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.

Produced by the NASA Center for Aerospace Information (CASI)
AERONOMY REPORT
NO. 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

Library of Congress ISSN 0568-0581
AERONOMY REPORT

No. 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.
# TABLE OF CONTENTS

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

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Arrangement of receiving antennas.</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>Auto- and cross correlation functions</td>
<td>7</td>
</tr>
<tr>
<td>2.3</td>
<td>Geometry of full-correlation analysis</td>
<td>10</td>
</tr>
<tr>
<td>3.1</td>
<td>Block diagram of the former partial-reflection system</td>
<td>12</td>
</tr>
<tr>
<td>3.2</td>
<td>Partial-reflection antenna arrays at the Aeronomy Laboratory Field Station.</td>
<td>13</td>
</tr>
<tr>
<td>3.3</td>
<td>Block diagram of the new partial-reflection system</td>
<td>16</td>
</tr>
<tr>
<td>3.4</td>
<td>Block diagram of new computer system</td>
<td>18</td>
</tr>
<tr>
<td>3.5</td>
<td>A/D interface circuit diagram</td>
<td>20</td>
</tr>
<tr>
<td>3.6</td>
<td>Timing diagram for new pulser</td>
<td>22</td>
</tr>
<tr>
<td>3.7</td>
<td>Circuit diagram for new pulser</td>
<td>23</td>
</tr>
<tr>
<td>4.1a</td>
<td>Flowchart for interrupt service routine (ISR) collection state.</td>
<td>26</td>
</tr>
<tr>
<td>4.1b</td>
<td>Flowchart for (ISR) wait state</td>
<td>27</td>
</tr>
<tr>
<td>4.1c</td>
<td>Flowchart for (ISR) stall state</td>
<td>28</td>
</tr>
<tr>
<td>4.2</td>
<td>Flowchart for main code DCORR.</td>
<td>30</td>
</tr>
<tr>
<td>4.3</td>
<td>Method used to calculate correlations functions</td>
<td>35</td>
</tr>
<tr>
<td>4.4a</td>
<td>Flowchart for 4 antenna postprocessing software</td>
<td>36</td>
</tr>
<tr>
<td>4.4b</td>
<td>Flowchart for computation of apparent and true velocity</td>
<td>39</td>
</tr>
<tr>
<td>4.4c</td>
<td>Flowchart for computation of ellipse</td>
<td>40</td>
</tr>
<tr>
<td>4.4d</td>
<td>Flowchart for curve fitting of cross correlations</td>
<td>41</td>
</tr>
<tr>
<td>5.1</td>
<td>Scatter plot at 75 km.</td>
<td>49</td>
</tr>
<tr>
<td>5.2</td>
<td>Scatter plot at 79.5 km.</td>
<td>50</td>
</tr>
<tr>
<td>5.3</td>
<td>Scatter plot at 84 km.</td>
<td>51</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.4</td>
<td>Scatter plot at 93 km.</td>
<td>52</td>
</tr>
<tr>
<td>5.5</td>
<td>Scatter plot at 97.5 km.</td>
<td>53</td>
</tr>
<tr>
<td>5.6</td>
<td>Drift velocity as a function of velocity perturbation factor</td>
<td>55</td>
</tr>
</tbody>
</table>
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:

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

for the autocorrelations, and

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

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-
Figure 2.1 Arrangement of receiving antennas.
The information that is extracted from the correlation functions 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:

\[
(t_{ij})^2 = (t'_{ij})^2 - (t'_{m})^2
\]

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:

\[
\text{MAG} |\mathbf{v}_{ij}| = d_{ij}/t_{ij}
\]

\[
\text{ANGLE} |\mathbf{v}_{ij}| = \theta_{ij}
\]
Figure 2.2 (a) Autocorrelation function.
(b) Cross correlation function.
are computed. The $d_{ij}$'s are the baseline distances in meters and the $\theta_{ij}$'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 $\theta_0$ 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 $\theta_0$ 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 $\phi$ 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;

$$V_{drift} = \frac{(V'_{c})_v^2}{V'}$$

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:

\[ V_E = V_{\text{drift}} \sin \phi \quad V_N = V_{\text{drift}} \cos \phi \]

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.
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:

<table>
<thead>
<tr>
<th>Character</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peak Power</td>
<td>35 kW</td>
</tr>
<tr>
<td>Frequency</td>
<td>2.66 MHz</td>
</tr>
<tr>
<td>Pulse Width</td>
<td>23 μsec</td>
</tr>
<tr>
<td>Output Impedance</td>
<td>50 ohm, unbalanced</td>
</tr>
<tr>
<td>Pulse Repetition Rate</td>
<td>10 Hz</td>
</tr>
</tbody>
</table>

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.
Figure 3.1 Block diagram of the former partial-reflection system.
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° phase shift, the ordinary or extra-ordinary 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:

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Center Frequency</td>
<td>2.66 MHz</td>
</tr>
<tr>
<td>Noise Figure</td>
<td>3 dB</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>40 kHz</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>200 μsec after removal of .1V</td>
</tr>
<tr>
<td></td>
<td>RMS input</td>
</tr>
<tr>
<td>Gain Variation</td>
<td>3 dB max</td>
</tr>
<tr>
<td>Rf Input Impedance</td>
<td>50 ohm, unbalanced</td>
</tr>
<tr>
<td>Output Impedance</td>
<td>10,000 ohms</td>
</tr>
<tr>
<td>Linearity</td>
<td>55 dB for 1 dB deviation</td>
</tr>
</tbody>
</table>

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-
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
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.
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 computer's 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.

<table>
<thead>
<tr>
<th>QUADRANT</th>
<th>VALUE IN HEX TO SELECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>NE</td>
<td>OE</td>
</tr>
<tr>
<td>NW</td>
<td>OD</td>
</tr>
<tr>
<td>SW</td>
<td>OB</td>
</tr>
<tr>
<td>SE</td>
<td>07</td>
</tr>
</tbody>
</table>

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
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
Figure 4.2 Flowchart for main code DCORR.
Figure 4.2 (cont.)
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 $T_1$ 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
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.)

- REORDER RESULTS FROM CURVE-FITTING
- OPEN TEXT FILE FOR OUTPUT. WRITE TIME
- INITIALIZE LOOPS TO PROCESS ALL HTS (H) & SETS (S)
- IS EV FOR H, S #0
  - YES
  - NO
- COMPUTE VECTORS THAT DEFINE ELLIPSE
- *CALCULATE ELLIPSE PARAMETERS
- COMPUTE APPARENT VELOCITY VECTORS V1 A1, V2 A2, V3 A3

Original page is of poor quality.
Figure 4.4a (cont.)
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.
The 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 $y_o$ the required differences are

$$
\Delta y_{-1} = y_o - y_{-1}
\Delta y_o = y_1 - y_o
\Delta^2 y_{-1} = y_1 - 2y_o + y_{-1}
\Delta^3 y_{-2} = y_1 - 3y_o + 3y_{-1} - y_{-2}
\Delta^3 y_{-1} = y_2 - 3y_o - y_{-1}
\Delta^4 y_{-2} = y_2 - 4y_1 + 6y_o - 4y_{-1} + y_{-2}
$$
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:

\[ a_0 = y_0 \quad a_3 = \frac{1}{12}(\Delta y_{-2} + \Delta y_{-1}) \]

\[ a_1 = \frac{1}{2}(\Delta y_{-1} + \Delta y_0) - \frac{1}{12}(\Delta^3 y_{-2} + \Delta^3 y_{-1}) \quad a_4 = \frac{1}{24}\Delta^4 y_{-2} \]

\[ a_2 = \frac{1}{2}(\Delta^2 y_{-1}) - \frac{1}{24}\Delta^4 y_{-2} \]

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

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

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

\[ \frac{d^2y}{dx^2} = a_1 + 2a_2 + 6a_3x + 12a_4x^2 \]

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

\[ TD = [(KM - 6) + x] \times 4 + CA \]

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 drift analysis (Weiland 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 ±4th 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

\[
y_{\text{max}} = y_0 - \left( \frac{y_{-1} - y_{+1}}{2} \right) x + \left( \frac{y_{-1} - 2y_0 + y_{+1}}{2} \right) x^2
\]

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; \( (V_{c_1'})_{12}, (V_{c_1'})_{13}, (V_{c_1'})_{23} \). The general form of an ellipse in polar coordinates that has been rotated through some angle \( \theta_0 \) 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 \( (r_1, \theta_1), (r_2, \theta_2), (r_3, \theta_3) \) 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 \( \theta_0 \) 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:

\[
a^2 = \frac{(K_3K_2 - K_1K_4)}{(K_2/r_2^2 - K_4/r_1^2)}
\]

\[
b^2 = \frac{(K_3K_2 - K_1K_4)}{(K_3/r_3^2 - K_1/r_2^2)}
\]

where

\[
K_1 = \frac{1 + \cos 2(\theta_1 - \theta_0)}{2}
\]

\[
K_2 = \frac{1 - \cos 2(\theta_1 - \theta_0)}{2}
\]

\[
K_3 = \frac{1 + \cos 2(\theta_2 - \theta_0)}{2}
\]

\[
K_4 = \frac{1 - \cos 2(\theta_2 - \theta_0)}{2}
\]

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 \( W_i \)'s is \( (r_i)^2 \). 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

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

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

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:

\[
X = \frac{-m}{m + 1}
\]

\[
Y = \frac{a}{m^2 + 1}
\]

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

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

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 $x_o$ and $y_o$ 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 $(x_o, y_o)$ 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

$$y' = V_a \left[ \sin(\phi_a - \theta_o) - m' \cos(\phi_a - \theta_o) \right]$$

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

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

$$y = \frac{m_t y'}{m_t - m'}$$

The magnitude of the true velocity is then

$$|v_{\text{drift}}| = \frac{(v_{c'}^2)}{v'} = \frac{(X_o^2 + Y_o^2)}{\sqrt{X^2 + Y^2}}$$

The direction of drift, $\theta_t$, must be modified by $\theta_o$ because the value calculated was in the rotated coordinate system. Therefore $\theta_t = \theta_t' + \theta_o$. 
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
DATE: 4/24/82
ALT = 75 km
PHO = 0.079
CONF LEVEL = 59.79%

Figure 5.1 Scatter plot at 75 km.
DATE: 4/24/82
ALT = 79.5 km
PHO = 0.540
CONF LEVEL = 96.3%

Figure 5.2 Scatter plot at 79.5 km.
Figure 5.3 Scatter plot at 84 km.
Figure 5.4 Scatter plot at 93 km.
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 agree-
ment.

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 correla-
tions 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 accept-
able 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.
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.

FORGET DRIFT

COLLECTION SOFTWARE FOR PARTIAL REFLECTION DRIFTS EXPERIMENT
COMPUTES AUTO AND CROSS CORRELATIONS FOR THE FOUR ANTENNA SYSTEM USING A HARDWARE MULTIPLIER

LIMTS:
1. EIGHT ALTITUDES MAXIMUM
2. SIX LAGS INCLUDING ZERO LAG
3. PLUS AND MINUS 5 LAGS FOR CROSS CORRELATIONS
4. MAXIMUM FRF WITH ABOVE LIMITS IS 200 Hz

WRITTEN BY RAYMOND L. RUGGERIO 1982

0000 TIME VARIABLES
0000 VARIABLE VART
0000 VARIABLE BAG
0000 VARIABLE PADDR2
0000 VARIABLE PADDR

INIT/0

R7 C782 C1 (SET DDRB FOR OUTPUT)
C0 C78B C1 (SET ACC)
00 C78E C5 C1 (DISABLE INTERRUPTS ON VIA)
03 C785 C1 (TIMER1 SET FOR 10KHz)
03 C705 C1 (PCR SET FOR POS DEP INTERRUPTS)
03 C703 C1 (PCR SET TO LATCH PORTS)
03 C703 C1 (DDRA SET FOR INPUT)
00 C703 C1 (CLEAR IER)

CODE DCCOR (MAIN CODE FOR COLLECTION)

00 C703 C1 (SAVE ZERO PAGE AND REPLACE)
00 C703 C1 (WITH ZERO PAGE NEEDED FOR PROGRAM)
00 C703 C1 (CLEAR MEMORY FOR LAG TABLE COMPUTATIONS)

00 C703 C1 (CLEAR ADDITIONAL MEMORY FOR LSB OF LAG TABLE)
00 C703 C1 (READ CLOCK AND WAIT FOR MINUTE TO FLIP)
(JUMP TO SUBROUTINE TO READ CLOCK)

(CONTINUE TO READ CLOCK UNTIL MINUTE FLIPS)

(READ TIME INTO MEMORY LOCATIONS 4078-407F)

(DISPLAY HRS AND MINUTES ON SCREEN)

(SET INTEGRATION TABLE POINTERS)

(FOR INCOMING DATA)

(THese ARE POINTERS FOR THE ISR)

(ENABLE INTERRUPTS ON VIA)

(CLEAR INTERRUPT STATUS FLAG)

(WAIT UNTIL SAMPLE READY FOR PROCESSING)

(CLEAR FLAGS FOR SAMPLE READY AND END OF MINUTE)

(SET LAG TABLE POINTERS FOR COMPUTATION)

(FIND ZERO LAG VALUES)

(SET COMPUTATION POINTERS)

(SET HTS=8)

(SET LAGS DONE = 0)

(LOAD LAG '0' VALUES INTO ZERO PAGE)

( FOR COMPUTATION)

(N-W ANTENNA QUAD 1)

(N-E ANTENNA QUAD 2)

(S-W ANTENNA QUAD 3)

(S-E ANTENNA QUAD 4)

(RESET Y REG FOR LAG TABLE INDEX)

(RESET OPERATION POINTER)

(RESET OPERATION POINTER POINT TO ANTENNA FROM LAG 'N')

(CALCULATE ADDRESS FOR NEXT LAG VARIABLE)
ORIGINAl PAGE IS OF POOR QUALITY

LOAD VALUE FROM LAG ZERO INTO ARITH PROCESSOR CARD

LOAD VALUE FROM LAG ZERO INTO ARITH PROCESSOR CARD

LOAD CMD INTO PROCESSOR FOR 32-BIT MULT

ADD TO APPROPRIATE MEMORY LOCATION

IN LAG TABLE

GO AFTER NEXT ANTENA TO BE PROCESSED FROM ZERO LAG

GO AFTER NEXT ANTENA FROM LAG 'N'

GO AFTER NEXT LAG

IF ALL LAGS AND OPERATIONS DONE INC

POINTERS AND GO FOR NEXT HT
A3 STA;  
04 LDA Y;  
A1 STA;  
A1 Y LDA;  
A1 STA;  
96 STA;  
93 Y LDA;  
93 STA;  
91 STA;  
A3 Y LDA;  
A3 STA;  
60 Y STA;  
00 DEC, {RESET Y REG.}  
IPNE PADD12 () JMP, {IF ALL HTS NOT DONE DO NEXT ONE}  
THEN,  
C05B STA,  
BEGIN LDA; {WAIT FOR NEXT SAMPLE}  
01 CMP;  
ENDNE;  
C052 STA;  
00 LDA; {READY FOR NEXT SAMPLE CLEAR FLAG}  
ENDIF STA,  
0F LDA; {CHECK TO SEE IF ALL SAMPLE TAKEN}  
IPNE PADD () JMP, {IF NOT RETURN TO PROCESSING}  
THEN,  
09 STA;  
CBE STA,  
C70F STA;  
02 LDA;  
F1 STA;  
F0 STA;  
A1 STA;  
BEGIN LDY,  
BEGIN,  
4E64 X LDA,  
P0 X STA;  
P0 STA,  
P321 X STA;  
P0 STA,  
P0 LDA,  
6E00 STA;  
6E00 X LDA,  
6E32 X STA,  
6E64 X STA,  
6E64 STA,  
6E64 LDA,  
6E64 STA,  
ENDPL;  
F1 STA;  
ENDNE;  
60 LDA; {CLEAR FLAGS}  
BD STA;  
BE STA,  
BEGIN X LDA,  
B7 3E60 X STA,  
3E50 X STA,  
3E50 LDA;  
0E X STA;  
0E STA,  
0E LDX,  
BEGIN,  
ENDNE,  
ENDCODE; {END OF CODE DCCOR}  
178 WAITISR IS PART OF THE ISR. IT IS USED TO SWITCH THE RELAYS  
179 BACK TO SAMPLE ALL THE QUADRANTS. IF ALL THE ANTENNAS  
180 HAVE BEEN SAMPLED THEN IT MODIFIES POINTERS AND SETS FLAGS TO  
181 CONTINUE THE PROCESSING IN DCCOR. IT ALSO SUMS UP THE VALUES  
182 IN THE LAST SAMPLE TO OBTAIN MEAN VALUES OF THE SIGNALS IN  
183 EACH QUADRANT AND AT EACH ST.  
201 CODE WAITISR (START OF CODE FOR WAITISR)  
350 01 LDA,  
264 STA; {SET STALL FLAG}  
380 BC A9C, {INC ANTENNA COUNTER}  
04 CMP,
IF NE, BC, STA;
( COMPARE FOR ALL ANTENNAS SAMPLED ? )
IF NOT JUST SWITCH ANTENNA )
CLEAR INTEGRATION COUNTER
INC STALL COUNTER FOR RELAY SWITCHING
RETURN FROM INTERRUPT

ELSE, BY
B6, PLA;
{ IF ALL ANTENNAS SAMPLED THEN ROTATE POINTERS 
THESE POINTERS ARE LOOK-UP VECTORS FOR POINTING }
{ THE NEXT SAMPLE

A3, STA;
( FIX POINTERS FOR ZERO LAG )
B8, X STA;
( SWITCH RELAY TO QUAD1 FOR INPUT )
A0, LDA,
BEGIN,
B7, LDA;
SOM VALUES FOR THIS SAMPLE TO COMPUTE
BEGIN, A3, X Y
ADC, 4654, X Y STA;
ADD, X Y, LDA;
A5, X Y, STA;
IPCS, Y STA;
D0, + LDA;
ENDPL, Y
D0, Y STA,
( CLEAR NEXT LAG AREA FOR INCOMING VALUES )
BEGIN, AD, Y STA,
( SET FLAG FOR NEXT SAMPLE READY )
ENDPL, Y
A0, LDA;
01, STA;
INC, CMP;
IF NE, INC, CMP;
IFNE, PLA;
RETURN FROM INTERRUPT
ELSE, D0, LDA;
( IF STALL COUNT = MAX COUNT CLEAR FLAGS )
RETURN FROM INTERRUPT
ELSE, LDA;
( IF ALL SAMPLES TAKEN CLEAR NEXT LAG AREA FOR INCOMING VALUES )
RETURN FROM INTERRUPT
}
ORIGINAL PAGE IS OF POOR QUALITY
### Memory Usage for Collection Software

<table>
<thead>
<tr>
<th>ADDRESS IN HEX</th>
<th>ABBREVIATED SYMBOL</th>
<th>DEFINITION OF VARIABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>80-81</td>
<td>—</td>
<td>Temp. storage for computation of mult.</td>
</tr>
<tr>
<td>86-87</td>
<td>RLLL</td>
<td>Lag table point to LSB</td>
</tr>
<tr>
<td>88-89</td>
<td>RLL</td>
<td>Lag table pointer to low byte</td>
</tr>
<tr>
<td>8A-8B</td>
<td>RLM</td>
<td>Lag table pointer to med byte</td>
</tr>
<tr>
<td>8C-8D</td>
<td>RLH</td>
<td>Lag table pointer to high byte</td>
</tr>
<tr>
<td>8E-8F</td>
<td>PA</td>
<td>Contains address for input port</td>
</tr>
<tr>
<td>90-99</td>
<td>—</td>
<td>Temp. storage for lag zero values during mult.</td>
</tr>
<tr>
<td>A6-A1</td>
<td>TEMP</td>
<td>Temporary storage byte</td>
</tr>
<tr>
<td>A2</td>
<td>HTS</td>
<td>Height counter used for processing</td>
</tr>
<tr>
<td>A3-A4</td>
<td>LOLB</td>
<td>Lag 0 low byte pointer</td>
</tr>
<tr>
<td>A5-A6</td>
<td>LOAB</td>
<td>Lag 0 high byte pointer</td>
</tr>
<tr>
<td>A7</td>
<td>LAG</td>
<td>Lag counter for processing</td>
</tr>
<tr>
<td>AA</td>
<td>CLCT</td>
<td>Integration counter</td>
</tr>
<tr>
<td>AB-AC</td>
<td>LCIHB</td>
<td>Incoming integration counter high byte</td>
</tr>
<tr>
<td>AD-AE</td>
<td>LCIILB</td>
<td>Incoming integration counter low byte</td>
</tr>
<tr>
<td>B1-B7</td>
<td>NLAG</td>
<td>Offset pointers for computation</td>
</tr>
<tr>
<td>N8-BB</td>
<td>ASW</td>
<td>Antenna switching values</td>
</tr>
<tr>
<td>BC</td>
<td>ANT</td>
<td>Antenna counter (index to ASW)</td>
</tr>
<tr>
<td>BD</td>
<td>FFC</td>
<td>Counter for samples collected vs. EOR</td>
</tr>
<tr>
<td>BE</td>
<td>DFLAG</td>
<td>End of record flag (EOR)</td>
</tr>
<tr>
<td>C0</td>
<td>SCT</td>
<td>Stall counter for ISR</td>
</tr>
<tr>
<td>C1</td>
<td>DFG</td>
<td>Done flag for sample ready</td>
</tr>
<tr>
<td>C2-C9</td>
<td>NLHT</td>
<td>Computation variable</td>
</tr>
<tr>
<td>-------</td>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>EO-E1</td>
<td>TEMP</td>
<td>Temp. storage bytes</td>
</tr>
<tr>
<td>E4</td>
<td>STALL</td>
<td>Stall flag byte</td>
</tr>
</tbody>
</table>
### I.2 Memory Map

#### LAG Table

<table>
<thead>
<tr>
<th>Address</th>
<th>Description</th>
<th>High Bytes</th>
<th>Low Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>7300</td>
<td></td>
<td>HTS 7 &amp; 8</td>
<td></td>
</tr>
<tr>
<td>7200</td>
<td></td>
<td>HTS 5 &amp; 6</td>
<td></td>
</tr>
<tr>
<td>7100</td>
<td></td>
<td>HTS 3 &amp; 4</td>
<td></td>
</tr>
<tr>
<td>7000</td>
<td>LSB's</td>
<td>HTS 1 &amp; 2</td>
<td></td>
</tr>
</tbody>
</table>

#### LAG Table

<table>
<thead>
<tr>
<th>Address</th>
<th>Description</th>
<th>High Bytes</th>
<th>Low Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>4E00</td>
<td>Incoming mean values</td>
<td>HTS 7 &amp; 8</td>
<td></td>
</tr>
<tr>
<td>4D00</td>
<td>High bytes of last 6 lags, Hts 1-8</td>
<td>HTS 5 &amp; 6</td>
<td></td>
</tr>
<tr>
<td>4C00</td>
<td>Low bytes of last 6 lags, Hts 1-8</td>
<td>HTS 3 &amp; 4</td>
<td></td>
</tr>
<tr>
<td>4B00</td>
<td></td>
<td>HTS 1 &amp; 2</td>
<td></td>
</tr>
<tr>
<td>4A00</td>
<td></td>
<td>HT #8</td>
<td></td>
</tr>
<tr>
<td>4900</td>
<td></td>
<td>HT #7</td>
<td></td>
</tr>
<tr>
<td>4800</td>
<td>Low Bytes</td>
<td>HT #6</td>
<td></td>
</tr>
<tr>
<td>4700</td>
<td></td>
<td>HT #5</td>
<td></td>
</tr>
<tr>
<td>4600</td>
<td></td>
<td>HT #4</td>
<td></td>
</tr>
<tr>
<td>4500</td>
<td></td>
<td>HT #3</td>
<td></td>
</tr>
<tr>
<td>4400</td>
<td></td>
<td>HT #2</td>
<td></td>
</tr>
<tr>
<td>4300</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4200</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4000</td>
<td>High Bytes</td>
<td>Time</td>
<td>Med Bytes</td>
</tr>
</tbody>
</table>

**Note:** The original page is of poor quality.
II.1 3-ANTENNAS.

10 REM I.CS8A00
11 DIM (H,1,3); AC(0,3,6), EV(8), AM(8), P(8,3), TD(8,3), VX(3), VY(3), CA(3)
20 DATA 4,0,0,123,10,3,84,87
30 OR(I) = 1.0,0,2,1.0,41,96,12,3
40 CR(I) = 0.8,51.1,0.01,0.1,1.1,1,0
50 PRINT "DATA" OR(I) CR(I)
60 PRINT "PRINT 1-INPUT PARAM FOR OUTPUT FILE"
70 PRINT "WRITE" "OPEN INPUT FILE" "INPUT SR"
80 PRINT "WRITE" "OPEN INPUT FILE" "INPUT VS"
90 PRINT "WRITE" "OPEN INPUT FILE" "INPUT ME"
100 PRINT "WRITE" "OPEN INPUT FILE" "INPUT VS"
110 FOR I = 1 TO 8:EV(I) = 0: NEXT I
120 FOR I = 1 TO 8:EV(I) = 0: NEXT I
130 FOR I = 1 TO 8:EV(I) = 0: NEXT I
140 FOR I = 1 TO 8:EV(I) = 0: NEXT I
150 FOR I = 1 TO 8:EV(I) = 0: NEXT I
160 FOR I = 1 TO 8:EV(I) = 0: NEXT I
170 FOR I = 1 TO 8:EV(I) = 0: NEXT I
180 FOR I = 1 TO 8:EV(I) = 0: NEXT I
190 FOR I = 1 TO 8:EV(I) = 0: NEXT I
200 FOR I = 1 TO 8:EV(I) = 0: NEXT I
210 FOR I = 1 TO 8:EV(I) = 0: NEXT I
220 FOR I = 1 TO 8:EV(I) = 0: NEXT I
230 FOR I = 1 TO 8:EV(I) = 0: NEXT I
240 FOR I = 1 TO 8:EV(I) = 0: NEXT I
250 FOR I = 1 TO 8:EV(I) = 0: NEXT I
260 FOR I = 1 TO 8:EV(I) = 0: NEXT I
270 FOR I = 1 TO 8:EV(I) = 0: NEXT I
280 FOR I = 1 TO 8:EV(I) = 0: NEXT I
290 FOR I = 1 TO 8:EV(I) = 0: NEXT I
300 FOR I = 1 TO 8:EV(I) = 0: NEXT I
310 FOR I = 1 TO 8:EV(I) = 0: NEXT I
320 FOR I = 1 TO 8:EV(I) = 0: NEXT I
330 FOR I = 1 TO 8:EV(I) = 0: NEXT I
340 FOR I = 1 TO 8:EV(I) = 0: NEXT I
350 FOR I = 1 TO 8:EV(I) = 0: NEXT I
360 FOR I = 1 TO 8:EV(I) = 0: NEXT I
370 FOR I = 1 TO 8:EV(I) = 0: NEXT I
380 FOR I = 1 TO 8:EV(I) = 0: NEXT I
390 FOR I = 1 TO 8:EV(I) = 0: NEXT I
400 FOR I = 1 TO 8:EV(I) = 0: NEXT I
410 FOR I = 1 TO 8:EV(I) = 0: NEXT I
420 FOR I = 1 TO 8:EV(I) = 0: NEXT I
430 FOR I = 1 TO 8:EV(I) = 0: NEXT I
440 FOR I = 1 TO 8:EV(I) = 0: NEXT I
450 FOR I = 1 TO 8:EV(I) = 0: NEXT I
460 FOR I = 1 TO 8:EV(I) = 0: NEXT I
470 FOR I = 1 TO 8:EV(I) = 0: NEXT I
480 FOR I = 1 TO 8:EV(I) = 0: NEXT I
490 FOR I = 1 TO 8:EV(I) = 0: NEXT I
500 FOR I = 1 TO 8:EV(I) = 0: NEXT I
510 FOR I = 1 TO 8:EV(I) = 0: NEXT I
520 FOR I = 1 TO 8:EV(I) = 0: NEXT I
530 FOR I = 1 TO 8:EV(I) = 0: NEXT I
540 FOR I = 1 TO 8:EV(I) = 0: NEXT I
550 FOR I = 1 TO 8:EV(I) = 0: NEXT I
560 FOR I = 1 TO 8:EV(I) = 0: NEXT I
570 FOR I = 1 TO 8:EV(I) = 0: NEXT I
580 FOR I = 1 TO 8:EV(I) = 0: NEXT I
590 FOR I = 1 TO 8:EV(I) = 0: NEXT I
600 FOR I = 1 TO 8:EV(I) = 0: NEXT I
610 FOR I = 1 TO 8:EV(I) = 0: NEXT I
620 FOR I = 1 TO 8:EV(I) = 0: NEXT I
630 FOR I = 1 TO 8:EV(I) = 0: NEXT I
640 FOR I = 1 TO 8:EV(I) = 0: NEXT I
650 FOR I = 1 TO 8:EV(I) = 0: NEXT I
660 FOR I = 1 TO 8:EV(I) = 0: NEXT I
670 FOR I = 1 TO 8:EV(I) = 0: NEXT I
680 FOR I = 1 TO 8:EV(I) = 0: NEXT I
690 FOR I = 1 TO 8:EV(I) = 0: NEXT I
700 FOR I = 1 TO 8:EV(I) = 0: NEXT I
710 FOR I = 1 TO 8:EV(I) = 0: NEXT I
720 FOR I = 1 TO 8:EV(I) = 0: NEXT I

POSTPROCESSING PROGRAMS.
\[
CC(H, C, K) + CC(H, C, K + 1) - CC(H, C, K - 1) \approx \frac{CC(H, C, K) + CC(H, C, K + 1)}{2} - \frac{CC(H, C, K - 1)}{2}
\]
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
RETURN
II. 2 4-ANTENNAS

II. 2 4-ANTENNAS

100 REM "INPUTS:" 1: INPUT MS
110 REM "ENDING FILE:" 1: INPUT ME
120 IF MS = "YES" THEN 8: IF ME = "NO" THEN C:
130 PRINT "BROADSAY"; INPUT "A99800"; PRINT "BROADSAY" ; INPUT "A99600"
140 IF H = 1 TO 8: FOR S = 1 TO 4: EV(H,B) = 0: NEXT S: NEXT H
150 NEXT H
160 FOR H = 1 TO 8
170 B1 = 33528: R2 = 35456: 33 = 38400
180 FOR H = 1 TO 8
190 B3 = 81 + 256 * (H - 1): B2 = 82 + 256 * (H - 1) + 1: B1 = 83 + 128 * (H - 1)
200 NEXT H
210 NEXT H
220 IF H = 1 TO 8: FOR S = 1 TO 4: B = EV(H,B) + S: NEXT S: NEXT H
230 NEXT H
240 NEXT H
250 IF H = 1 TO 8: FOR S = 1 TO 4: B = EV(H,B) + S: NEXT S: NEXT H
260 NEXT H
270 NEXT H
280 NEXT H
290 NEXT H
300 NEXT H
310 NEXT H
320 NEXT H
330 NEXT H
340 NEXT H
350 NEXT H
360 NEXT H
370 NEXT H
380 NEXT H
390 NEXT H
400 NEXT H
410 NEXT H
420 NEXT H
430 NEXT H
440 NEXT H
450 NEXT H
460 NEXT H
470 NEXT H
480 NEXT H
490 NEXT H
500 NEXT H
510 NEXT H
520 NEXT H
530 NEXT H
540 NEXT H
550 NEXT H
560 NEXT H
570 NEXT H
580 NEXT H
590 NEXT H
600 NEXT H
610 NEXT H
620 NEXT H
630 NEXT H
640 NEXT H
650 NEXT H
660 NEXT H
670 NEXT H
680 NEXT H
690 NEXT H
700 NEXT H
710 NEXT H
720 NEXT H
730 NEXT H
740 NEXT H
750 NEXT H
760 NEXT H
770 NEXT H
780 NEXT H
790 NEXT H
800 NEXT H
810 NEXT H
820 NEXT H
830 NEXT H
840 NEXT H
860 FOR I = 1 TO 200
861 IF ABS(F1) < .00001 THEN 830
862 F1 = F * X + X = (F + X * (Q + Q * Q + 4 * F1 * X))
863 NEXT I
864 NEXT L
865 NEXT G
866 END

ORIGINAL PAGE IS OF POOR QUALITY

72
ORIGINAL PAGE IS OF POOR QUALITY
APPENDIX III

CYBER PLOTTING PROGRAMS.

PROGRAM GC(INPUT,OUTPUT,CSPR,TAPE3=CSPR)

PROGRAM GC IS USED TO PRODUCE SCATTER

PLTS THAT COMPARE C.S. WINDS DATA TO

F.R. WIND DATA. REQUIRES A FILE CSPR

THAT CONTAINS THE WIND VALUES FOR

"MAXI" VALUES AT EACH ALTITUDE. WILL

ASK FOR THE DATE; MONTH, DAY

EXTERNALS: ZETA / GRAB,ZETA

PLOT COMMAND: ZPLOTZ,TAPE26/LENGTH=70/TIME=2A

WRITTEN BY RAYMOND RUGGERIO 1982

DIMENSION CS(7,20),PR(7,20),X(20),Y(20),HP(8)

DATA HP/70.5,75.7,79.5,84.,88.5,93.,97.5,102./

PRINT 90

90 FORMAT("INPUT:MAXI,MTNH,TDAY")

READ *,MAXI,MTNH,TDAY

READ(3,40)((CS(I,J),PR(I,J),J=1,MAXI),I=1,7)

40 FORMAT(2F10.1)

CALL PLOTS(0.,0.,0.,26)

CALL PLOT(1.0,3.0,0.,3)

DO 150 IH=1,7

DO 120 =1,MAXI

X(IH)=PR(IH,M)

Y(IH)=CS(IH,M)

120 CONTINUE

SI=0

S2=0

SS1=0

SS2=0

DO 950 M2=1,MAXI

SI=SI+PR(IH,M2)

S2=S2+CS(IH,M2)

950 SI=SI+CS(IH,M2)*PR(IH,M2)

S2=S2+CS(IH,M2)*CS(lH,M2)

5X1=SI/MAXI

52=S2/MAXI

5S1=SI/MAXI

552=S2/MAXI

PHO=(SI*1S2)/SQR((SI-SI*SI)*(S2-S2*SS2))

PRINT *,IH,PHO

CALL AX1S(0.,0.,"PR VELOCITY",-11.4,0.,-100.,50.)

CALL AX1S(0.,0.,"CS VELOCITY",11.4,0.,-100.,50.)

CALL SYMBOL(2.0,4.51,105,"ALT=","0.,5)

CALL NUMBER(2.42,4.51,105,HP(IH),0.,3)

CALL SYMBOL(2.84,4.51,105,"KM","0.,3)

CALL SYMBOL(2.4,4.72,105,"DATE",0.,3)

CALL NUMBER(3.1,4.72,105,MTNH,0.,1)

CALL SYMBOL(3.31,4.72,105,"",0.,1)

CALL NUMBER(3.615,4.72,105,TDAY,0.,1)

CALL SYMBOL(3.65,4.72,105,"",0.,3)

CALL SYMBOL(2.4,4.3,105,"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
PROGRAM RECOV(INPUT,OUTPUT,WIN,ROUT,TAPE3=OUT,TAPE1=IN, 
+RCS,TAPE4=RCS)
C
PROGRAM RECOV IS USED TO PROCESS DATA
C TRANSFERRED TO THE CYBER FROM THE APPLE COMPUTER
C REQUIRES INPUT FILE TO HAVE THE
C NAME WIN WHICH CAN BE ACCOMPLISHED BY
C USING THE RENAME CMD: RENAME,WIN=FILEX
C PRODUCES OUTPUT FILES ROUT AND RCS
C ROUT IS THE INPUT DATA REFORMATTED AND SCREENED
C RCS IS THE MEAN WIND VALUES PROJECTED
C INTO C.S. LINE OF SITE
C TAPES99 IS THE PLOT COMMAND FILE PRODUCED
C IT CONTAINS THE INFORMATION TO PRODUCE THE
C PLOTS FOR NS,EN WIND AND CORRELATION TIME AND POWER
C EXTERNALS: ZETA / GRAB,ZETA
C PLOT FILE USING CMD: PLOTZ,TAPES99/LENGTH=170/TIME=30
C
C WRITTEN BY RAY RUGGERIO 1982
C
C
C DIMENSION VT(8,240),TT(8,240),AX(8,240),BX(8,240), 
+TN(8,240),EN(8,240),AX(8,240),VPY(8),RNS(8,240) 
+AW(8,2),BD(5),PA1(8,240),PA2(8,240),PA3(8,240),TS(8,240) 
+,VCS(8,240),VCSM(8)
C INTEGER TIME
C COMMON VT,TT,AX,BK,N,NS,EN,AR,AW
C CALL PLOTS(0.,0.,99)
C CALL PLOT(1.,1.,-3)
C
C
C RVT=200
C RNTT=3
C VMAX=200
C T5MIN=15.0
C DO 50 MI=1,240
C READ(1,10)(HD(I),I=1,4),TIME
C IF(EOP(I).NE.0)GO TO 900
C 10 FORMAT(4A1,1X,17)
C IF(HD(1).NE.""""""""""GO TO 800
C IF(MI.EQ.1)CALL TVC(TIME,TMTH,TDAY,THR,TMIN)
C DO 50 IH=1,8
C READ(1,*),IH,PA1(MI,M),PA2(MI,M),PA3(MI,M),TS(MI,M)
C IF(IH.NE.0)GO TO 60
C READ(1,*),VT(H,I),TT(M,I),AX(M,I),BX(M,I),VCS, 
+TN(M,I),NRM,EV
C IF(VP.GT.RVT)GO TO 60
C IF(RNT.GT.RNTT)GO TO 60
C IF(VT(M,I).GT.VMAM)GO TO 60
C IF(TS(M,I).LT.T5MIN)GO TO 60
C GO TO 50
C 60
C VT(M,I)=0
C TT(M,I)=0
C AX(M,I)=0
C BX(M,I)=0
C TN(M,I)=0
C 50 IF(T5(M,I).LT.0.0)T5(M,I)=0.0
C 900 MAXT=M1-1
C IF(MAXT.LT.2)GO TO 800
C DO 200 MI=1,MAXT
C RNS(IN,M)=VT(IN,M)*COS(.0174533*TT(IN,M))
C EN(IN,M)=VT(IN,M)*SIN(.0174533*TT(IN,M))
C AR(IN,M)=EX(IN,M)/10.0
200 CONTINUE
DO 500 IM=1,8
SI=0
C1=0
DO 520 N=1,MAXI
VGS(IN,H)=RMS(IN,H)*.58776*EM(IN,H)*.699
IF(VI(IN,H).EQ.0.)GO TO 520
C1=C1+1
SI=SI+VGS(IN,H)
520 CONTINUE
VGS(IN)=SI/C1
500 CONTINUE
WRITE(4,540)TIME
540 FORMAT(I1,I7)
WRITE(4,550)(VGS(IN,H),IN=1,8)
550 FORMAT(6F20.1)
WRITE(4,555)(VGS(IN,H),IN=1,8)
555 FORMAT(""/"" MEAN VALUES""/(F20.1))
CALL ADISP(T5,MAXI,Tenth,TDAY,THER,TMIN)
CALL PDISP(PA1,PA2,PA3,MAXI,TENT,THER,THER,THER,THER,THER)
WRITE(3,450)
450 FORMAT(I1,I2X,"MIN,VT,TH,EM,AR,TH,POWER,HALF TIME")
DO 300 H=1,8
DO 310 N=1,MAXI
WRITE(3,410)N,VT(IN,H),TH(IN,H),EM(IN,H),EM(IN,H)
*+,*AR(IN,H),TH(IN,H),PA1(IN,H),T5(IN,H)
310 CONTINUE
WRITE(3,420)
420 FORMAT(I1)
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 *,(VVF(J),J=1,8)
DO 987 IU=1,8
IF(VVF(IU).EQ.1.0)CALL ADISP(MAXI,IN,TENT,TDAY,THER,TMIN)
987 CONTINUE
CALL PLOT(0.,0.,999)
GO TO 911
800 WRITE(3,430)
430 FORMAT(" SEQUENCE ERROR")
911 STOP
END
SUBROUTINE ADISP(T5,MAXI,TENT,TDAY,THER,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,H).GT.2.0)TS(J,M)=0.0
CALL AXIS(0.,0.,15,TIMES IN MINUTES,-15.,6.,0.,10.)
CALL AXIS(0.,0.,13,HALITUDE (EM),13.,7.,90.,70.,5.,4.5)
XSC=1.5
YSC=1.5
CALL SYMBOL(3.,8.3.,105,"START TIME"",0.0.11)
CALL NUMBER(4.155,8.3.,105,"TH",0.0.1)
CALL SYMBOL(4.356,8.3.,105,"TH",0.0.1)
CALL NUMBER(4.468,8.3.,105,"TH",0.0.1)
CALL SYMBOL(3.,8.51.,105,"DATE",0.0.5)
CALL NUMBER(4.1,8.51.,105,"DATE",0.0.1)
CALL SYMBOL(4.31,8.51.,105,"DATE",0.0.1)
CALL NUMBER(4.615,8.51.,105,"DATE",0.0.1)
CALL SYMBOL(4.85,8.51.,105,"DATE",0.0.1)
CALL PLOT(-.0625,8.,2)
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",0.0.16)
CALL SYMBOL(25.,8.51.,105,"1.5 SEC",0.0.7)
DO 200 TH=1,8
CALL PLOT(0., 0., 3)
CALL PLOT(6., 0., 2)
CALL PLOT(6., 0., 3)
CALL PLOT(1./XSC,TA5(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 PROC(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 60 J=1, MAXI
IF(PA1(IH,J).GT.SP MAX)SP MAX=PA1(IH,J)
50 IF(SP MIN.GT.PA1(IH,J)) SP MIN=PA1(IH,J)
60 CONTINUE
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=FLOAT(IFIX(SP MAX))
CALL SYMBOL(3., 8.3, 1.05, "START TIME", 0., 11)
CALL NUMBER(4.155, 8.3, 1.05, THR, 0., -1)
CALL SYMBOL(4.365, 8.3, 1.05, ",", 0., 1)
CALL NUMBER(4.47, 8.3, 1.05, TMIN, 0., -1)
CALL SYMBOL(3., 8.51, 1.05, "DATE ", 0., 0.5)
CALL NUMBER(4.1, 8.51, 1.05, TMTH, 0., -1)
CALL SYMBOL(4.31, 8.51, 1.05, "/", 0., 1)
CALL NUMBER(4.415, 8.51, 1.05, TDAY, 0., -1)
CALL SYMBOL(4.05, 8.51, 1.05, "/ DAY", 0., 0.3)
CALL PLOT(-0.0625, 8., 3)
CALL PLOT(0.0625, 8., 2)
CALL PLOT(0., 8.2)
CALL PLOT(0., 9., 2)
CALL PLOT(-0.0625, 9., 2)
CALL PLOT(0.0625, 9., 2)
CALL PLOT(-0.0625, 9., 3)
CALL PLOT(0.0625, 9., 3)
CALL SYMBOL(3., 8.51, 1.05, "POWER", 0., 0.5)
CALL NUMBER(3., 25., 8.3, 105, YSC, 0., -1)
CALL SYMBOL(3., 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)/100000.)
DATA B(70.5,75.,79.5,14.,11.5,93.,97.5,102.1)
CALL AIS(0.,0.,15BTIMI,1MKTIMUTII,-15,6.,0.,0.,10.)
IF(WD.EQ.0.)CALL ms(o.,o.,""MOITH TO SOUTH VELOCITY (M/S)",29,1.,90.,-200.,50.)
IF(WD.EQ.1.)CALL ms(o.,0.,"WEST TO EAST VELOCITY (M/S)",+27,0.0,90.,-200.,50.)
XBC=10.
YBC=50.
CALL SYMBOL(3.,8.3,105,"ALT=","0.0,0.5)
CALL NUMBER(3.42,8.3,105,HP(I),0.,1)
CALL SYMBOL(3.84,8.3,105,"KM","0.0,5")
CALL SYMBOL(3.,8.51,105,"START TIME=","0.0,11")
CALL NUMBER(4.153,8.51,105,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,"/S2","0.0,3")
CALL PLOT(0.0,4.0,-3)
DO 10 J=1,KAXI
IF(Y(J).EQ.0.0)GO TO 10
CALL SYMBOL(X(J)/XSEC-.054,Y(J)/YSEC-.054,11,"+",0.0,1)
10 CONTINUE
CALL PLOT(8.5,-4.0,-3)
RETURN
END
SUBROUTINE AVER(KAXI)
DIMENSION VT(8,240), TT(8,240) ,AI(8,240), D(8,240),
+INS(8,240) ,EW(8,240) ,AI(8,240) ,AW(1,2) ,Bl(8,240)
COMMON VT, TT,AX,BX,
+TH,RNS,EW,AI,AW
INTEGER 8, SPF(8)
DO 10 8=1,S
10 SPF(R)-C1
AW(B,1)-Sl/el
10 AW(R,2)-82/C1
WRITE(3,700)(AW(R,1),AW(H,2),SPF(H),B-1,S)
PRINT 700,(AW(H,1),AW(H,2),SPF(H),H=1,8)
700 FORMAT(1H11/2I," MEAN WIND VALUES NS AND EW "/(2F10.1,5I,I5))
RETURN
END
SUBROUTINE DISPW(KAXI,B,tKTH,TDAY,Tal,THIN)
DIMENSION X(240),Y(240), VT(8,240),TT(8,240) ,AX(8,240),
+EX(8,240),TH(8,240),RNS(8,240),EM(8,240),AX(8,240),EM(8,240),AX(8,240),
COMMON VT,TT,AX,BX,TH,RNS,EM,AX,AW
INTEGER H,SPF(H)
DO 10 J=1,MAXI
X(J)-J
10 Y(J)-RN8(H,J)
WD-O.O
CALL DISPW2(X,J,MAXI,WD,H,THTH,TDAY,THR,THMIN)
DO 20 J=1,MAXI
Y(J)-EM(H,J)
WD=1.0
CALL DISPW2(X,J,MAXI,WD,H,THTH,TDAY,THR,THMIN)
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 taping records, Instruction manual for ionospheric drift measurements.


Briggs, B. H. (1976), The physical significance of the correlation ellipse in ionospheric drift analysis, Radio Sci., 11, 817-819.


