512 00 3E91 C1
513 00 3E92 C1
514 00 3E93 C1
515 00 3E94 C1
516 00 3E95 C1
517 00 3E96 C1
518 00 3E97 C1
519 00 3E98 C1
520 00 3E99 C1
521 00 3E9A C1
522 00 3E9B C1
523 00 3E9C C1
524 00 3E9D C1
525 00 3E9E C1
526 00 3E9F C1
527 00 3E90 C1
528 00 3E91 C1
529 00 3E92 C1
530 00 3E93 C1
531 00 3E94 C1
532 00 3E95 C1
533 00 3E96 C1
534 00 3E97 C1
535 00 3E98 C1
536 00 3E99 C1
537 00 3E9A C1
538 00 3E9B C1
539 00 3E9C C1
540 00 3E9D C1
541 00 3E9E C1
542 00 3E9F C1
543 00 3E90 C1
544 00 3E91 C1
545 00 3E92 C1
546 00 3E93 C1
547 00 3E94 C1
548 00 3E95 C1
549 00 3E96 C1
550 00 3E97 C1
551 00 3E98 C1
552 00 3E99 C1
553 00 3E9A C1
554 00 3E9B C1
555 00 3E9C C1
556 00 3E9D C1
557 00 3E9E C1
558 00 3E9F C1
559 00 3E90 C1
560 00 3E91 C1
561 00 3E92 C1
562 00 3E93 C1
563 00 3E94 C1
564 00 3E95 C1
565 00 3E96 C1
566 00 3E97 C1
567 00 3E98 C1
568 00 3E99 C1
569 00 3E9A C1
570 00 3E9B C1
571 00 3E9C C1
572 00 3E9D C1
573 00 3E9E C1
574 00 3E9F C1
575 00 3E90 C1
576 00 3E91 C1
577 00 3E92 C1
578 00 3E93 C1
579 00 3E94 C1
580 00 3E95 C1
581 00 3E96 C1
582 00 3E97 C1
583 00 3E98 C1
584 00 3E99 C1
585 00 3E9A C1
586 00 3E9B C1
587 00 3E9C C1
588 00 3E9D C1
589 00 3E9E C1
590 00 3E9F C1
591 00 3E90 C1
592 00 3E91 C1
593 00 3E92 C1
594 00 3E93 C1
595 00 3E94 C1
596 00 3E95 C1
597 00 3E96 C1
598 00 3E97 C1
599 00 3E98 C1
600 00 3E99 C1
601 00 3E9A C1
602 00 3E9B C1
603 00 3E9C C1
604 00 3E9D C1
605 00 3E9E C1
606 00 3E9F C1
607 00 3E90 C1
608 00 3E91 C1
609 00 3E92 C1
610 00 3E93 C1
611 00 3E94 C1
612 00 3E95 C1
613 00 3E96 C1
614 00 3E97 C1
615 00 3E98 C1
616 00 3E99 C1
617 00 3E9A C1
618 00 3E9B C1
619 00 3E9C C1
620 00 3E9D C1
621 00 3E9E C1
622 00 3E9F C1
623 00 3E90 C1
624 00 3E91 C1
625 00 3E92 C1
626 00 3E93 C1
627 00 3E94 C1
628 00 3E95 C1
629 00 3E96 C1
630 00 3E97 C1
631 00 3E98 C1
632 00 3E99 C1
633 00 3E9A C1
634 00 3E9B C1
635 00 3E9C C1
636 00 3E9D C1
637 00 3E9E C1
638 00 3E9F C1
639 00 3E90 C1
640 00 3E91 C1
641 00 3E92 C1
642 00 3E93 C1
643 00 3E94 C1
644 00 3E95 C1
645 00 3E96 C1
646 00 3E97 C1
647 00 3E98 C1
648 00 3E99 C1
649 00 3E9A C1
650 00 3E9B C1
651 00 3E9C C1
652 00 3E9D C1
653 00 3E9E C1
654 00 3E9F C1
655 00 3E90 C1
656 00 3E91 C1
657 00 3E92 C1
658 00 3E93 C1
659 00 3E94 C1
660 00 3E95 C1
661 00 3E96 C1
662 00 3E97 C1
663 00 3E98 C1
664 00 3E99 C1
665 00 3E9A C1
666 00 3E9B C1
667 00 3E9C C1
668 00 3E9D C1
669 00 3E9E C1
670 00 3E9F C1
671 00 3E90 C1
672 00 3E91 C1
673 00 3E92 C1
674 00 3E93 C1
675 00 3E94 C1
676 00 3E95 C1
677 00 3E96 C1
678 00 3E97 C1
679 00 3E98 C1
680 00 3E99 C1
681 00 3E9A C1
682 00 3E9B C1
683 00 3E9C C1
684 00 3E9D C1
685 00 3E9E C1
686 00 3E9F C1
687 00 3E90 C1
688 00 3E91 C1
689 00 3E92 C1
690 00 3E93 C1
691 00 3E94 C1
692 00 3E95 C1
693 00 3E96 C1
694 00 3E97 C1
695 00 3E98 C1
696 00 3E99 C1
697 00 3E9A C1
698 00 3E9B C1
699 00 3E9C C1
700 00 3E9D C1
701 00 3E9E C1
702 00 3E9F C1
703 00 3E90 C1
704 00 3E91 C1
705 00 3E92 C1
706 00 3E93 C1
707 00 3E94 C1
708 00 3E95 C1
709 00 3E96 C1
710 00 3E97 C1
711 00 3E98 C1
712 00 3E99 C1
713 00 3E9A C1
714 00 3E9B C1
715 00 3E9C C1
716 00 3E9D C1
717 00 3E9E C1
718 00 3E9F C1
719 00 3E90 C1
720 00 3E91 C1
721 00 3E92 C1
722 00 3E93 C1
723 00 3E94 C1
724 00 3E95 C1
725 00 3E96 C1
726 00 3E97 C1
727 00 3E98 C1
728 00 3E99 C1
729 00 3E9A C1
730 00 3E9B C1
731 00 3E9C C1
732 00 3E9D C1
733 00 3E9E C1
734 00 3E9F C1
735 00 3E90 C1
736 00 3E91 C1
737 00 3E92 C1
738 00 3E93 C1
739 00 3E94 C1
740 00 3E95 C1
741 00 3E96 C1
742 00 3E97 C1
743 00 3E98 C1
744 00 3E99 C1
745 00 3E9A C1
746 00 3E9B C1
747 00 3E9C C1
748 00 3E9D C1
749 00 3E9E C1
750 00 3E9F C1
751 00 3E90 C1
752 00 3E91 C1
753 00 3E92 C1
754 00 3E93 C1
755 00 3E94 C1
756 00 3E95 C1
757 00 3E96 C1
758 00 3E97 C1
759 00 3E98 C1
760 00 3E99 C1
761 00 3E9A C1
762 00 3E9B C1
763 00 3E9C C1
764 00 3E9D C1
765 00 3E9E C1
766 00 3E9F C1
767 00 3E90 C1
768 00 3E91 C1
769 00 3E92 C1
770 00 3E93 C1
771 00 3E94 C1
772 00 3E95 C1
773 00 3E96 C1
774 00 3E97 C1
775 00 3E98 C1
776 00 3E99 C1
777 00 3E9A C1
778 00 3E9B C1
779 00 3E9C C1
780 00 3E9D C1
781 00 3E9E C1
782 00 3E9F C1
783 00 3E90 C1
784 00 3E91 C1
785 00 3E92 C1
786 00 3E93 C1
787 00 3E94 C1
788 00 3E95 C1
789 00 3E96 C1
790 00 3E97 C1
791 00 3E98 C1
792 00 3E99 C1
793 00 3E9A C1
794 00 3E9B C1
795 00 3E9C C1
796 00 3E9D C1
797 00 3E9E C1
798 00 3E9F C1
799 00 3E90 C1
800 00 3E91 C1
801 00 3E92 C1
802 00 3E93 C1
803 00 3E94 C1
804 00 3E95 C1
805 00 3E96 C1
806 00 3E97 C1
807 00 3E98 C1
808 00 3E99 C1
809 00 3E9A C1
810 00 3E9B C1
811 00 3E9C C1
812 00 3E9D C1
813 00 3E9E C1
814 00 3E9F C1
815 00 3E90 C1
816 00 3E91 C1
817 00 3E92 C1
818 00 3E93 C1
819 00 3E94 C1
820 00 3E95 C1
821 00 3E96 C1
822 00 3E97 C1
823 00 3E98 C1
824 00 3E99 C1
825 00 3E9A C1
826 00 3E9B C1
827 00 3E9C C1
828 00 3E9D C1
829 00 3E9E C1
830 00 3E9F C1
831 00 3E90 C1
832 00 3E91 C1
833 00 3E92 C1
834 00 3E93 C1
835 00 3E94 C1
836 00 3E95 C1
837 00 3E96 C1
838 00 3E97 C1
839 00 3E98 C1
840 00 3E99 C1
841 00 3E9A C1
842 00 3E9B C1
843 00 3E9C C1
844 00 3E9D C1
845 00 3E9E C1
846 00 3E9F C1
847 00 3E90 C1
848 00 3E91 C1
849 00 3E92 C1
850 00 3E93 C1
851 00 3E94 C1
852 00 3E95 C1
853 00 3E96 C1
854 00 3E97 C1
855 00 3E98 C1
856 00 3E99 C1
857 00 3E9A C1
858 00 3E9B C1
859 00 3E9C C1
860 00 3E9D C1
861 00 3E9E C1
862 00 3E9F C1
863 00 3E90 C1
864 00 3E91 C1
865 00 3E92 C1
866 00 3E93 C1
867 00 3E94 C1
868 00 3E95 C1
869 00 3E96 C1
870 00 3E97 C1
871 00 3E98 C1
872 00 3E99 C1
873 00 3E9A C1
874 00 3E9B C1
875 00 3E9C C1
876 00 3E9D C1
877 00 3E9E C1
878 00 3E9F C1
879 00 3E90 C1
880 00 3E91 C1
881 00 3E92 C1
882 00 3E93 C1
883 00 3E94 C1
884 00 3E95 C1
885 00 3E96 C1
886 00 3E97 C1
887 00 3E98 C1
888 00 3E99 C1
889 00 3E9A C1
890 00 3E9B C1
891 00 3E9C C1
892 00 3E9D C1
893 00 3E9E C1
894 00 3E9F C1
895 00 3E90 C1
896 00 3E91 C1
897 00 3E92 C1
898 00 3E93 C1
899 00 3E94 C1
900 00 3E95 C1
901 00 3E96 C1
902 00 3E97 C1
903 00 3E98 C1
904 00 3E99 C1
905 00 3E9A C1
906 00 3E9B C1
907 00 3E9C C1
908 00 3E9D C1
909 00 3E9E C1
910 00 3E9F C1
911 00 3E90 C1
912 00 3E91 C1
913 00 3E92 C1
914 00 3E93 C1
915 00 3E94 C1
916 00 3E95 C1
917 00 3E96 C1
918 00 3E97 C1
919 00 3E98 C1
920 00 3E99 C1
921 00 3E9A C1
922 00 3E9B C1
923 00 3E9C C1
924 00 3E9D C1
925 00 3E9E C1
926 00 3E9F C1
927 00 3E90 C1
928 00 3E91 C1
929 00 3E92 C1
930 00 3E93 C1
931 00 3E94 C1
932 00 3E95 C1
933 00 3E96 C1
934 00 3E97 C1
935 00 3E98 C1
936 00 3E99 C1
937 00 3E9A C1
938 00 3E9B C1
939 00 3E9C C1
940 00 3E9D C1
941 00 3E9E C1
942 00 3E9F C1
943 00 3E90 C1
944 00 3E91 C1
945 00 3E92 C1
946 00 3E93 C1
947 00 3E94 C1
948 00 3E95 C1
949 00 3E96 C1
950 00 3E97 C1
951 00 3E98 C1
952 00 3E99 C1
953 00 3E9A C1
954 00 3E9B C1
955 00 3E9C C1
956 00 3E9D C1
957 00 3E9E C1
958 00 3E9F C1
959 00 3E90 C1
960 00 3E91 C1
961 00 3E92 C1
962 00 3E93 C1
963 00 3E94 C1
964 00 3E95 C1
965 00 3E96 C1
966 00 3E97 C1
967 00 3E98 C1
968 00 3E99 C1
969 00 3E9A C1
970 00 3E9B C1
971 00 3E9C C1
972 00 3E9D C1
973 00 3E9E C1
974 00 3E9F C1
975 00 3E90 C1
976 00 3E91 C1
977 00 3E92 C1
978 00 3E93 C1
979 00 3E94 C1
980 00 3E95 C1
981 00 3E96 C1
982 00 3E97 C1
983 00 3E98 C1
984 00 3E99 C1
985 00 3E9A C1
986 00 3E9B C1
987 00 3E9C C1
988 00 3E9D C1
989 00 3E9E C1
990 00 3E9F C1
991 00 3E90 C1
992 00 3E91 C1
993 00 3E92 C1
994 00 3E93 C1
995 00 3E94 C1
996 00 3E95 C1
997 00 3E96 C1
998 00 3E97 C1
999 00 3E98 C1
1000 00 3E99 C1

```

```

501 04 JEC8 CI
502 00 JEC9 CI
503 ;
504 ; ISRSET ' ISR 3FE ' , ( SET ADDRESS FOR ISR )
505 ; ISRRESET FF65 3FE ' , ( RESET ADDRESS FOR ISR )
506 ; MAINDRIFT PAGE
507 PADDRC INITI/O *PINIT ISRSET
508
509 START OF COLLECTION
510
511 DCORR
512
513 FIRST MINUTE WASTED TO GET MEAN
514
515 MINUTES @ 0 DO
516 DCORR
517 STDOS CNT
518 " BSAVEDATA" S " I
519 " AS4000,LSC00" S: CR STNRM CR
520 4E9F 4E90 DO I C@ . " S. LOOP
521 LOOP
522 ISRRESET
523 ;
524 ; RESULTS
525 IF 00 DO 4000 I + C@ 100 *
526 4080 I + C@ + CR DEC I .
527 " S. . HEX LOOP CR HEX ;
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618

```

ORIGINAL PAGE IS
OF POOR QUALITY

I.1 MEMORY USAGE FOR COLLECTION SOFTWARE.

| ADDRESS IN HEX | ABBREVIATED SYMBOL | DEFINITION OF VARIABLE |
|-------------------|-----------------------|---|
| 80-81 | — | Temp. storage for computation of mult. |
| 86-87 | RLL | Lag table point to LSB |
| 88-89 | RLL | Lag table pointer to low byte |
| 8A-8B | RLM | Lag table pointer to med byte |
| 8C-8D | RLH | Lag table pointer to high byte |
| 8E-8F | PA | Contains address for input port |
| 90-99 | — | Temp. storage for lag zero values during mult. |
| A6-A1 | TEMP | Temporary storage byte |
| A2 | HTS | Height counter used for processing |
| A3-A4 | LOLB | Lag 0 low byte pointer |
| A5-A6 | LOAB | Lag 0 high byte pointer |
| A7 | LAG | Lag counter for processing |
| AA | CICT | Integration counter |
| AB-AC | LCIHB | Incoming integration counter high byte |
| AD-AE | LCILB | Incoming integration counter low byte |
| B1-B7 | NLAG | Offset pointers for computation |
| B8-BB | ASW | Antenna switching values |
| BC | ANT | Antenna counter (index to ASW) |
| BD | FFC | Counter for samples collected vs. EOR |
| BE | DFLAG | End of record flag (EOR) |
| C0 | SCT | Stall counter for ISR |
| C1 | DFG | Done flag for sample ready |

ORIGINAL PAGE IS
OF POOR QUALITY

66

| | | |
|-------|-------|----------------------|
| C2-C9 | NLHT | Computation variable |
| E0-E1 | TEMP | Temp. storage bytes |
| E4 | STALL | Stall flag byte |

I.2 MEMORY MAP.

LAG TABLE

| | | | |
|------|-------|-------|-----------|
| 7300 | | | HTS 7 & 8 |
| 7200 | | | HTS 5 & 6 |
| 7100 | | | HTS 3 & 4 |
| 7000 | LSB's | LSB's | HTS 1 & 2 |

LAG TABLE

| | | | | | |
|------|------------------------------------|--------------|-----------|-------------|-------|
| | | 4E90 | 4EAO | | |
| 4E00 | Incoming mean values | Input Buffer | | | |
| 4D00 | High bytes of last 6 lags, Hts 1-8 | | | | |
| 4C00 | Low bytes of last 6 lags, Hts 1-8 | | | | |
| 4B00 | | | | HTS 7 & 8 | |
| 4A00 | | | | HTS 5 & 6 | |
| 4900 | | | | HTS 3 & 4 | |
| 4800 | Low Bytes | Low Bytes | | HTS 1 & 2 | |
| 4700 | | | | HT #8 | |
| 4600 | | | | HT #7 | |
| 4500 | | | | HT #6 | |
| 4400 | | | | HT #5 | |
| 4300 | | | | HT #4 | |
| 4200 | | | | HT #3 | |
| 4100 | | | | HT #2 | |
| 4000 | High Bytes | Time | Med Bytes | Mean Values | HT #1 |

APPENDIX II
POSTPROCESSING PROGRAMS.

II.1 3-ANTENNAS.

```

1  REM 1 CS8A00
10  DIM CC(8,3,11),AC(8,3,6),EV(8),AM(8),P(8,3),TD(8,3),VX(3),VY(3),CA(3)
20  DATA 1,2,7,13,1,3,4,12,3
30  CA(1) = 0.0:CA(2) = 1.5:CA(3) = 2.356194
40  IM(1) = 6:IM(2) = 0:IM(3) = 9
50  ID(1) = 5:ID(2) = 15:ID(3) = 0
60  CA(1) = .2:CA(2) = .1:CA(3) = .3
70  PRINT
80  PRINT "INPUT PARM FOR OUTPUT FILE"
85  PRINT "SLOT#"; INPUT SR
86  PRINT "VOLUME#"; INPUT VR
87  PRINT "INPUT PARM FOR DATA FILES"
89  PRINT "STARTING FILE#"; INPUT MS
90  PRINT "ENDING FILE#"; INPUT ME
100  PRINT "STARTING VOLUME"; INPUT VS
105  PRINT "ENDING VOLUME"; INPUT VE
110  DS = CHR$(4)
115  PRINT DS;"OPENRESULTS,S";SR;"V";VR
120  PRINT DS;"WRITERESULTS"
125  PRINT DS;"START OF FILE"
130  PRINT DS;"CLOSE RESULTS"
135  FOR VZ = VS TO VE
140  FOR MF = MS TO ME
150  DS = CHR$(4); REM CTRL-D
160  PRINT & PRINT "WORKING ON FILE#";MF
165  PRINT & VOLUME#";VZ
170  PRINT DS;"BLOADDATA",MF;"AS8A00,S6,V";VZ
175  PRINT DS;"BLOADDAT",MF;"AS9600,S6,V";VZ
180  FOR H = 1 TO 8:EV(H) = 0:NEXT H
185  Z = 0: FOR N = 35448 TO 35455:Z = Z * 10 + PEEK(N) - 176
190  NEXT N
200  S1 = 35328:S2 = 35456:S3 = 37376:S4 = 38400
210  FOR H = 1 TO 8
215  B1 = S1 + 256 * (H - 1):B2 = S2 + 256 * (H - 1):B3 = S3 + 128 * (H - 1)
220  B4 = S4 + 128 * (H - 1)
225  A = 0: FOR I = 1 TO 3:IA = ID(I)
230  A = A + 1:L = 0: FOR IL = 0 TO 80 STEP 16
235  AC(H,A,L) = 16777216 * PEEK(B1 + IL + IA) + 65536 * PEEK(B2 + IL + IA) + 256 * PEEK(B3 + IL + IA) + PEEK(B4 + IL + IA)
240  NEXT IL: NEXT I: NEXT H
244  GOSUB 3000
245  FOR C = 1 TO 3: READ J1,J2,J3,J4
250  FOR H = 1 TO 8:B1 = S1 + 256 * (H - 1) + J3:B2 = S2 + 256 * (H - 1) + J3:B3 = S3 + 128 * (H - 1) + J3:L = 5
255  B4 = S4 + 128 * (H - 1) + J3
260  FOR IL = 0 TO 80 STEP 16
265  L = L + 1
270  CC(H,C,L) = 16777216 * PEEK(B1 + IL) + 65536 * PEEK(B2 + IL) + 256 * PEEK(B3 + IL) + PEEK(B4 + IL)
275  CC(H,C,L) = (CC(H,C,L) / (134 - L) - SM(H,J1) * SM(H,J2)) / SQR(AC(H,J1,1) * AC(H,J2,1))
280  NEXT IL: NEXT H: FOR H = 1 TO 8:B1 = S1 + 256 * (H - 1) + J4:B2 = S2 + 256 * (H - 1) + J4:B3 = S3 + 256 * (H - 1) + J4:L = 6
285  B4 = S4 + 128 * (H - 1) + J4
290  FOR IL = 16 TO 80 STEP 16:L = L - 1
295  CC(H,C,L) = 16777216 * PEEK(B1 + IL) + 65536 * PEEK(B2 + IL) + 256 * PEEK(B3 + IL) + PEEK(B4 + IL)
300  CC(H,C,L) = (CC(H,C,L) / (122 + L) - SM(H,J1) * SM(H,J2)) / SQR(AC(H,J1,1) * AC(H,J2,1))
305  NEXT IL: NEXT H: NEXT C
310  RESTORE
315  FOR H = 1 TO 8: FOR A = 1 TO 3: FOR L = 2 TO 6
320  AC(H,A,L) = AC(H,A,L) / AC(H,A,1)
325  NEXT L: SM(H,A) = INT(100 * LOG(AC(H,A,1))): AC(H,A,1) = 1: NEXT A
330  NEXT H
480  FOR H = 1 TO 8
490  FOR L = 2 TO 6
500  AC(H,1,L) = (AC(H,1,L) + AC(H,2,L) + AC(H,3,L)) / 3.0
510  NEXT L: NEXT H
520  FOR H = 1 TO 8
530  FOR L = 1 TO 8
540  K = 1: MX = 1: FOR L = 2 TO 6
550  IF MX < AC(H,1,L) THEN 650
560  MX = AC(H,1,L)
570  K = L
580  NEXT L
590  AM(H) = MX
600  IF K < 2 THEN EV(H) = 1
610  NEXT H
620  REM START OF CURVE FIT
630  FOR H = 1 TO 8: FOR C = 1 TO 3
640  IF EV(H) < 0 THEN 920
650  MX = CC(H,C,1): K = 1
660  FOR L = 2 TO 11

```

```

730 IF MX > CC(H,C,L) THEN 755
740 MX = CC(H,C,L)
750 NEXT L
760 IF ABS (K - 6) > 3 THEN 920
770 C1 = CC(H,C,K) + CC(H,C,K - 2)
780 C2 = CC(H,C,K) + CC(H,C,K - 1)
790 C3 = CC(H,C,K) + CC(H,C,K - 1)
800 C4 = CC(H,C,K) + CC(H,C,K - 1)
810 C5 = CC(H,C,K) + CC(H,C,K - 1)
820 C6 = CC(H,C,K) + CC(H,C,K - 1)
830 C7 = CC(H,C,K) + CC(H,C,K - 1)
840 C8 = CC(H,C,K) + CC(H,C,K - 1)
850 C9 = CC(H,C,K) + CC(H,C,K - 1)
860 C10 = CC(H,C,K) + CC(H,C,K - 1)
870 C11 = CC(H,C,K) + CC(H,C,K - 1)
880 C12 = CC(H,C,K) + CC(H,C,K - 1)
890 C13 = CC(H,C,K) + CC(H,C,K - 1)
900 C14 = CC(H,C,K) + CC(H,C,K - 1)
910 C15 = CC(H,C,K) + CC(H,C,K - 1)
920 GOTO 950
930 IF ABS (K - 6) > 4 THEN 925
940 GOTO 911
950 EV(H) = 2: GOTO 950
960 P(H,C) = (((P1 * X + Q) * X + R) * X + S) * X / 10 + MX
970 TD(H,C) = ((K - 6) + X) * .4 + CA(C)
980 NEXT C: NEXT H
990 REM END OF CURVE FIT
1000 PRINT "APPENDRESULTS";S";SR";"V";VR
1010 PRINT "WRITERESULTS";S";SR";"V";VR
1020 PRINT "TIME";Z
1030 FOR H = 1 TO 8
1040 PM = .5: GOSUB 7000: T5 = TT
1050 IF EV(H) < > 0 THEN 1430
1060 PM = P(H,1)
1070 GOSUB 7000
1080 IF EV(H) < > 0 THEN 1430
1090 V1 = 169 / SQR (TD(H,1) * TD(H,1) + TT * TT)
1100 PM = P(H,2)
1110 GOSUB 7000
1120 IF EV(H) < > 0 THEN 1430
1130 V2 = 169 / SQR (TD(H,2) * TD(H,2) + TT * TT)
1140 PM = P(H,3)
1150 GOSUB 7000
1160 IF EV(H) < > 0 THEN 1430
1170 V3 = 239 / SQR (TD(H,3) * TD(H,3) + TT * TT)
1180 A1 = CA(1): A2 = CA(2): A3 = CA(3)
1190 GOSUB 1300: IF EV(H) < > 0 THEN 1430
1200 V1 = 169 / SQR (TD(H,1) * TD(H,1) + TT * TT)
1210 V2 = 169 / SQR (TD(H,2) * TD(H,2) + TT * TT)
1220 V3 = 239 / SQR (TD(H,3) * TD(H,3) + TT * TT)
1230 NT = ABS (TD(H,1) + TD(H,2) - TD(H,3))
1240 NT = ABS (ABS (TD(H,1)) + ABS (TD(H,2)) + ABS (TD(H,3)))
1250 GOSUB 7799
1260 PRINT H";EV(H)";"SM(H,1)";"SM(H,2)";"SM(H,3)";"INT (100 * T5)
1270 )
1280 IF EV(H) < > 0 THEN 1465
1290 PRINT INT (VT / 2);"INT (TT * 57.296)";"
1300 PRINT INT (AX);"INT (10 * AX / BK)";"INT (10 * NT)";"
1310 PRINT INT (VC);"INT (57.296 * TN)";"
1320 PRINT INT (10 * RV)
1330 NEXT H
1340 PRINT "CLOSERESULTS"
1350 NEXT MP
1360 NEXT VZ
1370 PRINT "DONE PROCESSING"
1380 NEW: END
1390 B1 = 15572
1400 FOR H = 1 TO 8: FOR A = 1 TO 3
1410 I = IM(A)
1420 B2 = 65536 * PEEK (B1 + I) + 256 * PEEK (B1 + I + 1) + PEEK (B1 + I + 2)
1430 FOR L = 1 TO 6
1440 AC(H,A,L) = AC(H,A,L) / (129 - L) - (B2 * B2) / 16384.00
1450 NEXT L
1460 AC(H,A,1) = (4 * AC(H,A,2) - AC(H,A,3)) / 3.
1470 IF AC(H,A,1) < 0 THEN GOSUB 4000
1480 SM(H,A) = B2 / 128
1490 NEXT A: B1 = B1 + 256: NEXT H: RETURN
1500 EV(H) = 3
1510 AC(H,A,1) = 1.0
1520 RETURN
1530 REM SUB TO CAL PTS OF AUTOCORR
1540 IF PM > = AM(H) THEN 7030
1550 EV(H) = 6
1560 RETURN
1570 EV(H) = 5: RETURN
1580 AI = ABS (1 - PM)
1590 K = 1
1600 FOR L = 2 TO 6
1610 IF AI < ABS (PM - AC(H,L)) THEN 7090
1620 AI = ABS (PM - AC(H,L))
1630 K = L
1640 NEXT L: IF K > 5 THEN 7025
1650 IF K < > 1 THEN 7130

```

```

7110 TT = SQR (A1 / (1 - AC(H,1,2))) * .4
7120 RETURN
7130 A = .5 * (AC(H,1,K - 1) - 2 * AC(H,1,K) + AC(H,1,K + 1))
7140 B = .25 * (AC(H,1,K + 1) - AC(H,1,K - 1))
7150 C = AC(H,1,K)
7160 IF ABS(A) > .0001 THEN 7190
7170 TT = (K - 1 + (PM - C) / (2 * B)) * .4
7180 RETURN
7190 GH = B * B - A * (C - PM) : IF GH < 0.0 THEN 7170
7200 TT = (K - 1 - (B + SQR(GH)) / A) * .4
7210 RETURN
7220 REM SUB TO CAL ELLIPSE PARM
7230 D1 = 1 / (V1 * V1) - 1 / (V2 * V2)
7240 D2 = 1 / (V1 * V1) - 1 / (V3 * V3)
7250 TN = D1 * (COS(2 * A1) - COS(2 * A3)) - D2 * (COS(2 * A1) - COS(2 * A2))
7260 TN = SIN(2 * A1) - SIN(2 * A3) - D2 * (SIN(2 * A1) - SIN(2 * A2))
7270 ATN(TN)
7280 K1 = (1 + COS(2 * A1)) / (2 * (A1 - TN))
7290 K2 = (1 + COS(2 * A2)) / (2 * (A2 - TN))
7300 K3 = (1 + COS(2 * A3)) / (2 * (A3 - TN))
7310 AS = AS / (K3 / (V1 * V1) - K1 / (V2 * V2))
7320 BS = BS / (K2 / (V2 * V2) - K4 / (V1 * V1))
7330 IF TN < 0 THEN TN = TN + 3.14159
7340 IF BS < 0 THEN 7440
7350 K1 = AS : AS = BS : BS = K1 : TN = TN + 1.57079
7360 IF AS < 0 OR BS < 0 THEN 7460
7370 AX = SQR(AS) : BX = SQR(BS)
7380 RETURN
7390 EV(H) = 4 : RETURN
7400 REM SLINE FIT FOR APPARENT VEL
7410 W1 = 1 / (V1 * V1) : W2 = 1 / (V2 * V2) : W3 = 1 / (V3 * V3)
7420 WY = W1 * VY(1) + W2 * VY(2) + W3 * VY(3)
7430 WX = W1 * VX(1) + W2 * VX(2) + W3 * VX(3)
7440 XY = W1 * VX(1) * VY(1) + W2 * VX(2) * VY(2) + W3 * VX(3) * VY(3)
7450 XX = W1 * VX(1) * VX(1) + W2 * VX(2) * VX(2) + W3 * VX(3) * VX(3)
7460 SW = W1 + W2 + W3
7470 YI = SW * XY - WY * WX
7480 SP = (SW * XY - WY * WX) / YI : YI = (WY * XX - WX * XY) / YI
7490 RETURN
7500 REM SUB TO CAL APPARENT & TRUE VEL
7510 REM
7520 VX(1) = V1 * COS(A1) : VY(1) = V1 * SIN(A1)
7530 VX(2) = V2 * COS(A2) : VY(2) = V2 * SIN(A2)
7540 VX(3) = V3 * COS(A3) : VY(3) = V3 * SIN(A3)
7550 GOSUB 7480
7560 X = -YI / (SP + 1 / SP)
7570 Y = YI / (SP * SP + 1)
7580 VA = SQR(X * X + Y * Y)
7590 GOSUB 9000
7600 TA = TX - TN : IF TA = 0 THEN TA = .017
7610 SP = -1 / TAN(TA)
7620 YI = VA * SIN(TA) - SP * COS(TA)
7630 Y = BS / SQR(BS + SP * SP * AS)
7640 X = ABS(SP) * AS / SQR(BS + SP * SP * AS)
7650 IF ABS(SP - X * BS / (Y * AS)) < .0001 THEN 7990
7660 GOSUB 9000
7670 T1 = TX : X = -X : Y = -Y : GOSUB 9000
7680 IF ABS(T1 - TA) < 1.570796 THEN TT = T1
7690 IF ABS(TX - TA) < 1.570796 THEN TT = TX
7700 GOTO 8030
7710 X = -X : GOSUB 9000 : T1 = TX : X = ABS(X) : Y = -Y
7720 GOSUB 9000
7730 IF ABS(T1 - TA) < 1.570796 THEN TT = T1
7740 IF ABS(TX - TA) < 1.570796 THEN TT = TX
7750 X0 = X : Y0 = Y : X = YI / (Y0 / X0 - SP)
7760 Y = (YI * Y0 / X0) / (Y0 / X0 - SP)
7770 VC = X0 * X0 + Y0 * Y0
7780 RV = SQR(ABS(X * X + Y * Y) / (VC - 1))
7790 VT = VC / SQR(X * X + Y * Y) : VC = SQR(VC)
7800 TT = TT + TN : TA = TA + TN
7810 RETURN
7820 REM SUB FOR ARCTAN 0-360
7830 TX = 0
7840 IF X = 0.00 THEN 9060
7850 IF Y = 0.00 THEN 9090
7860 IF X > 0 AND Y > 0 THEN TX = ATN(Y / X)
7870 IF X > 0 AND Y < 0 THEN TX = 6.28 - ATN(-Y / X)
7880 IF X < 0 AND Y > 0 THEN TX = 3.1415 - ATN(Y / -X)
7890 IF X < 0 AND Y < 0 THEN TX = 3.1415 + ATN(-Y / -X)
7900 RETURN
7910 IF Y > 0.00 THEN TX = 1.5707963
7920 IF Y < 0.00 THEN TX = 4.71238
7930 RETURN
7940 IF X > 0.00 THEN TX = 0.00
7950 IF X < 0.00 THEN TX = 3.141592654
7960 RETURN

```

II.2 4-ANTENNAS

```

1  REM ! C88A00
10 DIM CC(8,6,11),AC(8,4,6),EV(8,4),AM(8,4),F(8,6),TD(8,6),VK(3),VY(3),C
    A(6),OA(12),TQ(8,4,3),FQ(8,4,3)
11 DIM SM(8,4)
20 DATA 3,4,4,1,2,4,8,2,1,4,12,3,2,3,9,6,1,3,13,7,1,2,14,11
30 OA(1) = 1.570796;OA(2) = 3.141593;OA(3) = 3.926991
60 OA(4) = OA(2);OA(5) = 4.712389;OA(6) = 5.497787
70 OA(7) = OA(5);OA(8) = 0;OA(9) = .785398
80 OA(10) = 0;OA(11) = OA(1);OA(12) = 2.356194
90 PRINT "STARTING FILE#"; INPUT MS
100 PRINT "ENDING FILE#"; INPUT ME
160 FOR MF = MS TO ME
161 DS = CHR$(4): REM CTRL-D
170 PRINT DS;"BLOADDATA";MF;"A88A00"; PRINT " BLOADDAT";MF;"A9600"
171 FOR H = 1 TO 8: FOR S = 1 TO 4:EV(H,S) = 0: NEXT S: NEXT H
180 Z = 0: FOR N = 35448 TO 35455:Z = Z * 10 + PEEK(N) - 176
181 NEXT N
200 S1 = 35328:S2 = 35456:S3 = 37376:S4 = 38400
210 FOR H = 1 TO 8
215 B1 = S1 + 256 * (H - 1):B2 = S2 + 256 * (H - 1):B3 = S3 + 128 * (H -
    1)
216 B4 = S4 + 128 * (H - 1)
220 A = 5: FOR IA = 0 TO 15 STEP 5
225 A = A - 1:L = 0: FOR IL = 0 TO 80 STEP 16
230 L = L + 1
235 AC(H,A,L) = 16777216 * PEEK(B1 + IL + IA) + 65536 * PEEK(B2 + IL +
    IA) + 256 * PEEK(B3 + IL + IA) + PEEK(B4 + IL + IA)
240 NEXT IL: NEXT IA: NEXT H
244 GOSUB 3000
245 FOR C = 1 TO 6: READ J1,J2,J3,J4
250 FOR H = 1 TO 8:B1 = S1 + 256 * (H - 1) + J3:B2 = S2 + 256 * (H - 1) +
    J3:B3 = S3 + 128 * (H - 1) + J3:L = 5
251 B4 = S4 + 128 * (H - 1) + J3
255 FOR IL = 0 TO 80 STEP 16
260 L = L + 1
265 CC(H,C,L) = 16777216 * PEEK(B1 + IL) + 65536 * PEEK(B2 + IL) + 25
    6 * PEEK(B3 + IL) + PEEK(B4 + IL)
270 CC(H,C,L) = (CC(H,C,L) / (134 - L) - SM(H,J1) * SM(H,J2)) / SQR(AC(
    H,J1,1) * AC(H,J2,1))
275 NEXT IL: NEXT H: FOR H = 1 TO 8:B1 = S1 + 256 * (H - 1) + J4:B2 = S2
    + 256 * (H - 1) + J4:B3 = S3 + 256 * (H - 1) + J4:L = 6
276 B4 = S4 + 128 * (H - 1) + J4
280 FOR IL = 16 TO 80 STEP 16:L = L - 1
285 CC(H,C,L) = 16777216 * PEEK(B1 + IL) + 65536 * PEEK(B2 + IL) + 25
    6 * PEEK(B3 + IL) + PEEK(B4 + IL)
295 CC(H,C,L) = (CC(H,C,L) / (122 + L) - SM(H,J1) * SM(H,J2)) / SQR(AC(
    H,J1,1) * AC(H,J2,1))
296 NEXT IL: NEXT H: NEXT C
300 RESTORE
310 FOR H = 1 TO 8: FOR A = 1 TO 4: FOR L = 2 TO 6
311 AC(H,A,L) = AC(H,A,L) / AC(H,A,1)
312 NEXT L:AC(H,A,1) = 1: NEXT A: NEXT H
480 FOR H = 1 TO 8
490 FOR L = 2 TO 6
500 S1 = AC(H,1,L) + AC(H,2,L) + AC(H,3,L)
510 S2 = AC(H,1,L) + AC(H,2,L) + AC(H,4,L)
520 S3 = AC(H,2,L) + AC(H,3,L) + AC(H,4,L)
530 S4 = AC(H,1,L) + AC(H,3,L) + AC(H,4,L)
540 AC(H,1,L) = S1 / 3.0
550 AC(H,2,L) = S2 / 3.0
560 AC(H,3,L) = S3 / 3.0
570 AC(H,4,L) = S4 / 3.0
580 NEXT L: NEXT H
590 FOR H = 1 TO 8: FOR S = 1 TO 4
600 K = 1:MX = 1: FOR L = 2 TO 6
610 IF MX < AC(H,S,L) THEN 650
620 MX = AC(H,S,L)
630 K = L
640 NEXT L
650 AM(H,S) = MX
660 IF K < 3 THEN EV(H,S) = 1
680 NEXT S: NEXT H
690 REM START OF CURVE FIT
695 CA(1) = .1:CA(2) = .2:CA(3) = .3
696 CA(4) = .1:CA(5) = .2:CA(6) = .1
700 FOR H = 1 TO 8: FOR C = 1 TO 6
705 IF EV(H,1) < > 0 THEN 920
710 MX = CC(H,C,1):K = 1
720 FOR L = 2 TO 11
730 IF MX > CC(H,C,L) THEN 755
740 MX = CC(H,C,L)
750 K = L
755 NEXT L
760 IF ABS(K - 6) > 3 THEN 920
770 CF = CC(H,C,K + 2) - 2 * CC(H,C,K) + CC(H,C,K - 2)
780 CG = CC(H,C,K + 2) - CC(H,C,K - 2)
790 CH = CC(H,C,K + 1) - CC(H,C,K - 1)
800 CK = CC(H,C,K + 1) - 2 * CC(H,C,K) + CC(H,C,K - 1)
810 P1 = (CF - 4 * CK) / 2.4
820 Q = (CG - 2 * CH) / 1.2
830 R = -(CF - 16 * CK) / 2.4
840 S = -(CG - 8 * CH) / 1.2
850 X = -.5 * S / R

```

```

860 FOR LQ = 1 TO 100
870 F1 = S + X * (R + R + X * (Q + Q + Q + 4 * P1 * X))
880 IF ABS (F1) < .0001 THEN 930
890 F2 = R + R + X * (S * Q + 12 * P1 * X)
900 X = X - F1 / F2
910 NEXT LQ
920 P(H,C) = - 2:TD(H,C) = 0: GOTO 950
930 P(H,C) = (((P1 * X + Q) * X + R) * X + S) * X / 10 +
940 TD(H,C) = (((K - 6) * X) * .4 + CA(C)
950 NEXT C: NEXT H
955 PRINT
1110 REM END OF CURVE FIT
1111 FOR H = 1 TO 6
1112 TO(H,1,1) = TD(H,6):FO(H,1,1) = F(H,6)
1113 TO(H,1,2) = TD(H,5):FO(H,1,2) = F(H,5)
1114 TO(H,1,3) = TD(H,4):FO(H,1,3) = F(H,4)
1115 TO(H,2,1) = TD(H,2):FO(H,2,1) = F(H,2)
1116 TO(H,2,2) = TD(H,6):FO(H,2,2) = F(H,6)
1117 TO(H,2,3) = TD(H,3):FO(H,2,3) = F(H,3)
1118 TO(H,3,1) = TD(H,1):FO(H,3,1) = F(H,1)
1119 TO(H,3,2) = TD(H,2):FO(H,3,2) = F(H,2)
1120 TO(H,3,3) = TD(H,4):FO(H,3,3) = F(H,4)
1121 TO(H,4,1) = TD(H,5):FO(H,4,1) = F(H,5)
1122 TO(H,4,2) = TD(H,1):FO(H,4,2) = F(H,1)
1123 TO(H,4,3) = TD(H,3):FO(H,4,3) = F(H,3)
1124 NEXT H
1190 PRINT "TIME=";Z
1200 FOR H = 1 TO 8
1210 FOR S = 1 TO 4
1220 IF EV(H,S) < > 0 THEN 1430
1230 PM = FQ(H,S,1)
1240 GOSUB 7000
1250 IF EV(H,S) < > 0 THEN 1430
1260 V1 = 169 / SQR (TQ(H,S,1) * TQ(H,S,1) + TT * TT)
1270 PM = FQ(H,S,2)
1280 GOSUB 7000
1290 IF EV(H,S) < > 0 THEN 1430
1300 V2 = 169 / SQR (TQ(H,S,2) * TQ(H,S,2) + TT * TT)
1310 PM = FQ(H,S,3)
1320 GOSUB 7000
1330 IF EV(H,S) < > 0 THEN 1430
1340 V3 = 239 / SQR (TQ(H,S,3) * TQ(H,S,3) + TT * TT)
1350 A3 = 3 * (S - 1)
1360 A1 = OA(A3 + 1):A2 = OA(A3 + 2):A3 = OA(A3 + 3)
1370 GOSUB 7300: IF EV(H,S) < > 0 THEN 1430
1375 PM = .5: GOSUB 7000: T5 = TT
1380 GOSUB 7300: IF EV(H,S) < > 0 THEN 1430
1390 V1 = 169 / TO(H,S,1)
1400 V2 = 169 / TO(H,S,2)
1410 V3 = 239 / TO(H,S,3)
1415 NT = ABS (TA(H,S,1) + TO(H,S,3) - TQ(H,S,2))
1416 NT = NT / (ABS (TQ(H,S,1)) + ABS (TQ(H,S,2)) + ABS (TQ(H,S,3)))
1420 GOSUB 7799
1430 PRINT H;S;" ";EV(H,S)
1431 IF EV(H,S) < > 0 THEN 1465
1432 PRINT INT (VT / 2)"; INT (TT * 57.296)";
1433 PRINT INT (AX)"; INT (BX)"; INT (T5 * 10)";
1434 PRINT INT (VC)"; INT (2A)"; INT (57.296 * TN)"; INT (10 * NT)";
1435 PRINT INT (10 * RV)
1465 NEXT S: NEXT H
1520 NEXT MF
2000 NEW : END
3000 B1 = 35572
3010 FOR H = 1 TO 8: FOR A = 1 TO 4
3020 I = (A - 1) * 3
3030 B2 = 65536 * PEEK (B1 + I) + 256 * PEEK (B1 + I + 1) + PEEK (B1 + I + 2)
3040 FOR L = 1 TO 6
3050 AC(H,A,L) = AC(H,A,L) / (129 - L) - (B2 * B2) / 16384.00
3060 NEXT L
3065 IF AC(H,A,1) < 0 THEN GOSUB 4000
3070 SM(H,A) = B2 / 128
3080 NEXT A: B1 = B1 + 256: NEXT H: RETURN
4000 EV(H,1) = 3:EV(H,2) = 3:EV(H,3) = 3:EV(H,4) = 3
4010 AC(H,A,1) = 70000
4020 RETURN
6999 REM SUB TO CAL PTS OF AUTOCORR
7000 IF PM > = AM(H,S) THEN 7030
7010 EV(H,S) = 10
7020 RETURN
7025 EV(H,S) = 11: RETURN
7030 AI = ABS (1 - PM)
7040 K = 1
7050 FOR L = 2 TO 6
7060 IF AI < ABS (PM - AC(H,S,L)) THEN 7090
7070 AI = ABS (PM - AC(H,S,L))
7080 K = L
7090 NEXT L: IF K > 5 THEN 7025
7100 IF K < > 1 THEN 7130
7110 TT = SQR (AI / (1 - AC(H,S,2))) * .4
7120 RETURN
7130 A = .5 * (AC(H,S,K - 1) - 2 * AC(H,S,K) + AC(H,S,K + 1))
7140 B = .25 * (AC(H,S,K + 1) - AC(H,S,K - 1))
7150 C = AC(H,S,K)
7160 IF ABS (A) > .0001 THEN 7190
7170 TT = (K - 1 + (PM - C) / (2 * B)) * .4
7180 RETURN
7190 GH = B * B - A * (C - PM): IF GH < 0.0 THEN 7170
7195 TT = (K - 1 - (B + SQR (GH)) / A) * .4
7200 RETURN

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

7300 REM SUB TO CAL ELLIPSE PARAM
7310 D1 = 1 / (V1 * V1) - 1 / (V2 * V2)
7320 D2 = 1 / (V1 * V1) - 1 / (V3 * V3)
7330 TN = D1 * (COS (2 * A1) - COS (2 * A3)) - D2 * (COS (2 * A1) - COS
(2 * A2))
7340 TN = SIN (2 * A1) - SIN (2 * A3)) - D2 * (SIN (2 * A1
) - SIN (2 * A2))
7350 TN = .5 * ATN ( - TN)
7360 K1 = .5 * (1 + COS (2 * (A1 - TN)))
7370 K2 = .5 * (1 - COS (2 * (A1 - TN)))
7380 K3 = .5 * (1 + COS (2 * (A2 - TN)))
7385 K4 = .5 * (1 - COS (2 * (A2 - TN)))
7390 AS = K3 * K2 - K1 * K4
7400 BS = AS / (K3 / (V1 * V1) - K1 / (V2 * V2))
7410 AS = AS / (K2 / (V2 * V2) - K4 / (V1 * V1))
7420 IF TN < 0 THEN TN = TN + 3.14159
7430 IF BS < AS THEN 7440
7435 K1 = AS:AS = BS:BS = K1:TN = TN + 1.57079
7440 IF AS < 0 OR BS < 0 THEN 7460
7445 AX = SQR (AS):BX = SQR (BS)
7450 RETURN
7460 EV(H,S) = 9: RETURN
7470 REM SLINE FIT FOR APPARENT VEL
7480 W1 = 1 / (V1 * V1):W2 = 1 / (V2 * V2):W3 = 1 / (V3 * V3)
7490 WY = W1 * VY(1) + W2 * VY(2) + W3 * VY(3)
7500 WX = W1 * VX(1) + W2 * VX(2) + W3 * VX(3)
7510 XY = W1 * VX(1) * VY(1) + W2 * VX(2) * VY(2) + W3 * VX(3) * VY(3)
7520 XK = W1 * VX(1) * VX(1) + W2 * VX(2) * VX(2) + W3 * VX(3) * VX(3)
7530 SW = W1 + W2 + W3
7540 YI = SW * XK - WX * WX
7550 SP = (SW * XY - WY * WX) / YI:YI = (WY * XK - WX * XY) / YI
7560 RETURN
7799 REM SUB TO CAL APPARENT & TRUE VEL
7800 REM
7820 VK(1) = V1 * COS (A1):VY(1) = V1 * SIN (A1)
7830 VK(2) = V2 * COS (A2):VY(2) = V2 * SIN (A2)
7840 VK(3) = V3 * COS (A3):VY(3) = V3 * SIN (A3)
7850 GOSUB 7480
7860 X = - YI / (SP + 1 / SP)
7870 Y = YI / (SP * SP + 1)
7880 VA = SQR (X * X + Y * Y)
7890 GOSUB 9000
7900 TA = TX - TN: IF TA = 0 THEN TA = .017
7905 SP = - 1 / TAN (TA)
7906 YI = VA * (SIN (TA) - SP * COS (TA))
7910 Y = BS / SQR (BS + SP * SP * AS)
7920 X = ABS (SP) * AS / SQR (BS + SP * SP * AS)
7930 IF ABS (SP - X * BS / (Y * AS)) < .0001 THEN 7990
7940 GOSUB 9000
7950 T1 = TX:X = - X:Y = - Y: GOSUB 9000
7960 IF ABS (T1 - TA) < 1.570796 THEN TT = T1
7970 IF ABS (TX - TA) < 1.570796 THEN TT = TX
7980 GOTO 8030
7990 X = - X: GOSUB 9000:T1 = TX:X = ABS (X):Y = - Y
8000 GOSUB 9000
8010 IF ABS (T1 - TA) < 1.570796 THEN TT = T1
8020 IF ABS (TX - TA) < 1.570796 THEN TT = TX
8030 XO = X:YO = Y:X = YI / (YO / XO - SP)
8040 Y = (YI * YO / XO) / (YO / XO - SP)
8045 VC = XO * XO + YO * YO
8046 RV = SQR ((X * X + Y * Y) / VC - 1)
8050 VT = VC / SQR (X * X + Y * Y):VC = SQR (VC)
8060 TT = TT + TN:TA = TA + TN
8080 RETURN
9000 REM SUB FOR ARCTAN 0-360
9004 TX = 0
9005 IF X = 0.00 THEN 9060
9006 IF Y = 0.00 THEN 9090
9010 IF X > 0 AND Y > 0 THEN TX = ATN (Y / X)
9020 IF X > 0 AND Y < 0 THEN TX = 6.28 - ATN ( - Y / X)
9030 IF X < 0 AND Y > 0 THEN TX = 3.1415 - ATN ( - Y / X)
9040 IF X < 0 AND Y < 0 THEN TX = 3.1415 + ATN (Y / X)
9050 RETURN
9060 IF Y > 0.00 THEN TX = 1.5707963
9070 IF Y < 0.00 THEN TX = 4.71238
9080 RETURN
9090 IF X > 0.00 THEN TX = 0.00
9100 IF X < 0.00 THEN TX = 3.141592654
9110 RETURN

```

APPENDIX III

CYBER PLOTTING PROGRAMS.

```

PROGRAM SC(INPUT,OUTPUT,CSPR,TAPK3=CSPR)
C   PROGRAM SC IS USED TO PRODUCE SCATTER
C   PLOTS THAT COMPARE C.S. WINDS DATA TO
C   F.R. WIND DATA.  REQUIRES A FILE CSFR
C   THAT CONTAINS THE WIND VALUES FOR
C   " MAXI " VALUES AT EACH ALTITUDE.  WILL
C   ASK FOR THE DATE : MONTH,DAY
C   EXTERNALS: ZETA / GRAB,ZETA
C   PLOT COMMAND: PLOTZ,TAPE26/LENGTH=70/TIME=24
C   WRITTEN BY RAYMOND RUGGERIO 1982
C   DIMENSION CS(7,20),FR(7,20),X(20),Y(20),HP(8)
C   DATA HP/70.5,75.,79.5,84.,88.5,93.,97.5,102./
C   PRINT 90
90  FORMAT("INPUT:MAXI,INTH,TDAY")
    READ *,MAXI,INTH,TDAY
    READ(3,40)((CS(I,J),FR(I,J),J=1,MAXI),I=1,7)
40  FORMAT(2F10,1)
    CALL PLOTS(0.,0.,26)
    CALL PLOT(1.0,3.0,-3)
    DO 150 IH=1,7
      DO 120 M=1,MAXI
        X(M)=FR(IH,M)
        Y(M)=CS(IH,M)
120  CONTINUE
        S1=0
        S2=0
        SS1=0
        SK1=0
        SS2=0
        DO 950 M2=1,MAXI
          S1=S1+FR(IH,M2)
          S2=S2+CS(IH,M2)
          SS1=SS1+FR(IH,M2)*FR(IH,M2)
          SS2=SS2+CS(IH,M2)*CS(IH,M2)
950  SK1=SK1+CS(IH,M2)*FR(IH,M2)
          S1=S1/FLOAT(MAXI)
          S2=S2/FLOAT(MAXI)
          SS1=SS1/FLOAT(MAXI)
          SS2=SS2/FLOAT(MAXI)
          SK1=SK1/FLOAT(MAXI)
          PHO=(SK1-S1*S2)/SQRT((S1-S1*S1)*(S2-S2*S2))
          PRINT *,IH,PHO
          CALL AXIS(0.,0., "FR VELOCITY",-11,4.,0.,-100.,50.)
          CALL AXIS(0.,0., "CS VELOCITY",11,4.,90.,-100.,50.)
          CALL SYMBOL(2.0,4.51,.105,"ALT=",0.,5)
          CALL NUMBER(2.42,4.51,.105,HP(IH),0.,1)
          CALL SYMBOL(2.84,4.51,.105," KM",0.,3)
          CALL SYMBOL(2.,4.72,.105,"DATE",0.,5)
          CALL NUMBER(3.1,4.72,.105,INTH,0.,-1)
          CALL SYMBOL(3.31,4.72,.105,"/",0.,1)
          CALL NUMBER(3.415,4.72,.105,TDAY,0.,-1)
          CALL SYMBOL(3.625,4.72,.105,"/82",0.,3)
          CALL SYMBOL(2.,4.3,.105,"PHO=",0.,4)
          CALL NUMBER(2.42,4.3,.105,PHO,0.,3)
          CALL PLOT(0.,0.,3)
          CALL PLOT(4.,4.,2)
          CALL PLOT(4.,4.,3)
          CALL PLOT(2.,2.,-3)
          DO 130 J=1,MAXI
130  CALL SYMBOL(X(J)/50,-.054,Y(J)/50,-.054,.11,"+",0.,1)
150  CALL PLOT(6.5,-2.,-3)
      CALL PLOT(0.,0.,999)
    STOP
  END

```


ORIGINAL PAGE IS
OF POOR QUALITY

76

```

200 CONTINUE
DO 500 IH=1,8
SI=0
CI=0
DO 520 M=1,MAXI
VCS(IH,M)=RMS(IH,M)*.58778-EW(IH,M)*.809
IF(VT(IH,M).EQ.0.)GO TO 520
CI=CI+1
SI=SI+VCS(IH,M)
520 CONTINUE
VCSM(IH)=SI/CI
500 CONTINUE
WRITE(4,540)TIME
540 FORMAT(1H1,"TIME=",I7)
WRITE(4,550)((VCS(IH,M),M=1,MAXI),IH=1,8)
550 FORMAT(6F20.1)
WRITE(4,555)(VCSM(J),J=1,8)
555 FORMAT(// " MEAN VALUES "/(F20.1))
CALL ADISP(T5,MAXI,TMTH,TDAY,THR,TMIN)
CALL PDISP(PA1,PA2,PA3,MAXI,TMTH,TDAY,THR,TMIN)
WRITE(3,450)
450 FORMAT(1H1,2X,"MIN,VT,TT,NS,EW,AR,TN,POWER,HALF TIME")
DO 300 IH=1,8
DO 310 M=1,MAXI
WRITE(3,410)M,VT(IH,M),TT(IH,M),RMS(IH,M),EW(IH,M)
+,AR(IH,M),TN(IH,M),PA1(IH,M),T5(IH,M)
410 FORMAT(2X,I6,8F15.4)
310 CONTINUE
WRITE(3,420)
420 FORMAT(1H1)
300 CONTINUE
CALL AVER(MAXI)
PRINT 501
501 FORMAT("INPUT VECTOR FOR DESIRED PLOTS")
PRINT 502
502 FORMAT(" A 1 MEANS PLOT, A 0 MEANS DO NOT PLOT")
READ *,(VPV(J),J=1,8)
DO 987 IU=1,8
IF(VPV(IU).EQ.1.0)CALL DISPW(MAXI,IU,TMTH,TDAY,THR,TMIN)
987 CONTINUE
CALL PLOT(0.,0.0,999)
GO TO 911
800 WRITE(3,430)
430 FORMAT(" SEQUENCE ERROR")
911 STOP
END
SUBROUTINE ADISP(T5,MAXI,TMTH,TDAY,THR,TMIN)
DIMENSION T5(8,240)
DO 10 J=1,8
DO 10 M=1,MAXI
T5(J,M)=T5(J,M)/100.
10 IF(T5(J,M).GT.2.0)T5(J,M)=0.0
CALL AXIS(0.,0.,15)TIME IN MINUTES,-15,6.,0.,0.,10.)
CALL AXIS(0.,0.,13)ALTITUDE (KM),13,7.,90.,70.5,4.5)
XSC=10.
YSC=1.5
CALL SYMBOL(3.,8.3,.105,"START TIME=",0.0,11)
CALL NUMBER(4.155,8.3,.105,THR,0.,-1)
CALL SYMBOL(4.365,8.3,.105,":",0.0,1)
CALL NUMBER(4.47,8.3,.105,TMIN,0.0,-1)
CALL SYMBOL(3.,8.51,.105,"DATE ",0.0,5)
CALL NUMBER(4.1,8.51,.105,TMTH,0.0,-1)
CALL SYMBOL(4.31,8.51,.105,"/",0.0,1)
CALL NUMBER(4.415,8.51,.105,TDAY,0.0,-1)
CALL SYMBOL(4.65,8.51,.105,"/82",0.0,3)
CALL PLOT(-.0625,8.,3)
CALL PLOT(.0625,8.,2)
CALL PLOT(0.,8.,2)
CALL PLOT(0.,9.,2)
CALL PLOT(-.0625,9.,2)
CALL PLOT(.0625,9.,2)
CALL PLOT(.0625,9.,3)
CALL SYMBOL(.25,8.51,.105,"CORRELATION TIME",0.,16)
CALL SYMBOL(.25,8.3,.105,"1.5 SEC",0.,7)
DO 200 IH=1,8

```

```

CALL PLOT(0.,0.,3)
CALL PLOT(6.,0.,2)
CALL PLOT(6.,0.,3)
CALL PLOT(1./XSC,T5(IH,1)/YSC,3)
DO 210 J=2,MAXI
210 IF(T5(IH,J).NE.0.)CALL PLOT(FLOAT(J)/XSC,T5(IH,J)/YSC,2)
CALL PLOT(FLOAT(MAXI)/XSC,T5(IH,MAXI)/YSC,3)
CALL PLOT(0.,1.,-3)
200 CONTINUE
CALL PLOT(8.5,-8.,-3)
RETURN
END
SUBROUTINE FDISP(PA1,PA2,PA3,MAXI,TMTH,TDAY,THR,TMIN)
DIMENSION PA1(8,240),PA2(8,240),PA3(8,240)
DO 10 IH=1,8
DO 20 J=1,MAXI
PA1(IH,J)=10*ALOG10(EXP(PA1(IH,J)/100.))
IF(PA1(IH,J).EQ.0.)GOTO 40
GO TO 20
40 IF(PA2(IH,J).NE.0)PA1(IH,J)=10*ALOG10(EXP(
+PA2(IH,J)/100.))
IF(PA1(IH,J).EQ.0.)PA1(IH,J)=10*ALOG10(EXP(
+PA3(IH,J)/100.))
IF(PA1(IH,J).EQ.0.)PA1(IH,J)=50.0
20 CONTINUE
10 CONTINUE
SPMIN=PA1(1,1)
SPMAX=PA1(1,1)
DO 50 IH=1,8
DO 50 J=1,MAXI
IF(PA1(IH,J).GT.SPMAX)SPMAX=PA1(IH,J)
50 IF(SPMIN.GT.PA1(IH,J))SPMIN=PA1(IH,J)
DO 60 IH=1,8
DO 60 J=1,MAXI
PA1(IH,J)=PA1(IH,J)-SPMIN
60 CONTINUE
CALL AXIS(0.,0.,15,TIME IN MINUTES,-15,6.,0.,0.,10.)
CALL AXIS(0.,0.,13,HALTITUDE (KM),13,7.,90.,70.5,4.5)
XSC=10.
YSC=FLOAT(IFIX(SPMAX))
CALL SYMBOL(3.,8.3,.105,"START TIME=",0.,11)
CALL NUMBER(4.155,8.3,.105,THR,0.,-1)
CALL SYMBOL(4.365,8.3,.105,"",0.0,1)
CALL NUMBER(4.47,8.3,.105,TMIN,0.,-1)
CALL SYMBOL(3.,8.51,.105,"DATE ",0.0,5)
CALL NUMBER(4.1,8.51,.105,TMTH,0.0,-1)
CALL SYMBOL(4.31,8.51,.105,"/",0.0,1)
CALL NUMBER(4.415,8.51,.105,TDAY,0.,-1)
CALL SYMBOL(4.65,8.51,.105,"/82",0.0,3)
CALL PLOT(-.0625,8.,3)
CALL PLOT(.0625,8.,2)
CALL PLOT(0.,8.,2)
CALL PLOT(0.,9.,2)
CALL PLOT(-.0625,9.,2)
CALL PLOT(.0626,9.,2)
CALL PLOT(.0626,9.,3)
CALL SYMBOL(.25,8.51,.105,"POWER",0.0,5)
CALL NUMBER(.25,8.3,.105,YSC,0.,-1)
CALL SYMBOL(.25,8.09,.105,"DB",0.,2)
DO 200 IH=1,8
CALL PLOT(0.,0.,3)
CALL PLOT(6.,0.,2)
CALL PLOT(6.,0.,3)
CALL PLOT(1./XSC,PA1(IH,1)/YSC,3)
DO 210 J=2,MAXI
210 CALL PLOT(FLOAT(J)/XSC,PA1(IH,J)/YSC,2)
CALL PLOT(FLOAT(MAXI)/XSC,PA1(IH,MAXI)/YSC,3)
CALL PLOT(0.,1.,-3)
200 CONTINUE
CALL PLOT(8.5,-8.0,-3)
RETURN
END
SUBROUTINE TRCV(TIME,TMTH,TDAY,THR,TMIN)
INTEGER TIME
TMTH=IFIX(FLOAT(TIME)/1000000.)

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

TDAY=IFIX(FLOAT(TIME)/10000.)*TMTH*100.
THR=IFIX(FLOAT(TIME)/100.)*TMTH*10000.-TDAY*100.
TMIN=FLOAT(TIME)-100*THR-10000*TDAY-1000000*TMTH
RETURN
END
SUBROUTINE DISPW2(X,Y,MAXI,WD,I,TMTH,TDAY,THR,TMIN)
DIMENSION X(240),Y(240),HP(8)
DATA HP/70.5,75.,79.5,84.,88.5,93.,97.5,102./
CALL AXIS(0.,0.,15*TIME IN MINUTES,-15,6.,0.,0.,10.)
IF(WD.EQ.0.)CALL AXIS(0.,0.,
+ "WORTH TO SOUTH VELOCITY (M/S)",29,8.,90.,-200.,50.)
IF(WD.EQ.1.)CALL AXIS(0.,0., "EAST TO WEST VELOCITY (M/S)",
+27,8.0,90.,-200.,50.)
XSC=10.
YSC=50.
CALL SYMBOL(3.,8.3,.105,"ALT=" ,0.0,5)
CALL NUMBER(3.42,8.3,.105,HP(I),0.,1)
CALL SYMBOL(3.84,8.3,.105," KM",0.,3)
CALL SYMBOL(3.,8.51,.105,"START TIME=",0.0,11)
CALL NUMBER(4.155,8.51,.105,THR,0.,-1)
CALL SYMBOL(4.365,8.51,.105,":",0.0,1)
CALL NUMBER(4.47,8.51,.105,TMIN,0.0,-1)
CALL SYMBOL(3.0,8.72,.105,"DATE " ,0.0,5)
CALL NUMBER(4.1,8.72,.105,TMTH,0.0,-1)
CALL SYMBOL(4.31,8.72,.105,"/",0.0,1)
CALL NUMBER(4.415,8.72,.105,TDAY,0.0,-1)
CALL SYMBOL(4.65,8.72,.105,"/82",0.0,3)
CALL PLOT(0.0,4.0,-3)
DO 10 J=1,MAXI
IF(X(J).EQ.0.0)GO TO 10
CALL SYMBOL(X(J)/XSC-.054,Y(J)/YSC-.054,.11,"+",0.0,1)
10 CONTINUE
CALL PLOT(8.5,-4.0,-3)
RETURN
END
SUBROUTINE AVER(MAXI)
DIMENSION VT(8,240),TT(8,240),AX(8,240),TN(8,240),
+RNS(8,240),EW(8,240),AR(8,240),AW(8,2),BX(8,240)
COMMON VT,TT,AX,BX,TN,RNS,EW,AR,AW
INTEGER H,SPF(8)
DO 10 H=1,8
S1=0
S2=0
C1=0.00
DO 20 M=1,MAXI
IF(VT(H,M).EQ.0)GO TO 20
C1=C1+1
S1=S1+RNS(H,M)
S2=S2+EW(H,M)
20 CONTINUE
SPF(H)=C1
AW(H,1)=S1/C1
10 AW(H,2)=S2/C1
WRITE(3,700)(AW(H,1),AW(H,2),SPF(H),H=1,8)
PRINT 700,(AW(H,1),AW(H,2),SPF(H),H=1,8)
700 FORMAT(1H1//2X," MEAN WIND VALUES NS AND EW "/(2F10.1,5X,I5))
RETURN
END
SUBROUTINE DISPW(MAXI,H,TMTH,TDAY,THR,TMIN)
DIMENSION X(240),Y(240),VT(8,240),TT(8,240),AX(8,240),
+BX(8,240),TN(8,240),RNS(8,240),EW(8,240),AR(8,240),AW(8,2)
COMMON VT,TT,AX,BX,TN,RNS,EW,AR,AW
INTEGER H
DO 10 J=1,MAXI
X(J)=J
10 Y(J)=RNS(H,J)
WD=0.0
CALL DISPW2(X,Y,MAXI,WD,H,TMTH,TDAY,THR,TMIN)
DO 20 J=1,MAXI
20 Y(J)=EW(H,J)
WD=1.0
CALL DISPW2(X,Y,MAXI,WD,H,TMTH,TDAY,THR,TMIN)
RETURN
END

```

APPENDIX IV
OPERATION MANUAL FOR PARTIAL-REFLECTION SYSTEM.

This is a step-by-step procedure that must be followed to operate the partial-reflection-drifts experiment.

- 1) Connect the pulser to the radar controller to obtain the 400 Hz control signal.
- 2) Turn on the power switches on the pulser, A/D, receiver, relay switching network, and all ac switches on the transmitter panel.
- 3) Unground the T/X-antenna and the R/X-antenna systems.
- 4) Slowly turn the filament voltage for the final stage amplifier tubes to 5.5 volts.
- 5) Turn the bias voltage to 28 volts.
- 6) Turn the switch located under the high voltage (5 kV) variac on, the bias voltage will jump to 35 volts.
- 7) Turn the variac clockwise until the high voltage is up to 2000 volts.
- 8) Press the low voltage reset on the 2 kV power supply.
- 9) Slowly turn up the 2 kV power supply up to 1100 volts while maintaining 2 kV on the high voltage power supply.
- 10) Turn the high voltage up to 3000 volts.
- 11) The T/X and R/X are now on the air and all that is required now is to bring up the computer system.

Computer System:

- 1) Place the following peripherals in the appropriate slots of the computer;

Corvus disk drive ----- Slot #6

Parallel interface ----- Slot #7

Arith proc ----- Slot #5

2) Sign on the system as user "PR".

3) Type "BRUNDRIFTCOL,S6,V49".

4) Type "MAINDRIFT".

5) The program will start putting data on volume 50 and continue to volume 67. File numbers will range from 0-19 for each volume.

6) Collection computer prompts the volume# and file# that corresponds to the last minute of data that was collected and stored.

To bring up the processing computer;

1) Sign on as user "PR".

2) Type "MAXFILES 1".

3) Type "BRUNBIN.DRIFTL2,S6,V49".

4) Enter information that is requested.

Example:

INPUT PARM FOR OUTPUT FILE

SLOT#4

VOLUME #254

INPUT PARM FOR DATA FILES

STARTING FILE# = 0

ENDING FILE# = 19

STARTING VOLUME # = 50

ENDING VOLUME# = 52

Program will display what file and volume it is working on.

5) The text output file generated will have the filename of "RESULTS".

6) The order of the results printed for each minute is:

LINE#1; Date and Time

*LINE#2; ALT#, Error vector, Power of antenna 1, Power antenna 2,
Power antenna 3, Autocorrelation halftime times 100.

LINE#3; True velocity, Direction of drift in degrees, Semimajor
axis "a", Axial ratio, VC', Rotation angle of ellipse, NTD, Random
variation times 10.

Once the data are stored on the floppy disk the results can be transferred to the University computer for plotting via a modem. Plotting routines for the computer are available for this.

*If the error vector is not equal to zero the next altitude is printed. The error vector indicates that the drift analysis failed to generate a true velocity.

REFERENCES

- Briggs, B. H. (1957), The determination of ionospheric drifts from three receiver tapping records, Instruction manual for ionospheric drift measurements.
- Briggs, B. H. (1968), On the analysis of moving patterns in geophysics - I., Correlation analysis, J. Atmos. Terr. Phys., 30, 1777-1788.
- Briggs, B. H. (1976), The physical significance of the correlation ellipse in ionospheric drift analysis, Radio Sci., 11, 817-819.
- Briggs, B. H. (1977), Ionospheric drifts, J. Atmos. Terr. Phys., 39, 1023-1033.
- Briggs, B. H. (1981), Radar observations of atmospheric winds and turbulence: a comparison of techniques, J. Atmos. Terr. Phys., 42, 823-833.
- Briggs, B. H., G. J. Phillips, and D. H. Shinn (1950), The analysis of observations on spaced receivers of the fading radio signals, Proc. of Physical Soc., B, LXIII, 106.
- Edwards, B. (1978), Research in Aeronomy, Progress Rep., 78-2, Aeron. Lab., Dep. Elec. Eng., Univ. of Ill., Urbana-Champaign.
- Fooks, G. F. (1965), Ionospheric drift measurements using correlation analysis methods of computation and interpretation of results, J. Atmos. Terr. Phys., 27, No. 9, 979.
- Fooks, G. F., and I. L. Jones (1961), Correlation analysis of the fading radio waves reflected vertically from the ionosphere, J. Atmos. Terr. Phys., 20, 229-242.
- Hagen, J. B., D. T. Farley (1973), Digital-correlation techniques in radio science, Radio Sci., 8, 775-784.

- Henry, G. W. Jr. (1966), Instrumentation and preliminary results from ship-board measurements of vertical incidence ionospheric absorption, Aeron. Rep. No. 13, Aeron. Lab., Dep. Elec. Eng., Univ. of Ill., Urbana-Champaign.
- Hines, C. O., and R. R. Rao (1968), Validity of three-station methods of determining ionospheric motions, J. Atmos. Terr. Phys., 30, 979-993.
- Mitra, S. N. (1949), A rapid method of measuring winds in the ionosphere, Proc. from Inst. of Elec. Engrs., III, 96, 441-446.
- Pirnat, C. R., and S. A. Bowhill (1968), Electron densities in the lower ionosphere deduced from partial-reflection measurements, Aeron. Rep. No. 29, Aeron. Lab., Dep. Elec. Eng., Univ. of Ill., Urbana-Champaign.
- Pitteway, M. L. V., J. W. Wright, and L. S. Fedor (1971), The interpretation of ionospheric radio drift measurements-III, Validation of correlation analysis by computer simulation, J. Atmos. Terr. Phys., 33, 635-660.
- Weiland, R. M., and S. A. Bowhill (1978), D-region differential-phase measurements and ionization variability studies, Aeron. Rep. No. 81, Aeron. Lab., Dep. Elec. Eng., Univ. of Ill., Urbana-Champaign.
- Weiland R. M., and S. A. Bowhill (1981), Investigation of the winds and electron concentration variability in the D region of the ionosphere by the partial-reflection radar technique. Aeron. Rep. No. 100, Aeron. Lab., Dep. Elec. Eng., Univ. of Ill., Urbana-Champaign.